Differential gene regulation

Master Thesis in Biostatistics (STA495)

by

Joël Meili
14-679-393

supervised by

Prof. Dr. Mark D. Robinson
Dr. Simone Tiberi

Zurich, May 2023

Differential gene regulation

Joël Meili

Version May 16, 2023

Contents

Preface

First, I would like to express my gratitude for my supervisor Dr. Simone Tiberi for his guidance and patience during this thesis. Writing this thesis took longer than I would have hoped, but unfortunately sometimes things are not in our control and there is nothing we can do but to accept that. I am also grateful for the support of family and friends during this time, and I am happy to finally submit this thesis.

Joël Meili
May 2023

Abstract

Single-cell RNA sequencing data has become more and more popular over the past few years. It allows biological questions to be answered that with bulk RNA sequencing could not be answered as cell-specific characteristics can be analyzed. In this thesis we investigate, in each cell cluster (e.g. cell type), how the relative abundance of spliced and unspliced reads varies between experimental conditions. Changes to these relative abundances are directly linked to gene regulation, therefore differences in these proportions are taken as a proxy for differences in gene regulation. Methods that are capable of detecting differences in relative abundance already exist (e.g. *BRIE2* and *eisaR*), however they neglect two sources of mapping uncertainty: i) multi-mapping reads across spliced and unspliced versions of a gene, and ii) reads compatible with multiple genes. Therefore, we propose a novel method, *DifferentialRegulation*, that tackles this issue and evaluate the performance of the existing methods (*BRIE2*, *DEXSeq* and *eisaR*) and our novel approach. For this, we created two semi-simulated data sets using real mouse kidney single-cell RNA sequencing data as an anchor data set to simulate from. From the analysis we conclude that the two methods that account for mapping uncertainty (*DEXSeq* and *DifferentialRegulation*) have significantly higher TPR and better control of FDR than the methods that ignore mapping uncertainty. Additionally, we studied methods robustness by investigating how gene abundance levels, and differential gene expression (a nuisance effect in this analysis), affect the results of each method. We also ran a null analysis on a real data set (where no differences between groups are expected), to study methods' false positive rates. Lastly, we ran a computational benchmark on the mouse kidney data to evaluate the computational burden of each method.

Chapter 1

Introduction

1.1 RNA sequencing

RNA sequencing (RNA-seq) is a technology for detecting and quantifying the mRNA molecules of a biological sample (Stark *et al.*, 2019). The invention of RNA-seq was a major breakthrough in the field of bioinformatics that replaced the use of microarray technology in the late 00's. In comparison to microarrays, RNA-seq allows the whole transcriptome to be fully sequenced whereas microarrays only profile predefined transcripts through hybridization (Rao *et al.*, 2019). Further, various protocols have since been derived from the standard RNA-seq protocol, e.g. single-cell RNA sequencing (Stark *et al.*, 2019).

1.1.1 Bulk RNA-seq

Bulk RNA sequencing allows an aggregated signal to be detected across a mixture of cells. There are many applications for bulk RNA-seq. For example, it can be used to study the differences of expression profiles between tissues in healthy vs disease or across treatments (Stark *et al.*, 2019). However, with bulk RNA-seq one can only estimate the average expression of each gene across a population of cells without regard for the differences between cell types. RNA-seq has several use cases. It can be used to study which genes are turned on in a cell and what their level of transcription is. This allows researchers to understand the biology of a cell at a deeper level. Further, RNA-seq allows the identification of variants and allele specific expression. It is also possible to study the patterns of alternative splicing, which are important to understand their contribution to cell differentiation and human disease.

1.1.2 Single-cell RNA-seq

Single-cell RNA sequencing was developed to overcome some of the limitations of bulk RNA sequencing. With scRNA-seq it is possible to estimate the distribution of expression levels for each gene across a population of cells. This allows new biological questions to be answered where cell-specific characteristics are important. However, there are some caveats with scRNA-seq (Haque *et al.*, 2017). scRNA-seq data in general is much more variable than bulk RNA-seq data due to both higher biological and technical variability at single-cell level (Haque *et al.*, 2017). Figure 1.1 shows the typical workflow of a scRNA-seq experiment. Such a workflow can be broadly summarized by the following steps:

1. RNA extraction
2. Reverse transcription into cDNA
3. Adapted ligation

4. Amplification
5. Sequencing
6. Downstream analysis using bioinformatics tools

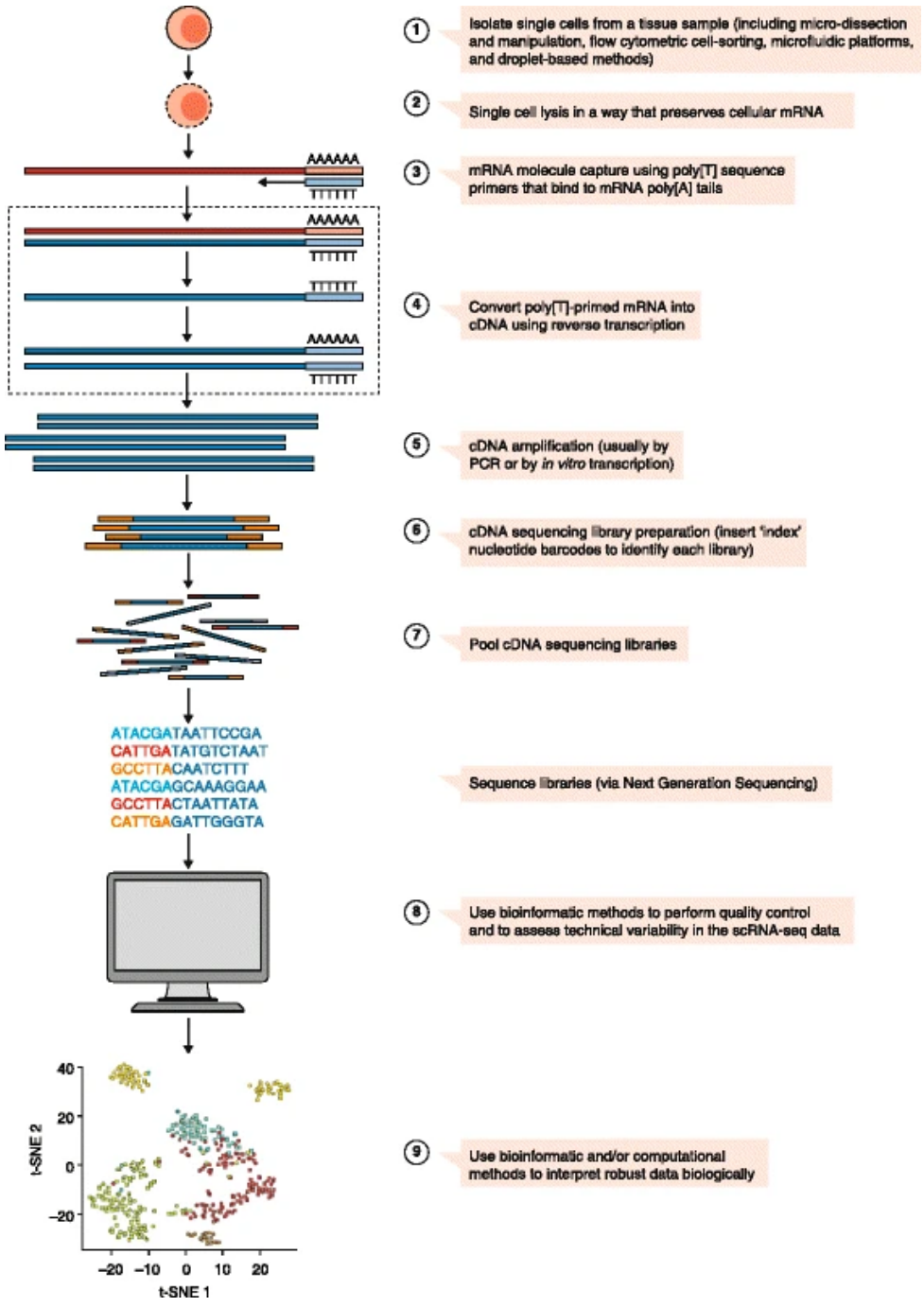


Figure 1.1: This figure describes the general workflow of a single-cell experiment. First, cells are extracted from a tissue sample and then prepared for sequencing. After sequencing bioinformatic tools are used to perform quality control and downstream analyses to describe patterns and interpret data biologically (Haque *et al.*, 2017).

1.1.3 Quantification of single-cell RNA-seq data

scRNA-seq data has distinct characteristics that prevent it from being processed by widely used tools developed for bulk RNA-seq data (He *et al.*, 2022). In general, quantification works by aligning the reads generated from the RNA-seq to the reference genome or transcriptome. There are several tools that allow to do that, notably: *STAR* (Dobin *et al.*, 2013), *kallisto | bustools* (Melsted *et al.*, 2021) and *alevin* (Srivastava *et al.*, 2019). However, there is a difference between the first tool and the other ones. *STAR* is a full aligner and maps reads to the genome; conversely, the other tools, known as quasi or pseudo-aligners, perform a heuristic mapping of reads directly to the transcriptome, and also quantify, via expectation-maximization algorithms, the abundance of transcripts. The difference between an full-aligner and a mapping tool is that the latter does not look for the exact location of the read, as a consequence pseudo-alignment is much faster than full-alignment. Here we focus on *alevin-fry*, and the method we have developed, which will be introduced later, has been built to work on the output of *alevin-fry*.

1.2 Objective

1.2.1 RNA velocity

We investigate spliced and unspliced reads from scRNA-seq data. During transcription, DNA is decoded into precursor messenger RNA (pre-mRNA). Pre-mRNA contains both coding (exons) and non-coding regions (introns). In a next step, introns are removed from the pre-mRNA which leaves only the mature mRNA. Figure 1.2 shows the process from DNA to mature mRNA, where α is the transcription rate, β is the splicing rate and γ is the degradation rate.

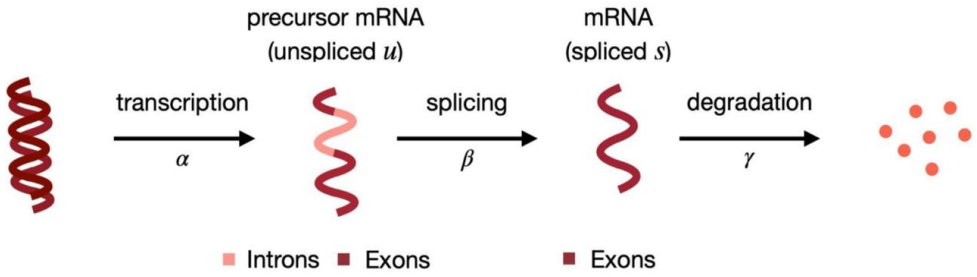


Figure 1.2: This figure describes the process from DNA to mature mRNA. First, DNA is transcribed with transcription rate α into pre-mRNA. As a next step, non-coding regions (introns) are removed from the pre-mRNA with splicing rate β which leaves the mature mRNA. In a last step mRNA is degraded with degradation rate γ (Weiler *et al.*, 2021).

It was assumed that there is a signal (RNA velocity) detectable in scRNA-seq data that could reveal the rate and direction of change of an entire transcriptome (La Manno *et al.*, 2018). To quantify the relationship between the abundance of pre-mRNA and mature RNA, a simple system of ordinary differential equations was assumed (1.1): The solution of such a system at equilibrium can then easily be estimated and used to explore the regulation of genes:

$$\begin{aligned} \frac{du}{dt} &= \alpha - \beta u \\ \frac{ds}{dt} &= \beta u - \gamma s \end{aligned} \tag{1.1}$$

The derivative of the spliced counts is then defined as the RNA velocity of cells. As a result, the balance of spliced and unspliced counts can be used to determine whether a gene is up- or downregulated. If a larger fraction of unspliced counts than expected are present at equilibrium, a gene is likely upregulated. This is because within a short time interval, the newly spliced

mRNA will exceed the amount of spliced mRNA that is degraded. In contrast, if more spliced counts are present at equilibrium than expected, a gene is likely downregulated.

1.2.2 Differential regulation

The abundance of spliced and unspliced reads is directly linked to the regulation of genes and RNA velocities (La Manno *et al.*, 2018). Our idea is to examine how the abundance of spliced and unspliced counts changes between experimental conditions and biological replicates. We translate this intuition into the comparison of two experimental conditions, e.g. healthy vs. disease. Following the same intuitive rationale of RNA velocity, if a gene has a higher abundance of unspliced (spliced) counts in group A compared to group B, then this gene is likely being up-regulated (down-regulated) in group A compared to group B. Thus, we explore the differences in abundance of spliced and unspliced counts to study the differences in regulation between experimental conditions.

If the data contains multiple cell clusters (e.g. cell types), similarly to differential state analyses (Crowell *et al.* (2020) and Tiberi *et al.* (2021)) we will perform differential analyses in each cluster of cells, hence identifying cell-cluster/cell-type specific changes between conditions. The idea of performing differential analyses on the abundance of spliced and unspliced or exonic and intronic reads is not completely novel as there are at least two other methods that achieve that: *eisaR* and *BRIE2*.

1.2.3 Existing methods

eisaR (Stadler *et al.*, 2020) is a R package implementation that allows for the split analysis between exons and introns. It allows one to measure changes in mature RNA and pre-mRNA across different experimental conditions. Ultimately, *eisaR* differential testing is based on edgeR (Robinson *et al.*, 2010). edgeR is a R package that performs differential expression analyses between groups of samples. It implements statistical methods that are based on the negative binomial distribution as a model for count variability.

BRIE2 (Huang and Sanguinetti, 2021) is a Bayesian hierarchical model that is implemented in Python and supports the analysis of splicing processes between spliced and unspliced RNA. There are two modes in which the tool can be used. First, the use of differential alternative splicing (DAS), where the aim is to quantify the proportions of alternative splicing isoforms. Second, the use of differential momentum genes (DMG), where the objective is to quantify the proportions of spliced and unspliced RNA in each gene and each cell.

Originally, *eisaR* and *BRIE2* were developed to analyse all cells, but can easily be adapted to perform cell-type specific differential analyses.

1.2.4 Mapping uncertainty

We can identify two main sources of mapping uncertainty concerning spliced and unspliced reads: i) multi-mapping reads across spliced and unspliced versions of a gene, and ii) reads compatible with multiple genes. In fact, it has been shown that many reads (5-40%) map to multiple genes (Dharshini *et al.* (2020), McDermaid *et al.* (2018)). In our real data analyses (see Section 3), we found approximately 20-30% of such multi-mapping reads across genes. We additionally found that a significant fraction of reads (6-19%) are compatible with both S and U versions of a gene. Therefore, the estimated spliced and unspliced counts carry a substantial amount of uncertainty, which should be accounted for in downstream analyses. However, both *eisaR* and *BRIE2* use estimated spliced and unspliced counts and neglect the mapping uncertainty. In this thesis, we propose two approaches that account for such mapping uncertainties.

Chapter 2

Methods

2.1 Alignment and quantification with *alevin-fry*

Alevin was developed to tackle the computational challenges that come with scRNA-seq data and to provide a tool that supports technologies other than 10x Genomics. *Alevin* works in two steps. First, it parses a read file that contains the cellular barcode and a unique molecule identifier to generate a frequency distribution of observed barcodes. Second, it maps the reads to the transcriptome and generates a cell-by-gene count matrix. *Alevin-fry* (He *et al.*, 2022) was designed to be the successor to alevin and achieves similar accuracy at significantly lower computational costs. It generates a permit list for cellular barcodes that will be quantified in subsequent steps. By using a multi-thread approach, *alevin-fry* filters and collates the mapping records for permitted cellular barcodes to produce a representation optimized for quantification (He *et al.*, 2022). We use *alevin* and *alevin-fry* for our analyses, in particular we focus on *alevin-fry* as it outputs unspliced, spliced and ambiguous (USA) counts and equivalence classes (EC) that are required by the approaches we propose (i.e. USA counts for *DEXSeq* and ECs for *DifferentialRegulation*).

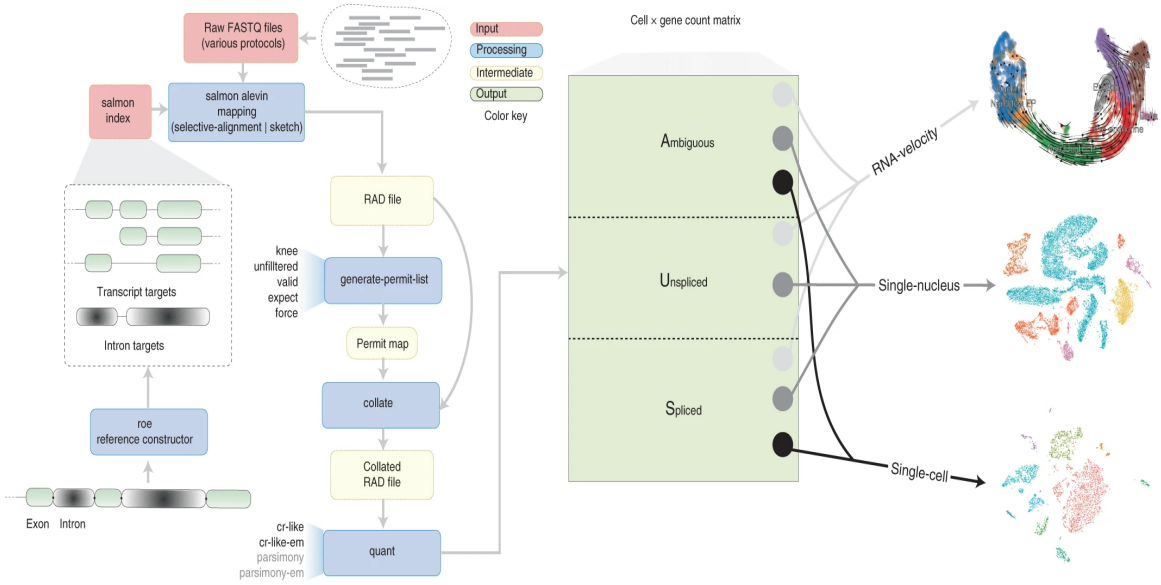


Figure 2.1: This figure describes the *alevin-fry* pipeline from start to finish. First, a *salmon* index is created which is then used to map the raw FASTQ files (obtained from sequencing) to the reference genome. From there a RAD file is created that is used for quality control (permit list and collate) and quantification. After quantification the counts can be used for various analyses e.g. RNA velocity plots (He *et al.*, 2022).

2.2 Read-level simulation with minnow

minnow is a read level simulator for droplet based scRNA-seq data that accounts for important sequence-level characteristics and model effects (Sarkar *et al.*, 2019). It matches the gene-level ambiguity characteristics that are present in real scRNA-seq experiments. With *minnow* it is possible to demonstrate the effect of gene-level sequence ambiguity on accurate quantification, which is used in this thesis to simulate mapping uncertainty between spliced and unspliced counts. It achieves this by either simulating sequences from the underlying de-Bruijn graph of the reference transcriptome or from the reference transcriptome directly (Sarkar *et al.*, 2019). The *minnow* framework essentially works in three steps: (i) selection of transcript, (ii) simulation of cell barcode (CB) and unique molecular identifier (UMI) tagging and (iii) simulation of polymerase chain reaction (PCR), fragmentation and sequencing. PCR is a laboratory technique to rapidly produce millions of copies of a specific DNA segment (Garibyan and Avashia, 2013). First, *minnow* uses a gene-count matrix as input that provides an estimated number of distinct molecules corresponding to each gene and cell in the sample. *minnow* treats the normalized values of a particular cell as a multinomial distribution, then samples such molecules from that distribution (Sarkar *et al.*, 2019). Figure 2.2 illustrates the process from input to simulated reads. *Minnow* is used in this thesis to simulate at the read-level, which are afterwards aligned and quantified with *alevin-fry*. This leads to a realistic simulation, which incorporates multi-mapping uncertainty, whose modelling is the primary objective of this thesis.

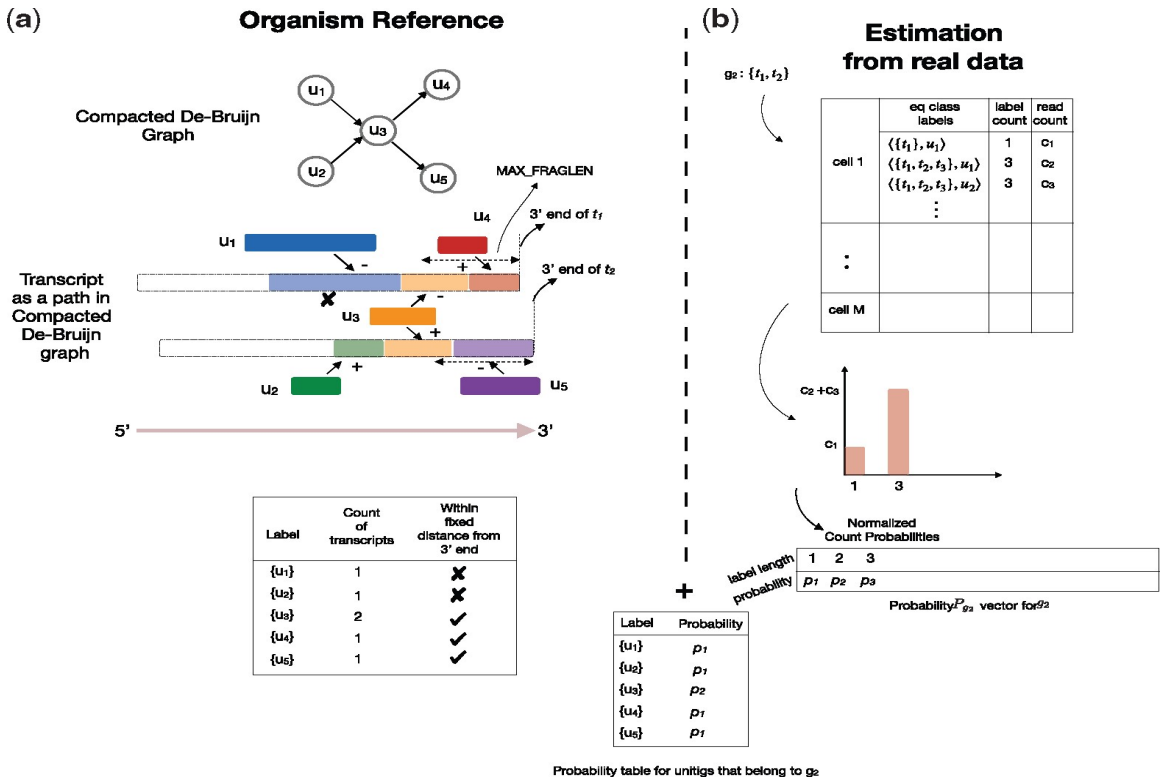


Figure 2.2: There are two possible ways *minnow* can be used to create simulated reads. First, a compacted De-Bruijn graph is used to simulate reads from a reference organism. Second, reads can be simulated from real data by sampling from a multinomial distribution (Sarkar *et al.*, 2019).

2.3 Differential methods

2.3.1 *eisaR*

Exon-intron split analysis (EISA) is a computational approach that measures mature RNA and pre-mRNA reads across different experimental conditions (Gaidatzis *et al.*, 2015). The method has been developed to quantify transcriptional and post-transcriptional regulation of gene expression. After quantification of exonic and intronic reads, both counts are normalized to the mean library size and log_2-transformed with the addition of a pseudocount of 8. Based on these expression levels, very lowly expressed genes (average log_2 expression of at least 5) in either exons and introns are removed, such that there is a fixed set of quantifiable genes. As absolute exonic and intronic counts have very different distributions it is difficult to model them in the same statistical model (Gaidatzis *et al.*, 2015). Therefore, the difference of Δexon and Δintron is modelled with a generalized linear regression model to mitigate this issue. The generalized linear regression approach uses an interaction term to determine the statistical significance between exonic and intronic counts and the experimental conditions, and is implemented within the *edgeR* framework (Robinson *et al.*, 2010) that is used to examine differential gene expression of replicated count data where the counts are modelled by a negative binomial distribution (2.1), where for the g-th gene and i-th sample:

$$Y_{gi} \sim NB(\text{mean} = M_i \rho_{gj}, \text{dispersion} = \phi_g) \tag{2.1}$$

where M_i is the library size (total counts) for i-th sample; ϕ_g is the dispersion for the g-th gene and ρ_{gj} is the relative abundance of gene g in experimental group j which sample i belongs to and

NB(a, b) denotes the negative binomial distribution with mean a and dispersion b. Ultimately, the significance of the interaction term is calculated by the use of a likelihood ratio test between the full model and a reduced model containing the experimental condition without the interaction term (Gaidatzis *et al.*, 2015). In this thesis we used the R package *eisaR* which implements the EISA framework to conveniently run the method (Stadler *et al.*, 2020).

2.3.2 *BRIE2*

BRIE2 is a scalable computational method that regresses single-cell RNA-seq data against cell-level features (Huang and Sanguinetti, 2021). Unavoidable difficulty in the quantification arises from the fundamental ambiguity of the data, because a majority of reads cannot be mapped unambiguously to a single isoform. *BRIE1* tackled this issue by regressing percentage of spliced-in (PSI) values through a Bayesian regression approach. However, *BRIE1* is not well suited to quantify differential splicing across cell types because sequence features are usually the same between individual cells (Huang and Sanguinetti, 2021). *BRIE2* again starts from a latent regression framework, however, it differs from *BRIE1* in two important ways: first, it augments the set of regressor features by including cell-specific information e.g. cell type; second, the added complexity considerably increases the computational cost. Therefore, because of its elevated computational complexity, *BRIE2* was developed to be used in conjunction with advanced software e.g. Tensorflow and graphics processing units (GPUs), which significantly increases computational acceleration. In this thesis, we focus on the DMG mode as it performs differential testing on the relative abundance of spliced and unspliced reads. Figure 2.3 summarizes the process from input to output in the *BRIE2* framework.

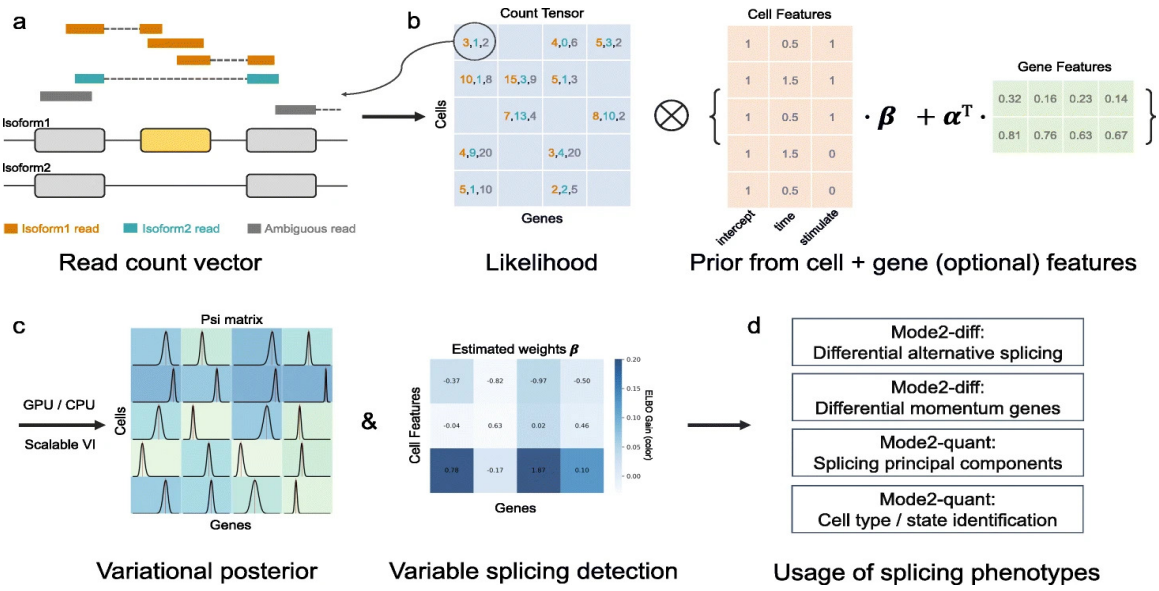


Figure 2.3: This figure describes the *BRIE2* framework from input to output. Reads are classified as isoform 1, isoform 2, or ambiguous based on their alignment identity, resulting in a cell-by-gene-by-3 tensor. The isoform percentage posterior distribution PSI is calculated by combining read-count likelihood and an informative prior predicted by cell-level variables and/or gene sequence characteristics. To approximate the precise posterior, a logit-normal variational posterior and coefficients on variables are optimized, and the evidence lower bound (ELBO) gain between adding and excluding a specific cell feature collection can be used to choose splicing phenotypes. The chosen differential splicing events or differential momentum genes on RNA velocity can be employed as markers for downstream analyses, and the calculated PSI can be used for dimension reduction to improve cell type/state identification (Huang and Sanguinetti, 2021).

2.3.3 *DEXSeq*

DEXSeq (Anders *et al.*, 2012) is a statistical method originally proposed to test for differential exon usage in RNA- seq data, which has been widely adopted in other contexts too, such as differential transcript usage (Love *et al.*, 2018). The model is based on the negative binomial distribution and allows for covariates such as batch effects to be taken into account to offer reliable control of false discoveries (Anders *et al.*, 2012). In its original implementation *DEXSeq* inputs how many reads map to each exon, but the method has also been used on transcript level counts [(Soneson *et al.*, 2016), (Tiberi and Robinson, 2020)]. Equation (2.2) shows that the read counts follow a negative binomial distribution where α is the dispersion parameter. Further, a generalized linear model is used to predict the mean via a log-linear link, where for gene i, exon l and sample j:

$$K_{ijl} \sim \text{NB}(\text{mean} = s_j \mu_{ijl}, \text{dispersion} = \alpha_{il}) \tag{2.2}$$

$$log(\mu_{ijl}) = \beta_i^G + \beta_{il}^E + \beta_{i\rho_j}^C + \beta_{i\rho_j l}^{EC} \tag{2.3}$$

where NB(a, b) denotes the negative binomial distribution with mean a and dispersion b, α_{il} is the dispersion parameter; s_j is the size factor for the j-th sample; μ_{ijl} is the expected value of the concentration of cDNA fragments of the l-th exon of i-th gene in sample j; ρ_j is the group sample j belongs to; β_i^G is the baseline expression strength of gene i; β_{il}^E is the coefficient for the

l-th exon in gene i; $\beta^{C}_{i\rho_j}$ is the coefficient for the group ρ_j in gene i; $\beta^{EC}_{i\rho_j l}$ is the exon-condition interaction term for condition ρ_j and exon l in gene i.

The dispersion parameter allows over-dispersed data (i.e. higher variance than mean) to be modeled. Here, we propose to use *DEXSeq* on estimated USA counts, and perform a differential usage test between experimental conditions. Therefore, for every gene, exons are replaced by the spliced, unspliced and ambiguous versions of the gene. This models ambiguous reads separately from spliced and unspliced or exonic and intronic, thus eliminating one of the main sources of mapping uncertainty. However, the uncertainty related to reads mapping to multiple genes is still neglected by this approach. To address both sources of mapping uncertainty we propose a novel method, developed by Simone Tiberi (Tiberi, 2022).

2.3.4 *DifferentialRegulation*

DifferentialRegulation is a recent statistical method for discovering differentially regulated genes between two groups of samples. The method targets differences in the relative abundance of spliced, unspliced and ambiguous counts, obtained from scRNA-seq data. *DifferentialRegulation* accounts for the sample-to-sample variability, and embeds multiple samples in a Bayesian hierarchical model, via a Dirichlet-Multinomial distribution that models, for each gene, the spliced, unspliced and ambiguous reads relative abundance. While all the other 3 methods considered work with estimated counts, *DifferentialRegulation* inputs equivalence classes instead, which indicate what gene(s) each read is compatible with. The model also accounts for both major sources of mapping uncertainty: i) A reads mapping to both S and U vesions of a gene, and ii) reads compatible with multiple genes. For the first source of uncertainty, similarly to *DEXSeq*'s approach outlined above, ambiguous reads are treated separately from spliced and unspliced reads. Regarding multi-gene mapping reads, the method uses a latent variable approach and treats the gene allocation of each read as an unknown parameter that, via a data augmentation approach, is also sampled.

DifferentialRegulation is built around two nested models: i) for each sample, a multinomial model ($\mathcal{MN}$) is used for the gene relative abundance, and ii), for each gene, a Dirichlet-multinomial ($\mathcal{DIR} - \mathcal{MN}$) model is employed for the relative abundance of S, U, and A reads within the gene:

$$Y_i \sim \mathcal{MN}\left(\rho_i^{(1)}, \dots, \rho_i^{(N_g)}\right) \qquad \text{for } i = 1, \dots, N \tag{2.4}$$

$$X_i^{(g)} \sim \mathcal{DIR} - \mathcal{MN}\left(\pi_S^{(g)}, \pi_U^{(g)}, \pi_A^{(g)}, \phi^{(g)}\right) \qquad \text{for } g = 1, \dots, N_g \text{ and } i = 1, \dots, N, \tag{2.5}$$

where $X_i^{(g)} = \left(X_{iS}^{(g)}, X_{iU}^{(g)}, X_{iS}^{(g)}\right)$ is the vector indicating the overall abundance of the spliced, spliced and ambiguous reads within gene g in sample i; $Y_i = \left(Y_i^{(1)}, \dots, Y_i^{(N_g)}\right)$, with $Y_i^{(g)} = X_{iS}^{(1)} + X_{iU}^{(1)} + X_{iA}^{(1)}$ indicating the overall abundance of gene g in sample i; parameter $\rho_i^{(g)}$ indicates the relative abundance for the g-th gene in the i-th sample; $\pi_S^{(g)}, \pi_U^{(g)}, \pi_A^{(g)}$ represent the relative abundance of spliced, unspliced and ambiguous reads, respectively, for the g-th gene in the i-th sample; and $\phi^{(g)}$ is the Dirichlet-multinomial precision parameter that models how the relative abundances vary across samples.

For inferring the parameters ϕ, *DifferentialRegulation* uses an empirical Bayes approach, where the mean, μ_ϕ, and standard deviation, σ_ϕ of $log(\phi)$ are estimated from the data. To accelerate the calculations, and with marginal loss of accuracy, in each cell type, a randomly selected sub-set of 1,000 randomly selected genes is used to estimate μ_ϕ and σ_ϕ. These estimates are then used to formulate a, cell-type specific, informative prior for $log(\phi)$, as: $log(\phi^{(g)}) \sim N\left(\mu_\phi, \sigma_\phi^2\right)$, for $g = 1, \dots, N_g$.

The Dirichlet-multinomial parameters are the key parameters of the method, and are used to perform differential testing between conditions. Instead, the multinomial parameters are only required, in conjunction with the Dirichlet-multinomial parameters, for the latent variable allocation of reads mapping to multiple genes. As an example, consider a read that is compatible with i) the S version of gene g and ii) the U version of gene j; this read is allocated to:

i) with probability $\propto \pi^{(g)}_{gene} * \pi^{(g)}_{S}$;

ii) with probability $\propto \pi^{(j)}_{gene} * \pi^{(j)}_{U}$.

A latent variable approach to allocate ambiguous reads to the spliced or unspliced version of a gene was also considered. However, such a framework would require a good estimate of the probability that an ambiguous read is spliced or unspliced; however, this probability depends on many factors, which are either unknown or hard to be modeled, and a good estimate cannot be accurately formulated. For this reason, similarly to *DEXSeq*'s approach described before, ambiguous reads were eventually kept separately.

Model parameters are inferred via a Metropolis-within-Gibbs Markov chain Monte Carlo (MCMC) technique (Andrieu *et al.*, 2003), where parameters and latent states are alternatively sampled from their conditional distributions. After inferring model parameters, for each gene, differential testing is then performed by comparing, via a multivariate Wald test, the posterior distributions of the Dirichlet-Multinomial group-level relative abundances for the spliced, unspliced and ambiguous reads. In particular, given groups two groups A and B, and generalizing the parameters in (2.5) with a group pre-subscript, the following system of hypotheses is tested:

$$H_0 \quad : \quad \left({}_A\pi^{(g)}_S, {}_A\pi^{(g)}_U, {}_A\pi^{(g)}_A\right) = \left({}_B\pi^{(g)}_S, {}_B\pi^{(g)}_U, {}_B\pi^{(g)}_A\right) \tag{2.6}$$

$$H_1 \quad : \quad otherwise. \tag{2.7}$$

2.4 Analysis of results

2.4.1 Unifold Manifold Approximation and Projection (UMAP)

Unifold Manifold Approximation and Projection (UMAP) is a dimension reduction technique that can be used to visualize data from a high-dimensional space into a low-dimensional space (McInnes *et al.*, 1802). UMAP is implemented on the following three assumptions about the data: first, the data is uniformly distributed on the Riemannian manifold (Lee, 2018); second, the Riemannian metric is locally constant (Lee, 2018); third, the manifold is locally connected. Essentially, UMAP constructs a high dimensional graph representation of the data and then tries to fit a low-dimensional graph that is as structurally similar as possible (McInnes *et al.*, 1802). In this thesis UMAP is used to assess how cells cluster i.e. based on their cell type or to which sample they belong. This knowledge is important, for example to evaluate whether there are structural differences between samples, although the samples are biological replicates. Figure 2.4 shows the clustered digits of the famous MNIST data set which consists of 28x28 pixel grayscale images of handwritten digits (0 through 9) (Deng, 2012). Each digit is described by a 784 dimensional vector which hereby was reduced to a two-dimensional representation by applying UMAP.

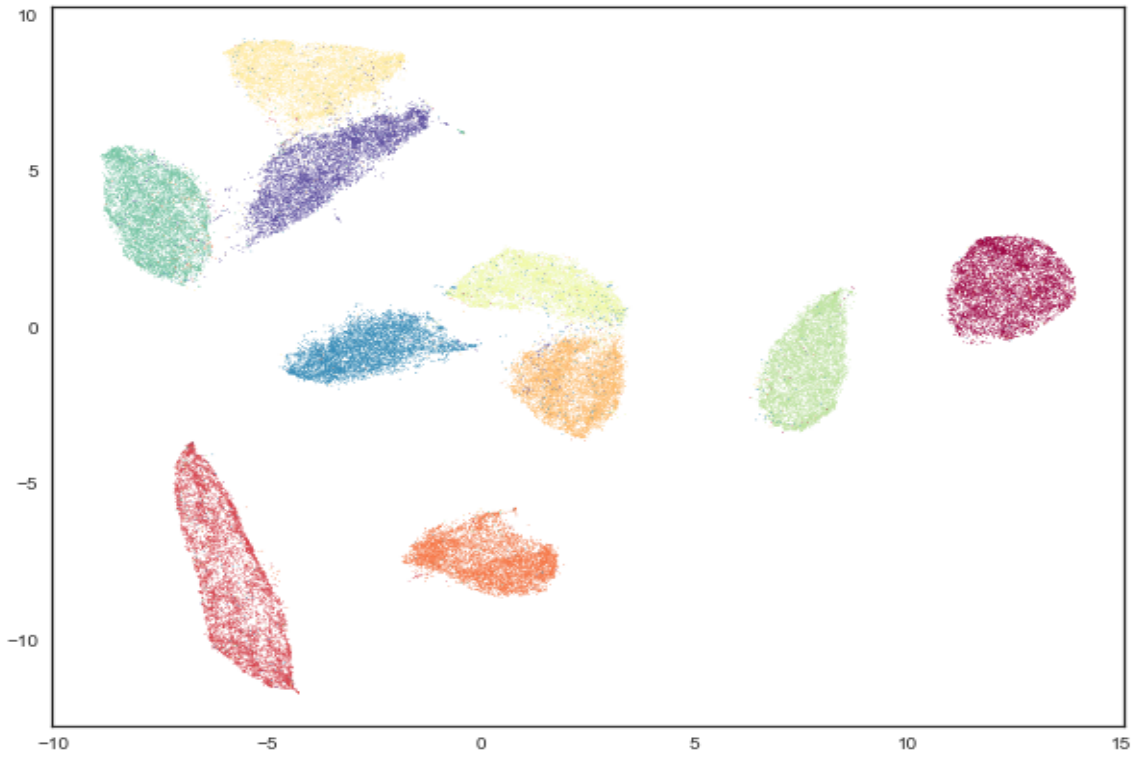


Figure 2.4: Example of a UMAP representation of high dimensional data into two dimensions highlighting the data clusters (Sainburg *et al.*, 2021)

2.4.2 Classification measurements

The true positive rate (TPR), false positive rate (FPR), and false discovery rate (FDR) are defined as:

$$\text{TPR} = \frac{|\text{TP}|}{|\text{TP} + \text{FN}|} \tag{2.8}$$

$$\text{FPR} = \frac{|\text{FP}|}{|\text{FP} + \text{TN}|} \tag{2.9}$$

$$\text{FDR} = \frac{|\text{FP}|}{|\text{TP} + \text{FP}|} \tag{2.10}$$

where TP, TN, FP, and FN indicate the sets of true positive, true negative, false positive, and false negative elements, respectively. TP are the number of elements that were correctly identified as the positive outcome. Similarly, TN are the number of elements that were correctly identified as the negative outcome. FP are those elements that were identified as the positive outcome, however they should have been identified as negative. In statistics, FP is usually referred to as Type 1 error. Similarly, FN are the elements that were incorrectly identified as negative - also referred to as the Type 2 error. Figure 2.5 illustrates the meaning of these measurements quite clearly. The TPR (true positive rate) also referred to as sensitivity, measures the proportion of positive elements that were correctly identified - often alluded to as statistical power (2.8). The FPR (false positive rate) measures the proportion of negative outcomes that are identified as positive (2.9). On the other hand, FDR (false discovery rate) measures the expected proportion of false positive among all positive predictions (2.10).

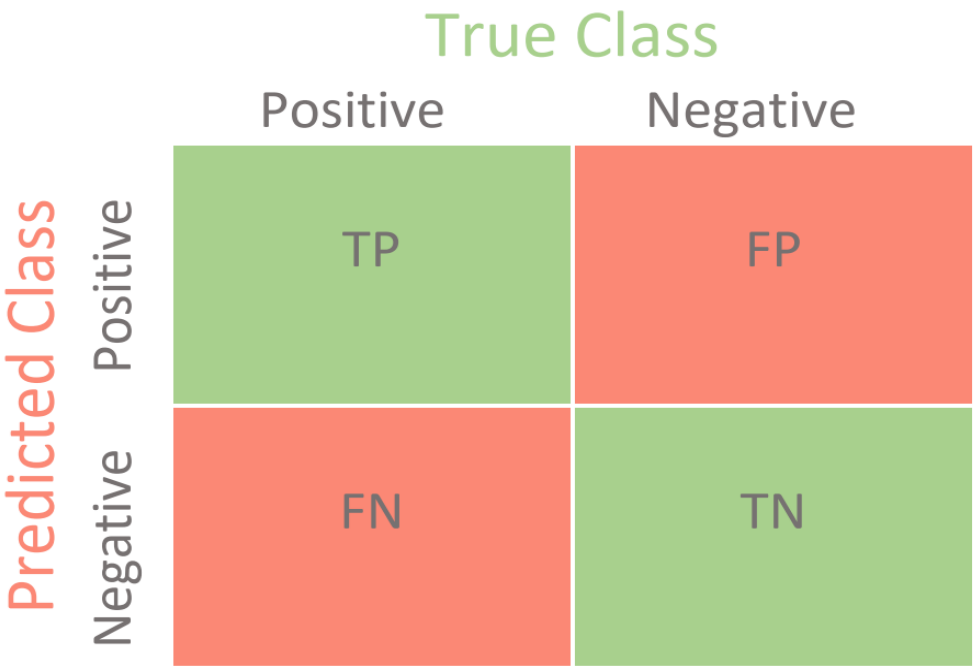


Figure 2.5: Confusion matrix reporting the performance of a binary classification problem (Mohajon, 2020)

2.4.3 ROC curve

The ROC curve is a performance measurement that is used for classification problems - in this case whether a gene is differently regulated or not. Essentially, the ROC curve plots the TPR (y-axis) against the FPR (x-axis). ROC curves above the diagonal indicate better performance than blind random guessing, denoted by the diagonal line. ROC curves are one of the performance evaluation methods used in this thesis.

2.4.4 TPR v. FDR curves

Similar to the ROC curve the TPR v. FDR curve is used to assess performance in classification problems. It plots the TPR (y-axis) against the FDR (x-axis). In addition, one often plots the achieved FDR that was observed for various theoretical thresholds - usually 1, 5 and 10%. FDR values are essentially adjusted p-values which can be calculated in various ways from raw p-values. However, testing on many genes leads to an increased number of significant results. For example, when testing 10'000 genes we expect at least 500 significant results based on the 5% significance level even under H_0, therefore, we adjust the raw p-values with a correction method. Arguably, the most popular one is the Benjamini-Hochberg correction (Benjamini and Hochberg, 1995). All the methods considered in this thesis, provide FDR adjusted p-values obtained via Benjamini-Hochberg correction.

When the observed FDR is lower than or equal to the specified threshold, the method controls for the FDR. However, if the observed FDR is greater than the threshold the method does not control for the FDR and there is an inflation of false positive predictions. In this thesis, the methods are evaluated by an adjusted p-value.

Chapter 3

Results

3.1 Exploratory Data Analysis

3.1.1 Mouse kidney cells

The first data set stems from a paper that investigates potential cellular targets of kidney disease in mice (Park *et al.*, 2018). The authors isolated and sequenced a total of 57'979 cells from whole kidney cell suspensions (one kidney per mouse) derived from seven healthy male mice using droplet-based single-cell RNA sequencing. The samples were labelled as: normal1, normal2, normal3, normal4, Ksp-cre-GFP, Scl-cre-GFP and Pod-cre-GFP. For our work, we decided to use the raw data from the four samples that were labelled as normal to have multiple biological replicates from the same experimental condition.

We used the *alevin-fry* pipeline to quantify the raw single-cell RNA sequencing data for further use in the R programming environment. Quality control is a crucial stage in data pre-processing as low-quality libraries can contribute to misleading results in downstream analyses (Amezquita *et al.*, 2020). Therefore, we filtered lowly abundant genes and low-quality cells to mitigate said problems to improve interpretability of the results.

To identify low-quality cells, cell-specific QC metrics were calculated with the *perCellQC-Metrics* function from the *scater* R package (McCarthy *et al.*, 2017). These metrics include the total number of expressed genes, the overall count across all genes, and the fraction of counts assigned to control genes such as mitochondrial genes. By setting a specific threshold on per-cell QC metrics, high-quality cells can be retained. In our setting, outliers are defined as cells with library sizes more than two median absolute deviations away from the median library size. Figure 3.1 summarizes the process from unprocessed to processed single-cell experiment.

After filtering, the data set consists of 23'543 cells and 18'537 genes. Next, we used the *SingleR* function from the *SingleR* R package (Aran *et al.*, 2019) for cell-type annotation. Cell-type annotation is important to determine what biological state is represented by cell clusters which helps the interpretability of the results and their implications (Amezquita *et al.*, 2020). *SingleR* is a method that assigns labels based on the reference samples with the highest Spearman rank correlation while only using marker genes between pairs of labels to focus on the relevant differences between cell types (Aran *et al.*, 2019). Figure 3.2 shows the UMAP of the cells coloured by their respective sample id. From Figure 3.2 one can observe that the projection of the cells is very similar across the samples. Further, Figure 3.3 also illustrates that cells from the same cell-type cluster together, as one would expect.

Figure 3.4 shows that the annotated cells were largely classified as either: Adipocytes, Epithelial cells and Hepatocytes. Additionally, one can observe that those three cell types are approximately evenly distributed across the four samples. For further analyses we focus on those cell types to investigate the performance of existing methods for the detection of differentially expressed genes and design our own simulation study.

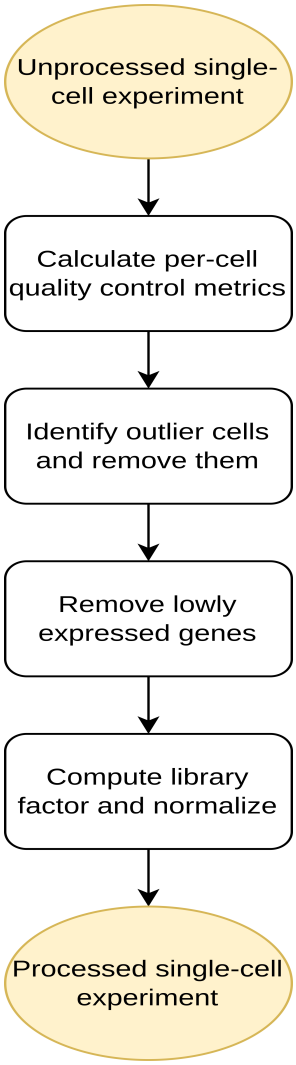


Figure 3.1: Quality control process from unprocessed, raw to processed, filtered single-cell experiment

After QC filtering and cell type annotation, we investigated the RNA velocities for each sample. Initially, we expected similar patterns with changing trajectories. First, we explored the proportions of spliced and unspliced counts in each sample. Figure 3.5 shows that the abundance of spliced counts is very high in comparison to unspliced counts. In samples 3 and 4 the abundance of unspliced counts approximately half, compared to samples 1 and 2. After exploring the velocity plots, there were no clear patterns that were consistent between the biological replicates as shown in Figure 3.6. We initially thought that we could relate differentially regulated genes to differential velocity between experimental conditions. However, this does not seem possible because RNA velocity images are hardly comparable across samples, which makes it hard to visually identify differences across experimental conditions. We therefore concluded that our discoveries (i.e. differences in the relative abundance of US or USA reads), cannot be taken as a proxy for "differential velocity". Although, the ideas of differential regulation and differential velocity are connected (i.e. RNA velocities are calculated on US estimated reads), we decided to keep the two concepts separate, and interpret our discoveries as differential regulated genes only.

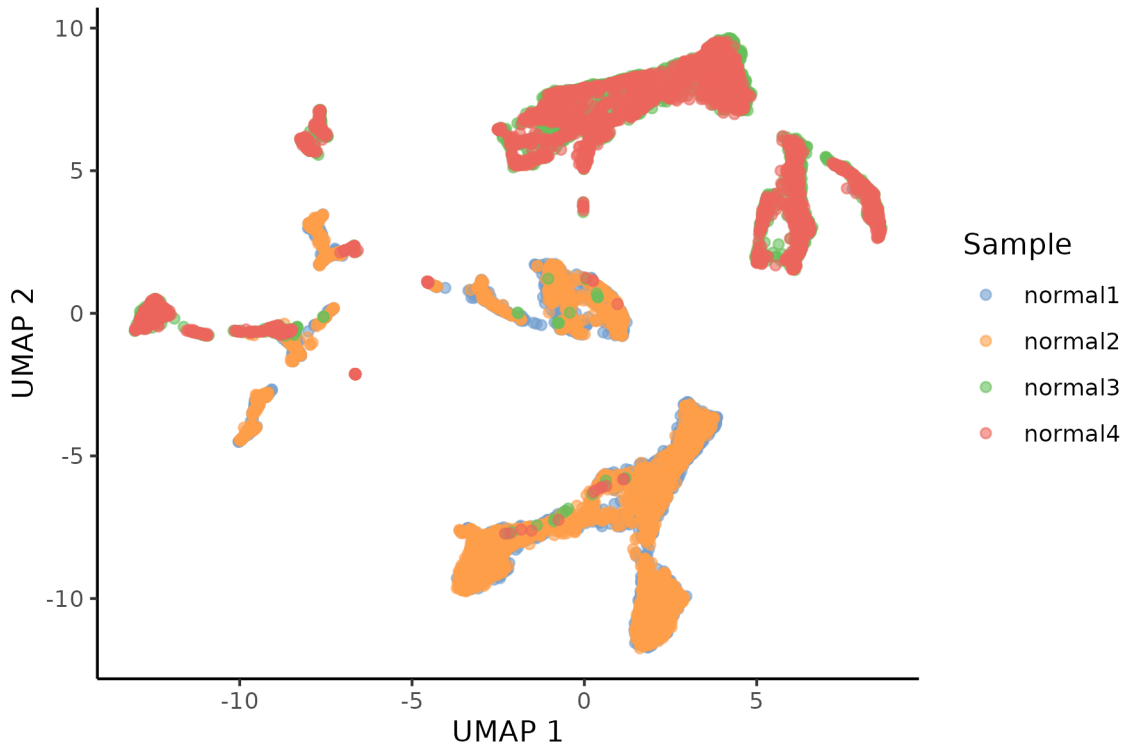


Figure 3.2: UMAP representation of the mouse kidney cells coloured by sample id

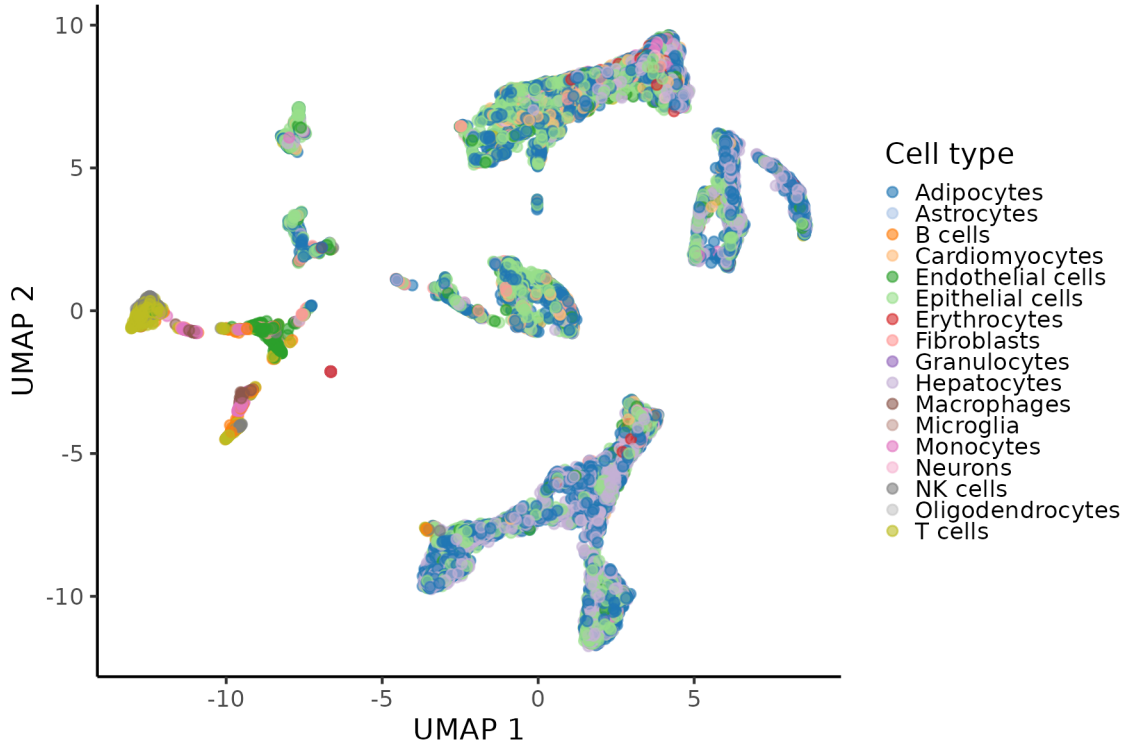


Figure 3.3: UMAP representation of the mouse kidney cells coloured by cell type

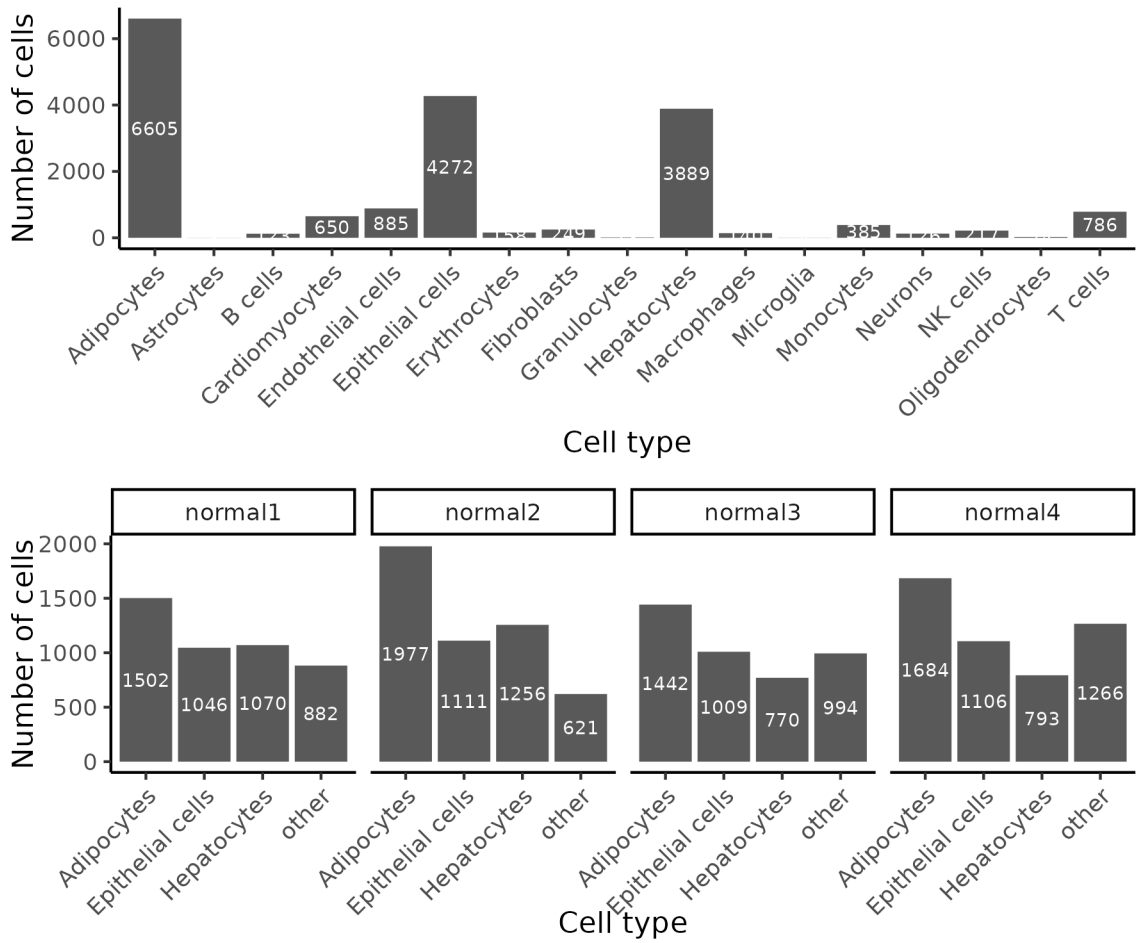
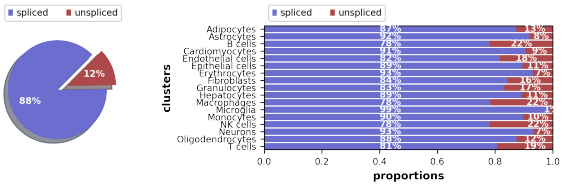
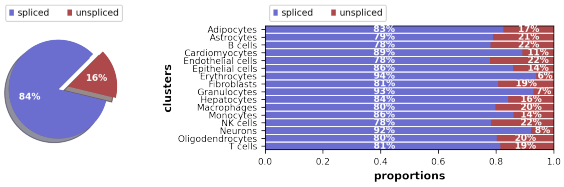


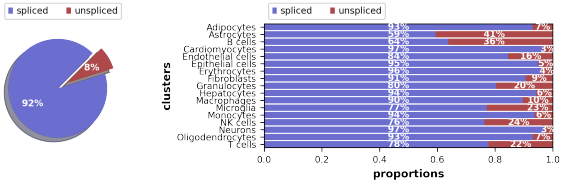
Figure 3.4: Frequency distribution of the cell types (annotated with *SingleR*) after quality control



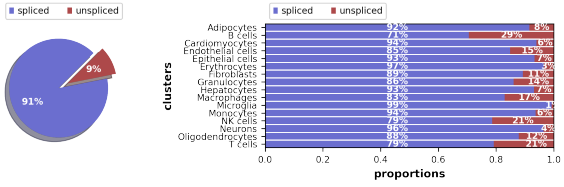
(a) normal1



(b) normal2

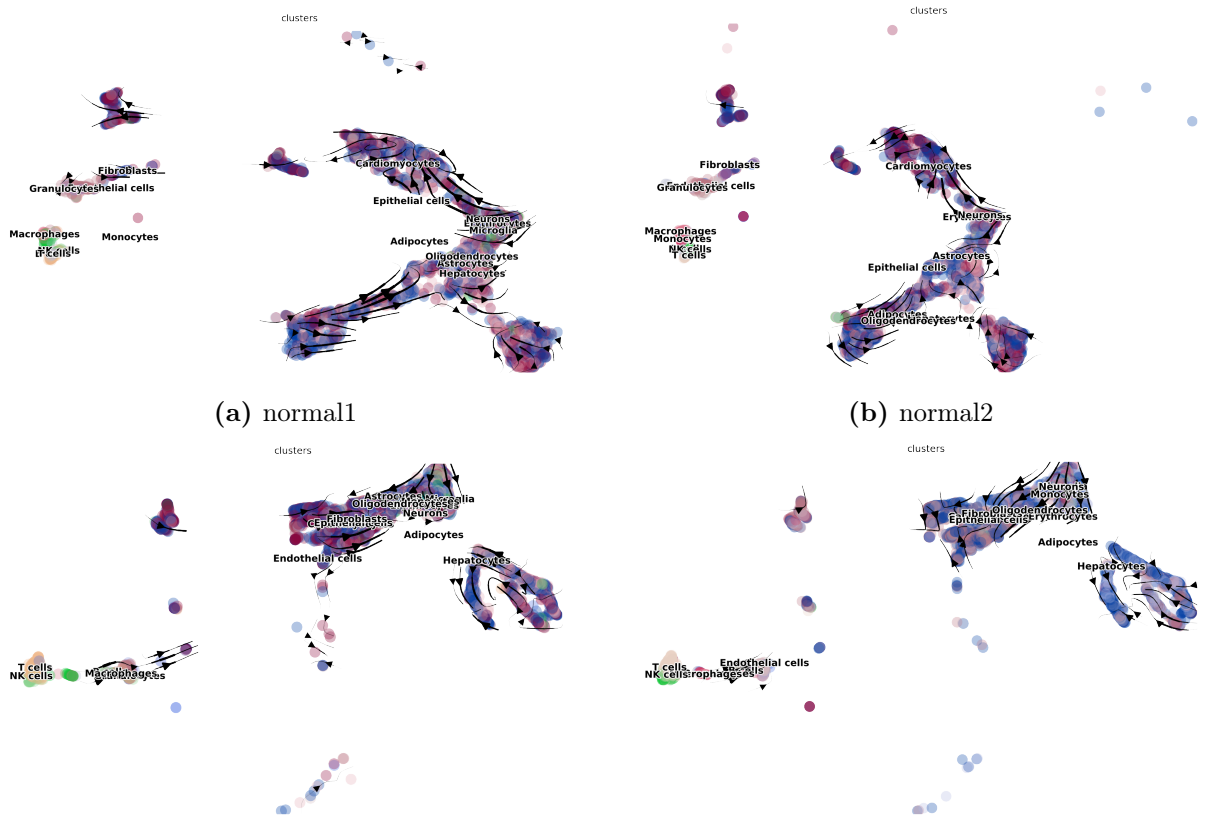


(c) normal3



(d) normal4

Figure 3.5: Relative abundance of spliced and unspliced counts across different cell types for all four kidney mouse samples



(a) normal1 **(b)** normal2

(c) normal3 **(d)** normal4

Figure 3.6: RNA velocity plots for all four four kidney mouse samples

3.2 Simulation study

3.2.1 Simulation strategy

We designed two simulations: i) one where we simulated differentially regulated (DR) genes only and ii) one where we simulated both DR and differentially gene expression (DGE). In the second simulation, DGE was added as a nuisance effect that makes DR detection more challenging. Below, we first describe how to simulate DR effects, and then illustrate how DGE was added in the second simulation. Initially, the simulation strategy was to invert the spliced and unspliced counts for 10% of genes for all cells that belong to an arbitrary group A. The group allocation was made based up on the consideration of the UMAP from Figure 3.2 where it is visible that samples tend to cluster in pairs: 1 with 2, and 3 with 4. Therefore, the group allocation of $Group_A = (sample_1, sample_3)$ and $Group_B = (sample_2, sample_4)$ was chosen to obtain a homogeneous representation of the groups. The set of genes, whose counts were to be inverted, was randomly drawn by a sampling algorithm without replacement (hypergeometric distribution). There are many different ways to introduce a differential effect, however inverting the spliced and unspliced counts to be an effective way to achieve this without actually modifying the originally estimated counts. This procedure was done separately for each cell type, so that differential genes are not the same across cell types. Additionally, in the second simulation only, DGE was introduced in 10% of genes in all cells that belong to said arbitrary group A. In order to introduce DGE, we additionally multiply the counts by 10 (ten-fold gene expression) for 10% randomly drawn genes in group A. Again, the set of genes was randomly drawn by the same sampling algorithm as before.

Starting from a real data set as an anchor data set, then artificially introducing a differential

effect, we essentially created two semi-simulated data sets. Compared to full simulations, semi-simulated approaches have the advantage of having a realistic structure, because it was indeed taken from real data. The genes and cells that were subject to change were stored as ground truth for further evaluation in the downstream analyses. Figure 3.7 illustrates the simulation process from the original mouse kidney data set to two simulated data sets.

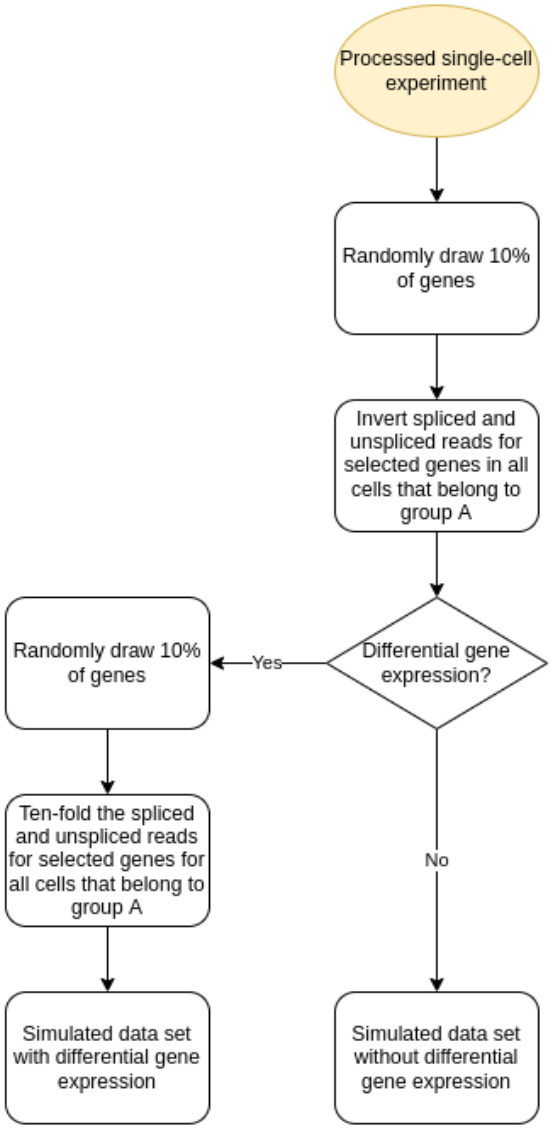


Figure 3.7: Simulation process from original mouse data set to simulated data sets

The goal of this thesis was to determine how well the aforementioned methods perform on detecting differentially regulated genes on read-level simulated data sets and the effect of multi-mapping uncertainty. To achieve this, two groups of methods were postulated as *eisaR* and *BRIE2* cannot take into account ambiguous reads. For this reason the classification performance of *eisaR* and *BRIE2* are compared with each other with the help of TPR v. FDR curves. As *alevin-fry* allows the estimation of ambiguous counts as well as spliced and unspliced, it was possible to assign the ambiguous counts to both spliced and unspliced counts (50-50 split) so that *eisaR* and *BRIE2* can be applied. We settled on assigning 50% of ambiguous count to spliced and the other 50% to unspliced because other methods, such as *alevin*, also use this approach. In a similar manner, *DEXSeq* and our own method *DifferentialRegulation* were used

to detect differentially regulated genes. However, in this case the ambiguous counts were used as an additional information. As a next step, *minnow* was used to introduce mapping uncertainty into the simulated data sets. The simulated matrix of US counts was provided to *minnow* to simulate scRNA-seq data at the read-level, which was then aligned and quantified with *alevin-fry*. In order to assess the impact of multi-mapping uncertainty, we fit *eisaR* and *BRIE2* (the only methods that require US counts), to the original US simulated matrix (i.e. the input of *minnow*), and to the US counts estimated from *alevin-fry* (after running *minnow*). The two analyses are shown in the next two sections, 3.2.2 and 3.2.3.

3.2.2 Simulation without mapping uncertainty

As a first step, we looked at the results from the naive simulation for both data sets with and without DGE. Figure 3.8 shows that both *eisaR* and *BRIE2* perform quite well in detecting differential genes. *eisaR* has a slightly higher TPR as shown in the TPR-FDR curve. From Figure 3.8, it is also shown that *eisaR* is well calibrated for FDR, whereas *BRIE2* is not.

After introducing DGE in the second data set, performance drops quite substantially as shown in Figure 3.8. However, *eisaR* is still well calibrated for FDR. Similar to the data set without DGE, *BRIE2* is not well calibrated for FDR. It seems that both methods are heavily affected by the introduction of DGE as performance dropped to almost half. However, we introduced a very strong effect - 10 fold change - therefore, it is to be expected that performance drops. Nevertheless, we wanted to investigate how the performance changes in an extreme scenario, hence the strong effect size.

From this naive simulation, we concluded that *BRIE2* has inflated FDR in both cases - with and without DGE. On the other hand, *eisaR* was well calibrated for FDR in both cases. However, TPR decreases by almost half after introducing DGE. In the next step, we use *minnow* to introduce multi-mapping uncertainty in to both data sets to make the simulation more realistic. It is to be expected that performance will further decrease after the introduction of multi-mapping uncertainty.

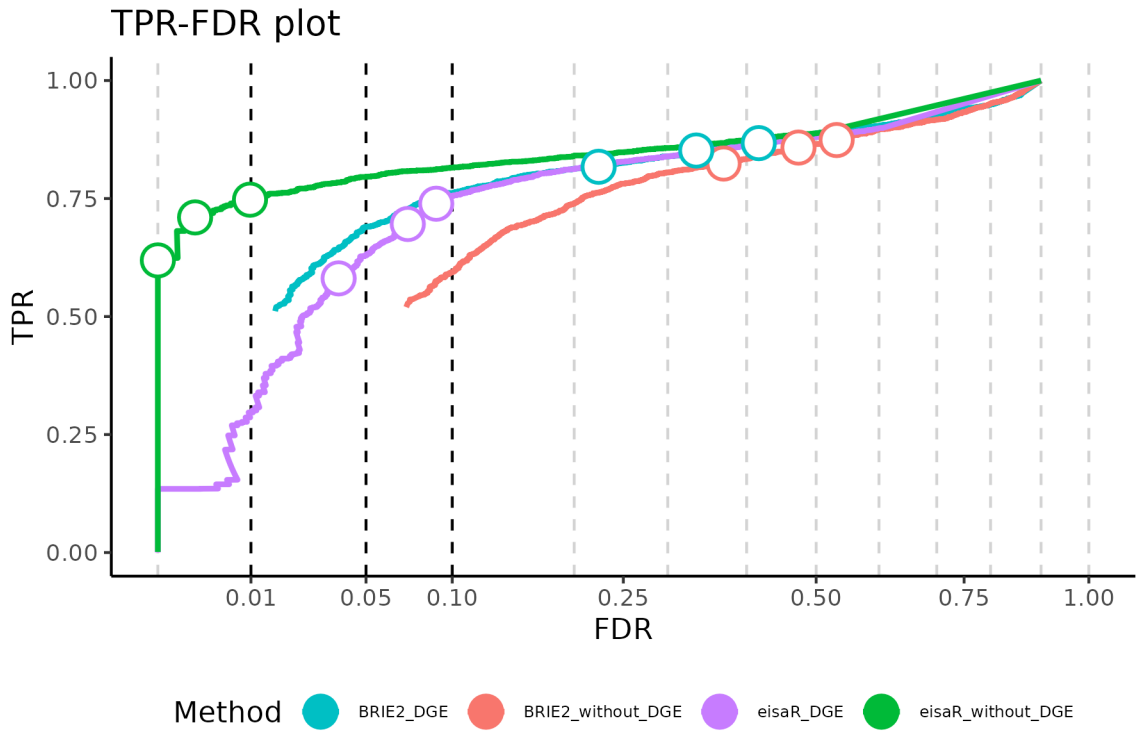


Figure 3.8: TPR *vs.* FDR plot of the naive simulation, without mapping uncertainty, comparing *BRIE2* and *eisaR* with and without DGE.

3.2.3 Simulation with mapping uncertainty

As a next step, multi-mapping uncertainty was introduced in to the two simulated data sets. First, *minnow* was used to simulate new reads from the previously simulated data sets. Second, we used *alevin-fry* to align the generated read files to the reference genome and for quantification of the reads. The newly generated data sets were then run on the aforementioned methods to identify differential genes.

Figure 3.9 shows the performance of all four methods without and without DGE. For visualization purposes we decided to omit results where spliced and unspliced counts are very similar because it is very hard to detect a differential effect. Therefore, only results are plotted where there is a minimum difference of 0.2 between spliced and unspliced counts. From the TPR v. FDR curve, we observe that *DEXSeq* and *DifferentialRegulation* have a similar TPR, although *DEXSeq* perform slightly better. Both *BRIE2* and *eisaR* have a lower TPR than the other two methods. From Figure 3.9 we observe that again *DEXSeq* and *DifferentialRegulation* have a similar performance profile. However, *DifferentialRegulation* controls better for FDR than *DEXSeq*. In contrast to before, *eisaR* controls quite well for FDR, however has half the TPR compared *DEXSeq* and *DifferentialRegulation*. On the other hand, *BRIE2* does not control well for FDR as there is strong inflation.

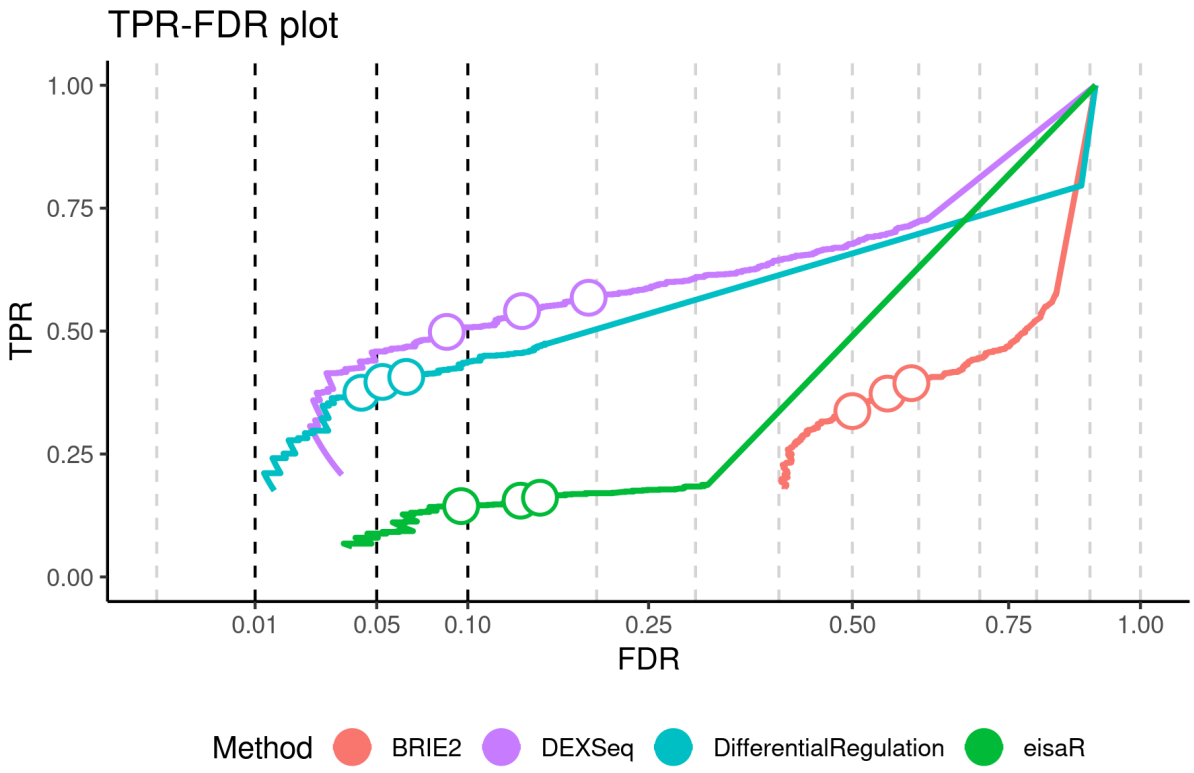
In the next step, we investigated how the performance changes after introduction of DGE into the data set. Figure 3.9 also shows the TPR v. FDR curve for all four methods. From the Figure we observe that the difference in TPR between *DEXSeq* and *DifferentialRegulation* is smaller than before. On the other hand, the performance based on TPR is approximately the same as before for *BRIE2* and *eisaR*. From Figure 3.9 it is shown that *DifferentialRegulation* still controls well for FDR, whereas the other three methods do not. The introduction of DGE as a nuisance parameter seems to affect the performance of *DEXSeq* and *eisaR* quite substantially in

terms of FDR calibration as FDR is almost doubled to before. The FDR calibration for *BRIE2* was not affected by DGE, although it was not well calibrated to begin with.

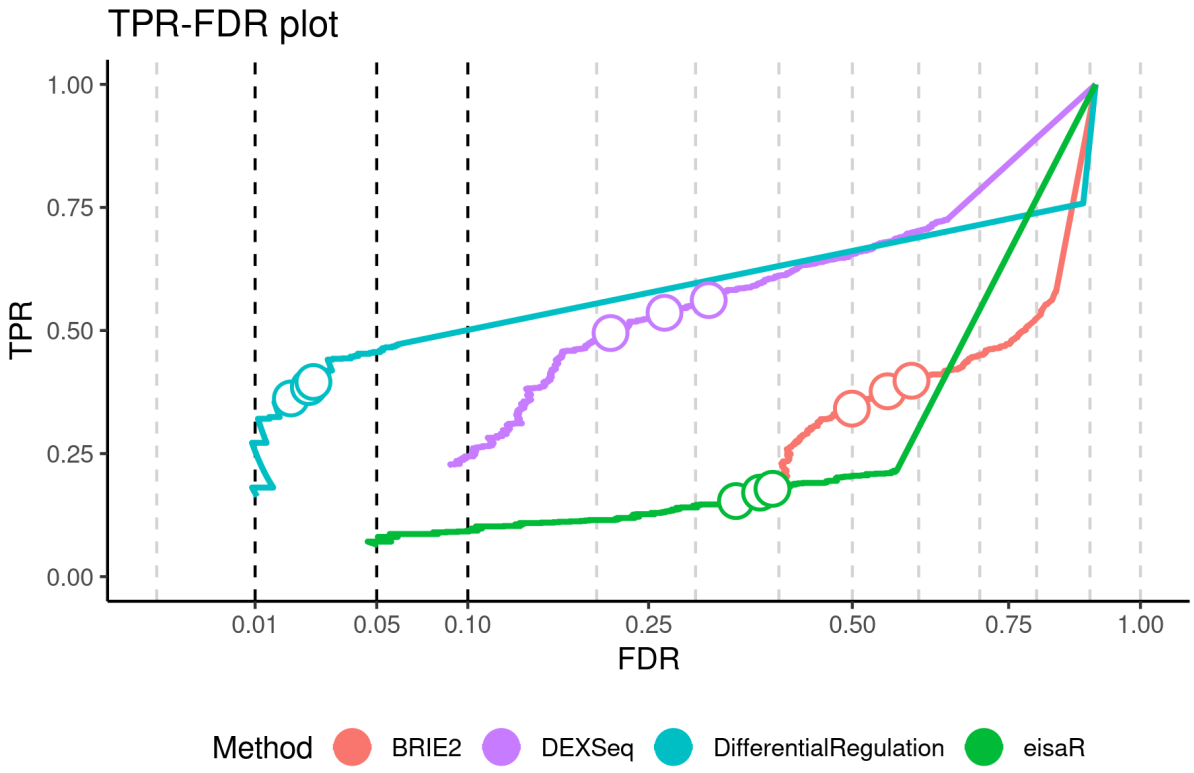
Furthermore, in order to assess how overall gene abundance affects results, we stratified the previous results by gene expression and evaluated performance, separately, for lowly, medium and highly abundant genes. This is achieved by separating results according to overall gene abundance into 3 groups, based on the quantiles of levels 1/3 and 2/3; in particular, lowly abundant genes have abundance below the first quantile (of level 1/3), moderately expressed genes have expression between the two quantiles, while highly abundant genes have expression above the second quantile (of level 2/3).

Figures 3.10 and 3.11 show the results stratified by the level of gene abundance. The Figures show that performance is consistent across gene abundance levels for *DifferentialRegulation*. TPR increases only marginally, which is expected as more data usually implies higher statistical power and FDR is also stable across gene abundance levels. Overall, *DifferentialRegulation* is not substantially impacted by gene abundance. For *DEXSeq*, TPR and FDR are both increasing as gene abundance rises. This effect is even larger for the simulation with DGE. For *BRIE2* there is a significant increase of TPR from low to highly abundant genes, however, the FDR is badly calibrated. With *eisaR* there is a marginal increase of TPR, whereas FDR is only slightly inflated without DGE. On the other hand, FDR is strongly inflated with DGE as gene abundance rises.

In this part, we investigated how robust the methods are to gene abundance with and without DGE. We concluded that the performance of *BRIE2* is low in all cases. *eisaR* has a low TPR but FDR is generally well calibrated without DGE. *DEXSeq* has good TPR and mild inflation of FDR without DGE, however, there is a strong increase in FDR with DGE. Generally, both TPR and FDR are increasing as gene abundance level rises. On the other hand, *DifferentialRegulation* is the only method with good TPR and well calibrated FDR with and without DGE across all levels of gene abundance.

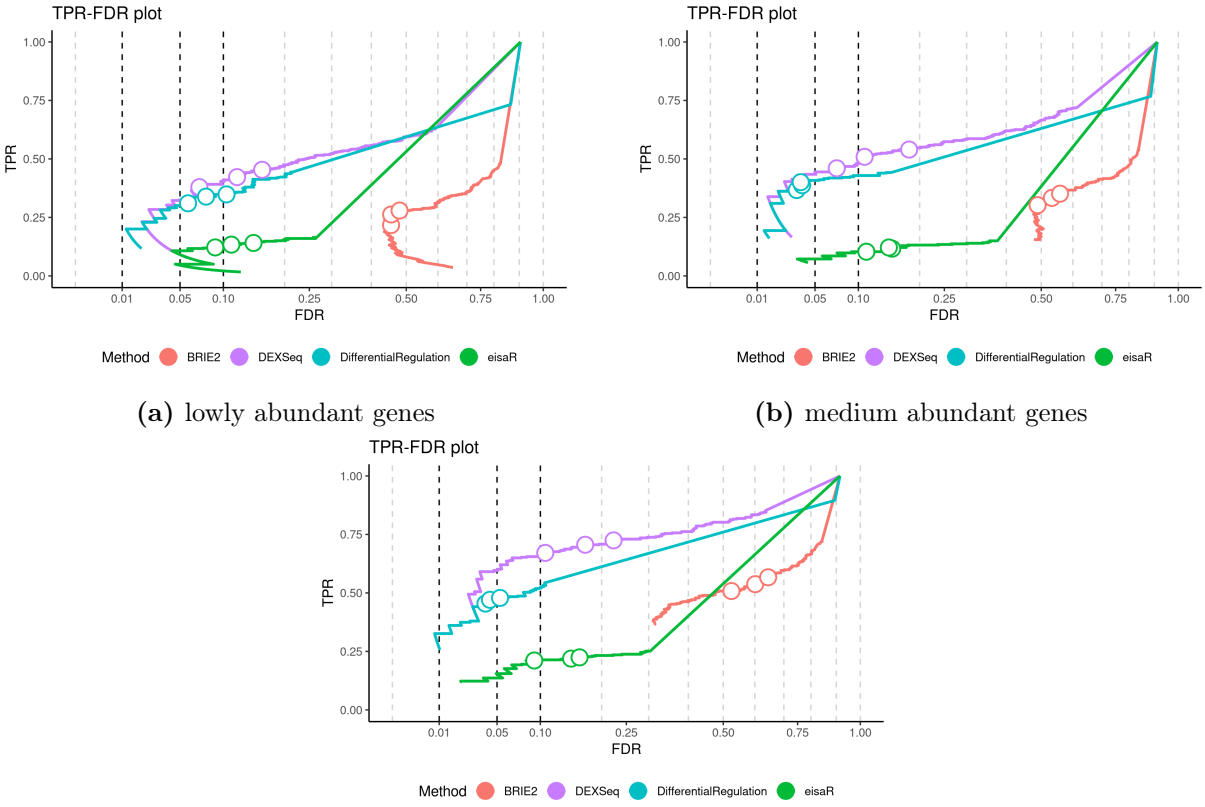


(a) without DGE



(b) with DGE

Figure 3.9: TPR *vs.* FDR plot comparing all the differential analysis methods for the simulation with mapping uncertainty (generated via *minnow*), with (bottom) and without (top) differential gene expression

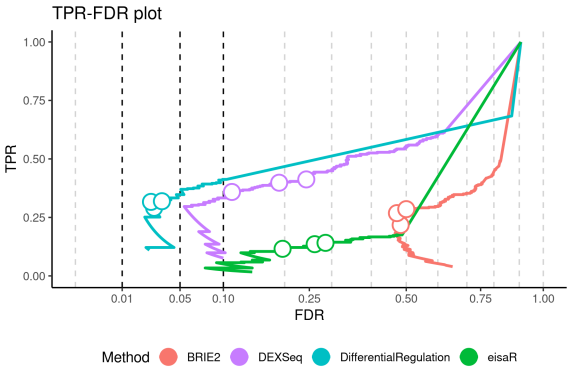


(a) lowly abundant genes

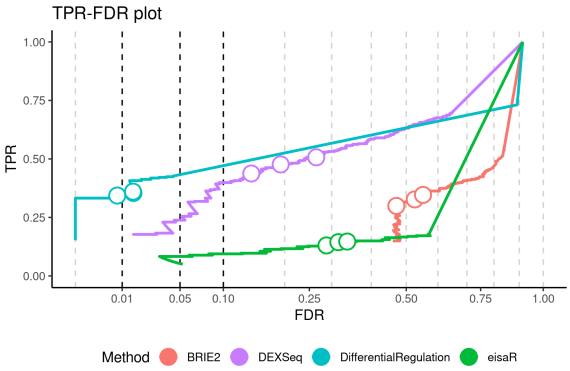
(b) medium abundant genes

(c) highly abundant genes

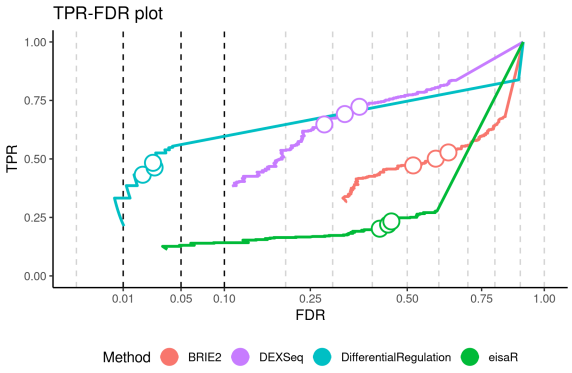
Figure 3.10: TPR *vs.* FDR plot comparing all the differential analysis methods for the simulation with mapping uncertainty (generated via *minnow*) without differential gene expression, stratified in lowly (top), medium (middle) and highly (bottom) abundant genes.



(a) lowly abundant genes



(b) medium abundant genes



(c) highly abundant genes

Figure 3.11: TPR *vs.* FDR plot comparing all the differential analysis methods for the simulation with mapping uncertainty (generated via *minnow*) with differential gene expression, stratified in lowly (top), medium (middle) and highly (bottom) abundant genes

3.3 Null analysis on the mouse kidney data

As a last step, a null analysis was conducted on the original mouse kidney data set to evaluate the methods on a real world data set. For that, all three possible group allocations were considered; namely, samples 1, 2 *vs.* 3, 4, samples 1, 3 *vs.* 2, 4, and samples 1, 4 *vs.* 2, 3. Under the null hypothesis, H_0, the p-values should be approximately uniformly distributed between zero and one, because all samples belong to the same experiment condition (i.e. normal). In particular, we were interested in checking for false positive detections that are indicated by inflated p-values towards zero. Figure 3.12 shows that the p-values are slightly inflated for the first group separation, which leads to a marginal separation between groups, as visible from the UMAP (Figure 3.2). Further, from Figure 3.12 it is shown that for *DifferentialRegulation* FPs are never inflated. However, p-values are inflated towards one, hence *DifferentialRegulation* is more conservative as compared to the other methods. *eisaR* is only inflated for the first group separation, but overall approximately uniformly distributed. On the other hand, *BRIE2* demonstrates inflated FPs for all three group separations, which is also consistent with the results from the simulation study. *DEXSeq* was not evaluated for p-value distribution as it does not provide raw p-values at gene-level, whereas the other methods do.

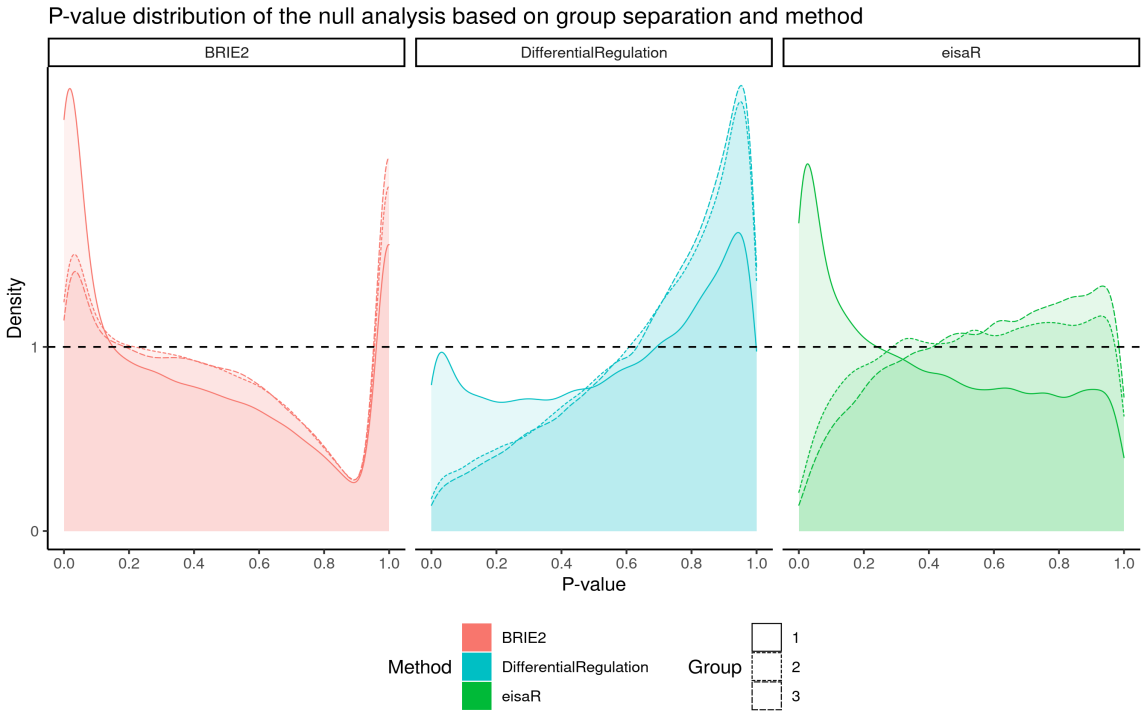


Figure 3.12: Dentity of raw p-values, obtained in the null real data analysis. The three lines refer to three possible group separations.

Further, we looked into the performance of the models based on the proportion of p-values and FDR values smaller or equal than the conventional significance levels of 10%, 5% and 1%. From Table 3.1 it is shown that *BRIE2* does not control the p-values particularly well as the values are inflated for all three levels. On the other hand, both *DifferentialRegulation* and *eisaR* do not have inflated p-values, whereas *DifferentialRegulation* is a bit more conservative than *eisaR*. Table 3.2 paints a similar picture as *BRIE2* produces by far larger values than the other methods including *DEXSeq*. With FDR values, one would usually want the values to be as small as possible. From that convention one can observe that *DifferentialRegulation* controls the FDR better than both *DEXSeq* and *eisaR*, whereas the latter two methods perform approximately

equally as good.

Table 3.1: Proportion of p-values smaller than the proposed significance levels 10%, 5% and 1%

	10%	5%	1%
BRIE2	0.215	0.155	0.083
DifferentialRegulation	0.059	0.035	0.014
eisaR	0.108	0.065	0.026

Table 3.2: Proportion of FDR values smaller than the proposed significance levels 10%, 5% and 1%

	10%	5%	1%
BRIE2	0.080	0.061	0.038
DEXSeq	0.016	0.012	0.006
DifferentialRegulation	0.006	0.005	0.003
eisaR	0.017	0.010	0.004

3.4 Computational benchmark

Ultimately, we compared the computational burden of the differential methods excluding alignment and quantification. Alignment and quantification were excluded from the benchmark as all methods use the same input generated from *alevin-fry*, therefore including it would be redundant. The computational benchmark was run on the null data and averaged across all three possible group separations. All methods were provided three cores on the same machine (internal server) to run the benchmark. However, due to a lack of control of parallel cores, *BRIE2* uses all cores available on a machine (up to 64 cores in our case); *eisaR* instead, does not allow for parallel coding, and only runs on one core. Figure 3.13 illustrates the average runtime of each differential method in minutes on a square root scaled axis, as there is a large difference in absolute runtime between the methods. From the Figure it is shown that *BRIE2* ran the longest - roughly 33.5 hours. It is fair to acknowledge that *BRIE2* developers suggest users to run the method on a GPU, which we did not do; this, would have likely decreased the computational burden of the approach. Although, at the same time, all methods used the same machine, and *BRIE2* used significantly more cores than any other method. *DifferentialRegulation*, which also uses intense computational Bayesian approach with full MCMC, took slightly less than one hour (55 minutes), but significantly less than *BRIE2* (about 37 times faster). The other two approaches, instead, require significantly less time: about 5 minutes for *DEXSeq*, and 1 minute for *eisaR*.

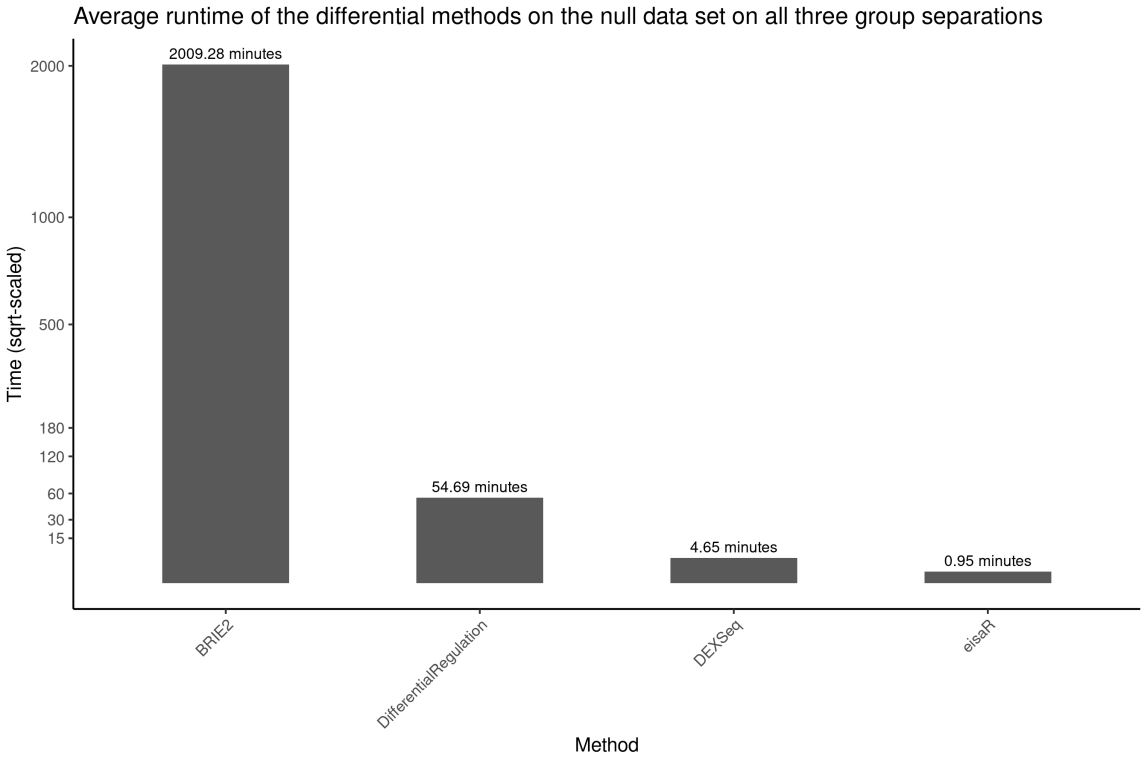


Figure 3.13: Runtime (in minutes) of the differential methods on the null data, averaged across all three possible group separations.

3.5 Data availability

Kidney mouse cells
The raw data can be downloaded from NCBI GEO (accession number GSE107585).
https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE107585

3.6 Code availability

All code for data preprocessing and analysis associated with the thesis is available at https://github.com/joelmeili/DifferentialRegulation.

Chapter 4

Discussion

4.1 Conclusion

In this thesis, we investigated how the relative abundance of spliced and unspliced reads differs between experimental conditions. Changes to these relative abundances are directly linked to gene regulation and methods that are capable of detecting these differences already exist (e.g. *BRIE2* and *eisaR*). We identified two main sources of mapping uncertainty: reads mapping to multiple genes, and reads mapping to both spliced and unspliced versions of a gene. We proposed two approaches to deal with these: i) *DEXSeq* on USA (unspliced / spliced / ambiguous) estimated counts, which accounts for the first, and *DifferentialRegulation* on the USA-based equivalence classes, which also uses a latent variable model for reads mapping to multiple genes, hence accounting for both sources of mapping uncertainty. We investigated the performance of all methods on two semi-simulated data sets. The semi-simulated data sets were created from real scRNA-seq data with four biological replicates. In a first step, we introduced an arbitrary differential effect to a subset of genes and cells by inverting the counts of spliced and unspliced reads. In one of the two simulations, we also added DGE to a second subset of genes and cells as an additional nuisance parameter. Multi-mapping uncertainty was considered next to make the semi-simulated data sets more realistic. This was achieved by simulating, at the read level, with *minnow* and subsequently aligning simulated reads with *alevin-fry*, where we used the two semi-simulated data sets as input for *minnow*. We then analysed the performance of *BRIE2*, *eisaR*, *DEXSeq* and *DifferentialRegulation* in detecting the differential genes by comparing TPR v. FDR curves. From this analysis we found that *DifferentialRegulation* controls the FDR well for both data sets in comparison to the other three methods. Additionally, we examined the results stratified by gene abundance levels to investigate how robust the methods are to gene abundance with and without DGE. From this analysis we concluded that *DifferentialRegulation* is the only method with good TPR and well calibrated FDR across all levels of gene abundance. Next, we did a null analysis on the original data set where we compared the distribution of p-values for all three possible group separations. We found that *BRIE2* had inflated p-values for all three group separations, whereas *DifferentialRegulation* had no inflated p-values at all. From the null analysis (where no differences between groups are expected) it was also shown that *DifferentialRegulation* seems to be marginally conservative as there was a tendency for inflation towards one. Ultimately, we ran a computational benchmark on the null data where we applied all four methods on all three possible group separations and averaged the runtime. From the computational benchmark it was shown that the better performance of *DifferentialRegulation* comes with a longer runtime, however, there is also the possibility to run the method without equivalence classes, which is slightly less accurate however much faster. From the previous analyses, we show that *DifferentialRegulation* had overall better results than the other methods, however, there are also some caveats to the method, for example, *DifferentialRegulation* cannot deal with additional covariates e.g. batch effects, whereas *BRIE2* and *DEXSeq* can.

4.2 Future directions

DifferentialRegulation has already been published on the Bioconductor project, which is an open source software that promotes reproducible analysis of data from emerging biological assays. Further, it is planned to extend the use case of *DifferentialRegulation* from only scRNA-seq data to bulk RNA-seq data, which allows to perform differential analyses at the transcript level. This will enable a double analysis framework: i) identify cell-type specific changes from scRNA-seq data, but at the gene-level due to low transcript resolution, and ii) discover individual differentially regulated transcripts from bulk RNA-seq data, although from an aggregation of cell types. Ultimately, the work of this thesis is part of a future paper which needs some additional analyses on different data sets and some changes to the simulation algorithm.

Bibliography

Amezquita, R. A., Lun, A. T., Becht, E., Carey, V. J., Carpp, L. N., Geistlinger, L., Marini, F., Rue-Albrecht, K., Risso, D., Soneson, C.,*et al.* (2020). Orchestrating single-cell analysis with bioconductor. *Nature methods*, **17**, 137–145. 15

Anders, S., Reyes, A., and Huber, W. (2012). Detecting differential usage of exons from rna-seq data. 10

Andrieu, C., De Freitas, N., Doucet, A., and Jordan, M. I. (2003). An introduction to mcmc for machine learning. *Machine learning*, **50**, 5–43. 12

Aran, D., Looney, A. P., Liu, L., Wu, E., Fong, V., Hsu, A., Chak, S., Naikawadi, R. P., Wolters, P. J., Abate, A. R., Butte, A. J., and Bhattacharya, M. (2019). Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage. *Nat. Immunol.*, **20**, 163–172. 15

Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, **57**, 289–300. 14

Crowell, H. L., Soneson, C., Germain, P.-L., Calini, D., Collin, L., Raposo, C., Malhotra, D., and Robinson, M. D. (2020). Muscat detects subpopulation-specific state transitions from multi-sample multi-condition single-cell transcriptomics data. *Nature communications*, **11**, 1–12. 5

Deng, L. (2012). The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, **29**, 141–142. 12

Dharshini, S. A. P., Taguchi, Y.-H., and Gromiha, M. M. (2020). Identifying suitable tools for variant detection and differential gene expression using rna-seq data. *Genomics*, **112**, 2166–2172. 5

Dobin, A., Davis, C. A., Schlesinger, F., Drenkow, J., Zaleski, C., Jha, S., Batut, P., Chaisson, M., and Gingeras, T. R. (2013). Star: ultrafast universal rna-seq aligner. *Bioinformatics*, **29**, 15–21. 4

Gaidatzis, D., Burger, L., Florescu, M., and Stadler, M. B. (2015). Analysis of intronic and exonic reads in rna-seq data characterizes transcriptional and post-transcriptional regulation. *Nature biotechnology*, **33**, 722–729. 8, 9

Garibyan, L. and Avashia, N. (2013). Research techniques made simple: polymerase chain reaction (pcr). *The Journal of investigative dermatology*, **133**, e6. 7

Haque, A., Engel, J., Teichmann, S. A., and Lönnberg, T. (2017). A practical guide to single-cell rna-sequencing for biomedical research and clinical applications. *Genome medicine*, **9**, 1–12. 2, 3

He, D., Zakeri, M., Sarkar, H., Soneson, C., Srivastava, A., and Patro, R. (2022). Alevin-fry unlocks rapid, accurate and memory-frugal quantification of single-cell rna-seq data. *Nature Methods*, **19**, 316–322. 4, 6, 7

Huang, Y. and Sanguinetti, G. (2021). Brie2: computational identification of splicing phenotypes from single-cell transcriptomic experiments. *Genome biology*, **22**, 1–15. 5, 9, 10

La Manno, G., Soldatov, R., Zeisel, A., Braun, E., Hochgerner, H., Petukhov, V., Lidschreiber, K., Kastriti, M. E., Lönnerberg, P., Furlan, A.,*et al.* (2018). Rna velocity of single cells. *Nature*, **560**, 494–498. 4, 5

Lee, J. M. (2018). *Introduction to Riemannian manifolds*, volume 176. Springer. 12

Love, M. I., Soneson, C., and Patro, R. (2018). Swimming downstream: statistical analysis of differential transcript usage following salmon quantification. *F1000Research*, **7**, 952. 10

McCarthy, D. J., Campbell, K. R., Lun, A. T. L., and Willis, Q. F. (2017). Scater: pre-processing, quality control, normalisation and visualisation of single-cell RNA-seq data in R. *Bioinformatics*, **33**, 1179–1186. 15

McDermaid, A., Chen, X., Zhang, Y., Wang, C., Gu, S., Xie, J., and Ma, Q. (2018). A new machine learning-based framework for mapping uncertainty analysis in rna-seq read alignment and gene expression estimation. *Frontiers in genetics*, **9**, 313. 5

McInnes, L., Healy, J., and Melville, J. (1802). Umap: Uniform manifold approximation and projection for dimension reduction. arxiv 2018. 12

Melsted, P., Booeshaghi, A., Liu, L., Gao, F., Lu, L., Min, K. H. J., da Veiga Beltrame, E., Hjörleifsson, K. E., Gehring, J., and Pachter, L. (2021). Modular, efficient and constant-memory single-cell rna-seq preprocessing. *Nature biotechnology*, **39**, 813–818. 4

Mohajon, J. (2020). Confusion matrix for your multi-class machine learning model. 14

Park, J., Shrestha, R., Qiu, C., Kondo, A., Huang, S., Werth, M., Li, M., Barasch, J., and Suszták, K. (2018). Single-cell transcriptomics of the mouse kidney reveals potential cellular targets of kidney disease. *Science*, **360**, 758–763. 15

Rao, M. S., Van Vleet, T. R., Ciurlionis, R., Buck, W. R., Mittelstadt, S. W., Blomme, E. A., and Liguori, M. J. (2019). Comparison of rna-seq and microarray gene expression platforms for the toxicogenomic evaluation of liver from short-term rat toxicity studies. *Frontiers in genetics*, **9**, 636. 2

Robinson, M. D., McCarthy, D. J., and Smyth, G. K. (2010). edger: a bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, **26**, 139–140. 5, 8

Sainburg, T., McInnes, L., and Gentner, T. Q. (2021). Parametric umap embeddings for representation and semisupervised learning. *Neural Computation*, **33**, 2881–2907. 13

Sarkar, H., Srivastava, A., and Patro, R. (2019). Minnow: a principled framework for rapid simulation of dscrna-seq data at the read level. *Bioinformatics*, **35**, i136–i144. 7, 8

Soneson, C., Matthes, K. L., Nowicka, M., Law, C. W., and Robinson, M. D. (2016). Isoform prefiltering improves performance of count-based methods for analysis of differential transcript usage. *Genome biology*, **17**, 1–15. 10

Srivastava, A., Malik, L., Smith, T., Sudbery, I., and Patro, R. (2019). Alevin efficiently estimates accurate gene abundances from dscrna-seq data. *Genome biology*, **20**, 1–16. 4

Stadler, M. B., Gaidatzis, D., Burger, L., and Soneson, C. (2020). eisar: Exon-intron split anaalysis (eisa) in r. R package version 1.0. 5, 9

Stark, R., Grzelak, M., and Hadfield, J. (2019). Rna sequencing: the teenage years. *Nature Reviews Genetics*, **20**, 631–656. 2

Tiberi, S. (2022). Differentialregulation: Differentially regulated genes from scrna-seq data. R package version 1.0.7. 11

Tiberi, S., Crowell, H. L., Weber, L. M., Samartsidis, P., and Robinson, M. D. (2021). distinct: a novel approach to differential distribution analyses. 5

Tiberi, S. and Robinson, M. D. (2020). Bandits: Bayesian differential splicing accounting for sample-to-sample variability and mapping uncertainty. *Genome biology*, **21**, 1–13. 10

Weiler, P., Van den Berge, K., Street, K., and Tiberi, S. (2021). A guide to trajectory inference and rna velocity. 4