Charles University
Faculty of Science

Study programme: Bioinformatics

Branch of study: Bioinformatics

František Kloda

Gene expression analysis on a subgene level

Analýza genové exprese na subgenové úrovni

Bachelor's thesis

Supervisor: RNDr. Karel Fišer, Ph.D.

Prague 2023

I declare that I carried out this bachelor thesis independently, and only with the cited sources, literature and other professional sources. It has not been used to obtain another or the same degree.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In . . . . . date. . . . .

Author's signature

I would like to thank my supervisor Karel Fišer for guidance, feedback and support during writing of this thesis. I would also like to thank my family and friends for having patience with me and helping me during my studies.

Title: Gene expression analysis on a subgene level

Author: František Kloda

Department: Department of Cell Biology

Supervisor: RNDr. Karel Fišer, Ph.D., Department of Bioinformatics, Second Faculty of Medicine, Charles University

Abstract: RNA sequencing allows investigation of expression of singular genes in cells. It is possible to interpret the arisen data on multiple levels, each level providing a different type of information. Apart from measuring expression of whole genes, it is possible to quantify expression of singular exons, or transcripts (gene isoforms), which allows more detailed study of regulatory mechanisms. The main difference between approaches is in determining the origin of short reads. This step is significantly more complex in analysis of expression of transcripts, as transcripts derived from the same gene have typically larger rate of sequential similarity. In this thesis, we describe eleven tools for subgene level expression analysis a as comparison we have tested three of these tools on real patient data. The results provided by all three tools proved to be very similar with the greatest difference being the time needed for the analysis.

Abstrakt: RNA sekvenování nám umožňuje zkoumat expresi jednotlivých genů v buňkách. Vzniklá data je možné interpretovat na více úrovních, kde každá úroveň poskytuje rozdílný typ informace. Kromě měření exprese celých genů je možné kvantifikovat expresi jednolivých exonů, nebo transkriptů (isoforem genů), což umožňuje podrobnější stadium regulačních mechanizmů. Hlavní rozdíl mezi přístupy je při určování původu krátkých readů. Tento krok je především složitější při analýze exprese jednotlivých transkriptů kvůli velké míře sekvenční podobnosti mezi transkripty pocházejícími ze stejného genu. V této práci jsme popsali jedenáct nástrojů pro analýzu exprese na subgenové úrovni a pro porovnání jsme tři z těchto nástrojů spustili na reálných pacientských datech. Výsledky poskytnuté všemi třemi nástroji byli velmi podobné, nejvýraznější rozdíl byl v čase analýzy.

Keywords: bioinformatics, RNA-seq, expression analysis, transcriptome, sequencing

Klíčová slova: bioinformatika, RNA-seq, analýza exprese, transkriptom, sekvenování

Contents

List of Abbreviations **3**

Introduction **4**

1 Biological background **5**
1.1 Gene expression 5
1.2 Subgene level 5
1.3 Sequencing 7
1.3.1 Approaches 7
1.3.2 Library preparation 7
1.3.3 Methods of sequencing by synthesis 8
1.3.4 Next generation sequencing 8
1.3.5 Third generation sequencing 9
1.3.6 Applications of NGS 9
1.3.7 RNA-seq 10

2 Expression analysis **12**
2.1 Data types used in expression analysis 12
2.1.1 FASTA 12
2.1.2 FASTQ 12
2.1.3 SAM/BAM 13
2.1.4 GFF/GTF 13
2.2 Gene expression analysis 13
2.2.1 Quality assessment and trimming 13
2.2.2 Mapping 14
2.2.3 Quantification of reads 14
2.3 Subgene level expression analysis 15

3 Tools **16**
3.1 Alignment-based tools 16
3.1.1 DEXSeq 16
3.1.2 JunctionSeq 17
3.1.3 DiffSplice 18
3.1.4 BitSeq 19
3.1.5 rSeqDiff 19
3.1.6 eXpress 20
3.1.7 Cufflinks 20
3.1.8 QuasR 21
3.1.9 featureCount 21
3.2 Alignment-free tools 22
3.2.1 Kallisto 22
3.2.2 Salmon 23

4 Results **25**
4.1 Materials and methods 25
4.1.1 Reference 25
4.1.2 Methods 26
4.2 Results 26

Conclusion **29**

Bibliography **30**

List of Figures **33**

Attachments **34**

List of Abbreviations

ALL Acute lymphoblastic leukemia

EM Expectation-Maximization

HTS Hight throughput sequencing

MPS Massively parallel sequencing

NGS Next generation sequencing

RPK Reads per kilobase

RPKM Reads per kilobase million

TPM Transcripts per kilobase million

WES Whole-exom sequencing

WGS Whole-genome sequencing

Introduction

The quantification of gene expression, by counting the number of RNA molecules transcribed from specific gene, is an important step of many biological studies. It allows understanding of processes occurring in given sample cells as well as providing important information for studies of regulatory mechanisms. Precise assessment of gene expression helps with diagnosis of diseases and treatment decision. While quantifying gene expression is the mostly used and most straightforward approach, it does not necessary provide the most detailed information available. It is possible to measure expression and compare usage of exons or known gene isoforms. The advantages and difficulties of measuring expression at different levels will be discussed further in this work.

There are multiple methods designed for measuring expression, but in the scope of this thesis, only methods related to sequencing will be considered. The greatest difference in measuring expression at different levels using a sequencing technology is in data analysis, as the raw data representing amount of RNA molecules present in given sample, are the same for each approach. Because of that, a great number of tools for such analysis has been developed, each offering a different approach. These tools will be presented, described, and compared in further chapters of this thesis. I have also tested three of these tools on patient data, which allows direct comparison of results yielded by each tool.

1 Biological background

1.1 Gene expression

Genetic information, stored in the form of nucleic acid is used for synthesis of most of biologically active molecules present in cells of any living organism. Despite all cells in a body of multicellular organism having the same genetic information, the cells themselves and processes that occur inside them are seldom the same. That is because of the fact, that not all parts of genome are always active. By being active, it is meant, that DNA is being transcribed into RNA, which can act as an active molecule, or serve as template for synthesis of a protein. This way, each cell creates only products of genes, that are relevant to given cell type. This process is called transcription regulation and concerns many different mechanisms and complex pathways.

The molecules of RNA, derived from specific locus in genome, have nucleotide sequence complement to that of the part of DNA that served as template for synthesis of given RNA molecule. Thanks to that, it is possible to determine, from which part of genome every RNA molecule found in a cell was derived. Measuring the amount of identical RNA molecules present in a cell at given time provides quantitative information about the scale, at which a gene, or other genomic feature, is being used, thus measuring expression of given feature. Measuring expression in a particular set of cells can provide us with important information about given sample. Not only can this information be used for identification of cells, given prior knowledge about genes, or sets of genes that are specifically transcribed for a particular cell type, but information from measuring gene expression can help understanding a physiological processes undertaking in cells of sample of interest[1]. This can lead for example to accurate determination of disease and applying adequate treatment. Expression analysis is an important approach in study of regulation of expression, as inhibition of transcription is the most energy efficient way of reducing the rate of synthesis of a specific protein, or other active molecule.

In majority of experiments, it is the expression of genes that is being measured. Genes have the advantage of being relatively well defined and well annotated in genome and are often provided with description of their function. Also measuring gene expression has the advantage, that genes have mostly distinct sequences, allowing determining the origin of RNA molecule with low level of uncertainty. Observing expression of a whole gene can provide sufficient information e.g. for diagnosis or monitoring a gene knockdown[2].

1.2 Subgene level

The transcribed sequence of an eukaryotic gene is divided into exons and introns, where only exons contain genetic information, that is to be used for synthesis of an active molecule (RNA or protein). The gene is transcribed as a whole, but the arisen mRNA undergoes post-transcriptional modifications. Splicing is one of these modifications and it results in removal of parts of the mRNA, that correspond to intronic regions, so that only the parts of RNA originated from exons remain in the

final transcript. Some exons can be also removed from the transcript during splicing resulting in multiple ways, how the original whole-gene transcript can be spliced. This mechanism is called alternative splicing. As a result, one gene can serve as a template to many distinct mRNA molecules, which than can serve as template for synthesis of different proteins. The set of mRNA molecules sharing the same gene as a template is then called splice variants, or isoforms of the given gene. In cells the whole process of alternative splicing is regulated by complex regulatory pathways.

Because of the nature of an eukaryotic gene, it is possible to measure expression of not only genes, as sum of expression of all exons present in given gene, but also other genomic features, such as isoforms of genes, or exons. By measuring expression of gene isoforms, it is possible to measure the abundance of particular transcripts in given sample, which provides a more detailed information about processes occurring in given cells. However, measuring transcript abundance comes at the cost of more complex quantification step. The reason is the fact, that transcripts originated from the same gene often share common parts of their sequences, making the determining of origin of RNA molecules more complex than in case of measuring whole gene expression. Finally, it is possible to measure the abundance of exons. This approach can be used to increase precision of differential gene, or isoform expression analysis[3] when the difference between samples is not great. Exon-level approach also allows pinpointing the exact differences between multiple transcripts of a single gene. This can be helpful in study of regulatory signals and functional domains, that may play a role in deciding, which isoform is to be transcribed[4], or may have a significant impact in development of a disease.

Each approach carries its own limitations and benefits, which are often related to the nature of data, that is mostly being used for expression analyses. Figure 1.1 illustrates alternative splicing of a single gene. In this case, given gene is composed of 5 exons and has 3 annotated isoforms, each leading to synthesis of different protein.

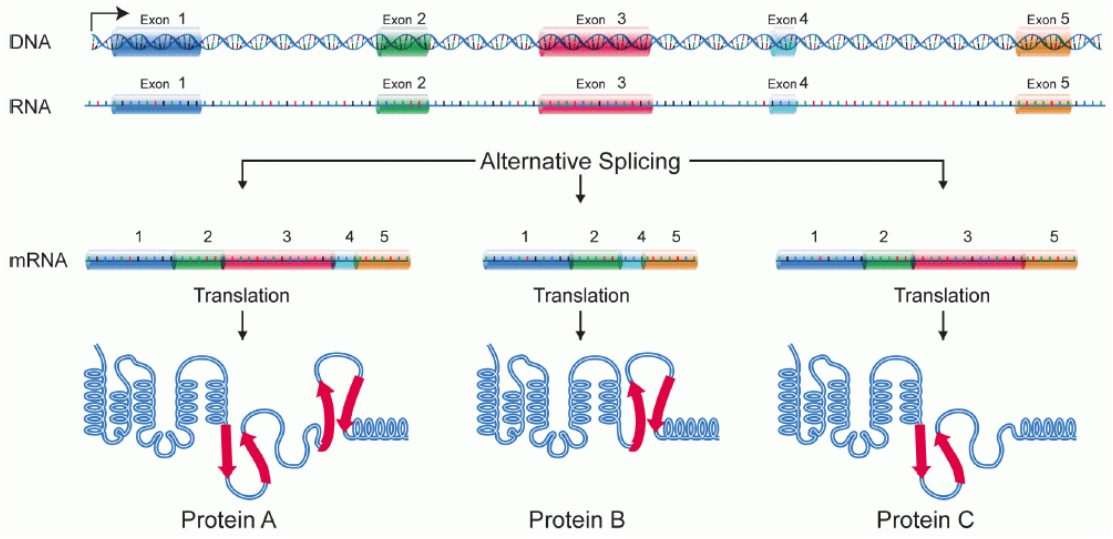

Figure 1.1 Alternative splicing. Source: Wikipedia[5]

1.3 Sequencing

DNA, RNA, and proteins, all of them very important biologically active substances, are all polymers, being composed of distinct monomers, connected in a chain. Determining sequence of these polymers has been an important task in molecular biology, as such information can provide insight into many aspects of ongoing processes in biological systems. For example, being able to determine DNA sequence can be useful for diagnosing a genetic disorder, understanding of principles of many DNA-related diseases and pathogens, or for identification in forensic cases and many other fields. Knowing a sequence of amino acids for a given protein can help describing function of given protein based on sequence homology, or using the technology of bioengineering, to produce such protein in a bacteria.[6]

1.3.1 Approaches

In time, two major approaches for sequencing have been developed. First can be called sequencing by fragmentation and second sequencing by synthesis. The first was used by Sangers in 1950 to determine the sequence of insulin. The basic idea of this method is fragmentation of given polymer, followed by determining sequence of each fragment. Based on the overlaps between fragments, it is then possible to reconstruct the whole sequence.[7] This method often uses electrophoresis to identify the fragments, which can be time consuming. Second approach is based on detecting a signal, that is related to incorporation of a nucleotide in a chain, that is being synthesized by a polymerase[8]. That means, that in the process of sequencing, a new molecule is being created with the original molecule serving as a template. This approach is only usable for sequencing of nucleic acids, as synthesis of proteins is much more complicated. Sequencing by synthesis is typically being used for sequencing DNA, as RNA can be converted to DNA with the reverse transcriptase enzyme.

Sequencing by synthesis is currently used much more than sequencing by fragmentation thanks to development of special technologies allowing higher speed of sequencing and sequencing of many DNA fragments in parallel. As current main approach to sequencing, DNA sequencing by synthesis deserve to be explained in more detail.

1.3.2 Library preparation

Prior to the sequencing itself, the sample, that is to be sequenced must be prepared. This step is usually referred to as library preparation. The typical workflow of library preparation for any current sequencing platform is as follows. First of all, the DNA has to be isolated from biological sample. Once pure DNA is obtained, it is often necessary to break the DNA into smaller fragments for the sequencing platform. Next specialized adapters are added to the fragments. These adapters allow binding of sequenced fragments to a flow cell, or other medium used by given sequencing platform. The adapters may also serve as binding sites for primers, necessary for activity of DNA polymerase. In this step, unique sequences can be added to the sequenced fragments, marking fragments, that originate from a single sample. This is called barcoding and allows sequencing of multiple libraries at once. Last, optional step of library

preparation is clonal amplification of studied DNA. This is mostly done by the polymerase chain reaction (PCR) method.

1.3.3 Methods of sequencing by synthesis

There are multiple methods, that use this principle of sequencing by synthesis, where each method detects a different signal. In pyrosequencing[9], the measured unit is pyrophosphate, that is being released during the incorporation of nucleotide. The pyrophosphate is then processed by enzymes ATP sulfurylase and luciferase, resulting in production of light, that is being measured. Ion semiconductor sequencing is another method for sequencing, developed by Ion Torrent Systems Inc. (USA, Gilford). The method is based on release of hydrogen during incorporation of nucleotide in a chain, which results in change in pH of the solution. This change is then recognized by ion sensor. The benefit of this approach is, that the change of pH can be detected directly, so no other enzymes are needed. The last approach is attaching a fluorescent marker to base, that is being added in given step, which acts as a reversible terminator. After the base is incorporated, the signal is measured using a laser excitation and fluorescence detection and the base is then converted to standard nonterminating nucleotide. This approach is known as terminator sequencing.

1.3.4 Next generation sequencing

Some of these methods are being used in so called “high throughput”, “next generation” or “massively parallel” sequencing (HTS, NGS, MPS), where millions of sequences are sequenced in parallel. The fragments of sequenced DNA are attached to solid base, where the amplification process takes place. The base can be for example flow cell (used in platforms manufactured by the Illumina Inc. company, USA, San Diego), or nanobeads (used in SOLiD sequencing, developed by Life Technologies Corporation, currently owned by Thermo Fisher Scientific Inc.). The amplification is necessary for the fluorescent signal to be detectable. The process of amplification is again platform specific, for example on Illumina platform it is done by bridge amplification. Adapters, that bind single strand DNA fragments to flow cell are added on both ends of the fragments, which leads to bending. DNA polymerase then synthesizes the second strand on such fragment. The double stranded fragment is then denaturised with each strand remaining bound to the flow cell on a single end and the whole process repeats. This way small areas made of identical DNA fragment are created, emitting strong signal when a new nucleotide has been incorporated during the sequencing.

After the amplification is done, the sequencing is performed in cycles, where either a single type of dNTP is added on the sequencing matrix (pyrosequencing), or a mixture all nucleotides is added (fluorescent sequencing). This step and the following steps are performed in parallel on millions of template fragment sequences allowing for massive amount of sequences being determined in single experiment. In the first two methods (pyrosequencing and ion semiconductor sequencing) described, the signal of incorporation of a given base is observed directly when the substrate is added, in the fluorescent based sequencing method the signal has to be recorded in a separate step. In all methods the substrate, that has not been incorporated into the fragments has to be washed away before next cycle can start. The number of cycles performed defines the

length of reads produced by given sequencing run, for example Illumina NextSeq 550 System allows maximum read length of 150 base pairs. Most of sequencing platform offer so called paired end sequencing, which means, that each fragment of DNA is sequenced from both ends, producing a pair of reads, instead of a single read per fragment. Having paired-end data allows more precise mapping of reads to reference genome, as additional information in form of relative distance from the second read in pair is provided.

1.3.5 Third generation sequencing

Not all methods rely on amplification of the sequenced molecule. These methods, often called "third generation sequencing" are able to determine the sequence from a single DNA molecule. Apart from methods based on synthesis, this group of methods contains also approaches based on a different principle. For example, nanopore sequencing[10] is a technology, that does not rely on synthesis, or labelling bases. The sequencing unit consists of a membrane with nanopores, through which the sequenced DNA molecule is being pulled. As the molecule transverses through the pore, the change in ionic current passing through the pore is being measured. The nature of the difference is characteristic for each base, allowing the sequence to be determined.

Each sequencing platform has its own strengths and weaknesses, that are to be considered before selecting one for a particular experiment. Currently, the most frequently used is Illumina platform[7], producing millions of short reads of length in lower hundreds of bases with 99,99% accuracy. In order to reconstruct the original sequence from short sequenced reads, further data processing and analysis is necessary. The steps of this process will be described further in this work.

1.3.6 Applications of NGS

Obtaining the sequence of DNA was a very important step in modern biology, as it can be used in great number of fields of study. It is theoretically possible to obtain the sequence of whole genome of any species. Comparison of genomes can help with building phylogenetic trees of species and solving systematic and evolutionary biology questions. Such whole-genome sequencing (WGS) can be also used to study intra-species diversity, polymorphisms and detect mutations and other genetic aberrations. WGS also has the potential to be used as a diagnosis tool by sequencing genomes of microorganisms responsible for disease[11]. The main challenge of WGS is the correct reconstruction of genome from short reads provided by sequencing platform.

Compared to WGS, other approaches limit the width of sequencing to only a portion of the genome. Such approaches are be called amplicon sequencing as sum of selected amplified regions are sequenced. One special case is whole-exome sequencing (WES), where only the parts of genome corresponding to exons of protein-coding genes are being sequenced. The method can be extended to also capture exons of genes coding other nonprotein functional elements[12]. The first step of this method is selectively capturing only target fragments from the DNA obtained for given sample. This is done by hybridizing the fragments containing exons to oligonucleotide probes, which are bind to magnetic probes, allowing selective filtering. WES has similar

applications as WGS, but the amount of material sequenced is greatly reduced, reducing cost and time needed for the experiment.

Sequencing can be used in study of binding sites used by DNA-binding proteins. Chromatin immunoprecipitation followed by sequencing (ChIP-Seq)[13] is the method used for this type of analysis. It consists of reinforcement of the protein-DNA bond, fragmenting of the DNA and then selective filtering using antibodies targeting specific proteins. Finally, the bond is undone and the resulting DNA is sequenced. DNA-binding proteins have an important role in gene expression, as the effect of binding of specific protein may affect not only adjacent genes, but also genes located further in the sequence.

Another method for studying regulatory elements is DNase-seq, where sites, that are hypersensitive to ligation by enzyme DNase I are sequenced[14]. In normal conditions, DNA in eukaryotic cell is wrapped around histone proteins, the complex of DNA and histone is called nucleosome. Nucleosomes form a higher structure called chromatin, which is formed into chromosomes. This hierarchical structure allows compaction of long DNA molecule into very limited space inside the nucleus. DNA in nucleosomes is less accessible to transcriptional factors and enzymes such as DNase I, so when the enzyme is added in solution containing DNA, only nucleosome-depleted DNA is fragmented. Nucleosome-depleted DNA is most likely targeted by transcriptional factors and serve as active regulatory elements. In DNase-seq the parts of DNA, that have been digested by DNase I are amplified and sequenced by a NGS platform. This way the position of regulatory sequencies can be determined allowing better understanding of the regulatory pathway. A less time-consuming alternative to DNase-seq is ATAC-seq (Assay for Transposase-Accessible Chromatin with high-throughput sequencing)[15]. The difference is that instead of digesting DNA by adding DNase I enzyme, the DNA is cut by hyperactive Tn5 transposase, which simultaneously cuts DNA and ligates adapters to the ends of arisen fragment. These adapters are then used for sequencing. It is then again possible to sequence the shorter fragments, which are derived from DNA that was not bound in a nucleosome. It is also possible to sequence the DNA originally protected by nucleosome and determine nucleosome positions in genome. The data can be separated, as the fragments arisen from unprotected DNA are shorter than fragments of “nucleosomal” DNA.

Last of the major types of analysis using sequencing method is analysis of transcriptome by RNA-seq, which will be introduced in next chapter, as it is closely related to rest of this thesis.

1.3.7 RNA-seq

RNA-seq[16], or RNA sequencing is sequencing of RNA molecules. The method can be potentially used on any kind of RNA, but is mostly used on mRNA, with the goal of measuring abundance of transcripts present in given sample, as this can provide information about gene regulation and mutations in given sample. The process follows the same steps as typical DNA sequencing experiment, which are library preparation, sequencing and further analysis with quality control. The only different step, compared to DNA sequencing, is library preparation, as the final product of this step is a DNA library, that is presented to a sequencing platform. The first step of library preparation for an RNA-seq experiment is RNA isolation. After the RNA has been isolated, it is

often necessary to filter out unwanted RNA molecules, typically removing ribosomal RNA, as rRNA is the most abundant type of RNA present in a cell and sequencing it could waste resources and reduce the detection rate of less abundant RNA species. One way of depletion of ribosomal RNA is degrading it by a specific enzyme. Another possibility[17] is hybridizing rRNA molecules to a substrate, such as magnetic beads, and then separating the hybridized and unhybridized molecules. It is also possible to use probes targeting mRNA molecules, instead of rRNA molecules. After obtaining a sample containing only the RNA species of interest, the molecules are fragmented to be of appropriate length for chosen sequencing platform. This step is not necessary in experiments regarding micro-RNA, as these molecules are typically shorter than 200 nucleotides, so no further fragmenting is needed. The last step of RNA-seq library preparation is converting RNA molecules to DNA. This is done by attaching short DNA primers of random sequencies to the RNA molecules. These primers serve as starting point for reverse transcription, transcription of RNA to DNA, using a reverse-transcriptase enzyme.

2 Expression analysis

In recent years, RNA-seq is the most popular technology for measuring gene expression[18], allowing rapid and accurate evaluation of mRNA in given sample. Even though the technology of this process measures directly the transcript-level expression, it is often used to describe gene-level expression. That is because of the grater complexity of quantification at transcript-level resolution, as was described in previous section. Even despite this limitation, many tools for accurate transcript quantification have been created. In this section I am going to describe a standard workflow for measuring gene-level expression using RNA-seq as described in[1] and then I will introduce some tools, that perform similar task on a subgene-level resolution.

2.1 Data types used in expression analysis

In the process of expression analysis files containing distinct information are being used and produced[19]. For many types of information, special file formats have been designed, allowing more efficient storage and usage of given data. Each of these file formats has its own set of rules, which will be discussed shortly.

2.1.1 FASTA

FASTA format is a file format for storing nucleotide, or amino acid sequences. The format name is derived from a sequence alignment software package. Each sequence is stored on two lines in plain text format. The first line starts with '>' symbol and contains information about the sequence, such as ID and description. The second line than contains the sequence itself. Multiple sequences can be present in single FASTA file. These files are often used for reference sequences, or for storing known sequences in a database.

2.1.2 FASTQ

FASTQ format is a simple extension of FASTA format. It is also a text file format, but in addition to information contained in header and sequence, PHRED quality score for each single base is also stored. PHRED quality score of a base call is stored in a single ASCII symbol, as each ASCII symbol is associated with a single numeric value. It is computed as decadic logarithm of probability, that given base call was an error, multiplied by -10. ($Q_{PHRED} = -10 * \log_{10}(P_e)$) As such, a single sequence in FASTQ file is stored on four lines, first two lines are the same as in corresponding FASTA file, third line contains only a single character '+' and on the fourth line, the sequence of ASCII symbols denoting quality of corresponding base is stored. The FASTQ file format has become the standard output format for most sequencing platforms, which typically output a single FASTQ file containing millions of short reads with corresponding base quality. However, the value of PHRED quality score is dependent on the given sequencing platform[19].

2.1.3 SAM/BAM

SAM (Sequence Alignment/Map format) is a file format for storing reads aligned to reference sequence. A SAM file is a tab-delimited file, that consists of two sections, a header section and alignment section. The header section may contain four types of information, each stored in a single line starting with the '@' symbol. The information is stored in the form of tag : value pairs. The description of each section and possible values can be found on the SAMtools website ([20], **http://www.htslib.org/**). The header contains information such as name of the reference, ID of the sequence, sequencing platform used etc. The alignment section contains sequences (for example all sequences from a single FASTQ file). Each sequence is stored on a single line with 11 or more tab-delimited sections. The eleven mandatory sections include the actual sequence that has been aligned, its name (often id of the read from FASTQ file), leftmost position in reference, where the read has been mapped, sequence of the read, CIGAR string and other fields. A CIGAR string is a compressed notation of match status of each base in given read, marking which bases have been matched, or mismatched and where were detected insertions and deletions. In order to reduce the size of SAM files, a binary version, BAM has been introduced. These files contain the exact same information, only in compressed, binary form. SAM/BAM files are typical output format of most tools performing alignment of reads to a reference.

2.1.4 GFF/GTF

GFF (General Feature Format) and GTF (General Transfer Format) -files are used for storing genomic annotation information, typically for reference genomes. GTF format is an extension to version 2 of GFF format. While both GFF2/GTF are widely used, they are deprecated and Currently GFF version 3 is used the most. All formats are text-based, tab-delimited files, where each line describes one feature (e.g. gene, exon…) of the reference. Each feature has 9 columns, that have to be filled with a value, or with the '.' symbol. These values describe: name of chromosome, or scaffold in a form, that is used by Ensembl[21], source, type of the feature, positions, where the given feature starts and ends, relative to the reference, score of the feature, strand, defined as '+' for forward strand and '-' for reverse strand, reading frame and attributes in form of tag-value pairs.

2.2 Gene expression analysis

2.2.1 Quality assessment and trimming

As was mentioned previously, the sequencing using a NGS platform yields a large amount of data in form of short reads stored typically in FASTQ files. Each base of each sequence obtained has assigned quality score by the sequencing platform. Based on these values, the quality of the whole sequence can be assessed and sequences with score lower than selected threshold are removed.[22] The term "trimming" refers to removal of adapter, and other technical sequences, that are not the aim of the sequencing experiment. Multiple tools are available for this task, such as Trimmomatic[23], or fastp[24]. FASTQ[25] is another popular software for FASTA file quality

control, however it does not perform the removal of low-quality reads and artificial sequences. The information provided by FASTQ can be used by another tool to remove sequences selected by FASTQ.

2.2.2 Mapping

The next step of the analysis is mapping reads to a specific locus in genome, from which the particular read is most likely derived from[1], based on a provided reference. The reference is most often the reference genome of given species in GTF/GFF format, but mapping to reference transcriptome is also possible. Reference transcriptome is usually represented by a single FASTA file containing all annotated transcripts for given species. There are many tools for read alignment/mapping of reads (for example TopHat2[26], STAR[27] or BowTie[28]), each having a slightly different approach, but in all of them, an index of either the reference, or the reads is built in order to quickly determine a set of positions, where the read most probably is originated from. After this a more specific algorithm finds the locus with best alignment to the read. The mapping step is typically the biggest bottleneck of every expression analysis in terms of time, as finding the ideal alignment for each read is computationally demanding. Also, the whole process is complicated by technical noise produced during sequencing and biological variability in form of insertions and deletions. Apart from this, present-day alignment tools for RNA-seq data are expected to be able to process both single-end and paired-end data. Another factor, increasing the complexity of mapping RNA-seq reads is the fact, that the DNA, that is being sequenced is derived from mRNA, that has already undergone post-transcriptional modifications. Because of that, many alignment tools have to be splice-aware to be able to correctly determine the origin of reads, that span two different exons, that do not form a linear sequence in reference genome. Another possibility is mapping reads to reference transcriptome. This way, the reads can be mapped by a splice-unaware tool, as the sequences in reference are also post-splicing sequences. This approach however introduces more uncertainty because of high degree of sequence overlap between transcripts, that are formed from same exons. After the mapping is done, the data is usually stored in SAM or BAM files, which contain not only the sequence of given read and position of its origin in reference genome, but also the information about the alignment and its quality.

2.2.3 Quantification of reads

Next step of the analysis is quantifying the amount of reads, that map to each gene. The mostly used approach for this task is taking in account every read, that has the same sequence as any exon of given gene. This approach however omits reads, which are mapped outside of annotated exons. Other approach is taking into account all reads, that are sequentially similar to any part of given gene. Both approaches have to however deal with reads, that can be mapped to multiple locations, this phenomenon occurs mainly with repetitive sequences. These reads are commonly referred to as multireads. One approach how to deal with these reads is discarding them and counting only reads, that are mapped uniquely. However, this method leads to information loss and underestimation of expression of genes containing repetitive sequences. An alternative approach is estimating the coverage based on uniquely mapped reads and assigning multireads based on such estimation.

After counting the reads the results are typically saved in tab delimited files containing ids of genes with corresponding counts. After this step a normalization of the data typically follows, as raw numbers of reads are biased by multiple factors such as length of given gene, or sampling depth of given sample, when performing differential analysis. For these reasons, the data is quantified in the form of reads per kilobase per million mapped reads (RPKM). To obtain RPKM for a single gene first the "per million" scaling factor is computed by summing the total number of reads provided for given sample and dividing that number by one million. Next the number of reads mapped to given gene is divided by the "per million" scaling factor. This step normalizes for sequencing depth of the sample. Finally, the value is divided by length of given gene in kilobases, normalizing the read count for gene length.

2.3 Subgene level expression analysis

When performing the analysis with the goal of quantifying exons, or gene isoforms instead of genes, most of the steps are identical to analysis on gene-level resolution. The main difference is in the step quantifying reads. In the case of measuring exon expression, the task is less complicated, as exons are non-overlapping segments. However alternative spice sites and boundaries are often present, which increases the variability. This issue is often resolved by dividing such exons into parts and each part is then quantified separately[4]. This approach however can lead to uncertainty, similar to that of assigning reads spanning exons present in multiple isoforms to a single transcript.

For transcript expression quantification, multiple tools with different approaches have been developed. Many of those implement EM (Expectation-Maximization) algorithm to decide the transcript of origin for each read[29]. EM algorithm is an algorithm for estimating unknown parameter in given model. In case of transcript expression analysis, the unknown parameter is the origin of given read. The algorithm is a repetitive procedure, in which the parameter is assigned with a set probability, which is then iteratively improved by computing the probabilities of observing given data assuming the set value of the parameter[30].

According to an article from Charlotte Soneson[31] gene-level expression estimates are more accurate than transcript-level. In this article, the Salmon[32] tool was used to estimate transcript expression and featureCounts[33] tool was used to obtain gene-level expression estimates. On top of that, gene level expression was estimated by summing the transcript expression estimates for all transcripts of a single gene. A dataset with known true expression values was used, allowing to measure accuracy of each approach. In the end it seems, that deriving gene level expression from transcript expression estimates is the most accurate possibility. Measuring the expression of transcripts however yields different type of information than measuring gene expression.

3 Tools

In this part, I will introduce current tools, that are available for performing expression analysis on subgene level. The tools can be categorized based on multiple parameters. One option is to divide the tools to those, that perform quantification on already aligned reads (alignment-based) and those, that perform analysis on raw, unaligned reads (alignment-free). This classification has been used further in this work, but only tools, that have their own method for alignment incorporated are considered alignment-free, as some tools offer wrapper functions to perform alignment, which actually use a separate tool for this task. Another way of classification is to divide the tools based on the scope at which the expression is being quantified. Most of the tools measure either exon expression, or transcript/isoform abundance. Apart from these two categories, some tools measure the usage of splice sites, or alternative splicing events, that can be derived from the sequenced reads. These tools will be further referenced as event-based.

Majority of tools presented here are designed for carrying out differential analysis on data, identifying exons, or transcripts, that are significantly more, or less expressed between groups of samples. For the purpose of this thesis, the approaches in methods for differential expression will not be described in detail, more detail will be provided regarding estimating expression of transcripts, or exons, as there lies the greatest difference between different scopes of measuring expression. In Table 3.1 summary of basic information for each tool is provided.

3.1 Alignment-based tools

These tools are not designed to perform the mapping step of the analysis. As such, typical input for these tools is aligned reads in SAM/BAM format.

3.1.1 DEXSeq

DEXSeq[4] is a R/Bioconductor package (current version 1.46.0) for testing for differential exon usage in RNA-seq data. As input, DEXSeq requires aligned reads in SAM/BAM format aligned using a splice-aware alignment tool. Also, DEXSeq needs a file containing the transcript reference in GTF format compatible with genome reference used for alignment. The first step of the analysis is “flattening” the reference. Based on the provided GTF file, exon counting bins are defined. Each counting bin refers to a single exon, or a part of exon, if there are alternative boundaries present. Alternative boundaries is a form of alternative splicing, where only a part of exon spliced this way may be present in some transcripts, whereas other transcripts contain the whole exon. Using these counting bins, expression of each exon is quantified with a function from GenomicAlignments[34] package or by HTSeq python script provided with DEXSeq. In both cases, the reads are associated with an exon bin based on positional information stored in the read and counting bin. If the position of the read spans the position of a particular counting bin, such read is counted towards this counting bin. Different options allow different approaches towards counting reads, that span more counting bins. These results are then stored in a table, containing the number of reads

aligned to each counting bin. On top of this structure, DEXSeq allows further normalization and analysis. It uses generalized linear models to model read counts and assumes, that the number of reads for each counting bin is a realization of random variable, that follows negative binomial distribution, which can be seen as generalization of Poisson distribution. These models are then used in the differential analysis.

3.1.2 JunctionSeq

JunctionSeq[35] is a R/Bioconductor package (current version 1.4.0) for detecting differential alternative isoform regulation between samples, that builds on statistical methods used in DEXSeq. On top of that, JunctionSeq also detects usage of splice junctions and based on these, it is able to detect unannotated isoforms. In order for that to be possible, the data has to be processed by a splice-aware alignment tool, that aligns reads across novel splice sites, such as RNA-Star[27], GSNAP[36] or TopHat2[26]. Thanks to this property, JunctionSeq is said to perform better, than tools, that do not detect unannotated transcripts, as such transcripts can influence the estimation of abundance of known transcripts. Based on this principle, accurate measuring expression levels of transcripts with incomplete annotation is very difficult and JunctionSeq should perform better in such cases than ordinary tools. For input, JunctionSeq requires aligned reads in SAM/BAM format and a reference transcript annotation file in GTF format. It is recommended to use the same reference, that was used for read alignment. To obtain counts, JunctionSeq uses a separate tool QoRTs[37], which returns coverage count on gene-level, as well as on exon-level and splice junction loci-level resolution. QoRTs also calculates variety of quality control metrics on top of all BAM files presented. According to QoRTs vignette[38], the processing of 1 million read pairs takes around 4-7 minutes. The methods for detecting differentialy expressed splice junctions are similar to those, that are used in DEXSeq for exons. In the default setting, JunctionSeq makes use of both junction-level and exon-level counts for differential analysis.

Finally, JunctionSeq also offers a robust visualization toolset to make result interpretation easier. Visualization may allow estimation of the processes occurring in the sample cells, that lead to the observed differential expression. The basic Coverage/Expression plots describe expression levels of a single gene between condition. In this plot, the coverage of all subunits (exons and splice junctions) is described, together with whole gene expression estimation. Visible is also set of known exons in the reference, which allows specification of location of each splice junction. For reference, a graph from JunctionSeq vignette is presented in Figure 3.1. The graph in figure contains expression estimates for each sample, rather than only for each condition.

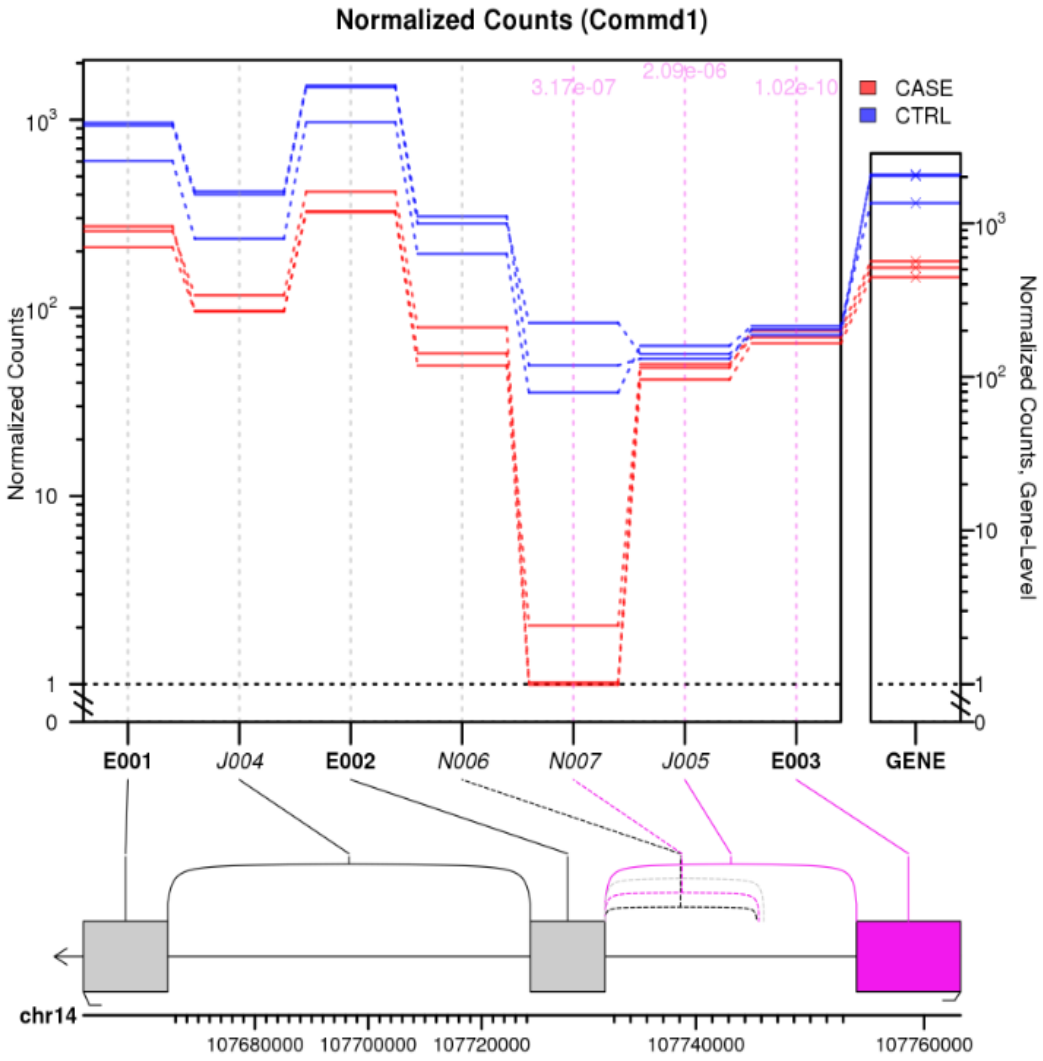

Figure 3.1. JunctionSeq graph of normalized counts across all samples. Source: JunctionSeq Package User Manual[39]

3.1.3 DiffSplice

DiffSplice[40] is another tool for detecting differential transcription. The software is possibly no longer available, as its website does not exist. However, I included the tool in the thesis, because the approach, that was implemented according to the research paper is distinct from majority of other tools. Unlike other tools, DiffSplice does not quantify annotated isoforms, but compares the expression of alternative splicing modules (ASM). The whole analysis starts with construction of a splice graph. This is an oriented acyclic graph, where nodes represent an exonic region and two regions are connected by an edge if there are any reads, that span both of these regions. This graph is build based on reads mapped to reference genome, or it can be built de novo only from raw RNA-Seq reads. In this graph, the ASMs are defined as subgraphs with a single-entry node, single exit node and multiple possible paths between these two. Each ASM represents an observed alternative isoform present in given samples. After all ASMs have been found, the expression of each alternative path in ASMs is estimated

for each sample. A generalized model, that takes into account not only expression of whole exons but takes into account also observed splice junction. For the estimation, it is presumed, that the total number of reads originated from a given transcript, that fall in a particular segment of the transcript, follows a binomial distribution. Based on the estimated expression of ASMs, the estimated expression of the whole gene is computed as mean of expression of all ASMs derived from the given gene.

3.1.4 BitSeq

BitSeq[41] allows estimation of transcript expression and differential analysis using Bayesian approach. It was available as R/Bioconductor package (most recent version 1.40.0, removed from Bioconductor version 3.17). BitSeq expects aligned reads in SAM or BAM format and reference transcriptome in FASTA format. For alignment, Bowtie[28] software is recommended. For estimation of transcript expression levels, a generative model is defined, that models the data as independent observations of individual reads. The observation depends on a noise parameter and on the relative abundance of transcripts fragments. The noise parameter determines the probability of a read being regarded as noise and therefore not considered in the analysis. For all reads, that are considered valid, the sequencing process is being modelled. In this model, another variable assigns reads to transcripts and then the probability of alignment of given read to transcript, or transcripts, it was mapped to, is computed. These probabilities are then used for computing relative expression of each transcript, but also serve as a measure of confidence, which can be used in further analysis. For quantifying transcript expression, BitSeq offers two methods. First uses Markov chain Monte Carlo algorithm, which uses a collapsed Gibbs sampler and assesses abundance of individual transcript based on samples produced this way. The second approach uses a variational Bayesian method to approximate the distribution of relative transcript abundance. The second approach is said to be much faster and more suitable, when the main goal of the analysis is estimating transcript abundance, whereas the first, slower method provides better measure of uncertainty, which is useful in differential expression analysis.

3.1.5 rSeqDiff

rSeqDiff[42] is a R package for differential transcript expression analysis. It is built on top of rSeq, a set of tools for analyzing RNA-seq data. For estimating expression of transcripts, a Poisson model as described in[43] is being solved. The model represents the sequencing process as sampling of reads independently and uniformly from each possible nucleotide. This way, the probability of a read coming from a specific transcript is based on length of the given transcript and number of copies of the transcript in given sample. The latter parameter has to be estimated. Based on the model, the number of reads coming from a specific region of an isoform follows binomial distribution, which can be approximated by Poisson distribution. For this Poisson model, a likelihood function for computing the likelihood of exons in a single gene having a specific expression is defined. This way, the problem, that is being solved is maximum likelihood estimation problem, for which numerical methods, that have not been further explained in the paper, are being used. RSeq offers quantification of reads on both gene and isoform level, both based on a list of transcripts, that serves as reference. It contains a mapping tool SeqMap, which allows performing whole

analysis of RNA-seq data with only this set of tools. RSeqDiff have not been updated since 2015 and official website states, it is still in beta version.

3.1.6 eXpress

EXpress[44] is a tool for efficient quantification of expression. It attempts to assign ambiguously mapping reads to a single sequence. These target sequences can be genes, gene-isoforms, or any other type of sequences. EXpress takes aligned reads in SAM/BAM format and a FASTA file containing the reference sequences. As such, it can be used for other experiments, where the origin of short reads is not certain, such as ChIP-Seq, or metagenomic experiments. To estimate expression of targets, an online version of Expectation Maximization algorithm is introduced. An online algorithm is such algorithm, that processes only a small part of data at given time, before taking a next part. In the context of RNA-seq data, this version of algorithm takes only a single read, based on which the parameters of the assignment likelihood functions are updated. This way the assignment of a single read depends only on the already processed reads. The main advantage of this approach lies in the possibility of processing a large amount of data without the need of keeping the whole dataset approachable in machine memory. As reference, eXpress takes a list of sequences, to which the fragments are to be aligned, typically a set of transcripts. Development of eXpress has however been stopped in 2017.

3.1.7 Cufflinks

Cufflinks[45] is a suite of tools for analyzing RNA-Seq data, which does not rely on existing gene annotation. One of the steps of Cufflinks is assembly of transcriptome directly from presented reads. The input data are fragments aligned by another tool, preferably by a splice-aware aligner, where one fragment represents one single-end read, or a pair of paired-end reads. These fragments are used to construct an overlap graph, based on which the transcripts present in given sample are going to be defined. In this graph, each node represents a single fragment, and two nodes are connected, when their alignment overlaps in the genome. This way, distinct compatibility classes of fragments are defined, each class representing a transcript represented by a path in the overlap graph. This allows Cufflinks to estimate abundance of only transcripts, that are viable for given sample, as the set of transcripts derived from the overlap graph is the minimal set of transcripts needed to “explain” all fragments present. Another benefit is the ability to detect novel gene isoforms, or splice-variants, that haven’t been yet annotated, but quantification based on existing reference is also supported. After assembling the set of possible transcripts, the abundance is estimated using a statistical model of the RNA-Seq experiment, similar to BitSeq and other methods. Based on the model, likelihood function for computing likelihood of transcript abundance is defined and maximum of this function is calculated using a numerical optimization procedure. Cufflinks is the name of suite of several specialized tools, that perform the whole expression analysis, as well as name of one of the tools used. The tools, that form the whole pipeline are Cufflinks, which assembles transcriptomes from RNA-Seq data, Cuffmerge, which combines transcriptomes from different libraries (transcriptomes created by Cufflinks), for quantifying gene and transcript expression, Cuffquant is used, Cuffnorm than performs normalization on the expression estimates and Cuffdiff tests for differential expression.

3.1.8 QuasR

QuasR[46] is a R/Bioconductor package, that performs analysis of sequencing data covering read preprocessing, alignment, and quantification, allowing performing the whole analysis with a single R script. The alignment step is performed by the Bowtie[28] tool, but processing of pre-aligned reads by another tool is also supported. For the quantification step, QuasR allows specification of different genomic intervals, such as genes or exons, which are specified as list of query sequencies. The number of alignments, that overlap a given query region is then quantified. Quantification of individual transcript abundance is not recommended, as QuasR offers only two basic options to resolve ambiguously mapping reads. The first option is counting the read once for each region it aligns to, the second option is based on the list of query regions, where the order of regions defines their hierarchy and each read is counted only towards the first region it aligns to. When performing quantification of genes, by default the second approach is used, as the final expression level is computed as sum of expression levels of all exons of given gene. For quantification of exon expression levels, the first approach is used, which can lead to overestimating expression of exons with common sequence. Apart from alignment and quantification, QuasR also contains qQCReport function, that creates various diagnostic plots, that can be used for estimating the quality of present data. It is possible to visualise quality of alignment, but also to measure properties of raw reads, such as nucleotide frequency and read quality score.

3.1.9 featureCount

FeatureCounts[33] is a program for summarizing aligned reads. It is possible to count genes, but also other genomic features, such as exons, or promotor regions. Overall featureCounts is mainly useful for counting of features, that have low rate of sequential similarity, which makes it unsuitable for estimating expression of gene isoforms, but useful for measuring exon expression. As input, featureCounts expects aligned read in form of SAM/BAM file and a GFF file containing the reference sequence, or multiple reference sequences. The workflow starts with creating a hash table with names of the reference sequences, for fast determining, which sequence is to be used based on the annotation of the read. Then, each reference sequence is divided into 128kb long bins, which are further divided into blocks based on the number of features present in the particular bin. Each block contains the same number of features and the number of blocks in one bin should be nearly equal to the number of features present in these blocks. For better understanding, one reference sequence can represent one chromosome of the given species and a feature can be a single exon. This data structure allows fast localization of the feature, to which given read corresponds. Reads, that correspond to multiple features are either ignored, or counted once towards each feature, they map to. The whole software is written in C++ programming language, which is more time and memory efficient than R or Python. FeatureCounts is available as function in R package Rsubreads, or in UNIX package Subreads.

3.2 Alignment-free tools

Some tools choose to avoid the time-consuming step of aligning each read and try to quantify expression directly from unaligned reads without pinpointing the exact position of their origin. These tools have their own lightweight methods for estimating the origin of reads, so the term "alignment-free methods" can be misleading. These tools tend to be much faster than alignment-based tools while keeping comparable precision[47]. However, these methods apparently tend to have lesser precision when it comes to quantification of small RNA molecules, or RNA molecules, that are expressed only in a small quantity[48].

3.2.1 Kallisto

Kallisto[49] is a quantification tool for RNA-seq data, which aims for a high speed and quality quantification. The idea is obtaining list of transcripts, that are compatible with given reads without mapping each individual base to a specific position. For this task Kallisto uses structure called de Bruijn graph. De Bruijn graph is a directed graph, where each vertex represents a sequence of symbols of length k. Two vertices are connected by a directed edge if the sequence of the first vertex starting at second position is the same as the sequence of the second vertex starting on first position and ending at position k-1. Each transcript is then represented as a path in de Bruijn graph constructed from reference transcriptome.

Inevitably, some k-mers are going to be associated with multiple transcripts. This leads to colouring of the de Bruijn graph, this means, that nodes are assigned colours where each colour corresponds to a specific transcript. Number of colours of a specific k-mer is then called k-compatibility class and a linear set of connected nodes with identical colouring is called a contig. Kallisto then creates a hash table, that maps each k-mer to the contig, where the given k-mer is present, along with position inside the contig. This structure is called kallisto index.

Reads are then pseudoaligned by taking the intersection of k-compatibility classes of each k-mer in the read and of the corresponding entries in kallisto index. To further reduce the time needed for this step, kallisto index also stores the position of end of each contig. This way when a k-mer is contained in a contig, Kallisto can check, if the last k-mer of the given contig is also present in the read that is being processed. If it is, Kallisto presumes that the k-mers between the first hit and the last k-mer of the contig are also present in the read and does not perform hash-lookups on these k-mers.

For quantification, Kallisto uses Expectation Maximalisation algorithm to optimise a likelihood function for RNA-seq. The function it uses is:

$$L(\alpha) \propto \prod_{f \in F} \sum_{t \in T} y_{f,t} \frac{\alpha_t}{l_t} = \prod_{e \in E} \left(\sum_{t \in e} \frac{\alpha_t}{l_t} \right)^{c_e}$$

Where F is set of fragments and T is set of transcripts, l_t is effective length of given transcript and $y_{f,t}$ is a compatibility element, that has the value of 1, if fragment f is compatible with transcript t, or has value of 0 otherwise. α_t are the parameters, which denote the probabilities of selecting fragments from transcripts. The latter part of

the equation describes the likelihood with help of equivalence classes. Here the number c_e stands for number of counts in equivalence class e. This way it is possible to perform the computation on smaller amount of data, as there are usually only hundreds of thousands of equivalence classes, but tens of millions fragments.

To assess the reliability of abundance estimates, Kallisto also performs bootstrapping. Bootstrapping is a statistical method, that uses newly generated data based on the observed data to estimate properties of the observed dataset. Kallisto performs this based on the equivalence class counts. Once the pseudoalignment of the N original fragments is completed, N artificial counts are generated. Each count belongs to an equivalence class and the probability of a count from a given class being sampled is proportional to number of observed fragments belonging to that equivalence class. Using the EM algorithm, Kallisto than computes the transcript abundances of the new samples. The generated samples are then stored in a compressed file, which can be used by other tool, sleuth[50] in downstream analysis.

3.2.2 Salmon

Salmon[32] is second alignment-free tool for estimating transcript abundance. Unlike Kallisto, it offers the possibility to quantify mapped reads in the form of SAM/BAM file, apart from being able to perform its own version of fast mapping-like procedure, called quasimapping. For both procedures, Salmon has to be provided with reference transcriptome containing transcripts, that are to be quantified, same as Kallisto. When quantifying the mapped reads from SAM/BAM, the reads have to be mapped to the same transcriptome, that is provided. Salmon uses dual-phase statistical inference procedure, in which a probabilistic model of the sequencing experiment is build. The first phase of the procedure estimates initial expression levels and model parameters using an online algorithm. The second phase then refines the expression estimates. Using this model, Salmon claims to consider information not used by Kallisto in the quantification process. With this information Salmon is able to correct for not only sequence-specific bias, but also GC-content and positional biases.

For mapping, similarly to Kallisto, salmon also prepares a data structure, that helps it to determine the location of each read. Salmon does this by creating a suffix array from the reference transcriptome. Suffix array is a sorted array of suffixes of the given text. The suffixes are ordered in alphabetical order and are stored in the form of indices referring to the original text, which allows it to be more memory efficient, than for example suffix trees, another data structure that can be used for the same tasks. Suffix arrays are often used in data compression and text to text comparisons. Apart from suffix array salmon also creates a hash table, that maps k-mers of sequence to corresponding positions in the suffix array.

The mapping is than performed in several steps. First, the read is scanned from one direction until a k-mer, that is present in the hash table is found. From the hash table, the range of all suffixes, that contain the specific k-mer is retrieved. It is an interval thanks to the fact, that the suffixes in the suffix array are sorted. Next, starting at the end of found k-mer, the longest part of the read, that exactly matches the reference suffixes is found. This section is called the maximal matching prefix (MMP). After finding a mismatch, salmon skips ahead 1 k-mer and repeats the process, until the end of the read is reached. Finally, the set of transcripts, that contain all MMPs found

are considered to be the mappings of given read. The output of the mapping step is not only list of transcripts, from which the read likely originated from, but also the orientation of the read and the positions of the transcripts in reference transcriptome. These information are used later during quantification.

Name	Approach	Reference used	Input data type	Year released	Citation
BitSeq	Isoform-based	Genome	Aligned reads	2012	[41]
Cufflinks	Isoform-based	None /Genome	Aligned reads	2010	[45]
DEXSeq	Exon-based	Transcript reference	Aligned reads	2012	[4]
DiffSplice	Isoform-based	None	Aligned reads	2013	[40]
JunctionSeq	Exon-based	Genome	Aligned reads	2016	[35]
Kallisto	Isoform-based	Transcriptome	Raw reads	2016	[49]
QuasR	Exon-based	Genome	Aligned/Raw reads	2015	[46]
Salmon	Isoform-based	Transcriptome	Raw reads	2017	[32]
eXpress	Isoform-based	Transcriptome	Aligned reads	2013	[44]
featureCount	Exon-based	Genome	Aligned reads	2014	[33]
rSeqDiff	Isoform-based	Transcriptome	Raw reads	2013	[42]

Table 3.1 Methods overview

4 Results

I have tested three tools used for subgene-level expression analysis, DEXSeq (v. 1.40), Kallisto (v. 0.46.2) and Salmon (v. 1.9.0). The tools were chosen because they are of the most cited tools in studies related to expression analysis. The data used for this testing was set of RNA-seq based expression values of childhood acute lymphoblastic leukaemia (ALL) patients. Acute lymphoblastic leukaemia is a type of cancer, where a large number of undeveloped lymphoid cells is being produced. It is also the most frequent type of childhood cancer[51]. A number of genetic markers associated with different types of ALL has been found, allowing better diagnosis and treatment selection. RNA-sequencing plays a key role in detecting fusions of genes, or deletions, which can act as drives for development of ALL[52]. The advantage of using RNA-seq is the ability to detect fusions, that have not been previously annotated as clinically active. Apart from this, detection of up- or down-regulation of gene expression can carry a valid information about mutation in regulatory sequence responsible for the change in expression.

All scripts used for this part of thesis are available on my Bitbucket site (https://bitbucket.org/klodaf/kloda_bcl/src/master/). Because of the nature of data used, it is not possible to provide the data.

4.1 Materials and methods

The data set from 16 ALL samples were provided by Department of Pediatric Hematology and Oncology, 2nd Medical Faculty, Charles University and Motol University Hospital.

Reads were sequenced by NextSeq platform (Illumina, USA, San Diego) and aligned using TopHat2 (version 2.1.1) to GRCH37.75 Human reference genome. For each sample, between 84 - 244 million reads were provided, with mean of 164,690,544 and median 159,572,607. All files underwent quality control procedure using FastQC program. Based on this procedure, mean of base quality across all positions of all reads was higher than 30. According to QualiMap software, on average 92% (with median 92.25%) of reads were successfully aligned to reference genome, with 45.7% of reads being duplicates on average (with median 44.12%).

For each sample, two FASTQ files containing raw paired end reads and single BAM file containing aligned reads were provided. Salmon and Kallisto accept raw data in form of FASTQ file, while DEXSeq expects already aligned reads in SAM/BAM file format, as DEXSeq does not cover mapping of reads.

4.1.1 Reference

As a reference, Kallisto and Salmon used human GRCh37.p13 transcriptome provided by Ensembl[21] in form of FASTA file containing 180253 separate transcripts. DEXSeq used human GRCH37 reference genome transcript annotation also from Ensembl, quantifying 644359 counting bins (exons, or parts of exons).

4.1.2 Methods

Each tool (Kallisto, Salmon, DexSeq) was run by a separate R script. The structure of all scripts is similar, as each method needs a form of reference and a path to directory containing input data. The scripts were written so that they can be easily run with only changing the input data path. All methods were run with default setting.

4.2 Results

Running time of Kalisto (5.5 hours) and Salmon (71 minutes) was significantly lower, than DEXSeq (21.8 hours) and Salmon turned out to be even faster than Kallisto. As DexSeq accepts SAM/BAM files as input, the reads had to be mapped using an alignment tool. This step was not included in the running time of DexSeq. Based on this, it is obvious, that the difference between processing speed of alignment-based method and alignment-free methods is truly significant.

Kallisto and Salmon both provide output in form of tab-delimited table, where each row contains a single transcript, its length, number of reads that mapped to this transcript and normalized expression of given transcript. The expression is normalized to transcripts per kilobase million (TPM) units. This normalization considers length of given transcript and sequencing depth of the whole sample similarly to RPKM. The computation of TPM is similar to computation of RPKM. To obtain TPM, the number of mapped reads is first divided by length of given transcript in kilobases and then by "per million" scaling factor. In computation of RPKM those two steps are in the opposite order.

Because the output of both tools is the same, direct comparison between these two methods is possible. Despite having the exact same reference file, results provided by Salmon contain 167,268 transcripts, whereas Kallisto provides estimated expression of all 180,253 transcripts provided. In defaults setting, all transcripts, that are sequentially identical to another transcript are removed by Salmon during index creation.

Results of both methods can be seen in Figure 4.1. Results from both methods were highly correlated (average correlation coefficient is 0.99). Each dot in the graph represents a single transcript. The scales are transformed by common logarithm, because of large number of low values. To perform log transformation, transcripts with zero value from at least one method have been removed. Mean value of number of transcripts removed this way for a single sample is 80,729. Majority of removed transcripts had estimated expression value 0 in results provided by both tools, the average is 60,040 of such transcripts per sample. Interestingly, Salmon had identified significantly more transcripts as non-expressed, despite these transcripts having a positive value in results provided by Kallisto. On average 16,978 of such transcripts per sample has been found. On the other hand, only 3710 transcripts on average have been estimated to have zero counts by Kallisto, where Salmon had estimated a different value. The average values have been computed as arithmetic mean. As can be seen from the graph, the results differ more in smaller values. Further differential analysis is necessary in order to determine importance of these differences.

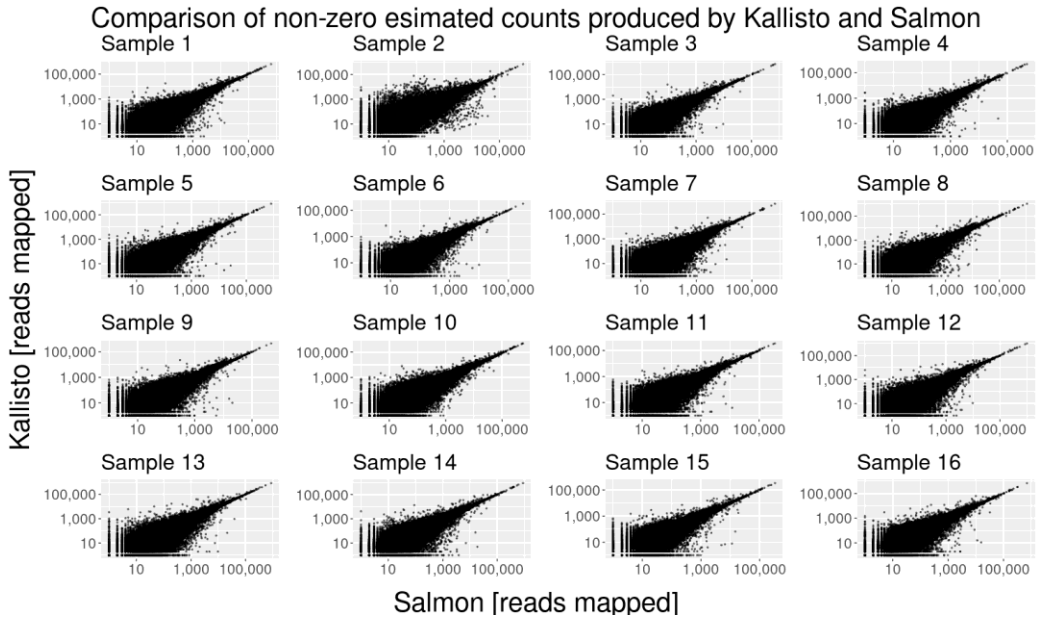

Figure 4.1. Comparison of expression levels estimate by Kallisto and Salmon

To compare DEXSeq results with the other tools, it would be best to have the expression estimates transformed to a common level, for example to gene-level expression, as it is not possible to determine transcipt-level expression from exon-level expression and vice versa. To provide a comparison of both approaches, I present estimated values for gene NRAS (Ensembl ID: ENSG00000213281). The gene is a member of the Ras gene family and codes the NRAS enzyme. This is a gene composed of 7 exons and has only 1 annotated transcript, which is composed of all 7 exons present. Thanks to that, the transcript-level estimate provided by Kallisto and Salmon can be expected to have the same value as the expression values for all 7 exons summed. Results of this comparison can be seen in Figure 4.2. Each histogram represents a single sample. The values are in reads mapped to the given transcript, or exons.

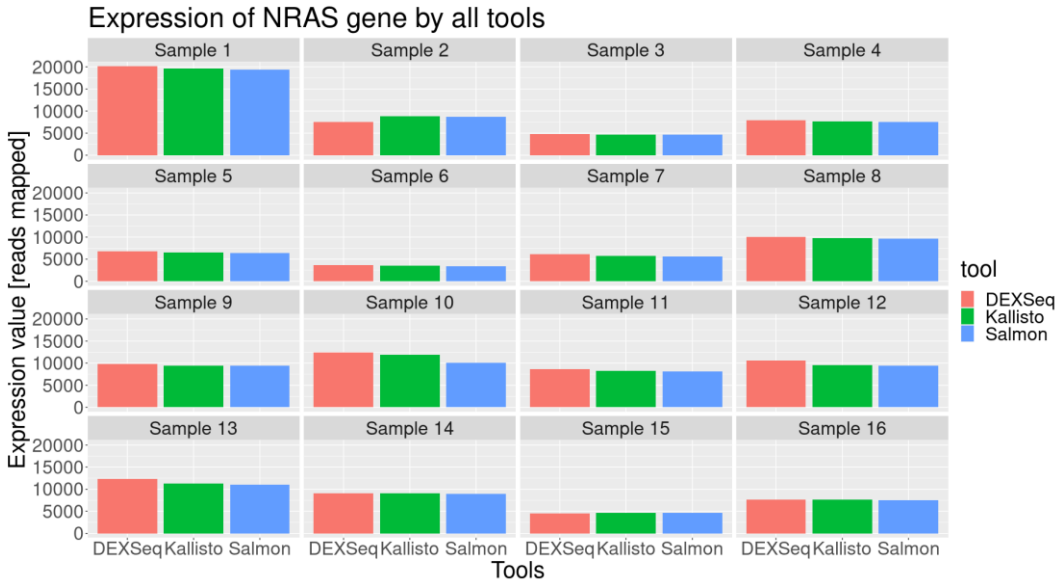

Figure 4.2. Histograms of expression values for NRAS gene by tools

It is visible, that both isoform-based tools offer more similar results and that the values provided by DEXSeq differ from those provided by Kallisto and Salmon, mostly yielding slightly larger values that the two isoform-based tools. To allow quantitative comparison, I have computed means of difference between each pair of tools. The mean difference of values between Kallisto and Salmon is 206.3125 reads, with median of 89.50, mean difference between Kallisto and DEXSeq is 432.1875 reads, median 345.5 and mean difference between Salmon and DEXSeq is 619.5 reads, with median 423.5. As the values differ significantly between samples, the numerical values of overall averages are not very informative, but it illustrates the difference between tools.

As final comparison, a density graph describing the distribution of values produced by all method is provided in Figure 4.3. It is obvious, that the distributions are quite similar, with Salmon and Kallisto showing grater similarity. Unfortunately, it is impossible to objectively evaluate precision of each tool due to the lack of known ground truth.

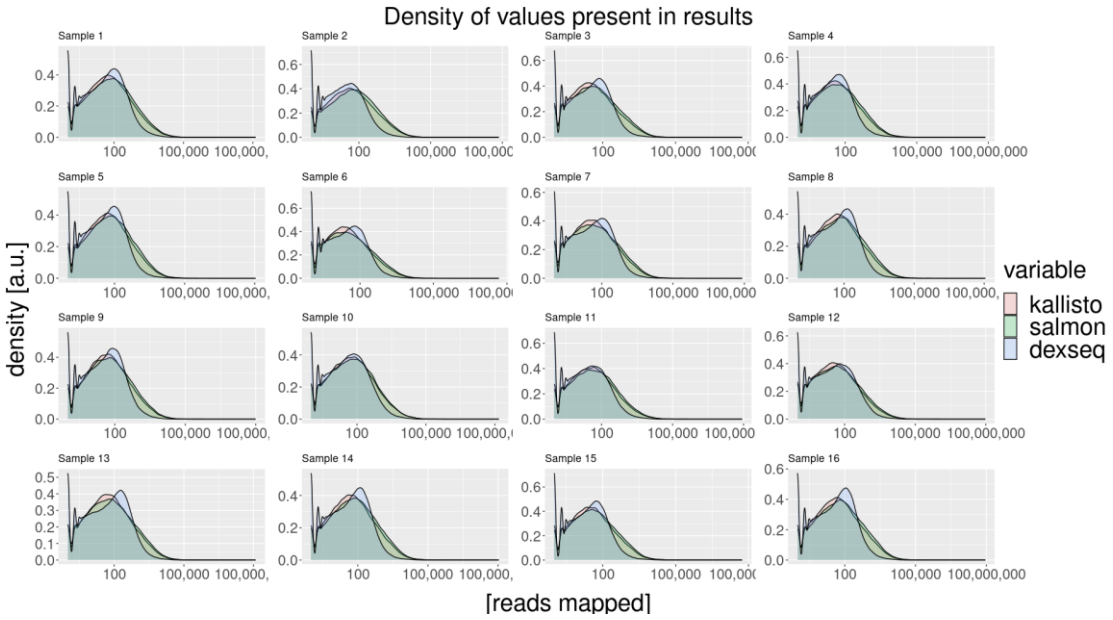

Figure 4.3. Density plot of result values

Conclusion

Expression analysis is an important step in number of different experiments. Even despite its importance, a single best approach for analysis of related data has not been decided. Analysis of exon and isoform expression both offer an additional level of information compared to gene expression analysis, which may be helpful in many fields of study, but may not be necessary in others. Additionally, measuring expression on level of individual transcripts introduces greater uncertainty in the analysis of sequencing data, but apparently this setback has been well handled with the usage of statistical models.

The range of tools currently available for expression analysis is quite large and none has been proven to perform significantly better than the others. Development of so-called “alignment-free” tools such as Kallisto or Salmon offers the opportunity to significantly reduce running time needed for the analysis. However, the older and proven alignment-based approaches remain as a viable option for many experiments.

By testing three selected tools, I have illustrated, that even despite each tool being different, the results are quite similar. The greatest difference between selected tools has proven to be the time necessary for the analysis, in which the alignment-free are decisively superior.

Bibliography

1. Finotello, F. & Camillo, B. D. Measuring differential gene expression with RNA-seq: Challenges and strategies for data analysis. *Briefings in Functional Genomics* **14,** 130–142 (2015).

2. Yi, L., Pimentel, H., Bray, N. L. & Pachter, L. Gene-level differential analysis at transcript-level resolution. *Genome Biology* **19,** 53 (2018).

3. Mehmood, A., Laiho, A. & Elo, L. L. Exon-level estimates improve the detection of differentially expressed genes in RNA-seq studies. *RNA Biology* **18,** 1739–1746 (2021).

4. Anders, S., Reyes, A. & Huber, W. Detecting differential usage of exons from RNA-seq data. *Genome Research* **22,** 2008–2017 (2012).

5. National Human Genome Research Institute. DNA alternative splicing. (2014). Available at: **https://commons.wikimedia.org/wiki/File:DNA_alternative_splicing.gif**.

6. Riggs, A. D. Bacterial production of human insulin. *Diabetes Care* **4,** 64–68 (1981).

7. Shendure, J. *et al.* DNA sequencing at 40: Past, present and future. *Nature* **550,** 345–353 (2017).

8. Pareek, C. S., Smoczynski, R. & Tretyn, A. Sequencing technologies and genome sequencing. *Journal of Applied Genetics* **52,** 413–435 (2011).

9. Harrington, C. T., Lin, E. I., Olson, M. T. & Eshleman, J. R. Fundamentals of pyrosequencing. *Archives of Pathology & Laboratory Medicine* **137,** 1296–1303 (2013).

10. Wang, Y., Zhao, Y., Bollas, A., Wang, Y. & Au, K. F. Nanopore sequencing technology, bioinformatics and applications. *Nature Biotechnology* **39,** 1348–1365 (2021).

11. Köser, C. U. *et al.* Routine use of microbial whole genome sequencing in diagnostic and public health microbiology. *PLoS Pathogens* **8,** e1002824 (2012).

12. Warr, A. *et al.* Exome sequencing: Current and future perspectives. *G3 Genes|Genomes|Genetics* **5,** 1543–1550 (2015).

13. Park, P. J. ChIP–seq: Advantages and challenges of a maturing technology. *Nature Reviews Genetics* **10,** 669–680 (2009).

14. Song, L. & Crawford, G. E. DNase-seq: A high-resolution technique for mapping active gene regulatory elements across the genome from mammalian cells. *Cold Spring Harbor Protocols* **2010,** pdb.prot5384 (2010).

15. Buenrostro, J. D., Wu, B., Chang, H. Y. & Greenleaf, W. J. ATAC-seq: A method for assaying chromatin accessibility genome-wide. *Current Protocols in Molecular Biology* **109,** (2015).

16. Chu, Y. & Corey, D. R. RNA sequencing: Platform selection, experimental design, and data interpretation. *Nucleic Acid Therapeutics* **22,** 271–274 (2012).

17. Chen, Z. & Duan, X. Ribosomal RNA depletion for massively parallel bacterial RNA-sequencing applications. 93–103 (2011). doi:**10.1007/978-1-61779-089-8_7**

18. Liu, Y., Wang, J., Wu, S. & Yang, J. A model for isoform-level differential expression analysis using RNA-seq data without pre-specifying isoform structure. *PLOS ONE* **17,** e0266162 (2022).

19. Zhang, H. Overview of sequence data formats. 3–17 (2016). doi:**10.1007/978-1-4939-3578-9_1**

20. The Sanger Institute. Samtools. *http://www.htslib.org/* (2023).

21. Cunningham, F. *et al.* Ensembl 2022. *Nucleic Acids Research* **50,** D988–D995 (2022).

22. Costa-Silva, J., Domingues, D. S., Menotti, D., Hungria, M. & Lopes, F. M. Temporal progress of gene expression analysis with RNA-seq data: A review on the relationship between computational methods. *Computational and Structural Biotechnology Journal* **21,** 86–98 (2023).

23. Bolger, A. M., Lohse, M. & Usadel, B. Trimmomatic: A flexible trimmer for illumina sequence data. *Bioinformatics* **30,** 2114–2120 (2014).

24. Chen, S., Zhou, Y., Chen, Y. & Gu, J. Fastp: An ultra-fast all-in-one FASTQ preprocessor. *Bioinformatics* **34,** i884–i890 (2018).

25. Cock, P. J. A., Fields, C. J., Goto, N., Heuer, M. L. & Rice, P. M. The sanger FASTQ file format for sequences with quality scores, and the solexa/illumina FASTQ variants. *Nucleic Acids Research* **38,** 1767–1771 (2010).

26. Kim, D. *et al.* TopHat2: Accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome Biology* **14,** R36 (2013).

27. Dobin, A. *et al.* STAR: Ultrafast universal RNA-seq aligner. *Bioinformatics* **29,** 15–21 (2013).

28. Langmead, B., Trapnell, C., Pop, M. & Salzberg, S. L. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biology* **10,** R25 (2009).

29. Zhang, C., Zhang, B., Lin, L.-L. & Zhao, S. Evaluation and comparison of computational tools for RNA-seq isoform quantification. *BMC Genomics* **18,** 583 (2017).

30. Do, C. B. & Batzoglou, S. What is the expectation maximization algorithm? *Nature Biotechnology* **26,** 897–899 (2008).

31. Soneson, C., Love, M. I. & Robinson, M. D. Differential analyses for RNA-seq: Transcript-level estimates improve gene-level inferences. *F1000Research* **4,** 1521 (2015).

32. Patro, R., Duggal, G., Love, M. I., Irizarry, R. A. & Kingsford, C. Salmon provides fast and bias-aware quantification of transcript expression. *Nature Methods* **14,** 417–419 (2017).

33. Liao, Y., Smyth, G. K. & Shi, W. featureCounts: An efficient general purpose program for assigning sequence reads to genomic features. *Bioinformatics* **30,** 923–930 (2014).

34. Lawrence, M. *et al.* Software for computing and annotating genomic ranges. *PLoS Computational Biology* **9,** e1003118 (2013).

35. Hartley, S. W. & Mullikin, J. C. Detection and visualization of differential splicing in RNA-seq data with JunctionSeq. *Nucleic Acids Research* gkw501 (2016). doi:**10.1093/nar/gkw501**

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

37. Hartley, S. W. & Mullikin, J. C. QoRTs: A comprehensive toolset for quality control and data processing of RNA-seq experiments. *BMC Bioinformatics* **16,** 224 (2015).

38. Hartley, S. QoRTs package user manual. *https://hartleys.github.io/QoRTs/doc/QoRTs-vignette.pdf* (2018).

39. Hartley, S. JunctionSeq package user manual. *http://hartleys.github.io/JunctionSeq/doc/JunctionSeq.pdf* (2017).

40. Hu, Y. *et al.* DiffSplice: The genome-wide detection of differential splicing events with RNA-seq. *Nucleic Acids Research* **41,** e39–e39 (2013).

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

42. Shi, Y. & Jiang, H. rSeqDiff: Detecting differential isoform expression from RNA-seq data using hierarchical likelihood ratio test. *PLoS ONE* **8,** e79448 (2013).

43. Jiang, H. & Wong, W. H. Statistical inferences for isoform expression in RNA-seq. *Bioinformatics* **25,** 1026–1032 (2009).

44. Roberts, A. & Pachter, L. Streaming fragment assignment for real-time analysis of sequencing experiments. *Nature Methods* **10,** 71–73 (2013).

45. Trapnell, C. *et al.* Transcript assembly and quantification by RNA-seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nature Biotechnology* **28,** 511–515 (2010).

46. Gaidatzis, D., Lerch, A., Hahne, F. & Stadler, M. B. QuasR: Quantification and annotation of short reads in r. *Bioinformatics* **31,** 1130–1132 (2015).

47. Zielezinski, A., Vinga, S., Almeida, J. & Karlowski, W. M. Alignment-free sequence comparison: Benefits, applications, and tools. *Genome Biology* **18,** 186 (2017).

48. Wu, D. C., Yao, J., Ho, K. S., Lambowitz, A. M. & Wilke, C. O. Limitations of alignment-free tools in total RNA-seq quantification. *BMC Genomics* **19,** 510 (2018).

49. Bray, N. L., Pimentel, H., Melsted, P. & Pachter, L. Near-optimal probabilistic RNA-seq quantification. *Nature Biotechnology* **34,** 525–527 (2016).

50. Pimentel, H., Bray, N. L., Puente, S., Melsted, P. & Pachter, L. Differential analysis of RNA-seq incorporating quantification uncertainty. *Nature Methods* **14,** 687–690 (2017).

51. Terwilliger, T. & Abdul-Hay, M. Acute lymphoblastic leukemia: A comprehensive review and 2017 update. *Blood Cancer Journal* **7,** e577–e577 (2017).

52. Brown, L. M. *et al.* The application of RNA sequencing for the diagnosis and genomic classification of pediatric acute lymphoblastic leukemia. *Blood Advances* **4,** 930–942 (2020).

List of Figures

1.1 Alternative splicing 6

3.1. JunctionSeq graph 18

4.1. Comparison of expression levels estimate by Kallisto and Salmon 27

4.2. Histograms of expression values for NRAS gene by tools 27

4.3. Density plot of result values 28

Attachments

kallisto_analysis.R

R script that performs analysis using Kallisto

DEXSeq_analysis.R

R script that performs analysis using DEXSeq

salmon_analysis.R

R script that performs analysis using Salmon

comparison.R

R script used to produce all comparison between results created by Kallisto, Salmon and DEXSeq. Produces all graphs present in Results section of this thesis.