



## Interpreting regulatory variants with predictive models

Jun Cheng

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

**Doktor der Naturwissenschaften (Dr. rer. nat.)**

genehmigten Dissertation.

**Vorsitzender:**

Prof. Dr. Björn Menze

**Prüfende der Dissertation:**

1. Prof. Dr. Julien Gagneur
2. Prof. Dr. Dr. Fabian Theis

Die Dissertation wurde am 03.07.2019 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 05.11.2019 angenommen.



# Acknowledgments

First and foremost, I would like to thank my supervisor Julien Gagneur for his outstanding support and guidance. He gave many great advice and was always ready to help. Moreover, he was patient and understanding for my many concerns. It was four wonderful years, both because of him and the lab he established.

I would like to thank other mentors of my scientific career:

- Maria von Korff for bringing me into science when I worked as a student helper in her lab. Without her I could have given up science.
- Anshul Kundaje for co-supervising me for a splicing project and for hosting me in his lab early 2019. I learned a lot during this time.
- My thesis advisory committee members Fabian Theis and Elena Conti for their valuable inputs.

I am grateful to current and former members of Gagneur lab. They are colleagues and also friends. Especially, I want to thank Žiga Avsec for collaboration and many discussions, Vicente Yopez for being an awesome office mate and organizing parties, Christian Mertes for keeping the lab cluster running, Leonhard Wachutka for showing the world, Georg Stricker for explaining the world, Daniel Bader for keeping things tidy, Basak Eraslan for encouragement, Michaela Müller for helping with translating, Thi Yen Duong Nguyen for starting the splicing project, Xueqi Cao for reading the thesis. I also thank students I worked with for their help: Falko Späh, Veronika Kotova, and particularly Muhammed Hasan Çelik. Additionally, I thank the members of Kundaje lab for a wonderful time together during my stay.

I would also like to thank my other collaborators:

- Kerstin Maier for performing the validation experiment
- Fabien Bonneau for discussions and following up with the validation experiment
- CAGI organizers and data providers for making the great competition, which truly helped my research

I also thank QBM graduate school for financial support and events. I thank QBM staff Mara and Markus for being supportive.

I can never thank my parents and friends enough. Their unconditional love and consolation made my life much more enjoyable.



# Abstract

Many human diseases are caused by genetic variants. Precision medicine requires understanding the genetic basis of diseases. With the advances of the next-generation sequencing techniques, the whole genome can be sequenced at a low cost. Large cohorts of individuals are now sequenced, and millions of variants have been identified. However, it remains challenging to interpret most of them. Even though statistical methods have been developed to associate variants with different phenotypes including diseases, precisely locating the causal variant is difficult.

On the other hand, tremendous high throughput genomics data are now publicly available; new assays keep being developed to better probe functions of genetic sequences. These data systematically covers almost all crucial biological processes, like protein DNA interaction, protein RNA interaction, splicing, and RNA degradation. They provide unique opportunities to train machine learning models to predict the functional impact of variants on specific biological processes and also overall variant pathogenicity.

Here, I developed machine learning models predicting two major gene expression steps solely from sequences: RNA splicing and degradation. Specifically, this thesis contributes to variant interpretation in threefold: First, I systematically investigated sequence elements regulating mRNA stability in the model organism *Saccharomyces cerevisiae*, for which we have high-quality genome-wide RNA half-life measured. A model integrating all sequence elements can explain 59% of mRNA half-life variation across genes. The analysis quantified the major role of codon usage in determining mRNA stability and revealed a new destabilizing motif ATATTC. Variants on 3' UTR motifs and upstream AUG codon (uAUG) have the largest effect on mRNA stability. Furthermore, the corresponding RNA degradation pathways through which different sequence elements affect mRNA stability were characterized.

Second, I developed MMSplice, a modular deep learning framework to predict effect of genetic variants on splicing in human cells. MMSplice outperformed state-of-the-art models and was the winning model of the 5th Critical Assessment of Genome Interpretation (CAGI) exon-skipping competition. MMSplice consists of modular models to score splicing-relevant regions. The framework can score variant effect on different splicing patterns, including exon skipping, alternative splice sites as well as intron retention. Moreover, MMSplice improved the prediction accuracy of pathogenicity of variants located near splice sites.

Third, I implemented MMSplice as a python package that can be directly applied to score variants from a Variant Call Format (VCF) file, enabling it to be easily incorporated into variant interpreting pipelines along with other tools. Moreover, I helped develop Kipoi, a platform to deposit and reuse predictive models in genomics.

In summary, the work in this thesis reveals novel biology about sequence determinants of mRNA stability, and provides resources and tools to interpret effects of variants on splicing and RNA degradation.

# Publications

## **Cis-regulatory elements explain most of the mRNA stability variation across genes in yeast**

**Jun Cheng**, Kerstin C Maier, Žiga Avsec, Petra Rus, Julien Gagneur

(2017) RNA, DOI:10.1261/rna.062224.117. Ref. [1]

**Author contribution** JC, JG conceived the project. JC carried the data analysis with help of ŽA. JC and KM designed the validation experiment. KM and PR carried the validation experiment. JC and JG wrote the manuscript with help of KM and ŽA.

## **MMSplice: modular modeling improves the predictions of genetic variant effects on splicing**

**Jun Cheng**, Thi Yen Duong Nguyen, Kamil J Cygan, Muhammed Hasan Çelik, William G Fairbrother, Žiga Avsec, Julien Gagneur

(2019) Genome Biology, DOI:10.1186/s13059-019-1653-z. Ref. [2]

**Author contribution** JC and JG designed the model, with the help of ŽA. JC implemented the software and analysed data. TYDN and ŽA contributed to developing the modules. JC and JG wrote the manuscript, with the help of ŽA, KJC, and WGF. KJC and WGF generated the MaPSy data. MHÇ wrote the VEP plugin.

## **CAGI5 splicing challenge: Improved exon skipping and intron retention predictions with MMSplice**

**Jun Cheng**, Muhammed Hasan Çelik, Thi Yen Duong Nguyen, Žiga Avsec, Julien Gagneur

(2019) Human Mutation, DOI:10.1002/humu.23788. Ref. [3]

## *Publications*

**Author contribution** JC and JG designed the model, with the help of ŽA. JC implemented the software and analysed data. TYDN and ŽA contributed to developing the modules. JC and JG wrote the manuscript. MHÇ wrote the VEP plugin.

### **The Kipoi repository accelerates community exchange and reuse of predictive models for genomics**

Žiga Avsec, Roman Kreuzhuber, Johnny Israeli, Nancy Xu, **Jun Cheng**, Avanti Shrikumar, Abhimanyu Banerjee, Daniel S Kim, Thorsten Beier, Lara Urban, Anshul Kundaje, Oliver Stegle, Julien Gagneur  
(2019) Nature Biotechnology, DOI:10.1038/s41587-019-0140-0. Ref. [4]

**Author contribution** ŽA, RK, JI, AS, AK, OS and JG conceived the Kipoi API. ŽA, RK and TB implemented the Kipoi API. ŽA and RK conceived and implemented kipoi\_veff. ŽA, RK and AS conceived and implemented kipoi-interpret. ŽA, RK and JC conceived and implemented kipoiseq. ŽA, RK, JI, NX and A.B. performed the analysis. DSK compiled the DNA accessibility dataset. ŽA, RK, JI, NX, AS and LU contributed models to the repository. AK, OS and JG designed and supervised research. ŽA, RK, AK, OS and JG wrote the manuscript.

### **Modeling positional effects of regulatory sequences with spline transformations increases prediction accuracy of deep neural networks**

Žiga Avsec, Mohammadamin Barekatin, **Jun Cheng**, Julien Gagneur  
(2019) Bioinformatics, DOI:10.1093/bioinformatics/btx727. Ref. [5]

**Author contribution** ŽA and JG conceived spline transformation. ŽA implemented spline transformation. ŽA performed the analysis with help from MB and JC. JG supervised research. ŽA and JG wrote the manuscript.



# Contents

<b>Acknowledgments</b>	<b>iii</b>
<b>Abstract</b>	<b>v</b>
<b>Publications</b>	<b>vii</b>
<b>Contents</b>	<b>ix</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Overview . . . . .	1
1.1.1 Outline . . . . .	1
1.2 Biological background . . . . .	2
1.2.1 The complex life of RNA . . . . .	2
1.2.2 RNA splicing . . . . .	2
1.2.3 RNA degradation . . . . .	5
1.2.4 RNA half-life measurement with metabolic labeling . . . . .	5
1.2.5 Splicing quantification with RNA-Seq . . . . .	7
1.2.6 Massively parallel reporter assay . . . . .	7
1.3 Variant interpretation . . . . .	9
1.3.1 Genetic variants and human diseases . . . . .	9
1.3.2 Interpreting regulatory variants . . . . .	10
1.3.3 Computational methods for variant interpretation . . . . .	11
1.3.3.1 Functional impact prediction . . . . .	11
1.3.3.2 Disease impact prediction . . . . .	13
1.4 Aims and scope of this thesis . . . . .	15
<b>2 Machine Learning background</b>	<b>17</b>
2.1 Supervised learning . . . . .	17
2.2 The learning objective . . . . .	17
2.3 Regularization . . . . .	19
2.4 Regression . . . . .	20
2.4.1 Linear regression . . . . .	20
2.4.2 Beta regression . . . . .	21
2.4.3 Linear mixed model . . . . .	22
2.5 Neural networks . . . . .	23
2.5.1 Vanilla Neural Networks . . . . .	24
2.5.2 Convolutional Neural Network . . . . .	24

## CONTENTS

2.6	Optimization . . . . .	27
<b>3</b>	<b>Discussion and Outlook</b>	<b>29</b>
3.1	Outlook . . . . .	30
3.1.1	Functional characterization for the new half-life regulating motif ATATTC . . . . .	31
3.1.2	Tissue-specific splice variant effect prediction . . . . .	31
3.1.3	Gene segmentation model for splice variant prediction . . . . .	31
3.1.4	5' capping and 3' polyadenylation prediction . . . . .	32
3.1.5	Join transcriptional and post-transcriptional signals for variant in- terpretation . . . . .	32
3.1.6	Hierarchical models for variant pathogenicity prediction . . . . .	32
<b>A</b>	<b>Appendix</b>	<b>35</b>
<b>B</b>	<b>Appendix</b>	<b>49</b>
<b>C</b>	<b>Appendix</b>	<b>67</b>
	<b>List of Figures</b>	<b>69</b>
	<b>List of Tables</b>	<b>71</b>
	<b>Bibliography</b>	<b>73</b>

# 1 Introduction

## 1.1 Overview

Many human diseases are related to genetic disorders, which are often consequences of genetic mutations. More than a hundred years ago, we realized the central role of heredity in controlling the physiology of life from Gregor Mendel's experiment [6]. Although the initial draft of the human genome was released in 2001 [7], we are far from understanding and interpreting all instructions encoded in the genome.

The human genome consists of around 3 billion base pairs, among which, only around 1% encode for proteins [7]. Even for protein-coding transcripts, the majority of the sequences are introns, which means that they do not encode proteins [8]. The rest of the sequences are responsible for regulating processes e.g. transcription factor binding, RNA splicing and degradation. Despite decades of research, we still lack a complete understanding of the protein-coding sequence, and even less for the non-coding part. Consequently, it remains difficult to interpret common-disease associated variants that fall mostly into regulatory regions [9]. Similarly, whole exome sequencing is only able to diagnose 25%-30% cases among large rare disease cohorts [10]. For the majority of cases, the causal variants are likely located in regulatory regions.

The aim of this thesis is to develop models to interpret variants from these regulatory regions. Particular, I am interested in interpreting them for their impact on two important (post-transcriptional) biological processes: 1) RNA splicing, a post-processing step to make RNA functionally mature. 2) mRNA degradation, a controlled turnover step to regulate gene expression level.

### 1.1.1 Outline

This cumulative dissertation is based on the researches published in my first author articles. In this thesis, I introduce the general motivation in section 1.1, the biological background and related work in section 1.2 and machine learning background in Chapter 2. In Chapter 3, I discuss my research in the context of current literature and provide an outlook for further studies. The full texts of my first-author articles are attached in the Appendices A-C. A detailed summary is provided in front of each corresponding article.

## 1.2 Biological background

### 1.2.1 The complex life of RNA

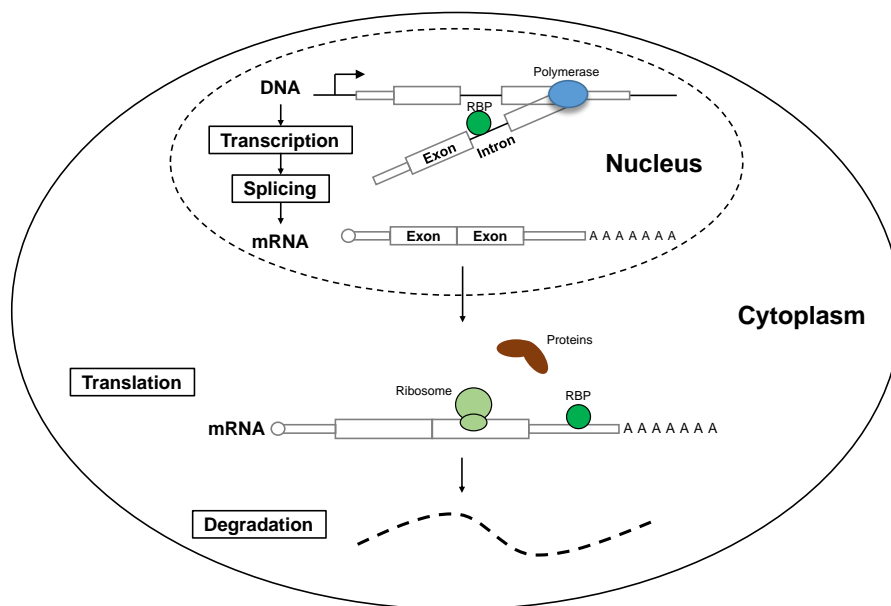
DNA is the central molecule where the genetic information is stored. Eukaryotic DNAs are densely packed in chromosomes. Humans have 23 pairs of chromosomes, among which 22 pairs are autosomes, and one pair are sex chromosomes. Genes are fragments from the DNA molecules that encode function information. Some genes are protein-coding, which means they can be translated into proteins. Humans have approximately 20,000 protein-coding genes [8]. Life of RNA starts from transcription, which is a process that transmits information from the DNA template to precursor RNA molecules (Figure 1.1). Precursor RNAs need to be further processed to be mature and functional. These processes include 5' capping, 3' polyadenylation, and splicing. After being processed, messenger RNA (mRNA), which encode for proteins, are transported from the nucleus to the cytoplasm. In the cytoplasm, information on mRNA is read by ribosomes and translated to proteins by assembling amino acids in the order stored in the mRNA molecule. Translation starts from an AUG codon (start codon) and ends when one of the UAG, UAA, and UGA stop codons is encountered. The untranslated regions before the start codon and after the stop codon on the transcripts are referred to as the 5' UTR and 3' UTR respectively. RNAs are unstable and are degraded in a controlled manner. The concentration of mRNA in the cell is determined jointly by the production (transcription) rate and the degradation rate.

RNA molecules are bound by proteins that regulate their function. These RNA binding proteins (RBPs) bind to RNA by physically interacting with specific nucleotide sequences (motifs). RBPs are the key elements regulating RNA dynamics, including 5' capping, 3' polyadenylation, RNA editing, splicing, and degradation. Since these regulations happen after RNA transcription, they are termed as *post-transcriptional regulation*. RBPs have binding preferences to specific RNA sequence motifs. Instead of directly studying the interactions between RBPs and RNAs, which are difficult to be measured experimentally, RNA sequence elements are often considered. These elements are referred to as cis-regulatory elements (CREs) or regulatory code in this thesis.

In the next two sections, I will briefly introduce the biological background of splicing and RNA degradation, which are the focuses of this thesis.

### 1.2.2 RNA splicing

Eukaryotic genes are discontinuous, with short coding sequences being interrupted by stretches of long non-coding sequences. The process of cutting out parts of the transcribed transcripts (introns) and concatenate the remaining regions (exons) is termed as splicing (Figure 1.2). Most genes in higher eukaryotes are spliced [8]. Splicing is a series of biochemical reactions. The first step is the branchpoint adenosine attack the 5' splice site (donor), resulting in an intron lariat. The second step is cutting the 3' splice site (acceptor) with mediation from the corresponding 5' splice site, leading into the removal of the intron lariat and concatenation of the 5' and 3' splice sites [11] (Figure 1.2). The removed intron lariat is unstable and is quickly degraded.



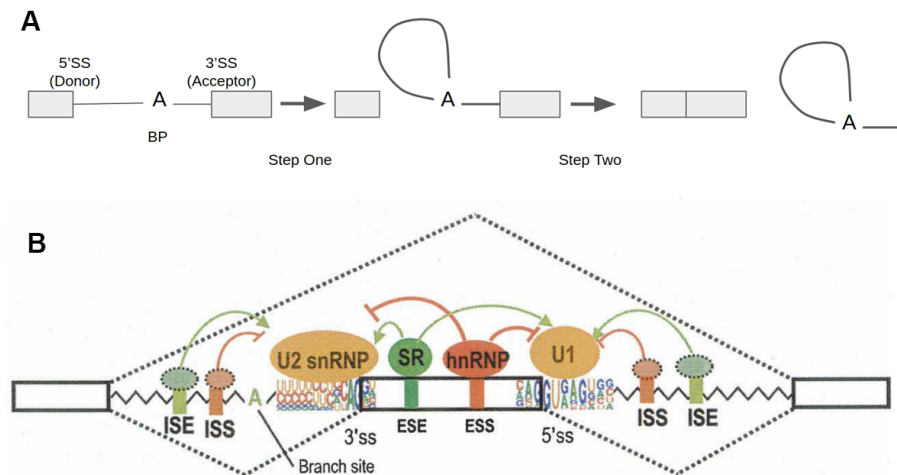
**Figure 1.1: The central dogma of biology (nuclear-encoded genes).** Genetic information is encoded in the DNA. RNA copies information from DNA through transcription. Mature RNA is produced from premature RNA by splicing out introns. Both transcription and splicing take place in the nucleus. Proteins are synthesized in the cytoplasm with messenger RNA (mRNA) as the template (translation). In the end, mRNA is degraded in the cytoplasm controlled by RNA degradation pathways (RNA degradation).

Alternative splicing refers to alternative ways of concatenating between donor splice sites and acceptor splice sites. As a consequence, different transcripts are created from a single gene. Proteins produced from these transcripts can show very different biochemical properties, e.g., solubility, membrane binding preference [12]. Approximately 92-94% of human genes are alternative spliced [13]. Common alternative splicing patterns include exon skipping (cassette exon), alternative 5' splice sites, alternative 3' splice sites, mutually exclusive exons, and intron retention (Figure 1.3). The most common type of alternative splicing pattern is exon skipping.

Alternative splicing adds high complexity on top of the genome [14]. For example, the *Drosophila* gene *Dscam* can produce as many as 38,016 different protein isoforms [15]. It is generally surprising that humans do not have significantly more protein-coding genes compared to other species; neither does the genome size is considerably larger. It is suggested that alternative splicing is one of the mechanisms used to achieve higher cellular complexity [16].

Given the importance of splicing, it is tightly regulated in different tissues and developmental stages. Several sequence elements are essential for splicing regulation through binding with splicing regulatory proteins. First of all, the sequence context around the

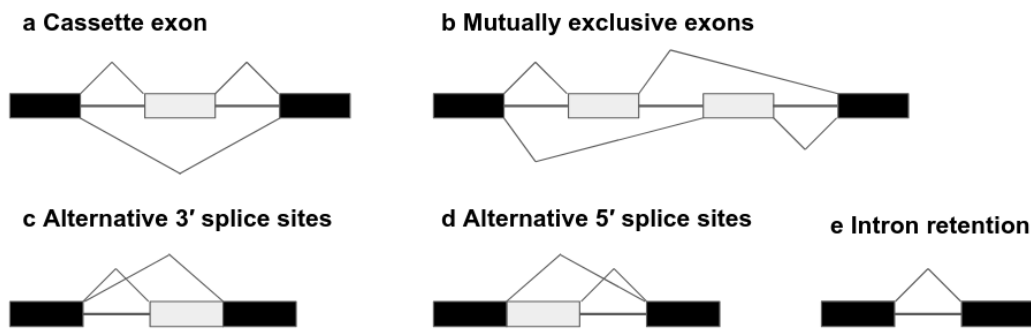
## 1 Introduction



**Figure 1.2: RNA splicing regulation.** Figure panel B is taken from [17]. A). Two-step procedure of RNA splicing. B). RNA splicing regulatory elements. Exons are shown as boxes, introns are shown as jagged lines. The majority of ( $\sim 95.5\%$ ) introns are recognized and spliced by the U2-dependent major spliceosome, while the remaining ones are by the U12-dependent minor spliceosome [18]. The consensus motifs of the 5' and 3' splice sites for the exon in the middle are shown as sequence logos. The recognition of the splice sites and branchpoint is a crucial step to initialize splicing. The recognition is modulated by exonic (ESE, ESS) and intronic (ISS, ISE) cis-elements. These cis-elements present with specific sequence patterns and are specifically recognized by their trans-acting partners, which are RNA binding proteins (SR proteins, hnRNP, etc).

exon-intron boundaries is important for the splicing machinery to recognize the splice sites correctly. Second, the branchpoint sequence context and position is critical for the splicing lariat formation. Third, many sequence motifs in the exon as well as in the intron also play an important roles. Some positively regulate splicing and are therefore named as exonic splice enhancers (ESE) or intronic splice enhancers (ISE), depending on their position. Likewise, negative regulators are called as exonic splice silencers (ESS) or intronic splice silencers (ISS) [19].

Abnormalities on splicing can lead to severe consequences. Abnormally spliced transcripts either encode completely different genetic information or quickly degraded by RNA quality control mechanisms. Both consequences lead to functional loss of the gene. Therefore, genetic mutations disrupting splicing can lead to a wide range of human diseases [11]. Typical disease-causing abnormal splicing events include cryptic splice site creation or activation [20], exon skipping [21] and intron retention [22].



**Figure 1.3: Common patterns of alternative splicing.** Gray boxes represent alternative exons, black boxes represent constitutive exons.

### 1.2.3 RNA degradation

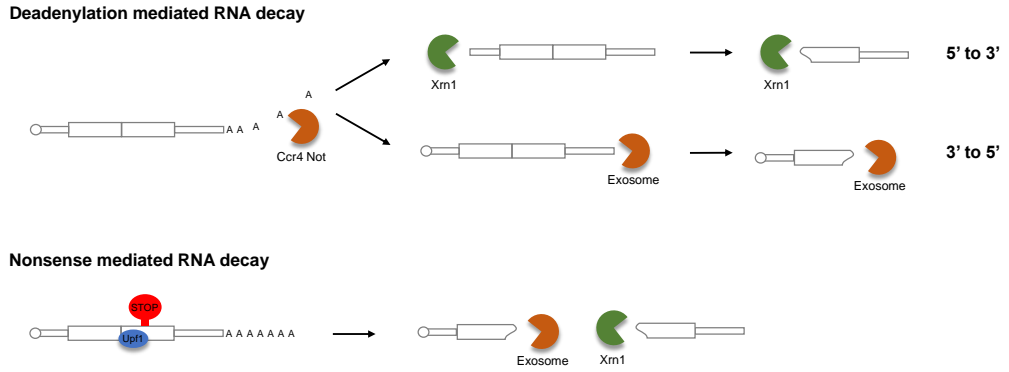
The degradation of RNA is an intensively regulated process. Different RNAs have different degradation rates (or half-life), which mainly depends on its sequence. Regulating RNA degradation rate is important in at least three folds: First, the degradation rate and the transcription rate of the transcript jointly determine its concentration in the cell. Second, several RNA quality control mechanisms clear out aberrant transcripts, which may otherwise be harmful to the cell. Third, the dynamics of RNA degradation rate enable the cell to adapt to the environmental stimulates quickly.

This thesis focuses on RNA degradation in yeast (*Saccharomyces cerevisiae*). Generally, RNA in yeast is degraded from 5' to 3' by Xrn1 or from 3' to 5' by exosome [23]. Both pathways start with removing the poly(A) tail (deadenylation) (Figure 1.4). Specialized RNA turnover pathways are triggered when abnormality of RNA is detected. One important RNA quality control mechanism is the nonsense-mediated decay (NMD) pathway (Figure 1.4). NMD primarily target transcripts with premature termination codons (PTC) to prevent the production of abnormal proteins [23]. Transcripts with PTC can be the result of abnormal splicing, translating from upstream AUG instead of the canonical AUG start codon or genetic mutations creating/disrupting canonical stop codons.

### 1.2.4 RNA half-life measurement with metabolic labeling

RNA half-life, as well as the splicing rate, can be estimated by metabolic labeling the nascent synthesized RNAs in the living cell [24, 25]. A commonly used labeling chemical is the 4-thiouracil (4-tU), which is a natural nucleotide analog of uracil. Briefly, 4-tU can be taken by the cell and incorporated into newly synthesized RNAs when added to the cell culture medium. In this way, one can measure how many new RNAs are synthesized in a given short time.

## 1 Introduction



**Figure 1.4: mRNA degradation pathways.** Typically, mRNA is degraded in the cytoplasm from 5' to 3' by Xrn1 or from 3' to 5' by exosome. Both pathways start by removing the polyA tail with Ccr4-Not protein complex.

Let the synthesis rate of RNA be  $\mu$ , splicing rate be  $\sigma$ , and the degradation rate be  $\lambda$ . We can model the amount of precursor RNA and mature RNA at time point  $t$  with the following first order ordinary differential equation (ODE) [26]:

$$\begin{aligned} \frac{d[\text{precursor RNA}]}{dt} &= \mu - \sigma[\text{precursor RNA}] \\ \frac{d[\text{mature RNA}]}{dt} &= \sigma[\text{precursor RNA}] - \lambda[\text{mature RNA}] \end{aligned} \quad (1.1)$$

The ODE system has the following initial condition at  $t = 0$  when 4-tU labeling starts:

$$\begin{aligned} [\text{precursor RNA}]_{\text{labelled}}|_{t=0} &= 0 \\ [\text{mature RNA}]_{\text{labelled}}|_{t=0} &= 0 \\ [\text{precursor RNA}]_{\text{unlabelled}}|_{t=0} &= \frac{\mu}{\sigma} \\ [\text{mature RNA}]_{\text{unlabelled}}|_{t=0} &= \frac{\mu}{\lambda} \end{aligned} \quad (1.2)$$

Under a short time scale, we can approximate the analytical solution of the ODE system with Taylor expansion [26]:

$$\begin{aligned} \mu &= \frac{[\text{precursor RNA}]_{\text{labelled}}(t)}{t} \\ \sigma &= \frac{[\text{precursor RNA}]_{\text{labelled}}(t)}{t[\text{precursor RNA}]_{\text{total}}} \\ \lambda &= \frac{[\text{precursor RNA}]_{\text{labelled}}(t)}{t[\text{mature RNA}]_{\text{total}}} \end{aligned} \quad (1.3)$$



This technique has been recently improved both experimentally (TT-Seq) and computationally [27, 26]. Genome-wide half-life data for human is now available, one can potentially understand mRNA kinetics less biased with these techniques.

Another commonly used technique for RNA half-life estimation is based on transcriptional arrest [28]. In this protocol, transcription is stopped by inactivating RNA polymerase II [29]. RNA materials are harvested and quantified after the inactivation with a time course. A first order ODE model can be fitted to this data to calculate RNA half-life genome-wide.

### 1.2.5 Splicing quantification with RNA-Seq

RNA-Seq is a next-generation sequencing technology that sequences the whole transcriptome in high throughput. It can not only quantify the expression level for all transcripts but also provide rich resources to study RNA splicing, allele-specific expression, and so on.

Alternative splicing level of exons, which is quantified as percentage spliced-in ( $\Psi$ ), can be calculated from RNA-Seq reads. For exon skipping events,  $\Psi$  can be calculated as the number of reads supporting the inclusion divided by the sum of inclusion reads and skipping reads (Figure 1.5) ([30]). The splicing ratio for alternative 5' splicing ( $\Psi_3$ ) for a given junction is calculated by dividing the junction supporting reads by all spliced reads starting from the same donor position (Figure 1.5). Similarly, the splicing ratio for alternative 3' splicing ( $\Psi_5$ ) can be calculated [30].

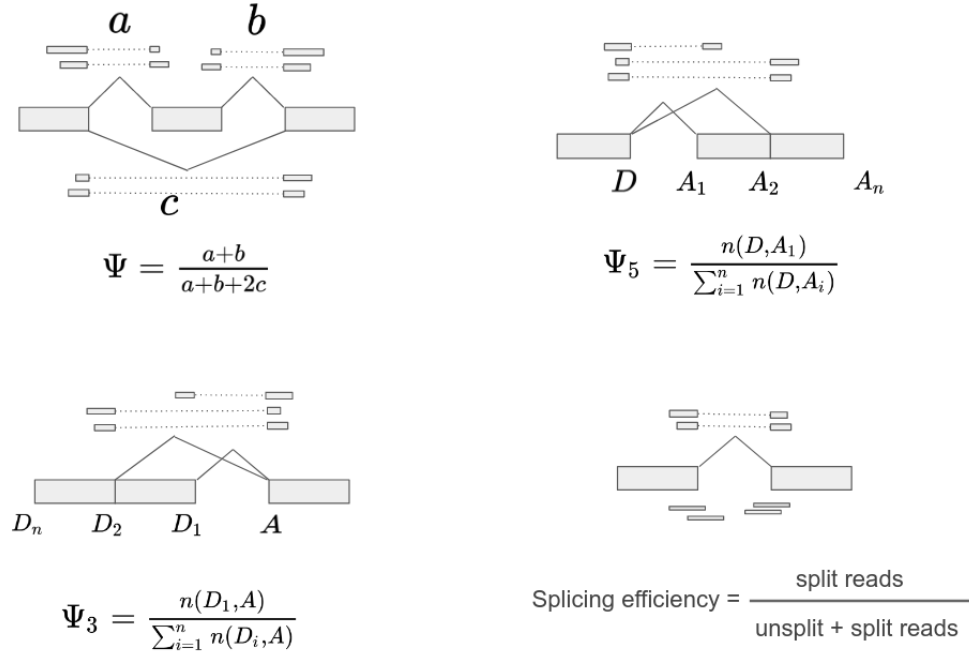
However, RNA-Seq read counts subject to many biases, for instance, sequence GC content [31, 32]. Furthermore, counts ratio is less reliable if the number of supporting reads is small. Several methods have been developed to normalize out the bias and achieved better  $\Psi$  estimation [33, 34, 35, 36, 37]. It is recommendable to use these tools in practice to estimate the splicing level.

### 1.2.6 Massively parallel reporter assay

Perturbing genomic sequence in living cells is a critical approach to study its function. Scientists have designed reporter gene systems to specifically study the functional role of particular sequences, for instance, promoter [38], splicing [39], and RNA degradation [40] 1.6. Such reporter genes typically are expressed on plasmids, which are small circular DNA molecules encoding one or few genes. Genes on plasmids can be transcribed in living cells. The target sequence is first cloned into the plasmid, and then the gene expression outcome is measured. There are two conventional approaches to measure gene expression outcomes: directly target the transcripts by PCR or measure the protein outcome. Techniques such as green fluorescence tagging have been developed to directly track the protein products *in vivo* [41].

Previous reporter genes only had low throughput, which means only a few sequence elements can be studied in one experiment. High throughput assays have been developed in recent years with the development of the next-generation sequencing technique, termed as massively parallel reporter assay (MPRA) 1.6. Specifically, instead of testing

## 1 Introduction



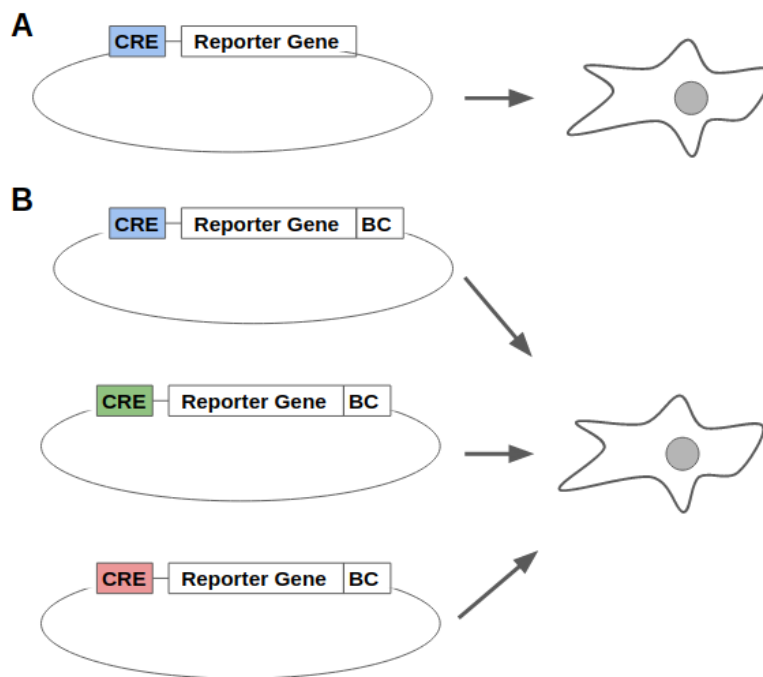
**Figure 1.5: RNA splicing quantification with RNA-Seq.** The definition of  $\Psi$ ,  $\Psi_5$ ,  $\Psi_3$  and splicing efficiency followed from [33, 32, 30, 22].

a single sequence construct, thousands of different constructs are cloned into different plasmids and expressed in living cells at the same time. Each of the sequence constructs typically contains also a barcode to uniquely associate the input test sequence to the gene expression outcome. RNA-Seq is used to measure the gene expression outcome. MPRA has been successfully applied to identify or validate cis-regulatory elements [42, 43, 44, 45, 46, 47, 48, 49] or directly used to test the effects of thousands of regulatory variants [50, 51, 52, 53, 54, 22, 55, 56].

Besides the high throughput, MPRA has other advantages. First, unlike statistical association studies which might suffer from other confounding factors (e.g., co-evolution of sequence elements), MPRA applies experimental perturbation and therefore can test for causal effects. Second, MPRA can not only examine regulatory elements from the genome but also synthesized sequences that never present in nature.

MPRA also has a few disadvantages. First, one MPRA experiment is typically done with one tissue, limiting the ability of MPRA to test for tissue-specific effects. Second, many gene expression activities depend on the native chromatin structure, which MPRA cannot test. Third, limited by the current nucleotide synthesis technique, which can only synthesis short ( $\sim 200$  nt) sequences, MPRA cannot test long-range dependent regulatory effects.

These shortcomings have now been partially overcome with the developing of the CRISPR/cas9 technique [57, 58]. Large-scale mutagenesis can now be carried natively on the chromosomes [59, 60].



**Figure 1.6: Reporter assays.** A). Traditional low-throughput reporter assay. The reporter gene is expressed on plasmids. The cis-regulatory element (CRE) is inserted to the corresponding position, either upstream of the reporter gene or in the gene body. Only one or few sequence constructs are tested in a single experiment. B). Massively parallel reporter assay. Thousands of different CREs are tested in a single experiment. Each sequence construct also attached with a unique barcode (BC), so that the outcome of each CRE can be uniquely mapped.

## 1.3 Variant interpretation

### 1.3.1 Genetic variants and human diseases

DNA sequences between individuals are not identical. Every human individual has on average one variant per thousand base pairs of sequence [61]. Genetic variants are typically defined as the DNA sequence differences compared to the reference genome released by the Genome Reference Consortium. The reference genome, however, is neither the wild-type sequence nor necessarily the consensus among the whole population. Common types of genetic variants include single nucleotide variants (SNVs), insertions/deletions (indels) or structure variants like copy number variants (CNVs). In population genetics, common (frequency  $> 1\%$ ) single nucleotide variations among the population are referred to as single nucleotide polymorphisms (SNPs). While most of those variants are benign, some are associated with higher disease risks, and some even directly cause disease with a single mutation.

Understanding genetic variants is crucial for human health. Many population sequencing initiatives have been launched, for example, the 1000 genome project [61] and the

## 1 Introduction

UK Biobank project [62]. Millions of variants have been identified from these projects. It is a long term goal for researchers to interpret their function.

Most of the variants are inherited from the parents, while a small fraction, on average 42 to 75 per human individual, are *de novo* [63]. These *de novo* variants arise from DNA copy errors during the reproduction of an individual. *De novo* mutations play a major role in causing severe early-onset genetic diseases such as intellectual disability, autism spectrum disorder, and other developmental disorders [64].

Most of the known disease-causing genetic variants are missense, which means they change the encoded protein sequence [65]. However, the current statistics might be biased due to two reasons: First, it is easier to functionally annotate a variant to be “missense” since the protein-coding genetic code is well understood. Second, whole exome sequencing (WES), which is targeted sequencing of the annotated exonic regions of the genome (including the coding regions), is more common than whole genome sequencing (WGS) due to its price advantage. Nevertheless, we anticipate seeing more WGS with the increasing acknowledgment of its value in diagnostic.

### 1.3.2 Interpreting regulatory variants

Interpreting regulatory variants is particularly challenging due to the lack of understanding of the regulatory sequences. One approach to interpret genetic variants is through statistical association tests. Two commonly used statistical approaches are: 1) expression quantitative trait loci (eQTL) [66], and 2) Genome-wide association studies (GWAS) [67]. Both approaches need a large number of genotyped individuals. The goal of eQTL is to find genomic loci that are significantly associated with gene expression level variations across individuals. Similar approaches can be applied to find loci that are associated with splicing variations (sQTL) [68]. In contrast, GWAS aims to find variants associated with certain phenotype such as disease.

Such statistical approaches suffer from two major limitations: First, due to the limitation of statistical power, it can only be applied to common variants (allele frequency  $>1\%$ ) with considerable effect. However, rare variants are the typical cause of rare diseases and maybe even common diseases as well [69]. Moreover, lethal variants may be even completely absent from the population. Even disease causing common variants can be overlooked by this approach. Common disease risk often involves a large number of variants with small effects, which would need an enormous sample size to be identified as statistically significant after multiple testing correction. Second, the association test does not imply causality. Frequently all variants in a specific region are significant, locating the causal variants is challenging.

On the other hand, machine learning methods are frequently used to study regulatory genomics and therefore to interpret regulatory variants. Genomics is a highly data-driven discipline where machine learning models are widely applied. The amount of data produced in genomics is comparable to astronomy, Twitter and YouTube [70]. This gives a unique chance for deep learning, one of the machine learning method categories that perform well with lots of data. The rise of modern deep learning methods also transformed how we study regulatory sequences [71, 72, 73], it is now often the model

of choice for many genomics applications. In contrast to GWAS, deep learning models can be applied to score variants irrelevant of its allele frequency and have been shown to capture disease causing variants in many cases [74, 75].

Therefore, this thesis is primarily focused on developing machine learning models to interpret regulatory variants. In particular, I focus on predicting the variant effect on splicing and mRNA degradation. On top of the functional impact prediction, a classifier to predict variant pathogenicity was trained. The next two subsections are dedicated to providing background review on computational methods for variant interpretation in terms of splicing and mRNA degradation. I give a brief introduction to the essential primers on machine learning for genomics in chapter 2.

#### 1.3.3 Computational methods for variant interpretation

Computational tools for variant interpretation can help to prioritize potential disease-causing variants, improving patient treatment outcomes and prognosis. For instance, computational tools have lead to successful diagnose of many rare diseases, which may otherwise be difficult with experimental methods due to the large test burden; BRCA1 and BRCA2 mutations are known to be associated with treatment outcomes and prognosis [76].

Variants cause disease by changing particular molecular phenotype. Many computational tools for variant interpretation have been developed. Most of the frequently used tools are designed to predict missense variants [77]. Training a model to directly predict disease risk from variants is difficult due to limited annotated data. An alternative way is to build hierarchical models. Specifically, the first level of models are trained by leveraging on large scale assay data to predict a specific biological process. For example, train a model to predict splicing outcomes from a massively parallel reporter assay that can probe millions of sequences in a single experiment as done by Rosenberg et al [49]. The second level is predicting disease on top of the predicted molecular consequences, possibly also combine with features like conservation. MutPred Splice is an example of such a model, it builds on top of models predicting splicing motifs to predict variant pathogenicity [78].

Here I provide a review of tools predicting the variant effect on splicing and degradation. Tools predicting the variant effect on molecular phenotype and tools directly predict disease impact are reviewed separately in 1.3.3.1 and 1.3.3.2.

##### 1.3.3.1 Functional impact prediction

**Splicing** In table 1.1 I summarized non-commercial models predicting variant impact on splicing. Early models focused on predicting donor and acceptor sites, therefore also score variants for their effect on creating/disrupting splice sites. These include GeneSplicer [79], NNSplice [80], NetGene2 [81], MaxEntScan [82] and the model from Sonnenburg et al [83]. Although neural networks were used, these models were in small scale and consequently, also have small receptive field. Many of these early tools have been integrated into the variant effect predictor (VEP) from Ensembl [84].

## 1 Introduction

Early models do not predict directly splicing quantities probably because there was no high throughput experiment to measure splicing on a large scale. This situation has changed since the introduction of the next generation sequencing techniques. RNA-Seq was used to quantify splicing systematically. Barash et al developed the first successful model to directly predict splicing quantity and variant effect [85]. Following this study, a similar model was developed for human (SPANR) [86]. ESRseqs are scores for all 4096 6-mers for their exonic effect on splicing [55]. The scores were derived from a massively parallel reporter assay (MPRA). The MPRA has a 3-exon reporter gene construct with a 6-mer random sequence on the middle-exon. Splicing outcomes associated with each of the random sequences were measured. Likewise, SMS scores were derived from a similar MPRA experiment, except the test sequences were generated by saturation mutagenesis [56]. HAL is a model developed by Rosenberg et al [49] to predict directly  $\Delta\Psi$ . It is a linear model trained from experimentally tested random k-mers from an MPRA experiment.

Even though previous methods have demonstrated success in specific applications, predicting the effect of any given variant on splicing is still challenging. First, the models scoring splice sites cannot score variants from exonic or intronic regulatory elements. Second, all three models which were trained from MPRA data, ESRseqs, SMS scores, and HAL can only score exonic variants, limiting their application cases. The only model that scores both intronic and exonic variants for human is SPANR. Third, it is challenging to score indels with all previous models due to the lack of proper implementation support. Fourth, the performance of all these models can be potentially improved with more data and better modeling techniques.

**RNA degradation** Several high throughput studies have investigated the potential effect of UTR elements on RNA stability [26, 87, 88, 89, 90, 91, 92]. These studies either directly associated UTR elements with RNA stability or indirectly through RNA expression level. Many known regulatory elements of mRNA stability have been identified, these include 3' UTR RBP binding motifs (e.g., AU-rich element [91]), microRNA binding sites [93], secondary structure [89] and translation related sequence elements (e.g., codon usage [40, 28]).

Despite these studies, we still do not know how far we are from a complete list of regulatory elements for mRNA stability. Even though models predicting mRNA half-life exist [94], predicting solely from sequence is an unsolved problem. Consequently, a model to predict the variant effect on mRNA half-life is lacking. Furthermore, we don't completely understand how certain sequence elements affect mRNA stability, namely the specific pathways employed are unknown.

One potential reason for these limitations is that the techniques to precisely measure RNA stability genome-wide are under mature. Results measured across experiments often correlate poorly [24, 1]. It is important to systematically benchmark these RNA stability data.

### 1.3.3.2 Disease impact prediction

**Splicing** Conventionally, variants are annotated in a five-tier terminology system: “pathogenic”, “likely pathogenic”, “uncertain significance”, “likely benign”, and “benign” [77]. However, when it comes to the predictions by computational tools, these definitions are irrational. Hence, most tools are trained to distinguish between “pathogenic” and “benign” variants. In general, conservation scores are good predictors for pathogenicity, although they are hard to interpret [95]. Accordingly, almost all variant pathogenicity tools use conservation features. Comparing to the tools predicting splice variant effect, relatively few tools directly predict pathogenicity from splicing (summarized in 1.2). MutPred Splice was trained from 16,257 variants, it scores exonic variant for their disease risks [78]. TraP predicts variant pathogenicity based on whether the variant changes the final outcome of the transcript, e.g. exon skipping, cryptic splice site activation, NMD [96]. It provides pre-computed scores for  $\sim 1.3$  billion possible single nucleotide variants from human protein-coding genes. The model was trained from a positive set with 75 synonymous rare-disease causing variants and a negative set with 402 *de novo* synonymous variants from presumably healthy individuals. S-CAP is a recent tool specifically designed to predict pathogenic splicing-relevant variants. The positive set of S-CAP training data combined 114,382 pathogenic SNVs from the HGMD [97] and ClinVar [65] databases, while the negative set was 15,833,389 SNVs curated from gnomAD database. The model considered three levels of features: the gene level (e.g., pLI [98]), the exon level (e.g. exon length) and the variant level (e.g., CADD score [95]). To this end, a gradient boosting tree classifier was trained to predict pathogenicity scores.

Except for S-CAP, which was published in parallel with the work of this thesis, the other two methods were trained with very few data points. Besides, MutPred Splice only scores exonic variant. With a better splice-variant effect prediction method, we can

Table 1.1: List of variant effect on splicing prediction tools. Sorted by publication year.

Tools	Prediction outcomes	Approach
NetGene2 (1991)	Predict score indicating splice site strength	Neural network
NNSplice (1997)	Predict score indicating splice site strength	Neural network
GeneSplicer (2001)	Predict score indicating splice site strength	Markov Model
MaxEntScan (2004)	Predict score indicating splice site strength	Maximum entropy principle
Sonnenburg et al., (2007)	SVM with weighted degree kernel	Maximum entropy principle
Human Splicing Finder (2009)	Webserver present multiple scores, e.g. splice site strength, splice enhancers/silencers	Combine several models
Barash et al (2010)	high/middle/low category of splice level	Splicing code assembly
ESRseqs (2011)	Predict splicing efficiency from 6-mers	Linear model
dbscSNV (2014)	Precomputed splice-altering score	AdaBoost and Random Forest
SPANR (2015)	Precomputed predictions of $\Delta\Psi$	Neural network
HAL (2015)	Predict $\Delta\Psi$	Linear model with k-mers
SMS scores (2018)	Predict splicing efficiency from 6-mers	Linear model
MMSplice (2019)	Predict $\Delta\Psi$	Neural network and linear model



likely improve the performance of all pathogenicity classifiers, which are almost always based on the ensemble of models.

Tools		Prediction outcomes	Approach
MutPred (2014)	Splice	Predict disease risk from VCF file	Random Forest
TraP (2017)		Precomputed scores	Random Forest
MMSplice (2019)		Predict disease risk from VCF file	Neural network & logistic regression
S-CAP (2019)		Precomputed scores	Gradient Boosting Tree

**Table 1.2:** List of pathogenicity prediction tools with splicing focus. Sorted by publication year.

**RNA degradation** To my knowledge, due to the lack of predictive model for mRNA stability, no model so far predict variant pathogenicity from its effect on RNA stability.

## 1.4 Aims and scope of this thesis

Post-transcriptional regulations are crucial regulatory steps. My PhD work is dedicated to model the sequence determinants for two of these processes: RNA degradation and splicing. The corresponding articles are provided in the Appendices.

### The key issues regarding the previous models are:

1. Many sequence determinants for mRNA half-life have been previously identified, however, no model can predict mRNA half-life solely from sequence even for yeast. As a consequence, it is not possible to quantify the contributions of different sequence elements to the overall half-life variation, neither do we know how far are we from a complete list of mRNA stability-regulating sequence elements.

2. Predicting variant effect on splicing is still challenging. Previous models can either only score certain regions (e.g., exon) or are only applicable to specific alternative splicing patterns. Besides, performances of existing models are unsatisfactory. Moreover, adequately implemented software is lacking such that indels cannot be easily scored.

### Briefly, the major contributions of this thesis include:

1. Systematically investigated the sequence determinants of messenger RNA (mRNA) stability in yeast. One novel 3' UTR motif was revealed.

2. From the investigated sequence determinants, a regularized linear model was able to explain 59% of the half-life variation between genes. Quantifying feature importance revealed the major role of codon usage in controlling mRNA stability, while mutations on 3' UTR motifs and upstream AUG have the largest effect size.

3. Developed a deep learning framework to predict the variant effect on splicing with state-of-the-art accuracy. The model can score variants, including indels, from both exons and introns. The improved variant effect prediction model can also improve the prediction of variant pathogenicity.



## 2 Machine Learning background

Machine learning assumes the world appears with certain patterns. The goal of machine learning is to find computer executable mathematical functions to express these patterns. However, the true pattern often remains unknown. In recent years, machine learning, in particular, deep learning has seen tremendous success in many areas such as computer vision, natural language processing, machine translation as well as computational biology.

Based on the task, machine learning can be categorized into several categories, for instance, supervised learning, unsupervised learning and reinforcement learning. This thesis focuses on the applications of supervised learning in studying regulatory genomics.

This section provides the necessary technical background knowledge on machine learning. For a more thorough introduction to machine learning we recommend the book of Bishop [99] and Goodfellow et al [100].

### 2.1 Supervised learning

Many predictive tasks can be formulated as predicting  $\mathbf{y}$  from some given input  $X$  with a function  $f: X \rightarrow \mathbf{y}$ . For instance, predicting house price from given information (features) e.g. location, size. In regulatory genomics, such tasks typically are predicting molecular phenotype (e.g. protein binding, splicing, stability) from DNA/RNA sequence. In most cases, it is difficult to predefine the function  $f$  as computer rules. Supervised learning is aimed to learn a function to perform the  $X \rightarrow \mathbf{y}$  mapping from previous “experiences”. These “experiences” are represented as training data consisting pairs of inputs mapped with outputs  $(\mathbf{x}, y) \in X \times \mathbf{y}$ .

### 2.2 The learning objective

We formally define the learning object. Assume we observed independent and identically distributed (i.i.d) training data  $\mathcal{D}_{\text{train}} : \{(x_1, y_1), \dots, (x_n, y_n)\}$  from an unknown distribution  $p_{\text{data}}(y|\mathbf{x})$ . We are interested in predicting future data points that are also drawn from the same distribution. To do so, we approximate the unknown true data distribution with a known distribution with parameter  $\boldsymbol{\theta}$ , denote as  $p_{\text{model}}(y|\mathbf{x}, \boldsymbol{\theta})$ . The learning task is to find a close approximation of the conditional distribution  $p_{\text{data}}(y|\mathbf{x})$  by tuning  $\boldsymbol{\theta}$ . We use Kullback-Leibler divergence (KL divergence) to measure the closeness between the data distribution and the model distribution. We minimize KL divergence to find the “best” approximation of  $p_{\text{data}}$ .

## 2 Machine Learning background

$$\boldsymbol{\theta}^* = \arg \min_{\boldsymbol{\theta}} D_{\text{KL}}(p_{\text{data}} || p_{\text{model}}) \quad (2.1)$$

The KL divergence is given by:

$$\begin{aligned} D_{\text{KL}}(p_{\text{data}} || p_{\text{model}}) &= \mathbb{E}_{y|\mathbf{x} \sim p_{\text{data}}} [\log p_{\text{data}}(y|\mathbf{x}) - \log p_{\text{model}}(y|\mathbf{x}; \boldsymbol{\theta})] \\ &= \mathbb{E}_{y|\mathbf{x} \sim p_{\text{data}}} [\log p_{\text{data}}(y|\mathbf{x})] - \mathbb{E}_{y|\mathbf{x} \sim p_{\text{data}}} [\log p_{\text{model}}(y|\mathbf{x}; \boldsymbol{\theta})] \end{aligned} \quad (2.2)$$

The term on the left  $\mathbb{E}_{y|\mathbf{x} \sim p_{\text{data}}} [\log p_{\text{data}}(y|\mathbf{x})] = -\mathbb{H}(p_{\text{data}})$  is the minus entropy of  $p_{\text{data}}$ , which does not depend on the parameter  $\boldsymbol{\theta}$ . Therefore, minimizing the KL divergence is equivalent to minimizing the term on the right, which is the cross-entropy between the two distributions.

$$\begin{aligned} \boldsymbol{\theta}^* &= \arg \min_{\boldsymbol{\theta}} D_{\text{KL}}(p_{\text{data}} || p_{\text{model}}) \\ &= \arg \min_{\boldsymbol{\theta}} -\mathbb{E}_{y|\mathbf{x} \sim p_{\text{data}}} [\log p_{\text{model}}(y|\mathbf{x}; \boldsymbol{\theta})] \\ &= \arg \max_{\boldsymbol{\theta}} \mathbb{E}_{y|\mathbf{x} \sim p_{\text{data}}} [\log p_{\text{model}}(y|\mathbf{x}; \boldsymbol{\theta})] \end{aligned} \quad (2.3)$$

However, we don't know the true data distribution  $p_{\text{data}}$  in general. Therefore, in practice we estimate the expectation with monte carlo from our training data, which are samples drawn from the true distribution  $p_{\text{data}}$ :

$$\mathbb{E}_{y|\mathbf{x} \sim p_{\text{data}}} [\log p_{\text{model}}(y|\mathbf{x}, \boldsymbol{\theta})] \approx \frac{1}{|\mathcal{D}_{\text{train}}|} \sum_{i=1}^{|\mathcal{D}_{\text{train}}|} \log p_{\text{model}}(y_i|\mathbf{x}_i; \boldsymbol{\theta}) \quad (2.4)$$

In summary, our optimization object becomes:

$$\boldsymbol{\theta}^* = \arg \max_{\boldsymbol{\theta}} \frac{1}{|\mathcal{D}_{\text{train}}|} \sum_{i=1}^{|\mathcal{D}_{\text{train}}|} \log p_{\text{model}}(y_i|\mathbf{x}_i; \boldsymbol{\theta}) \quad (2.5)$$

This is exactly the log-likelihood function. The above estimation is also known as the *maximum likelihood estimation (MLE)*.

Another way to interpret the maximum likelihood estimation is through the joint data likelihood under our model distribution. The data likelihood for  $\mathcal{D}_{\text{train}}$  is defined as  $p_{\text{model}}(\mathbf{y}|X, \boldsymbol{\theta})$ . The maximum likelihood estimator for  $\boldsymbol{\theta}$  is then defined as:

$$\begin{aligned} \boldsymbol{\theta}_{\text{ML}} &= \arg \max_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}) \\ &= \arg \max_{\boldsymbol{\theta}} p_{\text{model}}(\mathbf{y}|X; \boldsymbol{\theta}) \\ &= \arg \max_{\boldsymbol{\theta}} \prod_{i=1}^{|\mathcal{D}_{\text{train}}|} p_{\text{model}}(y_i|\mathbf{x}_i; \boldsymbol{\theta}) \end{aligned} \quad (2.6)$$

We can factor out the joint distribution because of the iid assumption. The product is numerically unstable, therefore a logarithm is often taken to transform the product into a summation. Since the logarithm is a monotonic increasing function, it does not change the optimum.

$$\begin{aligned}
\boldsymbol{\theta}_{\text{ML}} &= \arg \max_{\boldsymbol{\theta}} \sum_{i=1}^{|\mathcal{D}_{\text{train}}|} \log p_{\text{model}}(y_i | \mathbf{x}_i; \boldsymbol{\theta}) \\
&= \arg \max_{\boldsymbol{\theta}} \frac{1}{|\mathcal{D}_{\text{train}}|} \sum_{i=1}^{|\mathcal{D}_{\text{train}}|} \log p_{\text{model}}(y_i | \mathbf{x}_i; \boldsymbol{\theta}) \\
&= \arg \min_{\boldsymbol{\theta}} -\frac{1}{|\mathcal{D}_{\text{train}}|} \sum_{i=1}^{|\mathcal{D}_{\text{train}}|} \log p_{\text{model}}(y_i | \mathbf{x}_i; \boldsymbol{\theta})
\end{aligned} \tag{2.7}$$

Which is the same as equation 2.5. The last term above is also known as the *negative log-likelihood (NLL)*.

## 2.3 Regularization

Above we introduced the objective function of the maximum likelihood estimator 2.5. In machine learning, the objective function is typically referred to as the *loss function*. The loss function is used to quantify the error made by our predictive algorithms. MLE is one of the approaches to derive a loss function. In theory, we should optimize the loss function so that the model has the best performance on the upcoming unseen samples. The cost function is the overall error on the entire training data, which is often the mean of the per-sample loss. We express our learning objective above as the cost function:

$$J(\boldsymbol{\theta}) = \frac{1}{|\mathcal{D}_{\text{train}}|} \sum_{i=1}^{|\mathcal{D}_{\text{train}}|} L(f(\mathbf{x}_i; \boldsymbol{\theta}), y_i) \tag{2.8}$$

Where  $L$  is the per-sample loss function.  $L(f(\mathbf{x}_i, \boldsymbol{\theta}), y_i) = -\log p_{\text{model}}(y_i | \mathbf{x}_i, \boldsymbol{\theta})$ .

However, since the above cost function (as well as equation 2.5) is an approximation of equation 2.3 with finite data (in many cases, limited), it might lead to certain undesired failure. For instance, if the function  $f$  with parameters  $\hat{\boldsymbol{\theta}}$  can predict every data point in  $\mathcal{D}_{\text{train}}$  perfectly but fail miserably on the unseen test data  $\mathcal{D}_{\text{test}}$ . The parameters  $\hat{\boldsymbol{\theta}}$  are certainly the optimizer of the cost function  $J(\boldsymbol{\theta})$ , but the model is useless. In machine learning, such kind of failure in generalization is referred to as *overfitting*.

To solve the above issue, we impose certain regularization on the cost function:

$$J(\boldsymbol{\theta}) = \frac{1}{|\mathcal{D}_{\text{train}}|} \sum_{i=1}^{|\mathcal{D}_{\text{train}}|} L(f(\mathbf{x}_i; \boldsymbol{\theta}), y_i) + \lambda R(\boldsymbol{\theta}) \tag{2.9}$$

Where the  $\lambda$  controls the weighting between the loss and the regularization term  $R(\boldsymbol{\theta})$ .

## 2 Machine Learning background

Another scenario where regularization is necessary is model selection. For the generalization purpose, simpler models are preferred over complex ones if they have similar performance on the training data. Depends on the data, the learner might strengthen more on the loss function or the regularization term by changing  $\lambda$ . The optimum value of  $\lambda$  (as well as other hyper-parameters) is often determined by *cross-validation*.

### 2.4 Regression

In the previous sections I introduced the MLE approach to learn the model parameters. However, the model function  $f$  and the distribution assumption  $p_{\text{model}}$  haven't been specified yet. In the following, I show three specific examples in a regression setting.

#### 2.4.1 Linear regression

The first example is the standard linear regression, which assumes gaussian isotropic noise. Let the feature matrix  $X \in \mathbb{R}^{N \times D}$  and  $X = \begin{bmatrix} | & | & \cdots & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_D \\ | & | & \cdots & | \end{bmatrix}$ . The target  $\mathbf{y} \in \mathbb{R}^{N \times 1}$  and  $\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}$ . Specifically for function  $f$  and  $p_{\text{model}}$ , linear model assumes:

$$\begin{aligned} p_{\text{model}}(\mathbf{y}|X; \mathbf{w}, \sigma^2) &= \mathcal{N}(f(X; \mathbf{w}), \sigma^2 \mathbf{I}) \\ f(X; \mathbf{w}) &= X \mathbf{w} \end{aligned} \quad (2.10)$$

Here the parameters  $\boldsymbol{\theta}$  include  $(\mathbf{w}, \sigma)$ .

In summary:

$$\begin{aligned} y_i &= f(\mathbf{x}_i; \mathbf{w}) + \epsilon_i \\ &= \mathbf{w}^T \mathbf{x}_i + \epsilon_i \end{aligned} \quad (2.11)$$

The log-likelihood for the linear model is:

$$\mathcal{L}(\mathbf{w}, \sigma) = -\frac{1}{2\sigma^2} \sum_{i=1}^N (f(\mathbf{x}_i; \mathbf{w}) - y_i)^2 - ND \log \sqrt{2\pi} - N \log \sigma \quad (2.12)$$

where  $N = |\mathcal{D}_{\text{train}}|$  is the number of data points in the training data.  $D$  is the number of features. Only the first term of the log-likelihood function is related to  $\mathbf{w}$ , which is the parameter of the function  $f$ . Besides,  $\sigma^2 \geq 0$ , we can simplify the MLE objective function for  $\mathbf{w}$  as follows:

$$\begin{aligned}
\mathbf{w}_{\text{ML}} &= \arg \max_{\mathbf{w}} \mathcal{L}(\mathbf{w}) \\
&= \arg \min_{\mathbf{w}} \frac{1}{2} \sum_{i=1}^N (f(\mathbf{x}_i; \mathbf{w}) - y_i)^2
\end{aligned} \tag{2.13}$$

The above optimizing object is referred as the mean squared error (MSE) loss function. In the case of linear regression, i.e.  $f(\mathbf{x}_i; \mathbf{w})$  is an affine function in terms of  $\mathbf{w}$ , the MLE for  $\mathbf{w}$  can be derived analytically as:

$$\mathbf{w}_{\text{ML}} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} \tag{2.14}$$

However, the function  $f$  does not necessarily need to be affine in order for the mean squared loss to be applied. More complex functions such as neural networks follow the same principle. In this case, an analytical solution is likely not possible, and one would need to employ numerical optimization methods to obtain the MLE of the function parameters. We briefly introduce some of these optimization methods in section 2.6.

The MLE for  $\sigma$  can be done similarly as well.

### 2.4.2 Beta regression

The above example shows MLE with isotropic gaussian noise. This assumption however does not hold for certain data. For instance, when  $y|\mathbf{x} \in [0, 1]$ , then a beta distribution ( $\mathcal{B}$ ) which is defined between  $[0, 1]$  is more appropriate. Concretely, we assume that:

$$p_{\text{model}}(y_i|\mathbf{x}_i; \mathbf{w}, \phi) = \mathcal{B}(f(\mathbf{x}_i; \mathbf{w}), \phi) \tag{2.15}$$

here  $f$  is a function that maps the input  $\mathbf{x}$  to values between  $[0, 1]$  with parameters  $\mathbf{w}$ .

In fact, in generalized linear model (GLM),  $f(\mathbf{x}; \mathbf{w}) = g^{-1}(\mathbf{w}^T \mathbf{x})$ , where  $g(x) = \ln(\frac{x}{1-x})$  is the logit function ( $g(x)$  is referred as the link function in GLM) [101]. However, one can use other functions like neural network as well. We see later that GLM can be seen as an one-layer neural network and the activation function in the output layer is  $g^{-1}$ .

The beta regression uses an alternative density function for beta distribution proposed by Ferrari and Cribari-Neto [102]. The density function reparameterize the beta distribution with mean and precision (like the normal distribution) instead of the canonical  $\mathcal{B}(a, b)$  format. Specifically:

$$f(y; \mu, \phi) = \frac{\Gamma(\phi)}{\Gamma(\mu\phi)\Gamma((1-\mu)\phi)} y^{\mu\phi-1} (1-y)^{(1-\mu)\phi-1} \tag{2.16}$$

where  $\mu := f(\mathbf{x}; \mathbf{w})$  is the model predicted  $y$  in our case.

The log-likelihood for beta regression model is:

## 2 Machine Learning background

$$\mathcal{L}(\mathbf{w}, \phi) = N \log(\Gamma(\phi)) - \sum_{i=1}^N \left\{ -\Gamma(\mu_i \phi) - \log \Gamma(1 - \mu_i \phi) - (\mu_i \phi - 1) \log y_i + ((1 - \mu_i) \phi - 1) \log(1 - y_i) \right\} \quad (2.17)$$

where  $N = |\mathcal{D}_{\text{train}}|$  is the number of data points in the training data. MLE for  $\mathbf{w}$  and  $\phi$  do not have analytical solution in this case. The maximum likelihood estimator for  $\mathbf{w}$  can be simplified by dropping the terms irrelevant to  $\mathbf{w}$  as:

$$\mathbf{w}_{\text{ML}} = \arg \min_{\mathbf{w}} \sum_{i=1}^N \left\{ \Gamma(\mu_i \phi) + \log \Gamma(1 - \mu_i \phi) - \mu_i \phi \log y_i - (1 - \mu_i) \phi \log(1 - y_i) \right\} \quad (2.18)$$

$\phi$  is viewed as a constant in the above estimator for  $\mathbf{w}$ .

The readers can refer to [102] for more technical details.

### 2.4.3 Linear mixed model

The above two examples are under the assumption that the data are iid. However, this assumption often does not hold. For instance, we want to test whether a genetic variant is associated with certain phenotype, e.g., gene expression level. The individuals we consider might not be completely independent due to gender, family and ethnic background, etc.. In other words,  $p_{\text{data}}(y_i | x_i)$  are not independent unless we also control for other covariations. Consequently, we can no longer factor out the joint distribution  $p(\mathbf{y} | X)$  in 2.6 into a product.

Formally, let the feature matrix  $X \in \mathbb{R}^{N \times D}$  and  $X = \begin{bmatrix} | & | & \cdots & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_D \\ | & | & \cdots & | \end{bmatrix}$ , the target  $\mathbf{y} \in \mathbb{R}^{N \times 1}$  and  $\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}$ . Consider our data are drawn from the following hierarchical model:

$$\begin{aligned} \mathbf{u} &\sim \mathcal{N}(0, \mathbf{K}) \\ \mathbf{y} | \mathbf{u} &\sim \mathcal{N}(f(X; \mathbf{w}) + \mathbf{u}, \sigma^2 \mathbf{I}) \end{aligned} \quad (2.19)$$

where  $\mathbf{K} \in \mathbb{R}^{N \times N}$  is the relatedness matrix, and encodes the covariance structure of the input data. Elements in  $\mathbf{K}$  indicate the similarities between inputs  $\mathbf{x}_i$ .

From 2.19 we can derive the marginal distribution of  $\mathbf{y}$  as:

$$\begin{aligned} p(\mathbf{y} | f(X; \mathbf{w}); \sigma, \mathbf{K}) &= \int p(\mathbf{y} | \mathbf{u}) p(\mathbf{u}) d\mathbf{u} \\ &= \mathcal{N}(f(X; \mathbf{w}), \mathbf{K} + \sigma^2 \mathbf{I}) \end{aligned} \quad (2.20)$$



We see the covariance for the marginal distribution of  $\mathbf{y}$  is the sum of the covariances of  $\mathbf{u}$  and  $\mathbf{y}|\mathbf{u}$ . This can be intuitively interpreted as the randomness from  $\mathbf{y}$  and  $\mathbf{y}|\mathbf{u}$  are independent.

For simplicity, we consider again  $f$  is a linear function in  $\mathbf{w}$ :  $f(X; \mathbf{w}) := X\mathbf{w}$ . The above model 2.19 can be written as follow:

$$\mathbf{y} = X\mathbf{w} + \mathbf{u} + \epsilon \quad (2.21)$$

where  $\epsilon$  is the random noise term  $\epsilon \sim \mathcal{N}(0, \sigma^2\mathbf{I})$ .

This model is known as the *linear mixed model (LMM)* [103].

To perform MLE for the LMM, we need to derive the data likelihood under our model  $p_{\text{model}}(\mathbf{y}|X, \sigma, \mathbf{w})$ , which we have done in 2.20. The MLE follows the same logic as 2.6, expect that we can not make the independence assumption but consider the joint distribution instead (the joint distribution cannot be factored out). Since the joint distribution is multivariate normal, we can derive the close form solution for  $\mathbf{w}$  as:

$$\mathbf{w}_{\text{ML}} = (X^T\Sigma^{-1}X)^{-1}X^T\Sigma^{-1}\mathbf{y} \quad (2.22)$$

where  $\Sigma = \mathbf{K} + \sigma^2\mathbf{I}$ . The above estimator is also known as the *weighted least square*.

## 2.5 Neural networks

Neural networks (NNs) with even a single hidden layer and a sufficient number of hidden units are universal function approximators [Kurt Hornik 1991]. The activation function is essential but not limited to the sigmoid function. This theory cannot guarantee, however, neural networks work universally. One cannot train a NN with an infinite number of hidden units, for instance.

Deep neural network models (deep learning) have been widely successful in many fields, including computational biology. Several advantages have led to the success of deep learning. First, deep learning is scalable. Unlike other machine learning models, the performance of deep learning does not saturate with more data [104]. Second, deep learning models can be trained end-to-end. Deep learning does not need human experts to curate features [105]. It turns the feature engineering process into a trainable learning process instead. Layers in deep learning represent gradually from low-level features to high-level features. Features learned by deep learning models can be reused in related tasks (transfer learning). Such property enables deep learning to be easily applied to unstructured data like images, text or DNA/RNA sequences. Third, several deep learning architectures can capture dependencies between spatially distant features, e.g., LSTM [106], dilated convolution layers [107].

Genomics is a highly data-driven field. Large data consortiums like ENCODE [108], GTEx [109] and TCGA [110] as well as large scale massively parallel reporter assays provide enormous data for model training. Deep learning has been widely applied to predict transcription factor binding [75], RNA binding protein binding [5], splicing [86], DNA accessibility [74] and DNA methylation [72]. These models enabled *de novo* prediction of variant effect and provided powerful toolsets for genetic diagnosis.

### 2.5.1 Vanilla Neural Networks

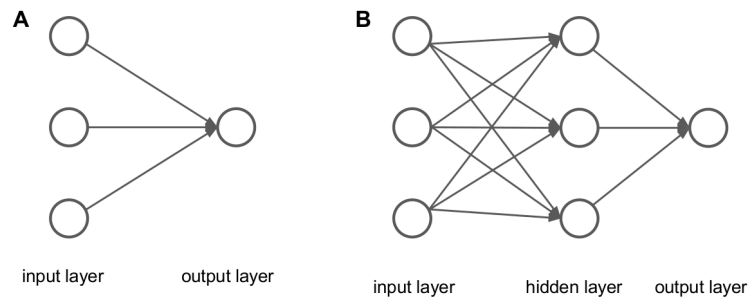
A vanilla fully connected neural network can be expressed as a stack of matrix multiplications and element-wise non-linearities. For instance, a simple neural network with two-layers is  $f(\mathbf{x}, W) = W_2\sigma(W_1\mathbf{x})$  (Figure 2.1). Rectified linear unit (ReLU), tanh and the sigmoid functions are the most frequently used activation functions. GLMs can be viewed as a NN without hidden layers (Figure 2.1). We notice that a NN without non-linearity collapse to a GLM:  $W_2W_1\mathbf{x} = \mathbf{w}^T x$ .

With a single hidden layer, NN can already represent functions a GLM cannot. We show with a XOR example (Figure 2.2). Consider the following classification problem with input  $X$  and output  $\mathbf{y}$ :

$$X = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

We know that linear models cannot classify the XOR example correctly. However, a neural network with one hidden layer and ReLU activation is able to predict  $\mathbf{y}$  correctly. An example solution with neural network is  $f(\mathbf{x}) = \text{ReLU}(W_1\mathbf{x} + b_1)^T W_2 + b_2$ . Where

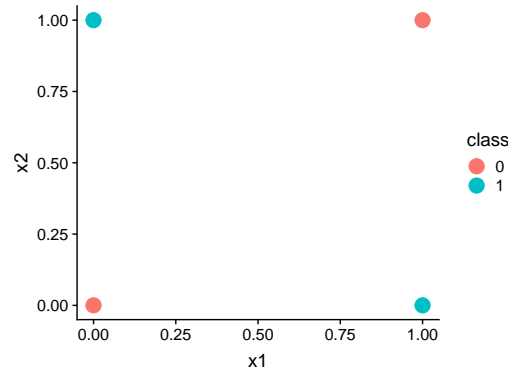
$$W_1 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \quad b_1 = \begin{bmatrix} 0 \\ -1 \end{bmatrix}, \quad W_2 = \begin{bmatrix} 1 \\ -2 \end{bmatrix}, \quad b_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$



**Figure 2.1:** Neural network. (A) Neural network without a hidden layer is a generalized linear model (GLM). (B) two-layer neural network with one hidden layer. This model is able to classify the XOR gate problem.

### 2.5.2 Convolutional Neural Network

Convolutional neural network (CNN, or ConvNet) uses learnable “filters” to scan the input space to find matching patterns (Figure 2.3). It is a type of deep learning architecture used for data with certain spatial patterns, such as images or texts. Applying convolution layer to DNA/RNA sequence is analog to scanning with motif positional weight matrix (PWM), which is a primary way to check whether a DNA/RNA motif is



**Figure 2.2:** XOR problem.

present in a given sequence [111]. The scanning also implies biological interpretations. Many proteins, in fact, find the correct binding site by scanning the sequence with an order, for instance, the ribosome scanning to find the translational start site [112].

The scanning process is in fact doing a dot product (on the flattened array) with a certain patch of sequence the same size as the filter. Unlike the fully connected layers, convolution layers use the same filters to multiply with all the scanned patches. It saves parameters significantly and therefore allows the model to scale up.

The output of the filter scanning is called the activation map. Each entry of the map corresponds to the result of one dot product (often also subject to a non-linearity function) (Figure 2.3). Multiple convolution layers are often applied consecutively to increase the receptive field and build up more complex feature representations gradually. A pooling operation (taking the max or average locally or globally along the sequence) is typically followed after several convolution layers (Figure 2.3). A pooling layer can potentially remove noise, reduce the number of parameters and increase the receptive field for the following layers.

Finally, a fully connected layer (recent architectures turned to use convolution layers as well) is applied in the end to transform all the learned features into a final prediction (class label or regression prediction) (Figure 2.3).

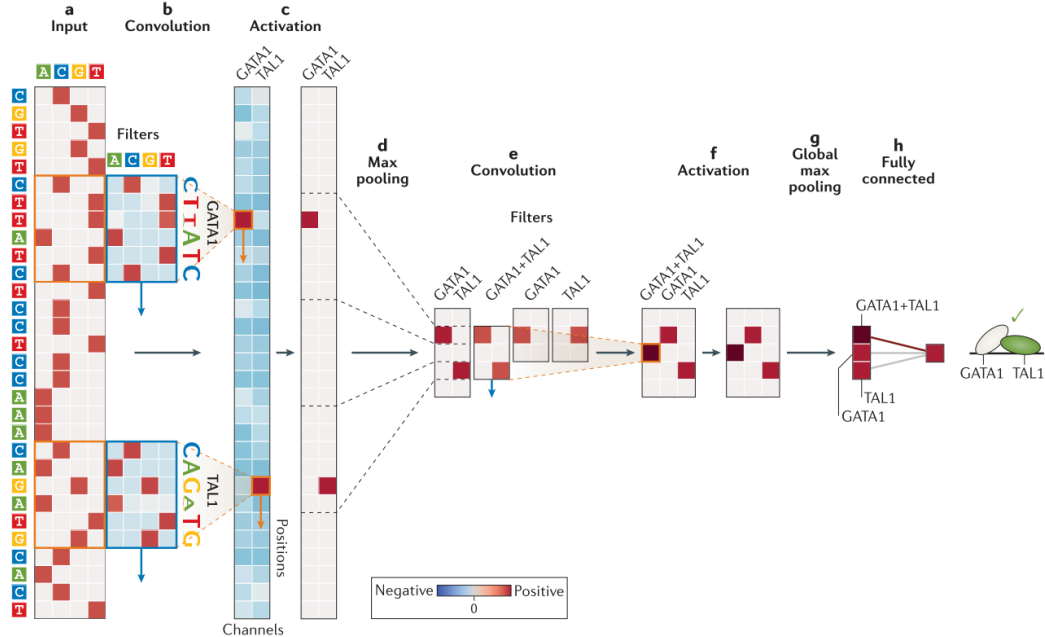
The convolutional neural networks have the advantage of end-to-end learning. They can be trained end-to-end to make predictions directly from one-hot encoded sequences. Therefore, they save the step of hand-craft feature engineering, which is complicated and not necessarily perform better. Convolutional neural network is a type of machine learning method called representation learning, which turns feature engineering into a learnable process.

Moreover, certain techniques have been developed to interpret CNN models. Early deep learning models in genomics tried to directly interpret the learned filters as positional weight matrices [75, 74, 72]. This approach faces the main challenge: CNN often learns a motif jointly with several filters. Namely, one filter rarely represents one motif. Other techniques have been developed to analyze contributions on the input space to the final prediction by perturbing the input [4] or by propagating the activation differences

## 2 Machine Learning background

(DeepLIFT) [113]. When applied to genomics, these techniques successfully identified biological meaningful sequence motifs [114].

Other neural network architectures were also applied to genomics. For example, the recurrent neural network (RNN) is another popular architecture to model sequence data. Systematical comparisons between RNN and CNN architectures in genomics is lacking. However, it seems that CNN is more common in genomics compared to RNN. A potential reason could be that, filter scanning is similar to PWM scanning, which is intuitive in biology.



**Figure 2.3:** Typical convolutional neural network in genomics. Figure taken from [[73], Figure 2]. The task here is to predict the binding of a transcription factor complex: TAL1–GATA1. **a** The input of the convolutional neural network is one-hot encoded DNA sequence. **b** The network architecture starts with a convolution layer which scans the input sequence with filters. Here we have two filters, one encodes the PWM of GATA1, and the other encodes the PWM of TAL1. **c** The scanning results are gated by the rectified-linear unit (ReLU) activation function, which set negative values to 0. **d** A max pooling is applied per region, which only keeps the maximum value in each region. **e** A second convolution operation is applied to search for the co-occurrence of the two motifs. In this way, the network learns the motif interaction. **f** A ReLU activation function is applied. **g** A global max pooling is applied along the input sequence direction. **h** A fully connected layer, which is similar to a generalized linear model applied to the learned features from previous layers, is used to make the final prediction.

Reprinted by permission from Springer Nature: Nature Reviews Genetics [73], © Springer Nature 2019

## 2.6 Optimization

In section 2.2 we introduced that supervised learning can be reduced to an optimization problem once we decided with the learning object (with regularization). Mathematical optimization problems can be categorized into convex and non-convex optimization depending on the convexity of the object function in terms of the parameters. In machine learning applications, the object function is typically differentiable, which means we can apply gradient-based optimization methods.

A commonly used optimization method based on the first order gradient is the gradient descent algorithm (GD). The gradient of the function  $\nabla f(x)$  indicates the slope of the function at  $x$ . Gradient descent algorithm follows the negative direction of the gradient at  $x$  with a small step size. The original GD update is as follows:

$$w^{k+1} \leftarrow w^k - \alpha \nabla f(w^k) \quad (2.23)$$

where  $w$  is the parameter to optimize and  $\alpha$  is the step size.

Gradient descent can be interpreted with Taylor expansion. Consider the Taylor expansion of  $f$  at  $w^k$ :

$$f(w^{k+1}) \approx f(w^k) + \nabla f(w^k)^T (w^{k+1} - w^k) + \frac{1}{2} (w^{k+1} - w^k)^T \nabla^2 f(w^k) (w^{k+1} - w^k) \quad (2.24)$$

the expression on the right hand side is the quadratic approximation of the function  $f(w^{k+1})$  (Figure 2.4).

For the first order method, we assume the Hessian  $\nabla^2 f(w^k)$  to be a constant  $\frac{1}{\alpha}$ . It is easy to optimize the right hand side in terms of  $w^{k+1}$ :

$$\arg \min_{w^{k+1}} f(w^k) + \nabla f(w^k)^T (w^{k+1} - w^k) + \frac{1}{2\alpha} \|w^{k+1} - w^k\|^2 \quad (2.25)$$

The solution of 2.25 gives the gradient descent update rule shown in 2.23.

If we use the full Hessian  $\nabla^2 f(w^k)$  and optimize the right hand side of 2.24:

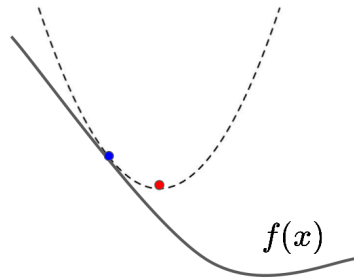
$$\arg \min_{w^{k+1}} f(w^k) + \nabla f(w^k)^T (w^{k+1} - w^k) + \frac{1}{2} (w^{k+1} - w^k)^T \nabla^2 f(w^k) (w^{k+1} - w^k) \quad (2.26)$$

we get:

$$w^{k+1} \leftarrow w^k - (\nabla^2 f(w^k))^{-1} \nabla f(w^k) \quad (2.27)$$

which is in fact the update rule for Newton's method [115]. As it requires the calculation of the Hessian matrix, it is a second order optimization method.

In deep learning, as we typically deal with a very large dataset, it is more efficient to update with small minibatch samples. In practice, optimizing with small batch size is not only more efficient but also yields better performance in deep learning models. The gradient descent algorithm that works with small minibatch instead of the whole



**Figure 2.4:** Quadratic approximation of the object function with Taylor expansion. The solid line indicates the object function  $f(x)$ , while the dashed line indicates the one-step quadratic approximation line from the blue point. The red point indicates the optimum of the approximated line.

batch of samples is termed as stochastic gradient descent (SGD). Several modifications of SGD have been shown to improve the performance of SGD. One particularly useful modification is adding momentum to the current gradient [116]. The momentum is a decayed moving average of gradients from previous steps ( $v^{k+1} = \mu v^k - \alpha \nabla f(w^k)$ ), so that the algorithm does not stuck at saddle points easily. Adam [117] is a widely used variation of SGD. Besides momentum, Adam also adapts the learning rate according to the scale of the gradient for different parameters: parameters seen larger gradient will have a smaller learning rate than parameters seen the smaller gradient.

In general, the cost functions of deep learning models are non-convex in terms of model parameters. SDG with momentum can effectively escape from saddle points, but would likely converge to local minima. Empirically deep learning models can generalize well even with local minima. Recent theoretical analyses show that local minima are “good” and global minima often means overfitting [118, 119].

Besides the model parameters, deep learning models are sensitive to hyperparameters. Important hyperparameters include the learning rate and the regularization strength. It is common to optimize hyperparameters with random search, Bayesian optimization approach, or tree-structured parzen estimator approach [120]. Because the learning rate is multiplied to the gradient (2.23), it has a multiplicative effect on the learning. Therefore it is common to search the optimal learning rate on a log scale.

### 3 Discussion and Outlook

Post-transcriptional regulation is one of the major mechanisms employed by the cell to regulate gene expression. Most of these regulations involve interactions between RNA binding proteins and sequence patterns on the RNA. The growing amount of data and advances in machine learning techniques enabled researchers to build models to predict quantities of these regulations. These models take genetic sequences as inputs and predicts certain molecular phenotype from these input sequences. Models of this kind are extremely helpful in interpreting genetic variants and predicting their pathogenicity.

I present in this thesis models for two important post-transcriptional regulations: splicing and mRNA degradation. The splicing model, MMSplice, accurately predicts variant effect on various splicing quantities, including  $\Psi$ ,  $\Psi_5$ ,  $\Psi_3$  and splicing efficiency. By building better models to predict the functional impact, we can also predict variant pathogenicity with improved accuracy.

The modular approach of MMSplice is flexible. However, since each module is trained independently, some long-range dependencies might be overlooked. During the review of the MMSplice manuscript, another method to predict variant effect on splicing was published [121]. The tool, SpliceAI, can accurately predict whether a given variant can create or disrupt a splice site. SpliceAI is a deep learning model with residual connections and dilated convolutions. It takes very large input sequence length (20k) to capture long-range dependencies. However, SpliceAI is not able to predict changes on splicing quantities mentioned above. For instance, if a variant disrupts or creates a splicing enhancer on an alternative spliced exon, SpliceAI is not able to predict the corresponding  $\Psi$  change. Furthermore, SpliceAI is trained from the reference sequence, while MMSplice took advantage of several perturbation data. As a consequence, SpliceAI might suffer more from correlative features result from coevolution. Therefore, it is recommendable to use SpliceAI along with MMSplice. Nevertheless, SpliceAI is a good example of end-to-end learning with a deep learning model.

The performance of MMSplice was evaluated with measured variant effects from two high throughput reporter assays: Vex-seq and MaPSy. Despite similarities, the exons tested by the two assays are different. Vex-seq used alternatively spliced exons while MaPSy used constitutive exons. It is to this point unclear whether variant primarily changes  $\Psi$  for alternatively spliced exons while primarily change splicing efficiency for constitutively spliced exons. One could potentially analyze the raw data from both assays to answer this question.

Sequence variants affecting mRNA stability have largely been ignored in diagnosis pipelines, mainly because the lack of corresponding computational tools to predict variant effect on mRNA stability. This thesis demonstrates one model trained for yeast. Some of the sequence determinants investigated in this thesis are likely conserved in

### 3 Discussion and Outlook

human. One example is the upstream AUG (uAUG) codon, which is frequent in human transcripts [73]. Variants creating or disrupting uAUG are likely to affect the stability of the corresponding transcript. Moreover, similar models can also be trained for humans. The improved experimental protocols have enabled better estimation of RNA kinetics in human [27, 122]. An end-to-end model predicting mRNA stability for human can be trained. Beside RNA kinetics data measured from reference cell lines, massively parallel reporter assays (MPRA), which was successfully applied in many other contexts [49, 123, 92], can be applied to study sequence determinants for human mRNA stability as well. Similar MPRA experiment has been successfully applied in zebrafish to study 3'UTR sequences that regulate mRNA decay [124].

We see from both the assay measurement and MMSplice prediction that, exonic variants can have a potentially large effects on splicing. Furthermore, changing codon usage also affect mRNA half-life. This leads to other interpretations of coding variants. Previously variants in the coding parts are mostly interpreted as missense or nonsynonymous depending on whether the encoded protein sequence changes. I recommend that future variant interpretation pipelines should also consider the effect of exonic variants on splicing and mRNA stability.

This thesis only considered variant effect on *cis*-elements, *trans* effects are ignored. One can imagine that, a variant disrupting an RBP-coding transcript is likely to affect many transcripts bound by the RBP. To build such a model, one would need at least two components. The first component predicts the variant effect on the RBP transcript (*cis*), the second component predicts effects on the targeted transcripts of the RBP (*trans*). Previously, such kind of effects are studied with expression quantitative trait loci (eQTL), which is a statistical association method. Depending on whether the effect is on *cis* (associations with the transcript harboring the variant) or in *trans* (associations with the targets regulated by the transcript harboring the variant), eQTL can be categorized into *cis*-eQTL or *trans*-eQTL. However, similar to GWAS, such statistical method fails to locate the causal variant. Machine learning models trained from perturbation data can be a better substitution with the increasing amount of data. For instance, one can systematically test *trans* effects by large scale CRISPR experiments. A recent study has proposed *crisprQTL* method combining CRISPR techniques with single-cell RNA sequencing. Instead of associating natural occurring variants across individuals, the method creates variants with CRISPR and test their effect across cells [125]. So far, *crisprQTL* is only used for *cis*-effects, extending it to study *trans*-effects should be possible.

## 3.1 Outlook

Here I point a few research directions that are interesting for the future work related to this thesis.



### 3.1.1 Functional characterization for the new half-life regulating motif ATATTC

We identified a new motif down-regulation mRNA half-life in yeast. The motif was validated for its effect but not its biological function. GO enrichment analysis indicates its potential function in respiration-related processes. Furthermore, it is interesting to find out the corresponding protein that binds to this motif. With the *trans*-regulating protein being identified, one can study its function by specifically knocking out the protein.

### 3.1.2 Tissue-specific splice variant effect prediction

Many exons are differentially alternative spliced across tissues [126]. Therefore, the same germline mutation may have different effects and consequences in various tissues. A tissue-specific variant effect prediction model for splicing could be highly relevant. The SPANR model was able to capture tissue-specific signals, but systematic benchmarking for tissue-specific prediction is lacking. Moreover, the public available scores computed with SPANR do not provide tissue-specific predictions, limiting its applications.

Large scale transcriptome profiling data is available [127], from which tissue-specific splicing models can be trained. One can potentially leverage on existing models like MMSplice and do transfer learning, or train a model from scratch to capture sequence determinants for the tissue-specificity.

### 3.1.3 Gene segmentation model for splice variant prediction

The current MMSplice model still needs annotation of exons to be applied. The model can be more powerful if we can make predictions without annotations. Here I present one potential approach for it.

An alternative way to view splicing is, it is a process to segment RNA transcripts in the cell. As the segmentation is mostly controlled by sequence elements on the RNA, it should be possible to train a model to approximate the decisions strategy of the cell. Semantic segmentation is a common task in computer vision, many well-established models exist and can be adapted to genomics. Such models typically follow an “encoder” and “decoder” architecture [128]. The “encoder” gradually transforms features into a more condensed but “deeper” space, and the “decoder” expand the information again to predict semantic information for every input pixel/character.

Such segmentation model for RNA can be trained by providing the following multi-class labels from the standard annotation: 5' UTR, exon, intron, and 3' UTR. Note that one nucleotide can be labeled as more than one class. Alternative splicing means ambiguity in the definition of exon or intron, which the model is trained to predict. Variants changing splicing patterns can be viewed as changing the “segmentation” of exon or introns, which is what the model is trained to predict for.

Such approaches have clear advantages: First, it needs annotation when training, but works annotation-independent when testing. It has the potential to capture cryptic or

weak splice sites [20] that most other models ignore. Second, it applies to all splicing patterns, including exon skipping, intron retention, alternative splice sites, etc.

#### 3.1.4 5' capping and 3' polyadenylation prediction

Other post-transcriptional processes can be modeled from the sequence as well. 5' capping and 3' polyadenylation are two important post-transcriptional processes, they define the transcript boundaries. Changing 5' capping or 3' polyadenylation sites can lead to different UTR length, and therefore different molecular consequences.

RNA-Seq data provide rich information about the capping and polyadenylation positions across tissues. They provide per base pair coverage profiles along the sequence. One can train a convolutional neural network or a recurrent neural network model to predict these profiles. The model should capture key tissue-specific signals of these biological processes, and can be potentially be applied for variant interpretation for these processes.

A recent MPRA experiment was done for human 3' polyadenylation process [123], the model, APARENT can precisely predict polyadenylation sites. One can do transfer learning from this model to adapt it for different tissues.

#### 3.1.5 Join transcriptional and post-transcriptional signals for variant interpretation

I show with MMSplice that improved functional impact prediction effectively improved variant pathogenicity prediction. The long term goal for (regulatory) variant interpretation should combine variant effect prediction for transcriptional and post-transcriptional processes. We already see the dual roles of some sequence elements, for instance, codon usage affects both protein translation and mRNA stability; abnormal transcript created by splice variants can trigger NMD; exonic variants that change protein coding sequence can also change splicing. Interpreting variants jointly for its effect on transcription, 5' capping, 3' polyadenylation, splicing, and degradation should lead to a more complete understanding of its impact. One recent progress in this regard is made by Zhou et al [129]. The study combined transcriptional and splicing signals, and highlighted the effect of noncoding mutations in autism spectrum disorder. Kipoi is another example of combining models from different modalities [4]. As a model repository, Kipoi consists of more than 2,000 models covering many transcriptional and post-transcriptional processes. The API of kipoi allows to jointly interpret variants with all models hosted in kipoi.

#### 3.1.6 Hierarchical models for variant pathogenicity prediction

Training a classifier to predict variant pathogenicity may have the danger to be biased due to the inherited bias from the training data. Current pathogenic variants are identified with the help of existing computational tools, therefore, the pathogenic annotations are biased to the variants that are “easy” for these tools. Classifiers trained with this

data will likely inherit these biases. Furthermore, new classifiers are often trained by ensemble of existing classifiers, exaggerating the biases even more.

Instead, more functional models should be encouraged. Models trained from targeted assay or perturbation data like MPRA are more likely to capture causal signals. Precisely, we should predict disease risks by building hierarchical from functional models that predict molecular phenotypes [130]. Instead of two model layers, I propose three: The first layer of models predicts fundamental DNA/RNA protein interactions, e.g., transcriptional factor binding, RBP binding. The second level of models predicts from these fundamental sequence protein interactions some specific biological process, e.g., transcription rate, splicing, RNA degradation. The final layer of models predicts disease risks by combining predictions from the previous two layers and additional features like conservation scores.

Such a hierarchical approach has three major advantages: First, it should be much easier to predict variant disease risks given their functional impact. Second, compared to variant pathogenicity annotation, functional data are much richer. An MPRA can simultaneously probe millions of sequences for their functional effects. Third, the model will suffer from less bias from the existing annotation.

An example of this kind of model is ExPecto [131]. ExPecto first trained a deep convolutional neural network to predict 2,002 different histone marks, transcription factor and DNA accessibility profiles across more than 200 tissues and cell types. Next, tissue-specific linear models were trained to predict tissue-specific expression levels. Finally, the model was applied to prioritize disease-causing variants.

In summary, regulatory variant interpretation can be improved from both experimental and computational sides. On the experimental side, more high throughput perturbation assays are needed to cover more regulatory processes and in more tissues. On the computational method side, more models are needed to predict from basic protein DNA/RNA binding to more complex gene expression processes, e.g., transcription, RNA degradation. Moreover, models are also needed to combine different models to make joint interpretations of the variant effect and predictions of pathogenicity. I foresee both sides to be significantly improved in the upcoming years with the rapid development of experimental techniques and end-to-end modeling frameworks.



# A Appendix

## Cis-regulatory elements explain most of the mRNA stability variation across genes in yeast

Regulating mRNA stability is an important way to regulate cellular mRNA concentration. The steady-state level of mRNA is jointly determined by the transcription rate and the degradation rate. Furthermore, mRNA stability is directly related to its dynamic response to environmental stimulus. Unstable mRNA can rapidly reach a new steady-state cellular level, while stable mRNA is robust to transcriptional perturbations. Therefore, mRNA half-life levels can vary as much as one to two orders of magnitude [132, 26, 27].

Sequence elements on mRNA play a major role in determining mRNA stability. Most RNA stability studies focused on the model organism *Saccharomyces cerevisiae* since the mechanism is conserved among eukaryotes. Previous studies have identified multiple sequence elements including but not limited to secondary structure, short motifs in 5' and 3' UTR, start and stop codon context and codon usage [23].

Many sequence determinants of mRNA stability were studied with reporter system, their genome-wide functions were often unclear. Besides, their overall contributions to the global half-life variations were obscure. Moreover, with the current list of features, we are unsure about how comprehensive we are.

In this study, I first evaluated known mRNA half-life determinants genome-wide. Second, I show evidence that metabolic labeling technique generates higher quality mRNA half-life data compared to other methods. Third, I used a linear mixed model to systematically discover 3' UTR motifs associated with mRNA half-life across genes. Fourth, I built a linear model with known and novel features to predict mRNA half-life from the primary mRNA sequence. Finally, to find the corresponding pathways, different sequence elements use to affect mRNA stability, I investigated 34 knockout strains, each with one key RNA degradation factor knocked out.

The analysis revealed one novel 3' UTR motif, ATATTC, that destabilizes mRNA. Genes harboring the motif are enriched with respiration-related function. The joint linear model was able to explain 59% of the total half-life variation across genes, and the median relative prediction error is 30%. Among the features used in the joint model, codon usage is the most predictive feature and is the primary determinant of mRNA half-life. Pathway analysis showed that codon usage affects mRNA stability through the canonical 5' to 3' mRNA degradation pathway mediated by Xrn1, instead of the previously speculated no-go-decay pathway. Furthermore, single-nucleotide variations (SNVs) on 3' UTR motifs and upstream AUG have the largest predicted effect on mRNA stability.

## A Appendix

Overall, this study provides a comprehensive analysis of sequence determinants of mRNA stability and their functional pathways. The predictive model revealed the dominant role of codon usage and also applicable for variant effect prediction.

This is the copyedited PDF of an article accepted for publication in RNA: Cheng, J., Maier, K. C., Avsec, Ž., Rus, P., & Gagneur, J. (2017). Cis-regulatory elements explain most of the mRNA stability variation across genes in yeast. *RNA*, 23(11), 1648-1659. DOI:10.1261/rna.062224.117

**License:** This is an open access article under the terms of the Creative Commons Attribution 4.0 License (CC BY 4.0), which permits use, distribution and reproduction in any medium or format, provided the original work is properly cited.

**Contribution of the thesis author:** model design and implementation, data analysis and visualization, literature review, results interpretation, manuscript composition, validation experiment design.

---

# Cis-regulatory elements explain most of the mRNA stability variation across genes in yeast

---

JUN CHENG,<sup>1,2</sup> KERSTIN C. MAIER,<sup>3</sup> ŽIGA AVSEC,<sup>1,2</sup> PETRA RUS,<sup>3</sup> and JULIEN GAGNEUR<sup>1,2</sup>

<sup>1</sup>Department of Informatics, Technical University of Munich, 85748 Garching, Germany

<sup>2</sup>Graduate School of Quantitative Biosciences (QBM), Ludwig-Maximilians-Universität München, 81377 München, Germany

<sup>3</sup>Department of Molecular Biology, Max Planck Institute for Biophysical Chemistry, 37077 Göttingen, Germany

## ABSTRACT

The stability of mRNA is one of the major determinants of gene expression. Although a wealth of sequence elements regulating mRNA stability has been described, their quantitative contributions to half-life are unknown. Here, we built a quantitative model for *Saccharomyces cerevisiae* based on functional mRNA sequence features that explains 59% of the half-life variation between genes and predicts half-life at a median relative error of 30%. The model revealed a new destabilizing 3' UTR motif, ATATTC, which we functionally validated. Codon usage proves to be the major determinant of mRNA stability. Nonetheless, single-nucleotide variations have the largest effect when occurring on 3' UTR motifs or upstream AUGs. Analyzing mRNA half-life data of 34 knockout strains showed that the effect of codon usage not only requires functional decapping and deadenylation, but also the 5'-to-3' exonuclease Xrn1, the nonsense-mediated decay genes, but not no-go decay. Altogether, this study quantitatively delineates the contributions of mRNA sequence features on stability in yeast, reveals their functional dependencies on degradation pathways, and allows accurate prediction of half-life from mRNA sequence.

**Keywords:** cis-regulatory elements; codon optimality; mRNA half-life

## INTRODUCTION

The stability of messenger RNAs is an important aspect of gene regulation. It influences the overall cellular mRNA concentration, as mRNA steady-state levels are the ratio of synthesis and degradation rate. Moreover, low stability confers high turnover to mRNA and, therefore, the capacity to rapidly reach a new steady-state level in response to a transcriptional trigger (Shalem et al. 2008). Hence, stress genes, which must rapidly respond to environmental signals, show low stability (Miller et al. 2011; Zeisel et al. 2011; Marguerat et al. 2014; Rabani et al. 2014). In contrast, high stability provides robustness to variations in transcription. Accordingly, a wide range of mRNA half-lives is observed in eukaryotes, with typical variations in a given genome spanning one to two orders of magnitude (Schwanhäusser et al. 2011; Eser et al. 2016; Schwab et al. 2016). Also, significant variability in mRNA half-life among human individuals could be demonstrated for about a quarter of genes in lymphoblastoid cells and estimated to account for more than a third of the gene expression variability (Duan et al. 2013).

How mRNA stability is encoded in a gene sequence has long been a subject of study. Cis-regulatory elements (CREs) affecting mRNA stability are mainly encoded in the

mRNA itself. Here we use the formal definition of CRE, i.e., a regulatory element affecting expression of the gene it belongs to in an allele-specific manner (Rockman and Kruglyak 2006; Skelly et al. 2009). CREs affecting mRNA stability include but are not limited to secondary structure (Rabani et al. 2008; Geisberg et al. 2014), sequence motifs present in the 3' UTR including binding sites of RNA-binding proteins (Olivas and Parker 2000; Duttagupta et al. 2005; Shalgi et al. 2005; Hogan et al. 2008; Hasan et al. 2014), and, in higher eukaryotes, microRNAs (Lee et al. 1993). Moreover, translation-related features are frequently associated with mRNA stability. For instance, inserting strong secondary structure elements in the 5' UTR or modifying the translation start codon context strongly destabilizes the long-lived *PGK1* mRNA in *S. cerevisiae* (Muhlrad et al. 1995; LaGrandeur and Parker 1999). Codon usage, which affects the translation elongation rate, also regulates mRNA stability (Hoekema et al. 1987; Pre-snyak et al. 2015; Bazzini et al. 2016; Mishima and Tomari 2016). Further correlations between codon usage and mRNA stability have been reported in *E. coli* and *S. pombe* (Boël et al. 2016; Harigaya and Parker 2016). Adjacent codon pairs were also demonstrated to associate with mRNA decay

---

Corresponding author: [gagneur@in.tum.de](mailto:gagneur@in.tum.de)

Article is online at <http://www.rnajournal.org/cgi/doi/10.1261/rna.062224.117>. Freely available online through the RNA Open Access option.

© 2017 Cheng et al. This article, published in *RNA*, is available under a Creative Commons License (Attribution-NonCommercial 4.0 International), as described at <http://creativecommons.org/licenses/by-nc/4.0/>.

in addition to individual codons in *S. cerevisiae* (Harigaya and Parker 2017).

Since the RNA degradation machineries are well conserved among eukaryotes, the pathways have been extensively studied using *S. cerevisiae* as a model organism (Garneau et al. 2007; Parker 2012). The general mRNA degradation pathway starts with the removal of the poly(A) tail by the Pan2/Pan3 (Brown et al. 1996) and Ccr4/Not complexes (Tucker et al. 2001). Subsequently, mRNA is subjected to decapping carried out by Dcp2 and promoted by several factors, including Dhh1 and Pat1 (Pilkington and Parker 2008; She et al. 2008). The decapped and deadenylated mRNA can be rapidly degraded in the 3' to 5' direction by the exosome (Anderson and Parker 1998) or in the 5' to 3' direction by Xrn1 (Hsu and Stevens 1993). Further mRNA degradation pathways are triggered when aberrant translational status is detected, including nonsense-mediated decay (NMD), no-go decay (NGD), and nonstop decay (NSD) (Garneau et al. 2007; Parker 2012).

Despite all this knowledge, prediction of mRNA half-life from a gene sequence is still not established. Moreover, most of the mechanistic studies so far were only performed on individual genes or reporter genes. It is therefore unclear how the measured effects generalize genome-wide. A recent study showed that translation-related features can be predictive for mRNA stability (Neymotin et al. 2016). Although this analysis supported the general correlation between translation and stability (Lackner et al. 2007), the model was not based purely on sequence-derived features. It also contained measured transcript properties such as ribosome density and normalized translation efficiencies. Hence, the question of how half-life is genetically encoded in mRNA sequence remains to be addressed.

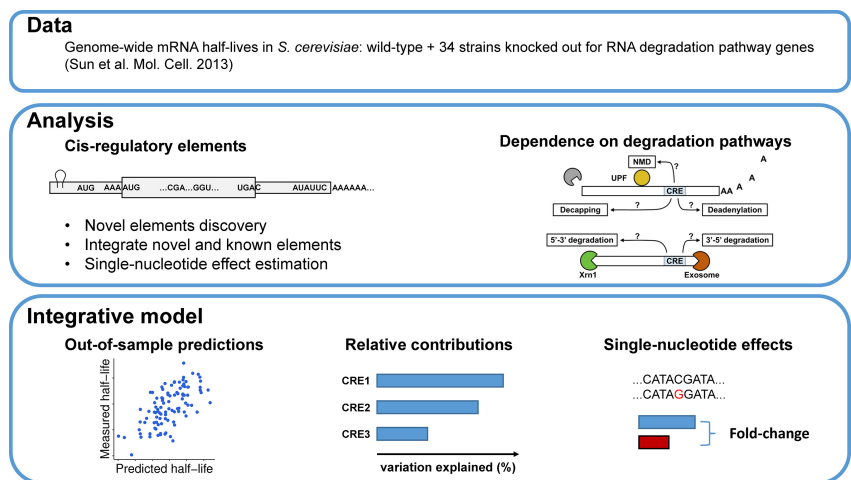
Additionally, the dependencies of sequence features to distinct mRNA degradation pathways have not been systematically studied. One example of this is codon-mediated stability control. Although a causal link from codon usage to mRNA half-life has been shown for a wide range of organisms (Hoekema et al. 1987; Presnyak et al. 2015; Bazzini et al. 2016; Mishima and Tomari 2016), the underlying mechanism remains poorly understood. In *S. cerevisiae*, reporter gene experiments showed that codon-mediated stability control depends on the RNA helicase Dhh1 (Radhakrishnan et al. 2016). However, it is unclear whether this generalizes to all mRNAs genome-wide. Also, the role of other closely related degradation path-

ways has not been systematically assessed with genome-wide half-life data.

Here, we mathematically modeled mRNA half-life as a function of its sequence. Applied to *S. cerevisiae*, our model can explain most of the between-gene half-life variance from sequence alone. Using a semimechanistic model, we could interpret individual sequence features in the 5' UTR, coding region, and 3' UTR. Quantification of the respective contributions revealed that codon usage is the major contributor to mRNA stability. Applying the modeling approach to *S. pombe* supports the generality of these findings. Moreover, we systematically assessed the dependencies of these sequence elements on mRNA degradation pathways using half-life data for 34 knockout strains. This analysis revealed in particular novel pathways through which codon usage affects half-life.

## RESULTS

To study *cis*-regulatory determinants of mRNA stability in *S. cerevisiae*, we chose the data set by Sun et al. (2013), which provides genome-wide half-life measurements for 4388 expressed genes of a wild-type laboratory strain and 34 strains knocked out for RNA degradation pathway genes (Fig. 1; Supplemental Table S1). When applicable, we also investigated half-life measurements of *S. pombe* for 3614 expressed mRNAs in a wild-type laboratory strain from Eser et al. (2016). We considered sequence features within five



**FIGURE 1.** Study overview. The goal of this study is to discover and integrate *cis*-regulatory mRNA elements affecting mRNA stability and assess their dependence on mRNA degradation pathways. (Data) We obtained *S. cerevisiae* genome-wide half-life data from wild-type (WT) as well as from 34 knockout strains from Sun et al. (2013). Each of the knockout strains has one gene closely related to mRNA degradation pathways knocked out. (Analysis) We systematically searched for novel sequence features associating with half-life from 5' UTR, start codon context, CDS, stop codon context, and 3' UTR. Effects of previously reported *cis*-regulatory elements were also assessed. Moreover, we assessed the dependencies of different sequence features on degradation pathways by analyzing their effects on the knockout strains. (Integrative model) We built a statistical model to predict genome-wide half-life solely from mRNA sequence. This allowed the quantification of the relative contributions of the sequence features to the overall variation across genes and assessing the sensitivity of mRNA stability with respect to single-nucleotide variants.



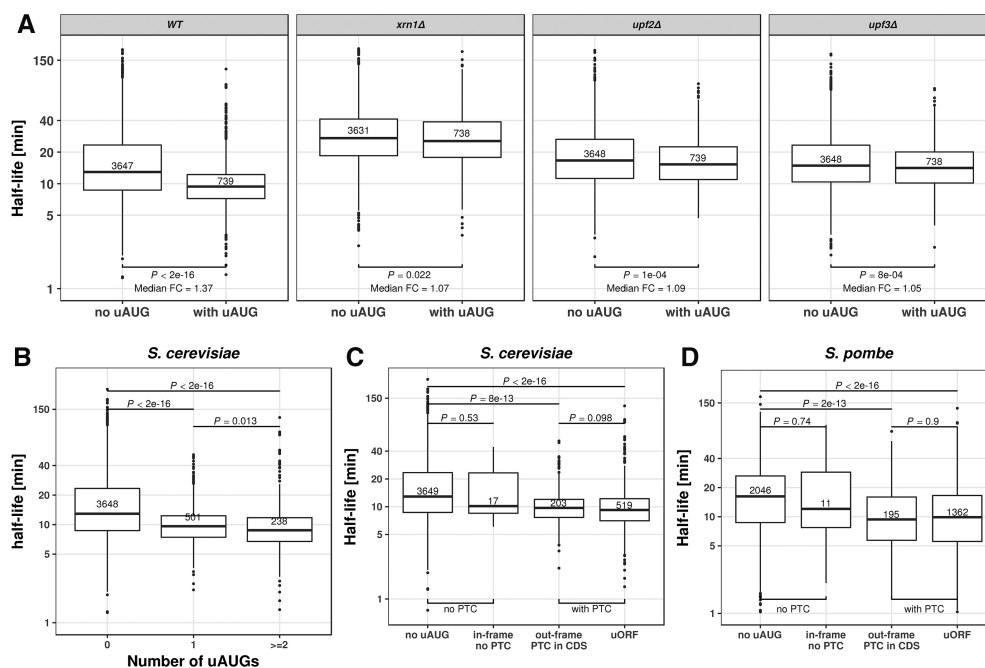
overlapping regions: the 5' UTR, the start codon context, the coding sequence, the stop codon context, and the 3' UTR. We assessed their effects in the wild type and in the 34 knockout strains (Fig. 1). Finally, we fitted a joint model to assess the contribution of individual sequence features and their single-nucleotide effects (Fig. 1). In all analyses, we considered the logarithm of half-life as the response variable rather than half-life in the natural scale. The primary motivation for choosing a logarithmic scale is that measurement noise for half-life is typically multiplicative. Also, the data did not provide supportive evidence discriminating between multiplicative or additive effects of the *cis*-regulatory elements on half-life (Supplemental Information). For simplicity, we used linear regressions, i.e., due to the logarithmic response, multiplicative models.

The correlations between sequence lengths, GC contents and folding energies (Materials and Methods) with half-life and corresponding *P*-values are summarized in Supplemental Table S2 and Supplemental Figures S1–S3. In general, sequence lengths correlated negatively with half-life and folding energies correlated positively with half-life in both yeast species, whereas correlations of GC content varied with species and gene regions.

In the following subsections, we describe first the findings for each of the five gene regions and then a model that integrates all these sequence features.

### Upstream AUGs destabilize mRNAs by triggering nonsense-mediated decay

Occurrence of an upstream AUG (uAUG) associated significantly with shorter half-life (median fold-change = 1.37,  $P < 2 \times 10^{-16}$ ). This effect was strengthened for genes with two or more AUGs (Fig. 2A,B). Among the 34 knock-out strains, the association between uAUG and shorter half-life was almost lost only for mutants of the two essential components of the nonsense-mediated mRNA decay (NMD) *UPF2* and *UPF3* (Leeds et al. 1992; Cui et al. 1995), and for the general 5' to 3' exonuclease *Xrn1* (Fig. 2A; Supplemental Fig. S6). The dependence on NMD suggested that the association might be due to the occurrence of a premature stop codon. Consistent with this hypothesis, the association of uAUG with decreased half-lives was only found for genes with a premature stop codon cognate with the uAUG (Fig. 2C). This held not only for cognate premature stop codons within the 5' UTR, leading to a potential upstream ORF, but also for cognate premature



**FIGURE 2.** Upstream AUG codons (uAUG) destabilize mRNA. (A) Distribution of mRNA half-lives for mRNAs without uAUG (*left*) and with at least one uAUG (*right*). From *left* to *right*: wild type, *XRN1*, *UPF2*, and *UPF3* knockout *S. cerevisiae* strains. Median fold-change (Median FC) calculated by dividing the median of the group without uAUG with the group with uAUG. A complete view of the effect of uAUG across different knockouts is provided in Supplemental Figure S6. (B) Distribution of mRNA half-lives for mRNAs with zero (*left*), one (*middle*), or more (*right*) uAUGs in *S. cerevisiae*. (C) Distribution of mRNA half-lives for *S. cerevisiae* mRNAs with, from *left* to *right*: no uAUG, with one in-frame uAUG but no cognate premature termination codon, with one out-of-frame uAUG and one cognate premature termination codon in the CDS, and with one uAUG and one cognate stop codon in the 5' UTR (uORF). (D) Same as in C for *S. pombe* mRNAs. All *P*-values were calculated with Wilcoxon rank-sum test. Numbers in the boxes indicate number of members in the corresponding group. Boxes represent quartiles, whiskers extend to the highest or lowest value within 1.5 times the interquartile range, and horizontal bars in the boxes represent medians. Data points falling further than 1.5-fold the interquartile distance are considered outliers and are shown as dots.



rather target a wide range of mRNAs, including aberrant and normal ones (He et al. 2003; Hug et al. 2015). In line with this, substrates of Upf proteins have lower codon optimality (Celik et al. 2017). Furthermore, we did not observe any change of effect upon knockout of *DOM34* and *HBS1* (Fig. 3B), which are essential genes for the No-Go decay pathway. This implies that the effect of codon usage is unlikely due to stalled ribosomes at nonoptimal codons.

Altogether, our analysis indicates that the so-called “codon-mediated decay” (Mishima and Tomari 2016) is not an mRNA decay pathway itself, but a regulatory mechanism of the common mRNA decay pathways.

### Stop codon context associates with mRNA stability

The first nucleotide 3' of the stop codon significantly associated with mRNA stability. This association was observed for each of the three possible stop codons, and for each codon a cytosine significantly associated with lower half-life (Supplemental Fig. S4, also for *P*-values and fold-changes). However, this feature was not significant in the joint model, and analysis of the knockout strains did not reveal clear pathway dependencies for it (Supplemental Fig. S6). A detailed description is provided in the Supplemental Information for interested readers.

### Sequence motifs in 3' UTR

De novo motif search identified four motifs in the 3' UTR to be significantly associated with mRNA stability (Fig. 4A, Materials and Methods). These include three described motifs: the Puf3 binding motif TGTAATA (FDR =  $3.2 \times 10^{-5}$ , median fold-change 1.29) (Gerber et al. 2004; Gupta et al. 2014), the Whi3 binding motif TGCAT (FDR =  $7 \times 10^{-4}$ , median fold-change 1.24) (Colomina et al. 2008; Cai and Futcher 2013), and a poly(U) motif TTTTSTA (FDR = 0.09, median fold-change 1.20), which can be bound by Pub1 (Duttagupta et al. 2005), or is part of the long poly(U) stretch that forms a looping structure with a poly(A) tail (Geisberg et al. 2014). Moreover, an uncharacterized motif, ATATTC, was associated with lower mRNA half-life (FDR =  $2 \times 10^{-5}$ , median fold-change 1.24). Genes harboring the ATATTC motif are significantly enriched for genes involved in oxidative phosphorylation (Bonferroni corrected  $P < 0.01$ , 4.4-fold enrichment, Gene Ontology analysis, Supplemental Methods; Supplemental Table S3). The motif ATATC preferentially localizes in the vicinity of the poly(A) site (Fig. 4B), and functionally depends on Ccr4 (FDR < 0.1, Supplemental Fig. S6), suggesting a potential interaction with deadenylation factors. Notably, the motif ATATTC was found in 13% of the genes (591 out of 4388) and significantly co-occurred with the other two destabilizing motifs found in 3' UTR: Puf3 motif (FDR = 0.01) and Whi3 motif (FDR =  $3 \times 10^{-3}$ ) binding motifs (Fig. 4F). This 3' UTR motif had been computationally identified by conservation analysis

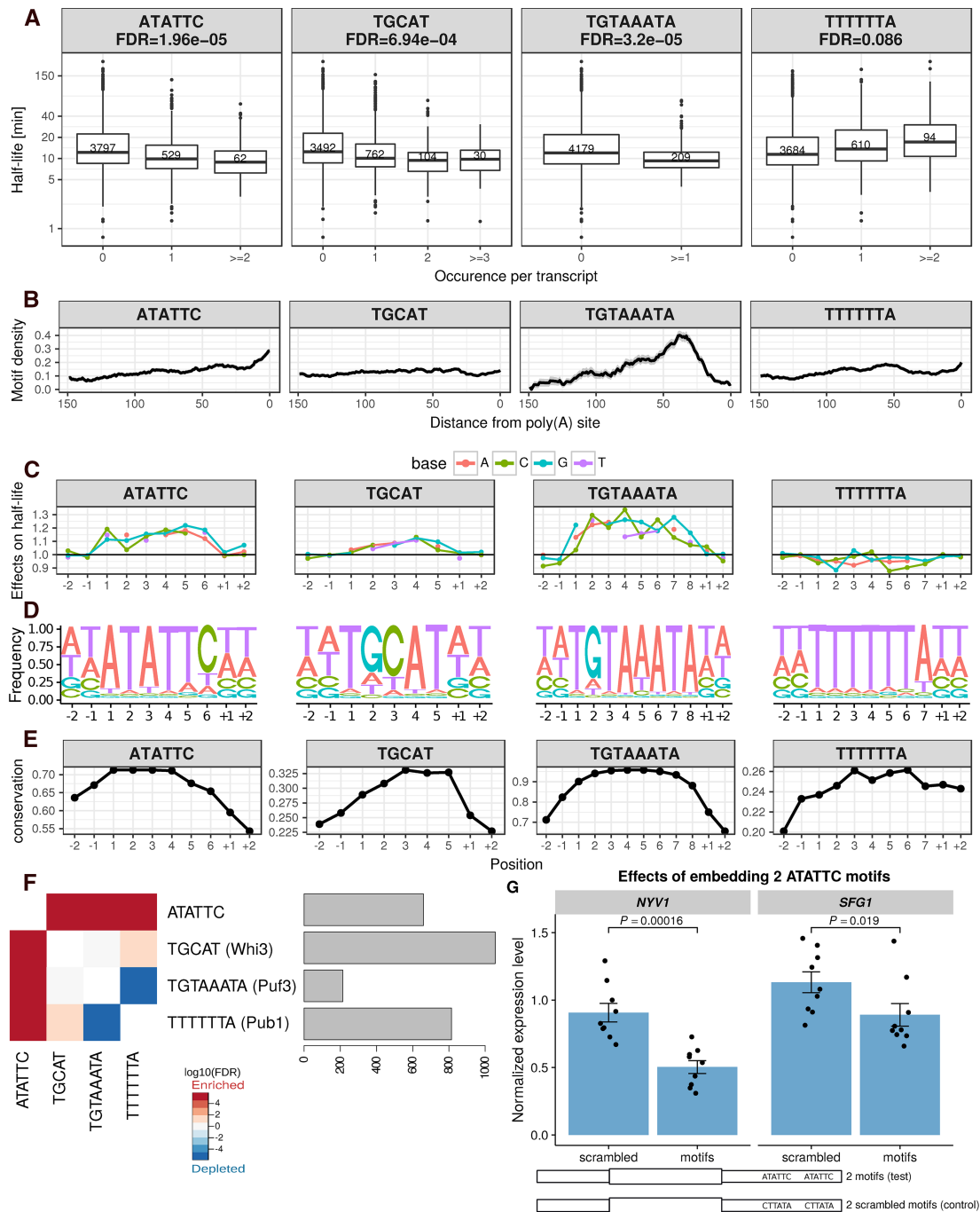
(Kellis et al. 2003), by regression of steady-state expression levels (Foat et al. 2005), and by enrichment analysis within gene expression clusters (Elemento et al. 2007). The motif was suggested to be named as PRSE (positive response to starvation element), because of its enrichment among genes that are up-regulated upon starvation (Foat et al. 2005). However, it was not experimentally validated for controlling of mRNA stability.

We validated the 3' UTR motif ATATTC with a reporter assay on two different genes, *SFG1* and *NYV1*. Given the predicted small effect of a single motif, we generated constructs with two instances of the motif and compared them to constructs harboring two scrambled motifs at the same locations (Fig. 4G, Materials and Methods). Both reporter genes showed decreased expression levels compared to scrambled controls ( $P = 0.019$  for *SFG1*,  $P = 0.00016$  for *NYV1*, Wilcoxon rank-sum test). Since the 3' UTR motif ATATTC is not significantly associated with mRNA synthesis rate ( $P = 0.38$ , Wilcoxon rank-sum test, synthesis rate of genes without motif versus genes with motif), we conclude that this decreased expression is due to decreased stability.

Consistent with the role of Puf3 in recruiting deadenylation factors, Puf3 binding motif localized preferentially close to the poly(A) site (Fig. 4B). The effect of the Puf3 motifs was significantly lower in the knockout of *PUF3* (FDR < 0.1, Supplemental Fig. S6). We also found a significant dependence on the deadenylation (*CCR4*, *POP2*) and decapping (*DHH1*, *PAT1*) pathways (all FDR < 0.1, Supplemental Fig. S6), consistent with previous single gene experiments showing that Puf3 binding promotes both deadenylation and decapping (Olivas and Parker 2000; Goldstrohm et al. 2007). Strikingly, the Puf3 binding motif switched to a stabilization motif in the absence of Puf3 and Ccr4 (all FDR < 0.1, Supplemental Fig. S6), suggesting that deadenylation of the Puf3 motif containing mRNAs is not only facilitated by Puf3 binding, but also depends on it.

Whi3 plays an important role in cell cycle control (Garí et al. 2001). Binding of Whi3 leads to destabilization of the *CLN3* mRNA (Cai and Futcher 2013). A subset of yeast genes are up-regulated in the Whi3 knockout strain (Cai and Futcher 2013). However, so far it was unclear whether Whi3 generally destabilizes mRNAs upon its binding. Our analysis showed that mRNAs containing the Whi3 binding motif (TGCAT) have a significantly shorter half-life (FDR =  $6.9 \times 10^{-4}$ , median fold-change 1.24). Surprisingly, this binding motif is extremely widespread, with 896 out of 4388 (20%) genes that we examined containing the motif on the 3' UTR region, which enriched for genes involved in several processes (Supplemental Table S3). Functionality of the Whi3 binding motif was found to be dependent on Ccr4 (FDR < 0.1, Supplemental Fig. S6).

The mRNAs harboring the TTTTSTA motif tended to be more stable (FDR = 0.086, median fold-change 1.22) and enriched for translation ( $P = 1.34 \times 10^{-3}$ , twofold enrichment; Supplemental Table S3). No positional preferences were



**FIGURE 4.** 3' UTR half-life determinant motifs in *S. cerevisiae*. (A) Distribution of half-lives for mRNAs grouped by the number of occurrence(s) of the motif ATATTC, TGCAT (Whi3), TGAAATA (Puf3), and TTTTTA (Pub1), respectively, in their 3' UTR sequence. Numbers in the boxes represent the number of members in each box. FDR were reported from the linear mixed effect model (Materials and Methods). (B) Fraction of transcripts containing the motif (y-axis) within a 20-bp window centered at a position (x-axis) with respect to poly(A) site for different motifs (facet titles). Positional bias was not observed when aligning 3' UTR motifs with respect to the stop codon. (C) Prediction of the relative effect on half-life (y-axis) for single-nucleotide substitution in the motif with respect to the consensus motif (y = 1, horizontal line). The motifs were extended two bases at each flanking site (positions +1, +2, -1, -2). (D) Nucleotide frequency within motif instances, when allowing for one mismatch compared with the consensus motif. (E) Mean conservation score (phastCons, Materials and Methods) of each base in the consensus motif with two flanking nucleotides (y-axis). (F) Co-occurrence significance (FDR, Fisher test *P*-value corrected with Benjamini-Hochberg) between different motifs (left). Number of occurrences among the 4388 mRNAs (right). (G) Steady-state expression level of *SFG1* and *NYV1* (normalized by *ACT1* and *TUB2* expression, Supplemental Methods). Bar height represents mean of each group, error bars represent  $\pm$  one standard error of the mean, each dot represents one biological replicate (jittered at x-axis to avoid overlapping). *P*-values were calculated by comparing the normalized expression level of constructs with two scrambled motifs embedded versus that with two functional ATATTC motifs embedded (Wilcoxon rank-sum test).

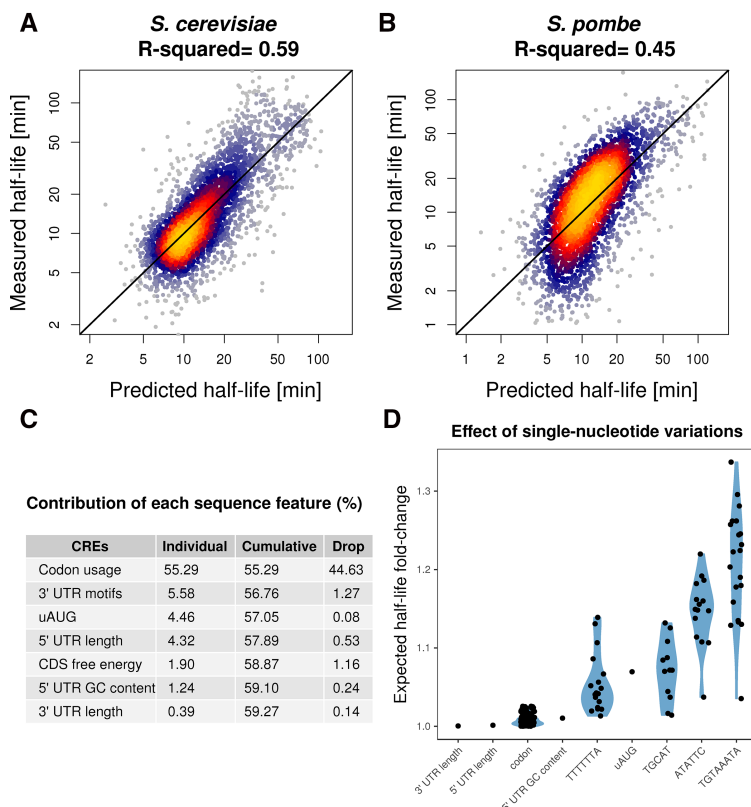


observed for this motif (Fig. 4B). The effect of this motif depends on genes from Ccr4–Not complex and Xrn1 (Supplemental Fig. S6).

An additional four lines of evidence further supported the functionality of our identified motifs. First, single-nucleotide deviations from the motif's consensus sequence associated with decreased effects on half-life (Fig. 4C, linear regression allowing for one mismatch, Materials and Methods). Moreover, the flanking nucleotides did not show further associations indicating that the whole lengths of the motifs were recovered (Fig. 4C). Second, when allowing for one mismatch, the motif still showed strong preferences (Fig. 4D). Third, the motif instances were more conserved than their flanking bases from the 3' UTR (Fig. 4E). Fourth, all four motifs show significant effects in the RNA half-life data set generated by Miller et al. (2011), which is also based on 4sU labeling, as well as in the data set of Presnyak et al. (2015), which is in contrast based on transcriptional arrest (Supplemental Fig. S7).

### Fifty-nine percent between-gene half-life variation can be explained by sequence features

We next asked how well one could predict mRNA half-life from these mRNA sequence features, and what their respective contributions were when considered jointly. To this end, we performed a multivariate linear regression of the logarithm of the half-life against the identified sequence features. The predictive power of the model on unseen data was assessed using 10-fold cross-validation (Materials and Methods; a complete list of model features and their *P*-values is provided in Supplemental Table S4). To prevent overfitting, we performed motif discovery on each of the 10 training sets and observed the same set of motifs across all the folds. Altogether, 59% of *S. cerevisiae* half-life variance in the logarithmic scale can be explained by simple linear combinations of the above sequence features (Fig. 5A; Supplemental Table S5). The median out-of-folds relative error across genes is 30%. A median relative error of 30% for half-life is remarkably low because it is in the order of magnitude of the expression variation that is typically physiologically tolerated, and it is also about the amount of variation observed between replicate experiments (Eser et al. 2016). To make sure that our



**FIGURE 5.** Genome-wide prediction of mRNA half-life from sequence features and analysis of the contributions. (A,B) mRNA half-life predicted (*x*-axis) versus measured (*y*-axis) for *S. cerevisiae* (A) and *S. pombe* (B), respectively. (C) Contribution of each sequence feature individually (*Individual*), cumulatively when sequentially added into a combined model (*Cumulative*), and explained variance drop when each single feature is removed from the full model separately (*Drop*). Values reported are the mean of 100 times of cross-validated evaluation (Materials and Methods). (D) Expected half-life fold-change of single-nucleotide variations on sequence features. For length and GC, dots represent median half-life fold-change of one nucleotide shorter or one G/C to A/T transition, respectively. For codon usage, each dot represents median half-life fold-change of one type of synonymous mutation; all kinds of synonymous mutations are considered. For uAUG, each dot represents median half-life fold-change of mutating out one uAUG. For motifs, each dot represents median half-life fold-change of one type of nucleotide transition at one position on the motif (Materials and Methods). Medians are calculated across all mRNAs.

findings are not biased to a specific data set, we fitted the same model to a data set using RATE-seq (Neymotin et al. 2014), a modified version of the protocol used by Sun et al. (2013). On these data, the model was able to explain 51% of the variance (Supplemental Fig. S8). Moreover, the same procedure applied to *S. pombe* explained 45% of the total half-life variance, suggesting the generality of this approach. Because the measures also entail measurement noise, these numbers are conservative underestimations of the total biological variance explained by our model.

The uAUG, 5' UTR length, 5' UTR GC content, 61 coding codons, CDS folding energy, all four 3' UTR motifs, and 3' UTR length remained significant in the joint model, indicating that they contributed individually to half-life (Supplemental Table S4). Most of them showed decreased effect in a joint model compared to marginal effects (Fig. 5C), likely

because they correlate with each other. In contrast, start codon context, stop codon context, 5' folding energy, the 5' UTR motif AACAAA (Supplemental Fig. S5), CDS length, and 3' UTR GC content dropped below the significance when considered in the joint model (Supplemental Table S4). This loss of statistical significance may be due to lack of statistical power. Another possibility is that the marginal association of these sequence features with half-life is a consequence of a correlation with other sequence features. Among all sequence features, codon usage as a group is the best predictor both in a univariate model (55.29%) and in the joint model (44.63 %) (Fig. 5C). This shows that, quantitatively, codon usage is the major determinant of mRNA stability in yeast. This explains why only a small fraction of mRNA stability variation can be explained by RNA-binding proteins (Hasan et al. 2014). The variance analysis quantifies the contribution of each sequence feature to the variation across genes. Features that vary a lot between genes, such as UTR length and codon usage, favorably contribute to the variation. However, this does not reflect the effect on a given gene of elementary sequence variations in these features. For instance, a single-nucleotide variant can lead to the creation of an uAUG with a strong effect on half-life, but a single-nucleotide variant in the coding sequence may have little impact on overall codon usage. We used the joint model to assess the sensitivity of each feature to single-nucleotide mutations as median fold-change across genes, simulating single-nucleotide deletions for the length features and single-nucleotide substitutions for the remaining ones (Materials and Methods). Single-nucleotide variations typically altered half-life by <10%. The largest effects were observed in the 3' UTR motifs and uAUG (Fig. 5D). Notably, although codon usage was the major contributor to the variance, synonymous variation on codons typically affected half-life by <2% (Fig. 5D; Supplemental Fig. S9). For those synonymous variations that changed half-life by more than 2%, most of them were variations that involved the most nonoptimized codons CGA or ATA (Supplemental Fig. S9; Presnyak et al. 2015).

Altogether, our results show that most of yeast mRNA half-life variation can be predicted from mRNA sequence alone, with codon usage being the major contributor. However, single-nucleotide variation at 3' UTR motifs or uAUG had the largest expected effect on mRNA stability.

## DISCUSSION

We systematically searched for mRNA sequence features associating with mRNA stability and estimated their effects at single-nucleotide resolution in a joint model. Up to GC content and length, all elements of the joint model are causal. One of them, the 3' UTR motif

ATATTC has been validated in this study. Overall, the joint model showed that 59% of the variance could be predicted from mRNA sequence alone in *S. cerevisiae*. This analysis showed that translation-related features, in particular codon usage, contributed most to the explained variance. This finding strengthens further the importance of the coupling between translation and mRNA degradation (Roy and Jacobson 2013; Huch and Nissan 2014; Radhakrishnan and Green 2016). Moreover, we assessed the dependencies of each sequence element on RNA degradation pathways. Remarkably, we identified that codon-mediated decay is a regulatory mechanism of the canonical decay pathways, including deadenylation- and decapping-dependent 5' to 3' decay and NMD (Figs. 3B, 6).

Predicting various steps of gene expression from sequence alone has long been a subject of study (Beer and Tavazoie 2004; Vogel et al. 2010; Zur and Tuller 2013; Wang et al. 2016). To this end, two distinct classes of models have been proposed: the biophysical models on the one hand and the machine learning models on the other hand (Zur and Tuller 2016). Biophysical models provide detailed understanding of the processes. On the other hand, machine learning approaches can reach much higher predictive accuracy but are more difficult to interpret. Also, machine learning approaches can pick up signals with predictive power that are correlative but not causal. Here we adopted an intermediate, semimechanistic modeling approach. We used a simple linear model that is interpretable. Also, all elements are functional, up to two covariates: GC content and length.

Our approach was based on the analysis of endogenous sequence, which allowed the identification of a novel *cis*-regulatory element. An alternative approach to the modeling of endogenous sequence is to use large-scale synthetic libraries (Dvir et al. 2013; Shalem et al. 2015; Wissink et al. 2016). Although very powerful to dissect known *cis*-regulatory elements or to investigate small variations around select genes, the sequence space is so large that these large-scale perturbation screens cannot uncover all regulatory motifs. It would be interesting to combine both approaches and design large-

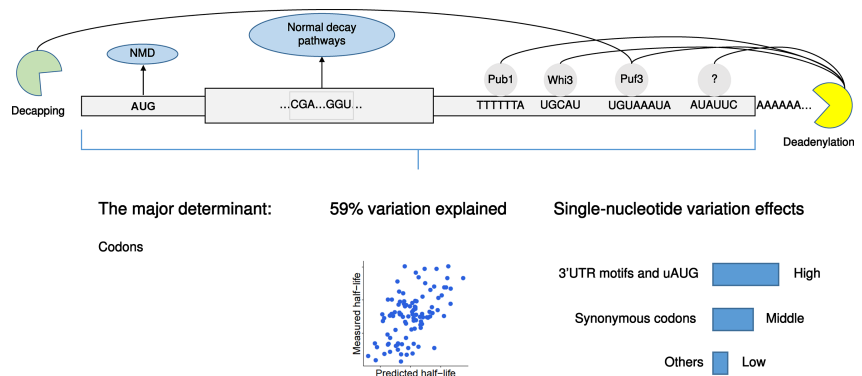


FIGURE 6. Overview and summary of conclusions from this study.

scale validation experiments guided by insights coming from modeling of endogenous sequences as we developed here.

Recently, Neymotin et al. (2016) showed that several translation-related transcript properties associated with half-life. This study derived a model explaining 50% of the total variance using many transcript properties including some not based on sequence (ribosome profiling, expression levels, etc.). Although non-sequence based predictors can facilitate prediction, they may do so because they are consequences rather than causes of half-life. For instance, increased half-life causes higher expression level. Also, increased cytoplasmic half-life, provides a higher ratio of cytoplasmic over nuclear RNA, and thus more RNAs available to ribosomes. Hence both expression level and ribosome density may help making good predictions of half-life, but not necessarily because they causally increase half-life. In contrast, we aimed here to understand how mRNA half-life is encoded in mRNA sequence and derived a model that is based on functional elements. This avoided using transcript properties that could be consequences of mRNA stability. Hence, our present analysis confirms the quantitative importance of translation in determining mRNA stability that Neymotin and colleagues quantified, and anchors it into pure sequence elements.

Confounding associations of sequence elements with mRNA stability could arise because of selection on expression levels acting at multiple stages of gene expression. For instance, genes that are selected for high protein expression levels may be enriched for elements that enhance translation and for elements that enhance mRNA stability. Functional validations are therefore needed to disentangle causality from co-selection. The sequence elements of our joint model, up to GC content and length, are all functional. However, we reported further elements that associate marginally with half-life. One of the interesting sequence elements that we found associated with half-life but did not turn out significant in the joint model is the start codon context. Given its established effect on translation initiation (Kozak 1986; Dvir et al. 2013), the general coupling between translation and mRNA degradation (Roy and Jacobson 2013; Huch and Nissan 2014; Radhakrishnan and Green 2016), as well as several observations directly on mRNA stability for single genes (LaGrandeur and Parker 1999; Schwartz and Parker 1999), the start codon context may nonetheless functionally affect mRNA stability. Consistent with this hypothesis, large-scale experiments that perturb 5' sequence secondary structure and start codon context indeed showed a wide range of mRNA level changes in the direction that we would predict (Dvir et al. 2013).

We are not aware of previous studies that systematically assessed the effects of *cis*-regulatory elements in the context of knockout backgrounds, as we did here. This part of our analysis turned out to be very insightful. By assessing the dependencies of codon usage mediated mRNA stability control systematically and comprehensively, we generalized results from recent studies on the Ccr4–Not complex and Dhh1, but also identified important novel ones including NMD fac-

tors, Pat1 and Xrn1. With the growing availability of knockout or mutant background in model organisms and human cell lines, we anticipate this approach to become a fruitful methodology to unravel regulatory mechanisms.

## MATERIALS AND METHODS

### Data and genomes

Wild-type and knockout genome-wide *S. cerevisiae* half-life data were obtained from Sun et al. (2013), whereby all strains are histidine, leucine, methionine, and uracil auxotrophs. A complete list of knockout strains used in this study is provided in Supplemental Table S1. *S. cerevisiae* gene boundaries were taken from the boundaries of the most abundant isoform quantified by Pelechano et al. (2013). Reference genome fasta file and genome annotation were obtained from the Ensembl database (release 79). UTR regions were defined by subtracting out gene body (exon and introns from the Ensembl annotation) from the gene boundaries. Processed *S. cerevisiae* UTR annotation is provided in Supplemental Table S6.

Genome-wide half-life data of *S. pombe* as well as refined transcription unit annotation were obtained from Eser et al. (2016). Reference genome version ASM294v2.26 was used to obtain sequence information. Half-life outliers of *S. pombe* (half-life less than 1 or larger than 250 min) were removed.

For both half-life data sets, only mRNAs with mapped 5' UTR and 3' UTR were considered. mRNAs with 5' UTR length shorter than 6 nt were further filtered out.

Codon-wise species-specific tRNA adaptation index (sTAI) of yeasts were obtained from Sabi and Tuller (2014). Gene-wise sTAIs were calculated as the geometric mean of sTAIs of all its codons (stop codon excluded).

### Analysis of knockout strains

The effect level of an individual sequence feature was compared against the wild-type with Wilcoxon rank-sum test followed by multiple hypothesis testing *P*-value correction (FDR < 0.1). For details, see Supplemental Methods.

### Motif discovery

Motif discovery was conducted for the 5' UTR, the CDS and the 3' UTR regions. A linear mixed effect model was used to assess the effect of each individual *k*-mer while controlling the effects of the others and for the region length as a covariate as described previously (Eser et al. 2016). For CDS we also used codons as further covariates. In contrast to Eser and colleagues, we tested the effects of all possible *k*-mers with lengths from 3 to 8. The linear mixed model for motif discovery was fitted with GEMMA software (Zhou et al. 2013). *P*-values were corrected for multiple testing using Benjamini–Hochberg's FDR. Motifs were subsequently manually assembled based on overlapping significant (FDR < 0.1) *k*-mers.

### Folding energy calculation

RNA sequence folding energy was calculated with RNAfold from ViennaRNA version 2.1.9 (Lorenz et al. 2011), with default parameters.

## S. cerevisiae conservation analysis

The phastCons (Siepel et al. 2005) conservation track for *S. cerevisiae* was downloaded from the UCSC Genome Browser (<http://hgdownload.cse.ucsc.edu/goldenPath/sacCer3/phastCons7way/>). Motif single-nucleotide level conservation scores were computed as the mean conservation score of each nucleotide (including two extended nucleotides at each side of the motif) across all motif instances genome-wide (removing NA values).

## Linear regression model for codon usage

Throughout the study, we modeled codon usage in the linear model with each codon as an independent covariate using its frequency.

$$\log(y_g) = \beta_0 + \sum_{c \in \text{Codons}} \beta_c x_c + \varepsilon_g, \quad (1)$$

where  $x_c = \frac{n_c}{L_g}$ ,  $n_c$  is the number of codon  $c$  in gene  $g$ ,  $L_g$  is the CDS length of gene  $g$ .

## Relation between codon regression coefficient and sTAI

The coefficients of codon frequencies have an analogous interpretation as species-specific tRNA adaptation index (sTAI). The same applies also to tAI. The sTAI of a gene is defined as the geometric mean of the sTAIs of all its coding codons (Sabi and Tuller 2014). For a gene  $g$  with  $N$  number of codons, its sTAI is defined as follows:

$$sTAI_g = \left( \prod_{i=1}^N w_i \right)^{\frac{1}{N}} = \sqrt[N]{w_1 w_2 \dots w_N}, \quad (2)$$

where  $w_i$  represent the sTAI of the  $i_{th}$  codon in the gene.

The logarithm of a gene sTAI with  $N$  codons is

$$\begin{aligned} \log(sTAI_g) &= \frac{1}{N} \left( \sum_{i=1}^N \log(w_i) \right) = \sum_{c \in \text{Codons}} 3 \log(w_c) \frac{n_c}{3N} \\ &= \sum_{c \in \text{Codons}} 3 \log(w_c) x_c, \end{aligned} \quad (3)$$

where  $x_c$  is defined in Equation 1,  $3N = L_g$  is the CDS length,  $n_c$  is the number of codon  $c$  in gene  $g$ ,  $w_c$  is the sTAI of codon  $c$ . Hence, in a linear model the regression coefficient  $\beta_c$  of Equation 1 has an analogous interpretation to the log of sTAI [ $\log(w_c)$ ].

## Linear model for genome-wide half-life prediction

Multivariate linear regression models were used to predict genome-wide mRNA half-life on the logarithmic scale from sequence features. Only mRNAs that contain all features were used to fit the models, resulting in 3838 mRNAs for *S. cerevisiae* and 3360 mRNAs for *S. pombe*. Out-of-fold predictions were applied with 10-fold cross validation for any prediction task in this study. For each fold, a linear model was first fitted to the training data with all sequence features as covariates, then a stepwise model selection procedure was applied to select the best model with Bayesian Information Criterion as criteria [*step* function in *R*, with  $k = \log$

( $n$ )]. L1 or L2 regularization was not necessary, as they did not improve the out-of-fold prediction accuracy (tested with the glmnet *R* package [Friedman et al. 2010]). Motif discovery was performed again at each fold. The same set of motifs was identified within each training set only. For details, see Supplemental Methods.

## Analysis of sequence feature contribution

Linear models were first fitted on the complete data with all sequence features as covariates, nonsignificant sequence features were then removed from the final models, ending up with 69 features for the *S. cerevisiae* model and 76 features for *S. pombe* (each single-coding codon was fitted as a single covariate). The contribution of each sequence feature was analyzed individually as a univariate regression and also jointly in a multivariate regression model. The contribution of each feature *individually* was calculated as the variance explained by a univariate model. Features were then added in a descending order of their individual explained variance to a joint model; “cumulative” variances explained were then calculated. The “drop” quantifies the drop of variance explained as leaving out one feature separately from the full model. All contribution statistics were quantified by taking the average of 100 times of 10-fold cross-validation.

## Single-nucleotide variant effect predictions

The same model used in sequence feature contribution analysis was used for single-nucleotide variant effect prediction. For motifs, effects of single-nucleotide variants were predicted with the linear model modified from Eser et al. (2016). When assessing the effect of a given motif variation, instead of estimating the marginal effect size, we controlled for the effect of all other sequence features using a linear model with the other features as covariates. For details, see Supplemental Methods. For other sequence features, effects of single-nucleotide variants were predicted by introducing a single-nucleotide perturbation into the full prediction model for each gene, and summarizing the effect with the median half-life change across all genes. For details, see Supplemental Methods.

## Construction of SFG1 and NYV1 mutant strains

One hundred base pair primers (IDT) containing the respective 3' UTR mutations were used to amplify the kanMX cassette from plasmid pFA6a-KanMX6 (Euroscarf). PCR products were used for transformation of strain BY4741 (MATa his3Δ1 leu2Δ0 met15Δ0 ura3Δ0, Euroscarf) by homologous recombination, and transformants were selected on G418 plates. Correct clones were confirmed by sequencing. Details of the reporter assay design are provided in the Supplemental Methods. Sequences of the constructs are given in Supplemental Table S7.

## Quantitative PCR

Cells were grown to OD<sub>600</sub> 0.8 in YPD from overnight cultures inoculated from single colonies. Cells were centrifuged at 4000 rpm for 1 min at 30°C and pellets were flash-frozen in liquid nitrogen. RNA was phenol/chloroform purified. cDNA synthesis was performed with 1.5 μg RNA using the Maxima Reverse Transcriptase



(Thermo Fisher). qPCR was performed on a qTower 2.2 (Analytik Jena) using a 2-min denaturing step at 95°C, followed by 39 cycles of 5 sec at 95°C, 10 sec at 64°C, and 15 sec at 72°C with a final step at 72°C for 5 min. qPCR was performed using the SensiFAST SYBR No-ROX Kit (Bioline). Primer efficiencies were determined by performing standard curves for all primer combinations. All primer pairs had efficiencies of 95% or higher. Sequence information of primer pairs and efficiencies are provided in Supplemental Table S7. Ct data from nine biological and three technical replicates were used for analysis. Details of analyzing qPCR data are described in Supplemental Methods.

## DATA DEPOSITION

Analysis scripts are available at [https://github.com/gagneurlab/Manuscript\\_Cheng\\_RNA\\_2017](https://github.com/gagneurlab/Manuscript_Cheng_RNA_2017).

## SUPPLEMENTAL MATERIAL

Supplemental material is available for this article.

## ACKNOWLEDGMENTS

We thank Patrick Cramer for supporting the motif validation experiment. We thank Fabien Bonneau (Max Planck Institute of Biochemistry) for helpful discussions on motifs and RNA degradation pathways, as well as useful feedback on the manuscript. We thank Björn Schwalb for communication on analyzing the knockout data. We thank Vicente Yépez for useful feedback on the manuscript and Patrick Cramer for institutional support. J.C. and Ž.A. are supported by a Deutsche Forschungsgemeinschaft fellowship through QBM.

Received May 24, 2017; accepted July 31, 2017.

## REFERENCES

- Anderson JS, Parker RP. 1998. The 3' to 5' degradation of yeast mRNAs is a general mechanism for mRNA turnover that requires the SKI2 DEVH box protein and 3' to 5' exonucleases of the exosome complex. *EMBO J* **17**: 1497–1506.
- Bazzini AA, Del Viso F, Moreno-Mateos MA, Johnstone TG, Vejnar CE, Qin Y, Yao J, Khokha MK, Giraldez AJ. 2016. Codon identity regulates mRNA stability and translation efficiency during the maternal-to-zygotic transition. *EMBO J* **35**: 2087–2103.
- Beer MA, Tavazoie S. 2004. Predicting gene expression from sequence. *Cell* **117**: 185–198.
- Boël G, Letso R, Neely H, Price WN, Wong K, Su M, Luff JD, Valecha M, Everett JK, Acton TB, et al. 2016. Codon influence on protein expression in *E. coli* correlates with mRNA levels. *Nature* **529**: 358–363.
- Brown CE, Tarun SZ Jr, Boeck R, Sachs AB. 1996. PAN3 encodes a subunit of the Pab1p-dependent poly(A) nuclease in *Saccharomyces cerevisiae*. *Mol Cell Biol* **16**: 5744–5753.
- Cai Y, Futcher B. 2013. Effects of the yeast RNA-binding protein Whi3 on the half-life and abundance of CLN3 mRNA and other targets. *PLoS One* **8**: e84630.
- Celik A, Baker R, He F, Jacobson A. 2017. High-resolution profiling of NMD targets in yeast reveals translational fidelity as a basis for substrate selection. *RNA* **23**: 735–748.
- Colomina N, Ferrezuelo F, Wang H, Aldea M, Garí E. 2008. Whi3, a developmental regulator of budding yeast, binds a large set of mRNAs functionally related to the endoplasmic reticulum. *J Biol Chem* **283**: 28670–28679.
- Cui Y, Hagan KW, Zhang S, Peltz SW. 1995. Identification and characterization of genes that are required for the accelerated degradation of mRNAs containing a premature translational termination codon. *Genes Dev* **9**: 423–436.
- Duan J, Shi J, Ge X, Dölken L, Moy W, He D, Shi S, Sanders AR, Ross J, Gejman PV. 2013. Genome-wide survey of interindividual differences of RNA stability in human lymphoblastoid cell lines. *Sci Rep* **3**: 1318.
- Duttagupta R, Tian B, Wilusz CJ, Khounh DT, Soteropoulos P, Ouyang M, Dougherty JP, Peltz SW. 2005. Global analysis of Pub1p targets reveals a coordinate control of gene expression through modulation of binding and stability. *Mol Cell Biol* **25**: 5499–5513.
- Dvir S, Velten L, Sharon E, Zeevi D, Carey LB, Weinberger A, Segal E. 2013. Deciphering the rules by which 5'-UTR sequences affect protein expression in yeast. *Proc Natl Acad Sci* **110**: E2792–E2801.
- Elemento O, Slonim N, Tavazoie S. 2007. A universal framework for regulatory element discovery across all genomes and data types. *Mol Cell* **28**: 337–350.
- Eser P, Wachutka L, Maier KC, Demel C, Boroni M, Iyer S, Cramer P, Gagneur J. 2016. Determinants of RNA metabolism in the *Schizosaccharomyces pombe* genome. *Mol Syst Biol* **12**: 857.
- Foat BC, Houshmandi SS, Olivas WM, Bussemaker HJ. 2005. Profiling condition-specific, genome-wide regulation of mRNA stability in yeast. *Proc Natl Acad Sci* **102**: 17675–17680.
- Friedman J, Hastie T, Tibshirani R. 2010. Regularization paths for generalized linear models via coordinate descent. *J Stat Softw* **33**: 1–22.
- Gaba A, Jacobson A, Sachs MS. 2005. Ribosome occupancy of the yeast CPA1 upstream open reading frame termination codon modulates nonsense-mediated mRNA decay. *Mol Cell* **20**: 449–460.
- Garí E, Volpe T, Wang H, Gallego C, Futcher B, Aldea M. 2001. Whi3 binds the mRNA of the G1 cyclin CLN3 to modulate cell fate in budding yeast. *Genes Dev* **15**: 2803–2808.
- Garneau NL, Wilusz J, Wilusz CJ. 2007. The highways and byways of mRNA decay. *Nat Rev Mol Cell Biol* **8**: 113–126.
- Geisberg JV, Moqtaderi Z, Fan X, Oszolak F, Struhl K. 2014. Global analysis of mRNA isoform half-lives reveals stabilizing and destabilizing elements in yeast. *Cell* **156**: 812–824.
- Gerber AP, Herschlag D, Brown PO. 2004. Extensive association of functionally and cytotopically related mRNAs with Puf family RNA-binding proteins in yeast. *PLoS Biol* **2**: E79.
- Goldstrohm AC, Seay DJ, Hook BA, Wickens M. 2007. PUF protein-mediated deadenylation is catalyzed by Ccr4p. *J Biol Chem* **282**: 109–114.
- Gupta I, Clauder-Münster S, Klaus B, Järvelin AI, Aiyar RS, Benes V, Wilkening S, Huber W, Pelechano V, Steinmetz LM. 2014. Alternative polyadenylation diversifies post-transcriptional regulation by selective RNA-protein interactions. *Mol Syst Biol* **10**: 719.
- Harigaya Y, Parker R. 2016. Analysis of the association between codon optimality and mRNA stability in *Schizosaccharomyces pombe*. *BMC Genomics* **17**: 895.
- Harigaya Y, Parker R. 2017. The link between adjacent codon pairs and mRNA stability. *BMC Genomics* **18**: 364.
- Hasan A, Cotobal C, Duncan CDS, Mata J. 2014. Systematic analysis of the role of RNA-binding proteins in the regulation of RNA stability. *PLoS Genet* **10**: e1004684.
- He F, Li X, Spatrick P, Casillo R, Dong S, Jacobson A. 2003. Genome-wide analysis of mRNAs regulated by the nonsense-mediated and 5' to 3' mRNA decay pathways in yeast. *Mol Cell* **12**: 1439–1452.
- Hoekema A, Kastelein RA, Vasser M, de Boer HA. 1987. Codon replacement in the PGK1 gene of *Saccharomyces cerevisiae*: experimental approach to study the role of biased codon usage in gene expression. *Mol Cell Biol* **7**: 2914–2924.
- Hogan DJ, Riordan DP, Gerber AP, Herschlag D, Brown PO. 2008. Diverse RNA-binding proteins interact with functionally related sets of RNAs, suggesting an extensive regulatory system. *PLoS Biol* **6**: e255.

- Hsu CL, Stevens A. 1993. Yeast cells lacking 5'-3' exoribonuclease 1 contain mRNA species that are poly(A) deficient and partially lack the 5' cap structure. *Mol Cell Biol* **13**: 4826–4835.
- Huch S, Nissan T. 2014. Interrelations between translation and general mRNA degradation in yeast. *Wiley Interdiscip Rev RNA* **5**: 747–763.
- Hug N, Longman D, Cáceres JF. 2015. Mechanism and regulation of the nonsense-mediated decay pathway. *Nucleic Acids Res* **44**: 1483–1495.
- Kellis M, Patterson N, Endrizzi M, Birren B, Lander ES. 2003. Sequencing and comparison of yeast species to identify genes and regulatory elements. *Nature* **423**: 241–254.
- Kozak M. 1986. Point mutations define a sequence flanking the AUG initiator codon that modulates translation by eukaryotic ribosomes. *Cell* **44**: 283–292.
- Lackner DH, Beilharz TH, Marguerat S, Mata J, Watt S, Schubert F, Preiss T, Bähler J. 2007. A network of multiple regulatory layers shapes gene expression in fission yeast. *Mol Cell* **26**: 145–155.
- LaGrandeur T, Parker R. 1999. The *cis* acting sequences responsible for the differential decay of the unstable MFA2 and stable PGK1 transcripts in yeast include the context of the translational start codon. *RNA* **5**: 420–433.
- Lee RC, Feinbaum RL, Ambros V. 1993. The *C. elegans* heterochronic gene *lin-4* encodes small RNAs with antisense complementarity to *lin-14*. *Cell* **75**: 843–854.
- Leeds P, Wood JM, Lee B, Culbertson MR. 1992. Gene products that promote mRNA turnover in *Saccharomyces cerevisiae*. *Mol Cell Biol* **12**: 2165–2177.
- Lorenz R, Bernhart SH, Höner Zu Siederdisen C, Tafer H, Flamm C, Stadler PF, Hofacker IL. 2011. ViennaRNA Package 2.0. *Algorithms Mol Biol* **6**: 26.
- Marguerat S, Lawler K, Brazma A, Bähler J. 2014. Contributions of transcription and mRNA decay to gene expression dynamics of fission yeast in response to oxidative stress. *RNA Biol* **11**: 702–714.
- Miller C, Schwalb B, Maier K, Schulz D, Dümcke S, Zacher B, Mayer A, Sydow J, Marcinowski L, Dölken L, et al. 2011. Dynamic transcriptome analysis measures rates of mRNA synthesis and decay in yeast. *Mol Syst Biol* **7**: 458.
- Mishima Y, Tomari Y. 2016. Codon usage and 3' UTR length determine maternal mRNA stability in zebrafish. *Mol Cell* **61**: 874–885.
- Muhlrad D, Decker CJ, Parker R. 1995. Turnover mechanisms of the stable yeast PGK1 mRNA. *Mol Cell Biol* **15**: 2145–2156.
- Neymotin B, Athanasiadou R, Gresham D. 2014. Determination of in vivo RNA kinetics using RATE-seq. *RNA* **20**: 1645–1652.
- Neymotin B, Ettore V, Gresham D. 2016. Multiple transcript properties related to translation affect mRNA degradation rates in *Saccharomyces cerevisiae*. *G3 (Bethesda)* **6**: 3475–3483.
- Olivas W, Parker R. 2000. The Puf3 protein is a transcript-specific regulator of mRNA degradation in yeast. *EMBO J* **19**: 6602–6611.
- Parker R. 2012. RNA degradation in *Saccharomyces cerevisiae*. *Genetics* **191**: 671–702.
- Pelechano V, Wei W, Steinmetz LM. 2013. Extensive transcriptional heterogeneity revealed by isoform profiling. *Nature* **497**: 127–131.
- Pilkington GR, Parker R. 2008. Pat1 contains distinct functional domains that promote P-body assembly and activation of decapping. *Mol Cell Biol* **28**: 1298–1312.
- Presnyak V, Alhusaini N, Chen Y-H, Martin S, Morris N, Kline N, Olson S, Weinberg D, Baker KE, Graveley BR, et al. 2015. Codon optimality is a major determinant of mRNA stability. *Cell* **160**: 1111–1124.
- Rabani M, Kertesz M, Segal E. 2008. Computational prediction of RNA structural motifs involved in posttranscriptional regulatory processes. *Proc Natl Acad Sci* **105**: 14885–14890.
- Rabani M, Raychowdhury R, Jovanovic M, Rooney M, Stumpo DJ, Pauli A, Hacohen N, Schier AF, Blackshear PJ, Friedman N, et al. 2014. High-resolution sequencing and modeling identifies distinct dynamic RNA regulatory strategies. *Cell* **159**: 1698–1710.
- Radhakrishnan A, Green R. 2016. Connections underlying translation and mRNA stability. *J Mol Biol* **428**: 3558–3564.
- Radhakrishnan A, Chen Y-H, Martin S, Alhusaini N, Green R, Collier J. 2016. The DEAD-box protein Dhh1p couples mRNA decay and translation by monitoring codon optimality. *Cell* **167**: 122–132.e9.
- Rockman MV, Kruglyak L. 2006. Genetics of global gene expression. *Nat Rev Genet* **7**: 862–872.
- Roy B, Jacobson A. 2013. The intimate relationships of mRNA decay and translation. *Trends Genet* **29**: 691–699.
- Sabi R, Tuller T. 2014. Modelling the efficiency of codon-tRNA interactions based on codon usage bias. *DNA Res* **21**: 511–526.
- Schwalb B, Michel M, Zacher B, Frühauf K, Demel C, Tresch A, Gagneur J, Cramer P. 2016. TT-seq maps the human transient transcriptome. *Science* **352**: 1225–1228.
- Schwahnhauser B, Busse D, Li N, Dittmar G, Schuchhardt J, Wolf J, Wei C, Selbach M. 2011. Global quantification of mammalian gene expression control. *Nature* **473**: 337–342.
- Schwartz DC, Parker R. 1999. Mutations in translation initiation factors lead to increased rates of deadenylation and decapping of mRNAs in *Saccharomyces cerevisiae*. *Mol Cell Biol* **19**: 5247–5256.
- Shalem O, Dahan O, Levo M, Martinez MR, Furman I, Segal E, Pilpel Y. 2008. Transient transcriptional responses to stress are generated by opposing effects of mRNA production and degradation. *Mol Syst Biol* **4**: 223.
- Shalem O, Sharon E, Lubliner S, Regev I, Lotan-Pompan M, Yakhini Z, Segal E. 2015. Systematic dissection of the sequence determinants of gene 3' end mediated expression control. *PLoS Genet* **11**: e1005147.
- Shalgi R, Lapidot M, Shamir R, Pilpel Y. 2005. A catalog of stability-associated sequence elements in 3' UTRs of yeast mRNAs. *Genome Biol* **6**: R86.
- She M, Decker CJ, Svergun DI, Round A, Chen N, Muhlrad D, Parker R, Song H. 2008. Structural basis of Dcp2 recognition and activation by Dcp1. *Mol Cell* **29**: 337–349.
- Siepel A, Bejerano G, Pedersen JS, Hinrichs AS, Hou M, Rosenbloom K, Clawson H, Spieth J, Hillier LW, Richards S, et al. 2005. Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. *Genome Res* **15**: 1034–1050.
- Skelly DA, Ronald J, Akey JM. 2009. Inherited variation in gene expression. *Annu Rev Genomics Hum Genet* **10**: 313–332.
- Sun M, Schwalb B, Pirkl N, Maier KC, Schenk A, Failmezger H, Tresch A, Cramer P. 2013. Global analysis of eukaryotic mRNA degradation reveals Xrn1-dependent buffering of transcript levels. *Mol Cell* **52**: 52–62.
- Tucker M, Valencia-Sanchez MA, Staples RR, Chen J, Denis CL, Parker R. 2001. The transcription factor associated Ccr4 and Caf1 proteins are components of the major cytoplasmic mRNA deadenylase in *Saccharomyces cerevisiae*. *Cell* **104**: 377–386.
- Vogel C, Abreu R de S, Ko D, Le SYY, Shapiro BA, Burns SC, Sandhu D, Boutz DR, Marcotte EM, Penalva LO. 2010. Sequence signatures and mRNA concentration can explain two-thirds of protein abundance variation in a human cell line. *Mol Syst Biol* **6**: 400.
- Wang X, Hou J, Quedenau C, Chen W. 2016. Pervasive isoform-specific translational regulation via alternative transcription start sites in mammals. *Mol Syst Biol* **12**: 875.
- Wissink EM, Fogarty EA, Grimson A. 2016. High-throughput discovery of post-transcriptional *cis*-regulatory elements. *BMC Genomics* **17**: 177.
- Zeisel A, Köstler WJ, Molotski N, Tsai JM, Krauthgamer R, Jacob-Hirsch J, Rechavi G, Soen Y, Jung S, Yarden Y, et al. 2011. Coupled pre-mRNA and mRNA dynamics unveil operational strategies underlying transcriptional responses to stimuli. *Mol Syst Biol* **7**: 529.
- Zhou X, Carbonetto P, Stephens M. 2013. Polygenic modeling with Bayesian sparse linear mixed models. *PLoS Genet* **9**: e1003264.
- Zur H, Tuller T. 2013. Transcript features alone enable accurate prediction and understanding of gene expression in *S. cerevisiae*. *BMC Bioinformatics* **14**: 1.
- Zur H, Tuller T. 2016. Predictive biophysical modeling and understanding of the dynamics of mRNA translation and its evolution. *Nucleic Acids Res* **44**: 9031–9049.

## B Appendix

### **MMSplice: modular modeling improves the predictions of genetic variant effects on splicing**

Genetic variants altering splicing constitute one of the most important classes of genetic determinants of rare and common diseases. Although various sequence-based models have been developed to predict the effects of genetic variants on splicing, quantitative prediction of how genetic variants affect splicing is still challenging. Many variant interpretation routines only check variants in close vicinity to splice sites. Therefore, many splice altering variants were overlooked previously.

In this study, I developed the framework MMSplice (modular modeling of splicing) with which I built the winning model of the CAGI5 exon-skipping prediction challenge. The MMSplice modules are neural networks scoring exon, flanking intronic sequence, as well as donor and acceptor splice sites. This modular approach allowed leveraging rich datasets including two high-throughput perturbation assays focusing on distinct aspects of splicing: (i) a massively parallel reporter assay with millions of random short sequences in intron and exon sequence (Rosenberg et al., 2015), and (ii) a high-throughput assay that quantifies the effect of naturally occurring exonic variants on the splicing of their exon (Adamson et al., 2018). These modules are combined to predict the effects of variants on exon skipping, splice site choice, splicing efficiency, and pathogenicity. MMSplice matched or outperformed the state-of-the-art models including Spidex, HAL, MaxEntScan on predicting the variant effect on various types of alternative splicing. Furthermore, benchmarked on ClinVar data, MMSplice improved distinguishing pathogenic variants from benign ones compared to previous models. An ensemble model including of MMSplice together with CADD and phyloP had shown similar performance compared to the ensemble of 5 previous models along with MMSplice, CADD, and phyloP on predicting ClinVar variant pathogenicity.

Besides single-nucleotide variants, my implementation of MMSplice also handles indels and automatically consider all possible exons a variant could affect. I provide MMSplice as a python package. Furthermore, all MMSplice modules and models are shared in the model repository Kipoi, which should allow other computational biologists to improve individual modules or to flexibly include modules into their own models. These features should facilitate the integration of MMSplice into bioinformatics pipelines at use in genetic diagnostic centers and may help to improve the discovery of pathogenic variants. I foresee that this modular approach will help the community to coordinate efforts and continuously and effectively build better variant effect prediction models for splicing.

## *B Appendix*

This is the original PDF of an article accepted for publication in *Genome Biology*: Cheng, J., Nguyen, T. Y. D., Cygan, K. J., Çelik, M. H., Fairbrother, W. G., Avsec, Ž., & Gagneur, J. (2019). MMSplice: modular modeling improves the predictions of genetic variant effects on splicing. *Genome Biology*, 20(1), 48. DOI:10.1186/s13059-019-1653-z

**License:** This is an open access article under the terms of the Creative Commons Attribution 4.0 License (CC BY 4.0), which permits use, distribution and reproduction in any medium or format, provided the original work is properly cited.


**Contribution of the thesis author:** model design and implementation, data analysis and visualization, literature review, results interpretation, manuscript composition.

METHOD

Open Access



# MMSplice: modular modeling improves the predictions of genetic variant effects on splicing

Jun Cheng<sup>1,2</sup>, Thi Yen Duong Nguyen<sup>1</sup>, Kamil J. Cygan<sup>3,4</sup>, Muhammed Hasan Çelik<sup>1</sup>, William G. Fairbrother<sup>3,4</sup>, Žiga Avsec<sup>1,2</sup> and Julien Gagneur<sup>1\*</sup> 

## Abstract

Predicting the effects of genetic variants on splicing is highly relevant for human genetics. We describe the framework MMSplice (modular modeling of splicing) with which we built the winning model of the CAG15 exon skipping prediction challenge. The MMSplice modules are neural networks scoring exon, intron, and splice sites, trained on distinct large-scale genomics datasets. These modules are combined to predict effects of variants on exon skipping, splice site choice, splicing efficiency, and pathogenicity, with matched or higher performance than state-of-the-art. Our models, available in the repository Kipoi, apply to variants including indels directly from VCF files.

**Keywords:** Splicing, Variant effect, Variant pathogenicity, Deep learning, Modular modeling

## Background

Genetic variants altering splicing constitute one of the most important class of genetic determinants of rare [1] and common [2] diseases. However, the accurate prediction of variant effects on splicing remains challenging.

Splicing is the outcome of multiple processes. It is a two-step catalytic process in which a donor site is first attacked by an intronic adenosine to form a branchpoint. In a second step, the acceptor site is cleaved and spliced (i.e., joined) to the 3' end of the donor site. The sequences of the donor site, of the acceptor site, and of the intronic region surrounding the branchpoint, which are recognized during spliceosome assembly, contribute to splicing regulation [3]. Moreover, many regulatory elements such as exonic splicing enhancers (ESEs) and silencers (ESSs) and intronic splicing enhancers (ISEs) and silencers (ISSs) also play key regulatory roles (reviewed by [4]). In addition to genetic variants at splice consensus sequence, distal elements can also affect splicing and cause disease [5]. Hence, predictive models of splicing need to integrate these various types of sequence elements.

Previous human splice variant interpretation methods can be grouped into two categories.

One category consists of algorithms that score sequence for being bona fide splice regulatory elements including splice sites [6, 7], and exonic and intronic enhancers and silencers [8–13]. Variants can be scored with respect to these regulatory elements by comparing predictions for the reference sequence and for the alternative sequence containing the genetic variant of interest. However, although methods combining several of these scores have been proposed, including Human Splicing Finder [14], MutPred splice [15], and more recently SPiCE [16], the resulting physical and quantitative effect of these variants on splicing remains difficult to assess with these algorithms.

The second category of models aimed at predicting relative amounts of alternative splicing isoforms quantitatively from sequence [17–19]. In this context, a quantitative measure that has retained much attention in the literature is the percent spliced-in (PSI, also denoted  $\Psi$ ), which quantifies exon skipping.  $\Psi$  is defined as the fraction of transcripts that contains a given exon [20]. It can be estimated as the fraction of exon-exon junction reads from an RNA-seq sample supporting inclusion of an exon of interest, over the sum of these reads plus those supporting the exclusion of this exon [20]. Two early models were

\*Correspondence: [gagneur@in.tum.de](mailto:gagneur@in.tum.de)

<sup>1</sup>Department of Informatics, Technical University of Munich, Boltzmannstraße, 85748 Garching, Germany

Full list of author information is available at the end of the article



fitted to predict direction of  $\Psi$  changes between tissues (exon inclusion, exon skipping, and no change) in mouse [21, 22] from sequence. State-of-the-art models for predicting  $\Psi$  from sequence are SPANR [17] and HAL [18] for human, and the model from Jha et al. [23] for mouse. The related quantity  $\Psi_5$  quantifies for a given donor site the fraction of spliced transcripts with a particular alternative 3' splice site (A3SS). The quantity  $\Psi_3$  has been analogously defined to quantify alternative 5' splice sites (A5SS) [24]. It should be noted that  $\Psi_5$  is often referred as  $\Psi$  for A3SS, and  $\Psi_3$  as  $\Psi$  for A5SS (e.g., [25, 26]). However, throughout this manuscript, we are consistently using the notations  $\Psi_5$  and  $\Psi_3$  as defined by Pervouchine et al. [24]. The recently published algorithm COSSMO [19] predicts  $\Psi_5$  from sequence by modeling the competition between alternative acceptor sites for a given donor site and analogously for  $\Psi_3$ . COSSMO has shown superior performance over MaxEntScan [7] on predicting the most frequently used splice site among competing ones. Furthermore, splicing efficiency has been proposed to quantify the amount of precursor RNA that undergo splicing (exon-skipped or misspliced transcripts are ignored) at a given splice site by comparing the amount of RNA-seq reads spanning an exon-intron boundary of interest to the corresponding exon-exon junction reads [27]. The latest model to predict variant effects on splicing efficiency is the SMS score, which is based on scores for exonic 7-mers estimated from a recently published saturation mutagenesis assay [28]. However, no model can be applied to all the abovementioned splicing quantities, although they are influenced by common regulatory elements. Furthermore, none of these software handle variant calling format (VCF) files natively, making their integration into genetic diagnostics pipelines cumbersome. Also, these software

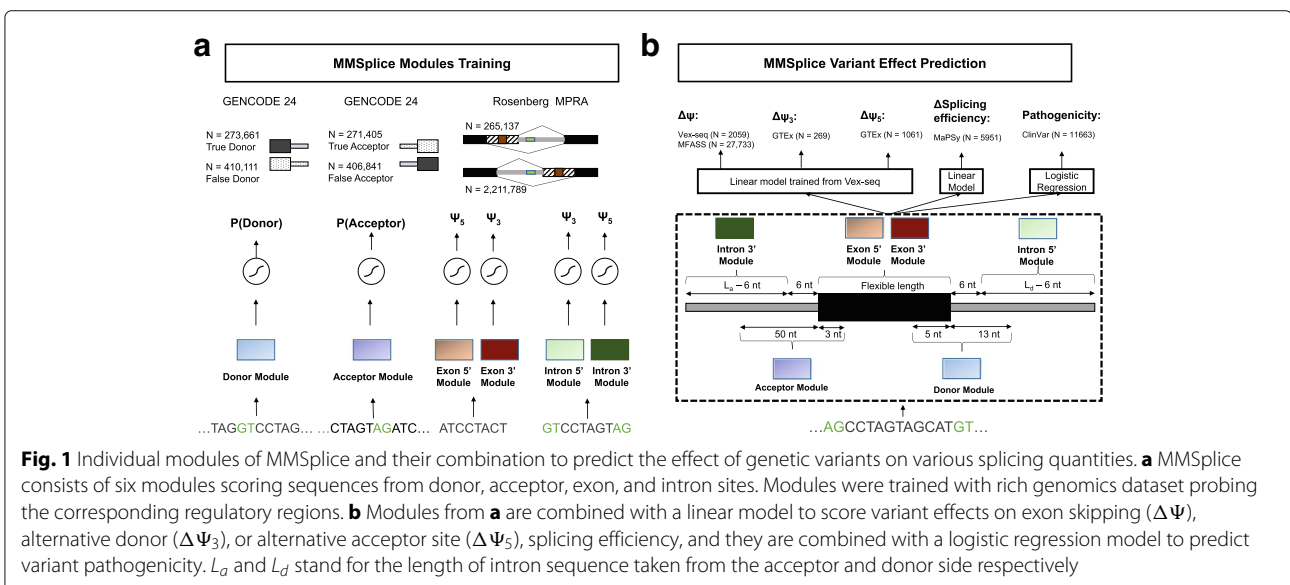
often do not handle indels (insertions and deletions), although indels are potentially the most deleterious variants.

Here, we trained building block modules separately for the exon, the acceptor site, and the donor site and for intronic sequence close to the donor and close to the acceptor sites. This modular approach allowed leveraging rich datasets from two high-throughput perturbation assays focusing on distinct aspects of splicing: (i) a massively parallel reporter assay (MPRA) with millions of random short sequences in intron and exon sequence [18], and (ii) a high-throughput assay that quantifies the effect of naturally occurring exonic variants on the splicing of their exon [29]. These building block modules could then be combined into distinct models predicting effects of variants on  $\Psi$ ,  $\Psi_5$ ,  $\Psi_3$ , splicing efficiency, and one model predicting splice variant pathogenicity trained on the database ClinVar [30]. We outperform state-of-the-art models for each task but  $\Psi_3$ , on which MMSplice and HAL both are the best. In particular, our model of exon skipping ranked first at the 5th challenge of the Critical Assessment of Genome Interpretation group (CAGI5, <https://genomeinterpretation.org/>). All our models are available open source in the model zoo Kipoi [31] and can be applied for variant effect prediction directly from VCF files.

## Results

### Modular modeling strategy

We designed neural networks to score five potentially overlapping splicing-relevant sequence regions: the donor site, the acceptor site, the exon, as well as the 5' end and the 3' end of the intron (Fig. 1a). The donor and the acceptor models were trained to predict annotated intron-exon



and exon-intron boundaries from GENCODE 24 genome annotation (see the “Methods” section, Fig. 1a, Additional file 1: Figure S1). The exon and intron models were trained from a MPRA that probed the effect of millions of random sequences altering either the exonic 3' end and the intronic 5' end for alternative 5' splicing (A5SS, quantified by  $\Psi_3$ ), or the exonic 5' end and the intronic 3' end for alternative 3' splicing (A3SS, quantified by  $\Psi_5$ ) (see the “Methods” section, Fig. 1a, Additional file 1: Figure S2) [18]. For later use, the modules were defined as the corresponding neural network models without the last activation layer. We have two intron modules, the intron

5' module that scores intron from the donor side and the intron 3' module that scores intron from the acceptor side. Likewise, we have two exon modules, the exon 5' module that trained from A3SS and exon 3' module that trained from A5SS (see the “Methods” section, Additional file 1: Figure S2). To score exonic sequence, only one of the exonic module is applied depending on the alternative splicing quantity. Training data and module architecture are summarized in Table 1. Next, we combined these modules to predict how genetic variants lead to (i) differences in  $\Psi$ , (ii) differences in  $\Psi_3$ , (iii) differences in  $\Psi_5$ , (iv) differences in splicing efficiency,

**Table 1** Summary of trained modules and models

MMSplice model	Training data	Architecture	Loss function	Target value	Parameters
Donor module	GENCODE 24, positive: annotated donors, negative: random sequence (“Methods” section)	Four layer neural network with dropout and batch normalization, Additional file 1: Figure S1A	Binary cross entropy	Positive vs. negative	18,049
Acceptor module	GENCODE 24, positive: annotated acceptors, negative: random sequence (“Methods” section)	Two layer conv. neural network with dropout and batch normalization, Additional file 1: Figure S1B	Binary cross entropy	Positive vs. negative	4833
Exon 5' module	MPRA [18] exonic sequence	One conv. layer shared with the Exon 3' module, followed with one specific dense layer, Additional file 1: Figure S2	Binary cross entropy	$\Psi_5$	6145
Exon 3' module	MPRA [18] exonic sequence	One conv. layer shared with the Exon 5' module, followed with one specific dense layer, Additional file 1: Figure S2	Binary cross entropy	$\Psi_3$	6145
Intron 5' module	MPRA [18] intronic sequence	One conv. layer shared with the Intron 3' module, followed with one specific dense layer, Additional file 1: Figure S2	Binary cross entropy	$\Psi_3$	13,825
Intron 3' module	MPRA [18] intronic sequence	One conv. layer shared with the Intron 5' module, followed with one specific dense layer, Additional file 1: Figure S2	Binary cross entropy	$\Psi_5$	13,825
$\Delta\text{logit}(\Psi)$ model	Vex-seq [29]	Linear regression	Huber loss	$\Delta\text{logit}(\Psi)$ , Eq. 2	9
Splicing efficiency model (in vivo)	MaPSy (“Methods” section)	Linear regression	Huber loss	Splicing efficiency, Eq. 10	5
Splicing efficiency model (in vitro)	MaPSy (“Methods” section)	Linear regression	Huber loss	Splicing efficiency, Eq. 10	5
Pathogenicity model (w/o phyloP and CADD)	ClinVar [30] [− 10, 10] around donor, [− 40, 10] around acceptor	Logistic regression	Binary cross entropy	Pathogenic vs. benign	14
Pathogenicity model (with phyloP and CADD)	ClinVar [30] [− 10, 10] around donor, [− 40, 10] around acceptor	Logistic regression	Binary cross entropy	Pathogenic vs. benign	18



and (v) to disease or benign phenotypes according to the ClinVar database (Fig. 1b). Specifically, we trained one linear model on top of the modules to predict  $\Delta\Psi$ . The same linear model was applied to predict  $\Delta\Psi_5$  and  $\Delta\Psi_3$  by modeling the competition of two alternative exons. Another linear model was trained to predict change of splicing efficiency and a logistic regression model was trained to predict variant pathogenicity from the modules (Fig. 1b).

### MMSplice improves the prediction of variant effect on exon skipping

To assess the performance of MMSplice for predicting effects of variants on exon skipping, we first considered the Vex-seq dataset [29]. Vex-seq is a high-throughput reporter assay that compared  $\Psi$  for constructs containing a reference sequence to  $\Psi$  for matching constructs containing one of 2059 Exome Aggregation Consortium (ExAC [32]) variants. The difference of  $\Psi$  for the variant allele to the reference allele is denoted  $\Delta\Psi$ . These variants consisted of both single nucleotide variants as well as indels from exons and introns (20 nt upstream, 50 nt downstream). The data for the HepG2 cell line was accessed through the Critical Assessment of Genome Interpretation (CAGI) competition [33]. The 957 variants from chromosome 1 to chromosome 8 were provided as training data. The remaining 1054 variants from chromosome 9 to 22 and chromosome X were held out for testing by the CAGI competition organizers and were not available throughout the development of the model. The test data consisted of 572 exonic and 526 intronic variants and included 44 indels.

The Vex-seq experiment is an exon skipping assay, whereas our exon modules were trained for A5SS ( $\Psi_3$ ) and A3SS ( $\Psi_5$ ). Because of high redundancy between these two modules, we used the exon 5' module as it was better at predicting exon skipping exonic variants on Vex-seq training data than the exon 3' module ( $R = 0.52$  v.s.  $R = 0.25$ ,  $P = 0.001$ , bootstrap, Additional file 1: Figure S3).

We built an MMSplice predictor for  $\Delta\Psi$  by training a linear model to combine the modular predictions and interaction terms between modules with overlapping scored regions from the Vex-seq training data (see the “Methods” section, Eq. 2). We compared MMSplice with three state-of-the-art splicing variant scoring models: SPANR [17], HAL [18], and MaxEntScan [7] on the held-out Vex-seq test data (“Methods” section). The methods HAL [18] and SPANR [17] have been reported to be the two best performed existing methods on a recent large-scale perturbation assay probing 27,733 rare variants [34], while MaxEntScan [7] was considered as a baseline reference model. SPANR scores exonic and intronic SNVs up to 300 nt around splice junctions. HAL scores exonic and donor (6 nt to the intron) variants.

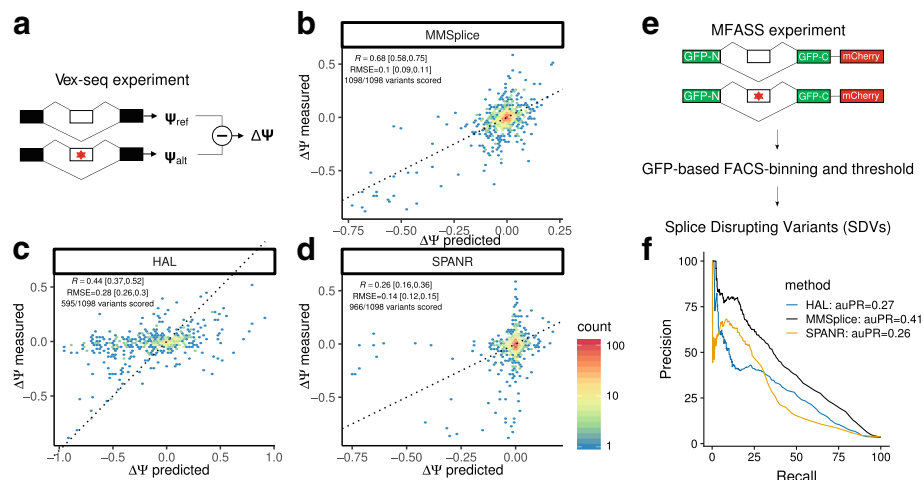
MaxEntScan scores  $[-3, +6]$  nt around the donor and  $[-20, +3]$  nt around the acceptor sites. The Vex-seq data was processed the same way for these models (“Methods” section). Unlike the other methods, SPANR does not take custom input sequences and could therefore score single nucleotide variants but not for indels. We evaluated the performance of  $\Delta\Psi$  predictions of MMSplice, HAL, and SPANR using root-mean-square errors (RMSE) on test data. MaxEntScan scores sequences but does not predict  $\Psi$ . We therefore compared the correlation of differences of MaxEntScan scores to  $\Delta\Psi$  and used Pearson correlation on test data as a common metric to compare all these methods.

On the Vex-seq data, MMSplice showed a large improvement over HAL and SPANR. First, MMSplice could score all 1098 variants of the test set whereas HAL could only score 572 (52.1%) and SPANR 966 (88%) of them. Second, the difference in  $\Psi$  predicted by MMSplice correlated better when restricted to the respective variants scored by the other methods ( $R = 0.68$  for MMSplice v.s.  $R = 0.44, 0.26$  for HAL and SPANR respectively, both comparison  $P = 0.001$ , bootstrap, Fig. 2b–d). A higher performance than other models was also obtained even when we bluntly summed the prediction scores from the five modules without fitting any parameter to the Vex-seq training data ( $R = 0.66$  and  $R = 0.67$  when using the exon 3' module in place of the exon 5' module, Additional file 1: Figure S4). This shows that the superior performance of our model is primarily due to the modules not the combination linear model that was trained from Vex-seq training data. Moreover, MMSplice showed higher accuracy than HAL and SPANR on these data when considering root-mean-square errors (RMSE = 0.1 for MMSplice versus 0.28 for HAL and 0.14 for SPANR, Fig. 2b–d).

We further compared our prediction for donor and acceptor site variants with the popular model MaxEntScan [7]. MMSplice performed better both in donor sequence ( $R = 0.87$  for MMSplice versus 0.66 for MaxEntScan5,  $P = 0.001$ , bootstrap, Additional file 1: Figure S5) and acceptor sequence ( $R = 0.81$  for MMSplice versus 0.69 for MaxEntScan3,  $P = 0.001$ , bootstrap, Additional file 1: Figure S6), when restricted to the subset of variants that MaxEntScan3 could score (42 donor variants and 149 acceptor variants). HAL performed better ( $R = 0.71$ ) than MaxEntScan5 ( $R = 0.66$ ) but worse than MMSplice ( $R = 0.87$ ) on donor variants ( $P = 0.001$  for both comparisons, bootstrap, Additional file 1: Figure S5).

Altogether, MMSplice outperformed SPANR, HAL, and MaxEntScan on predicting effects of genetic variants on exon skipping observed on this large-scale perturbation data, by covering more variants and also by providing more accurate predictions. Our model also ranked the first in the 2018 CAGI Vex-seq competition. A joint





**Fig. 2** MMSplice improves the prediction of variant effect on exon skipping. **a** Schema of the Vex-seq experiment [29]. The effect of 2059 ExAC variants (red star) from or adjacent to 110 alternative exons were tested with reporter genes by measuring percent splice-in of the reference sequence ( $\Psi_{ref}$ ) and of the alternative ( $\Psi_{alt}$ ) by RNAseq. **b–d** Measured (y-axis) versus predicted (x-axis)  $\Psi$  differences between alternative and reference sequence for MMSplice (**b**), HAL [18] (**c**), and SPANR [17] (**d**) on Vex-seq test data. Color scale represents counts in hexagonal bins. The black line marks the  $y = x$  diagonal. Each plot is shown with the subset of variants that the considered model can score. Pearson correlations ( $R$ ) and root-mean-square errors (RMSE) were also calculated based on the scored variants. The 95% confidence intervals for these two metrics were calculated with bootstrap (“Methods” section). **(e)** Schema of MFASS experiment [34]. Exon skipping effects of 27,733 ExAC SNVs (red star) spanning or adjacent to 2339 exons were tested by genome integration of designed construct. Splice-disrupting variant (SDV) is defined as a variant that change an exon with original exon inclusion index  $\geq 0.5$  by at least 0.5. **f** Precision-recall curve of MFASS SDV classification based on model predicted  $\Delta\Psi$ . Precision-recall curve for all three models was calculated for the sets of variants they can score. MMSplice (black) scored all 27,733 variants, SPANR (yellow) scored 27,663 variants (1,048 SDVs), and HAL (blue) scored 14,353 variants (489 SDVs)

publication with the organizers and challengers is in the planning.

### MMSplice classifies rare splice disrupting variants with higher precision and recall

To further compare models on predicting exon skipping level with independent datasets that no model has been trained on, we used the splicing functional assay from Cheung et al. [34]. Cheung et al. found 1050 splice-disrupting variants (SDVs); the majority are extremely rare, after examining 27,733 ExAC single-nucleotide variants (SNV) with Multiplexed Functional Assay of Splicing using Sort-seq (MFASS) (Fig. 2e). The author benchmarked several variant effect prediction methods including conservation-based methods like CADD [35], phastCons [36], and the state-of-the-art splicing variant scoring tools HAL and SPANR. Among all, the two splicing variant scoring methods performed much better than the others, thus MMSplice was compared with those two. MMSplice model with the final combination linear model trained from Vex-seq training data was applied to classify SDVs based on predicted  $\Delta\Psi$  solely from sequence. Our model achieved overall higher Area under the precision-recall curve (auPR, MMSplice: 0.41, HAL: 0.27, SPANR: 0.26,  $P = 0.001$  for both MMSplice versus HAL and MMSplice versus SPANR, bootstrap) when all models considering only their scored variants (Fig. 2f). In total,

MMSplice scored all variants, SPANR scored 99.7% of all variants, while HAL scored only 51.8% of them. When considering exonic variants only, MMSplice (auPR=0.29) performed similar to HAL (auPR = 0.27) ( $P = 0.326$ , bootstrap, Additional file 1: Figure S7). For intronic variants, MMSplice had an auPR of 0.55 in comparison to 0.43 for SPANR ( $P = 0.001$ , bootstrap, Additional file 1: Figure S7).

Overall, MMSplice demonstrated a substantial improvement over SPANR for both intronic and exonic variants and showed a similar performance to HAL for classifying exonic SDVs. This result also demonstrates the power of our model to score the effect of rare variants, for which association studies often lack of power.

### MMSplice predicts variants associated with competing splice site selection with high accuracy

The MMSplice modular framework allows modeling alternative splicing events other than exon skipping. To demonstrate this and assess the performance of MMSplice on other alternative splicing events, we built MMSplice models to predict association of variants around alternative donors on alternative 5' splicing (A5SS,  $\Psi_3$ ) and variants around alternative acceptors on alternative 3' splicing (A3SS) (“Methods” section) in GTEx.  $\Psi_5$  and  $\Psi_3$  values for homozygous reference variants as well as with

heterozygous and homozygous alternative variants were calculated from RNA-seq data of the GTEx consortium [37] (“Methods” section). Here too, our MMSplice models allowed handling indels. One example is the insertion variant rs11382548 (chr11:61165731:C-CA). It is a splice site variant that turns a CG acceptor to an AG acceptor. It showed the largest  $\Delta\Psi_5$  among all assessed variants.

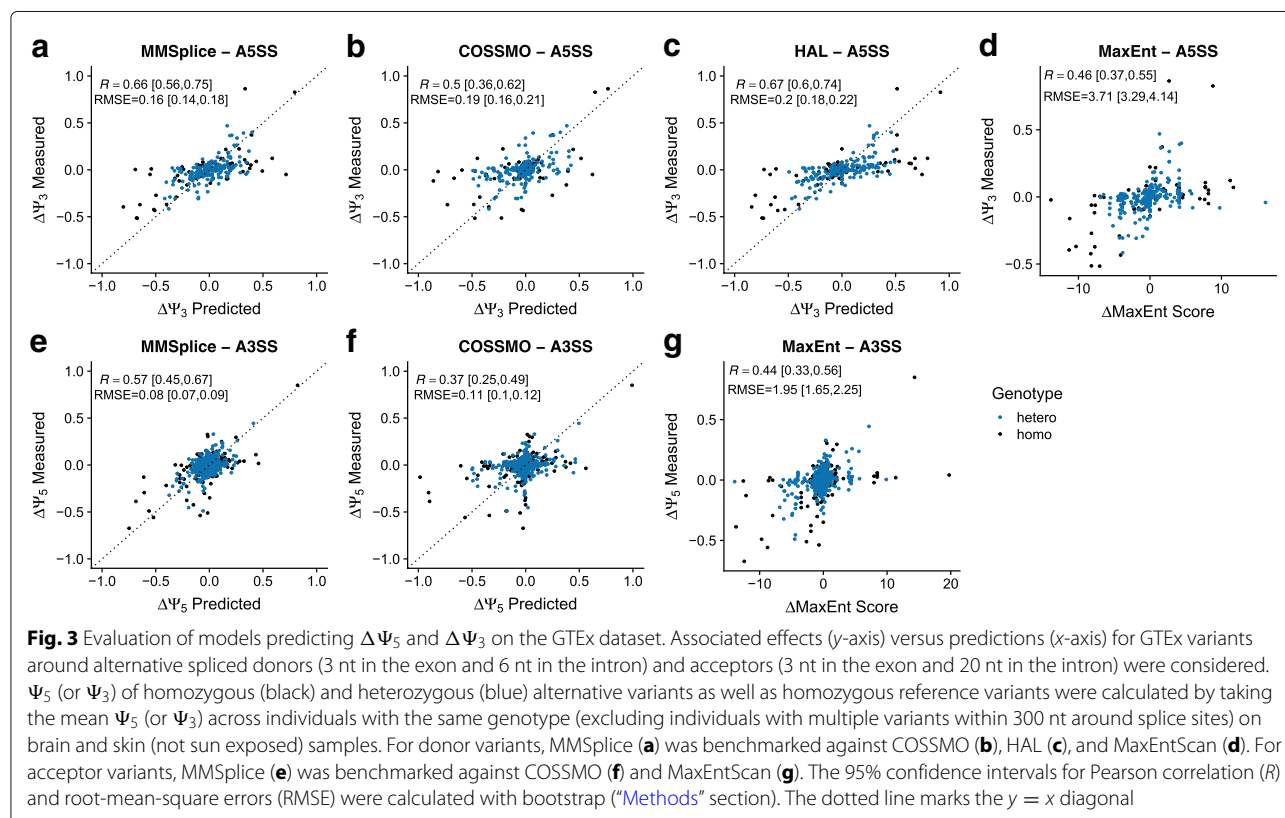
We benchmarked MMSplice against MaxEntScan, HAL, and COSSMO. Overall, MMSplice ( $R = 0.66$ ) significantly outperformed COSSMO ( $R = 0.5$ ,  $P = 0.016$ , bootstrap) and MaxEntScan ( $R = 0.46$ ,  $P = 0.001$ , bootstrap) and tied with HAL ( $R = 0.67$ ,  $P = 0.558$ , bootstrap) on predicting  $\Delta\Psi_3$  (Fig. 3a–d). On predicting  $\Delta\Psi_5$ , MMSplice ( $R = 0.57$ ) again significantly outperformed both COSSMO ( $R = 0.37$ ) and MaxEntScan ( $R = 0.44$ ) (all  $P = 0.001$ , Fig. 3e–g). This conclusion also holds when using RMSE as evaluation metric (Fig. 3). Even though HAL can predict A5SS donor variants well, the model has been trained for predicting A5SS and may not generalize well to other alternative splicing types. It only showed moderate performance when predicting donor variants from Vex-seq skipped exons (Additional file 1: Figure S5). In contrast, MMSplice showed consistent high performance across different types of alternative splicing events.

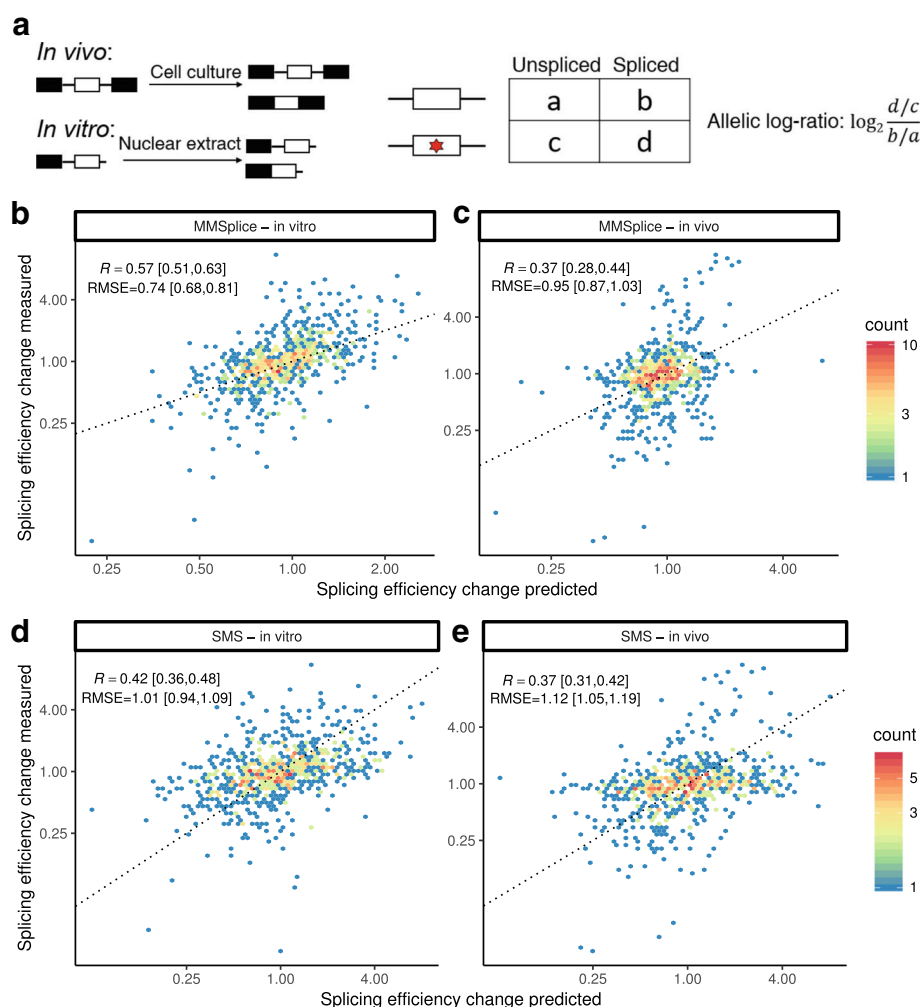
MMSplice outperformed COSSMO for both donor and acceptor variants even though COSSMO was trained from

estimated  $\Psi_5$  and  $\Psi_3$  values from GTEx data. One possible reason is that COSSMO was trained from reference sequence to predict  $\Psi_5$  and  $\Psi_3$ , ignoring the genetic variants of the GTEx dataset. In contrast, MMSplice was trained to predict  $\Delta\Psi$  from genetic perturbation data (Vex-Seq). Also, COSSMO was trained to predict splice site usage for an arbitrary number of alternative splice sites, while we focused here on the cases with only two alternative splice sites.

### Prediction of splicing efficiency

We next used our modular approach to derive a model that predicts splicing efficiency, i.e., the proportion of spliced RNAs among spliced and unspliced RNAs [27]. We have done so in the context of a second CAGI5 challenge (Fig. 4a), whose training dataset is based on a massively parallel splicing assay (MaPSy [27]) and which is described in the “Methods” section. This MaPSy dataset consists of splicing efficiencies, 5761 pairs of matched wild-type and mutated constructs, where each mutated construct differed from its matched wild-type by one exonic non-synonymous single-nucleotide variant (“Methods” section). The assay has been done both with an in vitro splicing assay and in vivo by transfection into HEK293 cells (“Methods” section). A test set of 797 construct pairs was held-out during the development of the model.





**Fig. 4** Splicing efficiency prediction. **a** MaPSy experiment (“Methods” section). Effect of 5761 published disease-causing exonic mutations on splicing efficiency is measured both *in vivo* and *in vitro*. Changes of splicing efficiency were quantified by allelic log-ratio. **b–e** Measured (y-axis) versus predicted (x-axis) allelic ratio for 797 variants in the test set for MMSplice (**b, c**) and the SMS score [28] (**d, e**). The dotted line marks the  $y = x$  diagonal. The 95% confidence intervals for Pearson correlation ( $R$ ) and root-mean-square errors (RMSE) were calculated with bootstrap (“Methods” section)

We trained a linear model on top of the modular predictions with MaPSy training data to predict differential splicing efficiency reported by the MaPSy data (“Methods” section). This linear model was trained the same way as for Vex-seq except that the response was the allelic log-ratio (Fig. 4a and “Methods” section) instead of  $\Delta \log_{it}(\Psi)$ . One model was trained for the *in vivo* data and another model was trained for the *in vitro* data. Our MMSplice model for differential splicing efficiencies predicted the effect of those non-synonymous mutations on the held-out test set reasonably well *in vitro* ( $R = 0.57$ , 4a) and well *in vivo* ( $R = 0.37$ , 4c). Also, our MMSplice model for differential splicing efficiencies outperformed the SMS score algorithm [28] on *in vitro* data ( $P = 0.001$ , bootstrap, 4d) and reached similar performance on the *in vivo* data ( $P = 0.524$ , bootstrap, 4e). MMSplice significantly outperformed SMS scores in both conditions when evaluated

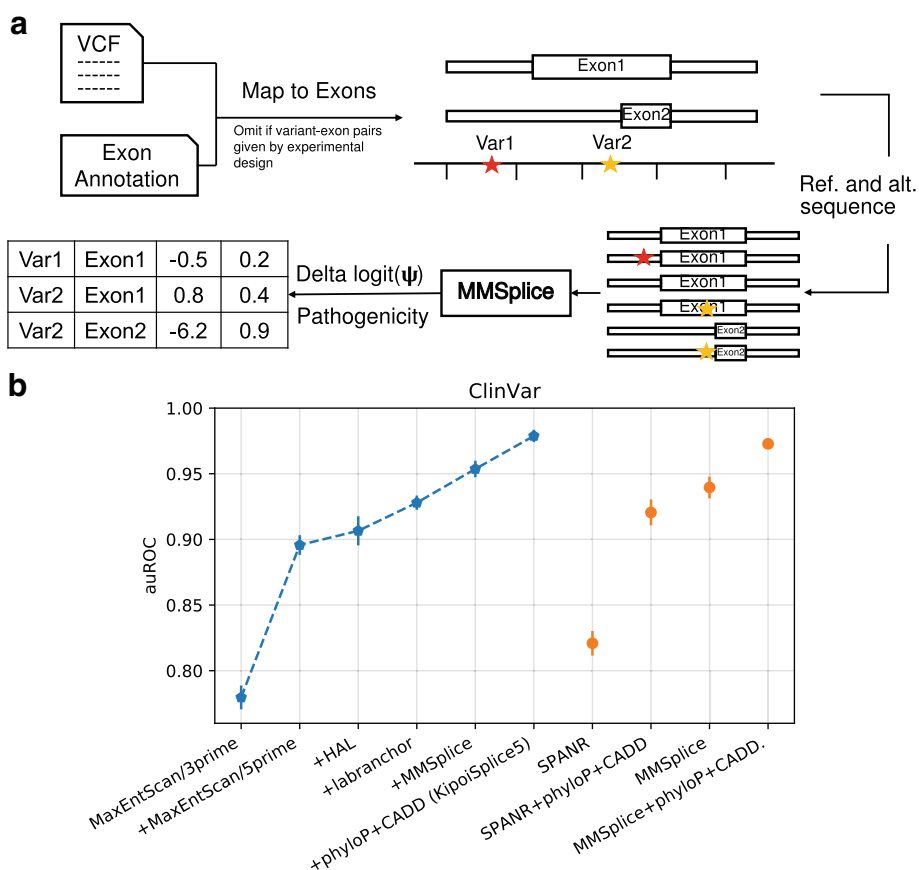
with RMSE (0.74 and 0.95 for MMSplice versus 1.01 and 1.12 for SMS scores,  $P = 0.001$  for both comparison, bootstrap). Several reasons may have led to the worse performance *in vivo*. One possible reason is that the *in vivo* assay may involve RNA degradation factors, which also regulate level of spliced RNA species by regulating RNA stability. Another possible reason is that the folding of RNAs *in vivo* may be more complex than *in vitro*, which in turn affects splicing [38], making the prediction *in vitro* more difficult.

#### MMSplice can contribute to improved predictions of splice site variant pathogenicity

Predicting variant pathogenicity is a central task of genetic diagnosis. However, large amount of variants are annotated as variant of uncertain significance (VUS). A good splice variant effect prediction model can help

interpreting VUSs. To evaluate the potential of MMSplice to contribute in predicting variant pathogenicity, we considered the ClinVar variants (version 20180429, [30]) that lie between 40 nt 5' and 10 nt 3' of an acceptor site or 10 nt either side of a donor site of a protein coding gene (Ensembl GRCh37 v75 annotation, "Methods" section) as potentially affecting splicing. Among these variants, we aimed at discriminating between the 6310 variants classified as pathogenic and the 4405 variants classified as benign. To this end, we built an MMSplice model that implements a logistic regression on top of the MMSplice modules ("Methods" section). Variants can potentially be in the vicinity of multiple exons. MMSplice handles this many-to-many relationship (Fig. 5a). Conveniently, MMSplice can be applied to a variant file in the standard format VCF [39] and a genome annotation file in the standard GTF format. Moreover, MMSplice is available as a Variant Effect Predictor Plugin (VEP [40]).

This MMSplice model was benchmarked against SPANR [17] and the ensemble of three other models: MaxEntScan [7], HAL [18], and the branch point predictor LaBranchoR [41]. We also compared our MMSplice model and competing models with phyloP and CADD scores as additional features (Additional file 1: Supplementary Methods). Model performances were benchmarked under 10-fold cross-validation (Fig. 5b). Globally on all the 10,715 considered variants, MMSplice alone (auROC = 0.940) outperformed SPANR (auROC = 0.821,  $P = 0.001$ , bootstrap) and the ensemble model combining MaxEntScan, HAL, and LaBranchoR (auROC = 0.928) ( $P = 0.001$ , bootstrap). Adding MMSplice to the ensemble model further improved the auROC to 0.954 ( $P = 0.001$ , bootstrap). Moreover, MMSplice with phyloP and CADD features (auROC = 0.973) achieved a performance close to the best ensemble model kipoSplice5 that included MMSplice (auROC = 0.979,  $P = 0.003$ , bootstrap, Fig. 5),



**Fig. 5** Predictions on ClinVar variants. **a** Variants are first mapped to potentially affected exons. Variants in the exon or in the intron, within  $L_d$  nt of the acceptor site or within  $L_d$  nt from the donor site are considered to affect splicing of the exon. Afterwards, reference and alternative sequences are retrieved and subjected to MMSplice for prediction. MMSplice gives a prediction for each variant-exon pair. **b** Model comparison on classifying pathogenicity of ClinVar splice variants. Models were trained and evaluated in 10-fold cross-validation. Error bars indicate one standard deviation calculated across folds. The six leftmost models (blue) are incrementally added to the ensemble model: "+phyloP+CADD" uses all five previous models as well as phyloP and CADD scores. Performance of MMSplice and SPANR alone as well as their performance with phyloP and CADD scores are on the right (orange)

indicating that MMSplice alone captured most of the sequence information captured by all other models.

We were then interested in delineating the added value of MMSplice per gene region. To this end, we grouped the variants based on their position yielding to (1) 832 exonic variants from the acceptor site region, (2) 1902 exonic variants from the donor site region, (3) 3575 intronic variants from the donor site region, and (4) 4393 intronic variants from the acceptor site region. On exonic variants, we further benchmarked against MutPred Splice [15] which predicts pathogenicity of exonic variants. Among the models that do not integrate phyloP and CADD features, MMSplice was the best in the acceptor site region (auROC = 0.602 for the exonic variants and auROC = 0.970 for the intronic variants, Additional file 1: Figure S8A,D). On the donor site region, MMSplice and the ensemble of MaxEntScan, HAL, and LaBranchoR were both the best models (auROC = 0.651 for the exonic variants and auROC = 0.977 for the intronic variants, Additional file 1: Figure S8B,C). MMSplice performed better than MutPred Splice on both exonic regions (MMSplice: auROC = 0.602, 0.651, MutPred: auROC = 0.594, 0.642, Additional file 1: Figure S8A,B), even though MutPred integrates conservation features [15]. Furthermore, the ensemble model that included MMSplice with phyloP and CADD features had a similar performance than the best ensemble model in all four regions (Additional file 1: Figure S9, auROC = 0.893, 0.917, 0.981, 0.982 versus auROC = 0.894, 0.919, 0.988, 0.985). Notably, phyloP and CADD had good performance on exonic variants (auROC = 0.874, 0.869), but close to random in the evaluated intronic variants (auROC = 0.505, 0.483). In contrast, all other splicing models without phyloP and CADD were performing better at intronic variants but much worse at exonic variants, likely because many pathogenic exonic variants do not affect splicing but have a functional impact on the protein.

Recently, SPiCE [16] has been proposed as a method to predict the probability of a splice site variant affecting splicing. SPiCE is a logistic regression model trained from 142 manually collected and experimentally tested variants. We thus benchmarked against SPiCE with 12,625 ClinVar variants (2312 indels) that SPiCE was able to score (it failed to score variants from sex chromosomes, “Methods” section). MMSplice (auROC = 0.911) outperformed SPiCE (auROC = 0.756,  $P = 0.001$ , bootstrap). Moreover, this higher performance of the MMSplice model also held when we fine-tuned the logistic regression model of SPiCE on the ClinVar training dataset (auROC = 0.760,  $P = 0.001$ , bootstrap, Additional file 1: Figure S10).

Altogether, these results show that MMSplice not only improves the predictions of the effects of variants on biophysical splicing quantities, but also helped improving variant pathogenicity predictions.

## Discussion

We have introduced MMSplice, a modular framework to predict the effects of genetic variants on splicing quantities. We did so by training individual modules scoring exon, intron, and splice sites. Models built by integrating these modules showed improved performance against state-of-the-art models on predicting the effects of genetic variants on  $\Psi$ ,  $\Psi_3$ ,  $\Psi_5$ , splicing efficiency, and pathogenicity. The MMSplice software is open source and can be directly applied on VCF files and handles single nucleotide variants and indels. Like other recent models [17–19], MMSplice score variants beyond the narrow region close to splice sites that is for now suggested by clinical guidelines [42]. We also implemented a VEP [40] plugin that wraps the python implementation. These features should facilitate the integration of MMSplice into bioinformatics pipelines at use in genetic diagnostic centers and may help in improving the discovery of pathogenic variants.

MMSplice leverages the modularity of neural networks and deep learning frameworks. MMSplice is implemented using the deep learning python library Keras [43]. All MMSplice modules and models are shared in the model repository Kipoi [31], which should allow other computational biologists to improve individual modules or to flexibly include modules into their own models. We hope this modular approach will help the community to coordinate efforts and continuously and effectively built better variant effect prediction models for splicing.

Variations across the reference genome or across natural genetic variations in the population may be limited by evolutionary confounding factors, limiting the model's ability to make predictions about rare genetic variants. Experimental perturbation assays are useful because they circumvent these confounding factors. Here, we have leveraged a massively parallel reporter assay [18] to build individual modules. Also, models predicting  $\Psi$  and splicing efficiencies were trained on large-scale perturbation datasets (Vex-seq [29] and MaPSy). We note however that MMSplice was not entirely fitted on perturbation assays: The donor site and the acceptor site modules have been trained on the GENCODE annotation, which is observational. Our models outperformed models based on the reference genome and natural variations and was only matched by models based on perturbation assays (HAL for  $\Delta\Psi_3$  and the SMS score for in vivo splicing efficiency changes). Nonetheless, one should remain cautious about how predictive rules learned from specific perturbation assays generalize to more general contexts. For instance, the Rosenberg MPRA dataset probed only two 25-nt-long sequences for a very specific construct. Hence, it is important to validate models on further independent perturbation data.



Our models have some limitations. First, splicing is known to be tissue-specific [44, 45], while our models are not. Nevertheless, our models can serve as a good foundation to train tissue-specific models. Second, RNA stability also plays a role in determining the ratio of different isoforms [29]. Models predicting RNA stability from sequence, as we recently developed for the *Saccharomyces cerevisiae* genome [46] could be integrated as further modules. Third, our exon and intron modules are developed from minigene studies, and the performance evaluation on predicting  $\Delta\Psi$  and splicing efficiency changes are also done with minigene experiment data. However, chromatin states are known to have a significant role in splicing regulation [47]. Hence, variant effect prediction for endogenous genes could possibly benefit from models taking chromatin states into account. Fourth, our exon and intron modules have only one convolutional layer, which is not enough to learn complex interaction effects of splicing regulatory elements [48]. We have explored using multiple convolutional layers, but the performance on the Vex-seq training data was similar (data not shown). We therefore chose the simpler architecture. The limitation may come from the training data, as the perturbation assay we are training from has 2.5 million random sequences of 25 nucleotides. This library is maybe not deep enough to probe motif interactions, relative distances, and orientations. Non-random libraries that probe the grammar of discovered motifs could be designed in the future and help studying motif interactions. Fifth, MMSplice can technically score variants arbitrarily deep into introns. However, as the training data of MMSplice did not cover deep intronic variants, we suggest to only consider up to 100 nt into introns, as we did here. Further models, such as SPANR which is able to score variants up to 300 nt into the intron, would need to be developed to cover deep intronic variants.

Like former splicing predictors [17–19, 21–23], the goal of MMSplice is to predict quantitatively physical measures of splicing and not variant pathogenicity. Whether affecting splicing at given locus leads to disease heavily depends on the function of the gene and of the splice isoforms. Moreover, existing pathogenicity annotations, such as from the ClinVar database, are probably biased toward tools such as MaxEntScan that are popular and have been in use for a long time. Nonetheless, our results indicate that MMSplice predictions could be potent predictive features for pathogenic variant scores such as S-CAP [49] or CADD [35].

## Methods

### Donor and acceptor modules

The donor and the acceptor modules were trained using the same approach. A classifier was trained to

classify positive donor sites (annotated) against negative ones (random, see below) and the same for the acceptor sites. The classifiers predicted scores can be interpreted as predicted strength of the splice sites.

### Donor and acceptor module training data

For the positive set, we took all annotated splice junctions based on the GENCODE annotation version 24 (GRCh38.p5). For the donor module, a sequence window with 5 nt in the exon and 13 nt in the intron around the donor sites was selected. For the acceptor module, the region around the acceptor sites spanning from 50 nt in the intron to 3 nt in the exon was selected in order to cover most branch points. In total, there were 273,661 unique annotated donor sites and 271,405 unique annotated acceptor sites. This set of splice sites was considered as the positive set. In particular, not only sites with the canonical splicing dinucleotides GT and AG for donor and acceptor sites, respectively, were selected, but also sites with non-canonical splicing dinucleotides were included as positive splice sites.

The negative set consisted of genomic sequences selected within the genes that contributed to positive splice sites, in order to approximately match the sequence context of the positive set. Negative splice sites were selected randomly around but not overlapping the positive splice sites. To increase the robustness of the classifiers, around 50% of the negative splice sites were selected to have the canonical splicing dinucleotides. In total, 410,111 negative donor sites and 406,841 negative acceptor sites were selected. During model training, we split 80% of the data for training and 20% of the data for validation. The best performing model on the validation set was used for variant effect prediction.

### Donor and acceptor module architecture

Neural network models were trained to score splice sites from one-hot-encoded input sequence. The donor model was a multilayer perceptron with two hidden layers with Rectified Linear Unit (ReLU) activations and a sigmoid output (Additional file 1: Figure S1A). The hidden layers were trained with a dropout rate [50] of 0.2 and batch normalization [51]. We chose a multilayer perceptron over a convolutional neural network because of the short input sequence of the donor model. The acceptor model was a convolutional neural network with two consecutive convolution layers, with 32  $15 \times 1$  convolution followed by 32  $1 \times 1$  convolution (Additional file 1: Figure S1B). The second convolutional layer was trained with a dropout rate of 0.2 and batch normalization. For these models, we found the number of layers and the number of neurons in each layer by hyperparameter optimization.

## Exon module

### Exon module training data

The exonic random sequences from the MPRA experiment by Rosenberg et al. [18] were used to train the exon scoring module. This MPRA experiment contains two libraries, one for alternative 5' splicing and one for alternative 3' splicing. The alternative 5' splicing library has 265,137 random constructs while the alternative 3' splicing library has 2,211,789. Each random construct has a 25-nt random sequence in the alternative exon and a 25-nt random sequence in the adjacent intron.  $\Psi_5$  and  $\Psi_3$  of different isoforms were quantified by RNA-Seq for each random construct [18]. Here, 80% of the data was used for model training and the remaining were used for validation. The best performing model on the validation set was used for variant effect prediction.

### Exon module architecture

Rosenberg et al. [18] showed that the effects of splicing-related features in alternative exons are strongly correlated with each other across the two MPRA libraries, reflecting that similar exonic regulatory elements are involved for both donor and acceptor splicing. We thus decided to train exon scoring module from the two MPRA libraries by sharing low-level convolution layers (128  $15 \times 1$  filters, Additional file 1: Figure S2). The inputs of the network were one-hot-encoded 25-nt random sequences. The output labels were  $\Psi_5$ , respectively  $\Psi_3$ , for the alternative exon. After training, the exon modules for each library were separated by transferring the corresponding weights to two separated modules with convolution layer with ReLU non-linearity followed by a global average pooling and a fully connected layer. We have used a global pooling after the convolution layer allowing to take exons of any length as input. This ended up with two exon scoring modules, one for alternative 5' end (exon 5' module) and one for alternative 3' end (exon 3' module).

### Intron module

Intron modules were trained in the same way as the exon modules (Additional file 1: Figure S2) by using intronic random sequences from the MPRA experiment as inputs, except that we used 256  $15 \times 1$  convolution filters, because intronic splicing regulatory elements from the donor side and the acceptor side are less similar [18]. This ended up with a module to score intron on the donor side (intron 5' module) and a module to score intron on the acceptor side (intron 3' module).

### Training procedure for the modules

All neural network models for the six modules were trained with binary cross-entropy loss (Eq. 1) and Adam

optimizer [52]. We implemented and trained these models with the deep learning python library Keras [43]. Bayesian optimization implemented in hyperopt package [53] was used for hyper-parameter optimization together with the kopt package ([github.com/avsecz/kopt](https://github.com/avsecz/kopt)). Every trial, a different hyper-parameter combination is proposed by the Bayesian optimizer, with which a model is trained on the training set, its performance is monitored by the validation loss. The model that had the smallest validation loss was selected.

$$\text{Loss}_i = -(\psi_i \log \hat{\psi}_i + (1 - \psi_i) \log(1 - \hat{\psi}_i)) \quad (1)$$

## Variant effect prediction models

### Variant processing

Variants are considered to affect the splicing of an exon if it is exonic or if it is intronic and at a distance less than  $L_a$  from an acceptor site or less than  $L_d$  from a donor site. The distances  $L_a$  and  $L_d$  were set to 100 nt in this study but can be flexibly set for MMSplice. MMSplice provides code to generate reference and alternative sequences from a variant-exon pair by substituting variants into the reference genome. Variant-exon pairs can be directly provided to MMSplice. This is the case for the perturbation assay data Vex-seq, MFASS, and MaPSy. MMSplice can also generate variant-exon pairs from given VCF files (Fig. 5a). For insertions, and for deletions that are not overlapping a splice site, the alternative sequence is obtained by inserting or deleting sequence correspondingly. For deletions overlapping a splice site, the alternative sequence is obtained by deleting the sequence and the new splice site is defined as the boundaries of the deletion. In all cases, the returned alternative sequence always have the same structure as the reference sequence, with an exon of flexible length flanked by  $L_a$  and  $L_d$  intronic nucleotides. Each variant is processed independently from the other variants, i.e., each mutated sequence contains only one variant (Fig. 5a). If a variant can affect multiple target (i.e., sites or exons), the MMSplice models return predictions for every possible target (Fig. 5a).

### Variant effect prediction for $\Psi$

Strand information of all Vex-seq assayed exons were first determined by overlapping them with Ensembl GRCh37 annotation release 75. Reference sequences were extracted by taking the whole exon and 100 nt flanking intronic sequence. Variant sequences were retrieved as described in the "Variant processing" in the "Methods" section, whereby variant-exon pairs were provided by the experimental design.

We modeled the differential effect on  $\Psi$  in the logistic scale with the following linear model:

$$\begin{aligned}
\Delta \logit(\Psi) &= \logit(\Psi_{alt}) - \logit(\Psi_{ref}) \\
&= \beta_0 + \beta_1 \Delta S_{3' \text{ intron}} \\
&\quad + \beta_2 \Delta S_{acceptor} + \beta_3 \Delta S_{exon} \\
&\quad + \beta_4 \Delta S_{donor} + \beta_5 \Delta S_{5' \text{ intron}} \\
&\quad + \beta_6 \mathbb{1}(\text{Exon overlap splice site modules}) \Delta S_{exon} \\
&\quad + \beta_7 \mathbb{1}(5' \text{ intron overlap donor module}) \Delta S_{5' \text{ intron}} \\
&\quad + \beta_8 \mathbb{1}(3' \text{ intron overlap acceptor module}) \Delta S_{3' \text{ intron}} \\
&\quad + \epsilon
\end{aligned} \tag{2}$$

where

$$\Delta S = S_{alt} - S_{ref} \tag{3}$$

for all five modules,  $\mathbb{1}(\cdot)$  is the indicator function,  $\epsilon$  is the error term, the suffix *alt* denotes the alternate allele, and the suffix *ref* denotes the reference allele. This model has nine parameters: one intercept, one coefficient for each of the five modules, and interaction terms for regions that were scored by two modules (Fig. 1). The latter interaction terms were useful to not double count the effect of variants scored by multiple modules. These nine parameters were the only parameters that were trained from the Vex-seq data. The parameters of the modules stayed fixed. To fit this linear model, we used Huber loss [54] instead of ordinary least squares loss to make the fitting more robust to outliers.

The model predicts  $\Delta \logit \Psi$  for the variant. We transform this to  $\Delta \Psi$  with a given reference  $\Psi$  as follows:

$$\begin{aligned}
\hat{\Psi}_{alt} &= \sigma(\Delta \logit \Psi + \logit(\Psi_{ref})) \\
\Delta \hat{\Psi} &= \hat{\Psi}_{alt} - \Psi_{ref}
\end{aligned} \tag{4}$$

where

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{5}$$

$$\logit(x) = \log \frac{x}{1-x} \tag{6}$$

To prevent infinite values in cases  $\Psi_{ref} = 0$  or  $\Psi_{ref} = 1$ ,  $\Psi_{ref}$  values were clipped to the interval  $[10^{-5}, 1 - 10^{-5}]$ .

HAL model is provided by the authors. A scaling factor required by HAL was trained on the Vex-seq training data using code provided by the authors [18]. The SPANR precomputed scores (which are called SPIDEX), were obtained from [http://www.openbioinformatics.org/annovar/spidex\\_download\\_form.php](http://www.openbioinformatics.org/annovar/spidex_download_form.php).

#### Performance on the MFASS dataset

MMSplice was applied the same way as for Vex-seq, except that module combining weights were learned from the Vex-seq training data, with MFASS data kept entirely unseen. SDVs were classified based on the predicted  $\Delta \Psi$  for a variant. Area under the precision-recall curve (auPR) were calculated with `trapz` function from R package `pracma`.

#### Variant effect prediction for $\Psi_3$ and $\Psi_5$

The Genotype-Tissue Expression (GTEx) [37] RNAseq data (V6) was used to extract variant effect on  $\Psi_3$  and  $\Psi_5$ . Variants  $[-3, +6]$  nt around alternative donors of alternative 5' splicing events and variants  $[-20, +3]$  nt around alternative acceptors for alternative 5' splicing events were considered. The skin (not sun exposed) samples and the brain samples with matched whole genome sequence data available were processed.  $\Psi_5$  and  $\Psi_3$  were calculated with MISO [20] for each sample. Altogether, 1057 brain samples and 211 skin samples could be successfully processed with MISO.  $\Psi_3$  and  $\Psi_5$  for homozygous reference variant, heterozygous variants, and homozygous alternative variants were calculated by taking the average across samples with the same genotype, excluding samples from individuals with more than one variants within 300 nt around the competing splice sites.

We predicted differences in  $\Psi_5$  as follows. We considered only donor sites with two alternative acceptor sites. We extracted the relevant sequences for the corresponding two alternative exons and apply the model of Eq. (2) which was fitted on Vex-seq training data. This returned a  $\Delta \logit(\Psi)$  for each alternative exon, denoted  $\Delta S_1$  and  $\Delta S_2$ , from which we calculate the predicted alternative  $\Psi_5$  as follows:

$$\Psi_{5alt} = \sigma(\Delta \logit(\Psi_5) + \logit(\Psi_{5ref})) \tag{7}$$

where we model the  $\Delta \logit(\Psi_5)$  considering the influence of variant on both alternative exon as follows (derivations provided in supplements):

$$\Delta \logit(\Psi_5) = \Delta S_1 - \Delta S_2 \tag{8}$$

The above computation applies to individual alleles. To handle heterozygous variants, we assumed expression from both alleles are equal. This led to the following predictions for homozygous and heterozygous variants:

$$\begin{aligned}
\Delta \Psi_{5homo} &= \Psi_{5alt} - \Psi_{5ref} \\
\Delta \Psi_{5hetero} &= (\Psi_{5ref} + \Psi_{5alt}) / 2 - \Psi_{5ref}
\end{aligned} \tag{9}$$

Analogous calculations were made to predict differences in  $\Psi_3$ .

Pre-trained COSSMO model [19] was obtained from the author website (<http://cossmo.genes.toronto.edu/>). The predicted  $\Delta \Psi_5$  (or  $\Delta \Psi_3$ ) values of COSSMO were calculated by taking the difference between the predicted  $\Psi_5$  (or  $\Psi_3$ ) from alternative sequence processed by MMSplice and reference sequence.

#### Splicing efficiency dataset (MaPSy data)

The splicing efficiency assay was performed for 5,761 disease causing exonic nonsynonymous variants both in vivo in HEK293 cells and in vitro in HeLa-S3 nuclear extract as previously described [27]. Here, the exons were derived from human exons and were reduced in size to be shorter



than 100 nt long by small deletions applied to both the reference and the alternative version of the sequence. This way, the wild-type and the mutated alleles differed from each other by a single point mutation and the wild-type allele differed from a human exon by the small deletions. The deletions were centered at the midpoint between the variant and the furthest exon boundary. The sequences of each substrate are listed in Additional file 2: Table S1 and also described further on the CAGI website (<https://genomeinterpretation.org/content/MaPSy>).

Overall, 4964 of the variants were in the training set and 797 were in the test set. The amount of spliced transcripts and unspliced transcripts for each construct with reference allele or alternative allele were determined by RNA-Seq. The effect of mutation on splicing efficiency for a specific reporter sequence was quantified by the allelic log-ratio, which is defined as:

$$\log_2 \left( \frac{m_o/m_i}{w_o/w_i} \right) \quad (10)$$

where  $m_o$  is the mutant spliced RNA read count,  $m_i$  is the mutant input (unspliced) RNA read count,  $w_o$  is the wild-type spliced RNA read count, and  $w_i$  is the wild-type input RNA read count. Transcripts with exon-skipped or misspliced are ignored.

#### Variant effect prediction for splicing efficiency (MaPSy data)

We fitted a model to predict differential splicing efficiency on the training set with a linear regression with a Huber loss as defined by Eq. 2, except that the response variable is the allelic log-ratio (Eq. 10) instead of  $\Delta \log(\Psi)$ . We used the exon 5' module for the splicing efficiency model. Performance on MaPSy data was reported on the held-out test set.

SMS scores was applied to wild-type and mutant sequence by summing up all 7-mer scores as described by Ke et al. [28]. The predicted allelic log-ratio is the SMS score difference between mutant and wild-type sequence.

#### Variant pathogenicity prediction

Processed ClinVar variants (version 20180429 for GRCh37) around splice sites were obtained from Avsec et al. [31]. Specifically, single-nucleotide variants [−40, 10] nt around the splicing acceptor or [−10, 10] nt around the splice donor of a protein-coding genes (Ensembl GRCh37 v75 annotation) were selected. Variants causing a premature stop codon were discarded. After the filtering, the 6310 pathogenic variants constituted the positive set and the 4405 benign variants constituted the negative set. The CADD [35] scores and the phyloP [55] scores were obtained through VEP [40]. MMSplice  $\Delta$ Score predictions of the five modules as well as indicator variables of the overlapping region were assembled with a logistic regression model

to classify pathogenicity. Performance was assessed by 10-fold cross-validation (Additional file 1: Supplementary Methods).

To compare MMSplice with SPiCE [16], we restricted to the regions that SPiCE scores, i.e., [−12, 2] nt around the acceptor or [−3, 8] nt around the donor of protein-coding genes. Variants causing a premature stop codon were discarded. SPiCE was trained to predict the probability of a variant to affect splicing (manually defined by experimental observations). To apply it for pathogenicity prediction, the logistic regression model of SPiCE was refitted with ClinVar pathogenicity as response variable. MMSplice model was applied as described above without conservation features. Models were compared under 10-fold cross-validation.

#### Bootstrapping for P value and confidence interval estimation

Significance levels when comparing the performance of two models were estimated with the basic bootstrap [56]. Denoting  $t_1$  the performance metric (Pearson correlation, auPRC, or auROC) of MMSplice and  $t_2$  the performance metric of a competing model, we considered the difference  $d = t_1 - t_2$ . We sampled with replacement the test data  $B = 999$  times and each time  $i$  computed the bootstrapped metric difference  $d_i^*$ . The one-sided  $P$  value was approximated as [56].

$$P = \frac{1 + \#\{d_i^* \leq 0; i = 1 \dots B\}}{B + 1} \quad (11)$$

We estimated confidence intervals of Pearson correlations and root-mean-square values, using the percentile bootstrap approach. Specifically, we generated 1000 bootstrap datasets of the same size by sampling with replacement. Noting the value of either of the statistics of interest as  $\theta^*$ , the reported 95% confidence interval is  $(\theta_{0.025}^*, \theta_{0.975}^*)$ , where  $\theta_{0.025}^*$  and  $(\theta_{0.975}^*)$  are the 2.5 and the 97.5 percentiles, respectively.

#### Additional files

**Additional file 1:** Supplementary methods and figures. (PDF 674 kb)

**Additional file 2:** Table S1: MaPSy splicing efficiency data. (CSV 3 kb)

#### Acknowledgements

We thank the Critical Assessment of Genome Interpretation (CAGI) organizers, especially Steven E. Brenner, John Moulton and Gaia Andreoletti for organizing the CAGI competition. We thank Scott I. Adamson, Brenton R. Graveley for formatting and providing Vex-seq experiment data through CAGI.

#### Funding

JC was supported by the Competence Network for Technical, Scientific High Performance Computing in Bavaria KONWIHR. ZA and JC were supported by a Deutsche Forschungsgemeinschaft fellowship through the Graduate School of Quantitative Biosciences Munich. This work was supported by NVIDIA hardware grant providing a Titan X GPU card.

**Availability of data and materials**

MMSplice and its VEP plugin are available at <https://github.com/gagneurlab/MMSplice> [57] and Kipoi: <https://kipoi.org/models/MMSplice> [58] under the MIT License. An archival version of MMSplice is available at zenodo: <https://doi.org/10.5281/zenodo.2555955> [59]. The analysis code is available at [https://github.com/gagneurlab/MMSplice\\_paper](https://github.com/gagneurlab/MMSplice_paper) [60]. Vex-seq data is available at <https://github.com/scottiamadson/Vex-seq> [61]. MFASS data is available at <https://github.com/KosuriLab/MFASS> [62]. GTEx data is available through Genotypes and Phenotypes (dbGaP) under accession number (phs000424.v6.p1). ClinVar data is available at [ftp://ftp.ncbi.nlm.nih.gov/pub/clinvar/vcf\\_GRCh37/archive\\_2.0/2018/clinvar\\_20180429.vcf.gz](ftp://ftp.ncbi.nlm.nih.gov/pub/clinvar/vcf_GRCh37/archive_2.0/2018/clinvar_20180429.vcf.gz) [63]. MaPSy data is available as Additional files 2 of this manuscript.

**Authors' contributions**

JC and JG designed the model, with the help of ZA. JC implemented the software and analysed data. TYDN and ZA contributed to developing the modules. JC and JG wrote the manuscript, with the help of ZA, KJC, and WGF. KJC and WGF generated the MaPSy data. MHC wrote the VEP plugin. All authors read and approved the final manuscript.

**Ethics approval and consent to participate**

Not applicable.

**Consent for publication**

Not applicable.

**Competing interests**

The authors declare that they have no competing interests.

**Publisher's Note**

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Author details**

<sup>1</sup>Department of Informatics, Technical University of Munich, Boltzmannstraße, 85748 Garching, Germany. <sup>2</sup>Graduate School of Quantitative Biosciences (QBM), Ludwig-Maximilians-Universität München, München, Germany. <sup>3</sup>Center for Computational Molecular Biology, Brown University, Providence, Rhode Island, USA. <sup>4</sup>Department of Molecular Biology, Cell Biology and Biochemistry, Brown University, Providence, Rhode Island, USA.

Received: 2 October 2018 Accepted: 12 February 2019

Published online: 01 March 2019

**References**

- López-Bigas N, Audit B, Ouzounis C, Parra G, Guigó R. Are splicing mutations the most frequent cause of hereditary disease? *FEBS Lett*. 2005;579(9):1900–3. <https://doi.org/10.1016/j.febslet.2005.02.047>.
- Li Y, van de Geijn B, Raj A, Knowles DA, Petti AA, Golan D, Gilad Y, Pritchard JK. RNA splicing is a primary link between genetic variation and disease. *Science*. 2016;352(6285):600–4. <https://doi.org/10.1126/science.aad9417>.
- Wahl MC, Will CL, Lührmann R. The spliceosome: design principles of a dynamic RNP machine. *Cell*. 2009;136(4):701–18.
- Wang Z, Burge CB. Splicing regulation: from a parts list of regulatory elements to an integrated splicing code. *Rna*. 2008;14(5):802–13.
- Scotti MM, Swanson MS. RNA mis-splicing in disease. *Nat Rev Genet*. 2015;17(1):19–32. <https://doi.org/10.1038/nrg.2015.3>.
- Reese MG, Eeckman FH, Kulp D, Haussler D. Improved splice site detection in Genie. *J Comput Biol*. 1997;4(3):311–23.
- Yeo G, Burge CB. Maximum entropy modeling of short sequence motifs with applications to RNA splicing signals. *J Comput Biol J Comput Mol Biol*. 2004;11(2-3):377–94. <https://doi.org/10.1089/1066527041410418>.
- Fairbrother WG, Yeh RF, Sharp PA, Burge CB. Predictive identification of exonic splicing enhancers in human genes. *Science*. 2002;297(5583):1007–13.
- Fairbrother WG, Yeo GW, Yeh R, Goldstein P, Mawson M, Sharp PA, Burge CB. RESCUE-ESE identifies candidate exonic splicing enhancers in vertebrate exons. *Nucleic Acids Res*. 2004;32(Web Server issue):187–90. <https://doi.org/10.1093/nar/gkh393>.
- Zhang XHF, Chasin LA. Computational definition of sequence motifs governing constitutive exon splicing. *Genes Dev*. 2004;18(11):1241–50. <https://doi.org/10.1101/gad.1195304>.
- Zhang XH-F, Kangsamaksin T, Chao MS, Banerjee JK, Chasin LA. Exon inclusion is dependent on predictable exonic splicing enhancers. *Mol Cell Biol*. 2005;25(16):7323–32.
- Wang Z, Xiao X, Van Nostrand E, Burge CB. General and specific functions of exonic splicing silencers in splicing control. *Mol Cell*. 2006;23(1):61–70.
- Ke S, Shang S, Kalachikov SM, Morozova I, Yu L, Russo JJ, Ju J, Chasin LA. Quantitative evaluation of all hexamers as exonic splicing elements. *Genome Res*. 2011;21(8):1360–74. <https://doi.org/10.1101/gr.119628.110>.
- Desmet FO, Hamroun D, Lalande M, Collod-Bérout G, Claustres M, Bérout C. Human Splicing Finder: an online bioinformatics tool to predict splicing signals. *Nucleic Acids Res*. 2009;37(9):67. <https://doi.org/10.1093/nar/gkp215>.
- Mort M, Sterne-Weiler T, Li B, Ball EV, Cooper DN, Radivojac P, Sanford JR, Mooney SD. MutPred Splice: machine learning-based prediction of exonic variants that disrupt splicing. *Genome Biol*. 2014;15(1):19. <https://doi.org/10.1186/gb-2014-15-1-r19>.
- Leman R, Gaildrat P, Gac GL, Ka C, Fichou Y, Audrezet M-P, Caux-Moncoutier V, Caputo SM, Boutry-Kryza N, Léone M, et al. Novel diagnostic tool for prediction of variant spliceogenicity derived from a set of 395 combined in silico/in vitro studies: an international collaborative effort. *Nucleic Acids Res*. 2018;46(15):7913–23.
- Xiong HY, Alipanahi B, Lee LJ, Bretschneider H, Merico D, Yuen RKC, Hua Y, Gueroussov S, Najafabadi HS, Hughes TR, Morris Q, Barash Y, Krainer AR, Jovic N, Scherer SW, Blencowe BJ, Frey BJ. The human splicing code reveals new insights into the genetic determinants of disease. *Science* (80-). 2015;347(6218):1254806. <https://doi.org/10.1126/science.1254806>.
- Rosenberg AB, Patwardhan RP, Shendure J, Seelig G. Learning the sequence determinants of alternative splicing from millions of random sequences. *Cell*. 2015;163(3):698–711. <https://doi.org/10.1016/j.cell.2015.09.054>.
- Bretschneider H, Gandhi S, Deshwar AG, Zuberi K, Frey BJ. COSSMO: predicting competitive alternative splice site selection using deep learning. *Bioinformatics (Oxford, England)*. 2018;34(13):429–37. <https://doi.org/10.1093/bioinformatics/bty244>.
- Katz Y, Wang ET, Airoldi EM, Burge CB. Analysis and design of RNA sequencing experiments for identifying isoform regulation. *Nat Methods*. 2010;7(12):1009–15. <https://doi.org/10.1038/nmeth.1528>. 9605103.
- Barash Y, Calarco JA, Gao W, Pan Q, Wang X, Shai O, Blencowe BJ, Frey BJ. Deciphering the splicing code. *Nature*. 2010;465(7294):53–9. <https://doi.org/10.1038/nature09000>.
- Xiong HY, Barash Y, Frey BJ. Bayesian prediction of tissue-regulated splicing using RNA sequence and cellular context. *Bioinformatics*. 2011;27(18):2554–62. <https://doi.org/10.1093/bioinformatics/btr444>.
- Jha A, Gazzara MR, Barash Y. Integrative deep models for alternative splicing. *Bioinformatics*. 2017;33(14):274–82. <https://doi.org/10.1093/bioinformatics/btx268>.
- Pervouchine DD, Knowles DG, Guigó R. Intron-centric estimation of alternative splicing from rna-seq data. *Bioinformatics*. 2012;29(2):273–4.
- Park E, Pan Z, Zhang Z, Lin L, Xing Y. The expanding landscape of alternative splicing variation in human populations. *Am J Hum Genet*. 2018;102(1):11–26. <https://doi.org/10.1016/j.ajhg.2017.11.002>.
- Vaquero-García J, Barrera A, Gazzara MR, González-Vallinas J, Lahens NF, Hogenesch JB, Lynch KW, Barash Y. A new view of transcriptome complexity and regulation through the lens of local splicing variations. *eLife*. 2016;5:11752. <https://doi.org/10.7554/eLife.11752>. [arXiv:1011.1669v3](https://arxiv.org/abs/1011.1669v3).
- Soemedi R, Cygan KJ, Rhine CL, Wang J, Bulacan C, Yang J, Bayrak-Toydemir P, McDonald J, Fairbrother WG. Pathogenic variants that alter protein code often disrupt splicing. *Nat Genet*. 2017;49(6):848–55. <https://doi.org/10.1038/ng.3837>.
- Ke S, Anquetil V, Zamalloa JR, Maity A, Yang A, Arias MA, Kalachikov S, Russo JJ, Ju J, Chasin LA. Saturation mutagenesis reveals manifold determinants of exon definition. *Genome Res*. 2018;28(1):11–24.
- Adamson SI, Zhan L, Graveley BR. Vex-seq: high-throughput identification of the impact of genetic variation on pre-mRNA splicing efficiency. *Genome Biol*. 2018;19(1):71.
- Landrum MJ, Lee JM, Riley GR, Jang W, Rubinstein WS, Church DM, Maglott DR. Clinvar: public archive of relationships among sequence variation and human phenotype. *Nucleic Acids Res*. 2013;42(D1):980–5.

31. Avsec Z, Kreuzhuber R, Israeli J, Xu N, Cheng J, Shrikumar A, Banerjee A, Kim DS, Urban L, Kundaje A, Stegle O, Gagneur J. Kipoi: accelerating the community exchange and reuse of predictive models for genomics. *bioRxiv*. 2018. <https://doi.org/10.1101/375345>. <https://www.biorxiv.org/content/early/2018/07/24/375345.full.pdf>.
32. Lek M, Karczewski KJ, Minikel EV, Samocha KE, Banks E, Fennell T, O'Donnell-Luria AH, Ware JS, Hill AJ, Cummings BB, Tukiainen T, Birnbaum DP, Kosmicki JA, Duncan LE, Estrada K, Zhao F, Zou J, Pierce-Hoffman E, Berghout J, Cooper DN, Deflaux N, DePristo M, Do R, Flannick J, Fromer M, Gauthier L, Goldstein J, Gupta N, Howrigan D, Kiezun A, Kurki MI, Moonshine AL, Natarajan P, Orozco L, Peloso GM, Poplin R, Rivas MA, Ruano-Rubio V, Rose SA, Ruderfer DM, Shakir K, Stenson PD, Stevens C, Thomas BP, Tiao G, Tusie-Luna MT, Weisburd B, Won HH, Yu D, Altshuler DM, Ardissino D, Boehnke M, Danesh J, Donnelly S, Elosua R, Florez JC, Gabriel SB, Getz G, Glatt SJ, Hultman CM, Kathiresan S, Laakso M, McCarroll S, McCarthy ML, McGovern D, McPherson R, Neale BM, Palotie A, Purcell SM, Saleheen D, Scharf JM, Sklar P, Sullivan PF, Tuomilehto J, Tsuang MT, Watkins HC, Wilson JG, Daly MJ, MacArthur DG. Analysis of protein-coding genetic variation in 60,706 humans. *Nature*. 2016;536(7616):285–91. <https://doi.org/10.1038/nature19057>. 030338.
33. Hoskins RA, Repo S, Barsky D, Andreoletti G, Moulton J, Brenner SE. Reports from CAGI: the critical assessment of genome interpretation. *Hum Mutat*. 2017;38(9):1039–41.
34. Cheung R, Insigne KD, Yao D, Burghard CP, Wang J, Hsiao Y-HE, Jones EM, Goodman DB, Xiao X, Kosuri S. A multiplexed assay for exon recognition reveals that an unappreciated fraction of rare genetic variants cause large-effect splicing disruptions. *Mol Cell*. 2019;73(1):183–94.
35. Kircher M, Witten DM, Jain P, O'roak BJ, Cooper GM, Shendure J. A general framework for estimating the relative pathogenicity of human genetic variants. *Nat Genet*. 2014;46(3):310.
36. Siepel A, Bejerano G, Pedersen JS, Hinrichs AS, Hou M, Rosenbloom K, Clawson H, Spieth J, Hillier LW, Richards S, et al. Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. *Genome Res*. 2005;15(8):1034–50.
37. Lonsdale J, Thomas J, Salvatore M, Phillips R, Lo E, Shad S, Hasz R, Walters G, Garcia F, Young N, et al. The genotype-tissue expression (gtex) project. *Nat Genet*. 2013;45(6):580.
38. Warf MB, Berglund JA. Role of RNA structure in regulating pre-mRNA splicing. *Trends Biochem Sci*. 2010;35(3):169–78. <https://doi.org/10.1016/j.tibs.2009.10.004>.
39. Danecek P, Auton A, Abecasis G, Albers CA, Banks E, DePristo MA, Handsaker RE, Lunter G, Marth GT, Sherry ST, McVean G, Durbin R. The variant call format and VCFtools. *Bioinformatics*. 2011;27(15):2156–8. <https://doi.org/10.1093/bioinformatics/btr330>. NIHMS150003.
40. McLaren W, Gil L, Hunt SE, Riat HS, Ritchie GRS, Thormann A, Flicek P, Cunningham F. The Ensembl variant effect predictor. *Genome Biol*. 2016;17(1):. <https://doi.org/10.1186/s11059-016-0974-4>.
41. Paggi JM, Bejerano G. A sequence-based, deep learning model accurately predicts RNA splicing branchpoints. *RNA*. 2018;24(12):1647–58. <https://doi.org/10.1261/rna.066290.118>.
42. Richards S, Aziz N, Bale S, Bick D, Das S, Gastier-Foster J, Grody WW, Hegde M, Lyon E, Spector E, Voelkerding K, Rehm HL. Standards and guidelines for the interpretation of sequence variants: a joint consensus recommendation of the American College of Medical Genetics and Genomics and the Association for Molecular Pathology. *Genet Med*. 2015;17(5):405–24. <https://doi.org/10.1038/gim.2015.30>. 15334406.
43. Chollet F, et al. Keras. 2015. <https://keras.io>, version: 2.2.4.
44. Consortium G, et al. The Genotype-Tissue Expression (GTEx) pilot analysis: multitissue gene regulation in humans. *Science*. 2015;348(6235):648–0.
45. Yeo G, Holste D, Kreiman G, Burge CB. Variation in alternative splicing across human tissues. *Genome Biol*. 2004;5(10):74. <https://doi.org/10.1186/gb-2004-5-10-r74>.
46. Cheng J, Maier KC, Avsec Z, Rus P, Gagneur J. Cis-regulatory elements explain most of the mRNA stability variation across genes in yeast. *RNA*. 2017;23(11):1648–59. <https://doi.org/10.1261/rna.062224.117>.
47. Kolasinska-Zwier P, Down T, Latorre I, Liu T, Liu XS, Ahringer J. Differential chromatin marking of introns and expressed exons by H3K36me3. *Nat Genet*. 2009;41(3):376–81. <https://doi.org/10.1038/ng.322>.
48. Han K, Yeo G, An P, Burge CB, Grabowski PJ. A combinatorial code for splicing silencing: UAGG and GGGG motifs. *PLoS Biol*. 2005;3(5):0843–60. <https://doi.org/10.1371/journal.pbio.0030158>.
49. Jagadeesh KA, Paggi JM, Ye JS, Stenson PD, Cooper DN, Bernstein JA, Bejerano G. S-CAP extends clinical-grade pathogenicity prediction to genetic variants that affect RNA splicing. *bioRxiv*. 2018;343749. <https://doi.org/10.1101/343749>.
50. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. *J Mach Learn Res*. 2014;15(1):1929–58.
51. Ioffe S, Szegedy C. Batch Normalization: accelerating deep network training by reducing internal covariate shift. *arXiv*. 2015. 1502.03167.
52. Kingma D, Ba J. Adam: a method for stochastic optimization. 2014. *arXiv preprint arXiv:1412.6980*.
53. Bergstra J, Komer B, Eliasmith C, Yamins D, Cox DD. Hyperopt: a python library for model selection and hyperparameter optimization. *Comput Sci Discov*. 2015;8(1):014008.
54. Huber PJ. Robust estimation of a location parameter. *Ann Math Stat*. 1964;35(1):73–101. <https://doi.org/10.1214/aoms/1177703732>. *arXiv:1111.1308v3*.
55. Pollard KS, Hubisz MJ, Rosenbloom KR, Siepel A. Detection of nonneutral substitution rates on mammalian phylogenies. *Genome Res*. 2010;20(1):110–21. <https://doi.org/10.1101/gr.097857.109>.
56. Davison AC, Hinkley DV. *Bootstrap methods and their applications*, vol. 1. Cambridge University Press; 1997.
57. Cheng J, Çelik MH. MMSplice: modular modeling improves the predictions of genetic variant effects on splicing. *GitHub*. <https://github.com/gagneurlab/MMSplice>.
58. Cheng J, Çelik MH, Avsec Z. MMSplice: modular modeling improves the predictions of genetic variant effects on splicing. *GitHub*. <https://github.com/kipoi/models/tree/master/MMSplice>.
59. Cheng J. MMSplice: modular modeling improves the predictions of genetic variant effects on splicing. *Zenodo*. <https://doi.org/10.5281/zenodo.2555955>.
60. Cheng J. MMSplice: modular modeling improves the predictions of genetic variant effects on splicing. *GitHub*. [https://github.com/gagneurlab/MMSplice\\_paper](https://github.com/gagneurlab/MMSplice_paper).
61. Adamson SI. Vex-seq: high-throughput identification of the impact of genetic variation on pre-mRNA splicing efficiency. *GitHub*. <https://github.com/scottiadamson/Vex-seq>. Accessed 16 Feb 2018.
62. Insigne KD. A multiplexed assay for exon recognition reveals that an unappreciated fraction of rare genetic variants cause large-effect splicing disruptions. *GitHub*. <https://github.com/KosuriLab/MFASS>. Accessed 15 Mar 2018.
63. Landrum MJ, Lee JM, Riley GR, Jang W, Rubinstein WS, Church DM, Maglott DR. ClinVar: public archive of relationships among sequence variation and human phenotype. *ClinVar*. [ftp://ftp.ncbi.nlm.nih.gov/pub/clinvar/vcf\\_GRCh37/archive\\_2\\_0/2018/clinvar\\_20180429.vcf.gz](ftp://ftp.ncbi.nlm.nih.gov/pub/clinvar/vcf_GRCh37/archive_2_0/2018/clinvar_20180429.vcf.gz). Accessed May 2018.

**Ready to submit your research? Choose BMC and benefit from:**

- fast, convenient online submission
- thorough peer review by experienced researchers in your field
- rapid publication on acceptance
- support for research data, including large and complex data types
- gold Open Access which fosters wider collaboration and increased citations
- maximum visibility for your research: over 100M website views per year

**At BMC, research is always in progress.**

Learn more [biomedcentral.com/submissions](https://biomedcentral.com/submissions)





## C Appendix

### CAGI5 splicing challenge: Improved exon skipping and intron retention predictions with MMSplice

Variants frequently affect splicing by changing exon skipping rate or intron retention rate. The fifth edition of the Critical Assessment of Genome Interpretation proposed two splicing prediction challenges based on experimental perturbation assays: Vex-seq, assessing exon skipping, and MaPSy assessing splicing efficiency. Vex-seq data is measured with alternatively spliced exons, while MaPSy measured the variant effect on constitutive splicing. I submitted prediction results from the modular modeling framework (MMSplice). MMSplice performed among the best on both challenges. This article provides insights into the modeling assumptions of MMSplice and its modules.

From the modular model predictions, a linear model was trained to predict the variant effect on exon skipping level ( $\Delta\Psi$ ), similarly also for the splicing efficiency change. Additionally, for the MaPSy challenge, a logistic regression classifier was trained to predict whether a given variant is an exonic splicing mutation. Features for the classifier include MMSplice predicted variant effect and several properties about the affected exon, e.g., exon length. MMSplice outperformed other models in both of the challenges.

Additionally, I show empirically that splice variants have an additive effect on the log odds ratios of  $\Psi$ . As a consequence, variants show on average smaller effect size when the reference  $\Psi$  is close to 0 or 1 and the larger effect size when the reference  $\Psi$  is around 0.5. This highlights an important rule to model variant effect on splicing. Furthermore, in-silico-mutagenesis analysis with MMSplice highlighted known splicing regulatory elements, such as the splicing donor and acceptor, and the Heterogeneous Nuclear Ribonucleoprotein A1 (HNRNPA1) binding site.

**Reprint Denied:** The reprint of this publication was rejected on open-access platforms. The publication can be found at <https://doi.org/10.1002/humu.23788>.

Copyright © 2019 John Wiley & Sons, Inc. Reprint denied.

Cheng, J., Çelik, M. H., Nguyen, Y. D., Avsec, Ž., & Gagneur, J. (2019). CAGI5 splicing challenge: Improved exon skipping and intron retention predictions with MMSplice. *Human Mutation*. DOI: 10.1002/humu.23788

**Contribution of the thesis author:** model design and implementation, data analysis and visualization, literature review, results interpretation, manuscript composition.



# List of Figures

1.1	The central dogma of biology . . . . .	3
1.2	RNA splicing regulation . . . . .	4
1.3	Common patterns of alternative splicing. . . . .	5
1.4	mRNA degradation pathways. . . . .	6
1.5	RNA splicing quantification with RNA-Seq . . . . .	8
1.6	Reporter assays . . . . .	9
2.1	Neural network . . . . .	24
2.2	XOR problem. . . . .	25
2.3	Typical convolutional neural network in genomics . . . . .	26
2.4	Quadratic approximation of the object function with taylor expansion . . . . .	28





# List of Tables

1.1	List of variant effect on splicing prediction tools . . . . .	14
1.2	List of pathogenicity prediction tools with splicing focus . . . . .	15



# Bibliography

- [1] Cheng, J., Maier, K. C., Avsec, Ž., Petra, R. U. & Gagneur, J. Cis-regulatory elements explain most of the mRNA stability variation across genes in yeast. *RNA* **23**, 1648–1659 (2017).
- [2] Cheng, J. *et al.* MMSplice: modular modeling improves the predictions of genetic variant effects on splicing. *Genome Biology* **20**, 48 (2019).
- [3] Cheng, J., Çelik, M. H., Nguyen, Y. D., Avsec, Ž. & Gagneur, J. Cagi5 splicing challenge: Improved exon skipping and intron retention predictions with mmsplice. *Human Mutation* (2019).
- [4] Avsec, Ž. *et al.* The Kipoi repository accelerates community exchange and reuse of predictive models for genomics. *Nature Biotechnology* **37**, 592–600 (2019). URL <https://www.biorxiv.org/content/early/2018/07/24/375345><http://www.ncbi.nlm.nih.gov/pubmed/31138913>.
- [5] Avsec, Ž., Barekatain, M., Cheng, J. & Gagneur, J. Modeling positional effects of regulatory sequences with spline transformations increases prediction accuracy of deep neural networks. *Bioinformatics* **34**, 1261–1269 (2017). URL <http://academic.oup.com/bioinformatics/advance-article/doi/10.1093/bioinformatics/btx727/4636216>.
- [6] Mendel, G. Versuche über pflanzenhybriden. *Verhandlungen des naturforschenden Vereins in Brunn* **4**, 3–47 (1866).
- [7] Venter, J. C. *et al.* The sequence of the human genome. *Science* **291**, 2001 (2001).
- [8] Harrow, J. *et al.* GENCODE: The reference human genome annotation for the ENCODE project. *Genome Research* **22**, 1760–1774 (2012).
- [9] Maurano, M. T. *et al.* Systematic Localization of Common Disease-Associated Variation in Regulatory DNA. *Science* **337**, 1190–1195 (2012). URL <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3771521&tool=pmcentrez&rendertype=abstract>.
- [10] Clark, M. M. *et al.* Meta-analysis of the diagnostic and clinical utility of genome and exome sequencing and chromosomal microarray in children with suspected genetic diseases. *npj Genomic Medicine* **3**, 1–10 (2018). URL <http://dx.doi.org/10.1038/s41525-018-0053-8>.

## BIBLIOGRAPHY

- [11] Scotti, M. M. & Swanson, M. S. RNA mis-splicing in disease. *Nature Reviews Genetics* **17**, 19–32 (2016). URL <http://www.nature.com/doi/10.1038/nrg.2015.3><http://www.ncbi.nlm.nih.gov/pubmed/2659342><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC5993438>.
- [12] Kelemen, O. *et al.* Function of alternative splicing. *Gene* **514**, 1–30 (2013). URL <http://www.ncbi.nlm.nih.gov/pubmed/22909801><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC5632952>.
- [13] Wang, E. T. *et al.* Alternative isoform regulation in human tissue transcriptomes. *Nature* **456**, 470–6 (2008). URL <http://www.ncbi.nlm.nih.gov/pubmed/18978772><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC2593745>.
- [14] Nilsen, T. W. & Graveley, B. R. Expansion of the eukaryotic proteome by alternative splicing. *Nature* **463**, 457 (2010).
- [15] Neves, G., Zucker, J., Daly, M. & Chess, A. Stochastic yet biased expression of multiple Dscam splice variants by individual cells. *Nature Genetics* **36**, 240–246 (2004).
- [16] Blencowe, B. J. Alternative Splicing: New Insights from Global Analyses. *Cell* **126**, 37–47 (2006).
- [17] Wang, Z. & Burge, C. B. Splicing regulation: from a parts list of regulatory elements to an integrated splicing code. *Rna* **14**, 802–813 (2008).
- [18] Turunen, J. J., Niemelä, E. H., Verma, B. & Frilander, M. J. The significant other: splicing by the minor spliceosome. *Wiley Interdisciplinary Reviews: RNA* **4**, 61–76 (2013).
- [19] Wang, Z. & Burge, C. B. Splicing regulation : From a parts list of regulatory elements to an integrated splicing code Splicing regulation : From a parts list of regulatory elements to an integrated splicing code. *RNA* **14**, 802–813 (2008).
- [20] Kremer, L. S. *et al.* Genetic diagnosis of Mendelian disorders via RNA sequencing. *Nature Communications* **8**, 15824 (2017). URL <http://biorxiv.org/lookup/doi/10.1101/066738><http://www.ncbi.nlm.nih.gov/pubmed/28604674><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC5499207>.
- [21] Ibrahim, E. C. *et al.* Weak definition of IKBKAP exon 20 leads to aberrant splicing in familial dysautonomia. *Human Mutation* **28**, 41–53 (2007). URL <http://www.ncbi.nlm.nih.gov/pubmed/16964593>.
- [22] Soemedi, R. *et al.* Pathogenic variants that alter protein code often disrupt splicing. *Nature Genetics* **49**, 848–855 (2017). URL <http://www.nature.com/doi/10.1038/ng.3837>.

- [23] Parker, R. RNA degradation in *Saccharomyces cerevisiae*. *Genetics* **191**, 671–702 (2012).
- [24] Sun, M. *et al.* Comparative dynamic transcriptome analysis (cDTA) reveals mutual feedback between mRNA synthesis and degradation. *Genome Research* **22**, 1350–9 (2012). URL <http://www.ncbi.nlm.nih.gov/pubmed/22466169><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC3396375>.
- [25] Neymotin, B., Athanasiadou, R. & Gresham, D. Determination of in vivo RNA kinetics using RATE-seq. *RNA* **20**, 1645–52 (2014). URL <http://www.ncbi.nlm.nih.gov/pubmed/25161313><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4174445>.
- [26] Eser, P. *et al.* Determinants of RNA metabolism in the *Schizosaccharomyces pombe* genome. *Molecular Systems Biology* **12**, 857–857 (2016).
- [27] Schwalb, B. *et al.* TT-seq maps the human transient transcriptome. *Science* **352**, 1225–8 (2016). URL <http://www.ncbi.nlm.nih.gov/pubmed/27257258>.
- [28] Presnyak, V. *et al.* Codon optimality is a major determinant of mRNA stability. *Cell* **160**, 1111–1124 (2015). URL <http://www.ncbi.nlm.nih.gov/pubmed/25768907><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4359748>.
- [29] Nonet, M., Sweetser, D. & Young, R. A. Functional redundancy and structural polymorphism in the large subunit of RNA polymerase II. *Cell* **50**, 909–915 (1987).
- [30] Pervouchine, D. D., Knowles, D. G. & Guigó, R. Intron-centric estimation of alternative splicing from rna-seq data. *Bioinformatics* **29**, 273–274 (2013).
- [31] Finotello, F. *et al.* Reducing bias in rna sequencing data: a novel approach to compute counts. *BMC bioinformatics* **15**, S7 (2014).
- [32] Kakaradov, B., Xiong, H. Y., Lee, L. J., Jojic, N. & Frey, B. J. Challenges in estimating percent inclusion of alternatively spliced junctions from rna-seq data. In *BMC bioinformatics*, vol. 13, S11 (BioMed Central, 2012).
- [33] Katz, Y., Wang, E. T., Airoidi, E. M. & Burge, C. B. Analysis and design of RNA sequencing experiments for identifying isoform regulation. *Nature Methods* **7**, 1009–15 (2010). URL <http://www.nature.com/doifinder/10.1038/nmeth.1528><http://www.ncbi.nlm.nih.gov/pubmed/15215377><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC441531><http://www.ncbi.nlm.nih.gov/pubmed/21057496><http://www.pubmedcentral.nih.gov/articlerender.f.9605103>.
- [34] Vaquero-Garcia, J. *et al.* A new view of transcriptome complexity and regulation through the lens of local splicing variations. *eLife* **5**, e11752 (2016). URL <http://elifesciences.org/lookup/doi/10.7554/eLife.11752><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4711175>.

## BIBLIOGRAPHY

- <http://www.ncbi.nlm.nih.gov/pubmed/26829591><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4801060>. arXiv:1011.1669v3.
- [35] Shen, S. *et al.* rMATS: robust and flexible detection of differential alternative splicing from replicate RNA-Seq data. *Proceedings of the National Academy of Sciences of the United States of America* **111**, E5593–601 (2014). URL <http://www.pnas.org/lookup/doi/10.1073/pnas.1419161111><http://www.ncbi.nlm.nih.gov/pubmed/25480548><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4280593>. arXiv:1408.1149.
- [36] Li, Y. I. *et al.* Annotation-free quantification of RNA splicing using LeafCutter. *Nature Genetics* **50**, 151–158 (2018). URL <http://www.ncbi.nlm.nih.gov/pubmed/29229983><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC5742080>.
- [37] Kahles, A., Ong, C. S., Zhong, Y. & Ratsch, G. Spladder: identification, quantification and testing of alternative splicing events from rna-seq data. *Bioinformatics* **32**, 1840–1847 (2016).
- [38] Long, Q. *et al.* Gata-1 expression pattern can be recapitulated in living transgenic zebrafish using gfp reporter gene. *Development* **124**, 4105–4111 (1997).
- [39] Liu, H.-X., Zhang, M. & Krainer, A. R. Identification of functional exonic splicing enhancer motifs recognized by individual sr proteins. *Genes & development* **12** (1998).
- [40] Hoekema, A., Kastelein, R. A., Vasser, M. & de Boer, H. A. Codon replacement in the PGK1 gene of *Saccharomyces cerevisiae*: experimental approach to study the role of biased codon usage in gene expression. *Molecular and Cellular Biology* **7**, 2914–24 (1987). URL <http://www.ncbi.nlm.nih.gov/pubmed/2823108><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC367910>.
- [41] Cubitt, A. B. *et al.* Understanding, improving and using green fluorescent proteins. *Trends in Biochemical Sciences* **20**, 448–455 (1995).
- [42] Kheradpour, P. *et al.* Systematic dissection of regulatory motifs in 2000 predicted human enhancers using a massively parallel reporter assay. *Genome Research* **23**, 800–811 (2013).
- [43] Melnikov, A. *et al.* Systematic dissection and optimization of inducible enhancers in human cells using a massively parallel reporter assay. *Nature Biotechnology* **30**, 271 (2012).
- [44] White, M. A. Understanding how cis-regulatory function is encoded in dna sequence using massively parallel reporter assays and designed sequences. *Genomics* **106**, 165–170 (2015).

- [45] White, M. A., Myers, C. A., Corbo, J. C. & Cohen, B. A. Massively parallel in vivo enhancer assay reveals that highly local features determine the cis-regulatory function of chip-seq peaks. *Proceedings of the National Academy of Sciences* **110**, 11952–11957 (2013).
- [46] Oikonomou, P., Goodarzi, H. & Tavazoie, S. Systematic identification of regulatory elements in conserved 3' utrs of human transcripts. *Cell reports* **7**, 281–292 (2014).
- [47] Smith, R. P. *et al.* Massively parallel decoding of mammalian regulatory sequences supports a flexible organizational model. *Nature Genetics* **45**, 1021 (2013).
- [48] Ernst, J. *et al.* Genome-scale high-resolution mapping of activating and repressive nucleotides in regulatory regions. *Nature Biotechnology* **34**, 1180 (2016).
- [49] Rosenberg, A. B., Patwardhan, R. P., Shendure, J. & Seelig, G. Learning the sequence determinants of alternative splicing from millions of random sequences. *Cell* **163**, 698–711 (2015). URL <http://dx.doi.org/10.1016/j.cell.2015.09.054><http://www.ncbi.nlm.nih.gov/pubmed/26496609>.
- [50] Patwardhan, R. P. *et al.* Massively parallel functional dissection of mammalian enhancers in vivo. *Nature Biotechnology* **30**, 265 (2012).
- [51] Rabani, M., Pieper, L., Chew, G.-L. & Schier, A. F. A massively parallel reporter assay of 3' utr sequences identifies in vivo rules for mrna degradation. *Molecular Cell* **68**, 1083–1094 (2017).
- [52] Kwasnieski, J. C., Mogno, I., Myers, C. A., Corbo, J. C. & Cohen, B. A. Complex effects of nucleotide variants in a mammalian cis-regulatory element. *Proceedings of the National Academy of Sciences* **109**, 19498–19503 (2012).
- [53] Tewhey, R. *et al.* Direct identification of hundreds of expression-modulating variants using a multiplexed reporter assay. *Cell* **165**, 1519–1529 (2016).
- [54] Adamson, S. I., Zhan, L. & Graveley, B. R. Vex-seq: high-throughput identification of the impact of genetic variation on pre-mrna splicing efficiency. *Genome Biology* **19**, 71 (2018).
- [55] Ke, S. *et al.* Quantitative evaluation of all hexamers as exonic splicing elements. *Genome Research* **21**, 1360–1374 (2011).
- [56] Ke, S. *et al.* Saturation mutagenesis reveals manifold determinants of exon definition. *Genome Research* **28**, 11–24 (2018).
- [57] Ran, F. A. *et al.* Genome engineering using the crispr-cas9 system. *Nature Protocols* **8**, 2281 (2013).
- [58] Shalem, O. *et al.* Genome-scale crispr-cas9 knockout screening in human cells. *Science* **343**, 84–87 (2014).

## BIBLIOGRAPHY

- [59] Findlay, G. M. *et al.* Accurate classification of brca1 variants with saturation genome editing. *Nature* **562**, 217 (2018).
- [60] Rajagopal, N. *et al.* High-throughput mapping of regulatory dna. *Nature Biotechnology* **34**, 167 (2016).
- [61] Consortium, T. . G. P. A map of human genome variation from population-scale sequencing. *Nature* **467**, 1061–1073 (2010). URL <http://www.nature.com/articles/nature09534>.
- [62] Sudlow, C. *et al.* UK biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLoS Medicine* **12**, e1001779 (2015). URL <http://www.ncbi.nlm.nih.gov/pubmed/25826379><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4380465>.
- [63] Michaelson, J. J. *et al.* Whole-genome sequencing in autism identifies hot spots for de novo germline mutation. *Cell* **151**, 1431–1442 (2012).
- [64] Acuna-Hidalgo, R., Veltman, J. A. & Hoischen, A. New insights into the generation and role of de novo mutations in health and disease. *Genome Biology* **17**, 241 (2016).
- [65] Landrum, M. J. *et al.* ClinVar: Public archive of relationships among sequence variation and human phenotype. *Nucleic Acids Research* **42**, D980–5 (2014). URL <https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/gkt1113><http://www.ncbi.nlm.nih.gov/pubmed/24234437><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC3965032>.
- [66] Shabalin, A. A. Matrix eQTL: Ultra fast eQTL analysis via large matrix operations. *Bioinformatics* **28**, 1353–1358 (2012).
- [67] Hirschhorn, J. N. & Daly, M. J. Genome-wide association studies for common diseases and complex traits. *Nature Reviews Genetics* **6**, 95 (2005).
- [68] Li, Y. I. *et al.* RNA splicing is a primary link between genetic variation and disease. *Science* **352**, 600–604 (2016). URL <http://www.sciencemag.org/cgi/doi/10.1126/science.aad9417><http://www.sciencemag.org/lookup/doi/10.1126/science.aad9417><http://www.ncbi.nlm.nih.gov/pubmed/27126046><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC5182069>.
- [69] Shendure, J., Findlay, G. M. & Snyder, M. W. Genomic Medicine—Progress, Pitfalls, and Promise. *Cell* **177**, 45–57 (2019). URL <https://doi.org/10.1016/j.cell.2019.02.003>.
- [70] Stephens, Z. D. *et al.* Big data: Astronomical or genetical? *PLoS Biology* **13**, 1–11 (2015).



- [71] Park, Y. & Kellis, M. Deep learning for regulatory genomics. *Nature Biotechnology* **33**, 825–826 (2015). URL <http://www.nature.com/doi/10.1038/nbt.3313>.
- [72] Angermueller, C., Pärnamaa, T., Parts, L. & Oliver, S. Deep Learning for Computational Biology. *Molecular Systems Biology* 878 (2016).
- [73] Eraslan, G., Avsec, Ž., Gagneur, J. & Theis, F. J. Deep learning: new computational modelling techniques for genomics. *Nature Reviews Genetics* (2019).
- [74] Zhou, J. & Troyanskaya, O. G. Predicting effects of noncoding variants with deep learning-based sequence model. *Nature Methods* **12**, 931–934 (2015). URL <http://www.nature.com/doi/10.1038/nmeth.3547>. 15334406.
- [75] Alipanahi, B., Delong, A., Weirauch, M. T. & Frey, B. J. Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. *Nature Biotechnology* **33**, 831–8 (2015). URL <http://dx.doi.org/10.1038/nbt.3300><http://www.ncbi.nlm.nih.gov/pubmed/26213851>. 9605103.
- [76] Godet, I. & Gilkes, D. M. BRCA1 and BRCA2 mutations and treatment strategies for breast cancer. *Integrative Cancer Science and Therapeutics* **4** (2017). URL <http://www.ncbi.nlm.nih.gov/pubmed/28706734><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC5505673>.
- [77] Richards, S. *et al.* Standards and guidelines for the interpretation of sequence variants: A joint consensus recommendation of the American College of Medical Genetics and Genomics and the Association for Molecular Pathology. *Genetics in Medicine* **17**, 405–424 (2015). 15334406.
- [78] Mort, M. *et al.* MutPred Splice: machine learning-based prediction of exonic variants that disrupt splicing. *Genome Biology* **15**, R19 (2014). URL <http://genomebiology.biomedcentral.com/articles/10.1186/gb-2014-15-1-r19>.
- [79] Pertea, M., Lin, X. & Salzberg, S. L. Genesplicer: a new computational method for splice site prediction. *Nucleic Acids Research* **29**, 1185–1190 (2001).
- [80] Reese, M. G., Eeckman, F. H., Kulp, D. & Haussler, D. Improved splice site detection in Genie. *Journal of Computational Biology* **4**, 311–23 (1997). URL <http://portal.acm.org/citation.cfm?doid=267521.267766><http://www.ncbi.nlm.nih.gov/pubmed/9278062>.
- [81] Brunak, S., Engelbrecht, J. & Knudsen, S. Prediction of human mRNA donor and acceptor sites from the DNA sequence. *Journal of Molecular Biology* **220**, 49–65 (1991).
- [82] Yeo, G. & Burge, C. B. Maximum entropy modeling of short sequence motifs with applications to RNA splicing signals. *Journal of Computational Biology* **11**, 377–94 (2004). URL <http://www.ncbi.nlm.nih.gov/pubmed/15285897>.

## BIBLIOGRAPHY

- [83] Sonnenburg, S., Schweikert, G., Philips, P., Behr, J. & Ratsch, G. Accurate splice site prediction using support vector machines. In *BMC Bioinformatics*, vol. 8, S7 (BioMed Central, 2007).
- [84] McCulloch, C. E. & Neuhaus, J. M. Generalized Linear Mixed Models. In *International Encyclopedia of the Social & Behavioral Sciences: Second Edition*, 845–852 (2015).
- [85] Barash, Y. *et al.* Deciphering the splicing code. *Nature* **465**, 53–59 (2010). URL <http://www.nature.com/doi/10.1038/nature09000>.
- [86] Xiong, H. Y. *et al.* The human splicing code reveals new insights into the genetic determinants of disease. *Science* **347**, 1254806–1254806 (2015). URL <http://www.sciencemag.org/cgi/doi/10.1126/science.1254806>.
- [87] Foat, B. C., Houshmandi, S. S., Olivas, W. M. & Bussemaker, H. J. Profiling condition-specific, genome-wide regulation of mRNA stability in yeast. *Proceedings of the National Academy of Sciences of the United States of America* **102**, 17675–17680 (2005). URL <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1295595&tool=pmcentrez&drendertype=abstract>.
- [88] Skeeles, L. E., Fleming, J. L., Mahler, K. L. & Toland, A. E. The Impact of 3' UTR Variants on Differential Expression of Candidate Cancer Susceptibility Genes. *PLoS ONE* **8** (2013).
- [89] Dvir, S. *et al.* Deciphering the rules by which 5'-UTR sequences affect protein expression in yeast. *Proceedings of the National Academy of Sciences of the United States of America* **110**, E2792–801 (2013). URL <http://www.ncbi.nlm.nih.gov/pubmed/23832786><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC3725075>.
- [90] Elemento, O., Slonim, N. & Tavazoie, S. A Universal Framework for Regulatory Element Discovery across All Genomes and Data Types. *Molecular Cell* **28**, 337–350 (2007). NIHMS150003.
- [91] Zhao, W. *et al.* Massively parallel functional annotation of 3' untranslated regions. *Nature Biotechnology* **32**, 387–391 (2014). URL <http://dx.doi.org/10.1038/nbt.2851>.
- [92] Litterman, A. J. *et al.* A massively parallel 3' UTR reporter assay reveals relationships between nucleotide content, sequence conservation, and mRNA destabilization. *Genome Research* 896–906 (2019). URL <http://genome.cshlp.org/lookup/doi/10.1101/gr.242552.118>.
- [93] Lee, R. C., Feinbaum, R. L. & Ambros, V. The *C. elegans* heterochronic gene *lin-4* encodes small RNAs with antisense complementarity to *lin-14*. *Cell* **75**, 843–854 (1993). URL <http://www.ncbi.nlm.nih.gov/pubmed/8252621>.

- [94] Neymotin, B., Ettore, V. & Gresham, D. Multiple Transcript Properties Related to Translation Affect mRNA Degradation Rates in *Saccharomyces cerevisiae*. *G3 (Bethesda, Md.)* **6**, 3475–3483 (2016). URL <http://www.ncbi.nlm.nih.gov/pubmed/27633789><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC5100846>.
- [95] Kircher, M. *et al.* A general framework for estimating the relative pathogenicity of human genetic variants. *Nature Genetics* **46**, 310–315 (2014). URL <http://dx.doi.org/10.1038/ng.2892>. NIHMS150003.
- [96] Gelfman, S. *et al.* Annotating pathogenic non-coding variants in genic regions. *Nature Communications* **8**, 1–10 (2017). URL <http://dx.doi.org/10.1038/s41467-017-00141-2>.
- [97] Cooper, D. N., Ball, E. V. & Krawczak, M. The human gene mutation database. *Nucleic Acids Research* **26**, 285–287 (1998).
- [98] Lek, M. *et al.* Analysis of protein-coding genetic variation in 60,706 humans. *Nature* **536**, 285–291 (2016). URL <http://www.nature.com/doi/10.1038/nature19057>. 030338.
- [99] Bishop, C. M. *Pattern Recognition and Machine Learning* (springer, 2006).
- [100] Goodfellow, I., Bengio, Y. & Courville, A. *Deep Learning* (MIT press, 2016).
- [101] Nelder, J. A. & Wedderburn, R. W. Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)* **135**, 370–384 (1972).
- [102] Ferrari, S. L. & Cribari-Neto, F. Beta regression for modelling rates and proportions. *Journal of Applied Statistics* **31**, 799–815 (2004).
- [103] McCulloch, C. E. Maximum Likelihood Algorithms for Generalized Linear Mixed Models. *Journal of the American Statistical Association* **92**, 162–170 (1997).
- [104] Ng, A. Machine learning yearning. URL: [http://www.mlyearning.org/\(96\)](http://www.mlyearning.org/(96)) (2017).
- [105] Krizhevsky, A., Sutskever, I. & Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 1097–1105 (2012).
- [106] Hochreiter, S. & Schmidhuber, J. Long Short-Term Memory. *Neural Computation* **9**, 1735–1780 (1997).
- [107] Yu, F. & Koltun, V. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122* (2015).
- [108] Dunham, I. *et al.* An integrated encyclopedia of DNA elements in the human genome. *Nature* **489**, 57–74 (2012).

## BIBLIOGRAPHY

- [109] GTEx Consortium. The Genotype-Tissue Expression (GTEx) project. *Nature Genetics* **45**, 580–5 (2013). URL <http://www.nature.com/doi/10.1038/ng.2653><http://www.ncbi.nlm.nih.gov/pubmed/23715323><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4010069>. NIHMS150003.
- [110] Cancer Genome Atlas Research Network *et al.* The Cancer Genome Atlas Pan-Cancer analysis project. *Nature Genetics* **45**, 1113–20 (2013). URL <http://dx.doi.org/10.1038/ng.2764><http://www.ncbi.nlm.nih.gov/pubmed/24071849><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC3919969>.
- [111] Stormo, G. D., Schneider, T. D., Gold, L. & Ehrenfeucht, A. Use of the ‘perceptron’ algorithm to distinguish translational initiation sites in *E. coli*. *Nucleic Acids Research* **10**, 2997–3011 (1982).
- [112] Kozak, M. The scanning model for translation: An update (1989).
- [113] Shrikumar, A., Greenside, P. & Kundaje, A. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning*, 3145–3153 (JMLR. org, 2017).
- [114] Shrikumar, A. *et al.* TF-MoDISco v0.4.2.2-alpha: Technical Note (2018). URL <http://arxiv.org/abs/1811.00416>. 1811.00416.
- [115] Battiti, R. First- and Second-Order Methods for Learning: Between Steepest Descent and Newton’s Method. *Neural Computation* **4**, 141–166 (2008).
- [116] Sutskever, I., Martens, J., Dahl, G. E. & Hinton, G. E. On the importance of initialization and momentum in deep learning. *International Conference on Machine Learning, 2013* **28**, 1139–1147 (2013). URL <http://dblp.uni-trier.de/db/conf/icml/icml2013.html#{#}SutskeverMDH13>.
- [117] Lee, D. & Myung, K. ADAM: Method for Stochastic Optimization. *2017 IEEE International Conference on Consumer Electronics, ICCE 2017* 434–435 (2017). URL <http://arxiv.org/abs/1412.6980>. 1412.6980.
- [118] Kawaguchi, K. Deep learning without poor local minima. In *Advances in Neural Information Processing Systems*, 586–594 (2016).
- [119] Choromanska, A., Henaff, M., Mathieu, M., Arous, G. B. & LeCun, Y. The loss surfaces of multilayer networks. In *Artificial Intelligence and Statistics*, 192–204 (2015).
- [120] Bergstra, J. S., Bardenet, R., Bengio, Y. & Kégl, B. Algorithms for hyperparameter optimization. In *Advances in Neural Information Processing Systems*, 2546–2554 (2011).

- [121] Jagadeesh, K. A. *et al.* S-CAP extends pathogenicity prediction to genetic variants that affect RNA splicing. *Nature Genetics* **51**, 755–763 (2019). URL <http://www.ncbi.nlm.nih.gov/pubmed/30804562>.
- [122] Wachutka, L., Caizzi, L., Gagneur, J. & Cramer, P. Global donor and acceptor splicing site kinetics in human cells. *eLife* **8** (2019).
- [123] Bogard, N., Linder, J., Rosenberg, A. B. & Seelig, G. A Deep Neural Network for Predicting and Engineering Alternative Polyadenylation. *Cell* 1–16 (2019). URL <https://linkinghub.elsevier.com/retrieve/pii/S0092867419304982>.
- [124] Rabani, M., Pieper, L., Chew, G. L. & Schier, A. F. A Massively Parallel Reporter Assay of 3'UTR Sequences Identifies In Vivo Rules for mRNA Degradation. *Molecular Cell* **68**, 1083–1094.e5 (2017). URL <https://doi.org/10.1016/j.molcel.2017.11.014>.
- [125] Gasperini, M. *et al.* A Genome-wide Framework for Mapping Gene Regulation via Cellular Genetic Screens. *Cell* **176**, 377–390.e19 (2019). URL <http://www.ncbi.nlm.nih.gov/pubmed/30612741>.
- [126] Yeo, G., Holste, D., Kreiman, G. & Burge, C. B. Variation in alternative splicing across human tissues. *Genome Biology* **5**, R74 (2004). URL [http://eutils.ncbi.nlm.nih.gov/entrez/eutils/elink.fcgi?dbfrom=pubmed{&}id=15461793{&}retmode=ref{&}cmd=prlinks\\$\\delimiter"026E30F\\$npapers2://publication/doi/10.1186/gb-2004-5-10-r74](http://eutils.ncbi.nlm.nih.gov/entrez/eutils/elink.fcgi?dbfrom=pubmed{&}id=15461793{&}retmode=ref{&}cmd=prlinks$\\delimiter).
- [127] Consortium, T. G. *et al.* The Genotype-Tissue Expression (GTEx) pilot analysis: Multitissue gene regulation in humans. *Science* **348**, 648–660 (2015). URL <http://www.sciencemag.org/content/348/6235/648.full>.
- [128] Long, J., Shelhamer, E. & Darrell, T. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3431–3440 (2015).
- [129] Zhou, J. *et al.* Whole-genome deep-learning analysis identifies contribution of noncoding mutations to autism risk. *Nature Genetics* (2019). URL <http://www.nature.com/articles/s41588-019-0420-0>.
- [130] Leung, M. K. K., DeLong, A., Alipanahi, B. & Frey, B. J. Machine Learning in Genomic Medicine: A Review of Computational Problems and Data Sets. *Proceedings of the IEEE* **104**, 176–197 (2016). URL <http://ieeexplore.ieee.org/document/7347331/>.
- [131] Zhou, J. *et al.* Deep learning sequence-based ab initio prediction of variant effects on expression and disease risk. *Nature Genetics* **50**, 1171 (2018).
- [132] Schwanhäusser, B. *et al.* Global quantification of mammalian gene expression control. *Nature* **473**, 337 (2011).