Understanding transcriptional regulation through computational analysis of single-cell transcriptomics

Chee Yee Lim

University of Cambridge

This dissertation is submitted for the degree of Doctor of Philosophy.

Wolfson College

August 2017

I Abstract

Understanding transcriptional regulation through computational analysis of single-cell transcriptomics

Chee Yee Lim

Gene expression is tightly regulated by complex transcriptional regulatory mechanisms to achieve specific expression patterns, which are essential to facilitate important biological processes such as embryonic development. Dysregulation of gene expression can lead to diseases such as cancers. A better understanding of the transcriptional regulation will therefore not only advance the understanding of fundamental biological processes, but also provide mechanistic insights into diseases.

The earlier versions of high-throughput expression profiling techniques were limited to measuring average gene expression across large pools of cells. In contrast, recent technological improvements have made it possible to perform expression profiling in single cells. Single-cell expression profiling is able to capture heterogeneity among single cells, which is not possible in conventional bulk expression profiling.

In my PhD, I focus on developing new algorithms, as well as benchmarking and utilising existing algorithms to study the transcriptomes of various biological systems using single-cell expression data. I have developed two different single-cell specific network inference algorithms, BTR and SPVAR, which are based on two different formalisms, Boolean and autoregression frameworks respectively. BTR was shown to be useful for improving existing Boolean models with single-cell expression data, while SPVAR was shown to be a conservative predictor of gene interactions using pseudotime-ordered single-cell expression data.

In addition, I have obtained novel biological insights by analysing single-cell RNAseq data from the epiblast stem cells reprogramming and the leukaemia systems. Three different driver genes, namely *Esrrb*, *Klf2* and *GY118F*, were shown to drive reprogramming of epiblast stem cells via different reprogramming routes. As for the leukaemia system, FLT3-ITD and IDH1-R132H mutations were shown to interact with each other and potentially predispose some cells for developing acute myeloid leukaemia.

II Declaration

This dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration except as declared in the Preface and specified in the text.

It is not substantially the same as any that I have submitted, or, is being concurrently submitted for a degree or diploma or other qualification at the University of Cambridge or any other University or similar institution except as declared in the Preface and specified in the text. I further state that no substantial part of my dissertation has already been submitted, or, is being concurrently submitted for any such degree, diploma or other qualification at the University of Cambridge or any other University or similar institution except as declared in the Preface and specified in the text.

The total length of the main body of this dissertation is 52,830 words and therefore does not exceed the limit of 60,000 words for such a dissertation.

Chee Yee Lim

August 2017

III Acknowledgements

I would like to thank my supervisor, Prof. Bertie Gottgens, and my co-supervisor, Dr. Jasmin Fisher, for their helpful discussions and other contributions which made my PhD thesis possible.

I would also like to thank my collaborators who generated the single-cell RNAseq data that are crucial for my PhD thesis, including Hannah Stuart and Tim Lohoff from Jose Silva’s lab, as well as Konstantinos Tzelepis from George Vassiliou’s lab.

I would like to extend my gratitude for the inspiring scientific discussions that I had with fellow researchers, including Dr. Nir Piterman, Dr. Lorenz Wernisch, Dr. Sreenivas Chavali, Dr. Huange Wang, Dr. Xiaonan Wang, Dr. Fernando Calero-Nieto, and Dr. Steven Woodhouse; as well as fellow PhD students, including Dr. Wajid Jawaid and Matthew Clarke. I would also like to thank everyone that I have met in various seminars and conferences for their inputs and discussions.

I would also like to thank the services provided by my PhD programme, college and university in facilitating my PhD study. Without the academic and pastoral supports provided, I would not have been able to finish my PhD study and plan for my future accordingly.

Lastly, I am grateful for the eternal love and emotional support from my family and my partner, Wen Chiy Liew. Her unwavering support acts as a beacon that guides me through the pitch black stormy sea I waded through in my PhD.

IV Contents

V List of Figures

VI List of Tables

1 Introduction

1.1 Transcriptomics and expression profiling

The central dogma of molecular biology describes the information flow from stored genetic materials to functional biological units in most living organisms (Crick 1970). This flow of information starts with DNA, which acts as the permanent storage of genetic information. DNA then goes through transcription to produce RNAs, usually in the form of messenger RNAs (mRNAs), which act as the transient information transfer medium. Finally, the mRNAs get translated into proteins, which act as the functional biological units in various biological processes. Exceptions to the rules have been found with recent discoveries, such as the discovery of the non-coding RNAs. In humans, only about 3% of the genome encodes for proteins while up to 80% of the genome is found to be transcribed (Dunham et al. 2012). These transcripts include functional non-coding RNAs, such as microRNAs and long non-coding RNAs. In summary, studies performed over the years suggest that both RNAs and proteins represent important functional biological units in facilitating biological processes.

From DNA to RNA to protein, transcription and translation represent two key stages where the rate and specificity of the flow of information can be regulated to achieve specific biological goals, such as to control spatial and temporal gene expression. Transcriptional regulation is particularly interesting and has been widely studied, as this process gives rise to both non-coding RNAs that have biological functions as well as coding RNAs that lead to protein productions. Studying transcriptional regulation has allowed a better understanding of developmental processes, such as in embryonic (Boyer et al. 2005; Xu et al. 2010) and blood development (Orkin & Zon 2008; Moignard et al. 2015); as well as disease developments, such as leukaemia (Tenen et al. 1997; Suzuki et al. 2009).

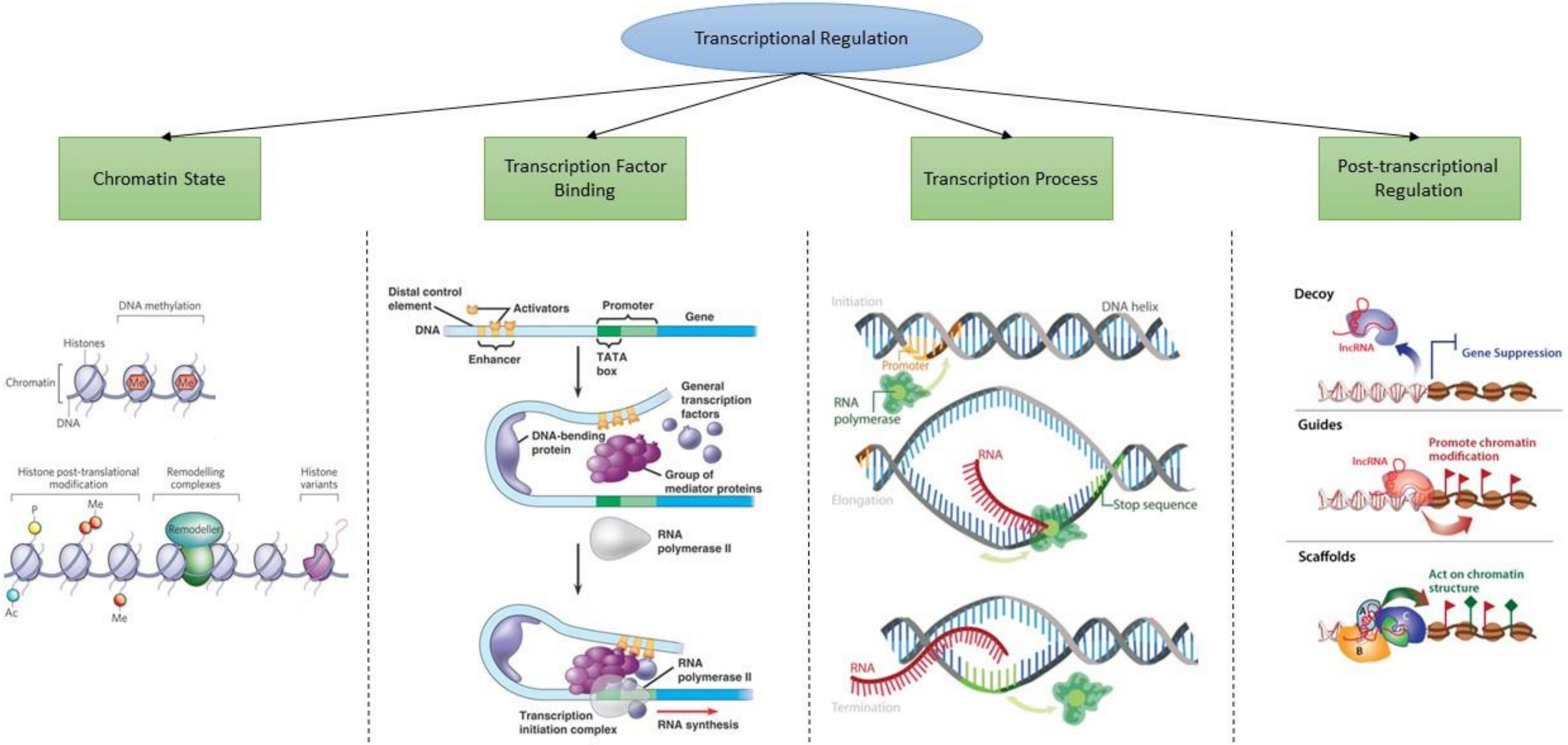


Figure 1.1 Overview of major stages of transcriptional regulation.
The diagrams show the major processes involved in each stage. Note that these processes do not occur in distinct stages, but with substantial crosstalks among the stages. Only mechanisms for lncRNA are used for illustrating post-transcriptional regulation due to space constraint. [Figure adapted from (Dulac 2010; Quia n.d.; CK-12 n.d.; BioCat n.d.; Wang & Chang 2011)]

1.1.1 Mechanisms of transcriptional regulations

The eukaryotic transcription of a gene is regulated in multiple stages (Figure 1.1), which are coordinated by many families of proteins. Note that for explanation purpose, the transcriptional regulation processes are described in distinct stages, but substantial crosstalks occur among the stages in reality. The first stage of transcription involves changing the state of chromatin. For inactive genes that are not transcribing, the local DNA around the genes is usually tightly compacted to prevent transcription by hindering protein access to the DNA (Grunstein 1990). This compaction of DNA is achieved with the help of histone proteins, which allow the DNA molecule to bind around the histone proteins. This protein-DNA complex forms a nucleosome subunit which in turn constitutes the chromatin. As the DNA enclosed in a nucleosome is generally transcriptionally repressed, large chromatin remodelling complexes, such as SWI/SNF family remodelers, are required to reposition and remove nucleosomes (Clapier & Cairns 2009). In addition, some chromatin remodelling complexes also catalyse the swapping of typical histone proteins with specific histone protein variants, such as the replacement of H2A histone protein by H2A.Z histone protein (Wu et al. 2005). H2A.Z is found to act as a buffer against gene silencing caused by the spread of heterochromatin proteins (Meneghini et al. 2003). Lastly, both the DNA and the histone proteins in the chromatin are usually modified to contain transcriptional signals such as methylation, phosphorylation and acetylation. The most well studied signal is the acetylation of histone by histone acetyltransferases (HATs) (Sterner & Berger 2000). HATs introduce acetyl groups to lysine residues in the histone, which neutralises the positive charge on histone tails. This process leads to the destabilisation of the chromatin structure which promotes transcription.

The second stage involves the binding of transcription factors, which can be activators or repressors, to the enhancer and promoter regions of the genes. Enhancer regions are defined as *cis*-acting transcriptional regulating DNA sequences that is independent of their orientation and distance relative to the transcriptional start site in a gene (Blackwood & Kadonaga 1998). In contrast, promoter regions, which are also *cis*-acting transcriptional regulating DNA sequences, are located immediately upstream of a gene. Each gene can possess multiple enhancer regions that contributes cumulatively to the spatial and temporal regulation of the gene, which also enables cell type-specific and development stage-specific expression of the gene. Transcription factors are classified into different families, such as SOX proteins and POU factors, that contain different binding domains that give rise to DNA sequence and protein binding specificity (Reményi et al. 2004). These transcription factors regulate transcription in a combinatorial fashion where multiple proteins act together in the form of a complex. The activation of transcription is achieved by promoting the recruitment and the establishment of the transcription complex, which contains a RNA polymerase at its core. In contrast, the inhibition of transcription can be achieved either by competitive binding between a repressor and an activator, or by the repressor binding directly to the activator protein to inhibit its activity (Latchman 1996).

Transcription occurs during the third stage, where a copy of RNA molecule which is complementary to the DNA molecule is created (Paule & White 2000). The transcription complex responsible for the process is made up of multiple proteins, which includes DNA helicases that unwind the two DNA strands, DNA-binding proteins that hold onto the DNA, and a RNA polymerase that synthesises the RNA. There are three stages in the transcription process, namely the initiation, elongation and termination stage. Each of these stages is further regulated by multiple proteins. For example, the mediator proteins promote transcription initiation by interacting with transcription initiation factors, such as TFIIE and TFIIH (Esnault et al. 2008). These transcription initiation factors are essential for maintaining the stability of the transcription complex to prevent abortive initiation (Saunders et al. 2006). In addition, histone acetylation is shown to increase the rate of transition from initiation to elongation by promoting RNA polymerase II escape from promoter (Stasevich et al. 2014). Transcription has also been observed to be paused during the elongation stage, possibly due to hindrance from nucleosomes (Core et al. 2008). The transcription process then continues until being terminated when the transcription complex arrives at a series of transcriptional termination signals, such as the transcription termination factor (TTF)-I that binds to the 3' end of a gene (Sander & Grummt 1997). The transcribed mRNAs then undergo several pre-processing steps that are regulated, such as splicing and polyadenylation, to yield the final mature mRNAs (Moore & Proudfoot 2009).

The fourth stage is the post-transcriptional regulation, which usually includes non-protein coding RNAs such as microRNAs (miRNAs) and long non-coding RNAs (lncRNAs). miRNAs consist of multiple families of 21-bp long RNAs that regulate gene expression by partnering with proteins, such as AGO proteins, to form ribonucleoprotein complexes (Peters & Meister 2007). miRNAs regulate gene expression via multiple ways, such as degrading the mRNAs by inducing deadenylation (Wu et al. 2006), inhibiting mRNA translation by interacting with the 5' cap and the ribosomes (Pillai et al. 2005; Chendrimada et al. 2007), or by inducing proteolysis in proteins that are being translated (Nottrott et al. 2006). In contrast, lncRNAs, which have only been studied in details recently, are RNAs that are longer than 200 bp and do not code for proteins [See review (Rinn & Chang 2012)]. Together with protein partners, lncRNAs regulate gene expression through three major ways: by acting as a decoy to competitively occupy transcription factors (Kino et al. 2010), by acting as a scaffold to recruit multiple proteins into forming complexes (Spitale et al. 2011), and by targeting other gene regulating proteins to specific genomic regions (Jeon & Lee 2011).

Lastly, the genes finally lead to protein productions via mRNAs after going through the multistage transcriptional regulation process. It should be noted that while some transcriptional regulators mentioned here represent core machinery that is commonly used in many cell types and biological processes, most transcriptional regulators are specific to certain cell types and biological processes. It is this transcriptional specificity that helps in driving the derivation of diverse cell types from the same genetic material in a multicellular organism. Therefore in order to dissect the biological mechanisms underlying any developmental process, it is important to understand what are the target genes of the transcriptional regulators and how they regulate a specific developmental process.

1.1.2 Variability in gene expression

The high variability of gene expression in single cells is widely known before the development of high-throughput single-cell expression profiling techniques [See review (Raj & van Oudenaarden 2008)]. This variability exists even when the cells have the same genetic identity and are cultured under the same environmental condition. The exact causes of this stochasticity are unknown, but studies suggest that the noise in gene expression is likely to be due to transcriptional regulatory mechanisms, such as the regulations exerted by the chromatin state and the transcription factors (Becskei et al. 2005; Murphy et al. 2007). Genes were observed to be transcribed in a burst-like fashion, in which the genes are randomly switching between active and inactive states (Figure 1.2). The transcriptional burst contributes to fluctuations in gene expression values due to the random time intervals between transcriptional bursts. Interestingly, studies found that genes which exhibit the same noise signature are typically found in the same pathways or biological processes (Bengtsson et al. 2005; Sigal et al. 2006). Therefore, it is possible to potentially locate genes involved in the same biological process by examining correlations in the noise signature between genes.

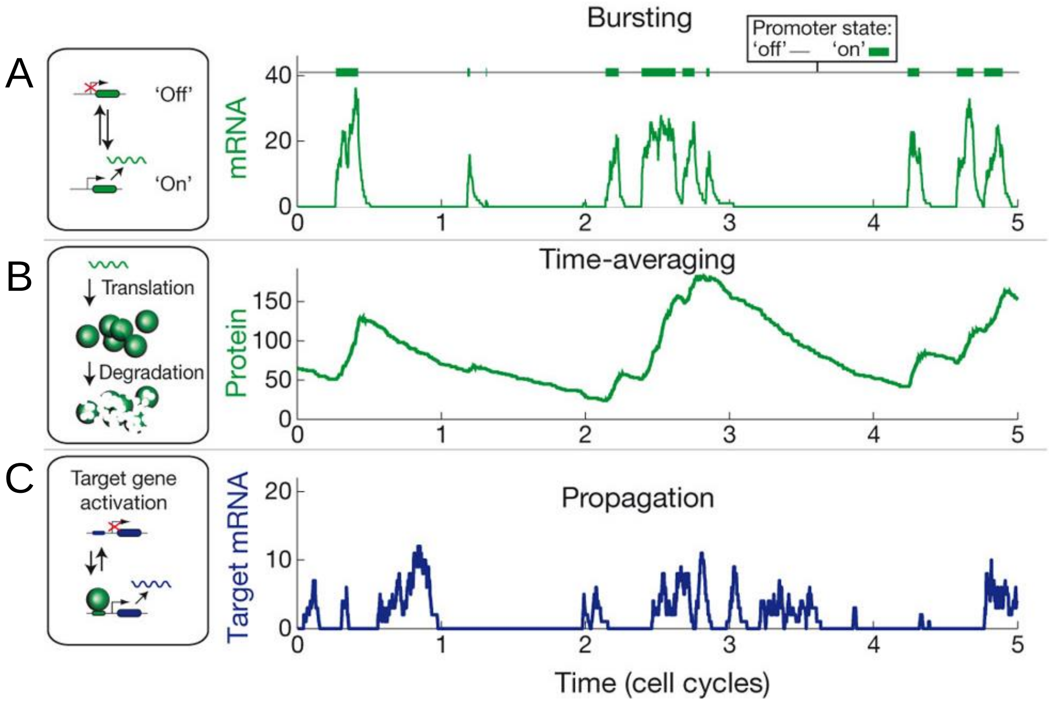


Figure 1.2 Current understanding of the noise at the transcript and protein levels. *(A) Changes in the transcript level over time due to fluctuations between on and off state of gene expression. (B) Changes in the protein level over time due to translation and degradation. (C) Changes in the transcript level of target gene where the gene is only expressed when bound by the protein encoded by the gene discussed in (A) and (B). [Figure adapted from (Eldar & Elowitz 2010)]*

The stochasticity discussed above was in terms of the variability observed in each individual gene. Studies have also been done to study the variability of gene expressions in a network of genes. They have found that in general noise in the upstream gene gets amplified along a gene cascade, but a long gene cascade can dampen rapid fluctuations in the expression of the upstream gene (Rosenfeld et al. 2005; Hooshangi et al. 2005). In addition, negative feedback in the gene network helps reducing noise, while positive feedback acts as a genetic switch in which the stochastic noise can sometimes flip the gene between the bi-stable state of an “on” state and an “off” state (Hasty et al. 2000; Austin et al. 2006).

The variability in a biological system is expected to be both beneficial and detrimental to biological processes. The reason for variability to be detrimental is easy to understand, as a noisy system will exhibit random behaviours and therefore make the system unreliable. However, variability can also be beneficial as it enables probabilistic differentiation of cells that are identical otherwise. Beneficial variability has been demonstrated in the development of olfactory system in mice (Vassar et al. 1993; Tsuboi et al. 1999). In order to develop the ability to distinguish many odours, each olfactory neuron randomly expresses a specific odorant receptor in a mutually exclusive fashion. Variability has also shown to be important for the blood development (Hume 2000). A study by (Chang et al. 2008) found that variability in Sca-1 protein in each individual cell correlates strongly with the probability of the cell developing into either erythroid or myeloid lineage. In addition, variability has been found to be important for early mouse embryo development (Yamanaka et al. 2010; Morris et al. 2010). Cells in inner cell mass have been shown to decide their cell fates through a stochastic process with a lineage bias.

1.1.3 Expression profiling tools

Many high-throughput expression profiling techniques have been developed, which include multiplex quantitative polymerase chain reaction (qPCR), microarray and RNA sequencing (RNAseq). However, the earlier versions of high-throughput expression profiling techniques were limited to measuring average gene expression across large pools of cells. In contrast, recent technological improvements have made it possible to perform expression profiling in single cells [See reviews (Shapiro et al. 2013; Wang & Song 2017)]. Protocols for the single-cell equivalent of microarray (Ramos et al. 2006), qPCR (Ståhlberg & Bengtsson 2010) and RNAseq (Tang et al. 2009) have been developed. In particular, single-cell RNAseq has been adopted widely due to the advantages of having absolute quantification and the ability to detect new isoforms. Multiple new sequencing library construction methods for single-cell RNAseq have been developed, which include Smart-seq2 (Picelli et al. 2014) and Drop-seq (Macosko et al. 2015) (Figure 1.3). New technology such as Drop-seq allows single-cell RNAseq to be scaled up to thousands of cells in a cost-effective manner.

One of the key advantages of single-cell expression profiling is that it enables the analysis of cell subpopulations that are rare in number, such as tissue-specific or cancer stem cells. The use of non-single-cell expression profiling techniques will result in the averaging of expression values across all cell subpopulations present in the samples, therefore masking any heterogeneity among cells present within the samples. The use of single-cell expression profiling has discovered the widespread presence of heterogeneity within seemingly homogenous cell populations, as evident in studies performed across different biological systems (Wilson et al. 2015; Buettner et al. 2015; Scialdone et al. 2016). In addition, single-cell expression profiling enables the dissection of spatial and temporal resolutions of transcriptional changes during biological processes, such as in embryonic development (Yan et al. 2013; Moignard et al. 2013; Moignard et al. 2015).

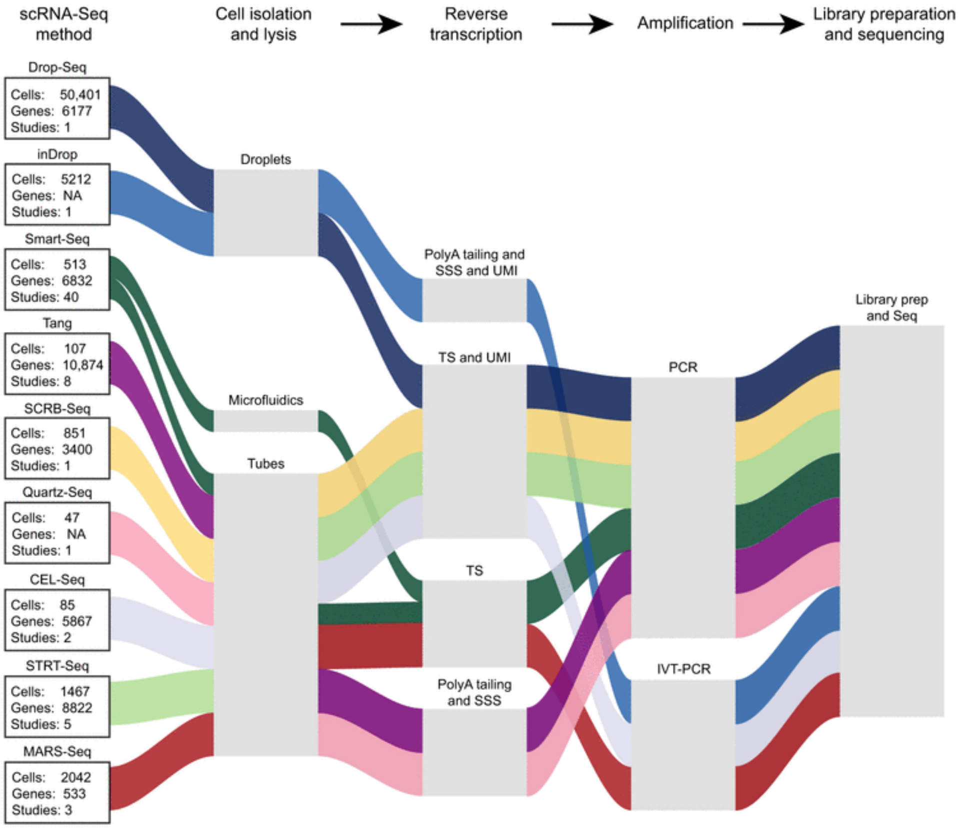


Figure 1.3 Protocols for single-cell RNA sequencing.
Tubes refer to manual isolation of cells using FACS or micropipetting into tubes. UMI, unique molecular identifiers; TS, template switching; SSS, second-strand synthesis; IVT-PCR, in vitro transcription polymerase chain reaction. [Figure adapted from (Kumar et al. 2017)]

The gene expression variability observed in single-cell RNAseq is likely to be a combination of both biological and technical variabilities. The sources of biological variability are varied and complex (as discussed in Section 1.1.2), where some noises may contribute to the phenotypes observed in the system while other noises are not relevant to the phenotypes studied. To complicate matters further, technical variability also comes from multiple sources. Some sources of technical variability are common to all experiments, such as the batch effect where the samples are being prepared in multiple batches with slightly different external conditions. Other sources of technical variability are specific to single-cell expression profiling, where the two major sources being drop-outs and overdispersion (Kharchenko et al. 2014). These noises arise due to the low amount of input mRNAs in a single cell. Drop-outs refer to genes that are expressed, but their expressions are not captured in the expression profiles of some cells. The occurrence of drop-outs is due to the low efficiency of mRNA capture from each cell, which is around 10% (Ramskold et al. 2012; Hashimshony et al. 2012). In contrast, overdispersion refers to unusually low or high levels of expression recorded for a gene in some cells. The causes of overdispersion are less clear, which may be due to the amplification bias of the PCR. The effects of both drop-outs and overdispersion can be partially mitigated by the use of improved protocols, such as the use of unique molecular identifiers (UMIs) (Islam et al. 2013).

1.2 Single-cell RNAseq bioinformatics tools

Because single cell analysis commonly reports expression states for hundreds or thousands of individual cells, this unique property offers new opportunities for the development of algorithms that can utilise this increased data resolution and handle the increased computational complexity due to the data volume. In addition, these algorithms also need to account for the increased technical noise which is uniquely present in single-cell RNAseq. Many algorithms have been developed for various analyses of single-cell RNAseq, ranging from normalisation to differential gene expression analysis. A typical single-cell RNAseq bioinformatics processing pipeline is shown in Figure 1.4.

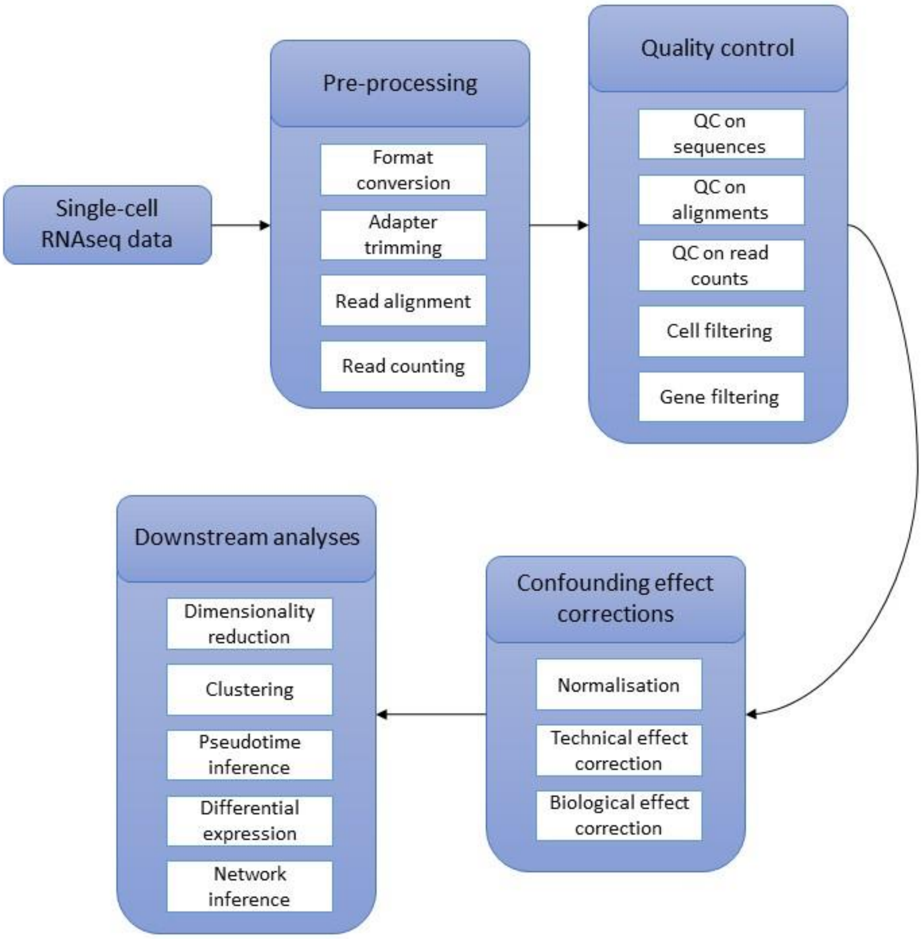


Figure 1.4 Typical single-cell RNAseq bioinformatics processing pipeline.

1.2.1 Normalisation

Normalisation is one of the most important pre-processing steps for all high-throughput expression profiling experiments. Normalisation aims to correct for the differences in library sizes across all samples in the experiments, so as to make them comparable in downstream analyses. The reads are also normalised by the gene lengths, as longer genes are more likely to have a higher number of reads detected. However, normalisation by gene lengths is usually not required for protocols that are biased for the 3' ends of the mRNAs.

Most normalisation methods that were initially used to normalise single-cell RNAseq data were originally developed for bulk RNAseq data. These methods include reads per million (RPM), DESeq normalisation (Anders & Huber 2010) and Trimmed Mean of M values normalisation (TMM) (Robinson & Oshlack 2010), which is implemented in edgeR (Robinson et al. 2010). RPM normalisation works by dividing reads from each gene with the total reads in each cell before multiplying by a million. This method may skew gene expression values if there are some genes that are both very highly expressed and differentially expressed among the cells. DESeq normalisation is done by scaling each cell with a size factor which is the median across genes on the ratio of each gene expression value to the gene's geometric mean across cells. TMM normalisation is computed by calculating the weighted mean of log fold change between the test and reference samples, after excluding the most highly expressed genes and the genes with the largest log fold change.

While both DESeq and TMM normalisation methods have been shown to be the best performing normalisation methods for bulk RNAseq (Dillies et al. 2013), both methods gave biased results for single-cell RNAseq (Vallejos et al. 2017). This is because single-cell RNAseq data possess very different technical properties from bulk RNAseq data. For example, the calculation of DESeq normalisation, in particular the use of geometric mean, is severely affected by the presence of zero inflation in the data due to drop-outs. Geometric mean is only clearly defined for genes with non-zero expression values across all cells.

Recent methods have been developed specifically for the normalisation of single-cell RNAseq data, which include scran (L. Lun et al. 2016). Scran normalisation calculates normalisation factors on pooled cells by summing expression values across cells in a pool and divide by an average reference background value. Scran normalisation is more robust than other bulk-based methods in the presence of zero inflation and unbalanced differential expression of genes across samples (Vallejos et al. 2017). Other single-cell normalisation methods are typically implemented as part of a data analysis pipeline (Fan et al. 2016), which typically consists of differential expression analysis. Such implementation limits the flexibility of using the normalisation in combination with other analyses. Some examples of these methods are BASiCS (Vallejos et al. 2015) and SAMstrt (Katayama et al. 2013). Note that SAMstrt utilise spike-ins for normalisation, which may not be ideal as spike-ins may vary in ways that are independent from other genes (Vallejos et al. 2017).

1.2.2 Confounding effects

Confounding effect corrections attempt to account for or remove confounding differences among samples or cells that are not of biological interest. Technically, normalisation can be considered a type of confounding effect correction that corrects specifically for differences in library sizes. Unfortunately, there exists many other known and unknown factors that are contributing to undesired variability among samples or cells besides library sizes. These factors may be due to technical reasons such as the batches in which the samples are being processed, or due to biological reasons such as the cell cycle phases.

The technical sources of undesired variability typically come from the variations in the experimental conditions due to changes in environment, equipment, reagents or personnel. These confounding effects, in particular those due to technical reasons, should be accounted for or corrected before performing further analyses. Otherwise, these confounding effects may lead to weak or invalid conclusions because the variability due to confounding effects is more influential than the variability due to biological factors of interest (Leek et al. 2010). Leek et al. found that in most high-throughput expression datasets, neither sample processing dates nor biological factors account for most of the variability observed, where most variability is shown to be caused by unknown sources. Recently, a study assessed the confounding effects in published single-cell RNAseq datasets and found high correlations between biological factors of interest and sequencing runs (Hicks et al. 2017). This result suggests that care should be taken when designing future experiments to reduce or prevent such correlations. This is because high correlations between biological factors of interest and technical variabilities cannot be easily corrected via post hoc bioinformatics corrections without proper experimental designs.

As for the biological sources of undesired variability, they usually come from the variations introduced by other known or unknown biological factors that are not of interest in the studies. One such example which is common in single-cell RNAseq is the differences in cell cycle phases. The differences in cell cycle phases may exert a very strong effect that mask the weaker signals from the differences in other biological processes. Algorithms have been developed to account for, such as the cyclone classifier (Scialdone et al. 2015); or to correct the cell cycle phase differences, such as single-cell latent variable model (scLVM) (Buettner et al. 2015). The cyclone classifier was trained on a single-cell RNAseq dataset where the cell cycle stage of each cell is known, and works by comparing the relative expressions of pairs of genes known to correspond to cell cycle phases. scLVM works by estimating a latent variable that best explained the variance observed in a set of known cell cycle genes, which is then used for variance decomposition to obtain the cell-cycle corrected expression data. Note that a further study suggests that sometimes the latent variable identified using scLVM may not be due to cell cycle, hence scLVM should be used with caution (McDavid et al. 2016).

There are two steps in handling confounding effects. Firstly, the confounding effects should be identified and quantified using exploratory statistical analyses, such as principal component analysis (PCA) and hierarchical clustering. In terms of PCA, the top principal components that do not correlate well with biological factors of interest are likely to be explained by confounding factors. In terms of hierarchical clustering, it is expected that the major differences among samples should be due to biological factors of interest. In addition, there are also algorithms that are specifically designed to estimate hidden confounding factors that are not known in advance. Some examples of these algorithms include the surrogate variable analysis (SVA) (Leek & Storey 2007) and svLVM (Buettner et al. 2015).

Once the confounding effects are identified and quantified, the confounding factors can then be either accounted for by including the factors in the formulation of models for downstream analyses, or corrected by regressing out the confounding effects to give a corrected expression data. Many existing algorithms support the accounting of confounding effects through the inclusion of additional factors into the models, which are typically linear models. For example, this can be done easily in differential expression analysis algorithms such as DESeq2. The correction of confounding factors is slightly more involved, but allows more freedom in downstream analyses as the corrected expression data can be used as it is. An example of such algorithms includes ComBat (Johnson et al. 2007) and PEER (Stegle et al. 2012). Note that ComBat is designed for microarray and therefore may not be suitable for single-cell RNAseq. Lastly, it should be noted that confounding effects can only be accounted for or corrected if they do not correlate fully with the biological factors of interest.

1.2.3 Dimensionality reduction

In terms of dimensionality reduction, most existing methods can be readily applied to single-cell RNAseq data. This is because single-cell RNAseq data is similar with other high-dimensional data typically analysed with dimensionality reduction methods, which are usually very noisy and can have up to millions of dimensions in the field of text and image processing. The aim of dimensionality reduction analyses is to convert the high-dimensional data into a low-dimensional map while preserving as much information as possible. The type of information preserved is dependent on the dimensionality reduction methods used. Some methods emphasise dissimilarities among the data points by showing disjoint clusters (e.g. tSNE), while other methods emphasise similarities among the data points by connecting them into a continuous process (e.g. diffusion map).

Dimensionality reduction methods can be separated into two major classes, namely linear and non-linear methods. An example of linear methods include principal component analysis (PCA) (Hotelling 1933); while examples of non-linear methods include kernel PCA (Scholkopf et al. 1998), diffusion map (Nadler et al. 2005) and t-distributed stochastic neighbour embedding (tSNE) (van der Maaten & Hinton 2008). PCA calculates orthogonal principal components that maximise the explained variations in the data. As for the non-linear methods, they rely on the use of kernels to represent data points in a low-dimensional non-linear subspace. Similar with other statistical approaches, trade-off exists between linear and non-linear methods. While non-linear methods may offer more informative low-dimensional maps, they typically have more hyperparameters to tune that may be subjected to human bias.

Most dimensionality reduction methods have been used to analyse single-cell RNAseq data, such as PCA and tSNE. The methods were used to either visualise the data for exploratory analysis, or to generate dimensionally reduced data for further analyses such as clustering or pseudotime inference. Examples of dimensionality reduction methods integrated as part of clustering or pseudotime inference algorithms will be discussed in the next sections. Among these dimensionality reduction methods, a new method, ZIFA, has been developed specifically for single-cell RNAseq to take account of the zero-inflation due to drop-outs (Pierson & Yau 2015). ZIFA is a non-linear method that extends factor analysisand has been shown to offer more robust results compared to PCA.

1.2.4 Clustering

Clustering methods represent an active field of new algorithm development for single-cell RNAseq. This is because the aim of most single-cell RNAseq experiments is to detect subpopulations of cells with different expression profiles in a cell population that is assumed to be homogeneous (Figure 1.5). In order to achieve this aim, clustering methods were used to cluster cells by their expression profiles into distinct groups. Most new clustering methods developed for single-cell RNAseq can be separated into two categories, namely hierarchical clustering-based and graph-based. The key difference between hierarchical and graph-based clustering is that hierarchical clustering assumes an underlying tree structure relationship that connects all clusters, but graph-based clustering is not constrainted to a tree structure and hence allows for a more complex relationship between the clusters. New hierarchical-based clustering methods usually use different distance metric and tree-building techniques. Some examples of these methods include ICGS (Olsson et al. 2016), SIMLR (Wang et al. 2017) and SC3 (Kiselev et al. 2017). ICGS performs iterative clustering with the HOPACH algorithm in order to select for final clusters with genes that are highly correlated within clusters but lowly correlated among clusters. SIMLR is based on a distance function that is a linear combination of several Gaussian kernels with different hyperparameters, where the weight of each kernel is learned from the expression data.

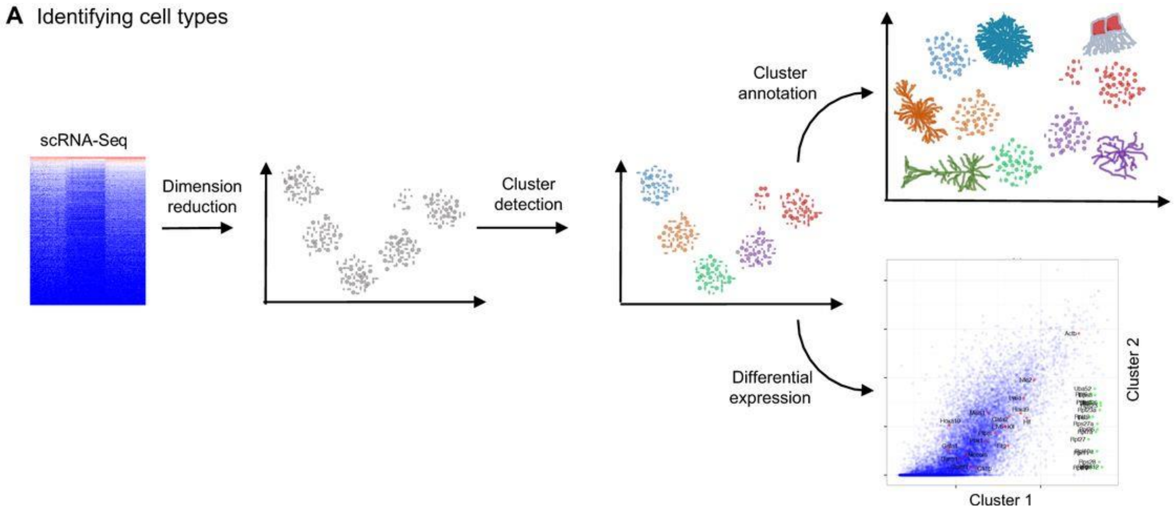


Figure 1.5 Application of clustering in single-cell RNAseq.
[Figure adapted from (Kumar et al. 2017)]

In graph-based clustering, each cell is considered a node, with the edge lengths between the cells being the similarity measures between the cells. Depending on the methods, the space where the cells are located in may not have any meaning, or it may be a dimensionally reduced space that relates to the similarity measures. Thresholding or preliminary clustering were usually applied to the fully connected graph of cells in order to get a sparser graph that prevents short-circuit edges that wrongly connect irrelevant pairs of cells (Balasubramanian et al. 2002). Clustering can then performed on the graph with community detection algorithms that search for groups of nodes in the graph (Fortunato 2009). Some examples of graph-based clustering designed for single-cell RNAseq include SNN-Cliq (Xu & Su 2015) and SPRING. SNN-Cliq searches for clusters in the form of quasi-cliques in the shared-nearest neighbour graph, which measures the similarity between two cells in terms of their connectivity to the neighbourhood. SPRING generates a force-directed graph from a k-nearest neighbour graph which are useful for identifying clusters (Weinreb et al. 2017).

The key consideration when using any clustering method is the input gene set used for clustering, which is a feature selection problem (Guyon & Elisseeff 2003). The gene set used for clustering is essential to obtain distinct clusters, as not all genes are differentially expressed among the cell subpopulations of interest. The inclusion of non-informative genes is likely to reduce the effectiveness of identifying cell subpopulations due to the presence of non-relevant variations in gene expression. Most clustering methods require manual selection of the gene set used for clustering, which can be done for example by selecting the set of all differentially expressed genes via performing differential expression analysis across known samples. Some clustering methods utilise an unbiased feature selection approach to choose the most informative gene set as part of the clustering routine, such as ICGS.

1.2.5 Pseudotime inference

Besides the detection of cell subpopulations, the other common aim of single-cell RNAseq experiments is to reconstruct the temporal progression in expression states across a biological process, such as during the derivation of differentiated cells from stem cells (Figure 1.6). This temporal reconstruction using single-cell RNAseq data can be achieved by pseudotime inference. Pseudotime inference algorithms aim to arrange the cells in a pseudotime trajectory that represent the underlying continuous biological process based on their expression profiles.

The concept of pseudotime is introduced to represent asynchronous developmental cellular progressions in which the cells collected from a particular time point are not all at the same state of developmental progression (Cannoodt, Saelens & Saeys 2016). In addition, most pseudotime inference algorithms also attempt to identify any branches in the trajectory that represent decision points in the underlying biological process. Understanding the decision points is particularly important for studying the developmental process, as the cells in two branches beyond a decision point have typically adopted two distinct cell identities. Reconstructing the branches can be considered as a particularly difficult clustering problem, as the algorithms need to identify disjoint clusters that are placed on branches, as well as the preceding common cluster that connects to these disjoint clusters in a continuous process.

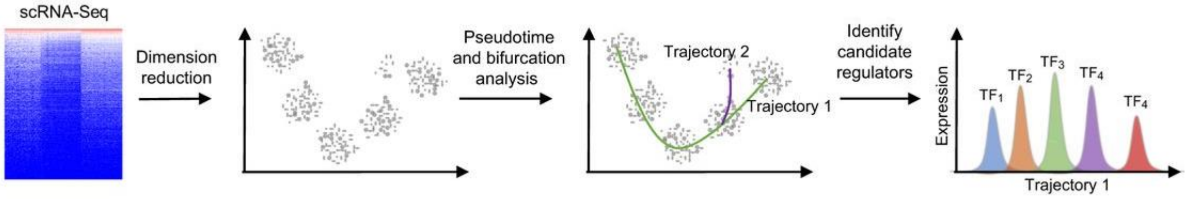


Figure 1.6 Application of pseudotime inference in single-cell RNAseq.
[Figure adapted from (Kumar et al. 2017)]

Most pseudotime inference algorithms contain two major steps, with some algorithms having an intermediate clustering step (Cannoodt, Saelens & Saeys 2016). The first step in pseudotime inference usually involves generating a low-dimensional, usually in two dimensions, representation of the high-dimensional expression data. The last step involves finding a path through the cells in the dimensionality reduced space, thereby giving an order to the cells in the form of a trajectory. Some algorithms only locate a single path through the cells, but most algorithms also attempt to detect divergent bifurcation points along the path which result in multiple branches. The distance among the cells along the trajectory path is measured in terms of pseudotime, which can be calculated using the reduced low-dimensional or original high-dimensional representation of the gene expression data.

One of the first pseudotime inference algorithms developed for single-cell RNAseq is Monocle (Trapnell et al. 2014). Monocle firstly uses independent component analysis (ICA) for dimensionality reduction, and then uses minimum spanning tree with Euclidean distance to connect the cells into a trajectory. Recently an improved version of Monocle, known as Monocle2, is developed (Qiu et al. 2017). In Monocle2, a non-linear principal graph-based method, DDRTree, is used for dimensionality reduction instead of the linear ICA. Besides

Monocle, another pseudotime inference algorithm, Wanderlust, which was initially designed for mass cytometry data was released around the same time (Bendall et al. 2014). Wanderlust firstly constructs a k-nearest neighbour graph, and then computes the average minimum path that passes through user-defined start cell and random waypoint cells on the graph. Recently Wanderlust has been extended into Wishbone, which has the ability to infer bifurcation branching points (Setty et al. 2016). There are also many other pseudotime inference algorithms developed, such as SLICER (Welch et al. 2016), embeddr (Campbell et al. 2015) and TSCAN (Ji & Ji 2016).

1.2.6 Differential expression

Differential expression analysis represents the key analysis step across most high-throughput expression profiling experiments. The aim of differential expression analysis is to detect gene expressions that are significantly different statistically between pairs of sample groups. The genes that are differentially expressed in different samples are assumed to be biologically important in distinguishing and understanding the different samples. The typical downstream analysis that follows differential expression analysis is gene set enrichment analysis that provides functional annotations to the sets of differentially expressed genes. This analysis allows the comparison of differentially regulated biological processes among sample groups, rather than just comparing differentially expressed genes. Note that the gene set enrichment analysis here refers to the broad class of methods for functionally annotating genes, and not specifically to the GSEA algorithm which tests for differences in pre-defined gene sets between two biological conditions (Subramanian et al. 2005).

The single-cell RNAseq algorithms for differential expression analysis can be separated into three major groups, namely conventional statistical tests, tests designed for bulk RNAseq and tests designed for single-cell RNAseq. Conventional statistical tests, which include Wilcoxon rank sum test and Kolmogorov-Smirnov test, are becoming increasingly applicable to single-cell RNAseq data due to the increased number of cells sequenced in each experiment. The higher number of cells offer increased statistical power to the conventional statistical tests used, which are typically non-parametric and therefore require less assumptions on the properties of expression data. As for the tests designed for bulk RNAseq, there are two major methods that are well-established, namely edgeR (Robinson et al. 2010) and DESeq2 (Love et al. 2014). Both edgeR and DESeq2 model gene expression in different samples with negative binomial linear models, and use likelihood ratio test or Wald test to compare these models in order to detect differentially expressed genes.

Single-cell RNAseq-specific differential expression tests that have been developed include SCDE (Kharchenko et al. 2014) and MAST (Finak et al. 2015). These methods typically contain model extensions that account for additional sources of technical noise specific to the single-cell RNAseq. SCDE performs differential expression analysis with a mixture of two distributions, with the first being a negative binomial distribution that models expression levels, and the second being a Poisson distribution that models dropouts. MAST utilises a two-part generalised linear model to model the fraction of cells that express a certain gene and the level of expression for each gene separately. Recently, SCDE has been improved further to yield better computation speed and results, as well as being incorporated into a pipeline called PAGODA (Fan et al. 2016).

Differential expression analysis can be performed across discrete variables (e.g. multiple samples), as well as across continuous variables (e.g. pseudotime). The most common use case of differential expression analysis in single-cell RNAseq is to compare sample differences, therefore requiring only discrete variables that distinguish each sample or continuous variables acting as weights for each sample. Most of the algorithms discussed above are designed with this use case in mind. However, with the advent of single-cell RNAseq data coupled with pseudotime inference, another interesting use case of differential expression analysis is to detect genes that are differentially expressed across the pseudotime, possibly also differentially expressed among different trajectory branches. The simplest method that can be used in this case is just simple correlation between gene expression and pseudotime. Another method, which is implemented as part of Monocle2, is to firstly fit a spline model on gene expression against pseudotime, then perform a likelihood ratio test between a model fitted on pseudotime and a null model. While the use case with temporal differential expression is similar in concepts with sample-wise differential expression, temporal differential expression contains additional information in the form of pseudotime and branches that will benefit from tailored differential expression algorithms.

1.3 Gene network inference

1.3.1 Characteristics of networks

Before going into details on biological networks, it is important to understand networks and their associated technical properties. A network, also known as a graph, is defined by a set of n nodes connected by edges. The edges can be directed or weighted to indicate the direction or the strength of the interactions respectively. The topological information in a network can be represented compactly with a $n \times n$ adjacency matrix, where each element in the matrix indicate the presence of an edge between two nodes with a binary value. The adjacency matrix is symmetric for an undirected network, and asymmetric for a directed network. In a weighted network, the elements in an adjacency matrix indicate the weights of edges instead, where zero values show the absence of edges while non-zero values correspond to the weights.

The topological properties of a network can also be described by several measures, such as lower-order measures (e.g. degrees) and higher-order measures (e.g. clustering coefficients). The degree of a node refers to the number of neighbouring nodes connected by edges. When viewing the network as a whole, the degree distribution $P(k)$ represents the fraction of nodes with degree k. The degree distribution is particularly interesting, as it is a measure that allows us to distinguish between different classes of networks, such as random and scale-free networks. Scale-free networks, where the degree distribution follows the power-law, are an important class of network, because many real-life networks such as social networks and biological networks are scale-free networks (Barabasi & Oltvai 2004). A scale-free network has mostly low-degree nodes and some high-degree nodes that connect to a significant number of other nodes, where in biology the high-degree nodes correspond to important global regulators such as the tumour suppressor p53 protein (Kruse & Gu 2009). Scale-free networks have a few interesting features, which include being highly robust against node failures (Albert et al. 2000) and have better dynamic controllability (Nepusz & Vicsek 2012). In contrast, higher-order measures like clustering coefficients describe the connectivity of local subgraphs, instead of focusing on individual nodes. The structures of local subgraphs, also known as motifs, have important roles in biological networks. The motifs act as fundamental components in forming modular networks, where each motif serve a defined function in the network (Wong et al. 2012). An example of motifs include feedforward loop, which plays important regulatory roles in many genetic systems (Mangan & Alon 2003).

1.3.2 Biological networks

Correctly identifying the underlying interaction network in biological systems has been a key common goal in many fields of biological studies. This is based on the understanding that while each individual biological factor may have unique functions, the ultimate contribution to the phenotypes exhibited by a biological system comes from a combination of multiple factors in the form of a network in the system (Barabasi et al. 2011). These biological factors contribute in varying degrees and in different ways to the system, as well as interacting with one other to mediate each other's response. While the bottom up approach of studying each biological factor in isolation allows a clear understanding of the properties of each gene, it is essential to ultimately view and analyse the biological system as a whole in a top down approach in order to understand the behaviour of a biological system (Figure 1.7).

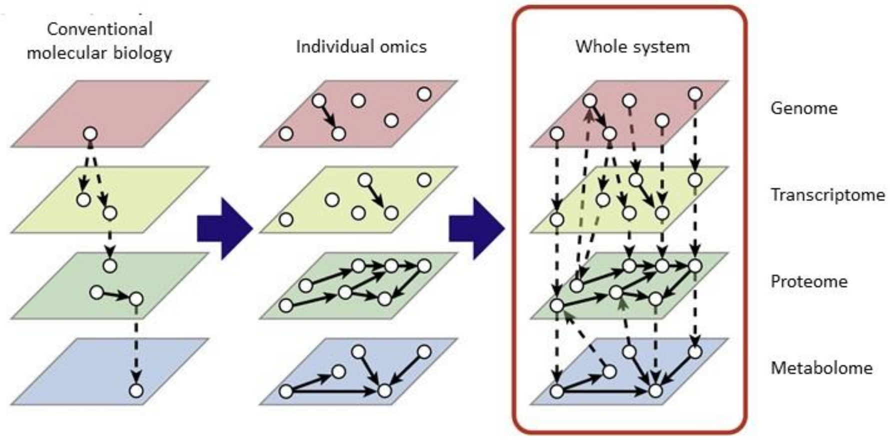


Figure 1.7 Studying a biological system as represented in multiple omics levels.
[Figure adapted from (Yugi et al. 2016)*]*

Due to the complexity involved in biological systems, there are multiple types of networks studied, such as metabolic networks, protein interaction networks and transcriptional regulatory networks (Figure 1.7). Metabolic networks describe multiple interconnected biochemical pathways, where the nodes represent metabolites and the edges connect metabolites that participate in the same reaction. Among the different networks, metabolic networks have the most well-established reconstruction procedures (Heinrich & Schuster 1998; Thiele & Palsson 2010) and have been studied in many organisms (Oberhardt et al. 2009). Metabolic network reconstruction is a time-consuming process that involves extensive manual curations partly assisted by automated reconstruction algorithms (Thiele & Palsson 2010; Buchel et al. 2013). In protein interaction networks, the nodes represent proteins and the edges represent interactions among them. Depending on the data used to generate the networks, the interactions can be direct physical interactions or indirect interactions as inferred from information such as protein co-occurrence (von Mering 2002). It is worth noting that most direct physical protein interactions detected in experiments may not be biologically relevant, as the proteins may only interact with each other under specific biological conditions (Hillenmeyer et al. 2008). Besides conducting experiments, protein interactions may be predicted computationally through various approaches such as physical modelling of protein binding (Deeds et al. 2006; Aloy & Russell 2006).

In contrast to both metabolic and protein interaction networks, transcriptional regulatory networks represent a relatively new development (Buchanan et al. 2010). In a transcriptional regulatory network, the nodes represent genes and the edges represent interactions among the genes. The genes considered are mostly transcription factor genes, but other elements such as non-coding RNA genes can also be considered. In a transcriptional regulatory network, these different regulators mediate the expression of one another, which lead to a cascade of expression changes that ultimately lead to changes in phenotypes. Note that transcriptional regulatory networks are abstracted, because the genes themselves can only interact with other genes through the RNAs and/or proteins they encode. While the abstraction makes the networks easier to understand, the reconstruction of transcriptional regulatory networks is difficult due to the need of considering multiple omics data. An ideal set of data for reconstructing transcriptional regulatory networks requires information from all omics level, including genomics, transcriptomics and proteomics data, to account for the entire transcriptional regulation process. However, obtaining such detailed data is often not possible, therefore most studies resort to using protein-DNA binding or transcriptomics data for network reconstruction.

Studies have found that transcriptional regulatory networks have a multi-layer hierarchical structure (Ma et al. 2004; Yu & Gerstein 2006). In this structure, global regulators at the top layers regulate many downstream regulators, while regulators at the bottom layers generally do not regulate upstream regulators. Both upstream and downstream regulators are controlled via external feedback mechanisms such as metabolite-protein interactions (Martínez-Antonio et al. 2006). In addition, transcriptional regulatory networks have been found to be highly interconnected and possess highly integrated network motifs connected by global regulators

(Guelzim et al. 2002; Dobrin et al. 2004; Resendis-Antonio et al. 2005). These structural properties are likely to play important roles for facilitating the rapid regulatory changes through the global regulators, as well as enabling fine tuning of the regulatory changes by downstream regulators.

Besides static topological information, the dynamic of the network plays an important role on the functional properties of a transcriptional regulatory network. Studies have shown that most regulatory interactions are condition-specific and vary significantly under different conditions (Segal et al. 2003; Luscombe et al. 2004). Due to these condition-specific changes of regulatory interactions, the same transcriptional regulators can be used under different conditions to achieve condition-specific response via regulating a different set of genes. In addition, the dynamic of the network is also facilitated by various network motifs, such as feedback loops and feedforward loops (Rosenfeld et al. 2002; Mangan et al. 2003). These network motifs have been shown to speed up or delay the propagation of regulatory signals along the network, in order to keep the network robust by filtering out noises while ensuring the signals are transmitted rapidly.

While these studies have elucidated the properties of transcriptional regulatory networks, the experiments were mostly performed in unicellular organisms, such as the bacteria *Escherichia coli* and the yeast *Saccharomyces cerevisiae*. It is likely that the insights are transferable to more complex organisms such as mice and humans, but they are likely to display more intricate controls of transcription that are specific to certain biological processes. Due to the difficulties in experimentally manipulating more complex organisms, the knowledge and data available on transcriptional regulations are less detailed than the simpler organisms. Recent advances have been made in studying transcriptional regulatory networks in specific biological processes in human, such as in cell cycle (Elkon et al. 2003), in blood cells (Zhu et al. 2010), in B cells (Basso et al. 2005), in brain tumours (Carro et al. 2010) and in glioblastoma (Sumazin et al. 2011). Interestingly, Basso et al. found that the transcriptional regulatory network in human B cells follows the hierarchical, scale-free network organisation as observed in simpler organisms. In addition, a study which investigated expression profiles from yeast, worms, flies and humans has found that many gene interactions are evolutionary conserved across organisms (Stuart et al. 2003).

1.3.3 Network reconstruction approaches

Biological networks can either be constructed through manual specification by curating literature or performing specific experiments, or through automatic prediction by running network inference algorithms on high-throughput data. The ultimate aim of network reconstruction is to accurately identify causal relationships among biological partners, and quantify the dynamics of such relationships. As the focus of this thesis is on understanding transcriptional regulation, the following section discusses network reconstruction methods related to transcriptional regulatory networks. Methods for manual specification of networks are simpler. The steps involve firstly using well-characterised networks from other species or biological systems to act as the starting points, and then extend the networks with additional information obtained by conducting experiments or curating literature (Faria et al. 2013).

Algorithm-based network reconstructions require some forms of omics data, which most commonly record the regulatory binding sites on genomic DNA or the gene expression levels. Regulatory binding sites data can be obtained from chromatin immunoprecipitation (ChIP) or DNase experiments combined with *in silico* regulatory binding predictions (Furey 2012), while gene expression data can be obtained from microarray or RNAseq experiments. While regulatory binding sites indicate where the proteins bind on genomic DNA, the distance-independent nature of such regulatory binding sites make the identification of downstream target genes difficult, and the identification of regulatory binding sites gave little indication of the dynamic of gene expression. The dynamic of transcriptional regulation can be more easily obtained by studying gene expression data, which also distinguish between the transcript isoforms expressed. The following section focuses on the discussion of network reconstruction using gene expression data.

Network inference algorithms for inferring transcriptional regulatory networks can be separated into two categories with different levels of granularity (Marbach et al. 2012). The first category predicts the presence or absence of gene interactions to give a static network; while the second category predicts the rate of gene interactions, given an underlying static network, to give a dynamic network. A static network describes only the topological information, while a dynamic network describes both topological and dynamic information. This means that a dynamic network can be simulated to generate *in silico* predictions that can be verified in experiments. Some established frameworks for working with dynamic networks include differential equation-based models (Davidson et al. 2002; Li & Wang 2013) and single-molecule simulation models (Drew 2001; Armbruster et al. 2009). However, such models rely on a higher number of parameters which are often difficult to obtain and verify without a large amount of data.

Besides the more detailed frameworks mentioned above, there are other commonly used major frameworks that require less data, namely correlation, mutual information, regression Bayesian networks and Boolean models. While correlation and mutual information-based methods generate static networks, the other three methods can give both static and dynamic networks depending on their implementations. Both correlation and mutual information-based methods are used to measure the association between the values of all pairs of variables. Some examples of popular algorithms include weighted gene correlation network analysis (WGCNA) (Langfelder & Horvath 2008) and CLR (Faith et al. 2007). The assumption is that genes with expression profiles that correlate to one another are likely to work in the same pathway. The networks inferred by both correlation and mutual information-based methods can be further refined by removing the effects of another variable through conditioning, which is done in generalisations such as partial correlation (Yuan et al. 2011) and conditional mutual information (Xiao et al. 2016).

Regression-based network inference methods can be viewed as an extension of correlation-based methods. An example of regression-based methods is TIGRESS (Haury et al. 2012). The main difference of regression-based methods is that they attempt to infer directed gene interactions instead of just quantifying their associations. Simple linear models can be used in regression-based networks, but the generalised forms are more commonly used for network inference due to non-linearity of gene interactions and high-dimensionality in the expression data. The generalisation to tackle non-linearity in regression-based methods involved extending a linear model into a generalised additive model that supports non-linear functions such as spline (Fan & Peng 2016). As for the high-dimensionality problem, regression-based methods often incorporate regularisations, such as Lasso, when estimating the regression coefficients (Haury et al. 2012). An alternative framework is the Bayesian network. In Bayesian networks, the network is a direct acyclic probabilistic graphical model where the nodes represent a set of random variables and the edges specify the conditional dependencies of the nodes (Friedman et al. 2000). An example of Bayesian network inference methods is ebdbNet (Rau 2016). Bayesian networks can be generalised into dynamic Bayesian networks, which support cyclic relationships among the nodes (Murphy & Mian 1999).

Boolean models represent a simple alternative to describe networks, in which the values of gene expressions are binary and the interactions among genes are described by Boolean logic. Some examples of Boolean model inference methods include REVEAL (Liang et al. 1998) and SCNS (Moignard et al. 2015). The simplifications mean that Boolean models represent one of the simplest dynamic networks that can be simulated. The properties of Boolean models depend hugely on the simulation update scheme used, which is most commonly either synchronous or asynchronous updates. Synchronous update scheme assumes changes in the expression of all genes happen simultaneously, which results in deterministic simulation. In contrast, asynchronous update scheme only allows one gene to update at each time step, which is closer to biological systems where different genes are expressed in different rates. Similar to other frameworks, Boolean models also have generalised forms such as probabilistic Boolean models and fuzzy logic models. In probabilistic Boolean models, each gene has several Boolean update functions, each of which has a probability of being chosen during simulation (Shmulevich et al. 2002; Liang & Han 2012). This probabilistic generalisation allows a better understanding of a stochastic system that can have multiple steady states. As the binary approximation may be too limiting, a Boolean model can be generalised into a fuzzy logic model which allows each variable to have multiple discrete levels of values (Schaub et al. 2007; Park et al. 2014).

In summary, these different network modelling frameworks are based on different assumptions, which lead to different pros and cons that make each modelling framework suitable in particular use cases. For example, correlation and mutual information-based approaches can be applied to biological systems without prior knowledge, and the algorithms are relatively scalable to accommodate a large number of genes. Signalling pathways can be easily modelled by Boolean models, as signals are assumed to be transmitted in binary form and with Boolean logic. These different network modelling frameworks complement one another, and the information encoded in them can eventually be combined to generate a consensus network. Consensus networks generated by combining the results of multiple methods have been shown to be superior than using results generated individually by each method (Marbach et al. 2012). In addition, these networks derived from different frameworks can be unified into a whole-cell model. As a proof of feasibility, a detail whole-cell simulation of *Mycoplasma genitalium* is created by combining multiple modelling frameworks, ranging from Boolean models to stochastic processes (Karr et al. 2012).

1.4 Thesis aims

While single-cell expression profiling techniques offer the advantage of increased data resolution, the data generated suffer from additional technical noise. Besides improving the experimental protocols to increase the efficiency of mRNA capture and to reduce amplification bias, new computational algorithms that can effectively account for the technical noise unique to single-cell RNAseq are still required, particularly for gene network inference. Besides developing new algorithms, existing algorithms should also be benchmarked against synthetic expression data or gold standard real expression data in order to assess their performance in an unbiased way. Assessing the performance of these algorithms are especially important, as most of the existing algorithms are either originally designed for non-single-cell expression data, or the characteristics of the algorithms are not properly investigated. Once the performance and the properties of these algorithms are understood, they can then be applied to single-cell RNAseq data collected from various biological systems to obtain novel insights.

This thesis aims to study transcriptional regulation by computational analyses of single-cell RNAseq data by following the objectives stated below:

1. To develop new network inference algorithms for transcriptional regulatory networks that are specifically designed for single-cell expression data. The frameworks used for new network inference algorithms development are Boolean model and regression.
2. To benchmark existing and newly developed algorithms that are critical for single-cell expression data analysis in an unbiased way by using synthetic data. The categories of algorithms to be investigated include pseudotime inference, gene network inference and differential expression analysis.
3. To utilise the investigated algorithms in understanding biological systems by analysing and studying single-cell expression data. The two biological systems studied are the epiblast stem cells reprogramming system and the acute myeloid leukaemia system.

2 Inferring gene regulatory networks with a Boolean model-based method

Sections of this chapter have been published during the course of this PhD (Lim et al. 2016).

2.1 Background

Boolean models are one of the simplest models that can describe the dynamics of a system without the need of many parameters [For reviews, see (de Jong 2002; Fisher & Henzinger 2007)]. In a Boolean model, each gene can take a value of 0 or 1, which represents the absence or presence of gene expression respectively. The interactions among genes in a Boolean model are described by Boolean operators like AND, OR and NOT, which closely resembles how biologists describe such interactions. Boolean models were first used to study gene regulatory networks by Kauffman in the 1970s, and since then have been used extensively to study different biological systems (Li et al. 2004; Fauré et al. 2006; Giacomantonio & Goodhill 2010; Dunn et al. 2014).

Single-cell expression data offer the advantage of capturing the expression profiles of many single cells, but the additional data resolution comes with the cost of increased technical noise, such as drop-outs. Therefore, network inference techniques that are robust to the effect of drop-outs are required when reconstructing networks using single-cell expression data. Among all network inference frameworks, Boolean models are implicitly robust to the presence of drop-outs. This is due to the binarisation of expression values in Boolean models by setting genes with high expression to 1 and genes with low expression to 0, while not trying to account for any intermediate expressions. Drop-outs are more likely to affect genes with low expressions, but genes with low expressions are often already binarised to 0.

In this chapter, a model learning algorithm BTR (BoolTraineR) that can reconstruct and train asynchronous Boolean models using single-cell expression data is described (Section 2.2). BTR differs from other algorithms described above in that it can infer both network structure and Boolean rules without needing information on trajectories through cell states. When inferring gene networks with BTR, there are 2 key steps. Firstly, BTR evaluates how well the predictions made by a Boolean model match with the single-cell expression data by using the Boolean state space (BSS) scoring function. BTR then iteratively modifies the Boolean model to generate a series of Boolean models that offer improving BSS scores through a swarming hill climbing strategy. At the end of the BTR optimisation process, the best scoring Boolean model represents the asynchronous Boolean model that can best explain a single-cell expression dataset. In Section 2.3, the BSS scoring function in BTR is shown to be a viable distance measures for Boolean models. Section 2.4 shows that BTR performed well in terms of network inference when compared with other established network inference algorithms. In Section 2.5, BTR predicted new gene interactions in blood cell development by training published Boolean models using independent single-cell expression data. The chapter then ends with conclusions in Section 2.6, and materials and methods in Section 2.7.

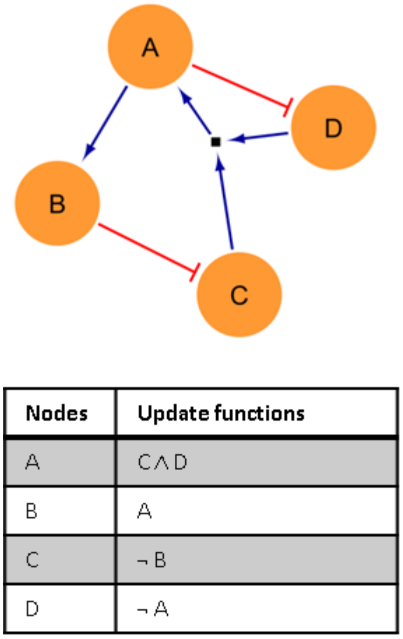
2.2 Framework of BTR

This section first explains the definitions of Boolean models, before exploring the concept and the framework underlying BTR.

2.2.1 Boolean models

A Boolean model *B* consists of n genes $x_1, \dots, x_n$ and n update functions $f_1, \dots, f_n: \{0,1\}^n \rightarrow \{0,1\}$, with each f_i being associated with gene x_i (Figure 2.1). Each gene x_i corresponds to a binary variable representing the expression value of the gene, i.e. $x \in \{0,1\}$. Gene x_i is a target gene when it acts as a response variable and an input gene when it acts as a predictor variable. Each update function f_i can be evaluated to give a value to a target gene x_i, and is expressed in terms of Boolean logic by specifying the relationships among a subset of the input genes $x_1, \dots, x_n$ using Boolean operators AND (∧), OR (∨) and NOT (¬). An update function f_i consists of an activation clause and an inhibition clause in the form of:

$$(activation\ clause\ ) \land \neg(inhibition\ clause)$$



Nodes	Update functions
A	C∧D
B	A
C	¬B
D	¬A

Figure 2.1 Representations of a Boolean model.
A Boolean model can be expressed graphically in terms of nodes and edges, as well as in tabular form in terms of update functions. Note that the small black node refers to AND interaction.

Each clause is individually expressed in disjunctive normal form, $(u_1) \vee (u_2) \vee (u_3) \vee \ldots \vee (u_n)$, where u represents a slot which can either take in a single input gene x_i or a conjunction of two input genes $x_i \wedge x_{i+1}$. An example update function $f_1(s_t)$ for a target gene x_1 with an input state s_t is given below:

$$x_1 = f_1(s_t) = ((x_3 \wedge x_4)) \wedge \neg((x_5) \vee (x_2 \wedge x_9))$$

A few constraints are imposed on the update functions during model learning in BTR. Firstly, the update function allows a conjunction of up to two input genes in each slot u. Secondly, each input gene x_i can only be present in a single update function once, but the same input gene x_i can be present in multiple update functions. Thirdly, a user can specify a soft limit on the number of input genes (i.e. in-degree) allowed per update function, where the default in BTR is 6 in-degree per gene. Lastly, by default no self-loop is allowed in BTR.

2.2.2 Boolean model states and simulations

A model state given by a Boolean model *B* is represented by a Boolean vector $s_t = \{x_{1t}, \ldots, x_{nt}\}$ at simulation step t. A model state space S represents the set of all model states s_t reachable from an initial model state s_1, i.e. $S = \{s_1, \ldots, s_t\}$. S can be obtained by simulating the model *B* starting from an initial model state s_1 using the asynchronous update scheme. The asynchronous update scheme specifies that at most one gene is updated between two consecutive states (Figure 2.2). If a state has already been encountered earlier, it is ignored. This results in a directed graph of states as exemplified in Figure 2.2, where any two connected states change in just one variable. Asynchronous updating is critical when modelling developmental systems that generate distinct differentiated cell types from a common progenitor, because synchronous updating generates fully deterministic models and therefore cannot capture the ability of a stem cell to mature into multiple different tissue cells.

Assuming we have a model state s_t which is not a steady state, there will be i $(i \geq 1)$ genes in s_t such that $x_{it} \neq f_i(s_t)$. Therefore at simulation step $t+1$, s_{t+1} would have i possible configurations s_{t+1}^i, where $s_{t+1}^i = \{x_{1t}, \ldots, f_i(s_t), \ldots, x_{nt}\}$. This simulation is repeated until it reaches a steady state. By definition, steady states are a set of states whose destination states also belong to the same set. That is, a steady state may be a single model state s_t, or it may consist of a cyclic sequence of model states $s_t, \ldots, s_{t+j}$. The initial state used in a simulation can be obtained from the expression values at time = 0 for a time-series expression dataset, or it can be obtained from the expression values of known parental cell types.

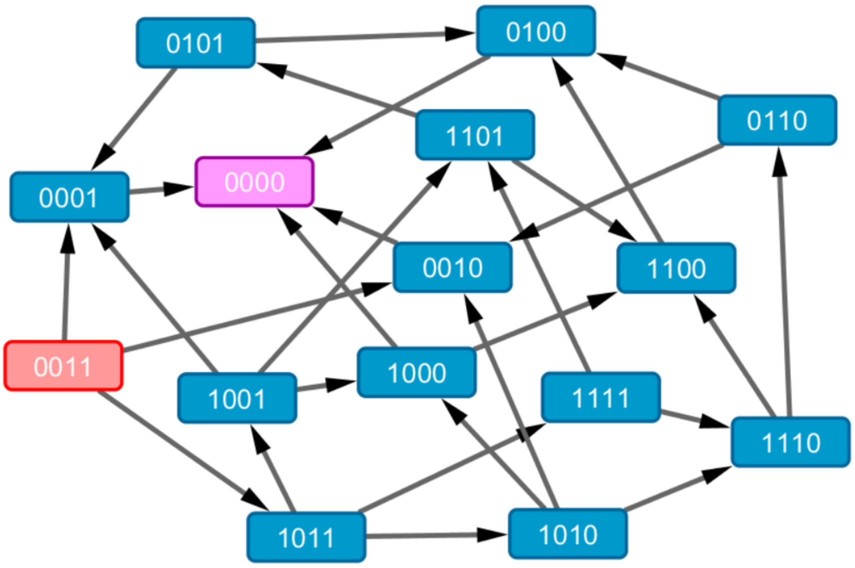


Figure 2.2 Asynchronous simulation of the Boolean model specified in Figure 2.1. *The asynchronous update scheme is best explained with the use of a graph representation of state space, in which each connected state differs in only one node. Starting from the initial state s_1={0,0,1,1} and evaluated using the update functions in (A), asynchronous simulation produces a model state space with 15 states. The initial state is shown in red node, while the final steady state is shown in pink node.*

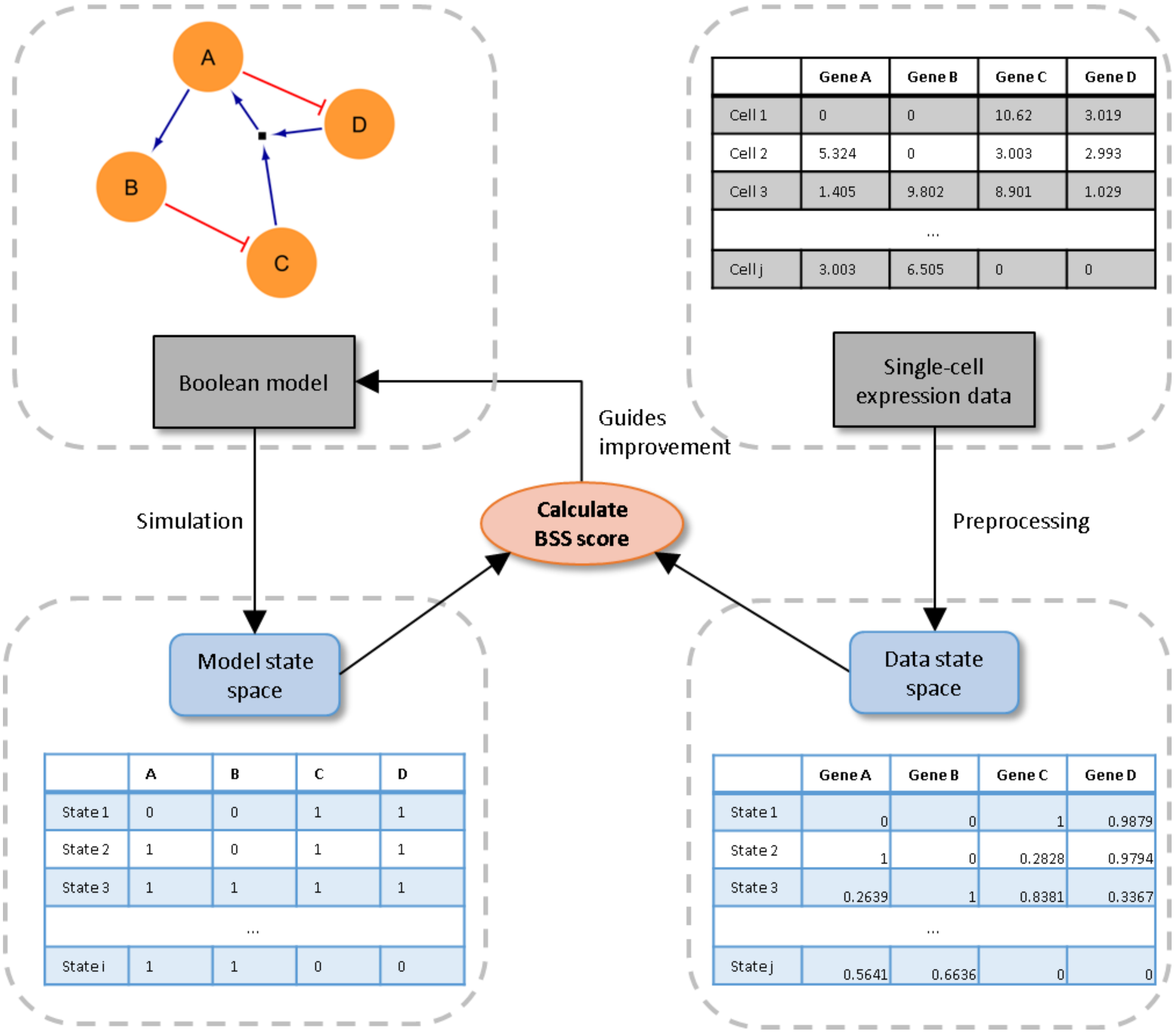
2.2.3 Single-cell expression data

The single-cell expression data used in this study are each a matrix consisting of n individual genes in the columns and k individual cells in the rows. The expression data are normalised and standardised to give $y_{kn} \in [0, 1]$. A data state $v_k = \{y_1, \dots, y_n\}$ represents the expression state of cell k for n genes that are observed in the cell. A data state space $V = \{v_1, \dots, v_k\}$ represents the set of all data states that are observed in an experiment.

2.2.4 General concept of BTR

The model state space of an asynchronous Boolean model resembles the data state space of a single-cell expression data. The model state space contains predicted expression states that are dictated by a known gene network that underlies a Boolean model; while the single-cell expression data can be viewed as a data state space which contains observed expression states that are dictated by an unknown gene network. By fine-tuning the network rules underlying the Boolean model, it should be possible to produce a predicted model state space that closely resembles an observed data state space, thereby allowing us to reconstruct the unknown gene network. BTR uses this framework to reconstruct a Boolean model from single-cell expression data (Figure 2.3).

By utilising the Boolean state space (BSS) scoring function (See Section 2.2.5.1), BTR evaluates how well a particular Boolean model explains the single-cell expression data by scoring the model state space with respect to the data state space. During the model training process, BTR uses a swarming hill climbing strategy (See Section 2.2.5.2) to generate minimally modified Boolean models based on an initial Boolean model. These minimally modified Boolean models are then scored using the BSS scoring function, and BTR selects the best scoring Boolean models for the next iteration. By performing this process iteratively, BTR reconstructs the asynchronous Boolean model that can best explain a single-cell expression dataset.



	Gene A	Gene B	Gene C	Gene D
Cell 1	0	0	10.62	3.019
Cell 2	5.324	0	3.003	2.993
Cell 3	1.405	9.802	8.901	1.029
		...		
Cell j	3.003	6.505	0	0

	A	B	C	D
State 1	0	0	1	1
State 2	1	0	1	1
State 3	1	1	1	1
		...		
State i	1	1	0	0

	Gene A	Gene B	Gene C	Gene D
State 1	0	0	1	0.9879
State 2	1	0	0.2828	0.9794
State 3	0.2639	1	0.8381	0.3367
		...		
State j	0.5641	0.6636	0	0

Figure 2.3 The framework underlying BTR.
A Boolean model can be simulated to give a model state space, while a single-cell expression data can be preprocessed to give a data state space. Boolean state space scoring function can then calculate the distance score between the model and data state spaces. Lastly, BTR uses the computed distance score to guide the improvement of the Boolean model through an optimisation process that minimises the distance between model and data state spaces.

2.2.5 BTR model learning algorithm

The aim of BTR is to identify a Boolean model B with x_n genes and f_n update functions, that can produce a model state space which closely resembles an independent single-cell expression data (i.e. data state space). Note that model state space and data state space are defined in a similar way, the only difference being that the n genes take continuous values in $[0,1]$ within a data state, while the n genes take binary values 0 and 1 in a model state. The distance between model and data state spaces is measured by the pairwise distance between pairs of model and data states, as stated in the scoring function (See below). By iteratively modifying an initial Boolean model B_1, the distance between the model and data state spaces can be minimised until a resulting final Boolean model $B_f$with less distance is obtained. BTR performs model learning by utilising techniques in discrete optimisation framework. In any optimisation problem, there are two important components, namely a scoring function and a search strategy.

2.2.5.1 BSS scoring function in BTR

The scoring function used in BTR is a novel scoring function, termed as Boolean state space (BSS) scoring function. The BSS scoring function $g(S,V)$ is a distance function, where S is the model state space and V is the data state space. $g(S,V)$ consists of a base distance variable and two penalty variables, and is given by:

$$g(S,V) = h(S,V) + \lambda_1\varepsilon_1 + \lambda_2\varepsilon_2$$

Where $h(S,V)$ = base distance, ε = penalty variable, λ = constant for penalty variable.

The base distance $h(S,V)$ is given by the following equation. To prevent multiple model states from matching to a single data state, one-to-one matching between model and data states is enforced if the number of data states, N_v, are more than or equal to the number of model states, N_s, i.e. $N_v \geq N_s$. For cases where $N_v < N_s$, one-to-one matching between model and data states is enforced greedily up until the point where all data states have been assigned a matching model state, then non-unique matching will occur for the remaining model states with respect to each corresponding data state with the minimum distance.

$$h(S,V) = \frac{\sum_{t=1}^{N_s} min_{k=1}^{Nv}(d(s_t,v_k))}{N_s\, n}$$

Where $d(s_t,v_k)$ = pairwise distance between each model state s_t and data state v_k ($0 \leq d(s_t,v_k) \leq 1$), N_s = number of model states, N_v = number of data states, n = number of genes.

The distance between model state s_t and data state v_k, $d(s_t,v_k)$, is defined as the sum of the absolute differences between values of each gene i in model state s_t and data state v_k.

$$d(s_t,v_k) = \sum_{i=1}^{n} | x_{ti} - y_{ki}|$$

Where $x_{ti} \in \{0,1\}$ is the value of gene i in model state s_t and $y_{ki} \in [0,1]$ is the value of gene i in data state v_k.

The two penalty variables, ε_1 and ε_2, in $g(S,V)$ are used to prevent underfitting and overfitting. ε_1 penalises the proportions of 0s, p_0, and 1s, p_1, across all genes and all states in a model state space. The concept of ε_1 is that it penalises complexity in Boolean models by their simulated model state spaces. As a Boolean model becomes more complex (i.e. increase in the number of edges), both p_0 and p_1 of its model state space will become closer to 0.5 (Figure 2.4), therefore making ε_1 a good penalty for model complexity.

$$\varepsilon_1 = e^{-a}, \qquad where\ a = \sum_{i \in \{0,1\}} \frac{(p_i - 0.5)^2}{0.5}$$

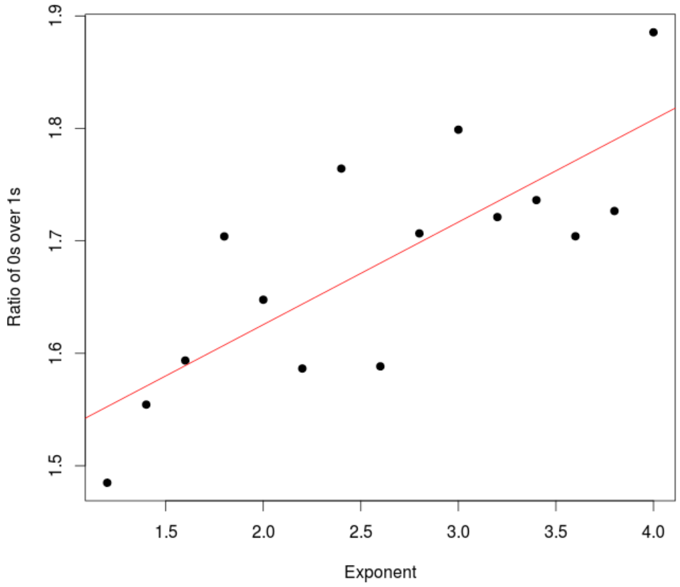


Figure 2.4 The ratios of 0s over 1s (p_0/p_1) in model state spaces plotted against the exponents of a power-law distribution.
A power-law distribution is used to model the number of in-degree and out-degree of each node. The smaller the exponent in a power-law distribution, the higher the number of in-degree and out-degree of each node. The red line is a linear model fitted to illustrate the general relationship between ratio of 0s over 1s and the exponent. As the proportion of 0s and 1s become more similar (i.e. approach 0.5), the ratio of 0s over 1s will be closer to 1.

ε_2 penalises based on the number of input genes present in each of the update function f_i in a Boolean model B, given a specified threshold z_{max}.

$$\varepsilon_2 = \sum_{i=1}^{n} w_i$$

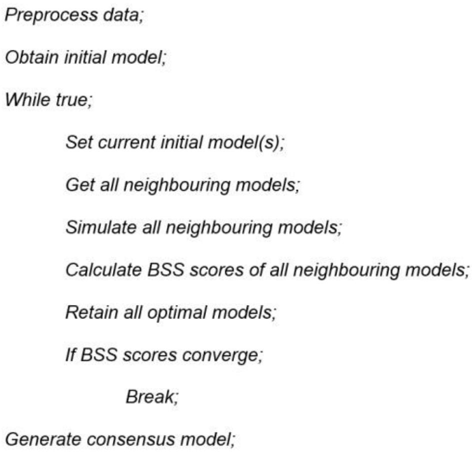
Where w_i the penalty for each update function f_i is given by:

$$w_i = \begin{cases} \dfrac{z_i - z_{max}}{n}, & if\ z_i > z_{max} \\ 0, & if\ z_i \leq z_{max} \end{cases}$$

Where z_i = the number of input genes in update function f_i, z_{max} = the maximum number of input genes allowed per update function. The default z_{max} in BTR is 6, which means that each target gene is encouraged to have not more than 6 input genes.

2.2.5.2 Search strategy in BTR

A good search strategy is required in optimisation to locate the optimal solutions within a high dimensional and complex solution space. The search strategy in BTR is a form of swarming hill climbing strategy, in which multiple optimal solutions are kept at each search step and the search only ends when the score converges for all the optimal solutions (Figure 2.5). In BTR search algorithm, the search starts from an initial Boolean model, and iteratively explores the neighbourhood of the current Boolean model in the solution space by minimal modification. When no initial model is given to BTR, it will generate a random initial model whose degree distribution satisfies a power-law distribution with a degree exponent $\gamma = 3$.



```
Preprocess data;
Obtain initial model;
While true;
    Set current initial model(s);
    Get all neighbouring models;
    Simulate all neighbouring models;
    Calculate BSS scores of all neighbouring models;
    Retain all optimal models;
    If BSS scores converge;
        Break;
Generate consensus model;
```

Figure 2.5 Pseudocode of the search algorithm in BTR.

The minimal modification of a Boolean model is performed by adding or removing a gene from a single update function in the Boolean model. The resulting modified model is then evaluated by the BSS scoring function. By repeating this procedure, BTR is able to explore the solution space and eventually arrives at a more optimal Boolean model. Due to the nature of Boolean models that multiple possible Boolean models can give rise to the exact same simulated state space, BTR usually retains a list of equally optimal Boolean models at the end of the search process. In such cases, a consensus model, whose edges are weighted according to the frequencies of their presence in the list of optimal Boolean models, will be generated. Due to the design of the search strategy, it is more geared towards a local search rather than a global search. Therefore in line with the results shown in Figure 2.9, BTR is best used for iteratively improving a gene network with known biological knowledge using an independent set of single-cell expression data.

2.3 Describing Boolean state space scoring function as a model distance measure for Boolean models

How well BTR performs depends heavily on the performance of the BSS scoring function. Among different modelling frameworks, the Bayesian network framework is known to possess several well-established scoring functions that evaluate how well a particular network fits a given dataset. These scoring functions include log-likelihood, Bayesian information criterion (BIC), Bayesian Dirichlet and K2 [See (Liu et al. 2012; Carvalho 2009) for reviews]. Since expression data have continuous values for gene expressions, the BIC scoring function, which can handle continuous variables, was selected as a scoring function from the Bayesian network framework for comparison purpose.

BSS and BIC scoring functions were evaluated using synthetic data. The true network and expression data in the synthetic data were generated using GeneNetWeaver (Schaffter et al. 2011), which is also used in the DREAM5 network inference challenge (Marbach et al. 2012). In order to simulate the zero-inflated property of single-cell expression data due to the presence of drop-outs, zero inflation was introduced into the synthetic data as described in the Methods section. An ideal scoring function should give an increasing distance score, as the evaluated network becomes increasingly different from the true network. In order to test this, a list of modified networks that are increasingly different from the true network in terms of edges was generated. As Bayesian networks and Boolean frameworks imposed different network structure constraints, the modified networks were generated separately to give a list of modified Bayesian networks and another list of modified Boolean networks. Although the modified Bayesian and Boolean networks are not identical, they possess the same number of differing edges when compared to the true network, ranging from 2 edges up to 40 differing edges. Five independent benchmark data, each with a different true network, true data and modified models, were used in the evaluation of scoring functions.

By evaluating networks using zero-inflated synthetic data, both BSS and BIC scoring functions performed well when acyclic networks are considered (Figure 2.6). Both scoring functions were able to give increasing distance scores as the underlying networks become increasingly different from the true network. The BSS scoring function achieves this by considering the input expression data as a data state space, and then computing the distance score by comparing the data state space with the model state space simulated from a given network. It is expected that as a network becomes increasingly different, its model state space will become increasingly different from the data state space, which is reflected in the distance score as shown in Figure 2.6C. The BSS scoring function, which is based entirely on the Boolean modelling framework, has been demonstrated to give comparable performance with a scoring function for Bayesian networks. The fluctuations in the scores computed by the BSS scoring function are due to the fact that small modications to the structure of a Boolean model may result in a drastic change in its simulated model state space in certain cases. The exact cause of this phenomenon is unknown, but it is likely to be due to the presence of particular motifs in the gene network that underlies the Boolean model.

As indicated in the results for Network 2 (Figure 2.6C), the BSS scoring function is dependent on the underlying true network structure in certain cases and will work better on distinguishing networks that are very different. However the BSS scoring function has a distinct advantage over scoring functions for Bayesian networks. The Bayesian networks are known to impose relatively strict constraints on permissible network structures, in particular Bayesian networks are not allowed to contain any cyclic network structure. Therefore scoring functions for Bayesian networks cannot be used to evaluate cyclic networks. Cyclic networks are ubiquitous in biological systems, in which cyclic motifs can be present in the form of negative and positive feedback loops. Boolean models on the other hand are allowed to have any number of cyclic motifs in the networks. Therefore, the BSS scoring function can be used to compute scores for cyclic networks. By using another five independent benchmark data with true networks that contain at least one cycle, the distance scores for modified networks were computed (Figure 2.7). The distance scores for cyclic networks have more fluctuations compared to acyclic networks due to the presence of cyclic motifs. However, the general trend where the distance scores increase as the underlying networks become increasingly different from the true network was still observed.

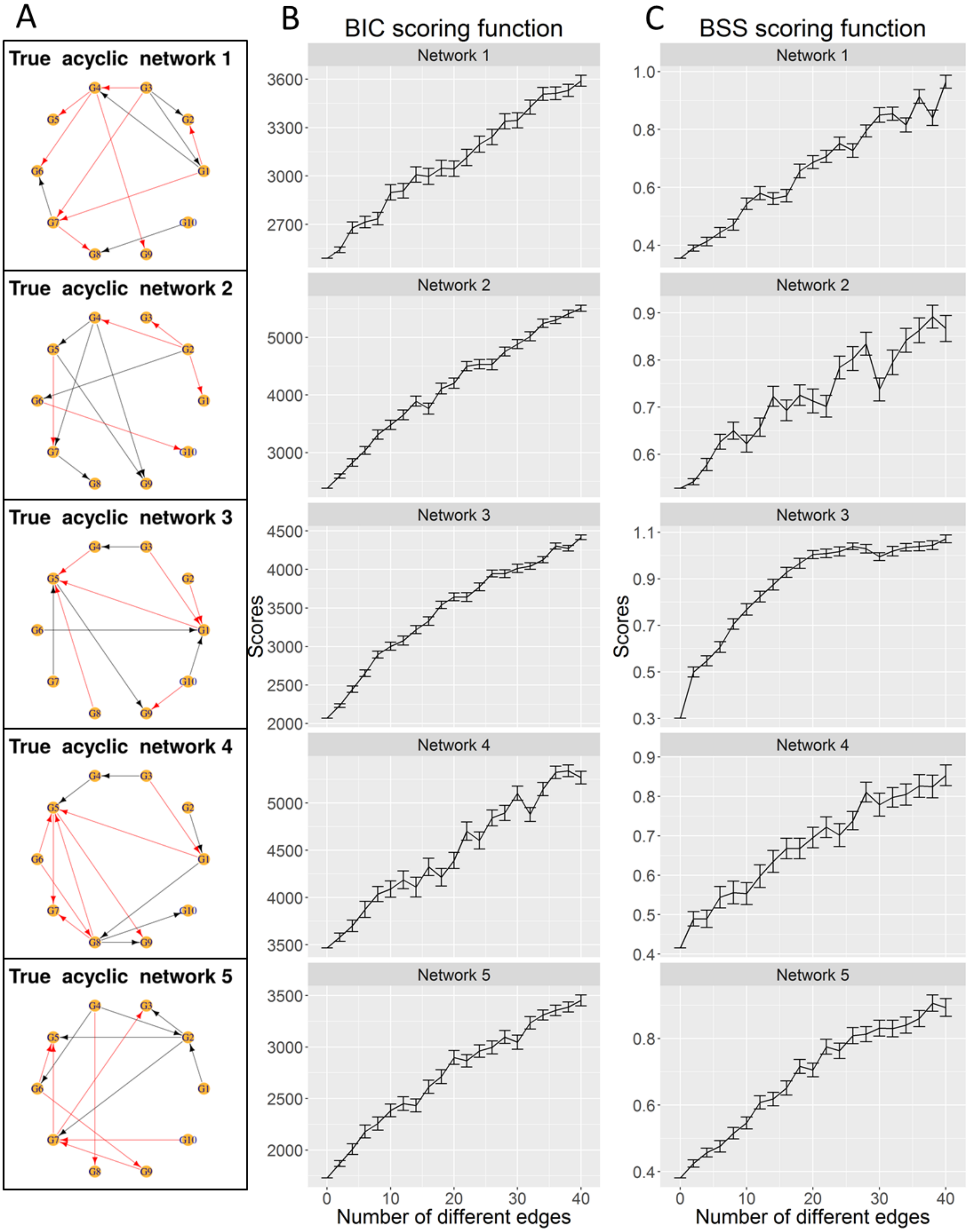


Figure 2.6 BSS scoring function compares favourably with BIC scoring function on acyclic networks using zero-inflated synthetic expression data.
(A) True acyclic networks. Each node corresponds to a gene. Black edges indicate activation interactions, while red edges indicate inhibition interactions. Mean distance scores computed using (B) BIC scoring function and (C) BSS scoring function. The error bar is the standard error of the mean.

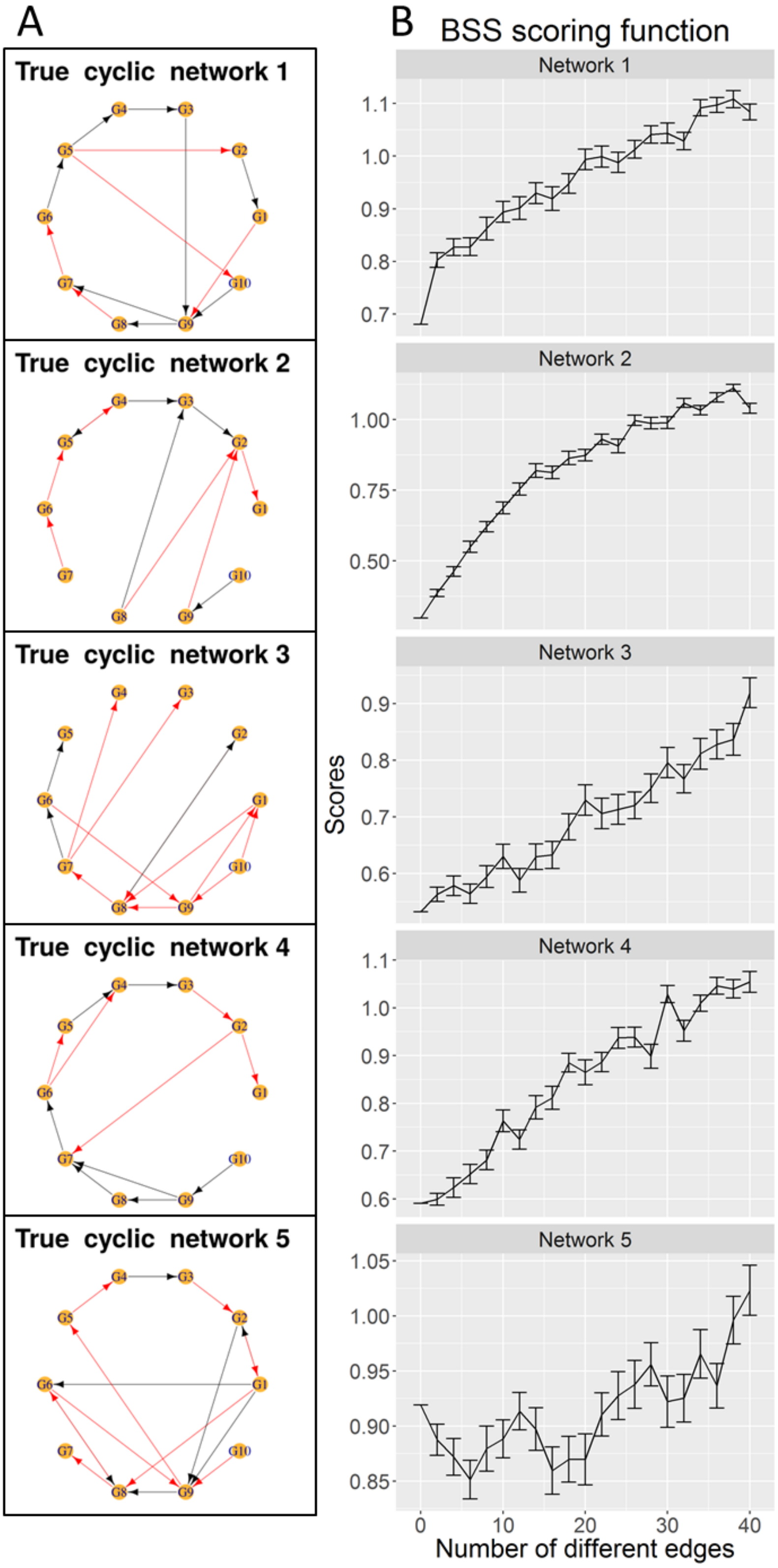


Figure 2.7 BSS scoring function is able to calculate distance scores for cyclic networks using zero-inflated synthetic expression data.
(A) True cyclic networks. Each node corresponds to a gene. Black edges indicate activation interactions, while red edges indicate inhibition interactions. (B) Mean distance scores computed using BSS scoring function. The error bar is the standard error of the mean.

The series of acyclic and cyclic networks were also investigated using non zero-inflated data. When the results computed with non zero-inflated data are compared to the results computed using zero-inflated data, we can see that zero-inflation has no effect on BIC scores and a small effect on BSS scores that does not affect the general trend (Figure 2.8A). In summary, the relative mean scores that average across the results of all networks (Figure 2.8B) show that although the BIC scoring function performs slightly better than the BSS scoring function, the BSS scoring function has the advantage that it can evaluate cyclic networks.

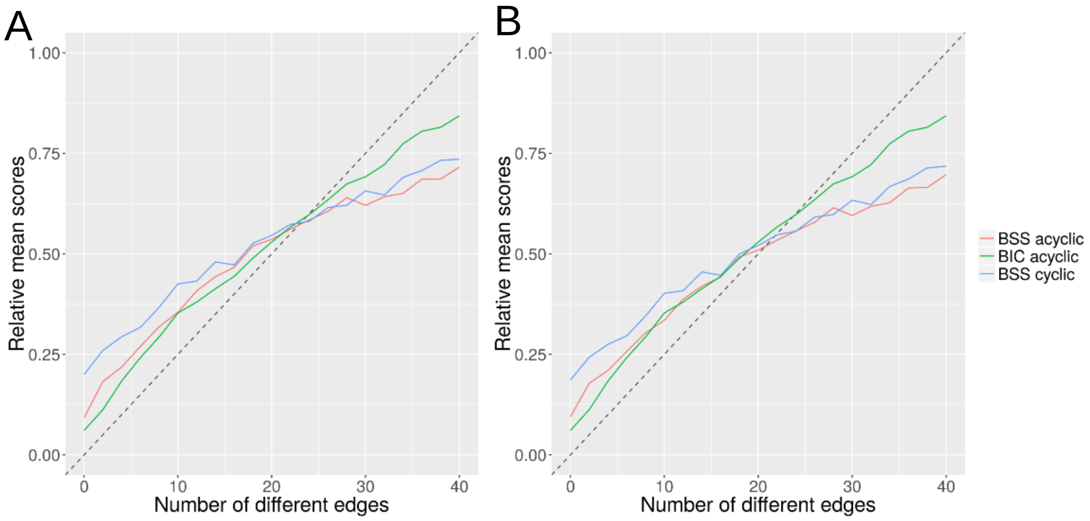


Figure 2.8   Summary of BIC and BSS scoring functions.
(A) Non zero-inflated synthetic expression data, (B) Zero-inflated synthetic expression data. Mean scores have been calculated across all networks (five acyclic networks and five cyclic networks) for BIC and BSS scoring functions calculated using zero-inflated synthetic expression data. All scores have been standardised for comparison purpose, such that the scores range from 0 to 1.

2.4 Assessing network inference performance of BTR using synthetic data

Next, the network inference performance of BTR was compared with other well-known network inference algorithms. Two search algorithms guided by the BSS Boolean and BIC Bayesian network scoring functions were included in the comparison, indicated as BTR and BIC respectively. The search algorithms used for both scoring functions are based on hill climbing. The additional network inference algorithms included in the comparison are BestFit (Lähdesmäki et al. 2003), ARACNE (Margolin et al. 2006), CLR (Faith et al. 2007), bc3net (de Matos Simoes & Emmert-Streib 2012), GeneNet (Opgen-Rhein & Strimmer 2007) and Genie3 (Huynh-Thu et al. 2010) (See Section 2.7.5 for brief details on the algorithms).

By using the same synthetic networks, as well as both non zero-inflated and zero-inflated synthetic data, network inferences were performed using the synthetic expression data alone without any extra information. In contrast to the DREAM5 challenge (Marbach et al. 2012) which also provides perturbed expression data, only a single type of expression data is provided to all the network inference algorithms, which is the wild type time course expression data in steady state. For BTR, besides performing inference with only expression data (indicated as BTR-WO), network inferences were also performed with both expression data and initial networks (indicated as BTR-WI) to show that BTR is able to use initial networks with partially known network structure to improve the inference process. The initial networks are generated randomly to contain 18 edges that are different compared with the true networks. The performance of the network inference algorithms is assessed in terms of F-scores (Sokolova et al. 2006) (Figure 2.9). In order to allow comparisons on the performance across all network inference algorithms tested, F-scores were calculated based only on the presence or absence of edges, while ignoring any additional information such as the types of edges.

In terms of acyclic networks, the results show that the top inference algorithms using either non zero-inflated or zero-inflated data are BTR-WI, CLR, BIC and BTR-WO. As for cyclic networks, the top inference algorithms differ between using non zero-inflated and zero-inflated data. BTR-WI, BTR-WO, CLR and BC3NET gave the best performance with non zero-inflated data, while BTR-WI, ARACNE, GENIE3 and CLR gave the best performance with zero-inflated data. When all results are taken together, BTR-WI, CLR, BTR-WO and GENIE3 gave the best performance overall. Note that the ranking of network inference algorithms in this study differs from the ranking of the DREAM study because different scoring criteria are used (F-score is used here as opposed to the area under the precision-recall (AUPR) and receiver operating characteristic (AUROC) curves in the DREAM study); and the DREAM study was done using multiple types of synthetic data, such as expression data with gene perturbations. In general, the presence of drop-outs affects the performance of network inference algorithms in different ways (Figure 2.9B). In cases such as bc3net and GeneNet, their performance decreases when drop-outs are present, while the impact of drop-outs on the performance of BTR is minimal. Interestingly, the performance of BestFit increases with the presence of drop-outs, possibly due to better binarisation of data due to the information given by drop-outs. As both BTR and BestFit are algorithms for inferring Boolean model, this result provides further support that Boolean models are robust to the presence of drop-outs in single-cell expression data.

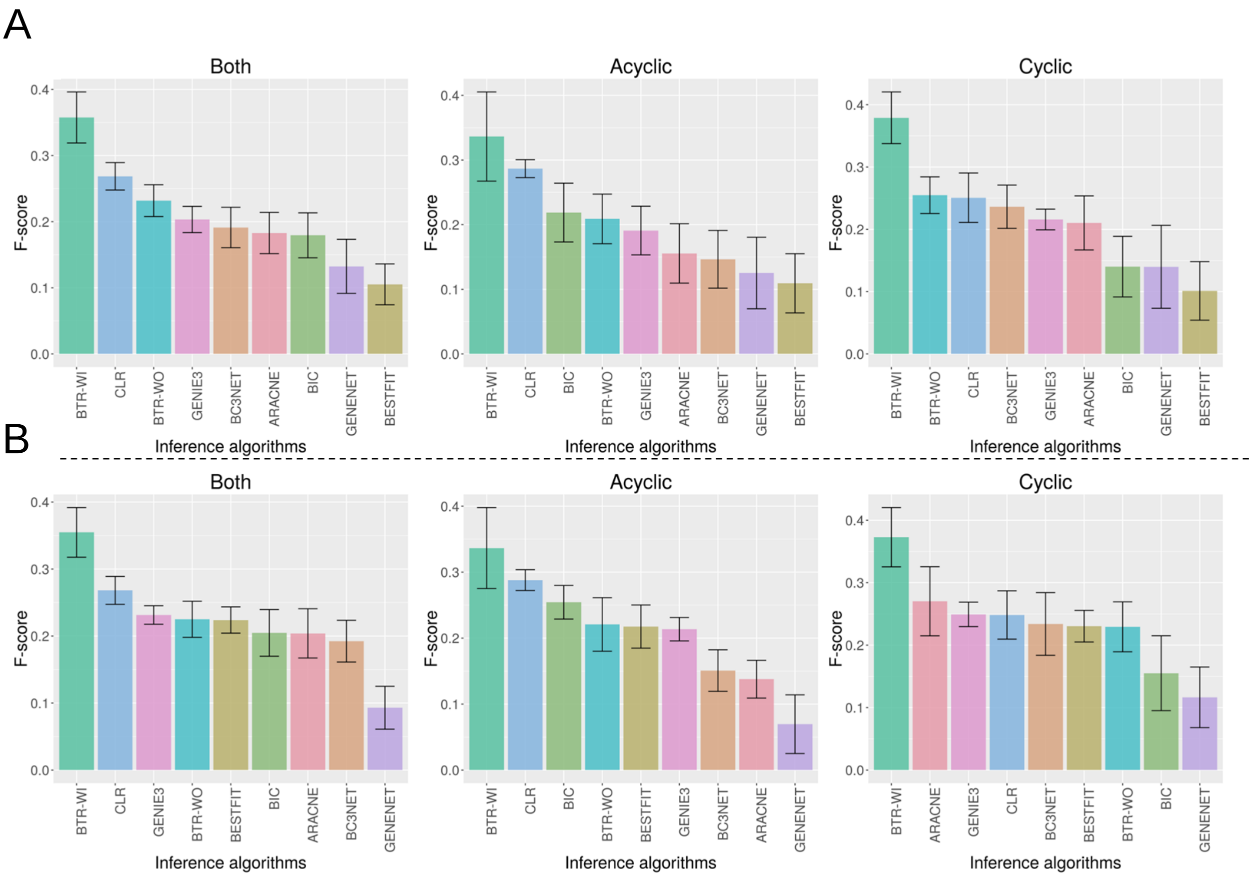


Figure 2.9 BTR outperforms other network inference algorithms.
Mean F-scores of network inference algorithms inferred using (A) non zero-inflated synthetic data and (B) zero-inflated synthetic data. Plots titled 'Both' show the combined results of acyclic and cyclic network inference. The error bar is the standard error of the mean.

When given an initial network as in BTR-WI, the BTR algorithm was able to perform very well in locating the true network. While the performance of the BTR algorithm without an initial network (BTR-WO) is comparable with other inference algorithms, BTR-WO scored less well compared to BTR-WI. This indicates that the greedy hill climbing search strategy implemented in BTR may not be able to traverse the solution space efficiently without any initial information. Taken together, while BTR can be used for reconstructing network models without initial information, BTR performed the best when it is used to train and improve on existing networks that contain a partially true structure. It is also worth noting that BTR produced a dynamic model with a directed underlying static network, in contrast to most other algorithms such as CLR that only produce an undirected static network.

2.5 Utilising BTR to train haematopoietic Boolean models with single-cell expression data

BTR was then applied to biological data to evaluate its performance on real data and to gain new biological insights. Haematopoiesis research has provided many paradigms for modern biological research, and was one of the first fields to embrace single cell expression profiling (Ramos et al. 2006; Pina et al. 2012; Moignard et al. 2013). Moreover, literature curated Boolean network models have been reported both for blood stem cell maintenance and blood progenitor differentiation (Bonzanni et al. 2013; Krumsiek et al. 2011). The single-cell expression data used here includes single-cell qPCR and single-cell RNA-Seq data, which are both obtained from (Wilson et al. 2015). The two Boolean models will be referred to as the Bonzanni model (Bonzanni et al. 2013) (Figure 2.10A) and the Krumsiek model (Krumsiek et al. 2011) (Figure 2.10C). Both models had been constructed via manual literature curation by the authors of the original papers. The Bonzanni model aims to capture haematopoietic stem cell (HSC) self-renewal capacity, while the Krumsiek model describes the differentiation process of the erythro-myeloid lineage in haematopoiesis.

The Bonzanni model was firstly trained using single-cell RNA-Seq data collected from HSCs. Compared to the original model, the resulting trained Bonzanni model (Figure 2.10B & Figure 2.11A) shows the deletions of 10 gene interactions and the additions of 13 gene interactions. The state space of the trained Bonzanni model contains 1486 states when simulated using the initial state used in the original study (Figure 2.12A). Of note, there are many densely connected transitional states in the state space, which may be related to the complexity of cell fate decision making processes in multipotent progenitor cells. Steady state analysis performed showed that the steady states of the trained Bonzanni model are almost identical to the steady states of the original Bonzanni model (Figure 2.13A), except with the absence of cyclic steady states. The authors suggested that the cyclic steady states in the original Bonzanni model correspond to the self-renewal maintenance loop in HSCs, which is not present in our trained model possibly because the number of cells profiled by single-cell RNA-seq is not enough to sufficiently capture the HSC self-renewal expression signature.

Next the Krumsiek model was trained by using single-cell qPCR data collected from over 450 cells along the erythro-myeloid lineage, which includes common myeloid progenitors, granulocyte-monocyte progenitors and myeloid-erythroid progenitors. In order to demonstrate that BTR can be used in cases where we may want to extend a current Boolean model by adding more genes to it, BTR was used to train and add two additional genes to the Krumsiek model. The resulting trained Krumsiek model (Figure 2.10D & Figure 2.11B) contains 3 deleted gene interactions and 12 added gene interactions when compared to the original Krumsiek model. For the two additional genes *Ldb1* and *Lmo2*, BTR has predicted gene interactions among *Ldb1*, *Lmo2*, *Fli1, Gata1* and *Gata2*. Previous studies have shown that genome-wide binding profiles for *Lmo2*, *Gata2* and *Fli1* show significant overlaps (Wilson et al. 2010), and that *Ldb1* also occupies nearly all of the binding sites of *Gata2* (Li et al. 2011), consistent with a model where these TFs engage in combinatorial interactions. The state space of the trained Krumsiek model contains 21 states when simulated using the initial state used in the original study (Figure 2.12B). The two steady states reachable in this state space may correspond well to cell populations that are primed for the erythrocyte and myeloid lineage divergence. When examining the steady states reachable from all possible initial states, the trained Krumsiek model produces additional steady states when compared with the original model due to the addition of two extra genes (Figure 2.13B), which may correspond to intermediate cell types along the erythro-myeloid differentiation pathway.

Taken together, the result suggests that both the trained Bonzanni and Krumsiek models have been trained by BTR to predict new gene interactions which give rise to interesting state spaces and steady state properties. Note that the state space of the trained Bonzanni model is substantially larger than the state space of the trained Krumsiek model due to the denser interactions among genes and a lower proportion of inhibitory edges in the trained Bonzanni model.

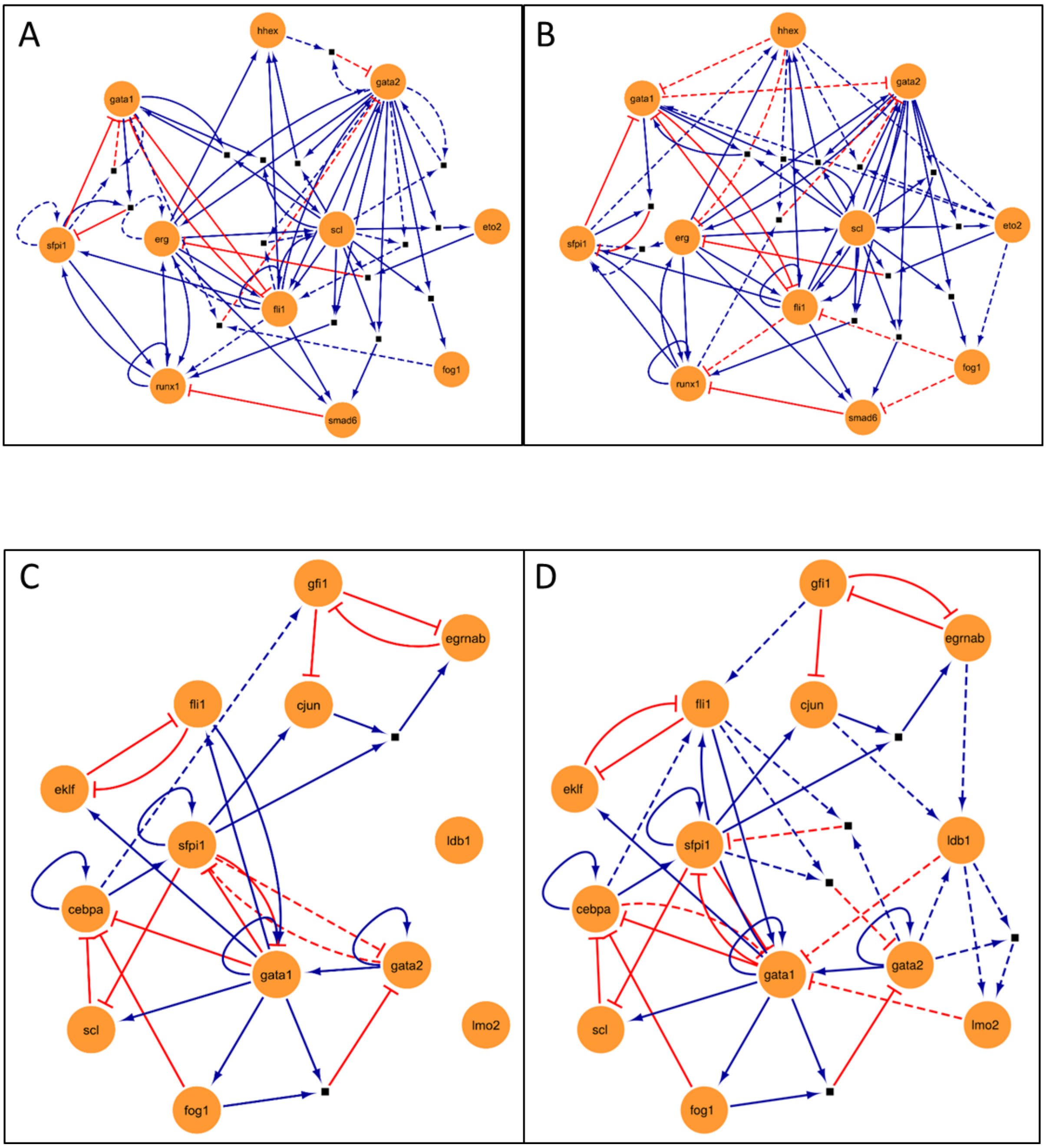
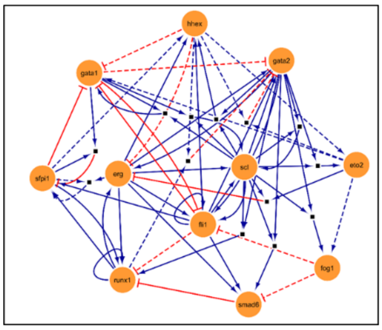


Figure 2.10 BTR predicts gene interactions by training the Bonzanni and Krumsiek Boolean models.

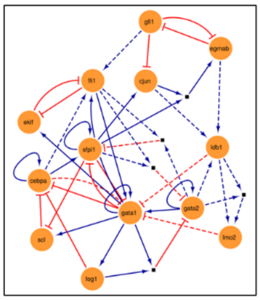
(A) Original Bonzanni model. (B) Trained Bonzanni model. (C) Original Krumsiek model. (D) Trained Krumsiek model. Round orange nodes indicate genes, square black nodes indicate AND gates that combine the two input gene interactions. Blue edges indicate activation interactions, red edges indicate inhibition interactions. Dashed lines in the original models indicate edges that are present in the original models, but are removed in the trained models. Dashed lines in the trained models indicate edges that are added to the trained models and are not present in the original models.

A Trained Bonzanni model



Target genes	Update functions
Erg	*(Runx1 ∨Fli1 ∨Gata2) ∧¬ ((Scl ∧ Eto2) ∨Hhex)*
Eto2	*((Gata2 ∧ Scl) ∨Hhex)*
Fli1	*(Fli1 ∨Erg ∨Scl∨Gata2) ∧¬ (Gata1 ∨Fog1)*
Fog1	*((Scl ∧ Gata2) ∨Eto2)*
Gata1	*(Eto2 ∨(Scl ∧ Gata1) ∧¬ (Sfpi1 ∨Fli1 ∨Hhex)*
Gata2	*(Scl ∨Fli1 ∨Erg) ∧¬ (Gata1 ∨(Hhex ∧ Eto2) ∨(Hhex ∧ Runx1))*
Hhex	*(Erg ∨Fli1 ∨ Sfpi1 ∨ (Scl ∧Gata2))*
Runx1	*(Runx1 ∨Sfpi1 ∨(Gata2 ∧ Scl) ∨ Erg) ∧¬ (Fli1 ∨Smad6)*
Scl	*((Gata1 ∧ Scl) ∨(Scl ∧ Gata2) ∨Fli1 ∨Erg)*
Sfpi1	*((Erg ∧ Sfpi1) ∨Runx1 ∨Fli1) ∧¬ ((Gata1 ∧ Sfpi1))*
Smad6	*((Gata2 ∧ Scl) ∨Fli1 ∨Erg) ∧¬ (Fog1)*

B Trained Krumsiek model



Target genes	Update functions
C/EBPα	*(C/EBPα) ∧¬ (Gata1 ∨Fog1 ∨Scl)*
cJun	*(Sfpi1) ∧¬ (Gfi1)*
EgrNab	*((Sfpi1 ∧ cJun)) ∧¬ (Gfi1)*
EKLF	*(Gata1) ∧¬ (Fli1)*
Fli1	*(C/EBPα ∨Gata1 ∨Gfi1) ∧¬ (EKLF)*
Fog1	*(Gata1)*
Gata1	*(Gata1 ∨Gata2 ∨Fli1) ∧¬ (Sfpi1 ∨Ldb1 ∨Lmo2 ∨C/EBPα)*
Gata2	*(Gata2) ∧¬ ((Gata1 ∧ Fog1) ∨(Sfpi1 ∧ Fli1))*
Gfi1	*¬ (EgrNab)*
Ldb1	*(cJun ∨EgrNab ∨Gata2)*
Lmo2	*(Ldb1 ∨(Gata2 ∧ Ldb1))*
Scl	*(Gata1) ∧¬ (Sfpi1)*
Sfpi1	*(C/EBPα ∨Sfpi1) ∧¬ (Gata1 ∨(Gata2 ∧ Fli1))*

Figure 2.11 BTR-trained Boolean models.
(A) Trained Bonzanni model, (B) Trained Krumsiek model.

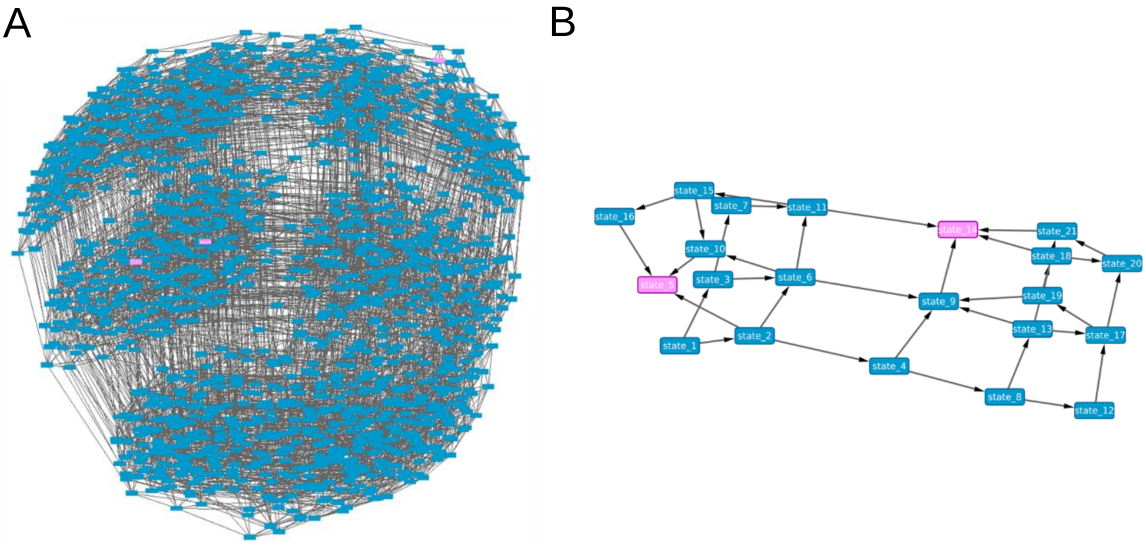


Figure 2.12 State spaces for the trained Bonzanni and Krumsiek Boolean models.
(A) State space of trained Bonzanni model. (B) State space of trained Krumsiek model. Blue nodes represent transitional model states, while pink nodes represent steady model states. Each arrow indicates transitions among states.

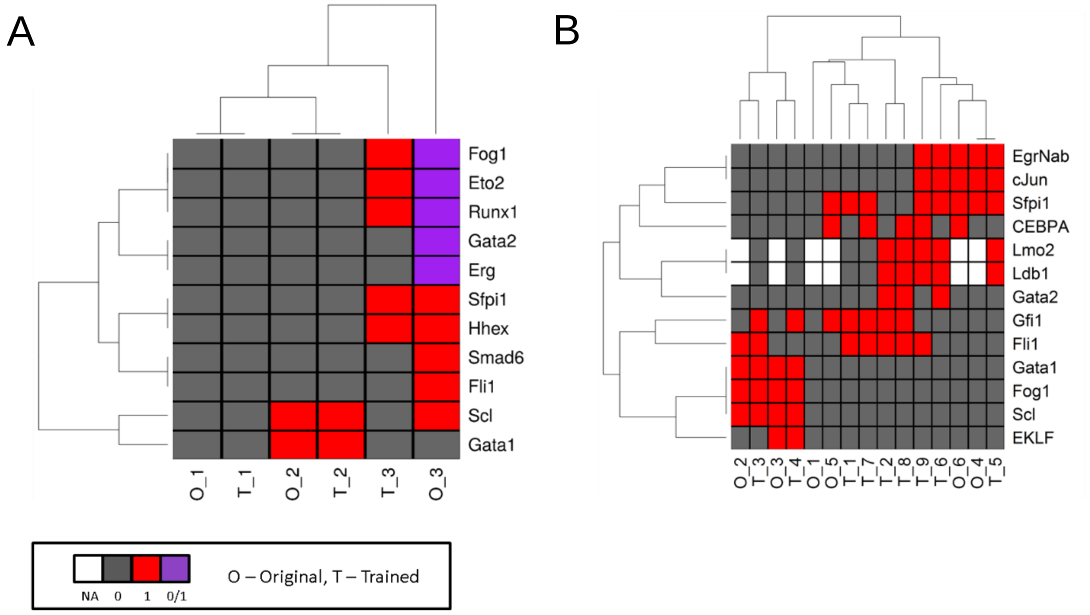


Figure 2.13 Steady states for the Bonzanni and Krumsiek Boolean models.
(A) Steady states of Bonzanni models. (B) Steady states of Krumsiek models.

2.6 Conclusions

The BTR model learning algorithm has been developed for training asynchronous Boolean models using single-cell expression data. The key component in BTR is a novel Boolean state space (BSS) scoring function, which BTR uses to infer a Boolean model through an optimisation process. The BSS scoring function has been shown to be capable of giving meaningful scores to networks when compared with the BIC scoring function for Bayesian networks. When compared to other network reconstruction algorithms, BTR gave the best result when initial networks were provided. In two case studies, BTR was able to suggest modifications to existing Boolean models based on information from single-cell qPCR and RNA-Seq data.

2.7 Materials and methods

2.7.1 Data preprocessing

BTR is capable of handling all types of expression data, including qPCR and RNA-Seq. Expression data should be processed and normalised before being used in BTR. In BTR, the expression data is further processed in order to facilitate score calculation by the BSS scoring function. Firstly, if the input data is qPCR expression data, it should be inversed such that the gene with a low expression level should have a low value and vice versa. Finally, the expression values for each gene in the data are scaled to continuous values with a range of $0 \leq x \leq 1$.

2.7.2 F-score as a measure of the performance of network inference algorithms

F-score, which is the harmonic average of precision and recall, represents precision and recall concisely (Sokolova et al. 2006), is often used to assess the performance of network inference algorithms. Precision denotes the proportion of edges that are truly present among all edges classified as present, while recall denotes the proportion of edges that are truly present among all correctly classified edges (including both edges that are present and absent) (Bockhorst & Craven 2005). The calculations were performed on a directed adjacency matrix.

Precision is defined as:

$$p = \frac{TP}{TP + FP}$$

Where TP = true positive and FP = false positive.

Recall is defined as:

$$r = \frac{TP}{TP + FN}$$

Where TP = true positive and FN = false negative.

F-score is defined as:

$$F = \frac{2pr}{r + p}$$

2.7.3 Synthetic data for benchmarking network inference algorithms

The synthetic data used for comparing scoring functions and network inference algorithms consist of true networks, expression data and lists of modified networks. The true networks and expression data were generated using GeneNetWeaver version 3.1.2 (Schaffter et al. 2011). The true networks contain 10 genes each and were extracted from the gene network of yeast. Each true network generated by GeneNetWeaver was then categorised into acyclic and cyclic networks. A total of five acyclic and five cyclic true networks were used in this study. The expression data were generated using ordinary and stochastic differential equations based on the true networks. A single time series expression data with 1000 observations were generated per true network, and the expression data were simulated under steady state wild type condition. A coefficient of 0.05 was used for noise term in the stochastic differential equations. The synthetic expression data as generated by GeneNetWeaver is used as non zero-inflated data. In addition, the synthetic expression data is converted into a zero-inflated data to simulate drop-outs in single-cell expression data by calculating the probability of a reading being a drop-out (i.e. zero value) based on its expression level. The probability of a reading being a drop-out, p_d, is modelled using the following equation:

$$p_d \ = \ 2^{-cy}$$

Where c = a constant (), and y = a reading of the expression level of a particular gene. In this study, $c = 6$ was estimated empirically by quantifying the distribution of gene expression values observed in real single-cell expression data.

The lists of modified networks were generated in R using the bnlearn package (Scutari 2010) for Bayesian networks and the BTR package for Boolean models. The modified networks were generated by modifying the number of edges that differ from the true network, ranging from 2 edges up to 40 differing edges. The modified Bayesian networks and the modified Boolean models were generated separately due to different underlying structural constraints imposed by each framework. In Bayesian framework all networks must be directed acyclic graphs, while

Boolean models do not have such restrictions. In contrast, Boolean models require explicit specification of activation and inhibition edges, while Bayesian networks handle activation and inhibition implicitly without modifying the edges. Although the generation of modified Bayesian networks and Boolean models were done separately and therefore they are not identical, all modified networks contain the same number of differing edges (2 to 40 edges) with respect to the true network. Note that the differences in edges for acyclic modified networks are not cumulative, due to difficulties in generating a directed acyclic graph with cumulative edge differences. The differences in edges for cyclic modified networks are also not cumulative to maintain consistency with the acyclic modified networks. For synthetic data, the initial state used for the simulation of Boolean models is the expression values at time $t = 0$.

2.7.4 Real experimental data from the haematopoietic system

Two Boolean models of haematopoiesis were used as initial models for model learning in this study, namely Krumsiek (Krumsiek et al. 2011) and Bonzanni models (Bonzanni et al. 2013). The update functions of both models were converted into functions with an activation clause and an inhibition clause, in which each of the clauses are individually expressed in disjunctive normal form. Note that one of the nodes (EgrNab) in the Krumsiek model comprises of three different genes, *Egr-1*, *Egr-2* and *Nab-2*. The initial states used in the simulation were obtained from both papers respectively.

A single-cell qPCR data and a single-cell RNA-Seq data, both obtained from Wilson et al. 2015 (Wilson et al. 2015), were used for model learning. The single-cell qPCR data contain 44 genes from 1626 cells (992 HSCs, 178 LMPPs, 147 CMPs, 185 GMPs and 124 MEPs), while the single-cell RNA-Seq data are collected from 96 HSCs. The expression data are processed and normalised as described in the original paper. For Bonzanni and Krumsiek models, the initial states used for the simulation Boolean models are obtained from each paper respectively.

2.7.5 Network inference algorithms and analyses software used

BIC and its associated hill-climbing algorithm are implemented in bnlearn (Scutari 2010). BestFit (Lähdesmäki et al. 2003) is an algorithm for inferring Boolean models under synchronous framework implemented in BoolNet (Müssel et al. 2010). ARACNE (Margolin et al. 2006) and CLR (Faith et al. 2007) are inference algorithms for inferring relevance networks based on mutual information. bc3net (de Matos Simoes & Emmert-Streib 2012) and GeneNet (Opgen-Rhein & Strimmer 2007) are inference algorithms based on Bayesian networks, while GENIE3 is a type of tree-based methods (Huynh-Thu et al. 2010).

Plots in this study were generated using ggplot2 (Wickham 2009), except network plots that were generated using Cytoscape (Shannon 2003) and heat maps that were generated using gplots (Warnes et al. 2015). Steady state analysis was performed using genYsis (Garg et al. 2008), which search for steady states reachable from all possible initial states.

3 Inferring gene regulatory networks with a pseudotime-ordered autoregression-based method

3.1 Background

In this chapter, a new framework for gene network inference, which is based on the autoregression formalism, is described. Autoregression is particularly suitable for predicting causal gene interactions using expression data with time information. This is because autoregression is a well-established method used widely for time series analysis, particularly in the field of economics (Granger 1981; Enders 2014). The fundamental concept of the autoregression framework is that if variable *a* affects variable *b*, a fluctuation in the value of variable *a* will lead to a fluctuation in the value of variable *b* at a later time point, assuming everything else is constant. This concept of inferring causality among variables is known as the Granger causality (Granger 1969). In the context of gene network inference, a target gene is regressed against all other genes with an autoregression formulation, and any non-zero coefficients inferred suggests the presence of gene interactions between the other genes with the target gene.

The typical implementation of autoregression, which uses the simple ordinary least square method for inferring coefficient, works very well with a large number of time points or samples. However it does not perform variable selection, which leads to an implicit assumption that all genes interact with all other genes and therefore is not suitable for gene network inference. To overcome this problem, regularisation terms are introduced to enable variable selection in penalised autoregression, such that the inferred coefficient matrix is sparse. While the use of regularisation enables variable selection, the presence of regularisation terms makes the calculation of uncertainty in inferred coefficients difficult. One way to ensure that the inferred coefficients are robust is through the use of random sampling techniques such as cross-validation and stability selection.

Here, I implemented stable penalised vector autoregression, SPVAR, which combines Elastic net regularisation with stability selection. SPVAR requires time-ordered expression data to infer a gene network, but most single-cell expression data do not contain high resolution time information. One way to overcome this problem is to introduce time information into the single-cell expression data by performing pseudotime inference. The pseudotime inference algorithm used here is diffusion map-diffusion pseudotime (DM-DPT), which is built on top of the diffusion map (Haghverdi et al. 2015) and diffusion pseudotime (Haghverdi et al. 2016) algorithms. SPVAR and DM-DPT were then used together as part of a single-cell network inference framework that works on single-cell RNAseq.

The single-cell network inference framework contains four main steps (Figure 3.1). The first step converts the single-cell RNAseq data into an expression data ordered by pseudotime, which is done by using DM-DPT. As the total number of genes is too big to be used for network inference, one way to reduce the number of genes to work with is to identify the genes that are differentially expressed as a function of pseudotime. This is done by performing likelihood ratio tests on negative binomial spline fitted models. Once the differentially expressed genes are identified, spline fit imputation is used to remove technical noise from the expression data. Finally, SPVAR can be used to infer gene networks on the pseudotime-ordered denoised single-cell RNAseq data.

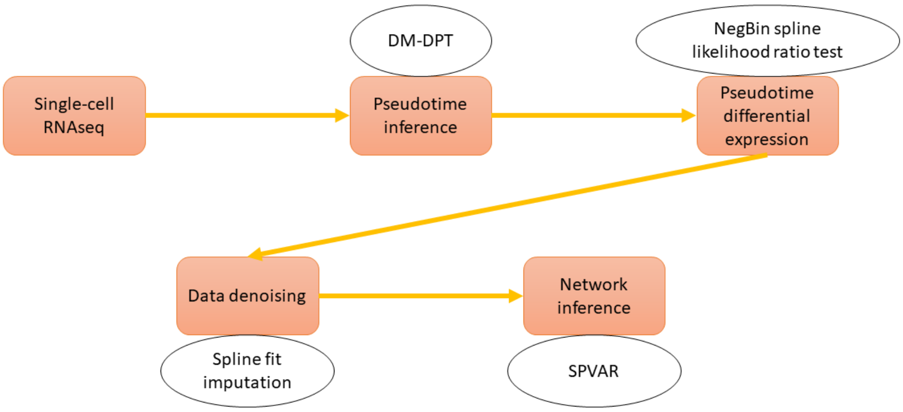


Figure 3.1 Single-cell network inference framework.

This chapter describes the single-cell network inference framework evaluated with synthetic data, with particular focus on the pseudotime inference method DM-DPT and the network inference algorithm SPVAR. Both DM-DPT and SPVAR are defined in Section 3.2. In Section 3.3, DM-DPT is shown to be superior to other existing pseudotime methods when tested with synthetic data. Section 3.5 shows the performance of SPVAR as a conservative network inference method that predicts fewer false positives than other network inference methods. In

Section 3.6, the entire network inference framework, which also includes pseudotime differential expression analysis and technical noise reduction, is shown to be a feasible framework for gene network inference from single-cell expression data. The chapter then ends with conclusions in Section 3.7, and materials and methods in Section 3.8.

3.2 Frameworks of PCA-based and DM-DPT pseudotime inference methods

This section describes the frameworks of the PCA-based and DM-DPT pseudotime inference methods.

3.2.1 PCA-based pseudotime trajectory

The PCA-based pseudotime inference method is implemented to infer a single developmental trajectory from a starting state to an ending state using single-cell expression data. In PCA-based pseudotime inference method, PCA was firstly performed on the $\log_2(x+1)$-transformed single-cell expression data, and the two principal components that best represent the developmental progression of cells are selected, which are usually from the first few principal components. Note that the expression data should be preprocessed and normalised as in a typical single-cell RNAseq processing pipeline. A polynomial curve with a degree of 3 was then fitted on the cells in the two dimension components representation.

Each cell was then projected onto the fitted curve at the projected point which has the shortest distance to the original coordinate of the cell (Figure 3.2). The projection was done analytically by solving for the minimum Euclidean distance between the original and the projected cell coordinates. Given a fitted curve of $y = \beta_3 x^3 + \beta_2 x^2 + \beta_1 x + \beta_0$, the Euclidean distance d_P between the original coordinate x_1, y_1 and the projected coordinate x_2, y_2 is given by $d_P = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$. The values of x_2, y_2 can then be obtained by solving for the first order differential of d_P in the form of $d'_P = 0$, which is equivalent to the equation for a line that is perpendicular to the curve at the point x_2, y_2 and passed through the point x_1, y_1. Since the $\log_2(x+1)$-transformed expression data and hence the PCA coordinates are bounded in terms of their numeric values, all possible solutions of x_2, y_2 were iteratively tested to find a coordinate x_2, y_2 that gives the minimum d_P. This x_2, y_2 coordinate was then regarded as the projected cell coordinate.

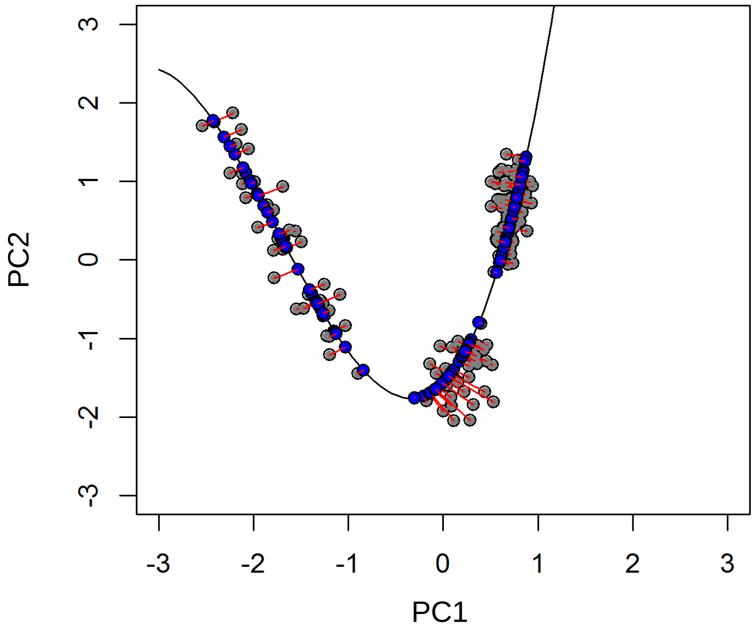


Figure 3.2 Orthogonal projections of cells onto the fitted curve in a dimensionally reduced space.
The grey points represent the original coordinates of the cells, while the blue points represent the projected coordinates of the cells. The red lines indicate the projections of cells onto the black fitted curve.

All projected cells were located on a single path, therefore the ordering among cells can be obtained by the orders of their projections on the curve. The two cells on the two ends of the path were identified as the tip cells, which can be further separated as the start and the end cells if external biological information about the tip cells (e.g. cell types) are available. The cells can then be ordered relative to the starting cell. Once the cell order is obtained, the next step involves inferring the distance between every pair of ordered cells, which can also be understood as the pseudotime between a pair of ordered cells. The pseudotime is calculated based on the distance between cells in a PCA represented by two principal components. Formally, the pseudotime t_P between two cells c_1 and c_2 was defined as the distance on the curve d_C between the PCA coordinates for projected cells $\hat{c}_1$ and $\hat{c}_2$. d_C between $\hat{c}_1$ and $\hat{c}_2$ is given by

$$d_C = t_P = \int_{x_1}^{x_2} \sqrt{1 + (y')^2}$$

Where y' is the first order differential of the curve y, x_1 and x_2 correspond to the x-coordinates of $\hat{c}_1$ and $\hat{c}_2$ respectively.

3.2.2 DM-DPT pseudotime trajectory

Besides the PCA-based pseudotime inference method, diffusion map-diffusion pseudotime (DM-DPT) pseudotime inference method was also implemented. DM-DPT is based on two related algorithms, diffusion map and diffusion pseudotime. Diffusion map (DM) is similar to kernel principal component analysis (PCA), which uses a kernel to represent the points in a low-dimensional non-linear subspace (Haghverdi et al. 2015). A normalised isotropic Gaussian kernel is used in DM to calculate a transition probability matrix based on the Euclidean distance in gene expression space. The idea of the transition probability as a diffusion process allows the modelling of the data as a continuously changing process, rather than other non-linear approaches such as tSNE which tends to split the data into disjoint groups.

Diffusion pseudotime (DPT) is computed based on the same transition probability matrix calculated for DM (Haghverdi et al. 2016). In DM, a few dominant eigenvectors of the transition probability matrix are used for the purpose of dimensionality reduction. However in DPT, the entire transition probability matrix is used to compute an accumulated transition probability matrix, which contains the sum of transition probabilities for each pair of cells across random walks of all lengths. DPT distance between any two cells is then defined as the Euclidean distance between the accumulated transition probabilities of the two cells.

The main motivation for building on top of DPT to yield DM-DPT is that while DPT gives the pseudotime between all pairs of cells, it does not explicitly specify the best approach to connect the cells into a connected graph, i.e. a trajectory. The easiest way is to iteratively connect two cells that share the minimum distance as measured by pseudotime. However this does not yield a good result as tested on a synthetic data (Section 3.3). An alternative way to construct a trajectory from the cells is to obtain the cell ordering information from the DM, as implemented in DM-DPT. The reason DPT is used for computing pseudotime rather than using the direct distances among cells on a diffusion map is because DPT considers all diffusion components when computing the pseudotime, while distances computed from a diffusion map only use information from two diffusion components. It should be noted that DM-DPT relies on two assumptions. Firstly, the diffusion components used in DM should ideally contain only one or multiple components that represent time progression, as well as one or multiple components that represent changes in gene expression due to time progression. Secondly, DM-DPT only works on a single unbranched trajectory, although this issue can be partly overcome by running DM-DPT multiple times separately on each of the branches.

The exact implementation is the same as the implementation of PCA-based pseudotime inference method as described in Section 3.2.1, with only two major differences. Firstly, a DM is used in this method in place of a PCA. Secondly, the pseudotime in this method is calculated using DPT rather than calculating direct distances among cells on the DM. Formally, the pseudotime t_P between two cells c_1 and c_2 was defined as the DPT distance between cells c_1 and c_2.

Theoretically a higher number of dimension components can work with this method, although the difficulties come in selecting the best dimension components that represent developmental progression, as well as the extra computational efforts required to compute projections of cells onto the fitted surface in a high dimensional setting. Note that no branch point identification is performed by DM-DPT, so this method will only work on a single trajectory path. Outlier cells that cluster separately from other cells for technical reasons should be excluded from the pseudotime inference analysis.

3.3 Testing PCA-based and DM-DPT pseudotime using synthetic data

In order to assess the performance of the proposed DM-DPT pseudotime inference methods, six independent synthetic time series gene expression datasets, which contain 250 cells and either 1000 or 3000 genes, were generated by GeneNetWeaver as described in Section 3.8.1. Three of the synthetic expression datasets consist of 1000 genes, while the other three synthetic expression datasets consist of 3000 genes. Each dataset varies in terms of the structure and kinetics of the underlying synthetic gene regulatory networks, as well as the genes that are perturbed. Note that the synthetic expression data were simulated from an underlying gene regulatory network, and hence the perturbation of the expression of one gene will lead to perturbations of the expression of downstream genes, thereby representing a synthetic system that resembles a biological cell.

The synthetic expression data is a time series data, in which the expression state at a previous time point is used to simulate the expression state at the current time point. Each expression state consists of the expression values of all genes, therefore can be viewed as a cell which is represented by the expression state. As time progresses, the expression state of the cell changes according to the underlying gene interactions and perturbations. Note that the described process is very similar to cells that undergo systematic changes in expression state due to biological development, which is known to be influenced by both gene interactions and environmental factors. In order to capture the changes in expression states in cells along a developmental process, single-cell expression profiling experiments such as single-cell RNAseq are often performed. Pseudotime inference algorithms are then used to resolve the temporal order of cells according to their expression states, which then allows the understanding how gene expression changes reflect the developmental process. Therefore these synthetic expression datasets can be used as a robust framework to assess the performance of pseudotime inference algorithms.

3.3.1 Technical properties of synthetic expression data

Before using the synthetic expression data for performance assessment, it is important to investigate the technical properties of the underlying data. The synthetic expression data represent the expression states of cells collected along a single developmental trajectory, in which a proportion of genes are either upregulated or downregulated while the remaining genes remain constant (Table 3.1). The perturbation strengths for upregulation and downregulation are randomly generated, which ranges in proportion terms from 0 (no effect on gene expression) to 1 (complete turn off of gene expression). Note that the expression values of genes not directly affected by perturbations may also change as time progresses, due to interactions among the genes.

The original synthetic expression data as generated by GeneNetWeaver are stochastic and contain a small amount of noise, but in general the original synthetic expression data represent the ideal expression data that suffer from very little technical bias. In order to simulate the technical bias experienced in single-cell expression data, drop-outs and overdispersion noise were introduced into the original synthetic expression data to generate a separate set of data with single-cell noise, hereby denoted as single-cell synthetic expression data (Section 3.8.1).

Datasets	**Number of genes**			**Perturbation strength**		
	Upregulated	**Downregulated**	**Unchanged**	**Minimum**	**Median**	**Maximum**
Dataset 1	171 (0.17)	192 (0.19)	637 (0.64)	0.0014	0.3607	0.9929
Dataset 2	149 (0.15)	163 (0.16)	688 (0.69)	0.0022	0.3829	0.9777
Dataset 3	184 (0.18)	171 (0.17)	645 (0.65)	0.0008	0.3664	0.9829
Dataset 4	479 (0.16)	532 (0.18)	1989 (0.66)	0.0001	0.3981	0.9993
Dataset 5	483 (0.16)	502 (0.17)	2015 (0.68)	0.0004	0.3741	0.9974
Dataset 6	477 (0.16)	543 (0.18)	1980 (0.66)	0.0018	0.3776	0.9975

Table 3.1 Summary of technical properties of synthetic expression data.
Values in brackets besides number of genes indicate the proportion relative to the total, rounded to 2 decimal places. The perturbation strength is represented in terms of proportions, which range from 0 to 1.

Diffusion map (DM) was performed as a low dimensional visualisation of the changes in expression states in the cells over time on both original and synthetic expression data (Figure 3.3). In general, it can be seen from Figure 3.3 that diffusion component 1 corresponds to the progression of cells in time as driven by the underlying changes in expression due to perturbations. For the original synthetic expression data, DM can be seen to captured the progression of cells very well, with almost all the cells lying on a perfect curve. In the single-cell synthetic expression data, DM can still capture the progression of cells rather well in a diffused arc even with the extra noise present. Note that although only the DM plots of two datasets are shown here, the results of both datasets are very similar and are representative of the rest of the datasets in terms of DM plots.

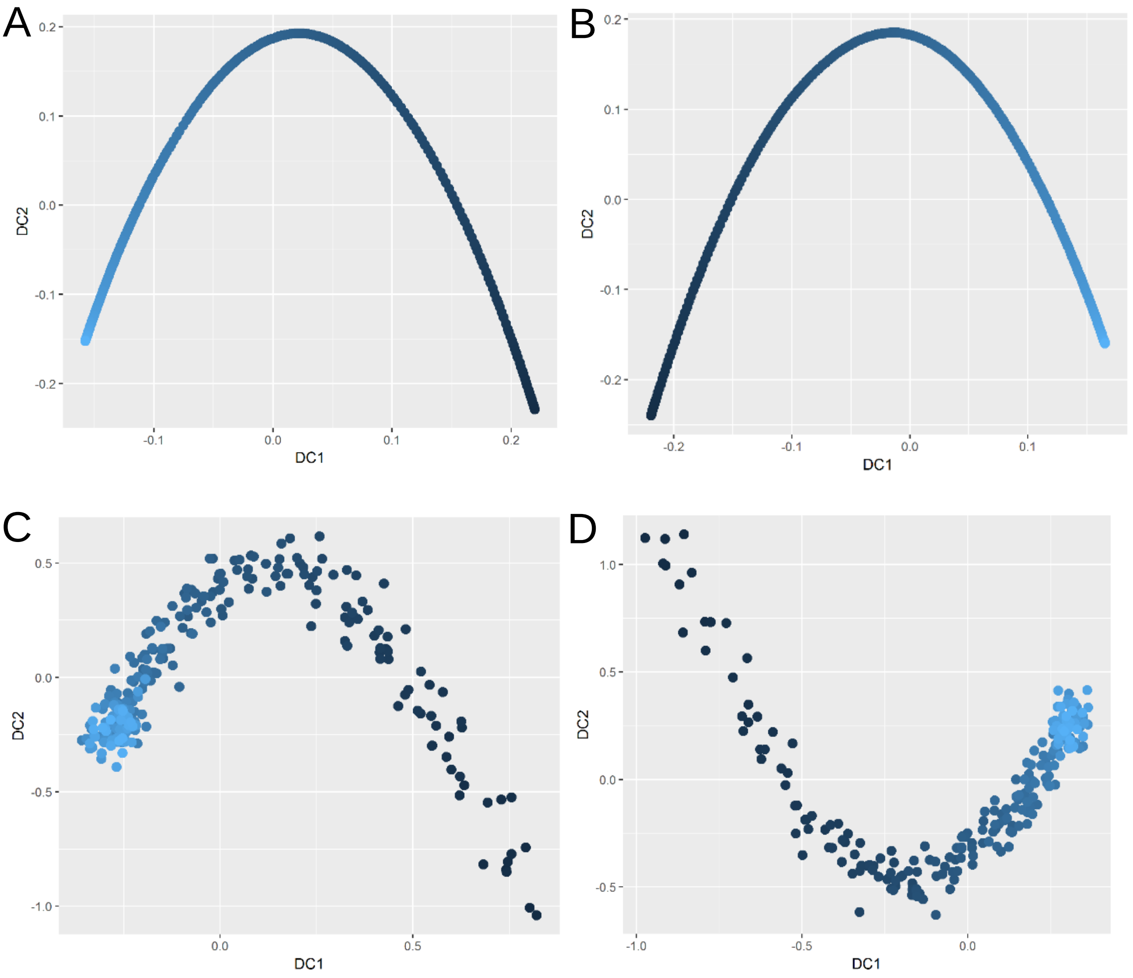


Figure 3.3 Diffusion map plots of the original and single-cell synthetic expression dataset 1 and 4.

(A) original synthetic dataset 1, (B) original synthetic dataset 4, (C) single-cell synthetic dataset 1, (D) single-cell synthetic dataset 4. The blue colour gradient indicates the time progression from starting cells (dark blue) to ending cells (light blue). Synthetic dataset 1 consists of 250 cells and 1000 genes, while synthetic dataset 4 consists of 250 cells and 3000 genes.

3.3.2 Assessing performance of pseudotime inference algorithms on original and single-cell synthetic expression data

With the technical properties of the original and single-cell synthetic expression data explored, pseudotime inference algorithms were run on these datasets to assess their performance. The basis for the performance assessment is that since the synthetic expression data is simulated in a sequential manner for each time point, the true cell ordering and time elapsed are known for each synthetic expression data. Therefore by comparing the true and inferred cell ordering, as well as the true time and inferred pseudotime, it is possible to objectively and quantitatively assess the performance of pseudotime inference algorithms. It is worth noting that the kinetics of gene expressions are highly non-linear, both in terms of responses to external perturbations and gene interactions. This leads to an interesting test case where the ability of pseudotime inference algorithms in dissecting non-linearity in the system can be tested.

The algorithms were assessed by two performance criteria, namely the cell ordering error and the pseudotime error, using all six synthetic datasets (Figure 3.4). The cell ordering error is quantified by the absolute difference between the true and inferred cell order, while the pseudotime error is measured by the absolute difference between the true and inferred pseudotime.

Seven pseudotime inference algorithms were tested here, which are Monocle2, TSCAN, SCORPIUS, SLICER, DPT, PCA-based pseudotime and DM-DPT. Monocle2 is based on principal graph-based method, DDRTree, for dimensionality reduction, and minimum spanning tree for trajectory inference (Qiu et al. 2017). TSCAN is based on principal component analysis and model-based clustering for dimensionality reduction, and travelling salesman problem (TSP) algorithm for trajectory inference (Ji & Ji 2016). SCORPIUS is based on multidimensional scaling for dimensionality reduction, and principal curve for trajectory inference (Cannoodt, Saelens, Sichien, et al. 2016). SLICER is based on locally linear embedding for dimensionality reduction, and shortest connected graph for trajectory inference (Welch et al. 2016). DPT is based on diffusion map for dimensionality reduction (Haghverdi et al. 2015; Haghverdi et al. 2016). In its original implementation, DPT only calculates a pseudotime distance matrix among all cells, and does not explicitly specify an ordering of cells in a trajectory. The DPT method used in comparison here was based on this cell distance matrix, and the trajectory was generated by assuming the order of each cell is based on their respective cell distance to the first cell in the trajectory.

PCA-based pseudotime is included here as a control, because it represents one of the simplest pseudotime inference algorithms possible. Wanderlust/Wishbone (Bendall et al. 2014; Setty et al. 2016) and embeddr (Campbell et al. 2015) pseudotime inference algorithms were also tested on these test datasets. However, complete results could not be obtained as implementation errors arise when running these algorithms on certain datasets, likely due to the relatively smaller number of cells present in the test datasets.

When assessed using the original synthetic expression data which is less noisy and does not possess single-cell specific noise (Figure 3.4A & B), the top three pseudotime inference algorithms with the least cell ordering errors were SLICER, DPT and DM-DPT. All top three algorithms achieved near perfect results when inferring cell orders using the original synthetic expression data. In terms of pseudotime error, the top three algorithms were DM-DPT, DPT and Monocle2. In the case of single-cell synthetic expression data, the top three algorithms were SCORPIUS, DM-DPT and PCA-based pseudotime in terms of cell ordering errors (Figure 3.4C & D). In terms of pseudotime errors, the top three algorithms were DM-DPT, Monocle2 and PCA-based pseudotime. It should be noted that both TSCAN and SLICER always assume equidistant pseudotime between each pair of consecutive cells, therefore the results from TSCAN and SLICER were not included when pseudotime errors were assessed because they will always get 0 error due to the underlying true time having equal time intervals.

In summary, DM-DPT is consistently among the best pseudotime inference algorithms when tested on both original and single-cell synthetic expression data. This is especially true when assessing the pseudotime errors, in which DM-DPT is the top algorithm largely due to its ability in dissecting the non-linear gene expression changes. Among the two custom implemented methods, PCA-based pseudotime performed badly in both criteria due to the linear nature of PCA and the use of only 2-dimensional PCA space for cell ordering and pseudotime inference.

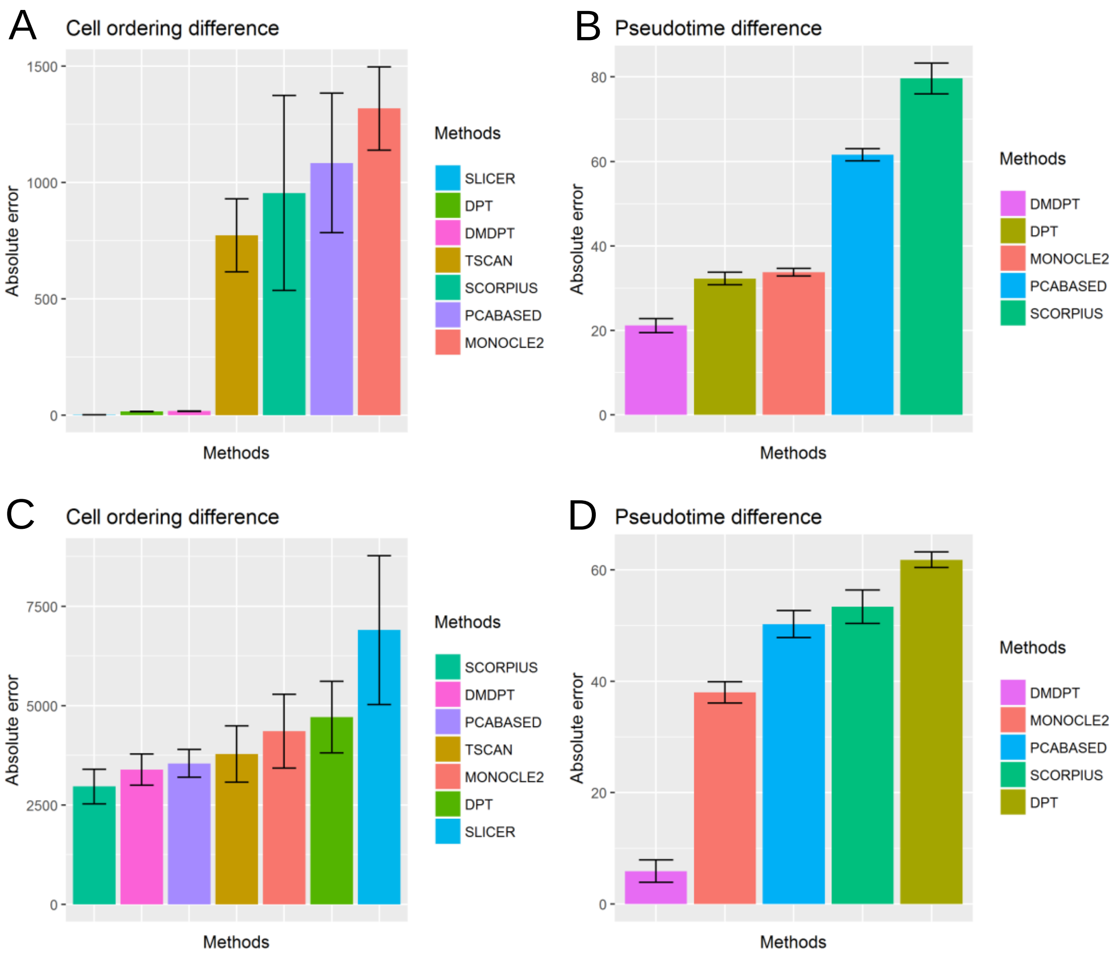


Figure 3.4 Cell ordering and pseudotime errors of pseudotime inference algorithms using the original and single-cell synthetic expression datasets.
(A) Cell ordering errors on original synthetic datasets, (B) Pseudotime errors on original synthetic datasets, (C) Cell ordering errors on single-cell synthetic datasets, (D) Pseudotime errors on single-cell synthetic datasets.

3.3.3 Assessing performance of pseudotime inference algorithms on sparsified synthetic expression data

Besides the presence of strong technical noise, expression profiling experiments can seldom sample the complete trajectory of expression state changes, due to both random sampling effects and experimental constraints. Random sampling effects can arise if the number of cells which is sampled for expression profiling is not enough to cover most of the trajectory, especially when the actual number of cells present is different along different developmental stages of the trajectory. This results in bias representations of the actual developmental trajectory. In addition, experimental constraints such as monetary issue can results in only certain stages of the trajectory from being sampled. This results in gaps in the expression data that correspond to the stages of trajectory that are not sampled. Therefore, it is important to evaluate pseudotime inference algorithms on synthetic expression data that also exhibit missing samples besides technical noise.

A set of synthetic expression data with missing samples was derived from the single-cell synthetic expression data, which is denoted as sparsified single-cell synthetic expression data (Section 3.8.1). Sparsified single-cell synthetic expression data possess missing samples in two forms. Firstly, some cells were randomly excluded and considered as missing along the entire trajectory. Secondly, huge gaps that represent entire sections of the trajectory that are not sampled were introduced by removing groups of adjacent cells on the trajectory. As before, the sparsified single-cell expression data were visualised through DM (Figure 3.5), in which the first diffusion component still represents the cell progression in the trajectory despite having a large amount of missing cells. There are fewer cells on the plots when compared to Figure 3.3 due to the missing cells, and there are also two obvious gaps in the plots which correspond to two sections of the trajectory that are not sampled.

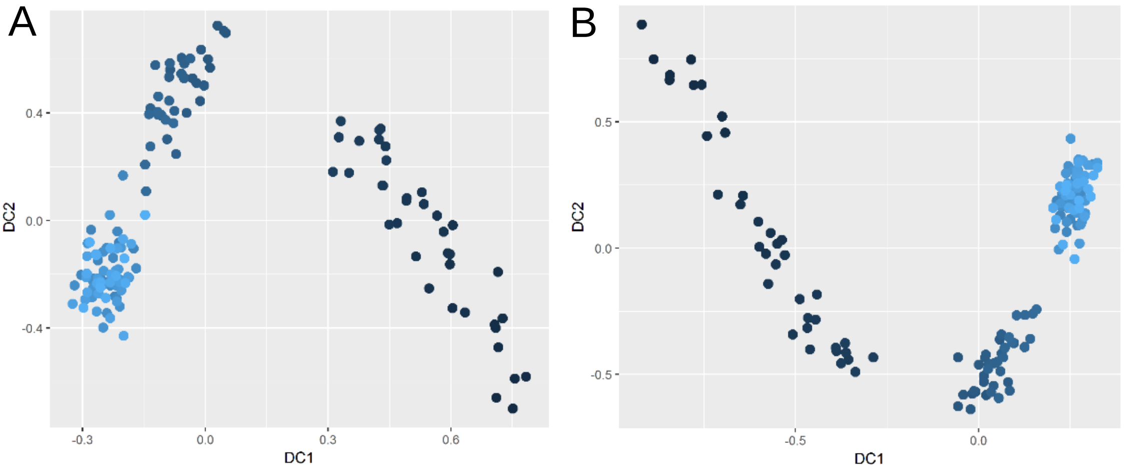


Figure 3.5 Principal component analysis and diffusion map plots of the sparsified single-cell synthetic expression dataset 1 and 4.
(A) DM of sparsified single-cell synthetic dataset 1, (B) DM of sparsified single-cell synthetic dataset 4. Synthetic dataset 1 consists of 250 cells and 1000 genes, while synthetic dataset 4 consists of 250 cells and 3000 genes.

The same set of seven pseudotime inference algorithms were tested on sparsified single-cell synthetic expression data, with cell ordering and pseudotime errors calculated as before (Figure 3.6). In terms of cell ordering errors, DM-DPT, SCORPIUS and PCA-based pseudotime were the top three performers. In terms of pseudotime errors, DM-DPT, Monocle2 and PCA-based pseudotime were the top three performers. It is worth noting that DM-DPT has the most consistent cell ordering and pseudotime results among all algorithms, with standard deviations of 385.97 and 2.24 respectively. In the presence of sparsified synthetic expression data, the performance of SLICER deteriorates drastically, which is due to the algorithm excluding cells from cell ordering and pseudotime inferences. In addition, the gaps in sparsified synthetic expression data may also cause each groups of cells separated by gaps to be categorised as a separate branch, especially for algorithms that rely on an additional clustering step. Note that since all synthetic data only contains a single trajectory, branch inference in the pseudotime inference algorithms was turned off where possible and the branch detection capability of the algorithms was not assessed here.

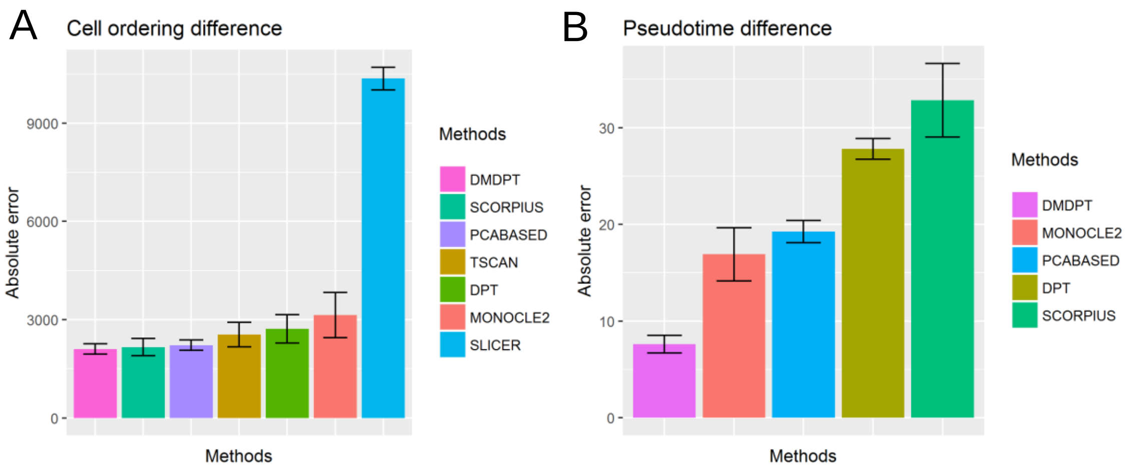


Figure 3.6 Cell ordering and pseudotime errors of pseudotime inference algorithms using the sparsified single-cell synthetic expression datasets.
(A) cell ordering errors on sparsified single-cell synthetic datasets, (B) pseudotime errors on sparsified single-cell synthetic datasets.

In summary, DM-DPT offered the best performance both in terms of inferring cell order and pseudotime in sparsified single-cell synthetic expression data, which suggests that DM-DPT is capable of inferring robust single unbranched trajectory from single-cell RNAseq data.

3.4 Framework of SPVAR gene regulatory network inference algorithm

This section describes the framework of the SPVAR gene regulatory network inference algorithm.

3.4.1 Definitions of stable penalised vector autoregressive model

The stable penalised vector autoregressive (SPVAR) model has been developed to infer gene regulatory network by using time-series gene expression data. The core concept that underlies SPVAR is the Granger causality, which basically states that causes must occur before the effects (Granger 1969). That is a change in the gene expression of gene A, x_A, at time *t* must be due to an event that occurs before time *t*, possibly due to the change of the gene expression of gene B, x_B, at time *t-1*. Such causal relationships can be detected through modelling time-series gene expression data using an autoregressive model, which is a well-established regression-based method for understanding time-series data widely used in economics (Granger 1981) and neuroscience (Eichler 2005). Hence by modelling time-series expression data using autoregression-based methods such as SPVAR, the resulting fitted model corresponds to a reconstructed gene regulatory network that can explain the temporal changes in gene expression due to gene interactions.

SPVAR works on both bulk time series and single-cell expression data, however it is designed for single-cell expression data due to the increased data resolution provided by the higher number of samples available. The effect of single-cell specific noise can be mitigated by running SPVAR on pseudo-expression values, which are estimated by fitting splines on actual expression values in a pseudotime trajectory. In the following subsections, I will describe the framework, which includes definitions and principles, that underlies SPVAR.

3.4.1.1 Vector autoregressive model

A stable penalised autoregressive model is a generalisation of a regression model, which can be represented in a general linear form as

$$y = \beta x + \varepsilon\ ,$$

Or a non-linear form as

$$y = f(z) + \varepsilon\ , z = \beta x,$$

Where *y* is the response variable, *x* is the independent variable, *β* is the regression coefficient, $f(.)$ represents the link function and the errors ε are assumed to be independently and identically distributed with $\varepsilon \sim N(0,\ \sigma^2)$.

A simple regression can be generalised into an autoregressive model, which is used to model time-series data. In a first order autoregressive model AR(1), the response variable x_t at time *t* is regressed on its past value with a time lag of 1, x_{t-1}. Note that while the errors ε_t are still normally distributed with mean 0 and constant variance, they are no longer independent from one another. β_0 is a variable-specific constant.

$$x_t = \beta_1 x_{t-1} + \beta_0 + \varepsilon_t$$

An autoregressive model can be further generalised into a multivariate vector autoregressive model where *p* variables are represented as $x_i,\ i\ \in \{1, 2, \ldots, p\}$, with each x_i having an equation of the form stated below.

$$x_{i,t} = \sum_{i=1}^{p} \beta_i x_{i,t-1} + \beta_0 + \varepsilon_t$$

A vector autoregressive model can be used to describe a gene regulatory network, where each variable x_i represents the expression level of a gene and β_i describes the interaction among the genes. For any regression model, the key is to estimate the values of *β* coefficients by minimising the sum of the squared differences between the observed and fitted data.

$$\hat{\beta} = \arg\min_{\beta} \sum_{i=1}^{p} (y_i\ - \beta_i x_i)^2$$

$$= \arg\min_{\beta} \sum_{i=1}^{p} \varepsilon_i^2$$

In a simpler regression model, the values of *β* coefficients can be estimated by ordinary least squares. However for more complex regression models that includes penalty terms, *β* coefficients have to be estimated through an optimisation process, such as gradient descent.

3.4.1.2 Variable selection via penalty terms

When fitting any model, variable or feature selection is a very important step in determining what variables should be included in the fitted model. The aim of variable selection is to select a subset of variables with the minimum number of variables, P_{min} where $P_{min} \subset P$, that can best explain the data observed. In the context of inferring gene regulatory network, this minimum set of variables corresponds to the set of the most important genes whose expression values influence the expression value of a target gene. Variable selection is especially important for gene regulatory network inference, as the number of genes in a complex biological system, such as mice, far exceeds the number of samples available.

There are many ways in which variable selection can be achieved, such as through stepwise or criterion-based procedures (Guyon & Elisseeff 2003). In a generalised linear model, penalty terms can be introduced into the model fitting process as regularisation to perform variable selection. Commonly used regularisation methods include Lasso (L1-norm), Ridge (L2-norm) and Elastic net (L1L2-norm). In summary, Lasso regularisation tends to set less important regression coefficients to zero, while Ridge regularisation tends to shrink the regression coefficients of correlated variables together. In SPVAR, the regularisation method used is the Elastic net, which offers a mixture of Lasso and Ridge regularisations that is controlled by the parameter α. $\alpha = 0.5$ is used in this algorithm, which means it has equal contributions from Lasso and Ridge regularisations. This enables Lasso regularisation to produce a sparse matrix where most $\hat{\beta}_i = 0$, while Ridge regularisation helps limit the $\hat{\beta}_i$ values when many of the variables are highly correlated (Friedman et al. 2009).

With Elastic net regularisation, β coefficients can be estimated by minimising the following objective function.

$$\hat{\beta} = \arg\min_{\beta} \sum_{i=1}^{p} (y_i \ - \beta_i x_i)^2 + \ \lambda \, [\alpha \|\beta\|_1 + \ (1-\alpha)\|\beta\|_2]$$

$$= \arg\min_{\beta} \sum_{i=1}^{p} (y_i \ - \beta_i x_i)^2 + \ \lambda \, [\alpha \sum_{i=1}^{p} |\beta_i| + \ (1-\alpha) \sqrt{\sum_{i=1}^{p} |\beta_i|^2}]$$

Where λ is a parameter that specifies the degree of regularisation, α is a parameter that specifies the mixed contribution of L1 and L2-norm, $||\beta||_1$ corresponds to L1-norm and $||\beta||_2$ corresponds to L2-norm.

SPVAR uses the GLMNET R package, which is based on cyclical coordinate descent optimisation, to estimate the $\hat{\beta}$ with Elastic net penalty (Friedman et al. 2009). $\hat{\beta}$ is estimated by fitting by a regularisation path which consists of a sequence of K λ values, where $K = 100$. λ_{max} is defined as the λ value when all $\hat{\beta} = 0$, while λ_{min} is defined as $c\lambda_{max}$, where $c = 0.001$. The sequence of K λ values is then taken as the series of λ values that decrease from λ_{max} to λ_{min} in the log scale.

3.4.1.3 Stability selection

Variable selection can reduce the number of variables to be included in a model by selecting for the subset of variables P_{min}. However, P_{min} inferred can vary depending on the variable selection techniques, parameters and data used, especially on high dimensional problems that involve a higher number of variables than the number of samples (Meinshausen & Bühlmann 2010). Validation methods that subsample data repeatedly such as bootstrapping and cross validation can be employed to ensure the generality of P_{min} irrespective of parameters and data used. One such method that is employed in SPVAR is known as stability selection (Meinshausen & Bühlmann 2010). In stability selection, the data is subsampled multiple times without replacement for model fitting with variable selection, and the variables that are present in a large proportion of the resulting models are selected as stable P_{min}.

As mentioned in Section 3.4.1.2, a regularisation path is used to fit a penalised linear model, which gives a vector $\hat{\beta}_i, i \in \{1,2,\dots,p\}$ for every λ. Stability selection is built upon the concept of regularisation path, by extending it into the concept of stability path. When a penalised linear model is fitted, a stability path is obtained by calculating the probability of a variable being selected in the results, also known as the selection probability P^S, for each variable across all sets of subsampled data such that there is a vector $P_i^S, i \in \{1,2,\dots,p\}$ for each λ, where $\lambda_{min} \leq \lambda \leq \lambda_{max}$. The selection probability P_i^S for gene variable i is defined as

$$P_i^S(\hat{\beta}_i \neq 0) = \frac{n_{\hat{\beta}_i \neq 0}}{n_D}$$

Where $n_{\hat{\beta}_i \neq 0}$ is the number of non-zero $\hat{\beta}$ coefficients and n_D is the total number of set of subsampled data. In SPVAR, 90% of all data is used per subsample as this provides better sampling coverage for the cases where the number of samples is small.

With a stability path, instead of deciding on a single λ value that gives the most optimal P^S, it is better to take a consensus $\hat{P}^S$ across all P_i^S obtained with a sequence of λ values as defined below.

$$\hat{P}^S = \max_{1 \le i \le n_\lambda} (P_i^S)$$

Where n_λ is the total number of λ values.

The set of stability selected variables S is then defined by using a threshold $p_{thre}, 0 < p_{thre} < 1$ as follows.

$$S = \{\hat{P}_i^S \ge p_{thre}\}, \qquad 1 \le i \le n_\lambda$$

The use of stability selection ensures that the selected variables are insensitive to p_{thre} and the sequence of λ values used (Meinshausen & Bühlmann 2010). Meinshausen and Bühlmann have suggested $p_{thre} \in (0.6, 0.9)$ with a lower bound of $p_{thre} \ge 0.5$, which is based on empirical evidence and the assumption that variable selection works better than by chance. In SPVAR, a threshold of 0.6 is used as single-cell RNAseq data is very noisy.

3.4.2 Implementation of stable penalised vector autoregressive model

SPVAR is implemented in R, and uses the GLMNET R package for fitting a penalised linear regression. SPVAR takes a $m \times p$ matrix as the input expression data, where each row m corresponds to a time point and each column p corresponds to a gene. Since autoregressive models work best when the time series data is stationary, that is there is no consistent overall trend in the gene expression values, the input expression data usually need to be differenced,

$$x_t' = (x_t - x_{t-1})$$

First order differencing calculates the difference between two consecutive time points as demonstrated above. The order of differencing depends on the strength of trend in the data and the number of time points available. Generally, a higher order is required for data with a stronger trend, however this is feasible only if there is a large enough number of time points available because each increasing order will reduce the number of time points available. In practise, the order is rarely more than 2.

Once the input data is ready, the next step is to obtain a fixed sequence of λ values as described in Section 3.4.1.2, and generate a set of subsampled data for stability selection. Model fitting is then performed on each subsampled data, and stably selected genes are selected based on the selection probabilities calculated. Note that the observations in the time series data should be weighted for model fitting if the time points are not evenly spaced, or if a subset of time points are known to be more important than the others. As differencing on the data may not be able to make the data stationary, this will confound the effects exerted by actual causal genes on a target gene. To reduce this confounding effect, the past expression values of a gene are assumed to not affect its future expression values after differencing.

Finally, after obtaining the set of stably selected causal genes for each target gene, the stably selected genes are then used to fit a model which gives a $\hat{\beta}_i$ for each gene i. The value of $\hat{\beta}_i$ can be interpreted as the strength of the gene interaction, which is proportional to the probability of the inferred interaction being a true interaction. SPVAR outputs a $p \times p$ directed real-valued adjacency matrix that describes the interactions among the genes. Note that due to the general difficulty in inferring self-interaction in a network inference problem and the potential confounding effects arising from trying to account for self-interaction, SPVAR explicitly does not model self-interaction of any gene. The algorithm of SPVAR is summarised in the form of a pseudocode in Figure 3.7.

```
Preprocess time series data
Calculate differenced data
Setup a model framework
    Remove self-interaction
    Scale time points
    Obtain a sequence of λ values
Generate a list of subsampled data
    Fit a model to each subsampled data
Calculate the selection probabilities
Obtain the set of stably selected variables with associated selection probabilities
Output a directed real-valued adjacency matrix
```

Figure 3.7 Pseudocode of SPVAR.

3.5 Testing SPVAR using synthetic data

Once a pseudotime trajectory was inferred for a single-cell expression data, the inferred cell ordering and pseudotime enable the single-cell expression data to be considered as a time series expression data. This effectively allows the inference of gene interactions by studying the changes in gene expression according to time, even if the explicit experimental time considered during the expression profiling experiment does not have a high enough resolution or is not present. By considering a pseudotime-ordered single-cell expression data as a time series expression data, it is possible to infer causal interactions among genes based on Granger causality (Granger 1981). Many frameworks can be used for inferring gene regulatory networks with a time series expression data, which includes regression models. Here SPVAR, which is based on regression and is formally defined in Section 3.4.1, was developed for inferring gene regulatory network from a time series expression data. Specifically, SPVAR infers a multivariate vector autoregressive model, which is fitted with Elastic net regularisation and stably selected, for modelling gene regulatory network.

3.5.1 Technical properties of synthetic expression data

The performance of SPVAR was assessed by comparing with other gene regulatory network inference algorithms using a set of synthetic expression data as described in Section 3.8.1. Note that the synthetic expression data used here contains much fewer genes, therefore is different from the synthetic expression data used for assessing pseudotime inference algorithms. This is because pseudotime inference algorithms scale well with increasing number of genes in terms of computational complexity, while gene regulatory network inference usually scales more poorly with the number of genes. 10 independent synthetic time series gene expression data, which contain 50 cells and either 10 or 20 genes, were generated by GeneNetWeaver as described in Section 3.8.1. Five of the synthetic expression datasets have 10-gene networks, while the other five synthetic expression datasets have 20-gene networks. The properties of the datasets with associated networks were summarised in Table 3.2. The synthetic networks have a good range of numbers of edges, types of edges and are sparsely connected, which agree well with known structural knowledge on biological networks (Jeong et al. 2000).

Networks	Number of				Average degree per node	Clustering coefficients
	Positive edges	Negative edges	Total edges	Total nodes		
Network 1	5	8	13	10	2.6	0.207
Network 2	4	6	10	10	2	0.273
Network 3	5	4	9	10	1.8	0
Network 4	8	5	13	10	2.6	0.188
Network 5	13	5	18	10	3.6	0.105
Network 6	15	14	29	20	2.9	0.186
Network 7	9	20	29	20	2.9	0.186
Network 8	20	13	33	20	3.3	0.233
Network 9	18	11	29	20	2.9	0.079
Network 10	17	14	31	20	3.1	0.061

Table 3.2 Properties of the synthetic networks.
Clustering coefficient measures the relative number of triangles in the graph.

As the purpose here is to test the performance of causal direct gene interactions, the knocked out synthetic expression data were used for assessing the performance of gene regulatory network inference algorithms. Knocked out expression data represent a cleaner and simpler test case than randomly perturbed expression data, as the expression of a single gene is gradually reduced to zero over the time course of an expression data, in which any large changes in other gene expressions must be due to gene interactions with the knockout gene. Note that small random fluctuations that are not due to gene knockouts are also present in the datasets.

For each dataset, each gene is simulated to be knocked out individually, and the expression of other genes changes as a result of gene interactions. Similar to the assessment of pseudotime inference algorithms, the gene regulatory network inference algorithms were also tested on two sets of synthetic data, namely the original and the single-cell synthetic expression data. The single-cell synthetic expression data were derived from the original synthetic expression data by introducing drop-outs and overdispersion noise as described in Section 3.8.1. In addition, first order differencing of both time series expression data was performed to calculate the difference between consecutive time points as described in Section 3.4.2. The same differenced expression data were used for assessing all network inference algorithms to enable a fair comparison of their performances.

The properties of the data can be visualised in Figure 3.8, where the expression of a gene is reduced to 0 due to gene knockout in network 1 with 50 time points. Both original and single-cell synthetic expression data are stochastic and noisy, however the single-cell synthetic expression data is a lot noisier due to drop-outs and overdispersion (Figure 3.8A & B). The corresponding differenced data were shown in Figure 3.8C & D.

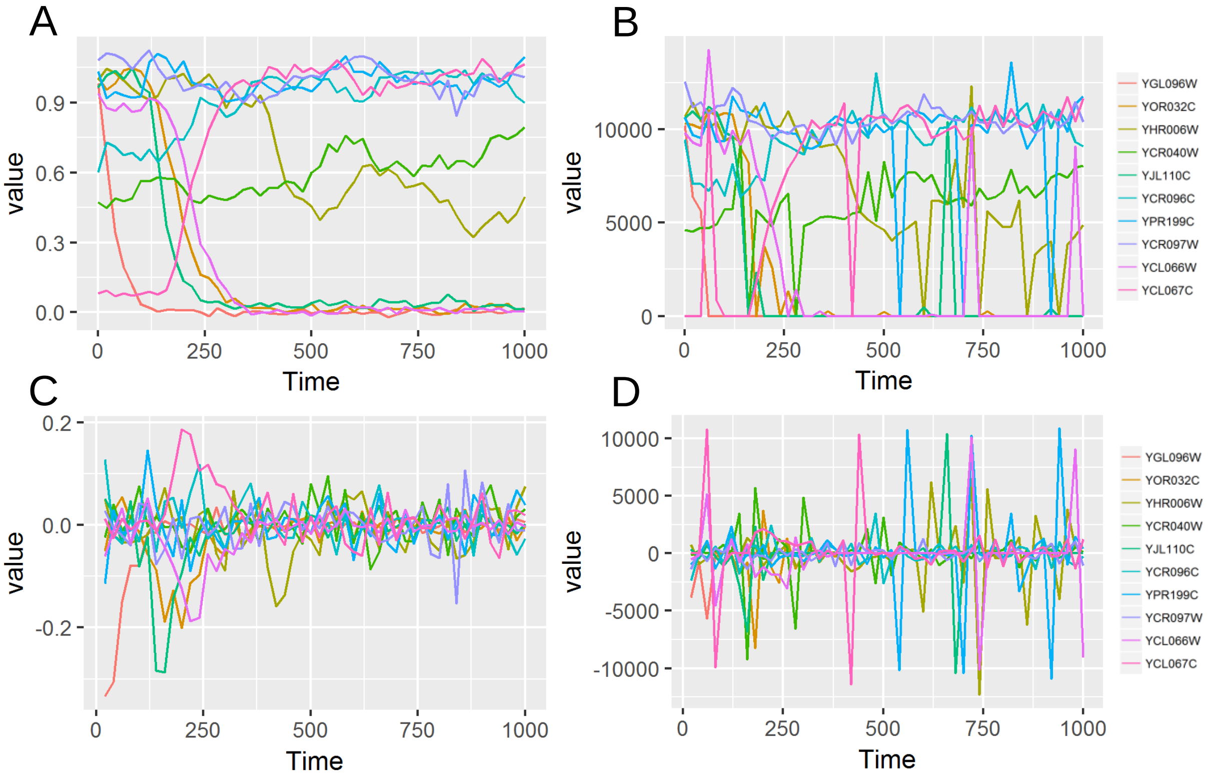


Figure 3.8 Expressions of genes over time with 50 time points.
This data contains 10 genes, where each coloured line corresponds to one gene. Gene 1 (red line, YGL096W) is knocked out in this data. (A) Original synthetic expression data, (B) Single-cell synthetic expression data, (C) Differenced original synthetic expression data, (D) Differenced single-cell synthetic expression data. (A) shows the original expression data without any additional noise or preprocessing. (B) shows the single-cell expression data, which is the original expression data in (A) with dropout and overdispersion effects added. (C) shows the difference in expression values between consecutive time points using the original expression data in (A). (D) shows the difference in expression values between consecutive time points using the single-cell expression data in (B).

3.5.2 Assessing performance of gene regulatory network inference algorithms on original and single-cell synthetic data

Five gene regulatory network inference algorithms were tested using the synthetic data, namely CLR, GENIE3, TIGRESS, EBDBNet and SPVAR. CLR is based on mutual information (Faith et al. 2007), GENIE3 is based on random forest (Huynh-Thu et al. 2010), TIGRESS is based on regression with variable selection using LARS (Haury et al. 2012), while EBDBNet is based on dynamic Bayesian network (Rau et al. 2010). CLR and GENIE3 are not designed for time-series data specifically, while TIGRESS has been adapted for time-series data current gene expression values on their past values. EBDBNet and SPVAR are designed for time-series data specifically.

In addition, random network inference results were also included in the study as controls, denoted as RMAT10 and RMAT100. Both random results were generated by sampling values from an exponential distribution, which ranges between 0 to 1, to construct artificial weighted adjacency matrices. RMAT100 uses the entire artificial weighted adjacency matrices without further modifications, while RMAT10 sets 90% of the elements in the matrices to 0 to simulate the sparse nature of biological gene networks.

The performance of gene regulatory network inference algorithms was firstly assessed by using the conventional receptor operating characteristic (ROC) and precision-recall (PR) curves (Figure 3.9). ROC and PR curves were used to assess the ability of algorithms to locate all gene interactions in both original and single-cell synthetic expression data. The most ideal algorithm that gives the best performance should have a ROC curve that passes through the (0,1) coordinate on the ROC graph, and a PR curve that passes through the (1,1) coordinate on the PR graph. Therefore a good algorithm should have ROC and PR curves that are as close as possible to the coordinates mentioned above. The ROC and PR curves of RMAT10 and RMAT100 can be interpreted as the baseline controls. The results in Figure 3.9 suggest that no network algorithm performs better than randomly generated networks. This suggests that gene network inference is a very hard problem, likely due to the high number of potential gene interactions available, the confounding effects from indirect gene interactions and complex network motifs. Note that the results from Figure 3.9 agrees with independent assessment of network inference algorithms on gene networks derived from yeasts (Marbach et al. 2012; Qi & Michoel 2012).

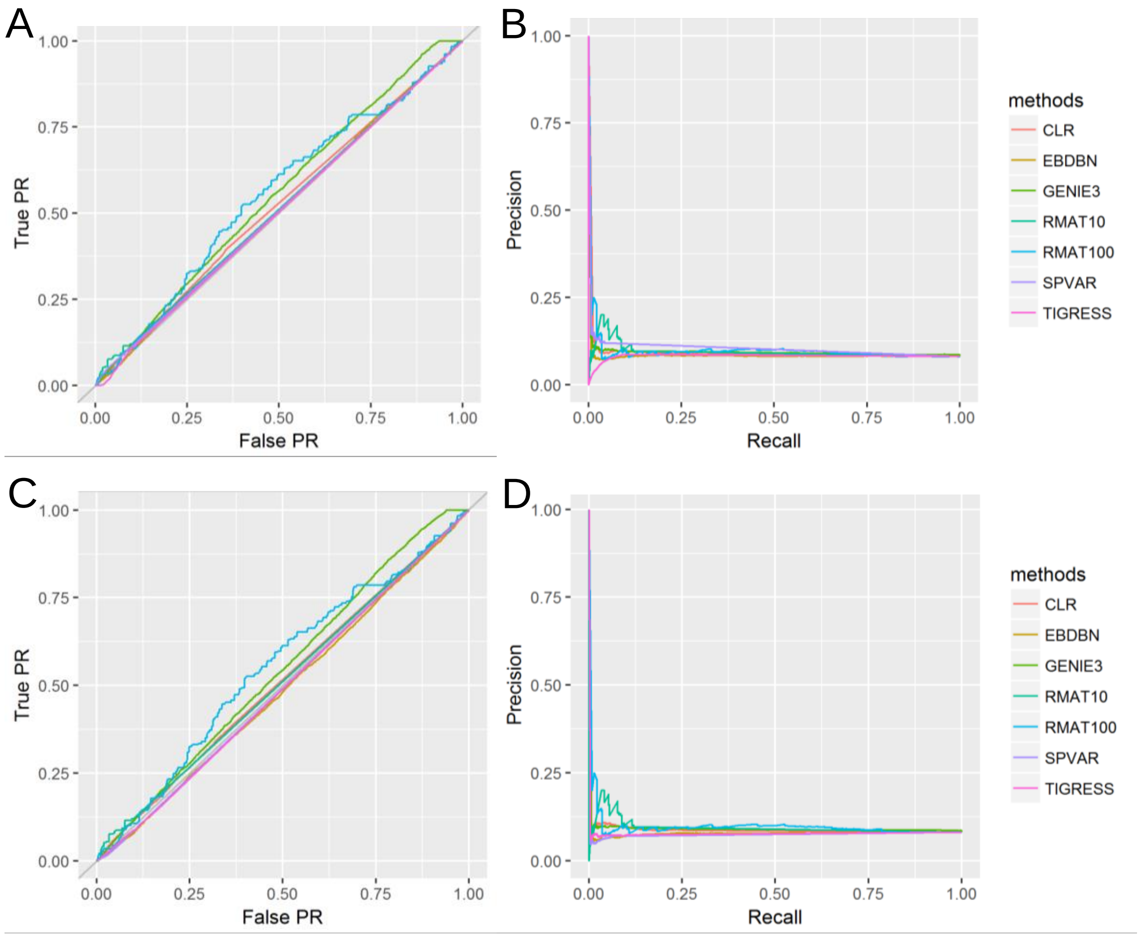


Figure 3.9 Receptor operating characteristic and precision-recall curves for each algorithm.
(A) ROC and (B) PR for original synthetic expression data, (C) ROC and (D) PR for single-cell synthetic expression data.

For a more straightforward evaluation of the performance of the algorithms, performance scores in the form of the F-score and the SAR measures were also computed. The F-score is a measure that summarises PR, while the SAR is a measure that combines accuracy, ROC and root mean squared error (Section 3.8.3). Note that the F-score and SAR measures provide complementary representations of the performance of network inference algorithms. F-score favours algorithms that predict more positives, while SAR measure favours algorithms that predict more negatives. A F-score and a SAR measure were computed for each algorithm with all networks combined.

When assessed using the original synthetic expression data which is less noisy and does not possess single-cell specific noise, the top three gene regulatory network inference algorithms were GENIE3, EBDBN and RMAT100 in terms of F-scores (Figure 3.10A), and SPVAR, TIGRESS and RMAT10 in terms of SAR measures (Figure 3.10B). When tested using single-cell synthetic expression data, the top three gene regulatory network inference algorithms with the best F-score GENIE3, EBDBN and RMAT100 (Figure 3.10C), which remained the same as before. However, the additional technical noise present in this data has resulted in RMAT10, SPVAR and TIGRESS being the top three algorithms (Figure 3.10D). The additional technical noise has resulted in poorer performance in the algorithms as expected.

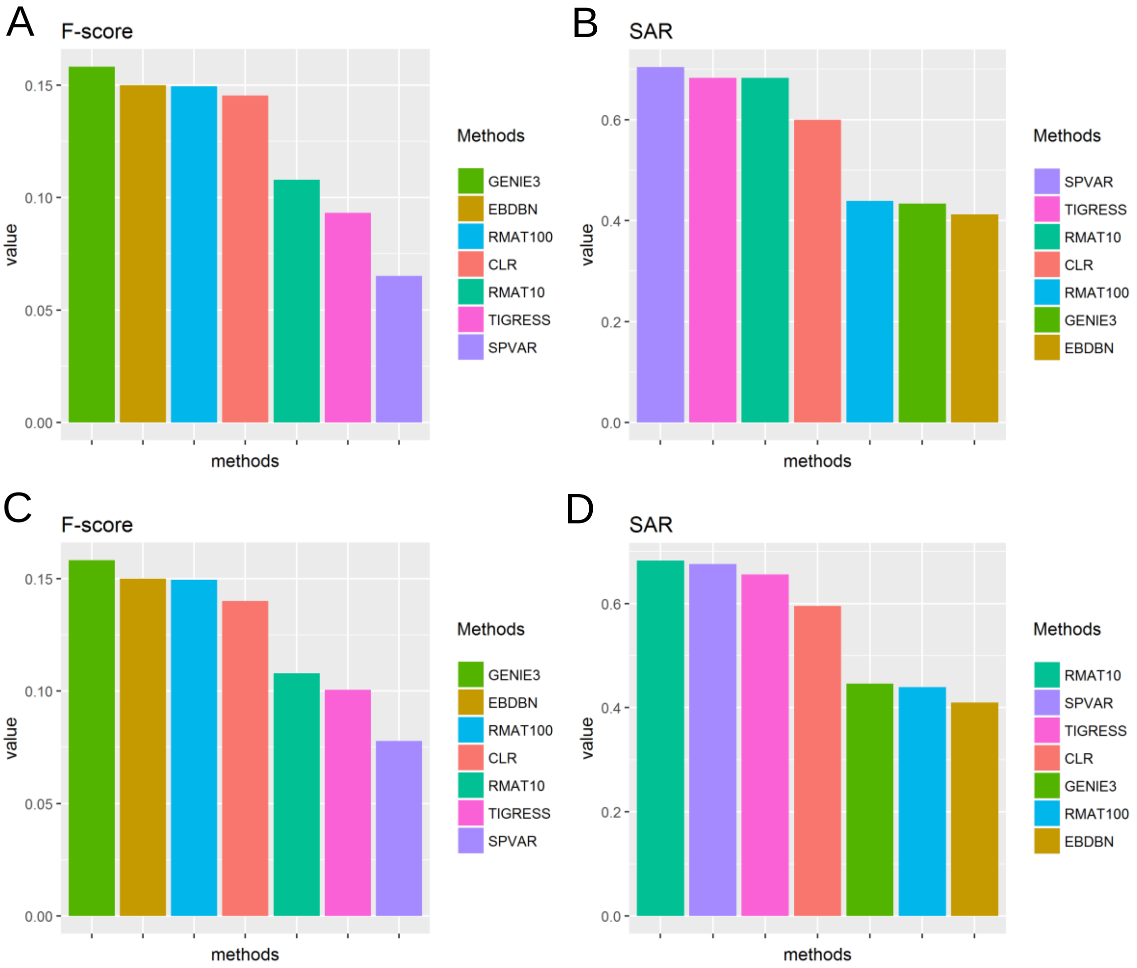


Figure 3.10 Performance scores of gene regulatory network inference algorithms. *(A) F-score on original synthetic expression data, (B) SAR measure on original synthetic expression data, (C) F-score on single-cell synthetic expression data, (D) SAR measure on single-cell synthetic expression data.*

The differences in the top algorithms between F-score and SAR measures are due to the properties of the scoring measures as indicated by the RMAT10 and RMAT100 controls. RMAT100 gives a fully weighted adjacency matrix without any zero, which results in a better F-score; while RMAT10 gives a sparse weighted adjacency matrix with only 10% of non-zero values, which results in a better SAR measure. This suggests that algorithms ranked higher in terms of the F-score tend to predict a fully weighted adjacency matrix that requires a user-defined threshold to extract a sparsely connected gene network. As for algorithms ranked higher in terms of the SAR measure, the algorithms tend to predict a sparse weighted adjacency matrix that does not require a user-defined threshold to extract a gene network. The sparsity in the adjacency matrix results from the use of regularisation during the model fitting process.

Taken together, the results show that the gene regulatory network inference algorithms tested here were unable to infer networks that are significantly better than by chance, even on synthetic data that possess a lower degree of noise than real data. The best network inference algorithms can be separated into two main categories, as indicated by F-score and SAR measures that rank algorithms based on different properties. Gene regulatory network inference algorithms which employed regularisation during the model fitting step, such as TIGRESS and SPVAR, result in sparser adjacency matrices that do not require users to define their own thresholds. Lastly, SPVAR has a similar level of performance compared to TIGRESS. However, SPVAR is more conservative in predicting the presence of gene interactions relative to TIGRESS as indicated by a higher SAR measure but a lower F-score.

3.6 Testing pseudotime differential expression and spline-fit imputation in single-cell network inference framework

It is important to test DM-DPT and SPVAR separately with synthetic expression data to assess their performance in their respective functions of inferring pseudotime trajectory and gene regulatory network. However it would be interesting to consider both DM-DPT and SPVAR in a single framework, by investigating how well SPVAR performs on pseudotime-series expression data inferred by DM-DPT when coupled with pseudotime differential expression and spline-fit imputation. The six independent synthetic expression datasets, which were used to assess the performance of pseudotime inference algorithms, were used to assess the performance of SPVAR on DM-DPT inferred pseudotime-series expression data. These datasets contain either 1000 or 3000 genes, which therefore represent a more realistic and difficult test case for SPVAR. Only the sparsified single-cell synthetic expression data, which represent the noisiest synthetic data, will be used for performance assessment here.

DM-DPT was ran on the single-cell synthetic expression data to obtain a pseudotime-series expression data as described in Section 3.3. As SPVAR does not scale well computationally relative to the number of genes present in the data, it is important to select the most important subsets of genes for inferring gene regulatory network. This was achieved here by using pseudotime differential expression analysis to select for the top 20 genes that are the most differentially expressed along the trajectory as a function of pseudotime.

The performance of negative binomial generalised linear model (GLM)-based pseudotime differential expression analysis was briefly assessed by comparing with Spearman rank correlation-based pseudotime differential expression analysis (Figure 3.11). The Spearman rank correlation pseudotime differential expression analysis works by calculating the correlation between each gene expression and the pseudotime. The negative binomial GLM pseudotime differential expression analysis works by firstly fitting a spline model for expression values of cells along the trajectory against the pseudotime for each gene. A likelihood ratio test is then performed on each of the fitted spline models against a null spline model where only an intercept is fitted for each gene. Genes with fitted models that had statistically significantly larger likelihood ratio test statistic ($p < 0.1$) were deemed as differentially expressed along the pseudotime.

Both Spearman rank correlation and negative binomial GLM differential expression analysis methods gave similar level of performance in these datasets, with Spearman rank correlation performing slightly better than negative binomial GLM differential expression analysis method in terms of F-score and SAR measure. However there are two things to note regarding the comparison of the two methods. Firstly, negative binomial GLM differential expression analysis method appears to be more conservative than Spearman rank correlation in predicting differentially expressed genes, with fewer genes predicted as differentially expressed (Figure 3.12). The conservative property of the negative binomial GLM may be a useful characteristic as this leads to fewer false positives. Secondly, the gene expressions in these datasets mostly exhibit monotonic relationships with pseudotime, which may lead to improved Spearman rank correlation results. This is because while both Spearman rank correlation and negative binomial GLM can detect non-linear relationship, Spearman rank correlation can only detect monotonic relationship, while negative binomial GLM can also detect non-monotonic relationship. It is expected that negative binomial GLM will perform better in real single-cell RNAseq data which exhibits non-monotonic relationships, and hence negative binomial GLM differential expression analysis method was chosen for detecting differentially expressed genes in this test case.

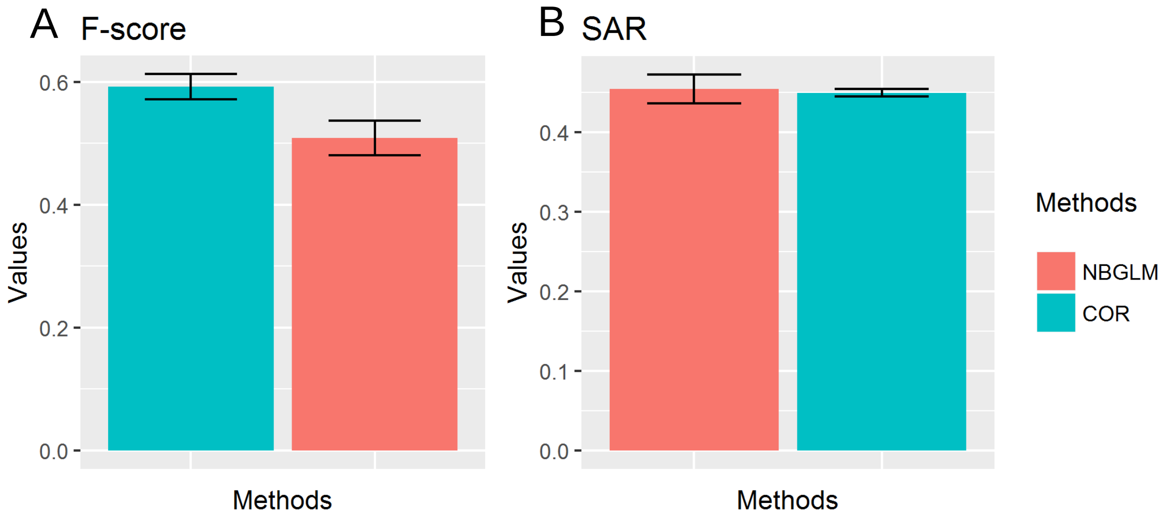


Figure 3.11 Performance comparison of two pseudotime differential expression analysis methods.
(A) F-score, (B) SAR measure. COR, Spearman rank correlation; NBGLM, negative binomial GLM.

Once the top 20 differentially expressed genes were selected, SPVAR can then be deployed to infer gene regulatory networks. As single-cell expression data is noisy, it is possible to use SPVAR either directly on the original expression data with single-cell noise or on the spline-imputed expression data, both of which are illustrated in Figure 3.13. The main advantage of the spline-imputed expression data is the reduced technical noise, which essentially average out the effects of outlier expression values that are caused by drop-outs and overdispersion.

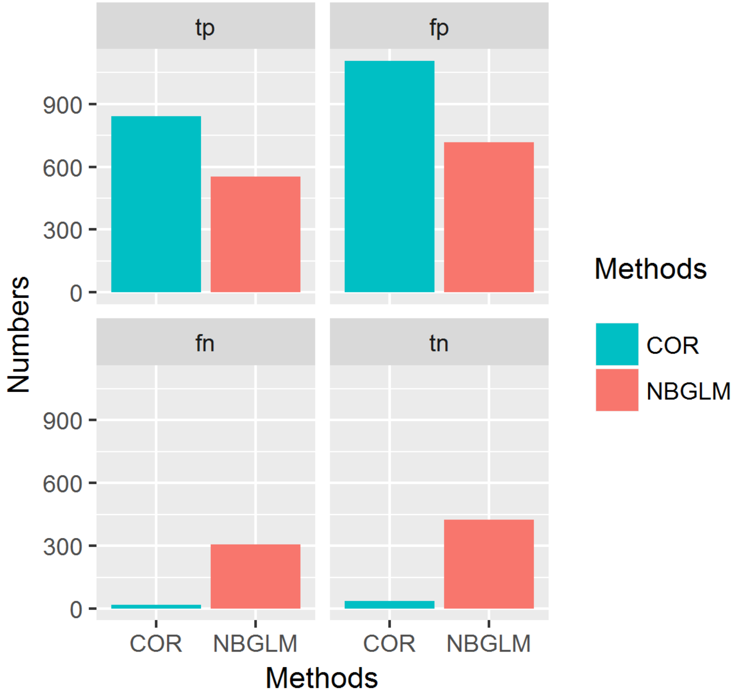


Figure 3.12   Numbers of true positive, false positive, true negative and false negative differentially expressed genes predicted by Spearman rank correlation and negative binomial GLM.

TP, true positive; FP, false positive; TN, true negative; FN, false negative; COR, Spearman rank correlation; MONO, negative binomial GLM.

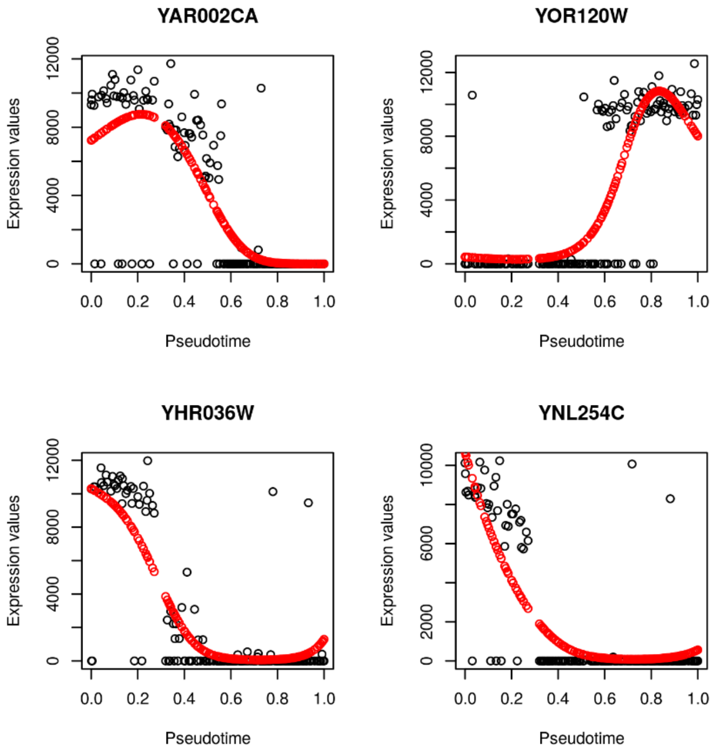


Figure 3.13 The expression values of four randomly selected genes which are differentially expressed as a function of pseudotime.
Black points represent original expression data, while red points represent spline-imputed expression data.

The performance of SPVAR on pseudotime-ordered time series expression data, as well as the impact of spline-imputed expression data on SPVAR were assessed in terms of F-score and SAR as before (Figure 3.14). When the spline-imputed expression data were used as an input for SPVAR, it can be seen that the performance of SPVAR improved in terms of F-score relatively to when the original expression data were used. Although it should be noted that the use of spline-imputed expression data led to higher variation in F-score, as the values of spline-imputed expression data depend on how well the splines were fitted. Note that SAR measure was higher on the original relative to the spline-imputed expression data due to SPVAR predicting more gene interactions with spline-imputed expression data. However, the gain in F-score was of a much higher degree than the drop in SAR measure, which justified the performance gain of using spline-imputed expression data. In summary, SPVAR has been demonstrated to perform well on DM-DPT pseudotime-ordered expression data, together with the use of negative binomial GLM for differential expression analysis and the use of spline imputation for reducing technical noise.

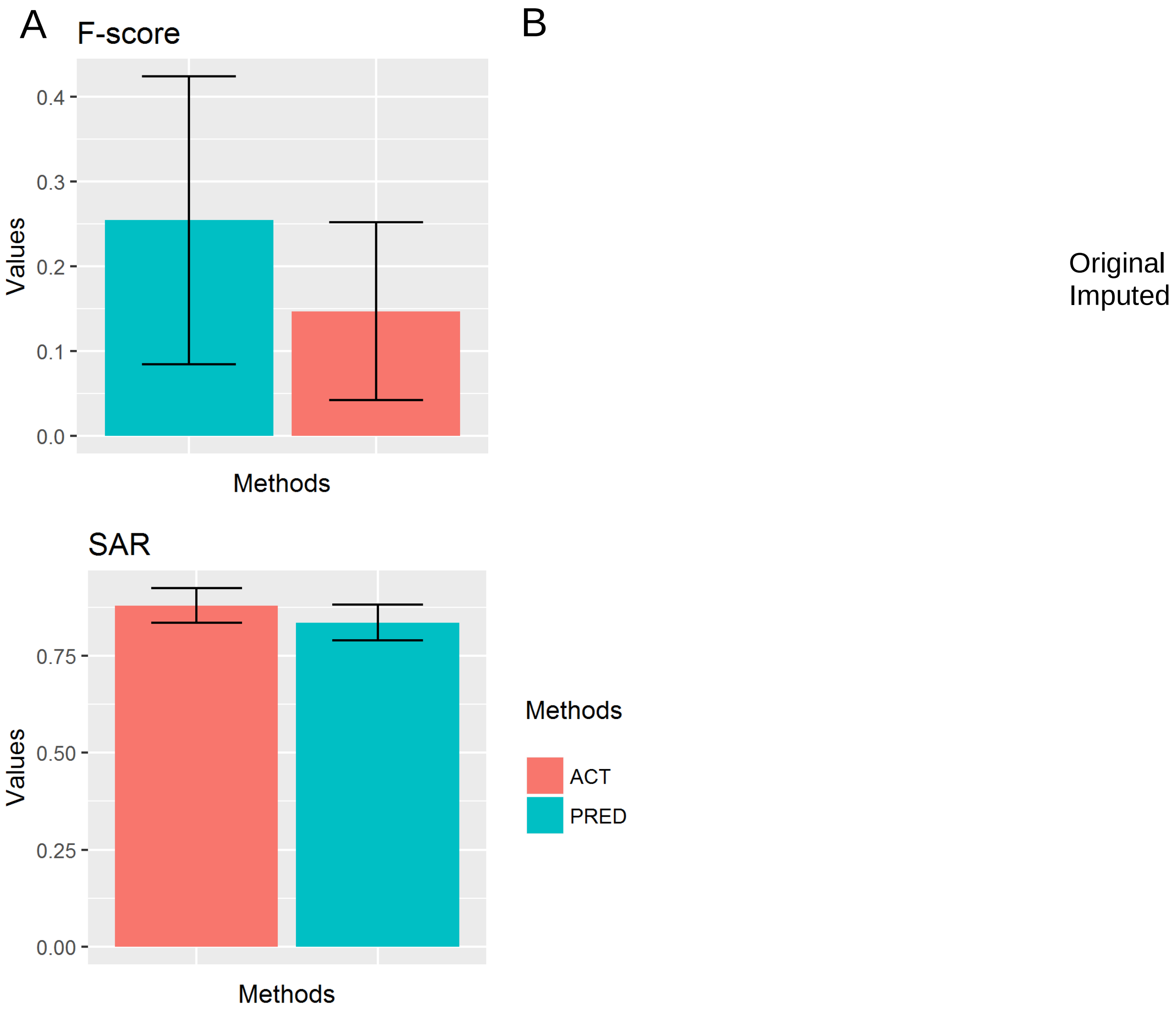


Figure 3.14 Mean performance scores of SPVAR on original and spline-imputed synthetic expression data.
(A) F-score, (B) SAR measure.

3.7 Conclusions

A framework for gene regulatory network inference was demonstrated and assessed using synthetic expression data in this chapter. The framework firstly involves converting a single-cell expression data into a pseudotime-series expression data by using DM-DPT. Then negative binomial GLM differential expression analysis is used to detect genes that are changing as a function of pseudotime. Once the differentially expressed genes are identified, splines are used to imputed expression values to reduce the effects of technical noise. The spline-imputed expression values of the differentially expressed genes can then be used for gene regulatory network inference using SPVAR.

Two key components of the framework, DM-DPT for pseudotime inference and SPVAR for network inference, were evaluated in comparison with other algorithms. DM-DPT has been shown to perform very well compared to other algorithms in both cell ordering and pseudotime inference when inferring robust single unbranched trajectory from single-cell RNAseq data. While SPVAR is not the best performing network inference algorithm, it is very conservative in predicting gene interactions and does not require user defined thresholds for extracting a gene network. The overall results suggest that the single-cell network inference framework performed well on synthetic expression data, and is a novel alternative for gene network inference using single-cell expression data.

3.8 Materials and methods

3.8.1 Synthetic data for the evaluation of pseudotime trajectory inference and gene regulatory network inference algorithms

The synthetic data used for the performance assessment of both pseudotime trajectory inference and gene regulatory network inference algorithms were generated using GeneNetWeaver version 3.1.2 (Schaffter et al. 2011). The synthetic networks were extracted from the gene regulatory network of yeast. The expression data were generated using ordinary and stochastic differential equations based on the synthetic networks. A coefficient of 0.05 was used for noise term in the stochastic differential equations. Note that the expression values generated by GeneNetWeaver range between 0 and 1.

The transiently perturbed time-series expression data generated from GeneNetWeaver were used for assessing the performance of the pseudotime inference algorithms and the entire single-cell network inference framework, with pseudotime differential expression analysis and spline imputation included. A total of six sets of synthetic networks and expression data were generated, in which three sets consist of 1000 genes and the other three sets consist of 3000 genes. For each set of data, there are multiple genes that are transiently upregulated or downregulated as the time progresses.

For comparing the network inference algorithms, the knockout time-series expression data generated from GeneNetWeaver were used. A total of 10 sets of synthetic networks and expression data were generated, in which five sets consist of 10 genes and the other five sets consist of 20 genes. For each set of data, there is a separate expression data for each gene in which the gene was knocked out.

The synthetic expression data as generated by GeneNetWeaver is used as original synthetic expression data. To introduce technical noise that is specific to single-cell RNAseq, overdispersion and zero-inflation noises were added into the synthetic expression data, which is used as the single-cell synthetic expression data. Overdispersion was introduced into the data by sampling the increase in expression values δ_{OD} with a probability of p_{OD}. The general assumption is that the higher a single expression value, the more likely it is to be inflated in values due to overdispersion. An exponential distribution f_{exp} is used to provide a continuous approximation to Poisson distribution to model the additional overdispersed component.

$$p_{OD} = \frac{2^{ax}}{2^{a}}$$

$$\delta_{OD} = \min\ (f_{exp}(bp_{OD}), c)$$

Where x is the expression value, and a, b, c are constants set at 5, 20 and 1 respectively.

Zero-inflation noise was introduced into the data by sampling from the probability of an expression value being a drop-out, p_D, and setting the expression value to 0. The general assumption is that the lower a single expression value, the more likely it is to be drop-out from the sequencing.

$$p_D\ =\ 2^{-ax}$$

Where x is the expression value, and a is a constant which is set to 3.

To simulate the numeric properties of RNAseq data, the synthetic expression data were converted into discrete values by multiplying each expression value with a constant of 10000.

In the case of synthetic expression data with missing cells, the missing cells were introduced in two ways. Firstly, each synthetic cell has a 10% uniform probability of being considered as missing, in order to simulate the phenomenon where the cells are not captured in an expression profiling experiment. Secondly, the synthetic cells were clustered into five groups using k-means clustering by their true cell orders. Cells belonging to the two intermediate groups (i.e. group 2 and 4 out of group 1 to 5) were considered as missing, in order to simulate the effects of sparse sampling in experiments which results in certain intervals of the expression trajectory not being captured by any cell.

3.8.2 Other pseudotime trajectory algorithms

DM-DPT was run with six other pseudotime inference algorithms on the same set of data. All algorithms were run using the recommended or default parameters. The algorithms were set to infer only a single trajectory without branching whenever the algorithms allow such constraint specifications. Note that some algorithms implicitly detect branches with no option or easy way to overcome this.

Monocle2 is implemented in the monocle R package (Trapnell et al. 2014). TSCAN is implemented in the TSCAN R package (Ji & Ji 2016). SCORPIUS is implemented in the SCORPIUS R package (Cannoodt, Saelens, Sichien, et al. 2016). SLICER is implemented in the SLICER R package (Welch et al. 2016). DPT is implemented in the destiny R package (Haghverdi et al. 2015; Haghverdi et al. 2016).

Both cell ordering and pseudotime errors calculated are the absolute differences between the true and inferred values for cell ordering and pseudotime respectively. Both true cell ordering and true pseudotime were obtained from the synthetic data generated from GeneNetWeaver.

3.8.3 Other gene regulatory network inference algorithms

SPVAR was run with four other network inference algorithms on the same set of data. The algorithms are CLR, GENIE3, TIGRESS and EBDBNet, and were run using the recommended or default parameters. CLR used is implemented in the minet R package (Meyer et al. 2008). GENIE3 used is as implemented by the authors in R scripts (Huynh-Thu et al. 2016). TIGRESS used is implemented in the metanetwork R package (Logsdon 2016). EBDBNet used is implemented in the EBDBNet R package (Rau 2016).

The results of gene regulatory network inference algorithms were assessed using several criteria, which include the area under the curve (AUC) for receiver operating characteristic

(ROC), the AUC for precision-recall (PR), F-score, accuracy, root mean squared error (RMSE) and the SAR measure. The F-score is a measure based on PR and is defined as described in Section 2.2.7. The SAR measure is a measure based on ROC and is defined as $SAR = \frac{1}{3} \times (Accuracy + AUC\ ROC + RMSE)$. The ROCR R package was used to calculate these measures (Sing et al. 2005).

3.8.4 Differentially expressed genes along the pseudotime trajectory

The pseudotime trajectory was inferred as described in Section 3.2.2. Once the pseudotime trajectory is inferred, the differentially expressed genes along the pseudotime trajectory were identified by using either the Spearman rank correlation or the negative binomial generalised linear model (GLM) as implemented by Monocle2 R package (Trapnell et al. 2014).

For the assessment of performance between the Spearman rank correlation and negative binomial GLM pseudotime differential expression analysis methods, the synthetic expression data used were as described in Section 3.8.1. The true differentially expressed genes were defined as the genes which expressions are perturbed externally and the genes which are immediately downstream of the externally perturbed genes. The information is provided by the GeneNetWeaver software, which is used to generate the synthetic expression data.

4 *iEsrrb*, *iKlf2* and *GY118F* transgene cell lines drive EpiSC reprogramming via different mechanisms

Results of this chapter are being used in preparing a manuscript for publication. RNAseq data used were generated by Hannah Stuart and Tim Lohoff from Jose Silva's lab as a collaboration.

4.1 Background

Stem cells are defined as cells that possess self-renewal capability and the ability to generate differentiated progeny (Lajtha 1979). There are three main types of stem cells, namely stem cells obtained from embryos, stem cells obtained from adults (i.e. adult stem cells), and induced stem cells generated from reprogramming (i.e. induced pluripotent stem cells) (Figure 4.1). Among the stem cells present in a developing embryo, there are embryonic stem cells (ESCs) that are derived from the inner cell mass, epiblast stem cells (EpiSCs) that are derived from post-implantation epiblast, and embryonic germ cells (EGCs) that are derived from primordial germ cells.

Among all the stem cells discussed here, epiblast stem cells (EpiSCs) represent a good system to study the naïve reprogramming process where a cell acquires the naïve stem cell identity (Nichols & Smith 2012). This is because EpiSC represents the primed pluripotent state, which is distinct but more similar to the naïve pluripotent state in embryonic stem cell (ESC) than other stem cells. By studying reprogramming in EpiSCs, this allows a closer inspection of the initiation of reprogramming without substantial influences from other biological processes such as differentiated cell-specific biological processes.

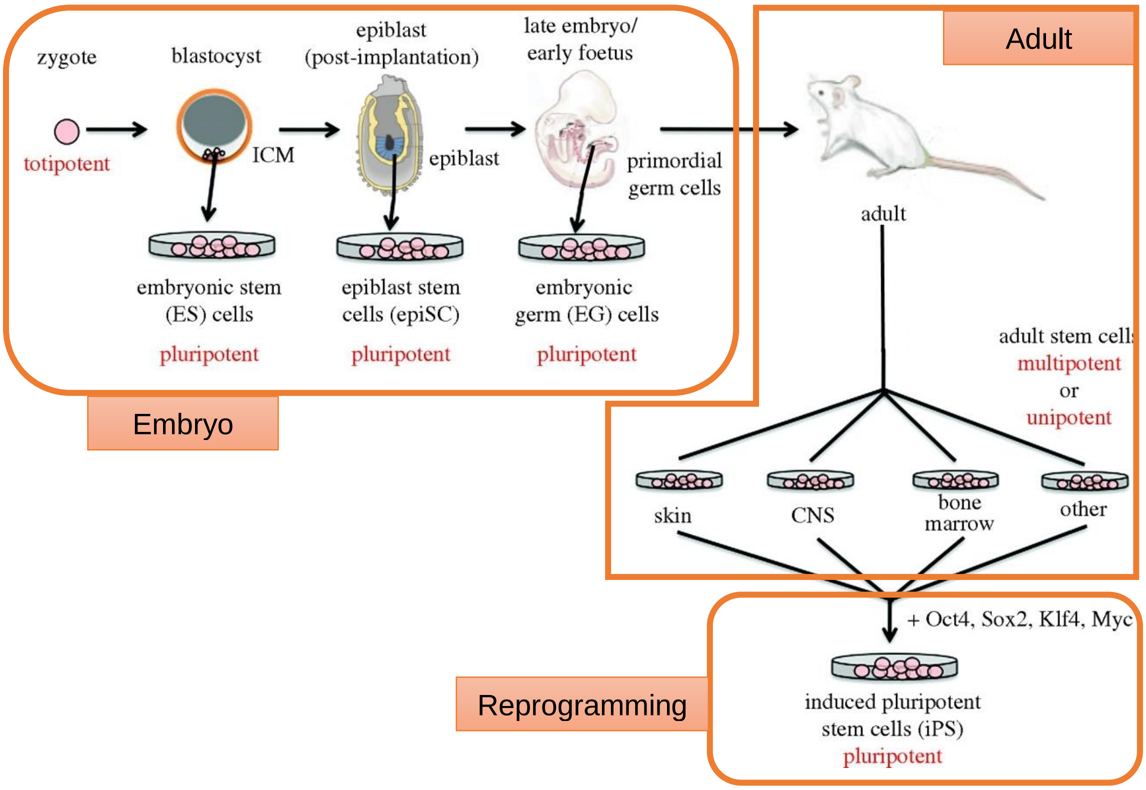


Figure 4.1 Different types of stem cells in mice.
[Figure adapted from (Watt & Driskell 2010; NIH n.d.; Staveley n.d.)*]*

Genetic factors, as introduced via transgenes, and environmental factors, as introduced via culture conditions, were known to be important for the successful reprogramming of cells into induced pluripotent stem cells (iPSCs) which possess naïve identity that is similar to ESCs (Yamanaka & Blau 2010; Robinton & Daley 2012). Both genetic and environmental factors result in gene expression changes in the transcriptional regulatory network responsible for establishing and maintaining naïve identity. The transcriptional regulatory network is known to contain key transcriptional regulator genes such as *Oct4*, *Sox2* and *Nanog*, which are widely studied and are known to be important for regulating ESC self-renewal and pluripotency (Nichols & Smith 2012).

In this study, we focus on studying the changes in transcriptomics along the reprogramming process due to the overexpression of three genes, *Esrrb*, *Klf2* and *GY118F*, in EpiSCs cultured under 2i-LIF condition. *GY118F* is a chimeric gene that leads to increased STAT3 protein phosphorylation and *Socs3* transcriptional activation. These three genes are involved in the naïve core transcriptional regulatory network, and are shown to be immediately downstream of the signalling pathways activated by 2i-LIF (Figure 4.2) (Hackett & Surani 2014). In addition, it has been shown that each of the individual transgenes is able to reprogram EpiSCs into iPSCs efficiently under 2i-LIF condition, where the identity of iPSCs was confirmed by colony formation assay and chimera formation (Hannah et al., unpublished).

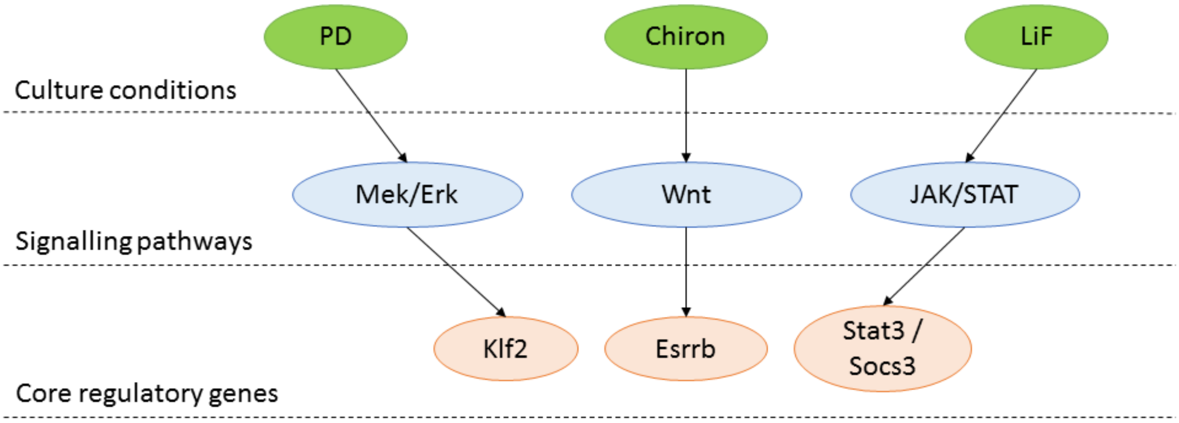


Figure 4.2 The genes studied are downstream of the signalling molecules used in the 2i+LiF culture condition.
PD and Chiron represent the 2i culture condition.

Esrrb is a transcription factor that is expressed in ESCs and is required for the self-renewal ability (Martello et al. 2012) and pluripotency (Festuccia et al. 2012) for ESCs. *Esrrb* is a class of nuclear receptors that can bind to DNA to activate transcription in the absence of exogenous ligand (Giguère 1999), and it is also a part of the pluripotency gene regulatory network (van den Berg et al. 2010; Chen et al. 2008). In addition, *Esrrb* is shown to be inhibited by *Tcf3*, which is itself is inhibited by Chiron, the GSK3 inhibitor used as part of the 2i ESC culture condition (Martello et al. 2012). Notably, the study shows that *Esrrb* overexpression can replace Chiron. In ESCs, the GSK3/TCF3 pathway is responsible for inducing differentiation of ESCs into EpiSCs (Berge et al. 2011; Wray et al. 2011). This is achieved by *Tcf3* acting as a transcriptional repressor that binds to the promoters of many pluripotency genes including *Nanog* and *Esrrb* (Martello et al. 2012).

Klf2 is a zinc-finger transcription factor that belongs to the Kruppel-like transcription factor family, and is known to regulate the proliferation and differentiation of many developmental processes including lung, blood and endothelial (McConnell & Yang 2010). *Klf2* has been used for the successful generation of iPSCs (Nakagawa et al. 2007). A recent study shows that *Klf2* is inhibited post-translationally by the Mek/Erk signalling pathway (Yeo et al. 2014). In ESCs, MEK/ERK pathway is responsible for initiating differentiation and lineage commitment (Kunath et al. 2007; Stavridis et al. 2007). The result of the study is interesting as the PD inhibitor, which is part of the 2i ESC culture condition, is previously known to affect MEK but the exact molecular mechanism was unknown. The inhibition of MEK stops the ERK2-mediated phosphodegradation of KLF2, which is critical for maintaining pluripotency.

GY118F is a chimeric LIF receptor that consists of human GCSF (granulocyte colony stimulating factor) receptor as the extracellular component, and GP130 signal-transducing component of the LIF receptor as the transmembrane and intracellular component (Niwa et al. 1998). In addition, the GP130 component contains a mutation that leads to specific activation of certain members of the JAK/STAT3 pathway when GCSF is present. In particular, GY118F cells stimulated with GCSF show increased STAT3 phosphorylation which leads to the transcriptional activation of its direct target *Socs3*. With wild type STAT3 protein, it is transiently activated via phosphorylation by associated kinases in response to specific cytokines and growth factors. Once phosphorylated, pSTAT3 protein mediates the expression of multiple genes important in biological processes such as differentiation and proliferation. As part of the larger STAT protein family, *Stat3* is widely studied in the context of cancer and development (Calò et al. 2003; Dorritie et al. 2014). In terms of reprogramming, increased activation of JAK/STAT3 pathway via the induction of GY118F is shown to promote the reprogramming of EpiSCs into iPSCs (Yang et al. 2010; van Oosten et al. 2012). Another study shows that JAK/STAT3 pathway may promote reprogramming by epigenetic regulation via inhibiting *Dnmts* and promoting the demethylation of *Oct4* and *Nanog* in mouse embryonic fibroblasts (Tang et al. 2012).

By studying *iEsrrb*, *iKlf2* and *GY118F* transgene cell lines (Figure 4.3), this chapter aims to investigate how the introduction of transgenes perturbs the expression levels of genes in the transcriptional regulatory network, and how the perturbed expressions contribute to the reprogramming of EpiSCs. Note that *iEsrrb* and *iKlf2* refer to induced *Esrrb* and induced *Klf2* respectively. Section 4.3 firstly describes the quality control and pre-processing of RNAseq data to reduce technical bias for downstream analyses. In Section 4.2, the differences in transcriptomics profiles among the three cell lines are discussed in details using bulk RNAseq. Section 4.4 then uses single-cell RNAseq to dissect the transcriptomics differences within the cell lines, as well as the changes in transcriptomes along pseudotime trajectory. The chapter then ends with conclusions in Section 4.5, and materials and methods in Section 4.6.

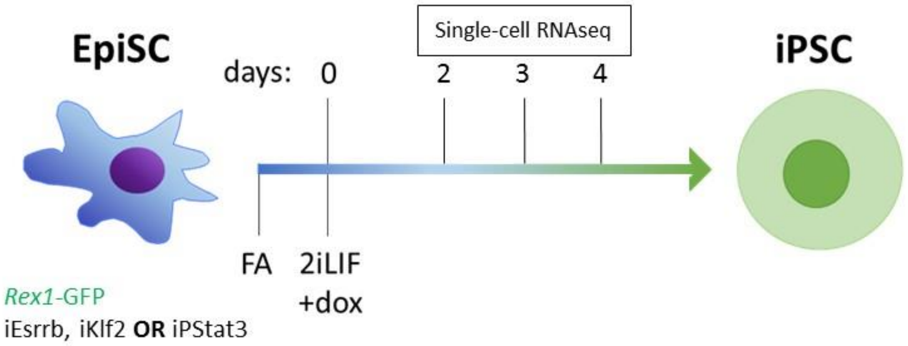


Figure 4.3 Overview of EpiSC reprogramming.
The brackets indicate cells were collected from day 2, 3 and 4 for single-cell RNA sequencing.

4.2 Bulk RNAseq shows that *iEsrrb*, *iKlf2* and *GY118F* transgene cell lines drive reprogramming via different routes

As an initial step to investigate the mechanism of reprogramming EpiSC into iPSC, bulk RNAseq data were generated for the three cell lines each with a separate transgene respectively (i.e. *iEssrb*, *iPStat3*, *iKlf2*) at multiple time points (Section 4.6.1). The time points start from 0 hour in EpiSC up to the end of reprogramming in iPSC, with intermediate time points at 1, 3, 6, 12, 24 (1 day), 48 (2 days), 72 (3 days), 96 (4 days) and 120 hours (5 days). For the sake of analysis, iPSCs were given a time point of 168 hours (7 days) because iPSCs in these cell lines have mostly finished reprogramming at 168 hours, as indicated by *Rex1* GFP reporter. *Rex1* is a good marker of naïve pluripotency, as it is expressed specifically in the naïve undifferentiated pluripotent cells and is downregulated very quickly at the beginning of differentiation (Toyooka et al. 2008). In addition to the *Rex1* GFP reporter, the pluripotency of iPSCs was verified through colony formation assay and chimera formation (Hannah et al., unpublished).

The bulk RNAseq data was analysed firstly by principal component analysis (PCA) and diffusion map (DM) (Figure 4.4). Both PCA and DM analyses had similar results, with DM showing less variations within each cell line (Figure 4.4). In both PCA and DM, the first component, which explains the most variations in the data, separated the samples by the degree of reprogramming of EpiSC into iPSC. The second component, which explains the second most variations in the data, separates the samples by the differences caused by the three different transgenes. The PCA loadings plot of top 20 genes with the highest loadings shows that the key signatures in PCA are mostly dominated by genes unique to iKlf2 cell line, with a smaller set of genes that are unique to the reprogrammed cells.

Among the top 20 genes, *Zfp42*, also known as *Rex1*, is the naïve pluripotency state marker used in the cell lines of this study. *Dppa5a* is a known pluripotency gene (Tanaka et al. 2002), *CrxOS* is a known self-renewal gene (Saito et al. 2009), while *Calcoco2* is shown to be associated with *Oct4* and is expressed in early embryonic tissues (Bortvin et al. 2003). The rest of the genes are likely to be associated with *iKlf2* cell line due to the high contribution to separation in PC2 of the PCA. By observing the PCA and DM results, it is clear that the induction of a single transgene is able to efficiently drive the reprogramming of EpiSC into iPSC, albeit with differences in the reprogramming paths taken by each transgene. It is hypothesised that the differences among transgenes are driven by differences both in the kinetics and the key genes involved.

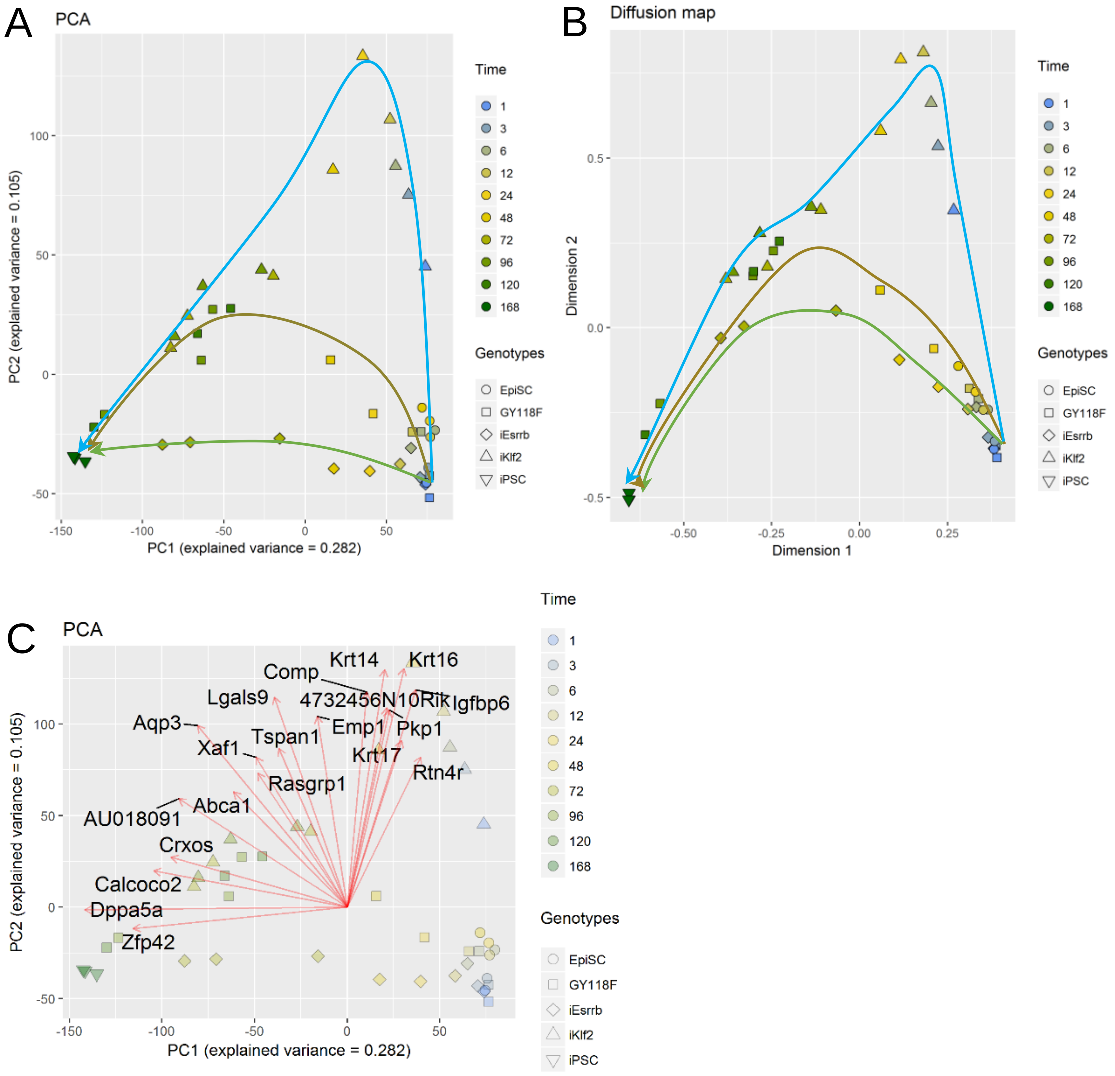


Figure 4.4 Similarities in expression profiles among reprogramming cells.
(A) Principal component analysis, (B) Diffusion map results and (C) PCA loadings. PCA loadings plot shows the top 20 genes with the highest absolute loadings value with both PC1

and PC2 combined. Each line indicates the reprogramming trajectory for each cell line from EpiSC to iPSC. Green line, iEsrrb; Brown line, GY118F; Blue line, iKlf2.

In order to investigate the differences between each transgene-driven reprogramming with respect to both EpiSCs and iPSCs, pairwise differential expression analyses were performed on the samples by comparing each transgene intermediates with both EpiSCs and iPSCs. The differentially expressed genes were combined from all pairwise differential expression analyses, which give a total of 1213 non-overlapping set of genes. The non-overlapping set of genes was then used for clustering the samples to visualise the relationships among the cell lines (Figure 4.5). The dendrogram in Figure 4.5 showed that at the early reprogramming stage, *iEsrrb* and *GY118F* cell lines were very similar to EpiSCs. *iKlf2* cell line showed the most distinct gene expression profile with 959 differentially expressed genes, as shown by the isolated *iKlf2* cluster and by the huge separation of *iKlf2* trajectory in the PCA (Figure 4.4). At the later reprogramming stage, *GY118F* was the most similar to iPSCs as shown by the dendrogram. However, this may be due to differences in the time points taken for each cell line. It should be noted that the time points taken for later reprogramming stage (day 2 and above) are different for each cell line, with time points taken respectively at day 2 and 3 for *iEsrrb*, day 3 and 4 for *iKlf2*, and day 4 and 5 for *GY118F*. Despite the time point differences, *iKlf2* cell line remained very different from other cell lines.

Gene Ontology (GO) analysis was then performed using the differentially expressed genes to investigate the differences in the biological functions of the three cell lines driven by *Esrrb*, *GY118F* and *Klf2* (Table 4.1 & Table 4.2). The enriched GO biological processes were different among cell lines, which suggest that the three transgenes were driving reprogramming in different ways.

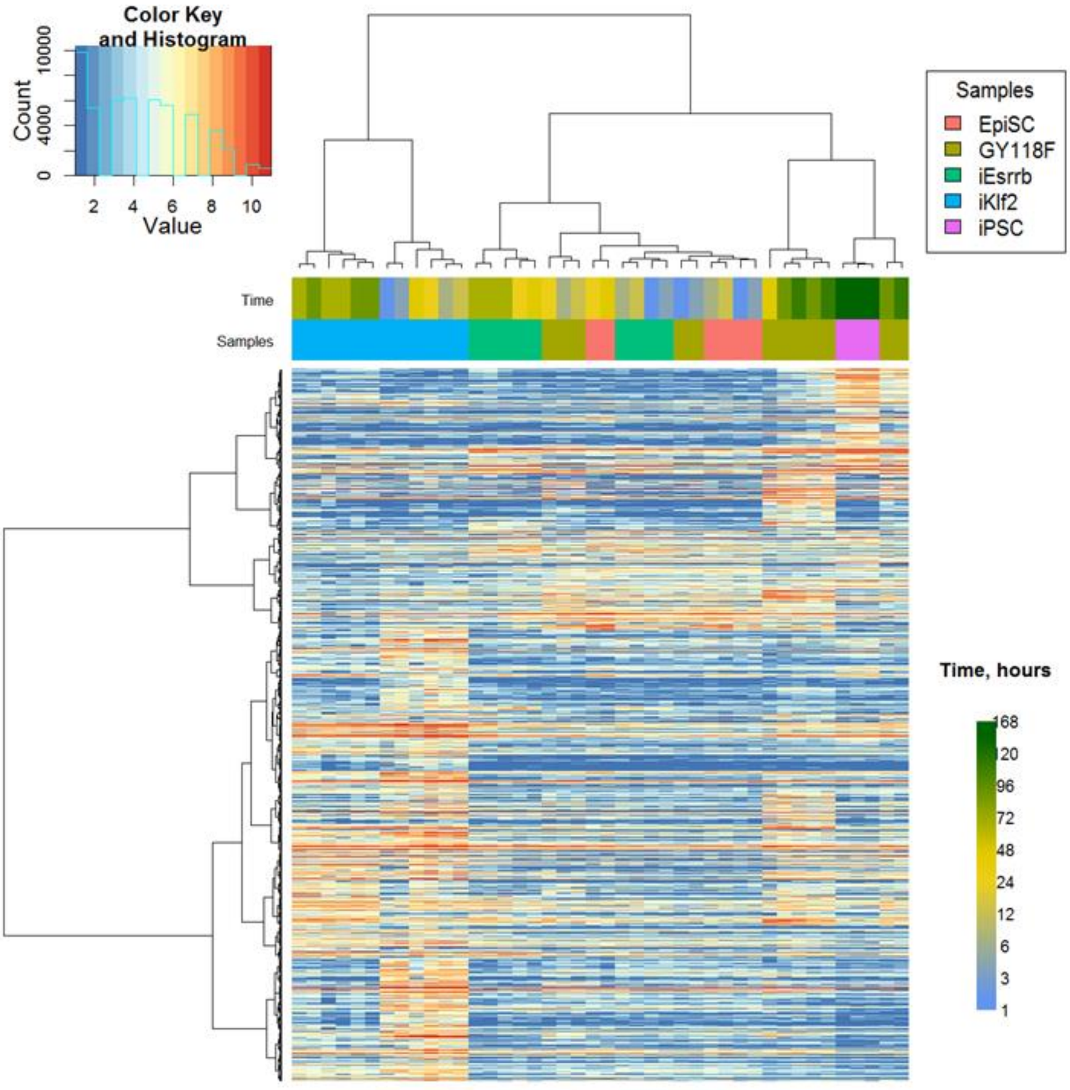


Figure 4.5 Heatmap of differentially expressed genes across all cell lines.
The non-overlapping set of differentially expressed genes across all cell lines with respect to both EpiSCs and iPSCs was used for this heatmap.

4.2.1 iEsrrb cell line reprograms by modulating transcriptional regulation responsible for establishing naïve ESC identity

Esrrb is a transcription factor which is important for the self-renewal and pluripotency of ESC. As expected, the results show that the top upregulated biological process in *iEsrrb* cell line is transcriptional regulation (Table 3.1). This suggests that *Esrrb* is likely to interact with other transcription factors during the reprogramming process. One of the genes upregulated in *iEsrrb* cell line and is involved in transcriptional regulation is *Gata6*, which is important for development and has been shown previously to be activated directly by *Esrrb* (Uranishi et al. 2016). Another transcription regulator that is upregulated by *Esrrb* is *Otx2*, which is a transcription factor that regulates the transition of naïve ESCs into primed EpiSCs (Acampora et al. 2013). Upregulation of *Otx2* activates the expression of FGF proteins and lowers the formation efficiency of chimeric embryos. In contrast, the downregulated biological processes in *iEsrrb* cell line may reflect the inhibition of differentiation as indicated by the cell type specific biological processes, cell-cell adhesion and signalling pathways (Table 4.2).

Taken together, it is likely that the *iEsrrb* cell line achieved reprogramming via directly modulating the expression of key genes important for the naïve ESC identity. In comparison to other cell lines, *iEsrrb* cell line has fewer differentially expressed genes and is more similar to EpiSCs and iPSCs as shown in the PCA (Figure 4.4) and the dendrogram (Figure 4.5). Note that the dendrogram also shows that *iEsrrb* in later time points as being more different from iPSCs than *GY118F*. This is because the time points taken for *GY118F* (day 4 & 5) are later than for *iEsrrb* (day 2 & 3).

4.2.2 GY118F cell line reprograms by regaining trophectoderm potential and downregulating BMP/SMAD pathway

As for *GY118F* cell line, the top upregulated biological processes are mostly related to developmental processes such as placenta development (Table 4.1). This suggests that *GY118F* cell line may be regaining the potential to give rise to trophoblast cells, which in turn contribute to placenta. Among the development related genes, many are transcription factors, such as *Cebpa*, *Gata2* and *Gata3*. A study on granulocyte development shows that the activation of STAT3 via GCSF leads to the upregulation of *Cebpa* expression, and activated STAT3 enhances the transcriptional activity of C/EBPA by binding to C/EBPA (Numata et al. 2005). *Cebpa* has been shown to enhance the reprogramming efficiency of B cells into iPSCs by post-transcriptionally enhancing the abundance of many important proteins for reprogramming such as *Lsd1* and *Brd4*. In embryonic development, both *Gata2* and *Gata3* have been shown to regulate trophoblast development (Ray et al. 2009) and later on blood development (Tsai et al. 1994; Pandolfi et al. 1995). Although *Gata2* or *Gata3* alone is not crucial for trophoblast development, embryos exhibit lethality if both *Gata2* and *Gata3* are knocked out (Home et al. 2017).

The top downregulated biological processes in *GY118F* cell line include BMP signalling pathway and SMAD family transcription factor, of which the enriched genes are *Bmp7*, *Fgf8*, *Nodal* and *T* (Table 4.2). BMP signalling pathway is important for many developmental processes, and it can act via both SMAD-dependent and independent pathways (Miyazono et al. 2010). In SMAD-dependent pathway, BMP ligands bind to TGF-beta receptors, which in turn activates SMAD family transcription factor. Studies have shown that the BMP signalling pathway is important in pre-implantation development (Graham et al. 2014; Papanayotou & Collignon 2014; Reyes de Mochel et al. 2015) as well as during gastrulation (Mishina et al. 1995; Arnold et al. 2006) in mouse embryo. Interestingly, BMP/SMAD pathway has been shown to be dispensable for maintaining naïve pluripotency, as BMPs act via non-SMAD

MEK5/ERK5 pathway in ESCs (Morikawa et al. 2016). Taken together, *GY118F* cell line may achieve reprogramming by regaining trophectoderm potential and downregulating BMP/SMAD pathway that may induce differentiation.

4.2.3 iKlf2 cell line reprograms by regulating cell surface proteins, cell proliferation and cell differentiation

In *iKlf2* cell line, the top upregulated biological processes are developmental processes, slower cell proliferation and membrane transport (Table 4.1). Many genes are upregulated in *iKlf2* cell line compared to other cell lines, which include many transcription factors and signalling proteins such as *Gata1*, *Bmp4* and *Notch1*. Cell cycle has been shown to be important in regulating proliferation and differentiation in the ESCs (Pauklin & Vallier 2013). Complete suppression of cell cycle is shown to trigger differentiation (Li & Kirschner 2014), but slow proliferation rate promotes reprogramming of fibroblasts (Xu et al. 2013). It is likely that a fine balance in cell cycle regulation needs to be achieved for optimal reprogramming efficiency. The upregulated membrane transport functions are particularly interesting, as they are less studied than developmental processes. A study which profiles the membrane proteins of ESCs, iPSCs and fibroblasts shows that there are pluripotency-associated membrane proteins, as well as a small subset of membrane proteins that differ between ESCs and iPSCs (Hao et al. 2013). The increase in membrane transport activities may reflect increased metabolism or signalling activities required for reprogramming.

The top downregulated biological processes in *iKlf2* cell line are differentiation, cell-cell adhesion and male meiosis (Table 4.2). Among the developmental-related processes, the most significant process is the inhibition of epithelial cell differentiation, potentially suggesting that the cells were progressing towards increasingly naïve pluripotent state that exhibits less epithelial properties. Cell-cell adhesion proteins, which often work together with signalling pathways, are known to be important for ESC maintenance (Pieters & van Roy 2014). *Fzd7* and *Fzd8*, which are part of the Wnt signalling pathway, as well as *Bmp7* and *Smad7*, which are part of the BMP/SMAD signalling pathway, were downregulated in the *iKlf2* cell line. Interestingly, male meiosis was detected as a significant process, which may be related to the changes on chromosome X due to reprogramming (Stadtfeld et al. 2008) of which *Klf2* has been shown to play a role in (Gillich et al. 2012). Taken together, the induction of *Klf2* exert a strong effect on the global expression profile when compared to other cell lines. In particular, slower cell proliferation, the upregulation of membrane transport functions, as well as the downregulation of epithelial cell differentiation and cell-cell adhesion support the notion that the *iKlf2* cell line was being reprogrammed into naïve pluripotency state.

The results of the bulk RNAseq data clearly suggest that the three transgenes drive reprogramming in different ways, which involve the activation of different biological processes. However, although the cell populations investigated in the bulk RNAseq were enriched for high reprogramming efficiency by gating for high REX1-GFP level, there is always a proportion of cells within the cell populations that will not be reprogrammed. Therefore, the results of the bulk RNAseq may be confounded by the cells that were not undergoing reprogramming. In order to investigate this further, single-cell RNAseq experiments were performed on the three cell lines driven by *iKlf2*, *iEssrb* and *GY118F* respectively.

Biological Process	**Num. of Annotated Genes**	**Num. of Observed Genes**	**Adjusted P-values**
***Esrrb* (Total: 46 upregulated genes)**			
Positive regulation of transcription from RNA polymerase II promoter	1008	8	0.00048
Cellular response to pH	20	2	0.00065
Positive regulation of cardiac muscle cell proliferation	21	2	0.00071
Regulation of transforming growth factor beta production	22	2	0.00078
Positive regulation of cell cycle arrest	29	2	0.00137
Positive regulation of histone methylation	30	2	0.00146
Cell fate determination	41	2	0.00272
Positive regulation of mitotic nuclear division	47	2	0.00356
Midbrain development	47	2	0.00356
Negative regulation of cytokine secretion	48	2	0.00371
***GY118F* (Total: 208 upregulated genes)**			
Cellular response to interferon-beta	39	7	8.00E-08
Embryonic placenta development	106	8	5.00E-06
Regulation of keratinocyte differentiation	29	5	7.70E-06
Tissue development	1690	43	7.80E-06
Positive regulation of osteoblast differentiation	65	6	3.80E-05
Regulation of water loss via skin	22	4	5.30E-05
Defense response	1210	42	6.20E-05
Cytokine-mediated signaling pathway	268	12	9.70E-05
Regulation of embryonic development	114	6	0.00042
Regulation of transcription regulatory region DNA binding	39	4	0.00052
***Klf2* (Total: 899 upregulated genes)**			
Inner ear morphogenesis	103	15	9.60E-07
Regulation of ion transmembrane transport	363	32	3.10E-06
Negative regulation of cell proliferation	594	45	4.10E-06
Angiogenesis	420	38	2.30E-05
Regulation of vascular permeability	27	7	3.70E-05
Regulation of embryonic development	114	14	4.20E-05
Peptide cross-linking	48	9	4.80E-05
Calcium ion transmembrane transport	216	22	6.30E-05
Skin development	254	34	6.40E-05
Positive regulation of amine transport	31	7	9.70E-05

Table 4.1 Top 10 upregulated biological processes in each cell line based on Gene Ontology (GO) analysis.

Biological Process	**Num. of Annotated Genes**	**Num. of Observed Genes**	**Adjusted P-values**
***Esrrb* (Total: 108 downregulated genes)**			
Platelet-derived growth factor receptor signaling pathway	51	5	4.10E-06
Leukocyte adhesion to vascular endothelial cell	24	3	6.40E-05
Positive regulation of phosphatidylinositol 3-kinase signaling	61	4	0.00019
Negative regulation of endothelial cell apoptotic process	27	3	0.00027
Cell-cell signaling	1203	7	0.00053
Positive regulation of leukocyte cell-cell adhesion	166	3	0.00095
Positive regulation of MAP kinase activity	169	4	0.001
Positive regulation of NF-kappaB transcription factor activity	105	4	0.00152
Negative regulation of inflammatory response	113	4	0.00199
Positive regulation of DNA replication	54	3	0.00209
***GY118F* (Total: 86 downregulated genes)**			
BMP signalling pathway	144	5	2.00E-05
Regulation of animal organ formation	38	3	0.00059
SMAD protein signal transduction	70	3	0.00084
Neuroepithelial cell differentiation	61	3	0.00122
Skeletal muscle adaptation	21	2	0.0014
Pharyngeal system development	22	2	0.00153
Male genitalia development	22	2	0.00153
Glomerulus vasculature development	22	2	0.00153
Signal transduction involved in regulation of gene expression	23	2	0.00168
Branching involved in salivary gland morphogenesis	28	2	0.00249
***Klf2* (Total: 447 downregulated genes)**			
Positive regulation of epithelial cell differentiation	53	5	0.00062
Positive regulation of cell development	527	16	0.00121
Single organismal cell-cell adhesion	669	22	0.00163
Synaptonemal complex assembly	20	3	0.00211
Male meiosis	43	4	0.00233
Ventricular cardiac muscle tissue morphogenesis	48	4	0.00349
Positive regulation of leukocyte migration	117	7	0.00349

Sulfur compound biosynthetic process	80	5	0.00391
Pyruvate metabolic process	93	4	0.00506
Positive regulation of potassium ion transmembrane transport	27	3	0.00507

Table 4.2 Top 10 downregulated biological processes in each cell line based on Gene Ontology (GO) analysis.

4.3 Pre-processing of RNAseq data

As a further investigation into the mechanism of reprogramming of EpiSC into iPSC, single-cell RNAseq data were generated for the three cell lines each with a separate transgene respectively (i.e. *iEssrb*, *GY118F*, *iKlf2*) at multiple time points. The time points start from 0 hour in EpiSC up to the end of reprogramming in iPSC, with intermediate time points at 48 (2 days), 72 (3 days), and 96 hours (4 days). As before, iPSCs were given a time point of 168 hours (7 days) because iPSCs in these cell lines have mostly finished reprogramming at 168 hours, as indicated by *Rex1* GFP reporter. The pluripotency of iPSCs was also verified through colony formation assay and chimera formation (Hannah et al., unpublished). The single-cell RNAseq data contain 360 cells in total.

Before any analysis is performed, it is important to determine the quality of the RNAseq data obtained, and to pre-process the data such that downstream analyses are not biased by technical differences in the data. The steps involved are similar for both bulk and single-cell RNAseq, although single-cell RNAseq requires less stringent thresholds as the data is noisier than bulk RNAseq. In the following subsections, the pre-processing results performed on the single-cell RNAseq data analysed in Section 4.4 are discussed. The pre-processing steps for bulk RNAseq data analysed in Section 4.2 gave similar results, therefore are not further discussed here.

4.3.1 Performing quality control

Quality controls can be performed on sequencing data at three levels, i.e. on the raw sequencing reads, on the aligned reads, and on the counted reads. Performing quality control on the raw sequencing reads allow us to assess the quality of the sequencing process and any potential problems with the sequencing library. The FastQC program was used for the quality control of raw sequencing reads (Andrews S 2010). The quality control performed using FastQC on the single-cell RNAseq data showed that all samples have very similar technical properties and do not exhibit major technical problems.

The FastQC results for a representative iPSC sample from the single-cell RNAseq data was analysed and used to illustrate the general technical properties of the data in Figure 4.6. Figure 4.6A show that each base sequence is detected unambiguously by the sequencing machine, as indicated by the high Phred quality score for each base across the reads. The slight decrease of quality at the end of the reads is expected and is due to degradation of sequencing chemistry as the read length increases. Figure 4.6B show that generally almost all reads were sequenced at the highest quality and there is no subset of reads that were sequenced with a lower quality. Figure 4.6C show the sequence content across the reads, which agrees with the expectation that each nucleotide should have equal probability (i.e. 25%) of occurring at each base position. This is true except for the first few base positions, which always show an identical bias pattern in sequence content composition across samples due to the random primers used for sequencing. Figure 4.6D show that the distribution of the observed GC content in all reads roughly agrees with the distribution of the expected GC content, which suggests that the sequences were not contaminated. Figure 4.6E show the number of duplicated reads is quite high with about 63% duplicated sequences. This is an expected observation for an RNAseq experiment as it is common to greatly over-sequence the transcripts in order to detect lowly expressed transcripts. Figure 4.6F show that the Nextera transposase sequencing primers are present at a small proportion in the reads, with a cumulative percentage of less than 5% at the end of a read. There is usually no need to trim off the primer sequences in the reads manually by using read trimming programs such as cutadapt or Trimmomatic, as most modern sequence aligners can perform soft-clipping to ignore the primer sequences during the alignment process.

Performing quality control on the aligned reads gives further insights on the quality of the data. The RSeQC program was used for investigating the quality of alignments (Wang et al. 2012). RSeQC showed that all samples from the single-cell RNAseq data have very similar technical properties and do not exhibit major technical problems. The same sample that was used in Figure 4.6 was used here as well to illustrate the results from RSeQC in Figure 4.7.

Figure 4.7A show that the aligned reads covered the gene body well in general, with only a slight expected bias for the 3’ end due to poly-A tail reverse transcription. The single-cell RNAseq data generally achieved lower gene body coverage than bulk RNAseq data, likely to be due to much lower amount of starting RNA materials. Figure 4.7B show properties of the clipping profile, which indicates the bases of reads that have been masked or soft-clipped and therefore not used in alignment of the reads. Figure 4.7C show the number of bases marked as deleted in the aligned reads, while Figure 4.7D show the number of bases marked as inserted in the aligned reads. Both deletion and insertion profiles show similar properties with higher number of insertion or deletion (indel) events inferred at the middle of the reads. Note that zero number of indel events were marked at both ends of the reads by default, because GSNAP aligner does not try to infer any indel events at the ends of the reads. Lastly, Figure 4.7E show the mutation profiles of the reads, which suggest the difference between the genomic sequence of the cell line used and the reference genomic sequence.

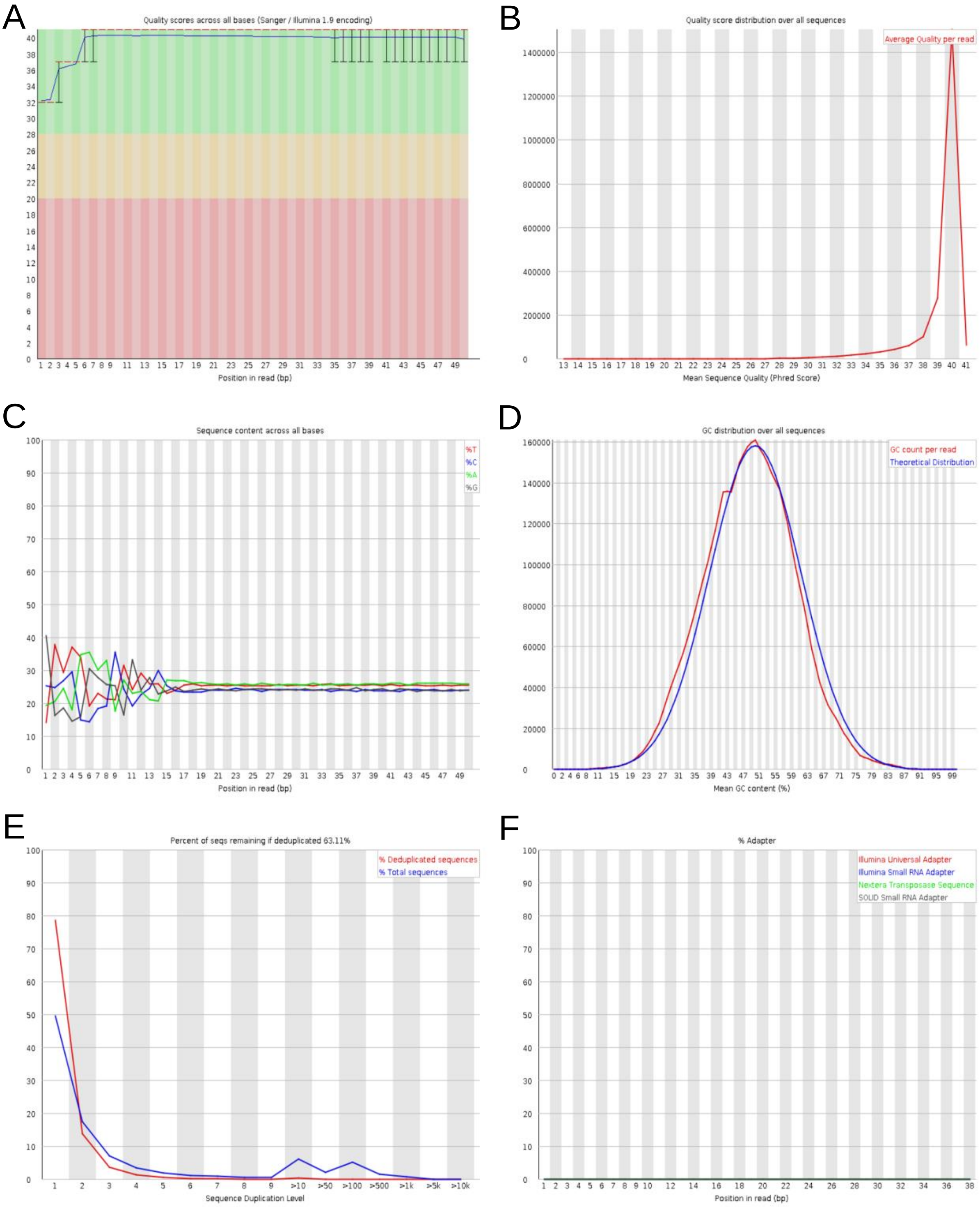


Figure 4.6   Quality control of raw sequencing reads of a single-cell RNAseq iPSC sample.
(A) Per base sequence quality, (B) Per sequence quality scores, (C) Per base sequence content, (D) Per sequence GC content, (E) Sequence duplication levels, (F) Adapter content.

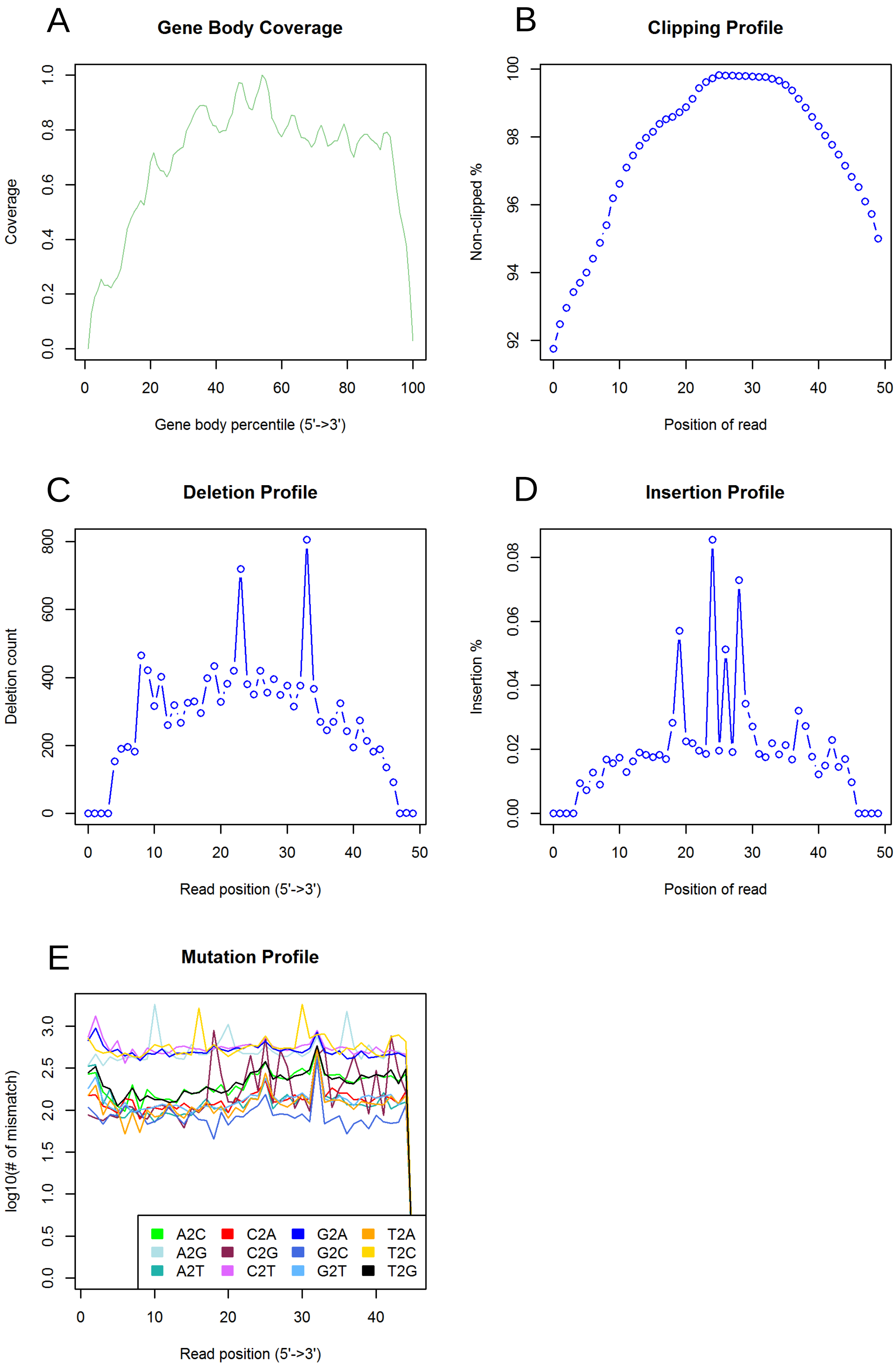


Figure 4.7 Quality control of aligned reads of a single-cell RNAseq iPSC sample. *(A) Gene body coverage, (B) Clipping profile, (C) Deletion profile, (D) Insertion profile, (E) Mutation profile.*

Lastly once the aligned reads are counted based on the genes that they aligned to, a final quality control should be performed on the counts of reads for each cell. This quality control is especially useful for filtering out outlier samples or cells that exhibit unusual technical properties, such as abnormally low number of total counts per cell or abnormally high proportions of ERCC control counts. Both observations indicate that the sample does not contain enough DNA materials to be sequenced, which may be due to the failure of PCR amplification or cell not being sorted into the well. Note that while the previously discussed quality controls on raw and aligned reads were not used to remove outlier cells, the outlier cells that may exhibit problematic properties in previous quality controls are very likely to exhibit differing technical properties from other cells in the quality control for the counted reads.

Figure 4.8 shows the technical properties of reads for all samples in the single-cell RNAseq experiment. In general, the number of total reads gives a good overview of sequencing depth. The fractions of mapped and unmapped reads indicate how well the alignment has performed. The fractions of spike-in, mitochondrial and other genes reads indicate the quality of the starting RNA materials. The fractions of no feature, ambiguous and low quality reads, which are calculated based on HTSEQ-count outputs, show the quality of the counted reads.

Five thresholds on the technical properties of reads were set to remove low quality outlier samples from downstream analysis. 24 cells were removed from downstream analysis based on the total reads (less than 0.5 million reads per cell), fraction of mapped reads (less than 45% mapped reads per cell), fraction of spike-in reads (more than 20% spike-in reads per cell), fraction of mitochondrial reads (more than 10% mitochondrial reads per cell) and number of genes with more than 10 reads per million (less than 3500 genes per cell). Note that the single-cell RNAseq consists of four 96-well plates, in which the 4th plate is known to possess batch effects due to being processed in a different facility, thereby contributing the highest proportion of outlier cells. However as most of the cells in each of the four plates were collected from different experimental conditions, batch effect correction for plate effects was not attempted as performing batch effect correction may reduce real biological effects.

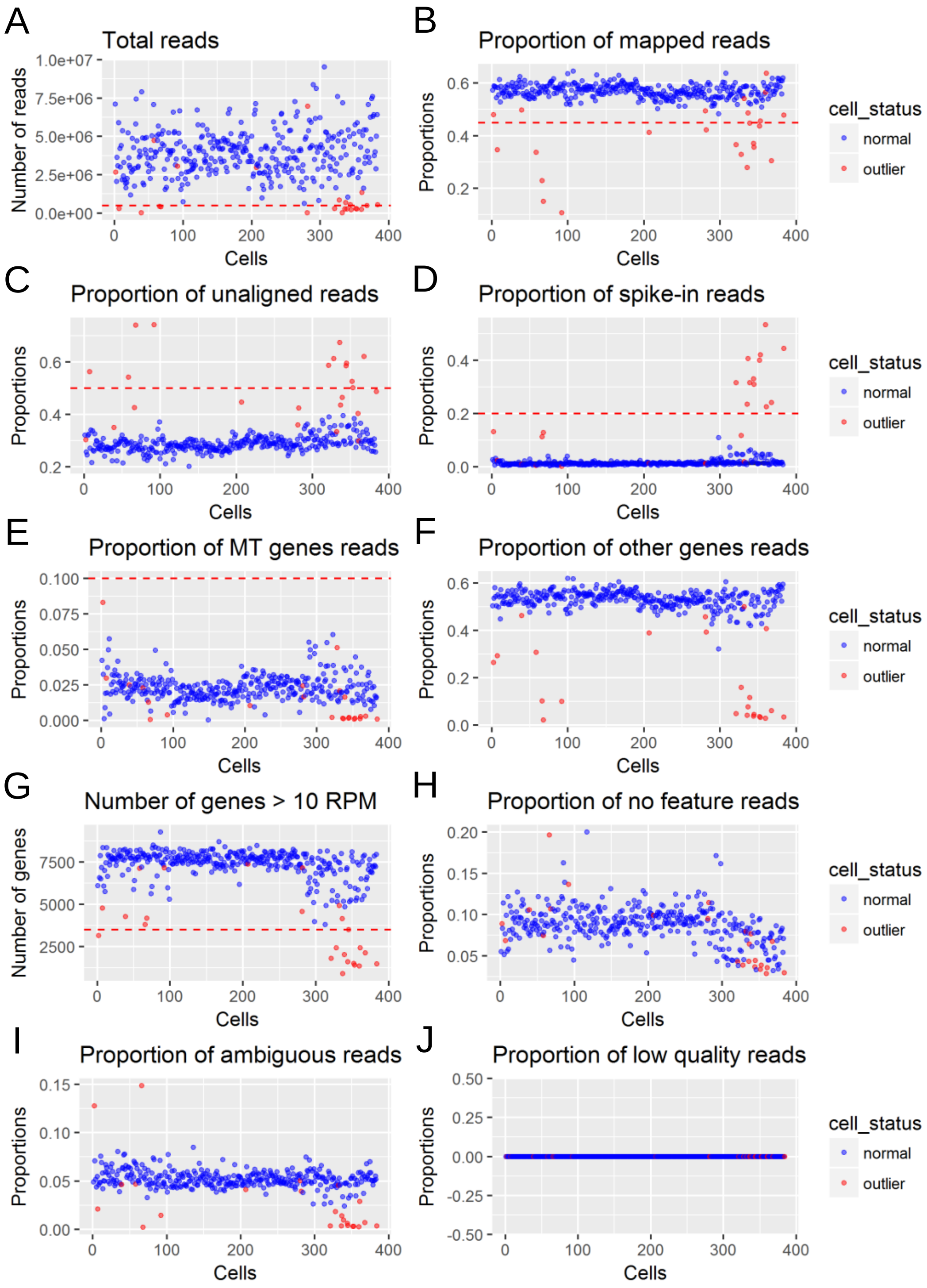


Figure 4.8 Quality control of counted reads in all cells.
Each dot in the plot corresponds to a cell. Each cell always has the same index in the x-axis across the plots. The red dashed line indicates the threshold used where a cell is labelled as an outlier.

4.3.2 Selecting the most suitable normalisation method

After quality control, the most important pre-processing step is the normalisation step. The main purpose of normalisation is to correct for differences in library sizes or sequencing depths (See Section 1.2.1). Due to the assumptions of different normalisation methods and the differences in the underlying technical properties of different RNAseq data, the best normalisation method may be different for different RNAseq data, especially in the context of single-cell RNAseq data. Here, the performance of three normalisation methods that are very different in their theoretical designs and assumptions were tested with this single-cell RNAseq data. The three normalisation methods tested are counts per million (CPM), DESeq (Anders & Huber 2010) and scran (L. Lun et al. 2016) normalisation methods.

CPM is the most conservative normalisation method which divide reads by total reads in each cell then multiplying by a million. Due to the strict assumption of CPM that all cells must have the same number of total normalised reads, it is possible that this method may skew gene expression values if there are some genes that are both very highly expressed and differentially expressed among the cells. DESeq normalisation is done by scaling each cell with a size factor which is the median across genes on the ratio of each gene expression value to the gene's geometric mean across cells. Note that DESeq normalisation is designed for bulk RNAseq data, and may fail for single-cell RNAseq due to the use of geometric mean which can only be calculated for genes with non-zero expression values across cells. Lastly, scran normalisation is designed specifically for single-cell RNAseq data to deal with the large number of drop-outs (i.e. zero expression values). Scran normalisation calculates normalisation factors on pooled cells by summing expression values across cells in a pool and divide by an average reference background value. The cells in each pool should ideally share similar expression profiles so as to not reduce the effects of differentially expressed genes among cell populations. As each cell is present in multiple pools, the normalisation factors on pooled cells can then be deconvolved into a separate normalisation factor for each cell.

Figure 4.9 showed the raw reads and reads normalised using the three methods with different settings for each sample. DESeq normalisation was tested using two different location estimators, namely the default median and the shorth estimator which is more suitable for low read counts. Scran normalisation was tested using different clustering methods, namely the default Spearman correlation-based hierarchical clustering and by just considering all the cells as a single cluster. The raw reads for each sample clearly show the need of normalisation, as the number of reads in each cell differ greatly both within and among samples (Figure 4.9).

The performance of normalisation methods was assessed using two main criteria, which are the principal component analysis (PCA) and cell-wise relative log expression (RLE). The ideal normalisation method should provide clear separation among samples in the PCA and give reduced spread in the RLE relative to the separation and the spread of the raw reads respectively. It can be seen from Figure 4.10 that normalisation methods have a significant effect on downstream analysis, such as PCA. The results of PCA were very different when reads were normalised using different methods. Reads normalised using both DESeq normalisation methods offered only marginal improvement when compared to the PCA of raw reads. Scran methods offer better PCA results in separating the cells, but it is unable to resolve the subgroup of iKlf2 cells that are very similar to EpiSCs and there is a higher degree of overlap among iKlf2, iEsrrb and GY118F cells in the intermediate states. In terms of PCA results, CPM normalisation gave the results with the clearest separation of samples and the result agrees the most with the PCA performed on bulk RNAseq (Figure 4.4).

Cell-wise RLE measure calculates the median log10 expression value for each gene, and then calculate the median of the resulting median log10 gene expression values for each cell. The calculation of RLE is similar to DESeq normalisation, and assumes that there are roughly equal number of gene upregulation and downregulation events across all cells. Therefore, a sample with normalised cells should have an RLE that is close to zero. Figure 4.11A shows that raw reads result in a huge spread of RLE values for the cells, while all normalisations reduced the spread of RLE values. Among all normalisation methods, CPM performed the best in terms of RLE by achieving the lowest spread of RLE values. Figure 4.11B shows that all normalisation methods roughly retain the variability among samples, although the degree of variability of each sample is skewed differently by different normalisation methods.

In summary, the results of PCA and cell-wise RLE show that CPM normalisation method is the best and most suitable for this dataset. In addition, there is no relationship between gene length and read count due to the 3’ bias of the sequencing library generation protocol (Figure 4.12). Therefore, there is no need to normalise read counts by gene lengths.

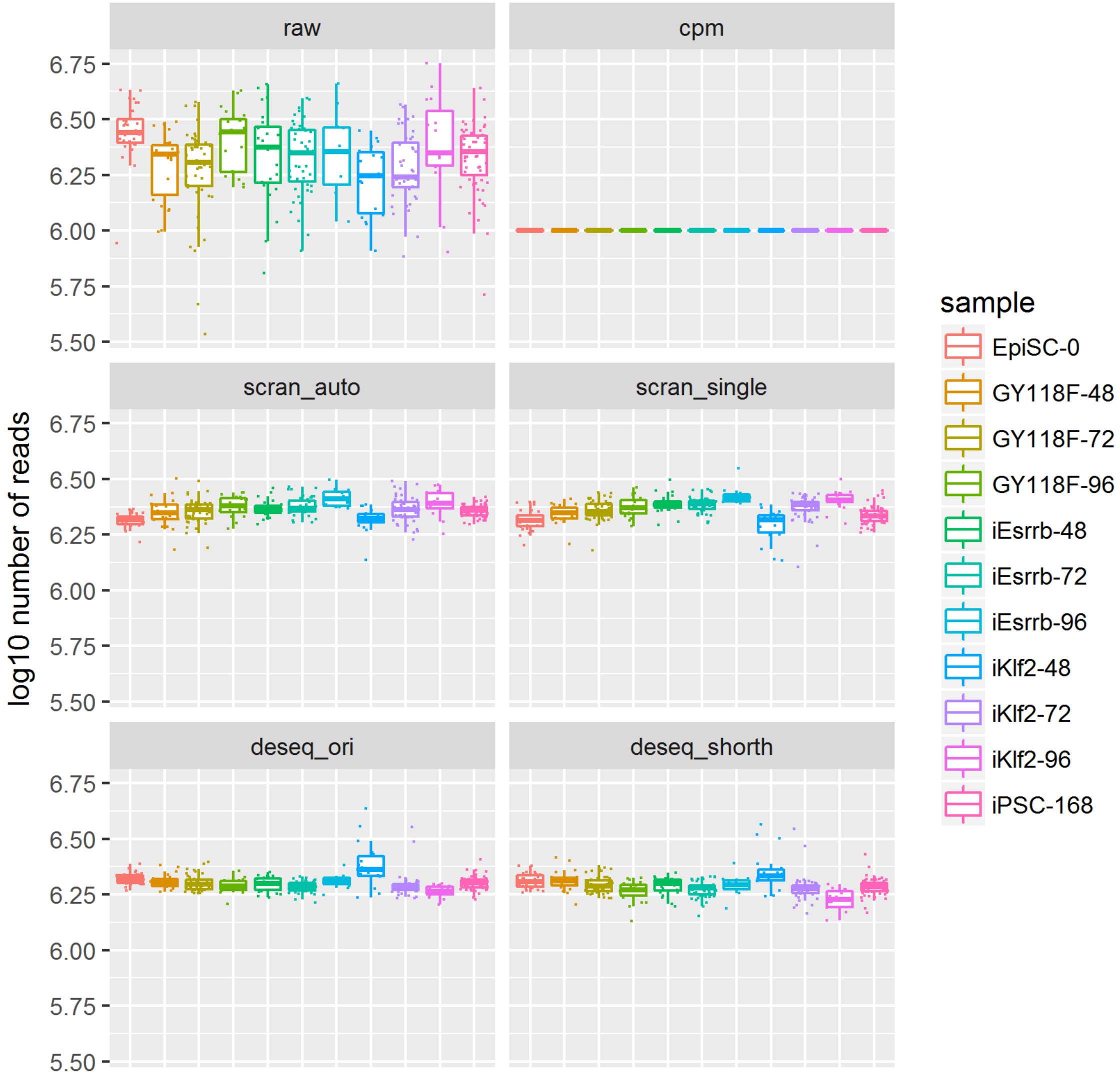


Figure 4.9 Log10 distributions of raw and normalised reads.
Y-axis scale is fixed to allow for comparisons among different methods. Sample names are of the following format, [cell line]-[hours]. CPM, counts per million; scran_auto, Scran with default clustering; scran_single, Scran with a single cluster; deseq_ori, DESeq with default median; deseq_shorth, DESeq with shorth estimator.

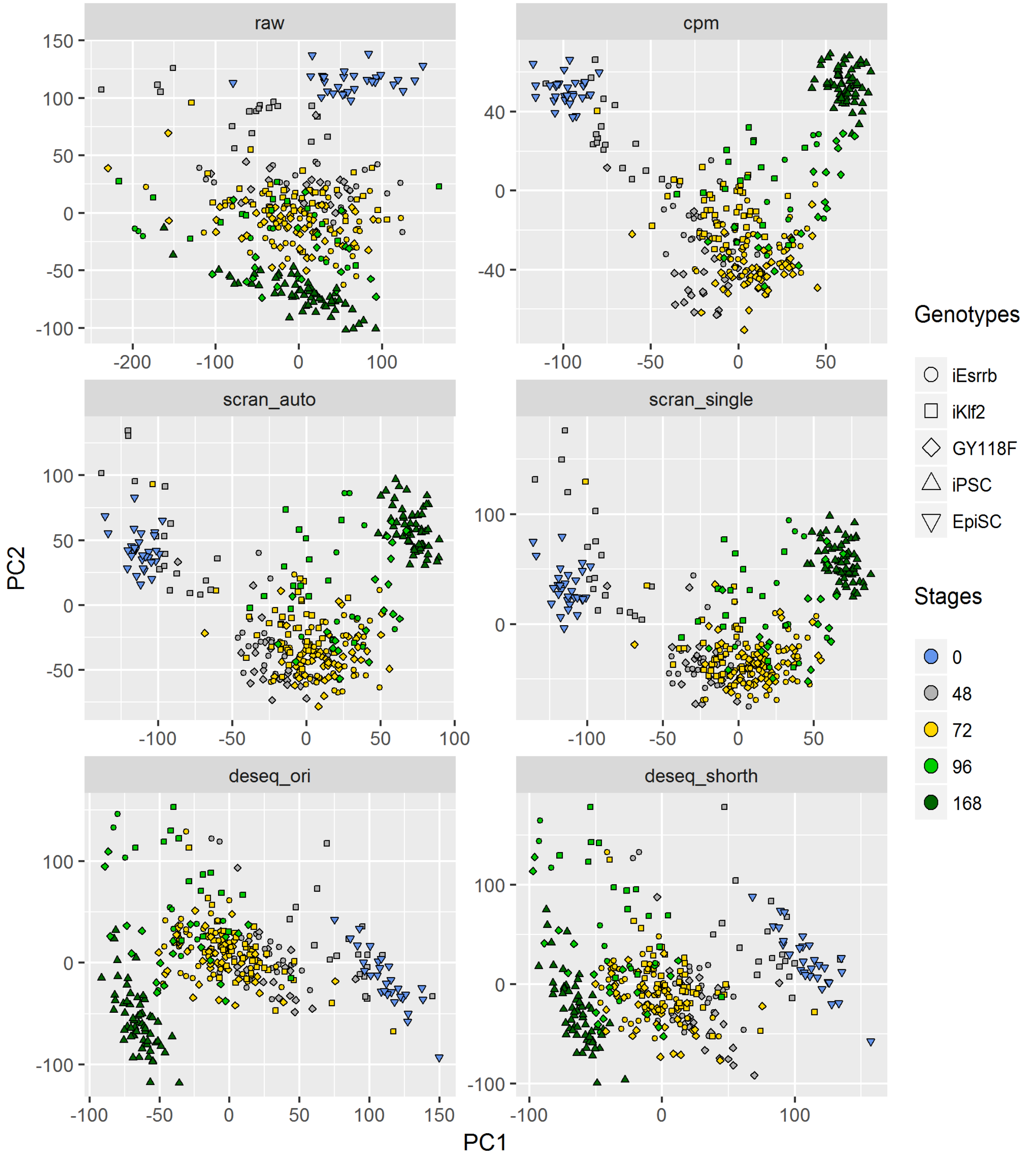


Figure 4.10 Principal component analyses of raw and normalised reads.
CPM, counts per million; scran_auto, Scran with default clustering; scran_single, Scran with a single cluster; deseq_ori, DESeq with default median; deseq_shorth, DESeq with shorth estimator.

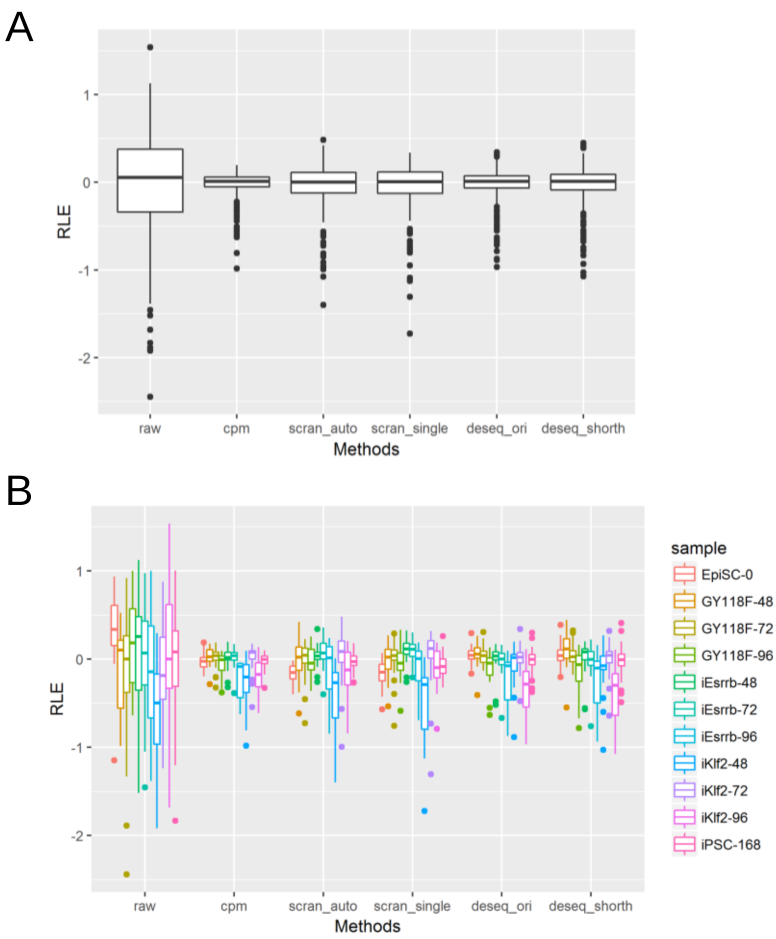


Figure 4.11 Cell-wise relative log expressions of raw and normalised reads. *Sample names are of the following format, [cell line]-[hours]. CPM, counts per million; scran_auto, Scran with default clustering; scran_single, Scran with a single cluster; deseq_ori, DESeq with default median; deseq_shorth, DESeq with shorth estimator.*

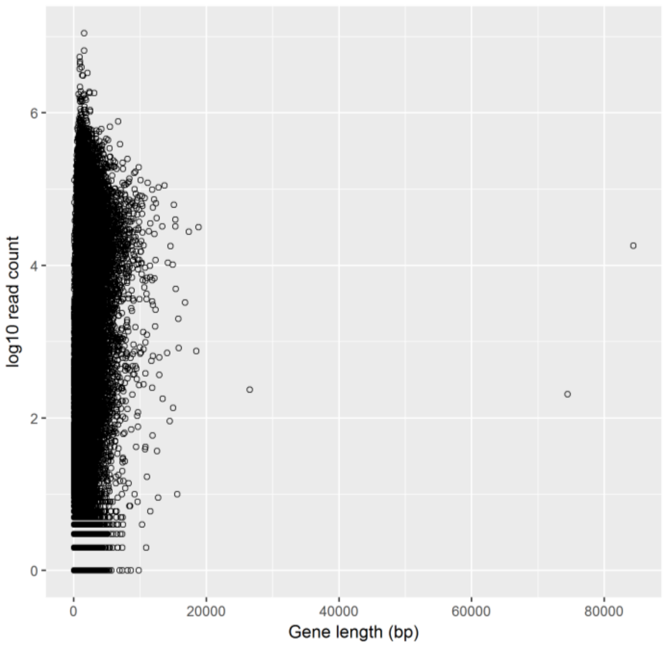


Figure 4.12 Relationship between log10 read counts and gene lengths.

4.3.3 Investigating potential batch effects

Among the technical factors in single-cell RNAseq experiments, the key confounding factor is the batches where the sequenced cells come from. This is because each 96-well plate is usually considered as a batch, and is processed separately during cell sorting and sequencing library generation. In addition, each 96-well plate is usually sequenced in a separate lane within each flow cell. All of these processing steps are known to contribute to technical noise in expression data due to the batch differences. This single-cell RNAseq data was generated using four 96-well plates, where each plate can be considered as a separate batch. Note that most of the samples on the four plates were completely confounded by batch effects, as each plate contain a different sample.

The plate information was used to check if the batch effect may be confounding the results by performing dimensionality reduction. The dimensionality reduction methods used for this purpose are PCA, DM and tSNE (Figure 4.13). The result of DM is very similar to PCA, therefore is not shown here. tSNE in particular is very useful for detecting any confounding factors, as it can detect complex non-linear relationships and display different samples as disjoint clusters. In Figure 4.13, the results of PCA and tSNE showed that specific plates were not particularly enriched for certain sub-populations of cells in EpiSCs and iPSCs. For the intermediate cells with three transgenes, the experimental time points were completely confounded with the plates (Figure 4.13), therefore making batch effect correction for plate effects inadvisable.

Batch effect correction using ComBat function in sva R package, which is a linear model-based method under Bayesian framework (Johnson et al. 2007), was attempted, but the corrected results were not meaningful as most samples are completely confounded by the batch effects. However no significant technical noise due to the plate effect is expected, because the PCA result on single-cell RNAseq data agrees with the PCA result on bulk RNAseq data (Figure 4.4). Therefore the single-cell RNAseq data was used for downstream analyses without correcting for batch effect.

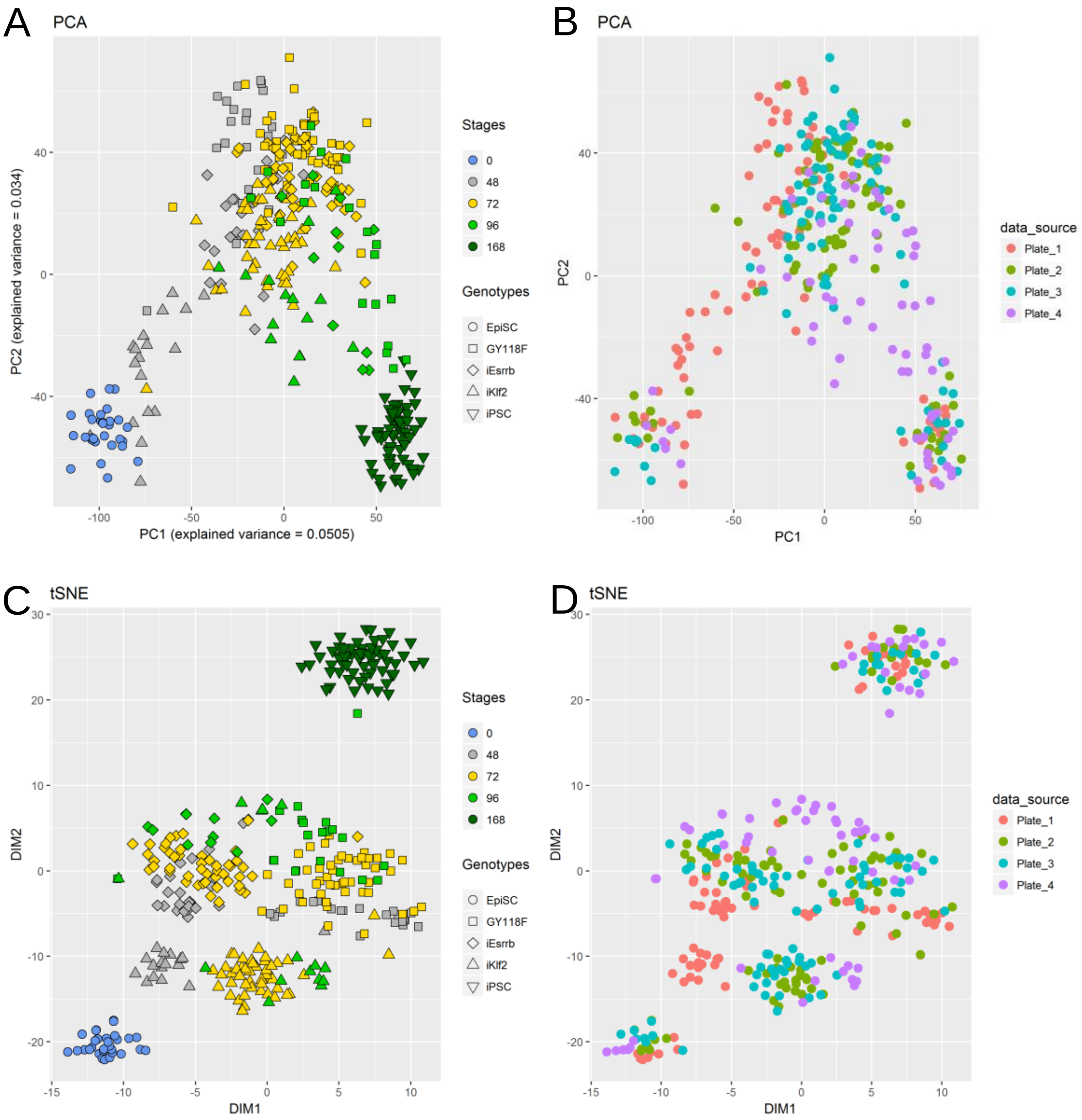


Figure 4.13 Dimensionality reduction analyses with plate information.
(A) PCA with experimental time and cell line labels, (B) PCA with plate labels, (C) tSNE with experimental time and cell line labels, (D) tSNE with plate labels.

4.3.4 Testing differential expression analysis methods for single-cell RNAseq

Since most differential expression analysis methods were developed for bulk RNAseq, and differential expression analysis is one of the key steps in analysing single-cell RNAseq, it would be important to investigate the performance of existing differential expression analysis methods on single-cell RNAseq. To enable a fair comparison of differential expression analysis methods, synthetic expression data with single-cell RNAseq technical noise were generated using a Beta-Poisson model with three parameters, α, β, and γ (Wills et al. 2013; Vu et al. 2016) (Section 4.6.5). Beta-Poisson model offers advantages over negative binomial or zero-inflated negative binomial models, as it can model over-dispersion and bimodality, as well as zero-inflation with a simple extension. In additions, the parameters in a Beta-Poisson model has biological meanings, where α describes the rate of activation of transcription, β describes the rate of inhibition of transcription and γ describes the rate of generation of transcripts while transcription is activated.

The synthetic expression data were generated such that they contain a set of genes that was differentially expressed between the two samples (Section 4.6.5). The synthetic expression data contain 10000 genes and 200 cells. Two different sets of synthetic expression data were generated, namely synthetic data with differing mean and variance between the two samples, and synthetic data with only differing variance between the two samples. Eight different methods were tested here, which include DESeq2 (Love et al. 2014), edgeR (Robinson et al. 2010), Wilcoxon rank sum test, Kolmogorov-Smirnov test, SCDE (Kharchenko et al. 2014), M3Drop (Andrews 2016), MAST (Finak et al. 2015), and Brennecke highly variable genes test (Brennecke et al. 2013). Out of all the methods, SCDE, M3Drop, MAST and Brennecke highly variable genes test are developed for single-cell RNAseq specifically, while DESeq2 and edgeR are developed for bulk RNAseq. SCDE performs differential expression analysis with a mixture of two distributions, with the first being a negative binomial distribution that models expression levels, and the second being a Poisson distribution that models dropouts. M3Drop fits a modified Michaelis-Menten equation to account for the presence of dropouts. MAST utilises a two-part generalised linear model to model the fraction of cells that express a certain gene and the level of expression for each gene separately. Brennecke highly variable genes test attempts to detect variable genes with variations that are higher than variations due to technical noise alone by using ERCC spike-ins or control samples. As for DESeq2 and edgeR, they are very similar in utilising likelihood ratio test or Wald test to compare negative binomial linear models fitted to the expression values.

When detecting differential gene expression on synthetic data with varying mean and variance, it can be seen that SCDE, DESeq2 and edgeR were the best performing algorithms in terms of area under the curves (AUC) for receptor operating characteristics (ROC) and precision-recall (PR) curves (Figure 4.14A & B). The superior performance of DESeq2 and edgeR in experiments with high replicate numbers were also verified by an independent study (Schurch et al. 2016). The test case described here represents the typical purpose of differential expression analyses, which is to find out the difference in expression values between two samples in terms of the measure of central tendency (e.g. median). However, in the case of single-cell expression data, another interesting objective is to detect for the difference in expression values between two samples in terms of the measure of variability (e.g. variance). This test case was explored using the synthetic data with varying variance but fixed mean. The results showed that MAST, Kolmogorov-Smirnov and Wilcoxon rank sum performed the best in terms of AUC for ROC and PR (Figure 4.14C & D). This is likely to reflect that most differential expression analysis methods were developed for detecting differences in sample means or medians, but not in sample variances.

Lastly, in terms of practical applications, it is important to take account of computing resources required for running differential expression analyses. The computing resources that are limiting for most people are the CPU cores and memory. Some algorithms do not scale well with the number of cells and/or number of CPU cores used, and may require prohibitively large amount of memory to be run. Here the computing times required per single CPU core for all differential expression analyses algorithms were investigated. The result indicated that while SCDE offers the best performance, it took 44 times longer than the second best performing algorithm, DESeq2, to run (~5 hours vs ~7 minutes) (Figure 4.15). The rest of the algorithms were finished in less than 7 minutes for 10000 genes and 200 cells.

In summary, when accounting for accuracy, error rates and computing times required, DESeq2 is the most suitable algorithm for detecting differentially expressed genes in terms of sample medians in single-cell RNAseq. This conclusion is conditioned on the distributions of the single-cell expression values and the set of differential expression algorithms tested here. Note that DESeq2 differential expression analysis tested here relies on DESeq2 normalisation, which has been shown to be not suitable for normalising single-cell RNAseq data (Section 4.3.2).

This problem can be overcome by using an alternative normalisation in place of the standard DESeq2 normalisation when running DESeq2 differential expression analysis.

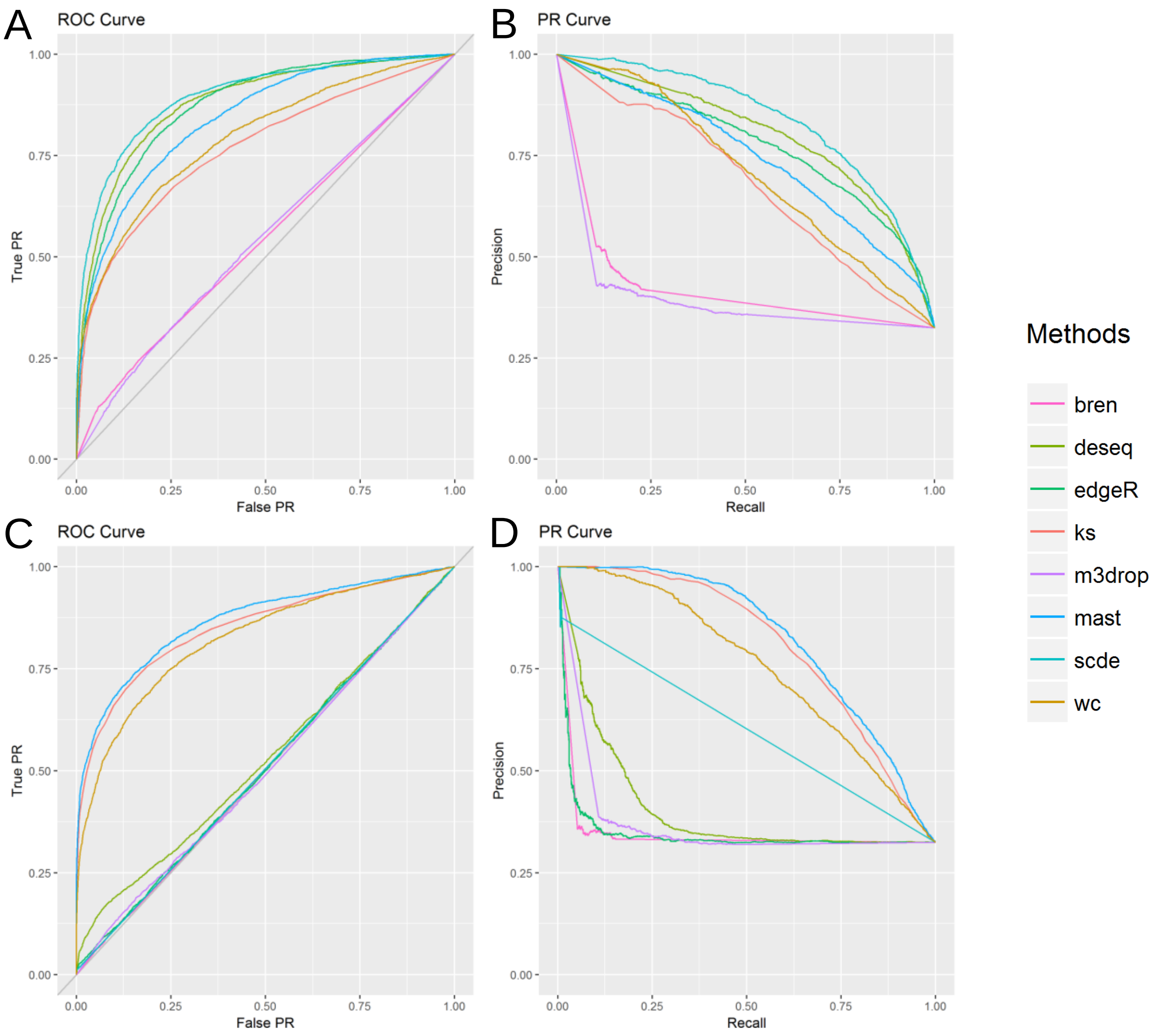


Figure 4.14 ROC and PR curves of differential gene expression analysis algorithms. *(A-B) ROC and PR curves on synthetic expression data with varying mean and variance, (C-D) ROC and PR curves on synthetic expression data with varying variance, but fixed mean.*

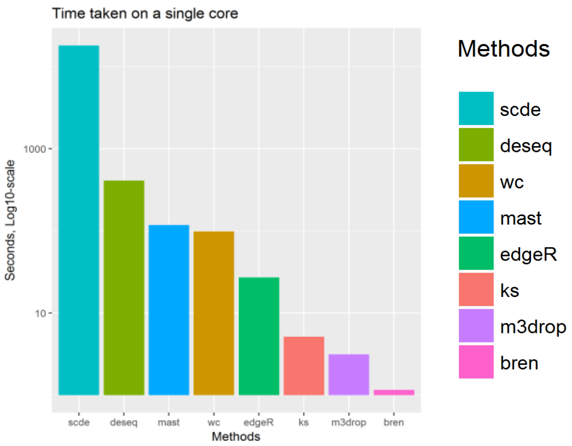


Figure 4.15 Algorithm run times for differential gene expression analysis algorithms. *Only a single core is used in each of the algorithms for comparison purpose. Time taken was recorded in seconds, and presented in log10 scale.*

4.4 Single-cell RNAseq reveals expression profile differences within each cell line during reprogramming

4.4.1 Most reprogramming cells are in G2/M phase

One of the key biological factors in single-cell RNAseq experiments regardless of the biological systems studied is the cell cycle effect. The aim here is two-fold, which is (1) to characterise the cell cycle profiles of the reprogramming cells, and (2) to detect and account for the cell cycle profiles in downstream analyses. Accounting for cell cycle profiles is very important in single-cell RNAseq analysis, as the expression variations of cell cycle-related genes often contribute to the clustering and the detection of sub-populations of cells (Buettner et al. 2015).

Here, the cell cycle phase for each cell was inferred using the cyclone function implemented in scran package as described in Section 4.6.4. Each cell is assigned a cell cycle phase of either G0/1, G2/M, S or unknown phase. The proportion of cells in each cell cycle phase for each cell type is summarised in Figure 4.16. It can be seen that almost all cells were in either G2/M or S phase in all cell types, with very few cells in G0/1 phase. This shows that the cells were actively growing and dividing. The cell cycle profiles of these cells are consistent with the understanding of the cell cycle in pluripotent stem cells, where G1 phase is shortened and G1 checkpoint regulation is absent (Savatier et al. 1996; Coronado et al. 2013). Recent studies have shown that the lengthening of the G1 phase is associated with differentiation in both mouse and human ESCs (Coronado et al. 2013; Calder et al. 2013).

The identified cell cycle phase information was then used to check if the cell cycle effect may be confounding downstream results by performing dimensionality reduction. The dimensionality reduction methods used for this purpose are PCA and tSNE (Figure 4.17). The result of DM is very similar with PCA, therefore is not shown here. In Figure 4.17, the results of PCA and tSNE showed that cell cycle variations were spread out rather uniformly, and were not particularly enriched within certain sub-populations of cells. This is true except for the small numbers of G0/1 cells that are enriched for earlier time points. However, cell-cycle effect is unlikely to confound the results of downstream analyses.

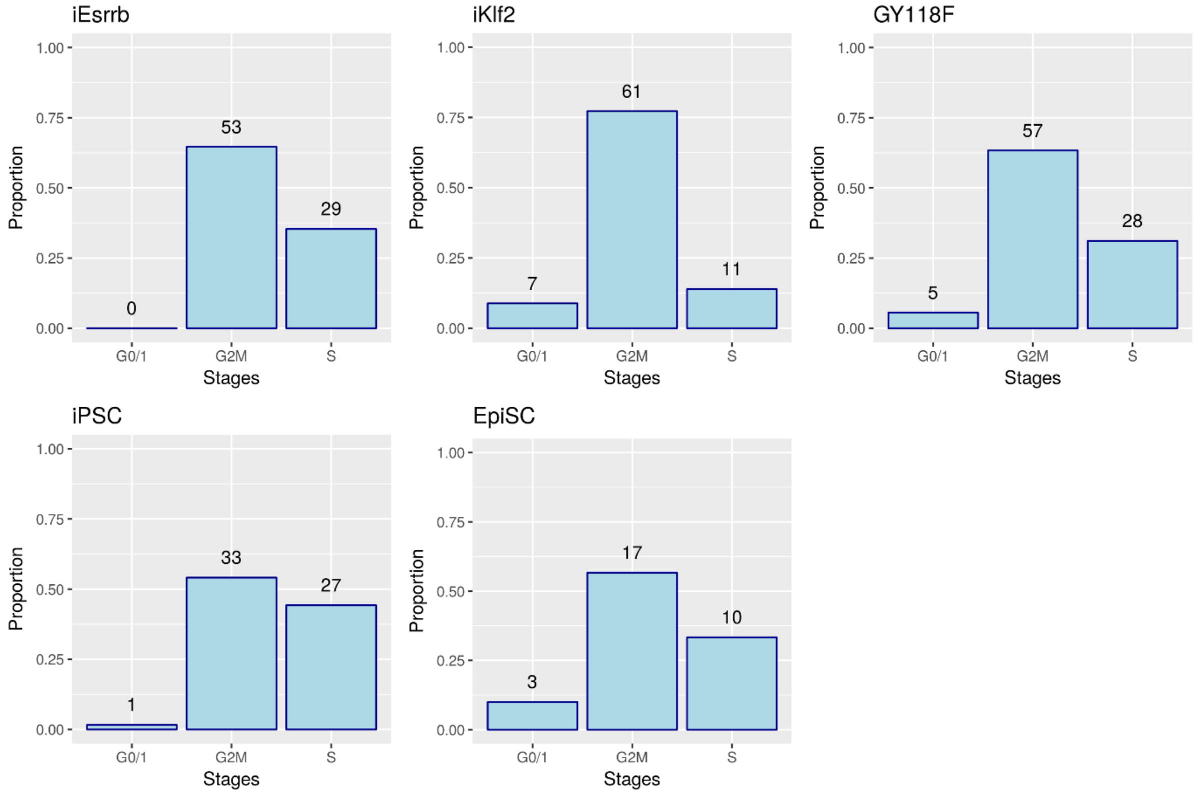


Figure 4.16 Number and proportion of cells per cell cycle stage.
Cell cycles were identified based on known gene expression profiles for each cell cycle stage.

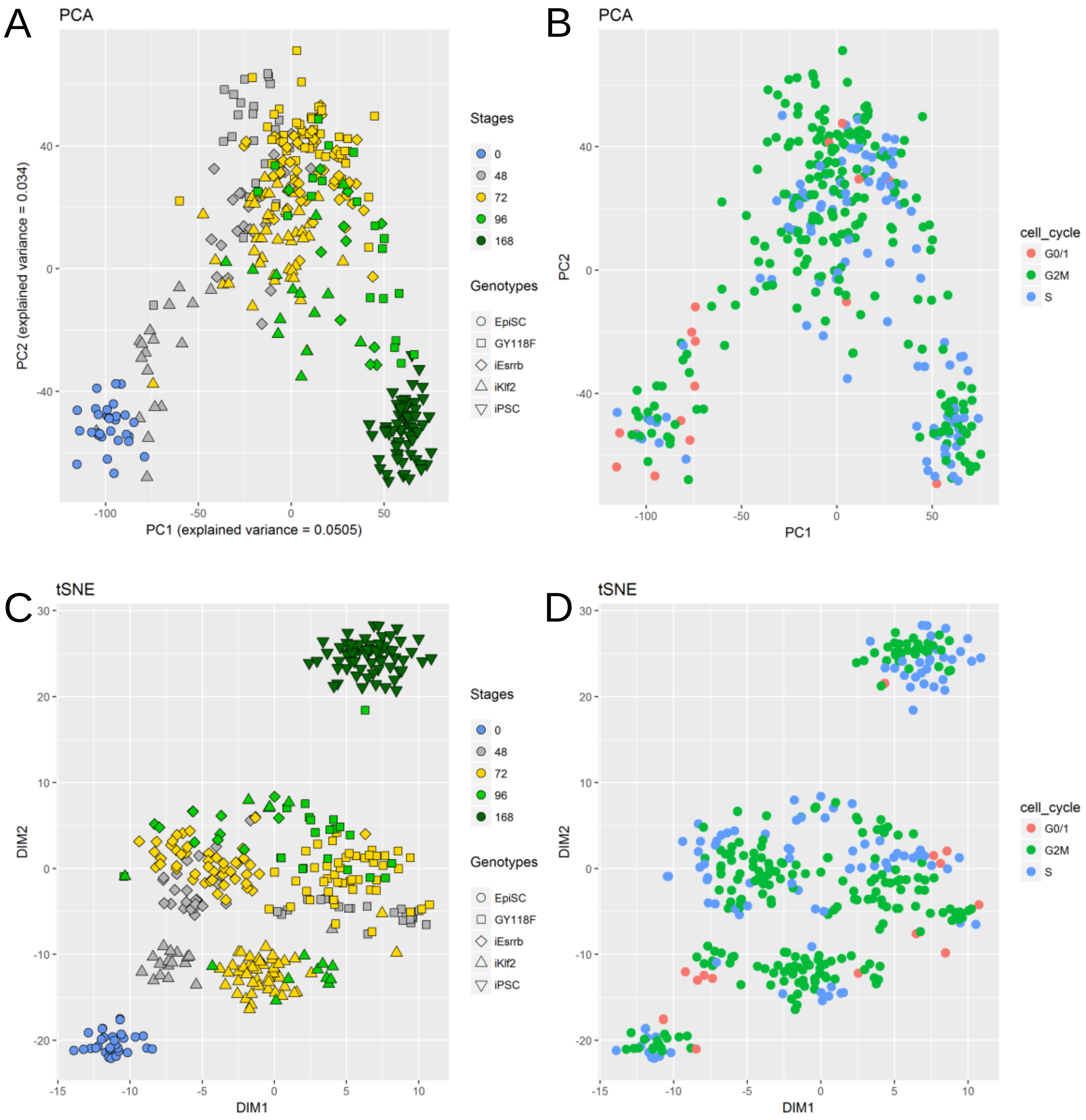


Figure 4.17 Dimensionality reduction analyses with cell cycle phase information. *(A) PCA with experimental time and cell line labels, (B) PCA with cell cycle phase labels, (C) tSNE with experimental time and cell line labels, (D) tSNE with cell cycle phase labels.*

4.4.2 iKlf2 cell line contains multiple subpopulations during reprogramming

The aim of most single-cell RNAseq experiments is to detect subpopulations within the cells that otherwise cannot be detected in bulk RNAseq experiments. Here, the identification of cell subpopulations in each cell line was attempted by investigating similarities and differences in their expression profiles through a clustering approach. ICGS clustering method (Olsson et al. 2016), which is an unbiased clustering method, was used to select for a set of most informative genes across all cells that facilitates clustering. The set of most informative genes contains 2029 genes, which was then used for clustering the cells by using hierarchical clustering (Figure 4.18 & Figure 4.19). In Figure 4.18, it can be seen that EpiSCs and iPSCs were located in distinct clusters, Cluster 7 and 4 respectively. This suggests that EpiSCs and iPSCs were very different from the intermediate cells.

Among the intermediate cells driven by the three transgenes, they were separated into six clusters, where cluster 1 contains *iEsrrb* early cells, cluster 2 contains *iKlf2* early cells, cluster 3 contains *GY118F* early cells, cluster 5 contains *iKlf2* early cells, cluster 6 contains *iEsrrb* and *GY118F* late cells, and cluster 8 contains *iKlf2* late cells (Figure 4.19). In general, *GY118F* and *iEsrrb* cells were more similar to one another than *iKlf2* cells. This can be seen in cluster 6, where *GY118F* and *iEsrrb* cells were located in the same cluster. In contrast, *iKlf2* cells occupied three almost exclusive clusters, which are cluster 2, 5 and 8, where only cluster 5 includes 2 *GY118F* cells. The similarities and differences exhibited by the cell lines and clusters were also clearly observed in dimensionality reduction analyses such as PCA, DM and tSNE (Figure 4.20). In particular, *GY118F* and *iEsrrb* cells can be seen to overlap more extensively with one another compared to *iKlf2* cells.

Interestingly, the single-cell RNAseq results showed that *iKlf2* cells are more similar with EpiSCs and iPSCs, while the bulk RNAseq results showed that *iEsrrb* cells are more similar with EpiSCs and iPSCs (Figure 4.5 & Figure 4.18). It is likely that there are subpopulations of cells within each cell line with different behaviours that influenced the single-cell results. As bulk RNAseq averages the expression of all cells in a pool, the differences among subpopulations can only be observed with single-cell RNAseq.

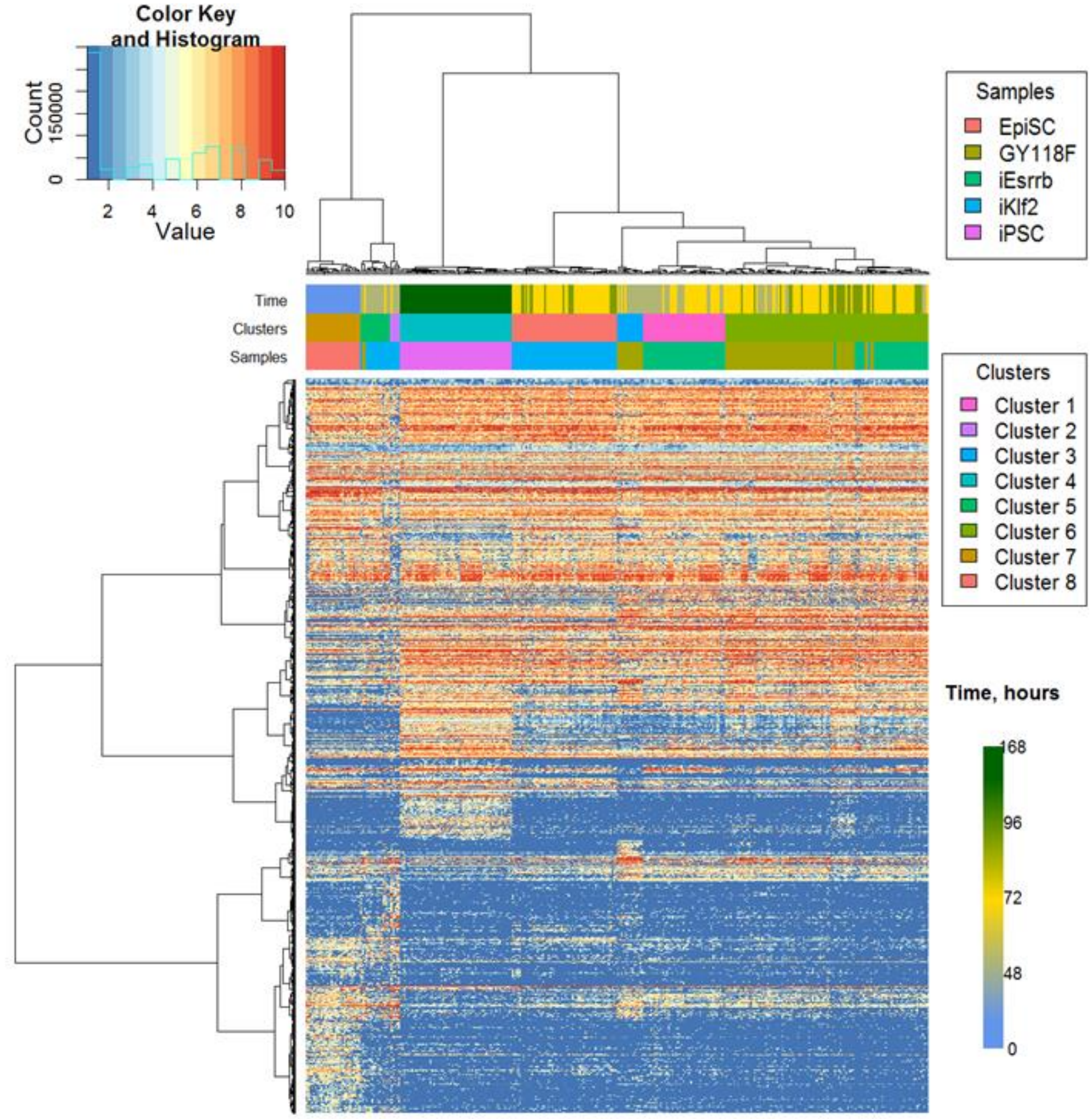


Figure 4.18 Heatmap based on most informative genes selected by ICGS.

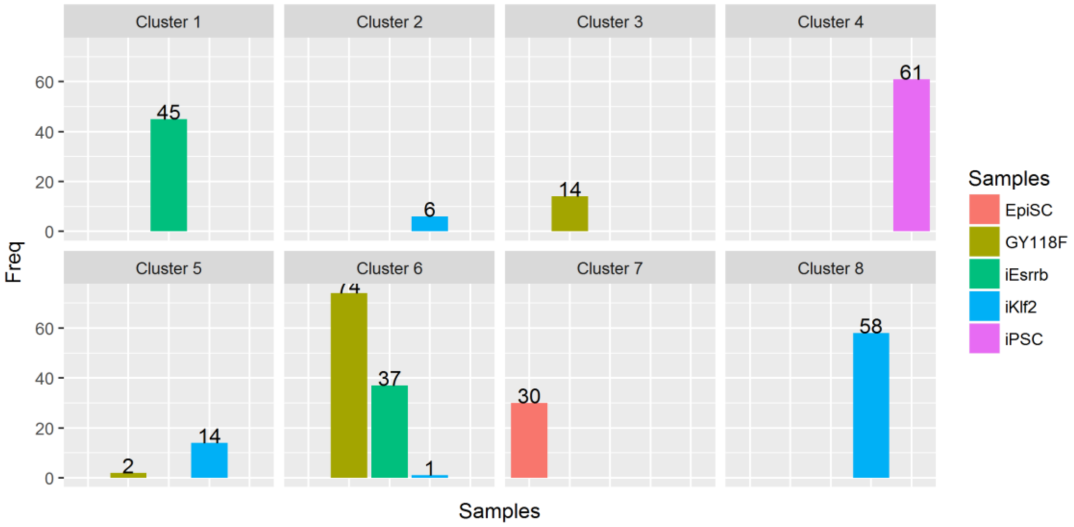


Figure 4.19 Number of cells for each cell type per cluster.

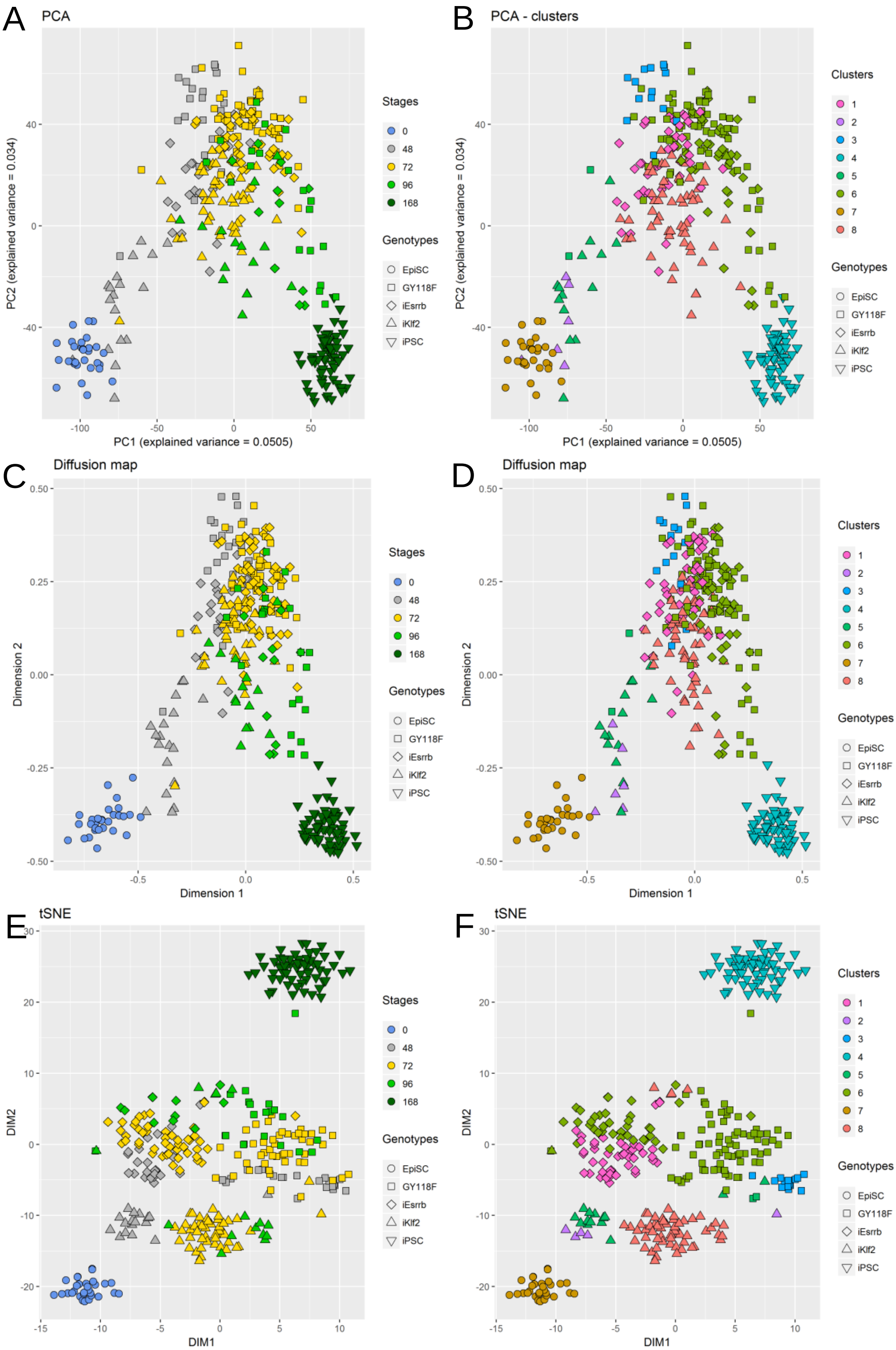


Figure 4.20 Dimensionality reduction analyses on all cells.
(A) PCA labelled by experimental time, (B) PCA labelled by clusters, (C) DM labelled by experimental time, (D) DM labelled by clusters, (E) tSNE labelled by experimental time, (F) tSNE labelled by clusters. The clusters identified were the same clusters from Figure 4.18.

The identified clusters offered an opportunity to dissect the expression and functional differences among the clusters of cells, particularly on cell subpopulations within the same cell line. To achieve this, differential expression gene analysis followed by Gene Ontology (GO) analysis were performed to investigate the genes and the biological processes that are unique to the intermediate reprogramming cells. This was done by comparing each cluster with respect to both EpiSCs and iPSCs combined.

Among the three cell lines, *iKlf2* cell line showed the most difference within a cell line. *iKlf2* cell line has three distinct clusters (i.e. cluster 2, 5, and 8), with *iKlf2* cells in cluster 2 being the most different (1937 differentially expressed genes). It is likely that these *iKlf2* clusters represent subpopulations of cells that were not successfully reprogrammed, or cells that underwent different reprogramming routes. For example, *iKlf2* cells in cluster 2 showed upregulation of cell-cell adhesion and downregulation of division and cell migration, while *iKlf2* cells in cluster 5 showed upregulation of apoptosis (Table 4.3 & Table 4.4). These results suggest that cells in cluster 2 were differentiating and cells in cluster 5 were dying. As for *iKlf2* cells in cluster 8, which had progressed later in development, the cells were upregulating neuron and epithelial differentiation-related processes, while downregulating apoptosis and cell adhesion.

In summary, each transgene drove reprogramming via different mechanisms, and the gene expression profile of *iKlf2* cell line was very different from *GY118F* and *iEsrrb* cell lines. Besides the differences among cell lines, there are also interesting differences within each cell line as illustrated by the subpopulations within *iKlf2* cell line. The single-cell RNAseq results showed that the reprogramming of EpiSCs is a dynamic process within a cell line, which involves changing gene expression profiles along reprogramming and the presence of subpopulations within this reprogramming process.

Biological Process	**Num. of Annotated Genes**	**Num. of Observed Genes**	**Adjusted P-values**
***iKlf2* cluster 2 (Total: 1154 upregulated genes)**			
Cell-cell adhesion	917	100	9.20E-13
Arp2/3 complex-mediated actin nucleation	31	11	6.20E-07
Small GTPase mediated signal transduction	477	70	6.80E-07
Negative regulation of cell migration	208	30	1.10E-06
Cytokinesis	116	16	8.30E-06
Membrane budding	42	11	1.70E-05
Actin filament capping	30	9	3.10E-05
Regulation of cell migration	682	84	3.70E-05
Establishment or maintenance of cell polarity	165	26	3.90E-05
Protein localization to vacuole	46	11	0.00017
***iKlf2* cluster 5 (Total: 685 upregulated genes)**			
Extrinsic apoptotic signaling pathway in absence of ligand	70	8	2.00E-06
Regulation of transcription regulatory region DNA binding	39	8	4.60E-06
Angiogenesis	410	25	6.30E-05
Negative regulation of regulated secretory pathway	22	5	0.00018
Blood vessel remodeling	50	7	0.00024
Negative regulation of cell proliferation	585	29	0.00026
Membrane assembly	24	5	0.00028
Lysosomal transport	65	7	0.00041
Negative regulation of cell migration	208	13	0.00041
Positive regulation of cell cycle process	183	14	0.00056
***iKlf2* cluster 8 (Total: 473 upregulated genes)**			
Regulation of neuron differentiation	632	15	2.70E-05
Negative regulation of extrinsic apoptotic signaling pathway in absence of ligand	31	4	0.00019
Negative regulation of cell adhesion	222	8	0.00039
Negative regulation of DNA binding	42	4	0.00061
Negative regulation of apoptotic process	798	21	0.00092
Inositol phosphate biosynthetic process	24	3	0.00138
Regulation of gene expression	3615	41	0.00153
Rhythmic process	286	9	0.00164
Oxidation-reduction process	850	16	0.00195
Regulation of odontogenesis	27	3	0.00196

Table 4.3 Top 10 upregulated biological processes in iKlf2 cell line based on Gene Ontology (GO) analysis.

Biological Process	Num. of Annotated Genes	Num. of Observed Genes	Adjusted P-values
***iKlf2* cluster 2 (Total: 908 downregulated genes)**			
Cell division	520	94	4.00E-27
Mitotic nuclear division	399	81	8.30E-16
DNA replication initiation	31	15	4.60E-13
Chromosome segregation	281	84	5.50E-12
Mitotic sister chromatid segregation	123	37	1.30E-11
Protein localization to chromosome	59	17	5.50E-11
Attachment of spindle microtubules to kinetochore	25	11	2.30E-09
DNA replication	245	55	4.80E-09
Mitotic spindle assembly checkpoint	26	10	5.90E-08
Male meiosis	42	12	1.30E-07
***iKlf2* cluster 5 (Total: 624 downregulated genes)**			
Multicellular organism development	4492	158	0.00043
Spermatogenesis	427	26	0.00069
Biosynthetic process	5025	152	0.00075
Cellular amino acid biosynthetic process	46	6	0.00104
Regulation of meiotic nuclear division	32	5	0.00119
Fatty acid beta-oxidation	68	7	0.0013
Regulation of long-term neuronal synaptic plasticity	33	5	0.00138
Synaptonemal complex assembly	20	4	0.00145
Aspartate family amino acid biosynthetic process	20	4	0.00145
Organic cyclic compound catabolic process	282	12	0.00162
***iKlf2* cluster 8 (Total: 1413 downregulated genes)**			
Tube formation	156	22	4.90E-05
Regulation of cell shape	129	19	6.50E-05
Podosome assembly	21	7	7.90E-05
Positive regulation of cell migration	403	43	9.60E-05
Response to reactive oxygen species	159	20	0.00012
Negative regulation of microtubule depolymerization	23	7	0.00015
Peptidyl-proline modification	31	6	0.00016
Cell differentiation involved in embryonic placenta development	31	8	0.00018
Neurotrophin TRK receptor signaling pathway	24	7	0.0002
Anion transmembrane transport	92	13	0.00028

Table 4.4 Top 10 downregulated biological processes in iKlf2 cell line based on Gene Ontology (GO) analysis.

4.4.3 Pseudotime trajectory reveals dynamic differences in gene expression and biological processes between the cell lines during reprogramming

Another interesting aspect of single-cell RNAseq is that it can capture heterogeneity or asynchronicity in developmental progression, in which each cell is at a slightly different stage of reprogramming despite each cell being collected at the same experimental time. The cells in this single-cell RNAseq data were collected across five experimental time points (0, 48, 72, 96 and 168 hours), but the time gaps between the experimental time points are quite big. Since this data possesses single-cell resolution, pseudotime inference algorithms can be used to infer the reprogramming time of each cell, thereby using it to smooth out the cells and offer increased time resolution along the reprogramming trajectory.

Here, the DM-DPT pseudotime inference algorithm as described in Chapter 3 was used to infer pseudotime for the single-cell RNAseq along the reprogramming trajectory. DM-DPT pseudotime algorithm is suitable for this dataset, because the reprogramming trajectory can be resolved well in DM, and the cell lines are assumed to always have a single starting point and a single ending point (Figure 4.21). The first component of DM (DC1) corresponds to the reprogramming time, while the second component of DM (DC2) corresponds to the differences in the reprogramming states. Note that DC2 was able to capture the fact that the starting EpiSCs and the ending iPSCs share more homogeneous gene expression profiles relative to the intermediate reprogramming cells. As reprogramming state (DC2) is a function of time (DC1), this justifies the suitability of using DC1 to explain DC2 in DM-DPT pseudotime inference algorithm. Monocle2 and Wanderlust were also tested on the dataset, but the results were not ideal as Monocle2 gave a high number of branches, while Wanderlust failed to run due to the low number of cells.

It is likely that in reality, there are multiple intermediate points, which represent different reprogramming pathways within a cell line, and multiple ending points, which represent the final states achieved by reprogramming and non-reprogramming cells. However, with the relatively low number of cells available in this single-cell RNAseq and large time gaps between experimental time points, it is not possible to confidently resolve any such differences. Therefore, it is assumed that there is a single main reprogramming trajectory for each cell line, and the aim is to investigate the differences among the trajectories.

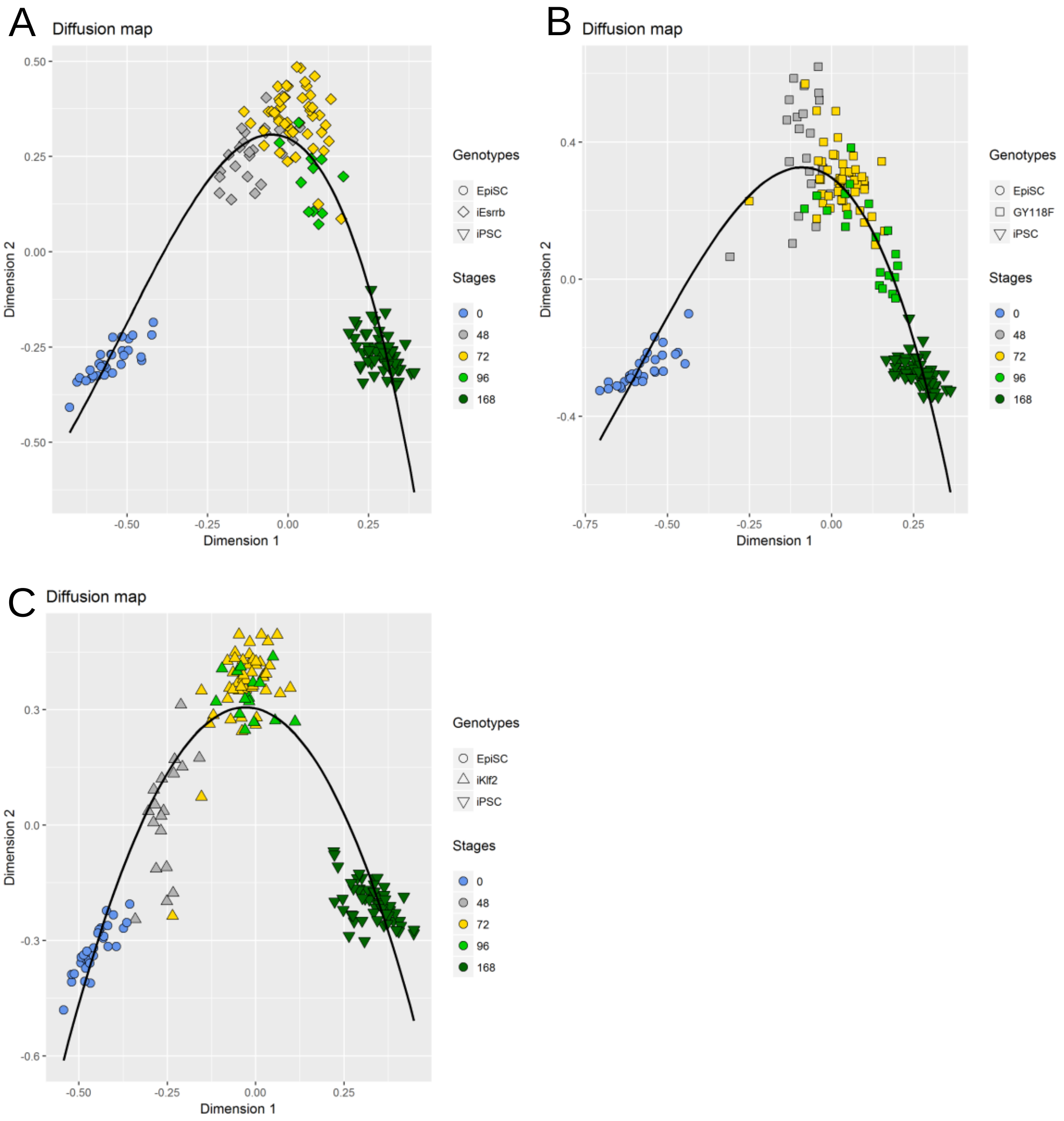


Figure 4.21 Diffusion maps with fitted pseudotime trajectory for each cell line.
(A) DM of iEsrrb cell line, (B) DM of GY118F cell line, (C) DM of iKlf2 cell line. The fitted line is a polynomial line that represents the pseudotime trajectory of each cell line.

With pseudotime inferred for each cell in each cell line, it is then possible to investigate changes in gene expression values as a function of time along the reprogramming trajectory. In order to detect genes that were differentially expressed as a function of pseudotime, likelihood ratio tests on negative binomial models, as implemented in Monocle2 R package (Trapnell et al. 2014), were performed. With an adjusted p-value threshold of 0.01, there were 7115 genes identified as differentially expressed in *iEsrrb* cell line, 7100 differentially expressed genes in *GY118F* cell line, and 6319 differentially expressed genes in *iKlf2* cell line. As these are large numbers of genes, and there is no straightforward way to calculate a fold change equivalent for pseudotime-based differential expression analyses, only the top 1000 differentially expressed genes in terms of p-values were chosen for further investigations. The differentially expressed genes can then be used to locate groups of genes which show similar expression profiles along the pseudotime by using hierarchical clustering with Spearman correlation distance measure (Figure 4.22). Four clusters of genes were obtained in each cell line, in which each cluster represents a different expression profile as described in Table 4.5.

Categories	**Expression changes across pseudotime**	**Corresponding gene clusters**		
		iEsrrb	**GY118F**	**iKlf2**
Cat 1	Low → High	2	2	1
Cat 2	High → Low	3	3	2
Cat 3	Low → High → Low	4	4	4
Cat 4	High → Low → High	1	1	3

Table 4.5   Categories of expression profiles.
The gene cluster number is the same as indicated in the heatmaps.

GO analysis was then performed on these different expression profiles for each cell line. Out of the four categories, it is expected that the genes in Cat 1 represent naïve pluripotency state in iPSCs, while the genes in Cat 2 represent primed pluripotency state in EpiSCs. These genes should also be highly similar across cell lines as shown in Figure 4.23A. 44.2% and 25.7% of the differentially expressed genes in Cat 1 and 2 respectively were the same across all three cell lines. In contrast, only 3.7% and 4.9% of the differentially expressed genes were the same in Cat 3 and 4. The same was observed in terms of GO terms, where Cat 1 and 2 share 17.7% and 5.8% of GO terms across all cell lines, while no GO terms were shared across cell lines in Cat 3 and 4.

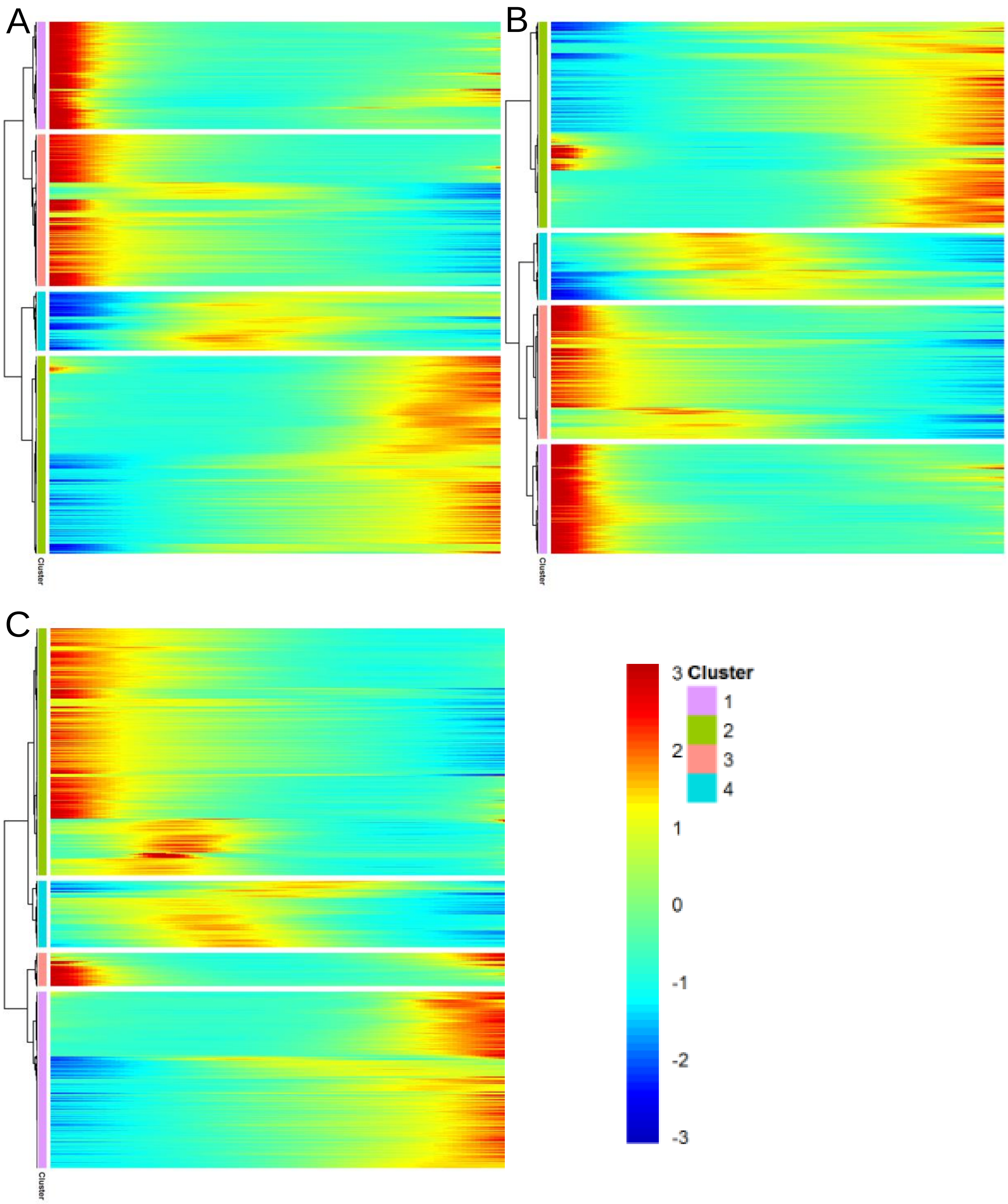


Figure 4.22 Heatmaps of top 1000 differentially expressed genes in each cell line. *(A) iEsrrb cell line, (B) GY118F cell line, (C) iKlf2 cell line. The cluster number is indicated by the colour scale on the top right of each plot.*

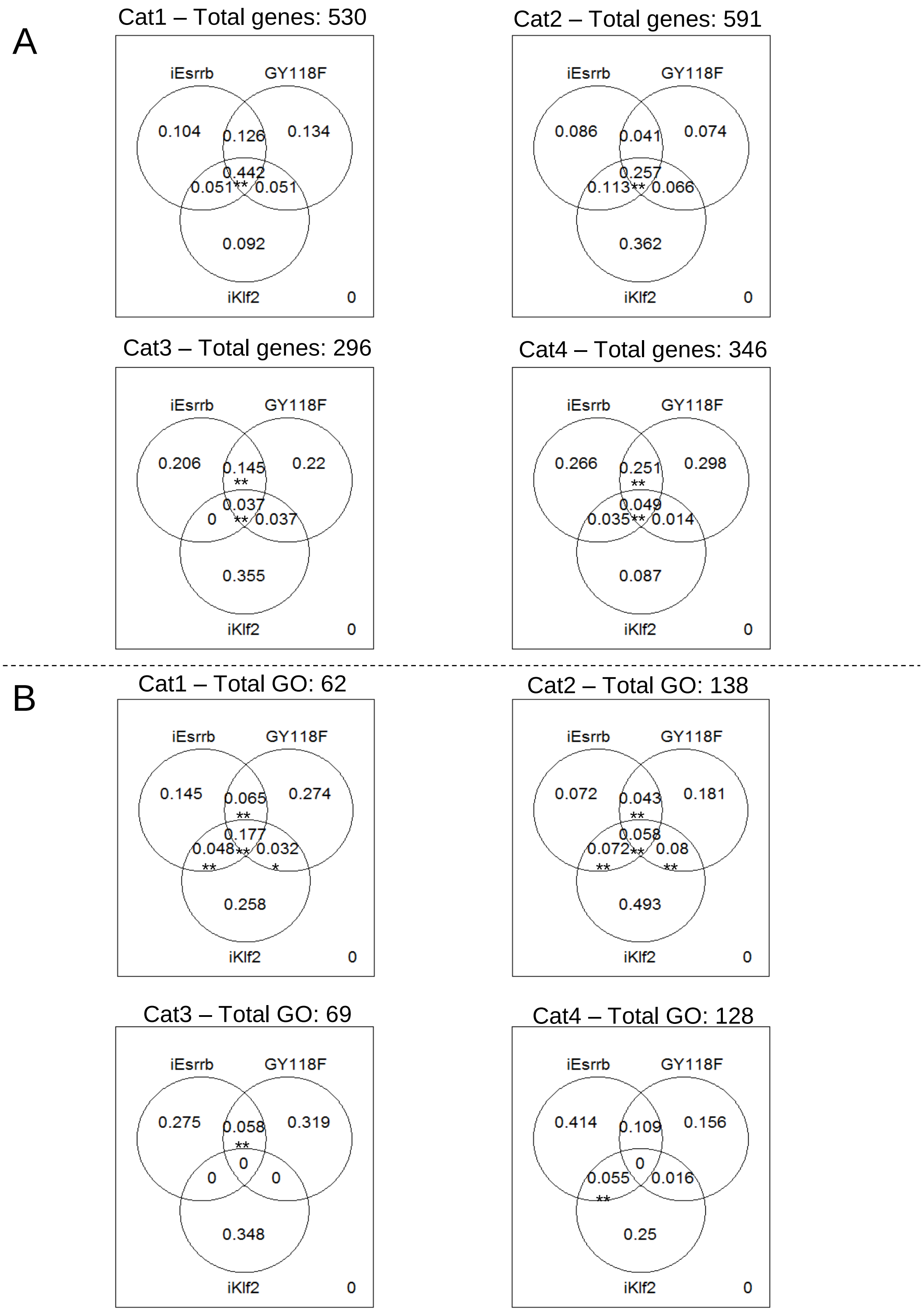


Figure 4.23 Venn diagrams of differentially expressed genes and Gene Ontology biological processes.
*Each category is as indicated in Table 4.5. *, p-value = 0.01; **, p-value = 0.001.*

The commonly enriched GO terms for genes in Cat 1 for all three cell lines contain spermatogenesis and other gametogenesis related biological processes (Table 4.6). Note that all cells analysed in this study are males. The result may suggest that the iPSCs identified here may contain some germline stem cells, which are pluripotent and have been shown to contribute to all germ layers under certain conditions (Donovan & de Miguel 2003). An alternative interpretation of the result is that the enrichment of gametogenesis terms may be suggesting that the iPSCs are similar to germline stem cells. Similarities are known to exist between ESCs and embryonic germ, which led to the suggestion that ESCs may be derived from primordial germ cells (Zwaka & Thomson 2004). Resolving ESCs and early germline stem cells is difficult because they share very similar expression profiles (Sharova et al. 2007).

As for the genes in Cat 2, the commonly enriched GO terms for all three cell lines are apoptotic process for epithelial cells and cell adhesion-related biological process (Table 4.7). The presence of apoptosis may suggest that some EpiSCs were dying before the initiation of reprogramming. Cell adhesion and other epithelial cells related processes are expected to be upregulated in EpiSCs with respect to ESCs or iPSCs, as EpiSCs form flat colonies and require associations among cells to survive (Li & Ding 2013).

The genes in Cat 3 and 4 correspond to genes that are uniquely perturbed during the reprogramming (Table 4.8 & Table 4.9). The GO results mostly agree with the observations obtained from bulk RNAseq as discussed in Section 4.2. For *iEsrrb* cell line, GO results indicated that the regulation of transcription was perturbed, the cells obtained morphology of ESCs and the inhibition of differentiation processes. For *GY118F* cell line, GO results indicated that gastrulation development was upregulated, cell adhesion and neural development were downregulated. For *iKlf2* cell line, GO results indicated that the downregulation of immune cell development and other differentiation processes.

In summary, these results suggest that while there is a group of shared biological processes during reprogramming as seen in Cat 1 and 2, the intermediate biological processes during reprogramming are different among cell lines. Each transgene was driving reprogramming via a different route as indicated by the GO biological processes enriched in Cat 3 and 4.

Biological Process	**Num. of Annotated Genes**	**Num. of Observed Genes**	**Adjusted P-values**
***iEsrrb* (Total: 208 differentially expressed genes)**			
Spermatogenesis	427	26	2.90E-06
Male meiosis	42	7	9.90E-06
Synaptonemal complex assembly	20	5	2.40E-05
Sister chromatid cohesion	49	6	6.20E-05
Fertilization	144	10	8.90E-05
Platelet-derived growth factor receptor signaling pathway	48	5	0.0018
Regulation of T cell migration	27	3	0.0019
Organic cyclic compound catabolic process	282	8	0.0022
Regulation of meiotic nuclear division	32	4	0.0026
Female meiotic division	33	4	0.0029
***GY118F* (Total: 337 differentially expressed genes)**			
Positive regulation of interferon-gamma production	62	7	0.00018
Organic cyclic compound catabolic process	282	11	0.00037
Neural crest cell migration	53	6	0.00051
Negative regulation of reproductive process	55	6	0.00063
Spermatogenesis	427	23	0.00082
Fertilization	144	7	0.00107
Osteoblast differentiation	203	9	0.00108
Male meiosis	42	5	0.0012
Mitotic spindle assembly checkpoint	26	4	0.00143
Lens fiber cell differentiation	28	4	0.0019
***iKlf2* (Total: 212 differentially expressed genes)**			
Spermatogenesis	427	24	1.70E-06
Positive regulation of leukocyte migration	106	6	0.00043
Fertilization	144	6	0.00046
Male meiosis	42	5	0.00051
Multicellular organism development	4492	96	0.00053
Negative regulation of nuclear division	54	5	0.00077
Cell differentiation	3541	79	0.00082
Somitogenesis	74	6	0.00112
Organic cyclic compound catabolic process	282	8	0.00115
Regulation of meiotic nuclear division	32	4	0.00156

Table 4.6 Top 10 enriched biological processes for Cat 1 expression profile in all cell lines based on Gene Ontology (GO) analysis.

Biological Process	Num. of Annotated Genes	Num. of Observed Genes	Adjusted P-values
***iEsrrb* (Total: 383 differentially expressed genes)**			
Cholesterol biosynthetic process	41	9	1.00E-08
Negative regulation of cell migration	208	12	0.00015
Negative regulation of angiogenesis	78	7	0.00021
Heterotypic cell-cell adhesion	38	5	0.00029
Positive regulation of stress fiber assembly	39	5	0.00032
Positive regulation of epithelial cell apoptotic process	25	4	0.00055
Isoprenoid biosynthetic process	26	4	0.00065
Outflow tract morphogenesis	70	6	0.00075
Actin filament capping	30	4	0.00113
Embryonic digestive tract development	32	4	0.00144
***GY118F* (Total: 472 differentially expressed genes)**			
Heterotypic cell-cell adhesion	38	5	0.00016
Positive regulation of MAPK cascade	416	16	0.00017
Positive regulation of stress fiber assembly	39	5	0.00019
Cholesterol biosynthetic process	41	5	0.00024
Membrane assembly	24	4	0.0003
Positive regulation of epithelial cell apoptotic process	25	4	0.00035
Cell-cell adhesion	917	29	0.00045
Response to tumor necrosis factor	112	7	0.00122
Cell chemotaxis	205	8	0.00127
Bicellular tight junction assembly	36	4	0.00145
***iKlf2* (Total: 399 differentially expressed genes)**			
Outflow tract morphogenesis	70	11	1.20E-06
Heterotypic cell-cell adhesion	38	8	3.50E-06
Anion transmembrane transport	92	10	6.40E-06
Positive regulation of extrinsic apoptotic signaling pathway	58	9	1.20E-05
Myelination	115	13	2.00E-05
Cell adhesion	1379	82	9.50E-05
Regulation of extrinsic apoptotic signaling pathway via death domain receptors	46	7	0.00014
Positive regulation of osteoblast differentiation	66	8	0.00023
Ventral spinal cord development	46	5	0.00023
Plasma membrane organization	261	22	0.00025

Table 4.7 Top 10 enriched biological processes for Cat 2 expression profile in all cell lines based on Gene Ontology (GO) analysis.

Biological Process	**Num. of Annotated Genes**	**Num. of Observed Genes**	**Adjusted P-values**
***iEsrrb* (Total: 294 differentially expressed genes)**			
Response to fluid shear stress	20	3	0.0002
Negative regulation of transcription from RNA polymerase II promoter	717	13	0.00025
Cell morphogenesis	1122	12	0.0003
Positive regulation of transcription from RNA polymerase II promoter	998	16	0.00044
Negative regulation of cell adhesion	222	7	0.00125
Stem cell population maintenance	156	6	0.00128
Developmental process involved in reproduction	644	10	0.00345
Single organismal cell-cell adhesion	651	9	0.00654
Respiratory system development	231	6	0.00671
Regulation of cellular component size	342	5	0.00692
***GY118F* (Total: 64 differentially expressed genes)**			
Negative regulation of myoblast differentiation	24	4	2.10E-05
Outflow tract morphogenesis	70	5	0.00012
Regulation of epidermal cell differentiation	46	4	0.00028
Cell development	1966	22	0.00069
Positive regulation of epidermis development	30	3	0.00115
Lung epithelial cell differentiation	32	3	0.00139
Cell fate commitment involved in formation of primary germ layer	32	3	0.00139
Gastrulation with mouth forming second	35	3	0.00181
Positive regulation of nitric oxide biosynthetic process	38	3	0.00229
Ossification	359	6	0.00413
***iKlf2* (Total: 259 differentially expressed genes)**			
Regulation of membrane depolarization	40	4	0.00015
Negative regulation of cell killing	20	3	0.00031
Regulation of cell migration	682	13	0.00119
Negative regulation of dendrite development	32	3	0.00127
Negative regulation of lymphocyte mediated immunity	33	3	0.00139
Regulation of natural killer cell mediated cytotoxicity	35	3	0.00165
Positive regulation of Notch signaling pathway	36	3	0.00179
Phospholipase C-activating G-protein coupled receptor signaling pathway	79	4	0.00196
Negative regulation of innate immune response	38	3	0.00209
Regulation of transcription regulatory region DNA binding	39	3	0.00226

Table 4.8 Top 10 enriched biological processes for Cat 3 expression profile in all cell lines based on Gene Ontology (GO) analysis.

Biological Process	Num. of Annotated Genes	Num. of Observed Genes	Adjusted P-values
***iEsrrb* (Total: 115 differentially expressed genes)**			
Negative regulation of transcription from RNA polymerase II promoter	717	22	1.20E-05
Regulation of morphogenesis of a branching structure	57	6	2.30E-05
Motor neuron axon guidance	32	5	2.30E-05
Neuroepithelial cell differentiation	61	5	5.70E-05
Glial cell differentiation	194	14	7.20E-05
Ventral spinal cord development	46	4	0.00033
Regulation of astrocyte differentiation	31	4	0.00034
Prostate gland epithelium morphogenesis	32	4	0.00039
Negative regulation of epithelial cell differentiation	37	4	0.00068
Neuron maturation	42	4	0.0011
***GY118F* (Total: 127 differentially expressed genes)**			
Synapse organization	229	10	5.60E-05
Negative regulation of embryonic development	28	4	0.00025
Negative regulation of gliogenesis	45	4	0.00044
Wnt signaling pathway, planar cell polarity pathway	35	4	0.00059
Axon guidance	194	11	0.0006
Cell adhesion	1379	30	0.00086
Anion transmembrane transport	92	5	0.00108
Cholesterol biosynthetic process	41	4	0.00108
Limb morphogenesis	162	7	0.00111
Neuron maturation	42	4	0.00119
***iKlf2* (Total: 130 differentially expressed genes)**			
Labyrinthine layer morphogenesis	25	4	1.40E-06
Odontogenesis of dentin-containing tooth	76	4	0.00012
Patterning of blood vessels	44	3	0.00043
Neural crest cell migration	53	3	0.00075
Hippocampus development	58	3	0.00097
Hormone secretion	306	4	0.00141
Diencephalon development	67	3	0.00148
Regulation of epidermal cell differentiation	46	3	0.00177
Morphogenesis of a branching epithelium	211	9	0.00215
Positive regulation of heart rate	21	2	0.00223

Table 4.9 Top 10 enriched biological processes for Cat 4 expression profile in all cell lines based on Gene Ontology (GO) analysis.

4.5 Conclusions

Here, we have shown that the *iEsrrb*, *iKlf2* and *GY118F* transgene cell lines drive EpiSC reprogramming via different mechanisms. This is firstly shown through investigating cell line differences using bulk RNAseq data. The upregulation of *Esrrb* drives reprogramming by modulating transcriptional regulation responsible for establishing naïve ESC identity, the upregulation of *Klf2* drives reprogramming by regulating cell proliferation and differentiation, while the upregulation of pSTAT3 in *GY118F* cell line drives reprogramming by regaining trophectoderm potential and downregulating BMP/SMAD pathway.

The follow up analysis using single-cell RNAseq allows the investigation of subpopulations within the cell lines and the gene expression dynamics along the pseudotime trajectory. *iKlf2* cell line is shown to possess multiple subpopulations with different biological properties that may affect their reprogramming successes. The inferred pseudotime along the reprogramming trajectory shows different gene expression dynamics in the three cell lines, particularly in the genes whose expressions are only perturbed during the intermediate reprogramming state.

Taken together, these observations enable a better understanding of the molecular mechanisms underlying each individual pathway as driven by upregulated *Esrrb*, *Klf2* and pSTAT3 in the context of EpiSC reprogramming. In addition, these three transgenes are downstream of the molecular pathways regulated by the conventional 2i+LIF ESC culture condition. The results in this study will offer further insights into the downstream mechanisms regulated by each individual component from 2i+LIF.

4.6 Materials and methods

4.6.1 Cell lines

The cell lines used in this study were generated by Hannah Stuart and Tim Lohoff from Jose Silva’s lab. The cell lines were generated by introducing Doxycycline-inducible PiggyBac expression plasmids, which contain the transgenes, the reverse tetracycline-controlled transactivator, and a monoallelic *Rex1*-destablised GFP reporter, into EpiSC lines. Successfully reprogrammed cells (i.e. iPSCs) were assessed via colony formation assays by selecting for blasticidin resistance and the formation of dome-shaped colonies, and chimera formations by checking for the presence of GFP-labelled cells in embryos and the coat colour of F1 offspring.

4.6.2 Processing of bulk and single-cell RNAseq data

The bulk RNAseq data consist of 92 samples, while the single-cell RNAseq data consist of 360 cells. Both RNAseq data contain samples/cells from the three cell lines with transgenes (i.e. iEssrb, iPStat3, iKlf2), and an empty vector control cell line. The single-cell RNAseq data also contain ESC cells. Both RNAseq data were generated using the same SmartSeq2 library preparation protocol, and sequenced on the Illumina HiSeq 4000 machine. However, the bulk and the single-cell RNAseq data differ in two technical aspects. Firstly, the bulk RNAseq data contain samples that were gated and enriched for different reprogramming success rates, while the single-cell RNAseq data contain only samples that were gated and enriched for cells with high reprogramming success rates. Secondly, the bulk RNAseq data has paired ends, while the single-cell RNAseq data has single end.

All RNAseq data were aligned using GSNAP version 2015-09-29 (Wu & Nacu 2010). Aligned reads were counted using HTSeq-count (Anders et al. 2015). Ensembl genome index and gene annotations release version 77 were used.

4.6.3 ICGS clustering and heatmaps

ICGS clustering (Olsson et al. 2016), which is implemented as part of the AltAnalyze software package version 2.0 (Emig et al. 2010), was used for selecting the most informative genes with default ICGS settings. The gene expression profiles based on the most informative genes were then used to hierarchical cluster all cells with Spearman correlation distance measure. The clusters were identified by cutree function in R with specified number of clusters. Heatmaps were plotted using gplots (Warnes et al. 2015) and pheatmap (Kolde n.d.) R packages.

4.6.4 Cell cycle analysis

Cell cycles were identified based on known gene expression profiles for each cell cycle stage as implemented by the cyclone function (Scialdone et al. 2015) in the scran package (Lun et al. 2016). This is done by assigning each cell a separate score for G0/1 and G2/M phases. Cells with G0/1 score of more than 0.5 were assigned with G0/1 phase, while cells with a G2/M score of more than 0.5 were assigned with G2/M phase. Cells with G0/1 score of less than 0.5 and G2/M score of less than 0.5 were assigned with S phase. All other cells were assigned with an unknown cell cycle phase.

4.6.5 Assessing differential gene expression analyses

The synthetic expression data used here was generated by using a Beta-Poisson model with three parameters, α, β, and γ (Wills et al. 2013; Vu et al. 2016). The model is a mixture of Poisson distributions, in which a Poisson distribution is obtained by first sampling a random value k from a beta distribution with parameters (α, β), then multiplied by k by γ to get a Poisson distribution with parameter $(k\gamma)$. The equations that describe mean and variance of a Beta-Poisson model are given below.

$$\mu = \frac{\alpha\gamma}{(\alpha + \beta)}$$

$$\sigma^2 = \frac{\alpha\beta\gamma^2}{(\alpha + \beta + 1)(\alpha + \beta)^2}$$

Custom R script was used to generate these synthetic expression data, each with 10000 genes and 200 cells, where $\alpha \sim N(-1, 0.5^2)$, $\beta \sim N(0, 0.5^2)$, $\gamma \sim N(3, 0.5^2)$. The distributions of α, β, and γ were selected empirically to give expression distributions that match observed real single-cell expression data. For each synthetic expression dataset, it contains cells separated into two sample groups, each with 100 cells. It is important to simulate the fact that the cells in the two sample groups should share similar but slightly different distributions of expression values. Therefore, for each gene, only one of out of the three parameters was multiplied by a constant in the second sample, while keeping the rest of the two parameters the same.

8 different methods were tested here, which include DESeq2, edgeR, Wilcoxon rank sum test, Kolmogorov-Smirnov test, SCDE, M3Drop, MAST, and Brennecke highly variable genes test. DESeq2 (Love et al. 2014), edgeR (Robinson et al. 2010), SCDE (Kharchenko et al. 2014), M3Drop (Andrews 2016) and MAST (Finak et al. 2015) were implemented by their authors as R packages; Brennecke highly variable genes test was implemented by their authors as an R script (Brennecke et al. 2013); while Wilcoxon rank sum and Kolmogorov-Smirnov tests were standard statistical tests implemented in base R (Team 2013).

4.6.6 Differential expression and Gene Ontology analyses

Differential expression analysis was performed using the DESeq2 R package (Love et al. 2014) with default parameters. Differentially expressed genes are defined by genes with more than log2 fold change of 1 and FDR-corrected p-value of less than 0.01. Gene Ontology (GO) analysis was performed using the topGO R package (Alexa & Rahnenfuhrer 2016). topGO implements gene set enrichment test for GO slightly differently from other GO R packages such as limma. It considers the hierarchical structure of GO when performing tests, and it places heavier emphasis on the more specific biological functions annotation over the broader annotations. Adjusted p-values corrected for multiple testing were computed using the Benjamini & Hochberg method.

4.6.7 Significance test for overlaps in Venn diagrams

The significance test for overlaps in Venn diagrams was calculated by estimating the underlying distribution of the number of overlaps among sets (i.e. a permutation test). This is done by performing 10000 iterations of randomly sampling genes for each set to see how many of these genes are common, therefore overlap, among sets. The p-value is then obtained by comparing the actual number of overlaps with the estimated distribution.

5 FLT3-ITD and IDH1-R132H mutations potentially act synergistically in acute myeloid leukaemia

RNAseq data used were generated by Konstantinos Tzelepis from George Vassiliou's lab as a collaboration.

5.1 Background

Cancer is a multistep process, which involves a succession of genetic, epigenetic and environmental events that drives the transformation of normal cells into malignant derivatives (Hanahan & Weinberg 2000; Hanahan & Weinberg 2011). Among different cancer types, this chapter focuses on the study of leukaemia, specifically the acute myeloid leukaemia. Leukaemia is the cancer of the blood or bone marrow which is usually characterised by a decrease in erythrocytes and platelets as well as an increase in leukocytes. Leukaemia is further broken down into four major categories that account for 85% of all leukaemia (Siegel et al. 2011), which include acute myeloid leukaemia (AML), chronic myeloid leukaemia (CML), acute lymphocytic leukaemia (ALL) and chronic lymphocytic leukaemia (CLL). AML involves the proliferation of myeloid cells and an arrest in their maturation, which often results in insufficient erythrocytes, platelets or granulocytes from being produced (Lowenberg et al. 1999). AML has a low overall 5-year survival rate of 30-40% (Dohner et al. 2010) and accounts for 42% of all leukaemia deaths (Siegel et al. 2011).

Similar to other cancers, the development of AML is a multistep process that requires the acquisition of multiple genetic mutations. Gilliland and Griffin proposed a model suggesting that at least two classes of mutations are required for a full-blown leukaemia, which is known as the two-hit model (Gilliland & Griffin 2002). The model consists of two classes of mutations, which are traditionally known as the class I and II mutations. Class I mutations are defined as mutations that affect signalling pathways and lead to enhanced cell proliferation, while class II mutations are defined as mutations that affect transcription factor regulations and lead to impaired cell differentiation (Takahashi 2011a; Grafone et al. 2012). Some examples of class I mutations include FLT3-ITD, FLT3-TKD and *Kit*, while some examples of class II mutations include *Runx1*, C/EBPA and MLL rearrangement.

However, recent studies discover new mutations in *Dnmt3a*, *Npm1* and *Idh1/2* that co-occur with both class I and II mutations, as well as possessing biological functions that do not fall into the definitions of class I and II mutations (Ley et al. 2010; Colombo et al. 2011; Paschka et al. 2010). Therefore, these observations suggest that AML is more complex and does not develop in just two stages with two classes of mutations as described in the simple two-hit model.

Among the mutated genes found in AML, the study in this chapter focuses on *Flt3* and *Idh1* genes, particularly on FLT3-ITD and IDH1-R132H mutations. *Flt3*, fms-related tyrosine kinase 3, encodes a membrane receptor which is a member of the class III receptor tyrosine kinase family. The activation of FLT3 protein by binding to the FL ligand leads to the activation of the PI3K/Akt and RAS/ERK pathways, which in turn help in regulating a wide range of biological processes, ranging from metabolism to proliferation (Grafone et al. 2012). The roles of *Flt3* in AML are very widely studied, because the *Flt3* mutations are one of the most frequently identified mutations in AML with approximately one-third of AML patients having mutations in this gene (Takahashi 2011b). In particular FLT3-ITD mutation is the most common *Flt3* mutation. FLT3-ITD contains an in-frame tandem duplication in the juxtamembrane domain of the gene, which results in the constitutive activation of FLT3. The presence of FLT3-ITD not only results in perturbed activation of signalling pathways regulated by wild-type FLT3, it also leads to potent activation of STAT5 signalling pathway and the inhibition of myeloid transcription factors (Takahashi 2011b) (Figure 5.1). These perturbations in turn lead to the inhibition of apoptosis and differentiation, as well as the activation of proliferation.

In contrast, *Idh1* is only discovered recently to play a role in AML and therefore is less well characterised (Mardis et al. 2009). *Idh1* codes for isocitrate dehydrogenase that catalyses the oxidative decarboxylation of isocitrate to produce α-ketoglutarate, which is important for the degradation of hypoxia-inducing factor (HIF). Besides playing an important role in cellular defence of oxidative damage, *Idh1* is also important for lipid metabolism and oxidative respiration (Reitman & Yan 2010). In AML, the IDH1 protein is usually mutated at R132 residue, which is evolutionary conserved and is located in the subtrate binding site of IDH1 protein (Bleeker et al. 2009). IDH1-R132 mutation results in more than 80% reduction in the production rate of α-ketoglutarate compared to the wild-type IDH1 protein (Zhao et al. 2009) (Figure 5.1). Interestingly, *Idh1* mutation tends to be heterozygous in tumours, which suggests that besides the loss of function, *Idh1* mutation may lead to a gain of function. This is supported by studies that show mutated IDH1 protein being capable of catalysing the conversion of α-ketoglutarate into 2-hydroxyglutarate (Dang et al. 2009; Ward et al. 2010). 2-hydroxyglutarate has been shown to interfere with the methylation process, which results in hypermethylation in AML patients (Zhao et al. 2009; Figueroa et al. 2010). Among different IDH1-R132 mutations, IDH1-R132H is the most common in glioma (90%) (Cui et al. 2016), but its frequency in AML is difficult to quantify due to the low frequency of occurrence (4.4% - 13.5%) (Byers et al. 2012).

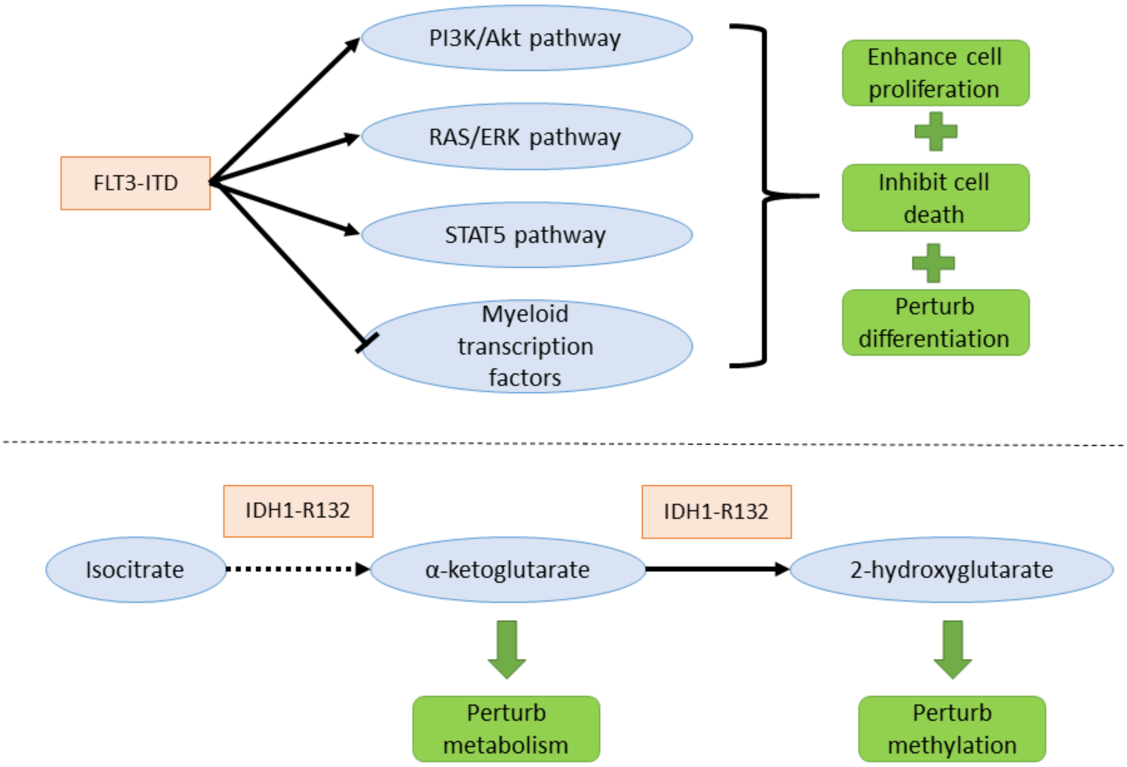


Figure 5.1 Pathways affected by FLT3-ITD and IDH1-R132 mutations.

Among the different mutations in AML, some of the mutations are found to co-occur very often, while the other mutations are mutually exclusive (Takahashi 2011a). The mutations that co-occur are assumed to have synergistic effects which confer growth and survival advantage due to gene interactions. One such example is FLT3-ITD and *Npm1* mutations which are strongly associated with one another (Thiede et al. 2006). In contrast, mutations that are mutually exclusive or co-occur at a low frequency are assumed to participate in the same biological processes, therefore are functionally redundant for the AML development. One such example is *Idh1/2* and *Tet2* mutations which are mutually exclusive (Metzeler et al. 2011).

Interestingly, studies have shown that FLT3-ITD and IDH1-R132 mutations co-occur at a low frequency that does not exceed the random expectation (Boissel et al. 2010; Schnittger et al. 2010; Andersson et al. 2011). The non-co-occurrence between FLT3-ITD and IDH1-R132 mutations suggest that they may affect the same biological process. Both FLT3-ITD and IDH1-R132 mutations were shown to upregulate HIF-1α (Jin et al. 2009; Zhao et al. 2009). HIF-1, which consists of α and β subunits, is important for cancer cell hypoxia adaptation, and is known to be overexpressed in many cancers including AML (Zhong et al. 2002). Therefore based on evidence from previous studies, it is unlikely that FLT3-ITD and IDH1-R132 mutations will interact with one another to confer increased leukaemia tendency given their non-co-occurrence and functional redundancy. However, the results presented in this chapter suggest that although both mutations may affect similar biological processes, the presence of both mutations will lead to perturbed expression profile that is very different from the expression profiles of just having each mutation individually.

Single-cell RNAseq was generated from mice haematopoietic stem cells (HSCs) and lymphoid-primed multipotential progenitors (LMPPs) with FLT3-ITD and IDH1-R132H mutations (Figure 5.2). By studying the single-cell RNAseq data, this chapter aims to investigate the expression profiles of FLT3-ITD and IDH1-R132, and whether these two mutations predispose some cells for developing AML. In addition, it would also be interesting to investigate if the two mutations have synergistic effects when both of them are present in the same cells. Section 5.2 firstly describes the quality control and pre-processing of RNAseq data to reduce technical bias for downstream analyses. Section 5.3 uses single-cell RNAseq to investigate the expression profiles and AML predisposition of FLT3-ITD and IDH1-R132 cells. The chapter then ends with conclusions in Section 5.4, and materials and methods in Section 5.5.

Note that in the rest of the chapter, cells with FLT3-ITD mutation are simply referred to as FLT3 cells, cells with IDH1-R132 mutation are referred to as IDH1 cells, while cells with both mutations are referred to as FLT3-IDH1 cells.

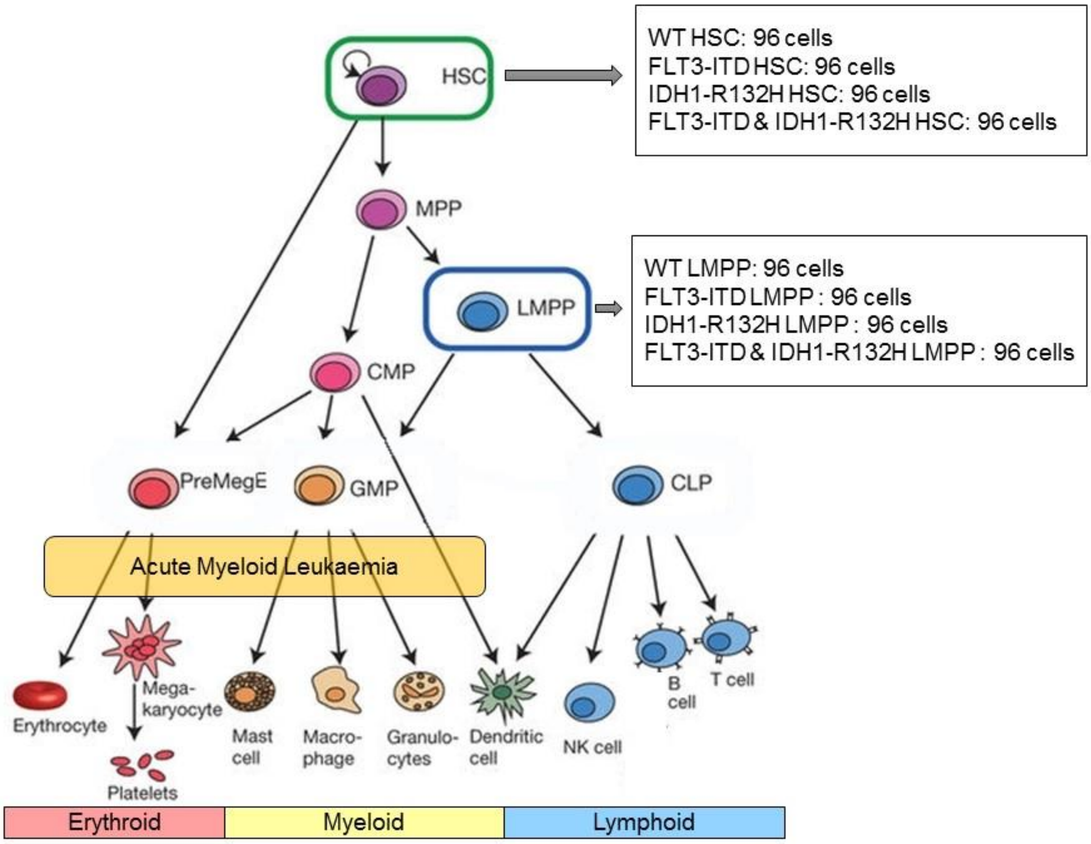


Figure 5.2 Overview of haematopoiesis.
The brackets indicate samples used for single-cell RNA sequencing. [Figure adapted from (Moignard et al. 2013)]

5.2 Pre-processing of RNAseq data

5.2.1 Performing quality control

As in Section 4.3.1, quality controls were performed on the RNAseq data on the raw sequencing reads, on the aligned reads, and on the counted reads. The quality of the raw sequencing reads as shown by the FastQC program (Andrews S 2010) is illustrated by using a pool of all 96 wild type HSCs (Figure 5.3). Using a pool of cells overcomes the potential pitfall of picking random single cells that may exhibit extreme properties by chance, and speeds up the computation process of quality control. Only a pool of cells is shown here for illustration purpose, as all other pools of cells by samples showed very similar technical properties.

The quality control in Figure 5.3 suggests that there is no fundamental technical problem with the single-cell RNAseq data. For detailed explanations of the meaning of each sub-figure, please refer to Section 4.3.1 and Figure 4.6. The results in Figure 5.3 agreed with the results in Figure 4.6, with the main differences due to slightly different protocols and the pooling of cells. The library construction protocol and sequencing machine used were the same for the single-cell RNAseq data in Chapter 4 and in this chapter. This single-cell RNAseq data has 100 base pairs and has paired ends, while the previous single-cell RNAseq data in Chapter 4 has 50 base pairs and has single ends.

Quality control was performed on the aligned reads using the same pool of all 96 wild type HSCs by RSeQC program (Wang et al. 2012) (Figure 5.4). For detailed explanations of the meaning of each sub-figure, please refer to Section 4.3.1 and Figure 4.7. The quality of the aligned reads indicates that there is no fundamental technical problem, and the results in Figure 5.4 are similar with the results in Figure 4.7. The main differences are in the results of gene body coverage and mutation profile, which are smoother in Figure 5.4 due to the pooling of cells. The extra analysis available here is the distribution of the mRNA insert size (Figure 5.4B) due to the use of paired ends in this data. The result indicates that the mRNA insert size is small for most reads (mean = 6.8 bp) with a lot of the paired end reads overlapping, which is indicated by the negative insert size values.

Lastly quality control was performed on the counted reads, which was used to remove outlier cells that exhibit unusual technical properties (Figure 5.5). For detailed explanations of the meaning of each sub-figure, please refer to Section 4.3.1 and Figure 4.8. Similar to before, filtering thresholds were set on five technical properties in Figure 5.5. 149 cells were removed from downstream analysis based on the total reads (less than 0.5 million reads per cell), fraction of mapped reads (less than 50% mapped reads per cell), fraction of spike-in reads (more than 25% spike-in reads per cell), fraction of mitochondrial reads (more than 10% mitochondrial reads per cell) and number of genes with more than 10 reads per million (less than 1000 genes per cell). Note that for plate 3, which contains all FLT3-IDH1 HSCs, was known to possess 41 empty wells that do not contain a cell due to the low number of HSCs available in FLT3-IDH1 mice. These 41 empty wells were detected as outlier cells within the quality control in Figure 5.5 and were removed. The presence of empty wells also lead to the greater sequencing depth experienced by FLT3-IDH1 HSCs relative to the other cells, which will be corrected by normalisation.

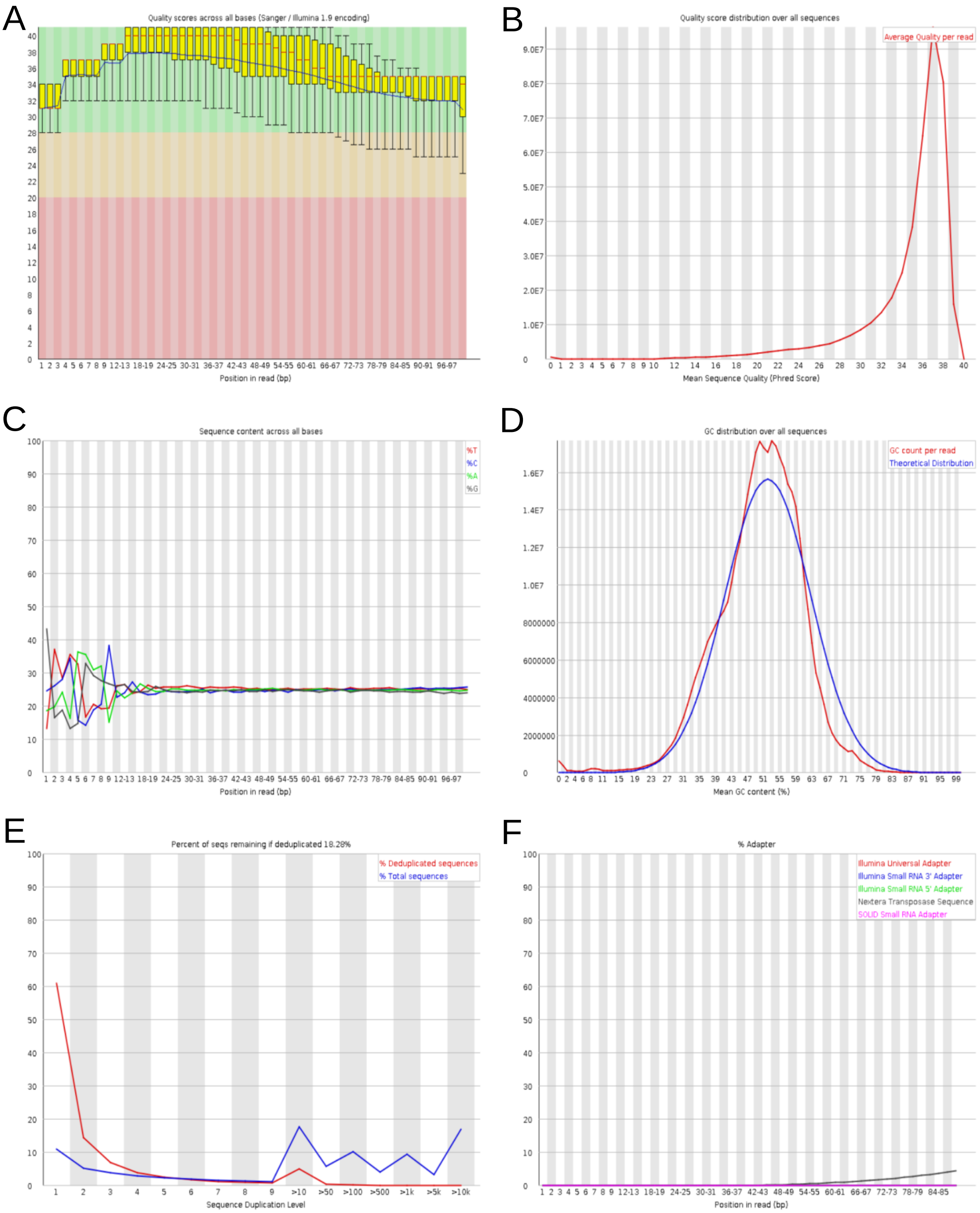


Figure 5.3 FastQC results on the quality of raw sequencing reads of a pool of 96 WT HSCs single-cell RNAseq.

(A) Per base sequence quality, (B) Per sequence quality scores, (C) Per base sequence content, (D) Per sequence GC content, (E) Sequence duplication levels, (F) Adapter content.

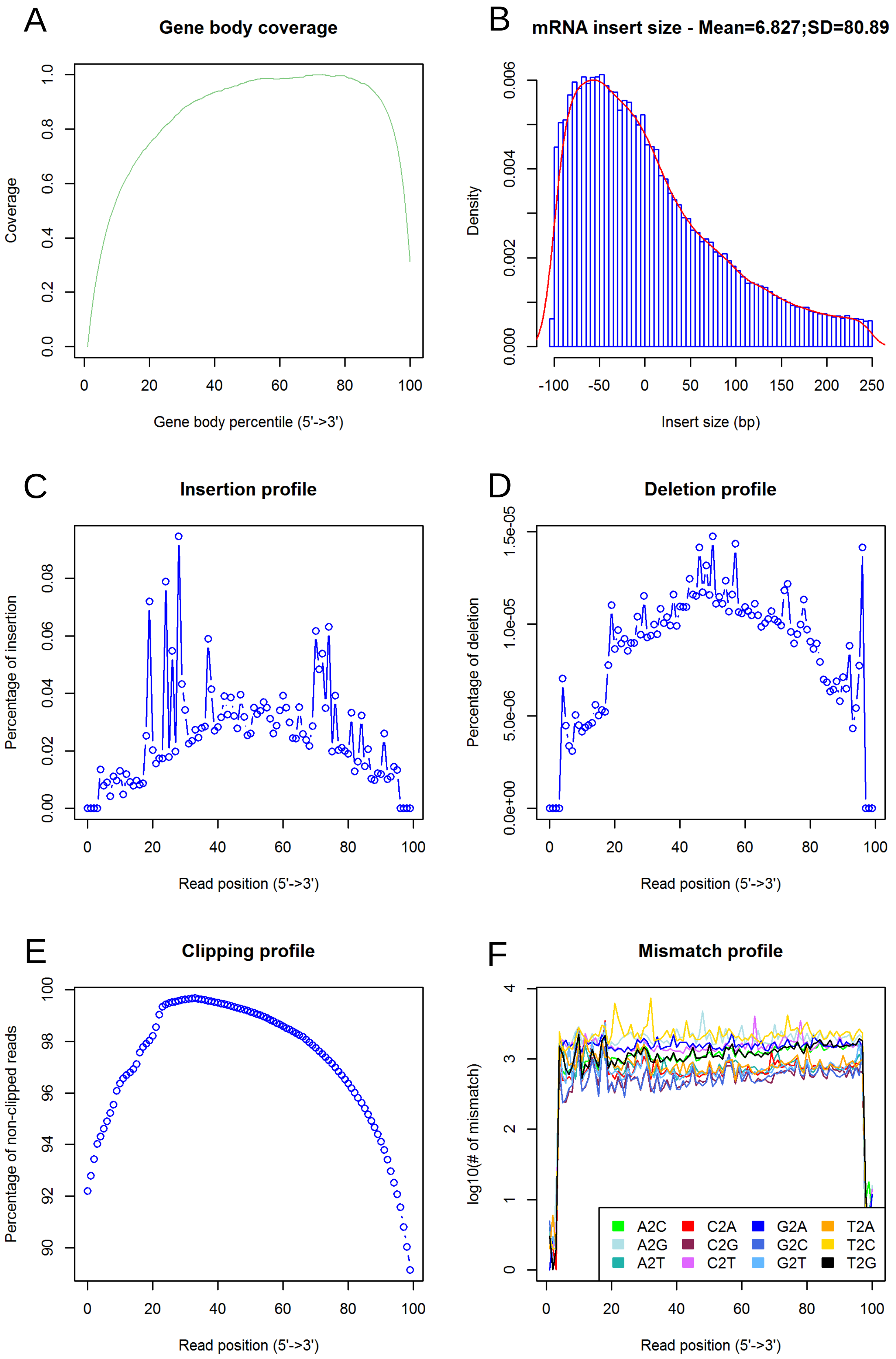


Figure 5.4 RSeQC results on the quality of aligned reads of a pool of 96 WT HSCs single-cell RNAseq.
(A) Gene body coverage, (B) mRNA insert size, (C) Insertion profile, (D) Deletion profile, (E) Clipping profile, (F) Mismatch profile.

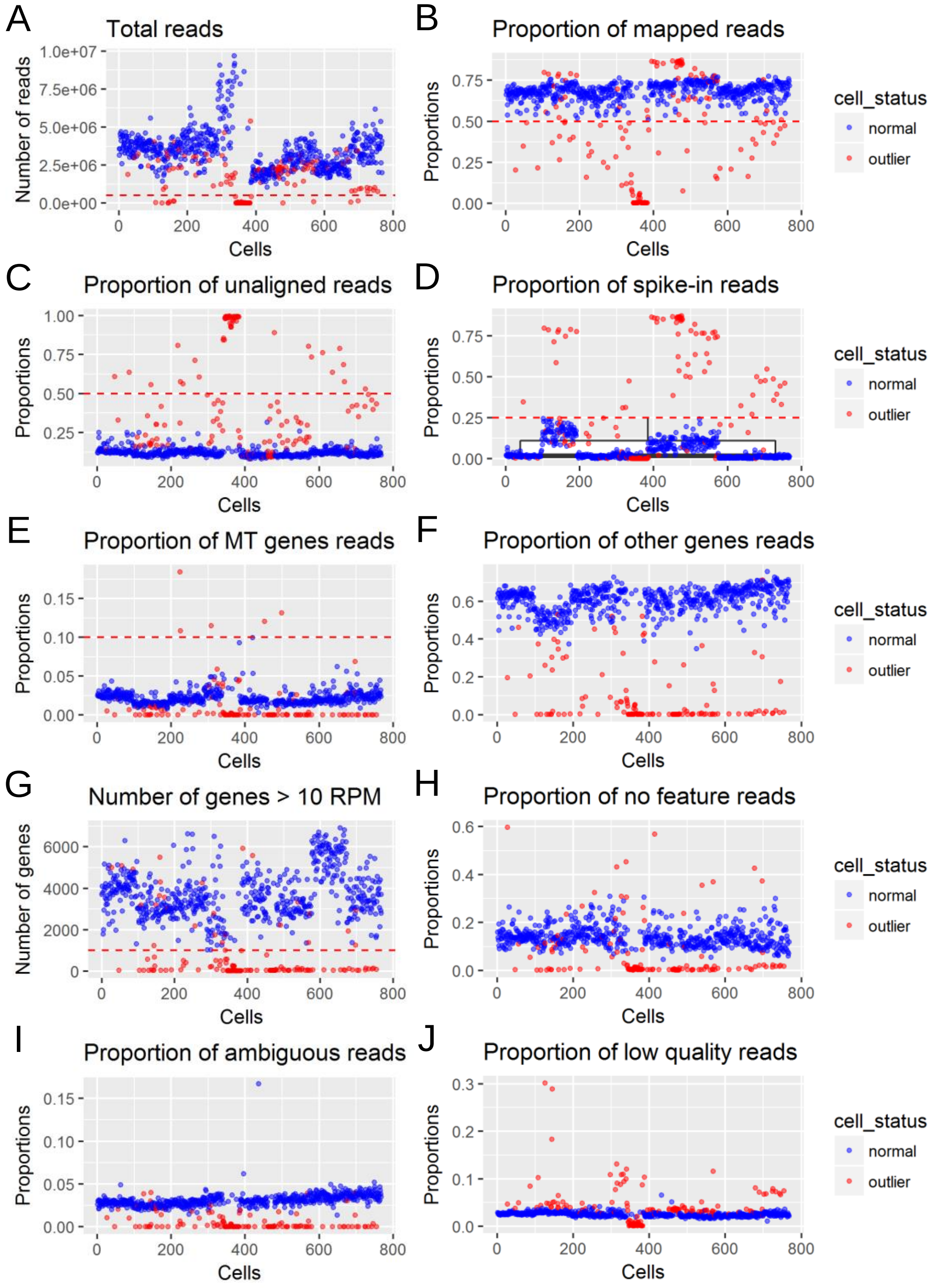


Figure 5.5 Quality control of counted reads in all cells.
Each dot in the plot corresponds to a cell. Each cell always has the same index in the x-axis across the plots. The red dashed line indicates the threshold used where a cell is labelled as an outlier.

5.2.2 Quantifying wild type and mutant reads

Since the cells in this experiment are heterozygous for the two genes, *Flt3* and *Idh1*, the RNAseq reads for the two genes consist of a mix of wild type and mutant reads. It would be interesting to verify that the mutant alleles were indeed expressed in the mutant cells, where the expression of both the wild type and mutant alleles for each gene should be at a similar level. The number of wild type and mutant reads were obtained by aligning all RNAseq reads against a custom-built genome index that contains both the wild type and mutant sequence versions of both genes. Despite careful considerations when generating the custom genomic index, it is possible that the sequence aligner software used may not be able to accurately quantify the number of wild type and mutant reads, due to the high degree of similarity between wild type and mutant sequences. In terms of the mutant sequences, FLT3-ITD contains repeated short sequences that are also present in wild type Flt3 sequence. As for IDH1-R132H, it contains mostly single nucleotide polymorphisms, with the rest of the sequence being identical to the wild type *Idh1* sequence.

Despite expecting similar expression levels for both wild type and mutant alleles, the actual reads detected for each allele are different as shown in Figure 5.6. Generally, HSCs showed lower expression of both genes relative to LMPPs, and mutant reads were only detected in mutant cells except in LMPPs, where *Flt3* mutant reads were detected in wild type LMPPs and IDH1 LMPPs. It is likely that these represent technical errors, where a proportion of wild type reads was wrongly recognised as mutant reads by the sequence aligner, due to the high level of *Flt3* expression observed in LMPPs and the similarity in sequences between the wild type and mutant alleles. The index switching issue in the Illumina HiSeq 4000 sequencing machine as reported by (Sinha et al. 2017) is not the cause here as each sample with different mutations is sequenced in a separate lane. In summary, the result confirmed that mutant alleles were indeed expressed in the mutant cells, and were not silenced epigenetically by the cells.

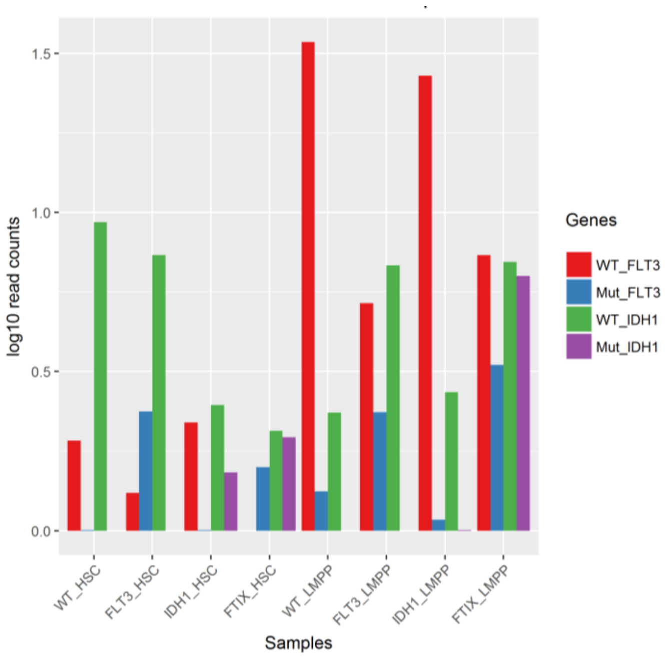


Figure 5.6 Wild type of mutant reads for Flt3 and Idh1 genes in all cell types.
Note that FTIX represents FLT3-IDH1 double mutant.

5.2.3 Selecting the most suitable normalisation method

Three normalisation methods were evaluated on this data, namely counts per million (CPM), DESeq (Anders & Huber 2010) and scran (L. Lun et al. 2016) normalisation methods. DESeq normalisation method was tested with two configurations, in which the location estimator used is different, namely the default median and the shorth estimator. Scran normalisation method was tested with two configurations, in which the clustering method used is different, namely the default Spearman correlation-based hierarchical clustering, and by considering all the cells as a single cluster. Please refer to Section 4.3.2 for more detailed discussions on each of the methods.

Figure 5.7 shows that in general, scran normalisation gave more similar library sizes among the different samples, when compared to DESeq normalisation. Note that CPM normalisation always restricts all samples to the same library size. However, instability of results occurred with both scran and DESeq normalisation methods, where a few cells are highly amplified in terms of the number of reads after normalisation. The same highly amplified cells did not possess high number of reads before normalisation.

The performance of the normalisation methods was further assessed by checking the results of the principal component analysis (PCA) (Figure 5.8) and cell-wise relative log expression (RLE) (Figure 5.9). The ideal normalisation method should provide clear separation among samples in the PCA and give reduced spread in the RLE relative to the separation and the spread of the raw reads. Interestingly, the use of different normalisation methods had very little effects on the spatial layout of cells on the PCA. This suggests that PCA results are not very informative on selecting the most suitable normalisation method. In terms of the RLE results, the normalisation methods similarly did not have a large impact on reducing the spread of normalised RLE values relative to the raw RLE values. Out of all normalisation methods, CPM gave the largest reduction of spread in RLE values, and relatively little outlier cells. Lastly, the relationship between read counts and gene lengths was also investigated (Figure 5.10). There was no obvious relationship between read counts and gene lengths, which suggests that there is no need to correct for gene lengths. The lack of relationship is likely to be due to the use of sequencing library generation protocol that favours capturing the 3’ end of a transcript.

Taken together, the results suggest that there is no one normalisation method that is distinctly better than the others. CPM was chosen for normalising the data for downstream analyses, because it is the most conservative normalisation method with the fewest assumptions and it does not generate highly amplified outlier cells that are present in other methods.

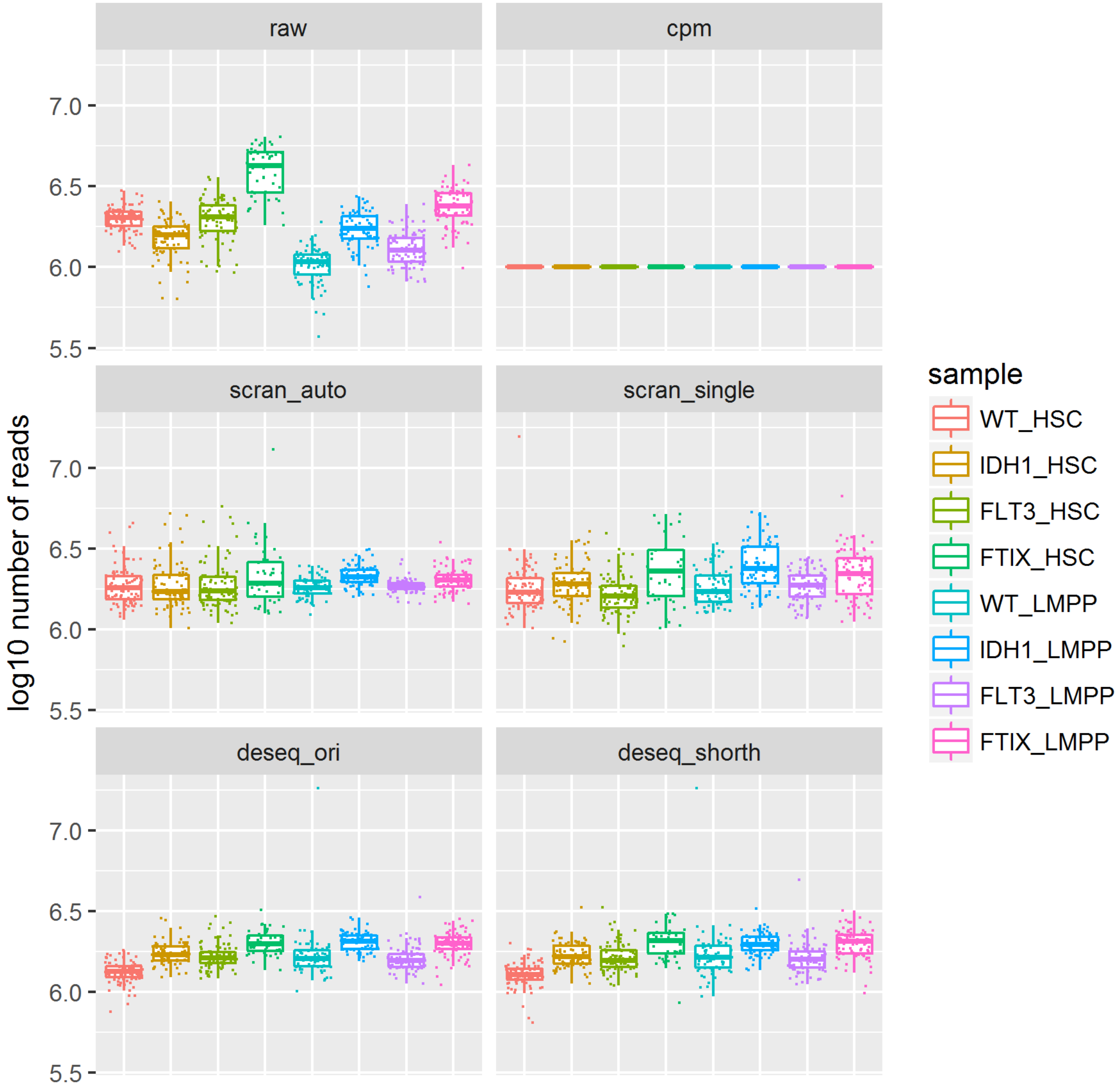


Figure 5.7 Raw and normalised reads using different normalisation methods.
Y-axis scale is fixed to allow for comparisons among different methods. CPM, counts per million; scran_auto, Scran with default clustering; scran_single, Scran with a single cluster; deseq_ori, DESeq with default median; deseq_shorth, DESeq with shorth estimator.

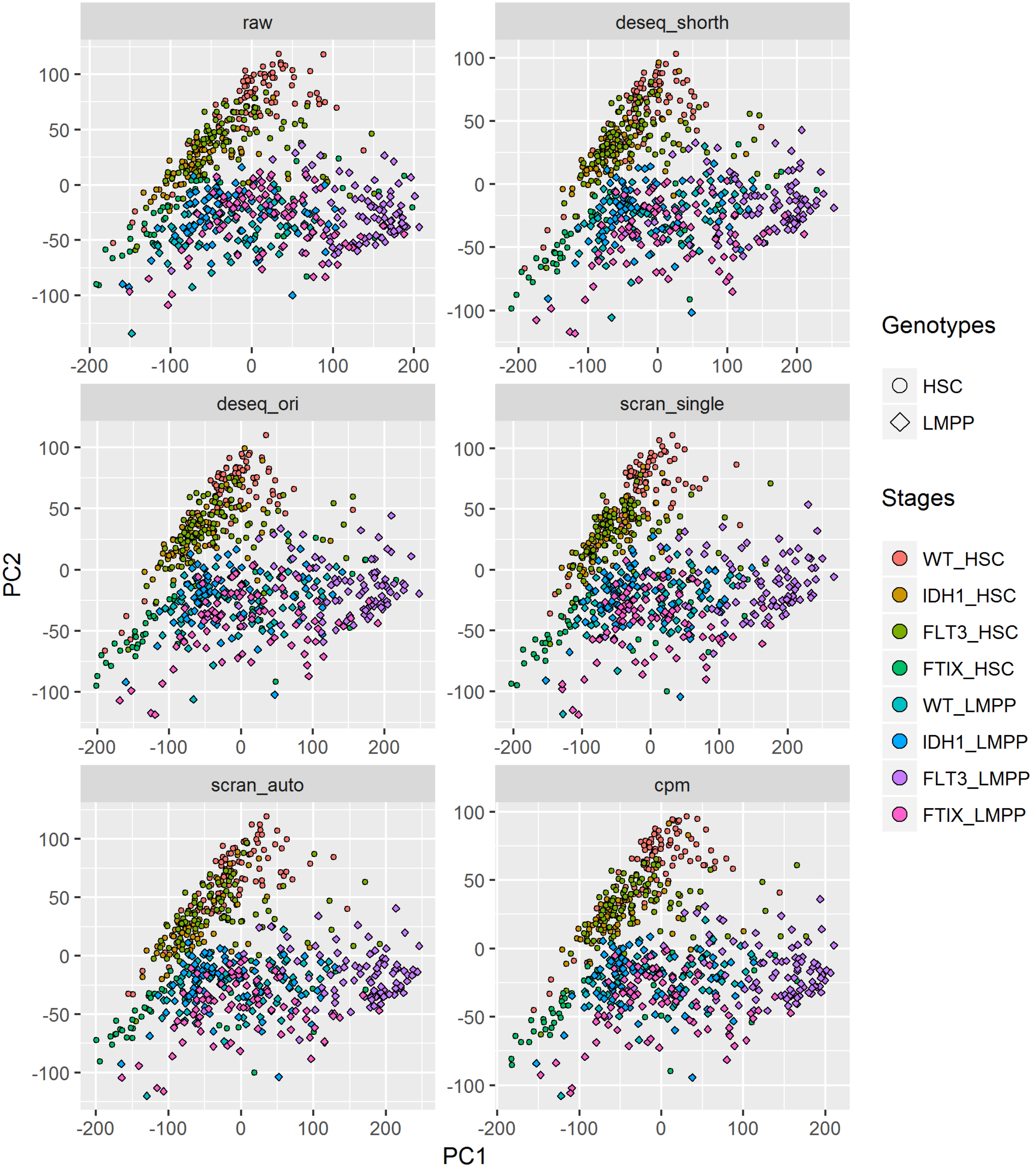


Figure 5.8 Principal component analysis of raw and reads normalised with different normalisation methods.
CPM, counts per million; scran_auto, Scran with default clustering; scran_sample, Scran with cells clustered by samples (i.e. cell type and genotype); scran_single, Scran with a single cluster; deseq_ori, DESeq with default median; deseq_shorth, DESeq with shorth estimator.

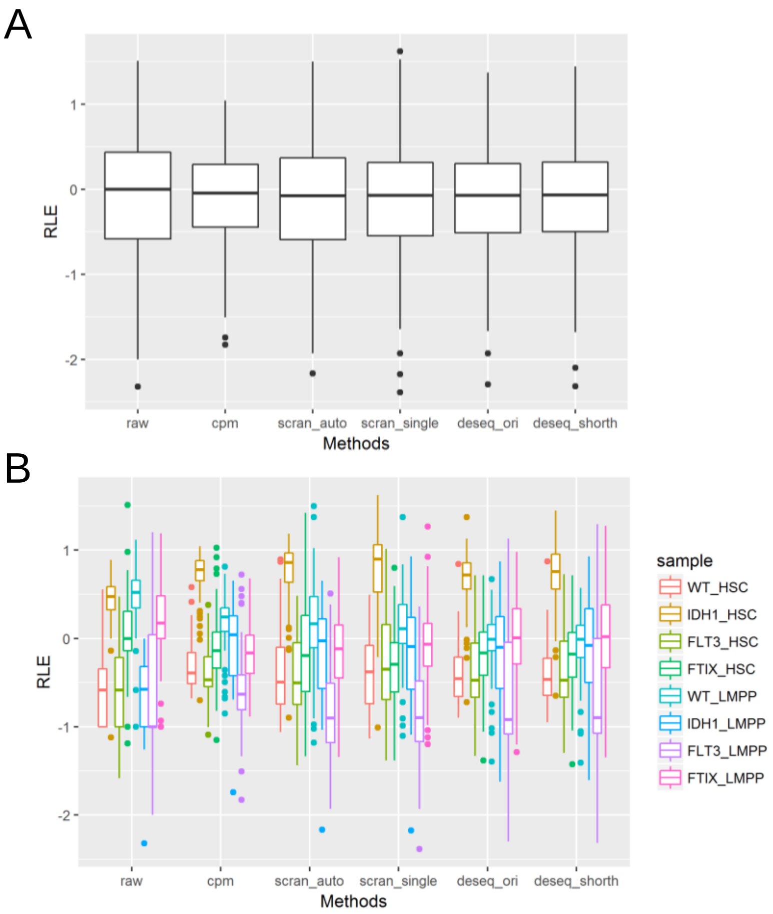


Figure 5.9 Cell-wise relative log expression of raw and reads normalised with different normalisation methods.

CPM, counts per million; scran_auto, Scran with default clustering; scran_sample, Scran with cells clustered by samples (i.e. cell type and genotype); scran_single, Scran with a single cluster; deseq_ori, DESeq with default median; deseq_shorth, DESeq with shorth estimator.

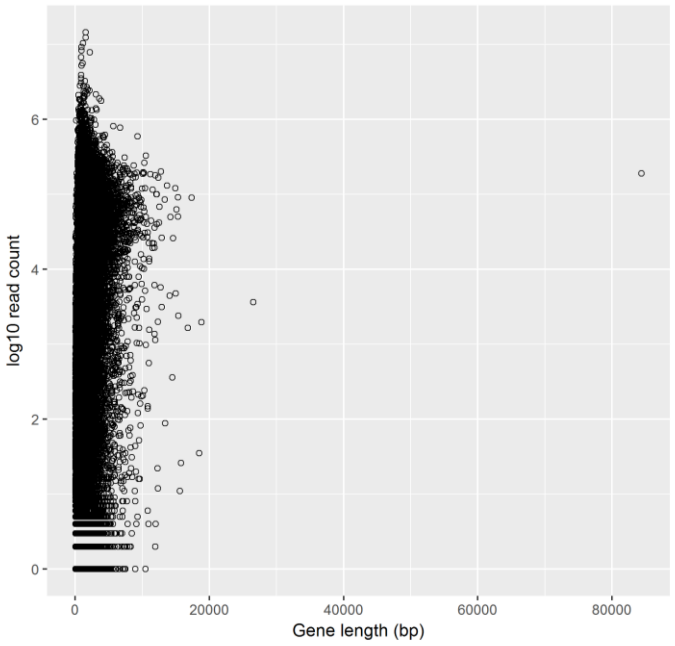


Figure 5.10 Relationship between log10 read counts and gene lengths.

5.2.4 Investigating potential batch effects

After correcting for library sizes through normalisation, it would be important to account for other types of technical noises or confounding biological effects, such as the sequencing plates and the cell cycle phases. Differences in plates and cell cycle phases are known to cause variations in the gene expression profile of each cell, which may lead to the detection of spurious or non-interesting subpopulations of cells. There is a total of eight plates in this experiment, with each plate containing 96 cells from a single sample, which corresponds to a unique pair of cell type (i.e. HSC and LMPP) and genotype (i.e. wild type, FLT3, IDH1, FLT3-IDH1) (Figure 5.2). In terms of the batch effect caused by the plates, this experiment was unfortunately designed to have all cells from each unique pair of cell type and genotype to locate on a separate plate. This caused each plate to differ from all other plates by both the biological variables of interest as well as the confounding plate effect, therefore making it very difficult to detect or correct for the confounding plate effect without adversely interfering with the effects of interest exerted by the cell type and the genotype. In order to prevent biasing the results due to the reason discussed above, any potential effect caused by the plates was not corrected in this single-cell RNAseq dataset.

5.3 Exploration of single-cell RNAseq data

5.3.1 FLT3-ITD has a more proliferative cell cycle profile compared to IDH1-R132H

Before performing other downstream analyses, it is important to firstly investigate and account for any potential cell cycle effect in the single-cell RNAseq data. Interestingly, cell cycle effect can be a biological variable of interest or a confounding biological variable depending on the aim of the experiment. Since the system studied here is a model system of cancer for the acute myeloid leukaemia, cell cycle effect is of biological interest. In addition, it is also important to account for its effect as it is likely to contribute to subpopulation of cells which may be detected in downstream analyses. In order to account for the cell cycle effects, the cell cycle phase of each cell was inferred and assigned to either G0/1, G2/M, S or unknown phase as described in Section 4.6.4.

The proportion of cells assigned to each cell cycle phase is illustrated in Figure 5.11. There seems to be a higher proportion of G0/1 phase cells in HSCs compared to LMPPs. It is likely that the HSCs are mostly dormant as their role is to act as a reserve of potent stem cells, while LMPPs are more actively dividing to generate more differentiated blood cells. The presence of FLT3-ITD mutation seemed to increase the number of proliferating cells, while IDH1-R132H mutation exerted less pronounced effect on the cell cycle phases of the cells. Interestingly, HSCs proliferate a lot less than ESCs and iPSCs investigated in Chapter 4 (Figure 4.16). This may be due to technical differences between the experiments, or it may reflect the functional differences of embryonic development-related and adult maintenance-related stem cells. Stem cells in embryonic development should proliferate quickly to contribute to the rapid developmental process with less emphasis on maintaining a pool of potent cells, while maintaining a pool of static potent cells is more important for adult maintenance-related stem cells.

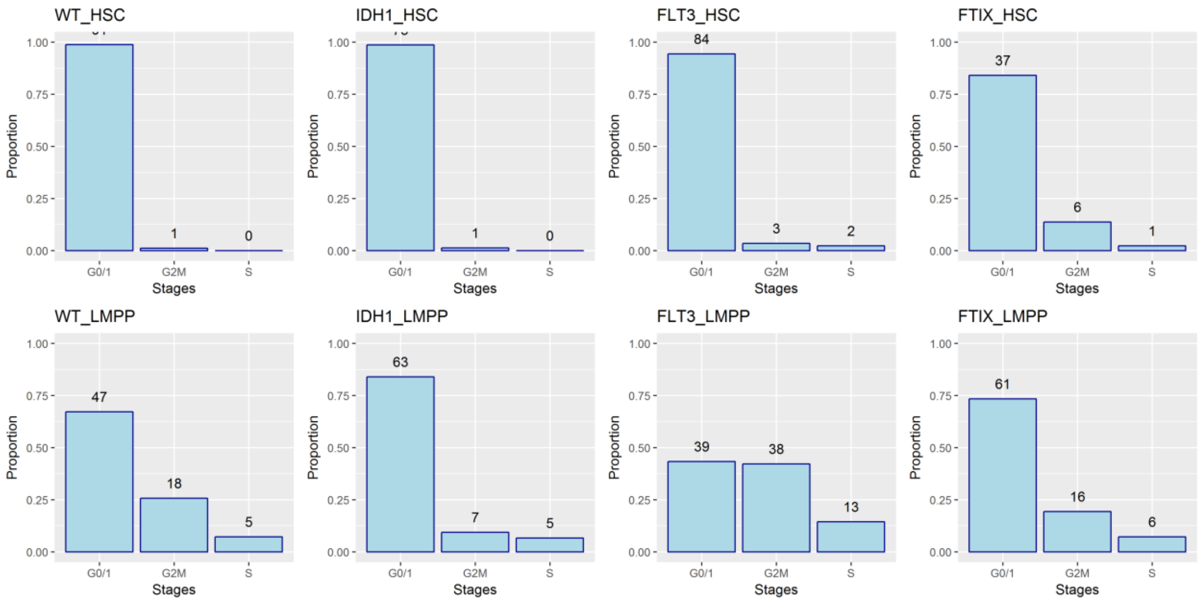


Figure 5.11   Number of cells per cell cycle stage.
Cell cycles were identified based on known gene expression profiles for each cell cycle stage. FTIX refers to FLT3-IDH1.

Next it is important to verify if the cell cycle phases contribute to the formation of subpopulation of cells, which can be detected and visualised with dimensionality reduction analyses (Figure 5.12). PCA can detect very distinct subpopulations, while tSNE is more sensitive than PCA in terms of detecting subpopulations. Firstly by just examining the PCA and tSNE results labelled by samples, it can be seen that HSCs and LMPPs were well separated in general. In HSCs, it is possible to see a gradual separation of single mutant cells leading up to the double mutant cells. Similar separation was not observed in LMPPs. The separation of cells in LMPPs is better visualised with tSNE, where IDH1 LMPPs were similar with wild type LMPPs, followed by FLT3-IDH1 mutant LMPPs and FLT3 LMPPs which are the most different. This suggests that in LMPPs, IDH1-R132H mutation had relatively little effect on the expression profiles compared to IDH1-R132H mutation in HSCs.

In terms of cell cycle phase difference, the results in Figure 5.12B and D indicate that both PCA and tSNE were separating the cells by cell cycle phase difference in the first dimension for PCA and the second dimension for tSNE. However, it should also be noted that the most of the HSCs were assigned to G0/1 phase, while LMPPs have a higher proportion of cells assigned to G2/M and S phase. This indicates that the cell cycle phase difference may be partially confounded with cell type difference. In addition, the presence of subpopulations within each sample seems to be driven by differences in cell cycle phases. This can be seen in wild type and IDH1 LMPPs, where both samples were separated into two clusters due to differences in cell cycle phases.

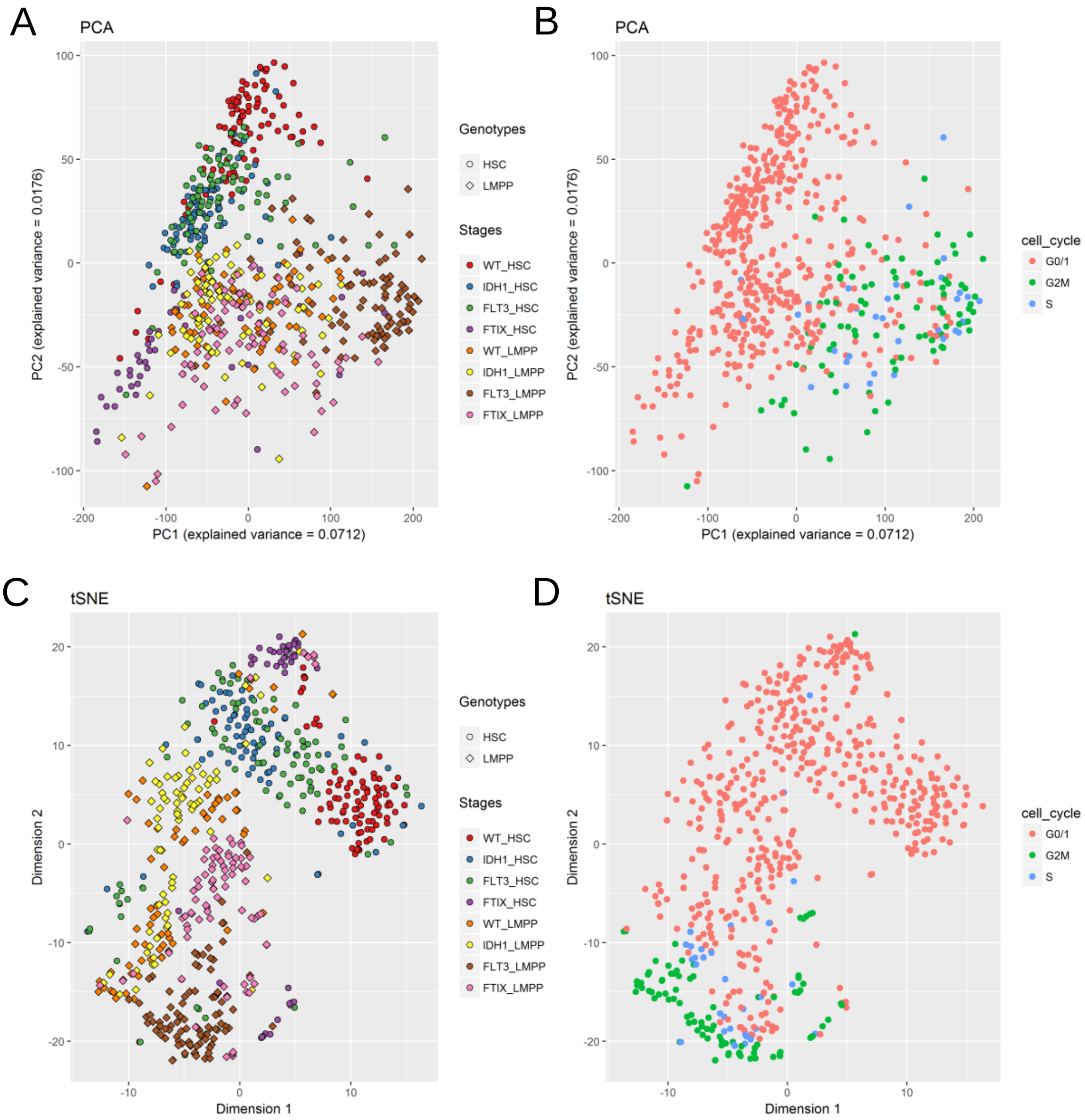


Figure 5.12 Dimensionality reduction analyses with cell cycle phase information. *(A) PCA with cell type and genotype labels, (B) PCA with cell cycle phase labels, (C) tSNE with cell type and genotype labels, (D) tSNE with cell cycle phase labels.*

As the differences in samples cannot be resolved clearly when both HSCs and LMPPs were combined, tSNE was employed to analyse the HSCs and LMPPs separately (Figure 5.13). Similar to observations made before, the results of tSNE suggest that most of the potential subpopulations of cells are separated by differences in cell cycle phases, which is particularly obvious for FLT3-IDH1 HSCs and FLT3 LMPPs. The separation of wild type, single and double mutants was also as observed before, where the most distinct samples in HSCs and LMPPs are the FLT3-IDH1 HSCs and the FLT3 LMPPs respectively.

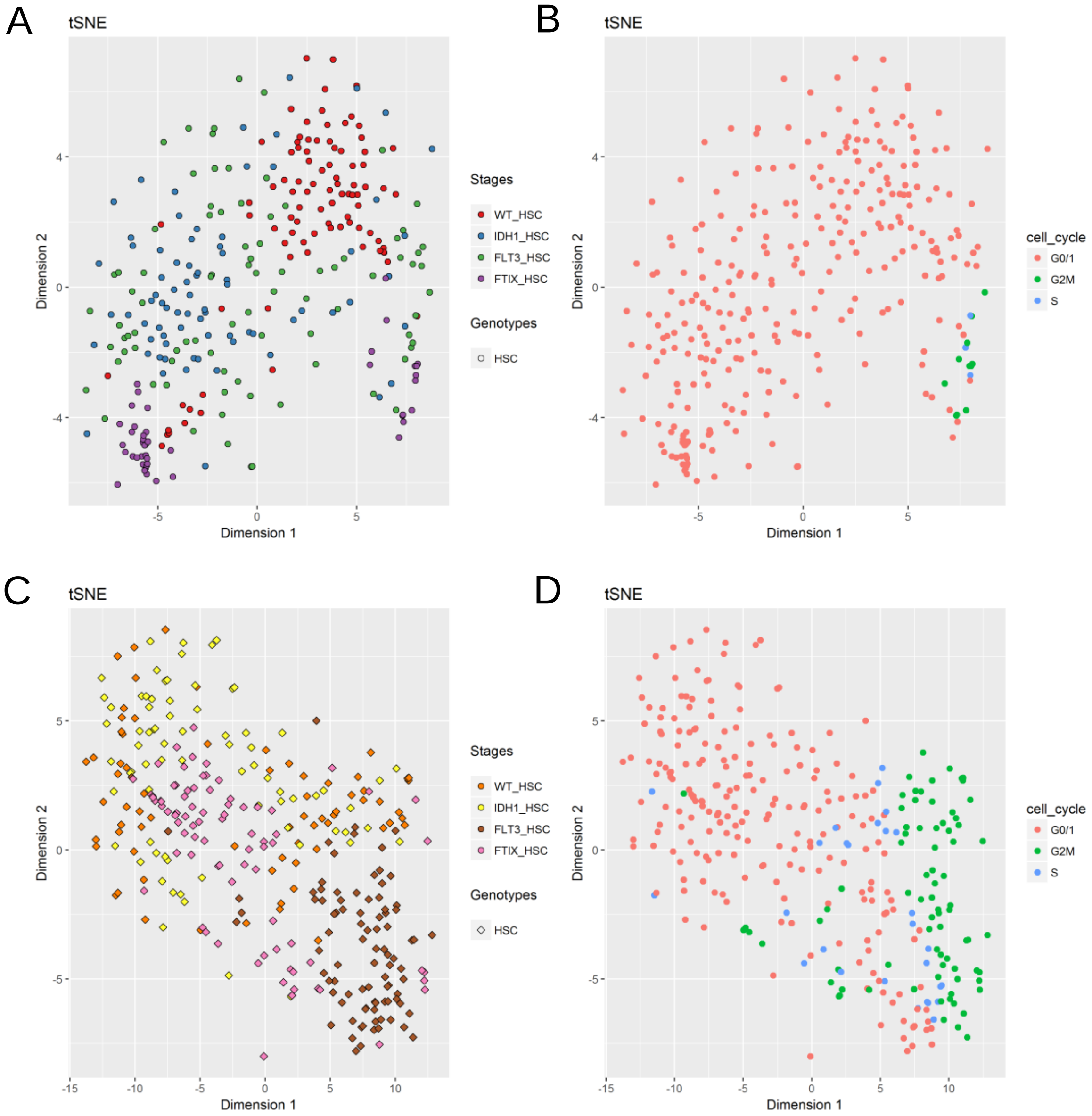


Figure 5.13 tSNE with cell cycle phase information.
(A) HSCs with cell type and genotype labels, (B) HSCs with cell cycle phase labels, (C) LMPPs with cell type and genotype labels, (D) LMPPs with cell cycle phase labels.

The results discussed above suggest that cell cycle phase difference is likely to be an important variable to consider when interpreting results from downstream analyses. The cell cycle difference can either be corrected or accounted for when performing downstream analyses. In terms of correcting for cell cycle effects, there are many methods, such as removing all cell cycle genes from downstream analyses or using model-based methods to regress out the cell cycle effects. The result of removing all cell cycle genes was investigated through PCA and tSNE (results not shown). The results indicate that removing cell cycle genes does not exert much effect on the PCA and tSNE analyses. It is likely that the variations in cell cycle genes were also associated with variations in other genes closely related to cell cycle, such as DNA replication or translation machinery related genes. Therefore, removing cell cycle genes alone is not enough to remove the effects exerted by differences in cell cycles.

There are a few algorithms for correcting cell cycle effect with model-based methods, such as sva (Leek et al. 2012) and scLVM (Buettner et al. 2015). sva, which is designed for microarray and RNAseq, uses a regression model to regress out the cell cycle effect; while scLVM, which is designed for single-cell RNAseq, uses a Bayesian latent variable model to locate the latent variable that represents cell cycle effect, which can then be accounted for. The method implemented in sva was tested on this dataset, and led to distortion in the expression values possibly due to sva not being designed for single-cell RNAseq data. The method implemented in scLVM is able to detect and remove latent variables, but a recent study has shown that it may be difficult to attribute the latent variables detected to be the cell cycle effect (McDavid et al. 2016). The use of model-based methods to correct for cell cycle may introduce unintended bias into the expression values, as the cell cycle phases in this dataset were not evenly distributed, with a high proportion of non-G0/G1 phase cells in FLT3-IDH1 HSCs and FLT3 LMPPs. In addition to the reasons discussed above, differences in cell cycles may be of biological interests in this experiment, as the system models acute myeloid leukaemia, which possesses perturbed proliferation and differentiation potentials.

In summary, HSCs were shown to be less actively dividing than LMPPs. FLT3-ITD mutation results in more active cell proliferation compared to both wild type and IDH1-R132H mutation. In addition, this single-cell RNAseq data was not corrected for cell cycle effect by transforming the expression values given the reasons discussed above. However, the cell cycle phases were instead accounted for when performing differential gene expression analysis to identify non-cell cycle related genes that may be differentially regulated by the mutations.

5.3.2 FLT3-ITD and IDH1-R132H act synergistically to promote perturbed immune cell differentiation

Before attempting to locate subpopulations in the single-cell RNAseq, it is important to understand the effects exerted by FLT3-ITD and IDH1-R132H mutations at the population level by considering all single cells from each sample. This is especially true for the IDH1-R132H mutation, which is less well studied compared to the FLT3-ITD mutation. Differential gene expression analysis was performed by comparing each mutant sample to the wild type sample, while accounting for the cell cycle phases and the potential interactions between FLT3-ITD and IDH1-R132H mutations (Figure 5.14 & Section 5.5.2). The differential gene expression analysis identified 303 and 421 upregulated genes in HSCs and LMPPs respectively, as well as 949 and 449 downregulated genes in HSCs and LMPPs respectively, across all pairs of comparison. The result indicates that most of the differentially expressed genes were uniquely differentially expressed between FLT3 and IDH1 single mutants. In addition, many genes were only differentially expressed in FLT3-IDH1 double mutants, suggesting that the interactions between FLT3-ITD and IDH1-R132H mutations led to unique perturbed gene expression profile. Note that for FLT3-IDH1 double mutants, the differential expression analysis was setup such that only genes that are differentially expressed in FLT3-IDH1 double mutants relative to both single mutants were considered. The result supports the hypothesis that FLT3-ITD and IDH1-R132H mutations may have a synergistic effect in the development of AML.

Gene Ontology (GO) analysis was then performed on the differentially expressed genes to infer enriched biological processes (Table 5.1-Table 5.4 & Figure 5.15). GO analysis offers a more comprehensive view than differential expression analysis in terms of illustrating sample differences, as GO analysis considers sets of genes rather than each gene separately. Figure 5.15 offers an overview of the similarities among upregulated and downregulated biological processes in all samples. In general, all mutant samples did not share overlapping perturbed biological processes, except between FLT3 and IDH1 samples. This suggests that FLT3-ITD and IDH1-R132H mutations are affecting a subset of similar biological processes, in which the largest proportion of overlapping upregulated biological processes is in HSCs. The shared upregulated biological processes between FLT3-ITD and IDH1-R132H mutations in HSCs were mostly related to myeloid development and immunological processes, which suggests that FLT3-ITD and IDH1-R132H mutations by themselves may be increasing the myeloid potential of HSCs and LMPPs.

Besides myeloid potential-related processes, FLT3-ITD and IDH1-R132H mutations affect other interesting biological processes as well. In HSCs, FLT3 HSCs downregulated protein processing and membrane transport-related functions (Table 5.2). Aberrant expression profiles for membrane transport-related genes have been observed in multiple studies on AML patients, which are associated with poor clinical outcomes (Chigaev 2015). Changes in membrane transport may relate to changes in energy consumption and pH regulation, which are known to be important for cancer. This is because tumour growth tends to require a large amount of energy which produces an acidic environment through increased anaerobic respiration (Diaz-Ruiz et al. 2011; Parks et al. 2013). As for IDH1 HSCs, they downregulated cell cycle and immune cells related functions (Table 5.2). The cell cycle-related functions were present despite accounting for cell cycle phase differences when performing differential expression analysis. This suggests that IDH1 HSCs possess perturbed expression profiles and cell cycle stages that cannot be recognised correctly by the cyclone cell cycle classifier. Note that cell cycle-related functions were only enriched in IDH1 HSCs compared to all other samples. In FLT3-IDH1 HSCs, immune system related functions were perturbed, with the upregulation of immune surface proteins and the downregulation of lymphoid lineage related processes, particularly on T cell development (Table 5.2). This suggests that the presence of both FLT3-ITD and IDH1-R132H mutations perturbed normal lymphoid development.

It was interesting to see if the same mutations exert similar or different effects in LMPPs, when compared to HSCs. Similar to HSCs, FLT3 LMPPs show perturbed myeloid development and immunological functions (Table 5.3 & Table 5.4). The effect of FLT3-ITD mutation is stronger in LMPPs than in HSCs based on the higher number of differentially expressed genes in LMPPs. Interestingly, IDH1 LMPPs do not share perturbed biological processes with FLT3 LMPPs (Figure 5.15). Instead, IDH1 LMPPs upregulated metabolism-related processes, as well as downregulated cell migration and differentiation (Table 5.3 & Table 5.4). The wide categories of biological processes affected are consistent with the understanding that IDH1-R132H mutation led to perturbed chromatin modifications. This is also supported by the GO terms associated with chromatin modifications such as meiotic division and genetic imprinting (Table 5.4). In FLT3-IDH1 double mutant LMPPs, cell motility and Notch signalling pathway were upregulated, while apoptosis was downregulated. Notch has been shown to have a tumour suppressive role in AML (Kannan et al. 2013; Lobry et al. 2013; Kato et al. 2015).

There are several observations that can be taken from these results. Firstly, the same mutations exert slightly different effects in HSCs and LMPPs. Secondly, when considering the single mutants, FLT3-ITD mutation was shown to possess increased myeloid potential with perturbed differentiation, while IDH1-R132H mutation was shown to cause perturbed chromatin modifications which lead to a wide range of perturbed developmental processes. Lastly, FLT3-ITD and IDH1-R132H mutations were shown to potentially interact and act synergistically to promote perturbed immune cell differentiation.

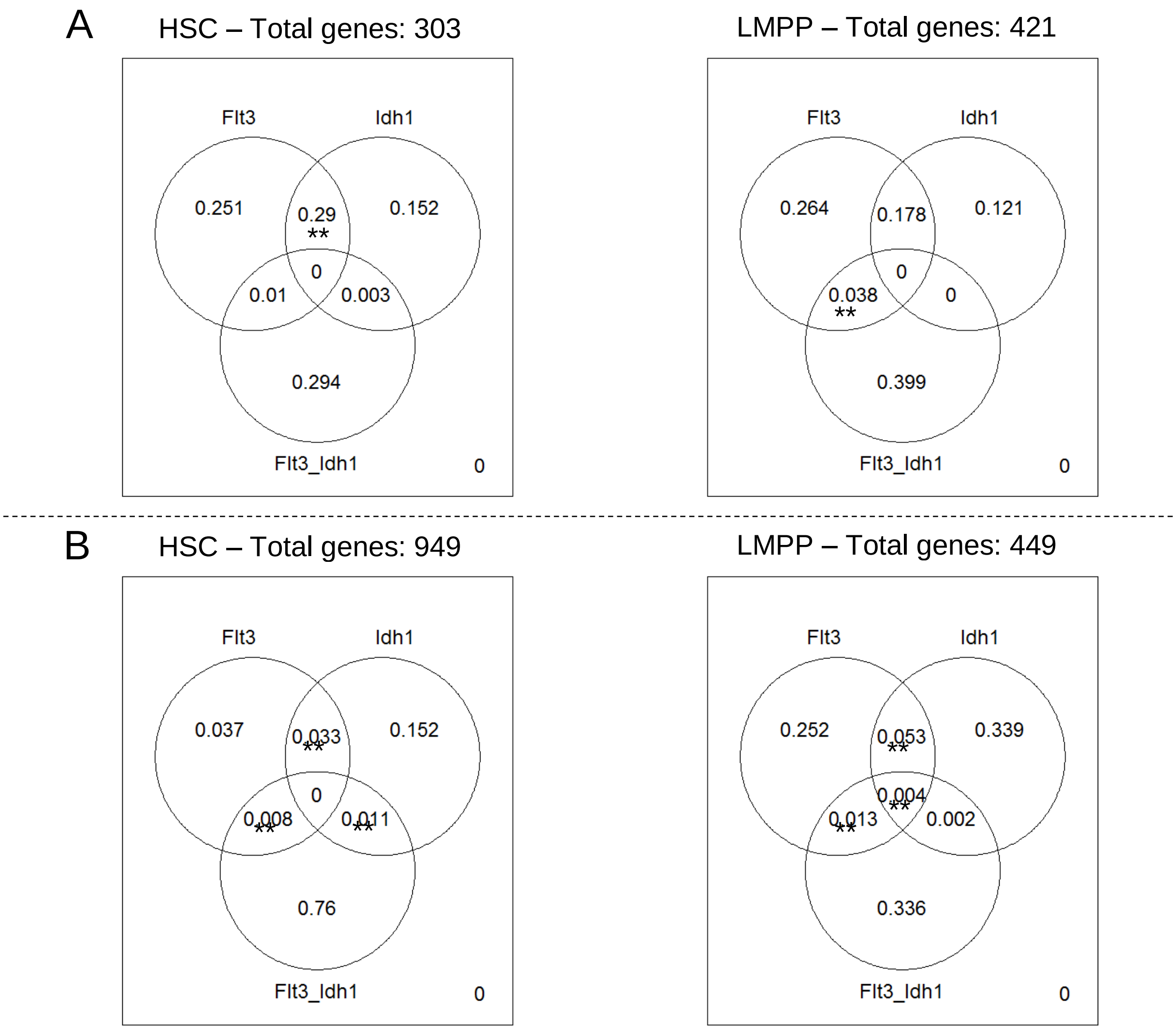


Figure 5.14 Venn diagrams of differentially expressed genes.
(A) Upregulated genes in HSCs and LMPPs, (B) Downregulated genes in HSCs and LMPPs.

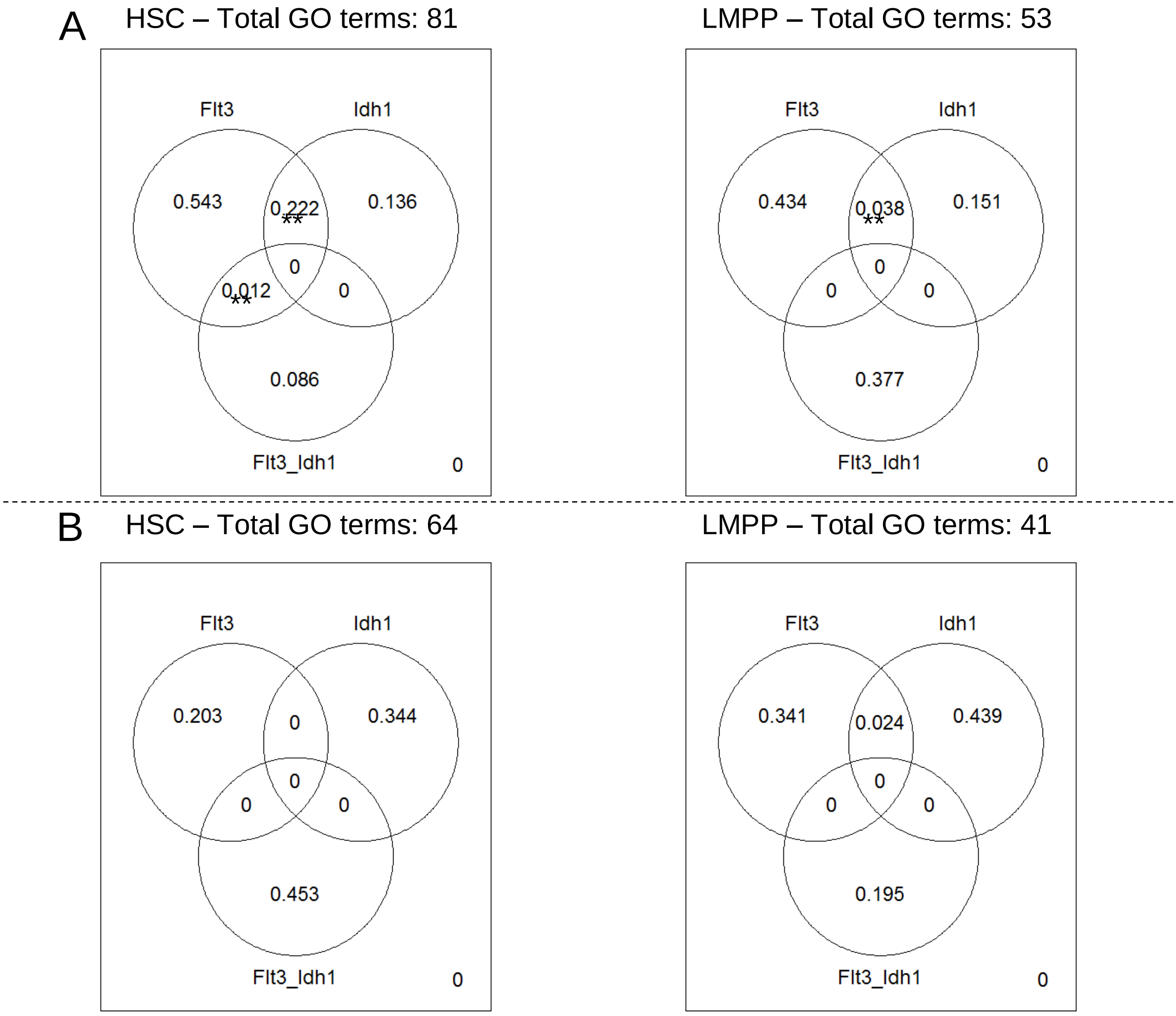


Figure 5.15 Venn diagrams of Gene Ontology analysis.
(A) Upregulated biological processes in HSCs and LMPPs, (B) Downregulated biological processes in HSCs and LMPPs.

Biological Process	Num. of Annotated Genes	Num. of Observed Genes	Adjusted P-values
FLT3 (Total: 138 upregulated genes)			
Monocyte chemotaxis	37	5	2.00E-05
Inflammatory response	517	21	2.00E-05
Leukocyte mediated cytotoxicity	84	6	5.60E-05
Negative regulation of growth	225	7	5.80E-05
Regulation of symbiosis, encompassing mutualism through parasitism	299	6	7.20E-05
Cellular extravasation	47	6	8.00E-05
Regulation of cell shape	126	7	0.00015
Chemokine-mediated signaling pathway	35	4	0.00027
Neutrophil chemotaxis	66	5	0.00033
Positive regulation of axon extension	42	4	0.00056
IDH1 (Total: 165 upregulated genes)			
Interleukin-8 secretion	20	3	0.00037
Negative regulation of growth	225	6	0.00062
Regulation of symbiosis, encompassing mutualism through parasitism	299	6	0.00071
Myeloid dendritic cell activation	29	3	0.00113
Erythrocyte development	30	3	0.00125
Apoptotic cell clearance	31	3	0.00138
Regulation of cell shape	126	5	0.0021
Phagocytosis, engulfment	36	3	0.00214
Regulation of alternative mRNA splicing, via spliceosome	36	3	0.00214
Cellular extravasation	47	3	0.00217
FLT3-IDH1 (Total: 101 upregulated genes)			
Immune response-regulating cell surface receptor signaling pathway	129	3	0.00085
Morphogenesis of a polarized epithelium	60	3	0.00181
Antigen processing and presentation of peptide or polysaccharide antigen via MHC class II	20	2	0.00438
Cellular response to extracellular stimulus	148	3	0.00479
Post-anal tail morphogenesis	21	2	0.00483
Cellular response to gamma radiation	21	2	0.00483
Negative regulation of cysteine-type endopeptidase activity involved in apoptotic process	73	3	0.00575
cGMP biosynthetic process	27	2	0.00792

Table 5.1 Top 10 upregulated biological processes in each genotype in HSCs based on Gene Ontology (GO) analysis.

Note that FLT3-IDH1 genotype has less than 10 significant results.

Biological Process	Num. of Annotated Genes	Num. of Observed Genes	Adjusted P-values
FLT3 (Total: 75 downregulated genes)			
Regulation of protein complex assembly	326	7	8.20E-06
Protein stabilization	117	5	9.50E-05
Regulation of synaptic vesicle exocytosis	23	3	9.60E-05
Cation transmembrane transport	432	4	0.0017
Negative regulation of cysteine-type endopeptidase activity involved in apoptotic process	73	3	0.0029
Glomerulus vasculature development	21	2	0.003
Response to vitamin	21	2	0.003
Post-anal tail morphogenesis	21	2	0.003
Regulation of lamellipodium assembly	27	2	0.005
Osteoclast differentiation	84	4	0.0051
IDH1 (Total: 186 downregulated genes)			
G2/M transition of mitotic cell cycle	84	6	0.00013
Regulation of cell cycle checkpoint	32	4	0.00027
Positive regulation of immune system process	670	18	0.00028
Regulation of antigen processing and presentation	21	3	0.00113
Cellular response to gamma radiation	21	3	0.00113
Synaptonemal complex organization	22	3	0.0013
Positive regulation of DNA recombination	23	3	0.00148
Positive regulation of amine transport	25	3	0.00189
Antigen processing and presentation of peptide antigen	59	4	0.00197
Mitotic spindle assembly checkpoint	26	3	0.00213
FLT3-IDH1 (Total: 28 downregulated genes)			
Regulation of alpha-beta T cell proliferation	26	7	6.70E-05
Response to oxidative stress	321	24	0.00023
Negative regulation of leukocyte apoptotic process	59	7	0.00065
Receptor metabolic process	146	10	0.00066
Establishment of spindle orientation	28	4	0.00066
Apoptotic cell clearance	31	6	0.00149
Alpha-beta T cell differentiation	91	8	0.00153
Positive regulation of alpha-beta T cell activation	55	6	0.00154
Vascular endothelial growth factor production	30	4	0.00171
Positive regulation of natural killer cell activation	22	5	0.00175

Table 5.2 Top 10 downregulated biological processes in each genotype in HSCs based on Gene Ontology (GO) analysis.

Biological Process	**Num. of Annotated Genes**	**Num. of Observed Genes**	**Adjusted P-values**
FLT3 (Total: 217 upregulated genes)			
Leukocyte mediated cytotoxicity	84	8	1.40E-06
Acute inflammatory response to antigenic stimulus	22	5	2.90E-06
Neutrophil mediated immunity	24	5	4.70E-06
Defense response to bacterium	163	10	6.50E-06
Negative regulation of growth	225	8	0.00011
Regulation of symbiosis, encompassing mutualism through parasitism	299	11	0.00012
Positive regulation of cell proliferation	737	17	0.00081
Phagocytosis	147	9	0.00094
Negative regulation of multi-organism process	133	7	0.00107
Granulocyte activation	23	3	0.0018
IDH1 (Total: 152 upregulated genes)			
Negative regulation of cell migration	198	8	5.80E-05
Cholesterol biosynthetic process	39	3	0.0023
Lysosome organization	53	3	0.0054
Negative regulation of neurogenesis	260	4	0.0064
Membrane lipid metabolic process	155	3	0.0067
Cellular protein complex disassembly	110	4	0.007
Negative regulation of oxidoreductase activity	20	2	0.0079
'De novo' posttranslational protein folding	20	2	0.0079
Positive regulation of angiogenesis	121	4	0.009
Monosaccharide catabolic process	22	2	0.0095
FLT3-IDH1 (Total: 190 upregulated genes)			
Positive regulation of cell motility	410	6	0.00017
Notch signaling pathway	147	6	0.00094
B cell activation involved in immune response	64	4	0.00106
Mature B cell differentiation	22	3	0.00107
Positive regulation of osteoclast differentiation	23	3	0.00122
Membrane assembly	24	3	0.00139
Negative regulation of endothelial cell proliferation	29	3	0.00242
Positive regulation of interleukin-12 production	30	3	0.00267
Genetic imprinting	35	3	0.00416
Regulation of alternative mRNA splicing, via spliceosome	36	3	0.00451

Table 5.3 Top 10 upregulated biological processes in each genotype in LMPPs based on Gene Ontology (GO) analysis.

Biological Process	Num. of Annotated Genes	Num. of Observed Genes	Adjusted P-values
FLT3 (Total: 156 downregulated genes)			
Antigen processing and presentation of peptide antigen via MHC class I	40	6	4.30E-07
Myeloid leukocyte cytokine production	21	3	0.00046
Negative regulation of protein import into nucleus	57	4	0.0006
Ovarian follicle development	60	4	0.00097
Positive regulation of interleukin-12 production	30	3	0.00134
Erythrocyte development	30	3	0.00134
Negative regulation of cell adhesion	207	6	0.00171
Cell surface receptor signaling pathway	1948	29	0.0029
Positive regulation of myeloid cell differentiation	81	6	0.00316
Gene expression	4296	40	0.00723
IDH1 (Total: 187 downregulated genes)			
Female meiotic division	34	4	0.00023
Positive regulation of osteoclast differentiation	23	3	0.00107
Regulation of endothelial cell differentiation	28	3	0.00192
Insulin secretion involved in cellular response to glucose stimulus	63	3	0.00211
Erythrocyte development	30	3	0.00235
Regulation of bone resorption	32	3	0.00283
Positive regulation of cysteine-type endopeptidase activity	102	3	0.00337
Negative regulation of epithelial cell differentiation	35	3	0.00367
Genetic imprinting	35	3	0.00367
T-helper 1 type immune response	36	3	0.00397
FLT3-IDH1 (Total: 165 downregulated genes)			
Regulation of myeloid cell apoptotic process	27	3	0.0015
Positive regulation of angiogenesis	121	5	0.0036
Blood coagulation	164	7	0.004
Cholesterol biosynthetic process	39	3	0.0043
Positive regulation of axon extension	42	3	0.0053
Chaperone-mediated protein folding	45	3	0.0064
Negative regulation of axonogenesis	56	4	0.0077
Negative regulation of leukocyte apoptotic process	59	3	0.0098

Table 5.4 Top 10 downregulated biological processes in each genotype in LMPPs based on Gene Ontology (GO) analysis.
Note that FLT3-IDH1 genotype has less than 10 significant results.

In order to investigate the cell type differences between FLT3-ITD and IDH1-R132H mutations, it was interesting to investigate the overlaps in enriched GO biological processes between the HSCs and the LMPPs (Figure 5.16). The overlaps in enriched GO biological processes were shown instead of the overlaps in differentially expressed genes, as enriched GO biological processes offer more robust results due to the use of sets of genes instead of considering single genes. The result in Figure 5.16 suggests in general FLT3-ITD and IDH1-R132H mutations exerted very different effects on expression profiles depending on the cell types. Note that both HSCs and LMPPs for the same genotypes were collected from the same mice, so the differences between HSCs and LMPPs are unlikely to be differences due to individual mice. However as each sample was sequenced separately, there is a chance that some of the variabilities may be due to random chance.

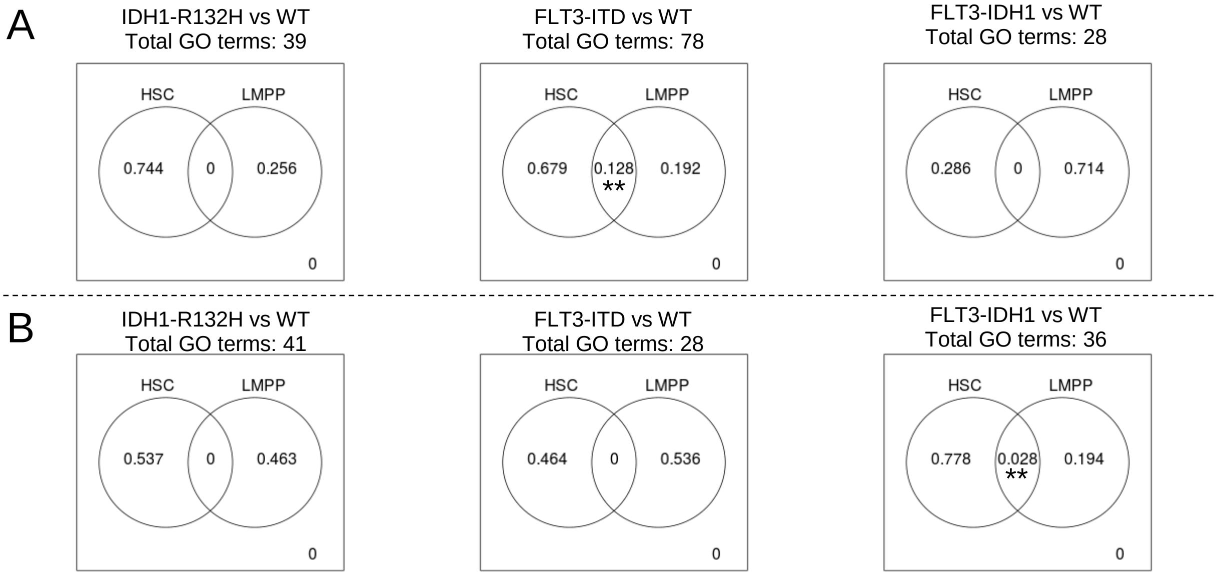


Figure 5.16 Venn diagrams of common differentially expressed genes between HSCs and LMPPs.
*(A) Upregulated genes, (B) Downregulated genes. *, p-value = 0.01; **, p-value = 0.001.*

5.3.3 Cell subpopulations are present in FLT3-ITD and IDH1-R132H HSCs and LMPPs

In order to locate any potential subpopulations of cells, different clustering and data preprocessing steps were tested. The tested data preprocessing steps include correcting for cell cycle phase difference and using only the highly variable genes that are above the technical noise threshold. The data preprocessed in different ways as described above were then used for clustering by using hierarchical and ICGS clustering. However, no meaningful clusters were obtained due to the presence of high biological and technical noise in the data. The biological noise is likely to be present due to the cells in this experiment being collected from mice, which represent an *in vivo* system that is usually noisier than an *in vitro* system. The technical noise present includes any unidentified batch effect and the cell cycle differences.

After evaluating many methods, the method that offered the best indication of cell subpopulations is by using differentially expressed genes for clustering cells via hierarchical clustering (Figure 5.17 & Table 5.5). There are eight samples in this dataset with different genotypes and cell types, therefore the number of clusters was set to eight by cutting the hierarchical cluster dendrogram into eight most distinct groups. The null hypothesis is that given there are originally eight samples present, the eight clusters determined should each have only cells that come from a single sample, with each cluster corresponding to a unique sample. If the result deviates from the expectation, this suggests that the differences within samples are stronger than the differences among samples, which indicate the presence of cell subpopulations.

Figure 5.17 shows that the total set of differentially expressed genes taken across all pairs of comparison is able to offer reasonable clusters of cells, where cells that come from the same sample were mostly clustered together. Similar results were not obtained by using other methods as described previously, where other methods cluster cells from the same sample randomly across multiple clusters. This is mostly due to the difficulty in selecting the set of most informative genes for clustering purpose. Note that although the cell cycle phase difference was accounted for during differential expression analysis, non-G0/G1 cell cycle phases can still be seen to be associated with several clusters (i.e. cluster 2, 3, 4 and 6). It is also worth noting that as discussed before, non-G0/G1 cell cycle phases were also more associated with LMPPs than HSCs.

In order to allow easier visualisation of cells being assigned to each cluster, a table containing the number and the type of cells assigned to each cluster was prepared (Table 5.5). In general, the HSCs and the LMPPs were clustered separately. In terms of HSCs, IDH1 and FLT3 HSCs were clustered together with wild type HSCs in cluster 1. This suggests that most IDH1 and FLT3 HSCs were very similar to wild type HSCs, with only a small proportion of cells that were assigned to other clusters. FLT3-IDH1 HSCs were very different from other HSCs, and were assigned to two clusters (i.e. cluster 4 and 7). As for LMPPs, the cells exhibited more variability than HSCs by occupying a higher number of clusters. Among the mutants, IDH1 LMPPs were the most strongly associated and therefore most similar with wild type LMPPs, which were split into two clusters (i.e. cluster 2 and 5). As for FLT3 and FLT3-IDH1 LMPPs, they were both distinct from one another, as well as being different from other LMPPs. Interestingly, FLT3-IDH1 LMPPs seem to be separated into two groups in cluster 3 and 8, which one of the groups shared similar properties to a subgroup of FLT3 LMPPs.

Taken together, the results offer several observations. Firstly, HSCs were more homogeneous than LMPPs as expected. Secondly, single mutation of either FLT3 or IDH1 in HSCs and LMPPs exerted smaller effects on expression profiles than FLT3-IDH1 double mutations. This holds true except for FLT3 LMPPs, which were very distinct from other LMPPs. The results suggest that FLT3-ITD and IDH1-R132H mutations may interact to produce a more perturbed expression profiles. Lastly, while cell subpopulations seem to be present among the cells, it is worth noting that these cell subpopulations may be detected due to the differences in cell cycle phase. Therefore, the cell subpopulations were not analysed further individually, as this RNAseq data has high biological and technical noise as discussed above that cannot be corrected for easily. In this chapter, interpretations were mostly drawn from all cells in a sample, which offer more robust conclusions due to the higher number of cell replicates, rather than interpreting cell subpopulations.

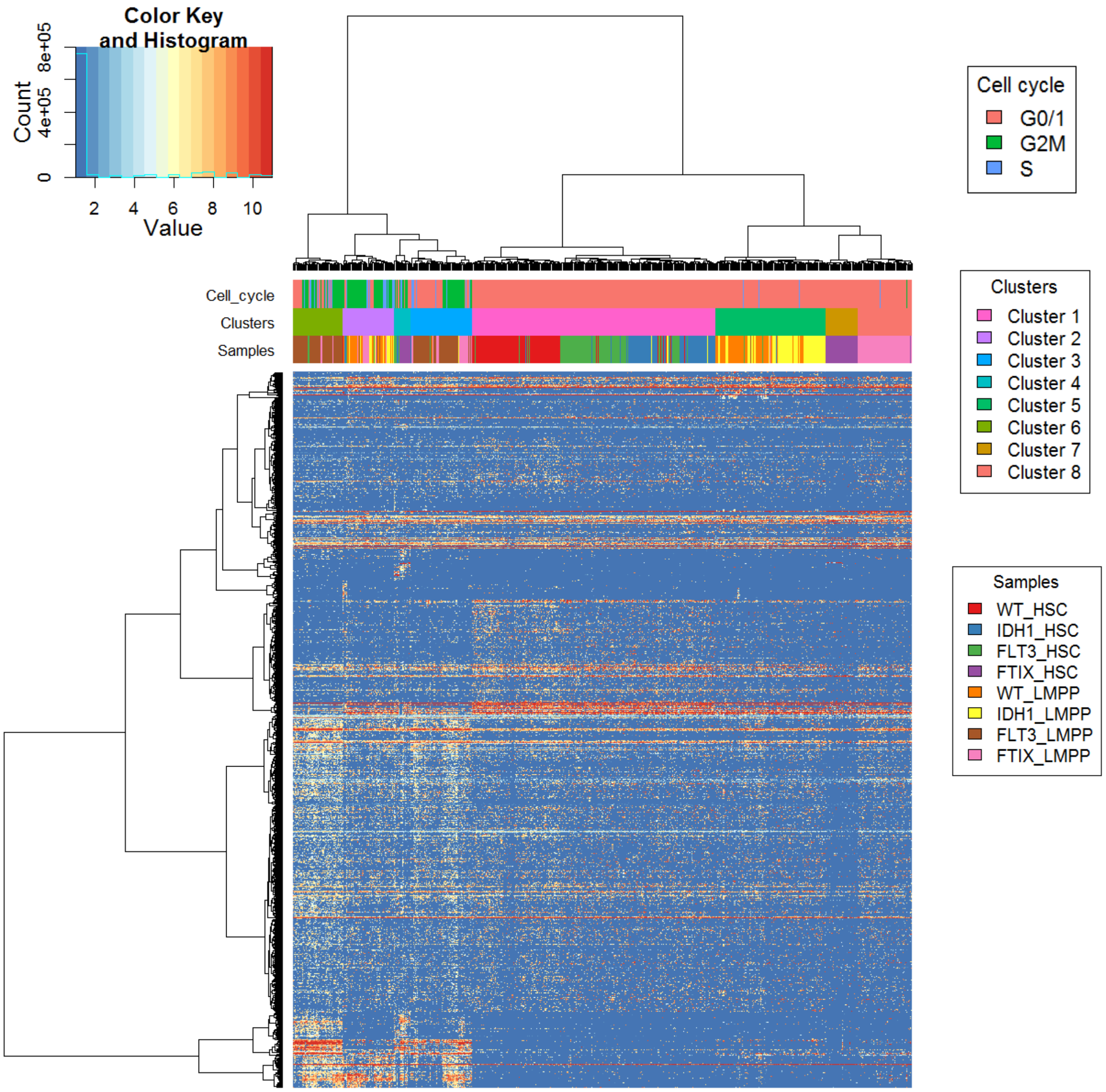


Figure 5.17 Heatmap of differentially expressed genes across all samples.
Note that the cell cycle phase difference was accounted for during differential expression analysis. FTIX refers to FLT3-IDH1.

Samples	**Clusters**							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
WT HSC	91	1	0	0	0	0	0	0
FLT3 HSC	78	2	1	4	2	2	0	0
IDH1 HSC	71	1	1	2	1	0	0	0
FLT3-IDH1 HSC	0	0	0	11	0	0	32	1
WT LMPP	0	23	1	0	46	0	0	0
FLT3 LMPP	0	3	41	0	1	45	0	0
IDH1 LMPP	3	12	1	0	59	0	0	0
FLT3-IDH1 LMPP	0	9	16	0	1	3	0	54

Table 5.5 Number of cells belonging to each cluster.

5.3.4 Increased myeloid potential in FTL3-ITD cells as identified via projections onto a diffusion map and clustering with a neural network

In a study by (Nestorowa et al. 2016), the authors profiled the single-cell expression profiles of more than 1600 haematopoietic stem and progenitor cells and constructed a map of blood cells development on a diffusion map. With the use of index sorting and broad sorting gates, they restrospectively identified and labelled a total of 12 types of commonly sorted blood cells on the diffusion map. In the 3-dimensional diffusion map, four extreme regions can be observed in the top, bottom, left and right corners (Figure 5.18). The top region corresponds to HSCs (green), while the three remaining regions, left, bottom and right corners correspond to lymphoid (orange), myeloid (blue) and erythroid (red) lineages. Only a subset of blood cell types relevant to this study were used for downstream analysis, which are illustrated by the diffusion maps below (Figure 5.19). In total six cell types were considered, including long term HSCs (LTHSCs), short term HSCs (STHSCs), lymphoid multipotent progenitors (LMPPs), megakaryocyte-erythrocyte progenitors (MEPs), common myeloid progenitors (CMPs), and granulocyte-monocyte progenitors (GMPs). For easier referencing, these diffusion maps will be referred to as the Nestorowa blood cells atlas.

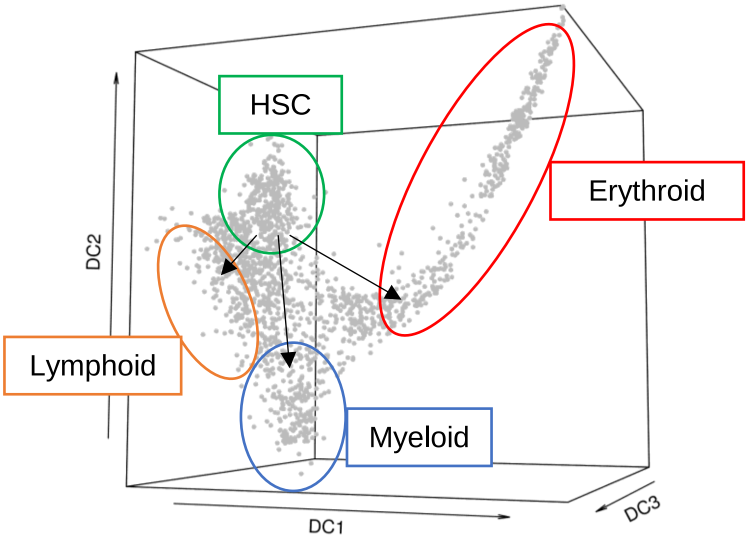


Figure 5.18 Nestorowa blood cells atlas.

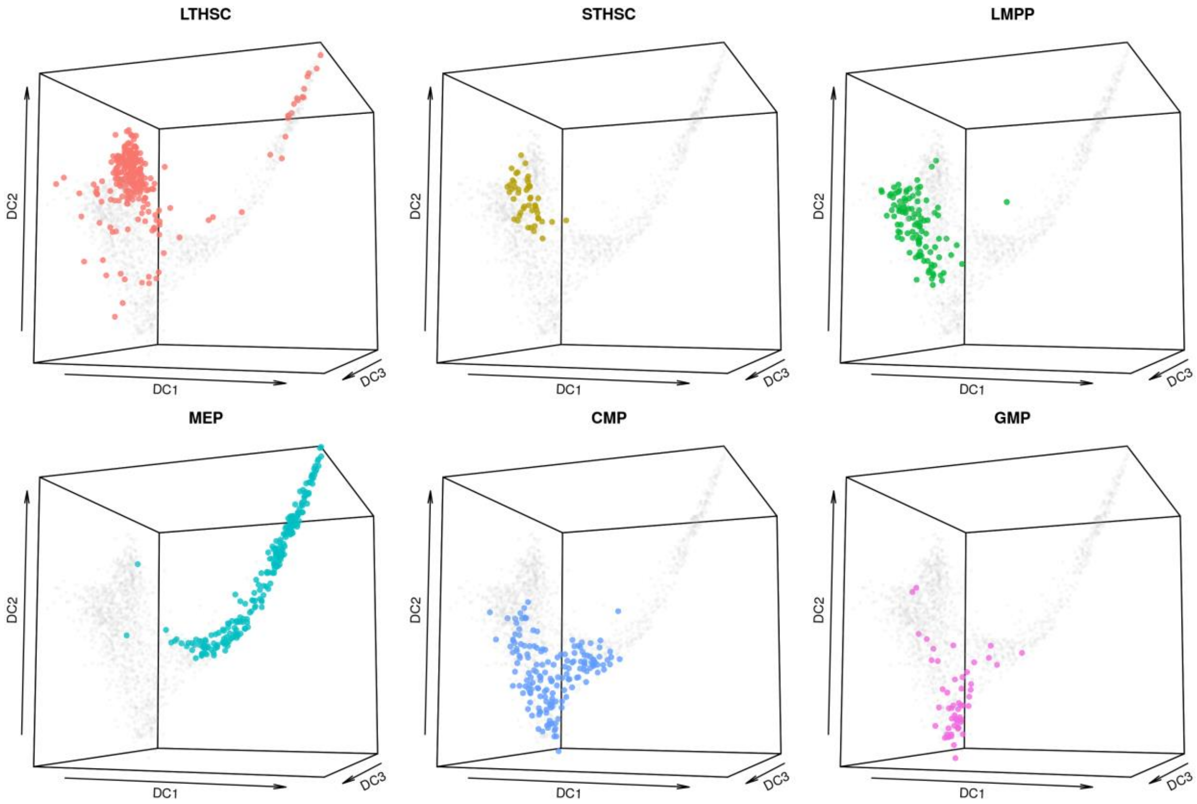


Figure 5.19   Different blood cells present in the Nestorowa blood cells atlas.

In Figure 5.20, the cells from this study were projected onto the Nestorowa blood cells atlas using the same parameters for diffusion maps as in the original paper (Nestorowa et al. 2016). Both wild type HSCs and LMPPs were projected onto the expected regions where the original wild type HSCs and LMPPs lied. This indicates that the projected coordinates of cells from this study may offer additional insights into the properties of FLT3 and IDH1 mutant cells. For any mutant cells, the presence of mutations tends to move cells out of the original regions occupied by wild type HSCs and LMPPs, and into the myeloid lineage region. The result suggests that both FLT3-ITD and IDH1-R132H mutations may be encouraging the mutant cells to acquire increased myeloid identity. The potential increase in myeloid identity was the most significant in FLT3 LMPPs, FLT3-IDH1 LMPPs and FLT3-IDH1 HSCs. As the 3-dimensional diffusion maps are difficult to visualised on a 2-dimensional surface, clustering algorithms were employed to estimate the cell identities of the projected cells. The clustering algorithms work by soft clustering the cells into different categories based on the relative distance between the coordinates of the projected cells and the coordinates of the underlying cells in the Nestorowa blood cells atlas.

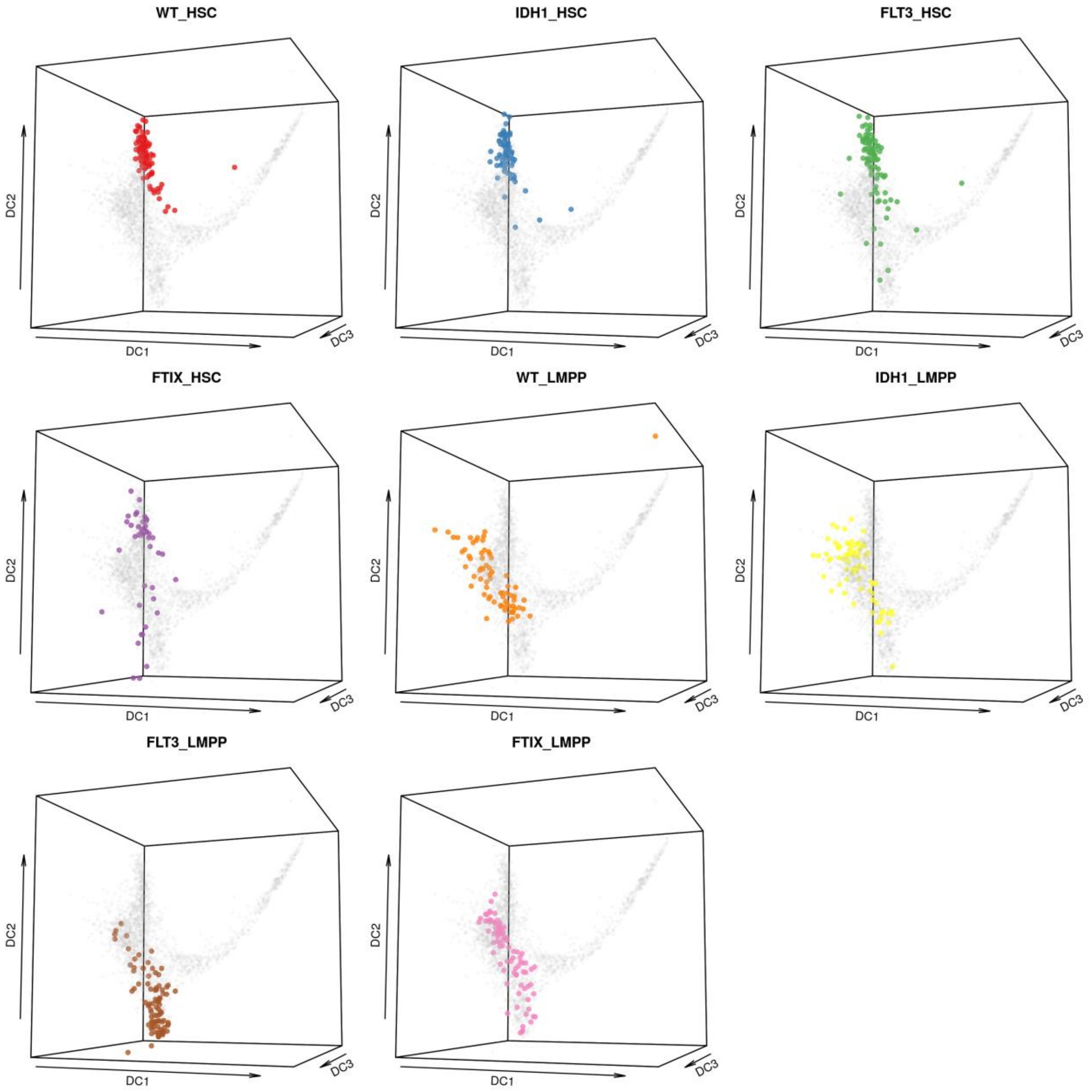


Figure 5.20 Wild type, Flt3 and Idh1 mutant cells projected onto the Nestorowa blood cells atlas.

FTIX refers to FLT3-IDH1.

Three different clustering algorithms were evaluated based on their accuracy of correctly classifying the six known cell types in the Nestorowa blood cells atlas. The three clustering algorithms are Naïve Bayes, random forest and neural network classifiers, which are all supervised learning methods based on different frameworks. Naïve Bayes classifier is based on Bayes' theorem from the Bayesian statistics framework and assumes independence among the variables (Hand & Yu 2001). Random forest classifier uses an ensemble of decision trees, which iteratively identify the most significant variable and its corresponding value that can give rise to the best homogeneous split of populations (Breiman 2001). Lastly, neural network is inspired by the architecture of biological neurons, which learns the unknown function that maps independent to dependent variables from the data (LeCun et al. 2015). It can also be viewed as a generalised form of the non-parametric regression model (Insua & Müller 1998).

Both Naïve Bayes and random forest classifiers were run with default settings, while neural network classifier requires the specifications of a few hyperparameters. The neural network classifier used here contains 3 input nodes, 10 hidden nodes, and 6 output nodes, which are arranged in 3 fully connected feedforward layers (Figure 5.21). The activation function is a softplus function, which acts an approximation to the rectified linear unit (ReLu) activation function. The softplus function was used here as it has a derivative that can be recognised by the neuralnet R package, while the non-standard derivative of ReLu function is not recognised. ReLu activation function offers several advantages over other activation functions, such as speeding up the training time and leading to better solutions (Nair & Hinton 2010). The error function used here is the standard sum of squared error, while the optimisation algorithm used here is a faster variant of the standard backpropagation algorithm, which is called the resilient backpropagation (Riedmiller & Braun 1993).

The results of the performance of the three clustering methods can be seen in Table 5.6. Naïve Bayes and random forest classifiers had similar accuracies with very fast computation time. Neural network requires substantial computation time in exchange for higher accuracies. Note that the computation time for neural network can be shortened with the use of more efficient optimisation methods and parallel processing. Based on the results of the classification, neural network classifier was used for classifying the cells projected onto the Nestorowa blood cells atlas diffusion map.

Methods	**Accuracy**		**Computation time taken (sec)**
	Training	**Validation**	
Naïve Bayes	0.72	0.71	0.005
Random forest	0.77	0.71	0.214
Neural network	0.83	0.78	80.998

Table 5.6 Performance of clustering methods.
Data available were split into training data (80% of data) and validation data (20% of data) randomly. All values are the average of 3 runs. All computation was performed on a single processing core.

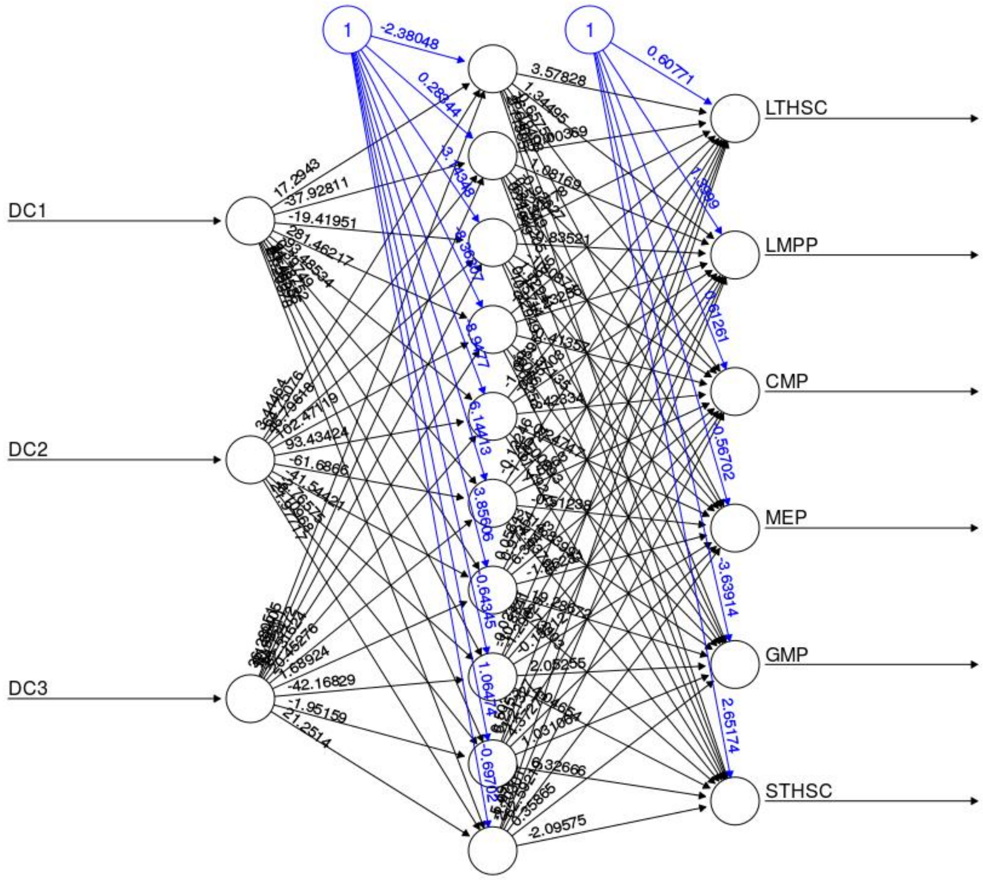


Figure 5.21 Neural network structure used.
There are three layers, corresponding to an input layer, a hidden layer and an output layer. Each circle corresponds to a node, with blue nodes representing the bias terms. Each line indicates a connection, with the associated numbers being the weights. DC1-3 refer to the first three diffusion components obtained from a diffusion map. The value of each output node corresponds to a similarity measure of the input cell with known cell types.

In Figure 5.22, each cell was assigned six probabilities of belonging to six of the cell types, which can also be interpreted as each cell having a cell identity that is defined by a set of six cell type identity measures. Consistent with results of previous analyses, wild type, IDH1 and FLT3 HSCs are very similar to one another, and possess strong LTHSC identity. However, FLT3-IDH1 HSCs possess weaker LTHSC identity and acquire myeloid identity, as indicated by CMP and GMP identities.

Due to the overlaps in diffusion map regions with LMPPs, CMPs and GMPs, it is more difficult to classify LMPPs cleanly into one category. Wild type and IDH1 LMPPs have similar cell identity profiles. As expected, FLT3 and FLT3-IDH1 LMPPs have very strong myeloid identities, although FLT3-IDH1 LMPPs have a weaker myeloid identity than FLT3 LMPPs. In summary, projections of cells onto the Nestorowa blood cells atlas followed by clustering are able to offer an independent method to visualise and categorise the data. The results point to increased myeloid potential caused by the acquisition of FLT3-ITD and IDH1-R132H mutations. However, it should be noted that the lymphoid lineage has a low resolution, as it is only represented by the LMPPs which are less differentiated, hence may not resolve the lymphoid potential of FLT3 and IDH1 mutant cells well. It is possible that FLT3-ITD mutation may also perturb lymphoid development, as FLT3-ITD mutation have been shown to be important in lymphoid leukaemia (Wellmann et al. 2005) and FLT3 is known to be important for early B cell development (Mackarehtschian et al. 1995).

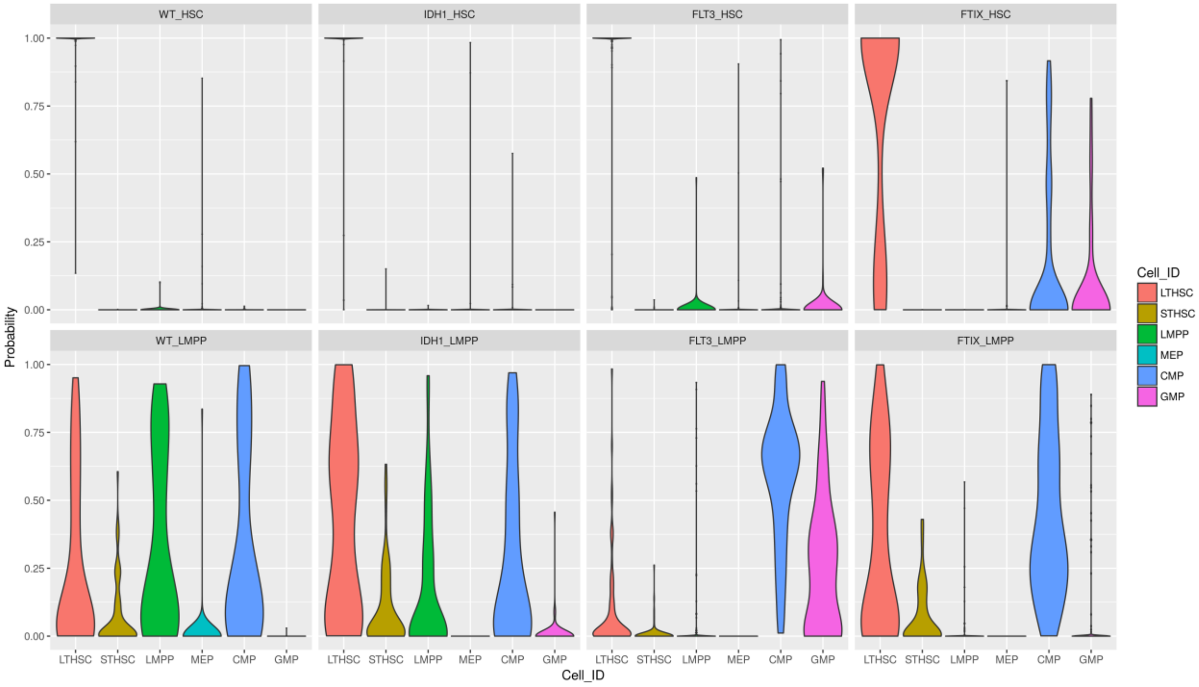


Figure 5.22 Inferred cell type identities on wild type blood cells.
Note that each cell is not hard clustered into a specific cell type, and each cell contains six probabilities of belonging to the six group of cells. This also means that each individual violin plot has the same number of cells.

5.4 Conclusions

In this chapter, we have explored the effects of FLT3-ITD and IDH1-R132H mutations on the expression profiles of HSCs and LMPPs. While FLT3-ITD and IDH1-R132H mutations share a small subset of perturbed biological processes in increasing the myeloid potential of the mutant cells, most perturbed biological processes are distinct both between the mutations FLT3-ITD and IDH1-R132H, as well as between the cell types HSC and LMPP. In addition, FLT3-ITD and IDH1-R132H have been shown to potentially have a synergistic effect in predisposing cells towards the myeloid lineage. This observation suggests that while FLT3-ITD and IDH1-R132H co-occur at a low frequency in AML patients, the two mutations are not mutually exclusive.

5.5 Materials and methods

5.5.1 Processing of single-cell RNAseq data

The single-cell RNAseq data consist of 768 cells in eight sample types, which include HSCs and LMPPs with four genotypes, namely wild type, IDH1, FLT3 and FLT3-IDH1. The RNAseq data were generated using SmartSeq2 library preparation protocol, and sequenced on the Illumina HiSeq 4000 machine. The data has paired ends, with 100 base pair read length.

All RNAseq data were aligned using GSNAP version 2015-09-29 (Wu & Nacu 2010). Aligned reads were counted using HTSeq-count (Anders et al. 2015). Ensembl genome index and gene annotations release version 77 were used.

5.5.2 Single-cell RNAseq analyses

ICGS clustering, heatmaps, cell cycle analysis, differential expression analysis, Gene Ontology analysis and significance of Venn diagram overlaps were performed as discussed in Section 4.6. The projection of cells onto the diffusion map was done using the destiny R package (Haghverdi et al. 2015). Differential gene expression analysis was performed with the DESeq2 R package (Love et al. 2014), with a design matrix that consider the effects exerted individually by FLT3-ITD and IDH1-R132H mutations, as well as the effect exerted by interaction between FLT3-ITD and IDH1-R132H mutations. Adjusted p-values corrected for multiple testing were computed using the Benjamini & Hochberg method.

5.5.3 Supervised clustering of cell projections onto diffusion maps

The three clustering methods used for supervised clustering of cell projections onto diffusion maps are Naïve Bayes, random forest and neural network classifiers. They are all implemented as R packages in e1071 (Meyer et al. 2017), randomForest (Liaw & Wiener 2002) and neuralnet (Fritsch & Guenther 2016) respectively.

6 Discussion

The direct interpretations of the results have been elaborated alongside the results within Chapter 2-5. Therefore, the focus in this chapter is on discussing the scientific contributions, the limitations, and the future directions related to the results.

6.1 Network inference with the Boolean model framework – BTR

A Boolean formalism-based network inference algorithm, BTR, has been described in Chapter 2. BTR uses a scoring function to evaluate how well the predictions made by a Boolean model match with single-cell expression data supplied to the algorithm. By using this score, BTR iteratively modifies the Boolean model so as to yield a final Boolean model whose simulated predictions are very close to the observed single-cell expression data. The performance of BTR was evaluated with synthetic expression data, and was shown to be performing well when BTR was supplied with a partial Boolean model that encodes some initial information.

Two recent studies reported algorithms for inferring Boolean models from single-cell expression data (Chen et al. 2014; Moignard et al. 2015). Chen *et. al.* developed SingCellNet, which uses a genetic algorithm to construct probabilistic Boolean models from expected trajectories through cell states (Chen et al. 2014). However, SingCellNet only determines the network structure and transition probabilities from single-cell expression data, while the Boolean rules are constructed via manual curation from the literature. In contrast, BTR automates the process of learning Boolean rules from the expression data. In another study, SCNS was developed by Moignard *et. al.* to infer an asynchronous Boolean model by analysing trajectories through a state transition graph (Moignard et al. 2015). In order to infer a Boolean model using SCNS, a connected state transition graph is required, which can be difficult to obtain from single-cell expression data. This is because the higher the number of genes to be included in SCNS, the more cells will be required to build a connected state transition graph. In addition, SCNS can only infer network structure by using discretised expression data, which not only leads to the loss of information, but also makes SCNS sensitive to the discretisation method used. In contrast, BTR does not assume a connected state transition graph is captured in the expression data and BTR is able to use continuous expression data without the need of discretising the values. In summary, BTR complements these existing algorithms by offering an algorithm that is capable of improving existing Boolean models by using information obtained from new single-cell expression data.

However, BTR suffers from several limitations. Firstly, BTR does not guarantee the identification of global optima within the score landscape as specified by the scoring function. It is likely that BTR may be trapped in local optima due to the greedy nature of the optimisation algorithm. However, this will not be a problem if BTR is supplied with an initial Boolean model which has some edges that are known to be true based on external information, such as information curated from the literature. This is because the initial Boolean model with some true edges should be relatively close to the global optima, therefore reducing the chance that BTR will get stuck in local optima. Secondly, the properties of the scoring function of BTR require further investigations and improvements. The scoring function of BTR typically gives rise to a score landscape that consists of multiple extended flat surfaces with intermittent steps. This is due to both the discrete nature of Boolean models from the use of only binary values, as well as the unknown relationship between the Boolean model and its associated asynchronously simulated state space. The use of only binary values restricts the range of scores that can be outputted by the scoring function, unlike continuous values that are more likely to give rise to non-flat score landscape. The use of only binary values is a trade-off that comes with using Boolean models which offer simpler specification, and it is something that cannot be improved on without generalising the Boolean models. However, the unknown relationship between the Boolean model and its associated asynchronously simulated state space can potentially be elucidated with further studies.

It is widely known that a Boolean model can be reconstructed exactly via logical inference with a synchronously simulated state space (e.g. by using a Karnaugh map (Karnaugh 1953)). This is not straightforward for an asynchronously simulated state space as not all Boolean variables are updated at each time step. It is likely that by studying the properties of an asynchronously simulated state space, it is possible to observe helpful relationships that can be encoded into the scoring function, which can ultimately lead to faster and more accurate results. One such relationship was described in Chapter 2 and encoded into the scoring function in the form of the penalty term ε_1, which penalises the proportions of 0s and 1s. It was observed that the more densely connected the nodes in a Boolean model are, the more similar the proportions of 0s and 1s in the asynchronously simulated state space. This relationship helps the derivation of the penalty term ε_1 in penalising Boolean models that are too densely connected.

Lastly in terms of practicalities, BTR suffers from a very slow computation speed. The slow computation speed is mostly due to the need of calculating all pairwise distances between each row of the simulated Boolean state space and each row of the observed single-cell expression state, and partly due to the inefficient greedy branching optimisation algorithm. The computation speed can potentially be sped up greatly by using a simpler distance function to approximate the distance between the entire Boolean state space and the entire observed single-cell expression state space. In addition, a more efficient optimisation algorithm based on techniques such as simulated annealing or genetic algorithm is likely to offer better computation speed and results. However, these optimisation algorithms require additional tuning parameters, therefore will require a better understanding of the score landscape.

6.2 Network inference with the pseudotime-ordered autoregression framework – SPVAR with DM-DPT

The two key components of the network inference framework based on pseudotime-ordered autoregression are the DM-DPT algorithm which is used for pseudotime inference, and the SPVAR algorithm which is used for network inference. DM-DPT is an extension of DPT pseudotime algorithm by combining the cell ordering inferred through orthogonal projection onto a polynomial curve in diffusion map (DM) space, with the pseudotime distance inferred by DPT. SPVAR is an autoregression-based algorithm that incorporates Elastic Net penalisation and uses stability selection for obtaining robust results. Both DM-DPT and SPVAR were verified by synthetic expression data and benchmarked with other algorithms. DM-DPT was shown to be the best performing pseudotime inference algorithm, while SPVAR was shown to be one of the most conservative network inference algorithm with reasonable performance.

Previous studies have developed autoregression with regularisation for gene network inference mostly using simulated or microarray data (Fujita et al. 2007; Shimamura et al. 2009; Haury et al. 2012). The method by Fujita et al. uses Lasso (i.e. L1-norm) for regularisation, while Shimamura et al. uses Elastic net (i.e. L1L2-norm) for regularisation. In (Haury et al. 2012), they coupled Lasso regularisation with stability selection into an algorithm called TIGRESS to further improve upon the network inference results. SPVAR is similar to TIGRESS as both are based on penalised regression and utilise stability selection. The main differences between the two algorithms are the optimisation methods used and the implementation of stability selection. TIGRESS uses the least angle regression technique (LARS) with L1-norm regularisation (i.e. Lasso) for fitting the regression model, while SPVAR uses the cyclical coordinate descent technique with L1L2-norm regularisation (i.e. Elastic net). In terms of stability selection, at each subsampling iteration, TIGRESS splits all samples into two sets randomly, in which each set is fitted separately to obtain the first five variables that were selected by LARS. The resulting weighted adjacency matrix from TIGRESS is then obtained by taking the proportions of variables that were selected by LARS out of all subsampling iterations. For each subsampling iteration in SPVAR, 90% of the samples are used for fitting regression, and all non-zero variables were recorded. The resulting proportions of non-zero variables in all subsampling iterations were then used to obtain a set of selected variables by keeping all variables above a threshold of more than 0.6. These sets of selected variables were then refitted with cross validation to obtain a set of coefficients that gives rise to the weighted adjacency matrix from SPVAR. The use of L1L2 norm regularisation in SPVAR offers advantage over L1 norm regularisation in TIGRESS, as the introduction of L2 norm regularisation can reduce the likelihood of selecting a set of correlated variables. In addition, the implementation of stability selection in SPVAR offers more robust results as it is not bound by a maximum number of selected variables per iteration, unlike TIGRESS that selects a maximum of 10 variables per iteration. In summary, SPVAR is an alternative implementation of stably selected regularised regression algorithm that does not require manual specification of thresholds and is able to offer conservative predictions of gene interactions.

In terms of DM-DPT, its performance relies on a few important assumptions. Firstly, DM-DPT requires the manual selection of diffusion components of the diffusion map that best capture both the time progression as well as the changes in gene expression due to time progression. DM-DPT was implemented for use with two diffusion components by fitting a curve, but it can be extended to more than two diffusion components by fitting a more complex surface. Secondly, DM-DPT assumes there is no branching point in the single-cell expression data, which may not be the case. This can be partially overcome by identifying the branches using independent methods, and run DM-DPT separately on each of the branches. The results can then be combined later to yield a branching trajectory. For future improvements, one of the key improvements is to replace the parametric polynomial curve fitting with a non-parametric approach such as spline. This is because currently the polynomial curve was fitted by minimising the residuals in only one of the two diffusion components, while the orthogonal projection of cells onto a single trajectory uses both diffusion components. A non-parametric approach that minimises the residuals in both diffusion components when fitting the polynomial curve should give a better fit and result for the orthogonal projection of cells.

As for SPVAR, it relies on a few key assumptions that may limit its performance. Firstly, SPVAR assumes linear relationships for gene interactions. While this offers simpler computation, gene interactions are most likely to possess non-linear property in real biological systems. This problem can be overcome by extending SPVAR from a linear model into a non-linear model such as generalised additive models with spline functions. Secondly, SPVAR assumes all gene interactions to possess the same rate of interactions that can be captured by a time lag of 1 unit. This assumes that when the expression of a gene changes, the expression of its downstream target gene changes in response at the next time step. However in real biological systems, it is likely that different transcription factors activate downstream genes with different rates, due to various reasons such as the need to form a protein complex or the need to be post-translationally modified. This issue can be easily overcome by also examining time lags of more than 1 unit, however this will lead to increased computational complexity.

6.3 Performance of network inference algorithms

In general, network inference algorithms do not perform at a level that is significantly different from the results obtained by generating random values (Figure 3.10). This observation is also supported by two independent studies (Marbach et al. 2012; Qi & Michoel 2012), which also uses a similar set of algorithms and synthetic data. Both papers show that when the underlying network is complex and sparse (as in the gene network of the yeast *Saccharomyces cerevisiae*), most network inference algorithms perform poorly and at a level that is similar to randomly generated results.

There are a few explanations for the apparent poor performance of network inference algorithms. Firstly, it is likely that the expression data used for network inference contains an imbalanced number of informative and non-informative data points, where there are significantly more non-informative data points. These non-informative data points may be a result of both biological and technical noises in real expression data. Data points are only informative if they contain fluctuations in values due to perturbations exerted by upstream interacting genes in previous time points. Inference of gene interaction is difficult if the occurrence of perturbations is random and cannot be controlled directly or identified easily, which is the kind of perturbations typically observed in single-cell RNAseq data.

Secondly, inferring gene networks from a large number of genes suffers from a combinatorial problem, in which the expression value of a downstream target gene can be explained by multiple combinations of upstream genes which are equally plausible according to their respective expression values. This problem can be overcome by using independent data sources, such as transcription factor binding sites, to constraint the total number of plausible gene combinations. Lastly, interconnections of gene interactions, which include feedback loops, result in complex patterns in the expression values. The presence of loops makes the identification of direct gene interactions difficult, but this problem is unlikely to be solvable without additional information on the gene connectivity and the rate of gene activity propagation along the network.

In order to bypass the problems described above and achieve an accurate dissection of the underlying gene regulatory network, an ideal experimental system is required. This system should allow the control of individual expression perturbations and live measurements of potential downstream genes. This can potentially be achieved by imaging fluorescence-tagged mRNA molecules in live cells (Lubeck et al. 2014) incubated in a microfluidic device with chemical perturbations (Roman et al. 2006). Besides the idealised experimental system which is costly and laborious, another way to improve the network inference performance is to incorporate multiple sources of independent but complementary information. For example, an accurate gene regulatory network is more likely to be inferred by combining multiple types of information, such as from transcriptomics, proteomics, transcription factor binding sites and experimental results from the literature (Wang et al. 2015; Zarayeneh et al. 2016).

6.4 Benchmark of differential expression analysis algorithms

Differential expression analysis represents one of the key analyses that is performed across almost all experiments with high-throughput expression data from microarray to RNAseq. This is because differential expression analysis is able to identify uniquely expressed genes that are specific to certain samples in the high-throughput expression data. In Section 4.3.4, differential expression analysis algorithms that are developed for bulk and single-cell expression data were benchmarked by using synthetic expression data. The results suggest that the best performing differential expression analysis algorithms for single-cell RNAseq are SCDE, DESeq2 and edgeR.

It should be noted that while the conventional differential expression analysis algorithms are very good in detecting differences in the measure of central tendency (i.e. mean and median), they are not designed for detecting differences in the measure of variability (i.e. variance). An example of an algorithm that is capable of detecting variability among single cells is the BASiCS algorithm (Vallejos et al. 2015), which breaks down variability into technical and biological components by using the spike-in reads. Comparing differences in the variability across samples have been made possible by single-cell expression data, and represent an important feature to explore in the different samples. However, experimentally recreating such variability in the expression of a gene under a controlled system is much more difficult than recreating differences in the central tendency of the expression values.

Lastly, it should be noted that as single-cell expression data offer an increasingly higher number of samples, traditional standard statistical tools, such as Kolmogorov-Smirnov (KS) and Wilcoxon rank sum (WC) tests, become increasingly powerful for differential expression analysis. Algorithms such as DESeq and edgeR were developed for bulk expression data with very few number of samples. As can be seen in Figure 4.14, while DESeq and edgeR perform better than KS and WC tests for detecting sample differences in central tendency, but KS and WC tests were able to detect sample differences in both central tendency and variability. In addition, KS and WC tests are non-parametric, unlike DESeq and edgeR, therefore are especially suitable for expression data with uncharacterised properties, such as expression data generated by a new protocol.

6.5 Epiblast stem cells reprogramming system

Chapter 4 describes the insights obtained from studying single-cell RNAseq data collected from three epiblast stem cell (EpiSC) lines that were each reprogrammed by a separate transgene (i.e. *Esrrb*, *Klf2*, *GY118F*) into induced pluripotent stem cells (iPSCs). The results suggest that while all transgene-driven cell lines are able to reprogramme successfully, each cell line underwent a different reprogramming route by activating and inhibiting different biological processes. The EpiSC reprogramming system was studied using multiple data sources including both bulk and single-cell RNAseq data, as well as multiple analysis techniques including clustering, pseudotime inference and network inference. When taken together, the results support the hypothesis by showing reprogramming differences among the cell lines. The upregulation of *Esrrb* drives reprogramming by modulating transcriptional regulation important for establishing naïve identity, the upregulation of *Klf2* drives reprogramming by regulating cell proliferation and differentiation, while the upregulation of *pStat3* in *GY118F* cell line drives reprogramming by regaining trophectoderm potential and downregulating BMP/SMAD pathway. These new biological insights help understand the key pathways that facilitate the reprogramming transition from the EpiSC primed pluripotency state into the ESC naïve pluripotency state.

This study can be improved further in various aspects. Firstly, the expression of the transgene may change drastically after the induction, but the earliest time point used in the analysis is 1 hour after the induction of transgene. A higher time resolution during this initial stage of transgene induction will be beneficial for the identification of genes that are directly downstream of the transgene. In addition, the 1 hour time point is only available for bulk RNAseq but not single-cell RNAseq, where the earliest time point is 2 days after induction. Secondly, the induction of transgene may result in an expression level that cannot be observed under normal physiological condition. This may lead to unintended side effects and make the inference of gene interactions difficult, as a highly expressed gene may interact with more partners than under a normal expression level observed under physiological condition. Lastly, while most analyses on this dataset are likely to offer reliable results, care should be taken when interpreting the results of inferred gene networks in Section **Error! Reference source not found.**. This is due to the reasons as discussed above, relating to the performance of both SPVAR and network inference algorithms in general. The inferred networks can be made more robust by integrating additional information obtained from other data sources such as verified transcription factor binding sites.

6.6 Acute myeloid leukaemia pre-leukaemic system

The last result chapter, Chapter 5, describes the insights obtained from studying single-cell RNAseq data collected from cells with FLT3-ITD and IDH1-R132H mutations. The aim of this study is to investigate the effects of these two mutations on establishing pre-leukaemia identities and the possibility of synergistic interactions between the two mutations that co-occur rarely in acute myeloid leukaemia (AML) patients. The results suggest that while both mutations exert changes on expression profiles, FLT3-ITD has a stronger effect on perturbing expression profiles and increasing myeloid potential than IDH1-R132H. Previous clinical genomic studies have shown that mutations in genes that destabilise the genome, such as *Idh1*, typically arise early in AML before leukaemogenic events, while mutations in genes that contribute to leukaemogenic events directly, such as *Flt3*, arise later (Shlush et al. 2014; Corces-Zimmerman et al. 2014). However, no previous study has investigated the effects of the co-occurrence of both *Idh1* and *Flt3* mutations in blood cells. In this study, we have shown that FLT3-ITD and IDH1-R132H interact synergistically which lead to perturbed immune cell differentiation.

This study can be improved further in various aspects. Firstly, the observed synergistic interactions between the two mutations may be due to a dosage compensation effect, as both mutations were heterozygous in this study and are shown to co-occur rarely in AML patients. It may also be due to differences in individual mice, as only one mice for each mutation is explored here. This experiment should ideally be repeated with more mice to see if the same observations occurred in multiple mice. Secondly, the choice of LMPPs for studying AML may not be ideal, as LMPPs are primed for the lymphoid lineage while AML is a defect in the myeloid lineage. In addition, both FLT3-ITD and IDH1-R132H may also play a role in lymphoblastic leukaemia (Wellmann et al. 2005; Zhang et al. 2012). However, both mutations are still more common in AML than in lymphoblastic leukaemic patients. Interestingly, perturbed lymphoid cells development was indeed observed in the FLT3-IDH1 double mutant cells (Table 5.2 & Table 5.3). This problem can be solved in future experiments by including a less lymphoid primed haematopoietic progenitor such as multi-potent progenitors (MPPs), or by using a myeloid lineage progenitor such as common myeloid progenitors (CMPs).

Lastly, the use of Nestorowa blood cells atlas (Figure 5.18), which is in the form of a diffusion map, for cell type identification by projecting independent expression data onto the atlas may be subjected to a certain degree of bias. The bias comes mainly from the constrained space of diffusion map for cell projections. The diffusion map space present in the atlas that is available for cell projection is actually more limited than visually observed. This is due to the folding manifold as introduced by the kernel used in diffusion map, which means that it is very unlikely to have any projected cell that falls outside of the space originally occupied by the cells in the atlas. This issue can be partially overcome by only projecting blood cells that are known to share similar expression profiles to the cells used in the atlas, and to increase the types of cells available in the atlas.

6.7 Concluding remarks

The overall contributions of this thesis can be separated into two domains, namely the technical and the biological domains. In terms of technical domain, this thesis has outlined both the development of new algorithms, i.e. new network inference algorithms BTR and SPVAR; as well as the verification and the benchmarking of existing algorithms, i.e. pseudotime inference, network inference and differential expression algorithms. These findings will hopefully help other researchers in selecting the most suitable algorithms for their specific use cases, and also encourage other researchers to actively benchmark their own algorithms using synthetic data with known properties before using them on real biological data.

As for the biological domain, this thesis has outlined new biological insights obtained from two analyses performed on single-cell RNAseq collected from EpiSC reprogramming system and AML pre-leukaemic system respectively. It is hoped that these studies produce findings that help researchers in identifying interesting biological insights that can then be verified experimentally. Despite the increased technical noise present in single-cell RNAseq compared to bulk RNAseq, single-cell RNAseq has been shown to offer a much higher data resolution that is very helpful in studying biological processes with a system-wide approach. Provided that the increased technical noise is carefully accounted for, single-cell RNAseq is expected to continue dominate the field of transcriptomics study, especially with the advent of new sequencing library construction protocols such as DropSeq (Macosko et al. 2015).

7 References

Acampora, D., Di Giovannantonio, L.G. & Simeone, A., 2013. Otx2 is an intrinsic determinant of the embryonic stem cell state and is required for transition to a stable epiblast stem cell condition. *Development*, 140(1), pp.43–55.

Albert, R., Jeong, H. & Barabasi, A.-L., 2000. Error and attack tolerance of complex networks. *Nature*, 406(6794), pp.378–382.

Alexa, A. & Rahnenfuhrer, J., 2016. topGO.

Aloy, P. & Russell, R.B., 2006. Structural systems biology: modelling protein interactions. *Nature Reviews Molecular Cell Biology*, 7(3), pp.188–197.

Anders, S. & Huber, W., 2010. Differential expression analysis for sequence count data. *Genome Biology*, 11(10), p.R106.

Anders, S., Pyl, P.T. & Huber, W., 2015. HTSeq--a Python framework to work with high-throughput sequencing data. *Bioinformatics*, 31(2), pp.166–169.

Andersson, A.K. et al., 2011. IDH1 and IDH2 mutations in pediatric acute leukemia. *Leukemia*, 25(10), pp.1570–7.

Andrews, T., 2016. M3Drop: Michaelis-Menten Modelling of Dropouts in single-cell RNASeq.

Andrews S, 2010. FastQC: a quality control tool for high throughput sequence data.

Armbruster, D. et al., 2009. Dynamic Simulations of Single-Molecule Enzyme Networks. *The Journal of Physical Chemistry B*, 113(16), pp.5537–5544.

Arnold, S.J. et al., 2006. Dose-dependent Smad1, Smad5 and Smad8 signaling in the early mouse embryo. *Developmental Biology*, 296(1), pp.104–118.

Austin, D.W. et al., 2006. Gene network shaping of inherent noise spectra. *Nature*, 439(7076), pp.608–611.

Balasubramanian, M. et al., 2002. The Isomap Algorithm and Topological Stability. *Science*, 295(5552).

Barabasi, A.-L., Gulbahce, N. & Loscalzo, J., 2011. Network medicine: a network-based approach to human disease. *Nat Rev Genet*, 12(1), pp.56–68.

Barabasi, A.-L. & Oltvai, Z.N., 2004. Network biology: understanding the cell's functional organization. *Nat Rev Genet*, 5(2), pp.101–113.

Basso, K. et al., 2005. Reverse engineering of regulatory networks in human B cells. *Nature Genetics*, 37(4), pp.382–390.

Becskei, A., Kaufmann, B.B. & van Oudenaarden, A., 2005. Contributions of low molecule number and chromosomal positioning to stochastic gene expression. *Nature Genetics*, 37(9), pp.937–944.

Bendall, S.C. et al., 2014. Single-Cell Trajectory Detection Uncovers Progression and Regulatory Coordination in Human B Cell Development. *Cell*, 157(3), pp.714–725.

Bengtsson, M. et al., 2005. Gene expression profiling in single cells from the pancreatic islets of Langerhans reveals lognormal distribution of mRNA levels. *Genome Research*, 15(10), pp.1388–1392.

van den Berg, D.L.C. et al., 2010. An Oct4-Centered Protein Interaction Network in Embryonic Stem Cells. *Cell Stem Cell*, 6(4), pp.369–381.

Berge, D. ten et al., 2011. Embryonic stem cells require Wnt proteins to prevent differentiation to epiblast stem cells. *Nature Cell Biology*, 13(9), pp.1070–1075.

BioCat, Long Non-Coding RNA. Available at: https://www.biocat.com/genomics/long-non-coding-rna-lncrna [Accessed July 11, 2017].

Blackwood, E.M. & Kadonaga, J.T., 1998. Going the distance: a current view of enhancer action. *Science (New York, N.Y.)*, 281(5373), pp.60–3.

Bleeker, F.E. et al., 2009. *IDH1* mutations at residue p.R132 (IDH1 ^{R132}) occur frequently in high-grade gliomas but not in other solid tumors. *Human Mutation*, 30(1), pp.7–11.

Bockhorst, J. & Craven, M., 2005. Markov Networks for Detecting Overlapping Elements in Sequence Data. In *Advances in Neural Information Processing Systems (NIPS)*. MIT Press.

Boissel, N. et al., 2010. Prognostic impact of isocitrate dehydrogenase enzyme isoforms 1 and 2 mutations in acute myeloid leukemia: a study by the Acute Leukemia French Association group. *Journal of clinical oncology : official journal of the American Society of Clinical Oncology*, 28(23), pp.3717–23.

Bonzanni, N. et al., 2013. Hard-wired heterogeneity in blood stem cells revealed using a dynamic regulatory network model. *Bioinformatics*, 29(13).

Bortvin, A. et al., 2003. Incomplete reactivation of Oct4-related genes in mouse embryos cloned from somatic nuclei. *Development*, 130(8).

Boyer, L.A. et al., 2005. Core Transcriptional Regulatory Circuitry in Human Embryonic Stem Cells. *Cell*, 122(6), pp.947–956.

Breiman, L., 2001. Random Forests. *Machine Learning*, 45(1), pp.5–32.

Brennecke, P. et al., 2013. Accounting for technical noise in single-cell RNA-seq experiments. *Nature methods*, 10(11), pp.1093–5.

Buchanan, M. et al., 2010. *Networks in Cell Biology*, Cambridge University Press, Cambridge, U.K.

Buchel, F. et al., 2013. Path2Models: large-scale generation of computational models from biochemical pathway maps. *BMC Systems Biology*, 7(1), p.116.

Buettner, F. et al., 2015. Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells. *Nature Biotechnology*, 33(2), pp.155–60.

Byers, R. et al., 2012. Detection of IDH1 R132H Mutation in Acute Myeloid Leukemia by Mutation-specific Immunohistochemistry. *Applied Immunohistochemistry & Molecular Morphology*, 20(1), pp.37–40.

Calder, A. et al., 2013. Lengthened G1 Phase Indicates Differentiation Status in Human Embryonic Stem Cells. *Stem Cells and Development*, 22(2), pp.279–295.

Calò, V. et al., 2003. STAT proteins: From normal control of cellular events to tumorigenesis. *Journal of Cellular Physiology*, 197(2), pp.157–168.

Campbell, K., Ponting, C.P. & Webber, C., 2015. Laplacian eigenmaps and principal curves for high resolution pseudotemporal ordering of single-cell RNA-seq profiles. *bioRxiv*.

Cannoodt, R., Saelens, W., Sichien, D., et al., 2016. SCORPIUS improves trajectory inference and identifies novel modules in dendritic cell development. *bioRxiv*.

Cannoodt, R., Saelens, W. & Saeys, Y., 2016. Computational methods for trajectory inference from single-cell transcriptomics. *European Journal of Immunology*, 46(11), pp.2496–2506.

Carro, M.S. et al., 2010. The transcriptional network for mesenchymal transformation of brain tumours. *Nature*, 463(7279), pp.318–25.

Carvalho, A.M., 2009. Scoring functions for learning Bayesian networks. In *INESC-ID Tec. Rep.* p. 54.

Chang, H.H. et al., 2008. Transcriptome-wide noise controls lineage choice in mammalian progenitor cells. *Nature*, 453(7194), pp.544–547.

Chen, H. et al., 2014. Single-cell transcriptional analysis to uncover regulatory circuits driving cell fate decisions in early mouse development. *Bioinformatics*, 31(7), pp.1060–1066.

Chen, X. et al., 2008. Integration of External Signaling Pathways with the Core Transcriptional Network in Embryonic Stem Cells. *Cell*, 133(6), pp.1106–1117.

Chendrimada, T.P. et al., 2007. MicroRNA silencing through RISC recruitment of eIF6. *Nature*, 447(7146), pp.823–828.

Chigaev, A., 2015. Does aberrant membrane transport contribute to poor outcome in adult acute myeloid leukemia? *Frontiers in pharmacology*, 6, p.134.

CK-12, Transcription. Available at: https://www.ck12.org/biology/Transcription/lesson/Transcription-Advanced-BIO-ADV/ [Accessed July 11, 2017].

Clapier, C.R. & Cairns, B.R., 2009. The Biology of Chromatin Remodeling Complexes. *Annual Review of Biochemistry*, 78(1), pp.273–304.

Colombo, E., Alcalay, M. & Pelicci, P.G., 2011. Nucleophosmin and its complex network: a possible therapeutic target in hematological diseases. *Oncogene*, 30(23), pp.2595–2609.

Corces-Zimmerman, M.R. et al., 2014. Preleukemic mutations in human acute myeloid leukemia affect epigenetic regulators and persist in remission. *Proceedings of the National Academy of Sciences of the United States of America*, 111(7), pp.2548–53.

Core, L.J., Waterfall, J.J. & Lis, J.T., 2008. Nascent RNA Sequencing Reveals Widespread Pausing and Divergent Initiation at Human Promoters. *Science*, 322(5909), pp.1845–1848.

Coronado, D. et al., 2013. A short G1 phase is an intrinsic determinant of naïve embryonic stem cell pluripotency. *Stem Cell Research*, 10(1), pp.118–131.

Crick, F., 1970. Central dogma of molecular biology. *Nature*, 227(5258), pp.561–3.

Cui, D. et al., 2016. R132H mutation in IDH1 gene reduces proliferation, cell survival and invasion of human glioma by downregulating Wnt/?-catenin signaling. *The International Journal of Biochemistry & Cell Biology*, 73, pp.72–81.

Dang, L. et al., 2009. Cancer-associated IDH1 mutations produce 2-hydroxyglutarate. *Nature*, 462(7274), pp.739–744.

Davidson, E.H. et al., 2002. A genomic regulatory network for development. *Science (New York, N.Y.)*, 295(5560), pp.1669–78.

Deeds, E.J., Ashenberg, O. & Shakhnovich, E.I., 2006. A simple physical model for scaling in protein-protein interaction networks. *Proceedings of the National Academy of Sciences of the United States of America*, 103(2), pp.311–6.

Diaz-Ruiz, R., Rigoulet, M. & Devin, A., 2011. The Warburg and Crabtree effects: On the origin of cancer cell energy metabolism and of yeast glucose repression. *Biochimica et Biophysica Acta (BBA) - Bioenergetics*, 1807(6), pp.568–576.

Dillies, M.-A. et al., 2013. A comprehensive evaluation of normalization methods for Illumina high-throughput RNA sequencing data analysis. *Briefings in bioinformatics*, 14(6), pp.671–83.

Dobrin, R. et al., 2004. Aggregation of topological motifs in the Escherichia coli transcriptional regulatory network. *BMC bioinformatics*, 5, p.10.

Dohner, H. et al., 2010. Diagnosis and management of acute myeloid leukemia in adults: recommendations from an international expert panel, on behalf of the European LeukemiaNet. *Blood*, 115(3), pp.453–474.

Donovan, P.J. & de Miguel, M.P., 2003. Turning germ cells into stem cells. *Current Opinion in Genetics & Development*, 13(5), pp.463–471.

Dorritie, K.A., McCubrey, J.A. & Johnson, D.E., 2014. STAT transcription factors in hematopoiesis and leukemogenesis: opportunities for therapeutic intervention. *Leukemia*, 28(2), pp.248–257.

Drew, D., 2001. A Mathematical Model for Prokaryotic Protein Synthesis. *Bulletin of Mathematical Biology*, 63(2), pp.329–351.

Dulac, C., 2010. Brain function and chromatin plasticity. *Nature*, 465(7299), pp.728–735.

Dunham, I. et al., 2012. An integrated encyclopedia of DNA elements in the human genome. *Nature*, 489(7414), pp.57–74.

Dunn, S.-J. et al., 2014. Defining an essential transcription factor program for naïve pluripotency. *Science (New York, N.Y.)*, 344(6188), pp.1156–60.

Eichler, M., 2005. A graphical approach for evaluating effective connectivity in neural systems. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 360(1457), pp.953–67.

Eldar, A. & Elowitz, M.B., 2010. Functional roles for noise in genetic circuits. *Nature*, 467(7312), pp.167–173.

Elkon, R. et al., 2003. Genome-Wide In Silico Identification of Transcriptional Regulators Controlling the Cell Cycle in Human Cells. *Genome Research*, 13(5), pp.773–780.

Emig, D. et al., 2010. AltAnalyze and DomainGraph: analyzing and visualizing exon expression data. *Nucleic Acids Research*, 38(Web Server), pp.W755–W762.

Enders, W., 2014. *Applied econometric time series*,

Esnault, C. et al., 2008. Mediator-Dependent Recruitment of TFIIH Modules in Preinitiation Complex. *Molecular Cell*, 31(3), pp.337–346.

Faith, J.J. et al., 2007. Large-Scale Mapping and Validation of Escherichia coli Transcriptional Regulation from a Compendium of Expression Profiles. *PLoS Biol*, 5.

Fan, J. et al., 2016. Characterizing transcriptional heterogeneity through pathway and gene set overdispersion analysis. *Nature Methods*, 13(3), pp.241–244.

Fan, Y. & Peng, Q., 2016. Inferring gene regulatory networks based on spline regression and Bayesian group lasso. In *2016 17th IEEE/ACIS International Conference on Software*

Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD). IEEE, pp. 39–42.

Faria, J.P. et al., 2013. Genome-scale bacterial transcriptional regulatory networks: reconstruction and integrated analysis with metabolic models. *Briefings in Bioinformatics* .

Fauré, A. et al., 2006. Dynamical analysis of a generic Boolean model for the control of the mammalian cell cycle. *Bioinformatics (Oxford, England)*, 22(14).

Festuccia, N. et al., 2012. Esrrb is a direct Nanog target gene that can substitute for Nanog function in pluripotent cells. *Cell stem cell*, 11(4), pp.477–90.

Figueroa, M.E. et al., 2010. Leukemic IDH1 and IDH2 Mutations Result in?a Hypermethylation Phenotype, Disrupt TET2 Function, and Impair Hematopoietic Differentiation. *Cancer Cell*, 18(6), pp.553–567.

Finak, G. et al., 2015. MAST: a flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell RNA sequencing data. *Genome Biology*, 16(1), p.278.

Fisher, J. & Henzinger, T.A., 2007. Executable cell biology. *Nat Biotech*, 25(11), pp.1239–1249.

Fortunato, S., 2009. Community detection in graphs.

Friedman, J., Hastie, T. & Tibshirani, R., 2009. Regularization Paths for Generalized Linear Models via Coordinate Descent.

Friedman, N. et al., 2000. Using Bayesian networks to analyze expression data. *Journal of computational biology : a journal of computational molecular cell biology*, 7(3–4), pp.601–20.

Fritsch, S. & Guenther, F., 2016. neuralnet: Training of Neural Networks.

Fujita, A. et al., 2007. Modeling gene expression regulatory networks with the sparse vector autoregressive model. *BMC Systems Biology*, 1(1), p.39.

Furey, T.S., 2012. ChIP-seq and beyond: new and improved methodologies to detect and characterize protein-DNA interactions. *Nature reviews. Genetics*, 13(12), pp.840–52.

Garg, A. et al., 2008. Synchronous versus asynchronous modeling of gene regulatory networks. *Bioinformatics (Oxford, England)*, 24(17), pp.1917–25.

Giacomantonio, C.E. & Goodhill, G.J., 2010. A Boolean model of the gene regulatory network underlying Mammalian cortical area development. *PLoS computational biology*, 6(9).

Giguère, V., 1999. Orphan Nuclear Receptors: From Gene to Function. *Endocrine Reviews*, 20(5), pp.689–725.

Gillich, A. et al., 2012. Epiblast stem cell-based system reveals reprogramming synergy of germline factors. *Cell stem cell*, 10(4), pp.425–39.

Gilliland, D.G. & Griffin, J.D., 2002. The roles of FLT3 in hematopoiesis and leukemia. *Blood*, 100(5), pp.1532–1542.

Grafone, T. et al., 2012. An overview on the role of FLT3-tyrosine kinase receptor in acute myeloid leukemia: biology and treatment. *Oncology reviews*, 6(1), p.e8.

Graham, S.J.L. et al., 2014. BMP signalling regulates the pre-implantation development of extra-embryonic cell lineages in the mouse embryo. *Nature Communications*, 5, p.5667.

Granger, C.W.J., 1969. Investigating Causal Relations by Econometric Models and Cross-spectral Methods. *Econometrica*, 37(3), p.424.

Granger, C.W.J., 1981. Some properties of time series data and their use in econometric model specification. *Journal of Econometrics*, 16(1), pp.121–130.

Grunstein, M., 1990. Nucleosomes: regulators of transcription. *Trends in Genetics*, 6, pp.395–400.

Guelzim, N. et al., 2002. Topological and causal structure of the yeast transcriptional regulatory network. *Nature Genetics*, 31(1), pp.60–63.

Guyon, I. & Elisseeff, A., 2003. An introduction to variable and feature selection. *The Journal of Machine Learning Research*, 3, pp.1157–1182.

Hackett, J.A. & Surani, M.A., 2014. Regulatory Principles of Pluripotency: From the Ground State Up. *Cell Stem Cell*, 15(4), pp.416–430.

Haghverdi, L. et al., 2016. Diffusion pseudotime robustly reconstructs lineage branching. *Nature Methods*, 13(10), pp.845–848.

Haghverdi, L., Buettner, F. & Theis, F.J., 2015. Diffusion maps for high-dimensional single-cell analysis of differentiation data. *Bioinformatics*, 31(18), pp.2989–2998.

Hanahan, D. & Weinberg, R.A., 2011. Hallmarks of Cancer: The Next Generation. *Cell*, 144(5), pp.646–674.

Hanahan, D. & Weinberg, R.A., 2000. The Hallmarks of Cancer. *Cell*, 100(1), pp.57–70.

Hand, D.J. & Yu, K., 2001. Idiot's Bayes: Not So Stupid after All? *International Statistical Review / Revue Internationale de Statistique*, 69(3), p.385.

Hao, J. et al., 2013. Reprogramming- and pluripotency-associated membrane proteins in mouse stem cells revealed by label-free quantitative proteomics. *Journal of Proteomics*, 86, pp.70–84.

Hashimshony, T. et al., 2012. CEL-Seq: Single-Cell RNA-Seq by Multiplexed Linear Amplification. *Cell Reports*, 2(3), pp.666–673.

Hasty, J. et al., 2000. Noise-based switches and amplifiers for gene expression. *Proceedings of the National Academy of Sciences of the United States of America*, 97(5), pp.2075–80.

Haury, A.-C. et al., 2012. TIGRESS: Trustful Inference of Gene REgulation using Stability Selection. *BMC Systems Biology*, 6(1), p.145.

Heinrich, R. & Schuster, S., 1998. The modelling of metabolic systems. Structure, control and optimality. *Biosystems*, 47(1–2), pp.61–77.

Hicks, S.C. et al., 2017. Missing Data and Technical Variability in Single-Cell RNA-Sequencing Experiments. *bioRxiv*.

Hillenmeyer, M.E. et al., 2008. The Chemical Genomic Portrait of Yeast: Uncovering a Phenotype for All Genes. *Science*, 320(5874).

Home, P. et al., 2017. Genetic redundancy of GATA factors in extraembryonic trophoblast lineage ensures progression of both pre and postimplantation mammalian development. *Development*, 144(5), p.dev.145318.

Hooshangi, S., Thiberge, S. & Weiss, R., 2005. Ultrasensitivity and noise propagation in a synthetic transcriptional cascade. *Proceedings of the National Academy of Sciences of the United States of America*, 102(10), pp.3581–6.

Hotelling, H., 1933. Analysis of complex statistical variables into principal components. *J. Educ. Psychol.*, 24, pp.417–441.

Hume, D.A., 2000. Probability in transcriptional regulation and its implications for leukocyte differentiation and inducible gene expression. *Blood*, 96(7), pp.2323–8.

Huynh-Thu, V.A. et al., 2016. GENIE3.

Huynh-Thu, V.A. et al., 2010. Inferring regulatory networks from expression data using tree-based methods. *PloS one*, 5(9).

Insua, D.R. & Müller, P., 1998. Feedforward Neural Networks for Nonparametric Regression. In Springer New York, pp. 181–193.

Islam, S. et al., 2013. Quantitative single-cell RNA-seq with unique molecular identifiers. *Nature Methods*, 11(2), pp.163–166.

Jeon, Y. & Lee, J., 2011. YY1 Tethers Xist RNA to the Inactive X Nucleation Center. *Cell*, 146(1), pp.119–133.

Jeong, H. et al., 2000. The large-scale organization of metabolic networks. *Nature*, 407(6804), pp.651–654.

Ji, Z. & Ji, H., 2016. TSCAN: Pseudo-time reconstruction and evaluation in single-cell RNA-seq analysis. *Nucleic Acids Research*, 44(13), pp.e117–e117.

Jin, G. et al., 2009. *FLT3-ITD induces ara-C resistance in myeloid leukemic cells through the repression of the ENT1 expression*,

Johnson, W.E., Li, C. & Rabinovic, A., 2007. Adjusting batch effects in microarray expression data using empirical Bayes methods. *Biostatistics*, 8(1), pp.118–127.

de Jong, H., 2002. Modeling and simulation of genetic regulatory systems: a literature review. *Journal of Computational Biology*, 9(1).

Kannan, S. et al., 2013. Notch activation inhibits AML growth and survival: a potential therapeutic approach. *The Journal of Experimental Medicine*, 210(2), pp.321–337.

Karnaugh, M., 1953. The map method for synthesis of combinational logic circuits. *Transactions of the American Institute of Electrical Engineers, Part I: Communication and Electronics*, 72(5), pp.593–599.

Karr, J.R. et al., 2012. A Whole-Cell Computational Model Predicts Phenotype from Genotype. *Cell*, 150, pp.389–401.

Katayama, S. et al., 2013. SAMstrt: statistical test for differential expression in single-cell transcriptome with spike-in normalization. *Bioinformatics*, 29(22), pp.2943–2945.

Kato, T. et al., 2015. Hes1 suppresses acute myeloid leukemia development through FLT3 repression. *Leukemia*, 29(3), pp.576–585.

Kharchenko, P. V, Silberstein, L. & Scadden, D.T., 2014. Bayesian approach to single-cell differential expression analysis. *Nature methods*, 11(7), pp.740–2.

Kino, T. et al., 2010. Noncoding RNA Gas5 Is a Growth Arrest- and Starvation-Associated Repressor of the Glucocorticoid Receptor. *Science Signaling*, 3(107), p.ra8-ra8.

Kiselev, V.Y. et al., 2017. SC3: consensus clustering of single-cell RNA-seq data. *Nature Methods*, 14(5), pp.483–486.

Kolde, R., pheatmap: Pretty Heatmaps.

Krumsiek, J. et al., 2011. Hierarchical Differentiation of Myeloid Progenitors Is Encoded in the Transcription Factor Network. *PLoS ONE*, 6(8).

Kruse, J.-P. & Gu, W., 2009. Modes of p53 regulation. *Cell*, 137(4), pp.609–22.

Kumar, P., Tan, Y. & Cahan, P., 2017. Understanding development and stem cells using single cell-based analyses of gene expression. *Development*, 144(1).

Kunath, T. et al., 2007. FGF stimulation of the Erk1/2 signalling cascade triggers transition of pluripotent embryonic stem cells from self-renewal to lineage commitment. *Development*, 134(16).

L. Lun, A.T., Bach, K. & Marioni, J.C., 2016. Pooling across cells to normalize single-cell RNA sequencing data with many zero counts. *Genome Biology*, 17(1), p.75.

Lähdesmäki, H., Shmulevich, I. & Yli-Harja, O., 2003. On Learning Gene Regulatory Networks Under the Boolean Network Model. *Machine Learning*, 52(1–2), pp.147–167.

Lajtha, L.G., 1979. Stem cell concepts. *Nouvelle revue francaise d'hematologie*, 21(1), pp.59–65.

Langfelder, P. & Horvath, S., 2008. WGCNA: an R package for weighted correlation network analysis. *BMC Bioinformatics*, 9(1), p.559.

Latchman, D.S., 1996. Inhibitory transcription factors. *The international journal of biochemistry & cell biology*, 28(9), pp.965–74.

LeCun, Y., Bengio, Y. & Hinton, G., 2015. Deep learning. *Nature*, 521(7553), pp.436–444.

Leek, J.T. et al., 2010. Tackling the widespread and critical impact of batch effects in high-throughput data. *Nat Rev Genet*, 11(10), pp.733–739.

Leek, J.T. et al., 2012. The sva package for removing batch effects and other unwanted variation in high-throughput experiments. *Bioinformatics (Oxford, England)*, 28(6), pp.882–3.

Leek, J.T. & Storey, J.D., 2007. Capturing Heterogeneity in Gene Expression Studies by Surrogate Variable Analysis. *PLoS Genetics*, 3(9), p.e161.

Ley, T.J. et al., 2010. *DNMT3A* Mutations in Acute Myeloid Leukemia. *New England Journal of Medicine*, 363(25), pp.2424–2433.

Li, C. & Wang, J., 2013. Quantifying Cell Fate Decisions for Differentiation and Reprogramming of a Human Stem Cell Network: Landscape and Biological Paths C. A. Sarkar, ed. *PLoS Computational Biology*, 9(8).

Li, F. et al., 2004. The yeast cell-cycle network is robustly designed. *Proceedings of the National Academy of Sciences of the United States of America*, 101(14), pp.4781–6.

Li, L. et al., 2011. Nuclear adaptor Ldb1 regulates a transcriptional program essential for the maintenance of hematopoietic stem cells. *Nature immunology*, 12(2), pp.129–36.

Li, V.C. & Kirschner, M.W., 2014. Molecular ties between the cell cycle and differentiation in embryonic stem cells. *Proceedings of the National Academy of Sciences of the United States of America*, 111(26), pp.9503–8.

Li, W. & Ding, S., 2013. Converting Mouse Epiblast Stem Cells into Mouse Embryonic Stem Cells by Using Small Molecules. In pp. 31–37.

Liang, J. & Han, J., 2012. Stochastic Boolean networks: an efficient approach to modeling gene regulatory networks. *BMC systems biology*, 6(1), p.113.

Liang, S., Fuhrman, S. & Somogyi, R., 1998. Reveal, a general reverse engineering algorithm for inference of genetic network architectures. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, pp.18–29.

Liaw, A. & Wiener, M., 2002. Classification and Regression by randomForest.

Lim, C.Y. et al., 2016. BTR: training asynchronous Boolean models using single-cell expression data. *BMC Bioinformatics*, 17(1), p.355.

Liu, Z., Malone, B. & Yuan, C., 2012. Empirical evaluation of scoring functions for Bayesian network model selection. *BMC bioinformatics*.

Lobry, C. et al., 2013. Notch pathway activation targets AML-initiating cell homeostasis and differentiation. *The Journal of Experimental Medicine*, 210(2), pp.301–319.

Logsdon, B., 2016. metanetwork.

Love, M.I., Huber, W. & Anders, S., 2014. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology*, 15(12), p.550.

Lowenberg, B., Downing, J.R. & Burnett, A., 1999. Acute Myeloid Leukemia. *New England Journal of Medicine*, 341(14), pp.1051–1062.

Lubeck, E. et al., 2014. Single-cell in situ RNA profiling by sequential hybridization. *Nature methods*, 11(4), pp.360–1.

Lun, A.T.L. et al., 2016. A step-by-step workflow for low-level analysis of single-cell RNA-seq data. *F1000Research*, 5, p.2122.

Luscombe, N.M. et al., 2004. Genomic analysis of regulatory network dynamics reveals large topological changes. *Nature*, 431(7006), pp.308–312.

Ma, H.-W., Buer, J. & Zeng, A.-P., 2004. Hierarchical structure and modules in the Escherichia coli transcriptional regulatory network revealed by a new top-down approach. *BMC Bioinformatics*, 5(1), p.199.

van der Maaten, L. & Hinton, G., 2008. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(Nov), pp.2579–2605.

Mackarehtschian, K. et al., 1995. Targeted disruption of the flk2/flt3 gene leads to deficiencies in primitive hematopoietic progenitors. *Immunity*, 3(1), pp.147–61.

Macosko, E.Z. et al., 2015. Highly Parallel Genome-wide Expression Profiling of Individual Cells Using Nanoliter Droplets. *Cell*, 161(5), pp.1202–1214.

Mangan, S. & Alon, U., 2003. Structure and function of the feed-forward loop network motif. *Proceedings of the National Academy of Sciences of the United States of America*, 100(21), pp.11980–5.

Mangan, S., Zaslaver, A. & Alon, U., 2003. The coherent feedforward loop serves as a sign-sensitive delay element in transcription networks. *Journal of molecular biology*, 334(2), pp.197–204.

Marbach, D. et al., 2012. Wisdom of crowds for robust gene network inference. *Nature methods*, 9(8), pp.796–804.

Mardis, E.R. et al., 2009. Recurring Mutations Found by Sequencing an Acute Myeloid Leukemia Genome. *New England Journal of Medicine*, 361(11), pp.1058–1066.

Margolin, A.A. et al., 2006. ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC bioinformatics*.

Martello, G. et al., 2012. Esrrb is a pivotal target of the Gsk3/Tcf3 axis regulating embryonic stem cell self-renewal. *Cell stem cell*, 11(4), pp.491–504.

Martínez-Antonio, A. et al., 2006. Internal-sensing machinery directs the activity of the regulatory network in Escherichia coli. *Trends in Microbiology*, 14(1), pp.22–27.

de Matos Simoes, R. & Emmert-Streib, F., 2012. Bagging statistical network inference from large-scale gene expression data. *PloS one*, 7(3).

McConnell, B.B. & Yang, V.W., 2010. Mammalian Krüppel-like factors in health and diseases. *Physiological reviews*, 90(4), pp.1337–81.

McDavid, A., Finak, G. & Gottardo, R., 2016. The contribution of cell cycle to heterogeneity in single-cell RNA-seq data. *Nature Biotechnology*, 34(6), pp.591–593.

Meinshausen, N. & Bühlmann, P., 2010. Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(4), pp.417–473.

Meneghini, M.D., Wu, M. & Madhani, H.D., 2003. Conserved histone variant H2A.Z protects euchromatin from the ectopic spread of silent heterochromatin. *Cell*, 112(5), pp.725–36.

von Mering, C., 2002. Comparative assessment of large-scale data sets of protein-protein interactions. *Nature*, 417, pp.399–403.

Metzeler, K.H. et al., 2011. *TET2* Mutations Improve the New European LeukemiaNet Risk Classification of Acute Myeloid Leukemia: A Cancer and Leukemia Group B Study. *Journal of Clinical Oncology*, 29(10), pp.1373–1381.

Meyer, D. et al., 2017. e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien.

Meyer, P.E., Lafitte, F. & Bontempi, G., 2008. minet: A R/Bioconductor Package for Inferring Large Transcriptional Networks Using Mutual Information. *BMC Bioinformatics*, 9(1), p.461.

Mishina, Y. et al., 1995. Bmpr encodes a type I bone morphogenetic protein receptor that is essential for gastrulation during mouse embryogenesis. *Genes & development*, 9(24), pp.3027–37.

Miyazono, K., Kamiya, Y. & Morikawa, M., 2010. Bone morphogenetic protein receptors and signal transduction. *Journal of Biochemistry*, 147(1), pp.35–51.

Moignard, V. et al., 2013. Characterization of transcriptional networks in blood stem and progenitor cells using high-throughput single-cell gene expression analysis. *Nat Cell Biol*, 15(4), pp.363–372.

Moignard, V. et al., 2015. Decoding the regulatory network of early blood development from single-cell gene expression measurements. *Nature Biotechnology*, 33(3), pp.269–276.

Moore, M.J. & Proudfoot, N.J., 2009. Pre-mRNA Processing Reaches Back toTranscription and Ahead to Translation. *Cell*, 136(4), pp.688–700.

Morikawa, M. et al., 2016. *BMP Sustains Embryonic Stem Cell Self-Renewal through Distinct Functions of Different Krüppel-like Factors*,

Morris, S.A. et al., 2010. Origin and formation of the first two distinct cell types of the inner cell mass in the mouse embryo. *Proceedings of the National Academy of Sciences*, 107(14), pp.6364–6369.

Murphy, K. & Mian, S., 1999. Modelling Gene Expression Data using Dynamic Bayesian Networks. *Technical report, University of California at Berkeley, Berkeley, CA.*

Murphy, K.F., Balázsi, G. & Collins, J.J., 2007. Combinatorial promoter design for engineering noisy gene expression. *Proceedings of the National Academy of Sciences of the United States of America*, 104(31), pp.12726–31.

Müssel, C., Hopfensitz, M. & Kestler, H.A., 2010. BoolNet--an R package for generation, reconstruction and analysis of Boolean networks. *Bioinformatics (Oxford, England)*, 26(10), pp.1378–80.

Nadler, B. et al., 2005. Diffusion Maps, Spectral Clustering and Eigenfunctions of Fokker-Planck operators.

Nair, V. & Hinton, G.E., 2010. Rectified Linear Units Improve Restricted Boltzmann Machines. , pp.807–814.

Nakagawa, M. et al., 2007. Generation of induced pluripotent stem cells without Myc from mouse and human fibroblasts. *Nature Biotechnology*, 26(1), pp.101–106.

Nepusz, T. & Vicsek, T., 2012. Controlling edge dynamics in complex networks. *Nature Physics*, 8(7), pp.568–573.

Nestorowa, S. et al., 2016. A single-cell resolution map of mouse hematopoietic stem and progenitor cell differentiation. *Blood*, 128(8), pp.e20-31.

Nichols, J. & Smith, A., 2012. Pluripotency in the embryo and in culture. *Cold Spring Harbor perspectives in biology*, 4(8), p.a008128.

NIH, Stem Cell Information. Available at: https://stemcells.nih.gov/info/2001report/appendixB.htm [Accessed September 29, 2017].

Niwa, H. et al., 1998. Self-renewal of pluripotent embryonic stem cells is mediated via activation of STAT3. *Genes & development*, 12(13), pp.2048–60.

Nottrott, S., Simard, M.J. & Richter, J.D., 2006. Human let-7a miRNA blocks protein production on actively translating polyribosomes. *Nature Structural & Molecular Biology*, 13(12), pp.1108–1114.

Numata, A. et al., 2005. Signal transducers and activators of transcription 3 augments the transcriptional activity of CCAAT/enhancer-binding protein alpha in granulocyte colony-stimulating factor signaling pathway. *The Journal of biological chemistry*, 280(13), pp.12621–9.

Oberhardt, M.A., Palsson, B.Ø. & Papin, J.A., 2009. Applications of genome-scale metabolic reconstructions. *Molecular Systems Biology*, 5.

Olsson, A. et al., 2016. Single-cell analysis of mixed-lineage states leading to a binary cell fate choice. *Nature*, 537(7622), pp.698–702.

van Oosten, A.L. et al., 2012. JAK/STAT3 signalling is sufficient and dominant over antagonistic cues for the establishment of naive pluripotency. *Nature Communications*, 3, p.817.

Opgen-Rhein, R. & Strimmer, K., 2007. From correlation to causation networks: a simple approximate learning algorithm and its application to high-dimensional plant gene expression data. *BMC systems biology*, 1(1), p.37.

Orkin, S.H. & Zon, L.I., 2008. Hematopoiesis: An Evolving Paradigm for Stem Cell Biology. *Cell*, 132(4), pp.631–644.

Pandolfi, P.P. et al., 1995. Targeted disruption of the GATA3 gene causes severe abnormalities in the nervous system and in fetal liver haematopoiesis. *Nature Genetics*, 11(1), pp.40–44.

Papanayotou, C. & Collignon, J., 2014. Activin/Nodal signalling before implantation: setting the stage for embryo patterning. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 369(1657), pp.20130539–20130539.

Park, J. et al., 2014. Identifying Functional Gene Regulatory Network Phenotypes Underlying Single Cell Transcriptional Variability. *Progress in biophysics and molecular biology*.

Parks, S.K., Chiche, J. & Pouysségur, J., 2013. Disrupting proton dynamics and energy metabolism for cancer therapy. *Nature Reviews Cancer*, 13(9), pp.611–623.

Paschka, P. et al., 2010. IDH1 and IDH2 Mutations Are Frequent Genetic Alterations in Acute Myeloid Leukemia and Confer Adverse Prognosis in Cytogenetically Normal Acute Myeloid Leukemia With NPM1 Mutation Without FLT3 Internal Tandem Duplication. *Journal of Clinical Oncology*, 28(22), pp.3636–3643.

Pauklin, S. & Vallier, L., 2013. The cell-cycle state of stem cells determines cell fate propensity. *Cell*, 155(1), pp.135–47.

Paule, M.R. & White, R.J., 2000. SURVEY AND SUMMARY Transcription by RNA polymerases I and III. *Nucleic Acids Research*, 28(6), pp.1283–1298.

Peters, L. & Meister, G., 2007. Argonaute Proteins: Mediators of RNA Silencing. *Molecular Cell*, 26(5), pp.611–623.

Picelli, S. et al., 2014. Full-length RNA-seq from single cells using Smart-seq2. *Nature Protocols*, 9(1), pp.171–181.

Pierson, E. & Yau, C., 2015. ZIFA: Dimensionality reduction for zero-inflated single-cell gene expression analysis. *Genome biology*, 16, p.241.

Pieters, T. & van Roy, F., 2014. Role of cell-cell adhesion complexes in embryonic stem cell biology. *Journal of Cell Science*, 127(12), pp.2603–2613.

Pillai, R.S. et al., 2005. Inhibition of Translational Initiation by Let-7 MicroRNA in Human Cells. *Science*, 309(5740).

Pina, C. et al., 2012. Inferring rules of lineage commitment in haematopoiesis. *Nature cell biology*, 14(3), pp.287–94.

Qi, J. & Michoel, T., 2012. Context-specific transcriptional regulatory network inference from global gene expression maps using double two-way t-tests. *Bioinformatics*, 28(18), pp.2325–2332.

Qiu, X. et al., 2017. Reversed graph embedding resolves complex single-cell developmental trajectories. *bioRxiv*.

Quia, Chapter 19 Eukaryotic Genomes. Available at: https://www.quia.com/jg/1277396list.html [Accessed July 11, 2017].

Raj, A. & van Oudenaarden, A., 2008. Nature, Nurture, or Chance: Stochastic Gene Expression and Its Consequences. *Cell*, 135(2), pp.216–226.

Ramos, C.A. et al., 2006. Evidence for diversity in transcriptional profiles of single hematopoietic stem cells. *PLoS genetics*, 2(9).

Ramskold, D. et al., 2012. Full-length mRNA-Seq from single-cell levels of RNA and individual circulating tumor cells. *Nature Biotechnology*, 30(8), pp.777–782.

Rau, A. et al., 2010. An Empirical Bayesian Method for Estimating Biological Networks from Temporal Microarray Data. *Statistical Applications in Genetics and Molecular Biology*, 9(1).

Rau, A., 2016. ebdbNet: Empirical Bayes Estimation of Dynamic Bayesian Networks.

Ray, S. et al., 2009. Context-dependent function of regulatory elements and a switch in chromatin occupancy between GATA3 and GATA2 regulate Gata2 transcription during trophoblast differentiation. *The Journal of biological chemistry*, 284(8), pp.4978–88.

Reitman, Z.J. & Yan, H., 2010. Isocitrate Dehydrogenase 1 and 2 Mutations in Cancer: Alterations at a Crossroads of Cellular Metabolism. *JNCI Journal of the National Cancer Institute*, 102(13), pp.932–941.

Reményi, A., Schöler, H.R. & Wilmanns, M., 2004. Combinatorial control of gene expression. *Nature Structural & Molecular Biology*, 11(9), pp.812–815.

Resendis-Antonio, O. et al., 2005. Modular analysis of the transcriptional regulatory network of E. coli. *Trends in Genetics*, 21(1), pp.16–20.

Reyes de Mochel, N.S. et al., 2015. BMP signaling is required for cell cleavage in preimplantation-mouse embryos. *Developmental Biology*, 397(1), pp.45–55.

Riedmiller, M. & Braun, H., 1993. A direct adaptive method for faster backpropagation learning: the RPROP algorithm. In *IEEE International Conference on Neural Networks*. IEEE, pp. 586–591.

Rinn, J.L. & Chang, H.Y., 2012. Genome Regulation by Long Noncoding RNAs. *Annual Review of Biochemistry*, 81(1), pp.145–166.

Robinson, M.D., McCarthy, D.J. & Smyth, G.K., 2010. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26, pp.139–140.

Robinson, M.D. & Oshlack, A., 2010. A scaling normalization method for differential expression analysis of RNA-seq data. *Genome Biology*, 11(3), p.R25.

Robinton, D.A. & Daley, G.Q., 2012. The promise of induced pluripotent stem cells in research and therapy. *Nature*, 481(7381), pp.295–305.

Roman, G.T. et al., 2006. Single-cell manipulation and analysis using microfluidic devices. *Analytical and Bioanalytical Chemistry*, 387(1), pp.9–12.

Rosenfeld, N. et al., 2005. Gene Regulation at the Single-Cell Level. *Science*, 307(5717).

Rosenfeld, N., Elowitz, M.B. & Alon, U., 2002. Negative autoregulation speeds the response times of transcription networks. *Journal of molecular biology*, 323(5), pp.785–93.

Saito, R. et al., 2009. CrxOS maintains the self-renewal capacity of murine embryonic stem cells. *Biochemical and Biophysical Research Communications*, 390(4), pp.1129–1135.

Sander, E. & Grummt, I., 1997. Oligomerization of the transcription termination factor TTF-I: implications for the structural organization of ribosomal transcription units. *Nucleic Acids Research*, 25(6), pp.1142–1147.

Saunders, A., Core, L.J. & Lis, J.T., 2006. Breaking barriers to transcription elongation. *Nature Reviews Molecular Cell Biology*, 7(8), pp.557–567.

Savatier, P. et al., 1996. Withdrawal of differentiation inhibitory activity/leukemia inhibitory factor up-regulates D-type cyclins and cyclin-dependent kinase inhibitors in mouse embryonic stem cells. *Oncogene*, 12(2), pp.309–22.

Schaffter, T., Marbach, D. & Floreano, D., 2011. GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods. *Bioinformatics*

(Oxford, England), 27(16), pp.2263–70.

Schaub, M., Henzinger, T. & Fisher, J., 2007. Qualitative networks: a symbolic approach to analyze biological signaling networks. *BMC Systems Biology*, 1(1), p.4.

Schnittger, S. et al., 2010. IDH1 mutations are detected in 6.6% of 1414 AML patients and are associated with intermediate risk karyotype and unfavorable prognosis in adults younger than 60 years and unmutated NPM1 status. *Blood*, 116(25).

Scholkopf, B., Smola, A. & Muller, K.-R., 1998. Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Computation*, 10(5), pp.1299–1319.

Schurch, N.J. et al., 2016. How many biological replicates are needed in an RNA-seq experiment and which differential expression tool should you use? *RNA*, 22(6), pp.839–851.

Scialdone, A. et al., 2015. Computational assignment of cell-cycle stage from single-cell transcriptome data. *Methods*, 85, pp.54–61.

Scialdone, A. et al., 2016. Resolving early mesoderm diversification through single-cell expression profiling. *Nature*, 535(7611), pp.289–293.

Scutari, M., 2010. Learning Bayesian Networks with the bnlearn R Package. *Journal of Statistical Software*, 35(3), pp.1–22.

Segal, E. et al., 2003. Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nature Genetics*, 34(2), pp.166–176.

Setty, M. et al., 2016. Wishbone identifies bifurcating developmental trajectories from single-cell data. *Nature Biotechnology*, 34(6), pp.637–645.

Shannon, P., 2003. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res.*, 13, pp.2498–2504.

Shapiro, E., Biezuner, T. & Linnarsson, S., 2013. Single-cell sequencing-based technologies will revolutionize whole-organism science. *Nature reviews. Genetics*, 14(9), pp.618–30.

Sharova, L. V. et al., 2007. Global gene expression profiling reveals similarities and differences among mouse pluripotent stem cells of different origins and strains. *Developmental Biology*, 307(2), pp.446–459.

Shimamura, T. et al., 2009. Recursive regularization for inferring gene networks from time-course gene expression profiles. *BMC Systems Biology*, 3(1), p.41.

Shlush, L.I. et al., 2014. Identification of pre-leukaemic haematopoietic stem cells in acute leukaemia. *Nature*, 506(7488), pp.328–333.

Shmulevich, I. et al., 2002. Probabilistic Boolean Networks: a rule-based uncertainty model for gene regulatory networks. *Bioinformatics (Oxford, England)*, 18(2), pp.261–74.

Siegel, R. et al., 2011. Cancer statistics, 2011. *CA: A Cancer Journal for Clinicians*, 61(4), pp.212–236.

Sigal, A. et al., 2006. Variability and memory of protein levels in human cells. *Nature*, 444(7119), pp.643–646.

Sing, T. et al., 2005. ROCR: visualizing classifier performance in R. *Bioinformatics*, 21(20), pp.3940–3941.

Sinha, R. et al., 2017. Index Switching Causes “Spreading-Of-Signal” Among Multiplexed Samples In Illumina HiSeq 4000 DNA Sequencing. *bioRxiv*.

Sokolova, M., Japkowicz, N. & Szpakowicz, S., 2006. *Beyond accuracy, f-score and ROC: a family of discriminant measures for performance evaluation* A. Sattar & B. Kang, eds., Berlin, Heidelberg: Springer Berlin Heidelberg.

Spitale, R.C., Tsai, M.-C. & Chang, H.Y., 2011. RNA templating the epigenome: long noncoding RNAs as molecular scaffolds. *Epigenetics*, 6(5), pp.539–43.

Stadtfeld, M. et al., 2008. Defining molecular cornerstones during fibroblast to iPS cell reprogramming in mouse. *Cell stem cell*, 2(3), pp.230–40.

Ståhlberg, A. & Bengtsson, M., 2010. Single-cell gene expression profiling using reverse transcription quantitative real-time PCR. *Methods (San Diego, Calif.)*, 50(4), pp.282–8.

Stasevich, T.J. et al., 2014. Regulation of RNA polymerase II activation by histone acetylation in single living cells. *Nature*, 516(7530), pp.272–275.

Staveley, B.E., Molecular & Developmental Biology. Available at: http://www.mun.ca/biology/desmid/brian/BIOL3530/DB_03/DBNVert1.html [Accessed September 29, 2017].

Stavridis, M.P. et al., 2007. A discrete period of FGF-induced Erk1/2 signalling is required for vertebrate neural specification. *Development*, 134(16).

Stegle, O. et al., 2012. Using probabilistic estimation of expression residuals (PEER) to obtain increased power and interpretability of gene expression analyses. *Nature protocols*, 7(3), pp.500–7.

Sterner, D.E. & Berger, S.L., 2000. Acetylation of histones and transcription-related factors. *Microbiology and molecular biology reviews : MMBR*, 64(2), pp.435–59.

Stuart, J.M. et al., 2003. A Gene-Coexpression Network for Global Discovery of Conserved Genetic Modules. *Science*, 302(5643), pp.249–255.

Subramanian, A. et al., 2005. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences of the United States of America*, 102(43), pp.15545–50.

Sumazin, P. et al., 2011. An Extensive MicroRNA-Mediated Network of RNA-RNA Interactions Regulates Established Oncogenic Pathways in Glioblastoma. *Cell*, 147(2), pp.370–381.

Suzuki, H. et al., 2009. The transcriptional network that controls growth arrest and differentiation in a human myeloid leukemia cell line. *Nature genetics*, 41(5), pp.553–62.

Takahashi, S., 2011a. Current findings for recurring mutations in acute myeloid leukemia. *Journal of hematology & oncology*, 4, p.36.

Takahashi, S., 2011b. Downstream molecular pathways of FLT3 in the pathogenesis of acute myeloid leukemia: biology and therapeutic implications. *Journal of hematology & oncology*, 4, p.13.

Tanaka, T.S. et al., 2002. Gene Expression Profiling of Embryo-Derived Stem Cells Reveals Candidate Genes Associated With Pluripotency and Lineage Specificity. *Genome Research*, 12(12), pp.1921–1928.

Tang, F. et al., 2009. mRNA-Seq whole-transcriptome analysis of a single cell. *Nature methods*, 6(5), pp.377–82.

Tang, Y. et al., 2012. Jak/Stat3 Signaling Promotes Somatic Cell Reprogramming by Epigenetic Regulation. *STEM CELLS*, 30(12), pp.2645–2656.

Team, R.C., 2013. R: A language and environment for statistical computing. . *R Foundation for Statistical Computing, Vienna, Austria.*

Tenen, D.G. et al., 1997. Transcription factors, normal myeloid development, and leukemia. *Blood*, 90(2), pp.489–519.

Thiede, C. et al., 2006. Prevalence and prognostic impact of NPM1 mutations in 1485 adult patients with acute myeloid leukemia (AML). *Blood*, 107(10), pp.4011–4020.

Thiele, I. & Palsson, B.O., 2010. A protocol for generating a high-quality genome-scale metabolic reconstruction. *Nat. Protocols*, 5(1), pp.93–121.

Toyooka, Y. et al., 2008. Identification and characterization of subpopulations in undifferentiated ES cell culture. *Development*, 135(5), pp.909–918.

Trapnell, C. et al., 2014. The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nature biotechnology*, 32(4), pp.381–6.

Tsai, F.-Y. et al., 1994. An early haematopoietic defect in mice lacking the transcription factor GATA-2. *Nature*, 371(6494), pp.221–226.

Tsuboi, A. et al., 1999. Olfactory neurons expressing closely linked and homologous odorant receptor genes tend to project their axons to neighboring glomeruli on the olfactory bulb. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 19(19), pp.8409–18.

Uranishi, K. et al., 2016. Esrrb directly binds to Gata6 promoter and regulates its expression with Dax1 and Ncoa3. *Biochemical and Biophysical Research Communications*, 478(4), pp.1720–1725.

Vallejos, C.A. et al., 2017. Normalizing single-cell RNA sequencing data: challenges and opportunities. *Nature Methods*, 14(6), pp.565–571.

Vallejos, C.A., Marioni, J.C. & Richardson, S., 2015. BASiCS: Bayesian Analysis of Single-Cell Sequencing Data Q. Morris, ed. *PLOS Computational Biology*, 11(6), p.e1004333.

Vassar, R., Ngai, J. & Axel, R., 1993. Spatial segregation of odorant receptor expression in the mammalian olfactory epithelium. *Cell*, 74(2), pp.309–18.

Vu, T.N. et al., 2016. Beta-Poisson model for single-cell RNA-seq data analyses. *Bioinformatics*, 32(14), pp.2128–2135.

Wang, B. et al., 2017. Visualization and analysis of single-cell RNA-seq data by kernel-based similarity learning. *Nature Methods*, 14(4), pp.414–416.

Wang, J. & Song, Y., 2017. Single cell sequencing: a distinct new field. *Clinical and Translational Medicine*, 6(1), p.10.

Wang, K.C. & Chang, H.Y., 2011. Molecular mechanisms of long noncoding RNAs. *Molecular cell*, 43(6), pp.904–14.

Wang, L., Wang, S. & Li, W., 2012. RSeQC: quality control of RNA-seq experiments. *Bioinformatics*, 28(16), pp.2184–2185.

Wang, P. et al., 2015. ChIP-Array 2: integrating multiple omics data to construct gene regulatory networks. *Nucleic acids research*, 43(W1), pp.W264-9.

Ward, P.S. et al., 2010. The Common Feature of Leukemia-Associated IDH1 and IDH2 Mutations Is a Neomorphic Enzyme Activity Converting ?-Ketoglutarate to 2-Hydroxyglutarate. *Cancer Cell*, 17(3), pp.225–234.

Warnes, G.R. et al., 2015. gplots: Various R Programming Tools for Plotting Data.

Watt, F.M. & Driskell, R.R., 2010. The therapeutic potential of stem cells. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 365(1537), pp.155–63.

Weinreb, C., Wolock, S. & Klein, A., 2017. SPRING: a kinetic interface for visualizing high dimensional single-cell expression data. *bioRxiv*.

Welch, J.D., Hartemink, A.J. & Prins, J.F., 2016. SLICER: inferring branched, nonlinear cellular trajectories from single cell RNA-seq data. *Genome Biology*, 17(1), p.106.

Wellmann, S. et al., 2005. FLT3 mutations in childhood acute lymphoblastic leukemia at first relapse. *Leukemia*, 19(3), pp.467–468.

Wickham, H., 2009. *ggplot2: elegant graphics for data analysis*, Springer New York.

Wills, Q.F. et al., 2013. Single-cell gene expression analysis reveals genetic associations masked in whole-tissue experiments. *Nature Biotechnology*, 31(8), pp.748–752.

Wilson, N.K. et al., 2010. Combinatorial transcriptional control in blood stem/progenitor cells: genome-wide analysis of ten major transcriptional regulators. *Cell stem cell*, 7(4), pp.532–44.

Wilson, N.K. et al., 2015. Combined Single-Cell Functional and Gene Expression Analysis Resolves Heterogeneity within Stem Cell Populations. *Cell Stem Cell*, 16(6), pp.712–24.

Wong, E. et al., 2012. Biological network motif detection: principles and practice. *Briefings in bioinformatics*, 13(2), pp.202–15.

Wray, J. et al., 2011. Inhibition of glycogen synthase kinase-3 alleviates Tcf3 repression of the pluripotency network and increases embryonic stem cell resistance to differentiation. *Nature Cell Biology*, 13(7), pp.838–845.

Wu, L., Fan, J. & Belasco, J.G., 2006. MicroRNAs direct rapid deadenylation of mRNA. *Proceedings of the National Academy of Sciences*, 103(11), pp.4034–4039.

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

Wu, W.-H. et al., 2005. Swc2 is a widely conserved H2AZ-binding module essential for ATP-dependent histone exchange. *Nature Structural & Molecular Biology*, 12(12), pp.1064–1071.

Xiao, F. et al., 2016. Inferring Gene Regulatory Networks Using Conditional Regulation Pattern to Guide Candidate Genes E. Hernandez-Lemus, ed. *PLOS ONE*, 11(5), p.e0154953.

Xu, C. & Su, Z., 2015. Identification of cell types from single-cell transcriptomes using a novel clustering method. *Bioinformatics*, 31(12), pp.1974–1980.

Xu, H. et al., 2010. Toward a complete in silico, multi-layered embryonic stem cell regulatory network. *Wiley interdisciplinary reviews. Systems biology and medicine*, 2(6), pp.708–33.

Xu, Y. et al., 2013. Proliferation rate of somatic cells affects reprogramming efficiency. *The Journal of biological chemistry*, 288(14), pp.9767–78.

Yamanaka, S. & Blau, H.M., 2010. Nuclear reprogramming to a pluripotent state by three approaches. *Nature*, 465(7299), pp.704–712.

Yamanaka, Y., Lanner, F. & Rossant, J., 2010. FGF signal-dependent segregation of primitive endoderm and epiblast in the mouse blastocyst. *Development*, 137(5), pp.715–

724.

Yan, L. et al., 2013. Single-cell RNA-Seq profiling of human preimplantation embryos and embryonic stem cells. *Nature structural & molecular biology*, 20(9), pp.1131–9.

Yang, J. et al., 2010. Stat3 activation is limiting for reprogramming to ground state pluripotency. *Cell stem cell*, 7(3), pp.319–28.

Yeo, J.-C. et al., 2014. Klf2 Is an Essential Factor that Sustains Ground State Pluripotency. *Cell Stem Cell*, 14(6), pp.864–872.

Yu, H. & Gerstein, M., 2006. Genomic analysis of the hierarchical structure of regulatory networks. *Proceedings of the National Academy of Sciences of the United States of America*, 103(40), pp.14724–31.

Yuan, Y. et al., 2011. Directed Partial Correlation: Inferring Large-Scale Gene Regulatory Network through Induced Topology Disruptions D. Di Bernardo, ed. *PLoS ONE*, 6(4), p.e16835.

Yugi, K. et al., 2016. Trans-Omics: How To Reconstruct Biochemical Networks Across Multiple “Omic” Layers. *Trends in Biotechnology*, 34(4), pp.276–290.

Zarayeneh, N. et al., 2016. Integrative Gene Regulatory Network inference using multi-omics data. In *2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, pp. 1336–1340.

Zhang, Y. et al., 2012. Mutation analysis of isocitrate dehydrogenase in acute lymphoblastic leukemia. *Genetic testing and molecular biomarkers*, 16(8), pp.991–5.

Zhao, S. et al., 2009. Glioma-Derived Mutations in IDH1 Dominantly Inhibit IDH1 Catalytic Activity and Induce HIF-1α. *Science*, 324(5924).

Zhong, H. et al., 2002. Nuclear expression of hypoxia-inducible factor 1α protein is heterogeneous in human malignant cells under normoxic conditions. *Cancer Letters*, 181(2), pp.233–238.

Zhu, J. et al., 2010. Characterizing Dynamic Changes in the Human Blood Transcriptional Network S. Miyano, ed. *PLoS Computational Biology*, 6(2), p.e1000671.

Zwaka, T.P. & Thomson, J.A., 2004. A germ cell origin of embryonic stem cells? *Development*, 132(2).