STATISTICAL METHODS FOR IMPROVING DATA QUALITY IN MODERN RNA SEQUENCING EXPERIMENTS

by
Zijian Ni

A dissertation submitted in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy
(Statistics)

at the

UNIVERSITY OF WISCONSIN–MADISON

2022

Date of final oral examination: May 6th, 2022

The dissertation is approved by the following members of the Final Oral Committee:
Christina Kendziorski, Professor, Biostatistics & Medical Informatics and Statistics, UW-Madison
Michael Newton, Professor, Biostatistics & Medical Informatics and Statistics, UW-Madison
Qiongshi Lu, Assistant Professor, Biostatistics & Medical Informatics, UW-Madison
Anru Zhang, Associate Professor, Biostatistics & Bioinformatics, Computer Science, Mathematics, and Statistical Science, Duke University
Beth Drolet, Professor, Dermatology, UW-Madison

This thesis is dedicated to my parents, who brought me to this beautiful world and have always loved me unconditionally throughout my life journey. This thesis is also dedicated to Shuyang, who keeps encouraging and supporting me during my doctoral study.

ACKNOWLEDGMENTS

First and foremost, a huge "Thank You" to my advisor Christina Kendziorski for her continuous support during my doctoral study and research. I've always been impressed by her wisdom in science and methodology as well as critical insights that brings out tons of amazing ideas. As the key person behind every of my scientific achievements, she has been giving unlimited supports on conceptualization, method development, application, writing, and presentation. More importantly, she always respects and encourages me to head towards the directions that fits my own research and career interests. She is one of the best advisors in human history, and being her student is one of the greatest decisions I've made during my doctoral study. I would like to give special thanks to my committee member Prof. Michael Newton, who provided key statistical advice on my thesis projects and supported and guided me in multiple collaborations. I also want to give deep thanks to the other committee members Prof. Qiongshi Lu and Prof. Anru Zhang for their helpful feedback on finalizing the theories and applications behind the statistical methods in my thesis, and Prof. Beth Drolet for her kind support on lab experiments for validating the ideas in my thesis project and for her professional insights of my work on real world applications.

I'm grateful to be part of the Kendziorski Group with my former and current peers: Prof. Rhonda Bacher, Dr. Ziyue Wang, Dr. Jared Brown, Chitrasen Mohanty, and Dr. Matthew Bernstein. I've enjoyed collaborating with them on various projects and learned a lot in their specialized fields. I also appreciate their daily discussions and critiques that helped improve my research projects.

I am very lucky and thankful to have the opportunities working with multiple groups of great collaborators across the campus. In particular, I would like to thank Dr. Ron Stewart (Morgridge Institute for Research), Dr. Aman Prasad (Department of Dermatology), Prof. Richard Halberg (Department of Medicine), Prof. Susan Thibeault and Dr. Ran An (Department of Surgery). With their helps I've gained valuable experiences and skills on interdisciplinary collaboration that are indispensable in my future careers. More importantly, I've enjoyed seeing myself solving biological and clinical problems with them that have huge impacts to benefit the real world. I want to give a special shout-out to Prof. Li-Fang Chu (University of Calgary). He was my first interdisciplinary collaborator during my doctoral study.

I really appreciated all the fun projects I've worked with him and he is the mentor of my first lesson of being a good collaborator.

Finally, I would like to express my utmost gratitude to my parents and friends for their understanding and support, especially during the COVID-19 pandemic. This has been a hard time for everyone, but things become better with their helps. Thank you to my parents for creating the opportunities for me and always supporting me to pursue my dreams. Thank you to my loved one Shuyang for encouraging me, listening to me, and sharing happiness with me.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

STATISTICAL METHODS FOR IMPROVING DATA QUALITY IN MODERN RNA SEQUENCING EXPERIMENTS

Zijian Ni

Under the supervision of Professor Christina Kendziorski
At the University of Wisconsin-Madison

Abstract

RNA sequencing (RNA-seq) has revolutionized the possibility of measuring transcriptome-wide gene expression in the last two decades. Modern RNA sequencing techniques such as single-cell RNA sequencing (scRNA-seq) and spatial transcriptomics (ST) have been developed in recent years, allowing researchers to quantify gene expression in single-cell resolution or to profile gene activity patterns in 2-dimensional space across tissue. While useful, data collected from these techniques always come with noise, and appropriate filtering and cleaning are required for reliable downstream analyses. In this dissertation, I investigate multiple quality-related issues in scRNA-seq and ST experiments, and I develop, implement, evaluate and apply statistical methods to adjust for them. A unifying theme of this work is that all these methods aim at improving data quality and allowing for better power and precision in downstream analyses.

For scRNA-seq data, the quality issue we discuss in this dissertation is distinguishing barcodes associated with real cells from those binding background noise. In droplet-based scRNA-seq experiments, raw data contains both cell barcodes that should be retained for downstream analysis as well as background barcodes that are uninformative and should be filtered out. Due to ambient RNAs presenting in all the barcodes, cell barcodes are not easily distinghished from background barcodes. Both misclassified background barcodes and cell barcodes induce misleading results in downstream analyses. Existing filtering methods test barcodes individually and consequently do not leverage the strong cell-to-cell correlation present in most datasets. To improve cell detection, we introduce CB2, a cluster-based approach for distinguishing real cells from background barcodes. As demonstrated in simulated and case study datasets, CB2 has increased power for identifying real cells which allows for the identification of novel subpopulations and improves downstream differential expression analyses.

We then present a benchmark study to evaluate the performance of cell detection methods, including CB2, on public scRNA-seq datasets covering a variety of experiment protocols. In recent years, variants of scRNA-seq techniques have been developed for specialized biological tasks. While the data structures remain the same as the standard scRNA-seq experiment, the underlying data properties can alter a lot. Here, we propose the first benchmark study to provide a thorough comparison across existing cell detection methods in scRNA-seq data, and to guide users to choose the appropriate methods for their experiments. Evaluation metrics include power, precision, computational efficiency, robustness, and accessibility. In addition, we provide investigation and guidance on appropriately choosing filtering parameters in order to improve data quality.

For ST data, we uncover, for the first time, a novel quality issue that genes expressed at one tissue region bleed out and contaminate nearby tissue regions. ST is a powerful and widely-used approach for profiling transcriptome-wide gene expression across a tissue with emerging applications in molecular medicine and tumor diagnostics. Recent ST experiments utilize slides containing thousands of spots with spot-specific barcodes that bind RNAs. Ideally, unique molecular identifiers at a spot measure spot-specific expression, but this is often not the case owing to bleed from nearby spots, an artifact we refer to as spot swapping. We design a creative human-mouse chimeric ST experiment to validate the existence of spot swapping. Spot swapping hinders inferences of region-specific gene activities and tissue annotations. In order to decontaminate ST data, we propose SpotClean, a probabilistic model that measures the spot swapping effect and estimates gene expression using EM algorithm. SpotClean is shown to provide a more accurate estimation of the underlying gene expression, increase the specificity of marker gene signals, and, more importantly, allow for improved tumor diagnostics.

1 INTRODUCTION

Determining the ways in which gene expression affects downstream phenotypes is a central question in a number of biological fields, and technologies for measuring gene expression continue to be rapidly developed. The massively parallel sequencing technology known as next-generation sequencing (NGS) has revolutionized the biological sciences. With its ultra-high throughput, scalability, and speed, NGS enables researchers to perform a wide variety of applications and study biological systems at a level never before possible. RNA sequencing (RNA-seq), one of the NGS protocols aimed at whole-transcriptome profiling to derive transcriptome-wide mRNA expression data from a population of cells, has proven useful in thousands of studies over the past two decades (Wang et al., 2009; Bacher and Kendziorski, 2016). Meanwhile, a lot of sophisticated computational methods for processing RNA-seq data have been developed to solve problems such as normalization (Anders and Huber, 2010), which corrects for non-biological artifacts during library preparation and sequencing; differential expression (DE) analysis (Love et al., 2014; Leng et al., 2013), which identifies genes that show marginal shifts in expression under different experimental conditions; and gene set enrichment analysis (Subramanian et al., 2005), which focuses on groups of genes that share common biological functions. Auer and Doerge (2010), Kukurba and Montgomery (2015), and Conesa et al. (2016) are great review papers about the design and computational analyses of RNA-seq experiments.

For practical reasons, the traditional RNA-seq experiment is usually conducted on samples comprising thousands to millions of cells (so-called bulk RNA-seq). However, this has hindered direct assessment of the fundamental unit of biology—the cell. To uncover cell-to-cell heterogeneity, single-cell RNA sequencing (scRNA-seq) has emerged as a novel technique to measure mRNA abundance in a single cell, with the scale from tens to thousands of cells in a single experiment done within a week (Tang et al., 2009; Islam et al., 2011; Macosko et al., 2015; Zheng et al., 2017). scRNA-seq allows researchers to directly investigate the activities of individual cells, study transcriptomic profiles of specific cell types, and discover unidentified cell types that were hidden in bulk RNA-seq due to pooling cells. The scRNA-seq technology has already enabled critical insights into novel subpopulations (Jaitin et al., 2014; Buettner et al., 2015), differentiation progression

(Treutlein et al., 2014; Trapnell et al., 2014), embryonic development (Xue et al., 2013; Deng et al., 2014), cancer (Patel et al., 2014; Chung et al., 2017), and neural diversity (Darmanis et al., 2015; Mathys et al., 2019). During the ongoing COVID-19 pandemic, scRNA-seq has also become a powerful tool to investigate the immune response of COVID-19 patients, accelerating the discovery and development of better diagnoses and treatments (Lee et al., 2020; Zhang et al., 2020; Stephenson et al., 2021).

Due to the novel cellular-level insights provided by scRNA-seq technologies, as well as the special data properties such as high dimensionality and sparsity, new computational methods have been developed, such as in normalization (Lun et al., 2016; Bacher et al., 2017; Brown et al., 2021), clustering (Satija et al., 2015; Kiselev et al., 2017), DE (Finak et al., 2015; Korthauer et al., 2016), and trajectory analyses (Trapnell et al., 2014; Street et al., 2018). A typical computational pipeline of scRNA-seq data starts from the raw gene-by-cell expression matrix, where rows are genes, columns are cells, and the matrix entries are the gene expression levels. Similar to bulk RNA-seq data, the expression matrix in scRNA-seq data is filtered to remove lowly expressed genes and poor-quality cells, and then normalized to remove sequencing artifacts. Following normalization, dimension reduction is performed usually by principal component analysis (PCA) using highly variable genes. The top PCs are used for clustering cells into groups with similar expression profiles. DE genes are identified for each cluster, which will then contribute to cell type annotation, gene set enrichment analysis, and regulatory network construction. The top PCs can also be used in trajectory analysis by estimating a pseudotime for each cell in order to investigate cell development and lineage hierarchies.

The increased throughput of sequencing has also fostered new experimental techniques in recent years that can directly assay the spatial context of variations in gene expression, the so-called spatial transcriptomics (ST) (Ståhl et al., 2016; Rodriques et al., 2019). Spatial resolution of gene expression is crucial for determining the functions and phenotypes of cells in multicellular organisms. Spatial expression variation can reflect communication between adjacent cells, position-specific states, or cells that migrate to specific tissue locations to perform their functions (Svensson et al., 2018). ST has been proven useful in various fields such as embryonic development (van den Brink et al., 2020; Asp et al., 2019), nephrology (Stewart and Clatworthy, 2020), neuroscience (Maynard et al., 2021), and cancer (He et al., 2020; Berglund et al., 2018; Thrane et al., 2018).

The basic computational pipeline for ST data is similar to scRNA-seq data, although novel methods have been developed using the additional spatial information. For instance, spatial variability analysis allows identifying genes with spatially varying expression across the tissue section (Svensson et al., 2018; Sun et al., 2020); spatial correlation analysis detects group of genes with spatially varying correlation (Bernstein et al., 2022); and spatial clustering incorporates spatial dependency of adjacent tissues for a clean and smooth 2-dimensional clustering result (Zhao et al., 2021). The data structure of ST data is similar to scRNA-seq data, that is, a raw gene-by-location expression matrix. In addition, the 2-dimensional coordinates of the locations are known and can be used to account for the spatial dependency of adjacent tissues. A histological image of the tissue is also often available to help with visual annotation of tissue types at different locations.

Nowadays, hundreds of computational methods have been developed to analyze scRNA-seq and ST data. However, less attention has been paid to the upstream quality control. Quality-related challenges in scRNA-seq and ST technologies must be correctly addressed to ensure powerful, efficient, and accurate downstream analyses. Towards this end, my PhD research focused on the development, validation, implementation, and application of statistical methods and software to improve the data quality in scRNA-seq and ST experiments.

Droplet-based scRNA-seq (Macosko et al., 2015; Klein et al., 2015; Zheng et al., 2017) is the state-of-the-art protocol of scRNA-seq experiments as it allows thousands of individual cells to be profiled simultaneously and efficiently. While useful, its raw gene-by-cell expression matrix is not straightforward to define, since cells are not clearly separated from background noise. As detailed in chapter 2, droplet-based scRNA-seq uses strings of nucleotides (referred to as barcodes) to estimate the abundances of genome-wide mRNA expression in individual cells. Unfortunately, the barcodes often bind to non-cellular mRNA and, consequently, an important problem in data pre-processing is distinguishing between barcodes binding mRNA from real cells versus those binding mRNA from background noise. As shown in chapter 2 and chapter 3, we developed CB2, a statistical method for accurate and powerful cell identification in droplet-based scRNA-seq data (Ni et al., 2020). The idea behind CB2 is to test groups of barcodes against the distribution of background noise using Monte-Carlo p-values. CB2 has been validated in both simulation and real data analyses, and benchmarked against other similar methods. Results suggested that CB2 achieves great power of identifying real cells while controlling false

positives. CB2 was implemented as an R package and is now publicly available.

In addition to CB2, as part of my thesis, we also identified and addressed a novel quality-related issue in recent ST technologies. As detailed in chapter 4, recent ST experiments utilize slides containing thousands of spots with spot-specific barcodes that bind mRNA. We discovered for the first time that in a typical ST experiment, mRNA at a given spot often bleeds to nearby spots, an artifact we refer to as spot swapping. Due to spot swapping, the raw gene-by-location expression matrix does not accurately measure the gene expression level at a given location. Instead, the observed expression at a given location contains a mixture of expressions from nearby locations. To address this, we developed SpotClean, a statistical method to adjust for spot swapping and recover the underlying true gene expression in ST data (Ni et al., 2021). The idea behind SpotClean is to model spot swapping using a kernel method and estimate the underlying expression by maximizing data likelihood. Simulation and real data analyses suggested that SpotClean accounts for the spot swapping effect and improves the power and precision of downstream analyses with the corrected gene expression. SpotClean was implemented as an R package and is now publicly available.

Besides these major projects, my work also involves multiple interdisciplinary collaborations in a variety of biological fields, such as method development for scRNA-seq data normalization (Brown et al., 2021), online scRNA-seq cancer database (Bernstein et al., 2021), spatially varying correlation in ST data (Bernstein et al., 2022), segmented regression of RNA-seq time course data (Bacher et al., 2018), text mining for drug discovery (Raja et al., 2020), variants-disease associations in whole-exome sequencing data (manuscript in review), as well as statistical analysis in the genomic and transcriptomic-level studies of embryonic development (Chu et al., 2019), diabetes (Nimkulrat et al., 2021), immune response (manuscript in progress), colorectal cancer (manuscript in progress), and COVID-19 (work in progress).

2 CB2 IMPROVES POWER OF CELL DETECTION IN DROPLET-BASED SINGLE-CELL RNA SEQUENCING DATA

Chapter Summary

An important challenge in pre-processing data from droplet-based single-cell RNA sequencing protocols is distinguishing barcodes associated with real cells from those binding background noise. Existing methods test barcodes individually and consequently do not leverage the strong cell-to-cell correlation present in most datasets. To improve cell detection, we introduce CB2, a cluster-based approach for distinguishing real cells from background barcodes. As demonstrated in simulated and case study datasets, CB2 has increased power for identifying real cells which allows for the identification of novel subpopulations and improves the precision of downstream analyses.

2.1 Background

Droplet-based single-cell RNA-seq (Macosko et al., 2015; Klein et al., 2015; Zheng et al., 2017) is currently the dominant single-cell RNA sequencing protocol as it is able to measure the transcriptomic profiles of thousands of cells at the same time with relatively high speed and low cost (Figure 2.1). Current commercial droplet-based technologies utilize gel beads, each containing oligonucleotide indexes made up of bead-specific barcodes combined with unique molecular identifiers (UMIs)(Islam et al., 2014) and oligo-dT tags to prime polyadenylated RNAs. Single cells of interest are combined with reagents in one channel of a microfluidic chip, and gel beads in another, to form gel-beads in emulsion, or GEMs (Figure 2.1b). Oligonucleotide indexes bind polyadenylated RNA within each GEM reaction vesicle before gel beads are dissolved releasing the bound oligos into solution for reverse transcription. By design, each resulting cDNA molecule contains a UMI and a GEM-specific barcode. Indexed cDNA is pooled for PCR amplification and sequencing resulting in a data matrix of UMI counts for each barcode (Figure 2.1c).

Ideally, each barcode will tag mRNA from an individual cell, but this is often not the case in practice. In most datasets, more than 90% of GEMs do not contain viable cells, but rather contain ambient RNA excreted by cells in solution or as a product

of cell lysis. As a result, an important challenge in pre-processing droplet-based scRNA-seq data is distinguishing those barcodes corresponding to real cells from those binding ambient, or background, RNA.

In a mathematical point of view, the raw data of such droplet-based scRNA-seq experiments is a gene-by-barcode matrix, where rows are genes, columns are barcodes, and the matrix entries are the gene expression levels measured by UMI counts. Note that this gene-by-barcode matrix is different from a gene-by-cell matrix, since there are both cell barcodes and background barcodes in the gene-by-barcode matrix. A computational method is required to filter out background barcodes in order to get the raw gene-by-cell matrix for downstream analysis.

Early methods to address this challenge defined real cells as those barcodes with total read counts exceeding some threshold (Macosko et al., 2015; Zheng et al., 2017). Such methods are suboptimal as they discard small cells as well as those expressing relatively few genes, and misclassify large background barcodes as cells. A more sophisticated method, EmptyDrops (ED) (Lun et al., 2019), identifies individual barcodes with distributions varying from a background distribution. Similar to previous approaches, ED identifies an upper threshold and defines real cells as those barcodes with counts above the threshold. As a second step, ED uses all barcodes with counts below a lower threshold to estimate a background distribution of ambient RNA against which remaining barcodes are tested. Those having expression profiles significantly different from the background distribution are deemed real cells. The ED approach is currently the most widely used in the field. However, given that ED performs tests for each barcode individually, it does not leverage the strong correlation observed between cells and, consequently, compromises power for identifying cells in many datasets.

To increase the power for identifying real cells, we developed CB2, a cluster-based approach for distinguishing real cells from background barcodes in droplet-based scRNA-seq experiments. CB2 extends the ED framework by introducing a clustering step that groups similar barcodes, then conducts a statistical test to identify groups with expression distributions that vary from the background. CB2 is implemented in the R package *scCB2* and is available at Bioconductor. section 2.2 gives details about the model. In section 2.3, simulation and real world evaluations demonstrate CB2's increased power for identifying real cells which allows for the identification of novel subpopulations and improves the precision of downstream analyses.

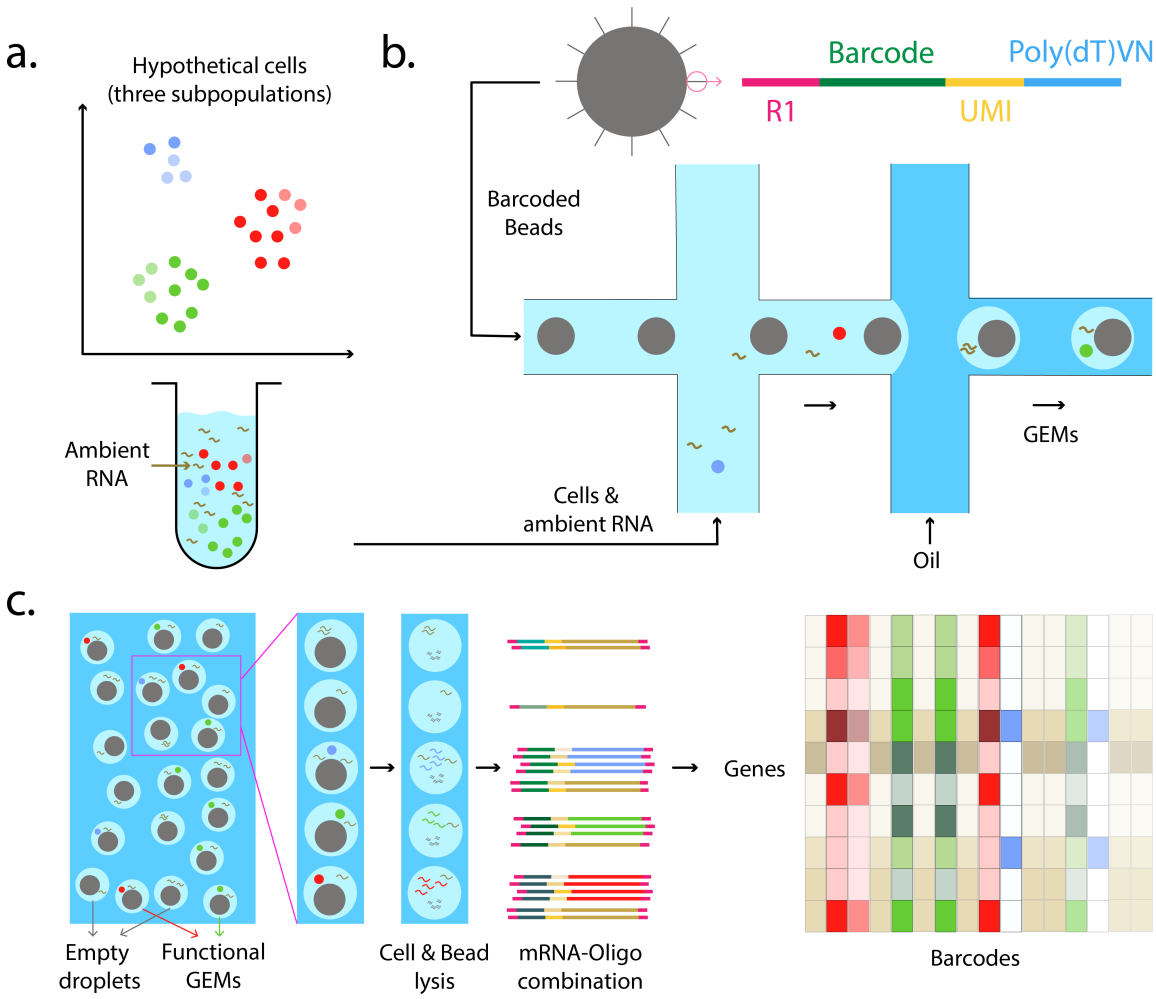

Figure 2.1: A typical droplet-based single-cell RNA-seq protocol. (a) Projection of a hypothetical cell population containing three subpopulations (red, green and blue where intensity corresponds to read depth). (b) In the 10x Chromium protocol (one of the most widely-used commercial single-cell RNA-seq protocols), gel beads containing oligonucleotide indexes made up of bead-specific barcodes combined with UMIs and oligo-dT tags to prime polyadenylated RNA are combined with single cells in one channel of a microfluidic chip and oil in another to form gel-beads in emulsion, or GEMs. (c) GEMs that capture individual cells are referred to as functional GEMs; those that fail to capture cells are empty droplets. Gel beads dissolve and release their oligos for reverse transcription of polyadenylated RNAs. Indexed cDNA is pooled for PCR amplification and sequencing to give a data matrix of UMI counts for each barcode.

2.2 Methods

In this section, we provide a detailed description of the CB2 model. Figure 2.2 shows an overview of the CB2 framework. CB2 expects as input a $G \times B$ gene-by-barcode matrix with G genes and B barcodes (Figure 2.2a). Barcodes having zero counts for all genes are filtered out, and the remaining barcodes are divided into three groups based on the sum of gene expression (UMI) counts within a barcode (Figure A.1). The background group, B_0, contains all barcodes with counts less than or equal to a pre-defined lower threshold (defaults to 100); the high-count barcodes, B_2, contain barcodes with counts exceeding an upper threshold (defaults to knee point); the remaining barcodes, B_1, are to be tested. Each barcode is assumed to independently follow a Multinomial distribution with parameters (N, p) where N denotes the total UMI counts of the barcode, and p is the probability vector with length G.

We assume that counts from a background barcode are distributed with probability vector p_{B_0} estimated by averaging the counts in B_0 and applying the Good-Turing algorithm (Gale and Sampson, 1995) to ensure that all probabilities are non-zero, denoted as $\hat{p}_{B_0}$. For a barcode $b \in B_1$ with probability vector p_b, our task is to test if it is a background barcode or cell barcode, that is, if $p_b = p_{B_0}$ or not. This setting is similar to ED. However, ED tests all barcodes from B_1 individually, while CB2 first clusters barcodes and then tests tight clusters to identify those that differ from the background (Figure 2.2b). As in methods for genome-wide association studies (Mieth et al., 2016), gene co-expression network analysis (Botía et al., 2017), and de novo transcriptome analysis (Malik et al., 2018), clustering prior to testing increases power by reducing the total number of tests and increasing the signal to noise ratio. CB2 proceeds as follows:

1. Barcodes grouped by size: CB2 orders barcodes in B_1 by total counts

$$B_1 = \{b_1, \ldots, b_{B_1}\} \quad \text{s.t.} \quad |X_{b_i}| \leqslant |X_{b_{i+1}}|$$

   where X_b denotes the count vector of barcode b, $|X_b|$ denotes the total UMI counts of barcode b, and $|B_1|$ denotes the number of barcodes in B_1. Groups of size S (defaults to 1000) are constructed consisting of barcodes ranging in size from smallest to largest:

$$B_{11} = \{b_1, \ldots, b_S\}, B_{12} = \{b_{S+1}, \ldots, b_{2S}\}, \cdots, B_{1K} = \left\{b_{(K-1)S+1}, \ldots, b_{|B_1|}\right\}$$

where $K = \frac{|B_1|}{S}$ is rounded up if not an integer. If $|B_{1K}| < \frac{S}{2}$, barcodes in B_{1K} are merged with those in $B_{1(K-1)}$. Sorting barcodes by size reduces bias in the clustering and testing steps that follow.

2. Barcodes clustered within group: Barcodes within each group B_{1j} are clustered using hierarchical clustering with pairwise Pearson correlation as the similarity metric. A cluster is considered tight if the average within-cluster pairwise Pearson correlation exceeds a data-driven threshold. Tight clusters are retained for further analysis as described in step 3, below. To determine thresholds, ten tight clusters of varying size are simulated by generating 100 samples from a Multinomial distribution with parameters (N, p) where N ranges from 100 to 1000 in increments of size 100. This range is chosen as we found little variation in thresholds for barcode sizes exceeding 1000; p is set to either p^{B_0} or p^{B_2}, whichever has larger Shannon entropy (Shannon, 1948) as the distribution with larger entropy is less affected by outlier genes. For each simulated cluster C, the threshold κ_C is defined by its average pairwise Pearson correlation. A cluster is considered tight if the average within-cluster pairwise Pearson correlation exceeds κ_C for the simulated cluster of closest size.

3. Tight clusters tested: For each tight cluster C, we conduct a Monte-Carlo test to assess dissimilarity from the background. Pairwise Pearson correlations are calculated between every barcode in C and $\hat{p}_{B_0}$; the test statistic for cluster C, T_C, is defined to be the median of these correlations. Similar to ED, to simulate background barcodes, we sample barcodes $X_1^*, \ldots, X_M^*$ from a Multinomial $(N, \hat{p}_{B_0})$ where N is the size of the barcode giving T_C. The Monte-Carlo p-value is:

$$p_C = \frac{\sum_{i=1}^{M} \left\{ cor_{X_i^*,0} \leqslant T_C \right\} + 1}{M + 1}$$

where $cor_{X_i^*,0}$ is the Pearson correlation between X_i^* and $\hat{p}_{B_0}$ (M defaults to 1000). Monte-Carlo p-values are calculated for each cluster followed by Benjamini-Hochberg (Benjamini and Hochberg, 1995) to control the FDR. All barcodes within a significant cluster are identified as real cells.

4. Individual barcodes tested: Barcodes that were not included in a tight cluster in Step 2 as well as those in a tight cluster that were not found to be significant in Step 3 are tested individually using ED. It is important to note that some of the barcodes identified in this step do not overlap with identifications made

when ED is applied to the full set of barcodes given differences in the rates of real cells to background barcodes and differences in error rate control.

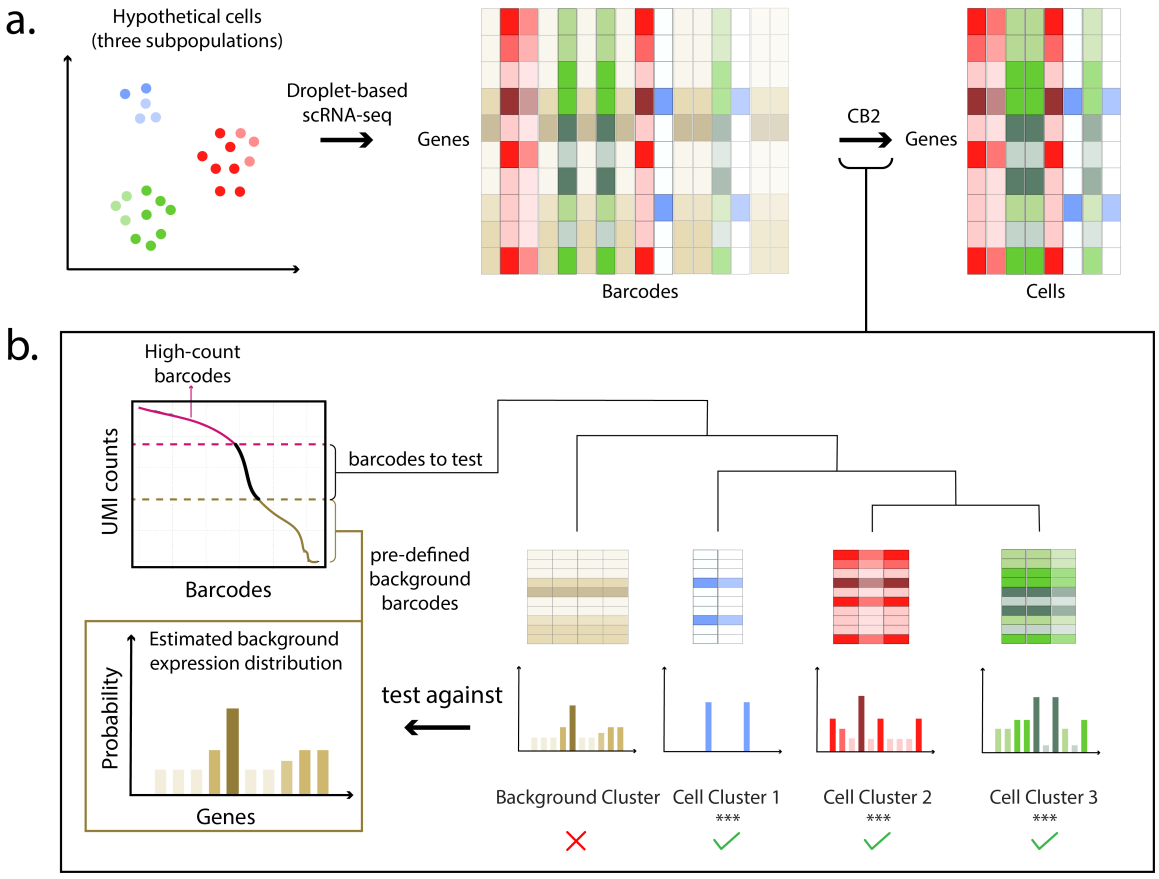

Figure 2.2: Overview of CB2. (a) Projection of a hypothetical cell population containing three subpopulations (red, green, and blue where intensity corresponds to read depth). CB2 takes as input a gene-by-barcode matrix of UMI counts and returns a gene by cell matrix. (b) High-count barcodes with counts above a pre-specified upper threshold are considered real cells; barcodes with counts below a lower threshold are used to estimate a background distribution (Figure A.1). The remaining barcodes are clustered, and tight clusters are tested as a group against the estimated background distribution; barcodes not in tight clusters are tested individually (not shown). High-count barcodes and those identified by CB2 are retained for downstream analysis.

2.3 Results

CB2 was evaluated and compared with ED on simulated and case study data.

Simulation set-up

We followed the same simulation framework as in Lun et al. (2019). Specifically, each simulated gene-by-barcode matrix is based on an input real world dataset. We constructed simulations from 10 datasets: Alzheimer (Mathys et al., 2019), PBMC8K, PBMC33K, mbrain1K, mbrain9K, PanT4K, MALT, PBMC4K, jurkat, and T293 (access to these datasets are available at Table A.1). For each input dataset, the inflection point of the barcode rank plot (plotting the total UMI counts against its rank for every barcode) is used to divide lower count from higher count barcodes. Lower count barcodes are pooled to calculate a background probability vector, where the vector length equals the number of genes. Background barcodes are simulated from Multinomial distributions with varying total counts and the same background probability vector. For each lower count barcode in the input dataset, a matched background barcode will be simulated with the same total UMI counts. As a result, the number and the UMI counts distribution of the simulated background barcodes match those of the low count barcodes in the input dataset.

Next, we have two simulations frameworks for generating real cells. In SIM IA, 2000 large cell barcodes (G1), 2000 medium cell barcodes (G2), and 2000 small cell barcodes (G3) are simulated. G1 cells were randomly drawn with replacement from the higher count barcodes of the input dataset. G2 and G3 cells are simulated similarly, but their UMI counts are further downsampled by 50% and 90% to give medium and small cells. The process for simulating data in SIM IB is identical to SIM IA except that in SIM IB, 10% of the genes in each simulated real cell are shuffled making the real cells more different from the background barcodes and, consequently, making real cells easier to identify. SIM IA is a more realistic simulation since the similarity between background barcodes and real cells closely resembles that in real world data (Figure 2.3). SIM IB is evaluated since it's the simulation setting in ED.

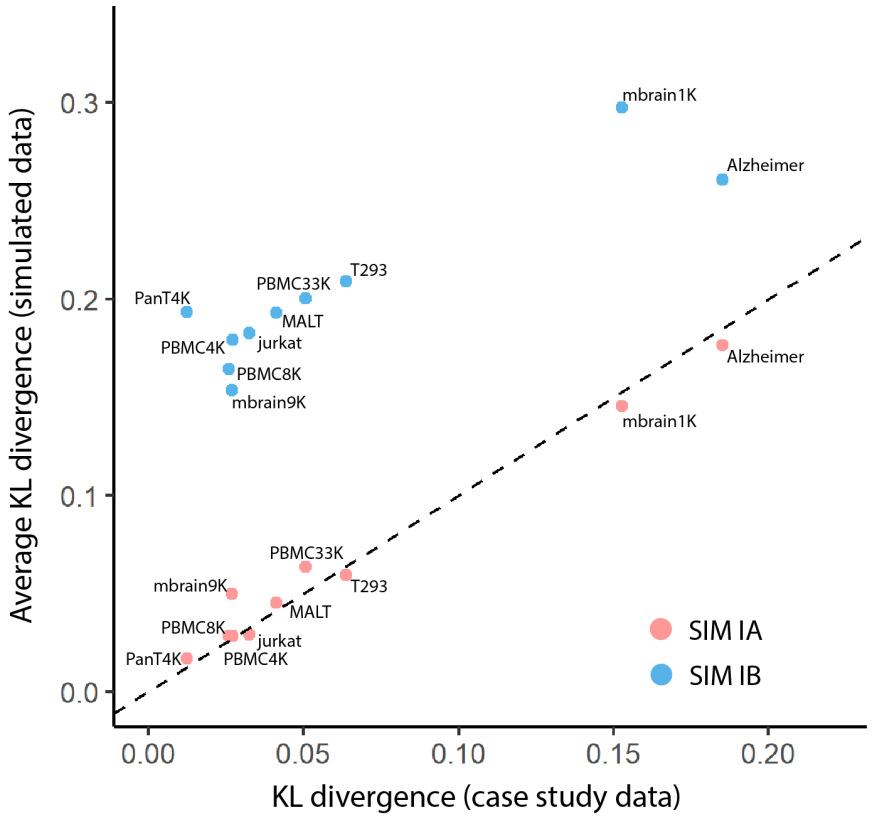

Figure 2.3: To measure the similarity between each simulated dataset and the case study data from which it is derived, for each dataset we evaluate the extent to which the distribution of real cells differs from the background distribution. Specifically, for each case study dataset, we calculate the Kullback-Leibler (KL) divergence to measure the difference between the expression distribution of the high-count barcodes and the background distribution. For the simulated data, we calculate the KL divergence between the G1 simulated cells and the background distribution. This is repeated to get a KL divergence for G2 simulated cells vs. background and G3 simulated cells vs. background. The average KL divergence (averaged over the three groups) for each simulation (SIM IA, SIM IB) is plotted against the KL divergence for each case study. The KL divergence in the SIM IA data is more similar to the case study data for each dataset considered. The increased KL divergence observed in the SIM IB data indicates that the SIM IB simulated barcodes differ from the background more than observed in the case study or SIM IA data, which makes differences easier to identify when applying CB2 or ED.

Simulation results

CB2 and ED were applied to the simulated data to calculate the power (number of simulated real cells that are successfully detected as real cells over number of simulated real cells) and FDR (number of simulated background barcodes that are falsely detected as real cells over number of detected real cells). Each simulation is repeated 5 times and the average performance is reported.

In SIM IA, Figure 2.4 shows increased power of CB2 with well controlled FDR for the 6 datasets considered in Lun et al. (2019) as well as 4 additional datasets. Specifically, CB2 shows leading power in all the three cell groups among all the ten datasets, especially in G1 and G2 groups. For both methods, the power decreases with cell size, since cell size (total UMI counts) is the sample size of both testing methods. CB2 also shows comparable FDR as ED, and even lower in a few datasets. The observed FDR is below the target FDR threshold (1%) in nine out of ten datasets. SIM IB is similar to SIM IA, but in SIM IB 10% of the genes in the real cells are shuffled making the real cells more different from the background and therefore easier to identify (Figure 2.3). Figure A.2 shows the increased power of CB2 is maintained. These results suggest that CB2 has an universally better performance than ED.

Additional simulations were conducted to evaluate the robustness of CB2 and ED under different lower thresholds. Recall that both CB2 and ED rely on a lower threshold to define the background group B_0 and estimate background distribution $\hat{p}_{B_0}$. Instead of using the default threshold 100, SIM IA was repeated with thresholds equal to 50 and 150. Figure A.3 and Figure A.4 show similar results as using the default threshold, indicating that both CB2 and ED are robust to different choices of lower threshold under the simulation setting. Note that in real world data, the lower threshold needs to be carefully chosen for an unbiased estimation of the background distribution. This will be further discussed in chapter 3.

Figure 2.4: The average power and average false discovery rate (FDR) of CB2 and ED for SIM IA data (average taken over 5 simulated datasets). Since both CB2 and ED automatically identify high count barcodes as real cells (they are not subject to statistical test; Figure A.1), we report results for all barcodes as well as those tested by CB2 and ED. The top panel shows the average power for tested barcodes; the middle panel for tested as well as high count barcodes. The bottom panel shows the average FDR. For the PanT4K dataset, all G1 cells are above the upper threshold and so no barcodes were tested (as a result, power for tested barcodes is not defined).

Case study

To further evaluate CB2, we applied CB2 and ED to the ten case study datasets used to generate the simulated data as well as one additional dataset considered in the ED case study and compared the number of cells identified in common as well as those uniquely identified by each approach. Both CB2 and ED were applied to each dataset using their default settings to get the gene-by-cell matrices.

Normalization was then performed using scran (Lun et al., 2016) to adjust for sequencing depth. The Seurat (Satija et al., 2015) pipeline was used to cluster cells and generate t-SNE plots from the top 4000 most highly variable genes and top 50 principal components. Mitochondrial and ribosomal genes were removed for better visualization, since a few of them usually have much higher expression levels, making it harder to investigate other genes. We define subpopulations as the clusters from the Seurat pipeline. For each subpopulation, we calculated the percentage of cells commonly identified by both CB2 and ED as well as those identified uniquely by CB2. Subpopulations for which over 80% of the cells are uniquely identified by CB2 are referred to as novel subpopulations.

Table 2.1 shows that CB2 finds 24% more cells on average (range 4%-81%). Of the extra cells identified, 88% on average (range 44%-100%) add to existing subpopulations. The remaining 12% (range 0%-56%) make up novel subpopulations. Table A.2 shows additional results using different thresholds for defining subpopulation.

As an example, Figure 2.5 and Figure A.5 show results from the Alzheimer data (Mathys et al., 2019) where CB2 identifies 18% more cells. A detailed look at the unique CB2 identifications suggests that the extra cells identified are not false positives, but rather they add to existing excitatory neuron and inhibitory neuron sub-populations, and also reveal a novel subpopulation consisting of 209 cells. Specifically, Figure 2.5b and Figure 2.5c show distribution plots and an expression heatmap of the 100 genes having the highest average expression in Subpop1 (the largest subpopulation) for cells identified by both CB2 and ED as well as those identified uniquely by CB2. As shown, cells uniquely identified by CB2 have a distribution similar to other cells, and they differ from the background. Using the marker genes from Mathys et al. (2019), Figure 2.5d and Figure A.5b suggest that cells identified uniquely by CB2 in Subpops 1-4 are neurons, as they show relatively high expression of neuron marker genes SYT1, SNAP25, and GRIN1. More specifically, the CB2 cells in Subpops 1-2 exhibit high expression of excitatory neuronal markers whereas the cells in Subpops 3-4 appear to be inhibitory neurons (Figure A.5c-d). The novel subpopulation (Subpop5) uniquely shows high expression of both oligodendrocyte and astrocyte marker genes, suggesting that this group may be mixed phenotype glial cells (Dyer et al., 2000) (Figure A.5e-f).

In the t-SNE plot (Figure 2.5a), cells uniquely identified by CB2 are grouped with common cells belonging to the same subpopulations, but are not well mixed with them. The reason is that t-SNE plots are sensitive to total counts (Townes

et al., 2019). By testing on clusters of barcodes, CB2 improves the power to identify cells having relatively low UMI counts and as a result many of the cells identified uniquely by CB2 have lower counts than those identified in common by both CB2 and ED. Figure 2.6a shows the total UMI counts distribution of cells identified by CB2 and ED. Most CB2 unique cells have counts between 200 and 2000, lower than most of the common cells identified by both CB2 and ED. Figure 2.6b shows a comparison between the t-SNE plot generated using the raw data of Subpop1 and the t-SNE plot where barcodes are downsampled so that all identifications have the same total UMI counts. After removing the effect of total counts, CB2 unique cells are well-mixed with the cells identified in common by both CB2 and ED. These results indicate that the non-mixture is due to different total counts.

By increasing the number of real cells identified, CB2 also improves the power to differentiate Alzheimer's patients from controls. Specifically, Mathys et al. (2019) profiled expression from the prefrontal cortex of 24 AD-pathology patients as well as 24 age-matched controls, and they validated differentially expressed (DE) genes in different cell types, including 9 genes in excitatory neurons and 9 in inhibitory neurons. In our analysis, we followed the procedure as in Mathys et al. (2019) (Wilcoxon rank-sum tests between cells from Alzheimer's cases and controls), and compared the DE results between DE cells and CB2 cells. Figure 2.7 shows that by identifying additional cells, CB2 improves downstream differential expression analysis by resulting in more significant p-values and stronger fold changes.

In a second case study (PBMC8K), CB2 increases the number of cells identified across six subpopulations by over 80% (Table 2.1). Results are shown in Figure 2.8 and Figure A.6. Similar to the Alzheimer's data analysis, Figure A.6b and Figure A.6c show that cells identified uniquely by CB2 in Subpop1 have an expression profile that is similar to other cells, and differs from the background. Figure 2.8 provides a detailed look at marker gene expression for the well characterized PBMC8K cells using markers considered in Zheng et al. (2017). As shown in Figure 2.8b, CB2 identifies additional CD14+ Monocytes, T-cells, B-cells, and megakaryocytes. Results from two additional datasets are shown in Figure A.7 and Figure A.8.

Dataset	High-count cells (untested)	Tested cells identified by both CB2 and ED	Cells uniquely identified by CB2	Cells uniquely identified by ED	CB2 unique cells in existing subpopulation	CB2 unique cells in novel subpopulation
Alzheimer	12143	57278	10689 / 57278 (18.66%)	50 / 57278 (0.09%)	6819 / 10689 (63.79%)	3870 / 10689 (36.21%)
PBMC8K	6708	1445	1165 / 1445 (80.62%)	2 / 1445 (0.14%)	1165 / 1165 (100%)	0 / 1165 (0%)
PBMC33K	23491	11762	424 / 11762 (3.60%)	0 / 11762 (0%)	424 / 424 (100%)	0 / 424 (0%)
mbrain1K	581	1469	221 / 1469 (15.04%)	16 / 1469 (1.09%)	166 / 221 (75.11%)	55 / 221 (24.89%)
mbrain9K	6048	5685	1265 / 5685 (22.25%)	98 / 5685 (1.72%)	1057 / 1265 (83.56%)	208 / 1265 (16.44%)
PanT4K	3398	1700	261 / 1700 (15.35%)	0 / 1700 (0%)	261/ 261 (100%)	0 / 261 (0%)
MALT	3378	981	494 / 981 (50.36%)	2 / 981 (0.20%)	216 / 494 (43.72%)	278 / 494 (56.28%)
PBMC4K	2145	6516	1003 / 6516 (15.39%)	0 / 6516 (0%)	1003 / 1003 (100%)	0 / 1003 (0%)
jurkat	2565	953	175 / 953 (18.36%)	0 / 953 (0%)	175 / 175 (100%)	0 / 175 (0%)
T293	2299	797	48 / 797 (6.02%)	2 / 797 (0.25%)	48 / 48 (100%)	0 / 48 (0%)
placenta	4349	2947	637 / 2947 (21.62%)	1 / 2947 (0.03%)	637 / 637 (100%)	0 / 637 (0%)

Table 2.1: The number of cells identified by CB2, ED, or both in case study datasets.

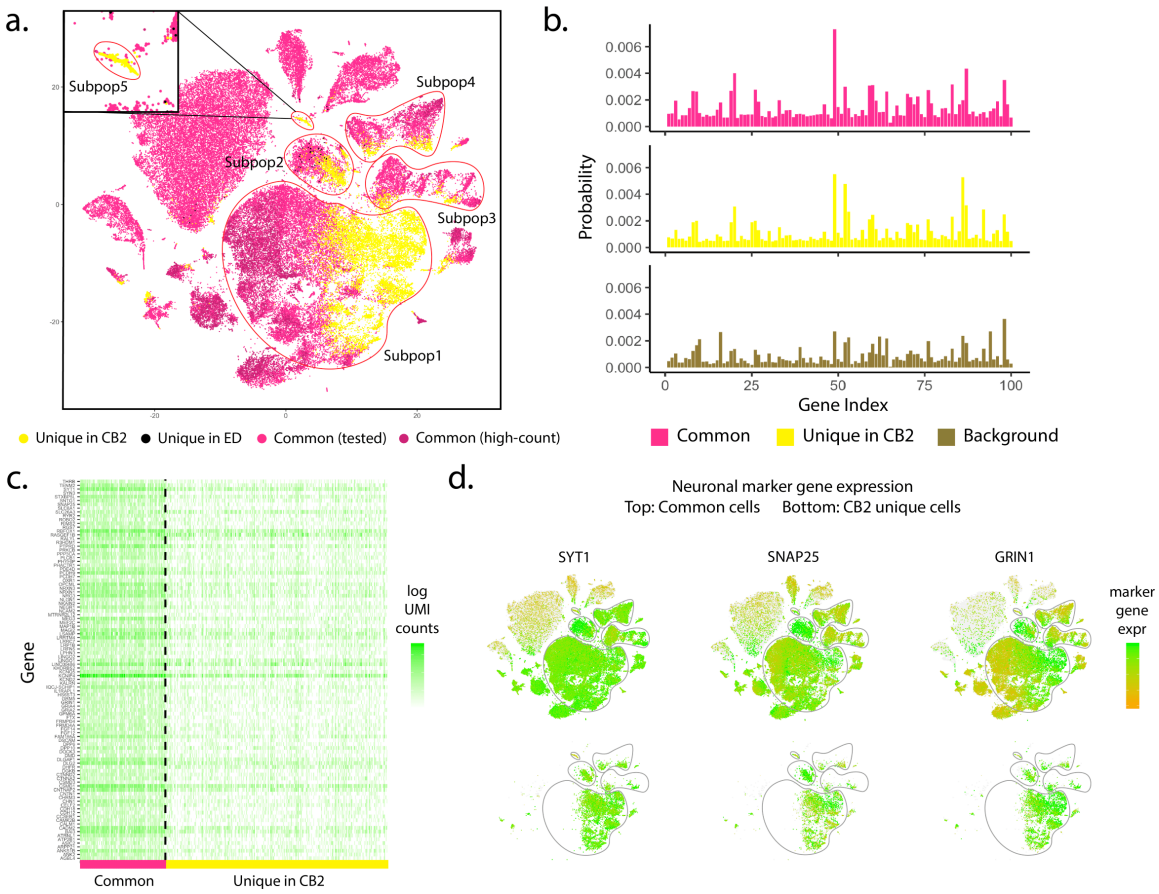

Figure 2.5: Results from the Alzheimer dataset. (a) t-SNE plot of cells identified by CB2 and EmptyDrops. High-count barcodes exceeding an upper threshold are identified as real cells by both methods without a statistical test (dark pink); barcodes identified as cells by both methods following statistical test are shown in pink. Cells identified uniquely by CB2 (yellow) and ED (black) are also shown. CB2 identifies an increased number of cells in existing sub-populations (Subpop1 – Subpop4) and also identifies a novel subpopulation (Subpop5). (b) Distribution plots of the 100 genes having highest average expression in Subpop1 are shown for cells identified by both CB2 and ED (upper) and identified uniquely by CB2 (middle). The estimated background distribution is also shown (lower). Cells uniquely identified by CB2 in Subpop1 have a distribution similar to other Subpop1 cells and differ from the background. (c) Heatmap of log transformed raw UMI counts for the same 100 genes for barcodes identified by CB2 and ED (left) and barcodes uniquely identified by CB2 (right). (d) t-SNE plots of cells colored by neuron marker genes SYT1, SNAP25, and GRIN1 in all cells (upper) and those identified uniquely by CB2 (lower).

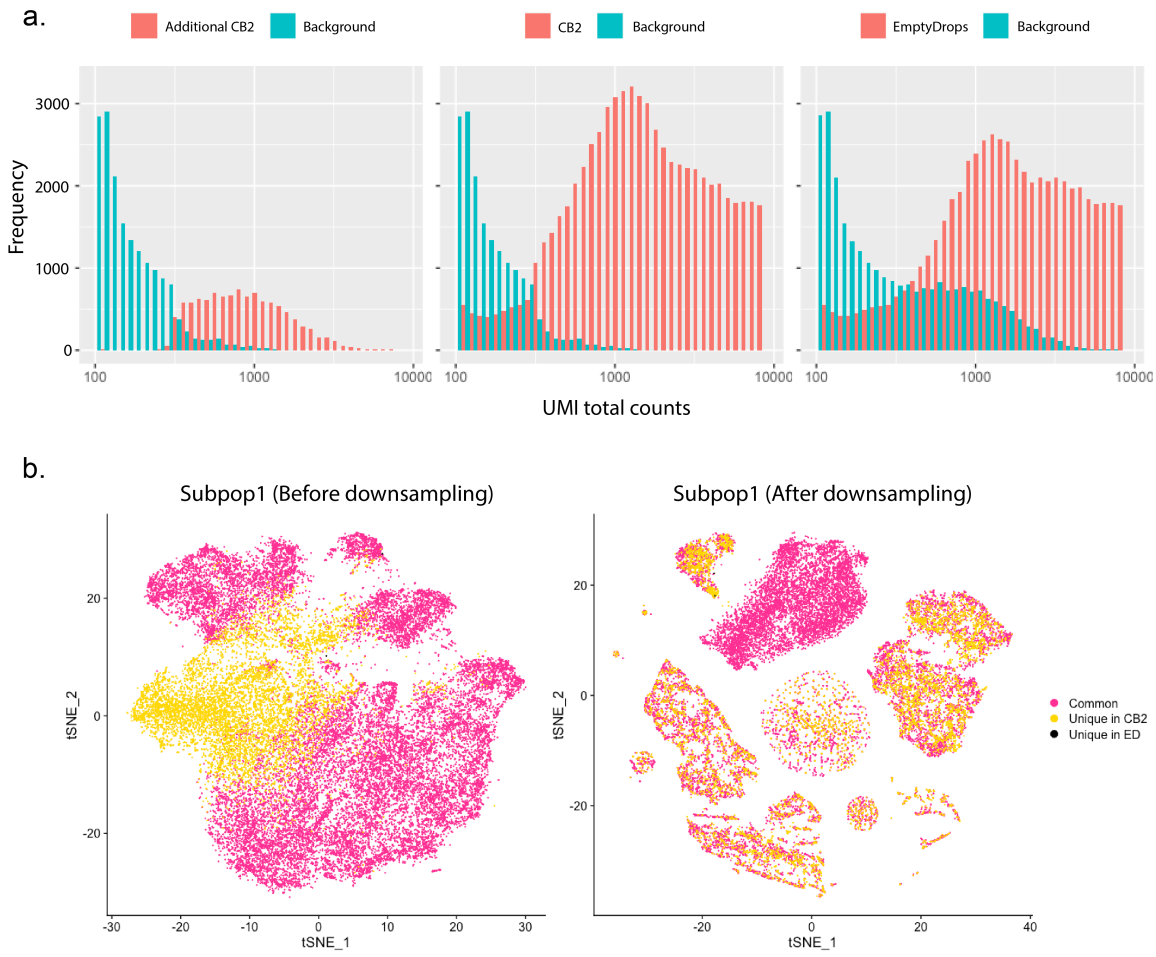

Figure 2.6: (a) Distribution of the number of cells and background barcodes identified uniquely by CB2 (left), identified in total by CB2 (middle), and identified by ED (right). The x-axis is log10 scaled ranging from lower threshold (100) to upper threshold (for the Alzheimer dataset, the upper threshold is 8136) since barcodes outside this range are not tested by either CB2 or ED. (b) t-SNE plot for Subpop1 before (left) and after (right) down-sampling. After down-sampling, the effect of total counts is removed, and CB2 cells are well-mixed with common cells.

Figure 2.7: Differential Expression analysis between Alzheimer's disease (AD) cases and controls was conducted using cells identified by CB2 (salmon) and ED (turquoise). Shown in the upper panels are the -log10 p-values for 9 genes known to be differentially expressed between AD-pathology and control cells in excitatory neurons (left: GOLT1B, ATF6B, DDRGK1, TUBB2A, BEX2, ATPIF1, RASGEF1B, NGFRAP1, LINGO1) and 9 genes known to be differentially expressed in inhibitory neurons (right: TCEAL4, SPCS1, FBXO2, COX4I1, ATPIF1, SOD1, NGFRAP1, TMSB4X, NDUFA4). Log2 fold changes of the mean expression in AD-pathology vs. control cells are also shown (lower panels) for each gene in each cell type. Given that some of the unique CB2 identifications are expressing stronger cell-type-specific marker genes (Figure A.5b-e), fold changes in the CB2 identified cells are more extreme. CB2 improves downstream DE analysis by showing more significant p-values and stronger fold changes.

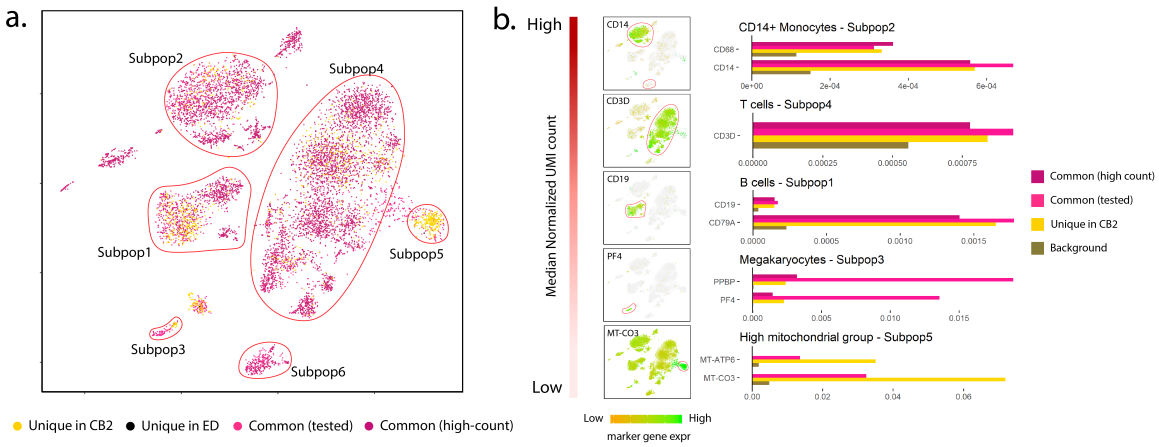

Figure 2.8: Results from the PBMC8K dataset. (a) t-SNE plot of cells identified by CB2 and ED. High-count barcodes exceeding an upper threshold are identified as real cells by both methods without a statistical test (dark pink); barcodes identified as cells by both methods following statistical test are shown in pink. Cells identified uniquely by CB2 (yellow) and ED (black) are also shown. CB2 increases the number of cells identified across the six subpopulations by over 80% Table 2.1. (b) Subpopulations 1-5 ordered by median normalized UMI count along with marker gene expression for each subpopulation. Marker gene expression in cells uniquely identified by CB2 is similar to that in other groups, and differs from the background. Subpopulation 5 contained no high count common cells; subpopulation 6 contained no unique CB2 identifications and is therefore not shown in panel (b).

2.4 Implementation

CB2 is implemented as an R package *scCB2*, which is publicly available at Bioconductor. The source code and vignette can be found at `http://www.bioconductor.org/packages/release/bioc/html/scCB2.html`.

The input data of *scCB2* is a gene-by-barcode matrix. Given that most droplet-based scRNA-seq experiments are conducted using the 10x Chromium platform (`https://www.10xgenomics.com/products/single-cell-gene-expression`), *scCB2* provides functions to directly read the raw files from 10x Chromium's computational pipeline and get the gene-by-barcode count matrix in R. For other platforms, users need to manually create the gene-by-barcode matrix.

The major step of *scCB2* is to filter out background barcodes and retain real cells. Besides the raw gene-by-barcode matrix, *scCB2* requires input of target FDR threshold, lower threshold, and upper threshold. The target FDR threshold controls how conservative the cell calling is. A lower FDR threshold produces fewer misclassified background barcodes, but also compromises power. The lower threshold divides the smallest barcodes into background to estimate the background distribution, and the upper threshold divides the largest barcodes to be known real cells. The default lower threshold is 100, based on empirical observation (Lun et al., 2019). The default upper threshold is estimated from the data (Figure A.1). The output contains both the filtered gene-by-cell matrix as well as summary statistics through the algorithm. Summary statistics contain (1) testing statistics (Pearson correlation to the background), p-values, and adjusted p-values for all candidate barcode clusters, (2) barcode IDs for all candidate barcode clusters, with the cluster name being its median barcode size, (3) test statistics (log likelihood under background distribution), p-values, and adjusted p-values for the remaining single barcodes not clustered, (4) the estimated background distribution as a numeric vector.

After the major cell calling step, *scCB2* provides functions to easily extract the gene-by-cell matrix from the output and perform additional filtering to remove low quality cells. The quality of a given cell is measured by the proportion of UMI counts from mitochondrial genes. Low quality cells (usually broken or dying cells) tend to have a higher mitochondrial concentration. The default filtering cutoff is 0.25, meaning a cell will be filtered out if its proportion of mitochondrial UMI counts exceeds 25%. *scCB2* also provides a wrapper function to connect the output with popular downstream computational pipelines (Satija et al., 2015).

scCB2 is implemented efficiently and runs in parallel mode. The computation time is usually less than 10 minutes for a typical dataset. We also provide an all-in-one function to run CB2 under default settings to skip most of the coding part.

2.5 Discussion and future work

The results presented in this chapter demonstrate that CB2 provides a powerful approach for distinguishing real cells from background barcodes which will increase the number of cells identified in existing cell subpopulations in most datasets and may facilitate the identification of novel subpopulations. While advantages are expected in many settings, users will benefit from the following considerations. CB2 does not test for doublets or multiplets and, consequently, some of the high count identifications may consist of two or more cells. Methods for identifying multiplets such as Scrublet (Wolock et al., 2019) or DoubletFinder (McGinnis et al., 2019) may prove useful after applying CB2. A second important post-processing step is filtering based on mitochondrial expression. As noted in Lun et al. (2019), any method for distinguishing cells from background barcodes is technically correct in identifying low-quality cells given that damaged cells exhibit expression profiles that differ from the background. Specifically, mitochondrial gene expression is often high in damaged cells; an example is shown in Subpopulation 5 of the PBMC8K data (Figure 2.8b). Such cells are typically not of interest in downstream analysis and should therefore be removed. The *GetCellMat*() function in *scCB2* may be used toward this end. In addition, the ambient RNAs do not only exist in background barcodes, but also in cell barcodes. Computational methods such as SoupX (Young and Behjati, 2020) and DecontX (Yang et al., 2020) are designed to remove ambient RNA contamination in cell barcodes. They may also prove useful alongside CB2; and they are further discussed in chapter 4.

Droplet-based scRNA-seq technologies provide unprecedented opportunity to address biological questions, but efficient pre-processing is required to maximize the information obtained in an experiment. CB2 allows investigators to maximize the number of cells retained, and consequently to increase the power and precision of downstream analysis.

3 BENCHMARKING CELL DETECTION ALGORITHMS FOR DROPLET-BASED SINGLE-CELL RNA SEQUENCING DATA

Chapter Summary

Droplet-based single-cell RNA-seq technology allows researchers to investigate transcriptome-wide gene expression at single-cell resolution across thousands of cells simultaneously. An important challenge in pre-processing data from droplet-based single-cell RNA-seq protocols is distinguishing barcodes associated with real cells from those binding ambient RNAs. In this study, we benchmarked four state-of-the-art computational methods for detecting cell barcodes: EmptyDrops, CB2, DIEM, and dropkick. Their performances were evaluated in both simulation and real world studies covering a variety of droplet-based single-cell RNA-seq experiments. In addition, we provide investigation and guidance on appropriately choosing filtering parameters in order to improve data quality.

3.1 Background

Droplet-based single-cell RNA-seq (Macosko et al., 2015; Klein et al., 2015; Zheng et al., 2017) is currently the most widely used single-cell RNA sequencing protocol as it allows researchers to quantify transcriptome-wide gene expression in thousands of cells at single-cell resolution. In chapter 2, we provided a detailed description of the experiment and a typical challenge that cell barcodes (barcodes representing cell droplets) and background barcodes (barcodes representing empty droplets) are not easily distinguishable due to the presence of ambient RNAs in the raw data (section 2.1). We developed a statistical method, CB2 (Ni et al., 2020), that identifies cell barcodes from background barcodes (section 2.2), and we showed that CB2 outperforms existing methods, namely EmptyDrops (Lun et al., 2019), and improves downstream computational analyses (section 2.3).

During the time CB2 came out, other computational methods for real cell detection were also under development. Debris Identification using Expectation Maximization (DIEM) is a computational method to quantify contamination and filter droplets in scRNA-seq and single-nucleus RNA-seq (snRNA-seq) experiments (Alvarez et al., 2020); snRNA-seq and scRNA-seq are very similar except that snRNA-

seq only quantifies RNAs within the nucleus instead of all RNAs in the whole cell. Droplet-based snRNA-seq have the same data structure as droplet-based scRNA-seq, and the snRNA-seq experiments also contain background barcodes that need to be removed from real nuclei barcodes. As a result, most computational methods, including CB2, can be applied to both scRNA-seq and snRNA-seq data.

DIEM first clusters barcodes using a multinomial mixture model. To estimate the parameters of the mixture model, DIEM performs semi-supervised expectation maximization by fixing barcodes that fall below a threshold of 100 counts as debris. The majority of these barcodes are assumed to contain ambient RNA. After fitting the model, DIEM assigns barcodes to clusters based on their posterior probability. For scRNA-seq data, barcodes in the debris clusters are considered as background and are removed. When applied to snRNA-seq data, barcodes are further scored based on their expression of genes enriched in the debris set, and are filtered based on their individual scores to remove background barcodes. DIEM is a semi-supervised clustering approach as it relies on a background threshold to get a predefined set of background barcodes and guide the debris annotation. This is similar to EmptyDrops and CB2, and will be discussed in section 3.4. DIEM has been compared with EmptyDrops in Alvarez et al. (2020), but there is no existing evaluation between DIEM and CB2.

dropkick (Heiser et al., 2021) is another computational method for real cell identification in scRNA-seq data. dropkick uses weakly supervised machine learning to build a model of single-cell gene expression in order to score and classify barcodes as real cells or background noise. dropkick first thresholds barcodes into three groups based on the number of expressed genes: a lower level containing uninformative barcodes (which are thrown away), an upper level containing barcodes with very high cell probability, and an intermediate level that consists of both background barcodes with high UMI counts and small real cells with relatively low UMI counts. The upper and intermediate barcode populations are labeled as real cells and putative empty droplets, respectively, and are used as the training set. dropkick then fits a logistic regression model with elastic net regularization using the gene expression matrix and barcode labels of the training set. Barcodes in the training set are re-labeled as real cells or empty droplets by thresholding on the fitted probabilities of being real cells. dropkick has been compared with EmptyDrops in Heiser et al. (2021), but there is no existing evaluation between dropkick and CB2.

In recent years, variants of droplet-based scRNA-seq techniques have been developed for specialized biological tasks. For example, single-cell targeted gene expression (`https://www.10xgenomics.com/products/targeted-gene-expression`) sequences a predefined subset of genes, such as cancer-related or immune-related genes, instead of the whole transcriptome, reducing sequencing costs by as much as 90%, or scaling up to a 10-fold increase in sample throughput. Low throughput sequencing (`https://support.10xgenomics.com/single-cell-gene-expression/software/pipelines/latest/what-is-LT`) is another cost-effective solution for smaller-scale and pilot studies for profiling whole transcriptomes at the single cell level for 100 - 1,000 cells per sample. While the data structures remain the same as the standard scRNA-seq experiment, the underlying data properties can change substantially. Given that existing cell detection methods were evaluated mainly using the standard scRNA-seq data, it is unclear whether they are robust under these new techniques.

We here present a benchmark study of cell detection methods in droplet-based scRNA-seq data. Specifically, we benchmarked EmptyDrops, CB2, DIEM, and dropkick (Table 3.1), the four state-of-the-art methods, on 17 public datasets covering a variety of droplet-based scRNA-seq techniques. Evaluation metrics include power, precision, computational efficiency, robustness, and accessibility. section 3.2 demonstrates the evaluation metrics. section 3.3 shows the performance of the four methods. section 3.4 summarizes the performance and makes user recommendations. Overall, we find that CB2 and DIEM outperform EmptyDrops and dropkick. This is the first benchmark study to provide a thorough comparison across existing cell detection methods in droplet-based scRNA-seq data, and to guide users to choose the appropriate methods for their experiments. In addition, we report the effect of over- or under-estimating the background threshold in EmptyDrops, CB2, and DIEM in section 3.3, and provide guidance on optimizing the choice of background threshold in section 3.4.

Method	Description	Built in	Reference
EmptyDrops	Estimate a background distribution using small barcodes, then test the remaining barcodes individually against the background distribution using a Monte-Carlo approach.	R	Lun et al. (2019)
CB2	Estimate a background distribution using small barcodes, then cluster the remaining barcodes and test the clusters against the background distribution using a Monte-Carlo approach.	R	Ni et al. (2020)
DIEM	Cluster barcodes into "debris" and different cell types using a multinomial mixture model estimated via the EM algorithm, then keep non-debris barcodes as cells. Designed for single-nucleus RNA-seq data, but also works for single-cell RNA-seq data.	R	Alvarez et al. (2020)
dropkick	Pre-label barcodes as putative empty droplets or cells based on barcode quality, then fit a logistic regression between gene expressions and barcode labels, and classify cell barcodes by thresholding the fitted probability.	Python	Heiser et al. (2021)

Table 3.1: A brief summary of the four cell detection methods compared in our study.

3.2 Benchmark metrics

Power and precision are two direct metrics to evaluate cell detection accuracy. In the simulation studies, power is the number of simulated real cells that are correctly detected as real cells over the number of simulated real cells, and precision is the number of simulated real cells that are correctly detected as real cells over the number of detected real cells. In real world data, unknown ground truth makes it harder to evaluate power and precision, and we instead compare the number of detected cells and their overlap across methods as an approximation. We also validate barcode identities by comparing the gene expression distribution with known background or cell barcodes as another way of approximation.

A good method needs to be computationally efficient to be useful in practice. Computational efficiency is here quantified by the running time of producing the final cell/background labels given the raw barcode matrix. In addition to computational efficiency, robustness is also critical for a method to be generalized to variants of scRNA-seq experiments, where data carry similar structures but different distributions. Our benchmark is conducted using datasets from different experimental protocols to test if a method works consistently well under different data distributions. Finally, we also briefly discuss the accessibility of each method in terms of whether the method implementation is publicly available and easy to install, whether there are tutorials with runnable examples, and whether the overall implementation is accessible to researchers with limited computational backgrounds.

3.3 Results

In this section, we will show benchmark results of the four methods in simulation and case studies. We will also investigate the effect of over- or under-estimating the background threshold (the key parameter in EmptyDrops, CB2, and DIEM) on general cell detection performance.

Simulation set-up

We first compare the performance of EmptyDrops, CB2, DIEM and dropkick in simulated datasets. The simulation setting is the same as SIM IA described in section 2.3. Briefly, given an input dataset, an inflection point dividing low from high count barcodes is determined. Low count barcodes are pooled to estimate the background distribution. Three groups of cell barcodes, G1, G2, G3, are sampled from the high count barcodes. The UMI counts in the G2 and G3 groups are downsampled by 50% and 90% to simulate cells with different sizes. We applied the simulation framework on 12 publicly available single-cell RNA-seq datasets from 10x Genomics, covering a wide range of datatypes including the standard Chromium data, targeted gene expression data where only a defined set of transcripts are profiled, and low throughput data targeting fewer cells and sequences. EmptyDrops, CB2, DIEM, dropkick were applied to the simulated data under their default settings to calculate the power (the proportion of cells that are successfully detected among

the simulated cells) and FDR (the proportion of background barcodes among the barcodes detected as cells, which is 1 minus precision). Each simulation is repeated 5 times and the average performance is reported.

Simulation results

Figure 3.1 shows the power and FDR for each of the four methods. In general, the power decreases when the cell sizes become smaller due to the reduced number of UMI counts and increased sparsity. The four methods have comparable power on the standard Chromium datasets; CB2 is more robust when generalizing to targeted gene expression data and low throughput data, showing increased power compared with the other three methods. EmptyDrops, CB2 and DIEM control FDR in general, while dropkick misclassifies many background barcodes into cell barcodes in two Chromium datasets (PanT4K and PBMC4K).

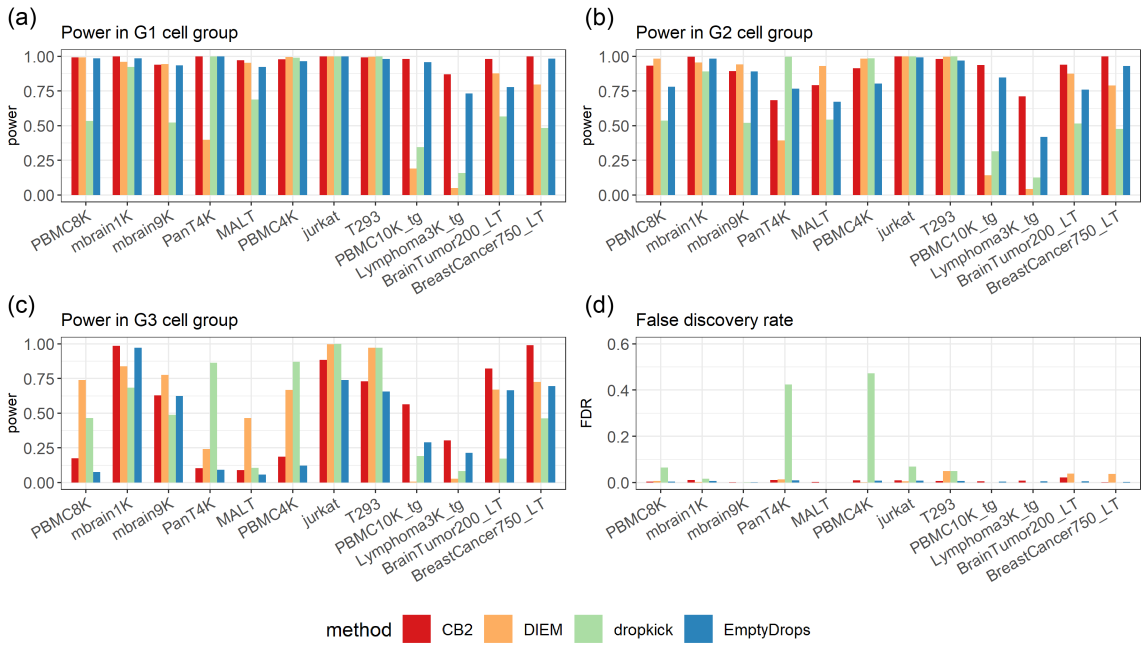

Figure 3.1: Performance of the four methods in the simulated datasets. The targeted gene expression datasets have the suffix "_tg". The low throughput datasets have the suffix "_LT". (a)-(c) The power of the four methods across different simulated datasets for large cells (G1 cell group), medium cells (G2 cell group), and small cells (G3 cell group). (d) The false discovery rate of the four methods across different simulated datasets.

Case study

The general cell detection performance of the four methods was also evaluated in real datasets. Specifically, for each of the 12 datasets, the four methods were applied on the raw count matrix under their default settings. Each method provided a set of cell barcodes, which were then used as inputs for the UpSet plot to evaluate the overlap among these barcodes. For CB2, DIEM and EmptyDrops, the choices of background thresholds were optimized as discussed later in this section. Cells with high mitochondrial gene expression are likely of low quality and were filtered out. Specifically, a cell barcode was excluded if the proportion of UMI counts from mitochondrial genes (gene names starting with "MT-" for human and "mt-" for mouse) exceeded 25%.

Figure B.1 and Figure B.2 show the UpSet plots of cells detected by each method when applied to the 12 real datasets used previously in the simulation study. Similar to the simulation results, the four methods showed comparable performance on most standard Chromium datasets. dropkick is not robust as the number of cells detected by dropkick disagrees the most with other methods, and dropkick is underpowered on the targeted gene expression datasets and low throughput datasets. The default DIEM pipeline failed at the PCA step when applied to the targeted gene expression datasets and another standard scRNA-seq data (PanT4K).

For a detailed evaluation of real world performance, we applied EmptyDrops, CB2, DIEM, and dropkick on a large single-cell RNA-seq dataset from an Alzheimer disease study (Mathys et al., 2019). Specifically, the four methods were applied on the raw count matrix under their default settings. The filtered count matrix containing the union of these four cell barcode sets was processed for downstream analyses. We applied scran (Lun et al., 2016) for data normalization. The normalized matrix then went through the Seurat pipeline (Satija et al., 2015) including variable feature selection, scaling, dimension reduction and clustering. Clusters were further annotated into excitatory neurons (ExN), inhibitory neurons (InN),oligodendrocytes (Olig), astrocytes (Ast), microglias (Mic), endothelial cells (End), and oligodendrocyte progenitor cells (OPCs) using marker genes reported in Mathys et al. (2019).

Figure 3.2a shows the UMAP plot of the union of cell barcodes from the four methods colored by annotated cell types, and Figure B.3 highlights the cell barcodes detected by each method. The UpSet plot in Figure 3.2b shows the overlap of the four cell barcode sets. dropkick detected less than 40,000 cells, which is much lower

than the other three methods, and is much lower than the number of cells (75,060) reported in the original study. Cells detected by EmptyDrops are mostly a subset of the cells detected by CB2. Given that the comparison between EmptyDrops and CB2 has been conducted in Ni et al. (2020) and dropkick is clearly underpowered, we focus on the distinction between CB2 cells and DIEM cells. There are 5095 cells detected in CB2 but not in DIEM, 1537 cells detected in DIEM but not in CB2, and 69034 cells detected in both methods. Figure 3.2c shows that both CB2 extra cells and DIEM extra cells spread across different cell types and do not form isolated clusters. A detailed investigation suggests that both CB2 extra cells and DIEM extra cells are real cells that add to existing cell types in the common cells. Specifically, Figure 3.2b and Figure 3.2c show distribution plots and an expression heatmap of the 50 genes having the highest average expression in oligodendrocytes for CB2 extra cells, DIEM extra cells, as well as common cells identified by both CB2 and DIEM. As shown, both CB2 extra cells and DIEM extra cells have distributions similar to the common cells representing the majority of oligodendrocytes, and they differ from the background. Although the extra cells in one method are likely false negatives for the other method, CB2 shows better power than DIEM since CB2 identifies more real cells (745 extra cells) than DIEM (183 extra cells). Similar results are shown in Figure B.4 for microglia and excitatory neurons.

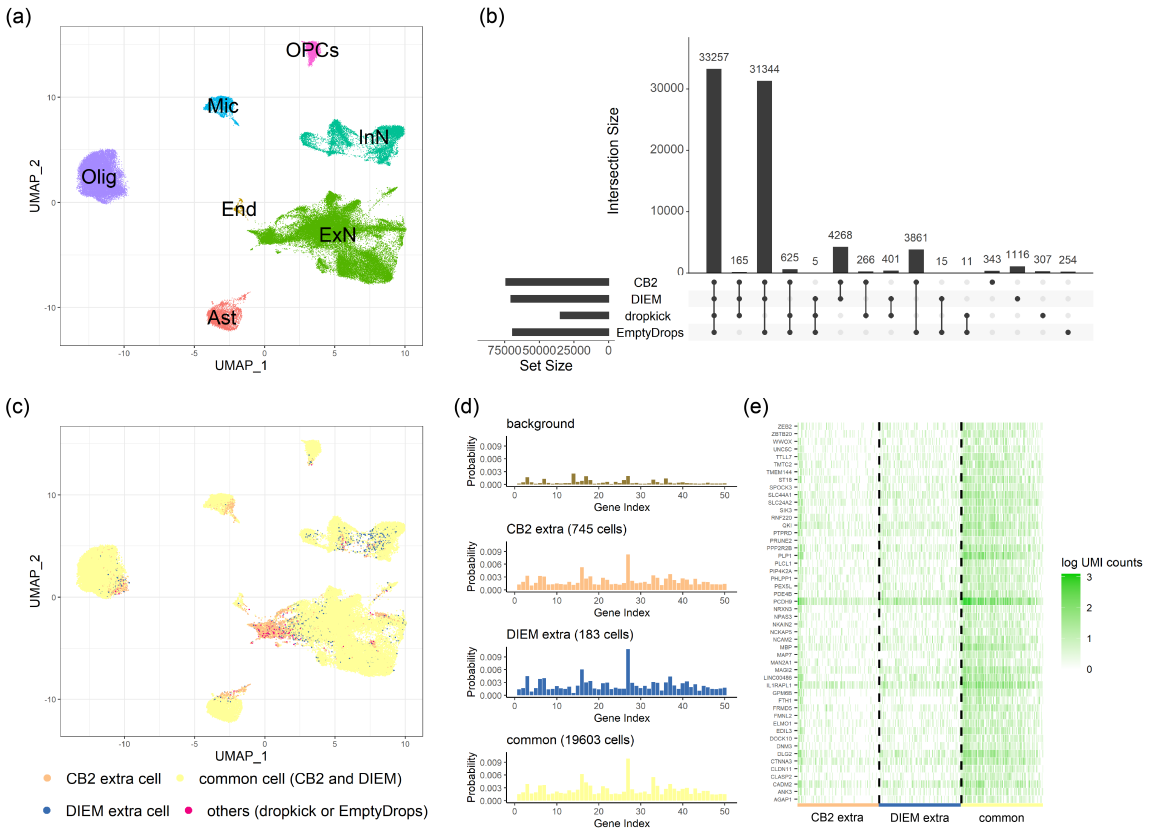

Figure 3.2: Performance of the four methods in the Alzheimer data. (a) UMAP plot of the union of the four cell barcode sets from the four methods colored by cell type annotations. Ast: Astrocytes. End: Endothelial cells. ExN: Excitatory neurons. InN: Inhibitory neurons. Mic: Microglia. Olig: Oligodendrocytes. OPCs: Oligodendrocyte progenitor cells. (b) UpSet plot showing the number of common and distinct barcodes of the four cell barcode sets as well as the size of the sets. (c) The same UMAP plot as in (a) but colored by common (common cell barcodes in CB2 and DIEM), CB2 extra (barcodes identified as cells in CB2 but not in DIEM), DIEM extra (barcodes identified as cells in DIEM but not in CB2), others (barcodes identified as cells in EmptyDrops or dropkick but not in CB2 or DIEM). (d) Distribution plots of the 50 genes having highest average expression in the common cells are shown for background barcodes, CB2 extra cells, DIEM extra cells, and common cells in Oligodendrocytes. (e) Heatmap of log transformed raw UMI counts for the same 50 genes for CB2 extra cells, DIEM extra cells, and common cells in Oligodendrocytes.

Computational efficiency

We further evaluated the computational efficiency of the four methods when applied to small and large scRNA-seq datasets. The computational efficiency is defined as the time a method takes processing a dataset from the raw barcode matrix to the final cell matrix. Two small datasets (mheart1K, PBMC1K) and two large datasets (mheart10K, PBMC10K) generated from 10x Chromium were selected to evaluate the computational efficiency of the four methods. The small and large datasets have the same number (6,794,880) of raw barcodes and similar number (~30,000) of genes, but the expected number of cells in the small datasets is 1,000, while the expected number of cells in the large datasets is 10,000. The four methods were applied under their default settings, and repeated 5 times in each dataset. The average time over the 10 runs (5 replicates in the 2 small datasets) were reported as the computational efficiency in small datasets. This is similarly defined for large datasets.

Table 3.2 shows that all four methods finish running on small size datasets in less than 2 minutes. DIEM is the most efficient method that takes less than half a minute. CB2 is the slowest method that takes about 2 minutes. For large size datasets , EmptyDrops is the most efficient method that finishes in 2 minutes. dropkick is the slowest method with more than 8 minutes running time. We also stress the fact that the computational efficiency of DIEM and dropkick seems to be linearly associated with the number of cells, while CB2 and EmptyDrops are more efficient than linear. This indicates that CB2 and EmptyDrops are more suitable for datasets with larger sizes.

Method	Small dataset (~1k cells, ~7m barcodes, ~30k genes)	Large dataset (~10k cells, ~7m barcodes, ~30k genes)
EmptyDrops	1.12	1.93
CB2	1.89	7.61
DIEM	0.48	4.95
dropkick	0.73	8.15

Table 3.2: Average running time of the four methods in small and large datasets. Unit: minutes.

Accessibility

A user-friendly implementation is essential for a method to be widely adopted. Here we briefly evaluate the accessibility of the four methods. CB2 is implemented in an R package *scCB2* and is publicly available at Bioconductor (Gentleman et al., 2004). The installation of *scCB2* is straightforward using Bioconductor's package management functions. The package contains a well-written tutorial with built-in toy example datasets, and an all-in-one function for non-computational users to skip most coding challenges. EmptyDrops is implemented in an R package *DropletUtils* and is also publicly available at Bioconductor. *DropletUtils* also has tutorials and example datasets. However, this package is a quality control toolkit and is not designed solely for cell calling. As a result, it does not illustrate the usage of cell calling in a detailed way. DIEM is implemented in an R package *diem* and is available at GitHub. *diem* contains a step-by-step tutorial with built-in datasets; *diem* also has a multi-stage processing pipeline, which can be challenging for non-computational users. dropkick is implemented in a Python package *dropkick* and is available at GitHub. *dropkick* also contains detailed tutorials and example datasets, and an integrated function to reduce the amount of coding.

Choice of background threshold

The background threshold is the key parameter required for defining the background distribution in EmptyDrops, CB2, and DIEM. Barcodes with total UMI counts less than the background threshold are considered known background bar-

codes, since they are assumed to be too small to be real cells. The default background threshold is set to be 100 in the three methods, which has empirically worked reasonably well in most standard high-quality scRNA-seq datasets. However, following the default setting does not always guarantee an accurate estimation of the background distribution, especially for datasets with special properties. A larger-than-optimal threshold treats small cells as background, resulting in biased background estimation and loss of small cells. A smaller-than-optimal threshold retains fewer background barcodes, increasing the variability of background estimation.

Table 3.3 provides an example of a smaller-than-optimal threshold. Specifically, Table 3.3 shows the proportion of barcodes and the proportion of total UMIs defined as background under background threshold=100 across 12 public datasets. All datasets except BreastCancer750_LT have close to or more than 90% barcodes and 1% UMIs within background, while BreastCancer750_LT only has 24.5% barcodes and 0.6% UMIs within background. A detailed investigation indicates that the default threshold fails to provide a reliable estimation of the background distribution in BreastCancer750_LT. First, the proportions of barcodes and UMIs in the background in BreastCancer750_LT are significantly lower than the other datasets. Increasing the background threshold to 300 yields 90.8% barcodes and 7.7% UMIs in the background. Next, we consider barcodes identified as cells with UMI counts between 101 and 300 by EmptyDrops (446 barcodes), CB2 (1906 barcodes), and DIEM (81 barcodes) under background threshold=100. Figure B.5 shows that these barcodes have similar overall distributions across methods, and they are similar to the barcodes with UMI counts below 100. A few mitochondrial genes are enriched in barcodes with UMI counts below 100 compared with barcodes with UMI counts between 101 and 300. This indicates the default threshold is too low to accurately capture background properties, especially overestimating the expression levels of mitochondrial genes,resulting in false positives during cell detection. Finally, the number of cells detected using threshold=300 are closer to the expected number of cells (750) in this dataset. Specifically, 722, 794, and 686 barcodes are detected as cells by EmptyDrops, CB2, and DIEM using threshold=300.

The default threshold can also be larger-than-optimal, especially when only a subset of the mRNAs within cells are measured. This is the case for the targeted expression data, which only captures a subset of genes instead of the whole transcriptome. For a given dataset, the amount of mRNAs measured can be approximated by the average number of reads per cell. In standard and low throughput

datasets, the average number of reads per cell ranges between 32,721 and 93,552. However, PBMC10K_tg and Lymphoma3K_tg only have 13,211 and 10,149 reads per cell (Table 3.3). Since the number of reads is positively correlated with the number of UMI counts for a given cell, the default threshold becomes too high in the targeted gene expression data. By lowering down the threshold to 10 instead of 100, EmptyDrops and CB2 detected 9755 and 10256 cells instead of 9623 and 10035 cells in PBMC10K_tg, and detected 2954 and 2967 cells instead of 2775 and 2792 cells in Lymphoma3K_tg. DIEM results are not shown since the default DIEM pipeline failed in the targeted expression data. Figure B.6 shows cells with UMI counts between 11 and 100 have different distributions from the background and are likely real cells. These small cells will be incorrectly filtered out under the default threshold.

dataset	#barcodes	#UMIs	%barcodes	%UMIs	#reads per cell
PBMC8K	409508	45009903	96.5815	11.1309	93,552
mbrain1K	231912	13829862	98.4675	10.2765	56,718
mbrain9K	562550	81890428	89.4852	4.8677	41,998
PanT4K	366341	21676973	98.4684	16.4551	73,864
MALT	937146	99500932	91.6421	1.3864	32,721
PBMC4K	272442	22034737	98.0587	12.2969	87,433
jurkat	296060	54260620	98.4993	5.9341	33,851
T293	290463	49089174	97.8858	8.5477	33,405
PBMC10K_tg	226588	11383824	95.5068	5.4203	13,211
Lymphoma3K_tg	108705	2663899	97.3718	9.2122	10,149
BrainTumor200_LT	9540	2622757	95.283	10.4816	54,761
BreastCancer750_LT	10612	16748716	24.4817	0.6192	62,379

Table 3.3: Number (#) and proportion (%) of barcodes below the background threshold=100, number and proportion of UMI counts within these barcodes under background threshold=100, and number of reads per cell for different real datasets.

3.4 Discussion and future work

A first step for preprocessing droplet-based single-cell RNA-seq data is to identify droplets containing cells from empty droplets. Here we benchmarked four state-of-the-art computational methods - EmptyDrops, CB2, DIEM, and dropkick - for the task of cell identification. Their performances were summarized in Table 3.4. Specifically, CB2 and DIEM achieved the strongest power in evaluations of simulations and real data. EmptyDrops and CB2 produced lower false positives, since both methods were built based on statistical testing where the FDR can be easily controlled. DIEM and dropkick had the best computational efficiency in small datasets, and EmptyDrops was faster in large datasets. All four methods are

efficient enough to control the computation time within 10 minutes in a typical scRNA-seq dataset. EmptyDrops and CB2 achieved robust performance when generalizing to variants of scRNA-seq experiments, such as targeted gene expression data and low throughput data. In contrast, the default DIEM pipeline failed in some datasets which required manual parameter adjustment, and dropkick was unstably underpowered, resulting in much fewer identified cells compared with the other three methods. CB2 and dropkick are the most user-friendly methods with simple installation and running commands as well as detailed step-by-step tutorials.

Additional filtering and cleaning is required to achieve better data quality. After identifying real cells from empty droplets, it is recommended to further filter the cell matrix to remove low quality cells (e.g. broken or dying cells) or cell doublets (droplets containing more than one cell). High proportion of mitochondrial gene expression is usually an indication of low quality, and filtering based on mitochondrial proportion cutoff is now widely used. A practical cutoff is 25%, meaning cells with more than 25% UMI counts coming from mitochondrial genes can be filtered out as low quality cells. However, this cutoff should be manually investigated for datasets with special properties, such as cells with high mitochondria activities or single-nucleus RNA-seq data. Cell doublets may also be present following cell isolation if more than one cell is captured into one droplet; multiple computational methods have been developed to identify and remove these doublets (McGinnis et al., 2019; Wolock et al., 2019).

Selecting an appropriate background threshold is crucial for EmptyDrops, CB2, and DIEM to accurately estimate the background distribution. There is no simple rule to find the optimal threshold, and in most cases researchers use 100 by default based on empirical performance. However, the optimal threshold is always data-driven, and sanity checks are required before simply following the default settings. As discussed in section 3.3, the proportion of barcodes and UMIs below the threshold can be used to assess whether the threshold is under-estimated; the number of reads per cell is also useful when assessing whether the threshold should be scaled. These criteria are easy to investigate, and we recommend always checking them in order to determine if the default background threshold should be changed.

Method	Power	FDR control	Speed (small dataset)	Speed (large dataset)	Robustness	Accessibility
EmptyDrops	2	1	2	1	1	2
CB2	1	1	3	3	1	1
DIEM	1	2	1	2	2	2
dropkick	2	3	1	3	3	1

Table 3.4: Summary of the performances of EmptyDrops, CB2, DIEM, and dropkick. They are ranked (1 is the best) in different categories for users to choose based on their own needs.

4 SPOTCLEAN ADJUSTS FOR SPOT SWAPPING IN SPATIAL TRANSCRIPTOMICS DATA

Chapter Summary

Spatial transcriptomics is a groundbreaking and widely-used approach for profiling transcriptome-wide gene expression across a tissue with emerging applications in molecular medicine and tumor diagnostics. Recent spatial transcriptomics experiments utilize slides containing thousands of spots with spot-specific barcodes that bind mRNA. Ideally, unique molecular identifiers at a spot measure spot-specific expression, but this is often not the case in practice owing to bleed from nearby spots, an artifact we refer to as spot swapping. We propose SpotClean to adjust for spot swapping and, in doing so, to increase the power and precision with which downstream analyses are conducted.

4.1 Background

Spatial transcriptomics (ST) is a groundbreaking and widely-used approach for profiling transcriptome-wide gene expression across a tissue (Ståhl et al., 2016; Stickels et al., 2021). In a typical ST experiment, fresh-frozen (or FFPE) tissue is sectioned and placed onto a slide containing spots, with each spot containing millions of capture oligonucleotides with spatial barcodes unique to that spot. The tissue is imaged, typically via Hematoxylin and Eosin (H&E) staining. Following imaging, the tissue is permeabilized to release mRNA which then binds to the capture oligonucleotides, generating a cDNA library consisting of transcripts bound by barcodes that preserve spatial information. Data from an ST experiment consists of the tissue image coupled with RNA sequencing data collected from each spot. A first step in processing ST data is tissue detection, where spots on the slide containing tissue are distinguished from background spots without tissue. Unique molecular identifier (UMI) counts at each spot containing tissue are then used in downstream analyses (Figure C.1).

Ideally, a gene-specific UMI at a given spot would represent expression of that gene at that spot. This is not the case in practice. As we demonstrate here, messenger RNAs bleed between and among nearby spots causing substantial contamination of

UMI counts, an artifact we refer to as spot swapping.

Spot swapping is related to, but distinct from, previously defined sources of contamination which have been widely recognized over the past decade in next-generation sequencing studies (Kircher et al., 2012). Specifically, improvements in sequencing technologies have greatly increased the speed and scale at which data can be obtained, but the advantages rely on multiplexing where indexes (or barcodes) are attached to each mRNA (or DNA) fragment in a sample prior to pooling so that sample-specific transcripts can be identified in the sequenced pool. In spite of the major advantages in reduced cost and increased efficiency, a disadvantage is that indexes from one sample may attach to transcripts from another at random, an error referred to as index hopping or barcode swapping. While present in most datasets, good statistical methods are in place to adjust for this type of contamination (Kircher et al., 2012; Griffiths et al., 2018; Larsson et al., 2018; Costello et al., 2018). Barcode swapping is distinct from the spot swapping artifact detailed here since spot swapping is not at random. Rather, with spot swapping, the probability of a spot-specific barcode binding reads from another spot increases as the distance between spots decreases. As the statistical methods developed to adjust for barcode swapping do not accommodate the spatial dependence inherent in spot swapping, they are not sufficient in this setting.

A second type of contamination is specific to single-cell RNA sequencing (scRNA-seq) experiments. In droplet based scRNA-seq, for example, each droplet ideally contains one cell, and barcodes specific to that droplet bind mRNA from the cell. In practice, however, ambient (cell free) RNA may also bind barcodes from a droplet. As with index hopping, robust statistical methods are in place to adjust for ambient RNA contamination in droplet based scRNA-seq experiments, but they are not appropriate for spatial data as they do not accommodate the spatial dependence. For example, SoupX (Young and Behjati, 2020) and DecontX (Yang et al., 2020) are currently the state-of-the-art methods for decontaminating ambient RNA in scRNA-seq data. Each of these methods begins by clustering single-cell data, or taking as input clustering information; decontamination is then performed within cluster. As will be shown in section 4.4, both methods result in poor estimates of expression and increased false discoveries in downstream analyses when applied to ST data. These results should not be taken as evidence that SoupX and DecontX perform poorly in general. Rather, it should be stressed that neither method was designed for spatial data and, consequently, it should not be surprising that they

are not sufficient in this setting.

In section 4.2, we demonstrate the effect of spot swapping in multiple ST experiments. While it is straightforward to quantify the extent of spot swapping from tissue spots to background spots, assessing the extent of spot swapping within tissue is challenging in most settings without prior information. Toward this end, we consider marker genes where expression is known to be high in particular tissue regions, and low in others. We also conduct a human-mouse chimeric experiment to evaluate the extent of human-specific transcripts in mouse regions, and vice versa.

To adjust for spot swapping in ST experiments, we propose a statistical approach called SpotClean, implemented in the R package R/SpotClean. section 4.3 gives details about the model. In section 4.4, simulations and case study analyses show that SpotClean increases the specificity of marker gene expression, increases the power for identifying differentially expressed genes, improves the specificity of clusters, and increases the accuracy of spot annotations. The impact of these improvements in studies of breast, pancreatic, and colorectal cancer is also demonstrated.

4.2 Experiments

Spot swapping in public datasets

We start by analyzing publicly available datasets of ST experiments to show evidence of spot swapping. Multiple ST platforms are considered, including 10x Visium (Ståhl et al., 2016), SpatialTranscriptomics (Ståhl et al., 2016), and Slide-seqV2 (Stickels et al., 2021). Links to these data are provided in Table C.5. For each Visium and SpatialTranscriptomics dataset considered, the count matrix was normalized via scran (Lun et al., 2016), following the Seurat (Satija et al., 2015) pipeline for dimension reduction, clustering, and visualization. For each Slide-seqV2 dataset, we inspected total UMI counts of all spatial barcodes in the raw count matrix.

Figure 4.1 shows spot swapping from tissue to background in a ST study of human brain from Maynard et al. (2021). Specifically, Figure 4.1b shows that UMI counts at background spots (which are zero in the absence of contamination) are far from zero, with the counts decreasing with increasing distance from the tissue. The distributions of total UMI counts in tissue and background spots show considerable overlap (Figure 4.1c); and the expression patterns at tissue spots and nearby background spots are similar, but distinct from distant background spots, as

shown for 50 genes in Figure 4.1d. As a result of expression similarity between the tissue and nearby background, tissue and background spots often cluster together. This is emphasized in Figure 1f, where spots on the slide are colored by membership in the graph-based clusters shown in Figure 4.1e. As shown, many of the clusters contain spots from the tissue and nearby background. Figure C.2, Figure C.3, Figure C.4, and Figure C.5 show similar results from 16 additional datasets; and Table C.1 shows that the proportion of UMI counts in background spots ranges from 5% to 20% in most datasets.

The results above demonstrate that spot swapping occurs from tissue to background. While this reduces expression levels at tissue spots, thereby reducing the power of the experiment, a bigger concern is spot swapping from one tissue spot to another, as this confounds downstream analyses. Evaluating the extent of spot swapping from tissue spot to tissue spot is challenging as it requires information about expected expression of specific genes at specific tissue locations. Toward this end, we first consider tissue-specific marker genes that identify distinct tissue layers in brain (Maynard et al., 2021). In the absence of spot swapping, expression for a layer-specific marker should be high within that layer, and low (or off) in other layers. When spot swapping occurs, marker expression is relatively high in adjacent layers and decreases with increasing distance from the layer. This is evident with GFAP, for example, a marker known to be up-regulated in white matter (WM) and in the first annotated layer of the dorsolateral prefrontal cortex (Layer1) (Maynard et al., 2021). Figure C.6 shows high expression of GFAP in WM and Layer1 spots, as expected, but also relatively high expression in tissue spots adjacent to WM and Layer1, with GFAP expression decreasing as distance from WM (or Layer1) increases. While it is possible that some increase in marker expression in adjacent tissue spots may be due to the presence of WM (or Layer1) cells at those spots, we note that the rate of expression decay into the background spots (where no cells are present) is similar to the rate of decay into adjacent tissue regions. Consequently, the possible presence of WM (or Layer1) cells in adjacent tissue spots is not sufficient to fully explain the observed expression pattern. Similar results are shown for a WM marker, MOBP (Figure C.6), as well as additional markers in multiple datasets (Figure C.7).

A study of human breast cancer provides another example. Figure C.8 shows expression for a highly specific breast cancer marker, ERBB2 (also called HER2). Because of its high specificity (it is typically expressed at a low level in normal

breast tissue, but highly expressed in many breast tumors (Browne et al., 2009)), ERBB2 is used in clinical practice as a target of a number of therapies (Oh and Bang, 2020). Figure C.8 shows high expression of ERBB2 in the tumor tissue, but also high expression in nearby normal tissue that decreases with increasing distance from the tumor. As mentioned above, the increased expression in adjacent normal tissue may be due to the presence of both tumor and normal cells in those spots. However, this is not sufficient to fully explain the effect as the rate of decay from tumor tissue to adjacent normal tissue is similar to the rate of decay from tumor into the background, where no cells are present.

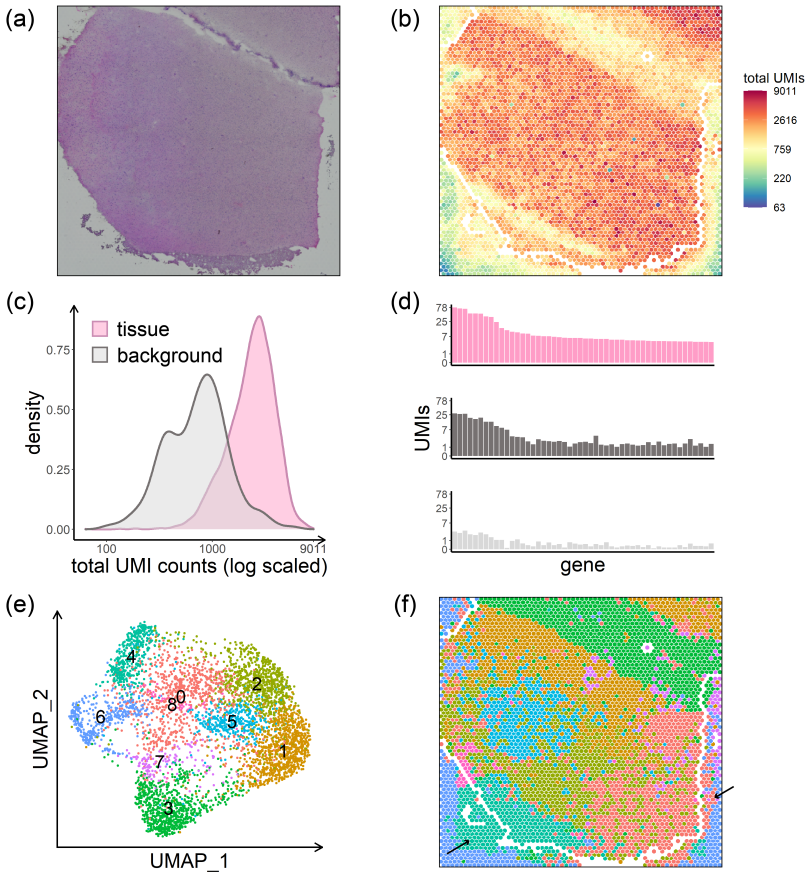

Figure 4.1: Human brain sample LIBD_151507. (a) H&E stained image. (b) UMI total counts in the background decrease with increasing distance from the tissue. Tissue and background spot annotations are taken from Maynard et al. (2021). The perimeter delineating tissue and background is shown in white.(c) UMI count densities for tissue and background spots show relatively high counts in the background. (d) Counts of the top 50 genes from a select tissue region (upper), from a nearby background region (middle), and from a distant background region (bottom) show the similarity between expression in tissue spots and nearby background spots due to spot swapping from tissue to background, an effect that decreases as distance from the tissue increases. The positions of the three regions are shown in Figure C.2. (e) Graph-based clustering of all spots identifies 9 clusters. (f) Spots on the slide are colored by their cluster membership shown in (e). Black arrows highlight areas of spot swapping of signal from tissue to background.

Experimental validation of spot swapping using chimeric samples

To more directly quantify the extent of spot swapping, we designed novel chimeric experiments where human and mouse tissues were placed contiguously during sample preparation. The experiments were carried out by Drs. Aman Prassad and Rich Halberg. The intuition behind the chimeric experiment is that spot swapping from tissue to tissue spots can be partially measured by mouse RNAs detected in human spots and human RNAs detected in mouse spots.

Tissue and cDNA library preparation

Fresh sections of normal human skin tissue were obtained with consent during routine dermatologic surgery under University of Wisconsin School of Medicine and Public Health Institutional Review Board (Approval #2010-0367). On the same day, fresh mouse tissue was harvested. All mouse husbandry and experimental procedures were performed in accordance and compliance with policies approved by the University of Wisconsin Research Animals Research and Compliance committee (Protocol #M5131). Three mixed species tissue blocks were then prepared under cold conditions as follows and frozen over a bed of dry ice and stored at -80°C in optimal tissue cutting (OCT) medium until they were ready to use:

HM-1: Duodenum from a 10-week-old C57BL/6J mouse as casing to a 4 mm punch section "cylinder" of human skin

HM-2: Colon from a 10-week-old C57BL/6J mouse as casing to a 4 mm punch section "cylinder" of human skin

HM-3: Heart from a 10-week-old C57BL/6J mouse encasing a 4 mm punch section "cylinder" of human skin

The Visium Spatial Tissue Optimization Slide & Reagent kit (10x Genomics) was used to optimize permeabilization conditions for the chimeric tissue according to manufacturer's protocol and yielded an optimal tissue permeabilization time of 12 minutes. The Visium Spatial Gene Expression Slide & Reagent kit (10x Genomics) was used to generate sequencing libraries. Sections were cut at 10 μm thickness and mounted onto Visium slide capture areas, stained with H&E, digitally imaged, and then permeabilized for library preparation. Sequencing libraries were prepared following the manufacturer's protocol. Initial quality control of the libraries was by analysis of 2x150 MiSeq data for each sample. The libraries were then sequenced on a NovaSeq 6000 (Illumina), with 29 bases from read 1 and 101 from read 2, at a

depth of 500k-600k reads per spot. The actual depth was 455652, 440024, 538709 reads per spot for sample HM-1, HM-2, HM-3, respectively.

Alignment and pre-processing in the chimeric experiment

The sequencing quality of each sample was evaluated using FastQC (Andrews et al., 2010) and MultiQC (Ewels et al., 2016). All FastQ files passed quality control. Tissues were manually aligned using the Loupe Browser. Reads were aligned to the GRCh38+mm10 reference genome (refdata-gex-GRCh38-and-mm10-2020-A at `https://support.10xgenomics.com/single-cell-gene-expression/software/downloads/latest`) and gene expression was quantified using Space Ranger under default parameters. Following alignment, we considered only those reads labeled confidently mapped by Space Ranger; confidently mapped reads are reads that map uniquely to a gene. We refer to a gene as a human gene if it has prefix GRCh38; a mouse gene has prefix mm10. UMI counts were normalized for differences in total counts across species by scaling total UMI counts in mouse to match total UMI counts in human. Genes having average expression <0.01 were removed.

Human and mouse tissue spot annotation in the chimeric experiment

For each experiment, we annotated the H&E images to identify species-specific regions. Tissue spots were labelled as human, mouse, or histopathological mixture based on visual inspection of the H&E images. A histopathological mixture spot is one with tissue contributions from both species that can be visually verified in the H&E stained image. A pure human or pure mouse spot was relabeled as a computational mixture spot if the spot label differed from the majority of UMIs. Specifically, a human (or mouse) spot was labelled as a computational mixture if the total UMI counts from mouse (human) exceeded the median of total UMI counts across all mouse spots (human spots). Background spots are defined as those spots on the slide outside the tissue region (not annotated as human, mouse, or mixture). Both histopathological or computational mixture spots were removed prior to analyses in an effort to ensure that the effects shown are not due to spots containing a mixture of the two species. Figure 4.2a shows the species annotation of sample HM-1.

Measurements of spot swapping in the chimeric experiment

Spot swapped reads include reads from one tissue spot binding background probes (tissue-to-background) as well as reads at one tissue spot binding probes at another tissue spot (tissue-to-tissue). It is not possible to directly measure tissue-to-tissue swapping in most cases. However, our chimeric experiment provides some insight into the extent of spot swapping tissue-to-tissue. Here we calculated the proportion of mouse-specific reads in human spots and human-specific reads in mouse spots (Figure 4.2, Figure C.9). This is a lower bound on the proportion of spot-swapped reads (LPSS) as it does not account for spot swapping within species (e.g. reads from human spot t bound by probes at human spot t'), or for reads in the background. LPSS ranges between 10-15% in these experiments (Table C.1).

Taken together, results from a comparison of tissue and background expression (Figure 4.1, Figure C.2, Figure C.3, Figure C.4, Figure C.5), analysis of marker genes in brain and breast cancer tissue (Figure C.6, Figure C.7, Figure C.8,), and the chimeric experiment (Figure 4.2, Figure C.9, Table C.1) demonstrate that spot swapping affects UMI counts in ST experiments. As we show in section 4.4, this nuisance variability decreases the power and precision of downstream analyses.

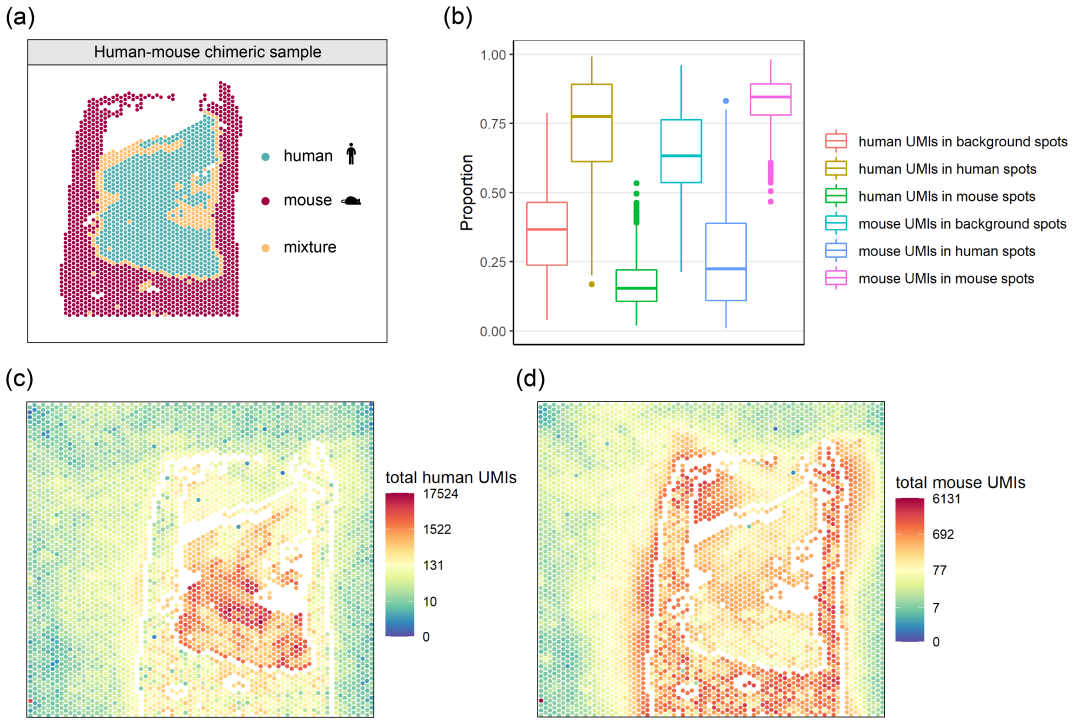

Figure 4.2: (a) Species annotation of sample HM-1, a chimeric tissue of human skin and mouse duodenum. Spots annotated as mixtures were removed prior to calculating the summaries in panels (b)-(d) in an effort to ensure that the effects shown are not due to spots containing a mixture of the two species. Panel (b) shows the spot-specific proportions of spot-swapped UMI counts (human-specific UMIs in background or mouse spots; mouse-specific UMIs in background or human spots). Also shown are the proportion of human-specific UMIs in human spots and mouse-specific UMIs in mouse spots. Note that there may be spot swapped reads in these latter proportions (e.g. reads from human spot t bound by probes at human spot t'), but they cannot be identified in this experiment. The total UMI counts in human-specific genes (panel (c)) and mouse-specific genes (panel(d)) for HM-1 are also shown. Similar plots for HM-2 and HM-3 are shown in Supplementary Figure 9. Tissue spots on the perimeter as well as spots annotated as mixtures were removed prior to calculating the proportions in panel (b) in an effort to ensure that the effects shown are not due to spots on the tissue-background boundary.

4.3 Methods

To adjust for the effects of spot swapping in ST experiments, we developed SpotClean. SpotClean is based on a probabilistic framework that accommodates spot-swapped reads to provide improved estimates of UMI counts for every gene at each spot.

Specifically, SpotClean models gene-specific expression at a given spot as a function of reads present in tissue at that spot, reads that bleed out into other spots, and reads that bleed in from other spots. Bleeding rates and the size of the neighborhood affected are estimated via gradient descent; latent expression levels are estimated using an EM algorithm.

The SpotClean model

Let K be the total number of spots, G be the set of genes, I_t be the set of tissue spots with cardinality $|I_t| = K_t$, and I_b be the set of background spots with cardinality $|I_b| = K_b$ where $K_t + K_b = K$. The true (i.e., uncontaminated) UMI counts are given by $\{Y_{g,t}\}_{g\in G, t\in I_t}$ and observed counts by $D = \{X_{g,j}\}_{g\in G, j\in I_t \bigcup I_b}$. As our interest here is to characterize the extent of spot swapping, we introduce the missing variable $B_{g,t,j}$ to be the UMI count for gene g leaving tissue spot t and binding to tissue (or background) spot j. Likewise we define $S_{g,t}$ to be the UMI count arising from gene g in tissue spot t that remain at that spot and thus are not subject to bleeding. We decompose $Y_{g,t}$ into a sum: $Y_{g,t} = S_{g,t} + B_{g,t}$, where $B_{g,t} = \sum_{k\in I_t \bigcup I_b} B_{g,t,k}$ counts all bleed-outs from spot t to other spots $k \neq t$. Extending notation, we set $Y_{g,b} = S_{g,b} = B_{g,b} = 0$ for background spots $b \in I_b$ since background spots do not express mRNA. With these missing variables defined, we note that the measured count $X_{g,j} = S_{g,j} + R_{g,j}$ where $R_{g,j} = \sum_{k\in I_t} B_{g,k,j}$ represents UMI counts received at spot j due to spot swapping. We leverage this missing-data formulation by flexibly modeling the component counts with independent Poisson distributions, which are known to be effective for UMI counts (Kim et al., 2020).

For a collection of spot and gene-specific parameters, as well as global parameters controlling the swapping rates, we parameterize the distributions as: $S_{g,t} \sim \text{Poisson}(\mu_{g,t}(1 - r_\beta))$ and $B_{g,t,j} \sim \text{Poisson}(\mu_{g,t} r_\beta [(1 - r_\gamma) w_{t,j} + r_\gamma \frac{1}{K}])$ where r_β is the bleeding rate; r_γ is a distal and $1 - r_\gamma$ is a proximal contamination rate. By taking the global bleeding rate $r_\beta \in [0, 1]$, it follows that the uncontaminated counts follow: $Y_{g,t} \sim \text{Poisson}(\mu_{g,t})$ for target parameters $\mu_{g,t}$ whose estimates constitute statistical estimates of the uncontaminated counts. Likewise for measured counts, $X_{g,j} \sim \text{Poisson}(\eta_{g,j})$, for induced gene and spot parameters. We define $w_{t,j}$ by a weighted Gaussian kernel: $w_{t,j} = K(d_{t,j}, \sigma) / \sum_{j'} K(d_{t,j'}, \sigma)$ where $d_{t,j}$ is the physical Euclidean distance between spots t and j measured in pixels in the slide image, σ is the kernel bandwidth, and $K(d, \sigma) = e^{(-d^2/2\sigma^2)}$ is a Gaussian kernel (Chung, 2020).

Parameter estimation

Plug-in estimates obtained by minimizing the residual sum of squares (RSS) between observed total counts and their expected values are used to estimate r_β,r_γ,and σ. Specifically,

$$(\widehat{r_\beta}, \widehat{r_\gamma}, \hat{\sigma}, \{\widehat{\mu_{\cdot t}}\}_{t\in I_t}) = \underset{r_\beta, r_\gamma, \sigma, \{\mu_{\cdot t}\}_{t\in I_t}}{\operatorname{argmin}} \sum_{j\in I_t \cup I_b} (X_{\cdot j} - \eta_{\cdot j})^2$$

where $X_{\cdot j}$, $\eta_{\cdot j}$, $\mu_{\cdot j}$ are the summations of $X_{g,j}$, $\eta_{g,j}$, $\mu_{g,j}$ among all genes, respectively, and

$$\eta_{\cdot j} = E(X_{\cdot j}) = \begin{cases} \sum_{t\in I_t} \mu_{\cdot t} r_\beta \left[r_\gamma \frac{1}{K} + (1 - r_\gamma) w_{t,j} \right], \text{ if } j \in I_b \\ \mu_{\cdot j} (1 - r_\beta) + \sum_{t\in I_t} \mu_{\cdot t} r_\beta \left[r_\gamma \frac{1}{K} + (1 - r_\gamma) w_{t,j} \right], \text{ if } j \in I_t \end{cases}$$

To reduce computational complexity, $\hat{\sigma}$ is taken as the minimum RSS calculated over a grid of candidate values. Explicit gradients are calculated for r_β and r_γ and estimates are obtained by L-BFGS-B gradient descent (Byrd et al., 1995). Rewriting the problem in matrix representation, let $\mu = (\mu_{\cdot 1}, \ldots, \mu_{\cdot K_t})^T$ and $X = (\{X_{\cdot j}\}_{j\in I_b}, \{X_{\cdot j}\}_{j\in I_t})^T$. Denote the proximal contamination weight matrix $W = \begin{pmatrix} W_1 \\ W_2 \end{pmatrix}$, where W_1 is a $K_b \times K_t$ matrix containing the Gaussian weights between background and tissue spots, and W_2 is a $K_t \times K_t$ matrix containing the Gaussian weights between pairs of tissue spots. The column sums of W are equal to 1 based on its definition. Let I_{K_t} be the identity matrix with dimension K_t. Let J_1 and J_2 be $K_b \times K_t$ and $K_t \times K_t$ matrices of ones representing the distal contamination weights. The residual sum of squares (RSS) becomes

$$RSS = \left\| X - \left[(1 - r_\beta) \begin{pmatrix} 0 \\ I_{K_t} \end{pmatrix} + r_\beta (1 - r_\gamma) \begin{pmatrix} W_1 \\ W_2 \end{pmatrix} + \frac{r_\beta r_\gamma}{K} \begin{pmatrix} J_1 \\ J_2 \end{pmatrix} \right] \mu \right\|_{L_2}^2$$

The gradients of unknown parameters are then calculated as

$$\begin{aligned}
\frac{\partial RSS}{\partial \mu} &= 2\left[(1-r_\beta)^2 I_{K_t} + {r_\beta}^2 (1-r_\gamma)^2 W^T W \right. \\
&\quad \left. + r_\beta (1-r_\beta)(1-r_\gamma)\left(W_2^T + W_2\right) + \frac{r_\beta r_\gamma (2 - r_\beta r_\gamma)}{K} J_2\right]\mu \\
&\quad - 2\left[(1-r_\beta)\begin{pmatrix} 0 & I_{K_t}\end{pmatrix} X + r_\beta (1-r_\gamma) W^T X + \frac{r_\beta r_\gamma}{K}\begin{pmatrix} J_1 \\ J_2\end{pmatrix}^T X\right] \\
\frac{\partial RSS}{\partial r_\beta} &= \mu^T\left[2(r_\beta - 1) I_{K_t} + 2 r_\beta (1-r_\gamma)^2 W^T W \right. \\
&\quad \left. + (1-2r_\beta)(1-r_\gamma)\left(W_2^T + W_2\right) + \frac{2r_\gamma - 2 r_\beta {r_\gamma}^2}{K} J_2\right]\mu \\
&\quad - 2X^T\left[(1-r_\gamma) W + \frac{r_\gamma}{K}\begin{pmatrix} J_1 \\ J_2\end{pmatrix} - \begin{pmatrix} 0 \\ I_{K_t}\end{pmatrix}\right]\mu \\
\frac{\partial RSS}{\partial r_\gamma} &= \mu^T\left[2{r_\beta}^2 (r_\gamma - 1) W^T W + r_\beta (r_\beta - 1)\left(W_2^T + W_2\right) + \frac{2r_\beta - 2{r_\beta}^2 r_\gamma}{K} J_2\right]\mu \\
&\quad - 2X^T\left[\frac{r_\beta}{K}\begin{pmatrix} J_1 \\ J_2\end{pmatrix} - r_\beta W\right]\mu
\end{aligned}$$

Since this optimization problem is not necessarily convex, it is important to choose appropriate initial values. For the initial values $\{\mu_{\cdot t}^{(0)}\}_{t\in I_t}$ of $\{\mu_{\cdot t}\}_{t\in I_t}$, we use the observed total UMI counts $\{X_{\cdot t}\}_{t\in I_t}$ in tissue spots and scale them up so that they sum to the total UMIs in the data. The initial bleeding rate, $r_\beta^{(0)}$, is the average expression in background spots divided by the average expression in all spots; and the initial distal contamination rate, $r_\gamma^{(0)}$, is defined by average expression in the 25th-50th percentile of all background spots divided by average expression in all background spots. With estimates $\widehat{r_\beta}$, $\widehat{r_\gamma}$, $\hat{\sigma}$ of the global parameters, true expression levels $\{\mu_{g,t}\}_{g\in G, t\in I_t}$ are readily estimated using an expectation-maximization (EM) algorithm (Dempster et al., 1977). Recall that the observed data $\mathcal{D} = \{X_{g,j}\}_{g\in G, j\in I_t\cup I_b}$ has log-likelihood

$$l_{\mathcal{D}} = \sum_{g\in G}\sum_{j\in I_t\cup I_b} l_{X_{g,j}} = \sum_{g\in G}\sum_{j\in I_t\cup I_b}\{X_{g,j}\log\eta_{g,j} - \eta_{g,j}\} + \text{constant}$$

and the complete data are $\mathcal{C} = \{S_{g,t}, B_{g,t,j}\}_{g\in G, t\in I_t, j\in I_t\cup I_b}$ with log-likelihood

$$\begin{aligned}
l_{\mathcal{C}} &= \sum_{g\in G}\sum_{t\in I_t} l_{S_{g,t}} + \sum_{g\in G}\sum_{t\in I_t}\sum_{j\in I_t\cup I_b} l_{B_{g,t,j}} \\
&= \sum_{g\in G}\sum_{t\in I_t}\left[S_{g,t}\log\left(\mu_{g,t}\left(1-r_\beta\right)\right) - \mu_{g,t}\left(1-r_\beta\right)\right] \\
&+ \sum_{g\in G}\sum_{t\in I_t}\sum_{j\in I_t\cup I_b}\left[B_{g,t,j}\log\left(\mu_{g,t}r_\beta\left[\left(1-r_\gamma\right)w_{t,j} + r_\gamma\frac{1}{K}\right]\right)\right. \\
&\left. - \mu_{g,t}r_\beta\left[\left(1-r_\gamma\right)w_{t,j} + r_\gamma\frac{1}{K}\right]\right] + \text{constant} \\
&= \sum_{g\in G}\sum_{t\in I_t}\left(S_{g,t}\log\left(\mu_{g,t}\left(1-r_\beta\right)\right)\right. \\
&\left. + \sum_{j\in I_t\cup I_b}\left[B_{g,t,j}\log\left(\mu_{g,t}r_\beta\left[\left(1-r_\gamma\right)w_{t,j} + r_\gamma\frac{1}{K}\right]\right)\right] - \mu_{g,t}\right) + \text{constant}
\end{aligned}$$

Let $\{\mu_{g,t}^{(n)}\}_{g\in G, t\in I_t}$ be the parameter values at the n-th iteration. The E-step involves computation of the expectation of latent variables conditioning on observed data and parameter values at the current iteration. Given the fact that if $U \sim \text{Poisson}(a)$, $V \sim \text{Poisson}(b)$, and U and V are independent, then $U|(U+V) \sim \text{Binomial}(U+V, \frac{a}{a+b})$, and we have

$$\begin{aligned}
S_{g,t}^{(n)} &\coloneqq E\left[S_{g,t}|\mathcal{D}\right] = E\left[S_{g,t}|X_{g,t}\right] = X_{g,t}\frac{\mu_{g,t}^{(n)}\left(1-r_\beta\right)}{\eta_{g,t}^{(n)}} \\
B_{g,t,j}^{(n)} &\coloneqq E\left[B_{g,t,j}|\mathcal{D}\right] = E\left[B_{g,t,j}|X_{g,j}\right] = X_{g,j}\frac{\mu_{g,t}^{(n)}r_\beta\left[\left(1-r_\gamma\right)w_{t,j} + r_\gamma\frac{1}{K}\right]}{\eta_{g,j}^{(n)}}
\end{aligned}$$

The M-step involves maximizing the complete log-likelihood after plugging in

the conditional expectations in the E-step:

$$
\begin{aligned}
l_{\mathcal{C}}^{(n)} &= \sum_{g \in G} \sum_{t \in I_t} \Big(S_{g,t}^{(n)} \log \left(\mu_{g,t} \left(1 - r_\beta \right) \right) \\
&\quad + \sum_{j \in I_t \cup I_b} B_{g,t,j}^{(n)} \log \left(\mu_{g,t} r_\beta \left[\left(1 - r_\gamma \right) w_{t,j} + r_\gamma \frac{1}{K} \right] \right) - \mu_{g,t} \Big) \\
\frac{\partial l_{\mathcal{C}}^{(n)}}{\partial \mu_{g,t}} &= \frac{S_{g,t}^{(n)} + \sum_{j \in I_t \cup I_b} B_{g,t,j}^{(n)}}{\mu_{g,t}} - 1
\end{aligned}
$$

The M-step becomes

$$
\mu_{g,t}^{(n+1)} = S_{g,t}^{(n)} + \sum_{j \in I_t \cup I_b} B_{g,t,j}^{(n)}
$$

For the initial values of true expressions $\{\mu_{g,t}^{(0)}\}_{g \in G, t \in I_t}$, we use the observed UMI counts $\{X_{g,t}\}_{g \in G, t \in I_t}$ and scale up each gene so that their summations are equal to the gene summations in all spots.

Estimation of spot-level contamination in observed data

For tissue spot t, let c_t be the proportion of contaminated UMIs from total observed UMIs. We estimate c_t using the estimated contamination received in t over its estimated contaminated total counts from model fitting:

$$
\widehat{c_t} = \frac{\hat{E} \left(\sum_{t' \in I_t - \{t\}} \sum_g B_{g,t',t} \right)}{\hat{E} \left(X_{\cdot t} \right)}
$$

where $\hat{E}(\cdot)$ is the plug-in estimation of the expectation of the random variables in the SpotClean model.

Minimum number of background spots required for parameter estimation

Given that the observed data is a single matrix with a fixed number of columns (spots), the number of unknown parameters is proportional to the number of tissue spots. In the extreme case where all spots are covered by tissue, we have more unknown parameters than observed data values. In this case the contaminated

expressions are confounded with true expressions, and SpotClean estimation becomes unreliable. We recommend that the input data have at least 25% of spots not occupied by tissue, so that SpotClean has enough information from background spots to reliably estimate contamination.

4.4 Results

In this section, we will show that SpotClean is able to recover true gene expression, provide more precise estimates of marker gene expression, and improve downstream analyses.

We start by evaluating the performance of SpotClean on adjusting for the spot swapping contamination and recovering true gene expression in simulated data. Results are compared with SoupX (Young and Behjati, 2020) and DecontX (Yang et al., 2020), the two existing decontamination methods for single-cell RNA-seq data. Next, we illustrate the benefits of SpotClean on marker gene estimation and on downstream DE and clustering analyses using real world data. More importantly, we show that SpotClean helps delineate tumor from normal tissue and reduce the risk of overestimating malignancy in cancer studies. In the end, we show some computational explorations of potential factors affecting the extent of spot swapping.

Simulation set-up

SimI simulates the spot swapping effect to get contaminated UMI counts given an input dataset. Specifically, starting from an input UMI count matrix of real data, 3000 genes with highest total UMI counts were selected. Expression for these genes was scaled to target the same average UMI total counts (average taken over spots) across input datasets. Denote the resulting matrix by $\{\mu_{g,t}\}_{t\in I_t}$. The bleeding rate r_β and distal contamination rate r_γ were estimated from the input data, using the same approach as described for obtaining initial values in SpotClean. The spot distances $\{d_{t,j}\}_{t\in I_t,\, j\in I_t\cup I_b}$ were calculated based on the spot coordinates in the H&E image of the input dataset; the contamination radius, σ, was set to 10; and the weights which describe the proportion of UMIs swapping locally from tissue spot t to any spot j, $w_{t,j}$, is given by a Gaussian kernel. The expected contamination of gene g from tissue spot t to spot j is then given by $\mu_{g,t} r_\beta \left[(1-r_\gamma)\, w_{t,j} + r_\gamma \frac{1}{K}\right]$. Summing contamination from all tissue spots to spot j and adding the UMIs that stay at j,

$\mu_{g,j}(1-r_\beta)$, gives the expected observed expression $\eta_{g,j}$. Simulated counts for gene g in spot j are sampled from Poisson $(\eta_{g,j})$.

Additional simulations are similar, but proximal contamination weights are not given by a Gaussian kernel. Rather, SimII, SimIII, and SimIV assume proximal contamination weights are given by a Linear, Laplace, and Cauchy kernel, respectively.

SimV is designed to evaluate the effect of decontamination on discovering spatially variable (SV) genes. Starting from a UMI count matrix of real data, we select the top 5000 most highly expressed genes; any gene having average expression less than 0.1 is removed. Next, we define true SV genes using SpatialDE (Svensson et al., 2018), which is a computational method to test for SV patterns of gene expression. SpatialDE is applied using default settings; the top 500 highest expressed genes with q-value <=0.01 are identified as true SV genes. For each SV gene, we simulate a matched non-SV gene by sampling independent Poisson counts parameterized by the average expression of the SV gene. As a result, we have 500 SV genes and 500 non-SV genes.

For each simulated data in each simulation setting, SpotClean, SoupX and DecontX were applied to get the decontaminated gene expression. Default parameters were used for SpotClean and DecontX. Since SoupX requires manual input of clusters, we first applied the state-of-the-art Seurat pipeline (Satija et al., 2015) on the raw tissue UMI count matrix to get cluster labels, with functions NormalizeData(), FindVariableFeatures(), ScaleData(), RunPCA(), FindNeighbors(), FindClusters() applied under default settings. Parameters for SoupX (soupRange in estimateSoup(), tfidfMin and soupQuantile in autoEstCont()) were manually tuned when the default settings failed. Some datasets did not run even after parameter tuning; results from these datasets are marked as NA.

Simulation results

Table 4.1 shows the mean squared error (MSE) between true and decontaminated gene expression in SimI simulated datasets; SpotClean provides better estimates of expression, reducing the MSE by over 20% in most simulations. SoupX and DecontX increase the MSE as they are not designed for ST data, and perform poorly on decontaminating spot swapping. Table C.2, Table C.3, and Table C.4 show similar results. In terms of downstream analysis of identifying SV genes, SpotClean reduces

Dataset	No decontamination	SpotClean	SoupX	DecontX
LIBD_151507	31.334	**15.041**	NA	83.016
LIBD_151508	26.477	**12.987**	NA	53.596
LIBD_151669	21.931	**11.745**	NA	267.124
LIBD_151670	17.903	**10.304**	NA	70.851
LIBD_151673	22.121	**11.682**	NA	56.747
LIBD_151674	25.861	**13.361**	108.916	57.304
mouse_brain	24.979	**9.896**	779.811	278.374
mouse_kidney	12.114	**7.810**	291.890	119.825
human_breast	14.278	**9.605**	139.924	72.129
human_lymphnode	113.216	**30.261**	486.128	189.395
human_spinalcord	126.197	**13.191**	163.898	187.928

Table 4.1: Average mean squared error (MSE) between true and decontaminated gene expression (average taken over 3000 genes) in 11 SimI datasets simulated using input from the dataset indicated. NA denotes datasets for which the corresponding method failed to run. The lowest MSE for each dataset is bolded.

false discovery rate (FDR), while SoupX and DecontX have increased FDR for many datasets as they impose variability on non-SV genes during decontamination (Figure 4.3).

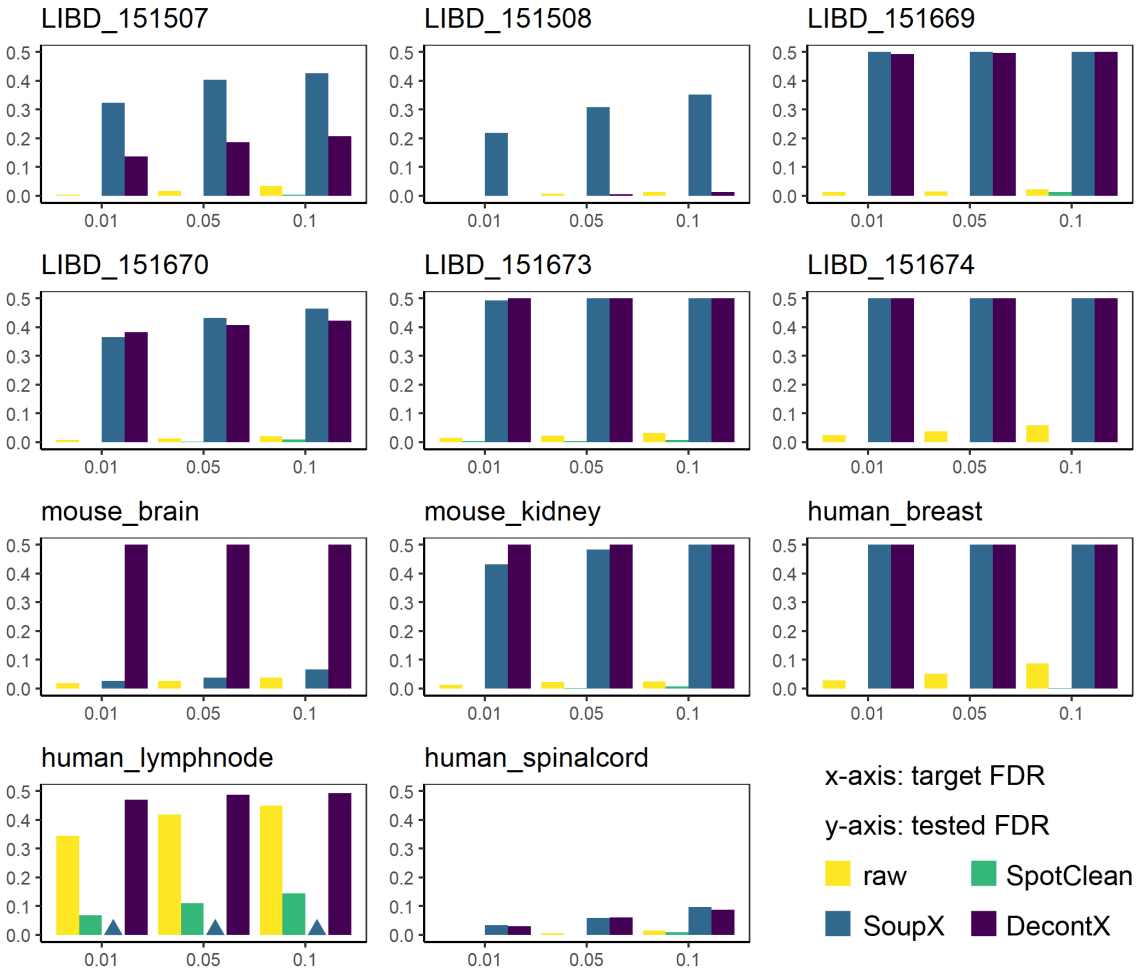

Figure 4.3: SimV simulated data containing spatially varying (SV) genes was simulated for 11 datasets using input from the publicly available dataset indicated. SpatialDE was applied to the simulated data following decontamination by SpotClean, SoupX, and DecontX. Shown is the observed false discovery rate (FDR) for the raw data and each dataset following decontamination for three target FDR cut-offs. SoupX failed to run on the human_lymphnode data (NA shown as triangles). SoupX and DecontX have increased FDR for many datasets as they impose variability on non-SV genes during decontamination.

Case study

SpotClean recovers true gene expression, provides more precise estimates of marker gene expression, and improves downstream analyses

We first show that SpotClean is able to provide more precise estimates of marker gene expression and improve downstream differential expression (DE) analysis

in the human brain data. Specifically, Figure 4.4a shows that SpotClean improves the specificity of GFAP by maintaining expression levels in WM and Layer1 and reducing spurious expression in the other layers. Figure C.7 shows similar results for four additional markers. In contrast, SoupX and DecontX perform poorly in the spatial setting. Figure C.10a-b shows results from SoupX and DecontX on the brain data, where the marker genes either show no change of expression or are reduced substantially. In addition, since both methods decontaminate all genes within a cluster simultaneously, artificial patterns are imposed upon genes showing no spatial changes (Figure C.10c).

We also identified DE genes between WM and Layer6 using raw and SpotClean decontaminated data. Specifically, we filtered the list of known DE genes from Maynard et al. (2021) and considered those genes having FDR<=10^{-4}. From those, we chose the top 100 highest expressors in the raw data, sorted by fold change, and selected the top 10 for each dataset. For the DE analysis, raw and decontaminated tissue matrices were normalized using scran (Lun et al., 2016); for each gene, p-values were obtained from a two-sample two-sided t-test between the 354 spots in WM and the 486 spots in Layer6. Summary statistics for the tests in Figure 4.2b are reported in Table C.6 and Table C.7. Figure 4.4b and Figure C.11 show results for these gold-standard DE genes. In most cases, data processed via SpotClean results in increased fold-changes and smaller p-values, further suggesting that SpotClean results in more accurate expression estimates.

Additional results are demonstrated in a study of breast cancer. Figure 4.5 shows expression for ERBB2 and MUC1, another breast cancer marker, before and after SpotClean. SpotClean increases specificity of these markers by maintaining expression in the tumor regions and reducing expression in the non-tumor regions. It also leads to improved separation of the tumor and non-tumor regions. Specifically, the Seurat pipeline was applied under default settings to the raw and decontaminated data to produce UMAP plots in Figure 4.5d. Tumor spots were clustered using k-means clustering (k=2) of the top 50 PCs calculated in the Seurat pipeline. In the H&E image, tissue spots were labelled as tumor and non-tumor based on visual inspection. The adjusted rand indexes (ARI) were calculated between cluster labels and tumor/non-tumor labels. SpotClean shows improved clustering of tumor and non-tumor spots both visually and quantitatively in ARI scores. Similar results are shown in Figure 4.6 in a study of pancreatic cancer (Moncada et al., 2020).

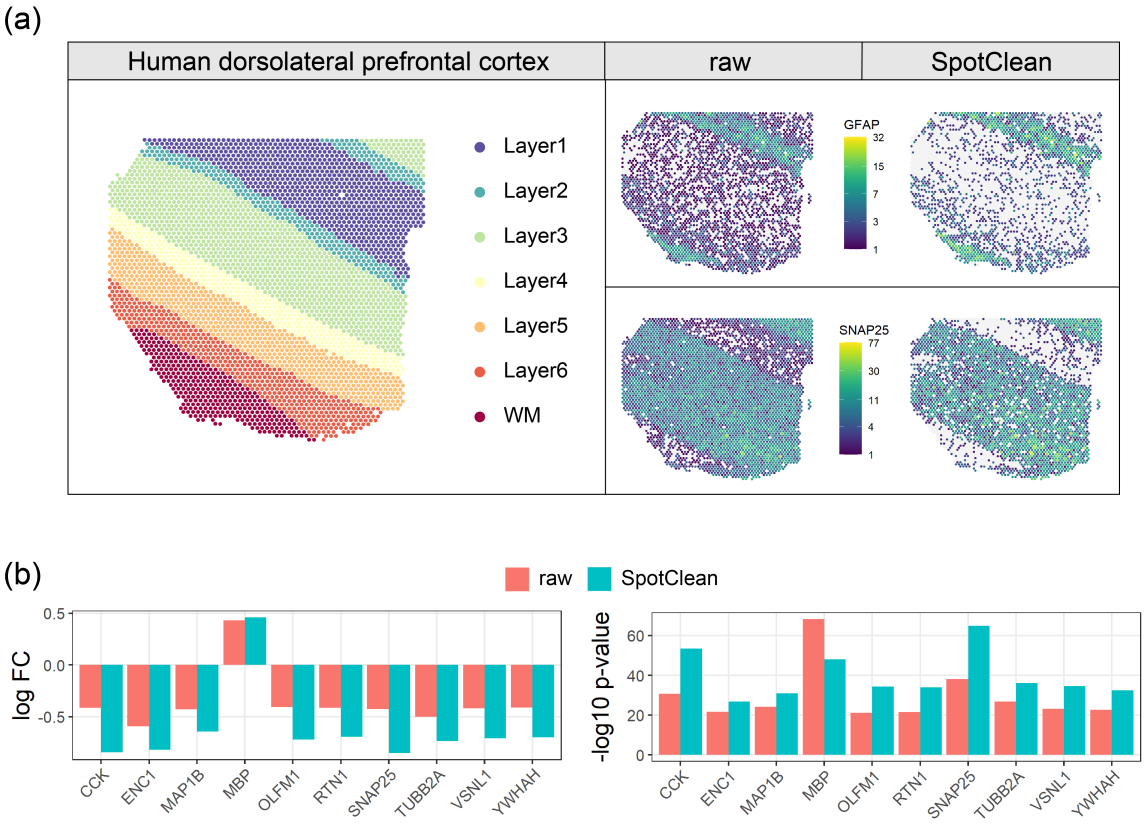

Figure 4.4: (a) Known annotation of different layers of the human brain sample LIBD_151507 (left); layer-specific marker gene expression in the raw (middle) and SpotClean decontaminated (right) data show that SpotClean provides improved specificity of marker gene expression for GFAP, a marker for WM and Layer1, and for SNAP25, a neuronal marker up-regulated in Layer2-Layer6. (b) An analysis of genes known to be differentially expressed (DE) between WM and Layer6 in raw and SpotClean decontaminated data shows that SpotClean results in increased fold-changes and smaller p-values for the majority of known DE genes.

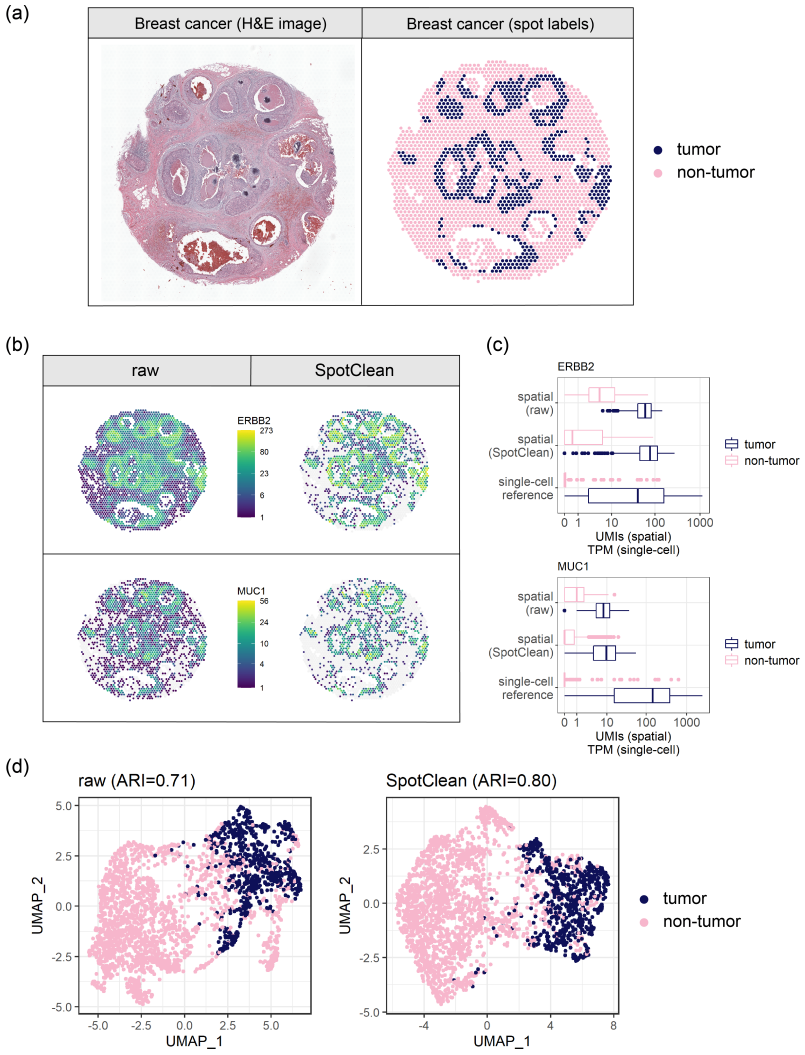

Figure 4.5: Data from a study of human breast cancer, sample human_breast_2. Panel (a) shows the H&E image (left) and spots annotated as tumor vs. non-tumor via a pathologist's visual inspection (right). Panel (b) shows expression of two tumor-specific markers in the raw (left) and SpotClean decontaminated (right) data. SpotClean increases specificity of these markers by maintaining expression in the tumor regions, and reducing expression in the non-tumor regions. Boxplots of the expression shown in panel (b) are shown in panel (c) and compared with expression in a breast cancer single-cell RNA-seq reference dataset (Chung et al., 2017). Panel (d) shows UMAP plots generated from raw and SpotClean decontaminated data colored by spot annotations. SpotClean decontaminated data leads to improved separation of the groups, as shown visually, and quantified by the ARI scores which show a 13% improvement in the SpotClean decontaminated data.

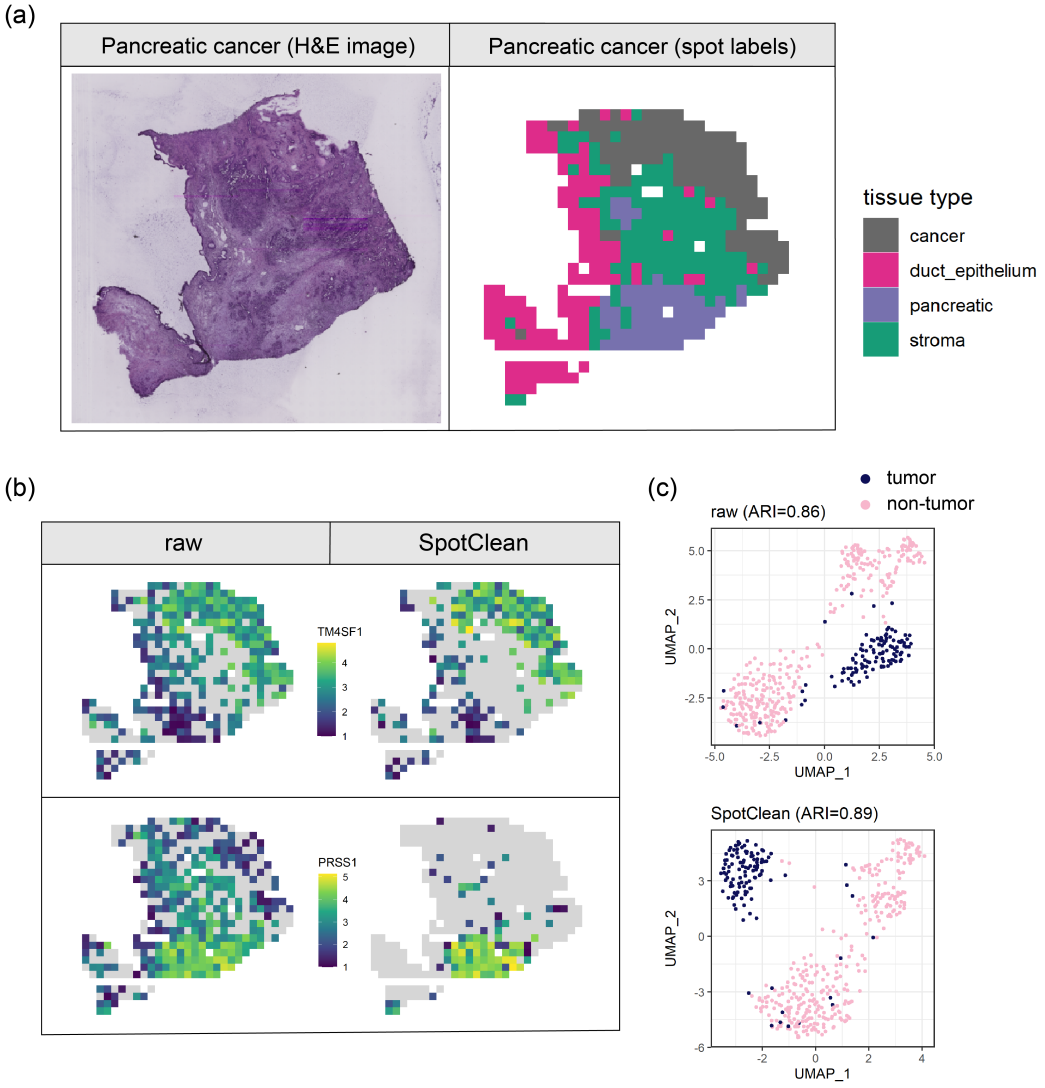

Figure 4.6: Data from a study of human pancreatic cancer (Moncada et al., 2020), sample PDAC-A. Panel (a) shows the H&E image (left) and spots annotated as tumor, duct epithelia, pancreatic tissue, and stroma from the original study (right). The upper panel (b) shows expression of the tumor-specific marker TMSF1 in the raw (left) and SpotClean decontaminated (right) data. SpotClean increases specificity of this marker by maintaining expression in the tumor region, and reducing expression in the non-tumor regions. A pancreatic-specific marker, PRSS1, is shown in the lower panel (b); as in the upper panel, the specificity of this marker is increased via SpotClean. Panel (c) shows UMAP plots generated from raw and SpotClean decontaminated data colored by spot annotations (tumor vs. non-tumor). The groups are well separated even in the raw data, but SpotClean decontaminated data leads to slightly improved separation of the groups.

SpotClean reduces the risk of overestimating malignancy and improves identification of tumor subtypes in cancer studies

As the diagnosis and extent and invasiveness of a tumor is typically estimated through evaluation of an H&E image by a pathologist, there is now considerable interest in using ST experiments, which couple the H&E image with molecular profiling data, to improve diagnosis and precision therapy. ST can provide additional information by identifying subtle collections of malignant cells, but accurate spot annotation is required for this information to be useful in clinical practice, and especially so as not to overcall tumor burden. SpotClean demonstrates advantage toward this end.

Since tumor cell populations are heterogeneous, and spots contain multiple cells, most spots containing malignant cells will also contain non-malignant cells. To estimate the cell type composition and score the malignancy level of each spot, we applied SPOTlight (Elosua-Bayes et al., 2021) to the Visium human breast cancer data and the Visium human colorectal cancer data. SPOTlight requires single-cell RNA-seq data to use as a reference; for this, we used the human breast cancer single-cell RNA-seq data from Chung et al. (2017) and the human colorectal cancer single-cell RNA-seq data from Li et al. (2017) to decompose the breast cancer and colorectal cancer ST data in this section. SPOTlight was applied to the raw data under default settings to estimate the cell type composition of every spot; SPOTlight was also applied to the SpotClean decontaminated data under default settings. As mentioned earlier, each spot contains a mixture of malignant and non-malignant cells. Towards this end, a spot's malignancy score is defined to be the proportion of tumor cells estimated by SPOTlight.

Following clinical practice, we label a spot as malignant if there is any evidence of malignancy. Specifically, we annotate spots as malignant if the estimated malignant cell composition exceeds 10%, which corresponds to approximately 1 malignant cell in the spot since the estimated number of cells in a spot is approximately 10 in Visium data (Elosua-Bayes et al., 2021). We further define non-malignant spots as "strongly non-malignant" if the non-malignant cell composition exceeds 95%, and "strongly malignant" if the malignant cell composition exceeds 30% in both raw and decontaminated data. "Questionably malignant" is used to refer to spots called malignant in the raw data, but not the SpotClean decontaminated data.

Figure 4.7a shows spots annotated using SpotClean data versus spots annotated

using data that has not been decontaminated via SpotClean for the breast cancer sample discussed above. Compared with the H&E image annotations shown in Figure 4.7a, which we consider to be a gold standard, the non-decontaminated data misidentifies many spots as malignant including those containing benign cells surrounding the tumor; the SpotClean decontaminated data more closely resembles identification of malignant cells on the H&E image. Specifically, over 13% of the spots are labelled malignant in the raw, but not SpotClean decontaminated, data. Figure 4.7b-c show that expression in these "questionably malignant" spots is more similar to spots known to harbor non-malignant cells suggesting that these questionably malignant spots are false calls.

Similar results are shown in Figure 4.8 in a study of colorectal cancer where SpotClean decontaminated data leads to improved delineation of tumor and non-tumor regions as evidenced by enhanced tumor malignancy scores in tumor spots, and lower malignancy scores in non-tumor spots, compared with raw data (Figure 4.8b). In order to investigate the effect of SpotClean on clustering tumor subtypes, we applied BayesSpace (Zhao et al., 2021), the state-of-the-art clustering algorithm for ST data, to tumor spots under default settings to obtain tumor clusters in raw and SpotClean decontaminated data. SpotClean identifies a novel cluster (SpotClean tumor cluster 1 shown in Figure 4.8c); and multiple analyses suggest that this cluster is a distinct tumor subtype containing both tumor and tumor-infiltrating immune cells. First, a careful look at the H&E image shows that this group of spots is non-normal, but distinct from other tumor regions (red boxes, Figure 4.8a). Second, 9 of the top 10 genes identified as DE between SpotClean tumor cluster 1 and other tumor clusters are immunoglobulin marker genes (from 74 total DE genes with adjusted p-value <=0.01); and immunoglobulin expression for these 9 genes is largely specific to this cluster (Figure 4.8d). Finally, the average malignancy score for this group is lower than other tumor clusters, but higher than normal spots, further suggesting that this group of spots contains both tumor cells and tumor-infiltrating immune cells (average malignancy scores at normal, tumor cluster 1, and other tumor spots are 0.384, 0.430, and 0.477, respectively). Taken together, this evidence suggests that the novel cluster identified by SpotClean maintains biologically relevant information and, in this case, provides for a more specific clustering that captures subtle structure present in the tissue.

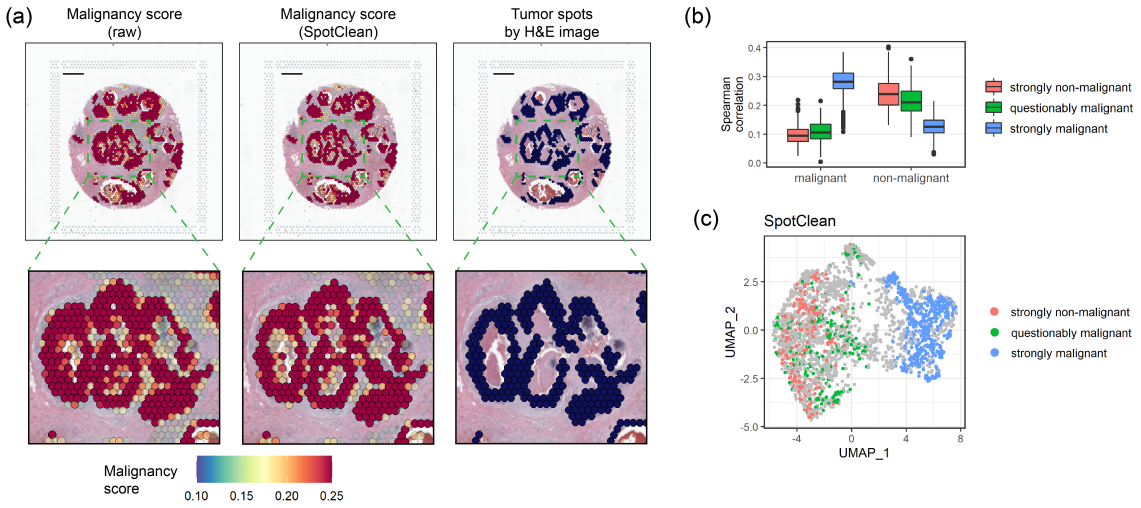

Figure 4.7: Data from a study of human breast cancer, sample human_breast_2. (a) Malignant spot composition as estimated via SPOTlight (Elosua-Bayes et al., 2021) is shown for the raw data (upper left) and SpotClean decontaminated data (upper middle). The raw data identifies many spots as malignant whereas the SpotClean decontaminated data more closely resembles the annotations derived from the H&E image (upper right). The insets highlighted in the upper panel are shown in the lower panel. (b) Spearman correlations between average expression in the malignant scRNA-seq cells and spot-specific expression were calculated. Boxplots of correlations are shown for 265 strongly non-malignant spots, 216 questionably malignant spots (spots labelled malignant in the raw data, but not the SpotClean decontaminated data), and 546 strongly malignant spots. Correlations with non-malignant scRNA-seq cells are also shown. The correlations show that expression in the questionably malignant spots more closely resembles that in non-malignant cells suggesting that the malignant classification in the raw data at these spots is likely false due to spot swapping. (c) The UMAP plot further demonstrates that the questionably malignant spots in the raw data are likely false positives as their expression more closely resembles that at non-malignant spots.

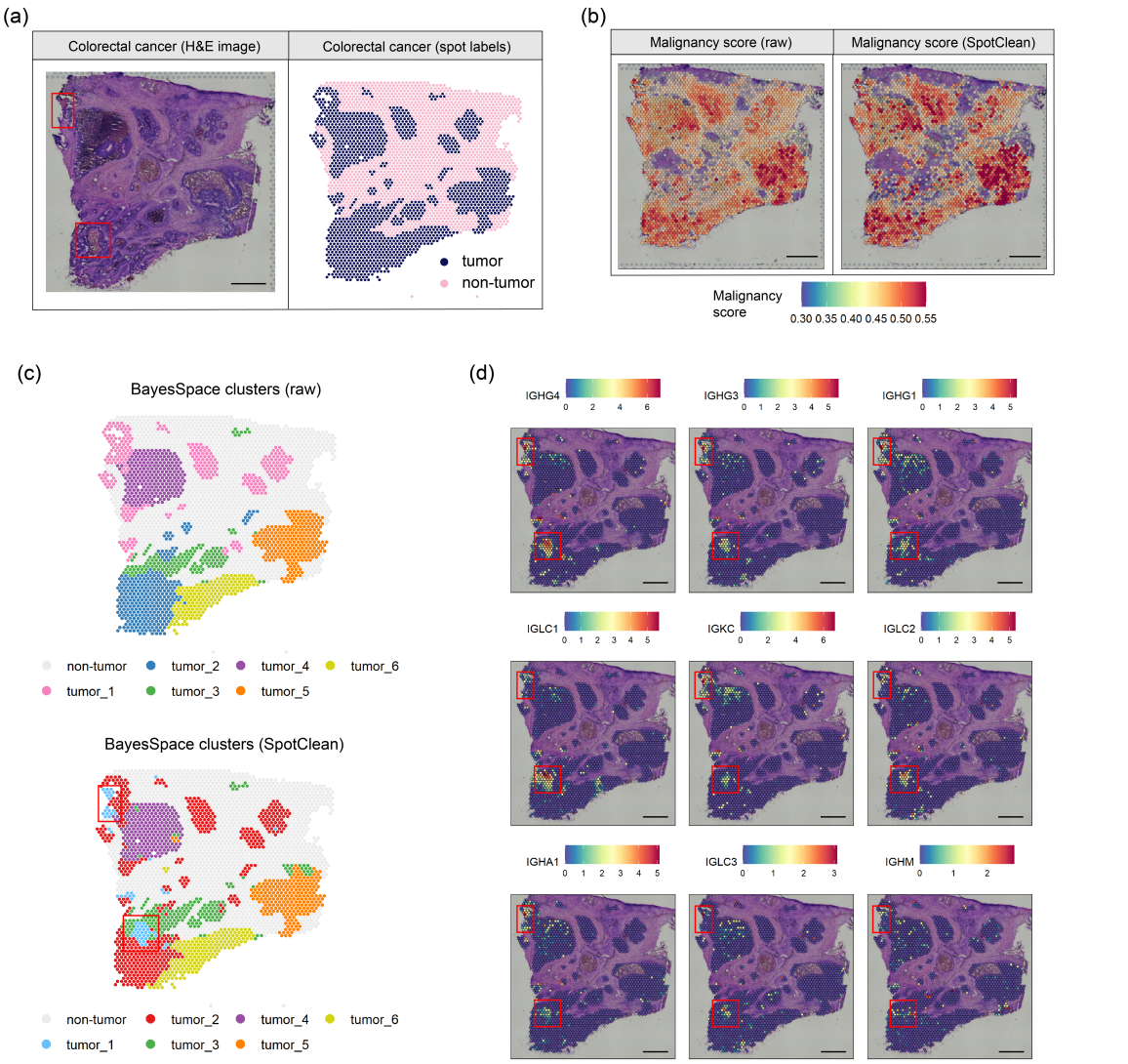

Figure 4.8: Data from a study of human colorectal cancer, sample human_colorectal. Panel (a) shows the H&E image (left) and spots annotated as tumor vs. non-tumor via a pathologist's visual inspection (right). Red boxes highlight the spots belonging to SpotClean's tumor cluster 1 (panel (c)). (b) Malignant spot composition as estimated via SPOTlight (Elosua-Bayes et al., 2021) is shown for the raw (left) and SpotClean decontaminated data (right). SpotClean results in higher malignancy scores in tumor regions, and lower in normal regions. (c) BayesSpace (Zhao et al., 2021) clustering for the raw data (top) and SpotClean decontaminated data (bottom). The SpotClean decontaminated data identifies a novel cluster (SpotClean tumor_1, red boxes). The SpotClean tumor_1 spots are distinct on the H&E image (red boxes in panel (a)) and likely contain tumor-infiltrating immune cells as evidenced by high expression in the immunoglobulin markers shown in panel (d).

Potential factors affecting the extent of spot swapping

In the SpotClean model, the bleeding rate directly measures the extent of spot swapping as it is the probability that an RNA transcript bleeds out from its original spot. SpotClean assumes a constant bleeding rate across all genes for a ST dataset. However, given that genes with different biotypes (protein-coding genes, long non-coding RNAs, mitochondrial genes, etc.) may have different binding affinities to poly-A capture regions, we would like to know if the extent of spot swapping differs among different gene biotypes. Here we approximate the gene-specific bleeding rate using the proportion of background UMI counts for each gene. Figure 4.9 shows boxplots of the proportion of background UMI counts for genes having at least 10 counts in the background (we did not consider genes with fewer than 10 counts since the variability of background proportion is too high, and the estimation is less reliable) in the 12 publicly available datasets and the chimeric experiments that we used in section 4.2. Each panel has two boxplots; the left (salmon) corresponds to the mitochondrial and long noncoding RNAs (mt_lnc); the right (turquoise) corresponds to the remaining genes. Results show that there is no consistent trend, indicating that there is no computational evidence that the extent of spot swapping varies in the different gene biotypes considered here.

We also investigated the relationship between overall bleeding rate and permeabilization time, another important factor in ST experiments. Permeabilization refers to the process that removes more cellular membrane lipids to allow mRNAs to get outside the cell. Permeabilization time plays a crucial role for the success of a ST experiment. Over-permeabilization can lead to higher diffusion rates between spots. To address this question, ideally we would like a series of ST experiments where only permeabilization times vary. Unfortunately, no such experiments exist. To get some insight computationally using existing data, we compared results from samples with different permeabilization times in different studies. We collected additional datasets with permeabilization times varying from 6 to 30 minutes. In this case, the bleeding rates cannot be approximated using the proportion of background UMIs since the number of background spots varies across datasets. We use the estimated bleeding rate from SpotClean instead. Figure 4.10 shows that there are no clear patterns between estimated bleeding rates and permeabilization times. Datasets coming from the same study tend to have similar bleeding rates, but we do not observe a clear trend of bleeding rates as a function of permeabilization

time. Note that the estimated effect of permeabilization is confounded by different species, tissue types, sample preparations, etc. As more data become available, it will be interesting to continue to assess whether or not there is a relationship between bleeding rate and permeabilization time.

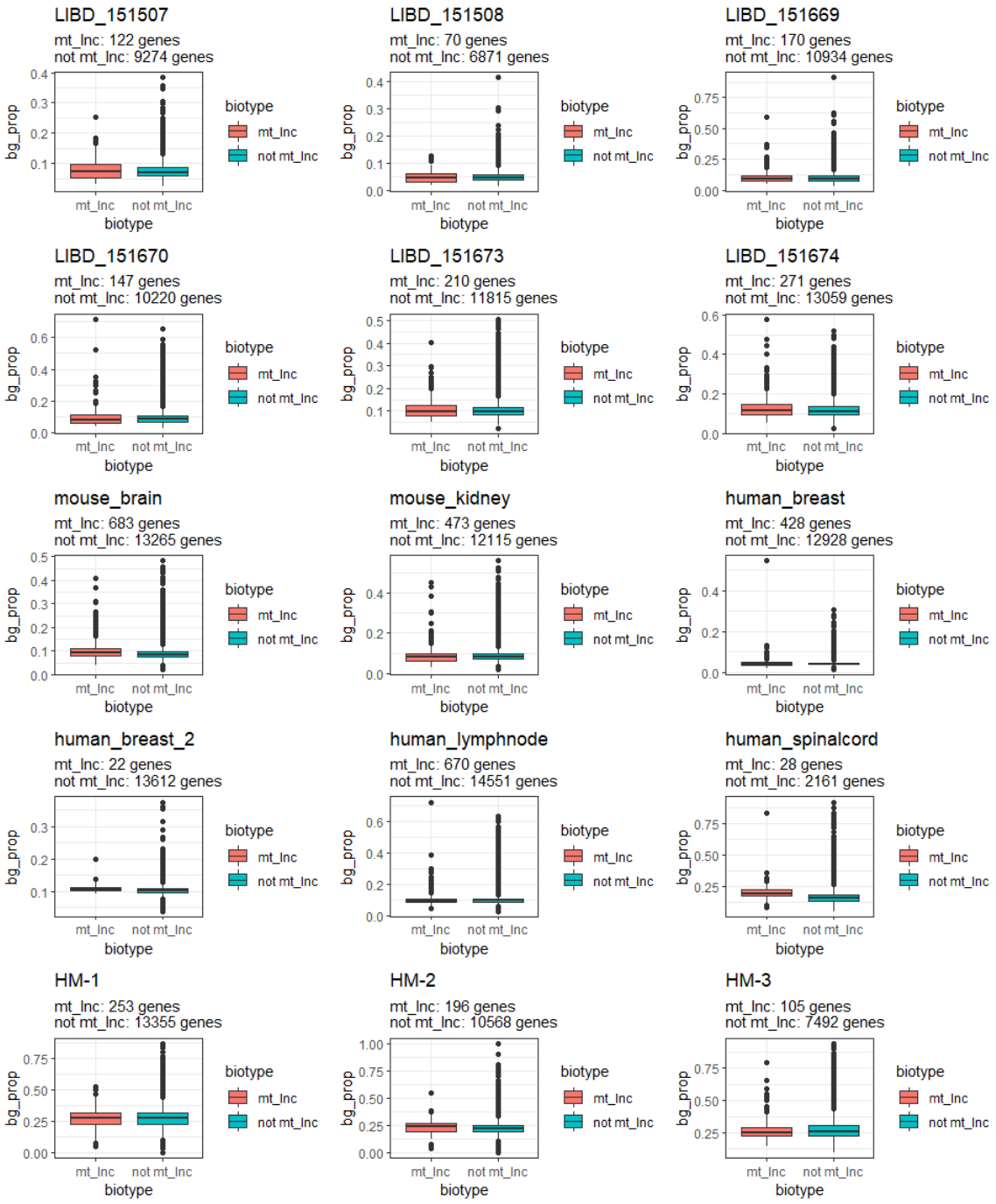

Figure 4.9: Boxplots of background UMI proportions for mitochondrial and long noncoding RNAs (mt_lnc) and the remaining genes (not_mt_lnc) in 12 publicly available datasets as well as the chimeric experiments.

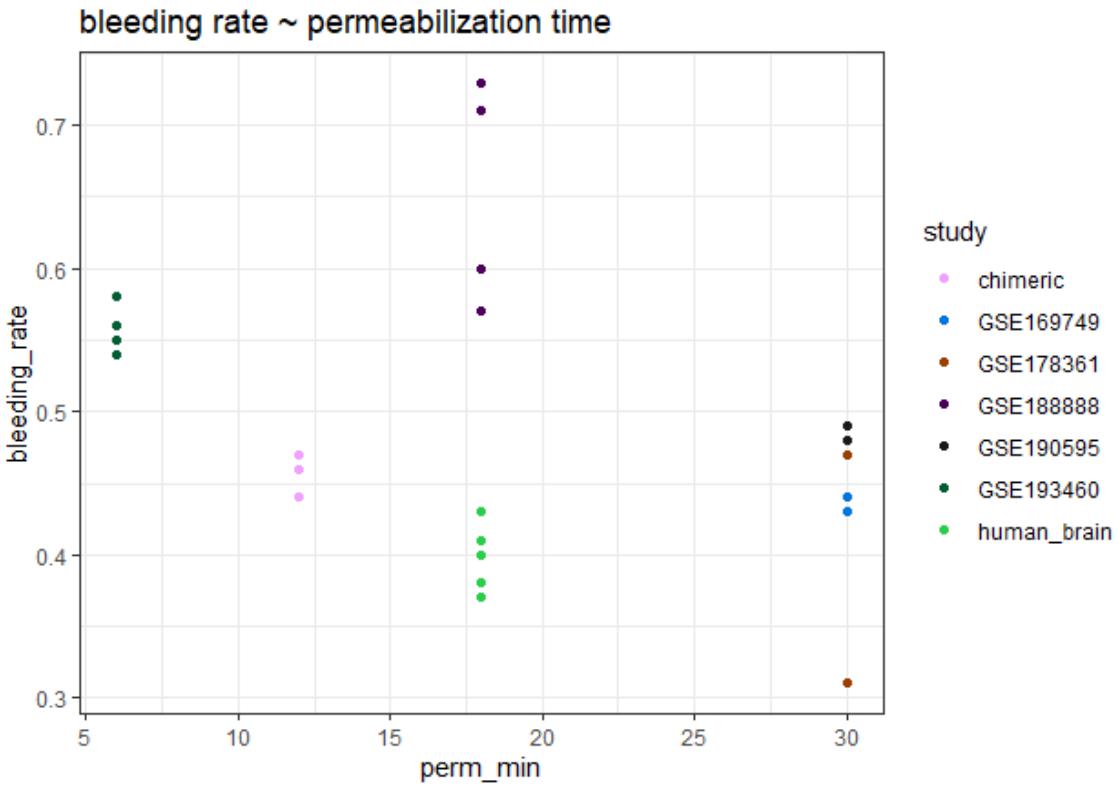

Figure 4.10: Scatter plot showing the relationship between estimated bleeding rates and permeabilization times for the human brain data, chimeric data, and additional ST datasets (figure legend shows GEO accession numbers of these data). There are no clear patterns between estimated bleeding rates and permeabilization times.

4.5 Implementation

SpotClean is implemented as an R package *SpotClean*, which is publicly available at GitHub. The source code and vignette can be found at `https://github.com/zijianni/SpotClean`.

The required input data of *SpotClean* is a gene-by-barcode matrix, as well as barcode-level information including barcode locations and whether a barcode is tissue or background. Additional data, such as the H&E image file, are not required by SpotClean, but can be useful in downstream analysis.

Given that most ST experiments are conducted using the 10x Visium platform, *SpotClean* provides functions to directly read the raw files from Space Ranger (10x Visium's computational pipeline) and get the gene-by-spot count matrix and spot-level information in R (barcode in Visium corresponds to spot). For other platforms, users need to manually create the gene-by-barcode matrix and barcode-level information.

The first step of running *SpotClean* is to create a "slide object", which is a bundled object containing the raw gene-by-spot count matrix, spot-level information, and other related data from Space Ranger. This helps put everything in one place for easy access and management. Next, the main function in the package is to perform decontamination. It takes the slide object of raw data as input together with some parameters for controlling optimization and convergence, and returns a slide object with decontaminated gene expressions and other model-related parameters and statistics appended to the spot-level information. To evaluate and visualize the performance of decontamination, *SpotClean* provides functions to plot the spots in the 2D space and color them by either expression levels of certain gene, or other spot-level numerical or categorical values. In addition, *SpotClean* provides functions to evaluate contamination levels of the data. For example, our model is able to estimate the proportion of contaminated expression at each tissue spot (i.e. expression at a tissue spot that orginated from a different spot due to spot swapping). Another contamination metric provided is the ambient RNA contamination (ARC) score. Intuitively, the ARC score is a conserved lower bound of the proportion of contamination in observed tissue spots. In the end, *SpotClean* provides functions to convert the slide object to other commonly used objects, such as the Seurat spatial object, for a smooth transfer to other downstream analysis pipelines. More details can be found in the package vignette.

The computational speed depends mainly on the number of tissue spots, as the number of unknown parameters is proportional to the number of tissue spots. SpotClean does not require parallel computation, and thus does not use up too many CPU or memory resources. As a reference, SpotClean running on a medium-size dataset (around 30,000 genes and 2,000 tissue spots) under default gene filtering takes less than 15 minutes.

4.6 Discussion and future work

Common sources of contamination in next-generation sequencing experiments such as barcode swapping (Griffiths et al., 2018; Larsson et al., 2018; Costello et al., 2018) and ambient RNA contamination (Young and Behjati, 2020; Yang et al., 2020) have been widely recognized over the past decade. We here identify spot swapping, a related but distinct form of contamination present in the 10x Visium (Ståhl et al., 2016), SpatialTranscriptomics (Ståhl et al., 2016), and Slide-seqV2 (Stickels et al., 2021) platforms. SpotClean adjusts for the effects of spot swapping using a probabalistic model that accommodates spot-swapped reads to provide improved estimates of gene-specific UMI counts at each spot. SpotClean may be used to obtain improved estimates of expression given data from the 10x Visium or SpatialTranscriptomics platforms; it is not applicable to platforms where background barcodes and/or accurate barcode positions are not provided (e.g. Slide-seqV2).

We have demonstrated the utility of SpotClean to adjust for spot swapping and, in doing so, to provide improved estimates of expression. Since the probability of a spot-specific barcode binding reads from another spot increases as the distance between spots decreases, most of the adjustments made by SpotClean are local (i.e. reads are reassigned from one spot to a nearby spot). Given this, SpotClean will have only a modest impact on some downstream analyses, but a more major impact on others. Specifically, since average expression within a region will remain largely unchanged post SpotClean, downstream analyses that rely on average expression (e.g. DE analyses) will show only slight improvements over the raw data, as shown here. Modest improvements can also be expected for data where clusters are easily separated. However, for more specific analyses and/or more subtle signals, the effects of SpotClean are greater. Specifically, SpotClean provides substantial improvements in marker gene analyses by decreasing expression in regions where

markers are known to be lowly expressed, while maintaining expression levels in other regions. In addition, SpotClean substantially improves clustering results and spot annotations in situations where regions are not easily separated, which may have important implications for clinical applications of the ST technology (e.g. in cancer diagnosis and staging).

Spot swapping is a novel artifact that has not been discovered before. As a result, there still remain a number of interesting problems to investigate. As shown in section 4.4, we have investigated the potential relationship between bleeding rate and gene biotype as well as permeabilization time in a computational way using publicly available data. However, a carefully designed experiment could better answer these questions. For example, the relationship between bleeding rate and permeabilization time can be more rigorously evaluated by conducting ST experiments with varying permeabilization times while controlling everything else. Other factors such as species, tissue type, and tissue preparation procedures may also be evaluated via controlled experiments.

In summary, spatial transcriptomics provides unprecedented opportunity to address biological questions and enhance patient care, but artifacts induced by spot swapping must be adjusted for to ensure that maximal information is obtained from these powerful experiments. SpotClean provides for more accurate estimates of expression, thereby improving spot annotations and increasing the power and precision of downstream analyses.

A APPENDIX OF "CB2 IMPROVES POWER OF CELL DETECTION IN DROPLET-BASED SINGLE-CELL RNA SEQUENCING DATA"

A.1 Supplementry Figures and Tables

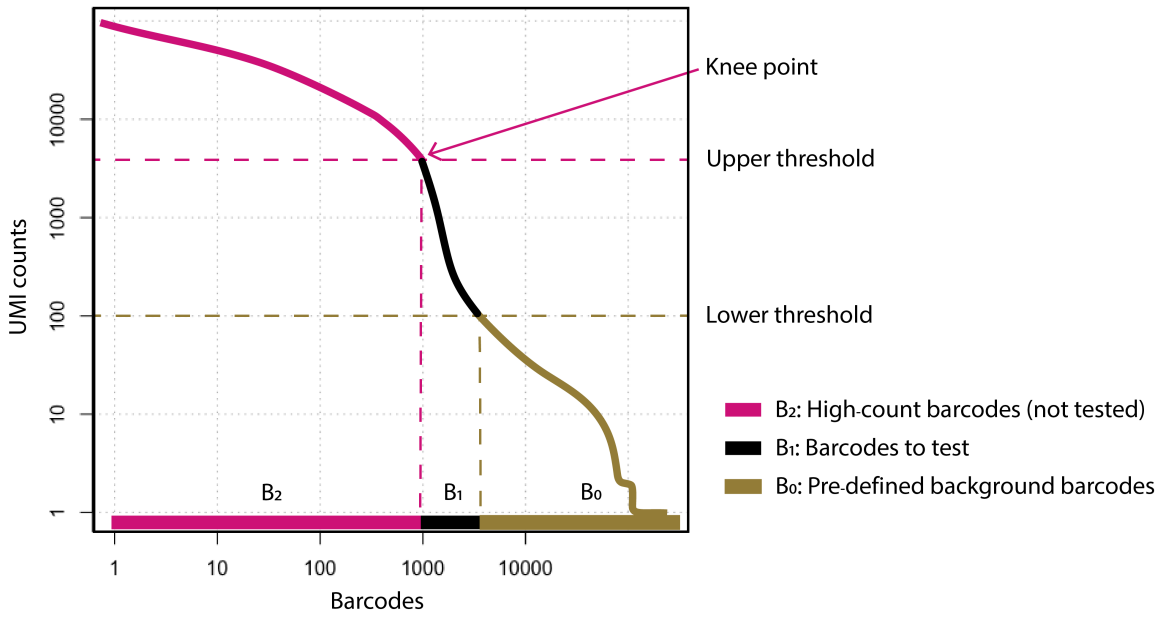

Figure A.1: Partition of barcode groups in CB2. UMI counts are plotted against barcodes that have been ordered by total count. Both CB2 and ED automatically call high count barcodes real cells (they are not tested); low count barcodes are considered background. The remaining barcodes are tested. By default, the high count (upper) threshold is defined by the knee point, where the counts vs. rank curve begins to drop rapidly; the low count threshold is set to 100. Either, or both, may be changed by a user.

Figure A.2: The average power and average FDR of CB2 and ED for SIM IB data (average taken over 5 simulated datasets). SIM IB is similar to SIM IA, but in SIM IB 10% of the genes in the real cells are shuffled making the real cells more different from the background and therefore easier to identify (Figure 2.3). Since both CB2 and ED automatically identify high count barcodes as real cells (they are not subject to statistical test; Figure A.1), we report results for all barcodes as well as those tested by CB2 and ED. The top panel shows the average power for tested barcodes; the middle panel for tested as well as high count barcodes. The bottom panel shows the average FDR. For the PanT4K dataset, all G1 cells are above the upper threshold and so no barcodes were tested (as a result, power for tested barcodes is not defined).

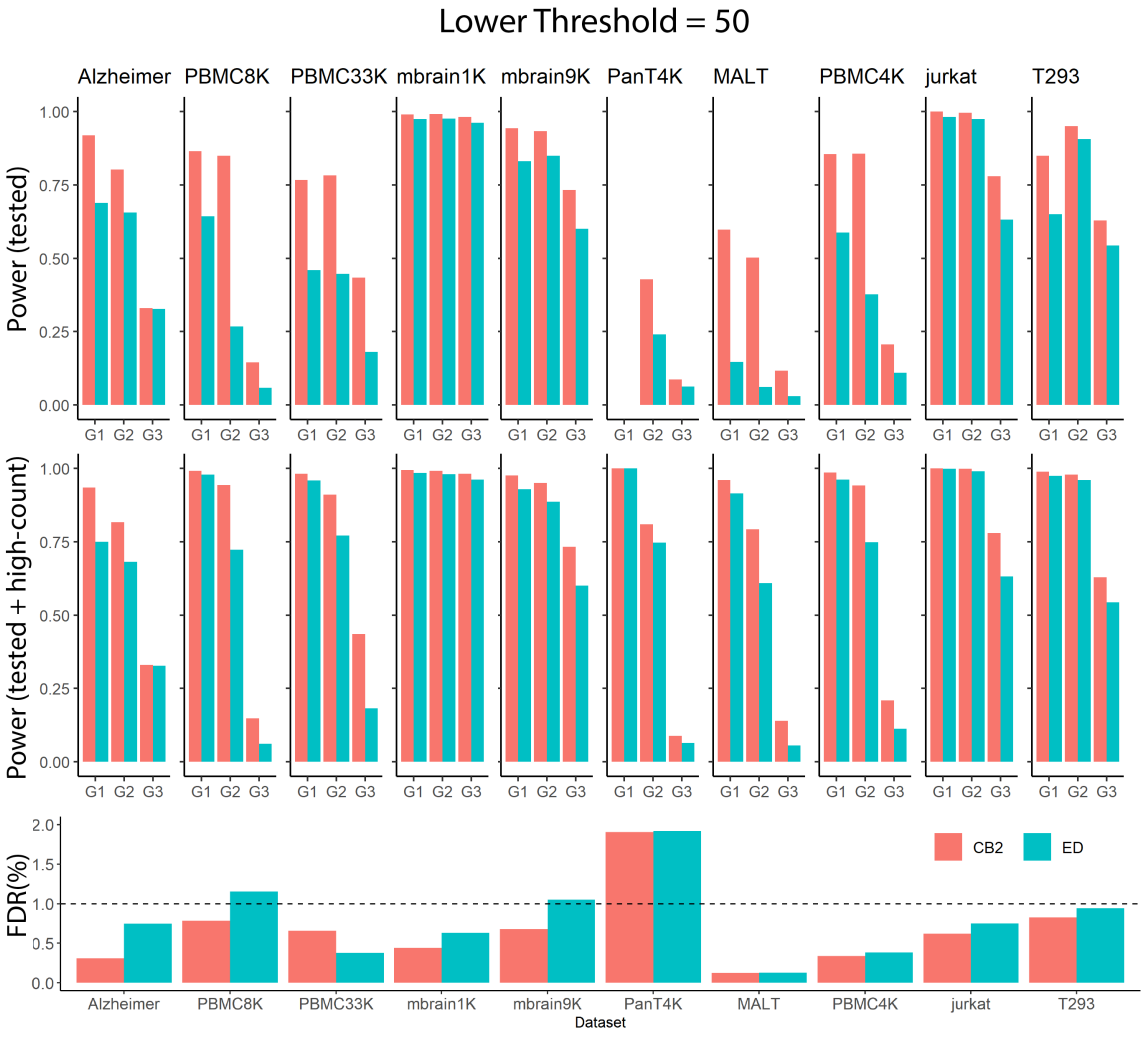

Figure A.3: The average power and average FDR of CB2 and ED for SIM IA data (average taken over 5 simulated datasets), with lower threshold = 50.

Figure A.4: The average power and average FDR of CB2 and ED for SIM IA data (average taken over 5 simulated datasets), with lower threshold = 150.

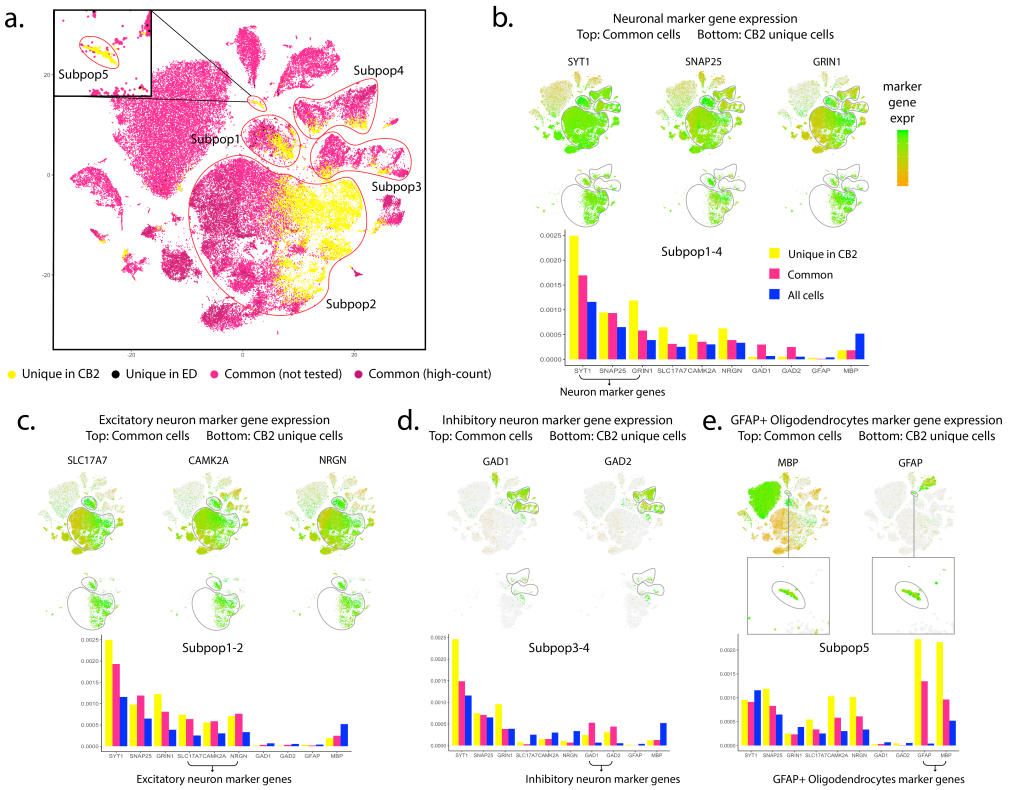

Figure A.5: Additional analysis of Alzheimer dataset. Mathys et al. (2019) considered marker genes to identify neurons (SYT1, SNAP25, GRIN1), excitatory neurons (SLC17A7, CAMK2A, NRGN), inhibitory neurons (GAD1, GAD2) and oligodendrocytes (MBP). (a) The same t-SNE plot as in Figure 2.5. Panels (b)-(e) show that t-SNE plot colored by marker gene expression in all cells (upper) as well as those identified uniquely by CB2 (lower). Distribution plots of the 10 marker genes within the specified subpopulations are shown in the bottom panels for the common cells (pink) and those uniquely identified by CB2 (yellow). Expression levels of these markers calculated across all cells are shown in blue as a reference. Cells uniquely identified by CB2 have marker gene expression patterns similar to those observed in the cells identified in common between CB2 and ED, providing further support that they are real cells. As shown in panel (e), the novel subpopulation identified by CB2 is the only subpopulation showing high expression of both oligodendrocyte and astrocyte marker genes, suggesting that this group may be mixed phenotype glial cells (GFAP+ oligodendrocytes) (Dyer et al., 2000). Panel (f) shows distribution plots of the 100 genes having highest average expression in Subpop5 for cells identified by both CB2 and ED (upper) and identified uniquely by CB2 (middle). The estimated background distribution is also shown (lower). Cells uniquely identified by CB2 in Subpop5 have a distribution similar to other Subpop5 cells and differ from the background.

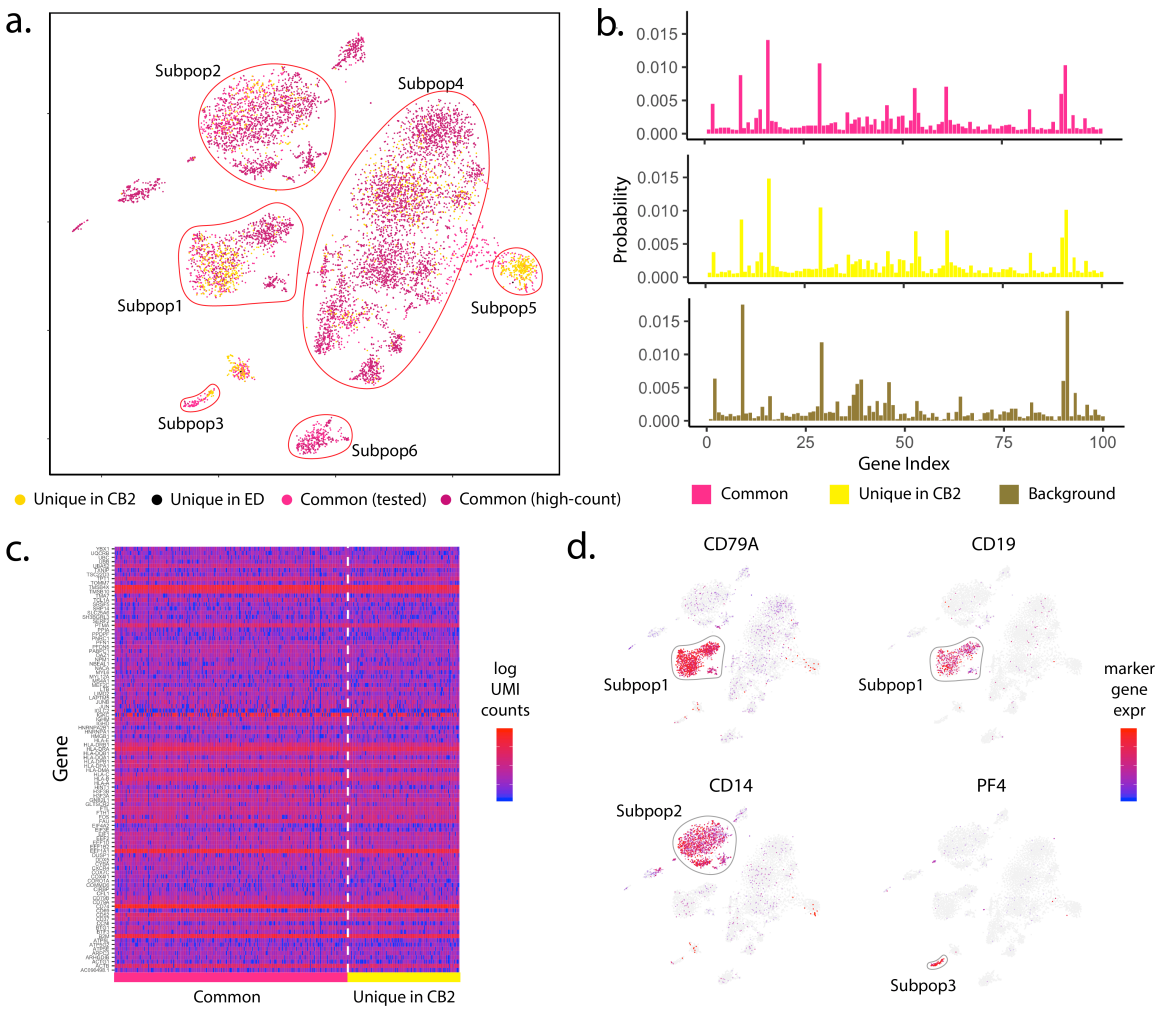

Figure A.6: Analysis of PBMC8K dataset. (a) t-SNE plot of cells identified by CB2 and ED. High-count barcodes exceeding an upper threshold are identified as real cells by both methods without a statistical test (dark pink); barcodes identified as cells by both methods following statistical test are shown in pink. Cells identified uniquely by CB2 (yellow) and ED (black) are also shown. (b) Distribution plots of the 100 genes having highest average expression in Subpop1 are shown for cells identified by both CB2 and ED (upper) and identified uniquely by CB2 (middle). The estimated background distribution is also shown (lower). Cells uniquely identified by CB2 in Subpop1 have a distribution similar to other Subpop1 cells and differ from the background. (c) Heatmap of log transformed raw UMI counts for the same 100 genes for barcodes identified by CB2 and ED (left) and barcodes uniquely identified by CB2 (right). (d) t-SNE plots of cells colored by known marker genes for B-cells (CD79A, CD19), CD14+ Monocytes (CD14), and Megakaryocytes (PF4)(Zheng et al., 2017)

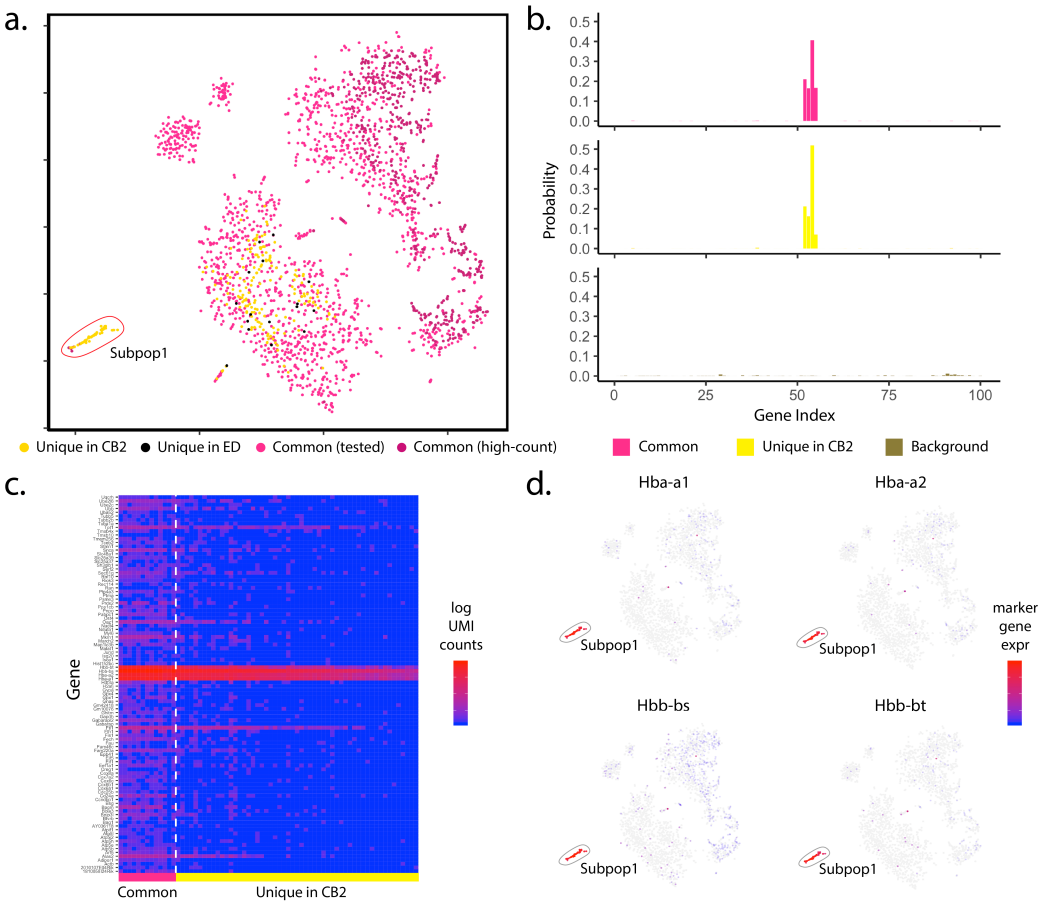

Figure A.7: Analysis of mbrain1K dataset. (a) t-SNE plot of cells identified by CB2 and ED. High-count barcodes exceeding an upper threshold are identified as real cells by both methods without a statistical test (dark pink); barcodes identified as cells by both methods following statistical test are shown in pink. Cells identified uniquely by CB2 (yellow) and ED (black) are also shown. (b) Distribution plots of the 100 genes having highest average expression in Subpop1 are shown for cells identified by both CB2 and ED (upper) and identified uniquely by CB2 (middle). The estimated background distribution is also shown (lower). Cells uniquely identified by CB2 in Subpop1 have a distribution similar to other Subpop1 cells and differ from the background. (c) Heatmap of log transformed raw UMI counts for the same 100 genes for barcodes identified by CB2 and ED (left) and barcodes uniquely identified by CB2 (right). (d) t-SNE plots of cells colored by expression of hemoglobin-related marker genes (Hba-a1, Hba-a2, Hbb-bs, Hbb-bt). (b)-(d) indicate that CB2 reveals a novel subpopulation of cells with expression patterns consistent with high stress and neurogenerative disorders in both human(Vanni et al., 2018) and mouse(Stankiewicz et al., 2014) brain studies.

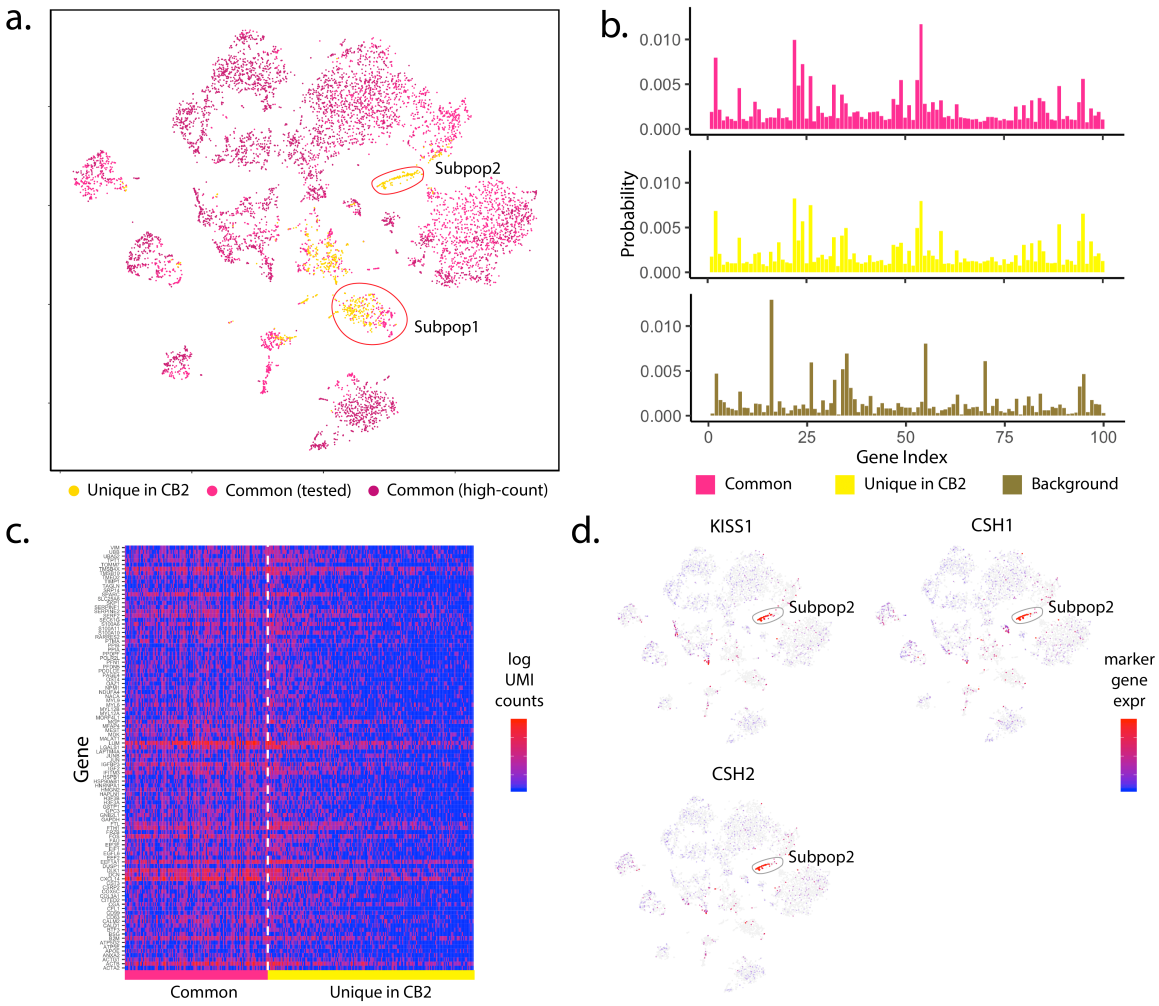

Figure A.8: Analysis of placenta dataset. (a) t-SNE plot of cells identified by CB2 and ED. High-count barcodes exceeding an upper threshold are identified as real cells by both methods without a statistical test (dark pink); barcodes identified as cells by both methods following statistical test are shown in pink. Cells identified uniquely by CB2 (yellow) and ED (black) are also shown. (b) Distribution plots of the 100 genes having highest average expression in Subpop1 are shown for cells identified by both CB2 and ED (upper) and identified uniquely by CB2 (middle). The estimated background distribution is also shown (lower). Cells uniquely identified by CB2 in Subpop1 have a distribution similar to other Subpop1 cells and differ from the background. (c) Heatmap of log transformed raw UMI counts for the same 100 genes for barcodes identified by CB2 and ED (left) and barcodes uniquely identified by CB2 (right). (d) t-SNE plots of cells colored by expression of a placenta-specific marker gene (KISS1) and marker genes related to placental lactogen secretion (CSH1, CSH2)(Mannik et al., 2010). Panels (a) and (d) indicate that CB2 reveals a novel subpopulation (Subpop2) that may be related to placental lactogen secretion.

Dataset	Link
Alzheimer	https://www.synapse.org/#!Synapse:syn16780177
PBMC8K	https://support.10xgenomics.com/single-cell-gene-expression/datasets/2.1.0/pbmc8k
PBMC33K	https://support.10xgenomics.com/single-cell-gene-expression/datasets/1.1.0/pbmc33k
mbrain1K	https://support.10xgenomics.com/single-cell-gene-expression/datasets/2.1.0/neurons_900
mbrain9K	https://support.10xgenomics.com/single-cell-gene-expression/datasets/2.1.0/neuron_9k
PanT4K	https://support.10xgenomics.com/single-cell-gene-expression/datasets/2.1.0/t_4k
MALT	https://support.10xgenomics.com/single-cell-gene-expression/datasets/3.0.0/malt_10k_protein_v3
PBMC4K	https://support.10xgenomics.com/single-cell-gene-expression/datasets/2.1.0/pbmc4k
jurkat	https://support.10xgenomics.com/single-cell-gene-expression/datasets/1.1.0/jurkat
T293	https://support.10xgenomics.com/single-cell-gene-expression/datasets/1.1.0/293t
placenta	https://jmlab-gitlab.cruk.cam.ac.uk/publications/EmptyDrops2017-DataFiles

Table A.1: Links to all datasets used in chapter 2.

Dataset	Threshold		
	90%	80%	70%
Alzheimer	1	2	2
PBMC8K	0	0	0
PBMC33K	0	0	0
mbrain1K	0	1	1
mbrain9K	0	1	1
PanT4K	0	0	0
MALT	1	1	2
PBMC4K	0	0	0
jurkat	0	0	1
T293	0	0	0
placenta	0	0	1

Table A.2: Number of novel subpopulations identified by CB2 in each dataset.

A.2 Software versions for reproducibility

Below are the versions of language and packages at the time generating results in chapter 2. For cell identification with *scCB2*-0.99.12 and *DropletUtils*-1.5.4, the latest version of R was used: 3.7-devel (2019-07-17 r76847). Other packages were not yet compatible or not stable with the R developers version and so for *scran*-1.12.1, *Seurat*-3.1.0, and *ggplot2*-3.2.1, R 3.6.0 (2019-04-24 r76423) was used.

A.3 Data and code availability

Links to all the public datasets used in chapter 2 are listed in Table A.1. The R package *scCB2* is available at `https://bioconductor.org/packages/release/bioc/html/scCB2.html`. All simulation codes and case study data analysis scripts are available at `https://github.com/zijianni/codes-for-CB2-paper`.

B Appendix of "Benchmarking cell detection algorithms for droplet-based single-cell RNA-seq data"

B.1 Supplementry Figures and Tables

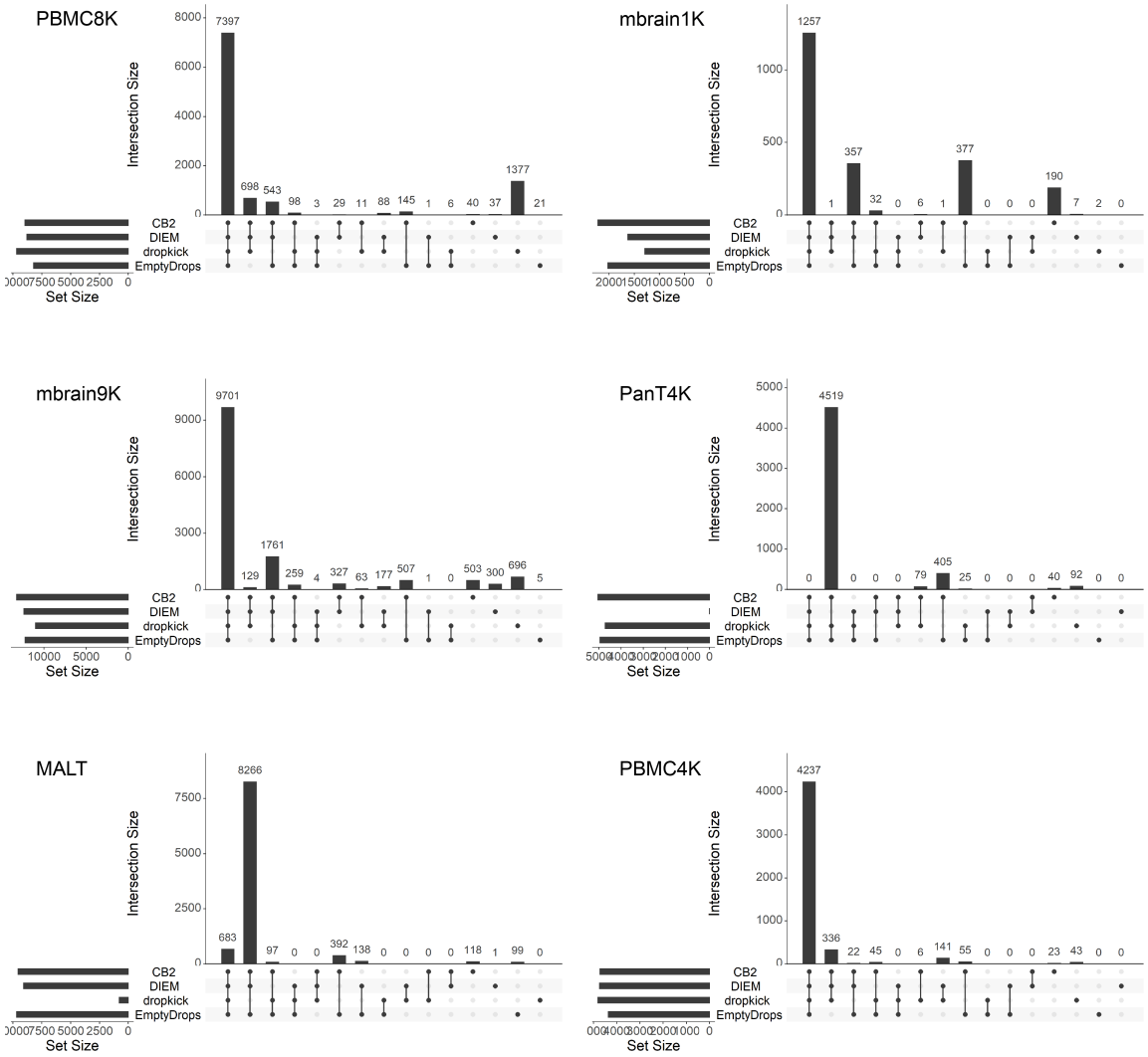

Figure B.1: UpSet plots for 6 datasets (PBMC8K, mbrain1K, mbrain9K, PanT4K, MALT, PBMC4K) showing the number of common and distinct barcodes of the four cell barcode sets as well as the size of the sets. For EmptyDrops, CB2, and DIEM, the background thresholds are 100, 100, 150, 100, 100, 100, respectively.

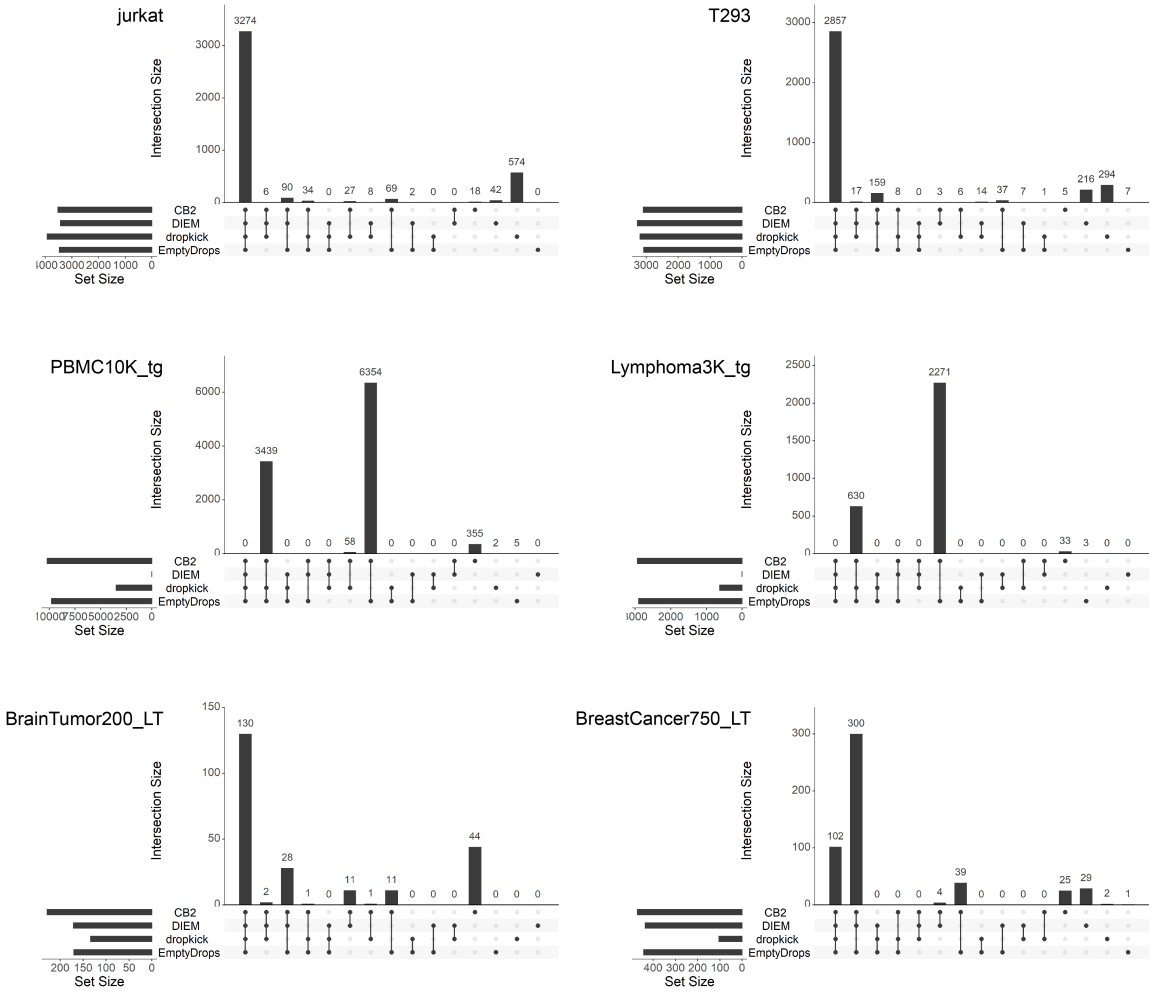

Figure B.2: UpSet plots for 6 datasets (jurkat, T293, PBMC10K_tg, Lymphoma3K_tg, BrainTumor200_LT, BreastCancer750_LT) showing the number of common and distinct barcodes of the four cell barcode sets as well as the size of the sets. For EmptyDrops, CB2, and DIEM, the background thresholds are 100, 100, 10, 10, 100, 300, respectively.

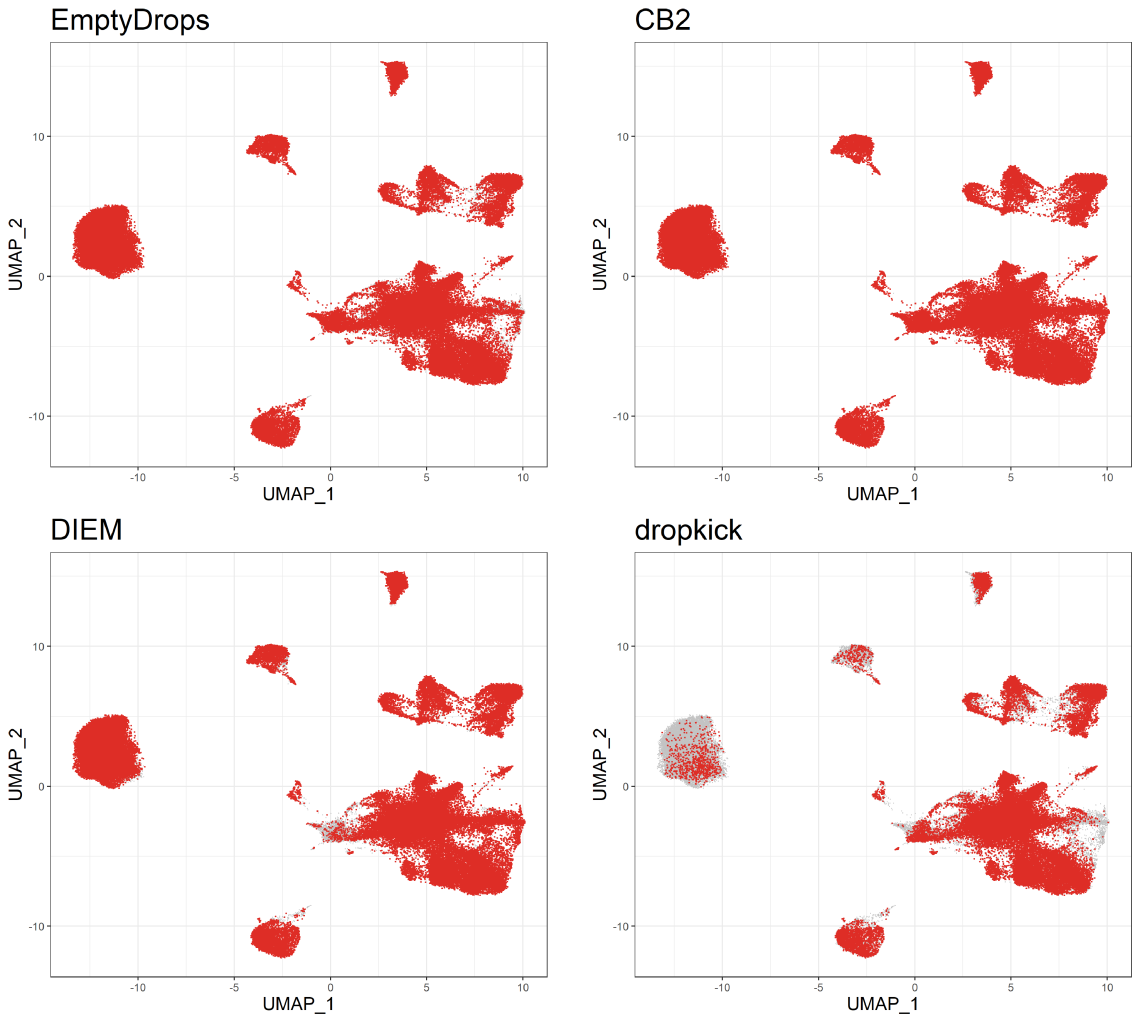

Figure B.3: UMAP plots of the Alzheimer data. Barcodes highlighted in red in each panel are the ones detected as cells by each of the four methods.

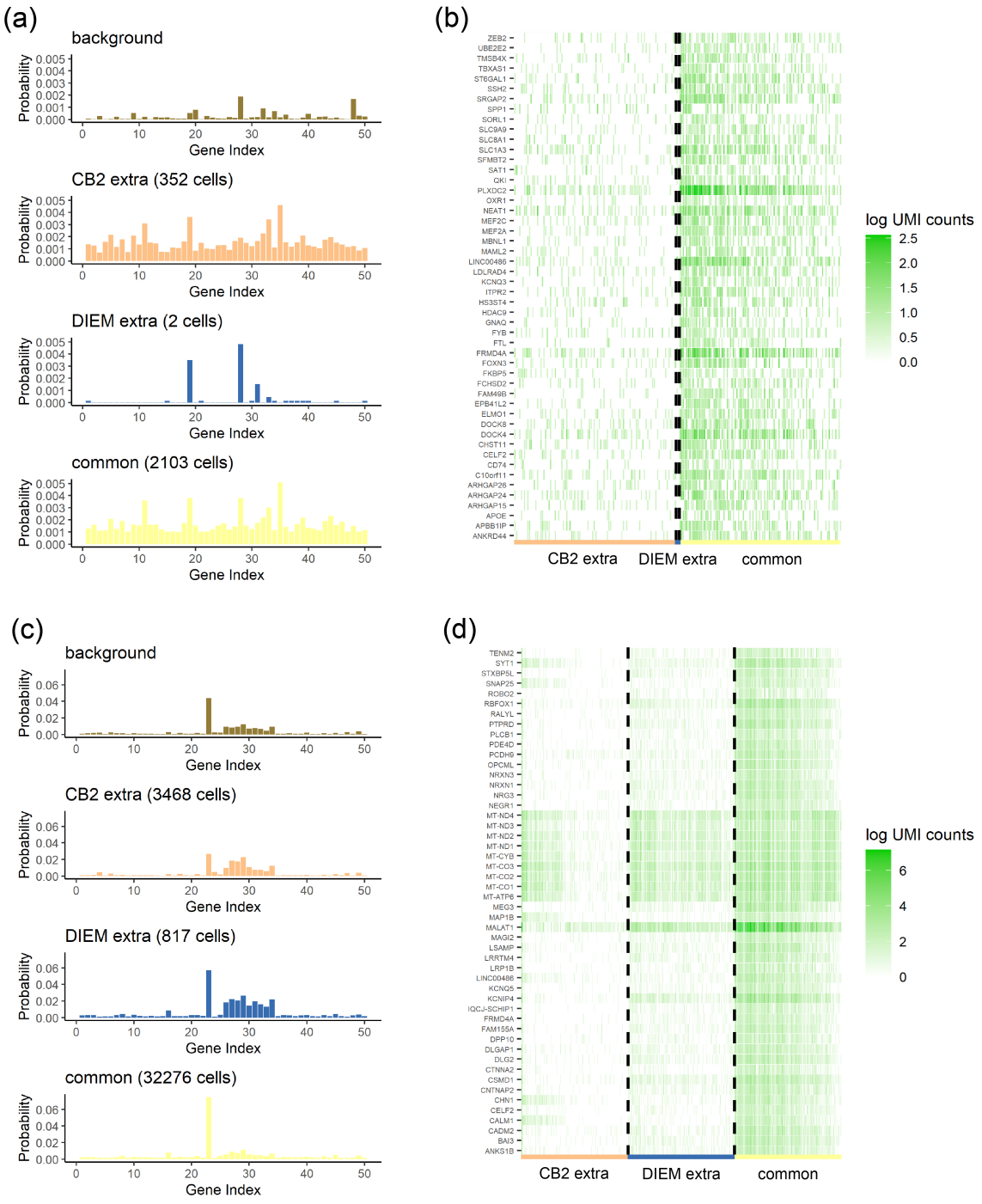

Figure B.4: Distribution plots and expression heatmaps of the top 50 genes in different barcode groups for (a)-(b) microglia and (c)-(d) excitatory neurons. For the excitatory neurons, housekeeping genes are not filtered out from the top 50 genes since they contributed the most to distinguishing cell barcodes against background distribution.

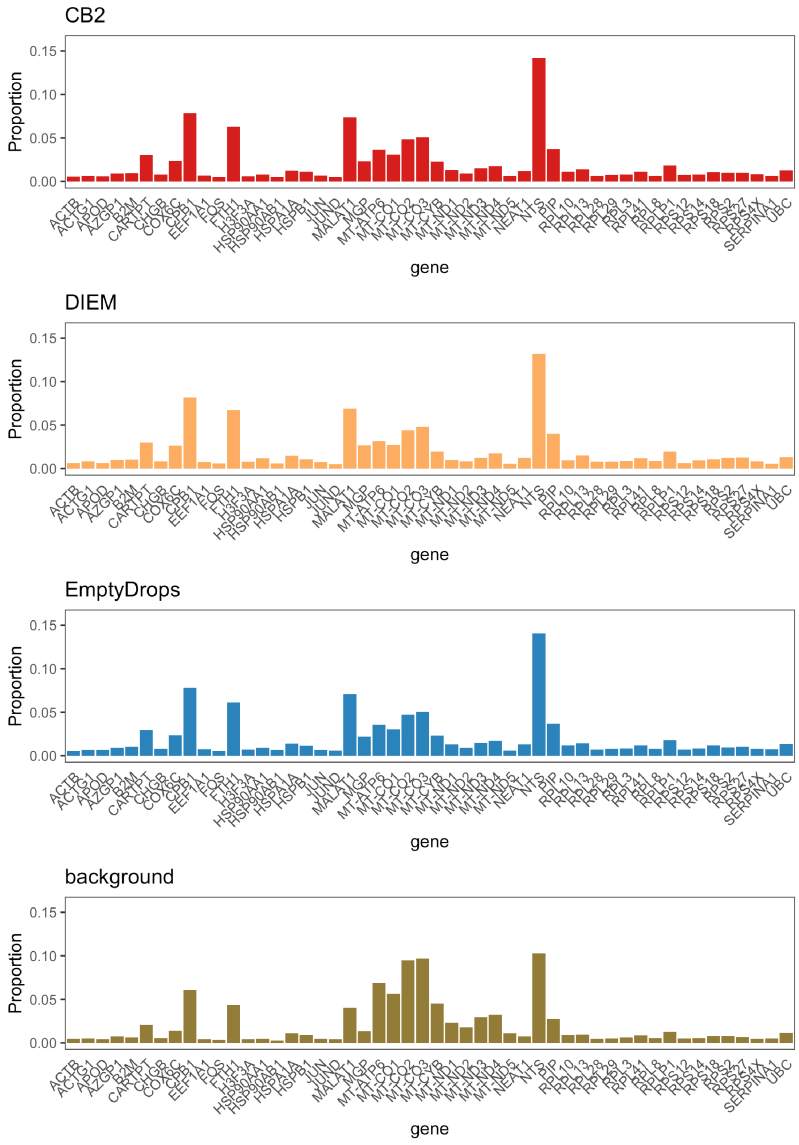

Figure B.5: Distribution plots of the top 50 genes in barcodes with UMI counts between 101 and 300 that are identified as cells by CB2, DIEM, and EmptyDrops for BreastCancer750_LT data. These barcodes have similar overall distributions across methods, and are similar to the background barcodes with UMI counts below 100, indicating that the default threshold=100 is not sufficiently high to accurately estimate background distribution, which results in false positive cells.

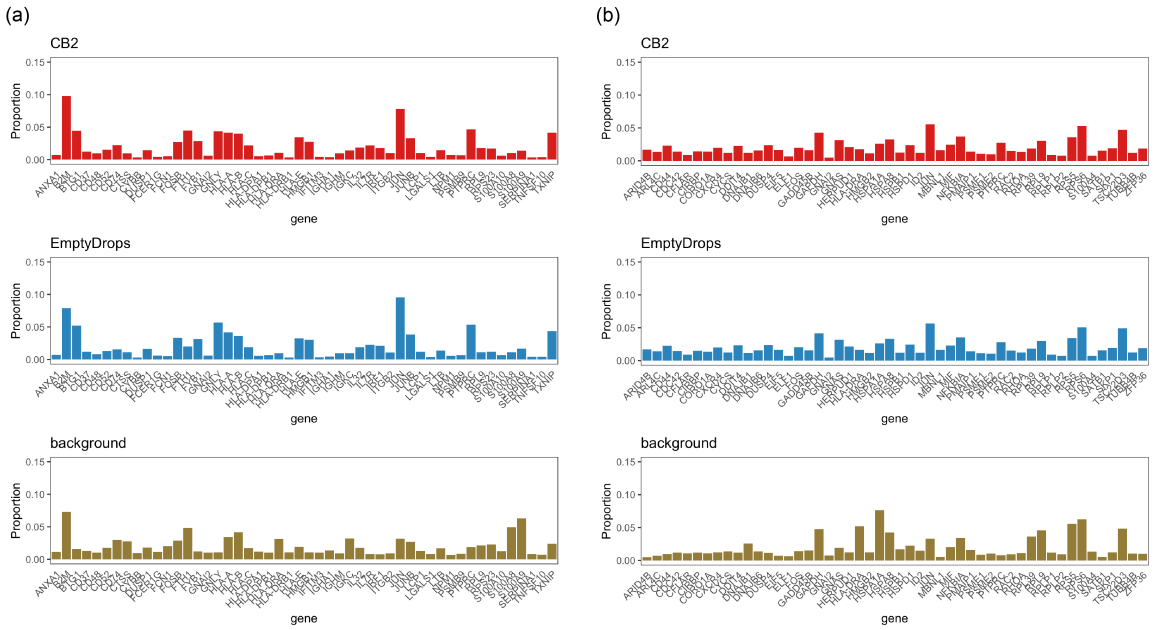

Figure B.6: Distribution plots of the top 50 genes in barcodes with UMI counts between 101 and 300 that are identified as cells by CB2, DIEM, and EmptyDrops for (a)PBMC10K_tg and (c)Lymphoma3K_tg data. These barcodes have similar overall distributions across methods, but they different from the background barcodes with UMI counts below 10, indicating that these barcodes are real small cells which will be incorrectly filtered out under the default threshold=100.

Dataset	Link
PBMC8K	https://support.10xgenomics.com/single-cell-gene-expression/datasets/2.1.0/pbmc8k
mbrain1K	https://support.10xgenomics.com/single-cell-gene-expression/datasets/2.1.0/neurons_900
mbrain9K	https://support.10xgenomics.com/single-cell-gene-expression/datasets/2.1.0/neuron_9k
PanT4K	https://support.10xgenomics.com/single-cell-gene-expression/datasets/2.1.0/t_4k
MALT	https://support.10xgenomics.com/single-cell-gene-expression/datasets/3.0.0/malt_10k_protein_v3
PBMC4K	https://support.10xgenomics.com/single-cell-gene-expression/datasets/2.1.0/pbmc4k
jurkat	https://support.10xgenomics.com/single-cell-gene-expression/datasets/1.1.0/jurkat
T293	https://support.10xgenomics.com/single-cell-gene-expression/datasets/1.1.0/293t
PBMC10K_tg	https://www.10xgenomics.com/resources/datasets/pbm-cs-from-a-healthy-donor-targeted-immunology-panel-3-1-standard-4-0-0
Lymphoma3K_tg	https://www.10xgenomics.com/resources/datasets/hodgkins-lymphoma-dissociated-tumor-targeted-gene-signature-panel-3-1-standard-4-0-0
BrainTumor200_LT	https://www.10xgenomics.com/resources/datasets/200-sorted-cells-from-human-glioblastoma-multiforme-3-lt-v-3-1-3-1-low-6-0-0
BreastCancer750_LT	https://www.10xgenomics.com/resources/datasets/750-sorted-cells-from-human-invasive-ductal-carcinoma-3-lt-v-3-1-3-1-low-6-0-0
mheart1K	https://www.10xgenomics.com/resources/datasets/1-k-heart-cells-from-an-e-18-mouse-v-3-chemistry-3-standard-3-0-0
mheart10K	https://www.10xgenomics.com/resources/datasets/10-k-heart-cells-from-an-e-18-mouse-v-3-chemistry-3-standard-3-0-0
PBMC1K	https://www.10xgenomics.com/resources/datasets/1-k-pbm-cs-from-a-healthy-donor-v-3-chemistry-3-standard-3-0-0
PBMC10K	https://www.10xgenomics.com/resources/datasets/10-k-pbm-cs-from-a-healthy-donor-v-3-chemistry-3-standard-3-0-0

Table B.1: Links to all datasets used in chapter 3.

B.2 Software versions for reproducibility

Below are the versions of language and packages at the time generating results in chapter 3: R-4.1.0. R/DropletUtils-1.14.1. R/scCB2-1.5.1. R/diem-2.3.0. R/Seurat-4.0.5. Python-3.7.10. Python/dropkick-1.2.6.

B.3 Data and code availability

The Alzheimer study dataset was downloaded from `https://www.synapse.org/#!Synapse:syn16780177`. Other public datasets in this study are available at the 10x Genomics website (https://support.10xgenomics.com/singlecell-gene-expression/datasets) (Table B.1). All simulation codes and case study data analysis scripts are available at `https://github.com/zijianni/codes_for_CB2_benchmark_paper`.

C APPENDIX OF "SPOTCLEAN ADJUSTS FOR SPOT SWAPPING IN SPATIAL TRANSCRIPTOMICS DATA"

C.1 Supplementary Figures and Tables

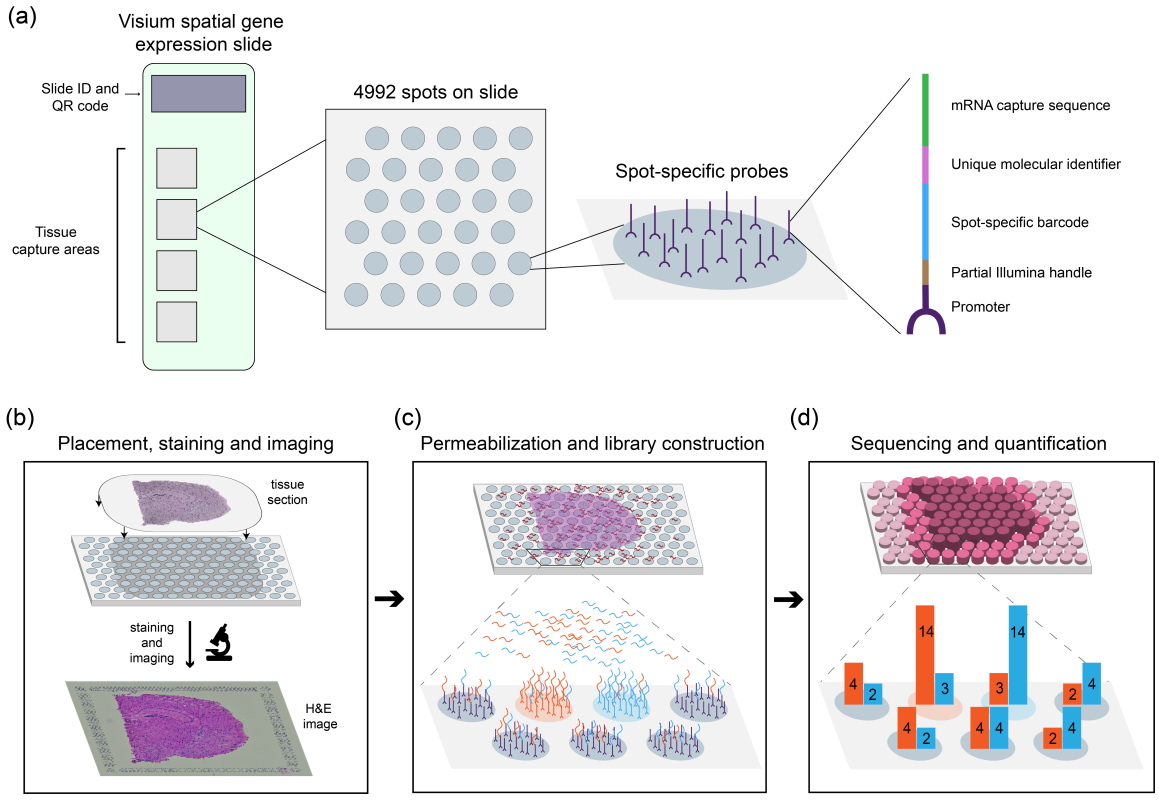

Figure C.1: Overview of the 10x Genomics Visium (10x) spatial transcriptomics experiment. (a) The 10x Visium spatial gene expression slide contains four tissue capture sites for processing multiple tissue samples simultaneously. Each capture site contains 4992 spots, with each spot containing millions of spot-specific probes that bind mRNA. (b) Fresh frozen or FFPE tissue is sectioned, placed on a capture area, and imaged, typically via Hematoxylin and Eosin (H&E) staining. (c) Following imaging, the tissue is permeabilized to release mRNA. The lower panel shows five background spots (gray) and two tissue spots (orange and blue). Due to spot swapping, mRNAs from one tissue spot bind probes at other spots. (d) The bound mRNAs are released, processed, sequenced and quantified to give a gene-by-spot matrix of UMI counts. In this hypothetical example, due to spot swapping, UMI counts at each of the seven spots are a mixture of mRNAs from the two distinct tissue spots.

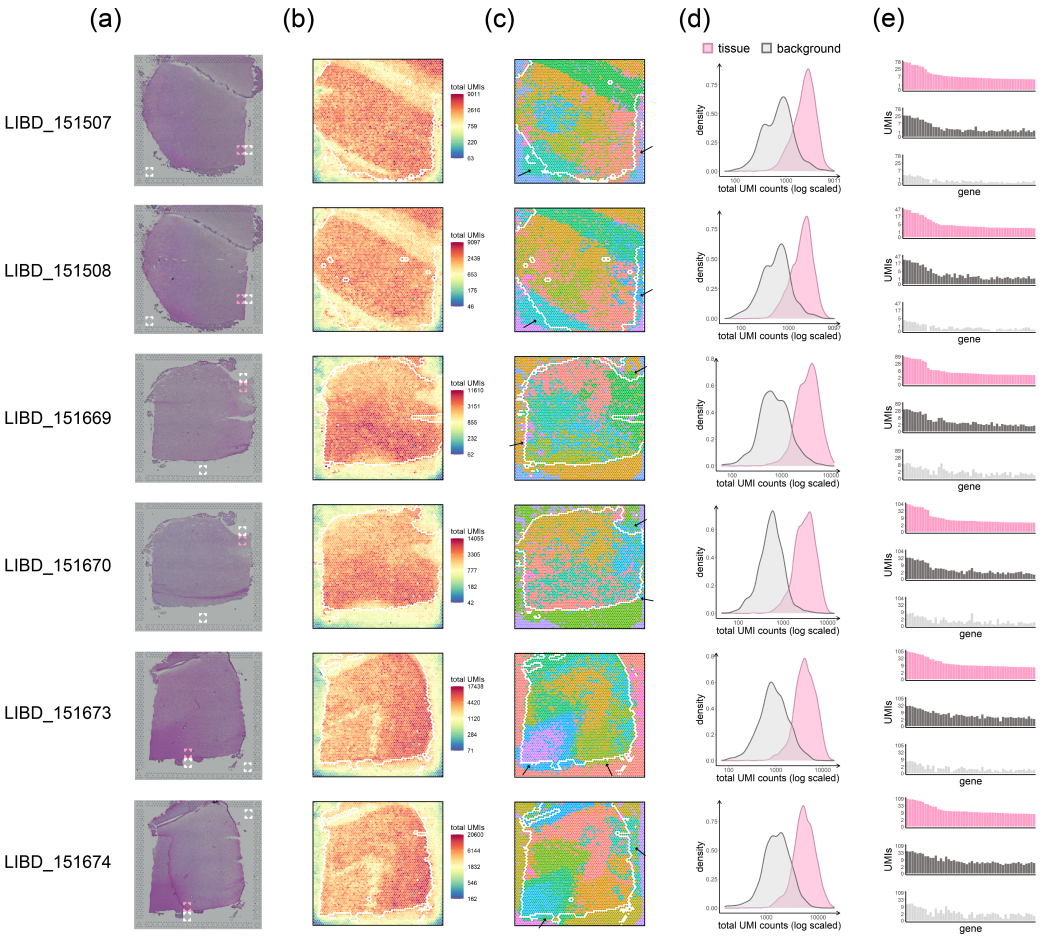

Figure C.2: Data from six human brain samples from Maynard et al. (2021). (a) H&E images for the six different samples. (b) UMI total counts in the background decrease with increasing distance from the tissue. (c) Spots on the slide are colored by their cluster membership via graph-based clustering (clusters not shown). Black arrows highlight areas of spot swapping. (d) UMI count densities for tissue and background spots show relatively high counts in the background. (e) Counts of the top 50 genes (genes with highest total UMI expression) from a select tissue region (upper), from a nearby background region (middle), and from a distant background region (bottom) show the similarity between expression in tissue spots and nearby background spots due to spot swapping from tissue to background, an effect that decreases as distance from the tissue increases. The tissue region and background regions used for each sample are highlighted in panel (a) in pink and white, respectively. Tissue spots on the perimeter (shown in white in panels (b) and (c)) were removed prior to calculating the summaries in panels (d)-(e) in an effort to ensure that the effects shown are not due to spots on the tissue-background boundary.

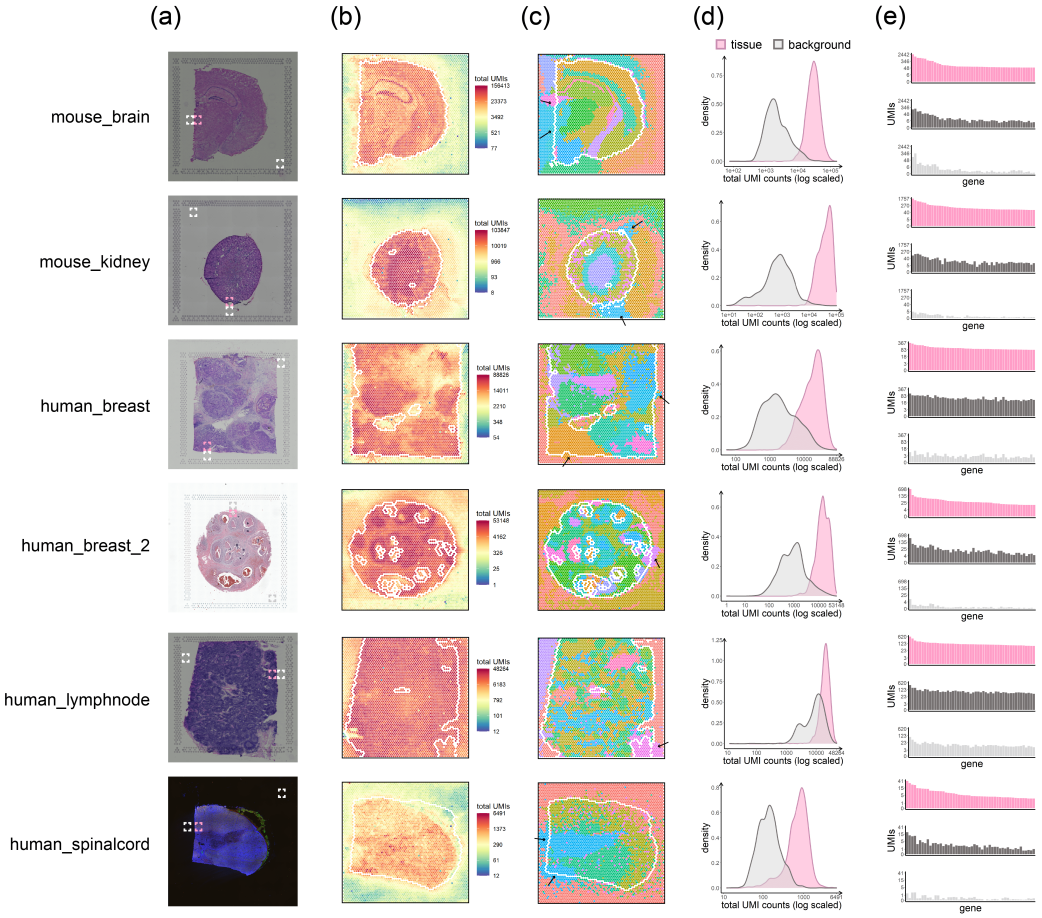

Figure C.3: Data from six publicly available 10x Visium datasets. (a) H&E images for the six different samples. (b) UMI total counts in the background decrease with increasing distance from the tissue. (c) Spots on the slide are colored by their cluster membership via graph-based clustering (clusters not shown). Black arrows highlight areas of spot swapping. (d) UMI count densities for tissue and background spots show relatively high counts in the background. (e) Counts of the top 50 genes (genes with highest total UMI expression) from a select tissue region (upper), from a nearby background region (middle), and from a distant background region (bottom) show the similarity between expression in tissue spots and nearby background spots due to spot swapping from tissue to background, an effect that decreases as distance from the tissue increases. The tissue region and background regions used for each sample are highlighted in panel (a) in pink and white, respectively. There is considerable overlap of tissue and background spots in the UMAP plots. Tissue spots on the perimeter (shown in white in panels (b) and (c)) were removed prior to calculating the summaries in panels (d)-(e) in an effort to ensure that the effects shown are not due to spots on the tissue-background boundary.

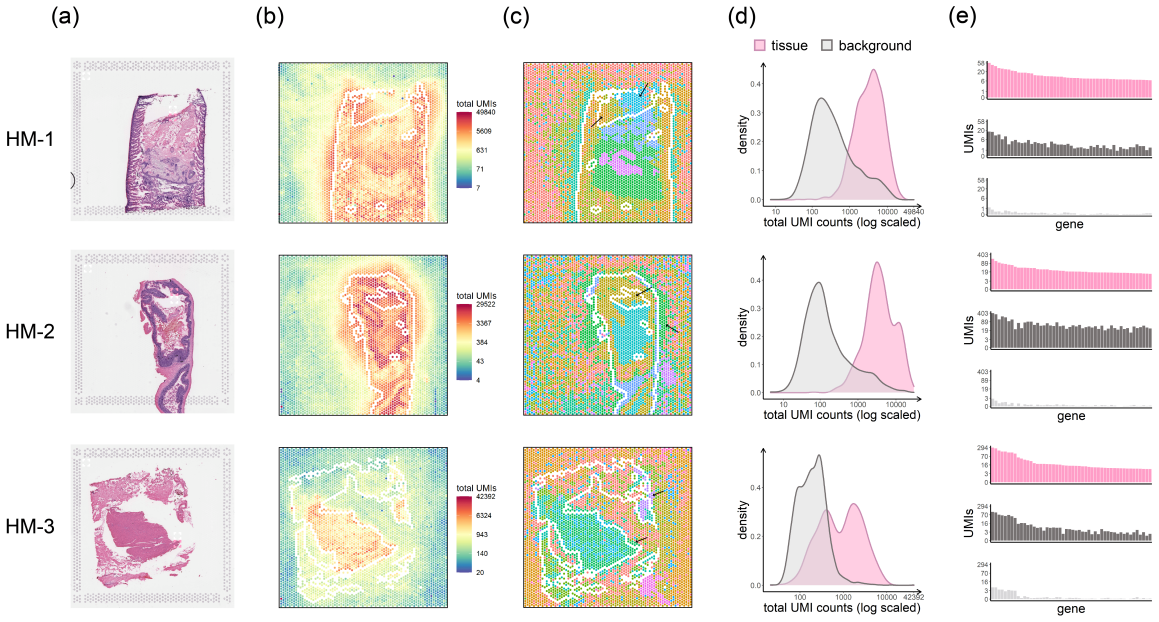

Figure C.4: Data from the three chimeric samples composed of human and mouse tissues. (a) H&E images for the three different samples. (b) UMI total counts in the background decrease with increasing distance from the tissue. (c) Spots on the slide are colored by their cluster membership via graph-based clustering (clusters not shown). Black arrows highlight areas of spot swapping. (d) UMI count densities for tissue and background spots show relatively high counts in the background. (e) Counts of the top 50 genes (genes with highest total UMI expression) from a select tissue region (upper), from a nearby background region (middle), and from a distant background region (bottom) show the similarity between expression in tissue spots and nearby background spots due to spot swapping from tissue to background, an effect that decreases as distance from the tissue increases. The tissue region and background regions used for each sample are highlighted in panel (a) in pink and white, respectively. There is considerable overlap of tissue and background spots in the UMAP plots. Tissue spots on the perimeter (shown in white in panels (b) and (c)) were removed prior to calculating the summaries in panels (d)-(e) in an effort to ensure that the effects shown are not due to spots on the tissue-background boundary.

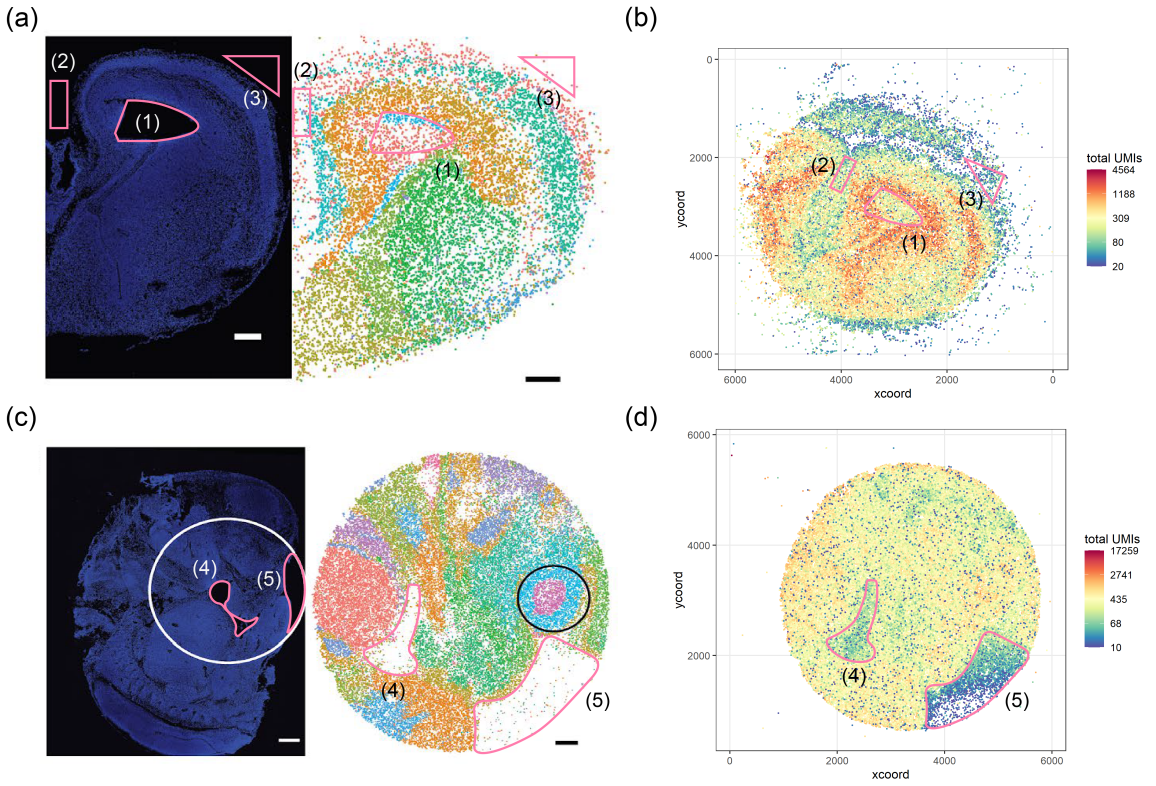

Figure C.5: Evidence of spot swapping in Slide-seqV2 data. (a) DAPI stain of E15 mouse brain (left) and Slide-seqV2 data of E15 brain with cluster labels (right) reported in Stickels et al. (2021). Three background regions identified from the DAPI image are shown in pink and labeled as (1), (2) and (3). The same regions are also identified in the graph-based clustering of beads. (b) Raw UMI counts data colored by total UMI counts for all spatial barcodes shows positive UMI counts detected in the three background regions. (c)-(d) are identical to (a)-(b), but for the E12.5 mouse embryo data. The black circle shown in (c) is part of the original image and not relevant here.

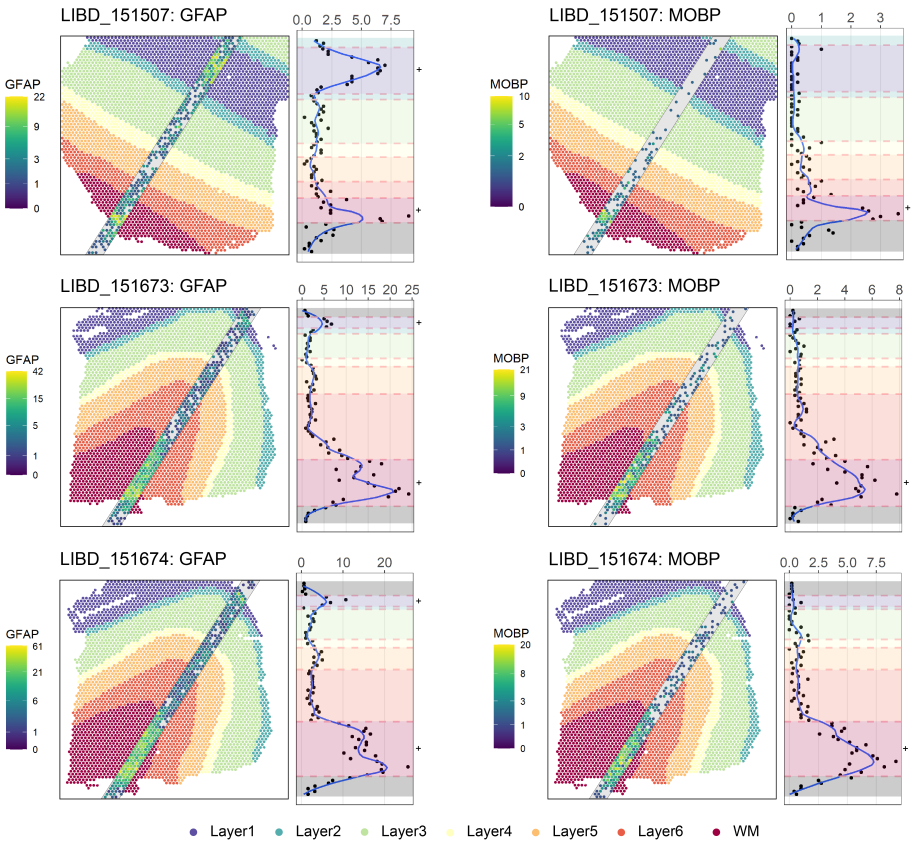

Figure C.6: The effect of spot swapping on layer-specific marker genes in three human brain sample datasets. The upper left panel shows the annotated LIBD_151507 sample and a stripe with width equal to five spots. Each spot in the stripe is colored by expression of GFAP, a marker for white matter (WM) and Layer1; the "+" denotes the regions where marker expression is expected to be high (here, WM and Layer 1). Average expression of each row in the stripe is shown in the right subpanel. The average is taken across the five spots for every row contained completely within a layer; for rows containing two layers, the average is taken across the three or four spots making up the major layer. When spot swapping occurs, marker expression is relatively high in nearby layers, as observed here. While it is possible that some increase in marker expression in adjacent tissue spots may be due to the presence of WM (or Layer1) cells at those spots, we note that the rate of expression decay into the background spots (where no cells are present) is similar to the rate of decay into adjacent tissue regions. Consequently, the possible presence of cells from a given layer in adjacent tissue spots outside that layer is not sufficient to fully explain the observed expression patterns shown here. The lower middle and lower left panels show the same plot for different tissue samples; the right panels show MOBP, another WM marker.

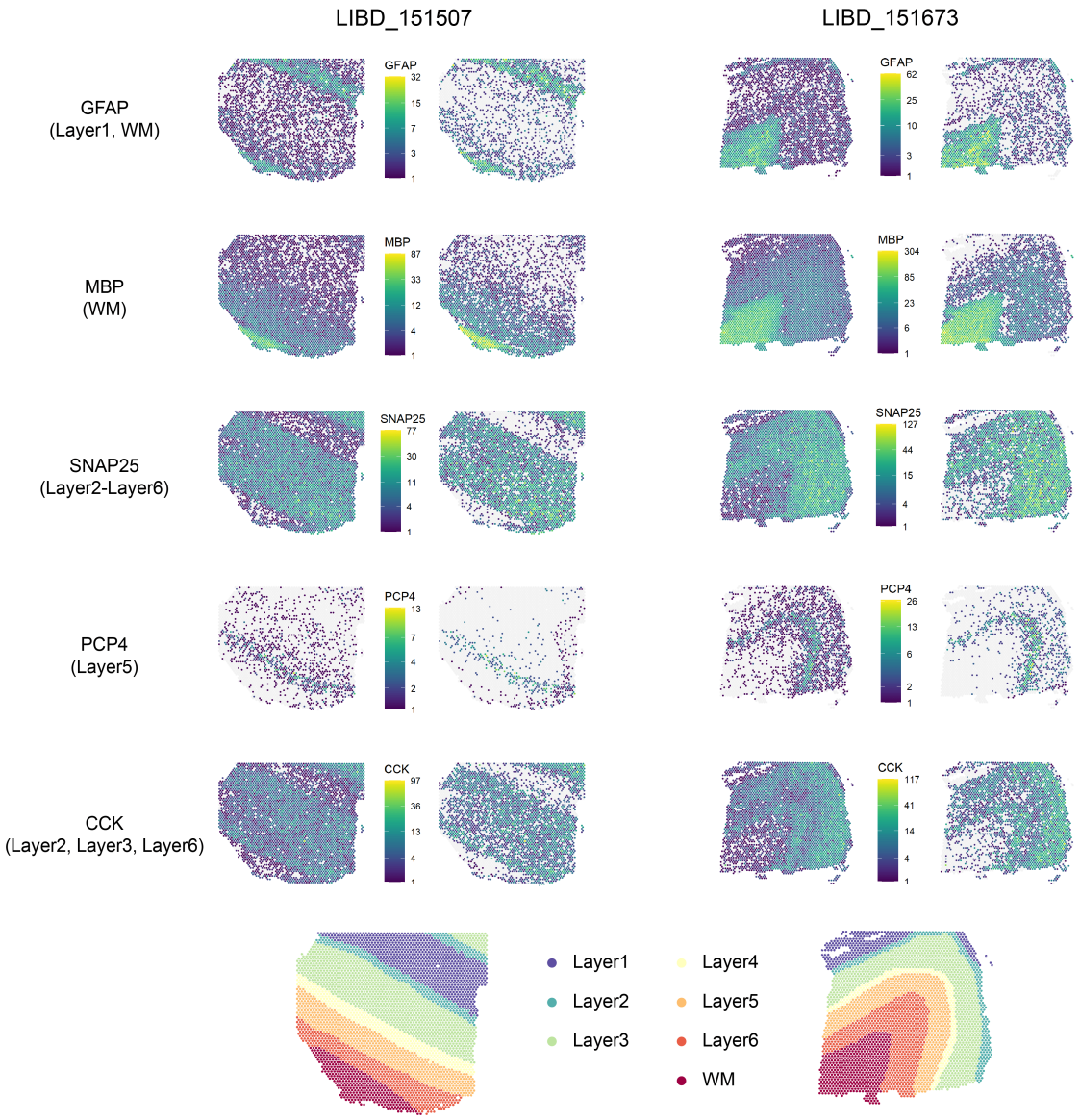

Figure C.7: Gene expression heatmaps for GFAP, MBP, SNAP25, PCP4, CCK in brain samples are shown for the raw and SpotClean decontaminated data. Columns 1 and 3 show raw expression in brain samples LIBD_151507 and LIBD_151673; columns 2 and 4 show SpotClean decontaminated data for these same samples. Brain layer annotations are also shown at the bottom. SpotClean maintains expression in the marker layers and reduces expression in adjacent layers, thereby increasing marker specificity.

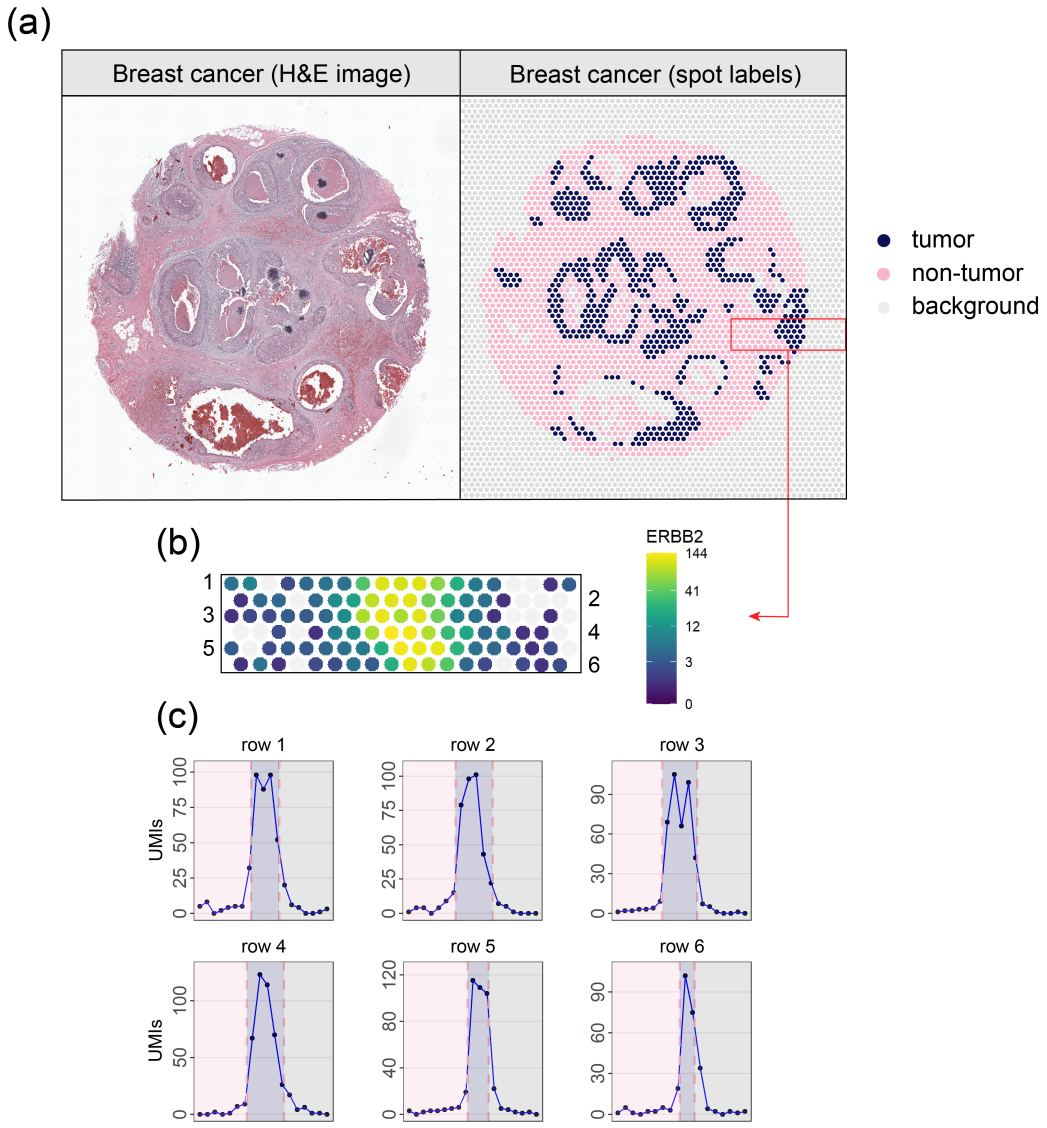

Figure C.8: The effect of spot swapping on breast cancer specific marker genes in a human breast cancer sample, human_breast_2. Panel (a) shows the H&E image (left) and spots annotated as tumor, non-tumor, and background via visual inspection (right). Panel (b) shows high expression for ERBB2 in the tumor spots of the inset; also shown is relatively high expression in adjacent spots that decreases with increasing distance from the tumor. Panel (c) shows ERBB2 expression for spots in each row of the inset. Some of the decrease into adjacent spots shown in panels (b) and (c) may be due to the presence of both tumor and normal cells in spots near the tumor tissue. However, this is unlikely given that the rate of decay into the adjacent spots is similar to the rate of decay into the background (where no tissue is present).

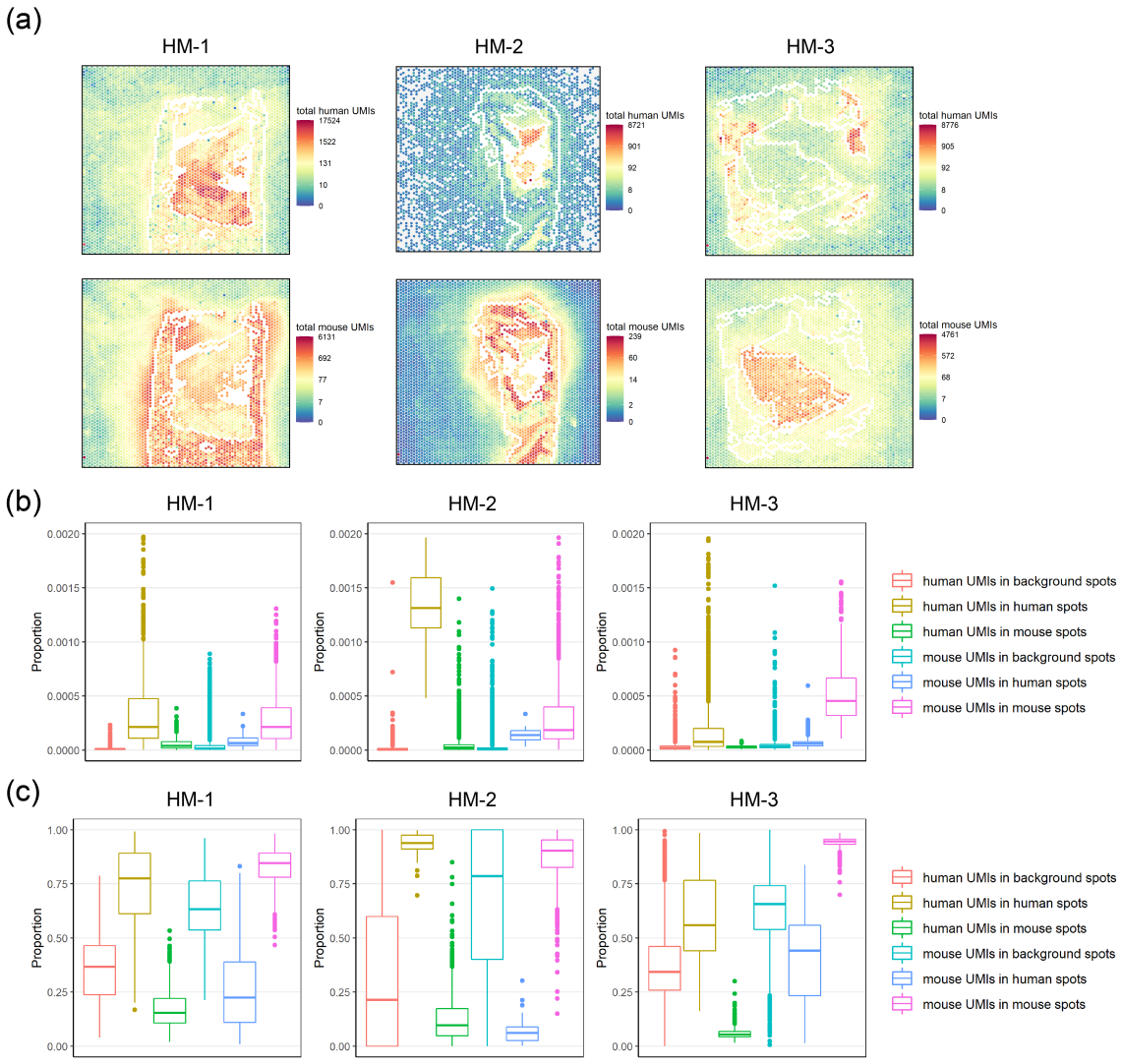

Figure C.9: Panel (a) shows total UMI counts in human-specific genes (upper) and mouse-specific genes (lower) for the three chimeric experiments. Panel (b) shows the proportion (out of total UMIs) of spot-swapped UMI counts (human-specific UMIs in background or mouse spots; mouse-specific UMIs in background or human spots). Also shown are the proportion of human-specific UMIs in human spots and mouse-specific UMIs in mouse spots. Note that there may be spot swapped reads in these latter proportions (e.g. reads from human spot t bound by probes at human spot t'), but they cannot be identified in this experiment. Panel (c) shows spot-specific proportions. Tissue spots on the perimeter as well as spots annotated as mixtures (shown in white in panel (a)) were removed prior to calculating the summaries in panels (b) and (c) in an effort to ensure that the effects shown are not due to spots on the tissue-background boundary.

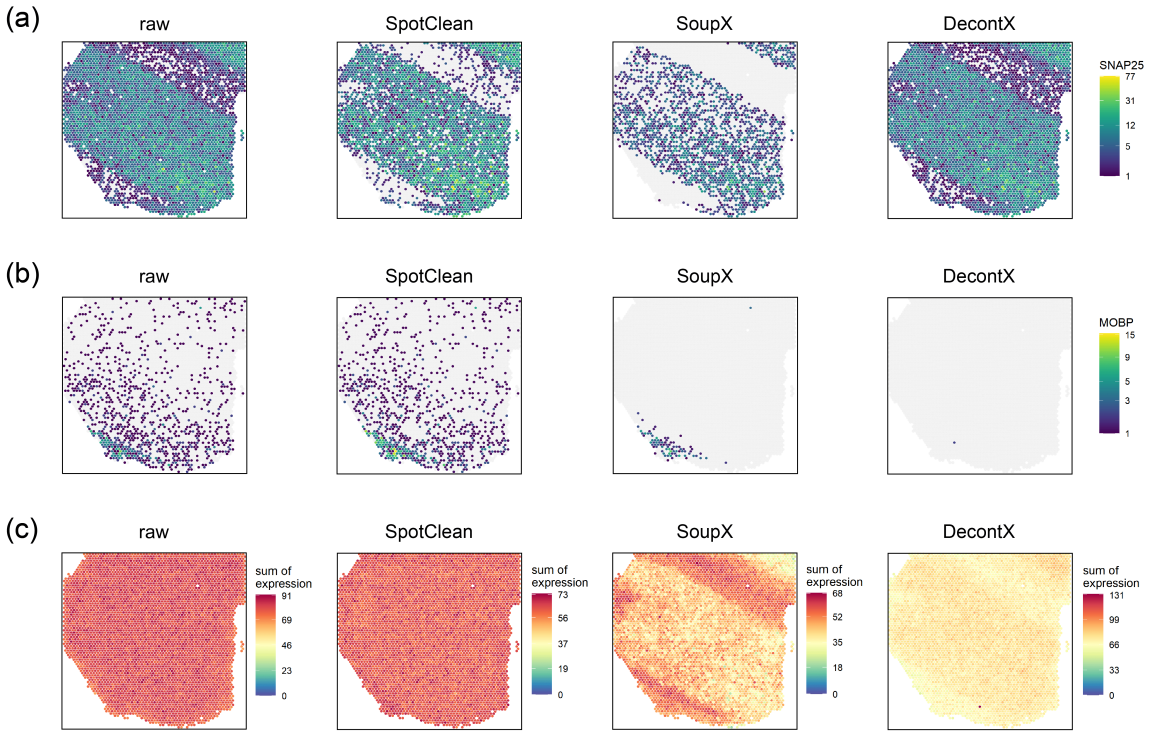

Figure C.10: Data from the human brain sample LIBD_151507. Data decontaminated by SpotClean (middle left), SoupX (middle right), and DecontX (right) for SNAP25 (a) and MOBP (b). The raw data is shown left. SoupX decontaminates SNAP25, but reduces expression even in Layer2-Layer6, where SNAP25 expression is expected to be high; DecontX imposes little change on this marker's expression. SoupX works well for MOBP, but DecontX removes almost all of the signal for this marker. Panel (c) shows results from SimV data generated using sample LIBD_151507; 500 spatially varying (SV) genes and 500 genes showing no change in expression across the slide were simulated (non-SV). Shown far left in panel (c) is summed expression for the 500 non-SV genes. To ensure that the summation is not dominated by a few highly expressed genes, gene-specific expression was scaled so that the maximum value of each gene equals 1. The same sum is shown for data decontaminated by SpotClean (middle left), SoupX (middle right), and DecontX (right). SoupX and DecontX impose artificial patterns upon non-SV genes.

Figure C.11: Shown for the six human brain sample datasets are the fold changes and p-values for 10 genes known to be differentially expressed (DE) between WM and Layer6 for the raw data (salmon) and SpotClean processed data (turquoise). By reducing noise due to spot-swapped UMIs, SpotClean improves fold changes and p-values for the majority of known DE genes.

Dataset	Prop. UMIs in Background	Number of background spots	Normalized prop. UMIs in Background	LPSS
LIBD_151507	6.4%	766	0.0084%	NA
LIBD_151508	4.2%	606	0.0069%	NA
LIBD_151669	8.3%	1329	0.0062%	NA
LIBD_151670	7.6%	1494	0.0051%	NA
LIBD_151673	8.6%	1353	0.0064%	NA
LIBD_151674	10.4%	1319	0.0079%	NA
mouse_brain	8.3%	2290	0.0036%	NA
mouse_kidney	7.7%	3554	0.0022%	NA
human_breast	4.3%	1005	0.0043%	NA
human_lymphnode	10.7%	957	0.0112%	NA
human_spinalcord	14.2%	2180	0.0065%	NA
HM-1	20.8%	2962	0.0070%	14.9%
HM-2	17.6%	3666	0.0048%	10.2%
HM-3	26.9%	3092	0.0087%	13.3%

Table C.1: The proportion of UMI counts in background spots, the number of background spots, the per-spot proportion of UMI counts in background spots, and the LPSS (which is only defined for the three chimeric datasets). The proportion of UMI counts in background spots serves as an underestimate of the proportion of spot-swapped UMI counts (since the proportion quantifies tissue-to-background swapping but does not account for tissue-to-tissue swapping). We also report the normalized proportion, which is the proportion of UMI counts in background spots divided by the number of background spots in each dataset. LPSS is defined in the chimeric experiment as the proportion of misclassified reads in tissue spots (mouse reads in human spots and human reads in mouse spots). This is a lower bound as it does not account for spot swapping within species, and it does not count reads in background spots.

Dataset	No decontamination	SpotClean	SoupX	DecontX
LIBD_151507	30.720	**14.957**	NA	58.651
LIBD_151508	26.102	**12.909**	NA	122.332
LIBD_151669	21.610	**12.001**	NA	266.682
LIBD_151670	17.452	**10.221**	NA	154.391
LIBD_151673	21.472	**12.172**	NA	74.319
LIBD_151674	25.131	**13.469**	NA	57.744
mouse_brain	24.161	**9.625**	824.134	284.147
mouse_kidney	12.165	**7.903**	319.903	121.810
human_breast	13.790	**9.987**	118.043	71.458
human_lymphnode	108.288	**31.581**	464.735	196.503
human_spinalcord	122.037	**14.431**	181.217	515.027

Table C.2: Average mean squared error (MSE) between true and decontaminated gene expression (average taken over 3000 genes) in 11 SimII datasets simulated using input from the dataset indicated. NA denotes datasets for which the corresponding method failed to run. The lowest MSE for each dataset is bolded.

Dataset	No decontamination	SpotClean	SoupX	DecontX
LIBD_151507	32.472	**14.998**	NA	87.570
LIBD_151508	27.371	**13.248**	NA	196.679
LIBD_151669	22.892	**11.710**	NA	81.707
LIBD_151670	18.538	**10.255**	62.989	63.855
LIBD_151673	23.502	**11.719**	NA	184.707
LIBD_151674	27.832	**13.372**	NA	61.567
mouse_brain	26.856	**9.508**	685.702	284.959
mouse_kidney	12.989	**7.912**	302.584	119.542
human_breast	15.222	**9.953**	135.952	74.705
human_lymphnode	120.495	**28.026**	534.534	195.524
human_spinalcord	133.396	**13.414**	186.904	563.552

Table C.3: Average mean squared error (MSE) between true and decontaminated gene expression (average taken over 3000 genes) in 11 SimIII datasets simulated using input from the dataset indicated. NA denotes datasets for which the corresponding method failed to run. The lowest MSE for each dataset is bolded.

Dataset	No decontamination	SpotClean	SoupX	DecontX
LIBD_151507	36.056	**15.388**	NA	144.595
LIBD_151508	30.580	**13.639**	NA	141.042
LIBD_151669	25.767	**12.677**	NA	79.926
LIBD_151670	21.365	**10.553**	NA	196.039
LIBD_151673	27.271	**12.769**	97.401	339.906
LIBD_151674	32.751	**13.967**	NA	67.959
mouse_brain	31.022	**9.419**	829.479	458.233
mouse_kidney	14.756	**7.771**	331.323	131.421
human_breast	16.731	**9.348**	136.516	76.562
human_lymphnode	132.168	**29.893**	523.708	209.594
human_spinalcord	152.092	**13.154**	223.067	215.555

Table C.4: Average mean squared error (MSE) between true and decontaminated gene expression (average taken over 3000 genes) in 11 SimIV datasets simulated using input from the dataset indicated. NA denotes datasets for which the corresponding method failed to run. The lowest MSE for each dataset is bolded.

Dataset	Link
human_brain	https://github.com/LieberInstitute/spatialLIBD
mouse_brain	https://support.10xgenomics.com/spatial-gene-expression/datasets/1.1.0/V1_Adult_Mouse_Brain
mouse_kidney	https://support.10xgenomics.com/spatial-gene-expression/datasets/1.1.0/V1_Mouse_Kidney
human_breast	https://support.10xgenomics.com/spatial-gene-expression/datasets/1.1.0/V1_Breast_Cancer_Block_A_Section_2
human_breast_2	https://support.10xgenomics.com/spatial-gene-expression/datasets/1.3.0/Visium_FFPE_Human_Breast_Cancer
human_lymphnode	https://support.10xgenomics.com/spatial-gene-expression/datasets/1.1.0/V1_Human_Lymph_Node
human_spinalcord	https://support.10xgenomics.com/spatial-gene-expression/datasets/1.2.0/Targeted_Visium_Human_SpinalCord_Neuroscience
human_colorectal	https://www.10xgenomics.com/resources/datasets/human-colorectal-cancer-whole-transcriptome-analysis-1-standard-1-2-0
human_pancreatic	https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE111672
Slide-seqV2	https://singlecell.broadinstitute.org/single_cell/study/SCP815/highly-sensitive-spatial-transcriptomics-at-near-cellular-resolution-with-slide-seqv2#study-download
GSE169749	https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE169749
GSE178361	https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE178361
GSE188888	https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE188888
GSE190595	https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE190595
GSE193460	https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE193460

Table C.5: Links to 30 publicly available spatial transcriptomics datasets used in chapter 4 (27 from the 10x Visium, 1 from the SpatialTranscriptomics, and 2 from the Slide-seqV2 protocol).

Gene	-log10_pval	logfc	t	CI	df
ENC1	21.6772	-0.59211	-10.0715	-0.77218,-0.52024	716.6266
TUBB2A	26.82923	-0.5005	-11.4299	-0.87942,-0.62153	611.6922
MBP	68.23467	0.43074	20.62996	1.31259,1.58892	502.7044
MAP1B	24.09136	-0.4268	-10.7987	-0.90317,-0.62517	558.5923
SNAP25	38.24431	-0.42293	-14.1973	-1.12206,-0.84929	526.9102
VSNL1	23.18776	-0.41835	-10.5489	-0.85222,-0.58468	578.5369
CCK	30.67843	-0.41324	-12.3489	-0.97287,-0.70589	605.6358
RTN1	21.56528	-0.41252	-10.0983	-0.77982,-0.5259	616.3649
YWHAH	22.66223	-0.40974	-10.4066	-0.85579,-0.58405	586.1957
OLFM1	21.10679	-0.4048	-9.99412	-0.8003,-0.53742	590.4734

Table C.6: Summary statistics of t-test results for known DE genes in LIBD_151507 raw data. -log10_pval: -log10 transformed p-value. logfc: log transformed fold change. t: t statistic. CI: 95% confidence interval. df: degrees of freedom.

Gene	-log10_pval	logfc	t	CI	df
ENC1	26.81212	-0.81797	-11.2768	-0.84106,-0.59168	828.6978
TUBB2A	36.07802	-0.73195	-13.3537	-1.07278,-0.79781	785.5271
MBP	47.97652	0.458097	16.37517	1.41258,1.79774	507.8979
MAP1B	30.97989	-0.64203	-12.3232	-1.15413,-0.83691	696.4673
SNAP25	64.89745	-0.85097	-19.0397	-1.70572,-1.38683	710.7584
VSNL1	34.64576	-0.70785	-13.0801	-1.12332,-0.83014	751.8196
CCK	53.54393	-0.84157	-16.7963	-1.49164,-1.17946	779.5883
RTN1	34.02306	-0.69379	-12.915	-1.0405,-0.76594	787.7664
YWHAH	32.53661	-0.6964	-12.6217	-1.1375,-0.83128	752.0688
OLFM1	34.35643	-0.71864	-13.0144	-1.08758,-0.80248	755.607

Table C.7: Summary statistics of t-test results for known DE genes in LIBD_151507 data decontaminated by SpotClean. -log10_pval: -log10 transformed p-value. logfc: log transformed fold change. t: t statistic. CI: 95% confidence interval. df: degrees of freedom.

C.2 Software versions for reproducibility

Below are the versions of language and packages at the time generating results in chapter 4: R-4.0.2; R/SpotClean-0.99.0; R/SoupX-1.5.0; R/celda-1.5.11; R/Seurat-3.2.2; R/scran-1.17.20; R/SPOTlight-0.1.7; R/reticulate-1.16; Python-3.7.4; Python/spatialde-1.1.3; FastQC-0.11.7; MultiQC-1.9; Space Ranger-1.2.2; Loupe Browser-4.2.0.

C.3 Data and code availability

Raw sequence data for the 3 human-mouse chimeric experiments are available at GEO (accession number: GSE178221). Links to 16 public spatial transcriptomics datasets are available in Table C.5. The human breast cancer single-cell RNA-seq data from Chung et al. (2017) is available at GEO (accession number: GSE75688). The human colorectal cancer single-cell RNA-seq data from Li et al. (2017) is available at GEO (accession number: GSE81861).

The R package *SpotClean* is available at `https://github.com/zijianni/SpotClean`. Codes for simulation and real data analyses as well as processed data can be found at `https://github.com/zijianni/codes_for_SpotClean_paper`.

REFERENCES

Alvarez, Marcus, Elior Rahmani, Brandon Jew, Kristina M Garske, Zong Miao, Jihane N Benhammou, Chun Jimmie Ye, Joseph R Pisegna, Kirsi H Pietiläinen, Eran Halperin, et al. 2020. Enhancing droplet-based single-nucleus rna-seq resolution using the semi-supervised machine learning classifier diem. *Scientific reports* 10(1): 1–16.

Anders, Simon, and Wolfgang Huber. 2010. Differential expression analysis for sequence count data. *Nature Precedings* 1–1.

Andrews, Simon, et al. 2010. Fastqc: a quality control tool for high throughput sequence data.

Asp, Michaela, Stefania Giacomello, Ludvig Larsson, Chenglin Wu, Daniel Fürth, Xiaoyan Qian, Eva Wärdell, Joaquin Custodio, Johan Reimegård, Fredrik Salmén, et al. 2019. A spatiotemporal organ-wide gene expression and cell atlas of the developing human heart. *Cell* 179(7):1647–1660.

Auer, Paul L, and RW Doerge. 2010. Statistical design and analysis of rna sequencing data. *Genetics* 185(2):405–416.

Bacher, Rhonda, Li-Fang Chu, Ning Leng, Audrey P Gasch, James A Thomson, Ron M Stewart, Michael Newton, and Christina Kendziorski. 2017. Scnorm: robust normalization of single-cell rna-seq data. *Nature methods* 14(6):584–586.

Bacher, Rhonda, and Christina Kendziorski. 2016. Design and computational analysis of single-cell rna-sequencing experiments. *Genome biology* 17(1):1–14.

Bacher, Rhonda, Ning Leng, Li-Fang Chu, Zijian Ni, James A Thomson, Christina Kendziorski, and Ron Stewart. 2018. Trendy: segmented regression analysis of expression dynamics in high-throughput ordered profiling experiments. *BMC bioinformatics* 19(1):1–10.

Benjamini, Yoav, and Yosef Hochberg. 1995. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)* 57(1):289–300.

Berglund, Emelie, Jonas Maaskola, Niklas Schultz, Stefanie Friedrich, Maja Marklund, Joseph Bergenstråhle, Firas Tarish, Anna Tanoglidi, Sanja Vickovic, Ludvig Larsson, et al. 2018. Spatial maps of prostate cancer transcriptomes reveal an unexplored landscape of heterogeneity. *Nature communications* 9(1):1–13.

Bernstein, Matthew N, Zijian Ni, Michael Collins, Mark E Burkard, Christina Kendziorski, and Ron Stewart. 2021. Charts: a web application for characterizing and comparing tumor subpopulations in publicly available single-cell rna-seq data sets. *BMC bioinformatics* 22(1):1–9.

Bernstein, Matthew N, Zijian Ni, Aman Prasad, Jared Brown, Chitrasen Mohanty, Ron Stewart, Michael A Newton, and Christina Kendziorski. 2022. Spatialcorr: Identifying gene sets with spatially varying correlation structure. *bioRxiv*.

Botía, Juan A, Jana Vandrovcova, Paola Forabosco, Sebastian Guelfi, Karishma D'Sa, John Hardy, Cathryn M Lewis, Mina Ryten, and Michael E Weale. 2017. An additional k-means clustering step improves the biological features of wgcna gene co-expression networks. *BMC systems biology* 11(1):1–16.

van den Brink, Susanne C, Anna Alemany, Vincent van Batenburg, Naomi Moris, Marloes Blotenburg, Judith Vivié, Peter Baillie-Johnson, Jennifer Nichols, Katharina F Sonnen, Alfonso Martinez Arias, et al. 2020. Single-cell and spatial transcriptomics reveal somitogenesis in gastruloids. *Nature* 582(7812):405–409.

Brown, Jared, Zijian Ni, Chitrasen Mohanty, Rhonda Bacher, and Christina Kendziorski. 2021. Normalization by distributional resampling of high throughput single-cell rna-sequencing data. *Bioinformatics* 37(22):4123–4128.

Browne, BC, N O'Brien, MJ Duffy, J Crown, and N O'Donovan. 2009. Her-2 signaling and inhibition in breast cancer. *Current cancer drug targets* 9(3):419–438.

Buettner, Florian, Kedar N Natarajan, F Paolo Casale, Valentina Proserpio, Antonio Scialdone, Fabian J Theis, Sarah A Teichmann, John C Marioni, and Oliver Stegle. 2015. Computational analysis of cell-to-cell heterogeneity in single-cell rna-sequencing data reveals hidden subpopulations of cells. *Nature biotechnology* 33(2):155–160.

Byrd, Richard H, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. 1995. A limited memory algorithm for bound constrained optimization. *SIAM Journal on scientific computing* 16(5):1190–1208.

Chu, Li-Fang, Daniel Mamott, Zijian Ni, Rhonda Bacher, Cathy Liu, Scott Swanson, Christina Kendziorski, Ron Stewart, and James A Thomson. 2019. An in vitro human segmentation clock model derived from embryonic stem cells. *Cell reports* 28(9):2247–2255.

Chung, Moo K. 2020. Gaussian kernel smoothing. *arXiv preprint arXiv:2007.09539*.

Chung, Woosung, Hye Hyeon Eum, Hae-Ock Lee, Kyung-Min Lee, Han-Byoel Lee, Kyu-Tae Kim, Han Suk Ryu, Sangmin Kim, Jeong Eon Lee, Yeon Hee Park, et al. 2017. Single-cell rna-seq enables comprehensive tumour and immune cell profiling in primary breast cancer. *Nature communications* 8(1):1–12.

Conesa, Ana, Pedro Madrigal, Sonia Tarazona, David Gomez-Cabrero, Alejandra Cervera, Andrew McPherson, Michał Wojciech Szcześniak, Daniel J Gaffney, Laura L Elo, Xuegong Zhang, et al. 2016. A survey of best practices for rna-seq data analysis. *Genome biology* 17(1):1–19.

Costello, Maura, Mark Fleharty, Justin Abreu, Yossi Farjoun, Steven Ferriera, Laurie Holmes, Brian Granger, Lisa Green, Tom Howd, Tamara Mason, et al. 2018. Characterization and remediation of sample index swaps by non-redundant dual indexing on massively parallel sequencing platforms. *BMC genomics* 19(1):1–10.

Darmanis, Spyros, Steven A Sloan, Ye Zhang, Martin Enge, Christine Caneda, Lawrence M Shuer, Melanie G Hayden Gephart, Ben A Barres, and Stephen R Quake. 2015. A survey of human brain transcriptome diversity at the single cell level. *Proceedings of the National Academy of Sciences* 112(23):7285–7290.

Dempster, Arthur P, Nan M Laird, and Donald B Rubin. 1977. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)* 39(1):1–22.

Deng, Qiaolin, Daniel Ramsköld, Björn Reinius, and Rickard Sandberg. 2014. Single-cell rna-seq reveals dynamic, random monoallelic gene expression in mammalian cells. *Science* 343(6167):193–196.

Dyer, Charissa A, Ady Kendler, Danielle Jean-Guillaume, Raj Awatramani, Albert Lee, Lisa M Mason, and John Kamholz. 2000. Gfap-positive and myelin marker-positive glia in normal and pathologic environments. *Journal of neuroscience research* 60(3):412–426.

Elosua-Bayes, Marc, Paula Nieto, Elisabetta Mereu, Ivo Gut, and Holger Heyn. 2021. Spotlight: seeded nmf regression to deconvolute spatial transcriptomics spots with single-cell transcriptomes. *Nucleic acids research* 49(9):e50–e50.

Ewels, Philip, Måns Magnusson, Sverker Lundin, and Max Käller. 2016. Multiqc: summarize analysis results for multiple tools and samples in a single report. *Bioinformatics* 32(19):3047–3048.

Finak, Greg, Andrew McDavid, Masanao Yajima, Jingyuan Deng, Vivian Gersuk, Alex K Shalek, Chloe K Slichter, Hannah W Miller, M Juliana McElrath, Martin Prlic, et al. 2015. Mast: a flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell rna sequencing data. *Genome biology* 16(1):1–13.

Gale, William A, and Geoffrey Sampson. 1995. Good-turing frequency estimation without tears. *Journal of quantitative linguistics* 2(3):217–237.

Gentleman, Robert C, Vincent J Carey, Douglas M Bates, Ben Bolstad, Marcel Dettling, Sandrine Dudoit, Byron Ellis, Laurent Gautier, Yongchao Ge, Jeff Gentry, et al. 2004. Bioconductor: open software development for computational biology and bioinformatics. *Genome biology* 5(10):1–16.

Griffiths, Jonathan A, Arianne C Richard, Karsten Bach, Aaron TL Lun, and John C Marioni. 2018. Detection and removal of barcode swapping in single-cell rna-seq data. *Nature communications* 9(1):1–6.

He, Bryan, Ludvig Bergenstråhle, Linnea Stenbeck, Abubakar Abid, Alma Andersson, Åke Borg, Jonas Maaskola, Joakim Lundeberg, and James Zou. 2020. Integrating spatial gene expression and breast tumour morphology via deep learning. *Nature biomedical engineering* 4(8):827–834.

Heiser, Cody N, Victoria M Wang, Bob Chen, Jacob J Hughey, and Ken S Lau. 2021. Automated quality control and cell identification of droplet-based single-cell data using dropkick. *Genome research* 31(10):1742–1752.

Islam, Saiful, Una Kjällquist, Annalena Moliner, Pawel Zajac, Jian-Bing Fan, Peter Lönnerberg, and Sten Linnarsson. 2011. Characterization of the single-cell transcriptional landscape by highly multiplex rna-seq. *Genome research* 21(7): 1160–1167.

Islam, Saiful, Amit Zeisel, Simon Joost, Gioele La Manno, Pawel Zajac, Maria Kasper, Peter Lönnerberg, and Sten Linnarsson. 2014. Quantitative single-cell rna-seq with unique molecular identifiers. *Nature methods* 11(2):163–166.

Jaitin, Diego Adhemar, Ephraim Kenigsberg, Hadas Keren-Shaul, Naama Elefant, Franziska Paul, Irina Zaretsky, Alexander Mildner, Nadav Cohen, Steffen Jung, Amos Tanay, et al. 2014. Massively parallel single-cell rna-seq for marker-free decomposition of tissues into cell types. *Science* 343(6172):776–779.

Kim, Tae Hyun, Xiang Zhou, and Mengjie Chen. 2020. Demystifying "drop-outs" in single-cell umi data. *Genome biology* 21(1):1–19.

Kircher, Martin, Susanna Sawyer, and Matthias Meyer. 2012. Double indexing overcomes inaccuracies in multiplex sequencing on the illumina platform. *Nucleic acids research* 40(1):e3–e3.

Kiselev, Vladimir Yu, Kristina Kirschner, Michael T Schaub, Tallulah Andrews, Andrew Yiu, Tamir Chandra, Kedar N Natarajan, Wolf Reik, Mauricio Barahona, Anthony R Green, et al. 2017. Sc3: consensus clustering of single-cell rna-seq data. *Nature methods* 14(5):483–486.

Klein, Allon M, Linas Mazutis, Ilke Akartuna, Naren Tallapragada, Adrian Veres, Victor Li, Leonid Peshkin, David A Weitz, and Marc W Kirschner. 2015. Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. *Cell* 161(5):1187–1201.

Korthauer, Keegan D, Li-Fang Chu, Michael A Newton, Yuan Li, James Thomson, Ron Stewart, and Christina Kendziorski. 2016. A statistical approach for identifying differential distributions in single-cell rna-seq experiments. *Genome biology* 17(1): 1–15.

Kukurba, Kimberly R, and Stephen B Montgomery. 2015. Rna sequencing and analysis. *Cold Spring Harbor Protocols* 2015(11):pdb–top084970.

Larsson, Anton JM, Geoff Stanley, Rahul Sinha, Irving L Weissman, and Rickard Sandberg. 2018. Computational correction of index switching in multiplexed sequencing libraries. *Nature Methods* 15(5):305–307.

Lee, Jeong Seok, Seongwan Park, Hye Won Jeong, Jin Young Ahn, Seong Jin Choi, Hoyoung Lee, Baekgyu Choi, Su Kyung Nam, Moa Sa, Ji-Soo Kwon, et al. 2020. Immunophenotyping of covid-19 and influenza highlights the role of type i interferons in development of severe covid-19. *Science immunology* 5(49):eabd1554.

Leng, Ning, John A Dawson, James A Thomson, Victor Ruotti, Anna I Rissman, Bart MG Smits, Jill D Haag, Michael N Gould, Ron M Stewart, and Christina Kendziorski. 2013. Ebseq: an empirical bayes hierarchical model for inference in rna-seq experiments. *Bioinformatics* 29(8):1035–1043.

Li, Huipeng, 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, et al. 2017. Reference component analysis of single-cell transcriptomes elucidates cellular heterogeneity in human colorectal tumors. *Nature genetics* 49(5):708–718.

Love, Michael I, Wolfgang Huber, and Simon Anders. 2014. Moderated estimation of fold change and dispersion for rna-seq data with deseq2. *Genome biology* 15(12): 1–21.

Lun, Aaron TL, Karsten Bach, and John C Marioni. 2016. Pooling across cells to normalize single-cell rna sequencing data with many zero counts. *Genome biology* 17(1):1–14.

Lun, Aaron TL, Samantha Riesenfeld, Tallulah Andrews, Tomas Gomes, John C Marioni, et al. 2019. Emptydrops: distinguishing cells from empty droplets in droplet-based single-cell rna sequencing data. *Genome biology* 20(1):1–9.

Macosko, Evan Z, Anindita Basu, Rahul Satija, James Nemesh, Karthik Shekhar, Melissa Goldman, Itay Tirosh, Allison R Bialas, Nolan Kamitaki, Emily M Martersteck, et al. 2015. Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. *Cell* 161(5):1202–1214.

Malik, Laraib, Fatemeh Almodaresi, and Rob Patro. 2018. Grouper: graph-based clustering and annotation for improved de novo transcriptome analysis. *Bioinformatics* 34(19):3265–3272.

Mannik, Jaana, Pille Vaas, Kristiina Rull, Pille Teesalu, Tiina Rebane, and Maris Laan. 2010. Differential expression profile of growth hormone/chorionic somatomammotropin genes in placenta of small-and large-for-gestational-age newborns. *The Journal of Clinical Endocrinology & Metabolism* 95(5):2433–2442.

Mathys, Hansruedi, Jose Davila-Velderrain, Zhuyu Peng, Fan Gao, Shahin Mohammadi, Jennie Z Young, Madhvi Menon, Liang He, Fatema Abdurrob, Xueqiao Jiang, et al. 2019. Single-cell transcriptomic analysis of alzheimer's disease. *Nature* 570(7761):332–337.

Maynard, Kristen R, Leonardo Collado-Torres, Lukas M Weber, Cedric Uytingco, Brianna K Barry, Stephen R Williams, Joseph L Catallini, Matthew N Tran, Zachary Besich, Madhavi Tippani, et al. 2021. Transcriptome-scale spatial gene expression in the human dorsolateral prefrontal cortex. *Nature neuroscience* 24(3):425–436.

McGinnis, Christopher S, Lyndsay M Murrow, and Zev J Gartner. 2019. Doubletfinder: doublet detection in single-cell rna sequencing data using artificial nearest neighbors. *Cell systems* 8(4):329–337.

Mieth, Bettina, Marius Kloft, Juan Antonio Rodríguez, Sören Sonnenburg, Robin Vobruba, Carlos Morcillo-Suárez, Xavier Farré, Urko M Marigorta, Ernst Fehr, Thorsten Dickhaus, et al. 2016. Combining multiple hypothesis testing with machine learning increases the statistical power of genome-wide association studies. *Scientific reports* 6(1):1–14.

Moncada, Reuben, Dalia Barkley, Florian Wagner, Marta Chiodin, Joseph C Devlin, Maayan Baron, Cristina H Hajdu, Diane M Simeone, and Itai Yanai. 2020. Integrating microarray-based spatial transcriptomics and single-cell rna-seq reveals tissue architecture in pancreatic ductal adenocarcinomas. *Nature biotechnology* 38(3): 333–342.

Ni, Zijian, Shuyang Chen, Jared Brown, and Christina Kendziorski. 2020. Cb2 improves power of cell detection in droplet-based single-cell rna sequencing data. *Genome biology* 21(1):1–10.

Ni, Zijian, Aman Prasad, Shuyang Chen, Richard B Halberg, Lisa Arkin, Beth Drolet, Michael Newton, and Christina Kendziorski. 2021. Spotclean adjusts for spot swapping in spatial transcriptomics data. *bioRxiv*.

Nimkulrat, Sutichot D, Matthew N Bernstein, Zijian Ni, Jared Brown, Christina Kendziorski, and Barak Blum. 2021. The anna karenina model of β-cell maturation in development and their dedifferentiation in type 1 and type 2 diabetes. *Diabetes* 70(9):2058–2066.

Oh, Do-Youn, and Yung-Jue Bang. 2020. Her2-targeted therapies—a role beyond breast cancer. *Nature Reviews Clinical Oncology* 17(1):33–48.

Patel, Anoop P, Itay Tirosh, John J Trombetta, Alex K Shalek, Shawn M Gillespie, Hiroaki Wakimoto, Daniel P Cahill, Brian V Nahed, William T Curry, Robert L Martuza, et al. 2014. Single-cell rna-seq highlights intratumoral heterogeneity in primary glioblastoma. *Science* 344(6190):1396–1401.

Raja, Kalpana, John Steill, Ian Ross, Lam C Tsoi, Finn Kuusisto, Zijian Ni, Miron Livny, James Thomson, and Ron Stewart. 2020. Skim-a generalized literature-based discovery system for uncovering novel biomedical knowledge from pubmed. *bioRxiv*.

Rodriques, Samuel G, Robert R Stickels, Aleksandrina Goeva, Carly A Martin, Evan Murray, Charles R Vanderburg, Joshua Welch, Linlin M Chen, Fei Chen, and Evan Z Macosko. 2019. Slide-seq: A scalable technology for measuring genome-wide expression at high spatial resolution. *Science* 363(6434):1463–1467.

Satija, Rahul, Jeffrey A Farrell, David Gennert, Alexander F Schier, and Aviv Regev. 2015. Spatial reconstruction of single-cell gene expression data. *Nature biotechnology* 33(5):495–502.

Shannon, Claude Elwood. 1948. A mathematical theory of communication. *The Bell system technical journal* 27(3):379–423.

Ståhl, Patrik L, Fredrik Salmén, Sanja Vickovic, Anna Lundmark, José Fernández Navarro, Jens Magnusson, Stefania Giacomello, Michaela Asp, Jakub O Westholm, Mikael Huss, et al. 2016. Visualization and analysis of gene expression in tissue sections by spatial transcriptomics. *Science* 353(6294):78–82.

Stankiewicz, Adrian M, Joanna Goscik, Artur H Swiergiel, Alicja Majewska, Marek Wieczorek, Grzegorz R Juszczak, and Paweł Lisowski. 2014. Social stress increases expression of hemoglobin genes in mouse prefrontal cortex. *BMC neuroscience* 15(1):1–16.

Stephenson, Emily, Gary Reynolds, Rachel A Botting, Fernando J Calero-Nieto, Michael D Morgan, Zewen Kelvin Tuong, Karsten Bach, Waradon Sungnak, Kaylee B Worlock, Masahiro Yoshida, et al. 2021. Single-cell multi-omics analysis of the immune response in covid-19. *Nature medicine* 27(5):904–916.

Stewart, Benjamin J, and Menna R Clatworthy. 2020. Applying single-cell technologies to clinical pathology: progress in nephropathology. *The Journal of pathology* 250(5):693–704.

Stickels, Robert R, Evan Murray, Pawan Kumar, Jilong Li, Jamie L Marshall, Daniela J Di Bella, Paola Arlotta, Evan Z Macosko, and Fei Chen. 2021. Highly sensitive spatial transcriptomics at near-cellular resolution with slide-seqv2. *Nature biotechnology* 39(3):313–319.

Street, Kelly, Davide Risso, Russell B Fletcher, Diya Das, John Ngai, Nir Yosef, Elizabeth Purdom, and Sandrine Dudoit. 2018. Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics. *BMC genomics* 19(1):1–16.

Subramanian, Aravind, Pablo Tamayo, Vamsi K Mootha, Sayan Mukherjee, Benjamin L Ebert, Michael A Gillette, Amanda Paulovich, Scott L Pomeroy, Todd R Golub, Eric S Lander, et al. 2005. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences* 102(43):15545–15550.

Sun, Shiquan, Jiaqiang Zhu, and Xiang Zhou. 2020. Statistical analysis of spatial expression patterns for spatially resolved transcriptomic studies. *Nature methods* 17(2):193–200.

Svensson, Valentine, Sarah A Teichmann, and Oliver Stegle. 2018. Spatialde: identification of spatially variable genes. *Nature methods* 15(5):343–346.

Tang, Fuchou, Catalin Barbacioru, Yangzhou Wang, Ellen Nordman, Clarence Lee, Nanlan Xu, Xiaohui Wang, John Bodeau, Brian B Tuch, Asim Siddiqui, et al. 2009. mrna-seq whole-transcriptome analysis of a single cell. *Nature methods* 6(5): 377–382.

Thrane, Kim, Hanna Eriksson, Jonas Maaskola, Johan Hansson, and Joakim Lundeberg. 2018. Spatially resolved transcriptomics enables dissection of genetic

heterogeneity in stage iii cutaneous malignant melanoma. *Cancer research* 78(20): 5970–5979.

Townes, F William, Stephanie C Hicks, Martin J Aryee, and Rafael A Irizarry. 2019. Feature selection and dimension reduction for single-cell rna-seq based on a multinomial model. *Genome biology* 20(1):1–16.

Trapnell, Cole, Davide Cacchiarelli, Jonna Grimsby, Prapti Pokharel, Shuqiang Li, Michael Morse, Niall J Lennon, Kenneth J Livak, Tarjei S Mikkelsen, and John L Rinn. 2014. The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nature biotechnology* 32(4):381–386.

Treutlein, Barbara, Doug G Brownfield, Angela R Wu, Norma F Neff, Gary L Mantalas, F Hernan Espinoza, Tushar J Desai, Mark A Krasnow, and Stephen R Quake. 2014. Reconstructing lineage hierarchies of the distal lung epithelium using single-cell rna-seq. *Nature* 509(7500):371–375.

Vanni, Silvia, Marco Zattoni, Fabio Moda, Giorgio Giaccone, Fabrizio Tagliavini, Stéphane Haïk, Jean-Philippe Deslys, Gianluigi Zanusso, James W Ironside, Margarita Carmona, et al. 2018. Hemoglobin mrna changes in the frontal cortex of patients with neurodegenerative diseases. *Frontiers in neuroscience* 12:8.

Wang, Zhong, Mark Gerstein, and Michael Snyder. 2009. Rna-seq: a revolutionary tool for transcriptomics. *Nature reviews genetics* 10(1):57–63.

Wolock, Samuel L, Romain Lopez, and Allon M Klein. 2019. Scrublet: computational identification of cell doublets in single-cell transcriptomic data. *Cell systems* 8(4):281–291.

Xue, Zhigang, Kevin Huang, Chaochao Cai, Lingbo Cai, Chun-yan Jiang, Yun Feng, Zhenshan Liu, Qiao Zeng, Liming Cheng, Yi E Sun, et al. 2013. Genetic programs in human and mouse early embryos revealed by single-cell rna sequencing. *Nature* 500(7464):593–597.

Yang, Shiyi, Sean E Corbett, Yusuke Koga, Zhe Wang, W Evan Johnson, Masanao Yajima, and Joshua D Campbell. 2020. Decontamination of ambient rna in single-cell rna-seq with decontx. *Genome biology* 21(1):1–15.

Young, Matthew D, and Sam Behjati. 2020. Soupx removes ambient rna contamination from droplet-based single-cell rna sequencing data. *Gigascience* 9(12): giaa151.

Zhang, Ji-Yuan, Xiang-Ming Wang, Xudong Xing, Zhe Xu, Chao Zhang, Jin-Wen Song, Xing Fan, Peng Xia, Jun-Liang Fu, Si-Yu Wang, et al. 2020. Single-cell landscape of immunological responses in patients with covid-19. *Nature immunology* 21(9):1107–1118.

Zhao, Edward, Matthew R Stone, Xing Ren, Jamie Guenthoer, Kimberly S Smythe, Thomas Pulliam, Stephen R Williams, Cedric R Uytingco, Sarah EB Taylor, Paul Nghiem, et al. 2021. Spatial transcriptomics at subspot resolution with bayesspace. *Nature biotechnology* 39(11):1375–1384.

Zheng, Grace XY, Jessica M Terry, Phillip Belgrader, Paul Ryvkin, Zachary W Bent, Ryan Wilson, Solongo B Ziraldo, Tobias D Wheeler, Geoff P McDermott, Junjie Zhu, et al. 2017. Massively parallel digital transcriptional profiling of single cells. *Nature communications* 8(1):1–12.