Deciphering splicing with sparse regression techniques in the era of high-throughput RNA sequencing.

Elsa Bernard

THÈSE DE DOCTORAT

de l'Université de recherche Paris Sciences et Lettres
PSL Research University

Préparée à MINES ParisTech

Deciphering splicing with sparse regression techniques in the era of high-throughput RNA sequencing

Etude de l'épissage grâce à des techniques de régression parcimonieuse dans l'ère du séquençqge haut débit de l'ARN

Ecole doctorale n°432

ECOLE DOCTORALE SCIENCES DES METIERS DE L'INGENIEUR

Spécialité BIO-INFORMATIQUE

COMPOSITION DU JURY :

M. Franck PICARD
LBBE, Président

M. Daniel GAUTHERET
Université Paris-Sud, Rapporteur

M. Wolfgang HUBER
EMBL, Rapporteur

M. Didier AUBOEUF
ENS Lyon, Examinateur

M. Claude HOUDAYER
Institut Curie, Examinateur

M. Jean-Philippe VERT
MINES ParisTech, Examinateur

Soutenue par Elsa BERNARD
le 21 septembre 2016

Dirigée par **Jean-Philippe VERT**

À mes parents

Abstract

The number of protein-coding genes in a human, a nematode and a fruit fly are roughly equal. The paradoxical miscorrelation between the number of genes in an organism's genome and its phenotypic complexity finds an explanation in the alternative nature of splicing in higher organisms.

Alternative splicing largely increases the functional diversity of proteins encoded by a limited number of genes. It is known to be involved in cell fate decision and embryonic development, but also appears to be dysregulated in inherited and acquired human genetic disorders, in particular in cancers.

High-throughput RNA sequencing technologies allow us to measure and question splicing at an unprecedented resolution. However, while the cost of sequencing RNA decreases and throughput increases, many computational challenges arise from the discrete and local nature of the data. In particular, the task of inferring alternative transcripts requires a non-trivial deconvolution procedure.

In this thesis, we contribute to deciphering alternative transcript expressions and alternative splicing events from high-throughput RNA sequencing data.

We propose new methods to accurately and efficiently detect and quantify alternative transcripts. Our methodological contributions largely rely on sparse regression techniques and takes advantage of network flow optimization techniques. Besides, we investigate means to query splicing abnormalities for clinical diagnosis purposes. We suggest an experimental protocol that can be easily implemented in routine clinical practice, and present new statistical models and algorithms to quantify splicing events and measure how abnormal these events might be in patient data compared to wild-type situations.

Résumé

Le nombre de gènes codant pour des protéines chez l'homme, le vers rond et la mouche des fruits est du même ordre de grandeur. Cette absence de correspondance entre le nombre de gènes d'un eucaryote et sa complexité phénotypique s'explique en partie par le caractère alternatif de l'épissage.

L'épissage alternatif augmente considérablement le répertoire fonctionnel de protéines codées par un nombre limité de gènes. Ce mécanisme, très actif lors du développement embryonnaire, participe au devenir cellulaire. De nombreux troubles génétiques, hérités ou acquis (en particulier certains cancers), se caractérisent par une altération de son fonctionnement.

Les technologies de séquençage à haut débit de l'ARN donnent accès à une information plus riche sur le mécanisme de l'épissage. Cependant, si la lecture à haut débit des séquences d'ARN est plus rapide et moins coûteuse, les données qui en sont issues sont complexes et nécessitent le développement d'outils algorithmiques pour leur interprétation. En particulier, la reconstruction des transcrits alternatifs requiert une étape de déconvolution non triviale.

Dans ce contexte, cette thèse participe à l'étude des événements d'épissage et des transcrits alternatifs à partir de données de séquençage à haut débit de l'ARN.

Nous proposons de nouvelles méthodes pour reconstruire et quantifier les transcrits alternatifs de façon plus efficace et précise. Nos contributions méthodologiques impliquent des techniques de régression parcimonieuse, basées sur l'optimisation convexe et sur des algorithmes de flots. Nous étudions également une procédure pour détecter des anomalies d'épissage dans un contexte de diagnostic clinique. Nous suggérons un protocole expérimental facilement opérant et développons de nouveaux modèles statistiques et algorithmes pour quantifier des événements d'épissage et mesurer leur degré d'anormalité chez le patient.

Contents

List of Figures

List of Tables

Abbreviations

DNA	**D**eoxyribo**N**ucleic **A**cid
RNA	**R**ibo**N**ucleic **A**cid
A	**A**denine
T	**T**hymine
G	**G**uanine
C	**C**ytosine
bp	**b**ase **p**air
UTR	**U**n**T**ranslated **R**egion
ESE	**E**xonic **S**plicing **E**nhancer
ESS	**E**xonic **S**plicing **S**ilencer
ISE	**I**ntronic **S**plicing **E**nhancer
ISS	**I**ntronic **S**plicing **S**ilencer
ESC	**E**mbryonic **S**tem **C**ell
NMD	**N**onsense **M**ediated **D**ecay
ASO	**A**nti**S**ense **O**ligonucleotide
EMT	**E**phytelial **M**esenchymal **T**ransition
EST	**E**xpressed **S**equence **T**ag
PCR	**P**olymerase **C**hain **R**eaction
NGS	**N**ext **G**eneration **S**equencing
RNA-seq	**RNA-seq**uencing
DAG	**D**irected **A**cyclic **G**raph
VUS	**V**ariant of **U**nknown **S**ignificance

Chapter 1

Preambule

Through alternative splicing of precursor messenger RNAs, eukaryote genes produce multiple transcript isoforms that may lead to proteins with distinct or even opposite functions.

Alternative splicing not only greatly increases the repertoire of proteins that can be encoded by a genome, it is also a fundamental regulatory mechanism of gene expression at the crossroad between transcription and translation. Alternative splicing is deeply involved in cell fate decision and tissue differentiation.

The importance of alternative splicing is underscored by the fact that splicing defects are responsible for many human diseases such as retinitis pigmentosa or Duchenne muscular dystrophy, and that splicing aberrations are believed to contribute to tumor progression in several cancers.

Detecting transcript isoforms in different cell types or samples is therefore crucial to understand the cells' regulatory programs and to identify splicing variants responsible for diseases. Furthermore, fully characterizing the transcripts expressed in tumor samples will contribute to our understanding of cancer mechanisms, provide new diagnostic and prognostic biomarkers and reveal possible drug targets, improving personalized patient treatment.

Recent technological advances decreased the cost of RNA sequencing while increasing the throughput. This allows the profiling of numerous RNA landscapes from various species, tissues and conditions and to get closer to RNA profiling in routine clinical practice.

High-throughput RNA sequencing is accelerating our understanding of alternative splicing regulation and dysregulation and gives a better insight into fascinating questions such as (i) how

much alternative splicing contributes to cell fate decision, (ii) to what extent alternative splicing events are functionally relevant, or (iii) whether there are splicing aberrations that drive tumorigenesis.

However, while an accurate reconstruction and quantification of transcript isoforms is a crucial step to answer the above questions and for downstream analysis such as differential analysis of transcript abundances, the task is not trivial due to the nature of RNA sequencing data. Indeed, recovering the structure of the transcripts and estimating their abundances from this data need an accurate deconvolution procedure. Furthermore, their discrete nature requires an appropriate statistical modeling, and their high dimensionality asks for the development of efficient algorithmic tools.

The contributions of this thesis lie in the fields of transcriptome assembly and alternative splicing events quantification from high-throughput RNA sequencing. We propose new methods to reconstruct transcript isoforms from one or several RNA sequencing samples, and we investigate means to query splicing abnormalities in a clinical diagnosis context.

Organization and contributions of the thesis

We detail below the organization of the thesis, and highlight, when appropriate, our contributions to the fields of transcriptome assembly and alternative splicing events quantification.

- Chapter 2 is an introductory chapter that briefly reviews the alternative splicing process, mentions some of its functional properties and discusses its implication in human diseases as well as emerging therapies tailored to correct splicing abnormalities.
- Chapter 3 is an other introductory chapter that surveys sequencing and profiling protocols developed since the 90's and which give access to alternative splicing, with a focus on modern high-throughput RNA sequencing that emerged a decade ago. We also describe the computational challenges associated with RNA sequencing data, and review to the best of our knowledge the state-of-the-art methods to assemble and quantify transcript isoforms. We finally introduce the notions of ℓ_1-norm penalization and network flow optimization that we intensively use in the following chapters.

- Chapter 4 describes a new method to reconstruct and quantify transcript isoforms from RNA sequencing data. The main novelty of our approach is to translate a computationally hard sparse regression problem formulated with a ℓ_1-penalized maximum likelihood estimation into a network flow optimization problem that can be solved very efficiently.

- Chapter 5 extends the sparse regression setting of the previous chapter to the joint analysis of several RNA sequencing samples. We formulate a convex problem that allows us to share information across samples when inferring transcript isoforms, hence increasing the power of the statistical inference and resulting performances.

- Chapter 6 describes a clinical diagnosis tool to detect and quantify alternative splicing events as well as full-length transcripts from targeted RNA sequencing experiments where the sequencing efforts are concentrated on a subset of the transcriptome. Our method focuses on revealing splicing abnormalities by measuring discrepancies between patient estimates and wild-type distributions derived from control samples. We apply our methodology on RNA sequencing data from patients characterized by mutations in a breast cancer susceptibility gene, and experimentally validate some of our results.

- Chapter 7 concludes the thesis by summarizing the main results and giving some prospects on how to extend the proposed methodologies to other emerging RNA sequencing protocols and on how the techniques we developed during the thesis could be used to answer other molecular biology questions.

Chapter 2

Splicing: from molecular mechanisms to personalized therapies

"The discovery of split genes has been of fundamental importance for today's basic research in biology, as well as for more medically oriented research concerning the development of cancer and other diseases"

"the genetic message, which gives rise to a particular product, is not definitely established at the stage when the RNA is first synthesized. Instead, it is the splicing pattern that determines the nature of the final product"

Nobel Prize Press Release, 1993.

Ce chapitre introductif fournit aux lecteurs les clés pour comprendre comment les eucaryotes peuvent exprimer plusieurs ARN messagers à partir d'un unique gène. Les notions d'épissage, d'épissage alternatif et de transcription alternative sont donc introduites. Les aspects fonctionnels de l'épissage sont également discutés, son rôle adaptatif et son implication dans le devenir cellulaire. Enfin, la dérégulation de l'épissage dans plusieurs maladies génétiques comme le cancer et l'émergence de thérapies ciblant les dysfonctionnements de l'épissage sont mentionnées.

In this introductory chapter, we start by explaining how eukaryotes can express several messenger RNAs (mRNAs) from the same gene, that is we introduce the concepts of splicing, alternative splicing and alternative transcription. We then discuss some functional aspects of alternative splicing as a fundamental gene expression regulatory mechanism that shows adaptative significance and is deeply involved in cell fate decision. We finally illustrate how alternative splicing can be dysregulated in human diseases and in particular in cancer, before discussing certain emerging therapies tailored to target splicing abnormalites.

2.1 Molecular mechanisms resulting in the expression of transcript isoforms

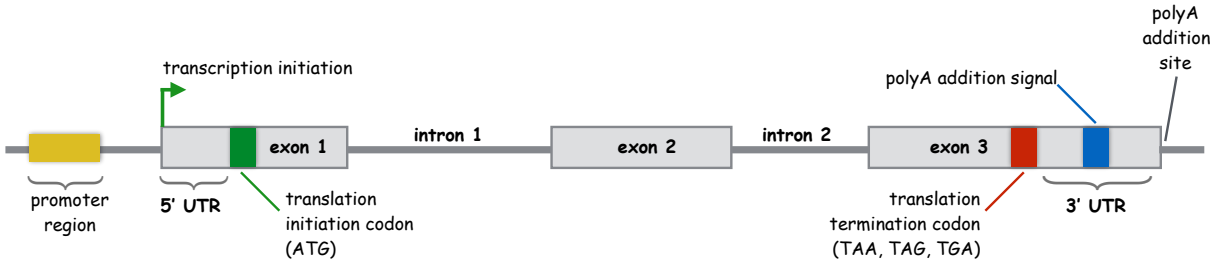
In this section we describe the molecular mechanisms behind splicing and resulting in the expression of several transcript isoforms from the same locus. We do not claim that the following explanations would satisfy the curiosity of a molecular biologist, but we hope they can benefit non-specialists by introducing some key concepts. In particular, we do not detail the different proteins known to be involved in the splicing machinery and their mechanisms of action, but we rather give a schematic view of their effects and refer to the literature for more detailed explanations of molecular mechanisms.

2.1.1 A bit of history: pre-mRNA splicing

The gene expression field made an important step forward in the late 80's when the split nature of most eukaryotic genes was discovered. In 1977, several groups working with adenoviruses that infect and replicate in mammalian cells obtained surprising results: RNA molecules from infected cells containing sequences from non-contiguous sites in the viral genome (Berget et al., 1977; Chow et al., 1977). What they termed "mosaic RNA" at the time was the result of the excision of what came to be called intragenic sequences (introns) from precursor mRNA. This process of removing or "splicing out" introns is now known as precursor mRNA splicing (*pre-mRNA splicing* or *splicing* in short form). However, the concept of pre-mRNA is nowadays thought to be a virtual entity due to the co-transcriptional nature of splicing (Merkhofer et al., 2014).

Formally, an *intron* is defined as a gene segment that is present in the primary (or precursor) transcript but absent from the mature RNA as a consequence of splicing. The term intron refers to both the DNA sequence within a gene and the corresponding sequence in the unprocessed RNA transcript. On the contrary, an *exon* denotes a gene segment that is or can be present in mature RNA. Most human genes contain multiple exons, and the average length of exons ($50 - 250$bp[1]) is much shorter than that of introns (frequently thousands of bp). Figure 2.1 illustrates the split nature of eukaryotic genes: figure 2.1(a) shows the exons and introns of a gene as well as the untranslated regions (UTRs), the initiation codon and the termination codon at the 5' and 3' ends of the first and last exons. It also depicts a promoter region

[1] bp denotes base pairs



(a) Double stranded DNA gene.

(b) Single stranded pre-mRNA.

(c) Mature RNA.

FIGURE 2.1: Typical structure of a multi-exon eukaryotic gene (a) and its associated pre-mRNA resulting from transcription, 5' capping and polyA addition (b) and mature mRNA resulting from splicing (c).

that contributes to define the transcription inition site and a polyadenylation (polyA) addition sequence signal that contributes to define the polyA addition site. The polyA addition site delineates the transcription termination site. Figure 2.1(b) shows the pre-mRNA that results from transcription, 5' capping (*i.e.* the addition of a methylated guanine at the 5' end of the pre-mRNA) and polyA addition. Finally figure 2.1(c) corresponds to the mature mRNA resulting from pre-mRNA splicing.

How splicing happens?

The biochemical mechanism by which splicing occurs is fairly well understood (Clancy, 2008). Introns are removed from primary transcripts by cleavage at conserved sequences called splice sites. These sites are found at the 5' end (donor site) and 3' end (acceptor site) of introns. The splice donor site includes an almost invariant sequence GU within a larger and less highly conserved region while the splice acceptor site terminates the intron with an almost invariant AG sequence. These consensus sequences are known to be critical, as changing one of the conserved nucleotides often results in the inhibition of splicing (Cartegni et al., 2002). Another important sequence occurs at what is called the branch point, characterized by an A residue, and located anywhere from 18 to 40 nucleotides upstream from the 3' end of an intron.

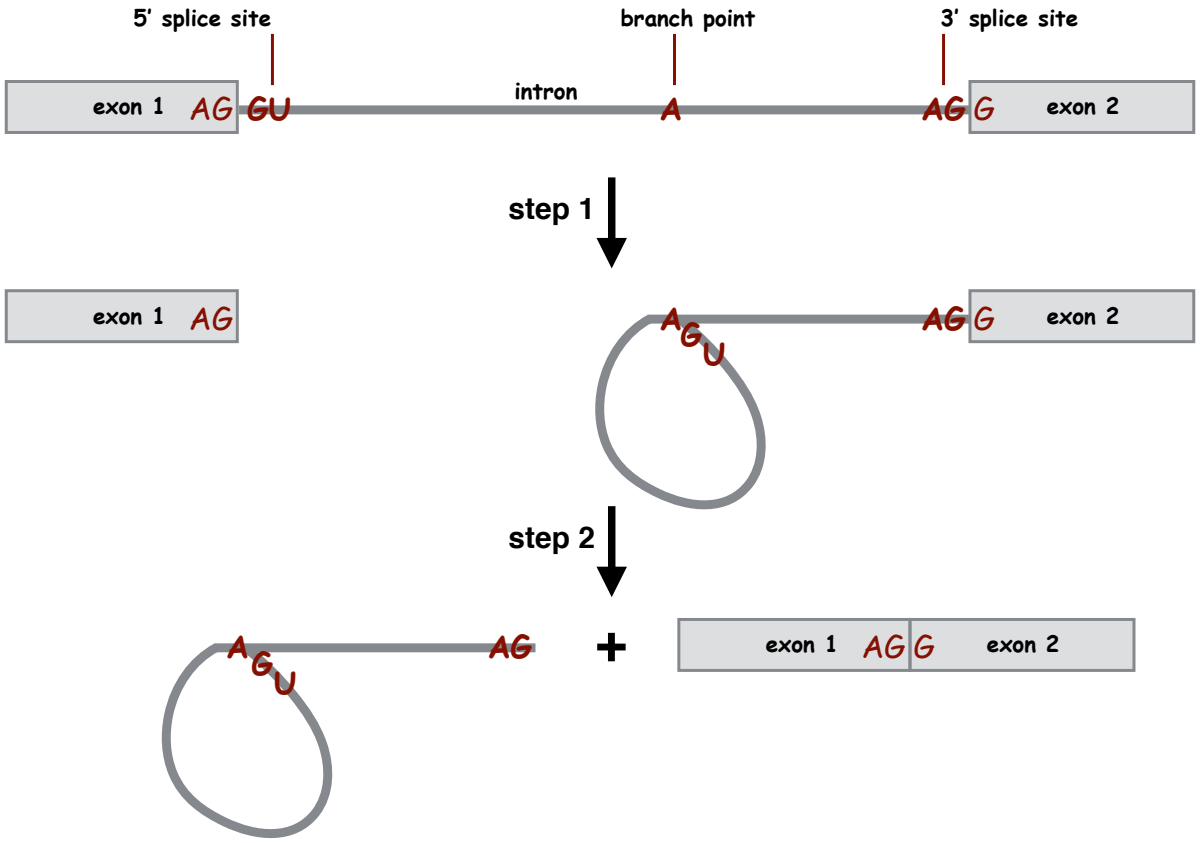


FIGURE 2.2: The two steps of the pre-mRNA splicing reaction.

Figure 2.2 schematically illustrates the two steps of the splicing chemical reaction: the A residue from the branch point interacts with the 5' splice site to form a so-called intronic lariat before ligation of the two exons and liberation of the intron. Splicing is carried out in the nucleus of eukaryote cells by the spliceosome, a megaparticle in which ribonucleoprotein particles (the so-called small nuclear ribonucleoprotein particles or snRNPs) and a large number of auxiliary proteins (denoted as splicing factors) cooperate to accurately recognize the splice sites and catalyse the two steps of the splicing reaction. A multitude of RNA-RNA, RNA-protein and protein-protein interactions allows for the precise excision of each intron and appropriate joining of the exons.

We refer to Hastings and Krainer (2001) and Black (2003) for more details about the splicing biochemistry.

2.1.2 Alternative splicing and alternative transcription

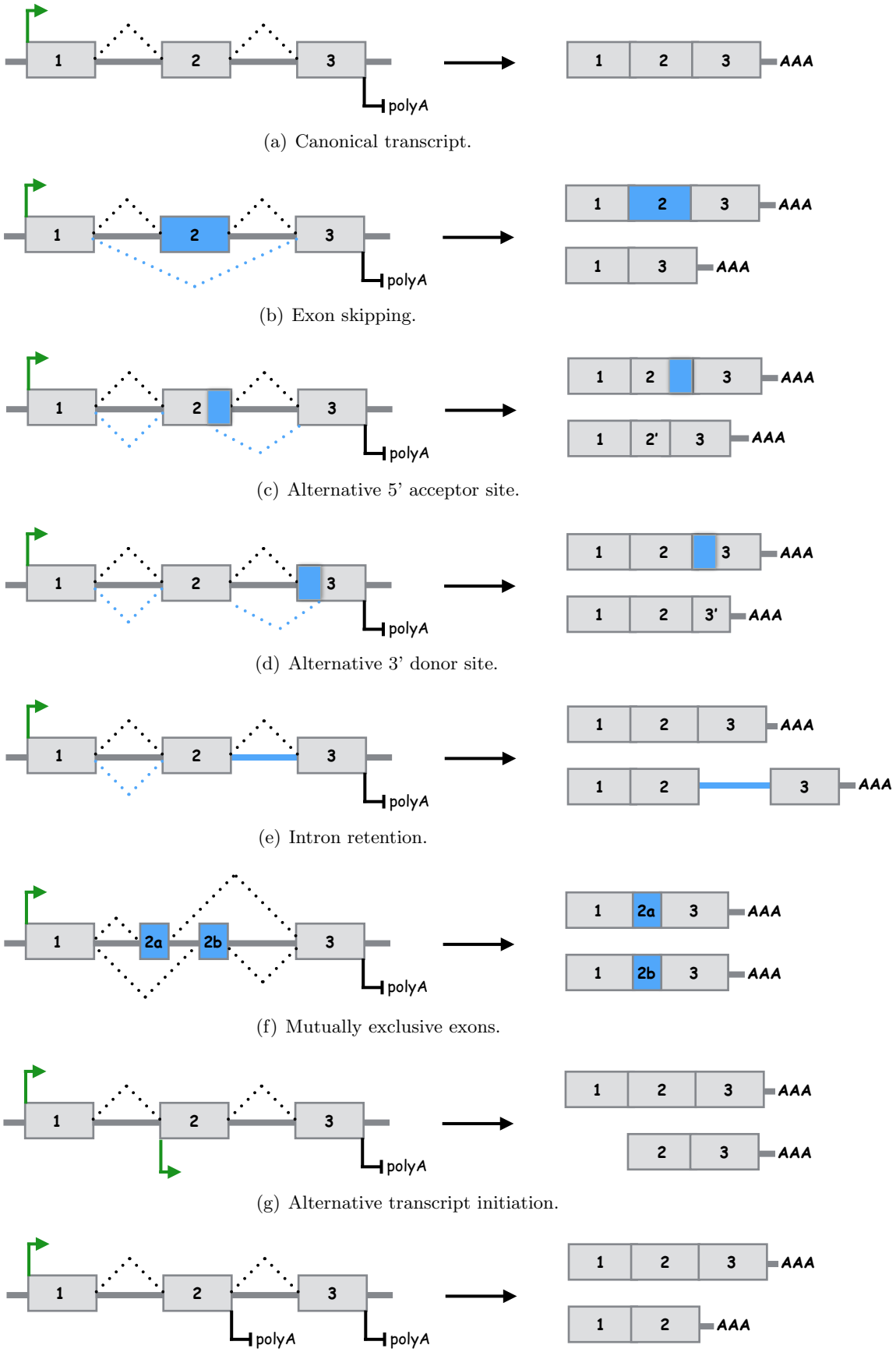
How come there are $\sim$ 120000 mRNA molecules mapped out in the human cells while the human genome contains only $\sim$ 25000 protein-coding genes? The solution lies in the alternative nature of splicing in eukaryotes.

Alternative splicing is the mechanism through which multiple mature mRNA transcripts (or mRNA *isoforms*) are expressed from a single gene. The ability of cells to exhibit variations of mature mRNA from the same pre-mRNA adds a layer of complexity to the central dogma DNA $\rightarrow$ RNA $\rightarrow$ protein of molecular biology. It is accomplished by excluding one or more exons (exon skipping), by moving exon/intron boundaries (acceptor or donor splice site shift) or by retention of introns. The main modes of alternative splicing are illustrated in figures 2.3(b), 2.3(c), 2.3(d), 2.3(e), 2.3(f). This widespread mechanism is estimated to affect $\sim$ 90% of mammalian protein-coding genes (Wang et al., 2008a) and is now considered a fundamental regulatory process at the crossroad between transcription and translation. Some functional aspects of alternative splicing are discussed in section 2.2.

Perhaps the most striking example of alternative splicing comes from *Drosophila melanogaster*. Its *Dscam* gene, which codes for a cell surface protein involved in neuronal connectivity, has 24 exons, with 12 alternative versions of exon 4, 48 versions of exon 6, 33 versions of exon 9 and 2 versions of exon 17. Each version of a particular exon is used to the exclusion of all the others. Thus the combinatorial use of alternative exons can potentially generate 38016 different protein isoforms (Schmucker et al., 2000). The *Dscam* gene exemplifies both the extreme expansion in coding capacity that alternative splicing provides and the tight regulation of alternative splicing that must be in place to somehow enforce mutual exclusion of the different versions of the exons.

In addition to the alternative splicing mechanisms mentioned above and illustrated in figure 2.3 (exon skipping, alternative acceptor or donor splice sites and intron retention), the exon composition of RNA transcripts can also vary by the differential selection of 5' end transcription initiation and 3' end termination sites – also known as multiple promoter or multiple polyA usage (Kornblihtt, 2005). Figures 2.3(g) and 2.3(h) illlustrate as well these two distinct mechanisms which are not splicing events *stricto sensu* but similarly participate to creating a variety of RNA transcripts from a single locus.

Identifying the different transcript isoforms produced by a single gene, that is the different combinations of exons included in the expressed mRNA, is the main scope of chapters 4 and 5.



(a) Canonical transcript.

(b) Exon skipping.

(c) Alternative 5' acceptor site.

(d) Alternative 3' donor site.

(e) Intron retention.

(f) Mutually exclusive exons.

(g) Alternative transcript initiation.

(h) Alternative polyadenylation.

FIGURE 2.3: Main modes of alternative splicing ((b) to (f)), alternative transcription initiation site (g) and alternative polyadenylation site (h).

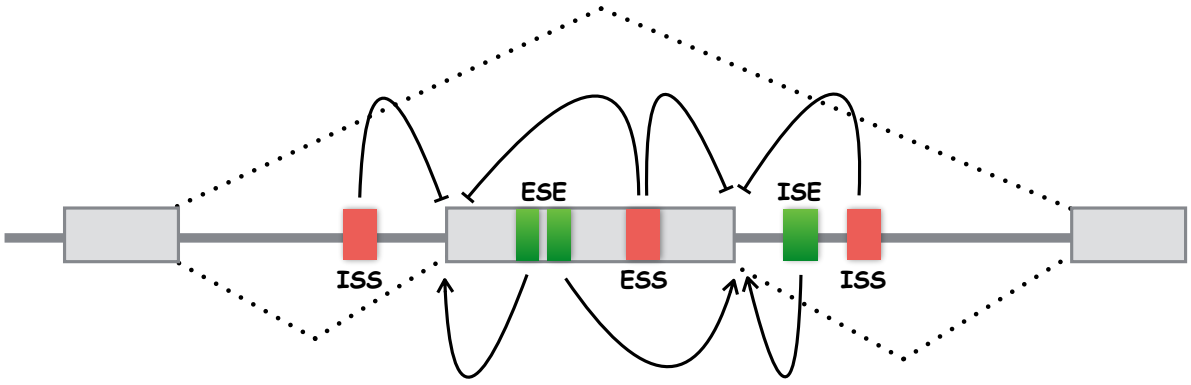


FIGURE 2.4: *Cis*-acting sequences regulating alternative splicing. ESE: exonic splicing enhancer, ISE: intronic splicing enhancer, ESS: exonic splicing silencer, ISS: intronic splicing enhancer. Enhancers can activate adjacent splice sites whereas silencers can repress splice sites. The competing influences of the different enhancers and silencers determine the inclusion or skipping of the exon. Figure is inspired from Matlin et al. (2005).

2.1.3 What makes splicing alternative?

The decision as to which exon is removed and which exon is included involves RNA sequence elements and protein regulators.

First of all, splice sites can be strong or weak depending on how far their sequences diverge from the consensus sequences, which determine their affinity for splicing factors. The relative position and use of weak and strong sites give rise to the different alternative splicing modes described in figure 2.3. Unsurprisingly, it has been shown that alternative exons possess weaker splice sites than constitutive exons (Sorek et al., 2004).

Second, the degree to which weak sites are used is regulated by both *cis*-regulatory sequences and *trans*-acting factors. Depending on the position and function of the *cis*-regulatory elements, they are divided into four categories: exonic splicing enhancers (ESEs), exonic splicing silencers (ESSs), intronic splicing enhancers (ISEs) and intronic splicing silencers (ISSs). *Trans*-acting factors include proteins and ribonucleoproteins that bind to the splicing enhancers and silencers. Figure 2.4 shows how these enhancers and silencers act combinatorially to regulate the alternative use of splice sites. Of note, a machine learning algorithm has been developed that is capable of automatically extracting combinations of *cis*-elements that are accurately predictive of brain, muscle, digestive and embryo versus adult specific alternative splicing patterns (Barash et al., 2010).

Finally, alternative splicing is also believed to be regulated by the secondary structure of the pre-mRNA transcript and by interactions with the transcription and chromatin machineries (Schwartz and Ast, 2010; Luco et al., 2011).

For accurate reviews of alternative splicing mechanisms and regulation we suggest Matlin et al. (2005), Chen and Manley (2009) and Kornblihtt et al. (2013).

In line with what has been presented above, chapter 6 focuses on detecting splicing defects on transcripts expressed from alleles harboring mutations in their *cis*-regulatory splicing enhancers or silencers.

2.2 Some aspects of the functional importance of alternative transcript expression

2.2.1 A word of evolution

Alternative splicing is believed to occur in all metazoan organisms, but is more prevalent in vertebrates. The number of protein-coding genes in vertebrates is not radically different from the number in invertebrates (for example the number of human genes is roughly equal the the number of nematode genes and barely four times the number of genes in budding yeast), suggesting a link between alternative splicing prevalence and phenotypic complexity (Nilsen and Graveley, 2010). Kim et al. (2007) studied in depth the different levels of splicing among eukaryotes and proposed alternative splicing as a possible solution to the paradoxical miscorrelation between the number of genes in an organism's genome and its phenotypic complexity.

The split organization of eukaryotic genes into exons and introns and the existence of pre-mRNA splicing process is believed to confer at least two evolutionary advantages. The first –relatively obvious– advantage is that alternative splicing allows a single gene to produce several mRNA variants, greatly expanding the coding capacity of eukaryotic genomes (Keren et al., 2010). The second advantage lies at a phylogenic level, as intronic recombination events (such events leave the exons intact) allow protein-coding exons to be placed together to form new genes. Recombined mRNAs have high chance of encoding novel functional polypeptides that combine functional domaines previously tested by natural selection. This mutational process is known as *exon shuffling* (Ast, 2004). Moreover it has been proposed that alternative splicing represents a major source of species-specific differences: for exemple Barbosa-Morais et al. (2012) recently

showed that there is a decline in alternative splicing frequency in vertebrates as the evolutionary distance from primates increases.

However, the prevalence of alternative splicing raises questions about its biological significance. What fraction of multiple mRNA isoforms expressed from each of $\sim$ 20000 alternatively spliced human genes has a functional impact? It has been proposed that many alternative splicing events do not have functional significance but rather represent stochastic noise in the splicing process (Melamud and Moult, 2009; Skandalis et al., 2010). In any case, the adaptive role of alternative splicing remains elusive, in part because few variant transcripts have been characterized functionally, making it difficult to assess the contribution of alternative splicing to the generation of phenotypic complexity and to study the evolution of splicing patterns (Mudge et al., 2011).

2.2.2 Alternative splicing regulation during development and cell fate decision

The *Drosophila* sex determination pathway provides a simple and central example of how a choice between different splicing patterns contributes to cell fate decision and tissue specificities. Indeed, sex determination in flies is a binary decision based on alternative splicing (Salz, 2011): splicing of the sex-lethal (*Sxl*) gene in females gives rise to a functional protein product, while in male alternative splicing leads to the inclusion of a stop codon so that the functional protein in not produced. Remarkably the *Sxl* gene is a splicing factor that regulates as well the splicing of its target genes also involved in the sex determination pathway. Interestingly, related insects such as the housefly do not splice the *Sxl* pre-mRNA in a sex-specific manner while the sex-determination cascade of the honeybee is different in almost all its components although relying on alternative splicing as well. This shows as previously discussed the evolutionary plasticity provided by alternative splicing (Nilsen and Graveley, 2010).

In addition to alterations by sex, metazoan organisms regulate the splicing of thousands of other transcripts depending on cell type, developmental state or external stimulus. High-throughput studies have shown that 50% or more of alternative splicing isoforms are differently expressed among tissues, indicating that most alternative splicing is subject to tissue-specific regulation (Yeo et al., 2004; Wang et al., 2008a).

Large-scale profiling studies have also revealed sets of alternative splicing events associated with changes in cell differentiation and development (Blencowe, 2006). In particular, alternative splicing has been identified to contribute to the differentiation of embryonic stem cells (ESCs) into distinct lineages. Wu et al. (2010) provided evidence that isoform complexity is more extensive in ESCs and becomes restricted and more specialized as ESCs differentiate, while Gabut et al. (2011) showed that an ESC-specific alternative splicing switch stimulates the expression of key pluripotency genes.

For a detailed review of the functional consequences of developmentally regulated alternative splicing we refer to Kalsotra and Cooper (2011).

2.2.3 Coupling of alternative splicing with nonsense-mediated decay

Most human genes exhibit alternative splicing, but not all alternatively spliced transcripts produce functional proteins. Some alternative splicing events in humans result in mRNA isoforms harboring a premature termination codon (PTC), *i.e.* a stop codon located upstream from the last exon. A single-nucleotide mistake during the pre-mRNA splicing process often results in a frameshift and consequent PTC appearance. These transcripts characterized by a PTC are predicted to be degraded by the nonsense-mediated mRNA decay (NMD) pathway (Lareau et al., 2007). Figure 2.5 illustrates the NMD degradation process.

NMD is then considered as an mRNA quality-control mechanism by degrading transcripts encoding truncated proteins with no or undesired functions. However, while it prevents the accumulation of potentially harmful polypeptides, NMD is also believed to regulate the expression of $10 - 20\%$ of normal transcripts. Briefly, it is the coupling of alternative splicing and NMD that allows the downregulation of specific transcripts: alternative splicing events that occur in exons located in the 3' UTR and that generate a PTC activate NMD even though the degraded transcript would have encoded a full-length protein. This regulation phenomenon is believed to restrict the expression of several stress-related mRNA under non-stress conditions (Lykke-Andersen and Jensen, 2015).

We will encounter the NMD pathway again in chapter 6 when we pay attention to its inhibition in a clinical diagnosis setting in order to reveal the expression of aberrant transcripts from mutated *BRCA1* alleles.

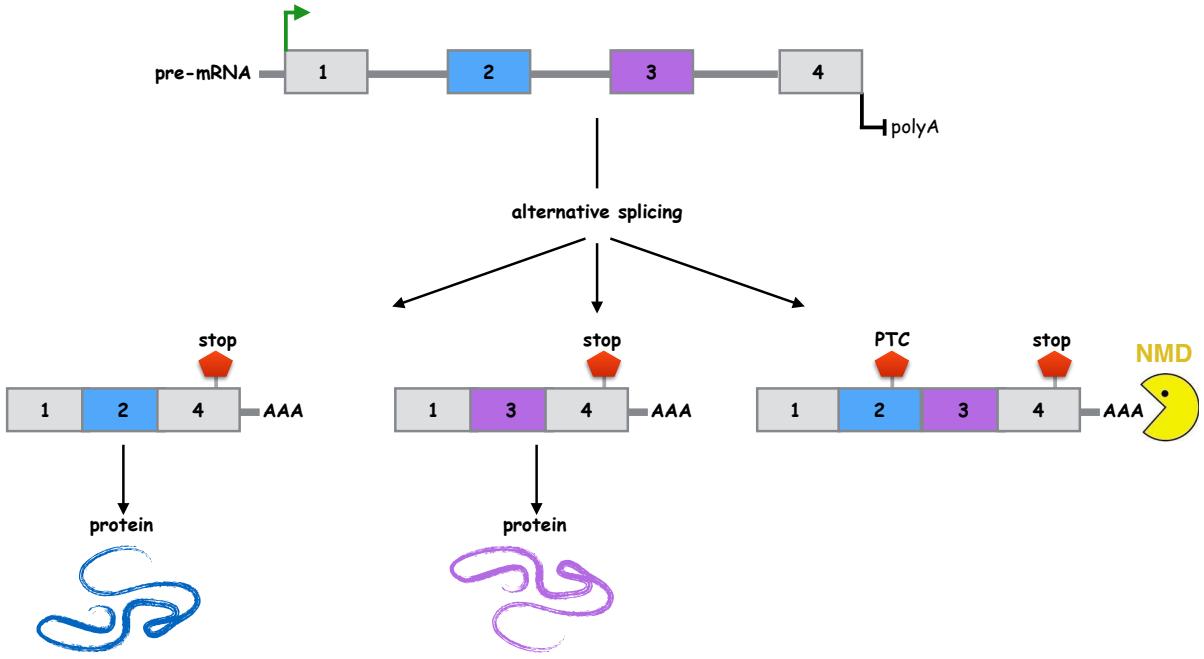


FIGURE 2.5: Coupling of alternative splicing and nonsense-mediated decay. PTC: premature stop codon, NMD: nonsense-mediated decay. In the depicted example, exons 2 and 3 are mutually exclusive, so that the simultaneous inclusion of both exons generate a PTC that activate NMD. Figure is inspired from Lareau et al. (2007).

2.3 Splicing dysregulation in human diseases

The link between alternative splicing and disease is well established (Scotti and Swanson, 2016a), and many different human diseases can be caused by errors in RNA splicing or its regulation. We briefly discuss here the link between splicing and human diseases via the alteration of both *cis*- or *trans*-acting factors, with a particular focus on cancer. In addition, we emphasize that the identification of abnormal splicing as a primary mechanism of diseases raises the possibility of therapeutic approaches targeting splicing.

2.3.1 Mutated regulatory sequences

Mutations in regulatory sequences that affect alternative splicing are a widespread cause of human hereditary diseases and cancers. These mutations can disrupt existing splicing enhancers or silencers or create new ones, thereby perturbing the use of alternative or constitutive exons. A single nucleotide mutation that does not change the encoded amino acid of a protein (silent mutation) can disrupt for instance a crucial splicing enhancer and be a disease-causing mutation (Wang and Cooper, 2007). Examples of human disease genes known to be targeted by synonymous and non-synonymous mutations often altering splicing regulatory elements include

the *BRCA1* (breast cancer 1) gene involved in heriditary breast cancer, the *SMN1* (survival of motor neuron 1) gene involved in spinal muscular atrophy and the *DMD* gene involved in Duchenne muscular dystrophy or the *MAPT* (microtubule-associated protein tau) gene involved in Alzheimer's disease (Cartegni et al., 2002).

It has been estimated that as many as 50% of disease mutations in exons may impact on splicing (Lopez-Bigas et al., 2005). This strongly suggests that, in a clinical diagnosis perspective, genetic variants that are linked with a disease phenotype need to be evaluated for disruption of the correct splicing patterns. For example, it is important to know that a mutation results in a loss of expression due to aberrant splicing and NMD-mediated degradation, rather than the expression of a wild-type level of a protein containing a missense mutation. Knowing that the primary effect of an exonic mutation is a splicing defect, rather than a protein-coding mutation, is crucial in order to understand the detailed pathogenic mechanism of a disease.

In chapter 6, we underline the importance of introducing routine transcript analysis in order to properly assess possible mechanisms accounting for human diseases, and propose a new methodology to implement such routine mRNA screenings.

2.3.2 *Trans*-acting factors

Mutations in genes encoding *trans*-acting factors that regulate alternative splicing can also cause diseases. Unlike the *cis*-acting mutations that only affect the compromised gene, this second type of mutation can affect large sets of genes. Mutations in different constituents of the spliceosome are involved in several diseases, such as retinal degenerative disorders and cancers. As an example, the familial form of retinitis pigmentosa –the most common form of blindness– is characterized by mutations in genes required for the proper assembly and function of a core component of the spliceosome (Wang and Cooper, 2007).

As we discuss in the next section, cancers are associated with splicing changes, such as switches of the expression level of the predominant transcript isoforms of developmental genes. Most of these cancer-associated splicing changes are not associated with nucleotide changes in the affected genes, implying an alteration of *trans*-acting factors (Srebrow and Kornblihtt, 2006). To illustrate this, recent large scale studies have uncovered recurrent somatic mutations in splicing factor genes linked to poor prognosis in myelodysplastic syndromes and chronic lymphocytic leukemia (Papaemmanuil et al., 2011; Malcovati et al., 2014; Yoshida and Ogawa, 2014).

2.3.3 A focus on cancer

Cancer is a heterogeneous and complex disease, and the role of alternative transcription (Davuluri et al., 2008) and alternative splicing (David and Manley, 2010) has been known to be implicated in cancer for long. As previously described, a combination of factors influences alternative splicing events in a cell-type and developmental-specific manner. The transcript isoforms produced by the cells are tightly regulated during normal development, but often dysregulated in tumors. In short, cancer cells use the flexibility brought by alternative splicing to express specific isoforms that confer survival advantages and drug resistance (Pal et al., 2012).

A striking phenomenon illustrating how alternative splicing is intrinsically linked to tumor's development is the existence of cancer-specific transcript isoforms. In particular, specific isoforms known to be involved in epithelial-mesenchymal transition (EMT) during embryonic development are reactivated in cancer cells, leading to enhance invasion and metastasis and associated with poor prognosis (Shapiro et al., 2011; Biamonti et al., 2012). These development-specific isoforms are important candidates in understanding the pathogenesis and progression of cancer (Pal et al., 2012).

Another common phenomenon in tumors related to the regulation of alternative splicing is the switch of the predominant transcript isoforms expressed in cancer cells compared to normal cells, while the protein isoforms produced often have opposite functions. As an example, transcripts from a large number of genes involved in apoptosis are alternatively spliced, resulting in isoforms with opposite roles in promoting or preventing cell death (Schwerk and Schulze-Osthoff, 2005). David and Manley (2010) provides a series of examples of such genes implicated in apoptosis that produce two isoforms with antagonist functions such that the pro-apoptotic form is overexpressed in several cancers.

2.4 Emerging therapies targeting splicing defects

2.4.1 Cancer-specific isoforms as biomarkers

Splicing abnormalities are commonly reported in various cancers (Wang and Cooper, 2007). Therefore, alternative spliced variants are potential biomarkers for the cancer diagnosis or prognosis and may be good targets for cancer therapies based on specific splicing correction treatments (Pal et al., 2012).

Zhang et al. (2013) recently reported that cancer cells could be more accurately discriminated from non-oncogenic cells using transcript isoform expression rather than solely gene expression, highlighting the importance of providing cancer signatures at the isoform level. By comparing matched tumor and normal tissues of hundreds of samples across several cancer types, other recent studies (Dvinge and Bradley, 2015; Danan-Gotthold et al., 2015; Tsai et al., 2015; Sebestyen et al., 2015) reported recurrent splicing alterations both across cancers and specific to cancer types. Splicing markers include cassette exons or intron retentions as well as switches in the predominant transcript isoforms.

2.4.2 Splice modulating therapies

Splice modulating therapies (Douglas and Wood, 2011; Scotti and Swanson, 2016b) are emerging as an opportunity to correct splicing defects and potentially treat numerous genetic disorders, including cancer. These emerging therapies are of two main types: some modulating the spliceosome's activity, others targeting specific transcript isoforms or aberrant regulatory sequences of the pre-mRNA.

The first category corresponds to small molecules (bacterial fermentation products) that show antitumoral activity by modulating the functions of the spliceosome (Bonnal et al., 2012). The second category corresponds to nucleic acid-based tools that target mRNA or pre-mRNA to correct or attenuate splicing defects (Spitali and Aartsma-Rus, 2012). Among these tools, RNA interference (RNAi) can target disease-specific transcript isoforms and inhibit their expression, while antisense oligonucleotides (AONs) can interact with splicing regulatory elements to specifically manipulate pre-mRNA splicing. AONs are short oligonucleotides synthesized to be complementary to a particular RNA sequence. By designing AONs that hybridize with specific splice sites or with enhancer or silencer elements, the splicing mechanism of the targeted pre-mRNA can be drastically manipulated. Figure 2.6 sketches the AON mode of action. AONs show particular promise in the therapeutic area as illustrated in the next section.

2.4.3 Antisense oligonucleotides: the example of Duchenne muscular dystrophy

As described above and in figure 2.6, splicing can be modulated with antisense oligonucleotides, offering prospects of personalized medicine tailored to specific mutations.

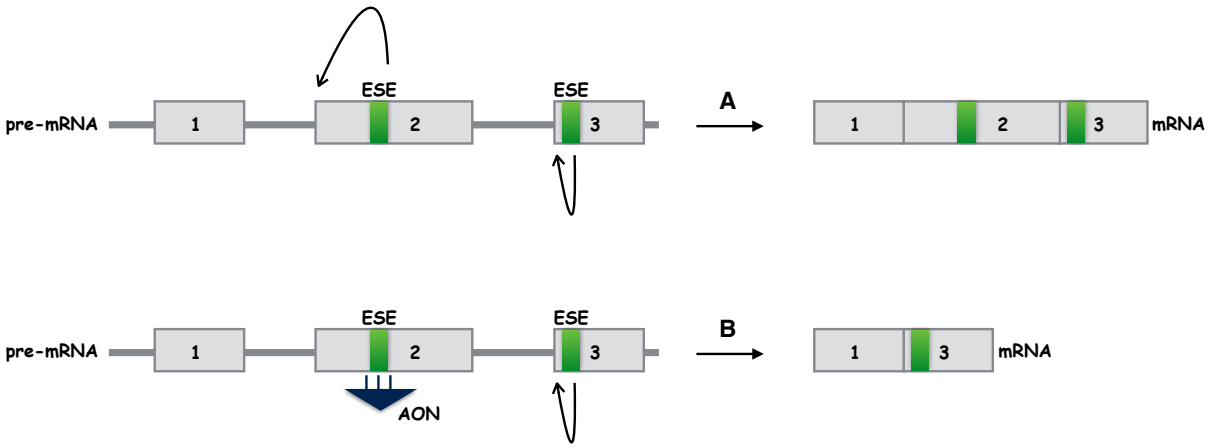


FIGURE 2.6: Use of antisense oligonucleotides to modulate pre-mRNA splicing. ESE: exonic splicing enhancer, AON: antisense oligonucleotide. In that specific example, an ESE located within the second exon activates the use of the exon's splice site (A). When the ESE interacts with AONs such that it becomes inaccessible to the splicing machinery, splicing is shifted toward exon 3 so that exon 2 is skipped (B).

A successful AON strategy has been developed for treating Duchenne muscular dystrophy (DMD). DMD is a progressive muscular disease that roughly affects 1 over 3500 newborn males. DMD mutations are often multi-exon deletions that cause frameshift at exon 51. The reading frame can however be restored by skipping of exon 51, leading to the production of internally deleted DMD proteins that retain partial function. This can be achieve in vivo by the binding of AONs to an exon 51 splicing enhancer that shift splicing to exon 52 (Scotti and Swanson, 2016b). Notably, AON strategies are currently under evaluation in DMD patients in clinical trials.

Chapter 3

Questioning splicing: from data to algorithms

"knowledge of sequences could contribute much to our understanding of living matter"

Frederick Sanger.

"the way we do RNA-seq now ... is you take the transcriptome, you blow it up into pieces and then you try to figure out how they all go back together again. If you think about it, its kind of a crazy way to do things"

Michael Snyder.

Ce chapitre recense les techniques experimentales existantes pour détecter les événements d'épissage et les transcrits alternatifs. Les méthodes de séquençage à haut-débit de l'ARN sont finement détaillées ainsi que les défits posés par l'analyse algorithmique des données. Les notions de pénalisation par la norme ℓ_1 et d'optimisation de flots, deux concepts clés dans le domaine de l'assemblage du transcriptome, sont introduites.

In this chapter, we review some sequencing or profiling techniques that can be used to detect and quantify alternative splicing events and transcript isoforms. We focus on describing high-throughput RNA sequencing technologies as well as the computational challenges associated with the data and the variety of methods that exist to assemble and quantify transcripts. We end the chapter by introducing the notions of ℓ_1-norm penalization and network flow optimization as two key concepts used in the field of transcriptome assembly.

3.1 Measuring splicing with data evolving in time

3.1.1 Heritage of Sanger sequencing

Sanger sequencing

Nucleic acid sequencing denotes a method for determining the exact order of nucleotides present in a given DNA or RNA molecule. A major foray into DNA sequencing was the Human Genome Project (ConsortiumInternational, 2004). It was completed in 2003 after a $3 billion and 13-year-long endeavor using techniques that relied on Sanger sequencing.

The Sanger sequencing technology, named after its inventor Frederick Sanger, was developed in 1977 (Sanger et al., 1977). It can be defined as a "chain-termination" enzymatic sequencing method. It uses the combination of a polymerase enzyme and fluorescently labeled terminator nucleotides to decipher a DNA nucleotidic sequence. More precisely, single stranded DNA is replicated by a polymerase in the presence of chemically altered versions of the A, C, G, and T bases among regular nucleotides. The altered bases stop the replication process when they are incorporated into the growing strand of DNA, resulting in varying lengths of short DNA. In addition, in the optimized version[1] of Sanger sequencing (Smith et al., 1986), each of the four altered base is incorporated with a different fluorescent dye. The DNA strands are then ordered by size (using capillary electrophoresis), and by reading the end letters (using laser excitation and spectral emission analysis) from the shortest to the longest piece, the whole sequence of the original DNA is revealed. Figure 3.1 illustrates the Sanger sequencing technique.

The key strength of Sanger sequencing is that it remains the most available technology nowadays and that it is very accurate in reading the nucleotidic bases. However, the requirement for electrophoretic separation of DNA fragments limits the number of samples that can be run in parallel and is the primary bottleneck for throughput.

Expressed sequence tag

The combination of reverse transcription of RNA to complementary DNA (cDNA) and Sanger sequencing was the first mean to generate abundant information on the transcriptome. This procedure, fully developed in the 90's initially as part of the human genome project (Adams

[1] in its first version the Sanger protocol divides a DNA sample into four separate sequencing reactions each one containing only one of the terminator nucleotide A,C,G or T.

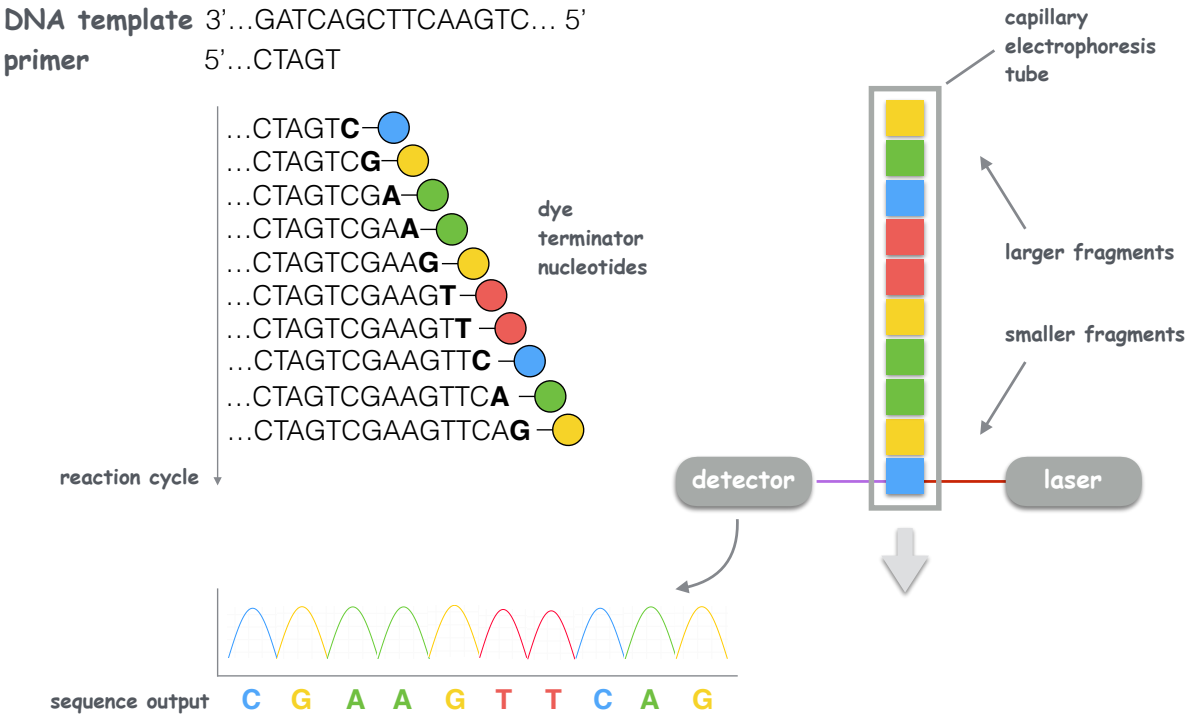


FIGURE 3.1: Illustration of the Sanger sequencing technique. Figure is inspired from https://www.abmgood.com/marketing/knowledge_base/next_generation_sequencing_introduction.php.

et al., 1991), produces the so-called expressed sequence tags (ESTs). Formally an EST is a short sub-sequence of a cDNA sequence. A RNA population is reverse transcribed to double-stranded cDNA using a specialized enzyme, the reverse transcriptase. The resultant cDNA is cloned[2] to make libraries representing a snapshot of the transcriptome of the original tissue. The cDNA clones are sequenced randomly in a single-pass run from either their 5' or 3' end, producing 100 to 800bp long ESTs. More than 70 million ESTs are available in public databases, such as GenBank (Benson et al., 2005).

Alignment of EST data to sequenced genomes afforded initial glimpses into the extend of alternative splicing and other forms of transcript processing complexity (Nagaraj et al., 2007). Analysis of 3' end EST data for instance gave significant insights into the use of polyA sites in human tissues. Gautheret et al. (1998) identified previously unreported polyA sites in human mRNAs and Yan and Marr (2005) demonstrated that at least 49% of human polyadenylated transcription units show alternative polyA sites. In addition to the study of polyA sites with EST data, Modrek et al. (2001) performed a genome-wide appreciation of alternative splicing. They

[2]molecular cloning corresponds to the process of amplification of DNA molecules via its replication in bacteria. Note that modern sequencing technologies rather use *in vitro* amplification with the polymerase chain reaction (PCR).

estimated that $\sim$ 40% of human protein coding genes are alternatively spliced. As explained in section 3.1.3, this number has increased significantly with the emergence of high-throughput RNA sequencing techniques, reaching an estimate of $\sim$ 90% (Pan et al., 2008).

While EST libraries have first provided genome-wide evidence of alternative splicing and alternative transcription sites, allowing the design of specific probes for microarray profiling (see section 3.1.2), it remains relatively low-throughput and generally not quantitative. Moreover, since ESTs are generated from the 5' and 3' ends of cDNA clones, detection of mRNA processing events is biased towards the ends of the transcripts. In comparison, section 3.1.3 describes how high-throughput RNA sequencing allows for the efficient detection and quantification of a diverse range of RNA processing events.

3.1.2 Successes and limitations of microarray splicing profiling

Microarray technologies have played a predominant role in shaping our understanding of transcriptome complexity and regulation (Blencowe, 2006). Microarray approaches rely on the hybridization of fluorescently labeled target RNA sequences to anchored oligonucleotides of known composition, often called *probes*, previously attached to a glass-slide. The abundance of target RNA is then inferred using laser fluorescence that measures the extent of hybridization on the probes. The development of custom microarrays with probe sets designed to detect individual exons and splice junction sequences overcame many of the obstacles encountered when analyzing EST data, in particular throughput and quantification aspects (Pan et al., 2004). Splicing microarrays can indeed be designed to hybridize to isoform-specific mRNA regions, which allows for the detection and quantification of distinct spliced isoforms. The concept of splicing microarrays is illustrated in figure 3.2.

Splicing microarray successes include the discovery of new alternative splicing events and the detection of cell- and tissue-specific alternative splicing events. For example, Johnson et al. (2003) used arrays with probes for all adjacent exon-exon junctions in 10000 human genes and hybridized these with samples from 52 human tissues, revealing tissue-specific clustering of alternative splicing events.

The major drawbacks of splicing microarray are two folds: the limited dynamic range of signal detection and the reliance upon an existing genomic sequence. Indeed, array measurements are

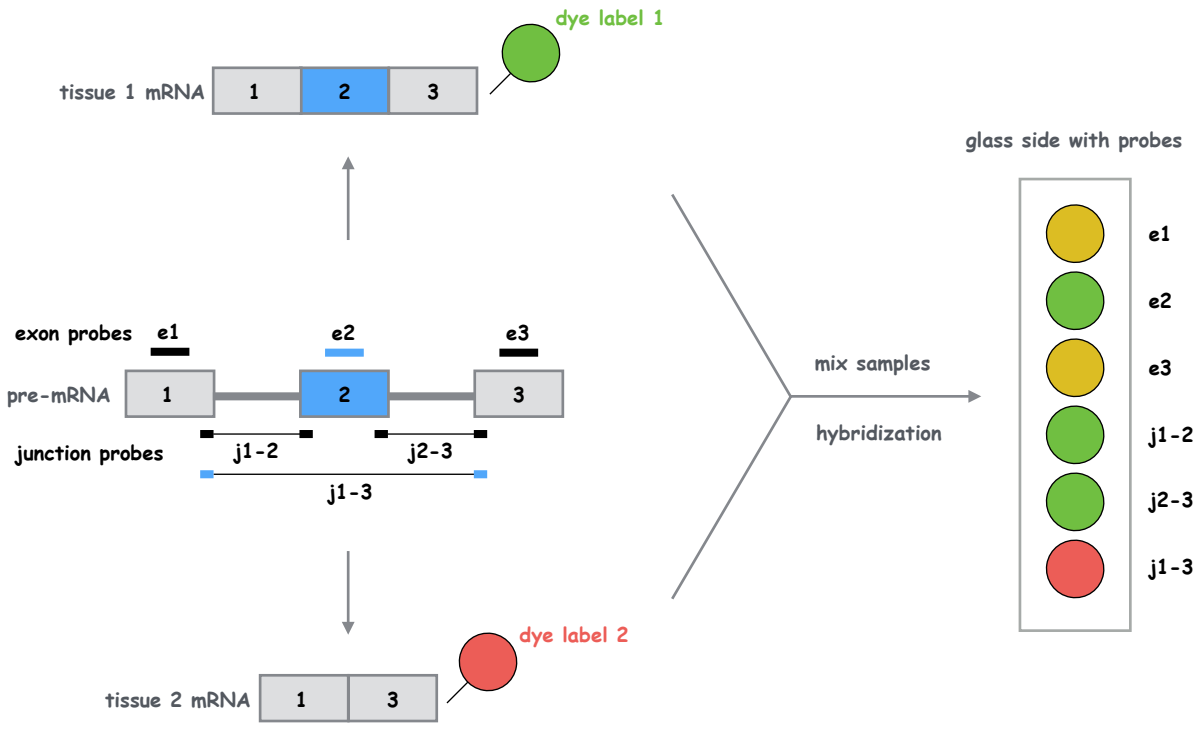


FIGURE 3.2: Illustration of a splicing microarray experiment. Probes are complementary to individual exons or exon-exon junctions. Dye-labeled mRNAs hybridize with the corresponding probes, which allows the comparison of expression levels between the two samples. Figure is inspired from Matlin et al. (2005).

limited by a strong background noise level and by saturation of high fluorescent signals. It also requires prior knowledge of target RNA sequences to design the probes.

The next section explains how modern high-throughput RNA sequencing does not require transcript-specific probes and measures a large dynamic range of expression levels.

3.1.3 High-throughput sequencing of the RNA as the new gold standard

Demand for cheaper and faster sequencing methods has increased greatly after the first human genome sequence was completed in 2003. This demand has driven the emergence of fast, cost-effective, accurate and high-throughput sequencing technologies. The so-called "next-generation" sequencing (NGS) technologies enable to sequence an entire human genome in less than one day by sequencing massive amount of DNA in parallel. High-throughput RNA sequencing or "RNA-seq" is an experimental protocol that uses NGS technologies to sequence RNA molecules within a biological sample.

In comparison to EST sequencing by Sanger technology, which is low-throughput and only detects the more abundant transcripts, RNA-seq can target lowly express transcripts and can

sequence millions of cDNA sequences in a single reaction. In contrast to other high-throughput technologies, such as hybridization-based microarrays, RNA-seq achieves base-pair level resolution, offers a much higher dynamic range of expression levels and does not require prior knowledge of the sequences to be profiled.

We explain below the principles of RNA-seq, that is the use of NGS technologies to sequence and latter quantify RNA molecules after their conversion to cDNA by reverse transcription. We refer to Goodwin et al. (2016) for a detailed description of the different existing NGS technologies and to Wang et al. (2009) for a focus on the RNA-seq protocol.

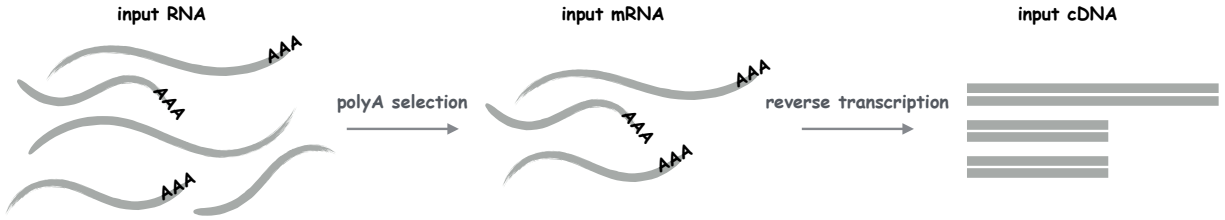
RNA-seq technology

Next-generation sequencing, also referred as "deep sequencing", "high-throughput sequencing", "massively-parallel sequencing" or "shotgun sequencing", has revolutionized genomics, epigenomics and transcriptomics by allowing massively parallel sequencing at a relatively low cost (Koboldt et al., 2013). The key strength of NGS technologies is to perform real-time identification of millions of nucleotidic sequences in parallel. This differs greatly from Sanger sequencing technology where complementary strands of target cDNA first have to be separated by size before being revealed.

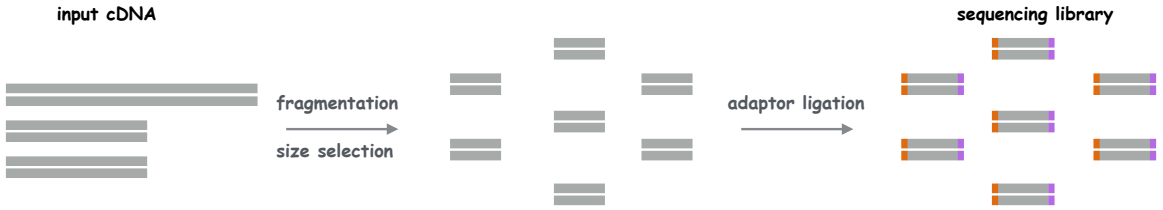
Various NGS platforms exist and use different chemistry or different ways to iteratively read the target nucleotides. Mardis (2011) and van Dijk et al. (2014) provide a comparison of the different NGS platforms. However, all technologies monitor the sequential addition of nucleotides to immobilized and spatially arrayed DNA templates. We choose here to focus on the strategy developed by the Illumina platform (Bentley et al., 2008), which is the most widely used NGS technology worldwide. Illumina sequencing uses a "sequencing by synthesis" approach (described below) combined with fluorescence image analysis. Note that other platforms use a "sequencing by ligation" technique (McKernan et al., 2009) or identify the growing nucleotidic strands with analysis of electric rather then fluorescent signals (Rothberg et al., 2011).

The different steps of RNA-seq, additionally illustrated in figure 3.3, are the following:

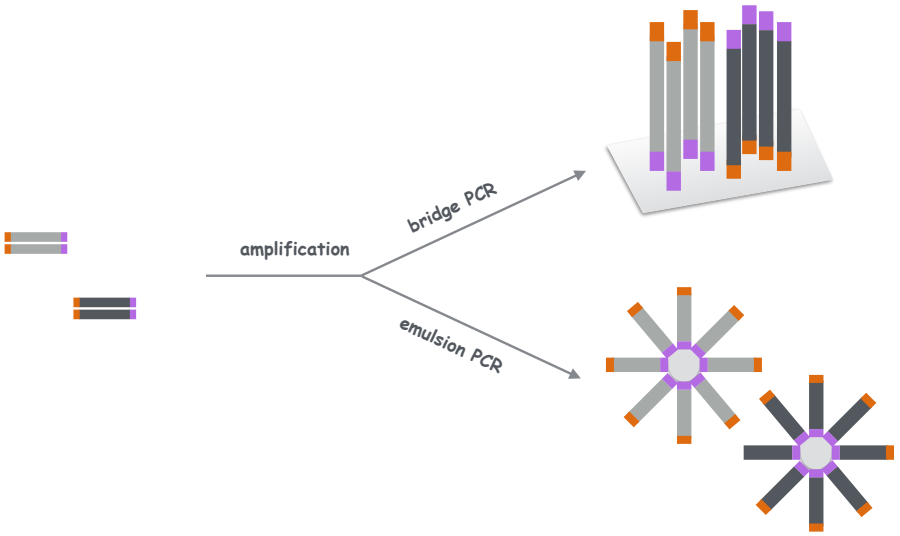
1. *Mature RNA selection and reverse transcription.* As ribosomal RNA (rRNA) constitutes the predominant fraction of the transcriptome, it needs to be removed to avoid wasting sequencing efforts on a few superabundant molecules. rRNA for which the sequence is



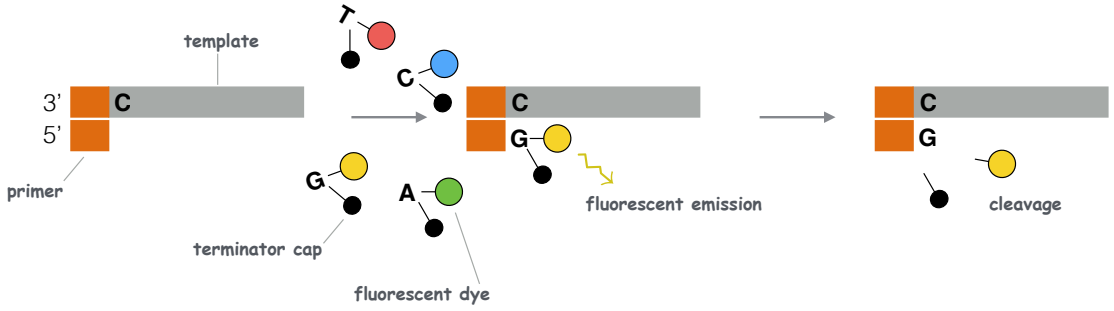
(a) Selection and reverse transcription



(b) Library preparation



(c) Amplification



(d) Sequencing by synthesis

FIGURE 3.3: A typical RNA-seq experiment. Figure is inspired from https://www.abmgood.com/marketing/knowledge_base/next_generation_sequencing_introduction.php.

known can be directly subtracted from the transcript pool, or alternatively mRNA harboring a polyA tail can be enriched by capture with oligo-dT[3]. Selected RNA molecules are converted to cDNA by a reverse transcriptase.

2. *Library preparation.* Starting material must be converted into a library of sequencing reaction templates which require fragmentation, size selection and adapter ligation.

   Given that most NGS technologies cannot sequence fragments longer than 1000 bases (often only a hundred bases), cDNA molecules need to be sheared into pieces so that all nucleotides of the molecules are sequenced. Fragmentation can be enzymatic or performed via hydrolysis or physical methods such as acoustic shearing or sonication. Note also that in some protocols fragmentation can be done at the RNA level before reverse transcription.

   Adapter ligation adds synthetic oligonucleotides of a known sequence onto the ends of the cDNA fragments, which serve as primers for downstream amplification and/or sequencing reactions. In strand-specific RNA-seq protocols (Levin et al., 2010), different primers are attached to the 5' and 3' ends of the RNA molecules, which further allows overlapping transcripts expressed from opposite strands of the genome to be distinguished.

3. *Template generation and amplification.* One of the key steps of NGS is to immobilize and separate the DNA fragments from a population (typically on a flow cell or on microbeads), allowing the downstream sequencing reaction to operate in parallel on millions of spatially distinct DNA templates.

   Additionally, a template amplification step is required for most sequencing platforms in order to obtain sufficient signal for base calling. Amplification strategies are based on a polymerase chain reaction (PCR) step (emulsion PCR onto microbeads or bridge amplification to form clusters on a flow cell).

   Note that amplification-free protocols are emerging as promising technologies. SMRT (single molecule real-time) platforms (Eid et al., 2009) are indeed based on single-molecule template sequencing hence bypassing the need for fragmentation and amplification. Amplification-based and single-molecule sequencing technologies have been respectively referred to as "second-generation" and "third-generation" sequencing.

4. *Sequencing and base calling.* The sequencing by synthesis strategy implemented by Illumina uses the cDNA library fragments as templates of which new DNA fragments are synthesized by a polymerase enzyme.

[3] oligo-dT are short sequences of deoxy-thymine nucleotides.

Similarly to Sanger sequencing it employs fluorescently-labeled terminator nucleotides. However, the key innovation compared to Sanger sequencing is the use of reversible terminators. Hence during each reaction cycle a single nucleotide is added to the growing DNA strand and the fluorescent dye is imaged to identify the base. The terminator is then enzymatically cleaved so that it allows incorporation of the next nucleotide.

Sequencing typically occurs solely at the ends of the cDNA fragments. The sequenced ends are called *reads*. Sequencing only one end of the fragments produces the so-called "single-end reads" whereas sequencing both the 5' and 3' ends produces "paired-end reads". An Illumina platform typically produces reads of $\sim$ 100bp.

The millions of reads produced by RNA-seq further need to be pre-processed and analyzed in order to answer relevant questions such as i) what are the levels of expression of the mRNA transcripts in a biological sample? ii) are some transcripts differentially expressed between different conditions or iii) are there any alternative splicing events specific to a given tissue?

In that context, section 3.2 focuses on the analysis of RNA-seq reads, in particular in the aim of identifying and quantifying the different transcript isoforms present in a given sample. Chapters 4 and 5 provide new computational methods to infer the transcript isoforms from RNA-seq data.

Opportunities raised by RNA-seq

A new appreciation of the complexity of the transcriptome has emerged with the use of RNA-seq data (Blencowe et al., 2009). The biological applications that RNA-seq makes it possible to target are very diverse (Ozsolak and Milos, 2011), ranging from the profiling of mRNA and non-coding RNA expression to the study of alternative splicing, alternative polyadenylation or transcription initiation sites as well as the study of small RNA, antisense transcripts or the detection of fusion genes.

In particular, datasets generated with the RNA-seq technology have facilitated the identification of thousands of regulated alternative splicing events in various biological contexts. Pioneer works that gave new insights into the complexity of alternative splicing include Pan et al. (2008); Wang et al. (2008b) and Mortazavi et al. (2008) By analysing mRNA-seq data across different human tissues, both Pan et al. (2008) and Wang et al. (2008b) estimated that $\sim$ 95% of human multi-exon genes undergo alternative splicing. Wang et al. (2008b) identified "switch-like" splicing events where exons exhibit dramatically different inclusion levels between different

tissues, and estimated that $\sim 85\%$ of multi-exon genes produce at least two distinct populations of mRNA isoforms such that the minor isoform exceeds 15% of the total expression level in a given tissue. Mortazavi et al. (2008) first quantified transcript expression levels using RNA-seq reads from a mouse transcriptome with a measure that allows direct comparison of transcript levels both within and between samples. The so-called RPKM measure (reads per kilobase per million mapped reads) is defined in section 3.2.2. Other important works include the ones by Nagalakshmi et al. (2008); Mortazavi et al. (2010) and Graveley et al. (2011), which studied in detail the transcriptome of model organisms such as *Drosophila melanogaster* and *C. elegans*, improving their reference annotations and providing enhanced transcriptome maps.

In addition, the ability offered by RNA-seq to detect and quantify rare and/or novel RNA transcript variants within a sample make it an appealing technology for clinical diagnosis purposes (Van Keuren-Jensen et al., 2014; Byron et al., 2016). As an example, one promising aspect of RNA-seq is the measurement of small amount of RNA molecules from blood samples containing fetal RNA or circulating tumor cells, which makes it possible to implement diagnostic tests and to monitor diseases in a non-invasive manner.

Considering translation of the RNA-seq technology into the clinic, we briefly describe below the targeted RNA-seq methodology. This variant of RNA-seq, by focusing on a set of transcripts from specific genes of interest, is well suited to a clinical environment.

Targeted RNA-seq

As described above, RNA-seq is a powerful tool to investigate the transcriptome, but it remains costly and generates complex data sets that limit its utility in routine molecular diagnosis testing. Targeted RNA-seq is a method for selecting and sequencing specific transcripts of interest (Mercer et al., 2012; Zheng et al., 2014). By focusing sequencing efforts on a subset of the transcriptome, it yields much higher coverage on the selected regions at a reduced sequencing cost and time. Enrichment of the regions of interest can be performed with several techniques (Mamanova et al., 2010), ranging from uniplex PCR or hybridization capture to multiplex PCR[4]. Targeted RNA-seq approaches have been used to detect fusion transcripts (Levin et al., 2009), allele-specific expression (Zhang et al., 2009) or RNA-editing events (Li et al., 2009) in a subset of transcripts.

[4] in multiplexed PCR several primer pairs are used in a single reaction, in contrast to uniplexed PCR where a single target is amplified in each reaction.

In chapter 6, we describe a targeted RNA-seq approach designed to reveal potential splicing abnormalities in patient sample characterized by the presence of mutations in their DNA *cis*-regulatory elements of splicing.

3.2 Computational challenges associated with RNA-seq reads

As previously explained, the RNA-seq technology allows the study of the transcriptome at an unprecedented resolution. It promises to be able to build a complete annotation and quantification of all genes and their isoforms across samples. However, to achieve this goal RNA-seq data need to be statistically analysed and computationally transformed. This section is devoted to highlighting the challenges in using RNA-seq reads to decipher a transcriptome and the different methodologies developed to overcome the difficulties inherent to the nature of the data. We first briefly explain the concept of read alignement (or "mapping") on a reference as this is the first step of many methodologies that try to infer the expressed isoforms. We then give an overview of how aligned reads can be statistically modeled (statistical models are at the core of the inference framework of many methodologies), and we finish by categorizing the existing approaches depending on their ultimate goals, input data or used algorithms.

Very good reviews describing the computational tools and methodologies that apply to RNA-seq data can be found in Garber et al. (2011); Martin and Wang (2011) and Alamancos et al. (2014).

3.2.1 Mapping RNA-seq reads

Mapping denotes the alignment of short sequence reads on a reference genome. The specificity of RNA-seq reads compared to reads derived from genome sequencing is that they are of two types: i) exon-body reads that map continuously on the reference genome and ii) junction reads that span the connection between different exons and map on the reference genome only if split in several pieces separated by large gaps. Junction reads are fundamental for the detection of alternative splicing events as they provide direct evidence of exon-exon joining events.

The main challenge in mapping RNA-seq reads arises from the fact that junction reads must be divided into short pieces that may be hard to map unambiguously. A "splice aligner" refers to a computational tool capable of mapping both exon-body and junction reads. Splice aligners

fall into two main categories (Garber et al., 2011): "exon-first" and "seed-and-extend". Exon-first methods first map reads continuously on the genome using an unspliced approach[5] to find read-clusters that represent potential exons. They then generate a database of potential splice junctions and map the remaining reads against these junctions. These methods include TopHat (Trapnell et al., 2009), MapSplice (Wang et al., 2010), SpliceMap (Au et al., 2010) and SOAPsplice (Huang et al., 2011). Seed-and-extend methods on the other hand, which include GNSAP (Wu and Nacu, 2010), PALMapper (Jean et al., 2010) and STAR (Dobin et al., 2013), break all reads into short substrings (called k-mers or seeds), first map these short pieces on the genome and then locally extend the seeds to find the best alignments for each read. Exon-first techniques generally ask for fewer computational resources than seed-extend methods but depend on sufficient coverage on potential exons to accurately map spliced reads and tend to be biased toward unspliced alignments (Chen, 2013).

Figure 3.4 corresponds to the result of RNA-seq reads aligned on the reference human genome. Reads are colored with respect to their strands[6] of origin. Mismatched bases are highlighted in different colors. The number of reads that sequence a given base is called *read count* or *coverage*. Coverage density along the reference is shown on the upper part of the figure. The coverage measure is of primordial importance for statistical inference of the expression levels of the transcript isoforms: intuitively it represents the "abundance" of a given nucleotidic base and, if integrated on several bases up to the entire exons, it would give information on the relative abundances of the exons, hence measuring the intensity of the different alternative splicing events. Section 3.2.2 gives a rigorous statistical analysis of the read counts.

Figure 3.5 gives additional insights on the coverage measure. It shows the histogram of read counts on a subpart of the *BRCA1* gene, resulting from the alignment of reads from a typical RNA-seq experiment with a total of 80 millions of mapped reads (a so-called *bulk* RNA-seq experiment) or from a targeted RNA-seq experiment with 40000 mapped reads in total. While the maximum read count on the bulk dataset is only 40, it reaches a value of 16300 on the targeted dataset. One can also note that some junctions between exons are supported by only one read on the bulk dataset. This allows us to anticipate that the statistical challenges will be different when analysing RNA-seq reads from a bulk versus a targeted experiment. In chapter 4, we use bulk RNA-seq read counts, like the ones depicted in figure 3.5, as input

[5] standard short sequence aligners (unspliced aligners) are based either on data compression techniques (such as the Burrows-Wheeler transform) or on hash tables, combined with computation of alignement scores such as the ones produce by the Smith-Waterman algorithm.

[6] in a paired-end RNA-seq dataset the two reads of a given pair come from opposite strands.

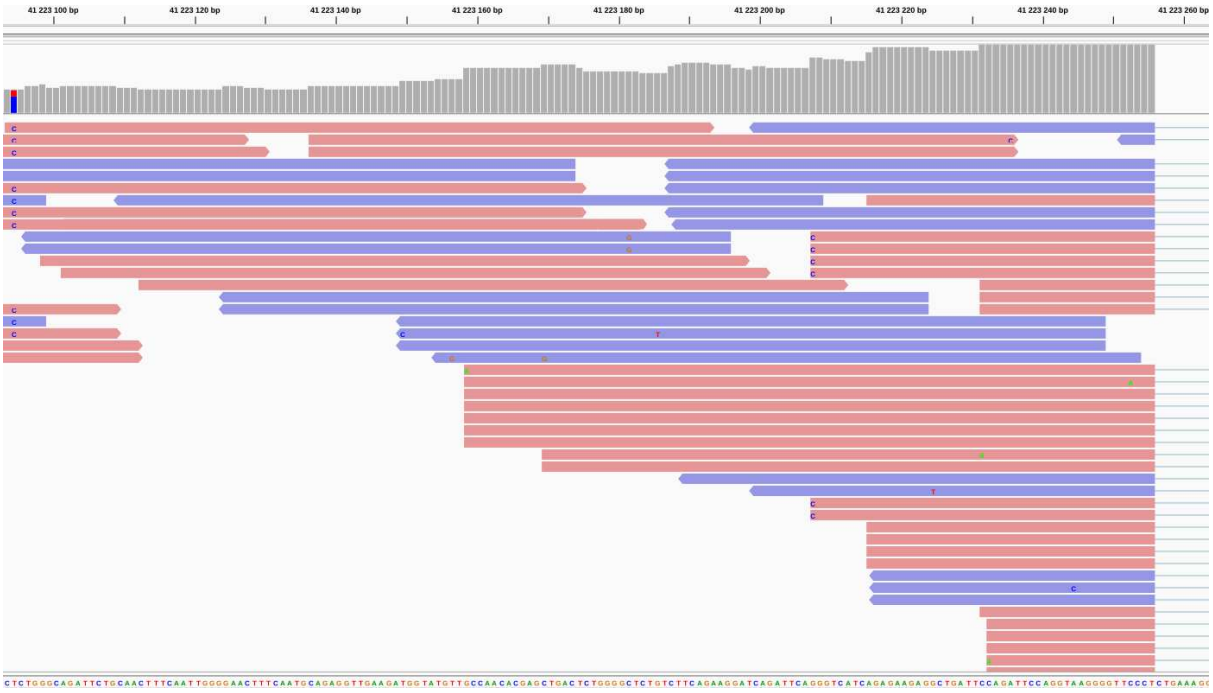


FIGURE 3.4: RNA-seq reads aligned on a reference genome. Forward reads are colored in red while reverse reads are colored in purple. Reads are 200bp long. The reference sequence is shown at the bottom. Base pairs inside the reads that disagree with the reference are shown in different colors. The second leftmost base pair probably corresponds to a heterozygous SNP (single nucleotide polymorphism) as we observe high proportions of both C and T. The thin blue lines on the right part of the figure indicate spliced junctions. The histogram at the top of the figure shows the coverage, *i.e.* the number of reads that sequence each base pair. The figure has been produced thanks to the Integrative genomic viewer tool (Robinson et al., 2011).

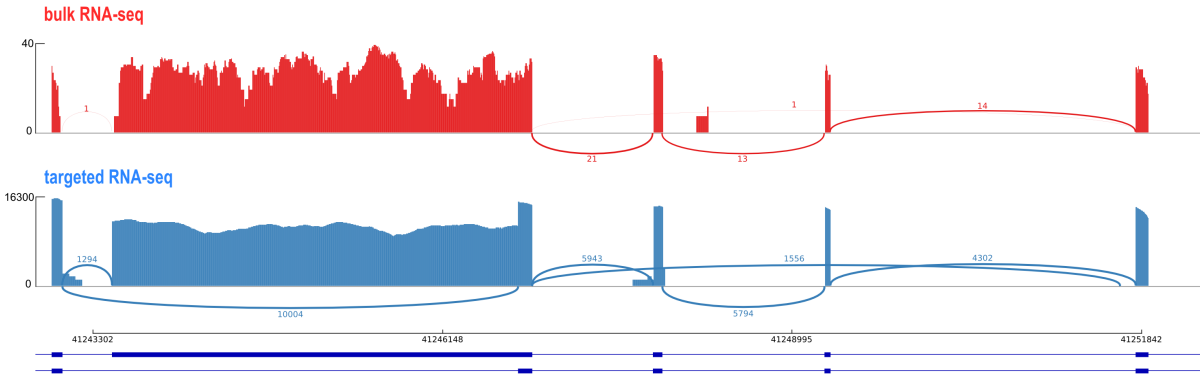


FIGURE 3.5: RNA-seq coverage density on a subpart of the *BRCA1* gene from both a bulk RNA-seq experiment (upper part) and a targeted RNA-seq experiment (lower part). Numbers on the arcs indicate the number of reads supporting a spliced junction.

to detect and quantify the isoforms of all sequenced genes. In chapter 5, we simultaneously analyse RNA-seq reads from several related samples as a way to increase coverage and therefore statistical power. In chapter 6 on the other hand we use targeted RNA-seq reads to decipher the transcriptomic landscape of a single gene. Note also that in these three chapters RNA-seq reads from simulations or real data have been mapped with the TopHat2 algorithm (Kim et al., 2013).

3.2.2 Modeling RNA-seq reads

Counting the reads that belong to a given region such as an exon is a starting point for statistical inference. However, read counts need to be appropriately modeled for accurate estimation of the transcript abundances. In this section, we give some clues about how to model the RNA-seq read counts and derive the needed likelihoods to infer the quantity of interest. We refer to Pachter (2011) for a detailed hierarchy of different models used in the context of transcript abundance estimation.

Notations

We first introduce some notations that we will use consistently throughout the thesis.

- We denote by $\mathcal{P}$ a set of possible transcript isoforms. The set can correspond to known transcripts or to a large number of candidate transcripts (see section 3.2.3). The notation $|.|$ refers to the cardinality of a given set, so that the number of transcripts is $|\mathcal{P}|$.
- We call a *bin* a succession of genomic positions that are continuous in at least one transcript. A bin typically corresponds to an exon or an ordered set of exons. We use the capital letter V to refer to a set of bins. Any transcript $p \in \mathcal{P}$ is a succession of bins. We denote by l_v the *effective length* of a given bin $v \in V$ and by l_p the *effective length* of a given transcript $p \in \mathcal{P}$, where the effective length[7] is defined as the number of genomic positions where a read can start to be included in a given bin or transcript. (see section 4.2.1 and figure 4.1 for a rigorous computation of the effective bin length).

[7] *stricto sensu* the effective length depends on the read length. For instance the effective length l_p of a transcript p is equal to $l_p =$ length of p $- L + 1$.

- The random variable Y_v counts the number of reads falling into[8] a bin $u \in V$. The observed value of Y_v is written in lower case y_v. The total number of reads is denoted as $N = \sum_{v \in V} y_v$.
- The quantities of interest are what we call the relative transcript abundances or transcript expression levels. We denote by $\boldsymbol{\beta}$ the vector of transcript abundances of size $|\mathcal{P}|$, such that each entry β_p of the vector is a non-negative value representing the abundance of transcript p. We also denote by α_p the number of copies of transcript p.

Note that the use of notations $\mathcal{P}$ and V to denote the sets of transcript and bins is not random but borrowed from the network flow literature, a choice that will be made clear when we develop in section 3.3.3 the equivalence between the transcript inference task and network flow optimization problems.

Multinomial distribution and uniform sampling

A simple though widely used model (Jiang and Wong, 2009; Pachter, 2011) of RNA-seq read counts is to assume that reads are sampled uniformly across the different transcripts and across the positions of the transcripts. If we further approximate the sampling of the reads by a draw with replacement, we end up with a Multinomial model of the read counts, where the probabilities of successes are linear combinations of the relative abundances of the transcripts.

Indeed, if we call p_v the "probability of success" that a read falls into a bin v and if we suppose a uniform sampling, p_v is equal to the ratio between the number of favorable positions and the total number of positions, that is

$$p_v = \frac{l_v \sum_{p \in \mathcal{P}: p \ni v} \alpha_p}{\sum_{v \in V} l_v \sum_{p \in \mathcal{P}: p \ni v} \alpha_p} \ .$$

One can easily check that $p_v \geq 0$ for all $v \in V$ and that $\sum_{v \in V} p_v = 1$. One can also note that by inverting the two sums of the denominator, the denominator quantity is equal to $\sum_{p \in \mathcal{P}} l_p \alpha_p$. Hence we have the following simplified form for the success probability

$$p_v = l_v \sum_{p \in \mathcal{P}: p \ni v} \beta_p \quad \text{with} \quad \beta_p = \frac{\alpha_p}{\sum_{p \in \mathcal{P}} l_p \alpha_p} \ , \tag{3.1}$$

[8] when we say that a read fall into a bin we mean that the read is contained in the bin

where the β_p are often the quantities of interest in the context of transcript abundance estimation. Rigorously, β_p represents the number of transcripts p per unit of length, and satisfies $\sum_{p\in\mathcal{P}} l_p\beta_p = 1$. If multiplied by $10^9/N$, it is consistent with the RPKM unit introduced by Mortazavi et al. (2008) in the early days of RNA-seq.

The likelihood of a model is a function of the model's parameters that is equal to the probability of observing the data under the model. In our case the likelihood is defined as $L(\boldsymbol{\beta}) = P(\{Y_v = y_v, v \in V\})$. Assuming that the reads are independent, the likelihood function under the Multinomial model is equal to

$$L(\boldsymbol{\beta}) = \frac{N!}{\prod_{v\in V} y_v!} \prod_{v\in V} p_v^{y_v} \quad \text{with} \quad p_v = l_v \sum_{p\in\mathcal{P}:p\ni v} \beta_p \ .$$

The log-likelihood of the model is hence given by

$$\log L(\boldsymbol{\beta}) = \sum_{v\in V} y_v \log\Big(l_v \sum_{p\in\mathcal{P}:p\ni v} \beta_p\Big) + \text{constant} \ , \tag{3.2}$$

where the constant does not depend on the parameters of interest. Note that $\log L(\boldsymbol{\beta})$ is a concave function (as the sum and compositions of several concave functions) for which any maximum is guaranteed to be a global maxima.

Several works (Jiang and Wong, 2009; Turro et al., 2011; Trapnell et al., 2010) on transcript abundances estimation model the read counts as Poisson distributed. This is usually justified by the fact that a Binomial distribution is well approximated by a Poisson distribution for a large number of trials. More specifically, if $\{Y_v, v \in V\}$ follows a Multinomial distribution characterized by N trials and $\{p_v, v \in V\}$ probabilities of successes and if the Y_v are independent then $Y_v \sim \mathcal{B}(N, p_v)$, *i.e.* follows a binomial distribution. When N is large $\mathcal{B}(N, p_v)$ is well approximated by the Poisson distribution $\mathcal{P}(Np_v)$. Figure 3.6 shows that the Binomial and Poisson distributions are already close for $N = 100$ and $p = 0.1$. Under the Poisson model, the likelihood denoted as $\tilde{L}$ is equal to

$$\tilde{L}(\boldsymbol{\beta}) = \prod_{v\in V} e^{-Np_v} \frac{(Np_v)^{y_v}}{y_v!} \ ,$$

so that the log-likelihood is equal to

$$\log \tilde{L}(\boldsymbol{\beta}) = \sum_{v\in V} \Big[- Nl_v \sum_{p\in\mathcal{P}:p\ni v} \beta_p + y_v \log\Big(Nl_v \sum_{p\in\mathcal{P}:p\ni v} \beta_p\Big)\Big] + \text{constant} \ , \tag{3.3}$$

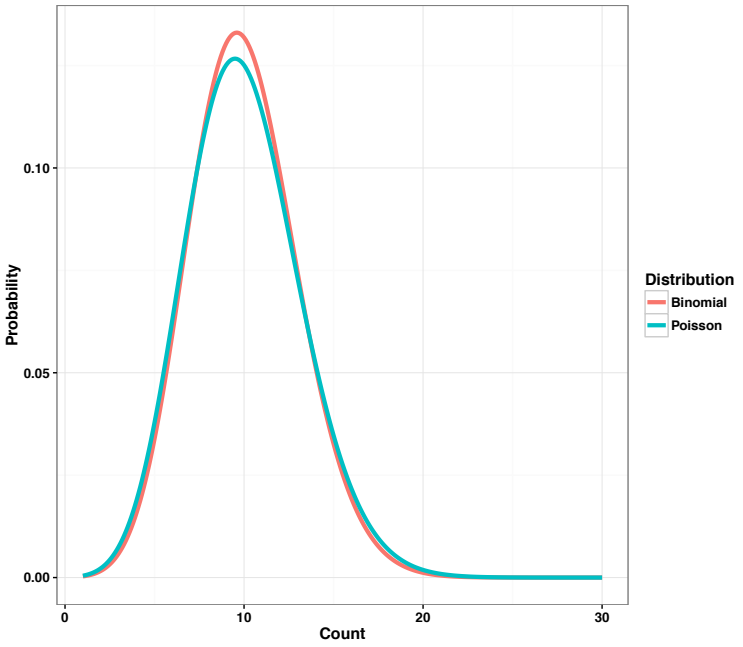


FIGURE 3.6: Comparison of Binomial $\mathcal{B}(N, p)$ and Poisson $\mathcal{P}(Np)$ distributions with $N = 100$ and $p = 0.1$.

where again the constant does not depend on the parameters of interest. Similarly to equation (3.2), $\log \tilde{L}(\boldsymbol{\beta})$ is a concave function. Note that in the following chapters 4, 5 and 6 we use the Poisson log-likelihood (3.3).

Beyond uniform sampling

The uniform sampling assumption is a simplified version of the reality of RNA-seq data. Indeed it has been observed that RNA-seq reads are subject to positional (Howard and Heber, 2010) and sequence (Zheng et al., 2011) biases, and that the non-uniformity of the counts is too high to fit a Poisson distribution with rates that are simply linear combinations of the isoform abundances (Li et al., 2010).

One possibility to go beyond the uniform sampling assumption (Pachter, 2011) is to model reads at the genomic position level instead of at the bin level, and to model the conditional probability $P(r_j \backslash p)$ that an observed read r_j originates at the position j from transcript p additionally characterized by a sequence composition context (Roberts et al., 2011). More formally, and similarly to equation (3.1), the probability of success that a read starts at position j can be modeled as

$$p_j = \sum_{p \in \mathcal{P}: p \ni j} a_{jp} \beta_p \ ,$$

where a_{jp} is a sampling rate that depends on the position j in the transcript p. The variety of models that deal with the p_j is quite large, ranging from Poisson (Li et al., 2010) to Quasi-Multinomial (Li and Jiang, 2012a), Beta-Binomial (Roberts and Pachter, 2013) or Negative-Binomial distributions (Li and Jiang, 2014). Under the Poisson model we simply have that $Y_j \sim \mathcal{P}(Np_j)$ where Y_j counts the number of reads that originate from position j, which generalizes the analysis resulting in equation (3.3) to a base pair resolution with various sampling rates.

But modeling each read may be computationally challenging as a single RNA-seq experiment produces millions of such reads. Reads that share the same sampling rate can however be collapsed into groups without loss of information. These groups are called "read classes" in Salzman et al. (2011) and "equivalent classes" in Nicolae et al. (2011) and Bray et al. (2016).

What we called a bin v can be in fact defined as a succession of genomic positions derived from collapsing a set of reads $\{r_j\}$ that are sampled from the same set of transcripts at the same rates $\{a_{jp}\}$. Because the sum of independent Poisson random variables is a Poisson random variables with summed parameters, and by denoting with a_{up} the sampling rates associated with bin v and transcript p (which is equal to any a_{jp} of the collapsing), we then have

$$Y_v = \sum_{j \in V} Y_j \sim \mathcal{P}(N l_v \sum_{p \in P : p \ni v} a_{vp} \beta_p) \ , \tag{3.4}$$

which again generalizes (3.3). If one is only interested in modeling sample rates that do not depend on the bin position (as sequence biases for instance), then $Y_v \sim \mathcal{P}(N \tilde{l}_v \sum_{p \in P : p \ni u} \beta_p)$ where $\tilde{l}_v = l_v a_v$. More details about modeling sampling rates a_{jp} can be found in Salzman et al. (2011).

Note that in chapter 6 we also tackle the non-uniformity of RNA-seq read counts. We do not explicitly model sampling rates a_{vp} that could be plugged in equation (3.4), but, in a similar flavor, we calculate scaling factors for each bin based on a set of control samples in order to attenuate non-uniformity.

Finally, note that we focused above on modeling single-end RNA-seq reads. The analysis can be extended to paired-end reads by additionally modeling the fragment length (or insert size) distribution, *i.e.* the size between the two reads of a pair. This goes beyond the scope of the thesis and we refer to Rossell et al. (2014) for a rigorous derivation of a paired-end statistical model.

3.2.3 The isoform deconvolution problem

We describe in this section the different approaches that exist to assemble the full-length transcript isoforms and estimate their expression levels (or abundances) from RNA-seq data. Detecting and/or quantifying the transcript isoforms from sequenced reads is a challenging task: reads are short, and transcript isoforms from the same gene share exons, making it difficult to resolve which isoform produced each read. We denote the task of piecing together short reads into transcripts and additionally estimating their relative expression levels as the *isoform deconvolution problem*.

We describe below the three main categories of approaches that tackle the isoform deconvolution problem:

- **Annotation-based transcript expression quantification**. Methods from this category consider a set of known transcripts as given and estimate the expression levels of each of the annotated isoforms. Databases that store known transcripts, such as RefSeq (Pruitt et al., 2005), Ensembl (Cunningham et al., 2015) or GENCODE (Harrow et al., 2012) for the human transcriptome, are valuable ressources for these annotation-based approaches. In addition, these methods can be run after a *de novo* transcriptome assembly step performed by some genome-independent or genome-guided tools described below.

  There is a lot of different annotation-based approaches. They are all based on modeling the RNA-seq reads, similarly to what we described in section 3.2.2, and computing a likelihood from their model that allows a maximum-likelihood estimation or a Bayesian estimation procedure.

  rSeq (Jiang and Wong, 2009) and MMSEQ (Turro et al., 2011) use a Poisson likelihood for the read counts at the level of bins with uniform sampling rate. They solve a convex program for the maximum likelihood estimation, and further provide an uncertainty of their estimates using importance sampling from the posterior distribution. CEM (Li and Jiang, 2012a) additionally incorporates a positional bias term in its model. Casper (Rossell et al., 2014) models paired-end reads at the level of bins (what it calls "exon path") and estimates the abundances in a Bayesian framework that incorporates a previously fitted read start distribution.

  rQuant (Bohnert and Rätsch, 2010) solves a linear model jointly over the transcript abundances and over some bias parameters.

RSEM (Li and Dewey, 2011), IsoEM (Nicolae et al., 2011), BitSeq (Glaus et al., 2012) and eXpress (Roberts and Pachter, 2013) derive a model at the read level and model paired-end fragment size. RSEM, BitSeq and eXpress additionally take non-uniformity into account. BitSeq relies on a Bayesian framework and RSEM also implements a Bayesian version of its model to report confidence intervals.

Finally, Sailfish (Patro et al., 2014) and Kallisto (Bray et al., 2016) are recent "lightweight" algorithms that avoid the mapping step. Using hash tables of k-mers they can quickly associate reads to transcripts (at the cost of losing the position information) which is enough to derive a uniform sampling model.

- **Genome-independent transcript reconstruction**. This category denotes methods that assemble the reads directly into transcripts without using a reference genome. Genome-independent assemblers merge reads into transcriptional units without a mapping step. As so, they can provide an initial set of transcripts in sample from organisms that do not have a high-quality reference genome or when the genome is affected by numerous rearrangements like in cancer cells. The reconstructed transcripts can be additionally fed to annotation-based methods to estimate their abundances. Genome-independent methods include Trans-AbySS (Robertson et al., 2010), Trinity (Grabherr et al., 2011), OASES (Schulz et al., 2012) and SOAPdenovo-Trans (Xie et al., 2014).

  The core concept of these methods is to find overlaps between the reads to assemble them into transcripts. Most of the tools build the so-called "de Bruijn graph" from short k-mers (sub-sequences of length k) derived from the reads. The k-mers represent the nodes of the graph and pairs of nodes are connected if shifting a sequence by one character creates an exact k-1 overlap between the two sequences. Transcripts are recovered as paths through the de Bruijn graph with sufficient read coverage. The major drawback of genome-independent techniques is that they are very sensitive to sequencing errors as these introduce branch points in the graph. The choice of the k-mer length also affects the assembly. Small values of k lead to a complex graph while large values of k restrict the overlap between the nodes.

- **Genome-guided transcript estimation**. Approaches from this last category rely on a reference genome to first map the RNA-seq reads with a splice aligner as described in section 3.2.1. Mapped reads are then assembled into transcripts and their expression levels are estimated with the use of read counts. Some methods only reconstruct the full-length transcripts while others both detect and quantify the transcript isoforms.

Our contributions to the isoform deconvolution problem lie in that category. Therefore, we dedicate the following section (section 3.3) to a more detailed description of genome-guided approaches. We classify the different techniques with respect to their rationale and their algorithms, and emphasize where the methods we developed –described in chapters 4 and 5– fit in the plethora of existing tools.

3.3 Genome-guided transcript estimation

The genome-guided transcript estimation task denotes the deconvolution of short reads aligned on a reference genome into a set of expressed transcripts, additionally with an estimation of their associated relative abundances. In this section, we first describe and categorize the different techniques that tackle the isoform deconvolution problem, before giving some intuitions and definitions on two techniques used in that context, namely the concept of sparse regression with ℓ_1 penalty and network flow optimization.

3.3.1 Inferring transcripts with various techniques

The first major distinction between the different genome-guided approaches is whether or not they use solely the locations of the mapped reads to reconstruct the full-length transcripts or if they also use the read count levels to help the reconstruction. Indeed, knowing the relative abundances of the different expressed regions of a gene (typically the exons or sub-parts of the exons that we denote as "sub-exons") may help to assemble the transcripts by resolving ambiguities.

To illustrate this claim, figure 3.7 depicts a toy example where taking into account the read count levels leads to a better inference of the expressed transcripts. In this example, we sample real RNA-seq data on 4 exons simultaneously and we repeat the procedure on the first 2 exons. This sampling simulates a 4-exon gene that expresses two transcripts (a 4-exon transcript and a shorter 2-exon transcript) with the same abundances on average. Performing the transcript reconstruction with a tool that uses the sole mapped read positions (we use the Cufflinks (Trapnell et al., 2010) software here, see below) only recovers the longer transcript. When using a tool that takes read count levels into account (we use the FlipFlop (Bernard et al., 2014) software here, see below and chapter 4), the shorter transcript pops up as well.

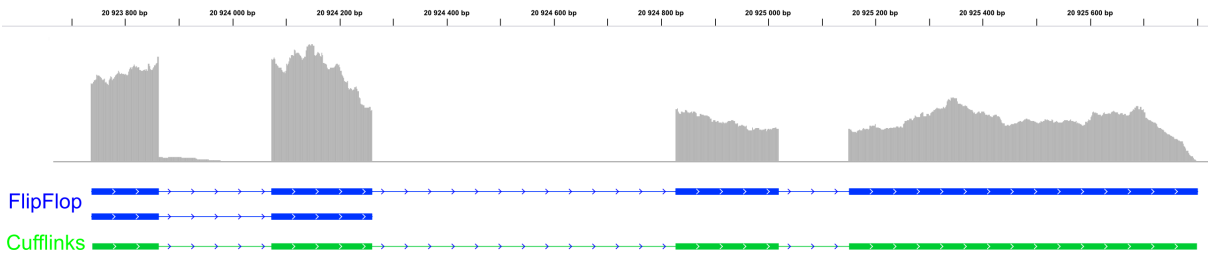


FIGURE 3.7: Benefits of using read count levels to assemble transcripts. Coverage density on exons is shown in grey. FlipFlop transcript predictions are shown in dark blue (with estimated expression levels of 56% and 44%). Cufflinks transcript prediction is shown in green.

The second important divergence between the existing approaches is related to the need of exhaustive enumeration of the candidate transcripts. In contrast to the annotation-guided methods described in section 3.2.3, genome-guided methods do not assume that the set of expressed transcripts is known *a priori*. This difference is fundamental when the ultimate goal is to annotate a given genome or to infer the transcriptome of cells that diverge from wild-type conditions (as cancer cells) and that are expected to express unknown transcripts.

All genome-guided methods rely on an explicit or implicit graph model. The type of graph differs depending on the methods: it can be at the level of individual reads (Trapnell et al., 2010; Guttman et al., 2010) or at the level of bins, where a bin typically represents an exon (Tomescu et al., 2013), a sub-exon (Li et al., 2011b,a) or an ordered set of exons (Bernard et al., 2014). Graphs at the level of bins are often called "splicing graph", first introduced by Heber et al. (2002) in the context of EST assembly. In its original definition, the nodes of the splicing graph represent exons, and the directed edges represent splice junctions. In all cases, because the graph models arise from alignements to a reference sequence, they are directed and acyclic graphs (DAGs), and any path in the DAG corresponds to a possible transcript. Each node and/or edge can be associated with an observed coverage value, and the problem of isoform identification and quantification can be cast as separating the coverage of the graph into individual paths. Most methods rely on an explicit enumeration of the candidate transcripts by exhaustively listing all the paths of their underlying graph model. If V denotes the set of nodes of the graph, the number of paths in the graph grows exponentially with $|V|$, typically of the order of $2^{|V|}$. Some other methods, including the one we describe in chapter 4, avoid an explicit enumeration (see below).

We describe below the different existing (to the best of our knowledge) genome-guided approaches categorized with respect to the algorithms they implement.

1. **Graph traversal.** Some approaches do not use the read counts to reconstruct the full-length transcripts but solely the alignement positions. Cufflinks (Trapnell et al., 2010) and Scripture (Guttman et al., 2010) are both precursor tools in the RNA-seq assembly field that belong to this category.

   Cufflinks builds a so-called "overlap graph" by connecting compatible aligned reads. Reads are compatible if they overlap and share the same splice patterns. Their resulting graph is a DAG. Cufflinks then uses a parsimony-based approach to infer the transcripts by computing a minimal set of transcripts through the graph that explain the reads, *i.e.* such that all reads are included in at least one path. The task of finding a set of paths which cover all the nodes of a directed graph with minimum cardinality is called a minimum path cover (MPC) problem. MPC problems are NP-hard[9] in general but solvable in polynomial time on DAGs. Note also that Cufflinks estimates the relative abundances of the reconstructed transcripts in a second step using a maximum likelihood approach similar to the one described in section 3.2.2.

   Scripture uses a similar graph as Cufflinks but enumerates all possible paths in the graph with some heuristics based on paired-end reads and coverage to filter some of the paths.

   CLASS (Song and Florea, 2013) is similar in spirit to Cufflinks but adds constraints in the transcript assembly process derived from paired-end reads, making the problem NP-hard.

2. **Integer linear programming.** Several tools including TRIP (Mangul et al., 2012), CLIIQ (Lin et al., 2012) and MiTie (Behr et al., 2013) formulate the transcript inference task as an integer linear program. All these programs are combinatorially difficult in the sense that their worst case complexity is as large as $O(2^{2^{|V|}})$. While TRIP and CLIIQ explicitly enumerate the candidate transcripts, MiTie avoids an explicit enumeration using branch-and-bound techniques (Behr et al., 2013).

   Note also that CLIIQ and MiTie can use several samples simultaneously to reconstruct the transcript isoforms. In chapter 5, we also describe an approach to solve the isoform deconvolution problem jointly across related samples, but in the framework of a convex sparse regression (see below).

3. **Sparse regression.** Several methods use the read counts to formulate the transcript inference task as a regression problem, and therefore minimize a loss function with respect

[9] roughly, a NP-hard problem is a problem for which it is likely that no polynomial time algorithm can solves every single instance of the problem.

to the transcript abundances. The loss function, or cost function, measures how well the estimated abundances explain the observed read counts.

Additionally, to avoid over-fitting and in order to select a parsimonious set of expressed transcripts, the methods add a penalization term to the regression problem that promotes sparse solutions, that is solutions involving few transcripts.

The MiTie method (Behr et al., 2013), cited in the above category, also belongs to this category as when penalizing its loss function with the number of selected transcripts, *i.e.* the number of transcripts associated with non-zero abundances. This penalization, known as the ℓ_0-pseudo-norm, however leads to a NP-hard regression task. Montebello (Hiller and Wong, 2013) also tries to solve a ℓ_0-penalized regression using Monte Carlo simulations.

To overcome the combinatorial difficulty of the ℓ_0-pseudo-norm, several other tools instead use a convex relaxation, such as the popular ℓ_1-norm. The ℓ_1-norm is defined in our setting as the sum of the non-negative transcript abundances. In section 3.3.2, we discuss the geometry of the ℓ_1-norm and the reasons why it promotes sparsity.

Methods such as IsoLasso (Li et al., 2011b), NSMAP (Xia et al., 2011), SLIDE (Li et al., 2011a) and CIDANE (Canzar et al., 2016) use convex optimization techniques to solve a quadratic program (or a more general convex program) involving a ℓ_1-penalization. IsoLasso, SLIDE and CIDANE use a least square loss function, while NSMAP uses a Poisson loss function derived from the log-likelihood given in equation (3.3). CEM (Li and Jiang, 2012b) defines a more subtle loss function in order to estimate additional bias parameters from the data at the cost of solving a non-convex ℓ_1-penalized program. iReckon (Mezlini et al., 2013) uses an esoteric penalization (the exponential of the sum the fourth square root of the transcript abundances) which is non-convex.

However, while using a ℓ_1-penalization is appealing for its convexity and successes in several fields (Mairal, 2010), all the tools mentioned above deal with an explicit enumeration of the candidate transcripts. Hence they solve a convex program with polynomial complexity with respect to $2^{|V|}$ variables. The exponential explosion makes the problem in fact intractable when $|V|$ grows, and is already very challenging for a gene with 20 exons or sub-exons.

In practice the tools resort to various heuristics to limit the exponential size of the candidate set. NSMAP and iReckon restrict the possible transcripts to the ones starting and ending at a known transcription start site and polyA site. SLIDE lists the transcripts from

genes that only have less than 10 exons, and IsoLasso uses strong filtering rules based on the coverage and transcription starting and polyA sites.

In chapter 4, we describe a method that also takes advantage of the sparsity-inducing properties of the ℓ_1-norm but does not need to enumerate the candidate transcripts. In other words, we provide a way to solve the isoform deconvolution problem within the ℓ_1-penalization framework without *ad hoc* filtering on the candidate transcripts set and by using efficient algorithms that are polynomial in $|V|$. In chapter 5, we tackle the isoform deconvolution problem simultaneously across several samples by using a generalization of the ℓ_1-norm to a multidimensional case (the so-called $\ell_{1,2}$-norm, see section 5.2).

Our method and associated software called FlipFlop (see Bernard et al. (2014) and appendix C) rely on network flow optimization techniques (see below and section 3.3.3).

4. **Network flow optimization.** Some methods take advantage of the structure of the problem, transcripts being paths in a DAG, to build equivalences between the transcript inference task and network flow optimization problems. By doing so, these methods avoid to enumerate all possible paths in the underlying graph that correspond to the candidate transcripts.

   The concept of network flow and associated optimization problems is introduced in section 3.3.3. Intuitively in our setting, a value can be attributed on every node or edge of a DAG by summing the abundances associated with each path (*i.e.* each transcript) of the graph. The set of computed values is called a *flow* and optimization problems can be solved equivalently by manipulated flows or transcript abundances.

   Traph (Tomescu et al., 2013) solves a minimum cost network flow problem over the splicing graph, which can be done in polynomial time. Traph then needs to decompose the optimal flow into a few paths, but decomposing a flow into a minimum number of paths is an NP-hard problem. Hence it uses approximation algorithms to split the flow into a few paths (Hartman et al., 2012).

   StringTie (Pertea et al., 2015) implements a greedy algorithm to harvest the heaviest path and estimate its expression through a maximum flow optimization, which can be done in polynomial time as well.

   The particularity of our method FlipFlop is to fit the ℓ_1-norm penalization into the network flow framework. Hence we estimate a flow that incorporates the sparsity constraint of the

regression problem. By doing so, we provide a way to combine the efficiency of network flow optimization techniques with a tight control on the sparsity of the solutions.

5. **Bayesian assembly.** Bayesembler (Maretty et al., 2014) is a probabilistic approach to transcriptome assembly. It infers the posterior distribution over the abundance levels of a Bayesian model using Gibbs sampling, and therefore quantifies the degree of confidence in the estimated transcripts.

Finally, table 3.1 summarizes what has been said above and gives an overview of the different genome-guided methods. The upper part of the table reports the methods that reconstruct the transcripts from the positions of the mapped reads only while the bottom part reports the methods that use the read count levels to decipher the expressed transcripts. Each of the two categories is further split in two again depending on the need to exhaustively enumerate the candidate transcripts. Additional information regarding the use of a penalization term, of specific graph algorithms and the ability to use several samples simultaneously is also given in the table.

3.3.2 ℓ_1-norm penalization

In this section, we briefly introduce the concept of penalization and we provide some insights on why penalizing a regression problem with the ℓ_1-norm encourages sparse solutions. We refer to Bach et al. (2012a) and Bach et al. (2012b) for detailed although accessible introductions to optimization with sparsity-inducing penalties.

In a regression problem, one wishes to estimate a vector of parameters $\boldsymbol{\beta}$ in $\mathbb{R}^{|\mathcal{P}|}$ by minimizing a loss function that measures how well $\boldsymbol{\beta}$ fits the observed data. We suppose that a loss function $\mathcal{L} : \mathbb{R}^{|\mathcal{P}|} \to \mathbb{R}_+$ that is smooth[10] and convex is given. The loss function is often derived from the negative log-likelihood of a statistical model designed to explain the observed data. The goal of a sparse regression (or sparse penalized regression) is to force the parameter vector $\boldsymbol{\beta}$ to be sparse, that is to contain a small number of non-zero components. In our context of transcript inference, we aim at selecting a few expressed transcripts among a very large number of candidate transcripts through the minimization of a loss function $\mathcal{L}$ that measures how well the transcript abundances explain the observed read counts. The loss function can be derived

[10] differentiable with Lipschitz-continuous gradient

		Software	Penalty	Graph algorithm	Multiple samples
without read counts	no enumeration	Cufflinks (Trapnell et al., 2010)		minimum path cover	✓*
		CLASS (Song and Florea, 2013)		minimum set cover	
	enumeration	Scripture (Guttman et al., 2010)			
with read counts	no enumeration	Traph (Tomescu et al., 2013)		minimum cost flow	
		MiTie (Behr et al., 2013)	ℓ_0		✓
		FlipFlop (Bernard et al., 2014, 2015)	$\ell_1, \ell_{1,2}$	minimum cost flow	✓
		StringTie (Pertea et al., 2015)		maximum flow	
	enumeration	IsoLasso (Li et al., 2011b)	ℓ_1		
		NSMAP (Xia et al., 2011)	ℓ_1		
		SLIDE (Li et al., 2011a)	ℓ_1		
		CEM (Li and Jiang, 2012b)	ℓ_1		
		TRIP (Mangul et al., 2012)			
		CLIIQ (Lin et al., 2012)			✓
		iReckon (Mezlini et al., 2013)	other**		
		Montebello (Hiller and Wong, 2013)	ℓ_0		
		Bayesembler (Maretty et al., 2014)			
		CIDANE (Canzar et al., 2016)	ℓ_1		

TABLE 3.1: Overview of genome-guided transcript estimation softwares. The tools are clustered depending on whether or not they use the read counts to assemble the transcripts and whether or not they need to exhaustively enumerate all the candidate transcripts. In each category the tools are ordered based on their publication date. *: the Cufflinks software does not use multiple samples *stricto sensu*. It however has a companion script called Cuffmerge that uses the predictions performed by Cufflinks on each sample and merge them with some heuristics to produce the final set of transcripts expressed across samples with different expression levels. **: if $\boldsymbol{\beta}$ denotes the transcript abundance vector and $\mathcal{P}$ the set of candidate transcripts, the penalty used by iReckon is equal to $\exp^{\sum_{p=1}^{|\mathcal{P}|}(\beta_p)^{1/4}}$ (this penalty has not been described elsewhere).

from negative log-likelihoods such as the ones given in section 3.2.2. In other words, we wish to estimate a vector $\boldsymbol{\beta}$ of transcript abundances that contains only a few non-zero entries.

A natural approach to control the sparsity of the solution would be to constrain explicitly the number of non-zero components in $\boldsymbol{\beta}$. The so-called ℓ_0-pseudo-norm, defined as $\| \boldsymbol{\beta} \|_0 = |\{p : \beta_p \neq 0\}|$ records the number of non-zero entries. Hence estimating a parameter vector with only a few non-zero components could be in theory performed through the following optimization problem:

$$\min_{\boldsymbol{\beta}} \quad \mathcal{L}(\boldsymbol{\beta}) \quad \text{such that} \quad \| \boldsymbol{\beta} \|_0 \leq k \ , \tag{3.5}$$

with $k \in \mathbb{N}$ a parameter controlling the size of the solution. However solving (3.5) requires an exhaustive search over all the possible combinations of $|\mathcal{P}|$ variables, a combinatorial problem (Natarajan, 1995) that becomes quickly intractable for $|\mathcal{P}|$ larger than a few tens. Moreover, the ℓ_0-pseudo-norm is not convex and therefore leads to the local minima problem.

A well-known approach to overcome the computational issue inherent to the ℓ_0-pseudo-norm is to instead use the ℓ_1-norm. The use of the ℓ_1-norm as a way to infer sparse models has been popularized in statistics by Tibshirani (1996) and independently in signal processing by Chen et al. (1998), and has been a topic of intensive research over the last decade (Mairal, 2010). The ℓ_1-norm, defined as

$$\| \boldsymbol{\beta} \|_1 = \sum_{p=1}^{|\mathcal{P}|} |\beta_p| \ ,$$

is convex (like any norm) and piecewise linear. The ℓ_1-norm is a convex surrogate of the ℓ_0-pseudo-norm that preserves its desired sparsity-inducing properties but is amenable to optimization. Indeed, estimating a parameter vector by solving

$$\min_{\boldsymbol{\beta}} \quad \mathcal{L}(\boldsymbol{\beta}) \quad \text{such that} \quad \| \boldsymbol{\beta} \|_1 \leq s \ , \tag{3.6}$$

with $s > 0$ often leads to sparse solutions. Moreover (3.6) is a convex program that can be solved with a variety of convex (although non-smooth) optimization algorithms, such as coordinate descent techniques (Daubechies et al., 2004), proximal methods (Beck and Teboulle, 2009) or active-set methods (Efron et al., 2004; Rosset and Zhu, 2007).

In equation (3.6) the ℓ_1-norm of the parameter vector $\boldsymbol{\beta}$ is controlled by the non-negative parameter s that can be seen as the maximal radius of a ℓ_1-ball. Looking at the geometry of the ℓ_1-norm and associated ℓ_1-ball gives insight on the reason why it promotes sparsity. Figure 3.8 shows the unit ball of the ℓ_1-norm in 2D, that is it shows $\{(\beta_1, \beta_2) : \| \boldsymbol{\beta} \|_1 = 1\}$, which has a diamond shape. The ℓ_1-ball is anisotropic and exhibits singular points due to the non-smoothness of the norm. Also shown in the figure are the level sets of a quadratic function that could represent the loss function $\mathcal{L}$. We see that the values of (β_1, β_2) minimizing the loss function while respecting the ℓ_1 constraint lie on one of the singularities of the ball, so that one of the coordinates vanishes ($\beta_2 = 0$). In comparison, the unit ball of the ℓ_2-norm[11] (*i.e.* the euclidean norm) is also shown on figure 3.8. In contrast to the ℓ_1-norm, the ℓ_2-norm is isotropic and hence does not favor some coordinates to be equal to zero. Figure 3.9 displays

[11] the ℓ_2-norm is defined as $\| \boldsymbol{\beta} \|_2 = \sqrt{\sum_p \beta_p^2}$

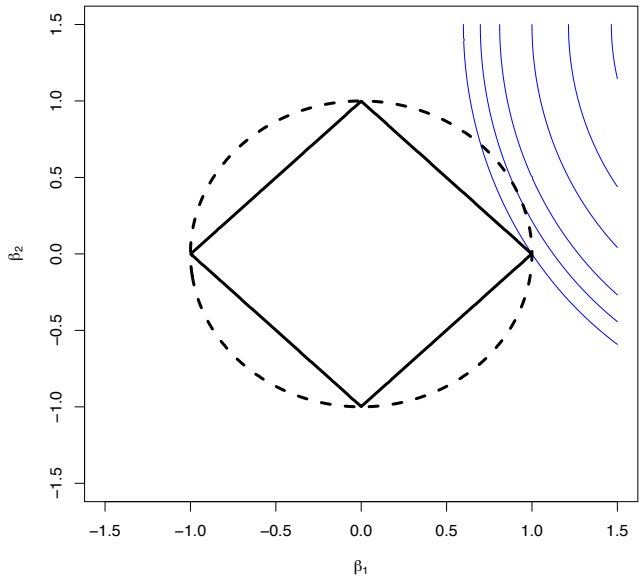


FIGURE 3.8: Sparsity induction by the ℓ_1-norm. The thick solid line represents the unit ℓ_1-ball characterized by the equation $|\beta_1| + |\beta_2| = 1$ while the thick dotted line represents the unit ℓ_2-ball characterized by the equation $\sqrt{\beta_1^2 + \beta_2^2} = 1$. The blue lines represent the level sets of a loss function (the lower value being in the upper right corner).

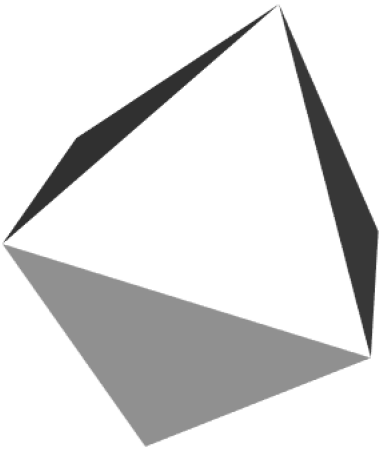


FIGURE 3.9: Pyramidal shape of the ℓ_1-ball.

the ℓ_1-unit-ball in 3D which has a pyramidal shape, highlighting even more strongly than in the 2D case the presence of singularities that would drive the solution of (3.6) towards the corners. Mairal et al. (2014) provides very good intuitions on why the ℓ_1-norm leads to sparse solutions, using both analytical as well as physical and geometrical arguments.

Equation (3.6) is sometimes called a constrained regression problem. A Lagrangian argument (Boyd and Vandenberghe, 2004) tells us that for any $s > 0$ there exists $\lambda > 0$ such that a solution of (3.6) is also a solution of

$$\min_{\boldsymbol{\beta}} \quad \mathcal{L}(\boldsymbol{\beta}) + \lambda \| \boldsymbol{\beta} \|_1 \ . \tag{3.7}$$

Given that the converse is also true, the two optimization problems (3.6) and (3.7) are strictly equivalent. Even though it is easier to understand the sparsity effect of the ℓ_1-norm by focusing on the constrained version (3.6) of the regression, the denomination *penalization* becomes clear when looking at (3.7) as an additional term (a penalization term) is added to the standard loss function. λ is a non-negative regularization parameter that controls the degree of sparsity of the solutions by adjusting the trade-off between a data fitting term and the ℓ_1-norm. Small values of λ lead to complex models with many selected variables while large values of λ favor solution vectors with many entries set to zero. Note that additional convex constraints on $\boldsymbol{\beta}$ can be added to the penalized regression (3.7). In our case we ask the transcript abundances to be non-negative, *i.e.* we constraint $\boldsymbol{\beta}$ to belong to $\mathbb{R}_+^{|\mathcal{P}|}$.

Finally, we emphasize that the ℓ_1-penalized problem (3.7) is a particular instance of a more general optimization of the form $\min_{\boldsymbol{\beta}} \quad \mathcal{L}(\boldsymbol{\beta}) + \lambda\Omega(\boldsymbol{\beta})$ where $\Omega(.)$ is any norm such that its geometry enforces some sparsity patterns of the solutions. In chapter 5, we use a norm (more specifically the $\ell_{1,2}$-norm, see section 5.2) that leads to the selection of a sparse set of transcripts that are jointly expressed across several samples.

3.3.3 Network flow optimization

Network flow optimization is a special class of constrained optimization problems for which dedicated algorithms exploiting the network structure have proven to be efficient (Ahuja et al., 1993). In this section, we provide some definitions before describing two standard network flow optimization problems relevant in the context of transcript reconstruction and we finally state an important theorem that will help us to understand why transcripts can be estimated within the framework of network flow optimization.

Pioneer works that bring together network flow optimization with sparsity-inducing penalization include Mairal et al. (2011) and Mairal and Yu (2013). Inspired from this literature in chapter

4, we develop a procedure to infer the transcript isoforms from RNA-seq data that combines the efficiency of network flow optimization techniques and a control for the size of the solutions.

Definitions

A *network* is a tuple $N = (G, s, t, b)$ where $G = (V, E)$ is a directed graph with V a set of vertices, $E \in V \times V$ a set of directed edges, s a particular vertex in V called *source* and t a vertex in V called *sink* such that there is no edge coming to s and no edge leaving from t, and $b : E \to \mathbb{R}+$ is a function assigning a *capacity* b_{uv} to every arc $(u, v) \in E$.

We say that a function $f : E \to \mathbb{R}_+$ assigning to every arc $(u, v) \in E$ a non-negative number f_{uv} is a *flow* over the network N if the following two linear constraints are satisfied:

1. **capacity constraint**:
$$\forall (u, v) \in E, \quad 0 \leq f_{uv} \leq b_{uv} \ ,$$

2. **conservation constraint** (incoming flow is equal to outgoing flow except for the source and the sink):
$$\forall v \in V \backslash \{s, t\}, \quad \sum_{u \in V} f_{uv} - \sum_{u \in V} f_{vu} = 0 \ .$$

Standard network flow problems

1. **maximum flow problem**. The value of a flow, often denoted as $|f|$, is the amount of flow $\sum_{v \in V;(s,v) \in E} f_{sv}$ outgoing from s, which is equal to the amount of flow $\sum_{v \in V;(v,t) \in E} f_{vt}$ incoming to t.

   A classical problem in network flow optimization is the *maximum-flow* problem (Ford and Fulkerson, 1956), which consists of finding a flow f of maximum value $|f|$ in the network N.

   The maximum-flow problem is a linear program, which can be solved efficiently with combinatorial algorithms that exploit the structure of the problem. The "push-relabel" algorithm introduced by Goldberg and Tarjan (1988) for instance solves the maximum-flow problem in strong[12] polynomial time with $O(|V|^2|E|)$ complexity. Other versions of the push-relabel algorithm have a $O(|V||E| \log(|V|^2/|E|)$ complexity (King et al., 1994).

[12] a problem can be solved in strong polynomial time when an exact solution can be obtained in a finite number of steps that is polynomial in $|V|$ and $|E|$

Among the methods reported in section 3.3.1 using network flow optimization, StringTie (Pertea et al., 2015) estimates the abundances of the transcripts by iteratively solving a maximum-flow problem.

2. **minimum-cost flow problem**. In a *minimum-cost flow* problem, one is additionally given flow cost functions $c_{uv} : \mathbb{R} \to \mathbb{R}$ on every arc $(u, v) \in E$, and is required to find a flow f which minimizes:

$$\sum_{(u,v)\in E} c_{uv}(f_{uv})$$

It is well-known that under the assumption that the flow cost functions $c_{uv}(.)$ are convex, a minimum-cost flow can be found in polynomial time. For instance the "cost scaling algorithm" (Goldberg and Tarjan, 1990) generalizes the push-relabel one to the minimum-cost flow problem with a $O(|V|^2|E|\log(|V|))$ complexity.

Both Traph (Tomescu et al., 2013) and FlipFlop (Bernard et al., 2014) solve a minimum-cost flow problem in the context of transcript recovery. The cost function they use are however different: Traph uses quadratic costs while FlipFlop uses costs derived from the Poisson likelihood modeling read counts (see section 3.2.2). In both cases, one of the key feature to be able to map the transcript inference task into a flow problem is the fact that the underlying loss function is *separable*, that is it corresponds to the sum of costs distributed on the nodes of an underlying graph model. In addition, FlipFlop incorporates the ℓ_1-penalization into the minimum-cost flow problem, as explained in detail in chapter 4.

Another key property to make use of minimum-cost flow techniques is the fact that a flow can be decomposed into a set of paths. We briefly describe this property below and give more insights in chapter 4.

From a flow to a set of paths

A *(s,t)-path flow* is defined as a flow vector carrying the same non-negative value on every arc of a path between the source s and the sink t. Intuitively, a (s,t)-path flow corresponds to sending a given quantity along a path from s to t.

The *flow decomposition* theorem (see Ahuja et al., 1993, theorem 3.5) says that every flow f in a DAG[13] can be decomposed into a collection of at most $|E|$ (s,t)-path flows. The converse

[13] when the graph is directed but not acyclic the flow decomposition might also involve cycles.

is also true, that is summing a set of (s,t)-path flows leads to a valid flow f distributed on the edges of the graph. Hence there is a strict equivalence between attributing a value f_{uv} of a flow on every arc (u, v) of the graph and looking at the quantity that should circulate on every (s,t)-path.

We now anticipate what will be made explicit in chapter 4: instead of working with the abundances of transcripts it is equivalent to work with flows on an appropriate DAG describing the structure of the problem.

Chapter 4

Efficient transcript isoform identification and quantification from RNA-seq data with network flows

The chapitre introduit une nouvelle méthode de détection et de quantification des transcrits alternatifs à partir de données RNA-seq. Plusieurs méthodes existantes font appel à des régressions pénalisées par la norme ℓ_1. Cependant, elles souffrent d'intractabilité algorithmique et ne peuvent considérer un grand nombre de transcrits candidates. Nous montrons qu'il est possible de résoudre le problème de sélection de transcrits via la pénalité ℓ_1 de façon exacte et efficace grâce à des techniques d'optimisation de flots.

In this chapter, we present a new method to detect and quantify transcript isoforms from RNA-seq data. Several state-of-the-art methods for isoform identification and quantification are based on ℓ_1-penalized regression. However, they need to explicitly enumerate the set of candidate transcripts, which becomes intractable for genes with many exons. For this reason, existing approaches using the ℓ_1-penalty are either restricted to genes with few exons, or only run the regression algorithm on a small set of pre-selected isoforms. We show how to efficiently tackle the sparse estimation problem on the full set of candidate isoforms by using network flow optimization. Our technique removes the need of a preselection step, leading to better isoform identification while keeping a low computational cost. In addition, we provide an open-source R package that implements our method, see section C.

Note that the material of this chapter is based on the following publication:

E. Bernard, L. Jacob, J. Mairal and J.-P. Vert. Efficient RNA isoform identification and quantification from RNA-seq data with network flows. *Bioinformatics*, 30(17):2447-2455, 2014.

4.1 Background and related works

As previously motivated in chapter 2, the identification and quantification of transcript isoforms present in a sample is of outmost interest for various reasons, from both developmental biology or clinical point of views. Alternative transcripts can be translated in proteins with potentially different or even opposite functions (David and Manley, 2010) so that the detection of isoforms whose presence or quantity varies between samples may lead to new biomarkers or clinical targets (Pal et al., 2012), and highlights novel biological processess invisible at the gene level.

In chapter 3, we emphasized that RNA-seq technologies facilitate the study of alternatively spliced genes. Next-generation RNA sequencing is indeed well suited to transcript quantification as the read count density observed along the different exons of a gene provide information on which alternatively spliced mRNAs are expressed in a sample, and in which proportions. Since the read length is typically smaller than the mRNA molecule of a transcript, identifying and quantifying the transcripts is however difficult: an observed read mapping to a particular exon may come from an mRNA molecule of any transcript containing this exon. Some methods consider that the set of expressed transcript isoforms (see *e.g* Jiang and Wong (2009) and section 3.2.3) is known in advance, in which case the problem is to estimate their expression levels. However, little is known in practice about the possible isoforms of genes, and restricting oneself to isoforms that have been described in the literature may lead to missing new ones.

Two main paradigms have been used to estimate expression at the transcript level from mapped RNA-seq reads while allowing de novo transcript discovery (see table 3.1 for a detailed review of the existing methods). On the one hand, the Cufflinks software package (Trapnell et al., 2010) proceeds in two separate steps to identify expressed isoforms and estimate their abundances. It first estimates the list of alternatively spliced transcripts by building a small set of isoforms containing all observed exons and exon junctions. In a second step, the expression of each transcript is quantified by likelihood maximization given the list of transcripts. Identification and quantification are therefore done independently. On the other hand, a second family of methods (Xia et al., 2011; Li et al., 2011b; Bohnert and Rätsch, 2010; Li et al., 2011a; Mezlini et al., 2013; Behr et al., 2013) jointly estimates the set of transcripts and their expression using a penalized likelihood approach. These methods model the likelihood of the expression of all possible transcripts, possibly after some pre-selection, and the penalty encourages sparse solutions that have a few expressed transcripts.

The two-step approach of Cufflinks (Trapnell et al., 2010) is reasonably fast, but does not exploit the coverage density along the gene, which can be a valuable information to identify the set of transcripts. This is indeed a conclusion drawn experimentally using methods from the second paradigm (see Xia et al., 2011; Li et al., 2011b; Bohnert and Rätsch, 2010; Li et al., 2011a; Mezlini et al., 2013). To summarize, the first paradigm is fast but can be less statistically powerful than the second one in some cases, while the second paradigm suffers from the exponential number of candidate isoforms and becomes intractable for genes with many exons.

The contribution of the work presented in this chapter is to allow ℓ_1-penalized regression methods from the second family to run efficiently without pre-filtering the set of isoform candidates, although they solve a non-smooth optimization problem over an exponential number of variables. To do so, we show that the penalized likelihood maximization can be reformulated as a convex cost network flow problem, which can be solved efficiently (Ahuja et al., 1993; Bertsekas, 1998; Mairal and Yu, 2013).

The rest of the chapter is organized as follows. Section 4.2 introduces the statistical model (section 4.2.1) and the penalized likelihood approach (section 4.2.2) we use. Our model is similar to the one used by Xia et al. (2011), but properly deals with reads that cover more than two exons, effectively taking advantage of longer reads. We then reformulate the model as a path selection problem over a particular graph (section 4.2.3), and present in sections 4.2.4-4.2.6 our method called FlipFlop (Fast Lasso-based Isoform Prediction as a FLOw Problem) for solving it efficiently. Section 4.3 empirically compares our approach with the state-of-the-art on simulated and real sequencing data. Our experiments show that our approach has higher accuracy in isoform discovery than methods which treat discovery and abundance estimation as two separate steps, and that it runs significantly faster than methods explicitly listing the candidate isoforms. We discuss the implications of our results in section 4.4.

4.2 Proposed approach

Our approach to transcript isoform deconvolution from RNA-seq data consists of fitting a sparse probabilistic model, like several existing methods including rQuant (Bohnert and Rätsch, 2010), NSMAP (Xia et al., 2011), IsoLasso (Li et al., 2011b), SLIDE (Li et al., 2011a) or iReckon (Mezlini et al., 2013). The read counts from RNA-seq data are modeled as a linear combination of isoforms expressions that are estimated by using the maximum likelihood principle. Because

the number of candidate isoforms grows exponentially with the number of exons, the above methods are either computationally expensive for genes with many exons (such as NSMAP or SLIDE), or include a preselection step to reduce the number of candidates, which may alter the method accuracy.

The main novelty of our approach is to tackle the sparse estimation problem efficiently *without pre-filtering*. In the methodological section, we show that the corresponding penalized maximum likelihood estimator can be computed in polynomial time with the number of exons despite the exponential number of candidate transcripts. The key is the use of a non-trivial optimization technique based on the concept of flow in a graph (Ahuja et al., 1993; Mairal and Yu, 2013).

4.2.1 Statistical model

We consider an extension of the uniform sampling model originally introduced by Jiang and Wong (2009), also used in NSMAP and previously described in section 3.2.2. to discover the expressed transcripts and estimate their expression levels. Note that we use here the same notations as the ones introduced in section 3.2.2, however, for the sake of clarity we recall below some of them.

Given a gene of interest, we assume that the list of its exons is known, and that the reads of the RNA-seq experiments have been mapped to a reference genome. In practice, an exon can either by constructed from the read alignment as a cluster of reads delineated with junction reads, or can be defined *a priori* from available gene annotation such as the one provided by the UCSC genome browser[1]. In the latter case, exons with alternative 5' donor and 3' acceptor sites are split in two separate exons.

We define a *bin* as an ordered set of exons. Each read is assigned to a unique bin, corresponding to the exact set of exons that it overlaps. The set of possible bins is denoted by V. Our model can involve bins with more than two exons. It is thus more general than the one of NSMAP, where bins are simply exons and exon-exon junctions. This extension of NSMAP is particularly useful for long reads, which often cover more than two exons. We summarize the read information by the counts y_v of reads falling into each bin $v \in V$.

We consider in our model all $\mathcal{P}$ possible candidate isoforms consisting of an ordered sequence of exons. Each candidate isoform also corresponds to a unique sequence of bins. This sequence

[1] http://genome.ucsc.edu/

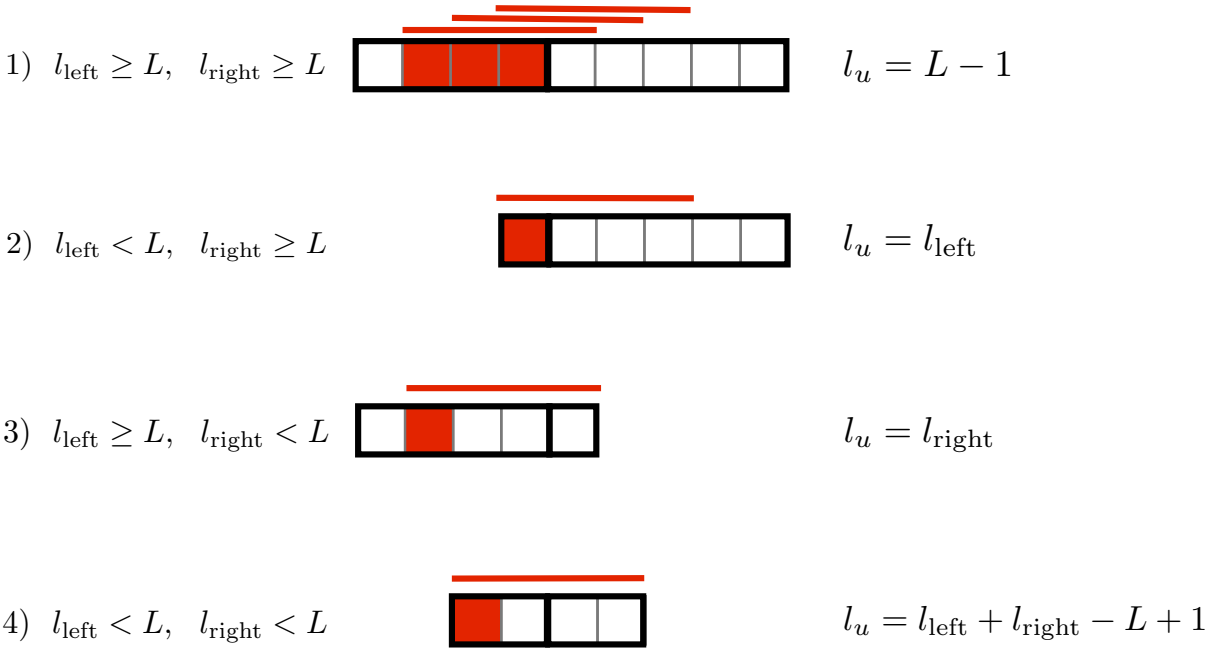


FIGURE 4.1: Computation of the effective length l_v. A bin v is composed of two exons of lengths l_{left} and l_{right}, drawn in solid black line. Red lines represent the reads of length L. The red squares correspond to the positions where a read can start and be assigned to the bin (a read is assigned to a bin if it overlaps all the exons of the bin and is contained in it). There are four possible cases depending of the relative order of the lengths l_{left}, l_{right} and L: when both l_{left} and l_{right} are bigger than L, the effective length only depends of the read length ($l_v = L-1$), when only one of the exons is strictly smaller than the read length then the effective length equal the length of that exon ($l_v = l_{\text{left}}$ or $l_v = l_{\text{right}}$), and when both exons are strictly smaller than L, the effective length is equal to $l_{\text{left}} + l_{\text{right}} - L + 1$. These four cases for a multi-exon bin can be written in a single formula: $l_v = \min(l_{\text{left}}, L-1) + \min(l_{\text{right}}, L-1) - L + 1$. Note that when a bin is composed of more than two exons, the reasoning is the same by replacing the read length L by $L - l_{\text{int}}$ where l_{int} is the total length of the internal exons of the bin.

can be generated by virtually moving a read along the candidate isoform, and recording the sets of exons that it successively overlaps.

The *effective length* l_v of a bin $v \in V$ is defined as the number of positions in the candidate isoform where reads can start and be assigned to the bin. A simple computation shows that for a bin involving a single exon of length l_e, we have $l_v = l_e - L + 1$, where L is the read length, while for bins involving several exons, $l_v = \min(l_{\text{left}}, L - l_{\text{int}} - 1) + \min(l_{\text{right}}, L - l_{\text{int}} - 1) - L + l_{\text{int}} + 1$, where l_{left} and l_{right} are the lengths of the leftmost and rightmost exons of the bin, and l_{int} is the total length of the internal exons of the bin. Interestingly, we note that the effective length of a bin does not depend on the candidate isoform it is associated with. Figure 4.1 shows how to compute the effective length.

We model read counts as independent Poisson random variables whose means are proportional to the bin effective lengths and to the total abundances of isoforms associated to each bin.

More formally, let us denote by $\beta_p \in \mathbb{R}_+$ the abundance of isoform $p \in \mathcal{P}$. We recall that β_p represents the expected number of reads per base in isoform p. Thus, $\sum_{p\in\mathcal{P}:p\ni v} \beta_p$ represents the sum of expressions of all isoforms involving bin v. We expect the observed count for bin v to be distributed around this value times the effective length l_v of the bin, and therefore model the read count Y_v as a Poisson random variable with parameter $p_v = l_v \sum_{p\in\mathcal{P}:p\ni v} \beta_p$.

For a vector $\boldsymbol{\beta} = [\beta_p]_{p\in\mathcal{P}}$ in $\mathbb{R}_+^{|\mathcal{P}|}$ this yields the negative log-likelihood

$$\mathcal{L}(\boldsymbol{\beta}) = \sum_{v\in V} \left[p_v - y_v \log p_v + \log(y_v!)\right], \tag{4.1}$$

where the scalars p_v depend linearly on $\boldsymbol{\beta}$.

Maximizing the likelihood (4.1) allows one to quantify the relative abundance of each transcript when the model only includes the list of "true" isoforms present in the sample (Jiang and Wong, 2009). Since this list is unknown a priori, we present in the next section the sparse estimation approach that can jointly quantify and identify the transcripts using all candidate isoforms, following Xia et al. (2011).

4.2.2 Isoform detection by sparse estimation

Since we do not assume that the list $\mathcal{P}$ of expressed isoforms –*i.e.*, such that $\beta_p \neq 0$– is known in advance, we endow $\boldsymbol{\beta}$ with an exponential prior $\beta_p \overset{\text{iid}}{\sim} E(\lambda)$ and maximize over all candidate isoforms the resulting posterior likelihood, leading to the estimator

$$\hat{\boldsymbol{\beta}}_\lambda = \underset{\boldsymbol{\beta}\in\mathbb{R}_+^{|\mathcal{P}|}}{\arg\min} [\mathcal{L}(\boldsymbol{\beta}) + \lambda \| \boldsymbol{\beta} \|_1] \ , \tag{4.2}$$

where λ is a regularization parameter, and the ℓ_1-norm is defined as $\| \boldsymbol{\beta} \|_1 = \sum_{p\in\mathcal{P}} |\beta_p|$. As previously described in section 3.3.2, it is well-known that the ℓ_1-norm penalty has a sparsity-inducing effect — that is, lead to estimators $\hat{\boldsymbol{\beta}}_\lambda$ that contain many zeroes (Tibshirani, 1996). The parameter λ controls the number of non-zero elements in the solution $\hat{\boldsymbol{\beta}}_\lambda$, *i.e.*, of selected isoforms, with larger λ corresponding to fewer isoforms.

Note that (6.1) is better adapted to long reads than the original formulation of NSMAP (Xia et al., 2011) thanks to the use of general bins. rQuant (Bohnert and Rätsch, 2010), IsoLasso (Li et al., 2011b), and SLIDE (Li et al., 2011a) solve a similar problem where the likelihood is a simpler quadratic function, corresponding to a Gaussian model for the read counts. A difficulty

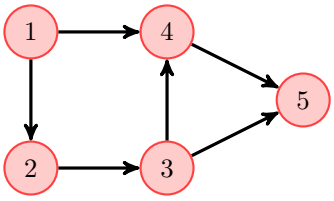
with these approaches is that the dimension $|\mathcal{P}|$ grows exponentially in $|V|$ making (6.1) intractable when $|V|$ is large. For example, Li et al. (2011a) restrict themselves to experiments involving genes with less than 10 exons, due to the high computational cost for larger genes. Xia et al. (2011) restrict themselves to genes with less than 80 exons, but only consider candidates with pairs of transcription start and polyadenylation sites already observed in annotations. Other approaches such as IsoLasso include a filtering step to reduce the number of isoforms. As pointed out in section 4.1, this filtering may lead to a loss of power in isoform detection, because it disregards the read density information when constructing the set of candidates. In the next section, we show that, surprisingly, problem (6.1) can be solved efficiently without pre-filtering the isoforms by using network flow algorithms.

4.2.3 Isoform detection as a path selection problem

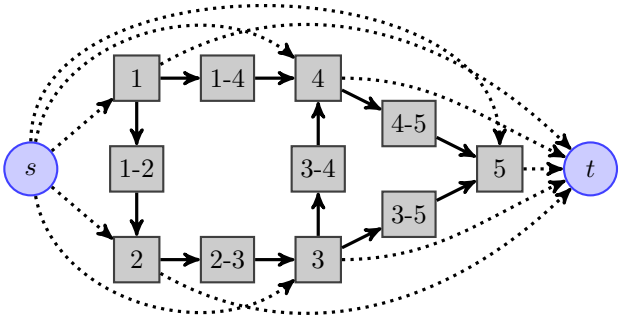
In this section, we reformulate the isoform detection problem as a path selection problem over a graph model. Remember that a graph $G = (V, E)$ is composed of a finite set of vertices V and edges $E \subseteq V \times V$. A path is a sequence of vertices $v_1, \ldots, v_k \in V$ such that (v_i, v_{i+1}) is an arc in E for all indices $1 \leq i < k$. A graph is a DAG if it contains no path $(v_1, \ldots, v_k)$ with $v_1 = v_k$. In other words, the graph does not contain any cycle.

We construct an oriented graph $G = (V, E)$ whose vertices are the bins with positive effective length defined in section 4.2.1 — each corresponding to an ordered set of exons. An edge connects bin u to v if v can be obtained from u by removing the first exon of its ordered set or by adding one extra exon at the end of the ordered set, depending on the lengths of the exons composing the bin (see figure 5.1). We call *starting bins* (respectively *stopping bins*) the bins that can contain a read at the left-most (respectively right-most) position of an isoform. The resulting graph is a DAG generalizing the splicing graph (Heber et al., 2002), whose vertices are single exons and edges are exon-exon junctions.

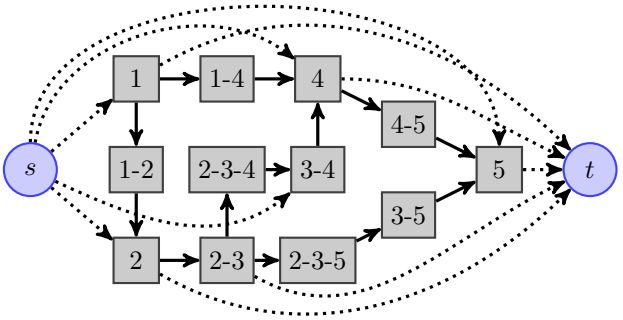
We also consider in the set V two additional vertices s and t respectively dubbed *source* and *sink*, that connect to all starting (resp. stopping) bins. We do not impose any restriction on the set of transcription starting sites and polyadenylation sites and each exon can potentially start or end an isoform. Consequently, the source s is connected to all bins modeling an exon start, and the sink t to all bins modeling an exon end. Hence the set of edges E also contains all edges of the form (s, v) where $v \in V$ is a starting bin, and (t, v) where $v \in V$ is a stopping bin. This



(a) Splicing graph for a gene with 5 exons.



(b) Graph G when all exons are bigger than the read length.



(c) Graph G when the length of exon 3 is smaller than the read length.

FIGURE 4.2: Illustration of the graph construction for a gene with 5 exons. The original splicing graph is represented in (a). The 5 exons are represented as vertices and an arrow between two vertices indicates a junction. The nodes of graph G in (b) and (c) are bins with positive effective length denoted by gray square, as well as source s and sink t represented as circles. G in (b) is the resulting graph when all exons are bigger than the read length. In that case, each bin either corresponds to a unique exon, or to a junction between two exons. G in (c) is the resulting graph when the length of exon 3 is smaller than the read length. Some bins involve then more than two exons, here bins (2-3-4) and (2-3-5). The source links all possible starting bins and conversely all possible stopping bins are linked to the sink. There is a one-to-one correspondence between (s,t)-paths in G (paths starting at s and ending at t) and isoform candidates. For example, the path $(s, 1, 1\text{-}4, 4, 4\text{-}5, 5, t)$ corresponds to isoform 1-4-5.

graph construction is illustrated in figure 5.1. Montgomery et al. (2010) use a similar graph structure in the context of estimating the expression of a set of known annotated transcripts.

By construction, one can check that the set of paths in G starting from s and ending at t (such paths are called (s,t)-paths) is in bijection with the set of candidate isoforms $\mathcal{P}$. Based on this one-to-one mapping, we can reformulate the penalized maximum likelihood problem (4.1)-(6.1) as follows: we want to find non-negative weights β_p for each path $p \in \mathcal{P}$ which minimize:

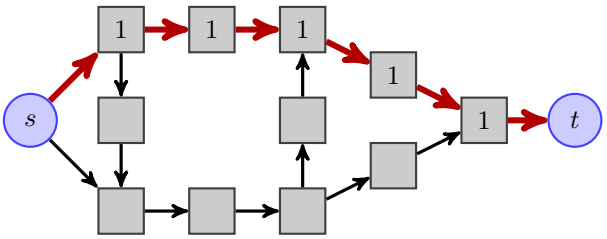
$$\sum_{v \in V} [p_v - y_v \log p_v] + \lambda \sum_{p \in \mathcal{P}} \beta_p \quad \text{with} \quad p_v = \Big(l_v \sum_{p \in \mathcal{P}: p \ni v} \beta_p \Big), \tag{4.3}$$

where the sum $\sum_{p \in \mathcal{P}} \beta_p$ is equal to the ℓ_1-norm $\|\beta\|_1$ since the entries of $\boldsymbol{\beta}$ are non-negative. Note that we have removed the constant term $\log(y_v!)$ from the log-likelihood since it does not play a role in the optimization. This reformulation is therefore a path selection (finding which β_p are non-zero) and quantification problem over G. The next section shows how (4.3) can further be written as a flow problem, *i.e.*, technically a constrained optimization problem over the edges of the graph rather than the set of paths in $\mathcal{P}$. A computationally feasible approach can then be devised to solve (4.3) efficiently, following Mairal and Yu (2013).

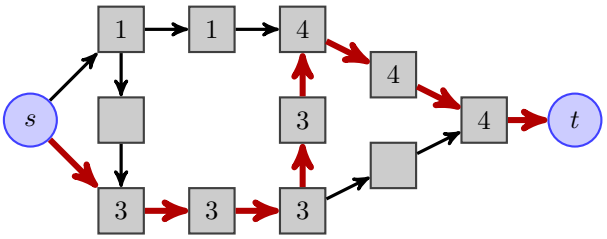
4.2.4 Optimization with network flows

The basics of network flows have been introduced in section 3.3.3. We recall here some definitions that are particularly useful to understand our optimization problem. A *flow* f on G is defined as a non-negative function on arcs $[f_{uv}]_{(u,v) \in E}$ that satisfies conservation constraints: the sum of incoming flow at a vertex is equal to the sum of outgoing flow except for the source s and the sink t. Such conservation property leads to a physical interpretation about flows as quantities circulating in the network, for instance, water in a pipe network or electrons in a circuit board. The source node s injects into the network some units of flow, which move along the arcs before reaching the sink t.

For example, given a path $p \in \mathcal{P}$ and a non-negative number β_p, we can make a flow by setting $f_{uv} = \beta_p$ when u and v are two consecutive vertices along the path p, and $f_{uv} = 0$ otherwise. This construction corresponds to sending β_p units of flows from s to t along the path p. Such simple flows are called (s,t)-*path flows*. More interestingly, if we have a set of non-negative weights $\boldsymbol{\beta} \in \mathbb{R}_+^{|\mathcal{P}|}$ associated to all paths in $\mathcal{P}$, then we can form a more complex flow by



(a) Reads at every node corresponding to one isoform.



(b) Reads at every node after adding another isoform.

FIGURE 4.3: Flow interpretation of isoforms using the same graph as in figure 5.1(b). For the sake of clarity, some edges connecting s and t to internal nodes are not represented, and the length of the different bins are assumed to be equal. In (a), one unit of flow is carried along the path in red, corresponding to an isoform with abundance 1. In (b), another isoform with abundance 3 is added, yielding additional read counts at every node.

superimposing all (s,t)-path flows according to

$$f_{uv} = \sum_{p \in \mathcal{P} : p \ni (u,v)} \beta_p \ , \tag{4.4}$$

where $(u,v) \in p$ means that u and v are consecutive nodes on p.

While (4.4) shows how to make a complex flow from simple ones, a converse exists, known as the *flow decomposition theorem* (see, e.g., Ahuja et al., 1993). It says that for any DAG, every flow vector can always be decomposed into a sum of (s,t)-path flows. In other words, given a flow $[f_{uv}]_{(u,v)\in E}$, there exists a vector $\boldsymbol{\beta}$ in $\mathbb{R}_+^{|\mathcal{P}|}$ such that (4.4) holds. Moreover, there exists linear-time algorithms to perform this decomposition (Ahuja et al., 1993). As illustrated in figure 4.3, this leads to a flow interpretation for isoforms.

We now have all the tools in hand to turn (4.3) into a flow problem by following Mairal and Yu (2013). Given a flow $f = [f_{uv}]_{(u,v)\in E}$, let us define the amount of flow incoming to a node v in V as $f_v \triangleq \sum_{u \in V : (u,v) \in E} f_{uv}$. Given a vector $\boldsymbol{\beta} \in \mathbb{R}_+^{|\mathcal{P}|}$ associated to f by the flow decomposition theorem, *i.e.*, such that (4.4) holds, we remark that $f_v = \sum_{p \in \mathcal{P} : p \ni v} \beta_p$ and that $f_t = \sum_{p \in \mathcal{P}} \beta_p$.

Therefore, problem (4.3) can be equivalently rewritten as:

$$\min_{f \in \mathcal{F}} \sum_{v \in V} [p_v - y_v \log p_v] + \lambda f_t \quad \text{with} \quad p_v = l_v f_v \ . \tag{4.5}$$

where $\mathcal{F}$ denotes the set of possible flows. Once a solution $f^\star$ of (4.5) is found, a solution $\boldsymbol{\beta}^\star$ of (4.3) can be recovered by decomposing $f^\star$ into (s,t)-path flows, as discussed in the next section.

The use of network flows has two consequences. First, (4.5) involves a polynomial number of variables, as many as arcs in the graph, whereas this number was exponential in (4.3). Second, problem (4.5) falls into the class of *convex cost flow* problems (Ahuja et al., 1993), for which efficient algorithms exist.[2] In our experiments, we implemented a variant of the scaling push-relabel algorithm (Goldberg, 1997), which also appears under the name of ε-relaxation method (Bertsekas, 1998). Note that the approach can be generalized to any concave likelihood function, including the Gaussian model used by IsoLasso and SLIDE.

Network flows have been used in several occasions in bioinformatics. Medvedev and Brudno (2009) solve a convex cost flow problem on a bidirected de Bruijn graph for maximum likelihood whole genome shotgun assembly. Montgomery et al. (2010) introduced the formalism of flows for RNA-seq data; however they did not perform isoform discovery but quantification from a set of known transcripts. Their formulation is a linear program, the dimension of which is the number of candidate transcripts considered, which is not a network flow problem. Singh et al. (2011) uses the terminology of flows for RNA-seq data in the context of testing differential transcription without reconstructing transcripts. Finally, Tomescu et al. (2013) describe a similar method in spirit that the one we just presented above. They also uses minimum cost flow techniques for isoform recovery. However, their method only involves bins corresponding to exons and exon-exon junction, and, more importantly, does not solve the *penalized* likelihood approach. They have therefore no principled way to balance the sparsity of the solution with its likelihood, and even mention that this leads to a NP-hard problem. To our knowledge, our work is the first to show that the sparsity-inducing ℓ_1 penalty can be integrated with the likelihood term in the language of network flow, in order to estimate a flow with large likelihood that can be easily decomposed in a number of paths as small as we wish.

[2] The function (4.5) can be decomposed into costs $C_v(f_v)$ over vertices v. The general convex cost flow objective function is usually presented as a sum of costs $C_{uv}(f_{uv})$ over arcs (u,v). It is however easy to show that costs over vertices can be reduced to costs over arcs by a simple network transformation (see Ahuja et al., 1993, section 2.4). Note that all arcs have zero lower capacities and infinite upper capacities.

4.2.5 Flow decomposition

We have seen that after solving (4.5) we need to decompose $f^\star$ into (s,t)-path flows to obtain a solution $\boldsymbol{\beta}^\star$ of (6.1). As illustrated in figure 4.3, this corresponds to finding the two isoforms from 4.3(b). Whereas the decomposition might not be ambiguous when $f^\star$ is a sum of few (s,t)-path flows, it is not unique in general. Our approach to flow decomposition consists of finding an (s,t)-path carrying the maximum amount of flow (equivalently finding an isoform with maximum expression), removing its contribution from the flow, and repeating until convergence. We remark that finding (s,t)-path flows according to this criterion can be done efficiently using dynamic programming, similarly as for finding a shortest path in a directed acyclic graph (Ahuja et al., 1993). We insist on the fact that the flow decomposition returns one solution of the ℓ_1-penalized estimator given by problem 4.3. This problem can have several solutions yielding the same objective value, and which are typically sparse in the number of transcripts. The non-uniqueness of the solution is not an artifact of our network flow approach, but a property of the ℓ_1-penalized estimator. Algorithms such as SLIDE, NSMAP, or others that explicitly enumerate the candidates and minimize the parameter in the candidate space also return one of several solutions.

4.2.6 Model selection

The last problem we need to solve is model selection: even if we know how to solve (6.1) efficiently, we need to choose a regularization parameter λ. For large values of λ, (6.1) yields solutions involving few expressed isoforms. As we decrease λ, more isoforms have a non-zero estimated expression β_p, leading to a better data fit but also leading to a more complex model. A classical way of balancing fit and model complexity is to use likelihood ratio tests. Xia et al. (2011) chose this approach, but we found the log likelihood ratio statistics to be empirically poorly calibrated due to the typically small number of samples units — exons — and the non-independence of the observed read counts. We choose a related approach, which we found better behaved, and select the model having the largest BIC criterion (Schwarz, 1978). An alternative approach taken by Li et al. (2011a) would be to use stability selection (Meinshausen and Bühlmann, 2010).

4.3 Experimental validation

We compare our proposed method FlipFlop to Cufflinks (Trapnell et al., 2010) version 2.0.0, IsoLasso (Li et al., 2011b) version 2.6.1 and NSMAP (Xia et al., 2011) on both simulated and real data. All experiments were run on a desktop computer on a single core of an Intel Xeon CPU X5460 3.16Ghz with 16Gb of RAM. Reads are aligned to a reference genome with TopHat (Trapnell et al., 2009) version 2.0.6, and the constructed alignment files are used as input to the methods we compare. IsoLasso, Cufflinks, and FlipFlop only use these aligned reads as input, and estimate their exon boundaries, transcription start sites (TSS) and polyadenylation sites (PAS) from read density. NSMAP additionally requires exon boundaries and known TSS/PAS as input. For paired-end experiments, we extended our initial model designed for single-end: a pair of reads is considered as a long single-end read. When the two reads of a pair span bins potentially separated by some exons, we use heuristics based on genomic distances to decide wether or not these exons are spliced. All softwares are used with default parameters, except that for paired-end experiments we provide fragment length mean and standard deviation to IsoLasso, Cufflinks and FlipFlop. Note that all results can be easily reproduced by following the tutorials available at `http://cbio.ensmp.fr/flipflop/experiments.html`.

4.3.1 Simulated human RNA-seq data

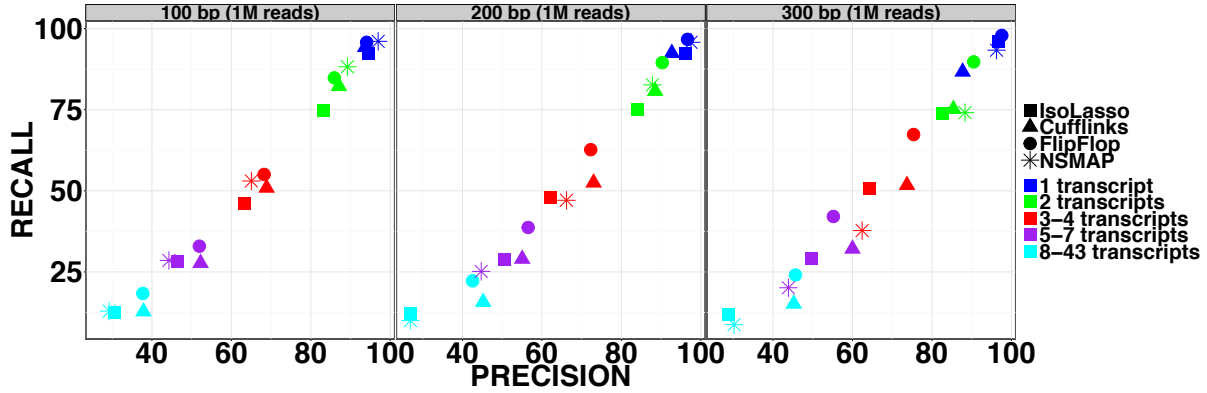
Since little is known about the true set of isoforms expressed in real data, we start our experimental validation with a set of simulations. We use the RNASeqReadSimulator software[3] to generate single-end and paired-end reads from the annotated human transcripts available in the UCSC genome browser (hg19). We restrict ourselves to the 1137 multi-exon genes on the positive strand of chromosome 1, corresponding to 3709 expressed transcripts.

We follow the protocol of Isolasso (Li et al., 2011b) and consider that a transcript from the annotation has been detected by a method if it predicts a transcript that (i) includes the same set of exons, and such that (ii) all internal boundary coordinates (*i.e.*, all the exon coordinates except the beginning of the first exon and the end of the last exon) are identical. The objective for each method is to recover a large proportion of transcripts that were used for read generation — high recall — without detecting too many transcripts that were not used to generate the reads — high precision.

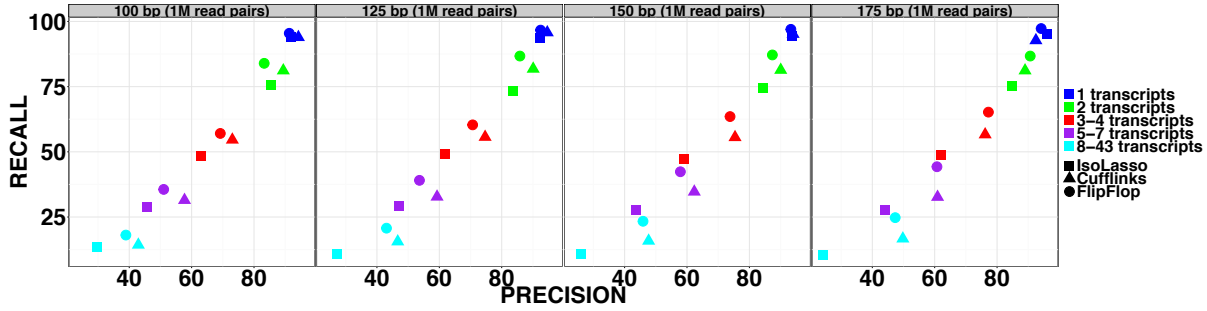
[3] `http://alumni.cs.ucr.edu/~liw/rnaseqreadsimulator.html`

Transcript number influence on isoform recovery

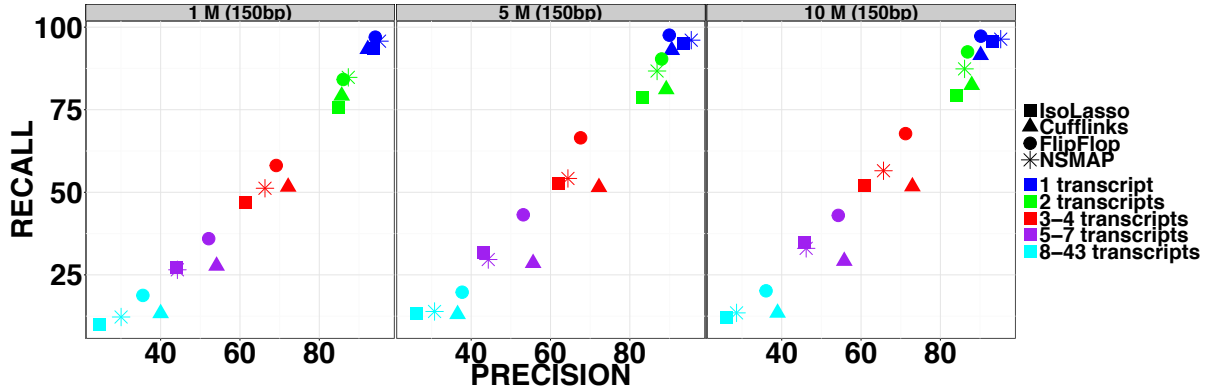
Figure 4.4 shows the precision and recall of the compared methods on a first set of single-end and paired-end simulations. Since we expect the difficulty of the deconvolution problem to increase with the number of transcripts of the gene, we stratify the result by this number: each dot represents the precision and recall of one method for genes with a particular number of transcripts in the UCSC annotation. As expected, genes with more transcripts lead to more difficult estimation problems and decreased performances for all methods. Figure 4.4(a) shows single-end results for different read lengths from 100bp to 300bp and a fixed number of 1 million reads per experiment. FlipFlop clearly takes advantage of longer reads: the longer the read the better the accuracy for all transcript levels. For 100bp long reads, FlipFlop and Cufflinks show similar results, while NSMAP gives slightly better precision and recall for 2 transcript level and degraded results compared to FlipFlop for more than 4 expressed transcripts. These differences might arise from the fact that NSMAP restricts its search to the TSS and PAS observed in the annotation whereas FlipFlop estimates them from reads, and the fact that the two methods use different graphs and model selection techniques. For 300bp long reads, FlipFlop outperforms all other methods as soon as there is more than one expressed transcript. For instance for the 3-4 transcripts level, FlipFlop achieves 75% of precision and 67% of recall, while Cufflinks reaches 74% and 52% and IsoLasso reaches 64% and 51%. This demonstrates that an adapted model for long reads is critical for isoform recovery. NSMAP optimizes a similar Poisson objective function as FlipFlop but only models reads at the exon or exon-exon junction levels; it looses statistical power when the read length increases. Figure 4.4(b) shows paired-end results for 400bp fragment length, 20bp standard deviation, 1 million read pairs, and read lengths from 100 to 175bp. Although our model is designed for single-end reads and is particularly adapted to long reads, it shows competitive or better accuracy for paired-end reads. Once again, when the read length increases, FlipFlop performance improves proportionally more than other methods. In figure 4.4(c), the read length is set to 150bp and the number of simulated reads varies from 1 million to 10 million. Increasing the coverage clearly helps FlipFlop whereas it does not change much for Cufflinks and IsoLasso. Indeed Cufflinks constructs its set of transcripts and estimates their abundances in two separate steps, and the construction of the set of returned transcripts does not take read density into account: it intends to find the smallest set of isoforms covering all the observed reads. IsoLasso is based on penalized likelihood maximization like FlipFlop and NSMAP, but starts from a restricted set of isoforms — the same set returned by Cufflinks



(a) Single-end reads with different lengths (100, 200, 300bp)



(b) Paired-end reads with different lengths (100, 125, 150, 175bp)



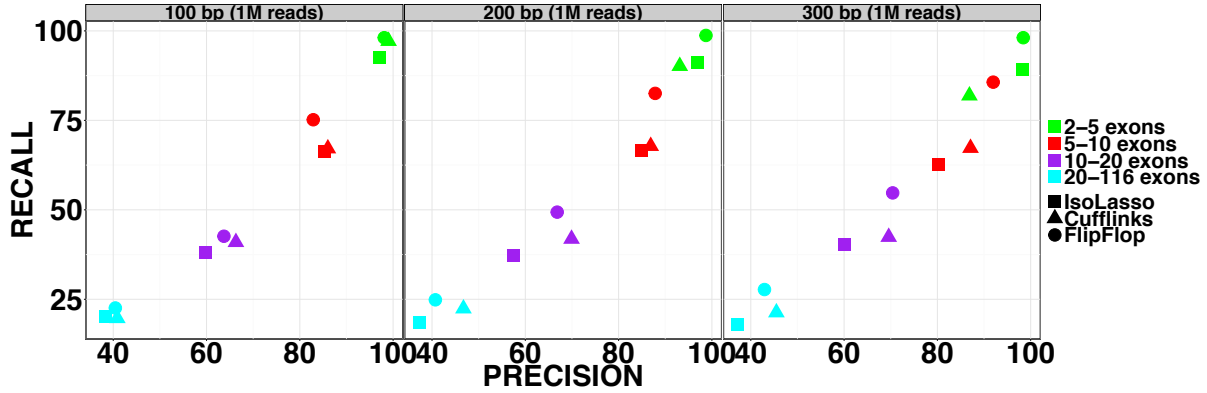
(c) Single-end reads with a fixed 150bp length and an increasing amout of material (1, 5, 10 million)

FIGURE 4.4: Precision and recall on simulated reads from the UCSC annotated human transcripts with a stratification based on the number of expressed transcripts.

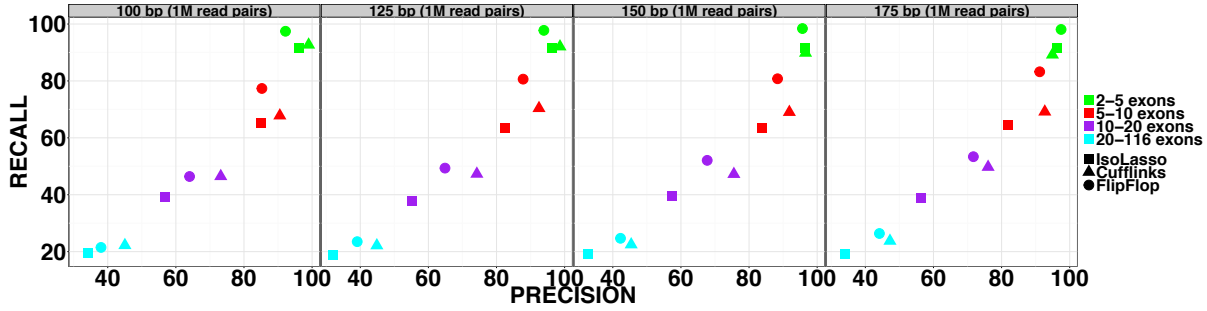
for single-end data. Consequently, this family of methods discards some information that can help identifying the set of expressed isoforms.

Gene size influence on isoform recovery

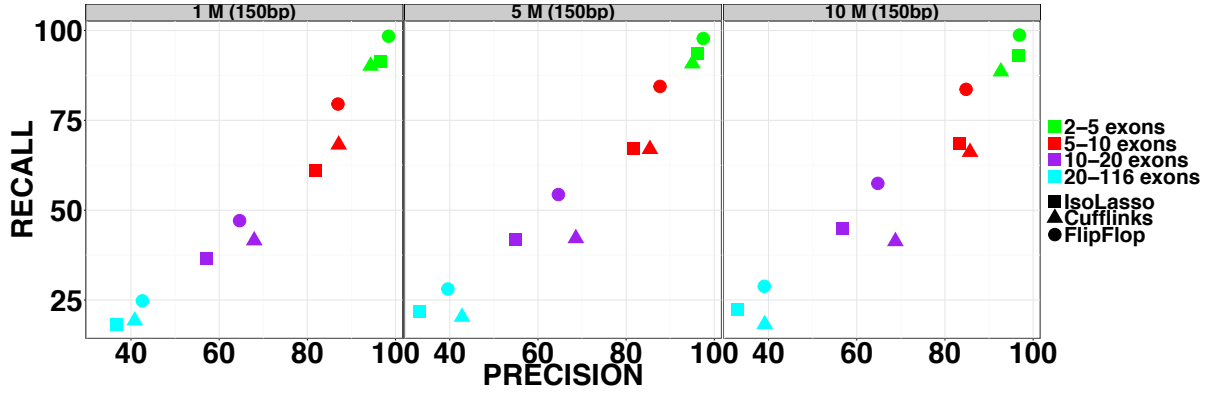
The number of exons of a gene is also a parameter that affects greatly the difficulty of the isoform deconvolution problem: indeed, the more exons the bigger the set of candidate transcripts.



(a) Single-end reads with different lengths (100, 200, 300bp) and 1 million reads by exon level



(b) Paired-end-end reads with different lengths (100, 125, 150, 175bp), 400bp mean fragment length and 1 million read pairs by exon level



(c) Single-end reads with a fixed 150bp length and an increasing amount of material (1, 5, 10 million)

FIGURE 4.5: Precision and recall on simulated reads from UCSC annotated human transcripts with an exon stratification.

Figure 4.5 shows similar experiments as the ones presented in figure 4.4 with a stratification by number of exons instead of number of transcripts. The number of exons varies from 2 to 116 and we compare here FlipFlop, Cufflinks and IsoLasso. For both single-end and paired-end reads, FlipFlop performance again increases greatly compared to Cufflinks and IsoLasso when the read length increases (figure 4.5(a) and figure 4.5(b)). For 300bp read length FlipFlop outperforms Cufflinks and IsoLasso for all genes with between 2 and 20 exons. Similarly to what we observed on simulations by transcript levels, and because FlipFlop predicts its transcripts by using both

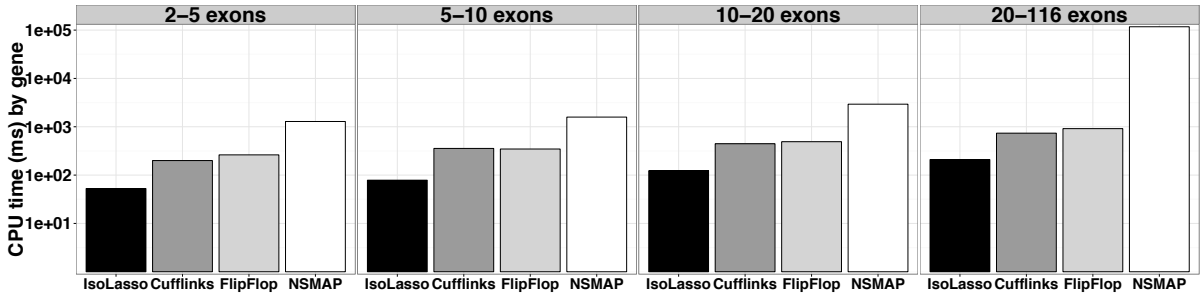


FIGURE 4.6: Average CPU times in milliseconds (logarithmic scale) for the compared methods to process a gene from human simulated 100bp single-end reads.

read alignment positions and read density without any filtering, an increase in coverage leads to better results for all exon levels (figure 4.5(c)).

Running times

Figure 4.6 shows the mean CPU time taken by each method to perform the deconvolution of genes with different sizes. Genes with more exons tend to have more candidate isoforms and experiments involving such genes are expected to take more time. Therefore, we stratify the observed times by exon number of the genes: each barplot represents the mean time taken by each method for finding the transcripts of genes having a particular number of exons. As expected, FlipFlop is always faster than NSMAP, more than a hundred times faster for genes with more than 20 exons. FlipFlop speed is comparable with Cufflinks, and about 4 times slower than IsoLasso. This is because IsoLasso maximizes its objective over a very restricted set of candidates — in these simulations never more than 9 and around 2-3 on average. Overall, FlipFlop estimates the set of expressed isoforms for 1137 genes in less than 9 minutes, *i.e.*, about 2 genes per second. Note also that the time for data pre-preprocessing (finding exon boundaries and read counts for exons and junctions) is taken into account for all methods except NSMAP.

More realistic simulations

We additionally perform more realistic simulations than the ones presented above using the Flux-Simulator (Griebel et al., 2012), a software designed to mimic *in silico* RNA-seq experiments workflow and to incorporate typical biases from library preparation and sequencing

We generate 2 million 150bp long single-end reads from the 4140 UCSC human transcripts of multi-exon genes of chromosome 1 and compare the results of Cufflinks and FlipFlop. We

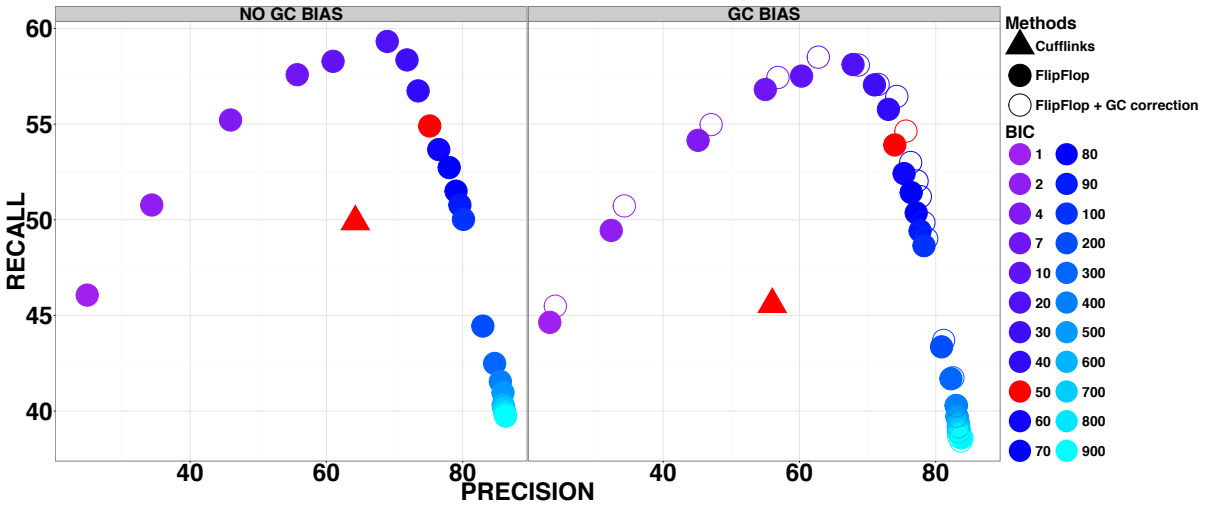


FIGURE 4.7: Precision and recall on simulated reads with FluxSimulator from 4140 UCSC human transcripts. Results obtained with default parameters are in red.

provided to Cufflinks the fragment length mean and standard deviation, while FlipFlop does not need that information for single-end experiments. Moreover we performed two kinds of simulations, with or without GC bias during the PCR amplification step. Precision and recall for Cufflinks and FlipFlop for the two experiments are shown in figure 4.7.

For both methods the inclusion of a GC bias affects the performance, but proportionally less for FlipFlop than for Cufflinks. Results with default parameters are shown in red, and for this particular set of experiments FlipFlop clearly outperforms Cufflinks both in precision and recall. We also show FlipFlop's results when applying a GC correction during the isoform recovery process. It simply corresponds to multiplying each Poisson parameter of each bin by the GC content of the bin. Using this correction slightly increases the accuracy of FlipFlop.

Finally we add FlipFlop's precision-recall curves, obtained when varying the BIC constant used for model selection (see section 4.2.6 on the model selection strategy). Surprisingly these curves have a bell shape: the recall increases first when the BIC constant decreases (light blue to dark blue colors) before to fall down for very small BIC constants. Using a small BIC constant corresponds to using a small regularization parameter λ in equation (2), and finally selecting a complex model with many isoforms. If the model is allowed to be very complex, several small isoforms are preferred to fewer long ones, and it might happen than some correct long isoforms are discarded from the solution. One way to deal with that problem in future work would be to penalize short isoforms by giving appropriate costs on the edges of the splicing graph.

Overall, these set of simulations confirm several facts. First, methods that identify and quantify

transcripts as a single penalized maximum likelihood problem show good performances and take clear advantage of an increase in coverage. Second, correctly modeling long reads leads to a great improvement of the accuracy of isoform reconstruction. Third, the proposed network flow solves the penalized likelihood approach quickly even when the set of candidate isoforms is extremely large.

4.3.2 Real RNA-Seq data

Our second round of experiments involves two independent real human embryonic stem cell data sets. They both contain about 50 million 75bp reads, either paired-end or single-end, with respectively NCBI SRA accession number SRR065504 and ERR361241.

In the experiments of section 4.3.1, we generated the reads based on a known set of transcripts. In the present case, the reads come from actual human tissues, and we do not have access to the true set of expressed transcripts. Following Xia et al. (2011) and Li et al. (2011a), we choose to use the UCSC annotation as ground truth in the evaluation. Admittedly, this is not perfect as some expressed transcripts may be missing from the annotation, and some annotated transcripts may not be expressed in this particular experiment. However, agreement of the prediction with the set of known transcripts could be a good sign.

Figure 4.8 shows precision and recall of each method for different FPKM levels. When considering all transcripts with predicted abundances higher than 1 FPKM, FlipFlop has a higher precision for both the paired-end and single-end data sets, while Cufflinks has a better recall. For transcripts with more than 5 FPKM abundance, all methods have a similar recall, with a slight advantage to Cufflinks, while FlipFlop shows a much better precision.

4.4 Conclusion

Simultaneously tackling identification and quantification using penalized likelihood maximization is known to be a powerful approach to estimate the set of expressed transcripts. However, existing ℓ_1-penalized regression techniques cannot deal with genes that contain too many exons as the set of candidate isoforms grows exponentially with the number $|V|$ of exons. By leveraging network flow optimization algorithms, we discover a few expressed transcripts among the

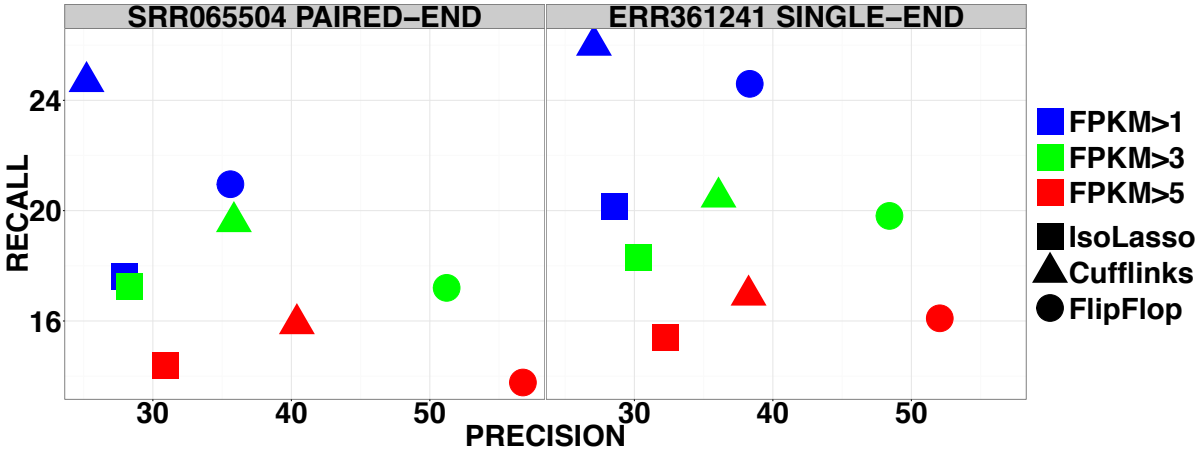


FIGURE 4.8: Precision and recall of compared methods on human embryonic stem cells data.

exponential number of candidates by solving a problem with a number of variables polynomial in $|V|$.

We compared our approach to existing ℓ_1-penalized likelihood maximization methods as well as methods that define expressed isoforms as the smallest set of transcripts covering all observed reads; the latter methods perform abundance estimation in a separate step. We observed on simulation data—where the true set of expressed transcripts is known—that, unlike the second set of methods, penalized likelihood maximization methods take advantage of an increase in read coverage. Moreover, we show that correctly modeling long reads is of primary importance for isoform recovery. Our approach, which models reads covering any number of exons, outperforms other methods for 300bp long reads. We believe this is an important improvement as RNA-seq technologies are moving forward longer reads. Our FlipFlop method has also shown to be competitive with state-of-the-art methods on real single-end and paired-end human stem cells data, especially for transcripts whose abundance was significant. In addition, the runtime of our method was comparable to the runtime of the second set of methods, and orders of magnitude faster than existing software for penalized likelihood maximization.

We believe these results have important practical implications. In addition to the obvious gain in time when estimating the expression of transcripts for a single gene and a single sample, our approach makes the task amenable in a reasonable amount of time for all genes in a large number of samples. Furthermore, accurately estimating the transcript level expression for all genes of all samples in a study may lead to improvements in molecular based diagnosis or prognosis tools, as well as in clustering of samples, *e.g* for cancer subtype discovery.

Chapter 5

A convex formulation for joint transcript isoform estimation from multiple RNA-seq samples

Ce chapitre propose de chercher les solutions au problème de déconvolution des isoformes de facon jointe pour plusieurs échantillons de données RNA-seq. L'hypothèse que plusieurs échantillons expriment des transcrits communs est formalisée par un problème d'optimisation convexe que nous proposons de résoudre de facon computationnellement efficace. Nous démontrons les bonnes performances de cette nouvelle approche sur des données simulées et réelles.

In this chapter, we propose a method for solving the isoform deconvolution problem jointly across multiple RNA-seq samples. We formulate and efficiently solve a convex optimization problem that leverages the hypotheses that some isoforms are likely to be present in several samples. We demonstrate the benefits of combining multiple samples on simulated and real data, and show that our approach outperforms pooling strategies and methods based on integer programming. Our multi-sample approach is implemented in an open-access R package, see section C.

Note that the material of this chapter is based on the following publication:

E. Bernard, L. Jacob, J. Mairal, E. Viara and J.-P. Vert. A convex formulation for joint RNA isoform detection and quantification from multiple RNA-seq samples. *BMC Bioinformatics*, i16:262, 2015.

5.1 Background and related works

As previously said in chapter 2 and recalled in chapter 4, alternative splicing is a regulated process that greatly increases the repertoire of proteins that can be encoded by a genome (Nilsen and Graveley, 2010). It also appears to be tissue-specific (Wang et al., 2008a; Xu et al., 2002) and regulated in development (Kalsotra and Cooper, 2011), as well as implicated in diseases such as cancers (Pal et al., 2012). Hence, detecting isoforms in different cell types or samples is an important step to decipher cellular regulatory programs or to identify alternative transcripts responsible for diseases.

In chapter 4, we described a method called FlipFlop (Bernard et al., 2014) to identify and quantify transcript isoforms from RNA-seq reads aligned on a reference genome. FlipFlop belongs to the "genome-guided transcript estimation" methods reported in table 3.1.

However, the performances of both FlipFlop and other state-of-the-art methods reported in the experimental section of chapter 4 show that the so-called isoform deconvolution problem is far from being solved and is still challenging. This is due in particular to identifiability issues (the fact that different combinations of isoforms can correctly explain the observed reads), particularly at low coverage, which limits the statistical power of the inference methods.

One promising direction to improve isoform deconvolution is to exploit several samples at the same time, such as biological replicates or time course experiments. If some isoforms are shared by several samples, potentially with different abundances, the identifiability issue may vanish and the statistical power of the deconvolution methods may increase due to the availability of more data for estimation. For example, the state-of-the-art methods CLIIQ (Lin et al., 2012) and MiTie (Behr et al., 2013) perform joint isoform deconvolution across multiple samples by formulating the problem as an NP-hard combinatorial problem solved by mixed integer programming. MiTie avoids an explicit enumeration of candidate isoforms using a pruning strategy, which can drastically speed up the computation in some cases but remains very slow in other cases. The Cufflinks/Cuffmerge (Trapnell et al., 2010) method uses a more naive and straightforward approach, where transcripts are first predicted independently on each sample, before being merged (with some heuristics) in a unique set.

In this chapter, we present a method for isoform deconvolution from multiple samples. When applied to a single sample, the method boils down to FlipFlop (Bernard et al., 2014); thus, we simply refer to the multi-sample extension of the technique as FlipFlop as well. We formulate the

isoform deconvolution problem as a continuous convex relaxation of the combinatorial problem solved by CLIIQ and MiTie, using the group-lasso penalty (Yuan and Lin, 2006; Lounici et al., 2009) to impose shared sparsity of the models estimated on each sample. The group-lasso penalty allows us to select a few isoforms among many candidates jointly over samples, while assigning sample-specific abundance values. By doing so, it shares information across samples but still considers each sample to be specific, without learning a unique model for all samples together as a merging strategy would do. Compared to CLIIQ or MiTie, FlipFlop addresses a convex optimization problem efficiently, and involves an automatic model selection procedure to balance the fit of the data against the number of detected isoforms.

The rest chapter is organized as follows. Section 5.2 formulates the isoform deconvolution problem jointly over several samples and describes an efficient convex optimization procedure to solve it. Section 5.3 shows experimentally, on simulated and real data, that FlipFlop is more accurate than simple pooling strategies and than other existing methods for isoform deconvolution from multiple samples. Section 5.4 discusses the results.

5.2 Proposed approach

The deconvolution problem for a single sample can be cast as a sparse regression problem of the observed reads against expressed isoforms, and solved by ℓ_1-penalized regression techniques, where the ℓ_1 penalty controls the number of expressed isoforms. When several samples are available, we propose to generalize this approach by using a convex penalty that leads to small sets of isoforms jointly expressed across samples, as we explain below.

5.2.1 Multi-dimensional splicing graph

The splicing graph for a gene in a single sample is a directed acyclic graph with a one-to-one mapping between the set of possible isoforms of the gene and the set of paths in the graph. The nodes of the graph typically correspond to exons, sub-exons (Li et al., 2011b,a; Behr et al., 2013) or ordered sets of exons (Montgomery et al., 2010; Bernard et al., 2014)—the definition we adopt here as it allows to properly model long reads spanning more than 2 exons (Bernard et al., 2014). The directed edges correspond to links between possibly adjacent nodes.

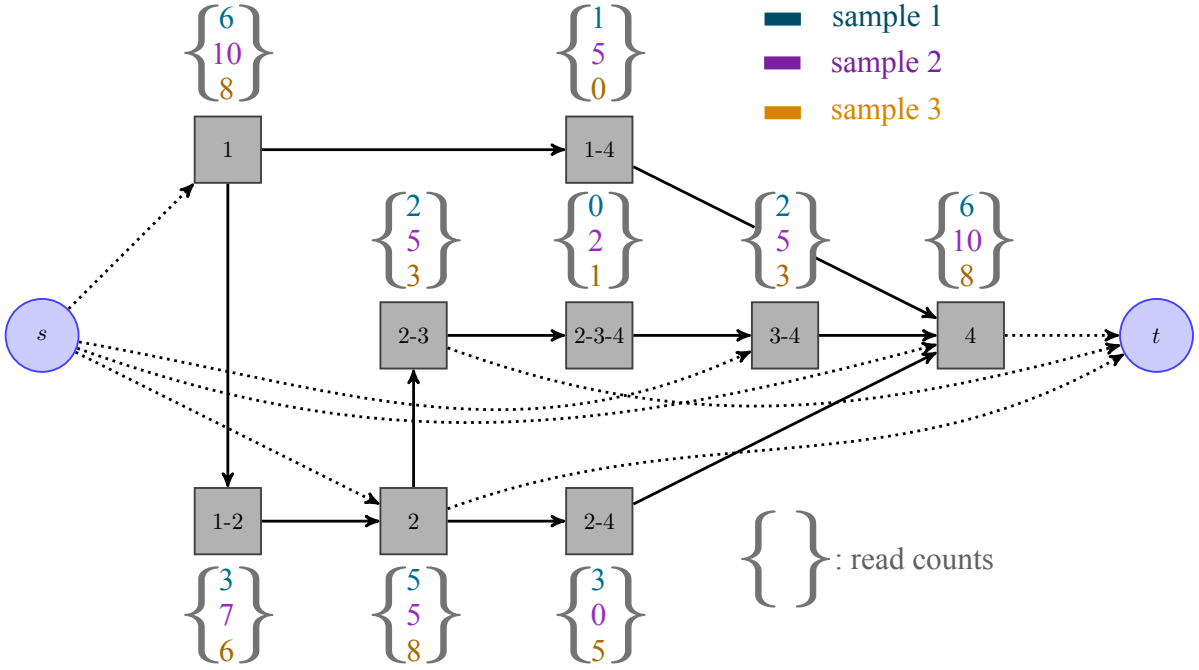


FIGURE 5.1: Multi-dimensional splicing graph with three samples. Each candidate isoform is a path from source node s to sink node t. Nodes denoted as grey squares correspond to ordered set of exons. Each read is assigned to a unique node, corresponding to the exact set of exons that it overlaps. Note that more than 2 exons can constitute a node, properly modeling reads spanning more than 2 exons. A vector of read counts (one component per sample) is then associated to each node of the graph. Note also that some components of a vector can be equal to zero.

When working with several samples, we choose to build the graph based on the read alignments of all samples pooled together. Since the exons used to build the graph are estimated from read clusters, this step already takes advantage of information from multiple samples, and leads to a more accurate graph. We associate a list of read counts, as many as samples, with each node of the graph. In other words, we extend the notion of splicing graph to the multiple-sample framework, using a shared graph structure with specific count values on each node. Our multi-dimensional splicing graph is illustrated in figure 5.1.

5.2.2 Joint sparse estimation

Before to explain our joint sparse estimation procedure, we recall below and extend some of the notations previously introduced in section 3.2.2. We call $G = (V, E)$ the multi-dimensional splicing graph where V is the set of vertices and E the set of edges. We denote by $\mathcal{P}$ the set of all paths in G. By construction of the graph, each path $p \in \mathcal{P}$ corresponds to a unique candidate isoform. We denote by y_v^t the number of reads falling in each node $v \in V$ for each sample $t \in \{1, \dots, T\}$, where T is the number of samples. We denote by $\beta_p^t \in \mathbb{R}_+$ the abundance

of isoform p for sample t. Finally, we define for every path p in $\mathcal{P}$ the T-dimensional vector of abundances $\boldsymbol{\beta}_p = [\beta_p^1, \beta_p^2, \ldots, \beta_p^T]$, and denote by $\boldsymbol{\beta} = [\boldsymbol{\beta}_p]_{p\in\mathcal{P}}$ the matrix of all abundances values with $|\mathcal{P}|$ rows and T columns.

We propose to estimate $\boldsymbol{\beta}$ through the following penalized regression problem:

$$\min_{\boldsymbol{\beta}} \quad \mathcal{L}(\boldsymbol{\beta}) + \lambda \sum_{p\in\mathcal{P}} \| \boldsymbol{\beta}_p \|_2 \quad \text{such that} \quad \boldsymbol{\beta}_p \geq 0 \quad \text{for all } p \in \mathcal{P}, \tag{5.1}$$

where $\mathcal{L}$ is a convex smooth loss function defined below, $\| \boldsymbol{\beta}_p \|_2 = \sqrt{\sum_{t=1}^{T} (\beta_p^t)^2}$ is the Euclidean norm of the vector of abundances of isoform p across the samples, and λ is a non-negative regularization parameter that controls the trade-off between loss and sparsity. The $\ell_{1,2}$-norm $\|\boldsymbol{\beta}\|_{1,2} = \sum_{p\in\mathcal{P}} \| \boldsymbol{\beta}_p \|_2$, sometimes called the group-lasso penalty, induces a shared sparsity pattern across samples: solutions of (5.1) typically have entire rows equal to zero (Yuan and Lin, 2006), while the abundance values in the non-zero rows can be different among samples. This shared sparsity-inducing effect corresponds exactly to our assumption that only a limited number of isoforms are present across the samples (non-zero rows of β). It can be thought of as a convex relaxation of the number of isoforms present in at least one sample, which is used as criterion in the combinatorial formulations of CLIIQ and MiTie.

We define the loss function $\mathcal{L}$ as the sum of the T sample losses, thus assuming independence between samples as reads are sampled independently from each sample. The loss is derived from the Poisson negative likelihood (the Poisson model has been successfully used in several RNA-seq studies (Jiang and Wong, 2009; Salzman et al., 2011; Xia et al., 2011; Bernard et al., 2014)) so that the general loss is defined as

$$\mathcal{L}(\boldsymbol{\beta}) = \sum_{t=1}^{T} \sum_{v\in V} \left[p_v^t - y_v^t \log p_v^t \right] \quad \text{with} \quad p_v^t = \left(N^t l_v \sum_{p\in\mathcal{P}: p\ni v} \beta_p^t \right),$$

where N^t is the total number of mapped reads in sample t and l_v is the effective length of node v, as defined in section 4.2.1. The sum $\sum \beta_p^t$ over all $p \in \mathcal{P}$ that contain node v represents the sum of expressions in sample t of all isoforms involving node v.

5.2.3 Candidate isoforms

Since $|\mathcal{P}|$ grows exponentially with the number of nodes in G, we need to avoid an exhaustive enumeration of all candidate isoforms $p \in \mathcal{P}$. FlipFlop efficiently solves problem (5.1) in the case

where $T = 1$, *i.e.*, the ℓ_1-regularized regression $\min_{\boldsymbol{\beta}_p \in \mathbb{R}_+} \mathcal{L}(\boldsymbol{\beta}) + \lambda \sum_{p \in \mathcal{P}} \beta_p$ using network flow techniques, without requiring an exhaustive path enumeration and leading to a polynomial-time algorithm in the number of nodes.

Unfortunately, this network flow formulation does not extend trivially to the multi-sample case. We therefore resort to a natural two-step heuristic: we first generate a large set of candidate isoforms by solving $T + 1$ one-dimensional problems—the T independent ones, plus the one corresponding to all samples pooled together—for different values of λ, and taking the union of all selected isoforms, and we then solve (5.1) restricted to this union of isoforms. This approach can potentially miss isoforms which would be selected by solving (5.1) over all paths $p \in \mathcal{P}$ and are not selected for any single sample or when pooling all reads to form a single sample, but it allows for an efficient approximation of (5.1). We observe that it leads to good results in various settings in practice, as shown in the experimental part.

5.2.4 Model selection

We solve (5.1) for a large range of values of the regularization parameter λ, obtaining solutions from very sparse to more dense (a sparse solution involves few non-zero abundance vectors $\boldsymbol{\beta}_p$). Each solution, *i.e.*, each set of selected isoforms obtained with a particular λ value, is then re-fitted against individual samples—without regularization but keeping the non-negativity contraint—so that the estimated abundances do not suffer from shrinkage (Tibshirani, 1996). The solution with the largest BIC criterion (Schwarz, 1978), where the degree of freedom of a group-lasso solution is computed as explained in (Yuan and Lin, 2006), is finally selected. Note that although the same list of isoforms selected by the group-lasso is tested on each sample, the refitting step lets each sample pick the subset of isoforms it needs among the list, meaning that all samples do not necessarily share *all* isoforms at the end of the deconvolution.

5.3 Experimental validation

We show results on simulated human RNA-seq data with both increasing coverage and increasing number of samples, with different simulation settings, and on real RNA-seq data. In all cases, reads are mapped to the reference with TopHat2 (Trapnell et al., 2009). We compare FlipFlop implementing the group-lasso approach (5.1) to the simpler strategy of pooling all samples together, running single-sample FlipFlop (Bernard et al., 2014) on the merged data, and

performing a fit for each individual sample data against the selected isoforms. We also assess the performance of MiTie (Behr et al., 2013) and of the version 2.2.0 of the Cufflinks/Cuffmerge package (Trapnell et al., 2010). Performances on isoform identification are summarized in terms of Fscore, the harmonic mean of precision and recall, as used in other RNA-seq studies (Lin et al., 2012; Behr et al., 2013). Of note, in all the following experiments, we consider a *de novo* setting, without feeding any of the methods with prior transcript annotations (*i.e.*, MiTie and FlipFlop first reconstruct sub-exons and build the splicing graph, then perform isoform deconvolution).

5.3.1 Influence of coverage and sample number

The first set of simulations is performed based on the 1329 multi-exon transcripts on the positive strand of chromosome 11 from the RefSeq annotation (Pruitt et al., 2005). Single-end 150bp reads are simulated with the RNASeqReadSimulator software (available at `http://alumni.cs.ucr.edu/~liw/rnaseqreadsimulator.html`). We vary the number of reads from 10 thousand to 10 million per sample (corresponding approximately to sequencing depth from 1X to 1000X) and the number of samples from 1 to 10. All methods are run with default parameters, except that we fix *region-filter* to 40 and *max-num-trans* to 10 in MiTie as we notice that choosing these two parameter values greatly increases its performances (see figure A.1 of appendix A for a comparison between MiTie with default parameters or not).

Figure 5.2 shows the Fscore in two different settings: the *Equal* setting corresponds to a case where all samples express the same set of transcripts at the same abundances (in other words each sample is a noisy realization of a unique abundance profile), while in the *Different* setting the abundance profiles of each sample are generated independently. Hence in that case the samples share the same set of expressed transcripts but have very different expression values (the maximum correlation between two abundance vectors is 0.088).

In all cases and for all methods, the higher the coverage or the number of samples, the higher the Fscore. In the *Equal* case, the group-lasso and merging strategies give almost identical results, which shows the good behavior of the group-lasso, as pooling samples in that case corresponds to learning the shared abundance profile. In the *Equal* case again, for all methods the different Fscore curves obtained with increasing number of samples converge to different plateaux. None of these levels reaches a Fscore of 100, but the group-lasso level is the highest (together with the merging strategy). In the *Different* case, the group-lasso shows equal or higher Fscore

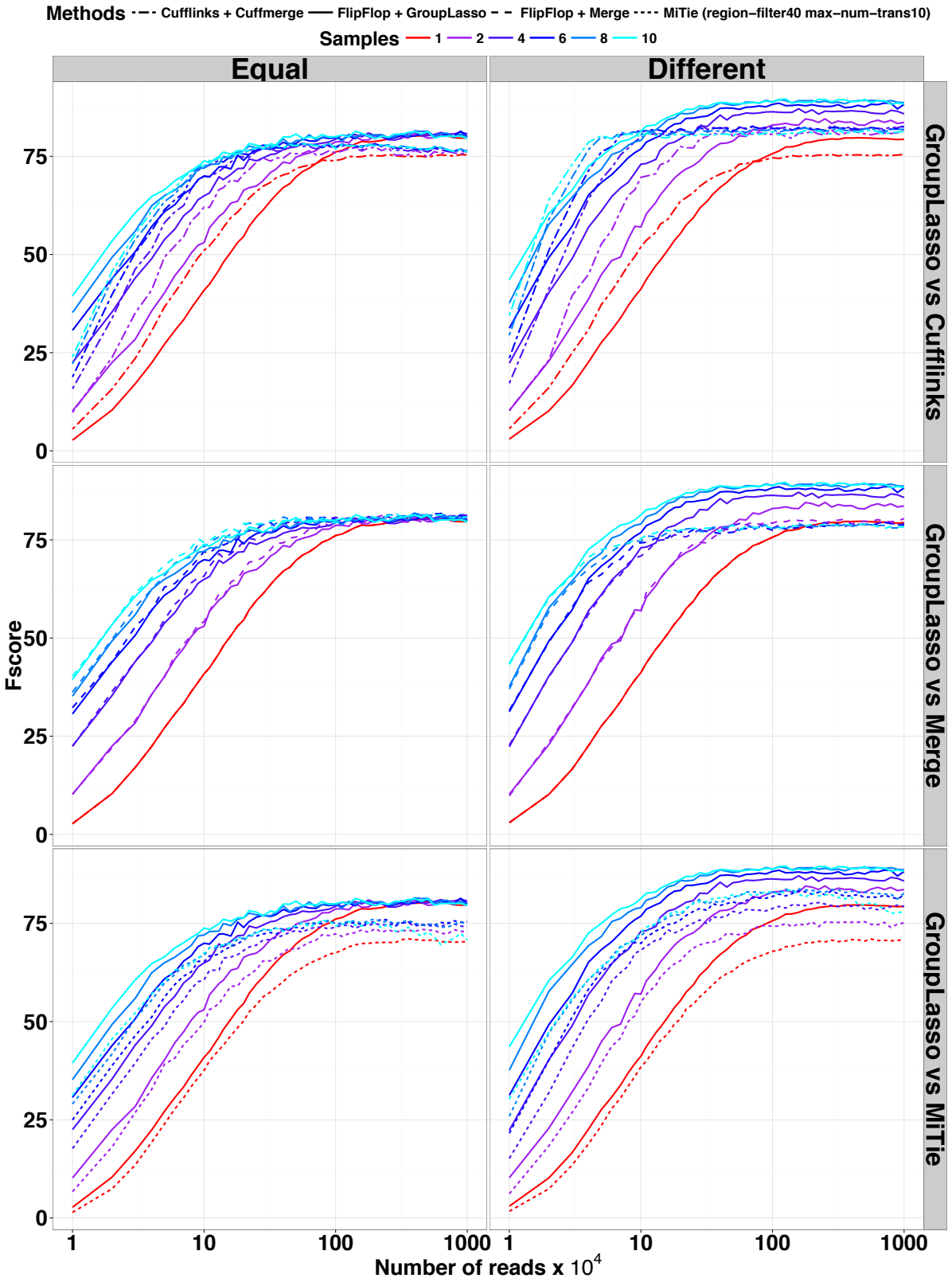


FIGURE 5.2: Human simulations with increasing coverage and number of samples.

	1-3 samples	4-6 samples	7-10 samples
1-10 coverage ($\times 10^4$ reads)	1	1	1
10-50 coverage ($\times 10^4$ reads)	1	0.035	$< 10^{-16}$
50-100 coverage ($\times 10^4$ reads)	0.040	$< 10^{-16}$	$< 10^{-16}$
100-1000 coverage ($\times 10^4$ reads)	$< 10^{-16}$	$< 10^{-16}$	$< 10^{-16}$

TABLE 5.1: Statistical testing to assess performances in the *Different* human simulation setting presented in figure 5.2, for different ranges of coverage and number of samples. Numbers correspond to the Benjamini Hochberg adjusted p-values when testing the null hypothesis that the Fscore obtained with FlipFlop+GroupLasso are lower than the Fscore obtained with Cufflinks+Cuffmerge (one-sided paired t-test). Note that when testing FlipFlop+GroupLasso against MiTie, all adjusted p-values are extremely small.

than the merging strategy, with a great improvement when the coverage or the number of samples increases. The group-lasso also outperforms the Cufflinks/Cuffmerge method for all numbers of samples when the coverage is larger than 80. When using more than 5 samples the group-lasso shows greater Fscore as soon as the coverage is bigger than 15. Finally, the group-lasso outperforms MiTie for all number of samples and all coverages. Of note, the group-lasso performances are better in the *Different* setting than in the *Equal* setting, showing that our multi-sample method can efficiently deal with diversity among samples. Statistical significance associated with results of figure 5.2 are given in table 5.1

We also investigate the influence of the read length on the performance of the compared methods in the *Different* setting. Figure 5.3 shows the obtained Fscore when using either 2 or 5 samples with a fixed 100×10^4 coverage, while read length varies from 75bp to 300bp. Because we properly model long reads in our splicing graph the group-lasso performance greatly increases with the read length, proportionally much more than other state-of-the-art methods. When using 5 samples and long 300bp reads, the group-lasso reaches a very high Fscore of 90 (compared to 84 for the second best Cufflinks/Cuffmerge method), showing that our method is very well adapted to RNA-seq design with long reads and several biological replicates.

We finally check that our method generalizes well to paired-end reads. Figure 5.4 provides a comparison of the tested methods on simulations in the *Different* setting using both paired or single-end reads at comparable coverages, both for "low" and "high" coverage cases and different number of samples. Our group-lasso method achieves the best performances in both the paired and single-end settings in the high coverage case and also in the low coverage case when using 9 samples. For the Cufflinks/Cuffmerge methods the paired-end setting is systematically a bit better than the single-end one, while for both our group-lasso approach and MiTie the two settings are either comparable or better in the single-end case.

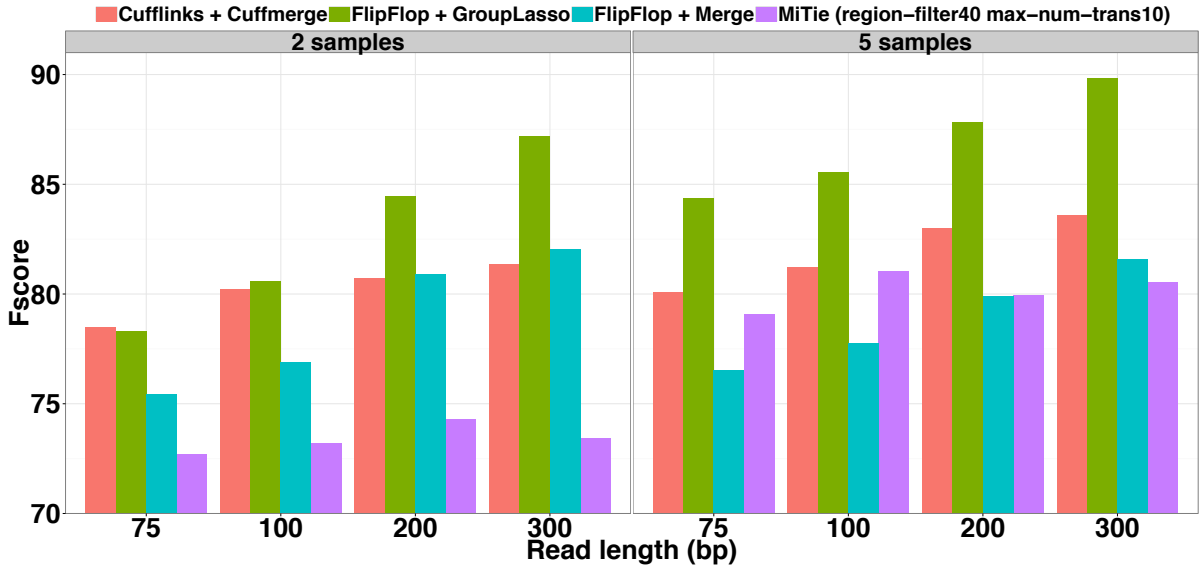


FIGURE 5.3: Human simulations with various read lengths.

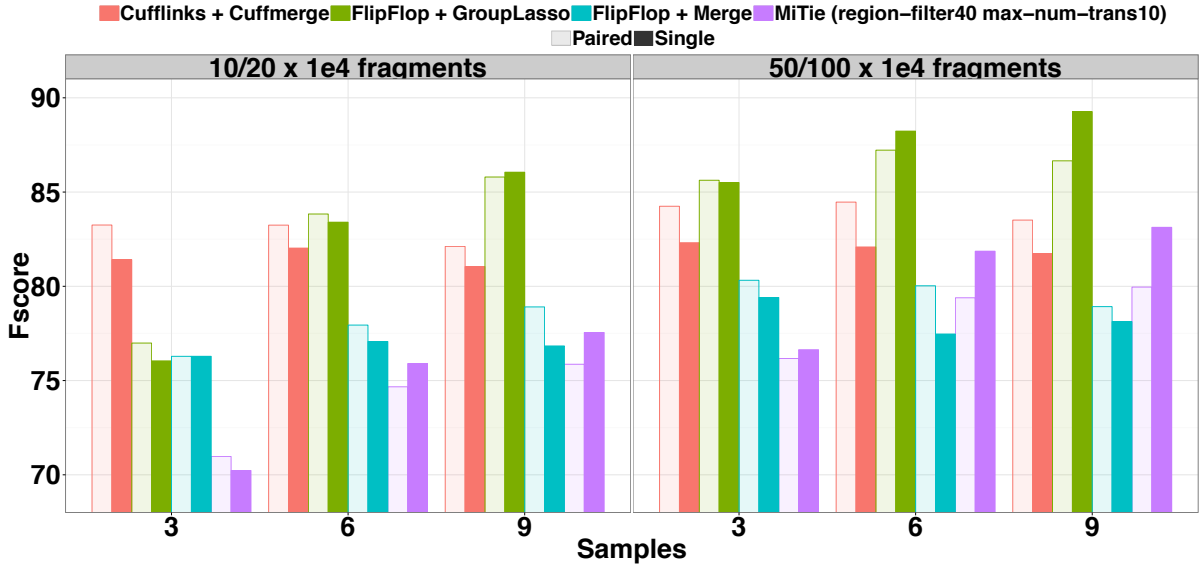


FIGURE 5.4: Simulation using both paired or single-end reads at comparable coverage. The legend 10/20 or 50/100 represents $10^4\times$ the number of sequenced fragments in the paired-end setting versus the single-end setting (the number of sequenced reads is then equal in the two settings, while the number of sequenced fragments is twice higher is the single-end setting). The read length is fixed to 150bp and the mean fragment size to 350bp in the paired-end setting.

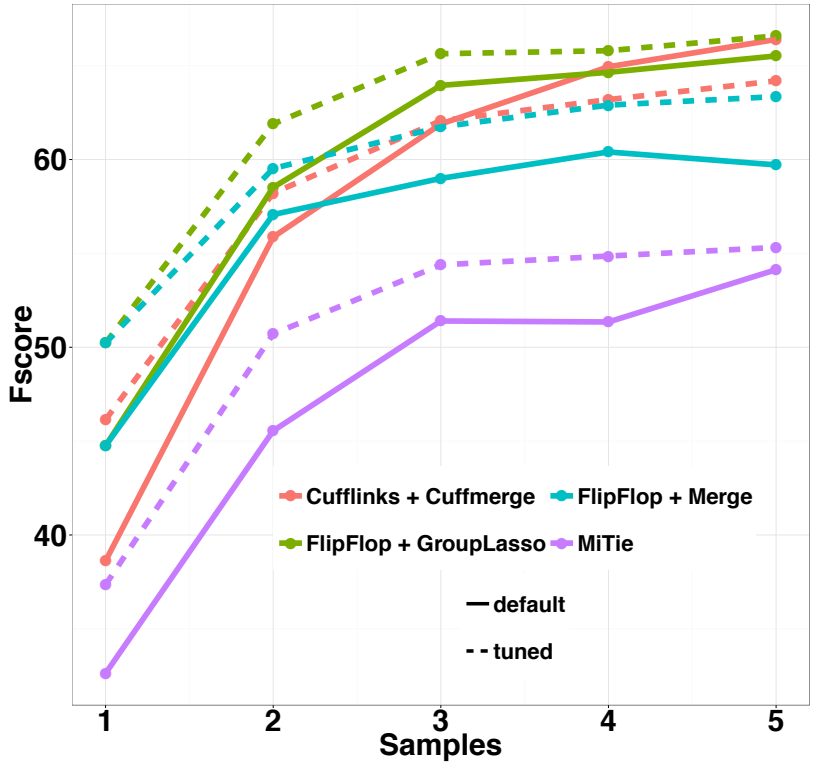


FIGURE 5.5: Fscore results on the Flux Simulator simulations.

5.3.2 Influence of hyper-parameters with realistic simulations

The second set of simulations is performed using a different and more realistic simulator, the Flux Simulator Griebel et al. (2012), in order to check that our approach performs well regardless the choice of the simulator. Coverage and single-end read length are respectively fixed to 10^5 reads and 150bp, and we run experiments for one up to five samples. We study the influence of hyper-parameters on the performances of the compared methods, and show that our approach leads to better results with optimized parameters as well. Hyper-parameters are first tuned on a training set of 600 transcripts from the positive strand of chromosome 11, which is subsequently left aside from the evaluation procedure after tuning. We start by jointly optimizing a set of pre-processing hyperparameters. We then keep the combination that leads to the best training Fscore, and we jointly optimize a set of prediction hyperparameters. More specifically, we optimize 7 values of 3 different pre-processing or prediction parameters (hence 7^3 different combinations in both cases), except that for MiTie we add 2 values of one pre-processing parameter and 3 values of a fourth prediction parameter (hence optimizing over 9×7^2 and 3×7^3 parameters). A more detailed description of the optimized parameters is given in tables B.1 and B.2 of appendix B.

Fscore is shown on figure 5.5 for 600 other test transcripts, for both default and tuned settings

(except that again we set *region-filter* to 40 and *max-num-trans* to 10 in MiTie instead of using all default parameters as it greatly improves its performances, see figure A.2 of appendix A for a comparison of several versions of MiTie). For all methods and for both default and tuned settings, performances increase with the number of samples. Except for Cufflinks/Cuffmerge for the last three sample numbers, all methods improve their results after tuning of their hyper-parameters. When using default parameter values, the group-lasso shows the largest Fscore for the first three sample numbers, while Cufflinks/Cuffmerge is slightly better for the very last sample number. When using tuned parameter values, the group-lasso approach outperforms all other methods for the first three sample numbers, and is slightly better or equal to the default version of Cufflinks/Cuffmerge for the last two sample numbers.

5.3.3 Experiments with real data

We use five samples from time course experiments on *D. melanogaster* embryonic development. Each sample corresponds to a 2-hour period, from 0 to 10 hours (0-2h, 2-4h, ..., 8-10h). Data is available from the modENCODE (Celniker et al., 2009) website. For each given period we pooled all 75bp single-end technical replicate reads available, ending up with approximately 25 to 45 million mapped reads per sample. A description of the samples is given in table B.3 of appendix B. Data from the same source were also used in the MiTie paper (Behr et al., 2013).

Because the exact true sets of expressed transcripts is not known, we validated predictions based on public transcript annotations. We built a comprehensive reference using three different databases available on the UCSC genome browser (Karolchik et al., 2004), namely the RefSeq (Pruitt et al., 2005), Ensembl (Cunningham et al., 2015) and FlyBase (Marygold et al., 2013) annotations. More specifically, we took the union of the multi-exon transcripts described in the three databases, while considering transcripts with the same internal exon/intron structure but with different length of the first or the last exon as duplicates. Reads were mapped to the reference transcriptome in order to restrict predictions to known genomic regions, and we perform independent analysis on the forward and reverse strands. All methods are run with default parameters.

Figure 5.6 shows the Fscore per sample when FlipFlop, MiTie, and Cufflinks are run independently on each sample or when multi-sample strategies are used. Results on the forward and reverse strands are extremely similar. All methods give better results than their independent versions, and the performances of the multi-sample approaches increase with the number

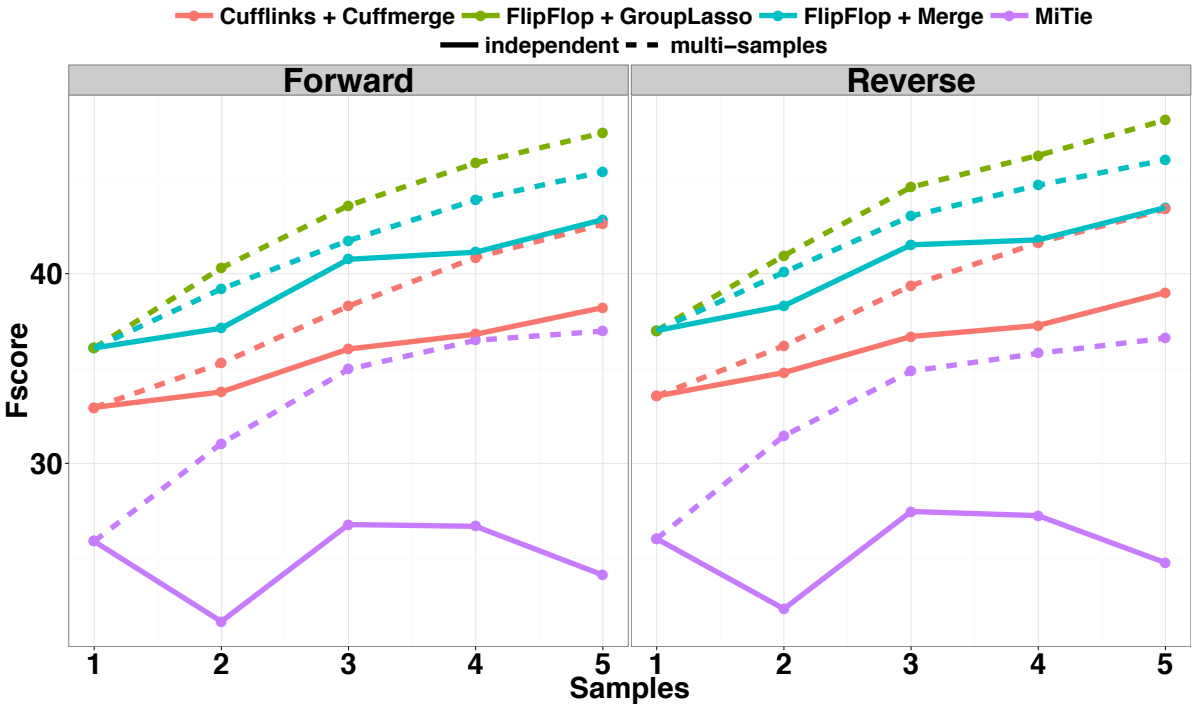


FIGURE 5.6: Fscore results on the modENCODE data.

of used samples. Again, the group-lasso strategy of FlipFlop seems more powerful than the pooling strategy, and gives better Fscore than MiTie and Cufflinks/Cuffmerge in that context.

Running times are given in figure 5.7. Each method was run on a 48 CPU machine at 2.2GHz with 256GB of RAM using 6 threads (all tools support multi-threading). When using only a single sample and 6 threads, Cufflinks, FlipFlop and MiTie respectively completed in $\sim$ 4.2min, $\sim$ 9.5min and $\sim$ 26.6min. When using 5 samples and 6 threads, Cufflinks/Cuffmerge, FlipFlop with group-lasso and MiTie took $\sim$ 0.45h, $\sim$ 1h and $\sim$ 25h.

5.3.4 Illustrative examples

We describe an example as a proof of concept that multi-sample FlipFlop with the group-lasso approach (5.1) can be much more powerful in some cases than its independent FlipFlop version, and than the merging strategy of Cufflinks/Cuffmerge. Figure 5.8 shows transcriptome assemblies of gene CG15717 on the first three modENCODE samples presented in the previous section, denoted as 0-2h, 2-4h and 4-6h on the figure. For each sample, we display the read coverage along the gene, the junctions between exons, and the single-sample FlipFlop and Cufflinks predictions. At the bottom of the figure, we show the 6 RefSeq records as well as the multi-sample predictions obtained with FlipFlop or with Cuffmerge. A predicted transcript is

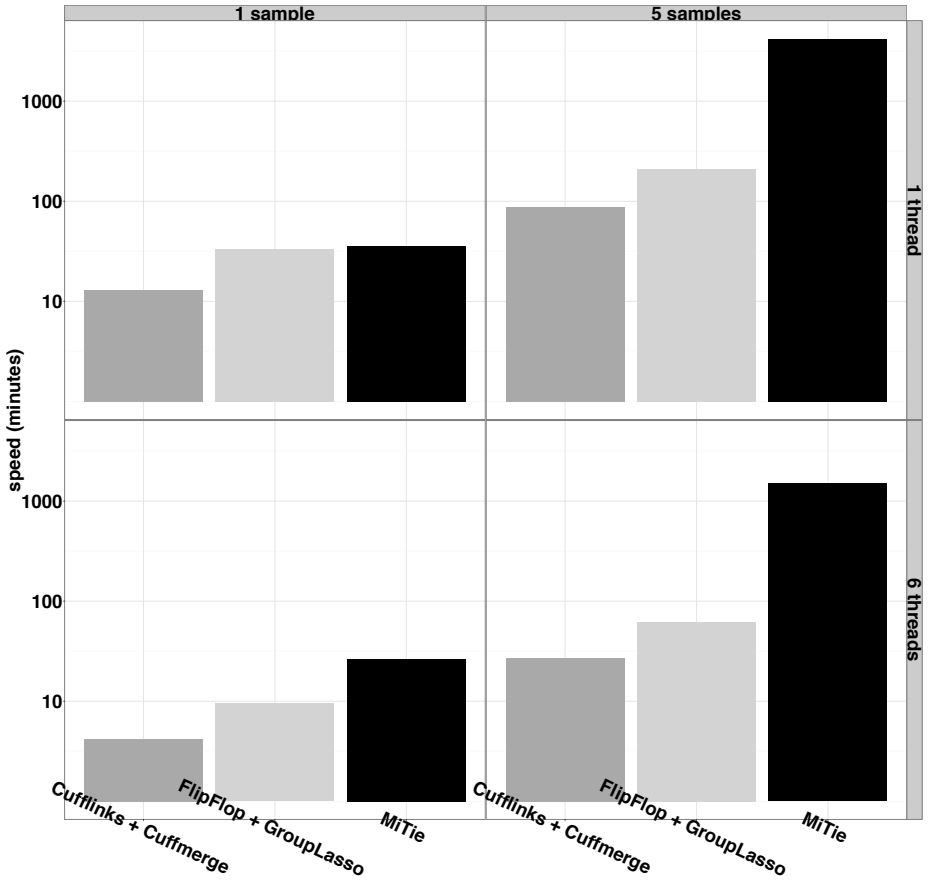


FIGURE 5.7: Running time on the *D.melanogaster* RNA-seq data (forward strand). Each method was run on a 48 CPU machine at 2.2GHz with 256GB of RAM, on either 1 or 6 threads (all tools support multi-threading). MiTie is more than 20 times slower than FlipFlop+GroupLasso when using 5 samples.

considered as valid if all its exon/intron boundaries match a RefSeq record (✓ and ✗ denote validity or not). The estimated abundances in FPKM are given on the right-hand side of each predicted transcript. Of note, the group-lasso predictions come with estimated abundances (one specific value per sample), whereas Cufflinks/Cuffmerge only reports the structure of the transcripts.

For single-sample predictions, FlipFlop and Cufflinks report the same number of transcripts for each sample (respectively 2, 2 and 3 predictions for samples 0-2h, 2-4h and 4-6h), with the same number of valid transcripts, except for the first sample where FlipFlop makes 2 good guesses against 1 for Cufflinks. This difference might be due to the fact that FlipFlop not only tries to explain the read alignement as Cufflinks does, but also the coverage discrepancies along the gene.

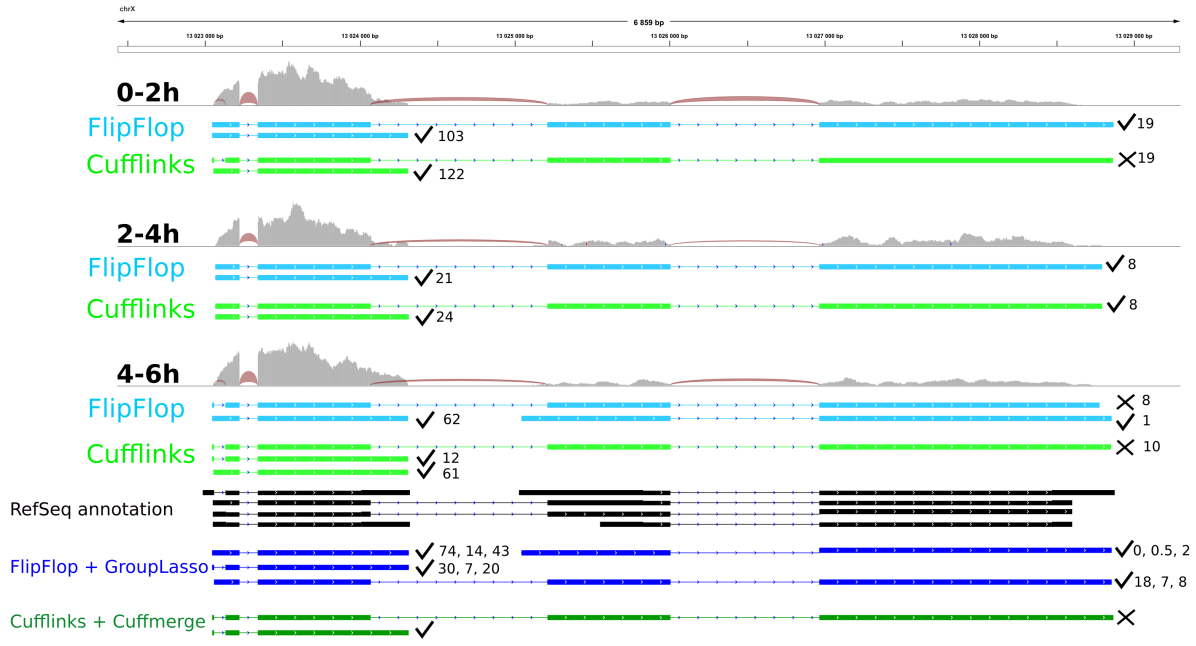


FIGURE 5.8: Transcriptome predictions of gene CG15717 from 3 samples of the modENCODE data. Samples name are 0-2h, 2-4h and 4-6h. Each sample track contains the read coverage (light grey) and junction reads (red) as well as FlipFlop predictions (light blue) and Cufflinks predictions (light green). The bottom of the figure displays the RefSeq records (black) and the multi-sample predictions of the group-lasso (dark blue) and of Cufflinks/Cuffmerge (dark green).

For multi-sample predictions, FlipFlop gives much more reliable results, with 4 validated transcripts (among 4 predictions), while Cufflinks/Cuffmerge makes only 1 good guess out of 2 predictions. FlipFlop uses evidences from all samples together to find transcripts with for instance missing junction reads in one of the sample (such as the one with 30, 7 and 20 FPKM) or lowly expressed transcripts (such as the one with 0, 0.5 and 2 FPKM). Cufflinks/Cuffmerge explains all read junctions but does not seek to explain the multi-sample coverage, which seems important in that example.

Importantly, one can note that the results of multi-sample group-lasso FlipFlop are different from the union of all single-sample FlipFlop predictions (the union coincides here to the results of FlipFlop on the merged sample—data not shown). This illustrates the fact that designing a dedicated multi-sample procedure can lead to more statistical power than merging individual results obtained on each sample independently. We display an additional example in figure A.3 of appendix A.

5.4 Conclusion

We proposed a multi-sample extension of FlipFlop, which implements a new convex optimization formulation for RNA isoform identification and quantification jointly across several samples. Experiments on simulated and real data show that an appropriate method for joint estimation is more powerful than a naive pooling of reads across samples. We also obtained promising results compared to MiTie, which tries to solve a combinatorial formulation of the problem.

Accurately estimating isoforms in multiple samples is an important preliminary step to differential expression studies at the level of isoforms Anders et al. (2012); Trapnell et al. (2013). Indeed, isoform deconvolution from single samples suffers from high false positive and false negatives rates, making the comparison between different samples even more difficult if isoforms are estimated from each sample independently. Although the FlipFlop formulation of joint isoform deconvolution across samples provides a useful solution to define a list of isoforms expressed (or not) in each sample, variants of FlipFlop specifically dedicated to the problem of finding differentially expressed isoforms may also be possible by changing the objective function optimized in (5.1).

Finally, as future multi-sample applications such as jointly analyzing large cohorts of cancer samples or many cells in single-cell RNA-seq are likely to involve hundreds or thousands of samples, more efficient implementations involving in particular distributed optimization may be needed.

Chapter 6

A time- and cost-effective clinical diagnosis tool to quantify abnormal splicing from targeted single-gene RNA-seq

Ce chapitre présente une technique d'aide au diagnostic pour interroger les anomalies d'épissage a partir de données RNA-seq sur gene unique. Notre méthodologie permet de détecter et quantifier les événements d'épissage et de mesurer leur degré d'anormalité par rapport à des échantillons normaux. Nous analysons les défauts d'épissage de patients caracterisés par des altérations germinales de la séquence génomique du gène suppresseur de tumeurs *BRCA1*. Nos résultats sont validés par séquencage Sanger et corroborent ceux d'études à plus grandes échelles realisées par des consortium internationaux avec des techniques différentes.

In this chapter, we present a procedure to query splicing abnormalities from targeted single-gene RNA-seq in a clinical diagnosis setting. We develop a methodology to detect and quantify splicing events from targeted data and measure how abnormal these events might be in patient samples compared to wild-type situations. We also extend ℓ_1-penalized regression techniques initially developed to infer alternative transcripts from bulk RNA-seq data to this new setting. We analyse with our method the splicing landscape of the *BRCA1* gene on a set of both control samples and patients with germline alterations of their genomic sequence. We corroborate our quantification of splicing events on control data with recent large-scale studies that used different techniques. We validate our findings of abnormal events detecting from patient samples with Sanger sequencing. We also analyse a cell line with *BRCA1* mutations recently studied by an international consortium and accurately quantify a complex splicing pattern of overlapping exon skippings.

6.1 Background

We describe below some challenges in molecular diagnosis associated with alternative splicing and introduce a targeted single-gene RNA-seq procedure based on amplicon sequencing.

6.1.1 Molecular diagnosis context

One of the key issues raised in molecular diagnosis is the correct interpretation of the biological consequences of so-called variants of unknown significance (VUS). VUSs correspond to modifications of the genomic sequence that can potentially affect normal pre-mRNA splicing. Indeed, the accuracy of pre-mRNA splicing is determined by the recognition of highly conserved consensus sequences, *i.e.*, the intronic dinucleotides at splice donor and acceptor sites and the intronic branch site, but more loosely defined motifs within exons or introns participate to enhancing or silencing splicing (Hastings and Krainer, 2001; Cartegni et al., 2002), see section 2.1.3 and figure 2.4. As a result, VUS can alter normal splicing and be deleterious via the disruption or creation of consensus sequences or alteration of splicing regulatory motifs (Spurdle et al., 2008).

Many human disease genes harbour mutations that affect pre-mRNA splicing, in particular in cancers (Krawczak et al., 1992; Wang and Cooper, 2007), As an example, one-half of the variations observed in *BRCA* genes are VUSs (Hofstra et al., 2008). Assessing the putative impact of VUSs on splicing is therefore a central issue in order to determine their pathogenicity, and one of the routine challenges faced by molecular geneticists in their day-to-day practice.

6.1.2 Targeted single-gene RNA-seq

Until recently, performing routine RNA screening for each VUS in order to detect a putative splicing anomaly was unrealistic in a diagnosis setting. A compromise was to be found between a time- and cost-effective RNA analysis and the risk of missing a deleterious mutation. To facilitate decision-making and genetic counseling, *in silico* splice tools that predict the impact of VUSs based on the sole DNA sequence can be used to restrict transcript analyses to the most appropriate cases (Houdayer et al., 2008, 2012). However, these tools provide the user with splice site score prediction, but no quantitative information on the amplitude of the splicing defects (Houdayer, 2011; Jian et al., 2014). Hence, analysis of RNA samples from the patient remains the most straightforward and reliable method to describe splicing defects.

Targeted RNA-seq strategies (Levin et al., 2009; Mamanova et al., 2010; Zhang et al., 2014) offer the opportunity to develop simple and robust methods providing qualitative as well as quantitative information on VUS impact at the RNA level. Such strategies, by combining the capture of a relevant subset of a transcriptome and high-throughput sequencing, provide efficient and cost-effective means to study the splicing landscape of regions of interest in great details.

Here we investigate such an approach, where we first amplify the transcripts of interest with (possibly several) long-range PCR, before sequencing the obtained fragments, called *amplicons*, with high-depth RNA-seq. We present new statistical models and algorithms to quantify splicing events from amplicon sequencing data, and measure how abnormal these events might be in patient data compared to wild-type situations. Our approach includes a data normalization procedure (sections 6.2.3 and 6.4.3), an accurate quantification of both splicing or retention events (and more generally of any splice or acceptor donor shift as well), see sections 6.2.4 and 6.2.5, and a full-length transcript prediction step (sections 6.2.6 and 6.4.4). We apply our pipeline to a case study on the *BRCA1* gene, and present promising results that we corroborate with both experimental validation and litterature comparison.

The rest of the chapter is organized as follows. Section 6.2 gives a broad overview of our procedure to query splicing abnormalities from amplicon targeted RNA-seq data and shows various results on *BRCA1* amplicon data, such as the effect of data normalization on the desired signal, the quantification of splicing events on a set of control samples and patient samples and the prediction of complex overlapping splicing events. Section 6.3 summarizes the results and discusses futur work. Section 6.4 details the experimental protocol as well as the statistical analysis of the data and the algorithms implemented to infer the levels of alternative splicing events and estimate the proportions of the full-length transcripts.

6.2 Results and discussion

6.2.1 A pipeline to query splicing abnormalities

We developed a methodology combining targeted single-gene RNA-seq and an associated bioinformatics pipeline to query splicing abnormalities in a clinical context. The method uses amplicon high-throughput sequencing of RNA extracted from lymphoblastoid cell lines derived from from patients' blood samples in order to detect and quantify splicing events. The bioinformatics

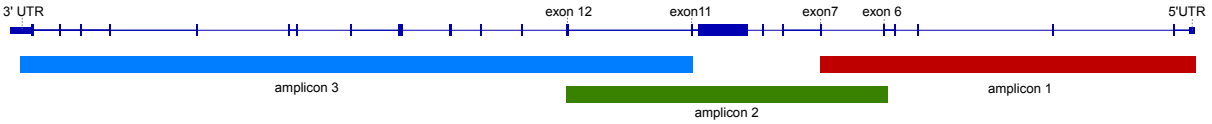


FIGURE 6.1: *BRCA1* amplicon design. The gene structure is displayed at the top of the figure with dark-blue boxes representing exons and thin lines representing introns. Overlapping amplicon regions are shown underneath, and the name of the regions (UTR and coding exons) containing the amplicon primers are written. More specifically primers located in coding exons are defined by the following positions: exon 6 (120-140), exon 7 (56-76), exon 11(8-28), exon 12 (135-155); where numbers into brackets denote the implicated base pairs of the corresponding exon, number 1 being the very 5' end of the exon.

pipeline include data processing and normalization, as well as estimation of splicing events and quantification of discrepancies between patient data and wild-type distributions. The method is described in more details in the section 6.4.4, and the bioinformatics pipeline will be shortly delivered as an open-access tool.

6.2.2 *BRCA1* pilot study

To validate our pipeline, we test it on a study of splicing anomalies of the breast cancer susceptibility gene *BRCA1*, a tumor suppressor gene involved in DNA repair pathways and cell cycle regulation (Roy et al., 2012). *BRCA1* is characterized by a complex alternative splicing landscape (Colombo et al., 2014; Romero et al., 2015) together with an often non-trivial diagnostic interpretation of its genomic alteration potentially disturbing physiological splicing.

Amplicon design

BRCA1 is composed of 23 exons, among which 22 are coding exons, on the long arm of chromosome 17. We use throughout the chapter the RefSeq notation (Pruitt et al., 2005) to name the different exons, more specifically the annotated *NM_007294* RefSeq transcript. It is characterized by a very long exon 10 of 3426bp, while the lengths of all other coding exons range from 42bp to 312bp. Since it is well documented (and we also observe, see section 6.2.4) that a large part of the 3' end of exon 10 is physiologically spliced, we separate it into two parts: exon 10a that contains the first 117bp and exon 10b that contains the following 3309bp.

Because of the length of exon 10, full-length PCR amplification with primers located in both UTRs is impossible. Instead, we perform several amplifications by building an overlapping

Run	Controls	Patients	VUS
run 1	4 controls	8 patients	4 misssense mutations 3 intronic mutations 1 codon deletion
run 2	3 controls	10 patients	all intronic mutations

TABLE 6.1: Summary of samples analyzed in the *BRCA1* pilot study.

amplicon design, with all primers located in different exons. We use 3 amplicons, with primers situated in the 5' UTR and exon 7 (first pair), in exons 6 and 12 (second pair) and in exon 11 and 3' UTR (third pair), as illustrated in figure 6.1. Moreover we locate the primers in constitutive parts of the exons, *i.e.*, not physiologically spliced, so that we maximize the splicing landscape captured with our design (see figure 6.13 for an illustrative example of the constraints on the splicing landscape under study generated by a given design). Such a design allows us to potentially reveal the splicing of any single exon.

Patient selection

We analyze a cohort of 18 patients with breast cancer family history based on the presence of VUS on their *BRCA1* genomic sequence, as summarized in table 6.1. Patients gave their informed consent for genetic testing. DNA was sequenced by Sanger sequencing to detect VUS, both intronic and exonic, and RNA was extracted from lymphoblastoid cell lines and treated with and without puromycin before amplicon sequencing. Puromycin is a translational inhibitor that prevents the non-sense-mediated-decay (NMD) pathway, a process that naturally degrades aberrant truncated transcripts with premature stop-codon (Popp and Maquat, 2013; Lykke-Andersen and Jensen, 2015). NMD inhibition is therefore crucial to reveal the expression of abnormal transcripts from mutated alleles, which further permits the assessment of the pathogenicity of the underlying VUS. RNA was sequenced in two runs, and control samples were added in each run.

6.2.3 Data normalization

For each control and patient we map the short sequenced fragments (the *reads*) to the reference *BRCA1* gene. Mapped reads give quantitative information on the relative abundances of the different regions of the gene (exons or introns, or more generally sub-parts of exons or introns), as well as crucial information about observed junctions between these regions.

In absence of technical and biological artefacts, the abundance of a given nucleotidic base should be roughly proportional to the number of reads that it generates, *i.e.*, the number of reads that start at its specific genomic location (what we call the *5' read count*). Figure 6.2 shows an example of such raw counting data on a control sample, at the nucleotidic base level, on the set of annotated exons in each amplicon. One can observe that physiologically spliced exon 10b is clearly associated with lower 5' read counts than other exons. But while the counting data does contain splicing signal, it also shows exon- and amplicon-specific artefacts: there are discrepancies in the 5' coverage between exons that are not due to splicing. The first amplicon is for instance associated with an artefactual decreasing trend, the second amplicon shows some large outliers in exon 12 and the third amplicon is characterized by a wavy shape. This different experimental biases highlight the need to normalize the raw data to accurately quantify splicing events.

Artefacts are reproducible across controls

Quantifying reproducibility across controls is essential in order to assess the potential of a method to reveal abnormal events, when abnormality is judged in comparison to the wild-type cases. If data across controls are not reproducible, there is no hope to be able to highlight events that deviate from the wild type situations.

To assess reproducibility of artefacts we focus on the 7 control samples from two different runs (see table 6.1), all analyzed in puromycin- and puromycin+ conditions. Given an amplicon and a control sample, we compute a coverage vector based on the cumulative coverage on each base of all exons covered by the amplicon. The Spearman correlation between all pairs of coverage vectors is shown on figure 6.3. Although a hierarchical clustering of the coverage control data reveals a run batch effect (data not shown), the correlation values are very high, both intra- and inter-run: the minimum values for amplicons 1, 2 and 3 are (0.91, 0.78 and 0.59) for intra-run controls and (0.93, 0.66 and 0.49) for inter-run controls. This indicates that using controls both to normalize data or highlight deviations from wild-type situations is a reasonable assumption. Of note, all correlation values are higher when using Pearson correlation instead of Spearman correlation. Note also that control samples do not cluster according to the presence or not of puromycin, and therefore we group controls from the two conditions when normalizing data (see below).

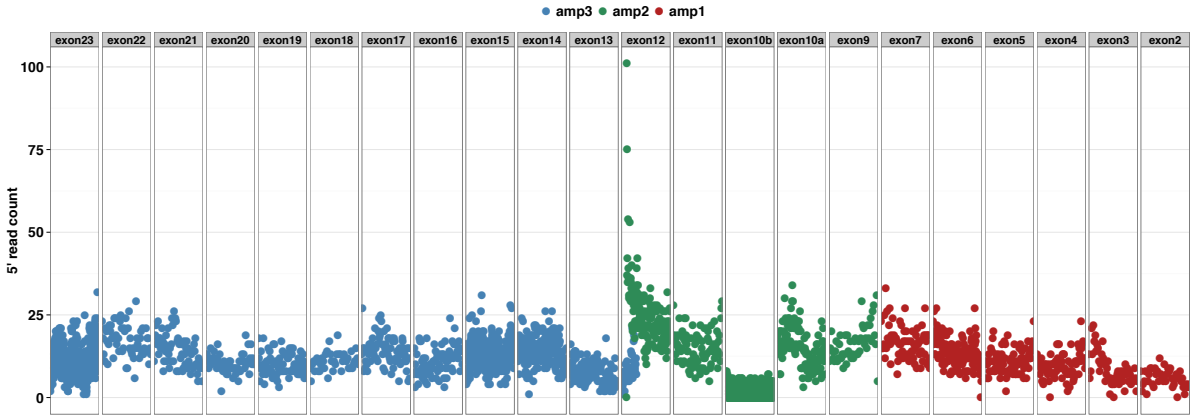


FIGURE 6.2: 5' read count on the set of *BRCA1* exons for each amplicon. Each dot represents the number of reads that start at a specific genomic position. The x-axis is scaled within each exon so that all exons are represented with the same width. As *BRCA1* is located on the reverse DNA strand, we draw the last exon (exon 23) on the leftmost part of the figure. Note that exon 1 is not drawn: given that the part of the exon overlapping with amplicon 1 is shorter than the read length (200bp) no read initiate in that exon. For the same reason exon 8 is not drawn.

Normalization

Raw data shown in figure 6.2 reveal that coverage is not uniform along a given amplicon, partly because of experimental biases. Indeed, physiologically spliced exons cannot explain here all observed coverage discrepancies (for instance, while exons 19 and 22 are not spliced – no junction reads are observed – exon 22 is characterized with a 5' read count of ~ 17 in average whereas the average 5' coverage is ~ 10 on exon 19). High correlations among controls show that experimental artefacts are reproducible across experiments, and allow us to elaborate a procedure to attenuate their amplitude.

Hence we build a normalization methodology to alleviate the non-uniformity of coverage along amplicons on each patient data, using corrections calculated on the controls (see Methods section 6.4.4). In short, scaling factors are estimated on each genomic region on the controls using the ratio of region specific values calculated from a smoothly fitted coverage curve, *i.e.*, a loess local regression (Cleveland and Devlin, 1988). As explained in section 6.4.3, using a smooth fit leverages the hypothesis that scaling factors might be continuous along regions, and the fact that physiologically spliced regions should not be intensively scaled.

Figure 6.4 shows the distribution of scaling factors estimated from the controls from the first run. Scaling factors are conserved across controls: regions that are not scaled (associated with a scaling factor of 1 on figure 6.4) are systematically the same across controls (exon 22 on

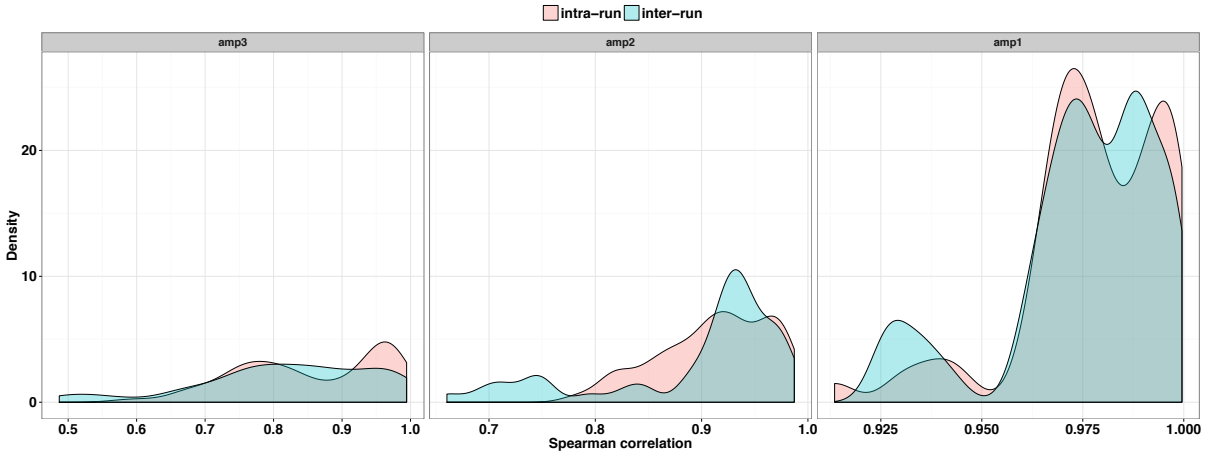


FIGURE 6.3: Distribution of Spearman correlation across the set of controls on each amplicon and for both intra- and inter-runs. Correlations are computed between all pairs of control cumulative coverage vectors. Mean values for amplicon 1, 2 and 3 are equal to (0.98, 0.91, 0.85) for intra-run controls and to (0.98, 0.89, 0.82) for inter-run controls, while minimum values are (0.91, 0.78, 0.59) for intra-run controls and (0.93, 0.66, 0.49) for inter-run controls.

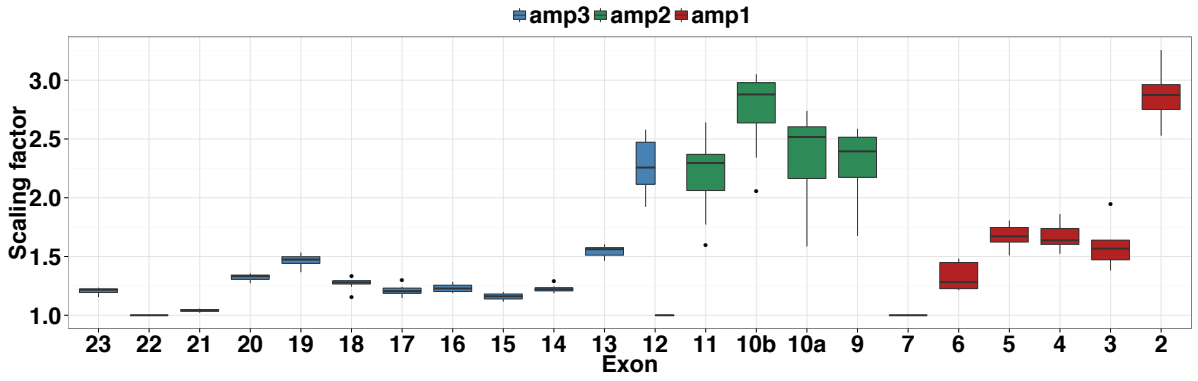


FIGURE 6.4: Scaling factors calculated on a set of 4 controls all analyzed in 2 conditions (with and without puromycin) from the same run. Boxplots show the distribution of the scaling factors across the 8 samples on each exon overlapping a given amplicon.

amplicon 3, exon 12 on amplicon 2 and exon 7 on amplicon 1), while the maximum discrepancy is attained on exon 11 from amplicon 2 with values ranging from 1.59 to 2.64. The narrowness of the boxplots formally demonstrates that coverage trends are conserved across the set of controls. Note that the distribution of scaling factors is very similar with the second run (data not shown). Figure 6.5 illustrates the effect of the normalization procedure on a given control on all amplicons. While raw data are subject to artifical waves and trends, normalized data are flattened. Figure 6.6 also shows the effect of data normalization, but on a patient sample with an abnormal splicing. Normalized data clearly highlight here the splicing defect signal.

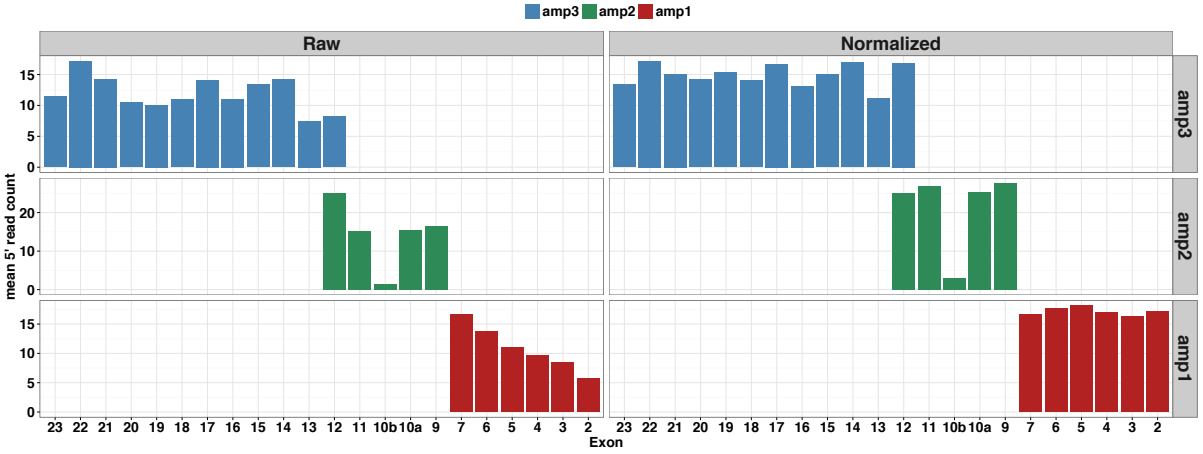


FIGURE 6.5: Effect of data normalization on a control sample. Each bar represents the average of the 5' read count on each exon overlapping an amplicon. Left panel corresponds to the raw counting data, as presented in figure 6.2, while the data on the right panel have been scaled with factors calculated using all controls as explained in section 6.4.3.

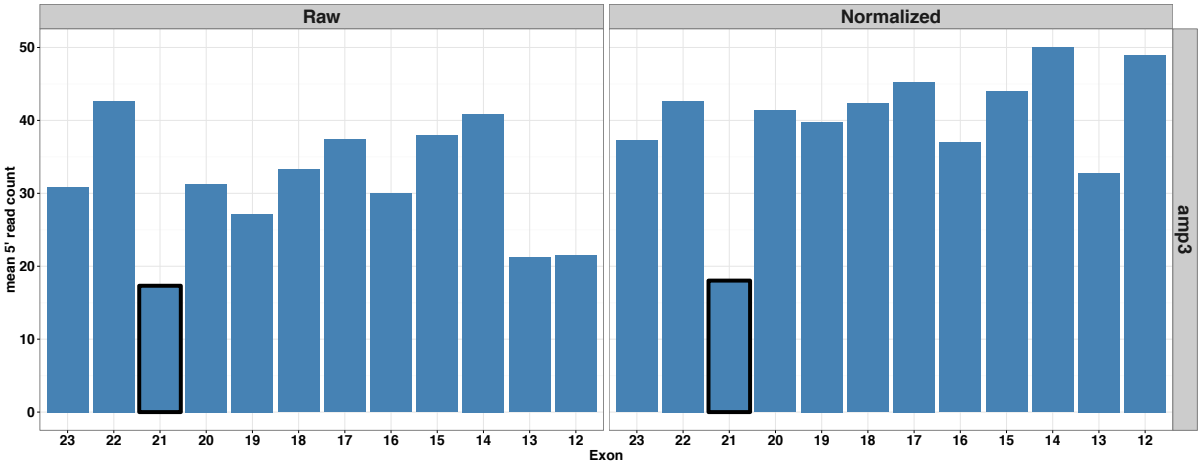


FIGURE 6.6: Effect of data normalization on a patient sample. Data are shown on amplicon 3 only. Exon 21 which is abnormally spliced on that patient is highlighted with a thick black contour. The raw data showed on the left panel indicate a low abundance of exon 21 ($\sim$ 17 mean 5' read count) compared to the one of its neighbour exons ($\sim$ 43 and $\sim$ 31 for exons 22 and 20), but comparable to the one of exon 12 for instance ($\sim$ 21 mean 5' read count). When data are normalized with factors calculated on the controls (right panel), exon 21 is almost not scaled ($\sim$ 18 normalized coverage), while exon 12 is more intensively scaled ($\sim$ 49 normalized coverage).

6.2.4 Quantifying splicing events on controls

We use the normalized data on each amplicon to report quantitative information about splicing or retention of (possibly part) of exons or introns. The percentage of splicing (resp. retention) of each exon (resp. intron), defined as the proportion of transcripts that do exclude (resp. include) the associated exon (resp.intron), can be calculated for a given amplicon. The estimation of the percentages, explained in more detailed in section 6.4.4, relies on the number of reads that map to both exons or junctions between exons. We denote in the later an *event* as either a splicing or a retention, and we use the generic *region* term to designate a part of exon or intron. Note that the reported event values for a given amplicon correspond to the percentage of splicing/retention of regions among all transcripts that are captured by the amplicon, *i.e.*, that contain the amplicon primer pair.

Figure 6.7 shows the events found in the controls, where we keep an event if it is seen with an amplitude of at least 3% in one of the controls. All these wild-type events correspond to splicing of exons or sub-part of exons. The strength of using high-throughput sequencing technologies appears here as we describe events at the base pair level, with for instance wild-type splicing of 3 base pairs in exon 7 or exon 13 and 6 base pairs in exon 1 (these formally correspond to splice donor or acceptor shifts), and are able to quantify any possible event. The amplitudes of events are conserved across controls, showing that deviations of amplitudes from control distributions could be considered as abnormal events and may deserve a deeper look by molecular geneticists.

Of note, the results in Colombo et al. (2014) that performed a large scale systematic analysis of naturally occurring *BRCA1* splicing events from blood-related RNA sources corroborate our findings. Using semi-quantitative capillary electrophoresis analysis of RT-PCR products, they also describe a predominant skipping of 6 base pairs in exon 1 ($\sim$ 50% of full-length signal), followed by a skipping of 3 base pairs in exon 13 and exon 7 and by the skipping of exons 8+9 ($\sim$ 30% of full-length signal). These quantifications are close to our estimates, see figure 6.7. Respectively, all events classified in Colombo et al. (2014) as "predominant" are reported by our methodology. Note that while figure 6.7 quantifies the splicing of exons 8 and 9 separately , the full-length transcript analysis (see section 6.2.6) of control samples tells us that exons 8 and 9 are indeed spliced together (data not shown). Colombo et al. (2014) could not quantify the skipping of the 3' end of exon 10 (exon 10b in figure 6.7), but they qualitatively assess its existence using other splicing assays. One should notice however that our estimate of the proportion of exon

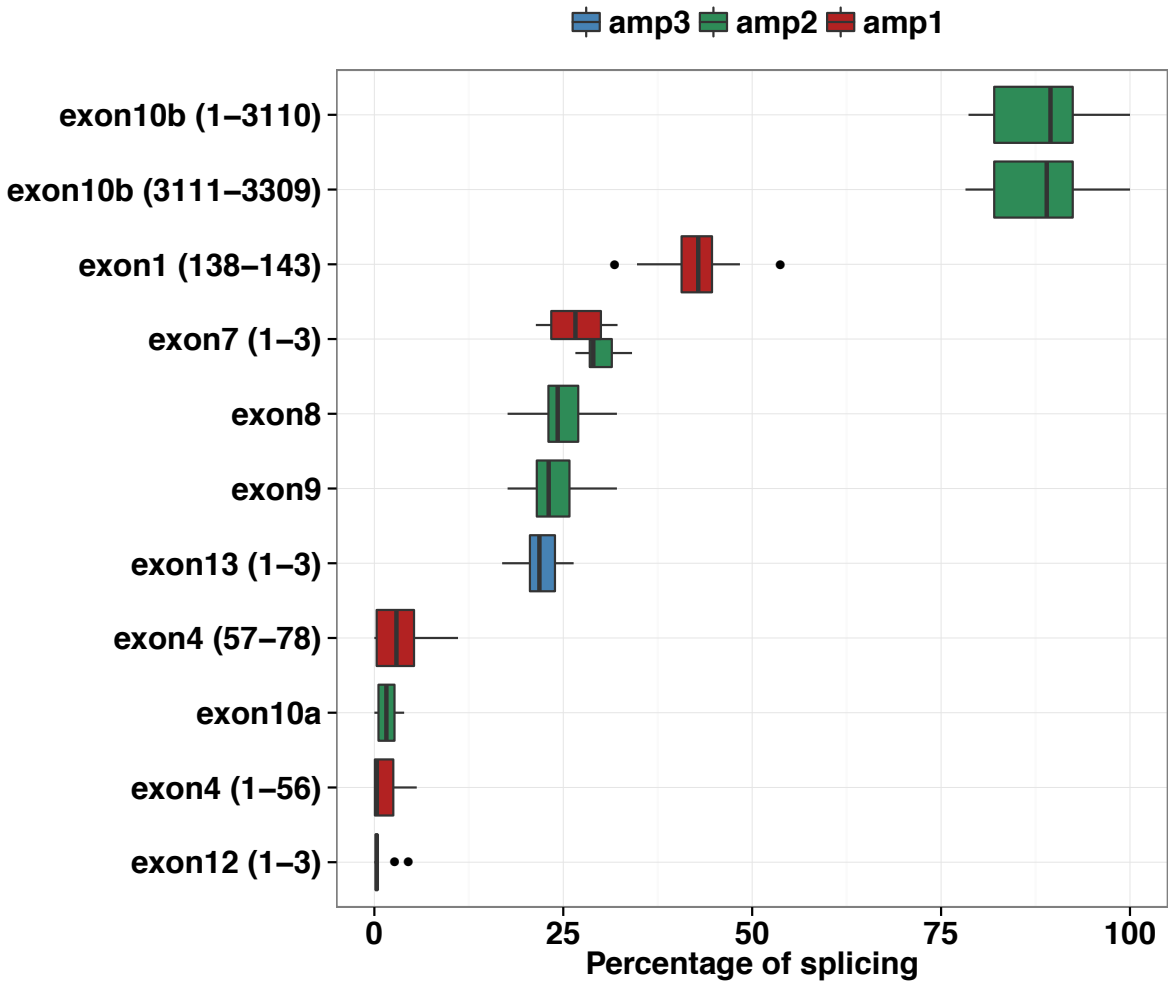


FIGURE 6.7: Percentage of splicing of different regions over the set of 7 controls from 2 runs, all analyzed with and without puromycin. The percentage of splicing of a given region and a given amplicon corresponds to the proportion of transcripts that both contain the amplicon primers and exclude the specific region. A region is denoted by the name of the exon, followed by the implicated base pairs into brackets, where the first base pair correspond to the very 5' end of the exon. When no brackets appear, it means that the exon is spliced in its full length. Low, first quantile, median, third quantile and high values are displayed in the boxplots.

10b skipping is very high ($> 85\%$) and almost surely biased by the preferential amplification of short transcripts excluding the 3309bp long exon 10b. However, this bias is not an issue *per se* as we further focus on deviation from the control distribution when analyzing patient samples (see section 6.2.5).

6.2.5 Detecting abnormal events as deviation from control distributions

For any region we have access to both the percentage of event for a given patient and across controls. We can therefore focus on deviation of the patient observation from the control

amplicon	event	percent of event (puro−)	percent of event (puro+)	p−value (puro−)	p−value (puro+)	mean over controls
amp3	exon21	52.26	47.10	<1e−16	<1e−16	0.29
amp1	intron4 (1441−1499)	3.99	0.00	<1e−16	1	0.00
amp1	exon4 (1−56)	7.90	2.58	2.2e−04	5e−01	1.40

FIGURE 6.8: Detection and quantification of abnormal splicing or retention events on a patient sample. The "event" column reports the names of the parts of exons or introns for which the percentages of splicing or retention are quantified. Names are given under the rules explained in figure 6.7. The "p-value" columns mesure the deviation of the patient observations from the control distribution for both puromycin- and puromycin+ conditions. The "mean over controls" shows the averaged percentages of splicing or retention across controls as a reference. This specific example reveals a clear abnormal splicing of exon 21 that has been further qualitatively validated with Sanger sequencing. Note that this patient sample is the same as the one presented in figure 6.6.

distribution. The rationale being that the larger the deviation the more likely the event to be abnormal, hence the closer the geneticists look should be. We recall that data for all patients are available with and without the addition of puromycin, an inhibitor of the surveillance NMD pathway that degrades mRNA carrying premature stop-codon. Puromycin may then reveal the expression of aberrant transcripts from a mutated allele.

P-values for both puromycin- and puromycin+ conditions are computed, based on the null hypothesis that the patient observation is generated from the control distribution. These p-values measure the distance from the wild-type situations, low p-values indicating that an event is likely to be abnormal. Results are reported on comprehensive tables as illustrated in figure 6.8. Events are ordered based on the minimum p-value across puromycin- and puromycin+, while events associated with very low p-values ($< 10^{-16}$) are re-ordered based on their amplitude. We believe that such an output, automatically reported by our bioinformatics pipeline, allows geneticists to quickly visualize statistically significant abnormal events on a patient sample, together with quantitative indication on the amplitude of those abnormal events.

Furthermore, we qualitatively validate the most significant abnormal events found on each patient with Sanger sequencing of cDNA from patient RNA. More precisely all events reported at the very top of our tables (similarly to the skipping of exon 21 presented in figure 6.8), all associated with low p-values $< 10^{-16}$ in at least puromycin+ condition, have been further observed on Sanger sequencing data. In addition, we compare in figure 6.9 the percentages of these qualitatively validated events estimated with our targeted RNA-seq methodology in the presence or not of puromycin, with distinction if the splicing or retention event leads to the apparition of a premature stop-codon. In the absence of a premature stop-codon the quantification curves are very close with or without puromycin, while in the presence of a premature stop-codon the amplitudes of abnormal splicing are larger in the presence of puromycin (2.6×10^{-7} p-value

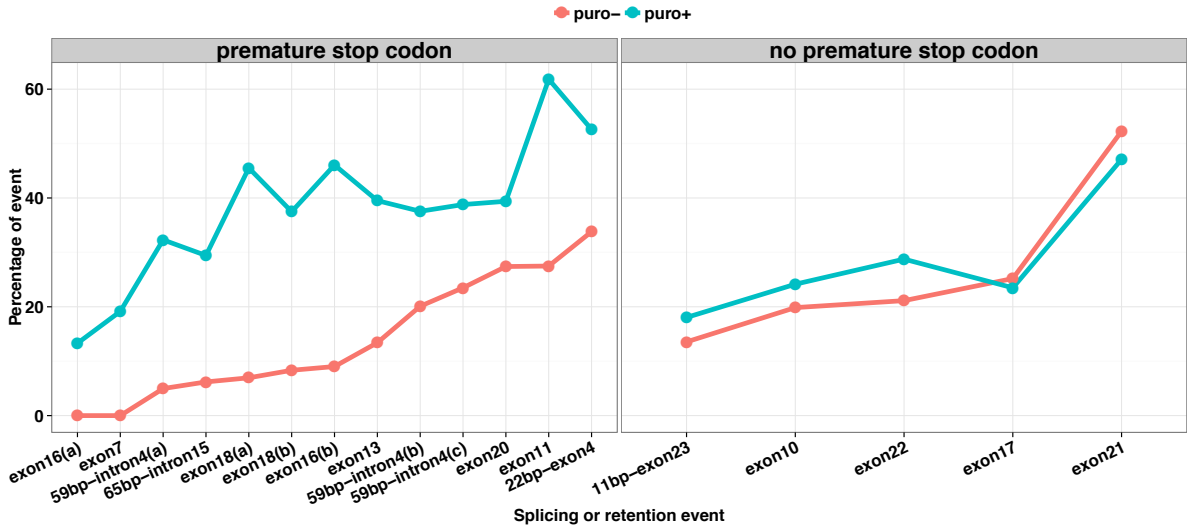


FIGURE 6.9: Effect of puromycin on the quantification of splicing abnormalities. The x-axis corresponds to the name of the splicing or retention events, with an additional letter into brackets when an event arise in several patients. Additionally, 59bp-intron4 and 65bp-intron15 refer to the retention of 59 and 65 base pairs of introns 4 and 15, while 22bp-exon4 and 11bp-exon23 refer to the skipping of 22 and 11 base pairs of exons 4 and 23. All the reported events have been further validated with Sanger sequencing. Each event is classified into "premature stop codon" or "no premature stop codon" depending on whether or not it creates a codon UGA, UAG or UAA upstream to the last exon (exon 23).

with a one-sided paired t-test with Benjamini-Hochberg correction). This demonstrates that the addition of puromycin makes it possible to assess loss of function by revealing abnormal splicing that can be quantified with our targeted RNA-seq methodology.

6.2.6 Deciphering complex splicing events with full-length transcript prediction

The analysis presented so far focuses on the detection of abnormal events at the *region level*, *i.e.*, it provides local information on the percentage of splicing (resp. retention) of part of exons (resp. introns); but it does not give insight about the possible combination of events into different transcripts. Ultimately, it would be interesting to work at the *transcript level*, with access to the proportions of all full-length transcripts, both wild-type or abnormal. Inferring the full-length transcripts from short-sequencing reads is known to be a hard problem (Steijger et al., 2013; Hayer et al., 2015) as reads do not generally map to a unique transcript, such that a non-trivial deconvolution step of the mapped reads into the transcripts is needed. Moreover, we need to implement a *de novo* transcript reconstruction approach, as the goal of our approach is to detect abnormal splicing events that might not be documented in databases.

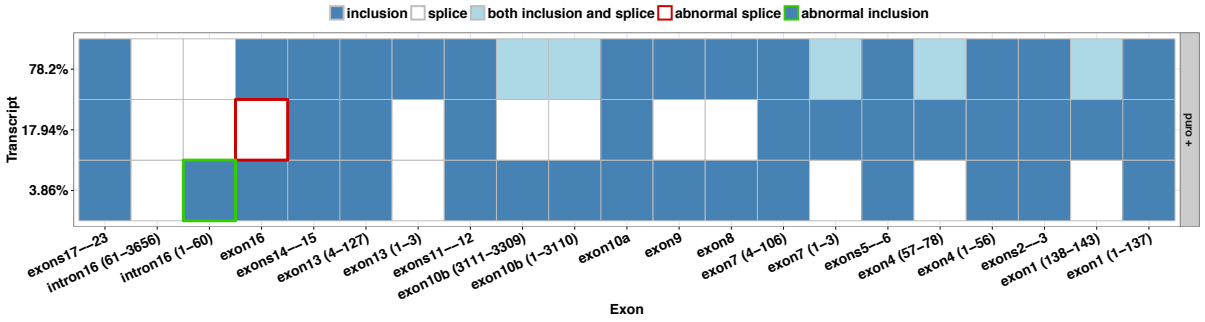


FIGURE 6.10: Visualization of the set of inferred transcripts with their proportions on a patient sample. Columns represent genomic regions, while rows correspond to transcripts. The names of the genomic regions follow the rules explained in figure 6.7, with an additional rule that exons that are continuously included in all transcripts are merged (exons 17 to 23 are merged into exons17-23 for example). The proportions of the inferred transcripts are shown on the left side of the figure. The structure of the transcripts is color coded: white boxes are associated to spliced regions while dark-blue refers to included regions. Additionally, by comparing the percentage of inclusion or splicing of each genomic region to the wild-type distribution (similarly to the procedure explained in section 6.2.5), abnormal events are labelled. Transcripts that differ only by wild-type events (such as the splicing of exon10b) are merged into a single structure with light-blue boxes pointing out the existing variations among them. This specific example underlines an abnormal splicing event as well as an abnormal retention event.

We developed a method to infer the full-length transcripts and their abundances, extending techniques designed for bulk RNA-seq (Xia et al., 2011; Li et al., 2011b,a; Mezlini et al., 2013; Behr et al., 2013; Bernard et al., 2014) to our amplicon sequencing data. Our method, based on sparse regularized regression, comes as a companion to the region level study in order to potentially reveal interesting combination of splicing events and is explained in more details in the Method section 6.4.4. In short, we formulate a convex optimization problem with sparsity constraints that can be efficiently solved to estimate a set of transcripts together with their proportions that explain well the observed amplicon data if their were captured with the given amplicon design. Of note, we also provide a user-friendly visualization of the inferred transcripts, with an automatic highlighting of abnormal events, so that geneticists rapidly spot non-physiological situations. We illustrate our transcript visualization in figure 6.10.

A focus on the ENIGMA cell line

A recent study from the ENIGMA consortium (de la Hoya et al., 2016) analyzed in depth the splicing pattern of exons 8 and 9 in lymphoblastoid cell lines carrying both mutations at the acceptor site of intron 8 and inside exon 9 (mutation *BRCA1*c.[594−2A>C; 641A>G] using the HGVS nomenclature (den Dunnen and Antonarakis, 2000)). They documented both a splicing of exons 8 and 9 together (which is naturally occurring), and an abnormal skipping of exon

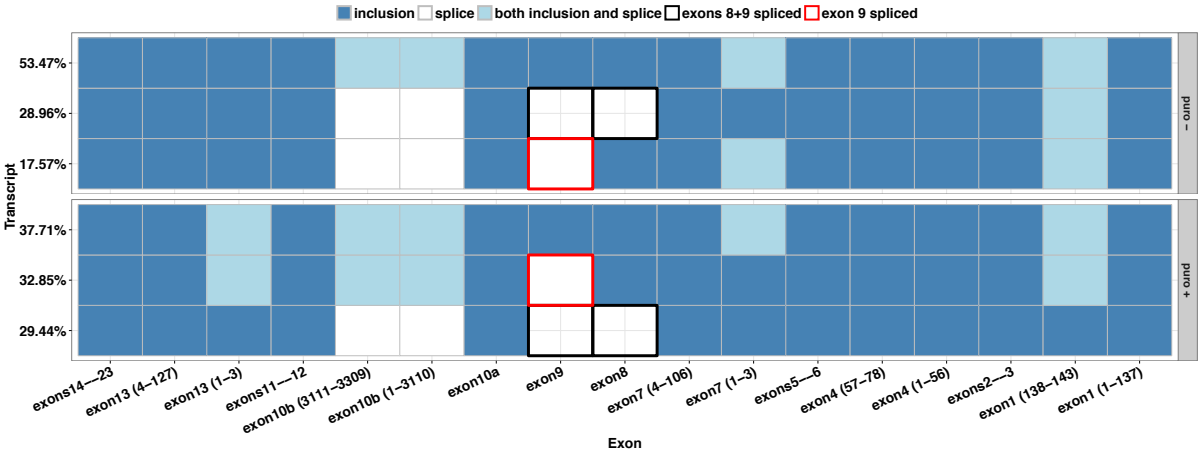


FIGURE 6.11: Transcripts inferred on the ENIGMA cell line. The upper panel corresponds to the puromycin- condition and the lower panel to the puromycin+ condition. Visualization follows the rules explained in figure 6.10. The splicing of exons 8+9 is shown with a thick black contour line, while the splicing of exon 9 alone is underlined with a thick red contour line.

9 alone. This pattern is of much interest as it corresponds to a deconvolution of overlapping splicing events.

We analyzed the same cell line with our amplicon-based targeted RNA-seq approach, and present the set of estimated transcripts in figure 6.11. We find an in-frame splicing of exons 8+9 both in the presence or absence of puromycin at a similar level of $\sim 29\%$. Out-of-frame skipping of exon 9 alone is also detected in both conditions, but (unsurprisingly) at a higher rate when cells are treated with puromycin, with $\sim 18\%$ in puromycin- and $\sim 33\%$ in puromycin+. Transcripts including both exons 8 and 9 are then estimated at a level of $\sim 53\%$ in puromycin- and $\sim 38\%$ in puromycin+.

Remarkably, our estimated proportions are very close to the ones that de la Hoya et al. (2016) reported using a totally different approach, namely capillary electrophoresis of RT-PCR products with appropriate primer design. Indeed, there reported $\sim 29\%$ of splicing of exons 8+9, $\sim 38\%$ of splicing of exon 9, and $\sim 31\%$ of full-length transcripts in the presence of puromycin, which is very close to our (29%, 33%, 38%) estimates. This finding is a proof of concept that our method is able to reconstruct transcripts with a complex splicing landscape and to infer accurate proportions.

6.3 Conclusion

We developed a methodology that uses amplicon sequencing data from targeted single-gene RNA-seq experiments to query splicing abnormalities in a clinical context. We provide a two-layer analysis by estimating local splicing events on each amplicon individually and by predicting full-length transcripts with associated proportions. On the one hand, the local estimation detects exon skipping or splice donor/acceptor shift with high sensitivity, and our methodology has shown to accurately quantify both physiological and abnormal splicing events. On the other hand, the transcript prediction step might help to decipher complex splicing patterns, such as overlapping splicing events. As a proof of concept we presented a transcript prediction on a recently studied cell line characterized by a *BRCA1* splicing of both exons 8+9 and exon 9. Being able to accurately estimate the proportions of these distinct events illustrates the clinical importance of our method as de la Hoya et al. (2016) showed that a physiological level of $20-30\%$ of transcripts lacking exons 8 and 9 might ensure enough tumor suppression function. Importantly, our analysis come with user-friendly outputs (ordered tables and graphs where abnormalities are hightlighted) so that geneticists promptly spot non-physiological events, and with an open-source and hands-on bioinformatics pipeline.

A further line of research would be to combine single-molecule long-read sequencing and high-throughput short-read sequencing to detect transcript structure and quantify proportions in similar targeted experiments. Testing our approach on other disease genes with other amplicon designs is also an appealing future research plan.

6.4 Methods

6.4.1 RNA isolation and sequencing

Patient genomic sequences were screened to assess the presence of VUS using DNA extracted from whole-blood sample. RNA was isolated from lymphoblastoid cell lines treated with and without puromycin. RNA was reverse-transcribed and amplified with long-range PCR. The design of the 3 overlapping amplicons is explained in figure 6.1. Primer sequences are detailed in table B.4. Amplicon RT-PCR products were fragmented at 200bp in average, barcoded, amplified and sequenced with a Ion Torrent Personal Genome Machine (Rothberg et al., 2011).

Two barcodes were used per sample in order two distinguish from which amplicon reads come from in regions where amplicon overlap.

6.4.2 Bioinformatics pre-processing

Targeted RNA-seq reads are pre-processed with standard bioinformatics procedures. Reads are mapped to the reference hg19 human genome with the splice aligner TopHat2 (Kim et al., 2013). Raw 5' read counts (as shown in figure 6.2) are calculated using BEDTools facilities (Quinlan and Hall, 2010) on the set of *BRCA1* annotated exons downloaded from the UCSC genome browser (Karolchik et al., 2004). We processed the mapped reads to form a *de novo* reconstruction of the expressed regions (any sub-parts of exons or introns) with associated bin counts (see section 6.4.4) with a method similar to the *processsam* program developed by Li et al. (2011b), and also used in Li and Jiang (2012b); Bernard et al. (2014); Maretty et al. (2014).

6.4.3 Data normalization

In section 6.2.3, we briefly described our normalization technique and showed some effects of the normalization on control and patient amplicon data (figures 6.5 and 6.6). The goal of the normalization is to alleviate artificial coverage trends while preserving coverage drops due to splicing signals. In that context, estimating scaling factors based on locally fitted coverage curves is natural, as a smooth fit should capture the main trends but would not dramatically fall with spliced base pairs. The detail of our normalization procedure based on the available controls, also illustrated in figure 6.12, is the following:

- average the raw 5' read counts so that the number of points is equal in each region.
- perform a loess fit on the averaged data points.
  - the smoothing parameter, that controls how local the fit is, is a parameter of our method (we used a default value of 0.5).
  - additionally, one can include prior knowledge to the procedure by giving a lower weight to data points from regions that are known to be physiologically spliced (in our case we gave a lower weight of 0.1 to data points from exon 10b).
- associate a value to each region as the mean of the fitted points.

- calculate a scaling factor per region as the ratio between the maximum value across regions over the region value.
- average the scaling factor values across all available controls.

When analyzing patient data, the counts from reads that come from a specific region are multiplied by the associated scaling factors. As illustrated in figure 6.6, artificially low coverage regions are scaled by a factor > 1, while low counts from abnormally spliced regions are not intensively scaled. Figure 6.12 additionally illustrates our loess-based normalization procedure on a given control. We clearly see that while large trends are captured with a smooth fit, for instance the decreasing trend of amplicon 1, wild-type splicing signals, such as exon 10b in amplicon 2, are preserved.

Finally, note that when performing the full-length transcript prediction step (see section 6.4.4) we add an additional normalization layer by scaling the amplicon data against each other for a given sample. To achieve it we simply calculate a scaling factor per amplicon such that the maxima of mean 5' read count across the regions overlapping the amplicons are equal.

6.4.4 Transcript prediction

In that section, we detail our algorithm to estimate both transcript structures and proportions from amplicon sequencing data. We first clarify the amplicon problem formulation, that is how to select a few transcripts that jointly explain the observed amplicon data, and then explain our penalized regression technique.

Amplicon problem formulation

Consider a gene with B regions, where we remind that we call a *region* any possible sub-part of exon or intron (regions are defined *de novo* in a pre-processing step of the sequencing data using the junction reads). In a targeted RNA-seq experiment, two regions $1 \leq i < j \leq B$ are selected a priori. They schematically serve as anchors to capture all mRNA in a sample containing both of them, and amplify the regions between them by RT-PCR. The amplified regions, called *amplicons*, are then sequenced by RNA-seq. An amplicon can hence be formally defined as a pair (i, j) of regions from the gene of interest. Depending on the experimental design, the number and positions of such amplicons vary. Of course, the choice of the positions

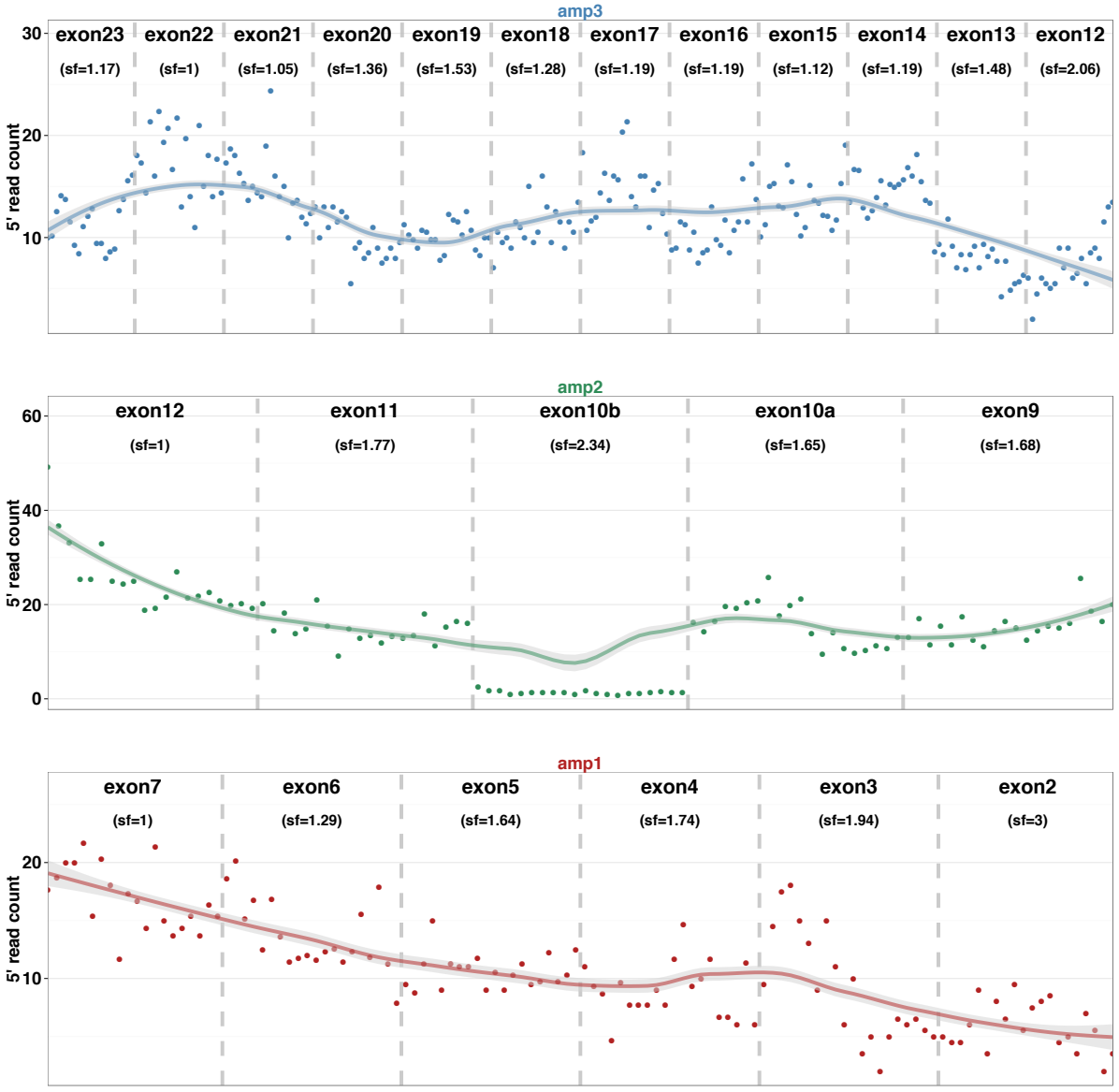


FIGURE 6.12: Illustration of the loess-based normalization procedure on a control sample. Raw 5' read counts are averaged so that the number of points is the same in each region (20 points here). A smooth loess curve is fitted on each amplicon (we used a smoothing parameters equal to 0.5 and gave a lower weight of 0.1 to data points belonging to exon 10b). The mean of smoothly fitted points are attributed to each region, and scaling factors are further calculated as the ratio between the maximum value across regions over the region values. Scaling factors, denoted as "sf", are shown into brackets in each region.

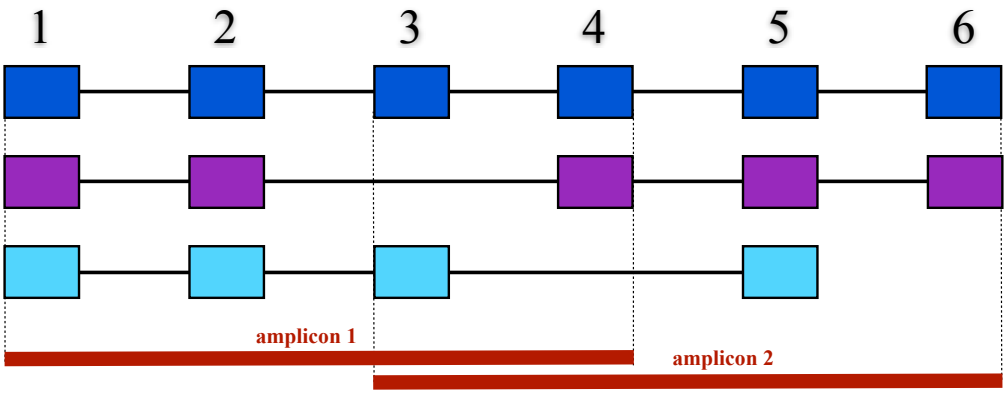


FIGURE 6.13: Schematic design with 2 amplicons and a 6 exons gene with 3 alternative transcripts. Color boxes represent exons and thick red lines correspond to the amplicon regions. Dotted vertical lines highlight the bin pairs defining the amplicons: pair $(1, 4)$ defines the position of amplicon 1 while pair $(3, 6)$ the position of amplicon 2. Given the amplicon design, the dark blue transcript is full-length captured by the two amplicons, the purple transcript is partly captured by only the first amplicon, and the cyan transcript is not captured at all.

of the amplicons constraints the mRNA landscape that can be sequenced. Some mRNA can be captured by all amplicons, while others can be partially captured or can escape to the design. Figure 6.13 illustrates such a situation, where a 2-amplicon design leads to a full sequencing of one transcript and to a partial sequencing of a second one, but misses a third one.

We suppose in the following that Q amplicons are used, with specific pairs $(i_1, j_1), \ldots, (i_Q, j_Q)$ determining their positions on the gene. For each amplicon we observed the number of reads falling into each one of the B regions or overlapping different regions (junction reads). We call a *bin* any ordered set of regions, such that each read is assigned to a unique bin corresponding to the exact set of regions that it overlaps. Amplicon sequencing data can hence be summarized as a count value per bin. The count value is by construction equal to 0 when a bin is located outside of the amplicon region. The goal is to find a set of isoforms together with their proportions that, if sequenced with the Q amplicons design, would have generated the observed count values. We expect the set of isoforms to be relatively sparse and to find isoforms that may explain the count data jointly over the amplicons.

More formally, to each amplicon q defined by a region pair (i_q, j_q) is associated a read count vector $\mathbf{y}_q$ that corresponds to the number of reads falling into each bin contained in the amplicon. Each $\mathbf{y}_q$ is a vector in $\mathbb{R}_+^{|V_q|}$ where V_q is the list of the amplicon bins. We suppose that we have access to a list of $\mathcal{P}$ candidate isoforms. One candidate $p \in \mathcal{P}$ is defined as its sequence of bins. How to generate this candidate list is explained below in the "design-compatible candidates" sub-section. We say that a given candidate is *compatible* with amplicon q if it contains the pair

(i_q, j_q). A compatible candidate can therefore, if selected with a non-zero proportion, participate to explaining the read counts of the amplicon. We define an proportion vector $\boldsymbol{\beta}$ of size $|\mathcal{P}|$, such that each component β_p corresponds to the proportion of the specific candidate p. We wish to estimate $\boldsymbol{\beta}$.

Sparse regression

To estimate the isoform proportion vector $\boldsymbol{\beta}$, we use a similar technique as one used for bulk RNA-seq, namely penalized regression (Xia et al., 2011; Li et al., 2011b,a; Mezlini et al., 2013; Behr et al., 2013; Bernard et al., 2014). When using only one amplicon ($Q = 1$), and assuming that the list of candidate isoforms $\mathcal{P}$ is compatible with the given amplicon, the estimation boils down to the previously studied RNA-seq isoform deconvolution problem (see chapter 4). It corresponds to estimate $\boldsymbol{\beta}$ through the following optimization:

$$\min_{\boldsymbol{\beta}} \quad \mathcal{L}(\boldsymbol{\beta}) + \lambda \| \boldsymbol{\beta} \|_1 \quad \text{such that} \quad \beta_p \geq 0 \quad \text{for all } p \in \mathcal{P} \ , \tag{6.1}$$

where $\mathcal{L}$ is a convex smooth loss function quantifying how well the selected isoforms explain the read counts, and λ is a non-negative regularization parameter that controls the trade-off between loss and sparsity. The ℓ_1-norm $\| \boldsymbol{\beta} \|_1 = \sum_{p \in \mathcal{P}} |\beta_p|$ has indeed a sparsity inducing effect (Tibshirani, 1996; Bach et al., 2012a), promoting solutions where many β_p are set to 0, see section 3.3.2.

When using several amplicons, we wish to select isoforms that simultaneously explain the count data over all the amplicons. Therefore we jointly model the amplicons and extend (6.1) with

$$\min_{\boldsymbol{\beta} \geq 0} \quad \sum_{q=1}^{Q} \mathcal{L}_q(\boldsymbol{\beta}) + \lambda \| \boldsymbol{\beta} \|_1 \ , \tag{6.2}$$

where $\mathcal{L}_q$ is an amplicon-specific loss function. The main difference compared to standard RNA-seq is that a selected isoform participates to explaining the data of a given amplicon only if compatible. Each term $\mathcal{L}_q$ should quantify how well the selected isoforms compatible with amplicon q explain the count data on the $|V_q|$ bins of the amplicon. Hence $\mathcal{L}_q$ should measure the distance on each bin between the observed read count and the sum of the transcript expression levels that are both compatible with amplicon q and contain the given bin. Formally this give

for $\mathcal{L}_q$ the following definition

$$\mathcal{L}_q(\boldsymbol{\beta}) = \sum_{v \in V_q} D\Big(\; y_q^v \; , \; l_v \sum_{\substack{p \in \mathcal{P} \\ p \ni (i_q, v, j_q)}} \beta_p \; \Big) \; ,$$

where $D(.,.)$ is a discrepancy measure between two scalars, y_q^v is the read count on bin v from amplicon q, and l_v is simply a bin factor such as the effective length of bin v, as defined in 4.2.1 and figure 4.1. If using a Euclidean distance measure we have $D(y,z) = \frac{1}{2}(y-z)^2$, whereas if the loss is derived from a Poisson negative likelihood, we have $D(y,z) = z - y \log z$, the choice we made as the Poisson model has been successfully used in several RNA-seq studies (Xia et al., 2011; Bernard et al., 2014; Jiang and Wong, 2009; Salzman et al., 2011; Bernard et al., 2015).

Finally a re-fitting and a model selection steps are performed to report an ultimate solution with a good level of regularization, *i.e.*, with a reasonable trade-off between explaining the observed count data and selecting a few number of transcripts. In practice we solve (6.2) for a large range of regularization parameter λ values, obtaining solutions from very sparse to more dense. Each solution, *i.e.* each set of selected transcripts obtained with a particular λ value, is then re-fitted, that proportions are attributed to the selected transcripts to minimize the loss function but without any sparsity penalization, so that the estimated proportions do not suffer from shrinkage (Tibshirani, 1996). Among all re-fitted solutions, the one with the largest BIC criterion (Schwarz, 1978) is finally selected as the preferred solution.

Note also that for a *local* estimation of the percentages of splicing or retention events as presented in sections 6.2.4 and 6.2.5, *i.e.*, without trying to infer the combinations of events into several transcripts, we simply solve (6.1) on each amplicon without any penalization (with $\lambda = 0$). In that way we fit a very complex model that is very sensitive and accurate in quantifying each possible splicing event.

Design-compatible candidates

The transcript inference performed via the optimization problem (6.2) is written over a set $\mathcal{P}$ of candidate transcripts. This candidate set has to be generated in such a way that it respects the amplicon design, *i.e.* such that the transcripts are compatible with one or more amplicons. Moreover as $|\mathcal{P}|$ grows exponentially with the number of regions (expressed sub-parts of exons or introns) of the gene of interest, we need to avoid an exhaustive enumeration of all candidates

$p \in \mathcal{P}$. We resort to a two-step procedure to generate the candidate set. We first generate candidates when restricting the problem to individual amplicons. When considering a given amplicon q we use techniques described in Bernard et al. (2014) and in chapter 4 –namely network flow optimization strategies– to solve (6.1) without need for exhaustive enumeration. We solve (6.1) for different values of λ and take the union of the selected transcripts as *short* candidates, *i.e.*,transcripts that are delimited by the primer pair (i_q, j_q) of the q amplicon. We then generate longer transcripts by appropriately connecting short transcripts as follows: we merge short transcripts generated from distinct amplicons that share the same structure in their overlap region. Such a procedure efficiently generates candidate transcripts that are all compatible with one (*short* candidates) or more (*long* candidates) amplicons.

Chapter 7

Discussion

In this thesis, we contributed to the fields of transcriptome assembly and alternative splicing events quantification from both methodological and clinical diagnosis perspectives.

First, we introduced a new method to detect and quantify alternative transcripts from RNA-seq data. The novelty of our approach is to translate a ℓ_1-penalized maximum likelihood estimation into a network flow optimization problem that can be solved efficiently. We postulate that our approach could be further improved by incorporating prior knowledge into the graph model. Information derived for instance from CAGE-seq[1] data or sequence polyadenylation signals could be used to give different weights to the nodes of the graph that correspond to transcription starting sites or polyadenylation sites. Our method called FlipFlop is implemented in an *R* package available from the Bioconductor website[2].

Second, we developed a multi-sample approach where we proposed to solve the isoform deconvolution jointly over several samples. By doing so, we share information across samples and partially resolve the low coverage issue. We believe that it would be fruitful to test the performances of other norms that lead to group-sparse patterns. Our multi-sample approach is also implemented in the FlipFlop package.

Finally, we examined means to explore the transcriptomic landscape of genes of interest in a clinical diagnosis context. We proposed a time- and cost-efficient experimental protocol to amplify and sequence regions of interest, and developed a methodology to query splicing abnormalities from targeted RNA-seq data. We tested our method on lymphoblastoid cell lines derived from

[1] CAGE (cap analysis of gene expression) sequencing is a method to sequence the 5' ends of mRNAs
[2] http://bioconductor.org/packages/release/bioc/html/flipflop.html

blood samples of patients harbouring mutations in their *BRCA1* gene, and experimentally validated some of our results. We plan to investigate the results of the method on fresh blood samples as this would more accurately represent the *in vivo* situation.

In addition to possible improvements of the methods cited above, we below describe some potential extensions of the work presented in the thesis to other molecular biology problems and different data:

- **complex genomic rearrangements.** Cancer genomes are often characterized by complex rearrangements with multiple genomic breakpoints and copy number alterations.

  The problem of estimating the chromosomal structures and copy numbers of a cancer genome contains several formal similarities to that of estimating transcript isoform abundances. Indeed, like splicing graphs derived from RNA-seq data, "breakpoint graphs" can be constructed from whole-genome shotgun sequencing data (McPherson et al., 2012). In such graphs, edges represent genomic breakpoints and vertices correspond to continuous genomic positions so that a read count value is associated with each vertex. In that case, the read count value can be seen as the copy number of a given region. A path in the breakpoint graph with an associated abundance then represents a putative tumor chromosome and its copy number. It is therefore tempting to apply a network flow methodology over the breakpoint graph to estimate the set of re-arranged chromosomes and their copy numbers.

  Zerbino et al. (2013) recently studied the equivalence between copy-numbers in breakpoint graphs and flows, but they parsimoniously decompose a flow using an ergodic sampling strategy. We believe that mapping a ℓ_1-penalized estimation into a network flow problem could be a valuable approach in the field a genomic rearrangement assembly.

- **single-cell RNA-seq.** The RNA-seq experiments that we described and used through this thesis were performed at the tissue level, that is when a collection of cells are sequenced collectively. Thus, heterogeneity between individual cells is not accessible to standard RNA-seq protocols. Single-cell RNA-seq (scRNA-seq) is a recent experimental technique that allows to sequence a given transcriptome at the level of individual cells (Navin, 2015; Gawad et al., 2016). By doing so, scRNA-seq might reveal the variability among cells of the same type, help to characterize tumoral sub-clones and bring us closer to the use of RNA measurements for clinical diagnosis, for example by sequencing circulating tumor cells (Navin, 2015).

However, scRNA-seq is characterized by specific data analysis challenges. The main issue when analyzing scRNA-seq data is the low read coverage. In particular, one striking phenomenon is the so-called "drop-out" event (Kharchenko et al., 2014), where some lowly expressed transcripts are not sequenced in a subset of cells due to technical reasons. Given that scRNA-seq experiments are usually performed on tens or hundreds of cells, methods that use several samples simultaneously might help to resolve the coverage issue and increase the statistical power of inference procedures. We speculate that the approach we presented in chapter 5 might be useful to the inference of single-cell transcript abundances, since it allows information sharing across samples within the framework of a group-sparse regression.

- **long-read sequencing.** Emerging sequencing techniques, sometimes called third-generation sequencing, are capable of sequencing single molecules, bypassing the need of amplification, and produce long-reads up to several kilobases (Eid et al., 2009). Long-reads provide crucial information on the structure of the full-length transcripts. In chapter 4, we described a splicing graph model that encompasses long-read information by capturing the connectivity between several exons, potentially more than two. It should be worth trying to incorporate the structural information provided by long-read sequencing into our graph model.

  Moreover, third-generation sequencers typically produce low read coverage. Hybrid methods that combine long-read information with short-read counts might therefore be valuable. We could create bins in our graph that encode the long-read structures and appropriately associate a count value with each bin from the short-reads.

  A recent method (Au et al., 2013) combined analysis of short- and long-reads to characterize a transcriptome. It uses a maximum a posteriori procedure where the prior distribution depends on the long-read information. However, it still relies on an exhaustive enumeration of the candidate transcripts and arbitrarily restricts the set of candidates to 50 transcripts. Using efficient network flow techniques over an appropriate graph structure that encodes long-read information without arbitrary restrictions seems a valuable future direction.

As explained above, third-generation sequencing techniques are capable of sequencing a single transcript to its full length. The need for transcriptome assembly will therefore probably disappear when this technology reaches a throughput similar to second-generation sequencing.

Martin and Wang (2011) conclude their review on next-generation transcriptome assembly by underlining that *hopefully, the future of transcriptome assembly will be "no assembly required"*. One may therefore question the value of the work presented in this thesis. Why not simply wait for long-read sequencing to reveal the full-length structure of the RNA transcripts?

To respond to this valid question we propose the following considerations. Sequencing technologies and algorithms are highly dependent on one another. An advance in one quickly results in commensurate progress in the other. New technologies give access to better quality biological information just as algorithms need to follow and support technological progress. Our contribution enables a more efficient analysis of today's RNA sequencing data, but is structurally doomed to obsolescence.

However, we believe that the core value of our work lies in the application of advanced mathematical methods to molecular biology problems. Through our efforts in adapting state-of-the art statistical tools to a specific biological problem, we have delivered results that contribute to and increase the current knowledge of our community. The rise of new technologies will bring new problems and new algorithmic challenges, in the genomic field and others. Hopefully, we will not have to start from scratch, but rather by building on our contributions among others we will be equipped to tackle the next generation of challenges in the field.

Appendix A

Supplementary figures

In this appendix, we provide additional figures that we refer to in the main chapters of the thesis.

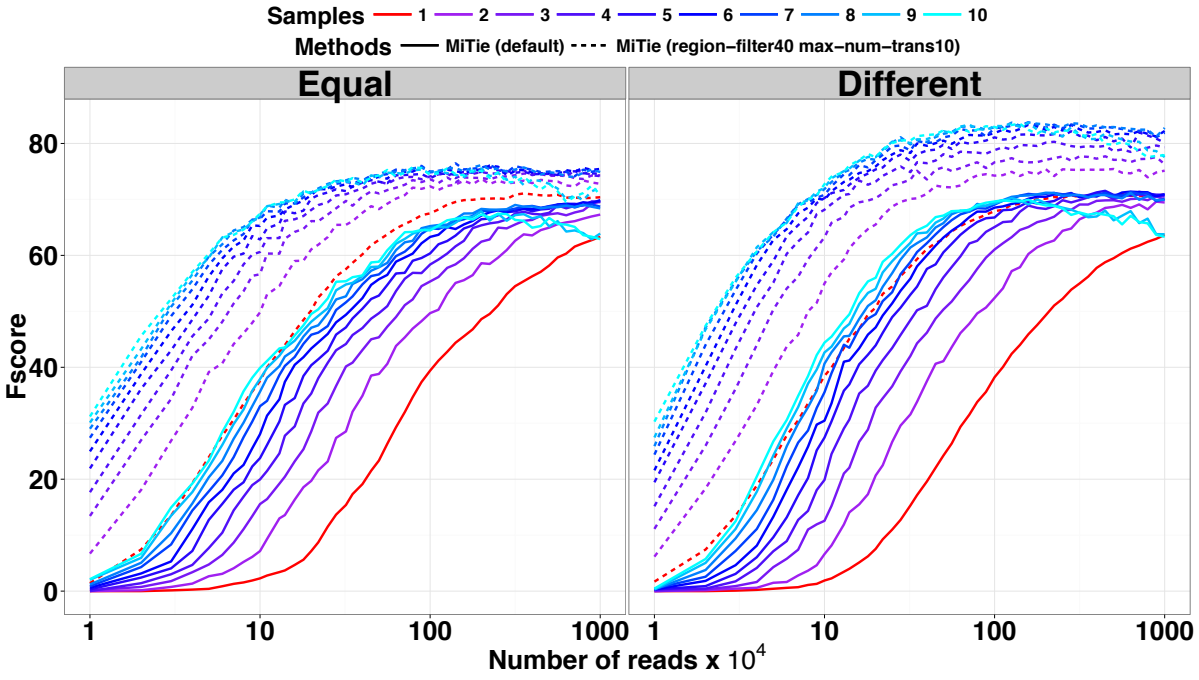


FIGURE A.1: MiTie results on a first set of human simulations when using default parameters or setting *region-filter* to 40 and *max-num-trans* to 10.

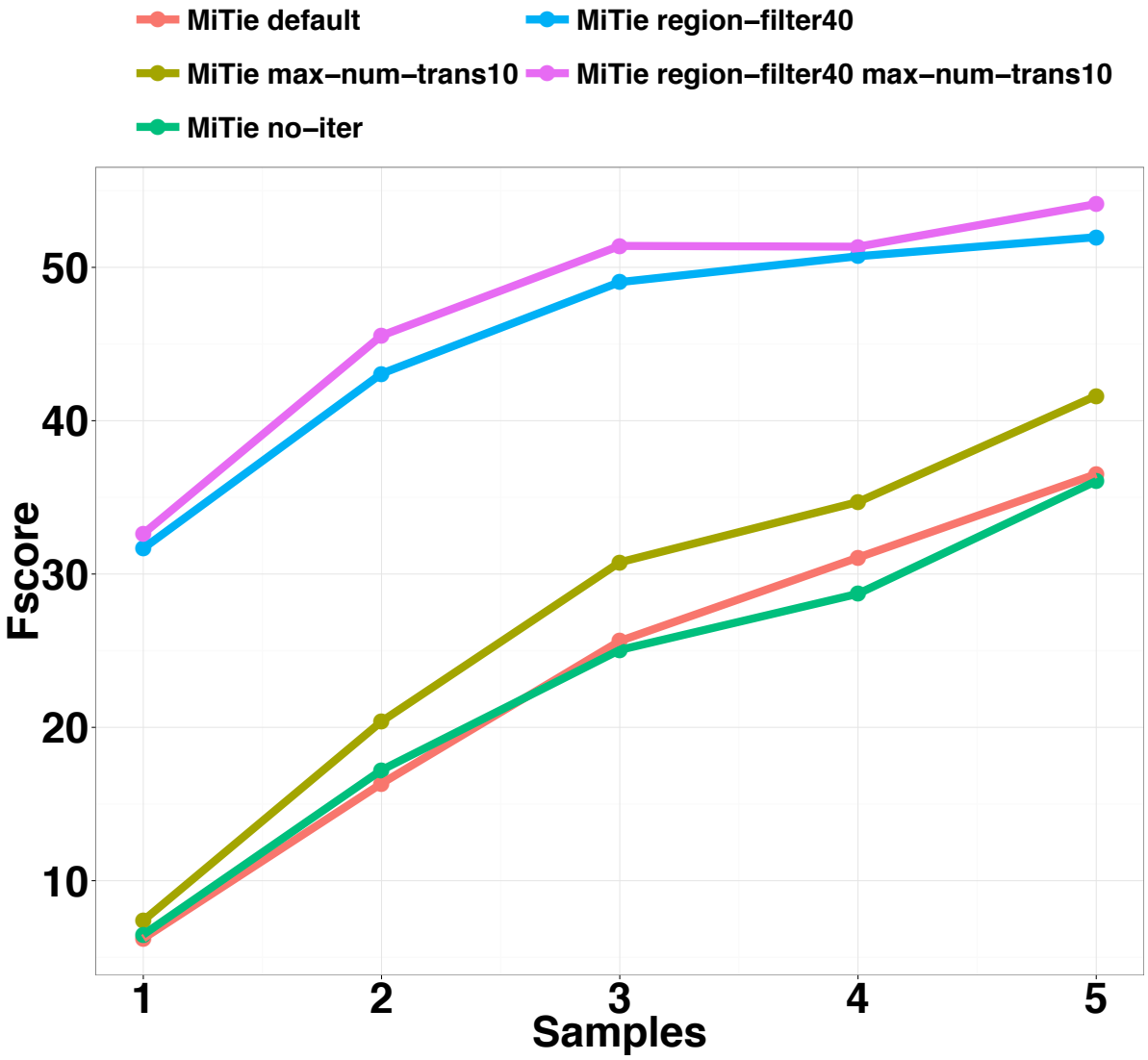


FIGURE A.2: MiTie results on a second set of human simulations when varying some parameters.

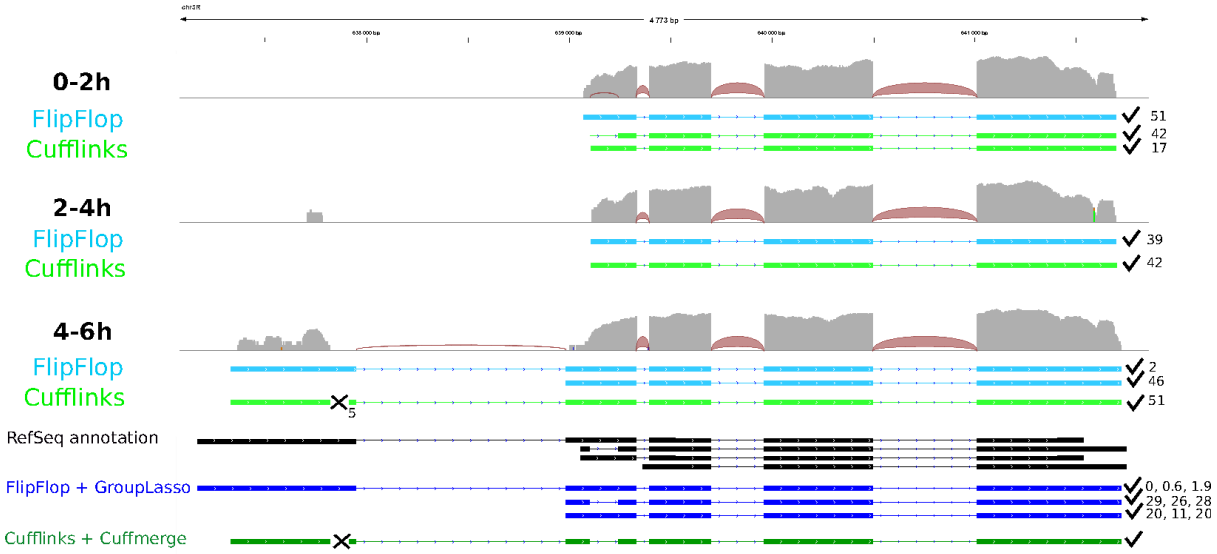


FIGURE A.3: Transcriptome predictions of gene CG1129 from 3 samples of the modENCODE data. Samples name are 0-2h, 2-4h and 4-6h. Each sample track contains the read coverage (light grey) and junction reads (red) as well as FlipFlop predictions (light blue) and Cufflinks predictions (light green). Here coverage is log-scale. The bottom of the figure displays the RefSeq records (black) and the multi-sample predictions of the group-lasso (dark blue) and of Cufflinks/Cuffmerge (dark green). Symbols ✓ and ✗ indicate if a predicted transcript matches a RefSeq record of not. Estimated abundances in FPKM are given on the right hand side of each transcript.

Figure A.3 illustrates that our group-lasso approach can be more powerful than individual predictions and than the merging strategy of Cuffmerge. Indeed, when using evidences from several samples (both junctions and coverage discrepancies) our approach finds a lowly expressed transcript (that was found in only 1 sample with individual predictions), and two well expressed transcripts, including one that was not previously found with individual predictions. On the other hand, Cufflinks/Cuffmerge is very conservative and only predicts a long transcript that does not explain the variations of coverage from the left to the right part of the gene.

Appendix B

Supplementary tables

In this appendix, we provide additional tables that we refer to in the main chapters of the thesis.

Methods	Pre-processing parameters (with default values)	Optimal values for each number of samples 1	2	3	4	5
MiTie	region-filter (1000)	50	50	50	50	10
	seg-filter (0.05)	0.01	0.01	0.01	0.01	0.01
	tss-tts-pval (10^{-4})	6×10^{-5}	6×10^{-5}	2×10^{-5}	6×10^{-5}	6×10^{-5}
Cufflinks	min-frags-per-transfrag (10)	29	17	17	17	29
	max-multiread-fraction (0.75)	0.15	0.15	0.15	0.15	0.15
	overlap-radius (50)	146	85	85	85	146
FlipFlop + Merge	minReadNum (40)	23	40	23	8	14
	minJuncCount (1)	1	1	1	1	1
	minCvgCut (0.05)	0.02	0.03	0.01	0.01	0.01
FlipFlop + GroupLasso	minReadNum (40)	23	40	23	8	14
	minJuncCount (1)	1	1	1	1	1
	minCvgCut (0.05)	0.02	0.01	0.01	0.01	0.01

TABLE B.1: Details on the optimized pre-processing parameters.

Methods	Prediction parameters (with default values)	Optimal values for each number of samples 1	2	3	4	5
MiTie	max-num-trans (2)	5	5	10	10	10
	C-exon (10)	29	50	17	50	29
	C-intron (100)	100	20	58	292	171
	C-num-trans (100)	20	20	20	20	34
Cufflinks	min-isoform-fraction (0.10)	0.02	0.03	0.02	0.02	0.02
	pre-mrna-fraction (0.15)	0.08	0.08	0.03	0.03	0.03
	junc-alpha (10^{-3})	2×10^{-4}	2×10^{-4}	2×10^{-4}	2×10^{-4}	2×10^{-4}
FlipFlop + Merge	BICcst (50)	10	50	50	85	50
	cutoff (1)	0	1	1	3	3
	delta (10^{-7})	10^{-11}	10^{-11}	10^{-10}	10^{-10}	10^{-11}
FlipFlop + GroupLasso	BICcst (50)	10	29	29	50	50
	cutoff (1)	0	0	0	0	1
	delta (10^{-7})	10^{-11}	10^{-9}	10^{-10}	10^{-10}	10^{-10}

TABLE B.2: Details on the optimized prediction parameters.

Sample descriptions	SRA accession names	Total number of reads mapped on the reference transcriptome
0-2h embryos	SRR023659 SRR023755 SRR023671 SRR023663 SRR023747	25 388 344
2-4h embryos	SRR023722 SRR023745 SRR023705 SRR023660	24 541 397
4-6h embryos	SRR023746 SRR023836 SRR023696 SRR023669 SRR035220	46 722 946
6-8h embryos	SRR023691 SRR023732 SRR023654 SRR023668 SRR024217	32 231 644
8-10h embryos	SRR023754 SRR023657 SRR023749 SRR023701 SRR023759 SRR024219 SRR023750	29 544 727

TABLE B.3: Description of the *D.melanogaster* RNA-seq data from the modENCODE project. Data can be found at the following adress: http://intermine.modencode.org/query/experiment.do?experiment=Developmental+Time+Course+Transcriptional+Profiling+of+D.+melanogaster+Using+Illumina+poly%28A%29%2B+RNA-Seq

Amplicon	**Exon**	**Position**	**Sequence 5'>3'**	**Property**
amp1	5' UTR	c.-162>c.-143	GCGCGGGAATTACAGATAAA	20bp, Tm: 60.1°C, GC: 45%
	exon 7	c.499>c.518	GGTTGTATCCGCTGCTTTGT	20bp, Tm: 60.1°C, GC: 50%
amp2	exon 6	c.422>c.441	AACCCGAAAATCCTTCCTTG	20bp, Tm: 60.3°C, GC: 45%
	exon 12	c.4322>c.4341	TTGTTCTGGATTTCGCAGGT	20bp, Tm: 60.6°C, GC: 45%
amp3	exon 11	c.4105>c.4124	GCATCTGGGTGTGAGAGTGA	20bp, Tm: 59.8°C, GC: 55%
	3' UTR	c.*519>c.*538	AATTTCCTCCCCAATGTTCC	20bp, Tm: 60.0°C, GC: 45%

TABLE B.4: Primer pairs defining each amplicon on the *BRCA1* study. The "Position" column gives the position of the 5' end of each primer on the *NM_007294* RefSeq transcript, using the HGVS nomenclature (http://www.hgvs.org/). Tm denotes melting temperature.

Appendix C

Software

Bioconductor package

Bioconductor (Huber et al., 2015) is an open-source project that assemble softwares dedicated to genomic analysis.

The methods described in chapters 4 and 5 to infer transcript isoforms from one or several RNA-seq data samples are implemented as an R/Bioconductor package.

The package is called FlipFlop for "Fast Lasso-based Isoform Prediction as a FLOw Problem" and available at `http://www.bioconductor.org/packages/release/bioc/html/flipflop.html`. Tutorials on how to use the package and how to reproduce the results described in Bernard et al. (2014) and Bernard et al. (2015) are also available from `http://cbio.mines-paristech.fr/flipflop/`.

The software package is compatible with Linux, Mac and Windows operating systems. It exploits multi-core CPUs when available.

Bibliography

Adams, M. D., Kelley, J. M., Gocayne, J. D., Dubnick, M., Polymeropoulos, M. H., Xiao, H., Merril, C. R., Wu, A., Olde, B., and Moreno, R. F. (1991). Complementary DNA sequencing: expressed sequence tags and human genome project. *Science*, 252(5013):1651–1656.

Ahuja, R., Natarajan, T., and K.R. Rao, K. (1993). *Network Flows*. Prentice Hall.

Alamancos, G. P., Agirre, E., and Eyras, E. (2014). Methods to study splicing from high-throughput RNA sequencing data. *Methods Mol. Biol.*, 1126:357–397.

Anders, S., Reyes, A., and Huber, W. (2012). Detecting differential usage of exons from rna-seq data. *Genome Res.*, 22:2008–2017.

Ast, G. (2004). How did alternative splicing evolve? *Nat. Rev. Genet.*, 5(10):773–782.

Au, K. F., Jiang, H., Lin, L., Xing, Y., and Wong, W. H. (2010). Detection of splice junctions from paired-end RNA-seq data by SpliceMap. *Nucleic Acids Res.*, 38(14):4570–4578.

Au, K. F., Sebastiano, V., Afshar, P. T., Durruthy, J. D., Lee, L., Williams, B. A., van Bakel, H., Schadt, E. E., Reijo-Pera, R. A., Underwood, J. G., and Wong, W. H. (2013). Characterization of the human ESC transcriptome by hybrid sequencing. *Proc. Natl. Acad. Sci. U.S.A.*, 110(50):E4821–4830.

Bach, F., Jenatton, R., Mairal, J., and Obozinski, G. (2012a). Optimization with sparsity-inducing penalties. *Foundations and Trends in Machine Learning*, 4(1):1–106.

Bach, F., Jenatton, R., Mairal, J., and Obozinski, G. (2012b). Structured sparsity through convex optimization. *Statist. Sci.*, 27(4):450–468.

Barash, Y., Calarco, J. A., Gao, W., Pan, Q., Wang, X., Shai, O., Blencowe, B. J., and Frey, B. J. (2010). Deciphering the splicing code. *Nature*, 465(7294):53–59.

Barbosa-Morais, N. L., Irimia, M., Pan, Q., Xiong, H. Y., Gueroussov, S., Lee, L. J., Slobodeniuc, V., Kutter, C., Watt, S., Colak, R., Kim, T., Misquitta-Ali, C. M., Wilson, M. D., Kim, P. M., Odom, D. T., Frey, B. J., and Blencowe, B. J. (2012). The evolutionary landscape of alternative splicing in vertebrate species. *Science*, 338(6114):1587–1593.

Beck, A. and Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Img. Sci.*, 2(1):183–202.

Behr, J., Kahles, A., Zhong, Y., Sreedharan, V. T., Drewe, P., and Ratsch, G. (2013). Mitie: Simultaneous rna-seq based transcript identification and quantification in multiple samples. *Bioinformatics*, 29(20):2529–2538.

Benson, D. A., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., and Wheeler, D. L. (2005). Genbank. *Nucleic Acids Res.*, 33(Database-Issue):34–38.

Bentley, D. R., Balasubramanian, S., Swerdlow, H. P., Smith, G. P., Milton, J., Brown, C. G., Hall, K. P., Evers, D. J., Barnes, C. L., Bignell, H. R., Boutell, J. M., Bryant, J., Carter, R. J., Keira Cheetham, R., Cox, A. J., Ellis, D. J., Flatbush, M. R., Gormley, N. A., Humphray, S. J., Irving, L. J., Karbelashvili, M. S., Kirk, S. M., Li, H., Liu, X., Maisinger, K. S., Murray, L. J., Obradovic, B., Ost, T., Parkinson, M. L., Pratt, M. R., Rasolonjatovo, I. M., Reed, M. T., Rigatti, R., Rodighiero, C., Ross, M. T., Sabot, A., Sankar, S. V., Scally, A., Schroth, G. P., Smith, M. E., Smith, V. P., Spiridou, A., Torrance, P. E., Tzonev, S. S., Vermaas, E. H., Walter, K., Wu, X., Zhang, L., Alam, M. D., Anastasi, C., Aniebo, I. C., Bailey, D. M., Bancarz, I. R., Banerjee, S., Barbour, S. G., Baybayan, P. A., Benoit, V. A., Benson, K. F., Bevis, C., Black, P. J., Boodhun, A., Brennan, J. S., Bridgham, J. A., Brown, R. C., Brown, A. A., Buermann, D. H., Bundu, A. A., Burrows, J. C., Carter, N. P., Castillo, N., Chiara E Catenazzi, M., Chang, S., Neil Cooley, R., Crake, N. R., Dada, O. O., Diakoumakos, K. D., Dominguez-Fernandez, B., Earnshaw, D. J., Egbujor, U. C., Elmore, D. W., Etchin, S. S., Ewan, M. R., Fedurco, M., Fraser, L. J., Fuentes Fajardo, K. V., Scott Furey, W., George, D., Gietzen, K. J., Goddard, C. P., Golda, G. S., Granieri, P. A., Green, D. E., Gustafson, D. L., Hansen, N. F., Harnish, K., Haudenschild, C. D., Heyer, N. I., Hims, M. M., Ho, J. T., Horgan, A. M., Hoschler, K., Hurwitz, S., Ivanov, D. V., Johnson, M. Q., James, T., Huw Jones, T. A., Kang, G. D., Kerelska, T. H., Kersey, A. D., Khrebtukova, I., Kindwall, A. P., Kingsbury, Z., Kokko-Gonzales, P. I., Kumar, A., Laurent, M. A., Lawley, C. T., Lee, S. E., Lee, X., Liao, A. K., Loch, J. A., Lok, M., Luo, S., Mammen, R. M., Martin, J. W., McCauley, P. G., McNitt, P., Mehta, P., Moon, K. W., Mullens, J. W., Newington, T., Ning,

Z., Ling Ng, B., Novo, S. M., O'Neill, M. J., Osborne, M. A., Osnowski, A., Ostadan, O., Paraschos, L. L., Pickering, L., Pike, A. C., Pike, A. C., Chris Pinkard, D., Pliskin, D. P., Podhasky, J., Quijano, V. J., Raczy, C., Rae, V. H., Rawlings, S. R., Chiva Rodriguez, A., Roe, P. M., Rogers, J., Rogert Bacigalupo, M. C., Romanov, N., Romieu, A., Roth, R. K., Rourke, N. J., Ruediger, S. T., Rusman, E., Sanches-Kuiper, R. M., Schenker, M. R., Seoane, J. M., Shaw, R. J., Shiver, M. K., Short, S. W., Sizto, N. L., Sluis, J. P., Smith, M. A., Ernest Sohna Sohna, J., Spence, E. J., Stevens, K., Sutton, N., Szajkowski, L., Tregidgo, C. L., Turcatti, G., Vandevondele, S., Verhovsky, Y., Virk, S. M., Wakelin, S., Walcott, G. C., Wang, J., Worsley, G. J., Yan, J., Yau, L., Zuerlein, M., Rogers, J., Mullikin, J. C., Hurles, M. E., McCooke, N. J., West, J. S., Oaks, F. L., Lundberg, P. L., Klenerman, D., Durbin, R., and Smith, A. J. (2008). Accurate whole human genome sequencing using reversible terminator chemistry. *Nature*, 456(7218):53–59.

Berget, S. M., Moore, C., and Sharp, P. A. (1977). Spliced segments at the 5' terminus of adenovirus 2 late mRNA. *Proc. Natl. Acad. Sci. U.S.A.*, 74(8):3171–3175.

Bernard, E., Jacob, L., Mairal, J., and Vert, J.-P. (2014). Efficient rna isoform identification and quantification from rna-seq data with network flows. *Bioinformatics*, 30(17):2447–2455.

Bernard, E., Jacob, L., Mairal, J., Viara, E., and Vert, J. P. (2015). A convex formulation for joint RNA isoform detection and quantification from multiple RNA-seq samples. *BMC Bioinformatics*, 16:262.

Bertsekas, D. (1998). *Network Optimization: Continuous and Discrete Models*. Athena Scientific.

Biamonti, G., Bonomi, S., Gallo, S., and Ghigna, C. (2012). Making alternative splicing decisions during epithelial-to-mesenchymal transition (EMT). *Cell. Mol. Life Sci.*, 69(15):2515–2526.

Black, D. L. (2003). Mechanisms of alternative pre-messenger RNA splicing. *Annu. Rev. Biochem.*, 72:291–336.

Blencowe, B. J. (2006). Alternative splicing: new insights from global analyses. *Cell*, 126(1):37–47.

Blencowe, B. J., Ahmad, S., and Lee, L. J. (2009). Current-generation high-throughput sequencing: deepening insights into mammalian transcriptomes. *Genes Dev.*, 23(12):1379–1386.

Bohnert, R. and Rätsch, G. (2010). rQuant.web: a tool for RNA-Seq-based transcript quantitation. *Nucleic Acids Res.*, 38(Web Server issue):W348–W351.

Bonnal, S., Vigevani, L., and Valcarcel, J. (2012). The spliceosome as a target of novel antitumour drugs. *Nat Rev Drug Discov*, 11(11):847–859.

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press, New York, NY, USA.

Bray, N. L., Pimentel, H., Melsted, P., and Pachter, L. (2016). Near-optimal probabilistic RNA-seq quantification. *Nat. Biotechnol.*, 34(5):525–527.

Byron, S. A., Van Keuren-Jensen, K. R., Engelthaler, D. M., Carpten, J. D., and Craig, D. W. (2016). Translating RNA sequencing into clinical diagnostics: opportunities and challenges. *Nat. Rev. Genet.*, 17(5):257–271.

Canzar, S., Andreotti, S., Weese, D., Reinert, K., and Klau, G. W. (2016). CIDANE: comprehensive isoform discovery and abundance estimation. *Genome Biol.*, 17:16.

Cartegni, L., Chew, S. L., and Krainer, A. R. (2002). Listening to silence and understanding nonsense: exonic mutations that affect splicing. *Nat. Rev. Genet.*, 3(4):285–298.

Celniker, E., Dillon, L. A. L., Gerstein, M. B., Gunsalus, K. C., Henikoff, S., Kerpen, G. H., Kellis, M., Lai, E. C., Lieb, J. D., MacAlpine, D. M., et al. (2009). Unlocking the secrets of the genome. *Nature*, 459(7249):927–930.

Chen, L. (2013). Statistical and Computational Methods for High-Throughput Sequencing Data Analysis of Alternative Splicing. *Stat Biosci*, 5(1):138–155.

Chen, M. and Manley, J. L. (2009). Mechanisms of alternative splicing regulation: insights from molecular and genomics approaches. *Nat. Rev. Mol. Cell Biol.*, 10(11):741–754.

Chen, S. S., Donoho, D. L., and Saunders, M. (1998). Atomic decomposition by basis pursuit. *SIAM J. Sci. Comput.*, 20(1):33–61.

Chow, L. T., Gelinas, R. E., Broker, T. R., and Roberts, R. J. (1977). An amazing sequence arrangement at the 5' ends of adenovirus 2 messenger RNA. *Cell*, 12(1):1–8.

Clancy, S. (2008). RNA splicing: introns, exons and spliceosome. *Nature Education*, 1(1):31.

Cleveland, S. and Devlin, S. (1988). Locally weighted regression: An approach to regression analysis by local fitting. *J. Amer. Statist. Assoc.*, 83(403):596–610.

Colombo, M., Blok, M. J., Whiley, P., Santamarina, M., Gutierrez-Enriquez, S., Romero, A., Garre, P., Becker, A., Smith, L. D., De Vecchi, G., Brandao, R. D., Tserpelis, D., Brown, M., Blanco, A., Bonache, S., Menendez, M., Houdayer, C., Foglia, C., Fackenthal, J. D., Baralle, D., Wappenschmidt, B., Diaz-Rubio, E., Caldes, T., Walker, L., Diez, O., Vega, A., Spurdle, A. B., Radice, P., and De La Hoya, M. (2014). Comprehensive annotation of splice junctions supports pervasive alternative splicing at the BRCA1 locus: a report from the ENIGMA consortium. *Hum. Mol. Genet.*, 23(14):3666–3680.

ConsortiumInternational, H. G. S. (2004). Finishing the euchromatic sequence of the human genome. *Nature*, 431(7011):931–945.

Cunningham, F., Amode, M. R., Barrell, D., et al. (2015). Ensembl 2015. *Nucleic Acids Res.*, 43(D1):D662–D669.

Danan-Gotthold, M., Golan-Gerstl, R., Eisenberg, E., Meir, K., Karni, R., and Levanon, E. Y. (2015). Identification of recurrent regulated alternative splicing events across human solid tumors. *Nucleic Acids Res.*, 43(10):5130–5144.

Daubechies, I., Defrise, M., and De Mol, C. (2004). An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics*, 57(11):1413–1457.

David, C. J. and Manley, J. L. (2010). Alternative pre-mRNA splicing regulation in cancer: pathways and programs unhinged. *Genes Dev.*, 24(21):2343–2364.

Davuluri, R. V., Suzuki, Y., Sugano, S., Plass, C., and Huang, T. H. (2008). The functional consequences of alternative promoter use in mammalian genomes. *Trends Genet.*, 24(4):167–177.

de la Hoya, M., Soukarieh, O., Lopez-Perolio, I., Vega, A., Walker, L. C., van Ierland, Y., Baralle, D., Santamarina, M., Lattimore, V., Wijnen, J., Whiley, P., Blanco, A., Raponi, M., Hauke, J., Wappenschmidt, B., Becker, A., Hansen, T. V., Behar, R., Investigators, K., Niederacher, D., Arnold, N., Dworniczak, B., Steinemann, D., Faust, U., Rubinstein, W., Hulick, P. J., Houdayer, C., Caputo, S. M., Castera, L., Pesaran, T., Chao, E., Brewer, C., Southey, M. C., van Asperen, C. J., Singer, C. F., Sullivan, J., Poplawski, N., Mai, P., Peto, J., Johnson, N., Burwinkel, B., Surowy, H., Bojesen, S. E., Flyger, H., Lindblom, A., Margolin, S., Chang-Claude, J., Rudolph, A., Radice, P., Galastri, L., Olson, J. E., Hallberg, E., Giles, G. G., Milne, R. L., Andrulis, I. L., Glendon, G., Hall, P., Czene, K., Blows, F., Shah, M.,

Wang, Q., Dennis, J., Michailidou, K., McGuffog, L., Bolla, M. K., Antoniou, A. C., Easton, D. F., Couch, F. J., Tavtigian, S., Vreeswijk, M. P., Parsons, M., Meeks, H. D., Martins, A., Goldgar, D. E., and Spurdle, A. B. (2016). Combined genetic and splicing analysis of BRCA1 c.[594-2A¿C; 641A¿G] highlights the relevance of naturally occurring in-frame transcripts for developing disease gene variant classification algorithms. *Hum. Mol. Genet.*

den Dunnen, J. T. and Antonarakis, S. E. (2000). Mutation nomenclature extensions and suggestions to describe complex mutations: a discussion. *Hum. Mutat.*, 15(1):7–12.

Dobin, A., Carrie, A., Schlesinger, F., Drenkow, J., Zaleski, C., Sonali, J., Batut, P., Chaisson, M., and Gingeras, T. R. (2013). STAR: ultrafast universal RNA-seq aligner. *Bioinformatics,*, 29(1):15–21.

Douglas, A. G. L. and Wood, M. J. A. (2011). RNA splicing: disease and therapy. *Briefings in Functional Genomics*, 10(3):151–164.

Dvinge, H. and Bradley, R. K. (2015). Widespread intron retention diversifies most cancer transcriptomes. *Genome Med*, 7(1):45.

Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). Least angle regression. *Ann. Stat.*, 32(2):407–499.

Eid, J., Fehr, A., Gray, J., Luong, K., Lyle, J., Otto, G., Peluso, P., Rank, D., Baybayan, P., Bettman, B., Bibillo, A., Bjornson, K., Chaudhuri, B., Christians, F., Cicero, R., Clark, S., Dalal, R., Dewinter, A., Dixon, J., Foquet, M., Gaertner, A., Hardenbol, P., Heiner, C., Hester, K., Holden, D., Kearns, G., Kong, X., Kuse, R., Lacroix, Y., Lin, S., Lundquist, P., Ma, C., Marks, P., Maxham, M., Murphy, D., Park, I., Pham, T., Phillips, M., Roy, J., Sebra, R., Shen, G., Sorenson, J., Tomaney, A., Travers, K., Trulson, M., Vieceli, J., Wegener, J., Wu, D., Yang, A., Zaccarin, D., Zhao, P., Zhong, F., Korlach, J., and Turner, S. (2009). Real-time DNA sequencing from single polymerase molecules. *Science*, 323(5910):133–138.

Ford, L. R. and Fulkerson, D. R. (1956). Maximal Flow through a Network. *Can. J. Math.*, 8:399–404.

Gabut, M., Samavarchi-Tehrani, P., Wang, X., Slobodeniuc, V., O'Hanlon, D., Sung, H. K., Alvarez, M., Talukder, S., Pan, Q., Mazzoni, E. O., Nedelec, S., Wichterle, H., Woltjen, K., Hughes, T. R., Zandstra, P. W., Nagy, A., Wrana, J. L., and Blencowe, B. J. (2011). An alternative splicing switch regulates embryonic stem cell pluripotency and reprogramming. *Cell*, 147(1):132–146.

Garber, M., Grabherr, M. G., Guttman, M., and Trapnell, C. (2011). Computational methods for transcriptome annotation and quantification using RNA-seq. *Nat. Methods*, 8(6):469–477.

Gautheret, D., Poirot, O., Lopez, F., Audic, S., and Claverie, J. M. (1998). Alternate polyadenylation in human mRNAs: a large-scale analysis by EST clustering. *Genome Res.*, 8(5):524–530.

Gawad, C., Koh, W., and Quake, S. R. (2016). Single-cell genome sequencing: current state of the science. *Nat. Rev. Genet.*, 17(3):175–188.

Glaus, P., Honkela, A., and Rattray, M. (2012). Identifying differentially expressed transcripts from RNA-seq data with biological variation. *Bioinformatics*, 28(13):1721–1728.

Goldberg, A. V. (1997). An efficient implementation of a scaling minimum-cost flow algorithm. *J. Algorithm*, 22(1):1–29.

Goldberg, A. V. and Tarjan, R. E. (1988). A new approach to the maximum-flow problem. *J. ACM*, 35(4):921–940.

Goldberg, A. V. and Tarjan, R. E. (1990). Finding minimum-cost circulations by successive approximation. *Mathematics of Operations Research*, 15(3):430–466.

Goodwin, S., McPherson, J. D., and McCombie, W. R. (2016). Coming of age: ten years of next-generation sequencing technologies. *Nat. Rev. Genet.*, 17(6):333–351.

Grabherr, M. G., Haas, B. J., Yassour, M., Levin, J. Z., Thompson, D. A., Amit, I., Adiconis, X., Fan, L., Raychowdhury, R., Zeng, Q., Chen, Z., Mauceli, E., Hacohen, N., Gnirke, A., Rhind, N., di Palma, F., Birren, B. W., Nusbaum, C., Lindblad-Toh, K., Friedman, N., and Regev, A. (2011). Full-length transcriptome assembly from RNA-Seq data without a reference genome. *Nat. Biotechnol.*, 29(7):644–652.

Graveley, B. R., Brooks, A. N., Carlson, J. W., Duff, M. O., Landolin, J. M., Yang, L., Artieri, C. G., van Baren, M. J., Boley, N., Booth, B. W., Brown, J. B., Cherbas, L., Davis, C. A., Dobin, A., Li, R., Lin, W., Malone, J. H., Mattiuzzo, N. R., Miller, D., Sturgill, D., Tuch, B. B., Zaleski, C., Zhang, D., Blanchette, M., Dudoit, S., Eads, B., Green, R. E., Hammonds, A., Jiang, L., Kapranov, P., Langton, L., Perrimon, N., Sandler, J. E., Wan, K. H., Willingham, A., Zhang, Y., Zou, Y., Andrews, J., Bickel, P. J., Brenner, S. E., Brent, M. R., Cherbas, P., Gingeras, T. R., Hoskins, R. A., Kaufman, T. C., Oliver, B., and Celniker, S. E. (2011). The developmental transcriptome of Drosophila melanogaster. *Nature*, 471(7339):473–479.

Griebel, T., Zacher, B., Ribeca, P., Raineri, E., Lacroix, V., Guigo, R., and Sammeth, M. (2012). Modelling and simulating generic rna-seq experiments with the flux simulator. *Nucleic Acids Res.*, 40(20):10073–10083.

Guttman, M., Garber, M., Levin, J. Z., Donaghey, J., Robinson, J., Adiconis, X., Fan, L., Koziol, M. J., Gnirke, A., et al. (2010). Ab initio reconstruction of cell type-specific transcriptomes in mouse reveals the conserved multi-exonic structure of lincrnas. *Nat. Biotech.*, 28(5):503–510.

Harrow, J., Frankish, A., Gonzalez, J. M., Tapanari, E., Diekhans, M., Kokocinski, F., Aken, B. L., Barrell, D., Zadissa, A., Searle, S., Barnes, I., Bignell, A., Boychenko, V., Hunt, T., Kay, M., Mukherjee, G., Rajan, J., Despacio-Reyes, G., Saunders, G., Steward, C., Harte, R., Lin, M., Howald, C., Tanzer, A., Derrien, T., Chrast, J., Walters, N., Balasubramanian, S., Pei, B., Tress, M., Rodriguez, J. M., Ezkurdia, I., van Baren, J., Brent, M., Haussler, D., Kellis, M., Valencia, A., Reymond, A., Gerstein, M., Guigo, R., and Hubbard, T. J. (2012). GENCODE: the reference human genome annotation for The ENCODE Project. *Genome Res.*, 22(9):1760–1774.

Hartman, T., Hassidim, A., Kaplan, H., Raz, D., and Segalov, M. (2012). How to split a flow? In *INFOCOM, 2012 Proceedings IEEE*, pages 828–836.

Hastings, M. L. and Krainer, A. R. (2001). Pre-mRNA splicing in the new millennium. *Curr. Opin. Cell Biol.*, 13(3):302–309.

Hayer, K. E., Pizarro, A., Lahens, N. F., Hogenesch, J. B., and Grant, G. R. (2015). Benchmark analysis of algorithms for determining and quantifying full-length mRNA splice forms from RNA-seq data. *Bioinformatics*, 31(24):3938–3945.

Heber, S. et al. (2002). Splicing graphs and EST assembly problem. *Bioinformatics*, 18(suppl 1):S181–S188.

Hiller, D. and Wong, W. H. (2013). Simultaneous isoform discovery and quantification from RNA-seq. *Stat Biosci.*, 5(1):100–118.

Hofstra, R. M., Spurdle, A. B., Eccles, D., Foulkes, W. D., de Wind, N., Hoogerbrugge, N., Hogervorst, F. B., Boffetta, P., Couch, F., de Wind, N., Easton, D., Eccles, D., Foulkes, W., Genuardi, M., Goldgar, D., Greenblatt, M., Hofstra, R., Hogervorst, F., Hoogerbrugge, N., Plon, S., Radice, P., Rasmussen, L., Sinilnikova, O., Spurdle, A., and Tavtigian, S. V. (2008). Tumor characteristics as an analytic tool for classifying genetic variants of uncertain clinical significance. *Hum. Mutat.*, 29(11):1292–1303.

Houdayer, C. (2011). In silico prediction of splice-affecting nucleotide variants. *Methods Mol. Biol.*, 760:269–281.

Houdayer, C., Caux-Moncoutier, V., Krieger, S., Barrois, M., Bonnet, F., Bourdon, V., Bronner, M., Buisson, M., Coulet, F., Gaildrat, P., Lefol, C., Leone, M., Mazoyer, S., Muller, D., Remenieras, A., Revillion, F., Rouleau, E., Sokolowska, J., Vert, J. P., Lidereau, R., Soubrier, F., Sobol, H., Sevenet, N., Bressac-de Paillerets, B., Hardouin, A., Tosi, M., Sinilnikova, O. M., and Stoppa-Lyonnet, D. (2012). Guidelines for splicing analysis in molecular diagnosis derived from a set of 327 combined in silico/in vitro studies on BRCA1 and BRCA2 variants. *Hum. Mutat.*, 33(8):1228–1238.

Houdayer, C., Dehainault, C., Mattler, C., Michaux, D., Caux-Moncoutier, V., Pages-Berhouet, S., d'Enghien, C. D., Lauge, A., Castera, L., Gauthier-Villars, M., and Stoppa-Lyonnet, D. (2008). Evaluation of in silico splice tools for decision-making in molecular diagnosis. *Hum. Mutat.*, 29(7):975–982.

Howard, B. E. and Heber, S. (2010). Towards reliable isoform quantification using RNA-SEQ data. *BMC Bioinformatics*, 11 Suppl 3:S6.

Huang, S., Zhang, J., Li, R., Zhang, W., He, Z., Lam, T. W., Peng, Z., and Yiu, S. M. (2011). SOAPsplice: Genome-Wide ab initio Detection of Splice Junctions from RNA-Seq Data. *Front Genet*, 2:46.

Huber, W., Carey, J., V., Gentleman, R., Anders, S., Carlson, M., Carvalho, S., B., Bravo, C., H., Davis, S., Gatto, L., Girke, T., Gottardo, R., Hahne, F., Hansen, D., K., Irizarry, A., R., Lawrence, M., Love, I., M., MacDonald, J., Obenchain, V., Ole's, K., A., Pag'es, H., Reyes, A., Shannon, P., Smyth, K., G., Tenenbaum, D., Waldron, L., Morgan, and M. (2015). Orchestrating high-throughput genomic analysis with Bioconductor. *Nature Methods*, 12(2):115–121.

Jean, G., Kahles, A., Sreedharan, V. T., De Bona, F., and Ratsch, G. (2010). RNA-Seq read alignments with PALMapper. *Curr Protoc Bioinformatics*, Chapter 11:Unit 11.6.

Jian, X., Boerwinkle, E., and Liu, X. (2014). In silico tools for splicing defect prediction: a survey from the viewpoint of end users. *Genet. Med.*, 16(7):497–503.

Jiang, H. and Wong, W. H. (2009). Statistical inferences for isoform expression in RNA-Seq. *Bioinformatics*, 25(8):1026–1032.

Johnson, J. M., Castle, J., Garrett-Engele, P., Kan, Z., Loerch, P. M., Armour, C. D., Santos, R., Schadt, E. E., Stoughton, R., and Shoemaker, D. D. (2003). Genome-wide survey of human alternative pre-mRNA splicing with exon junction microarrays. *Science*, 302(5653):2141–2144.

Kalsotra, A. and Cooper, T. A. (2011). Functional consequences of developmentally regulated alternative splicing. *Nat Rev Genet*, 12(10):715–729.

Karolchik, D., Hinrichs, A. S., Furey, T. S., et al. (2004). The ucsc table browser data retrieval tool. *Nucleic Acids Res.*, 32(supp1):D493–D496.

Keren, H., Lev-Maor, G., and Ast, G. (2010). Alternative splicing and evolution: diversification, exon definition and function. *Nat. Rev. Genet.*, 11(5):345–355.

Kharchenko, P. V., Silberstein, L., and Scadden, D. T. (2014). Bayesian approach to single-cell differential expression analysis. *Nat. Methods*, 11(7):740–742.

Kim, D., Pertea, G., Trapnell, C., Pimentel, H., Kelley, R., and Salzberg, S. L. (2013). TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome Biol.*, 14(4):R36.

Kim, E., Magen, A., and Ast, G. (2007). Different levels of alternative splicing among eukaryotes. *Nucleic Acids Res.*, 35(1):125–131.

King, V., Rao, S., and Tarjan, R. (1994). A faster deterministic maximum flow algorithm. *J. Algorithms*, 17(3):447–474.

Koboldt, D. C., Steinberg, K. M., Larson, D. E., Wilson, R. K., and Mardis, E. R. (2013). The next-generation sequencing revolution and its impact on genomics. *Cell*, 155(1):27–38.

Kornblihtt, A. R. (2005). Promoter usage and alternative splicing. *Curr. Opin. Cell Biol.*, 17(3):262–268.

Kornblihtt, A. R., Schor, I. E., Allo, M., Dujardin, G., Petrillo, E., and Munoz, M. J. (2013). Alternative splicing: a pivotal step between eukaryotic transcription and translation. *Nat. Rev. Mol. Cell Biol.*, 14(3):153–165.

Krawczak, M., Reiss, J., and Cooper, D. N. (1992). The mutational spectrum of single base-pair substitutions in mRNA splice junctions of human genes: causes and consequences. *Hum. Genet.*, 90(1-2):41–54.

Lareau, L. F., Brooks, A. N., Soergel, D. A., Meng, Q., and Brenner, S. E. (2007). The coupling of alternative splicing and nonsense-mediated mRNA decay. *Adv. Exp. Med. Biol.*, 623:190–211.

Levin, J. Z., Berger, M. F., Adiconis, X., Rogov, P., Melnikov, A., Fennell, T., Nusbaum, C., Garraway, L. A., and Gnirke, A. (2009). Targeted next-generation sequencing of a cancer transcriptome enhances detection of sequence variants and novel fusion transcripts. *Genome Biol.*, 10(10):R115.

Levin, J. Z., Yassour, M., Adiconis, X., Nusbaum, C., Thompson, D. A., Friedman, N., Gnirke, A., and Regev, A. (2010). Comprehensive comparative analysis of strand-specific RNA sequencing methods. *Nat. Methods*, 7(9):709–715.

Li, B. and Dewey, C. N. (2011). Rsem: accurate transcript quantification from rna-seq data with or without a reference genome. 12:323.

Li, J. and Jiang, H. (2014). Robust estimation of isoform expression with RNA-Seq data. *ArXiv e-prints*.

Li, J., Jiang, H., and Wong, W. H. (2010). Modeling non-uniformity in short-read rates in RNA-Seq data. *Genome Biol.*, 11(5):R50.

Li, J. B., Levanon, E. Y., Yoon, J. K., Aach, J., Xie, B., Leproust, E., Zhang, K., Gao, Y., and Church, G. M. (2009). Genome-wide identification of human RNA editing sites by parallel DNA capturing and sequencing. *Science*, 324(5931):1210–1213.

Li, J. J., Jiang, C.-R., J., B. B., Huang, H., and Bickel, P. J. (2011a). Sparse linear modeling of next-generation mRNA sequencing (RNA-Seq) data for isoform discovery and abundance estimation. *Proc. Natl. Acad. Sci. U.S.A.*, 108(50):19867–19872.

Li, W., Feng, J., and Jiang, T. (2011b). IsoLasso: a LASSO regression approach to RNA-Seq based transcriptome assembly. *J. Comput. Biol.*, 18(11):1693–1707.

Li, W. and Jiang, T. (2012a). Transcriptome assembly and isoform expression level estimation from biased RNA-Seq reads. *Bioinformatics*, 28(22):2914–2921.

Li, W. and Jiang, T. (2012b). Transcriptome assembly and isoform expression level estimation from biased RNA-Seq reads. *Bioinformatics*, 28(22):2914–2921.

Lin, Y.-Y., Dao, P., Hach, F., Bakhshi, M., Mo, F., Lapuk, A., Collins, C., and Sahinalp, S. C. (2012). Cliiq: Accurate comparative detection and quantification of expressed isoforms in a population. In Raphael, B. J. and Tang, J., editors, *WABI*, volume 7534 of *Lecture Notes in Computer Science*, pages 178–189.

Lopez-Bigas, N., Audit, B., Ouzounis, C., Parra, G., and Guigo, R. (2005). Are splicing mutations the most frequent cause of hereditary disease? *FEBS Lett.*, 579(9):1900–1903.

Lounici, K., Pontil, M., Tsybakov, A. B., and van de Geer, S. (2009). Taking advantage of sparsity in multi-task learning. In *Proceedings of the 22nd Conference on Information Theory*, pages 73–82.

Luco, R. F., Allo, M., Schor, I. E., Kornblihtt, A. R., and Misteli, T. (2011). Epigenetics in alternative pre-mRNA splicing. *Cell*, 144(1):16–26.

Lykke-Andersen, S. and Jensen, T. H. (2015). Nonsense-mediated mRNA decay: an intricate machinery that shapes transcriptomes. *Nat. Rev. Mol. Cell Biol.*, 16(11):665–677.

Mairal, J. (2010). *Sparse coding for machine learning, image processing and computer vision*. Theses, École normale supérieure de Cachan - ENS Cachan.

Mairal, J., Bach, F., and Ponce, J. (2014). Sparse modeling for image and vision processing. *Found. Trends. Comput. Graph. Vis.*, 8(2-3):85–283.

Mairal, J., Jenatton, R., Obozinski, G., and Bach, F. (2011). Convex and network flow optimization for structured sparsity. *J. Mach. Learn. Res.*, 12:2681–2720.

Mairal, J. and Yu, B. (2013). Supervised feature selection in graphs with path coding penalties and network flows. *J. Mach. Learn. Res.*, 14:2449–2485.

Malcovati, L., Papaemmanuil, E., Ambaglio, I., Elena, C., Galli, A., Della Porta, M. G., Travaglino, E., Pietra, D., Pascutto, C., Ubezio, M., Bono, E., Da Via, M. C., Brisci, A., Bruno, F., Cremonesi, L., Ferrari, M., Boveri, E., Invernizzi, R., Campbell, P. J., and Cazzola, M. (2014). Driver somatic mutations identify distinct disease entities within myeloid neoplasms with myelodysplasia. *Blood*, 124(9):1513–1521.

Mamanova, L., Coffey, A. J., Scott, C. E., Kozarewa, I., Turner, E. H., Kumar, A., Howard, E., Shendure, J., and Turner, D. J. (2010). Target-enrichment strategies for next-generation sequencing. *Nat. Methods*, 7(2):111–118.

Mangul, S., Caciula, A., Seesi, S. A., Brinza, D., Banday, A. R., and Kanadia, R. (2012). An integer programming approach to novel transcript reconstruction from paired-end RNA-seq reads. In *ACM International Conference on Bioinformatics, Computational Biology and Biomedicine*, pages 369–376.

Mardis, E. R. (2011). A decade's perspective on DNA sequencing technology. *Nature*, 470(7333):198–203.

Maretty, L., Sibbesen, J. A., and Krogh, A. (2014). Bayesian transcriptome assembly. *Genome Biol.*, 15(10):501.

Martin, J. and Wang, Z. (2011). Next-generation transcriptome assembly. *Nat. Rev. Genet.*, 12(10):671–682.

Marygold, S. J., Leyland, P. C., Seal, R. L., et al. (2013). Flybase: improvements to the bibliography. *Nucleic Acids Res.*, 41(D1):D751–D757.

Matlin, A. J., Clark, F., and Smith, C. W. (2005). Understanding alternative splicing: towards a cellular code. *Nat. Rev. Mol. Cell Biol.*, 6(5):386–398.

McKernan, K. J., Peckham, H. E., Costa, G. L., McLaughlin, S. F., Fu, Y., Tsung, E. F., Clouser, C. R., Duncan, C., Ichikawa, J. K., Lee, C. C., Zhang, Z., Ranade, S. S., Dimalanta, E. T., Hyland, F. C., Sokolsky, T. D., Zhang, L., Sheridan, A., Fu, H., Hendrickson, C. L., Li, B., Kotler, L., Stuart, J. R., Malek, J. A., Manning, J. M., Antipova, A. A., Perez, D. S., Moore, M. P., Hayashibara, K. C., Lyons, M. R., Beaudoin, R. E., Coleman, B. E., Laptewicz, M. W., Sannicandro, A. E., Rhodes, M. D., Gottimukkala, R. K., Yang, S., Bafna, V., Bashir, A., MacBride, A., Alkan, C., Kidd, J. M., Eichler, E. E., Reese, M. G., De La Vega, F. M., and Blanchard, A. P. (2009). Sequence and structural variation in a human genome uncovered by short-read, massively parallel ligation sequencing using two-base encoding. *Genome Res.*, 19(9):1527–1541.

McPherson, A., Wu, C., Wyatt, A. W., Shah, S., Collins, C., and Sahinalp, S. C. (2012). nFuse: discovery of complex genomic rearrangements in cancer using high-throughput sequencing. *Genome Res.*, 22(11):2250–2261.

Medvedev, P. and Brudno, M. (2009). Maximum likelihood genome assembly. *J. Compute Biol.*, 16(8):1101–1116.

Meinshausen, N. and Bühlmann, P. (2010). Stability selection. *J Roy Stat Soc B*, 72(4):417–473.

Melamud, E. and Moult, J. (2009). Stochastic noise in splicing machinery. *Nucleic Acids Res.*, 37(14):4873–4886.

Mercer, T. R., Gerhardt, D. J., Dinger, M. E., Crawford, J., Trapnell, C., Jeddeloh, J. A., Mattick, J. S., and Rinn, J. L. (2012). Targeted RNA sequencing reveals the deep complexity of the human transcriptome. *Nat. Biotechnol.*, 30(1):99–104.

Merkhofer, E. C., Hu, P., and Johnson, T. L. (2014). Introduction to cotranscriptional RNA splicing. *Methods Mol. Biol.*, 1126:83–96.

Mezlini, A. M., M., S. E. J., Fiume, M., Buske, O., Savich, G., Shah, S., Aparicion, S., Chiang, D., Goldenberg, A., and Brudno, M. (2013). iReckon: Simultaneous isoform discovery and abundance estimation from RNA-seq data. *Genome Res.*, 23(3):519–529.

Modrek, B., Resch, A., Grasso, C., and Lee, C. (2001). Genome-wide detection of alternative splicing in expressed sequences of human genes. *Nucleic Acids Res.*, 29(13):2850–2859.

Montgomery, S. B. et al. (2010). Transcriptome genetics using second generation sequencing in a Caucasian population. *Nature*, 464(7289):773–777.

Mortazavi, A. et al. (2010). Scaffolding a caenorhabditis nematode genome with RNA-Seq. *Genome Res.*, 20(12):1740–1747.

Mortazavi, A., Williams, B. A., McCue, K., and Schaeffer, L. (2008). Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nat. Methods*, 5(7):621–628.

Mudge, J. M., Frankish, A., Fernandez-Banet, J., Alioto, T., Derrien, T., Howald, C., Reymond, A., Guigo, R., Hubbard, T., and Harrow, J. (2011). The origins, evolution, and functional potential of alternative splicing in vertebrates. *Mol. Biol. Evol.*, 28(10):2949–2959.

Nagalakshmi, U., Wang, Z., Waern, K., Shou, C., Raha, D., Gerstein, M., and Snyder, M. (2008). The transcriptional landscape of the yeast genome defined by RNA sequencing. *Science*, 320(5881):1344–1349.

Nagaraj, S. H., Gasser, R. B., and Ranganathan, S. (2007). A hitchhiker's guide to expressed sequence tag (est) analysis. *Brief. Bioinform.*, 8(1):6–21.

Natarajan, B. K. (1995). Sparse approximate solutions to linear systems. *SIAM J. Comput.*, 24(2):227–234.

Navin, N. E. (2015). The first five years of single-cell cancer genomics and beyond. *Genome Res.*, 25(10):1499–1507.

Nicolae, M., Mangul, S., M?ndoiu, I. I., and Zelikovsky, A. (2011). Estimation of alternative splicing isoform frequencies from RNA-Seq data. *Algorithms Mol Biol*, 6(1):9.

Nilsen, T. W. and Graveley, B. R. (2010). Expansion of the eukaryotic proteome by alternative splicing. *Nature*, 463(7280):457–463.

Ozsolak, F. and Milos, P. M. (2011). RNA sequencing: advances, challenges and opportunities. *Nat. Rev. Genet.*, 12(2):87–98.

Pachter, L. (2011). Models for transcript quantification from RNA-Seq. *ArXiv e-prints*.

Pal, S., Gupta, R., and Davuluri, R. V. (2012). Alternative transcription and alternative splicing in cancer. *Pharmacol Ther*, 136(3):283–294.

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

Pan, Q., Shai, O., Misquitta, C., Zhang, W., Saltzman, A. L., Mohammad, N., Babak, T., Siu, H., Hughes, T. R., Morris, Q. D., Frey, B. J., and Blencowe, B. J. (2004). Revealing global regulatory features of mammalian alternative splicing using a quantitative microarray platform. *Mol. Cell*, 16(6):929–941.

Papaemmanuil, E., Cazzola, M., Boultwood, J., Malcovati, L., Vyas, P., Bowen, D., Pellagatti, A., Wainscoat, J. S., Hellstrom-Lindberg, E., Gambacorti-Passerini, C., Godfrey, A. L., Rapado, I., Cvejic, A., Rance, R., McGee, C., Ellis, P., Mudie, L. J., Stephens, P. J., McLaren, S., Massie, C. E., Tarpey, P. S., Varela, I., Nik-Zainal, S., Davies, H. R., Shlien, A., Jones, D., Raine, K., Hinton, J., Butler, A. P., Teague, J. W., Baxter, E. J., Score, J., Galli, A., Della Porta, M. G., Travaglino, E., Groves, M., Tauro, S., Munshi, N. C., Anderson, K. C., El-Naggar, A., Fischer, A., Mustonen, V., Warren, A. J., Cross, N. C., Green, A. R., Futreal, P. A., Stratton, M. R., and Campbell, P. J. (2011). Somatic SF3B1 mutation in myelodysplasia with ring sideroblasts. *N. Engl. J. Med.*, 365(15):1384–1395.

Patro, R., Mount, S. M., and Kingsford, C. (2014). Sailfish enables alignment-free isoform quantification from RNA-seq reads using lightweight algorithms. *Nat. Biotechnol.*, 32(5):462–464.

Pertea, M., Pertea, G. M., Antonescu, C. M., Chang, T. C., Mendell, J. T., and Salzberg, S. L. (2015). StringTie enables improved reconstruction of a transcriptome from RNA-seq reads. *Nat. Biotechnol.*, 33(3):290–295.

Popp, M. W. and Maquat, L. E. (2013). Organizing principles of mammalian nonsense-mediated mRNA decay. *Annu. Rev. Genet.*, 47:139–165.

Pruitt, K., Tatusova, T., and Maglott, D. R. (2005). Ncbi reference sequence (refseq): a curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic Acids Res.*, 33(supp1):D501–D504.

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

Roberts, A. and Pachter, L. (2013). Streaming fragment assignment for real-time analysis of sequencing experiments. *Nat. Methods*, 10(1):71–73.

Roberts, A., Trapnell, C., Donaghey, J., Rinn, J. L., and Pachter, L. (2011). Improving RNA-Seq expression estimates by correcting for fragment bias. *Genome Biol.*, 12(3):R22.

Robertson, G., Schein, J., Chiu, R., Corbett, R., Field, M., Jackman, S. D., Mungall, K., Lee, S., Okada, H. M., Qian, J. Q., Griffith, M., Raymond, A., Thiessen, N., Cezard, T., Butterfield, Y. S., Newsome, R., Chan, S. K., She, R., Varhol, R., Kamoh, B., Prabhu, A. L., Tam, A., Zhao, Y., Moore, R. A., Hirst, M., Marra, M. A., Jones, S. J., Hoodless, P. A., and Birol, I. (2010). De novo assembly and analysis of RNA-seq data. *Nat. Methods*, 7(11):909–912.

Robinson, J. T., Thorvaldsdottir, H., Winckler, W., Guttman, M., Lander, E. S., Getz, G., and Mesirov, J. P. (2011). Integrative genomics viewer. *Nat. Biotechnol.*, 29(1):24–26.

Romero, A., Garcia-Garcia, F., Lopez-Perolio, I., Ruiz de Garibay, G., Garcia-Saenz, J. A., Garre, P., Ayllon, P., Benito, E., Dopazo, J., Diaz-Rubio, E., Caldes, T., and de la Hoya, M. (2015). BRCA1 Alternative splicing landscape in breast tissue samples. *BMC Cancer*, 15:219.

Rossell, D., Stephan-Otto Attolini, C., Kroiss, M., and Stocker, A. (2014). Quantifying alternative splicing from paired-end rna-seq data. *Ann. Appl. Stat.*, 8(1):309–330.

Rosset, S. and Zhu, J. (2007). Piecewise linear regularized solution paths. *Ann. Stat.*, 35(3):1012–1030.

Rothberg, J. M., Hinz, W., Rearick, T. M., Schultz, J., Mileski, W., Davey, M., Leamon, J. H., Johnson, K., Milgrew, M. J., Edwards, M., Hoon, J., Simons, J. F., Marran, D., Myers, J. W., Davidson, J. F., Branting, A., Nobile, J. R., Puc, B. P., Light, D., Clark, T. A., Huber, M., Branciforte, J. T., Stoner, I. B., Cawley, S. E., Lyons, M., Fu, Y., Homer, N., Sedova, M., Miao, X., Reed, B., Sabina, J., Feierstein, E., Schorn, M., Alanjary, M., Dimalanta, E., Dressman, D., Kasinskas, R., Sokolsky, T., Fidanza, J. A., Namsaraev, E., McKernan, K. J., Williams, A., Roth, G. T., and Bustillo, J. (2011). An integrated semiconductor device enabling non-optical genome sequencing. *Nature*, 475(7356):348–352.

Roy, R., Chun, J., and Powell, S. N. (2012). BRCA1 and BRCA2: different roles in a common pathway of genome protection. *Nat. Rev. Cancer*, 12(1):68–78.

Salz, H. K. (2011). Sex determination in insects: a binary decision based on alternative splicing. *Curr. Opin. Genet. Dev.*, 21(4):395–400.

Salzman, J., Jiang, H., and Wong, W. H. (2011). Statistical modeling of RNA-Seq data. *Stat. Sci.*, 26(1):62–83.

Sanger, F., Nicklen, S., and Coulson, A. R. (1977). DNA sequencing with chain-terminating inhibitors. *Proc. Natl. Acad. Sci. U.S.A.*, 74(12):5463–5467.

Schmucker, D., Clemens, J. C., Shu, H., Worby, C. A., Xiao, J., Muda, M., Dixon, J. E., and Zipursky, S. L. (2000). Drosophila Dscam is an axon guidance receptor exhibiting extraordinary molecular diversity. *Cell*, 101(6):671–684.

Schulz, M. H., Zerbino, D. R., Vingron, M., and Birney, E. (2012). Oases: robust de novo RNA-seq assembly across the dynamic range of expression levels. *Bioinformatics*, 28(8):1086–1092.

Schwartz, S. and Ast, G. (2010). Chromatin density and splicing destiny: on the cross-talk between chromatin structure and splicing. *EMBO J.*, 29(10):1629–1636.

Schwarz, G. (1978). Estimating the Dimension of a Model. *Ann. Stat.*, 6(2):461–464.

Schwerk, C. and Schulze-Osthoff, K. (2005). Regulation of apoptosis by alternative pre-mRNA splicing. *Mol. Cell*, 19(1):1–13.

Scotti, M. M. and Swanson, M. S. (2016a). RNA mis-splicing in disease. *Nat. Rev. Genet.*, 17(1):19–32.

Scotti, M. M. and Swanson, M. S. (2016b). RNA mis-splicing in disease. *Nat. Rev. Genet.*, 17(1):19–32.

Sebestyen, E., Zawisza, M., and Eyras, E. (2015). Detection of recurrent alternative splicing switches in tumor samples reveals novel signatures of cancer. *Nucleic Acids Res.*, 43(3):1345–1356.

Shapiro, I. M., Cheng, A. W., Flytzanis, N. C., Balsamo, M., Condeelis, J. S., Oktay, M. H., Burge, C. B., and Gertler, F. B. (2011). An EMT-driven alternative splicing program occurs in human breast cancer and modulates cellular phenotype. *PLoS Genet.*, 7(8):e1002218.

Singh, D. et al. (2011). FDM: a graph-based statistical method to detect differential transcription using RNA-seq data. *Bioinformatics*, 27(19):2633–2640.

Skandalis, A., Frampton, M., Seger, J., and Richards, M. H. (2010). The adaptive significance of unproductive alternative splicing in primates. *RNA*, 16(10):2014–2022.

Smith, L. M., Sanders, J. Z., Kaiser, R. J., Hughes, P., Dodd, C., Connell, C. R., Heiner, C., Kent, S. B., and Hood, L. E. (1986). Fluorescence detection in automated DNA sequence analysis. *Nature*, 321(6071):674–679.

Song, L. and Florea, L. (2013). CLASS: constrained transcript assembly of RNA-seq reads. *BMC Bioinformatics*, 14 Suppl 5:S14.

Sorek, R., Lev-Maor, G., Reznik, M., Dagan, T., Belinky, F., Graur, D., and Ast, G. (2004). Minimal conditions for exonization of intronic sequences: 5' splice site formation in alu exons. *Mol. Cell*, 14(2):221–231.

Spitali, P. and Aartsma-Rus, A. (2012). Splice modulating therapies for human disease. *Cell*, 148(6):1085–1088.

Spurdle, A. B., Couch, F. J., Hogervorst, F. B., Radice, P., Sinilnikova, O. M., Boffetta, P., Couch, F., de Wind, N., Easton, D., Eccles, D., Foulkes, W., Genuardi, M., Goldgar, D., Greenblatt, M., Hofstra, R., Hogervorst, F., Hoogerbrugge, N., Plon, S., Radice, P., Rasmussen, L., Sinilnikova, O., Spurdle, A., and Tavtigian, S. V. (2008). Prediction and assessment of splicing alterations: implications for clinical testing. *Hum. Mutat.*, 29(11):1304–1313.

Srebrow, A. and Kornblihtt, A. R. (2006). The connection between splicing and cancer. *J. Cell. Sci.*, 119(Pt 13):2635–2641.

Steijger, T., Abril, J. F., Engstrom, P. G., Kokocinski, F., Hubbard, T. J., Guigo, R., Harrow, J., Bertone, P., Abril, J. F., Akerman, M., Alioto, T., Ambrosini, G., Antonarakis, S. E., Behr, J., Bertone, P., Bohnert, R., Bucher, P., Cloonan, N., Derrien, T., Djebali, S., Du, J., Dudoit, S., Engstrom, P., Gerstein, M., Gingeras, T. R., Gonzalez, D., Grimmond, S. M., Guigo, R., Habegger, L., Harrow, J., Hubbard, T. J., Iseli, C., Jean, G., Kahles, A., Kokocinski, F., Lagarde, J., Leng, J., Lefebvre, G., Lewis, S., Mortazavi, A., Niermann, P., Ratsch, G., Reymond, A., Ribeca, P., Richard, H., Rougemont, J., Rozowsky, J., Sammeth, M., Sboner, A., Schulz, M. H., Searle, S. M., Solorzano, N. D., Solovyev, V., Stanke, M., Steijger, T., Stevenson, B. J., Stockinger, H., Valsesia, A., Weese, D., White, S., Wold, B. J., Wu, J., Wu, T. D., Zeller, G., Zerbino, D., and Zhang, M. Q. (2013). Assessment of transcript reconstruction methods for RNA-seq. *Nat. Methods*, 10(12):1177–1184.

Tibshirani, R. (1996). Regression shrinkage and selection via the Lasso. *J. R. Stat. Soc. Series B Stat. Methodol.*, 58(1):267–288.

Tomescu, A., Kuosmanen, A., Rizzi, R., and Makinen, V. (2013). A novel min-cost flow method for estimating transcript expression with rna-seq. *BMC Bioinformatics*, 14 (Suppl 5):S15.

Trapnell, C., Hendrickson, D. G., Sauvageau, M., Goff, L., Rinn, J. L., and Patcher, L. (2013). Differential analysis of gene regulation at transcript resolution with rna-seq. *Nat. Biotechnol.*, 31(1):46–53.

Trapnell, C., Patcher, L., and Salzberg, S. L. (2009). TopHat: discovering splice junctions with RNA-Seq. *Bioinformatics*, 25(9):1105–1111.

Trapnell, C., Williams, B. A., Pertea, G., Mortazavi, A. M., Kwan, G., van Baren, M. J., L., S. S., Wold, B., and Pachter, L. (2010). Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nat. Biotechnol.*, 28(5):511–515.

Tsai, Y. S., Dominguez, D., Gomez, S. M., and Wang, Z. (2015). Transcriptome-wide identification and study of cancer-specific splicing events across multiple tumors. *Oncotarget*, 6(9):6825–6839.

Turro, E., Su, S. Y., Goncalves, A., Coin, L. J., Richardson, S., and Lewin, A. (2011). Haplotype and isoform specific expression estimation using multi-mapping RNA-seq reads. *Genome Biol.*, 12(2):R13.

van Dijk, E. L., Auger, H., Jaszczyszyn, Y., and Thermes, C. (2014). Ten years of next-generation sequencing technology. *Trends Genet.*, 30(9):418–426.

Van Keuren-Jensen, K., Keats, J. J., and Craig, D. W. (2014). Bringing RNA-seq closer to the clinic. *Nat. Biotechnol.*, 32(9):884–885.

Wang, E. T., Sandberg, R., Luo, S., Khrebtukova, I., Zhang, L., and Mayr, C. (2008a). Alternative isoform regulation in human tissue transcriptomes. *Nature*, 456(7721):470–476.

Wang, E. T., Sandberg, R., Luo, S., Khrebtukova, I., Zhang, L., Mayr, C., Kingsmore, S. F., Schroth, G. P., and Burge, C. B. (2008b). Alternative isoform regulation in human tissue transcriptomes. *Nature*, 456(7221):470–476.

Wang, G. S. and Cooper, T. A. (2007). Splicing in disease: disruption of the splicing code and the decoding machinery. *Nat. Rev. Genet.*, 8(10):749–761.

Wang, K., Singh, D., Zeng, Z., Coleman, S. J., Huang, Y., Savich, G. L., He, X., Mieczkowski, P., Grimm, S. A., Perou, C. M., MacLeod, J. N., Chiang, D. Y., Prins, J. F., and Liu, J. (2010). MapSplice: accurate mapping of RNA-seq reads for splice junction discovery. *Nucleic Acids Res.*, 38(18):e178.

Wang, Z., Gerstein, M., and Snyder, M. (2009). RNA-Seq: a revolutionary tool for transcriptomics. *Nat. Rev. Genet.*, 10(1):57–63.

Wu, J. Q., Habegger, L., Noisa, P., Szekely, A., Qiu, C., Hutchison, S., Raha, D., Egholm, M., Lin, H., Weissman, S., et al. (2010). Dynamic transcriptomes during neural differentiation of human embryonic stem cells revealed by short, long, and paired-end sequencing. *Proceedings of the National Academy of Sciences*, 107(11):5254–5259.

Wu, T. D. and Nacu, S. (2010). Fast and SNP-tolerant detection of complex variants and splicing in short reads. *Bioinformatics*, 26(7):873–881.

Xia, Z., J., W., Chang, C.-C., and Zhou, X. (2011). NSMAP: a method for spliced isoforms identification and quantification from RNA-Seq. *BMC Bioinformatics*, 12:162.

Xie, Y., Wu, G., Tang, J., Luo, R., Patterson, J., Liu, S., Huang, W., He, G., Gu, S., Li, S., Zhou, X., Lam, T. W., Li, Y., Xu, X., Wong, G. K., and Wang, J. (2014). SOAPdenovo-Trans: de novo transcriptome assembly with short RNA-Seq reads. *Bioinformatics*, 30(12):1660–1666.

Xu, Q., Modrek, K., and Lee, C. (2002). Genome-wide detection of tissue-specific alternative splicing in the human transcriptome. *Nucleic Acids Res.*, 30(17):3754–3766.

Yan, J. and Marr, T. G. (2005). Computational analysis of 3'-ends of ESTs shows four classes of alternative polyadenylation in human, mouse, and rat. *Genome Res.*, 15(3):369–375.

Yeo, G., Holste, D., Kreiman, G., and B., B. C. (2004). Variation in alternative splicing across human tissues. *Genome Biology*, 5(10):R74.

Yoshida, K. and Ogawa, S. (2014). Splicing factor mutations and cancer. *Wiley Interdiscip Rev RNA*, 5(4):445–459.

Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *J. R. Stat. Soc. Ser.*, 68(1):49–67.

Zerbino, D. R., Ballinger, T., Paten, B., Hickey, G., and Haussler, D. (2013). Representing and decomposing genomic structural variants as balanced integer flows on sequence graphs. *ArXiv e-prints*.

Zhang, J. D., Schindler, T., Kung, E., Ebeling, M., and Certa, U. (2014). Highly sensitive amplicon-based transcript quantification by semiconductor sequencing. *BMC Genomics*, 15:565.

Zhang, K., Li, J. B., Gao, Y., Egli, D., Xie, B., Deng, J., Li, Z., Lee, J. H., Aach, J., Leproust, E. M., Eggan, K., and Church, G. M. (2009). Digital RNA allelotyping reveals tissue-specific and allele-specific gene expression in human. *Nat. Methods*, 6(8):613–618.

Zhang, Z., Pal, S., Bi, Y., Tchou, J., and Davuluri, R. V. (2013). Isoform level expression profiles provide better cancer signatures than gene level expression profiles. *Genome Med*, 5(4):33.

Zheng, W., Chung, L. M., and Zhao, H. (2011). Bias detection and correction in RNA-Sequencing data. *BMC Bioinformatics*, 12:290.

Zheng, Z., Liebers, M., Zhelyazkova, B., Cao, Y., Panditi, D., Lynch, K. D., Chen, J., Robinson, H. E., Shim, H. S., Chmielecki, J., Pao, W., Engelman, J. A., Iafrate, A. J., and Le, L. P. (2014). Anchored multiplex PCR for targeted next-generation sequencing. *Nat. Med.*, 20(12):1479–1484.

Résumé

Le nombre de gènes codant pour des protéines chez l'homme, le vers rond et la mouche des fruits est du même ordre de grandeur. Cette absence de correspondance entre le nombre de gènes d'un eucaryote et sa complexité phénotypique s'explique en partie par le caractère alternatif de l'épissage. L'épissage alternatif augmente considérablement le répertoire fonctionnel de protéines codées par un nombre limité de gènes. Ce mécanisme, très actif lors du développement embryonnaire, participe au devenir cellulaire. De nombreux troubles génétiques, hérités ou acquis (en particulier certains cancers), se caractérisent par une altération de son fonctionnement.

Les technologies de séquençage à haut débit de l'ARN donnent accès à une information plus riche sur le mécanisme de l'épissage. Cependant, si la lecture à haut débit des séquences d'ARN est plus rapide et moins coûteuse, les données qui en sont issues sont complexes et nécessitent le développement d'outils algorithmiques pour leur interprétation. En particulier, la reconstruction des transcrits alternatifs requiert une étape de déconvolution non triviale.

Dans ce contexte, cette thèse participe à l'étude des événements d'épissage et des transcrits alternatifs à partir de données de séquençage à haut débit de l'ARN.

Nous proposons de nouvelles méthodes pour reconstruire et quantifier les transcrits alternatifs de façon plus efficace et précise. Nos contributions méthodologiques impliquent des techniques de régression parcimonieuse, basées sur l'optimisation convexe et sur des algorithmes de flots. Nous étudions également une procédure pour détecter des anomalies d'épissage dans un contexte de diagnostic clinique. Nous suggérons un protocole expérimental facilement opérant et développons de nouveaux modèles statistiques et algorithmes pour quantifier des événements d'épissage et mesurer leur degré d'anormalité chez le patient.

Mots-Clés

Epissage alternative, isoforms, RNA-seq, regression parcimonieuse, optimisation convexe, diagnostic clinique, variant de signification inconnu.

Abstract

The number of protein-coding genes in a human, a nematode and a fruit fly are roughly equal. The paradoxical miscorrelation between the number of genes in an organism's genome and its phenotypic complexity finds an explanation in the alternative nature of splicing in higher organisms.

Alternative splicing largely increases the functional diversity of proteins encoded by a limited number of genes. It is known to be involved in cell fate decision and embryonic development, but also appears to be dysregulated in inherited and acquired human genetic disorders, in particular in cancers.

High-throughput RNA sequencing technologies allow us to measure and question splicing at an unprecedented resolution. However, while the cost of sequencing RNA decreases and throughput increases, many computational challenges arise from the discrete and local nature of the data. In particular, the task of inferring alternative transcripts requires a non-trivial deconvolution procedure.

In this thesis, we contribute to deciphering alternative transcript expressions and alternative splicing events from high-throughput RNA sequencing data.

We propose new methods to accurately and efficiently detect and quantify alternative transcripts. Our methodological contributions largely rely on sparse regression techniques and takes advantage of network flow optimization techniques. Besides, we investigate means to query splicing abnormalities for clinical diagnosis purposes. We suggest an experimental protocol that can be easily implemented in routine clinical practice, and present new statistical models and algorithms to quantify splicing events and measure how abnormal these events might be in patient data compared to wild-type situations.

Keywords

Alternative splicing, isoforms, RNA-seq, sparse regression, convex optimization, clinical diagnosis, variant of unknown significance.