Saarland University
Center for Bioinformatics
Graduate School of Computer Science

Towards the understanding of transcriptional and translational regulatory complexity

Dissertation
zur Erlangung des Grades
des Doktors der Naturwissenschaften
der Fakultät für Mathematik und Informatik
der Universität des Saarlandes

vorgelegt von

Kerstin Neininger
(geb. Reuter)

Saarbrücken
November, 2017

Tag des Kolloquiums:	11. April 2018
Dekan der Fakultät:	Prof. Dr. Sebastian Hack
Vorsitzende:	Prof. Dr. Verena Wolf
Erstgutachter:	Prof. Dr. Volkhard Helms
Zweitgutachter:	Prof. Dr. Hans–Peter Lenhof
Akademischer Mitarbeiter:	Dr. Duy Nguyen

Abstract

Considering the same genome within every cell, the observed phenotypic diversity can only arise from highly regulated mechanisms beyond the encoded DNA sequence. We investigated several mechanisms of protein biosynthesis and analyzed DNA methylation patterns, alternative translation sites, and genomic mutations. As chromatin states are determined by epigenetic modifications and nucleosome occupancy, we conducted a structural superimposition approach between DNA methyltransferase 1 (DNMT1) and the nucleosome, which suggests that DNA methylation is dependent on accessibility of DNMT1 to nucleosome–bound DNA. Considering translation, alternative non–AUG translation initiation was observed. We developed reliable prediction models to detect these alternative start sites in a given mRNA sequence. Our tool *PreTIS* provides initiation confidences for all frame–independent non–cognate and AUG starts. Despite these innate factors, specific sequence variations can additionally affect a phenotype. We conduced a genome–wide analysis with millions of mutations and found an accumulation of SNPs next to transcription starts that could relate to a gene–specific regulatory signal. We also report similar conservation of canonical and alternative translation sites, highlighting the relevance of alternative mechanisms. Finally, our tool *MutaNET* automates variation analysis by scoring the impact of individual mutations on cell function while also integrating a gene regulatory network.

Zusammenfassung

Da sich in jeder Zelle die gleiche genomische Information befindet, kann die vorliegende phänotypische Vielfalt nur durch hochregulierte Mechanismen jenseits der kodierten DNA–Sequenz erklärt werden. Wir untersuchten Mechanismen der Proteinbiosynthese und analysierten DNA–Methylierungsmuster, alternative Translation und genomische Mutationen. Da die Chromatinorganisation von epigenetischen Modifikationen und Nukleosompositionen bestimmt wird, führten wir ein strukturelles Alignment zwischen DNA–Methyltransferase 1 (DNMT1) und Nukleosom durch. Dieses lässt vermuten, dass DNA–Methylierung von einer Zugänglichkeit der DNMT1 zur nukleosomalen DNA abhängt. Hinsichtlich der Translation haben wir verlässliche Vorhersagemodelle entwickelt, um alternative Starts zu identifizieren. Anhand einer mRNA–Sequenz bestimmt unser Tool *PreTIS* die Initiationskonfidenzen aller alternativen nicht–AUG und AUG Starts. Auch können sich Sequenzvarianten auf den Phänotyp auswirken. In einer genomweiten Untersuchung von mehreren Millionen Mutationen fanden wir eine Anreicherung von SNPs nahe des Transkriptionsstarts, welche auf ein genspezifisches regulatorisches Signal hindeuten könnte. Außerdem beobachteten wir eine ähnliche Konservierung von kanonischen und alternativen Translationsstarts, was die Relevanz alternativer Mechanismen belegt. Auch bewertet unser Tool *MutaNET* mit Hilfe von Scores und eines Genregulationsnetzwerkes automatisch den Einfluss einzelner Mutationen auf die Zellfunktion.

Acknowledgements

First of all, I would like to thank Prof. Dr. Volkhard Helms for giving me the opportunity to stay in his group and to write my PhD thesis under his supervision at the Chair of Computational Biology. The involvement in distinct research areas and projects greatly helped me to constantly expand my knowledge of current methodologies and approaches. His provision of an outstanding working environment, the possibility to attend interesting workshops and conferences, and his constant support were very encouraging. Thank you for all constructive suggestions, valuable discussions, and all your helpful advices, in particular when my ideas were digressing and a focus on the main purpose was necessary.

Moreover, I would like to thank all my collaboration partners and colleagues from different research groups. Dr. Martina Paulsen, from the (Epi)Genetics research group at Saarland University, made me familiar with the field of imprinted genes and mutation analysis in human genomic elements. Dr. Anke Steinbach, from the Helmholtz Institute for Pharmaceutical Research Saarland, introduced me to the pharmaceutical side of bacterial quorum sensing. Jun.–Prof. Dr. Tobias Marschall, from the Algorithms for Computational Genomics group, supported and complemented my work on human genetic variations in the broad field of population genetics. Prof. Dr. Jörn Walter and Karl Nordström, from the Saarland University (Epi)Genetics research group, initiated our project on DNA methylation patterns and provided us with the necessary experimental NOMe–seq dataset. This gave me the opportunity to extend my studies to the interesting field of three–dimensional X–ray structures. All these cooperations allowed an optimal combination of different expertise and to ideally tackle and solve upcoming scientific questions.

Furthermore, I would like to thank Prof. Dr. Hans–Peter Lenhof for reviewing my thesis. I would also like to thank the Saarbrücken Graduate School of Computer Science for funding me during the preparatory phase.

Special thanks are dedicated to all former and current members of the Chair of Computational Biology, in particular Thorsten Will, Jan Riehm, Michael Hutter, Rahmad Akbar, Mohamed Hamed, and Daria Gaidar. Thank you for valuable discussions, helpful suggestions, inspiring teamwork, successful tutorial and workshop organizations, and of course enjoyable lunch times. Further thanks go to Kerstin Gronow–Pudelek for always keeping an overview on all organizational issues.

Most importantly, I would like to thank my lovely family and friends, especially my parents, Pia and Werner Reuter, for their constant and never ending support in all aspects of life and for always having an open door and ear. Finally, I owe deep gratitude to my husband, Sebastian Neininger, for his unconditional love, his encouragement and support, and of course for our deep and perfect friendship through all the years.

Contents

List of Tables

List of Figures

CHAPTER 1

Introduction

Protein biosynthesis, the decoding of DNA into mRNA and proteins, is an essential process in living cells. The number of constituents that are in some way involved in these highly regulated processes is tremendous and range from epigenetic modifications, such as DNA methylation, up to the ribosome with all its subunits and additional translation factors. Moreover, with the progress in RNA sequencing, there is a constantly increasing amount of research projects concerning alternative regulatory mechanisms that complement our evidence for alternative transcription and translation initiation, alternative splicing, and alternative polyadenylation. This greatly extends our current knowledge about protein biosynthesis and supersedes our outdated views on a simple and transparent machinery. This thesis serves to improve our understanding of the complex processes and regulatory elements behind protein biosynthesis.

This chapter introduces protein biosynthesis, provides an outline on utilized data sources and bioinformatics software tools, summarizes statistical hypothesis testing, and shortly explains the background and the purpose of every project. Biological, computational, and statistical details concerning individual projects are given at the beginning of the respective chapters.

1.1 Genome organization: from double helix to condensed chromosomes

The hereditary material known as deoxyribonucleic acid (DNA) comprises the genetic information and biological instructions that are needed for development, reproduction, and cellular function as a whole in living organisms. DNA is located in the nucleus and inherited from the parent generation to the offspring. The previously unknown biological entity was first discovered in 1869 by Friedrich Miescher who isolated the molecule from the nucleus and termed it "nuclein" [1]. His work was published in 1871 [2]. Subsequent to this groundbreaking discovery, pioneers Watson and Crick [3] together with Franklin and Gosling [4] and Wilkins et al. [5] discovered in 1953 that the DNA molecule forms a two–stranded double helix.

The elementary building blocks of DNA are named nucleotides, which are composed of one of the four nitrogen bases adenine (A), cytosine (C), guanine (G), and thymine (T) that are bound to a sugar (deoxyribose) and a phosphate group [3, 6]. Figure 1.1 illustrates the ladder–like DNA double helix. Thereby, deoxyribose and phosphate alternate and form a sugar–phosphate backbone. The two–stranded helical DNA structure is built up by complementary base pairing of A with T and G with C via hydrogen bonds [3]. Thereby, A and G belong to the purine bases, whereas T and C are pyrimidine bases [3]. The complementary DNA strands are antiparallel with one strand proceeding from 5′ end to 3′ end, while the other one ranges from 3′ end to 5′ end [6]. Dependent on the direction, the strand is called sense or coding ($5' \rightarrow 3'$) and antisense or noncoding strand ($3' \rightarrow 5'$) [7]. In addition to the nuclear genome, DNA is also found in mitochondria known as mitochondrial DNA [8, 9]. Among other differences, the mitochondrial genome is circular instead of linear and has a much smaller size compared to the nuclear genome [8].

To fit into a cell, a DNA molecule folds into a complex coiled higher–order structure. This tight packaging is accomplished by formation of a hierarchical chromatin structure with DNA–

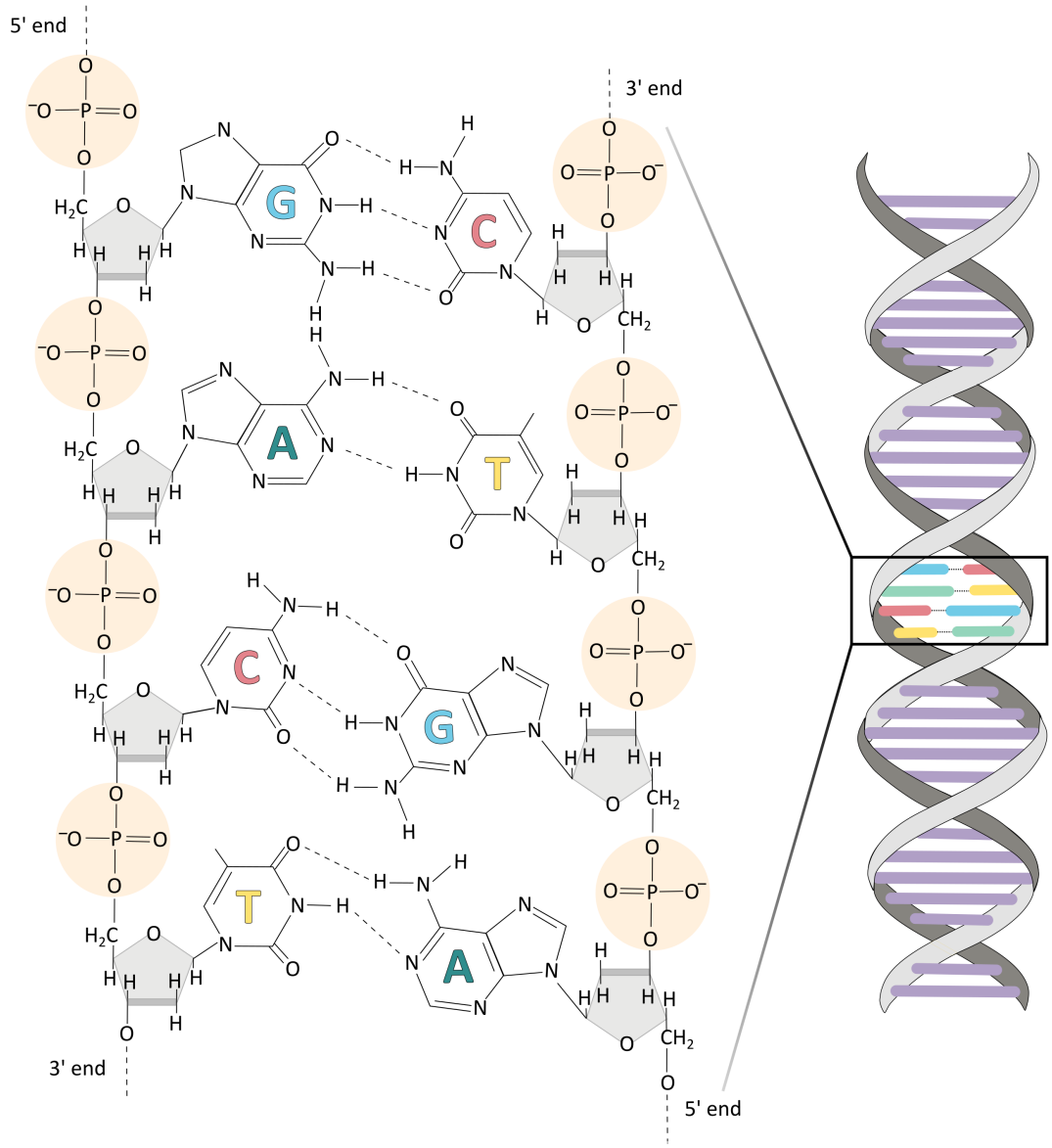

Figure 1.1: DNA double helix. DNA forms a ladder–like double helical structure that is composed of a sugar–phosphate backbone and complementary base pairings of A with T and G with C. Hydrogen bonds are shown as dashed lines. The basis of this figure was adapted from [6].

histone protein complexes, called nucleosomes, as basic repeating units [10, 11]. Nucleosomes are composed of about 145–147 base pairs (bp) of double–stranded DNA wrapped around a histone octamer. Such an octamer, as the name suggests, is composed of eight proteins with two copies of the four histone proteins H3, H4, H2A, and H2B each [10, 11, 12]. Histone variants that can replace canonical histone proteins exist as well such as histone H3.3 or histone H2A.Z, and several more known to date [13]. The amino acid sequence of the conserved histone H3.3 differs only slightly from canonical H3 histones, but histone variant H3.3 is expressed differently during the cell cycle [14]. Histone H3.3 was found to play a crucial role in mammalian development [14]. Histone variant H2A.Z is also highly conserved and plays a role in gene regulation [13]. Generally, histone variants were reported to affect chromatin states that influence gene expression as a whole [13].

Figure 1.2 shows an X–ray structure of the nucleosome core complex. Nucleosomes are then

connected via 10–80 bp linker DNA stretches, which resemble a string of beads. This is known as the primary chromatin structure [15]. Next, short–range nucleosome–nucleosome contacts enable a folding of these nucleosome beads into secondary chromatin structure [12, 16, 17]. The condensed chromosome then forms via long–range fiber–fiber interactions referred to as tertiary chromatin structure [12, 16, 17]. The human genome consists of 23 chromosome pairs, 22 pairs of autosomes and one pair of sex chromosomes, thus 46 chromosomes in total. This dynamic and adaptable chromatin structure controls genome accessibility and is thus crucial for appropriate gene regulation and hence gene expression [10, 12, 18]. There are two chromatin types: euchromatin and heterochromatin. While euchromatin, referred to as open chromatin, is prevalent at active genes, the closed and condensed chromatin, known as heterochromatin, is generally associated with less active genes and gene silencing [15, 19]. Additional chromatin modifications, such as DNA methylation, post–translational histone modifications, or chromatin remodeling complexes are essential factors that govern gene regulation [19, 20]. An erroneous regulation of chromatin structure was, for instance, found in tumor cells [10].

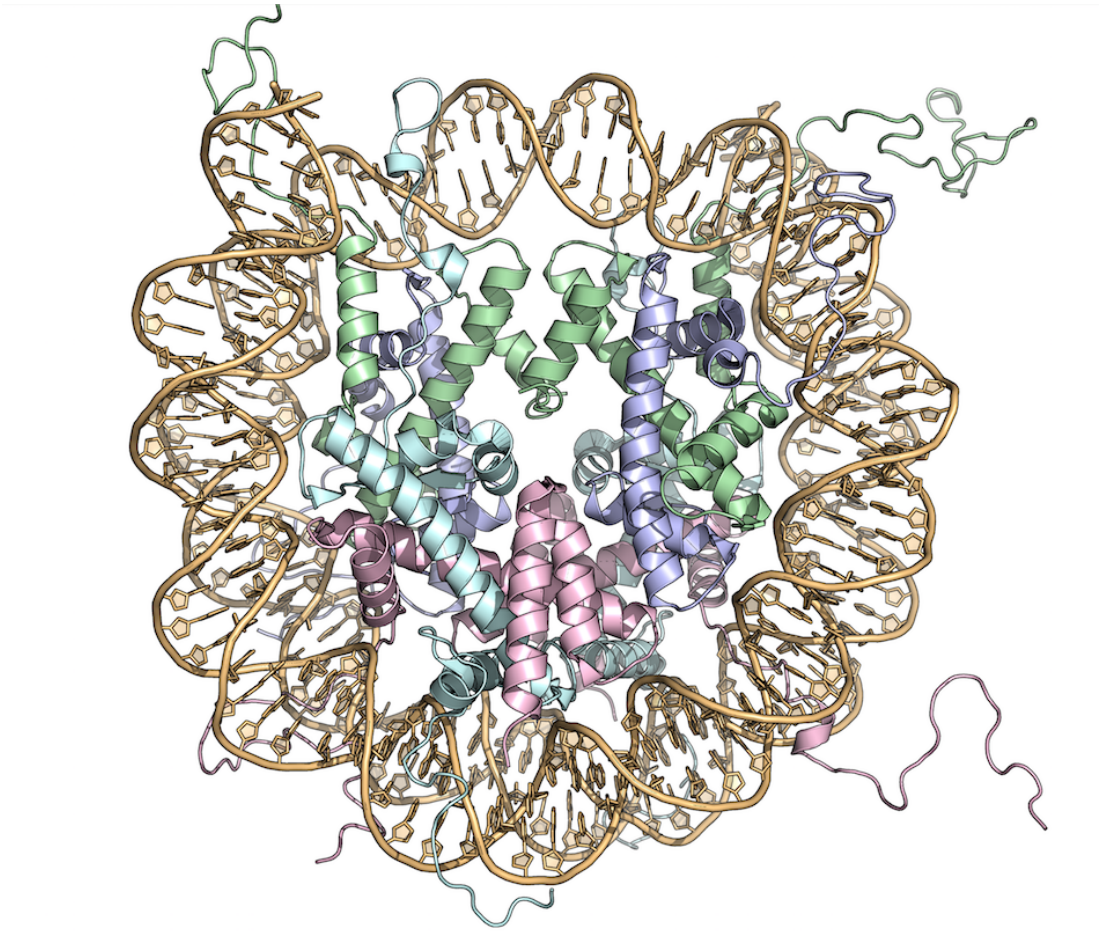

Figure 1.2: Nucleosome core complex. Shown is the DNA double helix (orange) wrapped around a histone octamer that consists of two copies of each histone protein: H3 (green), H4 (blue), H2A (cyan), and H2B (pink). Histone tails protrude out of the nucleosome core. The structure was determined by Davey et al. [10] and can be retrieved from the Protein Data Bank [21] via accession number 1KX5. The image was rendered using the PyMOL Molecular Graphics System [22].

1.2 Protein biosynthesis: evidence for alternative regulation

Protein biosynthesis describes the synthesis of protein molecules based on the genetic code. In a first step, double–stranded DNA is transcribed into single–stranded messenger RNA (mRNA) by RNA polymerase II [23]. Splicing and polyadenylation processes then ensure RNA maturation [24]. Following this, the mature mRNA molecule is translated into a protein by the ribosome scanning complex [23, 25]. Multiple regulatory mechanisms during transcription, splicing, polyadenylation, and translation allow the controlled expression of more than 80,000 protein–encoding transcripts that encode about 250,000 up to 1 Mio. proteins solely based on the genetic information of less than 20,000 genes [24]. This abundance can only be accomplished by a tight regulation together with alternative mechanisms such as an initiation at alternative transcription or translation start sites [24].

In the following, the distinct steps of protein biosynthesis together with their (alternative) regulation are explained. Since the investigation of alternative translation initiation was one of the central objectives in this thesis, translation is explained in more detail. Here, we refer to eukaryotic cells, although the described processes are very similar among eukaryotes and prokaryotes.

1.2.1 Transcription: from DNA to RNA

In a first step, the genetic information is transcribed into ribonucleic acid (RNA) whereby thymine is replaced by uracil (U), deoxyribose by ribose, and RNA is single–stranded rather than double–stranded [7]. Transcription is separated into three phases: initiation, elongation, and termination [26]. All phases are tightly regulated to guarantee a controlled and normal cell behavior during development, growth, and survival [26]. During initiation, a RNA polymerase binds to the promoter region directly upstream of a gene [23, 26]. Based on the eukaryotic DNA organization, additional proteins are required to unpack the chromatin and enable DNA accessibility [23]. A central element of eukaryotic promoter regions is the TATA box located about 25 to 35 bps upstream of the transcription initiation site [23]. This specific motif, with the consensus sequence "TATTAA", is recognized by specific transcription factors enabling proper start site usage [23]. In general, transcription factors control gene expression by recognizing and binding to specific DNA sequences (motifs) [27]. In eukaryotes, enhancer sequences additionally allow the regulation of gene expression through binding of activator proteins and the recruitment of RNA polymerase II [23]. Enhancer sequences can be located far away upstream or downstream of a gene and even within introns [27]. The distance between interacting promoter and enhancer regions is thereby defeated by DNA looping [23]. The looping is enabled via interaction of promoter– and enhancer–bound proteins, whereby activator proteins promote and repressor proteins inhibit DNA looping [23]. Additionally, multiple enhancers can influence a single RNA polymerase II promoter [27].

Upon initiation, the DNA double helix is opened and the RNA polymerase II elongates the growing RNA molecule at the 3′ end by reading along the DNA template strand [23]. In this manner, a complementary RNA chain is synthesized that grows in $5' \rightarrow 3'$ direction. The final transcript then corresponds to the DNA–coding strand, whereby T is substituted with U [23]. Transcription terminates when the RNA polymerase detects a termination site. This is then followed by release of the RNA transcript and the RNA polymerase from the DNA molecule [23, 26].

As mentioned, alternative mechanism of transcription initiation are known [24]. The usage of alternative transcription start sites and promoters can either result in varying first exons or a different 5′ untranslated region (5′ UTR) length. The resulting transcript abundance is then based on alternative open reading frames (ORFs) and alternative N–termini [24]. It was experimentally shown that the majority of gene promoters possesses several transcription start sites that are regulated and expressed dependent on the cell–type [28].

1.2.2 Splicing and polyadenylation: RNA maturation

Upon transcription, mRNAs undergo RNA splicing which results in a mature mRNA as a template for protein translation [29]. In simple terms, splicing refers to the removal of introns [29]. However, alternative splicing leads to an expansion of biological variety by combining and merging mRNA coding exons and introns in different ways. Examples are exon skipping, intron continuance, or alternative acceptor/donor sites [24]. A transcriptome–wide study estimated that about 95% of human multi–exon genes are subjected to alternative splicing [30]. Besides few exceptions, all eukaryotic mRNAs are further processed at the 3′ end by addition of a poly(A)–tail [31]. This step is known as polyadenylation. Alternative polyadenylation of transcripts can lead to different coding regions or result in diverging 3′ untranslated region (3′ UTR) lengths [24]. Moreover, alternative polyadenylation is associated with different expression levels and thus protein abundance [31, 32, 33].

1.2.3 Translation: from mature RNA to protein

Following transcription, the mature mRNA transcript is encoded based on the genetic code and, based on this mRNA template, the respective polypeptide is assembled by the ribosome [34, 35]. The mRNA sequence is translated in non–overlapping codons, or triplets, that comprise three consecutive nucleotides [25]. The decoding of a triplet into an amino acid is based on the genetic code shown in Table 1.1. All 20 natural amino acids together with their assigned one–letter as well as the three–letter code are summarized in Table 1.2. The sequential array of coding triplets then defines the open reading frame [25].

Table 1.1: The genetic code. The genetic code is a three–letter code that defines the translation from three sequential nucleotides into an amino acid [7, 35]. These consecutive nucleotides are referred to as codon or triplet. The amino acid one–letter and three–letter code is shown in Table 1.2.

	U		**C**		**A**		**G**	
U	UUU	Phe	UCU	Ser	UAU	Tyr	UGU	Cys
	UUC		UCC		UAC		UGC	
	UUA	Leu	UCA		UAA	Stop	UGA	Stop
	UUG		UCG		UAG		UGG	Trp
C	CUU	Leu	CCU	Pro	CAU	His	CGU	Arg
	CUC		CCC		CAC		CGC	
	CUA		CCA		CAA	Gln	CGA	
	CUG		CCG		CAG		CGG	
A	AUU	Ile	ACU	Thr	AAU	Asn	AGU	Ser
	AUC		ACC		AAC		AGC	
	AUA		ACA		AAA	Lys	AGA	Arg
	AUG	Met	ACG		AAG		AGG	
G	GUU	Val	GCU	Ala	GAU	Asp	GGU	Gly
	GUC		GCC		GAC		GGC	
	GUA		GCA		GAA	Glu	GGA	
	GUA		GCG		GAG		GGG	

Table 1.2: The 20 natural amino acids. Amino acids are presented using one–letter or three–letter codes, depending on which representation is more suitable in different contexts [7]. Different amino acids have different chemical and physical properties based on their side chains. Nevertheless, properties can overlap and the chemical groups shown here represent a broad classification.

Chemical group	Amino acid	Three–letter code	One–letter code
Nonpolar	Glycine	Gly	G
	Alanine	Ala	A
	Valine	Val	V
	Leucine	Leu	L
	Isoleucine	Ile	I
	Proline	Pro	P
	Phenylalanine	Phe	F
	Methionine	Met	M
	Tryptophan	Trp	W
	Cysteine	Cys	C
Polar	Asparagine	Asn	N
	Glutamine	Gln	Q
	Serine	Ser	S
	Threonine	Thr	T
	Tyrosine	Tyr	Y
Acidic	Aspartic acid	Asp	D
	Glutamic acid	Glu	E
Basic	Histidine	His	H
	Lysine	Lys	K
	Arginine	Arg	R

The ribosome is a complex, with a small and a large subunit, that accommodates ribosomal RNA (rRNA), transfer RNA (tRNA) and proteins [34]. The secondary structure of a tRNA resembles a cloverleaf with an anticodon loop that can recognize the respective codon on the mRNA transcript on one side and the matching amino acid on the other side [7]. An mRNA transcript consists of three regions: 5′ UTR or leader sequence, coding DNA sequence (CDS), and 3′ UTR [34]. In general, the ribosome scanning complex binds to the leader sequence, scans the transcript until a suitable start codon is found, and translates the sequence from 5′ to 3′ end [25, 36]. This is referred to as the linear scanning model [36]. Translation comprises three steps: initiation, elongation, and termination. Figure 1.3 illustrates canonical translation.

During initiation, the ribosomal initiation complex assembles on the mRNA transcript at the ribosome binding site that is located near the 5′ end [25]. This initiation complex is composed of eukaryotic initiation factors that bind to the small subunit of the ribosome (40S) as well as an initiator methionine–tRNA [37]. Although there are exceptions to this rule, Kozak [36] proposed a linear scanning model for eukaryotic translation stating that the 40S ribosomal subunit binds to the mRNA 5′ end, scans the transcript in 5′ to 3′ direction and initiates translation at the first AUG codon with a beneficial flanking sequence context, see below. The small

ribosomal subunit comprises three binding sites: an amino acid site (A–site), a polypeptide site (P–site), and an exit site (E–site). To provide the first protein amino acid for the growing polypeptide chain, the initiator aminoacyl–tRNA holds the covalently bound amino acid methionine and binds to the ribosome P–site located at an AUG start codon. Next, the large ribosomal subunit (60S) completes the ribosome scanning machinery by binding to the initiation complex. The initiation factors are then released. As the 40S subunit, the 60S ribosomal subunit comprises three binding sites as well, which serve for different purposes during translation. Base pairing of tRNA anticodon with the mRNA codon takes place at the A–site, the transfer of an amino acid from the tRNA to the growing polypeptide chain proceeds at the P–site, whereas the exerted tRNA is subsequently released to the cytoplasm in order to bind a new amino acid via the E–site. Except for the first amino acid, all subsequent amino acid carrying tRNAs bind to the A–site rather than the P–site.

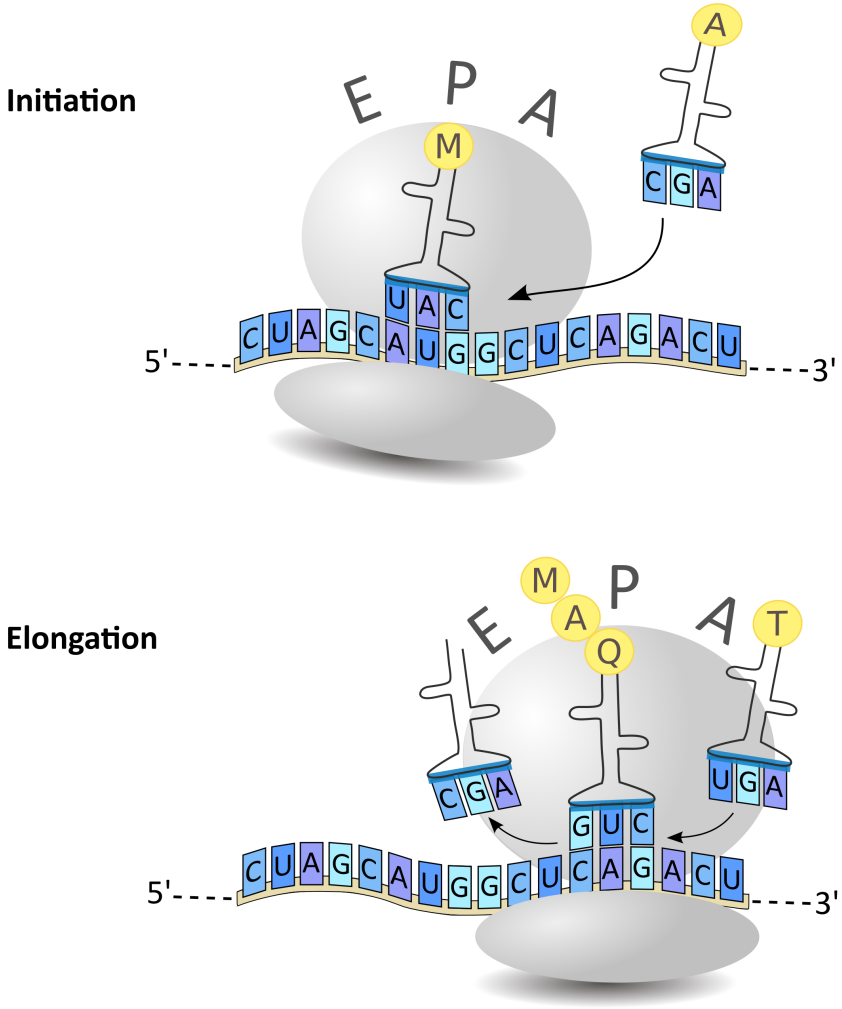

Figure 1.3: Canonical mRNA translation. The ribosome scans the mRNA sequence from 5′ end to 3′ end and translates the sequence based on the genetic code shown in Table 1.1. Depicted are the small and large ribosomal subunits together with tRNAs carrying amino acids. Note that mRNA codon and tRNA anticodon are complementary. The growing polypeptide chain is elongated with every incoming amino acid by forming peptide bonds. Incoming tRNAs bind to the A–site, are then moved to the P–site, and exit via the E–site.

Upon initiation, the growing polypeptide chain is elongated based on the triplets of the CDS and the genetic code shown in Table 1.1. Thereby, the tRNA anticodon is complementary to the transcript codon ensuring a valid translation from mRNA into protein. The next tRNA binds to the A–site and a peptide bond is formed between the incoming amino acid and the amino acid of the tRNA located at the ribosomal P–site. This reaction is dependent on elongation factors as well as the energy supplier guanosine triphosphate (GTP). The peptide bond is catalyzed at the peptidyl transferase center consisting of ribosomal RNA [38]. Thus, the ribosome is referred to as a ribozyme since RNAs, rather than proteins, are responsible for the peptidyl transferase activity [39]. After peptide bond formation, the tRNA is transferred from the A–site to the P–site for its release. This machinery of incoming tRNAs, peptide bond formation, and release of empty tRNAs into the cytoplasm is repeated until a stop codon is reached. Translation is terminated by release factors that recognize stop codons (UAA, UAG, and UGA). The completed polypeptide chain is released and the ribosome disassembles.

Alternative translation initiation, see Dever [25], Peabody [40], Kozak [41], Ivanov et al. [42], Lee et al. [43], Ingolia et al. [44], completes the collection of alternative mechanisms that guide processing from DNA to viable proteins and thus greatly contributes to biological complexity and diversity. The reliable detection of alternative initiation sites located in human 5′ UTRs is the objective of our web service *PreTIS*. Mechanistic details of alternative translation are hence elaborated in Chapter 3.

1.3 Statistical hypothesis testing

Statistical hypothesis testing allows to analyze whether there is adequate evidence that an assumed relationship between two datasets or a theory holds true (H_0: null hypothesis) or not (H_1: alternative hypothesis) based on the given data [45]. Statistical evaluations are the basis of data analysis studies and are applied in several projects in this thesis. Therefore, the Wilcoxon rank–sum test statistic [46], which was used several times, is briefly explained in the following. Subsequently, a short notice is given on the pitfalls of statistical misinterpretations, when testing multiple hypotheses or when having large sample sizes, and on how to solve these difficulties.

1.3.1 Wilcoxon rank–sum statistic

The Wilcoxon rank–sum statistic is a nonparametric test to evaluate the statistical difference between two given populations [46, 47]. The key idea is that a ranking of the given numerical data from $i = 1, 2, ..., N$ rather than the actual numerical values are used. In consequence, an estimation of the data distribution and parameters such as mean and variance can be omitted. This statistic is therefore different from parametric hypothesis tests like the Student's t–test, which assumes a normal distribution of the measurements [48, 49]. The Wilcoxon rank–sum statistic is computed as

$$U_1 = R_1 - \frac{n_1(n_1+1)}{2},$$
$$U_2 = R_2 - \frac{n_2(n_2+1)}{2}$$

with the sample sizes n_1 and n_2 as well as the sum of the assigned ranks R_1 and R_2 of the two populations. With a given significance level, for example $\alpha = 0.05$, the larger U value is then compared to the two–tailed critical value U_{crit} of the Wilcoxon rank–sum distribution. The null hypothesis H_0, which states that the two populations distributions are the same, is rejected at significance level α if one of the calculated U values is greater or equal to the critical value U_{crit}. Note that a large U_1 or U_2 results from an increased number of large ranks in one of the two populations.

1.3.2 Multiple hypothesis testing

In statistical analyses, application of several simultaneous hypothesis tests can result in statistical differences that arise only due to chance [45]. As a simple example, imagine that two groups of persons are tested for significant differences based on the following properties or features: gender, height, and age. With the addition of more and more features such as hair color, weight, favorite meal, or place of birth, you may detect that these two groups are significantly different based on one of these features just by chance. Considering a bioinformatics related example, when analyzing different genetic variant types in nine genetic elements, we tested for pairwise statistical significant differences using $n = \frac{9\times 8}{2} = 36$ hypothesis tests. Of course, a number of $n = 36$ comparisons is very small when considering that thousands or even millions of comparisons are possible when working with human genes or transcripts for example. Nevertheless, even for a small number of comparisons a correction for multiple testing is necessary. Several methods for multiple testing correction have been developed, out of which two commonly used methods, namely Bonferroni correction and Benjamini–Hochberg correction, are summarized here. Thereby, Bonferroni correction is more conservative compared to Benjamini–Hochberg correction.

Using Bonferroni correction, the null hypothesis H_0 is rejected if $p_i < \frac{\alpha}{n}$ with p_i the p–value of the i–th test, the significance level α, and the number of tests n [45]. Applying the Bonferroni method, the probability for type I error is then $\leq \alpha$ for a hypothesis test [45]. Type I error, also known as false positive case, refers to the false rejection of a true null hypothesis. Thus, this states that their is enough evidence for the assumed relationship although the relationship does not exist. Analogously, type II error, or a false negative decision, occurs if a false null hypothesis is not rejected, leading to an unrecognized but existing relationship. Note that rejecting H_0 refers to the positive case, whereas not rejecting H_0 refers to the negative case. A simple example for type I error is the detection of a disease that is actually not present, whereas type II error would not detect an existing disease.

Since Bonferroni correction is based on the attempt to decrease the probability of making even one false positive decision, it is seen as a quite conservative correction method [45]. Another widely used multiple testing correction method that relaxes this criterium is Benjamini–Hochberg correction. This approach aims at controlling the false discovery rate (FDR), which is defined as the proportion of type I errors out of all rejected null hypotheses [45]. Therefore, all p–values are sorted with $p_1 \leq p_2 \leq p_i \leq ... \leq p_n$. Next, we search for the maximum i out of all hypothesis tests such that it holds $p_i \leq \frac{i\alpha}{n}$. This is then referred to as rejection threshold t. The null hypothesis H_0 is then rejected for all comparisons i with $i \leq t$. Table 1.3 gives an example of Bonferroni and Benjamini–Hochberg correction with the example of five p–values.

Table 1.3: Multiple hypothesis testing example. Given are $n = 5$ hypotheses and a significance threshold of $\alpha = 0.05$. A rejection of H_0 is depicted by a checkmark ✓. The significance level when applying Bonferroni correction is calculated as $p_i < \frac{\alpha}{n} = \frac{0.05}{5} = 0.01$. This results in rejection of the null hypothesis for comparisons $i \in (1, 2)$. The Benjamini–Hochberg approach requires a calculation of $h_i = \frac{i\alpha}{n}$ for every hypothesis test i such that the largest i with $p_i \leq \frac{i\alpha}{n}$ can be assigned, here $i = 3$. Therefore, all comparisons $i \in (1, 2, 3)$ are assumed to be statistically significant upon application of Benjamini–Hochberg correction. The last column marks the naive approach with $p_i < \alpha$ of not applying any multiple testing correction.

i	p_i	Bonferroni	Benjamini–Hochberg		Naive
1	0.0003	✓	$h_1 = 0.01$	✓	✓
2	0.005	✓	$h_2 = 0.02$	✓	✓
3	0.02	×	$h_3 = 0.03$	✓	✓
4	0.048	×	$h_4 = 0.04$	×	✓
5	0.08	×	$h_5 = 0.05$	×	×

1.3.3 The p–value problem

As the sample size increases, p–values tend to converge to zero although the practically relevant difference between the two samples is negligible [50, 51]. To measure the practical or actual difference between two given distributions, the calculation of the effect size is helpful. In other words, a p–value only provides an estimation that there is a difference between two groups whereas the effect size reports the magnitude of the deviation. Different measurements can be used to calculate the effect size between two groups. One example is Cohen's d that measures the standardized difference between two sample means and that is not dependent on the sample size [50, 52]. Cohen's d is defined as

$$d = \frac{(\mu_1 - \mu_2)}{\sqrt{\frac{\sigma_1^2 + \sigma_2^2}{2}}} \tag{1.1}$$

with mean μ_1 and μ_2 as well as standard deviation σ_1 and σ_2 of the two distributions. The computed effect sizes can be categorized into small ($0.2 \leq d < 0.5$), medium ($0.5 \leq d < 0.8$), and large ($d \geq 0.8$) [50]. Table 1.4 illustrates the p–value problem by comparing sample size, mean μ, standard deviation σ, the p–value calculated using the Wilcoxon rank–sum statistic, and the Cohen's d value of three artificial datasets that were drawn from normal (Gaussian) distributions.

Table 1.4: The p–value problem. Shown are three artificial datasets, each representing two randomly drawn normal (Gaussian) distributions. With increasing sample size, the p–value decreases remarkably whereas the effect size reflects the actual difference between the two groups. The distributions and p–values were generated using `numpy.random.normal` and `scipy.stats.ranksums` methods provided by the Python programming language package. Cohen's d value was calculated using Formula 1.1.

	Sample size	**Mean $\mu \pm$ Std. dev. σ**		**p–value**	**Cohen's d**
1	100	79.91 ± 8.65	101.12 ± 9.28	$3.16e^{-28}$	2.36
2	100	109.34 ± 11.32	110.45 ± 10.23	0.46	0.103
3	100,000	108.99 ± 9.98	110.0 ± 9.99	$3.78e^{-108}$	0.102

Thus, a combination of the p–value, for instance calculated with the Wilcoxon rank–sum test, and a measure for the effect size such as Cohen's d is recommendable for statistical data evaluation.

1.4 Data sources and bioinformatics tools

Reliable data is the fundamental basis for every scientific study. We used several datasets either provided by public databases or by individual studies. All data sources are explained in detail prior to the individual projects in the respective chapter. Nevertheless, our data sources are shortly summarized in the following.

CpG and GpC DNA methylation data to analyze DNA methylation patterns was made available by our collaborators from the (Epi)Genetics research group located at Saarland University. This data was first utilized in Schmidt et al. [53]. Alternative translation start sites detected by experimental ribosome profiling technique were based on three individual studies, see Lee et al. [43], Calviello et al. [54], and Ingolia et al. [44]. The Ensembl Genomes project [55] served as basis for genomic mRNA sequences, whereas human genome annotations were retrieved from the UCSC Genome Browser [56]. Two large–scale genome sequencing projects,

namely the 1000 Genomes Project [57] and the Genome of the Netherlands project [58, 59], provided us with human sequence variants. Paired–end reads of *Escherichia coli* and *Staphylococcus aureus* bacterial strains were downloaded from the NCBI BioProject database [60] or taken from a former project [61], respectively. Any regulatory and antibiotic resistance information for these strains was retrieved from RegulonDB [62], AureoWiki [63], and PATRIC [64, 65].

Besides appropriate datasets, many bioinformatics software tools that cover diverse (computational) biology research fields have been developed by the scientific community. Some of these established and commonly used tools were applied throughout this thesis to facilitate analysis steps. The Basic Local Alignment Search Tool (BLAST) [66, 67] was applied for sequence similarity searches, the ViennaRNA Package [68] helped in the prediction of RNA secondary structure, MUSCLE (MUltiple Sequence Comparison by Log–Expectation) [69, 70] was used to compute sequence alignments, Bowtie [71] was utilized as a short read aligner, and VCFtools [72] supported the analysis of variation data. Moreover, a next–generation sequencing pipeline to call variants from FASTQ files required the Burrows–Wheeler Alignment (BWA) tool [73, 74], SAMtools [75, 76], and the VarScan2 software package [77].

BEDTools was used in several projects and is hence presented in the following. The BEDTools suite is a collection of highly efficient functions for comparing, annotating, and manipulating genomic data in file formats such as BED (Browser Extensible Data), SAM (Sequence Alignment Map), BAM (Binary Alignment Map), or GFF (General Feature Format) [78]. The current BEDTools version (v2.26.0) provides about 40 different tools. The most commonly used command line tools support interval arithmetic operations such as `intersectBed`, `mergeBed`, and `complementBed` to calculate the overlap between genomic intervals, to combine genomic intervals, or to report all genomic intervals not contained in the input file, respectively. We used BEDTools, more precisely `intersectBed` and `complementBed`, in several projects. The `intersectBed` function was applied to assign single nucleotide polymorphisms (SNPs) as well as insertions and deletions (indels) to genomic elements such as promoter regions, 5′ UTR, or coding region (see Chapter 4), and to assign filtered WGBS (whole–genome bisulfite sequencing) and NOMe–seq (Nucleosome Occupancy and Methylome sequencing) data to promoter regions, see Chapter 2. The `complementBed` function was applied in the latter project to derive regions with high nucleosome density compared to the local surrounding from annotated open regions. Both mentioned projects consider human data. Our tool *MutaNET* software embeds a function that overlaps SNPs with their respective genomic region within a given genome, see Chapter 5. This function is based on `intersectBed`. Since *MutaNET* is a stand–alone software package, a re–implementation rather than the integration of the original BEDTools `intersectBed` function was necessary due to copyright regulations. Moreover, the BEDTools `getFastaFromBed` function was used to extract nucleotide sequences from FASTA files given genomic coordinates, see Chapter 4.

Furthermore, Python programming language 2.7 together with the following additional packages was used in this thesis: Matplotlib [79], scikit–learn [80], pandas [81, 82], sqlite3 [83, 84], SciPy/NumPy [85, 86], Biopython [87], and seaborn [88]. Packages utilized for graphical visualizations are Matplotlib and seaborn, whereas pandas and sqlite3 were used for data storage and fast data access. The scikit–learn package is a comprehensive collection of various machine learning algorithms. SciPy is a fundamental Python package providing a large range of different functions for scientific computing. Some of the core packages of SciPy are NumPy, Matplotlib, and pandas. The Bio.PDB package [89] of the Biopython suite [87] together with the molecular visualization software PyMOL [22] was used for structural superimposition of two macromolecular X–ray structures, see Section 2.1.4 of Chapter 2. All packages used are open source. Furthermore, our developed dynamic web service is based on Hypertext Markup Language (HTML), Cascading Style Sheets (CSS), JavaScript (JS), and PHP: Hypertext Preprocessor (PHP). Detailed explanations on these publicly available data sources and software tools used throughout this thesis can be found in the respective chapters.

1.5 Objectives of this thesis

The goal of this thesis is to shed light on some central processes of eukaryotic protein biosynthesis by investigating complex DNA methylation patterns, analyzing mutation frequencies in genomic key elements, and deciphering sequence–encoded features of alternative translation initiation. Figure 1.4 illustrates the different projects and starting points of this thesis.

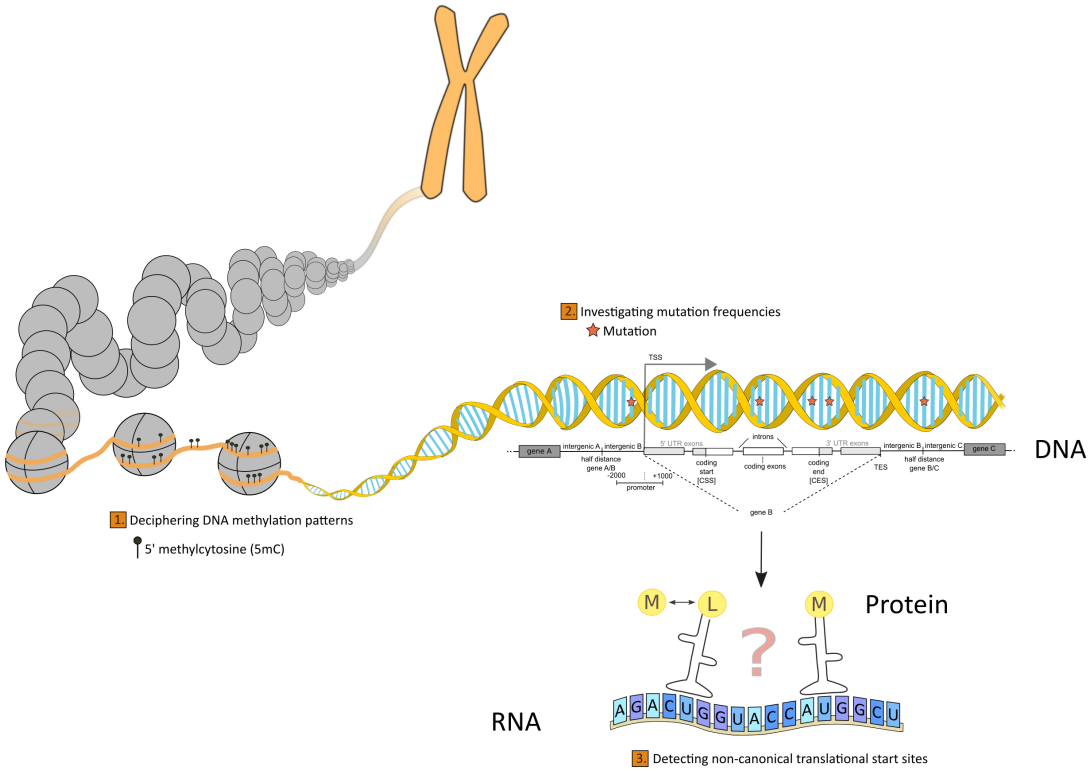

Figure 1.4: Overview of conducted projects. This thesis is separated into several projects that deal with DNA methylation patterns, mutation frequencies in genomic key elements and around transcription and translation start sites as well as alternative translation initiation. Note that the ribosome was omitted in this figure for clarity. Moreover, we automated the analysis of mutations, also integrating a gene regulatory network, to investigate their global impact on gene regulation within a given genome.

First, DNA methylation, which (in eukaryotes) mostly refers to the addition of a methyl group to a cytosine base, allows the formation of an additional gene regulatory level by not altering the DNA sequence [90]. DNA methylation plays a crucial role in cancer development when, for instance, promoter regions of tumor suppressor genes are hypermethylated [91]. DNA methylation is also found in the gene body [92]. We conducted a structural approach to explain different methylation patterns found experimentally by our collaborators from the (Epi)genetics research group located at Saarland University. The approach combines the structural organization of DNA within nucleosomes as well as the ability of DNA methyltransferases (DNMTs) to bind to specific histone–bound DNA segments in order to methylate a cytosine.

Next, alternative translation initiation, besides alternative transcription initiation, splicing and polyadenylation, contributes to the expansion of biological variety as well [24, 43, 44]. Alternative translation start sites can be be used in different tissues or in a specific cellular condition such as stress response [93]. We developed *PreTIS*, which is a web service to visualize and predict in– and out–of–frame AUG and near–cognate alternative translation start sites in human 5′ UTRs. The prediction model is based on linear regression with features derived from mRNA sequence information. All predicted start sites of one transcript

are postulated to have the potential to initiate translation. Datasets used by us are based on the experimental ribosome profiling technique. The provided web service *PreTIS* assists and considerably simplifies the analysis of mRNA sequences in terms of predicting possible translation start sites and their visualization. The *PreTIS* web service is accessible at `http://service.bioinformatik.uni-saarland.de/pretis`.

Moreover, mutations contribute to genetic variation and thus provoke phenotypic variation [7, 94, 95]. Hence, we conducted a systematic analysis of human SNPs and indels provided by the 1000 Genomes Project [57] and the Genome of the Netherlands Project [58, 59]. Beside several genomic elements such as promoter regions or coding exons, we investigated the distribution of mutations around transcription and translation start sites in more detail. Inspired by the *PreTIS* project, we intensively compared SNP patterns in direct vicinity of annotated canonical AUG and alternative translation start sites. This project was assisted by Tobias Marschall, who is affiliated with the Center for Bioinformatics Saar and the Saarbrücken Max Planck Institute for Informatics located at Saarland University.

Next, we automated mutation analysis and developed the *MutaNET* software package to score mutations in different genomic regions such as transcription factor binding sites or coding regions and estimate their impact on cellular function. The analyses were supported by the integration of an underlying gene regulatory network to assess the global impact of individual mutations on genome regulation. Our tool was then applied to several strains of *Staphylococcus aureus* and *Escherichia coli* pathogens to unravel genetic modifications underlying the development of antibiotic resistance in these bacterial strains. The *MutaNET* software can be downloaded from `https://sourceforge.net/projects/mutanet/`. Background information on the *MutaNET* software and a step–by–step tutorial is available at `http://service.bioinformatik.uni-saarland.de/mutanet/`.

Furthermore, bacterial quorum sensing (QS) was found to be targetable by modern antivirulence therapies to diminish the development of antibiotic resistance [96]. We presented the QS systems of three selected bacteria in a review article. The work on bacterial QS was supported by Anke Steinbach from the Helmholtz Institute for Pharmaceutical Research Saarland.

1.6 Projects and publications

Kerstin Reuter, Alexander Biehl, Laurena Koch, and Volkhard Helms. **PreTIS: A Tool to Predict Non–canonical 5′ UTR Translational Initiation Sites in Human and Mouse.** *PLoS Comput. Biol.*, 12(10):e1005170, 2016.

Abstract Translation of mRNA sequences into proteins typically starts at an AUG triplet. In rare cases, translation may also start at alternative non–AUG codons located in the annotated 5′ UTR which leads to an increased regulatory complexity. Since ribosome profiling detects translation start sites at the nucleotide level, the properties of these start sites can then be used for the statistical evaluation of functional open reading frames. We developed a linear regression approach to predict in–frame and out–of–frame translation start sites within the 5′ UTR from mRNA sequence information together with their translation initiation confidence. Predicted start codons comprise AUG as well as near–cognate codons. The underlying datasets are based on published translation start sites for HEK293 (human embryonic kidney 293) and mouse embryonic stem cells that were derived by the original authors from ribosome profiling data. The average prediction accuracy of true vs. false start sites for HEK293 cells was 80%. When applied to mouse mRNA sequences, the same model predicted translation initiation sites observed in mouse ES cells with an accuracy of 76%. Moreover, we illustrate the effect of *in silico* mutations in the flanking sequence context of a start site on the predicted initiation confidence. Our new web service *PreTIS* visualizes alternative start sites and their respective ORFs and predicts their ability to initiate translation. Solely, the mRNA sequence is required as input. *PreTIS* is accessible at `http://service.bioinformatik.uni-saarland.de/pretis`.

Kerstin Reuter, Anke Steinbach, and Volkhard Helms. **Interfering with Bacterial Quorum Sensing.** *Perspect. Medicin. Chem.*, 8:1–15, 2016.

Abstract Quorum sensing (QS) describes the exchange of chemical signals in bacterial populations to adjust the bacterial phenotypes according to the density of bacterial cells. This serves to express phenotypes that are advantageous for the group and ensure bacterial survival. To do so, bacterial cells synthesize autoinducer molecules, release them to the environment and take them up. Thereby, the autoinducer concentration reflects the cell density. When the autoinducer concentration exceeds a critical threshold in the cells, the autoinducer may activate the expression of virulence associated genes or of luminescent proteins. It has been argued that targeting the QS system puts less selective pressure on these pathogens and should avoid the development of resistant bacteria. Therefore, the molecular components of QS systems have been suggested as promising targets for developing new anti–infective compounds. Here, we focus on the QS systems of *Vibrio fischeri* and *Staphylococcus aureus*, and discuss various antivirulence strategies based on blocking different components of the QS machinery.

Kerstin Neininger, Tobias Marschall, and Volkhard Helms. **Mutation frequencies at transcription start sites and at canonical and alternative translational initiation sites in the human genome.** Submitted to *BMC Genomics*.

Abstract Single nucleotide polymorphisms (SNPs) are the most common form of genetic variation in humans and drive phenotypic variation. Due to evolutionary conservation, SNPs and indels (insertion and deletions) are depleted in functionally important sequence elements, such as coding exons or in flanking regions of translational initiation sites. Recently finalized population–scale sequencing efforts such as the 1000 Genomes Project and the Genome of the Netherlands Project have catalogued large numbers of SNPs and indels. Currently, there is a lack of systematic analyses of the spatial distribution of these polymorphisms. In this study we analyzed the distribution of different SNP types and indels in various genomic elements of the human genome (intergenic regions, CpG islands, promoters, 5′ UTRs, coding exons, 3′ UTRs, introns and intragenic regions) as well as around transcription and translation start sites. Indels were shown to exhibit distinct patterns in their prevalence and distribution throughout the human genome compared to SNPs. Focussing on translation start sites, we compared the SNP pattern in the flanking regions of canonical AUG start sites to that of alternative AUG and near–cognate start sites, which were identified by experimental ribosome profiling. Our analyses show that alternative translation initiation sites tend to have similar conservation profiles as canonical start sites. Most strikingly, we discovered a previously unreported accumulation of SNPs at the nucleotide position –1 directly in front of the transcription start site. By showing that alternative translation start sites exhibit similar SNP densities compared to canonical start sites, we provide further evidence for their importance in the human genome. We confirm a strong counter–selection against indels not only in protein–encoding exons, but also in elements with regulatory functions such as CpG islands. The significant enrichment of mutations just before transcription start sites, reported here and analyzed for the first time in more detail, has potential impact on models of gene regulation.

Kerstin Neininger, Karl Nordström, Jörn Walter, and Volkhard Helms. **Modelling DNMT–nucleosome–complexes to decipher DNA methylation patterns**. In preparation, 2017.

Abstract Heritable epigenetic modifications such as DNA methylation together with chromatin states are crucial for transcriptional gene regulation. This symmetric addition of a methyl group to the C–5 position of a cytosine base in CpG context is recognized by various proteins such as transcriptional regulators and is essential for genomic imprinting and X–chromosome inactivation. Aberrant DNA methylation is associated with disease phenotypes such as cancer formation as the most extensively studied example. The enzymes

that carry out DNA methylation are referred to as DNA methyltransferases (DNMTs) with DNMT1 responsible for maintenance DNA methylation and DNMT3a/3b functioning as *de novo* methyltransferases. Thereby, it was also shown that DNA can undergo methylation by DNMTs when wrapped around a histone octamer. The exact positioning of nucleosomes, the basic chromatin repeating units that are composed of 145–147 bp of DNA wrapped around a histone octamer, highly influences gene expression by making DNA accessible or unaccessible to regulatory proteins and the transcription machinery. It was reported that DNA methylation and nucleosome occupancy are highly dependent on each other, while this relationship seems to be bidirectional. NOMe–seq, short for nucleosome occupancy and methylome sequencing, is an experimental technique to detect nucleosome positions and DNA methylation at CpG sites from the same DNA strand in a genome–wide fashion. However, a complete understanding of all dependencies and influences by which dynamic methylation patterns are generated remains elusive. We hypothesize that methylation patterns and the observed varying distances between methylated CpG sites are dependent on a restricted accessibility of DNMTs to nucleosomal DNA that is caused by the structure of nucleosome core complex. We applied a structural superimposition approach of DNMT1 and the nucleosome core complex X–ray structures to determine histone–bound DNA positions at base resolution that are accessible for DNMT1–catalyzed DNA methylation. Statistical comparisons with experimental NOMe–seq data revealed that DNA methylation patterns in regions with high nucleosome density can be explained by structurally computed DNA accessibility scores.

Markus Hollander, Mohamed Hamed, Volkhard Helms, and **Kerstin Neininger**. **MutaNET: a tool for automated analysis of genomic mutations in gene regulatory networks**. *Bioinformatics*, doi:10.1093/bioinformatics/btx687, 2017.

Abstract Mutations in genomic key elements can influence gene expression and function in various ways, and hence greatly contribute to the phenotype. We developed *MutaNET* to score the impact of individual mutations on gene regulation and function of a given genome. *MutaNET* performs statistical analyses of mutations in different genomic regions. The tool also incorporates the mutations in a provided gene regulatory network to estimate their global impact. The integration of a next–generation sequencing pipeline enables calling mutations prior to the analyses. As application example, we used *MutaNET* to analyze the impact of mutations in antibiotic resistance genes and their potential effect on antibiotic resistance of bacterial strains. *MutaNET* is freely available at `https://sourceforge.net/projects/mutanet/`. It is implemented in Python and supported on macOS, Linux, and MS Windows. Step–by–step instructions are available at `http://service.bioinformatik.uni-saarland.de/mutanet/`.

1.7 Thesis outline

The aforementioned projects are each presented in individual chapters. The investigation of DNA methylation patterns using structural bioinformatics is presented in Chapter 2. This is followed by predicting alternative translation initiation starts which was addressed in our project and web service *PreTIS*. Our machine learning approach and development of the *PreTIS* web service are reported in Chapter 3. The investigation of mutations in key elements of the human genome together with a deep analysis of variations in the flanking sequences of transcription and translation start sites is depicted in Chapter 4. Analysis of these variations helped us to decipher the functional relevance of specific elements as well as individual positions in the human genome. Next, we automated the analysis of variations in a given genome by investigating the potential global impact of individual mutations on gene function. For this, we incorporated refined scoring schemes and a gene regulatory network. Our developed *MutaNET* pipeline and its application to bacterial genomes to decipher antibiotic resistance is presented in Chapter 5. Our review article on bacterial quorum sensing as alternative antivirulence therapy is presented in Chapter 6. A conclusion that summarizes our results together with an outlook is

finally given in Chapter 7. If not specified otherwise, all data integration, applied methodologies, and analysis steps were performed by me.

Prior to the presentation of our projects, each chapter starts with detailed explanations of prerequisites to relevant biological, statistical, and bioinformatics foundations. Unless otherwise specified, all (introductory) figures used in this thesis were redrawn and partially adapted for our purposes from the given references. Following the prerequisites section and prior to materials and methods as well as results and discussion sections, terms and background information are shortly recapped if necessary. A short project summary is given at the end of every chapter.

CHAPTER 2

DNMT1–nucleosome superimpositions decipher DNA methylation patterns

This chapter presents a structural approach to explain experimentally observed DNA methylation patterns. For this, we implemented a superimposition approach of DNA methyltransferase 1 (DNMT1) with the nucleosome core complex to calculate the accessibility of DNMT1 to nucleosome–bound DNA that is necessary prior to cytosine methylation. This project was in cooperation with the experimental (Epi)genetics research group from Saarland University headed by Jörn Walter. The working title of our project is currently "Modeling DNMT–nucleosome–complexes to decipher DNA methylation patterns. Kerstin Neininger, Karl Nordström, Jörn Walter, and Volkhard Helms". The manuscript will be submitted soon. My contribution was performance of all analysis steps, which encompass the structural superimposition approach, experimental data integration and filtering, the statistical evaluation and subsequent interpretation. The experimental data was provided by our collaborators from the (Epi)genetics research group. Karl Nordström provided the annotated open regions with lower nucleosome density than the local surrounding, which was inferred based on an observed higher GCH methylation, see also [53]. All authors helped in discussing the results.

2.1 Prerequisites

The term "epigenetics" was introduced in 1942 and describes heritable modifications that are not encoded in the DNA sequence [97, 98]. Examples for epigenetic modifications are DNA methylation, histone variants, or histone tail alterations. These epigenetic modifications were reported to be crucial for normal mammalian development [99]. In this project, we focused on DNA methylation and the interplay with nucleosome positioning. DNA methylation together with nucleosome positioning are associated with DNA accessibility as well as chromatin states, and hence responsible for the regulation of gene expression [100]. In the following, the establishment and maintenance of DNA methylation by DNA methyltransferases and nucleosome positioning is explained in detail. Moreover, the experimental NOMe–seq technique to detect DNA methylation and nucleosome position footprints from the same DNA strand is introduced. The PyMOL molecular visualization system [22] in combination with Biopython [87, 89], which were used for computational analysis of X–ray structures, is presented in the following as well.

2.1.1 DNA methylation: catalysis, function, and prevalence

DNA methylation plays a central role in gene regulation and is essential for cellular development. The following section specifies the term of DNA methylation, summarizes different

forms of DNA methylation, shows the distribution of methylated CpGs within the genome, and illustrates the catalysis by DNA methyltransferases.

5–methylcytosine: the 'fifth base' in DNA

DNA methylation describes the symmetric addition of a methyl group to the carbon atom at position five (C–5) of a cytosine base pyrimidine ring [90], see Figure 2.1. Since DNA methylation is involved in and crucial for various cellular processes, 5–methylcytosine (5mC) was referred to as "fifth base" in DNA [101, 102]. Thereby, S–adenosyl–L–methionine (SAMe) functions as methyl donor [103]. Demethylation of SAMe results in the formation of S–adenosyl–L–homocysteine (SAH). Hence, the catalyzed chemical reaction can be written as

$$\text{SAMe} + \text{DNA} \longrightarrow \text{SAH} + \text{DNA with 5mC}.$$

DNA methylation at CpG sites is symmetric, i.e. both cytosine bases on the opposite DNA strands are methylated during maintenance DNA methylation [99]. In mammals, DNA methylation takes place in CpG context in more than 98% of cases, although non–CpG methylation at CpA, CpT, and CpC sites was reported as well [104, 105, 106]. Thereby, methylation at CpA is more frequent compared to CpT and CpC dinucleotides [105]. Several transcriptional regulator proteins such as the CTCF transcription factor or the transcriptional repressor MeCP2, which comprises of a methyl–CpG–binding domain, can specifically detect methylated cytosines [102, 107, 108]. This emphasizes the importance of 5mC in regulation of gene expression and its designation as an additional DNA base.

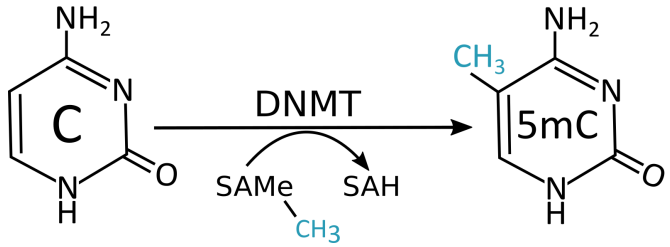

Figure 2.1: DNA methylation. A methyl group is added *in situ* to the C–5 position of a cytosine base in CpG context. Enzymes catalyzing this reaction are named DNA methyltransferases. SAMe functions as methyl group donor that becomes SAH through demethylation.

Generally, 5–methylcytosine is linked to heterochromatin, transcriptional repression, and gene silencing [102]. DNA methylation is necessary for normal cellular development [99], but is also essential for X–chromosome inactivation [109, 110], genomic imprinting [109, 111], and the repression of transposable elements [112, 113]. Genomic imprinting describes a monoallelic parent–of–origin–specific gene expression. Moreover, methylation patterns differ between cell types, tissues, and individuals [114, 115].

Hypomethylation, hypermethylation, and hydroxymethylation

One distinguishes between DNA hypomethylation and hypermethylation. Hypomethylation describes a decrease or loss of DNA methylation at CpG sites that are normally methylated, whereas hypermethylation refers to an increase of methylation at these sites. Both hypo– and hypermethylation are generally found in all types of cancer, although individual methylation patterns are cancer–specific and dependent on the tumor–stage [116, 117]. For instance, hypomethylation is frequently found in repeated DNA sequences in different cancers [118]. Moreover, promoter hypermethylation can lead to transcriptional silencing of tumor suppressor genes and thus promote cancer development [91]. A global change of DNA methylation in cancer cells further influences chromatin regulatory proteins and enzymes involved in histone

modification processes [117]. Hence, due to their specific occurrence, DNA hypomethylation patterns are used by prediction models for cancer types, stages, or the clinical course of the disease [117].

An oxidation of the 5mC group that is catalyzed by Ten–Eleven Translocation (TET) proteins was discovered as well [119, 120]. DNA 5–hydroxymethylation (5hmC) describes the addition of a $–CH_2–OH$ hydroxymethyl group, instead of a $–CH_3$ methyl group, to the C–5 position of a cytosine base. TET proteins belong to the protein family of methylcytosine dioxygenases and can oxidize 5mC to 5hmC [120, 121]. They are essential for various biological processes such as epigenetic reprogramming, development of the brain, and hematopoiesis [120, 121, 122, 123]. Mutation or deletion of the *TET2* gene was found in patients with acute myeloid leukemia [124, 125]. In general, mutagenesis of DNMT and TET enzymes seems to play an important role in cancer development [121, 126, 127].

Distribution of methylated CpGs is non–random

In general, there is a lack of CpGs in mammalian genomes, whereby 60–80 % of CpG sites are methylated [99]. It was suggested that the general lack of CpG dinucleotides is due to a higher mutation susceptibility of 5–methylcytosine to thymine via deamination [102]. Since thymine is a regular DNA base, the resulting G–T mismatch is repaired less efficiently compared to a G–U mismatch [128, 129]. In fact, only CpG islands (CGIs) that are unmethylated reflect the expected cytosine and guanine content in the genome [130]. CGIs exhibit an increased CG density while at the same time not even 10% of all CpGs are found there [99]. CGIs often overlap with promoter regions of specific gene classes such as housekeeping genes, tissue–specific genes, or genes encoding developmental regulators [99, 131]. The association of promoter regions with CGIs is frequently found (about 70%) in vertebrates [132]. Deaton and Bird [133] speculate that all CGIs could function as transcription initiation sites. The euchromatic CGI structure, which allows binding of transcription factors to promoter regions, strengthens this assumption [134]. As mentioned, 60–80 % of CpG dinucleotides are methylated [99]. An exception are those CGIs that are normally hypomethylated in all cell types [99]. This continuous demethylation of CGIs in mammalian genomes requires regulatory mechanisms. It was reported that the methylation–free state of CGIs seems to be associated with local transcription factor binding [135]. In general, DNA methylation seems to be locally dependent on DNA–binding factors [136].

DNA methyltransferases: de novo and maintenance DNA methylation

The enzyme family that is responsible for DNA methylation are DNA methyltransferases (DNMTs) [103]. The transfer of a methyl group to the C–5 position of a cytosine base is catalyzed by three DNMT enzymes that are active in the human genome: DNMT1, DNMT3a, and DNMT3b. They belong to two enzyme classes: maintenance and *de novo* DNA methyltransferases [102, 103]. While DNMT1 is responsible for maintenance DNA methylation to reestablish DNA methylation patterns during cell replication [102, 137], DNMT3a and DNMT3b are responsible for *de novo* DNA methylation [102, 138]. Another DNMT enzyme, namely the DNA methyltransferase 3–like protein (DNMT3L), is catalytically inactive and functions as regulatory factor [138, 139]. DNMT3L is paralogous to the DNMT3 enzymes [140].

The maintenance DNA methyltransferase DNMT1 targets hemimethylated CpG sites and is composed of several protein domains [137]. Hemimethylation describes the case when only one of the opposing CpG dinucleotides on both DNA strands is methylated [99]. Methylated cytosines form a base pair with guanine and are thus replicated as unmethylated cytosines. Therefore, methylation must be maintained by subsequently adding a methyl group to the replicated cytosine based on the hemimethylated template [102, 141, 142]. This enables that established methylation patterns are sustained during cell replication. Song et al. [137] determined the crystal structures of mouse (residues 650–1602) and human (residues 646–1600) DNMT1 bound to 19 base pair hemimethylated DNA and complexed with SAH, see Figure 2.2.

DNMT1 enzymes are composed of several protein domains: the N-terminal replication foci–targeting domain, the Cys–X–X–Cys (CXXC) domain that recognizes and binds to unmethylated CpG sites, the tandem bromo–adjacent homology (BAH1/2) domains that are connected by an alpha helix, and the C–terminal catalytic methyltransferase domain. The DNA-binding CXXC protein domain is highly conserved, binds to DNA with unmethylated CpGs sites, and is found in proteins that are involved in epigenetic regulation [99, 143]. The methyltransferase domain is separated into a catalytic core and a target recognition domain and is adjacent to the CXXC and BAH1 domains. An autoinhibitory linker connecting CXXC and BAH1 domains was found to shield unmethylated dinucleotides from the active site and in doing so ensures that only hemimethylated CpG dinucleotides, rather than unmethylated sites, are catalyzed during maintenance DNA methylation [137].

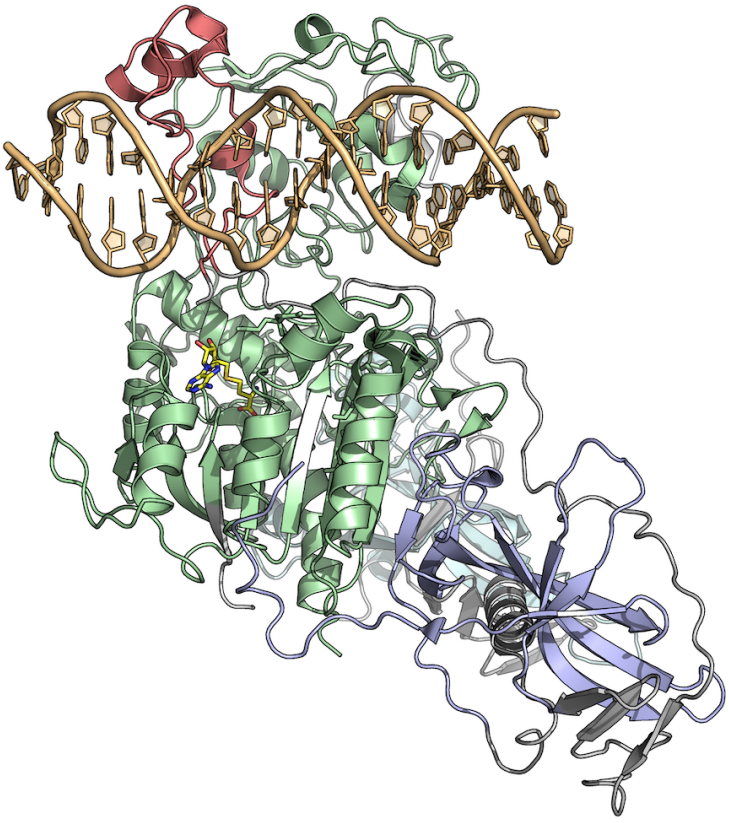

Figure 2.2: DNMT1 in complex with DNA. The DNMT1 structure was determined by Song et al. [137]. The structure was retrieved from the Protein Data Bank [21] via accession number 3PTA. Shown are the different DNMT1 domains: CXXC (red), BAH1 (purple), BAH2 (cyan), and the methyltransferase domain (green). Linker structures are shown in grey color. DNMT1 is bound to a hemimethylated 19 bp DNA stretch (orange). SAMe is presented as yellow sticks. The figure was adapted from Song et al. [137] and rendered using the PyMOL Molecular Graphics System [22].

The *de novo* DNA methyltransferases, DNMT3A and DNMT3B, are composed of a Pro–Trp–Trp–Pro (PWWP) domain, an ATRX–DNMT3–DNMT3L (ADD) domain, and a C–terminal catalytic methyltransferase domain [138, 144]. The regulatory protein DNMT3L is needed for activation of DNMT3 enzymes [140]. The PWWP domain is highly conserved and crucial to target pericentric heterochromatin [145]. The ADD domain is part of an autoinhibitory mechanism repressing the binding affinity of the methyltransferase domain towards the DNA [138]. The crystal structure of the DNMT3A–DNMT3L complex was solved by Guo et al. [138] and is shown in Figure 2.3. The complex is composed of a DNMT3L–DNMT3A–DNMT3A–DNMT3L tetramer

having a central DNMT3A–DNMT3A interface as well as two lateral DNMT3L–DNMT3A interfaces. It was reported that this complex has two active sites with a distance of one DNA–helix turn [138]. This leads to periodic methylation patterns with about eight and ten base pairs between two methylated CpG dinucleotides.

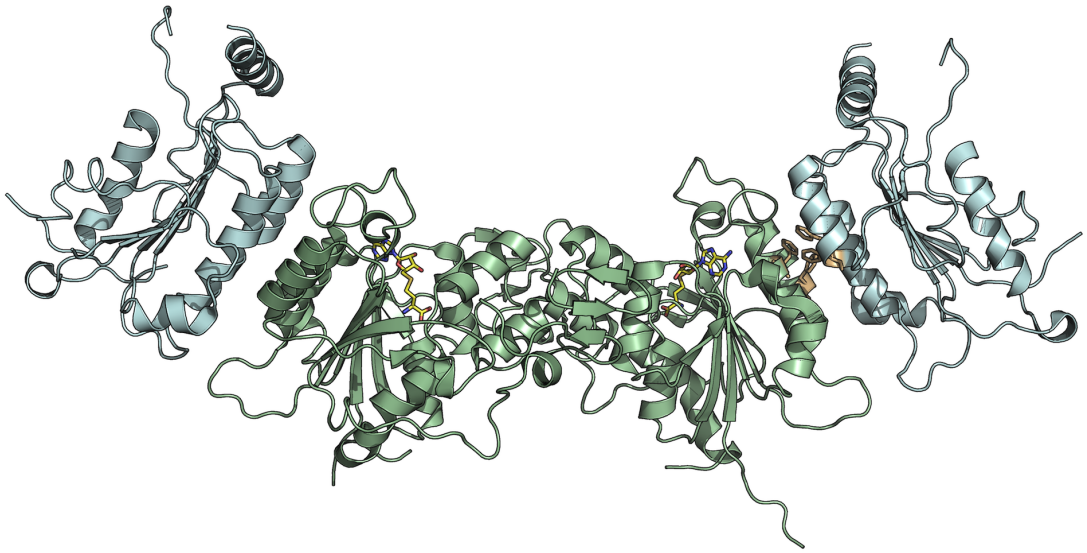

Figure 2.3: C–terminal tetrameric DNMT3A–DNMT3L structure. Shown is the DNMT3A (green, residues 623–908) structure in complex with DNMT3L (blue, residues 160–386). SAH (yellow) is presented as stick structure. The tetrameric DNMT3L–DNMT3A–DNMT3A–DNMT3L structure was determined by Jia et al. [140] and can be retrieved from the Protein Data Bank [21] via accession number 2QRV. The DNMT3A–DNMT3L interface with four phenylalanine residues is shown in orange color. The figure was adapted from Jia et al. [140] and rendered using the PyMOL Molecular Graphics System [22].

2.1.2 Nucleosome positioning: the key to genome regulation?

Nucleosome occupancy is highly organized, non–random, and the individual genomic nucleosome positions are essential for regulatory mechanisms [102]. Nucleosome positioning is required to be adjustable to guarantee dynamic gene regulation [102]. Thereby, rotational and translational settings refer to the arrangement of the DNA double helix and the positions of individual nucleosomes at a genomic locus, respectively. The rotational orientation refers to the wrapping of the DNA double helix around the histone octamer which is facing towards the core and outwards in an alternating manner [146]. The rotational setting is important for cellular function since only DNA bases facing away from the histone octamer are accessible to proteins or regulatory factors. The translational positioning relates to the favored nucleosome occupancy on a DNA sequence [146]. In total, the human genome is assumed to harbor about 15 million nucleosomes [147], while it is estimated that about 75–90% of DNA is nucleosome–bound [148]. The ordered and intrinsically favored nucleosome positions on a DNA sequence is known as phasing [146]. The experimental techniques ChIP–on–chip and chromatin immunoprecipitation combined with DNA sequencing (ChIP–seq) can be applied to determine whether a protein is bound to a specific location within a given genome [146, 149]. Thus, these technologies are also applicable to detect nucleosome positions and greatly helped in deciphering chromatin states [146]. ChIP–on–chip is based on chromatin immunoprecipitation and DNA microarray analysis [146, 149, 150], whereas ChIP–seq is a combination of chromatin immunoprecipitation and high–throughput DNA sequencing [149, 151, 152, 153].

Nucleosomes prefer specific base compositions

As the ability of the DNA double helical structure to bend is also dependent on the underlying sequence, feasible positions a nucleosome can reside in must be, to some extent, predetermined in the entire genome [102]. Thereby, a sharp DNA bending is necessary to enable an establishment of the nucleosome structure with its tightly wrapped DNA sequence [148, 154]. Since the ability of a DNA sequence to bend sharply depends on particular base compositions, a general association between nucleosome occupancy and underlying DNA sequence is crucial [148, 154, 155]. Periodic A/T–rich sequences forming two H–bonds between each base pair (see Figure 1.1) were found to be enriched in DNA minor grooves that face inwards in direction of the histone octamer, while G/C–rich sequences forming three H–bonds were prevalent at positions that face away from the octamer [147, 148, 154, 156]. While dinucleotides composed of A and T can broaden the DNA major groove, GC dinucleotides can narrow it [146]. Both properties are essential for an appropriate DNA bending and nucleosome formation and these DNA patterns were found to exhibit a periodicity of about 10 bp [147, 148]. Segal et al. [148] could explain approximately 50% of nucleosome positions in yeast solely based on the composition of nucleosome–preferred DNA sequence. They referred to this as the "nucleosome positioning code", which is assumed to be also necessary for binding of regulatory factors and initiation of transcription [148].

In this sense, Jiang and Pugh [146] compared the genome–wide nucleosome occupancy with the analogy of a roulette wheel in which the positions for the roulette ball are dictated. Independent of the number of nucleosomes that are placed onto the DNA, specific base compositions determine the canonical positions. This model was referred to as independent positioning model due to an unbiased positioning of neighboring nucleosomes [146]. Considering another model, which was declared as statistical positioning, a single fixed nucleosome determines the positions of all sequential nucleosomes [146]. Since these nucleosomes are organized in an array, movements to both sides are restricted. The distribution of nucleosomes in this array then resembles a probabilistic density distribution that does not rely on specific DNA sequence positions. It is suggested that the truth lies somewhere in between the one with a predetermined nucleosome boundary and the one of an array of nucleosomes that line up in a probabilistic way [146]. It was observed that especially nucleosomes around transcription start sites (TSSs) show a specific positioning pattern [100, 146]. Due to their explicit position around transcription start sites, these nucleosomes are numbered accordingly. The first nucleosome that is located upstream of the TSS is given number –1, while number +1 is assigned to the first (fixed) nucleosome downstream of the TSS that could function as predetermined start in statistical nucleosome positioning [146]. Figure 2.4 illustrates the nucleosome distribution around a TSS. Nucleosome –1 resides in the promoter region between –300 and –150 bp according to the TSS and can thus affect transcription initiation [146]. Thus, the flanking sequences around a TSS might harbor predetermined canonical nucleosome positions necessary for gene regulation, whereas nucleosome occupancy becomes more blurry in downstream direction when entering the gene body [146].

DNA methylation and nucleosome occupancy are interconnected

Jiang and Pugh [146] state that an understanding of the rules behind nucleosome positioning within the genome would explain gene regulation and shed light on aberrant transcriptional regulation found in diseases such as cancer. Beside sequence preferences, an association of nucleosome occupancy and epigenetic factors such as DNA methylation was observed as well [147, 157, 158]. Nevertheless, the knowledge of dependencies between DNA methylation and nucleosome occupancy is still incomplete. It was even reported that analysis of *in vivo* and *in vitro* data gave different results: while methylated linker DNA was prevalent in *in vivo* data, an increased methylation of nucleosome core DNA was observed in *in vitro* settings [147]. The experimental NOMe–seq technology to detect nucleosome occupancy and DNA methylation from the same DNA molecule is explained below.

Collings and Anderson [147] investigated nucleosome positioning and DNA methylation

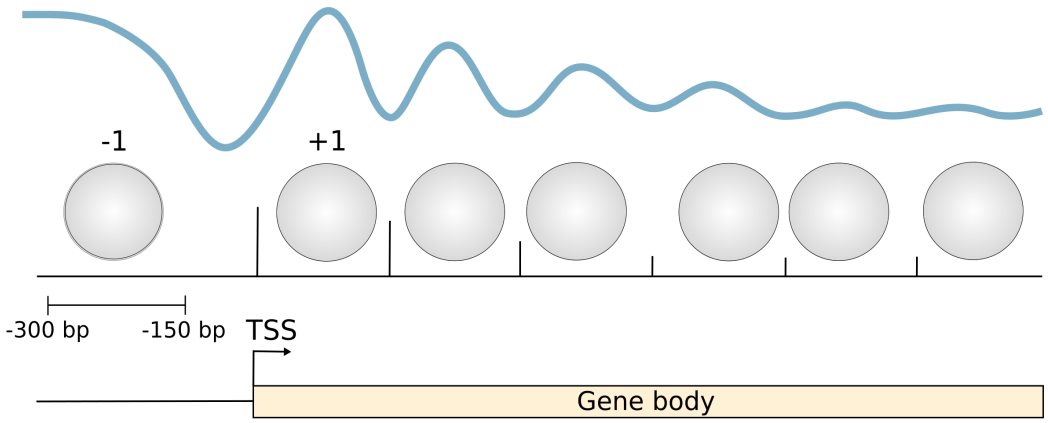

Figure 2.4: Nucleosome positioning around the TSS. Nucleosomes –1 and +1 are specifically placed relative to the TSS to enable transcriptional regulation. With the begin of the gene body, the distribution within the nucleosome array becomes fuzzier. The –1 nucleosome was reported to reside about –300 bp to –150 bp upstream of the TSS [146]. The figure was adapted from [146].

at CpG sites based on MNase–seq and NOMe–seq data in the human genome. They found a positive correlation between nucleosome occupied positions and the frequency of methylated CpG sites. They concluded that DNA methylation and nucleosome positioning are dependent on the underlying methylated CpG density. Thereby, a deviation from this pattern could be attributed to other epigenetic factors [147]. These results suggest that, by considering an unidirectional relationship, the methylated CpG density could regulate nucleosome positioning. Portela et al. [157] showed that the relationship between DNA methylation and nucleosome occupancy is even bidirectional. This means that methylated DNA sequences can dictate nucleosome positions, while genomic sequences occupied by nucleosomes can influence the methylome as well. They studied hypermethylated promoter CpG islands of tumor suppressor genes that are often found in cancer cells due to their gene silencing ability [91, 157].

Moreover, a periodicity of ten base pairs between methylated CpG sites of nucleosomal DNA was identified [158]. This distribution was found to be independent of the genomic region and thus could present a general pattern. Moreover, the same study found a higher methylation density in nucleosome–bound DNA compared to flanking DNA sequences. Based on this distribution, they suggested that DNMTs prefer nucleosomal DNA rather than flanking sequences. As the aforementioned studies, they also came to the conclusion that a general influence of nucleosome positions on DNA methylation exists [158]. Beside specific nucleosome positioning around transcription start sites, a higher nucleosome density in exonic regions with a preference on intron–exon boundaries was observed as well [158]. This nucleosome preference together with additional higher methylation rates in exons provides an argument that exons are tagged by methylated DNA [158]. Both, the higher amount of nucleosomes and DNA methylation in exonic regions emphasizes that DNMTs preferentially target nucleosomal DNA rather than flanking DNA and that CpG methylation takes place on nucleosomes [158]. Following this, the observed ten base pair periodicity can be explained by this specific wrapping of the DNA double helix around a histone octamer [158].

2.1.3 NOMe–seq reveals methylome and chromatin states

Kelly et al. [100] developed the method termed NOMe–seq (Nucleosome Occupancy and Methylome sequencing). This is an experimental technique that provides genome–wide nucleosome position footprints together with DNA methylation at CpG sites, both from the same DNA molecule. A GpC methyltransferase (M.CviPI) [159] methylates GpC sites that are not occupied by nucleosomes, whereas endogenous DNA methylation is detected at CpG sites.

Thereby, methylation at GpC sites is determined in GCH context, whereas CpG methylation frequencies are reported in HCG context (H=A, C, or T). The combination of GpC and CpG methylation information provides four epigenetic and chromatin states: regions are either nucleosome occupied or depleted and CpG sites are either methylated or unmethylated. All four chromatin structures are visualized in Figure 2.5. Whole–genome bisulfite sequencing [160] is then applied to enable a distinction between methylated and unmethylated sites. Thereby, the individual methylation frequency at a cytosine base is calculated as the number of methylated reads divided by all reads found at the respective position.

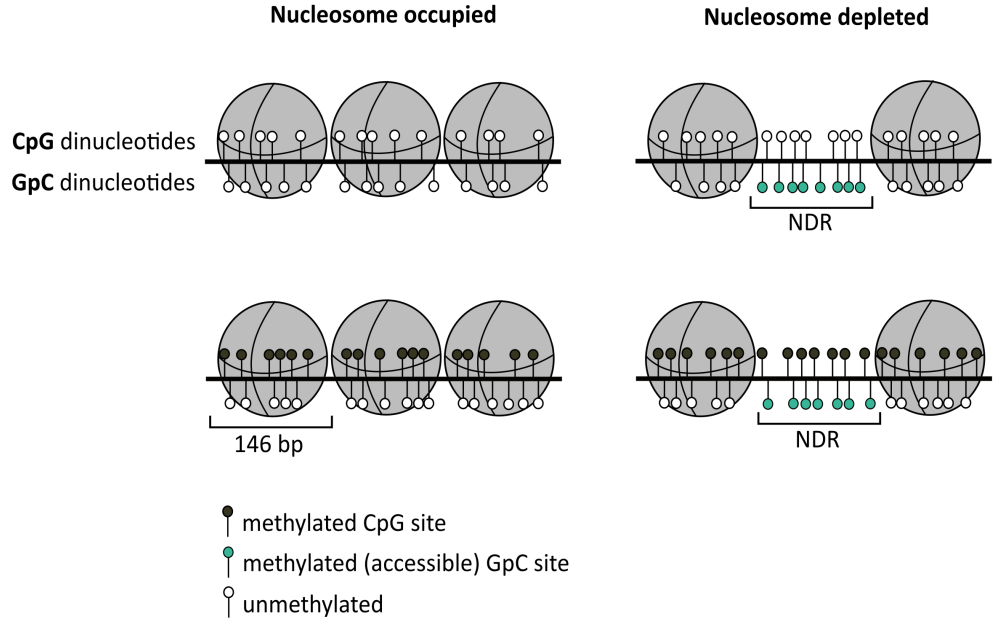

Figure 2.5: NOMe–seq experimental technique. Genome–wide information on nucleosome positioning together with endogenous DNA methylation can be obtained by NOMe–seq from the same DNA strand [100]. The combination of GpC and CpG methylation profiles allows to decipher four different chromatin structures: nucleosome occupied and depleted regions with methylated or unmethylated CpG dinucleotides. NDR stands for nucleosome depleted region. The figure was adapted from [100].

2.1.4 X–ray structure analysis with PyMOL and Biopython

The visualization program PyMOL in combination with the Biopython Bio.PDB module were used for processing, analysis, and final representation of the DNMT1 enzyme and the nucleosome core particle NCP147, both stored as three–dimensional PDB structures. The Bio.PDB module was used for the structural superimposition of DNMT1 and NCP147, which is presented in the following.

PyMOL molecular visualization system

PyMOL is a molecular visualization system for three–dimensional (macro)molecular structures such as proteins [22]. PyMOL can be applied to represent molecular structures in various ways including spheres and surfaces, and to render high quality ray–traced images. Structural data can be given, for instance, in the widely used Protein Data Bank (PDB) file format [21]. Moreover, PyMOL applications are expandable by Python programming language via an embedded command line interface. PyMOL together with the Bio.PDB package [89] of the Biopython

suite [87] were applied to perform a structural superimposition between DNMT1 and the nucleosome core complex for every nucleosome–bound DNA position.

Biopython in structural bioinformatics

The Bio.PDB module of the Biopython Project aims at facilitating the field of structural computational molecular biology by providing various methods for straightforward analysis of crystal structures provided as PDB files [89]. As the name suggest, Biopython is implemented in Python. We made use of this comprehensive module to superimpose DNMT1 bound to a short DNA stretch with the nucleosome core complex NCP147 to decipher DNA methylation patterns. In the following, the Bio.PDB module is described and the provided classes and methods are explained with regard to our superimposition approach that is outlined in Algorithm 2.1. Functions used in Algorithm 2.1 that are highlighted in bold and italic were provided by the Bio.PDB module.

A reasonable way to make a PDB structure available for structural computations is the construction of a structure object from a given PDB file, see Algorithm 2.1 (Lines 1 and 2). A structure object is based on the so–called SMCRA architecture. SMCRA stands for Structure–Model–Chain–Residue–Atom and hence illustrates the composition of a crystal structure by considering that these structures can be composed of model(s), a model exhibits chain(s), a chain exhibits residues, and residues are assembled by atoms. A structure object is created from a PDB file using `parser=Bio.PDB.PDBParser()` and applying `parser.get_structure("1KX5", "1KX5.pdb")` to a PDB query file. This will return a structure object of the PDB file with identifier "1KX5". Iterations over a structure object are then straightforward by using

```
for model in structure:
   for chain in model:
      for residue in chain:
         for atom in residue:
            \\ do something.
```

Many methods are then applicable to individual entities such as structures, chains, residues, and atoms. For instance, `model=structure[0]` allows specific access to a model, whereas `chain=model["A"]` can be used to directly extract a specific chain, here chain A. Since crystal structures are commonly composed of only one model, it is convenient to access the model at position zero and thus omit the first `for` loop in the example above. This simplification was also applied when we were working with the three–dimensional nucleosome (PDB identifier 1KX5) and the DNMT1 (PDB identifier 3PTA) structures. Moreover, parent entities can be reached using `get_parent()`, like `residue=atom.get_parent()`. It is also possible to retrieve all `atoms` or `residues` of a structure object by `structure.get_atoms()` and `structure.get_residues()`, respectively. Furthermore, atomic coordinates can be retrieved using `atom.get_coord()` and distances `d` (in angstrom Å) can be easily measured between two atoms by the minus operator via `d=atom1-atom2`.

Two crystal structures can be superimposed using the `Superimposer` class and constructing a `sup=Superimposer()` object, compare with Algorithm 2.1 (Line 6). Thereby, the rotation and translation matrix for the necessary transformations of the atomic coordinates is calculated such that the root–mean–square deviation (RMSD) of the two overlapping query structures is minimal. This minimization is done by the `sup.set_atoms(fixed, moving)` method with two lists of fixed (here nucleosome) and moving (here DNMT1) atoms as arguments. Thereby, as the names suggest, the moving DNMT1 atoms are put on top of the fixed nucleosome atoms. The calculated rotations and translations are then applied to the complete structure using `sup.apply(structure_atoms)`. To ensure a reliable superimposition between DNMT1 and NCP147, we incorporated atoms belonging to the DNA stretch of DNMT1 as well as to the respective DNA stretch (starting at position 1) of the DNA wrapped around the nucleosome. For convenience, we only considered central DNMT1 DNA atoms (positions 4 to 15 and 24 to 35 in

Algorithm 2.1 Superimposition(*nucleosome.pdb*, *dnmt.pdb*).

```
Init structure object of nucleosome from PDB file: nuc_structure
Init structure object of DNMT1 from PDB file: dnmt_structure
dnmt_bb_atoms ← Iterate dnmt_structure, extract DNA backbone atoms
while i ≤ end of wrapped DNA do
    nuc_bb_atoms ← Iterate nuc_structure, extract DNA_i backbone atoms
    sup ← Superimposer()
    sup.set_atoms(nuc_bb_atoms, dnmt_bb_atoms)
    sup.apply(dnmt_structure.get_atoms())

    {Save the edited (e.g. colors, helix representations) structures of shifted
    DNMT1 and nucleosome structures as one PDB file using the PDBIO()
    module and the Python pymol package.}

    nuc_atom_list ← unfold_entities(nuc_structure, 'A')
    dnmt_atom_list ← unfold_entities(dnmt_structure, 'A') {Atoms shifted
    based on rotation and translation matrix.}

    ns ← NeighborSearch(nuc_atom_list)
    for all dnmt_atom in dnmt_atom_list do
        center_coords ← dnmt_atom.get_coord()
        neighbors ← ns.search(center_coords, 5.0)
        for all neighbor_atom in neighbors do
            Init vdW radius of atom_dnmt.element: r1
            Init vdW radius of neighbor_atom.element: r2
            d_radius ← r1+r2
            if distance < d_radius then {sterical clash}
                ▷ Save information about clashing atoms.
            end if
        end for
    end for
    i ← i + 1
end while
return {Sterical clashes for every nucleosome position are stored.}
```

the PDB file) of the sugar phosphate backbone. This superimposition is then repeated for every wrapped DNA position in nucleosome core complex NCP147, see Algorithm 2.1 (Line 4).

Besides opening and parsing PDB structures, the Bio.PDB module also provides a class, `io=PDBIO()`, for writing a PDB file from a structure object. In addition to saving the file via `io.set_structure(structure)` followed by `io.save("outname.pdb")`, the structure can be modified by, for instance, changing colors or helix representations. This is provided by the Python pymol package and applying commands such as `pymol.cmd.show_as("cartoon")` to visualize the crystal structure as cartoon. We made use of this functionality to edit (for example specific coloring and modification of helix representations) and save both, DNMT1 and nucleosome structure, to a single PDB file.

Since we were interested in the accessibility of DNMT1 to hemimethylated DNA that is wrapped around a histone octamer (for details see below), the calculation of sterical clashes between these two structures was reasonable. The `NeighborSearch()` object, which is based on a KD tree data structure, can be applied to report all entities, like atoms or residues, that are detected within a radius of a given atomic position. A KD tree data structure, for a k–dimensional tree, is based on binary space splitting into half–spaces using separating hyperplanes [161]. This organization then allows for fast detection of nearest neighbors. The `ns=NeighborSearch()` object is initialized with a list of atoms to construct a KD tree, see Algorithm 2.1 (Line 15). Following this, the `ns.search(center_coords,radius)` method can be applied to the given atomic position `center_coords` and a `radius` in angstrom Å. In case, the entity level is given as atoms, this function returns all atoms within the specified `radius`. Prior to this, it is convenient to use `Selection.unfold_entities(structure,target_level)` with `target_level="A"` to select all atoms from a given structure, compare with Algorithm 2.1 (Lines 12 and 13). Thereby, entities are abbreviated as "S" for structure, "M" for model, "C" for chain, "R" for residue, and "A" for atom. The advantage of using `NeighborSearch()` with a radius of 5 Å is a reduction of the overall runtime.

The detection of neighbor atoms is then followed by computation of sterical clashes between DNMT1 and NCP147 for every possible position. We assume a sterical clash exists between two atoms if $d < r_1 + r_2$ with van der Waals (vdW) radii r_1 and r_2 and a distance d, see Algorithm 2.1 (Lines 22 and 23). The definition of a sterical clash is visualized in Figure 2.6. The vdW radius is dependent on the respective atom. We used the values suggested by Bondi [162], namely r_H=1.20Å, r_C=1.70Å, r_N=1.55Å, r_O=1.52Å, r_P=1.80Å, and r_S=1.80Å. The number of sterical clashes is then calculated for every nucleosome position and all necessary information on clashing atoms is subsequently stored for further analysis. Note that we excluded zinc ("ZN"), manganese ("MN"), and chlorine ("CL") elements as well as atoms of the DNMT1 DNA chain or atoms of wrapped DNA that are part of the superimposition from these sterical clash calculations. The overall sterical clash (in percent [%]) between the DNMT1 enzyme and a given DNA nucleosome position was then computed as the number of sterical clashes between DNMT1 atoms and nucleosome atoms divided by the total number of DNMT1 atoms. In summary, the procedure outlined in Algorithm 2.1 illustrates the convenient applicability of the Bio.PDB module concerning structural computations. The overall structural approach to decipher DNA methylation patterns is discussed and presented in the following sections. For a detailed description of the Bio.PDB module please refer to [89].

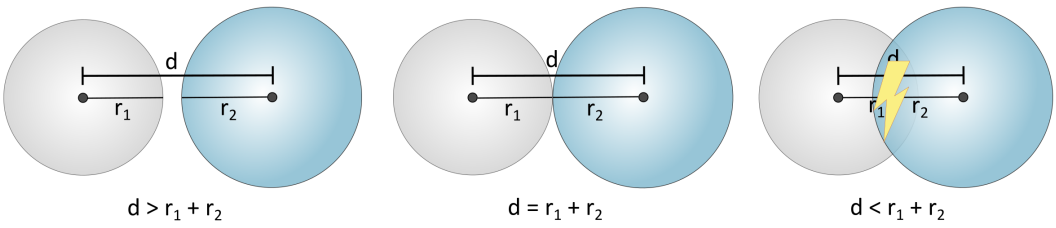

Figure 2.6: Illustration of a sterical clash. A sterical clash for two atoms with vdW radii r_1 and r_2 and distance d exists if $d < r_1 + r_2$ holds.

2.2 Aim of this work

As mentioned, it was suggested that nucleosome occupancy and DNA methylation are dependent on each other [158]. The same study reported a ten base periodicity of DNA methylation in nucleosomal DNA. DNMTs bind to the major groove of DNA that is located on the nucleosome outside, which would explain the observed ten nucleotide periodicity [158]. Moreover, it was reported that nucleosome–bound DNA is methylated to a higher extent compared to flanking DNA sequences [158]. Since these findings suggest that DNMTs are able to methylate nucleosome–bound DNA [158], we aimed at determining precise nucleosomal DNA positions (base pair resolution) that are accessible for DNMT1 such that a specific cytosine in CpG context can undergo methylation. By assuming that DNA methylation takes places on nucleosomes and that nucleosome–bound DNA can be methylated by DNMT1, we investigated which DNA positions are not reachable by the DNMT1 enzyme due to the organization of the nucleosome structure. For this, we implemented an *in silico* structural superimposition approach and subsequently compared computed accessibility scores to experimental NOMe–seq methylation data. We found that the experimental DNA methylation patterns in regions with high nucleosome density reflect the accessibility of DNMT1 to specific DNA positions when complexed as nucleosome structure. This strengthens the hypothesis that methylation can take place on nucleosomes as suggested [158], and that only specific DNA bases are accessible by DNMT1 which is due to the nucleosome core complex structure that in turn leads to the experimentally observed methylation patterns.

2.3 Materials and methods

This section starts with an explanation of our superimposition approach that generates a structural alignment of DNMT1 with every nucleosome–bound DNA position. This procedure allows to find physically feasible DNMT1–nucleosome compositions that can subsequently be used to derive structural accessibility scores. Following this, we compared these calculated scores with experimental methylation data using a sliding window approach.

2.3.1 Structural superimposition approach

Protein crystal structures of the nucleosome core particle NCP147 (PDB identifier: 1KX5 [10]; *Homo sapiens* and *Xenopus laevis*) and of the human DNMT1 complexed with a 19 bp DNA molecule (PDB identifier: 3PTA [137]; *Homo sapiens*; residues 646–1600) were retrieved from the Protein Data Bank (PDB) [21]. The crystal structures of NCP147 and DNMT1 are shown in Figure 1.2 and Figure 2.2, respectively. Using PyMOL (version 1.3) [22] together with the Biopython library (version 1.68) [87, 89], we performed a structural superimposition between DNMT1 and the nucleosome core complex for every nucleosome–bound DNA position. The structural approach is illustrated in Figure 2.7 and outlined in Algorithm 2.1.

This DNA–DNA structural alignment was constructed based on the atomic positions of the DNA sugar backbone (that means elements P, OP_1, OP_2, O_5, C_5, C_4, O_4, C_3, O_3, C_2, and C_1) of DNA positions 4 to 15. For every possible DNMT1–DNA–DNA-nucleosome structural alignment, we computed a superimposition with minimal RMSD by calculating the rotational and translational matrix M_{RT} that is required to place the sugar backbone DNMT1–DNA atoms onto the nucleosome–DNA atoms. M_{RT} is then applied to all atoms of the DNMT1 molecular complex for a spatial transformation onto the considered nucleosome–DNA position, see lower panel of Figure 2.7.

Next, we determined which DNMT1–nucleosome superimpositions are mechanistically feasible by computing a sterical clash score between DNMT1 and nucleosome for every binding position. To reduce computation time, we calculated a sterical clash between a DNMT1 atom and all nucleosome neighbor atoms in a distance of 5Å. We assume that a sterical clash exists between two atoms if $d < r_1 + r_2$ with the respective atom van der Waals (vdW) radii r_1

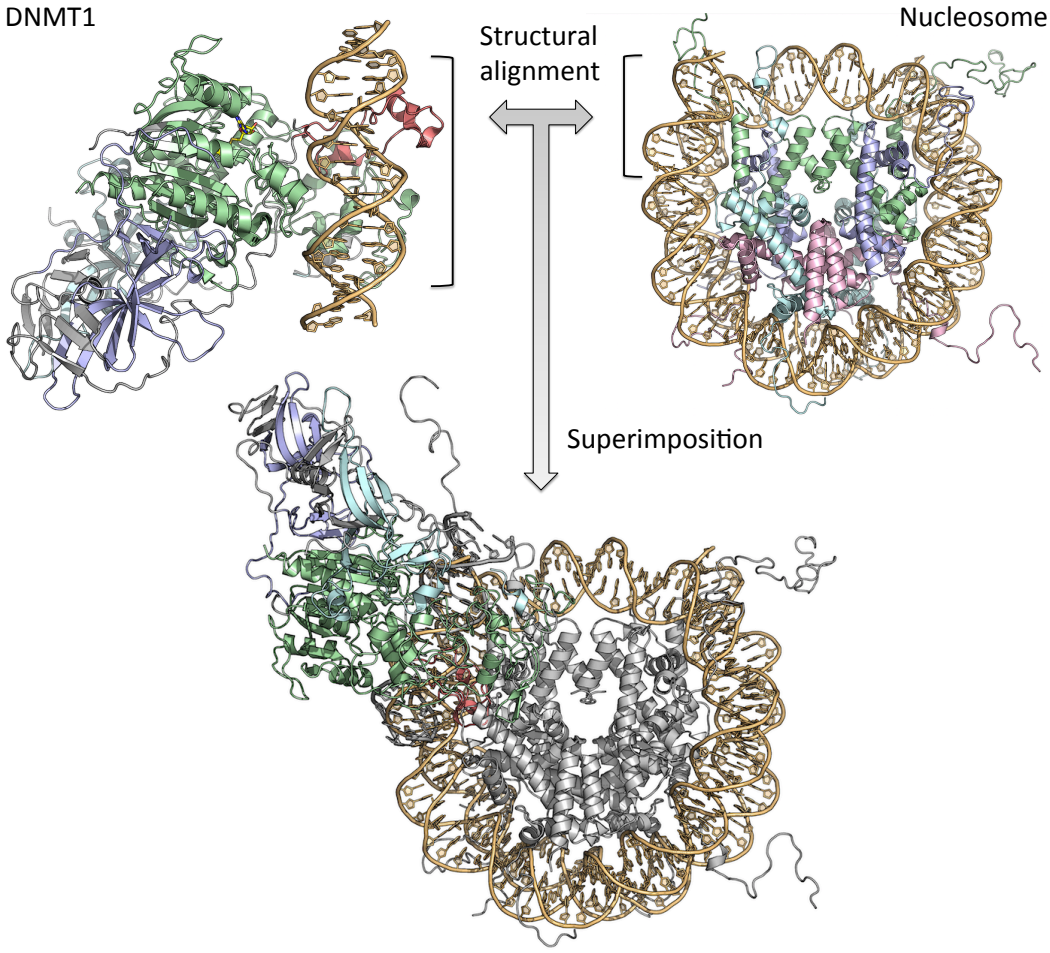

Figure 2.7: Superimposition of DNMT1 and nucleosome core complex. DNA positions 4–15 of DNMT1 are mapped onto every position of the nucleosomal DNA by calculating a rotation and translation matrix for transformations of the spatial coordinates. Shown is the mapping of DNMT1 to nucleosome DNA position 1. For this, a structural alignment between the DNA bases of DNMT1 (left panel) with the nucleosome–bound DNA (right panel) is generated. The bottom panel shows the superimposition of the DNA stretches. In this orientation, DNMT1 points away from the nucleosome.

and r_2 as well as the distance d, see Figure 2.6. To calculate the overall percentual sterical clash between the DNMT1 enzyme and the nucleosome at a DNA position, we calculated the number of sterical clashes between DNMT1 atoms and nucleosome atoms normalized by the total number of DNMT1 atoms.

2.3.2 Comparison with experimental methylation data

To validate our superimposition approach, we compared the *in silico* computed accessible DNA positions against experimental GCH and HCG methylation data. This data provides genome–wide information about methylated GpC and CpG sites that can be used to deduce nucleosome dense regions together with cytosine methylation rates in CpG context. For simplification, note that in the following we use the term "NOMe–seq" for methylation at GpC sites that was used to infer nucleosome positions, whereas CpG methylation is referred to as "WGBS". Figure 2.8 displays the overall approach. First, high nucleosome density regions (HNDRs) and low nucleosome density regions (LNDRs) were derived from experimental NOMe–seq data using a

two–state binomial HMM combined with Fisher's exact test (provided by Karl Nordström), compare with [53]. These regions together with position–specific CpG methylation (WGBS) were then assigned to promoter regions since promoters exhibit specific nucleosome arrangements to regulate transcription [100]. Since open and closed chromatin regions are broad regions rather than specific 146 bp sequences, we applied a sliding window approach and calculated matching–scores for every possible nucleosome position within HNDRs and LNDRs using experimental and randomized WGBS methylation data. In the following, a comparison between our *in silico* scores and experimental methylation data is explained in detail.

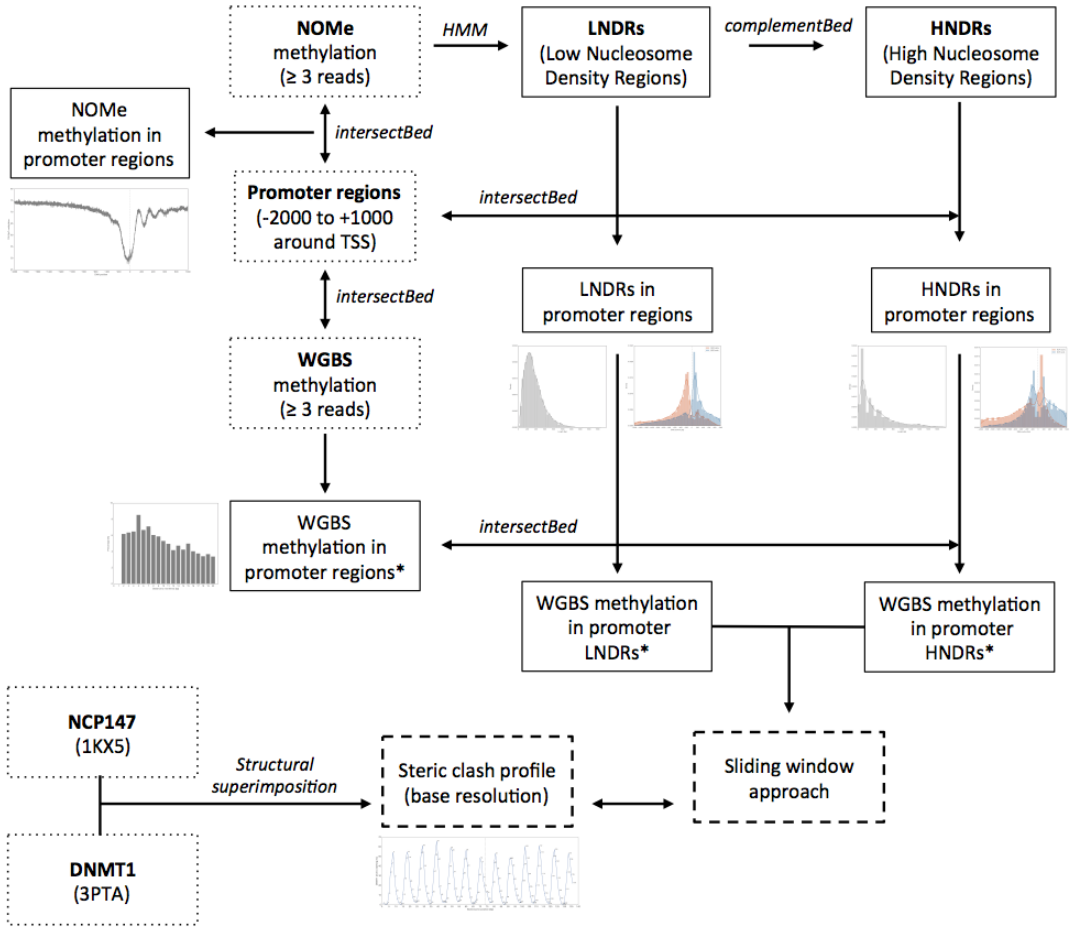

Figure 2.8: Comparison of *in silico* and experimental data. The flowchart depicts the overall approach and the comparison between *in silico* structural superimposition and experimentally observed methylation. Note that "NOMe" refers to methylation of GCH sites while "WGBS" refers to HCG methylation sites. An asterisk (*) denotes that randomized data was computed as well. All plots shown are explained and discussed in detail in Section 2.4.

GCH and HCG methylation rates in promoter regions

Annotated open regions with higher GCH methylation than the local surrounding and thus with lower nucleosome density (LNDRs) were determined based on GCH methylation data. As mentioned above, LNDRs were provided by Karl Nordström, compare with [53]. The here utilized NOMe–seq experimental data was first used in Schmidt et al. [53]. Since these regions exhibit higher GCH methylation density compared to the local surrounding, they have a higher probability of being open chromatin regions. We then considered the complement genomic positions of the annotated open regions as regions with higher nucleosome density

compared to the local surrounding (HNDRs), see Figure 2.8. These were extracted using the `complementBed` function from the BEDTools suite (version v2.26.0) [78]. The necessary hg19 chromosome lengths were taken from the UCSC genome browser [56].

Since promoter regions harbor specific nucleosome patterns to enable transcriptional regulation [100], we investigated these regions in detail with regard to DNA methylation and nucleosome occupancy. We defined the promoter region as the range from –2000 to +1000 bp around the TSS. Human Reference Sequence (RefSeq) gene annotations were downloaded from the UCSC genome browser hg19 assembly [56, 163]. We removed all microRNAs and small nucleolar RNAs, and genes with equal CDS start/end positions. Moreover, we included only genes located on chromosomes 1 to 22, X, and Y. To analyze nucleosome occupancy and DNA methylation in the promoter region, we used the `intersectBed` function of the BEDTools suite [78] (version v2.26.0) to assign the filtered WGBS (HCG) and NOMe–seq (GCH) data to the defined promoter regions, see also Figure 2.8. Thereby, we only included positions with a coverage of at least three reads in further analyses, compare with [100].

First, we determined the length [bp], start/end positions, and the corresponding GpC methylation rates of HNDRs and LNDRs to decipher a general nucleosome occupancy pattern around a TSS. For this, we calculated the average 100–GpC methylation rate for every position within the predefined promoter regions. Thereby, an increased 100–GpC rate corresponds to a decreased GpC methylation rate at this position and thus a region possibly occupied by nucleosomes (depending on the local surrounding). Next, we analyzed the general genome–wide CpG methylation patterns in the promoter regions by calculating the distance between individual methylated CpG sites. For this calculation, we considered CpG sites with methylation rates greater zero. To calculate the statistical significance, we then randomized WGBS data by permutating the methylation rates within the promoter regions (without replacement), see Figure 2.8. Annotated open and closed promoter regions together with the strand–specific experimental and randomized CpG methylation rates were then compared to the computed DNA accessibility scores, compare with Figure 2.8. This comparison is based on a sliding window approach that is explained below.

Sliding window approach in HNDRs and LNDRs

Processing of experimental methylation data only provides broad regions with higher GpC methylation rate compared to the local surrounding (LNDRs) and their complement (HNDRs). Thus, we applied a sliding window approach and computed matching–scores for every possible window location to quantify the match of *in silico* accessibility values and DNA methylation rates. The length of the sliding window amounts to 136 bp, which refers to the number of nucleosome positions with sterical clashing information, while the step size equals 1 bp. For this, computed accessibility values were normalized to a range between 0 and 100 to obtain the same scale as the methylation rate. The sliding window approach is outlined in Figure 2.9. For all promoter regions with HNDRs and LNDRs harboring methylated CpGs sites, we calculated a matching–score$_\text{w} \in [0, 1]$ for a possible sliding window position w:

$$matching - score_w = \frac{1}{\#CpGs_w} \times \sum_{i=1}^{\#CpGs_w} M_i,$$

whereby M_i was defined as

$$M_i = \begin{cases} 1, & \text{if } (mr_i > m_{thres} \wedge c_i \leq c_{thres}) \vee (mr_i \leq m_{thres} \wedge c_i > c_{thres}) \\ 0, & \text{otherwise} \end{cases}$$

with the methylation rate mr_i at CpG position i, a methylation threshold m_{thres} required to assume a CpG to be methylated, the accessibility/sterical clash c_i at position i calculated by the structural superimposition approach, and a tolerated sterical clash c_{thres}. Thus, $M_i = 1$ if a CpG is methylated and the sterical clash is tolerated or if a CpG is not methylated and the

sterical clash is not tolerated, which would mean DNMT1 is not able to bind to the nucleosome. The sum is then normalized by the number of CpG sites $\#CpGs_w$ in the considered sliding window. Elevated matching–scores indicate that the methylation rate reflects the calculated sterical clash. We compared matching–scores calculated using different parameter values for $c_{thres} \in \{5, 10, 20, 50\}$ and $m_{thres} \in \{0, 10, 20\}$. For simplicity, matching–scores are abbreviated as combination of c_{thres} and m_{thres}. For instance, *c5m0* denotes that the score was calculated using $c_{thres} = 5$ and $m_{thres} = 0$, whereby sterical clashes are tolerated if $c_i \leq 5\%$ and a CpG is assumed to be methylated if the methylation rate $mr_i > 0$.

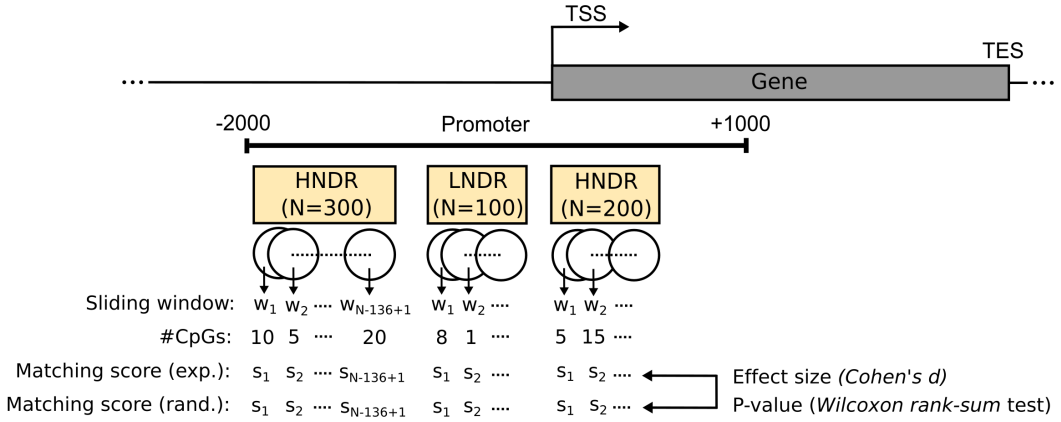

Figure 2.9: Sliding window approach. Shown is a promoter region, which was defined to range from –2000 to +1000 bp relative to the TSS. Matching–scores were calculated for every possible sliding position within HNDRs and LNDRs. To evaluate our results, scores between experimental and randomized data were then compared regarding Cohen's d values to measure effect size and p–values calculated with the Wilcoxon rank–sum test.

Next, we compared matching–scores of HNDRs and LNDRs with each other. To test for significance, we applied the Wilcoxon rank–sum test to compare the matching–scores of experimental and randomized methylation rates. With a sufficiently large sample size, p–values tend to become highly significant, even when the magnitude of the difference between the two groups is very small [50, 51]. Thus, to also interpret the magnitude of underlying differences, we computed the effect size by applying Cohen's d estimator that is independent of the sample size [50, 52]. Cohen's d is defined as

$$d = \frac{(m_e - m_r)}{\sqrt{\frac{s_e^2 + s_r^2}{2}}}$$

with experimental mean m_e, randomized mean m_r, experimental standard deviation s_e, and randomized standard deviation s_r. Effect sizes were separated into small ($0.2 \leq d < 0.5$), medium ($0.5 \leq d < 0.8$), and large ($d \geq 0.8$) according to [50].

2.4 Results and discussion

To explain the experimentally observed methylation patterns, we applied a structural superimposition approach considering DNMT1 and the nucleosome core complex. In the following, we present the DNA accessibility scores that were obtained by calculating the sterical clash between these two X–ray structures. Our *in silico* results are subsequently compared to experimental methylation data to evaluate our hypothesis that methylation of CpG sites is dependent on their accessibility by DNMT1, which is impaired through the nucleosome core complex. The results of our approach are explained in the following.

2.4.1 Superimposition detects accessible CpG sites

With the help of an *in silico* structural superimposition approach based on X–ray crystallographic structures of DNMT1 and the nucleosome core complex, we superimposed the complex of DNMT1 bound to a stretch of double–stranded DNA with every position of DNA wrapped around a histone octamer. As an example, Figure 2.10A and 2.10B show two superimpositions for DNA–nucleosome positions 2 and 18. For every aligned nucleosome–DNA position, we computed whether DNMT1 can bind reasonably well to the nucleosomal DNA such that an individual DNA position can be methylated. Figure 2.10C shows the sterical clash [%] as inverse measure of DNA accessibility for every histone octamer–bound DNA position. As an example, DNMT1 can bind reasonably well to nucleosome–bound DNA position 2 and hence methylate this CpG site, see Figure 2.10A. When DNMT1 is placed at position 18, nearly 60% of DNMT1 atoms overlap with the nucleosome, see Figure 2.10B. This arrangement is physically infeasible meaning that DNMT1 is not be able to methylate a CpG at DNA position 18 as long as the DNA is tightly wound around the histone octamer core. Calculation of the matching–score based on the sterical clash for every nucleosome–bound DNA position resembles a wave pattern that results from the wrapping pattern of DNA around the nucleosome.

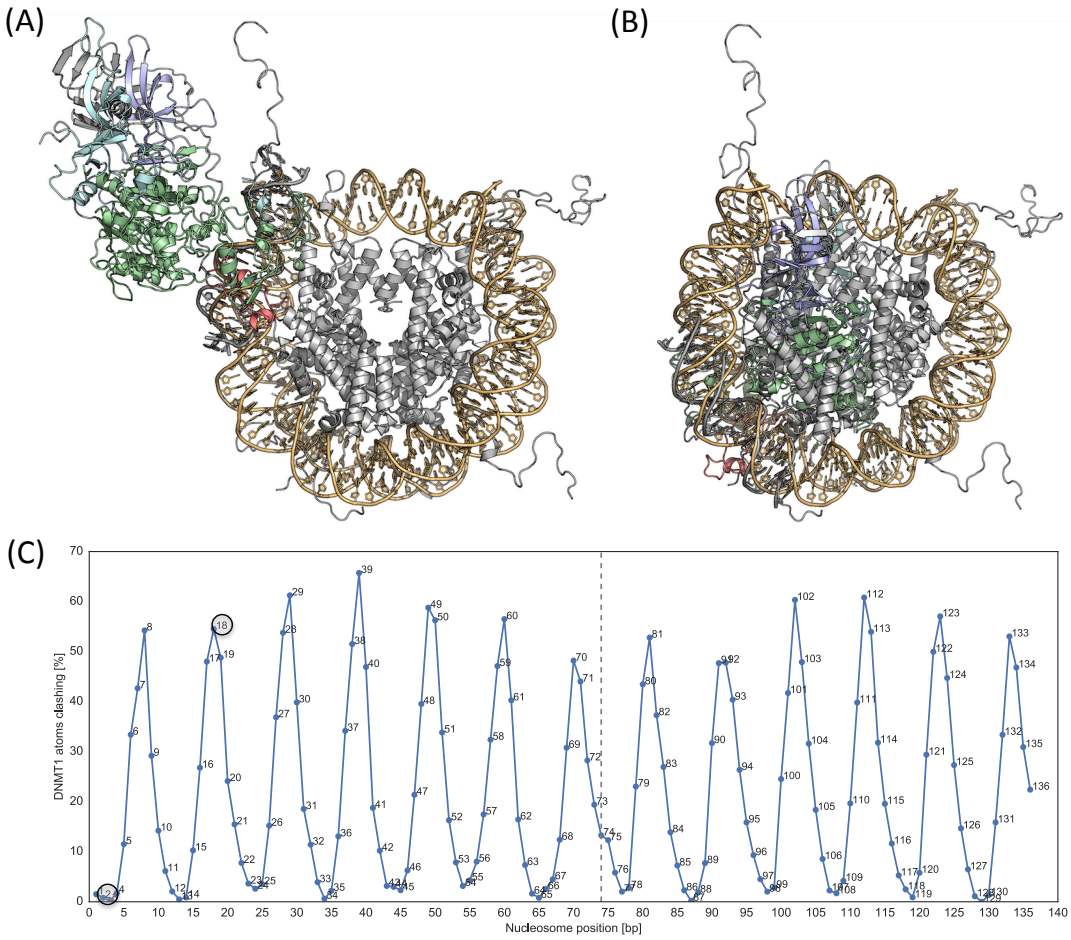

Figure 2.10: Sterical clash between DNMT1 and the nucleosome core complex. Superimposition of DNMT1 and the nucleosome core particle reveals a wavelike pattern of sterical clashes. Shown is the superimposition of DNMT1 and the nucleosome at (A) DNA position 2 and (B) DNA position 18. (C) The obtained wave pattern of sterical clashes results from superimpositions of DNMT1 and the nucleosome core complex for every DNA–nucleosome position. The scores for the two orientations 2 and 18 are marked by circles.

2.4.2 Experimental data evaluates structural approach

First, experimental GCH and HCG methylation data were filtered. In a preprocessing step, we removed all sites that were covered by less than three reads. This resulted in a reduction of about 6.0% for GpC (NOMe) positions from about 219,000,000 sites reduced to about 206,000,000 sites. Considering CpG (WGBS) sites, this step reduced our dataset from about 39,000,000 sites to 36,000,000 sites, which amounts to about 8.7%. In total, we incorporated 25,416 gene promoters that were located on chromosomes 1–22, X, or Y, into our analyses. Following this, we analyzed the distribution of nucleosomes and methylated CpG sites in gene promoters. Finally, the results of the sliding window approach were evaluated. To test for statistical significance, we incorporated randomized methylation data into our approach.

Unambiguous nucleosome phasing at the TSS

Next, we aimed at determining the nucleosome occupancy in gene promoters. For this, we calculated the average fraction of unmethylated GpCs for every DNA promoter position based on experimental GCH methylation data, see Figure 2.11A. Note that an elevated 100–GpC methylation rate corresponds to nucleosome protected DNA sequences. Thus, the flanking sequence around the TSS can be separated into LNDR and HNDRs. Figure 2.11A reveals a clearly visible nucleosome phasing pattern directly downstream of the TSS. Since the TSS is close to the downstream end of the LNDR, the TSS position is a reasonable reference point for the start of the next nucleosome. Due to the differing lengths of LNDRs between genes, the regions upstream of the TSS become on average more "disordered" and the nucleosome phasing is not clearly visible anymore. This pattern was observed before [100, 146] and reflects the nucleosome organization that is necessary to regulate gene transcription.

We then investigated the length distributions together with start/end positions of LNDRs and HNDRs. The results are shown in Figure 2.11B–E. In human gene promoters, we found that the average length of HNDRs is 424 ± 429 bp, while the length of LNDRs is 355 ± 216 bp on average. Since we considered promoter regions defined from –2000 to +1000 bp around the TSS, a HNDR of 3,000 bp is the maximum length possible. Figure 2.11C and E show that the majority of HNDRs end about –200 bp upstream of the TSS and start about 100 bp downstream of the TSS. LNDRs are by definition located at complementary positions. This distribution resembles the observation in Figure 2.11A and [100] and draws confidence in the annotation of HNDRs and LNDRs from experimental NOMe–seq data.

Distances between methylated CpGs diverge

Following the detailed analysis of accessible nucleosome–bound DNA positions and nucleosome occupancy in promoter regions, we next examined the general CpG methylation pattern in promoter regions. Therefore, we plotted the average CpG methylation level (see Figure 2.12A, black lines) as well as the distance distribution between CpGs with a methylation rate > 0 (see Figure 2.12B). Methylated CpG positions were detected by assigning the WGBS filtered data to the defined promoter regions, see Figure 2.8. In Figure 2.8, we compared both experimental and randomized CGH methylation rates. We observed that experimental CpG methylation levels decreased towards the TSS, had a small peak at the TSS, and increased again downstream of the TSS, see Figure 2.12A (black lines).

Moreover, we found that distances between experimentally detected methylated CpGs are uniformly distributed with a small peak at a distance of 5 bp, see Figure 2.12B. Thereby, the distance refers to the number of base pairs between two methylated cytosines. Thus, the smallest distance possible is 2 bp, for instance C*GTC*G, with * denoting cytosine methylation and a GT within two methylated cytosines. A distance of zero is not possible since methylation takes place in CpG context. Due to experimental restrictions, a composition of C*GC*G (distance of 1 bp) is not possible since a methylated CpG can only be reported in HCG context with H=A,

C, or T. As mentioned above, we randomized the WGBS data in order to evaluate the statistical significance of our hypothesis that DNA methylation is restricted by nucleosome structure. Figures 2.12A and C show the average CpG methylation and distances for the randomized data in promoter regions. In contrast to the experimental data, randomized methylation rates are equally distributed within the promoter region (Figure 2.12A, grey lines), whereas the distances only slightly change as depicted in Figure 2.12C (compare with Figure 2.12B).

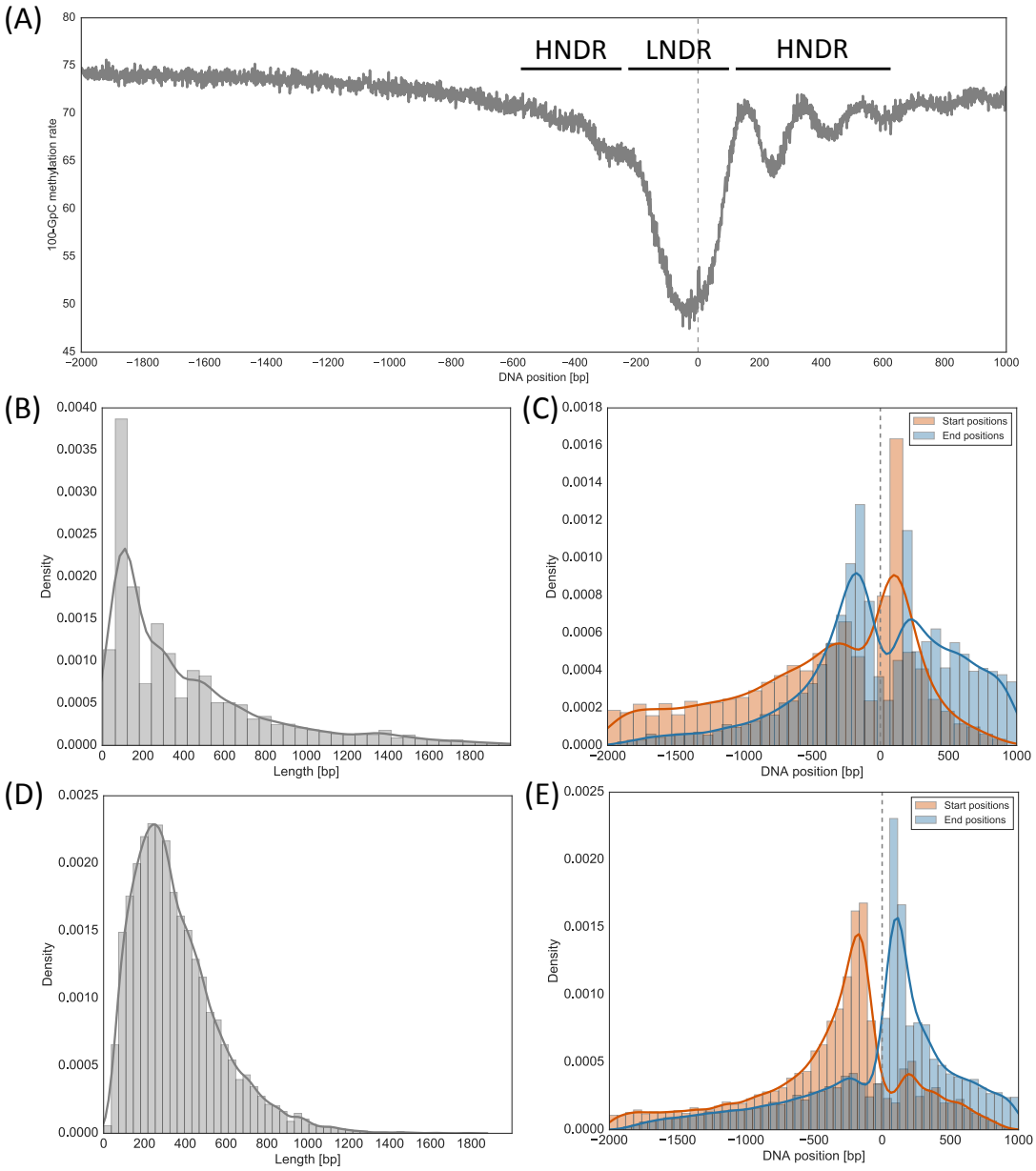

Figure 2.11: NOMe–seq GpC patterns in promoter regions. The TSS is located at position zero. (A) Average 100–GpC methylation ratios (fraction of unmethylated GpCs) indicate nucleosome depleted and occupied regions. (B–E) Regions with higher nucleosome density compared to the local surrounding (HNDRs) and regions with lower nucleosome density (LNDRs) were derived based on experimental GCH NOMe–seq data. Shown are distributions of region lengths and start/end positions of HNDRs (B,C) and LNDRs (D,E).

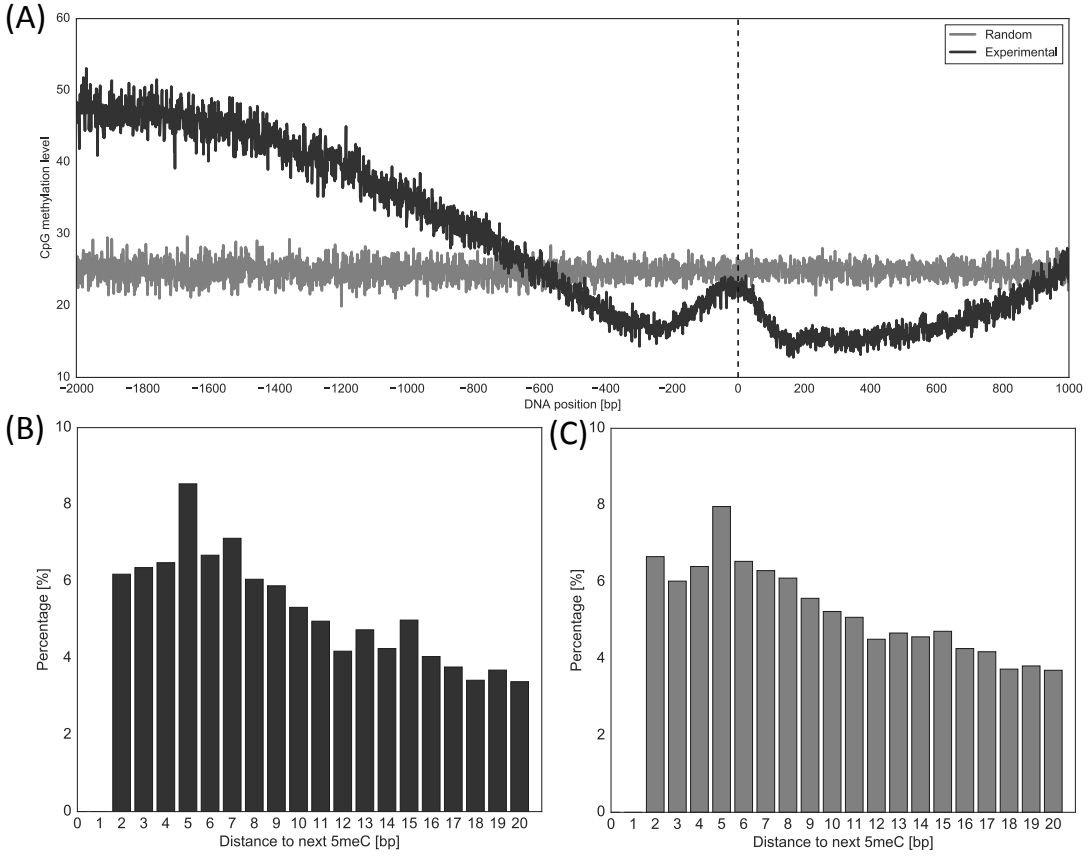

Figure 2.12: CpG methylation pattern of experimental and randomized data. We analyzed the general CpG pattern for experimental (black) and randomized (grey) methylation rates in promoter regions. The TSS is located at position zero. (A) Average experimental and randomized CpG methylation rates (fraction of methylated CpGs). (B,C) Distances between methylated CpGs for experimental (B) and randomized methylation rates (C). We assumed a methylation rate of $mr_i > 0$ and only display distances ≤ 20 bp for convenience.

Methylation patterns resemble DNA accessibility

As we observed differing CpG methylation periodicities, compare with Figure 2.12B, rather than individual methylations peak every 10 bp as detected by [158], we analyzed if there is a dependency between accessible nucleosome–bound and methylated CpG positions. As described in Section 2.3.2, we calculated a matching–score for every possible sliding window within promoter HNDRs and LNDRs to compare experimental and randomized CpG methylation levels with the estimated accessibility of DNMT1 to the nucleosome. The matching–scores for experimental and randomized methylation data were analyzed in terms of p–values (Wilcoxon rank–sum test) and effect size (Cohen's d). Since the number of CpGs within a sliding window strongly influences the reliability of the matching–score, we analyzed effect size and statistical significance dependent on the number of CpG sites.

Figure 2.13A and B as well as Figure 2.14A and B display the effect size and p–value dependent on the number of CpGs within a sliding window, by comparing experimental and randomized data within HNDRs and LNDRs. For HNDRs, we found clear differences of Cohen's d values depending on the values of the two parameters $m_{thres} \in [0, 10, 20]$ and $c_{thres} \in [5, 10, 20, 50]$. In general, $c_{thres} = 5$ with any m_{thres} value (c5m*) seems to give the most significant results, see Figure 2.13A. Since we assume that a CpG position can be methylated if the sterical clash between DNMT1 and the nucleosome is below 5%, the results obtained with this parameter selection support our hypothesis.

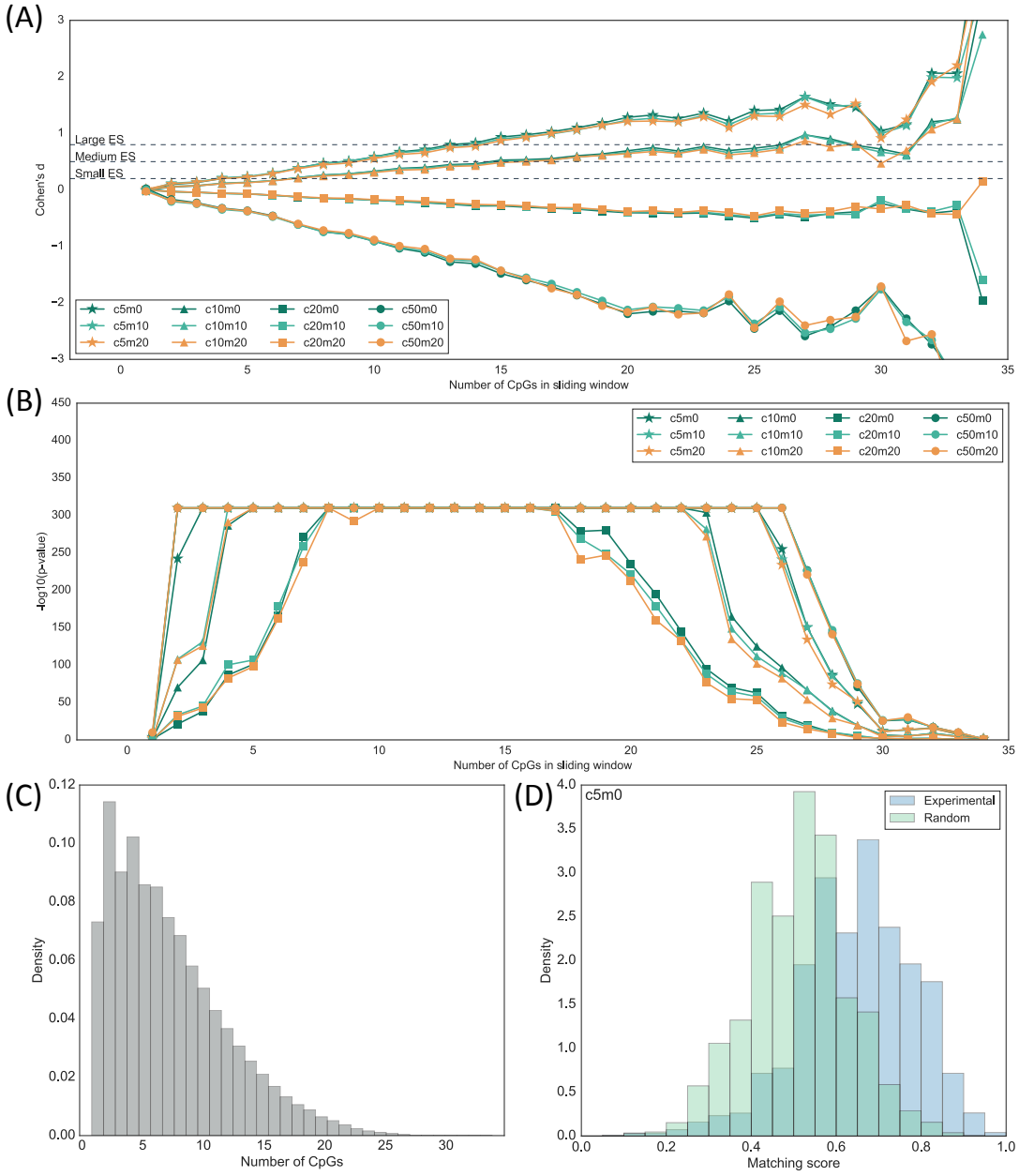

Figure 2.13: Results of sliding window approach in HNDRs. We applied a sliding window to HNDRs and compared experimental and randomized methylation data. (A) Cohen's d values, for convenience plotted from –3 to 3, for the different matching–scores. The respective numbers are shown in Table A.1. (B) p–values between experimental and randomized data were calculated with the Wilcoxon rank–sum test. For numerical reasons, $-log_{10}(p - value) = 310$ is the maximum, and thus the smallest p–value possible using Python version 2.7 with scipy package version 0.19. (C) Number of CpGs within different sliding windows of 136 bp length. The respective frequencies and percentages can be found in Table A.3. (D) Distribution of matching–scores for experimental and randomized methylation rates. Matching–scores were calculated with parameters $c_{thres} = 5$ and $m_{thres} = 0$ for all sliding windows with at least 15 CpGs.

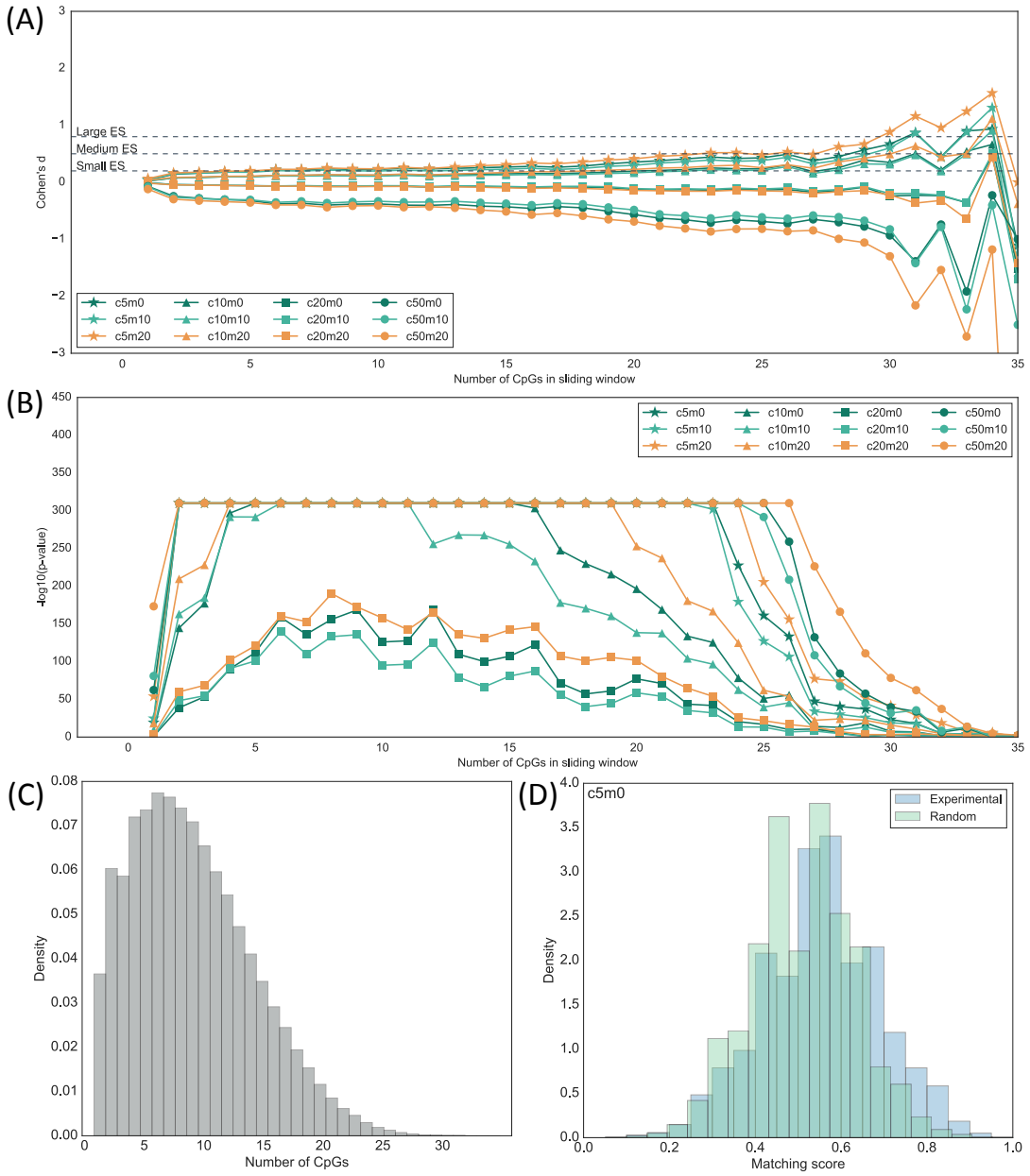

Figure 2.14: Results of sliding window approach in LNDRs. We applied a sliding window to LNDRs and compared experimental and randomized methylation data. (A) Cohen's d values (for convenience plotted from –3 to 3) for the different matching–scores. The respective numbers are shown in Table A.2. (B) p–values calculated with Wilcoxon rank–sum test between experimental and randomized data. For numerical reasons, $-log_{10}(p - value) = 310$ is the maximum, and thus the smallest p–value, possible using Python version 2.7 with scipy package version 0.19. (C) Distribution of number of CpGs within sliding windows of 136 bp length, compare with Figure 2.10. The respective frequencies and percentages are listed in Table A.4. (D) Distribution of matching–scores for experimental and randomized methylation rates. A matching–score was calculated with parameters $c_{thres} = 5$ and $m_{thres} = 0$ for all sliding windows with at least 15 CpGs.

For higher threshold values c_{thres}, the effect size decreases and for high clash values becomes even negative, meaning that the matching–scores of randomized methylation data were on average higher than for experimental methylation rates. Moreover, the parameter m_{thres} seems to have a smaller influence on the effect size than c_{thres}. This behavior is in accordance

with our hypothesis that DNA can be methylated when wrapped around the histone octamer core and only positions accessible by DNMT1 are methylated. As expected [50, 51] and mentioned beforehand, the p–values calculated with the Wilcoxon rank–sum test are highly significant due to the large sample size, see Figure 2.13B. This shows that there is a significant difference between experimental and randomized data, whereas the effect size quantifies the magnitude of the difference and is thus crucial for interpretation here.

Figure 2.13C shows the overall distribution of the number of CpGs within HNDR sliding windows. In total, HNDRs exhibit 2,277,851 possible sliding windows with on average 7.12 ± 4.88 CpGs and a median value of 6.0 CpGs per window, see Figure 2.13C and Table A.3. Figure 2.13D shows the distribution of experimental and randomized matching–scores when requiring at least 15 CpGs within a sliding window and using parameters $c_{thres} = 5$ and $m_{thres} = 0$ (c5m0). This number of at least 15 required CpG sites was selected as Cohen's d value then exceeds $d \geq 0.8$, which corresponds to a large effect size, see Figure 2.13A. Table A.3 shows that about 9% of the sliding windows within LNDRs contain at least 15 CpGs. All these findings draw confidence in our initial hypothesis that methylation patterns are dependent on the DNA accessibility of DNMT1.

As shown in Figure 2.14, LNDRs yielded lower Cohen's d values compared to HNDRs. All parameter curves behaved similarly and increased only slightly with an elevated number of CpGs, see Figure 2.14A. Figure 2.14B shows that, as for the HNDRs, most of the p–values are highly significant, which is also due to the large sample size. In total, LNDRs exhibit 4,390,558 possible sliding windows with on average 9.0 ± 5.22 CpGs and a median of 8.0 CpGs per window, see Figure 2.14C and Table A.4. Figure 2.14D displays the distribution of experimental and randomized matching–scores (c5m0) with at least 15 CpGs (as displayed for HNDRs). The density distributions of both groups overlap largely which underlines that the strong signal is only present at HNDRs.

2.5 Summary

In this project, we analyzed the three–dimensional structures of the DNMT1 enzyme and the nucleosome core complex in relation to observed DNA methylation patterns in the human genome. As data basis, we used the determined X–ray structures of DNMT1 bound to a 19 bp DNA molecule (PDB identifier: 3PTA [137]) and of the NCP147 nucleosome complex (PDB identifier: 1KX5 [10]). Both structures were retrieved from the Protein Data Bank (PDB) [21]. Methylation data for GCH and HCG methylation rates were detected by conducting NOMe–seq and WGBS experiments and were provided by our collaborators from the Saarland University (Epi)genetics department. With the help of a structural superimposition approach between DNMT1 and every base pair position of the NCP147 nucleosome core complex, we determined accessible nucleosome–bound DNA positions at nucleotide resolution. These accessibility scores were based on a computation of sterical clashes between DNMT1 and the NCP147 core complex. Next, we aimed at a statistical comparison between experimentally detected methylation rates and our computed accessibility scores. For this, we applied statistical tests and calculated Cohen's d effect sizes between experimentally observed and randomized methylation rates. By doing so, we could show that the observed DNA methylation patterns in regions with higher nucleosome density than the local surrounding can be explained by accessibility of DNMT1 to the nucleosome–bound DNA. Thereby, we compared different parameter thresholds for required methylation rates and tolerated sterical clashes. We found that large Cohens'd effect sizes between experimental and randomized methylation data were only present when the tolerated sterical clash was below 5%. This is in accordance with the initial hypothesis that nucleosome–bound DNA can methylated by DNMT1 and that DNMT1 is only able to methylated accessible CpG sites. For this study, we constraint our analyses to promoter regions as these regions show a specific nucleosome phasing around the TSS that was also observed in our initial analysis of the experimental data.

CHAPTER 3

Prediction of non–canonical 5′ UTR translational initiation sites

This chapter deals with the challenging prediction of eukaryotic alternative translation start sites in the 5′ UTR of a given mRNA sequence. Sections 3.2 to 3.6 of this chapter were adapted and expanded from our published manuscript "PreTIS: A Tool to Predict Non–canonical 5′ UTR Translational Initiation Sites in Human and Mouse. Kerstin Reuter, Alexander Biehl, Laurena Koch, and Volkhard Helms. *PLoS Computational Biology*, 12(10):e1005170, 2016". Alexander Biehl implemented a first functioning version of the web service, which was revised, improved, and partially reimplemented by me. Laurena Koch analyzed mRNA secondary structure and GC–content with a focus on their usability as features. Both, Alexander Biehl, and Laurena Koch contributed to this project in the course of their Bachelor's thesis and were advised by me. The *PreTIS* web service is accessible at `http://service.bioinformatik.uni-saarland.de/pretis`. Published supplementary information was omitted here. Please refer to our publication to examine the supplementary material. This chapter also complements the published version in certain regards. The biological background, the theory of machine learning, implementation details of the prediction model, and the web service application that were not explained in the publication are described in detail in Section 3.1, 3.3, and 3.6.

3.1 Prerequisites

This section describes the biological foundations of eukaryotic alternative translation initiation and of the experimental ribosome profiling technique that enables detection of translation start sites on a genome–wide scale. Moreover, machine learning is introduced with a focus on linear regression and support vector machines that were applied in this project. Advantages of web service development together with the fundamental languages of web programming are presented, followed by a description of the integrated data sources and bioinformatics tools.

3.1.1 'Death of a dogma': alternative translation initiation

Besides alternative transcription, splicing, and polyadenylation, there is also evidence for alternative translation initiation, see for instance Dever [25], Peabody [40], Kozak [41], Ivanov et al. [42], Lee et al. [43], Ingolia et al. [44]. The phrase "death of a dogma" was introduced by Mouilleron et al. [164]. It was reported that at least half of the mRNA transcripts exhibit two or more start sites [43, 44]. Experimental work showed that alternative start codons, additional to the canonical AUG–methionine, are recognized as start site during eukaryotic translation initiation [40, 41, 43, 44, 54]. It is assumed that a non–AUG translation initiation site (TIS) differs from AUG by one nucleotide, thus yielding CUG, UUG, GUG, ACG, AUA, AUC, AUU, AGG, and AAG [42]. Starck et al. [165] reported that an elongator tRNA carrying leucine was found to initiate translation at CUG codons. Moreover, several studies verified the existence of alternative TIS using mass spectrometry [166, 167] and ribosome profiling [43, 44, 54].

Interestingly, several studies report that the first amino acid incorporated into the polypeptide chain was methionine although the start site differed from AUG [40, 166, 168]. This suggests a general base mismatch between codon and anticodon during translation initiation [40]. Menschaert et al. [167] proposed a recoding event of leucine, valine, and threonine N–terminal amino acids back to methionine as these N–terminal start sites were incorporated by the usage of the non–AUG alternative start sites CUG, GUG, ACG, and UUG. Moreover, a hierarchy exists and some codons are used more frequently as start sites than other non–cognate alternative triplets [43, 44]. For instance, CUG and GUG codons were found to be the most frequent non–cognate TIS, whereas AGG and AUA are used less frequently [44].

Alternative ORFs and their biological impact

There are several possibilities how a single transcript can undergo alternative translation initiation and thus encode alternative ORFs. The formation of these alternative ORFs is illustrated in Figure 3.1. Note that in the following, the main reading frame is defined as the ORF that is initiated at the canonical AUG start site.

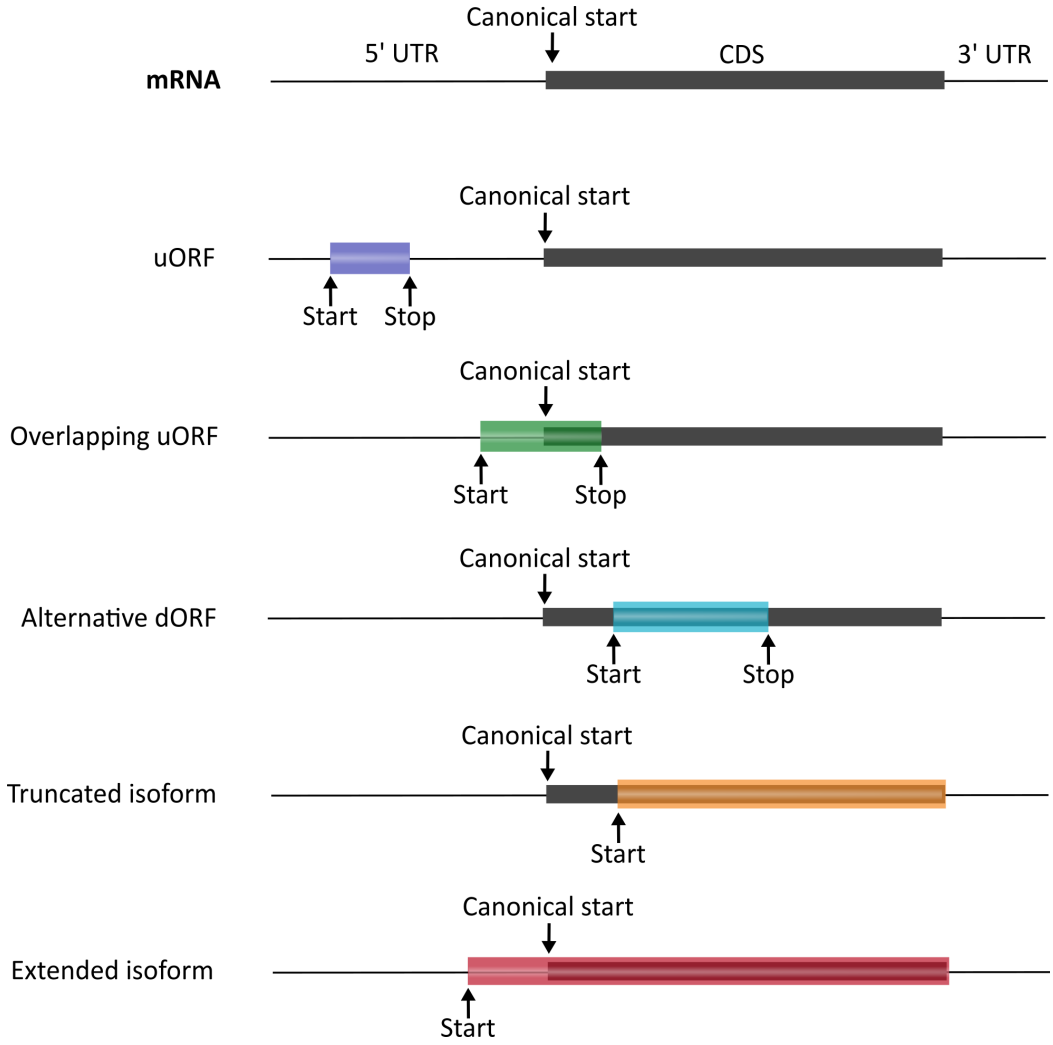

Figure 3.1: Alternative translation initiation. Dependent on the position of the start codon together with in–frame or out–of–frame translation, a single transcript can encode different protein isoforms. The figure was adapted from [169].

Short sequences encoded within 5′ UTRs are known as upstream ORFs (uORFs) [170]. These uORFs are involved in translational regulation of the main reading frame and thus have an influence on gene expression [171]. Furthermore, uORFs can overlap with the CDS, which is referred to as overlapping uORF, see Figure 3.1. In case a start site is located in the CDS and out–of–frame with the canonical start site, an alternative downstream ORFs (dORFs) is encoded. Additionally, dORFs that are in–frame with the main reading frame lead to truncated protein isoforms [169]. On the other hand, in–frame ORFs with a start site located within the 5′ leader sequence extend the CDS and thus result in protein isoforms with extended N–terminus [24]. An example is the TRPV6 Ca^{2+} channel protein, which was experimentally shown to exhibit an elongated N–terminus using mass spectrometry [166]. Thereby, the majority of alternative TISs is located in the 5′ UTR, whereas downstream TISs are scarcer and generally encompass an AUG codon [24, 43, 44]. It is also possible that a single transcript bears several alternative ORFs that can overlap, which is known as dual coding [172, 173]. Dual coding seems to be rare (about 1%), although this underestimation could be based on technical and experimental drawbacks [173].

Like alternative transcription or splicing, alternative translation further expands biological variety. Alternative translation influences protein abundance, the amino acid assembly defining a protein isoform, and thus overall protein diversity [24]. Consequently, regulation of such processes is of great importance for cell fate and requires a tight regulation [174]. Ivanov et al. [42] showed that the N–terminal extensions of several alternatively translated proteins were evolutionary conserved emphasizing their biological importance. Moreover, the function of alternatively truncated or elongated protein isoforms can differ from the function of the canonical proteins [175]. Starck et al. [93] reported that some uORFs initiated at non–AUG codons are associated with stress response suggesting that alternative ORFs are encoded at different cellular conditions.

Alternatively encoded ORFs can also influence regulatory processes. For example, there are different protein isoforms of c–Myc that are encoded by the proto–oncogene *c–Myc* [176]. Both proteins, c–Myc 1 and c–Myc 2, differ in composition and function. Thereby, c–Myc 1 is encoded at an upstream non–AUG start site, whereas c–Myc 2 refers to the canonical protein isoform that is translated at the canonical AUG codon. In general, the Myc protein was found to activate the *p53* tumor suppressor gene by binding an E box myc site within the *p53* promoter region [177, 178]. It was reported that cellular growth is only repressed in case *c–Myc 1* is overexpressed. Hann et al. [176] suggested that the elongated N–terminus leads to a conformational change of the c–Myc 1 trans–activation domain resulting in altered transcriptional regulation.

Furthermore, alternative initiation can change the N–terminal localization sequence that functions as sub–cellular target signal [179]. For instance, the sub–cellular localization of AtLIG1 (DNA ligase 1) is dependent on the start codon usage: the protein isoform translated from the first in–frame AUG is targeted to the mitochondria, whereas the protein encoded from the second in–frame AUG is transported to the nucleus [180]. Moreover, alternative start sites can also lead to both, a different function and a deviating cellular compartment [174, 181]. An example are four isoforms of *human fibroblast growth factor 2* that arise from alternative translation initiation [181].

mRNA sequence determines start site recognition

Marilyn Kozak conducted various experiments concerning the influence of the start site flanking sequence context [182, 183, 184, 185] and mRNA secondary structure [186, 187, 188] on translational initiation efficiency. It was reported that the start site flanking sequence context is crucial for translation initiation by eukaryotic ribosomes [185]. Especially positions –3R (R = purine) and +4G had proven to be essential for an efficient translation initiation. Position –3 was reported to be highly conserved in multiple vertebrate sequences, whereby an A is preferred over a G at this site [185]. An optimal eukaryotic initiation site was defined as the consensus sequence (GCC)GCC$\frac{\text{A}}{\text{G}}$CCAUGG [185].

Noderer et al. [189] applied high–throughput sequencing combined with fluorescence signaling (FACS–seq) to experimentally compare all possible flanking sequence contexts with a length of 11 nucleotides (position –6 to +5) concerning their ability to initiate translation at an AUG start site. Based on a dinucleotide position weight matrix (PWM), they derived efficiency values for all 4^8 = 65,536 sequences ranging from low, with a value of 12, up to most efficient, with a value of 150. For example, the sequence context GCCACCAUGGG described as optimal by Kozak [185] was assigned a value of 83. All these findings emphasize and illustrate that the flanking sequence context is essential for efficient translation initiation. The strength of the Kozak context also plays a role in start site selection as initiation at a specific start site seems to be dependent on the constitution of the upstream start codons. It was reported that an uORF, which is encoded from an AUG with strong Kozak context, entails that the main ORF is not translated [43].

Furthermore, the propensity and position of mRNA secondary structure residing directly downstream of a putative start site was shown to play an important role during translation initiation [186, 187, 188, 190]. Based on the free energy of this secondary structure, it is assumed that ribosome scanning decelerates and then halts with the AUG–recognition center directly positioned over the AUG start site, ready for translation initiation [188]. It was experimentally shown that mRNA secondary structure starting about 12 to 15 nucleotides downstream of a start codon and bearing a minimum free energy of $\Delta G = -19 \frac{kcal}{mol}$ can prevent leaky scanning and compensate for an unfavorable flanking sequence context [188, 190]. The largest effect was observed when the distance between the start site and the downstream hairpin structure amounted to 14 nucleotides [188]. The distance of 14 nucleotides enables that the ribosomal AUG–recognition center situates directly above the AUG start codon [188]. Translation initiation is hindered in case the hairpin structure is very stable with energy below $\Delta G = -50 \frac{kcal}{mol}$ [186, 187]. Free energies of about $\Delta G = -30 \frac{kcal}{mol}$ are only tolerated if the stem–loop structure is kept at distance from the start site [187]. It was proposed that a downstream mRNA secondary structure might be beneficial for the initiation at alternative non–AUG codons residing in upstream regions with increased CG–content [188].

Ribosome profiling detects translation start sites

The experimental technique that helped to decipher translational complexity on a genome–wide scale is called ribosome profiling and was developed by Ingolia et al. [170]. They first applied ribosome profiling to *Saccharomyces cerevisiae* budding yeast [170], followed by mouse embryonic stem cells [44]. Ribosome profiling data provides information on the density of ribosomes located at different regions of the transcript upon application of small chemicals that block the elongation process [44, 170]. The central idea of ribosome profiling is that regions which are protected by ribosomes are not digested upon application of nucleases [191]. Functional ORFs are then detected by deep–sequencing of ribosome–occupied mRNA fragments. These mRNA fragments have a length of about 30 nucleotides, which corresponds to the length of an RNA stretch protected by a ribosome. These fragments are called ribosome footprints [175]. Thus, this technique enables to precisely monitor translation *in vivo* at nucleotide resolution.

The ribosome profiling protocol involves three crucial steps: immobilization of active ribosomes, nuclease treatment together with digestion of mRNA fragments not occupied by ribosomes, and deep–sequencing of these fragments. The experimental approach is illustrated in Figure 3.2. This procedure enables to gain information on ribosome occupancy at different regions on the mRNA transcript. Thereby, treatment with the small chemical harringtonine, which specifically binds to and halts initiating ribosomes, facilitates to identify translation initiation sites [44]. More precisely, harringtonine binds to the 60S ribosomal subunit that is not complexed within the 80S ribosome and hinders ribosomal movement along the transcript [44, 192]. Thus, initiating ribosomes gather at and protect translation start sites from nuclease treatment [191]. These ribosome footprints are subsequently extracted, deep–sequenced and mapped to a reference genome [191]. High–throughput deep–sequencing enables to determine the nucleotide sequences of several billion short reads at the same time [193]. Based

on their short length, mapping of these reads into a reference genome is challenging [169]. Next, the integration of a machine learning approach based on support vector machines [194], resulted in the determination of translation initiation sites from ribosome footprints profiles with an accuracy of 86% [44].

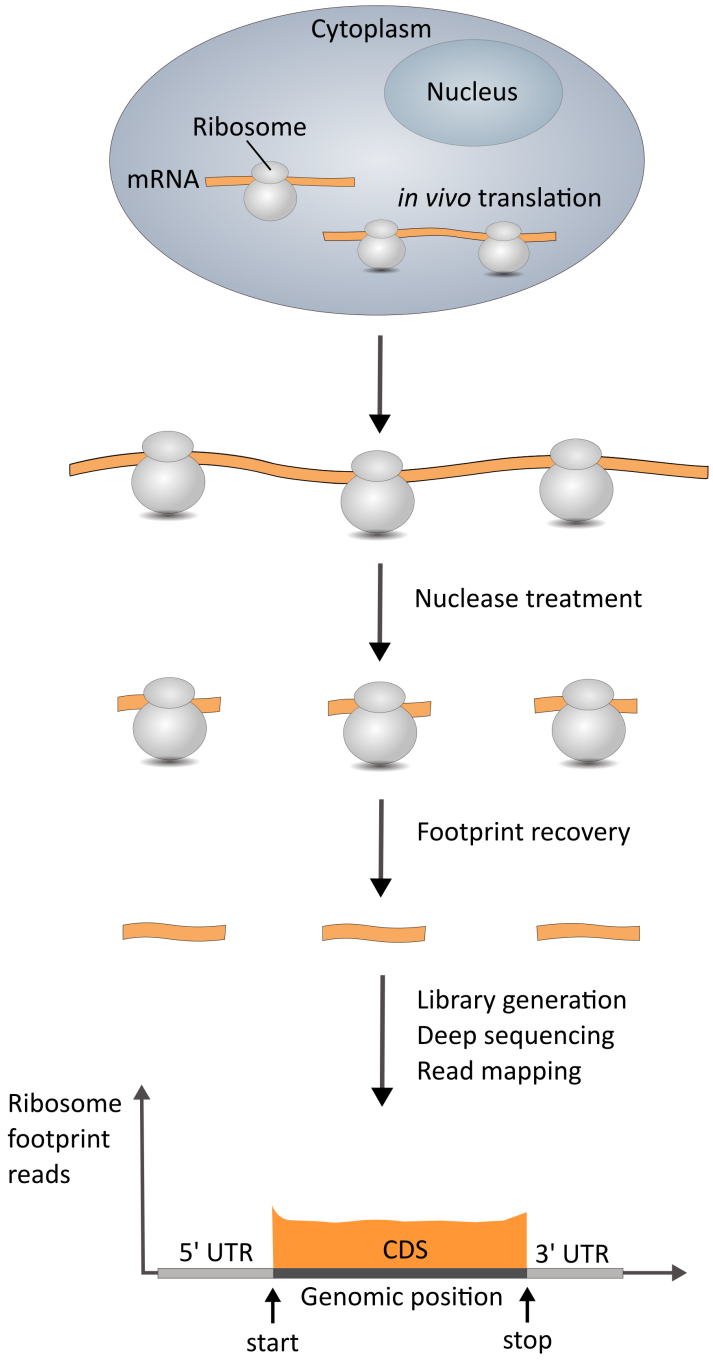

Figure 3.2: Ribosome profiling protocol. Ribosome profiling allows to precisely monitor translation at the nucleotide level. The immobilization of initiating ribosomes followed by nuclease treatment and digestion of mRNA fragments not protected by ribosomes together with deep–sequencing of ribosome–occupied footprints enables to decipher translational complexity in a genome–wide scale. The figure was adapted from [169, 191].

Ingolia et al. [44] applied ribosome profiling to mouse embryonic stem cells to define the proteome of a mammalian system. They found that more than half of the mRNA sequences encoded at least two ORFs, while some transcripts (16%) were reported to exhibit even more than four initiation sites. Moreover, most of the non–AUG start sites were found to initiate uORFs rather than dORFs. It was found that the overall start codon distribution is dependent on the location within the transcript: more than half of the start sites located in the CDS comprise AUG, whereas CUG and GUG, in addition to AUG, are widely used as upstream alternative start sites [44]. Lee et al. [43] applied ribosome profiling to human embryonic kidney 293 (HEK293) cells. As translation inhibitors they used cycloheximide (CHX) and lactimidomycin (LTM). Both chemicals bind to the ribosomal E–site. Thereby, CHX can bind to initiating and elongating ribosomes, whereas LTM prefers initiating ribosomes with a tRNA–depleted E–site [43]. Thus, by combining both inhibitors, it is possible to differentiate initiating from elongating ribosomes [43]. In total, they identified about 10,000 transcripts that harbor almost 17,000 potential ORFs with initiation sites located in the 5′ UTR, CDS, and 3′ UTR. Both datasets [43, 44], which comprise various alternative start sites, were used in our studies.

3.1.2 Machine learning

In the following, the broad field of machine learning is introduced with a focus on the models that were used in this thesis. There are numerous sources that provide theoretical and practical background information on statistical learning. This section is based on several references: Hastie et al. [195] and Bishop [196] provide statistical concepts to various methodologies from the statistical learning field, Boyd and Vandenberghe [197] focus on convex optimization methods, Boucheron et al. [198] demonstrate the theoretical background of classification models, Schölkopf [194] deals with support vector learning, and Smola and Schölkopf [199] provide an overview on support vector regression. For more information on these topics, please refer to the mentioned references. A very useful machine learning library is scikit–learn for the Python programming language [80]. Moreover, LIBSVM (A Library for Support Vector Machines) is a powerful software package for support vector classification and regression, which is also used within the scikit–learn SVM implementation [200].

The basic idea: learn from what you observe

In general, there are two types of learning: supervised and unsupervised learning. Here, we will focus on supervised learning. The main idea of supervised learning is to train a generalized prediction model based on observed data such that the learned estimator function is then applicable for the prediction of new unseen data. In more detail, we are given an $n \times d$ dimensional input data matrix X and an n dimensional output vector Y

$$X = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1d} \\ X_{21} & X_{22} & \cdots & X_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n1} & X_{n2} & \cdots & X_{nd} \end{bmatrix}, \quad Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix}$$

with n training samples $(X_{ij})_{i=1}^{n}$ that are represented by d features $(X_{ij})_{j=1}^{d}$ and an output vector $(Y_i)_{i=1}^{n}$. Dependent on the output space of Y, the learning method denotes regression with $Y_i \in \mathbb{R}$ or classification with $Y_i \in K$ and K classes or output labels. If $K = 2$, it is termed binary classification, whereby the labels K can be defined arbitrarily such as $Y_i \in \{-1, 1\}$ or $Y_i \in \{0, 1\}$. A simple linear model to predict the output $\hat{Y}$ dependent on observations $X^T = (X_1, \ldots, X_d)$ is then defined as

$$\hat{Y} = \hat{w}_0 + \sum_{j=1}^{d} \hat{w}_j X_j$$

with the weight vector or coefficients $\hat{w}$. The weight vector $\hat{w}_0$ is termed the intercept or bias and serves as a starting point for the model. For convenience, predicted values are denoted with a hat symbol such as $\hat{Y}$. In order to find the coefficients $\hat{w}$ that multiplied with the input X best approximate the output Y, a loss or error function must be optimized, and in this context this means minimized.

In that sense, the goal of supervised learning algorithms is to find an optimal function $f : \mathbb{X} \to \mathbb{Y}$ over all possible functions $\mathcal{F}$ such that $f(x) = \hat{y}$ best approximates the output y in terms of a minimized loss or cost objective function $L(f(x), y) = L(\hat{y}, y)$. Thereby, $\mathbb{X}$ denotes the input space and $\mathbb{Y}$ refers to the output space. Thus, learning algorithms can be described as optimization problem

$$\arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(f(X_i), Y_i) + \lambda R(f)$$

with a loss function $L(f(x), y)$ that is minimized over all points in the training dataset, and a regularization functional $R(f)$ together with a regularization parameter $\lambda > 0$. Regularization provides a trade–off between the complexity of a function and the error in order to avoid overfitting of the training data. The most popular and widely used loss function when considering a regression problem is the L_2 or squared error loss $L(f(x), y) = (f(x) - y)^2$. In contrast to the squared error loss, the estimation of the L_1 or absolute loss $L(f(x), y) = |f(x) - y|$ is more robust against outliers. Nevertheless, an optimal solution for the squared error loss can be computed easily compared to L_1, thus explaining the widespread application of the L_2 loss compared to the L_1 loss. Considering classification, the $0 - 1$ loss $L(f(x), y) = 1(f(x) \neq y)$ is commonly used and simply penalizes the number of misclassifications when the output y differs from the predicted value $f(x)$.

Linear least squares regression

As mentioned, linear regression models attempt to predict an output $\mathbb{Y} = \mathbb{R}$ based on observations $\mathbb{X} = \mathbb{R}^d$ using linear functions $f(x) = \hat{w}_0 + \hat{w}_1 x_1 + ... + \hat{w}_j x_j$. An approach for an estimation of coefficients $\hat{w} = (\hat{w}_0, \hat{w}_1, \hat{w}_2, \ldots, \hat{w}_d)^T$ is the widely used method of least squares that attempts to optimize the squared loss $L(f(x), y) = (f(x) - y)^2$ by minimizing the residual sum of squares

$$RSS(w) = \sum_{i=1}^{N} (y_i - f(x_i))^2 = \sum_{i=1}^{N} \left(y_i - w_0 - \sum_{j=1}^{d} w_j x_{ij}\right)^2$$

given the training data $\{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$. Taking the derivative with respect to the weight vector $\hat{w}$ results in the solution of the linear least squares regression

$$\hat{w} = (X^T X)^{-1} X^T Y.$$

Subsequently, predicted values $\hat{y}$ can be calculated by multiplying the weight vector $\hat{w}$ with the input matrix X

$$\hat{y} = X\hat{w} = X(X^T X)^{-1} X^T Y.$$

Here, we assume that a constant variable 1 is contained in the $n \times (d + 1)$ input matrix X and the intercept $\hat{w}_0$ is included in the weight vector $\hat{w}$. By doing so, the loss function optimizes the distance between the predicted values f(x) and the output y.

Support vector classification, regression, and the kernel trick

A support vector machine (SVM) is a widely used learner to solve classification and regression problems. One advantage of SVMs is their robustness against outliers. Moreover, a similar formulation of these optimization problems allows an application of SVMs to classification as well as regression problems. The difficulty of linearly non–separable cases is counteracted using the so–called kernel trick. Support vector machines, hard–margin and soft–margin case, together with the kernel–trick are described in the following.

Support vector classification The aim of support vector classification is to find an optimal separating hyperplane between two classes by maximizing the distance to the nearest point of each class. The training data is given as $x_i \in \mathbb{R}^d$ with the class labels $y_i \in \{-1, 1\}$. The support vector classifier is defined as

$$f(x) = sign[x^T \hat{w} + \hat{w}_0]$$

with the weight vector $\hat{w}$, the offset $\hat{w}_0$ and the signum function $sign(x)$. Hence, dependent on the sign of the linear separating hyperplane, points are classified as either $\hat{y}_i = 1$ if $f(x_i) > 0$ or $\hat{y}_i = -1$ if $f(x_i) < 0$. Since there are multiple possibilities for the values of $\hat{w}$ and $\hat{w}_0$, the idea of support vector classification is to maximize the distance between the two classes. This optimization results in a unique solution to optimally separate the training data. The closest points from either class are called support vectors giving this classifier its name, whereas the space between these training points is called margin. Figure 3.3 illustrates the hard–margin case of support vector classification that does not tolerate misclassification. The soft–margin case is explained below.

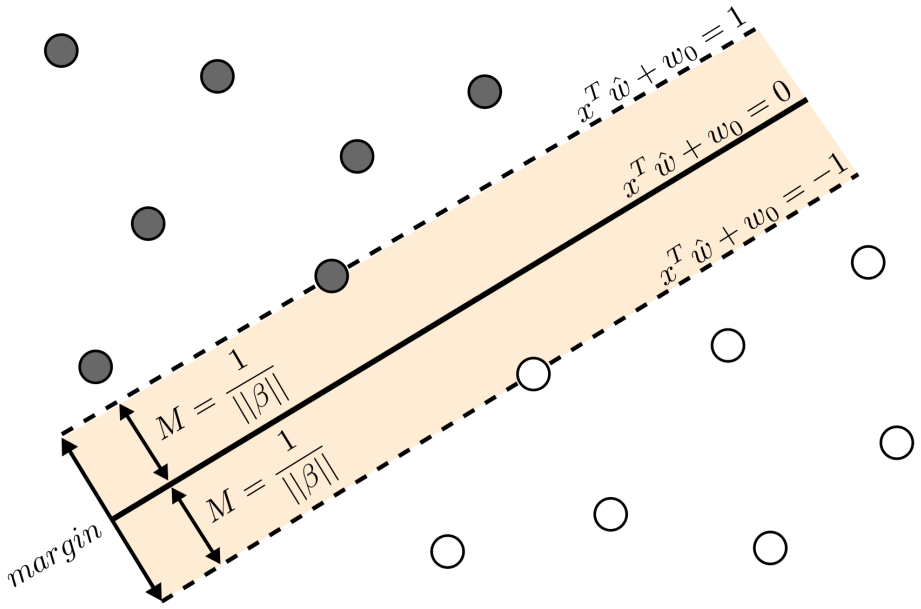

Figure 3.3: Hard–margin support vector classification. To find an optimal separating hyperplane, the distance between the training points from either class $y_i \in \{-1, 1\}$ is maximized. Training points $x_i \in \mathbb{R}^d$ that lie on the margin are called support vectors. The support vector classifier is relatively robust against outliers since only the support vectors have non–zero weights and hence an influence on the weight vector $\hat{w}$. The strict hard–margin case does not allow classification error. The figure was adapted from [195].

Thus, assuming the classes are separable by a linear decision boundary, the optimization problem to find an optimal separating hyperplane between two classes can be written as

$$\max_{w\in\mathbb{R}^d,\, w_0\in\mathbb{R}} \frac{1}{||w||}$$
$$\text{subject to: } y_i(x_i^T w + w_0) \geq 1, \;\; i = 1, \ldots, n$$

with the training data $x_i \in \mathbb{R}^d$, the class labels $y_i \in \{-1, 1\}$, and the margin $M = \frac{1}{||w||}$. The optimization problem is often equivalently written as differentiable convex optimization problem

$$\min_{w\in\mathbb{R}^d,\, w_0\in\mathbb{R}} \frac{1}{2}||w||^2$$
$$\text{subject to: } y_i(x_i^T w + w_0) \geq 1, \;\; i = 1, \ldots, n.$$

As mentioned, training points are by definition not allowed to fall into the margin, this criterion only holds for linearly separable data and is known as hard–margin case. This is equivalent to the requirement that the training error amounts to zero.

The strict criterion of not allowing any errors is weakened by considering the soft–margin case that introduces slack variables ξ to find a linear decision boundary between non–linearly separable data. Based on a better applicability to real datasets, the soft–margin case is normally used in practice. Thus, the soft–margin case allows misclassification of training data points and is in consequence more robust against overfitting. The soft–margin optimization problem can be formulated as

$$\min_{w\in\mathbb{R}^d,\, w_0\in\mathbb{R},\, \xi\in\mathbb{R}^n} \frac{1}{2}||w||^2 + C\sum_{i=1}^{n} \xi_i$$
$$\text{subject to: } y_i(x_i^T w + w_0) \geq 1 - \xi_i, \;\; \xi_i \geq 0, \; i = 1, \ldots, n$$

with slack variables $\xi_i \geq 0$ and cost parameter $C > 0$. The penalty parameter C regularizes the balance between a large margin (small C values), which allows training points to fall into the margin and actually being misclassified, and a small margin (large C values) leading to a smaller loss. Note that $C = \infty$ is thereby equivalent to the linearly–separable hard–margin case. The soft–margin case is depicted in Figure 3.4.

Support vector regression Beside support vector classification, support vector machines can also be applied to regression problems. The general idea of support vector regression is to find a function $f(x)$ with maximum flatness that has precision ϵ, which means $f(x)$ deviates at most ϵ from the actual outcome y_i for all training data points. Support vector regression can be formulated as the optimization problem

$$\text{minimize} \quad \tfrac{1}{2}||w||^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*)$$

$$\text{subject to} \quad \begin{cases} y_i - f(x_i) \leq \epsilon + \xi_i \\ f(x_i) - y_i \leq \epsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases}$$

with n training samples $(X_i)_{i=1}^n$, the weight vector w, slack variables ξ_i, ξ_i^* to tolerate some errors, and precision ϵ. The penalty parameter $C > 0$ regularizes the tradeoff between the accuracy ϵ and the flatness of the function $f(x)$. This formulation is very similar to the soft–margin support vector classification. Figure 3.5 visualizes the principle of support vector regression.

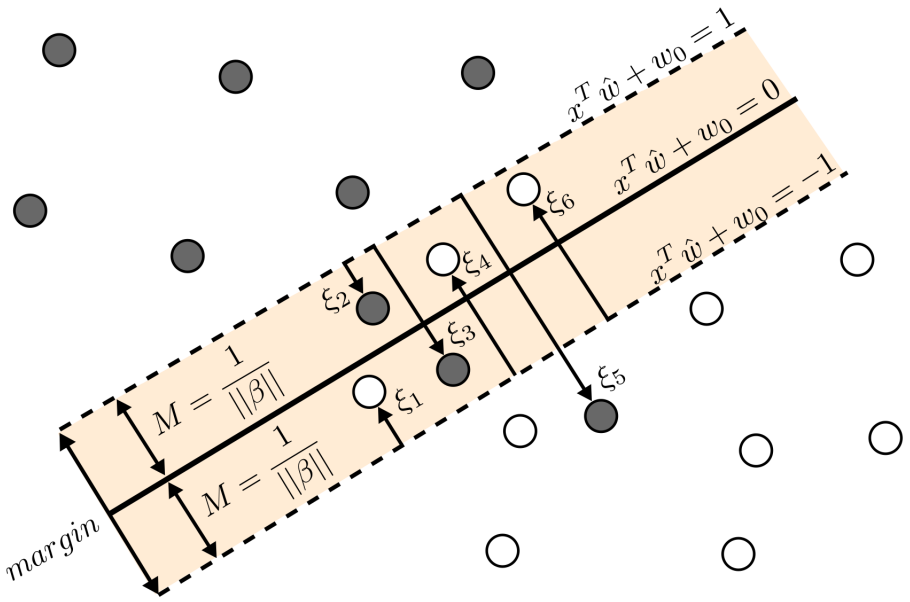

Figure 3.4: Soft–margin support vector classification. The soft–margin case is very similar to the hard–margin case, see Figure 3.3. Although compared to the strict hard–margin case, the soft–margin support vector classification is optimized with respect to the summed–up distances of all misclassified points ξ_i. The figure was adapted from [195].

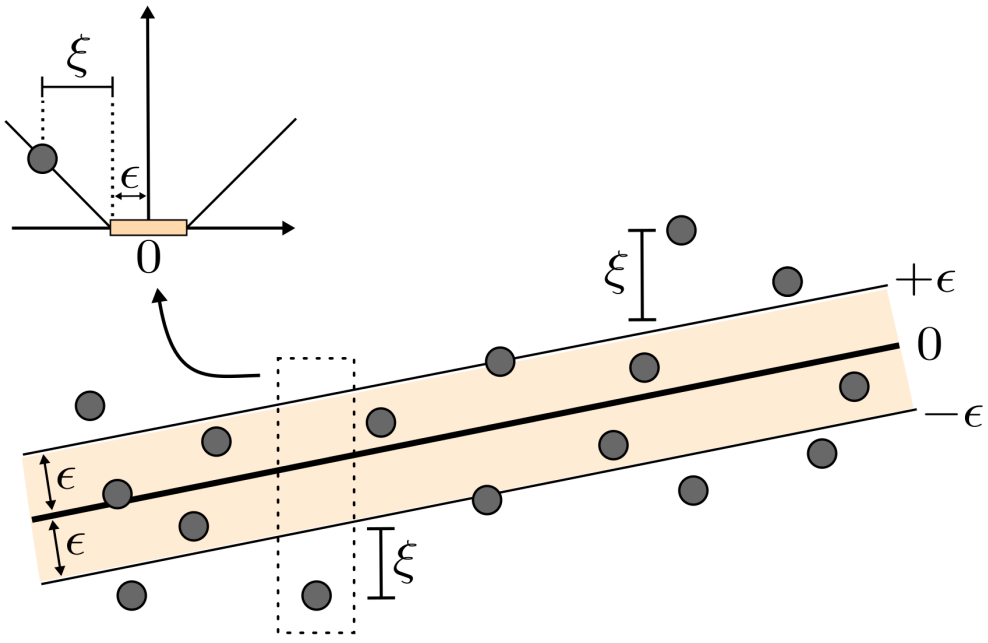

Figure 3.5: Support vector regression. Shown is the soft–margin case. Points are penalized linearly if the function $f(x)$ deviates more than precision ϵ from the output y_i. Thus, errors are ignored in case they are smaller than ϵ. The figure was adapted from [199].

Kernel trick Most often, data is not separable by linear decision functions, which calls for an integration of more universal decision boundaries. A widespread principle is the use of a kernel function $k(x, y) = \phi(x) \cdot \phi(y)$ that transforms input data from the input space, via a feature map Φ, into a high–dimensional feature space such that a linearly separating hyperplane can be generated, see Figure 3.6. This is known as the so–called kernel trick. Hence, a reasonable determination of a kernel function allows to find a linearly separating hyperplane in the high–dimensional feature space even though this data was not linearly separable in the original input space. Note that a linear decision surface in the feature space is equivalent to a non–linear hyperplane in the input space. Several kernel functions exist such as the radial basis function (RBF) kernel, which is defined as

$$k(x, y) = exp(-\frac{||x - y||^2}{c})$$

with $||x - y||^2$ denoting the squared Euclidean distance. A popular RBF kernel is the Gaussian kernel with $c = 2\sigma^2$.

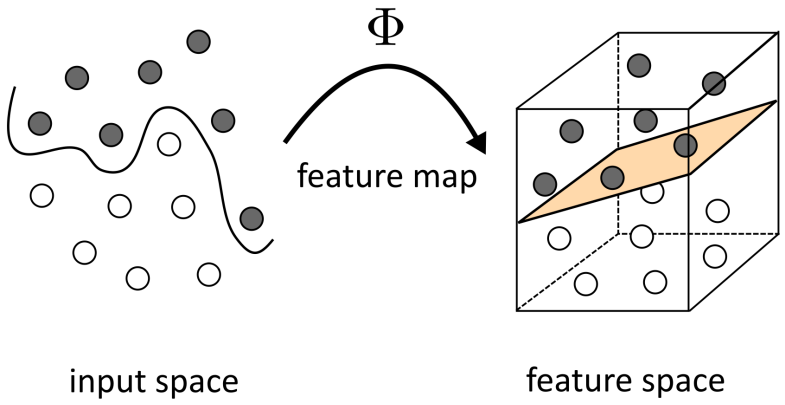

Figure 3.6: Geometric representation of the kernel trick. A mapping from the input space to a high–dimensional feature space using a feature map Φ allows to generate a separating hyperplane to classify linearly non–separable data in the input space. The figure was adapted from [201].

Find the best model: performance measurements and cross–validation

Performance measurements are applied for model and parameter selection and are thus used to assess a prediction model [195, 202]. Considering classification problems, common measurements used in computational biology are accuracy, sensitivity, specificity, and precision. All measurements contrast the number of correctly and incorrectly classified cases. This results in four possible classification scenarios: true positive (TP), false positive (FP), true negative (TN), and false negative (FN), compare with the confusion matrix in Table 3.1.

Table 3.1: Confusion matrix. Dependent on the actual and predicted class, cases are divided into TP, TN, FP, and FN classifications.

	Positive (Predicted)	**Negative (Predicted)**
Positive (Actual)	TP	FN
Negative (Actual)	FP	TN

Evaluation measures are then computed based on the number of TP, TN, FP, and FN classifications. The most common performance measurements are defined in Table 3.2. Dependent on the research field, the same formula can be named differently. For instance, sensitivity is referred to as recall in the Information Retrieval field.

Table 3.2: Performance measurements applied in classification problems. The depicted performance measurements are commonly used to assess the performance of classification models [202]. Dependent on the research field, true positive rate is also known as sensitivity or recall, whereas true negative rate denotes specificity.

Measurement	Formula	Description
True positive rate	$\frac{\text{TP}}{\text{TP+FN}}$	Proportion of positive cases correctly predicted as positive. This measurement is also known as sensitivity or recall.
True negative rate	$\frac{\text{TN}}{\text{TN+FP}}$	Proportion of negative cases correctly predicted as negative. This measurement is also known as specificity.
False positive rate	$\frac{\text{FP}}{\text{FP+TN}}$	Proportion of negative cases wrongly predicted as positive.
False negative rate	$\frac{\text{FN}}{\text{FN+TP}}$	Proportion of positive cases wrongly predicted as negative.
Precision	$\frac{\text{TP}}{\text{TP+FP}}$	Proportion of positive cases out of all cases predicted as positive.
Accuracy	$\frac{\text{TP+TN}}{\text{TP+FP+TN+FN}}$	Proportion of correctly predicted cases.

Another widely used performance assessment method in binary classification is Receiver Operating Characteristics (ROC) analysis that compares the false positive rate (FPR) with the true positive rate (TPR) for every possible discrimination threshold [195, 202, 203]. The FPR can be calculated as $1-$specificity while TPR denotes sensitivity, see also Table 3.2.

In performing ROC analysis, TPR is plotted against FPR, see Figure 3.7. Thus, the best performing classifier(s) can be found at the top left corner close to (0,1) with FPR $= 0\%$ and TPR $= 100\%$, whereas the worst performing learning methods are situated at the bottom right corner near (1,0) with FPR $= 100\%$ and TPR $= 0\%$, compare with Figure 3.7. A random classifier is denoted by the positive diagonal with FPR $=$ TPR. A common approach to quantitatively compare different classifiers by ROC analysis and finding the on average best performing model is to maximize the area under the ROC curve (AUC) [203]. Optimizing the AUC rather than minimizing the error rate can be a reasonable procedure when, for instance, skewed binary class distributions are present [203].

$K-$ fold cross–validation is often used to evaluate a learning method and select the best performing model [195]. Especially when data is sparse, $K-$ fold cross–validation enables a repeated data usage by partitioning data into $K-1$ training datasets and a k–th test dataset. This test dataset is often referred to as validation dataset. Thereby, the model is fitted to the training dataset and the prediction error is calculated by applying the learned model to the validation set. This is repeated $k = 1, ..., K$ times and the overall cross–validation prediction error is the combination of the K measurements. K–fold cross–validation is of practical relevance to avoid overfitting and thus allows for a generalization of the learning function. In practice, common choices for K are $K = 5$ or $K = 10$. The case where K equals the number of observations is known as leave–one–out cross–validation.

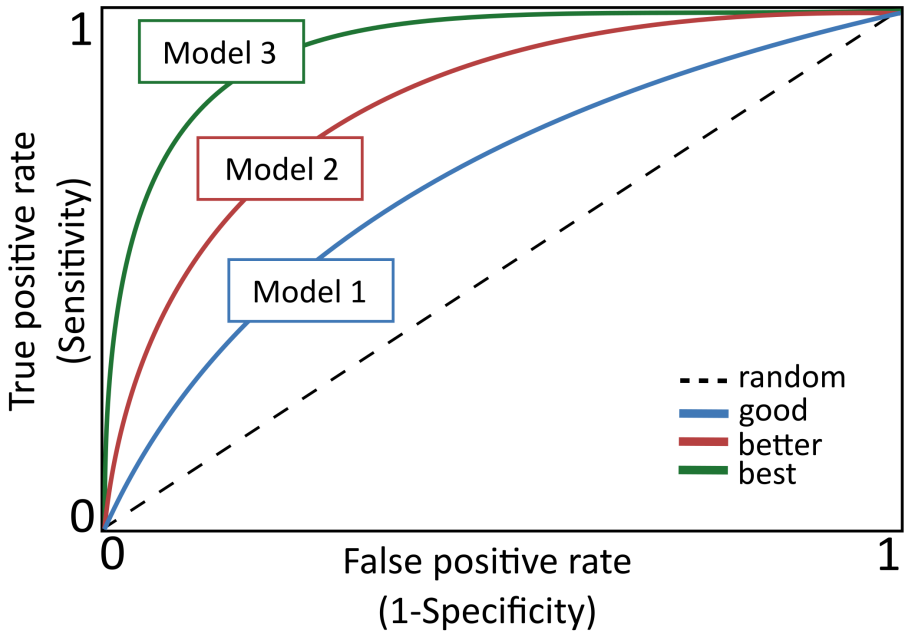

Figure 3.7: ROC analysis. ROC–curves are widely used to compare the performance of learning methods. The best classifier(s) are located in the upper left corner (FPR $= 0\%$, TPR $= 100\%$), whereas the worst performing classifiers can be found in the lower right corner (FPR $= 100\%$, TPR $= 0\%$). The positive diagonal represents a random classifier (FPR $=$ TPR) in the binary case.

3.1.3 Web development

An implementation of bioinformatics algorithms as web service bears the major advantage of easy reachability and usability by interested researchers. In contrast to a local installation, all required additional programs or command line tools are already installed on the web server. Thus, time–consuming further installations and adjustments to the local system can be avoided. The possibility to just test the usability of an application, at best when example data is provided that can be filled in by just clicking a button, rather than the need to install all required packages and tools, increases the probability that the service is used. On the other hand, when the web application stores generated results, a reliable and sufficiently large data storage system is needed to sustain enough free hard disk space. This is avoided when a bioinformatics tool is provided as a software package. Thus, a web service application and a local installation have both advantages and disadvantages. During this thesis, we decided to implement both, a web service application and a downloadable tool to be installed and utilized on a local machine. *PreTIS*, explained in this chapter, was implemented as web service application, whereas *MutaNET*, see Chapter 5, is a downloadable software suite that also provides a functional graphical user interface (GUI).

This section deals with the basics of web development and necessary (programming) languages to implement and provide interactive (bioinformatics) web applications. Most webpages are based on a combination of HTML, CSS, JavaScript, and a Document Object Model (DOM) for a well–presented and dynamic web application. This is known as Dynamic HTML and is explained in the following sections. We used HTML, CSS, JavaScript, PHP, and Python to develop a dynamic web service rather than a simple command line tool, to predict and visualize initiation confidences of translation start sites from user provided human 5′ UTR and CDS sequences.

Hypertext Markup Language

The Hypertext Markup Language (HTML) is the standard markup language to describe the content of a website and is thus the basis of the World Wide Web. In principle, HTML documents contain instructions on the composition of a webpage, which incorporates headings, paragraphs, tables, figures, lists, or hyperlinks. These web components are designated as HTML elements. HTML documents are interpreted and rendered using standard web browsers. HTML was developed in 1989 by Tim Berners–Lee at The European Organization for Nuclear Research (CERN) [204] and is currently maintained by the World Wide Web Consortium (W3C) founded and headed by Berners–Lee [205]. The current version is HTML5 (as of September 2017).

HTML elements are described by tags using smaller–than ("<") and larger–than (">") symbols. Starting with <> and ending with </> enables to describe paragraphs with `<p>Please select...</p>` or to render images using `<img src="img.png" height="800" width="600">`, to name two examples. HTML documents are separated into `<head>` and `<body>` elements. The `<head>` section is used to embed (technical) metadata information, which is not displayed by the browser when the page is accessed. Metadata describes the HTML document and is used to embed the title, style, or additional scripts such as JavaScript. The `<body>` element describes the content that is presented by a HTML document. This includes text, tables, figures, (dynamic) plots, lists, and hyperlinks to subpages or external webpages. To give an example for HTML code, a short and adapted extract from a *PreTIS* web service subpage can be found in Listing 3.1. Since this serves as a descriptive HTML example and due to better legibility, please note that the content of some HTML elements is shortened such as the (omitted) parameter `enctype="multipart/form-data"` that is necessary for data transmission and embedded in the form–data element.

Listing 3.1: Shortened and adapted HTML example from the *PreTIS* web service.

```
<html>
  <head>
    <title>PreTIS</title>
    <!-- Embedding of CSS style sheets. -->
    <link href="default.css" rel="stylesheet"/>
    <link href="fonts.css" rel="stylesheet"/>
  </head>
  <body>
    <h1><a href="index.php">PreTIS</a></h1>
    <p>Prediction of translation initiation sites</a></p>
    <!-- Data is forwarded after successful validation. -->
    <form method="POST" action="execute.php" onsubmit="return
        validate(document)">
      <div class="title"><h2>Sequence form</h2></div>
      <div class="title"><h3>1. Human mRNA sequence:</h3>

      <!-- Sequence form: paste and upload. -->
      <p>5' UTR sequence:</p>
      <textarea id="utrP" name="n1" rows="5" cols="50"></textarea>
      or upload a TXT file <input type="file" id="utrU" name="n2"/>
      <button type="submit" class="myButton">Submit</button>
    </form>
  </body>
</html>
```

The <head> part (Lines 2–7) defines the title and embeds the CSS style sheets via an external <link> for a nice visualization. The <title> describes what is shown in the browser tab or in the search engine results, here "PreTIS". The <body> consists of an interactive form–data with method="POST" for a "hidden" HTTP request (Line 12). In contrast, method="GET" appends data to the URL, which will then be visible to a client and should be omitted when working with sensitive data. Prior to submission to the web server, user provided data is validated via onsubmit="return validate(document)" (Line 12). Thereby the document object is a node in the HTML DOM that allows access to and application of methods on these HTML document objects via JavaScript. In general, the DOM structures a document, like HTML or XML, as a tree with every node representing an object within the hierarchy. HTML node objects are separated into element nodes (head, body, or a heading h1), attribute nodes (href, width, or id) that further specify elements, text nodes that denote the text written in an element node, and comment nodes. Thereby, the document object itself represents the root. For instance, the JavaScript function document.getElementById("id"); can be used to retrieve the element with the given attribute ID "id" for validation purposes. Data concerning the *PreTIS* web service comprises mRNA sequences that can be pasted in a multi–row text field, created using the <textarea...></textarea> tag (Line 18), or sequences can be uploaded from a local directory by defining <input type="file"...> (Line 19). HTML elements allow to additionally define parameters such as image sizes or identifiers. These identifiers ("id") are used to enable access to elements and retrieve their content. Different headings with decreasing size and visibility are marked using h1, h2, and h3. Comments are written as <!-...-> and shown in turquoise color in Listing 3.1.

Cascading Style Sheets

Cascading Style Sheets (CSS) determines the visual look of HTML documents and is maintained by the W3C [205]. In contrast to a HTML document, which describes the content and the structure of a web page, CSS is responsible for the visual layout. This separation of content and visualization allows to reduce redundancy and define a concurrent layout style for several documents. CSS defines font families, font sizes, colors, table design, borders, and margins to name a few. The syntax of CSS is straightforward. A CSS rule comprises a selector followed by a declaration block that consists of one or more property:value pairs, each ending with a semicolon. A declaration block is enclosed by curly brackets:

```
Selector {
   Property1 : Value1;
   Property2 : Value2;
   ...
}.
```

Selectors determine the HTML elements the style should be applied to. Examples of selector types are element name selector, class selector, and id selector. The element selector recognizes the element name such as the body, a heading h1, or all paragraphs p. The class selector, starting with a "." character, refers to all elements with the given class attribute. A style sheet comprising .title h2 {...} will layout all h2 headings with class="title". The id selector, starting with a "#" character, affects all HTML elements with the given id attribute. For instance, the section can be styled using the selector #menu-wrapper {...}. Since identifiers within a HTML document should be unique, this layout will be applied to an individual element. Listing 3.2 shows a short example on how to define the layout in terms of fonts, colors, and arrangements using three different selector types.

TEMPLATED [206] is a broad collection of various style sheets and templates that are released under the Creative Commons Attribution [207] license. Thus, by citing or referring to the TEMPLATED webpage, the provided style templates can be used for private and commercial website design without any additional costs. *PreTIS* and *MutaNET* CSS style templates were retrieved from TEMPLATED and were adapted for our purposes. Note that the *MutaNET* web

page only provides additional information on this software and does not embed a web service application, see Chapter 5.

Listing 3.2: Shortened CSS example from the *PreTIS* web service.

```
1  /* Element name selector */
2  body {
3      margin: 0px;
4      padding: 0px;
5      background: #333333;
6      font-family: 'Muli', sans-serif;
7      font-size: 12pt;
8      font-weight: 300;
9      color: #363636;
10 }
11
12 /* Class selector */
13 .title h2 {
14     text-transform: uppercase;
15     letter-spacing: 0.10em;
16     font-weight: 700;
17     font-size: 1.8em;
18     color: #00AABB;
19 }
20
21 /* Identifier (id) selector */
22 #menu-wrapper {
23       color: #00AABB;
24 }
```

JavaScript

JavaScript (JS) can be embedded in HTML documents and enables the generation of dynamic and interactive of websites. JS is executed client–sided, which means in the web browser and thus on the processor of a client rather than on the web server. This reduces web server load and allows fast executions as data transmission to the web server is avoided. Disadvantages are potential incompatibility of scripts with the utilized browser (versions) that can lead to errors or divergent output. The syntax resembles programming languages derived from C. JS can be applied to edit websites via the DOM to perform data validation, manipulate HTML content, display dialog boxes containing error messages, or visualize data in a dynamic way. JS functions are integrated in HTML documents using a `<script>...</script>` tag or are part of HTML elements such as `<form...onsubmit="return function()">` or `<button onclick="function()">`. For instance, the method `document.getElementById("id");` enables access to the element with the attribute identifier `id="id"`. This allows element accessibility to specifically modify HTML content, HTML attributes like an alteration of images via the "`src`" attribute, or website layout via CSS. JS functionality and the scope of application is very broad. A reliable reference to learn more about this web programming language is W3Schools [208].

Listing 3.3 gives a short example of JS functionality as it was used in the *PreTIS* web service. The function named "`validate(document)`" checks whether the given mRNA sequences are valid (Lines 4–23). The sequences, which are pasted by a user into the text forms, are fetched using the DOM function `document.getElementById(id).value;` with `id="utrP"` and

`id="cdsP"` referring to the identifiers of the respective 5′ UTR and CDS text areas (Lines 6 and 7). The small code snippet then simply checks whether the sequences are not empty and only consist of the valid nucleotide characters {A, C, T, G, U}. In case, a condition is violated, an error message is displayed on the web page (Lines 13 and 16). The `if` statements would also allow to alert more details on the type of error, for instance directly referring to problems in the 5′ UTR or CDS, respectively. If all validation checks are successful, the function returns `true` (Lines 19–21) and the data is transmitted to the web server for further processing.

Listing 3.3: Shortened and adapted JavaScript example from the *PreTIS* web service.

```
1  /*
2   * User input is validated prior to submission to the web server.
3   */
4  function validate(document) {
5      //Retrieve sequence element values (pasted in text field) using attribute IDs.
6      var utrP = document.getElementById("utrP").value;
7      var cdsP = document.getElementById("cdsP").value;
8
9      if (utrP != "" && isSequence(utrP)) {
10         if (cdsP != "" && isSequence(cdsP)) {
11             //do something
12         } else {
13             alert("Please provide valid 5' UTR and CDS sequences.");
14         }
15     } else {
16         alert("Please provide valid 5' UTR and CDS sequences.");
17     }
18     ...
19     if (All criteria are fulfilled) {
20         return true;
21     }
22     return false;
23 }
24
25 /*
26  * Only characters {A,C,G,T,U} are allowed in mRNA sequences.
27  * Upper and lower case are both tolerated.
28  */
29 function isSequence(s) {
30     s.toUpperCase();
31     for (var i = 0; i < s.length; i++) {
32         if (!(s[i]=="A"||s[i]=="C"||s[i]=="G"||s[i]=="T"||s[i]=="U")) {
33             return false;
34         }
35     }
36     return true;
37 }
```

We took advantage of JS to validate user input, exemplary shown in Listing 3.3, prior to data redirection to the web server. Moreover, we used the Highcharts library as part of our *PreTIS* web service to display predicted translation initiation confidences as interactive bar plot.

Highcharts is implemented in JS, based on Scalable Vector Graphics (SVG), and provides a wide range of interactive and dynamic charts for webpages [209]. To display further information on individual start sites we also made use of a tooltip (mouse–over) functionality, which is provided by the Highcharts library. JS was also used to interactively highlight putative start sites within the 5′ UTR sequence to facilitate the identification of candidate start codons.

PHP: Hypertext Preprocessor

The widely used scripting language PHP is a recursive acronym for "PHP: Hypertext Preprocessor" and is used to develop dynamic web pages and interactively process user input. Unlike JavaScript, PHP is executed on the server–side and returns a HTML document upon a client request. Thus, only the HTML output is displayed to a client while the scripts are hidden. Moreover, there is no need for the client to install additional plugins required for client–sided scripts. However, unauthorized access to the server via a server–sided PHP script poses a security risk. PHP code is enclosed by start and end tags `<?php...?>`. PHP syntax resembles C, Perl, and Java programming languages. PHP scripts have the file extension ".php" and in our case additionally comprised of HTML, CSS, and JS code.

PHP was used to call the Python scripts `blast.py` and `predict.py` to execute the BLAST search, feature calculations, and prediction of translation initiation confidences. These Python scripts constitute the bioinformatics core functionality of the web service. As example, the `predict.py` call via combining PHP and Python is shown in Listing 3.4 (Lines 1–7). Thereby, the `popen()` function opens a pipe with the command parameter given as argument. The `$python_print` variable contains all values printed within the Python script, which is subsequently parsed and saved as an PHP `array()` for further usage like result visualization. Moreover, to display the returned results, a list representation to outline the codons that were selected by the user on the input form was created dynamically using PHP, see Listing 3.4 (Lines 9–21). Note that the last codon is appended individually to omit the last comma. Furthermore, the combination of HTML and PHP was used to present the returned nucleotide extensions, respective codons, frames, ORFs, stop codon positions, and predicted initiation confidences in tabular form. This is exemplarily shown with two column entries (extension and predicted value) in Listing 3.4 (Lines 23–47). Note that the variable `$feature_list` is retrieved from the previously saved and reopened PHP `array()` that contains all calculated results. PHP was also applied to display the sequences reported by BLAST and to write a CSV file containing all generated sequence–encoded features and prediction values.

JavaScript Object Notation

JavaScript Object Notation (JSON) is a text format that was developed for a simplified data exchange between browser and web server [210]. JSON syntax is derived from JS objects and is based on a simple dictionary or map structure that consists of key:value pairs, separated by commas, and encompassed by curly brackets. Arrays are denoted by square brackets and double quoting is required for strings. Due to the plain syntax, JSON is readable by humans and machines making it suitable for data storage and browser–server communication. JSON text can be converted to a JS object for further processing, while the reverse conversion from a JS object back to JSON for subsequent transfer to a web server is also possible. The shortened JSON example

```
[
{"desc":null,"query":"ENSG00000196329","id":"ENST00000498181","seq":"GAGGA...","molecule":"dna"},
{"desc":null,"query":"ENSG00000196329","id":"ENST00000358647","seq":"ATGAC...","molecule":"dna"},
...
{"desc":null,"query":"ENSG00000196329","id":"ENST00000493304","seq":"CTTTC...","molecule":"dna"}
]
```

illustrates several cDNA (complementary DNA) sequences together with the Ensembl gene ID and the respective transcript IDs.

Listing 3.4: Shortened and adapted PHP example from the *PreTIS* web service.

```php
<?php
    // Call python script with human and mouse sequences as arguments.
    $command = sprintf("python predict.py %s %s %s %s %s", $human_utr,
        $human_cds, $mouse_utr, $mouse_cds, $codons);
    $python_print = popen($command,"r");
    ...
    $result_array = PHP array() with parsed/processed $python_print results;
?>

<?php
    // Dynamic list of user selected codons and presentation using HTML.
    $codons = $result_array[1];
    $codon_str = "";
    for($i = 0; $i < count($codons)-1; $i++) {
        $codon_str .= str_replace("T", "U", $codons[$i]).", ";
    }
    $codon_str .= str_replace("T", "U", $codons [$i]);

    echo "<p>";
    echo "<b>Selected codon(s): </b>". $codon_str."\n";
    echo "</p><br/>\n";
?>

<?php
    // Initialize Table and paste header information.
    echo "<div class=\"table_class\">\n<table id=\"table_id\">\n";
    $table_entries= array("Extension", "Codon", "Prediction");
    echo "<tr>";
    for ($i = 0; $i < count($table_entries); $i++) {
        echo "<td><b>" . $table_entries[$i] . "</b></td> ";
    }
    echo "</tr>\n";

    // Fill Table with values.
    $feature_list = $result_array[0];
    foreach ($feature_list as $ext => $feat_dict) {
        $codon = str_replace("T","U",$feat_dict["codon"]);
        $pred = $feat_dict["pred"];
        // Do some coloring based on predicted value using if...else statements.
        if...else (...) {
            $color_txt = ...;
        }
        $ext = "<font color=\".$color_txt.\">".$ext."</font>";
        $pred = "<font color=\".$color_txt.\">".round($pred,2)."</font>";
        echo "<tr><td>".$ext."</td><td>".$pred."</td></tr>\n";
    }
    echo "</table>\n</div>\n";
?>
```

These sequences of human gene *GIMAP5*, having gene ID ENSG00000196329, were retrieved from the Ensembl web server using the Ensembl REST API [211] with an appropriate URL (see below). The Ensembl Representational State Transfer (REST) web server allows access and retrieval of various datasets in JSON and FASTA format [211]. The respective request for the CDS, rather than cDNA, is very similar by slightly adapting the URL, which results in the following JSON text:

```
[
{"desc":null,"query":"ENSG00000196329","id":"ENST00000498181","seq":"ATGGG...","molecule":"dna"},
{"desc":null,"query":"ENSG00000196329","id":"ENST00000358647","seq":"ATGGG...","molecule":"dna"}
]
```

Listing 3.5 shows the implementation of these HTTP requests returning JSON formatted data, which is then processed using JS to specifically modify the HTML document. This was implemented as part of the *PreTIS* web service for the automatic retrieval of 5′ UTR and CDS sequences given an Ensembl gene ID.

A JSON text is returned (Lines 1–6) by the web server request, which is then converted to a JS object (Lines 8–12). Following this, JS in combination with HTML is used to modify the text areas with the identifiers id="utrP" and id="cdsP" to display the retrieved sequences to a user (Lines 14–16). Both requests, cDNA and CDS, are necessary to extract the 5′ UTR and CDS for feature calculation and prediction of initiation confidence values, see below. The term cDNA, for complementary DNA, denotes reverse transcribed mRNA [212]. Thus, a cDNA is derived from mRNA and is hence only composed of coding sequences. Beside the sequences, the respective transcript ID(s) are displayed to a *PreTIS* user as well. A client can then choose a transcript that should be scanned for alternative start sites by *PreTIS*.

Listing 3.5: HTTP request from the *PreTIS* web service including JSON and JavaScript.

```
1  // HTTP Request with example Ensembl gene ID for human gene GIMAP5.
2  var id = "ENSG00000196329";
3  var url_cDNA = "http://rest.ensembl.org/sequence/id/"+id+"?content-type=
       text/x-json;type=cdna;multiple_sequences=1";
4  var url_CDS = "http://rest.ensembl.org/sequence/id/"+id+"?content-type=
       text/x-json;type=cds;multiple_sequences=1";
5  var json_cDNA = Retrieve cDNA with XMLHttpRequest() and var url_cDNA;
6  var json_CDS = Retrieve CDS with XMLHttpRequest() and var url_CDS;
7
8  // Conversion of JSON data to JS object and processing.
9  var array_cDNA = JSON.parse(json_cDNA);
10 var array_CDS = JSON.parse(json_CDS);
11 var UTR_seq = 5' UTR sequence from processing cDNA and CDS arrays;
12 var CDS_seq = CDS sequence from processing cDNA and CDS arrays;
13
14 // Update the sequences in text areas with IDs id="utrP" and id="cdsP".
15 document.getElementById("utrP").value = UTR_seq;
16 document.getElementById("cdsP").value = CDS_seq;
```

3.1.4 Data sources and bioinformatics tools

This *PreTIS* project is based on several datasets that are necessary for the integration of mRNA sequence information and alternative start sites located in the 5′ UTR. Moreover, three bioinformatics software tools were applied to support the calculation of some sequence–encoded features that were used in the *PreTIS* prediction model. Both, the datasets and software tools are presented in the following.

Data integration: Ensembl BioMart and non–canonical start sites

The Ensembl Genomes project was developed from 2000 until 2009 as a cooperate work between the European Bioinformatics Institute and the Wellcome Trust Sanger Institute [55]. Ensembl aims at providing genomic data such as gene annotations, nucleotide, and protein sequences as well as sequence alignments, disease specific and regulatory information, GO annotations, or variation data. So far (as of July 2017) the latest release is Ensembl version 89 from May 2017 supporting data for 86 vertebrate genomes. The Ensembl Genomes BioMart is a comprehensive database that allows an easy retrieval of various genomic data mentioned beforehand. We used Ensembl BioMart to download and store annotated genomic mRNA sequences for human and mouse. Moreover, the Ensembl REST API [211] was used to automatically retrieve human 5′ UTR and CDS sequences with a human gene ID from the Ensembl web server. This was implemented in the *PreTIS* web service for user convenience.

In total, we used three datasets comprising alternative translation start sites that were detected by experimental ribosome profiling. Two datasets that were based on HEK293 cells [43] and mouse embryonic stem cells (ES cells) [44] provided various alternative AUG and near–cognate start codons. These datasets were used for feature calculation and establishment of the prediction model. A third dataset [54], only comprising AUG starts from a HEK293 cell line, was used for evaluation purposes.

BLAST: Basic Local Alignment Search Tool

The Basic Local Alignment Search Tool (BLAST) is the most widely used algorithm for sequence similarity searches [66, 67] with the publication from 1990 cited more than 66,000 times until today (as of July 2017, see `https://scholar.google.com/`). The heuristic search enables very fast computation of locally optimal sequence alignments, making BLAST applicable and indispensable for the bioinformatics research field. BLAST software is subject to continuous development and extensions. We applied `blastn` (nucleotide–nucleotide BLAST) to find orthologous mRNA sequences between human and mouse in order to calculate start site and 5′ UTR sequence conservation.

MUSCLE: MUltiple Sequence Comparison by Log–Expectation

MUSCLE, for MUltiple Sequence Comparison by Log–Expectation, software enables fast calculation of multiple sequence alignments for both nucleotide and protein sequences [69, 70]. The starting point is a heuristic generation of a progressive sequence alignment, which is then iteratively refined. The methodology is based on distance measurements, a sophisticated scoring function, and tree methods such as the neighbor–joining [213] and UPGMA [214] methods. MUSCLE was used in combination with BLAST [67] to generate human–mouse sequence alignments and calculate translation start codon and 5′ UTR sequence conservation.

ViennaRNA: RNA secondary structure prediction

The ViennaRNA Package supports the prediction of RNA secondary structures [68]. The dynamic programming algorithm is thereby based on the principle of energy minimization. By implementing different distance measures, the ViennaRNA Package can also be used for RNA secondary structure comparisons. We applied the `RNAfold` function of the ViennaRNA Package to predict mRNA secondary structure and minimum free energies from a given a mRNA sequence. The minimum free energy of an mRNA secondary structure downstream of a potential translation start codon was reported to have an effect on translation initiation [183, 187, 188].

3.2 Introduction

Translation initiation is a more complex process than reported in common textbooks. Experimental work showed that the canonical AUG–Methionine translation start is not always used to initiate eukaryotic translation [40, 41, 43, 44, 54]. Further alternative codons located upstream of the annotated AUG start can also serve as additional functional start sites and form additional or alternative ORFs. Those non–AUG triplets are postulated to differ from AUG by one nucleotide and hence comprise CUG, UUG, GUG, AAG, ACG, AGG, AUA, AUC and AUU [42]. Translation can proceed in–frame as well as out–of–frame relative to the main open reading frame [169]. This can, for example, lead to (small) upstream ORFs resulting in short peptides or to extended proteins when translation initiation takes place at an in–frame start codon located upstream of the canonical start.

Ribosome profiling data provides information on the density of ribosomes located at different regions of the transcript upon application of small chemicals that block the elongation process [44, 170]. Regions which are protected by ribosomes are not digested in the next step when the mRNA is treated with nucleases [191]. These ribosome footprints (RNA) have a length of about 30 nucleotides and are sequenced after nuclease treatment and subsequently mapped to a reference genome [191]. For example, Lee et al. [43] applied ribosome profiling to HEK293 cells. As translation inhibitors they used CHX and LTM, which both bind to the ribosome E–site [43]. While CHX can bind to both, initiating and elongating ribosomes, LTM prefers initiating ribosomes with a tRNA–empty E–site [43]. Thus, by combining both inhibitors, it is possible to differentiate initiating from elongating ribosomes [43]. Lee et al. identified 16,863 potential start sites out of about 10,000 transcripts whereby start sites were allowed to be located in the 5′ UTR, at the annotated start site, in the coding region, or in the 3′ UTR, respectively.

Possible biological reasons underlying alternative translation initiation are the expansion of biological variety, regulatory processes as well as targeting of the proteins to different compartments [174, 176, 181]. Touriol et al. [174] proposed that alternative translation initiation results in different proteoforms that can exhibit different functions as well as various cell localizations, which is of great importance for cell fate. Moreover, some codons (e.g. AUG or CUG) are more frequently used as translation initiation starts than other codons [43, 44].

So far, several bioinformatics studies have addressed the task of predicting alternative translation start sites or ORFs. The majority of these studies only considered AUG starts. Hatzigeorgiou [215] applied an artificial neural network embedding a linear search for AUG starts. They achieved 94% accuracy and were able to predict the correct start site in 60% of human cDNAs. Saeys et al. [216] developed a meta–tool that combines three simple AUG start site predictors that consider either PWMs, k–mer frequencies or the number of stop codons downstream of a start site. This combination of several simple predictors, named StartScan, resulted in a sensitivity of 80%, tested on human chromosome 21. Sparks and Brendel [217] argued that when one only searches for one translation start, predicting the leftmost (i.e. the most upstream) AUG as sole correct translation start yielded specificity and sensitivity of 94%, respectively. Chen et al. [218] used a flexible window and represented human DNA as k–tupels that reflect the nucleotide composition and also integrated the physicochemical properties of amino acids. For AUG codons, their method achieved an accuracy of 98%. A web service of their algorithm is available. Besides, there also exist several web–based tools for ORF identification. ORF Finder searches for ORFs given the accession number or sequence and the genetic code [219]. ORF–Predictor provides an *ab initio* prediction of ORFs based on expressed sequence tag or cDNA sequences and BLASTX alignments or intrinsic sequence signals [220].

Only few studies involved ribosome profiling data or considered in– and out–of–frame start codons differing from AUG. Ivanov et al. [42] studied annotated human 5′ UTRs via sequence alignments with orthologous species followed by a manual evaluation [42] to detect non–AUG initiation in human sequences. They predicted 42 novel genes with non–AUG upstream translation initiation. For 25 of these genes non–canonical translation initiation could be experimentally validated using Western blot as well as ribosome profiling data. They also confirmed 17 alternatively translated genes that were known at this time. Crappé et al. [221] applied an SVM approach to ribosome profiling data to detect conserved small open reading

frames (sORFs) in mouse that code for micropeptides ($10-100$ amino acids). Michel et al. [222] used ribosome profiling data to calculate translation initiation probabilities. In contrast to our work, they focused on the initiation strength of a putative start site as a function of the number of ribosome footprints. To our best knowledge, no study so far has evaluated the general properties of human start codons considering both AUG and all near–cognate codons, in– and out–of–frame, based on start sites identified by applying ribosome profiling, and exploited this to predict the initiation confidence from the mRNA sequence.

The aim of this work was to analyze alternative translation start sites (AUG and near–cognate codons) with respect to sequence–based features to differentiate between true and false start sites. We used start sites that were identified by applying ribosome profiling to HEK293 cells [43, 54] and mouse ES cells [44] as our primary datasets. Based on mRNA sequence information we generated SVM models as well as a linear regression model for human and mouse sequences. The learned model can then be applied to mRNA sequences not covered by ribosome profiling data or to investigate the impact of mutations in the flanking sequence context of a start site on its translation initiation confidence. Our web service *PreTIS* visualizes putative alternative start sites and the predicted initiation confidence in human.

3.3 Materials and methods

In the following, the materials and methods of our *PreTIS* project are presented. First, the datasets are introduced starting with an overview on the data integration approach, followed by a detailed presentation of the data processing and appropriate data storage. Next, the sequence–encoded features are calculated based on the former processed databases.

3.3.1 Data processing and integration

This project is based on several datasets that are necessary to develop reliable prediction models for alternative translation start sites in 5′ UTRs. In the following, all data sources, the generation of a negative set containing false translation start sites, and the integration of all these datasets as clear and fast accessible data structures is presented.

Genomic mRNA sequences

Annotated genomic mRNA sequences for human and mouse were retrieved from Ensembl biomart (Ensembl version 77 [55]). We only included curated mRNA sequences with available mRNA RefSeq identifier (starting with NM_). It was recently shown that 85% of the start sites used to initiate translation are conserved between human and mouse [43]. Thus, we used homologous pairs of human and murine sequences to calculate the conservation of putative start codons as well as the 5′ UTR sequence conservation (see below). We identified the respective murine orthologous mRNA sequences using the approach by Ivanov et al. [42] and used the first `blastn` [67] hit as the respective ortholog (default `blastn` parameters).

Alternative 'true' and 'false' translation start sites

To identify putative start sites, each 5′ UTR was scanned for all AUGs and for alternative near–cognate start codons that differ from generic AUG by one nucleotide (CUG, UUG, GUG, AAG, ACG, AGG, AUA, AUC und AUU) and that are located either in–frame or out–of–frame with the main open reading frame. Different sequence–based features were then calculated for all putative start codons that have a downstream in–frame stop codon. To establish reliable true positive and true negative translation start site datasets for training and testing purposes, we used the findings from different ribosome profiling experiments [43, 44, 54]. Each dataset was analyzed independently. Note that the datasets used here contain translation start sites derived

from ribosome profiling data by the original authors (gene accession number, position relative to annotated start site, codon). We did not include raw ribosome profiling (footprint) data in our approach. In total, we trained two start site prediction models: a human prediction model based on the HEK293 dataset [43] and a mouse prediction model based on the Mouse ES dataset [44]. The third HEK293–AUG dataset [54] was used as validation set to further evaluate the reliability and robustness of the developed prediction model.

For training and testing of every classifier, we considered each start site (AUG and near–cognate) that matched a start codon found by ribosome profiling as a true start. False start sites were defined as follows: remaining candidate start sites (AUG and near–cognate) that were not detected by ribosome profiling and that are, based on the assumption of a linear scanning model, located at least 99 nt downstream of the transcription start site as well as upstream of the most downstream reported true translation initiation start. Figure 3.8 shows an example mRNA sequence that illustrates the grouping of true positive and true negative start sites for training and testing purposes based on ribosome profiling data. This start sites categorization was executed for each of the three datasets, each time based on the individual ribosome profiling experimental results [43, 44, 54].

```
  1 CGGUGAGGGU UCUCGGGCGG GGCCUGGGAC AGGCAGCUCC GGGGUCCGCG GUUUCACAUC
 61 GGAAACAAAA CAGCGGCUGG UCUGGAAGGA ACCUGAGCUA CGAGCCGCGG CGGCAGCGGG
121 GCGGCGGGGA AGCGUAUACC UAAUCUGGGA GCCUGCAAGU GACAACAGCC UUUGCGGUCC
181 UUAGACAGCU UGGCCUGGAG GAGAACACAU GAAAGAAAGA ACCUCAAGAG GCUUUGUUUU
241 CUGUGAAACA GUAUUUCUAU ACAGUUGCUC CAAUGACAGA GUUACCUGCA CCGUUGUCCU
301 ACUUCCAGAA UGCACAGAUG UCUGAGGACA ACCACCUGAG CAAUACUGUA CGUAGCCAGA
361 AUGACAAUAG AGAACGGCAG GAGCACAACG ACAGACGGAG CCUUGGCCAC CCUGAGCCAU
421 ...
```

Figure 3.8: Categorization of true positive and true negative start sites. Suppose that a ribosome profiling experiment detected the following start sites for a given mRNA sequence: CUG at position –78 and CUG at position –120 (blue colored codons). These start sites were then assumed to be true positive start sites. In consequence, all near–cognate start sites not listed in the ribosome profiling dataset and upstream of the most downstream reported true start site were assumed to be true negatives (dark red colored codons). The light red colored codons are start sites not considered as false starts in the analyses since they are located downstream of the most downstream reported true start site. Note that the grey colored downstream part depicts the annotated CDS sequence whereas the italic (purple) upstream part marks the –99 upstream window needed to calculate some of the features (see below). All marked start sites (true positive and true negative) exhibit a surrounding window of ± 99 nucleotides as well as a downstream in–frame stop codon. In total, this mRNA sequence would provide 2 true start sites and 9 false start sites out of 23 putative starts.

Data integration: efficient data storage for fast access

Data that was retrieved from the mentioned sources was processed and then either stored as SQLite database [83], pandas dataframe [82], or Python dictionary. In the following, data integration of the *PreTIS* framework is presented. Table 3.3 illustrates the parsed and efficiently stored Ensemble gene and transcript data as fast accessible SQLite database.

For convenience, human and mouse 5′ UTR, CDS, and 3′ UTR sequences were organized as Python dictionaries and saved using the Python pickle module that allows to efficiently store objects by serialization. Experimentally found alternative start sites are based on the three mentioned datasets. The retrieved mRNA sequence, gene/transcript IDs, and verified alternative start site datasets were processed and stored as one unified database. The construction of this database with the example of gene *DHX9* is illustrated in Table 3.4. All information is stored separately for human and mouse datasets.

Table 3.3: SQLite database with Ensembl gene and transcript information. Retrieved gene and transcript data was processed and stored as a SQLite database to provide a structured data overview and fast access. Shown are the SQLite database column names with entries of gene *DHX9* as example.

Column name	Example
RefSeq_mRNA	NM_001357
RefSeq_protein	NP_001348
Ensembl_Gene	ENSG00000135829
Ensembl_Transcript	ENST00000367549
Ensembl_Protein	ENSP00000356520
Gene_Name	DHX9
Description	DEAH (Asp–Glu–Ala–His) box helicase 9

Table 3.4: SQLite database with sequence and TIS information. This SQLite database stores verified alternative translation initiation sites together with the necessary mRNA sequence information. The extension is given in nucleotides (nt). The respective experimentally found alternative start site is underlined in the sequence. In this example, CDS and 3′ UTR are displayed as shortened sequences. Another experimentally verified CTG–initiated translation start of *DHX9* can be found at position –25.

	Column name	Example
1	**RefSeq**	NM_001357
2	**Ensembl_Gene**	ENSG00000135829
3	**Ensembl_Trans**	ENST00000367549
4	**Gene_name**	DHX9
7	**nt_ext**	–46
5	**Codon_ext**	–16
8	**Start_codon**	CTG
6	**Reading_frame**	2
9	**UTR5**	GCGAGTTGCTGTGCGTTTCTCTGTTGTCTCGGTAGAAGGCCAGAGTCACACACGG TCCTAAGAGCTGGGCACCAGGAAGCGAAGGCTGATCTGAAGAAGACACTTGAATC
10	**CDS**	ATGGGTGACG...
11	**UTR3**	AACTTGGTTA...

The dataset that is organized as outlined in Table 3.4 is then the basis for the search of all possible alternative 5′ UTR translation start sites for every transcript. All putative start sites found by this sequence scan are then stored as shown in Table 3.5. This database then comprises the input datasets for the machine learning approach. Class labels are assigned dependent on the experimental confirmation of a start site. Start sites that were found using ribosome profiling (true starts) are assigned a class label of $y_i = 1$, whereas false starts (as described above) are assigned the label $y_i = 0$. This database was generated for each of the three experimental ribosome profiling datasets. As an example, the 5′ UTR of *DHX9* harbors 21 possible TIS with two of them experimentally verified by [43], compare with Table 3.5.

Table 3.5: SQLite database storing all possible alternative 5′ UTR start sites. The database is represented using the example of gene *DHX9*. Considering the 5′ UTR position and the stop position reveals the structure of the ORF, such as uORF or N–terminal extended ORF. Note that the combination of the RefSeq ID and extension is unique. Experimentally confirmed start sites are shown in bold and are retrieved from the SQLite database presented in Table 3.4. To maintain a triplet reading frame, stop positions are always in–frame with the start site. The given "Class" is later used as class label for the machine learning approach. The extension is given in nucleotides (nt).

	RefSeq	Gene_name	nt_ext	Class	Codon	ORF_len	Stop_position
1	NM_001357	DHX9	–3	0	ATC	3813	3810
...	...	...	...	...	...	...	...
6	NM_001357	DHX9	–22	0	ATC	3	–19
7	**NM_001357**	**DHX9**	**–25**	**1**	**CTG**	**6**	**–19**
8	NM_001357	DHX9	–28	0	AGG	9	–19
...	...	...	...	...	...	...	...
12	**NM_001357**	**DHX9**	**–46**	1	**CTG**	**27**	**–19**
...	...	...	...	...	...	...	...
21	NM_001357	DHX9	–105	0	TTG	27	–78

Next, a BLAST search [67] to find human and mouse orthologous mRNA sequences was executed. This was necessary for the calculation of sequence conservation between these two species. Thereby, the SQLite database as shown in Table 3.4 served as data basis for the BLAST approach. First, a BLAST database, which is generated from human and mouse reference mRNA sequences, is required and created using the

```
makeblastdb -in human.fasta -dbtype nucl
```

command with the human mRNA sequences "`human.fasta`" saved in FASTA format and `nucl` for nucleotide. The procedure is repeated for the mouse FASTA formatted mRNA sequences. Appropriate orthologous mouse sequences are then searched for every human mRNA using the

```
blastn -db mouse.fasta -query human.fasta -num_alignments 0 -out blast_out
```

command with the previously prepared human and mouse BLAST databases. A `blast_out` example for *DHX9* gene, having RefSeq ID NM_001357, as query is returned by `blastn` as follows:

```
Query= NM_001357|DHX9|cds_0_based:110-3922
Length=4240
                                                          Score      E
Sequences producing significant alignments:               (Bits)   Value

NM_007842|Dhx9|cds_0_based:98-4249                         4457     0.0

Lambda      K        H
    1.33    0.621     1.12

Gapped
Lambda      K        H
    1.28    0.460    0.850.
```

In this example, only one orthologous sequence was found. In case there are several hits, the first one is taken as respective ortholog. The BLAST result file is subsequently parsed and the results are stored as SQLite database, see Table 3.6.

Table 3.6: SQLite database with results from the BLAST search. This SQLite database is used to store all human mRNA sequences together with their orthologous mouse sequence that was detected using BLAST. All necessary information, such as the respective gene name and the mRNA sequence, is stored in this database. Thereby, the 5′ UTR, CDS, and 3′ UTR sequences are saved in separate columns.

	Column name	Example
1	**RefSeq_Human**	NM_001357
2	**Gene_name_Human**	DHX9
3	**UTR5_Human**	`GCGAGTTGCTGTGCGTTTCT...`
4	**CDS_Human**	`ATGGGTGACGTTAAAAATTT...TAA`
5	**UTR3_Human**	`AACTTGGTTATGTCAGTTCC...`
6	**RefSeq_Mouse**	NM_007842
7	**Gene_name_ Mouse**	Dhx9
8	**UTR5_Mouse**	`GCCGTTCTCGTGGAAGGTTG...`
9	**CDS_Mouse**	`ATGGGTGACATTAAAAATTT...TAA`
10	**UTR3_Mouse**	`GACTGGACTCTGTCGGAGCC...`

3.3.2 Features based on mRNA sequence information

All features used here are solely based on information derived from the mRNA sequences. In total, we considered 1,252 features, with three features based on PWMs, 20 biologically–motivated features (e.g. sequence conservation or start site flanking sequence context) and 1,229 features found by a k–mer search for $k = 1$ and $k = 3$. The features are explained in the following.

Position weight matrix

In mammalian cells, some codons (e.g. AUG and CUG) are more frequently used to initiate translation compared to other codons (e.g. AUA or AGG) [43, 44]. In the HEK293 dataset used here, 26.1% of the reported upstream initiation start codons are AUGs and 29.8% are CUGs [43]. The start codon information was considered by using position weight matrices (PWMs). To account for the important role of the flanking sequence context for translation initiation, we considered a window ranging from –15 to +10 with respect to the start site in a set of sequences S. First we calculated from the data in the training set a position frequency matrix (PFM) with the nucleotides $nt \in \{\text{A, C, U, G}\}$ as the rows and the sequence position i as the columns. The matrix entries were filled by dividing the sum of occurrences of a nucleotide at position i by the total number of sequences contained in S. The PWM was then calculated by dividing each entry in the PFM by the respective nucleotide background frequency and taking the natural logarithm, that means

$$\text{PWM}_{(nt,i)} = log\left(\frac{\text{PFM}_{(nt,i)}}{bg_{nt}}\right)$$

where the background frequency bg_{nt} is defined as the actual nucleotide frequency of the 5′ UTR in S. We calculated three PWMs, one based on the true start sites ($\text{PWM}_{positive}$), one based on the false start sites ($\text{PWM}_{negative}$), and one based on the log–ratio between true and false start sites (PWM_{ratio}) in the training set. The PWM_{score} for a sequence s was then computed as

$$\text{PWM}_{score}(s) = \sum_{i=0}^{len(s)} \text{PWM}_{nt_i,i}$$

where nt_i is the nucleotide occurring at position i in sequence s. With $\text{PWM}_{positive}$, a PWM_{score} greater than zero indicates that the given sequence s is more likely a true start than a false start while a PWM_{score} less than zero suggests a higher probability of being a false start site.

Sequence conservation

To calculate the conservation of a putative start site, sequence alignments between pairs of human and mouse sequences (5′ UTR and CDS), found by applying `blastn`, were generated using MUSCLE [70]. For this, 5′ UTR and CDS were translated into all three possible reading frames and were aligned accordingly. We then translated the protein alignment back into the associated (gap–free) nucleotide alignment, compare with [43]. A human start site was assumed to be conserved if it shares the same codon or amino acid with the murine ortholog at the respective position. This yielded two binary features: codon and amino acid conservation. We also calculated the average degree of 5′ UTR sequence conservation, using the human–mouse mRNA sequence alignment. For this we divided the number of matching nucleotides by the length of the 5′ UTR sequence. Gaps were ignored.

Start codon flanking sequence context

The flanking sequence context was assessed in two ways where we considered either only the positions –3R (R = purine) and +4G, which were determined to be crucial for initiation [182, 183], or experimentally determined translational start codon efficiencies [189]. In the first approach, the Kozak sequence context was discretized into strong (A or G at –3 and G at +4), intermediate (A or G at –3 and no G at +4), weak (no A and no G at –3 and G at +4) and no Kozak context. These categories were presented as the values 1 (no), 2 (weak), 3 (intermediate) and 4 (strong). In the second approach, we used the raw translational efficiency values reported by Noderer et al. [189] as feature for the respective flanking sequence context of a start site. These authors investigated the translational efficiency of all possible 11 nt long (position –6 to +5) flanking sequence contexts around the AUG translation start using high–throughput sequencing combined with fluorescence signaling. We assumed that alternative starts behave similarly as AUG codons and therefore use the same translational efficiency values for the alternative starts.

Minimum free energy of mRNA secondary structure

Secondary structure is an important factor for translation initiation [183, 187, 188]. Dependent on the propensity of the mRNA secondary structure downstream of a putative start codon, the ribosome scanning in downstream direction can pause and translation is initiated [188]. It was shown that a secondary structure with a minimum free energy of $\Delta G = -19 \frac{kcal}{mol}$ that starts 12–15 nt downstream of the translation start site can prevent leaky scanning and compensate for an unfavorable flanking sequence context [188, 190]. A secondary structure starting 14 nt from the translation initiation site was observed to have the largest effect [188]. Here, we considered different windows for calculating the minimum free energy of the secondary structure and then selected the most suitable one to differentiate between true and false start sites: a 60

nt window starting at position +14, a 60 nt window starting at position +20, a window from position –10 to +50 and a window from –50 to +50. Minimum free energies were calculated using RNAfold [68].

GC–content

It was shown that the GC–content continuously decreases from 5′ UTR across the CDS to the 3′ end in human [223]. We therefore analyzed whether the GC–content differs between true and false start sites using the same windows as for the minimum free energy (see above). Note that the minimum free energy of an mRNA secondary structure and its GC–content are related to each other since G–C pairs possess a higher degree of stability than A–U pairs due to their additional hydrogen bond [224].

Open reading frame length

It appears plausible that the length of open reading frames that code for functional proteins is generally longer than the ones resulting from arbitrary start sites in the mRNA sequence. Therefore, we also considered the length of the putative open reading frame.

5′ UTR nucleotide distribution

As mentioned before, the GC–content varies between 5′ UTR and CDS. If a part of the annotated 5′ UTR is actually used as CDS this may result in a different nucleotide composition compared to the actual 5′ UTR. Therefore, we calculated the percentage of all four nucleotides (e.g. $\frac{\#A}{5'\ UTR\ length}$) in the entire 5′ UTR. This resulted in four additional features.

5′ UTR length

We also tested the 5′ UTR length with respect to significant differences between true and false start sites by defining the 5′ UTR (nucleotide) length as further feature.

K–mer search

We counted the frequency of all possible k–mers of length $k = 1$ (position–specific k–mers) and $k = 3$ (codon and respective amino acid k–mers) in a window from –99 to +99 around the start site. k–mers were defined as all possible combinations of subsequences of length k, given an alphabet, here nucleotides $\{A, C, U, G\}$. We considered in–frame and out–of–frame k–mers as well as k–mers upstream and/or downstream of the start site as suggested in [225]. In total, this yielded 1,229 k–mers: position–specific k–mers in the predefined window of ± 99 amount to 198 positions $\times$ 4 nucleotides = 792 (e.g. "K–mer: position –12 is C"), 64 codons $\times$ 5 (counted in the complete ± 99 region, the upstream region, the downstream region as well as in–frame–downstream and in–frame–upstream) = 320, 20 amino acids $\times$ 5 = 100, 1 stop codon $\times$ 5 = 5 and 4 nucleotides ($k = 1$) $\times$ 3 (complete ± 99 region, upstream region and downstream region) = 12. This sums up to $792 + 320 + 100 + 5 + 12 = 1,229$ k–mers.

3.3.3 Regression approach

The prediction approach, shown in Figure 3.9, was applied to the human HEK293 [43] and mouse ES datasets [44]. The implementation was done in Python (version 2.7) and using the scikit–learn package (version 0.17) for the machine learning part [80]. First, as mentioned, all putative start sites in the 5′ UTR were defined as true positives or true negatives based on the

reported ribosome profiling data and their location in the mRNA sequence. We then balanced the size of the dataset so that it contains the same number of true and false start sites by randomly under–sampling from the larger dataset. We repeated the data balancing as well as the assignment of random training and test set 10 times to evaluate the model robustness and reported the average performance.

We applied Wilcoxon rank–sum test and Bonferroni correction (with a significance threshold of $p = \frac{0.01}{1{,}252} = 8 \times 10^{-6}$, with the total number of features as the denominator) to test for the statistical significance of the biological, the k–mer, and the PWM features to differentiate between true and false start sites. We subsequently calculated all pairwise Pearson correlations between the significant biological and PWM features as well as for the 50 most significant k–mer features and only used uncorrelated ($|r| < 0.7$) features in the training step. If two or more features were correlated, the one with the smallest p–value was used. The PWMs were calculated in each training step iteration to guarantee that the test set is independent on the calculated PWMs. All features were normalized (mean zero and unit variance) to ensure comparability. Next, several learning models were established and evaluated, see Figure 3.9. The *PreTIS* prediction model approach is outlined as Algorithm 3.1.

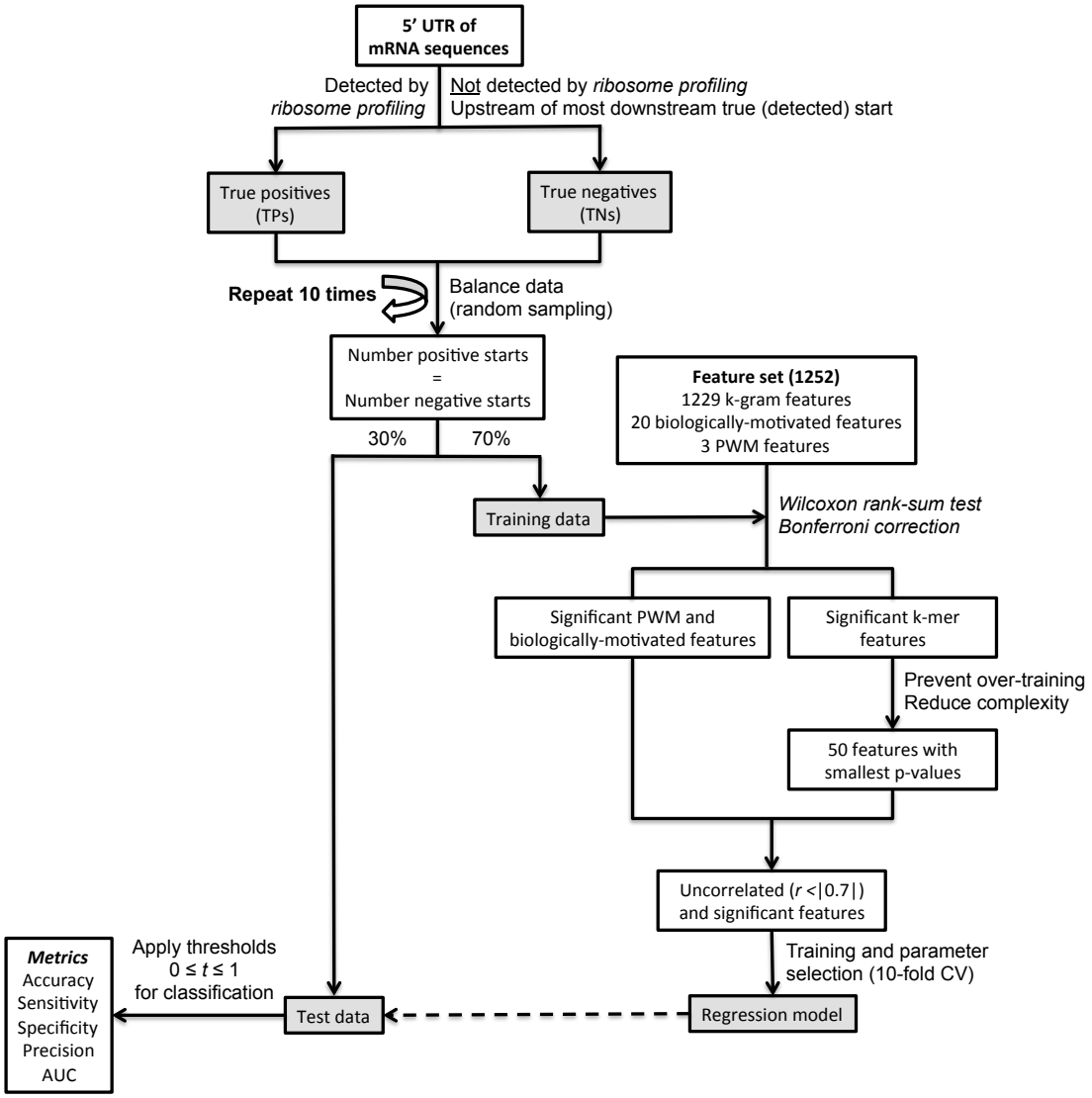

Figure 3.9: Flowchart of the *PreTIS* regression approach. Data balancing was repeated ten times to investigate model robustness. Significant features were identified by the Wilcoxon rank–sum test [46].

Algorithm 3.1 PredictionModel(**$databases$).

1: Init true start sites: ids_pos
2: Init false start sites: ids_neg
3: **for all** run from 1 to 10 **do**
4: Init true and randomly sampled false starts: ids
5: Init class labels of ids: $labels$
6: $train\_ids$, $test\_ids$, $ytrain$, $ytest$ ← ***train_test_split***(ids, $labels$, test_size=0.3)
7: $Xtrain$, $Xtest$ ← **make_sets**($train\_ids$, $test\_ids$) {normalized, significant, and uncorrelated features}
8: **for all** $predictor$ in ["SVR","LinearRegression"] **do**
9: **if** $predictor$ == "SVR" **then**
10: Init kernel parameters: $params$
11: $classifier$ ← ***GridSearchCV***(***SVR***(), $params$, cv=10, scoring=MSE)
12: **end if**
13: **if** $predictor$ == "LinearRegression" **then**
14: $classifier$ ← ***LinearRegression***()
15: **end if**
16: $classifier$.***fit***($Xtrain$, $ytrain$)
17: **for all** $threshold$ from 0.0 to 1.0 **do**
18: **for all** $test\_point$ presented by ($Xtest$, $ytest$) **do**
19: Init feature set of $test\_point$: $Xtest\_point$
20: Init class label of $test\_point$: $ytrue$
21: $ypred\_reg$ ← $classifier$.***predict***($Xtest\_point$)
22: **if** $ypred\_reg >= threshold$ **then**
23: $ypred$ ← 1
24: **else**
25: $ypred$ ← 0
26: **end if**
27: tp, fp, tn, fn ← **compare**($ytrue$, $ypred$)
28: **end for**
29: $spec$, sen, acc, $prec$ ← **evaluate**(tp, fp, tn, fn)
30: fpr, tpr, _ ← ***roc_curve***($ytest$, all $ypred$)
31: roc_auc ← ***auc***(fpr, tpr)
32: **end for**
33: **end for**
34: **end for**
35: **return** {Results for every run, $predictor$, and $threshold$ are stored.}

Functions used in Algorithm 3.1 that are written in bold and italic were provided by the scikit–learn library [80]. As input, several pre–computed databases that store information on experimentally verified (true) and not verified (false) start sites (AUG and near–cognate), were given as arguments. As mentioned, false start sites must also meet the criteria to be located upstream of most downstream true start site, due to the assumption of a linear scanning model (Line 2). This false start site set is further reduced by random sampling such that it is composed of the same number of sites as the true start sites set to omit a class size dependent bias (Line 4). Afterwards, the dataset was divided randomly into training (70%) and test (30%) set using the `train_test_split(`ids`, `$labels$`, test_size=0.3)` function provided by the scikit–

learn library (Line 6). Training and test datasets composed of normalized, significant, and uncorrelated features (described above, compare with Figure 3.9) were subsequently used for the learning procedure (Line 7).

Next, we generated simple linear as well as SVM regression models on 70% of this data and tested them on the remaining 30% of the data, using three different kernels for the SVM approach: linear, RBF and polynomial. We applied 10–fold cross–validation to find the best penalty parameter $C \in \{0.1, 1, 10, 100\}$ and ϵ–tube parameter $\in \{0.01, 0.1, 1, 10\}$ for the training dataset when applying SVR models. The remainder of parameters were kept at default values. This parameter selection was supported by the scikit–learn library (Line 12 and see below).

As mentioned, ϵ–support vector regression `SVR()` requires the selection of optimal parameters `C` and ϵ, which is optimized using the `GridSearchCV(estimator, param_grid, cv, scoring)` function (Line 12 of Algorithm 3.1) with the estimator `SVR()`, the `param_grid` specifying the parameter sets for C, ϵ, and the kernel as well as the cross–validation splitting parameter $cv = 10$ and the test set scoring function MSE for "mean squared error". We compared different kernels that can be specified via parameters "poly", "linear", and "rbf" for polynomial, linear, and radial basis function kernel, respectively. We also established an ordinary least squares linear regression using the scikit–learn package `LinearRegression()` function (Line 15). The estimator, `SVR()` or `LinearRegression()`, is then fitted to the given dataset taking the different parameter sets into consideration (only for `SVR()`) and using the scikit–learn `fit(Xtrain, ytrain)` function (Line 17). The resulting model with best performing parameters is subsequently used to `predict(Xtest_point)` unseen data given an appropriate feature set `Xtest_point` (Line 23).

The model is evaluated using specificity (`spec`), sensitivity (`sen`), accuracy (`acc`), and precision (`prec`) that were derived from the number of true positives (`tp`), false positives (`fp`), true negatives (`tn`), and false negatives (`fn`) points given a threshold (Lines 24–31, compare with Figure 3.9). The threshold is explained below. The area under the curve AUC was also calculated using methods provided by the scikit–learn package (Lines 32 and 33).

Since we applied a regression approach, we applied 100 classification thresholds $0.0 \leq t \leq 1.0$ in steps of 0.01 to the predicted output $ypred\_reg$ in order to classify every start site as true or false based on its model outcome and the given threshold t, see Figure 3.9 and Lines 19–34 of Algorithm 3.1. These thresholds can be interpreted as initiation confidences where start sites with a regression value $ypred\_reg \geq t$ are predicted as true start sites and the ones with $ypred\_reg < t$ as false start sites. If a start site is predicted with an initiation confidence > 1, we substituted this value by one. The same holds for start sites with a predicted confidence < 0, which were substituted with zero. We then compared the predicted class with the correct class and used the mentioned metrics for model assessment. The final model for the prediction of new mRNA sequences and for a SNP analysis was subsequently determined by comparing different model performances.

3.3.4 *In silico* SNP analysis

To investigate the effect of putative SNPs within the flanking sequence context of the start sites (position –15 to +10), we *in silico* substituted one nucleotide position at a time by all 3 remaining nucleotides, yielding 75 different contexts (the start codon itself was not mutated). We then recalculated the needed sequence features to investigate the mutational impact and subsequently applied our final prediction model to all contexts. We then report the effect of these substitutions on the predicted initiation probabilities.

3.4 Results

In this work we used ribosome profiling data from HEK293 cells [43] and mouse ES cells [44] to analyze sequence encoded differences between true and false translation initiation sites located in the mRNA 5′ UTR. A third dataset, only containing AUG starts, was used as validation

set [54]. Calculated sequence–based features were subsequently used to build a prediction model. In the following, we present the generated true and false datasets, the results of the regression approach, its application and an implementation as web service *PreTIS*.

3.4.1 Filtered dataset

The start sites reported by [43, 44, 54], based on ribosome profiling data, were filtered to include only starts matching AUG and near–cognate codons in the 5′ UTR. For HEK293 cells [43], this yielded 4,482 true start sites (i.e. reported in the experimental analysis) and 49,520 false start sites in 3,566 mRNAs. For mouse ES cells [44], this gave 3,009 true start sites and 19,864 false start sites in 1,632 mRNAs. True (reported) starts were assumed to be true positives and false (not reported and upstream of the most downstream reported) starts were assumed to be true negatives. For comparison, we also included a smaller dataset of Calviello et al. [54] who only determined AUG starts in HEK293 cells. Table 3.7 displays the three datasets. All reported analyses are based on these filtered datasets. Among the considered AUG and near–cognate start codons, AUG (human: 26%, mouse: 16%), CUG (human: 30%, mouse: 34%) and GUG (human: 13%, mouse: 19%) were the most prevalent translation start codons. Thus, CUG and GUG are more often used in mouse compared to human. This is in accordance with [43, 44] and shows that the start codon itself is very important for translation initiation.

Table 3.7: Datasets used in the *PreTIS* study. Three different datasets were used in this study to establish a human and mouse prediction model and to cross–validate the regression models. The numbers indicate the filtered start sites used in the prediction approach.

Cell line	mRNAs	Start codons	TPs	TNs	Used for	Source
HEK293	3,566	AUG and near–cognate	4,482	49,520	Human prediction model	[43]
HEK293	391	AUG	332	447	Validation set	[54]
Mouse ES	1,632	AUG and near–cognate	3,009	19,864	Mouse prediction model	[44]

HEK293 = Human embryonic kidney cells; Mouse ES = Mouse embryonic stem cells

3.4.2 Regression models predict initiation confidences

Table 3.7 illustrates that the negative sets outnumbered the positive sets by factors of 7 (mouse ES) and 11 (HEK293). To avoid a class size dependent bias, we randomly under–sampled the same number as true positive start sites from the true negative set. Next, we trained on 70% and tested on 30% of the data (randomly assigned). Table 3.8 lists the performance of human and mouse models together with the optimal thresholds t. All human models performed very similarly with accuracies of about 80%, while the average performance of the mouse model was lower with average accuracies of about 76%, see Table 3.8. We also computed ROC curves and the associated AUC. In accordance with the other metrics, also the AUC values were satisfactory with average values of about 80% and 76% for the human and mouse models, respectively.

Best performing prediction model

Since all models gave a very similar performance with accuracies of about 80%, we decided to choose the simple linear regression model that can be interpreted well. The best performing human linear regression model, with balanced performance metrics, was obtained in run 2

Table 3.8: Evaluation of the *PreTIS* regression approach. The prediction was repeated 10 times to evaluate the model robustness. Shown are the average performance measures.

	Accuracy	Specificity	Sensitivity	Precision	AUC	Thres. t
			HEK293			
Linear SVR	0.80±0.01	0.80±0.01	0.81±0.01	0.80±0.01	0.80±0.01	0.62±0.01
RBF SVR	0.82±0.01	0.81±0.01	0.83±0.02	0.82±0.01	0.82±0.01	0.55±0.02
Polynomial SVR	0.80±0.01	0.80±0.01	0.81±0.02	0.80±0.01	0.80±0.01	0.59±0.02
Linear Regression	0.80±0.01	0.80±0.01	0.81±0.01	0.80±0.01	0.80±0.01	0.55±0.01
			Mouse ES			
Linear SVR	0.75±0.01	0.75±0.01	0.76±0.01	0.75±0.01	0.76±0.01	0.65±0.03
RBF SVR	0.76±0.01	0.76±0.01	0.76±0.02	0.76±0.01	0.76±0.01	0.58±0.03
Polynomial SVR	0.75±0.02	0.75±0.01	0.76±0.02	0.75±0.02	0.75±0.02	0.62±0.03
Linear Regression	0.76±0.01	0.75±0.01	0.76±0.01	0.75±0.01	0.76±0.01	0.55±0.01

(the prediction was repeated 10 times). This model had an accuracy of 83%, a sensitivity of 84%, a specificity of 82% and a precision of 83% on the test data. It was then applied to predict unknown start sites of a gene of interest and to conduct an *in silico* mutation analysis. Moreover, it is embedded in the *PreTIS* web service. Therefore, this model is analyzed in more detail in the following. Figure 3.10 displays the predicted codon distribution when applying the best performing linear regression model of run 2 to the mRNA sequences in the test set and using the threshold $t = 0.54$ that gave the best overall performance.

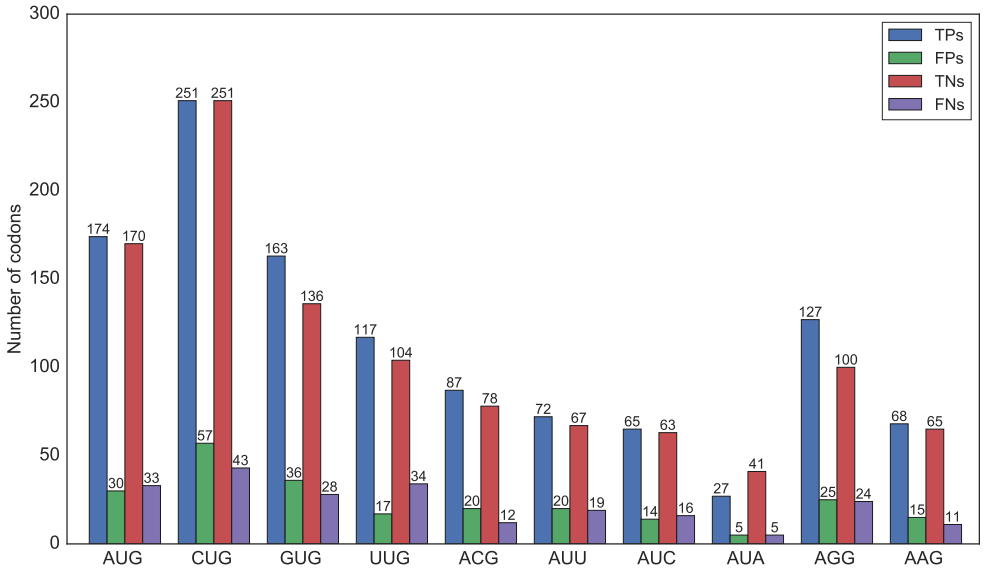

Figure 3.10: Codon distribution of test samples using the best performing human model. AUG, CUG and GUG were the most prevalent TP start sites in the test samples with $t = 0.54$.

The distribution of predicted codons agreed with the preferences found experimentally [43, 44]: AUG and CUG were the most prevalent start codons, whereas AUA or AAG were more often classified as true negatives. Nevertheless, our predictor also detects true negative AUGs and CUGs and true positive AUAs and AAGs.

The features that were used to build this prediction model are displayed in Table 3.9. The most significant feature is the length of the 5′ UTR ($p < 10^{-310}$). The 5′ UTR was found to be shorter on average for true starts sites (414±270 nt) compared to false start sites (675±545 nt). The second most significant biologically–motivated feature with a p–value of $p = 8.2 \times 10^{-190}$ was the conservation of the 5′ UTR. The values of 0.4±0.16 for the true start sites and 0.33±0.16 for false start sites suggest that 5′ UTRs harboring true start sites are in general more conserved. Another highly significant feature ($p = 5.1 \times 10^{-144}$) was the number of upstream AUGs. Considered false start sites had more upstream AUGs (0.59±0.9) than considered true start sites (0.22±0.57), see Table 3.9. This can be explained as follows: if AUG is located upstream of another putative start site, the linear scanning model of Kozak [36] implies that it is more probable that the AUG is used as start site instead.

$\text{PWM}_{positive}$ was also found to be highly significant ($p = 5.5 \times 10^{-173}$). The PWMs were recalculated for each training sample to achieve unbiased test samples in every run. The background frequencies of the best performing run 2 amounted to bg_A:0.16, bg_C:0.29, bg_U:0.21, and bg_G:0.34, while the average background frequencies of all training and test runs were calculated as bg_A:0.21±0.06, bg_C:0.27±0.06, bg_U:0.22±0.06, and bg_G:0.3±0.06. Thus, as expected [223], guanine and cytosine were prevalent in the 5′ UTR. Figure 3.11 shows the PWM scores calculated for the test samples in the run with best overall performance (run 2) based on the PWM generated using the true training samples ($\text{PWM}_{positive}$) in this run. The scores of the true (test) start sites were significantly higher (2.75 ± 1.5) than those of the false (test) start sites (-0.14 ± 2.82).

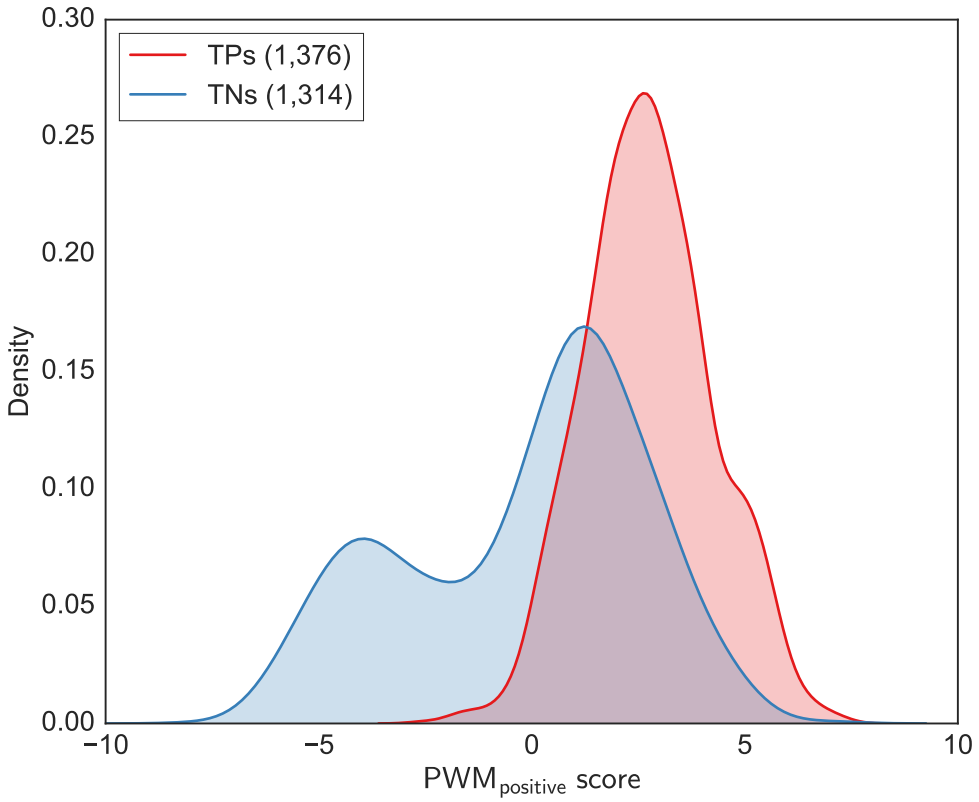

Figure 3.11: Frequency distribution of $\text{PWM}_{positive}$ scores. Shown are the results for all test samples of the best performing run 2. The PWM was established using the true start sites in the training data of run 2. The difference between TPs and TNs was found to be highly significant ($p = 5.5 \times 10^{-173}$, Wilcoxon rank–sum test).

Interestingly, the distribution of the false start sites was found to be bimodal. Thus, one might speculate that some of these considered false start sites with higher PWM values (i.e. start sites not found by the ribosome profiling technique and located upstream of the most

Table 3.9: Features of the best human *PreTIS* prediction model. Mean value and standard deviation of the 44 features that were used in the best human model (biologically–motivated and PWM features are shown in bold). All 4,482 true and 49,520 false start sites were considered for this analysis. All listed features showed significant differences between true and false start sites (p–values $< 1.6 \times 10^{-8}$). Note that due to numerical reasons, very small p–values ($< 10^{-310}$) are represented as 0.0 in Python programming language (scipy version 0.17.0). The PWM scores are based on the test data (compare to Figure 3.11).

	Feature	True starts	False starts	p–value
1.	**5′ UTR length**	414.41±270.48	675.41±545.35	$< 10^{-310}$
2.	**5′ UTR conservation**	0.4±0.16	0.33±0.16	8.2×10^{-190}
3.	**PWM positive**	2.75±1.5	-0.14±2.82	5.5×10^{-173}
4.	K–mer: upstream AUG	0.22±0.57	0.59±0.9	5.1×10^{-144}
5.	**5′ UTR: percentage A**	0.18±0.05	0.2±0.05	9.6×10^{-100}
6.	**Kozak sequence context**	2.67±1.07	2.3±1.11	9.2×10^{-95}
7.	**Translational efficiency of flanking sequence**	83.75±20.11	77.12±21.4	1.1×10^{-83}
8.	K–mer: position -12 is C	0.13±0.34	0.3±0.46	2.7×10^{-77}
9.	K–mer: upstream Asparagine	1.25±1.37	1.61±1.61	4.0×10^{-43}
10.	K–mer: downstream AUG	1.14±1.15	0.92±1.1	9.2×10^{-41}
11.	K–mer: upstream A	17.24±7.43	18.81±7.89	4.0×10^{-40}
12.	K–mer: in-frame upstream Alanine	3.69±2.6	3.16±2.29	4.0×10^{-37}
13.	K–mer: upstream Alanine	10.27±4.5	9.38±4.6	6.2×10^{-37}
14.	**5′ UTR: percentage G**	0.32±0.06	0.31±0.05	7.1×10^{-37}
15.	**Codon conservation**	0.23±0.42	0.12±0.32	3.2×10^{-36}
16.	K–mer: position –3 is A	0.31±0.46	0.2±0.4	3.4×10^{-35}
17.	K–mer: upstream CCG	2.98±2.43	2.56±2.31	7.1×10^{-34}
18.	K–mer: downstream CCA	2.04±1.54	1.75±1.45	1.1×10^{-32}
19.	K–mer: position –12 is A	0.3±0.46	0.19±0.4	4.0×10^{-32}
20.	K–mer: in–frame upstream Methionine	0.07±0.29	0.2±0.48	3.3×10^{-31}
21.	K–mer: upstream Arginine	12.15±4.34	11.33±4.64	1.5×10^{-29}
22.	K–mer: upstream Histidine	1.7±1.52	1.97±1.65	2.2×10^{-27}
23.	K–mer: GCC	6.4±3.87	5.77±3.75	1.1×10^{-25}
24.	K–mer: position 4 is G	0.37±0.48	0.28±0.45	2.3×10^{-25}
25.	K–mer: upstream Threonine	3.56±2.08	3.91±2.19	4.9×10^{-25}
26.	K–mer: upstream CGG	3.14±2.51	2.77±2.41	3.2×10^{-24}
27.	K–mer: upstream C	30.4±8.98	28.96±9.04	1.0×10^{-23}
28.	K–mer: position –2 is G	0.23±0.42	0.32±0.47	1.2×10^{-23}
29.	K–mer: upstream Stop	2.3±1.71	2.66±2.0	1.4×10^{-23}
30.	K–mer: UAG	1.34±1.2	1.57±1.35	5.6×10^{-23}
31.	K–mer: upstream CAU	0.58±0.85	0.73±0.95	3.4×10^{-22}
32.	K–mer: upstream Serine	9.44±3.29	8.93±3.14	5.7×10^{-22}
33.	K–mer: downstream Glutamine	3.57±2.01	3.26±1.88	2.4×10^{-21}
34.	K–mer: AGG	4.29±2.51	4.7±2.69	2.1×10^{-20}
35.	K–mer: AGC	4.4±2.43	4.02±2.19	2.1×10^{-20}
36.	K–mer: downstream ACC	1.45±1.26	1.27±1.17	2.0×10^{-19}
37.	K–mer: UAA	1.22±1.42	1.51±1.76	6.2×10^{-19}
38.	K–mer: downstream Proline	9.3±5.63	8.56±5.47	3.5×10^{-18}
39.	K–mer: upstream CAA	0.75±0.92	0.91±1.06	1.3×10^{-17}
40.	K–mer: in–frame upstream Histidine	0.54±0.77	0.67±0.87	1.7×10^{-17}
41.	K–mer: upstream GAU	0.63±0.85	0.77±0.96	2.1×10^{-16}
42.	K–mer: in–frame upstream GCC	1.21±1.4	1.02±1.22	6.7×10^{-16}
43.	K–mer: in–frame upstream GCG	1.14±1.42	0.97±1.27	6.2×10^{-14}
44.	**PWM negative**	1.94±1.34	1.59±1.09	1.6×10^{-08}

downstream reported true start, which are therefore considered as true negative starts, see also Figure 3.8) might be used as actual start sites in different cell types or cellular conditions. This also explains the overlap between the true positive and true negative start sites in Figure 3.11.

Another biologically important feature that also represents the flanking sequence context is the "Kozak sequence context" feature (see methods) with a p–value of 9.2×10^{-95}. As expected from experimental findings [36], true start codons more often exhibit a strong or intermediate Kozak context compared to false start sites that often show no Kozak context at all, see Table 3.9 and the supplementary material of our publication [226]. This is also in agreement with the observation that A at position –3 ($p = 3.4 \times 10^{-35}$) and G at position +4 ($p = 2.3 \times 10^{-25}$) were found (by the k–mer search) to be important for translation initiation. Similarly, the translational efficiency of the flanking sequence context, experimentally investigated in [189], was also highly significant ($p = 1.1 \times 10^{-83}$). The average efficiency of true start sites is, as expected, higher than the one calculated for the false start sites. Moreover, true start codons were found to be more often conserved between human and mouse sequences compared to false start sites ($p = 3.2 \times 10^{-36}$). Start site conservation was also mentioned in the original publication of the HEK293 dataset we used here [43].

Many significant features detected by the k–mer search contained upstream G–C patterns (e.g. "K–mer: upstream CCG" or "K–mer: upstream CGG") at higher frequencies for true start sites compared to false start sites. This reflects the generally higher GC–content in the 5′ UTR compared to the CDS and is in accordance with the finding that the GC–content decreases from the 5′ UTR to the CDS [223].

Consistent with the p–values of the features used in the best performing human linear regression model are the feature (weight) coefficients determined by the model training step. The respective figure is shown in the supplementary material of our publication [226]. The highest coefficients were assigned to the $\text{PWM}_{positive}$ and the number of upstream AUGs ("K–mer: upstream AUG'").

3.4.3 Transferability of the prediction model

To investigate the transferability of our best human prediction model, we analyzed its performance using the mouse ES data as well as the HEK293–AUG dataset, see Table 3.10. With the threshold of $t = 0.54$ that was found to be optimal for the trained HEK293 dataset, we obtained for the mouse ES dataset an accuracy of 76%, a sensitivity of 72% and a specificity of 77%. By scanning all possible thresholds, we found that $t = 0.52$ yields a more balanced performance of 75%, 76% and 74% for accuracy, sensitivity and specificity, respectively. Decreasing the threshold seems to be advantageous for the mouse dataset, since some true positives seem to possess weaker features for translation initiation (e.g. a weak flanking sequence context or a less common initiation codon), but are nevertheless true positive starts.

We then applied our best regression model to the start sites reported in the HEK293–AUG dataset that only contains AUG starts [54]. The categorization of true positive and true negative start sites was conducted as above for the HEK293 dataset (see Figure 3.8), with the only difference that the HEK293–AUG dataset only contains AUG start sites instead of AUG and all near–cognate codons. Thus, we defined again the false start sites as all AUG starts located in the 5′ UTR that were not detected by ribosome profiling and are located upstream of the most downstream true start site.

Differentiating only between true and false AUG start sites is particularly difficult because the AUG itself is a very strong signal for a true start site and just by random chance there might be AUGs with, for example, good flanking sequence, which are not used as translation start sites (or are not reported in the dataset). Moreover, our prediction model was trained on all possible cognate codons instead of AUG alone. Our best model with the determined threshold of $t = 0.54$ detected 77% of the true AUG starts in the HEK293–AUG dataset (sensitivity of 77%). Nevertheless, the specificity of this prediction is only 44% and thus the overall accuracy is only slightly better than a random decision (58%), compare to Table 3.10. However, when increasing the threshold from $t = 0.54$ to $t = 0.65$, we were able to increase the overall accuracy to 63%. A threshold of $t = 0.65$ was found to be optimal for this dataset. More information on

Table 3.10: Application of *PreTIS* to the independent datasets. Shown is the performance of the best human HEK293 model applied to the mouse ES and human HEK293–AUG datasets.

Unbalanced dataset								
	Mouse ES		**Mouse ES**		**HEK293–AUG**		**HEK293–AUG**	
Threshold	t=0.54		t=0.52		t=0.54		t=0.65	
	TP	TN	TP	TN	TP	TN	TP	TN
Pred. pos.	2,161	4,569	2,273	5,072	257	249	207	160
Pred. pos.	848	15,295	736	14,792	75	198	25	287
Total	3,009	19,864	3,009	19,864	332	447	332	447
Accuracy	0.76		0.75		0.58		0.63	
Sensitivity	0.72		0.76		0.77		0.62	
Specificity	0.77		0.74		0.44		0.64	
Precision	0.32		0.31		0.51		0.56	
Balanced dataset								
	Mouse ES		**Mouse ES**		**HEK293–AUG**		**HEK293–AUG**	
Threshold	t=0.54		t=0.52		t=0.54		t=0.64	
	TP	TN	TP	TN	TP	TN	TP	TN
Pred. pos.	2,161	689	2,273	763	257	185	211	125
Pred. pos.	848	2,320	736	2,246	75	147	121	207
Total	3,009	3,009	3,009	3,009	332	332	332	332
Accuracy	0.74		0.75		0.61		0.63	
Sensitivity	0.72		0.76		0.77		0.64	
Specificity	0.77		0.75		0.44		0.62	
Precision	0.76		0.75		0.58		0.63	

the detection of optimal threshold values can be found in the supplementary material of our publication [226]. Problematic was here the precision (i.e. the number of true positives out of all samples classified as positive ($\frac{TPs}{TPs+FPs}$)). Many starts that we assumed to be true negatives actually show properties of true positives and are therefore classified as false positives. Especially if the dataset is highly unbalanced (e.g. the number of mouse ES true starts is only 15% of the false start sites) this effect has a strong influence on the precision. When we balanced our datasets, the precision increased drastically from 31% to 75% for the mouse ES dataset and $t = 0.52$ and from 56% to 63% for the HEK293–AUG dataset and $t = 0.64$, see Table 3.10.

3.4.4 Applications of the prediction model

The established prediction model can, for example, be used to predict translation start sites which are not covered by ribosome profiling experiments or to analyze the impact of mutations in the flanking sequence around the start site.

Prediction of unknown start sites

We applied the final model to a gene of interest, GIMAP5 (ENST00000358647), that was not contained in the human ribosome profiling data. GIMAP5 codes for a GTPase binding GTP and is involved in the survival of T–cells [227]. The scan of *GIMAP5* resulted in 27 candidate start sites with an in–frame stop codon and a surrounding window of ± 99 nt to calculate the k–mer features in. Figure 3.12 shows the predicted initiation probabilities of the putative start sites. Out of these 27 candidate start sites, we found eight start codons with a confidence value above $t = 0.54$. Among these starts, we found one hot candidate (AUG at position –203) with a very high confidence value of 0.92 of being a true start site. Moreover, a CUG at position –36 was also predicted with a high confidence value of 0.81. We postulate that these start sites are able to initiate translation in a specific cell type or cellular condition (for instance cellular stress response). In this manner, the web service *PreTIS* can be used to visualize all putative start sites and subsequently to predict unknown translation start sites.

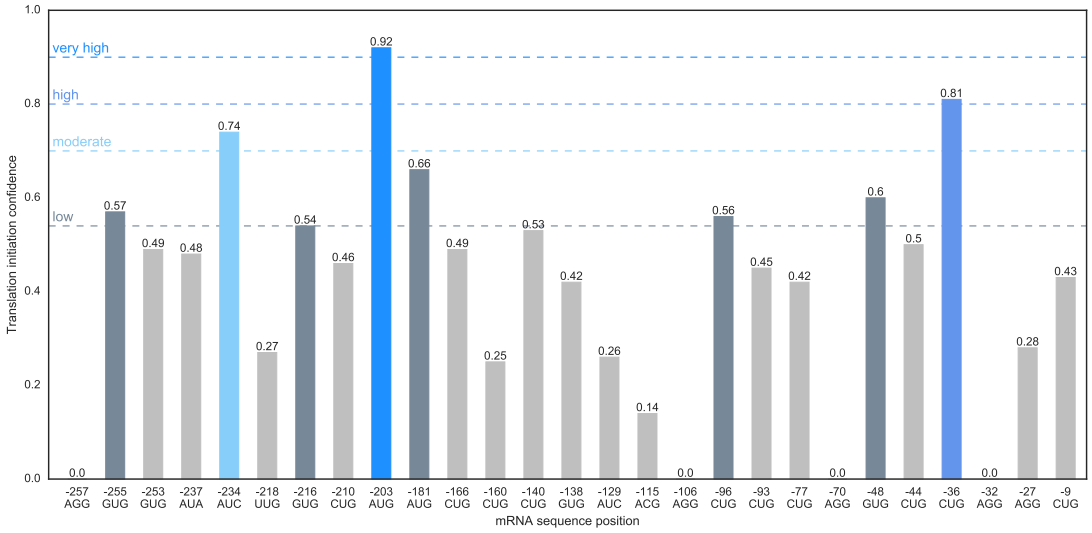

Figure 3.12: Alternative start codons of human gene *GIMAP5*. Predicted start sites were subdivided into four confidence groups and highlighted by different colors and dashed lines: very high (hot/best candidates with $c \geq 0.9$), high ($0.8 \leq c < 0.9$), moderate ($0.7 \leq c < 0.8$) and low ($t = 0.54 \leq c < 0.7$) initiation confidence c. For this gene, we found one hot candidate with a very high confidence value of 0.92 of being a true start site (AUG at position –203).

In silico mutation analysis

As an outlook where this methodology could be helpful as well, we investigated the effect of fictitious SNPs on the translation initiation confidence around all start sites of gene *GIMAP5*. We used the same surrounding window of –15 to +10 that was used to calculate the PWMs. Figure 3.13 shows three possible scenarios how *in silico* mutations in the flanking sequence context of a putative start site affect its predicted initiation confidence. In the first example, the initiation confidence value is, independent of the SNP, always above the threshold. This means that the start site is always predicted as true start site since the overall advantageous properties are not changed severely by a single SNP that is inserted (see Figure 3.13A). In the second case, the predicted initiation confidence value changes dependent on the SNP that is artificially inserted into the flanking sequence (see Figure 3.13BC). Take for instance, CUG at position –44 (Figure 3.13B): a C\G at position +4 increases the initiation confidence from 0.53 (see Figure 3.13) to 0.60. The same holds for a U\A SNP and U\G SNP at position –3 that increase the predicted initiation confidence to 0.66 and 0.57, respectively. For the AUA start site

at position –237 (Figure 3.13C), an U\A SNP and a U\G SNP at position –3 increased the initiation confidence from 0.48 to 0.63 and 0.55, respectively. Positions –3 and +4 were mentioned beforehand to be crucial for translation initiation [182, 183]. Moreover, SNPs at position –12, also found to be significant by the k–mer search (Table 3.9), seem to have an important influence on the translation initiation. A G\C SNP entails a dramatic drop of the initiation confidence value to 0.33 (Figure 3.13C) since far less true starts contain Cs at position –12 (0.13) compared to false starts (0.3), see Table 3.9. Finally, it may also happen that the initiation confidence is always below the given threshold, independent of the SNP that is inserted (Figure 3.13D). This is based on the overall disadvantageous properties of a start site such that a single mutation cannot "boost" the overall disability of this start to initiation translation.

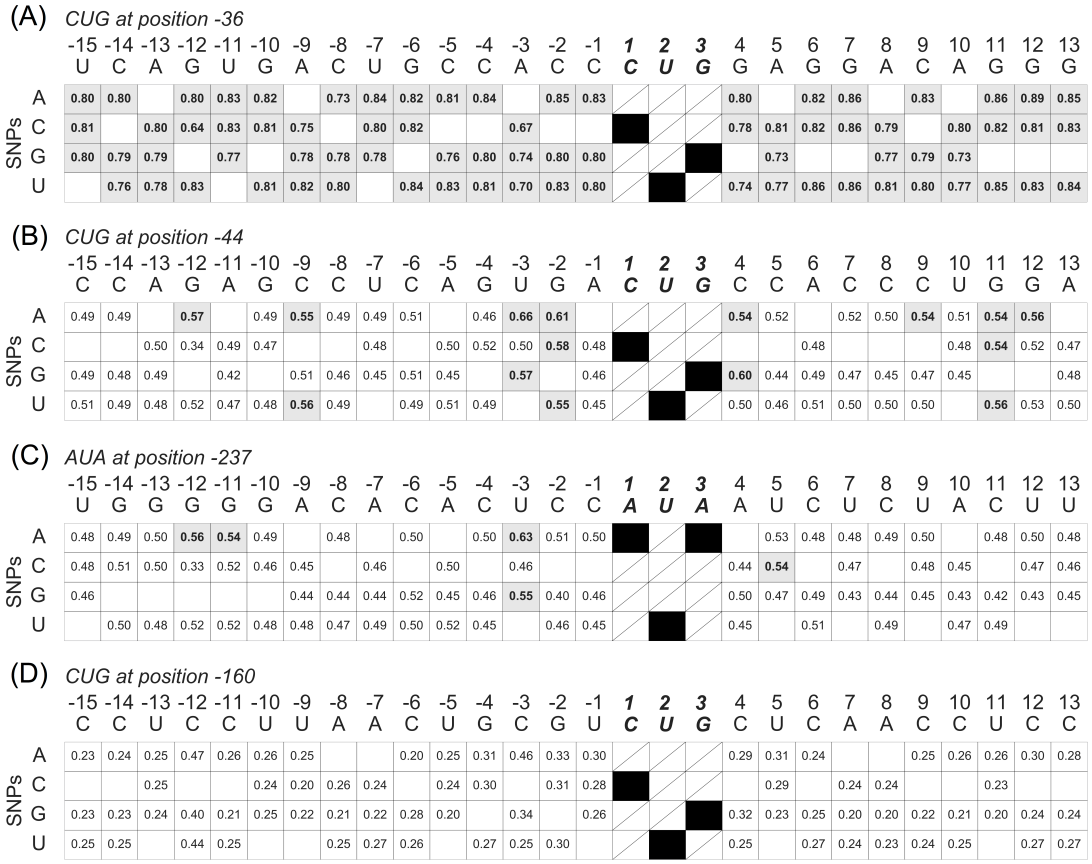

Figure 3.13: SNP analysis of gene GIMAP5. Mutation matrix showing the impact of the flanking sequence context of four putative start sites of gene *GIMAP5* on the predicted initiation confidence. In each case, only one nucleotide is mutated with respect to the reference sequence (top line). Grey means that the start was predicted as true translation start (predicted initiation confidence is greater than 0.54), whereas white means that the start was classified as false start. Mutations at the start sites itself were not considered. The numbers reflect the predicted initiation confidence values. A: CUG at position –36. B: CUG at position –44. C: AUA at position –237. D: CUG at position –160.

To investigate the influence on the predicted initiation confidence (IC) on a more general scale, we calculated the difference in the initiation confidence of a mutation (A, C, U, and G) compared to the wild–type sequence ($IC_{difference} = IC_{mutation} - IC_{wildtype}$) for all start sites in the 3,566 genes of the HEK293 dataset. The results are shown in Figure 3.14. For example, if adenine or guanine are inserted at position –3, the initiation confidence value increases, with median values of 0.11 and 0.06, respectively (see Figure 3.14). As mentioned, position –12 seems to play an important role in translation initiation. By comparing all start sites and possible mutations, a cytosine at this position lowers the initiation confidence by 0.16 on average.

Moreover, it was experimentally shown that positions +5 and +6 are important for efficient translation initiation, especially in non–AUG initiation [228]. More precisely, it was shown that the second codon (i.e. positions 4, 5 and 6) being G**AU** or G**CU** enabled an efficient translation initiation while G**UA** ablated initiation. Thus, an AU or CU seem to be important at position +5/+6 while UA is disadvantageous for translation initiation. This experimental finding can also be observed in Figure 3.14: A and C at position +5 increase and U at position +5 decreases the confidence value, while on the other hand at position +6, a U increases the confidence value.

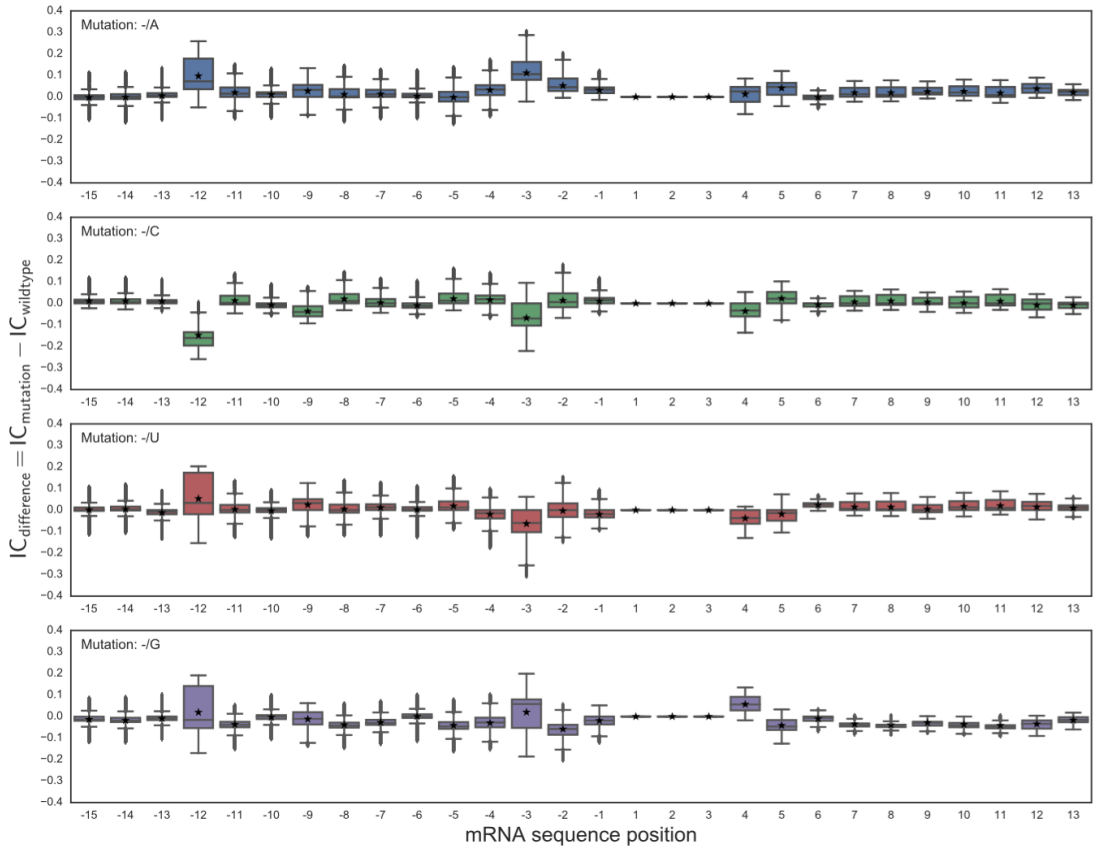

Figure 3.14: *In silico* mutation analysis of the HEK293 dataset. The flanking sequences of all possible start sites of all 3,566 genes in the HEK293 dataset were mutated. Shown is the difference in the predicted initiation confidence ($IC_{difference} = IC_{mutation} - IC_{wildtype}$). Positions –3 and –12 are prevalent and seem to have the largest influence on the prediction. Positions at the start site were not mutated.

3.5 Discussion

We were able to identify highly significant features belonging to three different feature classes (biologically–motivated features, PWM as well as k–mer features) that distinguish between true and false translation initiation sites. A simple linear regression model based on significant and uncorrelated features enabled us to reliably differentiate between true and false start sites. While a k–mer search enabled an unbiased scan, the biologically–motivated features reflect experimental observations regarding translation initiation. The PWM accounts for the flanking sequence context that is crucial to initiate translation. Also, it reflects the role of the start codon itself since it was shown that some codons (e.g. AUG and CUG) are used more often by the ribosome to initiate translation in mammals [43, 44].

Problematic was the inhomogeneous dataset that most likely contains some FPs and misses some TPs. Reasons for this may be experimental drawbacks, the proceeding steps of the raw data, or the cell line that was used (some start sites may only be used in specific cell lines). In general, several experimental steps have an influence on and can alter the data output: cell harvesting, nuclease treatment, and library generation [191]. The key idea of ribosome profiling is the inhibition of translation. This may introduce certain biases into the data. For example, if inhibition is slow, ribosomes can artificially accumulate at specific positions [191]. Moreover, RNA fragments (e.g. non–coding RNAs) can distort the translation readouts. Especially in sequence analysis, the mapping of the sequence reads from similar regions of different transcript variants is challenging. This is further complicated by the short length (about 30 nucleotides) of ribosome footprints [191]. Moreover, it is currently not possible to apply ribosome profiling to single cells, in contrast to mRNA–seq for instance [191].

Without doubt, the ribosome profiling technique is a huge innovation to understand translation initiation. However, it appears that the start codon selection based on the experimental outcome is challenging. For example, a GUG start in gene *RPLP1* at position –107 determined by Lee et al. [43] has the following flanking sequence context: GCC GCC AAG GUG CUC. In the light of the findings of Kozak [183, 184, 185], one may speculate whether the upstream AAG codon would be the more appropriate start codon. Nevertheless, the deep analyses of the different datasets presented here was able to point out crucial sequence features for which a solid experimental evidence exists (for example Kozak context) that significantly differed between the considered true and false start sites. This verifies and draws confidence that the overall ribosome profiling dataset(s) are suitable for the prediction of translation start sites.

Although, we used ribosome profiling applied to a specific cell line (HEK293) for training and testing, we propose that the predicted start sites have the potential to initiate translation in other cell types as well since the features used are only based on sequence properties. As a rather extreme example, we showed that the classifier trained on human HEK293 cells works reasonably well, albeit with lower accuracy, on mouse ES cells. Interestingly, we observed that the codon distribution of predicted start sites in the test set was similar to that of the experimentally observed start sites. This provides confidence in the quality of our prediction approach. Moreover, applying regression instead of classification enabled us to provide an initiation confidence value ranging from 0.0 to 1.0 rather than a strict decision between true and false start site. Subdividing start sites into different confidence classes c (very high: $c \geq 0.9$, high: $0.8 \leq c < 0.9$, moderate: $0.7 \leq c < 0.8$ and low: $t = 0.54 \leq c < 0.7$) helps to identify hot candidate start sites with very high initiation confidence values. The analysis of SNPs in the start site flanking sequence context showed that mutations can have a large impact on the initiation confidence. This not only holds true in our prediction approach but also in the context of *in vivo* translation. Kozak found that individual mRNA positions (–3, +4) are crucial for initiation [182, 183].

3.6 *PreTIS* web service

To make the *PreTIS* algorithm available to the scientific community, we implemented a web service to predict the initiation confidence of all reading frame–independent start sites (AUG and all near–cognate codons) located in the 5′ UTR given a human mRNA sequence. Thereby, the ribosome profiling datasets and the best human prediction model described above were used as underlying regression model. The web service application *PreTIS* requires an mRNA sequence and is accessible at:

`http://service.bioinformatik.uni-saarland.de/pretis`

Based on the given human mRNA sequence, all possible AUG and near–cognate start sites, with a surrounding window of at least $\pm$99 nt (needed to calculate k–mers) and an in–frame downstream stop codon, are identified in the 5′ UTR. *PreTIS* then calculates the required sequence features (see Table 3.9) for all detected start sites and subsequently predicts the initiation

confidence. Based on the predicted initiation confidence value and the given prediction threshold of $t = 0.54$, a start site is categorized into different initiation confidence classes. For start sites with confidence values c greater than the given threshold t, the four confidence groups were defined as follows: very high (hot/best candidates with $c \geq 0.9$), high ($0.8 \leq c < 0.9$), moderate ($0.7 \leq c < 0.8$) and low ($t \leq c < 0.7$) confidence, respectively. Especially start sites with very high confidence values can be considered as hot candidates for translation initiation.

The predicted initiation confidence for each start site is visualized by bar plots with the x–axis displaying the mRNA position (compare to Figure 3.12). This enables a comprehensive comparison of, for example, different flanking sequence contexts. Features calculated for each start site can also be downloaded as CSV files for further analyses. For the calculation of some features, an orthologous mouse sequence is required. This is automatically implemented by the embedded BLAST search [67]. The mRNA sequence found by `blastn` can be inspected afterwards and replaced, if desired. Furthermore, each job is given a Session–ID and a Job–Number, which enables unambiguous accession to the prediction results. In the following, details about the implementation of the *PreTIS* web service application are given.

Navigation bars allow a clear and expedient presentation

The *PreTIS* web service is presented by several subpages that are arranged by a horizontal navigation bar. Thereby, several *PreTIS* subpages are internally linked. In addition, BLAST and Ensembl websites as well as the journal website of the *PreTIS* publication can be reached via hyperlinks. The internal linking structure is illustrated in Figure 3.15. The webpages reachable by the HOME, ABOUT, and CONTACT tabs give a general overview on *PreTIS*, some background information such as presentation of the feature set, and a possibility to contact the developers, respectively. The NEW JOB tab opens an input form to provide a human mRNA sequence, separated into 5′ UTR and CDS, and to choose the start sites the provided sequence is searched for. Thereby, a homologous mouse sequence, needed to calculate sequence conservation, can be provided either by the user or retrieved by applying the embedded BLAST search. The input can subsequently be submitted for start site prediction.

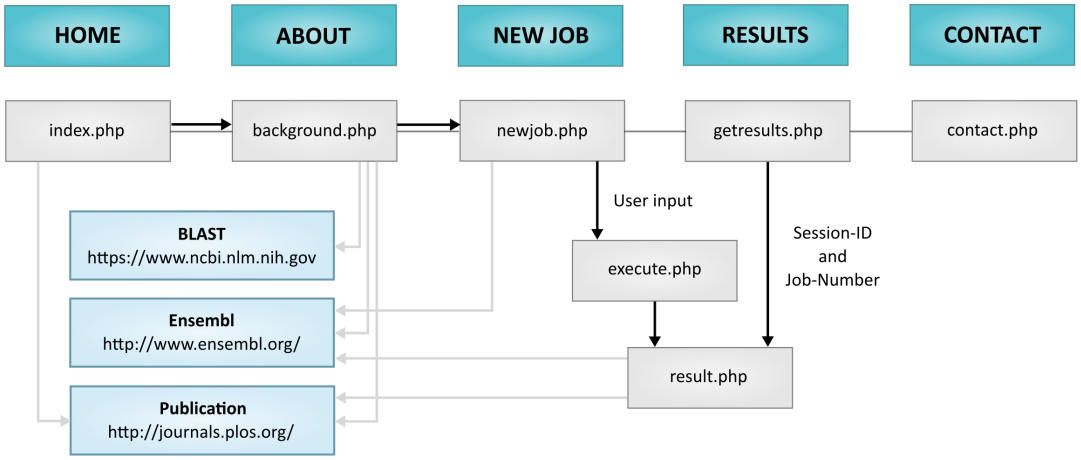

Figure 3.15: *PreTIS* web service subpage linking. The *PreTIS* webpage core element is a horizontal navigation bar allowing a clear subpage arrangement, here shown as turquoise rectangles. *PreTIS* embedded subpage files are colored in grey, whereas external websites, like Ensembl or BLAST, are highlighted in light blue. Linkage between *PreTIS* subpages is shown as black arrows, whereas external links are displayed as grey arrows.

Different requests are saved by assigning a unique Session–ID and Job–Number. Cookies are used to store Session–ID to only increase the Job–Number in case a user starts several *PreTIS* requests. Moreover, example data can be easily loaded for a first attempt to become familiar

with *PreTIS*. Previously computed results can be retrieved by providing the obtained Session–ID and Job–Number using the input fields of the RESULTS tab.

PreTIS core functionality: automatic initiation confidence prediction

The *PreTIS* web service was implemented using several (programming) languages: HTML, CSS, JS, PHP, and Python. The combination of the markup language HTML, the style sheet language CSS, and the programming languages JS, PHP, and Python enables the development of a dynamic web service for translation start site prediction from a mRNA sequence and a clear representation of the obtained results.

The client–sided front–end, implemented using HTML, CSS, JS, and PHP, represents the web service application and validates user input prior to data transmission to server. Upon validation, a BLAST search and start site prediction are executed on the server–sided back–end using PHP and Python. The connection between front– and back–end as well as the computational structure of *PreTIS* is shown in Figure 3.16. As mentioned previously, the HOME (index.php), ABOUT (background.php), and CONTACT (contact.php) tabs of the navigation bar allow access to the subpages that introduce *PreTIS* and give contact information. The functionality represented by the remaining tabs is explained in the following.

Sequence and parameter submission The central component of the *PreTIS* application is the submission form that is reachable via the NEW JOB tab. A user has several possibilities to submit a human mRNA sequence (5′ UTR and CDS). Sequences can be submitted either by an automatic sequence retrieval via a valid Ensembl gene ID such as "ENSG00000196329", by directly entering 5′ UTR and CDS sequences in the respective text area, or by uploading FASTA files from a local folder. It is only possible to submit a sequence by either entering the sequence or by uploading a text file. The fulfillment of this criterium is checked via a JS function. The retrieval of human mRNA sequences via an Ensembl gene ID is enabled by the Ensembl REST API [211]. REST stands for Representational State Transfer and allows to retrieve Ensembl sequence data in JSON file format, see Section 3.1.3. The requested sequence file is then parsed and all available transcripts are reported to a user. Dependent on the transcript ID that is selected by a user, a JS function is subsequently applied to display the respective sequence in the submission form. For convenience, the "Load example" button can be used to demand an example mRNA sequence (here *GIMAP5*, ENSG00000196329) to test *PreTIS* functionality in an appropriate way.

Upon input of a human mRNA sequence, a homologous mouse mRNA sequence is necessary for the calculation of 5′ UTR and start site conservation. This sequence can either be provided by a user or the embedded `blastn` function can be applied to search for the best mouse hit given a human mRNA sequence, see Figure 3.16. The latter option is set as the default, which can be changed using a checkbox in the submission form. Beside human and mouse sequences, the user can choose at least one out of the ten possible start sites: AUG, CUG, GUG, UUG, AAG, ACG, AGG, AUA, AUC, and AUU. The 5′ UTR is then scanned for all previously selected codons that are in– and out–of–frame with the canonical AUG. Following feature calculation, the developed machine learning model is applied to predict initiation confidences of all possible initiation sites. Thereby, the CDS is necessary to calculate some of the sequence–based features such as the k–mer features.

For security reasons, a validation of the provided user input is necessary prior to submission of the given sequences and selected codons to the server, see Figure 3.16. This validation step was implemented using JS. These functions verify that the provided sequences only contain the characters {A, C, T, G, U} (upper or lower case) while all other characters, like line breaks or spaces, are removed beforehand. In case, a sequence text file is uploaded, the text file must not exceed a size of 512 kB. JS is also used to check whether a provided human Ensembl gene ID is valid, that means it must start with "ENSG", have a length of 15 characters, and the string succeeding "ENSG" must be a number. Finally, at least one out of the ten possible start codons must be selected. Disregarding one of the above mentioned requirements will result in a notification via respective error messages.

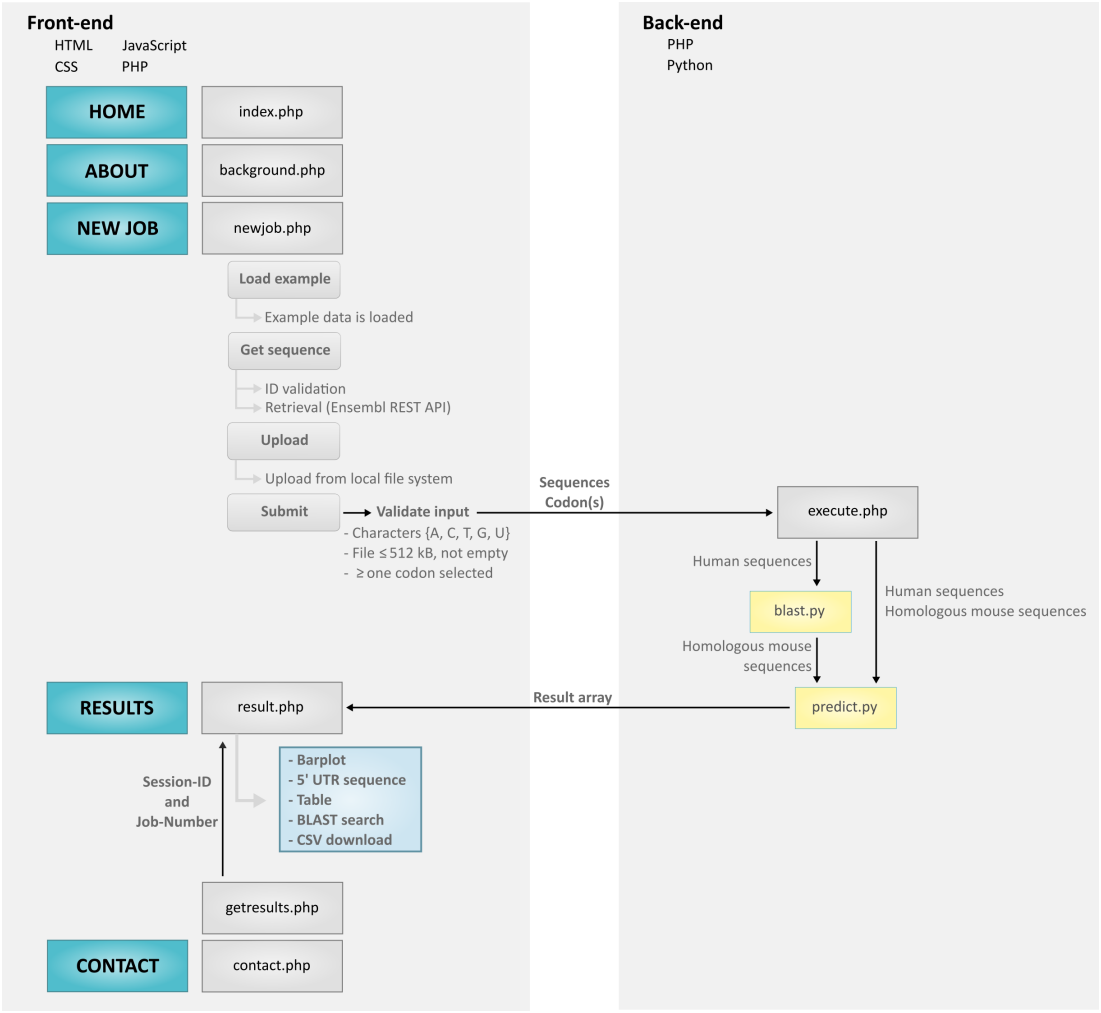

Figure 3.16: *PreTIS* internal web service structure. Combining HTML, CSS, JS, PHP, and Python enables the implementation of a dynamic web service for the prediction of start site initiation confidences for all candidate codons, given a mRNA sequence. The front–end provides background information and a submission form for sequence input and parameter specification. Following input validation, the server–sided back–end computes all necessary results such as codon positions and confidence values, which are then displayed to a user on the client–side.

Back–end result computation In case all validation criteria are fulfilled, data is transmitted to the server, compare with Figure 3.16. Assuming an automatic retrieval of an orthologous mouse sequence is intended by a user, the Python script `blast.py` is subsequently called for the execution of a BLAST search

```
blastn -db mouse_fasta -query human_query -num_alignments 0 -out result
```

with the preprocessed FASTA file containing mouse sequences `mouse_fasta`, the human mRNA sequence in FASTA format `human_query`, and a `result` file providing the detected BLAST hits. The `result` file is then parsed and the best hit is stored for further analysis. Next, the Python script `predict.py` is called with human mRNA sequences, mouse orthologous mRNA sequences, and the selected codons as arguments, in order to compute the sequence–based features and to predict translation initiation confidence values for every putative start codon.

The connection of and data exchange between Python and PHP is enabled via the PHP `popen()` function that opens a pipe to a program or process specified by the command that is given as argument, see Listing 3.4 (Lines 1–7). Here, the respective Python script together with parameters, like mRNA sequences, is provided. The results of the BLAST search and the initiation site prediction are then returned via the Python `print` function and subsequently parsed using PHP. For convenience, all final results are then saved as a PHP `array()`, see Listing 3.4. This functionality is implemented in execute.php, see Figure 3.16. The constant progress in the result computation is reported by a subpage that is displayed to the user in the meantime. Previously computed results can then be easily reached via the RESULTS subpage (getresults.php) by providing the assigned Session–ID and Job–Number.

Front–end result representation All computed results are represented by functions embedded in the result.php file. The combination of HTML, CSS, JS, and PHP allows a nice and clear presentation via an interactive bar plot depicting the confidence values, the provided mRNA sequence with codons colored according to the regression values, a table representation listing properties of all detected start sites, the result of the BLAST search with external Ensembl hyperlinks, and a CSV file with the summarized results as download. The *RESULT* subpage is separated into five parts: general information and CSV download, interactive bar plot, colored mRNA sequence, table representation, and BLAST results.

First, a short overview introduces and displays four initiation confidence categories, the Session–ID, Job–Number, and the selected codon(s). Predicted start sites are subdivided into different initiation confidence categories c with very high ($c \geq 0.9$), high ($0.8 \leq c < 0.9$), moderate ($0.7 \leq c < 0.8$), and low ($t = 0.54 \leq c < 0.7$) confidence to facilitate the identification of candidate initiation sites. Moreover, it is referred to the CSV download file containing information, like nucleotide extension, codon, or calculated feature values, for all predicted translation start sites. Our publication can be reached via a hyperlink as well.

The JS charting library Highcharts [209] was then used to visualize the predicted initiation confidence values as interactive bar plot, see Figure 3.17.

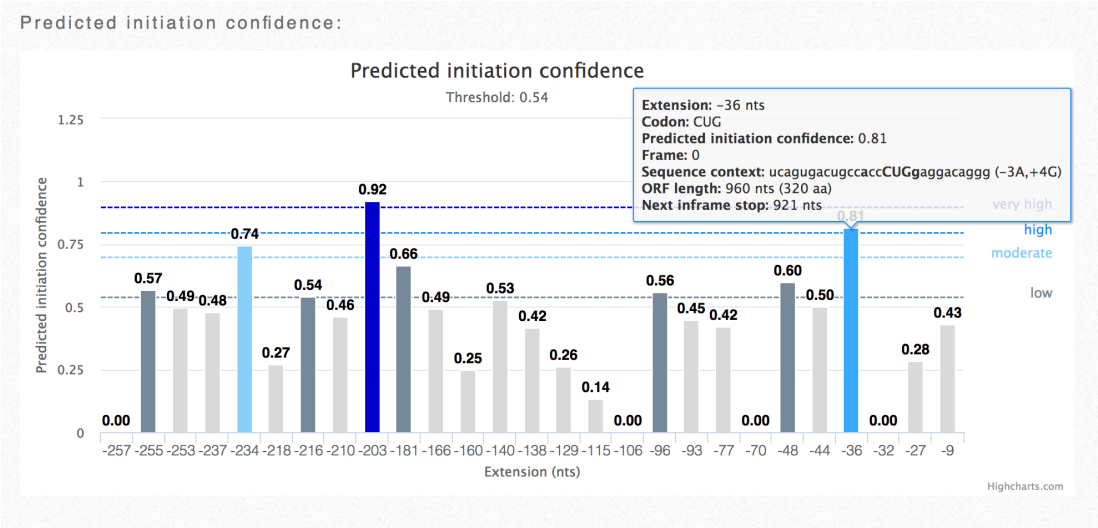

Figure 3.17: *PreTIS* web service bar plot representation of initiation confidences. Predicted initiation confidence values for every candidate start site of gene *GIMAP5* (ENSG00000196329) are presented as bar plot. This interactive visualization to display different start site properties was implemented using the JS Highcharts library [209]. The color scheme corresponds to the confidence categories c defined beforehand.

The mouse–over functionality provides additional information for every candidate start site, such as the nucleotide extension, the start codon, the frame with respect to the main ORF, the flanking sequence context, or the next in–frame downstream stop codon. This representation simplifies the comparison between the detected start sites concerning their confidence

values, positions in the 5′ UTR, or Kozak flanking sequence contexts. Candidate start sites are highlighted additionally within the complete mRNA sequence to emphasize the location of the individual candidate start sites together with the local sequence surrounding. This sequence visualization is denoted in Figure 3.18.

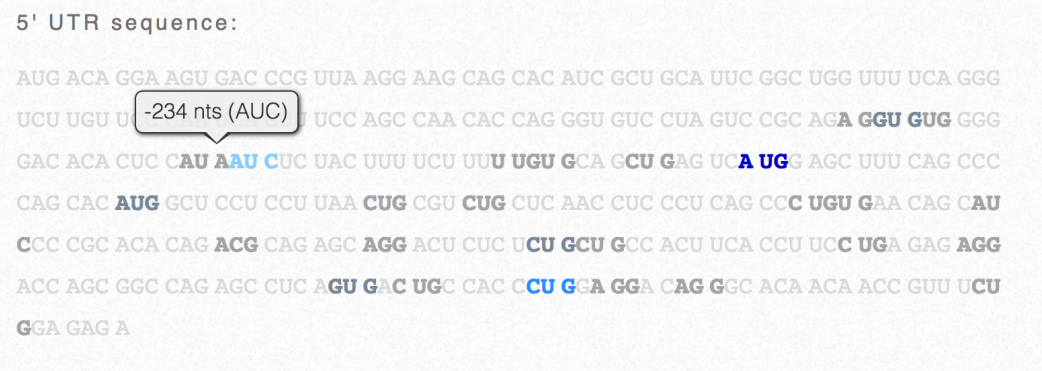

Figure 3.18: *PreTIS* web service 5′ UTR sequence visualization. Putative start sites of gene *GIMAP5* (ENSG00000196329) are highlighted within the 5′ UTR to emphasize their sequence positions. Mouse–over functionality is used to display the nucleotide extension with respect to the canonical start site. The color scheme corresponds to the confidence categories c defined beforehand.

A table representation, shown in Figure 3.19, gives a clear overview of all candidate start codons together with the nucleotide extension, codon, reading frame with respect to the main ORF, position of the next in–frame downstream stop codon, ORF length, and predicted initiation confidence. These values are also part of the downloadable CSV file. Finally, the best BLAST hit, in case a BLAST search was executed, is reported together with the respective Ensembl gene and transcript IDs (hyperlinked to the Ensembl webpage) and the 5′ UTR and CDS mouse sequences, compare with Figure 3.20.

Table view:

Extension (nts)	Codon	Frame	Position stop codon	ORF length	Predicted initiation confidence
-257	AGG	-2	-236	24	0
-255	GUG	0	921	1179	0.57
-253	GUG	-1	-169	87	0.49
-237	AUA	0	921	1161	0.48
-234	AUC	0	921	1158	0.74
-218	UUG	-2	-209	12	0.27
-216	GUG	0	921	1140	0.54
-210	CUG	0	921	1134	0.46
-203	AUG	-2	-137	69	0.92
-181	AUG	-1	-169	15	0.66
-166	CUG	-1	-76	93	0.49

Figure 3.19: *PreTIS* web service table view of features values. To precisely describe all putative 5′ UTR start sites of gene *GIMAP5* (ENSG00000196329), the exact position, codon, reading frame, and corresponding stop codon are displayed using a table representation. The color scheme helps to identify hot start site candidates for translation initiation.

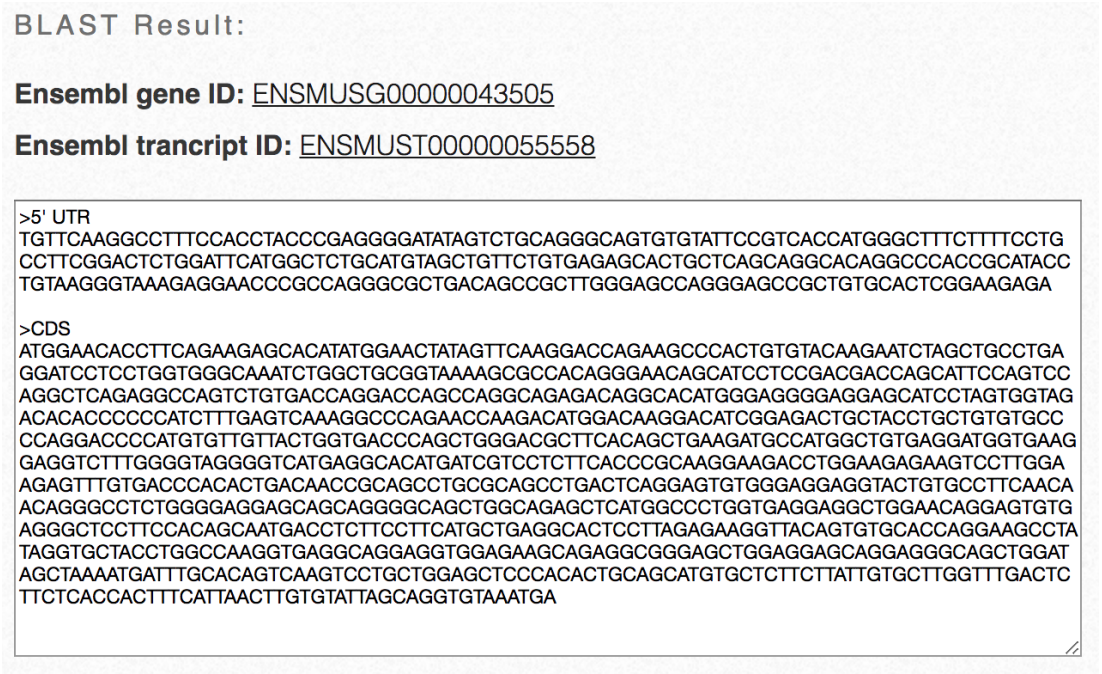

Figure 3.20: *PreTIS* web service BLAST result. In case the embedded BLAST search was made use of, the best homologous mouse hit is displayed together with Ensemble gene and transcript IDs that are hyperlinked to the respective Ensembl webpage. As the mRNA query sequence belongs to gene *GIMAP5* (ENSG00000196329), the reported mouse Ensembl gene ID corresponds to the mouse ortholog *Gimap5* (ENSMUSG00000043505).

Taken together, *PreTIS* is an intuitive tool that solely requires the human mRNA sequence as input. It gives access to various calculated sequence–encoded and experimentally shown important sequence properties for translation initiation. In addition, an initiation confidence value for each start site is calculated using an established regression model that is based on recent experimental data. All frame–independent AUG as well as alternative start codons are considered.

3.7 Summary

In this project, we dealt with alternative translation initiation in the 5′ UTRs of human and mouse mRNA sequences. We considered frame–independent non–AUG and AUG start sites that can, dependent on the exact location within the mRNA, lead to extended or alternative protein isoforms. Based on this, we developed a prediction model, named *PreTIS*, that detects alternative start sites in a given mRNA sequence. Thereby, all integrated features are based on mRNA sequence information. Our best performing model, with accuracies of about 80%, is based on experimental evidences of alternative starts found using ribosome profiling. In detail, *PreTIS* takes an mRNA sequence and assigns initiation confidence values to all potential alternative start sites located in the 5′ UTR. These initiation confidences are then categorized into "low", "medium", "high", and "very high" evidence, which helps to find hot candidate start sites for alternative initiation. As we observed that features concerning the flanking sequence context around alternative start sites can influence the performance of our prediction model, we can confirm that especially these regions are crucial for translation initiation. An additional investigation of the flanking sequence context via the introduction of *in silico* mutations also verified that translation initiation is prone to changes in the start site surrounding sequence. Our prediction model was also found to be transferable to the reliable detection of alternative starts in mouse sequences. To make *PreTIS* available to the scientific community, we decided

to embed the prediction model into a publicly available web service. This web service greatly supports the analysis of alternative translation in mRNA sequences by providing an interactive illustration of all predicted initiation confidences, by highlighting all putative start sites within the given mRNA sequence, and by summarizing all calculated features in downloadable tables.

CHAPTER 4

Mutation frequencies in key elements of the human genome

This chapter presents our investigations of mutation frequencies in several genomic key elements such as coding regions, CpG islands, and promoters. Inspired by our former project *PreTIS*, we were especially interested in variations in the flanking sequences of transcription and translation start sites. The manuscript "Mutation frequencies at transcription start sites and at canonical and alternative translation initiation sites in the human genome. Kerstin Neininger, Tobias Marschall, and Volkhard Helms." was submitted to *BMC Genomics*, see Sections 4.2 to 4.4. Data integration and analyses were performed by me. Tobias Marschall from the Algorithms for Computational Genomics group located at Saarland University assisted us in the discussion of the results and the write–up of the manuscript. Section 4.1 gives an introduction to the topic and provides background information on the used methods.

4.1 Prerequisites

The determination and analysis of human genetic variation plays an essential role in various research areas and revolutionized modern medicine. Various mutations are associated with disease phenotypes such as cancer. The influence of a mutation on the phenotype is highly dependent on the position of a mutation within the genome, for instance regulatory regions or the gene body. As even small changes can influence initiation mechanisms and start site recognition, we were especially interested in mutations that are located in direct vicinity to transcription and translation start sites. Especially the flanking sequence context of translation start sites is sensitive to polymorphisms, see Section 3.1.1 and Figure 3.14 of Section 3.4.4. Inspired by our previous project, see Chapter 3, we conducted a genome–wide in–depth analysis of mutations around these start sites together with a general investigation of mutation frequencies in several genomic key elements. In the following, variation types, their emergence, and the impact of human variations on (disease) phenotypes are presented. Moreover, the underlying data sources and bioinformatics tools used in this project are explained.

4.1.1 Genomic regions and their functional purpose

Eukaryotic genes comprise of several regions: intergenic region, promoter, 5′ UTR, coding exon(s), 3′ UTR, intron(s), and intragenic region [7]. The basic genomic regions are illustrated in Figure 4.1. Intragenic regions reside between a TSS and a transcription end site (TES) and are composed of 5′ UTR, coding exon(s), 3′ UTR, and intron(s). The 5′ UTR is located between the TSS and the coding start site (CSS), whereas the 3′ UTR resides between the coding or translation end site (CES) and the TES. Thereby, 5′ UTR and 3′ UTR refer to the exonic segments in these regions. Coding exons are defined by exon start and exon end positions. They reside between CSS and CES and are translated into a polypeptide sequence (canonical point of view). Introns are located within intragenic regions and are defined as the intervals between

the exonic sequences. The region between two genes refers to the intergenic region. Thereby, two intergenic regions enclose one gene. The first intergenic region ranges from the TSS of the considered gene to the mid–upstream position between the considered TSS and the TES of the next upstream gene. The second intergenic region is defined analogously in downstream direction. Genomic information such as transcription and coding start sites, strand (plus or minus), or exon positions can be retrieved from the UCCS genome browser [56, 163].

Promoter regions are located upstream of a TSS and can overlap with CpG islands [99, 131]. Promoter regions are essential for transcription initiation [7]. The core promoter is located close to the TSS, that means about 40 bp away from the TSS [7]. In our project, the gene promoter was defined as the region from 2000 bp upstream to 1000 bp downstream of the TSS. Normally, transcription factors can bind to the core promoter to attract RNA polymerase for transcription initiation [7]. The so–called TATA box core promoter is located between 28 and 34 bp upstream of a TSS [229]. As the name suggests the consensus sequence constitutes of adenine and thymine bases. TATA box–binding proteins can recognize and bind to this sequence to attract RNA polymerase II [7, 229, 230]. Transcription factors can also bind to enhancer and silencer DNA sequences, which play important roles in gene regulation by influencing activation or repression of their target genes [231].

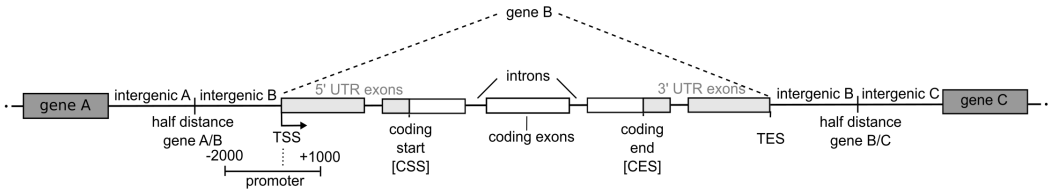

Figure 4.1: Definition of human genomic regions. Definition of the basic genomic regions: intergenic region, promoter region, 5′ UTR, coding exon(s), 3′ UTR, intron(s), intragenic region and CpG islands (not shown). Shown is the + strand, the – strand is analogous.

The flanking sequence context of translation start sites was experimentally shown to be prone to small base exchanges [185, 189]. A statistical start site analysis, which was carried out in our former project *PreTIS*, confirmed these tendencies, which were then integrated in our machine learning approach, see Section 3.4.4. Moreover, it was shown that mutations in promoter regions and transcription factor binding sites can have an impact on the transcription machinery (see below). A study that scored the influence of individual mutations on transcription factor binding is presented in Section 5.1.2. Thus, in addition to the mentioned "canonical" genomic regions, we defined four additional regions ranging from –200 to +200 bp as well as from –15 to +13 bp around transcription and translation start sites.

4.1.2 Human genetic variation

The Human Genome Project, initiated in 1990, was an international collaboration of various scientific research groups with the aim to fully annotate and sequence the three billion base pairs of the *Homo sapiens* genome [232, 233, 234]. The initial sequencing draft covered more than 90% of the human genome and revealed about 1.4 million SNPs [232]. The project was successfully completed in 2003 [233, 234]. The complete human genome sequence consisted of 2.85 billion nucleotides covering 99% of the euchromatic human genome [234]. A major aim of this large–scale project was the determination of all human (protein–coding) genes [234]. Finally, the International Human Genome Sequencing Consortium deciphered that the human genome consists of about 20,000 to 25,000 genes coding for proteins, not including RNA transcripts [234]. All data was made publicly available to the scientific community providing a basis for further research projects, for instance, focussing on the genetic variation between individuals.

Genetic variations and their emergence are diverse

Differences in individual bases between two genomes can affect the phenotype to a great extent. The three billion base pairs human genome consensus sequence from an European individual was reported to share 99.5% similarity with other individuals of European origin [235, 236]. These variations lead to a phenotypic variety and are mainly based on single nucleotide polymorphism (SNPs) [236, 237]. An A/C polymorphism is shown in Figure 4.2.

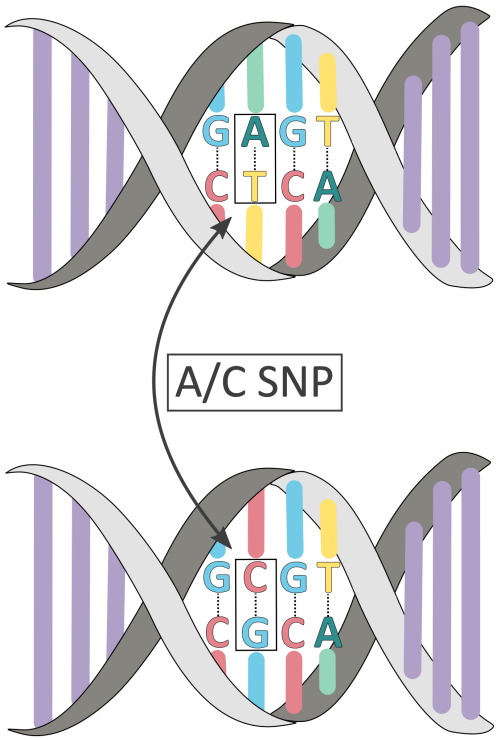

Figure 4.2: Definition of a single nucleotide polymorphism. Single nucleotide polymorphisms are defined as variations of individual nucleotides between the DNA strands of paired chromosomes or differences between separate genomes. Here, the reference base is an A, whereas the mutated sequences shows a C, resulting in an A/C SNP. The complementary base pairing leads to a change of A–T towards C–G, assuming the error is not corrected via repair mechanisms.

Variants can be acquired over time and can arise in somatic cells, which is known as somatic mutation, or can be passed from a parent to the offspring, which is entitled as germline mutation [238]. For instance, various mutations associated with cancer are somatic [238]. The development of variants in human populations is due to various diverse processes. For example, polymorphisms can be caused by DNA damage, erroneous DNA replication based on nucleotide misincorporation, incorrect DNA repair, or mobile genetic elements such as transposons [238, 239, 240]. Normally, mutations are eliminated by DNA repair mechanisms [241]. The number of new mutations per offspring was estimated to amount to approximately 100 polymorphisms [242, 243]. Moreover, about 70 *de novo* variants per offspring were reported based on sequencing of human parent–offspring trios [244]. Acuna-Hidalgo et al. [245] analyzed recent studies and reported an average of about 44 to 82 *de novo* mutations per offspring. Thereby, only one or two mutations were found in coding regions [245]. The association of *de novo* mutations with genetic disorders as well as their causes, genome–wide distribution and parental origin are reviewed in [245].

In general, one distinguishes between SNPs and the substitution of several nucleotides which is known as the collective term indels, short for insertions and deletions. SNPs are

separated into transitions and transversions, with transitions indicating an exchange between pyrimidine bases (C ↔ T) or purine bases (A ↔ G), whereas transversions denote the replacement of a purine base with a pyrimidine base or vice versa. The difference between transition and transversion SNPs is illustrated in Figure 4.3. Note that transitions are generally found more frequently compared to transversions, with a proportion of about two thirds to one third, based on their structural nature and the chemical conversion via deamination [128, 246].

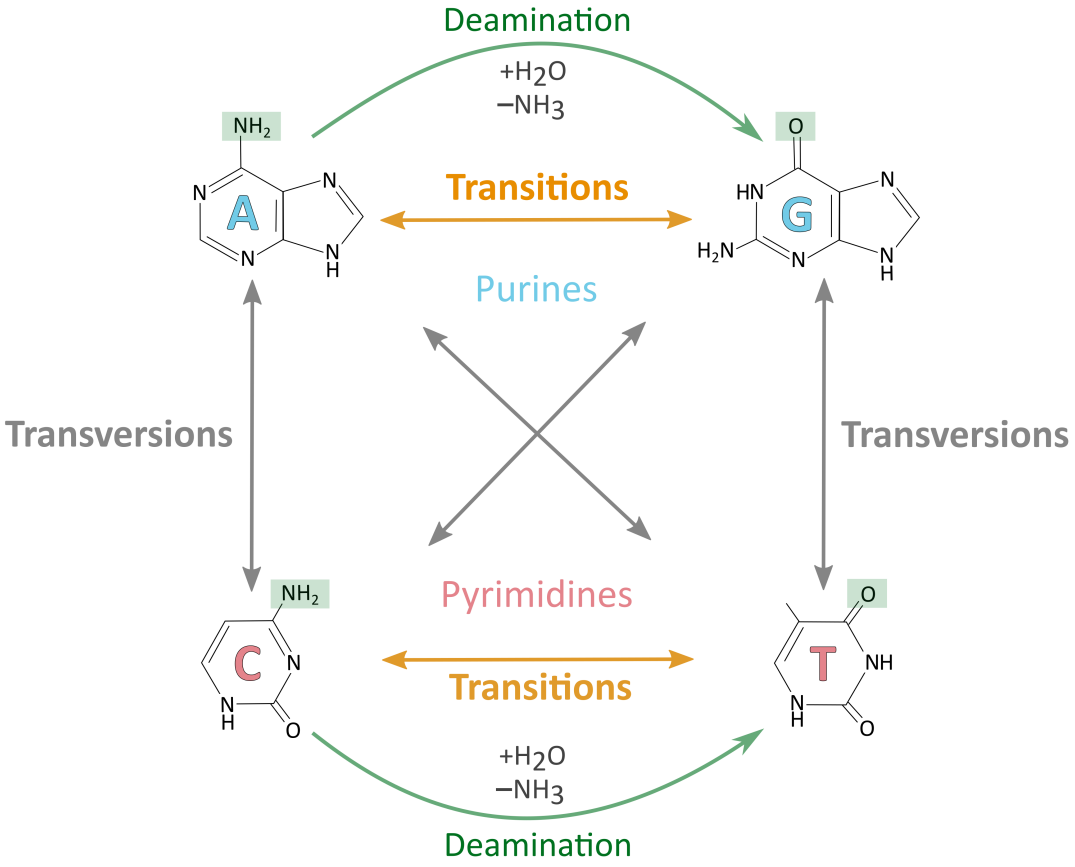

Figure 4.3: Transition and transversion SNPs. Polymorphisms of single bases are divided into transitions and transversions. Transitions underlie deaminations and occur much more frequently in the genome.

Common SNPs are assumed to occur at a minimum allele frequency of at least 1% within a population, which differentiates them from rare mutations with a frequency below 1% [238, 246]. The threshold of 1% was chosen arbitrarily and there is an ongoing debate on the exact definition of the terms "mutation", "polymorphisms", "point mutation", and "SNP" that are often used synonymously, see Karki et al. [238] for more details. SNPs are found frequently in human genomes and the general assumption is that a polymorphism is found generally every 1000 base pairs [235, 247, 248, 249], although their distribution within different genomic regions and even between chromosomes can vary [235, 250, 251]. Moreover, the distribution of SNPs within in the human genome was reported to be non–random, whereby the smallest amount of SNPs was found in coding regions [235].

Beside SNPs, copy–number variations contribute to genetic variations as well [252]. Copy–number variants, or copy–number polymorphisms, are defined as large duplication and deletion events of 100 kilo bases (kb) or even more [252, 253]. These genomic rearrangements also emerge in intragenic regions and can affect the phenotype of an individual by gene proliferation or deprivation [254, 255, 256]. Certain (inherited) alterations in the copy–number of an individual compared to the species average are associated with complex diseases [256]. For instance,

a specific duplication event was also associated with susceptibility to HIV and AIDS [257]. Leukodystrophy is an inherited neurological disorder that affects the central nervous system by a progressive loss of the myelin (white matter) sheath and can be caused by a duplication event [258]. Down syndrome, also known as trisomy 21, is caused by an additional copy–number of chromosome 21 [259]. Note that the main focus of our studies is on SNPs rather than copy–number variants.

SNP positions determine the impact on (disease) phenotypes

The influence of a SNP on the phenotype is dependent on the precise position or genomic region of the polymorphism [7, 95]. Mutations in coding regions can have an impact on the encoded protein and thus change the proteome, whereas mutations in regulatory sites, such as transcription factor binding sites, can affect gene expression and hence protein abundance. Of course, both alterations may shape a disease phenotype.

SNPs located within DNA coding regions can affect protein function or structure, although function and structure are in some way dependent on each other [7, 95]. Thereby, different types of mutations are distinguished. A synonymous mutation is present if the nucleotide substitution does not affect the encoded amino acid. This often applies to the third codon position, see Table 1.1 in Section 1.2. When the substituted nucleotide alters the encoded (wild–type) amino acid or has an effect on the reading frame, we speak of a non–synonymous mutation. Non–synonymous missense mutations result in an exchange of a single amino acid in the translated protein sequence, whereas readthrough and nonsense mutations lead to substitution of an amino acid with a stop codon and vice versa, respectively. A reading frame shift arises in the case when the number of substituted bases is not divisible by three. In consequence, the peptide sequence changes completely. Table 4.1 summarizes different coding mutation types and their impact on the encoded protein sequence.

Table 4.1: Different types of coding mutations. One distinguishes different coding mutations that arise dependent on the SNP position and the substitution. The nucleotide exchange is given together with the impact of the substitution on the amino acid sequence. The affected nucleotides are shown in bold.

Impact	**Description**	**mRNA**	**Protein**
Synonymous:	Wild–type and mutated amino acid are the same.	GG**C** → GG**A**	Gly → Gly
Non–synonymous:			
Missense	Translation of a different amino acid.	**A**CG → **G**CG	Thr → Ala
Readthrough	A stop is replaced by an amino acid resulting in an elongated isoform.	U**A**A → U**U**A	Stop → Leu
Nonsense	An amino acid is encoded as stop resulting in a shortened isoform.	U**C**G → U**A**G	Ser → Stop
Frameshift	The ORF is shifted based on an indel with the number of nucleotides not divisible by three.	**A**AG–CUG → **AC**A–GCU–G...	Lys–Leu → Thr–Ala–...

Mutations located in genic flanking sequences, like gene regulatory elements, can influence regulatory mechanisms and disease phenotypes. For example, *TERT* promoter mutations and overexpression of the telomerase reverse transcriptase (*TERT*) were observed in several cancer types such as glioblastoma, an aggressive brain tumor, or melanoma, a very malignant form of skin cancer [260, 261, 262]. Besides intragenic polymorphisms, mutations in the *TERT*

promoter region were found to be associated with a reduced life expectancy and increased the susceptibility to suffer from glioblastoma [260]. The up–regulation of *TERT* leads to maintenance of telomere elongation and consequently prevents apoptosis [260]. Promoter mutations in the *GJC2* gene were reported in hypomyelinating leukodystrophy patients [263].

Specific differences in individual genes or gene regions were reported to be associated with disease phenotypes as well [264]. Cystic fibrosis is based on a coding mutation in the CFTR transmembrane protein known as the "cystic fibrosis transmembrane conductance regulator" [265]. The risk to suffer from Alzheimer's disease is related to a polymorphism in both the apolipoprotein E encoded by the *APOE* gene and the *APOE* promoter [266]. Deletion of a C–C chemokine receptor type 5 (*CCR5*) segment is associated with resistance to HIV and *CCR5* is thus subject to studies aiming at identifying new target proteins in HIV research [267]. Sickle–cell anemia is caused by a mutation in the gene that encodes the β–subunit of the hemoglobin protein which is necessary for oxygen transport, resulting in a higher mortality rate and showing symptoms such as anemia [268, 269]. Beside the mentioned negative impact, the sickle–cell causing polymorphisms can also be advantageous in developing countries making a carrier of this mutation resistant to malaria [268]. Nevertheless, the genetic basis and causes of various genetic diseases are generally very complex [270].

Multiple research areas benefit from consideration of genetic variants

Human genetic variation is of great interest in various research fields such as population genetics, the development of new drugs, a profound understanding of genetic diseases like cancer, or the investigation of a connection between genotype and phenotype [246]. The clinical phenotype of various diseases such as diabetes, mental disorders as well as cardiovascular and autoimmune diseases are assumed to be associated with genetic variation in specific genes [246]. Moreover, DNA fingerprinting and the analysis of DNA profiles contributed tremendously to forensic evidences, which is reviewed elsewhere [271].

The field of personalized or precision medicine benefits from new sequencing technologies as well. Personalized medicine aims at medical treatments tailored to an individual and going beyond classical treatment in which the same medication is applied to millions of (genetically different) people [272]. Thereby, the consideration of DNA variants is essential [272, 273, 274]. For instance, colorectal cancer patients with a mutated *K–ras* gene were reported to respond better to the treatment compared to patients with the wild–type *K–ras* gene [274]. Moreover, survival rates in leukemia patients tested positive for the Philadelphia chromosome, an aberrant chromosomal translocation found in leukemia cancer cells, were found to increase upon treatment with a specific drug functioning as tyrosine kinase inhibitor [273]. This underlines that the consideration of the genotype and the resulting selection of an appropriate medication is crucial in the treatment of complex diseases such as cancer [275]. Most often, diseases are not caused by a single variant but are based on (a) particular genetic predisposition(s) together with external or environmental influences that contribute to these so–called complex diseases [246]. Studies that deal with the linkage between polymorphisms and a trait, such as the onset of a disease, are known as genome–wide association studies (GWAS) and are reviewed elsewhere [276, 277, 278]. Another study aimed at investigating the geographic distribution of polymorphisms [279]. They sequenced 3,000 citizens from Europe and analyzed their genotype with respect to DNA variants. Using principal component analysis (PCA), genetic variance was visualized in two dimensions (axis PC1 and axis PC2). Interestingly, this kind of visualization and the clusters that formed, mirrored the European geographic map. Thus, the genotype "correlated" with the geographic distribution, meaning that individuals could be assigned to the region they originate from just based on their genetic variants.

Moreover, two large–scale genome sequencing projects, the 1000 Genomes Project [57] and Genome of the Netherlands (GoNL) project [58, 59], reported tens of millions of polymorphisms that are publicly available. Their aim was to decipher and analyze human genetic variation in detail. Variation data from these two major projects was used in our project to examine mutation frequencies in several genomic elements with a focus on transcription and translation (alternative) start sites. The underlying datasets are explained below.

Calculating evolutionary conservation

Computation of sequence conservation between genomes or species can shed light on the functional importance and relevance of individual genomic regions. A simple measure of evolutionary sequence conservation is the number of SNPs in a predefined region, defined as the SNP density [250]. Thereby, a low SNP density is associated with a conserved region and vice versa [250].

Moreover, Nei and Li [280] introduced the term nucleotide diversity that is defined as the average number of differing bases between two genomes. Nucleotide diversity can be used as an evolutionary measure to determine sequence variation and compare DNA sequences of different genomes [264, 280]. For instance, the widely used Tajima's D statistic considers nucleotide diversity and can be applied to test for the neutral mutation hypothesis [281]. Neutral theory assumes that the vast majority of mutations are neutral with respect to selection. This means that evolutionary changes in DNA sequences and proteins are assumed to be based on a random drift of selectively neutral mutants [282]. Tajima's D test statistic was defined as the normalized difference between π and θ

$$D = \frac{\pi - \Theta}{\sqrt{Var(\pi - \Theta)}}$$

with the average number of nucleotide differences (average sequence diversity) π, and

$$\Theta = \frac{S}{\sum_{i=1}^{n-1} \frac{1}{i}}$$

with the number of segregating sites S and the population size n [281]. Segregating sites are defined as sites that show at least two different nucleotides when comparing homologous DNA sequences [283]. π can be estimated as the number of pairwise differences divided by the total number of $\binom{n}{2}$ possible pairwise alignments [284]. Tajima's D statistic was applied for the analysis of human polymorphisms in DNA coding regions [285], the association of rare variants with complex diseases [286] as well as pharmaceutical studies involving membrane transporters [287] to name some examples.

4.1.3 Data sources and bioinformatics tools

The analysis of human variants in different genomic elements and in the flanking sequence context of transcription and translation start sites requires a collection of several datasets. Data for this project was taken from the UCSC Genome Browser and two large–scale sequencing projects. VCFtools, BEDTools (explained in Section 1.4), and Bowtie were used in the analysis pipeline. The mentioned tools and datasets are presented in the following.

UCSC Genome Browser

The UCSC Genome Browser is a web service for the fast and comprehensive visualization of human genome annotations [56]. The browser can visualize whole–genome information like gene annotations and predictions, expression and regulatory information, or variation data. The UCSC Table Browser can be used for the (filtered) retrieval of various genome annotations from the Genome Browser Database [163]. The Table Browser complements the Genome Browser and enables a large–scale processing of data in common file formats such as tab–separated text files or BED format. For the analysis of mutations in different genomic elements, human genome annotations (RefSeq genes [288], hg19 assembly) were downloaded using the UCSC Table Browser.

Large–scale genome sequencing projects

Two of the most elaborate large–scale sequencing projects in recent years are the 1000 Genomes Project [57] and Genome of the Netherlands (GoNL) project [58, 59]. The 1000 Genomes Project reconstructed the genomes of 2,504 individuals from 26 populations. While this reconstruction was mainly based on low–coverage data, the GoNL consortium focused on a smaller number of individuals, 250 Dutch parent–offspring families, but sequenced at a higher coverage. Additionally, the GoNL experimental setting allowed the identification of *de novo* mutations [58, 289]. The variations provided by these sequencing projects were used as basic datasets for the analysis of mutations in various genomic elements such as promoter region, 5′ UTR, coding regions, and in the flanking sequence of transcription and translation start sites. All variants are provided in commonly used variant call format (VCF).

The VCF file format is the conventional text file format for storing variant calls together with necessary information such as genome position, allele frequency, or quality [72]. Thereby, variants comprise single nucleotide polymorphisms, insertions and deletions as well as structural variants. This file format was initially developed to efficiently store data from the 1000 Genomes Project, see [57] for the final publication of project phase 3. VCF files are composed of a header section that displays meta–information starting with "##". The last header line defines the column names and starts with "#". The body of a VCF file then lists the detected variants with additional information such as position or allele frequency. Thereby, eight tab–separated columns are mandatory, see Table 4.2. For more detailed information on VCF file format, please refer to Danecek et al. [72].

Table 4.2: VCF file format description. The VCF file format was initially developed to annotate and store variants from the 1000 Genomes Project [57]. The eight listed columns are required within the body of a VCF file. Header meta–information is given prior to the body and starts with "##". SVTYPE refers to the structural variant type such as a deletion (DEL) or inversion (INV). Detailed information on the variant call format is given by Danecek et al. [72].

Column name	Description	Example
CHROM	Chromosome	1
POS	Variant position (1–based)	1000
ID	Variant identifiers (unique)	rs123
REF	Reference allele	C
ALT	Alternative alleles (comma–separated)	G, CA
QUAL	Quality score (Phred scaling)	50
FILTER	Information on variant site filtering	PASS
INFO	Additional annotations (semicolon–separated)	SVTYPE=DEL

VCFtools

VCFtools is a software package that implements several functions for the analysis of variation data given as VCF files [72]. The provided functions include validation, filtering, comparisons, estimation of allele frequencies, quality control, and general statistics. The `TajimaD` function, provided by the VCFtools suite, was used by us to estimate the evolutionary pressure on different genomic elements. The data basis was provided by the 1000 Genomes Project [57]. Tajima's D statistic [281] is widely used to evaluate the neutral evolution hypothesis, see Section 4.1.2.

Bowtie

Bowtie is an efficiently implemented short read alignment program [71]. The speed is achieved by taking advantage of the Burrows–Wheeler transformation for indexing of the reference genome. Burrows–Wheeler transformation is explained in detail in Section 5.1.1. We applied Bowtie for the retrieval of genomic coordinates from mRNA sequence information. This was necessary for the analysis of the flanking sequence context around canonical and alternative translation start sites.

GO terms and functional gene annotation

The Gene Ontology Consortium initiated the Gene Ontology (GO) project in 1998 with the aim to develop a uniform vocabulary to describe the role of genes and gene products across organisms [290, 291]. They defined three separate ontologies referred to as biological process, molecular function, and cellular component. A biological process is defined as a cellular purpose or aim in which genes or gene products are involved in. Examples are the GO–terms "cell aging", "translation", and "transmembrane transport". A molecular function is referred to as all biochemical activities, thereby also considering binding to specific ligands or structures. For instance, "hydrolase activity", "DNA binding", and "penicillin binding" are examples for molecular functions. A cellular component is defined as the cellular location where the gene product performs its functionality. Examples for cellular components are "membrane", "nucleus", and "integrator complex".

Given a list of gene or protein identifiers, the DAVID Bioinformatics Resources database can be used for functional enrichment analysis [292]. The following analyses can be conducted: gene functional classification, functional annotation chart/clustering, and generation of a functional annotation table. The underlying database is composed of more than 40 annotation classes such as GO terms, pathways, expression data, and protein domains. We applied DAVID to investigate functional roles of genes with a CpG site located directly upstream of the transcription start site at position –1, which was found to harbor an elevated mutation rate compared to the local surrounding.

4.2 Introduction

Polymorphic sites in the genome are a major source of phenotypic variation in human populations [237]. These polymorphisms are caused by random mutational processes such as nucleotide misincorporation during DNA replication, DNA damage, or erroneous activity of DNA–processing enzymes [239, 240]. These mutational forces are typically counter–acted by DNA repair mechanisms [241] and by Darwinian selection [293]. In general, there exist two major types of point mutations, transition and transversion SNPs, as well as insertions and deletions (indels). Thereby, transitions (mutation from C to T, or G to A on the second strand), are the most frequent substitution found [128]. Hence, in our study we considered data from two large–scale sequencing projects that characterized the variability of human genome sequences: the 1000 Genomes Project [57] and the Genome of the Netherlands (GoNL) project [58, 59]. While the 1000 Genomes Project reconstructed genomes from 2,504 individuals from 26 populations, mostly from low–coverage data, the Genome of the Netherlands project focused on 250 Dutch parent–offspring families sequenced at higher coverage, which additionally allowed to identify *de novo* mutations [59, 289]. Venter and colleagues recently presented an analysis of variations in 10,000 human genomes [250].

In general, conservation can be measured either through multiple sequence alignments between species or through population analysis. Protein encoding sequences are typically strongly conserved whereby conservation increases from the 5′ UTR towards the coding sequence [294]. Outside of protein–encoding exons, highly conserved DNA elements are often associated with transcriptional regulation such as promoter regions upstream of the transcription start site of genes and tissue–specific enhancers. In consequence, polymorphisms occur

more frequently in genomic regions without a known function [295]. The SNP density, defined as the number of SNPs in a predefined region, can be seen as a simple measure for evolutionary sequence conservation, whereas a low SNP density indicates a strongly conserved region and vice versa [250]. Another widely used measure that can be applied to test for the neutral mutation hypothesis is Tajima's D statistic [281]. To name a few studies, Tajima's D statistic was used to analyze SNPs in coding regions of human genes [285], to investigate SNP and haplotype variation [296, 297], to examine drug response involving membrane transporter genes [287], to analyze rare variants and their contribution to complex diseases [286] and a human polymorphic inversion that disrupts a specific gene [298].

The start site flanking region of translated sequences was shown to play a crucial role in translation initiation and is important for the recognition of the start sites by the ribosome scanning complex [182, 183, 189, 299]. Pioneering work considering the influence of the flanking region on translation initiation was conducted by Marilyn Kozak in the 1980s. She found that a purine at position –3 and a guanine at position +4 are crucial for efficient translation initiation [182, 183]. Note that these positions are always given relative to the translation start site. When the position –3R (R = purine) is replaced by a pyrimidine, translational efficiency was reported to become more dependent on other positions, for example positions –1 and +4 [183]. Noderer et al. [189] analyzed the influence of all possible translation initiation starts between positions –6 and +5 using an experimental technique called FACS–seq. They confirmed the high influence of the sequence context in direct vicinity of the start site on translational efficiency. Moreover, mutations in the translation start site of protein KLHL24 resulted in a shortened polypeptide due to ribosomal read–through of the mutated start site and initiation at an in–frame downstream AUG–Methionine [299]. Thus, SNPs in and around translation start sites can influence on the efficiency of the start site recognition and thus the translation machinery.

Beside the canonical AUG, other codons that differ from AUG by one nucleotide were shown to also function as translation start sites [42, 43]. These so–called alternative start sites can occur in the 5′ UTR, CDS, or 3′ UTR. In general, 5′ UTRs comprise the largest fraction of alternative translation starts [24]. Dependent on the location of a 5′ UTR alternative start site relative to the annotated start site, translation can be in–frame or out–of–frame and therefore result in, for instance, small upstream ORFs, elongated proteins, or alternative proteins [169]. Alternative translation initiation, and thus the resulting alternative proteins, are involved in regulatory processes and can be targeted to different cell compartments [174, 176, 181]. Ribosome profiling is an experimental technique to determine (alternative) translation start sites [43, 44, 170, 191]. A prediction model called *PreTIS* was developed by us to assist in the analysis of 5′ UTR sequences and reveal alternative reading frame independent translation start sites in human and mouse [226]. Providing a mRNA sequence, *PreTIS* predicts the translation initiation confidence of all putative 5′ UTR start sites (AUG and all nine near–cognate codons). The prediction model is based on mRNA sequence–encoded features that, for instance, incorporate position weight matrices and the crucial positions –3 and +4 mentioned above. Thus, *PreTIS* can also be used to predict how mutations in the start site flanking region can influence start codon recognition.

In this study, we carried out a systematic analysis of transition and transversion SNPs as well as indels that occur in nine types of genetic elements (coding and non–coding regions) in the human genome. The investigated regions comprise the intergenic region, CpG islands, promoter, 5′ UTR, coding exons, 3′ UTR, all exons, introns and intragenic region, see Figure 4.1. As primary datasets, we used SNPs and indels reported in the European cohort of the 1000 Genomes Project [57] and the GoNL project [58, 59]. To test for neutrality, we applied the widely used Tajima's D statistic to different genetic elements [281]. Since SNPs around transcription and translation initiation start sites may have direct effects on gene transcription and protein translation, we investigated these SNP patterns in detail. Special attention was given to the start site flanking region that was defined as a window ranging from –15 to +13 with respect to the start site. With the common assumption that conservation reflects functional relevance, we also compared the SNP distribution of canonical start sites and experimentally detected [43] alternative start sites in the 5′ UTR.

4.3 Materials and methods

The aim of our study was a detailed investigation of mutation frequencies in several genomic key elements such as promoter regions and coding sequences to estimate their functional importance. Moreover, we analyzed mutations in the flanking sequences of transcription and translation start sites as these variations can impact the regulatory machinery. In the following, the data sources and analysis steps are explained.

4.3.1 Data integration and mutation analysis

This study is based on variation data form two major sequencing projects. Moreover, mutations were mapped in a genome–wide fashion to several genomic regions and analyzed subsequently. In the following, the datasets used in this study together with the applied bioinformatics tools are presented.

Variation data and genomic regions

Information about annotated SNPs and indels in human genes was used from the 1000 Genomes Project (1000G, phase 3, using only the EURopean super population, 503 individuals) [57] and from the Genome of the Netherlands project (GoNL, release 5) [58, 59]. Data was provided in VCF file format. For the analyses, we kept autosomal SNPs with a minor allele frequency larger than zero, whereby allele frequencies were calculated from the respective consortium, see [57, 58, 59]. These variants were assigned to four classes, namely transition SNPs, transversion SNPs, indels (without length cutoff), and the union of all variants.

Human gene annotations were downloaded from the UCSC genome browser hg19 assembly (RefSeq genes) [56, 163]. We removed genes coding for microRNAs and small nucleolar RNAs, genes with CDS start equal to the CDS end as well as genes located on chromosomes other than chromosome 1 to chromosome 22. Special care was taken of overlapping genes, where we distinguished between overlaps located inside other genes and staggered overlaps (genes overlap partially). Genes inside other genes were excluded. All genes with staggered overlap were collected and from each collection, only one gene was selected randomly to avoid overlapping genes. In total, about 5% of all genes were removed due to overlaps. If a gene has more than one transcript variant, only the longest transcript was retained.

For a general overview on SNP frequencies in the human genome, nine basic genetic regions were derived for every gene based on the genomic information provided by the UCSC genome browser. These regions comprise: intergenic region, CpG islands, promoter region, 5′ UTR, coding exons, 3′ UTR, all exons, introns, and intragenic region, see Figure 4.1. The regions were defined as described in Section 4.1.1. The information needed to calculate the genomic coordinates of these regions for every gene was downloaded from UCSC genome browser and includes chromosome, strand, TSS, TES, CDS start and end, exon starts and exon ends. Besides these nine general regions, we also considered narrow sequence windows of ± 200 bp around transcription and translation start sites as well as in their direct vicinity and ranging from –15 to +13 bp, see Figure 4.4.

Bioinformatics tools for mutation analysis

Any calculations requiring interval arithmetic and sequence mapping were implemented using the BEDTools suite (version v2.26.0) [78], samtools (version 1.3.1) [75, 76] and/or Bowtie (version 1.1.2) [71]. These operations include the assignment of SNPs and indels to their respective genes and genetic elements as well as the retrieval of genomic coordinates by a nucleotide sequence context and vice versa. SNP densities, defined as number of SNPs per kb, were then calculated for the different variant types and the nine basic types of genetic elements. Note that

Figure 4.4: Definition of sequence context around translation start sites. Per definition, positions around the translation start site are given relative to the start site, which is denoted as 1,2 and 3. Position zero is left out. Positions –3R (R = purine) and +4G were shown to be crucial for translation initiation [182, 183] and are therefore highlighted in red color.

the calculated SNP density is a function of the cohort size. The evaluation of the neutral evolution hypothesis was analyzed by the widely used Tajima's D statistic [281] for every genetic element. For this, we applied VCFtools (version v0.1.13) [72] with a bin size of 1 Mb (mega base) to filtered VCF formatted variant files that only contain variants found in the respective genomic regions. Tajima's D aims at testing for the neutral mutation hypothesis by comparing two nucleotide diversity measures for genetic variation: the number of segregating sites and the average sequence diversity or number of nucleotide differences [281]. Tajima's D was only applied to the SNP data from 1000G because it provides publicly accessible genotype information. Two–tailed Wilcoxon rank sum tests together with Bonferroni correction were used for the statistical comparison of different SNP types within the nine genetic elements. Thereby, we assume a p–value p to be significant if $p < 1.4 \times 10^{-3}$ with $\frac{0.05}{\#tests}$. The pairwise comparisons included $\#tests = \frac{9 \times 8}{2}$.

4.3.2 Statistical permutation testing

Since the flanking regions of transcription and coding start sites have direct effects on gene transcription and protein translation, we investigated these regions in more detail and at higher resolution with respect to their SNP and indel distribution. SNPs and indels can, for instance, influence the binding of transcription factors in the promoter region or the translation initiation of the ribosome scanning complex in the 5′ UTR [182, 183, 300]. We therefore examined the average SNP density in a range of ±200 bp around the TSS and CSS and subsequently focused on mutations in direct vicinity ranging from positions –15 to +13 of these start sites.

As translation initiation was shown to be especially dependent on the start site flanking region [182, 183, 189, 299], we analyzed annotated (RefSeq genes) as well as alternative start sites located in the 5′ UTR in detail. RefSeq genes were retrieved as described above while alternative start sites in human HEK293 cells were retrieved from experimental ribosome profiling data and used as annotated by the original authors [43]. To investigate the flanking region around translation start sites, we defined a sequence window from –15 to +13 relative to a start site that encompasses positions 1, 2 and 3, see Figure 4.4. Next, duplicated sequence contexts (for example from several transcript variants) and codons differing from AUG and near–cognate variants were removed. SNPs from 1000G and GoNL were then mapped to these sequence contexts. Indels were excluded from further analysis since the amount of indels located in the predefined sequence window from –15 to +13 is small such that a profound significance analysis is not possible.

We conducted a permutation test to investigate whether the negative peak at the CSS (positions 1, 2, and 3) is statistically significant. That means we calculated the probability to detect a negative peak of a certain magnitude by random sampling. For this, sequence contexts were represented as binary strings, with 1 representing a mutation at a position, and 0 otherwise. First, for all mutated translational sequence contexts detected by our analyses, in the following denoted as WT for wild–type sequence, we calculated the z–score

$$\phi_{WT} = \frac{\mu_{WT\{p\}} - C_p}{\sigma_{WT\{p\}}}$$

with the average number μ of SNPs over all positions, except positions at the CSS (i.e. $p \notin \{1, 2, 3\}$), the average number of SNPs C_p at positions $p \in \{1, 2, 3\}$, and the respective standard deviation σ. We then randomly shuffled (SH) all binary sequence contexts, for instance 00101 can be shuffled into 10100 or 11000 by switching positions randomly, and calculated analogously:

$$\phi_{SH} = \frac{\mu_{SH\{p\}} - C_p}{\sigma_{SH\{p\}}}.$$

The p–value p was then computed as

$$p = \frac{1}{r} \times \sum_{i=1}^{r} c(\phi_{WT}, \phi_{SH}) \quad with\ c(\phi_{WT}, \phi_{SH}) = \begin{cases} 1, & \phi_{SH} \geq \phi_{WT} \\ 0, & \phi_{SH} < \phi_{WT} \end{cases}$$

with r representing the number of shuffle repetitions, here $r = 10,000$. We assume a p–value to be significant (Bonferroni corrected) if $p < \frac{0.05}{4} = 0.0125$ with $\#tests = 4$ when considering canonical and alternative start sites as well as 1000G and GoNL data.

4.4 Results and discussion

In this study, our primary focus was to investigate mutation frequencies at transcription and translation start sites in predefined sequence windows. Furthermore, we compared canonical with alternative translation start sites detected by ribosome profiling to shed light on translational regulatory complexity. Before addressing these specific points, we will start with a general comparison of the data from the 1000G and GoNL sequencing projects with current literature.

4.4.1 Variant distribution in nine sequence elements

For our study, we selected SNPs and indels reported by (a) the European cohort of the 1000G project (23,938,159 variants remained after filtering), and (b) the GoNL project (20,706,633 variants remained after filtering. 63% of the annotated 1000G variants on human autosomes were transition SNPs resulting from deamination and tautomerization, whereas 30% were transversion SNPs. The remaining variants (7%) were indels. Considering GoNL data, the distribution was very similar with 65%, 29% and 6% representing transitions, transversions, and indels, respectively.

Since different DNA elements such as CpG islands, 5′ UTRs, protein–encoding exons or intergenic regions may exhibit different patterns of sequence conservation, we separately investigated these elements in the 16,604 RefSeq genes that remained after filtering. We used SNP density (SNPs per kb) and Tajima's D statistic [281] to compare the variant distribution of the different genomic elements with each other and to evaluate the neutral evolution hypothesis. Figure 4.5 illustrates the results for all SNP and indel types from the 1000G data. The results for the GoNL data are shown in Figure B.1.

Considering the 1000G data, median SNP densities were about 8–9 SNPs per kb for each genomic element and all variant types, see Figure 4.5A (leftmost group). Considering the GoNL data, median SNP densities were on average slightly lower (about 6–7 SNPs per kb) compared to 1000G for each genomic element and all SNP types, see Figure B.1. Figure 4.5A shows that protein–coding regions were conserved with a median SNP density of about 7 SNPs/kb for all SNP types. This is in accordance with earlier findings [250]. The large variance of the box plot for the 5′ UTR with a maximum SNP density value of about 35 SNPs per kb for 1000G data, (see Figure 4.5A) is due to the short 5′ UTR length of 230 bp on average (median 180 bp), compare with Figure B.2. In general, the data from 1000G and GoNL provided very similar results, see Figure 4.5A and Figure B.1.

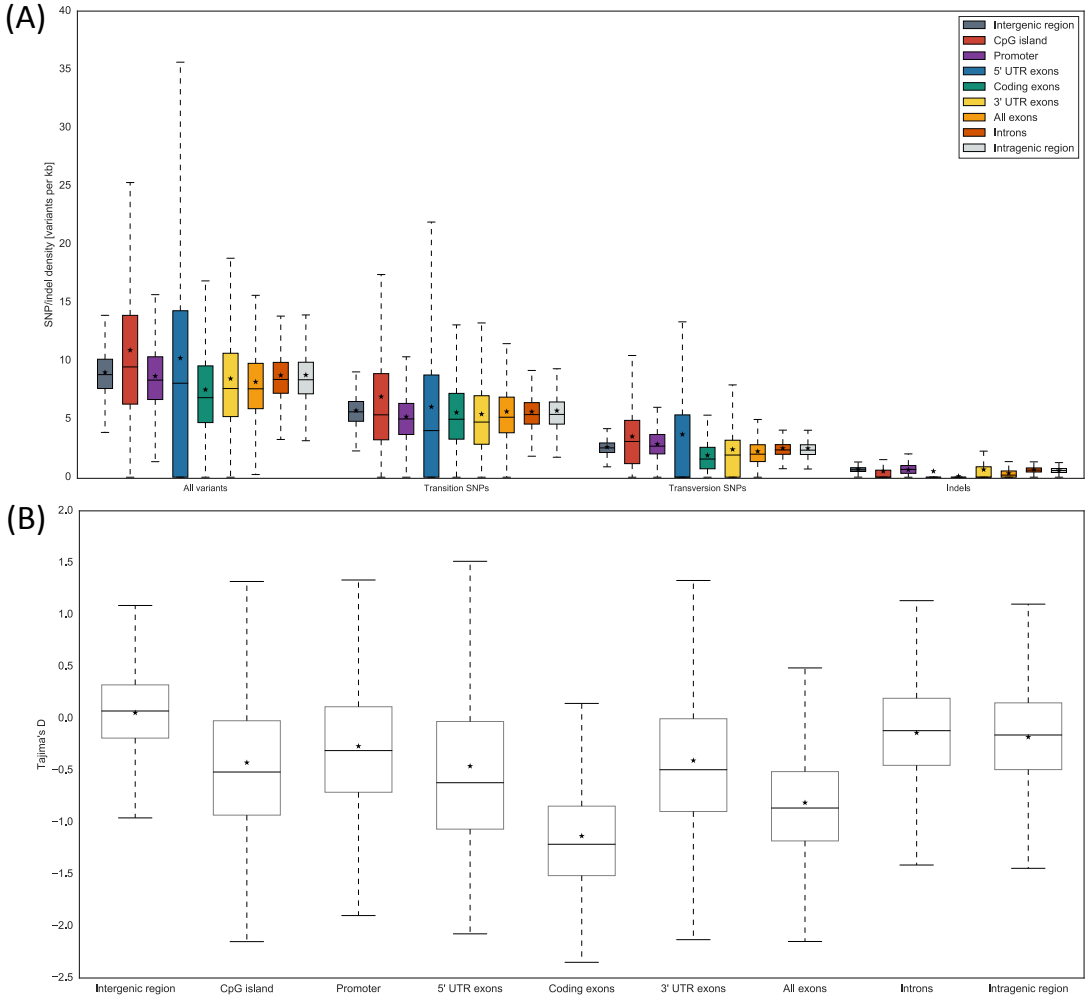

Figure 4.5: Mutations in key genomic elements considering the 1000G data. Shown are SNP and indel densities for all variant types and genomic key elements considering 1000G data (European cohort). (A) Distribution of SNP and indel densities for every gene, the horizontal line (–) represents the median value, the asterisk ($\star$) denotes the mean value. Note that the calculated SNP density is a function of the cohort size. In total, the 1000G European super population comprises 503 individuals. (B) Tajima's D statistic was applied to evaluate the neutral evolution hypothesis.

Indels were especially rare in coding exons since this type of mutation can cause frameshifts in the translated protein. The distribution of indels is shown as rightmost group in Figure 4.5A. Indel densities were significantly lower than SNP densities ($p << 1.4 \times 10^{-3}$). Especially CpG islands, 5′ UTRs, protein–encoding exons and 3′ UTRs showed a very low amount (median: 0.0) of indels, see Figure 4.5A (rightmost group). A decrease of deletions upstream of the transcription start site has been described in the literature [301]. To our knowledge, a similar effect has not been described yet for CpG islands. Indels might have more severe effects on transcription factor binding sites than base exchanges [302]. Hence, the low frequency of indels in CpG islands might be related to a strict conservation of functional sequences within this genomic (regulatory) element. The evolutionary stability of CpG islands is related to a high selective pressure on these regions, especially on CpG islands in the promoter regions of the mammalian genes [132].

To calculate Tajima's D statistic, we used VCFtools [72] and a bin size of 1 Mb to estimate the evolutionary pressure acting on the different sequence elements for 1000G data. Figure 4.5B

shows Tajima's D values for all genomic elements. Tajima's D values < 0 indicate a high number of rare alleles based on a growth in population size and/or purifying selection while Tajima's D values > 0 indicate a high number of alleles with average frequency. Intergenic regions (median: 0.07) were more or less neutral with values around 0. The smallest Tajima's D values were found in coding exons (median: –1.21), followed by all exons (median: –0.86), 5′ UTR exons (median: –0.62), CpG islands (median: –0.52), and 3′ UTR exons (median: –0.49). Thus, as expected, genetically important gene regions, such as coding exons or 5′ UTRs, were apparently subjected to purifying selection to preserve their functionality. A high conservation of protein–coding regions and a lower conservation of intergenic regions and introns were reported before [250]. Especially splice sites, that means exon–intron boundaries, were shown to be highly conserved [250].

In summary, we obtained a very similar picture on the conservation of genomic elements when either applying Tajima's D statistic or calculating simple SNP densities. Slight differences were observed when comparing CpG islands with intergenic regions: the SNP density was on average higher in CpG islands (9 vs. 11 SNPs per kb, see Figure 4.5A) while the respective Tajima's D index was smaller (median: –0.52 vs. 0.07, see Figure 4.5B). Moreover, considering 1000G and GoNL data provided very similar SNP distributions.

4.4.2 Mutation frequencies around the TSS and the CSS

SNPs and indels in promoter regions and 5′ UTRs may have direct effects on gene transcription and protein translation [182, 183, 300]. Thus, we investigated SNP densities and their distributions around transcription and translation start sites in more detail. Considering translation start sites, we also separately investigated canonical and alternative initiation sites and compared their conservation patterns with each other. This analysis is assumed to shed light onto the biological relevance of alternative mechanisms.

Mutations at transcription start sites peak at position –1

Figure 4.6A shows the local SNP density around the TSS in a range of $\pm$200 bp around the TSS. Both, 1000G and GoNL data show a decrease in SNP density at the TSS. The same pattern was observed before [250, 303]. Also, the indel frequency decreased slightly with the TSS as indels might perturb protein–coding regions more strongly compared to base exchanges. Nevertheless, the number of indels was in general very low such that we can only observe a slight indel depletion towards the TSS.

Next, we analyzed SNPs in direct vicinity to the TSS, see Figure 4.6B. The number of genes with at least one SNP in this sequence window is given in Table 4.3. In general, the number of SNPs directly downstream of the TSS was lower than directly upstream. Clearly noticeable is the very high peak at position –1. One might speculate that the last base in the intergenic region might simply be irrelevant for cellular function. It was recently reported that the mutation rate in human genomes is elevated at protein–bound DNA sites such as active transcription factor binding sites or nucleosome positions [304]. This was shown to be due to the interference of the nucleotide excision repair (NER) machinery and DNA–binding proteins that results in a decreased NER activity [304]. Thus, one might speculate that DNA–bound transcription factors could block repair enzymes resulting in higher mutation frequencies at these positions. However, only the position –1 deviates from the general SNP pattern. Thus, we analyzed the underlying DNA sequence at and around position –1 in more detail, also keeping in mind that some polymorphisms can occur more frequently (e.g. in the context of methylated CpGs) compared to other dinucleotides.

Figure 4.6C displays the dinucleotide distribution around the TSS while Figure 4.6D denotes the respective SNP distribution for individual dinucleotides. Figure 4.6C shows that the frequency of CpG dinucleotides (and CpA dinucleotides, see below) was increased at position –1 compared to the surrounding positions. A CpG at position –1 means that the C is located at position –1, while the respective G resides at position 1.

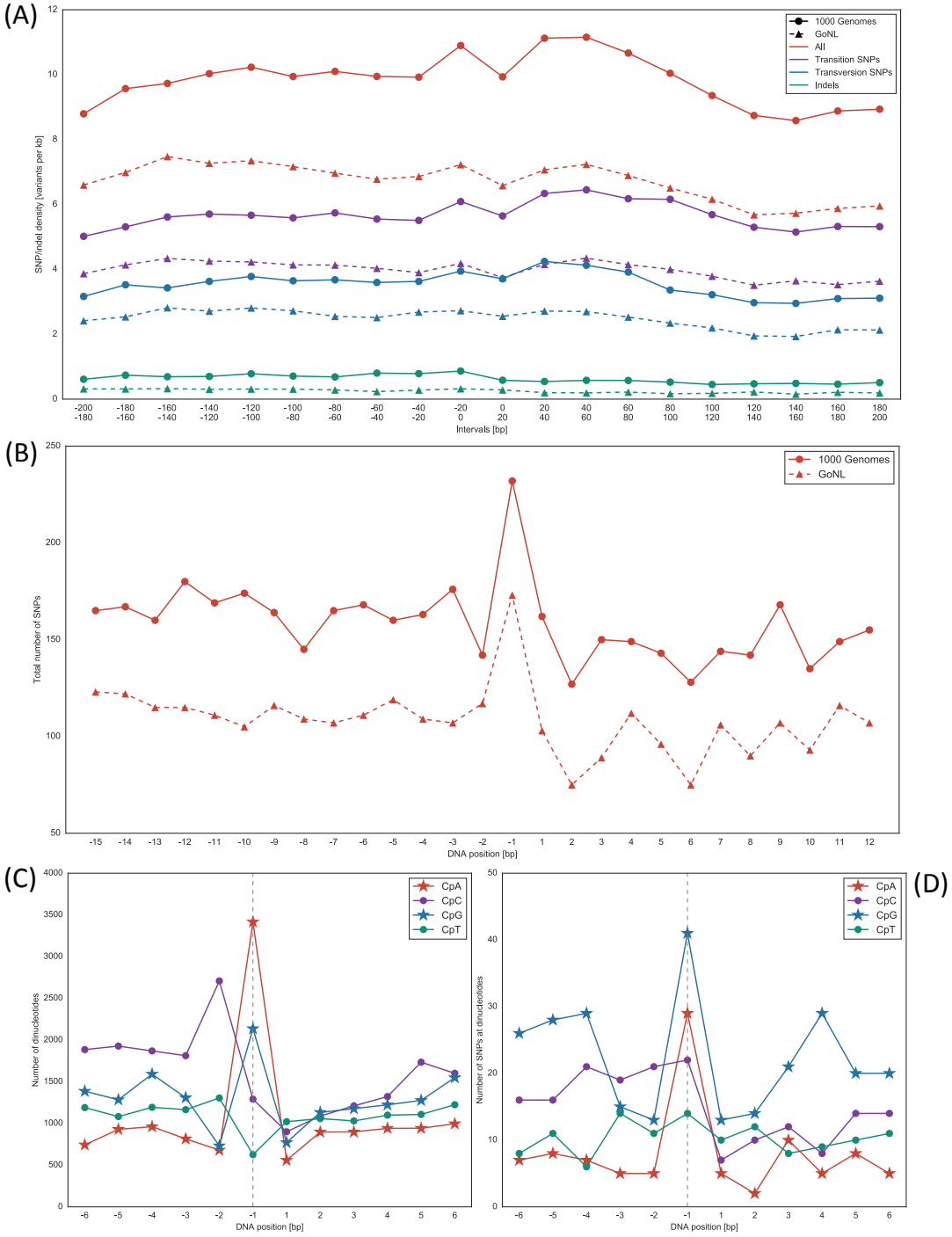

Figure 4.6: SNP and dinucleotide distributions around the TSS. (A) Average SNP and indel density (1000G and GoNL data) around the TSS (±200 bp) of all RefSeq genes. (B) SNP pattern in direct vicinity (–15 to +12) of the TSS considering 1000G and GoNL data. Position 1 denotes the first intragenic nucleotide. (C) Distribution of dinucleotides starting with cytosine in the flanking region of the TSS of all RefSeq genes. CpG and CpA dinucleotides are prevalent. (D) Number of SNPs (1000G data) at individual dinucleotides. The majority of SNPs resides at CpG dinucleotides.

In general, the number of CpG dinucleotides was increased around the TSS and peaks with the TSS, see Figure B.3. This symmetric CpG pattern was reported before [132]. Thereby, 2,136 out of 16,603 considered RefSeq genes (one sequence context was removed due to the

Table 4.3: Number of start sites in direct vicinity of TSS and CSS. Number of start sites (RefSeq and HEK293 genes) and SNPs (1000G and GoNL) in direct vicinity of TSS and CSS. We investigated genes with at least one SNP in this sequence window. For the CSS, we considered AUG as start site for the RefSeq dataset and alternative AUG and all near–cognate codons for the HEK293 dataset.

			Start sites harboring SNPs	**SNPs in these start sites**
TSS	RefSeq	1000G	3,777	4,472
		GoNL	2,681	3,043
CSS	RefSeq	1000G	3,319	3,819
		GoNL	2,277	2,519
	HEK293	1000G	1,276	1,487
		GoNL	838	942

occurrence of "N", i.e. an unknown base, in the hg19 reference genome) exhibited a CpG dinucleotide at position –1 corresponding to about 13% of all genes. Considering that there exist 16 dinucleotides, this number is twofold higher than expected randomly and about 50% higher than at the neighboring positions (see Figure 4.6C). CpA dinucleotides were found at positions –1/+1 for 3,415 out of 16,603 genes amounting to 21% of all RefSeq genes (threefold higher than expected). The distribution of all 16 dinucleotides in the enlarged window (–15 to +12) is shown in Figure B.4. Figure 4.6D displays the underlying SNP distribution (1000G data) for individual dinucleotides. At position –1, SNPs were most often found at CpG dinucleotides, followed by CpA dinucleotides. Thus, although there were more CpA than CpG dinucleotides at position –1 (see Figure 4.6C), SNPs were more frequently found in CpG context (mutation rates of $\frac{41}{2{,}136} = 0.02$ and $\frac{29}{3{,}415} = 0.008$, respectively). The frequencies for all 16 dinucleotides together with the underlying SNP pattern when considering 1000G and GoNL data can be found in Figure B.5 and Figure B.6, respectively.

The emergence of CpG (and CpA) dinucleotides especially at TSS position –1 was reported before [305, 306] and this partially explains the higher mutation rate detected at this position. Figure 4.6B shows that the total number of SNPs in 1000G data increases from about 160 per position to 230 at position –1. According to Figure 4.6D about 20 SNPs of this increase was found at CpGs and 20–25 SNPs at CpAs. The manifestation of polymorphisms at CpG sites, where cytosine is often methylated, is influenced by epigenetic marks [307]. Since deamination of 5–methylcytosine results in thymine which is a regular base of DNA, the resulting G–T mismatches are less efficiently corrected than G–U mismatches [128, 129]. It has been proposed that the methylation of cytosines is one cause of the general CpG depletion of vertebrate genomes, where only unmethylated CpG islands tend to have a CpG content that reflects the frequency of cytosine and guanine in the genome [130]. The evolutionary stability of CpG islands has been related to a high selective pressure on these regions, especially on CpG islands in the promoter regions of the mammalian genes [132]. As shown in Figure 4.6C, CpA dinucleotides were also highly enriched at position –1. Interestingly, non–CpG methylation was reported for different mammalian cell types (e.g. embryonic stem cells but also differentiated cells) and it has been suggested that methylation in non–CpG context is involved in the regulation of gene expression [104, 105, 106]. Beside CpG sites, methylation is most often found at CpA sites compared to CpT and CpC sites [105].

Finally, we analyzed the 2,136 genes with a CpG dinucleotide at position –1 by a GO term enrichment analysis using the DAVID–resource (version 6.8) [292]. As background gene set, we used all RefSeq genes considered here. DAVID default functional terms, Benjamini–Hochberg correction, and an EASE score threshold (corresponding to a modified Fisher exact p–value) of 0.05 were applied. The results of the functional annotation are displayed in Table 4.4. It is noteworthy that more than half of the inspected 2,136 genes are associated with the terms

"Phosphoprotein", "Alternative splicing" and "Protein binding", see Table 4.4. Another highly significant functional term was "Acetylation".

Table 4.4: Functional annotation results using the DAVID–resource. DAVID functional annotation [292] was applied to the 2,136 RefSeq genes with a CpG dinucleotide at TSS position –1. Duplicated terms from different databases were deleted and the one with smallest p–value was retained. Shown are terms with adjusted p–value of $p < 0.05$ (Benjamini correction).

	Term	# Genes	% Genes	Adjusted p–value
1.	Phosphoprotein	1110	52.0	3.5×10^{-13}
2.	Acetylation	507	23.7	2.1×10^{-9}
3.	Alternative splicing	1314	61.5	1.6×10^{-8}
4.	Cytoplasm	658	30.8	2.7×10^{-8}
5.	Nucleoplasm	412	19.3	7.2×10^{-7}
6.	Protein binding	1134	53.1	2.6×10^{-6}
7.	Protein transport	112	5.2	5.5×10^{-6}
8.	Nucleus	668	31.3	8.3×10^{-5}
9.	Vesicle–mediated transport	39	1.8	1.9×10^{-2}
10.	Cytoskeleton	177	8.3	8.6×10^{-4}
11.	Rab GTPase binding	34	1.6	1.0×10^{-2}
12.	Endocytosis	52	2.4	4.1×10^{-3}
13.	Cytosol	450	21.1	6.0×10^{-3}
14.	Guanine–nucleotide releasing factor	35	1.6	2.1×10^{-3}
15.	Cell cycle	106	5.0	3.0×10^{-3}
16.	Mitochondrion	162	7.6	4.1×10^{-3}
17.	Transport	271	12.7	1.4×10^{-2}
18.	Cell division	68	3.2	1.5×10^{-2}
19.	Electron transport	24	1.1	1.4×10^{-2}

We then repeated the GO term enrichment analysis at position –1/+1 for the remaining 15 dinucleotides and found only three other dinucleotides with significant GO term enrichments: CpA, TpA, and ApA, see Table B.1 to Table B.4. The genes harboring these three dinucleotides at position –1 were significantly associated with olfaction. None of these dinucleotides provided a similarly high statistical enrichment as genes with a CpG dinucleotide at this position. It was reported before that olfactory receptor genes are associated with high AT content in their promoter region [308]. Nevertheless, we only found the three mentioned dinucleotides to be significantly associated with olfaction instead of other A–T dinucleotide combinations. The occurrence of different promoter compositions suggest that specific dinucleotides right at the transcription start site might be involved in transcription (and translation) regulation. This involves particularly CpG and CpA dinucleotides, which can be additionally epigenetically modified by DNA methylation. The observation that the dinucleotide present at position –1 is linked with the activity and expression level of a TSS [305] together with dinucleotide associated GO terms indicates the involvement of specific dinucleotides right at the TSS in gene–group specific regulation.

Similar conservation patterns at canonical and alternative codons

Since changes at a single position in very close proximity to a start site can have an influence on translation initiation [182, 183, 189, 299], we especially focussed on the flanking region of canonical translation start sites and of alternative starts located in the 5′ UTR. First, we investigated the SNP and indel occurrence from –200 to +200 bp around translation start sites. Figures 4.7A and B shows that the SNP density decreased with the CSS for the annotated RefSeq start sites as well as for the alternative start sites located in the 5′ UTR that were identified by experimental ribosome profiling [43]. This depletion of SNPs in the coding region is most likely due to purifying selection. This effect was most prominent for indels, as this type of mutation has the potential to change the overall identity of proteins by shifts in the open reading frame. However, based on the overall small number of indels we can only observe a small decrease of the indel density towards the CSS. A depletion of indels especially in protein coding regions was observed before, while the observed Tajima's D values are also consistent with purifying selection, see Figure 4.5. SNP densities calculated based on the 1000G and GoNL data behave similarly. The vertical shift between those two major SNP projects can be attributed to the overall higher number of SNPs in the 1000G data.

Next, we focused on the start codon and the flanking region (–15 to +13) that has been shown to be crucial for translation initiation [182, 183, 189, 299]. In total, the RefSeq dataset provided 16,604 canonical translation start sites (i.e. flanking regions), whereas the number of alternative start sites located in the 5′ UTR amounted to 7,373 [43]. SNPs from 1000G and GoNL were mapped against these sequence contexts to determine their position in the interval from –15 to +13 with respect to the respective start site. Table 4.3 summarizes the number of translational sequence contexts in the RefSeq (AUG–only) and HEK293 datasets (AUG and near–cognate) that harbored SNPs as well as the number of SNPs residing at those start sites. Table 4.3 reveals that, on average, there was about one SNP per start site flanking region. In the next step, we investigated the distribution of these SNPs along the defined sequence window from positions –15 to +13.

Figure 4.7C shows the total number of SNPs in the flanking region of annotated and alternative start sites. We found that alternative start sites (AUG and near–cognate) located in the 5′ UTR showed similar conservation tendencies compared to annotated canonical start sites. As expected, the number of SNPs decreased remarkably with the start site, which reflects the importance to maintain translation start sites. To validate the statistical significance of this decrease in the number of SNPs at the start site, we performed a permutation test. We found that the drop at canonical start sites (RefSeq genes) was highly significant irrespective of the mutation dataset ($p < 0.01$), see Figure 4.7C. Considering alternative translation start sites in HEK293 cells, the decrease in the number of SNPs at the start codon was only moderate compared to the RefSeq starts and the p–values were not found to be significant. A negative peak can also be observed at position –3, which was shown to be crucial for translation initiation [182, 183]. However, this trend was also more prominent for annotated start sites compared to alternative start sites, see Figure 4.7C. Beside the start site itself and the prominent –3 position, several other positions also showed negatives peaks, for example positions 8 and 10. An experimental validation of the importance of these position is still lacking. Based on a simple permutation test, the drop at these positions was not statistically significant. Thus, canonical translation start sites are well conserved to preserve normal cell behavior. Alternative initiation start sites seem to be less conserved compared to canonical start codons. This could be due to the location of the considered alternative start sites in the generally less conserved 5′ UTR as well as be related to the general usage of these alternative sites. Non–canonical start codons can be used in specific cellular states, in specific cell types, or as cellular stress response [24, 181]. Moreover, the dataset might also contain some FP start sites and miss some TP start sites due to experimental and post–experimental (e.g. statistical evaluation) drawbacks [191].

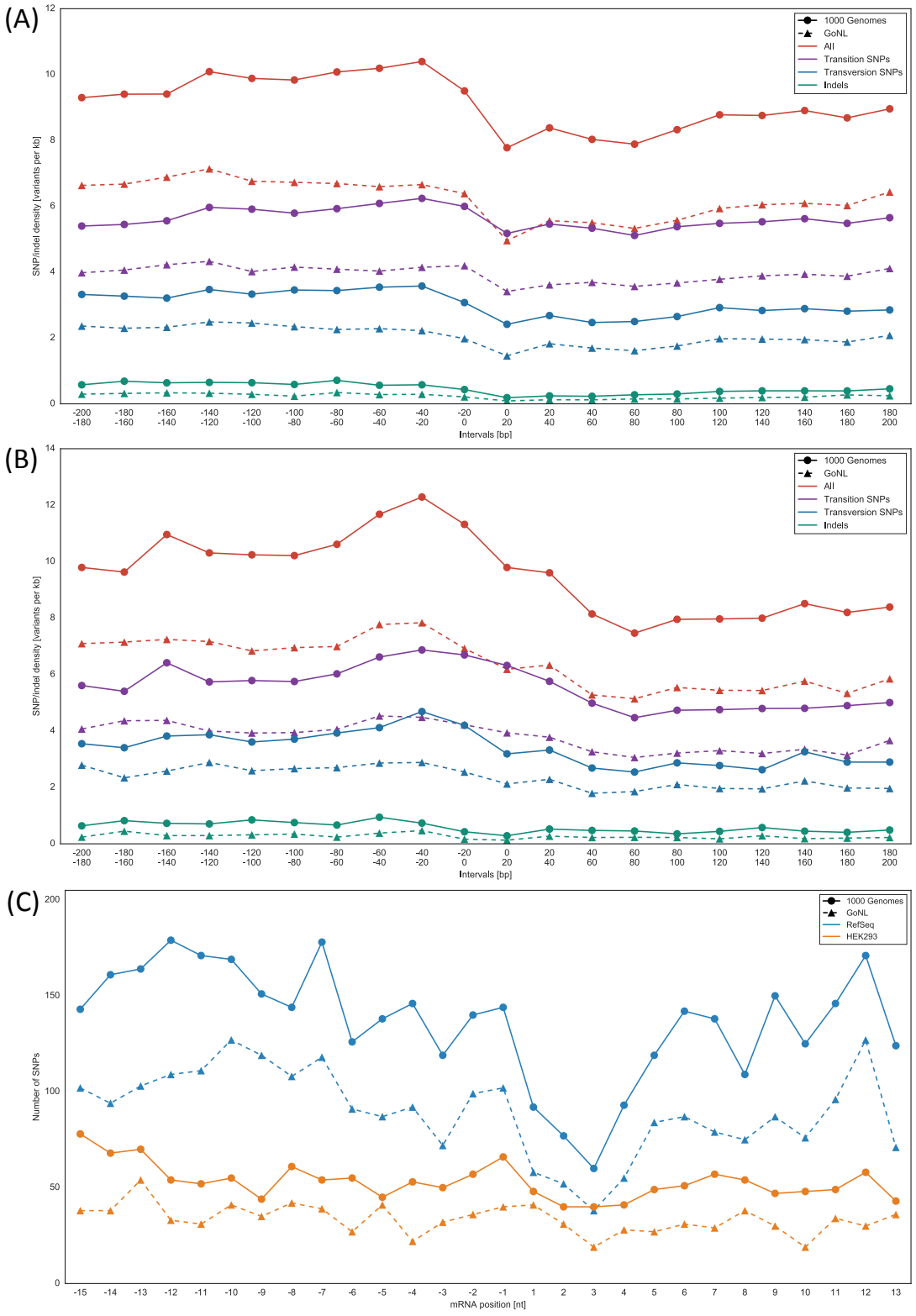

Figure 4.7: SNP distribution around the CSS. (AB) Average SNP density in a range of ±200 bp and 20 bp windows around the CSS (1000G and GoNL data). (A) Annotated translation start sites of (RefSeq genes). (B) Alternative translation starts detected by ribosome profiling applied to HEK293 cells. (C) SNP pattern in the flanking region (–15 to +13) of canonical and alternative starts. The applied permutation test provided the following p–values (curves from top to bottom): RefSeq+1000G: 0.0, RefSeq+GoNL: 0.0002, HEK293+1000G: 0.027, HEK293+GoNL: 0.244. With a significance threshold of $p < \frac{0.05}{4} = 0.0125$, the drop in the number of SNPs at the CSS is significant for canonical start sites.

4.5 Summary

Following the analysis of DNA methylation patterns and the development of *PreTIS*, we investigated genomic mutations in different regions of the human genome. As primary datasets, we used variations from the 1000 Genomes Project and the Genome of the Netherlands project. We investigated several genomic regions in the human genome, namely intergenic region, CpG islands, promoter, 5′ UTR, coding exons, 3′ UTR, all exons, introns and intragenic region. We reported conserved protein–coding regions and a significantly decreased indel density compared to SNPs. This is in accordance with findings from earlier studies. Based on our former projects, we were also interested in the distribution of mutations around transcription and translation start sites. Considering translation start sites, we separately analyzed alternative and canonical initiation sites. We observed an increased number of SNPs at position –1 relative to the transcription start site. Moreover, we observed that CpG and CpA dinucleotides were prevalent at this site, whereas an increased number of SNPs was found at CpGs. Next, using the DAVID–resource, we conducted a functional annotation analysis of genes with a CpG dinucleotide at TSS position –1. We found a significant enrichment of the terms "Phosphoprotein", "Alternative splicing", and "Protein binding". A repetition of this functional annotation for the remaining dinucleotides revealed significant enrichments associated with "olfaction" for genes with CpA, TpA, and ApA dinucleotides at position –1. We propose that considering an increased number of SNPs at CpG dinucleotides at this site, the susceptibility of a methylated cytosine to mutate, and the results of the functional annotation might indicate a gene–specific regulatory signal. When investigating the distribution of SNPs around translation start sites, we observed similar conservation patterns of canonical and alternative start sites. Applying statistical permutation tests, we calculated whether the detection of a negative peak, in this case the decreased number of SNPs, is statistically significant. By doing so, we found a statistical significance for the negative peak at canonical start sites. Although the decrease was not significant for alternative start sites, a general tendency of a decreased mutation rate was observed as well. This indicates the importance of alternative translation start sites for cellular function in the human genome.

CHAPTER 5

Automated analysis of mutations in gene regulatory networks

This chapter describes our software package *MutaNET*, which was developed to automatically score individual mutations in key genomic elements, such as transcription factor binding sites, regarding their influence on cellular function. The integration of an underlying gene regulatory network helped to assess the potential global impact of mutations on gene expression. The following chapter presents and extends our publication "MutaNET: a tool for automated analysis of genomic mutations in gene regulatory networks. Markus Hollander, Mohamed Hamed, Volkhard Helms, and Kerstin Neininger. *Bioinformatics*, doi:10.1093/bioinformatics/btx687, 2017". Markus Hollander implemented *MutaNET* software during his Bachelor's thesis in our group under my supervision. A variant calling pipeline based on next–generation sequencing paired–end reads was developed by Mohamed Hamed during his PhD in our group and first applied in [61]. This pipeline was then reimplemented by me in the Python programming language to be applicable in *MutaNET*. The *MutaNET* software for macOS, Linux, and Windows operating systems can be downloaded from `https://sourceforge.net/projects/mutanet/`. A web page that gives some background information together with a step–by–step tutorial is available at `http://service.bioinformatik.uni-saarland.de/mutanet/`.

5.1 Prerequisites

In this project, we automated the genome–wide analysis of mutations and their potential influence on cell function. For this, we defined scores that estimate the impact of individual mutations on protein function and regulator binding. Information on an underlying gene regulatory network was integrated as well, which helps to estimate the global effect of specific mutations. Prior to the mutation analysis, an embedded variant calling pipeline can be used to call mutations from paired–end reads. Our tool was then applied to decipher intrinsic antibiotic resistance mechanisms of *Escherichia coli* and *Staphylococcus aureus* bacterial strains. In the following, next–generation sequencing, the *MutaNET* embedded variant calling pipeline and the scoring schemes, gene regulatory networks, and prokaryotic genome regulation are presented. Moreover, *Escherichia coli* and *Staphylococcus aureus* bacteria are shortly introduced together with an overview on antibiotic resistance mechanisms.

5.1.1 Next–generation sequencing and variant calling

The international human genome project was successfully terminated in 2003 and therewith opened new challenges in sequencing technologies [233, 234]. Over the years, DNA sequencing technologies improved dramatically and entailed significant cost reduction and expansion of possible applications [309, 310]. In 2009, the costs for whole–genome sequencing of one human genome amounted to roughly $200,000 [311]. Within the next five years these costs significantly

dropped to about $1,000 per human genome [309, 311]. The initiative to achieve this $1,000 goal was referred to as the "$1,000 genome" [310, 312, 313].

Next–generation sequencing: a '$2 billion market'

The term next–generation sequencing (NGS) comprises modern sequencing technologies and emerged when the first high–throughput sequencing platform was released in the 2000s [309, 314]. NGS technologies comprise Illumina/Solexa sequencing, Roche 454 pyrosequencing, or SOLiD sequencing [309]. An increasing demand of NGS application constitutes whole–genome sequencing (WGS) projects due to its broad usability to solve various biological problems such as mutation analysis or epigenetic modifications [309]. For instance, the 1000 Genomes Project is based on WGS to analyze human genetic variation [57].

The company Solexa, which was founded in 1998, released their first sequencer in 2006 and constituted the basis of the well–known company Illumina [315]. In 2007, Illumina took over Solexa to integrate and sell their genome sequencing technologies [193, 315]. Solexa, as a hitherto industry leader in next–generation sequencing systems, had the aim to revolutionize these technologies for low–cost and time–efficient (whole–genome) sequencing of single DNA molecules and therefore entered this "$2 billion market" [316]. Illumina is the current market leader of accurate and low–cost sequencing technologies [309, 317]. Illumina provides various solutions for WGS, exome–sequencing, ChIP–seq to decipher protein–DNA interactions, or RNA sequencing (RNA–seq) [309]. The Illumina HiSeq X system is able to sequence more than 18,000 human genomes within a year (30x coverage) making it the sequencing platform with highest throughput to date [318].

Illumina provides technologies for single– and paired–end sequencing [319]. Single–end sequencing enables the sequencing of DNA fragments from one end, whereas in paired–end sequencing, DNA fragments are sequenced from both ends thus resulting in read pairs [320]. Paired–end sequencing increases mapping quality and simplifies the correct detection and alignment of repetitive elements or structural rearrangements such as insertions and deletions. Figure 5.1 illustrates the idea of paired–end sequencing.

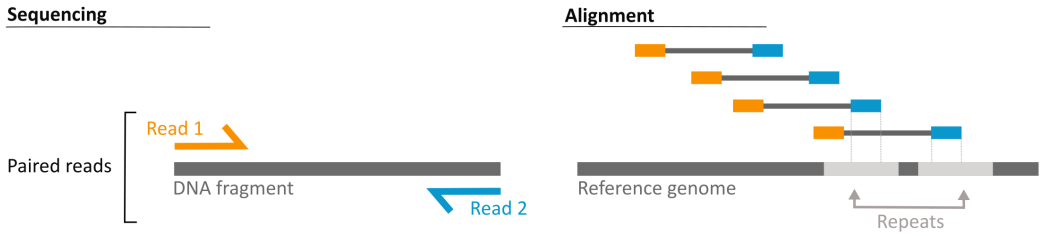

Figure 5.1: Paired–end DNA sequencing. Paired–end sequencing, in contrast to single–end sequencing, enables to sequence a DNA strand from both directions. A consideration of the distance between paired reads improves the alignment of repetitive elements against a reference genome. The figure was adapted from [319].

Moreover, there is a differentiation between short–read NGS and long–read NGS, both applied for different applications [309]. Short–read NGS techniques are categorized into sequencing by ligation or by synthesis. Thereby, a sequence read is defined as a nucleotide stretch from an individual DNA molecule [309]. Long–read NGS is, for instance, necessary to completely cover long repetitive regions to enable a precise genomic localization and determination of their length. Moreover, long–read NGS can span entire gene transcripts and thus aid in the exact analysis of mRNA transcripts to unveil exon positions and identify alternative transcripts. The development of long–read sequencing also allowed an extension of the GRCh37 reference genome by closing previously existing gaps [321]. NGS became a routine and with this continuous improvement, sequencing technologies also found their way to clinical applications [309].

File formats and quality control

A next–generation sequencing variant calling pipeline was developed in our group by Hamed et al. [61] to call SNPs and indels from given paired–end reads in FASTQ format. For compatibility reasons, this pipeline was reimplemented by me in Python programming language and subsequently embedded in the *MutaNET* software package. A description of the FASTQ file format, which is an extension to the well known FASTA format, together with an explanation of the Phred quality score is given below.

FASTA file format The FASTA format is commonly used to store nucleotide and peptide primary sequences. Sequences are presented using one–letter codes for nucleotides from the alphabet $\{A, C, T, G\}$ and amino acids from the alphabet $\{G, A, V, L, I, P, ...\}$, see Table 1.2. FASTA files maintain a specific (simple) format, starting with a greater–than (">") symbol followed by a single line description of the stored sequence data. The description line is then followed by (several) lines of nucleotide or protein sequences. For instance,

```
>TRPV6|Transient receptor potential cation channel subfamily V member 6
MGPLQGDGGPALGGADVAPRLSPVRVWPRPQAPKEPALHPMGLSLPKEKGLILCLWSKFC
RWFQRRESWAQSRDEQNLLQQKRIWESPLLLAAKDNDVQALNKLLKYEDCKVHQRGAMGE
TALHIAALYDNLEAAMVLMEAAPELVFEPMTSELYEGQTALHIAVVNQNMNLVRALLARR
ASVSARATGTAFRRSPCNLIYFGEHPLSFAACVNSEEIVRLLIEHGADIRAQDSLGNTVL
HILILQPNKTFACQMYNLLLSYDRHGDHLQPLDLVPNHQGLTPFKLAGVEGNTVMFQHLM
QKRKHTQWTYGPLTSTLYDLTEIDSSGDEQSLLELIITTKKREARQILDQTPVKELVSLK
WKRYGRPYFCMLGAIYLLYIICFTMCCIYRPLKPRTNNRTSPRDNTLLQQKLLQEAYMTP
KDDIRLVGELVTVIGAIIILLVEVPDIFRMGVTRFFGQTILGGPFHVLIITYAFMVLVTM
VMRLISASGEVVPMSFALVLGWCNVMYFARGFQMLGPFTIMIQKMIFGDLMRFCWLMAVV
ILGFASAFYIIFQTEDPEELGHFYDYPMALFSTFELFLTIIDGPANYNVDLPFMYSITYA
AFAIIATLLMLNLLIAMMGDTHWRVAHERDELWRAQIVATTVMLERKLPRCLWPRSGICG
REYGLGDRWFLRVEDRQDLNRQRIQRYAQAFHTRGSEDLDKDSVEKLELGCPFSPHLSLP
MPSVSRSTSRSSANWERLRQGTLRRDLRGIINRGLEDGESWEYQI
```

represents the TRPV6 protein sequence retrieved from UniProt (protein ID: Q9H1D0) [227]. The header information consists of gene and protein name followed by the protein sequence of TRPV6 isoform 1 with a total length of 765 amino acids.

FASTQ file format FASTQ files are an extension of FASTA files with additional information on quality scores for every base. Each entry in a FASTQ file is composed of four lines, whereas each line provides different information [322]. The first line presents a sequence identifier (starting with "@"), followed by optional information such as the instrument name or flowcell lane. The second line stores the sequence. The third line starts with a "+" followed by optional repetition of the first line, which is often omitted due to memory space. The forth line represents Phred quality scores for every base using ASCII characters (see below). For instance, the following information was extracted from a FASTQ file that was analyzed in [61]:

```
@M00214:74:000000000-A3DA3:1:1101:13948:2193 1:N:0:1
AACATTGTTATTTAACAAAATTATGTTAAAATTTAGCATTATAAAAGATACAAATCAATGAC...
+
ABBBBFFFFFFFGGGGGGGGGGHHHHHHHHHHGHHFGHHFHGHHHHHGGEHFCGGHDGFBDH...
@M00214:74:000000000-A3DA3:1:1101:18114:2205 1:N:0:1
TTCAAAATTCATTTCTTGGAGATGATTGATGCGTTGAAATATAACTAATTGCCATAATACTT...
+
1AAAAFFF1DFFGGGGGGGGGG1GF3DFBGHHCFFGHHHBG2GHFBHGHGFHHHHHHHHHGG...
@M00214:74:000000000-A3DA3:1:1101:19506:2206 1:N:0:1
GTATTTACAACAGAATATTCGGTTCGTACTGCCATGGAAGCTGTTTATCAATTACTAAATAT...
+
AABBBFFFFFFBGGGGGGGGGGGGFEEGGHHHHHHHFHHFEGHFBEGFHHHHGHHHHHHHHH...
...
```

FASTQ files are supported by common NGS technologies and are thus the format of choice to store sequencing information and enable across–platform data handling. Using software tools such as the BWA tool [73, 74], FASTQ sequence reads can be mapped against a reference genome for further downstream analysis.

Phred quality score Phred quality scores find application in high–throughput DNA sequencing by assigning a quality measure to individual base calls [323]. This allows to assess DNA sequence quality and sequencing methods. A Phred quality score, which is also known as Q score, is defined as follows

$$Q = -10\ log_{10}\ P \tag{5.1}$$

with the error probability P that a base call is incorrect. Thereby, high quality scores correspond to low error probabilities and thus indicate a higher (more reliable) quality base call. Table 5.1 illustrates the logarithmic relationship between quality scores and error probabilities. For instance, a Phred base quality value of Q30 corresponds to an error probability to detect one incorrect base call out of 1000 calls. This means that a sequence read with a of length 1000 bp probably contains one error. The corresponding base call accuracy of 99.9% is then defined as the probability that a base call is correct. Note that the Q30 value is often used as the default minimum value to guaranty reliable sequencing quality.

Table 5.1: Relationship between Phred quality score and error probability. Phred scores allow assessment of base calling quality from DNA sequencing. Thereby, the Phred quality score Q and the error probability P to detect an incorrect base are dependent on each other. Base call accuracy is defined as probability to call a correct base. Quality scores are calculated using Equation 5.1. For instance the value Q30 results from $Q = -10\ log_{10}(0.001) = 30$.

Phred score Q	Error probability P	Base call accuracy
0	10 in 10 = 1	0%
10	1 in 10 = 0.1	90%
20	1 in 100 = 0.01	99%
30	1 in 1000 = 0.001	99.9%
40	1 in 10,000 = 0.0001	99.99%
50	1 in 100,000 = 0.00001	99.999%
60	1 in 1,000,000 = 0.000001	99.9999%

Thus, a Phred score can be seen as a probability prediction to detect incorrect base calls. With use of the negative decadic logarithm, very small error probabilities can be displayed reasonably. Quality scores are assigned during the sequencing procedure and are stored within the FASTQ files.

NGS variant calling pipeline

The variant calling pipeline mentioned before is built up from a series of several software packages. First, paired–end sequence reads are mapped to a given reference genome using the BWA tool [73, 74]. Next, duplicated PCR reads and reads with low quality (Phred quality score <30) are removed by applying SAMtools [75, 76]. Subsequently, the final alignments are sorted using SAMtools. SNPs and indels are then called by VarScan2 [77]. This pipeline is embedded in our tool *MutaNET* to call SNPs and indels from a reference genome prior to further statistical analyses. The variant calling pipeline is displayed in Figure 5.2 and explained hereinafter.

Prior to executing this pipeline, raw sequencing reads can be quality checked using the FastQC software package [324].

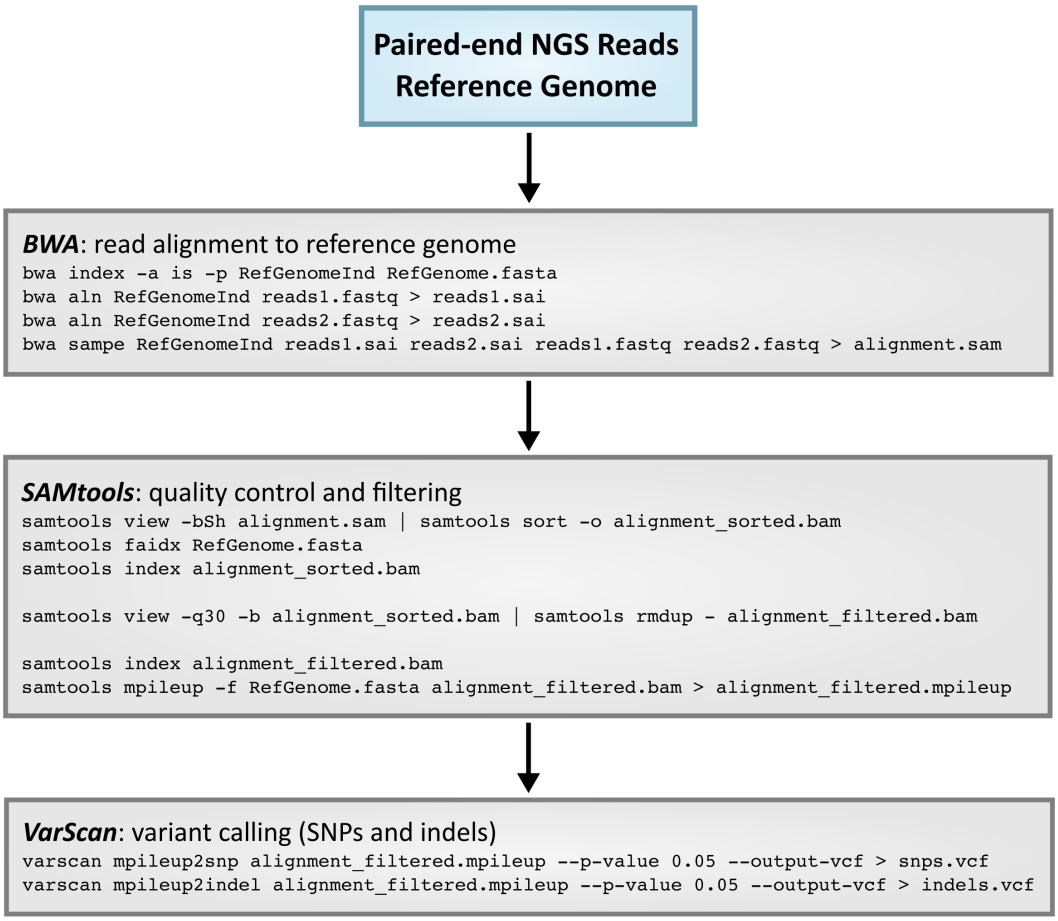

Figure 5.2: NGS variant calling pipeline. The variant calling pipeline consists of three essential steps: paired–end read alignment to a given reference genome, quality control and filtering, and the final variant calling (SNPs and indels). To achieve this, Burrows–Wheeler Alignment [73, 74], SAMtools [75, 76], and VarScan2 [77] are executed consecutively. Individual commands together with the parameters and options used here are explained in the main text.

BWA The Burrows–Wheeler Alignment (BWA) tool is applied to map sequence reads against a (large) reference genome [73, 74]. It implements algorithms for both, short (<100 bp) [73] and long (< 1Mb) [74] read alignments. As the name suggest, BWA takes advantage of Burrows–Wheeler transformation that enables a fast and efficient mapping. To tolerate base mismatches and gaps, BWA provides solutions for inexact sequence matching [73]. Thereby, one differentiates between mismatches, gap opening and closing to be relevant for practical applications. The results of a BWA call are reported in the common SAM file format [75].

The Burrows–Wheeler transform is a widely used data compression and indexing algorithm [325]. The algorithm was initially developed by D. J. Wheeler in 1983 and finally published in 1994 together with M. Burrows [325]. The algorithm is composed of three main steps: string rotation, lexicographical sorting, and extraction. More precisely, given a string S, all possible (distinct) cyclic rotations of S are generated, which is followed by sorting them in lexicographical order. The Burrows–Wheeler Transformation $BWT(S)$ is then constructed by concatenating the last characters in the last column of each cyclic string rotation. An example

with S = "papaya$" is given in Figure 5.3. As it is common for suffix arrays and suffix trees, a termination character $ is appended to S. Note that the Burrows–Wheeler matrix and suffix array are constructed similarly. A suffix array (SA) represents all possible suffixes of a given string S with maintained alphabetical order, see Figure 5.3. The lexicographical sorting step ensures that rotated strings with similar suffixes are arranged next to each other enabling fast lookup. As example, the localization of the substring "pa" in "papaya" and the determination of occurrences together with the exact position results in the SA interval $(4, 5)$ with the respective SA values 0 and 2 that give the 0–based positions of the substring occurrences in the given sequence S, see Figure 5.3. Note that the transformations $S \longleftrightarrow BWT(S)$ are reversible.

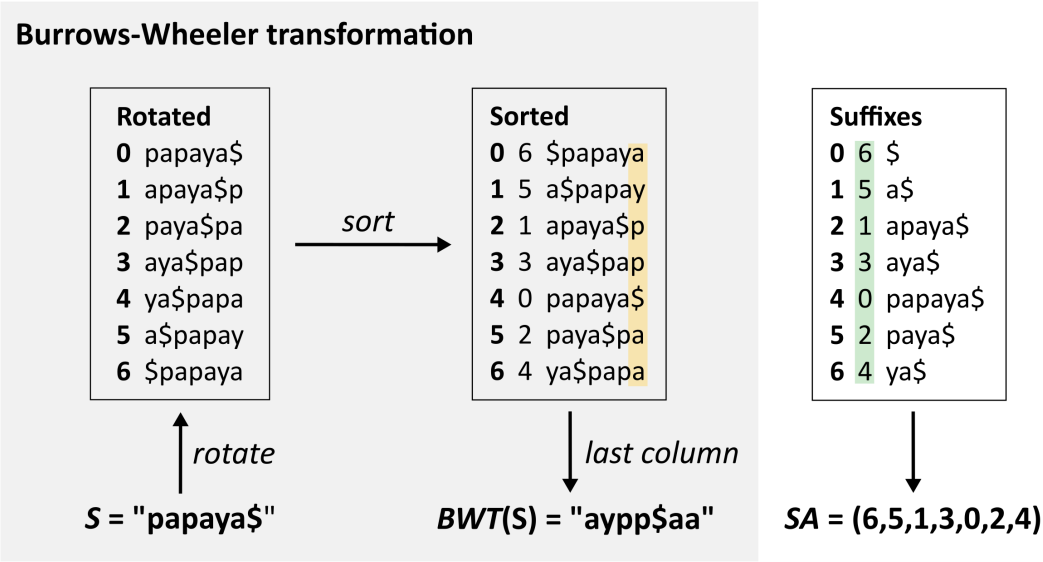

Figure 5.3: Example illustrating Burrows–Wheeler transformation. Given a string S, the Burrows–Wheeler transformation $BWT(S)$ is generated by constructing all cycling rotations of S followed by lexicographical sorting. $BWT(S)$ then corresponds to the last character of each sorted rotation in the Burrows–Wheeler matrix. Here, as an example the string was given as S = "papaya$" resulting in the $BWT(S)$ = "aypp$aa". The termination character $, which appears lexicographically prior to all other characters, was added to the end of S as it is commonly done when applying suffix array or suffix tree algorithms. The original string is given in row 4 of the final matrix. Note that construction of the Burrows–Wheeler matrix and a suffix array (SA) is related. The corresponding SA is shown on the right and demonstrates the similarity to the Burrows–Wheeler matrix. The SA interval of the substring "pa" in "papaya" is found as $(4, 5)$ with the respective SA values 0 and 2 that mark the substring positions in the string S. Note that indices are 0–based.

Coming back to BWA as a read aligner, the alignment of reads against a reference genome corresponds to the lookup of substrings in a given sequence S and determination of the appropriate SA intervals and thus the exact (genomic) position. When using BWA as a read aligner, the FM–index [326] of a given reference genome sequence must be generated using the `index` command in a first step. This transformation allows to store the complete reference sequence in a compressed format to index large sequences for a fast localization of substring positions and occurrences and thus serves as basis for efficient read mapping against a reference sequence.

The index construction is mainly based on the principles used in Burrows–Wheeler transformation and suffix arrays. Lookup speed and memory space are both crucial when working with large strings such as the human genome reference sequence consisting of almost 3 billion bases [234]. Note that in the following paired–end reads are denoted as `reads1` and `reads2`. The respective reference genome is referred to as `RefGenome`. Given a reference genome

`RefGenome` in FASTA format, a BWT index file with the filename `"RefGenomeInd"` (`-p` option) is generated using

```
$ bwa index -a is -p RefGenomeInd RefGenome.fasta
```

with the default index construction algorithm "IS" to create a suffix array (`-a` option). To generate a sequence alignment, the suffix array positions of the given paired–end reads are then determined with

```
$ bwa aln RefGenomeInd reads1.fastq.gz > reads1.sai
$ bwa aln RefGenomeInd reads2.fastq.gz > reads2.sai.
```

Finally, mapped paired–end reads are incorporated into one SAM file format for quality checks and filtering using

```
$ bwa sampe RefGenomeInd reads1.sai reads2.sai reads1.fastq.gz
   reads2.fastq.gz > alignment.sam
```

Application of BWA is followed by the usage of SAMtools [75, 76] for mapping quality control and filtering.

SAMtools The second step comprises alignment quality control and read filtering, which is enabled by SAMtools [75, 76]. The previously created alignment file is first converted to BAM format. File conversions, for instance from SAM to BAM file format, as well as sequence indexing are essential such that the sequentially connected command line tools are applicable and fast access is enabled. SAM file format is a widely used format for the storage of (short or large) read alignments against a reference genome [75]. BAM files store the same information but in a compressed binary format [75]. Application of file conversion, sorting of alignment coordinates, and indexing of the sorted BAM file is achieved by applying

```
$ samtools view -bSh alignment.sam | samtools sort -o alignment_sorted.bam
$ samtools faidx RefGenome.fasta
$ samtools index alignment_sorted.bam.
```

These operations are the basis for subsequent quality control. Note that the sorting step is required prior to indexing. Next, duplicated PCR reads (`rmdup`) and reads with low quality (Phred quality score < 30) are removed via

```
$ samtools view -q30 -b alignment_sorted.bam |
  samtools rmdup - alignment_filtered.bam
$ samtools index alignment_filtered.bam.
```

An indexing step is again necessary for further processing and generation of the final `mpileup` file. A SAMtools `mpileup` file is created by applying

```
$ samtools mpileup -f RefGenome.fasta alignment_filtered.bam
  > alignment_filtered.mpileup.
```

The usage of UNIX piping by "|" avoids the generation of temporary files.

VarScan2 As final step, VarScan2 [77] is applied to call variants (SNPs and indels) from the sequence alignment `mpileup` file and store them in VCF file format whilst executing

```
$ varscan mpileup2snp alignment_filtered.mpileup --p-value 0.05
  --output-vcf 1 > snps.vcf
$ varscan mpileup2indel alignment_filtered.mpileup --p-value 0.05
  --output-vcf 1 > indels.vcf.
```

VarScan2 is a software tool that implements a heuristic algorithm to detect somatic variants (SNPs and indels) and copy number variations from sequencing data [77]. VarScan2 was published in 2012, while a first version of VarScan was released a few years earlier in 2009 [327]. To apply VarScan2, read alignment information is needed in SAMtools (m)pileup file format. Variants are called based on an initial evaluation of allele frequency, read coverage, Phred base quality, and statistical testing. When all required threshold criteria for coverage and Phred base quality are fulfilled, the detected read bases at individual positions determine the genotype. Allele frequencies of variants are derived from observed read counts. The default threshold values are given as a minimum coverage of at least three reads, a Phred base quality score ≥ 20, 8% allele frequency, and a p–value threshold of $p < 0.05$ for statistical testing. For evaluation purposes, the authors of VarScan applied their tool to detect mutations in matched normal–tumor ovarian cancer samples that were retrieved from The Cancer Genome Atlas (TCGA) [328]. Thereby, samples were based on exome sequencing. VarScan2 variant detection showed 93% sensitivity and 85% precision for the reported single variants.

5.1.2 Mutation analysis using scoring schemes

Applying *MutaNET* software, mutations are assigned to the genomic regions using in–house scripts that are analogous to BEDTools [78]. The considered genomic regions comprise coding region, promoter region, and transcription factor binding sites (TFBSs). Note that a mutation can be associated with multiple genes or genomic regions due to overlaps. Since mutations at different positions throughout a genome sequence have different influences on gene function and regulation, we implemented appropriate scoring schemes separately for the differing regions. In the following, these scoring schemes for mutations in transcription factor binding sites and in coding regions are introduced.

Scoring mutations in transcription factor binding sites

The human genome is mainly composed of non–coding DNA, like regulatory elements or non–coding RNAs, rather than coding DNA sequences that serve as blueprint for proteins [329]. Thereby, the number of regulatory sites is assumed to exceed the number of genes [329]. As mentioned beforehand in Section 4.1.2, mutations can impact (disease) phenotypes heavily when located at crucial positions in non–coding regulatory sites or in coding regions. Hence, an estimation of the impact of mutations in non–coding elements is also of considerable importance to explain phenotypic variation. Following this, it was reported that cancer development is associated with variations in regulatory regions that can entail a disturbed gene regulation and thus affect gene expression [10, 146, 300].

Melton et al. [300] investigated the influence of individual mutations in TFBSs on regulator binding in several cancer subtypes. Based on observed mutations in TFBSs, they report a positive selection of these variations in regulatory regions and noticed that two scenarios are possible: the loss of a TFBSs that hinders transcription factor (TF) binding but also an establishment of novel binding sites TFs can interact with. Their approach aims at scoring the impact of individual point mutations in TFBSs on regulator binding [300]. Thereby, their methodology is based on a statistical comparison between observed and introduced random mutations. Their method is shown in Figure 5.4A. First, TF motif sequences were aligned with publicly available position weight matrices (PWMs) of TFBSs. Following this, they introduced a random mutation for every observed mutation, thereby maintaining the probabilities to convert one base to

another. Finally, matching scores of the wild–type sequence, the observed mutated sequence, and the randomly mutated sequence with the determined TF PWM were calculated.

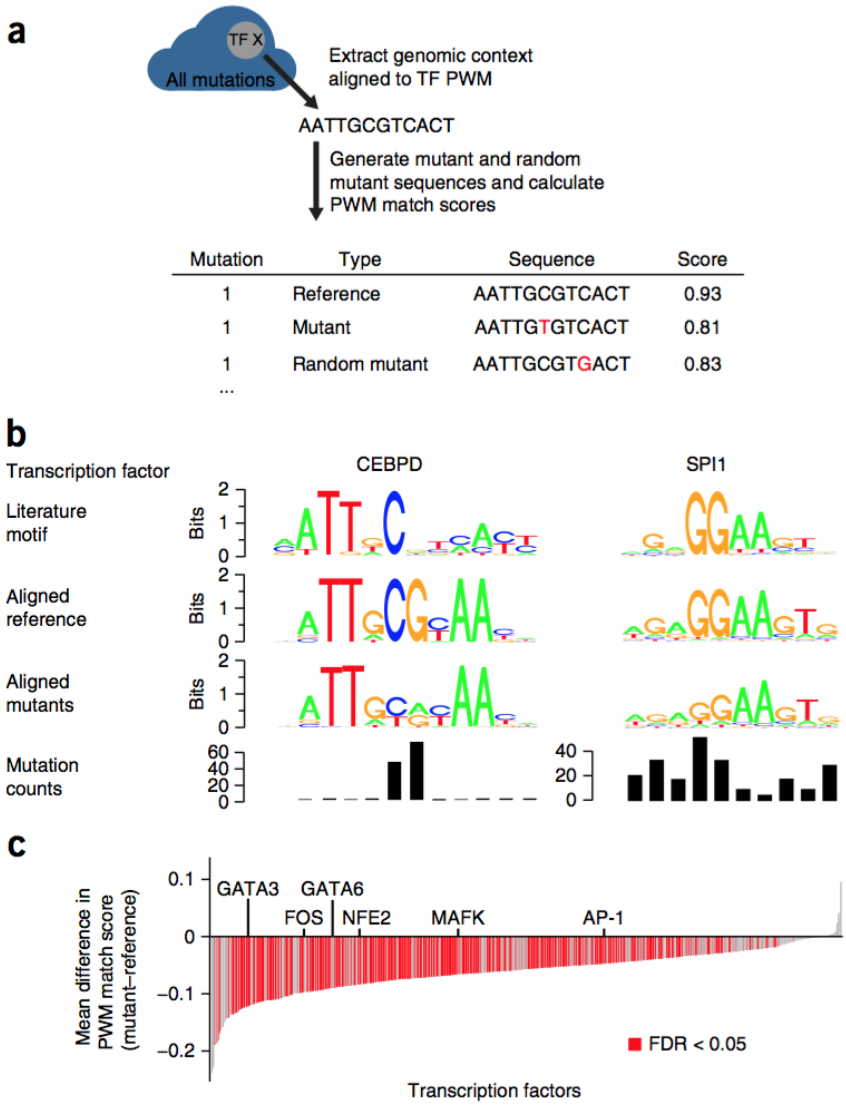

Figure 5.4: Scoring of mutations in TFBSs. Melton et al. [300] defined the following methodology to score individual mutations in TFBSs. (A) Binding sites were mutated randomly according to observed mutations. PWM binding site scores were subsequently calculated to estimate the strength of a mutation. (B) Sequence logos enable a comparison between wild–type, mutated, and randomly mutated motifs. The individual mutation count suggests that a mutation at the central CG site of the *CEBPD* motif highly affects TF binding. (C) Mean differences between PWM matching scores of the mutant and the reference show that the matching scores decrease significantly for mutated sites. Wilcoxon rank–sum test and FDR < 0.05 were applied. The figure was taken from [300].

Next, they illustrated their method using two binding sites: *CEBPD* and *SPI1*. Sequence logos of *CEBPD* and *SPI1* binding site alignments with the respective TF motif PWM are depicted in Figure 5.4B. Considering *CEBPD*, mutations were prevalent at a reference CG site located in the motif centre. On the other hand, mutations in the *SPI1* motif were distributed more equally throughout the binding site, see Figure 5.4B. Comparing the two mutational landscapes, the prevalent mutations in *CEBPD* suggest a specific mutational selection that has an impact on TF binding, thereby considering both, the inhibition of binding as well as a novel binding of

alternative TFs. Due to more equally distributed mutations in the binding motif of *SPI1*, this assumption might not hold true for this binding site [300]. The computation of PWM matching scores for every TFBS can support the detection of mutated binding sites with significantly decreased matching scores compared to their wild–type sequence, see Figure 5.4C. This approach by Melton et al. [300] to estimate the impact of mutations in TFBSs is embedded in our *MutaNET* software, whereby publicly available PWM of known TFBSs must be given as input. In summary, the impact of mutations in regulatory regions such as TF binding sites might be underestimated although their global influence on cellular function can be extensive.

Scoring mutations in coding regions

Considering coding mutations, *MutaNET* differentiates between synonymous, missense, nonsense, readthrough, and frameshift mutations. Since the position in the protein highly influences the impact of a mutation, protein domain(s) are incorporated in the analysis as well. The effect on the amino acid sequence is automatically assessed using a pairwise sequence alignment of the reference (R) and mutated (M) amino acid sequence together with an amino acid substitution matrix S (here: PAM10). The overall substitution score $Score_{cod} \in [0, 1]$ is computed as

$$Score_{cod} = \frac{\sum_{i=1}^{N} S_{R[i],M[i]} - \sum_{i=1}^{N} \min_{j \in AA} \{S_{R[i],j}\}}{\sum_{i=1}^{N} \max_{j \in AA} \{S_{R[i],j}\} - \sum_{i=1}^{N} \min_{j \in AA} \{S_{R[i],j}\}}$$

with the matrix entry $S_{R[i],M[i]}$ of the reference and mutated amino acid AA at sequence position i, and the aligned sequence length N. Thereby, a decreased score $Score_{cod}$ is associated with a probable higher impact of the mutation due to an increased deviation of the mutated amino acid sequence from the reference sequence.

5.1.3 Gene regulatory networks

Understanding how genes are regulated and therewith how different expression patterns are generated is the key to explain phenotypic diversity between individuals [330, 331]. The generation and comparative analysis of underlying gene regulatory networks (GRNs) greatly helps in deciphering these relationships. Generally, a GRN is composed of nodes and edges connecting these nodes. Thereby, nodes represent target genes and their regulators such as transcription factors or chromatin remodeling complexes, while edges specify their directed regulatory relationship such as repression or activation [330]. A TF can for instance bind to the promoter region or TFBSs and thereby regulate a target gene. Note that the expression level of one gene can depend on a combination of several regulators [330]. These connections can be formulated using mathematical regulatory functions such as differential equations [330]. Figure 5.5 illustrates a simple example for a gene regulatory network.

Furthermore, one distinguishes between *cis*–regulatory elements and *trans*–regulatory elements [332]. Thereby, *cis*–regulatory elements refer to non–coding sequence elements that regulate genes in direct vicinity, most often in downstream direction. For instance, TFBSs, promoter, and enhancer elements [332]. *Trans*–regulatory elements are regulators such as proteins and non–coding RNAs that regulate specific genes and are located more distantly from their target gene(s). Of course, both types of regulatory elements are crucial for the regulation of gene expression. A comparative analysis of gene regulatory networks is comprehensively reviewed elsewhere [330].

Cytoscape is a comprehensive and widely used software for data integration, analysis, and visualization of biological networks [333]. Our tool *MutaNET* provides a Graph Modeling Language (GML) formatted network file that enables further and specific investigations of the given network in various ways using, for instance, Cytoscape software.

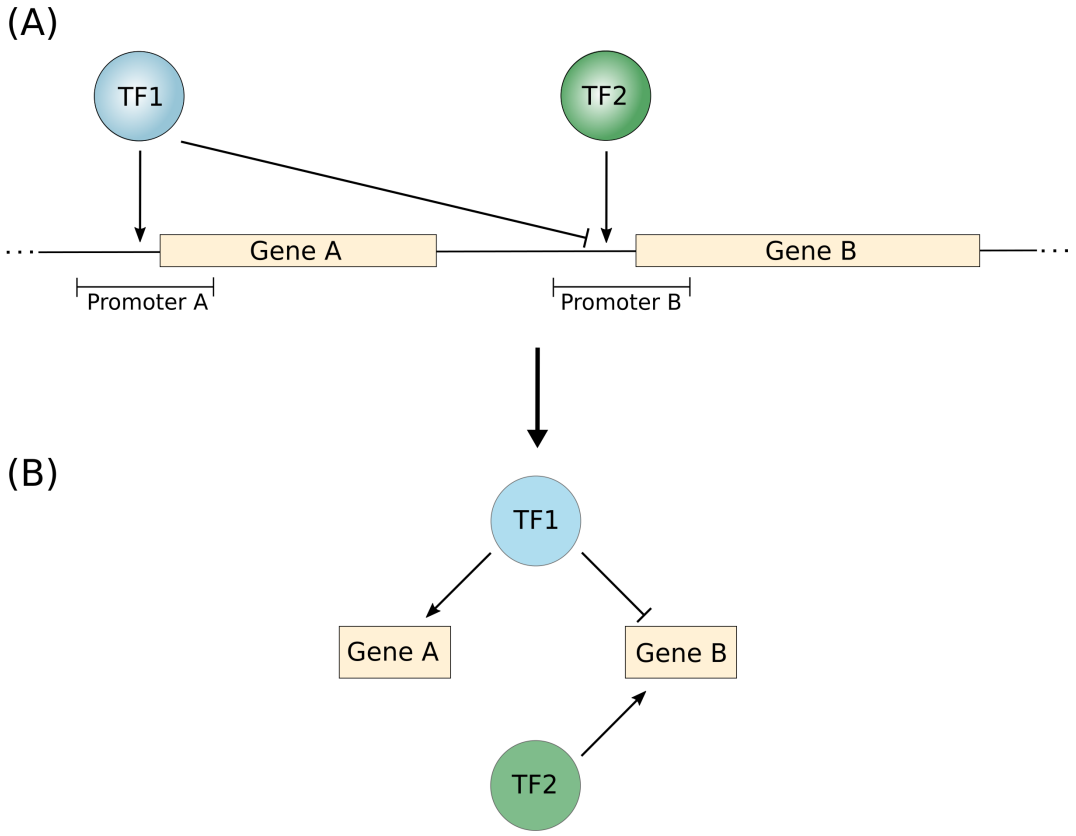

Figure 5.5: Simple GRN example. GRNs represent the relationships between target genes and their regulators. (A) In this simple example, two transcription factors (TF1 and TF2) regulate two target genes (Gene A and Gene B) by binding to their promoter regions. Transcription factor TF1 activates gene A and represses gene B, while transcription factor TF2 activates gene B. (B) The resulting GRN is composed of four nodes (gene A, gene B, TF1, and TF2) and three edges that represent the relations of regulators and target genes.

5.1.4 The bacterial kingdom and antibiotic resistance

As case studies, *MutaNET* was applied to strains of *Escherichia coli* and *Staphylococcus aureus* to decipher the genotypic differences that could confer antibiotic resistance. These analyses also involved an underlying GRN such that the global impact of candidate mutations can be estimated. In the following, these bacteria are shortly introduced. Next, a presentation of bacterial genome regulation and a summary of antibiotic resistance mechanisms that can render antibiotics ineffective follows.

Escherichia coli: a model organism

Escherichia coli (*E. coli*) is a gram–negative bacterium that belongs to the mammalian gut microbiome and is responsible for the production of vitamin K and vitamin B12 [334, 335]. Moreover, it was reported that *E. coli* is important for defense against pathogens that are situated in the gut as well [336]. The *E. coli* bacterium was discovered in 1884 by Theodor Escherich during his studies of the gut microflora in infants [337, 338]. *E. coli* has proven to be a valuable model organism for various studies that investigate gene expression, the proteome, and associated cellular processes [334, 339]. The number of studies involving *E. coli* is tremendous and it is not without reason that Blount [334] referred to *E. coli* as "best understood organism on the planet". In fact, what we know about prokaryotic genome structure is to a great extent based

on *E. coli* studies [340]. Moreover, the GRN of *E. coli* was intensively studied, resulting in the RegulonDB database as a reliable data source for genome–wide transcriptional regulatory network annotations of the *E. coli* K–12 strain [62].

Besides its positive properties for the gut microenvironment, *E. coli* was also found to be pathogenic and to trigger diseases such as bacteremia as well as infections of the urinary tract or bloodstream [334, 341, 342]. Moreover, resistance mechanisms were also observed for various *E. coli* strains, whereby the resistance against individual antibiotics was reported to differ between populations [342]. Note that beside pathogenic bacteria, also commensal bacteria are able to confer drug resistance [342]. We applied our tool *MutaNET*, which automatically scores mutations according to their influence on cellular function, to paired–end sequenced *E. coli* strains to identify candidate resistance mutations.

Staphylococcus aureus: a pathogenic bacterium

The gram–positive bacterium *Staphylococcus aureus* (*S. aureus*) can cause severe diseases of the skin and soft–tissue, bacteremia, endocarditis, sepsis and toxic shock syndrome [343, 344]. The mentioned infections and *S. aureus* pathogenesis are often triggered by intrinsic virulence factors such as enzymes or exotoxins [345, 346, 347]. These factors enable tissue adhesion, immune system evasion, or can harm the host cells otherwise. For instance, the toxic shock syndrome and pneumonia are based on virulence factor release [345, 348, 349, 350]. Virulence factor expression is regulated by cell–to–cell communication referred to as quorum sensing (QS) [351, 352]. The QS system of *S. aureus* enables an adaption to external influences, the control of beneficial group behavior, and provides novel targets for antivirulence therapies. This is presented in Chapter 6. Moreover, the AureoWiki database provides several comprehensive tables that contain detailed information on genes and proteins (position, strand, sequence), operons, and regulation such as target genes and regulators for several *S. aureus* strains [63].

Furthermore, *S. aureus* can develop antibiotic resistance against antimicrobials, which is referred to as methicillin–resistant *Staphylococcus aureus* (MRSA) [343, 345, 353]. The development of MRSA strains is a major burden in the healthcare sector that complicates antibacterial treatment. Our tool *MutaNET* was applied to the sequenced genomes of several *S. aureus* strains to detect candidate mutations in known antibiotic resistance genes or in their regulators that could confer antibiotic resistance.

Prokaryotic genome regulation

Prokaryotic and eukaryotic genome organization and regulation differ. Eukaryotic gene expression is regulated at different levels that include epigenetic modifications, transcription, post–transcriptional processes such as splicing, translation, or post–translational mechanisms. In contrast, prokaryotic regulation is solely controlled at the level of transcription [354]. Note that the eukaryotic genome organization is presented in Section 4.1.1, while the regulation of eukaryotic gene expression is explained in Section 1.2. Since we subjected several *Escherichia coli* and *Staphylococcus aureus* strains to our *MutaNET* software, genome regulation of prokaryotes is summarized in the following.

In contrast to eukaryotic gene expression, prokaryotic transcription and translation take place concurrently in the cytoplasm due to the nonexistent nucleus and the thus free DNA state [340]. Prokaryotic DNA is tightly packaged via supercoiling and usually located as one condensed single circular chromosome in the nucleoid [340]. Thereby, DNA topoisomerase and DNA gyrase are involved in these processes and are thus preferred targets in antibacterial treatment [340, 355]. Besides chromosomes, bacteria normally consist of linear or circular extrachromosomal DNA referred to as plasmids [340]. These plasmids are advantageous for defense against antibiotics as they can encode antibiotic resistance genes that can be transferred via conjugation as well [340, 356, 357].

The organization of prokaryotic genomes must meet the criteria to be adaptable to various environmental influences such as an unavailability of essential nutrients or the presence of sub-

stances harmful for the bacterial cell such as antibiotics [354]. The bacterial strategy to combat those influences are changes in gene expression such that necessary proteins are available for cellular function [354]. Prokaryotic genes can be organized as operons, which are composed of several closely located co–expressed genes [354, 358]. A single promoter upstream of the coding sequences ensures that all genes within an operon are transcribed concurrently leading to a single mRNA that is then translated into several proteins. Beside a gene promoter sequence, operons consist of further regulatory sites such as operator and terminator sequences [354]. Operator sites, which are situated between promoter and the coregulated genes, regulate transcriptional activation or repression. The terminator site determines the end of the operon and functions as the transcription stop. This genome organization and regulation enables a fast adaptation to environmental changes using very efficient regulatory mechanisms [354].

The *E. coli* lac operon is one of the best investigated examples and is therefore described in more detail. The composition and regulation of the lac operon was uncovered by Francois Jacob and Jacques Monod [359]. It is responsible for the metabolism of lactose and is composed of three structural genes named *lacZ*, *lacY*, and *lacA*, which together digest lactose as a nutrient and hence energy source. Expression of these genes in the lac operon is dependent on lactose and glucose availability. In detail, transcription is repressed in case lactose is absent and the transcription machinery is started once lactose is accessible and glucose is unavailable [360]. These dependencies are regulated by the lac repressor that leaves the operator site upon lactose binding. Moreover, the catabolite gene activator protein (CAP), together with cAMP, bind to the promoter sequence to activate transcription in case glucose is not present [360, 361]. Hence, transcription of the lac operon encoded genes is only initiated if CAP and cAMP are bound and the lac repressor does not occupy the operator. Figure 5.6 illustrates the organization of the *E. coli* lac operon.

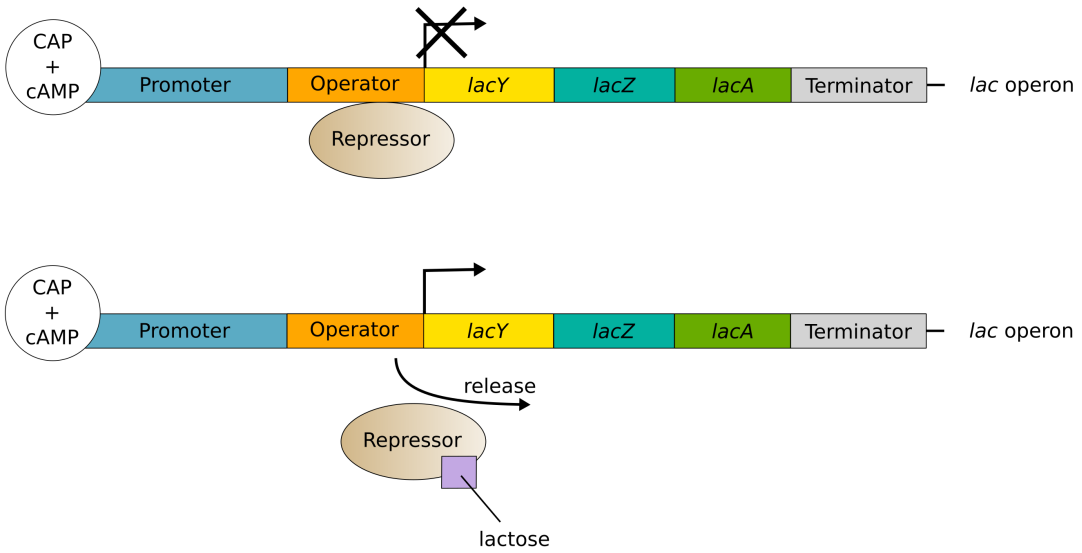

Figure 5.6: The *E. coli* lac operon. The *E. coli* lac operon is composed of three structural genes as well as a promoter, operator and terminator sequence. Gene expression is dependent on the presence of glucose and lactose. The transcriptional repressor is released upon lactose binding and transcription is initiated for lactose digestion.

Due to this organization, mutations located in promoter sequences could affect all genes expressed in the downstream operon in case binding of the RNA polymerase is hindered, whereas non–synonymous mutations in the coding sequence only influence the encoded protein of the underlying gene [354]. Nevertheless, mutations that lead to protein malfunction also influence genes in the complete regulatory cascade or pathway of the affected protein [354]. The analysis of these complex regulatory cascades and an estimation of the global impact on cellular function in case mutations are present is facilitated using *MutaNET*.

Antibiotic resistance complicates medical treatment

Antibiotics are administered to treat bacterial infections [362, 363, 364]. The mechanisms of action are mainly based on an inhibition of bacterial nucleic acid synthesis, suppression of protein synthesis, a modification of the bacterial cell wall or alteration of any other vital (metabolic) pathway. Examples for antimicrobials are β–lactams that inhibit synthesis of the bacterial cell wall leading to cell lysis [365], tetracyclines that interfere with protein biosynthesis [366], or quinolones that inhibit DNA synthesis [355]. Quinolones preferentially target DNA topoisomerase IV and DNA gyrase that are both crucial for DNA unwinding [355]. Streptomycin was found to interact with the small ribosomal subunit and thus affects bacterial protein synthesis [367, 368].

The emergence of antibiotic resistant bacteria such as MRSA strains is a major clinical problem, which considerably impedes appropriate treatment [343, 353, 369, 370]. Normally, the minimum inhibitory concentration is used to measure resistance upon antibiotic administration [371]. There are several mechanisms bacteria can rely on to render antibiotics inefficient and establish an antibiotic resistance against the administered medication. Antibiotics can be modified or exported out of the cell via efflux pumps [366, 372]. Tetracyclines, for instance, are removed from the cell via efflux pumps [366]. NorA, NorB, and NorC efflux pumps of *S. aureus* were reported to transport fluoroquinolones out of the bacterial cell [372]. In general, efflux pumps can be specific for an individual drug or can be multidrug resistant and thus allow a broad removal of various different compounds from a cell [372].

Considering another resistance mechanism, the drug target can be altered such that an interaction with the antibiotic is hindered. For instance, mutations in genes coding for rRNA, most often the genes that code for the ribosomal protein S12, were found to harbor mutations that confer antibiotic resistance to streptomycin [367]. These alterations generally prevent the drug from binding to the small ribosomal subunit such that bacterial protein synthesis is not adversely affected [367]. Antibiotic resistance to the competitive inhibitor chloramphenicol was also attributed to mutations in the target rRNA that is involved in peptidyl transfer [373, 374]. Moreover, resistance to quinolones is based on specific mutations in the DNA topoisomerase IV and DNA gyrase genes that hinder interaction between drug and target [355]. Beside alteration of the drug target, the drug itself can also be modified. Resistance to chloramphenicol was found to be due to on acetylation of the drug by acetyltransferases or phosphotransferases that render it inactive [375]. Moreover, a degradation of the drug is possible, as it is the case for β–lactamase enzymes that break down β–lactam antibiotics such as penicillin [365, 376]. Also effective is a previous hindrance of a drug to enter the bacterial cell. For the antimicrobial substance chloramphenicol, it was observed that the transport of the drug into the bacterial cell and thus to its target is prevented in advance [377, 378]. The entire set of antibiotic resistance genes is referred to as antibiotic resistome [379]. This gene collection also comprises precursor genes with decreased antimicrobial activity but with preferences to interact with antibiotics and that thus have the potential to confer resistance [379, 380].

Bacteria can either acquire resistance via the transfer of genetic material or they can be intrinsically resistant. Examples for intrinsic resistance are gram–negative bacteria that have an advantage based on the impermeability of their outer membrane as an additional drug impediment, a general absence of a drug target, or an overexpression of efflux transporters [381, 382]. Considering acquired resistance, there are two main mechanisms: vertical and horizontal gene transfer [383]. Vertical gene transfer designates the transfer of antibiotic resistance genes to the daughter cells during cell division [383]. In contrast, horizontal gene transfer refers to exchange of genetic material between individual bacteria and is independent of cell division [356, 383]. Horizontal gene transfer was also observed between phylogenetically distant bacteria [383]. This is further separated into conjugation, transduction, and transformation that refer to a transfer via plasmids through direct cell–to–cell contact, an intake of genetic material via bacteriophages, or from the environment even between distantly related pathogens, respectively [356].

As mentioned above and also in Section 4.1.2, genomic mutations can shape the observed phenotype, cause severe diseases, and considering pathogens, mutations were found to greatly contribute to increased bacterial survival and to confer resistance against major antibiotics. The

detection of candidate resistance mutations in bacterial genomes with their constant variability, which tremendously impedes antimicrobial treatment, is a major challenge. Therefore, we applied our software *MutaNET* to *S. aureus* and *E.coli* strains and aimed at identifying novel resistance mutations. The Pathosystems Resource Integration Center (PATRIC) is a comprehensive database that contains information on various bacterial strains concerning bacterial genomes, transcriptomes, proteomes, and resistomes [64, 65]. We used the PATRIC database to retrieve known antibiotic resistance genes.

5.2 Aim of this project

Mutations can affect an organismal phenotype in many ways, whereby the genomic position of a variant is of fundamental importance. Coding mutations can influence protein function [384], whereas those in regulatory sites can affect expression of the gene itself and of genes in that regulatory cascade [385]. Thereby, gene expression levels are regulated by TFs via binding to TFBSs [386]. We developed *MutaNET* that scores the potential impact of mutations on gene expression and protein function of a given genome. *MutaNET* statistically compares the mutational impact on coding regions and TFBSs using refined scoring schemes. If regulatory information is provided as well, a GRN is constructed to examine the global effect of individual mutations. To the best of our knowledge, a similar tool that implements a combinatory analysis of variant calling, statistical analysis, and incorporation of a GRN does not exist yet. Moreover, *MutaNET* supports statistical comparisons between different gene groups such as bacterial antibiotic resistance and non–antibiotic resistance genes. Since mutations in antibiotic resistance genes or in their regulatory sites can cause or affect antibiotic resistance of bacterial strains, we used *MutaNET* for a detailed analysis of mutations in antibiotic resistance genes and their possible impact on antibiotic resistance. In general, bacteria are very suitable to investigate the impact of mutations on the phenotype due to their haploid genome that comprises only one copy of every gene [340].

5.3 *MutaNET* facilitates mutation analysis

MutaNET consists of several analysis steps: a mutation calling pipeline, a statistical comparison of mutations in different genomic regions, and generation of the underlying GRN, see Figure 5.7. Mutations can either be called automatically from NGS paired–end reads using the embedded mutation calling pipeline presented in [61], or mutations can be provided by the user. Mutations are then assigned to different genomic regions (coding region, promoter, and TFBS) using in–house scripts analogous to BEDTools [78]. Statistically significant differences are identified based on the Wilcoxon rank–sum test.

Fo a detailed mutation analysis, *MutaNET* differentiates between synonymous, missense, nonsense, readthrough, and frameshift mutations. The effect of mutations in coding regions is assessed using an amino acid substitution matrix and a pairwise sequence alignment between reference and mutated protein sequence, see Section 5.1.2. Since the impact of a mutation is influenced by its position in the protein, protein domain information, which can be downloaded from UniProt [227], is incorporated in the analysis as well. Mutations in TFBS can increase or decrease the ability of the corresponding regulator to bind [300, 385]. A score is computed that indicates whether mutations in a TFBS are likely to increase or decrease the binding ability of the TF. This TFBS mutation score is based on a PWM constructed from TF motif sequence alignments and a comparison between observed and random mutations following the method by Melton et al. [300], see Section 5.1.2. Finally, a GRN is constructed to decipher the global effect of mutations. The nodes (genes) display the number of non–synonymous coding, promoter, and TFBS mutations. This allows to quickly identify genes with mutations that directly or indirectly regulate specific genes, such as antibiotic resistance genes. GRNs can be further processed using programs such as Cytoscape [333].

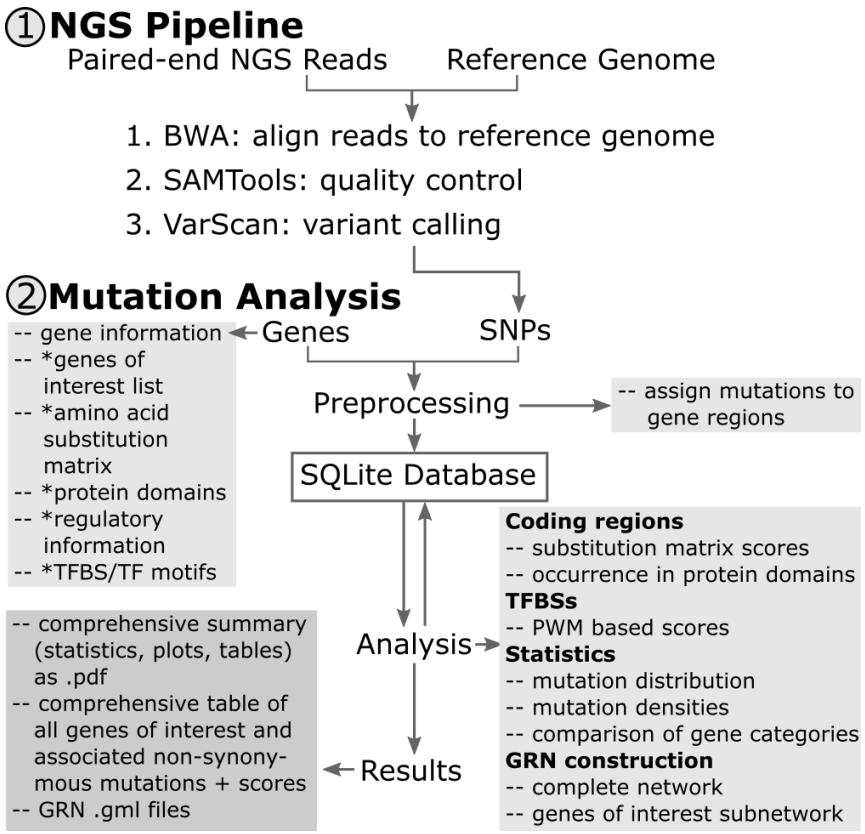

Figure 5.7: *MutaNET* workflow: NGS pipeline and mutation analysis. Shown is the *MutaNET* workflow. A variant calling pipeline can be executed to call variations from paired–end next–generation sequencing reads prior to a detailed mutation analysis. Mutations are subsequently compared using statistical tests and refined scoring schemes. An underlying GRN helps to find mutations with a global impact on gene regulation. A Python SQLite database serves as central storage system [83, 84]. An asterisk (*) denotes optional information. This figure was generated by Markus Hollander during the preparation of our manuscript for submission to the *Bioinformatics* journal.

5.4 Case study: decipher antibiotic resistance

To demonstrate one possible application, we applied *MutaNET* to *E. coli* K–12 and *S. aureus* NCTC 8325 reference strains. Paired–end reads were based on sequence type 131 (ST131) and clonal complex five (CC5) for *E. coli* and *S. aureus*, respectively [61, 387]. Mutations were called with the embedded NGS pipeline from a set of 300 *E. coli* and 30 *S. aureus* strains. The 30 *S. aureus* strains were subject to the work of Hamed et al. [61], while the genome sequences of 300 *E. coli* strains were downloaded from the NCBI BioProject database [60]. The BioProject accession numbers of the utilized *E. coli* strains can be found in Table 5.2. Regulatory and antibiotic resistance information for *E. coli* and *S. aureus* was taken from RegulonDB [62], AureoWiki [63], and PATRIC [64, 65]. Markus Hollander assembled these datasets.

We reported 93,204 and 18,447 mutations of which 3,035 and 372 were found in antibiotic resistance genes for *E. coli* and *S. aureus*, respectively. All numerical results are depicted in Table 5.3. The observed number of transitions, transversions, and indels reflect a generally higher amount of transitions (C $\leftrightarrow$ T and A $\leftrightarrow$ G), which is based on their chemical structures. For *S. aureus*, we found that the number of missense mutations was significantly lower ($p = 0.02$) in antibiotic resistance genes (21.3%) compared to non–antibiotic resistance genes (28.4%). Upon antimicrobial treatment, antibiotic resistance genes, like multidrug efflux pumps, are essential for bacterial survival and so that specific missense mutations in important protein domains

could result in a fitness loss [388]. A decreased number of missense mutations in antibiotic resistance genes was observed for *E. coli* as well, although this difference was statistically not significant.

Table 5.2: *E. coli* accession numbers of the NCBI BioProject database. In our case study, we analyzed 300 genome sequences of *E. coli* sequence type 131 (ST131). The sequences were retrieved from the NCBI BioProject database [60] and subsequently subjected to our variant calling pipeline with a *E. coli* K–12 reference genome. The respective BioProject accession numbers together with a short description are presented. Data retrieval was conducted by Markus Hollander.

BioProject accession	Description taken from the BioProject database [60]
PRJNA383781	"Escherichia coli ST131–O25b:H4 strain: 81009."
PRJDB3868	"Whole genome sequencing of Escherichia coli ST131."
PRJEB6262	"Four main virotypes Escherichia coli ST131."
PRJEB21171	"Escherichia coli ST131 in Germany."
PRJNA211153	"Escherichia coli O25b:H4–ST131 Genome sequencing."
PRJDB4303	"Comparative genomics of ESBL–producing Escherichia coli ST131 isolates."
PRJEB5004	"Escherichia coli ST131 indian strains."
PRJEB15503	"Detection of the high–risk clone ST131 of Escherichia coli carrying the colistin resistance gene mcr–1 and producing acute peritonitis."

5.4.1 Antibiotic resistance regulatory subnetworks

To analyze the global effect of mutations, antibiotic resistance regulatory subnetworks of *E. coli* and *S. aureus* were constructed using *MutaNET*. It was reported that, for instance, overexpression of efflux pumps is associated with antibiotic resistance as the drug can be expelled from the bacterial cell [389, 390, 391].

We found several severe mutations in the *E. coli* helix–turn–helix (HTH) domain of transcriptional regulator AcrR [392] that could lead to malfunction, see Figure 5.8A. In consequence, the repression of *acrA* and *acrB*, which code for multidrug–resistant efflux (MDRE) pump subunits, might be disturbed. This could lead to the development of antibiotic resistance due to overexpression of these MDRE pumps. The *acrAB* operon is negatively regulated by repressor MprA [62] for which a frameshift mutation in the HTH domain and a missense mutation were observed. Moreover, dysfunction of MprA could lead to overexpression of multidrug transporters EmrA, EmrB, and EmrE that were found to confer resistance to several antibiotics via multidrug efflux [393, 394].

Concerning *S. aureus*, Figure 5.8B shows that the NorG protein is one of the central regulators in the regulation of the multidrug efflux pumps encoded by *S. aureus* [395]. We observed a nonsense mutation (K5Stop) in the DNA binding HTH domain of NorG that results in a non–functional protein with a length of four amino acids. A loss of NorG would have severe consequences as activation of the regulator encoding genes *arlS*, *lexA*, *mgrA*, *sarR*, and *sarZ* as well as of efflux pump encoding gene *norB* is hindered [395]. Moreover, an upregulation of efflux pumps *abcA* and *norC* could be promoted due to the missing transcriptional repression by

Table 5.3: *MutaNET* results of mutations in *E. coli* and *S. aureus* strains. Comparison of *E. coli* and *S. aureus* strains was conducted by applying *MutaNET* to the datasets described in the text and using regulatory information from RegulonDB [62] and AureoWiki [63], and information on resistance genes from PATRIC [64, 65] and the literature (see supplementary material of our publication). Antibiotic resistance genes include MDRE pumps and their direct regulators. Numbers associated with antibiotic resistance genes are given in brackets. Density is defined as the number of mutations per kb. We assumed statistical significance if $p < 0.05$, which is denoted by an asterisk (*). We used the Wilcoxon rank–sum test to assess the distributions of antibiotic resistance genes against non–antibiotic resistance genes.

	E. coli	***S. aureus***
Strains in dataset	300	30
AR genes	97	47
MDRE pumps	39	13
Direct MDRE regulators	29	8
Non–AR genes	4,468	2,929
TFs	157 (35)	38 (10)
TFBSs	1,794 (113)	261 (4)
Mutations	93,204 (3,035)	18,447 (372)
Transitions [%]	73.7 (74.6)	66.9 (66.7)
Transversions [%]	25.8 (24.9)	31.8 (32.2)
Indels [%]	0.4 (0.5)	1.3 (1.1)
Synonymous [%]	80.4 (82.1)	64.6 (69.4)
Missense [%]	14.9 (13.6)	28.4 (21.3)*
Mean density synonymous	16.9 (18.9)*	4.4 (5.1)*
Mean density missense	3.8 (3.7)	2.5 (1.6)

NorG. Moreover, since *mgrA* activation is lost, an overexpression of efflux pump *norA* is possible due to nonexistent transcriptional repression of *norA* by the MgrA regulator, see Figure 5.8B. It was already shown that overexpression of *norA* is associated with antibiotic resistance [390].

Besides *norA* repression, the MgrA regulator also represses *tet38* and *norC* efflux transporters [396, 397], which can thus again lead to efflux pump upregulation. Thus, the functional loss of only one regulator protein can favor an upregulation of several multidrug efflux transporters, which could in turn lead to antibiotic resistance. This emphasizes the importance of specific mutations in key regulatory proteins and their potential global impact on antibiotic resistance. Despite the nonsense mutation in the NorG protein sequence, we also found two missense mutations, S28P ($Score_{cod} = 0.63$) and H29Q ($Score_{cod} = 0.64$), in the HTH DNA binding domain, which could also have the potential to disturb *NorG* function.

Moreover, we found a nonsense mutation in the sigma 70 domain of the SigB regulator protein, compare with Figure 5.8B. Assuming the function of SigB is restricted due to this mutation ($Score_{cod} = 0.92$), this can have an impact on the regulation of efflux pump genes *norA*, *msrA*, and *msrB* via a pathway composed of regulator proteins ArlS, SarA, and Rot. In summary, the generated regulatory subnetwork together with annotated mutations is a good basis for further speculations and interpretations, and hence greatly helps to decipher possible resistance mechanisms of a given strain.

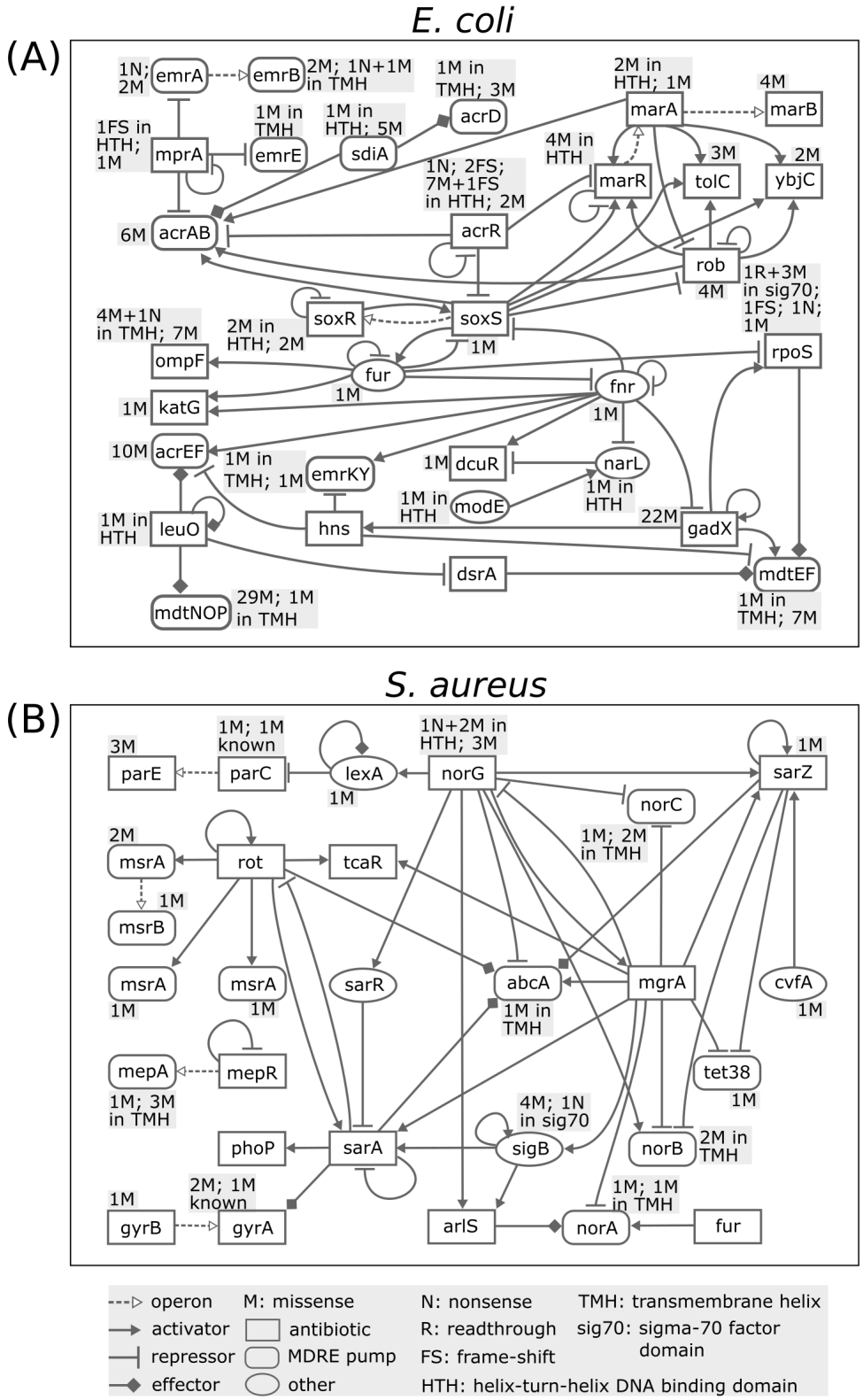

Figure 5.8: Antibiotic resistance GRNs of *E. coli* and *S. aureus*. Shown are the (truncated) antibiotic resistance GRNs of (A) *E. coli* K–12 and (B) *S. aureus* NCTC 8325 strains provided by *MutaNET*. Antibiotic resistance genes and their direct regulators are highlighted. Since antibiotic resistance is associated with an overexpression of multidrug efflux pumps [389, 390, 391, 398], the antibiotic resistance genes are composed of MDRE pumps and their direct regulators as well. The GRNs were generated by Markus Hollander during the preparation of our manuscript for submission to the *Bioinformatics* journal.

5.4.2 Mutation analysis across species

To decipher similar resistance mechanisms across several species, a comparison of candidate mutations in related genes and/or their regulators is reasonable. Applying *MutaNET* enables this mutation analysis across various species. As we compared the mutational landscape of different *S. aureus* and *E. coli* strains, we found several mutations in the genes *parC* and *gyrA* for both, *E. coli* and *S. aureus*, see Table 5.4. The genes *parC* and *gyrA* were found to be associated with antibiotic resistance to quinolones [227, 399].

Table 5.4: Mutations in *E. coli* and *S. aureus parC* and *gyrA* genes. We compared mutations found by *MutaNET* with known resistance mutations from the UniProt database [227]. Thereby, "yes" indicates that the mutation was reported, "no" otherwise. All mutations provided by the UniProt database were associated with resistance to quinolones.

Gene	Protein	Organism	Mutation	UniProt	*MutaNET*
parC	DNA topoisomerase	*E. coli*	S80L	yes	no
			S80I	no	yes
			S80R	no	yes
			E84K	yes	no
			E84P	yes	no
			E84V	no	yes
		S. aureus	S80F	yes	yes
			S80Y	yes	no
			E84K	yes	no
			E84L	yes	no
gyrA	DNA gyrase	*E. coli*	S83A	yes	no
			S83L	yes	yes
			S83W	yes	no
			D87N	yes	yes
			D87V	yes	no
			D87G	no	yes
		S. aureus	S84L	yes	yes
			S84A	yes	no
			E88K	yes	no

Some of the reported mutations by *MutaNET* are known resistance mutations, whereas the other mutations reported by *MutaNET* can be considered as candidate resistance mutations. A multiple sequence alignment of *parC* and *gyrA* genes, shown in Figure 5.9, highlights the mutated positions. This finding suggests a similar resistance mechanism on these *E. coli* and *S. aureus* strains involving *parC* and *gyrA* genes.

E. coli K-12: parC (1), gyrA (3); *S. aureus* NCTC 8325: parC (2), gyrA (4)

```
       ::***: ****:*:*******: *:::* ::* :. *: *:: :** *: *. .  *
(1) 57 SAKFKKSARTVGDVLGKYHPHGDSACYEAMVLMAQPFSYRYPLVDGQGNWGAPDDPKSFA 116
(2) 57 DKNFRKSAKTVGDVIGQYHPHGDSSVYEAMVRLSQDWKLRHVLIEMHGNNGSIDN-DPPA 115
(3) 60 NKAYKKSARVVGDVIGKYHPHGDSAVYDTIVRMAQPFSLRYMLVDGQGNFGSIDG-DSAA 118
(4) 61 DKSYKKSARIVGDVMGKYHPHGDSSIYEAMVRMAQDFSYRYPLVDGQGNFGSMDG-DGAA 119
```

Figure 5.9: Multiple sequence alignment of *E. coli* and *S. aureus* genes *parC* and *gyrA*. The alignment highlights the positions of reported resistance and candidate resistance mutations, see also Table 5.4. This figure was generated by Markus Hollander during the preparation of our manuscript for submission to the *Bioinformatics* journal.

5.5 Summary

We developed *MutaNET* to automate mutation analysis by providing a tool that is able call mutations from sequenced paired–end reads, followed by a detailed analysis of these variants in several genomic regions. For this analysis, we integrated scoring schemes to estimate the impact of mutations in coding regions and in TFBSs on gene function and regulation. Considering coding mutations, *MutaNET* differentiates between synonymous, missense, nonsense, readthrough, and frameshift mutations to optimally assess the impact of individual sequence variations. The influence of mutations in TFBSs is estimated by a comparison between observed and randomly introduced mutations, also using position weight matrices. Besides these scoring schemes, an additional integration of a gene regulatory network greatly aids in the analysis of mutations concerning their global impact on cell function. As a case study, we applied *MutaNET* to paired–end sequenced genomes of *E. coli* and *S. aureus* strains to analyze antibiotic resistance. We found severe mutations in key regulator proteins that could influence the resistance phenotype to a large extend by an overexpression of multidrug efflux pumps. Moreover, we found similar candidate resistance mutations in orthologous *E. coli* and *S. aureus* genes, that suggest similar resistance mechanisms across species.

CHAPTER 6

Targeting bacterial quorum sensing: a novel antivirulence strategy

Quorum sensing (QS) plays a crucial role in bacterial survival and is hence predestined as target in the development of novel antivirulence therapies. Different QS systems and how to target them were summarized in our review paper that was published in "Interfering with Bacterial Quorum Sensing. Kerstin Reuter, Anke Steinbach, and Volkhard Helms. *Perspectives in Medicinal Chemistry*, 8:1–15, 2016". This work was in cooperation with Anke Steinbach from the Helmholtz Institute for Pharmaceutical Research Saarland. The following chapter is a shortened and adapted version of our review article on bacterial QS. Interspecies and interkingdom communication as well as quorum sensing in *Pseudomonas aeruginosa* (*P. aeruginosa*) are also covered in our publication, but omitted here. *P. aeruginosa* is a gram–negative bacterium that causes chronic lung infections based on QS controlled biofilm formation and that is predominantly found in patients suffering from cystic fibrosis [400, 401, 402]. Please refer to the publication for details on the *P. aeruginosa* quorum sensing system and possible targeting strategies.

6.1 Quorum sensing: cell–to–cell communication

Quorum sensing (QS) is a signaling mechanism that is quite common in bacteria and involves the exchange of small chemicals between bacteria. It was first identified in the marine bacterium *Vibrio fischeri* [403, 404, 405]. QS describes the ability of an organism to adapt the activity of its gene expression machinery to the population density in the nearby environment. This allows bacteria to act as a community and thus express phenotypes that are beneficial for the group. Single bacteria release internally synthesized chemicals, known as autoinducers (AI), either by actively transporting them across the bacterial cell membrane or by passive diffusion through the membrane. In this manner, the external AI concentration automatically reflects the cell population density. When a certain cell population density, that means AI density, is reached, gene expression of bacterial cells is altered and transcription of certain genes is switched on or off. Thus, in adapting their behaviors to various environments, bacteria can regulate genes that are advantageous for their survival. Such cell–to–cell communication is important, for example, to organize light–emitting reactions (bioluminescence), to form biofilms, to produce antibiotics, to express virulence factors, or for the transfer of genetic material via conjugation or transformation [96, 406].

Various genera, such as *Aliivibrio*, *Escherichia*, *Pseudomonas* and *Staphylococcus*, utilize QS for cell–to–cell communication enabling them to adapt their gene expression levels in order to express phenotypes that are advantageous for the group. Inhibition of QS mechanisms in the course of antivirulence therapies has been discussed as an attractive way of combatting bacterial infections [96]. It is suggested that due to a reduced selective pressure on the bacterial population, an emergence of antibiotic resistance is diminished [96, 407]. In the following, the well understood QS systems of the model organism *Vibrio fischeri* and of the pathogen *Staphylococcus aureus* are presented together with pharmaceutical strategies to target the gene–regulatory QS machinery for the development of novel antivirulence strategies.

6.1.1 The QS *lux* system of *Vibrio fischeri*

The marine luminous bacterium *Vibrio (Aliivibrio) fischeri* (*V. fischeri*) forms a symbiotic relationship with various eukaryotic hosts. Thereby, *V. fischeri* benefits from nutrient supply while the host takes advantage of the luminescence reaction carried out by this bacterium [408]. Light emission is used in different ways, for example to produce counterillumination that prevents detection by natural enemies (camouflage), to support hunting, to provide protection against predators, or to help in alluring mates [408, 409, 410].

V. fischeri uses the well understood QS system, shown in Figure 6.1, to control and regulate the bioluminescence reaction. The signaling system requires two regulatory proteins, encoded by the genes *luxI* and *luxR*, to carry out central functions in the QS circuit. *luxI* is organized in the *luxICDABE* operon that also harbors the genes needed for the luminescence reaction itself. The two luciferase subunits, needed for the luminescence reaction, are expressed by *luxAB* while the proteins expressed from *luxCDE* are part of the reductase system essential for luciferase aldehyde biosynthesis [406].

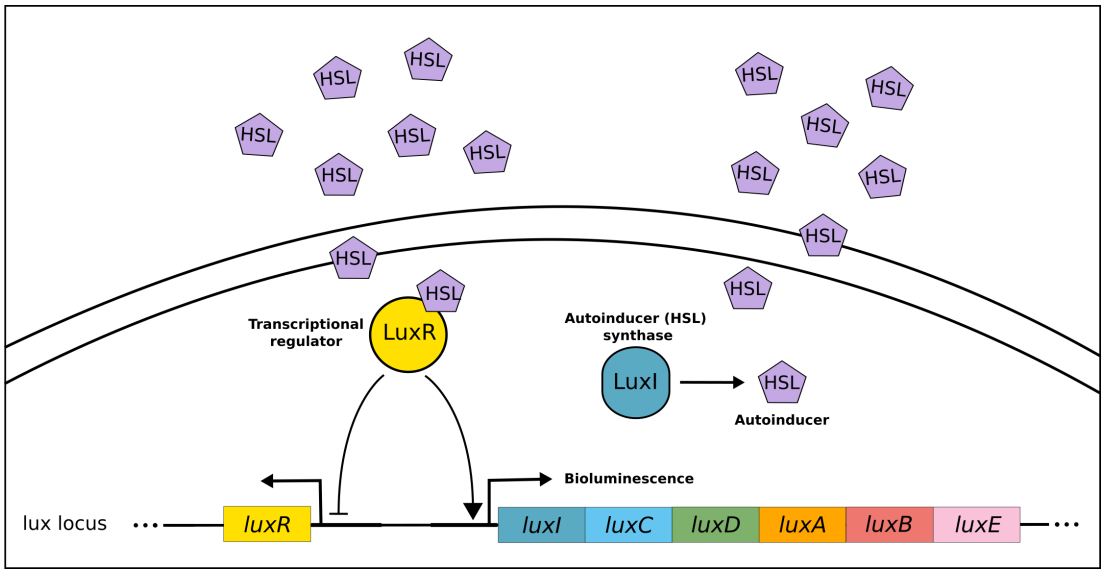

Figure 6.1: QS bioluminescence system of *V. fischeri*. The QS system of the marine bacterium *V. fischeri* requires the regulatory proteins LuxI and LuxR as well as LuxAB and LuxCDE that are needed for the luminescence reaction and luciferase aldehyde biosynthesis, respectively.

LuxI (the protein expressed from *luxI*) synthesizes a signaling molecule, or AI, homoserine lactone (HSL) that can passively diffuse between intra– and extracellular environment [406, 411]. In consequence, the HSL concentration is equally distributed between inside and outside of a bacterial cell. When a concentration threshold is reached, HSL binds to the intracellular transcriptional regulator LuxR [406, 408, 412]. The LuxR–HSL complex then activates the *luxICDABE* operon by binding to the 20 bp long lux box binding sequence, which is located upstream (–40 bp) of the *luxICDABE* operon, but at the same time represses transcription of *luxR* by binding to the *luxR* promoter [408, 413]. Thus, LuxR–HSL indirectly down–regulates the expression of *luxICDABE* via a negative feedback loop as well [408]. Thus, a low cell density entails a low transcription rate of *luxICDABE*, a low level of HSL and, finally, low light production. In contrast, high cell populations lead to synthesis of more AI molecules and light production increases.

6.1.2 The QS *agr* system of *Staphylococcus aureus*

Staphylococcus aureus (*S. aureus*) is a gram–positive bacterium responsible for infections of the skin and soft tissue, bacteremia, endocarditis, sepsis and toxic shock syndrome [343, 344].

Treating *S. aureus* is complicated due to the evolvement of multidrug resistant *S. aureus* strains, known as methicillin–resistant *Staphylococcus aureus* (MRSA) [343, 353]. In general, methicillin–resistant pathogens are a huge burden must be overcome, especially in the healthcare sector.

Various infections that are caused by *S. aureus* are facilitated by several (intrinsic) bacterial virulence factors. Virulence factors comprise a large spectrum of various enzymes and exotoxins that enable a evasion of the immune system, tissue adhesion, or cause damages of the host cell [345, 346, 347]. Sepsis, which is caused by virulence factors, is associated with enterotoxin release such as the toxic shock syndrome toxin [345]. Further virulence factors that are secreted by *S. aureus* are hemolysins, such as α–hemolysin, that trigger the destruction of membrane structures and that can cause pneumonia [348, 349, 350]. Thus, virulence factors are a crucial part of the pathogenesis of bacterial infections. Expression of different virulence factors depends on external influences [351] and is regulated by the cell–density–dependent QS accessory gene regulator (agr) system of *S. aureus* [351, 352], which is displayed in Figure 6.2.

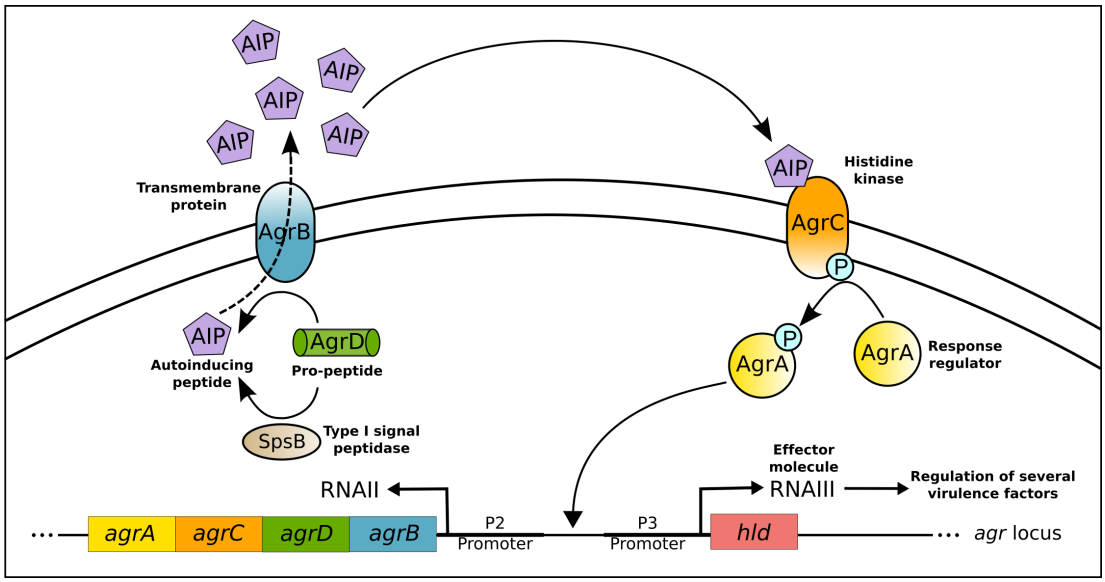

Figure 6.2: QS accessory gene regulator system of *S. aureus*. QS in *S. aureus* is based on genes *agrA*, *agrB*, *agrC*, *agrD*, and *hld* that have different functions in the QS system such as the transmembrane protein AgrB or histidine kinase AgrC.

The *agr* locus consists of the five genes *agrA*, *agrB*, *agrC*, *agrD*, and *hld*, with *agrA* to *agrD* organized in one operon [351]. Thereby, the *agr* operon and hld are controlled by different promoters, termed P2 and P3, respectively. Each of these proteins takes over a different function in the QS system: the transmembrane protein AgrB and the type I signal peptidase SpsB convert the pro–peptide AgrD into an autoinducing peptide (AIP), which is then used as cellular signaling molecule. While AgrB removes the charged AgrD–carboxy–tail [414, 415], SpsB is responsible for the removal of the amphipathic N–terminus [416]. In contrast to gram–negative bacteria, short peptides rather than homoserine lactones are used as signaling molecules in *S. aureus*. *S. aureus* encodes four different allelic AIP variants (AIP–I to AIP–IV), whereby the length of these AIPs varies between seven and nine amino acids [417]. Five residues form a thiolactone ring at the C–terminus [417] and each secreted AIP can bind specifically to the respective AgrC histidine kinase [351, 418].

Upon AIP synthesis, the signaling molecule is transported out of the cell by transmembrane protein AgrB. AIP then binds to the extracellular part of the integral membrane protein AgrC. As mentioned, AgrC functions as a histidine kinase that in turn autophosphorylates the response regulator AgrA. This autophosphorylation is established by an AIP induced change of the AgrC conformation that enables a connection between the sensor and kinase domains [351]. AgrA subsequently upregulates expression of the *hld* and *agr* operons by binding to the intergenic DNA between promoters P2 and P3. The *hld* gene encodes a RNAIII effector molecule

that posttranscriptionally regulates several virulence factors (for example α–hemolysin). In consequence, the *agr* system regulates expression of virulence factors but, in addition to other global regulators, also regulates its own expression. In total, RNAIII and AgrA regulate the transcription of about 200 genes also comprising virulence factors [419].

6.2 Novel approaches: interfering with QS

A number of studies have succeeded in exploiting the bacterial QS system as target for antibacterial treatments. Several studies are presented in the following. It is believed that targeting the QS system is advantageous over conventional therapeutic strategies due to an approach that only disrupts the communication between bacteria rather than an elimination of individual bacterial cells. This strategy is therefore assumed to lower the selective pressure and reduce the rate at which antibiotic resistance normally develops during treatment [96, 407]. Since bacteria use their QS system to also regulate expression of virulence factors and biofilm formation, inhibiting the signaling system should favor the viability of less virulent strains and prevent or minimize the establishment of pathogenic biofilms [96].

Various classes of chemical compounds and targets that interfere with different parts of the QS cascade have been proposed (see below). All QS systems share a general pattern or signaling cascade: an AI is synthesized, a certain AI concentration reaches a threshold, and the AI binds to a transcriptional regulator that subsequently activates or represses certain genes. This opens up four promising strategies for an anti–infective therapy: the AI synthesis can be suppressed, the AI can be attacked and decomposed in an enzymatic reaction or deactivated using antibodies, regulator antagonism, and binding of a regulator protein to the DNA can be hindered [96].

6.2.1 Targeting the QS system of *V. fischeri*

The QS system of *V. fischeri* was targeted in several studies. Schaefer et al. [420] investigated synthetic HSL analogs in terms of their binding affinity to LuxR and their ability to reduce the luminescence reaction. They identified several LuxR binders which induced a luminescence reaction and also identified competitive HSL compounds, which were not capable of activating the luminescence reaction and could thus be applied to inhibit QS dependent gene expression.

Both Piletska et al. [421] and Cavaleiro et al. [422] studied the ability of polymers to attenuate QS in *V. fischeri*. These synthetic polymers, which can be itaconic acid–based, were able to sequester the autoinducing signal and were therefore termed signal molecule–sequestering polymer [421]. These polymers showed affinities to the HSL signaling molecule and prevented the *V. fischeri* bioluminescence reaction by absorbing the AI [421]. One advantage of these polymers, in comparison to other anti–infectives, is the decrease of harmful side–effects [421].

6.2.2 Attacking the *S. aureus* QS system to treat infections

As mentioned several strategies can be applied to disrupt the sequential bacterial QS cascade. In the following, we present experimental possibilities on how to attack an AI, hinder an interaction between AI and the regulator protein, and how to prevent regulator binding to DNA. Subsequent to these experimental approaches, we introduce some *in silico* studies that investigated QS inhibitors.

Attacking the AI

Park et al. [423] applied an immunopharmacotherapeutic approach and investigated monoclonal antibodies in terms of their ability to neutralize the AI peptide AIP–IV via sequestration. Out of 20 produced anti–AIP–IV, one antibody (AP4–24H11) with high binding affinity was

highly specific towards AIP–IV. Moreover, applying AP4–24H11 to different *S. aureus* strains resulted in a decreased α–hemolysin production. This antibody was also successfully applied to an infected murine model that showed abscess formation [423]. These results highlight that the removal of an autoinducing signal peptide from a bacterial system can result in inhibition of QS dependent gene expression without manipulating bacterial genetic information.

Preventing AI–regulator interactions

Mansson et al. [424] investigated the potential of marine bacteria to decrease the pathogenicity of *S. aureus* by attacking its *agr* QS system. They showed that the investigated marine photobacterium produces two AI antagonists named solonamide A and B, see Figure 6.3. These antagonists were able to inhibit QS in a highly virulent community–acquired MRSA strain.

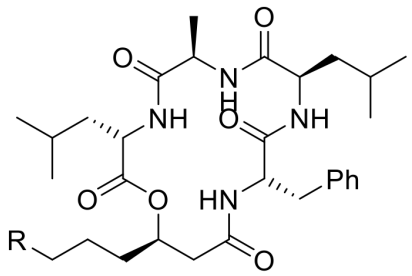

Figure 6.3: *S. aureus* QS inhibitors solonamide A and B. Dependent on the residue R, the inhibitor is categorized into solonamide A with $R = H$ and solonamide B with $R = C_2H_5$ [424]. This figure was generated by Anke Steinbach during the preparation of our manuscript.

Murray et al. [425] synthesized several small–molecule inhibitors that interact with the cytoplasmic membrane and appear to affect AIP–AgrC interaction as allosteric non–competitive inhibitors. The most potent inhibitor was tested in a mouse model that was infected with *S. aureus*. These experiments showed that the inhibitory effect towards the *agr* system could decrease nasal colonization in mouse.

Inhibiting regulator binding to DNA

Since the *S. aureus agr* system was shown to be involved in skin and soft tissue infections [344], Sully et al. [426] aimed at identifying a small molecule inhibitor that disrupts the *S. aureus* signaling cascade but, at the same time, omits suppressing that of commensal *Staphylococcus epidermidis* (*S. epidermidis*). The reason behind this was that *S. epidermidis* is an important gram–positive bacterium involved in host defense mechanisms against skin pathogens and is thus important for human skin flora [427]. To ensure specificity toward *S. aureus*, they investigated the structural differences between components of the *agr* systems of *S. aureus* and *S. epidermidis*. Since the AgrC residues, which are crucial for *agr* functionality, were found to be conserved between *S. aureus* and *S. epidermidis*, AgrA was selected as target protein. The authors applied high–throughput screening to 24,087 compounds and discovered inhibitors of the *agr* signaling cascade that suppressed up–regulation of virulence factors. The inhibitor was named savirin short for *S. aureus* virulence inhibitor [426]. Its structure is shown in Figure 6.4.

Savirin blocks binding of AgrA to the promoter region, which was confirmed by changing the P3 coupled product to GFP. To analyze the specificity of savirin binding to *S. aureus* AgrA, the *in silico* tool SwissDock [428] was applied to dock savirin to both AgrA of *S. epidermidis* and *S. aureus*. Since the critical AgrA residues were found to be not conserved between these bacteria, only the latter docking was successful. In consequence, the authors concluded that savirin preferentially binds to AgrA of *S. aureus* rather than *S. epidermidis* making AgrA a reliable target structure and savirin a promising *agr* signaling inhibitor.

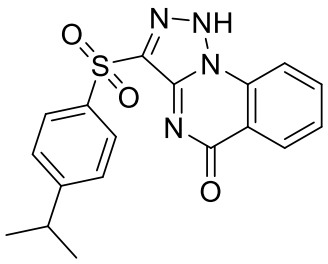

Figure 6.4: *S. aureus* QS inhibitor savirin. Savirin inhibits binding of AgrA to the promoter region. Savirin takes its name from the first syllables of *S. aureus* virulence inhibitor [426]. This figure was generated by Anke Steinbach during the preparation of our manuscript.

Daly et al. [429] recently reported that a polyhydroxyanthraquinone, which was named ω–hydroxyemodin (OHM), prevented *agr* signaling by all four *S. aureus agr* alleles at concentrations that are nontoxic to eukaryotic cells and subinhibitory to bacterial growth. The component OHM inhibited QS by direct binding to AgrA and decreased the *S. aureus* bacterial load in a mouse model.

6.3 *In silico* methods find promising QS inhibitors

Computer–based approaches have been applied in the field of discovering anti–QS substances with the aim of saving experimental time and costs by preselecting promising candidates via virtual screening. The following approaches are related to *P. aeruginosa* and *S. aureus*. More details on *P. aeruginosa* QS system can be found in the published paper.

Preventing AI–regulator interactions

Annapoorani et al. [430] carried out a virtual screening approach to find LasR and RhlR QS inhibitors in *P. aeruginosa*. Out of 1,920 compounds, docking identified five promising candidate substrates for the LasR and RhlR receptors. They verified their potential to suppress the expression of virulence factors protease, elastase, and hemolysin, by in vitro experiments.

By application of Boolean network modeling, Schaadt et al. [431] presented an *in silico* multi–level modeling approach to study time–dependent properties of the *Las*, *Rhl*, and *Pqs* signaling systems of *P. aeruginosa*. Their aim was to investigate the regulatory and metabolic interplay between QS inhibitors, receptor antagonism, signaling molecules, and expression of the virulence factors elastase, rhamnolipids, and pyocyanin. In their simulations, they found that signaling molecules HHQ and PQS are decreased when expression of *pqsBCD* is suppressed by appropriate inhibitors. Using this network approach they were able to predict the quantitative impact of Pqs inhibitors and PqsR antagonists on QS.

Inhibit regulator binding to DNA

Leonard et al. [432] determined the crystal structure of the AgrA LytTR domain in *S. aureus* that is necessary to bind DNA. They subsequently applied fragment virtual screening to a small library consisting of 500 compounds and found three inhibitors that disrupted binding of AgrA to DNA.

6.4 Limits of QS therapeutic strategies

Although several new discoveries in the field of bacterial QS and in the development of promising inhibitors have been reported recently, significant research gaps remain. First of all, it is unclear whether all molecular components of QS systems and the respective regulators have been discovered up to date. For example, Miller et al. [433] recently identified novel antagonists of pyocyanin production in *P. aeruginosa* that appear to act through a pathway that is independent of the known regulators LasR and RhlR.

An important issue in antimicrobial drug development is the treatment of bacterial biofilms. Infections that are based on biofilms have a preference to be chronic and resistant to antibiotics [434]. Thereby, QS can help bacteria to regulate group behavior in these densely packed bacterial biofilms [434]. Here, a combination therapy of QS inhibitors and antibiotics could be beneficial: inhibitors can enhance the susceptibility of bacterial biofilms to a treatment with antibiotics that resulted in increased *in vitro* killing and *in vivo* survival rates [435].

Despite the fact that multiple drugs have been tested *in vitro* and *in vivo*, very few clinical trials involving QS inhibitors have been conducted or initiated. Only three clinical trials, with verified status, are reported in the publicly available Clinical Trials database [436]. Moreover, Scutera et al. [96] speculated that the interest of pharmaceutical companies in the development of QS inhibitors is only moderate based on the imbalance between high costs for developing new drugs while the market for these drugs seems to be restricted. They also suggested that the apparent advantage of avoiding drug resistance by targeting the signaling system may have the downside that strains with increased virulence could be selected.

Discovering the complex intricacies of QS systems and understanding the genetic, and possibly also epigenetic, mechanisms of bacterial adaptation under selective pressure are important research questions. For example, it is possible that when a certain signaling system of specific species is targeted, other (pathogenic) bacterial species a patient is infected with, may have an increased selective advantage. Moreover, bacteria may also become resistant to QS inhibitors. In the case of the *S. aureus agr* system this can occur via the up–regulation of efflux transporters [426]. Fortunately, the recently discovered QS inhibitors reviewed in our article and elsewhere [96] are nice tools for such mechanistic studies.

CHAPTER 7

Conclusion and outlook

In this thesis, we considered several regulatory levels of protein biosynthesis that together shape the complex diversity of phenotypes. First, we analyzed the association between the human methylome and nucleosome occupancy to explain experimentally observed DNA methylation patterns using a structural superimposition approach. This was followed by the development of a statistical model to predict alternative non–cognate translation start sites in the 5′ UTR based on a given mRNA sequence. Alternative translation is important for cellular function and was found to be associated with an adaption to environmental conditions such as cellular stress response. Besides innate factors that contribute to a phenotype, we were also interested in genome variations and their potential influence on protein biosynthesis. Mutations in genomic key elements are associated with (disease) phenotypes, whereby the exact location of a variation, like coding or regulatory sites, is of crucial importance to estimate the genome–wide impact. Thus, we analyzed mutation frequencies in genomic regions such as coding regions and the flanking sequences of transcription and translation start sites in human. Subsequently, we developed a tool that automatically scores the impact of mutations in a given genome by also integrating an underlying gene regulatory network. This software was then applied to two prokaryotic genomes, namely *Escherichia coli* and *Staphylococcus aureus*, to investigate the contribution of individual variations on antibiotic resistance.

As mentioned, we analyzed DNA methylation patterns and their association with the organization of DNA within the nucleosome core complex. For this, we computed accessibility scores based on a structural alignment of maintenance DNA methyltransferase DNMT1 and the nucleosome core complex (NCP147) X–ray structures [137]. The calculation of unfeasible binding positions between DNMT1 and NCP147 using a sterical clash model enabled the detection of accessible nucleosome–bound CpG sites at nucleotide resolution. Our thus derived scores were then statistically compared to experimentally observed methylation pattern. Experimental data was thereby based on the NOMe–seq technique that simultaneously detects GpC methylated sites to derive nucleosome occupied and depleted regions together with genome–wide methylome data [100]. In this study it was crucial to bring together these two regulatory factors, DNA methylation and nucleosome positioning, to unravel their relationship [100]. Our results suggest that the distribution of methylated CpG sites throughout the human genome is dependent on the accessibility of DNMT1 to the nucleosome–bound DNA. We could show that this pattern is only present in regions with higher nucleosome density compared to the local surrounding and was absent in regions with low nucleosome density. It was suggested that DNA methylation and nucleosome occupancy are dependent on each other, maybe even in a bidirectional way [147, 157, 158]. Our results support this hypothesis and highlight the important association between (epi)genetic marks to ensure the complex and dynamic genome regulation present in our cells. Nevertheless, several other factors such as histone modifications were reported to influence DNA methylation [437]. The consideration of present and absent histone modifications could therefore improve the understanding of this process and help to decipher underlying methylation patterns. Moreover, due to the specific nucleosome positioning and regulatory function of DNA methylation, we limited our studies to promoter regions. A genome–wide analysis of CpG methylation together with nucleosome occupancy in several genomic key elements such as coding regions or nucleosome–dense intron–exon boundaries [158] could greatly contribute to a better understanding of genome regulation

in eukaryotic cells.

Next, we analyzed the sequence–encoded differences between genome–wide experimentally confirmed alternative translation initiation sites and the remaining set of putative start sites within human 5′ UTRs. This knowledge on sequence–encoded differences between these groups was then used to develop statistical regression models, which assign translation initiation confidence scores to all putative start sites found in a given mRNA sequence and therewith predict their ability to initiate translation. We were able to demonstrate that alternative start sites detected by experimental ribosome profiling [170] and features based on mRNA sequence information can be used to build reliable prediction models with accuracies of about 80% for start codon and open reading frame prediction in human. All predicted start sites of one transcript are postulated to have the potential to initiate translation. They could, for example, be used in different tissues or in a specific cellular condition such as stress response. Although there already exist several other approaches to predict translational initiation start sites, at the time when we published our study, none of them considered all in– and out–of–frame AUG and near–cognate codons. Very recently, a group following up on our work published an open–source software with apparently higher accuracy that was trained using deep learning [438]. Our provided web service *PreTIS* considerably simplifies and assists the analysis of mRNA sequences in terms of prediction of possible translation start sites and their visualization. *PreTIS* can also be used to estimate the impact of individual mutations in the start site flanking regions. The analysis of mutations in these regions and in further defined genomic elements was investigated by conducting a genome–wide variation analysis (see below).

To generate accurate predictor functions, the reliability of the underlying dataset is crucial. The availability of thousands of alternative start sites detected using the ribosome profiling technology is a major step forward to decipher translational complexity and greatly supports the development of statistical models. Nevertheless, the available datasets are probably composed of some false start sites and do not contain some true start sites. This inhomogeneity can be based on experimental drawbacks and subsequent processing steps of the raw data. Moreover, we limited our studies to start sites located in the 5′ UTR due to their ability to extend known protein isoforms as well as to generate completely new proteins in case the start site is out–of–frame with the canonical AUG codon. Thus, one could expand the existing regression model to predict start sites located in the CDS and 3′ UTR. Furthermore, an investigation of potential sequence–encoded differences and similarities between start codons in the 5′ UTR, CDS, and 3′ UTR could be worth the effort to generate individual predictors dependent on the specific mRNA element. For instance, features such as codon or overall sequence conservation are likely to loose some significance due to a generally higher conservation of coding sequences compared to their flanking regions. Moreover, a dependency between individual start sites on a single mRNA with different Kozak consensus sequences was observed [43]. Therefore, a sophisticated approach that also considers initiation confidences of all start sites located in upstream direction could hence be beneficial to improve the underlying statistical model, when expanding the approach to the complete mRNA sequence. Finally, a consideration of several cell lines could further shed light on alternative codon usage, especially when assuming that non–canonical start sites are used differently across cell types.

Following this, we examined the human genome based on data on tens of millions of mutations from two major sequencing projects. We separately investigated several genomic regions and analyzed their functional relevance based on sequence conservation. A detailed analysis of mutation pattern around transcription and canonical as well as non–canonical translation start sites was carried out as well. This investigation of SNPs and indels from the 1000 Genomes Project [57] and the Genome of the Netherlands project [58, 59] revealed pronounced differences in the distribution of several variant types across genomic elements, such as promoters, 5′ UTRs, and coding exons. The coding start site coincided with a decrease in SNP density, which is in agreement with the expected strong conservation of protein–encoding sequences. Also, we noticed a decreased SNP and indel density at the TSS suggesting strong purifying selection against indels within intragenic regions. As described before [302], we found that indels are not only rare in open reading frames but also in potential regulatory elements such as CpG islands. In general, mutation frequencies found here were in accordance with earlier findings [250]. However, we discovered a remarkable amount of genes with a CpG dinucleotide

upstream of the TSS at position –1 that coincided with an elevated number of SNPs at this position. Applying DAVID enrichment analysis [292], we found that most of these genes are significantly enriched in the annotations "Phosphoprotein", "Alternative splicing" and "Protein binding". One might speculate that a mutation–prone methylated CpG dinucleotide at this position functions as cellular signal for transcriptional regulation of specific gene groups. With respect to translation initiation sites, our investigations showed that alternative start sites located in the human 5′ UTR exhibit a similar conservation tendency in their flanking region compared to annotated canonical AUG start sites. In doing so, we considered AUG and near–cognate as well as in– and out–of–frame with the annotated start site detected by experimental ribosome profiling [170]. We found a pronounced decrease in the number of SNPs at the start site itself, but also at prominent position –3, which was experimentally shown to be crucial for translation initiation [182, 183]. In general, alternative starts are not as conserved as canonical start sites. Nevertheless, the similar conservation pattern found confirms the importance of alternative start codons and their relevant contribution to the expansion of biological variety and complexity.

Finally, we developed the *MutaNET* software that supports and facilitates the investigation of individual mutations and their impact on gene function and regulation in a given genome. The sequential analysis steps provide a detailed report of different mutation types in distinct genomic elements and also allows their statistical comparison between gene groups, such as antibiotic resistant and non–antibiotic resistant genes. We provide different scores that are calculated based on the location of a mutation such as coding regions or transcription factor binding sites. These mutation scores help to estimate to which extent a mutation can influence an encoded protein sequence or impact regulator binding. Moreover, integration of an underlying gene regulatory network greatly helps in estimating the global impact of mutations on gene expression. We then applied our software to antibiotic resistant *Escherichia coli* and *Staphylococcus aureus* bacterial strains. Application of *MutaNET* to these resistance gene datasets considerably simplified the confirmation of known resistance mutations as well as the identification of novel candidate resistance mutations. It was also possible to decipher possible similar resistance mechanisms across these species. As further application, *MutaNET* could be used to detect novel resistance mutations in cancer cell lines and to estimate their impact on the human regulome. Moreover, it would be possible to decipher similar resistance and adaptation mechanisms across the kingdoms of life when considering that bacterial resistance against antibiotics and the development of chemoresistant cancer cells upon treatment with cytostatica are based on similar adaptation strategies such as increased drug efflux via transmembrane transporters [372, 439, 440].

In summary, various processes involved in protein biosynthesis must be tightly regulated to ensure normal cell behavior. Even small changes within specific regions, such as promoter hypermethylation, can cause transcriptional silencing of tumor suppressor genes and thus favor cancer development. Moreover, individual mutations are associated with disease phenotypes, whereas the location is of crucial importance. In this sense, the establishment and analysis of the underlying gene regulatory network together with an investigation of the affected protein domains can help to find candidate mutations for phenotypes such as antibiotic resistant pathogens. The selection of translation start sites is highly influenced by the (flanking) mRNA sequence. Machine learning can thus help to decipher these pattern and find novel experimentally undetected but important initiation sites that could be used in different cellular states or that encode alternative protein isoforms with essential functions in different cell types.

Abbreviations

1000G	1000 Genomes Project
3′ UTR	3′ untranslated region
5′ UTR	5′ untranslated region
5hmC	5–hydroxymethylation
5mC	5–methylcytosine
A	adenine
A–site	amino acid site
ADD	ATRX–DNMT3–DNMT3L
agr	accessory gene regulator
AI	autoinducers
AIP	autoinducing peptide
AUC	area under the ROC curve
BAH1/2	tandem bromo–adjacent homology
BAM	Binary Alignment Map
BED	Browser Extensible Data
BLAST	Basic Local Alignment Search Tool
bp	base pair
BWA	Burrows–Wheeler Alignment
C	cytosine
CAP	catabolite gene activator protein
CC	clonal complex
CCR5	C–C chemokine receptor type 5
cDNA	complementary DNA
CDS	coding DNA sequence
CERN	The European Organization for Nuclear Research
CES	translation end site
CGI	CpG island
ChIP–seq	chromatin immunoprecipitation combined with DNA sequencing
CHX	cycloheximide
CSS	Cascading Style Sheets
CSS	coding start site
CXXC	Cys–X–X–Cys
DNA	deoxyribonucleic acid

DNMT	DNA methyltransferase
DNMT1	DNA methyltransferase 1
DNMT3a	DNA methyltransferase 3a
DNMT3b	DNA methyltransferase 3b
DNMT3L	DNA methyltransferase 3–like protein
DOM	Document Object Model
dORF	downstream ORF
E–site	exit site
E. coli	Escherichia coli
ES cell	embryonic stem cell
FACS–seq	fluorescence-activated cell sorting and high-throughput DNA sequencing
FDR	false discovery rate
FN	false negative
FP	false positive
FPR	false positive rate
G	guanine
GFF	General Feature Format
GML	Graph Modeling Language
GO	Gene Ontology
GoNL	Genome of the Netherlands
GRN	gene regulatory network
GTP	guanosine triphosphate
GUI	graphical user interface
GWAS	genome–wide association studies
HEK293	human embryonic kidney 293
HNDR	high nucleosome density region
HSL	homoserine lactone
HTH	helix–turn–helix
HTML	Hypertext Markup Language
IC	initiation confidence
indel	insertion and deletion
JS	JavaScript
JSON	JavaScript Object Notation
kb	kilo base
LNDR	low nucleosome density region
LTM	lactimidomycin
Mb	mega base
MDRE	multidrug–resistant efflux

mRNA	messenger RNA
MRSA	methicillin–resistant Staphylococcus aureus
MUSCLE	MUltiple Sequence Comparison by Log–Expectation
NDR	nucleosome depleted region
NER	nucleotide excision repair
NGS	next–generation sequencing
NOMe–seq	Nucleosome Occupancy and Methylome sequencing
nt	nucleotide
OHM	ω–hydroxyemodin
ORF	open reading frame
P–site	polypeptide site
P. aeruginosa	Pseudomonas aeruginosa
PATRIC	Pathosystems Resource Integration Center
PCA	principal component analysis
PDB	Protein Data Bank
PFM	position frequency matrix
PHP	PHP: Hypertext Preprocessor
PWM	position weight matrix
PWWP	Pro–Trp–Trp–Pro
QS	quorum sensing
RBF	radial basis function
RefSeq	Reference Sequence
REST	Representational State Transfer
RMSD	root–mean–square deviation
RNA	ribonucleic acid
RNA–seq	RNA sequencing
ROC	Receiver Operating Characteristics
rRNA	ribosomal RNA
S. aureus	Staphylococcus aureus
S. epidermidis	Staphylococcus epidermidis
SA	suffix array
SAH	S–adenosyl–L–homocysteine
SAM	Sequence Alignment Map
SAMe	S–adenosyl–L–methionine
SNP	single nucleotide polymorphism
sORF	small open reading frame
ST	sequence type
SVG	Scalable Vector Graphics

SVM	support vector machine
T	thymine
TCGA	The Cancer Genome Atlas
TERT	telomerase reverse transcriptase
TES	transcription end site
TET	Ten–Eleven Translocation
TF	transcription factor
TFBS	transcription factor binding site
TIS	translation initiation site
TN	true negative
TP	true positive
TPR	true positive rate
tRNA	transfer RNA
TSS	transcription start site
U	uracil
uORF	upstream ORF
V. fischeri	Vibrio (Aliivibrio) fischeri
VCF	variant call format
vdW	van der Waals
W3C	World Wide Web Consortium
WGBS	whole–genome bisulfite sequencing
WGS	whole–genome sequencing

Supplementary material

Appendix A: Decipher DNA methylation patterns

Table A.1: Cohen's d values for different matching scores in HNDRs. The matching–score and thus Cohen's d values are dependent on tolerated sterical clash c_{thres} and methylation threshold m_{thres} parameters. For instance, c5m0 denotes that the score was calculated using $c_{thres} = 5$ and $m_{thres} = 0$. The numbers are visualized in Figure 2.13.

i	c5m0	c5m10	c5m20	c10m0	c10m10	c10m20	c20m0	c20m10	c20m20	c50m0	c50m10	c50m20
1	–0.02	–0.01	0.01	–0.01	0.0	0.01	0.01	0.0	0.0	0.02	0.0	–0.03
2	0.1	0.12	0.11	0.05	0.06	0.06	–0.03	–0.03	–0.03	–0.17	–0.21	–0.19
3	0.13	0.14	0.14	0.07	0.08	0.07	–0.04	–0.05	–0.04	–0.23	–0.25	–0.24
4	0.2	0.22	0.2	0.11	0.12	0.11	–0.06	–0.07	–0.06	–0.32	–0.35	–0.33
5	0.23	0.24	0.23	0.13	0.13	0.13	–0.07	–0.07	–0.07	–0.37	–0.38	–0.37
6	0.29	0.3	0.29	0.16	0.17	0.16	–0.09	–0.1	–0.09	–0.46	–0.48	–0.46
7	0.39	0.39	0.37	0.22	0.22	0.21	–0.13	–0.12	–0.12	–0.61	–0.62	–0.6
8	0.47	0.47	0.44	0.26	0.26	0.24	–0.15	–0.15	–0.14	–0.74	–0.75	–0.72
9	0.51	0.51	0.48	0.29	0.29	0.26	–0.16	–0.16	–0.15	–0.79	–0.79	–0.76
10	0.58	0.58	0.55	0.33	0.32	0.31	–0.18	–0.18	–0.17	–0.91	–0.91	–0.88
11	0.67	0.66	0.63	0.38	0.37	0.35	–0.21	–0.2	–0.19	–1.03	–1.02	–0.99
12	0.71	0.69	0.66	0.4	0.38	0.36	–0.23	–0.22	–0.2	–1.11	–1.08	–1.05
13	0.8	0.77	0.74	0.45	0.43	0.41	–0.25	–0.24	–0.23	–1.27	–1.24	–1.22
14	0.83	0.8	0.77	0.46	0.44	0.43	–0.28	–0.27	–0.26	–1.31	–1.26	–1.23
15	0.94	0.9	0.87	0.53	0.51	0.48	–0.28	–0.27	–0.27	–1.48	–1.43	–1.43
16	0.97	0.94	0.93	0.54	0.52	0.5	–0.31	–0.29	–0.29	–1.6	–1.55	–1.58
17	1.03	0.99	1.0	0.55	0.53	0.53	–0.33	–0.31	–0.31	–1.71	–1.66	–1.74
18	1.1	1.06	1.07	0.6	0.58	0.58	–0.34	–0.33	–0.32	–1.86	–1.81	–1.86
19	1.18	1.14	1.14	0.64	0.62	0.61	–0.38	–0.35	–0.35	–2.02	–1.96	–2.05
20	1.28	1.23	1.21	0.69	0.66	0.64	–0.4	–0.39	–0.39	–2.19	–2.12	–2.16
21	1.33	1.27	1.22	0.75	0.71	0.68	–0.41	–0.39	–0.37	–2.15	–2.07	–2.08
22	1.26	1.21	1.21	0.69	0.66	0.64	–0.42	–0.4	–0.4	–2.15	–2.1	–2.2
23	1.36	1.32	1.29	0.77	0.74	0.72	–0.41	–0.39	–0.36	–2.18	–2.13	–2.18
24	1.22	1.16	1.1	0.69	0.66	0.62	–0.46	–0.44	–0.4	–1.97	–1.89	–1.85
25	1.4	1.34	1.31	0.74	0.69	0.66	–0.5	–0.48	–0.46	–2.46	–2.37	–2.44
26	1.42	1.36	1.3	0.79	0.75	0.72	–0.43	–0.41	–0.37	–2.14	–2.08	–1.98
27	1.65	1.65	1.51	0.97	0.98	0.87	–0.49	–0.45	–0.41	–2.59	–2.53	–2.4
28	1.52	1.48	1.34	0.9	0.88	0.76	–0.42	–0.43	–0.38	–2.42	–2.46	–2.3
29	1.46	1.5	1.53	0.79	0.77	0.82	–0.39	–0.43	–0.3	–2.14	–2.28	–2.25
30	1.04	0.99	0.92	0.72	0.67	0.47	–0.24	–0.18	–0.34	–1.73	–1.76	–1.71
31	1.16	1.14	1.24	0.63	0.61	0.7	–0.34	–0.33	–0.27	–2.28	–2.34	–2.68
32	2.07	1.99	1.91	1.21	1.16	1.07	–0.41	–0.38	–0.42	–2.73	–2.66	–2.56
33	2.06	1.99	2.2	1.25	1.28	1.26	–0.37	–0.27	–0.43	–3.36	–3.4	–3.91
34	5.0	3.59	4.29	3.46	2.75	4.25	–1.96	–1.59	0.15	–5.84	–5.35	–5.86

Table A.2: Cohen's d values for different matching scores in LNDRs. The matching–score and thus Cohen's d values are dependent on tolerated sterical clash c_{thres} and methylation threshold m_{thres} parameters. For instance, c5m0 denotes that the score was calculated using $c_{thres} = 5$ and $m_{thres} = 0$. The numbers are visualized in Figure 2.14.

i	c5m0	c5m10	c5m20	c10m0	c10m10	c10m20	c20m0	c20m10	c20m20	c50m0	c50m10	c50m20
1	0.04	0.04	0.06	0.02	0.03	0.04	–0.01	–0.01	–0.02	–0.07	–0.08	–0.12
2	0.14	0.15	0.16	0.08	0.08	0.09	–0.04	–0.04	–0.05	–0.24	–0.26	–0.3
3	0.16	0.16	0.18	0.08	0.08	0.09	–0.05	–0.05	–0.05	–0.28	–0.28	–0.32
4	0.18	0.18	0.19	0.1	0.1	0.11	–0.05	–0.05	–0.06	–0.3	–0.3	–0.34
5	0.18	0.18	0.2	0.1	0.1	0.11	–0.06	–0.06	–0.06	–0.32	–0.31	–0.36
6	0.22	0.2	0.23	0.12	0.11	0.12	–0.07	–0.06	–0.07	–0.38	–0.35	–0.4
7	0.22	0.2	0.23	0.12	0.11	0.12	–0.06	–0.06	–0.07	–0.37	–0.33	–0.4
8	0.24	0.21	0.25	0.13	0.12	0.13	–0.07	–0.07	–0.08	–0.4	–0.36	–0.44
9	0.23	0.2	0.24	0.12	0.11	0.13	–0.07	–0.07	–0.07	–0.38	–0.34	–0.41
10	0.23	0.2	0.24	0.12	0.11	0.13	–0.07	–0.06	–0.07	–0.38	–0.33	–0.41
11	0.25	0.21	0.26	0.14	0.12	0.15	–0.07	–0.06	–0.07	–0.4	–0.35	–0.44
12	0.24	0.2	0.24	0.13	0.11	0.13	–0.08	–0.07	–0.08	–0.4	–0.35	–0.43
13	0.24	0.21	0.27	0.14	0.12	0.15	–0.07	–0.06	–0.08	–0.39	–0.33	–0.45
14	0.26	0.22	0.29	0.14	0.12	0.16	–0.08	–0.06	–0.09	–0.42	–0.36	–0.49
15	0.27	0.23	0.31	0.15	0.13	0.17	–0.08	–0.07	–0.1	–0.43	–0.37	–0.51
16	0.28	0.25	0.34	0.15	0.14	0.19	–0.1	–0.08	–0.11	–0.46	–0.4	–0.57
17	0.26	0.23	0.33	0.15	0.13	0.18	–0.08	–0.07	–0.1	–0.43	–0.37	–0.54
18	0.28	0.24	0.36	0.16	0.14	0.2	–0.08	–0.07	–0.11	–0.45	–0.38	–0.59
19	0.32	0.28	0.39	0.18	0.15	0.21	–0.09	–0.08	–0.12	–0.51	–0.44	–0.65
20	0.35	0.3	0.41	0.2	0.17	0.23	–0.13	–0.11	–0.14	–0.57	–0.48	–0.69
21	0.38	0.34	0.46	0.21	0.19	0.25	–0.14	–0.12	–0.15	–0.63	–0.56	–0.76
22	0.41	0.36	0.48	0.22	0.2	0.26	–0.13	–0.11	–0.16	–0.66	–0.59	–0.81
23	0.44	0.39	0.52	0.25	0.22	0.29	–0.14	–0.12	–0.16	–0.71	–0.63	–0.86
24	0.42	0.37	0.52	0.24	0.21	0.3	–0.12	–0.1	–0.14	–0.66	–0.58	–0.82
25	0.43	0.38	0.48	0.23	0.21	0.26	–0.13	–0.12	–0.15	–0.69	–0.61	–0.82
26	0.49	0.44	0.53	0.31	0.28	0.31	–0.13	–0.1	–0.16	–0.72	–0.64	–0.86
27	0.38	0.32	0.49	0.19	0.16	0.25	–0.17	–0.15	–0.2	–0.65	–0.58	–0.84
28	0.45	0.39	0.62	0.26	0.22	0.36	–0.14	–0.13	–0.16	–0.7	–0.61	–0.99
29	0.57	0.47	0.67	0.39	0.32	0.42	–0.08	–0.08	–0.15	–0.78	–0.67	–1.06
30	0.67	0.6	0.88	0.35	0.32	0.49	–0.24	–0.2	–0.22	–0.93	–0.83	–1.3
31	0.87	0.86	1.16	0.5	0.48	0.64	–0.23	–0.2	–0.36	–1.39	–1.42	–2.16
32	0.45	0.44	0.96	0.22	0.2	0.45	–0.24	–0.23	–0.32	–0.74	–0.78	–1.54
33	0.9	0.88	1.24	0.54	0.49	0.5	–0.36	–0.35	–0.65	–1.92	–2.23	–2.71
34	0.93	1.31	1.57	0.66	0.91	1.12	0.46	0.5	0.44	–0.23	–0.4	–1.18
35	–1.11	–1.25	0.0	–1.51	–1.58	–0.38	–1.51	–1.7	–1.41	–1.0	–2.5	–9.19

Table A.3: Number of CpGs in HNDRs. Frequency and percentages of sliding windows with specific number of CpGs in HNDRs. In total, 2,277,851 sliding windows were considered with an average of 7.12 ± 4.88 CpGs and a median of 6.0 CpGs per window.

i	Nbr. of CpGs = i		Nbr. of CpGs $\geq$ i	
	Frequency	Percentage	Frequency	Percentage
1	161652	7.1%	2277851	100.0%
2	252595	11.09%	2116199	92.9%
3	199454	8.76%	1863604	81.81%
4	226055	9.92%	1664150	73.06%
5	189821	8.33%	1438095	63.13%
6	188064	8.26%	1248274	54.8%
7	164968	7.24%	1060210	46.54%
8	151376	6.65%	895242	39.3%
9	128280	5.63%	743866	32.66%
10	111503	4.9%	615586	27.02%
11	94823	4.16%	504083	22.13%
12	80939	3.55%	409260	17.97%
13	67733	2.97%	328321	14.41%
14	56286	2.47%	260588	11.44%
15	46635	2.05%	204302	8.97%
16	37215	1.63%	157667	6.92%
17	29316	1.29%	120452	5.29%
18	23394	1.03%	91136	4.0%
19	19429	0.85%	67742	2.97%
20	14169	0.62%	48313	2.12%
21	11247	0.49%	34144	1.5%
22	8117	0.36%	22897	1.01%
23	5386	0.24%	14780	0.65%
24	3499	0.15%	9394	0.41%
25	2388	0.1%	5895	0.26%
26	1579	0.07%	3507	0.15%
27	798	0.04%	1928	0.08%
28	513	0.02%	1130	0.05%
29	293	0.01%	617	0.03%
30	129	0.01%	324	0.01%
31	103	0.0%	195	0.01%
32	60	0.0%	92	0.0%
33	29	0.0%	32	0.0%
34	3	0.0%	3	0.0%

Table A.4: Number of CpGs in LNDRs. Frequency and percentages of sliding windows with specific number of CpGs in LNDRs. In total, we considered 4,390,558 sliding windows with an average of 9.0 ± 5.22 CpGs and a median of 8.0 CpGs per window.

i	Nbr. of CpGs = i		Nbr. of CpGs $\geq$ i	
	Frequency	Percentage	Frequency	Percentage
1	155775	3.55%	4390558	100.0%
2	257549	5.87%	4234783	96.45%
3	250314	5.7%	3977234	90.59%
4	306887	6.99%	3726920	84.88%
5	313941	7.15%	3420033	77.9%
6	330375	7.52%	3106092	70.74%
7	326022	7.43%	2775717	63.22%
8	315711	7.19%	2449695	55.79%
9	302240	6.88%	2133984	48.6%
10	279142	6.36%	1831744	41.72%
11	254193	5.79%	1552602	35.36%
12	232048	5.29%	1298409	29.57%
13	201351	4.59%	1066361	24.29%
14	174937	3.98%	865010	19.7%
15	148615	3.38%	690073	15.72%
16	124139	2.83%	541458	12.33%
17	104189	2.37%	417319	9.5%
18	82682	1.88%	313130	7.13%
19	65324	1.49%	230448	5.25%
20	49456	1.13%	165124	3.76%
21	36344	0.83%	115668	2.63%
22	26192	0.6%	79324	1.81%
23	19586	0.45%	53132	1.21%
24	12671	0.29%	33546	0.76%
25	8333	0.19%	20875	0.48%
26	5460	0.12%	12542	0.29%
27	3059	0.07%	7082	0.16%
28	1818	0.04%	4023	0.09%
29	1106	0.03%	2205	0.05%
30	545	0.01%	1099	0.03%
31	249	0.01%	554	0.01%
32	218	0.0%	305	0.01%
33	51	0.0%	87	0.0%
34	30	0.0%	36	0.0%
35	6	0.0%	6	0.0%

Appendix B: Mutations in genomic elements

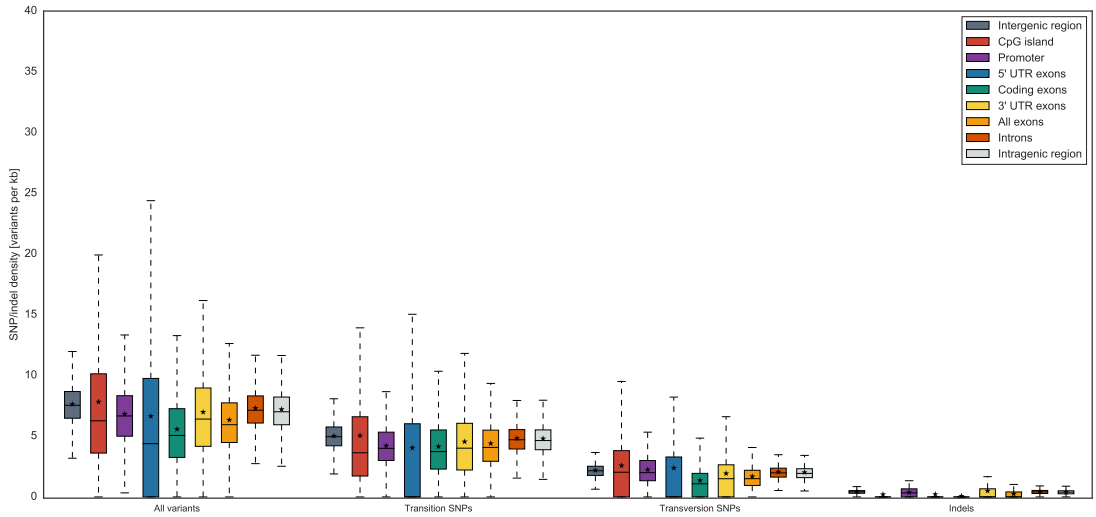

Figure B.1: Mutations in key genomic elements considering the GoNL data. Shown are SNP and indel densities for all genetic elements considering the GoNL data. The horizontal line (–) represents the median value, the asterisk ($\star$) denotes the mean value.

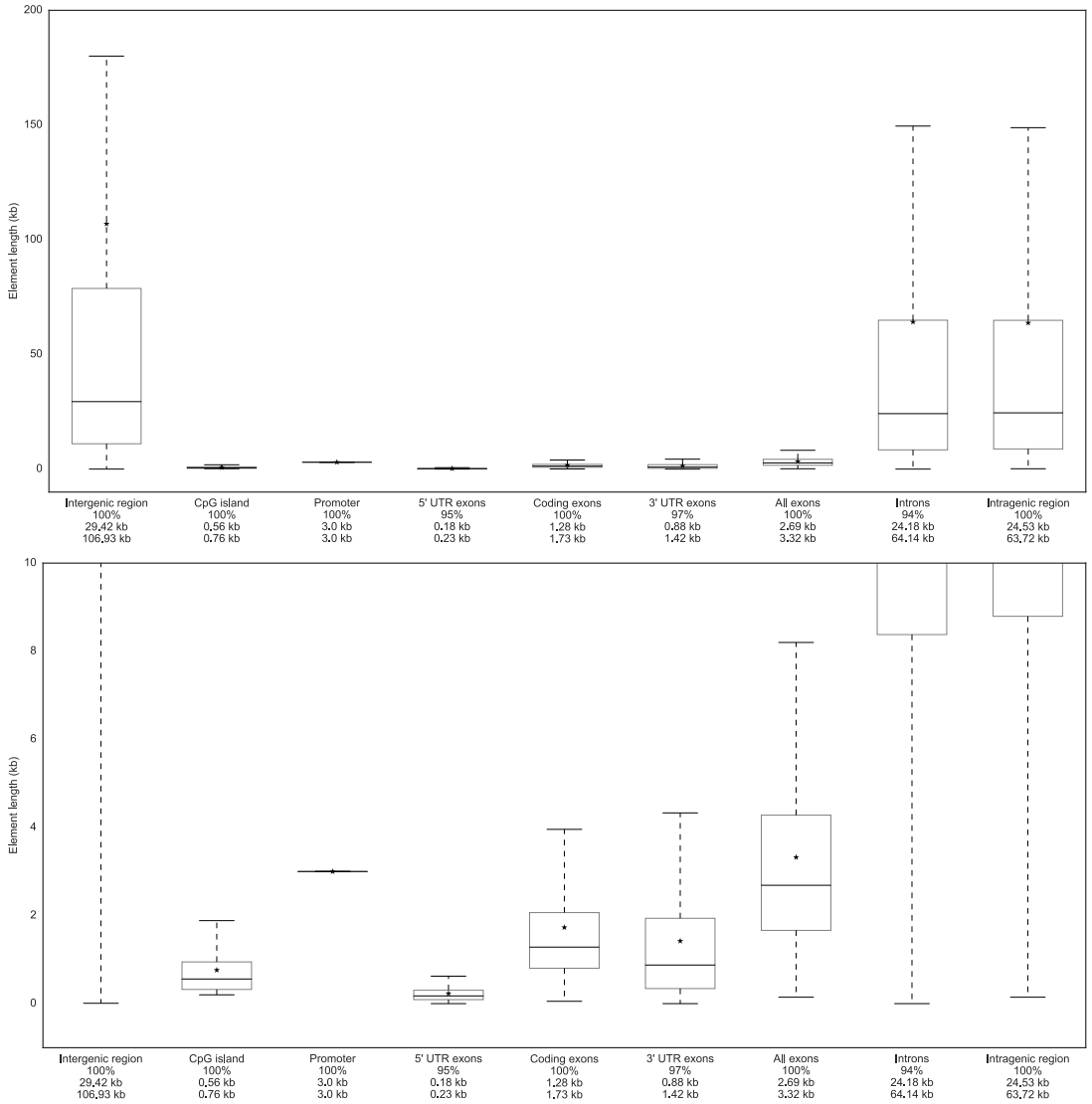

Figure B.2: Lengths of the genomic elements. Upper and lower panel show the same data but use different y–scales (0–200 kb and 0–10 kb). Each box plot is labeled with the number of genes exhibiting this element [%], the median ($\star$) [kb] and mean values (–) [kb]. 5′ UTRs are the shortest genetic elements with an median value of 180 bp (0.18 kb), whereas intergenic regions (median: $\sim$29 kb), introns (median: $\sim$24 kb) and the intragenic region (median: $\sim$25 kb) are the largest elements.

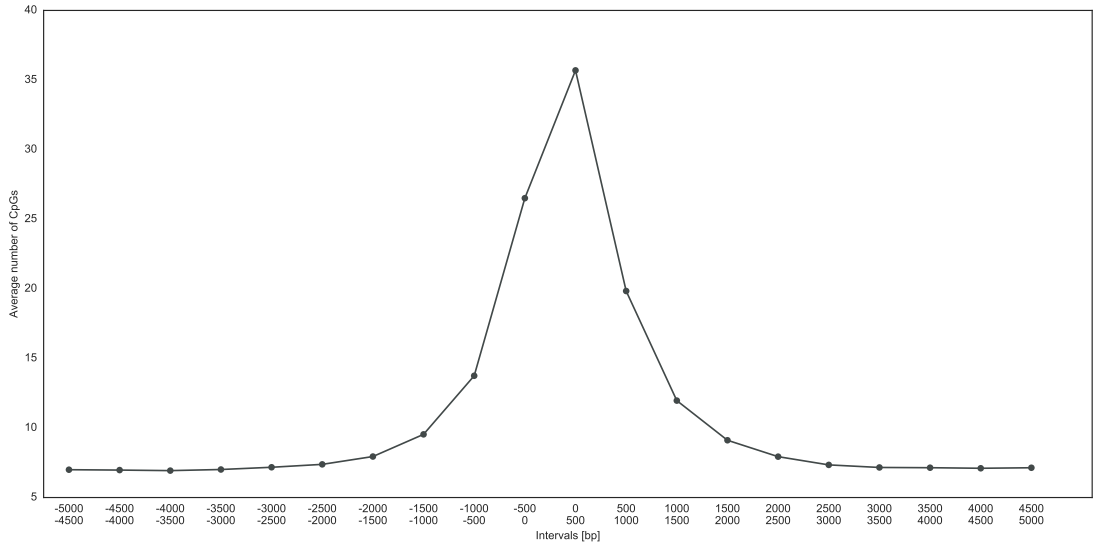

Figure B.3: Average number of CpGs around the TSS. We considered a window from –5000 bp to +5000 bp around the TSS of RefSeq genes. The number of CpGs peaks at the TSS, which was found earlier [132].

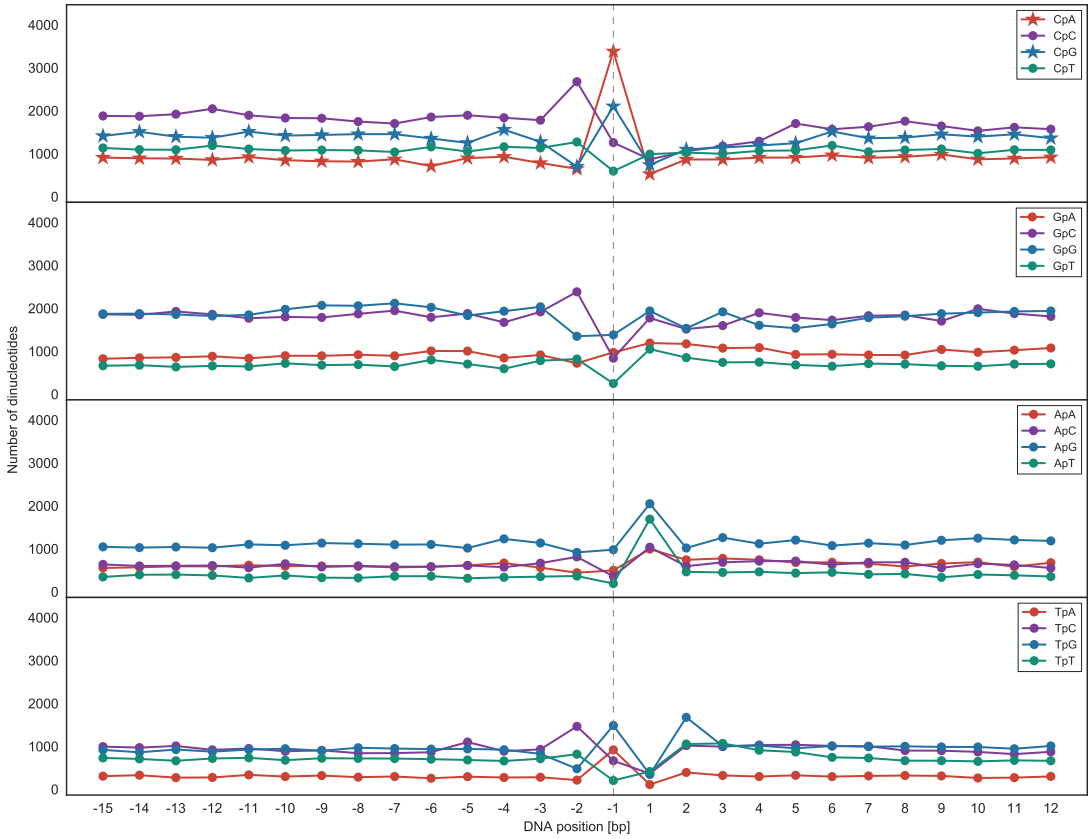

Figure B.4: Dinucleotide distribution in the TSS flanking region. We considered all RefSeq genes that remained after filtering. All 16 possible dinucleotides were compared. Position 1 denotes the first intragenic nucleotide. A CpG dinucleotide at position –1 means the C is located at position –1 while the G resides at position +1.

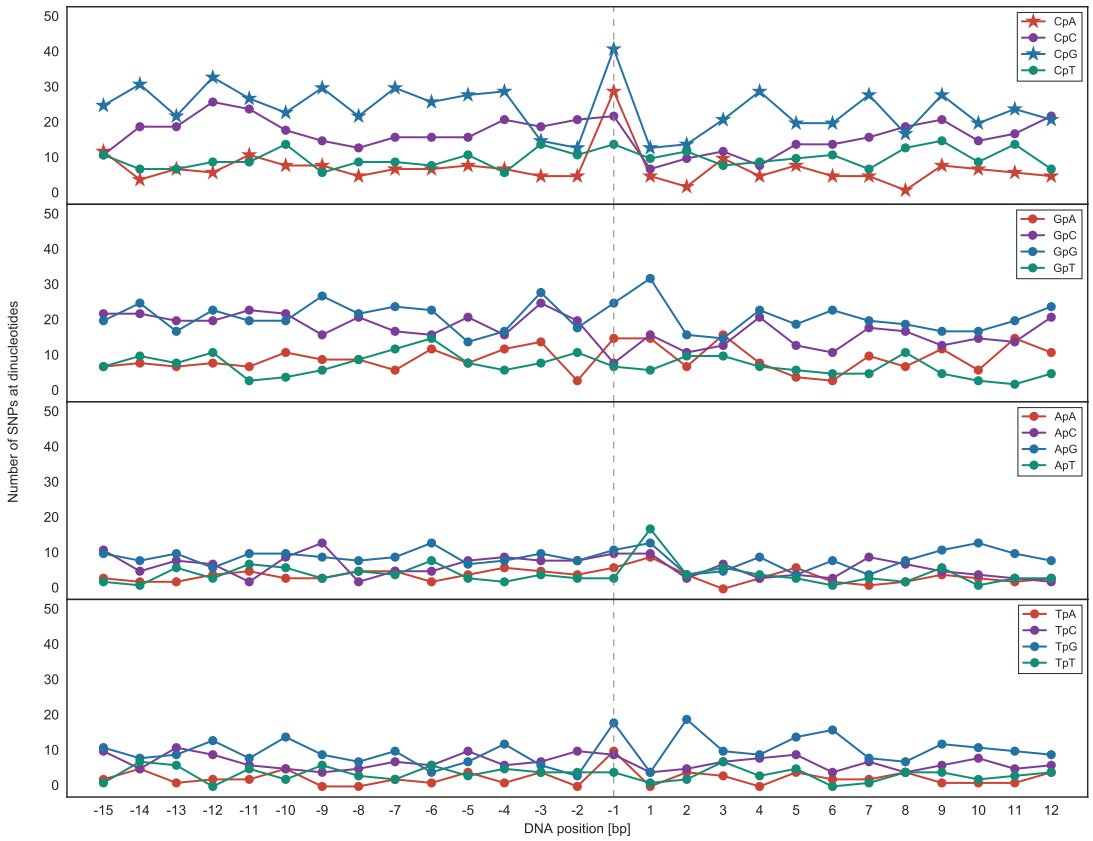

Figure B.5: Mutations at dinucleotides considering the 1000G data. Shown is the number of SNPs at individual dinucleotides in the flanking region of the TSS considering the 1000G data. SNPs were analyzed at individual dinucleotides in the flanking region of the TSS. Position 1 denotes the first intragenic nucleotide.

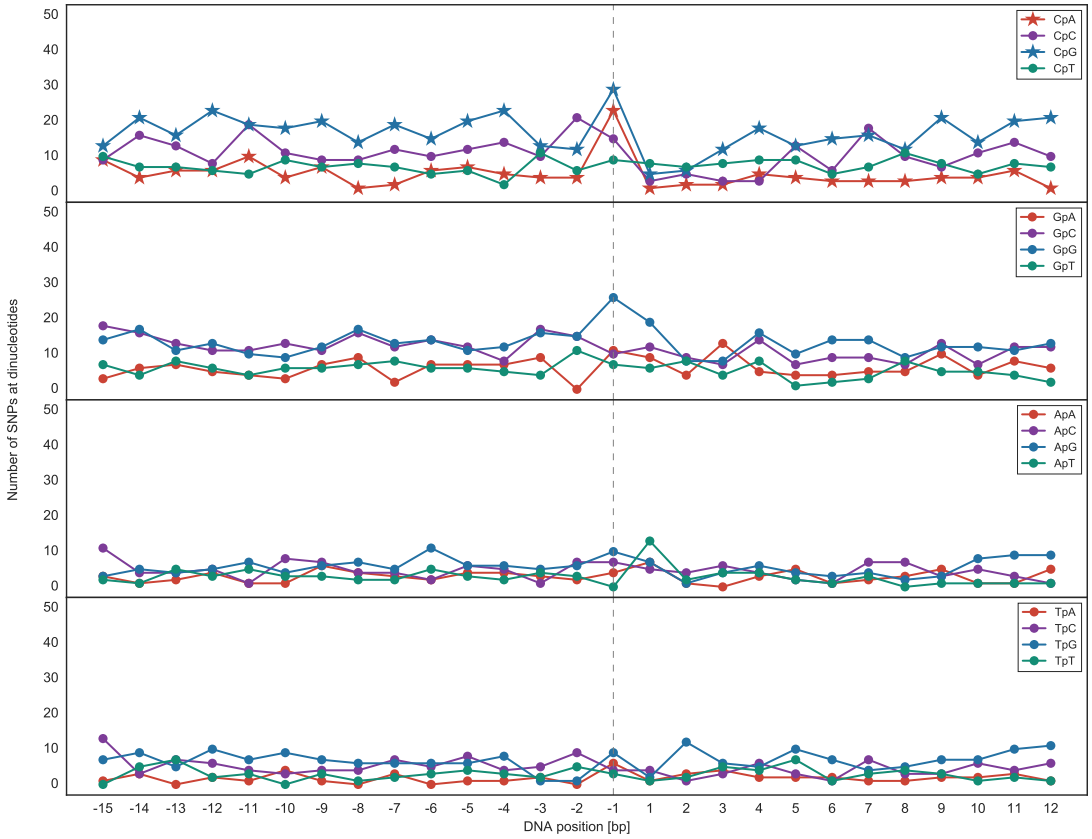

Figure B.6: Mutations at dinucleotides considering the GoNL data. SNPs were analyzed at individual dinucleotides in the flanking region of the TSS. Depicted is the number of SNPs at individual dinucleotides in the flanking region of the TSS considering the GoNL data. Position 1 denotes the first intragenic nucleotide.

Table B.1: Results of DAVID functional annotation considering Ap* dinucleotides. Results of DAVID functional annotation [292] for all genes differentiated by the dinucleotide present at TSS position –1. Duplicated terms from different databases were deleted and the one with smallest p–value was retained. Shown are terms with corrected p–value of $p < 0.05$ (Benjamini correction). If no significant GO term enrichment was found for a dinucleotide gene subset, only the first two terms are displayed for convenience. Number of genes (RefSeq identifiers accepted by DAVID tool) of every subgroup is given in brackets.

		Term	# Genes	% Genes	Adjusted p–value
ApA (530 genes)	1.	Detection of chemical stimulus involved in sensory perception of smell	97	18.3	1.4×10^{-61}
	2.	Olfactory receptor	97	18.3	1.1×10^{-61}
	3.	Olfactory receptor activity	97	18.3	3.5×10^{-61}
	4.	Olfaction	97	18.3	7.1×10^{-61}
	5.	Olfactory transduction	97	18.3	3.4×10^{-53}
	6.	Sensory transduction	103	19.4	5.8×10^{-52}
	7.	G-protein coupled receptor activity	108	20.4	1.6×10^{-49}
	8.	GPCR, rhodopsin–like, 7TM	105	19.8	1.5×10^{-46}
	9.	G–protein coupled receptor	112	21.1	1.5×10^{-46}
	10.	G–protein coupled receptor, rhodopsin–like	103	19.4	6.2×10^{-46}
	11.	Transducer	113	21.3	2.4×10^{-44}
	12.	G-protein coupled receptor signaling pathway	112	21.1	7.4×10^{-42}
	13.	Receptor	132	24.9	4.6×10^{-29}
	14.	Topological domain: extracellular	161	30.4	9.4×10^{-20}
	15.	Disulfide bond	162	30.6	2.8×10^{-18}
	16.	Topological domain: cytoplasmic	179	33.8	1.3×10^{-17}
	17.	Glycosylation site: N–linked	202	38.1	3.9×10^{-16}
	18.	Odorant binding	26	4.9	3.3×10^{-16}
	19.	Disulfide bond	173	32.6	2.8×10^{-16}
	20.	Glycoprotein	209	39.4	1.1×10^{-14}
	21.	Cell membrane	156	29.4	3.5×10^{-13}
	22.	Detection of chemical stimulus involved in sensory perception	22	4.2	1.2×10^{-11}
	23.	Plasma membrane	184	34.7	2.7×10^{-11}
	24.	Sensory perception of smell	24	4.5	1.4×10^{-9}
	25.	Transmembrane region	202	38.1	1.0×10^{-8}
	26.	Transmembrane helix	216	40.8	1.8×10^{-8}
	27.	Integral component of membrane	199	37.5	9.1×10^{-8}
	28.	Transmembrane	216	40.8	2.2×10^{-8}
	29.	Transmembrane signaling receptor activity	22	4.2	6.1×10^{-5}
	30.	Membrane	255	48.1	4.3×10^{-5}
ApC (392 genes)	1.	Extracellular space	45	11.5	3.6×10^{-1}
	2.	Biosynthesis of antibiotics	12	3.1	2.8×10^{-1}
ApG (1,011 genes)	1.	Domain: Leucine–zipper	17	1.7	5.7×10^{-1}
	2.	DNA-binding region: basic motif	20	2.0	9.5×10^{-1}
ApT (223 genes)	1.	Commissural neuron axon guidance	3	1.3	9.8×10^{-1}
	2.	Disulfide bond	55	24.7	7.6×10^{-1}

Table B.2: Results of DAVID functional annotation considering Cp* dinucleotides. Results of DAVID functional annotation [292] for all genes differentiated by the dinucleotide present at TSS position –1. Duplicated terms from different databases were deleted and the one with smallest p–value was retained. Shown are terms with corrected p–value of $p < 0.05$ (Benjamini correction). If no significant GO term enrichment was found for a dinucleotide gene subset, only the first two terms are displayed for convenience. Number of genes (RefSeq identifiers accepted by DAVID tool) of every subgroup is given in brackets.

		Term	# Genes	% Genes	Adjusted p–value
CpA (3,412 genes)	1.	Topological domain: extracellular	587	17.2	1.9×10^{-5}
	2.	G–protein coupled receptor signaling pathway	213	6.2	1.8×10^{-4}
	3.	Olfactory receptor activity	112	3.3	7.5×10^{-5}
	4.	GPCR, rhodopsin–like, 7TM	174	5.1	1.5×10^{-4}
	5.	G–protein coupled receptor, rhodopsin-like	170	5.0	1.0×10^{-4}
	6.	Olfactory receptor	112	3.3	7.4×10^{-5}
	7.	Detection of chemical stimulus involved in sensory perception of smell	111	3.3	2.5×10^{-4}
	8.	Olfaction	114	3.3	5.4×10^{-5}
	9.	Topological domain: cytoplasmic	701	20.5	3.7×10^{-4}
	10.	Transmembrane region	981	28.8	2.5×10^{-4}
	11.	Transducer	207	6.1	5.3×10^{-5}
	12.	G–protein coupled receptor	194	5.7	5.4×10^{-5}
	13.	Glycosylation site: N–linked (GlcNAc...)	844	24.7	8.1×10^{-4}
	14.	Olfactory transduction	112	3.3	2.5×10^{-4}
	15.	Integral component of membrane	989	29.0	1.0×10^{-3}
	16.	Glycoprotein	906	26.6	1.9×10^{-4}
	17.	Sensory transduction	146	4.3	1.9×10^{-4}
	18.	G-protein coupled receptor activity	163	4.8	1.7×10^{-3}
	19.	Transmembrane	1076	31.5	2.5×10^{-4}
	20.	Transmembrane helix	1072	31.4	2.4×10^{-4}
	21.	Cell membrane	635	18.6	4.1×10^{-4}
	22.	Receptor	350	10.3	6.8×10^{-4}
	23.	Nucleosome core	31	0.9	1.6×10^{-3}
	24.	Sensory perception of smell	44	1.3	8.6×10^{-2}
	25.	Histone–fold	35	1.0	4.0×10^{-2}
	26.	Keratin–associated matrix	11	0.3	3.6×10^{-2}
	27.	Disulfide bond	670	19.6	3.6×10^{-3}
	28.	TAF	10	0.3	4.1×10^{-2}
	29.	H4	10	0.3	4.1×10^{-2}
	30.	TATA box binding protein associated factor (TAF)	10	0.3	3.9×10^{-2}
	31.	Histone H4	10	0.3	3.9×10^{-2}
CpC (1,289 genes)	1.	Mitochondrion	119	9.2	3.6×10^{-1}
	2.	Neuron projection	30	2.3	5.0×10^{-1}
CpG (2,136 genes)	see Table 4.4				
CpT (625 genes)	1.	Signal	153	24.5	7.1×10^{-1}
	2.	Secreted	84	13.4	4.8×10^{-1}

Table B.3: Results of DAVID functional annotation considering Gp* dinucleotides. Results of DAVID functional annotation [292] for all genes differentiated by the dinucleotide present at TSS position –1. Duplicated terms from different databases were deleted and the one with smallest p–value was retained. Shown are terms with corrected p–value of $p < 0.05$ (Benjamini correction). If no significant GO term enrichment was found for a dinucleotide gene subset, only the first two terms are displayed for convenience. Number of genes (RefSeq identifiers accepted by DAVID tool) of every subgroup is given in brackets.

		Term	# Genes	% Genes	Adjusted p–value
GpA (1,005 genes)	1.	G–protein coupled receptor activity	61	6.1	9.0×10^{-3}
	2.	G–protein coupled receptor, rhodopsin–like	60	6.0	4.7×10^{-2}
	3.	G-protein coupled receptor	68	6.8	2.0×10^{-2}
	4.	Odorant binding	16	1.6	2.4×10^{-2}
	5.	GPCR, rhodopsin–like, 7TM	60	6.0	4.8×10^{-2}
GpC (871 genes)	1.	Calcium transport	16	1.8	4.0×10^{-2}
GpG (1,413 genes)	1.	Splice variant	637	45.1	3.2×10^{-2}
GpT (282 genes)	1.	IL12 and Stat4 Dependent Signaling Pathway in Th1 Development	4	1.4	1.6×10^{-1}
	2.	snRNA processing	3	1.1	1.0

Table B.4: Results of DAVID functional annotation considering Tp* dinucleotides. Results of DAVID functional annotation [292] for all genes differentiated by the dinucleotide present at TSS position –1. Duplicated terms from different databases were deleted and the one with smallest p–value was retained. Shown are terms with corrected p–value of $p < 0.05$ (Benjamini correction). If no significant GO term enrichment was found for a dinucleotide gene subset, only the first two terms are displayed for convenience. Number of genes (RefSeq identifiers accepted by DAVID tool) of every subgroup is given in brackets.

		Term	# Genes	% Genes	Adjusted p–value
TpA (951 genes)	1.	Olfaction	49	5.2	6.1×10^{-6}
	2.	Olfactory receptor activity	48	5.0	1.5×10^{-5}
	3.	Detection of chemical stimulus involved in sensory perception of smell	48	5.0	5.0×10^{-5}
	4.	Olfactory receptor	48	5.0	4.1×10^{-5}
	5.	Olfactory transduction	50	5.3	1.3×10^{-5}
	6.	Sensory transduction	58	6.1	8.5×10^{-5}
	7.	G–protein coupled receptor activity	59	6.2	7.2×10^{-3}
TpC (698 genes)	1.	Transcription factor activity, sequence–specific DNA binding	59	8.5	1.0×10^{-1}
	2.	DNA–binding	100	14.3	7.1×10^{-1}
TpG (1,519 genes)	1.	Xenobiotic metabolic process	16	1.1	9.7×10^{-1}
	2.	Protease inhibitor	20	1.3	8.2×10^{-1}
TpT (238 genes)	1.	Kinetochore binding	3	1.3	3.4×10^{-1}
	2.	Negative regulation of transcription, DNA–templated	16	6.7	7.6×10^{-1}

Bibliography

[1] R. Dahm. Friedrich Miescher and the discovery of DNA. *Dev. Biol.*, 278(2):274–288, 2005.

[2] F. Hoppe-Seyler. Über die chemische Zusammensetzung der Eiterzellen. *Med.-Chem. Unters.*, 4:486–501, 1871.

[3] J. D. Watson and F. H. Crick. Molecular structure of nucleic acids; a structure for deoxyribose nucleic acid. *Nature*, 171(4356):737–738, 1953.

[4] R. E. Franklin and R. G. Gosling. Molecular configuration in sodium thymonucleate. *Nature*, 171(4356):740–741, 1953.

[5] M. H. Wilkins, A. R. Stokes, and H. R. Wilson. Molecular structure of deoxypentose nucleic acids. *Nature*, 171(4356):738–740, 1953.

[6] L. A. Pray. Discovery of DNA Structure and Function: Watson and Crick. *Nature Education*, 1(1):100, 2008.

[7] M. J. Zvelebil and J. O. Baum. *Understanding Bioinformatics*. Garland Science, Taylor & Francis Group, LLC, 1 edition, 2008.

[8] R. W. Taylor and D. M. Turnbull. Mitochondrial DNA mutations in human disease. *Nat. Rev. Genet.*, 6(5):389–402, 2005.

[9] H. Chial and J. Craig. mtDNA and mitochondrial diseases. *Nature Education*, 1(1):217, 2008.

[10] C. A. Davey, D. F. Sargent, K. Luger, A. W. Maeder, and T. J. Richmond. Solvent mediated interactions in the structure of the nucleosome core particle at 1.9 a resolution. *J. Mol. Biol.*, 319(5):1097–1113, 2002.

[11] K. Luger, A. W. Mader, R. K. Richmond, D. F. Sargent, and T. J. Richmond. Crystal structure of the nucleosome core particle at 2.8 A resolution. *Nature*, 389(6648):251–260, 1997.

[12] K. Luger, M. L. Dechassa, and D. J. Tremethick. New insights into nucleosome and chromatin structure: an ordered state or a disordered affair? *Nat. Rev. Mol. Cell Biol.*, 13(7): 436–447, 2012.

[13] P. B. Talbert and S. Henikoff. Histone variants on the move: substrates for chromatin dynamics. *Nat. Rev. Mol. Cell Biol.*, 18(2):115–126, 2017.

[14] C. W. Jang, Y. Shibata, J. Starmer, D. Yee, and T. Magnuson. Histone H3.3 maintains genome integrity during mammalian development. *Genes Dev.*, 29(13):1377–1392, 2015.

[15] G. Felsenfeld and M. Groudine. Controlling the double helix. *Nature*, 421(6921):448–453, 2003.

[16] D. J. Tremethick. Higher-order structures of chromatin: the elusive 30 nm fiber. *Cell*, 128 (4):651–654, 2007.

[17] C. L. Woodcock and S. Dimitrov. Higher-order structure of chromatin and chromosomes. *Curr. Opin. Genet. Dev.*, 11(2):130–135, 2001.

[18] S. Venkatesh and J. L. Workman. Histone exchange, chromatin structure and the regulation of transcription. *Nat. Rev. Mol. Cell Biol.*, 16(3):178–189, 2015.

[19] D. Vermaak, K. Ahmad, and S. Henikoff. Maintenance of chromatin states: an open-and-shut case. *Curr. Opin. Cell Biol.*, 15(3):266–274, 2003.

[20] D. E. Schones and K. Zhao. Genome-wide approaches to studying chromatin modifications. *Nat. Rev. Genet.*, 9(3):179–191, 2008.

[21] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne. The Protein Data Bank. *Nucleic Acids Res.*, 28(1):235–242, 2000.

[22] The PyMOL Molecular Graphics System, Version 1.3, Schrödinger, LLC. https://www.pymol.org/. 2015.

[23] S. Clancy. DNA Transcription. *Nature Education*, 1(1):41, 2008.

[24] E. de Klerk and P. A. 't Hoen. Alternative mRNA transcription, processing, and translation: insights from RNA sequencing. *Trends Genet.*, 31(3):128–139, 2015.

[25] T. E. Dever. Molecular biology. A new start for protein synthesis. *Science*, 336(6089): 1645–1646, 2012.

[26] A. Saunders, L. J. Core, and J. T. Lis. Breaking barriers to transcription elongation. *Nat. Rev. Mol. Cell Biol.*, 7(8):557–567, 2006.

[27] T. Phillips and L. Hoopes. Transcription factors and transcriptional control in eukaryotic cells. *Nature Education*, 1(1):119, 2008.

[28] FANTOM Consortium, RIKEN PMI, and CLST (DGT). A promoter-level mammalian expression atlas. *Nature*, 507(7493):462–470, 2014.

[29] S. Clancy. RNA Splicing: Introns, Exons and Spliceosome. *Nature Education*, 1(1):31, 2008.

[30] Q. Pan, O. Shai, L. J. Lee, B. J. Frey, and B. J. Blencowe. Deep surveying of alternative splicing complexity in the human transcriptome by high-throughput sequencing. *Nat. Genet.*, 40(12):1413–1415, 2008.

[31] S. Danckwardt, M. W. Hentze, and A. E. Kulozik. 3' end mRNA processing: molecular mechanisms and implications for health and disease. *EMBO J.*, 27(3):482–498, 2008.

[32] E. de Klerk, A. Venema, S. Y. Anvar, J. J. Goeman, O. Hu, C. Trollet, G. Dickson, J. T. den Dunnen, S. M. van der Maarel, V. Raz, and P. A. 't Hoen. Poly(A) binding protein nuclear 1 levels affect alternative polyadenylation. *Nucleic Acids Res.*, 40(18):9089–9101, 2012.

[33] T. Ni, Y. Yang, D. Hafez, W. Yang, K. Kiesewetter, Y. Wakabayashi, U. Ohler, W. Peng, and J. Zhu. Distinct polyadenylation landscapes of diverse human tissues revealed by a modified PA-seq strategy. *BMC Genomics*, 14:615, 2013.

[34] S. Clancy and W. Brown. Translation: DNA to mRNA to Protein. *Nature Education*, 1(1): 101, 2008.

[35] F. H. Crick, L. Barnett, S. Brenner, and R. J. Watts-Tobin. General nature of the genetic code for proteins. *Nature*, 192:1227–1232, 1961.

[36] M. Kozak. The scanning model for translation: an update. *J. Cell Biol.*, 108:229–241, 1989.

[37] R. J. Jackson, C. U. Hellen, and T. V. Pestova. The mechanism of eukaryotic translation initiation and principles of its regulation. *Nat. Rev. Mol. Cell Biol.*, 11(2):113–127, 2010.

[38] P. B. Moore and T. A. Steitz. After the ribosome structures: how does peptidyl transferase work? *RNA*, 9(2):155–159, 2003.

[39] P. Nissen, J. Hansen, N. Ban, P. B. Moore, and T. A. Steitz. The structural basis of ribosome activity in peptide bond synthesis. *Science*, 289(5481):920–930, 2000.

[40] D. S. Peabody. Translation initiation at non-AUG triplets in mammalian cells. *J. Biol. Chem.*, 264(9):5031–5035, 1989.

[41] M. Kozak. Context effects and inefficient initiation at non-AUG codons in eucaryotic cell-free translation systems. *Mol. Cell. Biol.*, 9(11):5073–5080, 1989.

[42] I. P. Ivanov, A. E. Firth, A. M. Michel, J. F. Atkins, and P. V. Baranov. Identification of evolutionarily conserved non-AUG-initiated N-terminal extensions in human coding sequences. *Nucleic Acids Res.*, 39:4220–4234, 2011.

[43] S. Lee, B. Liu, S. Lee, S. X. Huang, B. Shen, and S. B. Qian. Global mapping of translation initiation sites in mammalian cells at single-nucleotide resolution. *Proc. Natl. Acad. Sci. U.S.A.*, 109:E2424–2432, 2012.

[44] N. T. Ingolia, L. F. Lareau, and J. S. Weissman. Ribosome profiling of mouse embryonic stem cells reveals the complexity and dynamics of mammalian proteomes. *Cell*, 147(4): 789–802, 2011.

[45] L. Wasserman. *All of Statistics: A Concise Course in Statistical Inference*. Springer Texts in Statistics. Springer New York, 2010.

[46] F. Wilcoxon. Individual comparisons of grouped data by ranking methods. *J. Econ. Entomol.*, 39:269, 1946.

[47] J. Zar. *Biostatistical Analysis*. Pearson International Edition, 2010.

[48] Student. The probable error of mean. *Biometrika*, 6(1):1–25, 1908.

[49] C. Eisenhart. On the Transition from "Student's" z to "Student's" t. *The American Statistician*, 33(1):6–12, 1979.

[50] G. M. Sullivan and R. Feinn. Using Effect Size - or Why the P Value Is Not Enough. *J. Grad. Med. Educ.*, 4(3):279–282, 2012.

[51] M. Lin, H. C. Lucas, and G. Shmueli. Research Commentary - Too Big to Fail: Large Samples and the p-Value Problem. *Inf. Syst. Res.*, 24(4):906–917, 2013.

[52] J. Cohen. *Statistical Power Analysis for the Behavioral Sciences*. Lawrence Erlbaum Associates, Hillsdale, Second edition, 1988.

[53] F. Schmidt, N. Gasparoni, G. Gasparoni, K. Gianmoena, C. Cadenas, J. K. Polansky, P. Ebert, K. Nordstrom, M. Barann, A. Sinha, S. Frohler, J. Xiong, A. Dehghani Amirabad, F. Behjati Ardakani, B. Hutter, G. Zipprich, B. Felder, J. Eils, B. Brors, W. Chen, J. G. Hengstler, A. Hamann, T. Lengauer, P. Rosenstiel, J. Walter, and M. H. Schulz. Combining transcription factor binding affinities with open-chromatin data for accurate gene expression prediction. *Nucleic Acids Res.*, 45(1):54–66, 2017.

[54] L. Calviello, N. Mukherjee, E. Wyler, H. Zauber, A. Hirsekorn, M. Selbach, M. Landthaler, B. Obermayer, and U. Ohler. Detecting actively translated open reading frames in ribosome profiling data. *Nat. Methods*, 2015.

[55] R. J. Kinsella, A. Kähäri, S. Haider, J. Zamora, G. Proctor, G. Spudich, J. Almeida-King, D. Staines, P. Derwent, A. Kerhornou, P. Kersey, and P. Flicek. Ensembl BioMarts: a hub for data retrieval across taxonomic space. *Database (Oxford)*, 2011:bar030, 2011.

[56] W. J. Kent, C. W. Sugnet, T. S. Furey, K. M. Roskin, T. H. Pringle, A. M. Zahler, and D. Haussler. The human genome browser at UCSC. *Genome Res.*, 12(6):996–1006, 2002.

[57] The 1000 Genomes Project Consortium. A global reference for human genetic variation. *Nature*, 526(7571):68–74, 2015.

[58] The Genome of the Netherlands Consortium. Whole-genome sequence variation, population structure and demographic history of the Dutch population. *Nat. Genet.*, 46(8): 818–825, 2014.

[59] L. C. Francioli, P. P. Polak, A. Koren, A. Menelaou, S. Chun, I. Renkens, C. M. van Duijn, M. Swertz, C. Wijmenga, G. van Ommen, P. E. Slagboom, D. I. Boomsma, K. Ye, V. Guryev, P. F. Arndt, W. P. Kloosterman, P. I. de Bakker, and S. R. Sunyaev. Genome-wide patterns and properties of de novo mutations in humans. *Nat. Genet.*, 47(7):822–826, 2015.

[60] The National Center for Biotechnology Information (NCBI). BioProject. URL `https://www.ncbi.nlm.nih.gov/bioproject/`. Accessed on 02.10.2017.

[61] M. Hamed, D. P. Nitsche-Schmitz, U. Ruffing, M. Steglich, J. Dordel, D. Nguyen, J. H. Brink, G. S. Chhatwal, M. Herrmann, U. Nubel, V. Helms, and L. von Muller. Whole genome sequence typing and microarray profiling of nasal and blood stream methicillin-resistant Staphylococcus aureus isolates: Clues to phylogeny and invasiveness. *Infect. Genet. Evol.*, 36:475–482, 2015.

[62] S. Gama-Castro, H. Salgado, A. Santos-Zavaleta, D. Ledezma-Tejeida, L. Muniz-Rascado, J. S. Garcia-Sotelo, K. Alquicira-Hernandez, I. Martinez-Flores, L. Pannier, J. A. Castro-Mondragon, A. Medina-Rivera, H. Solano-Lira, C. Bonavides-Martinez, E. Perez-Rueda, S. Alquicira-Hernandez, L. Porron-Sotelo, A. Lopez-Fuentes, A. Hernandez-Koutoucheva, V. Del Moral-Chavez, F. Rinaldi, and J. Collado-Vides. RegulonDB version 9.0: high-level integration of gene regulation, coexpression, motif clustering and beyond. *Nucleic Acids Res.*, 44(D1):D133–143, 2016.

[63] AureoWiki consortium. AureoWiki. URL `http://aureowiki.med.uni-greifswald.de/`. Accessed on 02.10.2017.

[64] A. R. Wattam, D. Abraham, O. Dalay, T. L. Disz, T. Driscoll, J. L. Gabbard, J. J. Gillespie, R. Gough, D. Hix, R. Kenyon, D. Machi, C. Mao, E. K. Nordberg, R. Olson, R. Overbeek, G. D. Pusch, M. Shukla, J. Schulman, R. L. Stevens, D. E. Sullivan, V. Vonstein, A. Warren, R. Will, M. J. Wilson, H. S. Yoo, C. Zhang, Y. Zhang, and B. W. Sobral. PATRIC, the bacterial bioinformatics database and analysis resource. *Nucleic Acids Res.*, 42(Database issue):D581–591, 2014.

[65] J. J. Gillespie, A. R. Wattam, S. A. Cammer, J. L. Gabbard, M. P. Shukla, O. Dalay, T. Driscoll, D. Hix, S. P. Mane, C. Mao, E. K. Nordberg, M. Scott, J. R. Schulman, E. E. Snyder, D. E. Sullivan, C. Wang, A. Warren, K. P. Williams, T. Xue, H. S. Yoo, C. Zhang, Y. Zhang, R. Will, R. W. Kenyon, and B. W. Sobral. PATRIC: the comprehensive bacterial bioinformatics resource with a focus on human pathogenic species. *Infect. Immun.*, 79 (11):4286–4298, 2011.

[66] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *J. Mol. Biol.*, 215(3):403–410, 1990.

[67] C. Camacho, G. Coulouris, V. Avagyan, N. Ma, J. Papadopoulos, K. Bealer, and T. L. Madden. BLAST+: architecture and applications. *BMC Bioinformatics*, 10:421, 2009.

[68] R. Lorenz, S. H. Bernhart, C. Honer Zu Siederdissen, H. Tafer, C. Flamm, P. F. Stadler, and I. L. Hofacker. ViennaRNA Package 2.0. *Algorithms Mol Biol.*, 6:26, 2011.

[69] R. C. Edgar. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res.*, 32(5):1792–1797, 2004.

[70] R. C. Edgar. MUSCLE: a multiple sequence alignment method with reduced time and space complexity. *BMC Bioinformatics*, 5:113, 2004.

[71] B. Langmead, C. Trapnell, M. Pop, and S. L. Salzberg. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biol.*, 10(3):R25, 2009.

[72] P. Danecek, A. Auton, G. Abecasis, C. A. Albers, E. Banks, M. A. DePristo, R. E. Handsaker, G. Lunter, G. T. Marth, S. T. Sherry, G. McVean, R. Durbin, and 1000-Genomes-Project-Analysis-Group. The variant call format and VCFtools. *Bioinformatics*, 27(15): 2156–2158, 2011.

[73] H. Li and R. Durbin. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 25(14):1754–1760, 2009.

[74] H. Li and R. Durbin. Fast and accurate long-read alignment with Burrows-Wheeler transform. *Bioinformatics*, 26(5):589–595, 2010.

[75] H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, and R. Durbin. The Sequence Alignment/Map format and SAMtools. *Bioinformatics*, 25 (16):2078–2079, 2009.

[76] H. Li. A statistical framework for SNP calling, mutation discovery, association mapping and population genetical parameter estimation from sequencing data. *Bioinformatics*, 27 (21):2987–2993, 2011.

[77] D. C. Koboldt, Q. Zhang, D. E. Larson, D. Shen, M. D. McLellan, L. Lin, C. A. Miller, E. R. Mardis, L. Ding, and R. K. Wilson. VarScan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome Res.*, 22(3):568–576, 2012.

[78] A. R. Quinlan and I. M. Hall. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics*, 26(6):841–842, 2010.

[79] J. D. Hunter. Matplotlib: A 2D graphics environment. *Comput. Sci. Eng.*, 9(3):90–95, 2007.

[80] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.*, 12:2825–2830, 2011.

[81] W. McKinney. Data Structures for Statistical Computing in Python. *Proceedings of the 9th Python in Science Conference*, pages 51–56, 2010.

[82] W. McKinney. Pandas: a Foundational Python Library for Data Analysis and Statistics. *Python for High Performance and Scientific Computing*, 2011.

[83] SQLite. https://www.sqlite.org/. Accessed on 27.07.2017.

[84] G. Häring. Sqlite3: DB–API 2.0 interface for sqlite databases.

[85] S. van der Walt, S. C. Colbert, and G. Varoquaux. The NumPy Array: A Structure for Efficient Numerical Computation. *Comput. Sci. Eng.*, 13(2):22–30, 2011.

[86] E. Jones, T. Oliphant, P. Peterson, et al. SciPy: Open source scientific tools for Python, 2001. URL `http://www.scipy.org/`. Accessed on 27.07.2017.

[87] P. J. Cock, T. Antao, J. T. Chang, B. A. Chapman, C. J. Cox, A. Dalke, I. Friedberg, T. Hamelryck, F. Kauff, B. Wilczynski, and M. J. de Hoon. Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics*, 25(11): 1422–1423, 2009.

[88] M. Waskom, O. Botvinnik, P. Hobson, J. P. Cole, Y. Halchenko, S. Hoyer, A. Miles, T. Augspurger, T. Yarkoni, T. Megies, L. P. Coelho, D. Wehner, cynddl, E. Ziegler, diego0020, Y. V. Zaytsev, T. Hoppe, S. Seabold, P. Cloud, M. Koskinen, K. Meyer, A. Qalieh, and D. Allan. Seaborn: v0.5.0. 2014. URL `https://seaborn.pydata.org/`. Accessed on 27.07.2017.

[89] T. Hamelryck and B. Manderick. PDB file parser and structure class implemented in Python. *Bioinformatics*, 19(17):2308–2310, 2003.

[90] B. Jin, Y. Li, and K. D. Robertson. DNA methylation: superior or subordinate in the epigenetic hierarchy? *Genes Cancer*, 2(6):607–617, 2011.

[91] S. B. Baylin. DNA methylation and gene silencing in cancer. *Nat. Clin. Pract. Oncol.*, 2 Suppl 1:4–11, 2005.

[92] D. Jjingo, A. B. Conley, S. V. Yi, V. V. Lunyak, and I. K. Jordan. On the presence and role of human gene-body DNA methylation. *Oncotarget*, 3(4):462–474, 2012.

[93] S. R. Starck, J. C. Tsai, K. Chen, M. Shodiya, L. Wang, K. Yahiro, M. Martins-Green, N. Shastri, and P. Walter. Translation from the 5′ untranslated region shapes the integrated stress response. *Science*, 351(6272):aad3867, 2016.

[94] V. G. Cheung and R. S. Spielman. Genetics of human gene expression: mapping DNA variants that influence gene expression. *Nat. Rev. Genet.*, 10(9):595–604, 2009.

[95] A. C. Syvänen. Accessing genetic variation: genotyping single nucleotide polymorphisms. *Nat. Rev. Genet.*, 2(12):930–942, 2001.

[96] S. Scutera, M. Zucca, and D. Savoia. Novel approaches for the design and discovery of quorum-sensing inhibitors. *Expert Opin. Drug Discov.*, 9(4):353–366, 2014.

[97] C. H. Waddington. The epigenotype. *Endeavour*, 1:18–20, 1942.

[98] D. E. Handy, R. Castro, and J. Loscalzo. Epigenetic modifications: basic mechanisms and role in cardiovascular disease. *Circulation*, 123(19):2145–2156, 2011.

[99] Z. D. Smith and A. Meissner. DNA methylation: roles in mammalian development. *Nat. Rev. Genet.*, 14(3):204–220, 2013.

[100] T. K. Kelly, Y. Liu, F. D. Lay, G. Liang, B. P. Berman, and P. A. Jones. Genome-wide mapping of nucleosome positioning and DNA methylation within individual DNA molecules. *Genome Res.*, 22(12):2497–2506, 2012.

[101] R. Lister and J. R. Ecker. Finding the fifth base: genome-wide sequencing of cytosine methylation. *Genome Res.*, 19(6):959–966, 2009.

[102] S. Pennings, J. Allan, and C. S. Davey. DNA methylation, nucleosome formation and positioning. *Brief. Funct. Genomic Proteomic*, 3(4):351–361, 2005.

[103] S. Dhe-Paganon, F. Syeda, and L. Park. DNA methyl transferase 1: regulatory mechanisms and implications in health and disease. *Int. J. Biochem. Mol. Biol.*, 2(1):58–66, 2011.

[104] R. Lister, M. Pelizzola, R. H. Dowen, R. D. Hawkins, G. Hon, J. Tonti-Filippini, J. R. Nery, L. Lee, Z. Ye, Q. M. Ngo, L. Edsall, J. Antosiewicz-Bourget, R. Stewart, V. Ruotti, A. H. Millar, J. A. Thomson, B. Ren, and J. R. Ecker. Human DNA methylomes at base resolution show widespread epigenomic differences. *Nature*, 462(7271):315–322, 2009.

[105] V. Patil, R. L. Ward, and L. B. Hesson. The evidence for functional non-CpG methylation in mammalian cells. *Epigenetics*, 9(6):823–828, 2014.

[106] S. E. Pinney. Mammalian Non-CpG Methylation: Stem Cells and Beyond. *Biology (Basel)*, 3(4):739–751, 2014.

[107] S. Kim, N. K. Yu, and B. K. Kaang. CTCF as a multifunctional protein in genome regulation and gene expression. *Exp. Mol. Med.*, 47:e166, 2015.

[108] T. Clouaire and I. Stancheva. Methyl-CpG binding proteins: specialized transcriptional repressors or structural components of chromatin? *Cell. Mol. Life Sci.*, 65(10):1509–1522, 2008.

[109] M. S. Bartolomei and A. C. Ferguson-Smith. Mammalian genomic imprinting. *Cold Spring Harb. Perspect. Biol.*, 3(7), 2011.

[110] B. Panning and R. Jaenisch. DNA hypomethylation can activate Xist expression and silence X-linked genes. *Genes Dev.*, 10(16):1991–2002, 1996.

[111] E. Li, C. Beard, and R. Jaenisch. Role for DNA methylation in genomic imprinting. *Nature*, 366(6453):362–365, 1993.

[112] R. K. Slotkin and R. Martienssen. Transposable elements and the epigenetic regulation of the genome. *Nat. Rev. Genet.*, 8(4):272–285, 2007.

[113] J. A. Law and S. E. Jacobsen. Establishing, maintaining and modifying DNA methylation patterns in plants and animals. *Nat. Rev. Genet.*, 11(3):204–220, 2010.

[114] M. Ehrlich, M. A. Gama-Sosa, L. H. Huang, R. M. Midgett, K. C. Kuo, R. A. McCune, and C. Gehrke. Amount and distribution of 5-methylcytosine in human DNA from different types of tissues of cells. *Nucleic Acids Res.*, 10(8):2709–2721, 1982.

[115] B. Zhang, Y. Zhou, N. Lin, R. F. Lowdon, C. Hong, R. P. Nagarajan, J. B. Cheng, D. Li, M. Stevens, H. J. Lee, X. Xing, J. Zhou, V. Sundaram, G. Elliott, J. Gu, T. Shi, P. Gascard, M. Sigaroudinia, T. D. Tlsty, T. Kadlecek, A. Weiss, H. O'Geen, P. J. Farnham, C. L. Maire, K. L. Ligon, P. A. Madden, A. Tam, R. Moore, M. Hirst, M. A. Marra, B. Zhang, J. F. Costello, and T. Wang. Functional DNA methylation differences between tissues, cell types, and across individuals discovered using the M&M algorithm. *Genome Res.*, 23(9): 1522–1540, 2013.

[116] M. Ehrlich. DNA hypomethylation in cancer cells. *Epigenomics*, 1(2):239–259, 2009.

[117] M. J. Hoffmann and W. A. Schulz. Causes and consequences of DNA hypomethylation in human cancer. *Biochem. Cell Biol.*, 83(3):296–321, 2005.

[118] M. Ehrlich. DNA methylation in cancer: too much, but also too little. *Oncogene*, 21(35): 5400–5413, 2002.

[119] S. Kriaucionis and N. Heintz. The nuclear DNA base 5-hydroxymethylcytosine is present in Purkinje neurons and the brain. *Science*, 324(5929):929–930, 2009.

[120] R. Richa and R. P. Sinha. Hydroxymethylation of DNA: an epigenetic marker. *EXCLI J*, 13:592–610, 2014.

[121] H. Wu and Y. Zhang. Mechanisms and functions of Tet protein-mediated 5-methylcytosine oxidation. *Genes Dev.*, 25(23):2436–2452, 2011.

[122] M. Wossidlo, T. Nakamura, K. Lepikhov, C. J. Marques, V. Zakhartchenko, M. Boiani, J. Arand, T. Nakano, W. Reik, and J. Walter. 5-Hydroxymethylcytosine in the mammalian zygote is linked with epigenetic reprogramming. *Nat. Commun.*, 2:241, 2011.

[123] J. U. Guo, Y. Su, C. Zhong, G. L. Ming, and H. Song. Hydroxylation of 5-methylcytosine by TET1 promotes active DNA demethylation in the adult brain. *Cell*, 145(3):423–434, 2011.

[124] F. Delhommeau, S. Dupont, V. Della Valle, C. James, S. Trannoy, A. Masse, O. Kosmider, J. P. Le Couedic, F. Robert, A. Alberdi, Y. Lecluse, I. Plo, F. J. Dreyfus, C. Marzac, N. Casadevall, C. Lacombe, S. P. Romana, P. Dessen, J. Soulier, F. Viguie, M. Fontenay, W. Vainchenker, and O. A. Bernard. Mutation in TET2 in myeloid cancers. *N. Engl. J. Med.*, 360(22):2289–2301, 2009.

[125] S. M. Langemeijer, R. P. Kuiper, M. Berends, R. Knops, M. G. Aslanyan, M. Massop, E. Stevens-Linders, P. van Hoogen, A. G. van Kessel, R. A. Raymakers, E. J. Kamping, G. E. Verhoef, E. Verburgh, A. Hagemeijer, P. Vandenberghe, T. de Witte, B. A. van der Reijden, and J. H. Jansen. Acquired mutations in TET2 are common in myelodysplastic syndromes. *Nat. Genet.*, 41(7):838–842, 2009.

[126] M. Y. Shah and J. D. Licht. DNMT3A mutations in acute myeloid leukemia. *Nat. Genet.*, 43(4):289–290, 2011.

[127] B. Jin and K. D. Robertson. DNA methyltransferases, DNA damage repair, and cancer. *Adv. Exp. Med. Biol.*, 754:3–29, 2013.

[128] C. Jiang and Z. Zhao. Directionality of point mutation and 5-methylcytosine deamination rates in the chimpanzee genome. *BMC Genomics*, 7:316, 2006.

[129] C. Schmutte, A. S. Yang, R. W. Beart, and P. A. Jones. Base excision repair of U:G mismatches at a mutational hotspot in the p53 gene is more efficient than base excision repair of T:G mismatches in extracts of human colon tumors. *Cancer Res.*, 55(17):3742–3746, 1995.

[130] A. Bird, M. Taggart, M. Frommer, O. J. Miller, and D. Macleod. A fraction of the mouse genome that is derived from islands of nonmethylated, CpG-rich DNA. *Cell*, 40(1):91–99, 1985.

[131] F. Larsen, G. Gundersen, R. Lopez, and H. Prydz. CpG islands as gene markers in the human genome. *Genomics*, 13(4):1095–1107, 1992.

[132] S. Saxonov, P. Berg, and D. L. Brutlag. A genome-wide analysis of CpG dinucleotides in the human genome distinguishes two distinct classes of promoters. *Proc. Natl. Acad. Sci. U.S.A.*, 103(5):1412–1417, 2006.

[133] A. M. Deaton and A. Bird. CpG islands and the regulation of transcription. *Genes Dev.*, 25(10):1010–1022, 2011.

[134] J. Tazi and A. Bird. Alternative chromatin structure at CpG islands. *Cell*, 60(6):909–920, 1990.

[135] D. Macleod, J. Charlton, J. Mullins, and A. P. Bird. Sp1 sites in the mouse aprt gene promoter are required to prevent methylation of the CpG island. *Genes Dev.*, 8(19):2282–2292, 1994.

[136] M. B. Stadler, R. Murr, L. Burger, R. Ivanek, F. Lienert, A. Scholer, E. van Nimwegen, C. Wirbelauer, E. J. Oakeley, D. Gaidatzis, V. K. Tiwari, and D. Schubeler. DNA-binding factors shape the mouse methylome at distal regulatory regions. *Nature*, 480(7378):490–495, 2011.

[137] J. Song, O. Rechkoblit, T. H. Bestor, and D. J. Patel. Structure of DNMT1-DNA complex reveals a role for autoinhibition in maintenance DNA methylation. *Science*, 331(6020): 1036–1040, 2011.

[138] X. Guo, L. Wang, J. Li, Z. Ding, J. Xiao, X. Yin, S. He, P. Shi, L. Dong, G. Li, C. Tian, J. Wang, Y. Cong, and Y. Xu. Structural insight into autoinhibition and histone H3-induced activation of DNMT3A. *Nature*, 517(7536):640–644, 2015.

[139] H. Denis, M. N. Ndlovu, and F. Fuks. Regulation of mammalian DNA methyltransferases: a route to new mechanisms. *EMBO Rep.*, 12(7):647–656, 2011.

[140] D. Jia, R. Z. Jurkowska, X. Zhang, A. Jeltsch, and X. Cheng. Structure of Dnmt3a bound to Dnmt3L suggests a model for de novo DNA methylation. *Nature*, 449(7159):248–251, 2007.

[141] K. D. Robertson and P. A. Jones. Dynamic interrelationships between DNA replication, methylation, and repair. *Am. J. Hum. Genet.*, 61(6):1220–1224, 1997.

[142] M. Wigler, D. Levy, and M. Perucho. The somatic replication of DNA methylation. *Cell*, 24(1):33–40, 1981.

[143] L. E. Risner, A. Kuntimaddi, A. A. Lokken, N. J. Achille, N. W. Birch, K. Schoenfelt, J. H. Bushweller, and N. J. Zeleznik-Le. Functional specificity of CpG DNA-binding CXXC domains in mixed lineage leukemia. *J. Biol. Chem.*, 288(41):29901–29910, 2013.

[144] G. Rondelet, T. Dal Maso, L. Willems, and J. Wouters. Structural basis for recognition of histone H3K36me3 nucleosome by human de novo DNA methyltransferases 3A and 3B. *J. Struct. Biol.*, 194(3):357–367, 2016.

[145] T. Chen, N. Tsujimoto, and E. Li. The PWWP domain of Dnmt3a and Dnmt3b is required for directing DNA methylation to the major satellite repeats at pericentric heterochromatin. *Mol. Cell. Biol.*, 24(20):9048–9058, 2004.

[146] C. Jiang and B. F. Pugh. Nucleosome positioning and gene regulation: advances through genomics. *Nat. Rev. Genet.*, 10(3):161–172, 2009.

[147] C. K. Collings and J. N. Anderson. Links between DNA methylation and nucleosome occupancy in the human genome. *Epigenetics Chromatin*, 10(1):18, 2017.

[148] E. Segal, Y. Fondufe-Mittendorf, L. Chen, A. Thastrom, Y. Field, I. K. Moore, J. P. Wang, and J. Widom. A genomic code for nucleosome positioning. *Nature*, 442(7104):772–778, 2006.

[149] A. Barski, S. Cuddapah, K. Cui, T. Y. Roh, D. E. Schones, Z. Wang, G. Wei, I. Chepelev, and K. Zhao. High-resolution profiling of histone methylations in the human genome. *Cell*, 129(4):823–837, 2007.

[150] B. E. Bernstein, A. Meissner, and E. S. Lander. The mammalian epigenome. *Cell*, 128(4): 669–681, 2007.

[151] I. Albert, T. N. Mavrich, L. P. Tomsho, J. Qi, S. J. Zanton, S. C. Schuster, and B. F. Pugh. Translational and rotational settings of H2A.Z nucleosomes across the Saccharomyces cerevisiae genome. *Nature*, 446(7135):572–576, 2007.

[152] D. S. Johnson, A. Mortazavi, R. M. Myers, and B. Wold. Genome-wide mapping of in vivo protein-DNA interactions. *Science*, 316(5830):1497–1502, 2007.

[153] T. S. Mikkelsen, M. Ku, D. B. Jaffe, B. Issac, E. Lieberman, G. Giannoukos, P. Alvarez, W. Brockman, T. K. Kim, R. P. Koche, W. Lee, E. Mendenhall, A. O'Donovan, A. Presser, C. Russ, X. Xie, A. Meissner, M. Wernig, R. Jaenisch, C. Nusbaum, E. S. Lander, and B. E. Bernstein. Genome-wide maps of chromatin state in pluripotent and lineage-committed cells. *Nature*, 448(7153):553–560, 2007.

[154] J. Widom. Role of DNA sequence in nucleosome stability and dynamics. *Q. Rev. Biophys.*, 34(3):269–324, 2001.

[155] E. N. Trifonov. Sequence-dependent deformational anisotropy of chromatin DNA. *Nucleic Acids Res.*, 8(17):4041–4053, 1980.

[156] E. N. Trifonov and J. L. Sussman. The pitch of chromatin DNA is reflected in its nucleotide sequence. *Proc. Natl. Acad. Sci. U.S.A.*, 77(7):3816–3820, 1980.

[157] A. Portela, J. Liz, V. Nogales, F. Setien, A. Villanueva, and M. Esteller. DNA methylation determines nucleosome occupancy in the 5'-CpG islands of tumor suppressor genes. *Oncogene*, 32(47):5421–5428, 2013.

[158] R. K. Chodavarapu, S. Feng, Y. V. Bernatavichute, P. Y. Chen, H. Stroud, Y. Yu, J. A. Hetzel, F. Kuo, J. Kim, S. J. Cokus, D. Casero, M. Bernal, P. Huijser, A. T. Clark, U. Kramer, S. S. Merchant, X. Zhang, S. E. Jacobsen, and M. Pellegrini. Relationship between nucleosome positioning and DNA methylation. *Nature*, 466(7304):388–392, 2010.

[159] M. Xu, M. P. Kladde, J. L. Van Etten, and R. T. Simpson. Cloning, characterization and expression of the gene coding for a cytosine-5-DNA methyltransferase recognizing GpC. *Nucleic Acids Res.*, 26(17):3961–3966, 1998.

[160] M. Frommer, L. E. McDonald, D. S. Millar, C. M. Collis, F. Watt, G. W. Grigg, P. L. Molloy, and C. L. Paul. A genomic sequencing protocol that yields a positive display of 5-methylcytosine residues in individual DNA strands. *Proc. Natl. Acad. Sci. U.S.A.*, 89(5): 1827–1831, 1992.

[161] J. L. Bentley. Multidimensional binary search trees used for associative searching. *Commun. ACM*, 18(9):509–517, 1975.

[162] A. Bondi. van der Waals Volumes and Radii. *J. Phys. Chem.*, 68(3):441–451, 1964.

[163] D. Karolchik, A. S. Hinrichs, T. S. Furey, K. M. Roskin, C. W. Sugnet, D. Haussler, and W. J. Kent. The UCSC Table Browser data retrieval tool. *Nucleic Acids Res.*, 32(Database issue):D493–496, 2004.

[164] H. Mouilleron, V. Delcourt, and X. Roucou. Death of a dogma: eukaryotic mRNAs can code for more than one protein. *Nucleic Acids Res.*, 44(1):14–23, 2016.

[165] S. R. Starck, V. Jiang, M. Pavon-Eternod, S. Prasad, B. McCarthy, T. Pan, and N. Shastri. Leucine-tRNA initiates at CUG start codons for protein synthesis and presentation by MHC class I. *Science*, 336(6089):1719–1723, 2012.

[166] C. Fecher-Trost, U. Wissenbach, A. Beck, P. Schalkowsky, C. Stoerger, J. Doerr, A. Dembek, M. Simon-Thomas, A. Weber, P. Wollenberg, T. Ruppert, R. Middendorff, H. H. Maurer, and V. Flockerzi. The in vivo TRPV6 protein starts at a non-AUG triplet, decoded as methionine, upstream of canonical initiation at AUG. *J. Biol. Chem.*, 288(23):16629–16644, 2013.

[167] G. Menschaert, W. Van Criekinge, T. Notelaers, A. Koch, J. Crappe, K. Gevaert, and P. Van Damme. Deep proteome coverage based on ribosome profiling aids mass spectrometry-based protein and peptide discovery and provides evidence of alternative translation products and near-cognate translation initiation events. *Mol. Cell Proteomics*, 12(7):1780–1790, 2013.

[168] D. S. Peabody. Translation initiation at an ACG triplet in mammalian cells. *J. Biol. Chem.*, 262(24):11847–11851, 1987.

[169] N. T. Ingolia. Ribosome profiling: new views of translation, from single codons to genome scale. *Nat. Rev. Genet.*, 15:205–213, 2014.

[170] N. T. Ingolia, S. Ghaemmaghami, J. R. Newman, and J. S. Weissman. Genome-wide analysis in vivo of translation with nucleotide resolution using ribosome profiling. *Science*, 324:218–223, 2009.

[171] C. Barbosa, I. Peixeiro, and L. Romão. Gene expression regulation by upstream open reading frames and human disease. *PLoS Genet.*, 9(8):e1003529, 2013.

[172] W. Y. Chung, S. Wadhawan, R. Szklarczyk, S. K. Pond, and A. Nekrutenko. A first look at ARFome: dual-coding genes in mammalian genomes. *PLoS Comput. Biol.*, 3(5):e91, 2007.

[173] A. M. Michel, K. R. Choudhury, A. E. Firth, N. T. Ingolia, J. F. Atkins, and P. V. Baranov. Observation of dually decoded regions of the human genome using ribosome profiling data. *Genome Res.*, 22(11):2219–2229, 2012.

[174] C. Touriol, S. Bornes, S. Bonnal, S. Audigier, H. Prats, A. C. Prats, and S. Vagner. Generation of protein isoform diversity by alternative initiation of translation at non-AUG codons. *Biol. Cell*, 95:169–178, 2003.

[175] N. T. Ingolia, G. A. Brar, S. Rouskin, A. M. McGeachy, and J. S. Weissman. The ribosome profiling strategy for monitoring translation in vivo by deep sequencing of ribosome-protected mRNA fragments. *Nat. Protoc.*, 7(8):1534–1550, 2012.

[176] S. R. Hann, M. Dixit, R. C. Sears, and L. Sealy. The alternatively initiated c-Myc proteins differentially regulate transcription through a noncanonical DNA-binding site. *Genes Dev.*, 8:2441–2452, 1994.

[177] T. K. Blackwell, L. Kretzner, E. M. Blackwood, R. N. Eisenman, and H. Weintraub. Sequence-specific DNA binding by the c-Myc protein. *Science*, 250(4984):1149–1151, 1990.

[178] D. Reisman, N. B. Elkind, B. Roy, J. Beamon, and V. Rotter. c-Myc trans-activates the p53 promoter through a required downstream CACGTG motif. *Cell Growth Differ.*, 4(2): 57–65, 1993.

[179] G. A. Bazykin and A. V. Kochetov. Alternative translation start sites are conserved in eukaryotic genomes. *Nucleic Acids Res.*, 39(2):567–577, 2011.

[180] P. A. Sunderland, C. E. West, W. M. Waterworth, and C. M. Bray. Choice of a start codon in a single transcript determines dna ligase 1 isoform production and intracellular targeting in arabidopsis thaliana. *Biochemical Society Transactions*, 32(4):614–616, 2004.

[181] S. Vagner, C. Touriol, B. Galy, S. Audigier, M. C. Gensac, F. Amalric, F. Bayard, H. Prats, and A. C. Prats. Translation of CUG- but not AUG-initiated forms of human fibroblast growth factor 2 is activated in transformed and stressed cells. *J. Cell Biol.*, 135:1391–1402, 1996.

[182] M. Kozak. Compilation and analysis of sequences upstream from the translational start site in eukaryotic mRNAs. *Nucleic Acids Res.*, 12:857–872, 1984.

[183] M. Kozak. Point mutations define a sequence flanking the AUG initiator codon that modulates translation by eukaryotic ribosomes. *Cell*, 44:283–292, 1986.

[184] M. Kozak. At least six nucleotides preceding the AUG initiator codon enhance translation in mammalian cells. *J. Mol. Biol.*, 196:947–950, 1987.

[185] M. Kozak. An analysis of 5'-noncoding sequences from 699 vertebrate messenger RNAs. *Nucleic Acids Res.*, 15:8125–8148, 1987.

[186] M. Kozak. Influences of mRNA secondary structure on initiation by eukaryotic ribosomes. *Proc. Natl. Acad. Sci. U.S.A.*, 83(9):2850–2854, 1986.

[187] M. Kozak. Circumstances and mechanisms of inhibition of translation by secondary structure in eucaryotic mRNAs. *Mol. Cell. Biol.*, 9:5134–5142, 1989.

[188] M. Kozak. Downstream secondary structure facilitates recognition of initiator codons by eukaryotic ribosomes. *Proc. Natl. Acad. Sci. U.S.A.*, 87:8301–8305, 1990.

[189] W. L. Noderer, R. J. Flockhart, A. Bhaduri, A. J. Diaz de Arce, J. Zhang, P. A. Khavari, and C. L. Wang. Quantitative analysis of mammalian translation initiation sites by FACS-seq. *Mol. Syst. Biol.*, 10:748, 2014.

[190] M. Kozak. Structural features in eukaryotic mRNAs that modulate the initiation of translation. *J. Biol. Chem.*, 266:19867–19870, 1991.

[191] G. A. Brar and J. S. Weissman. Ribosome profiling reveals the what, when, where and how of protein synthesis. *Nat. Rev. Mol. Cell Biol.*, 16(11):651–664, 2015.

[192] M. Fresno, A. Jimenez, and D. Vazquez. Inhibition of translation in eukaryotic systems by harringtonine. *Eur. J. Biochem.*, 72(2):323–330, 1977.

[193] D. R. Bentley, S. Balasubramanian, H. P. Swerdlow, G. P. Smith, J. Milton, C. G. Brown, K. P. Hall, D. J. Evers, C. L. Barnes, H. R. Bignell, et al. Accurate whole human genome sequencing using reversible terminator chemistry. *Nature*, 456(7218):53–59, 2008.

[194] B. Schölkopf. *Support Vector Learning*. Dissertation, Technische Universität Berlin, 1997.

[195] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics. Springer, Second edition, 2009.

[196] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[197] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2009.

[198] S. Boucheron, O. Bousquet, and G. Lugosi. Theory of Classification: A Survey of Some Recent Advances. *ESAIM: Probability and Statistics*, 9:323–375, 2005.

[199] A. J. Smola and B. Schölkopf. A tutorial on support vector regression. *Stat. Comput.*, 14 (3):199–222, 2004.

[200] C. C. Chang and C. J. Lin. LIBSVM: A library for support vector machines. *ACM TIST*, 2:27:1–27:27, 2011.

[201] G. Zararsiz, F. Elmali, and A. Ozturk. Bagging support vector machines for leukemia classification. 2012.

[202] D. M. W. Powers. Evaluation: from precision, recall and f-measure to ROC, informedness, markedness and correlation. *J. Mach. Learn. Tech.*, 2(1):37–63, 2011.

[203] C. Marrocco, R. P. W. Duin, and F. Tortorella. Maximizing the area under the ROC curve by pairwise feature combination. *Pattern Recognit.*, 41(6):1961–1974, 2008.

[204] T. Berners-Lee. Information Management: A Proposal. CERN, 1989. URL `https://www.w3.org/History/1989/proposal.html`. Accessed on 01.09.2017.

[205] World Wide Web Consortium (W3C), 1994. URL `https://www.w3.org/`. Accessed on 01.09.2017.

[206] TEMPLATED developers. TEMPLATED: Free CSS, HTML5 and Responsive Site Templates, 2014. URL `https://templated.co/`. Accessed on 01.09.2017.

[207] Creative Commons, 2001. URL `https://creativecommons.org/`. Accessed on 01.09.2017.

[208] W3Schools developers. W3Schools: The World's Largest Web Developer Site, 1998. URL `https://www.w3schools.com/`. Accessed on 01.09.2017.

[209] Highsoft AS. Highcharts: Interactive JavaScript charts for your webpage, 2009. URL `https://www.highcharts.com/`. Accessed on 01.09.2017.

[210] Ecma International. Standard ECMA-262 8th Edition: ECMAScript 2017 Language Specification, 2017. URL `http://www.json.org/`. Accessed on 01.09.2017.

[211] A. Yates, K. Beal, S. Keenan, W. McLaren, M. Pignatelli, G. R. Ritchie, M. Ruffier, K. Taylor, A. Vullo, and P. Flicek. The Ensembl REST API: Ensembl Data for Any Language. *Bioinformatics*, 31(1):143–145, 2015.

[212] V. Trevino, F. Falciani, and H. A. Barrera-Saldana. DNA microarrays: a powerful genomic tool for biomedical and clinical research. *Mol. Med.*, 13(9-10):527–541, 2007.

[213] N. Saitou and M. Nei. The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Mol. Biol. Evol.*, 4(4):406–425, 1987.

[214] R. R. Sokal and C. D. Michener. A statistical method for evaluating systematic relationships. *Univ. Kans. Sci. Bull.*, 28:1409–1438, 1958.

[215] A. G. Hatzigeorgiou. Translation initiation start prediction in human cDNAs with high accuracy. *Bioinformatics*, 18:343–350, 2002.

[216] Y. Saeys, T. Abeel, S. Degroeve, and Y. Van de Peer. Translation initiation site prediction on a genomic scale: beauty in simplicity. *Bioinformatics*, 23:i418–423, 2007.

[217] M. E. Sparks and V. Brendel. MetWAMer: eukaryotic translation initiation site prediction. *BMC Bioinformatics*, 9:381, 2008.

[218] W. Chen, P. M. Feng, E. Z. Deng, H. Lin, and K. C. Chou. iTIS-PseTNC: a sequence-based predictor for identifying translation initiation site in human genes using pseudo trinucleotide composition. *Anal. Biochem.*, 462:76–83, 2014.

[219] T. Tatusov and R. Tatusov. ORF Finder. URL `http://diyhpl.us/~bryan/irc/protocol-online/protocol-cache/gorf.html`. Accessed on 14.01.2015.

[220] X. J. Min, G. Butler, R. Storms, and A. Tsang. OrfPredictor: predicting protein-coding regions in EST-derived sequences. *Nucleic Acids Res.*, 33:W677–680, 2005.

[221] J. Crappé, W. Van Criekinge, G. Trooskens, E. Hayakawa, W. Luyten, G. Baggerman, and G. Menschaert. Combining in silico prediction and ribosome profiling in a genome-wide search for novel putatively coding sORFs. *BMC Genomics*, 14:648, 2013.

[222] A. M. Michel, D. E. Andreev, and P. V. Baranov. Computational approach for calculating the probability of eukaryotic translation initiation from ribo-seq data that takes into account leaky scanning. *BMC Bioinformatics*, 15:380, 2014.

[223] L. Zhang, S. Kasif, C. R. Cantor, and N. E. Broude. GC/AT-content spikes as genomic punctuation marks. *Proc. Natl. Acad. Sci. U.S.A.*, 101:16855–16860, 2004.

[224] I. R. Gould and P. A. Kollman. Theoretical Investigation of the Hydrogen Bond Strengths in Guanine-Cytosine and Adenine-Thymine Base Pairs. *J. Am. Chem. Soc.*, 116:2493–2499, 1994.

[225] H. Liu and L. Wong. Data mining tools for biological sequences. *J. Bioinform. Comput. Biol.*, 1:139–167, 2003.

[226] K. Reuter, A. Biehl, L. Koch, and V. Helms. PreTIS: A Tool to Predict Non-canonical 5′ UTR Translational Initiation Sites in Human and Mouse. *PLoS Comput. Biol.*, 12(10): e1005170, 2016.

[227] The UniProt Consortium. UniProt: a hub for protein information. *Nucleic Acids Res.*, 43: D204–212, 2015.

[228] R. Boeck and D. Kolakofsky. Positions +5 and +6 can be major determinants of the efficiency of non-AUG initiation codons for protein synthesis. *EMBO J.*, 13(15):3608–3617, 1994.

[229] A. Sandelin, P. Carninci, B. Lenhard, J. Ponjavic, Y. Hayashizaki, and D. A. Hume. Mammalian RNA polymerase II core promoters: insights from genome-wide studies. *Nat. Rev. Genet.*, 8(6):424–436, 2007.

[230] G. A. Patikoglou, J. L. Kim, L. Sun, S. H. Yang, T. Kodadek, and S. K. Burley. TATA element recognition by the TATA box-binding protein has been conserved throughout evolution. *Genes Dev.*, 13(24):3217–3230, 1999.

[231] P. Kolovos, T. A. Knoch, F. G. Grosveld, P. R. Cook, and A. Papantonis. Enhancers and silencers: an integrated and simple model for their function. *Epigenetics Chromatin*, 5(1): 1, 2012.

[232] International Human Genome Sequencing Consortium. Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860–921, 2001.

[233] F. S. Collins, E. D. Green, A. E. Guttmacher, and M. S. Guyer. A vision for the future of genomics research. *Nature*, 422(6934):835–847, 2003.

[234] International Human Genome Sequencing Consortium. Finishing the euchromatic sequence of the human genome. *Nature*, 431(7011):931–945, 2004.

[235] J. C. Venter, M. D. Adams, E. W. Myers, P. W. Li, R. J. Mural, G. G. Sutton, H. O. Smith, M. Yandell, C. A. Evans, R. A. Holt, et al. The sequence of the human genome. *Science*, 291(5507):1304–1351, 2001.

[236] S. Levy, G. Sutton, P. C. Ng, L. Feuk, A. L. Halpern, B. P. Walenz, N. Axelrod, J. Huang, E. F. Kirkness, G. Denisov, Y. Lin, J. R. MacDonald, A. W. Pang, M. Shago, T. B. Stockwell, A. Tsiamouri, V. Bafna, V. Bansal, S. A. Kravitz, D. A. Busam, K. Y. Beeson, T. C. McIntosh, K. A. Remington, J. F. Abril, J. Gill, J. Borman, Y. H. Rogers, M. E. Frazier, S. W. Scherer, R. L. Strausberg, and J. C. Venter. The diploid genome sequence of an individual human. *PLoS Biol.*, 5(10):e254, 2007.

[237] L. B. Barreiro, G. Laval, H. Quach, E. Patin, and L. Quintana-Murci. Natural selection has driven population differentiation in modern humans. *Nat. Genet.*, 40(3):340–345, 2008.

[238] R. Karki, D. Pandya, R. C. Elston, and C. Ferlini. Defining "mutation" and "polymorphism" in the era of personal genomics. *BMC Med. Genomics*, 8:37, 2015.

[239] H. Echols and M. F. Goodman. Fidelity Mechanisms in DNA Replication. *Annu. Rev. Biochem.*, 60:477–511, 1991.

[240] M. S. Cooke, M. D. Evans, M. Dizdaroglu, and J. Lunec. Oxidative DNA damage: mechanisms, mutation, and disease. *FASEB J.*, 17(10):1195–1214, 2003.

[241] D. Branzei and M. Foiani. Regulation of DNA repair throughout the cell cycle. *Nat. Rev. Mol. Cell Biol.*, 9(4):297–308, 2008.

[242] J. F. Crow. Spontaneous mutation as a risk factor. *Exp. Clin. Immunogenet.*, 12(3):121–128, 1995.

[243] A. S. Kondrashov. Contamination of the genome by very slightly deleterious mutations: why have we not died 100 times over? *J. Theor. Biol.*, 175(4):583–594, 1995.

[244] S. Besenbacher, S. Liu, J. M. Izarzugaza, J. Grove, K. Belling, J. Bork-Jensen, S. Huang, T. D. Als, S. Li, R. Yadav, A. Rubio-Garcia, F. Lescai, D. Demontis, J. Rao, W. Ye, T. Mailund, R. M. Friborg, C. N. Pedersen, R. Xu, J. Sun, H. Liu, O. Wang, X. Cheng, D. Flores, E. Rydza, K. Rapacki, J. Damm Sørensen, P. Chmura, D. Westergaard, P. Dworzynski, T. I. Sørensen, O. Lund, T. Hansen, X. Xu, N. Li, L. Bolund, O. Pedersen, H. Eiberg, A. Krogh, A. D. Børglum, S. Brunak, K. Kristiansen, M. H. Schierup, J. Wang, R. Gupta, P. Villesen, and S. Rasmussen. Novel variation and de novo mutation rates in population-wide de novo assembled Danish trios. *Nat. Commun.*, 6:5969, 2015.

[245] R. Acuna-Hidalgo, J. A. Veltman, and A. Hoischen. New insights into the generation and role of de novo mutations in health and disease. *Genome Biol.*, 17(1):241, 2016.

[246] A. J. Brookes. The essence of SNPs. *Gene*, 234(2):177–186, 1999.

[247] M. Malkki and E. W. Petersdorf. Genotyping of single nucleotide polymorphisms by 5′ nuclease allelic discrimination. *Methods Mol. Biol.*, 882:173–182, 2012.

[248] D. G. Wang, J. B. Fan, C. J. Siao, A. Berno, P. Young, R. Sapolsky, G. Ghandour, N. Perkins, E. Winchester, J. Spencer, L. Kruglyak, L. Stein, L. Hsie, T. Topaloglou, E. Hubbell, E. Robinson, M. Mittmann, M. S. Morris, N. Shen, D. Kilburn, J. Rioux, C. Nusbaum, S. Rozen, T. J. Hudson, R. Lipshutz, M. Chee, and E. S. Lander. Large-scale identification, mapping, and genotyping of single-nucleotide polymorphisms in the human genome. *Science*, 280(5366):1077–1082, 1998.

[249] E. Dawson, Y. Chen, S. Hunt, L. J. Smink, A. Hunt, K. Rice, S. Livingston, S. Bumpstead, R. Bruskiewich, P. Sham, R. Ganske, M. Adams, K. Kawasaki, N. Shimizu, S. Minoshima, B. Roe, D. Bentley, and I. Dunham. A SNP resource for human chromosome 22: extracting dense clusters of SNPs from the genomic sequence. *Genome Res.*, 11(1):170–178, 2001.

[250] A. Telenti, L. C. Pierce, W. H. Biggs, J. di Iulio, E. H. Wong, M. M. Fabani, E. F. Kirkness, A. Moustafa, N. Shah, C. Xie, S. C. Brewerton, N. Bulsara, C. Garner, G. Metzker, E. Sandoval, B. A. Perkins, F. J. Och, Y. Turpaz, and J. C. Venter. Deep sequencing of 10,000 human genomes. *Proc. Natl. Acad. Sci. U.S.A.*, 113(42):11901–11906, 2016.

[251] R. Sachidanandam, D. Weissman, S. C. Schmidt, J. M. Kakol, L. D. Stein, G. Marth, S. Sherry, J. C. Mullikin, B. J. Mortimore, D. L. Willey, et al. A map of human genome sequence variation containing 1.42 million single nucleotide polymorphisms. *Nature*, 409 (6822):928–933, 2001.

[252] J. Sebat, B. Lakshmi, J. Troge, J. Alexander, J. Young, P. Lundin, S. Maner, H. Massa, M. Walker, M. Chi, N. Navin, R. Lucito, J. Healy, J. Hicks, K. Ye, A. Reiner, T. C. Gilliam, B. Trask, N. Patterson, A. Zetterberg, and M. Wigler. Large-scale copy number polymorphism in the human genome. *Science*, 305(5683):525–528, 2004.

[253] D. R. Schrider and M. W. Hahn. Gene copy-number polymorphism in nature. *Proc. Biol. Sci.*, 277(1698):3213–3221, 2010.

[254] A. J. Iafrate, L. Feuk, M. N. Rivera, M. L. Listewnik, P. K. Donahoe, Y. Qi, S. W. Scherer, and C. Lee. Detection of large-scale variation in the human genome. *Nat. Genet.*, 36(9): 949–951, 2004.

[255] D. P. Locke, R. Segraves, L. Carbone, N. Archidiacono, D. G. Albertson, D. Pinkel, and E. E. Eichler. Large-scale variation among human and great ape genomes determined by array comparative genomic hybridization. *Genome Res.*, 13(3):347–357, 2003.

[256] S. A. McCarroll and D. M. Altshuler. Copy-number variation and association studies of human disease. *Nat. Genet.*, 39(7 Suppl):37–42, 2007.

[257] E. Gonzalez, H. Kulkarni, H. Bolivar, A. Mangano, R. Sanchez, G. Catano, R. J. Nibbs, B. I. Freedman, M. P. Quinones, M. J. Bamshad, K. K. Murthy, B. H. Rovin, W. Bradley, R. A. Clark, S. A. Anderson, R. J. O'connell, B. K. Agan, S. S. Ahuja, R. Bologna, L. Sen, M. J. Dolan, and S. K. Ahuja. The influence of CCL3L1 gene-containing segmental duplications on HIV-1/AIDS susceptibility. *Science*, 307(5714):1434–1440, 2005.

[258] Q. S. Padiath, K. Saigoh, R. Schiffmann, H. Asahara, T. Yamada, A. Koeppen, K. Hogan, L. J. Ptacek, and Y. H. Fu. Lamin B1 duplications cause autosomal dominant leukodystrophy. *Nat. Genet.*, 38(10):1114–1123, 2006.

[259] J. R. Lupski. Genomic rearrangements and sporadic disease. *Nat. Genet.*, 39(7 Suppl): S43–47, 2007.

[260] M. A. Mosrati, A. Malmstrom, M. Lysiak, A. Krysztofiak, M. Hallbeck, P. Milos, A. L. Hallbeck, C. Bratthall, M. Strandeus, M. Stenmark-Askmalm, and P. Soderkvist. TERT promoter mutations and polymorphisms as prognostic factors in primary glioblastoma. *Oncotarget*, 6(18):16663–16673, 2015.

[261] S. Horn, A. Figl, P. S. Rachakonda, C. Fischer, A. Sucker, A. Gast, S. Kadel, I. Moll, E. Nagore, K. Hemminki, D. Schadendorf, and R. Kumar. TERT promoter mutations in familial and sporadic melanoma. *Science*, 339(6122):959–961, 2013.

[262] J. Vinagre, A. Almeida, H. Populo, R. Batista, J. Lyra, V. Pinto, R. Coelho, R. Celestino, H. Prazeres, L. Lima, M. Melo, A. G. da Rocha, A. Preto, P. Castro, L. Castro, F. Pardal, J. M. Lopes, L. L. Santos, R. M. Reis, J. Cameselle-Teijeiro, M. Sobrinho-Simoes, J. Lima, V. Maximo, and P. Soares. Frequency of TERT promoter mutations in human cancers. *Nat. Commun.*, 4:2185, 2013.

[263] L. Gotoh, K. Inoue, G. Helman, S. Mora, K. Maski, J. S. Soul, M. Bloom, S. H. Evans, Y. Goto, L. Caldovic, G. M. Hobson, and A. Vanderver. GJC2 promoter mutations causing Pelizaeus-Merzbacher-like disease. *Mol. Genet. Metab.*, 111(3):393–398, 2014.

[264] A. Chakravarti. To a future of genetic medicine. *Nature*, 409(6822):822–823, 2001.

[265] B. P. O'Sullivan and S. D. Freedman. Cystic fibrosis. *Lancet*, 373(9678):1891–1904, 2009.

[266] S. M. Laws, E. Hone, S. Gandy, and R. N. Martins. Expanding the association between the APOE gene and the risk of Alzheimer's disease: possible roles for APOE promoter polymorphisms and alterations in APOE transcription. *J. Neurochem.*, 84(6):1215–1236, 2003.

[267] L. Lopalco. CCR5: From Natural Resistance to a New Anti-HIV Strategy. *Viruses*, 2(2): 574–600, 2010.

[268] F. B. Piel, A. P. Patil, R. E. Howes, O. A. Nyangiri, P. W. Gething, T. N. Williams, D. J. Weatherall, and S. I. Hay. Global distribution of the sickle cell gene and geographical confirmation of the malaria hypothesis. *Nat. Commun.*, 1:104, 2010.

[269] K. Jarrett, M. Williams, S. Horn, D. Radford, and J. M. Wyss. "Sickle cell anemia: tracking down a mutation": an interactive learning laboratory that communicates basic principles of genetics and cellular biology. *Adv. in Physiol. Educ.*, 40(1):110–115, 2016.

[270] C. Gonzaga-Jauregui, J. R. Lupski, and R. A. Gibbs. Human genome sequencing in health and disease. *Annu. Rev. Med.*, 63:35–61, 2012.

[271] L. Roewer. DNA fingerprinting in forensics: past, present, future. *Investig. Genet.*, 4(1): 22, 2013.

[272] N. J. Schork. Personalized medicine: Time for one-person trials. *Nature*, 520(7549):609–611, 2015.

[273] B. J. Druker, C. L. Sawyers, H. Kantarjian, D. J. Resta, S. F. Reese, J. M. Ford, R. Capdeville, and M. Talpaz. Activity of a specific inhibitor of the BCR-ABL tyrosine kinase in the blast crisis of chronic myeloid leukemia and acute lymphoblastic leukemia with the Philadelphia chromosome. *N. Engl. J. Med.*, 344(14):1038–1042, 2001.

[274] C. S. Karapetis, S. Khambata-Ford, D. J. Jonker, C. J. O'Callaghan, D. Tu, N. C. Tebbutt, R. J. Simes, H. Chalchal, J. D. Shapiro, S. Robitaille, T. J. Price, L. Shepherd, H. J. Au, C. Langer, M. J. Moore, and J. R. Zalcberg. K-ras mutations and benefit from cetuximab in advanced colorectal cancer. *N. Engl. J. Med.*, 359(17):1757–1765, 2008.

[275] I. V. Tuxen, L. Jønson, E. Santoni-Rugiu, J. P. Hasselby, F. C. Nielsen, and U. Lassen. Personalized oncology: genomic screening in phase 1. *APMIS*, 122(8):723–733, 2014.

[276] W. Y. Wang, B. J. Barratt, D. G. Clayton, and J. A. Todd. Genome-wide association studies: theoretical and practical concerns. *Nat. Rev. Genet.*, 6(2):109–118, 2005.

[277] M. I. McCarthy, G. R. Abecasis, L. R. Cardon, D. B. Goldstein, J. Little, J. P. Ioannidis, and J. N. Hirschhorn. Genome-wide association studies for complex traits: consensus, uncertainty and challenges. *Nat. Rev. Genet.*, 9(5):356–369, 2008.

[278] N. R. Wray, J. Yang, B. J. Hayes, A. L. Price, M. E. Goddard, and P. M. Visscher. Pitfalls of predicting complex traits from SNPs. *Nat. Rev. Genet.*, 14(7):507–515, 2013.

[279] J. Novembre, T. Johnson, K. Bryc, Z. Kutalik, A. R. Boyko, A. Auton, A. Indap, K. S. King, S. Bergmann, M. R. Nelson, M. Stephens, and C. D. Bustamante. Genes mirror geography within Europe. *Nature*, 456(7218):98–101, 2008.

[280] M. Nei and W. H. Li. Mathematical model for studying genetic variation in terms of restriction endonucleases. *Proc. Natl. Acad. Sci. U.S.A.*, 76(10):5269–5273, 1979.

[281] F. Tajima. Statistical method for testing the neutral mutation hypothesis by DNA polymorphism. *Genetics*, 123(3):585–595, 1989.

[282] M. Kimura. *The Neutral Theory of Molecular Evolution*. Cambridge University Press, New York, 1983.

[283] Y. X. Fu. Statistical properties of segregating sites. *Theor. Popul. Biol.*, 48(2):172–197, 1995.

[284] D. L. Hartl and A. G. Clark. *Principles of Population Genetics*. Sinauer Associates, Inc.; 4th edition, Massachusetts, 2006.

[285] M. Cargill, D. Altshuler, J. Ireland, P. Sklar, K. Ardlie, N. Patil, N. Shaw, C. R. Lane, E. P. Lim, N. Kalyanaraman, J. Nemesh, L. Ziaugra, L. Friedland, A. Rolfe, J. Warrington, R. Lipshutz, G. Q. Daley, and E. S. Lander. Characterization of single-nucleotide polymorphisms in coding regions of human genes. *Nat. Genet.*, 22(3):231–238, 1999.

[286] J. A. Tennessen, A. W. Bigham, T. D. O'Connor, W. Fu, E. E. Kenny, S. Gravel, S. McGee, R. Do, X. Liu, G. Jun, H. M. Kang, D. Jordan, S. M. Leal, S. Gabriel, M. J. Rieder, G. Abecasis, D. Altshuler, D. A. Nickerson, E. Boerwinkle, S. Sunyaev, C. D. Bustamante, M. J. Bamshad, and J. M. Akey. Evolution and functional impact of rare coding variation from deep sequencing of human exomes. *Science*, 337(6090):64–69, 2012.

[287] M. K. Leabman, C. C. Huang, J. DeYoung, E. J. Carlson, T. R. Taylor, M. de la Cruz, S. J. Johns, D. Stryke, M. Kawamoto, T. J. Urban, D. L. Kroetz, T. E. Ferrin, A. G. Clark, N. Risch, I. Herskowitz, and K. M. Giacomini. Natural variation in human membrane transporter genes reveals evolutionary and functional constraints. *Proc. Natl. Acad. Sci. U.S.A.*, 100(10):5896–5901, 2003.

[288] N. A. O'Leary, M. W. Wright, J. R. Brister, S. Ciufo, D. Haddad, R. McVeigh, B. Rajput, B. Robbertse, B. Smith-White, D. Ako-Adjei, et al. Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation. *Nucleic Acids Res.*, 44(D1):D733–745, 2016.

[289] W. P. Kloosterman, L. C. Francioli, F. Hormozdiari, T. Marschall, J. Y. Hehir-Kwa, A. Abdellaoui, E. W. Lameijer, M. H. Moed, V. Koval, I. Renkens, et al. Characteristics of de novo structural changes in the human genome. *Genome Res.*, 25(6):792–801, 2015.

[290] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet.*, 25(1):25–29, 2000.

[291] The Gene Ontology Consortium. Expansion of the Gene Ontology knowledgebase and resources. *Nucleic Acids Res.*, 45(D1):D331–D338, 2017.

[292] W. Huang da, B. T. Sherman, and R. A. Lempicki. Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. *Nat. Protoc.*, 4(1):44–57, 2009.

[293] R. Nielsen, M. J. Hubisz, I. Hellmann, D. Torgerson, A. M. Andres, A. Albrechtsen, R. Gutenkunst, M. D. Adams, M. Cargill, A. Boyko, A. Indap, C. D. Bustamante, and A. G. Clark. Darwinian and demographic forces affecting human protein coding genes. *Genome Res.*, 19(5):838–849, 2009.

[294] S. A. Shabalina, A. Y. Ogurtsov, I. B. Rogozin, E. V. Koonin, and D. J. Lipman. Comparative analysis of orthologous eukaryotic mRNAs: potential hidden functional signals. *Nucleic Acids Res.*, 32(5):1774–1782, 2004.

[295] S. Levy, S. Hannenhalli, and C. Workman. Enrichment of regulatory signals in conserved non-coding genomic sequence. *Bioinformatics*, 17(10):871–877, 2001.

[296] J. C. Stephens, J. A. Schneider, D. A. Tanguay, J. Choi, T. Acharya, S. E. Stanley, R. Jiang, C. J. Messer, A. Chew, J. H. Han, J. Duan, J. L. Carr, M. S. Lee, B. Koshy, A. M. Kumar, G. Zhang, W. R. Newell, A. Windemuth, C. Xu, T. S. Kalbfleisch, S. L. Shaner, K. Arnold, V. Schulz, C. M. Drysdale, K. Nandabalan, R. S. Judson, G. Ruano, and G. F. Vovis. Haplotype variation and linkage disequilibrium in 313 human genes. *Science*, 293(5529):489–493, 2001.

[297] B. A. Salisbury, M. Pungliya, J. Y. Choi, R. Jiang, X. J. Sun, and J. C. Stephens. SNP and haplotype variation in the human genome. *Mutat. Res.*, 526(1-2):53–61, 2003.

[298] M. Puig, D. Castellano, L. Pantano, C. Giner-Delgado, D. Izquierdo, M. Gaya-Vidal, J. I. Lucas-Lledo, T. Esko, C. Terao, F. Matsuda, and M. Caceres. Functional Impact and Evolution of a Novel Human Polymorphic Inversion That Disrupts a Gene and Creates a Fusion Transcript. *PLoS Genet.*, 11(10):e1005495, 2015.

[299] Y. He, K. Maier, J. Leppert, I. Hausser, A. Schwieger-Briel, L. Weibel, M. Theiler, D. Kiritsi, H. Busch, M. Boerries, K. Hannula-Jouppi, H. Heikkila, K. Tasanen, D. Castiglia, G. Zambruno, and C. Has. Monoallelic Mutations in the Translation Initiation Codon of KLHL24 Cause Skin Fragility. *Am. J. Hum. Genet.*, 99(6):1395–1404, 2016.

[300] C. Melton, J. A. Reuter, D. V. Spacek, and M. Snyder. Recurrent somatic mutations in regulatory regions of human cancer genomes. *Nat. Genet.*, 47(7):710–716, 2015.

[301] M. S. Taylor, C. Kai, J. Kawai, P. Carninci, Y. Hayashizaki, and C. A. Semple. Heterotachy in mammalian promoter evolution. *PLoS Genet.*, 2(4):e30, 2006.

[302] T. G. Clark, T. Andrew, G. M. Cooper, E. H. Margulies, J. C. Mullikin, and D. J. Balding. Functional constraint and small insertions and deletions in the ENCODE regions of the human genome. *Genome Biol.*, 8(9):R180, 2007.

[303] K. Higasa and K. Hayashi. Periodicity of SNP distribution around transcription start sites. *BMC Genomics*, 7:66, 2006.

[304] R. Sabarinathan, L. Mularoni, J. Deu-Pons, A. Gonzalez-Perez, and N. Lopez-Bigas. Nucleotide excision repair is impaired by binding of transcription factors to DNA. *Nature*, 532(7598):264–267, 2016.

[305] P. Carninci, A. Sandelin, B. Lenhard, S. Katayama, K. Shimokawa, J. Ponjavic, C. A. Semple, M. S. Taylor, P. G. Engstrom, M. C. Frith, A. R. Forrest, W. B. Alkema, S. L. Tan, C. Plessy, R. Kodzius, T. Ravasi, T. Kasukawa, S. Fukuda, M. Kanamori-Katayama, Y. Kitazume, H. Kawaji, C. Kai, M. Nakamura, H. Konno, K. Nakano, S. Mottagui-Tabar, P. Arner, A. Chesi, S. Gustincich, F. Persichetti, H. Suzuki, S. M. Grimmond, C. A. Wells, V. Orlando, C. Wahlestedt, E. T. Liu, M. Harbers, J. Kawai, V. B. Bajic, D. A. Hume, and Y. Hayashizaki. Genome-wide analysis of mammalian promoter architecture and evolution. *Nat. Genet.*, 38(6):626–635, 2006.

[306] M. C. Frith, E. Valen, A. Krogh, Y. Hayashizaki, P. Carninci, and A. Sandelin. A code for transcription initiation in mammalian genomes. *Genome Res.*, 18(1):1–12, 2008.

[307] C. F. Mugal, P. F. Arndt, L. Holm, and H. Ellegren. Evolutionary consequences of DNA methylation on the GC content in vertebrate genomes. *G3 (Bethesda)*, 5(3):441–447, 2015.

[308] E. J. Clowney, A. Magklara, B. M. Colquitt, N. Pathak, R. P. Lane, and S. Lomvardas. High-throughput mapping of the promoters of the mouse olfactory receptor genes reveals a new type of mammalian promoter and provides insight into olfactory receptor gene regulation. *Genome Res.*, 21(8):1249–1259, 2011.

[309] S. Goodwin, J. D. McPherson, and W. R. McCombie. Coming of age: ten years of next-generation sequencing technologies. *Nat. Rev. Genet.*, 17(6):333–351, 2016.

[310] E. C. Hayden. Technology: The $1,000 genome. *Nature*, 507(7492):294–295, 2014.

[311] Illumina. The $1,000 Genome, 2015. URL `https://www.illumina.com/content/dam/illumina-marketing/documents/company/featured-articles/the-1000-dollar-genome.pdf`. Accessed on 28.09.2017.

[312] S. T. Bennett, C. Barnes, A. Cox, L. Davies, and C. Brown. Toward the 1,000 dollars human genome. *Pharmacogenomics*, 6(4):373–382, 2005.

[313] K. A. Phillips, M. J. Pletcher, and U. Ladabaum. Is the "$1000 Genome" really $1000? Understanding the full benefits and costs of genomic sequencing. *Technol. Health Care*, 23(3):373–379, 2015.

[314] S. Behjati and P. S. Tarpey. What is next generation sequencing? *Arch. Dis. Child. Educ. Pract. Ed.*, 98(6):236–238, 2013.

[315] Illumina. History of sequencing by synthesis, 2017. URL `https://www.illumina.com/science/technology/next-generation-sequencing/illumina-sequencing-history.html`. Accessed on 30.08.2017.

[316] S. Bennett. Solexa Ltd. *Pharmacogenomics*, 5(4):433–438, 2004.

[317] Illumina. Sequencing and array-based solutions for genetic research, 1998. URL `https://www.illumina.com/`. Accessed on 30.08.2017.

[318] Illumina. HiSeq X, 2017. URL `https://www.illumina.com/systems/sequencing-platforms/hiseq-x.html`. Accessed on 30.08.2017.

[319] Illumina. Advantages of paired-end and single-read sequencing, 2017. URL `https://www.illumina.com/science/technology/next-generation-sequencing/paired-end-vs-single-read-sequencing.html`. Accessed on 30.08.2017.

[320] B. P. Hodkinson and E. A. Grice. Next-Generation Sequencing: A Review of Technologies and Tools for Wound Microbiome Research. *Adv. Wound Care (New Rochelle)*, 4(1):50–58, 2015.

[321] M. J. Chaisson, J. Huddleston, M. Y. Dennis, P. H. Sudmant, M. Malig, F. Hormozdiari, F. Antonacci, U. Surti, R. Sandstrom, M. Boitano, J. M. Landolin, J. A. Stamatoyannopoulos, M. W. Hunkapiller, J. Korlach, and E. E. Eichler. Resolving the complexity of the human genome using single-molecule sequencing. *Nature*, 517(7536):608–611, 2015.

[322] P. J. Cock, C. J. Fields, N. Goto, M. L. Heuer, and P. M. Rice. The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants. *Nucleic Acids Res.*, 38(6):1767–1771, 2010.

[323] B. Ewing and P. Green. Base-calling of automated sequencer traces using phred. II. Error probabilities. *Genome Res.*, 8(3):186–194, 1998.

[324] S. Andrews. FastQC - A quality control tool for high throughput sequence data, 2010. URL `https://www.bioinformatics.babraham.ac.uk/projects/fastqc/`. Accessed on 29.08.2017.

[325] M. Burrows and D. J. Wheeler. A block-sorting lossless data compression algorithm. 1994.

[326] P. Ferragina and G. Manzini. Opportunistic data structures with applications. In *Proceedings 41st Annual Symposium on Foundations of Computer Science*, pages 390–398, 2000.

[327] D. C. Koboldt, K. Chen, T. Wylie, D. E. Larson, M. D. McLellan, E. R. Mardis, G. M. Weinstock, R. K. Wilson, and L. Ding. VarScan: variant detection in massively parallel sequencing of individual and pooled samples. *Bioinformatics*, 25(17):2283–2285, 2009.

[328] The Cancer Genome Atlas Research Network. Integrated genomic analyses of ovarian carcinoma. *Nature*, 474(7353):609–615, 2011.

[329] ENCODE Project Consortium. An integrated encyclopedia of DNA elements in the human genome. *Nature*, 489(7414):57–74, 2012.

[330] D. Thompson, A. Regev, and S. Roy. Comparative analysis of gene regulatory networks: from network reconstruction to evolution. *Annu. Rev. Cell Dev. Biol.*, 31:399–428, 2015.

[331] F. Jacob and J. Monod. Genetic regulatory mechanisms in the synthesis of proteins. *J. Mol. Biol.*, 3:318–356, 1961.

[332] P. J. Wittkopp and G. Kalay. Cis-regulatory elements: molecular mechanisms and evolutionary processes underlying divergence. *Nat. Rev. Genet.*, 13(1):59–69, 2011.

[333] M. E. Smoot, K. Ono, J. Ruscheinski, P. L. Wang, and T. Ideker. Cytoscape 2.8: new features for data integration and network visualization. *Bioinformatics*, 27(3):431–432, 2011.

[334] Z. D. Blount. The unexhausted potential of E. coli. *Elife*, 4, 2015.

[335] R. Bentley and R. Meganathan. Biosynthesis of vitamin K (menaquinone) in bacteria. *Microbiol. Rev.*, 46(3):241–280, 1982.

[336] D. E. Chang, D. J. Smalley, D. L. Tucker, M. P. Leatham, W. E. Norris, S. J. Stevenson, A. B. Anderson, J. E. Grissom, D. C. Laux, P. S. Cohen, and T. Conway. Carbon nutrition of Escherichia coli in the mouse intestine. *Proc. Natl. Acad. Sci. U.S.A.*, 101(19):7427–7432, 2004.

[337] T. Escherich. The intestinal bacteria of the neonate and breast-fed infant. 1884. *Rev. Infect. Dis.*, 10(6):1220–1225, 1988.

[338] S. T. Shulman, H. C. Friedmann, and R. H. Sims. Theodor Escherich: the first pediatric infectious diseases physician? *Clin. Infect. Dis.*, 45(8):1025–1029, 2007.

[339] P. S. Lee and K. H. Lee. Escherichia coli–a model system that benefits from and contributes to the evolution of proteomics. *Biotechnol. Bioeng.*, 84(7):801–814, 2003.

[340] A. Griswold. Genome Packaging in Prokaryotes: the Circular Chromosome of E. coli. *Nature Education*, 1(1):57, 2008.

[341] G. Kahlmeter. An international survey of the antimicrobial susceptibility of pathogens from uncomplicated urinary tract infections: the ECO.SENS Project. *J. Antimicrob. Chemother.*, 51(1):69–76, 2003.

[342] A. Erb, T. Sturmer, R. Marre, and H. Brenner. Prevalence of antibiotic resistance in Escherichia coli: overview of geographical, temporal, and methodological variations. *Eur. J. Clin. Microbiol. Infect. Dis.*, 26(2):83–90, 2007.

[343] F. D. Lowy. Staphylococcus aureus infections. *N. Engl. J. Med.*, 339(8):520–532, 1998.

[344] G. Y. Cheung, R. Wang, B. A. Khan, D. E. Sturdevant, and M. Otto. Role of the accessory gene regulator agr in community-associated methicillin-resistant Staphylococcus aureus pathogenesis. *Infect. Immun.*, 79(5):1927–1935, 2011.

[345] R. J. Gordon and F. D. Lowy. Pathogenesis of methicillin-resistant Staphylococcus aureus infection. *Clin. Infect. Dis.*, 46 Suppl 5:S350–359, 2008.

[346] M. E. Powers and J. Bubeck Wardenburg. Igniting the fire: Staphylococcus aureus virulence factors in the pathogenesis of sepsis. *PLoS Pathog.*, 10(2):e1003871, 2014.

[347] M. M. Dinges, P. M. Orwin, and P. M. Schlievert. Exotoxins of Staphylococcus aureus. *Clin. Microbiol. Rev.*, 13(1):16–34, 2000.

[348] E. C. Pesci, J. B. Milbank, J. P. Pearson, S. McKnight, A. S. Kende, E. P. Greenberg, and B. H. Iglewski. Quinolone signaling in the cell-to-cell communication system of Pseudomonas aeruginosa. *Proc. Natl. Acad. Sci. U.S.A.*, 96(20):11229–11234, 1999.

[349] G. M. Wiseman. The hemolysins of Staphylococcus aureus. *Bacteriol Rev*, 39(4):317–344, 1975.

[350] J. Bubeck Wardenburg and O. Schneewind. Vaccine protection against Staphylococcus aureus pneumonia. *J. Exp. Med.*, 205(2):287–294, 2008.

[351] K. L. Painter, A. Krishna, S. Wigneshweraraj, and A. M. Edwards. What role does the quorum-sensing accessory gene regulator system play during Staphylococcus aureus bacteremia? *Trends Microbiol.*, 22(12):676–685, 2014.

[352] J. M. Yarwood, D. J. Bartels, E. M. Volper, and E. P. Greenberg. Quorum sensing in Staphylococcus aureus biofilms. *J. Bacteriol.*, 186(6):1838–1850, 2004.

[353] F. D. Lowy. Antimicrobial resistance: the example of Staphylococcus aureus. *J. Clin. Invest.*, 111(9):1265–1273, 2003.

[354] A. Ralston. Operons and Prokaryotic Gene Regulation. *Nature Education*, 1(1):216, 2008.

[355] B. Fournier, X. Zhao, T. Lu, K. Drlica, and D. C. Hooper. Selective targeting of topoisomerase IV and DNA gyrase in Staphylococcus aureus: different patterns of quinolone-induced inhibition of DNA synthesis. *Antimicrob. Agents Chemother.*, 44(8):2160–2165, 2000.

[356] C. Dutta and A. Pan. Horizontal gene transfer and bacterial diversity. *J. Biosci.*, 27(1 Suppl 1):27–33, 2002.

[357] P. M. Bennett. Plasmid encoded antibiotic resistance: acquisition and transfer of antibiotic resistance genes in bacteria. *Br. J. Pharmacol.*, 153 Suppl 1:S347–357, 2008.

[358] M. D. Ermolaeva, O. White, and S. L. Salzberg. Prediction of operons in microbial genomes. *Nucleic Acids Res.*, 29(5):1216–1221, 2001.

[359] F. Jacob and J. Monod. On the Regulation of Gene Activity. *Cold Spring Harb. Symp. Quant. Biol.*, 26:193–211, 1961.

[360] T. Kuhlman, Z. Zhang, M. H. Saier, and T. Hwa. Combinatorial transcriptional control of the lactose operon of Escherichia coli. *Proc. Natl. Acad. Sci. U.S.A.*, 104(14):6043–6048, 2007.

[361] W. S. Reznikoff. The lactose operon-controlling elements: a complex paradigm. *Mol. Microbiol.*, 6(17):2419–2422, 1992.

[362] M. H. Kollef and V. J. Fraser. Antibiotic resistance in the intensive care unit. *Ann. Intern. Med.*, 134(4):298–314, 2001.

[363] S. L. Kaplan. Review of antibiotic resistance, antibiotic treatment and prevention of pneumococcal pneumonia. *Paediatr. Respir. Rev.*, 5 Suppl A:S153–158, 2004.

[364] E. E. Dawson-Hahn, S. Mickan, I. Onakpoya, N. Roberts, M. Kronman, C. C. Butler, and M. J. Thompson. Short-course versus long-course oral antibiotic treatment for infections treated in outpatient settings: a review of systematic reviews. *Fam. Pract.*, 34(5):511–519, 2017.

[365] X. Zeng and J. Lin. Beta-lactamase induction and cell wall metabolism in Gram-negative bacteria. *Front. Microbiol.*, 4:128, 2013.

[366] B. S. Speer, N. B. Shoemaker, and A. A. Salyers. Bacterial resistance to tetracycline: mechanisms, transfer, and clinical significance. *Clin. Microbiol. Rev.*, 5(4):387–399, 1992.

[367] B. Springer, Y. G. Kidan, T. Prammananan, K. Ellrott, E. C. Bottger, and P. Sander. Mechanisms of streptomycin resistance: selection of mutations in the 16S rRNA gene conferring resistance. *Antimicrob. Agents Chemother.*, 45(10):2877–2884, 2001.

[368] A. P. Carter, W. M. Clemons, D. E. Brodersen, R. J. Morgan-Warren, B. T. Wimberly, and V. Ramakrishnan. Functional insights from the structure of the 30S ribosomal subunit and its interactions with antibiotics. *Nature*, 407(6802):340–348, 2000.

[369] P. C. Appelbaum. Microbiology of antibiotic resistance in Staphylococcus aureus. *Clin. Infect. Dis.*, 45 Suppl 3:S165–170, 2007.

[370] G. Sakoulas and R. C. Moellering. Increasing antibiotic resistance among methicillin-resistant Staphylococcus aureus strains. *Clin. Infect. Dis.*, 46 Suppl 5:S360–367, 2008.

[371] A. Brauner, O. Fridman, O. Gefen, and N. Q. Balaban. Distinguishing between resistance, tolerance and persistence to antibiotic treatment. *Nat. Rev. Microbiol.*, 14(5):320–330, 2016.

[372] K. Poole. Efflux pumps as antimicrobial resistance mechanisms. *Ann. Med.*, 39(3):162–176, 2007.

[373] C. I. Montero, M. R. Johnson, C. J. Chou, S. B. Conners, S. G. Geouge, S. Tachdjian, J. D. Nichols, and R. M. Kelly. Responses of wild-type and resistant strains of the hyperthermophilic bacterium Thermotoga maritima to chloramphenicol challenge. *Appl. Environ. Microbiol.*, 73(15):5058–5065, 2007.

[374] J. L. Hansen, P. B. Moore, and T. A. Steitz. Structures of five antibiotics bound at the peptidyl transferase center of the large ribosomal subunit. *J. Mol. Biol.*, 330(5):1061–1075, 2003.

[375] S. Schwarz, C. Kehrenberg, B. Doublet, and A. Cloeckaert. Molecular basis of bacterial resistance to chloramphenicol and florfenicol. *FEMS Microbiol. Rev.*, 28(5):519–542, 2004.

[376] A. Fleming. On the antibacterial action of cultures of a penicillium, with special reference to their use in the isolation of B. influenza. *Br. J. Exp. Pathol.*, 10:226–236, 1929.

[377] J. L. Burns, L. A. Hedin, and D. M. Lien. Chloramphenicol resistance in Pseudomonas cepacia because of decreased permeability. *Antimicrob. Agents Chemother.*, 33(2):136–141, 1989.

[378] P. A. Nielsen and J. A. Close. Edetate disodium-mediated chloramphenicol resistance in Pseudomonas cepacia. *J. Pharm. Sci.*, 71(7):833–834, 1982.

[379] G. D. Wright. The antibiotic resistome: the nexus of chemical and genetic diversity. *Nat. Rev. Microbiol.*, 5(3):175–186, 2007.

[380] V. M. D'Costa, K. M. McGrann, D. W. Hughes, and G. D. Wright. Sampling the antibiotic resistome. *Science*, 311(5759):374–377, 2006.

[381] Z. Gang and F. Jie. The intrinsic resistance of bacteria. *Yi Chuan*, 38(10):872–880, 2016.

[382] G. Cox and G. D. Wright. Intrinsic antibiotic resistance: mechanisms, origins, challenges and solutions. *Int. J. Med. Microbiol.*, 303(6-7):287–292, 2013.

[383] J. G. Lawrence. Horizontal and vertical gene transfer: the life history of pathogens. *Contrib. Microbiol.*, 12:255–271, 2005.

[384] N. Woodford and M. J. Ellington. The emergence of antibiotic resistance by mutation. *Clin. Microbiol. Infect.*, 13(1):5–18, 2007.

[385] S. Grkovic, M. H. Brown, and R. A. Skurray. Transcriptional regulation of multidrug efflux pumps in bacteria. *Semin. Cell Dev. Biol.*, 12(3):225–237, 2001.

[386] H. de Jong. Modeling and simulation of genetic regulatory systems: a literature review. *J. Comput. Biol.*, 9(1):67–103, 2002.

[387] M. H. Nicolas-Chanoine, X. Bertrand, and J. Y. Madec. Escherichia coli ST131, an intriguing clonal group. *Clin. Microbiol. Rev.*, 27(3):543–574, 2014.

[388] J. Sun, Z. Deng, and A. Yan. Bacterial multidrug efflux pumps: mechanisms, physiology and pharmacological exploitations. *Biochem. Biophys. Res. Commun.*, 453(2):254–267, 2014.

[389] S. Kumar, M. M. Mukherjee, and M. F. Varela. Modulation of Bacterial Multidrug Resistance Efflux Pumps of the Major Facilitator Superfamily. *Int. J. Bacteriol.*, 2013, 2013.

[390] J. L. Yu, L. Grinius, and D. C. Hooper. NorA functions as a multidrug efflux protein in both cytoplasmic membrane vesicles and reconstituted proteoliposomes. *J. Bacteriol.*, 184 (5):1370–1377, 2002.

[391] T. Rahman, B. Yarnall, and D. A. Doyle. Efflux drug transporters at the forefront of antimicrobial resistance. *Eur. Biophys. J*, 46(7):647–653, 2017.

[392] C. C. Su, D. J. Rutherford, and E. W. Yu. Characterization of the multidrug efflux regulator AcrR from Escherichia coli. *Biochem. Biophys. Res. Commun.*, 361(1):85–90, 2007.

[393] O. Lomovskaya, K. Lewis, and A. Matin. EmrR is a negative regulator of the Escherichia coli multidrug resistance pump EmrAB. *J. Bacteriol.*, 177(9):2328–2334, 1995.

[394] C. Ma and G. Chang. Structure of the multidrug resistance efflux transporter EmrE from Escherichia coli. *Proc. Natl. Acad. Sci. U.S.A.*, 104(9):3668, 2007.

[395] Q. C. Truong-Bolduc, P. M. Dunman, T. Eidem, and D. C. Hooper. Transcriptional profiling analysis of the global regulator NorG, a GntR-like protein of Staphylococcus aureus. *J. Bacteriol.*, 193(22):6207–6214, 2011.

[396] Q. C. Truong-Bolduc, J. Strahilevitz, and D. C. Hooper. NorC, a new efflux pump regulated by MgrA of Staphylococcus aureus. *Antimicrob. Agents Chemother.*, 50(3):1104–1107, 2006.

[397] Q. C. Truong-Bolduc, P. M. Dunman, J. Strahilevitz, S. J. Projan, and D. C. Hooper. MgrA is a multiple regulator of two new efflux pumps in Staphylococcus aureus. *J. Bacteriol.*, 187(7):2395–2405, 2005.

[398] S. B. Levy and B. Marshall. Antibacterial resistance worldwide: causes, challenges and responses. *Nat. Med.*, 10(12 Suppl):S122–129, 2004.

[399] S. Bagel, V. Hullen, B. Wiedemann, and P. Heisig. Impact of gyrA and parC mutations on quinolone resistance, doubling time, and supercoiling degree of Escherichia coli. *Antimicrob. Agents Chemother.*, 43(4):868–875, 1999.

[400] D. S. Wade, M. W. Calfee, E. R. Rocha, E. A. Ling, E. Engstrom, J. P. Coleman, and E. C. Pesci. Regulation of Pseudomonas quinolone signal synthesis in Pseudomonas aeruginosa. *J. Bacteriol.*, 187(13):4372–4380, 2005.

[401] J. C. Davies. Pseudomonas aeruginosa in cystic fibrosis: pathogenesis and persistence. *Paediatr. Respir. Rev.*, 3(2):128–134, 2002.

[402] N. Høiby, O. Ciofu, and T. Bjarnsholt. Pseudomonas aeruginosa biofilms in cystic fibrosis. *Future Microbiol.*, 5(11):1663–1674, 2010.

[403] A. Eberhard, A. L. Burlingame, C. Eberhard, G. L. Kenyon, K. H. Nealson, and N. J. Oppenheimer. Structural identification of autoinducer of Photobacterium fischeri luciferase. *Biochemistry*, 20(9):2444–2449, 1981.

[404] K. H. Nealson, T. Platt, and J. W. Hastings. Cellular control of the synthesis and activity of the bacterial luminescent system. *J. Bacteriol.*, 104(1):313–322, 1970.

[405] C. Lupp and E. G. Ruby. Vibrio fischeri uses two quorum-sensing systems for the regulation of early and late colonization factors. *J. Bacteriol.*, 187(11):3620–3629, 2005.

[406] W. L. Ng and B. L. Bassler. Bacterial quorum-sensing network architectures. *Annu. Rev. Genet.*, 43:197–222, 2009.

[407] K. Gauwerky, C. Borelli, and H. C. Korting. Targeting virulence: a new paradigm for antifungals. *Drug Discov. Today*, 14(3-4):214–222, 2009.

[408] M. B. Miller and B. L. Bassler. Quorum sensing in bacteria. *Annu. Rev. Microbiol.*, 55: 165–199, 2001.

[409] S. H. Haddock, M. A. Moline, and J. F. Case. Bioluminescence in the sea. *Ann. Rev. Mar. Sci.*, 2:443–493, 2010.

[410] K. H. Nealson and J. W. Hastings. Bacterial bioluminescence: its control and ecological significance. *Microbiol. Rev.*, 43(4):496–518, 1979.

[411] H. B. Kaplan and E. P. Greenberg. Diffusion of autoinducer is involved in regulation of the Vibrio fischeri luminescence system. *J. Bacteriol.*, 163(3):1210–1214, 1985.

[412] J. Engebrecht, K. Nealson, and M. Silverman. Bacterial bioluminescence: isolation and genetic analysis of functions from Vibrio fischeri. *Cell*, 32(3):773–781, 1983.

[413] A. M. Stevens, K. M. Dolan, and E. P. Greenberg. Synergistic binding of the Vibrio fischeri LuxR transcriptional activator domain and RNA polymerase to the lux promoter region. *Proc. Natl. Acad. Sci. U.S.A.*, 91(26):12619–12623, 1994.

[414] L. Zhang, L. Gray, R. P. Novick, and G. Ji. Transmembrane topology of AgrB, the protein involved in the post-translational modification of AgrD in Staphylococcus aureus. *J. Biol. Chem.*, 277(38):34736–34742, 2002.

[415] R. Qiu, W. Pei, L. Zhang, J. Lin, and G. Ji. Identification of the putative staphylococcal AgrB catalytic residues involving the proteolytic cleavage of AgrD to generate autoinducing peptide. *J. Biol. Chem.*, 280(17):16695–16704, 2005.

[416] J. S. Kavanaugh, M. Thoendel, and A. R. Horswill. A role for type I signal peptidase in Staphylococcus aureus quorum sensing. *Mol. Microbiol.*, 65(3):780–798, 2007.

[417] C. L. Malone, B. R. Boles, and A. R. Horswill. Biosynthesis of Staphylococcus aureus autoinducing peptides by using the synechocystis DnaB mini-intein. *Appl. Environ. Microbiol.*, 73(19):6036–6044, 2007.

[418] P. R. Hall, B. O. Elmore, C. H. Spang, S. M. Alexander, B. C. Manifold-Wheeler, M. J. Castleman, S. M. Daly, M. M. Peterson, E. K. Sully, J. K. Femling, M. Otto, A. R. Horswill, G. S. Timmins, and H. D. Gresham. Nox2 modification of LDL is essential for optimal apolipoprotein B-mediated control of agr type III Staphylococcus aureus quorum-sensing. *PLoS Pathog.*, 9(2):e1003166, 2013.

[419] J. P. O'Rourke, S. M. Daly, K. D. Triplett, D. Peabody, B. Chackerian, and P. R. Hall. Development of a mimotope vaccine targeting the Staphylococcus aureus quorum sensing pathway. *PLoS ONE*, 9(11):e111198, 2014.

[420] A. L. Schaefer, B. L. Hanzelka, A. Eberhard, and E. P. Greenberg. Quorum sensing in Vibrio fischeri: probing autoinducer-LuxR interactions with autoinducer analogs. *J. Bacteriol.*, 178(10):2897–2901, 1996.

[421] E. V. Piletska, G. Stavroulakis, K. Karim, M. J. Whitcombe, I. Chianella, A. Sharma, K. E. Eboigbodin, G. K. Robinson, and S. A. Piletsky. Attenuation of Vibrio fischeri quorum sensing using rationally designed polymers. *Biomacromolecules*, 11(4):975–980, 2010.

[422] E. Cavaleiro, A. S. Duarte, A. C. Esteves, A. Correia, M. J. Whitcombe, E. V. Piletska, S. A. Piletsky, and I. Chianella. Novel linear polymers able to inhibit bacterial quorum sensing. *Macromol. Biosci.*, 15(5):647–656, 2015.

[423] J. Park, R. Jagasia, G. F. Kaufmann, J. C. Mathison, D. I. Ruiz, J. A. Moss, M. M. Meijler, R. J. Ulevitch, and K. D. Janda. Infection control by antibody disruption of bacterial quorum sensing signaling. *Chem. Biol.*, 14(10):1119–1127, 2007.

[424] M. Mansson, A. Nielsen, L. Kjærulff, C. H. Gotfredsen, M. Wietz, H. Ingmer, L. Gram, and T. O. Larsen. Inhibition of virulence gene expression in Staphylococcus aureus by novel depsipeptides from a marine photobacterium. *Mar. Drugs*, 9(12):2537–2552, 2011.

[425] E. J. Murray, R. C. Crowley, A. Truman, S. R. Clarke, J. A. Cottam, G. P. Jadhav, V. R. Steele, P. O'Shea, C. Lindholm, A. Cockayne, S. R. Chhabra, W. C. Chan, and P. Williams. Targeting Staphylococcus aureus quorum sensing with nonpeptidic small molecule inhibitors. *J. Med. Chem.*, 57(6):2813–2819, 2014.

[426] E. K. Sully, N. Malachowa, B. O. Elmore, S. M. Alexander, J. K. Femling, B. M. Gray, F. R. DeLeo, M. Otto, A. L. Cheung, B. S. Edwards, L. A. Sklar, A. R. Horswill, P. R. Hall, and H. D. Gresham. Selective chemical inhibition of agr quorum sensing in Staphylococcus aureus promotes host defense with minimal impact on resistance. *PLoS Pathog.*, 10(6): e1004174, 2014.

[427] T. Iwase, Y. Uehara, H. Shinji, A. Tajima, H. Seo, K. Takada, T. Agata, and Y. Mizunoe. Staphylococcus epidermidis Esp inhibits Staphylococcus aureus biofilm formation and nasal colonization. *Nature*, 465(7296):346–349, 2010.

[428] A. Grosdidier, V. Zoete, and O. Michielin. SwissDock, a protein-small molecule docking web service based on EADock DSS. *Nucleic Acids Res.*, 39(Web Server issue):W270–277, 2011.

[429] S. M. Daly, B. O. Elmore, J. S. Kavanaugh, K. D. Triplett, M. Figueroa, H. A. Raja, T. El-Elimat, H. A. Crosby, J. K. Femling, N. B. Cech, A. R. Horswill, N. H. Oberlies, and P. R. Hall. ω-Hydroxyemodin limits staphylococcus aureus quorum sensing-mediated pathogenesis and inflammation. *Antimicrob. Agents Chemother.*, 59(4):2223–2235, 2015.

[430] A. Annapoorani, V. Umamageswaran, R. Parameswari, S. K. Pandian, and A. V. Ravi. Computational discovery of putative quorum sensing inhibitors against LasR and RhlR receptor proteins of Pseudomonas aeruginosa. *J. Comput. Aided Mol. Des.*, 26(9):1067–1077, 2012.

[431] N. S. Schaadt, A. Steinbach, R. W. Hartmann, and V. Helms. Rule-based regulatory and metabolic model for Quorum sensing in P. aeruginosa. *BMC Syst. Biol.*, 7(81), 2013.

[432] P. G. Leonard, I. F. Bezar, D. J. Sidote, and A. M. Stock. Identification of a hydrophobic cleft in the LytTR domain of AgrA as a locus for small molecule interactions that inhibit DNA binding. *Biochemistry*, 51(50):10035–10043, 2012.

[433] L. C. Miller, C. T. O'Loughlin, Z. Zhang, A. Siryaporn, J. E. Silpe, B. L. Bassler, and M. F. Semmelhack. Development of potent inhibitors of pyocyanin production in Pseudomonas aeruginosa. *J. Med. Chem.*, 58(3):1298–1306, 2015.

[434] T. Bjarnsholt, O. Ciofu, S. Molin, M. Givskov, and N. Høiby. Applying insights from biofilm biology to drug development - can a new approach be developed? *Nat. Rev. Drug. Discov.*, 12(10):791–808, 2013.

[435] G. Brackman, P. Cos, L. Maes, H. J. Nelis, and T. Coenye. Quorum sensing inhibitors increase the susceptibility of bacterial biofilms to antibiotics in vitro and in vivo. *Antimicrob. Agents Chemother.*, 55(6):2655–2661, 2011.

[436] U. S. National Library of Medicine. ClinicalTrials.gov, 2000. URL `https://clinicaltrials.gov/`. Accessed on 30.09.2017.

[437] H. Cedar and Y. Bergman. Linking DNA methylation and histone modification: patterns and paradigms. *Nat. Rev. Genet.*, 10(5):295–304, 2009.

[438] S. Zhang, H. Hu, T. Jiang, L. Zhang, and J. Zeng. TITER: predicting translation initiation sites by deep learning. *Bioinformatics*, 33(14):i234–i242, 2017.

[439] P. Ughachukwu and P. Unekwe. Efflux pump-mediated resistance in chemotherapy. *Ann. Med. Health Sci. Res.*, 2(2):191–198, 2012.

[440] C. Holohan, S. Van Schaeybroeck, D. B. Longley, and P. G. Johnston. Cancer drug resistance: an evolving paradigm. *Nat. Rev. Cancer*, 13(10):714–726, 2013.