Monica Golumbeanu

**Statistical Analysis of PAR-CLIP data**

Master’s Thesis

**Supervisors:**  Prof. Erik Aurell, KTH University  
Harri Lähdesmäki, Aalto University  
Prof. Niko Beerenwinkel, ETH Zürich

**Instructor:**  Pejman Mohammadi, ETH Zürich
Preface

The reported work constitutes my Master Thesis and concludes my Master Degree within the Erasmus Mundus Master program in Systems Biology (euSysBio) under the supervision of the KTH and Aalto University. The work presented in the current Thesis has been carried out in the Department of Biosystems Science and Engineering at the ETH University. I would like to express my gratitude towards several persons that have helped me during this process.

I am especially grateful to Professor Erik Aurell and to Nicolas Innocenti who have continuously supervised me during my two years of Master studies and taught me a variety of useful methods in Computational Biology that I have used throughout my projects.

I would like to thank Professor Niko Beerenwinkel and Pejman Mohammadi for offering me the possibility to work in their group as well as for their valuable guidance throughout my work. Moreover, I would also like to thank my colleagues from the BSSE Department for their support and useful ideas.

Furthermore, I would like to thank Harri Lähdesmäki, Professor Juho Rousu and Guillaume Beslon for their help and useful feedback throughout my thesis.

Finally, I would like to thank my friends and my parents for their continuous help and support during these 6 months.

Basel, 3.06.2013

Monica Golumbeanu
Contents

Preface 2

Table of contents 3

Abbreviations and acronyms 4

1 Introduction 5

2 The landscape of RNA-binding proteins 8
   2.1 Design and function of RNA-protein assemblies 8
      2.1.1 Binding mechanisms 8
      2.1.2 mRNPs and miRNPs 9
   2.2 Link to human diseases 10
   2.3 Experimental methods for studying RNA-protein interactions 10
      2.3.1 Pioneering methods 10
      2.3.2 Recent technologies 11
      2.3.3 Detection of RNA-protein interaction sites using PAR-CLIP 11

3 Existing methods for the analysis of PAR-CLIP data 13
   3.1 The CLIPZ environment 13
   3.2 The PARalyzer method 14
   3.3 The wavClusteR package 16

4 Proposed pre-processing pipeline for PAR-CLIP data 19
   4.1 Description of the raw data 19
   4.2 Data pre-processing 20
   4.3 Preliminary analysis 22

5 Proposed model for the analysis of PAR-CLIP data 26
   5.1 Practical assumptions and terminology 26
   5.2 Formal definition of the model 29
      5.3 Inference method 30
         5.3.1 Convergence evaluation 32
         5.3.2 Inference complexity and limitations 32

6 Model testing 35
   6.1 Synthetic experiments 35
      6.1.1 Synthetic data generation method 35
      6.1.2 Analysis of the model convergence 36
      6.1.3 Testing results and analysis 38
   6.2 Real data experiment 41
      6.2.1 Trimming method 41
      6.2.2 Putative binding sites on chromosome 21 42

7 Discussion 44

Appendix 46
Abbreviations and acronyms

**PAR-CLIP** Photoactivatable-Ribonucleoside-Enhanced Crosslinking and Immunoprecipitation

**HEK** Human Embryonic Kidney Cells

**RBP** RNA-binding protein

**RNP** ribonucleoprotein complexes

**miRNP** micro RNA-containing ribonucleoprotein complexes

**mRNP** messenger RNA-containing ribonucleoprotein complexes

**cDNA** complementary DNA

**UV** ultraviolet

**MOV10** Moloney leukemia virus 10

**RSF** relative substitution frequency

**cDNA** complementary DNA

**mRNA** messenger RNA

**miRNA** micro RNA

**pre-miRNA** precursor miRNA

**RBD** RNA binding domain

**ncRNA** non-coding RNA

**siRNA** small interfering RNA

**RISC** RNA induced silencing complex

**KDE** kernel density estimator

**MCMC** Markov Chain Monte Carlo

**SNR** signal to noise ratio

**GR** Gelman-Rubin
1 Introduction

"The pace of discovery is going unbelievably fast."
James D. Watson

Molecular biology has passed through an intensive metamorphosis during the last decades and has ultimately started to prevail in synergy with different other sciences such as statistics, physics and, recently, with computer science. The tradition until the early 90’s was to adopt a reductionist approach in position to study biological systems. Thus, in order to gain more insight, researchers would break a system into its individual parts, study these parts independently and then try to reunite them in a bottom-up manner. In the beginning of 1990, due to technical progress and increasing amount of data, the research methods have started drifting towards a holistic approach which has opened the possibility of analyzing biological systems from a completely new angle. Now, the view over a biological system is not limited to its independent components, but goes beyond a large network of interactions offering a panacea of causes and effects in a wide scale interplay.

The passage from reductionism to holism was interdependent with the rapid advances in technology and especially sequencing. Experiments have gained automation, high scale and high throughput, therefore removing the bottleneck of information on different levels (genome, transcriptome, proteome). The first available genome was that of *Haemophilus Influenzae* and it was published in 1995. Eight years later, the first human genome was being published. Sequencing speed and cost per sequenced base pair was highly optimized by pioneering technologies such as Pyrosequencing released in 2005 [41] and, nowadays, there are technologies allowing sequencing of a human genome during one day for $1,000 [15].

The progress made in both technology and methodology was the driving force for a daunting understanding of underlying molecular processes in biological systems which, before, were complete mystery for scientists. In essence, experiments would reveal that DNA was not the keeper of all the information necessary to define a biological system and, consequently, the central dogma of molecular biology had to be reassessed. As a result, the image of the information flow of a gene which is transcribed into RNA and then translated into protein shifted to the concept of a genome transcribed into transcriptome which then gives rise through translation to the proteome. Long-established beliefs such as, for example, the chromosomes being organized as a "bowl of spaghetti" inside the nucleus were shattered by recent projects such as ENCODE [8]. Another important piece of evidence received from recent research projects was that co- and post-transcriptional mechanisms are much more intricate than it was thought before [35]. Contrary to oversimplified previous beliefs, since its inception from RNA polymerase, RNA is the playground of an overwhelming amount of other entities, such as proteins, which bind, interact and control every step in RNA metabolism.

Active both inside and outside the nucleus, RNA-binding proteins (RBPs) are key players during the lifetime of RNA and coordinate a myriad of processes with high impact on the flow of genetic information in the cell. RBPs posses diversity on several levels, relating to their targets as well as binding specificity, stability, association and function [16]. Both messenger RNAs (mRNA) and non coding RNAs (ncRNA) are the
targets of RBPs and they are bound through a collection of mechanisms such as single and double stranded RNA recognition motifs or binding Zinc fingers [20]. Some of the functions of RBPs include alternative splicing, RNA editing, polyadenylation, RNA export, mRNA localization and translation. Substantial research has been conducted on model organisms such as the nematode, Caenorhabditis elegans, and many RBPs with roles in development have been discovered [24], [18]. In humans, there is a plethora of RBPs interactions as new discoveries are made every year [43].

Since RBPs are actively involved in a multitude of cellular processes, aberrations occurring within their binding mechanism can induce a variety of diseases. For example, one class of diseases commonly caused by malfunctioning RBPs are the neurodegenerative disorders [21]. The reason why these disorders are usually associated with RBPs is because the brain encompasses a considerable amount of alternative splicing events, which intensively involve RBPs. Therefore, gaining more insight into RBPs binding locations has become a prominent subject for research. As a result, the methods detecting the binding sites of proteins and protein complexes onto the RNA have gained particular importance. The first of such methods appeared in 2003 in which the authors utilize an immunoprecipitation approach coupled with microarray profiling (RIP-chip [44]). This methodology was later improved by ultraviolet crosslinking and immunoprecipitation (CLIP [45]) and subsequently, in 2010, by the inclusion of photoactivatable ribonucleosides (PAR-CLIP [19]) which will be described later in the current Thesis. Combining high-throughput sequencing to these protocols has opened the perspective towards genome-wide studies offering at the same time more sequencing depth.

Given the large amount of data produced by the coupling of RBPs binding site detection protocols with next generation sequencing, the detection of the binding sites remains a highly challenging task. Generally, the RBPs sequencing data is processed in three stages: alignment, binding site detection and analysis of the results. Several challenges are encountered within each stage. Due to the short length of the reads obtained by sequencing, the alignment needs to be performed with caution and an optimal combination of alignment parameters needs to be reached. The identification of the binding sites is also not trivial since the data is not noise-free. Due to extensive experimental manipulation, the samples are exposed to high levels of external RNA contamination. As a result, non specific binding may occur and erroneous peaks would be reported and combined with the real ones. In addition to this, the expression level at the binding sites is highly heterogeneous and binding sites with low expression levels are prevalent. Additional spatial analysis is often required in the interest of classifying the functions of a protein based on the locations of the binding sites.

The present Master Thesis focuses on the analysis of PAR-CLIP data and addresses two problems. First, it proposes a general pipeline for the preprocessing of PAR-CLIP sequencing data, yielding appropriate signals ready to be used for the detection of the binding sites. The second goal of the Thesis is the detection of binding sites in essence through a novel statistical model using Bayesian inference in order to separate true binding sites from the noise in an efficient manner.

The next chapter of the thesis provides additional background information in the view of interactions between RNA and proteins as well as the different experimental protocols
used for detecting [RBP] binding sites. The chapter also contains an extensive specification of the PAR-CLIP protocol. Chapter 3 surveys the current literature and presents the existing computational methods for the analysis of PAR-CLIP data exposing their advantages and disadvantages. Chapter 4 proposes an in house pre-processing pipeline for PAR-CLIP data while Chapter 5 presents a statistical model for detecting the RNA-protein binding sites. Chapter 6 reports the results on both synthetic and real data. The Thesis concludes with discussions in Chapter 7.
2 The landscape of RNA-binding proteins

From its creation until its ultimate degradation, RNA is always enveloped with binding proteins. These directly bound proteins form complexes with the nascent RNA, then guide and support the RNA molecule through its different steps of existence (transcription, passage into the cytoplasm, translation and degradation). Overall, RBPs have a primary role in the regulation of gene expression and are protagonists of an extensive range of functions such as alternative splicing, repression, stability, cleavage or localization. Hence, unraveling the intricacies of the gene regulation machinery needs an understanding of how RNA is coordinated by RBPs throughout its metabolism. This chapter describes the means by which RBPs form complexes with RNA molecules, their implications in diverse diseases as well as methods for detecting their interactions.

2.1 Design and function of RNA-protein assemblies

There have been discovered more than 500 RBPs in humans, each with its specific ligation characteristics. The binding apparatus of these proteins resides in the existence of RNA binding domains (RBDs) within the protein structure. RBPs bind to both coding and non-coding RNAs in a sequence-specific, but sometimes also in a sequence-independent manner.

2.1.1 Binding mechanisms

An RBD is a structural unit which contains, on average, less than 100 residues and which is located within an RBP allowing the latter to couple with its RNA targets. The number of discovered types of RBDs resides around 40, the most encountered RBDs being: the RNA recognition motif (RRM), the K homology domain (KH), the double stranded domain (dsRBD) and the Piwi/Argonaute/Zwille (PAZ) domain. Most of these domains use the surface of a β sheet as recognition scaffold.

RRM domain The RNA recognition motif domain prevails the RBPs domains that bind to single stranded RNA in a sequence dependent way. With its structure defined by a four-stranded β sheet arranged across two α helices, the RRM domain comprises between 80 and 90 amino acids and from 4 to 8 recognizable nucleotides. In order to bind to the RNA backbone, the RRM domain employs the Arginine or Lysine residues forming a salt bridge, as well as aromatic residues which create interactions with the bases within the RNA. At present, there are over 10,000 RRMs, typically encountered in RBPs which are key players in post-transcriptional regulation.

KH domain Approximately 70 amino acids form the K homology domain which is organized into a three-stranded β sheet arranged across three α helices. The KH domain comprises 4 recognition nucleotides and, like RRM, binds single stranded RNA in a sequence specific manner. However, the binding mechanism differs from the one of RRMs, being attained through different interactions such as Hydrogen bonds, shape matching or electrostatic synergy.

dsRBD domain The double stranded RNA binding domain has 70 to 90 amino acids and, unlike the two previous domains, binds to double stranded RNA by sequence independent means. Formed of a three-stranded antiparallel β sheet with two α helices on
one of its sides, the dsRBD binds to the RNA by interacting with the $OH$ groups within the spacing between the RNA strands (called also grooves) [20][38].

**PAZ domain** Conventionally encountered in proteins responsible for the metabolism of micro RNA (miRNA), such as Dicer or Argonaute, the PAZ domain is constituted of 110 amino acids and recognizes both miRNAs and small interfering RNAs (siRNA). After target recognition, it attaches via Hydrogen bonds or noncovalent interactions between the aromatic rings [5].

As presented above, most of the RBDs have few recognition nucleotides which confers instability to the binding of singular domains. Furthermore, if RBPs were to possess only one RBD, then the recognition patterns would have been considerably limited. RBPs have managed to overcome the bottleneck of limited variety of RBD by the means of several skillful mechanisms [16]. The modular structure of RBPs allows them to target multiple RNA molecules by combining differently their inherent RBDs.

### 2.1.2 mRNPs and miRNPs

RBPs dynamically bind to all types of RNA molecules forming functional complexes called ribonucleoprotein complexes (RNPs). These units assure the stability of the bound proteins and the integrity of the RNA-protein interactions during different molecular processes [31]. Depending on the type of the bound RNA molecule, RNPs can be classified as micro RNA-containing ribonucleoprotein complexes (miRNP), when the RBPs bind to a miRNA molecule, or messenger RNA-containing ribonucleoprotein complexes (mRNP), when the RBPs bind to an mRNA molecule.

Micro RNAs are non-coding RNAs no longer than 23 nucleotides, that usually are encapsulated into protein complexes which have the main role of repressing translation [6]. Throughout their lifespan, miRNAs are shaped by RBPs. The formed RNP either continuously contribute to the development of the mature miRNA molecule, or collaborate with the included miRNA to regulate gene expression. For instance, the protein Exportin5 binds to precursor miRNAs (pre-miRNA) duplexes and escorts them from the nucleus to the cytoplasm where the Dicer protein binds through a PAZ domain and processes the pre-miRNA into miRNAs [47]. The mature miRNA duplex is subsequently bounded by Argonaute proteins and incorporated into the RNA induced silencing complex (RISC) miRNP [36]. During this process, one strand of the miRNA (called the passenger strand) is usually removed. The RISC complex uses the contained miRNA to recognize mRNA targets where it binds by partial complementarity and performs post-transcriptional gene regulation by repressing translation [17].

Throughout its existence, the mRNA molecule is incorporated into a myriad of RNPs. One example of mRNA is the Exon Junction Complex (EJC) which contains more than 10 proteins bound to RNA splice junctions in a sequence independent manner and which remain bound from transcription until the RNA translation [22].
2.2 Link to human diseases

\(\text{RBPs}\) are involved in several biological processes. Consequently, \(\text{RBPs}\) aberrations may lead to a wide range of diseases, notably, neurodegenerative disorders, muscular deterioration and cancer \[31\]. These aberrations can hold roots in different sources. For example, copy number alterations in the genes essential for the creation of functional \(\text{RNP}\)s may lead to the accumulation of \textit{toxic} RNA associated with different forms of cancer \[27\][48]. A vast variety of \(\text{RBPs}\) disruptions are also associated with different types of cancer since \(\text{RBPs}\) are responsible also for mechanisms such as cell growth, proliferation and apoptosis. Malfunctions encountered within these processes are inevitably triggering cancer \[32\],[12].

As mentioned in the Introduction, aberrant \(\text{RBPs}\) are often associated with neuropathies for the fundamental reason that the brain encompasses a large amount of splicing events which actively involve \(\text{RBPs}\). One such example is the \textit{Fragile X syndrome} (FXS) which is the most prevalent hereditary mental retardation disorder \[1\]. The cause of the disease is an excessive amount of the repeated \textit{CGG} group (200 versus 6 to 54 in the normal case) within the gene \textit{FMR1}. Due to increased amount of \textit{CpG} islands, hypermethilation occurs in these repeating regions which leads to gene silencing and the protein \textit{FMRP} is not produced. The lost protein is an \(\text{RBP}\) usually present in the \(\text{RNP}\) within the cytoplasm of neurons, is responsible for synaptic translation regulation \[31\].

2.3 Experimental methods for studying RNA-protein interactions

Due to their importance in a multitude of regulatory molecular processes, the identification and examination of \(\text{RBPs}\) has gained a considerable attention during the last 30 years \[16\]. Despite a modest beginning consisting mostly of \textit{in vitro} approaches, the recent progress in technology has boosted the experimental procedures opening a plethora of directions and possibilities in the field. Nevertheless, technological progress brought new opportunities but also new questions to be answered. Analyzing \(\text{RBPs}\) and their associated \(\text{RNPs}\) is nowadays not a trivial task, many challenges remaining to be solved.

2.3.1 Pioneering methods

The first attempts to study RNA-protein interactions were exclusively \textit{in vitro} and consisted mainly of mobility shift electrophoresis (EMSA) \[13\] and ultraviolet (UV) crosslinking. One of the pioneering methods in the field is SELEX (Systematic Evolution of Ligands by Exponential Enrichment) \[40\] and consists of an \textit{in vitro} protocol that generates oligonucleotide sequences of RNA or complementary DNA (cDNA) which bind to a given set of proteins, peptides or other organic compounds of limited size. The generated sequences that do not bind to the given ligands are discarded and those that bind are kept and amplified by PCR and then passed through several selection procedures. In the end, the sequences with the strongest binding power remain. The method allows the detection of RNA recognition motifs that strongly bind to their substrates. However, one major disadvantage of the method is that many reported bindings do not occur \textit{in vivo} \[37\][31].
Assays such as RIP (RNA immunoprecipitation) primarily allowed to investigate RNA-protein interactions in vivo and identify individual components of RNPs. The RIP experiments involve formaldehyde-based crosslinking of proteins to RNA and immunoprecipitation of the bound RNA followed by RT-PCR amplification [39]. The combination of RIP with microarray analysis led to Ribonucleoprotein-immunoprecipitation-microarray assays (RIP-ChIP) and allowed the genome-wide detection of RNA sequences bound by the same multi-targeting RBP. One disadvantage of the method is that, due to the aggregation of RBPs into miRNPs, there is a considerable risk of non specific RBPs immunoprecipitation together with the specific RBPs [37]. This effect hardens the exact detection of the RNA - RBP interactions.

Alternative methods of immunoprecipitation and crosslinking involve the use of mass spectrometry. Ribotrap is an in vivo method which focuses on describing the characteristics of RNPs and was first developed on yeast [2]. The procedure employs an engineered reporter RNA sequence which easily binds to a tagged target RBP to form a corresponding RNP. The formed RNP is captured using a bead matrix and immunoprecipitation and its protein contents is further analyzed through methods such as SDS-PAGE, MS or immunoblotting [31][47]. To analyze the captured RNA, techniques such as northern blot, RT-PCR or sequencing can be used.

2.3.2 Recent technologies

Early methods for analyzing RNA-protein interactions gave limited information on the individual binding sites of RBPs and reached very low resolution. This was addressed by later methods such as CLIP (cross-linking and immunoprecipitation) which applies crosslinking by UV radiations of 254 nm and immunoprecipitation of RNA sequences that are bound by a specific RBP [23]. Therefore, separate experiments need to be carried out for different RBPs. After immunoprecipitation, the bound RNA sequences are purified and treated such that the bound protein is removed and the RNA remains unaffected. The obtained RNA can be subject to different procedures such as RT-PCR amplification or sequencing. CLIP followed by high throughput sequencing (CLIP-seq) increases the resolution and enables the search for interaction sites of specific RBPs on a genome-wide scale. Diverse variants of the method (iCLIP [28], PAR-CLIP [19], HITS-CLIP [10]) would eventually be used to construct interaction maps for RBPs.

Another recently developed procedure is CLASH (Crosslinking, ligation and sequencing of hybrids) which attempts to detect RNA-RNA interactions, notably between small RNA bound by proteins and their target RNA molecules [29]. One example of interactions that can be studied with CLASH is between the RISC miRNP complex and the targeted ribosomal RNA.

2.3.3 Detection of RNA-protein interaction sites using PAR-CLIP

A variant of CLIP, the PAR-CLIP protocol, first published in 2010 by Hafner et al [19], detects the binding sites of RBPs throughout the transcriptome. The method is based on the in vivo inclusion of photoreactive ribonucleoside analogs (e.g. $^{4S}U$ (4-thiouridine) and $^{6S}G$ (6-thioguanosine)) into primary RNA transcripts of living cells. This process does not leave any harm to the cells and promotes the RNA crosslinking of RBPs.
Figure 1: Workflow of the PAR-CLIP protocol. 4-thiouridine $^{4}$S\textsubscript{U} is incorporated into the RNA of the cell culture. UV exposure causes the crosslinking of $^{4}$S\textsubscript{U} to miRNP complexes. The bound complexes are immunoprecipitated using a targeted antibody, size selected with SDS-PAGE and then digested with protease. The dissociated RNA sequences are converted into cDNA and subjected to deep sequencing. Due to the incorporated uridine in the RNA sequences, the sequencing machine is more likely to introduce T to C transitions when uridine is present.

In the first step of the protocol (figure 1), cultured living cells are exposed to $^{4}$S\textsubscript{U}. As a result, uridine (U) gets incorporated into the RNA of the living cells. Following UV exposure with a 365 nm wavelength, uridine is crosslinked to a respective RBP. Using an antibody directed against the protein of interest, the fragments of RNA together with the attached proteins are immunoprecipitated. The parts of RNA which are not bound to RBP are cut out using RNase. Size selection by SDS-PAGE of the precipitated proteins retains the RNA fragments bound by the RBP of interest. The selected RNP complexes are thereafter extracted from the gel and digested with protease. The RNA segments are then recovered and converted into cDNA and then passed to a next generation sequencing machine. Because of the incorporated uridine, sequencing errors appear at the crosslinking positions, precisely, induced T to C transitions. These induced transitions are diagnostic of binding sites.
3 Existing methods for the analysis of PAR-CLIP data

Currently, there are several methods for analysis of PAR-CLIP data, nevertheless, only few of them involve rigorous statistical techniques. In the following, we introduce the most well known of these methods and present their advantages and disadvantages.

3.1 The CLIPZ environment

CLIPZ [26] is an online public database resource which displays binding site information for a selection of RBPs. The database contains information from the alignment results of next generation sequencing data following various PAR-CLIP experiments with a standard pipeline published in [3]. Conforming to this pipeline, the reads obtained from the alignment are clipped from their adapters and then the Oligomap software [3] aligns them to the human reference genome as well as to a range of genomes of bacteria, viruses or fungi. The reason for this operation is that during the PAR-CLIP experiment, there is a large number of opportunities to contaminate the sample with external RNA. After the alignment, the reads that overlap by at least one alignment position group into clusters. For each cluster, several characteristic measures such as the number of T to C transitions are computed and available in the database interface. The CLIPZ environment (figure 2) allows to perform a series of operations such as to classify or filter the clusters according to a range of computed statistics. The visualization of binding sites locations is also possible through a Genome Browser.

Figure 2: The overall infrastructure of the CLIPZ environment. A database stores information about numerous PAR-CLIP experiments, notably the alignment results of next generation sequencing reads. The aligned reads that overlap for at least one genomic position are grouped into clusters. The CLIPZ environment allows visualizing the clusters through a genome browser as well as several operations such as cluster sorting, motif finding within clusters or analysis of the dynamics of a cluster through different experiments (super-clustering). Quantitative measures such as the number of T to C transitions are also provided.

Given several different experiments and correspondent obtained clusters, the CLIPZ interface also enables looking at an individual cluster throughout different experiments and see how it changes its configuration (position, length, etc.). In this way, super-clusters
can be formed by overlapping or joining the clusters throughout multiple samples. The possibility to search for the genes that contain the binding sites is also available. In addition to this, a tool allows to identify a specific motif shared by several clusters.

The CLIPZ database is a useful public resource for visualizing PAR-CLIP data and sharing experiments worldwide. Users can add their own data, visualize and compare it with other data sets available throughout the database. Nevertheless, the computational tools are limited since no elaborate statistical tool or modeling is provided. The formed clusters do not account for eventual noise and are very large, therefore, the location of the binding sites stays at a relative level. Increased resolution can be attained through the super-clustering feature but this operation requires at least two available data sets for the same RBP performed within similar experiments.

3.2 The PARalyzer method

Another method dedicated to the study of PAR-CLIP data is PARalyzer [9]. This approach utilizes a Gaussian kernel density based estimator and exploits the rate of $T \rightarrow C$ substitutions in order to detect the RBP’s binding sites.

Figure 3: General workflow of the PARalyzer method. The pre-processing stage is standard and includes read clipping, alignment and clustering. A kernel density estimator (KDE) based method separates the relevant coverage signal from the background noise and is combined with motif finding operations in order to detect the binding sites.

In the pre-processing part of the method (figure 3), next generation sequencing reads are clipped from their adapters and only those larger than 13 nucleotides are kept. Afterwards, the remaining reads are aligned to the genome and overlapping reads are grouped into clusters. The coverage signal is constructed for each cluster. For the alignment, the Bowtie software is used and two mismatches are allowed, both restricted to $T \rightarrow C$. Reads which had any other mismatch than $T \rightarrow C$ substitutions are discarded.

To construct the classifier, two types of kernel density estimates are employed. Precisely, for each cluster which contains at least 5 reads, two densities are estimated, for
the conversion and non conversion of $T \rightarrow C$:

$$
\text{conversion: } f_{T \rightarrow C}(j) = \sum_{i=1}^{L} \frac{x_{T \rightarrow C}^{(i)}}{n_{T \rightarrow C}} \times N(i|j, \lambda^2)
$$

$$
\text{non conversion: } f_{T \rightarrow T}(j) = \sum_{i=1}^{L} \frac{x_{T \rightarrow T}^{(i)}}{n_{T \rightarrow T}} \times N(i|j, \lambda^2)
$$

where $f_{T \rightarrow C}(j)$ is the conversion density and $f_{T \rightarrow T}(j)$ is the non conversion density for a certain read cluster of length $L$. Moreover, $x_{T \rightarrow C}^{(i)}$ represents the number of times a $T \rightarrow C$ conversion occurs at position $i$ in the selected cluster, while $x_{T \rightarrow T}^{(i)}$ represents the number of non conversion events in the given cluster at position $i$. Similarly, $n_{T \rightarrow C}$ represents the total number of $T \rightarrow C$ conversions throughout the entire cluster, while $n_{T \rightarrow T}$ is the total number of non $T \rightarrow C$ conversions in the cluster. The parameter $\lambda$ of the Gaussian kernel estimator has fixed value $\lambda = 3$.

The above computed densities are then normalized in order to provide estimates for the density of $T \rightarrow C$ substitutions or non-substitutions events for each position in a given read cluster:

$$
k_{T \rightarrow C}(j) = \frac{f_{T \rightarrow C}(j)}{\sum_{j=1}^{L} f_{T \rightarrow C}(j)}
$$

$$
k_{T \rightarrow T}(j) = \frac{f_{T \rightarrow T}(j)}{\sum_{j=1}^{L} f_{T \rightarrow T}(j)}
$$

Using the previously calculated estimators, the definition of the classifier is straightforward. Therefore, the positions $j$ where $k_{T \rightarrow C}(j) \geq k_{T \rightarrow T}(j)$, are marked as interaction sites, whereas those where the inequality is not satisfied are considered as non interaction sites.

**Figure 4:** Figure from Corcoran et al [9] displaying an example of use of the kernel density classifier. The regions with a high density of $T \rightarrow C$ substitutions (yellow) are extended by 5 nucleotides to form the final binding site (pink).
Figure 4 illustrates an example of the use of the kernel density classifier to a cluster of reads from the *Pumilio2* library. The regions where the density of $T \rightarrow C$ substitutions (marked as red in the figure) is above the density of non-substitution events (marked as blue in the figure) are considered as binding sites (orange region). These regions are afterwards extended by a fixed number of nucleotides (for example 5 - pink region in the figure) or until they include the ends of the reads involved in the binding site. If, after extension, several detected binding sites overlap, they are grouped into a single binding site.

After the application of the kernel density classifier, PARalyze possesses a downstream motif finding module which allows *de novo* motif finding as well as incorporation and search of already known motifs available in literature. The motif finding module allows refining the found binding sites and also stands as a validation tool.

PARalyze offers an initial dive into the statistical modeling opportunities for analyzing PAR-CLIP data. The kernel density based classifier provides a high-level tool for detecting binding sites. Its particular strength relies in the fact that even though there are no $T \rightarrow C$ substitutions at the binding sites, the influence of the neighboring $T \rightarrow C$ substitutions allow detecting these particular sites. However, there are a couple of residual details of the method that still need to be reasoned. First, the choice of extending the binding sites by 5 nucleotides or the fixing of the bandwidth parameter of the kernel density function at $\lambda = 3$ are both arbitrary and not rigorously justified. In addition to this, the classifier does not provide any significance value. Moreover, the method does not account for non-experimentally induced $T \rightarrow C$ substitutions. As a result, the regions containing $T \rightarrow C$ transitions due to SNPs will have a high conversion density and will most likely predict a binding site even though it is not the case. Finally, allowing only $T \rightarrow C$ mismatches in the alignment procedure biases the global mismatch distribution.

3.3 The wavClusteR package

One of the most recent methods for analyzing PAR-CLIP data was developed by Sievers et al [42] and is available as the R package wavClusteR. The method was used to study the function of the *Moloney leukemia virus 10* (MOV10) protein which is an RNA helicase that usually binds miRNAs forming miRNPs and also interacts with the Argonaute2 and Polycomb proteins [47].

What wavClusteR brings new in comparison to the two previous presented methods is the distinction between experimental and non-experimental $T \rightarrow C$ substitutions. By experimental substitutions, one acknowledges the transitions that were specifically induced by the PAR CLIP protocol while the non experimental substitutions include the variations introduced due to sequencing errors, contamination as well as structural variants (e.g. SNPs).

The method detects the RBP binding sites in a two step approach (figure 5). First, the experimental $T \rightarrow C$ transitions are detected using a mixture model with two components. Second, a peak caller detects preliminary high-confidence positions within the binding site. Once the two steps are completed, the final candidate binding sites (coined clusters) are constructed by extending the found peaks both ways using a difference quotient
**Figure 5**: General workflow of the **wavClusteR** method. Preceded by adapter clipping and alignment, the method detects the binding sites by combining two independent steps. First, it identifies the experimentally induced $T \to C$ substitutions. Second, a peak calling procedure is run throughout the genome and the significant peak locations are retained. Finally, the results of the two analysis are combined such that significant peaks with neighboring experimental $T \to C$ transitions are classified as binding sites.

of the coverage signal. The extended stretch has to encompass at least one found experimental $T \to C$ substitution in order to be considered as candidate binding site.

Each specific type of substitution is seen as a mixture of an experimental and non-experimental component:

$$p_s(x) = \frac{\lambda_{s,1} p_1(x)}{\text{non-experimental}} + \frac{\lambda_{s,2} p_2(x)}{\text{experimental}}.$$  

(3)

and it is assumed that the totality of non-experimental substitutions have the same distribution of occurrence regardless the type of substitution $s$: $p_{s,1}(x) = p_1(x) \forall s$. Moreover, the method assumes that the number of non experimental substitutions is roughly the same for all kinds of substitutions.

In the above mixture model, $p_s(x)$ represents the probability density function of the relative substitution frequency (RSF) $x_{s,i}$ at position $x$ on the genome. The RSF of a particular substitution $s$ at a certain position $i$ on the genome is defined as the number substitutions $s$ encountered at position $i$ ($y_{s,i}$) divided by the coverage at the respective position ($z_i$):

$$x_{s,i} = \frac{y_{s,i}}{z_i}.$$  

(4)

The mixture coefficients $\lambda_{s,k}$ with $k \in \{1, 2\}$ verify $\lambda_{s,k} \geq 0$ and $\sum_{k=1}^{2} \lambda_{s,k} = 1$. In addition to this, due to the fact that PAR-CLIP induces only $T \to C$ substitutions, the mixture model resumes to two equations, according to the type of substitution $s$, i.e. $T \to C$ or non $T \to C$:

\[
\begin{cases}
    p_s(x) = p_1(x) & \text{for } s \neq T \to C \\
    p_s(x) = \lambda_1 p_1(x) + \lambda_2 p_2(x) & \text{for } s = T \to C
\end{cases}
\]  

(5)
where the parameters $\lambda_1$ and $\lambda_2$ are estimated using the following relations:

$$\hat{\lambda}_2 = \frac{f(T \rightarrow C) - \tilde{f}}{f(T \rightarrow C)}$$

$$\hat{\lambda}_1 = 1 - \hat{\lambda}_2$$

The quantity $f(T \rightarrow C)$ represents the number of positions on the genome where there is at least a $T \rightarrow C$ substitution and $\tilde{f} = \arg\max_n f(n)$ is the number of positions on the genome with any type of substitution $n$ except $T \rightarrow C$. The probability density functions $p_1$ and $p_2$ were deduced using a Bayesian setting using the observations of substitutions over the entire genome.

In order to decide if a substitution $s$ at a certain position $x$ is experimentally induced or not, the two probabilities in the equation system (5) are calculated and the larger probability indicates the type of substitution. The method considers only $T \rightarrow C$ transitions that have their RSF values in the interval $[0, 0.7]$. After the experimental $T \rightarrow C$ substitutions are detected, a peak calling step follows where high-confidence peaks are detected by using a wavelet transform of the coverage signal. The provided motivation for the use of the wavelet transform is the assumption that the binding sites are fundamentally narrow rectangle functions. The detected peaks are then extended to the right until a negative difference quotient is followed by a positive one. In the same way, the extension to the left continues until a positive difference quotient is followed by a negative difference quotient. The extended regions are considered valid binding sites or wavclusters if and only if they contain at least one experimentally induced $T \rightarrow C$ substitution.

The wavClusteR method constitutes the most statistically elaborate method among the previously presented PAR-CLIP analysis approaches. Nevertheless, several aspects can be further improved in order to gain more confidence in the returned binding sites. First of all, an enhanced validation of the binding sites is required. The 17053 found wavclusters for MOV10 were only compared against CLIPZ clusters. As a result, only 14.4% wavclusters overlapped with 841 CLIPZ top ranked clusters. Secondly, the procedure employed to estimate the parameters of the mixture model is ambiguous and treats only a special case. Finally, the performed motif analysis did not produce any significant results. The 30 motifs found but these sequences were encountered too seldom among the 17053 binding sites. Another detriment of this method is that the whole analysis takes into consideration only $T \rightarrow C$ transitions which occur at a frequency within the interval $[0.2, 0.7]$, in the rest of the cases the model not being applicable.
4 Proposed pre-processing pipeline for PAR-CLIP data

The last step in the PAR-CLIP protocol is the next generation sequencing of the RNA fragments targeted by the studied RBP. The produced raw data after sequencing cannot be directly used for detecting the binding sites, it needs prior processing in order to be rendered into a utilizable form. In the following, we present a pre-processing pipeline tailored for PAR-CLIP sequencing data which extracts from the raw data the necessary information used afterwards for modeling.

4.1 Description of the raw data

We use a public PAR-CLIP data set published by Sievers et al. [42] which investigates transcriptome-wide the RNA binding affinities of the MOV10 protein in the nucleus of Human Embryonic Kidney Cells (HEK) cells. The MOV10 protein is an RNA putative helicase which interacts with the RISC miRNA in the cytoplasm and therefore it has a key role in the metabolic pathway of miRNA and post-transcriptional gene silencing [7]. When located into the nucleus, MOV10 binds to chromatin [47]. Furthermore, it has been shown that MOV10 interacts with proteins from the Polycomb family which are responsible for various mechanisms determining cell fate and are regularly disrupted in cancer [4]. Precisely, studies have revealed that the knockdown of MOV10 in fibroblasts induces the up-regulation of the tumor suppressor gene INK4a which is responsible for triggering cellular apoptosis [11].

The experimental procedure used to obtain the raw data consisted of the standard PAR-CLIP protocol published in [19] with the following modifications: the nuclei of the cells were isolated prior immunoprecipitation and the MOV10 protein was tagged with Streptavidin for accessible identification [42]. The Solexa Illumina Genome Analyzer platform was used for sequencing.

@SRR490650.2499998 BS-DSU-ELLAC:1::8:6:15454:5237 length=36
ATTTACACATACTGTTTCATCCTAAAATTTAGTCGT
+SRR490650.2499998 BS-DSU-ELLAC:1::8:6:15454:5237 length=36
GHHHHGFGBHEGGGGGGGGEGDDEGDGGGGDDDDG
@SRR490650.2499999 BS-DSU-ELLAC:1::8:6:15508:5241 length=36
GACCAGGCTACACACGTGCTTCTCGTATGCGCTCTTT
+SRR490650.2499999 BS-DSU-ELLAC:1::8:6:15508:5241 length=36
?3,BBB?,BB=,=,='502 (@@BCC8C?????)
...

Figure 6: Extract from the PAR-CLIP raw data file showing the 4 line format used to represent reads in .fastq format. Here, two reads are displayed.

The obtained raw data set consists of a file which follows the .fastq format and contains ~ 60 million reads where each read is represented by four lines in the following format:
• The first line starts with the special character '@' followed by an identifier and an optional description.

• The second line contains the nucleotide sequence of the read.

• The third line starts with the special character '+' and is followed by the same identifier as on the first line.

• The forth line contains the qualities for every nucleotide in the read sequence from the second line; these qualities are ASCII encoded.

Figure 6 displays an extract of the .fastq file containing the reads obtained after the sequencing in PAR-CLIP.

4.2 Data pre-processing

In order to be able to use the provided data for modeling the RBP binding sites, we set up a fully automated pipeline which pre-processes the raw data. The pipeline (figure 7) consists of three modules: adapter clipping, alignment and summary.

![Pipeline Diagram]

**Figure 7:** Overall workflow of the proposed PAR-CLIP pre-processing pipeline. The procedure comprises three main steps: clipping the adapters from the reads, alignment of the clipped reads and the construction of an alignment summary.

**Adapter Clipping** Some of the reads within the raw data usually come with an attached part of the sequencing adapter. In order to be used for further analysis, the adapter sequences need to be clipped off. We perform the clipping operation with the fastx_clipper tool from the FASTX toolbox. The searched adapter sequence is: TCGTATGCCGTCTTCTGCTTG. We discard all the read sequences which after adapter removal are less than 13 nucleotides long. After the clipping of adapters, approximately 6% of the reads were discarded and the average length of the remaining reads was approximately 20 nucleotides (figure 8).

---

1Freely available at [http://hannonlab.cshl.edu/fastx_toolkit/index.html](http://hannonlab.cshl.edu/fastx_toolkit/index.html)
Read alignment
The next step in the data pre-processing is the alignment of the clipped reads. The alignment operation consists in finding, for each read, its origin(s) on the Human Genome. Due to its performance with short read sequences, the bowtie aligner [30] was used with the alignment parameters displayed in table 1. The reference assembly sequence used was *Homo Sapiens GRCh37* [2]. With the specified settings, approximately 47% of the available clipped reads were aligned to the reference genome. The bowtie parameters were empirically optimized such that a high $T \rightarrow C$ general mismatch ration would be obtained.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>seed length</td>
<td>50</td>
</tr>
<tr>
<td>number of allowed mismatches</td>
<td>3</td>
</tr>
<tr>
<td>number of allowed alignments</td>
<td>100</td>
</tr>
<tr>
<td>reported alignments per read</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 1: Main parameters used for the alignment with Bowtie. Three mismatches were allowed for each alignment, only reads with less than 100 alignments were considered and only the best quality alignment was reported.

The reason why we allow reads to have 3 mismatches is in order to be able to detect as much $T \rightarrow C$ transitions as possible. A strict alignment procedure would discard a large quantity of reads with $T \rightarrow C$ substitutions which would remove essential information for the identification of the binding sites.

Read Summary After alignment, a summary of the reads is constructed. The summary contains, for each position on the genome, the number and types of mismatches which

\[^{2}\text{described at }\url{http://www.ensembl.org/Homo_sapiens/Info/Index}\]
have occurred at that specific position (figure 9). This file can be afterwards efficiently processed to represent information in different convenient ways. In the following, we will present several applications of preliminary analysis based on the summary file.

# Reference nucleotides (Column 3) are encoded as: A:1 C:2 G:3 T:4
#
# Chr Pos Ref.nt. A C G T N -A -C -G -T -N cov -cov
#
1 60120 1 2 0 0 0 0 0 0 0 0 2 0
1 60121 4 0 0 0 2 0 0 0 0 0 0 2 0
1 60122 4 0 0 0 2 0 0 0 0 0 0 2 0
1 60123 4 0 0 0 2 0 0 0 0 0 0 2 0
1 60124 3 0 0 2 0 0 0 0 0 0 0 2 0
1 60125 1 2 0 0 0 0 0 0 0 0 0 2 0

Figure 9: Extract from the read summary. The first column represents the chromosome code, the second column reports the position on the chromosome and the third column encodes the reference nucleotide (1 for A, 2 for C, 3 for G and 4 for T). The 10 following columns display how many mismatches of each type are encountered at the respective position on the forward and reverse (-) strand. The column N stands for an unidentified nucleotide. The last two columns give the coverage at the current position for the forward and reverse strand.

The created read summary facilitates generation of two signals for downstream data analysis: read coverage and read depth. Both being a function of the position \( i \) on the genome, these signals provide information about the number of aligned reads that map at position \( i \) (read coverage) and, respectively, the number of reads whose alignment starts at position \( i \) on the genome (read depth).

4.3 Preliminary analysis

Starting from the read summary, we already have the possibility to conduct a feasibility study of the provided data. We address several inquiries related to the \( T \rightarrow C \) substitution patterns throughout the genome. First, we perform a similar analysis as [42] to compare the mismatch profiles of the different types of substitutions within a certain mismatch frequency boundary. Afterwards, we explore the distribution of \( T \rightarrow C \) substitutions given a certain mismatch frequency.

The first question we want to address is whether the substitution signals presented in [42] (figure 10) are present for the alignment parameters we used. Sievers et al observe that the \( T \rightarrow C \) substitutions are prevalent at relative substitution frequencies (RSFs) less than 82%. We recall that the RSF for a specific type of substitution \( s \) at a position \( i \) on the genome is given by the ratio of the amount of type \( s \) substitutions within the reads aligned at position \( i \) and the total number of substitutions that take place at the respective position. Sievers et al use a very strict alignment protocol, only one mismatch being allowed and all the clipped reads shorter than 15 nucleotides are discarded.
Figure 10: Mismatch profiles obtained by Sievers et al (figure source: [42]) by counting the amount of substitutions at positions on the genome within two classes of RSFs. The first class (sub-figure a) considers the genomic positions whose RSFs fall in the interval \([0.82, 1]\), while the second class (sub-figure b) takes into account only the positions with RSFs in \([0.1, 0.82)\). Only positions with read coverage larger than 20 were counted. A strong prevalence of \(T \rightarrow C\) substitutions is observed in the second case. The used alignment procedure allows only one mismatch and all the clipped reads shorter than 15 nucleotides are discarded.

Figure 11 shows the mismatch profiles obtained from our pre-processing pipeline described in the previous section. The appearance of the mismatch profiles is similar to the result obtained by Sievers et al (figure 10) for both classes of RSFs. The less strict mismatch constrain (3 versus 1) leads to a higher amount of reads retained for further analysis. This result suggests that permitting more than one mismatch does not dilute away the signal from diagnostic mutations, on the contrary, it provides more information about the binding sites by accumulating a larger number of \(T \rightarrow C\) substitutions.

Figure 11: Mismatch profiles obtained with the in house designed pre processing pipeline. Three mismatches were allowed and only positions with read coverage larger than 20 were counted. As observed in [42], the \(T \rightarrow C\) substitutions dominate when the mismatch frequency is less than 82%. The mismatch profiles keep their shape as in [42] even though a less strict alignment setting is used.
Further, we studied the distribution of the probability of having a $T \rightarrow C$ substitution in regions of the genome given a certain mismatch frequency. To calculate this probability, we introduce several quantities and notations.

A mismatch $s$ is comprised of two elements: the nucleotide in the reference genome and the mismatch nucleotide. For example, in a $T \rightarrow C$ mismatch, $T$ would be the reference nucleotide and $C$ would be the mismatch nucleotide. Let $r(s)$ give the reference nucleotide of the mismatch $s$ i.e. $r(T \rightarrow C) = T$.

For each position $i$ on the genome, we introduce the mismatch frequency $f_{i,s,R}$ as the ratio between the total number $n_{i,s}$ of mismatches $s$ at position $i$ and the total number $N_{i,R}$ of mismatches at position $i$ which have their reference nucleotide the same as $s$, i.e. $R = r(s)$:

$$f_{i,s,R} = \frac{n_{i,s}}{N_{i,R}}, \text{ with } r(s) = R$$

and:

$$N_{i,R} = \sum_{s, \ r(s)=R} n_{i,s}.$$  \hfill (8)

Additionally, at each position on the genome, we define the marginal mismatch frequency $M_{i,R}$ as the ratio of mismatches which have the reference nucleotide $R$ out of the total number of aligned reads at the respective position:

$$M_{i,R} = \frac{N_{i,R}}{c_i}$$

where $c_i$ is the total number of aligned reads at position $i$ or the coverage of position $i$.

Finally, we separate the interval $[0, 1]$ in bins $B$ such as:

$$[0, 1] = \{[0, 0.1), [0.1, 0.2), ..., [0.9, 1]\}.$$  \hfill (10)

For each bin and each reference nucleotide, we associate a mismatch set $M_{B,R}$ defined as the set of all the positions on the genome where the previously defined marginal mismatch frequency is in the bin $B$:

$$M_{B,R} = \{i | M_{i,R} \in B\}.$$  \hfill (11)

We define the probability of having a substitution $s$ given a frequency bin $B$, as the weighted average of mismatch frequencies for positions where the marginal mismatch frequency is in the bin $B$:

$$P(s|B) = \frac{\sum_{i \in M_{B,R}} f_{i,s,R} N_{i,R}}{\sum_{i \in M_{B,R}} N_{i,R}} = \frac{\sum_{i \in M_{B,R}} n_{i,s}}{\sum_{i \in M_{B,R}} N_{i,R}}$$

For the particular case $s = T \rightarrow C$, the probability $P(T \rightarrow C|B)$ is written as:

$$P(T \rightarrow C|B) = \frac{\sum_{i \in M_{B,T}} n_{i,s}}{\sum_{i \in M_{B,T}} N_{i,T}}.$$
Figure 12: Conditional probability distribution of having a particular substitution type for various mismatch frequencies for the forward strand of the genome given the marginal mismatch frequency. We observe that there is a high chance of getting $T \rightarrow C$ substitutions notably in regions with a mid-range mismatch frequency. As the marginal mismatch frequency increases, the chance of those mismatches being $T \rightarrow C$ transitions decreases.

Figure [12] displays, for different possible marginal mismatch frequency bins and using LOWESS smoothing, the distributions of the chance of getting different types of substitutions. As a result, the less mismatch frequency, the higher the probability of having $T \rightarrow C$ substitutions. As the mismatch frequency grows, there is a higher chance to deal with SNPs and not with induced $T \rightarrow C$ substitutions. This result confirms the existence of experimentally induced substitutions by the PAR-CLIP protocol.

The current preliminary analysis enables an initial enclosure of the PAR-CLIP data and provides some insight into the mismatch distribution throughout the genome. In the following, we will use the signals generated by pre-processing in a Bayesian statistical model in order to identify the RBP binding sites.
5 Proposed model for the analysis of PAR-CLIP data

The detection of RBP binding sites is crucial for understanding the underlying functions of RBPs and gaining more insight into the RNA regulatory processes. Nevertheless, the identification of these sites is not a trivial task due especially to noise and to the large size of the genomic data.

The Bayesian statistical framework offers a wide range of modeling possibilities and constitutes a useful tool for analyzing PAR-CLIP data for different reasons. First of all, it provides flexibility to represent details of biological processes in a formal scheme at the same time being able to account for noise and ambiguities in systematic fashion. Second, there exists a diverse and well established palette of inference strategies for Bayesian models which can be applied based on the complexity and size of the model.

Bayesian modeling supposes building a probabilistic model and fitting it to observed data in order to estimate its parameters and learn about unobserved quantities. Here we propose a Bayesian model to predict the starting points of the binding sites (coined as hits). The model is provided with two sources of information based on the amount of aligned reads as well as their observed T to C substitutions.

5.1 Practical assumptions and terminology

In order to determine the starting points of the binding sites, we use two sources of evidence, namely read depth and mismatch depth. As described before, the read depth of a certain genomic position is defined as the number of reads starting at that position. Similarly, the mismatch depth of a given genomic position is defined as the number of reads whose alignments start at that position and who carry at least one $T \rightarrow C$ substitution within their length. We designate $X_n$ as the observed read depth at position $n$, and $Y_n$ as the correspondent observed mismatch depth.

![Figure 13: The expansion effect observed within the read depth signal of real data.](image)

Ideally, one would expect that the aligned reads (grey segments) pile up yielding clear starting points (sub-figure a). In reality, the reads are distributed over a neighborhood, most of them falling over the central position (sub-figure b). We associate to every position $n$ on the genome a latent amount of reads, $\lambda_n$, which, when aligned, gets distributed within a certain region of length $L$. 

26
In an ideal setting, the starts of the binding sites would be represented by a pile of aligned reads, all starting at the same point (figure 13-a). However, in reality, when we look at the read depth signal throughout the genome, the starting sites tend to expand and occupy several positions (figure 13-b). We refer to this phenomenon as the "expansion effect". The causes of this effect range from the existence of mismatches within the reads to the presence of experimental noise. For this reason, for every position \( n \) on the genome, we consider that there exists a latent amount of starting aligned reads, \( \lambda_n \), which gets distributed within a certain region of length \( L \) (expansion length), and that the real starting point of the alignment is at the beginning of this region.

![Empirical shape of the read depth peaks for Chromosome 1](image)

**Figure 14:** Empirical shape for the local peaks of Chromosome 1. The shape was constructed by selecting all the local peaks which were higher than 50 from the read depth signal of Chromosome 1, normalizing them according to their height and then averaging, for each position of the peak (0 denotes the center of the peak), the corresponding ratio.

In order to set a general idea about the expansion shape of the \( \lambda_n \) amount of reads, we have studied the read depth signal of Chromosome 1 from the MOV10 dataset published in [42]. Precisely, we have identified all the local peaks where the read depth is higher than 50, we have normalized the values within the peak shoulders with regard to the height at the center and then averaged the values for each position within the shoulder (figure 14). The obtained empirical shape suggests the assumption that the latent expression level \( \lambda_n \) of a peak is distributed symmetrically and monotonically decreasing from the center towards the boundaries of the peak.

We model the symmetrical distribution of \( \lambda_n \) within an expansion region of length \( L \) (figure 15-a) by using a set of probabilities \( p_i, 1 \leq i \leq L \), such that:

\[
\lambda_n = \sum_i \lambda_n p_i, \quad \sum_i p_i = 1 \tag{14}
\]

and \( p_i = p_{L-i+1} \) (symmetric expansion) with the central probability, \( p_{[L/2]} \), being the largest.

We assume that the expansion pattern is shared by all the binding sites, thus the \( p_i \) probabilities being the same for all the expansion regions of interest.
Figure 15: Modeling of the expansion effect for the special case when the length of the expansion is $L = 3$. (a) The latent amount of reads $\lambda_n$ is allocated over three consecutive positions with probabilities $p_1$, $p_2$ and $p_3$ such that $p_3$ is the largest, $p_1 = p_2$ and $p_1 + p_2 + p_3 = 1$. The distributed amounts of reads will be $\lambda_np_1$, $\lambda_np_2$ and $\lambda_np_3$. (b) Illustration of the situation of overlapping expansion. The distributed $\lambda_n$, $\lambda_{n+1}$ and $\lambda_{n+2}$ contribute to the observed read depth at position $n + 1$. The scale has been truncated for an accessible understanding.

One important aspect that needs to be taken into consideration is that, at position $n$, the observed read depth $X_n$ consists of the sum of the individual contributions of the neighboring $\lambda$s. For example, if $L = 3$ (figure 15-b), then $X_{n+1}$ will receive the effect from $\lambda_n$, $\lambda_{n+1}$ and $\lambda_{n+2}$ allocated through their corresponding expansion probabilities $p_i$. In the general case, for a given genomic position $n$, this can be written as:

$$E[X_{n+1}] = \lambda_n p_3 + \lambda_{n+1} p_2 + \lambda_{n+2} p_1$$

In the general case, for a given genomic position $n$, this can be written as:

$$E[X_n] = \sum_{i=1}^{L} p_i \lambda_{n+\left\lfloor \frac{L}{2} \right\rfloor - i + 1}$$

where $\left\lfloor \frac{L}{2} \right\rfloor$ represents the floor function of the ratio $\frac{L}{2}$ (e.g. $\left\lfloor \frac{3}{2} \right\rfloor = 1$ if $L = 3$).

Similar assumptions made for modeling the expansion effect encountered for the read depth are applied for modeling the $T \rightarrow C$ mismatch depth, $Y_n$. We assume that the expansion effect of the number of starting aligned reads $\lambda_n$ over a region of length $L$ is transmitted to the observed number of reads that have at least a $T \rightarrow C$ mismatch. We assign for every position on the genome a universal probability $p_p$ of having a PAR-CLIP induced $T \rightarrow C$ substitution. Likewise, we also introduce a probability $p_0$ of having a $T \rightarrow C$ substitution by chance at a genomic position. Hence, we can write an equivalent rule to equation 15 for the contribution of neighboring peaks to $Y_n$:

$$E[Y_n] = (p_0 + p_p - p_0p_p) \sum_{i=1}^{L} p_i \lambda_{n+\left\lfloor \frac{L}{2} \right\rfloor - i + 1}$$

Equations 15 and 16 represent the fundamental relations for the model. In the following, we will slightly modify these two rules in order to take into account additional components such as the noise and to establish a complete and functional definition of the model.
5.2 Formal definition of the model

The proposed model combines two sources of information from the observed random variables $X_n$ (read depth at position $n$) and $Y_n$ ($T \rightarrow C$ mismatch depth at position $n$). We introduce a binary latent random variable $H_n$ to represent whether a binding site starts at position $n$ or not. This random variable would allow us to couple or decouple the observations $X_n$ and $Y_n$ within the model. In addition to this, we choose a supplementary random variable $\lambda_0$ to describe a global background noise level in the read depth signal. Figure 16 displays the full graphical representation of the proposed model with the described dependency structure.

![Graphical representation of the proposed model](image)

**Figure 16:** Graphical representation of the proposed model. The random variables are enclosed with a circle symbol. Variables $X_n$ and $Y_n$ are observed, all the rest being latent. The plates indicate an iteration over the length $L$ of a starting site and $N$ genomic positions. All the model dependencies are specified through directed arrows. The dashed arrow between the nodes $H$ and $X_n$ indicates the possibility of coupling or decoupling.

The latent variable $H_n$ allows us to explore two inference scenarios by coupling or decoupling the two sets of observations $X_n$ and $Y_n$. In the coupled case, we assume that when a position is a hit, then we expect to see both a large read depth as well as a high corresponding $T \rightarrow C$ depth. Inversely, if a position is not a hit, the read depth will be low. In the decoupled case, the read depth is used only to deconvolute the expansion effect within the shape of a hit, while the $T \rightarrow C$ depth will be suggestive for a position being a hit or not. In the following, we will refer to these two scenarios as the **coupled** and **decoupled** forms of the model.

Using the assumptions made in the previous section, notably equations 15 and 16 as well as the newly introduced random variables $H_n$ and $\lambda_0$, we can write the probabilistic
definition of the coupled form of the model as follows:

\[
X_n \sim \text{Pois} \left( \lambda_0 + \sum_{i=1}^{L} p_i \lambda_m H_m \right)
\]

\[
Y_n \sim \text{Pois} \left( p_0 \lambda_0 + (p_0 + p_p(1-p_0)) \sum_{i=1}^{L} p_i \lambda_m H_m \right)
\]

\[
m = n + \left\lfloor \frac{L}{2} \right\rfloor - i + 1
\]

where Pois denotes the Poisson distribution. We use the notation \(X_n\) to denote the random variable associated to the observed data point \(x_n\) and we note with \(X\) the complete set of observed values: \(X = \{x_n\}_{n=1}^{N}\). The same notations are used for \(Y_n\).

Similarly, the definition of the decoupled form of the model is:

\[
X_n \sim \text{Pois} \left( \lambda_0 + \sum_{i=1}^{L} p_i \lambda_m \right)
\]

\[
Y_n \sim \text{Pois} \left( p_0 \lambda_0 + (p_0 + p_p(1-p_0)) \sum_{i=1}^{L} p_i \lambda_m H_m + p_0 \sum_{i=1}^{L} p_i \lambda_m (1 - H_m) \right)
\]

\[
m = n + \left\lfloor \frac{L}{2} \right\rfloor - i + 1
\]

Note the additional term \(p_0 \sum_{i=1}^{L} p_i \lambda_m (1 - H_m)\) which introduces the possibility of having a high read depth with a low corresponding \(T \rightarrow C\) depth when there is no hit, situation which is not handled by the coupled model.

Table 2 completes the given definition of the proposed model providing a full specification of the parameters and the chosen distributions for their priors.

<table>
<thead>
<tr>
<th>parameter</th>
<th>modeled quantity</th>
<th>prior distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>(H_n)</td>
<td>presence of a binding site at position (n)</td>
<td>Bernoulli((p_h))</td>
</tr>
<tr>
<td>(\lambda_n)</td>
<td>latent expression level</td>
<td>Gamma((\alpha_l, \beta_l))</td>
</tr>
<tr>
<td>(\lambda_0)</td>
<td>background noise</td>
<td>Gamma((\alpha_0, \beta_0))</td>
</tr>
<tr>
<td>(p_i)</td>
<td>expansion probability</td>
<td>Dirichlet(\frac{1}{L})</td>
</tr>
<tr>
<td>(p_0)</td>
<td>probability of a (T \rightarrow C) transition by chance</td>
<td>Beta((\alpha_{p0}, \beta_{p0}))</td>
</tr>
<tr>
<td>(p_p)</td>
<td>probability of an induced (T \rightarrow C) transition</td>
<td>Beta((\alpha_{pp}, \beta_{pp}))</td>
</tr>
</tbody>
</table>

Table 2: Description of the model parameters and their associated prior distributions.

5.3 Inference method

Our proposed model consists of 6 classes of variables: \(H_n, \lambda_n, \lambda_0, p_i, p_0\) and \(p_p\). The absolute number of parameters is \(2N + L + 3\) where \(N\) represents the length of the genomic sequence (e.g. chromosome, genome) and \(L\) is the expansion length of the peaks at the starting positions of the binding sites. Taking into consideration the definition rules of
the model and the prior distributions of the random variables (Table 2), the posterior distribution of the model \( (M) \) can be written as:

\[
P(M|D) \propto P(D|M)P(M)
\]

\[
\propto P(X,Y|M)P(M)
\]

\[
\propto \left( \prod_{n=1}^{N} P(x_n|M) \right) \left( \prod_{n=1}^{N} P(y_n|M) \right) P(M)
\]

\[
\propto \left( \prod_{n=1}^{N} \text{Pois}(x_n, \Lambda_{x_n}) \right) \left( \prod_{n=1}^{N} \text{Pois}(y_n, \Lambda_{y_n}) \right) P(M)
\]

(19)

where \( D \) represents the complete observed data which in our case are the two observed signals \( X = \{x_n\}_{n=1}^{N} \) and \( Y = \{y_n\}_{n=1}^{N} \). Both \( \Lambda_{x_n} \) and \( \Lambda_{y_n} \) are used for simplification and stand for the right hand sides in the equation systems 17 or 18, depending on which form of the model is used (coupled or decoupled):

\[
\Lambda_{x_n} = \lambda_0 + \sum_{i=1}^{L} p_i \lambda_m H_m
\]

\[
\Lambda_{y_n} = p_0 \lambda_0 + (p_0 + p_p(1 - p_0)) \sum_{i=1}^{L} p_i \lambda_m H_m
\]

for the coupled form, or

\[
\Lambda_{x_n} = \lambda_0 + \sum_{i=1}^{L} p_i \lambda_m H_m
\]

\[
\Lambda_{y_n} = p_0 \lambda_0 + (p_0 + p_p(1 - p_0)) \sum_{i=1}^{L} p_i \lambda_m H_m + p_0 \sum_{i=1}^{L} p_i \lambda_m (1 - H_m)
\]

(21)

for the decoupled form, \( m = n + \left\lfloor \frac{L}{2} \right\rfloor - i + 1 \).

The overall prior probability \( P(M) \) can be written as:

\[
P(M) = \text{Ber}(p_h)G(\alpha_l, \beta_l)G(\alpha_0, \beta_0)D(\frac{1}{L})B(\alpha_{p0}, \beta_{p0})B(\alpha_{pp}, \beta_{pp})
\]

(22)

where \( \text{Ber} \) stands for the Bernoulli distribution, \( \mathcal{G} \) stands for the Gamma distribution, \( \mathcal{D} \) represents the Dirichlet distribution and \( \mathcal{B} \) the Beta distribution.

Given the complex form of the posterior, the model definition cannot be reduced to a closed form. One way to alleviate this problem is through sampling. To estimate all the model latent variables and parameters, we use Markov Chain Monte Carlo (MCMC) sampling and uninformative priors for the parameter distributions.

The model was implemented over the OpenBUGS platform [34]. OpenBUGS is a freely-distributed package which allows the specification of statistical models by writing the dependencies between variables in a special format (BUGS format). The software then detects the most suitable MCMC method for inference (usually based on Gibbs sampling) and performs the sampling procedure. A specialized user interface allows monitoring the estimated variables throughout the sampling process.
5.3.1 Convergence evaluation

MCMC inference techniques are based on the common principle of sampling from specified probability distributions and forming a Markov chain which would have the stationary distribution of interest. The sampling process usually takes a large amount of steps until it converges towards the true posterior distribution. Therefore, one needs to be sure that the sampler has reached convergence before starting to interpret the estimated parameters of the model.

There are several sanity checks for convergence diagnosis. The straightforward way is to visually observe the evolution of the model parameters throughout the successive iterations and assume convergence when the estimated values do not suffer systematic changes over a large amount of iterations. This can be done with more confidence through several chains and the Gelman-Rubin (GR) convergence criterion [14] may then be used. According to this criterion, when the GR statistic stabilizes to a rule of thumb value of 1.1, then the MCMC sampling is considered to have reached the steady state.

Due to the extensive number of parameters in our model, we use a simplification and assume that the overall convergence over the entire set of parameters coincides with the convergence of the log likelihood of the observed data. Therefore, instead of observing all the parameters of the model, we only look at the evolution of the log likelihood as a proxy. The log likelihood of the data can be expressed as following:

\[
\log P(D|M) = \log P(X,Y|D) = \log \left[ \left( \prod_{n=1}^{N} P(x_n|M) \right) \left( \prod_{n=1}^{N} P(y_n|M) \right) \right] = \log \left[ \left( \prod_{n=1}^{N} \text{Pois}(x_n, \Lambda_{x_n}) \right) \left( \prod_{n=1}^{N} \text{Pois}(y_n, \Lambda_{y_n}) \right) \right] \\
= \log \left[ \left( \prod_{n=1}^{N} \frac{\Lambda_{x_n}^{x_n}x_n^{x_n-1}e^{-\Lambda_{x_n}}}{x_n!} \right) \left( \prod_{n=1}^{N} \frac{\Lambda_{y_n}^{y_n}y_n^{y_n-1}e^{-\Lambda_{y_n}}}{y_n!} \right) \right] \\
\propto \left( \sum_{n=1}^{N} x_n \log \Lambda_{x_n} \right) - N\Lambda_{x_n} + \left( \sum_{n=1}^{N} y_n \log \Lambda_{y_n} \right) - N\Lambda_{y_n}
\]

5.3.2 Inference complexity and limitations

The large number of parameters of the model which is proportional to the length of the observed signals places a considerable burden over the inference task. The speed of the sampling operation and parameter convergence decreases as the size of the input genomic sequence increases. Therefore, it is impossible to apply the model implementation directly on the entire human genome or even on one chromosome and additional optimization steps are required in order to be able to apply the model on the real data.

One way to rend the model applicable to the real data is to artificially "reduce" the size of the input genomic sequence. This idea originates from the observation that the read depth signal is very sparse throughout the genome. A very large amount of genomic positions do not have any correspondent aligned reads. For example, on chromosome 1, 99.8% of the positions are null. Consequently, replacing the long stretches of 0 with
a reduced amount of zeros (for example $L_i$), would considerably reduce the size of the input data. In addition to this, additional trimming procedures such as the removal of the $0 - 1 - 0$ islands in the signal may be also appropriate.

Another problem that may appear within the inference scheme is the risk that the sampling process would get stuck in a local state which does not necessarily correspond to the true posterior. This would subsequently lead to erroneous estimates. In order to mitigate this effect, we use several MCMC chains and we stop when all the chains have converged to the same state.

The sampling convergence speed is also proportional to the length of the input genomic sequence. Even for a measure order of a thousand base pairs, reaching convergence may require a large number of iterations leading to several hours of execution. A common method for the contraction of the convergence time is the initialization of the sampler with "smart" initial values so that it will know earlier the "good direction" towards the steady state. In the following, we propose three algorithms for the identification of initial values for the variables $H_n$, $p_i$ and $\lambda_n$.

**Initial values for $H_n$** One way to initialize our binary indicator for the starts of the binding sites is to set to 1 all the positions where local peaks occur in the signal $\bar{X}$ (algorithm 1). The signal $\bar{X}$ is obtained by sliding a window of length $L$ over the observed read depth signal $X$ and summing up the values within the window. The rationale behind this initialization is that the start sites we are looking for are expanded over regions of size $L$ and the real starting sites are located at the beginning of these regions.

**Algorithm 1:** Computation of the initial values $h_n$ for the random variable $H_n$.

1. Input: $X$, $L$;
2. Construct $\bar{X}$ by sliding a window of length $L$ over $X$ and summing up the contained values.;
3. for each element $\bar{x}_n$ of $\bar{X}$ do
   - if $\bar{x}_n$ is a local peak in $\bar{X}$ then $h_n \leftarrow 1$;
   - else $h_n \leftarrow 0$;
end
4. Return $h_n$;

**Initial values for $\lambda_n$** We initialize $\lambda_n$ in a similar fashion as $H_n$. We consider the same signal, $\bar{X}$, obtained by sliding a window of size $L$ over the read depth signal $X$ and summing up the consecutive values within the window. Whenever there is a peak in the signal $\bar{X}$, we initialize $\lambda_n$ with the corresponding value $\bar{x}_n$ and the rest of the remaining are initialized by zero.
Algorithm 2: Computation of the initial values $\hat{\lambda}_n$ for the random variable $\lambda_n$.

1. Input: $X, L$;
2. Construct $\bar{X}$ by sliding a window of length $L$ over $X$ and summing up the contained values;
3. **for each element** $\bar{x}_n$ of $\bar{X}$ **do**
   - *if* $\bar{x}_n$ is a local peak in $\bar{X}$ **then** $\hat{\lambda}_n \leftarrow \bar{x}_n$;
   - *else* $\hat{\lambda}_n \leftarrow 0$;
**end**
4. Return $\lambda_n$;

**Initial values for** $p_i$ The same constructed signal $\bar{X}$ is used to find initial values for the probabilities $p_i$. Whenever we have a local peak in the signal $\bar{X}$, we divide the neighboring values from $X$ by their corresponding peak in $\bar{X}$ within a region of size $L$ centered in the respective peak. The median of the obtained values over all the peaks for every position in window $L$ gives the initial values for the correspondent $p_i$, $1 \leq i \leq L$.

Algorithm 3: Computation of the initial values $\hat{p}_i$ for the probabilities $p_i$.

1. Input: $X, L$;
2. Construct $\bar{X}$ by sliding a window of length $L$ over $X$ and summing up the contained values;
3. **for each element** $\bar{x}_n$ of $\bar{X}$ **do**
   - if $\bar{x}_n$ is a local peak in $\bar{X}$ **then**
     - for $i = 1$ to $L$ **do**
       - $P_i = P_i \cup \frac{x_n}{\bar{x}_n}$
     **end**
**end**
4. for $i = 1$ to $L$ **do**
   - $\hat{p}_i = \text{median}(P_i)$
**end**
5. Return $\hat{p}_i$;
6 Model testing

In this chapter, we apply the proposed model from Chapter 5 to a synthetic and real data set.

The model is first tested with synthetic data in order to provide a preliminary image about its behavior and accuracy. We evaluate the performance of our model on synthetic data by studying the correlation between the real values used to simulate the data and the corresponding predicted posterior mean values for each $\lambda_n$ and $H_n$. Our test cases include also an analysis of the improvement on convergence speed when the model is provided with the initial values versus the case when the model is initialized with random variables.

We apply our model likewise to the PAR-CLIP data from the MOV10 protein published in [42] and we aim at detecting the starts of putative binding sites.

6.1 Synthetic experiments

6.1.1 Synthetic data generation method

The formal definition of the model from the previous chapter was used to generate synthetic data sets with different characteristics.

In order to provide realistic values for the probabilities $p_0$ and $p_p$, we performed an analysis of Chromosome 21 from the MOV10 data-set. Accordingly, the probability of having a $T \rightarrow C$ mismatch by chance has been estimated as follows:

$$\hat{p}_0 = \frac{\sum_{s \in S} n_s}{3 \sum_{s \notin S_T} n_s}$$

where $s \in S$ represents any kind of mismatch (for example $A \rightarrow G$) and $S_T$ denotes the set of all the possible mismatches that do not constitute transitions from nucleotide $T$ i.e. $S_T = S \setminus \{T \rightarrow C, T \rightarrow G, T \rightarrow A\}$. The number $n_s$ is equal to the total number of substitutions $s$ encountered within the chromosome.

The probability $p_p$ of having an experimentally induced $T \rightarrow C$ substitution was estimated by:

$$\hat{p}_p = \frac{n_{T\rightarrow C}}{n_{T\rightarrow X}} - p_0$$

where $n_{T\rightarrow C}$ is the total number of $T \rightarrow C$ transitions and $n_{T\rightarrow X}$ is the total number of substitutions represented by any transition from nucleotide $T$.

Using the information from Chromosome 21, we have obtained the following estimates:

$$\hat{p}_0 = 0.014$$
$$\hat{p}_p = 0.625$$

Synthetic data sets of different length were generated and the model was applied on
them using uninformative priors on the parameters:

\[ \alpha_l = 1 \]
\[ \beta_l \sim \text{Gamma}(0.1, 0.1) \]
\[ \alpha_0 = \beta_0 = \alpha_{p0} = \beta_{p0} = \alpha_{pp} = \beta_{pp} = 0.1 \]  

(27)

To give a realistic value for \( p_n \), the parameter of the Bernoulli distribution of \( H_n \), we have considered the ratio of the number of peaks in the observed data \( X \):

\[ \hat{p}_n = \frac{n_{\text{peaks}}}{|X|} \]  

(28)

6.1.2 Analysis of the model convergence

As mentioned in the previous section, the convergence speed of the model can be considerably improved by initializing the sampler with appropriate values. We illustrate this behavior in the following analysis.

We have generated a synthetic data set of length 1000 with a signal to noise ratio (SNR) of 5.85 and a peak expansion of \( L = 3 \). We have applied the coupled form of the proposed model within three different sampling initialization settings: initialization with random initial values (model \( M_0 \)), initialization with initial values only for \( H_n \) which were computed as described in Section 5.3.2 (model \( M_1 \)) and initialization with initial values for \( H_n \), \( p_i \) and \( \lambda_l \) obtained following the algorithms in the same section (model \( M_2 \)). The models were run each using three MCMC chains per model. To decide whether the sampling has converged, the evolution of the data log likelihood was monitored throughout the sampling process as well as the steadiness of the GR statistic below the threshold of 1.1. The auto-correlation between the samples was also observed.

Figure 17 displays the convergence results for the three models \( M_0 \) (figure 17a), \( M_1 \) (figure 17c) and \( M_2 \) (figure 17b). The model \( M_0 \) is clearly the slowest to converge, while the other two models converge within similar amounts of iterations. Thus, after less than 10,000 iterations, both models \( M_1 \) and \( M_2 \) have reached convergence while model \( M_0 \) is still on its way to convergence even after 60,000 iterations. The model \( M_0 \) has finally converged after 130,000 iterations corresponding to three hours of waiting time while the other two models took less than a quarter of an hour to converge.

In order to extract the predicted binding sites, we summarize the posterior distributions of the latent variables \( H_n \) and \( \lambda_n \) using the mean estimate. The positions where the mean estimate for \( H_n \) is equal to 1 are considered predicted starting sites and the corresponding height is given by the mean estimate of \( \lambda_n \) at the respective positions.

To check if the three models have converged towards the same steady state, we have analyzed the correlation between the predicted mean posterior values for the latent expression of the starting sites (\( \lambda_n \)). Figure 18 displays the ratios of the predicted mean posterior values for the heights of the starting sites between models \( M_0 \) and \( M_1 \) respectively \( M_2 \).
Figure 17: Distribution of the log likelihood of the proposed model applied on a data set of length 1000 with three different initialization settings for the MCMC chains. M0 denotes the model with random initial values, M1 represents the model with astute pre-calculated values for $H_n$ while M2 is the model with pre-calculated values for $H_n$, $\lambda_n$ and $p_i$. Model M0 converges at a considerably slower rate than the other two models.

This analysis shows that the initialization of the model with initial values calculated in Section 5.3.2 considerably increases convergence speed and leads to the same inference result. Furthermore, not all the three variables $H_n$, $\lambda_n$ and $p_i$ need to be initialized. Initializing $H_n$ is sufficient in order to obtain the same result with the same speed.
Figure 18: Study of the correlation between the steady states of the models $M_0$, $M_1$ and $M_2$. The figure presents the distribution for the ratios of the predicted mean posterior values for the amplitude $\lambda_n$ of the predicted start sites between models $M_0$ and $M_1$ respectively $M_2$. All the values are very close to 1 which suggests a very high agreement.

The following tests on synthetic data were performed by using the same uninformative priors (specified by the equality 27) and with pre-calculated initial values for $H_n$. For sampling, we used only one MCMC chain and a burn-in period of 50,000 samples. The burn-in period is defined as the number of iterations needed until convergence. After the burn-in, 5000 samples were used to estimate the posterior predicted values.

6.1.3 Testing results and analysis

Synthetic data experiments were performed in order to test the coupled as well as the decoupled form of the proposed model.

One preliminary synthetic data-sets of length 1000 with $L = 3$ was generated in order to test the coupled form of the proposed model. During the sampling process, it took 87s on one 2.6 GHz Intel i7 core to generate 1000 samples from the corresponding model with one chain. The posterior distributions for the model parameters were analyzed. The positions where the mean estimate for $H_n$ was equal to 1 were considered predicted starting sites.

Figure 19 displays the preliminary synthetic experiment for testing the coupled form of the model. Both the observed read depth and $T \rightarrow C$ depth are represented in figure 19a. The generated synthetic data set has a high background noise as we wanted to test the ability of the model to deal with a high noise level. The comparison between predicted and latent starting sites is displayed in figure 19b and shows that the model accurately detects the starting points.
Figure 19: Synthetic testing experiment for the coupled form of the model using a generated sequence of length 1000 and a peak expansion length $L = 3$. Figure 19a contains the generated synthetic datasets for the read depth and the $T \rightarrow C$ depth while figure 19b presents the absolute comparison of the predicted and real starting points.

Figure 20: Posterior distribution of the parameters $p_i$, $\lambda_0$, $p_0$, and $p_p$. The black lines indicate the true values used for the generation of the synthetic data set.
Figure 20 presents the posterior distributions of the parameters of the coupled model compared to the real values used for generating the synthetic data set. With the exception of $p_0$ which is over-estimated, the rest of the parameters are accurately inferred.

We performed a synthetic test experiment also on the decoupled model (figure 21). Like the coupled model, the decoupled model performed well on the synthetic data set, detecting the latent binding sites.

(a) Synthetic data set

(b) Comparison between latent and predicted start heights

Figure 21: Synthetic testing experiment using a generated sequence of length 1000 and a peak expansion length $L = 3$ with the decoupled model. Figure 21a contains the generated synthetic datasets for the read depth and the T→C depth while figure 21b presents the absolute comparison of the predicted and real starting points.

We were also interested in studying the error on the predicted amplitude of the starting sites. In order to do so, we performed 70 synthetic tests on the coupled form of the model by using different data-sets of length 1000 and an expansion length $L = 3$.

For each individual test result, we have applied the following procedure. First, we have calculated, for every detected start site, the difference between the predicted start height and the real start height. The standard error of the model was defined as the relative percentage error to the real start height:

$$e_n = \frac{|\hat{\lambda}_n - \lambda_n|}{\lambda_n} \times 100$$  \hspace{1cm} (29)
where $|\hat{\lambda}_n - \lambda_n|$ is the absolute value of the difference between the predicted ($\hat{\lambda}_n$) and real ($\lambda_n$) amplitude of the start sites. Afterwards, we have computed the median over all the obtained $e_n$ which would give a median percentage error for each data set. The 70 obtained percentage errors were at least 3%, did not exceed 7.5% and averaged around 5% (figure 22).

![Figure 22](image)

**Figure 22:** Distribution of the error obtained by comparing the start heights of the peaks over 70 different data sets. The error for a data set was represented by the median value over the relative differences between the predicted start heights versus and real values.

### 6.2 Real data experiment

#### 6.2.1 Trimming method

Given that the *OpenBUGS* platform does not support large scale input sequences, we had to perform an additional trimming operation in order to reduce the size of our input signals.

We performed the trimming operation by discarding the regions where the read depth signal was inferior to a fixed threshold and introduced 0-buffers of length $L$ instead of the trimmed regions. We chose a threshold of 20 and we used the removed signal values to update noise priors.

We recall the prior distributions for the background noise, $\lambda_0$, and for the probability of having a $T \rightarrow C$ transition by chance, $p_0$:

\[
\lambda_0 \sim \text{Gamma}(\alpha_0, \beta_0) \\
p_0 \sim \text{Beta}(\alpha_{p0}, \beta_{p0})
\]  

(30)

Let $\bar{X}$ be the set of discarded elements from the read depth signal $X$ and $\bar{Y}$ the set of
discarded elements from the $T \rightarrow C$ depth signal.

The interpretation of the hyperparameters for the $\text{Gamma}$ distribution is $\alpha_0$ total occurrences in $\beta_0$ intervals. Since we model the background read depth noise with a $\text{Gamma}$ distribution, the total number of occurrences would correspond to the total number of reads that start within the discarded regions while the number of intervals corresponds to the total number of discarded positions:

$$\alpha_0 = \sum_{x_i \in \bar{X}} x_i$$
$$\beta_0 = |\bar{X}|$$

Furthermore, we model the probability of having a $T \rightarrow C$ transition by chance with a $\text{Beta}$ distribution whose hyperparameters stand for the number of successes and respectively failures. The successes in this case would be the starting reads within the discarded regions that contain at least one $T \rightarrow C$ transition, while the failures would be the starting reads in the discarded regions that do not encompass any $T \rightarrow C$ transition:

$$\alpha_{p0} = \sum_{y_i \in \bar{Y}} y_i$$
$$\beta_{p0} = \sum_{x_i \in \bar{X}} x_i - \sum_{y_i \in \bar{Y}} y_i$$

### 6.2.2 Putative binding sites on chromosome 21

We applied the coupled and decoupled forms of our model on the Chromosome 21 from the MOV10 dataset published in [42] and pre-processed using our proposed pipeline. After trimming with a threshold fixed to 20, the length of the input signals was 3355.

The coupled form of our model proved to be too sensitive and flexible and concluded in predicting a binding site at almost every position in the input sequence (figure 23a). One reason for this behavior is that there is a high amount of $T \rightarrow C$ substitutions in the real data and, furthermore, the $T \rightarrow C$ depth is often equivalent to the read depth. As a result, the proposed noise model is not able to describe the high $T \rightarrow C$ relative to the read depth and considers them as hits.

The decoupled form proved to deal better with the prevalence of $T \rightarrow C$ in the data (figure 23b) and returned 17 putative starting sites (the full list is provided in Appendix). Out of these 17 hits, 8 were located near annotated genomic elements (table 3). We note several miRNAs but also two protein coding genes - $DOPEY2$ and $MRPS6$. 

42
Figure 23: Extract of the performance of both coupled and decoupled forms of the model applied to chromosome 21. The coupled model is highly sensitive to the large amount of $T \rightarrow C$ transitions in the data and predicts starting sites at almost every position in the input sequence. On the contrary, the decoupled model predicts binding sites with the same peak shape and high $T \rightarrow C$ substitution events.

<table>
<thead>
<tr>
<th>predicted position</th>
<th>Id</th>
<th>type</th>
<th>real location</th>
</tr>
</thead>
<tbody>
<tr>
<td>9,827,044</td>
<td>ENSG00000264462</td>
<td>miRNA</td>
<td>9,825,832 - 9,826,011</td>
</tr>
<tr>
<td>9,827,521</td>
<td>ENSG00000264063</td>
<td>miRNA</td>
<td>9,826,203 - 9,826,263</td>
</tr>
<tr>
<td>24,437,789</td>
<td>ENSG00000230972</td>
<td>processed transcript</td>
<td>24,437,813 - 24,503,528</td>
</tr>
<tr>
<td>26,946,293</td>
<td>ENSG00000207795</td>
<td>miRNA</td>
<td>26,946,292 - 26,946,356</td>
</tr>
<tr>
<td>35,345,824</td>
<td>ENSG00000237945</td>
<td>lincRNA 649</td>
<td>35,287,838 - 35,349,992</td>
</tr>
<tr>
<td>37,557,594</td>
<td>ENSG00000142197</td>
<td>protein $MRPS6$</td>
<td>37,529,080 - 37,666,572</td>
</tr>
<tr>
<td>44,201,384</td>
<td>ENSG00000233754</td>
<td>lincRNA</td>
<td>44,201,184 - 44,202,343</td>
</tr>
</tbody>
</table>

Table 3: List of putative binding sites predicted using the decoupled version of the proposed model.
7 Discussion

In the present Master Thesis, we explored various possibilities of statistically analyzing PAR-CLIP data for detecting RNA-protein binding sites. PAR-CLIP is an experimental protocol for detecting the binding sites of RBP, with the characteristic of introducing $T \rightarrow C$ substitutions at the binding sites. First, we introduced a general pipeline for pre-processing the data and extracting relevant information useful for further modeling. Second, we proposed a statistical model using Bayesian inference in order to detect the starts of the RBP binding sites. For pre-processing as well as binding site detection, we used the dataset of the MOV10 protein published in [42].

According to the proposed pre-processing pipeline, the raw next generation sequencing reads were clipped of their adapter and aligned to the reference genome. The alignment result was included into a read summary which allowed the construction of signals such as read depth or $T \rightarrow C$ depth used for modeling. The constructed read summary also provided a preliminary view of the data through the mismatch distribution within the aligned reads. For the MOV10 dataset, the mismatch profiles revealed a strong $T \rightarrow C$ prevalence in regions with a mid-range mismatch frequency.

We proposed a statistical model to detect the starts of RBP binding sites which used two sources of evidence: the read depth and the $T \rightarrow C$ depth, both functions of every genomic position. The read depth represents the number of reads that start at each position, while the $T \rightarrow C$ depth represents the amount of reads that start at each position and encompass at least one $T \rightarrow C$ transition within their length. The model was defined such that it was possible to couple or decouple the two observed signals.

To estimate the latent variables of the model, we used Markov Chain Monte Carlo sampling through the OpenBUGS software platform. To improve the convergence speed, we initialized our sampler with pre-calculated values for certain latent variables. We tested our implementation on several synthetic data sets before applying the model on the MOV10 data set.

We applied the coupled and decoupled forms of the model using the read depth and $T \rightarrow C$ depth signals for chromosome 21 from the MOV10 data set. Due to the limited computational capacity of the OpenBUGS software, we performed a trimming operation on the signals prior inference. The coupled form of the model proved to be too sensitive and did not produce reasonable results notably due to the high amount of $T \rightarrow C$ transitions present in the data. On the contrary, the decoupled model was able to overcome this effect and predicted 17 binding sites. Out of these, 8 proved to be in the vicinity of annotated genomic elements, mainly non-coding RNAs.

The model implementation exposed in the current work represents a proof of concept and a good starting point for developing an elaborate method for detecting RBP binding sites. The OpenBUGS platform is a rapid and straightforward alternative to the implementation of a custom sampler. However, in the future, in order to be able to optimize and control the sampling method as well as to deal with larger input sequences, the implementation of an in-house inference scheme would be required. Another aspect that can be further extended would be to design a method to handle the large regions of zero
in the observed signals either by using a zero-inflated noise model or a more rigorous way to choose the noise threshold. One alternative would also be to perform the alignment over the exome instead of the reference genome. This would reduce considerably the null regions within the observed signals.

The proposed model can be further extended in order to detect complete binding sites and not just their starting points. One method to detect the whole binding sites would be to apply the same model using the same observed signals but this time concerning the ends of the aligned reads. Since we would expect the same expansion effect at both ends of a binding site, this would result in a similar behavior for the ends of the reads at the terminus of a binding site like for the starts of the reads at the beginning of a binding site. By combining the starts and the ends, one can then have an idea about the complete binding sites of the protein of interest.

Finally, the present Master Thesis focuses on the statistical modeling of the binding sites. Further work is required in order to analyze and validate the obtained results on the MOV10 data set. One possibility would be through replicate experiments and comparison of the found binding sites throughout the different data sets. In the case of binding sites found within miRNAs, the existing open access databases such as miRBase\(^3\) can be used in order to check whether the protein of interest is within the predicted miRNA targets.

\(^3\)http://www.mirbase.org
Appendix

List of inferred binding sites positions for chromosome 21:

9827044
9827421
9827423
9827428
9827429
9827430
9827521
17363276
23993327
24437789
26231772
26231775
26946293
35345824
35476493
37557594
44201384
References


49


