BIOINFORMATICS METHODS FOR THE GENETIC AND MOLECULAR CHARACTERISATION OF CANCER

Dissertation zur Erlangung des Grades des Doktors der Naturwissenschaften der Naturwissenschaftlich-Technischen Fakultäten der Universität des Saarlandes

VON
DANIEL STÖCKEL

Universität des Saarlandes,
MI – Fakultät für Mathematik und Informatik,
Informatik

BETREUER
PROF. DR. HANS-PETER LENHOF

6. DEZEMBER 2016

Daten der Verteidigung

Datum	25.11.2016, 15 Uhr ct.
Vorsitzender	Prof. Dr. Volkhard Helms
Erstgutachter	Prof. Dr. Hans-Peter Lenhof
Zweitgutachter	Prof. Dr. Andreas Keller
Beisitzerin	Dr. Christina Backes
Dekan	Prof. Dr. Frank-Olaf Schreyer

This thesis is also available online at:
`https://somweyr.de/~daniel/dissertation.zip`

ABSTRACT

Cancer is a class of complex, heterogeneous diseases of which many types have proven to be difficult to treat due to the high genetic variability between and within tumours. To improve therapy, some cases require a thorough genetic and molecular characterisation that allows to identify mutations and pathogenic processes playing a central role for the development of the disease. Data obtained from modern, biological high-throughput experiments can offer valuable insights in this regard. Therefore, we developed a range of interoperable approaches that support the analysis of high-throughput datasets on multiple levels of detail.

Mutations are a main driving force behind the development of cancer. To assess their impact on an affected protein, we designed *BALL-SNP* which allows to visualise and analyse single nucleotide variants in a structure context. For modelling the effect of mutations on biological processes we created *CausalTrail* which is based on causal Bayesian networks and the do-calculus. Using *NetworkTrail*, our web service for the detection of deregulated subgraphs, candidate processes for this investigation can be identified. Moreover, we implemented *GeneTrail2* for uncovering deregulated processes in the form of biological categories using enrichment approaches. With support for more than 46,000 categories and 13 set-level statistics, GeneTrail2 is the currently most comprehensive web service for this purpose. Based on the analyses provided by *NetworkTrail* and *GeneTrail2* as well as knowledge from third-party databases, we built *DrugTargetInspector*, a tool for the detection and analysis of mutated and deregulated drug targets.

We validated our tools using a Wilm's tumour expression dataset and were able to identify pathogenic mechanisms that may be responsible for the malignancy of the blastemal tumour subtype and might offer opportunities for the development of novel therapeutic approaches.

ZUSAMMENFASSUNG

Krebs ist eine Klasse komplexer, heterogener Erkrankungen mit vielen Unterarten, die aufgrund der genetischen Variabilität, die zwischen und innerhalb von Tumoren herrscht, nur schwer zu behandeln sind. Um eine bessere Therapie zu ermöglichen ist daher in einigen Fällen eine sorgfältige, genetische und molekulare Charakterisierung nötig, welche es erlaubt die Mutationen und pathogenen Prozesse zu identifizieren, die eine zentrale Rolle während der Krankheitsentwicklung spielen. Daten aus modernen, biologischen Hochdurchsatzexperimenten können hierbei wertvolle Einsichten liefern. Daher entwickelten wir eine Reihe interoperabler Ansätze, welche die Analyse von Hochdurchsatzdatensätzen auf mehreren Detailstufen unterstützen.

Mutationen sind die treibende Kraft hinter der Entstehung von Krebs. Um ihren Einfluss auf das betroffene Protein beurteilen zu können, entwarfen wir die Software *BALL-SNP*, welche es erlaubt einzelne Single Nucleotide Variations in einer Kristallstruktur zu visualisieren und analysieren. Um den Effekt einer Mutation innerhalb eines biologischen Prozesses modellieren zu können, erstellten wir *CausalTrail*, das auf kausalen bayesschen Netzwerken und dem do-calculus basiert. Unter der Verwendung von *NetworkTrail*, unserem Web-Service zur Detektion deregulierter Subgraphen, können Prozesse identifiziert werden, die als Kandidaten für eine solche Untersuchung dienen können. Zur Detektion deregulierter Prozesse in der Form von biologischen Kategorien mittels Enrichment-Ansätzen implementierten wir *GeneTrail2*. *GeneTrail2* unterstützt mehr als 46.000 Kategorien und 13 Statistiken zur Berechnung von Enrichment-Scores. Basierend auf den Analysemethoden von *NetworkTrail* und *GeneTrail2*, sowie dem Wissen aus Drittdatenbanken konstruierten wir *DrugTargetInspector*, ein Werkzeug zur Detektion und Analyse von mutierten und deregulierten Wirkstoffzielen.

Wir validierten unsere Werkzeuge unter Verwendung eines Wilm's Tumor Expressionsdatensatzes, für den wir pathogene Mechanismen identifizieren konnten, die für die Malignität des blastemreichen Subtyps verantwortlich sein können und bisweilen die Entwicklung neuartiger, therapeutischer Ansätze ermöglichen könnten.

ACKNOWLEDGEMENTS

The thesis you hold in your hands marks the end of a six year journey. When I started my PhD, a fledgling Master of Science, I thought I knew everything about science. Surely I would be done in three years tops and surely my work would be nothing but revolutionising. All the others were certainly just slacking off! Oh, how naïve I was back then! And with the passage of time I learned a lot. Some of the things I learned concerned science, while other did not. I learned that succeeding as a researcher not only takes a lot of theoretical knowledge, but also a lot of dedication and hard work. I learned that the solution to some problems cannot be rushed, but instead requires many small and deliberate steps. But most importantly, I learned that science is not an one man show, but a team effort. Because of this I would like to take the time to thank the persons without which this thesis would not have been possible.

First of all, I would like to thank my advisor Prof. Dr. Hans-Peter Lenhof for giving me the opportunity to work at his chair. Hans-Peter also gave me the freedom to independently pursue the research topics I was interested in. If, however, a problem arose he would selflessly sacrifice his precious time to help with finding a solution. For this I am deeply grateful. Next, thanks is in order for Prof. Dr. Andreas Keller for reviewing my thesis, bringing many interesting research opportunities to my attention and the energy and enthusiasm he is infusing into the people that have the pleasure to work with him.

Naturally, this work would have been impossible without my family and, for obvious reasons, without my parents Thomas and Sabine Stöckel. It was them who woke my interest in science and technology. As long as I can remember they supported and encouraged me to pursue the career I have chosen for myself and, thus, without them I would not have been able to achieve what I achieved. Similarly, I would like to thank Charlotte De Gezelle for her patience whenever I postponed yet another vacation because: "I am done soon™!" Whenever I would become frustrated and needed a shoulder to cry on I could rely on her having an open ear for me. You truly have made my world a brighter place!

Of course, there is another group of people who's importance I cannot stress enough. My proofreaders Lara Schneider, Tim Kehl, Fabian Müller, Florian Schmidt, and Andreas Stöckel fought valiantly against spelling mistakes, layout issues, and incomprehensible prose. Thank you for setting my head straight whenever I was again convinced that what I wrote was perfectly clear. Hint: it wasn't. Equally, I would like to thank my coworkers Alexander Rurainski, Christina Backes, Anna-Katharina Hildebrandt, Oliver Müller, Marc Hellmuth, Lara Schneider, Patrick Trampert, and Tim Kehl for the pleasant and familial working environment. Much of the work presented in this thesis has been a joint endeavour in which they had integral parts. Similarly, I would like to

praise all the awesome people from the Saarland University, as well as the Universities Mainz and Tübingen that I had the pleasure to work with. A special shoutout goes to my colleagues and good friends from the MPI for Computer Science for the many enjoyable discussions during lunch. A thank you is also in order for my students, although I am not certain whether they enjoyed being taught by me as much as I enjoyed the opportunity to teach them. Finally, I am grateful for all my friends and the time we spent together playing board games, celebrating, or simply hanging out. Without you, I would certainly not have managed to pull through and finish this thesis.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF NOTATIONS

Sets and intervals

$\mathbb{N}$	The set of natural numbers $\{1, 2, 3, \ldots\}$. Inclusion of zero is indicated by an explicit subscript $\mathbb{N}_0$.
$\mathbb{R}$	The set of real numbers. Subsets such as the set of positive numbers and the set of non-negative numbers are indicated using super- and subscripts e.g. $\mathbb{R}_0^+$.
$\mathbb{B}$	The set of boolean numbers $\{0, 1\}$.
$\lvert A \rvert$	The number of elements in a (finite) set.
$\mathcal{P}(X)$	The power set of the set X.
$A \setminus B$	The set difference between A and B.
A^c	The complement of set A.
$A \subset B$	A is a proper subset of B.
$A \subseteq B$	A is equal to or a subset of B.
$[a, b] \subset \mathbb{R}$	The closed interval between a and b.
$(a, b) \subset \mathbb{R}$	The open interval between a and b.
$[a, b)$, $(a, b]$	Half-open intervals between a and b.

Linear Algebra

$\vec{v} \in X^n$	A n-dimensional (column) vector with name v.
$\boldsymbol{A} \in X^{n \times m}$	A matrix with n rows and m columns.
$\boldsymbol{A}^t$, $\vec{v}^t$	The transposed of the matrix $\boldsymbol{A}$ and vector $\vec{v}$.
$\langle \vec{u}, \vec{v} \rangle$	The dot product between the vectors $\vec{u}$ and $\vec{v}$

Probability Theory

$\alpha \in (0, 1)$	A confidence level used in a statistical test.
θ	A set of unspecified parameters of a statistical model.
μ	The expected value of a probability distribution.
$\bar{x}$	The sample mean.
σ, σ^2	The standard deviation and variance of a probability distribution.
s, s^2	The sample standard deviation and variance.
Ω	The set of results in a probability space.

Σ	The σ-algebra of events in a probability space.
$\Pr(X = x)$, $\Pr(x)$	The probability that the random variable X takes on the value x.
X, $\boldsymbol{X}$	A random sample or a matrix of random samples. In the latter case the rows represent the samples and the columns represent measured variables. The number of rows and columns are represented by n and p.
$\Sigma \in \mathbb{R}^{n \times n}$	The covariance matrix.
$\Omega := \Sigma^{-1}$	The precision matrix.
$\boldsymbol{S} \in \mathbb{R}^{n \times n}$	The sample covariance matrix.

1 INTRODUCTION

You can have data without information, but you cannot have information without data.

— DANIEL KEYS MORAN

Anthropologists classify eras of human history according to defining technologies, concepts, or resources that have irreversibly shaped societies during those times. For example, we refer to ancestral periods as the stone, bronze or iron age. Later periods became known as the "age of enlightenment" or the "industrial revolution". While it is unknown how future generations will refer to our times, a reasonable proposition seems to be that, today, we live in the "age of information". Via the internet, an unprecedented amount of knowledge is open to more humans than ever before. The omnipresence of networked electronic devices allows to capture profiles of our everyday lives. Social interactions, shopping preferences, location data, and other information is routinely stored in tremendous quantities (cf. [McA+12]). How to make use of this "Big Data", for better or worse, remains an open question. Whatever will turn out to be the answer to this question, it will likely redefine every aspect of our modern society. Science is no exception. While natural scientists were among the first users of computers, the amount of captured experimental data remained at manageable scales for a long time; with the notable exception of various high-profile physics projects [BHS09]. However, this has changed during the past decade. In biology, thanks to the development of ever more potent high-throughput methods, the size of recorded data sets has increased dramatically. It is possible to capture complete genomes, transcriptomes and sizable parts of both, the proteome and the metabolome with a single experiment each. Projects like ENCODE [ENC04], DEEP[1], TCGA [McL+08], or 1000 Genomes [10010] have collected vast archives of high-throughput datasets.

1 exabyte (EB) = 1000 petabytes (PB) = 10^{18} bytes

Stephens et al. [Ste+15] estimate that by 2025 data from genomics alone will require 2 EB to 40 EB of storage. This far exceeds the projected requirements of social platforms such as YouTube (1 EB to 2 EB), Twitter (1 PB to 17 PB), but also of applications from astronomy such as the data captured by telescopes (1 EB). It is thus fair to say that biology has arrived in the realm of "Big Data". Generating a large amount of information is, however, only a prerequisite for understanding the biological processes that take place in an organism. The determination of the human genome's sequence [Lan+01; Ven+01] already showed conclusively that the main challenge posed by an open biological problem is not necessarily the data generation process, but rather the analysis of the generated measurements. Similarly, making sense of and interpreting the data repeatedly proofs to be the main bottleneck in biological high-throughput experiments. For example, knowing the expression

1 `http://www.deutsches-epigenom-programm.de/`

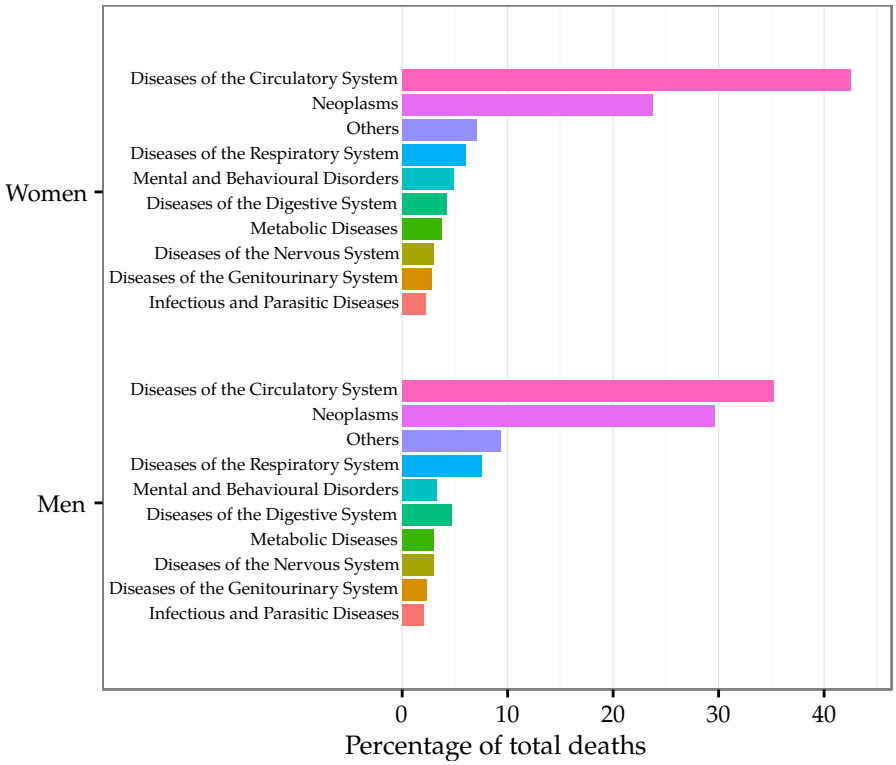


Figure 1.1: Most common causes of death in Germany in 2014. The classification into categories is according to the ICD-10 definitions. Neoplasms (cancer) are the second most common cause of death after diseases of the circulatory system and amounts for ≈ 25 % to 30 % of all deaths [Sta16].

value of a single gene at one or even multiple points in time hardly offers any insight into the current state of the cellular program [Gyg+99; Gre+03]. Instead, the relationships between genes, transcripts, proteins, metabolites, and how they interact, need to be elucidated to be able to truly understand the molecular mechanisms of a single cell and, ultimately, an organism. Thus, a primary goal of computational biology is to unravel the relationships between the various measured entities, given the available wealth of information.

1.1 MOTIVATION

In bioinformatics a main driver behind the development of better data analysis methods are complex, heterogeneous diseases that have time and again proven to be especially difficult to treat without detailed genotypic and phenotypic knowledge. The prime example for a class of such diseases is cancer. In 2014, roughly 25 % to 30 % of the deaths recorded in Germany [Sta16] could be attributed to malign neoplasms (Figure 1.1). Despite intensive research efforts over the last decades, the timely detection of a tumour remains as one of the most effective weapons in the fight against cancer. This is predicated by the nature of tumourigenesis. Most tumours develop due to (epi-)genetic aberrations. These aberrations are governed by a stochastic process and

It should be noted that, due to the heterogeniety of cancer, "neoplasms" is an extremely broad class of diseases, whereas "diseases of the circulatory system" is comparatively narrow.

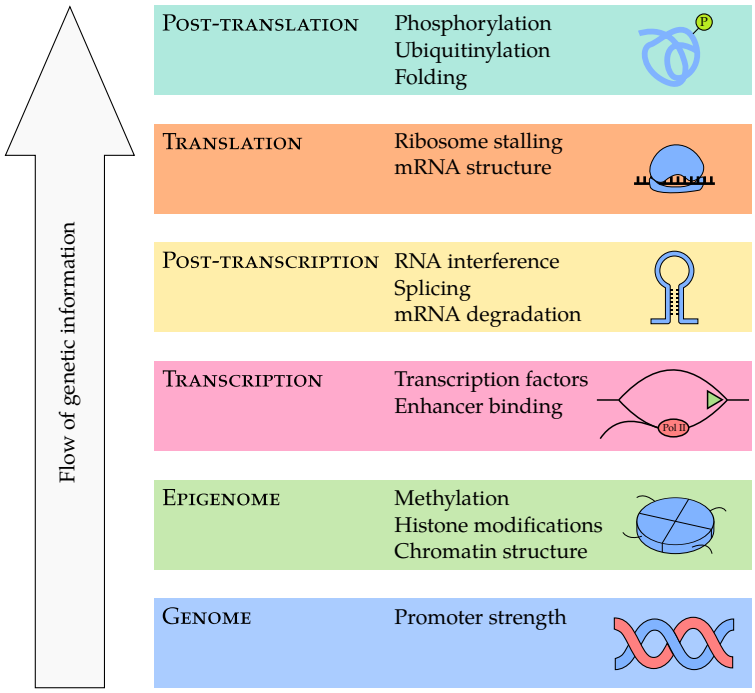


Figure 1.2: Levels of cellular regulatory mechanisms. The flow of genomic information is from bottom to top. At each level one or more regulatory mechanisms exist that can prevent (or enable) the flow of information to the next level. Regulatory mechanisms at the lower levels tend to take longer to come into effect but are also more persistent.

the likelihood with which they occur depends on a variety of patient-specific factors that include, but are not limited to environmental influences, lifestyle, and genetic predisposition. As a result, every case of cancer is unique. Even tumours within the same histopathological type can display substantially variability. In fact, due to the uncontrolled proliferation of cancer cells and thus the increased likelihood of accumulating additional mutations, each tumour is composed of multiple cancer cell populations that all carry a unique set of genetic aberrations [Sha+09; Wei13]. Thus, each case of cancer must be carefully examined individually to ensure an "optimal" therapy [Jai05]. Hence, it is necessary to thoroughly characterise the tumour on a molecular level to allow biologists and clinicians to decipher the prevalent pathogenic mechanisms and, accordingly, to decide on the best available treatment.

High-throughput assays, as mentioned above, are promising technologies for collecting data on which the analysis of a tumour sample can be based. However, due to the size and noisy nature of the captured data, efficient and robust statistical analysis software is needed for working with this information. In this regard, one of the main challenges to overcome is the variability of and within the tumour. Current data sets commonly contain measurements from a mixture of cells. This disregards the genetic differences that exist in a population of tumour cells. Recent advances in sequencing technology suggest that in the near future it will be possible to obtain data for a sizeable popu-

Longer-term developments such as time-series data with a single-cell-resolution are also imaginable.

lation of single cells together with their spatial location in a tumour biopsy [Ang+16; WN15; Uso+15]. How to handle this kind of data effectively, however, currently is an open research problem.

Furthermore, it is important to note that although a single high-throughput method is able to capture comprehensive datasets for one so-called *omics* type, the collected information only provides a small glimpse at the pathogenic processes that actually take place in a tumour. This is due to the fact that many of the cellular processes, and thus the emergence of complex traits, are subject to a tight regulatory program constraining the flow of genetic information (Figure 1.2). This program manifest as interactions between genes, proteins, miRNAs, and other biological entities that can be modelled as biological networks, where edges represent interactions and nodes represent interaction partners. An integrative analysis of data from multiple omics is, therefore, mandatory to gain deeper insights into the deregulated processes that drive tumour development. Performing such an analysis necessitates theoretical models and software that, in addition to the above requirements, can also cope with heterogeneous data sets.

1.2 OVERVIEW

While we describe our methods with regard to their applicability in tumour research, most of them can be used to investigate arbitrary datasets.

This dissertation introduces tools and methods we developed for the analysis of high-throughput data. Our main goal is to support the genetic and molecular characterisation of tumour samples. Commonly, this entails an iterative process in which each step provides information that is used to guide further, more focused examinations. To support such workflows, the approaches we conceived are each targeted at different levels of detail: some methods are especially suited to quickly narrow down large amounts of data into a set of interesting systems. These can then be analysed using more specialised software. Naturally, this requires that the employed tools are interoperable, which was an important concern we considered when designing our approaches. Finally, as outlined above, the ability to draw on knowledge from heterogeneous datasets is essential for understanding complex diseases. Thus, we put special emphasis on methods that allow to perform integrative analyses on multi-omics data.

Throughout this thesis, we illustrate the capabilities of the tools we developed using a Wilm's tumour dataset (Section 2.4). Wilm's tumours are childhood renal tumours that, due to various properties such as the relative independence from environmental influences and a comparatively low mutation rate, are an ideal model disease. The main goal of our investigation was the detection of pathogenic mechanisms that may explain the increased malignancy of blastemal subtype tumours compared to other Wilm's tumour subtypes. On the basis of this case study, we highlight contributions that may be suitable for the creation of systems that assist researchers and physicians in devising effective, personalised cancer treatments.

The remainder of this work is structured as follows: in Chapter 2 the required biological background and experimental techniques are intro-

duced. To this end, a basic discussion of cancer biology is provided. Accompanying this general discussion, we describe Wilm's tumours in more detail. Furthermore, we discuss current treatment options and, especially, the trend towards targeted therapy. Based on this, we motivate the need of advanced, computational methods to enable a more accurate, personalised medicine. Afterwards we introduce some of the biological assays that are available to create detailed, biological patient profiles and discuss their properties, advantages, and drawbacks.

An important prerequisite for personalised treatments is that the pathogenic processes that play a central role for a given tumour have been identified. As biological networks are a natural way to model regulatory and other processes, Chapter 3 introduces a set of methods for their analysis, along with some of the most common network types. In particular, we introduce *CausalTrail* which we implemented to assess the causal dependencies in regulatory cascades. Using predefined causal Bayesian networks and the do-calculus, CausalTrail is able to assess the impact of a mutation or a drug on nodes down-stream of a target regulator. For the end user, both, a command line application as well as a graphical user interface are provided. With CausalTrail we developed, to the best of our knowledge, the first, freely available tool for computing the effect of interventions in a causal Bayesian network structure.

While many biological networks stem from network databases, approaches for inferring a topology directly from data exist. Here, we take a look at the theory behind *Gaussian graphical models (GGMs)*, which can be used to infer a network representing the partial correlation structure between a set of variables. While we will not use GGMs for network inference purposes, we later explore their applicability for enrichment analyses (Section 4.4).

A common characteristic of cancer cells is that their regulatory networks have been reprogrammed. To detect affected parts of this network, methods for the search of deregulated subgraphs can be employed. Deregulated subgraphs are small, usually connected parts of a biological network that contain a large number of e.g. differentially expressed genes. In Section 3.5, we discuss our *integer linear programming (ILP)* formulation for discovering deregulated subgraphs in regulatory biological networks [Bac+12]. In contrast to many competing methods, our approach computes an exact solution to the *rooted maximum-weight subgraph* problem. Furthermore, we describe the first approach for quantifying the influence of the network topology on the detected subgraphs using a combination of sampling and machine learning techniques. Our study underlines the need and provides the basis for the development of a rigorous framework that allows to determine the significance of a deregulated subgraph. We later revisit the ILP formulation during the introduction of the NetworkTrail web service in Section 5.4.

In Chapter 4, we turn to approaches that are closely related to the detection of deregulated subgraphs. While the aforementioned more or less are free to choose any set of connected genes from an input network, *enrichment methods* rely on predefined categories of biological

entities to detect deregulated pathways. To be able to discuss these algorithms, we first provide an introduction into the theory behind statistical hypothesis tests. Next, a general framework for enrichment analysis is presented. Guided by this framework, we discuss some popular enrichment methods in detail. Additionally, we perform an evaluation of the presented algorithms. Based on its results, we derive a set of guidelines for choosing the appropriate enrichment method for a specific set of input data. Finally, an alternative scheme for performing enrichment analyses based on Hotelling's T^2-test [Hot31] is introduced and evaluated with respect to its applicability in practice.

Providing and maintaining native, high-performance software that works on multiple operating systems and computer configurations can place a substantial burden on a development team. Web services allow to circumvent this problem by providing a centrally managed installation of a given tool on which users can rely. To facilitate the construction of such web services, we created the *Graviton* platform (Chapter 5). Services based on Graviton automatically are equipped with user session management, a self-documenting workflow system, and are scriptable via a RESTful application programming interface. Furthermore, the platform provides a comprehensive collection of algorithms and data structures that offer solutions to tasks commonly encountered by bioinformatics tools, such as identifier mapping and the parsing of input data. Using Graviton, we created the web services *GeneTrail2*, *NetworkTrail*, and *DrugTargetInspector (DTI)* which facilitate the analysis of multi-omics datasets.

With GeneTrail2, we implemented a web service for conducting enrichment analyses on multi-omics datasets (Section 5.3). GeneTrail2 implements most of the enrichment methods presented in Chapter 4 and is, at the time of writing, the most comprehensive web service for enrichment analysis in existence. It supports data from 14 organisms, integrates information collected from over 30 databases and offers novel ways for visualising computed enrichments. Using GeneTrail2, we were able to detect molecular mechanisms that may explain the increased malignancy of blastemal subtype Wilm's tumours when compared to other subtypes.

In order to complement our web service for enrichment analysis, we created NetworkTrail, a web service for detecting deregulated subgraphs in regulatory networks (Section 5.4). It provides a user-friendly web interface to our ILP formulation [Bac+12] as well as the FiDePa [Kel+09] algorithm that uses ideas from enrichment analysis to search for the most deregulated path of a predefined length.

A common problem in cancer therapy is that tumours develop a resistance against anti-cancer drugs. For example, the proteins that are targeted by a drug may no longer be functional due to mutations and, thus, the drug will have no effect. We built DTI [Sch+15] with the goal of assisting with the stratification of treatment options based on user provided gene expression and mutation data (Section 5.5). In particularly, it allows the detection, annotation, and analysis of deregulated as well as mutated drug targets. For this, the knowledge from DrugBank [Law+14], Variant Effect Predictor [McL+10], and experimental

resistance data from the GDSC project [Bar+12] has been integrated into DTI. By combining this information with the input data, an overview of potential, deregulated drug targets as well as promising treatment options is generated. By leveraging the deep integration of DTI into GeneTrail2 and NetworkTrail, which is made possible by Graviton, further advanced analyses can be started by the user with a single click.

For gauging the effect of a drug on a mutated target, it is essential to take the structure of the affected protein into consideration. To complement DTI in this regard, we created *BALL-SNP* [Mue+15]. It allows to examine the effect of point mutations by visualising them in and performing analyses on the respective crystal structure. To identify possible collaborative effects between single nucleotide variants, BALL-SNP allows to detect point mutations that are in close spatial proximity by performing cluster analyses. Furthermore, BALL-SNP integrates the output of *in silico* methods for predicting the impact of a mutation. To build the software, we relied on the *PresentaBALL* (cf. Section 6.3) infrastructure offered by the *BALLView* molecular viewer [Mol+05; Mol+06], which is part of the BALL library [KL00; Hil+10]. PresentaBALL was created specifically with the idea in mind to make the functionality offered by BALL available for a wider range of applications such as interactive installations, teaching, and the presentation of research results. In a similar vein, we integrated BALL and BALLView with the Galaxy workflow system [Goe+10; Hil+14a]. For this, we created a suite of command line tools, built on top of BALL, for working with structure data.

Finally, Chapter 7 closes with a summary and discussion of the described work. In particular, we provide a perspective on possible, further developments in the presented fields.

CONTRIBUTIONS Research projects in bioinformatics often require the expertise and the effort of more than one person to be successful. Due to this, I cannot claim the exclusive authorship for all of the work presented in this thesis. To ensure that all contributions can be attributed fairly, sections that discuss shared work are prefaced with a short list of the main contributors. More detailed information can be found in the author lists and the contributions sections of the respective publications.

2 BIOLOGICAL BACKGROUND

If you try and take a cat apart to see how it works, the first thing you have on your hands is a non-working cat.

— DOUGLAS ADAMS, the salmon of doubt (2002)

During the last century the life expectancy in developed countries has increased dramatically. Between 1970 and 2010 alone, the life expectancy at birth has increased by roughly ten years in Germany (see Figure 2.1). This change can be attributed to improved living conditions due to significant advances in technology, sanitation, and medicine [Ril01]. For example, the introduction of *cyclosporin* into clinical practice allowed more reliable organ transplants [Cal+79]. Imaging techniques like *magnetic resonance imaging (MRI)* and *computer tomography (CT)* made it possible to capture high-resolution images from otherwise difficult to reach regions of the body [Bec14]. The invention of genetically modified organisms [MH70] allowed the production of human insulin for the treatment of diabetes [Joh83]. During this period, smallpox were eradicated due to a rigorous vaccination campaign [Fen93]. Thanks to the discovery of antiretroviral drugs and effective treatment regimens, HIV positive patients at age 20 can expect to live another 30 years or more [ART08]. Owing to this development, many of the once common, deadly diseases are no longer an issue in the 21st century. Still, modern medicine suffers from certain flaws and shortcomings. Through the widespread use of antibiotics, resistant strains are beginning to form that are only treatable with great difficulties [Neu92; Nik09]. This development has lead to the return of old killers, such as tuberculosis, in some countries [Kam95]. Diseases like SARS [Cho+04], bird flu [Pei+04], or Ebola [Tea+15] that spread quickly via aerosols or body fluids have claimed many victims, especially in lesser developed countries. Also, genetic diseases such as *Huntington's disease* [Wal07] or *cystic fibrosis* [Rio+89] pose challenges when it comes to finding a cure. In particular, many tumour types have eluded effective therapy for decades, despite remarkable advances that have been made in cancer research.

In 1983, the human immunodeficiency virus (HIV) had just been discovered [Mon02] and has, since then, claimed millions of lives [HIV+14].

Reasons for these problems are manifold. In the case of most viral infections only few effective antiviral drugs are available [De 04]. For influenza, vaccinations exist that, however, only protect against a limited set of viral strains [Bri+00]. For Ebola, no approved vaccinations exist [Gal+14], although several candidates are in the drug development pipeline at the time of writing [Agn+16]. In the case of genetic disorders the cause of the disease is not an external influence, but instead is encoded in the patient's genes. Though a set of best practices has been established for a large number of cancer types (cf. Figure 2.4), the treatment for certain tumours needs to be decided on a case by case basis due to their heterogeneity. To combat the above diseases, it is necessary

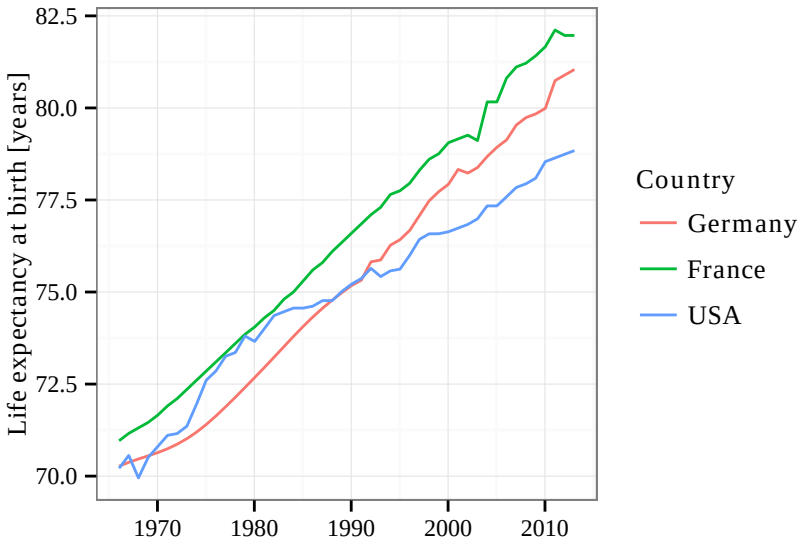


Figure 2.1: Life expectancy at birth in years. Between 1966 and 2013, life expectancy at birth in Germany increased from 70 to 81 years. *Source: World Bank* `http://data.worldbank.org/indicator/SP.DYN.LE00.IN`. *Retrieved: 2015-12-03*

to be able to quickly identify and analyse the predominant pathogenic mechanisms. Based on this information, a treatment that has been specifically tailored towards the patient needs to be devised. Ideally, this would entail the possibility to synthesise novel drugs if necessary.

Whilst we are, technologically and ethically, far away from being able to create new drugs on demand, considerable advances have been made towards such a personalised medicine (cf. Section 2.2). In particular, improved experimental methods for data generation provide an essential building block for reaching this goal. For example, the analysis of DNA/RNA samples is more and more becoming a routine procedure in biological laboratories and has started to enter clinical practice [BKE11; Bie12; KGH12]. Methods like cDNA microarrays (Section 2.3.2) and high-throughput sequencing (Section 2.3.3) allow to analyse the genetic state of tissue samples in great detail. Recent research suggests that reliably analysing the mRNA abundances within individual cells is not too far off in the future [Naw14], promising e.g. the availability of high resolution profiles of tumour biopsies [NH11]. Other techniques such as mass spectrometry have enabled capturing sizeable parts of the metabolome [DBJ05] and proteome [WWY01]. For the latter, it is possible to detect protein modifications such as phosphorylations and ubiquitinations that play an important role in the regulatory circuitry of our cells [Man+02]. Given this data, it should be possible to conceive more thorough ways of performing personalised medicine. However, the analysis of high-throughput data proves to be a difficult task from both a practical as well as a theoretical point of view. To understand some of these issues, this chapter gives an overview of the relevant biological background. In addition we will discuss some of the ex-

The techniques presented in this thesis are generally applicable and not tied to cancer specifically.

perimental methods that are frequently encountered throughout this thesis.

We use cancer as a model disease to illustrate the capabilities of our methods. There are various reasons for this choice: first, the disease is of great clinical, but also social and economical relevance. Second, a vast corpus of experimental data derived from tumour biopsies has been compiled and is publicly available. And third, although cancer research has progressed tremendously over the last decades, many fundamental aspects of the acting pathogenic mechanisms remain unresolved. Thus, studying cancer not only serves the improvement of therapy, but also promises to yield exciting biological insights.

We start with a discussion of cancer biology in general and the properties of Wilm's tumours specifically. We then proceed to a more precise definition of personalised medicine and outline the problems that need to be solved for implementing personalised medicine schemes. Finally, we introduce the basics of gene expression and discuss the *cDNA microarray* as well as the *RNA-seq* technology for capturing expression profiles.

2.1 CANCER

Much of the information in this section is taken from "The Biology of Cancer" by Weinberg [Wei13]. Additional sources are quoted explicitly.

Cancer is a heterogeneous class of diseases characterised by the abnormal growth of tissue. Causes for the development of cancer can be sought in many factors including life-style, exposure to pathogens and radiation, mutations or epigenetic alterations. Ultimately, these factors result in a disruption of the regulatory circuitry of previously healthy cells, which transforms them into tumour cells.

Unless specified otherwise the term "mutation" is used in its broadest meaning: a change or aberration in the genome.

Somatic mutations are a well researched mechanism that leads to the formation of tumour cells. While mutations are usually detected and neutralised by internal cellular controls or the immune system, some mutations go unnoticed. Interestingly, only few so-called *driver mutations* are believed to be sufficient for establishing cancer [SCF09]. Examples are mutations that lead to the transformation of "normal" genes to oncogenes: genes that have the potential to induce cancer. One of the first reported transformation mechanisms is the activation of the T24 oncogene in human bladder carcinomas due to the exchange of a single nucleotide [Red+82]. Once driver mutations have established a foothold, additional mutations are accumulated.

Genes that can be transformed to oncogenes are called proto-oncogenes.

The definitions here pertain to solid tumours. However, many properties such as tumour heterogeneity directly carry over to non-solid tumours such as leukaemia.

As mentioned above, cancer and especially cancer cells are characterised by their ability to form new tissue and, thus, are able to freely divide. Two non-exclusive hypotheses explain this behaviour. Either, cancer cells derive from stem cells or they dedifferentiate from mature cells in order to obtain stem cell like characteristics [Sel93]. The newly formed tissue is denoted as *neoplasm* or *tumour*. Not every neoplasm is immediately life threatening. For example moles, also known as *melanocytic nevi* are pigmented neoplasms of the skin that are, usually, harmless. To reflect this, neoplasms are often classified as either benign or malignant. Whereas benign tumours grow locally and do not invade adjacent tissues, malignant tumours commonly invade into

The term neoplasm can be directly translated as "new tissue".

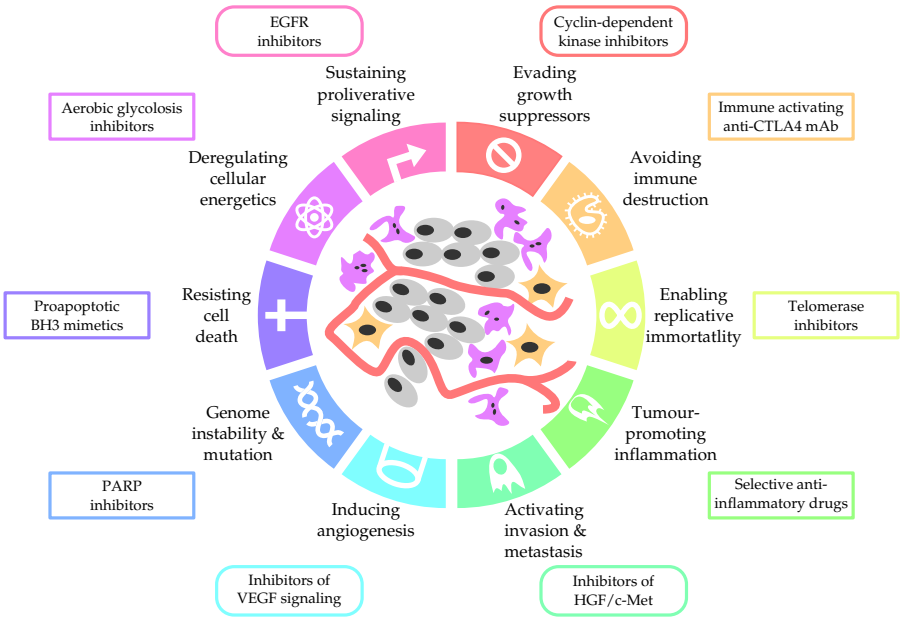


Figure 2.2: The hallmarks of cancer as proposed by Hanahan and Weinberg [HW00]. The authors postulate that during tumourigenesis normal cells successively acquire the "hallmark stages" as they evolve towards a malignant tumour. Coloured boxes indicate treatment options for counteracting the molecular changes constituting the respective hallmark.

nearby, healthy tissues and may also able to spawn new tumours, so called *metastases*, at different locations.

In its development from healthy to neoplastic tissue, cancer cells go through an evolution that allows them to grow successfully in a host organism. This evolution is driven by selection pressure that is, for example, exerted by the immune system, the competition for nutrients with surrounding cells, and eventual anti-cancer drugs. Naturally, each successful cancer cell needs to develop the ability to avoid detection by the immune system and must resist programmed cell death (apoptosis). Furthermore, tumours start to induce the growth of new blood vessels (angiogenesis) to account for the increased energy consumption due to uncontrolled growth and replication. Hanahan and Weinberg [HW00] summarised these objectives under the term *hallmarks of cancer*. The hallmarks describe a set of characteristics tumour cells typically achieve during their development (Figure 2.2). In a later publication, Hanahan and Weinberg [HW11] added four additional characteristics to the six original hallmarks.

Tumours can be classified according to the cell types they stem from. The most common type of human tumours, carcinomas, derive from epithelial cells and are hence also dubbed epithelial tumours. Epithelia are tissues that can be found throughout the body. They are composed of sheets of cells and act as covers of organs, blood vessels, and cavities. The second class of tumours are called sarcomas and only account for roughly 1 % of all tumours. They stem from e.g. fibroblasts, adipocytes, osteoblasts, or myocytes. Tumours stemming from blood-forming cells

are more common. Examples are leukaemias and lymphomas. The last class are tumours that derive from cells of the nervous system. Examples are brain tumours such as gliomas and glioblastomas. While accounting for large part of all cancer types, some tumours such as melanomas or small-cell lung carcinomas derive from cell lineages that do not directly fit into this classification, as they stem from different embryonic tissues or have an unclear origin altogether.

In contrast to healthy tissue that is composed of cells with well-defined tasks, a tumour is usually significantly more heterogeneous. The key to this heterogeneity lies in the rapid rate with which tumour cells tend to accumulate mutations and epigenetic aberrations [LKV98; FT04]. As a specific mutation event only occurs locally in a single cell, it remains specific to this cell and its progeny. This effectively establishes lineages of cancer cell subpopulations. As a direct consequence, measurements from tumour samples often only provide values that have been averaged over an ensemble of genetically diverse cells. Also, not all cells involved in a tumour necessarily carry a defective (epi-)genome. For instance, cells forming blood vessels, cells from healthy tissue, and immune cells can be found there, too. Characterising a tumour or predicting its reaction to treatment reliably is thus difficult. Experimental techniques such as single cell sequencing only solve this problem to a certain degree. While they may provide complete knowledge about a set of individual cells, no knowledge is obtained about the billions of remaining cells that make up the tumour.

Similar problems exist in HIV genotyping, where only the sequence of one or few of the most abundant sub-species can be reliably resolved.

2.1.1 *Wilm's Tumour*

Wilm's tumours (WTs) are childhood renal tumours. They comprise 95 % of all diagnosed kidney tumours and six percent of all cancers in children under the age of 15. Most WTs are diagnosed in children younger than five years [Chu+10]. The clinical name *nephroblastoma* derives from the fact that WTs develop from the *metanephrogenic blastema*, an embryonic tissue. Most often WTs exhibit a triphasic histopathological pattern composed of blastemal, stromal, and epithelial cells although other compositions do exist [BP78]. According to the *Société Internationale D'Oncologie Pédiatrique (SIOP)* Wilm's tumours after preoperative chemotherapy are classified as follows [Vuj+02]: first, the presence of an anaplasia is determined. For this, the tumour must contain poorly differentiated cells, e.g. cells that lost defining morphological characteristics or display other potentially malignant transformations such as a nuclear pleomorphism. If the tumour is not anaplastic, the reduction in tumour volume due to the chemotherapy is quantified. If no living tumour tissue can be detected, the tumour is labelled as *completely necrotic*. If its volume decreased by more than two thirds the tumour is classified as *regressive*. Otherwise, the classification is based on the predominant cell type. If more than two thirds of the living cells are either *blastemal*, *epithelial* or *stromal* cells, the tumour is labelled accordingly. Tumours with an even distribution of cell types are classified as *triphasic*. Based

Still, WTs are comparatively rare due to the low prevalence of child cancers.

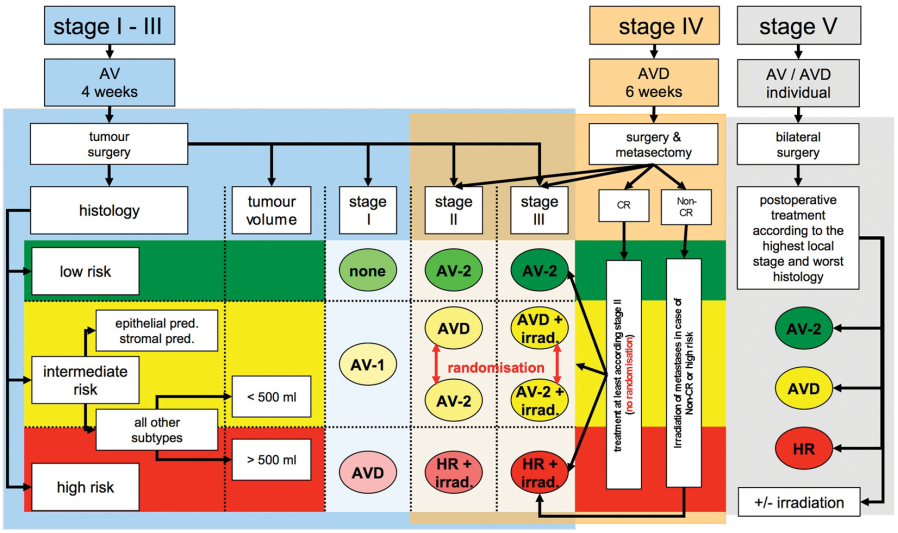


Figure 2.3: An exemplary WT treatment schema as used in the SIOP 2001 study. Tumours are stratified into risk groups based on tumour stage and histology. Abbreviations: dactinomycin (A), vincristine (V), doxorubicin (D), doxorubicin/carboplatin/cyclophosphamide/etoposide x 34 weeks (HR). Image taken from Dome, Perlman, and Graf [DPG14].

Risk group	**Classification**
Low risk	Completely necrotic
Intermediate risk	Regressive Epithelial type Stromal type Triphasic type Focal anaplasia
High risk	Blastemal type Diffuse anaplasia

Table 2.1: Wilm's tumour classification and risk groups after preoperative chemotherapy according to the SIOP reference [Vuj+02]. Blastemal subtype and diffuse anaplasia tumours are the most aggressive nephroblastoma subtypes.

on tumour stage, volume, and histology further treatment decisions are made (Figure 2.3).

While WTs are generally associated with high 5-year survival rates of $\approx 85\,\%$ [Chu+10; Sre+09], the prognosis for some subtypes is significantly worse. An example for this are the *blastemal subtype* tumours. While WTs with a high content of blastemal cells generally respond well to chemotherapy, this is not true in about 25 % of the cases. Such resistant, blastema-rich tumours, which account for almost 10 % of all WTs, are among the most malignant WT types (cf. Table 2.1) [Kin+12; Heu+15].

One of the earliest identified mutations associated with nephroblastomas is the inactivation of the tumour suppressor WT1 that likely

plays an important role in genitourinary development [LH01; Roy+08] and is also associated with human leukaemia [MBS92; Ino+97]. A more recent study by Wegert et al. [Weg+15] links mutations in genes such as SIX1, SIX2, DROSHA, TP53, and IGF2 to specific sub- and phenotypes of WTs.

While other, more aggressive cancer types such as lung cancer my be of greater medical interest, WTs possess properties that make them an ideal model disease. First, while a range of histological subtypes exist for this tumour, they are relatively homogeneous in the sense that they carry a comparatively low amount of mutations [Weg+15]. Thus, most biological processes remain intact, which greatly eases the interpretation of the data. Second, as the disease commonly occurs in young children, environmental effects play comparatively small roles in the development of the tumour. Third, as WT are quite rare [BP78], most cases are treated by a small community of experts. Accordingly, they are well and consistently documented.

2.1.2 *Cancer Therapy*

A core problem of cancer therapy is the timely detection of neoplasms. Many cancers can be treated effectively if detected at an early stage. For example, early stage melanomas can simply be excised with little to no adverse effects. Polyps in the colon, which may later develop to cancer, can be removed during routine colonoscopies [Win+93]. However, when a tumour is not detected during early stages, treatment becomes more difficult. Compared to early stage tumours, late stage tumours had considerably more time to acquire hallmarks of cancer traits (Figure 2.2). Consequently, they possess a less well-defined boundary as they begin to invade healthy, adjacent tissue preventing effective surgery (cf. [Suz+95]). Also, metastases, which spread via blood or lymph and form new, aggressive tumours in other parts of the body, may have been established [Fid03]. This makes it difficult to reliably assess the success of a treatment as it is uncertain whether all cancer cells could be successfully removed [Bec+98; Wet+02]. While chemotherapy may be used to combat these developments, drug resistance often limits its efficacy (cf. [Per99; RY08]). In the following we give a more detailed overview over the available options for diagnosis and treatment.

Diagnosis

Performing routine cancer screens can considerably lower the risk of developing cancer [Wen+13; Bre+12]. However, this creates a dilemma when planning an effective scheme for preventive care. First, no single test covering all possible tumour classes exists and it can be assumed that none will exist in the foreseeable future. This means that significant parts of the population would need to undergo several medical screenings in fixed or maybe age dependent intervals. However, schemes relying on unconditional screenings are problematic. Besides the considerable financial implications, there are also statistical issues

that make such an approach infeasible. The problem is that the likelihood to develop a certain type of cancer at a certain point is comparatively low (let's assume about 0.1 %). While in itself not problematic, this fact can have important implications together with a fundamental property of tests based on empiric data: all tests make errors. Generally, one can distinguish between two kinds of errors (cf. Table 4.1). The first class of errors is to detect an effect (e.g. diagnose cancer) when there is none, while the second class is to *not* detect an effect when there is one. Making an error of the second kind may result in an unfavourable diagnosis because the tumour is detected too late. Making an error of the first kind means that a healthy person must undergo unnecessary and potentially risky follow-up examinations such as biopsies. This is both, a waste of resources [OM15] and a burden on the health of the patient. Unfortunately, we can expect the first kind of error, diagnosing cancer although the patient is healthy, to occur far more often than the second kind because we previously assumed that the prevalence of cancer is only 0.1 % (c.f. [ZSM04; Etz+02]). The resulting amount of unnecessary follow-up examinations render broad, regular screenings both ethically as well as economically questionable. Instead, stratification into risk groups and targeted tests based on a patient's case history must be used to achieve effective, early diagnostics and is in fact recommended by studies and treatment guidelines [Wen+13; Bre+12; Bur+13].

Statistical tests are discussed in more detail in Section 4.1.

Techniques from bioinformatics may help in this targeted approach. As sequencing costs for a human genome have dropped dramatically in the last decade (Figure 2.10), genotype information as a supplement to classical factors used in risk determination, such as family history and lifestyle, has the potential to improve the accuracy of diagnosis tremendously. Additionally, the development of minimally invasive tests with high specificity can help to reduce the risk for patients significantly. To this end, the search for specific biomarkers is a prolific field of computational biology [Saw08]. In this context, a biomarker is a molecule or a set thereof that can be measured objectively and can be used as an indicator of the state of a pathogenic process. For this purpose also proteins and miRNA isolated from the blood stream have been used (cf. [Kel+06; Mit+08]).

Various definitions of a "biomarker" exist. Strimbu and Tavel [ST10] provide an overview over commonly used definitions.

Therapy

Cancer therapy is a large field that is impossible to discuss exhaustively. Here, we give a rough outline of the available treatment options.

For treating cancer three main angles of attack exist: surgery, radiation therapy, and chemotherapy [DLR14]. Each of these approaches come with distinct advantages and disadvantages. Surgery can be an extremely efficient treatment option, as possibly large tumour volumes can be removed. In ideal cases, all tumour mass can be removed in a single session [Cof+03]. However, it should be noted that the applicability of surgery is severely limited due to its invasiveness: tumours that are difficult to reach, too large in volume, or have developed metastases can prove difficult to treat with surgery alone [Pet+15]. Besides

curative purposes, surgery is also an important tool for diagnosis, as often only a biopsy allows to determine whether cancer is present and if yes, how treatment should be commenced (cf. [Hei+14]). Also, all approaches that rely on gene expression analysis or genotyping require that a biopsy is conducted.

In recent years technologies for isolating tumour cells from blood have been developed, which may alleviate the need for surgery (cf. [AP13]).

In combination to or instead of surgery, radiation therapy can be used for treatment [Tim+10]. Commonly, radiation therapy directs a beam of ionising radiation at a target tumour. Cells along the beam absorb energy leading to the formation of free radicals that attack and fragment DNA. In order to minimise the damage to healthy tissue, the beam is applied from multiple angles leading to an accumulation of an effective radiation dose only in the tumour [DLR14]. Still, radiation therapy can have severe adverse effects that can lead to the induction of secondary cancer (cf. [Tuc+91; Tsa+93; Kry+05; Hal06]).

Complementing surgery and radiation therapy, chemotherapy allows to treat tumours via the administration of drugs. As cancer cells remain by and large human cells, finding drugs that specifically target cancer cells is difficult [KWL07]. To this end, classical chemotherapy uses cytotoxic or cytostatic agents. These drugs target rapidly dividing cells, where they induce cell death or prevent proliferation. This often results in harsh adverse effects such as anæmia, fatigue, hair loss, nausea, or infertility. In contrast to this, targeted therapy attempts to attack specific molecules, most of the time proteins that are responsible for the deregulation of signalling pathways that takes place in cancer cells [Saw04]. Targeted drugs are not universally applicable, but require tumours to fulfil certain properties in order to be effective. Examples are the presentation of certain antigenes on the cell membrane or the target protein carrying a mutation [Saw08]. To achieve the necessary specificity, often biomolecules such as antibodies, which are able to detect cancer cell specific epitopes, are used. Thus, to apply targeted therapy effectively, an analysis of the tumour on the molecular level is necessary. For cases in which the preconditions for targeted drugs are met, impressive treatment success with comparatively few adverse effects has been reported. A popular example for targeted therapy are drugs targeting the EGF receptor (EGFR) and are only effective in (breast) cancers that carry a certain point mutation in the EGFR kinase domain [Pae+04; Mas+12]. Other types of targeted therapy attempt to attack tumour stem cells or the tumours microenvironment by e.g. preventing blood vessel formation [AS07; EH08].

To assist with this, we created the DrugTargetInspector tool [Sch+15] which we present in Section 5.5.

Usually, none of the previously described techniques is used in isolation. Often a combination of surgery, chemotherapy, and radiation therapy is used in different stages of the therapy. Figure 2.4 shows the treatment recommendations of the *European Society of Medical Oncology (ESMO)* for early breast cancer. In this recommendation, various factors such as tumour volume or the efficacy of previous treatments are considered to recommend the next step in the therapy [Sen+15].

Combining multiple therapy options is sensible for various reasons. First, a single treatment option is often not enough to guarantee the removal of all tumour cells. For example, residues of the tumour (cf. *minimal residual disease* [Kle+02]) that are difficult to excise during

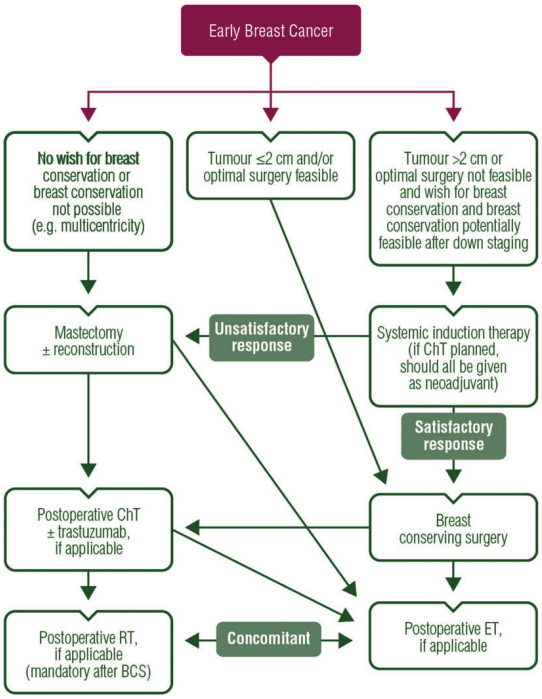


Figure 2.4: Flow chart for the treatment of early breast cancer. Abbreviations: chemotherapy (Cht), breast-conserving surgery (BCS), endocrine therapy (ET), radiotherapy (RT). Image taken from Senkus et al. [Sen+15].

surgery may be treated using adjuvant chemotherapy. Second, as tumours underlie a highly accelerated selection process due to their high proliferation and mutation rates, it is possible that some tumour cells develop a resistance against the employed cancer drugs. Similarly to HIV therapy, using multiple agents and treatment options increases the selection pressure and thus reduces the amount of escape mutants [SS08b; HZ12; BL12]. Third, using multiple, cancer specific drugs during chemotherapy has been reported to be more effective than using only a single formulation as this increases the specificity of the treatment [Coi+02; Ban+10].

2.2 PERSONALISED MEDICINE

In classical medicine, a doctor makes a diagnosis based on the patient's history (anamnesis), the displayed symptoms, and, if required, additional measurements. Based on this, the physician chooses an appropriate therapy. Often, this means administering one of the drugs designed for treating the disease. In recent years, it has become clear that for some diseases the genetic, epigenetic, and biomolecular properties of the patient as well as the disease need to be considered to determine an optimal therapy. The level of detail that needs to be considered can vary by a large margin. In some cases, the membership in a particular ethnic group can provide enough information on the genomic background. For example the drug *BiDil*, which was approved by the FDA

for treating congestive heart-failure, is only effective for the African American parts of the US population [BH06]. In HIV therapy the genome of the virus can provide crucial information on its resistance or susceptibility to certain antiretroviral drugs [Bee+02]. In heterogeneous diseases like cancer, taking the genes expressed by the tumour into account can help to dramatically increase treatment success [Kon+06]. These observations also become increasingly important for "classical" diseases such as bacterial infections due to the emergence of multi-resistant strains for which an effective antibiotic needs to be chosen [Neu92]. The development of treatment strategies that are tailored specifically towards a specific case of a disease is called *personalised medicine*. Deploying personalised medicine, however, proofs to be difficult due to a variety of issues. First, the identification of genomic factors that contribute to the risk of developing a disease is problematic. To this end, *genome wide association studies (GWAS)* are commonly applied. For them to succeed, large cohorts ($\geq$ 10,000 samples) are required [HD05]. Even then, rare variations, which are likely to have a large impact on the disease risk, continue to elude these studies [CG10].

Sometimes the term precision medicine is used.

Even if a molecule or process that plays a key role in a diseases has been identified, the selection of an appropriate drug is not always possible. This may be because no appropriate drug exists. However, even if a drug existed, it may be overlooked as only a small percentage of the available drugs is annotated with pharmacogenomic information [Fru+08]. To remedy this situation, studies such as the *Cancer Cell Line Encyclopaedia (CCLE)* [Bar+12] and the *Genomics of Drug Sensitivity in Cancer (GDSC)* [Gar+12] attempt to compile libraries of the effect of drugs on cell lines with a known set of mutations. However, these *in vitro* measurements are only the first step into the direction of more comprehensive pharacogenomic information: the overlap between the studies has been reported as "reasonable" by the authors [C+15] and as "highly discordant" [Hai+13] by independent researchers.

Pharmacogenomics concerns itself with the influence of genomic variations on the effect of drugs.

Drugs that are only effective given a certain mutation must, as all other drugs, undergo clinical trials to ensure safety and efficacy of the drug. Unfortunately, classical study designs, where a control group of patients receiving the standard treatment is monitored in comparison to a group receiving the modified treatment, are inefficient for validating the merits of patient specific drugs [Sch15b]. If e.g. the genomic trait responsible for the susceptibility to a drug is comparatively rare throughput the population, the number of patients that will respond can be expected to be low and hence the difference between uniformly selected control and sample groups is likely small. To reliably determine drugs that are only effective for parts of the population, improved study designs need to be employed [Sin05; Fre+10].

For bioinformatics, the development of effective personalised medicine schemes poses several challenges such as the reliable analysis of large-scale genomic data, the compilation of databases containing pharmacogenetic knowledge, as well as the training of reliable, statistical recommendation procedures [Fer+11]. Partly due to this, the road towards an "ideal", personalised medicine is still long. Nevertheless, significant steps are currently being made towards this goal [Les07]. In

this work, several tools that my have the potential to assist in choosing personalised treatments are presented. The primary example is DrugTargetInspector (Section 5.5), which employs pharmacogenomic information to judge the influence of somatic mutations on drug efficacy. Similarly, the enrichment and network analyses provided by our proposed tools GeneTrail2 (Section 5.3) and NetworkTrail (Section 5.4) can be used to gain insights into patient data.

2.3 BIOLOGICAL ASSAYS

An important task in bioinformatics is the processing of data obtained from biological experiments using methods from statistics and computer science. While it is convenient to test new algorithms on synthetic data, every bioinformatics method eventually needs to work with actual measurements. As a consequence, the developed methods need to take the properties of the data generation process and, thus, the used biological assays, into account. Furthermore, with the development of high-throughput assays such as microarrays, short read sequencing, or modern mass spectrometry the amount of generated biological data has grown to a staggering amount [Ste+15]. This places two further requirements on computational methods. First, the approach needs to be efficient enough to potentially process terabytes of data. Second, the employed statistical methods need to be able to cope with the high dimensionality of the data.

Here, we discuss the microarray and short-read sequencing technologies as the remainder of this thesis will mostly be concerned with data obtained from these two experimental setups. Notably, the focus will lie on expression datasets. To this end, the next section will provide a basic introduction of the fundamental mechanisms that underlie gene expression in eukaryotic cells. Readers familiar with this concept may want to directly skip to the introduction of the microarray platform in Section 2.3.2.

2.3.1 *Gene Expression*

This process is called transcription, as a text using letters from the DNA alphabet is copied into the same text using letters from the RNA alphabet.

The activity of a gene is usually defined as the rate with which it is read by the *RNA Polymerase II (PolII)* [Kor99] and subsequently translated into protein. This process is also known as *gene expression*. First, PolII *transcribes* RNA copies of the gene by reading the anti-sense DNA strand in a 3′ to 5′ direction (Figure 2.5). As these copies carry the information of the gene out of the nucleus to the translation machinery they are called *messenger RNA (mRNA)*. After transcription a 5′ cap structure as well as a 3′ polyadenyl tail is added to prevent too rapid degradation of the mRNA [Sha76; BKW93]. As eukaryotic genes do not only contain protein coding sequences, but rather are organised into protein coding *exons* and non-coding *introns*, further processing is required prior to export from the nucleus [KMR78]. Via *splicing*, all non-coding parts are removed from the precursor mRNA [WL11]. During

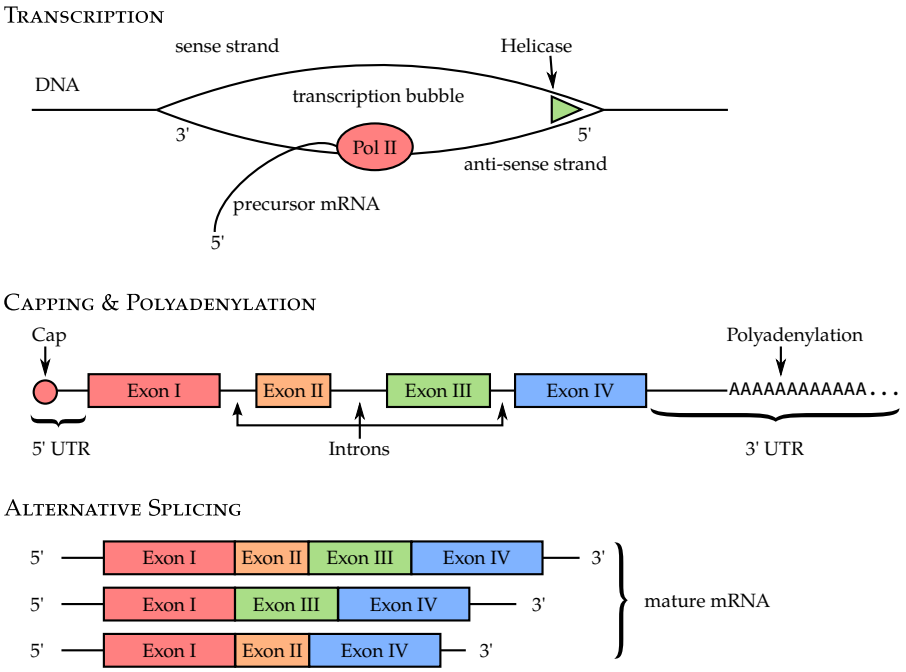


Figure 2.5: Basic overview of transcription in an eukaryotic cell. After RNA Polymerase II binds at the promoter of a gene, it translates the DNA anti-sense strand to mRNA. For transcription, the DNA is temporarily unwound by a helicase protein creating the so-called *transcription bubble*. After the mRNA has been transcribed, a 5′ cap and a 3′ polyadenyl tail are added to prevent degradation. Mature mRNA is created by the (alternative) splicing process, which excises intronic and sometimes exonic sequences.

this process some exons may be skipped and thus splicing can produce multiple *isoforms* of a single gene [Bla03]. This *alternative splicing* is one mechanism that explains how relatively few genes can give rise to a far larger amount of diverse proteins. After splicing, the mRNA is exported to the cytosol, where it is *translated* into proteins.

Besides mRNA, non-coding classes of RNAs exist that play important roles in the regulation and catalysis of cellular processes [Cec87; Edd01; MM06]. One such kind of RNA that has increasingly entered the focus of the scientific community during the last 20 years are so-called *micro RNAs (miRNAs)*. Each miRNA consist of 21 to 25 nucleotides and possesses the ability to bind the 3′ *untranslated region (UTR)* of a mRNA [Fil+05]. Such binding events can result in the mRNA's degradation or may inhibit its translation into proteins. Thus, miRNAs form an additional regulatory mechanism that controls the rate with which a gene is translated into proteins (cf. Figure 1.2). As with traditional genes, miRNAs are transcribed by PolII yielding a 3′ polyadenylated and 5′ capped *primary miRNA (pri-miRNA)* [Lee+04]. This transcript is then processed into *precursor miRNA (pre-miRNA)* by the Drosha protein [Den+04] and exported to the cytosol. There, it is cleaved by Dicer and loaded into the *Argonaute (AGO)* protein of the *miRNA-ribonucleoprotein (miRNP)* complex [Fil+05]. The loaded complex then detects and binds complementary mRNA in its 3′ UTR.

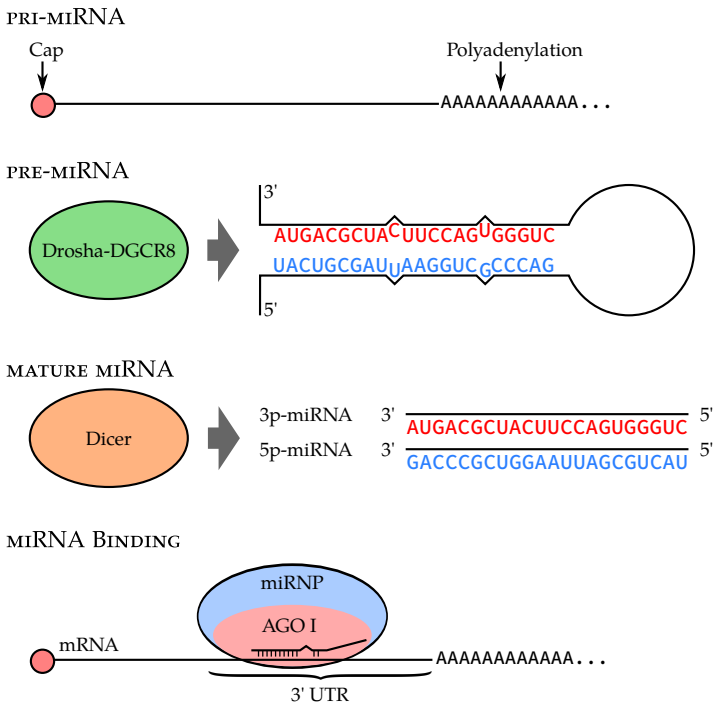


Figure 2.6: Maturation and binding of miRNAs. Primary miRNA is transcribed by PolII and carries the same 3′ polyadenyl tail and 5′ cap structure as mRNA. A stem loop is excised by the Drosha protein resulting in precursor miRNA. This pre-miRNA is further processed by Dicer yielding 5p and 3p mature miRNAs. The miRNA is then loaded into the AGO domain of the miRNP complex, which is then guided to a mRNA with a complementary binding motive in the 3′ UTR.

Due to post-transcriptional regulation this assumption often breaks down.

A general assumption is that the expression or effect of a gene is proportional to the number of mRNA or miRNA transcripts that are available at a given point. Assays that allow to measure the concentration of a given transcript should thus allow to deduce the regulatory state of a sample and, in turn, the currently active biological processes. With mRNA/miRNA microarrays and RNA-seq, methods for measuring complete transcription profiles have been developed.

2.3.2 *Microarrays*

Further common use cases are the measurement of DNA methylation states or protein abundances.

Microarrays or *gene chips* are experimental platforms that are most commonly used for measuring gene expression levels. They are based on the hybridisation of complementary DNA strands. In principle a microarray is a glass, plastic, or silicon slide onto which patches, so-called *spots*, of oligonucleotide *probes* have been fixated. In general, the probe sequences are chosen such that they are highly specific for the targeted transcripts. Modern mRNA microarrays contain millions of probes that allow them to cover all known exons of the human genome [Kro04].

To analyse a sample (cf. Figure 2.7), the mRNA is extracted and transcribed back to DNA (reverse transcription). In this process, nucleotides carrying fluorescent labels are incorporated into the resulting,

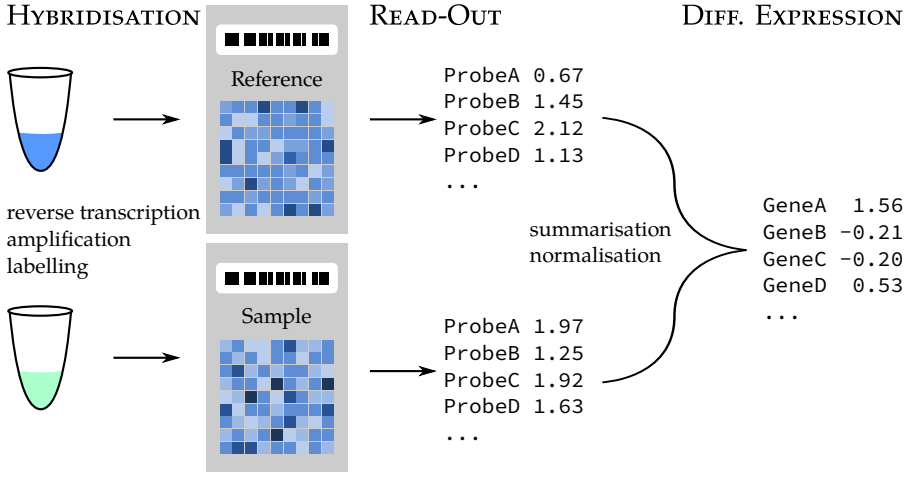


Figure 2.7: Overview of a microarray-based differential gene expression study. The mRNA is extracted from the sample and reference specimens. It is reverse-transcribed to cDNA, amplified, and labelled with fluorescent nucleotides. The cDNA is then hybridised to a microarray. After fixation and washing steps, the array can be read out using a laser scanner. The obtained raw expression values are normalised and summarised to obtain per-gene expression values. Afterwards, differentially expressed genes can be detected.

complementary DNA (cDNA). This labelled cDNA is then given onto the chip where it hybridises to the respective complementary probes (see Figure 2.7). After a washing step that removes excess cDNA and unspecific binding events, the chips can be read out. Using a laser, the amount of fluorescence for every spot and hence the amount of bound cDNA can be determined.

Due to the advent of RNA-seq, only few microarray manufacturers remain. The major ones are Affymetrix, Agilent and Illumina.

Though the principles underlying the microarray technology are straightforward, various practical challenges need to be solved to obtain reliable expression values. A central issue is the fixation of probes onto the microarray slide. For this purpose, a set of fundamentally different approaches exists. Chips by Affymetrix and Agilent use an *in situ* process for creating the probes directly on the chip substrate. Affymetrix uses a photolithographic process for synthesising the probes directly on the substrate (Figure 2.8). Agilent uses and ink-jet-like process similar to a regular printer in which nucleotides are successively sprayed onto the chip and incorporated into the probe sequence. The length of the synthesised probes is, however, limited, as with growing probe length errors start to accumulate depending on the used technology. Thus, the probe length varies from manufacturer to manufacturer. For chips from Affymetrix (e.g. GeneChip HuGene 2.0 ST arrays), the probe length is 25 bases [Aff07] whereas chips from Agilent (e.g. SurePrint G3 arrays) use 60 bases [LeP08]. To account for the shorter probe length, Affymetrix chips use multiple probes for the same transcript, whereas Agilent relies on a single probe [Mul+16]. In addition, Affymetrix chips carry *perfect match (PM)* and *mismatch (MM)* spots for the same transcript that, in theory, should allow to better quantify the amount of unspecific cross-hybridisation. In practice, the value provided by MM probes is questionable [Iri+03]. Chips man-

The classic pin-spotting technology is plagued by problems such as mechanical wear on the pins. Nowadays, it is hardly being employed.

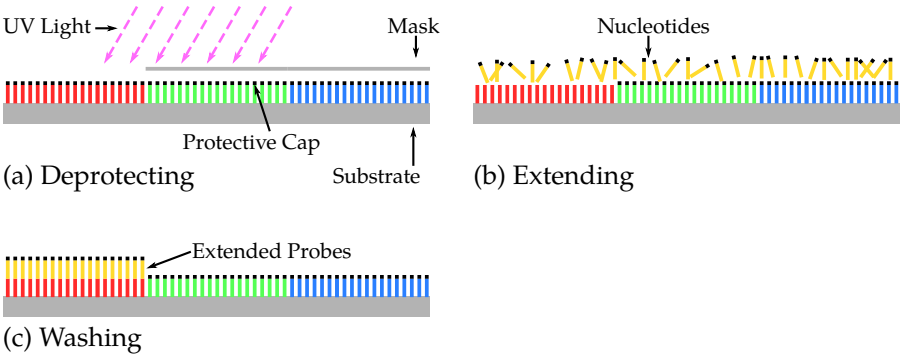


Figure 2.8: Manufacturing process of Affymetrix microarrays. (a) The 5′ ends of the probes are protected with a cap. For extending a probe, a photolithographic mask is placed on the chip. Ultraviolet light removes the caps and (b) allows new nucleotides to bind. After a washing step (c), the process is repeated with a different mask [INC05].

ufactured by Illumina use the so-called BeadArray technology. Here, beads carrying probes of length 22 to 24 are immobilised in small wells etched into the chip surface. In contrast to other microarray technologies where the location of a probe on the chip is well known, the beads and hence the probes are randomly distributed across a BeadArray. Accordingly, the beads occupying the wells need to be identified prior to measuring. To this end, techniques based on coding theory developed for "Sequencing by Hybridisation" [SME92] are applied. As no DNA needs to be synthesised on the chip, higher packing densities are possible than for the previous two technologies [Gun+04].

Microarray Preprocessing

Microarray data contains measurement errors and uncertainty. Usually two types of noise are being distinguished: technical noise and biological noise. Whereas biological noise quantifies the natural fluctuations that can occur between two different samples *under the same conditions*, technical noise accounts for the variability introduced by sample preparation and the experiment itself. In addition, batch effects and confounding factors need to be taken into account during data analysis. Batch effects arise whenever a set of samples needs to be measured in multiple chunks e.g. due to a large sample size. Examples are changes in external conditions such as temperature or lighting. The same is true for samples analysed using different master mix solutions or conducted by different experimenters from different labs. These factors introduce systematic errors that can lead to shifts in expression intensity and variance between groups of chips. Confounding factors lead to similar problems, but depend on the analysed samples themselves. For example, the age or gender of the patients in the control and samples groups can lead to biases in the measured expression values.

For obtaining expression values from microarray experiments, multiple steps are necessary.

Figure 2.9: Affymetrix microarrays. Source: `https://commons.wikimedia.org/w/index.php?title=File:Affymetrix-microarray.jpg&oldid=165509627`

Microarrays can analyse paired samples simultaneously by using different dyes (dual-channel experiments). Here, we only discuss experiments using one dye (single-channel experiments).

Scanning and Image Analysis Capturing the raw data of a chip is achieved using a laser scanner. This will create a grey-scale image that needs to be further analysed. Depending on the chip technology multiple image processing steps are needed to detect and address the fluorescence signal from the probes. These may account for slight rotations and spot irregularities. Once the pixels belonging to a spot have been identified, a raw expression value for this spot is computed by integrating over the measured intensities.

Background Correction While the washing step in microarray protocols eliminates most unspecific binding events, a residual amount of unspecific hybridisations remains. To remove this effect from raw data a background correction step is performed.

Normalisation Once raw expression values have been created, it is necessary to conduct a normalisation step to ensure the compatibility between probes (intra-array) as well as samples (inter-array). Intra-array normalisation also entails the aforementioned background correction step. For inter-array normalisation, the expression value distributions of the chips are made comparable. Various normalisation algorithms and software packages are available. Examples are the *variance stabilising normalisation (VSN)* [Hub+02] or the *quantile normalisation* technique [Bol+03].

Summarisation To make the measurements more robust, multiple probes, or copies of a single probe that all match to the same mRNA are

distributed across the chip. These raw values need then to be summarised to a single expression value. To protect against outliers, algorithms such as the median polish procedure [Hol+01] are used during summarisation.

Batch-effect removal As mentioned above, microarray experiments are highly sensitive to changes in the external conditions. In practice, this means that experiments that were broken up into batches due to a high sample count will incur systematic errors that differ between the batches. This can lead to artefacts that obscure actual, biological signal when analysing the data using standard workflows [Lee+10]. To account for this effect, batch-effect removal techniques such as SVA [LS07], COMBAT [Che+11], or RUV2 [GS12] can be applied. Alternatively, it is possible to include batch information in addition to other, known, confounding factors into the design matrix of the experiment when computing scores for differential expression. This approach is chosen by the limma R package [Rit+15]. In all cases, the design of the experiment must account for batch-effect removal. This means that one or more samples of every phenotype should be measured in each batch. Otherwise, assuming a linear batch effect model, the phenotype and the confounding factor become linearly dependent making it impossible to reconstruct the actual expression values.

Differential Expression Once the raw data has been pre-processed, it is possible to perform the actual analysis. Microarray data is usually assumed to be normally distributed, although other distributions such as the Laplace distribution can arise (cf. Section 4.3.1). Nevertheless, statistics such as the t-test are commonly applied and provide good results when computing differential expression [C+03]. The previously mentioned limma package uses ANOVA for this purpose. As the p-value computed by statistical tests makes no statement about the biological relevance of the detected effect, it is advisable to examine the *effect size* [RCH94]. An increasingly more popular visualisation technique, which helps to quickly identify significant, biologically interesting genes, is the so-called *volcano plot*. In a volcano plot the negative logarithm of the computed p-value is plotted against the log-fold change (e.g. Figure 5.12). We discuss methods for detecting differentially expressed genes in more detail in Section 4.2.2

2.3.3 *High-Throughput Sequencing*

Although the technology has been available for a decade it is still often referred to as next-generation sequencing (NGS).

In this section, the basic principles behind high-throughput sequencing are introduced. As in this thesis sequencing and the RNA-seq technology will only play a minor role (cf. Section 5.3), we refer the interested reader to the literature for a more thorough treatment.

The ability to sequence an organism's genome is an essential building block for understanding the differences between species or between individuals within a species. Using the classical Sanger sequencing

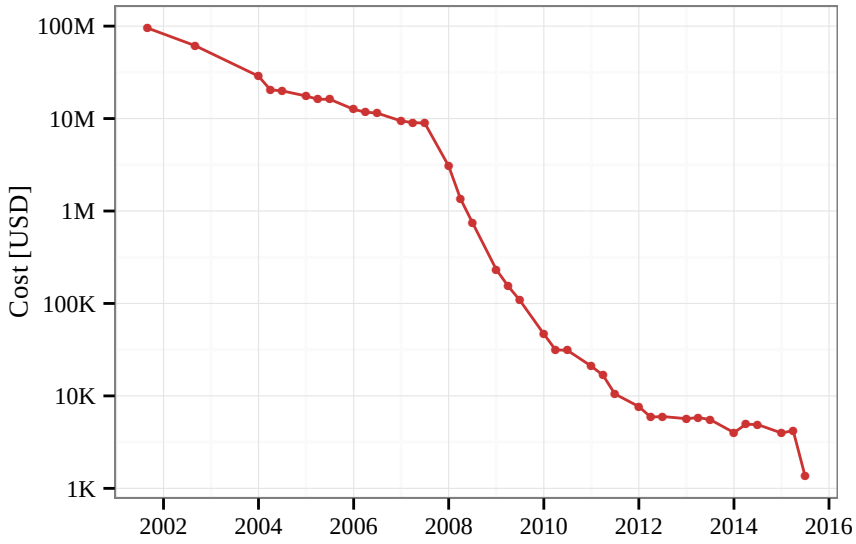


Figure 2.10: Development of the costs for sequencing a complete genome as reported by the NHGRI. Due to the development of high-throughput sequencing techniques, sequencing costs have dropped exponentially since the publication of the first human genome [Wet15].

procedure, relatively long *reads* can be determined. For example, the *Sanger ABI 3730xl* platform [GAT16], is able to determine the sequence of DNA fragments with a length of up to 1100 *base pairs (bp)* in a single experiment. This particular platform is also able to sequence 96 reads in parallel yielding a throughput of roughly 1 Mbp per day. Based on a back-of-the-envelope calculation, sequencing the human genome consisting of roughly 3 Gbp with a tenfold coverage requires sequencing at least 30 Gbp or over 30 million reads. Accordingly, 100 machines would need 300 days for processing all reads. A substantial amount of money and time was, thus, required to sequence even a single genome and made it infeasible to sequence a complete genome in routine research and medical settings, such as analysing cancer biopsies. This changed roughly a decade ago, with the advent of high-throughput sequencing. Instead of sequencing a single DNA strand, high-throughput sequencing allows to analyse millions of reads in parallel. This is commonly achieved by creating micro reaction environments in each of which an independent sequencing run takes place.

How these environments are created is dependent on the used technology. For example the Roche 454 sequencers, one of the earliest high-throughput sequencers, use small beads carrying a short adapter DNA on their surface (Figure 2.11). These beads are contained in small water droplets in a water-in-oil emulsion. The DNA reads are ligated with complementary adapter DNA and subsequently captured by the beads. Due to the used dilution, only one read is expected to hybridise with one bead. The water droplet then serves as a reaction environment for a PCR. The beads with the amplified DNA are then placed onto a micro reaction plate (PicoTiterPlate) where the actual sequencing takes

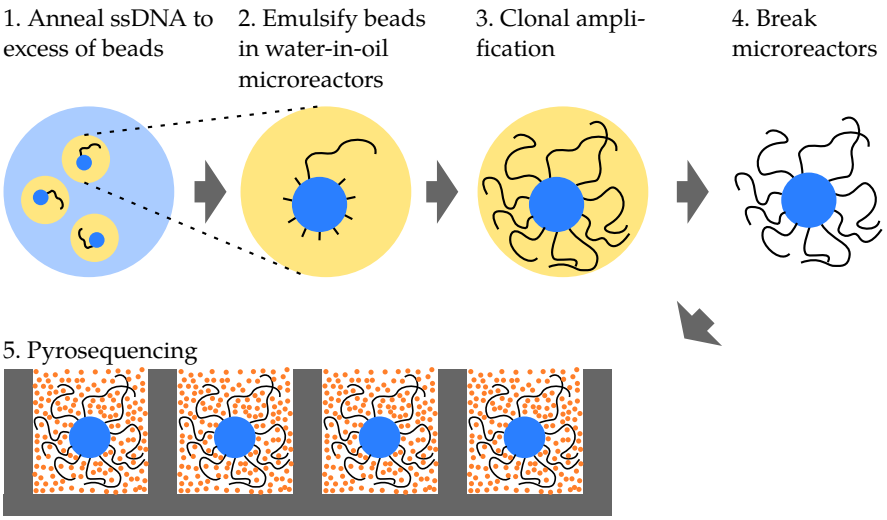


Figure 2.11: Principle of Roche 454 sequencing. Sample DNA is captured in a reaction microenvironment where it is amplified and attached to a bead. The beads are placed in a micro reaction plate such that exactly one bead occupies one well. The sequence of the sample DNA is then recovered using pyrosequencing. Adapted from: `http://www.lifesequencing.com/pages/protocolo-de-secuenciacion?locale=en`

place. Similarly, the Illumina HiSeq platform uses reaction plates on which clusters of cDNA strands are synthesised in etched nano-wells. While the Roche 454 platform employs pyrosequencing [Ahm+00], Illumina technology relies on fluorescent labels. The *Roche 454 GS FLX Titanium XL+* system produces reads with an average length of 700 bp [45416], whereas the *Illumina HiSeq X* platform produces shorter reads of length 150 bp [Ill16]. Newer high-throughput sequencers such as the *Oxford Nanopore MinION* or *PacBio RS II* are able to achieve longer read length of roughly 1 kbp to 2 kbp and 10 kbp to 15 kbp length, respectively [Lav+15; RCS13].

A disadvantage of the MinION and PacBio technology are the relatively high error rates.

Usually the results are stored in the FASTQ format.

The result of a sequencing run is a list of reads with associated quality scores. After optional error-correction steps (e.g. [Le+13; Heo+14; Med+11]), it is possible to perform a *de novo* assembly or align the captured reads to a reference genome. The reference genome based approach is computationally much less demanding and poses no problem to modern hardware [LD09; Dob+13]. However, especially the reliable detection of indels (insertions or deletions) and other genomic aberrations remains a challenge [CWE15]. Fortunately, due to the development of sequencers that produce far longer reads, it becomes more feasible to conduct *de-novo* assembly for increasingly larger genomes [Li+10; Kor+12; CWE15].

Applications of High-Throughput Sequencing

The availability of fast, inexpensive sequencing technologies enabled various, innovative experimental designs beyond "merely" sequencing a genome. Using the bisulfite technique [Fro+92], which converts unmethylated cytosine into uracil it is possible to determine the methyla-

tion state of the genome down to base resolution. Chromatin immunoprecipitation in combination with sequencing (ChIP-seq) allows to determine the occupancy of DNA with binding proteins [Bar+07b; Par09]. For this, the proteins are cross-linked to the DNA. Then, using nucleases or sonication, the DNA is broken into fragments. After that, all proteins of interest are selected using specific antibodies. Due to the cross-linking, these proteins carry a DNA fragment with them. In the next step, the bound DNA is removed from the purified protein and sequenced subsequently. Aligning the obtained reads to a reference genome reveals occupied protein binding sites. This technique is routinely applied to analyse the binding sites of transcription factors [Val+08] and histone variants [Bar+07b]. Finally, just as mRNA/miRNA microarrays, RNA-seq allows to capture the complete transcriptome of a tissue sample. We will now discuss this particular application in more detail.

RNA-seq

Although providing a more direct measure of transcriptional activity than microarrays, RNA-seq does not (yet) report abolute transcript abundances.

For measuring expression using high-throughput sequencing, RNA is extracted from a sample and, depending on the used protocol, filtered for e.g. short RNA or mRNA [Mar+08a]. Subsequently, the purified RNA is transcribed back into cDNA, which is then sequenced [WGS09]. By determining the amount of sequencing reads that originated from a given transcript, the abundance of this transcript can be estimated. Traditionally, alignments against a reference genome or transcriptome are used to determine the transcripts a read originated from. To this end, aligners should be used that are able to map reads across splice junctions such as the STAR aligner [Dob+13]. Recently, alignment free methods such as Sailfish [PMK14] and Kallisto [Bra+15] have been published. These approached rely on k-mer based algorithms for creating "pseudo-alignments". Salmon [PDK15] computes "lightweight" alignments by detecting approximate positions of a read in the reference. In contrast to the classical, alignment based workflows these algorithms offer a much improved runtime behaviour with little to no degradation in accuracy.

After determining a read-to-transcript assignment, the number of reads that map to a transcript is computed. To directly compare the read counts between two transcripts they need to be normalised with respect to the transcript length. For this purpose the *fragments per kilobase of exon per million fragments mapped (FPKM)* [Tra+10] and *transcripts per million (TPM)* [WKL12] statistics are used. Differentially expressed genes or transcripts are computed with specialised software such as DESeq2 [LHA14] or EdgeR [RMS10]. These packages take advantage of the discrete probability distribution of read counts to detect differentially expressed genes more reliably. Nevertheless, software originally developed for microarray analysis such as limma is used and produces good results [Rit+15].

2.3.4 *Comparison of Microarrays and RNA-seq*

Microarray have established themselves as a well understood, battle-proven experimental technology for capturing expression profiles. In recent years, the RNA-seq technology is gaining popularity, though. A large advantage of RNA-seq over microarrays is that the raw data can be easily reanalysed for newer genome builds and annotations as it is not tied to a particular probe design. As raw reads are available it is also possible to call genomic variations in coding regions [Che+09]. In addition, due to the increasing read length for modern high-throughput sequencers, reliably resolving expressed isoforms becomes more and more feasible [Chh+15]. A further point in favour of RNA-seq is that the obtained read counts are a more direct representation of the actual RNA abundance than the indirect fluorescence signals obtained from microarrays. Yet, microarray platforms continue to be used due to well established protocols and analysis pipelines. Furthermore, the expression profiles generated using current microarray platforms remain competitive with RNA-seq based methods in terms of quality. Ultimately, however, RNA-seq is more versatile and can be expected to replace microarrays for most use cases [Zha+14; Man+14; Zha+15].

Due to the required read amplification and normalisation based on fragment length an absolute quantification is difficult even with RNA-seq.

2.4 WILM'S TUMOUR DATA

CONTRIBUTIONS Most WT tissue samples were collected by the group of Prof. Manfred Gessler. Some samples as well as clinical data were provided by the group of Prof. Norbert Graf. The microarray assays were performed by Nicole Ludwig from the group of Prof. Eckart Meese. Normalisation of the raw expression values was performed by Patrick Trampert.

Parts of this section have been adapted from Stöckel et al. [Stö+16].

In Section 2.1.1 we introduced Wilm's tumour and some of its common subtypes. We noted that tumours containing a high amount of blastemal cells after pre-operative chemotherapy, the so called *blastemal subtype*, is one of the most aggressive Wilm's tumour subtypes. To investigate why some tumours with a high blastem content respond well to chemotherapy while others do not, we generated an expression dataset of Wilm's tumours that were treated with pre-operative chemotherapy according to the SIOP protocol (cf. [Isr+13; DPG14]).

Patient Samples The dataset consists of 40 mRNA and 47 miRNA expression profiles from 47 tumour biopsies collected from 39 patients. These biopsies contain four healthy tissue samples as well as 16 blastemal, nine mixed type and 17 miscellaneous tumour samples that were labelled as described in Section 2.1.1. Before surgery, patients were treated using the SIOP standard regimen consisting of Actinomycin-D, Vincristine, and, in the case of metastases, Doxorubicin (cf. Figure 2.3).

Clinical details of the patients included in the analysis and an overview over the generated expression profiles is given in Appendix B. The research was approved by the local ethical committee (Ethikkommission der Ärztekammer des Saarlandes, No. 136/01; 09/16/2010).

RNA Isolation Total RNA including miRNAs was isolated from tumour and control tissue using the miRNeasy Kit (Qiagen, Hilden, Germany) according to manufacturer's instructions. The RNA concentration and integrity were assessed using NanoDrop2000 (Thermo Fisher Scientific, Waltham, MA, USA) and Bioanalyzer runs using the Pico-RNA Chip (Agilent Technologies, Santa Clara, CA, USA).

miRNA Expression Profiles miRNA expression profiles were measured using the *SurePrint G3 8x60k* miRNA microarray (miRBase v16, Cat. no. G4870A) according to the manufacturer's recommendations. Background-corrected, log-transformed expression values were extracted using the Agilent *Feature Extraction Software* and normalised using quantile normalisation. In total, 47 miRNA expression profiles were generated.

mRNA Expression Profiles All mRNA expression profile were measured using *SurePrint G3 Human Gene Expression 8x60K v2* microarrays (Cat. no. G4851B, Agilent Technologies, Santa Clara, CA, USA) according to the manufacturer's recommendations. Expression values were extracted, background-corrected, and log-transformed using the Agilent *Feature Extraction Software* and normalised using the *GeneSpring* software [CSK01]. In total 40 mRNA expression profiles were generated.

3 BIOLOGICAL NETWORKS

The whole is more than the sum of its parts.

— ARISTOTLE, metaphysica

Cells are highly complex, molecular machines, each of which is composed of billions of biological entities such as genes, proteins, RNAs, and metabolites. These entities fulfil their tasks through interaction. Examples are proteins binding metabolites to catalyse a metabolic reaction or two proteins forming a complex. When a complex is formed, it again can participate in further interactions. Similarly, miRNAs can bind to mRNAs in order to make the translation process less efficient or completely inhibit it (Section 2.3.1). On the genome level, enhancers and genes can associate with transcription factors. This can lead to the initiation or repression of expression or even to the remodelling of the chromatin structure. Which molecules are able to interact is determined by the chemical and structural properties of the putative interaction partners. In practice, the physical location of the molecules inside the cell also plays an important role. For example, a molecule inside the nucleus will not react with another molecule in the cytoplasm as they are separated by the nuclear membrane. Moreover, not every gene is expressed in every cell and hence not every protein is available everywhere. Consequently, reactions that are possible in one cell type need not be possible in another cell type.

Interactions are typically stochastic. That means that even a highly specific binding interface still allows for unspecific binding, albeit with low probability [VB86].

Viewed together, the interactions in a cell form a *network* in which the entities are connected with edges representing interactions. Considering that there exist approximately 20,000 human genes encoding roughly 80,000 proteins, the amount of possible interactions already ranges in the billions [Har+06; Har+12]. This completely disregards the vast amount of RNA and metabolites that are present at any point in time. Even when excluding interactions that are impossible due to chemical considerations or tissue specificity, it is clear that charting these networks is a daunting task. In a valiant effort towards reaching this goal, interaction databases have been created with which researchers attempt to catalogue all known biological entities and their interactions in a structured and well-defined fashion. As opposed to the unstructured representation of knowledge found in the literature, the availability of databases enables researchers to efficiently search the stored data. Furthermore, it enables the use of bioinformatics analysis pipelines, which is imperative for the ability to deal with genome-scale datasets. This development lead to the emergence of *systems biology*. In contrast to the classical, bottom-up approach of biology [Laz02], in which parts of a biological system are studied in isolation, systems biology attempts to detect general patterns on a global level [Kir05; Kit02]. This means that instead of studying biological processes in minute detail, systems biology is dedicated to discovering general architectures and building blocks that nature repeatedly uses to create complex bio-

The number of (protein coding) genes and their transcripts continues to vary. The reported numbers were taken from GENCODE 24.

logical systems. To achieve this, systems biology relies on computational methods and machine learning techniques for extracting the information stored in biological databases.

In this chapter, we will take a look at various methods that assist researchers in reasoning about biological systems. In each section, we give a short discussion of the respective state of the art. In addition, we present algorithms that exploit the knowledge stored in interaction databases to improve the analysis of experimental data. For this, we first introduce some general terminology. Next, an overview over the available types of databases is provided. After this, we take a look at how *Bayesian networks (BNs)*, and especially *causal Bayesian networks*, can be employed to test hypotheses about cellular processes. Based on this, we introduce our tool *CausalTrail* [Sch15a; Stö+15] which, to the best of our knowledge, is the first tool that allows researchers to evaluate causal queries on a causal Bayesian network structure. Afterwards, we give a short introduction to *Gaussian graphical models (GGMs)*. While we will not use GGMs to infer networks, Section 4.4 demonstrates how to use the underlying theory to construct an "enrichment algorithm" that corrects for the correlation structure in biological categories.

While the previously mentioned approaches model network structures from a probabilistic point of view, also purely graph theoretical algorithms exist. As an example, we present an ILP approach for finding deregulated subgraphs in regulatory networks [Bac+12]. In contrast to many other approaches, our algorithm is exact and highly efficient. While algorithms for detecting deregulated subgraphs by design are influenced by the provided topology, it is unclear how large its impact on the computed results is. To this end we perform an evaluation demonstrating how the network topology and input scores affect the produced solutions.

3.1 GRAPH THEORY

Before we begin with the explanation of the methods and concepts presented in this chapter, we first introduce some basic terminology from graph theory. In mathematics, graphs describe binary relations (*edges*) between a set of objects called *nodes* or *vertices*. Thus, graphs lend themselves for modelling biological networks. Formally a graph is defined as follows:

Definition 1 (Graph)**.** Let V be an arbitrary, finite set of *nodes* or *vertices* and $E \subseteq V \times V$ the set of *edges*. We call the pair of node and edge set a *graph* and write $G(V, E)$. G is *undirected*, if for every edge $(u, v) \in E$ also $(v, u) \in E$ holds.

A graph G is called *simple*, if no edge $(v, v) \in E$ exists. Unless mentioned otherwise, we assume that all graphs mentioned in this thesis are simple.

An often encountered attribute of nodes are their *degrees*:

Definition 2 (In- and out-degree)**.** Given a graph $G = (V, E)$, the *in-degree* d_v^- of a node v is the number of edges that end in v. Conversely, the *out-degree* d_v^+ of v is the number of edges starting in v.

Another important property of a node is whether it is possible to reach it from another node. To give a proper definition of this notion, we first introduce the concept of a *path*.

Definition 3 (Path)**.** Let $G(V, E)$ be a graph. If the sequence

$$p = \langle v_1, v_2, \ldots, v_k \rangle$$

of vertices with $k \in \mathbb{N},\ k \geq 2$ fulfils $(v_i, v_{i+1}) \in E$ for all $i < k$ it is called a *path* of length k.

If for two vertices $u, v \in V$ there exists a path starting in u and ending in v, we say that v is *reachable* from u. The set of vertices that is reachable from u is called the *descendants* of u, whereas all vertices that can reach v are called its *ancestors*. Descendants and ancestors that are only one edge away are called the *children* and *parents* of a node, respectively.

A graph is called *strongly connected* if every node v can be reached from every other node $v \neq u$. With $\tilde{E} = E \cup \{(v, u) | (u, v) \in E\}$ we denote the *symmetric closure* of the edge set. G is called *connected*, if $H(V, \tilde{E})$ is strongly connected. A *subgraph* $H \subset G$ is a graph $H(V', E')$ with $V' \subset V$ and $E' \subset E$. A *connected component* of G is a maximal, connected subgraph.

In some graphs, it is possible to find paths that start from and end in the same vertex v.

Definition 4 (Cycle and circle)**.** A path $C = \langle v_1, v_2, \ldots, v_k, v_1 \rangle$ is called a *cycle*. If no vertex, except v_1, appears twice in C, we call C a *circle*. If a graph does not contain any cycle, it is called *acyclic*.

In some cases, we refer to edge or node labels. Commonly, these can be thought of as functions mapping the nodes and edges to a value.

Definition 5 (Node and edge labels)**.** Let $G(V, E)$ be a graph. A *node label* for G is a function $w : V \to X$, where X is an arbitrary set. Similarly an *edge label* is a function $l : E \to X$.

We often refer to vertices using an index $i \in \mathbb{N}$. For convenience, we use the notation $w_i := w(v_i)$ for referring to the node label of $v_i \in V$. Examples for node labels are scores for differential expression that are attached to each node. We call such labels *node weights* or simply *scores*. In a regulatory network it is common to label each edge with the type of the interaction it describes.

3.2 TYPES OF BIOLOGICAL NETWORKS

Also, specialising on one network type makes modelling simpler and allows curators to focus on their area of expertise.

The sheer number of biological interactions makes it difficult to completely catalogue them. To simplify this task, repositories that specialise in storing one or a selected few interaction types have been created. We call the different kinds of networks that are stored in these databases *network types*. Various factors have lead to the creation of a substantial amount of interaction databases for each network type. For example, the sources from which interaction data is derived can differ substantially. While some databases rely on manually curated data obtained from the literature, others use text-mining approaches or prediction tools. Again, others only consider interactions confirmed by reliable, highly specific experiments. Finally, some databases also consider evidence from high-throughput experiments as sufficient.

Here, we shortly discuss the most common network types: protein-protein interaction, metabolic, and regulatory networks. For each network type, a list of representative databases is provided. In addition, we introduce methods to infer networks directly from high-throughput data. Of these methods we only discuss co-expression networks in this section. Bayesian networks and Gaussian graphical models are introduced separately in Section 3.3 and Section 3.4, respectively.

3.2.1 *Network Types*

Commonly, research has focused on the following three network types: *protein-protein interaction*, *metabolic*, and *regulatory networks*. We briefly discuss each of these network types and list databases from which they can be obtained. However, it should be noted that no single, clear definition of these network types exists and thus some databases contain interactions that stem from multiple types.

Protein-Protein Interaction (PPI) Networks

A protein-protein interaction (PPI) network is a, usually undirected, graph in which nodes represent proteins and edges indicate that two proteins interact (Figure 3.1). PPIs can be determined in a high throughput fashion using experimental setups such as *yeast-two-hybrid* [Uet+00; FS89] or *affinity purification* screens [Rig+99; Tin11; BVR04]. However, several lower throughput, but higher confidence methods for detecting PPIs including co-immunoprecipitation [Sal+02; Bar+07a] or Far-Western Blotting [WLC07], exist. Instances of PPI network databases are DIP [Sal+04], HPRD [Pra+09], and MINT [Lic+12] that store manually curated PPI data. DIP provides interactions for a wide range of organisms, but is limited to a small set of proteins and high-confidence interactions. MINT also provides entries for a large number of organisms, but is less conservative than DIP. HRPD focuses on human proteins alone. However, with more than 41,000 PPIs it far more comprehensive than DIP and MINT. STRING [Szk+14] is a metadatabase that incorporates knowledge from primary databases such as DIP and MINT as

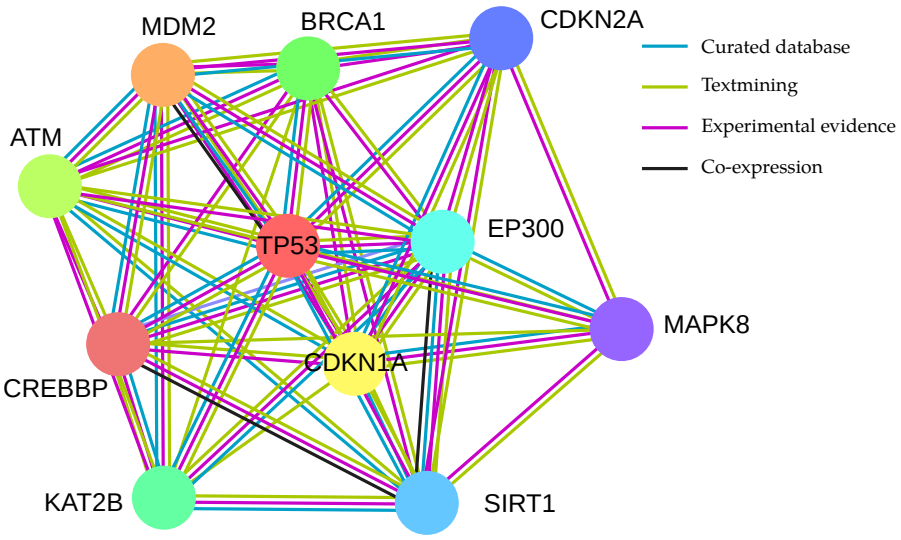


Figure 3.1: Part of the STRING v10 [Szk+14] protein-protein interaction network for human. Coloured edges indicate the types of evidence supporting the interactions of the incident nodes. This network depicts direct interaction partners of TP53, a transcription factor that plays an important role in the apoptosis. Proteins such as ATM and MDM2 are known activators and repressors of TP53 respectively.

well as the results of text-mining and prediction tools (cf. Section 3.2.2). Consequently, STRING contains a large number of PPIs that may, however, be of low confidence.

Metabolic Networks

Metabolic networks describe the chemical reactions that take place in an organism. As most biological reactions involving small molecules are catalysed by enzymes, metabolic networks are commonly depicted as directed, bipartite graphs, in which the first node set consists of metabolites and the second node set consists of enzymes. Alternatively, metabolic networks can be represented as directed hypergraphs, where nodes are metabolites and edges connect educts with products. In this case, the edges are labelled with the enzyme or the class of enzymes that is able to catalyse the reaction. Examples for databases storing metabolic information are KEGG [KG00], Reactome [Jos+05], and MetaCyc [Cas+08]. An example for a metabolic (sub)network can be seen in Figure 3.2

Regulatory Networks

In contrast to the network types described above, regulatory networks are less well-defined. In general, a regulatory network comprises interactions that regulate biological processes. Examples for regulatory interactions are transcription factors binding to DNA, thereby regulating gene expression or the activity of kinases which (de)activate other proteins via phosphorylation. While the interactions in a regulatory network are usually directed, sometimes protein-protein binding events

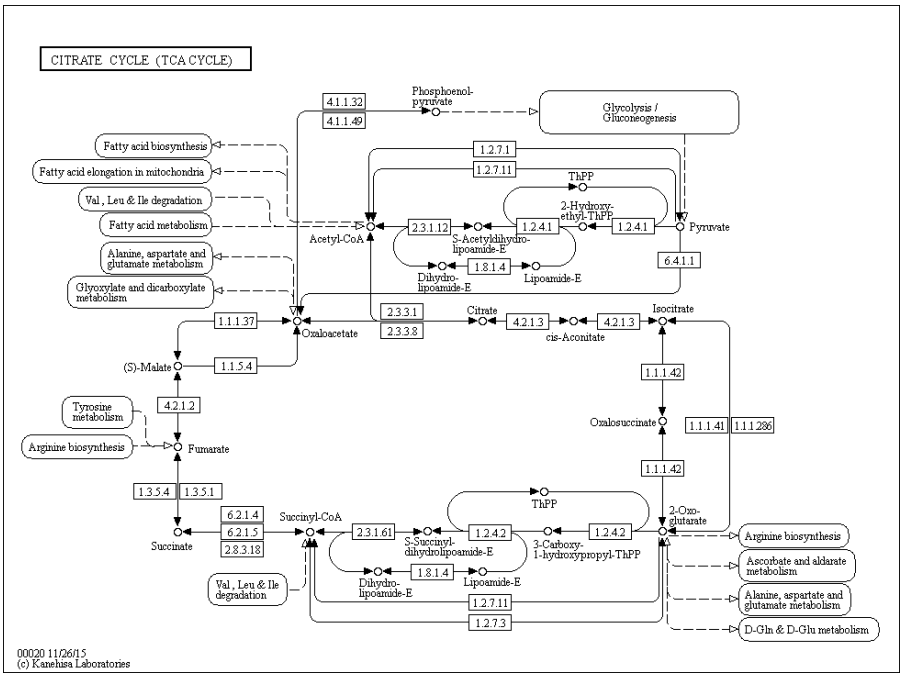


Figure 3.2: KEGG [KG00] reference pathway for the Citrate cycle (TCA cycle). Nodes represent metabolites and edges represent conversions between metabolites. Edge labels are *enzyme commission* numbers [Web+92] that represent the set of enzymes that is able to catalyse this reaction. Dashed edges represent links to and from other metabolic pathways.

are contained in the network, if the resulting complex has a regulatory function. This results in heterogeneous graphs with multiple edge types. When interpreting this network, each of these edge types needs to be treated differently.

KEGG and Reactome contain both, metabolic and regulatory networks.

Regulatory networks usually contain interactions that have been obtained by curating the literature. Most databases organise their data into subnetworks or *regulatory pathways* that represent well defined signalling cascades or biological processes. Databases that are organised in this fashion are KEGG [KG00], Reactome [Jos+05], WikiPathways [Kel+12], and BioCarta [Nis01]. In the case of KEGG, BioCarta, or WikiPathways the pathways are stored isolated from each other. To create a complete regulatory network, they need to be stitched together by the user. An example regulatory network is given in Figure 3.3.

3.2.2 *Network Inference*

The STRING database also incorporates interactions obtained via prediction algorithms.

In the discussion above we already discussed some of the methodology used for constructing biological networks. Most databases are either manually curated, meaning that interactions are only inserted into the database after being reviewed by an editor, or employ text mining algorithms, which automatically extract interactions from the literature. In both cases, the interaction information is usually based on specialised assays that provide evidence for an interaction. To exploit available

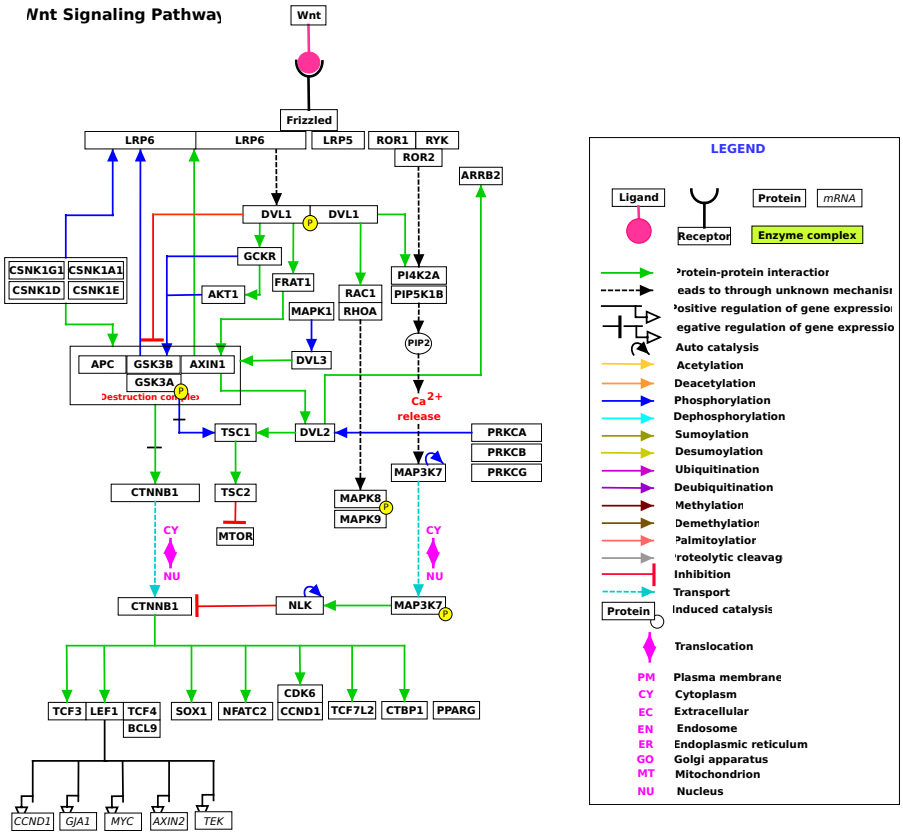


Figure 3.3: WNT pathway obtained from WikiPathways [Kel+12; Kan+10a]. Rectangular nodes represent proteins and mRNA whereas edges represent regulatory interactions.

high-throughput datasets, models using supervised or unsupervised techniques from statistics and machine learning, which can extract networks from expression profiles, have been developed [YVK04; MV08; DLS00]. In this chapter, we discuss the theory behind three types of models that are commonly used for network inference: co-expression networks, Bayesian networks (BNs) and Gaussian graphical models (GGMs). We will only shortly touch co-expression networks, as they are closely related to, but not identical with GGMs. We will provide more details on BNs and GGMs, which are based on substantial corpus of theoretical work. Both network types are graphical representations of probability distributions. Due to this, they are commonly referred to as *probabilistic graphical models* or simply *graphical models*. As they are based on an explicit mathematical model, each interaction (or absence thereof) has a well defined interpretation.

Co-expression Networks

Co-expression networks capture associations between genes or, to be more precise, between the expression patterns of genes. They can be interpreted as an approximation to regulatory networks, in the sense that two genes being co-expressed can be an indication for the presence of a regulatory mechanism. The associations are commonly inferred from

microarray or RNA-seq expression datasets (cf. Section 2.3) by computing the Pearson or Spearman (cf. Section 3.5.3 and Appendix A.1.2) correlation between the measured genes. In theory, co-expression networks can be created from any expression dataset. In practice, a large number of samples is required to reduce the noise inherent to expression data, though. COXPRESdb [Oka+14] focuses on co-expression networks for common model organisms. Most networks therein are derived from microarray expression data obtained from public repositories. In contrast, the GeneFriends [DCM15] database is built upon expression profiles measured using the Illumnia HiSeq2000 platform. Other databases such as BAR [Tou+05] or GeneCat [Mut+08] are dedicated to plants.

3.3 CAUSAL BAYESIAN NETWORKS

A more thorough introduction to (C)BNs can be found in Pearl [Pea09] or Koller and Friedman [KF09].

Often, regulatory networks are constructed from interactions identified in the literature. Examining the network structure gives insight into the cellular processes in which the nodes of the network take part. To make a quantifiable prediction about the behaviour of the system, the representation purely as a graph is, however, insufficient. For this, the network needs to be adorned with a mechanistic or probabilistic interpretation. One such class of interpretable networks are *Bayesian networks (BNs)*. Bayesian networks are *directed, acyclic graphs (DAGs)*. Each node represents a random variable and each edge encodes a dependence of the target on the source. Due to this structure, BNs are well suited to model hierarchical dependencies and are especially popular for modelling signalling cascades. A small example is given in Figure 3.4. Outside of computational biology, BNs are often employed for the construction of expert systems [SC92] in medicine [Die+97], epidemiology [Won+03], psychology [GT07], and in economics [BA00; CVY07]. Let us start by giving a more formal definition of a BN. For this, we need to define when two variables are conditionally independent.

Definition 6 (Conditional Independence [Pea09])**.** Let X, Y, Z be sets of random variables. We call X and Z conditionally independent given Y if

$$\Pr(X \mid Y, Z) = \Pr(X \mid Y)$$

holds whenever $\Pr(Y, Z) > 0$.

Using the notion of conditional independence, it is now possible to give a definition of a BN.

Definition 7 (Bayesian Network [KF09])**.** A Bayesian network is a directed acyclic Graph $G(V, E)$, where each node $v \in V$ represents a random variable. Let $\text{pa}_v := \{w \mid (w, v) \in E\}$ denote the parents of v. For every node $u \in V$ which is not a descendant of v, it holds that u and v are conditionally independent given pa_v (*Markov property*).

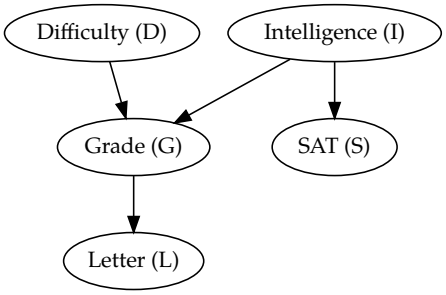


Figure 3.4: The *student network* is an example of a small BN. Each of the nodes represents a binary variable. Whether a student earns a good grade (G), is dependent on whether the course was difficult (D) and whether the student possesses a low or a high intelligence (I). Based on the grade, a lecturer may be more or less inclined to write a letter of recommendation (L). Whether the student passes the SAT test only depends on the student's intelligence.

Given a Bayesian network for a probability distribution, we can write the joint probability density as a product of conditional probabilities:

$$\Pr(X_1, \ldots, X_n) = \prod_{k=1}^{n} \Pr(X_k \mid \mathrm{pa}_{X_k})$$

Often, an edge from parent to child in a BN is assumed to model a causal relationship. Yet, this is *not* true in general. To see why, let us take a look at the chain rule that can be used to represent a joint probability distribution as a product of conditional distributions:

$$\Pr(X_1, \ldots, X_n) = \prod_{k=1}^{n} \Pr(X_k \mid X_1, \ldots, X_{k-1})$$

To proof this, use $\Pr(X|Y) = \Pr(X,Y)/\Pr(Y)$ *and collapse the resulting telescoping product.*

Clearly, this factorisation gives us a simple way to represent every probability distribution as a BN. Each factor in the product represents a child (X_k) to parent(s) ($X_1, \ldots, X_{k-1}$) relation. However, this decomposition is not unique, as it solely depends on the ordering of the variables [Pea09]. As a consequence, we obtain different graph structures, depending on the ordering of the variables. In fact, it is possible to generate cases in which node X_i is a parent of X_j and cases where the opposite is true. If a causal relationship between the two variables exists, only one of the directions encodes this relationship faithfully. Hence, the topological order in a BN cannot be assumed to model a causal relationship.

When *does* an edge in a BN stand for an actual, causal dependency? To answer this question, it is important to understand the concept of interventions. By observing a system, e.g. by measuring the expression of all genes in a cell, for a sufficient amount of time we are able to deduce patterns in its behaviour. An example for such a pattern would be: "if node A is in state a_1, then node B is in state b_2". However, these observations are only sufficient to conclude that the state of the two nodes is correlated. It is uncertain whether node B is in state b_2 because node A is in state a_1 or vice-versa. It is also possible that the state of the

two nodes causally depends on a third variable and neither direction represents a valid causal relationship. To determine which of the three possibilities is the case, the experimental setup needs to be changed. Instead of passively observing the system, an external perturbation must be introduced. For example, if the state of a node A is artificially fixed at state a_1, we can observe the state of node B. If B remains in state b_2 regardless of the other variables, this indicates that the state of A is causal for the state of B. An example for such perturbations in a biological setting is the artificial knock-out or overexpression of a gene.

Pearl [Pea09] realised that the concept of interventions is not only helpful for detecting causal relationships, but also is fundamental for giving a sound definition of causality. Consider the factorisation of the joint probability induced by the student network (Figure 3.4):

$$\Pr(D, I, G, S, L) = \Pr(L \mid G) \Pr(G \mid D, I) \Pr(S \mid I) \Pr(D) \Pr(I)$$

If we want to determine the probability of getting a letter in the case that we *know* the grade is good, we set the value of G to "good" and remove the factor $\Pr(G \mid D, I)$ from the factorisation:

$$\Pr_{G=\text{good}}(D, I, S, L) = \Pr(L \mid G = \text{good}) \Pr(S \mid I) \Pr(D) \Pr(I)$$

We call $\Pr_{G=\text{good}}$ an *interventional distribution*. This allows use to define a *causal Bayesian network (CBN)*.

Definition 8 (Causal Bayesian network [Pea09])**.** A *causal Bayesian network (CBN)* for a distribution P is a Bayesian network which is consistent with all possible *interventional distributions* $P_{X=x}$. Here, consistent means that the network encodes a valid factorisation of $P_{X=x}$ after the node corresponding to X and all edges going into X have been deleted.

Now we can provide an answer to our initial question: an edge in a BN can only be interpreted as causal, if the BN is, in fact, a CBN.

Due to the lack of measurements of exogenous influences, a CBN can not always be uniquely determined. The resulting set of equivalent CBNs is called a causal structure.

This definition gives a natural way to examine the effect of external influences on the observed system. Given a CBN, the effect of e.g. a gene knock-out can be modelled by setting the state of the node representing the target gene to unexpressed and deleting all edges from the parent nodes. As the CBN is consistent with the corresponding interventional distribution, we can now query the network for probabilities using standard algorithms. This approach was formalised by Pearl [Pea95a] as the so-called *do-calculus*. It allows to test hypotheses on how external changes (*interventions*) such as our gene knockout affect a system's behaviour. Using the do-calculus it is also possible to answer *counterfactual* questions such as: "Would the patient have recovered when given drug B, knowing that he did not recover given drug A?" or "Would the plane have crashed, if it had been inspected before take-off, given that it had not been inspected before?". As such, counterfactuals allow to re-evaluate decisions retrospectively in the light of new evidence.

Due to the above properties, BNs and CBNs are popular tools in bioinformatics and have been applied in many scenarios [M+99; Fri+00;

Hus03; Sac+05]. Accordingly, a considerable amount of software has been written to train and work with BNs. In general, the available tools solve problems from one of three classes: *topology determination*, *parameter training*, and *reasoning*. Naturally, before being able to use a BN, its topology must be determined. Unfortunately, even the inference of an approximate network topology has been shown to be NP-complete [DL93]. To work around this limitation, a wide variety of heuristic algorithms, ranging from greedy to Monte-Carlo approaches, has been conceived. Examples for such tools are, among many others, BANJO [Smi+06], BNFinder2 [Doj+13], or SMILE [Dru99]. CBNs can be determined using variants of the *inductive causation (IC)* algorithm. In practice the most commonly used implementation of the IC algorithm is the PC algorithm [SGS00] as implemented in the pcalg R package [Kal+12]. Once the topology of the BN has been determined, the parameters of the network need to be derived from data. As the parameters simply correspond to the conditional probabilities of a node given its parents, this can be accomplished using trivial counting statistics in the case of discrete data. For scenarios where only few samples are available, regularisation techniques such as Laplace/Lidstone smoothing [CG96] should be utilised. In case of missing data, an efficient *Expectation-Maximisation (EM)* [DLR77] scheme can be applied [KF09]. While many tools that infer a topology also compute *conditional probability tables (CPTs)* as they go, approaches based on the IC algorithm require an additional estimation step. Furthermore, if new data becomes available, retraining the parameters may be preferable to recomputing the network topology. After parameter training, the model can be subjected to *Bayesian reasoning*. Under Bayesian reasoning we understand the computation of conditional probabilities given a network structure and trained parameters. In this framework, queries such as "In which state is node B, if node A is in state a_3?" or "What is the probability of node C being in state c_1, if node A and node B are in states a_1 and b_1, respectively?" can be answered. If the BN under investigation is a CBN, it is also possible to evaluate interventional and counterfactual queries as explained above. We call this *causal reasoning*.

A comprehensive list of software for BNs is maintained by Kevin Murphy at `https://www.cs.ubc.ca/~murphyk/Software/bnsoft.html`

Evaluating interventions or counterfactuals is possible for every BN topology; But not necessarily meaningful.

While a large selection of programs for performing Bayesian reasoning, such as SMILE [Dru99] or the Bayes Net Toolbox [Mur+01], exists, hardly any tool supports causal reasoning. To fill this gap, we developed *CausalTrail* [Sch15a; Stö+15], a tool for performing causal reasoning using the do-calculus. In the remainder of this section, we give insights into CausalTrail's implementation and give examples of possible application scenarios.

3.3.1 *CausalTrail*

CONTRIBUTIONS The original implementation of CausalTrail was written by Florian Schmidt [Sch15a] in the context of his Master's thesis supervised by me. The publication on CausalTrail [Stö+15]

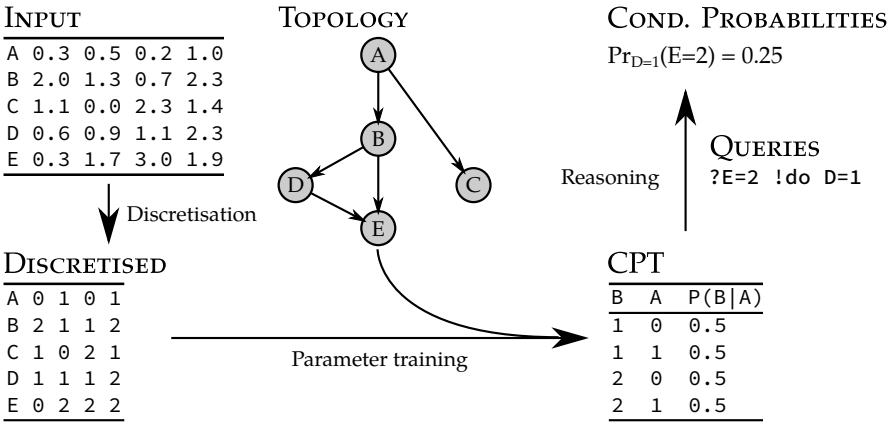


Figure 3.5: The flow of information in CausalTrail. The user provides input data as well as a predetermined topology. After a discretisation step, the input data is used to determine parameters for the topology. This yields the conditional probability parameters that fully describe the joint probability distribution. Using the query language Bayesian and causal reasoning can be performed.

was written by me with substantial feedback by Florian Schmidt and Hans-Peter Lenhof. Improvements to the initial code were contributed by me and Florian Schmidt.

Unlike other tools for working with BN structures, CausalTrail does not attempt to infer the topology of a BN. Instead, the topology as well as data from which parameters should be learned are specified by the user. For this, software such as the pcalg [Kal+12] package can be used. Afterwards, CausalTrail allows to perform Bayesian and causal reasoning using a flexible query language. To our knowledge, CausalTrail is the only freely available software that allows to perform causal reasoning, in addition to Bayesian reasoning, on CBNs. The only other tool we are aware of that also support this is the commercial BayesiaLab software[1].

Workflow

A typical workflow is structured as follows (Figure 3.5): the user specifies a network topology which can be provided in the *simple interaction format (SIF)* and *trivial graph format (TGF)* formats (see below). Next, a dataset containing measurements for each node in the network needs to be supplied. Here, it is important to note that the software is able to cope with missing values. This makes CausalTrail especially suited for the analysis of multi-omics datasets, where some measurements may be missing or must be discarded due to quality issues. To simplify the parameter learning process, CausalTrail currently requires that each variable in the network only assumes discrete values. Thus, if

1 `http://www.bayesialab.com/`

the provided data contains continuous variables, the user must select one of several implemented discretisation methods for each such variable. After this step, parameters are trained. If the dataset does not contain missing values, a maximum likelihood estimate is computed. Otherwise, the EM algorithm as described by Koller and Friedman [KF09] is used to provide robust estimates.

Once the model has been trained, the user can perform reasoning using the provided query language (see below). When using the GUI, queries can be created interactively or entered manually (Section 3.3.1). In this case, the validity of the query is checked during runtime.

We will now take a closer look at the implementation of CausalTrail. Further details can be found in the Master's thesis by Schmidt [Sch15a].

File Formats

For specifying topologies the SIF and TGF file formats are supported. The SIF file format is a simple, line-based format. Each line contains one or more edges starting at a single node. Edges are encoded by a single node id followed by an interaction type, followed by multiple node ids:

The SIF and NA file format specification is available at `http://wiki.cytoscape.org/Cytoscape_User_Manual/Network_Formats`.

```
1 -> 2 3
```

The interaction type is ignored by CausalTrail. For specifying interpretable node names, the node attribute (NA) file format is used. Each NA file starts with a header stating the name of the stored attribute and the type of the attribute as a Java class name:

```
NodeName (class=java.lang.String)
1 = NodeA
2 = NodeB
3 = NodeC
```

The TGF file format allows to store node names and edges in a single file. The file starts with a newline-separated list of node ids and their names. After the last node a `#` character is inserted. Afterwards a list of edges follows:

```
1 NodeA
2 NodeB
3 NodeC
#
1 2
1 3
```

For specifying measurements, CausalTrail expects a matrix. Each row in the matrix corresponds to the name of a node in the network, and each column corresponds to a sample. Missing values can be specified using the string `"na"`.

```
{
  "Grade": {
    "method": "bracketmedians", "buckets": "2"
  },
  "Letter": {
    "method": "threshold", "threshold": "2.0"
  },
  "SAT": { "method": "pearsontukey" },
  "Intelligence": { "method": "median" },
  "Difficulty": { "method": "none" }
}
```

Listing 3.1: Example JSON file containing discretisation settings for each variable of the student network.

Discretisation Methods

The *ceiling*, *floor*, and *round* methods discretise the inputs to the nearest integers. In contrast, thresholding-based methods like the *arithmetic* or *harmonic mean*, *median*, *z-score*, and fixed *threshold* methods create binary output. The *bracket medians* and *Pearson-Tukey* [CCB11] procedures yield three or more output classes. Discretisation methods can be directly specified using the GUI or via a JSON formatted input file (Listing 3.1).

Parameter Learning

In the case of a dataset without missing values, parameter learning amounts to counting the frequency with which each combination of parent and child values occurs. If the data contains missing values, the EM procedure described by Koller and Friedman [KF09] is used. The idea behind this algorithm is to estimate the most likely state of a missing value given the current parameters. Based on this *imputation*, a new set of parameters is computed as if no missing values were present. The procedure is iterated until either the parameters converged to a final value or a fixed number of iterations has been reached (Figure 3.6). As the EM algorithm is a greedy procedure [Edm71; DLR77], the computed parameters are not necessarily the globally optimal parameters. To increase the chances of reaching the global optimum, the algorithm is restarted multiple times using different initialisation schemes such as random sampling.

Bayesian and Causal Reasoning

The basic tool for performing Bayesian reasoning is the ability of computing conditional probabilities of the form $\Pr(X = x \mid Z = z)$, where X and Z stand for disjoint sets of variables. To do so, the values of X and Z are kept fixed while all remaining variables $Y \notin X \cup Z$ are

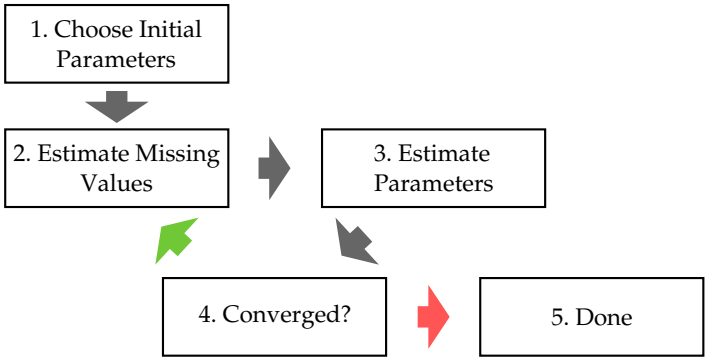


Figure 3.6: The flow of the basic steps of the parameter learning. First a set of initial parameter is guessed. These parameters are used to impute the missing values. New parameters are then estimated from the imputed data. If the parameters only changed slightly compared to the previous iteration, the algorithm terminates.

summed out:

$$\Pr(X = x \mid Z = z) = \sum_{y} \Pr(X = x, Y = y \mid Z = z)$$

This process is called marginalisation. Unfortunately, straight forward implementations of this algorithm are not useful in practice. This is due to the fact that for summing out all variables in Y all possible combinations of their values need to be considered. Even for small examples this quickly leads to a combinatorial explosion of terms inside the sum and hence to exponential runtime. However, in the case of a BN, the factorisation of the joint probability can be exploited to perform the marginalisation more efficiently. The basic idea is that factors which do not have variables in common can be marginalised successively and independently:

$$\Pr(X = x \mid Z = z) = \sum_{y,w} \Pr(x \mid y) \Pr(z \mid w) = \sum_{y} \Pr(x \mid y) \sum_{w} \Pr(z \mid w)$$

This approach to computing conditional probabilities is called the *variable elimination* algorithm. While, in theory, it is possible to create factorisations for which the runtime remains exponential in the number of variables, in practice variable elimination greatly improves the performance for computing conditional probabilities. The algorithm is discussed in detail in Koller and Friedman [KF09].

Using this basic functionality, it is possible to also compute the most likely state of a variable, by simply evaluating all possible value assignments. The implementation of fixed-value interventions is straightforward. For each node that should be fixed, the value of the node is set to the interventional value. Then all edges from the parents to the fixed node are removed. Afterwards the remainder of the query is evaluated using the variable elimination algorithm.

The interventions introduced by the do-calculus allow to compute counterfactuals using a three step procedure consisting of the *abduction*, *action*, and *prediction* stages [Pea09]. For illustration, assume we

Recall that, counterfactuals are questions of the form "Would A have been a_1 had B been b?".

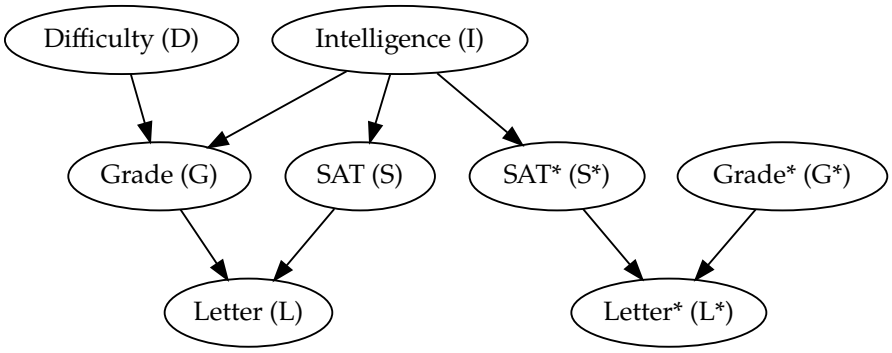


Figure 3.7: Twin network for a modified (added edge between "SAT" and "Letter") student network and the query `? L = 1 | G = 0 ! do G = 1`. Nodes with a * indicate copies of the original network nodes. The exogenous variables "Intelligence" and "Difficulty" are connected to their original children as well as the respective copies. The intervention was executed on the node G^* and thus the edges to D and I were deleted.

want to compute the probability that a student would have gotten a letter of recommendation ($L = 1$) if her grade had been good ($G = 1$), knowing that, in fact, it was bad ($G = 0$). First, the joint probability distribution $\Pr(D, I, G, S, L)$ is updated with the evidence $G = 0$ provided in the query (*abduction*). This yields the *posterior distribution* $\Pr(D, I, G, S, L \mid G' = 0)$. Here we use the notation $G' = 0$ to differentiate between the current variable G and the evidence that we previously observed G in state 0. Second, all interventions in our question are applied (*action*) yielding $\Pr_{G=1}(D, I, S, L \mid G' = 0)$. Last, the desired probability is evaluated on the modified posterior distribution (*prediction*). The problem with this approach is that explicitly computing the posterior distribution can require excessive amounts of memory. This is due to the fact that incorporating the evidence ($G' = 0$) can render the remaining variables dependent which invalidates the factorisation encoded in the CBN. Hence, a full probability table needs to be computed which is often infeasible for larger networks. Instead, CausalTrail uses the twin network approach [Pea09] for computing counterfactuals. In this approach, a copy of all endogenous variables (non-source nodes, here G, L, and S) of the network is created. Exogenous nodes (source nodes, here D and I) retain their original edges and gain a connection to the copies of their children. The interventions are then executed on the copied network. When evaluating the query, the original network is used for conditioning (abduction) and the copy is used for prediction. In Figure 3.7 the twin network for a modified student network is given. It should be noted that counterfactuals can only be formulated for endogenous variables, as only they are fully dependent on the state of the remaining network. In contrast, exogenous variables depend on external influences and, thus, cannot be controlled.

In addition to interventions formalised in the do-calculus, CausalTrail also supports adding and removing edges to and from the network. To this end, the network is simply retrained with the respective edge added to or removed from the topology.

```
<query> ::= '?' queries condition? intervention?
<queries> ::= 'argmax' '(' Node ')'
            | assignments
<condition>    ::= '|' assignments
<intervention> ::= '!' intervs
<assignments>  ::= assign (',' assignments)?
<intervs> ::= ('do' assign | edge) (',' intervs)?
<edge>    ::= ('+'|'-') Node Node
<assign>  ::= Node '=' Value
```

Listing 3.2: Grammar of the CausalTrail Query Language

Query Language

For formulating hypotheses, CausalTrail provides an intuitive query language. The goal of the language is to allow users to specify a conditional probability or interventional expression that should be computed by CausalTrail. Accordingly, the language tries to mimic the respective mathematical notation. Consider the student network from before (Figure 3.4 and Figure 3.8). The question how likely it is that a student receives a letter of recommendation ($L = 1$) given that the student has obtained a good SAT score ($S = 1$) and we know that he has a bad grade ($G = 0$) can be stated as:

$$\Pr_{G=0}(L = 1 \mid S = 1)$$

In the query language the same statement is expressed as follows:

```
? L = 1 | S = 1 ! do G = 0
```

Every query starts with a '?' followed by a list of nodes for which the posterior probability of a certain state should be computed. Alternatively, it is possible to indicate that the most likely state should be computed by using the `argmax` function. Following the '|' character, it is possible to list nodes on which the probability should be conditioned. Similarly, interventions can be stated after '!'. They can be expressed by using the notation `do N = v`. Edge additions and removals between the nodes `N` and `M` are written as `+N M` and `-N M`, respectively. Using edge removal, we can, for instance, query how likely a good score in the SAT is, given that the student had a good grade and under the assumption that the difficulty of the course did not factor into the grade:

```
? S = 1 | G = 1 ! -D G
```

Further example queries are given in Table 3.1. The full grammar is given in Listing 3.2.

Graphical User Interface

All functionality provided by CausalTrail can be accessed via the command line interface (CLI). For convenience and better accessibility we

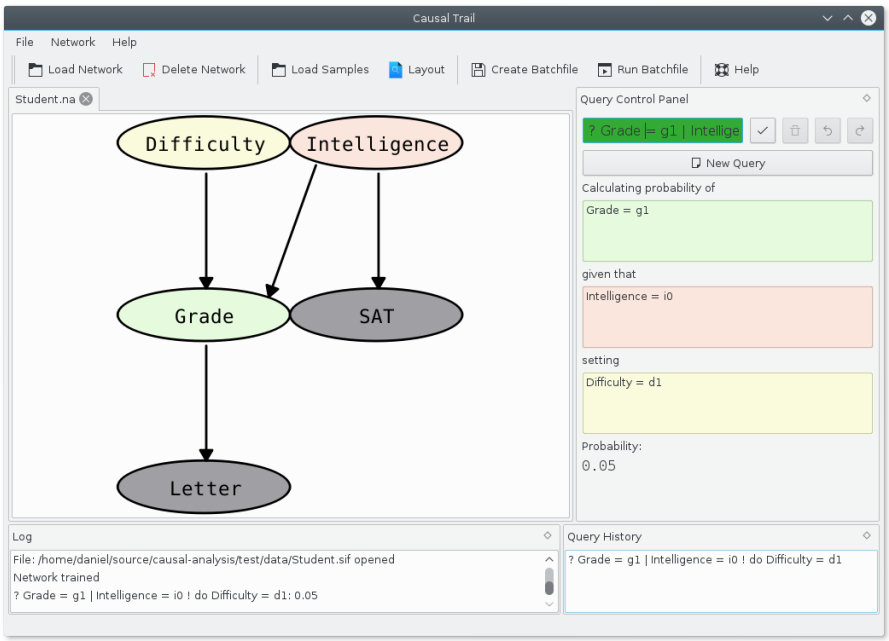
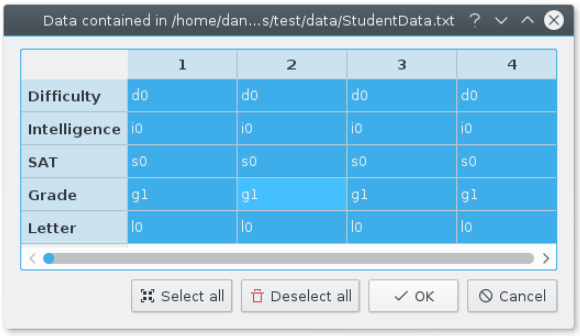
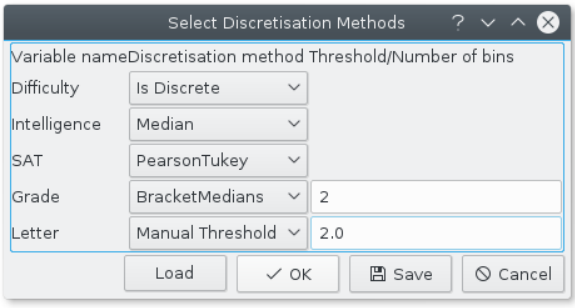


Figure 3.8: Screenshot of the CausalTrail main window. The current network is shown in the central pane. Nodes that are part of the current query (right pane) are coloured according to their role in the query.



(a) Load samples dialog



(b) Select discretisation method dialog

Figure 3.9: When loading samples, first, the dataset can be examined. Next, a discretisation method needs to be chosen for each variable. Selected discretisation methods can be saved to and restored from a JSON file.

also provide a GUI (cf. Figure 3.8). The GUI is centred around a rendering of the Bayesian network as directed, acyclic graph. The layout for this graph is generated using the Graphviz [GN00] suite. Queries can be built intuitively via a context menu that can be invoked by right-clicking on network nodes and edges. Alternatively, queries can be entered directly in the "Query Control Panel" (Figure 3.8, top of right pane). Queries entered this way are automatically validated while typing. If detected, syntactical and semantical errors such as mistyped node names or values, are highlighted. In addition to the plain text representation in the input field, the "Query Control Panel" also provides a structured view, which breaks each query down into components. Nodes or edges that are part of the current query are colour coded in the network representation to improve clarity.

An important part of CausalTrail is the management of datasets. To be able to use a BN for reasoning, a dataset needs to be imported, examined, and discretised. To this end, when a dataset is loaded, an interactive table showing its contents is displayed to the user that can be used to exclude certain samples from the analysis (Section 3.3.1). Afterwards, the user is offered the choice of a discretisation method for each variable. After choosing appropriate methods, the user can store them in a JSON file (cf. Section 5.1.7) for later use. Internally, CausalTrail manages the state of the application as a session. In each session, multiple networks can be loaded and worked on simultaneously. The state of the session can be saved and restored at any point.

Implementation

CausalTrail is written in C++ using features from the C++14 standard[2]. To facilitate code reuse, all GUI agnostic functionality such as routines concerning parameter inference and reasoning are implemented in a separate, `core` library. As dependency only the Boost library collection is used[3]. On top of the `core` library the command line interface (CLI) and a library providing a graphical user interface (GUI) have been built. For creating the GUI the Qt5 toolkit[4] was used, which allows to port CausalTrail to all major platforms. For computing graph layouts, the Graphviz [GN00] suite is automatically detected and used at runtime.

We put special emphasis on the performance and reliability of the implemented methods. To this end, CausalTrail is equipped with a unit test suite written using the Google Test framework[5]. CMake[6] is used for the build system. CausalTrail can be compiled on Linux and Windows, although only the former platform is officially supported. The source code is licensed under GPLv3 and can be obtained from Github[7].

2 `http://www.iso.org/iso/catalogue_detail.htm?csnumber=64029`
3 `http://www.boost.org/`
4 `https://www.qt.io/`
5 `https://github.com/google/googletest`
6 `https://cmake.org/`
7 `https://github.com/unisb-bioinf/causaltrail`

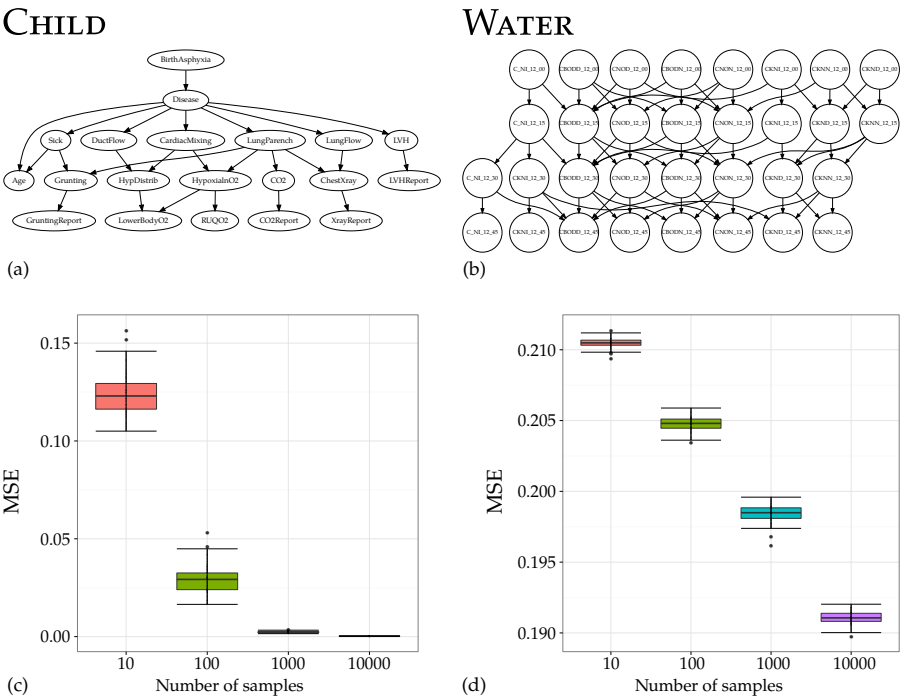


Figure 3.10: Dependence of the parameter fitting procedure on the number of samples and network topology. The performance is shown for (a) the CHILD (20 nodes, 25 edges) and (b) the WATER (32 nodes, 66 edges) networks obtained from the *Bayesian Network Repository* [Scu16]. With increasing sample count a clear improvement in performance can be seen. In the case of the CHILD (c) network, around 1000 samples are enough to achieve excellent parameter fits. The more complex WATER network (d) requires substantially more data. (Graphs taken from Schmidt [Sch15a]).

3.3.2 *Examples*

We first demonstrate the quality and convergence properties of the implemented parameter learning algorithm. For this purpose we take a look at the mean square error (MSE) achieved for the parameters by training on data sampled from two benchmark networks. These are the CHILD network, which models child disease due to birth complications [SC92], and the WATER network, which describes an expert system for the monitoring of waste water treatment [Jen+89]. Both networks were retrieved from the "Bayesian Network Repository" [Scu16] and are of medium size. We chose these networks, to show that not only the number of nodes, but also the connectivity between them can be a limiting factor for the quality of parameter learning. For each network we drew a varying number of random samples from the joint distribution. With an increasing number of samples, the parameters for the CHILD network, consisting of 20 nodes and 25 edges, quickly converge to the true values. Already with just 100 samples, the estimates reach an acceptable error level of ≈ 0.03. The more complex WATER network (32 nodes, 66 edges) requires substantially more data points to obtain good estimates. Even with 10,000 samples, the MSE remains at ≈ 0.19. For

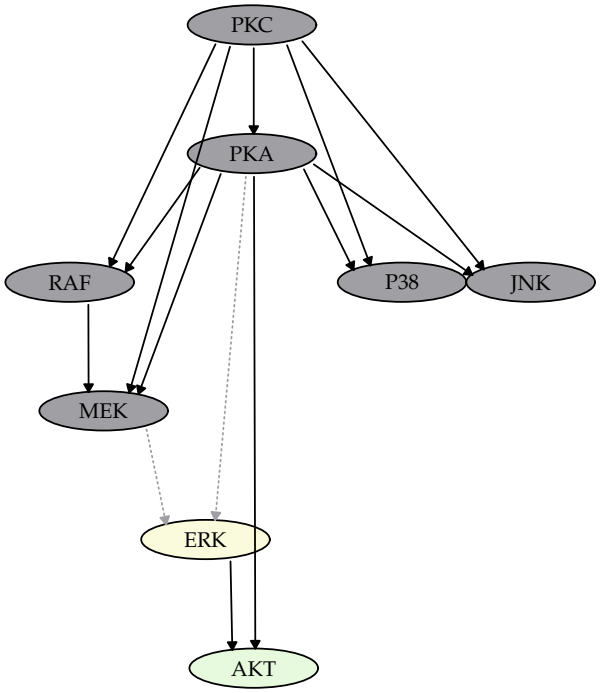


Figure 3.11: The CBN constructed by Sachs et al. [Sac+05], rendered using CausalTrail's SVG export functionality. Nodes represent proteins and edges represent causal dependencies between phosphorylation states. Nodes for which probabilities should be computed are coloured light green. Nodes with fixed values due to an intervention are coloured light yellow. The dashed edges are not considered during evaluation due to the intervention on ERK.

both networks, parameter training is completed in a few milliseconds. Considering that the network parameters are probabilities, an error of 0.19 hints at severe problems in the training procedure. While both networks have a similar number of nodes, the CHILD network is substantially less densely connected. This immediately has an impact on the size of the required CPTs. In addition, the WATER network also has, on average, more states for each node than the CHILD network. This again increases the size of the CPTs and leads to more variability during parameter training.

We further demonstrate an application of CausalTrail using the protein signalling network inferred by Sachs et al. [Sac+05]. Here, we consider the largest connected component of the network which consists of eight nodes. Each node represents a protein for which the degree of phosphorylation was measured. For inferring the topology of the network, the authors used single-cell data obtained from $CD4^+$ primary cells using *multicolour flow cytometry*. This technique allows to determine the phosphorylation state of multiple target proteins using fluorescently labelled antibodies on a single cell level [PN02]. A visualisation of the network, created using CausalTrail's SVG export functionality, is given in Figure 3.11. The authors validated the existence of the edge between ERK and AKT by showing that an intervention on ERK changes the phosphorylation level of AKT, but has no effect on PKA. To this end, the phosphorylation of AKT and PKA was measured with

Query	Result	Probability
`? argmax(AKT)`	1	0.354
`? argmax(AKT) ! do ERK = 2`	2	0.774
`? argmax(AKT) ! do ERK = 0`	0	0.691
`? argmax(PKA)`	2	0.336
`? argmax(PKA) ! do ERK = 2`	2	0.336
`? argmax(PKA) ! do ERK = 0`	2	0.336
`? argmax(PKA) | ERK = 2`	2	0.505
`? argmax(PKA) | ERK = 0`	0	0.423

Table 3.1: Example queries for the Sachs et al. [Sac+05] dataset. High phosphorylation levels for ERK increase the likelihood of AKT being phosphorylated. In contrast, no such influence is detectable for PKA. The last two rows show the effect of *conditioning* on ERK.

ERK being (i) unperturbed, (ii) stimulated, and (iii) knocked down using siRNAs. Stimulation of ERK was achieved using antibodies targeting CD3 and CD28. Whereas this stimulation had no effect on PKA, it lead to an increase in AKT phosphorylation. For the knockdown, again no change of PKA phosphorylation could be detected whilst the phosphorylation of AKT dropped slightly below the level of the unperturbed case.

To test whether the inferred network models the experimental data faithfully, we used the dataset and topology provided by Sachs et al. [Sac+05] to train the parameters of a CBN. We then examined the edge between ERK and AKT more closely. The dataset contains 11672 measurements of each protein's phosphorylation level. These levels were discretised into the classes `low` (0), `medium` (1), and `high` (2) using the *bracket medians* procedure. We then computed the most likely phosphorylation state of AKT and PKA in (i) unperturbed, (ii) stimulated, and (iii) ERK knockout cells, which we modelled using interventions that fix the ERK phosphorylation level to `high` and `low`, respectively. The computed queries are given in Table 3.1. We find that the *in silico* stimulation of ERK leads to an increased AKT phosphorylation level. When ERK is knocked out, AKT phosphorylation drops to `low`, showing that the previous increase was in fact mediated by ERK. In contrast the activity of ERK has no effect on the phosphorylation of PKA. Note that using an intervention is essential for this observation as conditioning on ERK would render PKA dependent on ERK resulting in a different prediction.

3·3·3 *Summary*

While tools for Bayesian reasoning are commonly available, no freely available programs for causal reasoning exist. With CausalTrail we attempt to fill this gap. CausalTrail enables its users to harness the additional expressivity offered by the do-calculus to formulate and test

biological hypotheses *in silico*. Our software offers efficient implementations for parameter learning and query evaluation that allow to examine experimental data in an interactive fashion. The showcased application of causal reasoning demonstrates that CausalTrail may be a valuable addition to a bioinformatician's toolbox for the interpretation of Bayesian networks.

3.4 GAUSSIAN GRAPHICAL MODELS

Besides Bayesian networks, *Gaussian graphical models (GGMs)* are a popular choice for inferring network structure from data. In contrast to BNs, GGMs are undirected and encode a multivariate Gaussian distribution. This means that two nodes in a GGM are connected if and only if the corresponding random variables are conditionally dependent. As expression data is commonly assumed to be approximately normal distributed, GGMs are often used for analysing the dependence structure between gene expression measurements. By assuming a concrete underlying distribution, the model is more constrained than BNs. Still, the size of the networks that can be reliably estimated in practice strongly depends on the number of available samples. In the following, we discuss GGMs in more detail. To this end, we introduce the central notions of *precision matrix* and *partial correlations*. We continue with methods for training GGMs and give a short example of a GGM trained on the Wilm's tumour dataset.

Normality of expression data is a strong and not necessarily true assumption. See Section 4.3.

3.4.1 *Multivariate Gaussian Distributions*

The multivariate Gaussian distribution is a generalisation of the Gaussian distribution to higher dimensional spaces $\mathbb{R}^p$. For this, the mean $\mu \in \mathbb{R}$ is replaced by the vector of means $\vec{\mu} \in \mathbb{R}^p$. For the variance σ^2 scaling up to a multivariate distribution is not as simple. This is due to the fact that the constituent random variables of the distribution can be dependent. To account for this, a covariance matrix $\Sigma \in \mathbb{R}^{p \times p}$ is used. Its diagonal elements are the variances of each random variable, while the off-diagonal elements correspond to the covariance between pairs of random variables. If all variables are independent, the covariance matrix thus reduces to a diagonal matrix. The density function of a multivariate Gaussian distribution is given by

$$f(\vec{x}) = \frac{1}{\sqrt{(2\pi)^p \det(\Sigma)}} \exp\left(-\frac{1}{2}(\vec{x} - \vec{\mu})^t \Sigma^{-1} (\vec{x} - \vec{\mu})\right)$$

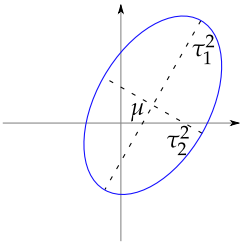


Figure 3.12: Two-dimensional multivariate Gaussian distribution. The principal variances $\tau_{1,2}^2$ correspond to the eigenvalues of Σ.

The matrix $\Omega := \Sigma^{-1}$ is called the *precision matrix*. It can be interpreted as a transformation that rescales the samples such that their covariance matrix becomes the identity. This idea is pursued further in Section 4.4. In addition, Ω also contains information about the conditional dependences between the dimensions. Whereas Σ contains the covariance of the constituent random variables, Ω contains the *partial covariances*. The

The precision matrix is also known as concentration or, less common, information matrix.

partial covariance between two random variables can be understood as the part of their covariance that cannot be explained by the remaining variables. To explain this in more detail, we first introduce the notion of a least-squares fit. Assume that $f : \mathbb{R}^p \to \mathbb{R}$ is a function of the form $f(\vec{x}) = \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p$. Furthermore, assume that the *outcome* y is determined via the relation $y = f(\vec{x}) + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2)$ is a normally distributed noise variable. The task is to find the best estimate f^* of f given a set of N samples $(y_i, \vec{x}_i)$. Here, best is defined as the function minimising the least-squares loss function:

$$\sum_{i=1}^{N} (y_i - f(\vec{x}_i))^2 = \sum_{i=1}^{N} \left(y_i - \langle \vec{x}_i, \vec{\beta} \rangle \right)^2 = \langle \vec{y} - \boldsymbol{X}\vec{\beta}, \vec{y} - \boldsymbol{X}\vec{\beta} \rangle$$

Note that, assuming centred predictors, $\boldsymbol{X}^t\boldsymbol{X}$ is proportional to the covariance matrix.

Here, we use $\vec{y} \in \mathbb{R}^N$ as the vector of observations and $\boldsymbol{X} \in \mathbb{R}^{N \times p}$ as the matrix of predictors. After applying basic calculus, we obtain a formula for the weight vector [FHT09]: $\vec{\beta}^* = (\boldsymbol{X}^t\boldsymbol{X})^{-1}\boldsymbol{X}\vec{y}$. We refer to f^* as the *least-squares fit* of $\vec{y}$ given $\boldsymbol{X}$ with coefficients $\vec{\beta}^*$. It is now possible to define the partial covariance:

Definition 9 (Partial Covariance [BSS04])**.** Let X, Y be two random variables and let $Z = \{Z_1, \ldots, Z_n\}$ be a set of random variables. Denote the coefficients of the least-squares fit of X and Y given Z as β_X and β_Y, respectively. Then, the partial covariance is defined as the covariance of the residuals:

$$\begin{aligned} r_X &:= X - \langle \vec{Z}, \vec{\beta_X} \rangle \\ r_Y &:= Y - \langle \vec{Z}, \vec{\beta_Y} \rangle \\ \operatorname{cov}(X, Y|Z) &:= \operatorname{cov}(r_X, r_Y) \end{aligned}$$

As with the usual variance the partial covariance permits the definition of a measure of correlation. This correlation is called the *partial correlation* (cf. [BSS04]).

Definition 10 (Partial Correlation)**.** Let X, Y be two random variables and let $Z = \{Z_1, \ldots, Z_n\}$ be a set of random variables. The partial correlation of X and Y is defined as

$$\rho_{XY|Z} := \frac{\operatorname{cov}(X, Y|Z)}{\sqrt{\operatorname{var}(X|Z)}\sqrt{\operatorname{var}(Y|Z)}}$$

using $\operatorname{var}(X|Z) := \operatorname{cov}(X, X|Z)$ or, given Ω:

$$\rho_{XY|Z} := -\frac{\Omega_{XY}}{\sqrt{\Omega_{XX}\Omega_{YY}}}$$

It can be shown that two variables in a GGM are conditionally independent if and only if their partial correlation equals zero [Lau96; BSS04]. Equivalently, two distinct random variables are conditionally independent if and only if the corresponding entry in Ω is zero.

3.4.2 *Covariance Matrix Estimation*

In theory, fitting a GGM is straightforward. First, the sample covariance matrix is computed. Given a dataset $X = \{x_1, \ldots, x_n\}$ with $n \in \mathbb{N}$ samples and $p \in \mathbb{N}$ variables. The sample covariance matrix $\boldsymbol{W}$ is then given by:

$$\boldsymbol{W} = \frac{1}{n-1} \sum_{i=1}^{n} (\vec{x_i} - \vec{\mu})(\vec{x_i} - \vec{\mu})^t$$

where $\vec{\mu}$ is the sample mean. More robust estimates of the (co)variances can be used to obtain estimates of the precision matrix that are less susceptible to noise in the input data. An example is the method proposed by Schäfer and Strimmer [SS05] which is based on James-Stein shrinkage [JS61]. For details the reader is referred to the cited literature.

More information on James-Stein Shrinkage can be found in Section 4.2.2.

3.4.3 *Precision Matrix Estimation*

The precision matrix can be obtained by computing the inverse of $\boldsymbol{W}$. In practice, the required matrix inversion creates various problems. If few samples are available, compared to the number of variables, small errors in the estimation of the covariance matrix are amplified by the *condition number* of the matrix [Pre+07]. For small sample sizes, where $\boldsymbol{W}$ is close to singular, the condition number is likely to be high. Accordingly, large differences between the true and estimated precision matrix can be expected. To alleviate the effect of small sample sizes, a regularisation approach should be chosen. A direct way to introduce regularisation is to exploit the eigenvalue decomposition of Σ. Given a symmetric matrix $\boldsymbol{A} \in \mathbb{R}^{p \times p}$, its eigenvalue decomposition is given by

The condition number of a matrix is the ratio of the magnitudes of the highest and lowest singular value.

$$\boldsymbol{A} = \boldsymbol{V} \Gamma \boldsymbol{V}^t$$

where $\boldsymbol{V} \in \mathbb{R}^{p \times p}$ is the orthogonal matrix of eigenvectors and $\Gamma \in \mathbb{R}^{p \times p}$ is the diagonal matrix of eigenvalues. As Σ is positive definite, the eigenvalues are guaranteed to be non-negative. It is then possible to compute $\Sigma^{-1} = \boldsymbol{V} \Gamma^{-1} \boldsymbol{V}^t$ where $\tilde{\Gamma}^{-1}$ is the (pseudo-)inverse of Γ:

$$\Gamma_{ii}^{-1} = \begin{cases} 1/\Gamma_{\boldsymbol{ii}} & \text{if } \Gamma_{\boldsymbol{ii}} > 0 \\ 0 & \text{otherwise} \end{cases} \tag{3.1}$$

For small eigenvalues this results in large entries in Γ^{-1}. Under the assumption that eigenvectors corresponding to small eigenvalues can mostly be attributed to noise, setting the inverse of these small eigenvalues to zero prevents the amplification of this noise. We thus define $\tilde{\Gamma}^{-1}$ in which only eigenvalues above a user-defined threshold $\rho \in \mathbb{R}_0^+$ are retained:

$$\tilde{\Gamma}_{ii}^{-1} = \begin{cases} 1/\Gamma_{\boldsymbol{ii}} & \text{if } \Gamma_{\boldsymbol{ii}} > \rho \\ 0 & \text{otherwise} \end{cases} \tag{3.2}$$

A similar idea can be derived from linear regression. As linear regression requires the inversion of a covariance matrix to compute the

The L_2 regularisation in ridge regression is a special case of Tikhonov regularisation [Tik63].

regression coefficients, it also suffers from cases where Σ is near singular and, thus, has a high condition number. To counteract this, a small constant can be added to the diagonal of the covariance matrix to serve as a regulariser. This ensures that the solution to the regression problem is unique. Ledoit and Wolf [LW04] explore this idea for computing a precision matrix and describe an optimisation problem that allows to obtain the optimal value of the regularisation parameter.

If only few variables are expected to be conditionally dependent, an estimator that prefers a sparse precision matrix is advantageous. To this end, a considerable amount of research has been dedicated towards the computation of L_1 penalised estimates of the precision matrix. The idea of using a L_1 penalty term stems from the popular lasso regression [Tib96] procedure, which tends toward setting small coefficient to zero, thereby yielding sparse linear models. For the estimate of a precision matrix Ω the penalised log-likelihood is given by

$$l(\Theta) = \log(\det(\Theta)) - \mathrm{tr}(\Sigma\Theta) - \rho||\Theta||_1 \tag{3.3}$$

where Θ denotes the estimate for Ω, tr denotes the trace of a matrix and $||.||_1$ is the L_1 matrix norm. The parameter $\rho \geq 0$ determines the strength of the regularisation. For $\rho = 0$, Equation (3.3) reduces to the conventional log-likelihood of the precision matrix. For larger values of ρ the size of the entries of Θ is increasingly penalised and thus the resulting matrix tends to include less non-zero values. To see that Equation (3.3) in fact corresponds to the log-likelihood, we start with the logarithm of the probability density. W.l.o.g. we assume that the mean of the samples $\vec{x}_i$ is zero.

$$\begin{aligned}
L(\Theta) &= \log\left[\prod_{i=1}^{N} \frac{\det(\Theta)^{1/2}}{(2\pi)^{p/2}} \exp(-\frac{1}{2}\vec{x}_i^t\Theta\vec{x}_i)\right] \\
&= \sum_{i=1}^{N} \log\left[\frac{\det(\Theta)^{1/2}}{(2\pi)^{p/2}} \exp\left(-\frac{1}{2}\vec{x}_i^t\Theta\vec{x}_i\right)\right] \\
&= \frac{N}{2}\log(\det(\Theta)) - \frac{Np}{2}\log(2\pi) - \frac{1}{2}\sum_{i=1}^{N}\vec{x}_i^t\Theta\vec{x}_i \\
&= \frac{N}{2}\log(\det(\Theta)) - \frac{Np}{2}\log(2\pi) - \frac{1}{2}\sum_{i=1}^{N}\sum_{l=1}^{p}\sum_{k=1}^{p} x_{il}\Theta_{lk}x_{ik} \\
&= \frac{N}{2}\log(\det(\Theta)) - \frac{Np}{2}\log(2\pi) - \frac{1}{2}\sum_{l=1}^{p}\sum_{k=1}^{p}\Theta_{lk}(\boldsymbol{X}^t\boldsymbol{X})_{lk} \\
&= \frac{N}{2}\log(\det(\Theta)) - \frac{Np}{2}\log(2\pi) - \frac{1}{2}\sum_{l=1}^{p}(\Theta(\boldsymbol{X}^t\boldsymbol{X}))_{ll} \\
&= \frac{N}{2}\log(\det(\Theta)) - \frac{Np}{2}\log(2\pi) - \frac{N}{2}\mathrm{tr}(\Theta\Sigma)
\end{aligned}$$

For fitting the model we can omit the constant $(Np/2)\log(2\pi)$ and rescale with $2/N$ to obtain Equation (3.3) without the penalty term. To

find an approximate solution, Meinshausen and Bühlmann [MB06] employ the lasso in an iterative fashion to compute the partial covariances as in Definition 9. Based on this work and the results of Banerjee, El Ghaoui, and d'Aspremont [BEd08], Friedman, Hastie, and Tibshirani [FHT08] devised the graphical lasso approach. In contrast to the method proposed by Meinshausen and Bühlmann [MB06], their formulation solves the problem accurately and offers a significantly improved runtime behaviour for sparse precision matrices. As optimisation procedure Friedman, Hastie, and Tibshirani [FHT08] propose a cyclic gradient-descent-based method, which they published as the `glasso` R [R C16] package. The principle behind the cyclic gradient decent is the realisation that all entries of the precision matrix are dependent. Instead of optimising one variable completely, each variable takes a short step in the direction of the (sub-)gradient in a round-robin fashion. Cai, Liu, and Luo [CLL11] propose an alternate optimisation target, which they solve for using a set of linear programs. In the remainder of this work, we restrict ourselves to the more popular graphical lasso approach.

3.4.4 *Example*

To illustrate a possible application of GGMs we fitted a coexpression model using data from the Wilm's tumour dataset (Section 2.4) as input. To restrict the number of genes to a manageable size, we used the TFs found in the consensus network published by Juan et al. [Jua+16], which is based on the colocalisation of epigenomic marks. The network itself was determined using a GGM based on ChIP-seq data. Edge directions were added based on prior knowledge.

First, we computed the precision matrix for all 47 TFs contained in the network by pseudo inversion of the covariance matrix (Figure 3.13, left). As the network contains more genes than Wilm's tumour samples, the computed covariance matrix is singular. Accordingly, the precision matrix does not possess a full rank either. This gives rise to a clearly visible block structure in the precision matrix due to a linear dependence between the columns and rows. By adding a small constant value $\lambda = 0.01$ to the diagonal of the covariance matrix, this block structure is greatly reduced (Figure 3.13, right). This demonstrates that proper regularisation is crucial to prevent artefacts that hamper the interpretability of the model. While using the ridge strategy is a clear improvement over naïve matrix inversion, the resulting precision matrix still is dense, meaning that every variable has a non-zero partial correlation with the other variables. This makes it difficult to visualise the model as a network structure. Applying the graphical lasso with a relatively strict regularisation factor of $\rho = 0.5$ (Figure 3.14) results in a much sparser precision matrix.

Of the detected 116 edges, 86 could be verified using a custom database collected from TRANSFAC [Mat+06], SignaLink [Faz+13], ChIP-

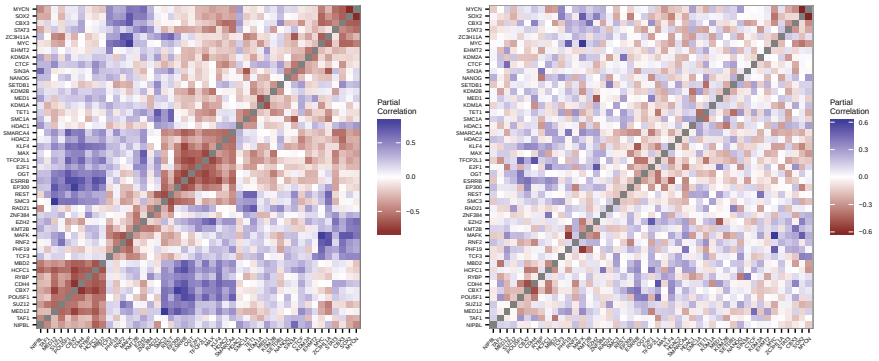


Figure 3.13: Partial correlations computed for the Wilm's tumour dataset using the naïve, matrix (pseudo)inversion based method (left). A clear block structure is visible due to the covariance matrix being singular. Adding a small, constant value $\lambda = 0.01$ to the diagonal of Σ greatly reduces this block structure (right).

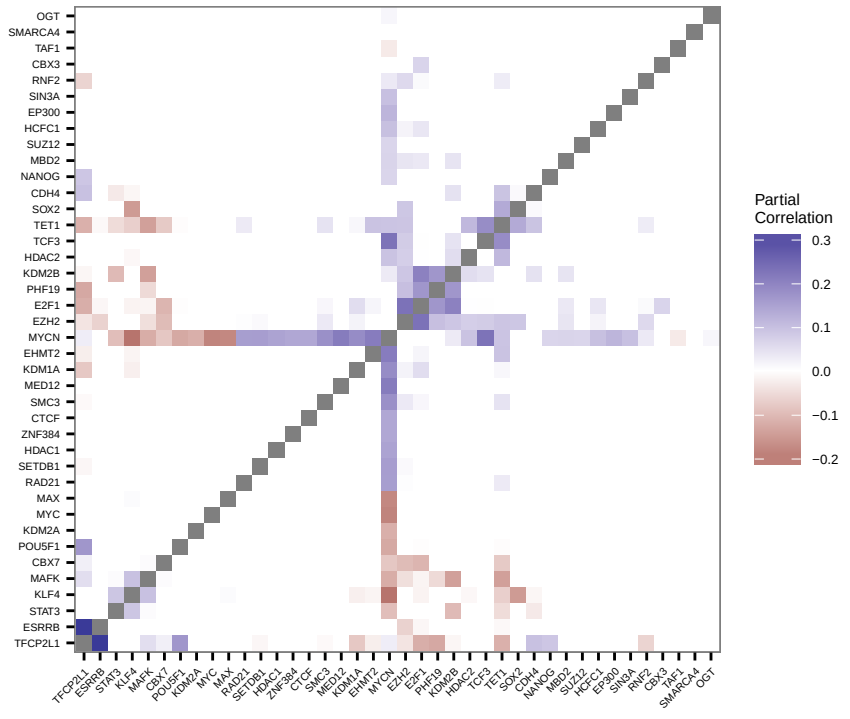


Figure 3.14: Partial correlations inferred from the Wilm's dataset using the `glasso` package. The penalty parameter ρ was set to 0.5. Red squares represent negative correlations while blue squares represent positive correlations. The values range between -0.2 and 0.3.

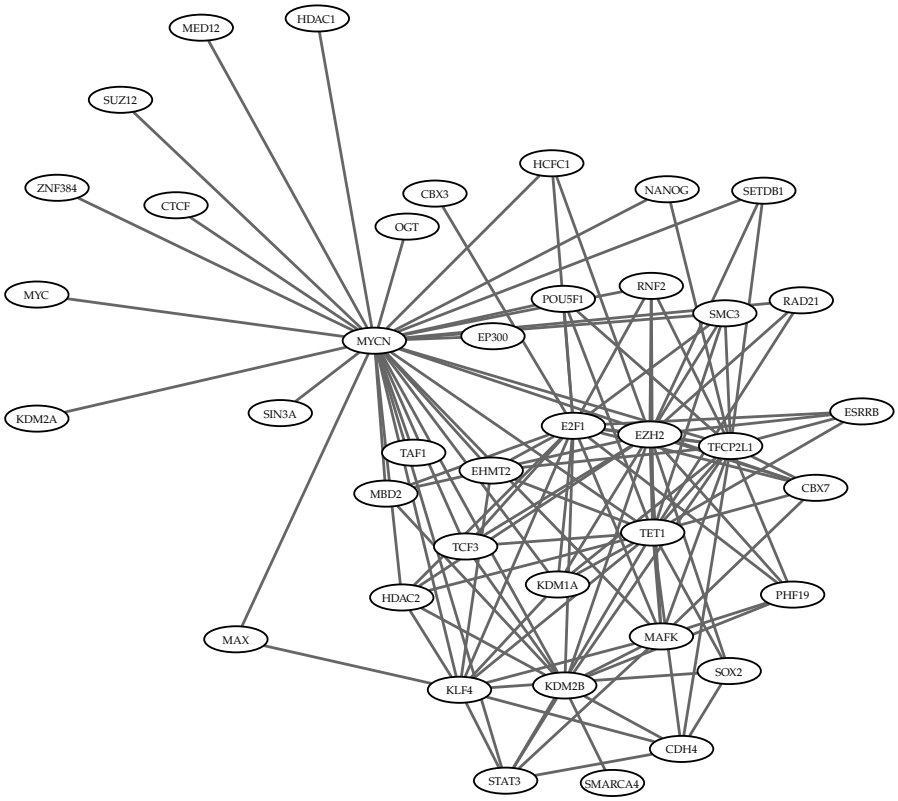


Figure 3.15: Gaussian graphical model inferred from the Wilm's dataset using the `glasso` package. Edges correspond to non-zero partial correlations. The penalty parameter $\rho = 0.5$ was used to ensure a network sparse enough for visualisation. Isolated nodes have been discarded from the network.

Atlas[8], ChipBase [Yan+13], ENCODE [ENC04], Jaspar [San+04], ChEA [Lac+10], as well as the publications by Cole et al. [Col+08] and Marson et al. [Mar+08b]. Thus, the graphical lasso procedure yields 30 false positive edges. To assess the performance of the method we randomly rewired our computed network ten times and calculated the number of false positives. We obtained a mean of 18.5 and a standard deviation of 1.78 false positives. This means that our graphical lasso procedure performs significantly worse than creating edges at random. Why is this the case? From the network structure (Figure 3.15), we can clearly see that MYCN plays a central role in the transcription factor network. This is surprising, as MYCN only has few verified targets, of which only POU5F1 is present in the network of Juan et al. [Jua+16]. In contrast, known master regulators [Col+08; Riz09; Gol+11] such as SOX2, POU5F1, CTCF, or TCF3, which possess a large amount of targets, are only connected to one or few other genes. A likely explanation may lie in the tendency of L_1 penalised methods such as the lasso to select exactly one out of a set of correlated variables [Tib+13; ZH05]. As the correlation between the genes in the network is high, the fact that

Edges by detected by our method that could not be found in a database might also be new discoveries. However, our randomisation experiment suggest, that most detected edges are likely not real.

8 http://chip-atlas.org/

MYCN is chosen as a hub may simply be an artefact of the method. Another explanation may lie in the way many of the selected genes fulfil their functions. Binding sites for transcription factors such as STAT3, TCF3, CTCF, ZNF384 can be found in a large number of genes. Due to this, the transcription factors are highly unspecific and rely on building complexes with more specific partners [Wol99; Rav+10]. However, GGMs only consider associations between pairs of genes and, thus, are unable to detect higher-order effects. In addition, the small amount of data used for training the network is likely to be responsible for the observed results. The inclusion of prior knowledge in the form of known TF binding sites could, in theory, help to improve the performance of the method. To this end, the `glasso` package offers facilities to adjust the penalty parameter ρ on a per-edge basis. Accordingly, edges that are present in databases could receive a lower penalty. Initial experiments with this approach have, however, not lead to substantial improvements. Finally, ρ represents a tuning parameter that needs to be adjusted to obtain the best possible results. For this, however, independent test and training networks would be required (cf. Appendix A.2).

We are nevertheless positive that GGMs can serve as a valuable tool given that the initial conditions are appropriate. This is shown by results such as the network by Juan et al. [Jua+16], which was determined using GGMs. However, in this case the authors could rely on data from ChIP-seq experiments from which they derived a wealth of co-occurrences of transcription factor binding sites and epigenetic marks. To obtain a similar dataset for gene expression experiments, several thousand samples would need to be produced. This is also the case for a recent breakthrough in protein folding that relies on the conditional mutual information, which is closely related to partial correlations, between residues to achieve tremendous improvements for the accuracy of *de novo* protein structure prediction [MHS12].

3.5 DEREGULATED SUBGRAPHS

The idea of searching deregulated groups of entities is also used by enrichment algorithms (see Chapter 4).

Until now, we have only considered methods, such as causal Bayesian networks and Gaussian graphical models that operate on comparatively small networks. For larger topologies, such as complete PPI or regulatory networks (Section 3.2), these methods are usually not applicable. Nevertheless, a common task during the analysis of e.g. expression datasets is to identify a portion or *subgraph* of, for instance, a regulatory network that contains a large amount of differentially expressed genes. The idea behind this is that (connected) subgraphs are likely part of the same biological process. Hence, identifying a subgraph with a high amount of differentially expressed genes hints at the fact that the corresponding process may be deregulated. This is of interest for analysing tumour samples where the activity of biological processes can be severely altered due to mutations and other pathogenic mechanisms.

To make this problem definition more precise, consider a graph $G(V, E)$ with node scores $w : V \to \mathbb{R}$. G represents our regulatory

network and w are scores that measure the differential expression of each gene or protein in G. Assume that a procedure that searches for "interesting" subgraphs inspects all members of a family of admissible subgraphs $H = \{H_i \subset G \,|\, i = 1, \ldots, n\}$. For each H_i, the procedure computes a score $s_i = S(H_i, w)$ that represents the degree of deregulation of a graph. Each H_i that is assigned a score surpassing a predefined threshold is called a *deregulated subgraph*. The deregulated subgraph with the highest score is called the *most deregulated subgraph*.

In this chapter, we discuss an algorithm for searching the most deregulated subgraph in regulatory networks. The central idea behind its development was the ability to detect molecular key players that are able to trigger regulatory cascades and are thus likely to be causal for the deregulation observed for the identified subgraph. To achieve this, we developed an algorithm based on *integer linear programming (ILP)* that searches for a connected subgraph of fixed size that contains a *root node*. In this context, a root node is a node from which all other vertices in the subgraph are reachable.

After introducing the ILP, we turn our attention to an open problem concerning algorithms for the search of deregulated subgraphs in general. As described above, a subgraph is considered as deregulated, whenever its score exceeds a predefined threshold. Yet, this definition is somewhat arbitrary and is not connected to any measure of statistical significance. It is thus not possible to assess whether the selected subgraph refers to a significantly deregulated biological process or not. To remedy this situation, an appropriate H_0 hypothesis needs to be chosen first. However, already this step proves to be difficult, as various, equally reasonable but not equivalent choices are possible. Here, we will not attempt to find an answer to this problem. Instead, we investigate a closely related, even more fundamental problem. In particular, we present an approach for estimating the likelihood with which a node is selected as part of the most deregulated subgraph, given uninformative scores.

3.5.1 *Related Work*

In the literature sometimes the term dysregulated subgraph is used.

Detecting deregulated subgraphs is a large field in computational biology that has brought forth a wide range of conceptually different approaches. The first published algorithm for searching deregulated subgraphs is the approach by Ideker et al. [Ide+02] which uses a simulated annealing scheme to iteratively improve the score of the detected subnetworks. As scoring function they generate a background distribution of subgraph weights. Each subgraph score is normalised with the number of vertices. The background generation and the simulated annealing procedure lead to considerable runtime requirements for larger networks. In addition, the size of the detected subgraph is often too large to be interpretable. To prevent this behaviour, various improvements to the scoring function and search heuristics have been published (e.g. Rajagopalan and Agarwal [RA05]). Ulitsky et al. [Uli+10] reduce the problem of finding a deregulated subnetwork to the NP-hard connec-

ted set cover problem. They present a range of heuristics with provable performance guarantees for solving their problem formulation.

The first formulation allowing for exact solutions on undirected graphs was stated by Dittrich et al. [Dit+08]. Similarly to Ulitsky et al. [Uli+10], they reduce the task to a well known problem from computer science: the prize-collecting Steiner Tree problem. To solve this, they apply a previously published ILP solution [Lju+06]. In contrast to the ILP presented in this work, the Steiner Tree based formulation uses both, edge and node weights. Recently, Bomersbach, Chiarandini, and Vandin [BCV16] formulated an ILP approach for solving the connected maximum coverage problem for detecting subnetworks that contain frequently mutated genes in cancer patients. Alcaraz et al. [Alc+11] use ant colony optimisation for extracting deregulated pathways. Later versions of their tool, which added support for the inclusion of prior knowledge as well as robustness analyses [Alc+14], are also available as a web service [Lis+16]. Maxwell, Chance, and Koyutürk [MCK14] describe an algorithm that allows to efficiently enumerate all induced connected subgraphs in a biological network that fulfil a predefined "hereditary property". Further algorithms can be found in a comprehensive review by Al-Harazi et al. [AlH+15].

3.5.2 *Subgraph ILP*

CONTRIBUTIONS The ILP approach for the detection of deregulated networks was implemented by Alexander Rurainski with later, maintenance related contributions by me. Christina Backes wrote the initial draft of the paper, which was completed by Oliver Müller, me, and Hans-Peter Lenhof.

Example for results obtained using our ILP are given in Section 5.4.5 and Section 5.5.6.

Here, we describe our ILP based algorithm for detecting deregulated subgraphs in directed graphs such as regulatory networks [Bac+12]. As mentioned before, a key characteristics of our algorithm is that it allows for the detection of molecular key players that sit at the top of a regulatory cascade. To this end, we require that the subnetwork is *rooted*. This means that there exists at least one node from which all other nodes in the selected subnetwork are reachable. The rational behind this requirement is that a change in a single regulator such as a mutation or epigenetic aberration can influence the expression level of a large number of downstream elements. Examples are driver mutations in cancer (cf. Section 2.1) that transform healthy cells into tumour cells by disrupting key regulatory processes, such as "cell cycle control" (cf. Section 2.1). By requiring the presence of a root, we thus aim to detect possible driver genes, which are causal for the deregulation of the subnetwork. For undirected graphs, in which every node can serve as the root node, the problem solved by our algorithm is equivalent to the *maximum-weight connected subgraph (MWCS)* problem [Dit+08].

Here, "causal" is not meant in the strict sense as defined in Section 3.3.

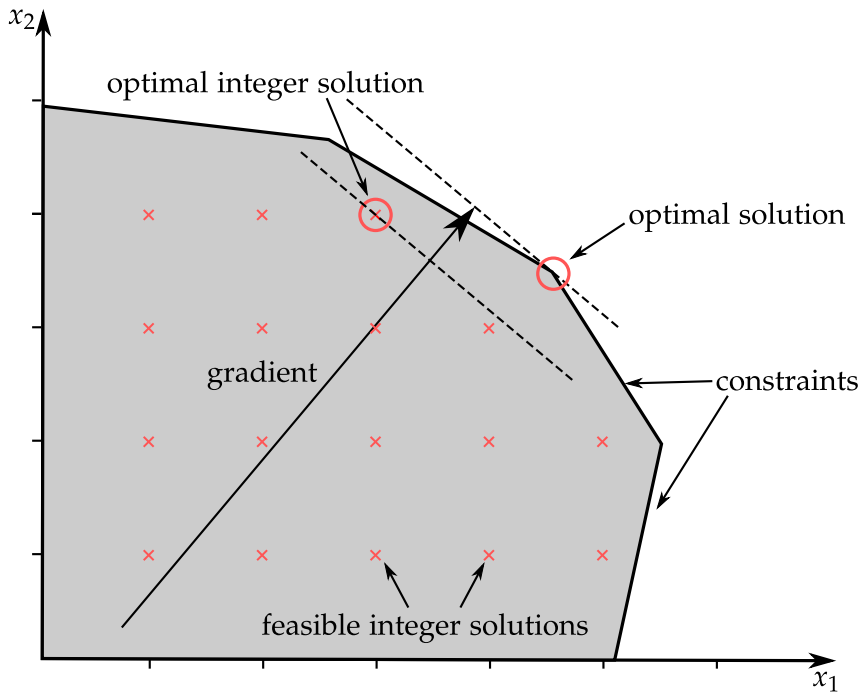


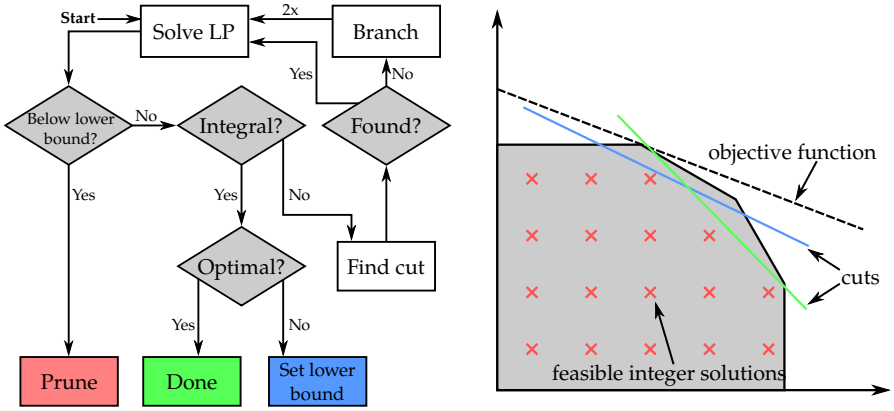
Figure 3.16: Two dimensional LP. The constraints of the LP define the polytope of feasible solutions. The optimal solution (red circle) of a LP can be found in one of the vertices of the polytope. Feasible integer solutions (red crosses) do not generally coincide with vertices. As a consequence, the optimal LP and ILP solution are commonly neither identical nor in direct proximity.

Additionally, our formulation requires that each subnetwork consists of exactly k nodes. This restriction was added to guarantee that the size of the computed subnetworks remains interpretable. While this may seem limiting, it is possible to perform a sweep across a range of subgraph sizes. This provides additional advantages. For example, the stability of the computed solutions can be assessed: if the subgraph topology changes dramatically between different values of k there likely exist multiple, equally relevant parts of the network that should not be interpreted in isolation. Before we are able to state the ILP formulation, we first need to introduce some basic concepts about linear programs.

Linear Programs

A *linear program (LP)* is an optimisation problem that can be stated in terms of a linear objective function subject to a set of linear constraints. Let $\vec{x} \in \mathbb{R}^n$ be a vector of variables, $\vec{c} \in \mathbb{R}^n$ a vector representing the coefficients of the objective function, $\boldsymbol{A} \in \mathbb{R}^{p \times n}$ a matrix representing the constraints, and $\vec{b} \in \mathbb{R}^p$ a vector containing the right-hand sides of the constraints. We can write every linear program in the following,

A LP does not necessarily posses an optimal solution or a solution at all for that matter.



(a) The branch-and-cut algorithm. (b) Effect of cutting planes.

Figure 3.17: Overview of a branch-and-cut algorithm. (a) First, the integrality constraints are relaxed and the LP is solved. If the found solution is worse than a previous, integral solution, the instance is discarded. If the solution is integral and no other instances need to be solved it is returned as the optimal solution. Otherwise, (b) a cut that removes no integer solution is searched and added to the LP. If no effective cut could be found, the algorithm branches on a variable and solves the two resulting instances.

canonical form:

$$\begin{aligned} \max_{x \in \mathbb{R}^n} \quad & c^t x \\ \text{s.t.} \quad & \boldsymbol{A}\vec{x} \leq \vec{b} \\ & x_i \geq 0 \quad \forall_i \end{aligned}$$

Every constraint defines a closed half-space in which a solution must lie. The intersection of these half-spaces results in a convex polytope containing all feasible solutions (Figure 3.16).

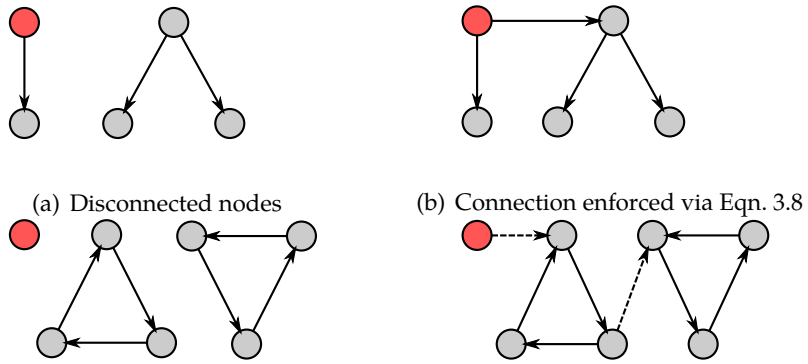
The ellipsoid method is also an interior point method, however the required computations are too time consuming to be practically applicable.

Linear programs, can be solved in polynomial time, even in the presence of an exponential amount of constraints using the ellipsoid method [Kha80; Kar84]. In real-world applications, the solution can be found using other interior point [LMS91] or simplex [D+55] algorithms. The latter traces the edges of the polytope in the direction of the objective function's gradient. This procedure is guaranteed to reach the optimal solution, as it can be easily shown that the optimal solution of a LP can be found in one of the vertices of the polytope.

An ILP for e.g. vertex cover provides a reduction that proofs NP-hardness.

For solving ILPs the situation is different, though. Integer linear programs (ILPs) are LPs in which the variables are restricted to integer values. Due to this additional constraint, vertices seldom coincide with feasible ILP solutions and hence the optimal solution cannot be found using algorithms for solving LPs. In fact, it can be easily shown that solving an ILP is a NP-hard.

However, due to the ubiquity of ILP formulations in both science and economy, efficient, exact solvers have been developed. These state-of-the-art solvers typically employ *branch-and-cut* strategies. In prin-



(a) Disconnected nodes (b) Connection enforced via Eqn. 3.8

(c) Disconnected cycles (d) Connection enforced via Eqn. 3.9.

Figure 3.18: Effect of the *in-edge constraint* Equation (3.8) and the *cycle constraint* Equation (3.9). Without the in-edge constraint the resulting subgraph does not need to be connected (a). With the constraint a connection to the root is enforced (b). However, disconnected cycles are still legal (c). With the cycle constraint, each cycle either needs to contain the root or has an in-edge from a chosen node outside of the cycle. The root node is indicated by a red background.

ciple these approaches work as follows (Figure 3.17): in a first step, the algorithm solves a so called *relaxed* version of the ILP, in which the integrality constraints have been dropped. The objective value of this solution serves as an upper bound on the optimal integer solution. If the solution is not integer, a new constraint is derived that cuts away the vertex containing the found solution, thereby tightening the polytope around the contained integral points. If no effective cut could be found, the algorithm branches on a variable, meaning that the ILP is divided into two instances. One, where the variable has a value lower than a computed bound and one where its value is higher. For binary variables this effectively means that two instances are created where the value of the branching variable is fixed to zero and one, respectively. Both instances are then solved separately. The objective value of the best current integer solution is kept as a lower bound. This allows to prune instances from the search tree whenever its relaxed LP solution has a lower objective value than the current lower bound. The algorithm iterates until an integer solution can be proven to be optimal. This is the case when e.g. no other branches remain to be solved. For more details on (integer) linear programming, we refer the reader to the comprehensive introduction by Bertsimas and Tsitsiklis [BT97].

ILP Formulation

The network is allowed to contain more than one node that can act as root. The constraints ensure that one of these nodes is picked as designated root node.

Our ILP formulation is based around two types of variables. Let $n \in \mathbb{N}$ be the number of nodes in the network. Each node i is associated with a binary variable $x_i \in \mathbb{B}$ that indicates whether it is part of the deregulated subgraph ($x_i = 1$) or not ($x_i = 0$). We group these variables into the vector $\vec{x} \in \mathbb{B}^n$. Similarly, each node is associated with a binary variable $y_i \in \mathbb{B}$ that is equal to one if and only if the node was selec-

ted as the designated root node. As with $\vec{x}$, $\vec{y} \in \mathbb{B}^n$ is the vector of the y_i. As we are searching for a maximally weighted subgraph, we aim to maximise the sum of node weights $w_i \in \mathbb{R}$ yielding the objective function $\vec{w}^t\vec{x} = \sum_{i=1}^{n} w_i x_i$ (Equation (3.4)). Common sources for node weights are scores for differential gene expression or differential protein abundances. Variables for the root nodes are not included in the objective function as their only purpose is to ensure that each subnetwork contains exactly one designated root (Equation (3.5)). Let us now state the complete ILP formulation:

$$\max_{\vec{x},\vec{y}\in\mathbb{B}^n} \quad \vec{w}^t\vec{x} \tag{3.4}$$

$$\text{s.t.} \quad \sum_{i=1}^{n} y_i = 1 \tag{3.5}$$

$$\sum_{i=1}^{n} x_i = k \tag{3.6}$$

$$y_i \leq x_i \qquad \forall i \tag{3.7}$$

$$x_i - y_i - \sum_{j\in \mathrm{In}(i)} x_j \leq 0 \qquad \forall i \tag{3.8}$$

$$\sum_{i\in C}(x_i - y_i) - \sum_{j\in \mathrm{In}(C)} x_j \leq |C| - 1 \qquad \forall C \tag{3.9}$$

To understand this, we first introduce additional notation. C denotes a subset of nodes that make up a cycle. The function $\mathrm{In}(i)$ returns the set of nodes j for which an edge (j, i) exists. $\mathrm{In}(C)$ is a shorthand notation for $\bigcup_{i\in C} \mathrm{In}(i)$. The constraint in Equation (3.6) ensures that exactly k nodes are selected for the final solution. Equation (3.7) assures that a node can only be used as a root node, if it is also chosen as part of the subgraph. So far, the constraints do not guarantee that the solution is reachable via the root node or that the graph is connect in any way (Figure 3.18 (a)). Equation (3.8) enforces that every node which is part of the solution is either a root or has a predecessor that is selected, too (Figure 3.18 (b)). Unfortunately, this constraint can be satisfied by a set of disconnected circles, as every node in the circle has a predecessor (Figure 3.18 (c)). Thus, a constraint needs to be added for *every* (directed) circle in the graph (Figure 3.18 (d)). To prevent this, Equation (3.9) requires that either the root node is in the circle C or some node in the circle has a predecessor outside of the circle that is also part of the solution. Unfortunately, the number of circles in a graph scales roughly exponentially with the average node degree. This means that enumerating all these constraints is not possible due to runtime and memory limitations. We will explain how to circumvent these problems in practice below.

Implementation

We implemented the ILP formulation using the branch-and-cut [PR91] framework provided by the CPLEX solver [Bac+12; IBM12]. In particular, we made use of CPLEX's capabilities to lazily add constraints to

the problem. This is important as an exponential number of cycle constraints can be required. However, only a fraction of these constraints are likely to be ever evaluated during solving as we can expect that the most deregulated subgraph will mainly centre around the highest scoring nodes. Thus, it is usually more efficient to first solve the problem without any cycle constraints at all. After a solution has been obtained, it can be checked for possibly violated constraints in a callback routine. If any such constraints can be identified, they are added to the problem instance. This procedure is iterated until no further violated constraints can be detected.

Similarly, we use CPLEX's functionality to turn a LP solution into an ILP solution using a heuristics. To this end, we use a simple, multi-greedy approach. Given a non-integer solution we select all nodes for which x_i is larger than zero. This induces a subgraph consisting of one or more connected components. Each of these components is then expanded or shrunk until it contains k nodes and then returned as a potential solution.

A compiled version of our implementation can be downloaded under `http://genetrail.bioinf.uni-sb.de/ilp/Home.html`.

3.5.3 *Topology Bias*

CONTRIBUTIONS This investigation was adapted from Miriam Bah's Bachelor's thesis [Bah12] supervised by Marc Hellmuth and me.

We previously explained that assessing the significance of a deregulated subgraph is an open problem (Section 3.5). A first issue lies in the fact that it is unclear on the basis of which H_0 hypothesis a p-value should be computed (cf. Section 4.1). At least three different formulations with slightly different interpretations come to mind:

1. Is the score obtained for the selected subgraph significant?
2. Is the obtained score significant?
3. Is selecting the current subgraph with the associated score significant?

While the first and the second question can probably be answered by sampling a substantial amount of score permutations, the third question is more difficult to answer, as it additionally requires enumerating all possible subgraphs.

Here, we do not further pursue the computation of a p-value. Instead, we concern ourselves with a different, but closely related problem: was a member of the computed subgraph chosen "by chance" or because it contributed valuable information? As an example, consider Figure 3.19. There, connected subgraphs of size $k = 3$ were selected from a path of length six. To simplify reasoning about the involved

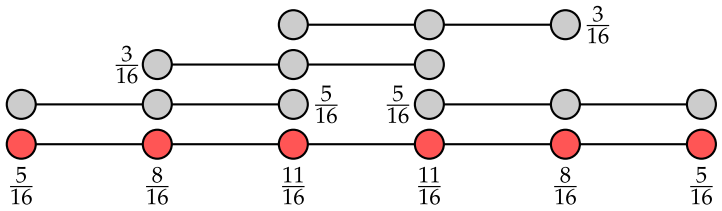


Figure 3.19: Subgraphs of size $k = 3$ in a path of length six. The number below each node indicates the probability with which it is part of the selected subgraph. Numbers next to subgraphs indicate the probability with which the subgraph is chosen. Probabilities were determined by evaluating all permutations of the scores $\{2^i \mid i = 0, \ldots, 5\}$. The two inner subgraphs are less likely to be chosen than the outermost subgraphs. However, the innermost nodes are more than twice as likely to be selected than the outermost nodes.

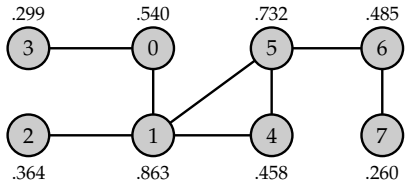


Figure 3.20: Bias of node selection probabilities (numbers next to nodes) in a more complex topology. Subgraphs of size $k = 4$ were selected using all permutations of the scores $\{2^i \mid i = 0, \ldots, 7\}$. Node 7 is more than three times less likely to be chosen than node 1.

probabilities, we generated nodes scores w_i as powers of two, such that each score is larger than the sum of all smaller scores. Thus for every node i, we have an associated score $w_i = 2^i$. This guarantees that the node with the highest node score is part of the most deregulated subgraph. Next, we enumerated all possible permutations of the scores and computed the probability with which each node and subgraph is selected given a random score permutation. It can be seen that the probabilities for neither subgraphs nor nodes are uniformly distributed. Instead a significant bias towards some subgraphs is present. The probability of selecting a node is given as the sum of the subgraph probabilities and hence, is similarly biased. It is important to note that this bias is purely due to the chosen topology as we enumerated all score permutations. With a slightly more complex topology, such as in Figure 3.20, this effect becomes more pronounced, putting more and more probability mass on hub-like nodes such as node 1 and 5.

Naturally, finding deregulated subgraphs relies on the topology of the input network. However, the above observation suggests that the topology has a significant impact on the obtained solutions. To be useful for analysing e.g. expression data, subgraph methods need to maintain a balance between the constraints imposed via the topology and the information provided by the input scores. If, on the one hand, the solution depends to a large degree on the topology with only minor influence from the input data, the method is not able to uncover interesting biological phenomena. On the other hand, if the topology has no influence on the result, a simpler, topology free method could be employed

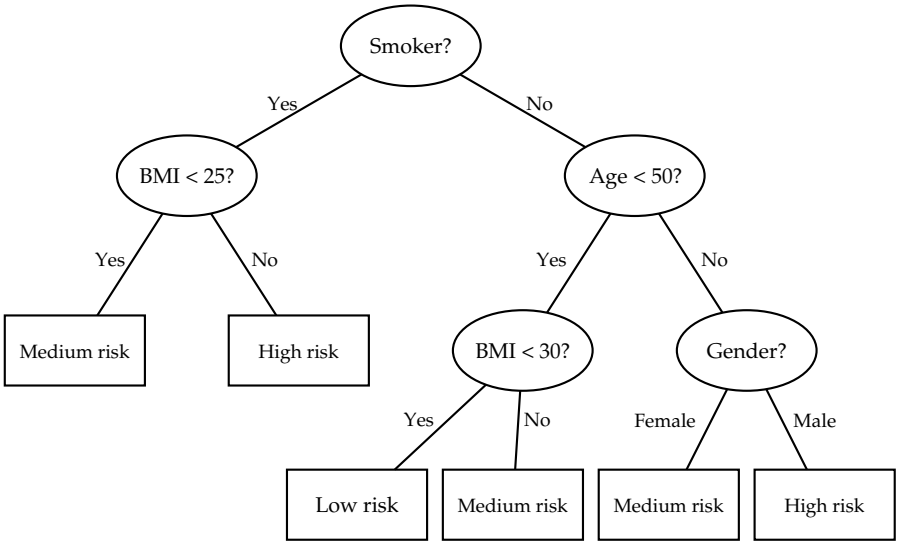


Figure 3.21: Example for a (fictive) decision tree stratifying patients into low, medium, and high risk groups for congestive heart disease. In each decision node (round), a predictor (Smoker, Age, BMI, Gender) is used to partition the training data. Predictions for new datapoints can be made by following a path from the root to a leaf node (rectangles), which contain the predicted value.

instead. Quantifying the influence of topology and input data on the results of a deregulated subnetwork algorithm is difficult, though.

Here, we propose a procedure that allows to judge the bias incurred by methods for the detection of deregulated subgraphs towards certain nodes due to the network topology. It is based on predicting the probability that a given node is selected as part of the subnetwork. To this end, we train a *random forest* model [Bre01] for predicting node inclusion probabilities that were empirically determined by sampling deregulated subnetworks given uniformly distributed input scores. This model uses a set of node topology descriptors as features. Applying the trained predictor to an unknown network allows to estimate the topology bias without a computationally intensive sampling step.

For a brief introduction to machine learning refer to Appendix A.2.

In the following we explain our procedure in detail. To be able to do so, we first need to introduce the random forest predictor as well as the used topology descriptors. Next, the used training and validation scheme is outlined. Finally, we present the results of our study.

Random Forests

Random forests are non-linear models for classification and regression (cf. Appendix A.2). In particular, they are an ensemble method that is based on averaging the output of $B \in \mathbb{N}$ decision trees $\hat{f}_b$ that were trained on bootstrap samples of the data (*bagging*). A decision tree is a machine learning method that partitions its input space via recursive, binary cuts [Qui86; Qui93]. These cuts can be represented as nodes in a tree (cf. Figure 3.21). Which split is used in each node is determined by evaluating a loss function for each dimension. Examples are the squared-error loss for regression or the Gini index for classifica-

For more details on random forests we refer to the original paper by Breiman [Bre01] and the introduction by Friedman, Hastie, and Tibshirani [FHT09].

tion. The variable that achieves the smallest loss is chosen for cutting. In contrast to common decision trees, which always evaluate all variables to decide on a split, the random forest procedure randomly selects k predictors from which the best candidate must be chosen. Furthermore, each tree is built on a bootstrap sample of the training data. A bootstrap sample for a dataset containing N samples is generated by randomly drawing N samples from the dataset *with* replacement. Each tree is grown until it perfectly predicts its training data. In a regression scenario, the final random forest model is defined as the average prediction of all trained decision trees:

$$\hat{f}(x) = \sum_{b=1}^{B} \hat{f}_b(x)$$

In a classification scenario, the output class label is determined via a majority vote.

Often $B \approx 500$ and $k = \sqrt{p}$ for classification and $k = \lfloor p/3 \rfloor$ for regression, with p being the number of predictors, is used.

Random forests have various advantages that make them especially suited for quickly obtaining classifiers that offer a performance that is competitive with other state-of-the-art approaches. A reason for this is that random forests models are comparatively straightforward to set up. As inputs they can use continuous, ordinal, and categorical inputs and therefore require no special feature encoding techniques. In addition, the model exposes only two tuning parameters: the number of trees B and the number of features considered in each split k. For both parameters good default values are known. Choosing different, but reasonable settings commonly has no critical impact on performance. Due to this, the model is robust with respect to overtraining [Bre01; FHT09].

As each tree is grown on a different bootstrap sample, not every sample has been used for every tree. Thus, for each tree $\hat{f}_b$ the prediction error for the samples that have not been used to train $\hat{f}_b$ can be determined. This yields the so-called *out-of-bag (OOB)* error that can serve as an estimate of the test error. Thus cross-validation or a separate tuning set are not strictly required for tuning random forests.

Finally, it is possible to compute feature importances. The feature importance reflects how large the impact of a feature on the predicted value is. There are two ways to compute these importances. The first version is based on the OOB error. For each tree, the OOB error is computed. Then, to obtain the importance of variable j its values are permuted and the OOB error is computed again. The difference between the two values is averaged across all trees and returned as the variable importance [Bre01]. Alternatively, the decrease of the loss function in each split gives an indication of the influence of the chosen split variable. These values are accumulated across all trees and splits to obtain the final importance [FHT09].

Of course, random forests also have disadvantages. Probably the most important one is that the final model is difficult to interpret. While we can extract the importance of single variables, random forests are highly non-linear and thus the importance measure does not permit a straightforward interpretation such as: "If the value of variable A in-

creases by 10 % the risk for cancer increases by 20 %." In contrast, the weights computed by linear models (Section 3.4.1) or SVMs with linear kernels [FHT09; SS02] have this property. Also, for SVMs with a well-performing, non-linear kernel, examining the feature maps allows to gain mechanistic insight into the learning problem. Furthermore, in the presence of categorical variables, random forests tend to favour those variables with more levels. Consequently, the computed variable importance measures are not reliable in this case and need to be adjusted using sampling techniques [Alt+10].

Topology Descriptors

The terms feature and predictor are often used synonymously.

Arguably the most important prerequisites for applying machine learning algorithms is the selection of a suitable set of features and their representation. For our task, features are needed that capture the topological properties of a graph node. To this end we use a set of topology descriptors which have originally been developed for analysing the importance of nodes in large network structures such as the topology of the internet. In this section we describe and define the used descriptors.

First and foremost, we use simple, graph theoretical properties such as the in- and out-degree of a node. These statistics, however, characterise the node almost in isolation and pay little regard to the surrounding topology. To help assessing the importance of a node on a global scale, more sophisticated measures such as the PageRank [Pag+99] or various *centrality* measures have become popular. Let us, however, start with more basic measures.

A natural choice for topology descriptors are the node degrees as introduced in Definition 5. The degree of a node, especially relative to the degrees of the remaining nodes reflects the number of connections a node has and should thus be a crude approximation for how "hub-like" the node is. Nodes with a high in-degree are also more likely to be leafs in a selected subgraph, whereas nodes with a high out-degree should be more likely to act as root nodes. The clustering coefficient is a generalisation of this idea and also considers how the neighbours of the node are connected. More formally:

Definition 11 (Clustering coefficient)**.** A triangle is a triple of connected nodes. Let $\tau_\Delta(v)$ denote the number of triangles containing $v \in V$ and let $k_v \in \mathbb{N}$ be the number of adjacent nodes. Then the clustering coefficient for v is defined as

$$\mathrm{cl}(v) = \frac{\tau_\Delta(v)}{k_v(k_v - 1)}$$

In contrast to the node degrees, centrality measures put a node in a more global context. For example the *betweenness centrality* [Fre77] counts the number of shortest paths that include a given node.

Definition 12 (Betweenness centrality)**.** Let $\sigma(s, t \mid v)$ denote the number of shortest paths between nodes s and t that include v. Furthermore, let $\sigma(s, t)$ be the total number of shortest paths between the two nodes.

Then the *betweenness centrality* is given by

$$c_B(v) = \frac{1}{(|V|-1)(|V|-2)} \sum_{s \neq t \neq v} \frac{\sigma(s,t|v)}{\sigma(s,t)}$$

The closeness centrality [Bav50; Bea65] is also based on the notion of shortest paths. However, it considers the reciprocal of the average length of all shortest paths starting from the node of interest v. If all remaining nodes in the graph can be reached quickly, v receives a high centrality value. Otherwise, if the length of the shortest paths is long, v has a low centrality.

Definition 13 (Closeness centrality)**.** For a node $v \in V$ the closeness centrality is given by

$$c_C(v) = \frac{n-1}{\sum_{u \in X_v} \text{dist}(v,u)}$$

Here $\text{dist}(v,u)$ denotes the length of the shortest path from v to u and $X_v := \{u \,|\, \text{dist}(v,u) < \infty\}$.

Whereas betweenness and closeness centrality use shortest paths to determine the importance of a node, Bonacich [Bon72] proposed to use the dominant eigenvalue of the adjacency matrix as centrality measure.

Definition 14 (Eigenvector centrality)**.** Let $\boldsymbol{A}$ be the adjacency matrix of G with $A_{vu} = 1$ if $(v,u) \in E$ and 0 otherwise. Then the out-eigenvector centrality $c_{\text{out}}(v)$ is defined as the v-th entry of the principal eigenvector of $\boldsymbol{A}$. The in-eigenvector centrality $c_{\text{in}}(v)$ is the v-th component of the principal eigenvector of $\boldsymbol{A}^t$.

The principal eigenvector of a matrix can be easily computed using the power iteration [MP29]

$$\vec{v}_{t+1} = \frac{\boldsymbol{A}\vec{v}_t}{\|\boldsymbol{A}\vec{v}_t\|}.$$

Interpreting the eigenvector centrality, though, is not straightforward as no inherent meaning is attached to an entry of the principal eigenvector. Probably the best intuition is that a node is assumed to be more central, if it is connected to other central nodes [Ruh00].

Effectively, the PageRank interprets the adjacency matrix as a Markov process.

A measure that is closely related to the eigenvector centrality is the PageRank. It models the probability with which a user clicking random links (edges) on a web page (nodes) ends up on a certain site. To avoid local attractors such as nodes with no out-edges, the PageRank includes a constant, uniformly distributed restart probability into its model:

Definition 15 (PageRank)**.** Let $\text{In}(v) = \{u \,|\, (u,v) \in E\}$ be the in-neighbourhood of v. The PageRank for v and a damping factor $\alpha \in \mathbb{R}^+$ is defined as

$$\text{PR}(v) = \frac{1-\alpha}{|V|} + \alpha \sum_{u \in \text{In}(v)} \frac{PR(u)}{d^+(u)}$$

The damping factor α controls the influence of the random restart. The recursive formulation of the PageRank can be rewritten in terms of a matrix power, which allows to obtain the limit distribution via an eigenvalue decomposition [Pag+99].

The above graph measures are well known and implemented in popular software packages such as Cytoscape [Sha+03; Ass+08]. As all of them are general purpose measures, none directly estimates the number of subgraphs of size k in which a node is contained in. Hence, to supplement the existing measures, we devised our own descriptor. It is based on the number of spanning trees that use the outgoing edges of a node. The idea behind using spanning trees is that every rooted, deregulated subgraph can be trivially transformed into a spanning tree of the selected nodes (cf. [Dit+08]). Thus, knowing the number of spanning trees that are present in a neighbourhood of a node, should allow a good estimate of the number of deregulated subgraphs the node is a part of.

The measure relies on Kirchhoff's theorem [Kir47] which states that the number of spanning trees rooted in a vertex can be obtained from the cofactors of the Kirchhoff matrix. Let us start with the definition of a cofactor.

Definition 16 (Cofactor)**.** Given a matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$. The ij-th cofactor of $\boldsymbol{A}$ is given by

$$C_{ij}(\boldsymbol{A}) := (-1)^{i+j} \det(\boldsymbol{M_{ij}})$$

where $\boldsymbol{M_{ij}} \in \mathbb{R}^{(n-1)\times(m-1)}$ is the matrix obtained by deleting the i-th row and j-th column from $\boldsymbol{A}$.

The Kirchhoff matrix, also known as graph Laplacian, is based on the adjacency matrix. It can be interpreted as a discrete analogue of the Laplacian operator from calculus or the Laplace-Beltrami operator from vector analysis.

Definition 17 (Directed Kirchhoff Matrix)**.** Given the directed graph $G(V, E)$. Its Kirchhoff matrix $\boldsymbol{K}$ is defined as:

$$K_{ij} = \begin{cases} d_i^- & \text{if } i = j \\ -1 & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$$

By construction, $\boldsymbol{K}^t$ has an eigenvalue of zero corresponding to the eigenvector $\not\Vdash^n$, the vector where each component has the value 1. If G has m connected components, the kernel of $\boldsymbol{K}$ has dimension $m + 1$ as for every connected component we can create an eigenvector for the eigenvalue zero by simply setting all coefficients that belong to nodes inside the connected component to one and all others to zero.

We will now describe the relation between the Kirchhoff matrix and the number of spanning trees. For this we first need to define the notion of an out-tree.

Definition 18 (Out-tree)**.** An out-tree is a connected, directed acyclic graph which is rooted in a single vertex. The root vertex has in-degree zero and every other vertex has in-degree one.

Given the definitions above we are able to state the following theorem by Tutte [Tut48] that links the number of spanning trees rooted in a vertex with the cofactors of the Kirchhoff matrix. It is a generalisation of Kirchhoff's Matrix-Tree theorem [Kir47] for undirected graphs.

Theorem 3.5.1 (Directed Matrix-Tree Theorem)**.** Let $\boldsymbol{K}$ be the Kirchhoff matrix of $G(V, E)$. For any vertex $i \in V$, the number of out-trees rooted at i equals $C_{ij}(\boldsymbol{K})$ for an arbitrary choice of j.

We can use this theorem to compute the number of spanning trees of G. For this, we simply need to select a vertex i from which all other vertices are reachable. If no such vertex exists, G has no spanning tree. Otherwise we can select an arbitrary column j and compute the number of spanning trees as $C_{ij}(\boldsymbol{A})$. For our purposes, this trick is not directly useful, as the number of spanning trees is necessarily the same for all vertices that can serve as the root of a spanning tree. However, we can use this technique to compute the number of spanning trees that use a given edge e by computing the number of spanning trees in G minus the number of spanning trees that remain in the graph after e has been removed:

$$\#\text{spTree}(e) = \#\text{spTree}(G) - \#\text{spTree}(G - e)$$

This yields an importance measure for every edge: the more spanning trees use this edge, the more important it is. For computing vertex importances, we can simply accumulate the importances of all incident edges.

Definition 19 (Spanning tree weight)**.** Let $G(V, E)$ be a directed graph. Define $E_v \subseteq E$ be the set of incident in- and out-edges of the node $v \in V$. We define the spanning tree weight of v as

$$\text{sp}(v) = \sum_{e \in E_v} \#\text{spTree}(e)$$

For computing the degrees, betweenness and closeness centrality, and the clustering coefficient we used the NetworkAnalyzer [Ass+08] Cytoscape [Lop+10] plugin. The PageRank and spanning tree weights were computed using a C++ implementation based on the BGL [SLL01], Eigen 3[9], GMP[10], and MPFR[11] libraries. The eigenvalue centrality was computed using Python [VD09] as well as the NetworkX [SS08a] and SciPy [J+01] packages.

Predicting ILP Counts

We finally have all required pieces in place to predict the likelihood with which a node is selected as part of a deregulated subgraph. To this end, our procedure for obtaining the node selection probabilities

9 `http://eigen.tuxfamily.org`
10 `https://gmplib.org/`
11 `http://www.mpfr.org/index.html`

is structured in the following way: first, we generate uniformly distributed scores for each node in the network. Subsequently, the scores are permuted 20,000 times and for each permutation the corresponding subgraph of size $k = 10$ is computed. For each node we count the number of times it is selected as part of a network. When training our random forest predictor, these "ILP counts" are used as a surrogate for each node's probability to be selected by the algorithm.

To derive our feature set, we computed the topology descriptors presented above. For training the model, two thirds of the vertices were randomly assigned to the training set and one third to the test set. We used the random forest implementation provided in the `randomForest` [LW02] R [R C16] package with default parameters.

Results

The degree sequence is the sorted list of all node degrees.

Using the above procedure we trained predictors for the KEGG regulatory network, as available in the NetworkTrail [Stö+13] web server (cf. Section 5.4.4), and a random network with the same degree sequence. The latter was generated using the methodology described by Newman, Strogatz, and Watts [NSW01].

Figure 3.22 shows a scatter plot of the predicted vs. the empirical ILP counts for the KEGG network. Considering that the model was not tuned, the overall agreement is remarkable with a normalised *root-mean-square-error (RMSE)* of about 0.07. The normalised RMSE is computed by dividing the RMSE with the range of input ILP counts. This clearly proofs that it is possible to predict the likelihood with which a node is selected purely from topological features. For the human KEGG network, our results show that subgraphs obtained using the subgraph ILP are subject to a substantial bias due to the underlying topology.

For assessing the importance of the features, we examine both, the feature importance determined by the random forest model and Spearman's correlation coefficient of the feature with the ILP counts. Spearman's correlation [Spe04] is a non-linear version of Pearson's correlation coefficient [Pea95b]: given two lists X, Y of length k, each value x_i, y_i is assigned its position or *rank* $R(X_i), R(Y_i)$ in the sorted lists $\tilde{X}, \tilde{Y}$. Spearman's correlation is then simply the Pearson correlation of the ranks:

$$\rho_S(X,Y) = \frac{\sum_i (R(X_i) - \bar{R}(X)(R(Y_i) - \bar{R}(Y))}{\sqrt{\sum_i (R(X_i) - \bar{R}(X))^2 \sum_i (R(Y_i) - \bar{R}(Y))^2}}$$

Here, $\bar{R}(X)$ and $\bar{R}(Y)$ denote the mean ranks of X and Y, respectively. The obtained importance weights are given in Table 3.2. Based on this data, the spanning tree measure is clearly the most important predictor for the ILP count.

Similar results can be obtained when training the model on the mouse KEGG regulatory network (normalised RMSE $\approx$ 0.08) or our randomly generated graphs (normalised RMSE $\approx$ 0.06). In addition,

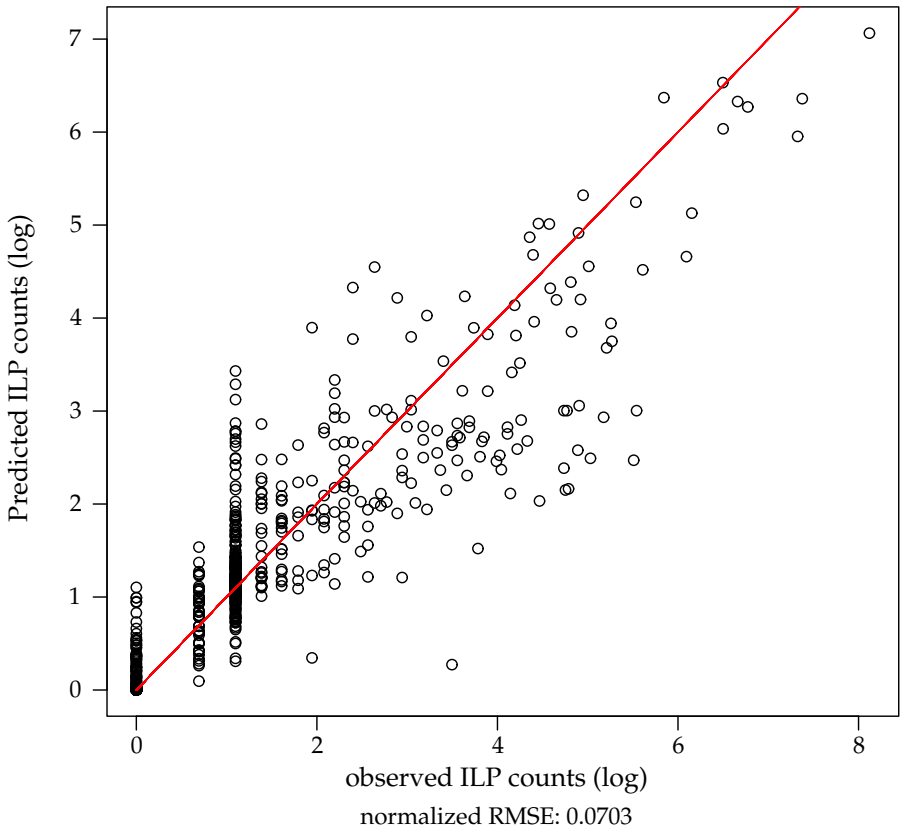


Figure 3.22: Log-log plot of the observed vs. predicted ILP counts for the human KEGG network test data. The counts were predicted by applying the random forest model to the test data.

Topology descriptor	ρ_S	Feature importance
Spanning tree weight	0.8799	1298.23
In-degree	0.4641	376.76
Betweenness centrality	0.4889	324.29
PageRank (d = 0.99)	0.5124	262.57
Clustering coefficient	0.3898	120.18
Out-degree	0.0961	102.25
Out-eigenvector centrality	0.3633	94.26
Closeness centrality	-0.1449	93.94
In-eigenvector centrality	0.7369	57.29

Table 3.2: Analysis of the performance of the individual topology descriptors. Both Spearman's correlation ρ_S and the feature importance as determined by the random forest model is given. The spanning tree measure dominates all other descriptors.

training the model on the mouse KEGG network and predicting the ILP counts for the human network results in a normalised RMSE of ≈ 0.09. This result suggests that our predictor is not overfitted for the network it was trained on. Instead, the model generalises to yet unseen data. This further confirms that the node selection bias can in fact be predicted from purely topological properties.

Discussion

We presented theoretical results that suggest that the probability with which a node is selected as part of the most deregulated subgraph highly depends on its topological properties. To confirm this, we measured the bias of the Subgraph ILP induced by the underlying network topology by using a sampling technique. To further confirm that this bias is purely based on a nodes topological properties, we trained a random forest model on a set of topology descriptors. With this model, we were able to obtain good predictors for the node inclusion frequency. In particular, our novel spanning tree measure prove to be an excellent predictor. This suggests that the most deregulated subgraphs obtained for scores from an experiment might be influenced considerably by the network topology. While this is desirable to a certain extent, the user should keep this bias in mind when interpreting a deregulated subgraph.

For counteracting the described effect, a node reweighing strategy may be employed. We conducted initial experiments in this regard, in which we down-weighed the scores s_i of a node i according to a function of its probability p_i to be selected. Examples for potential functions are s_i/p_i, $s_i/\sqrt{p_i}$, and s_i/p_i^2. However, our preliminary results have not shown a significant impact on the selection probability.

There are various ways to continue the presented study. First, confirming the reported results for a different, published algorithm would demonstrate that the described topology bias is in fact a universal phenomenon and is not linked to the particularities of the Subgraph ILP. A candidate algorithm would need to support computing a high amount of random samples to obtain reliable estimates for the node inclusion probabilities. Unfortunately, we were not yet able to find a second algorithm meeting this requirement. Additionally, the analysis should be rerun using networks from a different source. While our random networks share no common nodes with the KEGG networks, they are based on the same degree sequence and thus contain the same number of hubs and leaf nodes. Using a completely independent network could therefore result in different distributions of the inclusion probabilities. We are, however, confident that a similar bias is also detectable for different algorithms as well as different networks.

Finally, it should be noted that the reported biases were obtained for random score permutations and thus non-informative scores. In a real-world scenario the scores used for computing a deregulated subgraph are likely to contain a signal that manifests as e.g. the differential expression of specific gene sets. In this case, the score distribution

may, to a large degree, overrule the topology bias. For confirming this, a large scale analysis of diverse expression datasets could be used.

4 ENRICHMENT ALGORITHMS

The analytical power should not be confounded with simple ingenuity; for while the analyst is necessarily ingenious, the ingenious man is often remarkably incapable of analysis.

— EDGAR ALLEN POE, the murders in the rue morgue (1841)

Comparative microarray and RNA-seq studies (Section 2.3) are experimental techniques for detecting the differences in the transcriptional state between a sample and reference group. Both techniques typically result in large matrices of measurements where each row corresponds to all expression values of a single gene and each column contains all measurements of a single sample. Due to the high number of measured genes it is difficult to interpret this data manually. Instead, robust statistical methods for performing automated analyses are required that are capable of condensing the information contained in these high dimensional datasets into an interpretable report. Ideally, these reports would allow to quickly deduce the molecular key players and regulatory mechanisms that are responsible for the difference between sample and reference groups.

There are several ways to approach this task. For instance, a classifier that uses genes as features can be trained to discriminates between the two groups. Once training is complete, the parameters of the classifier can be examined to derive feature importance measures. The k most important features are then reported as the "key players" that are likely to be responsible for the observed differences. The drawback of this approach is that training a good classifier is by no means a trivial affair. Even then, the identified features are, while certainly discriminative, not necessarily causal for the observed effect. Thus, the features, which have not been considered by the classifier in lieu of "more informative" predictors, may still carry valuable information that is not captured using this approach. Alternatively, techniques that do not focus on classification performance, but rather on providing better understanding of the input data can be employed to circumvent these issues. Examples are techniques such as *principal component analysis (PCA)* [Pea01] (see Section 3.4) or *partial least squares (PLS)* (cf. [FHT09]) that decompose the data into orthogonal coordinates which either explain the observed variance (PCR) or the group differences (PLS). These coordinates can, again, serve as feature importances and can, hence, be used to extract the most relevant features.

In practice regularised versions of PCA or PLS would be necessary.

The above approaches do not employ any kind of prior knowledge. However, a large collection of biological databases such as the *Gene Ontology (GO)* [Ash+00] exist that contain carefully collected, functional annotations for genes, proteins, and other biological entities. Instead of identifying novel patterns in the data, we can use these annotations to check whether known categories of entities behave differently between

Algorithms for searching deregulated subgraphs (Section 3.5) are enrichment methods that use the admissible subgraphs as categories.

the sample and reference group. *Enrichment algorithms* or *enrichment methods* are a class of procedures that make use of this information. As input, an enrichment algorithm relies on one or more sets of entities, so-called *categories*. Each category commonly corresponds to an annotation such as a GO term. In addition, the algorithm requires a set of measurements such as a gene expression matrix. As output, it computes an *enrichment score*, which reflects the overall "degree of deregulation", and an associated significance measure for each input category.

Owing to this somewhat loose problem definition a multitude of enrichment procedures have been devised. In this chapter, we discuss a selection of popular enrichment methods. In particular, we examine their strengths and weaknesses as well as the considerations that are necessary when choosing between algorithms. First, however, as the computation of the significance of an enrichment score is a central part of all enrichment algorithms, we introduce the concepts of a *hypothesis tests* and p-values.

4.1 STATISTICAL TESTS

A central task in science is the verification or falsification of hypotheses. To this end, experiments must be conducted which then provide evidence in favour of or against a hypothesis. Evaluating this evidence carefully and objectively is thus a common responsibility in a research setting. For this purpose, statisticians have devised testing strategies that can be used to determine if a hypothesis remains valid or needs to be rejected in the light of new evidence.

Two frameworks for conducting such *hypothesis tests* exist. The first framework was devised by Fisher [Fis25] and is centred around computing a p-value that measures how strongly the data contradicts a *null hypothesis*. The framework by Neyman and Pearson [NP33], on the other hand, has the goal of deciding between a null and an alternative hypothesis. This is achieved by introducing a significance level α that controls the admissible rate of cases in which the null hypothesis is wrongly rejected. In today's scientific practice a hybrid between the two models is used that explicitly uses both a p-value and a significance level [Lew12]. The validity of this hybrid approach is heavily debated as it is, according to some statisticians, an "incoherent mishmash" [Gig93]. The central point of critique is that the current use of hypothesis tests neither allows the assessment of the strength of evidence nor allows to control the long-term error rates. Nevertheless, we will introduce concepts from both frameworks without explicitly differentiating between the two.

4.1.1 *Null and Alternative Hypothesis*

The central concept of all hypothesis tests is the so-called null hypothesis H_0. It represents a default, commonly accepted believe that is

assumed to be true unless evidence against it is established. If the H_0 hypothesis is rejected, a different explanation for an effect needs to be found. This alternative explanation is commonly formulated as the alternative hypothesis H_1. We will formulate hypotheses in terms of the parameters of the underlying distribution. For instance, when a decision should be made whether two groups stem from the same or two different generating normal distributions, we write

H_1 and the decision between H_0 and H_1 stems from the Neyman-Pearson framework.

$$H_0 : \mu_1 = \mu_2$$
$$H_1 : \mu_1 \neq \mu_2$$

This means that we assume the population means of both groups to be equal unless our data provides sufficient evidence against this hypothesis. In this case we assume H_1 to be true. In order to decide the strength of the evidence for rejecting H_0 we compute a test statistics T. A test statistics is a function transforming input data into a scalar value. As the input data can be modelled as a random variable, T itself is a random variable, too. To be able to better gauge the value of a test statistics, we can assign a so-called p-value to each outcome.

Definition 20 (p-value)**.** Let H_0 be the null hypothesis and T a test statistics. Let T_0 be the value of a test statistic obtained from a dataset. The p-value is defined as the probability to obtain a result from T that is at least as extreme as T_0 given that H_0 holds. More formally

Here we use T as a random variable.

$$p := \Pr(T \geq T_0 \mid H_0)$$

for a suitable definition of T.

It is important to note that the p-value is a conditional probability. It can be interpreted as the surprise to obtain a certain sample under the assumption that the null hypothesis is true. As such, if we obtain a low p-value we either observed an unlikely event or made an error in assuming that H_0 holds. Unfortunately, the p-value is often misunderstood and misused, despite regular warnings from the statistical community. The gravity of the problem is illustrated by the fact that the American Statistical Association felt forced to release an offical statement containing basic principles for the use of p-values [WL16].

If a binary decision between H_0 and H_1 is required we need to establish a significance level α. The significance level determines a threshold for the test statistics above which the user is willing to discard H_0 and adopt H_1. Alternatively, it determines the size of the tail(s) of the null distribution into which the test statistics of an observation has to fall in order to be considered significant (Figure 4.1). Once we made this choice, there are four possible outcomes of the test (Table 4.1). If the H_0 or H_1 hypothesis holds and is in fact chosen, the test procedure worked correctly. If, however, H_0 holds but is rejected, a so called *type I error* is committed. Vice versa, if H_1 holds and H_0 is not rejected, this results in a *type II error*. The significance level α directly controls the type I error of the test. This means that the expected number of type I errors when conducting the experiment n times with different samples is given as

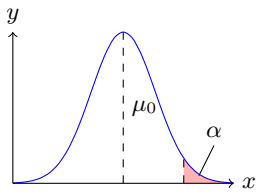


Figure 4.1: An illustration of the critical region in a hypothesis test.

	H_0 correct	H_1 correct
H_0 chosen	-	Type II error (β)
H_1 chosen	Type I error (α)	-

Table 4.1: Possible errors in which applying a significance test can result in. The type I error can be controlled by choosing an appropriate significance threshold α. The type II error is more difficult to control and besides α also depends on the test statistics as well as the number of samples.

$\alpha \cdot n$. The type II error (β) is not directly controlled by α alone. Instead it additionally depends on the sample size and the chosen test statistics. We call the quantity $1 - \beta$ the *power* of the test. Similar to choosing α *a priori* an acceptable power level should be chosen while planning the experiment. Then, in dependence of α the appropriate sample size for obtaining the required power needs to be determined. In practice an *a priori* power analysis can prove difficult, e.g. when analysing public datasets, as collecting a sufficient number of samples is not possible then. Also, for some test statistics it is not possible to reliably estimate their power. Due to this, the power analysis is unfortunately often omitted.

4.1.2 *Multiple Hypothesis Testing*

An illustration of this issue can be found under `https://xkcd.com/882/`.

In most studies more than one hypothesis is tested. This is especially true for biological high-throughput experiments such as microarray analyses, in which thousands of genes are tested for differential expression or categories are tested for enrichment. For each individual test a fixed significance threshold α is used to determine whether to keep or reject the null hypothesis. Remember that the significance threshold as introduced in Section 4.1, controls the type I error. Thus, when conducting 100 tests at significance level $\alpha = 0.05$ we can expect $\alpha \cdot 100 = 5$ false positive predictions by chance. In order to control for this effect, either the significance threshold or the p-values obtained from the hypothesis tests need to be adjusted. Probably the simplest method to do so is the Bonferroni procedure [Bon35; Bon36]. Assume we conducted n tests, yielding p-values $p_1, ..., p_n$. Instead of accepting p_i when it is below the significance threshold we accept p_i if $n \cdot p_i \leq \alpha$ holds. This ensures that the probability of making one or more errors is less than α.

Proof. Let $H_1, ..., H_n$ be the set of tested hypotheses and let I_0 with $n_0 = |I_0|$ be the subset of true null hypotheses. The probability of making at least one false discovery is then given by the probability of rejecting one of the null hypotheses in I_0

$$\Pr\left[\bigcup_{I_0}(p_i \leq \alpha)\right]$$

Substituting α with α/n and applying Boole's inequality yields

$$\begin{aligned}\Pr\left[\bigcup_{I_0}\left(p_i \leq \frac{\alpha}{n}\right)\right] &\leq \sum_{I_0}\Pr\left(p_i \leq \frac{\alpha}{n}\right)\\ &\leq n_0\frac{\alpha}{n} \leq n\frac{\alpha}{n} \leq \alpha\end{aligned}$$

□

The probability of making at least one false discovery is also called *family-wise error rate (FWER)*. Methods that ensure that the FWER is at most as high as a given threshold are said to *control* the FWER. As such, the Bonferroni procedure is one of the strictest available methods. Other methods that achieve tighter bounds, such as the Šidák [Šid68; Šid71] or the Holm-Bonferroni [Hol79] procedures are available.

In a biological setting, where the statistical power is already low due to small sample sizes, controlling the FWER is too conservative [Nak04]. Benjamini and Hochberg [BH95] introduced the concept of *false discovery rate (FDR)* as well as a method using this statistics as a target for adjustment. In contrast to the FWER, the FDR is the *expected* proportion of false discoveries among all significant discoveries. Controlling the FDR instead of the FWER gives more control over the number of acceptable false positive discoveries and is especially helpful in scenarios, in which false positive detections can be tolerated.

If the hypothesis are dependent the procedure by Benjamini and Yekutieli [BY01] should be used instead.

The Benjamini-Hochberg procedure controls the FDR for a set of independent hypothesis and works as follows: given the set of input p-values $p_1, \ldots, p_n$ which w.l.o.g. we assume to be sorted in ascending order, find the largest k such that

$$P_k \leq \frac{k}{n}q$$

where q is the desired FDR level. Given k, reject the null-hypothesis for all H_i with $i = 1, \ldots, k$. Alternatively, each p-value can be adjusted using the formula [YB99]:

$$q_i = \min\left\{q_{i+1},\ \frac{n}{i}p_i,\ 1\right\}$$

Note that while the resulting values still fall into the interval $[0, 1]$, they can no longer be interpreted as p-values. Instead, they indicate the FDR level that can be obtained by rejecting the null-hypothesis for all $q_i \leq q$.

4.1.3 *Enrichment Methods and Hypothesis Tests*

We made a similar argument concerning deregulated subgraphs (Section 3.5).

In terms of hypothesis tests, another motivation for the use of enrichment methods, besides the efficient incorporation of prior knowledge, exists. Consider the search for differentially expressed genes, where each gene is tested for a significant difference in its expression level between two groups. A limiting factor for these analyses is the statistical power of the used hypothesis tests. In this case, the power quantifies the ability to detect truly differentially expressed genes. When

applied at large scale, such as screening for differentially expressed genes, classical hypothesis tests tend to have poor power [Efr07]. Efron [Efr08] argues that by pooling entities into predefined groups, and thus performing less tests, a significant increase in statistical power can be achieved.

Enrichment Method Naming Issues

Unfortunately, the terminology around enrichment methods is not consistent throughout literature. The terms "gene-set approaches", "gene-set analysis", or "gene-set enrichment analysis" are used interchangeably. Furthermore, these names imply that the underlying methodology is only applicable for analysing gene expression data. In general though, these methods can be applied to any annotation defined over a set of biological entities such as proteins, genes, or miRNAs. We, thus do not use the term gene-set enrichment analysis and will instead use the more general term *enrichment method*. Furthermore, this choice avoids possible mixups with the general concept of enrichment algorithms and the *gene set enrichment analysis (GSEA)* method published by Subramanian et al. [Sub+05]. Accordingly, we simply call the result of an enrichment method an *enrichment*. Instead of using the term *gene*, we use *(biological) entity* in order to refer to genes, mRNA, miRNA, or proteins.

4.2 A FRAMEWORK FOR ENRICHMENT ANALYSIS

A detailed description of the mentioned test statistics follows in Section 4.2.4.

The basic principle behind an enrichment algorithm is to determine whether a category contains e.g. more differentially expressed genes or abundant proteins than expected given the remainder of the dataset. There are various ways to compute this. The popular *gene set enrichment analysis (GSEA)* [Sub+07] sorts the input entities in descending order with respect to a relevance score (Section 4.2.4). Next a running-sum is computed by traversing the list and increasing the sum each time a member of the category is encountered. Conversely, the sum is decreased for every entity that is not a member of the category. The maximum deviation of the sum from zero is then the final enrichment score for which a p-value can be computed.

Alternatively, we can determine a set of "interesting" entities, e.g. by imposing a significance cut-off on differential expression. Given a category, we can then count the number of category members among these "interesting" entities. Using a combinatorial model it is possible to determine how likely it is to obtain at least as many category members by chance. This is the idea behind the *overrepresentation analysis (ORA)* [Dră+03] approach (Section 4.2.4).

What both approaches have in common is that they first require a measure or score that reflects the relevance of each input entity. These *entity-level* scores, in combination with a set of categories, are then used

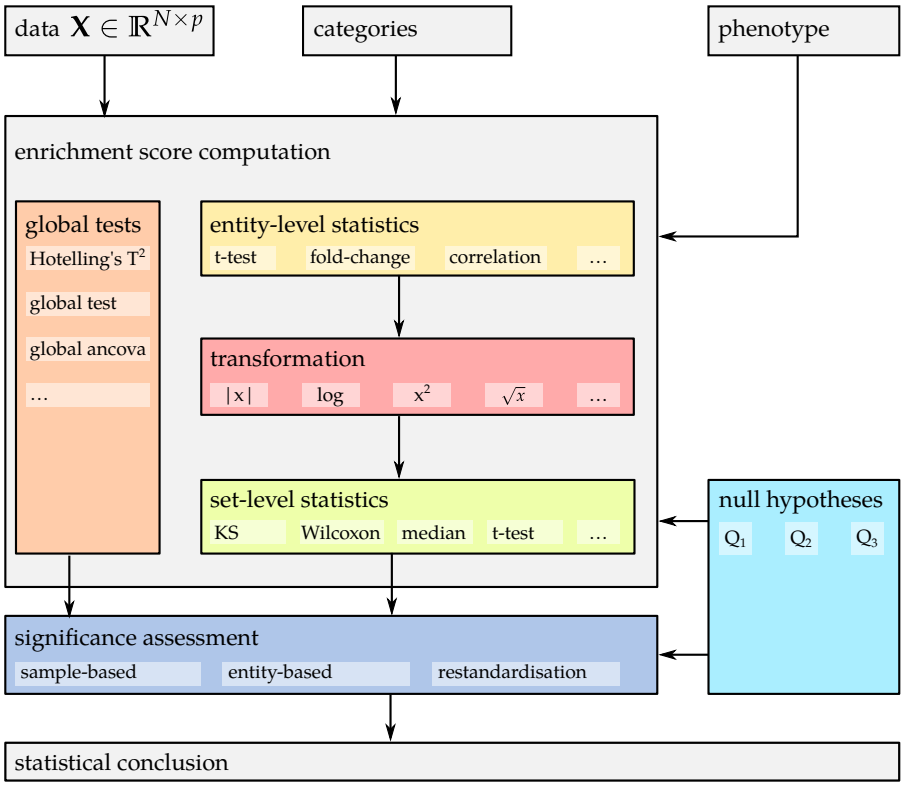


Figure 4.2: Schematic of an enrichment method adapted from Ackermann and Strimmer [AS09]. All enrichment methods depend on data, categories, and phenotypic information. For computing an enrichment score, either a global test of a three step procedure is used. Afterwards, a p-value for the obtained enrichment score is computed.

to compute a *set-level* statistics for which, in turn, a p-value is computed. Ackermann and Strimmer [AS09] recognised these similarities and proposed a generalised framework for building and classifying enrichment methods (Figure 4.2). In this framework, the input is assumed to be a matrix of (normalised) measurements as well as a set of categories that should be tested for significance. Additionally, a phenotype annotation is required to partition the input data into a sample and a reference group.

Given this input, an enrichment score is computed for each category. Here, the framework distinguishes between two schemes for computing an enrichment score. The first scheme comprises three steps: evaluating an *entity-level* statistics, a *transformation* step, as well as the *set-level* statistics. The entity-level statistics reduces the input data from multiple samples to a single *score* for each entity and will be introduced in Section 4.2.2. In Section 4.2.3 we will give a short overview over the transformations that can be applied to the entity-level scores in order to improve the overall performance of the method. Given the transformed scores, we can now use one of the set-level statistics presented in Section 4.2.4 to compute a final enrichment score for each category. The second scheme directly transforms the input data into enrichment scores. As tests adhering to this scheme do not rely on precomputed

scores but use the complete data matrix, they can explicitly incorporate interactions between entities into their underlying models. For this reason, Ackermann and Strimmer [AS09] refer to them as *global tests* (see Section 4.2.5). After enrichment scores have been computed, the significance of the individual scores must be assessed. For this purpose three strategies exist: *sample-based* evaluation, *entity-based* evaluation, and the restandardisation method. We will discuss these strategies in more detail in Section 4.2.1. After this, the raw p-values must be adjusted for multiple testing in order to prevent false positive discoveries (Section 4.1.2). First, however, we will start with how to choose the enrichment method that is the most appropriate for a given research task.

4.2.1 *Choosing the Null-Hypothesis*

At the beginning of every research task, ideally before any data has been generated, a hypothesis that should either be confirmed or disproved, must be chosen. Naturally, this is also true for enrichment analyses. According to Tian et al. [Tia+05], there are two natural ways to state a null hypothesis for deciding the significance of a single category (gene set):

Hypothesis Q_1: "The genes in a gene set show the same pattern of associations with the phenotype compared with the rest of the genes."

Hypothesis Q_2: "The gene set does not contain any genes whose expression levels are associated with the phenotype of interest."

It is important to note that although Q_1 and Q_2 appear similar, there is a subtle difference. Q_1 considers how strongly the entities in a category are associated with the phenotype, compared to entities that are *not* members of the category. As a result, even if the category does not contain entities that are significantly associated with the phenotype, it can be declared significant, as it (seemingly) shows a stronger pattern of association than the remaining genes. In contrast, Q_2 only considers the entities in the category of interest and their association with the phenotype. While Q_2 circumvents the problem of Q_1, it has its own drawbacks. In particular larger categories, which are more likely to contain relevant genes by chance, would be significant under Q_2. Owing to these interpretations of Q_1 and Q_2, Goeman and Bühlmann [GB07] categorise enrichment methods employing Q_1 as *competitive* and methods employing Q_2 as *self-contained*.

Ackermann and Strimmer [AS09] also mention a Q_3 hypothesis that is based on Efron [Efr08] and is not related to the hypothesis presented here.

Nam and Kim [NK08] argue that some enrichment methods, including the popular GSEA procedure [Sub+05], neither compute significance values with respect to Q_1 nor to Q_2, but rather operate on a third null hypothesis:

Hypothesis Q_3: "None of the gene sets considered is associated with the phenotype."

Data: Entity-level scores S, set-level score r, and category c.
Result: Empirical p-value for c.
Set $i = 0$, $k = 0$
while $i < n_{\text{perm}}$ **do**
  $\hat{S}$ = permuteScores(S)
  $\hat{r}$ = setStatistics($\hat{S}$, c)
  if $\hat{r} \geq r$ **then**
    k = k + 1
  end
  i = i + 1
end
return $p = (k + 1)/n_{\text{perm}}$

(a) Entity based.

Data: Data X, set-level score r, and category c.
Result: Empirical p-value for c.
Set $i = 0$, $k = 0$
while $i < n_{\text{perm}}$ **do**
  $\hat{X}$ = permuteSamples(X)
  $\hat{S}$ = entityStatistics($\hat{X}$)
  $\hat{r}$ = setStatistics($\hat{S}$, c)
  if $\hat{r} \geq r$ **then**
    k = k + 1
  end
  i = i + 1
end
return $p = (k + 1)/n_{\text{perm}}$

(b) Sample based.

Algorithm 4.1: Algorithms for computing an empirical p-value for a given category. The essential difference between the entity- and the sample-based statistics is when and in which dimension the randomisation takes place. In both cases, a pseudocount is added to avoid p-values of value zero [Kni+09].

In contrast to Q_1 and Q_2, Q_3 does not make a statement about a single category, but rather assesses a set of categories for a significant association with the phenotype. It can be interpreted as a mixture of Q_1 and Q_2. Nam and Kim [NK08] argue that this is the case for GSEA because it uses the competitive *Kolmogorov-Smirnov (KS)* statistics [Kol33; Smi48] as set-level statistics but employs a self-contained strategy for significance assessment.

The above observations also have practical implications. First, as in the case of GSEA, the chosen set-level statistics can implicitly determine the H_0 hypothesis and thus needs to be chosen with this in mind. Furthermore, the method used for assessing the significance of the enrichment scores needs to be chosen to match the desired hypothesis. To this end, a choice can be made between two main alternatives.

Imagine a category C_i with a score c_i. We can now ask ourselves, how likely it is that another category C_j, consisting of $|C_i|$ randomly selected entities, has a score larger or equal than c_i. We call this scenario, which reflects Q_1, the *entity-based* evaluation strategy for which we can compute empirical p-values by using a *permutation test* (Algorithm 4.1). In essence, a permutation test estimates the null distribution of the set-level scores. This is done by generating random permutations of the entity-level scores that are then used as input for recomputing a collection of null set-level scores. The number of times such a permuted set-level score was equal to or exceeded the original score is counted and divided by the total number of permutations yielding an empirical p-value. The advantage of the entity-based strategy is that it takes the distribution of all entity-level scores into account. Its disadvantage is that the correlation structure of the entities in a category is neglected. This correlation structure is likely to be important as categories, by design,

For some hypothesis tests, such as the KS test or the t-test family, exact entity-based p-values can be computed. We indicate this in the respective discussion.

tend to consist of functionally related entities [ET07].

However, we can also ask the question, how likely it is that C_i was assigned a score larger or equal than c_i given that the phenotype labels of the sample were randomly assigned. We call this scenario, which reflects Q_2, the *sample-based* evaluation strategy. Clearly, permuting the samples leaves the correlation structure inside a category intact. Unfortunately, the entity-level score distribution is destroyed, as all scores are recomputed after sample permutation and only the entities in a category have an influence on the final enrichment score [ET07].

The terminology used by [ET07] is randomisation for the entity-based and permutation for the sample-based strategy.

Efron and Tibshirani [ET07] argue that in order to combine the strengths of the entity-based and the sample-based evaluation strategy, both should be executed simultaneously. They name this combined strategy the *restandardisation* method and apply it together with their proposed maxmean statistics.

From a computational point of view, the entity-based strategy is often more efficient, as it can exploit the distribution of the enrichment scores. For example it is possible to compute p-values for t-test based methods using Student's t-distribution. In contrast, the sample-based as well as the restandardisation strategy always necessitate the use of permutation tests, which can require considerable amounts of runtime to compute.

4.2.2 *Entity-level Statistics*

Entity-level scores are scalar values that reflect the "importance" of an entity. Commonly, entity-level scores are derived from the differential analysis of two sample groups. Scores are computed by evaluating a statistics that reduces the input data to a singe scalar value. Ackermann and Strimmer [AS09] report that the choice of entity-level score only has little effect on the overall results of the enrichment procedure. It should be noted, though, that for small sample sizes the choice of an appropriate entity-level statistics may well have a higher influence.

See Appendix A.1.3.

In the following we will list a few popular choices for such entity-level statistics. Let $\boldsymbol{X} = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^{p \times n}$ and $\boldsymbol{Y} = (y_1, y_2, \ldots, y_n) \in \mathbb{R}^{p \times m}$ be the sample and reference set, respectively. The sample mean and sample variance of entity k are denoted by $\bar{x}_k$ and s^2_{Xk}, respectively.

Fold-Change The most basic entity-level score is the *fold change*. It is defined as the ratio between the mean expression of the two groups $\boldsymbol{X}$ and $\boldsymbol{Y}$:

$$\text{fc}(k) = \frac{\bar{x}_k}{\bar{y}_k}$$

As expression values are larger than zero, the fold change is also positive and distributed around 1. However, the distribution is asymmetric as cases where $\bar{y}_k > \bar{x}_k$ holds occupy the interval $(0, 1)$ while the converse cases occupy the interval $[1, \infty)$. To make both ranges comparable it is common to apply a logarithm yielding the *log-fold-difference*:

$$\log_2(\text{fc}(k)) = \log_2(\bar{x}_k) - \log_2(\bar{y}_k)$$

For an expression dataset, a $\log_2$-fold-change of 0 is thus equivalent with no change in expression between $\boldsymbol{X}$ and $\boldsymbol{Y}$ while a $\log_2$-fold-difference of ± 1 can be interpreted as an over- or under-expression of factor two, respectively.

The t-Test Family As the fold change is simply defined as the ratio of two means, it does not respect the variance of the measurements. This can be problematic as, for instance, a shift of the mean expression between the sample and reference group for an entity that exhibits a high variability is not as significant as a similar change for an entity that exhibits a much lower variability. Accordingly, fold changes are biased towards entities with high variability as it is more likely that they obtain a larger score than other entities. To account for this, statistics that incorporate the sample variances should be used. The most common choice for such a statistics is Welch's t-test [Wel47; Sat46], which is appropriate for normally distributed samples.

The degrees of freedom for Welch's t-test are computed via the Welch-Satterthwaite equation.

$$t_k = \frac{\bar{x}_k - \bar{y}_k}{\sqrt{\frac{s^2_{Xk}}{n} + \frac{s^2_{Yk}}{m}}}$$

For paired samples, Student's t-test can be applied to the pairwise differences $d_{ki} := x_{ki} - y_{ki}$ resulting in the test statistics:

Paired samples imply that $n = m$.

$$t_k = \frac{\sqrt{n} \cdot \bar{d}_k}{s_{Dk}}$$

If applicable, the paired version of the t-test has the advantage that it offers higher statistical power than the independent version. Additionally, simpler versions of the t-test can be applied if the mean of a sample is to be compared against a known, global mean.

Shrinkage t-Test While high-throughput experiments are able to capture profiles for thousands of entities simultaneously, most of the time the number of measured samples is considerably lower. The practical implication of this is that all estimates obtained from the data, such as sample means and sample variances, are subject to a sizeable amount of uncertainty. For the t-test family, errors in the sample variance are especially problematic, as too small or too large variances can over- or understate the importance of an entity. To combat this, Opgen-Rhein and Strimmer [OS07] introduced the *shrinkage t-statistic*. The idea behind the shrinkage t-statistic is to move ("shrink") the estimated entity variances closer to the overall median variance, thereby increasing small and decreasing large variances. James and Stein [JS61] showed that, if multiple, simultaneous estimates are required for some parameters, this combined strategy is guaranteed to perform better than estimating the parameters individually. Most intriguingly, the estimated quantities do not need to be related in *any* way [EM77]. This phenomenon is often entitled "Stein's paradox". Another advantage of the James-Stein framework is that the optimal amount of shrinkage is determined from the data alone. As a consequence no additional tuning parameter is introduced into the estimation process.

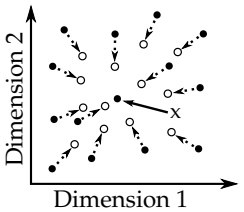


Figure 4.3: Sketch of the effect of shrinkage procedures. The original data points (solid circles) are shifted towards the shrinkage target x, resulting in new, "shrunk" data points.

In the following we introduce the estimator for the sample variance by Opgen-Rhein and Strimmer [OS07]. Recall the usual, unbiased estimator for the sample variance (Equation (A.1))

$$s_{Xk}^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_{ki} - \bar{x}_k)^2$$

Instead of directly using s_{Xk}^2 in the denominator of the t-statistic, we replace it with a convex combination of s_{Xk}^2 and the median standard deviation of all genes s_{median}:

$$s_{Xk}^\star = \lambda^\star s_{\text{median}} + (1 - \lambda^\star) s_{Xk}$$

This effectively shifts all variances closer to s_{median}. The optimal mixing proportion $\lambda^\star$ is given by

$$\lambda^\star = \min\left(1, \frac{\sum_{k=1}^{n} \widehat{\text{var}}(s_{Xk})}{\sum_{k=1}^{n} (s_{Xk} - s_{\text{median}})^2}\right) \tag{4.1}$$

The term $\widehat{\text{var}}(s_{Xk})$ denotes an estimate of the variance of the sample standard deviations and can easily be computed using the unbiased variance estimator. This formula can be interpreted as follows: if the variances can be reliably estimated from the data and, thus, $\widehat{\text{var}}(s_{Xk})$ is small, only little shrinkage will take place. Conversely, if they cannot be estimated reliably, the nominator is large and more shrinkage towards the median is used. On the other hand, the denominator in Equation (4.1) measures how well the median standard deviation represents all standard deviations. If this difference is large, less shrinkage is used. For large sample counts, the shrinkage t-statistic converges to the usual t-statistic. This means that the former can be used unconditionally used as a replacement for the latter.

Z-score In some cases, such as when analysing a patient sample for treatment optimisation, it is desirable to compare just a single sample against a larger reference group. As it is not possible to compute variances for a single sample, it is also not possible to use a t-test for computing entity-level scores. To at least incorporate the variance of the reference set, the z-score statistics can be used. It is defined as follows:

$$z_k = \frac{x_k - \bar{y}_k}{s_{Yk}}$$

Using the z-score, normally distributed data can be transformed such that it follows a standard normal distribution $\mathcal{N}(0, 1)$, which can be used to determine a p-value for x_k.

Just as Spearman's correlation (Section 3.5.3) is a rank-based version of Pearson's correlation, the Wilcoxon test is a rank-based version of Student's t-test.

Wilcoxon rank-sum test The *Wilcoxon* rank-sum test [Rin08] is a non-parametric test. It shines as an alternative to the independent two-sample t-test, if the input data does not follow a normal distribution. Even in cases where the t-test is applicable, the Wilcoxon rank-sum test offers competitive performance and greater robustness against outliers.

Just as the t-test, it can be used check whether the expected values of two samples coincide. Compared to the latter, it is solely based on the relative order of the values between the two samples. The test statistics is based on the running sum W:

$$W = \sum_{i=1}^{n} R(x_i) \tag{4.2}$$

Here, $R(x_i)$ represents the rank of value x_i in the sorted and pooled list of sample values. For $n > 25$ and $m > 25$, W is approximately normally distributed [Rin08] with mean μ_W and variance s_W^2.

$$\mu_W = \frac{n(n+m+1)}{2} \qquad s_W^2 = \frac{n \cdot m(n+m+1)}{12} \tag{4.3}$$

For larger sample sizes, this allows to efficiently approximate p-values. For small sample sizes, the exact p-values for W can be looked up in a table [Kan06]. When the Wilcoxon rank-sum test is used as a set-level statistics (see below), these p-values need to be interpreted with respect to the entity-based strategy.

4.2.3 *Transformation*

Instead of directly using the computed entity-level scores as input for the set-level statistics (cf. Section 4.2.4), it is possible to first apply a score transformation to them. Examples for such transformations are the square root, logarithm, absolute value, or square of the input scores. The main reason for using a transformation is to improve the performance of the enrichment method. In this regard, the most valuable transformations are probably the absolute value and the square, as they allow to fold negative and positive scores together. This is useful when only the deregulation of a given category is of interest and not, whether its constituents are up- or downregulated. Also, for some methods in which positive and negative scores are likely to cancel out, applying the absolute value of square transformation can be crucial (cf. Section 4.3.2). Applying the square root or logarithm transformation can be helpful for dampening the effect of outliers.

4.2.4 *Set-level Statistics*

The actual enrichment can be obtained by computing a set-level statistics. Given a list of entity-level scores, it computes a *set-level* (or *enrichment*) *score* for a given category. The choice of the set-level statistics can have a significant impact on the enrichment method's ability to detect enriched categories. Consequently, a large number of set-level statistics has been published. The most popular choices are the KS statistics, used in GSEA [Sub+05], and the hypergeometric test, used in *overrepresentation analysis (ORA)* [Dră+03]. In the following, we will describe a selection of the most commonly used set-level statistics in more detail.

See Section 4.3.

In order to do so, we categorise the methods into *non-parametric*, *parametric*, and *averaging* methods. The hypergeometric test is not directly comparable to the other methods and will, thus, be treated in its own section. We start with the discussion of non-parametric statistics.

Non-Parametric Statistics

Naturally, parametric tests tend to have higher power than non-parametric competitors, if their assumptions hold (cf. [HL56] or [She03]).

Many hypothesis tests, such as the t-test, assume that the input data follows a certain probability distribution. In real measurements, this is rarely the case. While applying such parametric statistics will nonetheless yield p-values, these are likely to be skewed. Non-parametric statistics do not make any assumptions about the distribution underlying their input data. Thus, they are extremely robust and universally applicable. Even in cases where a parametric statistics applies, non-parametric statistics offer a competitive, albeit slightly lower power [HL56].

Two examples for non-parametric set-level statistics that are commonly employed for enrichment analysis are the Wilcoxon statistics and the KS statistics. Both methods are rank-based and are computed by evaluating a running sum. The formulation of the Wilcoxon statistics as a set-level statistics is identical to the entity-level version presented in Section 4.2.2.

Whereas the Wilcoxon statistics counts how often samples from group $\boldsymbol{X}$ precede samples from group $\boldsymbol{Y}$ and vice-versa, the KS statistics traverses a sorted list of samples and adds or subtracts a constant value depending on whether a sample from group $\boldsymbol{X}$ or $\boldsymbol{Y}$ has been encountered. The maximum deviation from zero is then the output of the KS statistics. More formally, let $L = \{l_1, l_2, \ldots, l_n\}$ be a list of entities ordered according to their entity-level scores with $n := |L|$ and $C \subset L$ be a category with $m := |C|$. Now, we are able to define the running sum RS:

$$\begin{aligned} \text{RS}(0) &= 0 \\ \text{RS}(i) &= \text{RS}(i-1) + \begin{cases} n-m & \text{if } l_i \in C \\ -m & \text{otherwise} \end{cases} \end{aligned}$$

The version presented here is more space efficient than the version published by Keller, Backes, and Lenhof [KBL07].

The value of the test statistics is the maximum deviation RS_{max} of RS from zero. Figure 4.4 provides an example for the KS running sum. For the entity-based strategy, an exact p-value can be computed using dynamic programming [KBL07]. The idea of this approach is to compute the number Z of permutations that achieve an (absolute) score less than RS_{max}. The final p-value can then be computed as

$$p = 1 - \frac{Z}{\binom{n}{m}}$$

Let $M \in \mathbb{N}^{(m+1)\times(n-m+1)}$ be the dynamic programming matrix. In or-

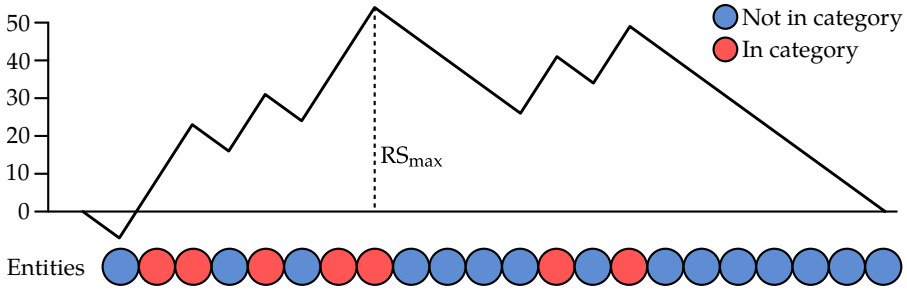


Figure 4.4: An example KS running sum. Red dots represent genes in the category under consideration. The maximum deviation of the running sum from zero (RS$_{max}$) is marked with a dashed line and serves as the value of the test statistics.

der to initialise M we set

$$M(i,0) = \begin{cases} 1 & \text{if } i \cdot (n-m) < |\text{RS}_{\text{max}}| \\ 0 & \text{otherwise} \end{cases}$$

$$M(0,k) = \begin{cases} 1 & \text{if } k \cdot m < |\text{RS}_{\text{max}}| \\ 0 & \text{otherwise} \end{cases}$$

The recurrence is given by

$$M(i,k) = \begin{cases} M(i-1,k) + M(i,k-1) & \text{if } (*) \\ 0 & \text{otherwise} \end{cases}$$

where $(*)$ is $|\text{RS}_{\text{max}}| < i \cdot (n-m) - k \cdot m < |\text{RS}_{\text{max}}|$.

Subramanian et al. [Sub+05] proposed a weighted version of the KS statistic. For each entity $l_i \in C$ that is a member of the category, the value w_i, which is derived from the entity's score $w(l_i)$, is added to the running sum.

$$w_i = \frac{|w(l_i)|^p}{N_R}$$

The term $N_R := \sum_{l_i \in C} |w(l_i)|^p$ is used as a normalisation factor. The parameter $p \in \mathbb{R}_0^+$ allows to control how much high scoring entities are preferred to low scoring ones. For entities that are not a member of C the value $(n-m)^{-1}$ is subtracted from the running sum. For $p = 0$ this formulation is equivalent to the unweighted KS statistics. One benefit of using weights is that it allows to control for scores that are closely distributed around zero and thus should have little influence on the value of the statistics. Additionally, artefacts stemming from entities with a low abundance, such as transcripts with low copy number, are mitigated by using weights. Unfortunately, the weighted version of the test does not allow the computation of an exact p-value even when using the entity-based strategy. Instead the permutation test approach presented in Algorithm 4.1 needs to be applied.

Parametric Statistics

Representatives of the parametric set-level statistics are the shrinkage and Student's t-test (cf. Section 4.2.2). As with the Wilcoxon statistics,

the formulations for the t-test family statistics are the same as in the entity-level case. Advantages of using the t-test are that scores as well as p-values for the entity-based statistics can be computed more efficiently compared to the non-parametric case. Two ways of applying the t-test are possible. First, the means of the category members and the remaining genes can be compared (*two-sample t-test*). Second, the mean of the category members can be compared to the global mean. In the latter case, only the variance of the category is considered (*one-sample t-test*).

Averaging Statistics

A trivial way to compute an enrichment score for a category, is to compute the average of its member's entity-level scores. Similarly, the median or sum of all category entries can be used. Remarkably, these simple statistics perform competitively compared to more sophisticated methods such as the KS statistics [AS09]. Based on these *averaging methods* Efron and Tibshirani [ET07] devised the *maxmean* statistics. Instead of computing the overall mean of the category, the authors propose to compute the mean of the positive and negative member separately and using the maximum value as a test statistics in order to avoid cancellation of large values with opposing signs (cf. Section 4.3).

The Hypergeometric Test

Often a researcher obtains a list of "interesting" entities, which we will henceforth call the *test set*. Examples for test sets are list of differentially expressed genes or the proteins detected in a sample. The question is whether the members of the test set share a common characteristic. To this end, functional annotations and thus categories can be used. To determine whether the members of a category are over- or under represented relative to what is expected by chance, a simple urn model can be used. Suppose we are given a universe of possible entities R with m members, which we will call the *reference set*. Additionally, we are given a *test set* $T \subset R$ with $n := |T|$ of entities and a category $C \subset R$ with $l := |C|$. We can ask ourselves the question: "Is the number of entities $k := |T \cap C|$ significantly different from the number expected for any randomly chosen set T'?"

The probability of drawing a set of length n containing k elements from C can be calculated using basic combinatorics (Figure 4.5):

$$P_H(k, l, m, n) = \frac{\binom{l}{k}\binom{m-l}{n-k}}{\binom{m}{n}}$$

$P_H(k, l, m, n)$ is the probability density function of the hypergeometric distribution (cf. [Riv+07; Bac+07]).

The proof uses Vandermond's identity.

The expected value of hits for a randomly chosen test set with l members is given by $k' = \frac{l \cdot n}{m}$. It is now possible to compute a one-sided

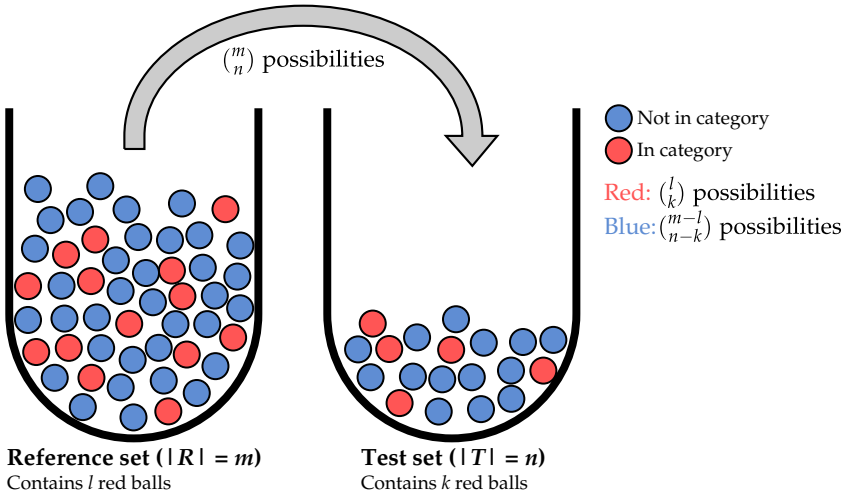


Figure 4.5: Urn model for the hypergeometric test. The reference set contains entities that either belong to the category (red) or not (blue). From the reference set the test set is drawn.

p-value for our input test set T:

$$p_c = \begin{cases} \sum_{i=k}^{n} P_H(i,l,m,n) & \text{if } k' < k \\ \sum_{i=0}^{k} P_H(i,l,m,n) & \text{if } k' \geq k \end{cases}$$

If the test set T contains entities that are not members of R, Fisher's exact test should be used instead of the hypergeometric test:

$$P_F(i,k,l,m,n) = \frac{\binom{n}{i}\binom{m}{l+k-i}}{\binom{m+n}{l+k}}$$

The p-values are defined analogously:

$$p_c = \begin{cases} \sum_{i=k}^{n} P_F(i,k,l,m,n) & \text{if } k' < k \\ \sum_{i=0}^{k} P_F(i,k,l,m,n) & \text{if } k' \geq k \end{cases}$$

Due to its reliance on the test and reference sets, the hypergeometric test is not directly comparable to the other enrichment methods. In this regard, requiring a test set is both, the biggest advantage and disadvantage of the hypergeometric test. On the one hand, the independence from entity-level scores allows to employ the hypergeometric test in settings where the remaining methods are not applicable. An example would be a differential proteomics pipeline in which the presence (or absence) of proteins is detected, but no quantitative information has been collected. On the other hand, constructing the test set in cases where entity-level scores are available can be difficult. Consider a score measuring the degree of differential expression for each gene. Which cut-off should be chosen to distinguish between differentially

expressed and "normal" genes? In essence such a thresholding step introduces a tuning parameter that can have a large influence on the obtained results. Similarly, the choice of the reference set is also critical for the performance of the method. A larger reference set makes it less likely to randomly draw members of a category and, thus, leads to overall lower p-values. On the other hand, choosing a reference set that is too small may lead to low statistical power.

How to choose an appropriate reference set is not always clear. In general, it should comprise all entities that can be detected by the experimental platform. For mRNA microarrays this means that the reference set should consist of the genes for which probes are present on the array. For RNA-seq the answer is less clear. As, in theory, all genes that can be expressed can be detected by the method, a reasonable choice for the reference set would be a list of all actively transcribed genes. For proteomics pipelines, which, at the time of writing, are not capable to capture all proteins that are present in a sample, there is no clear answer. One approach would be to use the union of all detected proteins across all samples. Alternatively a list of all proteins can be used. In this case, the resulting, lower p-values should then be compensated by a stricter significance threshold.

4.2.5 *Global Tests*

Tests that directly transform measurement data into enrichment scores are called "global tests" by Ackermann and Strimmer [AS09]. The advantage of global tests, as opposed to the previously presented methods, is that they can incorporate the correlation between entities into their respective models. Examples are Hotelling's T^2-test [Hot31], the global ANCOVA approach [M+05], and the aptly named *global test* procedure [Goe+04].

In the remainder of the thesis, global tests will not play a further role. An exception is Hotelling's T^2-test which we will describe and evaluate in Section 4.4.2. For more details on the remaining methods, we refer the reader to the cited literature.

4.2.6 *Summary*

Enrichment analysis is a large field and accordingly a wide variety of enrichment algorithms has been created. Commonly an enrichment algorithm is composed of a procedure for deriving entity-level scores, set-level scores, and finally a method for significance assessment. Owing to the choices made for the above components, each algorithm is based on different assumptions on how an *enriched category* is defined. Hence, each algorithm comes with a trade-off between advantages and disadvantages that need to be carefully vetted each time an enrichment analysis is required. Consequently, a wide variety of reviews exist that attempt to characterise the performance of enrichment methods under different conditions [AS09; ET07; HSL09; Hun+11; KSB12; Nae+12].

This shows that choosing an appropriate enrichment method is by no means a trivial task. In the next section, we also attempt to provide a comparison of some enrichment algorithms. However, we focus on deriving general advice in contrast to recommending a single technique.

4.3 EVALUATION

Choosing a "good" enrichment algorithm is difficult. As outlined before, there exist no experimental datasets for which the truly enriched categories are known. Hence, conducting studies on synthetic data remains as the only viable option to obtain performance estimates for an enrichment method. In this section, we conduct an evaluation based on synthetic data with the goal to derive general guidelines for choosing an enrichment algorithm that is appropriate for a given situation. It should be noted that this evaluation *by no means* provides a full coverage of every possible usage scenario.

Our evaluation is based on the following procedure: first, we randomly create a set of categories. Next, we generate a dataset lacking any kind of differential expression. We will refer to this dataset as the *null dataset*. Evaluating our categories on a null dataset gives us a baseline for the number of false positives that can be expected for each method. Then, all methods are run on datasets that were generated to produce a randomly chosen subset of enriched categories. We can treat the information whether a category is enriched or not as a class label. This allows us to compute performance metrics for the individual methods that are usually used for evaluating classifiers.

Using only categories created by uniformly sampled genes does not necessarily reflect how collections of categories are structured. As, for example, some genes fulfil a wider spectrum of functions or are more thoroughly researched, we expect that they can be found more often in categories than other genes. Accordingly, the enrichment scores for real-world categories can be expected to be more correlated than for our synthetic categories. If this is the case, it may prove difficult to distinguish the truly enriched from merely correlated categories. To account for this effect, we also evaluate the algorithms on a set of categories extracted from Reactome [Jos+05]. In summary, the following experiments were executed:

1. Synthetic categories on data without differential expression.
2. Synthetic categories on data with differential expression.
3. Real-world categories on data without differential expression.
4. Real-world categories on data with differential expression.

Before we present the results of the experiments, we first outline the data generation procedure for categories as well as datasets.

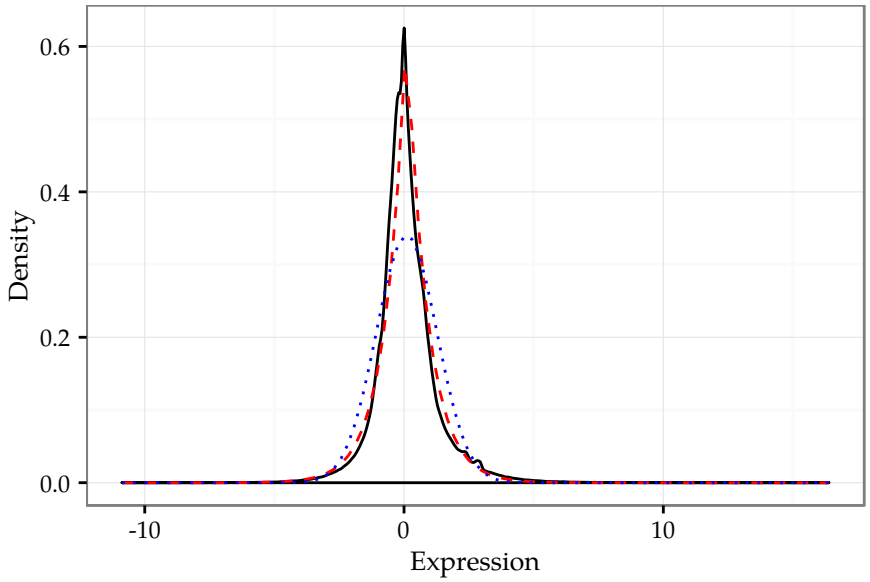


Figure 4.6: Empirical distribution of normalised expression values (black, solid) in the Wilm's tumour dataset. The data is well described by a Laplace distribution (red, dashed). In contrast the fitted normal distribution (blue, dotted) has too little probability mass in the centre and falls off too slowly. Also, too little probability mass is placed on the tails.

4.3.1 *Data Generation*

Generating synthetic data for the evaluation of algorithms has various advantages and disadvantages. The major advantage is that it is possible to create data following a known theoretical model and, thus, that all properties of the generated dataset are known. In the context of enrichment analysis this means that the categories containing significant enriched genes have been determined *a priori*. However, this is also the biggest disadvantage of synthetic data. As the model for data generation can be freely specified, it is likely that the conducted analyses are biased towards methods that share the same model assumptions. If the model does not reflect experimental data well enough, the results obtained using the generated data only contain limited amounts of information about the performance of the methods in a real-world scenario. Nevertheless, using synthetic data allows to learn about the properties of a method in a controlled environment. Most importantly, though, using synthetic data is often the only way to validate methods for which no gold standard is available.

Expression Values

The data generation model used in this study was designed to reproduce the Wilm's tumour microarray expression dataset introduced in Section 2.4. Generally, the expression of microarray data is assumed to follow a normal distribution, which works well for most applications. However, when examining the distribution more carefully, it becomes

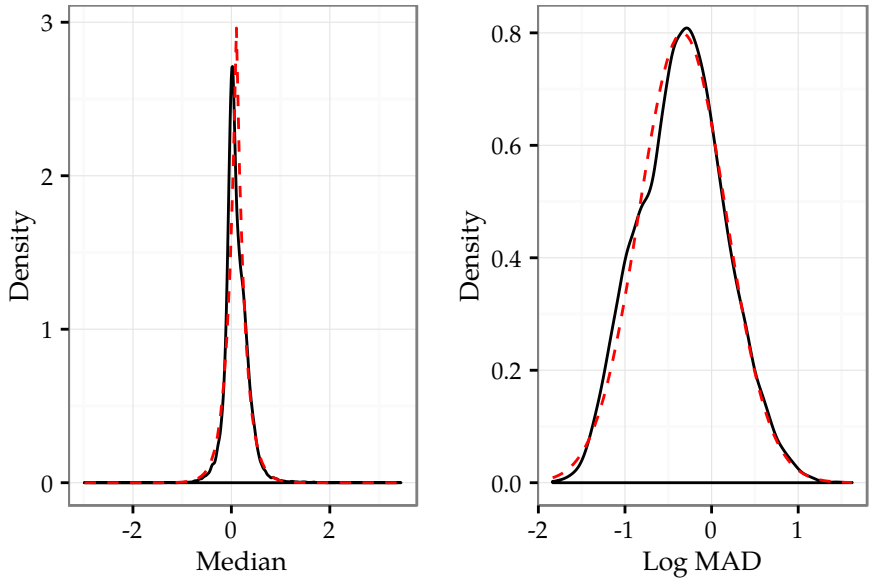


Figure 4.7: Empirical distribution of the per-gene medians and MADs (black, solid) in the Wilm's tumour dataset. The median follow a Laplace distribution, while the MADs roughly follow a log-normal distribution (red, dashed).

apparent that a normal distribution does not fit the data perfectly. The experimental density is narrower and has smaller support than the theoretical normal fit. A distribution that models the data much more faithfully is the Laplace distribution:

$$f(x) = \frac{1}{2\sigma} e^{-\frac{1}{2}\frac{|x-\mu|}{\sigma}}$$

The Laplace distribution is closely related to the normal distribution. Both methods are parametrised using the location parameter μ and the scale parameter σ. The maximum likelihood estimators for μ and σ in the normal case are the mean and standard deviation, respectively. In the Laplace case, they are the median and *mean of absolute deviations (MAD)*

$$\frac{1}{N}\sum_{i=0}^{N}|x-\mu|$$

are used. To indicate that a random variable x follows a normal distribution with mean μ and standard deviation σ we write $x \sim \mathcal{N}(\mu, \sigma^2)$. To indicate the same for the Laplace distribution we write $x \sim \mathcal{L}(\mu, \sigma)$, with μ and σ representing the median and MAD. Purdom and Holmes [PH05] propose to use an asymmetric extension of the Laplace distribution, to better adapt to skewness in expression data sets. Here, we restrict ourselves to the ideal, symmetric case.

In our data generation scheme, we need to sample expression values for each gene from a gene-specific distribution. This allows to simulate differential expression by adding a shift to each genes expression values. In order to derive per-gene distributions that are compatible with the overall distribution of the expression values, we examined

the distribution of the gene medians (μ_0) and MADs (σ_0). Again, the Laplace distribution is a good fit for the distribution of the medians. The MADs, however, follow a log-normal distribution (cf. Figure 4.7):

$$\mu_0 \sim \mathcal{L}(0, \sigma_m)$$
$$ln(\sigma_0) \sim \mathcal{N}(\mu_v, \sigma_v^2)$$

Here, σ_m represents the MAD of the gene medians. The variables μ_v and σ_v are the mean and standard deviation of the log gene standard deviations, respectively. Together, σ_m, μ_v, and σ_v are hyperparameters which we will estimate from the experimental data. We omitted the median of the gene medians from the model, as a constant shift of all expression values will cancel out in the entity-level statistics.

Expression values generated following the above scheme do not correspond to differentially expressed genes. Instead, they constitute the null model or "normal" genes. To introduce differential expression, the expression values in the sample group need to be shifted relative to the expression in the reference group. To create a significant shift, the median expression value of the sample group is shifted to the corresponding $\alpha = 0.05$ significance threshold, which corresponds to a shift of $\delta \approx 2.303$ for the standard Laplace distribution. To create some variation in the gene scores, this adjusted median is modified by adding Gaussian noise. Finally, the shift is scaled with the gene expression MAD in order to account for the different magnitude of dispersion across the genes. Put together, this yields the following generative models for null and differential gene expression:

$$x_{\text{null}} \sim \mathcal{L}(\mu_0, \sigma_0)$$
$$x_{\text{diff}} \sim \mathcal{L}(\mu_0 + \epsilon\sigma_0, \sigma_0)$$
$$\epsilon \sim \mathcal{N}(\delta, \sigma_\epsilon^2)$$

The noise parameter σ_ϵ^2 is set to a constant value of 0.4.

The above model only generates overexpressed genes. This means that the distribution of the expression values is asymmetric and all differentially expressed genes are being shifted to the right. As this may unfairly bias the results towards certain methods, we additionally generate data where the direction of the shift is chosen via a coin flip.

Categories

The used categories were created by uniformly sampling genes from a list of gene names obtained from the Agilent SurePrint G3 microarray platform which was used for measuring the Wilm's tumour data. In total 500 categories were created. As biological categories are usually not uniformly distributed across the genome, but tend to cluster around well-studied genes and pathways, we additionally perform the same evaluation using real biological categories. For both sets of categories, we randomly selected 10 % of their members to serve as enriched categories. To artificially enrich them, we selected 33 % and 66 %

of their genes as differentially expressed. The expression values were then generated as described above.

As biological categories we use the pathway information contained in the Reactome database [Jos+05] to examine the behaviour of the enrichment algorithms on biological categories. In total we extracted 1508 categories from Reactome. As with the artificial categories, we randomly selected 10 % of the categories as differentially expressed and chose 33 % and 66 % of their members as differentially expressed.

4.3.2 *Results*

We evaluated the above model using the parameter settings derived from the Wilm's tumour dataset (cf. Figure 4.7) as well as the noise parameter σ_ϵ, which we fix to a small value:

$$\mu_v \approx -0.336 \qquad \sigma_v \approx 0.499$$
$$\sigma_m \approx 0.167 \qquad \sigma_\epsilon = 0.4$$

Furthermore, we consider the following algorithms implemented in the GeneTrail2 server: *mean*, *sum*, *median*, *one sample t-test (1s-t-test)*, *two sample t-test (2s-t-test)*, *Wilcoxon test*, *weighted* and *unweighted KS*, as well as *ORA*. The ORA method could only be evaluated for the entity-based p-value-strategies, due to limitations in the used C++ implementation. To create the required test set for ORA, we sorted the gene list by score and selected the upper and lower 0.025 % into the test set. All computations were repeated ten times with differing random seeds in order to compute standard deviations.

Null-Model

In order to judge how the algorithms behave if no signal is present in the data, we generated a dataset without any differentially expressed genes. For this experiment, we tested all methods using the entity- and sample-based p-value strategies. We evaluate the number of false discoveries at the 0.01 significance threshold.

Not many differences for the synthetic categories could be detected between the methods (Figure 4.8). For the entity-based strategy the ORA method achieves the lowest number of false positives. For the sample-based strategy no clear winner exists. It should be noted that in all cases, the number of detected false positives is higher than the expected proportions of $500 \cdot 0.01 = 5$ false discoveries. This highlights the importance of adjusting for multiple hypothesis testing.

For the Reactome based categories (Figure 4.9), the results for the entity-based p-value strategy show a higher variance, than those for the sample-based strategy. Again the ORA method *on average* detects substantially less false positives than competing approaches. For the sample-based strategy, as with the synthetic categories, no clear best algorithm can be identified.

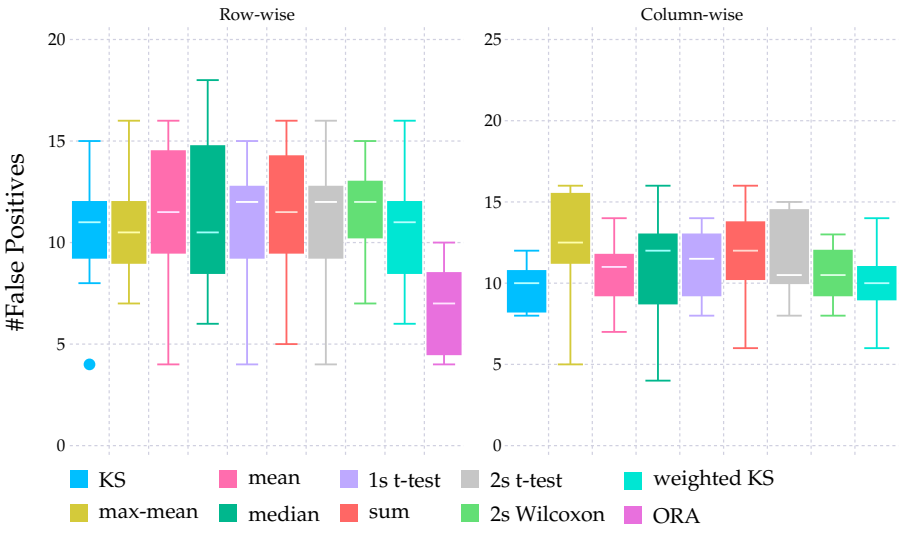


Figure 4.8: Number of false positive categories detected for the synthetic null-dataset using both, the entity-based (left) and the sample-based (right) p-value computation strategy. For both strategies the number of false positives was evaluated at the p-value cut-off 0.01. ORA was only evaluated for the entity-based strategy. In total, 500 categories were being evaluated.

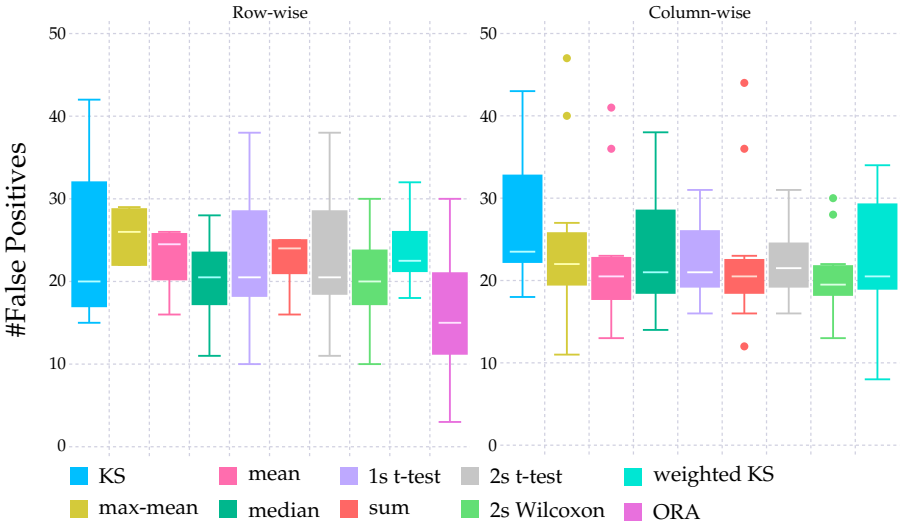


Figure 4.9: Number of false positive categories detected for the Reactome null-dataset using both, the entity-based (left) and the sample-based (right) p-value computation strategy. For both strategies the number of false positives was evaluated at the p-value cut-offs 0.01. ORA was only evaluated for the entity-based strategy. In total, 1508 categories were being evaluated.

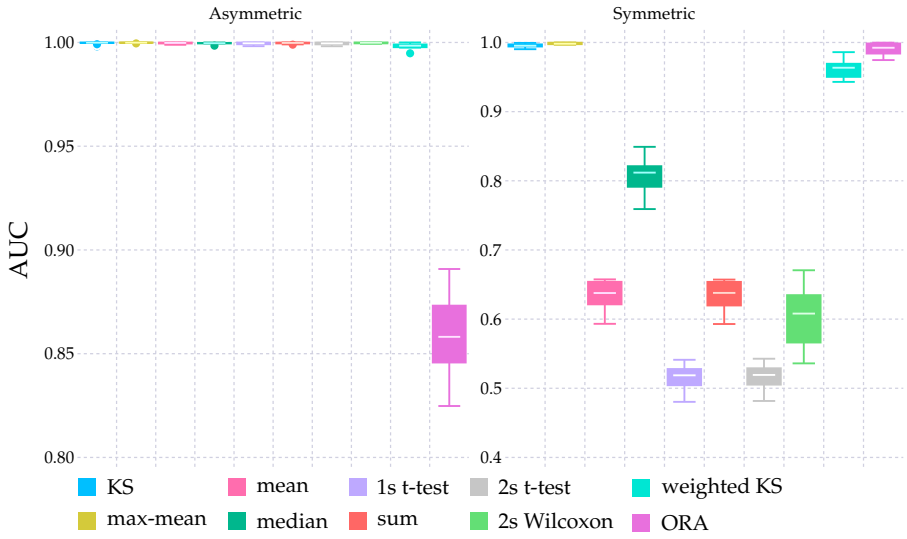


Figure 4.10: AUC of various set-level statistics on synthetic categories where differentially expressed genes are distributed asymmetrically or symmetrically. For p-value computation the entity-based strategy was used. If a category was chosen as significant, 66 % of the genes were generated as differentially expressed.

Significant Categories

Next, we take a look at the results of the datasets for which significant categories were generated as explained above. For the datasets where only 33 % of the genes in an enriched category were chosen as differentially expressed, all methods perform poorly with AUC values of approximately 0.5. For datasets with 66 % significant genes, the performance of the methods improved significantly (Figure 4.10). The most apparent difference can be seen between the symmetric and the asymmetric datasets. Here, all methods except ORA hardly misclassify any categories for the asymmetrically distributed case. The failure of ORA in this case can be attributed to the thresholding used to determine the most up- *and* downregulated genes, whereas the dataset only contains upregulated genes. In the symmetric case, only the (weighted) KS, max-mean, and ORA set-level statistics manage to achieve near perfect scores. The Wilcoxon statistics relies on comparing the sum of ranks of category members against the ranks of the non-members. If the entity level scores are symmetrically distributed each category roughly contains the same number low ranking and high ranking genes which effectively cancel out. This effect is also visible for the remaining methods that are all based on averaging. As with the Wilcoxon test, if genes with a positive and a negative enrichment score are member of a category their sum is expected to lie in the vicinity of zero. Consequently these methods should only be employed if the scores have been transformed using the absolute value or square transformation (cf. Section 4.2.3). When applying the set-level statistics to the biological categories, no significant change in behaviour could be detected. (cf. Appendix C.2).

4.3.3 *Summary*

All enrichment methods are able to detect most of the of the enriched categories, given that the data contains enough signal. If too few genes are differentially expressed, the detection rate drops severely. To ensure optimal performance, the methods and eventual pre-processing steps need to be chosen carefully depending on the problem setting. If the scores of the differentially expressed genes are expected to be distributed symmetrically around zero, a transformation such as squaring or the absolute value should be applied in order to prevent cancellation effects. Alternatively, methods like the (weighted) KS or maxmean statistics, which are not as sensitive to the symmetry of the scores, should be used.

Besides sensitivity to the symmetry of the input scores, the number of produced false positives can be an issue. To combat this, a more conservative significance threshold can be used. This choice is likely to have a larger impact on the number of false positives than the chosen set-level statistics.

If a clear cut-off for distinguishing differentially from normally expressed genes is known, the ORA method offers superior performance for the entity-based permutation scheme. However, as this is seldom the case, using other methods is recommended in practice. Given the choice between two equally performing methods, the one offering a superior runtime behaviour should be chosen. To this end, methods that offer an exact p-value computation method are in an advantageous position for the entity-based case. Examples are ORA, the unweighted KS statistics, the t-tests, and the Wilcoxon test. In the sample-based case, the performance of the entity-level statistics tend to dominate the computation time.

To close, we would like to reiterate that the analysis presented here should be taken with a grain of salt. First of all, not all parameters of the enrichment methods and the data generation procedure were tested exhaustively. This means that the performance of some methods may have been underestimated due to unfavourable parameter settings. The data creation process and especially the creation of significantly enriched categories can attribute unfair advantages to some of the tested methods. Here, especially the shift of differentially expressed genes to the significance threshold for $\alpha = 0.05$ can be criticised as too conservative. The reason for this is that, especially together with the random noise term governed by σ_ϵ, some gene expression values actually fall below the significance threshold. We argue that this is not an issue as the gene expression values are symmetrically distributed around the threshold. Due to this, the mean of the expression still, on average, coincides with the threshold. Also, the differentially expressed genes still should have a higher mean expression than the "null" genes. Hence, as no method used an explicit significance cut-off to detect differentially expressed genes, the exact magnitude of the shift should not be an issue. Nevertheless, by examining the distribution of the differentially expressed genes in experimental data, a more

ORA selected the upper and lower quantiles instead of imposing a fixed threshold.

realistic, albeit more complex, model could be derived. Another bias may stem from the fact that the significantly enriched genes are randomly chosen and considered independent from each other. This assumption closely resembles the urn model underlying the hypergeometric test used by ORA. However, for biological datasets, gene expression is most certainly not independent within a category. Therefore the sample-based p-value strategy, which considers the correlation structure within a category in fact is at a disadvantage in this evaluation. On the other hand the ORA method depends on the choice of the threshold that selects the differentially expressed genes. While in this case, the 95 % quantile is appropriate due to the data generation procedure, the true value of the threshold is usually not known *a priori*. Nevertheless, the chosen parameters reflect the settings a user of the GeneTrail2 server would use by default and, therefore, the presented evaluation gives some insight into possible real-world usage.

4.4 HOTELLING'S T^2-TEST

The sample-based permutation strategy (Section 4.2.1) accounts for a part of the entity interdependence.

We previously discussed enrichment procedures, like the Over Representation Analysis (ORA) [Dră+03] and the Gene Set Enrichment Analysis (GSEA) [Sub+05], that treat the entities contained in each category as independent. However, we previously assumed that a category consists of genes that are functionally related and thus are likely to be coregulated. Of course, this suggests that these entities are anything but independent.

Here, we examine the suitability of Hotelling's T^2-test as an enrichment algorithm. Hotelling's T^2-test is a global enrichment procedure (Figure 4.2) that accounts for the correlation structure between entities. We structure our presentation as follows: first we give a general introduction to Hotelling's T^2-test. Afterwards, we perform a brief evaluation to illustrate some of its properties.

4.4.1 *Mahalanobis Distance*

A concept that is central for understanding Hotelling's T^2-test [Hot31] is the *Mahalanobis distance* [Mah36]. To explain it, we need to recall the multivariate generalisation of the Gaussian distribution as introduced in Section 3.4.1. Again, consider the univariate Gaussian distribution

$$\mathcal{N}(x, \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{4.4}$$

Here, $\mu, \sigma^2 \in \mathbb{R}$ represent the mean and variance of the distribution, respectively. Disregarding the normalisation factor in front of the exponential function, the value of the distribution solely depends on the squared distance of x to the mean which is scaled inversely by the variance: $(x-\mu)^2/(2\sigma^2)$. In the multivariate Gaussian distribution

$$\mathcal{N}(\vec{x}, \vec{\mu}, \Sigma) = c \cdot \exp\left(-\frac{1}{2}(\vec{x}-\vec{\mu})^t \Sigma^{-1} (\vec{x}-\vec{\mu})\right) \tag{4.5}$$

this is replaced by the term $(\vec{x} - \vec{\mu})^t \Sigma^{-1} (\vec{x} - \vec{\mu})$. In other words, the distance of $\vec{x}$ from the centroid $\vec{\mu} \in \mathbb{R}^p$ is rescaled by the inverse covariance (or precision) matrix $\Sigma^{-1} \in \mathbb{R}^{p \times p}$. The constant

$$c := \det(\Sigma)^{-\frac{1}{2}} / (2\pi)^{\frac{p}{2}}$$

serves as the normalisation factor.

The covariance matrix and, thus, Σ^{-1} are positive definite matrices. Hence, Σ^{-1} defines a scalar product and a corresponding metric. This metric is called the Mahalanobis distance [Mah36].

4.4.2 *Hotelling's T^2-Statistics*

A common question in statistics is, whether two samples originate from the same or two different distributions (Section 4.1). The entity- and set-level statistics presented in earlier sections (cf. Section 4.2.2) answer this question for a range of univariate cases. However, if the samples follow a multivariate Gaussian distribution, Hotelling's T^2-statistics is the appropriate test statistics. It is a straightforward generalisation of Student's t-test [Stu08] (Section 4.2.2). Let N be the number of samples and n, m the size of each group, respectively. Using $\boldsymbol{S_x} := \sum_{i=1}^{n_x} (\vec{x_i} - \vec{\mu_x})(\vec{x_i} - \vec{\mu_x})^t$ and the analogously defined $\boldsymbol{S_y}$, we obtain the pooled covariance matrix $\boldsymbol{S}$:

$$\boldsymbol{S} = \frac{\boldsymbol{S_x} + \boldsymbol{S_y}}{N - 2} \tag{4.6}$$

This allows us to compute the T^2-statistics as the weighted difference of the sample means

$$T^2 = \frac{n_x n_y}{N} (\vec{\mu_x} - \vec{\mu_y})^t \boldsymbol{S}^{-1} (\vec{\mu_x} - \vec{\mu_y}) \tag{4.7}$$

In order to evaluate the significance of T^2, the appropriate quantiles can be computed from the F-distribution via the relationship

$$\frac{N - p - 1}{(N - 2)p} T^2 \sim F(p, N - 1 - p) \tag{4.8}$$

The probability density function of the F-distribution is given as:

$$f(x, d_1, d_2) = \frac{\sqrt{\frac{(d_1 x)^{d_1} d_2^{d_2}}{(d_1 x + d_2)^{d_1 + d_2}}}}{x B\left(\frac{d_1}{2}, \frac{d_2}{2}\right)} \tag{4.9}$$

where $d_1, d_2 \in \mathbb{R}^+$ and

$$B(x, y) = \int_0^1 t^{x-1} (1 - t)^{y-1} \, dt \tag{4.10}$$

is the *beta-function*. As the parameters of the F-distribution must be positive, computing an exact p-value requires more samples than observed genes ($N \geq p + 2$). Thus, it is infeasible to test for a difference

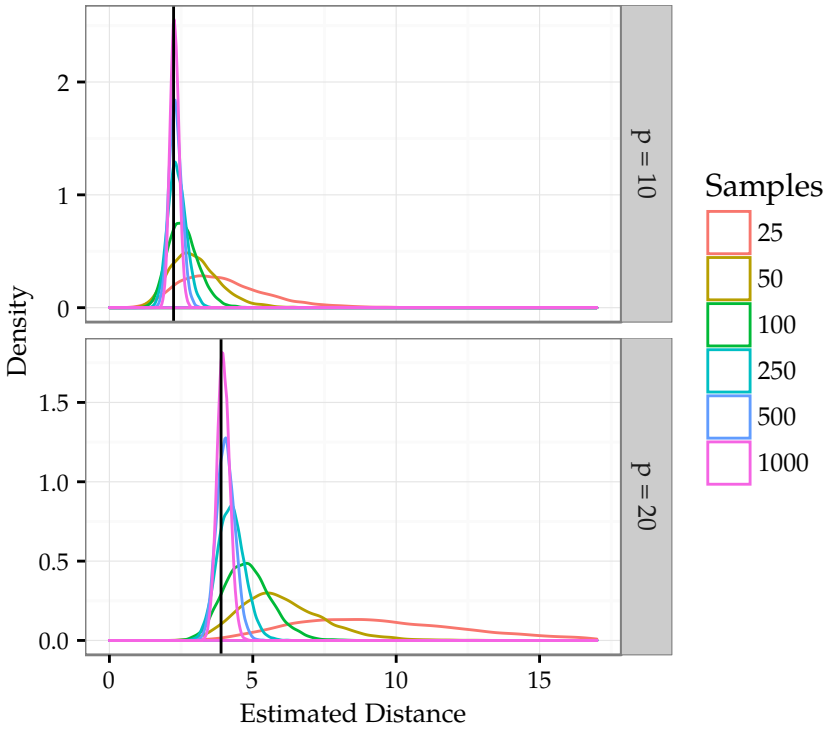


Figure 4.11: Density plots of the estimated Mahalanobis distance between two multivariate Gaussian distributions of dimensionality $p \in \{10, 20\}$ computed for increasing sample counts. The sample count in the legend reflects the number of samples for *each* distribution. The vertical line represents the true Mahalanobis distance. More than 100 samples per group are required to achieve reliable distance estimates. Especially for $p = 20$ low sample counts lead to a large variability in the estimated distance.

between a large number of genes, e.g. as obtained from a microarray, where the number of available samples is much smaller than the number of genes. However, computing T^2 and according p-values for small gene sets ($p \approx 50$) is unproblematic for many microarray studies that are available in public repositories like the NCBI Gene Expression Omnibus [Bar+13; EDL02].

In practice, the Hotelling T^2-test comes with serious usability problems. The reason for this lies in the required inversion of $\boldsymbol{S}$ which can amplify noise in the input data as explained in Section 3.4.3. Furthermore, Hotelling's T^2 has been shown to possess a low statistical power [BS96]. Chen et al. [Che+12] report that regularised versions of the test show a considerably improved performance. To this end, they compute p-values based on an empirical distribution determined via resampling. In addition they propose using bootstrapping to increase the power for high-dimensional datasets.

4.4.3 *Evaluation*

To ensure a certain power level for a statistical test, an appropriate number of samples needs to be selected (Section 4.1.1). The fact that estimating the covariance matrix requires to effectively determine $N(N-1)/2$ parameters suggests that Hotelling's T^2 statistics relies on a considerable amount of samples [BS96]. To illustrate this, we created two random, multivariate Gaussian distributions of dimensionality $p \in \{10, 20\}$. For each of the distributions, a fixed number of samples was drawn and used to estimate the pooled covariance matrix as well as distribution means. Next, the Mahalanobis distance between the two sample groups was computed. Each computation was repeated 10,000 times and density plots were generated. In Figure 4.11 it can be seen that the distance estimates vary considerably for small sample sizes. Only for samples sizes > 100 for each group (and thus > 200 samples in total) the estimates become more reliable. Note that this effect becomes more pronounced as the number of dimensions grows and thus a further increase in the number of samples is required. This makes it infeasible to directly apply Hotelling's T^2-test in an enrichment scenario, as from experience the number of available samples in biological studies rarely is larger than 100.

To give an example for the behaviour of Hotelling's T^2-test in a real world application, we computed enrichments for categories derived from KEGG (cf. Section 5.3) using the Wilm's tumour data (Section 2.4) as input. To this end, we implemented a version of the test statistics using ridge regularisation (cf. Section 3.4.3, Tikhonov [Tik63], or Ledoit and Wolf [LW04]). Empirical p-values were computed using a sample-based permutation strategy as suggested by Chen et al. [Che+12]. To determine the effect of the ridge parameter r the settings $r = \{0, 0.001, 0.005, 0.01, 0.05, 0.1\}$ were used.

In case of no regularisation, p-values could only be computed for categories with less than 30 members due to the low number of input samples. The most significant categories were computed for $r = 0.001$ with 13 categories being significant at FDR ≤ 0.2 (Table 4.2). Other settings of the ridge parameter yield fewer significant categories. Hence, in contrast to local enrichment methods that yield in the order of 100 significant categories on the same dataset (Section 5.3.5), the sensitivity of the T^2-test is comparatively low.

4.4.4 *Summary*

We described and evaluated the application of Hotelling's T^2-test as a global enrichment method. Unfortunately, applying this test directly requires rigorous regularisation, as the number of degrees of freedom quickly overtakes the number of available observations. To further increase the robustness of the method, Ackermann and Strimmer [AS09] argue that the estimator for the covariance matrix should be replaced with their more efficient shrinkage estimator [Str08]. Nevertheless, the

Category	p	.001	.005	.01	.05	.1
Chemical carcinogenesis	66	0.014	0.078	0.071	0.071	0.078
Protein digestion and absorption	86	0.014	0.056	0.028	0.028	0.028
Drug metabolism - cytochrome P450	57	0.047	0.226	0.234	0.282	0.338
Cytokine-cytokine receptor interaction	256	0.085	0.090	0.090	0.096	0.096
Dilated cardiomyopathy	90	0.102	0.423	0.432	0.502	0.548
Rheumatoid arthritis	86	0.122	0.078	0.071	0.071	0.078
Prion diseases	36	0.141	0.384	0.420	0.569	0.687
Inflammatory bowel disease (IBD)	65	0.169	0.423	0.432	0.525	0.567
Basal cell carcinoma	55	0.169	0.527	0.582	0.691	0.687
Leishmaniasis	70	0.186	0.423	0.432	0.423	0.426
Pathways in cancer	326	0.2	0.503	0.511	0.525	0.555
Amoebiasis	108	0.2	0.078	0.071	0.071	0.078
Retinol metabolism	53	0.2	0.735	0.722	0.620	0.562

Table 4.2: Significant categories at FDR ≤ 0.2 detected by Hotelling's T^2-test using ridge regularisation. Names of the categories as well as the number of contained genes **p** are given in the first two columns. Subsequent columns contain Benjamini-Hochberg adjusted p-values for various settings of the ridge parameter. In the case of no ridge parameter no categories were significant and no p-values could be computed for the remaining categories. Hence, results for $r = 0$ have been excluded.

comparatively low power of the T^2-test results in many categories being wrongly classified as not enriched, whereas they are detected by "classical" enrichment algorithms.

As alternatives to the T^2-test, various competing approaches have been conceived. An example, is the *SAM-GS* procedure [Din+07] which uses the *SAM* statistics [TTC01] for scoring gene sets. The SAM statistics, in turn, is based on the approximation of Hotelling's T^2-statistics by Dempster [Dem58]. The *global test* algorithm by Goeman et al. [Goe+04] also considers the covariance matrix, but avoids the computation of its inverse. Kong, Pu, and Park [KPP06] propose a dimensionality reduction method based on *principal component analysis (PCA)*. Lu et al. [Lu+05] avoid the construction of the complete covariance matrix by using a search algorithm that maximises the T^2 distance between sample and reference group. Methods like Gene Graph Enrichment Analysis (GGEA) [Gei+11] or EnrichNet [Gla+12] account for the dependencies

between entities by explicitly including connectivity information obtained from known, biological networks.

5 GRAVITON

All problems in computer science can be solved by another level of indirection. But that usually will create another problem.

— DAVID WHEELER, ACCORDING TO beautiful code (2007)

An important part of bioinformatics is the development of new methods for the analysis of experimental data. Due to this, a large toolbox of specialised software is available to the end user. In many areas of bioinformatics, such as DNA sequencing, the state-of-the-art is constantly evolving. The same is true for biology, where the size of public databases that organise the available knowledge is constantly growing. Keeping up with this change requires a considerable amount of time and sophistication on the user-side. For researchers writing software, the maintenance burden of ensuring that their code can be deployed on any available platform takes time away from developing new, improved methods. A way to avoid this burden is to offer tools as web services, which comes with the advantage that only a single, centralised installation of the required third-party software and databases needs to be maintained. Additionally, web based user interfaces work on every computer with a browser and, if designed properly, can be easily controlled via client-side scripts. Users of a service no longer need to update their software regularly and do not need to track databases for updates.

Still, despite the fact that maintaining a web server requires far less effort than maintaining a native application, useful bioinformatics web services that integrate well into existing workflows, are challenging to construct. As with traditional software packages a considerable amount of code for parsing file formats, handling malformed user input, and performing statistical analyses needs to be implemented. In addition, business logic for handling user data and sessions or controlling database access is required. Furthermore, a functional and appealing user interface needs to be designed. Especially this latter part is often not necessary when rolling out a package for a scripting language or a native command line application.

In order to avoid reimplementing common functionality for every web service, we built the *Graviton* framework. To give a comprehensive description of the framework, we first need to familiarise ourselves with the available technologies for implementing a web service. Next, we outline the requirements that Graviton should fulfil. We then give a general overview of the architecture of Graviton followed by a more in-depth discussion of the individual components GeneTrail2, NetworkTrail, and DrugTargetInspector.

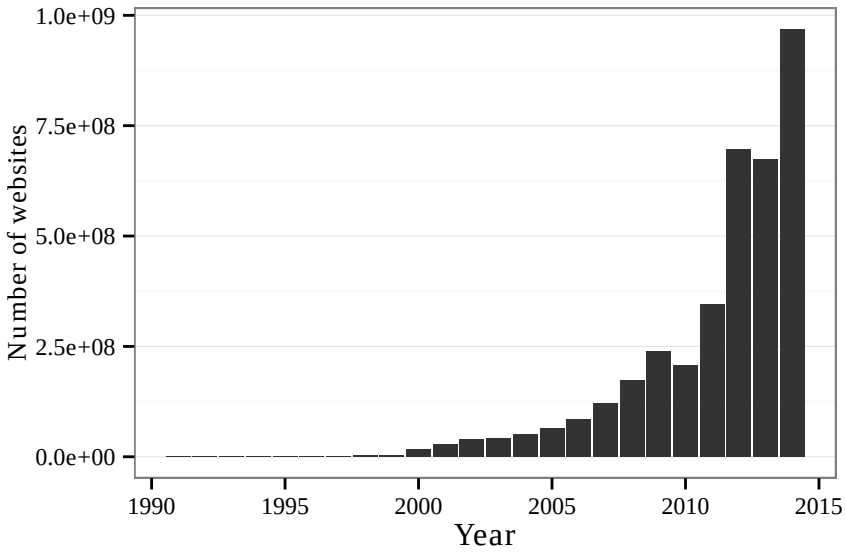


Figure 5.1: Estimated number of available websites starting from the first available site. The WWW grows nearly exponentially. Drops in the statistics are due to changes in the methods used for counting live websites (*Total number of Websites* [16]; Date of retrieval 12.04.2016).

5.1 THE WORLD WIDE WEB

Berners-Lee et al. [Ber+04], commonly referred to as the "father" of the *World Wide Web* (*WWW*, or simply *web*), describes this vital communication infrastructure as follows: "The World Wide Web is an information space in which the items of interest, referred to as resources, are identified by global identifiers called Uniform Resource Identifiers (URI). " The web is accessible via the Internet TCP/IP infrastructure and is open to everyone with a valid IP address. This openness is reflected by the fact that, since its inception, the web has grown exponentially and, at the time of writing, consists of almost 1 billion websites [16] (cf. Figure 5.1).

The WWW is built upon a number of standardised technologies such as HTTP, HTML, CSS, and JavaScript. HTTP serves as the transport mechanism via which documents are transferred to the user. These documents are typically, but not necessarily, HTML documents (cf. Section 5.1.2) containing arbitrary, structured data. Their appearance can be controlled via the CSS stylesheet language (see Section 5.1.3). Dynamic documents that react to user input or retrieve additional data from the web on demand, can be realised using JavaScript (see Section 5.1.5). In the following, we give short introductions into these and other web technologies, which will be needed once we discuss the implementation of the Graviton framework. We start with an introduction to HTTP.

5.1.1 *The Hypertext Transfer Protocol*

The web is centred around sending documents from server to client (and to a lesser extend from client to server, too). The Hypertext Transport Protocol (HTTP) provides the facilities with which server and client negotiate e.g. the requested document, the used encoding, and the document format. Before discussing the protocol in detail, we introduce the terminology and philosophy behind the design of HTTP.

The REST Principle

A way to view the web is as an information space. Each piece of information, a so-called *resource*, is identified by a globally unique *uniform resource identifier (URI)* (see Section 5.1.4). Each HTTP URI encodes a host that stores the requested resource, as well as a path on the host to the location of the resource. HTTP provides the communication channel with which a client and a server (the host) exchange information about the *state* of a resource. The simplest example of such an exchange is the client issuing a *request* for obtaining a resource from the server. The server then sends a *response* containing a *representation* of the requested resource to the client. It is important to note that a resource can never be sent directly as it represents an abstract piece of information that exists independently from any particular data format. Thus, an encoding step that transforms a resource into a representation is required before the data can be made available. Other possible mutations of the state of a resource, such as updating or deleting it, are, of course, possible.

HTTP and other web technologies were designed pragmatically. Further refinement and formalisation of the underlying principles happened post hoc.

Fielding and Taylor [FT02] formalised the above concepts under the term *representational state transfer (REST)*. In Section 5.1.8 we will revisit REST and explain how it can serve as a guideline for implementing web *application programming interfaces (APIs)*.

Protocol

HTTP is a stateless, text-based protocol for document retrieval. It was first defined in the *Internet Engineering Taskforce (IETF) request for comments (RFC)* 1945 [BFF96] and has been updated multiple times since then. In 2015, a successor to the first protocol version, HTTP 1.1, was published under the name HTTP/2. Here, we present the protocol in terms of HTTP 1.1, as it is in many regards simpler than its successor. Nevertheless, the introduced concepts apply to both versions.

For each resource that a client wants to obtain or modify, a HTTP *request* message is issued to the server. The server answers each request by sending a *response* containing the requested data. How the data is represented (encoded) is negotiated by client and server via the use of special *header* fields.

Requests A request starts with a *request method*, specifying how the request should be interpreted, followed by a path to a resource and the protocol version. Several request methods are available. The ones most

commonly used are `GET`, `POST`, `PUT`, and `DELETE`. The methods *should* be interpreted as follows:

`GET` Retrieve a document from the server.

`POST` Upload a new document to the server, where it exists below the hierarchy of the request URI.

`PUT` Upload a document to the location on the server identified by the request URI or update an already existing resource.

`DELETE` Delete the specified document from the server.

In the following lines, a set of *header* fields can be specified. Each field consists of a field name and a value separated by a colon. Which values are admissible, is specific for the respective field. HTTP comes with a set of predefined header fields. Additional fields are usually standardised in their own RFC. The end of the header is indicated by an empty line (carriage return, line feed). As HTTP is a text-based protocol, simple requests can be created using a terminal software such as `telnet`. A sample request is shown in Listing 5.1.

Example HTTP request header fields

Host specifies the authority component of the request URI.

Accept lists the *media types* the client understands together with a priority expressing the clients preferences. A list of admissible *media types* is maintained by the *Internet Assigned Numbers Authority (IANA)* as specified in RFC6838.

Accept-Language lists the preferred languages of the client, with an associated preference measure.

Accept-Encoding lists the encodings the client is able to handle. This can be used by the server to compress the data stream for reducing the bandwidth consumption.

Content-Type states which encoding has been used for the (optional) data that should be transferred to the server. As for the **Accept** header, media types are used.

In the case of `POST` and `PUT` requests, an additional payload can be transferred to the server by appending it after the header. This payload is called the *message body*. The format of the contained data is not specified in the protocol and must be communicated to the server using the `Content-Type` header.

Responses In order to answer a client request, the server sends a response that consists of a status code, header data, and, if the request was successful, the contents of the requested resource. A sample response is shown in Listing 5.2.

```
POST  /api/job/setup/  HTTP/1.1
Host: genetrail2.bioinf.uni-sb.de
User-Agent: Mozilla/5.0 (X11; Fedora; Linux x86_64;
    rv:40.0) Gecko/20100101 Firefox/40.0
Accept: text/html,application/xhtml+xml,
    application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-GB,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive

name=gsea
```

Listing 5.1: Example HTTP POST request. The first line contains the HTTP request method (red), the path of the requested resource (blue), and the protocol version (green). The following lines contain the request header. Each header is composed of a field name (orange) and a value (black). The optional request body (grey) starts after an empty line and contains arbitrarily encoded data for transfer to the server.

Properties From the description of the protocol we can make two observations that have important implications on how web services can be implemented. First, the HTTP protocol *itself* is stateless. This means, the protocol does not keep track of previous requests by the client or responses by the server. For web applications that rely on state, such as user logins, being tracked, custom state handling must be implemented on top of HTTP. Another implication of statelessness is that subsequent `GET` request for the same resource should always receive the same response, as `GET` leaves all resources unchanged. Accordingly, `GET` responses can be cached by the browser to avoid unnecessary round-trip times. In addition, caching can greatly reduce bandwidth consumption, which is especially important on bandwidth constraint connections such as mobile devices. For web application developers, this implies that to achieve optimal performance the amount of changes to existing resources should be minimised such that caching can take effect.

Second, due to the ordering between requests and responses, it is not possible to send information from server to client without the client sending a request first. This implies that the client must periodically query for the status of a long running computation on the server side in order to be notified about its progress.

WebSockets alleviate this to some degree.

5.1.2 *The Hypertext Markup Language*

The next core technology of the web is the Hypertext Markup Language (HTML) [HH16]. HTML allows to represent documents as a semantically annotated hierarchical structure. The hierarchy is created by nesting so-called *elements*. An element usually consists of a start (`<p>`) and end (`</p>`) *tag* that delimit a sequence of text and other elements. Each element can furthermore be supplied with one or more at-

HTML was a Standard Generalized Markup Language (SGML) (ISO 8879:1986) based specification. However, recent versions have forgone all SGML compatibility.

```
HTTP/1.1 200 OK
Cache-Control: public
Content-Length: 36
Content-Type: text/plain
Last-Modified: Sun, 13 Sep 2015 09:46:05 GMT
Date: Fri, 16 Oct 2015 19:29:07 GMT

This is the content of the response!
```

Listing 5.2: Example HTTP response. The first line contains the protocol version (green) and a status code (blue). Subsequent lines contain response headers. Each header is composed of a field name (orange) and a value (black). The end of the header and start of the response body (gray), which contains the requested document, is indicated by an empty line.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Hello world!</title>
</head>
<body>
  <h1>Hello world!</h1>
  This is an example HTML page.
  <p>And this is a new paragraph.</p>
</body>
</html>
```

Listing 5.3: Example HTML5 document. The type of the document is controlled via the `DOCTYPE` statement. Metainformation is encoded in the `<head>` tag, whereas user visible information is encoded in the `<body>` tag.

tributes (`<input type="text">`) that determine element specific properties. While HTML offers tags like `<i>` (italic), `<b>` (bold), `<br>` (line break), and `<font>` (font properties) that directly control the appearance of the HTML in a browser, the use of these tags is deprecated in modern documents. Instead, the used annotations should express the semantics of their content. For example, HTML offers tags to indicate that some text should be interpreted as an address (`<address>`), section (`<section>`), paragraph (`<p>`), dates and times (`<time>`), or heading (`<h1>`). One annotation that deserves special attention is the *anchor tag* (`<a>`). It allows the creation of *hyperlinks*, which are cross-document and even cross-site references that can point to other resources. The importance of these links cannot be overstated, as they make it possible to add additional information to a document by referencing other, supplementary or explanatory documents. This allows the creation of discoverable document systems such as Wikipedia, that, since its inception in 2001, has rendered most printed encyclopediæ obsolete.

Hyperlinks add the "Web" to World Wide Web.

HTML is available in two flavours: the traditional, SGML inspired syntax (`text/html`) and a more recent, XML [W3C06] compliant syntax (`application/xhtml+xml`). While the first version has a few syn-

XML is, in fact, a simplified version of SGML

tactic irregularities, such as tags lacking end tags or attributes without values, it is often preferred by web developers due to its relative conciseness when compared to the XML variant. The latter, however, can be processed using standard XML parsers and thus is better suited for automatic processing.

The much-hyped fifth version of HTML adds additional semantic tags to the language that allow the creation of documents that are more friendly towards algorithms such as screen readers. Examples are tags like `<figure>`, `<summary>`, and `<header>`. In addition, tags for embedding audio (`<audio>`) and video (`<video>`) into sites have been introduced. Moreover, owing to the desire of web developers to offer less static user experiences, the interoperability with JavaScript has greatly been improved.

5.1.3 *Cascading Style Sheets*

Although it should be avoided, HTML also allows non-semantic markup.

An important task of a web browser is to generate a visual representation of an HTML document. As HTML is meant to only provide semantic annotation, the document itself possesses no explicit information on how to visualise its content. This means that the final rendering is completely up to the browser and cannot be controlled by the user. While this guarantees a uniform appearance of the documents, the creation of visually appealing sites requires more direct control over their styling. This can be achieved via the *cascading stylesheet (CSS)* language [Not15]. CSS allows to specify visual properties for HTML elements by specifying a set of rules. Each rule consists of a selector and a properties part. Rules can either match to a certain element type, a set of elements annotated with a given *class*, or a single element with a given *id*. Classes and ids can be assigned to elements via accordingly named attributes. Once a rule matches an element, the properties specified in the rule are applied to it. If multiple rules match an element all of their properties are applied. In the case of conflicts, the most specific rule takes precedence. Using CSS allows to decouple the semantics of the document specified in the HTML file from its representation as determined by the browser [KN09]. This allows to change the appearance of a document by simply exchanging a CSS file. Furthermore, it ensures that documents are interpretable without the styling information which is important for implementing accessibility features such as enlarged font sizes or screen readers.

Accessibility refers to the ability of handicapped users to access the information of the document.

5.1.4 *Uniform Resource Identifier*

The anchor tag and its ability to refer to arbitrary documents is a central pillar of HTML and the architecture of the web. Unsurprisingly, hyperlinks are ubiquitous in web-based applications. The anatomy of a link, meaning how a valid link needs to be formatted, is specified in RFC3986 [BFM05]. Similarly to the HTTP protocol, which is not tied to one document type, links are not required to refer to other web pages.

Figure 5.2: Simplified HTTP URL. Each URL consists of a scheme and scheme-specific part. For HTTP the scheme-specific part contains the host (authority), the path to the resource on the host, additional query information, and a fragment string.

Such a generalised link is called an *Uniform Resource Identifier (URI)*. As the name says, each URI identifies a resource. To this end a *namespace* or *scheme* is given that specifies the domain in which the resource is valid. In the case of HTTP and other network protocols the used URI are, in fact, *Uniform Resource Locators (URLs)*, a more constrained subset of URIs. Each HTTP URL consists of an authority section containing the *host* name of the server offering the document and optionally authentication and port information. This is followed by the *path* that identifies the resource. Optional parameters used for refining the query can be supplied in the query string. For example consider a website offering scientific publications. Usually the full text of a publication is shown if its site is accessed. If the query string `?display=abstract` is part of the URL, only the publication's abstract could be shown instead. Finally, an optional fragment string, which is often used for indicating a location in the document, can be appended. An example URL is shown in Figure 5.2.

5.1.5 *JavaScript*

HTML pages display static information. Whilst this is sufficient for displaying plain documents, web applications that react to user input are tedious and inefficient to implement using this model, as every reaction to user input requires to reload the whole application.

Weakly typed means that `1.0 + "1"` is a valid expression.

JavaScript, officially called ECMAScript, is a scripting language that can be used to create *dynamic* HTML documents [15]. JavaScript code can be added to HTML documents via the `<script>` tag and is interpreted by the browser. The language itself is weakly and dynamically typed and allows to mix functional, imperative, and object oriented programming styles.

Once the script has been loaded by the browser, it can arbitrarily manipulate the elements contained in the document. This access is provided via the *document object model (DOM)* [W3C16] tree. The DOM tree represents the element hierarchy as encoded by the HTML tags. Elements expose events that can be used to trigger JavaScript code on user input. For example an element representing a button exposes the `onclick` event that fires when the button is clicked.

```
<record>
  <person>
    <name>Max</name>
    <surname>Mustermann</surname>
    <age>35</age>
  </person>
  <grades>
    <grade>1.0</grade>
    <grade>2.3</grade>
    <grade>5.0</grade>
  </grades>
  <registered>false</registered>
</record>
```

Listing 5.4: XML representation of the data of a student. Field names are stated twice, in the start and end tag. Arrays are represented by nesting tags.

5.1.6 *Asynchronous JavaScript and XML*

In traditional web applications, a full reload of the application is required once a user input requires new data to be transferred from the server. *Asynchronous JavaScript and XML (AJAX)* is a technique that allows JavaScript web-applications to request additional data from the server without reloading the current page. It uses regular HTTP requests and thus supports, despite the XML part of the name, the exchange of arbitrary data. By avoiding unnecessary full reloads, web applications employing AJAX behave more like their native counterparts. As a side effect, using AJAX significantly reduces the amount of data that needs to be transferred over the network.

5.1.7 *JavaScript Object Notation*

Especially for web applications that make use of AJAX, it is useful to be able to transfer structured data. A common choice for this are XML formats that can be used to represent arbitrary data structures and databases (cf. Listing 5.4). The disadvantage of using XML is that the format is comparatively verbose, difficult to read, and slow to parse (cf. [LS00; NJ03]).

A much simpler, yet flexible alternative is the *JavaScript Object Notation (JSON)* format [Bra14]. As the name suggests, JSON derives from the JavaScript syntax for defining objects. A valid JSON message can represent the basic types *number*, *string*, *bool*, *null*. The types *object* and *array* can be composed of the basic types as well as other objects and arrays. An example can be seen in Listing 5.5. Due to its lightweight syntax, JSON has become the "Lingua Franca" of the web and is supported well by browsers as well as by server-side libraries.

```
{
  "person": {
    "name": "Max",
    "surname": "Mustermann",
    "age": 35
  }
  "grades": [1.0, 2.3, 5.0],
  "registered": false
}
```

Listing 5.5: JSON representation of the data of a student. Little information is duplicated, making it easier to read and modify the data.

5.1.8 *RESTful APIs*

With the emergence of mobile devices, it has become increasingly more important to be able to run computationally expensive operations on compute servers due to power constraints. In the field of bioinformatics, however, dedicated web applications, which are only usable via a web page, suffer from the fact that they are difficult to integrate into existing workflows. These are often based on scripting languages such as Python [VD09] and R [R C16] or workflow systems such as Galaxy [Goe+10] and Taverna [Wol+13]. To enable these tools to connect to a web service, it is advantageous to structure web applications as dedicated back and front ends. In this architecture, a pure web service, which only offers an API, serves as back end. On top of this, specialised front ends, such as a web page, can be created.

Multiple ways to realise the back end API are imaginable. However, in recent years the observation that HTTP natively covers many needs of such APIs, has lead to a set of guidelines on how web services should be designed. These, so called *RESTful* APIs focus on the manipulation of resources, as defined by the REST principle, instead of providing arbitrary sets of operations. Here, the basic observation is that most web services provide a thin layer of business logic around a set of resources, which are often stored in a database server. Typical database servers provide the basic operations *Create*, *Read*, *Update*, and *Delete (CRUD)*. In RESTful APIs, these operations are, per convention, mapped to the HTTP verbs `POST`, `GET`, `PUT`, and `DELETE`, respectively. Additionally, HTTP features such as content type negotiation, status codes, or browser caching are used. These properties result in comparatively easy to use APIs that can be seamlessly integrated into existing web applications. To see further advantages of RESTful API design, we first must take a look at the disadvantages of competing techniques.

A large variety of protocols for performing so called *remote procedure calls (RPC)* has been designed in the past. Examples are Sun-RPC, Corba, or DCOM that require specialised data encodings and transport protocols. In practice, this means that dedicated library and networking support is necessary for using these technologies. Especially the networking support can severely limit the reliability of the service, as

highly controlled environments like corporate networks often limit firewall traversal to a few, well-known protocols such as HTTP. Other, popular RPC schemes like *XML-RPC* [MAL06] and its successor *Simple Object Access protocol (SOAP)* [Gud+03] thus support HTTP as a transport protocol. Both methods send standardised, XML encoded messages. However, as SOAP supports multiple other transport mechanisms besides HTTP, it is not able to rely on the facilities already provided by HTTP. These features need to be duplicated in the SOAP protocol leading to unwanted redundancy. Furthermore, the mandatory XML encoding leads to large message size which can be problematic on unreliable, bandwidth constrained connections such as mobile networks. XML-RPC suffers from a similar constraint, as only one content type, namely XML, is supported.

RESTful APIs avoid all these problems. Thanks to the ubiquity of HTTP, basically every platform also supports RESTful APIs without needing to install special software. The transferred data type is, by convention, JSON or XML but can be changed to a different, more appropriate format at any time using content type negotiation.

Other RPC schemes remain relevant where special requirements, e.g. on latency and efficiency must be met.

5.2 THE GRAVITON ARCHITECTURE

CONTRIBUTIONS The initial code for the Graviton platform was written by Oliver Müller and me. Large parts of this code base have since then been rewritten according to a new design devised by me. This revised implementation and the underlying C++ library were created by Tim Kehl and me.

In the previous section we discussed the available technologies for implementing web applications and services. In this section, we introduce the Graviton platform for implementing bioinformatics web services. We start by stating the requirements that influenced the design of the platform. Next, a general overview over the Graviton architecture is provided. Afterwards, we take a closer look at some core implementation details.

5.2.1 *Requirements*

Computational biology is a multi-disciplinary field with researchers stemming from varying backgrounds, such as biology, mathematics, physics, chemistry, medicine, and computer science. This heterogeneous group of potential users makes developing bioinformatics applications a challenging task, as a wide range of user experiences and usage scenarios need to be covered. On the one hand, a simple and approachable interface, which is applicable in the most common use-cases, must be provided to be usable by researchers less versed in computer science and statistics. On the other hand, the tool needs to offer

enough flexibility to cover non-standard usage scenarios and should allow bioinformaticians to integrate it into their existing pipelines. Naturally, these requirements are difficult to fulfil simultaneously. Whilst the ideal user interface for a bioinformatician consists of an API or a sophisticated GUI allowing to tweak all parameters, a biologist requires an interface with few, essential or no parameter settings at all. Similarly, the presentation of the results needs to be clean and should contain as little elements as possible to allow the user to focus on relevant pieces of information. To accommodate both needs, Graviton provides a RESTful API on top of which specialised user interfaces can be created. While expert users can choose to directly access this programming interface to integrate the offered facilities into their workflows, non-experts may choose the web-based GUI. In case of the GeneTrail2 server (Section 5.3), a simplified and an advanced web interface are provided.

Another requirement that must be met by a research tool is to produce reproducible results. At the absolute minimum this entails that all used input data as well as every parameter setting is recorded. For input data not provided by the user, such as identifier mapping tables or biological pathways, detailed provenance information containing origin and date of retrieval, must be made accessible. In some cases, lossy transformations must be applied to user data. The prime example for this is identifier mapping (Section 5.2.3) where an input identifier can be mapped to multiple targets or not at all. To ensure reproducibility and minimise unpleasant surprises such transformations should keep detailed logs that allow to audit the flow of information.

5.2.2 *Design*

Graviton is a framework for building fully integrated bioinformatics webservices. To this end, it provides implementations of basic functionality such as identifier mapping, file parsers, database access, job scheduling and workflow organisation. This functionality is organised in layers that each provide abstractions over the layers below them. The following layers can be distinguished:

1. Plumbing layer
2. `Resource` layer
3. API layer
4. Front end layer

In the lowest layer, the *plumbing layer*, implementations of largely independent, basic functionality is provided. The `Resource` *layer* ties these blocks together by means of the `Resource` and `Job` abstractions. Based on these concepts, the *API layer* presents a RESTful API to the client. On top of this API, a front end can be developed. We will now discuss the individual layers in more detail.

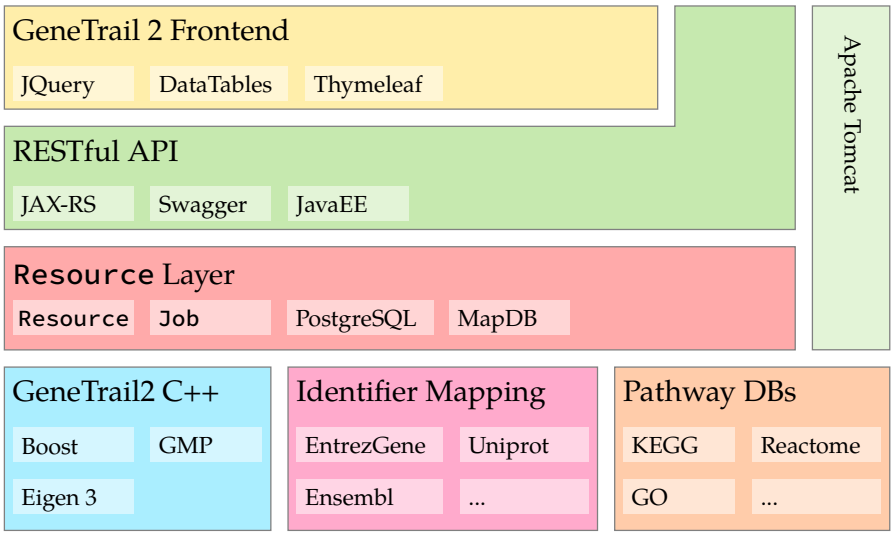


Figure 5.3: The GeneTrail2 architecture. Core algorithms are implemented in an optimised C++ library based on Boost, Eigen 3, and GMP. On top of this library we implemented a JAX-RS based RESTful API. The frontend is based on the Thymeleaf template engine and JQuery. As application server we use Apache Tomcat. Figure adapted from Stöckel et al. [Stö+16]

Plumbing Layer

The most common tasks in bioinformatics are the parsing of files and the conversion between database identifiers. As a web service needs to support a wide range of user-supplied input data, solid support for both tasks must be available in Graviton.

At the time of writing, Graviton supports parsers for identifier lists, score lists, matrices, the *variant call format (VCF)*, the *gene matrix transposed (GMT)* format, the *browser extensible data (BED)* format, and the GEO [Bar+13] GSE and GDS formats. Besides reading the content of a file, the parsers are also responsible for determining the format of user supplied data and validating its contents. More information about the supported file types can be found in Section 5.3. For mapping identifiers, a flexible system, which can use mapping information from a wide range of sources (Section 5.2.3), has been implemented. Furthermore, the plumbing layer offers implementations of statistical methods such as hypothesis tests (Section 4.1) and enrichment procedures (Chapter 4).

Resource Layer

The Resource layer introduces a first level of abstraction from the services offered by the plumbing layer. This is achieved by offering the interfaces `Resource` and `Job` that provide abstractions over data and analyses. A `Session` represents a collection of `Resources` and `Jobs` that were created by the same user. Both anonymous as well as authenticated users are supported. For representing the latter the type `User` is employed.

Graviton `Resources` are not HTTP resources. To avoid confusion we will capitalise the Graviton version.

A Resource is a representation of data stored on the file system together with assorted metadata. Every Resource is associated with a unique identifier through which it can be accessed from the Graviton API. Besides the path to the location of the represented file on disk, a resource stores the used identifier type and organism. For documentation purposes, every Resource also tracks the date of its creation as well as the time its metadata was last modified. A Resource belongs to exactly one user session. New Resources can be created by uploading data or by executing an analysis. Once created, the Resource is immutable in the sense that the file to which the Resource refers cannot be changed. However, it is possible to add further metadata and use it as input for an arbitrary number of Jobs. To model different kinds of data, Graviton comes with several, predefined Resource subtypes that place restrictions on the data they represent. Currently available subclasses are *Scores*, *Identifiers*, *ExpressionMatrix*, *Category*, *Enrichment*, *Subgraph*, and *Variations*.

Properties of a Graviton Resource

id A unique id, through which the Resource can be accessed.

creationDate The time a Resource was created.

modificationDate The last time the metadata of a Resource has been changed.

session The Session this Resource is a part of.

organism The organism from which the data represented by the Resource has been derived.

identifierType The database identifier used in the represented file.

comment A free text field that can carry any user supplied information.

shared Boolean flag indicating that the file represented by this Resource is shared between multiple sessions. Each user still owns a private copy of the Resources metadata. This mechanism allows to substantially reduce the on-disk memory consumption of Graviton.

intermediate Boolean flag indicating that the Resource is an intermediate result of an analysis or workflow. This allows frontends to hide such Resources unless otherwise requested.

normalized Boolean flag indicating that the identifiers used in the file are guaranteed to conform to Gravitons internal identifier database.

displayName A user visible name that should be displayed to identify the Resource. This name does not necessarily need to be unique.

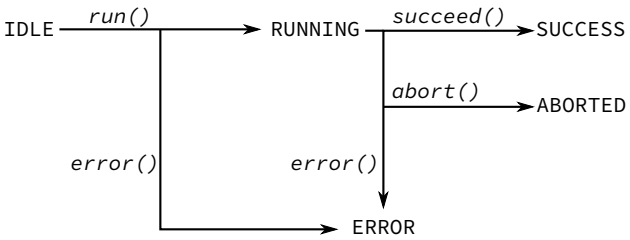


Figure 5.4: `Jobs` manage their internal state via a finite state machine. Each successfully created `Job` starts in the `IDLE` state. Once started, it moves to the `RUNNING` state. If the `Job` completes successfully or is aborted by the user, the state changes to `SUCCESS` or `ABORTED`, respectively. In each step runtime errors can occur. If this is the case the state moves to the `ERROR` state.

Properties of a Graviton `Resource` (cont.)

metadata A key-value store of arbitrary metadata that is not covered by the fields above.

Whereas a `Resource` represents data, a `Job` describes any process that creates or transforms data. Every `Resource` is created by exactly one `Job` and every `Job` that terminated successfully creates at least one `Resource`. The input of a `Job` are a set of `Job`-specific parameters and optionally one or more `Resources`. As with the `Resource` type, a `Job` has a unique identifier and tracks the time of its creation as well as a modification date. Besides the obvious purpose of documenting which analyses were run by the user and which input data was used to create its results, a `Job` serves a second important purpose in Graviton. As bioinformatics analyses can run for a considerable amount of time it is not feasible to run them inside the web server process. `Jobs` offer a simple abstraction for managing long running tasks that can optionally be executed out of process. To facilitate the creation of such `Jobs`, Graviton offers the subclass `AsyncJob`. The status of a `Job` is tracked via the finite state machine depicted in Figure 5.4.

An additional incentive for running out-of-process are the memory limitations of the Java Virtual Machine.

New kinds of `Jobs` can be added to Graviton by implementing the Java `Job` interface. Essentially, this amounts to writing code for validating input parameters, preparing the generated results and starting the actual computation. The `Job` and its input parameters are recognised by the Graviton system via custom Java annotations. Constructors for detected `Job` classes are automatically inserted into a factory object and are made accessible through the RESTful API. Furthermore, the annotations contain all necessary documentation for the `Job` and its parameters. This information is also available through the API which ensures that the documentation and implementation always stay synchronised.

Properties of a Graviton `Job`

id An unique id, through which the `Job` can be accessed.

creationDate The time the `Job` was created.

modificationDate The last time the `Job` was modified.

name The type of Job represented by this object.

displayName An user visible name that should be displayed in the front end.

command The command line that was run when the `Job` was executed.

parameters A key-value store of parameters that are required for the Job to run. Input `Resources` are kept in a separate list.

session The session this `Job` is a part of.

status Indicates whether the `Job` is idle, running, has completed successfully, or has completed with errors.

inputResources A list of `Resources` that serve as input to the job as well as the name of the parameter the `Resource` should be used for.

results A list of `Resources` with associated names that contains all results generated by the `Job`.

Together, `Resources` and `Jobs` can be represented as a bipartite directed acyclic graph that models the history of all computations required for obtaining a specific result (cf. Figure 5.5). In particular, Graviton explicitly models all intermediate steps, such as identifier mappings, as a `Job`-`Resource` pair. This makes documenting results obtained with Graviton as simple as providing a link to the result itself. Given the `Resource` representing the result, it is then possible to navigate to the `Job` that created it. Given the `Job`, its input `Resources` can be examined in turn. Using this strategy, all necessary parameter settings for all computations that took place on the server can be viewed.

As previously mentioned, metadata plays an important role when working with `Jobs` and `Resources`. A reason for this is that the file represented by a `Resource` cannot be changed once the `Resource` is created. This guarantees that code can rely on the fact that the data associated with a `Resource` will never change and thus allows sharing as well as reusing it. However, having the ability to add additional information to the `Resource` is useful for caching the results of expensive operations. Determining whether a file contains negative values or which samples are present in an expression matrix are examples for such operations. In addition, the user can provide and change inform-

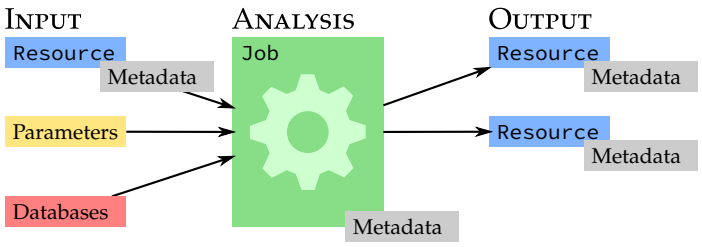


Figure 5.5: Bipartite dependency graph of `Jobs` and `Resources`. Each `Job` can have zero or more `Resources` as input and produces at least one results. The generated resources are annotated with additional, `Job`-specific metadata.

```
BASE_URL='https://genetrail2.bioinf.uni-sb.de/api'
SESSION=`curl -s ${BASE_URL}/session | \
    jq -r .session`
SCORES=`curl -s --form file=@scores.txt \
    ${BASE_URL}/upload?session=${SESSION} | \
    jq .results.result.id`
curl -s --data "input=${SCORES}&"\
    "categories=['9606-gene-kegg-pathways']" \
    ${BASE_URL}/job/gsea/setup?session=${SESSION}
```

Listing 5.6: Unix Shell script for starting a gene set enrichment analysis on the GeneTrail2 web server. `curl` is an application for interacting with web servers. `jq` is a command line JSON processor.

ation or annotations such as comments or the name of the `Resource`. Metadata can be arbitrarily structured. To account for this, Graviton stores metadata as a JSON object (Section 5.1.7).

API Layer

While `Resources` and `Jobs` make it straight forward to extend Graviton with new functionality, the `Resource` layer does not handle server-client interactions. For this purpose a RESTful API (Section 5.1.8) has been implemented. As a consequence it is possible to incorporate Graviton based web services into existing programs and scripts with relative ease. For example, Listing 5.6 shows a Unix Shell script which uploads a score file to the GeneTrail2 server (Section 5.3) and prepares a gene set enrichment analysis. The main functionality of the API layer is the creation and manipulation of `Jobs` as well as the upload and download of `Resources`. The API is self documenting, meaning that it provides end points that allow to obtain the parameters that are required for starting a specific `Job`. Admissible values for the parameters are documented in the same fashion. In addition to this, basic functionality for `User` and `Session` management as well as useful tools such as identifier mapping of `Resources` are offered. For conveniently accessing the API, Julia [Bez+14] and Python [VD09] wrappers are available.

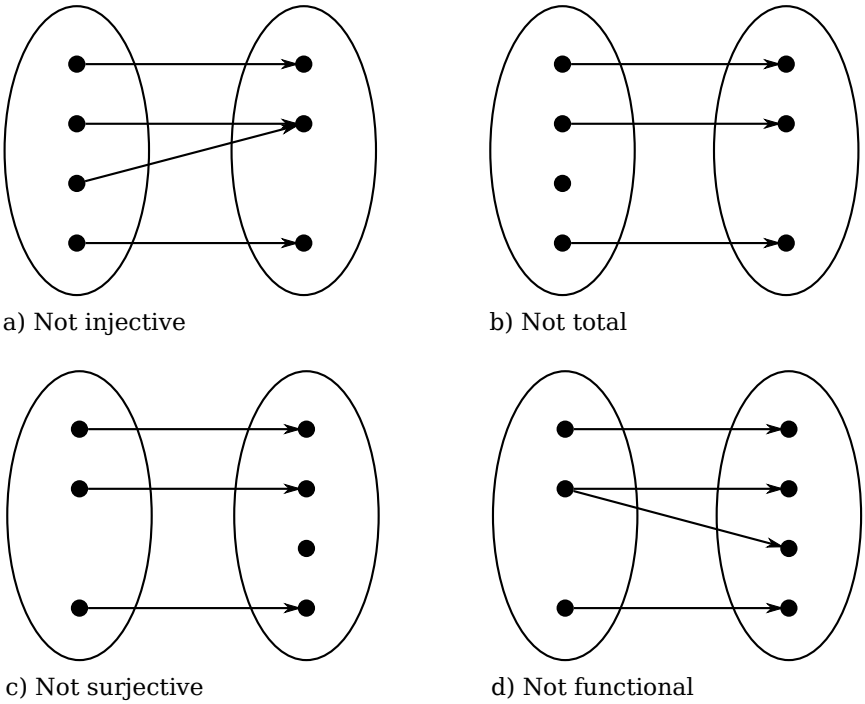


Figure 5.6: The four problems that can lead to data loss or biases due to identifier mapping. Note that these cases are not necessarily errors, but can reflect actual biological processes such as a gene encoding for multiple splice variants or a gene being targeted by multiple miRNAs.

Front end Layer

The front end layer provides a user interface on top of the RESTful API. To this end, Graviton provides a templating mechanism built using the Thymeleaf[1] templating engine and the Bootstrap[2] CSS framework. This includes a set of JavaScript bindings for communicating with the API, as well as reusable components with which workflows and result visualisations can be constructed. It should be noted, though that using the front end layer is strictly optional as all relevant information is already provided by the API.

5.2.3 *Identifier Mappings*

Identifiers are unique names for biological entities. Different biological databases often use different identifiers. Thus, to be able to work with data from multiple databases, a way to translate between their identifiers is needed. This *identifier mapping* is a central (and unsolved) problem in computational biology [Ier+10; Day+11; Dur+09]. In its essence, identifier mapping reduces to a simple question: "Given an identifier from database A, determine the identifiers from database B that match best." To be more precise, we can distinguish two kinds of tasks: map-

1 `http://www.thymeleaf.org/`
2 `https://getbootstrap.com/`

ping between identifier types that describe the same biological entity and mapping between identifiers for related, but distinct entities. We call the first *intra-* and the second *inter-species* mappings. An example for intra-species mappings is to map between gene identifiers from the *European Bioinformatics Institute (EBI)* and the *National Center for Biotechnology Information (NCBI)*. Instances of intra-species mappings are mapping genes to their expressed proteins or mapping miRNAs to their target transcripts.

Inter-species mappings are problematic in so far that the mappings between the identifiers can be expected to anything but bijective. In fact, it is possible that an identifier is mapped to zero, one, or more identifiers or, conversely, that zero, one, or more identifiers are mapped onto it (cf. Figure 5.6). For mappings between miRNAs and their target genes, the situation is especially grave [GE15; BLG15]. In general miRNAs target multiple genes, with some miRNAs having hundreds of experimentally confirmed targets [Cho+16]. Naturally, a gene can also be targeted by multiple miRNAs. Conversely, for many genes no targeting miRNA is known or exists.

For intra-species mappings, the situation is substantially less complicated. Still, cases exist where identifiers cannot be properly mapped. Reasons for this are missing and outdated data or incompatible curation policies. In the case of gene-gene mappings, the source and target database might have been created using different genome assemblies. Also, the definitions of a gene that are used by institutions such as the EBI and the NCBI differ slightly. Hence, some regions of the genome may be annotated as a gene in one, but not the other database. Joint efforts such as the *Consensus Coding Sequence (CCDS)* [Pru+09] project by the EBI, NCBI, *Wellcome Trust Sanger Institute (WTSI)*, and *University of Santa Cruz (UCSC)* to find a common definition for protein coding genes may help to eliminate this problem in the future.

A further complication stems from the fact that pure identifier mapping is seldom needed alone. Instead, identifiers are often associated with some additional data. For instance, the entity-level scores used in enrichment algorithms are usually stored as a list of identifier-score pairs. When mapping these lists, the associated data needs to be transferred appropriately. Due to possible ambiguities that occur while mapping, this means that in some cases multiple values can be assigned to the same target entity. In this cases a merge strategy must be applied.

Implementation

The core mapping algorithm employed by Graviton operates as follows. Given a mapping table, the applied mapping algorithm for translating identifiers from database A to B simply searches all target identifiers $l'_1, \ldots, l'_n$ for a source identifier l. If multiple identifiers were mapped onto the same target identifier, duplicate removal strategies can be applied. If no scores or expression values were associated with l, either all, the first, the last, or the middle identifier are kept depending on a user setting. If scores or other data was associated with the identifiers, a merge strategy can be specified. To this end, the mean, median, sum,

Data: List of identifiers L, mapping $m : A \to \mathcal{P}(B)$,
data $d : A \to X$.
Result: List of mapped identifiers L', mapped data $d' : B \to X$.
$L' = []$
$d' = \emptyset$
$d_{tmp} = \emptyset$
// Perform mapping and collect data
for $l \in L$ **do**
  for $l' \in m(l)$ **do**
    $d_{tmp}(l') = d_{tmp}(l') \cup d(l)$
  end
  $L' = \text{appendAll}(L', m(l))$
end
// Merge the mapped data
for $l' \in L'$ **do**
  $d'(l') = \text{merge}(d_{tmp}(l'))$
end
$L' = \text{removeDuplicates}(L')$
return (L', d')

Algorithm 5.1: Core mapping algorithm used in Graviton. The "merge" and "removeDuplicate" functions can be supplied by the user. We model data as a function that assigns a value from a set X to each identifier.

maximum, or minimum value can be used. Support for non-numerical data has not yet been implemented. Pseudo-code for the mapping algorithm is given in Algorithm 5.1.

The GeneTrail2 web service uses (HGNC) gene symbols internally.

Graviton uses identifier mapping to solve three distinct problems. First, user provided input data needs to be converted to the same identifier types as used internally. Second, user input must be sanitised before it can be used for further analysis. Reasons for this may be invalid, outdated, or misspelled identifiers in the input data that, when not corrected for, can bias downstream statistics. Third, Graviton supports inter-species mappings for advanced analyses and, in the case of GeneTrail2, for expanding the range of available categories. Each of these tasks has different requirements on the mapping engine which we will outline in the following.

Mapping from the identifiers in the user data to internal identifiers mainly exercises the core mapping algorithm. From a software engineering point of view, its only requirement are the availability of efficient lookup tables. For this, high performance dictionary implementations using b-trees such as LMDB[3], LevelDB[4], and MapDB[5] are available. As Graviton is written in Java, the Java-based MapDB engine was chosen to ease the integration with the remaining server code.

Sanitising user input requires that a list of known, "good" identifiers is kept for reference. Each identifier that is read from user data

3 https://symas.com/products/lightning-memory-mapped-database/
4 http://leveldb.org/
5 http://www.mapdb.org/

must then be validated against this list. This lookup must take the peculiarities of some identifier types into account. For example, mirBase identifiers are case sensitive, whereas gene symbols and other identifiers are commonly case *in*sensitive. In addition, outdated identifiers and known aliases need to be upgraded to the most current or canonical version. For this, database vendors sometimes provide mapping files that can be used to upgrade user inputs. If available, these files are integrated into the Graviton mapping database.

Intra-species mappings are, despite the problems mentioned above, no different from mappings between external and internal identifiers. Inter-species mappings, however, commonly exist in multiple versions. As an example consider miRNA–target mappings, where different criteria can be used to conclude whether a miRNA targets a certain transcript or not. Accordingly, a wide range of miRNA–target mappings have been produced that differ in the considered experimental methods and chosen significance cut-offs. To accommodate for this, Graviton labels each mapping with a quadruple

$$(\text{organism}, \text{from}, \text{to}, \text{variant}).$$

While the first three items depend on the source `Resource` and the desired output identifier type, the variant field allows to choose between alternative mapping definitions. For example the mapping

$$(\texttt{9606}, \texttt{mirBase21}, \texttt{GeneSymbol}, \texttt{mirTarBase-westernBlot})$$

corresponds to a mapping of human miRNA identifiers to target genes obtained from mirTarBase [Cho+16] that were validated using a western blot.

To make the mapping process transparent, Graviton exposes the available mappings on a web page[6] and via the RESTful API. Additionally, for each performed mapping, an audit log is kept that allows to reconstruct which identifiers of the source `Resource` were mapped onto which identifiers in the target `Resource`.

5.3 GENETRAIL2

CONTRIBUTIONS The GeneTrail2 web server and the enrichment methods it offers were implemented by Tim Kehl and me. The included data was collected and curated by Tim Kehl, Patrick Trampert, and me. The corresponding manuscript [Stö+16] was written by me and Hans-Peter Lenhof.

As shown in Chapter 4 a large variety of enrichment algorithms exists. We argued that it is difficult to determine a "best" enrichment algorithm, as the published methods assume different null hypotheses.

6 `https://genetrail2.bioinf.uni-sb.de/mappings.html`

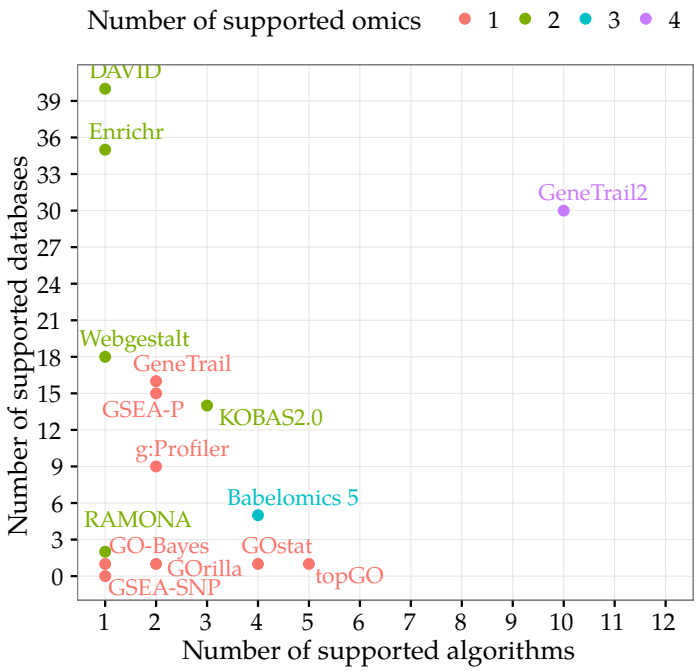


Figure 5.7: A comparison of selected enrichment tools. The *number of supported databases* refers to the number of unique data sources from which categories have been obtained. Databases are counted across all supported species and omics. The *number of supported algorithms* refers to the number of algorithms offered for analysis. Related methods (e.g. network algorithms) have been included. A tool was defined as *supporting an omics*, if it provides dedicated biological categories for this omics type. Figure adapted from Stöckel et al. [Stö+16].

This section uses figures, tables, and paragraphs from Stöckel et al. [Stö+16]. The WT case study subsection was to a large part adopted from the publication. However, additional findings and clarifications have been added.

To assist with this choice, we gave a set of recommendations based on a small study using synthetic data (Section 4.3). In addition, more pragmatic reasons for choosing certain algorithms exists. Based on the available input data, a large range of algorithms can often be ruled out *a priori*. For example, global tests and the sample-based permutation strategy rely on the availability of the full matrix of measurements. If only a sorted list of identifiers is available, the unweighted KS and the ORA procedures remain as the only applicable set-level statistics. If just a set of e.g. differentially expressed genes is provided, only ORA can be applied.

In order to enable the user to make such an informed choice, we implemented the GeneTrail2 web server. GeneTrail2 is built atop the Graviton framework and thus is tightly integrated with other services based on the same infrastructure. In total, we implemented 13 entity-level statistics (Section 4.2.2), 13 set-level statistics (Section 4.2.4), two p-value computation strategies (Section 4.2.1), and eight p-value adjustment methods (Section 4.1.2). For human alone, it features over 46,000 categories collected from over 30 databases including KEGG, Reactome, GO, WikiPathways, DrugBank, Pfam, miRWalk, and miRDB (cf. Appendix D.2). The server supports transcriptomics, miRNomics, proteomics, and genomics data and can convert between 32 common

identifier types. Data from all major omics is supported, making it possible to analyse and explore heterogeneous datasets in an interactive fashion using GeneTrail2's web interface. The web interface is built on top of modern web technologies with special attention on usability. Non-expert users can quickly perform comprehensive analyses using the predefined workflow, which is complemented with thorough documentation. Moreover, the interface enables users to integrate enrichments obtained from multiple omics using the integrated mapping procedures (Section 5.2.3) and our side-by-side view. For further analysis tasks, we offer a deep integration into existing applications like the network visualisation tool BiNA [Ger+14] or the NetworkTrail [Stö+13] web service (Section 5.4). As GeneTrail2 was built using Graviton, it exposes a RESTful API, through which power users can execute enrichment analyses directly from their preferred scripting environment. This allows the seamless integration of GeneTrail2 into workflow systems such as Galaxy [Goe+10] or Taverna [Wol+13]. The implementation of the enrichment methods relies on highly optimised C++ code leading to excellent runtime behaviour. GeneTrail2 can be accessed at `https://genetrail2.bioinf.uni-sb.de`. The C++ code is available on GitHub[7]. On average, the service is used by more than 150 unique visitors each month.

Workflow

GeneTrail2 allows to arbitrarily combine all implemented methods at every stage of an enrichment algorithm (cf. Figure 4.2). The number of available algorithms at every stage are:

1. Entity-level statistics: 13
2. Set-level statistics: 13
3. P-value strategies: 3
4. Multiple testing corrections: 2

In addition, various input data formats are supported (see below). Due to the resulting combinatorics, a considerable number of analysis workflows is possible (Figure 5.8). A typical interaction with the server looks as follows: first, the user uploads the data to be analysed, e.g. a matrix containing expression measurements. Next, the data points contained in the matrix can be distributed into sample and reference sets. These groups are then used as input for the computation of entity-level scores. After score computation, a set-level statistic is applied for the biological categories chosen by the user.

In each step the user can adjust all parameters exposed by the respective method. As this usually requires considerable expert knowledge, we provide defaults that should be applicable for most use-cases and that have been chosen conservatively in order to prevent false discoveries. Furthermore, uncommon settings are only accessible from an "advanced" user interface to prevent user errors.

7 `https://github.com/unisb-bioinf/genetrail2`

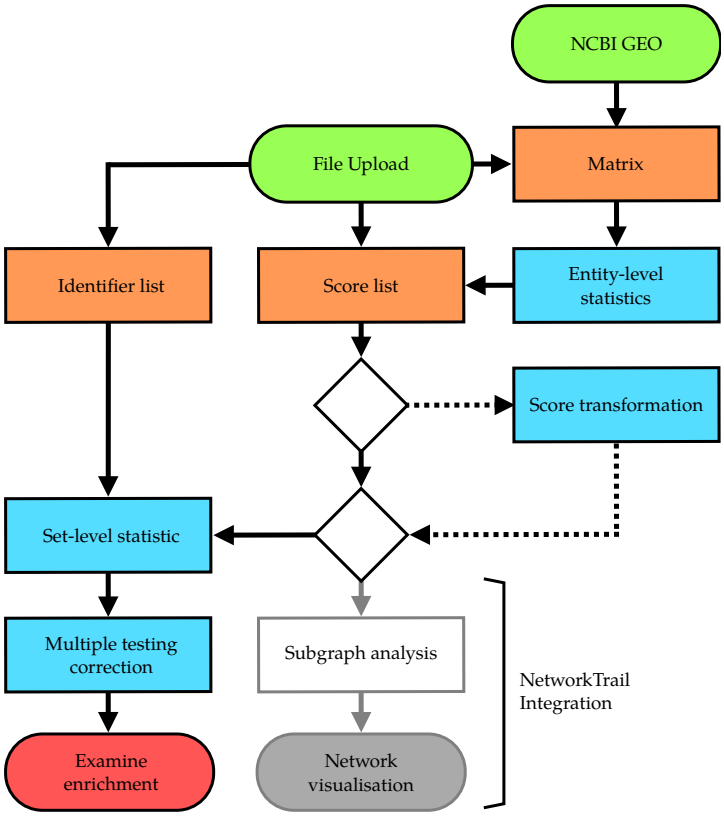


Figure 5.8: Simplified flowchart of the GeneTrail2 workflow. Round green and red boxes depict start and end states, respectively. Boxes with orange background represent input files types, whereas a blue background represents processing steps. Diamonds are decision nodes. Figure adapted from Stöckel et al. [Stö+16].

Supported Input Data

GeneTrail2 is able to read various input file formats through which the user can provide measurement data or categories that should be analysed. Using the infrastructure provided by Graviton, GeneTrail2 will try to automatically detect type and format of the uploaded data. In addition, the database identifiers and the organism, from which the data stems, are estimated. In the following, we discuss the expected input formats and the assumptions GeneTrail2 makes about their contents.

Identifier Lists The simplest way to provide input data to GeneTrail2 is to upload a list of identifiers. To this end, GeneTrail2 accepts a plain text file containing exactly one identifier per line. There are two ways to interpret such a file: as an unordered set or a sorted list. In the first case, the list can only be used as an input to ORA as no scores are available and, thus, all identifiers need to be considered as equally important. In the second case, the order of the identifiers defines an importance ranking, which can be used by non-parametric methods such as the unweighted KS statistics.

Entity-Level Score Lists Similarly to identifier lists, score lists can be provided in a text-based format. In addition to an identifier, each line contains an entity-level score. The two columns are separated by a tab or space character. In general, score lists are preferable to identifier lists as a score list can be used as an input for virtually every enrichment algorithm. In addition, score lists are less prone to problems which are frequently encountered with identifier list. A common example for this are unsorted identifier lists being used as input to the KS statistics. It should be noted that GeneTrail2 does not verify whether the uploaded scores follow a certain distribution or not. While most of the implemented methods work surprisingly well if their assumptions are violated, it is strongly recommend to avoid such unsound statistics. If the distribution of the data is unclear or unknown, the (unweighted) KS-statistics and the Wilcoxon test are non-parametric enrichment methods that do not require a specific score distribution (cf. Section 4.2.4).

Measurements GeneTrail2 provides support for directly analysing matrices containing high-throughput measurements. These can be normalised expression values obtained from microarray or RNA-seq experiments or protein abundances from mass-spectrometry runs. Additionally, rudimentary support for analysing raw count data obtained via RNA-seq is offered. More involved processing steps such as quality control, batch effect removal, and normalisation must be performed by the user.

Measurements can be uploaded as a plain text, tab-separated matrix. Optionally, the first row provides names for each of the contained samples. Each subsequent row contains the measurement data for one identifier in all samples. Thus each row except the first starts with an identifier followed by N numerical values, where $N \in \mathbb{N}$ is the number of samples.

```
Sample1 Sample2 Sample3
GeneA 0.1 4.3 2.3
GeneB 3.2 -1.2 1.1
GeneC 2.7 9.1 0.3
```

The main advantage of uploading matrices of measurements instead of entity-level scores is that sample-based instead of row-based permutation schemes can be used for determining the significance of enriched categories (cf. Section 4.2.1).

Microarray data A major use case of GeneTrail2 is the analysis of microarray data. For this platform, well established normalisation pipelines exist that usually generate normal or log-normal distributed expression values. GeneTrail2 can directly work with this normalised data and offers a range of statistics that can be used to derive scores from expression matrices.

RNA-seq data RNA-seq data usually comes in the form of count data. This means that for each transcript, the number of mapped reads is reported. The distribution of this data is fundamentally different to the distribution of microarray data, and hence specialised methods for the analysis of count data have been developed (Section 2.3.3). GeneTrail2 implements basic support for some of these methods. The user can choose between the DESeq2 [LHA14], edgeR [RMS10], and RUVSeq [Ris+14] algorithms for computing entity-level scores. All implementations are based on R packages from the Bioconductor [Gen+04] repository.

Note that currently sample-based permutations are not possible for count data due to the prohibitive runtime of the score computation process. In addition, while the used packages provide some level of normalisation, GeneTrail2 performs no quality control or batch effect removal.

miRNA Data Besides mRNA expression data, GeneTrail2 also supports the analysis of miRNA expression data (cf. Section 2.3). Two main analysis modes are available: specialised miRNA categories and mappings to miRNA target genes. For the first mode, we integrated categories obtained from the mirWalk 2.0 [DG15], HMDD 2.0[Li+13], and TAM [Lu+10] miRNA databases. We also defined categories based on target information from mirTarBase [Hsu+10; Hsu+14; Cho+16]. Enrichments of chromosomal regions can be computed using categories derived from mirBase [Gri+06]. In addition, custom categories created by Backes et al. [Bac+16] have been integrated. All performed computations are completely analogous to the mRNA case.

The target mapping strategy, however, works fundamentally different. As miRNAs are able to target a wide range of genes, transferring miRNA scores directly to target genes would result in a skewed score distribution that contains many ties. To circumvent this, a set of differentially expressed miRNAs is selected. These are then mapped to their targets as found e.g. in mirTarBase. These targets then serve as the input to an ORA enrichment using gene categories. The idea behind this is that the function of a miRNA is defined by its target genes and hence the annotations of the target genes can be "transferred" to the miRNA. The advantage of this strategy is that it vastly increases the number of categories that are available for miRNA data. However, various disadvantages exist. As miRNAs can have many targets, the test set for the subsequent ORA analysis can grow rapidly and unpredictably. Also, the target genes in the test set are highly dependent, which violates the assumptions underlying the hypergeometric test. This can lead to low statistical power and considerable artefacts [BLG15; GE15].

Protein Data Using GeneTrail2, it is also possible to analyse protein abundances. Basically, the same strategies as for the miRNA data apply. However, as a protein usually can be mapped unambiguously to its encoding gene, the mapping based strategy is considerably better behaved than in the miRNA case. Hence, protein abundances can be

used as gene scores which allows to use other enrichment methods such as the KS statistics and averaging based approaches. Nevertheless, GeneTrail2 also includes specialised protein categories extracted from Pfam [Bat+04] and Reactome [Jos+05].

Categories While GeneTrail2 offers a large collection of categories that have been derived from a number of third-party databases (Appendix D.2), it can be desirable to create custom categories that should be checked for enrichment. An example would be a set of potential targets of a transcription factor that have been identified by a Chip-seq experiment. For specifying categories, GeneTrail2 uses the *Gene Matrix Transposed (GMT)* format [Sub+07]. In this format every line represents a category. Each line is divided into columns by a tab character. The first column corresponds to the name of the category and the second column to an optional description. Each subsequent column defines a category member.

The specification for the GMT file format can be found under: `https://www.broadinstitute.org/cancer/software/gsea/wiki/index.php/Data_formats`

```
CategoryA http://test.url/A GeneA GeneB GeneC GeneD
CategoryB http://test.url/B GeneA GeneD
CategoryC http://test.url/C GeneD GeneE GeneH
```

Reference Sets Besides the list of relevant entities, the ORA method requires a second list of identifiers which represents the universe of identifiers that can be detected by an experiment. The input format is the same as for identifier lists. As the choice of the reference set is crucial for the performance of the method, a set should be chosen that best fits the assay used for data generation (Section 4.2.4).

5.3.1 *Provenance Data*

A common problem with bioinformatics tools relying on external data, such as gene categories, is that the integrated databases require regular maintenance to remain up to date. Due to the considerable amount of work that is required to do so, this is often neglected. Further issues add additional hurdles. For example, the release schedules between databases are often not coordinated. Consequently, the databases may use incompatible versions of the same identifiers, making it difficult to distil the available information into a consistent snapshot. A second, more technical, problem stems from the fact that database schemas, formats, URLs, and nomenclature can change between releases. In the best case, this is detected during integration and can be fixed by updating the code of the import scripts. In the worst case, wrong information is silently integrated.

Tools that work on outdated data are prone to creating false, misleading findings, for which there no longer exists any supporting evidence, as well as to missing true positive discoveries due to a lack of information. To make it possible to confirm that an analysis was conducted using up-to-date information, it is important to make the origin and date of retrieval of this data transparent. In GeneTrail2 this

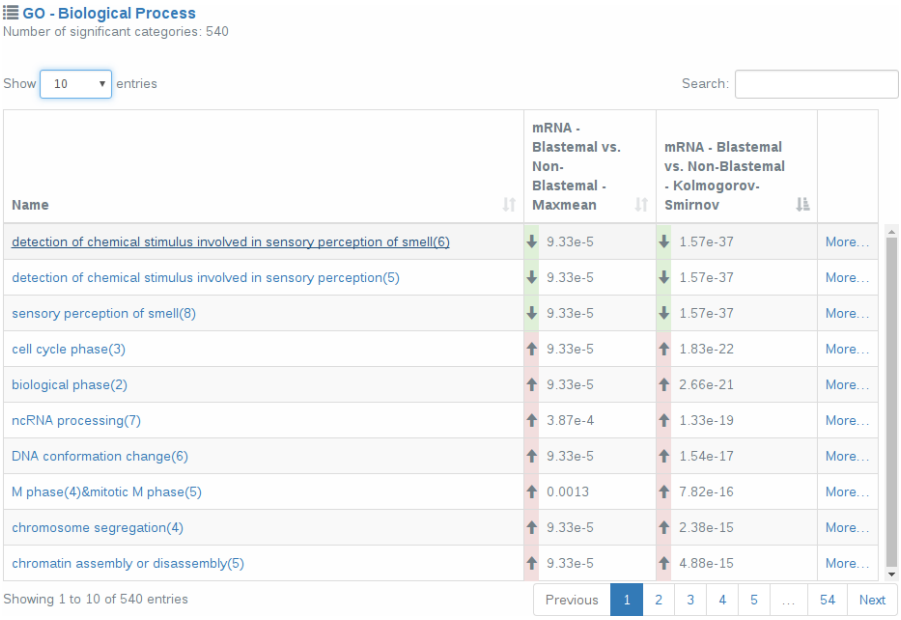


Figure 5.9: GeneTrail2's comparative enrichment view displaying common categories of two enrichments for the WT dataset (Section 5.3.5). For both enrichments p-values are shown.

is achieved by maintaining and, more importantly, providing provenance information for any data obtained from a third-party database. This includes:

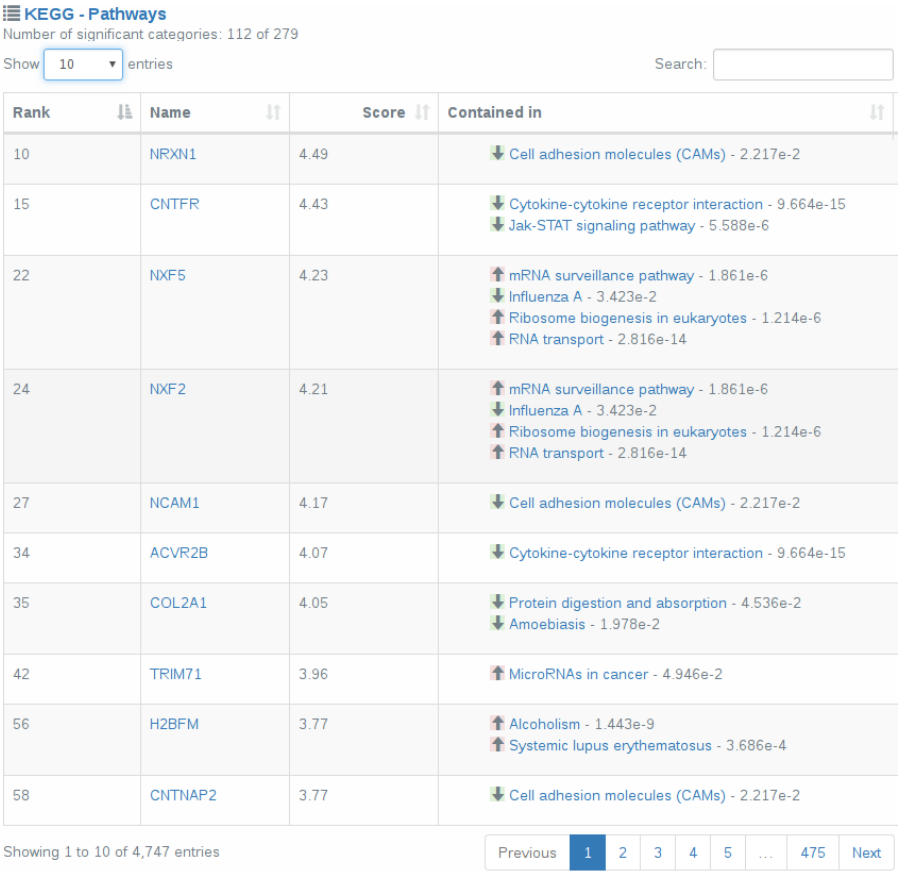
- Retrieval date
- Source URL
- Editor (Name & Email)

Here, the editor refers to the person that retrieved and integrated the database into GeneTrail2. Having this information available allows to efficiently resolve problems and questions concerning specific parts of the integrated databases.

5.3.2 *Comparative Enrichment View*

In some datasets, samples that were taken from different locations, like blood and tissue, are available for each patient. These sample cannot be directly compared and, hence, are often analysed in isolation. For enrichment analysis this means that two separate enrichments are computed. Detecting similarities and differences between these two enrichments can help to reduce false positives and to identify consistently deregulated pathways. However, no standard tools for this task exist, making it slow and labour intensive.

To solve this, we implemented the *comparative enrichment view*: a specialised view that allows to compare an arbitrary number of enrichments in a side-by-side fashion (Figure 5.9). Currently, two modes are



KEGG - Pathways

Number of significant categories: 112 of 279

Show 10 entries Search:

Rank	Name	Score	Contained in
10	NRXN1	4.49	Cell adhesion molecules (CAMs) - 2.217e-2
15	CNTFR	4.43	Cytokine-cytokine receptor interaction - 9.664e-15 Jak-STAT signaling pathway - 5.588e-6
22	NXF5	4.23	mRNA surveillance pathway - 1.861e-6 Influenza A - 3.423e-2 Ribosome biogenesis in eukaryotes - 1.214e-6 RNA transport - 2.816e-14
24	NXF2	4.21	mRNA surveillance pathway - 1.861e-6 Influenza A - 3.423e-2 Ribosome biogenesis in eukaryotes - 1.214e-6 RNA transport - 2.816e-14
27	NCAM1	4.17	Cell adhesion molecules (CAMs) - 2.217e-2
34	ACVR2B	4.07	Cytokine-cytokine receptor interaction - 9.664e-15
35	COL2A1	4.05	Protein digestion and absorption - 4.536e-2 Amoebiasis - 1.978e-2
42	TRIM71	3.96	MicroRNAs in cancer - 4.946e-2
56	H2BFM	3.77	Alcoholism - 1.443e-9 Systemic lupus erythematosus - 3.686e-4
58	CNTNAP2	3.77	Cell adhesion molecules (CAMs) - 2.217e-2

Showing 1 to 10 of 4,747 entries Previous 1 2 3 4 5 … 475 Next

Figure 5.10: GeneTrail2's inverse enrichment view showing the top 10 over-expressed genes in KEGG for the WT dataset (Section 5.3.5) that are members of an enriched category.

implemented in the view: intersection and union. While the intersection mode only displays categories that are significantly enriched in all enrichments, the union displays any category that is significantly enriched at least once.

This feature also allows to compare the results from different omics datasets. For example, protein abundance data can be mapped to genes in order to compute an enrichment that can then be compared with an enrichment for gene expression values. Finally, it is possible to use the view to eliminate false positive discoveries. To this end, multiple enrichments on the same dataset using different set-level statistics are computed. Using the intersection mode, only the enriched categories detected by all methods are retained, yielding a more robust combined enrichment. We explore this strategy in the evaluation of GeneTrail2 (Section 5.3.5).

5.3.3 *Inverse Enrichment View*

Commonly, enrichments are displayed as a list of categories with associated p-value and enrichment score. Additionally a list of member

		Broad GSEA	GeneTrail 2
KS	entity	400s (± 7.3s)	*9.3s (± 0.15s)
	sample	428.8s (± 4.32s)	84.5s (± 0.6s)
Mean	entity	N/A	3s (±0.02)
	sample	N/A	74.8s (± 1.7s)

Table 5.1: Performance data for enrichments computed on the KEGG categories using the (unweighted) KS and mean set-level statistics. For comparison the KS implementation of the Broad GSEA package [Sub+07] was used. Mean run times over five runs are given in seconds; standard deviations are provided in parenthesis. Both, entity- and sample-based p-value strategies were measured. In the comparison, the t-test was used as scoring method, no p-value correction has been performed, and 10,000 iterations were used for permutation tests. Results marked with a * employed an exact p-value computation method. Timings were obtained on an Intel Core i7-3770 processor. Table adapted from [Stö+16].

genes is displayed. This view is extremely helpful for quickly identifying deregulated processes. Sometimes, however, it is of interest to know in which deregulated processes a particular gene takes part. Answering this question using the traditional enrichment representation is tedious. To make this more efficient, we crated the *inverse enrichment view*. It shows every gene that is a member of at least one enriched category together with its entity-level score as well as the categories it is contained in (Figure 5.10). This allows to conveniently assess the influence of a gene on the computed enrichments.

5.3.4 *Performance*

Enrichment analysis is a basic building block for bioinformatics workflows and, hence, the runtime performance of these methods is critical. To this end, the algorithms included in GeneTrail2 use an efficient C++ implementation to guarantee optimal throughput. In general, the main bottleneck concerning execution time is the computation of p-values. The entity-based strategy commonly executes one order of magnitude faster than the sample-based strategy (Table 5.1). For the latter, the choice of the entity-level statistics can have a significant influence on the computation time, as it needs to be reevaluated for every sample permutation. In contrast, the used set-level statistics has little influence on the overall computation time. GeneTrail2 significantly outperforms the Broad Institute GSEA application [Sub+05] for both, the entity-based and the sample-based strategy (cf. Table 5.1).

5.3.5 *Case study: Wilm's Tumour*

In Section 2.1.1 we introduced *Wilm's tumours (WTs)*, a type of childhood renal tumours. Based on the composition of a biopsy after pre-

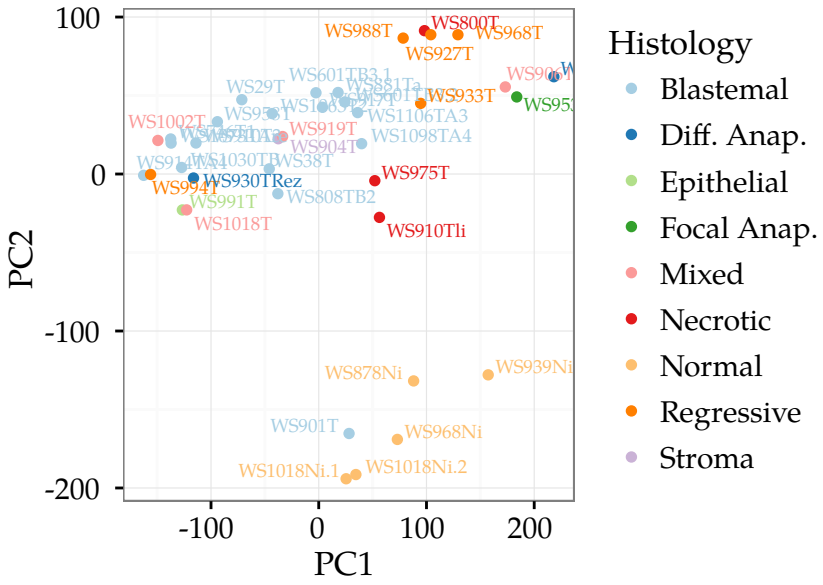


Figure 5.11: Plot of the first two principal components of the WT mRNA expression dataset. Cancer samples are clearly separated from the healthy control group. No clear separation of the tumour samples is visible upon visual inspection. The sample WS901T clusters with control samples and thus been discarded. Figure adapted from Stöckel et al. [Stö+16].

operative chemotherapy WTs can be categorised into subtypes. A subtype of special interest is the so-called blastemal subtype. It is composed to more than two third of living, blastemal cells. While tumours with a high blastem content generally respond well to chemotherapy, blastemal subtype tumours count to the most aggressive WT subtypes. Here, we used GeneTrail2 to analyse a WT expression dataset (Section 2.4) in order to determine key players influencing the malignancy of blastemal subtype tumours. To this end, we compare samples from the blastemal subtype with samples from other subtypes.

To obtain an overview of the general structure of the dataset, we computed a principal component analysis [Pea01] of the mRNA dataset (Figure 5.11). While a clear separation between healthy tissue and tumours can be seen, there is no reliable clustering of the tumour subtypes themselves. While blastemal subtype tumours occupy the upper left region of the plot, the remaining tumour types are scattered over the upper half. The sample WS901T, which clusters with the control group, was removed from further analysis as it is likely an outlier.

One of the first analyses that is applied to expression data is to determine the differentially expressed genes. In Figure 5.12 the adjusted p-value was plotted against the log-fold difference. Only eight genes show a significant differential expression at a 20 % FDR level. Of these, DPP10-AS1 encodes for a non-coding RNA with unknown function. HOXD1 and ATOH7 are transcription factors that play a role in developmental processes. The activator protein RSPO1 induces crypt cell proliferation in mice and is associated with an increased resistance to chemotherapy [Gu+15]. CDH7, MYO15A and PLOD2 play a role in the

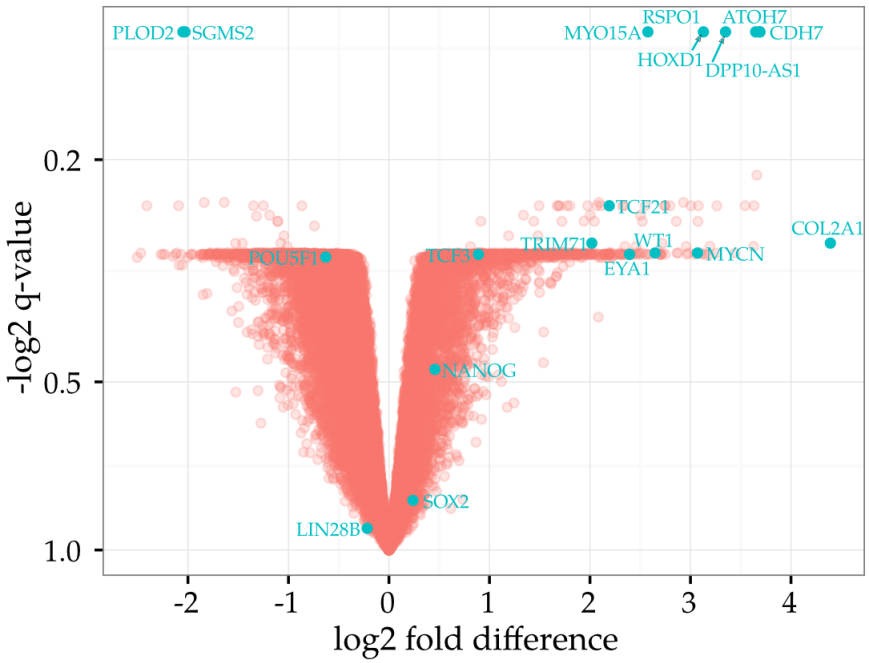


Figure 5.12: Volcano plot of the WT mRNA expression values. At a FDR level of 20 % only few genes are differentially expressed. The adjusted p-values were computed using the shrinkage t-test and Benjamini-Hochberg adjustment [BH95; YB99].

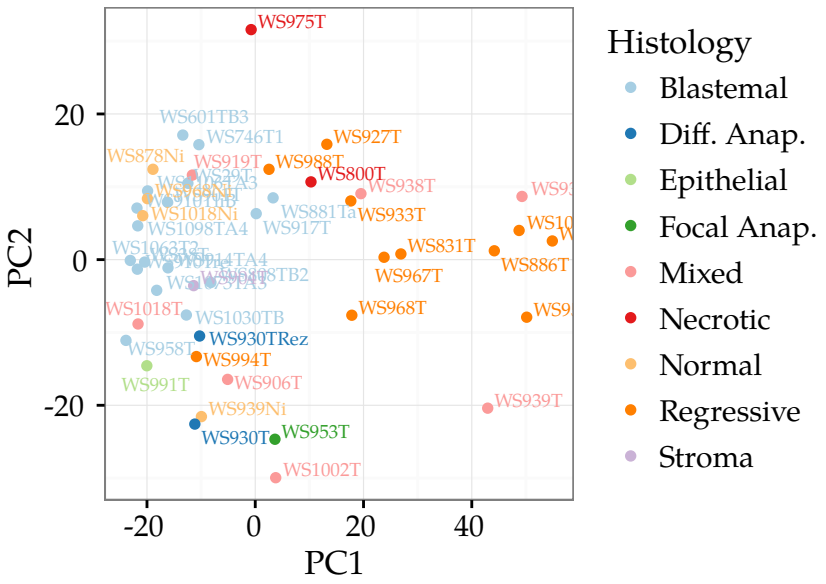


Figure 5.13: Plot of the first two principal components of the WT miRNA expression dataset. No clear separation between tumour subtypes can be detected. Control samples cluster with other cancer samples. Figure adapted from Stöckel et al. [Stö+16].

cytoskeleton and cell-cell adhesion.

In comparison to the mRNA data, the miRNA data seems to carry less information. For instance, no clear separation between the control group and the cancer samples is present in the PCA plot (Figure 5.13). However, substantially more miRNAs are differentially expressed than in the mRNA case (Table B.5).

While the analysis of differentially expressed genes uncovers some interesting results, the information is insufficient to form a hypothesis about the differences between the two groups. In the following we apply the algorithms provided by GeneTrail2 to demonstrate that enrichment algorithms in general and the functionality offered by the server in particular are instrumental for the analysis of biological datasets. To demonstrate how GeneTrail2 can be used in practice, the results of our analysis are layed out as a learning process that was guided by the computed enrichments. In particular we interpret the computed enrichments in the light of previously reported findings from the literature. Further, we augment our results by performing simple statistical analyses that are not directly offered by GeneTrail2.

This section is based on Stöckel et al. [Stö+16].

Methods The following workflow was used for computing enrichments: first, normalised expression matrices were uploaded to GeneTrail2. Blastemal tumours were assigned to the sample group and all remaining tumours were assigned to the reference group. Next, entity-level scores were computed using the *shrinkage t-test* (cf. Section 4.2.2). For each implemented set-level statistic, except ORA, we conducted an enrichment analysis using the entity-based p-value strategy (see Section 4.2.1). For using the ORA method a set of differentially expressed genes needs to be selected. This selection introduces a parameter for which a tuning step needs to be performed in order to find its "optimal" value. The method is thus difficult to compare with the remaining set-level statistics. Thus, to avoid possibly unfair comparisons, ORA was not considered in the presented study.

For methods requiring a permutation test (Algorithm 4.1), we set the number of permutations to 1,000,000 leading to a minimal theoretical p-value of 1e-6. In all cases, we used the Benjamini-Hochberg adjustment procedure (cf. Section 4.1.2) and a significance threshold of 0.05. Categories with less than three or more than 700 members were excluded from the enrichment computations to avoid statistical artefacts. Similarly, too large categories were excluded, as the results provided by them are not interpretable enough to be informative.

For computing the enrichments we used the GeneTrail2 RESTful API via a Python 3 [VD09] script[8]. The resulting enrichments are available online[9] and, due to their size, as supplementary files in the electronic version of this thesis.

8 `https://github.com/dstoeckel/Graviton.py`

9 `https://genetrail2.bioinf.uni-sb.de/results.html?session=a9e84e92-ea41-42ab-9ee7-c0f8515f9234`

Method	#Categories
Two-sample t-test	3866
One-sample t-test	3852
Two-sample Wilcoxon	3685
GSEA	3518
Mean	3424
Sum	3406
Weighted GSEA	2497
Maxmean	2057
Median	1989

Table 5.2: Number of significantly enriched ($p < 0.05$) mRNA categories found by each enrichment method for the blastemal vs. non-blastemal score list.

Consensus of Enrichment Approaches We use GeneTrail2's comparative enrichment view (cf. Figure 5.9) to analyse the computed enrichments. Despite a considerable overlap, the differences between them are substantial. While the union of all enrichments contains 1436 *GO - Biological Process* categories, their intersection only comprises 343 categories. Especially categories with a p-value close to the significance level are removed by this procedure. This may indicate that they were only reported due to idiosyncrasies of the corresponding method. The number of significant categories per method can be found in Table 5.2.

The above observation suggests a simple way to increase the specificity of the computed enrichments: only categories consistently reported by most set-level statistics are regarded as significant. Doing so should reduce most method specific false positives at the cost of eliminating some true positives. Thus, in the remainder of this case study we only consider categories that are reported by seven out of nine enrichment algorithms.

General Observations For mRNA the upregulation of categories like *mRNA Processing*, *Cell Cycle*, and *DNA Replication* suggests a clear increase in mitotic activity in blastemal tumours which may be explained by the larger amount of necrotic tissue in the reference group. For the miRNA data, categories associated with various cancer types, including *HMDD - renal cell carcinoma*, are significantly enriched. The same is true for miRNA categories involved in hormone regulation, immune response, apoptosis, and tumour suppression. No miRNA family is significant in *all* enrichments. However, the families *miR-302*, *miR-515*, *miR-30*, *miR-17*, and *let-7* are significant for at least seven out of nine tests (Supplementary Table 2). The miR-302 and miR-515 families are associated with the activation of the canonical WNT pathway [Ant+11]. Additionally, the miR-17 family is known for its roles as an oncogene and in stem cell development [MR13].

Deregulation of let-7 via LIN28B and TRIM71 The let-7 miRNA family has previously been reported to play a vital role in WT suppres-

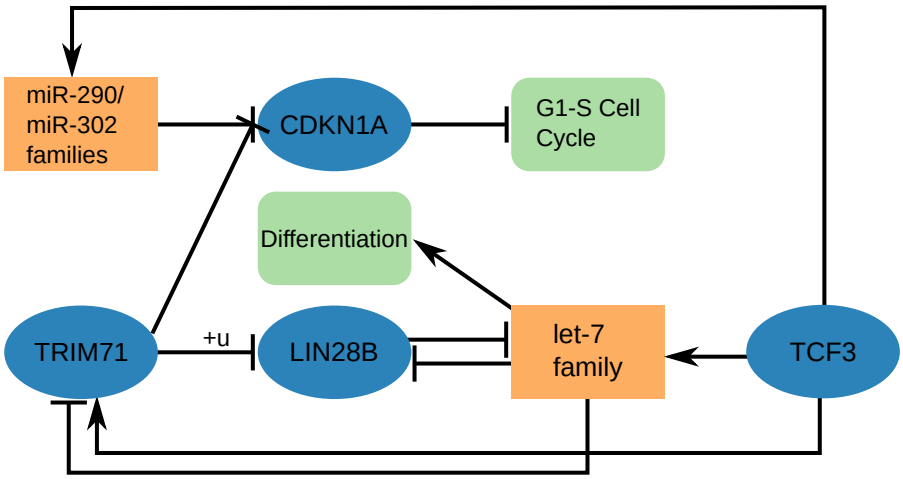


Figure 5.14: TRIM71 degrades LIN28B using ubiquitin-mediated proteosomal degradation. LIN28B regulates the maturation of let-7 miRNAs [Pis+11], which promote cell differentiation and act on TRIM71 and LIN28B via a negative feedback loop [Mar+08b]. TRIM71 as well as AGO2 in complex with miR-290/302 miRNAs repress CDKN1A expression leading to increased proliferation [Lee+14]. TCF3 acts on most of the above players [Mar+08b], effectively amplifying the currently predominant signal in the feedback loop. Figure adapted from Stöckel et al. [Stö+16].

miRNA	**t**	**Effect Size**
hsa-let-7f-5p	3.858	792.184
hsa-let-7a-5p	3.486	1622.645
hsa-let-7g-5p	2.834	199.181
hsa-let-7e-5p	2.566	94.476
hsa-let-7d-5p	2.273	15.424

Table 5.3: Significantly upregulated let-7 miRNA family members. t is the value of the shrinkage t-statistic as computed by GeneTrail2. The effect size is the difference between the unlogarithmised means of the sample and reference groups.

sion [Urb+14]. However, many highly abundant family members are upregulated (see Table 5.3), which is unexpected due to let-7 miRNAs acting as tumour suppressors. A possible explanation for this behaviour may be the differential expression of TRIM71, which is among the top-scoring genes ($t \approx 3.96$) and has the highest correlation of all genes with the absolute blasteme content $r \approx 0.76$. TRIM71 degrades LIN28B via ubiquitin-mediated proteosomal degradation [Lee+14] (cf. Figure 5.14). However, LIN28B in turn suppresses the maturation of pri-let-7 miRNA. Hence, the upregulation of TRIM71 induces an upregulation of the let-7 family, which, in theory, promotes cell differentiation [Urb+14]. However, Chang et al. [Cha+12] found that TRIM71 can promote rapid *embryonic stem cell (ESC)* proliferation in mice and report that it inhibits the expression of CDKN1A, a cyclin-dependent kinase inhibitor, which acts as a cell-cycle regulator. As high expression

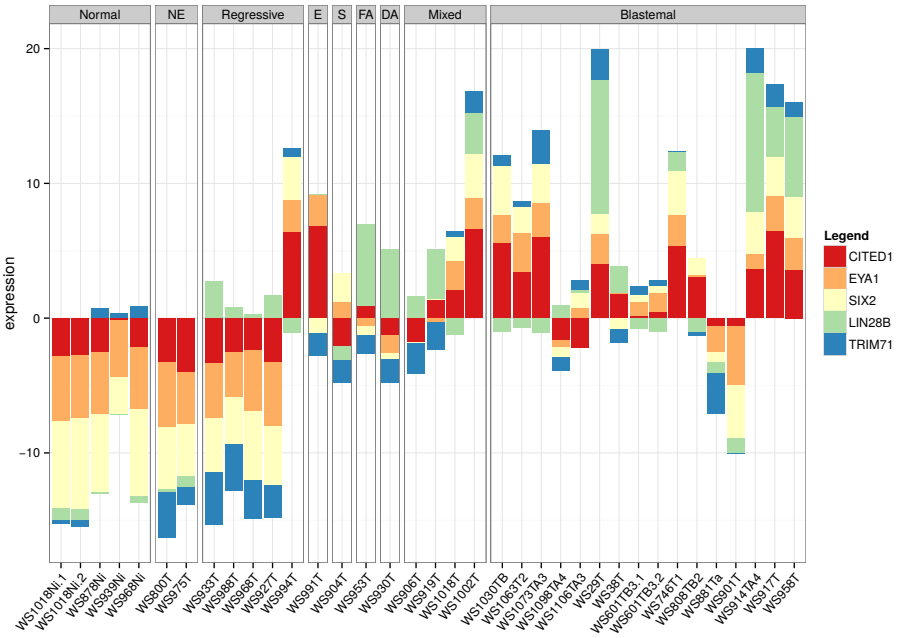


Figure 5.15: Expression of TRIM71 and LIN28B in comparison with the cap mesenchyme stem cell markers CITED1, EYA1, and SIX2. Samples are classified as *normal*, *necrotic (NE)*, *regressive*, *epithelial (E)*, *stromal (S)*, *focal anaplasia (FA)*, *diffuse anaplasia (DA)*, *mixed*, and *blastemal*. The size of a coloured bar represents the absolute expression value of the associated gene. Figure taken from Stöckel et al. [Stö+16].

levels of TRIM71 are commonly observed in undifferentiated cells, the authors conclude that TRIM71 is an important factor for maintaining proliferation in stem cells. Urbach et al. [Urb+14] report that LIN28B is able to induce WTs under certain conditions and note that in these tumour the cap mesenchymal (CM) specific stem cell markers CITED1, EYA1, and SIX2 are upregulated. Generally, this trend is present in our data, however, we find that the expression of the markers is more consistent with TRIM71's expression pattern (Figure 5.15). In summary, our initial results indicate that miRNA and genes associated with stem cell fate play an essential role in blastemal tumours.

Activation of cancer related WNT signalling Deregulation of the WNT signalling pathway is often prevalent in cancer samples [Pol12]. Indeed, our enrichment analysis contains categories associated with the *WNT pathway* (see Supplementary Table 1). This is especially visible in the inverse enrichment view (Section 5.3.3). There it becomes apparent that the previously discussed RSPO1 is a member of multiple, upregulated categories directly associated with WNT signalling (cf. Table 5.4). RSPO1 has been reported to regulate the WNT pathway via inhibition of ZNRF3 [Hao+12] and DKK1 [Kim+08]. In fact, DKK1 is downregulated in blastemal tumours, suggesting that the overexpression of RSPO1 may causally affect the measured expression signature. In addition, signalling molecules activated by RSPO1 have been shown to be sufficient for the induction of ovarian cancer [Zho+16].

Category	p-value
Positive regulation of canonical WNT signaling pathway	1.144e-2
Regulation of WNT signaling pathway	1.062e-3
Regulation of canonical WNT signaling pathway	1.529e-2
Regulation of endocytosis	3.340e-2
Canonical WNT signaling pathway	8.367e-3
Non-canonical WNT signaling pathway	1.756e-3
Positive regulation of WNT signaling pathway	2.775e-3
Receptor mediated endocytosis	4.959e-2

Table 5.4: Enriched categories for RSPO1. The gene is one of the most deregulated genes in the blastemal vs. non-blastemal comparison ($t \approx 5.17$) and plays a role in the activation of WNT signalling.

Besides RSPO1, other members of the WNT signalling pathway such as TCF21 can be found via the inverse enrichment view. These results are consistent with the pathway's previously reported, prominent role in most WTs and particularly in blastemal WTs [Fuk+09]. The activation of the canonical WNT pathway usually leads to degradation of the destruction complex that, as long as it is functional, degrades the transcriptional coactivator β-catenin [SW13]. Thus degradation of the destruction complex leads to higher amounts of β-catenin in the cytoplasm that is transported to the nucleus where it builds complexes with TCF/LEF proteins. Degradation of the destruction complex lies at the core of developmental processes, ESC self-renewal, and differentiation. As a result, it changes the transcriptional landscape of the cell dramatically. This is also consistent with reports that RSPO1 activates β-catenin in mammalian ovaries thereby controlling the differentiation process [Cha+08].

TCF3 as potential WT master regulator We argued that factors associated with stem cell fate and the canonical *WNT pathway* play an essential role in blastemal tumours. To further substantiate this claim we take a closer look at TCF21. TCF21 has been reported to bind the *transcription factor (TF)* TCF3, thereby inhibiting the expression of the KISS1, a known metastasis suppressor [Ara+11]. TCF3 itself is a well known link between the WNT pathway and the core regulatory circuitry of ESCs. Together with the pluripotency factors POU5F1 (OCT4), NANOG, and SOX2, TCF3 constitutes the set of "ESC master regulators" [Col+08]. If the WNT pathway is inactive, TCF3 is mainly repressing pluripotency factors and promoting differentiation. However, if the WNT pathway is activated, the repressive complex converts to an activating complex, promoting pluripotency [Col+08]. To study the influence of ESC master regulators, we constructed a new set of gene categories that we subjected to the KS test using the blastemal vs. non-blastemal scores as input. For each of the four TFs, we defined two categories containing genes for which "strong evidence" exists that they are regulated by the respective TF. In particular, we add a gene to a category for a TF if the

Gene	Correlation	Gene	Correlation
BMI1	0.91	CCND2	0.75
CDK4	0.83	EYA1	0.91
HMGA2	0.75	IGF2	0.89
LEFTY1	0.74	MAX	-0.76
MEIS1	0.68	MYC	-0.57
MYCN	0.86	NOTCH1	0.6
SMAD3	0.74	TP53	0.76
TRIM71	0.65		

Table 5.5: Pearson correlation coefficient between the expression values of a set of selected genes and TCF3.

TF occupies a site in the gene's promoter region and the correlation between the TF's and the gene's expression is larger than 0.5 (positive category) or smaller than -0.5 (negative category). For the identification of the promoters occupied by the master regulators, we used the mouse ESC ChIP-Chip dataset of Cole et al. [Col+08] and the ChIP-Seq data set of Marson et al. [Mar+08b]. Using this procedure we obtained more than 1500 genes, including many other TFs and genes involved in ESC fate, influenced by mainly TCF3 and OCT4. For a selection see Table 5.5. Our Kolmogorov-Smirnov enrichment[10] revealed that genes positively regulated by TCF3 ($p \approx 10^{-40}$) and NANOG ($p \approx 10^{-13}$) are strongly enriched, whereas genes negatively regulated by TCF3 ($p \approx 10^{-40}$) are strongly depleted. Conversely, genes positively regulated by OCT4 are strongly depleted, and genes negatively regulated by OCT4 are strongly enriched. This is consistent with a correlation of TCF3 with OCT4 of -0.7. SOX2 and NANOG both seem to be of lesser importance in our data.

However, the four master regulators do not only regulate protein coding genes. Marson et al. [Mar+08b] revealed that they are also "associated with promoters for miRNAs that are preferentially expressed in ESCs". Examples are the miR-302, miR-515, and let-7 families which we previously discussed (cf. Figure 5.14). Additionally, our data indicates that TCF3 regulates the expression of the miR-17 cluster (all correlations > 0.5).

IGF2 as Putative WNT Activator In the above section, we have outlined how the ESC regulatory circuitry is driven by TCF3 via the WNT pathway. However, the mechanisms that activate WNT signalling still remain unclear. Whereas certain genetic mutations occur with relatively low frequency (≤ 30 %), among them genes that may induce WNT signalling, the loss of heterozygosity and imprinting at the IGF2/H19 locus have been reported for 81 % of all blastemal WTs [Weg+15] leading to an overexpression of IGF2. Morali et al. [Mor+01] showed that IGF2 can induce the expression and import of β-catenin and TCF3 into

10 `https://genetrail2.bioinf.uni-sb.de/results.html?session=cbc86903-4248-47a2-b916-bc682924c242`

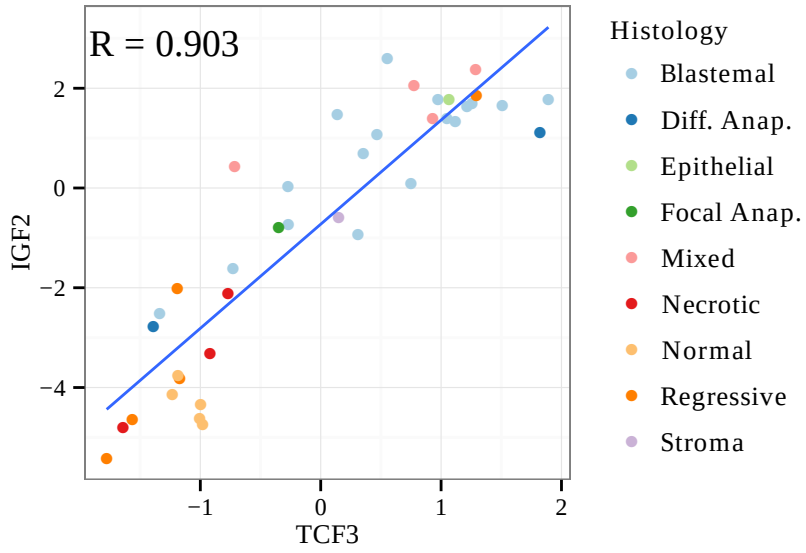


Figure 5.16: Scatterplot of IGF2 vs. TCF3 expression values. The correlation is $r \approx 0.89$. Figure adapted from Stöckel et al. [Stö+16].

the nucleus even in the absence of WNT proteins. This triggers a switch-over from the epithelial to the mesenchymal cell state, which is in accordance with the expression patterns of the mesenchymal stem cell markers shown in Figure 5.15. Additionally, TCF3 binding sites have been found in the IGF2 gene [Col+08]. Remarkably, we observe an extreme correlation of ≈ 0.89 between the TCF3 and IGF2 expression (see Figure 5.16). This suggests that TCF3 in turn regulates IGF2 leading to a self-sustaining feedback loop which is likely to be causal for the stem cell character of blastemal tumours.

Summary

In the presented case study, we showed how GeneTrail2 can be applied in a real world research scenario. The computed enrichments help to quickly identify interesting processes in a sample. This is especially true when combining multiple enrichments via the comparative enrichment feature (Section 5.3.2) or when annotating differentially expressed genes via the inverse enrichment view (Section 5.3.3). The ability to combine the results from multiple enrichment algorithms allows to focus on the most prominent signal in the data first. In particular, we were able to detect significant differences between "regular" and blastemal subtype WTs. Most notably the activation of WNT pathway via *RSPO1* and the subsequent stabilisation of WNT signalling by a putative β-catenin – TCF3 feedback loop. This feedback loop manifests in a correlation of 0.89 between TCF3 and IGF2. Furthermore, our analysis of the expression pattern of TCF3 regulated genes shows that they are consistently upregulated. This hints at the important role played by TCF3 in blastemal tumours.

5.4 NETWORKTRAIL

CONTRIBUTIONS The NetworkTrail web service and the application note [Stö+13] was written by Oliver Müller and me. Tim Kehl created the site design and implemented the Cytoscape.js visualisation, the scoring methods, and the FiDePa algorithm. Andreas Gerasch implemented the BiNA visualisation plugin.

The deregulation of signalling pathways plays an important role during tumour development (cf. Section 2.1). Enrichment methods allow to search for deregulated pathways but treat them as "bags of genes" (Chapter 4). This neglects the interaction data that is stored in pathway databases. However, methods that exploit this information to improve their accuracy by explicitly taking network structure into account exist. In Section 3.5 we previously discussed some of the available methods for detecting deregulated subgraphs. As an example, we presented the theory around the approach by Backes et al. [Bac+12]. Here, we focus on the practical side and present *NetworkTrail*, a webservice for detecting deregulated subgraphs which we built using the Graviton framework. We start with a short survey of available software packages for detecting deregulated subnetworks.

5.4.1 *Software for Searching Deregulated Subgraphs*

The simulated-annealing-based method of Ideker et al. [Ide+02] for the detection of active subgraphs is available as a Cytoscape [Sha+03] plug-in. The BioNet software developed by Dittrich et al. [Dit+08] can be downloaded as a R package [R C16], which implements an exact solver for the prize-collecting Steiner-Tree problem. The algorithms developed by Keller et al. [Kel+09] and Dao et al. [Dao+11] are provided as C++ source code. The original implementation of the ILP proposed by Backes et al. [Bac+12] is available as an executable.

The majority of the tools above does neither provide a *graphical user interface (GUI)* nor an option to visualise the resulting subgraphs directly. Instead, most need to be executed from the command line and the produced output needs to be processed using third party packages. Thus, besides installing the software properly, users are also required to possess intricate technical knowledge of the complete tool chain. According to our experiences, both tasks can be challenging for non-expert users. As we argued at the beginning of this chapter, a common strategy to prevent these problems is to make the tool accessible via a web interface. The same line of reasoning not only holds for enrichment methods, but also for algorithms that detect deregulated subnetworks. The HotNet algorithm [VUR11], for instance, is accessible as MATLAB source code and via a basic web interface. Also, List et al. [Lis+16] made the KeyPathwayMiner algorithm [Alc+11; Alc+14] available as a web service.

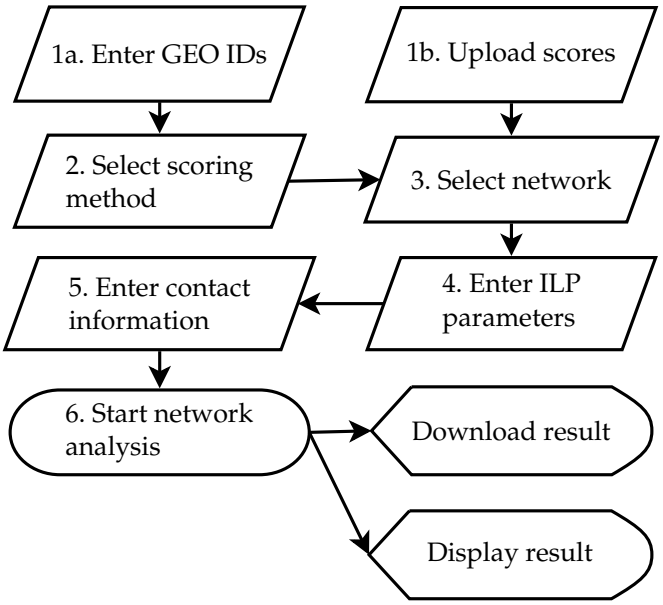


Figure 5.17: The basic workflow of a NetworkTrail analysis. Once data has been uploaded and scores have been computed, the user can select the network on which the analysis should be conducted. Then, the parameters of the chosen algorithm can be modified. After the computation has completed, the network can be visualised using Cytoscape.js or BiNA.

Using Graviton, we are able to provide an approachable web interface that integrates well with other services such as GeneTrail2. Accordingly, we developed a service called NetworkTrail [Stö+13] that allows to detect deregulated subgraphs in biological networks using our ILP-based approach [Bac+12] and the FiDePa algorithm [Kel+09].

FiDePa systematically enumerates all paths of length k and scores them using the Kolmogorov-Smirnov statistics (cf. Section 4.2.4).

5.4.2 *Implementation*

In order to integrate network analyses into Graviton, only a few modifications were necessary. To represent computed subnetworks a new kind of `Resource` was introduced. This `Resource` bundles all computed data into single archive. In particular, it contains the computed subnetworks, the scores used for the computation, and mappings of nodes to external database identifiers. Edges are encoded in the Cytoscape simple interaction format (.sif) whereas node properties are encoded in the node attribute format (.na) (Section 3.3.1). For documentation purposes a file containing all used parameters is included in the archive. `Job` subclasses handling the respective parameters were created for all supported algorithms. To ensure optimal efficiency both, the Subgraph ILP as well as FiDePa, have been implemented in C++. The ILP formulation uses the Branch & Cut framework offered by the commercial CPLEX library for solving the ILP instances (Section 3.5.2). We chose CPLEX as it provides superior performance to open source alternatives and free licenses are available for academic use.

5.4.3 *Workflow*

An overview over an example workflow is given in Figure 5.17.

With Graviton, NetworkTrail and GeneTrail2 share a common basis. Thus, the accepted input formats are, to a large degree, the same (cf. Section 5.3). As input, a list of entity-level scores is required, which can either be uploaded directly or computed from an expression matrix or a *Gene Expression Omnibus (GEO)* [EDL02] record. Once scores are available, the user can select the network that should be used for the analysis. Next, for both implemented algorithms, a range of desired subnetwork sizes can be entered. Subsequently, the algorithm computes the maximally deregulated subnetwork.

The result can be visualised using a custom, Cytoscape.js [Fra+16] based view or the *Biological Network Analyzer* [Ger+14]. Both tools display the union of all computed subgraphs using a hierarchical layout. Detailed information about each node is available. For genes, known aliases and references to external databases are displayed. For protein families and complexes the respective members are shown recursively. The subnetworks for specific values of k can be highlighted individually. The degree of deregulation of each node is indicated by its colour. White represents no change and shades of red and green represent up- and downregulation, respectively. Moreover, the BiNA based visualisation permits the complete customisation of the visualisation, including layout, colours, and node styles (cf. Figure 5.18).

5.4.4 *The KEGG Regulatory Network*

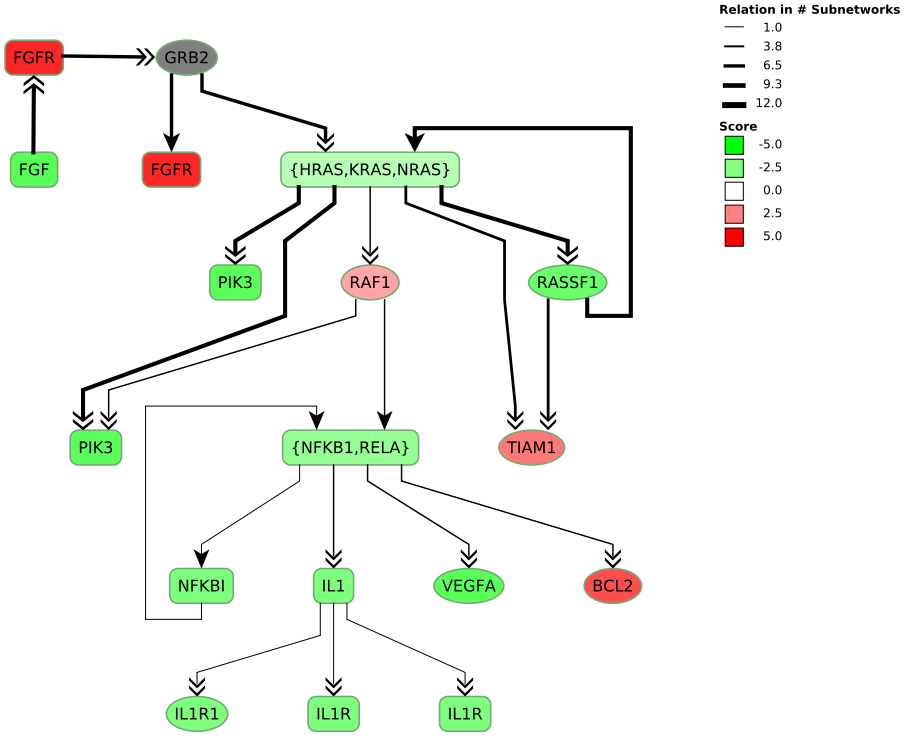
The KEGG database [KG00; Kan+10b; Kan+06] is a comprehensive catalogue of regulatory and metabolic processes (Section 3.2.1). Similar to other pathway databases such as Reactome [Jos+05] and WikiPathways [Kel+12] it is structured into pathways that describe a specific biological process in detail. We extracted the regulatory information contained in KEGG to create a complete regulatory network. To understand this process it is important to know how KEGG structures its information. Each pathway is described by a file in the KGML format. Every file contains the pathway's nodes as well as their interactions. In general three node types can be distinguished:

Gene A gene, unsurprisingly, represents a single gene.

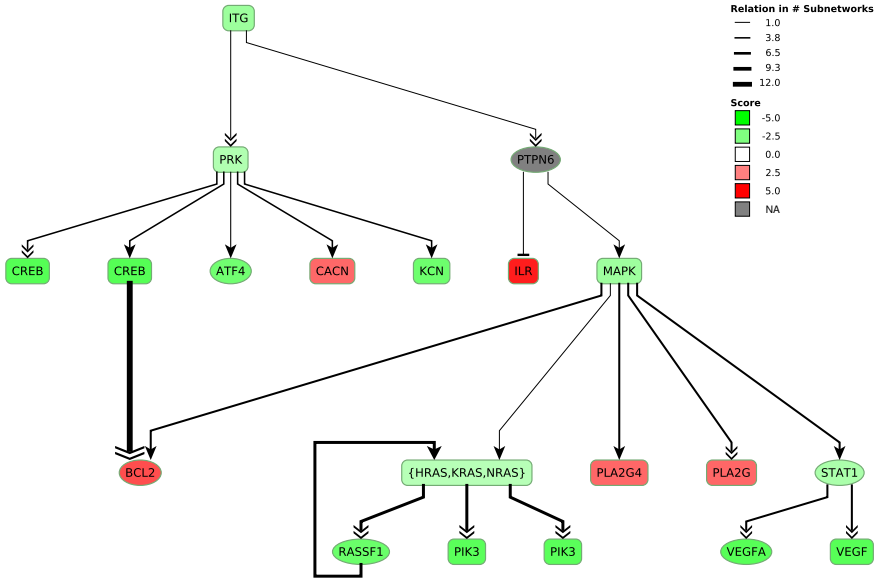
Family A family represents a group of genes. Each of the genes in a family are able to perform the interactions the node is a part of.

Complex A complex represents a group of genes or smaller complexes that must form a physical complex before they can take part in specific interactions.

Each interaction is labelled with an interaction type. This can be activation, inhibition, genomic interaction, or association events. Additional modifiers, such as indicators for phosphorylation and ubiquitination, can be attached to each edge. It also possible that complexes



(a) Subgraph for $k = 18$.



(b) Subgraph for $k = 20$.

Figure 5.18: Most deregulated subgraphs of size 18 and 20 for the Wilm's Tumour dataset (cf. Section 2.4). The image was created using the integrated BiNA [Ger+14] visualisation. Red and green backgrounds represent up and downregulated nodes, respectively. Gray backgrounds indicate that no score could be mapped to this node. The thickness of an edge is proportional to the number of subgraphs it is a part of (i.e. both target and source node were selected for a subgraph). Ellipses represent single genes, whereas rounded rectangles represent families of genes.

contain families of genes that can take each other's place in the complex. To import data from KEGG, NetworkTrail uses a custom import application written in Java. For *Homo sapiens* the resulting network consists of 4186 interactions and 2579 nodes.

5.4.5 *Example*

To give an impression of how to apply NetworkTrail in practice, we present a short example. We used the entity-level scores computed in the GeneTrail2 evaluation (Section 5.3.5) to compute the most deregulated subnetwork using our ILP (Section 3.5). The scores were computed using the shrinkage t-test. Blastemal and miscellaneous tumour samples served as sample and reference group, respectively. The scores were uploaded to NetworkTrail and the KEGG regulatory network was chosen as network topology. We computed the most deregulated subgraphs for sizes 3 to 20. The most deregulated subgraph of size 18 and 20 are shown in Figure 5.18.

The computed networks are not stable in the sense that nodes are successively added to a core deregulated network structure. Instead the root as well as the remaining selected nodes frequently change for close values of k. For example the subgraphs depicted in Figure 5.18 only share a few common nodes: HRAS, KRAS, NRAS, PIK3, VEGFA, and BCL2. This may be an indication that several effects are taking place simultaneously in the sample such that a single key player is insufficient for explaining all differentially expressed genes.

Interestingly hardly any gene previously discussed in Section 5.3.5 can be found in the computed subgraphs. Instead most genes such as the PIK3 family or BCL2 can be associated with immune response mechanisms. A possible explanation for this can be found in the increased resistance of blastemal tumours to chemotherapy. The overexpressed BCL2 is known for promoting chemotherapy resistance and its central role in inducing leukaemia [Ota+07]. Conversely, as non-blastemal subtypes are less aggressive and responded better to chemotherapy, more necrotic cells are present in the collected samples and hence higher immune system activity can be expected in the reference group, thus explaining the observed underexpression. Furthermore, as the KEGG databases contains especially well-curated regulatory pathways for the immune system, these subgraphs may have been preferred to the subgraphs containing our previously identified genes.

Upon further inspection, more limitations of the KEGG network become apparent. For example multiple family nodes containing PIK3, CREB, or IL1R genes have been selected. These nodes represent groups of genes fulfilling the same function at this place of the network. Consequently, family nodes can overlap, but are not necessarily identical. This makes interpreting the returned subgraph difficult, as the score of a single gene can have contributions to multiple nodes.

5.4.6 *Summary*

Here, we presented NetworkTrail, a web service for detecting deregulated subnetworks in biological network data. The availability of such a web service allows non-experts to carry out network analyses without having to struggle with technical details such as compiling/installing software or learning cryptic commands. We are convinced that such easy-to-use web services will help to elucidate pathogenic mechanisms and that they may also proof useful for therapy stratification in cancer therapy. Our example showed that NetworkTrail makes computing deregulated subgraphs simple. The integration with GeneTrail2 allows to directly reuse scores that were used for computing enrichments. However, the limiting factor of the analysis is the underlying network. Due to the topology of KEGG, some pathways seem to be preferred over other parts of the network (cf. Section 3.5.3). Also, the concept of family nodes and, to a lesser extent, complexes makes arguing about the network difficult. In the future, NetworkTrail will be enhanced by including further analysis methods and more complete, predefined networks.

5.5 DRUG TARGET INSPECTOR

CONTRIBUTIONS DrugTargetInspector [Sch+15] was implemented by Lara Schneider with design input and code review by me. Andreas Gerasch implemented the BiNA visualisation plugin.

As outlined in Section 2.2, a detailed understanding of a tumour's molecular properties can be essential for optimising the treatment for a patient. To this end, biological high throughput assays can provide valuable information complementing the classical, histological examination of a biopsy. For example, gene expression profiles allow to detect parts of the metabolism that are deregulated in tumour cells when compared to healthy tissue (cf. Section 2.3). Mutations can dramatically alter the function of a protein or even lead to the creation of new, fusion proteins [Sod+07]. Also, mutations in untranslated parts of the genome can result in increased or decreased transcriptional efficiencies for a gene as enhancer or promoter regions my be impacted [Lei+95; WKG83; BSM93]. Thus, knowledge about the mutations accumulated by a tumour can provide critical information necessary for interpreting the obtained expression patterns. Furthermore, epigenetic marks, such as methylation and histone occupancies, may prove useful for augmenting the knowledge derived from the mutation data. Finally, protein abundances can give a more detailed picture about the actual physiological processes that are taking place in a tumour cell. Making sense out of this host of information is a difficult task. For a physician trying to determine the optimal treatment for a patient, a succinct summary of

the data can serve as a stepping stone for assessing treatment options. As previously discussed, enrichment methods are a useful tool to focus on essential pathogenic processes. Nevertheless, an enrichment does not provide a quick way to determine possible treatment options. Instead, a considerable amount of work is required to interpret the computed results. For each category, it must be determined whether the corresponding biological process is of clinical relevance. If this is the case, the reason *why* the category is reported as significant needs to be determined. In particular it is important to identify potential key players that are e.g. differentially expressed and are the target of a cancer drug. To do so requires substantial time and expertise. We thus argue that enrichments alone are a suboptimal starting point for selecting a treatment.

As an alternative, we created the DrugTargetInspector (DTI) web service [Sch+15]. DTI integrates expression and mutation data as well as the knowledge about available (anticancer) drugs and established treatment regimes into a condensed, single-page overview. Starting from this overview, additional information can be accessed and advanced analyses can be performed. The service is built on top of the Graviton framework (Section 5.2) and, hence, integrates tightly with GeneTrail2 (Section 5.3) and NetworkTrail (Section 5.4).

In the following, we give a detailed explanation of DTI's features. We start with the information directly visible in the main view. Afterwards we discuss additional information that can be accessed via analyses. Finally, we describe DTI's integration into GeneTrail2, NetworkTrail, and the BiNA network visualisation tool.

5.5.1 *Integrated Databases*

A key feature of DTI is the aggregation of information from external databases into an internal storage. For each case, DTI uses this knowledge base to identify pieces of information that are relevant for the samples which are currently being examined. This information is then displayed in a succinct summary. Here, we shortly discuss the integrated databases. As the primary data source we use *DrugBank* [Wis+06]. The database entries for a wide range of drugs including the known targets for each drug. We use this information to classify drugs into categories such as cancer drugs, vitamins, or inhibiting drugs and to assign them to their target genes. The list of recommended drugs for each cancer type was obtained from the *American Cancer Society (ACS)*[11]. To be able to better assess the influence of a mutation on the efficacy of a drug we incorporated the pharmacogenomics data provided by the *Genomics of Drug Sensitivity in Cancer (GDSC)* project [Gar+12]. Information about each gene is provided by the *Entrez Gene* database [Mag+05]. For each drug–gene combination, links to relevant PubMed [Med97] publications are displayed.

11 `http://www.cancer.org/`

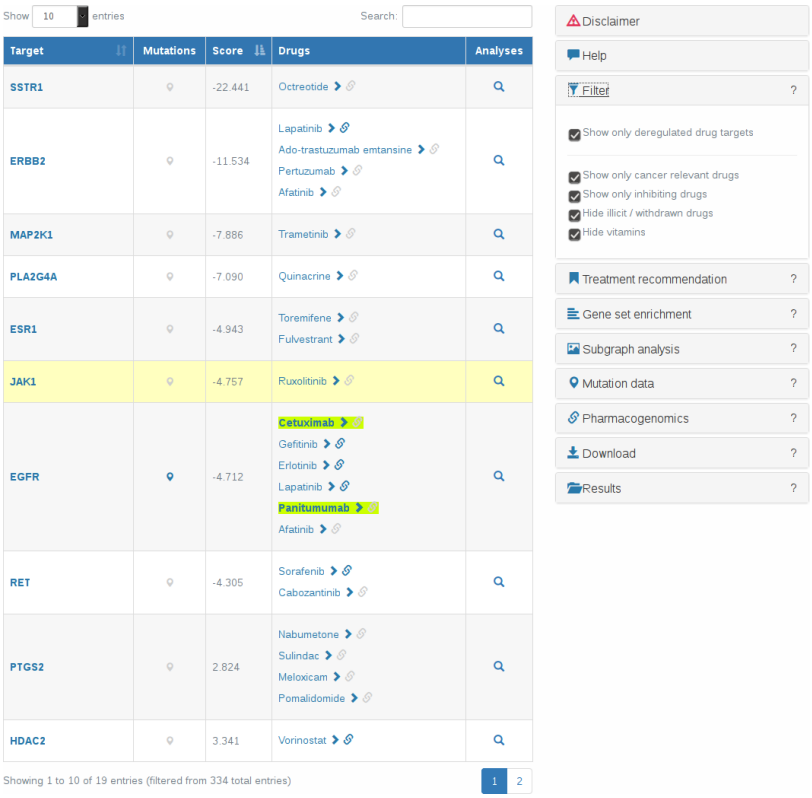


Figure 5.19: The DrugTargetInspector main view. On the right additional information, analyses and filters can be accessed. On the left the main table is depicted. For each gene, the name, possible mutations, gene score, and targeting drugs can be seen. Gene specific actions can be triggered using the last column. Recommended drugs for a specific cancer type are highlighted in green.

5.5.2 *Required Input*

DTI primarily operates on entity-level scores. Thus, a list of scores per gene is sufficient to use the service. As with NetworkTrail, all inputs that are accepted by GeneTrail2 are accepted, too. However, DTI works best if more information is provided. To be able to give treatment recommendations, the cancer type that is currently being analysed needs to be supplied by the user. If mutation data is available, the user can upload it as a VCF file. This file is annotated using the *Variant Effect Predictor (VEP)* tool [McL+10] to provide a prediction of the mutation's effect.

5.5.3 *The Main View*

DTI exposes most of the aggregated information via a table henceforth called the *main view* (Figure 5.19). Each row in the main view corresponds to a known drug target. For every drug target, the following columns are shown:

Target Name the HGNC symbol of the drug target. For each gene a link to the respective Entrez Gene [Mag+05] site is provided.

Mutations Indicates whether mutations for this gene were found. If so, the detected mutations can be accessed by hovering over the displayed symbol.

Score The degree of differential expression between the sample and the reference.

Corresponding Drugs Drugs known to target the current gene.

Analyses Further, gene-specific analyses can be performed by clicking on the displayed symbol.

Rows in the main view can be sorted in ascending as well as descending order according to the "Target Name" and "Score" column. Filtering for specific genes or drugs is possible using a search field. For each drug a link to its DrugBank page as well as a link to relevant PubMed [Med97] entries is available. Drugs can be shown or hidden based on whether they are cancer relevant, inhibiting, vitamins or illicit. If the investigated sample is a tumour sample, DTI highlights drugs recommended by the *American Cancer Society* for the respective tumour type. Mutations in drug targets are detected and annotated using the *Ensembl Variant Effect Predictor* [McL+10].

5.5.4 *Analyses*

Besides displaying information aggregated from databases, DTI is able to perform additional analyses of the input data. For every drug target, it is possible to compute an enrichment using the Kolmogorov-Smirnov statistics (Section 4.2.4). To this end, categories extracted from the KEGG database [KG00; Kan+10b; Kan+06], which contain the drug target, are subjected to an enrichment procedure. All categories are then displayed in the side bar with an indicator for enrichment or depletion of the category. In addition, links to the relevant KEGG site are provided for each category.

Besides computing enrichments, it is possible to conduct a network analysis using the ILP formulation by Backes et al. [Bac+12] (cf. Section 5.4 and Section 3.5). Here, in contrast to the ILP presented in the original paper, the root of the network is predetermined as the query drug target. This allows to identify the most deregulated subnetwork downstream of the drug target. This information may be helpful for more reliably assessing the effect of a drug, as often drug targets are receptors acting via a signalling cascade.

5.5.5 *Integration Into Other Tools*

DTI is tightly integrated into other tools. For its target specific analyses, the functionality of GeneTrail2 and NetworkTrail is used. Furthermore,

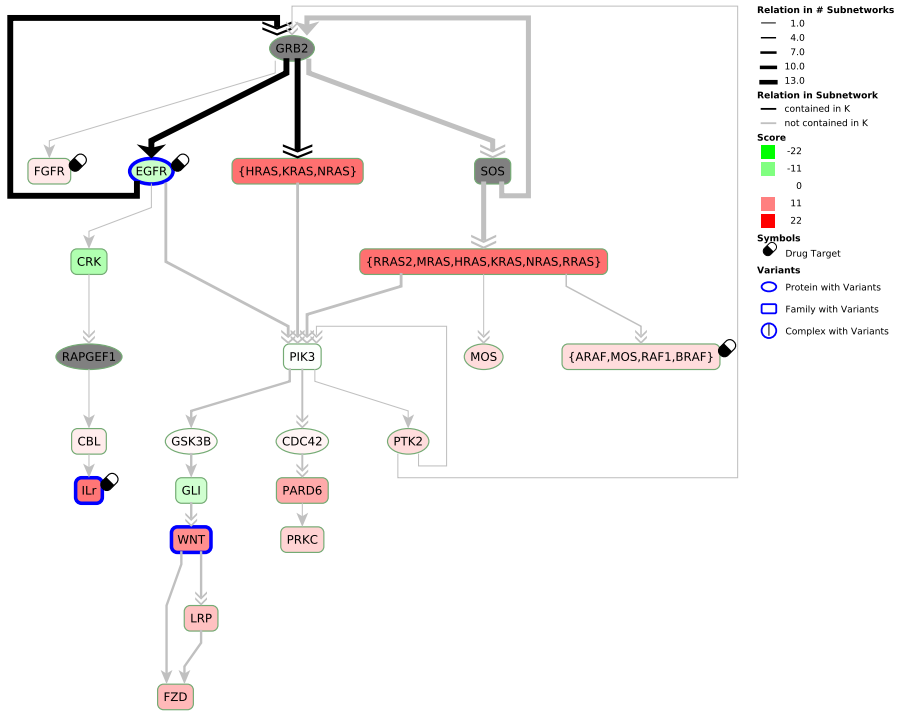


Figure 5.20: Consensus network for the deregulated subgraphs of size $k \in \{3, \ldots, 15\}$ rooted in EGFR. The network for size $k = 3$ is highlighted. The direct descendants of EGFR, PIK3 and CRK are downregulated. Ellipsoid nodes represent genes, whereas rounded rectangles represent gene families.

options are offered to analyse DTI's input directly using the aforementioned services. For visualisation, DTI uses BiNA's network visualisation and genome viewer capabilities. BiNA has been extended with a DTI specific plugin to indicate drug targets and mutations in the displayed signalling network (cf. Figure 5.20). Mutations can be examined in BiNA's built-in genome viewer. In the future, we plan to integrate the EpiToolKit service for assisting in the design of custom vaccines [Fel+08; Fel+08] into NetworkTrail.

5.5.6 *Example*

We analyse a colon adenocarcinoma dataset obtained from *The Cancer Genome Atlas (TCGA)* [Wei+13] to demonstrate the potential of DTI. To this end, we compare the sample `TCGA-AA-3542` against nine normal tissue samples using the z-score entity-level statistics (Section 4.2.2). We focused on a single sample to provide a realistic treatment optimisation scenario. Mutation information was converted from the MAF to the VCF format using the vcf2maf[12] tool.

In DTI we selected the option to only show differentially expressed drug targets for which inhibiting drugs are known. Vitamins or illicit drugs were hidden. Of the remaining drug targets only EGFR carries a

12 `https://github.com/mskcc/vcf2maf`

point mutation. Two recommended drugs, Cetuximab and Panitumumab, are targeting EGFR, which is downregulated in this sample (cf. Figure 5.19). A network analysis with DTI's NetworkTrail integration reveals that the direct descendants of EGFR, PIK3 and CRK are down regulated. As EGFR serves as an activator of both gene families, this suggests that the mutation may, in fact, be a loss of function mutation. Thus, a treatment regimen for this patient might want to forego the recommended drugs Cetuximab and Panitumumab in favour of e.g. the also recommended Bevacizumab which, amongst other proteins, targets the slightly upregulated *vascular endothelial growth factor A (VEGFA)*.

Further treatment options may include targeting highly upregulated genes such as the DNA polymerase POLB or the matrix metalloproteases MMP3 and MMP7. To this end the drugs Cytarabine and Marimastat could be administered. The analysis can be accessed at the GeneTrail2 website[13].

5·5·7 *Summary*

We presented DrugTargetInspector, a Graviton-based webservice for detecting and evaluating deregulated drug targets. DTI integrates user provided information, such as expression measurements and mutation data, with knowledge stored in databases such as DrugBank or GDSC. This is a first step towards assisting users to make informed choices for optimising e.g. cancer therapy. In this regard, the tool could also be used to assess the efficacy of recommended treatment options. "Drug repurposing" is another usage-scenario for which DTI might be employed. There, the task is to detect drugs that were designed for different disease but may be effective in the examined sample as they target a specific mutation or regulatory pattern that can be observed in the data.

13 `http://genetrail2.bioinf.uni-sb.de/drug_targets.html?session=c2761f9c-4a3a-430b-9712-8f833d320e2d&scores=10078&vcf=10080&subtype=colonandrectumcancer`

6 BALL

Structure: the arrangement of and relations between the parts or elements of something complex.

— DEFINITION BY OXFORD DICTIONARIES (2016)

In previous chapters, we focused on methods and use-cases centred around genetic variation and expression data. For example, we used biological networks (Chapter 3) to capture and explain the interplay of proteins, genes, and other biological entities. Enrichment methods (Chapter 4) allowed us to search for groups of entities that exhibit unusual patterns in their input data. DrugTargetInspector (Chapter 5) supports examining the impact of mutations affecting a drug target on the efficacy of the targeting drug. However, little information has been provided on *why* a mutation has the predicted effect. To be able to explain this, we need to consider the structural change the protein affected by the mutation undergoes.

Recent advances have shown that many RNAs also possess a catalytic function.

The genome is often termed the "blueprint" of an organism. Sticking with this metaphor, proteins can be said to constitute a major part of its "workforce". Among other tasks, they play integral parts in metabolic reactions, signal transduction, and transcriptional regulation. Furthermore they form filaments and scaffolds like the cytoskeleton that lend form and structure to a cell.

Each protein consists of one or more chains of amino acids. Each amino acid carries a residue that determines its chemical properties. Hydrogen-bond interactions within the protein backbone are responsible for the formation of secondary structures such as α-helices or β-sheets. In the watery cytosol, hydrophilic amino acids form the interface to the cellular environment, whereas hydrophobic amino acids tend to group towards the centre of proteins. These forces lead to the formation of the final, tertiary structure. As this process is driven by properties of the amino acid residues, it is fair to say that the protein's amino acid sequence is the major driver behind the protein's three-dimensional structure. This structure determines the function of the protein, as it defines with which molecules the protein can interact. In the case of an enzyme, the residues that are part of the binding pocket determine the reactions that can be catalysed by the protein.

RNA structure is also highly important. However, it is much less rigorously studied than protein structure.

The importance of the 3D structure has led to the development of visualisation tools that allow experts to assess the properties of a given protein. In fact, some of the first bioinformatics tools and applications of computer graphics were programs for the visualisation of molecules. Since the publication of the first, high-resolution structures of myoglobin [Ken+58] and hæmoglobin [Per+60] the number of available protein structures has grown exponentially. During the same time, the molecular weights and resolutions of the resolved structures have increased tremendously. In addition, computers have grown more power-

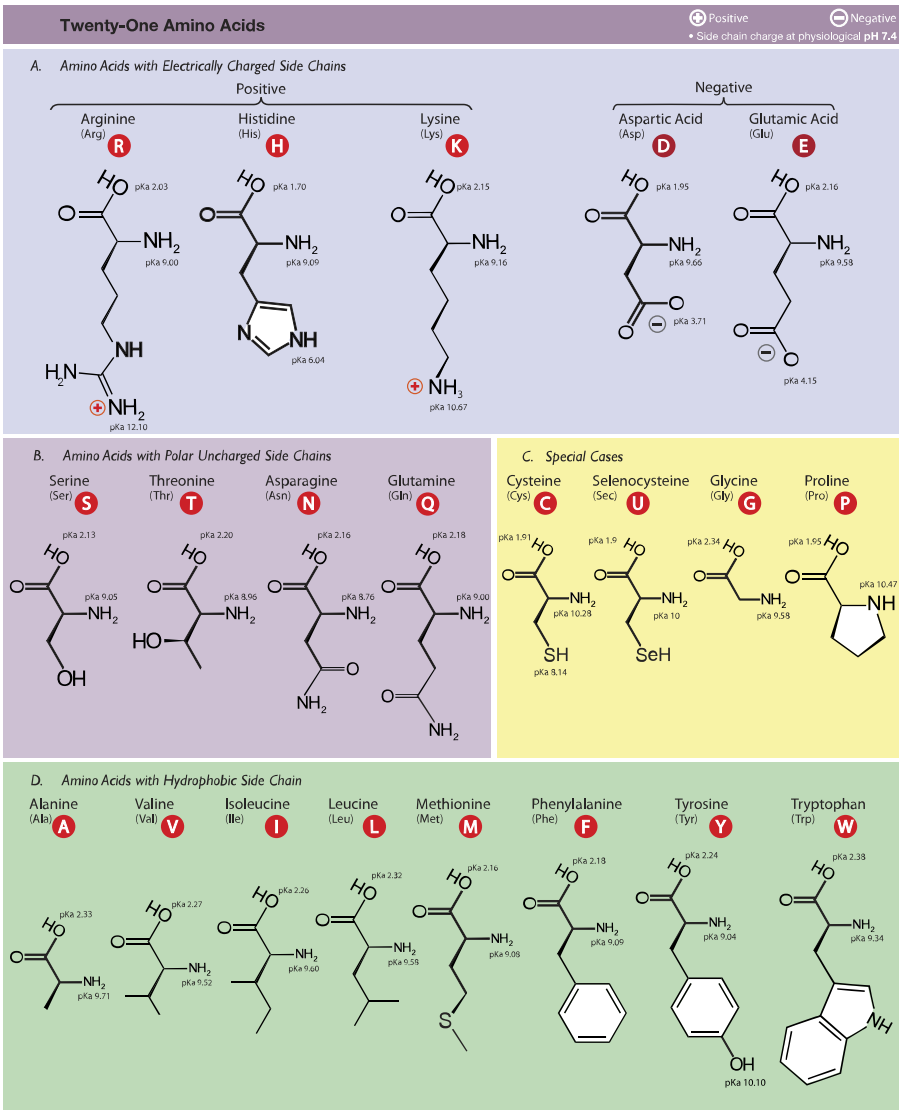


Figure 6.1: The structure of the 20 human amino acids and Selenocysteine. Typically, amino acids are classified according to their chemical properties such as hydrophobicity or charge. Image by Dancojocari [Dan16].

ful allowing to employ more advanced analyses and clearer visualisation techniques.

In this chapter, we present the *Biochemical Algorithms Library (BALL)* [KL00; Hil+10], a software package for the analysis and visualisation of molecular structures. In particular, we describe the ballaxy tool suite [Hil+14a] that allows to integrate structure-based analyses into existing workflow systems such as Galaxy [Goe+10]. In addition, we highlight the BALL-SNP application for visualising the effect of single nucleotide variants (SNVs) on protein structures. For both features, we first need to describe several, recent additions to BALL and its visualisation component BALLView [Mol+05; Mol+06]. We first start with a brief introduction of the BALL library itself and then turn to discussing BALL's plugin system that provides the underlying architecture for creating BALL "extensions". We then discuss the ballaxy suite,

which makes use of this system for interfacing with a ballaxy server. A further application of the plugin system is the PresentaBALL plugin [Nic+13]. PresentaBALL enables researchers to quickly create interactive showcases and presentations without the need to modify the BALLView source code. BALL-SNP uses the functionality provided by PresentaBALL to provide an interactive and appealing user interface.

6.1 THE BIOCHEMICAL ALGORITHMS LIBRARY

The *Biochemical Algorithms Library (BALL)* [KL00; Hil+10] is a library designed for rapid prototyping of structure-based bioinformatics tools. It is being developed by scientists from the Universities Saarbrücken, Tübingen, and Mainz. BALL is divided into two main parts: the BALL library, containing foundation classes as well as the code for working with and manipulating 3D structures, and the VIEW library that implements visualisation capabilities and user interface components. Both libraries are written in C++ and comprise around 347,000 lines of code[1]. BALL is available for the MS Windows, MacOS X, and Linux platforms. For the creation of small, custom extensions which do not warrant to write or modify C++ source code, BALL provides bindings for the Python 2 [VD09] scripting language.

BALL represents structures using a hierarchical datastructure. The root node of the datastructure is called a `System` and acts as a container for a set of molecules. `Molecules` themselves can contain `Chains` of `Residues` or simply a set of `Atoms`. This hierarchy is realised by implementing the *Composite* design pattern [Vli+95]. BALL provides parsers (and generators) for the most common structure file formats such as PDB files[2], SYBYL[3] MOL and MOL2 files, as well as HyperChem[4] HIN files. The implementation of the parsers is highly efficient and, in some cases, even beats implementations using template metaprogramming techniques [LCB10].

Due to experimental constraints, structural data is often incomplete. To enable scientists to work with 3D structures, while avoiding most of the tedious preprocessing steps, BALL offers automatic curation facilities via a database of well-known fragments. This allows to complete missing or partial residue information as well as the physically plausible placement of hydrogen atoms. For simulating molecular dynamics, implementations of the AMBER [PC03], CHARMM [Bro+83], and MMFF94 [Hal96] molecular force fields are available. For locally minimising the energy of a conformation, various efficient algorithms such as conjugated gradients [FR64] and the memory-limited BFGS method [AlB99] have been implemented. Additionally, computing electrostatic potentials using finite differences Poisson-Boltzmann [NH91] is possible. Besides molecular mechanics, geometric tools for the design of

1 `https://github.com/BALL-Project/ball`
2 `http://www.wwpdb.org/documentation/file-format`
3 `https://www.certara.com/software/molecular-modeling-and-simulation/sybyl-x-suite/`
4 `http://www.hyper.com/`

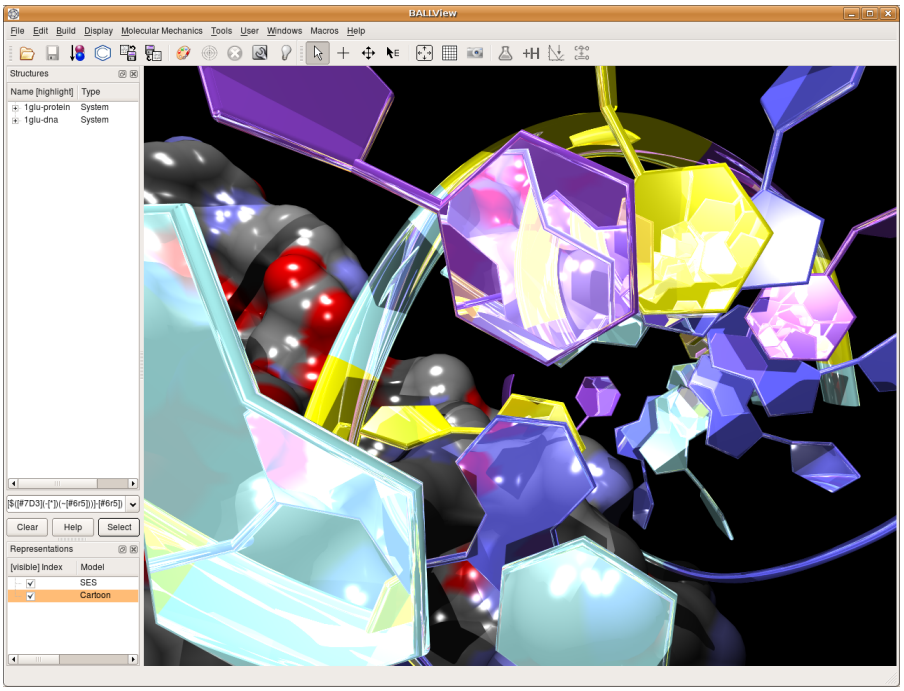


Figure 6.2: Screenshot of BALLView using the RTfact-based realtime raytracing renderer. The displayed structure is the glucocorticoid receptor (NR3C1) interacting with DNA. The crystal structure was taken from the PDB [Ber+00] record 1GLU [Lui+91]. Image by Andreas Hildebrandt.

docking algorithms [Koh12] and the analysis of docking results are provided. Examples are binding pocket detection with the PASS algorithm [BS00], procedures for determining the *solvent accessible (SAS)* and *solvent excluded (SES)* molecular surfaces [Con83], as well as an highly efficient clustering algorithm that is able to deal with large numbers of docking poses [Hil+14b].

The VIEW library contains methods for visualising structures and associated datasets, such as electrostatic fields, as well as components for building user interfaces. It is based around the concept of `Representations` that define how a structure, or a part thereof, should be visualised. Each `Representation` consists of a collection of geometric objects such as spheres, cylinders, or meshes together with colouring information. Amongst others, available representations are the SES, SAS, Ball-and-Stick, and Cartoon models. Applications built on top of VIEW, the prime example here being BALLView, are able to produce interactive, stereoscopic renderings of the loaded molecules. Publication quality images can be generated using the integrated, RTfact-based [GS08] realtime raytracing renderer (cf. Figure 6.2). Besides traditional keyboard and mouse input, virtual reality input devices are supported via an integration of the VRPN daemon [Tay+01].

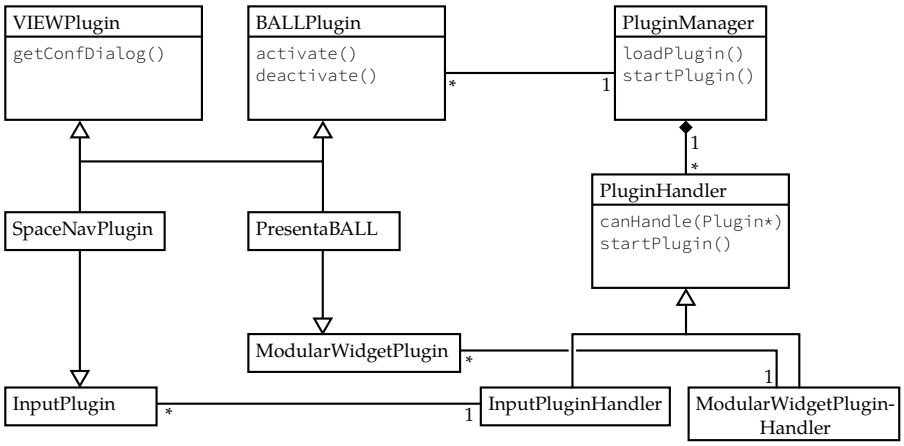


Figure 6.3: Simplified UML Diagram of the BALL plugin system. Each plugin, such as the `SpaceNavigatorPlugin` or the `PresentaBALLPlugin`, inherits from the interfaces it is going to provide. The `PluginManager` is responsible for loading plugins. Once loaded, plugins are dispatched to a `PluginHandler` instance that is specialised for the implemented plugin interfaces.

```
class BALLPlugin {
  public:
    virtual ~BALLPlugin();
    virtual QString getName() const = 0;
    virtual QString getDescription() const = 0;
    virtual bool isActive() = 0;
    virtual bool activate() = 0;
    virtual bool deactivate() = 0;
};
```

Listing 6.1: The `BALLPlugin` interface. All methods are purely virtual and must be implemented by every plugin.

6.1.1 *The BALL Plugin System*

CONTRIBUTIONS The plugin system, on which most of the presented additions are based, was designed and implemented by me. The `ModularWidgetPlugin` and various plugin implementations were created by the BALL developers.

The aforementioned integration of the VRPN daemon is not directly a part of BALL, but rather uses the exposed plugin mechanism to extend the library (Figure 6.3). A plugin is a dynamic library that can be loaded into the application at runtime. To enable this, every plugin has a well defined entry point that allows to query which functionality is offered by it. In BALL, this entry point is defined by the `BALLPlugin` interface (Listing 6.1), which allows to query the name, a description, and the status of the plugin. To provide further functionality, such as

methods that allow the query the state of an input device, a plugin needs to implement further interfaces defined in the BALL and VIEW library. Every plugin is loaded by the `PluginManager` class. It is then dispatched by the `PluginManager` to the appropriate `PluginHandler`. The task of a `PluginHandler` is to provide an interface through which the application can access loaded plugins. An example is a class managing a list of available input devices.

Currently, BALL comes with interfaces that permit to implement additional input devices (`InputPlugin`) and for adding new *graphical user interface (GUI)* components (`ModularWidgetPlugin`). We also created a `RendererPlugin` interface which makes it possible to move the rendering logic of BALLView into plugins[5]. In turn, this allows to simplify the VIEW library considerably, while making it easier to introduce new rendering techniques. As a proof of concept, we created a new, OpenGL-based [S+09] renderer [Bür13] [6].

6.2 BALLAXY

CONTRIBUTIONS The ballaxy tool suite was implemented in a joint effort by researchers of the Universities Tübingen and Saarbrücken. The manuscript and the BALLView plugin was written by Anna-Katharina Hildebrandt and Andreas Hildebrandt.

Workflow systems offer a user-friendly interface for building pipelines from individual, specialised tools. In contrast to shell scripts, workflows can easily be created by researchers that have no prior training in using the command line and programming. Additionally, the created workflows are self-documenting, highly reproducible, and allow sharing within research communities. Popular examples for such systems are Galaxy [Goe+10], Taverna [Wol+13], and the commercial KNIME software [Ber+08]. A default distribution of Galaxy is well-equipped for working with sequencing data. Support for other areas of bioinformatics, however, is lacking. For structural bioinformatics only few platforms such as MoSGrid [Her+12] that offer limited workflow functionality to the user are available. To remedy this situation, we prepared a Galaxy distribution called ballaxy [Hil+14a]. It consists of a suite of command line programs with associated Galaxy tool definitions (cf. Figure 6.4). In addition we created a `ModularWidgetPlugin` that allows to use data from BALLView directly in ballaxy and vice-versa. The ballaxy suite can be installed from the BALL source code or via a Docker[7] image[8].

5 https://github.com/dstoeckel/BALL/tree/renderer_plugins
6 https://github.com/dstoeckel/BALL/tree/modern_gl_renderer
7 https://www.docker.com/
8 https://hub.docker.com/r/anhi/ballaxy/

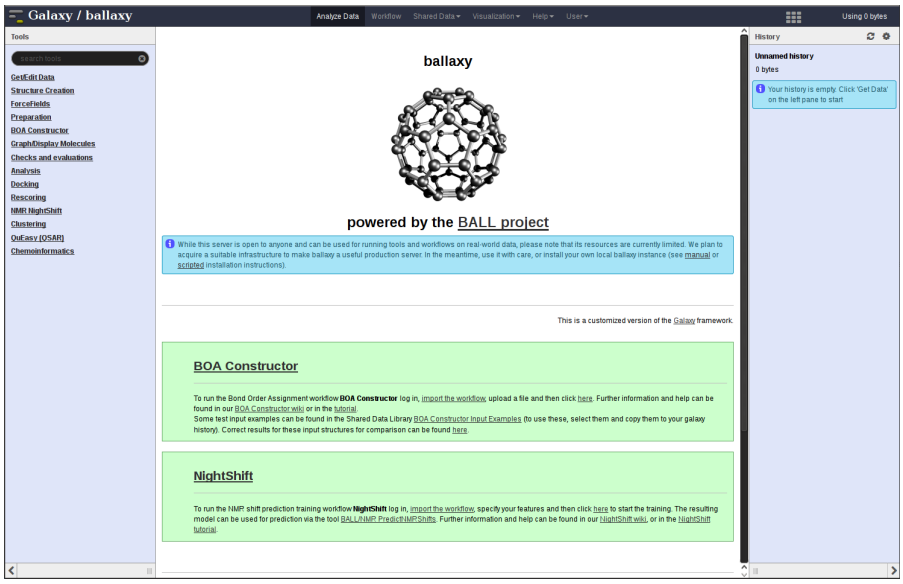


Figure 6.4: The starting page of the ballaxy web service. The system is based on Galaxy [Goe+10]. The tools can be accessed from the left panel. Data and results are tracked in the right panel.

6.2.1 *Tools*

The ballaxy tools are regular command line applications. They are being maintained in the same source code repository as the BALL library and are, thus, part of the default installation. As the goal of ballaxy is to support a wide range of usage scenarios, it comes equipped with a broad selection of general tools. Examples are file type conversion, structure creation, bond order assignment [Deh+11], or force field computations. Additionally, tools for generating output like reports or plots fall into this category. In addition, utilities for tasks like adding hydrogen atoms, separating protein chains into individual files, or removing water molecules from a `System` are provided.

The docking functionality is scheduled to be released with BALL 1.5.

These general tools are complemented by two sets of specialised applications. In particular, support for preparing, performing, and analysing docking runs and for predicting NMR shifts has been implemented [Koh12]. The docking support includes basic building blocks of docking algorithms, such as the detection of potential binding pockets and precomputing potential energy grids for the use in scoring functions. For managing and summarising the produced docking poses, ballaxy provides efficient clustering algorithms [Hil+14b]. Quick and dirty docking runs can be performed using the included multi-greedy algorithm. To more reliably filter for promising docking poses, a set of rescoring functions can be employed.

For an introduction to NMR we refer to the book by Günther [Gün13].

Besides X-Ray crystallography, *nuclear magnetic resonance (NMR)* is the primary experimental technique for revealing new molecular structures. NMR works by measuring the radiation emitted by the atoms inside a magnetic field that have been stimulated with radio waves of a specific wave length. The difference between the measured waves

and the expected, theoretical value of an isolated atom, the *chemical shift*, allows to derive distance constraints between the atoms. From these, ensembles of possible models can be determined. The inverse step, predicting NMR shifts given a structure model, also plays an important role in structural biology. Various algorithms for predicting NMR shifts exist that are based on differing methodology. Recently, hybrid methods that combine explicit formulas for NMR shift from physics with machine learning techniques have shown to provide excellent performance [Han+11]. The *NightShift* model Dehof et al. [Deh+13] further improves upon the performance of previously published methods and is included in ballaxy.

6.2.2 *BALLView Plugin*

As already outlined above, working with structure data sometimes requires to examine 3D representations. An important example is verifying that a docking algorithm produces sensible results. For this purpose, molecular visualisation tools such as RasMol [SB92; OS00], PyMOL [DeL02], VMD [HDS96], Chimera [Pet+04], and our tool BALLView [Mol+06; Mol+05] have been created. Replicating the complete functionality of one of these tools in a web interface is a monumental task. To combine the rich functionality of BALLView with the workflow interface offered by ballaxy, we implemented a `ModularWidgetPlugin` that is able to communicate with a running ballaxy instance. As the communication is bidirectional, the client can start computations on the server and the server can push data to the client. For example, it is possible to prepare a structure in BALLView. This structure can then be directly uploaded to the ballaxy server for running the workflow. Once the computation has completed, the results can again be downloaded to be visualised within BALLView.

6.3 PRESENTABALL

Contributions The PresentaBALL system has been created by me. It was later substantially improved by Stefan Nickels and Sabine Müller.

An important part of academic practice is teaching and the presentation of results. This is both, simple and difficult at the same time for structural computational biology. While renderings of protein or RNA structures are in themselves impressive, it is challenging to generate renderings that not only are aesthetically pleasing, but also easy to interpret. A reason for this is that 2D depictions of 3D structures can obscure important details due to the choice of perspective. The same is true for the choice of representation and colour, which can either help to emphasise important parts of the structure or distract the viewer

by adding "visual noise". On the other hand, interactive renderings allow users (readers) to examine a structure as they see fit. However, without additional explanatory information, users are prone to miss important details. Due to this, an interactive presentation system must allow to seamlessly incorporate such information. This, however, is not achievable with the current tools for molecular visualisation. To fill this gap we implemented PresentaBALL [Nic+13]: a system for creating interactive presentations for structure data. PresentaBALL is realised as a BALLView plugin and uses the *QtWebEngine* technology to provide HTML based annotations and explanatory texts. Custom operations written in C++ and Python can be triggered via special hyperlinks. This allows to couple explanatory texts with representational changes such as changes of perspective or model parameters.

In 2011, PresentaBALL has been deployed in the educational *MS Wissenschaft* project funded by the *German Ministry for Education and Science (BMBF)*. Furthermore, PresentaBALL is the basis of the BALL-SNP application, which we will now discuss in detail.

6.4 BALL-SNP

CONTRIBUTIONS BALL-SNP was created by Sabine Müller. The paper was written by Sabine Müller and Andreas Keller. I contributed the Windows port of the software.

Single nucleotide polymorphisms are SNVs that occur throughout a subpopulation. As such, SNPs are usually heritable whereas SNVs commonly remain "private".

Mutations play a crucial role in the characterisation of hereditary diseases and the development of cancer. An important class of mutations are *single nucleotide variants (SNVs)* which only change a single base in the genome. The effects of a SNV can vary largely. Due to the robustness of the genetic code, an exchanged nucleotide can simply result in another codon of the same amino acid. In these cases the SNV has no effect at all and is called *synonymous SNV (sSNV)*. The effect of *non-synonymous SNVs (nsSNVs)* can be more dramatic. If exchanging the nucleotide results in a codon for a different amino acid, a slightly altered protein is created. This alone is sufficient for the development of serious, hereditary diseases like sickle cell anæmia. There, the exchange of a glutamic acid (E) residue into a valine (V) residue leads to the formation of deformed, sickle-like red blood cells [Ing57; Pau+49]. Such SNVs are called *missense* mutations. By exchanging a nucleotide, also a stop codon can be created. This causes the truncation of the peptide chain. Due to this, the resulting proteins often do not share any structural resemblance or function with the wild type protein. Such SNVs are referred to as *nonsense* mutation. In the remainder of this discussion we focus on non-synonymous SNVs which we will, for the sake of simplicity, refer to as SNV.

For assessing the effect of a missense mutation on protein coding genes, it is important to know which parts of the structure are affected

by the mutation [WM01]. Especially mutations in highly conserved regions, which are important for maintaining the structure of the protein or are a part of the protein's active site, are likely to have a significant impact on the function of the protein [MK01; NH06]. For predicting the effect of a SNV, various packages that use statistical learning methods or consult databases of known SNVs exist. Examples are the *Variant Effect Predictor* [McL+10], SnpEff [Cin+12], and ANNOVAR [WLH10]. However, as these tools operate on the sequence level, the interpretation of the prediction is difficult. If multiple SNVs accumulate in the same protein, structure information becomes crucial for investigating potential, synergistic effects. Visualising the mutated residues in a three-dimensional representation can facilitate the interpretation of SNV data and allows to quickly check for the presence of interactions between SNVs. To this end, we developed BALL-SNP: a tool for the visualisation of SNVs in protein structures [Mue+15].

6.4.1 *Implementation*

BALL-SNP integrates structure data available from the PDB [Ber+00] with knowledge about non-synonymous SNVs. To display this information, it uses the PresentaBALL (Section 6.3) framework which allows to combine a structural representation of the protein with pathogenicity information. This information is derived from annotation databases such as SwissProt/UniProt [Bai+05] and the dbSNV [She+01] based ClinVar [Lan+14]. Annotation is performed using the ANNOVAR [WLH10] package. As input, BALL-SNP accepts the widely used *variant call format (VCF)*. Alternatively, the user can provide input for BALL-SNP in a simple, tab-separated file format. Protein structures are downloaded from the PDB using the largest structure referenced in the corresponding UniProt annotation. Alternatively, the desired PDB record can be entered manually.

As is illustrated by the example of sickle cell anæmia, a SNV can impact the structure of a protein. To account for this, BALL-SNP provides facilities to compute putative changes in protein stability. For this a web service running the I-Mutant 2.0 software [CFC05] is provided. Similar to mutations that impact protein stability, mutations in the active site of the protein can be highly relevant for pathogenicity. In order to help the user to find these binding pockets, BALL-SNP uses BALL's implementation of the PASS algorithm for binding pocket detection [BS00]. To detect groups of potentially interacting SNVs, a cluster analysis of the SNVs can be performed based on the euclidean distance and average linkage clustering. This is especially useful for identifying cooperative mutations.

6.4.2 *Example*

To illustrate the capabilities of BALL-SNP, we perform an analysis of SNP data from a *dilated cardiomyopathy (DCM)* high throughput sequen-

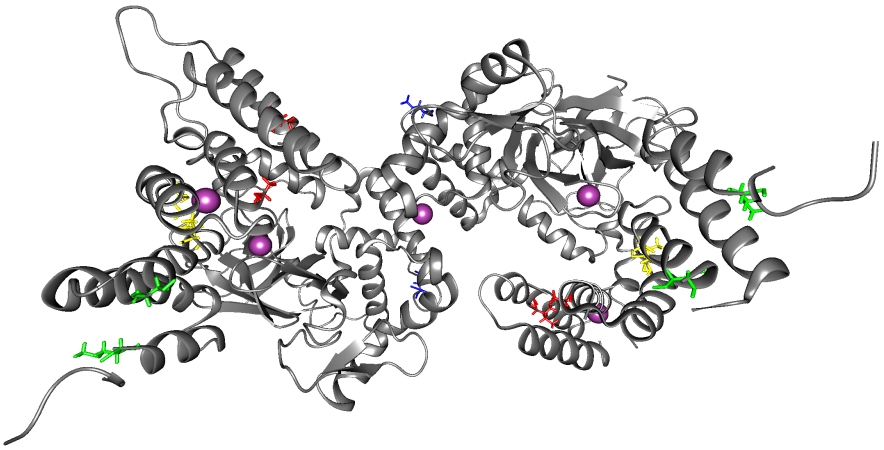


Figure 6.5: Example of a BALL-SNP cluster analysis for the SMYD2 gene. The structure of the protein is displayed as a dark-grey cartoon model. SNVs are displayed as stick models of the exchanged amino acids. SNVs belonging to the same cluster are coloured identically. Pink spheres indicate the computed centers of binding pockets. Here, the clustering for a cut-off of 16 Å is shown. As the crystal structure is a homodimer, the clusters are visible twice.

cing study conducted by Haas et al. [Haa+15]. DCM is a heart disease that is responsible for 30 % to 40 % of heart failures in large clinical studies [Haa+15]. It is characterised by the dilation of the left or both ventricles and is accompanied by impaired muscle contractility [DF94]. The cause of DCM is, as of yet, unknown.

Here, we examine SNV clusters in the SMYD2 gene, which is responsible for the methylation of lysines in H3 histones as well as transcription factors such as RB1 and TP53 [Bro+06; Hua+06; Sad+10]. As input we use the mutation data provided on the BALL-SNP homepage[9]. None of the SNVs in the input data is annotated with pathogenicity information. However, when examining the computed clusters (Figure 6.5), it can be seen that several SNVs form pairs that may be indicative of a synergistic effect. In Figure 6.6, close-ups of the SNV clusters are shown. The Cα atoms of the SNVs Y370C and M384V are $\approx$ 9 Å apart and are predicted to decrease protein stability. The remaining pairs G394C and I430M as well as V301I and V349A are farther apart with a distance of 12 Å and 16 Å, respectively. Intriguingly, in each pair one SNV leads to an increase in protein stability, whereas the other SNV leads to a decrease.

6.5 SUMMARY

To understand the effect of mutations at a molecular level, it is important to understand their effect on protein structure. To this end, tools

9 `http://www.ccb.uni-saarland.de/software/ballsnp/`

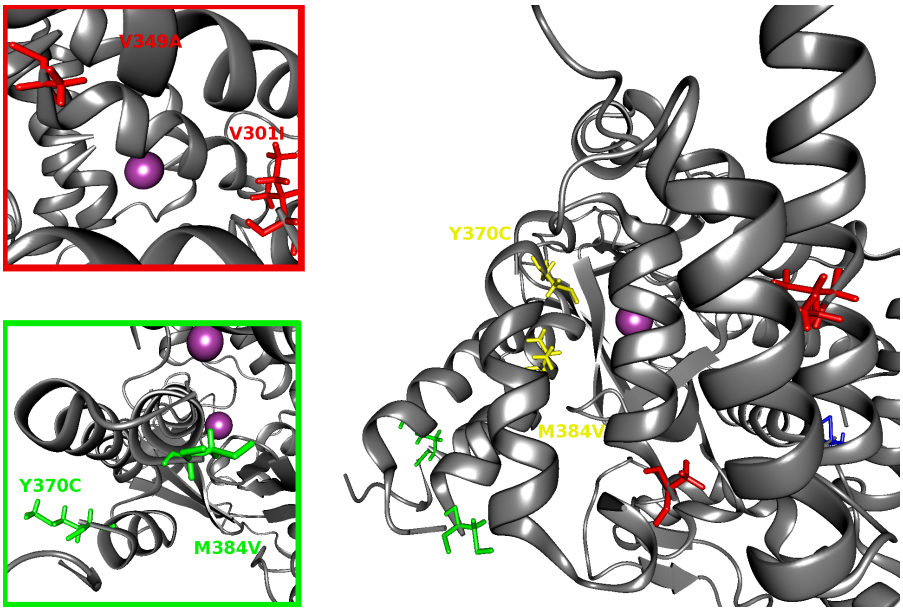


Figure 6.6: SNV pairs in the SMYD2 protein. Coloured frames indicate close-ups. SNVs belonging to the same cluster are coloured identically. Pink spheres indicate the centere of a computed binding pocket. The distances are 9 Å for Y370C and M384V, 12 Å for G394C and I430M, and 16 Å for V301I and V349A.

for the analysis and visualisation of structures are needed. Here, we presented the BALL library together with several recent additions such as PresentaBALL, ballaxy, and BALL-SNP. With PresentaBALL we conceived a system for quickly creating appealing presentations and showcases featuring structure data. The ballaxy tool-suite allows to integrate structure-based analyses into bioinformatics workflows. Finally, BALL-SNP allows to assess genetic variation data, obtained from e.g. high-throughput sequencing experiments, in the light of their impact on protein structure. In this regard, the program is a first step towards the integrative analysis of high-throughput data. For example, sets of interesting, mutated genes can be determined via the DrugTargetInspector (Section 5.5) and the GeneTrail2 (Section 5.3) web services. Individual genes can then be examined more closely using the methods provided by BALL-SNP. To make this more user friendly, the BALL-SNP pipeline could be directly coupled with the web-services using the workflow systems provided by ballaxy and Graviton.

7 CONCLUSION

There is no real ending. It's just the place where you stop the story.

— FRANK HERBERT, INTERVIEW WITH PROF. WILLIS E. MCNELLY (1969)

Advances in research concerning complex, heterogeneous diseases like cancer have shown that in many cases detailed genomic knowledge is required for devising optimal treatment regimens (Section 2.2). Thanks to rapid developments in biological data acquisition techniques, it is nowadays possible to create comprehensive profiles of the genome, transcriptome, and the proteome (Section 2.3). To help with the interpretation of the produced results, we created a range of methods and applications that allow to analyse these high-throughput datasets. Instead of creating a set of independent tools, we focused on approaches that naturally complement each other with the goal of enabling the integrative analysis of multi-omics datasets. In this chapter, we review the presented work and provide a short discussion during which we highlight challenges and opportunities for future research.

7.1 SUMMARY

Biological processes are often represented in the form of biological networks. Accordingly, approaches for analysing these networks are important tools for the identification of potential pathogenic processes. Chapter 3 thus introduced methods, such as our application *CausalTrail* [Sch15a; Stö+15], for studying biological networks. Particularly, it allows to examine causal relationships between a set of biological entities using *causal Bayesian networks (CBNs)* and the do-calculus. For instance, given a regulatory pathway, CausalTrail can be used to evaluate the impact of inhibiting a regulator with a drug on downstream elements. Whereas many tools for determining the structure of a CBN exist, CausalTrail is, to the best of our knowledge, the first, freely available tool for evaluating causal queries on an existing topology. We illustrated the importance of supporting proper causal queries using a regulatory network derived by Sachs et al. [Sac+05].

Next, we presented our ILP formulation [Bac+12] for detecting deregulated subgraphs in regulatory networks (Section 3.5). As opposed to previous work, which relies on heuristics, we proposed an efficient and exact algorithm for solving the *rooted maximum-weight connected subgraph* problem on a directed graph. Up until now, little to no research was available on how biases in the network structure affect the computation of deregulated subgraphs. Using our ILP, we implemented a sampling and machine-learning-based scheme that allows to investigate whether certain subgraphs are selected more often than oth-

ers given non-informative scores (Section 3.5.3). Based on this we discovered that some nodes are far more likely to be selected as part of a deregulated subgraph than other nodes, solely due to their topological properties. This work constitutes a fundamental step towards assessing the significance of deregulated subgraphs.

Unfortunately, the applicability of network-based methods is limited as current networks are still lacking entries for many biological entities as well as the interactions between them. Instead, it is often necessary to resort to set-based approaches such as enrichment methods (Chapter 4). These algorithms allow to quickly detect deregulated categories of biological entities in high-throughput data. To make state-of-the-art enrichment analyses available to all researchers we designed the *Graviton* platform (Chapter 5): a framework for the creation of bioinformatics web-services. The platform offers code for solving common problems in bioinformatics such as file parsing and identifier mapping. Moreover, we implemented a generic `Job–Resource` model that allows creating arbitrary, self-documenting workflows (Section 5.2). In addition, a RESTful API (cf. Section 5.1.8), with which analyses can be controlled from within the user's preferred scripting environment, is automatically generated for algorithms that are realised as a Graviton `Job`.

On top of this platform, we created *GeneTrail2* [Stö+16] the, at the time of writing, most comprehensive web service for enrichment analysis (Section 5.3). For a seamless user experience, GeneTrail2 provides methods for data preprocessing, a large range of integrated categories, and support for multiple omics types. We demonstrated how GeneTrail2 can be instrumental in the analysis of multi-omics datasets and identified pathogenic pathways that may be responsible for the malignancy of blastemal subtype Wilm's tumours (Section 5.3.5). In particular, our findings indicate that blastemal WTs have reverted to a stem cell-like state which is maintained by a feedback loop between IGF2 and the master regulator TCF3. Furthermore, WNT signalling may be stabilised by *RSPO1* which is secreted into the extra cellular matrix.

To complement GeneTrail2 and make the power of approaches for detecting deregulated subgraphs (Section 3.5), such as our ILP formulation [Bac+12] and the FiDePa algorithm [Kel+09], available in the form of a web interface, we created the NetworkTrail [Stö+13] web service (Section 5.4).

A further feature of Graviton is that services implemented on top of it can be trivially integrated with each other; a property we exploit for our *DrugTargetInspector (DTI)* [Sch+15] web service, which builds on the capabilities provided by GeneTrail2 and NetworkTrail (Section 5.5). Given an expression dataset and, optionally, genetic variation data, DTI creates an overview of deregulated and mutated drug targets. This information can be used to assess the efficacy of potential cancer treatment options. To this end, knowledge from a wide range of databases has been integrated into DTI. This data is made available to the user by condensing it into a concise report for the analysed sample. From there, additional, in-depth information and analyses can be accessed with a single click. Internally, this is achieved by leveraging the tight integration with GeneTrail2 and NetworkTrail. We demonstrated DTIs

capabilities using an adenocarcinoma sample from TCGA for which we could detect a loss-of-function mutation in the EGFR gene. This suggests that some recommended drugs targeting EGFR may not be effective in this case and should be foregone in favour of other options.

Unfortunately, determining the effect of genomic variations solely from sequence information is difficult. To be able to make better predictions, we implemented *BALL-SNP* [Mue+15], a tool for visualising and analysing SNVs in a structure context (Section 6.4). For each SNV its malignancy and its effect on the stability of the affected protein can be predicted. Possible collaborative SNVs can be detected using the built-in cluster analysis. For constructing BALL-SNP we relied on the functionality provided by the *Biochemical Algorithms Library (BALL)* [KL00; Hil+10] maintained by the universities Saarbrücken, Tübingen, and Mainz. To make BALL's features available to a more diverse group of users, we developed technologies that increase the flexibility of the library. Examples for this are facilities such as *PresentaBALL* [Nic+13], which allows to create interactive presentations (Section 6.3), and the *ballaxy* [Hil+14a] suite, which integrates BALL into the *Galaxy* [Goe+10] workflow system.

7.2 DISCUSSION

The methods outlined in this thesis allow to gain a deeper insight into the physiological processes that take place in a tumour, as is illustrated by our Wilm's tumour study. In particular, our methods enable researchers and physicians to perform analyses on multiple levels of detail. Using GeneTrail2 or NetworkTrail, it is possible to quickly identify relevant pathological mechanisms. DTI uses this information to enable the discovery of promising treatment options for a specific tumour. To this end, drug targets carrying a SNV can be examined on the structure level using BALL-SNP. This allows to gauge whether the targeting drug will actually be effective. Finally, if the detected target is part of a regulatory cascade, it is possible to use CausalTrail to model the response of downstream elements to e.g. a knockout.

Naturally, for each presented method many details can be improved and extended in several ways. CausalTrail can be enhanced with support for continuous random variables and more robust parameter fitting procedures by leveraging shrinkage techniques. Furthermore, we can expand GeneTrail2, NetworkTrail, and DTI with additional methods, add more supported data formats, and provide better heuristics and guidelines for determining optimal parameter settings. BALL and BALLView can be extended with additional algorithms and advanced visualisation capabilities, respectively.

Each of these additions may significantly improve the flexibility and applicability of the discussed methods and may advance the state-of-the-art of the respective field of computational biology. However, we also believe that it is of utmost importance to increase the interoperability of available approaches to enable examining multi-omics datasets. Thus, the major theme of this work has been the attempt to bridge

gaps between methods to facilitate such integrative analyses. With the creation of the Graviton framework, we made a significant contribution in this regard. In particular, the support for a wide range of input formats and datatypes as well as utilities such as the comparative enrichment view, help to perform joint evaluations of e.g. expression, protein abundance, and genomic variation data.

As integrative methods require to explicitly take the relationships between biological entities into account, network-based methods are a natural choice as an algorithmic basis. Currently, interpreting the results of, for instance, methods for detecting deregulated subgraphs is difficult, though. This is due to the fact that it is unknown how to compute the significance of the obtained subgraphs. To chart the terrain in this direction, we provided a scheme for quantifying the likelihood with which a subgraph is selected given non-informative node scores. The presented results (Section 3.5.3) suggest that there is a pressing need for a rigorous framework for assessing the significance of deregulated subgraphs that helps to ensure the interpretability of these methods.

For an application in a medical setting, methods that allow to jointly analyse multi-omics data are only a part of the puzzle. To assist in choosing an appropriate treatment for a tumour, assistance platforms are needed that support the clinician in every step of the decision making process. To achieve this, the tools should, on the one hand, aggregate and visualise relevant information from external databases and, on the other hand, allow their users to trigger further, in-depth analyses. DTI and BALL-SNP are examples for how such assistance platforms could be structured. They allow to study genomic data in combination with expression profiles and protein crystal structures, respectively. Ideally, the functionality of both applications would be merged to enable workflows where potential drug targets carrying mutations can be identified and seamlessly analysed on the structure level.

Bioinformatics methods have shown that they can provide valuable information for combatting heterogeneous diseases. A prime example is HIV therapy optimisation for which platforms such as Geno2Pheno [Bee+03; TL12] are routinely used in practice. For complex, heterogeneous diseases like cancer, which have proven to be even more elusive than HIV, integrative methods may give physicians and researchers the means to devise effective, personalised treatments. However, until this is the case, the task of making these tools reliable enough to be applicable in practice, will provide the grounds for a wide range of exciting and challenging research.

A MATHEMATICAL BACKGROUND

If $u = f_1(x, y)$ and $v = f_2(y, z)$ be two functions of the three variables x, y, z, and these variables be selected at random so that there exists no correlation between x,y, y,z, or z,x, there will still be found to exist correlation between u and v. Thus a real danger arises when a statistical biologist attributes the correlation between two functions like u and v to organic relationship.

— KARL PEARSON, ON A FORM OF SPURIOUS CORRELATION ... (1896)

The idea behind many of the methods presented in this thesis can be appreciated without a deep mathematical foundation. Nevertheless, the employed concepts are rooted in probability theory and statistics. Controversies around the proper use of the p-value show that a good grasp of the underlying theory is, in fact, necessary to work with many bioinformatics tools. Here, we lay the mathematical foundations necessary for understanding the concepts used in this thesis.

A.1 COMBINATORICS AND PROBABILITY THEORY

The field of probability theory is a fundamental part of mathematics that concerns itself with the study of systems that show random behaviour. The discovery of probability theory dates back to the 17th century. It was conceived by the, back then, dominant French mathematical community and was mainly applied to the study of games of dice, cards, or roulette. The foundations of probability theory were laid by the mathematician Blaise Pascal in close collaboration with Pierre de Fermat and Christiaan Huygens [Sha93]. In his book *Ars conjectandi* Bernoulli [Ber13] formalised much of combinatorics and probability theory, most notably giving a proof of a first version of the *law of large numbers*. In the 18th century significant contributions were made by Pierre-Simon Laplace, who conceptualised much of the Bayesian view of probabilities as well as a first, analytic view of probability theory. Carl Friedrich Gauß' close examination of the normal distribution and the method of least squares showed that probability theory can be applied to correct for uncertainties in measurements [Abb71].

With his famous wager Pascal cast the decision of whether to believe in god's existence or not into a probability theoretic setting: believing that god exists is an outcome with small probability but infinite gain, thus making it a "safer bet" than the alternative.

In this section, we will give a short introduction to probability theory that is tailored towards the needs in bioinformatics research in general and the analysis of biological high-throughput data, such as microarray experiments, specifically. We first start with the basic concepts of a probability space and random variables.

A.1.1 *Probability Spaces*

In order to be able to talk about probabilities of some events, a formal definition of an observable event is required. As a next step, a way to express how probable it is to observe such a result is needed. This will lead us to the definition of a *probability space*.

Consider a simple random experiment where a six-sided dice is rolled once. We can now ask how likely it is that we roll a 6 or that the number shown is even. Whereas "rolling a 6" corresponds to only observing the *outcome* 6, "rolling an even number" corresponds to observing the outcomes 2, 4, or 6. Observable *events* can thus consist of individual or multiple outcomes. This means that the set of events Σ is a subset of the power set $\mathcal{P}(\Omega)$ of the set of outcomes Ω. In addition to that, some additional constraints need to be fulfilled:

Definition 21 (σ-Algebra)**.** Let Ω be a set. A set $\Sigma \subseteq \mathcal{P}(\Omega)$ is called a σ-algebra if it fulfils the following conditions:

1. $\emptyset \in \Sigma$
2. Closed under complement: $A \in \Sigma \rightarrow \Omega \setminus A \in \Sigma$
3. Closed under countable unions: $A_1, A_2, \ldots \in \Sigma \rightarrow \bigcup_i A_i \in \Sigma$

For our dice example we have $\Omega = \{1, 2, 3, 4, 5, 6\}$ and $\Sigma = \mathcal{P}(\Omega)$.

The definition of a σ-algebra ensures that our events have some convenient properties. Given an event A, there is a well defined complementary event A^c such that either A or A^c takes place. Furthermore, given the events A, B, we can ask whether both happened ($A \cap B$) or at least one of both happened ($A \cup B$). This is possible as both, the intersection as well as the union, of two events are guaranteed to be events again.

Having defined the space of events, we now need a way to assign probabilities to each of the events.

The 2nd condition is not strictly necessary as it follows from $\Pr(A_i) = \Pr(A_i \cup \emptyset) = \Pr(A_i) + \Pr(\emptyset)$

Definition 22 (Probability Measure)**.** Let Ω be a set of outcomes and Σ a corresponding σ-algebra of events. A probability measure on Σ is a function $\Pr : \Sigma \rightarrow [0, 1]$ that fulfils the following conditions:

1. $\Pr(\Omega) = 1$
2. $\Pr(\emptyset) = 0$
3. Let $\{A_i\} \subset \Sigma$ be a countable set of pairwise disjoint events, then

$$\Pr\left(\bigdot\bigcup_i A_i\right) = \sum_i \Pr(A_i)$$

We are now ready to define a *probability space*, which ties the concepts of outcome space, event space, and probability measure together.

Definition 23 (Probability Space)**.** A probability space (Ω, Σ, p) is a triple where Ω is the set of possible outcomes, $\Sigma \subseteq \mathcal{P}(\Omega)$ is a σ-algebra on Ω, and $p : \Sigma \rightarrow [0, 1]$ is a probability measure.

A random experiment is completely described given its probability space. Spelling out the probability space explicitly, however, can be downright impossible. Fortunately, explicit knowledge of the probability space is often unnecessary, as most of the time we are interested in quantities *derived* from the outcomes as well as long-time *average* behaviour of a random system. An example for this are wins and losses during gambling. The primary interest of a gambler lies not in which side of a dice is facing upwards, but rather in how much income a particular roll translates. Biological systems are real life instances of probabilistic systems for which it is infeasible to state the probability space explicitly. To reason about them, researchers design experiments which produce indirect readouts such as fluorescence intensity or staining patterns that serve as proxies for unobservable random processes. For example, gene expression values measured using microarrays or RNA-seq (Section 2.3) serve as proxies for a set of complex binding events and the transcriptional activity of the RNA polymerase II. In probability theory, such derived quantities are modelled via the concept of random variables:

The definition of a random variable can be extended from $\mathbb{R}$ to any measurable space.

Definition 24 (Random Variable)**.** Let $(\Omega, \Sigma, \Pr)$ be a probability space. A function $X : \Omega \to \mathbb{R}$ with the property $\{\omega \in \Omega : X(\omega) \leq x\} \in \Sigma$ for every $x \in \mathbb{R}$ is called a *random variable* [GS01].

As a convention, upper-case letters X are used for denoting random variables, while lower-case letters x stand for the value obtained by evaluating the random variable.

Using random variables, we are now able to answer the question of how likely it is to make a profit during a game of chance. We write $\Pr(X \leq x)$ as shorthand notation for $\Pr(\{\omega \in \Omega : X(\omega) \leq x\})$. The likelihood of losing money during gambling is then $\Pr(X \leq 0)$. We can also compute the probability of an outcome falling into an interval using the convention

$$\Pr(x_a < X \leq x_b) := \Pr(X \leq x_b) - \Pr(X \leq x_a)$$

Evaluating $\Pr(X \leq x)$ is so common in statistics that it deserves its own name:

Definition 25 (Cumulative Distribution Function.)**.** Let $(\Omega, \Sigma, \Pr)$ be a probability space and $X : \Omega \to \mathbb{R}$ a random variable.

$$F_X(x) := \Pr(X \leq x)$$

is called the *(cumulative) distribution function (CDF)* of X.

There are "artificial" probability distributions, such as the Dirac δ distribution that do not exhibit this problem.

Until now we silently assumed that the number of outcomes was *countable*. This allowed to assign a probability to each individual outcome and obtain a sensible definition of the probability measure by simply applying its axioms. This, however, does not work for probability spaces with an uncountable number of outcomes. Assume that each of the uncountably many outcomes was assigned a positive probability. Then the sum of probabilities would be larger than one contradicting

Here "most" means: all except countably many.

the definition of a probability measure. Hence, for most outcomes ω we have $\Pr(\{\omega\}) = 0$. This is intuitively correct, as a single, infinitesimally small point out of an uncountable set of possibilities should never be selected twice by a random process. However, when we look at intervals as opposed to single points, some intervals that are more likely to contain the values of a random variable than other intervals exist. We can take this observation to the extreme by observing smaller and smaller intervals around a point x.

$$\Pr(x - \epsilon < X \leq x) = \Pr(X \leq x) - \Pr(X \leq x - \epsilon) = F_X(x) - F_X(x - \epsilon)$$

The difference above suggests that we should be able to write F_X as the antiderivative of some function f_X. Let us make this notion concrete:

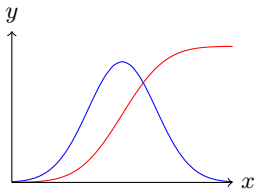


Figure A.1: The cumulative (red) and probability (blue) density function of a Gaussian normal distribution.

Definition 26 (Continous random variable)**.** The random variable X is called *continuous* if its CDF F_X can be expressed as

$$F_X(x) = \int_{-\infty}^{x} f_X(u)\,\mathrm{d}u$$

for some integrable function $f_X : \mathbb{R} \to [0, \infty)$ called the *(probability) density function* of X [GS01].

If a random variable only maps to a countable subset of $\mathbb{R}$, a simpler definition can be given:

Definition 27 (Discrete random variable)**.** The random variable X is called *discrete* if it takes values in some countable subset $\{x_1, x_2, \ldots\} \in \mathbb{R}$, only. The discrete random variable X has the *(probability) density function* $f_X : \mathbb{R} \to [0, 1]$ given by $f_X(x) = \Pr(X = x)$ [GS01].

With random variables we can describe random processes in terms of higher level statistics instead of the underlying probability spaces. In the following, two fundamental examples for such statistics are introduced: the *expected value* and the *variance*.

A.1.2 *Expected Value and Variance*

Actually, we use the average to gain information about the expected value …

Often the outcome of a single random experiment is not of interest for an investigation. For example consider a study that tries to determine the effect of a drug administered to a patient. As every patient reacts slightly differently to a drug, measuring a single response does not provide sufficient evidence that a drug is actually working. Instead, we are interested in the effect the drug has on average. In statistics this average effect is captured by the *expected value*.

Definition 28 (Expected Value)**.** Let $X : \Omega \to \mathbb{R}$ be a random variable. If the limit

$$E(X) := \int_{-\infty}^{\infty} x f_x(x)\,\mathrm{d}x$$

or, for discreet variables

$$E(X) := \sum_{i=1}^{\infty} x_i \Pr(X = x_i)$$

exists, $E(X)$ is called the *expected value* of X.

Per convention the Greek letter μ is used to designate the expected value of a random variable. Note that not every random variable possesses an expected value, as the value of the integral or the series in the above definitions does not necessarily need to be defined.

While the expected value provides information about the outcome of a random experiment "on average", it should not be confused with the most likely outcome of a random experiment. Instead, the expected value may be the result of two likely events cancelling each other. For example in a game where the player wins a set amount of money if a fair coin shows "heads" and looses the same amount of money if the coin shows "tails" the expected value is zero. To be able to get a better understanding of a random experiment a measure is needed that tells us how the results vary around the expected value. This measure is the variance:

Definition 29 (Variance)**.** Let $X : \Omega \to \mathbb{R}$ be a random variable with expected value $E(X)$. Then the *variance* $\operatorname{var}(X)$ is defined as

$$\operatorname{var}(X) := E\left[(X - E(X))^2\right]$$

We will also use σ^2 to denote the variance. The square root of the variance σ is called the *standard deviation*.

In other words, the variance measures the expected, squared deviation from the expected value. Again, the variance only makes a statement about the distribution of outcomes in the long run. It furthermore does not capture any information about the shape of the distribution with the exception of its spread. It is possible to compute higher (central) *moments* of a distribution, such as the *skewness* or the *kurtosis*, to obtain more information concerning the distributions shape. Here, we will refrain from defining these concepts, as they will not be needed in the remainder of this thesis.

The variance measures how the results of a single random variable are spread. When two dependent random variables are given, it is often of interest to examine how the variances of the two variables are related. To this end, we define the covariance of two random variables.

Definition 30 (Covariance)**.** Let $X, Y : \Omega \to \mathbb{R}$ be two random variables with expected values $E(X)$ and $E(Y)$. Furthermore, assume that $E(XY)$ exists. The *covariance* $\operatorname{cov}(X, Y)$ is defined as

$$\operatorname{cov}(X, Y) := E\left[(X - E(X))(Y - E(Y))\right]$$

Note that the covariance is a generalisation of the variance and it holds that $\operatorname{var}(X) = \operatorname{cov}(X, X)$. A commonly used statistics that can

be derived from the covariance is Pearson's correlation coefficient r [Pea95b].

$$r := \frac{\text{cov}(X, Y)}{\sqrt{\text{var}(X)\text{var}(Y)}}$$

Pearson's correlation coefficient measures the strength of the linear dependency between two variables. If $r = 0$ there is no linear dependency. In contrast, if $r = \pm 1$ then a scatter plot of the two variables would result in a perfect line. It is important to note that a correlation of 0 does not imply that the two random variables are independent. Conversely dependent variables do not guarantee that their correlation is different from 0.

A.1.3 *Populations, Samples, and Estimators*

In practice, the exact parameters of a distribution, for example the expected value or the variance, are unknown. In this case, it may be necessary to estimate the parameters from observations. The available observations for doing so are called the *population*. Populations can be countable or, at least in theory, uncountable [Ass13]. Examples for populations are all products produced by some factory or all citizens of a country. Given a population, consisting of observations $x_1, ..., x_n$ from a random variable X we can define estimators for the parameters of X. We call a function $f(x_1, ..., x_n) = \hat{\theta}$ that estimates the value of a parameter θ of the underlying distribution an *estimator*. We will use the convention to denote the estimate of some quantity q with $\hat{q}$. Given the above definition, a valid estimator for the mean of any distribution would be the constant function returning 0. Naturally, this is a terrible estimator for all intents and purposes. A good estimator for the population mean μ is given by $\mu := \frac{1}{n}\sum_{i=1}^{n} x_i$. Similarly, the population variance σ^2 is given by [Liu74]

$$\sigma^2 := \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2 = \frac{1}{n}\sum_{i=1}^{n} x_i^2 - \mu^2$$

Note that "sample" can refer to both: a set of observations and a single observation.

Sometimes, not the complete population is available. This can be the case if not all members of the population can be observed or the population is prohibitively large. In this case we need to draw conclusions from a subset of observations.

Definition 31 (Random sample)**.** Let P be a population of observations obtained from a random variable X. A subset $S = \{x_1, ..., x_n\} \subset P$ is called a random sample of P. If the observations are chosen independently of each other we call them *independent identically distributed (iid)*.

To obtain better estimators the definition of an estimator needs to be narrowed down by requiring further properties. A reasonable property is that the estimator should yield better results when we increase the sample size.

Definition 32 (Consistent estimator)**.** An estimator f for a property θ is consistent if it converges in probability to the true parameter. This means that for a sample $S = \{x_1, ..., x_n\}$ and all $\epsilon > 0$

$$\lim_{n\to\infty} P(|f(S) - \theta| \geq \epsilon) = 0$$

Another useful property is that the estimator returns the true value on average. This leads us to the definition of unbiased estimators.

Definition 33 (Unbiased estimator)**.** Let $X \sim D(\theta)$ be a random variable following some distribution D with parameters θ and furthermore let $S = \{x_1, ..., x_n\}$ be samples of X. We call a function $f(S) = \hat{\theta}$ an *unbiased estimator* for θ iff

$$E(f(S)) = \theta$$

An unbiased estimator for the population mean is given by the sample mean $\bar{x} := \frac{1}{n}\sum_{i=1}^{n} x_i$.

Proof. The proof is straight-forward and exploits the linearity of the expected value.

$$\begin{aligned} E(\bar{x}) &= E\left[\frac{1}{n}\sum_{i=1}^{n} x_n\right] = \frac{1}{n}\sum_{i=1}^{n} E(x_n) \\ &= \frac{1}{n}\sum_{i=1}^{n} E(X) = \frac{1}{n} n E(X) = E(X) \end{aligned}$$

□

Similarly the sample variance

$$\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2 \tag{A.1}$$

The $n-1$ in the denominator is due to the fact that a degree of freedom has already been "consumed" when computing $\bar{x}$.

where $\bar{x}$ denotes the sample mean, is an unbiased estimator for the population variance. We omit the proof that is no more difficult, but somewhat lengthier, than for the sample mean and refer the reader to one of the many proofs available online (e.g. Anderson [And99]).

Using unbiased estimators guarantees that the obtained estimate is eventually the true value of some parameter and is not affected by a systematic error. While generally desirable, a considerable number of samples may be required for this property to be beneficial. In the case that only few samples are available, it may pay off to use biased estimators that introduce a systematic error, while on the other hand being much less susceptible to outliers and noise in the samples. An example for such a biased estimator is the shrinkage estimator for the sample variance presented in Section 4.2.2. Intentionally introducing bias into an estimator is an instance of the bias variance trade-off, which we will investigate more closely in Appendix A.2.2.

A.2 MACHINE LEARNING

This section introduces the basic notions of statistical learning. For more details we refer to the excellent books by Friedman, Hastie, and Tibshirani [FHT09] as well as Bishop [Bis06].

Biological processes and conditions are inherently difficult to observe without disrupting or even destroying the system. This becomes even more problematic in a medical application, where invasive procedures should be avoided if possible. However, in vitro assays are not a panacea as they may consume a considerable amount of time and money to perform and, most importantly, may not allow to reproduce the desired behaviour. Statistical methods that allow to model biological processes from easily obtainable, cheap measurements are thus desirable. In many cases, theoretical models are the only way to investigate a given system.

Besides mechanistic modelling techniques that use e.g. differential equations to simulate reactions, approaches from the field of statistical learning are frequently applied. In statistical learning, data points are modelled as samples drawn from unknown probability distributions. A distinction between two tasks can be made: *supervised* and *unsupervised* learning. In a supervised learning scenario the objective is to create a model from a *training set* of samples. A training set consists of measurements associated with a *response* or *outcome*, e.g. a class label or a real number. This model is then used to predict the response of previously unseen data called the *test set*. In contrast, unsupervised learning does not rely on previously determined outcomes, but rather attempts to uncover the structure in the data itself. Examples for this are the detection of clusters of data points or associations between measured random variables.

A.2.1 *Supervised Learning*

Consider a set $X = \{x_1, x_2, ..., x_n\}$, with $x_i \in \mathbb{R}^p$ being *independent, identically distributed (iid)* samples drawn from a multivariate distribution. Each x_i is called a *sample* and each entry x_{ij} is a measurement of the j-th *feature* or *predictor*. In addition to the samples, we are given a set of responses $Y = \{y_1, y_2, ...y_n\}$. We call the set $T = \{(y_i, x_i) \,|\, y_i \in Y, x_i \in X\}$ the *training set*. We assume that y_i and x_i are related via a function f. However, after applying f to x_i, a random error term ϵ is added to the result. Our final model can hence be formulated as:

$$y_i = f(x_i) + \epsilon$$

The task is now to select a function $\hat{f} \in \mathcal{M}$ that best approximates f. Usually, we are not free to choose any $\hat{f}$ but restrict ourselves to a certain class of functions that constitute our model space $\mathcal{M}$. A common choice for $\mathcal{M}$ is the space of linear functions (cf. Section 3.4.1). We measure the approximation quality of $\hat{f}$ using a *loss-function* L. Accordingly, we select the $\hat{f} \in \mathcal{M}$ that minimises the overall loss:

$$\tilde{f} = \arg\min_{\hat{f} \in \mathcal{M}} \sum_{i=1}^{N} L(y_i, \hat{f}(x_i))$$

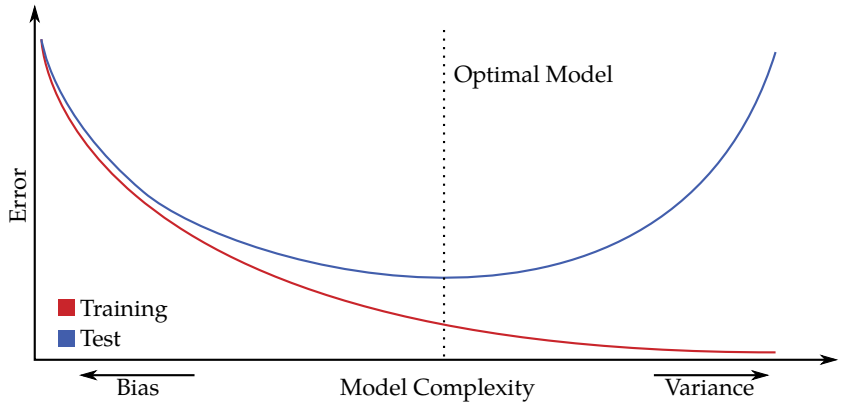


Figure A.2: Training and test error for a supervised model depending on the model's complexity. The training error generally decreases with increasing complexity due to the models ability to better capture the properties of the training set. This can have detrimental effects on the test error, as the model is no longer able to generalise well to unseen data.

However, the model that "performs best" on the test set, need not perform well on new, unseen data. To this end, the final model needs to be evaluated on a set of samples and responses that were not used for fitting the model. This set is henceforth called the *test set*. How well the fitted model is able to generalise depends strongly on the choice of model space $\mathcal{M}$ and loss function L. While a small model space may not offer sufficient flexibility to find a good model (undertraining), a model space that is large may contain functions that do not only capture the properties of the original function f, but also adapt to the random noise that stems from the error term ϵ (overtraining). The choice of loss function may push model selection towards "simple" or more "complex" models and can thus be used to avoid over- and undertraining.

A.2.2 *Bias-Variance Trade-off*

Let us examine over- and undertraining more closely. Figure A.2 depicts the qualitative behaviour of the training and test error depending on the model complexity. With model complexity or *degrees of freedom* of a model, we handwavingly denote how well the model can adapt to a given training set. In general, model complexity is hard to define, as it not only depends on offered tuning parameters and the type of model, but also on the training procedure and used features. Some theory for formalising model complexity, such as the Vapnik-Chervonenkis theory [Vap13; FHT09], exists. However, for the discussion in this thesis the intuitive definition is sufficient.

For supervised regression models, we can decompose the expected prediction error into a *bias* and a *variance* term. To this end, let us introduce the *k-nearest neighbour (k-NN)* predictor. Assume a we want to predict the outcome of a point x_0. The simplest way to do this is to

Figure A.3: Splitting the available data into independent training, tuning, and test sets. Commonly splits that achieve a ratio of 2:1:1 are used.

select the $k \in \mathbb{N}$ nearest points in the training set $N_k(x_0)$ and to assign their average outcome to x_0:

$$\hat{y}_0 = \frac{1}{k} \sum_{i \in N_k(x_0)} y_i$$

We denote the fit of the k-NN predictor with $\hat{f}_k$. Assume that $Y = f(X) + \epsilon$ with $E(\epsilon) = 0$ and $\text{var}(\epsilon) = \sigma^2$. Then the *expected prediction error (EPE)* can be written as [FHT09]

$$\begin{aligned} \text{EPE}_k(x_o) &= E[(Y - \hat{f}_k(x_0))^2 | X = x_0] \\ &= \sigma^2 + \text{Bias}^2(\hat{f}_k(x_0)) + \text{var}_T(\hat{f}_k(x_0)) \end{aligned}$$

Here var_T denotes the variance on the training set. The bias measures the systematic error incurred by the model due to its inability to adapt to the data. Conversely, for a model with high variance, small changes in the input features can lead to large changes in the output. Generally, simple models have a high bias and a low variance as they come with a large systematic error, whereas complex models have a low bias and a high variance as they are able to adapt to the random noise in the training set. Thus, to select the predictor that generalises the best to unseen data, it might pay off to select a model with a slightly larger bias if this leads to a substantial decrease in variance and vice versa.

A.2.3 *Model Selection and Evaluation*

Probably the most important topic in machine learning is proper model selection and evaluation. Finding the sweet spot of parameters that yield a good training error without overtraining is fundamental for the quality of the predictor. Here, it is of utmost importance to properly validate the model on unseen data. For this, the available data is separated into a training and a test set. If parameters need to be tuned, usually an additional tuning set, on which the error for various parameter setting is evaluated, is required (Figure A.3).

However, in many scenarios from computational biology too few samples are available to afford the luxury of a completely independent test set. To work around this issue, strategies such as *k-fold cross-validation* are commonly applied. Here, the idea is to divide the training set into k equally sized *folds*. Each of the folds is then used as a test set once, whereas the remaining $k - 1$ folds serve as the training set. An approximation of the test error is then computed as the average of the errors achieved in each fold.

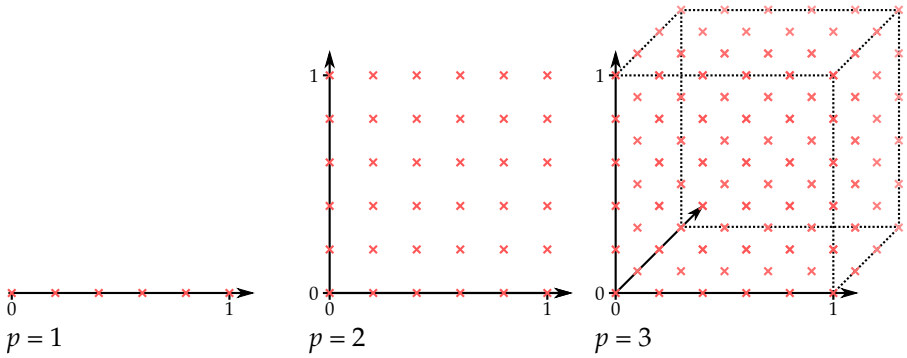


Figure A.4: The required number of data points to for achieving a uniform sampling grows exponentially with the number of dimensions p. This phenomenon is called the "curse of dimensionality".

If tuning parameters need to be fitted, it may be necessary to nest two cross-validations. The outer cross-validation computes an approximation of the test-error. The inner cross-validation again splits the current $k - 1$ folds and computes the tuning error for the parameter settings.

In addition to cross-validation several other methods for estimating the test error exist. For example, methods such as random forests use so-called *out-of-bag* samples to avoid a cross-validation (Section 3.5.3). Also, theoretical models such as the *Akaike information criterion (AIC)* or the *Bayes information criterion (BIC)* [FHT09] can be used to this end.

A.2.4 *Feature Selection*

A common problem in computational biology and statistical learning in general, is that more predictors are available than samples ($p > n$). Fitting a model under these circumstances would lead to severe overtraining as even simple, linear models would be able to predict all training points perfectly. Even if more samples are available than predictors ($p < n$), problems can arise. This is due to the phenomenon that randomly sampled data points tend to be further apart the more the number of dimensions increases. As most predictors rely on the assumption that close data points also have similar outcomes, more and more data points are required to counteract the influence of increasing dimensionality (Figure A.4). This effect is often called the "curse of dimensionality" [Bel03].

A way to avoid the above problems is to reduce the set of predictors prior to model fitting. This processes is often referred to as *feature subset selection* or simply *feature selection*. Other tactics include shrinkage methods that penalise complex models in order to use as few features as possible (cf. Section 3.4.3). It should be noted that, in any case, feature selection is part of the model fitting process and, thus, should be validated using e.g. a cross-validation.

B WILM'S DATASET - SUPPLEMENTARY TABLES

This chapter contains tables for the Wilm's Tumour dataset introduced in Section 2.4. In particular, it contains the phenotypic and clinical annotations for all samples, as well as a list of differentially expressed miRNAs.

Table B.1: A list of the collected biopsies with classification into tumour subtypes.

SIOP	WS Number	Histology	Comment
9800	WS29T	blastemal	
9821	WS38T	blastemal	
11546	WS601TB3	blastemal	
11782	WS746T1	blastemal	
11831	WS800T	necrotic	
11844	WS808TB2	blastemal	
11882	WS831T	regressiv	
11963	WS881Ta	blastemal	
11966	WS878T	regressiv	
11966	WS878Ni	normal	
11977	WS886T	regressiv	
11987	WS901T	blastemal	
11992	WS910TliB	blastemal	Origin uncertain; Excluded
11992	WS910Tre	blastemal	Origin uncertain; Excluded
11996	WS906T	mixed	
11997	WS904T	stromareich	
12009	WS914TA4	blastemal	
12015	WS917T	blastemal	
12022	WS919T	mixed	
12032	WS930T	diffuse anaplasia	
12032	WS930Trez	diffuse anaplasia	Relapse
12033	WS927T	regressiv	
12038	WS933T	regressiv	
12041	WS938T	mixed	
12041	WS938Trez	mixed	Relapse
12044	WS939T	mixed	
12044	WS939Ni	normal	Healty kidney
12055	WS954T	regressiv	
12058	WS953T	focal anaplasia	
12069	WS958T	blastemal	
12082	WS967T	regressiv	
12096	WS975T	necrotic	
12101	WS968T	regressiv	
12101	WS968Ni	normal	Healthy kidney
12112	WS994T	regressiv	
12121	WS991T	epithelial	

SIOP	WS Number	Histology	Comment
12124	WS988T	regressiv	
12125	WS1001T	regressiv	
12146	WS1002T	mixed	
12171	WS1018T	mixed	
12171	WS1018Ni	normal	Healthy kidney
12197	WS1030TA	blastemal	No WT; Excluded
12197	WS1030TB	blastemal	
12260	WS1063T2	blastemal	
12263	WS1073TA3	blastemal	
12322	WS1106TA3	blastemal	
12331	WS1098TA4	blastemal	

Table B.2: Histology of the collected biopsies. Content of blastemal, epithelial, and stroma cells is relative to the non-necrotic proportion of the tumour.

			%						Stage	
SIOP	**WS No.**	**Volume**	**Necrosis**	**Blastem**	**Epithelial**	**Stroma**	**Age [m]**	**Malignancy**	**Local**	**Global**
9800	WS29	25					55	high risk	II	II
9821	WS38	57					20	high risk	I	I
11546	WS601	25	70	100	0	0	41	high risk	I	I
11782	WS746	432	30	70	0	30	78	high risk	I	I
11831	WS800	452	100	0	0	0	101	low risk	III	IV
11844	WS808	24	20	75	5	20	8	high risk	I	I
11882	WS831	282	99	34	33	33	124	intermediate risk	III	IV
11963	WS881	565					146	high risk	II	II
11966	WS878	226	40	20	20	40	22	intermediate risk	I	I
11977	WS886	18	80				65	intermediate risk	II	II
11987	WS901	153	15	98	2	0	40	high risk	II	IV
11992	WS910	3	30				44	high risk	III	III
11996	WS906	350	30	0	50	50	35	intermediate risk	II	II
11997	WS904	9	2	10	1	89	41	intermediate risk	I	I
12009	WS914	235	15				87	high risk	III	IV
12015	WS917	100	50	90	5	5	43	high risk	III	III
12022	WS919	126	20	0	60	40	27	intermediate risk	I	I
12032	WS930	170	70	0	100	0	92	high risk	III	III

			%						Stage	
SIOP	**WS No.**	**Volume**	**Necrosis**	**Blastem**	**Epithelial**	**Stroma**	**Age [m]**	**Malignancy**	**Local**	**Global**
12033	WS927	34	75	20	70	10	64	intermediate risk	III	III
12038	WS933	200	80	10	25	65	58	intermediate risk	I	I
12041	WS938	496	40	20	40	40	4	intermediate risk	I	I
12044	WS939	197	30				37	intermediate risk	III	III
12055	WS954	330	90				29	intermediate risk	III	IV
12058	WS953	297	80	15	15	70	22	intermediate risk	I	I
12069	WS958	26	30	70	25	5	46	high risk	III	V
12082	WS967	5	90	100	0	0	45	intermediate risk	I	I
12096	WS975	70	100	0	0	0	42	low risk	I	I
12101	WS968	217	80	90	10	0	60	intermediate risk	I	IV
12112	WS994	530	75				148	intermediate risk	I	I
12121	WS991	384	5	1	99	0	6	intermediate risk	I	I
12124	WS988	5					116	intermediate risk	II	II
12125	WS1001	71	90	90			46	intermediate risk	I	IV
12146	WS1002	840	20	50	35	10	28	intermediate risk	I	I
12171	WS1018	25	20	50	50	0	51	intermediate risk	I	I
12197	WS1030	20	10				82	high risk	I	I
12260	WS1063	24	30	98			26	high risk	II	II
12263	WS1073	217	25	95	5	0	97	high risk	I	I
12322	WS1106	330	60	95	5	0	104	high risk	II	IV
12331	WS1098	168	35	80	10	10	7	high risk	I	I

SIOP	Biopsy ID	Array ID	Slide	Array
9800	WS29T	WS29T	253949426868	2_3
9821	WS38T	WS38T	253949422238	1_2
11546	WS601TB3	WS601TB3.1	253949426867	2_1
11546	WS601TB3	WS601TB3.2	253949426869	1_3
11782	WS746T1	WS746T1	253949422238	1_3
11831	WS800T	WS800T	253949422238	1_1
11844	WS808TB2	WS808TB2	253949422256	1_3
11963	WS881Ta	WS881Ta	253949422256	1_4
11966	WS878Ni	WS878Ni	253949426869	1_4
11987	WS901T	WS901T	253949422238	1_4
11992	WS910TliB	WS910Tli	253949426867	2_2
11992	WS910Tre	WS910Tre	253949422239	2_3
11996	WS906T	WS906T	253949426868	1_1
11997	WS904T	WS904T	253949422256	1_1
12009	WS914TA4	WS914TA4	253949422256	1_2
12015	WS917T	WS917T	253949426868	1_3
12022	WS919T	WS919T	253949422238	2_3
12032	WS930T	WS930T	253949426869	1_1
12032	WS930Trez	WS930TRez	253949426868	1_4
12033	WS927T	WS927T	253949426868	1_2
12038	WS933T	WS933T	253949426869	1_2
12044	WS939Ni	WS939Ni	253949426869	2_1
12058	WS953T	WS953T	253949426867	2_4
12069	WS958T	WS958T	253949422238	2_4
12096	WS975T	WS975T	253949426867	1_2
12101	WS968Ni	WS968Ni	253949426869	2_2
12101	WS968T	WS968T	253949426868	2_1
12112	WS994T	WS994T	253949422239	1_2
12121	WS991T	WS991T	253949422239	1_3
12124	WS988T	WS988T	253949426867	2_3
12146	WS1002T	WS1002T	253949426867	1_1
12171	WS1018Ni	WS1018Ni.1	253949426869	2_3
12171	WS1018Ni	WS1018Ni.2	253949426869	2_4
12171	WS1018T	WS1018T	253949426868	2_2
12197	WS1030TA	WS1030TA	253949422238	2_1
12197	WS1030TB	WS1030TB	253949422238	2_2
12260	WS1063T2	WS1063T2	253949426868	2_4
12263	WS1073TA3	WS1073TA3	253949422239	2_4
12322	WS1106TA3	WS1106TA3	253949426867	1_4
12331	WS1098TA4	WS1098TA4	253949426867	1_3

Table B.3: Mapping of biopsy IDs to mRNA array IDs.

SIOP	Biopsy ID	Array ID	Slide	Array
11546	WS601TB3	WS601TB3	253118114312	2_4
11782	WS746T1	WS746T1	253118113842	1_2
11831	WS800T	WS800T	253118113840	1_4
11844	WS808TB2	WS808TB2	253118113840	2_1
11882	WS831T	WS831T	253118113874	1_1
11966	WS878Ni	WS878Ni	253118113873	1_2
11966	WS878T	WS878T	253118113874	1_2
11963	WS881Ta	WS881Ta	253118113840	2_3
11977	WS886T	WS886T	253118113874	1_3
11987	WS901T	WS901T	253118113842	1_3
11997	WS904T	WS904T	253118113840	2_2
11996	WS906T	WS906T	253118113874	1_4
11992	WS910TliB	WS910TliB	253118114312	1_2
11992	WS910Tre	WS910Tre	253118114312	1_4
12009	WS914TA4	WS914TA4	253118113842	1_4
12015	WS917T	WS917T	253118113842	2_1
12022	WS919T	WS919T	253118113842	2_2
12033	WS927T	WS927T	253118113873	1_1
12032	WS930T	WS930T	253118113874	2_1
12032	WS930Trez	WS930TRez	253118113841	2_2
12032	WS938Trez	WS938Trez	253118113874	2_4
12038	WS933T	WS933T	253118113874	2_2
12041	WS938T	WS938T	253118113874	2_3
12044	WS939Ni	WS939Ni	253118113875	1_2
12044	WS939T	WS939T	253118113875	1_1
12058	WS953T	WS953T	253118113875	1_3
12055	WS954T	WS954T	253118113875	1_4
12069	WS958T	WS958T	253118113873	1_3
12082	WS967T	WS967T	253118113875	2_1
12101	WS968Ni	WS968Ni	253118113873	1_4
12101	WS968T	WS968T	253118113875	2_2
12096	WS975T	WS975T	253118113873	2_2
12124	WS988T	WS988T	253118113873	2_1
12121	WS991T	WS991T	253118113842	2_3
12112	WS994T	WS994T	253118113842	2_4
12125	WS1001T	WS1001T	253118113875	2_3
12146	WS1002T	WS1002T	253118113875	2_4
12171	WS1018Ni	WS1018Ni	253118113873	2_3
12171	WS1018T	WS1018T	253118113873	2_4
12197	WS1030TA	WS1030TA	253118113840	2_4
12197	WS1030TB	WS1030TB	253118113841	2_1
12260	WS1063T2	WS1063T2	253118114312	1_3
12263	WS1073TA3	WS1073TA3	253118114312	2_3
12331	WS1098TA4	WS1098TA4	253118114312	2_1
12322	WS1106TA3	WS1106TA3	253118114312	2_2

Table B.4: Mapping of biopsy IDs to miRNA array IDs.

miRNA	**p-value**	$\log_2$ **fold-change**
hsa-miR-143-3p	6.5e-06	3.3
hsa-miR-3926	2e-05	2
hsa-miR-126-3p	2.9e-05	2.9
hsa-miR-1825	4.8e-05	-2.1
hsa-miR-4290	5e-05	-2.2
hsa-miR-32-3p	5.7e-05	-3.6
hsa-miR-195-3p	5.9e-05	-2.3
hsa-miR-30b-5p	5.9e-05	2.4
hsa-miR-2278	6e-05	-3.3
hsa-miR-4299	6.1e-05	2.5
hsa-miR-1281	6.4e-05	-2.2
hsa-miR-595	6.6e-05	-3.2
hsa-miR-3148	6.7e-05	-3.3
hsa-miR-3149	6.7e-05	-3.5
hsa-miR-101-3p	8e-05	2.4
hsa-miR-1306-3p	0.00011	-2.8
hsa-miR-1180	0.00012	-2.8
hsa-miR-574-5p	0.00016	-2.6
hsa-miR-378c	0.00018	2.2
hsa-miR-670	0.00019	-2.8
hsa-miR-340-5p	2e-04	2.1
hsa-miR-130a-3p	0.00022	2.8
hsa-miR-125a-5p	0.00023	2
hsa-miR-4284	0.00023	2.3
hsa-miR-26b-5p	0.00028	2.5
hsa-miR-335-5p	0.00033	3
hsa-miR-26a-5p	0.00039	3
hsa-miR-3653	0.00039	2.5
hsa-miR-1	4e-04	2.3
hsa-miR-297	4e-04	-2.5
hsa-miR-539-5p	0.00046	-2.3
hsa-miR-10b-5p	0.00059	2.7
hsa-miR-1228-5p	0.00062	-2.9
hsa-miR-3923	0.00078	-2.2
hsa-let-7f-5p	0.00079	2.3
hsa-miR-99b-5p	9e-04	2
hsa-miR-424-5p	0.0014	2.1
hsa-miR-941	0.0018	-2.1
hsa-miR-494	0.0019	2.3
hsa-miR-125b-5p	0.0025	2.3
hsa-miR-19a-3p	0.0026	2
hsa-miR-214-3p	0.0032	2.5
hsa-miR-206	0.0035	-2.1

Table B.5: List of differentially expressed miRNAs ($p < 0.05$) in the Wilm's tumour datasets as presented in Section 5.3.5. Only miRNAs with absolute $\log_2$ fold-change greater than 2 have been selected. The reported p-values were corrected using the Benjamini-Hochberg procedure.

C

ENRICHMENT EVALUATION - RESULTS

C.1 ENRICHMENTS ON SYNTHETIC CATEGORIES

Method	Sens	Spec	AUC	Sens	Spec	AUC
Significant	33.3 %			66.6 %		
KS	0.092	0.912	0.505	0.954	0.988	0.995
weighted KS	0.07	0.935	0.495	0.634	0.99	0.962
max-mean	0.106	0.902	0.481	0.966	0.988	0.998
mean	0.014	0.985	0.498	0.082	0.989	0.641
median	0.01	0.985	0.486	0.404	0.99	0.803
sum	0.014	0.985	0.498	0.08	0.99	0.641
1s-t-test	0.006	0.992	0.5	0.004	0.99	0.52
2s-t-test	0.006	0.992	0.5	0.004	0.989	0.521
Wilcoxon	0.01	0.99	0.499	0.042	0.989	0.602
ORA	0.102	0.903	0.486	0.952	0.995	0.991

Table C.1: Performance of various set-level statistics on synthetic categories. Significantly expressed genes are distributed **symmetrically** around the mean. The used p-value strategy is the **entity-based** strategy.

Method	Sens	Spec	AUC	Sens	Spec	AUC
Significant	33.3 %			66.6 %		
KS	0.1	0.898	0.504	0.994	0.989	1
weighted KS	0.104	0.904	0.493	0.95	0.994	0.998
max-mean	0.1	0.902	0.495	0.992	0.995	1
mean	0.106	0.9	0.505	0.992	0.99	1
median	0.096	0.9	0.496	0.988	0.99	1
sum	0.106	0.9	0.505	0.992	0.991	1
1s-t-test	0.106	0.9	0.505	0.99	0.99	0.999
2s-t-test	0.106	0.9	0.505	0.99	0.989	0.999
Wilcoxon	0.106	0.902	0.508	0.99	0.988	1
ORA	0.086	0.921	0.49	0.396	0.992	0.859

Table C.2: Performance of various set-level statistics on synthetic categories. Significantly expressed genes are distributed **asymmetrically** around the mean. The used p-value strategy is the **entity-based** strategy.

Method	Sens	Spec	AUC	Sens	Spec	AUC
Significant	33.3 %			66.6 %		
KS	0.09	0.917	0.505	0.964	0.996	0.996
weighted KS	0.258	0.742	0.503	0.882	0.791	0.91
max-mean	0.956	0.04	0.513	1	0.011	0.516
mean	0.196	0.804	0.498	0.542	0.685	0.649
median	0.026	0.971	0.489	0.49	0.945	0.797
sum	0.192	0.805	0.498	0.53	0.686	0.65
1s-t-test	0.004	0.997	0.498	0	0.998	0.522
2s-t-test	0.002	0.997	0.5	0	0.999	0.52
Wilcoxon	0.006	0.996	0.5	0.01	0.998	0.616

Table C.3: Performance of various set-level statistics on synthetic categories. Significantly expressed genes are distributed **symmetrically** around the mean. The used p-value strategy is the **sample-based** strategy.

Method	Sens	Spec	AUC	Sens	Spec	AUC
Significant	33.3 %			66.6 %		
KS	0.094	0.907	0.505	0.974	0.995	0.999
weighted KS	0.874	0.133	0.516	0.974	0.104	0.56
max-mean	0.982	0.024	0.499	1	0.006	0.507
mean	0.532	0.469	0.501	1	0.505	0.754
median	0.26	0.755	0.496	1	0.54	0.944
sum	0.532	0.468	0.502	1	0.505	0.756
1s-t-test	0.532	0.473	0.506	1	0.505	0.754
2s-t-test	0.088	0.917	0.505	0.928	0.997	0.998
Wilcoxon	0.08	0.915	0.508	0.968	0.998	1

Table C.4: Performance of various set-level statistics on synthetic categories. Significantly expressed genes are distributed **asymmetrically** around the mean. The used p-value strategy is the **sample-based** strategy.

C.2 ENRICHMENTS ON REACTOME CATEGORIES

Method	Sens	Spec	AUC	Sens	Spec	AUC
Significant	33.3 %			66.6 %		
KS	0.429	0.573	0.504	0.75	0.47	0.648
weighted-KS	0.286	0.706	0.502	0.344	0.727	0.581
max-mean	0.631	0.372	0.503	0.899	0.317	0.635
mean	0.148	0.838	0.494	0.219	0.83	0.557
median	0.26	0.746	0.498	0.674	0.584	0.656
sum	0.146	0.838	0.494	0.22	0.831	0.557
1s-t-test	0.009	0.994	0.495	0.01	0.988	0.484
2s-t-test	0.009	0.994	0.495	0.01	0.988	0.484
Wilcoxon	0.048	0.95	0.489	0.085	0.929	0.538
ORA	0.619	0.376	0.503	0.936	0.292	0.644

Table C.5: Performance of various set-level statistics on the Reactome categories. Significantly expressed genes are distributed **symmetrically** around the mean. The used p-value strategy is the **entity-based** strategy.

Method	Sens	Spec	AUC	Sens	Spec	AUC
Significant	33.3 %			66.6 %		
KS	0.614	0.389	0.505	0.953	0.295	0.655
weighted-KS	0.662	0.34	0.503	0.913	0.3	0.624
max-mean	0.689	0.313	0.5	0.963	0.273	0.637
mean	0.685	0.322	0.502	0.967	0.273	0.636
median	0.64	0.369	0.5	0.968	0.285	0.638
sum	0.685	0.323	0.503	0.967	0.273	0.638
1s-t-test	0.565	0.441	0.504	0.871	0.344	0.644
2s-t-test	0.565	0.441	0.504	0.871	0.344	0.644
Wilcoxon	0.608	0.397	0.505	0.943	0.307	0.652
ORA	0.591	0.413	0.503	0.818	0.415	0.641

Table C.6: Performance of various set-level statistics on the Reactome categories. Significantly expressed genes are distributed **asymmetrically** around the mean. The used p-value strategy is the **entity-based** strategy.

Method	Sens	Spec	AUC	Sens	Spec	AUC
Significant	33.3 %			66.6 %		
KS	0.427	0.581	0.506	0.776	0.467	0.653
weighted-KS	0.448	0.57	0.504	0.541	0.553	0.577
max-mean	0.828	0.179	0.502	0.996	0.132	0.587
mean	0.341	0.651	0.495	0.514	0.573	0.563
median	0.255	0.753	0.498	0.699	0.573	0.661
sum	0.341	0.649	0.493	0.511	0.572	0.563
1s-t-test	0.001	0.999	0.495	0.001	0.999	0.487
2s-t-test	0.001	0.999	0.495	0.001	0.999	0.484
Wilcoxon	0.013	0.98	0.491	0.052	0.96	0.549

Table C.7: Performance of various set-level statistics on the Reactome categories. Significantly expressed genes are distributed **symmetrically** around the mean. The used p-value strategy is the **sample-based** strategy.

Method	Sens	Spec	AUC	Sens	Spec	AUC
Significant	33.3 %			66.6 %		
KS	0.531	0.469	0.504	0.921	0.349	0.662
weighted KS	0.746	0.266	0.5	0.924	0.227	0.581
max-mean	0.854	0.151	0.504	0.999	0.111	0.576
mean	0.823	0.181	0.505	0.999	0.138	0.591
median	0.621	0.38	0.498	0.994	0.261	0.653
sum	0.822	0.181	0.503	0.998	0.14	0.592
1s-t-test	0.588	0.417	0.5	0.847	0.277	0.559
2s-t-test	0.359	0.643	0.501	0.613	0.555	0.627
Wilcoxon	0.477	0.518	0.501	0.861	0.396	0.665

Table C.8: Performance of various set-level statistics on the Reactome categories. Significantly expressed genes are distributed **asymmetrically** around the mean. The used p-value strategy is the **sample-based** strategy.

GENETRAIL2 - TABLES

D

D.1 SUPPORTED ORGANISMS

Name	KEGG Code	Taxon ID
Anopheles gambiae	age	180454
Arabidopsis thaliana	ath	3702
Aspergillus fumigatus	afm	746128
Bos taurus	bta	9913
Caenorhabditis elegans	cel	6239
Canis familiaris	cfa	9615
Corynebacterium glutamicum	cgl	1718
Danio rerio	dre	7955
Dictyostelium discoideum	ddi	44689
Drosophila melanogaster	dme	7227
Encephalitozoon cuniculi	ecu	6035
Escherichia coli K-12	eco	83333
Escherichia coli O157:H7 str. Sakai	ecs	386585
Gallus gallus	gga	9031
Homo sapiens	hsa	9606
Macaca mulatta	mmc	9544
Mus musculus	mmu	10090
Pan troglodytes	ptr	9598
Plasmodium falciparum 3D7	pfa	36329
Rattus norvegicus	rno	10116
Saccharomyces cerevisiae	sce	4932
Schizosaccharomyces pombe	spo	4896
Staphylococcus aureus	sau	1280
Sus scrofa	ssc	9823
Toxoplasma gondii	tgo	508771
Xenopus laevis	xla	8355
Xenopus tropicalis	xtr	8364

D.2 LIST OF HUMAN CATEGORIES

Omics	Database	Category Type
GENE	GO	Biological Process
		Cellular Component
		Molecular Function
	WikiPathways	Pathways
	KEGG	Pathways
	SignaLink	Pathways
		Transcription Factor targets
	Pfam	Protein families

Omics	Database	Category Type
GENE	mirDB	Predicted targets: score > 50
		Predicted targets: score > 70
		Predicted targets: score > 90
	NIA	Phenotypes
	Reactome	Pathways
	BioCarta	Pathways
	NCI	Pathways
	SMPDB	Pathways
	PharmGKB	Pathways
	HG19 GRCh37	Cytogenic Bands
		Chromosomes
	HG19 GRCh38	Cytogenic Bands
		Chromosomes
	miRecords	Predicted targets
	miRTarBase	Validated targets
	PicTar	Predicted targets
	TargetScan	Predicted miRNA families
	DrugBank	Validated targets
	ConsensusPathDB	BioCarta
		EHMN
		HumanCyc
		INOH
		KEGG
		NetPath
		PharmGKB
		PID
		Reactome
		Signalink
		SMPDB
		WikiPathways
	miRWalk	Predicted miRNA targets: 3-UTR
		Predicted miRNA targets: 5-UTR
		Predicted miRNA targets: CDS
		Predicted miRNA targets: Promotor
		Validated miRNA targets
	Phosphosite	Diseases
	TRANSFAC	Validated miRNA targets
		Validated TF targets
		Validated TF complex targets
		Validated TF family targets
MIRNA	HMDD	Phenotypes
	miRTarBase	Targets
PROTEIN	Phosphosite	Diseases
	SMPDB	Pathways
	ConsensusPathDB	BioCarta
		EHMN
		HumanCyc

Omics	Database	Category Type
PROTEIN	ConsensusPathDB	INOH
		KEGG
		NetPath
		PharmGKB
		PID
		Reactome
		Signalink
		SMPDB
		WikiPathways
	NCI	Pathway Interaction Database
	Pfam	Protein families
	Reactome	Pathways
SNP	GWAS catalogue	
	PheWAS catalogue	

BIBLIOGRAPHY

[10010] 1000 Genomes Project Consortium. "A map of human genome variation from population-scale sequencing". In: *Nature* 467.7319 (2010), pp. 1061–1073.

[15] *ECMA-262 6th Edition, The ECMAScript 2015 Language Specification*. English. ECMA International. June 2015. URL: http://www.ecma-international.org/ecma-262/6.0/.

[16] *Total number of Websites*. English. NetCraft and Internet Live Stats. 2016. URL: http://www.internetlivestats.com/total-number-of-websites/.

[45416] Roche 454. *Roche 454 - GS FLX+ System*. July 2016. URL: http://454.com/products/gs-flx-system/index.asp.

[Abb71] Cleveland Abbe. "A historical note on method of least-squares". In: *American Journal of Science* 6 (1871), pp. 411–415.

[Aff07] Affymetrix, Inc. *Affymetrix HuGene 2.0 ST Array Datasheet*. 2007. URL: http://media.affymetrix.com/support/technical/datasheets/hugene_2_st_datasheet.pdf.

[Agn+16] Selidji T Agnandji et al. "Phase 1 Trials of rVSV Ebola Vaccine in Africa and Europe". In: *New England Journal of Medicine* 374.17 (2016), pp. 1647–1660.

[Ahm+00] Afshin Ahmadian et al. "Single-nucleotide polymorphism analysis by pyrosequencing". In: *Analytical Biochemistry* 280.1 (2000), pp. 103–110.

[AlB99] Mehiddin Al-Baali. "Improved Hessian approximations for the limited memory BFGS method". In: *Numerical Algorithms* 22.1 (1999), pp. 99–112.

[Alc+11] Nicolas Alcaraz et al. "KeyPathwayMiner: detecting case-specific biological pathways using expression data". In: *Internet Mathematics* 7.4 (2011), pp. 299–313.

[Alc+14] Nicolas Alcaraz et al. "KeyPathwayMiner 4.0: condition-specific pathway analysis by combining multiple omics studies and networks with Cytoscape". In: *BMC Systems Biology* 8.1 (2014), p. 99.

[AlH+15] Olfat Al-Harazi et al. "Integrated Genomic and Network-Based Analyses of Complex Diseases and Human Disease Network". In: *Journal of Genetics and Genomics* (2015).

[Alt+10] André Altmann et al. "Permutation importance: a corrected feature importance measure". In: *Bioinformatics* 26.10 (2010), pp. 1340–1347.

[And99] Scott D. Anderson. *Proof that Sample Variance is Unbiased Plus Lots of Other Cool Stuff*. 1999. URL: http://pascencio.cos.ucf.edu/classes/Methods/Proof%20that%20Sample%20Variance%20is%20Unbiased.pdf.

[Ang+16] Christof Angermueller et al. "Parallel single-cell sequencing links transcriptional and epigenetic heterogeneity". In: *Nature methods* (2016).

[Ant+11] Roman Anton et al. "A systematic screen for micro-RNAs regulating the canonical Wnt pathway". In: *PloS One* 6.10 (2011), e26257.

[AP13] Catherine Alix-Panabières and Klaus Pantel. "Circulating tumor cells: liquid biopsy of cancer". In: *Clinical chemistry* 59.1 (2013), pp. 110–118.

[Ara+11] Khelifa Arab et al. "Epigenetic deregulation of TCF21 inhibits metastasis suppressor KISS1 in metastatic melanoma". In: *Carcinogenesis* 32.10 (2011), pp. 1467–1473.

[ART08] ART-CC. "Life expectancy of individuals on combination antiretroviral therapy in high-income countries: a collaborative analysis of 14 cohort studies". In: *The Lancet* 372.9635 (2008), pp. 293–299.

[AS07] Adriana Albini and Michael B Sporn. "The tumour microenvironment as a target for chemoprevention". In: *Nature Reviews Cancer* 7.2 (2007), pp. 139–147.

[AS09] Marit Ackermann and Korbinian Strimmer. "A general modular framework for gene set enrichment analysis". In: *BMC Bioinformatics* 10.1 (2009), p. 47.

[Ash+00] Michael Ashburner et al. "Gene Ontology: tool for the unification of biology". In: *Nature Genetics* 25.1 (2000), pp. 25–29.

[Ass+08] Yassen Assenov et al. "Computing topological parameters of biological networks". In: *Bioinformatics* 24.2 (2008), pp. 282–284.

[Ass13] Walter Assenmacher. *Induktive Statistik*. Springer-Verlag, 2013.

[BA00] Jeffrey G Blodgett and Ronald D Anderson. "A Bayesian network model of the consumer complaint process". In: *Journal of Service Research* 2.4 (2000), pp. 321–338.

[Bac+07] Christina Backes et al. "GeneTrail – advanced gene set enrichment analysis". In: *Nucleic Acids Research* 35.suppl 2 (2007), W186–W192.

[Bac+12] Christina Backes et al. "An integer linear programming approach for finding deregulated subgraphs in regulatory networks". In: *Nucleic Acids Research* 40.6 (2012), e43–e43.

[Bac+16] Christina Backes et al. "miEAA: microRNA enrichment analysis and annotation". In: *Nucleic Acids Research* (2016), gkw345.

[Bah12] Miriam Bah. "Measuring the relevance of topology in networks for finding deregulated subgraphs". Bachelor's Thesis. Saarland University, 2012.

[Bai+05] Amos Bairoch et al. "The universal protein resource (UniProt)". In: *Nucleic Acids Research* 33.suppl 1 (2005), pp. D154–D159.

[Ban+10] Yung-Jue Bang et al. "Trastuzumab in combination with chemotherapy versus chemotherapy alone for treatment of HER2-positive advanced gastric or gastro-oesophageal junction cancer (ToGA): a phase 3, open-label, randomised controlled trial". In: *The Lancet* 376.9742 (2010), pp. 687–697.

[Bar+07a] Alessandra Baragli et al. “Heterooligomerization of human dopamine receptor 2 and somatostatin receptor 2: co-immunoprecipitation and fluorescence resonance energy transfer analysis”. In: *Cellular signalling* 19.11 (2007), pp. 2304–2316.

[Bar+07b] Artem Barski et al. “High-resolution profiling of histone methylations in the human genome”. In: *Cell* 129.4 (2007), pp. 823–837.

[Bar+12] Jordi Barretina et al. “The Cancer Cell Line Encyclopedia enables predictive modelling of anticancer drug sensitivity”. In: *Nature* 483.7391 (2012), pp. 603–607.

[Bar+13] Tanya Barrett et al. “NCBI GEO: archive for functional genomics data sets—update”. In: *Nucleic Acids Research* 41.D1 (2013), pp. D991–D995.

[Bat+04] Alex Bateman et al. “The Pfam protein families database”. In: *Nucleic Acids Research* 32.suppl 1 (2004), pp. D138–D141.

[Bav50] Alex Bavelas. “Communication patterns in task-oriented groups.” In: *Journal of the acoustical society of America* (1950).

[BCV16] Anna Bomersbach, Marco Chiarandini, and Fabio Vandin. “An Efficient Branch and Cut Algorithm to Find Frequently Mutated Subnetworks in Cancer”. In: *International Workshop on Algorithms in Bioinformatics*. Springer. 2016, pp. 27–39.

[Bea65] Murray A Beauchamp. “An improved index of centrality”. In: *Behavioral Science* 10.2 (1965), pp. 161–163.

[Bec+98] NE Beck et al. “Detection of residual disease following breast-conserving surgery”. In: *British journal of surgery* 85.9 (1998), pp. 1273–1276.

[Bec14] Elizabeth C Beckmann. “CT scanning the early days”. In: *The British journal of radiology* (2014).

[BEd08] Onureena Banerjee, Laurent El Ghaoui, and Alexandre d’Aspremont. “Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data”. In: *The Journal of Machine Learning Research* 9 (2008), pp. 485–516.

[Bee+02] Niko Beerenwinkel et al. “Diversity and complexity of HIV-1 drug resistance: a bioinformatics approach to predicting phenotype from genotype”. In: *Proceedings of the National Academy of Sciences* 99.12 (2002), pp. 8271–8276.

[Bee+03] Niko Beerenwinkel et al. “Geno2pheno: Estimating phenotypic drug resistance from HIV-1 genotypes”. In: *Nucleic Acids Research* 31.13 (2003), pp. 3850–3855.

[Bel03] Richard E Bellman. *Dynamic Programming*. Dover Publication Inc., 2003. ISBN: 978-0486428093.

[Ber+00] Helen M Berman et al. “The protein data bank”. In: *Nucleic Acids Research* 28.1 (2000), pp. 235–242.

[Ber+04] Tim Berners-Lee et al. “Architecture of the world wide web, volume one”. In: (2004).

[Ber+08] Michael R Berthold et al. *KNIME: The Konstanz information miner*. Springer, 2008.

[Ber13] Jakob Bernoulli. *Ars conjectandi*. Impensis Thurnisiorum, fratrum, 1713.

[Bez+14] Jeff Bezanson et al. "Julia: A fresh approach to numerical computing". In: *arXiv preprint arXiv:1411.1607* (2014).

[BFF96] Tim Berners-Lee, R. Fielding, and H. Frystyk. *Hypertext Transfer Protocol – HTTP/1.0*. IETF - Network Working Group. May 1996. URL: `https://www.ietf.org/rfc/rfc1945.txt`.

[BFM05] Tim Berners-Lee, R. Fielding, and L. Masinter. *RFC3986: Uniform Resource Identifier (URI): Generic Syntax*. IETF - Network Working Group. 2005. URL: `https://tools.ietf.org/rfc/rfc3986.txt`.

[BH06] Howard Brody and Linda M Hunt. "BiDil: assessing a race-based pharmaceutical". In: *The Annals of Family Medicine* 4.6 (2006), pp. 556–560.

[BH95] Yoav Benjamini and Yosef Hochberg. "Controlling the false discovery rate: a practical and powerful approach to multiple testing". In: *Journal of the Royal Statistical Society. Series B (Methodological)* (1995), pp. 289–300.

[BHS09] Gordon Bell, Tony Hey, and Alex Szalay. "Beyond the data deluge". In: *Science* 323.5919 (2009), pp. 1297–1298.

[Bie12] Leslie G Biesecker. "Opportunities and challenges for the integration of massively parallel genomic sequencing into clinical practice: lessons from the ClinSeq project". In: *Genetics in Medicine* 14.4 (2012), pp. 393–398.

[Bis06] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. 1st ed. Springer, 2006. ISBN: 978-0387310732.

[BKE11] Jonathan S Berg, Muin J Khoury, and James P Evans. "Deploying whole genome sequencing in clinical practice and public health: meeting the challenge one bin at a time". In: *Genetics in Medicine* 13.6 (2011), pp. 499–504.

[BKW93] Silke Bienroth, Walter Keller, and Elmar Wahle. "Assembly of a processive messenger RNA polyadenylation complex." In: *The EMBO journal* 12.2 (1993), p. 585.

[BL12] Christoph Bock and Thomas Lengauer. "Managing drug resistance in cancer: lessons from HIV therapy". In: *Nature Reviews Cancer* 12.7 (2012), pp. 494–501.

[Bla03] Douglas L Black. "Mechanisms of alternative pre-messenger RNA splicing". In: *Annual review of biochemistry* 72.1 (2003), pp. 291–336.

[BLG15] Thomas Bleazard, Janine A Lamb, and Sam Griffiths-Jones. "Bias in microRNA functional enrichment analysis". In: *Bioinformatics* (2015), btv023.

[Bol+03] Benjamin M Bolstad et al. "A comparison of normalization methods for high density oligonucleotide array data based on variance and bias". In: *Bioinformatics* 19.2 (2003), pp. 185–193.

[Bon35] C. E. Bonferroni. "Il calcolo delle assicurazioni su gruppi di teste." In: *Studi in Onore del Professore Salvatore Ortu Carboni* (1935).

[Bon36] C. E. Bonferroni. "Teoria statistica delle classi e calcolo delle probabilita." In: *Pubblicazioni del R Istituto Superiore di Scienze Economiche e Commerciali di Firenze* (1936).

[Bon72] Phillip Bonacich. "Factoring and weighting approaches to status scores and clique identification". In: *Journal of Mathematical Sociology* 2.1 (1972), pp. 113–120.

[BP78] JB Beckwith and NF Palmer. "Histopathology and prognosis of Wilms tumor Results from the first national wilms' tumor study". In: *Cancer* 41.5 (1978), pp. 1937–1948.

[Bra+15] Nicolas Bray et al. "Near-optimal RNA-Seq quantification". In: *arXiv preprint arXiv:1505.02710* (2015).

[Bra14] Tim Bray. "The javascript object notation (json) data interchange format". In: (2014).

[Bre+12] Independent UK Panel on Breast Cancer Screening et al. "The benefits and harms of breast cancer screening: an independent review". In: *The Lancet* 380.9855 (2012), pp. 1778–1786.

[Bre01] Leo Breiman. "Random forests". In: *Machine learning* 45.1 (2001), pp. 5–32.

[Bri+00] Carolyn Buxton Bridges et al. "Effectiveness and cost-benefit of influenza vaccination of healthy working adults: a randomized controlled trial". In: *Jama* 284.13 (2000), pp. 1655–1663.

[Bro+06] Mark A Brown et al. "Identification and characterization of Smyd2: a split SET/MYND domain-containing histone H3 lysine 36-specific methyltransferase that interacts with the Sin3 histone deacetylase complex". In: *Molecular cancer* 5.1 (2006), p. 1.

[Bro+83] Bernard R Brooks et al. "CHARMM: a program for macromolecular energy, minimization, and dynamics calculations". In: *Journal of computational chemistry* 4.2 (1983), pp. 187–217.

[BS00] G Patrick Brady Jr and Pieter FW Stouten. "Fast prediction and visualization of protein binding pockets with PASS". In: *Journal of computer-aided molecular design* 14.4 (2000), pp. 383–401.

[BS96] Zhidong Bai and Hewa Saranadasa. "Effect of high dimension: by an example of a two sample problem". In: *Statistica Sinica* (1996), pp. 311–329.

[BSM93] Gabriele Basi, Elisabeth Schmid, and Kinsey Maundrell. "TATA box mutations in the Schizosaccharomyces pombe nmt1 promoter affect transcription efficiency but not the transcription start point or thiamine repressibility". In: *Gene* 123.1 (1993), pp. 131–136.

[BSS04] Kunihiro Baba, Ritei Shibata, and Masaaki Sibuya. "Partial correlation and conditional correlation as measures of conditional independence". In: *Australian & New Zealand Journal of Statistics* 46.4 (2004), pp. 657–664.

[BT97] Dimitris Bertsimas and John N Tsitsiklis. *Introduction to linear optimization*. Vol. 6. Athena Scientific Belmont, MA, 1997.

[Bur+13] Randall W Burt et al. "Colorectal cancer screening". In: *Journal of the National Comprehensive Cancer Network* 11.12 (2013), pp. 1538–1575.

[Bür13] Jonas Bürse. "Advanced Rendering for BALLView". MA thesis. Saarland University, 2013.

[BVR04] Adam Brymora, Valentina A Valova, and Phillip J Robinson. "Protein-Protein Interactions Identified by Pull-Down Experiments and Mass Spectrometry". In: *Current protocols in cell biology* (2004), pp. 17–5.

[BY01] Yoav Benjamini and Daniel Yekutieli. "The control of the false discovery rate in multiple testing under dependency". In: *Annals of statistics* (2001), pp. 1165–1188.

[C+03] Xiangqin Cui, Gary A Churchill, et al. "Statistical tests for differential expression in cDNA microarray experiments". In: *Genome Biol* 4.4 (2003), p. 210.

[C+15] Cancer Cell Line Encyclopedia Consortium, Genomics of Drug Sensitivity in Cancer Consortium, et al. "Pharmacogenomic agreement between two cancer cell line data sets". In: *Nature* 528.7580 (2015), pp. 84–87.

[Cal+79] RY Calne et al. "Cyclosporin A initially as the only immunosuppressant in 34 recipients of cadaveric organs: 32 kidneys, 2 pancreases, and 2 livers". In: *The Lancet* 314.8151 (1979), pp. 1033–1036.

[Cas+08] Ron Caspi et al. "The MetaCyc Database of metabolic pathways and enzymes and the BioCyc collection of Pathway/Genome Databases". In: *Nucleic Acids Research* 36.suppl 1 (2008), pp. D623–D631.

[CCB11] Mihaela-Daciana Craciun, Violeta Chis, and Cristina Bala. "Methods for discretizing continuous variables within the framework of bayesian networks". In: *Proceedings of the International Conference on Theory and Applications in Mathematics and Informatics, ICTAMI*. 2011, pp. 433–443.

[Cec87] Thomas R Cech. "The chemistry of self-splicing RNA and RNA enzymes". In: *Science* 236.4808 (1987), pp. 1532–1539.

[CFC05] Emidio Capriotti, Piero Fariselli, and Rita Casadio. "I-Mutant2.0: predicting stability changes upon mutation from the protein sequence or structure". In: *Nucleic Acids Research* 33.suppl 2 (2005), W306–W310.

[CG10] Elizabeth T Cirulli and David B Goldstein. "Uncovering the roles of rare variants in common disease through whole-genome sequencing". In: *Nature Reviews Genetics* 11.6 (2010), pp. 415–425.

[CG96] Stanley F Chen and Joshua Goodman. "An empirical study of smoothing techniques for language modeling". In: *Proceedings of the 34th annual meeting on Association for Computational Linguistics*. Association for Computational Linguistics. 1996, pp. 310–318.

[Cha+08] Anne-Amandine Chassot et al. "Activation of β-catenin signaling by Rspo1 controls differentiation of the mammalian ovary". In: *Human molecular genetics* 17.9 (2008), pp. 1264–1277.

[Cha+12] Hao-Ming Chang et al. "TRIM71 cooperates with microRNAs to repress Cdkn1a expression and promote embryonic stem cell proliferation". In: *Nature communications* 3 (2012), p. 923.

[Che+09] Iouri Chepelev et al. "Detection of single nucleotide variations in expressed exons of the human genome using RNA-Seq". In: *Nucleic Acids Research* 37.16 (2009), e106–e106.

[Che+11] Chao Chen et al. “Removing batch effects in analysis of expression microarray data: an evaluation of six batch adjustment methods”. In: *PloS One* 6.2 (2011), e17238.

[Che+12] Lin S Chen et al. “A regularized Hotelling’s T^2-test for pathway analysis in proteomic studies”. In: *Journal of the American Statistical Association* (2012).

[Chh+15] Sagar Chhangawala et al. “The impact of read length on quantification of differentially expressed genes and splice junction detection”. In: *Genome biology* 16.1 (2015), pp. 1–10.

[Cho+04] Pek Yoon Chong et al. “Analysis of deaths during the severe acute respiratory syndrome (SARS) epidemic in Singapore: challenges in determining a SARS diagnosis”. In: *Archives of pathology & laboratory medicine* 128.2 (2004), pp. 195–204.

[Cho+16] Chih-Hung Chou et al. “miRTarBase 2016: updates to the experimentally validated miRNA-target interactions database”. In: *Nucleic Acids Research* 44.D1 (2016), pp. D239–D247.

[Chu+10] Anna Chu et al. “Wilms’ tumour: a systematic review of risk factors and meta-analysis”. In: *Paediatric and perinatal epidemiology* 24.5 (2010), pp. 449–469.

[Cin+12] Pablo Cingolani et al. “A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of Drosophila melanogaster strain w1118; iso-2; iso-3”. In: *Fly* 6.2 (2012), pp. 80–92.

[CLL11] T. Cai, W. Liu, and X. Luo. “A Constrained l_1 Minimization Approach to Sparse Precision Matrix Estimation”. In: *Journal of the American Statistical Association* 106.494 (2011), pp. 594–607.

[Cof+03] John Calvin Coffey et al. “Excisional surgery for cancer cure: therapy at a cost”. In: *The Lancet Oncology* 4.12 (2003), pp. 760–768.

[Coi+02] Bertrand Coiffier et al. “CHOP chemotherapy plus rituximab compared with CHOP alone in elderly patients with diffuse large-B-cell lymphoma”. In: *New England Journal of Medicine* 346.4 (2002), pp. 235–242.

[Col+08] Megan F Cole et al. “Tcf3 is an integral component of the core regulatory circuitry of embryonic stem cells”. In: *Genes & development* 22.6 (2008), pp. 746–755.

[Con83] Michael L Connolly. “Solvent-accessible surfaces of proteins and nucleic acids”. In: *Science* 221.4612 (1983), pp. 709–713.

[CSK01] Lillian Chu, Eric Scharf, and Takashi Kondo. “GeneSpring TM: tools for analyzing microarray expression data”. In: *Genome Informatics* 12 (2001), pp. 227–229.

[CVY07] Robert G Cowell, Richard J Verrall, and YK Yoon. “Modeling operational risk with Bayesian networks”. In: *Journal of Risk and Insurance* 74.4 (2007), pp. 795–827.

[CWE15] Mark JP Chaisson, Richard K Wilson, and Evan E Eichler. “Genetic variation and the de novo assembly of human genomes”. In: *Nature Reviews Genetics* (2015).

[D+55] George B Dantzig, Alex Orden, Philip Wolfe, et al. “The generalized simplex method for minimizing a linear form under linear inequality restraints”. In: *Pacific Journal of Mathematics* 5.2 (1955), pp. 183–195.

[Dan16] Dancojocari. *Twenty-One Amino Acids*. 2016. URL: https://commons.wikimedia.org/w/index.php?curid=9176441.

[Dao+11] Phuong Dao et al. "Optimally discriminative subnetwork markers predict response to chemotherapy". In: *Bioinformatics* 27.13 (2011), pp. i205–i213.

[Day+11] Roger S Day et al. "Identifier mapping performance for integrating transcriptomics and proteomics experimental results". In: *BMC Bioinformatics* 12.1 (2011), p. 1.

[DBJ05] Warwick B Dunn, Nigel JC Bailey, and Helen E Johnson. "Measuring the metabolome: current analytical technologies". In: *Analyst* 130.5 (2005), pp. 606–625.

[DCM15] Sipko van Dam, Thomas Craig, and João Pedro de Magalhães. "GeneFriends: a human RNA-seq-based gene and transcript co-expression database". In: *Nucleic Acids Research* 43.D1 (2015), pp. D1124–D1132.

[De 04] Erik De Clercq. "Antiviral drugs in current clinical use". In: *Journal of Clinical Virology* 30.2 (2004), pp. 115–133.

[Deh+11] Anna K. Dehof et al. "Automated bond order assignment as an optimization problem". In: *Bioinformatics* 27.5 (2011), pp. 619–625.

[Deh+13] A. K. Dehof et al. "NightShift: NMR shift inference by general hybrid model training-a framework for NMR chemical shift prediction". In: *BMC Bioinformatics* 14.1 (2013), p. 98.

[DeL02] Warren L DeLano. "The PyMOL molecular graphics system". In: (2002).

[Dem58] Arthur P Dempster. "A high dimensional two sample significance test". In: *The Annals of Mathematical Statistics* (1958), pp. 995–1010.

[Den+04] Ahmet M Denli et al. "Processing of primary microRNAs by the Microprocessor complex". In: *Nature* 432.7014 (2004), pp. 231–235.

[DF94] G William Dec and Valentin Fuster. "Idiopathic dilated cardiomyopathy". In: *New England Journal of Medicine* 331.23 (1994), pp. 1564–1575.

[DG15] Harsh Dweep and Norbert Gretz. "miRWalk2.0: a comprehensive atlas of microRNA-target interactions". In: *Nature methods* 12.8 (2015), pp. 697–697.

[Die+97] Francisco Javier Diez et al. "DIAVAL, a Bayesian expert system for echocardiography". In: *Artificial Intelligence in Medicine* 10.1 (1997), pp. 59–73.

[Din+07] Irina Dinu et al. "Improving gene set analysis of microarray data by SAM-GS". In: *BMC Bioinformatics* 8.1 (2007), p. 242.

[Dit+08] Marcus T Dittrich et al. "Identifying functional modules in protein–protein interaction networks: an integrated exact approach". In: *Bioinformatics* 24.13 (2008), pp. i223–i231.

[DL93] Paul Dagum and Michael Luby. "Approximating probabilistic inference in Bayesian belief networks is NP-hard". In: *Artificial intelligence* 60.1 (1993), pp. 141–153.

[DLR14] Vincent T. DeVita, Theodore S. Lawrence, and Steven A. Rosenberg. *Cancer: Principles & Practice of Oncology*. 10th ed. 2014. ISBN: 978-1451192940.

[DLR77] Arthur P Dempster, Nan M Laird, and Donald B Rubin. "Maximum likelihood from incomplete data via the EM algorithm". In: *Journal of the royal statistical society. Series B (methodological)* (1977), pp. 1–38.

[DLS00] Patrik D'haeseleer, Shoudan Liang, and Roland Somogyi. "Genetic network inference: from co-expression clustering to reverse engineering". In: *Bioinformatics* 16.8 (2000), pp. 707–726.

[Dob+13] Alexander Dobin et al. "STAR: ultrafast universal RNA-seq aligner". In: *Bioinformatics* 29.1 (2013), pp. 15–21.

[Doj+13] Norbert Dojer et al. "BNFinder2: Faster Bayesian network learning and Bayesian classification". In: *Bioinformatics* (2013), btt323.

[DPG14] Jeffrey S. Dome, Elizabeth J. Perlman, and Norbert Graf. "Risk stratification for Wilms Tumor: Current Approach and Future Directions". In: American Society of Clinical Oncology. 2014.

[Dră+03] Sorin Drăghici et al. "Global functional profiling of gene expression". In: *Genomics* 81.2 (2003), pp. 98–104.

[Dru99] Marek J Druzdzel. "SMILE: Structural Modeling, Inference, and Learning Engine and GeNIe: a development environment for graphical decision-theoretic models". In: *AAAI/IAAI*. 1999, pp. 902–903.

[Dur+09] Steffen Durinck et al. "Mapping identifiers for the integration of genomic datasets with the R/Bioconductor package biomaRt". In: *Nature protocols* 4.8 (2009), pp. 1184–1191.

[Edd01] Sean R Eddy. "Non–coding RNA genes and the modern RNA world". In: *Nature Reviews Genetics* 2.12 (2001), pp. 919–929.

[EDL02] Ron Edgar, Michael Domrachev, and Alex E Lash. "Gene Expression Omnibus: NCBI gene expression and hybridization array data repository". In: *Nucleic Acids Research* 30.1 (2002), pp. 207–210.

[Edm71] Jack Edmonds. "Matroids and the greedy algorithm". In: *Mathematical programming* 1.1 (1971), pp. 127–136.

[Efr07] Bradley Efron. "Size, power and false discovery rates". In: *The Annals of Statistics* (2007), pp. 1351–1377.

[Efr08] Bradley Efron. "Simultaneous inference: When should hypothesis testing problems be combined?" In: *The annals of applied statistics* (2008), pp. 197–223.

[EH08] Lee M Ellis and Daniel J Hicklin. "VEGF-targeted therapy: mechanisms of anti-tumour activity". In: *Nature reviews cancer* 8.8 (2008), pp. 579–591.

[EM77] Bradley Efron and Carl N Morris. *Stein's paradox in statistics*. WH Freeman, 1977.

[ENC04] ENCODE Project Consortium. "The ENCODE (ENCyclopedia of DNA elements) project". In: *Science* 306.5696 (2004), pp. 636–640.

[ET07] Bradley Efron and Robert Tibshirani. "On testing the significance of sets of genes". In: *The annals of applied statistics* (2007), pp. 107–129.

[Etz+02] Ruth Etzioni et al. "Overdiagnosis due to prostate-specific antigen screening: lessons from US prostate cancer incidence trends". In: *Journal of the National Cancer Institute* 94.13 (2002), pp. 981–990.

[Faz+13] Dávid Fazekas et al. "SignaLink 2–a signaling pathway resource with multi-layered regulatory networks". In: *BMC Systems Biology* 7.1 (2013), p. 1.

[Fel+08] Magdalena Feldhahn et al. "EpiToolKit—a web server for computational immunomics". In: *Nucleic Acids Research* 36.suppl 2 (2008), W519–W522.

[Fen93] Frank Fenner. "Smallpox: emergence, global spread, and eradication". In: *History and philosophy of the life sciences* (1993), pp. 397–420.

[Fer+11] Guy Haskin Fernald et al. "Bioinformatics challenges for personalized medicine". In: *Bioinformatics* 27.13 (2011), pp. 1741–1748.

[FHT08] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. "Sparse inverse covariance estimation with the graphical lasso". In: *Biostatistics* 9.3 (2008), pp. 432–441.

[FHT09] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*. 2nd ed. Vol. 1. Springer series in statistics Springer, Berlin, 2009.

[Fid03] Isaiah J Fidler. "The pathogenesis of cancer metastasis: the'seed and soil'hypothesis revisited". In: *Nature Reviews Cancer* 3.6 (2003), pp. 453–458.

[Fil+05] Witold Filipowicz et al. "Post-transcriptional gene silencing by siRNAs and miRNAs". In: *Current opinion in structural biology* 15.3 (2005), pp. 331–341.

[Fis25] Ronald Aylmer Fisher. *Statistical methods for research workers*. 5th ed. Oliver and Boyd, 1925.

[FR64] Reeves Fletcher and Colin M Reeves. "Function minimization by conjugate gradients". In: *The computer journal* 7.2 (1964), pp. 149–154.

[Fra+16] Max Franz et al. "Cytoscape. js: a graph theory library for visualisation and analysis". In: *Bioinformatics* 32.2 (2016), pp. 309–311.

[Fre+10] Benjamin French et al. "Statistical design of personalized medicine interventions: the Clarification of Optimal Anticoagulation through Genetics (COAG) trial". In: *Trials* 11.1 (2010), p. 108.

[Fre77] Linton C Freeman. "A set of measures of centrality based on betweenness". In: *Sociometry* (1977), pp. 35–41.

[Fri+00] Nir Friedman et al. "Using Bayesian networks to analyze expression data". In: *Journal of computational biology* 7.3-4 (2000), pp. 601–620.

[Fro+92] Marianne Frommer et al. "A genomic sequencing protocol that yields a positive display of 5-methylcytosine residues in individual DNA strands." In: *Proceedings of the National Academy of Sciences* 89.5 (1992), pp. 1827–1831.

[Fru+08] Felix W Frueh et al. "Pharmacogenomic biomarker information in drug labels approved by the United States food and drug administration: prevalence of related drug use". In: *Pharmacotherapy: The Journal of Human Pharmacology and Drug Therapy* 28.8 (2008), pp. 992–998.

[FS89] Stanley Fields and Ok-kyu Song. "A novel genetic system to detect protein protein interactions". In: (1989).

[FT02] Roy T Fielding and Richard N Taylor. "Principled design of the modern Web architecture". In: *ACM Transactions on Internet Technology (TOIT)* 2.2 (2002), pp. 115–150.

[FT04] Andrew P Feinberg and Benjamin Tycko. "The history of cancer epigenetics". In: *Nature Reviews Cancer* 4.2 (2004), pp. 143–153.

[Fuk+09] R Fukuzawa et al. "Canonical WNT signalling determines lineage specificity in Wilms tumour". In: *Oncogene* 28.8 (2009), pp. 1063–1075.

[Gal+14] Alison P Galvani et al. "Ebola vaccination: if not now, when?" In: *Annals of internal medicine* 161.10 (2014), pp. 749–750.

[Gar+12] Mathew J Garnett et al. "Systematic identification of genomic markers of drug sensitivity in cancer cells". In: *Nature* 483.7391 (2012), pp. 570–575.

[GAT16] GATC Biotech AG. *Sanger ABI 3730xl*. 2016. URL: `https://www.gatc-biotech.com/de/gatc/sequenziertechnologien/sanger-abi-3730xl.html`.

[GB07] Jelle J Goeman and Peter Bühlmann. "Analyzing gene expression data in terms of gene sets: methodological issues". In: *Bioinformatics* 23.8 (2007), pp. 980–987.

[GE15] Patrice Godard and Jonathan van Eyll. "Pathway analysis from lists of microRNAs: common pitfalls and alternative strategy". In: *Nucleic Acids Research* (2015), gkv249.

[Gei+11] Ludwig Geistlinger et al. "From sets to graphs: towards a realistic enrichment analysis of transcriptomic systems". In: *Bioinformatics* 27.13 (2011), pp. i366–i373.

[Gen+04] Robert C Gentleman et al. "Bioconductor: open software development for computational biology and bioinformatics". In: *Genome biology* 5.10 (2004), p. 1.

[Ger+14] Andreas Gerasch et al. "BiNA: A Visual Analytics Tool for Biological Network Data". In: *PloS One* 9.2 (2014), e87397.

[Gig93] Gerd Gigerenzer. "The superego, the ego, and the id in statistical reasoning". In: *A handbook for data analysis in the behavioral sciences.* Ed. by Gideon Keren and Charles Lewis. L. Erlbaum Associates, 1993, pp. 311–339.

[Gla+12] Enrico Glaab et al. "EnrichNet: network-based gene set enrichment analysis". In: *Bioinformatics* 28.18 (2012), pp. i451–i457.

[GN00] Emden R Gansner and Stephen C North. "An open graph visualization system and its applications to software engineering". In: *Software Practice and Experience* 30.11 (2000), pp. 1203–1233.

[Goe+04] Jelle J Goeman et al. “A global test for groups of genes: testing association with a colinical outcome”. In: *Bioinformatics* 20.1 (2004), pp. 93–99.

[Goe+10] Jeremy Goecks et al. “Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences”. In: *Genome Biol* 11.8 (2010), R86.

[Gol+11] Michal Golan-Mashiach et al. “Identification of CTCF as a master regulator of the clustered protocadherin genes”. In: *Nucleic Acids Research* (2011), gkr1260.

[Gre+03] Dov Greenbaum et al. “Comparing protein abundance and mRNA expression levels on a genomic scale”. In: *Genome biology* 4.9 (2003), p. 1.

[Gri+06] Sam Griffiths-Jones et al. “miRBase: microRNA sequences, targets and gene nomenclature”. In: *Nucleic Acids Research* 34.suppl 1 (2006), pp. D140–D144.

[GS01] Geoffrey Grimmett and David Stirzaker. *Probability and random processes*. Oxford university press, 2001.

[GS08] Iliyan Georgiev and Philipp Slusallek. “Rtfact: Generic concepts for flexible and high performance ray tracing”. In: *Interactive Ray Tracing, 2008. RT 2008. IEEE Symposium on*. IEEE. 2008, pp. 115–122.

[GS12] Johann A Gagnon-Bartsch and Terence P Speed. “Using control genes to correct for unwanted variation in microarray data”. In: *Biostatistics* 13.3 (2012), pp. 539–552.

[GT07] Alison Gopnik and Joshua B Tenenbaum. “Bayesian networks, Bayesian learning and cognitive development”. In: *Developmental science* 10.3 (2007), pp. 281–287.

[Gu+15] Xuefeng Gu et al. “Silencing of R-Spondin1 increases radiosensitivity of glioma cells”. In: *Oncotarget* 6.12 (2015), p. 9756.

[Gud+03] Martin Gudgin et al. “SOAP Version 1.2”. In: *W3C recommendation* 24 (2003), p. 12.

[Gun+04] Kevin L Gunderson et al. “Decoding randomly ordered DNA arrays”. In: *Genome research* 14.5 (2004), pp. 870–877.

[Gün13] Harald Günther. *NMR spectroscopy: basic principles, concepts and applications in chemistry*. John Wiley & Sons, 2013.

[Gyg+99] Steven P Gygi et al. “Correlation between protein and mRNA abundance in yeast”. In: *Molecular and cellular biology* 19.3 (1999), pp. 1720–1730.

[Haa+15] Jan Haas et al. “Atlas of the clinical genetics of human dilated cardiomyopathy”. In: *European heart journal* 36.18 (2015), pp. 1123–1135.

[Hai+13] Benjamin Haibe-Kains et al. “Inconsistency in large pharmacogenomic studies”. In: *Nature* 504.7480 (2013), pp. 389–393.

[Hal06] Eric J Hall. “Intensity-modulated radiation therapy, protons, and the risk of second cancers”. In: *International Journal of Radiation Oncology* Biology* Physics* 65.1 (2006), pp. 1–7.

[Hal96] Thomas A Halgren. "Merck molecular force field. I. Basis, form, scope, parameterization, and performance of MMFF94". In: *Journal of computational chemistry* 17.5-6 (1996), pp. 490–519.

[Han+11] Beomsoo Han et al. "SHIFTX2: significantly improved protein chemical shift prediction". In: *Journal of biomolecular NMR* 50.1 (2011), pp. 43–57.

[Hao+12] Huai-Xiang Hao et al. "ZNRF3 promotes Wnt receptor turnover in an R-spondin-sensitive manner". In: *Nature* 485.7397 (2012), pp. 195–200.

[Har+06] Jennifer Harrow et al. "GENCODE: producing a reference annotation for ENCODE". In: *Genome Biol* 7.Suppl 1 (2006), S4.

[Har+12] Jennifer Harrow et al. "GENCODE: the reference human genome annotation for The ENCODE Project". In: *Genome research* 22.9 (2012), pp. 1760–1774.

[HD05] Joel N Hirschhorn and Mark J Daly. "Genome-wide association studies for common diseases and complex traits". In: *Nature Reviews Genetics* 6.2 (2005), pp. 95–108.

[HDS96] William Humphrey, Andrew Dalke, and Klaus Schulten. "VMD: visual molecular dynamics". In: *Journal of molecular graphics* 14.1 (1996), pp. 33–38.

[Hei+14] Axel Heidenreich et al. "EAU guidelines on prostate cancer. Part 1: screening, diagnosis, and local treatment with curative intent—update 2013". In: *European urology* 65.1 (2014), pp. 124–137.

[Heo+14] Yun Heo et al. "BLESS: bloom filter-based error correction solution for high-throughput sequencing reads". In: *Bioinformatics* 30.10 (2014), pp. 1354–1362.

[Her+12] Sonja Herres-Pawlis et al. "Workflow-enhanced conformational analysis of guanidine zinc complexes via a science gateway". In: *Studies in health technology and informatics* 175 (2012), pp. 142–151.

[Heu+15] MM van den Heuvel-Eibrink et al. "Outcome of localised blastemal-type Wilms tumour patients treated according to intensified treatment in the SIOP WT 2001 protocol, a report of the SIOP Renal Tumour Study Group (SIOP-RTSG)". In: *European Journal of Cancer* 51.4 (2015), pp. 498–506.

[HH16] Ian Hickson and David Hyatt. "HTML5: A vocabulary and associated APIs for HTML and XHTML". In: *W3C Recommendation* (Oct. 2016).

[Hil+10] Andreas Hildebrandt et al. "BALL-biochemical algorithms library 1.3". In: *BMC Bioinformatics* 11.1 (2010), p. 531.

[Hil+14a] Anna K. Hildebrandt et al. "ballaxy: web services for structural bioinformatics". In: *Bioinformatics* (2014), btu574.

[Hil+14b] Anna K. Hildebrandt et al. "Efficient computation of root mean square deviations under rigid transformations". In: *Journal of computational chemistry* 35.10 (2014), pp. 765–771.

[HIV+14] Joint United Nations Programme on HIV/AIDS (UNAIDS) et al. *Global report: UNAIDS report on the global AIDS epidemic 2013. Geneva, Switzerland: UNAIDS; 2013. Available from: h ttp*. 2014.

[HL56] JOSEPH L Hodges Jr and ERIC L Lehmann. "The efficiency of some nonparametric competitors of the t-test". In: *The Annals of Mathematical Statistics* (1956), pp. 324–335.

[Hol+01] Daniel Holder et al. "Statistical analysis of high density oligonucleotide arrays: a SAFER approach". In: *GeneLogic Workshop on Low Level Analysis of Affymetrix GeneChip Data*. 2001.

[Hol79] Sture Holm. "A simple sequentially rejective multiple test procedure". In: *Scandinavian journal of statistics* (1979), pp. 65–70.

[Hot31] H. Hotelling. "The generalization of Student's ratio". In: *The Annals of Mathematical Statistics* 2.3 (1931), pp. 360–378.

[HSL09] Da Wei Huang, Brad T Sherman, and Richard A Lempicki. "Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists". In: *Nucleic Acids Research* 37.1 (2009), pp. 1–13.

[Hsu+10] Sheng-Da Hsu et al. "miRTarBase: a database curates experimentally validated microRNA–target interactions". In: *Nucleic Acids Research* (2010), gkq1107.

[Hsu+14] Sheng-Da Hsu et al. "miRTarBase update 2014: an information resource for experimentally validated miRNA-target interactions". In: *Nucleic Acids Research* 42.D1 (2014), pp. D78–D85.

[Hua+06] Jing Huang et al. "Repression of p53 activity by Smyd2-mediated methylation". In: *Nature* 444.7119 (2006), pp. 629–632.

[Hub+02] Wolfgang Huber et al. "Variance stabilization applied to microarray data calibration and to the quantification of differential expression". In: *Bioinformatics* 18.suppl 1 (2002), S96–S104.

[Hun+11] Jui-Hung Hung et al. "Gene set enrichment analysis: performance evaluation and usage guidelines". In: *Briefings in bioinformatics* (2011), bbr049.

[Hus03] Dirk Husmeier. "Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks". In: *Bioinformatics* 19.17 (2003), pp. 2271–2282.

[HW00] Douglas Hanahan and Robert A Weinberg. "The hallmarks of cancer". In: *cell* 100.1 (2000), pp. 57–70.

[HW11] Douglas Hanahan and Robert A Weinberg. "Hallmarks of cancer: the next generation". In: *cell* 144.5 (2011), pp. 646–674.

[HZ12] Che-Ming Jack Hu and Liangfang Zhang. "Nanoparticle-based combination therapy toward overcoming drug resistance in cancer". In: *Biochemical pharmacology* 83.8 (2012), pp. 1104–1111.

[IBM12] IBM. *ILOG CPLEX 12.4*. 2012. URL: `http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/`.

[Ide+02] Trey Ideker et al. "Discovering regulatory and signalling circuits in molecular interaction networks". In: *Bioinformatics* 18.suppl 1 (2002), S233–S240.

[Ier+10] Martijn P van Iersel et al. "The BridgeDb framework: standardized access to gene, protein and metabolite identifier mapping services". In: *BMC Bioinformatics* 11.1 (2010), p. 1.

[Ill16] Illumina. *HiSeq X Series of Sequencing Systems*. July 2016. URL: http://www.illumina.com/content/dam/illumina-marketing/documents/products/datasheets/datasheet-hiseq-x-ten.pdf.

[INC05] Affymetrix INC. *The GeneChip® System – An Integrated Solution for Expression and DNA Analysis*. 2005. URL: http://www.affymetrix.com/support/technical/brochures/genechip_system_brochure.pdf.

[Ing57] Vernon M Ingram. "Gene mutations in human haemoglobin: the chemical difference between normal and sickle cell haemoglobin". In: *Nature* 180.4581 (1957), pp. 326–328.

[Ino+97] Kazushi Inoue et al. "Aberrant overexpression of the Wilms tumor gene (WT1) in human leukemia". In: *Blood* 89.4 (1997), pp. 1405–1412.

[Iri+03] Rafael A Irizarry et al. "Exploration, normalization, and summaries of high density oligonucleotide array probe level data". In: *Biostatistics* 4.2 (2003), pp. 249–264.

[Isr+13] Trijn Israels et al. "SIOP PODC: clinical guidelines for the management of children with Wilms tumour in a low income setting". In: *Pediatric blood & cancer* 60.1 (2013), pp. 5–11.

[J+01] Eric Jones, Travis Oliphant, Pearu Peterson, et al. *SciPy: Open source scientific tools for Python*. [Online; accessed 2016-07-27]. 2001. URL: http://www.scipy.org/.

[Jai05] Kewal K Jain. "Personalised medicine for cancer: from drug development into clinical practice". In: *Expert opinion on pharmacotherapy* 6.9 (2005), pp. 1463–1476.

[Jen+89] FV Jensen et al. *An expert system for control of waste water treatment — a pilot project*. Tech. rep. Technical report, Judex Datasystemer A/S, 1989.

[Joh83] Irving S Johnson. "Human insulin from recombinant DNA technology". In: *Science* 219.4585 (1983), pp. 632–637.

[Jos+05] G Joshi-Tope et al. "Reactome: a knowledgebase of biological pathways". In: *Nucleic Acids Research* 33.suppl 1 (2005), pp. D428–D432.

[JS61] William James and Charles Stein. "Estimation with quadratic loss". In: *Proceedings of the fourth Berkeley symposium on mathematical statistics and probability*. Vol. 1. 1961. 1961, pp. 361–379.

[Jua+16] David Juan et al. "Epigenomic Co-localization and Co-evolution Reveal a Key Role for 5hmC as a Communication Hub in the Chromatin Network of ESCs". In: *Cell reports* 14.5 (2016), pp. 1246–1257.

[Kal+12] Markus Kalisch et al. "Causal Inference Using Graphical Models with the R Package pcalg". In: *Journal of Statistical Software* 47.11 (2012), pp. 1–26. URL: http://www.jstatsoft.org/v47/i11/.

[Kam95] SL Kamholz. "Resurgence of tuberculosis: the perspective a dozen years later." In: *Journal of the Association for Academic Minority Physicians* 7.3 (1995), pp. 83–86.

[Kan+06] Minoru Kanehisa et al. "From genomics to chemical genomics: new developments in KEGG". In: *Nucleic Acids Research* 34.suppl 1 (2006), pp. D354–D357.

[Kan+10a] Kumaran Kandasamy et al. "NetPath: a public resource of curated signal transduction pathways". In: *Genome biology* 11.1 (2010), p. 1.

[Kan+10b] Minoru Kanehisa et al. "KEGG for representation and analysis of molecular networks involving diseases and drugs". In: *Nucleic Acids Research* 38.suppl 1 (2010), pp. D355–D360.

[Kan06] Gopal K Kanji. *100 statistical tests*. Sage, 2006. ISBN: 978-1446222508.

[Kar84] Narendra Karmarkar. "A new polynomial-time algorithm for linear programming". In: *Proceedings of the sixteenth annual ACM symposium on Theory of computing*. ACM. 1984, pp. 302–311.

[KBL07] Andreas Keller, Christina Backes, and Hans-Peter Lenhof. "Computation of significance scores of unweighted Gene Set Enrichment Analyses". In: *BMC Bioinformatics* 8.1 (2007), pp. 1–7. ISSN: 1471-2105. DOI: 10.1186/1471-2105-8-290. URL: http://dx.doi.org/10.1186/1471-2105-8-290.

[Kel+06] Andreas Keller et al. "A minimally invasive multiple marker approach allows highly efficient detection of meningioma tumors". In: *BMC Bioinformatics* 7.1 (2006), p. 1.

[Kel+09] Andreas Keller et al. "A novel algorithm for detecting differentially regulated paths based on gene set enrichment analysis". In: *Bioinformatics* 25.21 (2009), pp. 2787–2794.

[Kel+12] Thomas Kelder et al. "WikiPathways: building research communities on biological pathways". In: *Nucleic Acids Research* 40.D1 (2012), pp. D1301–D1307.

[Ken+58] John C Kendrew et al. "A three-dimensional model of the myoglobin molecule obtained by x-ray analysis". In: *Nature* 181.4610 (1958), pp. 662–666.

[KF09] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[KG00] M. Kanehisa and S. Goto. "KEGG: kyoto encyclopedia of genes and genomes." eng. In: *Nucleic Acids Res* 28.1 (Jan. 2000), pp. 27–30.

[KGH12] Alexander Kohlmann, Vera Grossmann, and Torsten Haferlach. "Integration of next-generation sequencing into clinical practice: are we there yet?" In: *Seminars in oncology*. Vol. 39. 1. Elsevier. 2012, pp. 26–36.

[Kha80] Leonid G Khachiyan. "Polynomial algorithms in linear programming". In: *USSR Computational Mathematics and Mathematical Physics* 20.1 (1980), pp. 53–72.

[Kim+08] Kyung-Ah Kim et al. "R-Spondin family members regulate the Wnt pathway by a common mechanism". In: *Molecular biology of the cell* 19.6 (2008), pp. 2588–2596.

[Kin+12] Yoshiaki Kinoshita et al. “The prognostic significance of blastemal predominant histology in initially resected Wilms’ tumors: A report from the Study Group for Pediatric Solid Tumors in the Kyushu Area, Japan”. In: *Journal of pediatric surgery* 47.12 (2012), pp. 2205–2209.

[Kir05] Marc W Kirschner. “The meaning of systems biology”. In: *Cell* 121.4 (2005), pp. 503–504.

[Kir47] G Kirchhoff. “Über die Auflösung der Gleichungen, auf welche man bei der Untersuchung der linearen Verteilung galvanischer Ströme geführt wird”. In: *Poggendorf Ann. Physik* 72 (1847), pp. 497–508.

[Kit02] Hiroaki Kitano. “Systems biology: a brief overview”. In: *Science* 295.5560 (2002), pp. 1662–1664.

[KL00] Oliver Kohlbacher and Hans-Peter Lenhof. “BALL—rapid software prototyping in computational molecular biology”. In: *Bioinformatics* 16.9 (2000), pp. 815–824.

[Kle+02] Christoph A Klein et al. “Genetic heterogeneity of single disseminated tumour cells in minimal residual cancer”. In: *The Lancet* 360.9334 (2002), pp. 683–689.

[KMR78] Alan J Kinniburgh, Janet E Mertz, and Jeffrey Ross. “The precursor of mouse β-globin messenger RNA contains two intervening RNA sequences”. In: *Cell* 14.3 (1978), pp. 681–693.

[KN09] Matthias Keller and Martin Nussbaumer. “Cascading style sheets: a novel approach towards productive styling with today’s standards”. In: *Proceedings of the 18th international conference on World wide web*. ACM. 2009, pp. 1161–1162.

[Kni+09] Theo A Knijnenburg et al. “Fewer permutations, more accurate P-values”. In: *Bioinformatics* 25.12 (2009), pp. i161–i168.

[Koh12] Oliver Kohlbacher. “CADDSuite-a workflow-enabled suite of open-source tools for drug discovery.” In: *J. Cheminformatics* 4.S-1 (2012), O2.

[Kol33] Andrej N Kolmogorov. *Sulla determinazione empirica di una legge di distribuzione*. Vol. 4. G. Inst. Ital. Attuari, 1933.

[Kon+06] Gottfried E Konecny et al. “Activity of the dual kinase inhibitor lapatinib (GW572016) against HER-2-overexpressing and trastuzumab-treated breast cancer cells”. In: *Cancer research* 66.3 (2006), pp. 1630–1639.

[Kor+12] Sergey Koren et al. “Hybrid error correction and de novo assembly of single-molecule sequencing reads”. In: *Nature biotechnology* 30.7 (2012), pp. 693–700.

[Kor99] Roger D Kornberg. “Eukaryotic transcriptional control”. In: *Trends in Biochemical Sciences* 24.12 (1999), pp. M46–M49.

[KPP06] Sek Won Kong, William T Pu, and Peter J Park. “A multivariate approach for integrating genome-wide expression data and biological knowledge”. In: *Bioinformatics* 22.19 (2006), pp. 2373–2380.

[Kro04] Mel N Kronick. “Creation of the whole human genome microarray”. In: *Expert review of proteomics* 1.1 (2004), pp. 19–28.

[Kry+05] Stephen F Kry et al. "The calculated risk of fatal secondary malignancies from intensity-modulated radiation therapy". In: *International Journal of Radiation Oncology* Biology* Physics* 62.4 (2005), pp. 1195–1203.

[KSB12] Purvesh Khatri, Marina Sirota, and Atul J Butte. "Ten years of pathway analysis: current approaches and outstanding challenges". In: *PLoS computational biology* 8.2 (2012), e1002375.

[KWL07] Alexander Kamb, Susan Wee, and Christoph Lengauer. "Why is cancer drug discovery so difficult?" In: *Nature Reviews Drug Discovery* 6.2 (2007), pp. 115–120.

[Lac+10] Alexander Lachmann et al. "ChEA: transcription factor regulation inferred from integrating genome-wide ChIP-X experiments". In: *Bioinformatics* 26.19 (2010), pp. 2438–2444.

[Lan+01] Eric S Lander et al. "Initial sequencing and analysis of the human genome". In: *Nature* 409.6822 (2001), pp. 860–921.

[Lan+14] Melissa J Landrum et al. "ClinVar: public archive of relationships among sequence variation and human phenotype". In: *Nucleic Acids Research* 42.D1 (2014), pp. D980–D985.

[Lau96] Steffen L Lauritzen. *Graphical models*. Clarendon Press, 1996.

[Lav+15] T Laver et al. "Assessing the performance of the Oxford nanopore Technologies MinION". In: *Biomolecular detection and quantification* 3 (2015), pp. 1–8.

[Law+14] Vivian Law et al. "DrugBank 4.0: shedding new light on drug metabolism". In: *Nucleic Acids Research* 42.D1 (2014), pp. D1091–D1097.

[Laz02] Yuri Lazebnik. "Can a biologist fix a radio?—Or, what I learned while studying apoptosis". In: *Cancer cell* 2.3 (2002), pp. 179–182.

[LCB10] Sébastien Loriot, Frédéric Cazals, and Julie Bernauer. "ESBTL: efficient PDB parser and data structure for the structural and geometric analysis of biological macromolecules". In: *Bioinformatics* 26.8 (2010), pp. 1127–1128.

[LD09] Heng Li and Richard Durbin. "Fast and accurate short read alignment with Burrows–Wheeler transform". In: *Bioinformatics* 25.14 (2009), pp. 1754–1760.

[Le+13] Hai-Son Le et al. "Probabilistic error correction for RNA sequencing". In: *Nucleic Acids Research* (2013), gkt215.

[Lee+04] Yoontae Lee et al. "MicroRNA genes are transcribed by RNA polymerase II". In: *The EMBO journal* 23.20 (2004), pp. 4051–4060.

[Lee+10] Jeffrey T Leek et al. "Tackling the widespread and critical impact of batch effects in high-throughput data". In: *Nature Reviews Genetics* 11.10 (2010), pp. 733–739.

[Lee+14] Seo Hyun Lee et al. "The ubiquitin ligase human TRIM71 regulates let-7 microRNA biogenesis via modulation of Lin28B protein". In: *Biochimica et Biophysica Acta (BBA)-Gene Regulatory Mechanisms* 1839.5 (2014), pp. 374–386.

[Lei+95] Philip A Leighton et al. "An enhancer deletion affects both H19 and Igf2 expression." In: *Genes & Development* 9.17 (1995), pp. 2079–2089.

[LeP08] Emily LeProust. "Agilent's microarray platform: How high-fidelity DNA synthesis maximizes the dynamic range of gene expression measurements". In: *Agilent Technologies, Santa Clara, CA* (2008).

[Les07] LJ Lesko. "Personalized medicine: elusive dream or imminent reality?" In: *Clinical Pharmacology & Therapeutics* 81.6 (2007), pp. 807–816.

[Lew12] Michael J Lew. "Bad statistical practice in pharmacology (and other basic biomedical disciplines): you probably don't know P". In: *British journal of pharmacology* 166.5 (2012), pp. 1559–1567.

[LH01] Sean Bong Lee and Daniel A Haber. "Wilms tumor and the WT1 gene". In: *Experimental cell research* 264.1 (2001), pp. 74–99.

[LHA14] Michael I Love, Wolfgang Huber, and Simon Anders. "Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2". In: *Genome Biol* 15.12 (2014), p. 550. URL: `http://www.biomedcentral.com/content/pdf/s13059-014-0550-8.pdf`.

[Li+10] Ruiqiang Li et al. "De novo assembly of human genomes with massively parallel short read sequencing". In: *Genome research* 20.2 (2010), pp. 265–272.

[Li+13] Yang Li et al. "HMDD v2. 0: a database for experimentally supported human microRNA and disease associations". In: *Nucleic Acids Research* (2013), gkt1023.

[Lic+12] Luana Licata et al. "MINT, the molecular interaction database: 2012 update". In: *Nucleic Acids Research* 40.D1 (2012), pp. D857–D861.

[Lis+16] Markus List et al. "KeyPathwayMinerWeb: online multi-omics network enrichment". In: *Nucleic Acids Research* (2016), gkw373.

[Liu74] TZEN-PING Liu. "Bayes estimation for the variance of a finite population". In: *Metrika* 21.1 (1974), pp. 127–132.

[Lju+06] Ivana Ljubić et al. "An algorithmic framework for the exact solution of the prize-collecting Steiner tree problem". In: *Mathematical programming* 105.2-3 (2006), pp. 427–449.

[LKV98] Christoph Lengauer, Kenneth W Kinzler, and Bert Vogelstein. "Genetic instabilities in human cancers". In: *Nature* 396.6712 (1998), pp. 643–649.

[LMS91] Irvin J Lustig, Roy E Marsten, and David F Shanno. "Computational experience with a primal-dual interior point method for linear programming". In: *Linear Algebra and Its Applications* 152 (1991), pp. 191–222.

[Lop+10] Christian T Lopes et al. "Cytoscape Web: an interactive web-based network browser". In: *Bioinformatics* 26.18 (2010), pp. 2347–2348.

[LS00] Hartmut Liefke and Dan Suciu. "XMill: an efficient compressor for XML data". In: *ACM Sigmod Record*. Vol. 29. 2. ACM. 2000, pp. 153–164.

[LS07] Jeffrey T Leek and John D Storey. "Capturing heterogeneity in gene expression studies by surrogate variable analysis". In: *PLoS Genet* 3.9 (2007), e161.

[Lu+05] Yan Lu et al. "Hotelling's T^2 multivariate profiling for detecting differential expression in microarrays". In: *Bioinformatics* 21.14 (2005), pp. 3105–3113.

[Lu+10] Ming Lu et al. "TAM: a method for enrichment and depletion analysis of a microRNA category in a list of microRNAs". In: *BMC Bioinformatics* 11.1 (2010), p. 1.

[Lui+91] Bf F Luisi et al. "Crystallographic analysis of the interaction of the glucocorticoid receptor with DNA". In: (1991).

[LW02] Andy Liaw and Matthew Wiener. "Classification and Regression by randomForest". In: *R News* 2.3 (2002), pp. 18–22. URL: `http://CRAN.R-project.org/doc/Rnews/`.

[LW04] Olivier Ledoit and Michael Wolf. "A well-conditioned estimator for large-dimensional covariance matrices". In: *Journal of multivariate analysis* 88.2 (2004), pp. 365–411.

[M+05] U Mansmann, R Meister, et al. "Testing differential gene expression in functional groups Goeman's global test versus an ANCOVA approach". In: *Methods Inf Med* 44.3 (2005), pp. 449–453.

[M+99] Kevin Murphy, Saira Mian, et al. *Modelling gene expression data using dynamic Bayesian networks*. Tech. rep. Technical report, Computer Science Division, University of California, Berkeley, CA, 1999.

[Mag+05] Donna Maglott et al. "Entrez Gene: gene-centered information at NCBI". In: *Nucleic Acids Research* 33.suppl 1 (2005), pp. D54–D58.

[Mah36] Prasanta Chandra Mahalanobis. "On the generalized distance in statistics". In: *Proceedings of the National Institute of Sciences (Calcutta)* 2 (1936), pp. 49–55.

[MAL06] P. Merrick, S. Allen, and J. Lapp. *XML remote procedure call (XML-RPC)*. US Patent 7,028,312. Apr. 2006. URL: `https://www.google.com/patents/US7028312`.

[Man+02] Matthias Mann et al. "Analysis of protein phosphorylation using mass spectrometry: deciphering the phosphoproteome". In: *Trends in biotechnology* 20.6 (2002), pp. 261–268.

[Man+14] Kirk J Mantione et al. "Comparing bioinformatic gene expression profiling methods: microarray and RNA-Seq". In: *Medical science monitor basic research* 20 (2014), pp. 138–141.

[Mar+08a] John C Marioni et al. "RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays". In: *Genome research* 18.9 (2008), pp. 1509–1517.

[Mar+08b] Alexander Marson et al. "Connecting microRNA genes to the core transcriptional regulatory circuitry of embryonic stem cells". In: *Cell* 134.3 (2008), pp. 521–533.

[Mas+12] Hiroko Masuda et al. "Role of epidermal growth factor receptor in breast cancer". In: *Breast cancer research and treatment* 136.2 (2012), pp. 331–345.

[Mat+06] Volker Matys et al. "TRANSFAC® and its module TRANSCompel®: transcriptional gene regulation in eukaryotes". In: *Nucleic Acids Research* 34.suppl 1 (2006), pp. D108–D110.

[MB06] Nicolai Meinshausen and Peter Bühlmann. "High-dimensional graphs and variable selection with the lasso". In: *The annals of statistics* (2006), pp. 1436–1462.

[MBS92] H Miwa, M Beran, and GF Saunders. "Expression of the Wilms' tumor gene (WT1) in human leukemias." In: *Leukemia* 6.5 (1992), pp. 405–409.

[McA+12] Andrew McAfee et al. "Big data". In: *The management revolution. Harvard Bus Rev* 90.10 (2012), pp. 61–67.

[MCK14] Sean Maxwell, Mark R Chance, and Mehmet Koyutürk. "Efficiently Enumerating All Connected Induced Subgraphs of a Large Molecular Network". In: *International Conference on Algorithms for Computational Biology*. Springer. 2014, pp. 171–182.

[McL+08] Roger McLendon et al. "Comprehensive genomic characterization defines human glioblastoma genes and core pathways". In: *Nature* 455.7216 (2008), pp. 1061–1068.

[McL+10] William McLaren et al. "Deriving the consequences of genomic variants with the Ensembl API and SNP Effect Predictor". In: *Bioinformatics* 26.16 (2010), pp. 2069–2070.

[Med+11] Paul Medvedev et al. "Error correction of high-throughput sequencing datasets with non-uniform coverage". In: *Bioinformatics* 27.13 (2011), pp. i137–i141.

[Med97] National Library of Medicine. "Free Web-Based Access to NLM Databases". In: *NLM Technical Bulletin* 296 (1997).

[MH70] Morton Mandel and Akiko Higa. "Calcium-dependent bacteriophage DNA infection". In: *Journal of molecular biology* 53.1 (1970), pp. 159–162.

[MHS12] Debora S Marks, Thomas A Hopf, and Chris Sander. "Protein structure prediction from sequence variation". In: *Nature biotechnology* 30.11 (2012), pp. 1072–1080.

[Mit+08] Patrick S Mitchell et al. "Circulating microRNAs as stable blood-based markers for cancer detection". In: *Proceedings of the National Academy of Sciences* 105.30 (2008), pp. 10513–10518.

[MK01] Mark P Miller and Sudhir Kumar. "Understanding human disease mutations through the use of interspecific genetic variation". In: *Human molecular genetics* 10.21 (2001), pp. 2319–2328.

[MM06] John S Mattick and Igor V Makunin. "Non-coding RNA". In: *Human molecular genetics* 15.suppl 1 (2006), R17–R29.

[Mol+05] Andreas Moll et al. "BALLView: an object-oriented molecular visualization and modeling framework". In: *Journal of computer-aided molecular design* 19.11 (2005), pp. 791–800.

[Mol+06] Andreas Moll et al. "BALLView: a tool for research and education in molecular modeling". In: *Bioinformatics* 22.3 (2006), pp. 365–366.

[Mon02] Luc Montagnier. "A history of HIV discovery". In: *Science* 298.5599 (2002), pp. 1727–1728.

[Mor+01] Olivier G Morali et al. "IGF-II induces rapid beta-catenin relocation to the nucleus during epithelium to mesenchyme transition". In: *Oncogene* 20.36 (2001), pp. 4942–4950.

[MP29] RV Mises and Hilda Pollaczek-Geiringer. "Praktische Verfahren der Gleichungsauflösung." In: *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik* 9.2 (1929), pp. 152–164.

[MR13] Elena Mogilyansky and Isidore Rigoutsos. "The miR-17/92 cluster: a comprehensive update on its genomics, genetics, functions and increasingly important and numerous roles in health and disease". In: *Cell Death & Differentiation* 20.12 (2013), pp. 1603–1614.

[Mue+15] Sabine C Mueller et al. "BALL-SNP: combining genetic and structural information to identify candidate non-synonymous single nucleotide polymorphisms". In: *Genome medicine* 7.1 (2015), pp. 1–8.

[Mul+16] Becky Mullinax et al. "Agilent's microarray platform: How high-fidelity DNA synthesis maximizes the dynamic range of gene expression measurements". In: *Agilent Technologies, Santa Clara, CA* (2016).

[Mur+01] Kevin Murphy et al. "The bayes net toolbox for matlab". In: *Computing science and statistics* 33.2 (2001), pp. 1024–1034.

[Mut+08] Marek Mutwil et al. "GeneCAT—novel webtools that combine BLAST and co-expression analyses". In: *Nucleic Acids Research* 36.suppl 2 (2008), W320–W326.

[MV08] Fantine Mordelet and Jean-Philippe Vert. "SIRENE: supervised inference of regulatory networks". In: *Bioinformatics* 24.16 (2008), pp. i76–i82.

[Nae+12] Haroon Naeem et al. "Rigorous assessment of gene set enrichment tests". In: *Bioinformatics* 28.11 (2012), pp. 1480–1486.

[Nak04] Shinichi Nakagawa. "A farewell to Bonferroni: the problems of low statistical power and publication bias". In: *Behavioral Ecology* 15.6 (2004), pp. 1044–1045.

[Naw14] Tal Nawy. "Single-cell sequencing". In: *Nature methods* 11.1 (2014), pp. 18–18.

[Neu92] Harold C Neu. "The crisis in antibiotic resistance". In: *Science* 257.5073 (1992), pp. 1064–1073.

[NH06] Pauline C Ng and Steven Henikoff. "Predicting the effects of amino acid substitutions on protein function". In: *Annu. Rev. Genomics Hum. Genet.* 7 (2006), pp. 61–80.

[NH11] Nicholas Navin and James Hicks. "Future medical applications of single-cell sequencing in cancer". In: *Genome Med* 3.5 (2011), p. 31.

[NH91] Anthony Nicholls and Barry Honig. "A rapid finite difference algorithm, utilizing successive over-relaxation to solve the Poisson–Boltzmann equation". In: *Journal of computational chemistry* 12.4 (1991), pp. 435–445.

[Nic+13] Stefan Nickels et al. "PresentaBALL—A powerful package for presentations and lessons in structural biology". In: *Biological Data Visualization (BioVis), 2013 IEEE Symposium on*. IEEE. 2013, pp. 33–40.

[Nik09] Hiroshi Nikaido. "Multidrug resistance in bacteria". In: *Annual review of biochemistry* 78 (2009), p. 119.

[Nis01] Darryl Nishimura. "BioCarta". In: *Biotech Software & Internet Report: The Computer Software Journal for Scient* 2.3 (2001), pp. 117–120.

[NJ03] Matthias Nicola and Jasmi John. "Xml parsing: a threat to database performance". In: *Proceedings of the twelfth international conference on Information and knowledge management*. ACM. 2003, pp. 175–178.

[NK08] Dougu Nam and Seon-Young Kim. "Gene-set approach for expression pattern analysis". In: *Briefings in bioinformatics* 9.3 (2008), pp. 189–197.

[Not15] W3C Working Group Note. *CSS Snapshot 2015*. Oct. 2015. URL: `https://www.w3.org/TR/css-2015/`.

[NP33] J. Neyman and E. S. Pearson. "On the Problem of the Most Efficient Tests of Statistical Hypotheses". In: *Philosophical Transactions of the Royal Society of London* 231 (Aug. 1933), pp. 289–337.

[NSW01] Mark EJ Newman, Steven H Strogatz, and Duncan J Watts. "Random graphs with arbitrary degree distributions and their applications". In: *Physical review E* 64.2 (2001), p. 026118.

[Oka+14] Yasunobu Okamura et al. "COXPRESdb in 2015: coexpression database for animal species by DNA-microarray and RNAseq-based expression data with multiple quality assessment systems". In: *Nucleic Acids Research* (2014), gku1163.

[OM15] Mei-Sing Ong and Kenneth D Mandl. "National expenditure for false-positive mammograms and breast cancer overdiagnoses estimated at $4 billion a year". In: *Health Affairs* 34.4 (2015), pp. 576–583.

[OS00] RISC OS. "Recent changes to RasMol, recombining the variants". In: *Biophys. Res. Commun* 266 (2000), pp. 284–289.

[OS07] Rainer Opgen-Rhein and Korbinian Strimmer. "Accurate ranking of differentially expressed genes by a distribution-free shrinkage approach". In: *Statistical Applications in Genetics and Molecular Biology* 6.1 (2007), p. 9.

[Ota+07] Yoko Otake et al. "Overexpression of nucleolin in chronic lymphocytic leukemia cells induces stabilization of bcl2 mRNA". In: *Blood* 109.7 (2007), pp. 3069–3075.

[Pae+04] J Guillermo Paez et al. "EGFR mutations in lung cancer: correlation with clinical response to gefitinib therapy". In: *Science* 304.5676 (2004), pp. 1497–1500.

[Pag+99] Lawrence Page et al. "The PageRank citation ranking: bringing order to the web." In: (1999).

[Par09] Peter J Park. "ChIP–seq: advantages and challenges of a maturing technology". In: *Nature Reviews Genetics* 10.10 (2009), pp. 669–680.

[Pau+49] Linus Pauling et al. "Sickle cell anemia". In: *Science* 110 (1949), pp. 543–8.

[PC03] Jay W Ponder and David A Case. "Force fields for protein simulations". In: *Advances in protein chemistry* 66 (2003), pp. 27–85.

[PDK15] Rob Patro, Geet Duggal, and Carl Kingsford. "Salmon: Accurate, Versatile and Ultrafast Quantification from RNA-seq Data using Lightweight-Alignment". In: *bioRxiv* (2015), p. 021592.

[Pea01] Karl Pearson. "LIII. On lines and planes of closest fit to systems of points in space". In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2.11 (1901), pp. 559–572.

[Pea09] Judea Pearl. *Causality: models, reasoning and inference*. 2nd. Cambridge Univ Press, 2009.

[Pea95a] Judea Pearl. "Causal diagrams for empirical research". In: *Biometrika* 82.4 (1995), pp. 669–688.

[Pea95b] Karl Pearson. "Note on regression and inheritance in the case of two parents". In: *Proceedings of the Royal Society of London* 58 (1895), pp. 240–242.

[Pei+04] JSM Peiris et al. "Re-emergence of fatal human influenza A subtype H5N1 disease". In: *The Lancet* 363.9409 (2004), pp. 617–619.

[Per+60] Max F Perutz et al. "Structure of hæmoglobin: a three-dimensional Fourier synthesis at 5.5-Å. resolution, obtained by X-ray analysis". In: *Nature* 185 (1960), pp. 416–422.

[Per99] Aris Persidis. "Cancer multidrug resistance". In: *Nature biotechnology* 17.1 (1999), pp. 94–95.

[Pet+04] Eric F Pettersen et al. "UCSF Chimera—a visualization system for exploratory research and analysis". In: *Journal of computational chemistry* 25.13 (2004), pp. 1605–1612.

[Pet+15] Fausto Petrelli et al. "A systematic review and meta-analysis of adjuvant chemotherapy after neoadjuvant treatment and surgery for rectal cancer". In: *International journal of colorectal disease* 30.4 (2015), pp. 447–457.

[PH05] Elizabeth Purdom and Susan P Holmes. "Error distribution for gene expression data". In: *Statistical applications in genetics and molecular biology* 4.1 (2005).

[Pis+11] Elena Piskounova et al. "Lin28A and Lin28B inhibit let-7 microRNA biogenesis by distinct mechanisms". In: *Cell* 147.5 (2011), pp. 1066–1079.

[PMK14] Rob Patro, Stephen M Mount, and Carl Kingsford. "Sailfish enables alignment-free isoform quantification from RNA-seq reads using lightweight algorithms". In: *Nature biotechnology* 32.5 (2014), pp. 462–464.

[PN02] Omar D Perez and Garry P Nolan. "Simultaneous measurement of multiple active kinase states using polychromatic flow cytometry". In: *Nature biotechnology* 20.2 (2002), pp. 155–162.

[Pol12] Paul Polakis. "Wnt signaling in cancer". In: *Cold Spring Harbor perspectives in biology* 4.5 (2012), a008052.

[PR91] Manfred Padberg and Giovanni Rinaldi. "A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems". In: *SIAM review* 33.1 (1991), pp. 60–100.

[Pra+09] TS Keshava Prasad et al. "Human protein reference database — 2009 update". In: *Nucleic Acids Research* 37.suppl 1 (2009), pp. D767–D772.

[Pre+07] William H. Press et al. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007.

[Pru+09] Kim D Pruitt et al. "The consensus coding sequence (CCDS) project: Identifying a common protein-coding gene set for the human and mouse genomes". In: *Genome research* 19.7 (2009), pp. 1316–1323.

[Qui86] J. Ross Quinlan. "Induction of decision trees". In: *Machine learning* 1.1 (1986), pp. 81–106.

[Qui93] J. Ross Quinlan. *C4.5: programs for machine learning*. Elsevier, 1993. ISBN: 978-1558602380.

[R C16] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, 2016. URL: `https://www.R-project.org/`.

[RA05] Dilip Rajagopalan and Pankaj Agarwal. "Inferring pathways from gene lists using a literature-derived network of biological relationships". In: *Bioinformatics* 21.6 (2005), pp. 788–793.

[Rav+10] Timothy Ravasi et al. "An atlas of combinatorial transcriptional regulation in mouse and man". In: *Cell* 140.5 (2010), pp. 744–752.

[RCH94] Robert Rosenthal, H Cooper, and LV Hedges. "Parametric measures of effect size". In: *The handbook of research synthesis* (1994), pp. 231–244.

[RCS13] Richard J Roberts, Mauricio O Carneiro, and Michael C Schatz. "The advantages of SMRT sequencing". In: *Genome Biol* 14.6 (2013), p. 405.

[Red+82] E Premkumar Reddy et al. "A point mutation is responsible for the acquisition of transforming properties by the T24 human bladder carcinoma oncogene". In: (1982).

[Rig+99] Guillaume Rigaut et al. "A generic protein purification method for protein complex characterization and proteome exploration". In: *Nature biotechnology* 17.10 (1999), pp. 1030–1032.

[Ril01] James C Riley. *Rising life expectancy: a global history*. Cambridge University Press, 2001.

[Rin08] Horst Rinne. *Taschenbuch der Statistik*. Harri Deutsch Verlag, 2008. ISBN: 978-3817116959.

[Rio+89] John R Riordan et al. "Identification of the cystic fibrosis gene: cloning and characterization of complementary DNA". In: *Science* 245.4922 (1989), pp. 1066–1073.

[Ris+14] Davide Risso et al. "Normalization of RNA-seq data using factor analysis of control genes or samples". In: *Nature biotechnology* 32.9 (2014), pp. 896–902. URL: `http://www.nature.com/nbt/journal/v32/n9/abs/nbt.2931.html`.

[Rit+15] Matthew E Ritchie et al. "limma powers differential expression analyses for RNA-sequencing and microarray studies". In: *Nucleic Acids Research* (2015), gkv007.

[Riv+07] Isabelle Rivals et al. “Enrichment or depletion of a GO category within a class of genes: which test?” In: *Bioinformatics* 23.4 (2007), pp. 401–407.

[Riz09] Angie Rizzino. “Sox2 and Oct-3/4: a versatile pair of master regulators that orchestrate the self-renewal and pluripotency of embryonic stem cells”. In: *Wiley Interdisciplinary Reviews: Systems Biology and Medicine* 1.2 (2009), pp. 228–236.

[RMS10] Mark D Robinson, Davis J McCarthy, and Gordon K Smyth. “edgeR: a Bioconductor package for differential expression analysis of digital gene expression data”. In: *Bioinformatics* 26.1 (2010), pp. 139–140. URL: `http://bioinformatics.oxfordjournals.org/content/26/1/139.short`.

[Roy+08] Brigitte Royer-Pokora et al. “Clinical relevance of mutations in the Wilms tumor suppressor 1 gene WT1 and the cadherin-associated protein β1 gene CTNNB1 for patients with Wilms tumors”. In: *Cancer* 113.5 (2008), pp. 1080–1089.

[Ruh00] Britta Ruhnau. “Eigenvector-centrality—a node-centrality?” In: *Social networks* 22.4 (2000), pp. 357–365.

[RY08] S Raguz and E Yagüe. “Resistance to chemotherapy: new treatments and novel insights into an old problem”. In: *British journal of cancer* 99.3 (2008), pp. 387–391.

[S+09] Dave Shreiner, Bill The Khronos OpenGL ARB Working Group, et al. *OpenGL programming guide: the official guide to learning OpenGL, versions 3.0 and 3.1*. Pearson Education, 2009.

[Sac+05] Karen Sachs et al. “Causal protein-signaling networks derived from multiparameter single-cell data”. In: *Science* 308.5721 (2005), pp. 523–529.

[Sad+10] Louis A Saddic et al. “Methylation of the retinoblastoma tumor suppressor by SMYD2”. In: *Journal of Biological Chemistry* 285.48 (2010), pp. 37733–37740.

[Sal+02] Kamran Salim et al. “Oligomerization of G-protein-coupled receptors shown by selective co-immunoprecipitation”. In: *Journal of Biological Chemistry* 277.18 (2002), pp. 15482–15485.

[Sal+04] Lukasz Salwinski et al. “The database of interacting proteins: 2004 update”. In: *Nucleic Acids Research* 32.suppl 1 (2004), pp. D449–D451.

[San+04] Albin Sandelin et al. “JASPAR: an open-access database for eukaryotic transcription factor binding profiles”. In: *Nucleic Acids Research* 32.suppl 1 (2004), pp. D91–D94.

[Sat46] Franklin E Satterthwaite. “An approximate distribution of estimates of variance components”. In: *Biometrics bulletin* 2.6 (1946), pp. 110–114.

[Saw04] Charles Sawyers. “Targeted cancer therapy”. In: *Nature* 432.7015 (2004), pp. 294–297.

[Saw08] Charles L Sawyers. “The cancer biomarker problem”. In: *Nature* 452.7187 (2008), pp. 548–552.

[SB92] Roger Sayle and Andrew Bissell. “RasMol: A program for fast, realistic rendering of molecular structures with shadows”. In: *Proceedings of the 10th Eurographics UK*. Vol. 92. 1992, pp. 7–9.

[SC92] DAVID J Spiegelhalter and ROBERT G Cowell. "Learning in probabilistic expert systems". In: *Bayesian statistics* 4 (1992), pp. 447–465.

[SCF09] Michael R Stratton, Peter J Campbell, and P Andrew Futreal. "The cancer genome". In: *Nature* 458.7239 (2009), pp. 719–724.

[Sch+15] Lara Schneider et al. "DrugTargetInspector: An assistance tool for patient treatment stratification". In: *International Journal of Cancer* (2015), n/a–n/a. ISSN: 1097-0215. DOI: 10.1002/ijc.29897. URL: http://dx.doi.org/10.1002/ijc.29897.

[Sch15a] Florian Schmidt. "CausalTrail – Testing Hypotheses Using Do-Calculus". MA thesis. Saarland University, Mar. 2015.

[Sch15b] Nicholas J Schork. "Personalized medicine: time for one-person trials". In: *Nature* 520.7549 (2015), pp. 609–611.

[Scu16] Marco Scutari. *Bayesian Network Repository*. May 2016. URL: http://www.bnlearn.com/bnrepository/.

[Sel93] Stewart Sell. "Cellular origin of cancer: dedifferentiation or stem cell maturation arrest?" In: *Environmental health perspectives* 101.Suppl 5 (1993), p. 15.

[Sen+15] E Senkus et al. "Primary breast cancer: ESMO Clinical Practice Guidelines for diagnosis, treatment and follow-up". In: *Annals of Oncology* 26.suppl 5 (2015), pp. v8–v30.

[SGS00] Peter Spirtes, Clark N Glymour, and Richard Scheines. *Causation, prediction, and search*. MIT press, 2000.

[Sha+03] Paul Shannon et al. "Cytoscape: a software environment for integrated models of biomolecular interaction networks". In: *Genome research* 13.11 (2003), pp. 2498–2504.

[Sha+09] Mark Shackleton et al. "Heterogeneity in cancer: cancer stem cells versus clonal evolution". In: *Cell* 138.5 (2009), pp. 822–829.

[Sha76] AJ Shatkin. "Capping of eucaryotic mRNAs". In: *Cell* 9.4 (1976), pp. 645–653.

[Sha93] Glenn Shafer. "The early development of mathematical probability". In: *Companion Encyclopedia of the History and Philosophy of the Mathematical Sciences* 2 (1993), pp. 1293–1302.

[She+01] Stephen T Sherry et al. "dbSNP: the NCBI database of genetic variation". In: *Nucleic Acids Research* 29.1 (2001), pp. 308–311.

[She03] David J Sheskin. *Handbook of parametric and nonparametric statistical procedures*. crc Press, 2003.

[Šid68] Zbynek Šidák. "On multivariate normal probabilities of rectangles: their dependence on correlations". In: *The Annals of Mathematical Statistics* (1968), pp. 1425–1434.

[Šid71] Zbynek Šidák. "On probabilities of rectangles in multivariate Student distributions: their dependence on correlations". In: *The Annals of Mathematical Statistics* (1971), pp. 169–175.

[Sin05] Emily Singer. "Personalized medicine prompts push to redesign clinical trials". In: *Nature medicine* 11.5 (2005), pp. 462–462.

[SLL01] Jeremy G Siek, Lie-Quan Lee, and Andrew Lumsdaine. *Boost Graph Library: User Guide and Reference Manual, The*. Pearson Education, 2001.

[SME92] EM Southern, U Maskos, and JK Elder. "Analyzing and comparing nucleic acid sequences by hybridization to arrays of oligonucleotides: evaluation using experimental models". In: *Genomics* 13.4 (1992), pp. 1008–1017.

[Smi+06] V Anne Smith et al. "Computational inference of neural information flow networks". In: *PLoS computational biology* 2.11 (2006), e161.

[Smi48] Nickolay Smirnov. "Table for estimating the goodness of fit of empirical distributions". In: *The annals of mathematical statistics* 19.2 (1948), pp. 279–281.

[Sod+07] Manabu Soda et al. "Identification of the transforming EML4–ALK fusion gene in non-small-cell lung cancer". In: *Nature* 448.7153 (2007), pp. 561–566.

[Spe04] Charles Spearman. "The proof and measurement of association between two things". In: *The American journal of psychology* 15.1 (1904), pp. 72–101.

[Sre+09] Simone T Sredni et al. "Subsets of very low risk Wilms tumor show distinctive gene expression, histologic, and clinical features". In: *Clinical Cancer Research* 15.22 (2009), pp. 6800–9. ISSN: 1078-0432.

[SS02] Bernhard Schölkopf and Alexander J Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2002. ISBN: 978-0262194754.

[SS05] Juliane Schäfer and Korbinian Strimmer. "A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics". In: *Statistical applications in genetics and molecular biology* 4.1 (2005), p. 32.

[SS08a] Daniel A Schult and P Swart. "Exploring network structure, dynamics, and function using NetworkX". In: *Proceedings of the 7th Python in Science Conferences (SciPy 2008)*. Vol. 2008. 2008, pp. 11–16.

[SS08b] Robert W Shafer and Jonathan M Schapiro. "HIV-1 drug resistance mutations: an updated framework for the second decade of HAART". In: *AIDS reviews* 10.2 (2008), p. 67.

[ST10] Kyle Strimbu and Jorge A Tavel. "What are biomarkers?" In: *Current Opinion in HIV and AIDS* 5.6 (2010), p. 463.

[Sta16] Statistisches Bundesamt. *Gesundheit — Todesursachen in Deutschland*. 2016. URL: `https://www.destatis.de/DE/ZahlenFakten/GesellschaftStaat/Gesundheit/Todesursachen/Todesursachen.html`.

[Ste+15] Zachary D Stephens et al. "Big data: astronomical or genomical?" In: *PLoS Biol* 13.7 (2015), e1002195.

[Stö+13] Daniel Stöckel et al. "NetworkTrail: a web service for identifying and visualizing deregulated subnetworks". In: *Bioinformatics* 29.13 (2013), pp. 1702–1703.

[Stö+15] Daniel Stöckel et al. "CausalTrail: Testing hypothesis using causal Bayesian networks". In: *F1000Research* 4 (2015). DOI: `10.12688/f1000research.7647.1`. URL: `http://f1000research.com/articles/4-1520/`.

[Stö+16] Daniel Stöckel et al. "Multi-omics Enrichment Analysis using the GeneTrail2 Web Service". In: *Bioinformatics* (2016), btv770.

[Str08] Korbinian Strimmer. "Comments on: Augmenting the bootstrap to analyze high dimensional genomic data". In: *Test* 17.1 (2008), pp. 25–27.

[Stu08] Student. "The probable error of a mean". In: *Biometrika* 6.1 (1908), pp. 1–25.

[Sub+05] Aravind Subramanian et al. "Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles". In: *Proceedings of the National Academy of Sciences of the United States of America* 102.43 (2005), pp. 15545–15550.

[Sub+07] Aravind Subramanian et al. "GSEA-P: a desktop application for Gene Set Enrichment Analysis". In: *Bioinformatics* (2007). URL: `https://bioinformatics.oxfordjournals.org/content/23/23/3251.short`.

[Suz+95] Kimitaka Suzuki et al. "Intraoperative irradiation after palliative surgery for locally recurrent rectal cancer". In: *Cancer* 75.4 (1995), pp. 939–952.

[SW13] Jennifer L Stamos and William I Weis. "The β-catenin destruction complex". In: *Cold Spring Harbor perspectives in biology* 5.1 (2013), a007898.

[Szk+14] Damian Szklarczyk et al. "STRING v10: protein–protein interaction networks, integrated over the tree of life". In: *Nucleic Acids Research* (2014), gku1003.

[Tay+01] Russell M Taylor II et al. "VRPN: a device-independent, network-transparent VR peripheral system". In: *Proceedings of the ACM symposium on Virtual reality software and technology*. ACM. 2001, pp. 55–61.

[Tea+15] WHO Ebola Response Team et al. "West African Ebola epidemic after one year—slowing but not yet under control". In: *N Engl J Med* 372.6 (2015), pp. 584–7.

[Tia+05] Lu Tian et al. "Discovering statistically significant pathways in expression profiling studies". In: *Proceedings of the National Academy of Sciences of the United States of America* 102.38 (2005), pp. 13544–13549.

[Tib+13] Ryan J Tibshirani et al. "The lasso problem and uniqueness". In: *Electronic Journal of Statistics* 7 (2013), pp. 1456–1490.

[Tib96] Robert Tibshirani. "Regression shrinkage and selection via the lasso". In: *Journal of the Royal Statistical Society. Series B (Methodological)* (1996), pp. 267–288.

[Tik63] Andrei Nikolaevich Tikhonov. "Regularization of incorrectly posed problems". In: SOVIET MATHEMATICS DOKLADY. 1963.

[Tim+10] Robert Timmerman et al. "Stereotactic body radiation therapy for inoperable early stage lung cancer". In: *Jama* 303.11 (2010), pp. 1070–1076.

[Tin11] Philip Tinnefeld. "Protein-protein interactions: Pull-down for single molecules". In: *Nature* 473.7348 (2011), pp. 461–462.

[TL12] Alexander Thielen and Thomas Lengauer. "Geno2pheno [454]: a Web server for the prediction of HIV-1 coreceptor usage from next-generation sequencing data". In: *Intervirology* 55.2 (2012), pp. 113–117.

[Tou+05] Kiana Toufighi et al. “The Botany Array Resource: e-Northerns, Expression Angling, and promoter analyses”. In: *The Plant Journal* 43.1 (2005), pp. 153–163.

[Tra+10] Cole Trapnell et al. “Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation”. In: *Nature biotechnology* 28.5 (2010), pp. 511–515.

[Tsa+93] RW Tsang et al. “Glioma arising after radiation therapy for pituitary adenoma. A report of four patients and estimation of risk”. In: *Cancer* 72.7 (1993), pp. 2227–2233.

[TTC01] Virginia Goss Tusher, Robert Tibshirani, and Gilbert Chu. “Significance analysis of microarrays applied to the ionizing radiation response”. In: *Proceedings of the National Academy of Sciences* 98.9 (2001), pp. 5116–5121.

[Tuc+91] MA Tucker et al. “Therapeutic radiation at a young age is linked to secondary thyroid cancer”. In: *Cancer research* 51.11 (1991), pp. 2885–2888.

[Tut48] WT Tutte. “The dissection of equilateral triangles into equilateral triangles”. In: *Mathematical Proceedings of the Cambridge Philosophical Society*. Vol. 44. 04. Cambridge Univ Press. 1948, pp. 463–482.

[Uet+00] Peter Uetz et al. “A comprehensive analysis of protein–protein interactions in Saccharomyces cerevisiae”. In: *Nature* 403.6770 (2000), pp. 623–627.

[Uli+10] Igor Ulitsky et al. “DEGAS: de novo discovery of dysregulated pathways in human diseases”. In: *PLoS one* 5.10 (2010), e13367.

[Urb+14] Achia Urbach et al. “Lin28 sustains early renal progenitors and induces Wilms tumor”. In: *Genes & development* (2014).

[Uso+15] Dmitry Usoskin et al. “Unbiased classification of sensory neuron types by large-scale single-cell RNA sequencing”. In: *Nature neuroscience* 18.1 (2015), pp. 145–153.

[Val+08] Anton Valouev et al. “Genome-wide analysis of transcription factor binding sites based on ChIP-Seq data”. In: *Nature methods* 5.9 (2008), pp. 829–834.

[Vap13] Vladimir Vapnik. *The nature of statistical learning theory*. Springer Science & Business Media, 2013.

[VB86] Peter H Von Hippel and Otto G Berg. “On the specificity of DNA-protein interactions”. In: *Proceedings of the National Academy of Sciences* 83.6 (1986), pp. 1608–1612.

[VD09] Guido Van Rossum and Fred L Drake. “PYTHON 2.6 Reference Manual”. In: (2009).

[Ven+01] J Craig Venter et al. “The sequence of the human genome”. In: *science* 291.5507 (2001), pp. 1304–1351.

[Vli+95] John Vlissides et al. “Design patterns: Elements of reusable object-oriented software”. In: *Reading: Addison-Wesley* 49.120 (1995), p. 11.

[Vuj+02] Gordan M Vujanić et al. “Revised International Society of Paediatric Oncology (SIOP) working classification of renal tumors of childhood”. In: *Medical and pediatric oncology* 38.2 (2002), pp. 79–82.

[VUR11] Fabio Vandin, Eli Upfal, and Benjamin J Raphael. "Algorithms for detecting significantly mutated pathways in cancer". In: *Journal of Computational Biology* 18.3 (2011), pp. 507–522.

[W3C06] W3C. "Extensible markup language (XML) 1.1". In: (2006).

[W3C16] W3C. *Document Object Model (DOM)*. English. W3C Web Applications Working Group. 2016. URL: `http://www.w3.org/DOM/`.

[Wal07] Francis O Walker. "Huntington's disease". In: *The Lancet* 369.9557 (2007), pp. 218–228.

[Web+92] Edwin C Webb et al. *Enzyme nomenclature 1992. Recommendations of the Nomenclature Committee of the International Union of Biochemistry and Molecular Biology on the Nomenclature and Classification of Enzymes.* Ed. 6. Academic Press, 1992.

[Weg+15] Jenny Wegert et al. "Mutations in the SIX1/2 Pathway and the DROSHA/DGCR8 miRNA Microprocessor Complex Underlie High-Risk Blastemal Type Wilms Tumors". In: *Cancer cell* 27.2 (2015), pp. 298–311.

[Wei+13] John N Weinstein et al. "The cancer genome atlas pan-cancer analysis project". In: *Nature Genetics* 45.10 (2013), pp. 1113–1120.

[Wei13] Robert Weinberg. *The biology of cancer*. 2nd ed. Garland Science, 2013.

[Wel47] Bernard L Welch. "The generalization of Student's problem when several different population variances are involved". In: *Biometrika* 34.1/2 (1947), pp. 28–35.

[Wen+13] Richard Wender et al. "American Cancer Society lung cancer screening guidelines". In: *CA: a cancer journal for clinicians* 63.2 (2013), pp. 106–117.

[Wet+02] Antoinette Wetterwald et al. "Optical imaging of cancer metastasis to bone marrow: a mouse model of minimal residual disease". In: *The American journal of pathology* 160.3 (2002), pp. 1143–1153.

[Wet15] KA Wetterstrand. *DNA Sequencing Costs: Data from the NHGRI Genome Sequencing Program (GSP)*. National Human Genome Research Institute. Oct. 2015. URL: `http://www.genome.gov/sequencingcosts/`.

[WGS09] Zhong Wang, Mark Gerstein, and Michael Snyder. "RNA-Seq: a revolutionary tool for transcriptomics". In: *Nature Reviews Genetics* 10.1 (2009), pp. 57–63.

[Win+93] Sidney J Winawer et al. "Randomized comparison of surveillance intervals after colonoscopic removal of newly diagnosed adenomatous polyps". In: *New England Journal of Medicine* 328.13 (1993), pp. 901–906.

[Wis+06] David S Wishart et al. "DrugBank: a comprehensive resource for in silico drug discovery and exploration". In: *Nucleic Acids Research* 34.suppl 1 (2006), pp. D668–D672.

[WKG83] Hans Weiher, Monika König, and Peter Gruss. "Multiple point mutations affecting the simian virus 40 enhancer". In: *Science* 219.4585 (1983), pp. 626–631.

[WKL12] Günter P Wagner, Koryu Kin, and Vincent J Lynch. "Measurement of mRNA abundance using RNA-seq data: RPKM measure is inconsistent among samples". In: *Theory in Biosciences* 131.4 (2012), pp. 281–285.

[WL11] Cindy L Will and Reinhard Lührmann. "Spliceosome structure and function". In: *Cold Spring Harbor perspectives in biology* 3.7 (2011), a003707.

[WL16] Ronald L. Wasserstein and Nicole A. Lazar. "The ASA's statement on p-values: context, process, and purpose". In: *The American Statistician* (2016). DOI: 10.1080/00031305.2016.1154108.

[WLC07] Yuliang Wu, Qiang Li, and Xing-Zhen Chen. "Detecting protein–protein interactions by far western blotting". In: *Nature protocols* 2.12 (2007), pp. 3278–3284.

[WLH10] Kai Wang, Mingyao Li, and Hakon Hakonarson. "ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data". In: *Nucleic Acids Research* 38.16 (2010), e164–e164.

[WM01] Zhen Wang and John Moult. "SNPs, protein structure, and disease". In: *Human mutation* 17.4 (2001), pp. 263–270.

[WN15] Yong Wang and Nicholas E Navin. "Advances and applications of single-cell sequencing technologies". In: *Molecular cell* 58.4 (2015), pp. 598–609.

[Wol+13] Katherine Wolstencroft et al. "The Taverna workflow suite: designing and executing workflows of Web Services on the desktop, web or in the cloud". In: *Nucleic Acids Research* (2013), gkt328.

[Wol99] Cynthia Wolberger. "Multiprotein-DNA complexes in transcriptional regulation". In: *Annual review of biophysics and biomolecular structure* 28.1 (1999), pp. 29–56.

[Won+03] Weng-Keen Wong et al. "Bayesian network anomaly pattern detection for disease outbreaks". In: *ICML*. 2003, pp. 808–815.

[WWY01] Michael P Washburn, Dirk Wolters, and John R Yates. "Large-scale analysis of the yeast proteome by multidimensional protein identification technology". In: *Nature biotechnology* 19.3 (2001), pp. 242–247.

[Yan+13] Jian-Hua Yang et al. "ChIPBase: a database for decoding the transcriptional regulation of long non-coding RNA and microRNA genes from ChIP-Seq data". In: *Nucleic Acids Research* 41.D1 (2013), pp. D177–D187.

[YB99] Daniel Yekutieli and Yoav Benjamini. "Resampling-based false discovery rate controlling multiple test procedures for correlated test statistics". In: *Journal of Statistical Planning and Inference* 82.1 (1999), pp. 171–196.

[YVK04] Yoshihiro Yamanishi, J-P Vert, and Minoru Kanehisa. "Protein network inference from multiple genomic data: a supervised approach". In: *Bioinformatics* 20.suppl 1 (2004), pp. i363–i370.

[ZH05] Hui Zou and Trevor Hastie. "Regularization and variable selection via the elastic net". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 67.2 (2005), pp. 301–320.

[Zha+14] Shanrong Zhao et al. “Comparison of RNA-Seq and microarray in transcriptome profiling of activated T cells”. In: *PloS One* 9.1 (2014), e78644.

[Zha+15] Wenqian Zhang et al. “Comparison of RNA-seq and microarray-based models for clinical endpoint prediction”. In: *Genome biology* 16.1 (2015), pp. 1–12.

[Zho+16] Linyan Zhou et al. “Rspo1-activated signalling molecules are sufficient to induce ovarian differentiation in XY medaka (Oryzias latipes)”. In: *Scientific reports* 6 (2016), p. 19543.

[ZSM04] Per-Henrik Zahl, Bjørn Heine Strand, and Jan Mæhlen. “Incidence of breast cancer in Norway and Sweden during introduction of nationwide screening: prospective cohort study”. In: *Bmj* 328.7445 (2004), pp. 921–924.