

New protocols and computational tools for scRNAseq analysis

by

Luyi Tian

ORCID: [0000-0003-3420-3685](https://orcid.org/0000-0003-3420-3685)

A thesis submitted in total fulfillment for the
degree of Doctor of Philosophy

in the
Department of Medical Biology
Faculty of Medicine, Dentistry and Health Sciences
THE UNIVERSITY OF MELBOURNE

December 2020

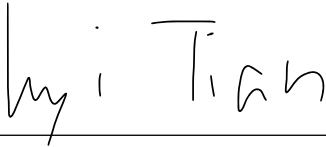
Abstract

The fast development of single cell RNA sequencing (scRNAseq) presents new challenges in data analysis and opportunities for protocol development. To address challenges in data preprocessing, I developed *scPipe*, an R/Bioconductor package that integrates bar-code demultiplexing, read alignment, UMI-aware gene-level quantification and quality control of raw sequencing data generated by multiple protocols. Results from *scPipe* can be used as input for downstream analyses and can be easily incorporated into R-based pipelines with other tools. In order to compare different computational methods for scRNAseq data, I generated a realistic benchmark experiment that included single cells and admixtures of cells or RNA to create ‘pseudo cells’ from up to five distinct cancer cell lines. Multiple datasets were generated using both droplet and plate-based scRNA-seq protocols and processed by *scPipe*. We found pipelines suited to different types of data for different tasks. Our data and analysis provide a comprehensive framework for benchmarking most common scRNA-seq analysis steps. Finally, I developed single cell full-length transcript sequencing by sampling (FLT-seq), together with the computational pipeline *FLAMES* to perform isoform discovery and quantification, splicing analysis and mutation detection in single cells. With FLT-seq and *FLAMES*, I performed a comprehensive characterization of the full-length isoform landscape in single cells of different types and species and found conserved functional modules that were enriched for alternative transcript usage in different cell populations, including ribosome biogenesis and mRNA splicing. The datasets, protocols and tools that I developed and generated are useful resources for the single cell research community.

Declaration of Authorship

I, Luyi Tian, declare that this thesis titled, ‘New protocols and computational tools for scRNAseq analysis’ and the work presented in it are my own. I confirm that:

- The thesis comprises only my original work towards the Doctor of Philosophy except where indicated in the preface;
- due acknowledgement has been made in the text to all other material used; and
- the thesis is fewer than the maximum word limit in length, exclusive of tables, maps, bibliographies and appendices as approved by the Research Higher Degrees Committee.

Signed:  _____

Date: Jan 04 2021

Preface

This thesis contains work that has been published or is under review.

The second chapter was published in *PLoS Computational Biology* on 10/08/2018. I wrote the *scPipe* package and the manuscript and contributed 80% of the work in this publication.

The third chapter was published in *Nature Methods* on 27/05/2019 and I contributed 70% of the work in this publication. I designed and conducted the experiment to generate the benchmarking data and wrote manuscript. Xueyi Dong performed the methods comparison for data normalization, Saskia Freytag helped compare the clustering methods and Kim-Anh Lê Cao and Abolfazl Jalal Abadi helped with the data integration comparison.

The fourth chapter was submitted to *Nature Methods* on 26/08/2020 and I contributed 70% of the work in this study. I conducted the experiment to generate data related to the *scmixology* samples. Jafar Jabbari developed the FLT-seq protocol with my assistance. Members of James Ryall's lab performed experiments related to mouse muscle stem cells. Rachel Thijssen and colleagues performed experiments related to the CLL patient sample. Shanika Amarasinghe performed the scATACseq data analysis and generated Figure 1F and Figure S4D. I developed the *FLAMES* software, performed data analysis to generate the remaining figures and wrote the manuscript.

More detailed contribution information for each author can be found in the 'Author contributions' section in each chapter.

Acknowledgements

First I want to acknowledge my supervisor Matt Ritchie for his fantastic scientific mentorship. Matt gives a lot of time for us students and is always kind and supportive of new ideas and thoughts, even when he disagrees with them. I will miss the Ritchie lab lunches that are held more frequently than in any other lab at WEHI. Secondly I want to acknowledge my co-supervisor Shalin Naik for his passion and endless new ideas. If I have ever learnt any new knowledge or developed an interest in immunology during my PhD, I need to give credit to him. I also thank my co-supervisor Terry Speed for all of his blunt, but very helpful feedback on my project.

Over the years I have had many great collaborations and discussions with the Ritchie lab and Naik lab members. Daniela Zalcenstein and Jaring Schreuder taught me how to do experiments, which is hard for a dry lab person. Shian Su helped me in software development and data analysis in all projects. I also thank the Walter and Eliza Hall Institute of Medical Research for creating a friendly and supportive environment for students.

Lastly I want to thank my family and friends who supported me throughout my PhD. My mother Guizhen Liu is always my role model and her braveness and eagerness to face new challenges and learn new knowledge has indisputably shaped my character. Finishing a PhD during a global pandemic is not easy and I thank my partner Siying Tan who mentally supported me during these hard times.

Contents

Abstract	i
Declaration of Authorship	ii
Preface	iii
Acknowledgements	iv
List of Figures	vii
List of Tables	ix
1 Introduction	1
1.1 The development of single cell RNA sequencing technology	1
1.2 Computational challenges in single cell transcriptomics	3
1.2.1 Data preprocessing	3
Quality control	4
Normalization	5
Imputation	6
1.2.2 Clustering and pseudo-time ordering	7
1.2.3 Differential gene expression analysis in the single cell context . . .	9
1.2.4 Benchmarking single cell RNA sequencing analysis methods	11
2 scPipe: A flexible R/Bioconductor preprocessing pipeline for single-cell RNA-sequencing data	13
2.1 Introduction	14
2.2 Design and implementation	15
2.3 Results	16
2.4 Availability and future directions	23
2.5 References	24
3 Benchmarking single cell RNA-sequencing analysis pipelines using mixture control experiments	27
3.1 Introduction	28
3.2 Results	29
3.2.1 scRNAseq mixology provides ground truth for benchmarking . . .	29

3.2.2	Comparisons of normalization and imputation methods	31
3.2.3	Comparisons of clustering methods	32
3.2.4	Comparisons of trajectory analysis methods	33
3.2.5	Comparisons of data integration methods	35
3.3	Discussion	37
3.4	Methods	38
3.5	References	44
4	Comprehensive characterization of single cell full-length isoforms in human and mouse with long-read sequencing	48
4.1	Introduction	49
4.2	Results	50
4.2.1	High-throughput single-cell full-length transcriptome sequencing with FLT-seq	50
4.2.2	Single-cell isoform detection and quantification with FLAMES	53
4.2.3	Characterization of isoforms reveals the distinct splicing landscapes of different cell populations	54
4.2.4	Common classes of genes with differential transcript usage across samples	56
4.2.5	FLT-seq links coding sequence variation to transcriptome heterogeneity	60
4.3	Discussion	61
4.4	Methods	63
4.5	References	70
5	Concluding remarks	74
A	Additional Material for Chapter 3	77
B	Additional Material for Chapter 4	91
B.1	Supplementary figures	92
B.2	Supplementary tables	100
C	Other Research Activities	102
C.1	Publications with significant contributions	102
C.2	Conference presentations	103
	Bibliography	104

List of Figures

2.1	The scPipe workflow	17
2.2	Example QC plots that can be created using output from scPipe to assess data quality	19
2.3	A pairwise scatter plot of the quality control metrics collected by scPipe for the mouse blood cell dataset	20
2.4	Screenshot of an HTML report created by scPipe for the mouse blood cell dataset	21
2.5	Analysis results produced with SingleCellExperiment-compatible packages from Bioconductor	22
2.6	Comparing scPipe and CellRanger	23
3.1	Overview of the scRNAseq mixology experimental design and benchmark analysis	30
3.2	Comparisons of normalization and imputation methods using multiple mixture datasets	32
3.3	Comparisons of scRNAseq clustering methods	34
3.4	Comparisons of scRNAseq trajectory analysis methods	35
3.5	Comparisons of data integration methods for batch effect correction for the RNA mixture and the four single cell experiments	36
3.6	Summary of results from methods comparisons using scRNAmixology datasets	38
4.1	Overview of experimental design. FLT-seq and FLAMES methods and basic summary statistics	52
4.2	Overview of the single-cell isoform-level analysis from FLAMES	54
4.3	Summary of differential transcript usage results from FLAMES	58
4.4	Summary of differential allele frequency analysis to detect coding mutations	60
A.1	Overview of the RNA mixture and cell mixture experimental designs	78
A.2	Violin plots of quality control metrics for cells from each benchmarking dataset	79
A.3	Visualization of representative benchmarking datasets using t-SNE and UMAP and violin plot of the number of doublets	80
A.4	Violin plots of silhouette widths for different normalization methods	81
A.5	Example clustering results and summary of clustering performance using ARI and the number of clusters	82
A.6	Coefficients from linear models used to quantify the impact different methods have on the trajectory analysis and data integration results	83

A.7	Visualization of results from the trajectory analysis methods evaluated in our study	84
A.8	Additional data integration results for the single cell and RNA mixture datasets	85
A.9	Performance of clustering on the RNA mixture datasets after data integration is applied	86
B.1	Dimensionality reduction (UMAP) of all samples	92
B.2	Comparing FLAMES with other methods using the SIRV dataset	93
B.3	Percentage of transcript coverage of individual reads stratified by length	94
B.4	Comparing FLAMES results with database annotations and other methods	95
B.5	Binary splicing and downsampling analysis	96
B.6	Genes with differential transcript usage	97
B.7	Identification of genomic variation from long read by FLAMES	99

List of Tables

2.1	Summary of the data preprocessing software currently available for scRNAseq analysis and the particular tasks covered by each package	22
A.1	Summary of the benchmarking datasets generated	87
A.2	Summary of the data characteristics and data analysis tasks that can be compared by each experimental design	88
A.3	Summary of integrative methods used to combine data from different protocols and datasets	89
A.4	Individual performance metrics obtained from benchmarking analysis, organized by task	90
A.5	Run times for different analysis pipelines	90
B.1	Summary of samples included in this study	100
B.2	Results of differential transcript usages analysis	101
B.3	Results of GO enrichment analysis using genes with differential transcript usage	101

Chapter 1

Introduction

1.1 The development of single cell RNA sequencing technology

Single cell RNA sequencing (scRNAseq) refers to the whole transcriptome quantification of single cell gene expression by next generation sequencing. The core concept is to use cellular barcodes to label transcripts from different cells which enables them to be pooled, which greatly reduces the cost. Falling costs along with increasing throughput has been a key driver behind the rapid expansion of this field [1]. scRNAseq protocols have different mechanisms for isolating cells and preparing nucleic acids for sequencing, with approaches such as Indrop [2] and Seqwell [3] using template switching while CEL-seq2 [4] and MARS-seq [5] use in vitro transcription (IVT). While throughput has increased, the breadth of single cell sequencing applications has also expanded to include multi-omics, such as coupling the measurement of surface markers [6], epigenetic modifications [7], chromatin states [8], the genome (i.e. DNA rather than RNA) [9] or CRISPR screen output together with transcriptome data on the same cells [10]. The application of these technologies has led to the identification of new cell types in complex tissues and led to the revision of differentiation hierarchies for stem cells and boosted other new findings where the heterogeneity of individual cells is important. The combination of multiple omics allows researchers to study the epigenetic landscape on top of transcriptomic changes [7], and to dissect regulatory circuits by combining gene knockout information together with transcriptomic measurements using protocols

such as CROP-seq [11] and Perturb-seq [10]. While short-read sequencing is commonly used for scRNAseq profiling, recent developments in other technologies has provided researchers with additional options. For example, emerging in-situ sequencing methods aim to characterize gene expression profiles while retaining spatial information in the tissue context, which is typically lost in regular scRNAseq approaches. The gene expression levels are quantified by imaging and represented by fluorescent intensity [12].

A major short-coming of single-cell transcriptome technology is its inability to measure alternative splicing. Most droplet-based scRNAseq protocols are based on 3' end tag sequencing, which lacks full-transcript coverage and cannot be used to detect alternative splicing. Some plate-based protocols, such as SMART-seq have similar coverage compared to bulk RNAseq and have been used to detect alternative splicing events [13]. However, these protocols are based on Illumina short-read sequencing, which can detect changes in single splicing sites, but have limited ability to recover full-length transcripts and correlate distal alternative splicing events [14]. The rapid development of long-read sequencing platforms provides an alternative to short-read sequencing with the added benefit of being able to sequence full-length transcripts in single cells. However, the lower throughput and accuracy compared to short-read sequencing is a major limitation of this approach, and currently, it cannot replace short-read sequencing. Methods have been developed to resolve these issues. R2C2 [15] employs rolling circle amplification to sequence the same read multiple times and call consensus reads to reduce error rate, which will unavoidably reduce the sequencing throughput. RAGE-seq [16] reduces the number of genes assessed by targeted sequencing and cannot profile the whole transcriptome. The recently published ScNaUmi-seq protocol [17] utilized the high-throughput of PromethION platform and deep sequencing for error correction with UMIs. However, this method is not cost-effective as high sequencing saturation is required for UMI correction and the PromethION platform is still roughly 10 times more expensive than Illumina sequencing for a similar throughput. Taken together, there are still limitations with current approaches and new methods are needed to address these issues.

1.2 Computational challenges in single cell transcriptomics

1.2.1 Data preprocessing

The rapid development of scRNAseq technology not only brings new opportunities, but also presents new challenges in data analysis. Compared with bulk RNA-sequencing (RNAseq), it is harder to find a good method for scRNAseq data that can tease apart the technical and biological noise to accurately recover the original mRNA abundance [18]. The technical noise in scRNAseq data comes from several sources. RNA capture efficiency for scRNAseq is usually 10-30% [19] due to the sampling and enzyme efficiency in the protocol. Usually the mRNA is captured by polydT primers and reverse transcribed (RT) to cDNA. The capture and RT efficiency can significantly alter the quality of the library, as it is the first step of the protocol. Apart from the RT step, there are also several rounds of library amplification and clean up, which may result in further loss of material [4]. Taken together, the effect of these factors will be to reduce the level of complexity in the final sequencing libraries and introduce bias. Moreover, the competitive nature of RNAseq means a certain number of reads will be distributed between genes.

Given that scRNAseq usually involves cell barcoding and pooling of materials from multiple single cells, differences in library size between cells can arise from variation in cell size, library preparation and other technical artifacts. The sequencing depth also plays a role in data quality and there have been several studies that have focused on the differences between shallow sequencing with larger numbers of cells, or deep sequencing with fewer cells [20]. In summary, the technical noise, sometimes also referred to as unwanted variation, introduced during the scRNAseq experiment and sequencing, is unlikely to be completely removed due to the stochastic nature of RNAseq. The general goals for data preprocessing, including quality control, normalization and imputation, is to remove technical noise and other systematic effects while leaving the biological variation [21]. Because of the differences in barcoding strategies, processing the raw FASTQ files to generate a gene count matrix is a non-trivial task. Although there have been tools developed to process data from certain protocols [22, 23], there were none that worked across protocols. The lack of availability of an easy-to-use, cross-platform preprocessing tool led me to develop the *scPipe* package [24] described later.

Quality control Data quality control (QC) is an essential step that generally follows acquisition of a gene count matrix from the raw FASTQ file. QC can have a profound impact on the interpretation of the final results, as the presence of low quality cells can confound the downstream analysis. Since manual QC may introduce bias and uncertainty, efforts have been made to automate the quality control process. Ilicic *et al.* [25] performed a systematic study of the quality control metrics that are useful for identifying low quality cells by acquiring independent quality annotations (via images) through the Fluidigm C1 platform. They then used these labels to train a support vector machine that could be used to identify low quality cells in new datasets. However, such methods have not been widely used because of the need for training data, which can be hard to access for some protocols, such as Drop-seq, where visual inspection of the individual cells is not possible. To overcome this limitation, methods that rely on the quality control metrics alone have been proposed such as in *scrn* [26].

These methods use the distribution of one or more quality control metrics to look for outlier cells. The basic underlying assumption is that the majority of the cells are high quality, thus the low quality cells will be the outliers in terms of QC metrics such as the total counts or number of genes detected. The results of such automatic quality control methods should be still be carefully explored and visual inspection of the quality control metrics is recommended to avoid systematically removing cells of one particular type that may have extreme QC metrics (i.e. a low number of genes detected) that are driven by real biological differences. The heterogeneous nature of single cell and scRNAseq protocols increases the difficulty of defining poor quality cells. Poor quality cells can arise from multiple sources, and have different characteristics in quality control measures [27], such that broken or dying cells may have increased numbers of mitochondrial reads or failure of the library preparation will reduce the total counts and the ERCC spike-in counts (where present). The ambiguity in the definition of the cell quality highlights the importance of manually checking the quality control metrics. Even though a number of methods and strategies exist to identify poor quality cells, there are still some open questions in the field, such as how to distinguish between genuine single cells with low transcript abundance and low quality cells, and how to find doublets in droplet-based protocols [28].

Normalization A wide range of different assumptions and models are used in scRNAseq data normalization, partly due to the complex noise structure where capture efficiency, cell size and sampling noise can each contribute to cell-to-cell variability. Normalization methods can be roughly split into two classes depending on the model assumptions. The first is normalization by scaling, where the key assumption is that most of the genes are not differentially expressed (DE) between cells, so the major factors driving trends of count differences between cells are related to technical variability. Such methods therefore summarize the cell-to-cell technical variability to a scaling factor for each cell and apply it to the gene counts to obtain an expected gene expression value. Global scaling methods have been widely used for bulk RNAseq data normalization [29] and have been adapted to the scRNAseq setting in the *scrn* package [30]. To overcome the sparsity of scRNAseq data, *scrn* constructs a pseudo-reference RNAseq dataset from pooling single cell data and deconvolves the global scaling factor into single cell specific scaling factors. In contrast to the approach of *scrn* where single scaling factors are used for each cell, *SCnorm* [31] tries to group genes with different sequencing depths by quantile regression and estimates scale factors within each group, in order to adjust the gene specific bias that cannot be removed with universal scale factors. Another example of scaling by regression is *Linnorm* [32], in which scaling is based on a group of house-keeping genes.

In contrast to the scaling methods, other normalization methods model the read counts as a distribution where parameters are both influenced by technical and biological variability. The commonly used distributions are Negative Binomial and Gamma distributions, because of their ability to model the over-dispersion of read counts that exceeds the Poisson sampling variance [29]. The *BASiCS* [33] method uses a Bayesian hierarchical model, where the parameters of the Negative Binomial are also random variables that follow distributions that are centered around technical artifacts such as cell size and capture efficiency. Similar to *BASiCS*, Ding *et al.* [34] considers gene expression to follow a Gamma distribution, and use a Gamma regression model to fit the detected and added ERCC spike-in molecules in order to calculate the molecular concentration of genes. The issue of sparsity in scRNAseq data has led to the development of methods that update the existing Negative Binomial model to include drop-out events, which are usually defined as the zero gene counts that result from inefficiencies in the protocol, and represent missing transcripts. Although the existence of drop-outs in scRNAseq

data has been widely accepted [21], there has been much debate about its distribution. Some have shown that these drop-outs exceed the theoretical frequency predicted by the Negative Binomial, and methods such as *ZIFA* [35] and *ZINB-WaVE* [36] have been developed to include separate terms to model drop-out events. Others have shown that the drop-out from technical replicates follows a Poisson distribution [37], indicating that the drop-out events can be described by Negative Binomial statistics and that zero-inflation is caused by biological heterogeneity instead of technical artifacts. With different assumptions, these methods are likely to suit different scenarios. To overcome this limitation, ensemble-based approaches such as *scone* [38] have been proposed to run multiple normalization methods on the same dataset and rank the performance based on quality control metrics.

Imputation Although several normalization methods have been proposed to adjust for drop-out events in scRNAseq data, a fundamental limitation of normalization is that the normalized gene expression values are scaled from the original values by either applying a scaling factor or estimating the distribution parameters. As a result, if a data point is zero then after normalization, it will remain zero. Since this effect is the same for all genes, especially lowly expressed ones which suffer most from the drop-out events, the cell-to-cell technical variability will be reflected by the different frequencies of zeros, which can introduce unwanted correlation structures in the data [39]. To address this issue, several imputation frameworks have been proposed to recover the lost counts due to drop-out [40–42]. The most common assumption made by these methods is that the drop-out rate is correlated to the true gene expression. The relationship between drop-out rate and average gene expression is then calculated by logistic regression and the true expression is recovered by considering the drop-out rate [40]. Instead of explicitly modelling the drop-out events, *MAGIC* [42] smooths the data, assuming the underlying structure is a manifold. Similar to *MAGIC*, the method developed by Wagner *et al.* [37] uses KNN-based smoothing to recover the drop-out loss, where the UMI count for each gene is averaged with its neighboring cells. Although not in widespread use at present, the fundamental limit of normalization in recovering signal for the drop-out genes is likely to see an increase in the application of imputation methods in the future [27].

1.2.2 Clustering and pseudo-time ordering

A major application for scRNAseq is in the characterization of gene expression heterogeneity, which is hidden in bulk RNAseq, with the objective of finding new cell types or cell states. This goal is usually achieved by clustering cells with similar transcriptomic profiles. The clustering methods available for single cell analysis can be classified into two main groups, based on whether or not they take established knowledge as input. Some methods, such as *RCA* [43] and *scrn*, incorporate prior knowledge during the clustering procedure, including bulk RNAseq data and marker genes. The majority of the methods available do not require this kind of input. Instead they often allow users to specify the number of clusters expected in the data.

Apart from the degree of supervision, clustering methods also vary in terms of their scope. Some assume the data has been cleaned and non-informative genes have been filtered out. Other packages integrate clustering with preprocessing steps such as normalization and feature selection. Considering the levels of noise and high dimensionality of scRNAseq data, most clustering methods perform dimensionality reduction on the gene expression matrix and then cluster cells using a low dimensional representation of the original matrix [27]. A range of different dimensionality reduction algorithms are used for this step, with PCA, t-SNE and UMAP being the most popular choices [44].

The next essential step after dimensionality reduction is partitioning cells into multiple clusters, which is usually achieved by grouping cells by their distance or similarity. Some methods, like *SIMLR* [45], are designed around the similarity matrix calculation. The cell-cell distance is also calculated non-explicitly during the clustering, such as in *k*-means, in which the distance matrix calculation are calculated within each iteration. A distance measure is required by all clustering methods, and the choice here can have a substantial influence on the results. In early studies there was limited focus on the distance calculation and Euclidean pairwise distances are usually the default, with most methods emphasizing feature selection and dimensionality reduction [46]. However, the critical importance of the choice of distance measures has been recognized in some recent studies, such as *RAFSIL* [47] and *SIMLR* [45], using random forest and multi-kernel methods respectively to calculate distances instead of the Euclidean measure. Ensemble-based methods, which utilize multiple clustering algorithms for robustness, have also been proposed recently. The *SC3* [46] package performs consensus clustering on the

results of multiple clustering methods, followed by k -means clustering on the consensus matrix, which can also be considered as a generalized distance matrix. Similar to *SC3*, the *clusterExperiment* [48] package applies multiple clustering methods and combines these result by consensus clustering. In general, ensemble-based methods are shown to be more robust than individual methods.

While in many cases, cell heterogeneities are described as multiple cell types or cell states coexisting in a cohort, it is notable that the transcriptome of these cells does not form discrete clusters that represent different cell states, but exhibits a broad continuum with cells in different cell states and transient stages [49]. For example, in hematopoietic stem cell differentiation, the multi-potent stem and progenitor cells differentiate to more than one mature blood cell type in a continuous fashion [50], involving multiple transient stages such as from pro-B cells to mature B cells [51, 52]. As a result, clustering methods may not be optimal in this situation and additional analysis is required to distinguish cells within different lineages.

Several computational approaches have been proposed to reconstruct these lineages from scRNAseq data. Although each method has its own assumptions, most of them involve the construction of a similarity matrix [27], as per clustering methods. Instead of grouping cells by similarity, these methods will build a tree structure to reflect the branching dynamics, thus giving a pseudo-temporal ordering [53]. The trajectory construction method can be summarized into three major steps. The first is to perform dimensionality reduction on the gene expression matrix and generate a similarity matrix. There are various techniques to achieve this goal, such as diffusion maps [54], PCA and Laplacian eigenmaps [55]. The second is to construct the tree structure, with minimum spanning trees (MSTs) being the most popular method for this task [56–59]. The third step usually involves interpretation of the lineage tree, such as giving the pseudo-time value for each cell or finding the genes associated with a branching event [60].

Although designed for the same purpose, these methods still differ in several aspects, apart from the algorithm they use. The first is the preprocessing stage of the method. Despite the fact that most of these methods take a gene expression matrix as input, the way they specify and treat this input varies considerably. Some methods are only designed for scRNAseq data, while others also suit mass cytometry data [61]. Some methods, such as *SLICER* [62] will perform feature selection after data input, in order

to remove noise from the data. Meanwhile, instead of sub-setting genes, *ECLAIR* [57] subsamples the cells to generate multiple trees. Then an ensemble of the trees is generated to obtain a consensus tree. The second major difference relates to when they build the lineage tree. Some methods, such as *Slingshot* [48] and *ECLAIR*, find clusters through the dimensionally-reduced representation of original data and connecting the clusters with a tree structure. Others such as *DPT* [63] and *SLICER* apply lineage construction directly on the low dimensional matrix. The third main difference is in the level of supervision allowed by the methods to guide the pseudo-time ordering. Usually, the user-specified parameters are related to the underlying algorithms that each method uses. As trajectory analysis involves the construction of a tree structure, most methods allow users to specify the starting state, or root of the tree and some also allow users to give marker genes as prior knowledge [39]. The methods that use *KNN* usually also apply a method to find the optimal value of k from the data, but users can also specify k from their own experience.

1.2.3 Differential gene expression analysis in the single cell context

Differential expression (DE) analysis is a long-standing topic in transcriptomic research, with a huge amount of effort dedicated to methodology for this task since microarray technology launched the field in the early 2000s. A classic DE analysis aims to prioritize genes that correlate to conditions or phenotypes, which can be discrete or continuous. There are many well-established tools for bulk RNAseq DE analysis [29, 64]. Unlike bulk RNAseq, scRNAseq data exhibits more noise and generally has lower library sizes. The sparsity of scRNAseq data poses new challenges in the DE analysis and a number of new methods have been developed to deal with this problem. *SCDE* [65] uses a two-component mixture model to capture the drop-out events, which is similar to what has been used in normalization and imputation. The DE analysis can then be reformulated as a comparison between the posterior distributions of different sub-populations. Similar to *SCDE*, *BASiCS* can also perform DE analysis after model fitting, by estimating the posterior distributions of the log-fold-change in gene expression between the known groups [66]. Although the DE methods tailored for scRNAseq data all claim to outperform those designed for bulk RNAseq data, a recent study that systematically evaluated 30 different DE methods on a number of real and simulated scRNAseq datasets showed that ‘*methods developed for bulk RNAseq do not perform notably worse*

than those developed for *scRNAseq*' [67]. More studies are needed to elucidate whether *scRNAseq* DE analysis benefits significantly from more complex modeling or whether the over-dispersed nature of the data can be handled sufficiently with classical methods.

In contrast to the standard DE analysis paradigm, where samples are split into different experimental groups representing known conditions, *scRNAseq* data can have tremendous heterogeneity within a given group. So the DE analysis is often preceded by dimensionality reduction and clustering, with the hope of finding new cell populations or cell states [68]. The DE analysis is then performed on clusters or trajectories, in order to find marker genes that are enriched in certain clusters or correlated with pseudotime. Such DE analysis has been embedded in several popular *scRNAseq* frameworks such as *Seurat* [69], and has resulted in the discovery of new cell populations and marker genes [70]. However, extra care should be taken when interpreting the results, because the clustering and pseudotime analysis are first performed on the same gene expression data, with the results used to guide the DE analysis. It has been shown in a recent benchmarking study that the error rate is inflated using these methods [67], raising concerns about the inherent bias in such analyses. The fact that *scRNAseq* measures gene expression at the single cell level presents us with the opportunity to quantify not only the mean expression of a gene but also its variance and distribution. Because of the complexity of biological variability, gene expression kinetics may result in distributions with multi modality. The normalization and imputation methods that model the gene expression as a distribution will naturally yield the variance estimate. Some packages such as *BASiCS* also provides functions to analyse differential dispersion of genes in different groups. The *scDD* package [71] can look beyond the variance and also find genes with different distributions using mixture modeling. Although the variance of *scRNAseq* suffers from technical noise and the analysis of variability should be conducted with care, such methods can give new insights into gene expression dynamics that cannot be seen in bulk RNAseq analysis [66].

With the development of new *scRNAseq* protocols, the context of differential gene expression has been extended beyond comparing different groups. New methods such as RNA velocity computationally generate pseudotime ordering on cell differentiation trajectories [72], and new protocols on single cell lineage-tracing now facilitate the mapping of clonal relationships to construct the cellular history in a more unambiguous

way. They enable differential gene expression analysis with pseudotime or differentiation stages used as covariates to correlate gene expression with cellular differentiation. Spatial transcriptomics provides another avenue by allowing differential gene expression analysis in space [73]. Although these methods correlate gene expression with time and space, the gene expression measurements are still the same and can be characterized by Negative Binomial distributions. The emergence of full-length transcript sequencing provides makes it possible to quantify transcript-level expression which allows differential transcript usage analysis. However, computational methods for this task are still limited. Methods such as *DRIMseq* and *DEXseq* have been developed for bulk differential transcript usage analysis, while similar methods have not been reported for single cells.

1.2.4 Benchmarking single cell RNA sequencing analysis methods

Because of the importance of understanding the characteristics of different omics methods, there have been many studies aimed at benchmarking different methods and platforms from microarray to RNAseq, using specially designed control experiments. There are several ways to design a control experiment and acquire the necessary ground truth. Spike-in controls, which are a set of mRNA molecules of known composition and concentration, were popular in early research studies to evaluate the performance of Affymetrix microarrays [74]. Control experiments often involve the use of reference RNA samples (e.g. MAQC, and SEQC used commercially available human reference RNAs) or well characterized cell lines in which gene expression is known and stable. Samples can then be run in their pure form, or mixed in different proportions to induce controlled biological variability [75]. A third strategy is to dilute the RNA sample to create systematic decreases or increases in gene expression affecting every gene in the sample in a predictable way when comparisons are made back to the undiluted reference sample.

The rapid development of transcriptomic technology for single cell analysis has created a need for systematic benchmarking in order to understand the strengths and weaknesses of different platforms and computational methods. To date, there has been a growing interest on the comparison of protocols and computational methods. Svensson *et al.* [19] compared the quality of many publicly available datasets based on spike-in controls, together with in-house data generated using mouse embryonic stem cells and human

brain total RNA. They found that the detection rate, defined by the minimum input spike-in molecules required for a reliable detection, is among 10-100 molecules for most protocols. They also found that the minimum read depth to get saturation is around 0.5 to 1 million reads per cells. Two large scale benchmarking studies have been published recently, both utilizing mixtures from multiple species and cell types as reference to compare different scRNAseq protocols [76, 77]. Results from multiple benchmarking studies showed considerable heterogeneity of different protocols ability to detect cell-type markers, impacting their predictive value and suitability for integration into reference cell atlases.

Apart from comparisons of protocols and platforms, the benchmarking of different computational methods is emerging as a major topic, as the number of methods continues to expand (> 713 methods as at August 2020 according to Zappia *et al.* [78], <https://www.scrna-tools.org/>). Methods comparisons for scRNAseq data are challenging for two main reasons. The first stems from the diversity of tasks (normalization, clustering, trajectory analysis as reviewed above) and approaches taken by different methods (for instance some clustering methods focus on robustness, while others focus on rare cluster detection). The second arises from the complex nature of scRNAseq data that make it hard to design realistic control datasets that mimic the properties of a real scRNAseq experiment. Current benchmarking studies are performed using publicly available datasets, to get a good representation of different biological and technical variabilities. Sonesson *et al.* [67] published a comprehensive comparison on differential expression analysis of scRNAseq data, evaluating 30 methods on dozens of datasets. A framework that ranks different normalization methods called *scone* [38] has been developed to allow consistent data processing and methods ranking based on QC measures and other unwanted variations. A comprehensive comparison of more than 20 trajectory methods using simulated and public datasets to evaluate their performance in various areas, including different types of trees and code availability was also recently published [53]. These studies highlight the heterogeneity of computational methods and the importance of choosing reliable methods for different tasks.

Chapter 2

**scPipe: A flexible
R/Bioconductor preprocessing
pipeline for single-cell
RNA-sequencing data**

scPipe: a flexible R/Bioconductor preprocessing pipeline for single-cell RNA-sequencing data

Luyi Tian^{1,2}, Shian Su¹, Xueyi Dong^{1,3}, Daniela Amann-Zalcenstein¹, Christine Biben¹, Azadeh Seidi⁴, Douglas J. Hilton^{1,2}, Shalin H. Naik¹, Matthew E. Ritchie^{1,2,5}

1 Molecular Medicine Division, The Walter and Eliza Hall Institute of Medical Research, 1G Royal Parade, Parkville, 3052, Australia.

2 Department of Medical Biology, The University of Melbourne, Parkville, 3010, Australia.

3 College of Life Science, Zhejiang University, 866 Yuhangtang Road, Hangzhou, Zhejiang Province, 310058, P.R. China.

4 Australian Genome Research Facility, 1G Royal Parade, Parkville, 3052, Australia.

5 School of Mathematics and Statistics, The University of Melbourne, Parkville, 3010, Australia.

Abstract

Single-cell RNA sequencing (scRNA-seq) technology allows researchers to profile the transcriptomes of thousands of cells simultaneously. Protocols that incorporate both designed and random barcodes have greatly increased the throughput of scRNA-seq, but give rise to a more complex data structure. There is a need for new tools that can handle the various barcoding strategies used by different protocols and exploit this information for quality assessment at the sample-level and provide effective visualization of these results in preparation for higher-level analyses.

To this end, we developed *scPipe*, a R/Bioconductor package that integrates barcode demultiplexing, read alignment, UMI-aware gene-level quantification and quality control of raw sequencing data generated by multiple protocols that include CEL-seq, MARS-seq, Chromium 10X, Drop-seq and Smart-seq. *scPipe* produces a count matrix that is essential for downstream analysis along with an HTML report that summarises data quality. These results can be used as input for downstream analyses including normalization, visualization and statistical testing. *scPipe* performs this processing in a few simple R commands, promoting reproducible analysis of single-cell data that is compatible with the emerging suite of open-source scRNA-seq analysis tools available in R/Bioconductor and beyond. The *scPipe* R package is available for download from <https://www.bioconductor.org/packages/scPipe>.

Introduction

Advances in single-cell transcriptomic profiling technologies allow researchers to measure gene activity in thousands of cells simultaneously, enabling exploration of gene expression variability [1], identification of new cell types [2] and the study of transcriptional programs involved in cell differentiation [3]. The introduction of cellular barcodes, sequences distinct for each cell attached to the dT-primer, has increased the throughput and substantially reduced the cost of single-cell RNA sequencing (scRNA-seq). These barcodes allow for the demultiplexing of reads after cells are pooled together for sequencing. Apart from cellular barcodes, molecular barcodes or unique molecular identifiers (UMIs), are frequently employed to remove PCR duplicates and allow identification of unique mRNA molecules, thereby reducing technical noise. The multiple levels of barcoding used in scRNA-seq experiments create additional challenges in data processing together with new opportunities for quality control (QC). Different protocols use different barcode configurations, which means a flexible approach to data preprocessing is required.

A large number of software tools have already been tailored to scRNA-seq analysis [4], the majority of which are focused on downstream tasks such as clustering and trajectory analysis. Methods for preprocessing tend to focus on specific tasks such as *UMI-tools* [5], *umitools* (<http://brwnj.github.io/umitools/>) and *umis* [6] which have been developed for handling random UMIs and correcting UMI sequencing errors. Other tools such as *CellRanger* [7], *dropEst* [8] and *dropseqPipe* (<https://github.com/Hoohm/dropSeqPipe>) on the other hand offer a complete preprocessing solution for data generated by droplet based protocols. Other packages such as *scater* [9], and *scrn* [10] work further downstream by preprocessing the counts to perform general QC and normalization of scRNA-seq data.

scPipe was developed to address the lack of a comprehensive R-based workflow for processing sequencing data from different protocols that can accommodate both UMIs and sample barcodes, map reads to the genome and summarise these results into gene-level counts. Additionally this pipeline collates useful metrics for QC during preprocessing that can be later used to filter genes and samples. In the remainder of this article we provide an overview of the main features of our *scPipe* software and demonstrate its use on various in-house generated and publicly available scRNA-seq datasets.

Design and Implementation

Single-cell RNA-seq datasets analysed

Mouse hematopoietic lineage dataset

Single cell expression profiling of the main hematopoietic lineages in mouse (erythroid, myeloid, lymphoid, stem/progenitors) was performed using a modified CEL-seq2 [11] protocol. B lymphocytes (B220+ FSC-Alow), erythroblasts (Ter119+ CD44+, FSC-Amid/high), granulocytes (Mac1+ Gr1+) and high-end progenitor/stem (Lin- Kit+ Sca1+) were sorted from the bone marrow of a C57BL/6 10-13 week old female mouse. T cells (CD3+ FSC-Alow) were isolated from the thymus of the same mouse. Bone marrow and thymus were dissociated mechanically, washed and stained with antibodies for 1hr on ice. Single cells were deposited on a 384 well plate using an Aria cell sorter (Beckman). Index data was collected and our adapted CEL-seq2 protocol used to generate a library for sequencing. The reads were sequenced by an Illumina Nextseq 500 and processed by *scPipe*. This dataset is available under GEO accession number GSE109999.

Human lung adenocarcinoma cell line dataset

The cell lines H2228, NCI-H1975 and HCC827 were retrieved from ATCC (<https://www.atcc.org/>) and cultured in Roswell Park Memorial Institute (RPMI) 1640 medium with 10% fetal calf serum (FCS) and 1% Penicillin-Streptomycin. The cells were grown independently at 37°C with 5% carbon dioxide until near 100% confluency. Cells were PI stained and 120,000 live cells were sorted for each cell line by FACS to acquire an accurate equal mixture of live cells from the three cell lines. This mixture was then processed by the Chromium 10X single cell platform using the manufacturer's (10X Genomics) protocol and sequenced with an Illumina Nextseq 500. Filtered gene expression matrices were generated using *CellRanger* (10X Genomics) and *scPipe* independently. For *CellRanger*, we used the default parameters, with `--expect-cells=4000`. For *scPipe*, we processed the 4,000 most enriched cell barcodes. The GRCh38 human genome and ENSEMBL v91 human gene annotation were used for both the *scPipe* and *CellRanger* analysis. In *scPipe*, the number of components used in the outlier detection step was set to `comp=2` for quality control to remove poor quality cells. This dataset is available under GEO accession number GSE111108.

Ischaemic Sensitivity of Human Tissue dataset

A publicly available Chromium 10X dataset from the Human Cell Atlas project was also preprocessed using *scPipe* and analysed with highly compatible Bioconductor packages. This dataset of roughly 2,000 cells comes from the first spleen harvested in a project seeking to study sensitivity to ischaemia in 3

different tissues types. For this data *scPipe* was run with quality and sequence filters turned off in `sc_trim_barcode`, cellular barcodes were detected from a sample of the first 5 million reads using `sc_detect_bc`, this resulted in 2,273 detected barcodes. The sequences were aligned to the GRCh38 human genome using *Rsubread* and counts were assigned based on ENSEMBL v91 human gene annotations using *scPipe* and outliers were detected with `comp=2` in the automatic outlier detection step.

***scPipe* implementation details**

scPipe is an R [12] / Bioconductor [13] package that can handle data generated from all popular 3' end scRNA-seq protocols and their variants, such as CEL-seq, MARS-seq, Chromium 10X and Drop-seq. Data from non-UMI protocols generated by Smart-seq and Smart-seq2 can also be handled. The pipeline begins with FASTQ files and outputs both a gene count matrix and a variety of QC statistics. These are presented in a standalone HTML report generated by *rmarkdown* [14] that includes various plots of QC metrics and other data summaries. The *scPipe* package is written in R and C++ and uses the *Rcpp* package [15,16] to wrap the C++ code into R functions and the *Rhtslib* package [17] for BAM input/output. The key aspects are implemented in C++ for efficiency.

Results

The *scPipe* workflow

FASTQ reformatting

The *scPipe* workflow (Figure 1) generally begins with paired-end FASTQ data which is passed to the function `sc_trim_barcode`, which reformats the reads by trimming the barcode and UMI from the reads and moving this information into the read header. There are options to perform some basic filtering in this step, including removing reads with low quality sequence in the barcode and UMI regions and filtering out low complexity reads, most of which are non-informative repetitive sequences such as polyA. The output FASTQ file contains transcript sequences, with barcode information merged into the read names.

Read alignment, exon mapping and barcode demultiplexing

The next stage of preprocessing involves sequence alignment. For this task, any popular RNA-seq read aligner that produces BAM formatted output can be used. By default, *scPipe* uses the fast and convenient R-based *Rsubread* [18] aligner which is available on Linux and Mac OS operating systems. For publicly available datasets from sources such as GEO, FASTQ files are often available per cell and these need to be aligned first before they can be processed further using *scPipe*. For all different types of data, aligned reads in the BAM file are then assigned to exons by the `sc_exon_mapping` function according to a user provided annotation. Using a similar strategy to *featureCounts* [19], we divide chromosomes into non-overlapping bins and assess the overlap of aligned fragments with exons that fall within the bin to reduce the search complexity. This function records the mapping result, together with the UMI and cell barcodes available from the optional fields of the BAM file with specific BAM tags. By default we use the official BAM tag BC for cell barcode and OX for UMI sequence.

Next, the `sc_demultiplex` function is used to demultiplex results per cell using the sample barcode information. For CEL-seq and MARS-seq, the demultiplexing is based on the list of designed barcode sequences that the user provides, and allows for mismatches during the cell barcode matching step. For 10X and Drop-seq where the cell barcode sequences need to be identified from the reads, the function `sc_detect_bc` can be applied to identify enriched barcodes in the data that can then be supplied to `sc_demultiplex`. This function can accept a list of possible barcodes that may be present in the dataset (if available), such as the white list provided by 10X. Data from non-UMI protocols such as Smart-seq and Smart-seq2 can be handled by setting `has_UMI` to `FALSE` when running `sc_demultiplex`. The `sc_detect_bc` function summarises the sequences in the cell barcode region, collapsing reads with a

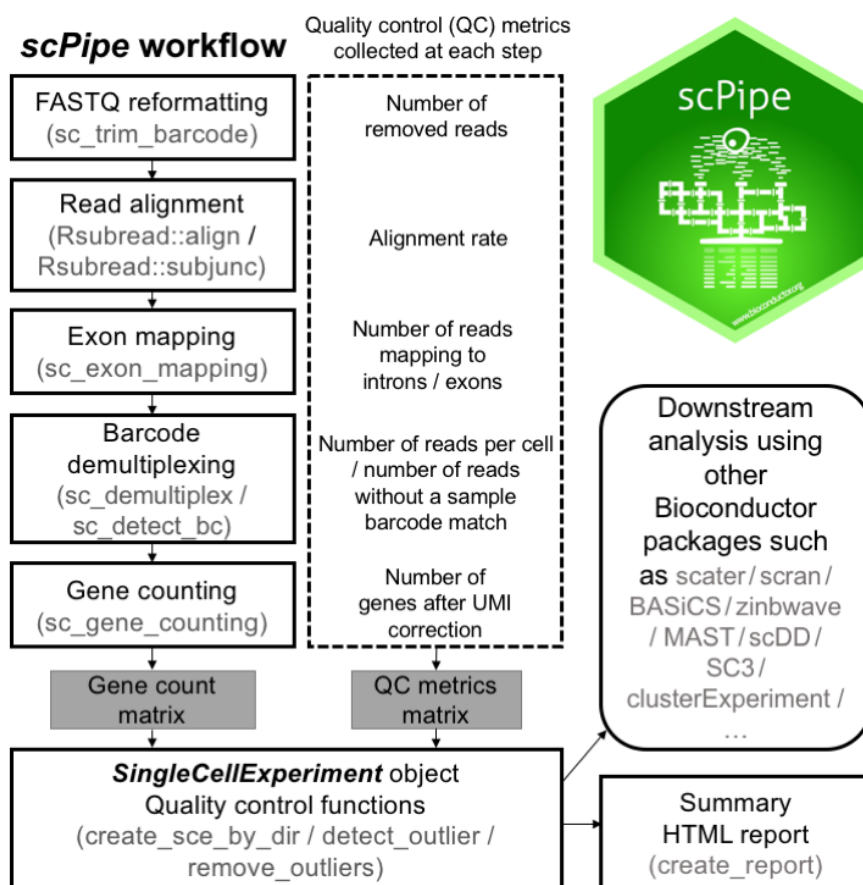


Fig 1. The *scPipe* workflow. *scPipe* is a R/Bioconductor package that uses functionality from a number of other packages, including *Rsubread* to align reads to a reference genome (although in practice any aligner that produces BAM files can be used for read alignment) and *SingleCellExperiment* to organise the counts and sample annotation information. The major steps in the preprocessing pipeline of *scPipe* are shown along with the quality control (QC) statistics collected at each stage. The final output of this process is a matrix of counts and QC metrics for use in downstream analysis and an HTML report that summarises the analysis. The resulting *SingleCellExperiment* object can be used as input to other Bioconductor packages to perform further downstream analysis.

small edit distance, which are indicative of sequencing errors. It can report a given number of cell barcodes or keep the cell barcodes based on a read number cutoff. A relaxed threshold is generally recommended when one specifies the number of barcodes to retain, for instance if the estimated number of cells in the library is 1,000-5,000, a threshold of 10,000 would ensure data for all cells is collected. The motivation behind this is that unlike *CellRanger*, we don't recommend detecting low quality cells or empty barcodes solely based on the number of reads or UMI counts per cell. Instead, more robust and meaningful results can be produced using multiple QC metrics. Hence the `sc_detect_bc` function should be run to allow more barcodes than cells. The quality control function in *scPipe* can be used to remove erroneous cell barcodes at a later stage. The demultiplexed data are output in a csv file for each cell, where each row corresponds to a read that maps to a specific gene, and columns that include gene id, UMI sequence and mapping position. The overall barcode demultiplexing results are recorded and can be plotted as shown in Figure 2A for the mouse blood cells generated by the CEL-seq2 protocol.

Obtaining a gene count matrix and performing quality assessment

The next stage of preprocessing makes use of the `sc_gene_counting` function to remove PCR duplicates (also known as UMI deduplication) to generate a gene count matrix. *scPipe* uses a greedy approach based on hamming distance to do this. The detected UMIs are compared to each other and if two UMI sequences are found to be within a certain hamming distance apart (the default is 1) and one UMI has more than twice the count of the other, then the two different UMIs will be treated as instances of the same UMI. The approach is based on the fact that erroneous UMI sequences due to sequencing error usually just have one or two reads. PCR errors, however, can be more problematic, and other more sophisticated network-based methods such as the approach taken in *UMI-tools* have been developed to deal with such errors. When PCR error rates are low, the approach used in *scPipe* should give good results. Our pipeline outputs the raw UMI sequences that map to each gene, which can be used as input for more sophisticated UMI correction methods when the PCR error rate is high. After UMI deduplication we get a gene count matrix that can be used for further analysis. QC information is collected at each step (Figure 1) and includes the total number of mapped reads per cell, UMI deduplication statistics, per cell barcode demultiplexing statistics, UMI correction statistics and External RNA Controls Consortium (ERCC) spike-in control statistics (where present).

scPipe uses the S4 infrastructure provided by the *SingleCellExperiment* package [20] for data storage. A *SingleCellExperiment* object can be constructed by the `create_sce_by_dir` function using the data folder generated during preprocessing, or by manually specifying all the required slots. Alongside the gene count matrix, this object stores QC information obtained during preprocessing, and the type of gene id and organism name, which can both be useful in downstream functional enrichment analysis.

Next, alignment statistics for each cell can be plotted using the `plotMapping` function (Figure 2B). Data shown here is again from the mouse blood cell dataset described previously. Monitoring these QC statistics across experiments (Figures 2C-D) can be particularly useful for labs that routinely process single-cell data, allowing them to assess the impact changes in lab processes and protocols have on data quality. Pairwise scatter plots of QC metrics such as the total molecules per cell and the number of genes detected can be generated with the `plotQC_pair` function (Figure 3).

scPipe implements a multivariate outlier detection method for discovering low quality cells to remove from further analysis. The method uses up to 5 metrics (log-transformed number of molecules detected, the number of genes detected, the percentage of reads mapping to ribosomal, or mitochondrial genes and ERCC recovery (where available)) as input (Figure 3) to a Gaussian mixture model (GMM). There are three stages in the outlier detection process. First the Mahalanobis distances between cells based on the quality control metrics chosen are calculated, with extreme cells removed before fitting the GMM using the *mclust* package [21]. The QC metrics of the remaining cells are used to capture the overall heterogeneity in the dataset. The mixture component with systematically lower QC metrics, such as lower numbers of genes detected and lower numbers of molecular counts are marked as outliers. In the final stage, the Mahalanobis distance is calculated for cells from the mixture component with the highest QC metric values with outliers automatically detected based on this distance. Outlier detection can then be reconciled with visual inspection of the QC metrics through the afore-mentioned QC plotting options to fine tune the sample-specific filtering thresholds chosen. Depending on the data, the only argument that needs to be specified is the maximum number of components in the GMM. In most cases, this is set to 1 or 2 for good quality data with only a small proportion of poor quality cells. For data generated by droplet-based protocols where larger proportions of poor quality cells may be expected, the number of components can be set to 2 or 3 to model extra components that correspond to the good and poor quality cells. Apart from the number of components, the user can also adjust the confidence interval for choosing outliers (0.99 by default).

An HTML report generated using *rmarkdown* [14] that includes all run parameters used, QC statistics and various types of dimension reduction plots of both the gene expression data and QC metrics can be generated using the `create_report` function (Figure 4). As *scPipe* generates consistent QC measures across different protocols and experiments, these QC metrics can be easily combined (Figures 2C,2D) to facilitate comparisons between multiple datasets.

After QC, the data can be easily passed to downstream packages. Use of the *SingleCellExperiment*

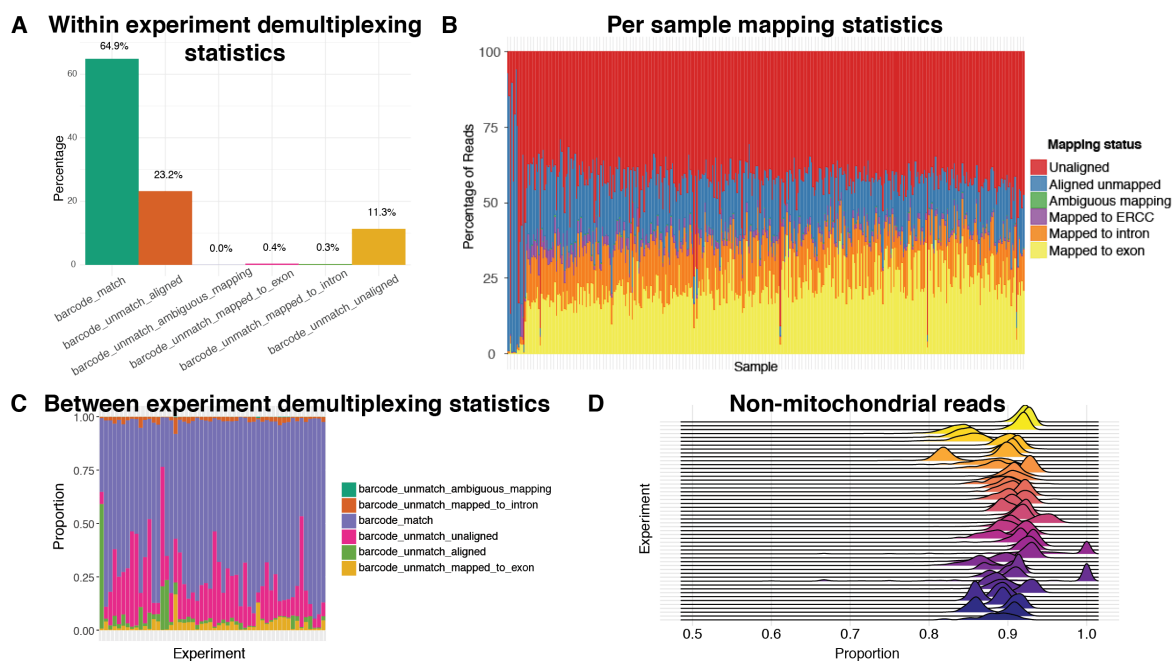


Fig 2. Example QC plots that can be created using output from *scPipe* to assess data quality both within and between experiments. Within experiment displays include (A) a bar plot illustrating the overall cell barcode matching results to assess sequencing accuracy across all samples and (B) a stacked bar plot showing the mapping rate, separated into reads that map to exon, intron and ERCCs and those that are ambiguously mapped, map elsewhere in the genome and are unaligned for each cell in an experiment (ordered by exon mapping rate). Between experiment displays include (C) a stacked bar plot showing the cell barcode matching results from panel (A) from multiple experiments and (D) a ridgeline plot presenting the distribution of proportions of non-mitochondrial read counts for cells across multiple experiments.

class ensures the output of *scPipe* is fully interoperable with a range of other Bioconductor packages. For instance, packages such as *scater*, *scraper* and *BASiCS* [22] can be used for normalization, *zinbwave* [23] for dealing with zero inflation in the data and removing unwanted variation, *SC3* [24] and *clusterExperiment* [25] for clustering and *MAST* [26] and *scDD* [27] for differential expression analysis.

Using *scPipe*

A vignette accompanying the package provides further details on implementation and an example use case on the CEL-seq2 mouse blood dataset.

This example dataset contains 100 million reads and takes under 2 hours to process on a standard Linux server. *scPipe* creates a *SingleCellExperiment* object from the count matrix and mapping statistics output by the pipeline which can be manipulated using other Bioconductor packages for single cell analysis. *scPipe*'s outlier detection has been used to remove low quality cells (Figure 3). For the mouse blood dataset, *scPipe*'s outlier removal retained 359 of 383 cells for further analysis and in the human spleen dataset, 1,397 of the 2,273 detected cells were retained for further analysis. We applied *scraper* to compute size factors for normalization. Differential expression analysis was performed between the *a priori* defined B-lymphocyte and T-lymphocyte cells (based on FACS sorting information available for the cells in each well) using a generalised linear model from *edgeR* [28] after performing conversion to a *DGEList* using *scater* (Figure 5A). From this analysis 636 differentially expressed genes were discovered based on likelihood-ratio testing (LRT) at a 0.05 false-discovery-rate (FDR) threshold.

A second example uses data from the pilot Human Cell Atlas ischaemic sensitivity dataset, which was

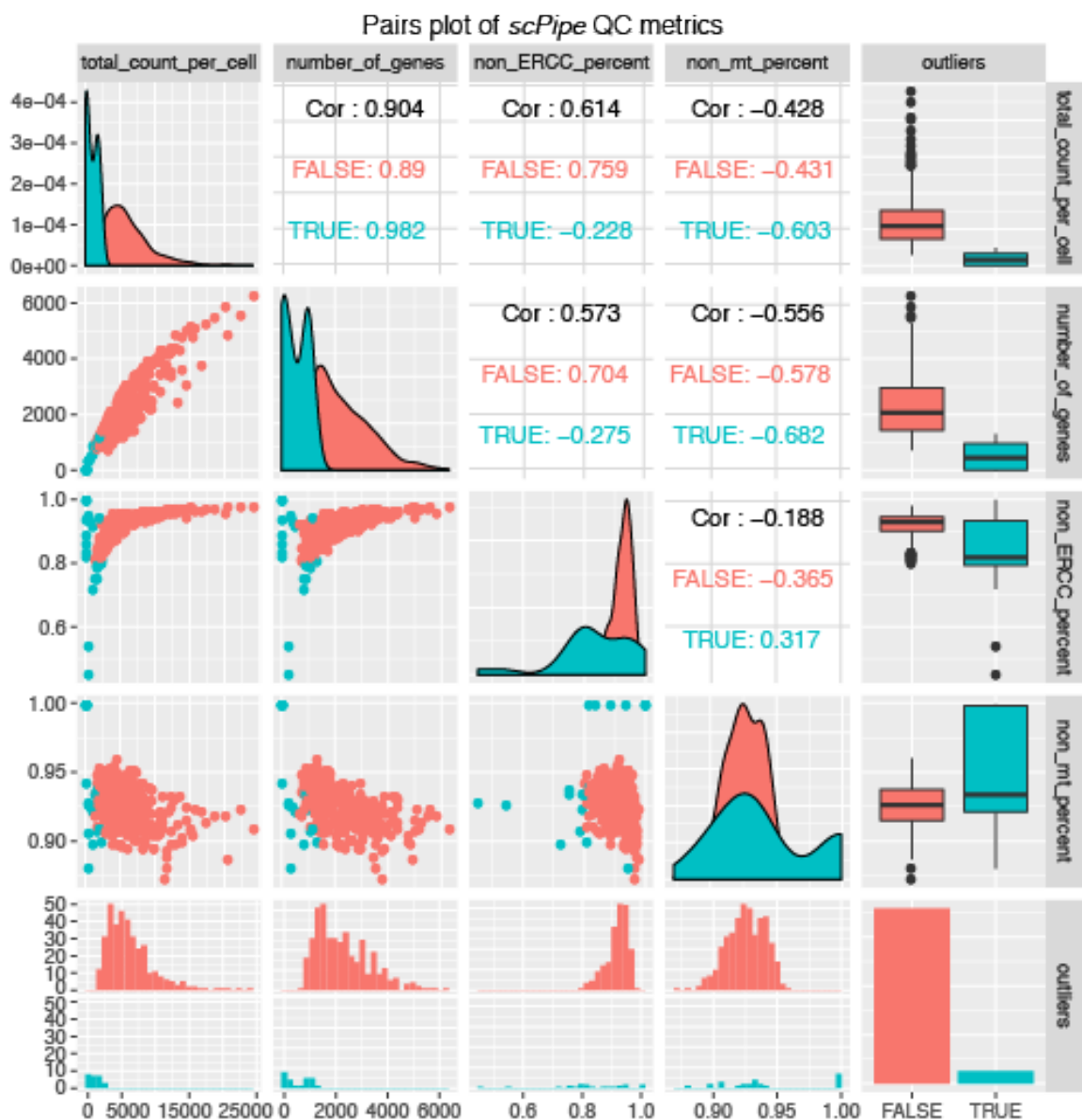


Fig 3. A pairwise scatter plot of the quality control metrics collected by *scPipe* for the mouse blood cell dataset. Sample-specific metrics include the total read count, the number of genes detected and the proportion of non-ERCC or non-mitochondrial reads. The good quality and outlier samples detected by *scPipe*'s automatic outlier detection method are indicated in each panel by a different colour.

again preprocessed by *scPipe*, with automatic outlier detection leaving 1,397 cells, which were then normalized using *scan*. Next, we performed cell clustering with *SC3*, where the number of clusters produced was optimised by *SC3*'s `estimate_k` function to give 10 clusters. To annotate these clusters, the mean normalized expression of each cluster was correlated with a microarray dataset of 12 immunological lineages [29]. The lineage with the highest correlation was considered the most likely cell type candidate for each cluster. Using this method, the 10 clusters identified by *SC3* were assigned to 5 of the 12 potential candidate lineages (Figure 5B).

1. parameters for each preprocessing step
1.1 parameters for <code>sc_trim_barcode</code>
1.2 parameters for alignment
1.3 parameters for <code>sc_exon_mapping</code>
1.4 parameters for <code>sc_demultiplex</code>
1.5 parameters for <code>sc_gene_counting</code>
2. data summary
quality control
data normalization
dimensionality reduction using high variable genes
get FACS data

scPipe report for sample CB51

Code ▾

1. parameters for each preprocessing step

1.1 parameters for `sc_trim_barcode`

file path

- input fastq1: /wehisan/general/user_managed/grpu_naik.s_2/2017 Sequencing Runs/CB51/christineb160517/Undetermined_S0_R2.fastq.gz
- input fastq2: /wehisan/general/user_managed/grpu_naik.s_2/2017 Sequencing Runs/CB51/christineb160517/Undetermined_S0_R1.fastq.gz
- output fastq: /wehisan/general/user_managed/grpu_naik.s_1/CB51_24.05.17/CB51.fq

read structure

assume read1 contains the transcript

- barcode in read1: NA
- barcode in read2: BC2_INFO__
- UMI in read2: start at position 0, length 6

read filter

- remove reads that have N in its barcode or UMI: TRUE
- remove reads with low quality: TRUE

1.2 parameters for alignment

- input fastq: /wehisan/general/user_managed/grpu_naik.s_1/CB51_24.05.17/CB51.fq
- output bam file: /wehisan/general/user_managed/grpu_naik.s_1/CB51_24.05.17/CB51.aligned.bam
- genome index: /home/users/allstaff/tian.I/MM.GRCm38/subRead_MM_GRCm38_ERCC_phiX_WPRE

1.3 parameters for `sc_exon_mapping`

- input bam file: /wehisan/general/user_managed/grpu_naik.s_1/CB51_24.05.17/CB51.aligned.bam
- output bam file: /wehisan/general/user_managed/grpu_naik.s_1/CB51_24.05.17/CB51.aligned.mapped.bam

Fig 4. Screenshot of an HTML report created by *scPipe* for the mouse blood cell dataset. This report organises the output of *scPipe*, including run parameters and QC metrics and also generates basic dimension reduction plots of the data. Such reports provide a convenient format for communicating basic QC information to collaborators to help them evaluate the overall quality of an experiment.

A third example of running *scPipe* on the Chromium 10X cell line data is, which has 400 million reads from around 1,000 cells takes about 10 hours on a standard Linux server. We also provide examples of using *scPipe* with an alternate aligner (STAR [30]).

Code for each of these example analyses is provided as Supplementary Information from <http://bioinf.wehi.edu.au/scPipe/>.

scPipe is a fully modular pipeline, with each step generating independent results. It is also flexible with its inputs, accepting FASTQ as outlined above, or BAM files produced by *CellRanger* and Dropseq tools (<http://mccarrolllab.com/dropseq/>) from which a gene count matrix can be generated. This approach ensures interoperability with tools both inside and outside of Bioconductor to give the data analyst maximum flexibility when configuring their scRNA-seq analysis pipelines.

Comparing *scPipe* with other pipelines

We reviewed the tools currently available for scRNA-seq data preprocessing identified through the ‘Alignment’ and ‘UMIs’ categories provided by the <https://www.scrna-tools.org/> website [4]. The search revealed 9 tools that have overlapping functionality with *scPipe*. Our software is the only Bioconductor package that spans all major single cell platforms (both plate and droplet-based) and the full range of preprocessing tasks (alignment, UMI handling and QC). The shell program *zUMIs* [31] is another program that covers the breadth of platforms and preprocessing tasks. Other software tools, such as *dropSeqPipe*, *CellRanger* and *dropEst* [8] only deal with data from droplet-based protocols, *Sharq* [32] is suitable for plate-based approaches, and the remaining handle the UMI processing step only. *CellRanger*

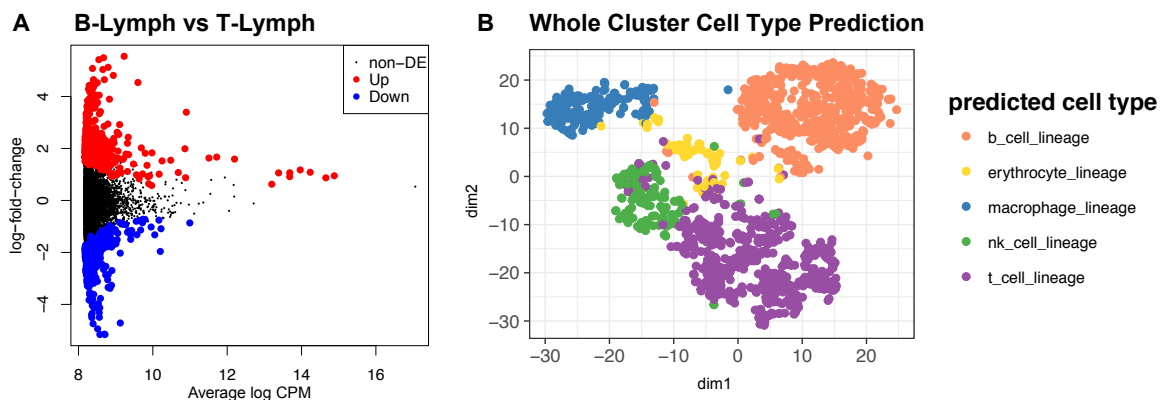


Fig 5. Analysis results produced with *SingleCellExperiment*-compatible analysis packages from Bioconductor. (A) Differential gene expression results from comparing B-lymphocytes and T-lymphocytes known *a priori* in the CEL-seq2 mouse blood dataset. Highlighted points are genes determined to be significantly differentially expressed from LRT with 0.05 FDR cutoff. (B) tSNE coloured by cell type predictions obtained for *SC3* clusters for the first spleen sample from the Human Cell Atlas ischaemic sensitivity dataset. The 10 clusters from *SC3* were assigned to 5 distinct cell types from the reference set of 12 lineages.

is the only tool that also offers downstream analysis, with clustering and differential expression testing included in its default output.

Software	Language	Input	Multiple technologies	Analysis task			
				Alignment	UMI handling	Quality control	Downstream analysis
scPipe	R	FASTQ	✓	✓	✓	✓	✗
zUMIs	Shell	FASTQ	✓	✓	✓	✓	✗
dropSeqPipe	Snakemake	FASTQ	✗	✓	✓	✓	✗
Sharq	Python/Perl/R	FASTQ	✗	✓	✓	✓	✗
CellRanger	Python/R	BCL / FASTQ	✗	✓	✓	✓	✓
dropEst	C++/R	FASTQ	✗	✓	✓	✓	✗
UMI-tools	Python	FASTQ	✓	✗	✓	✗	✗
umis	Python	FASTQ	✓	✗	✓	✗	✗
sircel	Python	FASTQ	✓	✗	✓	✗	✗
TRUmiCount	R/Shell	BAM	✓	✗	✓	✗	✗

Table 1. Summary of the data preprocessing software currently available for scRNA-seq analysis and the particular tasks covered by each package.

In order to highlight the differences between *scPipe* and the popular *CellRanger* pipeline that processes raw data from the Chromium 10X platform, we performed a benchmark experiment that included 3 human lung adenocarcinoma cell lines (see Materials and Methods) and processed the raw data using both *scPipe* and *CellRanger* (Figure 6A). The dataset contains about 1,000 cells. *CellRanger* returns 1,027 cells and *scPipe* 981 cells after QC (Figure 6B). The t-SNE plot [33] generated from the *CellRanger* output shows that the 48 cells that appear in the *CellRanger* results but not in *scPipe* tend to cluster together (Figure 6C). Inspection of their QC metrics (Figure 6D) shows that these cells have higher proportions of mitochondrial gene counts, suggesting they may be dead cells that should be

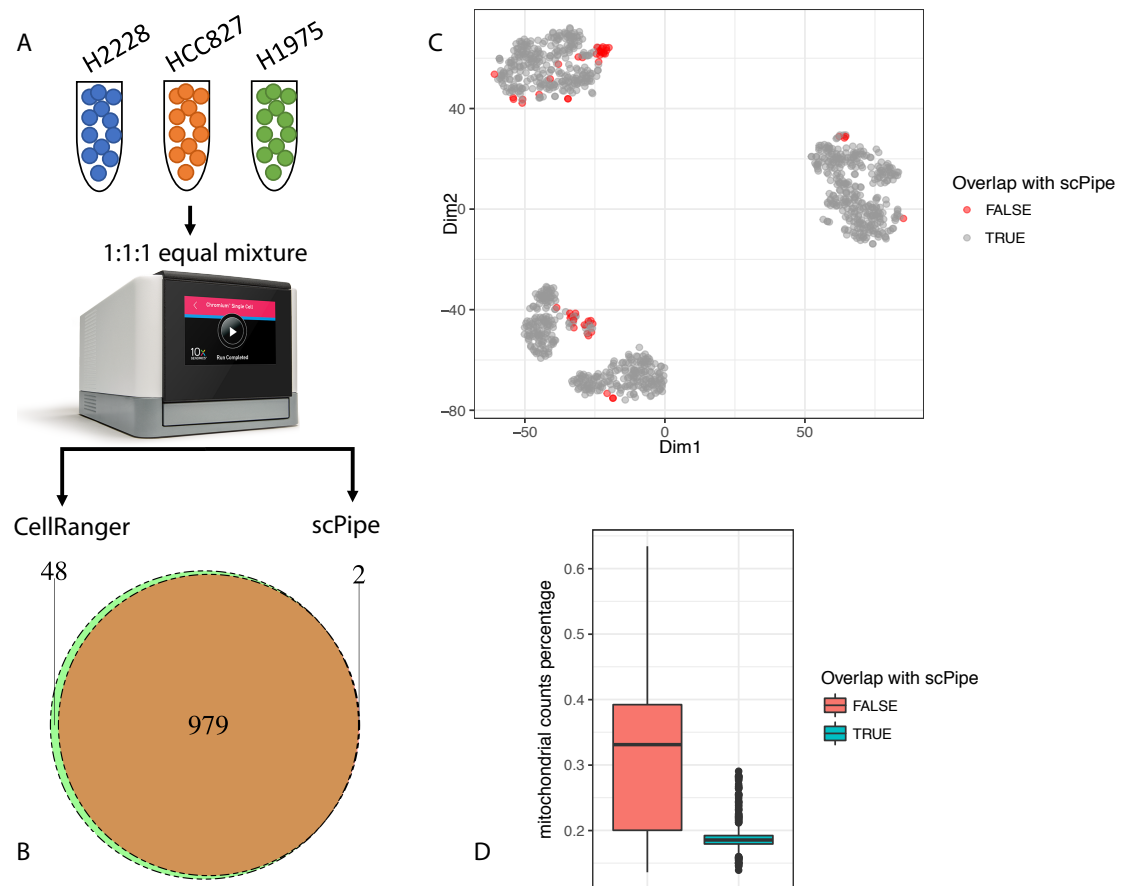


Fig 6. Comparing *scPipe* and *Cell Ranger*. (A) The workflow for the comparison. An equal mixture of cells from three cell lines are sequenced using the Chromium 10X platform (see Materials and Methods). Data were processed by *scPipe* and *Cell Ranger*. (B) The pie chart shows the overlap of cell barcode detected by *scPipe* and *Cell Ranger*. (C) The t-SNE plot generated using *Cell Ranger* output. Cell barcodes that only exist in *Cell Ranger* are highlighted. (D) Box plots showing the percentage of mitochondrial gene counts in cells that overlap with *scPipe* or only exist in the *Cell Ranger* results.

excluded from downstream analysis. Since *Cell Ranger* only uses the UMI counts per cell as a QC cutoff, the results generated by *Cell Ranger* may contain dead cells and benefit from a further round of QC. The *scPipe* analysis on the other hand uses multiple QC metrics by default (Figure 3) to achieve a robust measure of cell quality to ensure low quality cells are discarded. This comparison shows the benefit of *scPipe*'s built-in QC step.

Availability and Future Directions

The *scPipe* package is available from <https://www.bioconductor.org/packages/scPipe>. Code for each of the example analyses described in the 'Using *scPipe*' section above is available from <http://bioinf.wehi.edu.au/scPipe/>. Bug reports and questions about using *scPipe* should be posted on the Bioconductor support site (<https://support.bioconductor.org/>).

With the growing popularity of scRNA-seq technology, many tools have been developed for normalization, dimensionality reduction and clustering. There are relatively few packages designed to handle the raw data obtained from the various 3' end sequencing protocols with their associated UMIs

and cell-specific barcodes from beginning to end and collect detailed quality control information. The *scPipe* package bridges this gap between the raw FASTQ files with mixed barcode types and transcript sequences and the gene count matrix that is the entry point for all downstream analyses. *scPipe* outputs numerous QC metrics obtained at each preprocessing step and displays these results in an HTML report to assist end users in QC evaluations. Future improvements that are planned for *scPipe* include support for new scRNA-seq protocols as they emerge and parallelization of the various preprocessing steps to enable scalability to larger datasets. We also plan to generate a more comprehensive scRNA-seq benchmark dataset to ensure the default UMI correction and quality control methods used in *scPipe* are optimal that will also allow for a more detailed comparison of *scPipe* with other relevant analysis pipelines, such as *zUMIs*. The *scPipe* package is used extensively in the following chapters, ensuring a consistent data preprocessing pipeline across multiple scRNAseq protocols in the benchmarking work of Chapter 3 and the preprocessing of data from single-cell long-read protocols in Chapter 4.

Acknowledgments

The authors are grateful to Saskia Freytag, Carolyn de Graaf, Aaron Lun, Rowland Mosbergen, Othmar Korn and Valerie Obenchain for their advice and feedback on the *scPipe* software, Roberto Bonelli for designing the *scPipe* logo and Clare Weeden and Marie-Liesse Asselin-Labat for providing the cell lines used in the Chromium 10X experiment.

Author Contributions

L.T. developed the software, performed analysis and wrote the manuscript. S.S. and X.D. contributed to software development and data analysis. D.A.-Z., C.B., A.S., D.J.H. and S.H.N. performed or supervised data generation. S.H.N. and M.E.R. supervised the research and M.E.R. wrote the manuscript. All authors read and approved the final manuscript.

References

1. Patel A, Tirosh I, Trombetta J, Shalek A, Gillespie S, Wakimoto H, et al. Single-cell RNA-seq highlights intratumoral heterogeneity in primary glioblastoma. *Science*. 2014;344(6190):1396–1401. doi:10.1126/science.1254257.
2. Macosko E, Basu A, Satija R, Nemesh J, and M Goldman KS, Tirosh I, et al. Highly Parallel Genome-wide Expression Profiling of Individual Cells Using Nanoliter Droplets. *Cell*. 2015;161(5):1202–1214. doi:10.1016/j.cell.2015.05.002.
3. Trapnell C, Cacchiarelli D, Grimsby J, Pokharel P, Li S, Morse M, et al. The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nat Biotechnol*. 2014;32(4):381–386. doi:10.1038/nbt.2859.
4. Zappia L, Phipson B, Oshlack A. Exploring the single-cell RNA-seq analysis landscape with the scRNA-tools database. *bioRxiv*. 2017;doi:https://doi.org/10.1101/206573.
5. Smith T, Heger A, Sudbery I. UMI-tools: modeling sequencing errors in Unique Molecular Identifiers to improve quantification accuracy. *Genome Res*. 2017;27:491–99.
6. Svensson V, et al. Power analysis of single-cell RNA-sequencing experiments. *Nat Methods*. 2017;14:381–87.
7. Zheng GXY, Terry JM, Belgrader P, Ryvkin P, Bent ZW, Wilson R, et al. Massively parallel digital transcriptional profiling of single cells. *Nat Commun*. 2017;8:14049. doi:doi:10.1038/ncomms14049.

8. Petukhov V, Guo J, Baryawno N, Severe N, Scadden D, Kharchenko PV. Accurate estimation of molecular counts in droplet-based single-cell RNA-seq experiments. *bioRxiv*. 2017;(August 2017). doi:10.1101/171496.
9. McCarthy DJ, et al. Scater: pre-processing, quality control, normalisation and visualisation of single-cell RNA-seq data in R. *Bioinformatics*. 2017;33:1179–86.
10. Lun ATL, McCarthy DJ, Marioni JC. A step-by-step workflow for low-level analysis of single-cell RNA-seq data with Bioconductor. *F1000Res*. 2016;5:2122.
11. Hashimshony T, Senderovich N, Avital G, Klochendler A, de Leeuw Y, Anavy L, et al. CEL-Seq2: sensitive highly-multiplexed single-cell RNA-Seq. *Genome Biol*. 2016;17:77.
12. R Core Team. R: A Language and Environment for Statistical Computing; 2017. Available from: <https://www.R-project.org/>.
13. Huber W, Carey VJ, Gentleman R, Anders S, Carlson M, Carvalho BS, et al. Orchestrating high-throughput genomic analysis with Bioconductor. *Nature Methods*. 2015;12(2):115–21.
14. Allaire J, Xie Y, McPherson J, Luraschi J, Ushey K, Atkins A, et al.. rmarkdown: Dynamic Documents for R; 2017. Available from: <https://CRAN.R-project.org/package=rmarkdown>.
15. Eddelbuettel D, François R. Rcpp: Seamless R and C++ Integration. *Journal of Statistical Software*. 2011;40(8):1–18.
16. Eddelbuettel D. Seamless R and C++ Integration with Rcpp. New York: Springer; 2013.
17. Hayden N, Morgan M. Rhtslib: HTSlib high-throughput sequencing library as an R package; 2017.
18. Liao Y, Smyth GK, Shi W. The Subread aligner: fast, accurate and scalable read mapping by seed-and-vote. *Nucleic Acids Res*. 2013;41:e108.
19. Liao Y, Smyth GK, Shi W. featureCounts: an efficient general-purpose program for assigning sequence reads to genomic features. *Bioinformatics*. 2014;30(7):923–30.
20. Lun A, Risso D. SingleCellExperiment: S4 Classes for Single Cell Data; 2017.
21. Scrucca L, Fop M, Murphy TB, Raftery AE. mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. *The R Journal*. 2016;8(1):205–233.
22. Vallejos CA, Marioni JC, Richardson S. BASiCS: Bayesian Analysis of Single-Cell Sequencing Data. *PLoS Comput Biol*. 2015;11:e1004333.
23. Risso D, Perraudeau F, Gribkova S, Dudoit S, Vert JP. A general and flexible method for signal extraction from single-cell RNA-seq data. *Nat Commun*. 2018;9:284. doi:10.1038/s41467-017-02554-5.
24. Kiselev VY, Kirschner K, Schaub MT, Andrews T, Yiu A, Chandra T, et al. SC3: consensus clustering of single-cell RNA-seq data. *Nat Methods*. 2017;14(5):483–486. doi:10.1038/nmeth.4236.
25. Risso D, Purvis L, Fletcher R, Das D, Ngai J, Dudoit S, et al. clusterExperiment and RSEC: A Bioconductor package and framework for clustering of single-cell and other large gene expression datasets. *bioRxiv*. 2018;doi:10.1101/280545.
26. Finak G, McDavid A, Yajima M, Deng J, Gersuk V, Shalek A, et al. MAST: A Flexible Statistical Framework for Assessing Transcriptional Changes and Characterizing Heterogeneity in Single-Cell RNA Sequencing Data. *Genome Biol*. 2015;16:278.
27. Korthauer K, Chu L, Newton M, Yuan L, Thomson J, Stewart R, et al. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. *Genome Biol*. 2016;17:222.

28. McCarthy DJ, Chen Y, Smyth GK. Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation. *Nucleic acids research*. 2012;40(10):4288–4297.
29. Novershtern N, Subramanian A, Lawton LN, Mak RH, Haining WN, McConkey ME, et al. Densely interconnected transcriptional circuits control cell states in human hematopoiesis. *Cell*. 2011;144(2):296–309.
30. Dobin A, Davis CA, Schlesinger F, Drenkow J, Zaleski C, Jha S, et al. STAR: ultrafast universal RNA-seq aligner. *Bioinformatics*. 2013;29:15–21.
31. Parekh S, Ziegenhain C, Vieth B, Enard W, Hellmann I. zUMIs: A fast and flexible pipeline to process RNA sequencing data with UMIs. *bioRxiv*. 2018;doi:10.1101/153940.
32. Candelli T, Lijnzaad P, Muraro MJ, Kerstens H, Kemmeren P, van Oudenaarden A, et al. Sharq, a versatile preprocessing and QC pipeline for Single Cell RNA-seq. *bioRxiv*. 2018;doi:10.1101/250811.
33. van der Maaten LJP, Hinton GE. Visualizing High-Dimensional Data Using t-SNE. *Journal of Machine Learning Research*. 2008;9:2579–2605.

Chapter 3

Benchmarking single cell RNA-sequencing analysis pipelines using mixture control experiments

Benchmarking single cell RNA-sequencing analysis pipelines using mixture control experiments

Luyi Tian^{1,2*}, Xueyi Dong^{1,3}, Saskia Freytag^{1,4}, Kim-Anh Lê Cao⁵, Shian Su¹, Abolfazl JalalAbadi⁵, Daniela Amann-Zalcenstein^{1,2}, Tom S. Weber^{1,2}, Azadeh Seidi⁶, Jafar S. Jabbari⁶, Shalin H. Naik^{1,2}, Matthew E. Ritchie^{1,2*}

1 The Walter and Eliza Hall Institute of Medical Research, Parkville, Victoria, Australia.

2 Department of Medical Biology, The University of Melbourne, Parkville, Victoria, Australia.

3 College of Life Science, Zhejiang University, Hangzhou, Zhejiang Province, P.R. China.

4 Harry Perkins Institute of Medical Research, Nedlands, Western Australia, Australia.

5 Melbourne Integrative Genomics, School of Mathematics and Statistics, The University of Melbourne, Parkville, Victoria, Australia.

6 Australian Genome Research Facility, Victorian Comprehensive Cancer Centre, Melbourne, Victoria, Australia.

Abstract

Single cell RNA-sequencing (scRNA-seq) technology has undergone rapid development in recent years, leading to an explosion in the number of tailored data analysis methods. However, the current lack of gold-standard benchmark datasets makes it difficult for researchers to systematically compare the performance of the many methods available. Here, we generated a realistic benchmark experiment that included single cells and admixtures of cells or RNA to create ‘*pseudo cells*’ from up to 5 distinct cancer cell lines. In total 14 datasets were generated using both droplet and plate-based scRNA-seq protocols. We compared 3,913 combinations of data analysis methods for tasks ranging from normalization and imputation, to clustering, trajectory analysis and data integration. Evaluation revealed pipelines suited to different types of data for different tasks. Our data and analysis provides a comprehensive framework for benchmarking most common scRNA-seq analysis steps.

Introduction

The rapid development of computational methods for single cell gene expression analysis has created a need for systematic benchmarking in order to understand the strengths and weaknesses of different algorithms. The performance of particular scRNA-seq analysis methods have been evaluated for tasks including normalization [1], feature selection [2], differential gene expression analysis [3], clustering [4, 5] and trajectory analysis [6]. These studies compare methods using either experimental data where cell type labels are available or simulated datasets. Such ground truth is however imperfect, with simulations relying on assumptions that may not reflect the true nature of scRNA-seq data, and by focusing on specific tasks, these studies lack a complete picture of performance at the pipeline level.

Considering the heterogeneity between scRNA-seq datasets in terms of the number of clusters (cell types/states) and the presence of various technical artifacts, we set out to design a realistic gold-standard scRNA-seq control experiment that combines ground truth with varying levels of biological complexity. Two strategies are commonly employed to create gold-standard gene expression datasets. The first, that has been widely adopted in scRNA-seq studies [7], uses small collections of exogenous spike-in controls such as ERCCs [8] that vary in expression in a predictable way. The second involves either the dilution of RNA from a reference sample or mixing of RNA or cells from two or more samples to induce systematic genome-wide changes. An early example from Brennecke *et al.* [9] involved a dilution series to explore sensitivity of the Smart-seq protocol. Grün *et al.* [10] generated a benchmark dataset using single mouse embryonic stem cells (mESC) together with bulk RNA extracted from the same population, diluted to single cell equivalent amounts to quantify biological and technical variability. A limitation of these experiments is their lack of biological heterogeneity which makes them less useful for comparing analysis methods. Mixture designs, in which RNA or cells are mixed in different proportions to generate biological heterogeneity with in-built truth have been successfully used to benchmark microarray [11], RNA-seq [12] and scRNA-seq [13] data.

To combine the strengths of these approaches, we designed a series of experiments using mixtures of either cells or RNA from up to 5 cancer cell lines that included a dilution series to simulate variations in the RNA content of different cells as well as ERCC spike-in controls where possible. Data were generated across four single-cell technology platforms. Our *scRNA-seq mixology* design simulates varying levels of biological noise, with sample sizes varying from around 200 cells to 4,000 cells and known population structure to allow benchmarking of different analysis tools. We specifically evaluated combinations of methods for normalization and imputation, clustering, trajectory analysis and data integration. The methods chosen for each task use different algorithms and are mostly implemented in R [14] and Bioconductor [15] for convenience.

Results

scRNA-seq mixology provides ground truth for benchmarking

Our scRNA-seq benchmarking experiment spanned 2 plate-based (CEL-seq2 and SORT-seq) and 2 droplet-based (10X Chromium and Drop-seq) protocols and involved 3 different experimental designs with replicates, yielding 14 datasets in total (Supplementary Table 1). The experimental design involved single cells, mixtures of single cells and mixtures of RNA from up to 5 human lung adenocarcinoma cell lines (see Methods, Figure 1A-B and Supplementary Figure 1).

The three designs incorporate ground truth in various ways. For the *single cell* datasets (Figure 1A), the ground truth is the cell line identity which can be determined for each cell based on known genetic variation (see Methods). For the *'pseudo cell'* mixture datasets, the known composition of cells/RNA serves as ground truth. The *RNA mixture* and *cell mixture* datasets (Figure 1B and Supplementary Figure 1) contain 7 and 34 groups respectively that give a continuous structure. Moreover, the *RNA mixture* dataset contains technical replication and a dilution series (Figure 1B), which is ideal for benchmarking normalization and imputation methods that are intended to deal with such technical variability. The data characteristics and analysis tasks each design is suited to benchmark are

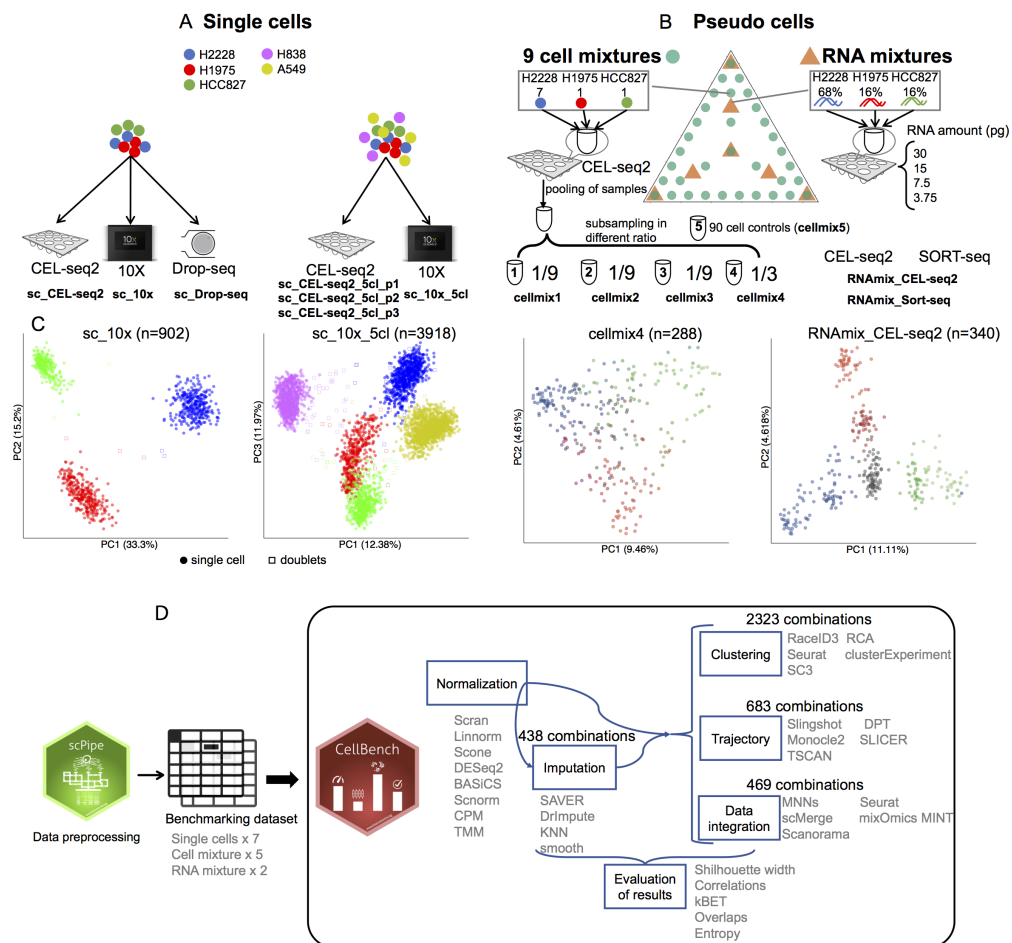


Figure 1. Overview of our scRNA-seq mixology experimental design. The benchmark experimental design involving single cells (A) and ‘pseudo cells’ (B). (C) PCA plots from representative datasets for each design (normalized using *scraper*) highlight the structure present in each experiment. Percentage variation explained by each PC is included in the respective axis labels and sample sizes are indicated by *n*. (D) Workflow for benchmarking different analysis tasks using the *CellBench* R package.

summarised in Supplementary Table 2.

By comparing a range of quality control metrics collected across datasets using *scPipe* [16], we observed that data from all platforms were of consistently high quality in terms of their exon mapping rates and the total unique molecular identifier (UMI) counts per cell (Supplementary Figure 2). Interestingly, we found substantial differences in the percentage of reads mapping to intron regions in datasets generated from different protocols and experimental designs (Supplementary Figure 2A). After normalization by *scrna* [17], Principal Component Analysis (PCA) plots from four representative datasets show that our single cell and mixture designs successfully recapitulate the expected population structure (Figure 1C; t-SNE and UMAP visualizations provided in Supplementary Figure 3A). Doublet rates were observed to vary between datasets (Supplementary Figure 3B).

A summary of the benchmarking workflow is shown in Figure 1D. After preprocessing and quality control with *scPipe*, each dataset was analysed using different combinations of normalization (8 methods) and imputation (3 methods). Each normalized and imputed gene expression matrix was used as input to various downstream steps, including clustering (7 methods), trajectory analysis (5 methods) and data integration (6 methods). The *CellBench* R package (see Methods) was developed to manage the 3,913 dataset $\times$ method combinations obtained through this analysis, with performance evaluated using metrics tailored to the ground truth available.

Comparisons of normalization and imputation methods

We evaluated 8 popular normalization methods, including techniques developed primarily for bulk RNA-seq such as *TMM* [18], *CPM* [19] and *DESeq2* [20], and others tailored for scRNA-seq, including *scone* [1], *BASiCS* [21], *SCnorm* [22], *Linnorm* [23] and *scrna* [17]. Three scRNA-seq imputation methods, including *kNN-smoothing* [24], *DrImpute* [25] and *SAVER* [26] were also evaluated using input data normalized by different methods. Altogether, 438 analyses representing combinations of datasets, normalization and imputation methods were used for benchmarking.

Performance was evaluated using 2 metrics: the silhouette width of clusters for all datasets, and the Pearson correlation coefficient of normalized gene expression within each group for the *RNA mixture* data (see Methods). Although a wide range of silhouette widths was observed amongst different methods in different datasets, most normalization methods, apart from *TMM*, markedly increased silhouette width relative to the unnormalized data (Figure 2A, Supplementary Figure 4A-B). Methods tailored to scRNA-seq data tended to perform better than methods designed for bulk RNA-seq analysis, with the exception of *DESeq2*, which generated good results. Across all the methods compared, *Linnorm* was the top performer on average, followed by *scrna* and *scone*.

The *RNA mixture* experiment included pseudo cells with varying amounts of input RNA to simulate dropout events, making it an ideal dataset for evaluating the performance of imputation methods. Example PCA plots on an RNA mixture dataset after different normalization combinations are shown in Figure 2B-E. In general, imputation induces higher intra-group correlation, although considerable differences are observed depending on the normalization method chosen (Figure 2F). *kNN-smoothing* and *DrImpute* show similar results with different normalization methods. *SAVER* has greatest variation in performance, with either the best or worst depending upon the input normalization method applied. We also investigated whether spurious cell states could be introduced during imputation. We chose cells from two distinct groups (pure H2228 and HCC827) in the *RNA mixture* dataset and examined the correlations among samples. Overall, the sample correlation within the same group is lower when the mRNA amount decreases (Figure 2G) and correlation increases after imputation (Figure 2H-J). All three methods clearly separate the two pure groups, however *kNN-smoothing* introduces a spurious intra-group correlation structure (Figure 2H) that is also shown in the PCA (Figure 2C), which is consistent with observations from another recent study [27]. Moreover, we found the extra clusters were related to the input RNA amount, which implies that *kNN-smoothing* is sensitive to dropout events.

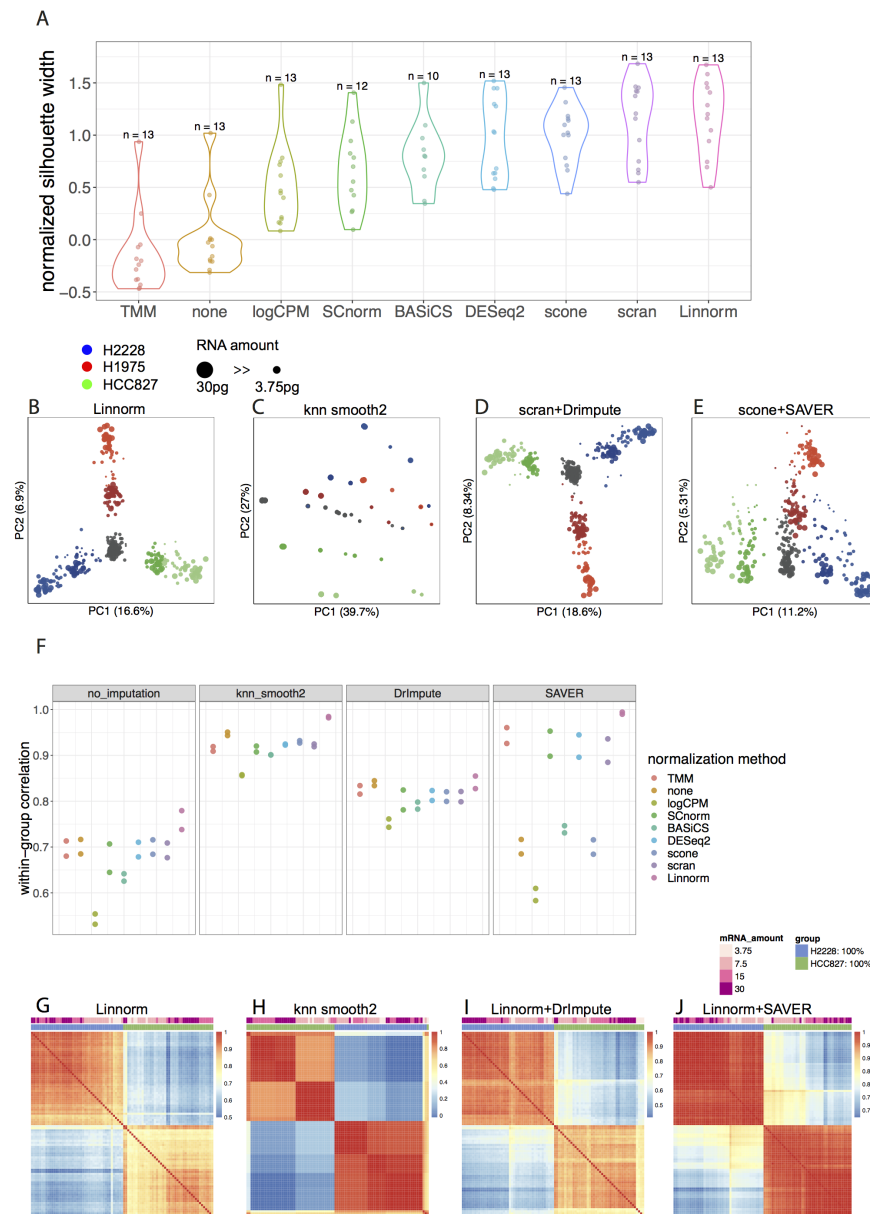


Figure 2. Comparisons of normalization and imputation methods using multiple mixture datasets (A) Silhouette widths calculated using the known cell/mixture groups after different normalization methods, summarised across all datasets and normalized against the Silhouette widths obtained without normalization ('none'). (B-E) Example PCA plots after normalization or with imputation by different methods using the *RNAmix_CEL-seq2* dataset (n=340). Percentage variation explained by each PC is included in the respective axis labels. (F) Average Pearson correlation coefficients for 'pseudo cells' within the same groups in the *RNAmix_CEL-seq2* and *RNAmix_Sort-seq* datasets (n=2) for different combinations of normalization and imputation methods. (G-J) Heatmap of Pearson correlation coefficients of samples in *RNAmix_CEL-seq2* dataset that have pure H2228 (n=45) or HCC827 (n=44) RNA obtained from different imputation methods.

Comparisons of clustering methods

Performance of 5 representative clustering methods, including *RaceID3* [28], *RCA* [29], *Seurat* [30], *clusterExperiment* [31] and *SC3* [32], were evaluated across all datasets. As there is no function to choose

the optimal number of clusters in *Seurat*, two resolutions, 0.6 and 1.6 were used. The resolution parameter controls the number of clusters, with higher values producing more clusters. In addition to the various normalized gene expression matrices, we also evaluated *Seurat* using its own default pipeline (denoted *Seurat_pipe*) that starts from the raw gene count matrix. PCA and t-SNE plots showing clustering results from 3 methods across 3 different datasets is shown in Supplementary Figure 5. We measured the performance of clustering methods by calculating both the entropy of cluster accuracy (ECA) and entropy of cluster purity (ECP) (see Methods). We consider these two metrics together to account for both under and over-clustering, with methods that have both low ECP and low ECA having optimal cluster assignments (Figure 3A). Good correlation was observed between these two metrics and the Adjusted Rand Index (ARI) [33] (data not shown), which is commonly used to evaluate clustering performance by computing similarity to the annotated clusters.

As shown in Figure 3B-E, which presents the two best results for each method on different datasets, no particular algorithm consistently outperformed others across all experimental designs under default settings. In general, *Seurat* achieved a good balance between under-clustering and over-clustering across all datasets, and performs best when there was clear separation between cell types, as was the case in the single cell datasets (Figure 3D-E). *RaceID3* outperformed all other methods in the more complex cell mixture datasets. The accuracy of all methods was lowest in the *cell mixture* datasets (Figure 3B), due to the continuous population structure which gives less separation between different clusters compared to the other experiments. For the single cell datasets, we note that the true cluster number given by genotype information is likely to be an underestimate, with subtle sub-clusters present within each cell line, as highlighted in the t-SNE and UMAP visualisations (Supplementary Figure 3A). Methods that over-cluster the single cell data may well capture true biological signal.

In addition to showing the best results for each method, a linear model was fitted to 2,323 analysis combinations (different normalization, imputation and clustering methods applied across different datasets) using either ARI or the true number of clusters as dependent variables and the different methods as covariates (Supplementary Figures 5A-B) to further investigate the contribution of each method to the results. Similar to what has been shown in Figure 3B, C and E, the *clusterExperiment* method frequently failed to recover the expected population structure, and *SC3* under-clusters most datasets (Figure 3B-C). Interestingly, we found that *kNN-smoothing* was associated with lower ARI and higher cluster numbers (Supplementary Figures 5B-C), which is consistent with our previous results (Figure 2C, H), and serves as a further indication that this method can introduce spurious clusters.

Comparisons of trajectory analysis methods

Five methods, including *Slingshot* [35], *Monocle2* [36], *SLICER* [37], *TSCAN* [38] and *DPT* [39] were evaluated using the *RNA mixture* and *cell mixture* datasets. These datasets were chosen as they both contain clear ‘pseudo trajectory’ paths from one pure cell line to another that are driven by variations in composition of cells/RNA. For simplicity, we chose H2228 as the root state of the trajectory (Figure 4A). We evaluated the correlation between the pseudotime generated from each method and the rank order of the path from H2228 to the other cell lines based on the cells/RNA mixture information (Figure 4B) to examine whether each method can position cells in the correct order. In addition, we calculated the coverage of the trajectory path (Figure 4C), which is the proportion of cells that have been assigned to the correct path, and assesses the sensitivity of the method. We used data generated from combinations of normalization and imputation methods as input to each trajectory analysis method, to assess their impact on performance. In total, 683 analysis combinations of normalization, imputation and trajectory analysis methods applied to different datasets were evaluated.

For each method in each dataset, we selected the best results from all combinations based on the performance metrics (Figure 4B, 4C). In addition to that, a linear model was used to characterize the average performance for each method (Supplementary Figure 6A-B). *Slingshot* and *Monocle2* showed robust results according to both metrics and generated meaningful representations of the trajectory, while *Slingshot* sometimes gave an extra trajectory path (Figure 4A, Supplementary Figure 7). In contrast, *SLICER* places all cells in the correct path but was unable to order them correctly or recover the expected structure induced by the mixture designs. Despite the similar performance of *Slingshot* and

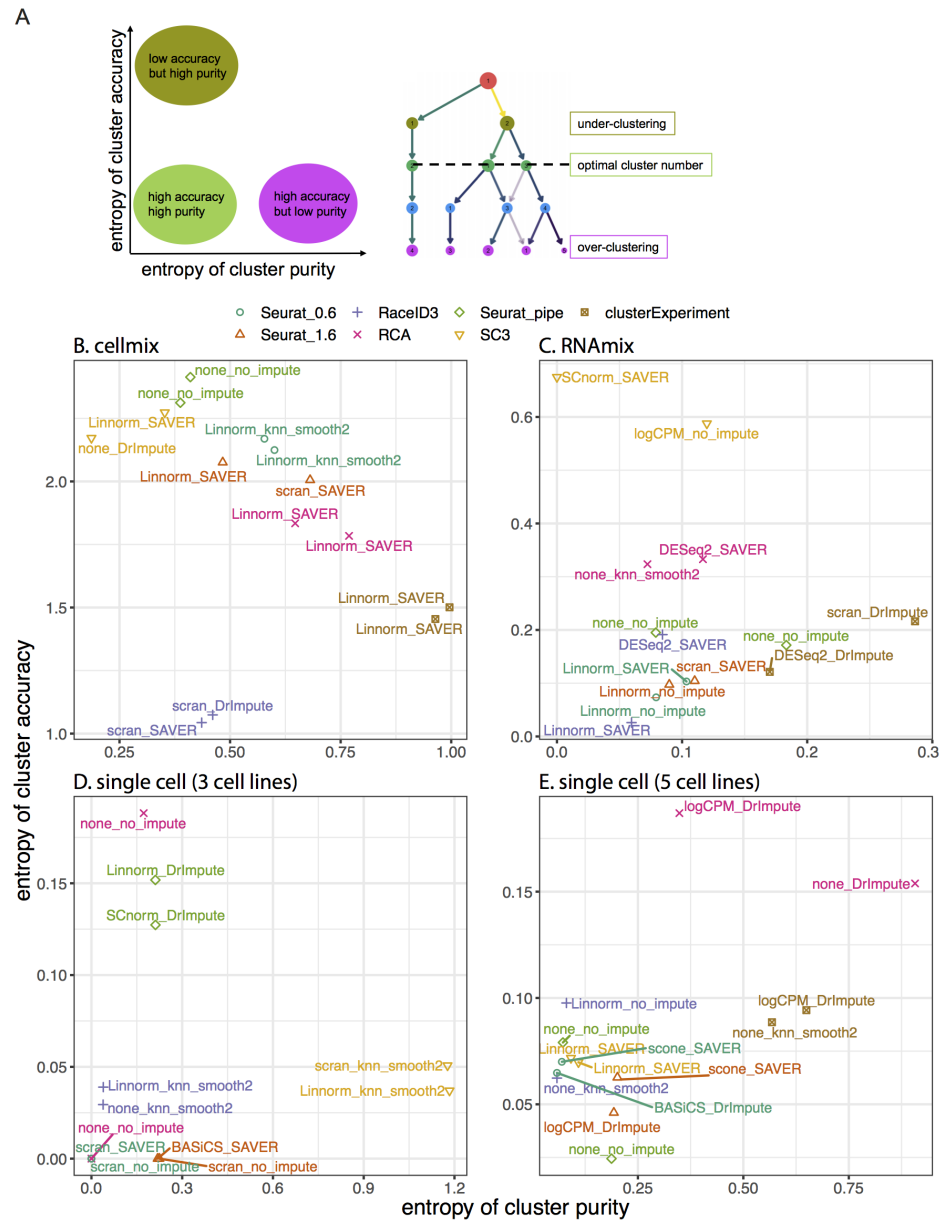


Figure 3. Comparisons of scRNA-seq clustering methods. (A) An overview of the evaluation approach. Entropy of cluster accuracy measures the degree of over-clustering, while Entropy of cluster purity measures under-clustering. The clustering tree adapted from the package *Clustree* [34] allows conceptual visualisation of the two measurements. (B-E) Entropy of cluster purity and entropy of cluster accuracy for the top performing combinations of each method for different datasets. Colours denote different clustering methods and labels indicate the combination of normalization and imputation methods used as input to the clustering algorithms).

Monocle2, their results differ in terms of the way they position cells. *Slingshot* does not perform dimensionality reduction itself and presents the result as is, whereas *Monocle2* uses DDR-tree for dimensionality reduction by default, and tends to place cells at the nodes of the tree rather than in

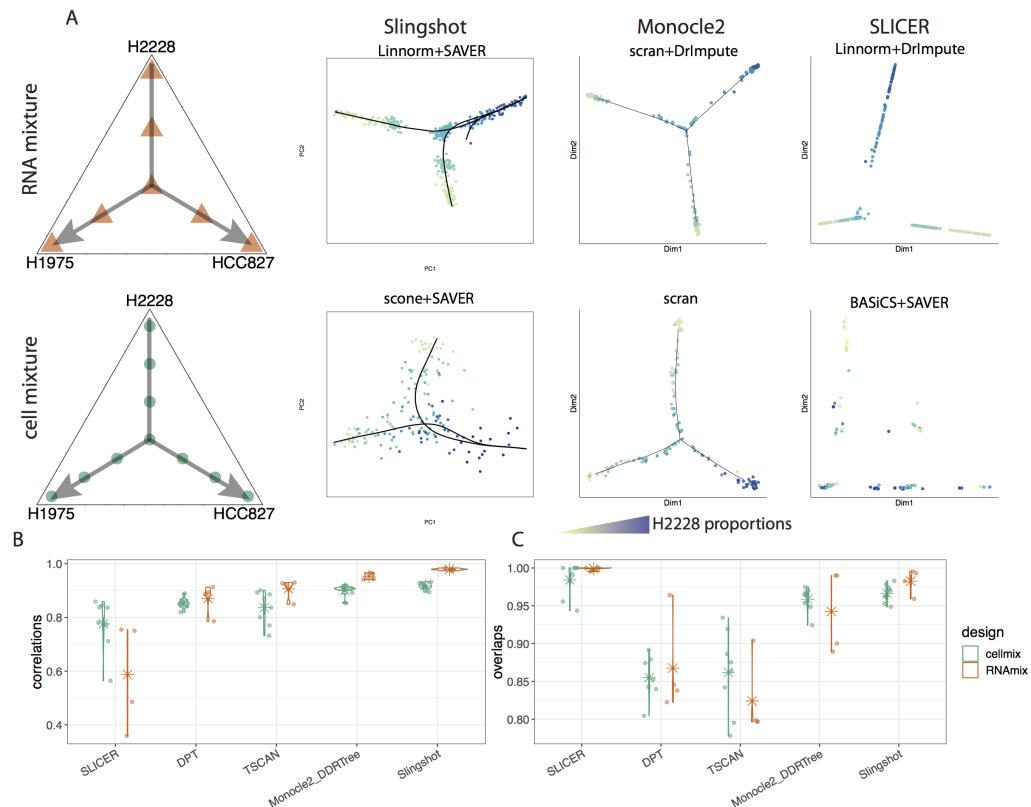


Figure 4. Comparisons of scRNA-seq trajectory analysis methods. (A) The trajectory path chosen for the *RNA mixture* dataset (top) (n=340) and cell mixture dataset (bottom) (n=169) along with visualisations of the output from *Slingshot*, *Monocle-DDRTree* and *SLICER*. Cells are coloured by the proportion of H2228 RNA present, which was chosen as the root of the trajectory. (B) Violin plot showing the Pearson correlation coefficient between the calculated pseudotime and the ground-truth, for the best performing combination of each method on each dataset. Stars indicate the mean values (n=8 for cellmix, n=4 for RNAmix). (C) Violin plot showing the proportion of cells that are correctly assigned to the trajectory. Stars indicate the mean values (n=8 for cellmix, n=4 for RNAmix) in panels B and C.

transition between two nodes (Figure 4A). For example, the *RNA mixture* dataset has 7 clusters by design which are equally distributed along the path between one pure cell line and another. *Monocle2* assigns most of the cells to the three terminal states, leaving only a few in between, which does not reflect the designed structure. Indeed, this feature might fit real situations in cell differentiation, where most cells are in defined cell states with only a small proportion in transition between different groups. However, such an assumption may not always hold and care is therefore needed when interpreting the results.

Comparisons of data integration methods

State-of-the-art methods including *MNNs* [40], *Scanorama* [41], *scMerge* [42], *Seurat* [43] and *MINT* [44] were compared using the *single cell* (*sc_CEL-seq2*, *sc_10X*, *sc_Drop-seq* and *sc_10x_5cl*) and *RNA mixture* (*RNAmix_CEL-seq2* and *RNAmix_Sort-seq*) datasets. The 5 cell line *CEL-seq2* datasets (*sc_CELseq2_5cl_p1*, *sc_CELseq2_5cl_p2* and *sc_CELseq2_5cl_p3*) were excluded due to their high doublet rates (Supplementary Figure 3B). The input to each data integration analysis were from a particular

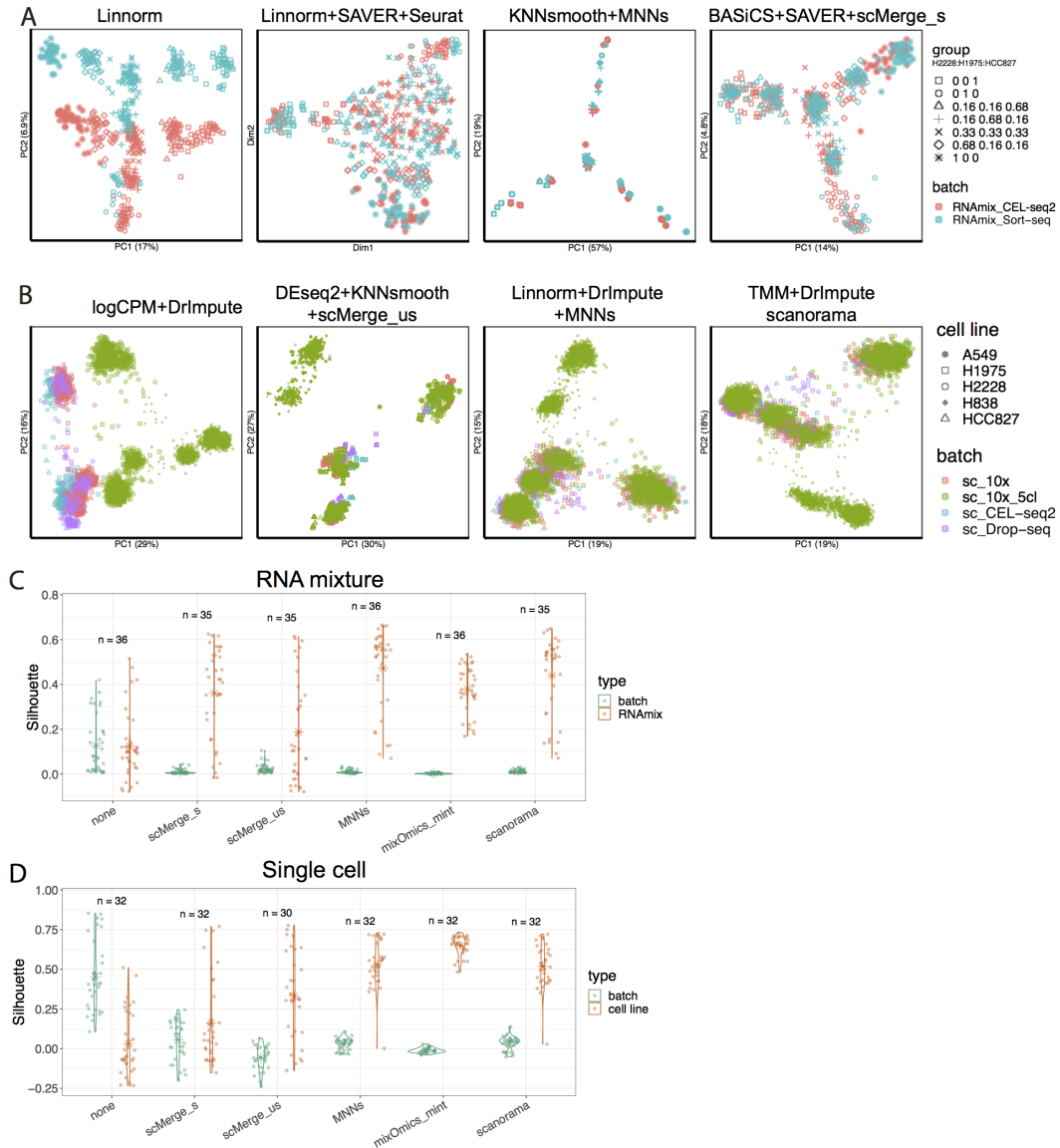


Figure 5. Comparisons of data integration methods for batch effect correction for the RNA mixture (A, B) and the 4 single cell experiments (C, D). (A,C) Silhouette width calculated on either batch information or known sample group. Input data were based on a combination of different normalization and imputation methods, with top performing combination indicated for each method. (B,D) Examples of dimension reduction visualisations for some data integration methods. *Seurat* results were visualised with t-SNE and other methods' results with PCA. *scMerge_s*: supervised *scMerge*; *scMerge_us*: unsupervised *scMerge*, (n=636 for RNA mixture and n=5,319 for single cell).

combination of normalization and imputation applied across the datasets to be combined. This gave rise to 469 different analyses across the two experimental designs. As expected, when naively combining the independent datasets, clear separations related to the different single cell protocols were observed in the PCA plots, with data integration methods removing these effects to varying degrees (Figure 5A-B and Supplementary Figure 8C-D).

MNNs, *Scanorama* and *scMerge* generate batch-corrected gene expression matrix which can then be analyzed using other downstream analysis tools, while Diagonal Canonical Correlation Analysis combined with Dynamic Time Warping from *Seurat* and *MINT* output a low-dimensional representation of the data. *MINT* includes an embedded gene selection procedure whilst projecting the data into a lower dimensional space. We assessed each methods' performance with the silhouette width according to protocol (batch) and known cell line or mixture group information (Figure 5C-D). As *Seurat* uses t-SNE for dimension reduction where distances are not preserved, we also used *kBET* [45] to quantify the remaining batch effect variation (Supplementary Figure 8A-B). When considering the best normalization and imputation methods as input data, most methods performed similarly for the *single cell* design, with the exception of *Seurat* that had a low *kBET* acceptance rate, Supplementary Figure 8A. We observed differences in performance between methods for the *RNA mixture* design that includes a larger number of groups (7) and a smaller number of cells per group than the *single cell* design. *MNNs* gave the best performance according to silhouette width distance, while *Seurat* had the highest *kBET* acceptance rate in this analysis (Supplementary Figure 8B), suggesting a homogeneous mix of batches consistent with the t-SNE visualisation. In addition to the silhouette width and *kBET* acceptance rate, we also performed clustering on the RNA mixture data post integration to assess whether the expected mixture groups were retained. Most methods except *Seurat* have high ARI and gave the correct number of clusters (Supplementary Figure 9) after appropriate normalization and imputation, indicating that they preserve biological signal.

Discussion

By incorporating many different combinations of normalization and imputation in downstream analyses, we were able to assess the robustness and variability of the final outputs in light of their inputs. As summarised in Figure 6, the performance of methods varied across different datasets, with no clear winners in all situations. However, consistently satisfactory results were observed for *scraper*, *Linnorm*, *DrImpute* and *SAVER* for normalization and imputation; *Seurat* for clustering; *Monocle2* and *Slingshot* for trajectory analysis as well as *MNNs* for data integration. Some normalization and imputation method combinations were also found to give good results in most downstream analysis tasks, such as *Linnorm* and *SAVER*. Variations were also observed in the ability of methods to handle different inputs. Methods, such as *Linnorm* and *SC3* produced relatively consistent results regardless of their input dataset, while others such as *SAVER* was more sensitive to these inputs. By evaluating the results from method combinations across different tasks, we observed a number of interesting trends related to the suitability of different preprocessing methods in downstream analysis. For example, we found that although imputation generally improves the results of clustering and trajectory analyses, it can lead to poor mixing of data from different batches in data integration analyses (Supplementary Figure 6D).

Interestingly, the various ensemble methods, which combine results from multiple algorithms in a bid to improve performance, did not always outperform individual methods. The ensemble methods *SC3* and *clusterExperiment* did not outperform other clustering methods and *scone* normalization also gave mixed results on different datasets.

Our comparison is subject to a number of limitations such as the linear mixture settings, which may not be a realistic model for developmental trajectories where regulatory gene expression may be non-linear and non-systemic. Also methods are mostly compared under default settings, which may not give optimal performance across all datasets. The number of methods for each task can be easily expanded using our *CellBench* software for a more in depth analysis of specific tasks or to explore the effect different starting parameters have on the results. Our benchmarking platform will benefit future package developers as it allows new methods to be evaluated on the same standards, avoiding ambiguity

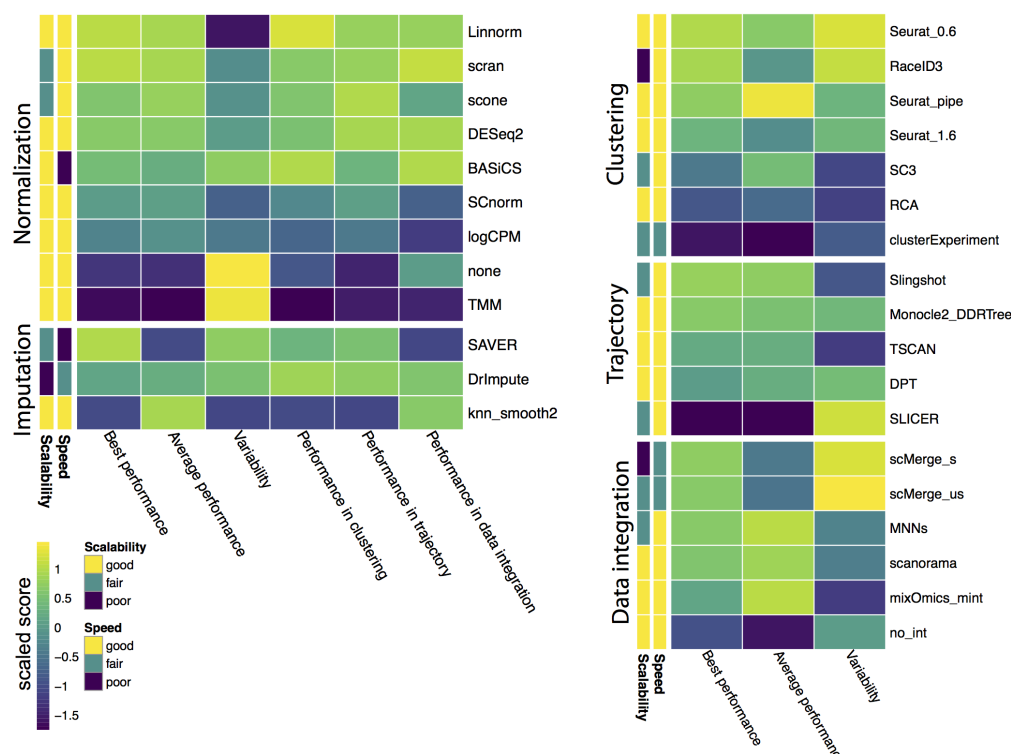


Figure 6. Summary of results from methods comparisons using scRNA-mixology datasets. Methods are ranked by the best performance in each category. The average performance and variability show the average result and the variance when given different input dataset and processed by different upstream methods. For normalization and imputation, the impact each method have on downstream analysis is also shown, which measures the changes in results when applying such method. Results have been scaled and standardised to apply to the same colour scale.

caused by cherry-picking evaluation datasets. We hope that this study will reinvigorate interest in the important area of benchmark data generation and analysis, providing new insights into current best practice and guide the development of better scRNA-seq algorithms in the future to ensure the biological insights derived from single cell technology stand the test of time.

Methods

Study design

Five human lung adenocarcinoma cell lines (HCC827, H1975, A549, H838 and H2228) were cultured separately and the same batch was processed in three different ways (Figure 1). Firstly, single cells from each cell line were mixed in equal proportions, with libraries generated using three different protocols: CEL-seq2, Drop-seq with Dolomite equipment and 10X Chromium.

Secondly, RNA was extracted in bulk from three cell lines (HCC827, H1975 and H2228), mixed in 7 different proportions and diluted to single cell equivalent amounts (Supplementary Figure 1A). In total, there are 8 mixtures in the plate layout with 49 replicates of each mixture. The mix1 and mix2 samples have the same proportions of the three cell lines ($\frac{1}{3} : \frac{1}{3} : \frac{1}{3}$) but were prepared separately in order to assess the variation introduced during the RNA dilution and mixture step. In addition to the RNA

mixtures, we also designed a dilution series in the same plate to create variations in the amount of RNA added. The amounts ranged from 3.75pg to 30pg (Supplementary Figure 1A) and were intended to simulate differences in cell size. In total, each mixture had 4 different RNA starting amounts with replicates per mixture of 6, 14, 14 and 14 for the 3.75pg, 7.5pg, 15pg, 30pg dilutions respectively.

Thirdly, single cells from three cell lines (HCC827, H1975 and H2228) were sorted into 384-well plates, with an equal number of cells per well in different combinations. For most of the wells, we sorted 9 cells in total, with different combinations of three cell lines distributed in ‘*pseudo trajectory*’ paths (Supplementary Figure 1B), where the major trajectory is similar to the *RNA mixture* design while the minor trajectory is the combination that only contains cells from two cell lines instead of three, which is similar in design to our previous study [46]. For the major trajectory, we also included a population control for each combination, which includes 90 cells in total (i.e. a large sample) instead of 9. Apart from the trajectory design, we also varied the cell numbers and qualities to study the data characteristics in these configurations. We included 9 replicates with 3 cells in total with one cell from each cell line, to simulate “small cells” The 9-cell wells were sub-sampled after pooling to get single cell equivalents of RNA, with three replicates in 1/9 and one in 1/3. We applied different clean up ratios to the three replicates after library generation to induce batch effects of a purely technical nature and study how clean up affects the data.

Cell culture and mRNA extraction

The cell lines were retrieved from ATCC (<https://www.atcc.org/>) and cultured in Roswell Park Memorial Institute (RPMI) 1640 medium with 10% fetal calf serum (FCS) and 1% Penicillin-Streptomycin. Three cell lines H2228, H1975 and HCC827 were cultured for the *cell mixture*, *RNA mixture* and *single cell* experiments and later, five cell lines (H2228, H1975, HCC827, A549 and H838) were grown for another *single cell* experiment. The cells were grown independently at 37°C with 5% carbon dioxide until near 100% confluency.

For the three cell lines (HCC827, H1975 and H2228), cells were dissociated into single cell suspensions in FACS buffer and sorted for the *cell mixture* and *single cell* experiment (see below for sorting strategy). The remaining cells were centrifuged and frozen at -80°C and subsequently RNA was extracted using a Qiagen RNA miniprep kit. The amount of RNA was quantified using both Invitrogen Qubit fluorometric quantitation and an Agilent 4200 bioanalyzer to get an accurate estimation. The extracted RNA was diluted to 60 ng/ μ l and mixed in different proportions, according to the study design. The different mixtures were further diluted to create an RNA series that ranged from 3.75pg to 30pg, each of which was dispensed into CEL-seq2 and SORT-seq primer plates using a Nanodrop II dispenser. Prepared RNA mixture plates were sealed and immediately frozen upside down at -80°C.

Cell sorting and single cell RNA sequencing

For CEL-seq2, single cells were flow sorted into chilled 384-well PCR plates containing 1.2 μ l of primer/ERCC mix using a BD FACSAria III flow cytometer. Sorted plates were sealed and immediately frozen upside down at -80°C. These plates, together with the RNA mixture plates, were taken from -80°C and processed using an adapted CEL-Seq2 protocol with the following variations. The second strand synthesis was performed using NEBNext Second Strand Synthesis module in a final reaction volume of 8 μ l and NucleoMag NGS Clean-up and Size select magnetic beads were used for all DNA purification and size selection steps. For the 9-cell mixture plates, clean up of the PCR product was performed with 2 $\times$ 0.7-0.9 bead/DNA ratio. For the *single cell* and *RNA mixture* plates, two different clean up ratios for the PCR product were used (0.8 followed by 0.9). The choice of clean up ratio was optimized from the QC results of the 9-cell mixture data and the SORT-seq protocols. For the 5 cell line single cell mixture experiment, the protocol was further optimised by pooling the sample after first strand cDNA synthesis.

The cell mixture plates were sorted with most wells containing 9 cells in total in different combinations that were processed using our adapted CEL-seq2 protocol described above with variations in the pooling step. After the second strand synthesis, materials from the 9-cell mixtures and 90-cell population controls were pooled separately into different tubes and the volumes were measured. Then for

the 9-cell mixture sample, $3 \times 1/9$ and $1 \times 1/3$ of the total pooled material were taken and these four samples were processed separately in the following step. At the PCR product clean up stage, the clean up ratios for the $3 \times 1/9$ samples were 0.7, 0.8 and 0.9 respectively, and 0.7 for the $1/3$ 9-cell mixture sample and the 90-cell population controls. The SORT-seq protocol is similar to CEL-seq2 but uses oil to prevent evaporation. This allows reductions in the reaction volume which can be dispensed using the Nanodrop II liquid handling platform (GC biotech). In summary, $2.0 \mu\text{l}$ vapor-lock oil was added to each well of the plate, followed by $0.1 \mu\text{l}$ of primer/ERCC mix. The reaction volume for RT and first strand synthesis were $0.075 \mu\text{l}$ and $0.568 \mu\text{l}$ respectively. The composition of the various mixes was the same as for CEL-seq2. The sample pooling was achieved by centrifuging the plates upside down into a container covered with parafilm and carefully separating the oil from the other materials. The PCR clean up ratio used for SORT-seq was 0.8 followed by 0.9. We experienced significant sample loss during sample pooling such that only $\sim 60\%$ of the total volumes were recovered, which was lower compared to the CEL-seq2 protocol (90%).

For the 10X and Drop-seq protocols, cells were PI stained and 120,000 live cells were sorted for each cell line by FACS to acquire an accurate equal mixture of live cells from the three cell lines. This mixture was equally split into three parts, where one part was then processed by the Chromium 10X single cell platform using the manufacturer's (10X Genomics) protocol. The second part was processed by Dolomite Drop-seq with standard Drop-seq protocols [47]. The third part was sorted in a 384-well plate and processed using the standard CEL-seq2 protocol, with a PCR clean up ratio of 0.8 followed by 0.9. For the five cell line experiment, cells were counted using Chamber Slides and roughly 2 million cells from each cell line were mixed and processed by 10X. Three CEL-seq2 plates were sorted from the same sample. All samples, including Drop-seq, 10X and CEL/SORT-seq, were sequenced on an Illumina Nextseq 500.

Data preprocessing and quality control

scPipe (1.3.0) was used for data preprocessing and quality control to generate a UMI-deduped gene count matrix per dataset. All data were aligned using *Rsubread* (1.28.1) [48] to the GRCh38 human genome and its associated annotations, with ERCC spike-in sequences added as extra chromosomes. For 10X, we processed the 5,000 most enriched cell barcodes, with `comp=3` used in the function `scPipe::detect_outliers` for quality control to remove poor quality cells. For CEL-seq2 and SORT-seq, we used the known cell barcode sequences for cell barcode demultiplexing and `comp=2` was used in the function `scPipe::detect_outliers` for quality control. The background contamination was high for Drop-seq, so we first ran `scPipe::detect_outliers` with `comp=3` to remove outlier cells and ran it again with `comp=2` to remove background noise caused by droplets without beads. The quality control metrics, including intron reads for each cell, were generated during cell barcode demultiplexing by the function `scPipe::calculate_QC_metrics`. Intron reads are defined as any reads that map to the gene body but do not overlap an annotated exon. The PCA and t-SNE results were generated using `runPCA` and `runTSNE` in the *scater* (1.8.2) package with default parameters and perplexity set to 30. The UMAP results [49] were calculated using the `umap` function from the *umap* (0.2.0) R package [50].

Data normalization and imputation

The raw counts were used as input to each normalization algorithm and all methods were blind to the mixture groups. To have a fair comparison, the normalized counts from algorithms such as *BASiCS* (1.4.0) and *SCnorm* (1.4.2) which do not generate values on a log-scale were $\log_2$ transformed after an offset of 1 was added to the counts. The raw counts were also $\log_2$ transformed before calculating the Pearson correlation and silhouette widths.

We used *edgeR* (3.24.2) to calculate count-per-million (CPM) and TMM (trimmed mean of *M*-values) normalized values. The *BASiCS* method requires spike-in genes, so we did not apply it to our datasets generated by 10X or Drop-seq which both lacked ERCC spike-ins. For *scone* (1.6.0), we set the maximum number of unwanted variation components to 0 for the removal of unwanted variation method (RUV) and ignored QC metrics.

Data analysis using *scrn* (1.8.2), *Linnorm* (2.6.0), *DrImpute* (1.0), *DESeq2* (1.20.0), and *SCnorm* (1.4.2) was performed with default settings. For the *RNA mixture* data, *kNN-smoothing* (2.0) was run with $k = 16$. The `size.factor` parameter was set to 1 in *SAVER* (1.1.1) to override its internal normalization procedure. *BASiCS* (1.4.0) was run with 5,000 MCMC iterations, 500 warm up iterations and a thinning parameter of 10.

The Pearson correlation coefficient was calculated using gene expression after normalization or imputation for samples with the same RNA mixture proportion, as these samples are replicates and any differences in gene expression should be contributed by variations in RNA amount and technical noise. We performed PCA using normalized counts and calculated the silhouette width on the first two PCs to assess whether normalization was able to preserve the known structure. For any clustering of n samples (here a cluster refers to a particular mixture or a cell line), the silhouette width of sample i is defined as

$$sil(i) \equiv \frac{b(i) - a(i)}{\max(a(i), b(i))} \in [-1, 1] \quad (1)$$

where $a(i)$ denotes the average distance (Euclidean distance over the first two PCs of expression measures) between the i th sample and all other samples in the cluster to which i belongs to, and $b(i)$ is calculated as below: for all other clusters C ,

$$b(i) = \min_C d(i, C) \quad (2)$$

where $d(i, C)$ denotes the average distance of i to all observations to C . Methods with better performance have higher silhouette width. The function `silhouette` from the package *cluster* (2.0.7) [51] was used to calculate the silhouette width.

Clustering

Our comparison of clustering methods used all mixture datasets except *cellmix5*, which is the population control data. To obtain truth for the *single cell* datasets (*sc_CEL-seq2*, *sc_10x*, *sc_Drop-seq*, *sc_10x_5cl*, *sc_CEL-seq2_5cl_p1*, *sc_CEL-seq2_5cl_p2* and *sc_CEL-seq2_5cl_p3*) we used *Demuxlet* [52], which exploits the genetic differences between the different cell lines to determine the most likely identity of each cell. The predicted cell identities in each dataset corresponded largely to clusters seen when visualising the data. Five methods, including *clusterExperiment* (2.2.0), *RaceID3* (0.1.3), *RCA* (1.0), *SC3* (1.10.0) and *Seurat* (2.3.4) were compared. Each method is used as specified by the authors in its accompanying documentation. For each dataset, the inputs for each method were normalized and imputed by different methods and the top 1,000 highly variable genes were selected using the `trendVar` and `decomposeVar` functions in *scrn*. The same gene selection method was also applied in other downstream analyses such as trajectory analysis and data integration. The *Seurat* package has its own data preprocessing pipeline that takes raw UMI counts as inputs and includes normalization and gene selection (referred to as *Seurat_pipe* in the results). Most methods besides *Seurat* have functions to help choose the optimal cluster numbers. Therefore two resolutions, 1.6 (*Seurat.1.6*) and 0.6 (*Seurat.0.6*) were applied to get greater or fewer clusters respectively.

In order to compare the performance of the clustering methods, we looked at two measures: entropy of cluster accuracy, $H_{accuracy}$, and entropy of cluster purity, H_{purity} . With M and N representing the cluster assignment generated from clustering methods and annotations (ground truth), we define these measures as follows:

$$H_{accuracy} = - \frac{\sum_{i=1}^M \sum_{j=1}^{N_i} p(x_j) \log(p(x_j))}{M} \quad (3)$$

$$H_{purity} = - \frac{\sum_{i=1}^N \sum_{j=1}^{M_i} p(x_j) \log(p(x_j))}{N} \quad (4)$$

For $H_{accuracy}$, M denotes the clustering generated by a given method, and N_i are the true clusters in the i th generated cluster. Similarly, in H_{purity} , N denotes the true clusters while M_i is the method

assigned cluster for the i th true cluster. The entropy of cluster accuracy measures the diversity of the true group labels within each cluster assigned by the clustering algorithm. A low value indicates that the cells in a cluster identified by a given method are homogeneous and from the same group. $H_{accuracy}$ does not account for over-clustering though, and in an extreme case, a method that assigns a unique cluster for each cell will have an $H_{accuracy}$ of 0. In contrast, the entropy of cluster purity measures the diversity of the calculated cluster labels within each of the true groups and offers no control of under-clustering. A method that assigns all cells into one cluster will have an H_{purity} value of 0, which is indicative of high cluster purity.

Trajectory analysis

The comparison of trajectory analysis methods used all 9-cell mixture datasets (*cellmix1* to *cellmix4*) and the *RNA mixture* datasets generated by the CEL-seq2 and SORT-seq protocols. For each dataset, the gene count matrix is normalized using different normalization methods and imputation methods. The top 1,000 most highly variable genes were selected using the `trendVar` and `decomposeVar` functions in *scrn*. Five methods, including *Slingshot* (1.0.0), *Monocle2* (2.8.0), *SLICER* (0.2.0), *TSCAN* (1.20.0) and *DPT* (0.6.0) were compared using these data. *Slingshot* requires the dimensionality reduced matrix and cluster assignment as input. Similar to the approach described by the authors, we used PCA (`scater::runPCA`) for dimensionality reduction and fitted a Gaussian mixture model to the first two PCs to obtain the cluster assignments using the `Mclust` function from the *mclust* (5.4.2) R package [53]. Next, the first two PCs and the clustering results were used as input for *Slingshot*. DDR-Tree, a scalable reversed graph embedding algorithm, was used for *monocle2* for dimensionality reduction and tree construction. *SLICER* applies locally linear inference to extract features and reduce dimensions. To make it easier for comparison, the pure H2228 samples were selected as the root cells or root state when generating the trajectory and computing pseudotime. Then for the branching structure generated by each method, we searched for the best match to the two branches: H2228 to H1975 and H2228 to HCC827 and calculated the proportion of overlap of cells between the real path and the branch obtained by each method.

Data integration

The main characteristics of the data integration methods applied are described in Supplementary Table 3. These analyses made use of the R packages *scrn* (1.8.2) for *MNNs*, *Seurat* (2.3.4) for Diagonal Canonical Correlation Analysis (CCA) and *scMerge* (0.1.14), *mixOmics* (6.6.1) for MINT and the *Scanorama* Python library (1.0). The input data for each analysis were the normalized and imputed results from different methods for each dataset. The *scMerge* method was run in both unsupervised (*scMerge_us*) and supervised (*scMerge_s*) modes using cell line identity or RNA mixtures as groups.

We calculated the silhouette width coefficient to compare how well the different methods combined data from different protocols. In the *single cell* and *RNA mixture* datasets, the clusters are defined based on either the known batch/protocol information or cell/mixture groups. Silhouette coefficients were calculated on the first two principal components from PCA for each method that output a data matrix (*MNNs*, *scMerge* and *Scanorama*) or the first two components from *MINT*. A high value for the batch silhouette width indicates that a strong protocol effect remains, whilst a high value for the biological group silhouette width indicates that the biological signal is retained after data integration. The kBET acceptance rate was calculated using *kBET* (0.99.5) with default parameters, with a high rate indicating homogeneous mixing of samples from different batches. The *Seurat* package (2.3.4) with default parameters was used to perform clustering on the dimensionality reduced RNA mixture datasets after integration. The clustering results were evaluated using ARI and the estimated number of clusters, with results shown in Supplementary Figure 9.

Pipeline analysis with *CellBench*

CellBench (0.99.4) contains functions and data structures that simplify the testing of combinations of analysis methods without duplicating code. In addition to a benchmarking framework, *CellBench* also

provides functions to access preprocessed data objects for the samples described in this study. To use *CellBench*, methods are modified with wrapper functions to ensure that the methods that perform the same task are able to accept a common input format and produce a common output format. Lists of wrappers for one pipeline stage are passed to the `apply_methods` function along with the data. Successive application of the `apply_methods` function with lists of wrappers containing downstream pipeline steps automatically generate combinations between the methods and all upstream combinations. *CellBench* allows individual combinations to fail without affecting the execution of other pipelines, and the function `time_methods` can be used to measure running times. The scripts containing the wrappers and code used for this analysis can be found at https://github.com/LuyiTian/CellBench_data/tree/master/script.

Performance summary

The results from all analysis combinations including the performance scores are listed in Supplementary Table 4 and are also available from GitHub. Results were plotted using the *ggplot2* (3.1.0) [54] and *pheatmap* (1.0.10) [55] R software. Clustering of the Pearson correlation coefficients in Figure 2 was performed using the `hclust` function in base R. To summarise the results of each analysis we fitted a linear model using the `lm` base R function with the performance score as the dependent variable, and type of method and experimental designs as binary covariates. The coefficient of each method indicates the degree to which the method is positively or negatively associated with performance. This analysis is summarised in Supplementary Figure 5B-C and 8.)

Figure 6 summarises the performance across all evaluated methods. For each task, we considered a specific metric. Clustering performance was assessed with the ARI coefficient, trajectory analysis with the correlation between pseudotime and ground truth and integration across protocols, normalization and imputation with the silhouette width coefficient and *kBET* results. The best performance is defined as the average of the best two results for each design, and the average performance is calculated across all results. Variability refers to the variation of all results. For normalization and imputation methods, the coefficients of the linear model were scaled and shown in the heatmap to summarise performance and indicate which method yields better results in the downstream analysis compared with others.

The running time for each combination is given in Supplementary Table 5. For each method, a linear model was fitted (using `lm`) that regressed the running time against the number of cells on the log-scale from the different datasets. The cell number coefficient indicates the scalability of the method. The scalability is classified as poor if the coefficient is larger than 2, good if the coefficient is smaller than 1 and fair if between 1 and 2. The running time for a method was regarded as poor if it was longer than 30 minutes, good if shorter than 5 minutes and fair if in between.

Data Availability

Raw data are available under GEO SuperSeries GSE118767. A summary of the individual accession numbers is given in Supplementary Table 1. The processed `SingleCellExperiment` R objects are available from https://github.com/LuyiTian/CellBench_data.

Code Availability

Code used to perform the comparative analyses and generate the figures is available from https://github.com/LuyiTian/CellBench_data. The *CellBench* R package was developed for benchmarking single cell analysis methods and is available from <https://github.com/Shians/CellBench> and Bioconductor (<https://www.bioconductor.org/packages/CellBench>).

Acknowledgements

We thank C. Weeden and M.-L. Asselin-Labat for providing the cell lines used in this study, J. Schreuder and D. Lin for assistance in conducting experiments and I. Virshup for assistance in the data integration analysis.

Author Contributions

L.T. designed the study, cultured the cells and performed the CEL-seq2, SORT-seq and 10X experiments, conducted data analysis, generated the figures and wrote the manuscript with inputs from all authors. X.D. performed data analysis for the comparison of normalization methods. S.F. helped with data analysis for the comparison of clustering methods. K.A.L.C. and A.J. performed data analysis for the comparison of data integration methods. S.S. wrote the *CellBench* package. D.A.-Z. performed the CEL-seq2 experiments. A.S. and J.S.J. performed the 10X experiments. T.S.W. performed the Drop-seq experiment. S.H.N. and M.E.R. designed the study. M.E.R. supervised the analysis and wrote the manuscript. All authors read and approved the final manuscript.

References

1. Cole, M. B. *et al.* Performance Assessment and Selection of Normalization Procedures for Single-Cell RNA-Seq. *bioRxiv* (2017).
2. Yip, S. H., Sham, P. C. & Wang, J. Evaluation of tools for highly variable gene discovery from single-cell RNA-seq data. *Briefings in Bioinformatics* 1–7 (2018).
3. Soneson, C. & Robinson, M. D. Bias, robustness and scalability in single-cell differential expression analysis. *Nature Methods* **15**, 255–261 (2018).
4. Freytag, S., Tian, L., Lönnstedt, I., Ng, M. & Bahlo, M. Comparison of clustering tools in R for medium-sized 10x Genomics single-cell RNA-sequencing data. *F1000Research* **7**, 1297 (2018).
5. Duò, A., Robinson, M. D. & Soneson, C. A systematic performance evaluation of clustering methods for single-cell RNA-seq data. *F1000Research* **7**, 1141 (2018).
6. Saelens, W., Cannoodt, R., Todorov, H. & Saeys, Y. A comparison of single-cell trajectory inference methods. *Nature Biotechnology* (2019).
7. Svensson, V. *et al.* Power analysis of single-cell RNA-sequencing experiments. *Nature Methods* **14**, 381–387 (2017).
8. Jiang, L. *et al.* Synthetic spike-in standards for RNA-seq experiments. *Genome Research* **21**, 1543–1551 (2011).
9. Brennecke, P. *et al.* Accounting for technical noise in single-cell RNA-seq experiments. *Nature Methods* **10**, 1093–1098 (2013).
10. Grün, D., Kester, L. & Van Oudenaarden, A. Validation of noise models for single-cell transcriptomics. *Nature Methods* **11**, 637–640 (2014).
11. Cope, L. M., Irizarry, R. A., Jaffee, H. A., Wu, Z. & Speed, T. P. A benchmark for Affymetrix GeneChip expression measures. *Bioinformatics* **20**, 323–331 (2004).
12. Sequencing Quality Control Consortium. A comprehensive assessment of RNA-seq accuracy, reproducibility and information content by the Sequencing Quality Control Consortium. *Nature Biotechnology* **32**, 903–914 (2014).

13. Tung, P.-Y. *et al.* Batch effects and the effective design of single-cell gene expression studies. *Scientific Reports* **7**, 39921 (2017).
14. R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria (2018). URL <https://www.R-project.org/>.
15. Huber, W. *et al.* Orchestrating high-throughput genomic analysis with Bioconductor. *Nature Methods* **12**, 115–121 (2015).
16. Tian, L. *et al.* scPipe: a flexible R/Bioconductor preprocessing pipeline for single-cell RNA-sequencing data. *PLoS Computational Biology* **14**, e1006361 (2018).
17. Lun, A. T., Bach, K. & Marioni, J. C. Pooling across cells to normalize single-cell RNA sequencing data with many zero counts. *Genome Biology* **17**, 75 (2016).
18. Robinson, M. D. & Oshlack, A. A scaling normalization method for differential expression analysis of RNA-seq data. *Genome Biology* **11**, R25 (2010).
19. Robinson, M. D., McCarthy, D. J. & Smyth, G. K. edgeR: A Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics* **26**, 139–140 (2009).
20. Love, M. I., Huber, W. & Anders, S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology* **15**, 550 (2014).
21. Vallejos, C. A., Marioni, J. C. & Richardson, S. BASiCS: Bayesian Analysis of Single-Cell Sequencing Data. *PLoS Computational Biology* **11**, e1004333 (2015).
22. Bacher, R. *et al.* SCnorm: Robust normalization of single-cell RNA-seq data. *Nature Methods* **14**, 584–586 (2017).
23. Yip, S. H., Wang, P., Kocher, J. P. A., Sham, P. C. & Wang, J. Linnorm: improved statistical analysis for single cell RNA-seq expression data. *Nucleic Acids Research* **45**, e179 (2017).
24. Wagner, F., Yan, Y. & Yanai, I. K-nearest neighbor smoothing for high-throughput single-cell RNA-Seq data. *bioRxiv* (2018).
25. Gong, W., Kwak, I.-Y., Pota, P., Koyano-Nakagawa, N. & Garry, D. J. DrImpute: imputing dropout events in single cell RNA sequencing data. *BMC Bioinformatics* **19**, 220 (2018).
26. Huang, M. *et al.* SAVER: gene expression recovery for single-cell RNA sequencing. *Nature Methods* **15**, 539–542 (2018).
27. Andrews, T. S. & Hemberg, M. False signals induced by single-cell imputation. *F1000Research* **7**, 1740 (2018).
28. Herman, J. S., Sagar & Grün, D. FateID infers cell fate bias in multipotent progenitors from single-cell RNA-seq data. *Nature Methods* **15**, 379–386 (2018).
29. Li, H. *et al.* Reference component analysis of single-cell transcriptomes elucidates cellular heterogeneity in human colorectal tumors. *Nature Genetics* **49**, 708–718 (2017).
30. Macosko, E. Z. *et al.* Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. *Cell* **161**, 1202–1214 (2015).
31. Purdom, E. & Risso, D. *clusterExperiment: Compare Clusterings for Single-Cell Sequencing* (2017). R package version 1.4.0.
32. Kiselev, V. Y. *et al.* SC3: Consensus clustering of single-cell RNA-seq data. *Nature Methods* **14**, 483–486 (2017).

33. Hubert, L. & Arabie, P. Comparing partitions. *Journal of Classification* **2**, 193–218 (1985).
34. Zappia, L. & Oshlack, A. Clustering trees: a visualization for evaluating clusterings at multiple resolutions. *GigaScience* **7**, giy083 (2018).
35. Street, K. *et al.* Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics. *BMC Genomics* **19**, 477 (2018).
36. Qiu, X. *et al.* Reversed graph embedding resolves complex single-cell trajectories. *Nature Methods* **14**, 979–982 (2017).
37. Welch, J. D., Hartemink, A. J. & Prins, J. F. SLICER: Inferring branched, nonlinear cellular trajectories from single cell RNA-seq data. *Genome Biology* **17**, 047845 (2016).
38. Ji, Z. & Ji, H. TSCAN: Pseudo-time reconstruction and evaluation in single-cell RNA-seq analysis. *Nucleic Acids Research* **44**, e117–e117 (2016).
39. Haghverdi, L., Büttner, M., Wolf, F. A., Büttner, F. & Theis, F. J. Diffusion pseudotime robustly reconstructs lineage branching. *Nature Methods* **13**, 845–848 (2016).
40. Haghverdi, L., Lun, A. T., Morgan, M. D. & Marioni, J. C. Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. *Nature Biotechnology* **36**, 421 (2018).
41. Hie, B. L., Bryson, B. & Berger, B. Panoramic stitching of heterogeneous single-cell transcriptomic data. *bioRxiv* (2018).
42. Lin, Y. *et al.* scMerge: Integration of multiple single-cell transcriptomics datasets leveraging stable expression and pseudo-replication. *bioRxiv* (2018).
43. Butler, A., Hoffman, P., Smibert, P., Papalexi, E. & Satija, R. Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nature Biotechnology* **36**, 411–420 (2018).
44. Rohart, F., Eslami, A., Matigian, N., Bougeard, S. & Lê Cao, K.-A. Mint: a multivariate integrative method to identify reproducible molecular signatures across independent experiments and platforms. *BMC Bioinformatics* **18**, 128 (2017).
45. Büttner, M., Miao, Z., Wolf, F. A., Teichmann, S. A. & Theis, F. J. A test metric for assessing single-cell RNA-seq batch correction. *Nature Methods* **16**, 43–49 (2019).
46. Holik, A. Z. *et al.* RNA-seq mixology: Designing realistic control experiments to compare protocols and analysis methods. *Nucleic Acids Research* **45**, e30 (2017).
47. Klein, A. M. *et al.* Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. *Cell* **161**, 1187–1201 (2015).
48. Liao, Y., Smyth, G. K. & Shi, W. The subread aligner: fast, accurate and scalable read mapping by seed-and-vote. *Nucleic Acids Research* **41**, e108–e108 (2013).
49. McInnes, L., Healy, J. & Melville, J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *ArXiv e-prints* (2018).
50. McInnes, L., Healy, J., Saul, N. & Grossberger, L. Umap: Uniform manifold approximation and projection. *The Journal of Open Source Software* **3**, 861 (2018).
51. Maechler, M., Rousseeuw, P., Struyf, A., Hubert, M. & Hornik, K. *cluster: Cluster Analysis Basics and Extensions* (2018). R package version 2.0.7-1.

-
52. Kang, H. M. *et al.* Multiplexed droplet single-cell RNA-sequencing using natural genetic variation. *Nature Biotechnology* **36**, 89–94 (2018).
 53. Scrucca, L., Fop, M., Murphy, T. B. & Raftery, A. E. mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. *The R journal* **8**, 289 (2016).
 54. Wickham, H. *ggplot2: Elegant Graphics for Data Analysis* (Springer-Verlag New York, 2016).
 55. Kolde, R. *pheatmap: Pretty Heatmaps* (2018). URL <https://CRAN.R-project.org/package=pheatmap>. R package version 1.0.10.

Chapter 4

Comprehensive characterization of single cell full-length isoforms in human and mouse with long-read sequencing

Comprehensive characterization of single cell full-length isoforms in human and mouse with long-read sequencing

Luyi Tian^{1,2*}, Jafar S. Jabbari³, Rachel Thijssen^{1,2}, Quentin Gouil^{1,2}, Shanika L. Amarasinghe^{1,2}, Hasaru Kariyawasam¹, Shian Su^{1,2}, Xueyi Dong^{1,2}, Charity W. Law^{1,2}, Alexis Lucattini³, Jin D. Chung⁴, Timur Naim⁴, Audrey Chan⁴, Chi Hai Ly^{4,5}, Gordon S. Lynch⁴, James G. Ryall^{4,6}, Casey J.A. Anttila¹, Hongke Peng^{1,2}, Mary Ann Anderson^{1,2,7}, Andrew W. Roberts^{1,2,7,8,9}, David C.S. Huang^{1,2}, Michael B. Clark¹⁰, Matthew E. Ritchie^{1,2*}

1. The Walter and Eliza Hall Institute of Medical Research, Parkville, VIC, Australia.
2. Department of Medical Biology, The University of Melbourne, Parkville, VIC, Australia.
3. Australian Genome Research Facility, Victorian Comprehensive Cancer Centre, Melbourne, VIC, Australia.
4. Centre for Muscle Research, Department of Physiology, The University of Melbourne, Melbourne, VIC, Australia
5. Present address: Department of Neurology, Stanford University, Stanford, California, USA.
6. Present address: VOW, North Parramatta, NSW, Australia
7. Clinical Haematology, Peter MacCallum Cancer Centre and Royal Melbourne Hospital, Melbourne, VIC, Australia.
8. Centre for Cancer Research, University of Melbourne, Melbourne, VIC, Australia.
9. Victorian Comprehensive Cancer Centre, Melbourne, VIC, Australia.
10. Centre for Stem Cell Systems, Department of Anatomy and Neuroscience, The University of Melbourne, Parkville, VIC, Australia

* Corresponding authors: tian.l@wehi.edu.au; mritchie@wehi.edu.au

Abstract

Alternative splicing shapes the phenotype of cells in development and disease. Long-read RNA-sequencing recovers full-length transcripts but has limited throughput at the single-cell level. Here we developed single-cell full-length transcript sequencing by sampling (*FLT-seq*), together with the computational pipeline *FLAMES* to overcome these issues and perform isoform discovery and quantification, splicing analysis and mutation detection in single cells. With *FLT-seq* and *FLAMES*, we performed the first comprehensive characterization of the full-length isoform landscape in single cells of different types and species and identified thousands of unannotated isoforms. We found conserved functional modules that were enriched for alternative transcript usage in different cell populations, including ribosome biogenesis and mRNA splicing. Analysis at the transcript-level allowed data integration with scATAC-seq on individual promoters, improved correlation with protein expression data and linked mutations known to confer drug resistance to transcriptome heterogeneity. Our methods reveal previously unseen isoform complexity and provide a better framework for multi-omics data integration.

Main

Single-cell RNA-sequencing (scRNA-seq) is a widely adopted method for profiling transcriptomic heterogeneity in health and disease¹. However, assessing transcript-level changes between cell types using current scRNA-seq protocols is challenging due to their reliance on short-read sequencing. Previous studies using plate-based methods^{2,3} have focused on individual alternative splicing events such as exon skipping, due to the fundamental limitation of short-read sequencing in linking distal splicing outcomes belonging to the same transcript. The Smartseq3 protocol⁴ can achieve full-length transcript coverage but is still unable to assemble the complete transcript sequence and is heavily reliant on the reference annotation. Droplet-based methods⁵ such as 10x only sequence the 3' or 5' end of transcripts which largely precludes isoform identification. Long-read sequencing can overcome this limitation and generate full-length transcript information in single cells, as illustrated in several recent studies⁶⁻⁹. However, the throughput of current long-read sequencing platforms is still not comparable to short-read platforms and the per-base accuracy is also lower, which together create many issues. Limited sequencing throughput introduces a trade-off between the per-cell sequencing depth and the number of cells or genes processed. Protocols such as ScISO-Seq⁶ perform shallow sequencing per cell while RAGE-Seq⁷ focuses on specific transcripts rather than the whole transcriptome. On top of the current protocol limitations, another pressing issue is the lack of data analysis pipelines for long-read transcriptome data, especially for single cells. Methods such as *FLAIR*¹⁰ and *TALON*¹¹ have been developed for isoform annotation and quantification but have not been benchmarked and lack the ability to perform quantification at the single-cell level. Therefore, we need both new protocols and computational pipelines to overcome these limitations.

To this end, we developed FLT-seq and *FLAMES* to perform single-cell isoform sequencing and data analysis. Adapted from the popular 10x Chromium platform, FLT-seq is a cost-effective approach to discover and quantify isoforms in single cells by integrating data from short- and long-read sequencing technologies. By subsampling single cells from a full 10x run and applying nanopore long-read sequencing, FLT-seq can achieve comparable sequencing depth per cell to that obtained from short-read platforms. For data analysis, we developed a computational framework to perform single-cell full-length analysis of mutations and splicing (*FLAMES*), which includes cell barcode, UMI assignment from nanopore reads and semi-supervised isoform discovery and quantification. We applied FLT-seq and *FLAMES* to human and mouse samples containing different cell types and highlight shared splicing patterns in human cancer cells and mouse quiescent muscle stem cells. Differential transcript usage analysis pinpointed common functional modules and genes across samples. We also demonstrate that FLT-seq is a promising tool for detecting coding variants of clinical relevance. Taken together, our new protocol and data analysis pipeline enable comprehensive characterization of the full-length isoforms present in single cells that are currently missed by short-read sequencing datasets.

High-throughput single-cell full-length transcriptome sequencing with FLT-seq

FLT-seq is based on the Chromium scRNA-seq platform (10x Genomics), with optimizations to better amplify the full-length cDNA. Since the throughput of long-read platforms is still limited compared to Illumina sequencing platforms, we developed a strategy to subsample 10-20% of the 10x Chromium generated Gel Bead-in-Emulsions (GEMs) after reverse transcription (Figure 1A). This is equivalent to sampling 10-20% of the cells as the cDNA of each cell is still within the GEMs. After subsampling, the GEMs are pooled separately for library preparation. Part of the amplified cDNA from the 10-20% subsample is used for Oxford Nanopore Technologies long-read library preparation and sequencing on a PromethION. The remainder of the cDNA from the 10-20% sample together with the GEMs from the 80-90% sample are used for regular 10x library preparation and Illumina sequencing in parallel. In the end, long-read data from the 10-20% subsample of cells and Illumina short-read data for all cells are generated by this protocol.

We demonstrate FLT-seq by profiling 16,660 cells from diverse biological systems, 2,737 of which were sequenced by both long-read and short-read technologies (Figure 1A, Table S1). Firstly, we used our previously published *scmixology* design¹², which involved an equal mixture of cells from five cell lines (H2228, H838, H1975, HCC827, A549). Two biological replicates were profiled (*scmixology1* and *scmixology2*) with FLT-seq together with 10x scATAC-seq for the second replicate. In addition to the cell-line mixtures, we sequenced freshly isolated quiescent and activated muscle stem cells (*MuSCs*) from mouse. Lastly, we applied FLT-seq to a cryogenically preserved peripheral blood mononuclear cell (PBMC) sample from a patient (*CLL2*) whose chronic lymphocytic leukemia (CLL) had progressed on venetoclax treatment after a durable response. The sample was prepared together with the 10x CITE-seq assay with 17 antibody markers. This demonstrated the broad utility of FLT-seq, which is compatible with different 10x transcriptomic assays and can be applied to both fresh and frozen samples. The Uniform Manifold Approximation and Projection (UMAP) visualization presented the cell populations as expected (Figure S1) and revealed no obvious bias in the sampling process of FLT-seq (Figure 1B). Collectively, we generated 230 million long reads across all samples, together with scATAC-seq and CITE-seq data for the *scmixology2* and *CLL2* samples respectively.

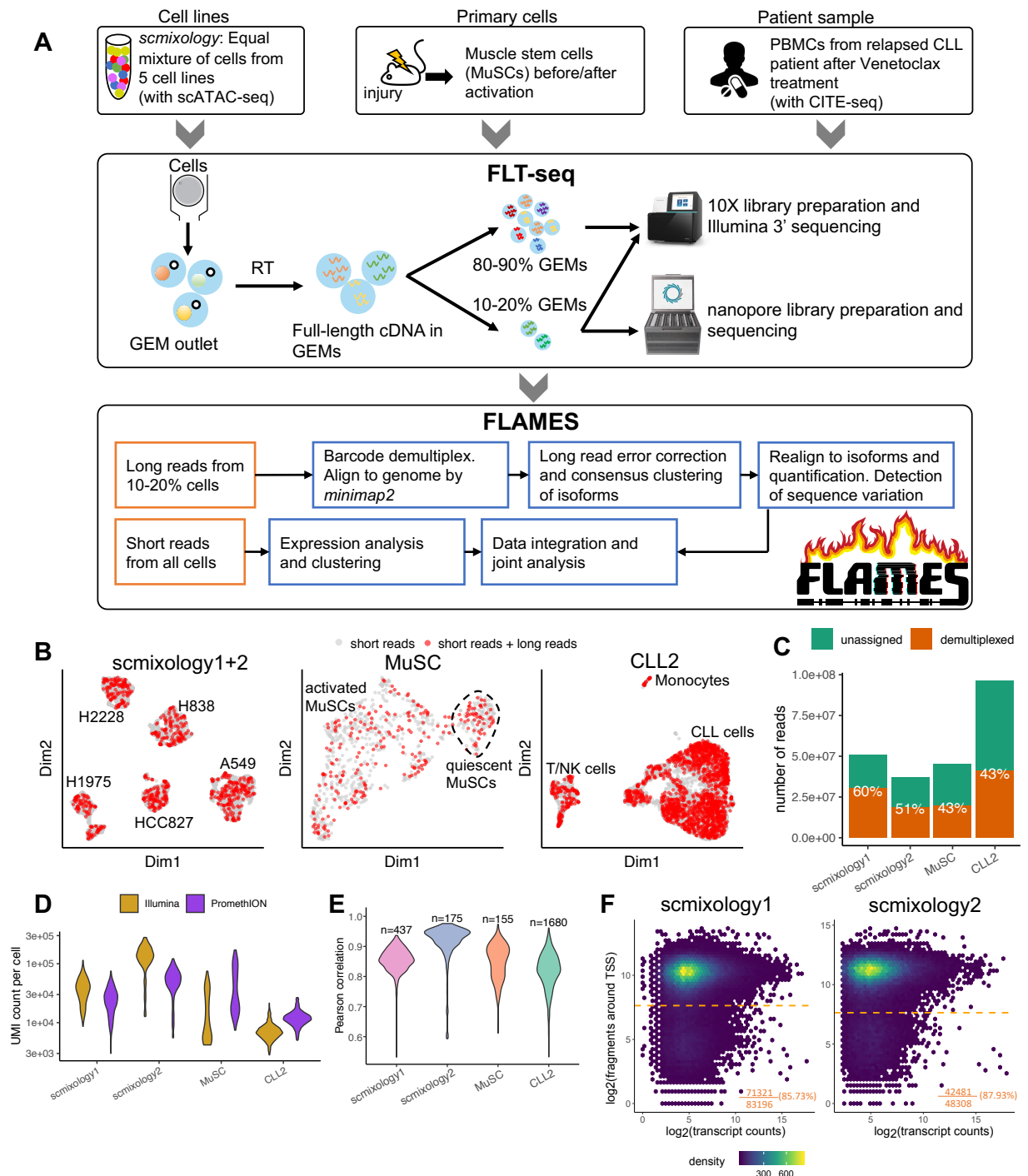


Figure 1. Overview of experimental design, FLT-seq and FLAMES methods and basic summary statistics.

(A) Summary of the study design, with an overview of the FLT-seq protocol and FLAMES data processing pipeline.

(B) UMAP visualization of cells in each sample, cells colored in red are sampled for long-read sequencing. *scmixology1* and *scmixology2* were integrated and shown together in one plot. All UMAP visualizations are based on short-read data

(C) The number of nanopore reads generated from each sample, and the percentage of reads that were assigned a cell barcode.

(D) Distribution of UMI counts per cell for Illumina and nanopore data in each sample.

(E) Correlations between gene UMI counts generated from nanopore long-read and Illumina short-read data.

(F) Density scatter plot shows the relationship between transcript-level counts and scATAC-seq read counts around the TSS regions. The horizontal orange line shows the threshold calculated that separates open chromatin from the background. The percentage shows the transcripts that have their TSSs in open chromatin regions.

Single-cell isoform detection and quantification with *FLAMES*

We developed a flexible computational framework called *FLAMES* (Full-Length Analysis of Mutations and Splicing in long-read data) to detect and quantify isoforms for both single-cell and bulk long-read data (Figure 1A and Methods). Input to *FLAMES* are fastq files generated from the long-read platform. Using the cell barcode annotation obtained from short-read data as the reference, it identifies and trims cell barcodes/UMI sequences from the long reads. After barcode assignment, all reads were aligned to the relevant genome to obtain a draft read alignment. The draft alignment is then polished and grouped to generate a consensus transcript assembly. All reads are aligned again using the transcript assembly as the reference and quantified.

Next, we benchmarked *FLAMES* on a bulk SIRV spike-in dataset¹³ for which the isoform structure and abundances are known *a priori*. We compared our new method to *FLAIR*, *TALON* and *StringTie2*¹⁴ which are designed for bulk long-read isoform detection and quantification. Our results clearly show that *FLAMES* outperforms other tools both in terms of the isoform detection (Figure S2A) and quantification (Figure S2B). It is notable that *TALON* and *FLAIR* still have many false positive transcripts after filtering by abundance (82% and 78% respectively) compared to *FLAMES* (3%), highlighting the importance of choosing an appropriate method.

After validating the method, we used *FLAMES* to preprocess and analyze the four datasets generated. 40-60% of the long reads could be assigned to an expected cell barcode and were kept for further analysis (Figure 1C). The transcript coverage of reads realigned to assembled transcripts showed that the percentages of full-length reads decreased for longer transcripts (Figure S3), which also been shown in another study¹⁵. Reads that are not full-length and cannot be uniquely assigned to transcripts are discarded during data processing. The average UMI count per cell ranges from 10,000-60,000, varied by cell type, and was comparable to the short-read counts from the same cells (Figure 1D). Gene-level UMI counts between the matched nanopore and Illumina data were also found to be highly correlated (Figure 1E). The data processed by *FLAMES* showed that FLT-seq generated high quality long-read data that was comparable to the short-read Illumina data when analyzed at the gene-level.

To validate the transcription start sites (TSSs) from the isoforms generated by *FLAMES*, we compared them to the FANTOM5 TSS peaks¹⁶ and found that around 75% of the TSSs are within the FANTOM5 annotation (Figure S4A). For the *scmixology* data where scATAC-seq data from the same populations were also available, we aggregated scATAC-seq signals around the TSSs as an indicator of open promoters. The result showed *scmixology1* and *scmixology2* have similar open promoters and more than 85% of the TSSs are within active promoters, supporting the existence of these transcripts (Figure 1F). In contrast, when we process the *scmixology* data using *TALON*, *FLAIR* and *StringTie2* the results were less optimal. The majority of transcripts generated by *FLAIR* and *TALON* did not match the reference annotations (Figure S4B) and *FLAIR*, *TALON* and *StringTie2* had fewer

TSSs overlapping the FANTOM5 annotations compared to *FLAMES* (Figure S4C). We found similar results when comparing the scATAC-seq signals around the TSS regions from transcripts generated by different methods (Figure S4D), with *TALON* and *FLAIR* having around 40% and 50% of their TSSs in open chromatin regions respectively. In summary, our comparisons yield similar results between a spike-in dataset and the *scmixology* dataset, with *FLAMES* outperforming *StringTie2*, *FLAIR* and *TALON* with the latter two generating many false transcripts.

Characterization of isoforms reveals the distinct splicing landscapes of different cell populations

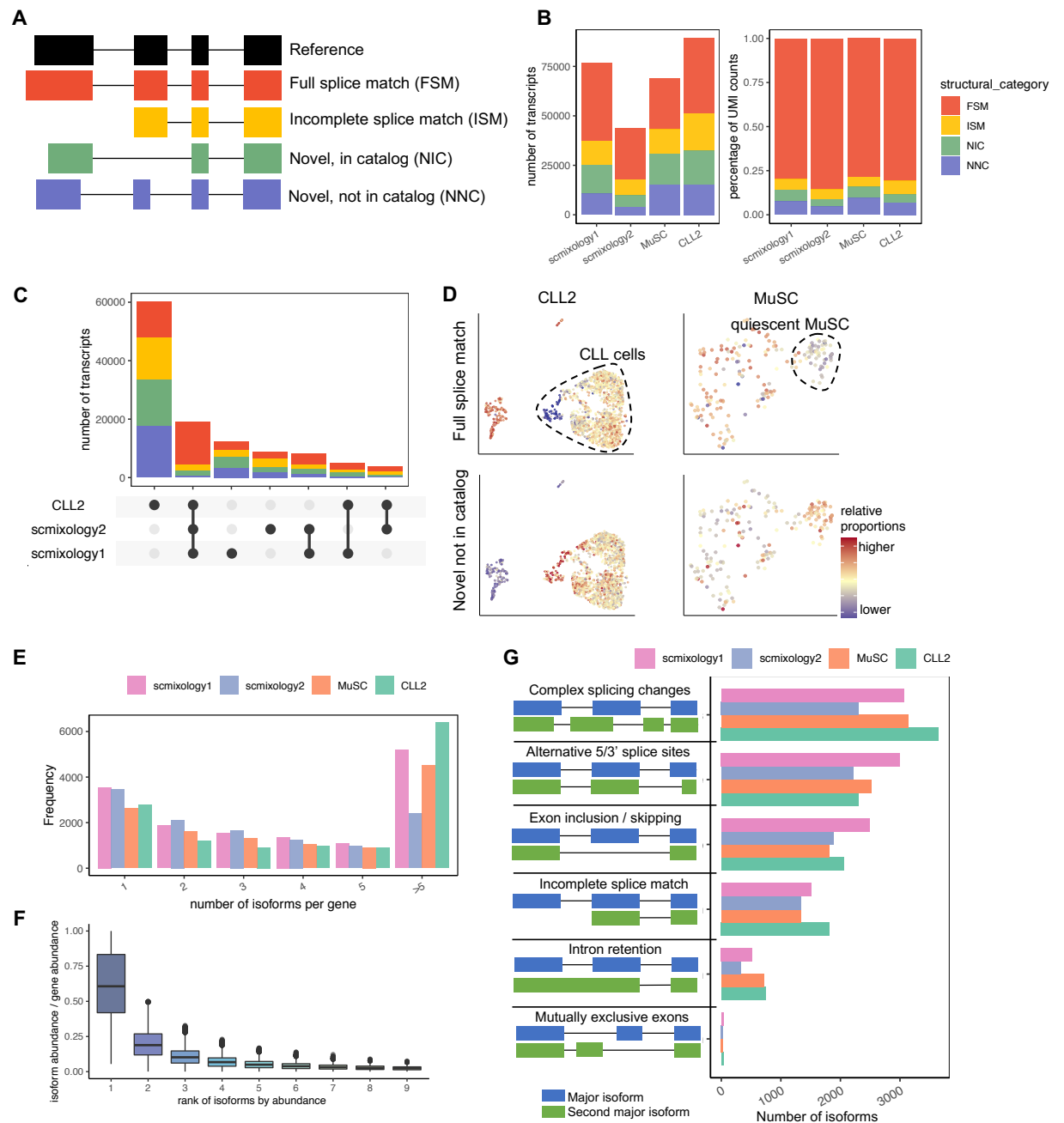


Figure 2. Overview of the single-cell isoform-level analysis from *FLAMES*.

(A) Classification of transcripts according to their splice sites when compared to reference annotations. (B) Summary of transcripts in different categories in (A) both in numbers (left) and in the percentage of UMI counts (right).

- (C) UpSet plot showing overlap of transcripts in human datasets, where number of transcripts shared by different sets of samples are indicated in the top bar chart, colored by categories specified in (A).
- (D) UMAP visualization of *CLL2* and *MuSC* dataset on the cells sampled for long-read sequencing. Colored by percentage of UMI counts of transcripts in FSM (top) and NNC (bottom) categories. CLL cells and quiescent MuSCs are annotated on the plot.
- (E) Bar plot of number of distinct transcripts expressed per gene. Genes with more than five distinct transcripts are merged.
- (F) Box plot showing the percentage of transcript abundance relative to gene abundance for genes express multiple transcripts. Transcripts are ranked by abundance, shown on the x-axis.
- (G) Summary of the type of alternative splicing between the two most abundant transcripts of each gene. The 'Complex splicing changes' category represents transcripts with more than one type of constitutive alternatively spliced event.

We compared the transcripts generated by *FLAMES* to the reference annotation and classified them using the scheme introduced in *SQANTI*¹⁷, including transcripts with all splice junctions matching to reference transcripts (full splice match, FSM) or partially matching to consecutive splice junctions for a reference transcript (incomplete splice match, ISM) and transcripts with novel splice junctions with new (Novel, not in catalog, NNC) or known (Novel, in catalog, NIC) donor and acceptor sites (Figure 2A). We observed that the number of transcripts detected varies between samples and was correlated with the sequencing throughput as shown in Figure 1C. While around half of the transcripts detected were novel, the majority of reads aligned to known transcripts with novel transcripts having lower abundance in general. In addition to the comparison with reference annotations, we also compared the transcripts generated from the three human samples to each other (Figure 2C) and found many transcripts unique to each sample (22%, 27% and 68% for *scmixology1*, *scmixology2* and *CLL2* respectively). A majority of transcripts (~60%) were shared between biological replicates and among these 30% were novel, suggesting that many conserved alternative splicing events in these cell lines have not been annotated.

Within samples we found consistent alterations in splicing patterns between cell populations (Figure 2D). CLL cells had higher proportions of novel transcripts, especially transcripts with novel splice junctions (NNC), compared to non-CLL cells from the same sample, including T cells, NK cells and monocytes. Similarly, quiescent muscle stem cells also had higher proportions of novel transcripts compared to activated stem cells. Analysis of intronic reads from RNA-seq data¹⁸ has demonstrated that intron retention, which would produce novel transcripts, is increased in quiescent mouse muscle stem cells and is essential for these cells to exit the quiescent state. This is consistent with our results and suggests that differences in splicing patterns between different cell populations may act as a regulatory mechanism.

After comparing the transcripts identified against reference annotations, we sought to characterize isoforms within the same gene. Around 80% of genes can be expressed as multiple isoforms. The average number of isoforms expressed per gene ranges between 3 and 6 and varied between the different samples (Figure 2E). The distribution of isoform expression is skewed with only a few abundant transcripts dominantly expressed for most genes (Figure 2F). On average, the two most highly expressed isoforms account for 80% of the total counts (median 85%). Next, we categorized the types of alternative splicing between the two most highly expressed isoforms (Figure 2G). Alternative splicing has mostly been studied based on particular events such as exon skipping or alternative 5' splice sites using short-read sequencing technology¹⁹. Here we found that around 30% of genes have more than one type of

alternative splicing event between the top two isoforms. This means that the 2 most highly expressed isoforms differ by complex splicing changes involving multiple exons, which may not be accurately characterised by short reads because two isoforms could have a skipped exon near the 5' end and a different splice site near the 3' end.

In addition to summarizing isoforms at the sample-level, we examined how splicing variants are captured at the single-cell level. Individual alternative splicing events in single cells might follow a unimodal binomial distribution and a cassette exon is skipped or included at the same probability for all cells. Or cells may not share the same parameter of the binomial distribution and create the bimodal splicing pattern, with some cell always skipping an exon while others always include it. The latter is supported by an early single cell study². Recently by analyzing the splice junctions using short-read scRNA-seq data, splicing in homogeneous cell populations has been found to be largely unimodal²⁰. Here we expanded the binomial model from considering single splice junctions to complete isoforms in order to estimate the isoform choice between the major isoform and other isoforms (see Methods). These results showed the percentage of cells with binary isoform choice is strongly related to the gene abundance and can be approximated by a unimodal binomial distribution (null distribution), which indicates the cell randomly chooses isoforms with a certain frequency (Figure S5A). We also noticed that the *MuSC* and *CLL2* samples contained more genes with binary splicing patterns, especially for genes with low abundance. This could result from the difference in mRNA amounts where the cell lines are transcriptionally active, and the other samples have lower numbers of mRNA molecules per cells. Or it could be that the *MuSC* and *CLL2* samples have heterogeneous populations with different splicing patterns. To further explore the effect of recovery on isoform identification and splicing analysis, we sampled reads from the *scmixology* dataset to retain 20-80% of the total reads for analysis. The number of transcripts detected decreased with lower sampling rate (Figure S5B), especially for novel transcripts, while binary splicing still showed a unimodal distribution (Figure S5C). This suggested that the reliability of isoform choice estimation may be influenced by multiple factors, including the transcriptional activity and heterogeneity of the cell identities.

Common classes of genes with differential transcript usage across samples

Following the analysis of isoform abundance, we investigated whether genes with multiple isoforms exhibited differential transcript usage (DTU) between the clusters or cell types shown in Figure S1. We focused on changes in the use of internal splice junctions, grouping transcripts with the same intron chain. To mitigate the high dropout rate in single-cell data, we aggregated the transcript counts into pseudo-bulk values per cluster and filtered out transcripts with low abundance. Next, we evaluated how sequencing level influenced the results of a DTU analysis using the same downsampling scheme used in the last section (see Methods). We observed that although as expected the number of genes with DTU decreases with decreasing sequencing depth per cell, 58% of the changes could still be detected with 20% of the reads (Figure S5). We then performed DTU analysis on all samples and identified between 500 and 1,000 genes with differential transcript usage (gDTU) in each sample (Figure 3A, Table S2). Nearly half of the gDTU (244 out of 573) are shared between *scmixology1* and *scmixology2*. Besides this overlap, gDTU were largely unique for each sample (Figure S6A). The functional annotation clustering of these genes

revealed shared pathways across different samples, especially pathways related to transcription and translation, such as mRNA splicing and ribosome biogenesis (Figure 3B, Table S3). As an example, different gDTU related to mRNA splicing were found in different samples. PQBP1, which is involved in pre-mRNA splicing, is a gene with DTU in the *scmixology* samples (Figure S6B), while SRSF2 and SRSF3, which belong to a family that act both as general splicing factors and as regulators of alternative splicing²¹, exhibit DTU in the *CLL2* sample (Figure S6C).

Apart from the gDTU that are unique to each sample, we also found a few genes (18) that appeared in all samples (Figure S6A), including a ribosomal protein RPS24 that had the smallest *P*-value among different samples (Table S2). RPS24 is a highly conserved gene between mouse and human, with exon 5 and 6 alternatively spliced to generate different protein coding sequence (Figure 3C, 3D). Further analysis showed transcript usage of this gene is altered at multiple levels in different samples. Firstly, the major transcript of RPS24 was distinct between different samples. Transcripts without exon 5 and 6 were most abundant in the cell lines (*scmixology1* and *scmixology2*, Figure 3C), while the transcript with only exon 5 was highly expressed in the patient derived PBMC sample (*CLL2*, Figure 3C) and *MuSC* sample (Figure 3D). In addition to the major transcript, additional transcripts showed differential transcript usage in different cells within the same sample. The transcript with exon 5 and 6 was more frequently expressed in CLL cells while the transcript without exon 5 and 6 was preferentially expressed in non-CLL cells (Figure 3E). Besides these known transcripts, we also identified a new isoform in mouse (Rps24-001) that was preferentially expressed in quiescent MuSCs (Figure 3F). Different RPS24 isoforms have tissue-specific expression²² and some have been linked to tumor progression²³, however the functional differences between the encoded proteins remains unclear. In summary, we have highlighted the heterogeneity of expression of the RPS24 transcript that is missed in a typical gene-level analysis (Figure 3E, 3F).

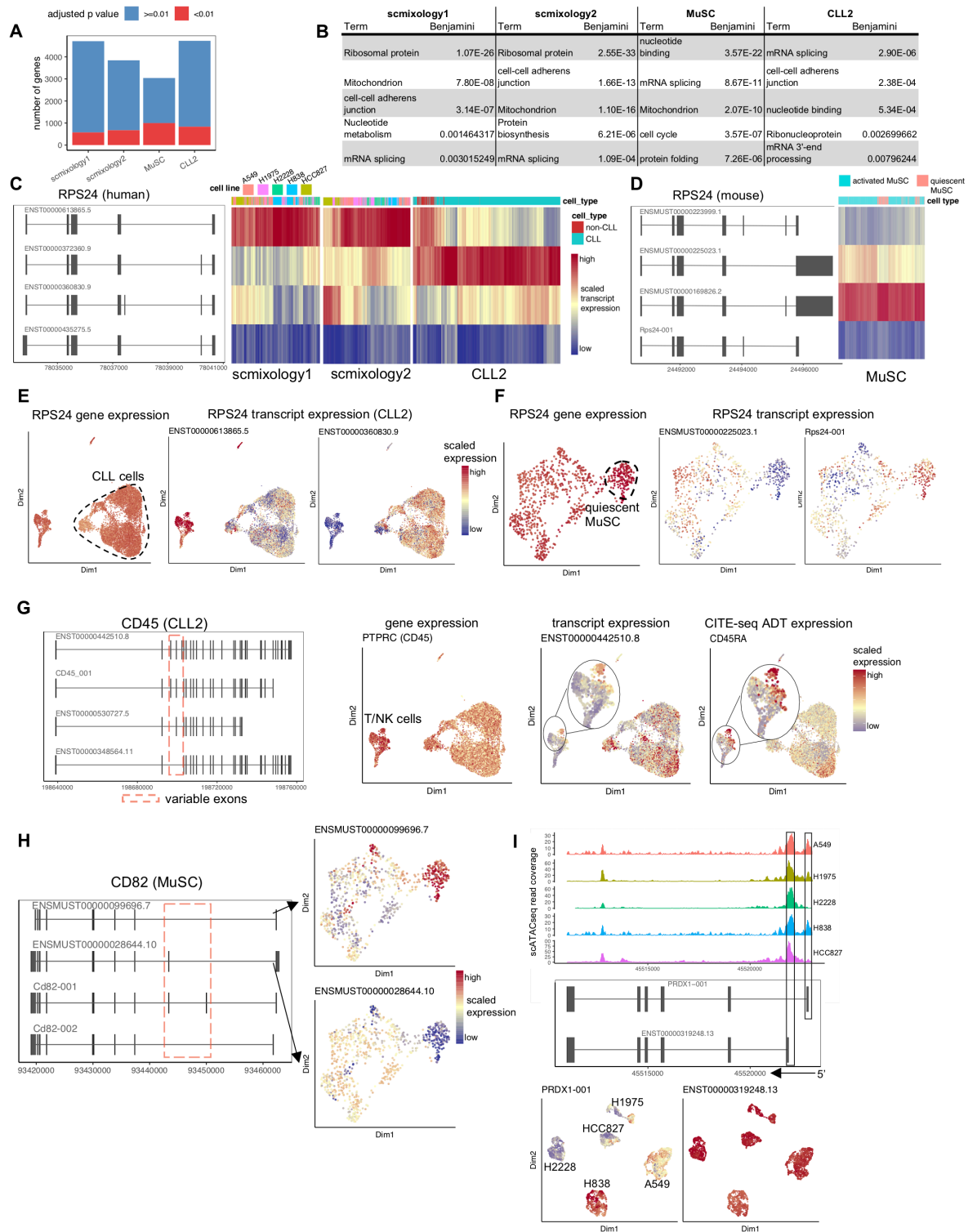


Figure 3. Summary of differential transcript usage results from FLAMES.

(A) Summary of results from the statistical testing of DTU detected many significant genes per sample (adjusted P -value < 0.01).

(B) Table of common functional categories among different samples from the functional enrichment analysis of gDTU.

(C) Top 4 most abundant isoforms of RPS24 in human and heatmap of their expression at the single-cell level in the *scmixology1*, *scmixology2* and *CLL2* samples.

(D) Top 4 most abundant isoforms of RPS24 in mouse and heatmap of their expression at the single cell level in *MuSCs*.

- (E) UMAP of cells in *CLL2*, colored by RPS24 gene expression and transcript expression. Two transcripts with differential expression on different populations were selected. Transcript expression in each cell is colored by scaled relative expression to highlight the difference between different populations.
- (F) Similar to (E), UMAP of cells in *MuSC* sample, colored by RPS24 gene expression and transcript expression.
- (G) Top 4 most abundant isoforms of CD45 in *CLL2* and UMAP visualizations of the cells colored by (from left to right) gene expression, transcript expression and corresponding protein expression.
- (H) Top 4 most abundant isoforms of CD82 in *MuSC*, with UMAP visualization of cells colored by expression of two isoforms that have differential expression between quiescent and activated *MuSC*.
- (I) scATAC-seq read coverage for PRDX1 with cells from each cell line aggregated and plotted together. UMAP plots showing isoform expression, with each cell colored by scaled transcript expression.

Another category of gDTU of interest is cell-surface proteins. Genes encoding cell-surface proteins produce alternative mRNA isoforms, usually by changing the combinations of consecutive exons corresponding to certain functional domains. Some of the isoforms have been characterized, such as CD45, where alternative splicing of exons 4 to 6 are expressed in different lymphocytes²⁴. By analyzing the *CLL2* data where the surface marker expression is available through CITE-seq, we detected multiple isoforms of CD45 (Figure 3G). We found similar expression patterns between the transcript and the protein it encodes, where the protein was quantified by counting antibody derived tags (ADT) from CITE-seq data. The result both validated the isoform quantification from the *FLAMES* pipeline and showed that transcript-level analysis can provide better correlation between mRNA and protein expression that cannot be achieved using gene-level quantification. We also observed other genes encoding important cell surface markers with DTU, such as CD82 (Figure 3H), CD47 (Figure S6D) and CD44 (Figure S6E), with different combinations of consecutive exons. Notably, some of the novel transcripts in CD47 can be observed in different samples (Figure S6D), suggesting conserved alternative splicing patterns that are missing from the reference annotation. In sum, we observed previously overlooked isoform diversity of cell-surface proteins, which could introduce functional diversity and contribute to development and diseases^{25,26}.

Alternative promoters have been shown to regulate cancer-specific transcription²⁷. Here we found transcripts with different TSSs expressed in different cancer cell lines. We have shown that open chromatin captured by scATAC-seq is correlated with the promoter region indicated by the TSS (Figure 1F). As an example, we found multiple isoforms of PRDX1, including a new isoform PRDX1-001 with a different TSS. The new isoform expressed in H838 and A549 contains a novel first exon that is not found in the other cell lines (Figure 3I). Additional open promoter regions are observed in the scATAC-seq data from H838 and A549 that coincide with the new exon (Figure 3I). This result suggests that FLT-seq can be integrated with scATAC-seq at the promoter level to further enhance the resolution of integration and reveal promoter heterogeneity that cannot be found via gene-level integration²⁸.

FLT-seq links coding sequence variation to transcriptome heterogeneity

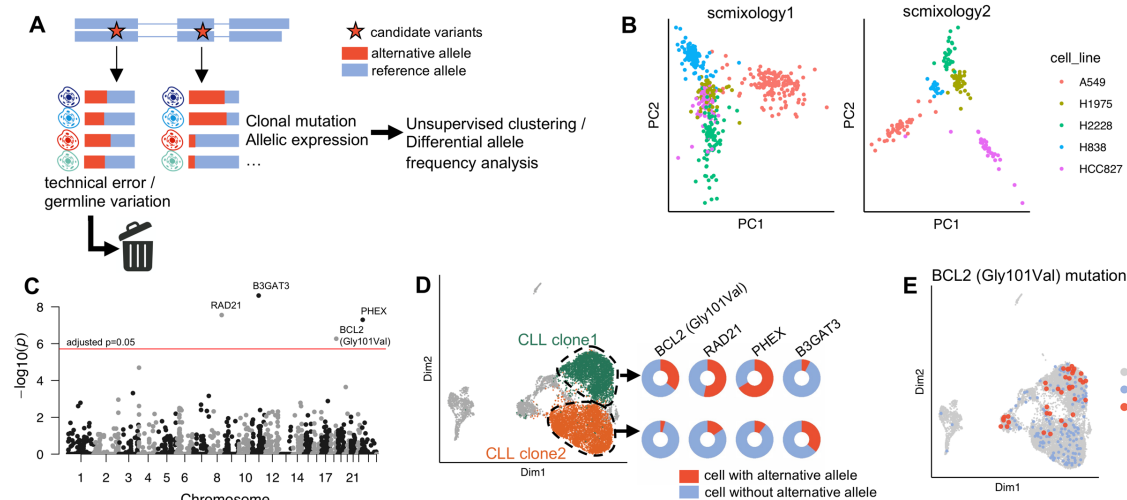


Figure 4. Summary of differential allele frequency analysis to detect coding mutations.

- (A) Summary of variation filtering and analysis pipeline implemented in *FLAMES*. Candidate variants are filtered based on allele frequency first, then based on per cell allele frequency to remove technical artefacts. The remaining variants are used for differential allele frequency analysis.
- (B) PCA on alternative allele matrix using the variants after filtering, colored by unsupervised clustering results using top PCs and annotated with cell lines.
- (C) Manhattan plot of P -values from a differential allele frequency analysis with Benjamini–Hochberg adjustment. Genes with significant variants are labelled.
- (D) UMAP visualization highlighting two CLL populations that have differential allele frequency for the significant variants.
- (E) UMAP visualization of cells colored by Gly101Val mutation status.

Nanopore long-read sequencing allows full transcript coverage compared to 3' or 5'-end short-read sequencing, which provides us with a better chance of identifying coding variations. But the high error rate in sequencing presents an obstacle²⁹. To overcome this challenge, we first exclude homopolymer regions since they have higher sequence error rates^{30,31}. After excluding these regions, ~95% of regions have a reference allele frequency >90% (Figure S7A). The remaining single-nucleotide variants (SNVs) were filtered again using a statistical test in *FLAMES* to enrich for true positives and clonal mutations, based on the assumption that the occurrence of sequencing errors is independent of cell barcode and will occur randomly in all cells (Figure 4A and Methods). We tested our approach on the *scmixology* samples where the five cell lines carry distinct mutations. The principal-component analysis (PCA) on the filtered allele count matrix successfully recapitulated the expected population structure (Figure 4B). Louvain clustering on leading principal components generated similar results to the cell type assignment obtained from running *Demuxlet*³² on the short-read data (98.7% and 99.4% concordance for *scmixology1* and *scmixology2* respectively), which indicated that we can successfully capture mutations in these different cell lines using long-read data. We then performed differential allele frequency analysis to find SNVs that were specific to each cell line. The results showed a high precision (80.1%) with the SNVs called from bulk RNA-seq data from individual cell lines.

After we confirmed the approach using the cell lines, a similar analysis was performed on the *CLL2* dataset to examine the relationship between transcriptome heterogeneity

and sequence variations. We searched for SNVs that only existed in the cancer cells and had a differential allele frequency across different CLL transcriptional clusters. We found four significant mutations (adjusted P -value < 0.05 , chi-square test, Benjamini–Hochberg correction) associated with different CLL clusters (Figure 4C, 4D) and their existence was confirmed by bulk genomic sequencing (data not shown). By investigating the allele frequency of these mutations across clusters (Figure S7B), we identified two subclones, where mutations in the genes *BCL2*, *RAD21* and *PHEX* are enriched in *subclone1* and the mutation located in the gene *B3GAT3* is enriched in *subclone2* (Figure 4D). Although the analysis itself does not require genomic sequence data as a reference, it is important to validate the mutations detected since the changes in allele frequency may also come from allele-specific expression or RNA editing. The Gly101Val mutation has been confirmed to promote resistance to venetoclax treatment by reducing the affinity of *BCL2* for venetoclax³³, and patient *CLL2* was known to carry ~25% Gly101Val mutations. Using FLT-seq, we showed that the mutation is not just subclonal, but also linked to specific transcriptional clusters. The distribution of the Gly101Val mutation was further confirmed by capture enrichment of the *BCL2* gene (Figure S7C), which greatly increased the coverage of the mutation site (Figure S7D). Taken together, FLT-seq provides unbiased high-throughput linking of single-cell mutations and transcriptomic heterogeneity.

Discussion

Transcriptional activity is typically summarized at the gene-level due to the limitations of short-read technology, especially in scRNA-seq studies. The recent development of long-read sequencing technology promises sequencing of the full-length transcript, but its application to single cells has been slowed by a lack of protocols and data analysis pipelines. By optimizing the 10x Genomics scRNA-seq protocols we have developed the scalable and affordable FLT-seq approach by cell sampling to achieve per-cell sequencing depths similar to short reads at a much lower cost than other methods⁶. It combines the advantages of throughput and accuracy from short-read sequencing to identify cell barcodes and perform clustering with the strengths of long-read sequencing that enables isoform discovery and quantification, the results of which are combined in an integrated analysis. FLT-seq is compatible with a variety of 10x transcriptomic assays and could be potentially applied to any single-cell transcriptomic protocol with cell barcoding, including Visium spatial transcriptomics. It could also be combined with other single-cell long-read sequencing approaches such as R2C2³⁴ or other sequencing platforms including PacBio Sequel II. As the throughput and accuracy of long-read sequencing platforms increases, it may be possible to skip FLT-seq's sampling step and apply long-read sequencing only rather than the current hybrid approach to achieve similar aims in future studies.

To analyze noisy long-read data, we developed the *FLAMES* pipeline that can detect and quantify novel and known isoforms in single-cell and bulk samples. It can also find cell-type-specific isoforms and mutations through comparative analysis. We showed that *FLAMES* outperforms other methods both on a synthetic spike-in dataset³⁵ and experimental datasets. To date, *FLAMES* is the only package that can be used to process single-cell long-read sequencing data that has been validated both using synthetic spike-in data and other omics data. The recently published SiCeLoRe method claims to be able to detect and quantify isoforms, although it has not been as

extensively validated. *FLAMES* implements a strict filter on consensus read clustering and filters out spurious reads that are mostly truncated reads with incomplete 3' or 5' ends. It also has two rounds of alignment using the genome and the recovered isoforms as reference to reduce alignment artefacts and distinguish reads aligned to similar isoforms. Our comparison indicated that certain methods generate many spurious transcripts, highlighting the importance of designing appropriate methods to detect novel isoforms.

With FLT-seq and *FLAMES*, we performed the first comprehensive characterization of full-length isoforms at the single-cell level across species and cell types. We detected thousands of novel transcripts expressed at low levels, most of which were unique to each sample. Novel transcripts also showed distinct enrichment patterns at the single-cell level where CLL cells and quiescent muscle stem cells often have more novel transcripts. We frequently found more than one differentially spliced junction between isoforms for many genes, suggesting the inadequacy of studying individual splicing junctions via short reads. Furthermore, we found that single cells often express more than one mRNA isoform per gene and the isoform choice is largely random, challenging the argument that individual cells have binary outcomes in splicing³⁶. In addition to the overall stochasticity of isoform choice, we observed cell type specific isoform usage for genes that are enriched for common functions such as mRNA splicing and ribosome biogenesis. The expression of ribosomal and spliceosome proteins has low correlation with their gene-level expression³⁷, indicating some protein expression levels might be regulated by alternative splicing. Genes encoding cell-surface proteins such as CD44 and CD47 often have cell-type-specific splice variants, some of which have not been previously annotated, and may result in different functions³⁶. The examples we show indicate that identification of cell-type-specific isoforms may have broad applications in many fields such as immunotherapy, where cell surface proteins play an important role.

Compared to short-read scRNA-seq, FLT-seq provides better linkage between proteome and genome as it can resolve the full-length transcript. Transcript abundance can be aligned directly to single-cell protein measurements, which is particularly helpful for proteins with multiple isoforms such as CD45. We also showed that FLT-seq facilitates integration with scATAC-seq data, such that open chromatin signals are not summarized at the gene-level but at the exact transcription start site. Full transcript coverage unlocks the potential to detect mutations and allelic expression for specific isoforms. Through unsupervised analysis, we not only identified coding mutations that cannot be easily detected by 3' or 5'-end short-read sequencing, but we could also associate mutations to different transcriptional signatures. This can improve our understanding of the correlation between the single-cell transcriptome and genome in cancer.

FLT-seq and *FLAMES* provide new avenues for characterizing single-cell transcriptomic heterogeneity at the transcript-level and unveils new questions and challenges. Benchmarking studies that compare the performance of emerging long-read sequencing protocols and computational methods are needed to help our understanding of the strengths and weaknesses of different approaches. Many unanswered questions about transcript expression and alternative splicing in single cells also remain. For example, do the majority of novel transcripts simply reflect stochastic noise in the splicing machinery³⁸, or are they indicative of a genuine

increase in protein diversity? Our methods and analysis provide a starting point for addressing these questions to accelerate isoform-level studies in single cells.

Acknowledgements

We thank Clare Weeden and Marie-Liesse Asselin-Labat for providing the cell lines used in this study, Stephen Wilcox for Illumina sequencing, Kathleen Zeglinski for designing the *FLAMES* logo and Joan Heath for providing critical feedback on this manuscript. This work was supported by funding from the Chan Zuckerberg Initiative DAF, an advised fund of Silicon Valley Community Foundation (grant numbers 2018-182819 and 2019-002443 to M.E.R.), a Melbourne Research Scholarship to L.T., a Leukemia & Lymphoma Society (LLS) of America Fellowship to R.T. (5467-18) and LLS Specialized Center of Research [SCOR] grant no. 7015-18 to A.W.R. and D.C.S.H., a Cure Cancer and Cancer Australia grant (1186003 to R.T.), fellowships and grants from the Australian National Health and Medical Research Council (NHMRC: Project Grant 1143163 to M.E.R.; Program Grants 1016701 to D.C.S.H. and 1113577 to A.W.R.; Research Fellowships 1079560 to A.W.R., 1156024 to D.C.S.H., 1072662 to M.B.C. and 1104924 to M.E.R.; Investigator Grant 1174902 to A.W.R.), the Australian Research Council (Discovery Project 200102903 to M.E.R.), the Genomics Innovation Hub, Victorian State Government Operational Infrastructure Support and Australian Government NHMRC IRIISS.

Author contributions

L.T. developed the *FLAMES* package, performed experiments, conducted data analysis and wrote the manuscript. J.S.J. developed the FLT-seq protocol and performed experiments. R.T., M.A.A, H.P. and C.J.A.A collected patient samples and performed experiments related to *CLL2*. C.H.L., J.D.C., T.N. and A.C. performed experiments related to *MuSC*. Q.G., H.P. and A.L performed nanopore sequencing. S.L.A., H.K., X.D. and C.L. analyzed data and S.S. developed software. G.S.L, J.G.R., A.W.R. and D.C.S.H. facilitated the *MuSC* and *CLL2* studies and provided helpful comments and discussion. J.S.J, Q.G and M.B.C. planned the research and provided helpful comments and discussion. M.E.R. planned and supervised the research and wrote the manuscript. All authors read and approved the final manuscript.

Declaration of interests

The authors declare no competing interests.

Methods

Human Cell Lines

The cell culture and sample preparation of the *scmixology* cell lines are as previously described. Briefly the five cell lines (H2228, H838, H1975, HCC837, A549) were retrieved from the ATCC (<https://www.atcc.org/>) and cultured in Roswell Park Memorial Institute (RPMI) 1640 medium with 10% fetal calf serum and 1% penicillin-streptomycin. The cells were grown independently at 37°C with 5% carbon dioxide until near 100% confluency. The cells were then counted and mixed in equal numbers. The mix was used for FLT-seq library preparation. The first batch was derived from

the same sample featured in a previous study (*scmixology1*). The cell lines were cultured again using the same protocol to create a second batch (*scmixology2*) processed by both FLT-seq and scATAC-seq.

Mouse Muscle Stem Cells

Animals

All procedures were approved by the Animal Ethics Committee of The University of Melbourne and conformed to the Australian code of practice for the care and use of animals for scientific purposes as stipulated by the National Health and Medical Research Council of Australia. Mice were housed in the Biological Research Facility at The University of Melbourne under a 12h light-dark cycle, with drinking water and standard chow provided ad libitum. Pax7^{creERT2}R26-eYFP^{fl/fl} mice were generated from Pax7^{creERT2} and R26-eYFP^{fl/fl} founder mice strains^{39,40} both on a C57BL6 background.

Muscle Injury and Stem Cell Isolation

At 3 months of age, Pax7^{creERT2}; R26R^{YFP} mice received daily 100µL tamoxifen (20mg/ml in corn oil) for five days to label MuSCs with YFP. Two weeks after the final injection, animals were killed and hindlimb muscles excised and dissociated as described previously⁴¹. Cells were sorted on a FACS Aria III (BD Biosciences) with gating based on YFP.

To obtain activated MuSCs, mice were anaesthetized with isoflurane and muscles were injured with 1.2% barium chloride (Sigma). Lower hindlimbs received 40µL barium chloride via an intramuscular injection. Activated cells were isolated 72 hrs post injury while quiescent MuSCs were isolated from uninjured muscles as described previously.

Human CLL Sample

After providing written informed consent, the patient sample was collected after progression on venetoclax treatment⁴² (Human Research Ethics Committee approvals: Melbourne Health 2011.044, 2016.305; Peter MacCallum Cancer Centre 11/18; Walter and Eliza Hall Institute 05/04). Blood was collected in EDTA tubes and processed within 2 hours. Peripheral blood mononuclear cells (PBMCs) were isolated using Ficoll-Paque Plus (#17144002, lot:10258101, GE Healthcare) density gradient centrifugation and cells were cryopreserved. PBMCs were thawed, rested for 2 hours, and incubated with Fc Receptor blocking solution (Human TruStain FcX, Biolegend) for 10 minutes prior to staining with TotalSeq C antibodies (Biolegend) at 4 degrees for 30 min. PBMCs were washed three times and stained with propidium iodide (PI, Sigma). Viable cells (PI negative) were flow sorted using the FACS Aria (BD) and diluted to 1000 cells/µL.

FLT-seq

Single-cell capture and cDNA amplification for mouse skeletal muscle stem cell and cell lines was performed using 10x Genomics Chromium Single cell 3' Library and Gel Bead (v2 for *MuSC* and *scmixology1* and v3 for *scmixology2*) and for *CLL* sample using Chromium Single Cell 5' Library & Gel Bead Kit (v1) according to manufacturer instructions. Full-length cDNA generation for FLT-seq was carried out as described in detail at protocols.io⁴³. Briefly we followed the standard 10x genomic user guide, with RT time increased to 2 hours to increase reverse transcription of longer transcripts. After GEM-RT we transferred 10-20% volumes of GEMs into a new tube and

performed subsequent steps in parallel for both 10-20% and 80-90% subsample where each subsample is treated as a separate sample according to the 10x user guide. Volume of the Cleanup Mix was reduced accordingly. cDNA generation for hybrid capture input was similar to FLT-seq except that cDNA from the remaining sample (i.e. 80-90%, excluding the 10-20% subsample) was used for amplification using primers 10x Partial R1: CTACACGACGCTCTTCCGATCT and T5' PCR Primer IIA: AAGCAGTGGTATCAACGCAGAG in place of FPSfilA and RPSfilBr.

To capture the BCL2 gene, all annotated isoforms from Gencode and Ensembl data bases and isoforms identified in ~20% subsamples were integrated and used for IDT xGEN Lockdown capture probe design with one probe for each exon up to 1Kb. To design probes for exons shorter than 120 nucleotide they were concatenated to flanking exons and two probes were designed to include short exons and part of flanking exons. Hybrid capture was done with 2µg of cDNA input using IDT xGEN Hybridisation and Wash Kit. Captured cDNA were pulled down with streptavidin beads and washed with IDT buffers. The targeted cDNA library was amplified similar to the FLT-seq protocol for 12 cycles with Takara LA Taq DNA polymerase in place of PrimeSTAR GXL DNA Polymerase.

Standard single cell Illumina libraries were prepared according to 10x protocol. Illumina HiSeq2500 was used for sequencing *scmixology1* (2x125 cycles) and *MuSC* (2x150 cycles). Other libraries were sequenced on NextSeq 500 (1x 28 / 1x91 cycles plus 8 base index cycle) using the v2 150 cycle High Output kit (Illumina) as per the manufacturer's instructions. The base calling and quality scoring were determined using Real-Time Analysis on board software v2.4.6, while the FASTQ file generation and de-multiplexing utilised bcl2fastq conversion software v2.15.0.4. Full-length cDNA libraries from subsamples or capture cDNA were prepared using Oxford Nanopore Technologies SQK-LSK109 Ligation Sequencing Kit with the following modifications: incubation times for end-preparation and A-tailing were lengthened to 15 minutes, and all washes were performed with 1.8X Ampure beads to conserve smaller fragments. SFB was used for the final wash of the libraries. 50 fmol per library were sequenced on PromethION FLO-PRO002 R9.4.1 flow cells according to manufacturer protocols.

Single cell ATAC-seq (scATAC-seq)

The cells from five cell lines were counted and mixed equally. Cell nuclei were isolated and washed according to the Nuclei Isolation for Single Cell ATAC Sequencing (10x Genomics) protocol, with 1 million cells to start with (0.2 million from each cell line) and 3 minutes lysis on 100µL buffer. Nuclei were then used to generate scATAC-seq libraries according to the Chromium Single Cell ATAC Reagent Kits User Guide (10x Genomics; CG000168 Rev B) Sequencing libraries were loaded on an Illumina sequencer with 2 × 75 paired-end kits using the following read length: 72 bp read 1N, 8 bp i7 index, 16 bp i5 index and 72 bp read 2N. In the sequencing reaction, reads 1N and 2N contain the DNA insert, while the index reads, i5 and i7, capture the cell barcodes and sample indices, respectively. Cells were sequenced on Illumina HiSeq2500 with near around 300 million read pairs in total.

scATAC-seq data analysis

scATAC-seq sequencing data was demultiplexed, preprocessed and aligned with the default settings of the single cell ATAC *Cell Ranger* platform (1.0). The reference used

for alignment through the *Cell Ranger* platform was hg38. Next, *Picard tools* (<http://broadinstitute.github.io/picard/>) was used to remove the PCR duplicates. *Samtools*⁴⁴ (1.7) was used to extract read pairs that have mapping quality (MAPQ)>30, were nonmitochondrial and not chimerically mapped. *bedtools*⁴⁵ (v.2.26.0) was used to identify reads in mate pairs (i.e. fragments) and adjust the start of the paired reads to account for the 9bp region that the transposase enzyme occupies during transposition (i.e. +4 bp for + strand and -5 bp for - strand). Next, *Demuxlet* was used to identify the cell lines of each cell barcode using the genotypic information acquired in our previous benchmarking studies^{12,46}.

The count matrix was generated for each barcode separated BAM file using the *featureCounts* function in the *Rsubread* package⁴⁷ (1.32.4) in the *R* environment (v. 3.5.1). The annotation features were promoter regions (i.e. TSS - 500 bp to TSS + 200 bp) corresponding to isoforms identified by the various long-read analysis methods (i.e. *FLAMES*, *TALON*, *FLAIR*, *StringTie2*). To determine the background of scATAC-seq and identify the open promoter, we performed analysis on randomized TSS, where a random position in each gene body was used as the TSS. The 90% percentile of the fragment count around random TSS (7.64) was used as the threshold to determine the open promoters and is annotated in Figures 1F and S4D.

For Figure 3I, scATAC-seq coverage was calculated from the aligned BAM files using *compute_coverage* in the *plyranges* package⁴⁸ (1.7.14) and visualised using *view_coverage* in the *superintronic*⁴⁹ package (0.99.4).

Illumina short-read data analysis

The fastq data were processed by *scPipe* to generate a gene count matrix for all samples except *CLL2*, which was processed by *Cell Ranger* (3.0.0) to generate the antibody and gene count matrix. Each gene count matrix was used as input to the standard *Seurat* pipeline with normalization performed by *SCTransform*⁵⁰. Clustering was performed for the *MuSC* and *CLL2* samples with resolution equal to 0.6. The cell line annotation for *scmixology* was acquired using *Demuxlet* with the same parameters as our previous benchmark study. Integration of the *scmixology1* and *scmixology2* datasets was performed using *Seurat*. The clustering results and cell line annotation is shown in Figure S1.

Nanopore sequencing and data preprocessing

We performed basecalling on the raw fast5 data using *Guppy* (1.8.1 for *MuSC* sample and 3.1.5 for *scmixology* and *CLL2*) from Oxford Nanopore Technologies. For each read, we locate the barcode sequence by searching for the flanking sequence before the cell barcode. The cell barcodes identified from the short-read data provide a reference to search for and trim in the long reads. An edit distance of up to 2 is allowed during cell barcode matching. Reads that failed to match any cell barcode were discarded. Sequences following the cell barcode were used as UMIs and trimmed. For the 3' end protocol, the polyA tail after the UMI sequence was trimmed and sequences after the polyA tail were kept. The cell barcode and UMI sequence was integrated into the fastq read header as per *scPipe*⁵¹. The processed fastq was used as input for genome alignment and further analysis.

Detection and quantification of isoforms

Reads were aligned to the genome by *minimap2*⁵² (-ax splice --junc-bonus 1 -k14 --secondary=no --junc-bed) using Gencode reference (human hg38.v33, mouse mm10.vM24). *FLAMES* summarizes the alignment for each read by grouping reads with similar splice junctions (<5bp) to get a raw isoform annotation. The raw isoform annotation is compared against the reference annotation to correct potential splice site and transcript start/end errors. Transcripts that have similar splice junctions (<5bp) and transcript start/end (<100bp) to the reference transcript were merged with the reference. This process will also collapse isoforms that are likely to be truncated transcripts. Next, the sequence of each polished transcript was extracted and used as the updated reference. The reads were realigned to this reference by *minimap2*. The transcript coverage of individual reads is summarized in Figure S3. We noticed that the *scmixology2* data contained more reads that were not full-length, which might relate to the difference in sample preparation time or the v2 and v3 10x chemistry. The transcripts with fewer than 5 full-length aligned reads (>95% coverage) were discarded. The reads were assigned to transcripts based on both alignment score, fractions of reads aligned and transcript coverage. Reads that cannot be uniquely assigned to transcripts or had low transcript coverage (<60%) were discarded. The UMI transcript count matrix was generated by collapsing the reads with the same UMI in a similar way to what is done for short-read scRNA-seq data, but allowing for an edit distance of up to 2. Downsampling was performed by randomly selecting reads in bam files that aligned to the genome according to the sampling ratio. The counts of transcripts from the same gene were aggregated to generate the gene-level UMI count and compared to the gene count generated from the short-read data in Figure 1E. *FLAMES* is written in python and uses other packages, including *pysam*⁴⁴ (0.15.2), *numpy*⁵³ (1.14.2) and *editdistance* (0.5.3) (pypi.org/project/editdistance).

Comparison of *FLAMES* to other tools

Direct RNA sequencing data from SIRV spike-in E2 mix which contains 69 synthetic isoform transcripts (from 7 SIRV genes) was downloaded from NCBI (SRX3204589). Alignment was performed using *minimap2* (2.17), with “-ax splice --splice-flank=no -k14 --secondary=no” and “--junc-bed”. For *TALON*, the mapped reads were processed using *TranscriptClean* (1.02) to correct for mismatches and microindels. Following correction, long reads were collapsed into a transcript isoform quantification table in *TALON* (4.1), using the SIRV annotation. We ran the *FLAIR* pipeline (1.4) using default parameters with the SIRV annotation supplied to aid isoform detection. We ran *StringTie2* (2.0.4) with “-L -G -c 10” and uses the SIRV annotation. For *FLAMES*, we used the default parameters except “strand_specific:1” and we filtered transcripts with at least 10 reads. For comparison, transcripts generated by *TALON* and *FLAIR* were also filtered to have a read count of at least 10. The *scmixology1* data was processed in a similar way to the SIRV data. Gencode human hg38.v33 was used as the gene reference annotation for each method. *TALON* was run in parallel on each chromosome to reduce compute time and the results were aggregated later.

Isoform classification and splicing analysis

SQANTI2 (4.1, <https://github.com/Magdoll/SQANTI2>) was used to compare the transcripts identified to the reference with parameter “-g --cage_peak --coverage --force_id_ignore”. We used the FANTOM5 cage peak dataset on hg19 and mm9 and lifted these to the hg38 and mm10 reference using UCSC’s *liftOver* tool (<https://genome.ucsc.edu/cgi-bin/hgLiftOver>). The isoform classification was extracted from the *SQANTI2* result and plotted in Figure 2B. The *gffcompare*⁵⁴ (0.11.2) program

was used with parameter “-T -R -M” to compare isoform annotations generated from different samples (Figure 2B). It was also used to compare isoform annotations obtained after downsampling (Figure S5B). Results from these comparisons were plotted using *UpsetR*⁵⁵ (1.4.0). We ranked transcript abundance for each gene that had multiple isoforms and obtained the alternative splicing events from the most expressed transcript and the second most expressed transcript. We used a common model to classify the splicing events⁵⁶, where alternative 5/3' splice site includes alternative promoter and alternative polyadenylation. Transcripts with more than one splicing event were classified as *complex splicing changes* (Figure 2G).

The binary splicing analysis was performed for each multi-isoform gene by calculating the percentage of cells that only expressed the major isoform (isoform with highest abundance) or only expressed other isoforms. The null distribution for each gene assumes each cell randomly chooses isoforms and the chance of picking the major isoform is proportional to the major isoform expression p_m , calculated by dividing the major isoform's total UMI count by the total gene UMI count across all cells. Next, we assume the count of the major isoform follows a binomial distribution $B(n, p_m)$ where n is the UMI gene count for a cell. Therefore, the probability of binary splicing is $\Pr(0, n, p_m) + \Pr(n, n, p_m)$. We used the UMI gene count for each cell $\Pr(0, n, p_m) + \Pr(n, n, p_m)$ take the average. In Figure S5A, S5C we fit the relationship between the average binary splicing probability and total gene count (black lines) using *geom_smooth()* in *ggplot2*⁵⁶ (3.3.1) which uses generalised additive models.

Differential transcript usage analysis

We filtered genes to have at least two isoforms, each with more than 15 UMI counts. For each gene, the per cell transcript counts were merged by group to generate pseudo bulk samples. For scmixology the groups are based on cell line identify inferred by known genetic variation, and for the MuSC and *CLL2* data the groups are based on the clusters acquired from *Seurat* clustering, shown in Figure S1. The top 2 highly expressed transcripts for each group were selected and a UMI count matrix where the rows are selected transcripts and columns are groups was used as input to a chi-square test of independence (*chisq.test* in R). *P*-values were adjusted by Benjamini–Hochberg correction⁵⁷ and results were summarized in Table S2. We performed functional clustering with DAVID⁵⁸ using genes with significant DTU (adjusted *P*-value < 0.01) as input (Table S3). The transcript structures in all figures were plotted using *geom_alignment* in *ggbio*⁵⁹ (1.36.0). We performed imputation of transcript counts on cells that were not sampled for long-read sequencing using the shared nearest neighbour network constructed by *Seurat*⁶⁰ (3.1.5). We then scaled the transcript expression matrix for each gene and the results were used in heatmaps and UMAP visualizations (available in *Seurat*) in Figure 3 and Figure S6. The imputed results were used for visualization purposes only.

Variant calling and clonal analysis

First, we identified candidate SNVs using *FLAMES* by excluding homopolymer regions (runs >3 of the same nucleotides), positions with coverage of fewer than 100 reads and positions with reference allele frequency less than 10% or greater than 90%. For each candidate SNV, we generated an allele count matrix of the reference and alternative allele. Next, we collect the cells that have reads with the reference allele and the alternative allele and performed a binomial test on the allele counts in a similar way as the binary splicing analysis, assuming that under the null hypothesis each cell

has the same probability of having the alternative and reference allele. P -values were adjusted by Benjamini–Hochberg correction and candidate SNVs with an adjusted P -value < 0.05 were kept for further analysis. PCA was applied to the filtered alternative allele count matrix from the *scmixology* data and first two PCs were plotted in Figure 4B. The top 5 PCs were selected to build the shared neighbour network (*scrans::buildSNNGraph*⁶¹ $k=20$, $d=5$) which is used for Louvain clustering (*igraph::cluster_louvain*⁶²) in order to examine whether the allele count matrix captured the variants in the cell lines. Next, we performed differential allele frequency analysis in a similar way to the DTU analysis, but with the allele counts aggregated for each cluster. The candidate SNVs with adjusted P values < 0.05 were selected and compared to the VCF reference generated from a previous study using bulk RNA-seq. After analyzing the *scmixology* dataset, we processed the *CLL2* data in the same way and conducted differential allele frequency analysis on the *CLL2* clusters shown in Figure S1. The bar plot of cells with alternative alleles of significant SNVs were shown in Figure S7B and summarized in Figure 4D. The capture enrichment data was analyzed in similar way, with the reads processed by *FLAMES* and an allele count matrix generated for the Gly101Val mutation. All plots were generated using *ggplot2* unless otherwise specified and most of the analysis was performed in R ⁶³ (4.0) unless otherwise specified.

Data and code availability

Raw data are available from GEO under accession numbers GSE126906 and GSE154869 (*scmixology1*), GSE154870 and GSE142285 (*scmixology2*), GSE154868 (*MuSC*) and the patient data (*CLL2*) is available from EGA. Refer to Table S1 for a summary of these datasets. The *FLAMES* source code is available from <https://github.com/LuyiTian/FLAMES>. The processed data and scripts used in this study are available from https://github.com/LuyiTian/FLTseq_data.

References

1. Han, X. *et al.* Construction of a human cell landscape at single-cell level. *Nature* **581**, 303–309 (2020).
2. Shalek, A. K. *et al.* Single-cell transcriptomics reveals bimodality in expression and splicing in immune cells. *Nature* **498**, 236–240 (2013).
3. Song, Y. *et al.* Single-Cell Alternative Splicing Analysis with Expedition Reveals Splicing Dynamics during Neuron Differentiation. *Mol. Cell* **67**, 148–161.e5 (2017).
4. Hagemann-Jensen, M. *et al.* Single-cell RNA counting at allele and isoform resolution using Smart-seq3. *Nat. Biotechnol.* **38**, 708–714 (2020).
5. Zheng, G. X. Y. *et al.* Massively parallel digital transcriptional profiling of single cells. *Nat. Commun.* **8**, 14049 (2017).
6. Gupta, I. *et al.* Single-cell isoform RNA sequencing characterizes isoforms in thousands of cerebellar cells. *Nat. Biotechnol.* 364950 (2018). doi:10.1038/nbt.4259
7. Singh, M. *et al.* High-throughput targeted long-read single cell sequencing reveals the clonal and transcriptional landscape of lymphocytes. *Nat. Commun.* **10**, 3120 (2019).
8. Lebrigand, K., Magnone, V., Barbry, P. & Waldmann, R. High throughput, error corrected Nanopore single cell transcriptome sequencing. *bioRxiv* **1**, (2019).
9. Volden, R. & Vollmers, C. Highly Multiplexed Single-Cell Full-Length cDNA Sequencing of human immune cells with 10X Genomics and R2C2. *bioRxiv* (2020). doi:10.1101/2020.01.10.902361
10. Tang, A. D. *et al.* Full-length transcript characterization of SF3B1 mutation in chronic lymphocytic leukemia reveals downregulation of retained introns. *Nat. Commun.* **11**, 1438 (2020).
11. Wyman, D. *et al.* A technology-agnostic long-read analysis pipeline for transcriptome discovery and quantification. *bioRxiv* (2019).
12. Tian, L. *et al.* Benchmarking single cell RNA-sequencing analysis pipelines using mixture control experiments. *Nat. Methods* **16**, 479–487 (2019).
13. Garalde, D. R. *et al.* Highly parallel direct RNA sequencing on an array of nanopores. *Nat. Methods* **15**, 201–206 (2018).
14. Kovaka, S. *et al.* Transcriptome assembly from long-read RNA-seq alignments with StringTie2. *Genome Biol.* **20**, 278 (2019).
15. Soneson, C. *et al.* A comprehensive examination of Nanopore native RNA sequencing for characterization of complex transcriptomes. *Nat. Commun.* **10**, 3359 (2019).
16. Forrest, A. R. R. *et al.* A promoter-level mammalian expression atlas. *Nature* **507**, 462–470 (2014).
17. Tardaguila, M. *et al.* SQANTI: extensive characterization of long-read transcript sequences for quality control in full-length transcriptome identification and quantification. *Genome Res.* 1–16 (2018). doi:10.1101/gr.222976.117
18. Yue, L., Wan, R., Luan, S., Zeng, W. & Cheung, T. H. Dek Modulates Global Intron Retention during Muscle Stem Cells Quiescence Exit. *Dev. Cell* **53**, 661–676.e6 (2020).
19. Faigenbloom, L. *et al.* Regulation of alternative splicing at the single-cell level. *Mol. Syst. Biol.* **11**, 845 (2015).

20. Buen Abad Najar, C. F., Yosef, N. & Lareau, L. F. Coverage-dependent bias creates the appearance of binary splicing in single cells. *Elife* **9**, 2019.12.19.883256 (2020).
21. Manley, J. L. & Krainer, A. R. A rational nomenclature for serine/arginine-rich protein splicing factors (SR proteins). *Genes Dev.* **24**, 1073–1074 (2010).
22. Gazda, H. T. *et al.* Ribosomal Protein S24 Gene Is Mutated in Diamond-Blackfan Anemia. *Am. J. Hum. Genet.* **79**, 1110–1118 (2006).
23. Zhang, H. *et al.* Comparative transcriptome analysis reveals that the extracellular matrix receptor interaction contributes to the venous metastases of hepatocellular carcinoma. *Cancer Genet.* **208**, 482–491 (2015).
24. Dawes, R. *et al.* Combinations of CD45 Isoforms Are Crucial for Immune Function and Disease. *J. Immunol.* **176**, 3417–3425 (2006).
25. Ray, T. A. *et al.* Comprehensive identification of mRNA isoforms reveals the diversity of neural cell-surface molecules with roles in retinal development and disease. *Nat. Commun.* **11**, 3328 (2020).
26. Chen, C., Zhao, S., Karnad, A. & Freeman, J. W. The biology and role of CD44 in cancer progression: Therapeutic implications. *J. Hematol. Oncol.* **11**, 1–23 (2018).
27. Demircioğlu, D. *et al.* A Pan-cancer Transcriptome Analysis Reveals Pervasive Regulation through Alternative Promoters. *Cell* **178**, 1465–1477.e17 (2019).
28. Stuart, T. *et al.* Comprehensive Integration of Single-Cell Data. *Cell* **177**, 1888–1902.e21 (2019).
29. Rang, F. J., Kloosterman, W. P. & de Ridder, J. From squiggle to basepair: computational approaches for improving nanopore sequencing read accuracy. *Genome Biol.* **19**, 90 (2018).
30. Cretu Stancu, M. *et al.* Mapping and phasing of structural variation in patient genomes using nanopore sequencing. *Nat. Commun.* **8**, 1326 (2017).
31. Amarasinghe, S. L. *et al.* Opportunities and challenges in long-read sequencing data analysis. *Genome Biol.* **21**, 30 (2020).
32. Kang, H. M. *et al.* Multiplexed droplet single-cell RNA-sequencing using natural genetic variation. *Nat. Biotechnol.* **36**, 89–94 (2018).
33. Blombery, P. *et al.* Acquisition of the Recurrent Gly101Val Mutation in BCL2 Confers Resistance to Venetoclax in Patients with Progressive Chronic Lymphocytic Leukemia. *Cancer Discov.* **9**, 342–353 (2019).
34. Volden, R. *et al.* Improving nanopore read accuracy with the R2C2 method enables the sequencing of highly multiplexed full-length single-cell cDNA. *Proc. Natl. Acad. Sci.* **115**, 9726–9731 (2018).
35. Dong, X. *et al.* The long and the short of it: unlocking nanopore long-read RNA sequencing data with short-read tools. *bioRxiv* 2020.06.28.176727 (2020). doi:10.1101/2020.06.28.176727
36. Westoby, J., Herrera, M. S., Ferguson-Smith, A. C. & Hemberg, M. Simulation-based benchmarking of isoform quantification in single-cell RNA-seq. *Genome Biol.* **19**, 191 (2018).
37. Buccitelli, C. & Selbach, M. mRNAs, proteins and the emerging principles of gene expression control. *Nat. Rev. Genet.* (2020). doi:10.1038/s41576-020-0258-4
38. Melamud, E. & Moulton, J. Stochastic noise in splicing machinery. *Nucleic Acids Res.* **37**, 4873–4886 (2009).
39. Murphy, M. M., Lawson, J. A., Mathew, S. J., Hutcheson, D. A. & Kardon, G. Satellite cells, connective tissue fibroblasts and their interactions are crucial for

- muscle regeneration. *Development* **138**, 3625–3637 (2011).
40. Srinivas, S. *et al.* Cre reporter strains produced by targeted insertion of EYFP and ECFP into the ROSA26 locus. *BMC Dev. Biol.* **1**, 1–8 (2001).
 41. Liu, L., Cheung, T. H., Charville, G. W. & Rando, T. A. Isolation of skeletal muscle stem cells by fluorescence-activated cell sorting. *Nat. Protoc.* **10**, 1612–1624 (2015).
 42. Roberts, A. W. *et al.* Targeting BCL2 with Venetoclax in Relapsed Chronic Lymphocytic Leukemia. *N. Engl. J. Med.* **374**, 311–322 (2016).
 43. Jabbari, J. S. & Tian, L. Massively parallel long-read sequencing of single cell RNA isoforms. *protocols.io* (2019). doi:10.17504/protocols.io.8d9hs96
 44. Li, H. *et al.* The Sequence Alignment/Map format and SAMtools. *Bioinformatics* **25**, 2078–2079 (2009).
 45. Quinlan, A. R. & Hall, I. M. BEDTools: A flexible suite of utilities for comparing genomic features. *Bioinformatics* **26**, 841–842 (2010).
 46. Holik, A. Z. *et al.* RNA-seq mixology: Designing realistic control experiments to compare protocols and analysis methods. *Nucleic Acids Res.* **45**, 1–18 (2017).
 47. Liao, Y., Smyth, G. K. & Shi, W. FeatureCounts: An efficient general purpose program for assigning sequence reads to genomic features. *Bioinformatics* **30**, 923–930 (2014).
 48. Lee, S., Cook, D. & Lawrence, M. plyranges: a grammar of genomic data transformation. *Genome Biol.* **20**, 4 (2019).
 49. Lee, S. *et al.* Covering all your bases: incorporating intron signal from RNA-seq data. *bioRxiv* 352823 (2019). doi:10.1101/352823
 50. Hafemeister, C. & Satija, R. Normalization and variance stabilization of single-cell RNA-seq data using regularized negative binomial regression. *Genome Biol.* **20**, 296 (2019).
 51. Tian, L. *et al.* scPipe: A flexible R/Bioconductor preprocessing pipeline for single-cell RNA-sequencing data. *PLOS Comput. Biol.* **14**, e1006361 (2018).
 52. Li, H. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics* **34**, 3094–3100 (2018).
 53. van der Walt, S., Colbert, S. C. & Varoquaux, G. The NumPy Array: A Structure for Efficient Numerical Computation. *Comput. Sci. Eng.* **13**, 22–30 (2011).
 54. Pertea, G. & Pertea, M. GFF Utilities: GffRead and GffCompare. *F1000Research* **9**, 304 (2020).
 55. Conway, J. R., Lex, A. & Gehlenborg, N. UpSetR: an R package for the visualization of intersecting sets and their properties. *Bioinformatics* **33**, 2938–2940 (2017).
 56. Frankiw, L., Baltimore, D. & Li, G. Alternative mRNA splicing in cancer immunotherapy. *Nat. Rev. Immunol.* (2019). doi:10.1038/s41577-019-0195-7
 57. Benjamini, Y. & Hochberg, Y. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *J. R. Stat. Soc. Ser. B* **57**, 289–300 (1995).
 58. Huang, D. W., Sherman, B. T. & Lempicki, R. A. Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. *Nat. Protoc.* **4**, 44–57 (2009).
 59. Yin, T., Cook, D. & Lawrence, M. ggbio: an R package for extending the grammar of graphics for genomic data. *Genome Biol.* **13**, R77 (2012).
 60. Satija, R., Farrell, J. A., Gennert, D., Schier, A. F. & Regev, A. Spatial reconstruction of single-cell gene expression data. *Nat. Biotechnol.* **33**, 495–

- 502 (2015).
61. L. Lun, A. T., Bach, K. & Marioni, J. C. Pooling across cells to normalize single-cell RNA sequencing data with many zero counts. *Genome Biol.* **17**, 75 (2016).
 62. Csardi, G. & Nepusz, T. The igraph software package for complex network research. *InterJournal Complex Sy*, 1695 (2006).
 63. R Core Team. R: A Language and Environment for Statistical Computing. (2019).

Chapter 5

Concluding remarks

In this thesis, we developed a pipeline for data preprocessing (Chapter 2), performed experiments to generate a benchmarking dataset that was used to compare the performance of many different computational methods (Chapter 3) and developed a new protocol that combines scRNAseq and long-read sequencing together with a novel data analysis pipeline (Chapter 4).

The field of single cell RNA sequencing has developed rapidly in the past few years which has resulted in many different protocols for capturing and sequencing mRNA from single cells. In 2018 there were limited options available for preprocessing the raw data that these different technologies generated in a unified way. The scRNAseq data preprocessing pipeline (*scPipe*) that we developed [79] filled this gap, enabling the flexibility to handle data arising from different protocols, including both plate and droplet-based methods. Processing the data generated by different protocols in a consistent way makes quality control between different samples easier. *scPipe* is an open-source R package that is publicly available from the Bioconductor project (<https://bioconductor.org/packages/scPipe/>). Since publication, this work has been cited 33 times (Source: Google Scholar, 29 Aug 2020) and has been used by other researchers to process their scRNAseq data. *scPipe* provided the basic data preprocessing infrastructure that was used in the remainder of this thesis.

We next created data resources to facilitate the systematic benchmarking of scRNAseq analysis methods. The availability of ground truth, which refers to independent cell labels in this context, across multiple datasets makes it easier to compare the performance

of different scRNAseq analysis methods. For example, to compare clustering algorithms we can use the cell identity labels to compare the solutions generated, and to compare data integration methods we also need multiple datasets that include the same populations processed by different scRNAseq protocols. To meet this need, we generated a comprehensive benchmarking dataset that included ground truth across multiple technologies that followed a mixture design, as used in the past in microarray and RNAseq studies. We designed a series of mixture experiments that involved mixing either cells or mRNAs in different combinations or proportions to create known structure in the datasets. Next we used these datasets to compare different data analysis methods for scRNAseq, ranging from normalization and imputation to clustering, trajectory analysis and data integration. Of note, we not only compared the performance of methods for individual tasks, but also compared combinations of methods at the pipeline-level to reveal pipelines suited to different types of data for different tasks [80]. Since publication, this study has been cited 55 times (Source: Google Scholar, 29 Aug 2020), with the datasets used in other benchmarking studies to evaluate the performance of different computational methods, including data preprocessing [81], imputation [82] and cell type annotations [83]. These data have also been used by method developers to compare and validate new algorithms, such as *scAlign* [84], *scclusteval* [85], *ROGUE* [86], *scIGANs* [87], *TooManyCells* [88], *DecontX* [89] and *CellAssign* [90]. This benchmark design could be extended to generate new datasets to compare methods for other single cell genomics applications, such as single cell multi-omics.

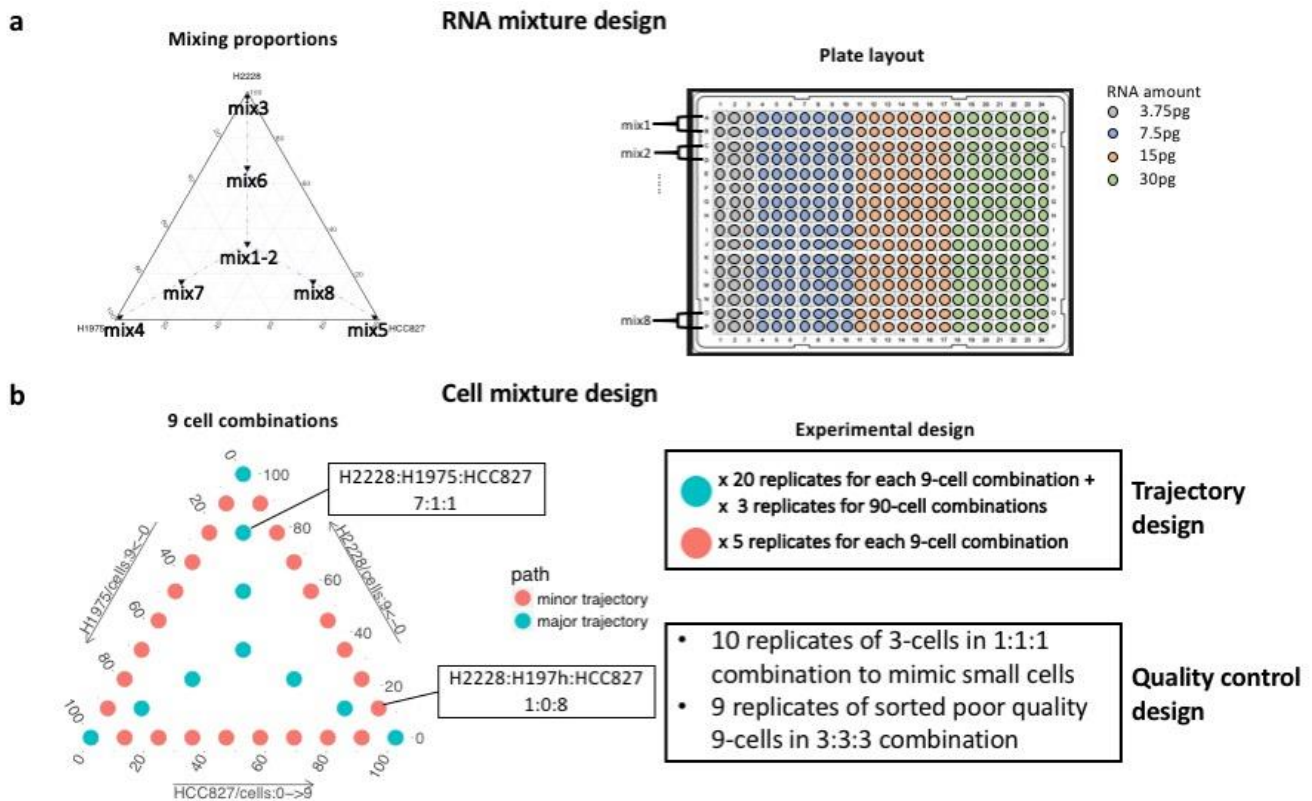
Recent improvements in long-read sequencing technology from Oxford Nanopore Technologies and Pacific Biosciences has seen increases in the throughput and accuracy of these platforms, making it possible to quantify gene expression at the transcript-level. A small number of studies that have combined scRNAseq and long-read sequencing to detect and quantify isoforms at the single cell level. However, further development of both the protocols and computational methods are required to broaden the applicability of this approach, motivating the final chapter of this thesis. The FLT-seq protocol samples the droplets containing cells generated by the popular 10X scRNAseq method for long-read sequencing technologies, which is combined with the advantages of throughput and accuracy from short-read sequencing on all cells to identify cell barcodes and perform clustering. The open-source computational pipeline (*FLAMES*, available from <https://github.com/LuyiTian/FLAMES>) was developed to process the long-read data

to facilitate isoform analysis and mutation detection. We generated multiple datasets using FLT-seq from different species and cell types, profiling isoforms at the single cell level. These data highlighted many common features that were shared between different cell type and species, such as conserved functional modules that were enriched for alternative transcript usage in different cell populations, including ribosome biogenesis and mRNA splicing. There have been several other studies published recently on this topic, some of which apply long-read sequencing to spatial transcriptomics [17, 91], while others focused on capture enrichment of specific target genes [92]. This work will hopefully encourage more researchers to investigate transcript-level heterogeneity in single cells.

In summary, we have developed computational tools, datasets and a new protocol that enables researchers to make better use of scRNAseq technology to generate novel biological insights. The tools were built around scRNAseq, but could also be extended to other single-cell omics protocols. The *scPipe* package implemented a general and versatile pipeline for scRNAseq data preprocessing and can be extended to other modalities such as single-cell ATACseq. The experimental design for our *scmixology* study could be extended to single-cell multi-omics technologies which currently lack gold-standard benchmarking datasets to support methods comparisons and development. And finally, the FLT-seq protocol could be adapted to work with the recently released 10X multi-ome kits which allow the generation of scATACseq and scRNAseq profiles simultaneously on the same cells.

Appendix A

Additional Material for Chapter 3

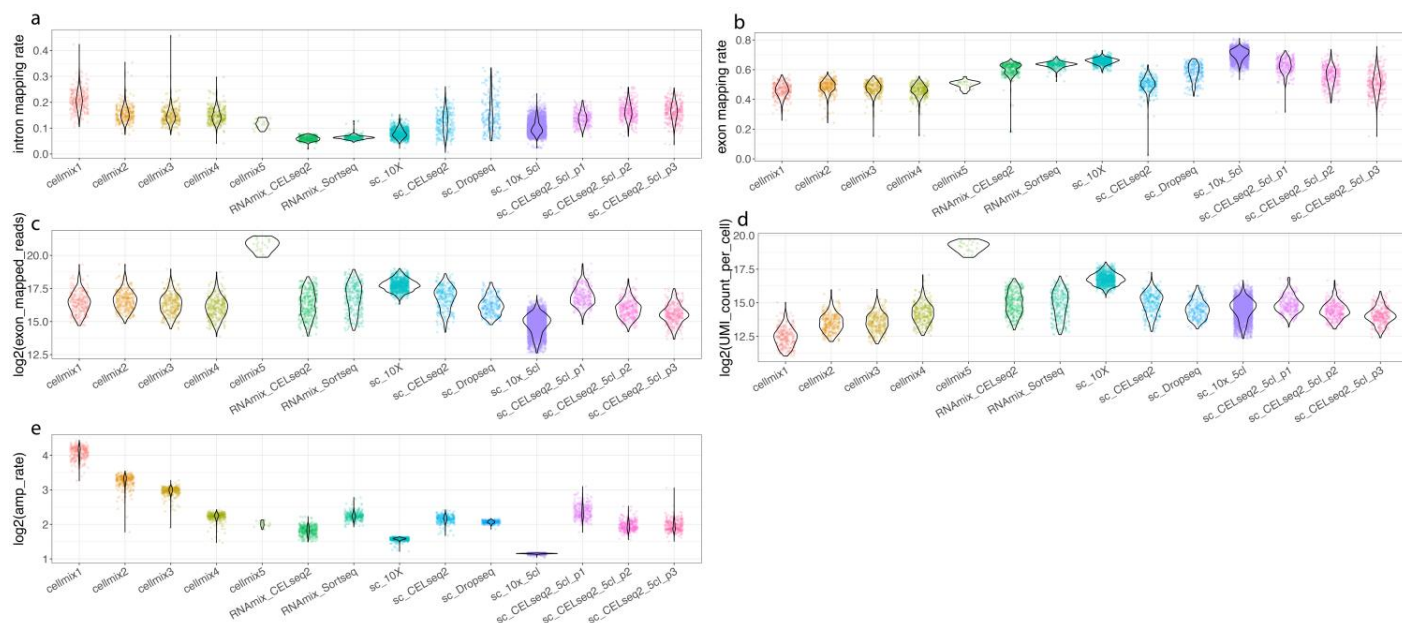


Supplementary Figure 1

Overview of the *RNA mixture* and *cell mixture* experimental designs.

(a) Ternary plot and plate layout of the *RNA mixture* design, which involved extracting RNA from 3 cell lines (H2228, H1975 and HCC827) in bulk, and mixing it in known proportions to get 8 mixtures that were then diluted to single cell equivalent amounts ranging from 3.75pg to 30pg.

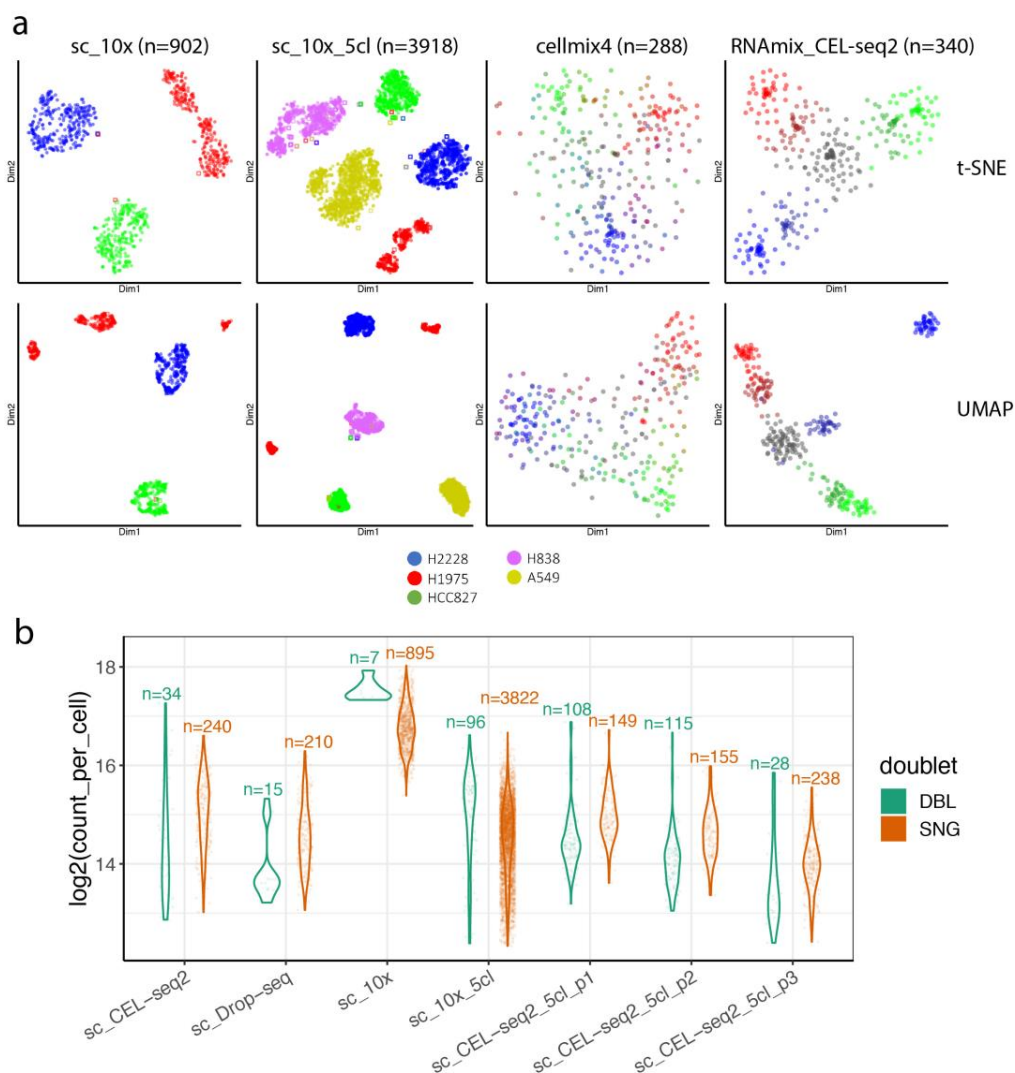
(b) Ternary plot of *cell mixtures*. Various 9 cell combinations were obtained using the same 3 cell lines. The number of replicates for each combination varies, as does the number of low quality samples included.



Supplementary Figure 2

Violin plots of quality control metrics for cells from each benchmarking dataset.

(a) The proportion of reads that map to introns. (b) The proportion of reads that map to exons. (c) The number of reads that map to exons. (d) The total number of counts per cell after UMI deduplication. (e) The amplification rate, which is defined by the ratio between the reads mapping to exons and the UMI counts after UMI deduplication. This measure reflects the library complexity. The sample size for each dataset is shown in Supplementary Table 1.

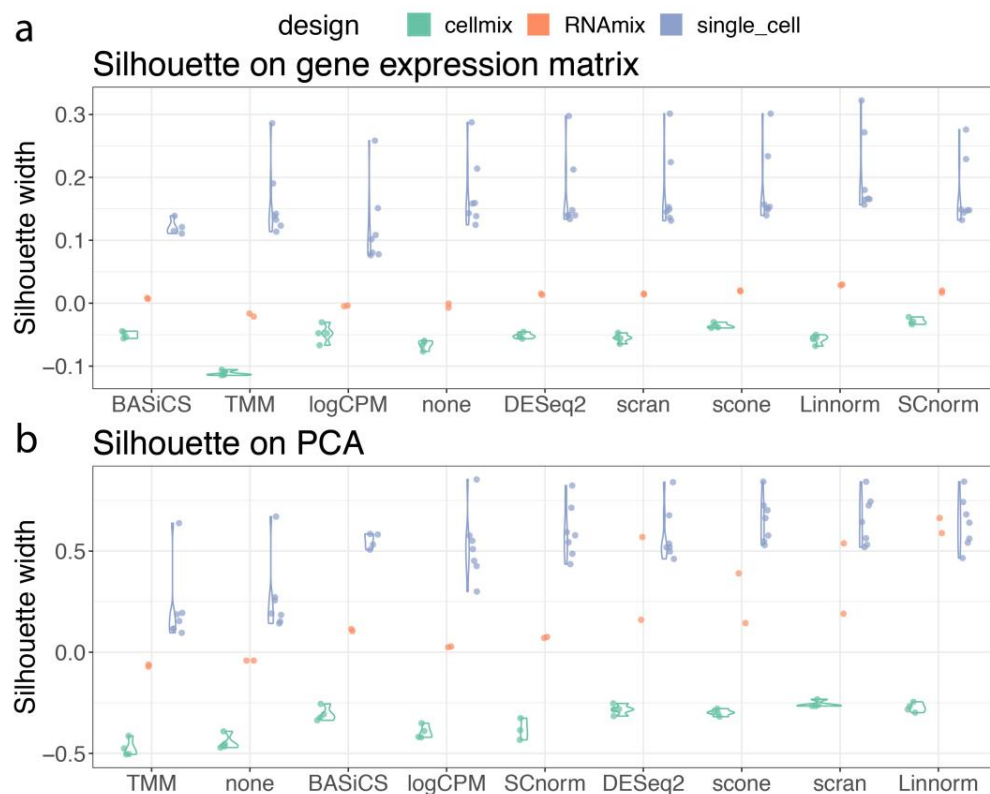


Supplementary Figure 3

Visualization of representative benchmarking datasets using t-SNE and UMAP and violin plot of the number of doublets.

(a) t-SNE and UMAP visualizations of 4 datasets. From left to right: 10X single cell using 3 cell lines; 10X single cell using 5 cell lines; a CEL-seq2 cell mixture and a CEL-seq2 RNA mixture. Each point represents a cell or 'pseudo cell' and the number of cells in each plot is indicated in the title.

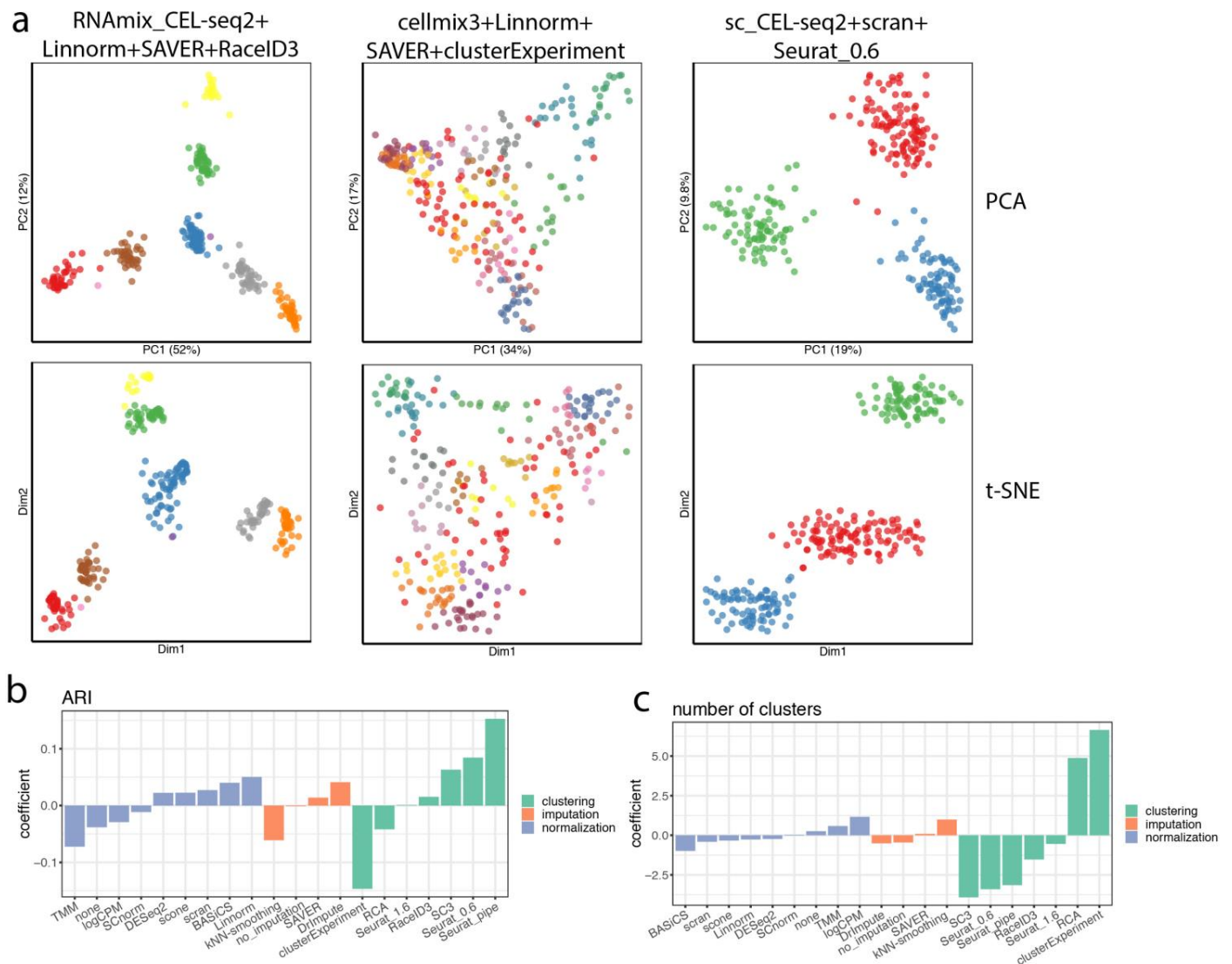
(b) Violin plot of doublets in each dataset, identified using Demuxlet (DBL: doublet, SNG: single cell). The number of single cells and doublets are shown on top of each violin. Doublets were excluded when calculating the performance metrics.



Supplementary Figure 4

Violin plots of silhouette widths for different normalization methods.

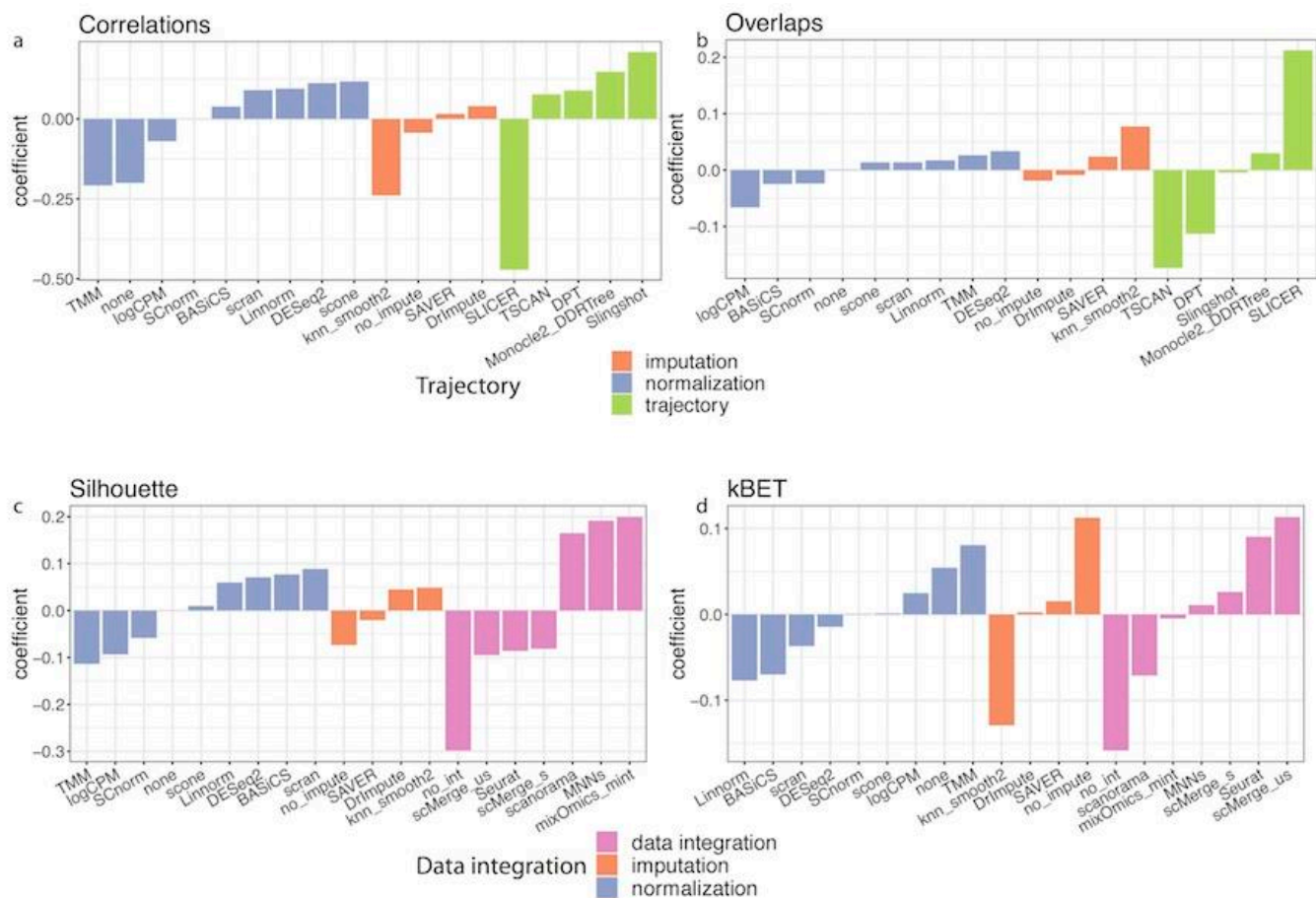
Silhouette widths calculated using the known biological groups after data have been normalized by different methods. The input to the silhouette width calculation is the distance between cells, which have been calculated using either **(a)** the gene expression matrix with the 1,000 most highly variable genes or **(b)** the first two PCs obtained from PCA. The sample sizes in this plots are the same as shown in Figure 2A.



Supplementary Figure 5

Example clustering results and summary of clustering performance using ARI and the number of clusters.

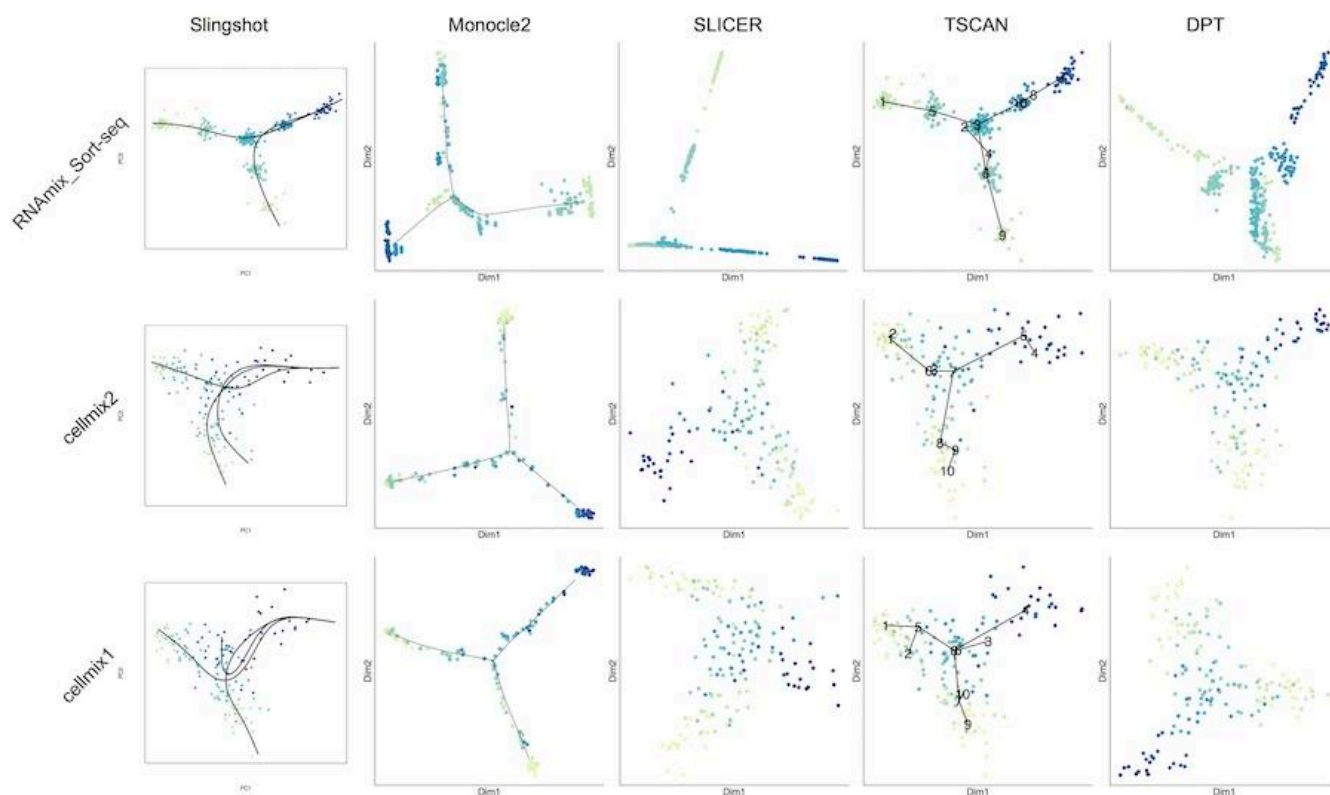
(a) Examples of clustering results visualized by PCA (top) and t-SNE (bottom), with different colours representing the cluster assignments made by the selected method. The sample sizes are 340 for *RNAmix_CEL-seq2*, 285 for *cellmix3* and 274 for *sc_CEL-seq2*. Coefficients obtained from linear models fitted using the ARI (b) or the number of clusters (c) as dependent variables, and experimental design, normalization methods, imputation methods and clustering methods as covariates. The coefficients measure whether particular features have positive or negative associations with the dependent variables.



Supplementary Figure 6

Coefficients from linear models used to quantify the impact different methods have on the trajectory analysis and data integration results.

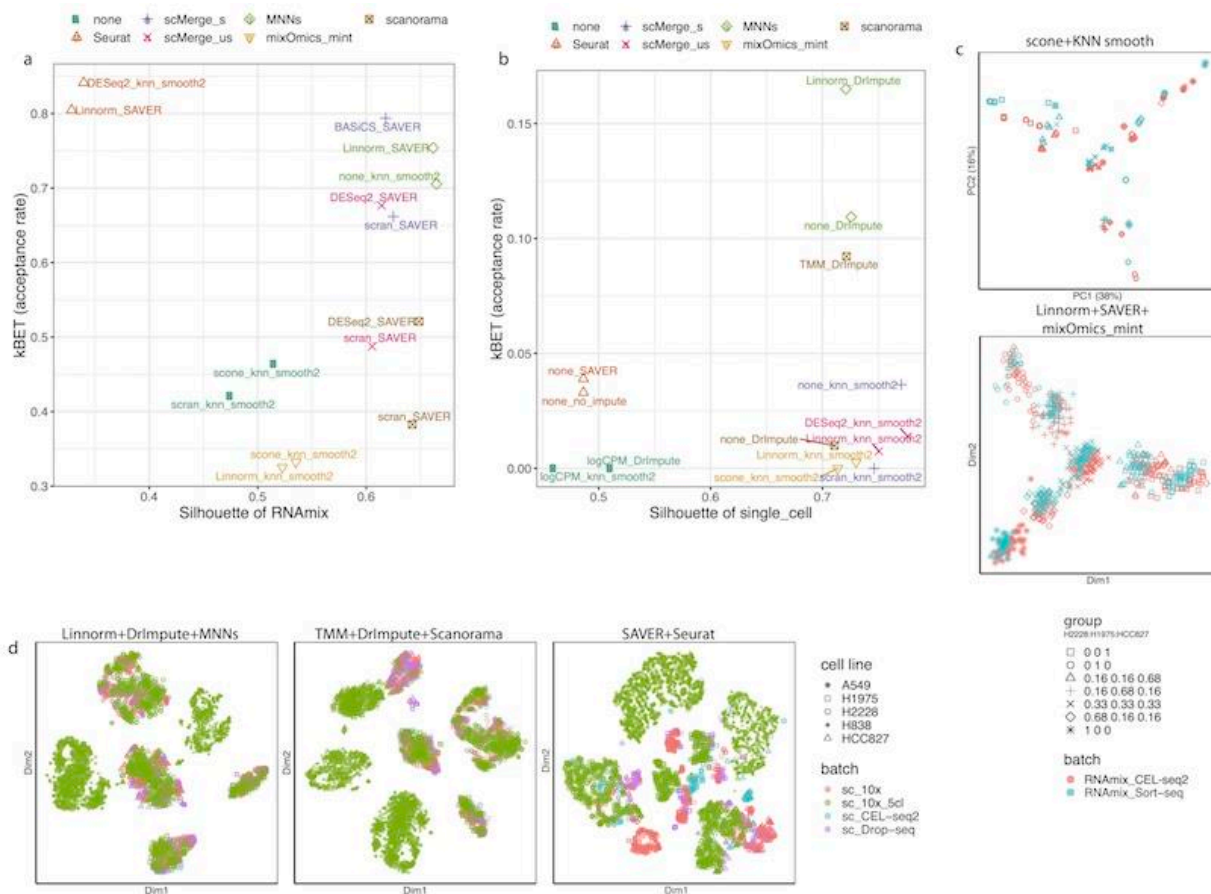
Linear models were fitted using the evaluation metrics as dependent variables, with experimental design, normalization methods, imputation methods and either trajectory analysis or data integration methods as covariates. Positive coefficients indicates that a method is positively associated with the performance metrics. The evaluation metrics used as dependent variables for each plot were: (a) the correlations between calculated pseudotime and ground truth; (b) the overlap between the calculated trajectory and the known trajectory; (c) the average silhouette width of the known groups and (d) the kBET acceptance rate.



Supplementary Figure 7

Visualization of results from the trajectory analysis methods evaluated in our study.

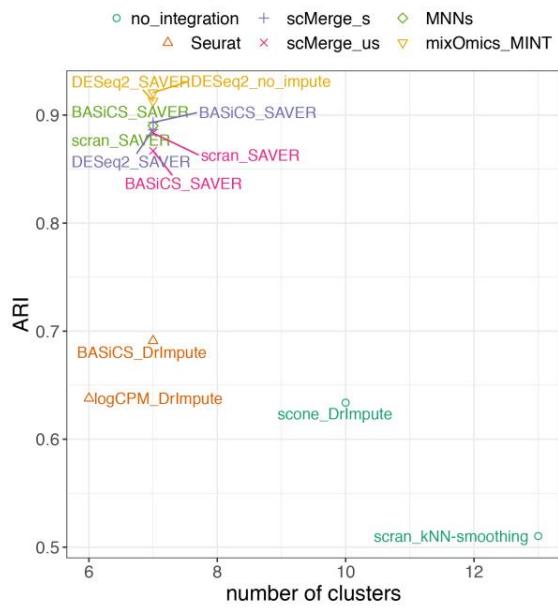
Results for the *RNAmix_Sort-seq*, *cellmix2* and *cellmix1* analyses are shown. The dimension reduction method chosen for each method was as follows: PCA for Slingshot and TSCAN, DDR tree for Monocle2, diffusion map for DPT and locally linear embedding (LLE) for SLICER. The sample sizes are 266 for *cellmix1*, 268 for *cellmix2* and 296 for *RNAmix_Sort-seq*.



Supplementary Figure 8

Additional data integration results for the single cell and RNA mixture datasets.

The kBET acceptance rate versus silhouette width coefficient for each method for the two combinations that had the highest silhouette width from the RNA mixture (a) and single cell (b) data integration analyses. The silhouette width assesses the ability of a given method to group biologically similar cells together while kBET assesses whether different batches are homogeneous after batch effect correction (scMerge_s: supervised scMerge; scMerge_us: unsupervised scMerge) (c) Additional PCA plots from the *RNA mixture* analysis and (d) t-SNE (perplexity = 30) plots from the single cell analysis to visualize the results of different method combinations. Cells are coloured according to protocol/batch information (t-SNE for MNNs and scanorama were based on batch corrected expression matrices). The samples sizes are n=636 and n=5,319 in panels c and d respectively.



Supplementary Figure 9

Performance of clustering on the *RNA mixture* datasets after data integration is applied.

Scatter plot of the ARI versus the number of clusters detected for the top performing normalization and imputation combinations. The true number of clusters for the *RNA mixture* experiment is 7.

Supplementary Table 1: Summary of the benchmarking datasets generated. Information on the 3 experimental designs employed, the single cell protocols used, the GEO accession numbers and parameters applied when generating cDNA libraries is listed.

Dataset name	Sample size	Experimental design	Protocol	GEO number	Protocol parameters
sc_CEL-seq2	274	single cells from the mixture of three cell lines	CEL-seq2	GSM3336845	1x384 well plate x0.8 then x0.9 clean up for PCR products
sc_10x	902	single cells from the mixture of three cell lines	10X Chromium	GSM3022245	standard 10X scRNA-seq protocol
sc_Drop-seq	225	single cells from the mixture of three cell lines	Drop-seq Dolomite	GSM3336849	standard Dolomite Drop-seq protocol
sc_10x_5cl	3918	single cells from the mixture of five cell lines	10X Chromium	GSM3618014	standard 10X scRNA-seq protocol
sc_CEL-seq2_5cl_p1	297	single cells from the mixture of five cell lines	CEL-seq2	GSM3618022	1x384 well plate. Pooling after first strand synthesis
sc_CEL-seq2_5cl_p2	307	single cells from the mixture of five cell lines	CEL-seq2	GSM3618023	1x384 well plate. Pooling after first strand synthesis
sc_CEL-seq2_5cl_p3	305	single cells from the mixture of five cell lines	CEL-seq2	GSM3618024	1x384 well plate. Pooling after first strand synthesis
cellmix1	266	9 cell mixtures from three cell lines	CEL-seq2	GSM3295024	subsampling 1/9 from the same 384 plate. 2x0.7 clean up for PCR products
cellmix2	268	9 cell mixtures from three cell lines	CEL-seq2	GSM3295025	subsampling 1/9 from the same 384 plate. 2x0.8 clean up for PCR products
cellmix3	285	9 cell mixtures from three cell lines	CEL-seq2	GSM3295026	subsampling 1/9 from the same 384 plate. 2x0.9 clean up for PCR products
cellmix4	288	9 cell mixtures from three cell lines	CEL-seq2	GSM3295027	subsampling 1/3 from the same 384 plate. 2x0.7 clean up for PCR products
cellmix5	21	90 cell mixture (population controls)	CEL-seq2	GSM3295023	24 samples 2x0.7 clean up for PCR products
RNAmix_CEL-seq2	340	mixture of RNA extracted from bulk population	CEL-seq2	GSM3305230	1x384 plate x0.8 then x0.9 clean up for PCR products
RNAmix_Sort-seq	296	mixture of RNA extracted from bulk population	Sort-seq	GSM3305231	1x384 plate x0.8 then x0.9 clean up for PCR products

Supplementary Table 2: Summary of the data characteristics and data analysis tasks that can be compared by each experimental design.
 The suitability of each experimental design to benchmark specific tasks is indicated by the scale * < ** < *** i.e. the *RNA mixture* datasets include a dilution series which induces different dropout levels, making it an ideal dataset for comparing

		Experimental design		
		Single cell	Cell mixture	RNA mixture
Data characteristics	Number of datasets	7	5	2
	Number of protocols	3	1	2
	Number of cell populations	3 (5)	34	7
	Population controls	Bulk RNA-seq from previous study	RNA-seq from 90 cell controls (cellmix5)	
	Technical replicates	No	No	Yes
	Source of biological variation	Different cell lines	Cell combinations from three cell lines	RNA mixing proportion from three cell lines
	Source of gene count noise	Gene expression noise + technical noise	Gene expression noise + sampling noise + technical noise	Technical noise
	Annotations	Cell identity from Demuxlet	Cell combination	RNA proportion and amount
Analysis tasks	Protocol comparison	***		**
	Quality control	*	***	*
	Normalization	**	**	***
	Imputation	*	*	***
	Differential expression analysis	**	**	***
	Clustering	**	*	***
	Trajectory analysis		***	**
	Data integration	***	*	***

Supplementary Table 3
 Summary of integrative methods used to combine data from different protocols and datasets.

Integration Method	Correct ¹	Adjust ²	Dimensionality Reduction ³	# of genes ⁴	Main parameters ⁵	Reference
MNNs	YES	NO	NO	HVGs	i. Number of nearest neighbours ii. Bandwidth of smoothing kernel	[39]
MINT Supervised	NO	NO	YES	All Genes	i. Number of components ii. If gene selection: Number of genes to select	[43]
Seurat	NO	YES	YES	HVGs	i. Number of components ii. Reference dataset iii. If multiCCA: number of iterations	[42]
scMerge	YES	NO	NO	HVGs + SEGs	i. Number of K-means clusters ii. Number of factors iii. Ratio of pseudo replicates iv. Distance metric	[41]
Scanorama	YES	NO	NO	HVGs	i. Number of HVGs ii. Number of nearest neighbours (NN) iii. Choice of approximate kNN iv. Gaussian kernel function	[40]

1. Method outputs batch-corrected count data matrix
 2. The batch effect is accounted for in the model but not batch-corrected count data matrix is output
 3. Method outputs components or factors summarising the batch-corrected data
 4. The genes utilised in the method. HVGs: Highly Variable Genes. SEGs: Stably Expressed Genes
 5. Main model parameters. Grey indicates parameters with set default values in the model which can be altered

Supplementary Table 4. The table is too large and can be viewed at [Github](#)

TABLE A.4: Individual performance metrics obtained from benchmarking analysis,
organized by task

Supplementary Table 5. The table is too large and can be viewed at [Github](#)

TABLE A.5: Run times for different analysis pipelines

Appendix B

Additional Material for Chapter 4

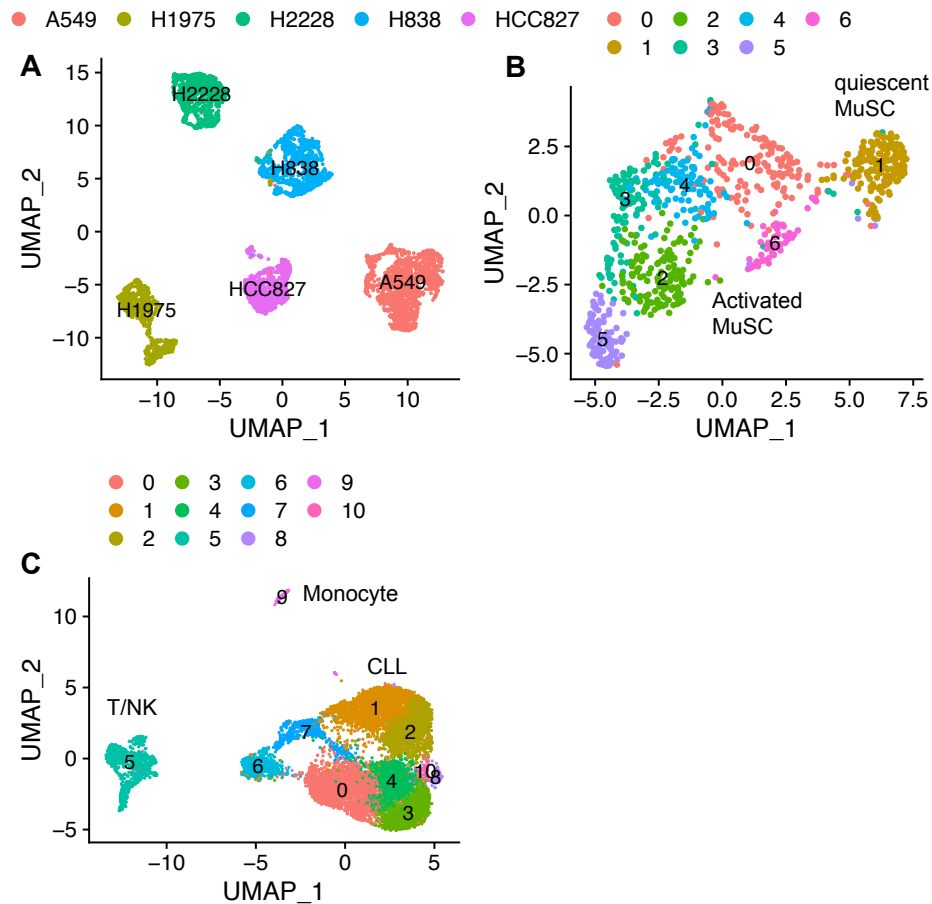


Figure S1. Dimensionality reduction (UMAP) of all samples.

UMAP visualization of cells from (A) *scmixology1* and *scmixology2* combined, (B) mouse muscle stem cells (*MuSCs*) and (C) patient *CLL2*. Cells are either colored by cell line identity (*scmixology1* and *scmixology2*) or by *Seurat* clusters and are annotated with major cell populations.

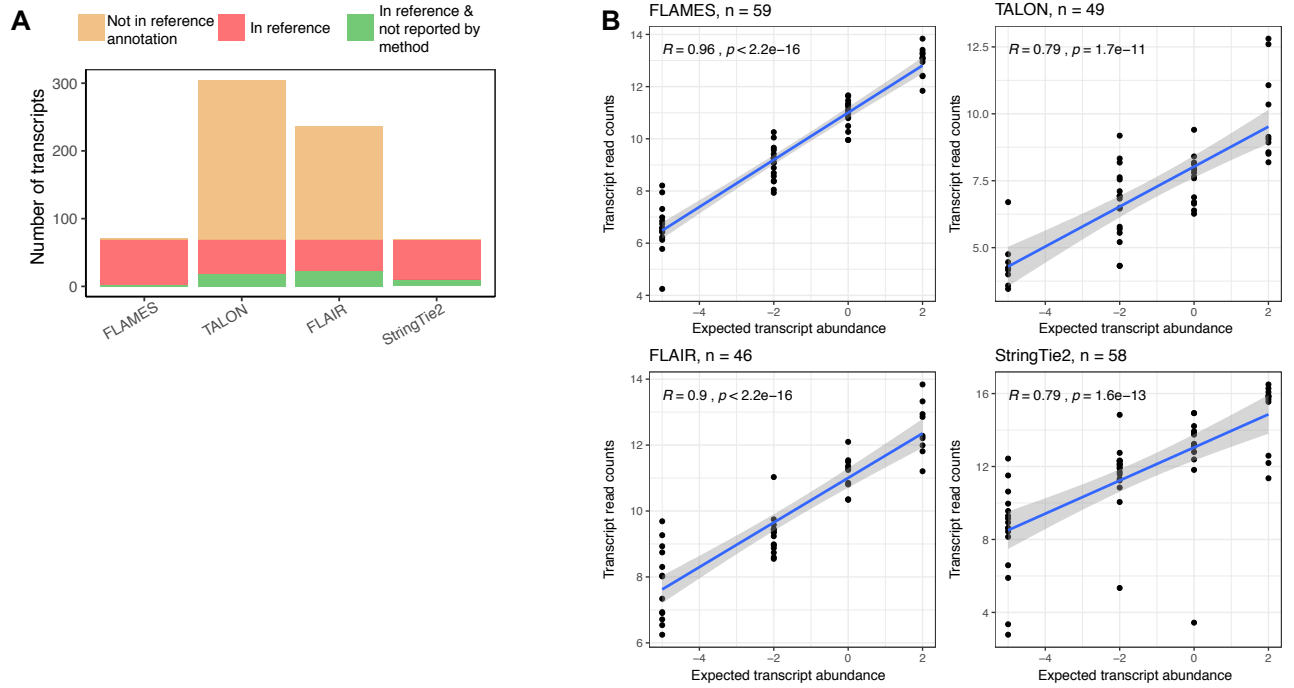


Figure S2. Comparing *FLAMES* with other methods using the SIRV dataset.

(A) Number of transcripts detected and not detected by each method, colored by whether transcript is present in the reference annotation.

(B) Scatter plot of transcript quantification where the x-axis is the expected transcript abundance based on the known concentration in the spike-in sample and the y-axis is the recovered transcript count ($\log_2$) generated by each method. The Spearman correlation coefficient (R) and P -value and the number of transcripts (n) included in the analysis are annotated in each plot.

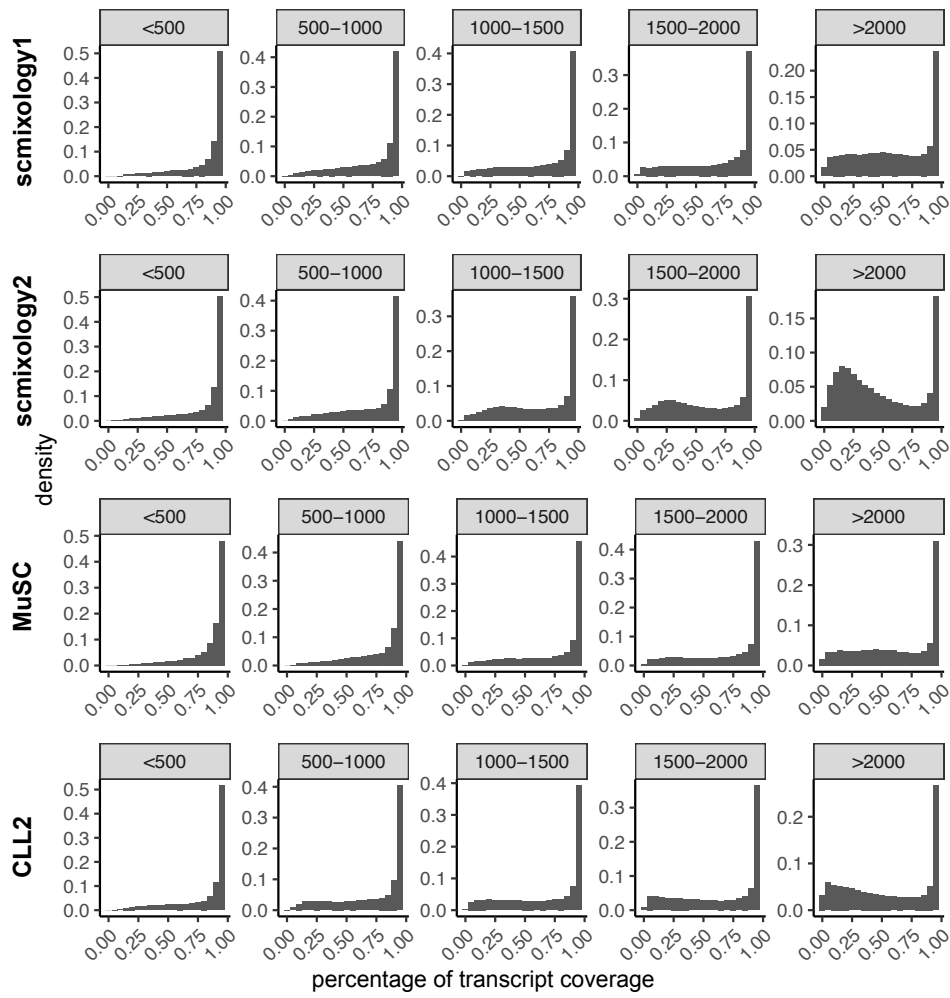


Figure S3. Percentage of transcript coverage of individual reads stratified by length.

The histogram shows the distribution of raw transcript coverage of reads aligned to the transcripts generated by *FLAMES*. Transcripts are stratified by length and divided into equal 500bp bins.

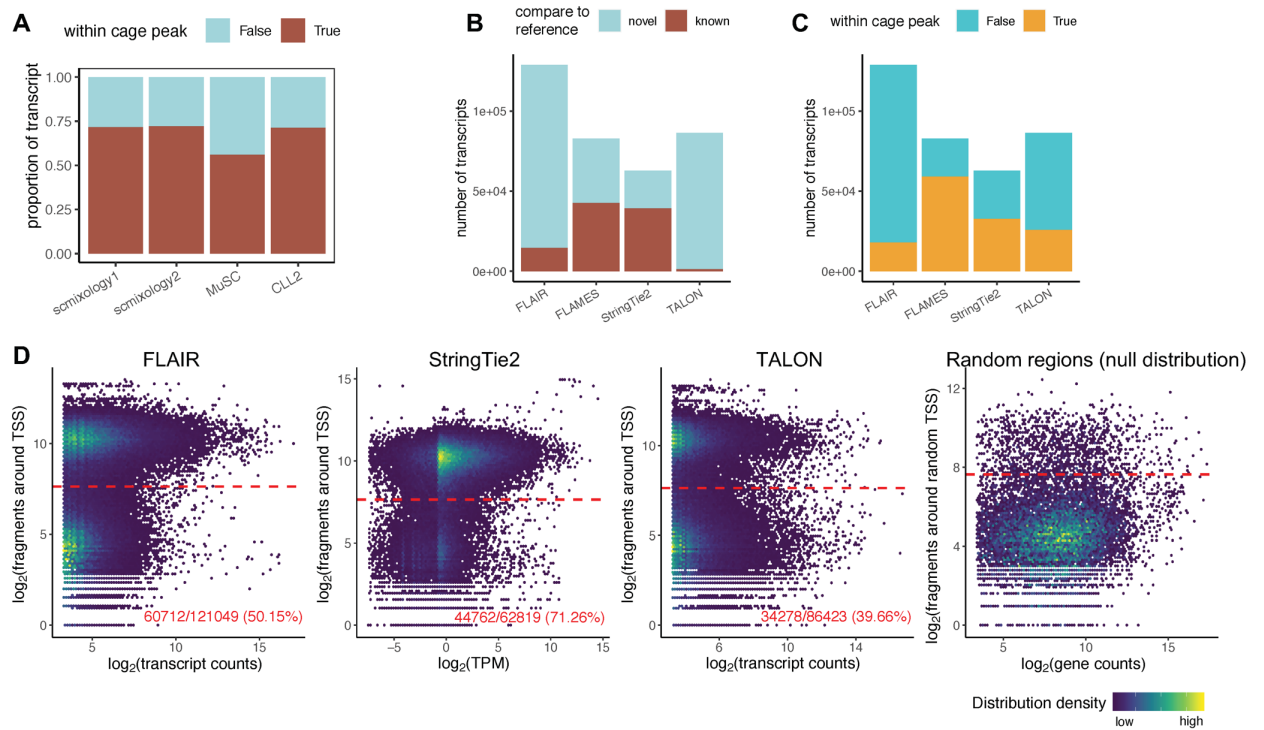


Figure S4. Comparing *FLAMES* results with database annotations and other methods.

(A) Proportion of transcript for which the TSS is within the FANTOM cage peak annotations for each sample.

(B) Number of transcripts generated by each method and the number of transcripts that match to the reference annotation.

(C) Similar to (B) and colored by the number of transcripts that fall within the FANTOM cage pack annotations.

(D) From left to right, density scatter plot showing total transcript count versus scATAC-seq read count around the TSS regions for *FLAIR*, *StringTie2* and *TALON*, together with a similar plot based on random TSS. The red line shows the threshold that separate the open chromatin and the background. The percentage shows the transcripts that have their TSSs in open chromatin regions.

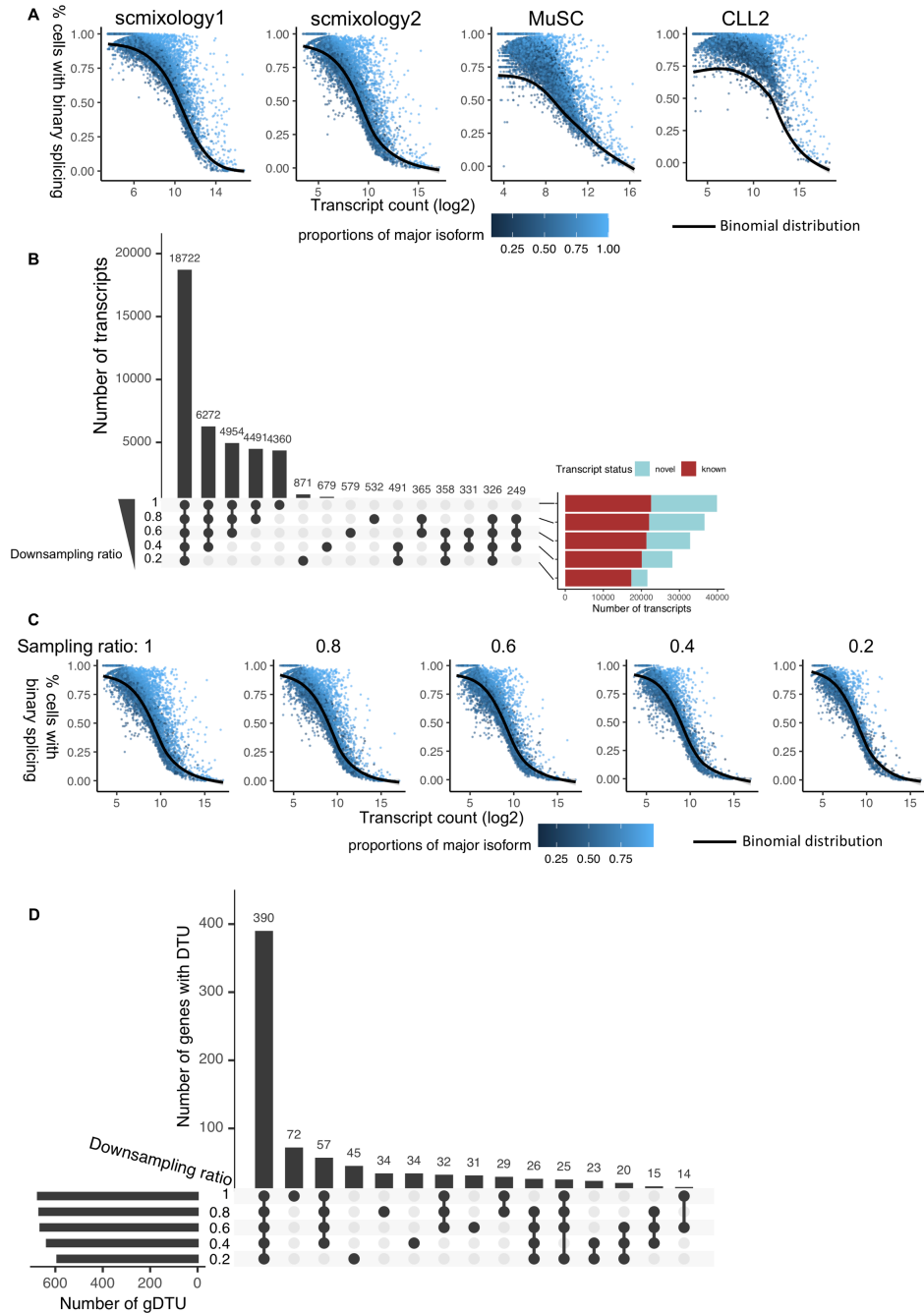


Figure S5. Binary splicing and downsampling analysis.

(A) scatter plot shows the total transcript counts for each gene versus the percentage of binary splicing, colored by the percentage of major isoform counts. Black line shows the null distribution that follows binomial distribution.

(B) Upset plot shows the overlap of transcripts generated from downsampling analysis of *scmixology2* with sampling ratio from 0.2 to 0.8, annotated with number of transcripts match to reference annotation.

(C) Similar to (A), scatter shows the binary splicing of samplings with different downsampling ratio of *scmixology2*.

(D) Upset plot shows the overlap of genes with significant DTU on *scmixology* data with different downsampling ratios.

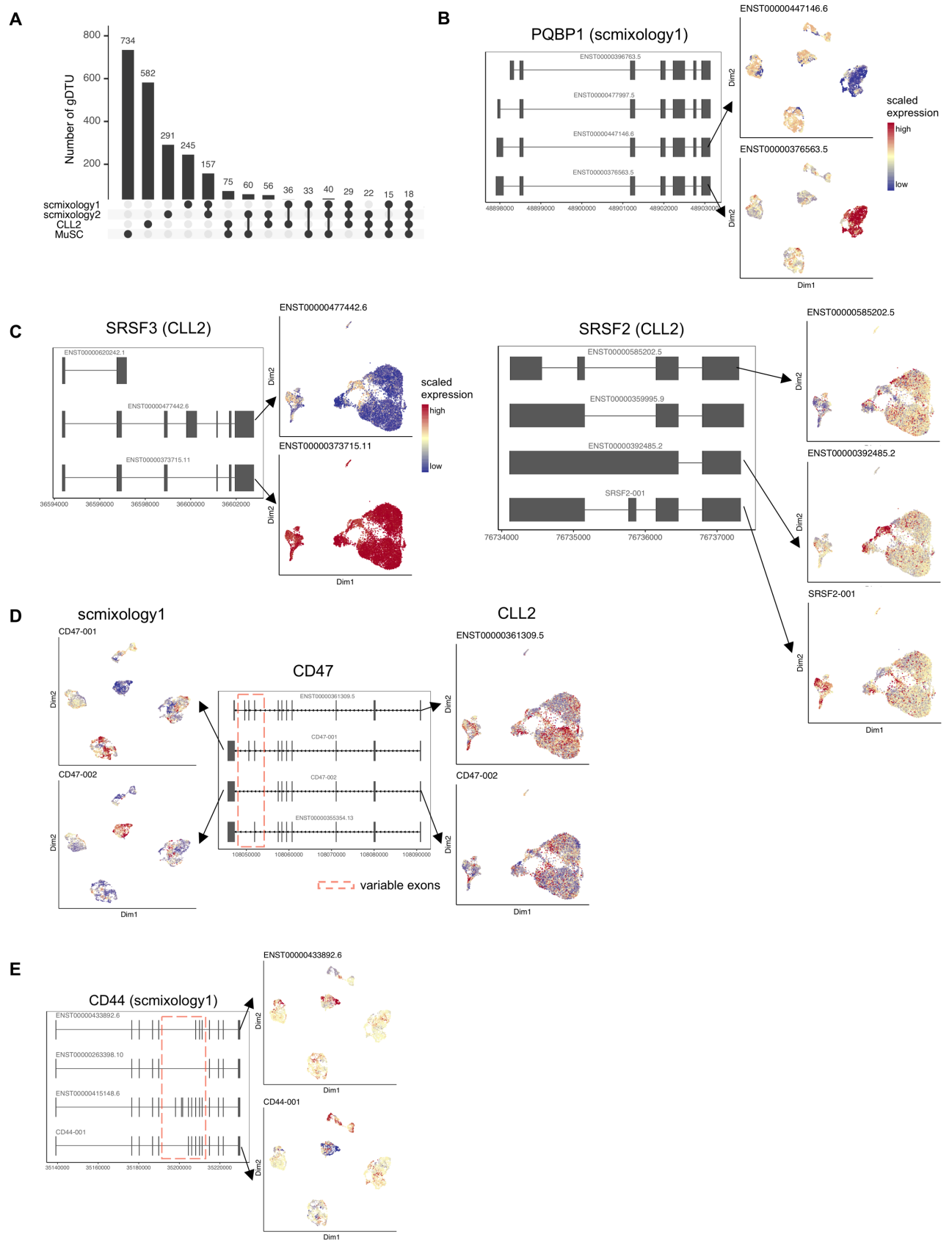


Figure S6. Genes with differential transcript usage.

(A) Upset plot shows the overlap of genes with differential transcript usage (gDTU) across different samples.

- (B) Top 4 most abundant isoforms of PQBP1 in *scmixology1* and UMAP visualization colored by the expression of two isoforms with differential expression across different cell lines.
- (C) Isoforms of SRSF3 in *CLL2*, with UMAP visualization colored by two isoforms with differential expression across different clusters
- (D) Top 4 most abundant isoforms of CD47 in *scmixology1* and *CLL2*, together with UMAP visualization of expression of selected transcripts. Exons with alternative splicing was highlighted.
- (E) Top 4 most abundant isoforms of CD44 in *scmixology1*, and UMAP visualization of transcript expression.

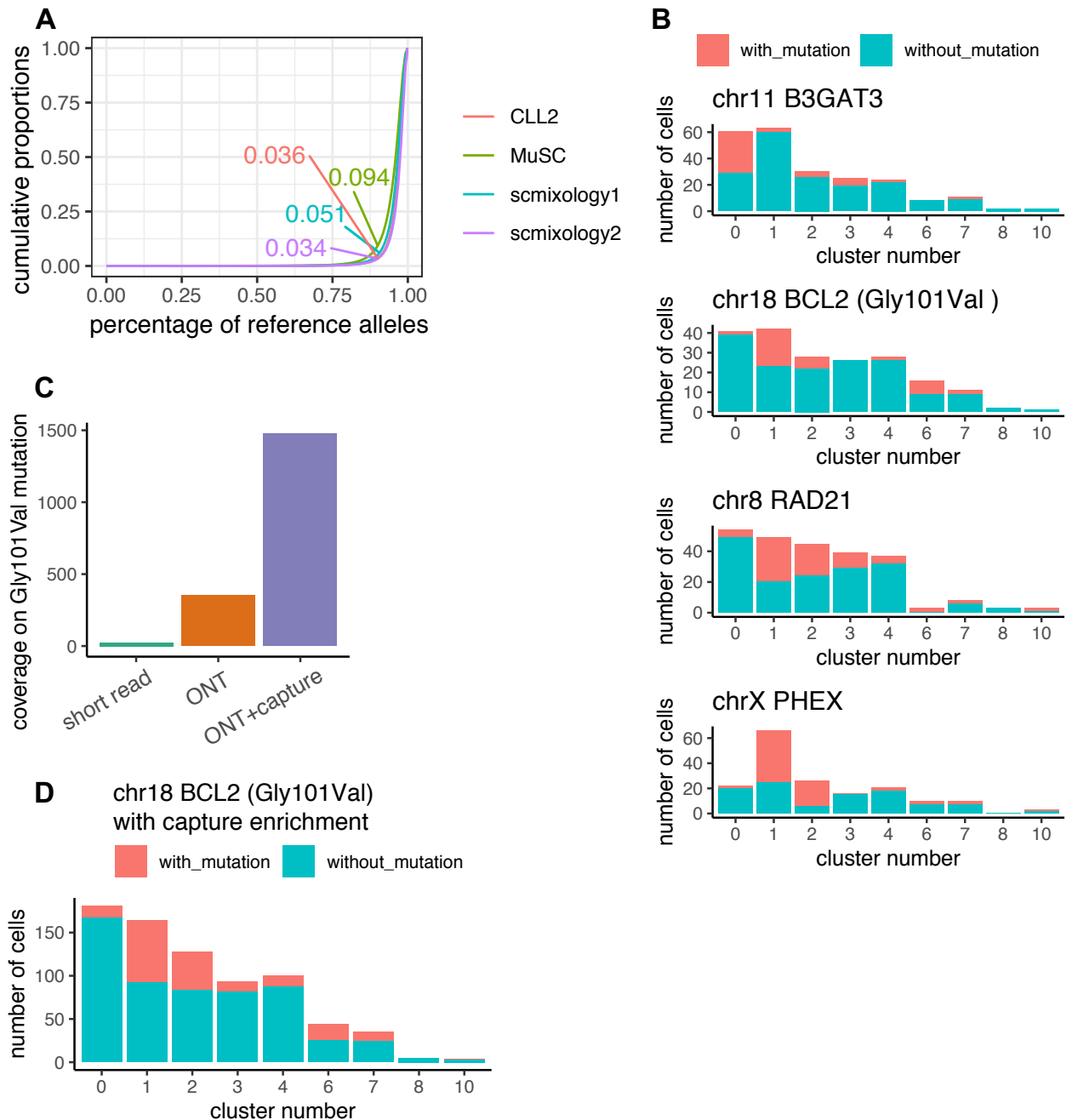


Figure S7. Identification of genomic variation from long read by *FLAMES*.

(A) Cumulative proportions of bases with certain percentage of reference alleles. The cumulative proportions of bases with 90% allele frequency were labelled on the plot, represent the percentage of reads that have lower than 90% reference allele frequency.

(B) Bar plot of four significant mutations shows the distribution of cells in each cluster (x axis) that detected the mutation.

(C) The read coverage of the Gly101Val BCL2 mutation with different methods, short read stands for scRNA-seq from the same cells.

(D) Similar to (B), the bar plot of the distributions of cells with Gly101Val BCL2 mutation across clusters, with data acquired from capture enrichment (see Methods).

Sample name	Experiment design	Total sample size	Cells sampled for long read sequencing	Protocol	Data accession number	Other data on this sample
scmixology1	equal mixture of 5 cell lines	3918	465	FLT-seq with 10x single cell gene expression v2 chemistry	GSE154869	NA
scmixology2	equal mixture of 5 cell lines	1438	183	FLT-seq with 10x single cell gene expression v3 chemistry	GSE154870	scATACseq on the same cell mixture (GSE142285 Lib90)
MuSC	mouse quiescent muscle stem cell and activated stem cells	853	242	FLT-seq with 10x single cell gene expression v2 chemistry	GSE154868	NA
CLL2	frozen PBMC sample from relapsed CLL patient after venetoclax treatment	10451	1847	FLT-seq with 10x single cell immune profiling	data in submission	antibody driven count of 17 surface proteins

Table S1. Summary of samples included in this study

Table S2. The table is too large to include and can be viewed at [Github](#)

TABLE B.2: Results of differential transcript usages analysis

Table S3. The table is too large to include and can be viewed at [Github](#)

TABLE B.3: Results of GO enrichment analysis using genes with differential transcript usage

Appendix C

Other Research Activities

C.1 Publications with significant contributions

- Clonal multi-omics reveals Bcor as a novel regulator of dendritic cell development and its action at a single cell level.

Contributions: Joint-first author. Performed data analysis and generated figures.

Status: Under review at *Immunity*.

- A novel lymphoid-primed progenitor marked by Dach1 downregulation identified with single cell multi-omics.

Contributions: Joint-first author. Performed data analysis and generated figures.

Status: In press at *Nature Immunology*.

- Longitudinal single-cell immune profiling revealed distinct innate immune response in asymptomatic COVID-19 patients.

Contributions: Joint-corresponding author. Supervised data analysis and wrote the manuscript.

Status: Submitted to *Immunity*.

C.2 Conference presentations

- **OZ Single Cells conference** (*WEHI, Melbourne, Australia*). Nov 2017
Selected Speaker. Title: Tools for preprocessing and benchmarking single cell RNA sequencing data.
- **Single Cell Biology** (*The Wellcome Trust Genome Campus, Hinxton, UK*). Mar 2018
Poster Presentation. Title: Designing a single cell RNA-seq Benchmarking dataset to compare protocols and analysis methods.
- **Collaborative Computational Tools for the Human Cell Atlas** (*Aptos, CA, USA*). May 2018
Project Title: CellBench: generating data and software for comparing single cell analysis methods.
- **ISSCR meeting** (*Melbourne, Australia*). Jun 2018
Selected Speaker. Title: Dach1 expression segregates lymphoid and myeloid fate in early hematopoiesis.
- **AMSI BioinfoSummer** (*Perth, Australia*). Dec 2018
Poster Presentation. Title: scRNA-seq mixology: towards better benchmarking of single cell RNA-seq protocols.
- **Single Cell Genomics** (*Stockholm, Sweden*). Sep 2019
Poster Presentation. Title: Benchmarking single cell RNA-sequencing analysis pipelines using mixture control experiments.
- **Oxford Nanopore Day** (*Melbourne, Australia*). Oct 2019
Invited Speaker. Title: Characterization of single cell full-length transcriptomes using Nanopore sequencing.
- **London Calling 2020** (*Virtual event*). Jun 2020
Invited Speaker. Title: Characterization of full-length isoforms in single cells with Nanopore long-read sequencing.

Bibliography

- [1] Valentine Svensson, Roser Vento-Tormo, and Sarah A Teichmann. Exponential scaling of single-cell RNA-seq in the last decade. *arXiv*, 13(4):599–604, 2017. ISSN 1754-2189. doi: 10.1038/nprot.2017.149. URL <https://arxiv.org/ftp/arxiv/papers/1704/1704.01379.pdf><http://arxiv.org/abs/1704.01379>.
- [2] Allon M. Klein, Linas Mazutis, Ilke Akartuna, Naren Tallapragada, Adrian Veres, Victor Li, Leonid Peshkin, David A. Weitz, and Marc W. Kirschner. Droplet Bar-coding for Single-Cell Transcriptomics Applied to Embryonic Stem Cells. *Cell*, 161(5):1187–1201, may 2015. ISSN 00928674. doi: 10.1016/j.cell.2015.04.044. URL <http://linkinghub.elsevier.com/retrieve/pii/S0092867415005000>.
- [3] Todd M Gierahn, Marc H Wadsworth, Travis K Hughes, Bryan D Bryson, Andrew Butler, Rahul Satija, Sarah Fortune, J Christopher Love, Alex K Shalek, Marc H Wadsworth II, Travis K Hughes, Bryan D Bryson, Andrew Butler, Rahul Satija, Sarah Fortune, J Christopher Love, and Alex K Shalek. Seq-Well: portable, low-cost RNA sequencing of single cells at high throughput. *Nature Methods*, 14(4):1–8, 2017. ISSN 1548-7091. doi: 10.1038/nmeth.4179. URL <http://www.nature.com/doifinder/10.1038/nmeth.4179>.
- [4] Tamar Hashimshony, Naftalie Senderovich, Gal Avital, Agnes Klochender, Yaron de Leeuw, Leon Anavy, Dave Gennert, Shuqiang Li, Kenneth J Livak, Orit Rozenblatt-Rosen, Yuval Dor, Aviv Regev, and Itai Yanai. CEL-Seq2: sensitive highly-multiplexed single-cell RNA-Seq. *Genome biology*, 17:77, 2016. ISSN 1474-760X. doi: 10.1186/s13059-016-0938-8. URL <http://www.ncbi.nlm.nih.gov/pubmed/27121950><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4848782>.

- [5] Diego Adhemar Jaitin, Hadas Keren-Shaul, Naama Elefant, and Ido Amit. Each cell counts: Hematopoiesis and immunity research in the era of single cell genomics. *Seminars in Immunology*, 27(1):67–71, 2015. ISSN 10445323. doi: 10.1016/j.smim.2015.01.002. URL <http://linkinghub.elsevier.com/retrieve/pii/S1044532315000056>.
- [6] Marlon Stoeckius, Christoph Hafemeister, William Stephenson, Brian Houck-Loomis, Pratip K Chattopadhyay, Harold Swerdlow, Rahul Satija, and Peter Smibert. Large-scale simultaneous measurement of epitopes and transcriptomes in single cells. *bioRxiv*, page 113068, 2017. doi: 10.1101/113068. URL <https://www.biorxiv.org/content/early/2017/06/07/113068>.
- [7] Gavin Kelsey, Oliver Stegle, and Wolf Reik. Single-cell epigenomics: Recording the past and predicting the future. *Science*, 127(2015):2017, 2017. ISSN 0022-2151. doi: 10.1017/S0022215115001383.
- [8] Jason Daniel Buenrostro, Ryan Corces, Beijing Wu, Alicia N Schep, Caleb Lareau, Ravindra Majeti, Howard Chang, and William Greenleaf. Single-cell epigenomics maps the continuous regulatory landscape of human hematopoietic differentiation. *doi.org*, page 109843, 2017. ISSN 1573-661X. doi: 10.1101/109843. URL <http://biorxiv.org/content/early/2017/02/21/109843.1.abstract><https://www.biorxiv.org/content/early/2017/02/21/109843.1>.
- [9] Freeman Lan, Benjamin Demaree, Noorsher Ahmed, and Adam R. Abate. Single-cell genome sequencing at ultra-high-throughput with microfluidic droplet barcoding. *Nature Biotechnology*, 35(7):640–646, 2017. ISSN 15461696. doi: 10.1038/nbt.3880. URL <http://dx.doi.org/10.1038/nbt.3880>.
- [10] Atray Dixit, Oren Parnas, Biyu Li, Jenny Chen, Charles P. Fulco, Livnat Jerby-Arnon, Nemanja D. Marjanovic, Danielle Dionne, Tyler Burks, Raktima Raychowdhury, Britt Adamson, Thomas M. Norman, Eric S. Lander, Jonathan S. Weissman, Nir Friedman, and Aviv Regev. Perturb-Seq: Dissecting Molecular Circuits with Scalable Single-Cell RNA Profiling of Pooled Genetic Screens. *Cell*, 167(7):1853–1866.e17, 2016. ISSN 00928674. doi: 10.1016/j.cell.2016.11.038. URL <http://linkinghub.elsevier.com/retrieve/pii/S0092867416316105>.

- [11] Paul Datlinger, Christian Schmidl, Andre F Rendeiro, Peter Traxler, Johanna Klughammer, Linda Schuster, and Christoph Bock. Pooled CRISPR screening with single-cell transcriptome read-out. *bioRxiv*, 2016. ISSN 1548-7091. doi: 10.1101/083774.
- [12] Chee-Huat Linus Eng, Michael Lawson, Qian Zhu, Ruben Dries, Noushin Koulana, Yodai Takei, Jina Yun, Christopher Cronin, Christoph Karp, Guo-Cheng Yuan, and Long Cai. Transcriptome-scale super-resolved imaging in tissues by RNA seqFISH+. *Nature*, 568(7751):235–239, apr 2019. ISSN 0028-0836. doi: 10.1038/s41586-019-1049-y. URL <http://dx.doi.org/10.1038/s41586-019-1049-y><http://www.nature.com/articles/s41586-019-1049-y>.
- [13] Yan Song, Olga B. Botvinnik, Michael T. Lovci, Boyko Kakaradov, Patrick Liu, Jia L. Xu, and Gene W. Yeo. Single-Cell Alternative Splicing Analysis with Expedition Reveals Splicing Dynamics during Neuron Differentiation. *Molecular Cell*, 67(1):148–160.e6, 2017. ISSN 10974164. doi: 10.1016/j.molcel.2017.06.003. URL <http://dx.doi.org/10.1016/j.molcel.2017.06.003>.
- [14] Jennifer Westoby, Pavel Artemov, Martin Hemberg, and Anne Ferguson-Smith. Obstacles to detecting isoforms using full-length scRNA-seq data. *Genome Biology*, 21(1):74, dec 2020. ISSN 1474-760X. doi: 10.1186/s13059-020-01981-w. URL <http://www.ncbi.nlm.nih.gov/pubmed/32293520><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC7087381><https://genomebiology.biomedcentral.com/articles/10.1186/s13059-020-01981-w>.
- [15] Roger Volden, Theron Palmer, Ashley Byrne, Charles Cole, Robert J. Schmitz, Richard E. Green, and Christopher Vollmers. Improving nanopore read accuracy with the R2C2 method enables the sequencing of highly multiplexed full-length single-cell cDNA. *Proceedings of the National Academy of Sciences*, 115(39):9726–9731, sep 2018. ISSN 0027-8424. doi: 10.1073/pnas.1806447115. URL <http://www.pnas.org/lookup/doi/10.1073/pnas.1806447115>.
- [16] Mandeep Singh, Ghamdan Al-Eryani, Shaun Carswell, James M. Ferguson, James Blackburn, Kirston Barton, Daniel Roden, Fabio Luciani, Tri Giang Phan, Simon Junankar, Katherine Jackson, Christopher C. Goodnow, Martin A. Smith, and Alexander Swarbrick. High-throughput targeted long-read single cell sequencing

- reveals the clonal and transcriptional landscape of lymphocytes. *Nature Communications*, 10(1):3120, dec 2019. ISSN 2041-1723. doi: 10.1038/s41467-019-11049-4. URL <https://www.biorxiv.org/content/early/2018/09/24/424945.full.pdf+html>{%}0A<https://www.biorxiv.org/content/biorxiv/suppl/2018/09/24/424945.DC1/424945-1.pdf><http://www.nature.com/articles/s41586-019-1369-y><http://www.nature.com/articles/s41467-019-11049-4>.
- [17] Kevin Lebrigand, Virginie Magnone, Pascal Barbry, and Rainer Waldmann. High throughput error corrected Nanopore single cell transcriptome sequencing. *Nature Communications*, 11(1):4025, dec 2020. ISSN 2041-1723. doi: 10.1038/s41467-020-17800-6. URL <http://dx.doi.org/10.1038/s41467-020-17800-6><http://www.nature.com/articles/s41467-020-17800-6>.
- [18] Dominic Grün, Lennart Kester, and Alexander van Oudenaarden. Validation of noise models for single-cell transcriptomics. *Nature methods*, 11(6):637–40, 2014. ISSN 1548-7105. doi: 10.1038/nmeth.2930. URL <http://www.ncbi.nlm.nih.gov/pubmed/24747814>.
- [19] Valentine Svensson, Kedar Nath Natarajan, Lam-Ha Ly, Ricardo J Miragaia, Charlotte Labalette, Iain C Macaulay, Ana Cvejic, and Sarah A Teichmann. Power analysis of single-cell RNA-sequencing experiments. *Nature Methods*, 14(4):381–387, 2017. ISSN 1548-7091. doi: 10.1038/nmeth.4220. URL <http://www.nature.com/doifinder/10.1038/nmeth.4220>.
- [20] Aaron M Streets and Yanyi Huang. How deep is enough in single-cell RNA-seq? *Nature Biotechnology*, 32(10):1005–1006, 2014. ISSN 1087-0156. doi: 10.1038/nbt.3039. URL <http://www.nature.com/doifinder/10.1038/nbt.3039>.
- [21] Oliver Stegle. Computational and analytical challenges in single-cell transcriptomics. *Nature Publishing Group*, 16(January 2014):133–145, 2015. ISSN 1471-0056. doi: 10.1038/nrg3833. URL <http://dx.doi.org/10.1038/nrg3833>.
- [22] Swati Parekh, Christoph Ziegenhain, Beate Vieth, Wolfgang Enard, and Ines Hellmann. zUMIs: A fast and flexible pipeline to process RNA sequencing data with UMIs. *bioRxiv*, pages 2–7, 2017. doi: 10.1101/153940. URL <http://www.biorxiv.org/content/early/2017/06/22/153940?collection=>.

- [23] Kamran Shaikh, Henry Avers, Frcpc Objective, Measurement Group, and Oscar Emr. UMI-tools: Modelling sequencing errors in Unique Molecular Identifiers to improve quantification accuracy. *bioRxiv*, 2016.
- [24] Luyi Tian, Shian Su, Daniela Amann-Zalcenstein, Christine Biben, Shalin H. Naik, and Matthew E. Ritchie. scPipe: a flexible data preprocessing pipeline for single-cell RNA-sequencing data. *bioRxiv*, page 175927, 2017. doi: 10.1101/175927. URL <https://www.biorxiv.org/content/early/2017/08/14/175927>.
- [25] Tomislav Ilicic, Jong Kyoung Kim, Aleksandra A. Kolodziejczyk, Frederik Otzen Bagger, Davis James McCarthy, John C. Marioni, and Sarah A. Teichmann. Classification of low quality cells from single-cell RNA-seq data. *Genome Biology*, 17(1):29, 2016. ISSN 1474-760X. doi: 10.1186/s13059-016-0888-1. URL <http://genomebiology.com/2016/17/1/29>.
- [26] Davis J. McCarthy Lun, Aaron T. L. and John C. Marioni. A step-by-step workflow for basic analyses of single-cell RNA-seq data. *F1000Research*, 5:2122(0), 2016. ISSN 2046-1402. doi: 10.12688/f1000research.9501.1.
- [27] Pablo G Camara. Methods and challenges in the analysis of single-cell RNA-sequencing data. *Current Opinion in Systems Biology*, 2017. ISSN 24523100. doi: 10.1016/j.coisb.2017.12.007. URL <http://linkinghub.elsevier.com/retrieve/pii/S2452310017301944>.
- [28] Dominic Grün and Alexander van Oudenaarden. Design and Analysis of Single-Cell Sequencing Experiments. *Cell*, 163(4):799–810, 2015. ISSN 00928674. doi: 10.1016/j.cell.2015.10.039. URL <http://linkinghub.elsevier.com/retrieve/pii/S0092867415013537>.
- [29] M. D. Robinson, D. J. McCarthy, and G. K. Smyth. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26(1):139–140, 2010. ISSN 1367-4803. doi: 10.1093/bioinformatics/btp616. URL <http://bioinformatics.oxfordjournals.org/cgi/doi/10.1093/bioinformatics/btp616>.
- [30] Aaron T. L. Lun, Karsten Bach, and John C. Marioni. Pooling across cells to normalize single-cell RNA sequencing data with many zero counts. *Genome Biology*,

- 17(1):75, 2016. ISSN 1474-760X. doi: 10.1186/s13059-016-0947-7. URL <http://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-0947-7>.
- [31] Rhonda Bacher, Li Fang Chu, Ning Leng, Audrey P. Gasch, James A. Thomson, Ron M. Stewart, Michael Newton, and Christina Kendzierski. SCnorm: Robust normalization of single-cell RNA-seq data. *Nature Methods*, 14(6):584–586, 2017. ISSN 15487105. doi: 10.1038/nmeth.4263.
- [32] Shun H. Yip, Panwen Wang, Jean-Pierre A. Kocher, Pak Chung Sham, and Junwen Wang. Linnorm: improved statistical analysis for single cell RNA-seq expression data. *Nucleic Acids Research*, (October):1–12, 2017. ISSN 0305-1048. doi: 10.1093/nar/gkx828. URL <http://academic.oup.com/nar/article/doi/10.1093/nar/gkx828/4160405/Linnorm-improved-statistical-analysis-for-single>.
- [33] Catalina a. Vallejos, John C. Marioni, and Sylvia Richardson. BASiCS: Bayesian Analysis of Single-Cell Sequencing Data. *PLOS Computational Biology*, 11(6): e1004333, 2015. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1004333. URL <http://dx.plos.org/10.1371/journal.pcbi.1004333>.
- [34] B. Ding, L. Zheng, Y. Zhu, N. Li, H. Jia, R. Ai, a. Wildberg, and W. Wang. Normalization and noise reduction for single cell RNA-seq experiments. *Bioinformatics*, 31(February):2225–2227, 2015. ISSN 1367-4803. doi: 10.1093/bioinformatics/btv122. URL <http://bioinformatics.oxfordjournals.org/cgi/doi/10.1093/bioinformatics/btv122>.
- [35] Christopher Yau and Emma Pierson. ZIFA: Dimensionality reduction for zero-inflated single-cell gene expression analysis. *Genome Biology*, page 019141, 2015. ISSN 1474-760X. doi: 10.1101/019141. URL <http://biorxiv.org/content/early/2015/06/14/019141.abstract>.
- [36] Davide Risso, Fanny Perraudeau, Svetlana Gribkova, Sandrine Dudoit, and Jean-Philippe Vert. ZINB-WaVE: A General and Flexible Method for Signal Extraction from Single-Cell RNA-Seq Data. *bioRxiv*, 9(1): 125112, 2017. ISSN 2041-1723. doi: 10.1101/125112. URL <http://www.nature.com/articles/s41467-017-02554-5https://www.biorxiv.org/content/biorxiv/early/2017/04/06/125112.full.pdf>.

- [37] Florian Wagner, Yun Yan, and Itai Yanai. K-nearest neighbor smoothing for single-cell RNA-Seq data. *bioRxiv*, 2017. doi: 10.1101/217737.
- [38] Michael B Cole, Davide Risso, Allon Wagner, David DeTomaso, John Ngai, Elizabeth Purdom, Sandrine Dudoit, and Nir Yosef. Performance Assessment and Selection of Normalization Procedures for Single-Cell RNA-Seq. *bioRxiv*, page 235382, 2017. doi: 10.1101/235382. URL <https://www.biorxiv.org/content/early/2017/12/16/235382>.
- [39] Lars Velten, Simon F. Haas, Simon Raffel, Sandra Blaszkiewicz, Saiful Islam, Bianca P. Hennig, Christoph Hirche, Christoph Lutz, Eike C. Buss, Daniel Nowak, Tobias Boch, Wolf-Karsten Hofmann, Anthony D. Ho, Wolfgang Huber, Andreas Trumpp, Marieke A. G. Essers, and Lars M. Steinmetz. Human haematopoietic stem cell lineage commitment is a continuous process. *Nature Cell Biology*, 19(4):271–281, 2017. ISSN 1465-7392. doi: 10.1038/ncb3493. URL <http://www.nature.com/doifinder/10.1038/ncb3493>.
- [40] Wei Vivian Li and Jingyi Jessica Li. scImpute: accurate and robust imputation for single cell RNA-seq data. *bioRxiv*, pages 1–19, 2017. doi: 10.1101/141598. URL <https://www.biorxiv.org/content/biorxiv/early/2017/06/04/141598.full.pdf>.
- [41] Peijie Lin, Michael Troup, and Joshua W. K. Ho. CIDR: Ultrafast and accurate clustering through imputation for single-cell RNA-seq data. *Genome Biology*, 18(1):59, 2017. ISSN 1474-760X. doi: 10.1186/s13059-017-1188-0. URL <http://genomebiology.biomedcentral.com/articles/10.1186/s13059-017-1188-0>.
- [42] David van Dijk, Juozas Nainys, Roshan Sharma, Pooja Kathail, Ambrose J Carr, Kevin R Moon, Linas Mazutis, Guy Wolf, Smita Krishnaswamy, and Dana Pe’er. MAGIC: A diffusion-based imputation method reveals gene-gene interactions in single-cell RNA-sequencing data. *bioRxiv*, 2017. doi: 10.1101/111591. URL <http://dx.doi.org/10.1101/111591>.
- [43] Huipeng Li, Elise T Courtois, Debarka Sengupta, Yuliana Tan, Kok Hao Chen, Jolene Jie Lin Goh, Say Li Kong, Clarinda Chua, Lim Kiat Hon, Wah Siew Tan, Mark Wong, Paul Jongjoon Choi, Lawrence J K Wee, Axel M Hillmer, Iain Beehuat Tan, Paul Robson, and Shyam Prabhakar. Reference component analysis of single-cell

- transcriptomes elucidates cellular heterogeneity in human colorectal tumors. *Nature Genetics*, (November 2016), 2017. ISSN 1061-4036. doi: 10.1038/NG.3818. URL <http://www.nature.com/ng/journal/vaop/ncurrent/full/ng.3818.html>.
- [44] Saskia Freytag, Ingrid Lonnstedt, Milica Ng, and Melanie Bahlo. Cluster Headache: Comparing Clustering Tools for 10X Single Cell Sequencing Data. *bioRxiv*, (4), 2017. doi: 10.1101/203752.
- [45] Bo Wang, Junjie Zhu, Emma Pierson, Daniele Ramazzotti, and Serafim Batzoglou. Visualization and analysis of single-cell RNA-seq data by kernel-based similarity learning. *Nature methods*, (March 2017):052225, 2017. ISSN 1548-7091. doi: 10.1101/052225. URL <http://biorxiv.org/lookup/doi/10.1101/052225>.
- [46] Vladimir Yu Kiselev, Kristina Kirschner, Michael T Schaub, Tallulah Andrews, Andrew Yiu, Tamir Chandra, Kedar N Natarajan, Wolf Reik, Mauricio Barahona, Anthony R Green, and Martin Hemberg. SC3: consensus clustering of single-cell RNA-seq data. *Nature Methods*, 14(5):483–486, 2017. ISSN 1548-7091. doi: 10.1038/nmeth.4236. URL <http://www.nature.com/doifinder/10.1038/nmeth.4236>.
- [47] Maziyar Baran Pouyan and Dennis Kostka. Random forest based similarity learning for single cell RNA sequencing data. *bioRxiv*, 2018. doi: 10.1101/258699.
- [48] Kelly Street, Davide Risso, Russell B Fletcher, Diya Das, John Ngai, Nir Yosef, Elizabeth Purdom, and Sandrine Dudoit. Slingshot: Cell lineage and pseudotime inference for single-cell transcriptomics. *bioRxiv*, page 128843, 2017. doi: 10.1101/128843. URL <https://www.biorxiv.org/content/early/2017/04/19/128843>.
- [49] Cole Trapnell, Davide Cacchiarelli, Jonna Grimsby, Prapti Pokharel, Shuqiang Li, Michael Morse, Niall J Lennon, Kenneth J Livak, Tarjei S Mikkelsen, and John L Rinn. The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nature biotechnology*, 32(4):381–6, 2014. ISSN 1546-1696. doi: 10.1038/nbt.2859. URL <http://www.ncbi.nlm.nih.gov/pubmed/24658644>.

- [50] Andre Olsson, Meenakshi Venkatasubramanian, Viren K. Chaudhri, Bruce J. Aronow, Nathan Salomonis, Harinder Singh, and H. Leighton Grimes. Single-cell analysis of mixed-lineage states leading to a binary cell fate choice. *Nature*, pages 1–22, 2016. ISSN 0028-0836. doi: 10.1038/nature19348. URL <http://www.nature.com/doifinder/10.1038/nature19348>.
- [51] Sean C. Bendall, Kara L. Davis, El Ad David Amir, Michelle D. Tadmor, Erin F. Simonds, Tiffany J. Chen, Daniel K. Shenfeld, Garry P. Nolan, and Dana Pe’Er. Single-cell trajectory detection uncovers progression and regulatory coordination in human b cell development. *Cell*, 157(3):714–725, 2014. ISSN 10974172. doi: 10.1016/j.cell.2014.04.005. URL <http://dx.doi.org/10.1016/j.cell.2014.04.005>.
- [52] Martin Wahlestedt, Vasileios Ladopoulos, Isabel Hidalgo, Manuel Sanchez Castillo, Rebecca Hannah, Petter Säwén, Haixia Wan, Monika Dudenhöffer-Pfeifer, Mattias Magnusson, Gudmundur L Norddahl, Berthold Göttgens, and David Bryder. Critical Modulation of Hematopoietic Lineage Fate by Hepatic Leukemia Factor. *Cell reports*, 21(8):2251–2263, 2017. ISSN 2211-1247. doi: 10.1016/j.celrep.2017.10.112. URL <http://www.ncbi.nlm.nih.gov/pubmed/29166614>.
- [53] Wouter Saelens, Robrecht Cannoodt, Helena Todorov, and Yvan Saeys. A comparison of single-cell trajectory inference methods: towards more accurate and robust tools. *bioRxiv*, 2018. doi: 10.1101/276907. URL <https://www.biorxiv.org/content/biorxiv/early/2018/03/05/276907.full.pdf>.
- [54] Philipp Angerer, Laleh Haghverdi, Maren Büttner, Fabian J Theis, Carsten Marr, and Florian Buettner. destiny - diffusion maps for large-scale single-cell data in R. *Bioinformatics*, 32(December 2015):btv715–, 2015. ISSN 1367-4811. doi: 10.1093/bioinformatics/btv715. URL <http://bioinformatics.oxfordjournals.org/content/early/2015/12/13/bioinformatics.btv715.abstract>.
- [55] Kieran Campbell, Chris P Ponting, and Caleb Webber. Laplacian eigenmaps and principal curves for high resolution pseudotemporal ordering of single-cell RNA-seq profiles. *bioRxiv*, page 027219, 2015. doi: 10.1101/027219. URL <http://biorxiv.org/content/early/2015/09/18/027219.abstract>.

- [56] Xiaojie Qiu, Qi Mao, Ying Tang, Li Wang, Raghav Chawla, Hannah A. Pliner, and Cole Trapnell. Reversed graph embedding resolves complex single-cell trajectories. *Nature Methods*, 14(10):979–982, 2017. ISSN 15487105. doi: 10.1038/nmeth.4402.
- [57] Gregory Giecold, Eugenio Marco, Sara P. Garcia, Lorenzo Trippa, and Guo-Cheng Yuan. Robust lineage reconstruction from high-dimensional single-cell data. *Nucleic Acids Research*, page gkw452, 2016. ISSN 0305-1048. doi: 10.1093/nar/gkw452. URL <http://nar.oxfordjournals.org/lookup/doi/10.1093/nar/gkw452>.
- [58] Miguel Juliá, Amalio Telenti, and Antonio Rausell. Sincell : an R/Bioconductor package for statistical assessment of cell-state hierarchies from single-cell RNA-seq. *Bioinformatics*, (June):btv368, 2015. ISSN 1367-4803. doi: 10.1093/bioinformatics/btv368. URL <http://bioinformatics.oxfordjournals.org/lookup/doi/10.1093/bioinformatics/btv368>.
- [59] Russell B. Fletcher, Diya Das, Levi Gadye, Kelly N. Street, Ariane Baudhuin, Allon Wagner, Michael B. Cole, Quetzal Flores, Yoon Gi Choi, Nir Yosef, Elizabeth Purdom, Sandrine Dudoit, Davide Risso, and John Ngai. Deconstructing Olfactory Stem Cell Trajectories at Single-Cell Resolution. *Cell Stem Cell*, 20(6):817–830.e8, 2017. ISSN 18759777. doi: 10.1016/j.stem.2017.04.003. URL <http://dx.doi.org/10.1016/j.stem.2017.04.003>.
- [60] Charles A. Herring, Amrita Banerjee, Eliot T. McKinley, Alan J. Simmons, Jie Ping, Joseph T. Roland, Jeffrey L. Franklin, Qi Liu, Michael J. Gerdes, Robert J. Coffey, and Ken S. Lau. Unsupervised Trajectory Analysis of Single-Cell RNA-Seq and Imaging Data Reveals Alternative Tuft Cell Origins in the Gut. *Cell Systems*, pages 1–15, 2017. ISSN 24054712. doi: 10.1016/j.cels.2017.10.012. URL <http://linkinghub.elsevier.com/retrieve/pii/S2405471217304490>.
- [61] Manu Setty, Michelle D Tadmor, Shlomit Reich-Zeliger, Omer Angel, Tomer Meir Salame, Pooja Kathail, Kristy Choi, Sean Bendall, Nir Friedman, and Dana Pe’er. Wishbone identifies bifurcating developmental trajectories from single-cell data. *Nature Biotechnology*, (April):1–14, 2016. ISSN 1087-0156. doi: 10.1038/nbt.3569. URL <http://www.nature.com/doifinder/10.1038/nbt.3569>.
- [62] Joshua Welch, Alexander Hartemink, and Jan F Prins. SLICER: Inferring Branched, Nonlinear Cellular Trajectories from Single Cell RNA-seq Data. *bioRxiv*,

- page 047845, 2016. ISSN 1474-760X. doi: 10.1101/047845. URL <http://biorxiv.org/content/early/2016/04/09/047845.abstract>.
- [63] Laleh Haghverdi, Maren Büttner, F. Alexander Wolf, Florian Buettner, and Fabian J. Theis. Diffusion pseudotime robustly reconstructs lineage branching. *Nature Methods*, 13(10):845–848, 2016. ISSN 15487105. doi: 10.1038/nmeth.3971.
- [64] Matthew E Ritchie, Belinda Phipson, Di Wu, Yifang Hu, Charity W Law, Wei Shi, and Gordon K Smyth. limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic acids research*, 43(7):e47, 2015. ISSN 13624962. doi: 10.1093/nar/gkv007.
- [65] Peter V Kharchenko, Lev Silberstein, and David T Scadden. Bayesian approach to single-cell differential expression analysis. *Nature methods*, 11(7):740–2, 2014. ISSN 1548-7105. doi: 10.1038/nmeth.2967. URL <http://www.ncbi.nlm.nih.gov/pubmed/24836921>.
- [66] Catalina A Vallejos, Davide Risso, Antonio Scialdone, Sandrine Dudoit, and John C Marioni. Normalizing single-cell RNA sequencing data: challenges and opportunities. *Nature Methods*, 14(6), 2017. ISSN 1548-7091. doi: 10.1038/nmeth.4292. URL <http://www.nature.com/doifinder/10.1038/nmeth.4292>.
- [67] Charlotte Sonesson and Mark D Robinson. Bias, robustness and scalability in single-cell differential expression analysis. *Nature Methods*, (June 2017), 2018. ISSN 1548-7091. doi: 10.1038/nmeth.4612. URL <http://www.nature.com/doifinder/10.1038/nmeth.4612>.
- [68] Rahul Satija, Jeffrey a Farrell, David Gennert, Alexander F Schier, and Aviv Regev. Spatial reconstruction of single-cell gene expression data. *Nature Biotechnology*, 33(5), 2015. ISSN 1087-0156. doi: 10.1038/nbt.3192. URL <http://www.nature.com/doifinder/10.1038/nbt.3192>.
- [69] Andrew Butler and Rahul Satija. Integrated analysis of single cell transcriptomic data across conditions, technologies, and species. *bioRxiv*, pages 1–6, 2017. doi: 10.1002/bies.201500025.
- [70] Itay Tirosh, Andrew S. Venteicher, Christine Hebert, Leah E. Escalante, Anoop P. Patel, Keren Yizhak, Jonathan M. Fisher, Christopher Rodman, Christopher

- Mount, Mariella G. Filbin, Cyril Neftel, Niyati Desai, Jackson Nyman, Benjamin Izar, Christina C. Luo, Joshua M. Francis, Aanand A. Patel, Maristela L. Onozato, Nicolo Riggi, Kenneth J. Livak, Dave Gennert, Rahul Satija, Brian V. Nahed, William T. Curry, Robert L. Martuza, Ravindra Mylvaganam, A. John Iafrate, Matthew P. Frosch, Todd R. Golub, Miguel N. Rivera, Gad Getz, Orit Rozenblatt-Rosen, Daniel P. Cahill, Michelle Monje, Bradley E. Bernstein, David N. Louis, Aviv Regev, and Mario L. Suvà. Single-cell RNA-seq supports a developmental hierarchy in human oligodendroglioma. *Nature*, 539(7628), 2016. ISSN 0028-0836. doi: 10.1038/nature20123.
- [71] Keegan D. Korthauer, Li-Fang Chu, Michael A. Newton, Yuan Li, James Thomson, Ron Stewart, and Christina Kendzierski. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. *Genome Biology*, 17(1):222, 2016. ISSN 1474-760X. doi: 10.1186/s13059-016-1077-y. URL <http://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y>.
- [72] Gioele La Manno, Ruslan Soldatov, Amit Zeisel, Emelie Braun, Hannah Hochgerner, Viktor Petukhov, Katja Lidschreiber, Maria E. Kastriti, Peter Lönnerberg, Alessandro Furlan, Jean Fan, Lars E. Borm, Zehua Liu, David van Bruggen, Jimin Guo, Xiaoling He, Roger Barker, Erik Sundström, Gonçalo Castelo-Branco, Patrick Cramer, Igor Adameyko, Sten Linnarsson, and Peter V. Kharchenko. RNA velocity of single cells. *Nature*, 560(7719):494–498, aug 2018. ISSN 0028-0836. doi: 10.1038/s41586-018-0414-6. URL <http://www.nature.com/articles/s41586-018-0414-6>.
- [73] Shila Ghazanfar, Yingxin Lin, Xianbin Su, David Ming Lin, Ellis Patrick, Ze-Guang Han, John C. Marioni, and Jean Yee Hwa Yang. Investigating higher-order interactions in single-cell data with scHOT. *Nature Methods*, 17(8):799–806, aug 2020. ISSN 1548-7091. doi: 10.1038/s41592-020-0885-x. URL <http://www.nature.com/articles/s41592-020-0885-x>.
- [74] Rafael A. Irizarry, Zhijin Wu, and Harris A. Jaffee. Comparison of Affymetrix GeneChip expression measures. *Bioinformatics*, 22(7):789–794, 2006. ISSN 13674803. doi: 10.1093/bioinformatics/btk046.

- [75] Aliaksei Z. Holik, Charity W. Law, Ruijie Liu, Zeya Wang, Wenyi Wang, Jaeil Ahn, Marie Liesse Asselin-Labat, Gordon K. Smyth, and Matthew E. Ritchie. RNA-seq mixology: Designing realistic control experiments to compare protocols and analysis methods. *Nucleic Acids Research*, 45(5):1–18, 2017. ISSN 13624962. doi: 10.1093/nar/gkw1063.
- [76] Elisabetta Mereu, Atefeh Lafzi, Catia Moutinho, Christoph Ziegenhain, Davis J. McCarthy, Adrián Álvarez-Varela, Eduard Batlle, Sagar, Dominic Grün, Julia K. Lau, Stéphane C. Boutet, Chad Sanada, Aik Ooi, Robert C. Jones, Kelly Kaihara, Chris Brampton, Yasha Talaga, Yohei Sasagawa, Kaori Tanaka, Tetsutaro Hayashi, Caroline Braeuning, Cornelius Fischer, Sascha Sauer, Timo Trefter, Christian Conrad, Xian Adiconis, Lan T. Nguyen, Aviv Regev, Joshua Z. Levin, Swati Parekh, Aleksandar Janjic, Lucas E. Wange, Johannes W. Bagnoli, Wolfgang Enard, Marta Gut, Rickard Sandberg, Itoshi Nikaido, Ivo Gut, Oliver Stegle, and Holger Heyn. Benchmarking single-cell RNA-sequencing protocols for cell atlas projects. *Nature Biotechnology*, 38(6):747–755, jun 2020. ISSN 1087-0156. doi: 10.1038/s41587-020-0469-4. URL <http://dx.doi.org/10.1038/s41587-020-0469-4><http://www.nature.com/articles/s41587-020-0469-4>.
- [77] Jiarui Ding, Xian Adiconis, Sean K. Simmons, Monika S. Kowalczyk, Cynthia C. Hession, Nemanja D. Marjanovic, Travis K. Hughes, Marc H. Wadsworth, Tyler Burks, Lan T. Nguyen, John Y. H. Kwon, Boaz Barak, William Ge, Amanda J. Kedaigle, Shaina Carroll, Shuqiang Li, Nir Hacohen, Orit Rozenblatt-Rosen, Alex K. Shalek, Alexandra-Chloé Villani, Aviv Regev, and Joshua Z. Levin. Systematic comparison of single-cell and single-nucleus RNA-sequencing methods. *Nature Biotechnology*, 38(6):737–746, jun 2020. ISSN 1087-0156. doi: 10.1038/s41587-020-0465-8. URL <http://dx.doi.org/10.1038/s41587-020-0465-8><http://www.nature.com/articles/s41587-020-0465-8>.
- [78] Luke Zappia, Belinda Phipson, and Alicia Oshlack. Exploring the single-cell RNA-seq analysis landscape with the scRNA-tools database. *bioRxiv*, page 206573, 2017. doi: 10.1101/206573. URL <https://www.biorxiv.org/content/early/2017/10/20/206573>.
- [79] Luyi Tian, Shian Su, Xueyi Dong, Daniela Amann-Zalcenstein, Christine Biben, Azadeh Seidi, Douglas J. Hilton, Shalin H. Naik, and Matthew E. Ritchie. scPipe:

- A flexible R/Bioconductor preprocessing pipeline for single-cell RNA-sequencing data. *PLOS Computational Biology*, 14(8):e1006361, aug 2018. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1006361. URL <http://www.biorxiv.org/content/early/2017/08/14/175927><https://dx.plos.org/10.1371/journal.pcbi.1006361>.
- [80] Luyi Tian, Xueyi Dong, Saskia Freytag, Kim-Anh Lê Cao, Shian Su, Abolfazl JalalAbadi, Daniela Amann-Zalcenstein, Tom S. Weber, Azadeh Seidi, Jafar S. Jabbari, Shalin H. Naik, and Matthew E. Ritchie. Benchmarking single cell RNA-sequencing analysis pipelines using mixture control experiments. *Nature Methods*, 16(6):479–487, jun 2019. ISSN 1548-7091. doi: 10.1038/s41592-019-0425-8. URL <http://www.nature.com/articles/s41592-019-0425-8>.
- [81] Lauren L. Hsu and Aedin C. Culhane. Impact of Data Preprocessing on Integrative Matrix Factorization of Single Cell Data. *Frontiers in Oncology*, 10(June), jun 2020. ISSN 2234-943X. doi: 10.3389/fonc.2020.00973. URL <https://www.frontiersin.org/article/10.3389/fonc.2020.00973/full>.
- [82] Wenpin Hou, Zhicheng Ji, Hongkai Ji, and Stephanie C. Hicks. A systematic evaluation of single-cell RNA-sequencing imputation methods. *Genome Biology*, 21(1): 218, dec 2020. ISSN 1474-760X. doi: 10.1186/s13059-020-02132-x. URL <https://genomebiology.biomedcentral.com/articles/10.1186/s13059-020-02132-x>.
- [83] Tamim Abdelaal, Lieke Michielsen, Davy Cats, Dylan Hoogduin, Hailiang Mei, Marcel J. T. Reinders, and Ahmed Mahfouz. A comparison of automatic cell identification methods for single-cell RNA sequencing data. *Genome Biology*, 20(1): 194, dec 2019. ISSN 1474-760X. doi: 10.1186/s13059-019-1795-z. URL <https://genomebiology.biomedcentral.com/articles/10.1186/s13059-019-1795-z>.
- [84] Nelson Johansen and Gerald Quon. scAlign: a tool for alignment, integration, and rare cell identification from scRNA-seq data. *Genome Biology*, 20(1):166, dec 2019. ISSN 1474-760X. doi: 10.1186/s13059-019-1766-4. URL <https://www.biorxiv.org/content/early/2018/12/22/504944?%7Dcollection=https://genomebiology.biomedcentral.com/articles/10.1186/s13059-019-1766-4>.
- [85] Ming Tang, Yasin Kaymaz, Brandon Logeman, Stephen Eichhorn, ZhengZheng S. Liang, Catherine Dulac, and Timothy B. Sackton. Evaluating single-cell cluster

- stability using the Jaccard similarity index. *bioRxiv*, page 2020.05.26.116640, 2020. doi: 10.1101/2020.05.26.116640.
- [86] Baolin Liu, Chenwei Li, Ziyi Li, Dongfang Wang, Xianwen Ren, and Zemin Zhang. An entropy-based metric for assessing the purity of single cell populations. *Nature Communications*, 11(1):3155, dec 2020. ISSN 2041-1723. doi: 10.1038/s41467-020-16904-3. URL <http://dx.doi.org/10.1038/s41467-020-16904-3><http://www.nature.com/articles/s41467-020-16904-3>.
- [87] Yungang Xu, Zhigang Zhang, Lei You, Jiajia Liu, Zhiwei Fan, and Xiaobo Zhou. scIGANs: single-cell RNA-seq imputation using generative adversarial networks. *Nucleic Acids Research*, pages 1–16, jun 2020. ISSN 0305-1048. doi: 10.1093/nar/gkaa506. URL <https://academic.oup.com/nar/advance-article/doi/10.1093/nar/gkaa506/5862684>.
- [88] Gregory W. Schwartz, Yeqiao Zhou, Jelena Petrovic, Maria Fasolino, Lanwei Xu, Sydney M. Shaffer, Warren S. Pear, Golnaz Vahedi, and Robert B. Faryabi. TooManyCells identifies and visualizes relationships of single-cell clades. *Nature Methods*, 17(4):405–413, apr 2020. ISSN 1548-7091. doi: 10.1038/s41592-020-0748-5. URL <http://dx.doi.org/10.1038/s41592-020-0748-5><http://www.nature.com/articles/s41592-020-0748-5>.
- [89] Shiyi Yang, Sean E. Corbett, Yusuke Koga, Zhe Wang, W. Evan Johnson, Masanao Yajima, and Joshua D. Campbell. Decontamination of ambient RNA in single-cell RNA-seq with DecontX. *Genome Biology*, 21(1):57, dec 2020. ISSN 1474-760X. doi: 10.1186/s13059-020-1950-6. URL <https://genomebiology.biomedcentral.com/articles/10.1186/s13059-020-1950-6>.
- [90] Allen W. Zhang, Ciara O’Flanagan, Elizabeth A. Chavez, Jamie L. P. Lim, Nicholas Ceglia, Andrew McPherson, Matt Wiens, Pascale Walters, Tim Chan, Brittany Hewitson, Daniel Lai, Anja Mottok, Clementine Sarkozy, Lauren Chong, Tomohiro Aoki, Xuehai Wang, Andrew P. Weng, Jessica N. McAlpine, Samuel Aparicio, Christian Steidl, Kieran R. Campbell, and Sohrab P. Shah. Probabilistic cell-type assignment of single-cell RNA-seq for tumor microenvironment profiling. *Nature Methods*, 16(10):1007–1015, oct 2019. ISSN 1548-7091. doi: 10.1038/s41592-019-0529-1. URL <http://dx.doi.org/10.1038/s41592-019-0529-1><http://www.nature.com/articles/s41592-019-0529-1>.

- [91] Anoushka Joglekar, Andrey Prjibelski, Ahmed Mahfouz, Paul Collier, Susan Lin, Anna Katharina, Jordan Marrocco, Stephen R Williams, Bettina Haase, Ashley Hayes, Jennifer G Chew, Neil I Weisenfeld, Man Ying Wong, Alexander N Stein, Simon Hardwick, Toby Hunt, Zachary Bent, Olivier Fedrigo, Steven A Sloan, Davide Risso, Erich D Jarvis, Paul Flicek, Wenjie Luo, and Hagen U. Tilgner. Cell-type, single-cell, and spatial signatures of brain-region specific splicing in postnatal development. *bioRxiv*, 2020.
- [92] Thomas A Ray, Kelly Cochran, Chris Kozlowski, Jingjing Wang, Graham Alexander, Martha A Cady, William J Spencer, Philip A Ruzyski, Brian S Clark, Annelies Laeremans, Ming-xiao He, Xiaoming Wang, Emily Park, Ying Hao, Alessandro Iannaccone, Gary Hu, Olivier Fedrigo, Nikolai P Skiba, Vadim Y Arshavsky, and Jeremy N Kay. Comprehensive identification of mRNA isoforms reveals the diversity of neural cell-surface molecules with roles in retinal development and disease. *Nature Communications*, 11(1):3328, dec 2020. ISSN 2041-1723. doi: 10.1038/s41467-020-17009-7. URL <http://dx.doi.org/10.1038/s41467-020-17009-7><http://www.nature.com/articles/s41467-020-17009-7>.