Summer 2023

Statistical Methods for Single Cell Sequencing Data Analysis

Fei Qin

STATISTICAL METHODS FOR SINGLE CELL SEQUENCING DATA ANALYSIS

by

Fei Qin

Bachelor of Medicine
Nanjing Medical University, 2015

Master of Science
Nanjing Medical University, 2018

Submitted in Partial Fulfillment of the Requirements

For the Degree of Doctor of Philosophy in

Biostatistics

The Norman J. Arnold School of Public Health

University of South Carolina

2023

Accepted by:

Feifei Xiao, Major Professor

James W Hardin, Committee Member

Bo Cai, Committee Member

Dongjun Chung, Committee Member

Guoshuai Cai, Committee Member

Ann Vail, Dean of the Graduate School

Acknowledgements

I would like to express my heartfelt appreciation to my supervisor, Dr. Feifei Xiao, for her invaluable mentorship and unwavering support during my PhD journal. Without her patient guidance and encouragement, this achievement would not have been possible. I am also deeply grateful to Dr. Guoshuai Cai. I feel fortunate to have had the chance to be advised by both Dr. Xiao and Dr. Cai, who have taught me to remain curious about different research topics and to be patient while finding answers to those questions. Furthermore, I would like to express my gratitude to the other members of my dissertation committee, Dr. James W Hardin, Dr. Bo Cai, and Dr. Dongjun Chung, for their valuable suggestions during the dissertation process.

I am also grateful to other PhD students in the Biostatistics department, including Dr. Xizhi Luo, Xuanxuan Yu, and Dayuan Wang. I have always enjoyed discussing research questions and sharing daily life with them.

Finally, I cannot express my gratitude enough to my parents and my soulmate, Ms. Karen Yong, who have supported me throughout my entire journal. Their unwavering trust in me has always kept my spirit and motivation high.

Abstract

The recent emergence of single cell sequencing (SCS) technology has provided us with single-cell DNA or RNA sequencing (scDNA/RNA-seq) information to investigate cellular evolutionary relationships. Despite many analysis methods have been developed to infer intra-tumor genetic heterogeneity, cluster cellular subclones, detect genetic mutations, and investigate spatially variable (SV) genes, exploring SCS data remains statistically challenging due to its noisy nature.

To identify subclones with scDNA-seq data, many existing studies use an independent statistical model to detect copy number profile in the first step, followed by classical clustering methods for subclone identification in downstream analyses. However, spurious results might be generated in this two-step clustering strategy due to the falsely identified copy number aberrations (CNAs) in the first copy number profiling step. Furthermore, although advances in spatial transcriptomics enable gene expression profiling with molecular resolution while preserving spatial information of the tissue, it is still challenging to identify spatially variable (SV) genes by modeling transcriptomic data with hundreds of spatial locations.

To address these issues, we developed two methods. First, we developed a subclone clustering method based on a fused lasso model, referred to as FLCNA, which can simultaneously detect CNAs with scDNA-seq data. Extensive simulations and a real data application have demonstrated the desirable performance of FLCNA to cluster subclones

and estimate copy number profiles in scDNA-seq data. Second, we developed SPADE, a spatial pattern and differential expression analysis method, to accurately identify SV genes within or between groups in spatial transcriptomic data. To facilitate the application of these two methods, an R package has been developed for each method, respectively.

Our investigation into the analysis of SCS data is expected to help investigators gain deep insights into various single cell studies, ultimately improving the understanding of cellular evolution and designs of treatment approaches for various diseases.

Table of Contents

List of Figures

Chapter 1: Introduction

1.1 Single cell sequencing

In recent years, existing genotyping technologies have shifted from array comparative genomic hybridization (aCGH) and single nucleotide polymorphism (SNP) array to next-generation sequencing (NGS) and single-cell sequencing (SCS). SCS technology involves sequencing the genome or transcriptome of individual cells, providing single-cell DNA sequencing (scDNA-seq) and RNA sequencing (scRNA-seq) data to investigate cellular evolutionary relationship (Tang *et al.*, 2019). Empowered by these technologies, researchers have profiled many diseases and biological processes at the single-cell level, such as neuron genomic heterogeneity, carcinogenesis and tumor evolution, and early embryo development (Hu *et al.*, 2016).

Numerous methods have been developed specifically for the analysis of SCS data. For scDNA-seq data, methods are available to infer intra-tumor genetic heterogeneity (Ross and Markowetz, 2016), cluster cellular subclones (Yu *et al.*, 2022), detect doublets (Weber *et al.*, 2021), and identify novel somatic mutations (Wang *et al.*, 2020). For scRNA-seq data, over 1,000 tools have been provided for different analyses (https://www.scrna-tools.org), and R packages like *Seurat* (Satija *et al.*, 2015) have been developed to perform data filtering, normalization, scaling, dimension reduction, clustering, and visualization. Other methods like MAGIC (van Dijk *et al.*, 2018) can

impute scRNA-seq data, ZINB_WaVE (Risso *et al.*, 2018) can remove batch effects, Monocle3 (Trapnell *et al.*, 2014) can identify pseudotime trajectory, and DEsingle (Miao *et al.*, 2018) can perform differential expression analysis.

Despite these advancements, processing SCS data remains challenging due to various technical errors. scRNA-seq measurements often suffer from a large fraction of zeros, where a given gene in a cell has no unique molecular identifier or reads mapping to it (Lähnemann *et al.*, 2020). In contrast, scDNA-seq analysis is more complicated than scRNA-seq due to coverage nonuniformity and false-positive (FP) errors (Navin, 2014). Coverage nonuniformity is caused by significant GC bias during whole-genome amplification (WGA), which is used to amplify the genome of a single cell (Grün and van Oudenaarden, 2015). FP errors occur at random sites due to the infidelity of the WGA polymerase (Lasken, 2007).

1.2 Subclone clustering in single cell sequencing data

Cancer is a disease characterized by cellular growth and division. As a single cell evolves into a malignant mass of tumor cells, distinct subgroups of cells (subclones) form, leading to intra-tumor heterogeneity (ITH), which plays a significant role in treatment resistance (Dagogo-Jack and Shaw, 2018). For instance, treatment-resistant cell subclones were found to adaptively change after neoadjuvant chemotherapy in patients with breast cancer (Kim *et al.*, 2018). In head and neck squamous cell carcinoma, higher levels of ITH were associated with poor survival outcomes (Mroz and Rocco, 2013). Therefore, accurately assessing ITH and identifying tumor subclones is essential to understand tumor progression and resistance to therapy (Dagogo-Jack and Shaw, 2018).

To asses ITH and identify subclones, various technologies such as RNA sequencing (Davis-Marcisak *et al.*, 2019), DNA sequencing (Zhang *et al.*, 2014), and SNP array (Zucker *et al.*, 2019) have emerged. Studies using data generated by these technologies have found that most tumors contain more than one major subclones, and the level of subclonal diversity (the number of major subclones) is associated with the tumor subtypes (Baslan *et al.*, 2020; Miles *et al.*, 2020; Morita *et al.*, 2020). For instance, there is greater subclonal diversity in estrogen receptor-negative than in estrogen receptor-positive breast cancers (Baslan *et al.*, 2020). However, these studies are limited to only reporting an average signal from a complex population of cells, and fail to identify cellular subclones (Navin, 2015). The advances in scDNA-seq data have enabled the assessment of tumor heterogeneity without the confounding effect of mixed cells (Navin *et al.*, 2011; Wang *et al.*, 2014). Moreover, subclonal populations from the same tumor tissue can be identified using scDNA-seq data to infer the tumor evolutionary history, which can promote the development of targeted therapies (Jiang *et al.*, 2016).

Numerous methods have been developed specifically for detecting subclones from scDNA-seq data using different variants, including single nucleotide variants (SNVs) (Yu *et al.*, 2022) or copy number aberrations (CNAs) (Chen *et al.*, 2020). The SCClone (Yu *et al.*, 2022) method clusters subclones from SNVs data by leveraging a probability mixture model for binary data and infer subclones via an expectation-maximization algorithm. SiCloneFit (Zafar *et al.*, 2019) proposes using a nonparametric Bayesian method to reconstruct clonal populations and display the evolutionary relationship between the clones using SNVs data. RobustClone (Chen *et al.*, 2020), designed for both single cell SNVs and CNAs data, recovers the true genotypes of subclones based on an extended

robust principal component analysis model. Despite these advances, technical issues such as allele dropout, FP errors, and doublets make scDNA-seq data incomplete and error-prone, posing significant challenges in accurately inferring clonal heterogeneity.

1.3 Copy number aberrations for subclone clustering

To identify subclones (Cariati *et al.*, 2019), chromosomal CNAs have been commonly used as most cells from the same subclone are found to share genetic variants (Wang *et al.*, 2020). CNAs refer to deletions or duplications of the DNA segments compared to a reference genome, and such variation has been found to be a significant risk factor of multiple human diseases, such as neurodevelopmental diseases (Asadollahi *et al.*, 2014; Castellani *et al.*, 2014) and cancer (Al-Sukhni *et al.*, 2012; Liu *et al.*, 2012). Recently, many well-performed copy number profiling methods have been developed for aCGH, SNP array and NGS data, such as, HMMcopy (Shah *et al.*, 2006), PennCNV (Wang *et al.*, 2007), and EXCAVATOR2 (D'Aurizio *et al.*, 2016). However, these bulk genotyping technologies are unable to describe high throughput data at the single-cell level, which is crucial for understanding tumor evolution. The development of scDNA-seq allows researchers to characterize cellular CNAs and reduce ambiguity in elucidating cancer evolutionary history (Wang *et al.*, 2020). However, detecting copy number changes in scDNA-seq data remains challenging due to its shallow and uneven coverage of depth.

In essence, detection of CNAs requires identifying the breakpoints or boundaries of copy number regions from chromosomal segments, which is also known as segmentation. Most segmentation methods rely on either circular binary segmentation (CBS) or hidden Markov model (HMM) (Wang *et al.*, 2018; Garvin *et al.*, 2015). However, these methods

are designed only for single sample setting, without considering shared signals across samples, which is essential for investigating scDNA-seq data. Moreover, in scDNA-seq data, multiple cells from the same subclone may share the same breakpoints (Wang *et al.*, 2020). Thus, it is crucial to consider subclone information within a complex cell population when detecting CNAs, which ensures precise CNA detection and subclone clustering results in scDNA-sep data.

1.4 Spatial transcriptomics

Advances in SCS technologies have enabled the characterization and discovery of transcriptionally distinct cell types and cell states (Atta and Fan, 2021). However, current studies are based on dissociating cells from tissues, thereby losing potentially valuable spatial information that may inform how cell types and cell states are organized within tissues (Maniatis *et al.*, 2021). Such spatial information could ultimately impact tissue phenotypes and function (Maniatis *et al.*, 2021). Spatial transcriptomic technology now provides exceptional views of single cells while retaining information about spatial context. Incorporating spatial information into transcriptomics is essential for understanding neurobiology and tumor biology, as biological activities are closely tied to the specific organization of cells (Method of the Year 2020: spatially resolved transcriptomics., 2021).

Various computational methods have been developed for analyzing spatial transcriptomic data. To identify genes with significant spatial patterns, different methods have been designed based on Gaussian processes (Svensson *et al.*, 2018), generalized linear models (Sun *et al.*, 2020), and spatial autocorrelation analysis model (Miller *et al.*,

2021). Identifying spatially variable (SV) genes can provide insight into position-specific phenotypes (Atta and Fan, 2021) and help detect putative cell-cell communication networks (Almet *et al.*, 2021). Nevertheless, given the nascency of such spatial transcriptomic technology, only a few datasets are available, and unique challenges remain to be addressed. For example, in technologies with large pixel size, transcripts from multiple cells may be captured in each spatially resolved pixel, hindering the accurate identification of genes with specific spatial patterns (Atta and Fan, 2021). The SPARK (Sun *et al.*, 2020) method, designed specifically for spatial transcriptomic data, builds upon a generalized linear spatial model (GLSM) (Li *et al.*, 2009) and utilizes various spatial kernels to identify SV genes. However, fixed parameters are used in these spatial kernels for each gene, causing a loss of power in identifying genes with specific spatial patterns. Furthermore, all existing methods were developed to identify SV genes within groups, and none of them can identify SV genes between groups.

1.5 Gaps in the existing studies

There are still challenges to analyze SCS data. The first challenge is the accurate identification of subclones, which is crucial for understanding tumor progression and therapy resistance. Existing studies use genetic variants such as CNAs to cluster subclones. However, most studies usually utilize two separate steps to identify subclones (i.e., CNA detection, subclone clustering). This two-step protocol may generate many false-positive variants during the procedure of CNA detection, especially in the context of genome-wide studies testing thousands of variants. Subsequently, spurious information from this first step may diminish the accuracy of subclone identification from SCS data in

the second step. Theoretically, it is hard to design a single model which can handle both clustering and change points detection problems.

The second challenge is identifying genes with spatial expression patterns in spatial transcriptomic data, which is an essential first step for characterizing complex tissues with spatial transcriptomics (Sun *et al.*, 2020). Due to the recent emergence of such technology, limited methods (Edsgärd *et al.*, 2018; Svensson *et al.*, 2018; Sun *et al.*, 2020) are available for accurately identifying genes with spatial patterns. Existing methods, such as SpatialDE (Svensson *et al.*, 2018) and SPARK (Sun *et al.*, 2020), only utilize fixed length-scale hyperparameter values in the kernels among all genes for SV gene identification, which may lose power for finding SV genes with specific spatial patterns. Moreover, existing methods were only designed to detect SV genes within groups, all of which were not capable of identifying SV genes between groups with different treatment conditions or time phrases.

To address these challenges, we proposed two methods. First, we proposed a CNA detection method based on the fused lasso model, FLCNA, which can simultaneously identify subclones in SCS data (Chapter 2). Second, we proposed SPADE, an accurate Spatial Pattern And Differential Expression analysis method, based on a Gaussian process regression (GPR) model, to overcome the challenge of identifying SV genes in spatial transcriptomic data (Chapter 3). To facilitate the application of our proposed methods, we developed R software for each of our two methods (Chapter 4). Through extensive simulations and real data analyses, FLCNA was found to provide superior performance in subclone identification and CNA detection with scDNA-seq data. SPADE was also observed to be powerful to identify SV genes both within and between groups in

spatial transcriptomic data. These two methods are expected to help investigators obtain deep insights into various single cell studies such as CNA detection, subclone clustering, and spatial expression analysis. Additionally, they are anticipated to contribute to the understanding of diseases evolution and aid in the development of novel therapies.

Chapter 2: A statistical learning method for simultaneous copy number estimation and subclone clustering with single cell sequencing data

Abstract

The availability of single cell sequencing (SCS) enables us to assess intra-tumor heterogeneity and identify cellular subclones without the confounding effect of mixed cells. Copy number aberrations (CNAs) have been commonly used to identify subclones in SCS data using various clustering methods, since cells comprising a subpopulation are found to share genetic profile. However, currently available methods may generate spurious results (e.g., falsely identified CNAs) in the procedure of CNA detection, hence diminishing the accuracy of subclone identification from a large complex cell population. In this study, we developed a CNA detection method based on a fused lasso model, referred to as FLCNA, which can simultaneously identify subclones in single cell DNA sequencing (scDNA-seq) data. Spike-in simulations were conducted to evaluate the clustering and CNA detection performance of FLCNA benchmarking to existing copy number estimation methods (SCOPE, HMMcopy) in combination with the existing and commonly used clustering methods. Interestingly, application of FLCNA to a real scDNA-seq dataset of breast cancer revealed remarkably different genomic variation patterns in neoadjuvant chemotherapy (NAC) treated samples and pre-treated samples. We show that FLCNA is a practical and powerful method in subclone identification and CNA detection with scDNA-seq data.

2.1 Introduction

In cancer, a small population of cancer stem cells evolves into a malignant mass of tumor cells, which then diverges and forms distinct subclones, contributing to intra-tumor heterogeneity (ITH). The level of ITH is associated with tumor progression and is sensitive to clinical treatments (Stanta and Bonin, 2018). Intrinsic mechanistic processes, such as inherent genomic variation, clonal competition, and tumor-host interactions, contribute to ITH (Gerlinger *et al.*, 2012; Yachida *et al.*, 2010; Vogelstein *et al.*, 2013; Polyak, 2014). Therefore, accurate assessment of ITH and identification of subclones is essential to understand the mechanisms of tumor progression and resistance to therapy (Dagogo-Jack and Shaw, 2018).

Most existing studies (Zhang *et al.*, 2014; Deshwar *et al.*, 2015; Li and Li, 2014; Oesper *et al.*, 2013; Ha *et al.*, 2014) characterize clonal diversity using bulk DNA sequencing, which was limited in only reporting an average signal from a complex population of cells (Navin, 2015). The emerging single cell sequencing (SCS) technology enables the assessment of ITH on a single-cell basis (Navin *et al.*, 2011; Wang *et al.*, 2014), providing single cell DNA or RNA sequencing (scDNA/RNA-seq) information to reveal cellular evolutionary relationship (Tang *et al.*, 2019). Subclonal populations can be identified from the same tumor tissue using SCS data, allowing for the inference of tumor evolution and providing insights into the development of targeted therapy for different tumor types (Jiang *et al.*, 2016). We take breast cancer for illustration, which is the most common malignancy in adult women. Triple-negative breast cancer (TNBC) is an important class of breast cancer which constitutes 12-18% of breast cancer patients (Foulkes *et al.*, 2010), and many studies have shown that patients with TNBC harbor high

levels of somatic mutations (Gao *et al.*, 2016; Wang *et al.*, 2014), which partially result in extensive ITH. NAC is the standard therapy for TNBC patients who show low level of the estrogen, progesterone and HER2 receptors and are not eligible for hormone or HER2-targeted therapy (Foulkes *et al.*, 2010). Still, due to chemoresistance, poor overall survival performance is observed in about 50% TNBC patients (Foulkes *et al.*, 2010). Identifying subclones and detecting mutations in this subpopulation is critical to untangle the molecular mechanisms of chemoresistance in these patients.

Cells in a subpopulation share genetic variant characteristics (Cariati *et al.*, 2019), with chromosomal copy number aberration (CNA) being one of the most important genetic variants, which is the gain or loss of DNA segments. CNAs stimulate the stemness of other tumor cells to new cancer stem cells, ultimately giving rise to clonal evolution in cancers (Dai and Liu, 2021). Therefore, CNAs serve as good biomarkers to assess ITH and identify subclones. Basically, the detection of CNAs is to find the breakpoints or boundaries of copy number regions from chromosomal segments. Though great efforts have been made in CNA detection methodology for scDNA-seq data, it is challenging to detect CNAs in such data due to shallow and uneven depth of coverage (Mallory *et al.*, 2020). We use the existing methods applied in our study for brief illustration. SCOPE (Wang *et al.*, 2020) was recently developed for copy number estimation with whole genome scDNA-seq data which uses a Poisson latent factor model for normalization and an expectation-maximization (EM) algorithm to estimate copy number profile. HMMcopy was designed for array comparative genomic hybridization (aCGH) and next generation sequencing (NGS) data (Shah *et al.*, 2006; Ha *et al.*, 2012) using a Hidden Markov Model, and has also been extensively applied in SCS data (Laks

et al., 2018). Relevant to our proposed method in this study, Rojas & Wahlberg (Rojas and Wahlberg, 2014) used a fused lasso method and converted the problem into a convex optimization problem to detect chromosomal breakpoints. However, studies based on these methods used an independent statistical model to detect copy number profile in the first step, followed by classical clustering methods (e.g., Hierarchical (Bridges, 1966)) for subclone identification in downstream analyses. This two-step framework will generate spurious results due to the carry-over of noisy signals in the first copy number profiling step into the process of subclone clustering. We herein developed a method to achieve accurate copy number profiling and subclone clustering without copy number state calling as a middle step. It was the first time that the fused lasso model was explored in copy number profile based subclone clustering.

We herein developed FLCNA, a CNA detection method based on the fused lasso model, which simultaneously identifies subclones with scDNA-seq data. The FLCNA method was benchmarked against existing copy number profile detection methods (i.e., SCOPE (Wang *et al.*, 2020) and HMMcopy (Laks *et al.*, 2018)) coupled with classical Hierarchical (Bridges, 1966) and K-means (James and others, 1967) clustering methods. Spike-in simulations with pre-defined "true" subclones demonstrated the superior clustering performance of our method. Comparable performance for FLCNA in detecting copy number profile was also observed in simulation studies. Application of FLCNA to a breast cancer dataset successfully clustered subclones based on shared breakpoints of cells in three breast cancer patients. In conclusion, FLCNA was powerful in subclone clustering and CNAs detection for SCS data, especially for data with a large proportion of shared CNAs in a cluster.

2.2 Methods

FLCNA was developed based on a fused lasso model (Tibshirani *et al.*, 2005) to cluster subclones and simultaneously detect CNAs with scDNA-seq data. Note that though our framework was motivated by the detection of somatic copy number change in tumor tissues, it can also be naturally extended to copy number variation detection as a type of germline genetic variability. For consistency, our framework and evaluation will use the term CNA in describing and evaluating the methods across the manuscript.

2.2.1 Quality Control and pre-processing of scDNA-seq data

We used scDNA-seq read count data as input. In order to reduce artifacts for copy number detection, a quality control procedure was implemented to remove markers with extreme GC content (< 20% and > 80%) or those with low mappability (< 0.9) (Wang *et al.*, 2020). Then a two-step median normalization approach (Magi *et al.*, 2013) was used to further remove the effect of biases from the GC-content and mappability. Basically, the normalized read counts were defined as: $RC_i' = RC_i \frac{m}{m_o}$, where RC_i was the read counts for marker i, m_o was the median read counts of all markers with the same o value (where o=[the GC-content, mappability]) as the i-th marker, and m was the overall median read counts of all the markers. We further computed the ratio of the normalized read counts and its sample specific mean, the logarithm transformation of which (i.e., *log2R*) was obtained as the main signal intensities.

2.2.2 Notations and models

Considering we have *N* cells in total, let $\boldsymbol{X} = (x_{i,j})_{P\times N}$ be the normalized read counts data (i.e., *log2R*), where $x_{i,j}$ denotes the value of the i-th (i=1, …, P) marker from the j-th cell. $\boldsymbol{x_j} = \left(x_{1,j}, \ldots, x_{P,j}\right)^T$ is the data vector for the j-th sample. The samples sharing common biological characteristics (e.g., genomic variation) belong to a same cluster. Starting from a general K-cluster problem with a Gaussian mixture model (GMM), the observations $\boldsymbol{x_j}$ are assumed to be independent and are generated from a probability density function $g\left(\boldsymbol{x_j}\right) = \sum_{k=1}^{K} \pi_k f_k(\boldsymbol{x_j};\ \boldsymbol{\mu_k},\ \boldsymbol{\Sigma_k})$, where the "weights" $\pi_k{}'s$ ($\pi_k \geq 0$ for each cluster, $1 \leq k \leq K$ and $\sum_{k=1}^{K} \pi_k = 1$) are the mixing proportions. For the k-th cluster, $f_k(\boldsymbol{x_j};\ \boldsymbol{\mu_k}, \boldsymbol{\Sigma_k})$ denotes the Gaussian density function with mean vector $\boldsymbol{\mu_k} = (\mu_{1,\,k} \ldots \mu_{P,\,k})^T$ and covariance matrix $\boldsymbol{\Sigma_k}$. $\boldsymbol{\mu_k}$ denotes the mean genetic intensity of each marker in the k-th cluster, and $\boldsymbol{\Sigma_k}$ captures the correlations among the markers. In this study, we assume that the covariance matrix $\boldsymbol{\Sigma_k} = \boldsymbol{\Sigma} = \mathrm{diag}(\sigma_1^2,\ \sigma_2^2,\ \ldots \sigma_P^2)$ is fixed for all clusters. Parameters including π_k, $\boldsymbol{\mu_k}$ and $\boldsymbol{\Sigma}$ are unknown and need to be estimated.

The problem of copy number profile detection is equivalent to chromosomal breakpoints detection and has been initially explored in the fused lasso model (Rojas and Wahlberg, 2014). Fused lasso was extended from the classical lasso model and first designed to select variables and penalize the difference of successive features (Tibshirani *et al.*, 2005). Its ability in identifying and quantifying significant features is closely related to our problem of breakpoints detection on locating significant signals from a wide range of constant signals (Rojas and Wahlberg, 2014). Utilizing the fused lasso

penalty term for change points detection, the penalized log likelihood function in the FLCNA method is given by

$$Q = \sum_{j=1}^{N} \log\left[\sum_{k=1}^{K} \pi_k f_k(\boldsymbol{x}_j;\, \boldsymbol{\mu}_k, \boldsymbol{\Sigma})\right] - \lambda \sum_{k=1}^{K} \sum_{i=1}^{P-1} \tau_{i,\, i+1}^{(k)} \left|\mu_{i,k} - \mu_{i+1,k}\right|. \quad (1)$$

In the second term of Equation (1), a tunning hyperparameter λ and pre-defined adaptive weights (Zou, 2006) $\tau_{i,\, i+1}^{(k)}$ are utilized to shrink the absolute difference of the mean shift values $\left|\mu_{i,k} - \mu_{i+1,k}\right|$ in consecutive markers in the k-th cluster, ultimately disclosing change points. The tuning hyperparameter λ is used to control the overall number of change points that less change points tend to be generated with larger λ value (Figure 1). To shrink each pair of consecutive markers with the same weight, the tunning hyperparameter λ is fixed within each cluster. Such strategy may naturally decline the accuracy of penalization, ultimately affecting the clustering performance. To improve the accuracy, an adaptive penalization weight (Zou, 2006) $\tau_{i,\, i+1}^{(k)} = |\tilde{\mu}_{i,k} - \tilde{\mu}_{i+1,k}|^{-1}$ is applied, where $\tilde{\mu}_{i,k}$ is estimated from the same model without any penalization ($\lambda = 0$). This adaptive penalization weight is pre-defined to dynamically penalize each pair of successive markers. For example, if there is large difference between $\tilde{\mu}_{i,k}$ and $\tilde{\mu}_{i+1,k}$ in the model without penalization, a change point is expected to appear between the i-th and $(i + 1)$-th marker, and this change point tends to be informative for subclone clustering. In this case, according to $\tau_{i,\, i+1}^{(k)} = |\tilde{\mu}_{i,k} - \tilde{\mu}_{i+1,k}|^{-1}$, $\tau_{i,\, i+1}^{(k)}$ will be small, consequently the difference between $\mu_{i,k}$ and $\mu_{i+1,k}$ in Equation (1) will be lightly penalized, and will be more informative for the subclone clustering. Otherwise, the difference between $\mu_{i,k}$ and $\mu_{i+1,k}$ will be heavily penalized and will be less informative for the subclone clustering in

the FLCNA method. With the focus on change points detection and subclone identification, our goal is to maximize Equation (1) to estimate the parameter set $\boldsymbol{\Theta} = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}, \pi_k\}_{k=1}^{K}$.

2.2.3 Parameter estimation using expectation–maximization algorithm

In FLCNA, the parameter set $\boldsymbol{\Theta} = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}, \pi_k\}_{k=1}^{K}$ are estimated using expectation–maximization (EM) algorithm (Dempster *et al.*, 1977). We initialize $\boldsymbol{\Theta}$ with parameters estimated from the model without penalty (λ=0) and then we update these parameters by alternating between E- and M- steps in the EM algorithm. First, in the M-step, given the starting values of $\boldsymbol{\Theta}$, the probability for the j-th sample belonging to the k-th cluster is calculated by dividing the density of the k-th cluster by the sum of densities from all clusters. Thereafter, an E-step is used to update the estimated values of $\boldsymbol{\Theta}$. Specifically, the "weight" for the k-th cluster π_k and the variance for the i-th marker σ_i^2 are estimated by taking the first derivative of $Q(\boldsymbol{\Theta})$ w.r.t. π_k and σ_i^2, respectively. The estimates of the cluster means $\widehat{\boldsymbol{\mu}}_k$ are computed with a local quadratic approximation algorithm (Fan and Li, 2001). These updated $\boldsymbol{\Theta}$ estimation values were iteratively computed between E- and M- steps until convergence. We refer readers to Appendix A for a thorough description of this part of algorithm. After this step, all the parameters of our interest $\boldsymbol{\Theta} = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}, \pi_k\}_{k=1}^{K}$ are successfully estimated, based on which copy number profile will be assigned by the estimated cluster means (described below) and cells are clustered according to the estimated cluster weights.

2.2.4 Copy number profile identification and hyperparameters estimation

With the estimated cluster means ($\widehat{\boldsymbol{\mu}}_k$), we locate and quantify all the shared change points and identify copy number segments in each cluster. Based on these identified shared segments, we assign the most likely copy number state for each segment in each cell. A GMM-based clustering strategy (Xiao *et al.*, 2019) is implemented for CNA clustering using the normalized read counts data (i.e., *log2R*). Segments sharing similar intensity levels in a cell are identified as the ones with same copy number states. Each segment is classified using a five-state classification scheme with deletion of double copies (Del.d), deletion of a single copy (Del.s), normal/diploid, duplication of a single copy (Dup.s) and duplication of double copies (Dup.d). Besides, there are two hyperparameters to be pre-defined in the FLCNA method, including the number of clusters K and the tuning parameter λ. To find the optimal values of K and λ, we use a Bayesian information criterion (BIC) (Guo *et al.*, 2010) and the clustering model with smallest BIC value is selected as the optimal model.

2.2.5 Data description

We utilized two publicly available scDNA-seq datasets for illustration and evaluation of FLCNA in simulations and real data analyses. As described below, the BRCA5 dataset was used to mimic real data signals in simulations, and the TNBC dataset was analyzed for real data applications.

The BRCA5 dataset consists of 10,088 cells from a frozen breast tumor tissue that were sequenced using the 10× Genomics platform (https://www.10xgenomics.com), which utilizes microfluidic droplets to barcode cells and performs library construction.

We generated a read depth matrix of 28,760 markers and 10,088 cells from BAM files after binning with a 100kb bin size. To save computational time, we randomly selected 220 cells from the dataset and used the read counts from the entire genome to mimic real data in our simulations. More information on the simulation setting is available in the Spike-in simulations section, as described below.

The TNBC dataset consists of data from triple-negative breast cancer patients in the NCBI Sequence Read Archive (SRP114962) (Kim *et al.*, 2018). TNBC is characterized by extensive intratumor heterogeneity and frequently develops resistance to NAC treatment. Three patients (i.e., KTN126, KTN129, KTN302) were used for our analyses, where tumor cells were only reported in the pre-treatment samples. For each patient, cells were sequenced at two time points (pre- and mid/post-treatment) with 93 cells (46 pre- and 47 post-treatment) in the KTN126 patient, 90 cells (46 pre- and 44 post-treatment) in the KTN129 patient, and 92 cells (47 pre- and 45 mid-treatment) in the KTN302 patient, respectively. For these samples, FASTQ files were generated with Fastq-dump from SRA-Toolkit (Leinonen *et al.*, 2011), and then aligned to the NCBI hg19 reference genome and converted to BAM files. Raw read depth of coverage data were generated from the BAM files with a bin size 100kb (Wang *et al.*, 2020).

2.2.6 Spike-in simulations

We compared FLCNA to existing copy number profile detection methods, including SCOPE (Wang *et al.*, 2020) and HMMcopy (Laks *et al.*, 2018), using spike-in simulations. Since these two methods were only designed for the detection of CNAs without cell clustering, they were followed by two commonly used clustering methods,

Hierarchical (Bridges, 1966) and K-means (James and others, 1967). The simulation mimicked a scDNA-seq dataset of frozen breast tissue, the BRCA5 dataset (Section 2.2.5 Data description), by randomly selecting 220 cells from the dataset and using SCOPE and HMMcopy to remove genetic regions. Genetic regions with copy number changes detected by either method were excluded from the analysis, and the remaining sequences were treated as copy number-free sequences. Among these cells, 20 cells were randomly selected as reference cells for the SCOPE method, and signals of spiked-in CNAs were added to the remaining cells. To evaluate the robustness of FLCNA, we randomly generated CNAs of varied sizes (super short: 2~5 markers, short: 5~10 markers, medium: 10~20 markers, or long: 20~35 markers) and simulated different numbers of clusters (three or five). We evaluated varied copy number states, including Del.d, Del.s, Dup.s, and Dup.d, respectively. Moreover, data with a mixture of above four different copy number states were generated to mimic real-world data. Because a CNA may not be shared by all the cells in a cluster, different CNA sharing proportions (20%, 40%, 60%, 80%, 100%) were considered. For each cluster, we added 50 CNA segments to the background sequences. Besides, we evaluated the performance on scenarios with different numbers of CNAs among clusters by assigning random numbers of CNAs (20~80) to each cluster. For each of these scenarios, signals were spiked in by multiplying the background depth of coverage by $c/2$, where c is a normal random variable following $N(0.4, 0.1^2)$ for Del.d, $N(1.2, 0.1^2)$ for Del.s, $N(2.8, 0.1^2)$ for dup.s and $N(4.2, 0.1^2)$ for dup.d (Wang *et al.*, 2020). Adjusted Rand Index (ARI) (Vinh *et al.*, 2009) was calculated to evaluate the clustering performance of these methods by comparing the identified clusters of each method to the pre-defined "true" classes. ARI

gets close to 1 if clusters identified are completely consistent with the ground truth and close to 0 for random clustering. The performance of CNA detection for these methods was assessed using $F1$ score $\left(2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}\right)$ with precision rate defined as $\frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$ and recall rate defined as $\frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$.

2.2.7 FLCNA application to the TNBC dataset for breast cancer subcloning

With the dataset introduced previously, FLCNA was also applied to the TNBC dataset of breast cancer with three unrelated patients (i.e., KTN126, KTN129, KTN302) who have been treated with NAC. We identified shared CNAs using FLCNA and mapped them to 575 significant genes from the genome-wide association studies (GWAS) with breast cancer curated in the NHGRI-EBI GWAS Catalog (Sollis *et al.*, 2023). Pathway and network analyses were conducted for these genes; Gene Set Enrichment Analysis (GSEA) (Subramanian *et al.*, 2005) was conducted with enrichment of Kyoto Encyclopedia of Genes and Genomes (KEGG) (Kanehisa *et al.*, 2016). Further, the summary statistics from GSEA were used to generate connection networks for these three patients using the Enrichment Map implemented in Cytoscape (Shannon *et al.*, 2003).

2.3 Results

To capture the biological heterogeneity between potential subclones, we developed the FLCNA method based on a fused lasso model, which can simultaneously identify subclones and detect breakpoints in scDNA-seq data. The framework of the FLCNA method is summarized and illustrated in Figure 2.1. First, quality control and normalization procedures were used for pre-processing the datasets. Subclone clustering and breakpoints detection were achieved simultaneously using GMM combined with a

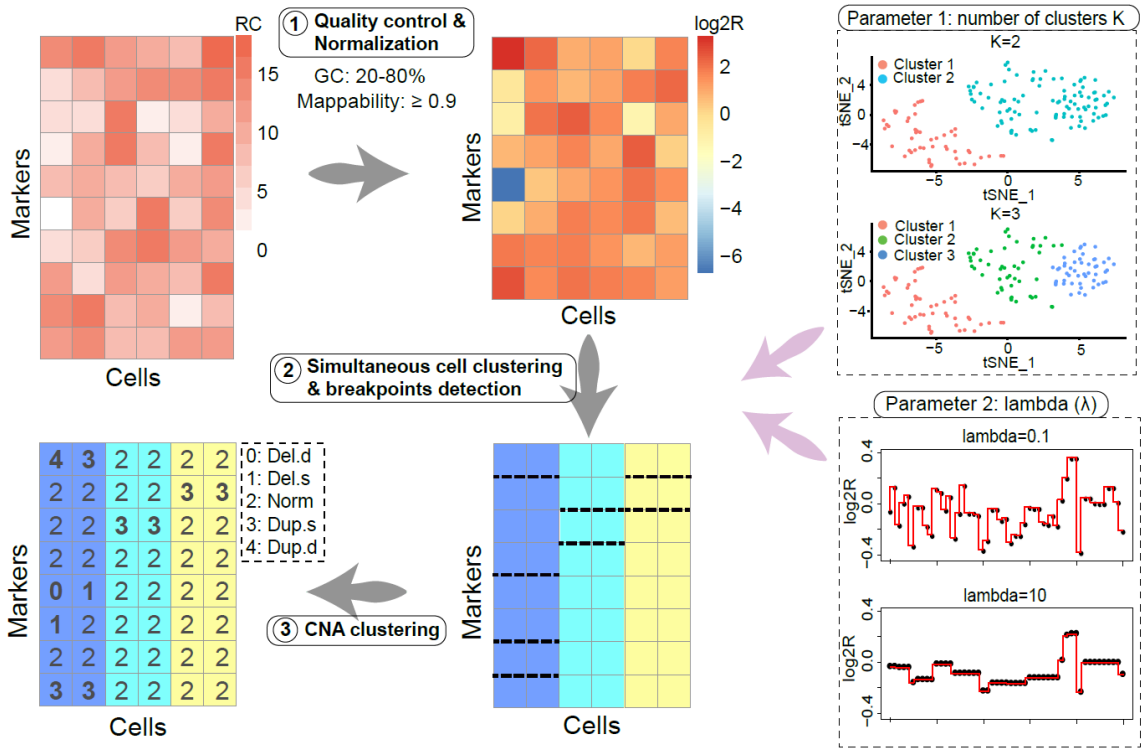

Figure 2.1 Analysis workflow of FLCNA. FLCNA first implemented a quality control procedure based on GC content and mappability. A two-step median normalization approach was then conducted to sequentially remove the effect of biases. The logarithm transformed ratio of normalized read counts and its sample specific mean (*log2R*) was then used in the main step of the FLCNA method. With *log2R*, we clustered subclones and simultaneously detected shared breakpoints using a Gaussian mixture model (GMM) after adding a fused lasso penalty term. Finally, based on these shared breakpoints in each cluster, segments for each cell were clustered into five different CNA states (Del.d, Del.s, Norm, Dup.s and Dup.d) using *log2R*. There are two hyperparameters in the FLCNA model including the number of cell clusters K and a tuning parameter λ, which controls the number of breakpoints. Del.d: Deletion of double copies; Del.s: Deletion of a single copy; Norm: Normal/diploid; Dup.s: Duplication of a single copy; Dup.d: Duplication of double copies.

fused lasso penalty term. Finally, based on these shared breakpoints in each cluster, candidate CNA segments for each cell were clustered into different CNA states using a GMM-based clustering strategy. Through extensive simulations, we evaluated the performance of FLCNA in clustering subclones and detecting CNAs. A real dataset of

breast cancer (i.e., TNBC) was also utilized to demonstrate the application of FLCNA in scDNA-seq data.

2.3.1 Evaluation of FLCNA via spike-in simulations

We first conducted spike-in simulations to assess the clustering performance of FLCNA, compared to two other copy number estimation methods (SCOPE and HMMcopy) coupled with different clustering methods (Hierarchical or K-means). Using a mixture of four different copy number states (Del.d, Del.s, Dup.s and Dup.d), we found that FLCNA outperformed the other methods in all scenarios with different numbers of clusters and CNAs in a cluster (Figure 2.2 and Figure A.1-A.2). Specifically, with five pre-defined clusters and a fixed number of CNAs (i.e., 50) in each cluster, FLCNA's clustering performance was incrementally improved as the CNA sharing proportion increased, and it generally presented better ARI than the other methods, especially with larger CNA sharing proportion (> 40%) (Figure 2.2). However, when the CNA sharing proportion decreased to 20%, almost all methods failed to provide desirable clustering performance (ARI < 0.50).

Moreover, the overall clustering performance of all methods was improved in scenarios with three pre-defined clusters (Figure A.1). For example, for super short CNAs shared by 60% of samples, ARI of FLCNA improved from 0.781 in the scenario of five pre-clusters (Figure 2.2) to 0.985 in that of three pre-clusters (Figure A.1). This improvement was likely due to the increased sample size in each cluster given a fixed total sample size. Comparing to the scenario with a fixed number of shared CNAs in each cluster, it was more challenging for all methods to identify subclones when varied numbers of shared CNAs presented, due to fewer shared CNAs generated in some

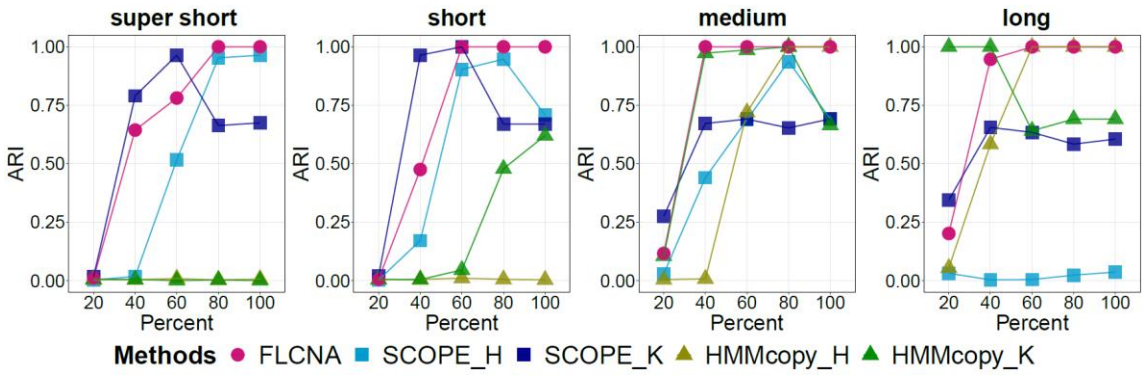

Figure 2.2 Assessment of FLCNA using simulation data with five clusters and mixed CNA states. Clustering results from FLCNA were compared to existing methods (i.e., SCOPE and HMMcopy) coupled with different clustering methods. For each of five clusters, we added signals of 50 CNA segments to the background signals with varied lengths (super short: 2~5 markers, short: 5~10 markers, medium: 10~20 markers, and long: 20~35 markers) and varied CNA proportions (20%, 40%, 60%, 80%, 100%), respectively. Signals of mixed CNA states (i.e., Del.d, Del.s, Norm, Dup.s and Dup.d) were spiked in. ARI: Adjusted Rand Index; SCOPE_H: SCOPE_Hierarchical; SCOPE_K: SCOPE_K-means; HMMcopy_H: HMMcopy_Hierarchical; HMMcopy_K: HMMcopy_K-means.

clusters by randomness (Figure A.2). For samples with a single type of copy number state (i.e., Del.d, Del.s, Dup.s, Dup.d), FLCNA still outperformed other methods in detecting deletions or duplications (Figure A.3-A.5). All methods provided higher ARI in deletions than duplications, and in double copy changes than single copy, due to the stronger signals presented in the intensities for the former than the latter cases. Overall, FLCNA was the best method in detecting commonly shared CNAs.

We also evaluated the accuracy of FLCNA in CNA detection by comparing it to SCOPE and HMMcopy (Figure 2.3 and Figure A.6-A.10). FLCNA demonstrated improved accuracy in identifying CNAs as the proportion of shared CNAs increased. For example, with a fixed number of CNAs (i.e., 50) shared by all samples in each of five clusters, FLCNA had an *F*1 score of 0.896, while SCOPE only had a score of 0.435 and

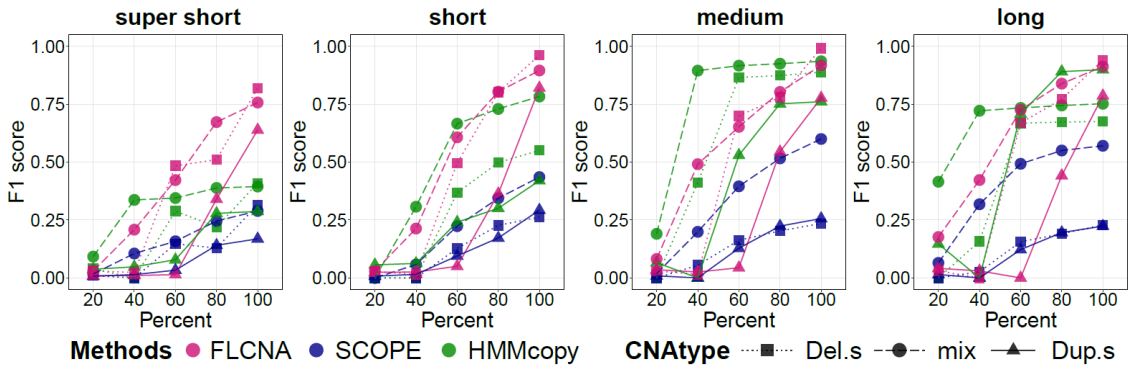

Figure 2.3 Assessment of FLCNA to detect CNAs using simulation data with five clusters. CNA calls were generated by FLCNA, SCOPE and HMMcopy, respectively. For each of five clusters, we added signals of 50 CNA segments to the background signals with varied lengths (super short: 2~5 markers, short: 5~10 markers, medium: 10~20 markers, and long: 20~35 markers) and varied CNA proportions (20%, 40%, 60%, 80%, 100%), respectively. Deletion of a single copy (Del.s), mixed CNA states (mix) and duplication of a single copy (Dup.s) were spiked in separately. F1 score was utilized to evaluate the performance of CNA detection for each method.

HMMcopy had a score of 0.783 (Figure 2.3). In general, FLCNA outperformed the other methods in detecting super short and short CNAs.

In summary, FLCNA showed overall great performance in clustering subclones with copy number changes shared by a large proportion of samples within a cluster. In copy number profile detection, FLCNA presented its advantage in detecting short CNAs shared by a relatively larger proportion of samples within a cluster.

2.3.2 Application to the TNBC breast cancer single cell study

FLCNA was also applied to a single cell study of three breast cancer patients, resulting in the identification of three clusters in the KTN126 patient and two in the other two patients (Figure 2.4 and Figure A.11-A.12). In particular, for the KTN126 patient, 64 cells (17 cells with pre-treatment and 47 cells with post-treatment) were clustered in cluster A, with 9 cells in cluster B and 20 cells in cluster C, all from the pre-treatment group. Similar copy number profile patterns were observed within clusters. Interestingly,

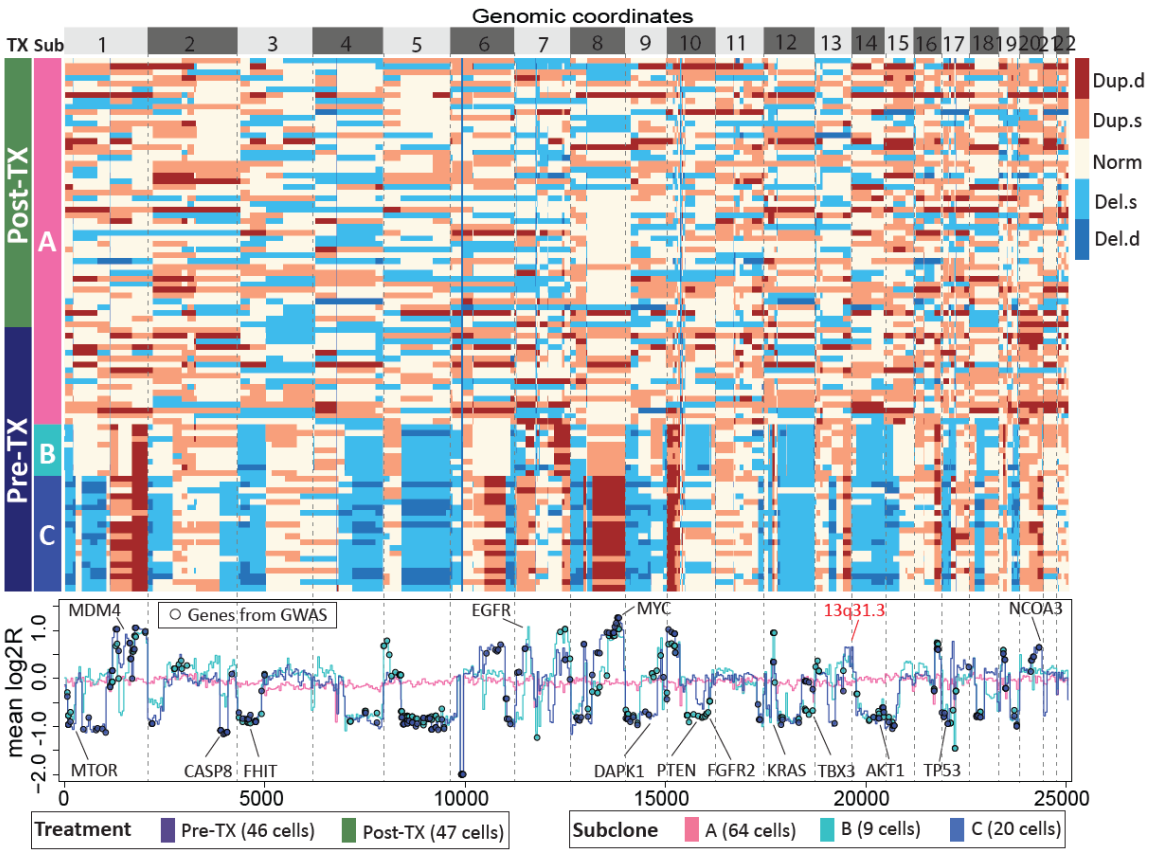

Figure 2.4 Subclone clustering of the KTN126 patient using FLCNA. Cell clusters and copy number profile with different CNA states (Del.d, Del.s, Norm, Dup.s and Dup.d) were generated using FLCNA. Mean log2R were provided for each cluster. Shared CNAs identified using FLCNA were matched to significant genes from the genome-wide association studies (GWAS) in the NHGRI-EBI GWAS Catalog. Del.d: Deletion of double copies; Del.s: Deletion of a single copy; Norm: Normal/diploid; Dup.s: Duplication of a single copy; Dup.d: Duplication of double copies; log2R: Logarithm transformation of ratio between normalized read counts and its sample specific mean; Pre-TX: pre-treatment; Post-TX: post-treatment.

clusters B and C, which included more pre-treatment samples, showed higher variation in genetic intensities, indicating that some copy number aberrations were treatment-specific. A more interesting and feasible interpretation is that the treatment of NAC may lead to extinction of tumor cells with these copy number changes in this patient. Similar patterns were observed in the other two patients (Figure A.11-A.12), and the locations of these

treatment-specific copy number changes were consistent across patients KTN129 and KTN302.

FLCNA identified 264, 154 and 156 shared CNAs in KTN126, KTN129 and KTN302 patients, respectively. Consensus CNA located genes (e.g., *DAPK1*, *TBX3*, *NCOA3*, *KRAS*) among patients implied the shared common evolutionary path of the tumor cells. Mutations in these genes have been shown to play importance roles in the evolution (*DAPK1* (Zhao *et al.*, 2015), *TBX3* (Fischer and Pflugfelder, 2015)) or the therapy (*NCOA3* (Burandt *et al.*, 2013), *KRAS* (Tokumaru *et al.*, 2020)) of breast cancer. The shared CNAs were mapped to 436 out of 575 breast cancer risk-associated genes identified from existing GWAS. These mapped genes were enriched in pathways related to cancer, hormones, immunity, and epithelial-mesenchymal transition (EMT) (Figure A.13). Most pathways were shared by all three patients and consistent with findings from existing studies on breast cancer. For instance, PATHWAYS_IN_CANCER was found to be hyperactivated in the human tumor tissue (Seryakov *et al.*, 2021). EMT (e.g., ADHERENS_JUNCTION(Liu *et al.*, 2016)) and immune (e.g., TOLL_LIKE _RECEPTOR(Shi *et al.*, 2020)) related pathways were associated with the invasion and metastasis of tumor cells. Various hormones also played importance roles in the occurrence and progression of breast cancer (Subramani *et al.*, 2017). We also observed a novel CNA in 13q31.3 (e.g., *LINC01040*) among all three patients, with duplications in KTN126 and KTN129 patients and deletions in the KTN302 patient. This discovery of a new CNA may provide new insights in understanding the progression of breast cancer.

To evaluate the heterogeneity of CNAs among cells, the sharing proportion of CNAs was evaluated by cluster. The mean sharing proportion was 30.3% in the KTN126 patient

(18.6% in cluster A, 36.4% in cluster B, and 34.7% in cluster C), 25.7% in the KTN129 patient, and 26.4% in the KTN302 patient (Figure A.14). Overall, our method detected CNAs with a wide range of length, varying from 2 to 1,000 bins (Figure A.15). For computational speed, with 93 cells from the KTN126 patient (Table A.1), FLCNA was much faster (1.2 hours) than SCOPE (10.5 hours) using a high-performance cluster with 12GB RAM, whereas HMMcopy took 0.5 hours.

2.4 Discussion

The importance of copy number change in modulating human disease is increasingly being recognized. scDNA-seq enables researchers to profile diseases and biological processes at the single-cell level. Accurate detection of CNAs with scDNA-seq data is crucial for identifying copy number profiles at a single-cell resolution, ultimately leading to a better understanding of how tumor lineages evolve (Baslan and Hicks, 2017; Ren *et al.*, 2018). Numerous scDNA-seq studies (Ferronika *et al.*, 2017; Gao *et al.*, 2016; Hou *et al.*, 2016) have used CNAs to characterize tumor subclones, and have found that most tumors contain multiple subclonal lineages. In our study, we developed FLCNA based on the fused lasso model for copy number estimation and simultaneous subclone clustering. Our simulations demonstrated the desirable performance of our method in clustering subclones and estimating copy number with scDNA-seq data, especially for data with a large sharing proportion of CNAs within a cluster.

Recent advances in NGS and SCS technologies have provided emerging tools for computational methods to infer subclones with different genomic variants, including single nucleotide alteration (SNAs) and CNAs. SAPPH (Xia *et al.*, 2015) estimated CNAs and infer tumor subclone proportion from paired tumor-normal data. The SCClone

(Yu *et al.*, 2022) and SiCloneFit (Zafar *et al.*, 2019) methods clustered subclones using single cell SNAs data to reconstruct the clonal populations and the evolutionary relationship between the clones. Elyanow *et al*. (Elyanow *et al.*, 2021) overcame the challenges in inferring CNAs from RNA-sequencing data by utilizing spatial information of a small group of cells to help identify CNAs and the spatial distribution of clones within a tumor sample. A major distinguishing feature of our method is its simultaneous clustering of cells and detection of CNAs, which solves the problem of declined clustering performance resulting from falsely identified variants in the stage of variants estimation.

The superior clustering performance of FLCNA has been demonstrated in extensive simulations. Our method was developed specifically for scDNA-seq with a large proportion of shared CNAs, which are more prevalent in single cell data than in bulk sequencing data or combined-cell samples (Cai *et al.*, 2014). Consistently, our spike-in simulations provided powerful clustering results from FLCNA for the samples with a large proportion of shared CNAs. Using a whole genome dataset with breast cancer, FLCNA successfully clustered subclones and provided clearly different genomic variation patterns in different clusters. With real dataset, we also found that treatment of NAC may lead to extinction of tumor cells despite of the possible bias brought by the fact that partial of the post-treatment samples were collected adjacent to normal tissues. In addition to prediction accuracy, the high dimensionality nature of scDNA-seq datasets desires for high computational efficiency. Even with 100kb in the binning size instead of 500kb as used in SCOPE, our method remained a high computational speed, meanwhile inclusively allowing the accurate detection of short CNAs.

As with any study, there are limitations in our study. First, only shared breakpoints identified in a cluster were used to estimate the underlying copy number profile for each cell, which might not be desirable when the goal is to identify CNAs at the single cell level with a relatively low sharing proportion in this cluster. Additionally, errors due to doublets in single cell sequencing data, where ≥ 2 cells are accidently mixed for sequencing (Zaccaria and Raphael, 2021), were not considered in our modeling. Doublets have been found to affect the accuracy of copy number profile estimation from previous studies (Zaccaria and Raphael, 2021). However, since only shared breakpoints were utilized for CNA detection in our study, which may neutralize the bias arising from the existence of doublets since the effect of a few doublets might be diluted by other majority cells from the same cluster. As a result, breakpoints can still be accurately detected.

In conclusion, our FLCNA method sheds new light on the methodology development of CNA detection and subclone identification in single-cell sequencing data using a novel statistical learning strategy. Modern sequencing technologies are continually emerging with multi-omics, spatial or temporal capabilities, which can facilitate better understanding of how subclonal lineages associate with cellular phenotypes, invasiveness and treatment responses (Evrony *et al.*, 2021). Our method has the potential to extend to other data types, such as scRNA-seq, and integrate different dimensions of sequencing information (e.g., gene expression, spatial information) to improve the identification of subclones and benefit research in cancer outcome-related targeted therapy. For example, we can incorporate the spatial information into our model for more comprehensive inference of subclones with scDNA-seq data, as cells located nearby are more likely to

share similar genetic patterns and consequently tend to reside within same subclones (Elyanow *et al.*, 2021).

Chapter 3: Spatial pattern and differential expression analysis with spatial transcriptomic data

Abstract

The advent of spatial transcriptomic technologies has opened new avenues to investigate cellular organization while preserving the spatial context of tissues. With data generated using such technologies, the identification of differential expression patterns is an essential first step for understanding tissue landscapes and biological processes. However, methods developed for the analysis of spatial transcriptomic data are still in their infancy. None of the existing methods can investigate SV genes between groups, which will be crucial for the discovery of significant biomarkers and development of targeted therapy in diseases. Therefore, in our study, we developed a spatial pattern and differential expression analysis method, referred to as SPADE, to identify SV genes with spatial transcriptomic data. SPADE is based on a Gaussian process regression model with a Gaussian kernel to detect SV genes within groups. In addition, SPADE provides a framework for identifying SV genes between groups using a crossed likelihood-ratio test. Extensive simulations and real data analyses were utilized for the evaluation of SPADE to identify SV genes in spatial transcriptomic data by comparing it to other existing methods (SpatialDE, SPARK, MERINGUE). In conclusion, superior performance was observed in the SPADE method compared to other methods to detect SV genes both within and between groups.

3.1 Introduction

The availability of single cell RNA sequencing (scRNA-seq) technologies has revolutionized our understanding of transcriptionally distinct subpopulations in tissues. However, current protocols for investigating scRNA-seq overlook potentially valuable spatial information that is essential for explaining how subpopulations of cells are organized in space and how they impact tissue functions (Maniatis *et al.*, 2021). Recently, spatial transcriptomic technologies such as 10x Visium (Ji *et al.*, 2020), Slide-seq (Stickels *et al.*, 2021) and Stereo-seq (Chen *et al.*, 2022) have emerged, enabling gene-expression profiling with molecular and single-cell resolution while preserving spatial information of tissues (Zhuang, 2021). These technologies are essential for understanding neurobiology and tumor biology, where biological activities are closely tied to the specific organization of cells (Method of the Year 2020: spatially resolved transcriptomics., 2021). For example, spatial transcriptomics has helped to reveal a detailed landscape of melanoma metastases, providing insights into tumor progression and therapy outcome (Thrane *et al.*, 2018). However, modeling transcriptomic data with hundreds of spatial locations remains statistically and computationally challenging (Sun *et al.*, 2020).

To characterize complex tissues using spatial transcriptomics, identifying spatially variable (SV) genes is an essential first step (Sun *et al.*, 2020). These genes are defined as those with uneven, aggregated, or patterned spatial distribution of expression measures, and are crucial for biomarker discovery and drug development. Nevertheless, current computational methods for identifying SV genes are still in their early stages of development. We illustrate three previously developed methods in this study. SpatialDE

(Svensson *et al.*, 2018) decomposes expression variability into spatial and non-spatial components, then quantifies the spatial variance using the ratio between these two components to detect SV genes. SPARK (Sun *et al.*, 2020) uses a linear spatial model with multiple Gaussian and periodic kernels to identify SV genes with various spatial patterns. MERINGUR (Miller *et al.*, 2021) uses spatial autocorrelation and cross-correlation analysis to identify SV genes in spatial transcriptomic data. However, these existing methods only use fixed hyperparameter values in their kernels, which may limit their ability to capture specific spatial patterns. The hyperparameter in kernels, known as the characteristic length scale, determines how rapidly the covariance decays as a function of distance. Furthermore, these methods can only identify SV genes within a single group and cannot detect genes that show differential spatial patterns between groups, which is essential for understanding changes in spatial patterns under different treatment conditions or time phases.

To address these limitations, we developed SPADE, a spatial pattern and differential expression analysis method, for identifying SV genes in complex tissues using spatial transcriptomic data. This method is based on a Gaussian process regression (GPR) model (Rasmussen and Williams, 2006) with a Gaussian kernel that estimates a gene-specific length-scale hyperparameter in the kernel function to increase the accuracy of SV gene identification within a single group. Moreover, SPADE provides a framework for detecting SV genes between groups from different treatment conditions or time phases using a crossed likelihood-ratio test, making it a valuable tool for discovering potential biomarkers for drug development. To evaluate the performance of SPADE to identify SV genes within groups, we benchmarked it against existing spatial pattern investigation

methods (i.e., SpatialDE (Svensson *et al.*, 2018), SPARK (Sun *et al.*, 2020), MERINGUE (Miller *et al.*, 2021)) using extensive simulation studies and real data analyses with predefined "truth" SV genes. We also assessed the performance of SPADE in detecting SV genes between groups using simulations and a real dataset with axolotl. The results showed that SPADE outperformed other methods in identifying SV genes both within and between groups, demonstrating its superior performance in disclosing biological discoveries. In conclusion, SPADE is a powerful method for identifying SV genes in spatial transcriptomic data and has the potential to improve biomarker discovery and drug development.

3.2 Methods

SPADE was developed based on a GPR model with a Gaussian kernel to identify SV genes using spatial transcriptomic data. GPR model can model the relationship between gene expression and other covariates (i.e., cell groups) incorporating the spatial information of multiple cells. Besides identifying SV genes within groups, SPADE is also capable of detecting SV genes between different groups using a crossed likelihood-ratio test. Note that the tissue composition can consist of either single cells or small sets of cells located in small regions known as spots. For consistency, we will use spots below to describe the model across the manuscript.

3.2.1 Data normalization

Read counts data were utilized as input in the SPADE method, and were first normalized into continuous data using a two-step normalization strategy to speed up the parameter estimation process described below. In particular, to stabilize the mean-variance

dependency which was observed in spatial transcriptomic data (Shang and Zhou, 2022), we transformed the counts data using an Anscombe's transformation strategy with $R_i' = \log(R_i + 1/\phi_i)$ (ANSCOMBE, 1948), where R_i was the read counts for i-th gene. ϕ_i was the overdispersion parameter, which was introduced to capture the dependency of mean and variance for each gene with $Var(R_i) = E(R_i) + \phi_i \cdot E(R_i)^2$ (McCarthy *et al.*, 2012). Thereafter, to correct for the effects of library size in detecting SV genes, we used a linear regression to regress out library size from the above transformed expression data R_i' (Svensson *et al.*, 2018). After library size correction, the normalized continuous data were utilized in the main step of the SPADE method as below.

3.2.2 Notation and models

For genes of our interest, let the normalized continuous expression data denote as $\boldsymbol{Y} = (y_{ij})_{P\times N}$, where y_{ij} is the expression value for the i-th ($i = 1, \ldots, P$) gene and the j-th spot in the spatial pattern, considering we have N spots in total. The two-dimensional spatial coordinates for this j-th spot can be expressed as $\boldsymbol{s}_j = (s_{j1},\ s_{j2})$. To identify SV genes, for each gene, a GPR model (Rasmussen and Williams, 2006) is utilized to model gene's expression level across different spot locations. In the GPR model, the normalized continuous data y_{ij} can be modeled as a linear combination of three terms

$$y_{ij} = \boldsymbol{x}_{ij}^T\boldsymbol{\beta}_i + Z_{ij} + \varepsilon_{ij}, \qquad (1)$$

where $\boldsymbol{x}_{ij}$ is a k vector of covariates that include k -1 covariates (e.g., batch effect, cell type) and a scalar one for the intercept. Intercept term represents the mean log-expression of the gene across all spots. $\boldsymbol{\beta}_i$ is the vector of coefficients for these covariates and ε_{ij} is

the residual error that is independently and identically distributed from $N(0,\ \tau_{i2})$. Z_{ij} is a zero-mean stationary Gaussian process modeling the spatial correlation pattern with $\boldsymbol{Z}_i = (Z_{i1}, \dots, Z_{iN})^T \sim \mathrm{MVN}(0, \tau_{i1}\boldsymbol{K}_i)$, where $\boldsymbol{K}_i$ is a covariance kernel function for the i-th gene based on pairwise distances of all the pairs of spots, τ_{i1} is a scaling factor, and MVN denotes a multivariate normal distribution. Kernel functions are widely used to map data with spatial information into high-order feature space and capture the nonlinear interactions among spots (Ton *et al.*, 2018). In our method, we utilize a Gaussian kernel to model the covariance matrix among spots which has been commonly used in other studies on spatial transcriptomics (Svensson *et al.*, 2018; Sun *et al.*, 2020). Gaussian kernel for the i-th gene can be denoted as $\boldsymbol{K}_i = \exp\left(-\frac{D^2}{2\theta_i^2}\right)$, where $D = \|\boldsymbol{s} - \boldsymbol{s}'\|$ is the Euclidean distance matrix (Dokmanic *et al.*, 2015) of spots based on spatial coordinates. θ_i is the hyperparameter of length-scale in the kernel function $\boldsymbol{K}_i$ for the i-th gene. This hyperparameter controls the curvature of the covariance to the distance between spots that larger θ_i corresponds to smoother covariance changes (Rasmussen and Williams, 2006). Thus, the total covariance for the continuous expression data $\boldsymbol{y}_i = (y_{i1}, \dots, y_{iN})^T$ is $\boldsymbol{\Sigma}_i = \tau_{i1}\boldsymbol{K}_i + \tau_{i2}\mathbf{I}$, where $\boldsymbol{I}$ is an n-dimension identify matrix. τ_{i1} effectively measures the expression variance in $\boldsymbol{y}_i$ captured by spatial patterns and τ_{i2} measures the expression variance in $\boldsymbol{y}_i$ owing to random noise. As far, the problem of identifying SV genes has been transferred into testing the null hypothesis "$H_0: \tau_{i1} = 0$". Also, the power of SV gene identification relies on how well spatial kernel matrix $\boldsymbol{K}$ matches the significant spatial patterns shown by genes of our interest. Therefore, our goal is to estimate $\boldsymbol{\Theta}_i = (\boldsymbol{\beta}_i, \theta_i, \tau_{i1}, \tau_{i2})$ for the i-th gene.

3.2.3 Parameter estimation and identification of spatially variable genes

To test whether a gene displays spatially variable pattern, we first estimate the optimal hyperparameter θ_i for the i-th gene, which can improve the accuracy of identifying SV genes. After marginalizing over function values $\boldsymbol{Z}_i$, we obtain the multivariate normally distributed marginal likelihood of the expression values $p(\boldsymbol{y}_i|\boldsymbol{X}_i, \boldsymbol{\Theta}_i) = MVN(\boldsymbol{y}_i|\boldsymbol{X}_i^T\boldsymbol{\beta}_i, \boldsymbol{\Sigma}_i)$, where $\boldsymbol{X}_i = (\boldsymbol{x}_{i1}, \dots, \boldsymbol{x}_{iN})^T$. The calculation of $p(\boldsymbol{y}_i|\boldsymbol{X}_i, \boldsymbol{\Theta}_i)$ is sped up through spectral decomposition (Runcie and Crawford, 2019) of the covariance matrix $\boldsymbol{K}_i$. To estimate $\boldsymbol{\Theta}_i$, we optimize the log marginal likelihood with respect to the coefficients of covariates $\boldsymbol{\beta}_i$, kernel hyperparameter θ_i, spatial variance τ_{i1}, and random variance τ_{i2} using an optimization strategy of golden section search (Anderson, 1974). More details about the estimation of these parameters are available in Appendix B.1.

After the optimal length-scale hyperparameter ($\hat{\theta}_i$) is selected for the i-th gene, under the null hypothesis ($\tau_{i1} = 0$), the coefficients for covariates can be estimated ($\widehat{\boldsymbol{\beta}}_i$) using a score-based variance-component test, SKAT (Wu *et al.*, 2011). Thus a Q score statistic can be easily calculated based on the GPR model with a quadratic form $Q_i = \left(\boldsymbol{y}_i - \boldsymbol{X}_i^T\widehat{\boldsymbol{\beta}}_i\right)^T \widehat{\boldsymbol{K}}_i\left(\boldsymbol{y}_i - \boldsymbol{X}_i^T\widehat{\boldsymbol{\beta}}_i\right)$ (Wang *et al.*, 2015). $\widehat{\boldsymbol{K}}_i$ is the estimate of kernel covariance matrix based on the optimal length-scale hyperparameter $\hat{\theta}_i$. Notably, Q_i follows a mixture of independent chi-square distributions with mixing weights that depend on the eigenvalues of the kernel matrix. Based on this chi-square mixture distribution and Q_i value, an exact method based on the Davies method (Moschopoulos and Canada, 1984) is utilized to compute P-value for the i-th gene. P-values across all genes are adjusted with

the Benjamini and Hochberg (BH) method (Haynes, 2013) to correct for occurrence of false positives.

Furthermore, the SPADE method can identify SV genes between groups (e.g., group A and B) based on a crossed likelihood-ratio test with spatial transcriptomic data. To achieve this, we first estimate the optimal hyperparameter in the kernel function for each group, respectively. Thus, for each gene, the log likelihood ($\mathcal{L}$) in group A and B can be easily calculated with its optimal kernel matrix, referred to as $\mathcal{L}_A$ and $\mathcal{L}_B$, separately. Then we use the optimal hyperparameter values estimated from each group to compute alternative log likelihoods for their counterparts ($\mathcal{L}_{AB}$, $\mathcal{L}_{BA}$). By comparing alternative likelihoods to optimal log likelihoods, we can test whether these two groups have the same spatial patterns. The likelihood-ratio test statistic is computed with $\lambda_{LR} = 2 * (\mathcal{L}_A + \mathcal{L}_B - \mathcal{L}_{AB} - \mathcal{L}_{BA})$ and *F* test with degree freedom of one is used to calculate P-values. Also, P-values across all genes were adjusted using the BH method (Haynes, 2013).

3.2.4 Data description

Four publicly available spatial transcriptomic datasets, including MOB (Ståhl *et al.*, 2016), SeqFISH (Shah *et al.*, 2016), MERFISH (Moffitt *et al.*, 2018) and ARTISTA (Wei *et al.*, 2022), were utilized in our study to evaluate the performance of SPADE for identifying SV genes. The MOB, SeqFISH and MERFISH datasets were used to mimic real data in simulations. In real data analyses, the SeqFISH and MOB datasets were utilized for evaluating SPADE to detect SV genes within groups, while the ARTISTA dataset was used to evaluate the performance of SPADE in detecting SV genes between

groups. Raw read counts were provided for each dataset. Details of these four datasets were illustrated below:

The MOB dataset (Ståhl *et al.*, 2016) contains spatial transcriptomic data with mouse olfactory bulb (MOB) which is available in Spatial Transcriptomics Research (http://www.spatialtranscriptomicsresearch.org). After filtering out genes expressed in less than 10% spots and spots with less than 10 total read counts (Sun *et al.*, 2020), 11,274 genes on 260 spots were kept in the final set for simulation studies and real data analyses.

The SeqFISH dataset (Shah *et al.*, 2016) was collected on the mouse hippocampus with 249 genes measured on 257 spots of which 35 genes were pre-defined in its original study as cell identity markers. After filtering out cells located at the boundary to relieve border artifacts (Edsgärd *et al.*, 2018), a final set of 249 genes measured on 131 spots were retained for the evaluation of SPADE to identify SV genes within groups in both simulations and real data analyses.

The MERFISH dataset (Moffitt *et al.*, 2018) was obtained from the preoptic region of the hypothalamus in mouse, which was available in Dryad platform (https://datadryad.org/stash/dataset/doi:10.5061/dryad.8t8s248). Slice from animal 18 was utilized for analysis, which included 160 gene and a large number (>5,000) of single cells. After removing the ambiguous cells that were identified as putative doublets (Sun *et al.*, 2020), a final set of 160 genes on 4,975 spots was kept in the simulation studies for evaluating SPADE to identify SV genes within groups.

The ARRISTA dataset provided visualization of gene expression across the regeneration stages of axolotl telencephalon at single cell resolution (Wei *et al.*, 2022),

which can be downloaded from https://db.cngb.org/stomics/artista/. The high-resolution Stereo-seq technology was utilized to generate spatial transcriptomic data for two (2DPI) and five (5DPI) days post injury brain tissue of axolotl. These two post injury groups were utilized to evaluate the performance of SPADE for identifying SV genes between groups. After filtering out less informative genes which expressed in less than 20% spots and spots with less than 1,000 total read counts, 2,168 consistent genes were left in both groups with 3,961 spots in the 2DPI group and 3,128 spots in the 5DPI group. 349 pre-defined cell type markers from its original study (Wei *et al.*, 2022) were used for the evaluation of SPADE.

3.2.5 Assessment of SPADE to identify spatially variable genes within groups

To evaluate SPADE's ability to identify SV genes within a single group, we conducted simulation studies and real data analyses, comparing it to existing methods including SpatialDE (Svensson *et al.*, 2018), SPARK (Sun *et al.*, 2020) and MERINGUE (Miller *et al.*, 2021). Simulations were performed in two sets. In the first set with more standard patterns, simulations mimicked the MOB dataset, as introduced in Section 3.2.4 Data description, to generate simulation data using a commonly used simulator for scRNA-seq data, *splatter* (Zappia *et al.*, 2017). We generated 200 genes and 200 spots with spots coordinates simulated with a random-point-pattern Poisson process (Gómez-Rubio, 2016). These simulation data were treated as "pattern free" data of which 50 genes were randomly picked to assign different spatial patterns as "true" SV genes for the evaluation of SPADE. Two distinct spatial patterns were included in the simulations, including the hotspot and streak patterns, and different proportions (10%, 20%, 30%) of marked spots were considered for pattern. Moreover, to evaluate how signal strength affects the

performance of different methods, we utilized various expression fold changes (1.2, 1.3, 1.4, 1.5, 1.6) for these marked spots.

In the second set of simulations with more complicated patterns, we simulated gene expression data with spot coordinates from the SeqFISH, MERFISH and MOB datasets, respectively. Data were simulated based on a negative binomial distribution to first generate "pattern free" data with parameters estimated from each original dataset. Representative genes from real datasets were then mimicked to construct spatially variable patterns in simulation studies. The SV genes identified by all analysis methods were defined as representative genes in simulations, including *lvye* in the SeqFISH dataset, *Cd24a* in the MERFISH dataset, and *Fabp7* in the MOB dataset. Based on gene expression of these representative genes, spots with expression value > 0.75 quantile of all spots were assigned as potential marked spots, and given various fold changes (1.5, 2.0, 2.5, 3.0, 3.5). In simulations, we generated data with 200 genes and the same number of spots for each dataset. Still, 50 genes were randomly selected as "truth" SV genes for the assessment of SPADE. In both sets of simulations, the performance of the SPADE method to detect SV genes was assessed by *F*1 score.

We also assessed the performance of SPADE to detect SV genes within groups using real datasets, including the SeqFISH and MOB datasets (details are shown in Section 3.2.4 Data description). In the SeqFISH dataset, 35 genes were pre-selected from its original study as cell identity markers which were utilized as "true" SV genes for comparing SPADE to existing methods (SpatialDE (Svensson *et al.*, 2018), SPARK (Sun *et al.*, 2020) and MERINGUE (Miller *et al.*, 2021)). In the MOB dataset, two different lines of evidence were used to validate the SV genes identified from each method. First,

10 highlighted marker genes in the olfactory system from its original study (Ståhl *et al.*, 2016) were used as "true" markers. Second, 902 cell type-specific marker genes from a recent single-cell RNA sequencing study (Tepe *et al.*, 2018) in the olfactory bulb were utilized as "gold standard" for the assessment of different methods. Similarly, all methods were evaluated using precision rate, recall rate and *F*1 score.

3.2.6 Assessment of SPADE to identify spatially variable genes between groups

SPADE was also evaluated to identify SV genes between groups using simulation studies and real data analyses. In simulations, SPADE was benchmarked to existing methods which were developed for typical differential expression (DE) analysis in RNA-seq data including edgeR (Robinson *et al.*, 2010), DESeq2 (Love *et al.*, 2014), limma-voom (Law *et al.*, 2014) and MAST (Finak *et al.*, 2015). Still, the MOB dataset was used to generate two groups of simulation data as "pattern free" data with 200 spots each and 200 genes in total. In the first group, 50 genes were randomly selected as "truth" SV genes and assigned with different spatial patterns (i.e., hotspot and streak). We also considered different proportions (10%, 20%, 30%) and signal strengths with various fold changes (1.5, 2.0, 2.5, 3.0, 3.5) for these marked spots. The receiver operating characteristic (ROC) curve was generated and the areas under the ROC curve (AUC) were utilized to assess the power of different methods.

Moreover, we assessed the robustness of SPADE in detecting SV genes between groups using data with varying spots coordinates and pattern directions. Specifically, to evaluate how spots coordinates in two groups affect the performance of SPADE, we compared SPADE in two groups with the same coordinates to those with different

coordinates using AUC. We also benchmarked SPADE in groups with the same pattern directions to those with different directions. Note that patterns with the same signal shapes and strength but with different directions were regarded as non-SV patterns in our method. Thus, false discovery rate (FDR) was used to assess the type I error of SPADE. Simulation studies included both hotpots and streak patterns, with different proportions (10%, 20%, 30%) of marked spots and various signal strengths with fold changes of 1.5, 2.0, 2.5, 3.0, and 3.5.

We further evaluated the performance of SPADE to identify SV genes between groups using a real spatial transcriptomic dataset (i.e., ARRISTA) with axolotl telencephalon, which has been described previously. With two groups of data from different post injury stages (i.e., 2DPI, 5DPI) and 2,020 consistent genes, SV genes between these two groups were identified using the SPADE method and compared to pre-selected cell type identity markers from its original study (Wei *et al.*, 2022). Pearson's chi-square test (Plackett, 1983) was utilized to investigate the association between SV genes from SPADE and the "true" markers.

3.3 Results

The SPADE method was developed based on a GPR model with a Gaussian kernel to identify SV genes from spatial transcriptomic data. Besides detecting SV genes within a single group, SPADE was also capable of identifying SV genes between groups. The framework of the SPADE method is summarized and illustrated in Figure 3.1. The first step of SPADE involved normalizing the original read count data into continuous data using a two-step normalization strategy. Next, instead of using a fixed length-scale

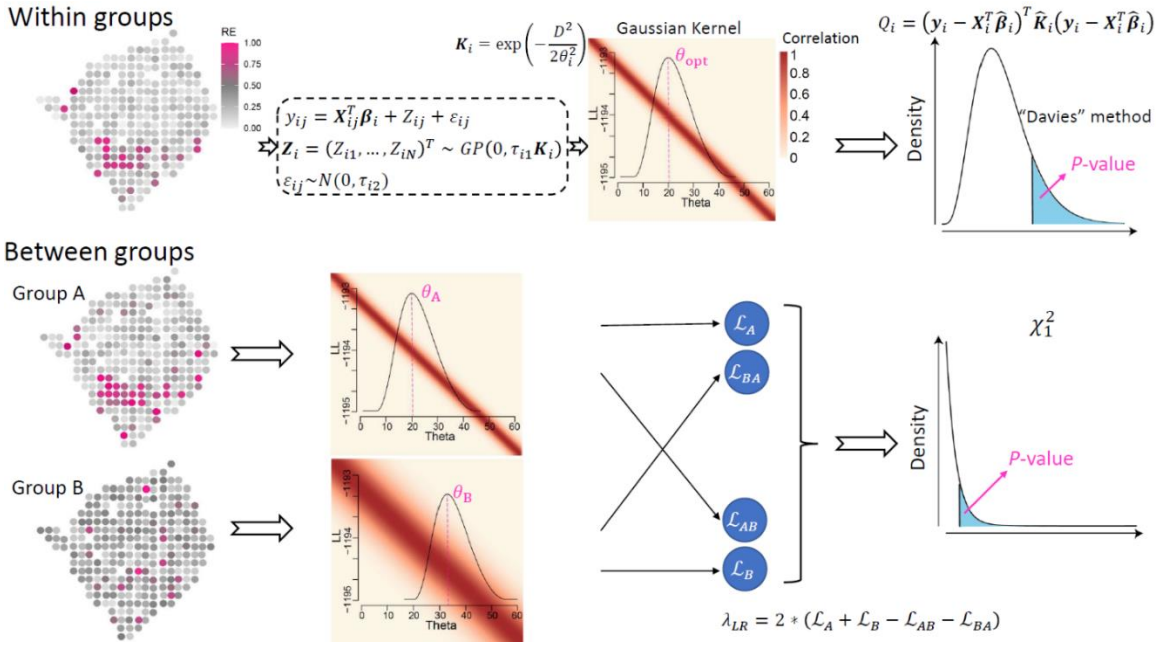

Figure 3.1 Framework of SPADE. SPADE was developed based on a Gaussian process regression model with Gaussian kernel to identify spatially variable (SV) genes within groups and between groups. The optimal length-scale hyperparameter in the kernel function was estimated for each gene and the *P*-value was calculated based on a quadratic *Q* statistic with the Davie method. To detect SV genes between groups, SPADE exchanged the optimal hyperparameters estimated from two groups and then utilized a crossed likelihood-ratio test to calculate the *P*-values.

hyperparameter across genes in the covariance kernel of the GPR model, SPADE estimated the optimal hyperparameter for each gene. A quadratic Q statistic with the Davies method was then used to compute a *P*-value for each gene. To detect SV genes between groups, SPADE exchanged the optimal hyperparameters estimated from two groups and utilized a crossed likelihood-ratio test to calculate the *P*-values. The performance of SPADE was evaluated through extensive simulation studies and real data analyses for both SV gene identification within groups and between groups.

3.3.1 Evaluation of SPADE to identify spatially variable genes within groups

We first assessed the performance of SPADE to identify SV genes within groups using two sets of simulations and three real datasets (SeqFISH, MERFISH, MOB). SPADE was evaluated by comparing its ability to identify SV genes to existing methods (SpatialDE, SPARK and MERINGUE) using *F*1 scores. In the first set of simulations, mimicking 200 genes and 200 spots using the MOB dataset, different proportions of marked spots (10%, 20%, 30%) and different fold changes (1.2, 1.3, 1.4, 1.5, 1.6) were simulated with different spatial patterns (hotspot, streak). For both hotspot and streak patterns, SPADE showed overall the highest *F*1 score among all methods in all scenarios (Figure 3.2), followed by SPARK, SpatialDE, and MERINGUE. For example, with a fold change of 1.4 for 20% marked spots in the hotspot pattern, SPADE performed better, with an *F*1 score of 0.83 to identify SV genes than other existing methods (SPARK: 0.75, SpatialDE: 0.51, MERINGUE: 0.04) **(**Figure 3.2A**)**. In particular, the performance of each method to identify SV was improved when the proportion of marked spots increased from 10% to 30%, as more marked spots contributed to the identification of spatial patterns. When the fold change of expression values for marked spots increased, it was easier for each method to identify SV genes because the expression pattern signal was more obvious. A smaller *F*1 score was observed for the streak pattern than that of the hotspot pattern, especially for scenarios with relatively low marker proportions and small fold changes. For instance, with 10% marked spots and a fold change of 1.4, the *F*1 score of SPADE for the hotspot pattern was 0.54, while that for the streak pattern was only 0.15. This suggested that hotspots patterns provided stronger signal information than streak patterns.

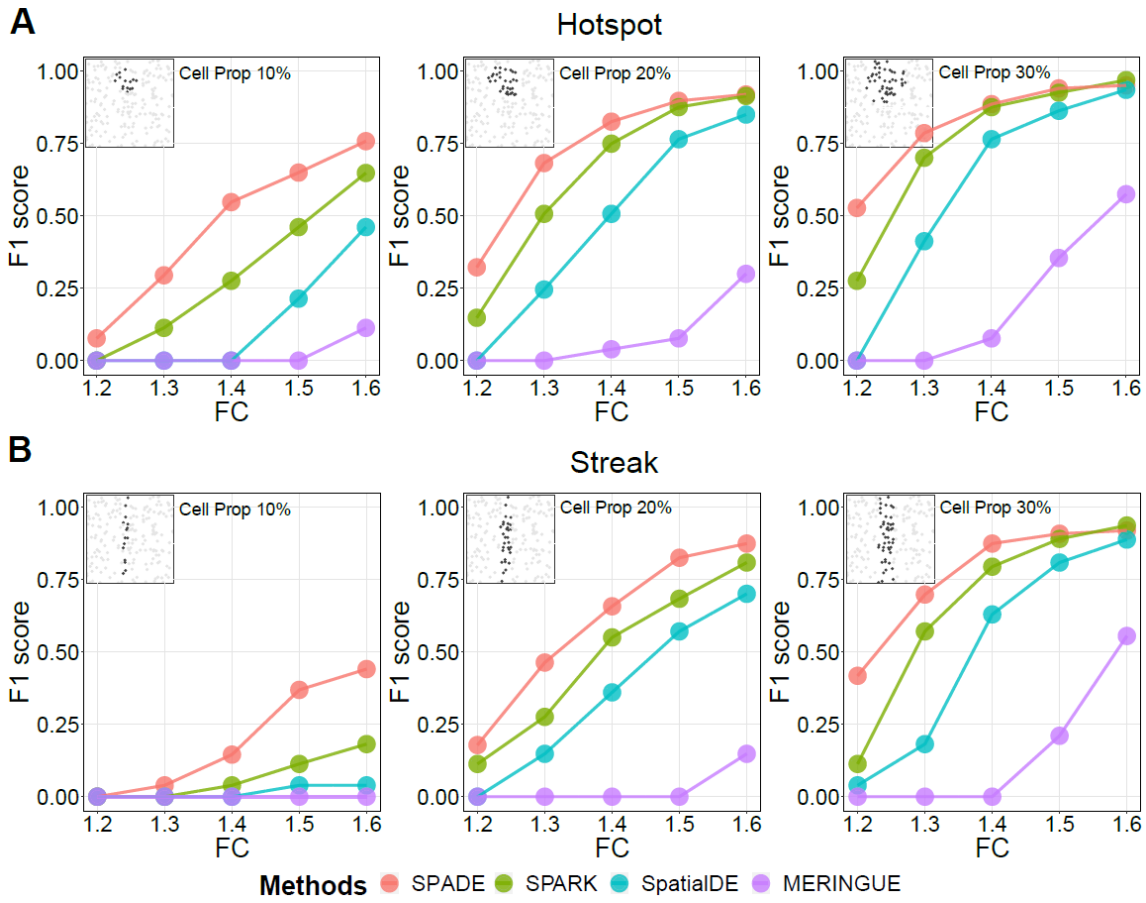

Figure 3.2 Assessment of SPADE to identify spatially variable genes within groups using simulations. Simulations were utilized to evaluate the performance of SPADE to identify spatially variable (SV) genes within groups. SPADE was compared to existing methods with *F*1 scores including SpatialDE, SPARK and MERINGUE. Two distinct spatial patterns were included in the simulations, including the hotspot (A) and streak (B) patterns. Different proportions (10%, 20%, 30%) of marked spots were considered in each pattern. We also utilized various expression fold changes (1.2, 1.3, 1.4, 1.5, 1.6) for the marked spots to evaluate how signal strength affects the performance of different methods.

In the second type of simulations, with significant patterns simulated real datasets (SeqFISH, MERFISH, MOB), SPADE was compared to existing methods using marked spots of various fold changes (1.5, 2.0, 2.5, 3.0, 3.5). SPADE was found to be the overall best method in all three datasets in all the scenarios (Figure 3.3). For example, in the SeqFISH dataset simulation with a fold change of 3.0 (Figure 3.3A), SPADE had the

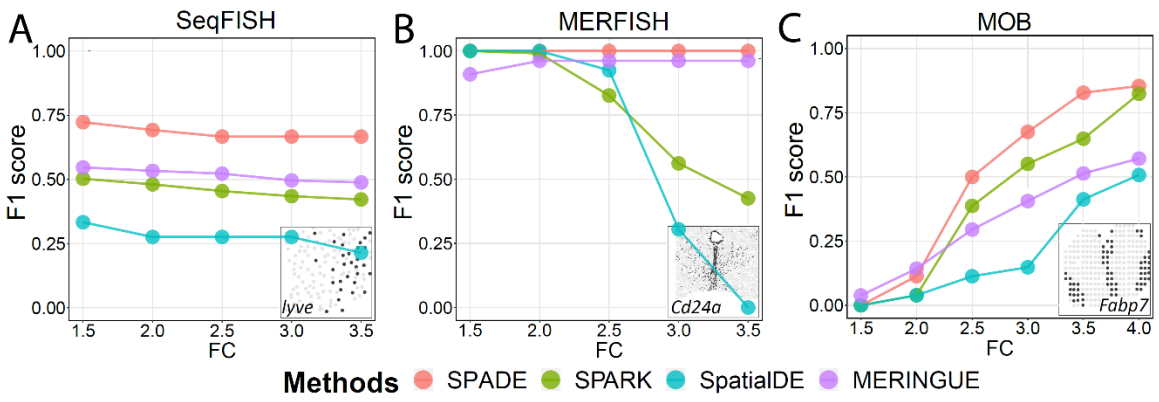

Figure 3.3 Assessment of SPADE to identify spatially variable genes within groups using real data-based simulations. Simulations with significant spatial patterns from real datasets, including SeqFISH (A), MERFISH (B) and MOB (C), were utilized for the evaluation of SPADE to identify spatially variable (SV) genes. SPADE was compared to existing methods with *F*1 scores including SpatialDE, SPARK and MERINGUE. Based on these predefined differential spatial patterns, spots with expression value > 0.75 quantile of all spots were assigned as marked spots in simulation data, including *lvye* in the SeqFISH dataset, *Cd24a* in the MERFISH dataset, and *Fabp7* in the MOB dataset. These marked spots were given expression values with various fold changes (1.5, 2.0, 2.5, 3.0, 3.5).

highest *F*1 score of 0.67, which was larger than that of SPARK (0.43), SpatailDE (0.28), and MERINGUE (0.50). In the MERFISH dataset with a relatively large number of spots (4,975) (Figure 3.3B), SPADE was the only method that provided consistent and the highest *F*1 score (1.0) among all fold changes. Interestingly, in this dataset, the performance of SPARK and SpatialDE declined as the fold change of marker spots ascended. Interestingly, the performance of SPARK and SpatialDE dramatically declined in detecting SV patterns with a larger difference in gene expression, characterized by more sparse marker spots. Additionally, a higher variation in the estimated length-scale hyperparameter values was given as the expression difference in fold change increased (Figure B.1), which suggested that existing methods with fixed hyperparameters failed to provide good identification performance in more complicated patterns (e.g., sparse marker spots). Increased power of all methods was observed in detecting the spatial

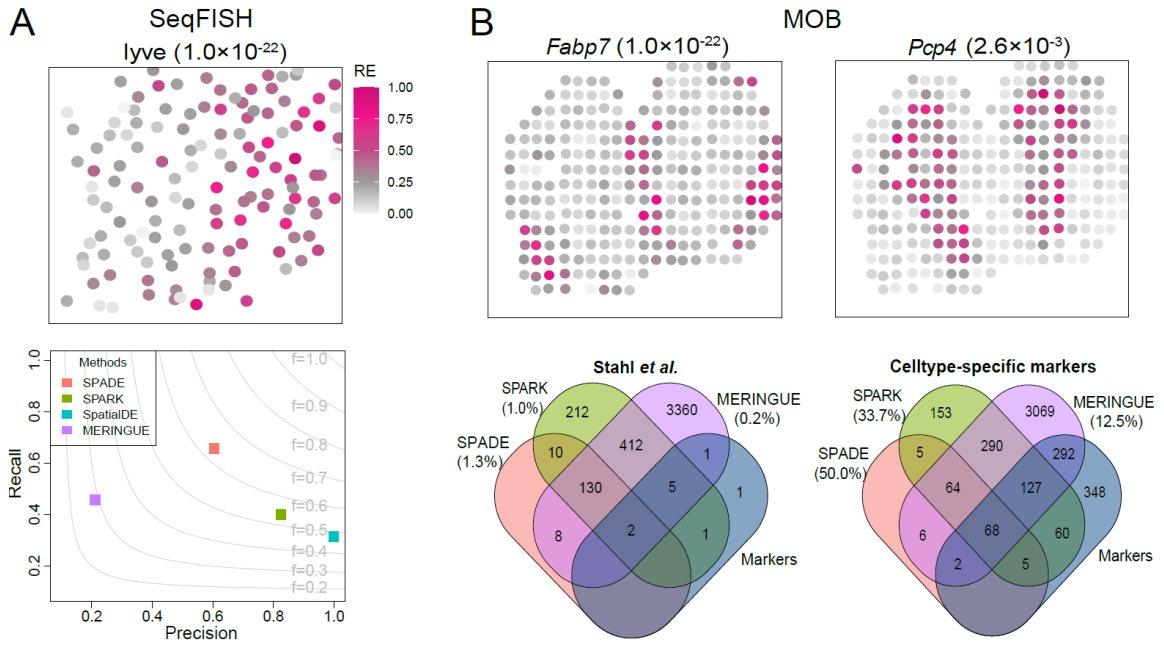

Figure 3.4 Assessment of SPADE to identify spatially variable genes within groups using real data analyses. Real datasets, including SeqFISH (A) and MOB (B), with predefined "true" markers were used for the evaluation of SPADE to identify spatially variable (SV) genes within groups. SPADE was compared to existing methods including SpatialDE, SPARK and MERINGUE. In the seqFISH dataset, 35 genes were pre-selected from its original study as cell identity markers for the assessment of SPADE. In the MOB dataset, two different lines of predefined "gold standard" were used to validate the SV genes identified from each method, including 10 highlighted marker genes in the olfactory system and 902 cell type-specific marker genes from a recent single-cell RNA sequencing study. Representative SV genes identified by SPADE in each dataset are shown, with *P* values from SPADE inside parentheses.

pattern with increased fold changes in the MOB dataset (Figure 3.3C), which is consistent with the detection of simulated spot and streak patterns discussed above.

With real data analyses, we further evaluated how well SPADE can identify SV genes within groups using the SeqFISH and MOB datasets with pre-defined "true" marker genes (Figure 3.4). SPADE was compared to existing methods (SpatialDE, SPARK and MERINGUE) using *F*1 score. In the SeqFISH dataset, SPADE performed better than other methods to successfully identify SV genes (e.g., *lyve*) with an *F*1 score of 0.63 for

SPADE, and 0.54, 0.48, and 0.29 for SPARK, SpatialDE and, MERINGUE, respectively (Figure 3.4A). In the MOB dataset, 150 SV genes (e.g., *Fabp7, Pcp4*) were identified by the SPADE method compared to 772 SV genes and 3,918 SV genes detected in the SPARK and MERINGUE methods, separately (Figure 3.4B**)**. The SpatialDE method did not identify any SV genes. We utilized two lines of evidence, highlighted markers and cell type-specific markers, to evaluate all methods. First, with SV genes identified from each method in the MOB dataset, we found that genes from SPADE (1.33%) included a higher proportion of highlighted markers than those from the SPARK (1.04%) and MERINGUE (0.20%) methods. Second, a higher proportion of SV genes from SPADE (50.0%) were cell type-specific marker genes compared to those from SPARK (33.7%) and MERINGUE (12.5%). These results partially validated that SPADE was the superior method for detecting SV genes within groups in the MOB dataset. For computational speed, with 249 genes and 131 spots in the SeqFISH dataset, the time required for SPADE to detect SV genes was 1.16 mins, which was longer than that of SPARK (1.08 mins) and MERINGUE (0.01 mins) but shorter than that of SpatialDE (1.68 mins) using a high-performance cluster with 12GB RAM.

In summary, with two sets of simulations and real data analyses, SPADE was found to be more accurate than existing methods in detecting SV genes within groups in spatial transcriptomic data.

3.3.2 Evaluation of SPADE to identify spatially variable genes between groups

We also evaluated the performance of the SPADE method to identify SV genes between groups using both simulation and real data analyses with predefined “true” marker genes.

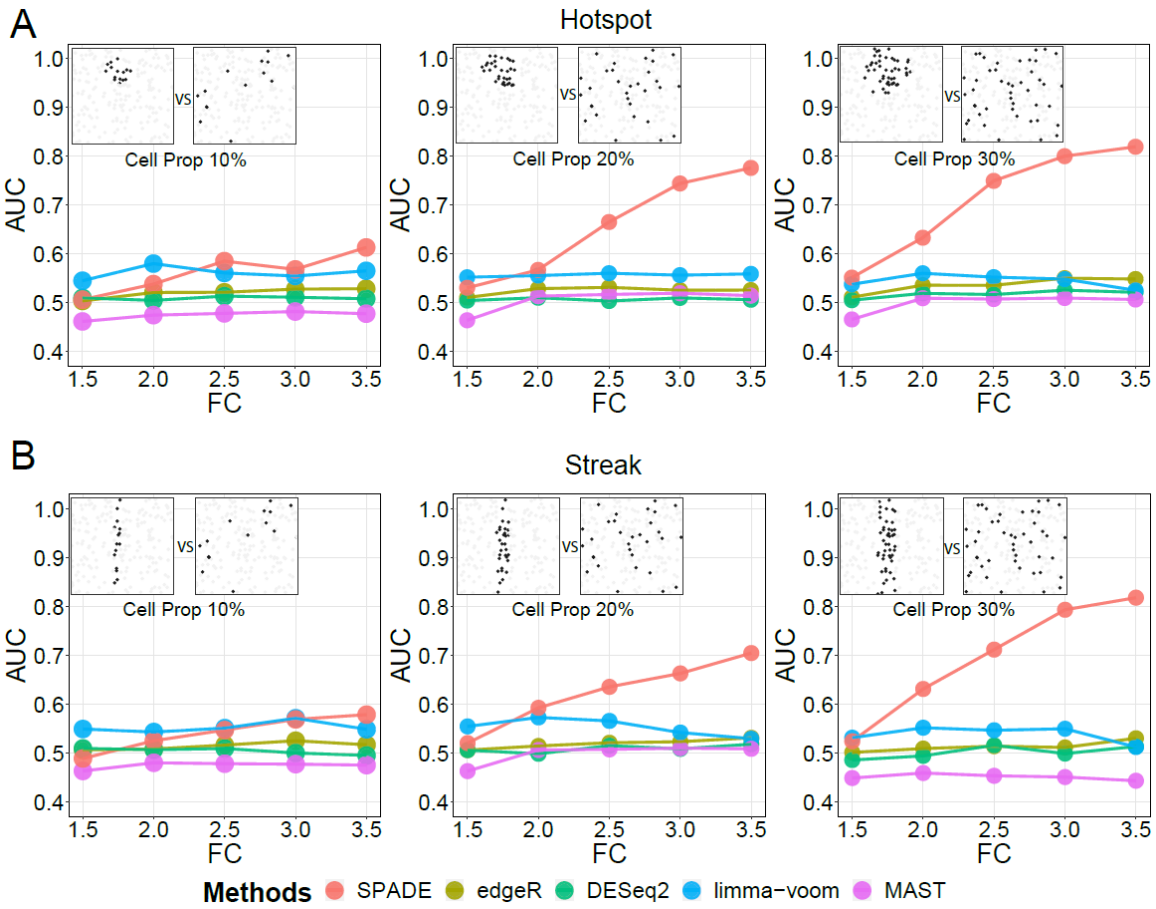

Figure 3.5 Assessment of SPADE to identify spatially variable genes between groups using simulations. SPADE was benchmarked to existing methods which were developed for typical differential expression (DE) analysis for RNA-seq data including edgeR, DESeq2, limma-voom and MAST. In simulations, two groups of data were generated with one group assigned with different spatial patterns including the hotspot (A) and streak (B) patterns. We considered different proportions (10%, 20%, 30%) of marked spots and different signal strengths with various fold changes (1.5, 2.0, 2.5, 3.0, 3.5) for these marked spots. With known "truth" SV genes, the receiver operating characteristic (ROC) curve was generated and the areas under the ROC curve (AUC) were utilized to assess the power of different methods.

Mimicking the MOB dataset, simulation data with 200 genes were generated which included two groups and 200 spots in each group. One group was assigned patterns of different marker proportions (10%, 20%, 30%), fold changes (1.5, 2.0, 2.5, 3.0, 3.5), and signal shapes (hotspot, streak). SPADE was benchmarked to other existing DE analysis methods using AUC, including edgeR, DESeq2, limma-voom, and MAST. SPADE

outperformed all other methods in all the scenarios, and existing DE analysis methods failed to accurately identify SV genes in spatial transcriptomic data (Figure 3.5). For example, with the hotspot pattern, 20% marked spots, and a fold change of 3.0, the AUC of SPADE was 0.74, while the AUC values of all other methods were only around 0.5 (Figure 3.5A). This suggested that classic DE analysis methods were only capable of identifying genes with a significant mean difference between two groups, which failed to identify SV genes from groups with similar mean expression values but different spatial patterns. Similarly, as fold change and proportion of marked spots increased, it was easier for SPADE to identify SV genes between groups.

To validate the robustness of SPADE, we assessed its performance to detect SV genes between groups with different spots coordinates or different pattern directions. With the streak pattern, we found that SPADE provided lower but close AUC in groups with different coordinates compared to those with the same coordinates (Figure B.2-B.4A). This showed that the performance of our method was not strongly affected by two groups with inconsistent spots locations. Moreover, similar performances with small FDR values were given by SPADE in groups with different signal directions compared to those with the same signal directions. This suggested that SPADE was robust to the changes of pattern directions and successfully indicated genes from two groups with similar pattern shapes but with different directions as non-SV genes. A similar performance was observed in the hotspot pattern for SPADE to identify SV genes between groups (Figure B.2-B.4B).

The performance of SPADE to detect SV genes between groups was further evaluated using a real dataset of ARRISTA which included groups of two different post injury

states (i.e., 2DPI, 5DPI) from brain tissue of axolotl. SV genes between these two states identified using the SPADE method were compared to pre-selected cell type-specific markers. In a total of 2,020 genes, 481 SV genes (e.g., *CDH1D*, *PDE2A*) between the 2DPI and 5DPI stages were identified by the SPADE method, of which 116 SV genes were included in the cell type marker list (Figure 3.6). A higher proportion (21.1%) of SV genes from SPADE were cell type-specific markers compared to genes (5.7%) that were

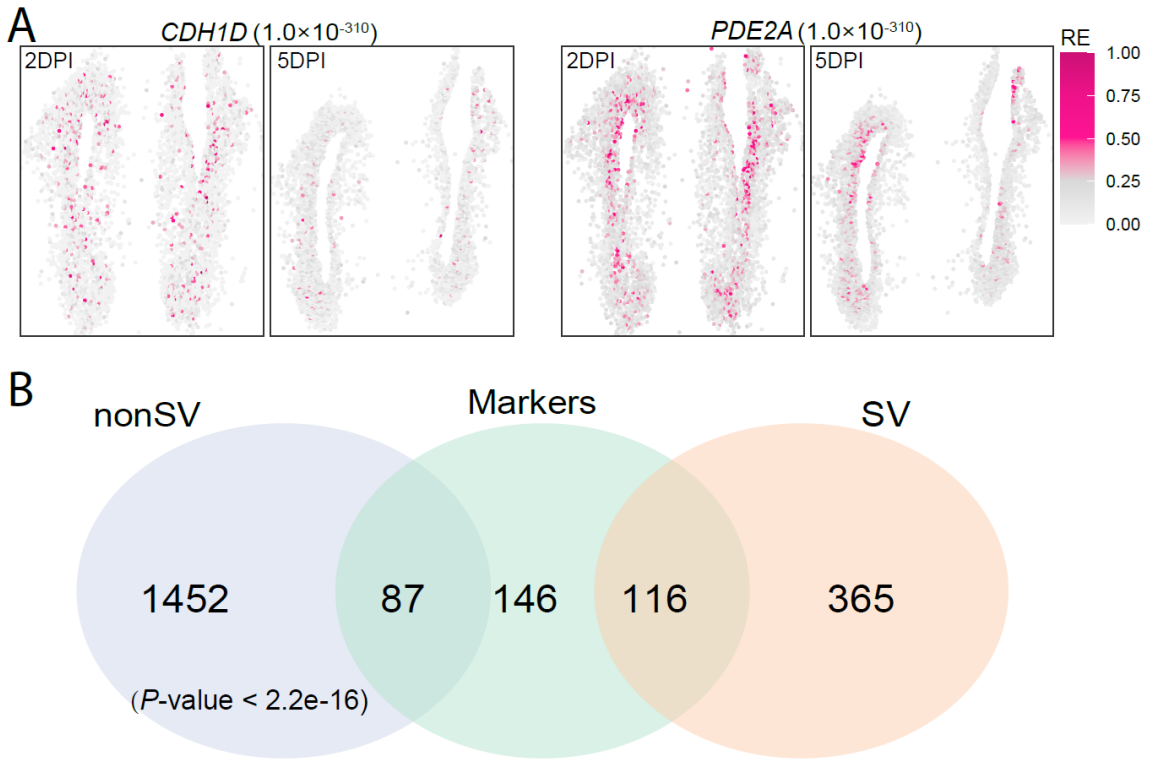

Figure 3.6 Assessment of SPADE to identify spatially variable genes between groups using real data analyses. We evaluated the performance of SPADE to identify spatially variable (SV) genes between groups using a real spatial transcriptomic dataset with axolotl telencephalon (ARRISTA). Two different post injury stages (i.e., 2DPI, 5DPI) were included in the analysis. SV genes between these two groups were identified using the SPADE method and then compared to pre-selected cell type identity markers from its original study. (A) Representative SV genes identified by SPADE are shown, with *P* values from SPADE inside parentheses. (B) Pearson's Chi-square test was utilized for the evaluation of the association between SV genes from SPADE and those "true" markers.

not identified by the SPADE method (P-value $< 2.2 \times 10^{-16}$). This partially indicated that SV genes from SPADE can contribute to the classification of cell types and will be useful to better understand the development of brain tissue in axolotl.

3.4 Discussion

Spatial transcriptomics has enabled us to study transcriptomes in the context of cellular organization, making significant progress in various biomedical domains (Ståhl *et al.*, 2016; Fawkner-Corbett *et al.*, 2021). With data generated using such technologies, accurate identification of genes that display spatial expression patterns is essential for understanding tissue landscape and biological processes. In this study, we developed an accurate spatial pattern and differential expression analysis method, SPADE, to identify genes with spatially variable patterns in spatial transcriptomic data. The length-scale hyperparameter in the kernel function was optimized for each gene to improve the accuracy of SV gene identification. In addition to identifying SV genes within groups, SPADE also provided a framework to detect SV genes between groups from different treatment conditions or time phases. Through extensive simulations and real data analyses, we demonstrated the desirable performance of our method to investigate SV genes both within and between groups.

The emergence of spatial transcriptomics has provided numerous analysis tools to investigate cellular structures in tissues. To overcome the limited resolutions of spatial transcriptomic data, SPOTlight (Elosua-Bayes *et al.*, 2021) and RCTD (Cable *et al.*, 2022) decomposed cell type mixtures in spatial transcriptomics by incorporating scRNA-seq data and correcting for difference across sequencing technologies. SSAM (Park *et al.*,

2021) and spage2vec (Partel and Wählby, 2021) conducted subcellular structure analyses using high-resolution spatial transcriptomic data for more accurate cell type inference and tissue characterization. To study cellular interactions, Giotto investigated how adjacent cell types in spatial transcriptomic data affected the co-expression of known ligand-receptor pairs (Dries *et al.*, 2021). Our method provides an accurate differential expression analysis method based on a GPR model with a Gaussian kernel to identify SV genes in spatial transcriptomic data. Besides identifying SV genes within groups, our method is the first method to compare patterns between groups and identify genes with differential spatial patterns. Comparing two spatial patterns is always challenged by similar mean measures, diverse spots coordinates, and signal directions in these two patterns. Simulations in our study have demonstrated the superior performance of our method to detect SV genes between groups, comparing to existing DE analysis methods. This suggests that SPADE relies on the spatial pattern difference instead of mean expression difference between groups to identifying SV genes. Moreover, with simulation studies, the performance of SPADE was found to be robust to changes in spots coordinates and signals directions. In a real dataset of axolotl that included two different post-injury states, we successfully identified numerous SV genes between these two groups. In the future, as a growing number of data are generated for spatial transcriptomics, our method will be widely applied to discover more biomarkers between different treatment conditions or time phases.

As SPADE estimate parameters for each gene, computational time can be challenging for modeling large dataset with spatial information (e.g., > 5,000 spots). For computational speed (Table B.1), with 249 genes and 131 spots in the SeqFISH dataset,

the time required for SPADE to detect SV genes was 1.16 mins, which was longer than that of SPARK (1.08 mins) and MERINGUE (0.01 mins) but shorter than that of SpatialDE (1.68 mins) at a desktop workstation with an Intel Core i5 CPU 2.6 GHz processor and 8.0 GB of RAM. Nevertheless, given the improved performance of SPADE to identify SV genes, we consider the computational time acceptable for small/medium datasets. For large datasets, advanced computational infrastructure such as high-performance computing could be utilized to largely alleviate this issue.

In general, our SPADE method has provided great insights into the investigation of differential expression patterns in spatial transcriptomic data. Our method has the potential to be extended by incorporating other tissue information (e.g., histopathological images, temporal information) into the model for more accurate inference of significant gene markers. Histopathological images provide high-resolution cellular information, which will be helpful to enhance the spatial expression patterns (Shan *et al.*, 2022). Incorporating temporal information can contribute to the discovery of spatial expression patterns associated with dynamic changes of cell states and types (Ding *et al.*, 2022).

Chapter 4: Software development and application

We developed R packages for the FLCNA method in Chapter 2 and the SPADE method in Chapter 3, respectively. Both packages have been submitted to the GitHub platform, which is a provider of internet hosting for software development. In particular, the FLCNA R package can be found in https://github.com/FeifeiXiaoUSC/FLCNA, and the SPADE R package is available in https://github.com/thecailab/SPADE.

4.1 Development of R package for the FLCNA method

4.1.1 Software development

To develop the FLCNA R package, the *devtools* package created by Hadley Wickham was used to simplify the building process. Based on devtools:create("FLCNA") function, we began by creating a folder for the FLCNA package, which contains four files, namely DESCRIPTION, FLCNA.Rproj, NAMESPACE, and R. In the DESCRIPTION document, important metadata about this package were provided such as Name, Title, Version, Authors, License, Description about the tool, *et al.*. The FLCNA.Rproj file was specific to R studio, while the NAMESPACE file summarized the information that needed to be exposed to other users. The R file contained the subfolder where the R scripts were stored. All R functions in our package were documented with the help of the *roxygen2* package. Another important file in an R package was Vignettes, which

was generated using R Markdown tool to help users quickly understand how to apply the FLCNA tool. After building all these files, the FLCNA folder was submitted to GitHub platform and the method was tested for executability. The FLCNA package was created based on R software version 4.1.0.

4.1.2 Application of the FLCNA package

FLCNA is a CNA detection method based on the fused lasso model, which can simultaneously identify subclones with scDNA-seq data. In this section, I provide a step-by-step guide to easily understand and apply the FLCNA package with seven significant components including Data pre-processing, Data input, Quality control, Data normalization, Parameter estimation, Data output, and CNA clustering.

Data pre-processing

For public data from NCBI SRA (e.g., TNBC), starting with SRA files, FASTQ files were generated with Fastq-dump from SRA-Toolkit (Leinonen *et al.*, 2011), and then aligned to NCBI hg19 reference genome and converted to BAM files. For the 10× Genomics datasets (e.g., BRCA5), we demultiplexed the original integrated BAM file into separate BAM files using a python script *split_script.py*, which was stored in the package. Raw read counts data, GC content and mappability were obtained from the BAM files using R functions from the SCOPE package (Wang *et al.*, 2020). Different bin sizes (100 kb in our study) could be predefined in the R function to generate raw read depth of coverage data.

Input data

Users need to input data that include following components:

(1) Read counts data matrix: a data matrix for scDNA-seq data. Each row is a cell, and each column is a bin.

(2) Number of clusters: a single number/vector to provide candidate numbers of clusters.

(3) Tuning parameter lambda: a single number or a vector to provide candidate tuning parameters in the penalty term of FLCNA model, which controls the number of breakpoints for CNA detection. The default value is set at three.

(4) N: The maximum number of iterations in the EM algorithm with default value of 100.

To better illustrate how to use the FLCNA package, the KTN126 patient in the TNBC dataset Details in Section 2.2.5 Data description) with breast cancer was utilized in this vignette. In the KTN126 patient, cells were sequenced at two time points with 93 cells (46 pre- and 47 post-treatment). To simplify computation, only the first 3,000 markers were included in the example data (Figure 4.1), which have been stored in the package.

```
# The example data have 3,000 markers and 93 cells.
library(FLCNA)
data(KTN126_data_3000)
data(KTN126_ref_3000)
RD <- KTN126_data_3000
dim(RD)

## [1] 3000   93

ref <- KTN126_ref_3000
head(ref)

## GRanges object with 6 ranges and 2 metadata columns:
##       seqnames        ranges strand |        gc      mapp
##          <Rle>     <IRanges>  <Rle> | <numeric> <numeric>
##   [1]     chr1      1-100000      * |     38.21  0.275225
##   [2]     chr1 100001-200000      * |     33.76  0.119393
##   [3]     chr1 200001-300000      * |     15.78  0.141733
##   [4]     chr1 300001-400000      * |     33.08  0.115785
##   [5]     chr1 400001-500000      * |     32.38  0.155596
##   [6]     chr1 500001-600000      * |     35.18  0.206961
##   -------
##   seqinfo: 24 sequences from hg19 genome
```

Figure 4.1 Example data in the FLCNA package

Quality control

FLCNA_QC() R function (Figure 4.2) was used to remove bins that have extreme GC content (less than 20% and greater than 80%) and low mappability (less than 0.9) to reduce artifacts. After this step, only 2,572 markers were kept for data normalization and parameter estimation.

```
QCobject <- FLCNA_QC(Y_raw=RD, ref_raw=ref,
                     mapp_thresh = 0.9,
                     gc_thresh = c(20, 80))

## Excluded 233 bins due to extreme GC content.

## Excluded 210 bins due to low mappability.

## There are 93 samples and 2572 bins after QC step.
```

Figure 4.2 Quality control function in the FLCNA package

Data normalization

A two-step median normalization approach was implemented to remove the effect of biases from the GC-content and mappability. We further calculated the ratio of normalized RC and its sample specific mean, and the logarithm transformation of this ratio (*log2R*) was used in the main step of the FLCNA method. FLCNA_normalization() R function (Figure 4.3) was used for the normalization.

```
log2Rdata <- FLCNA_normalization(Y=QCobject$Y,
                                 gc=QCobject$ref$gc,
                                 map=QCobject$ref$mapp)
```

Figure 4.3 Normalization function in the FLCNA package

We used a fused lasso model to construct change points detection and subclone clustering simultaneously. Parameters were estimated using an EM algorithm with the FLCNA() R function (Figure 4.4), including mean vector for each cluster, covariance

matrix, and “weights” capturing the contribution of each cluster for each cluster. Optimal number of clusters and optimal tuning parameter lambda were selected using a BIC criterion with the lowest BIC value.

```
# K: The number of clusters.
# lambda: The tuning parameter in the penalty term, the default is 3.
output_FLCNA <- FLCNA(K=c(3,4,5), lambda=3, Y=log2Rdata)
```

Figure 4.4 Parameter estimation function in the FLCNA package

Output data

The output of FLCNA() function (Figure 4.5) includes:

(1) mu.hat.best: The estimated mean matrix in the optimal model. This matrix consists of mean values estimated from the FLCNA model for each cluster and each marker.

(2) sigma.hat.best: The estimated covariance matrix in the optimal model.

(3) p.hat.best: A vector shows the estimated cluster proportions in the optimal model. The length of the vector is the optimal number of clusters selected using BIC criterion.

(4) s.hat,best: Clustering index for each cell in the optimal model.

(5) lambda.best: The optimal lambda value selected from candidate lambda values (Input data) according to the BIC criterion.

(6) K.best: The optimal number of clusters selected from candidate numbers of clusters (Input data) according to the BIC criterion.

(7) bic.best: The BIC value of the optimal model.

```
# The number of clusters in the optimal model
output_FLCNA$K.best

## [1] 4

# The estimated mean matrix for K clusters
output_FLCNA$mu.hat.best[,1:11]

##            [,1]       [,2]       [,3]       [,4]       [,5]        [,6]
## [1,] -1.3428226 -0.1932012 -0.1931969 -0.3443138 -0.8830260 -0.11986545
## [2,] -1.2005804 -0.3594904 -0.3594859 -0.6050274 -0.6075824 -0.25178873
## [3,] -1.0563435 -0.2089481 -0.2089425 -0.4364831 -0.7848446 -0.04731857
## [4,] -0.5065039 -0.5040336 -0.3992638 -0.3992600 -0.3992577 -0.39925142
##             [,7]        [,8]        [,9]       [,10]       [,11]
## [1,] -0.09089903 -0.09089832  0.11729189  0.11729300  0.15844515
## [2,] -0.04883023 -0.04882879 -0.04882691 -0.04882715 -0.04882654
## [3,] -0.04731449 -0.04731633  0.19133971  0.19133986  0.19133914
## [4,] -0.39925090 -0.39925766 -0.04146510 -0.04146562 -0.04243215

# The cluster index for each cell
output_FLCNA$s.hat.best[1:20]

##  [1] 1 1 1 1 1 1 1 1 1 2 1 1 2 4 3 2 2 2 1

table(output_FLCNA$s.hat.best)

##
##  1  2  3  4
## 62 18 11  2
```

Figure 4.5 Parameter estimation output for the FLCNA package

CNA clustering

After the mean vector was estimated for each cluster, we located and quantified all the shared changes points in each cluster, and identified segments that shared the same underlying copy number profile. CNA.out() R function (Figure 4.6) was used for the clustering of candidate CNAs. Change-points were identified from the estimate of mean vector where for the marker before and after the change point showed different values. Typically, different CNA states were required to be assigned for each segment to help locate significant CNA signatures. For each cluster, to assign the most likely copy number state for each segment and each cell, we further implemented a GMM-based

clustering strategy for CNA clustering based on normalized data *log2R*. Each segment was classified using a five-state classification scheme with double deletion (Del.d), single deletion (Del.s), normal/diploid, single duplication (Dup.s) and double duplication (Dup.d).

```
# mean.matrix: The cluster mean matrix estimated from FLCNA R function.
# cutoff: Cutoff value to further control the number of CNAs,
#         the larger cutoff, the smaller number of CNAs.
#         The default is 0.35.
# L: Repeat times in the EM algorithm, defaults to 100.
CNA.output <- CNA.out(mean.matrix = output_FLCNA$mu.hat.best, LRR=log2Rdata,
                      Clusters=output_FLCNA$s.hat.best, ref=ref,
                      cutoff=0.35, L=100)
head(CNA.output)

##   sampleID samplename Cluster  chr start end start.coor end.coor
width_bins
## 1        1 SRR5964097       1 chr1     2   6     100001   600001
5
## 2        1 SRR5964097       1 chr1     6  14     500001  1400001
9
## 4        1 SRR5964097       1 chr1    15  27    1400001  2700001
13
## 5        1 SRR5964097       1 chr1    81 140    8000001 14000001
60
## 6        1 SRR5964097       1 chr1   212 214   21100001 21400001
3
## 7        1 SRR5964097       1 chr1   473 700   47200001 70000001
228
##   state
## 1 Del.d
## 2 Del.s
## 4 Del.s
## 5 Del.s
## 6 Del.d
## 7 Del.s
```

Figure 4.6 CNA clustering in the FLCNA package

4.2 Development of R package for the SPADE method

We also developed the R package SPADE to facilitate the identification of spatially variable genes in spatial transcriptomic data. The technical points for building this R package were similar to those of the FLCNA package discussed in Section 4.1. The

SPADE package consists of two main modules including identifying SV genes within groups and between groups.

4.2.1 Identifying spatially variable genes within groups

SPADE was developed based on a GPR model with a Gaussian kernel to detect SV genes in spatial transcriptomic data. I would describe this package with five important components including Input data, Normalization, Parameter estimation, Testing, and Output data.

Input data

To help illustrate how SPADE package could be applied to identify SV genes within groups, the SeqFISH dataset was provided in the package, which was collected on the mouse hippocampus with 249 genes measured on 131 spots (details are shown in Section 3.2.4 Data description). Two data files were given in the package including read counts data and coordinates information (Figure 4.7). Users need to input data that include:

(1) Read counts data matrix: a data matrix for spatial transcriptomics data. Each row is a gene, and each column is an observation (e.g., a cell).

(2) Coordinate information: a data frame with two columns (i.e., vertical location, horizontal location) to show the two-dimensional locations of spots. The number of columns in the read counts data needs to be consistent with the number of rows in coordinates data.

```
library(SPADE)
data(SeqFISH)
data(info)
readcounts <- SeqFISH

dim(readcounts)

## [1] 249 131

dim(info)

## [1] 131   2

readcounts[1:5,1:5]

##        C1 C2 C3 C4 C5
## Tal1    3  3 17 20  6
## Dmbx1  11  9  1  4  0
## Emx2   13 14  2  9  0
## Uncx    5 21  3  5  4
## Paxip1 15  9 10 13  7

head(info)

##      x   y
## C1 686 352
## C2 488 572
## C3 614 698
## C4 516 726
## C5 308 208
## C6 204 225
```

Figure 4.7 Example data in the SPADE package for within-groups SV gene identification

Normalization

A two-step normalization strategy was implemented to transform original read counts data into continuous data. First, the mean-variance dependency was stabilized using Anscombe's transformation strategy. Second, a linear regression was utilized to regress out library size from the above transformed expression data. After this step, the normalized continuous data were utilized for the parameter estimation and hypothesis testing in the SPADE method. SPADE_norm() R function (Figure 4.8) was used for the normalization.

```
data_norm <- SPADE_norm(readcounts=as.matrix(readcounts), info=info)
```

Figure 4.8 Normalization function in the SPADE package

Parameter estimation

The GPR model was used to model the expression of each gene across cells with different locations. Theoretically, the problem of finding SV genes in the SPADE method was to test how well the candidate covariance matrix fits the spatial transcriptomic data. For each gene, we selected the optimal length-scale hyperparameter in the Gaussian kernel to increase the accuracy of SV gene identification. The kernel with the optimal hyperparameter value was used in covariance matrix to test whether candidate gene was the significant gene with spatially variable pattern. SPADE_estimate() R function (Figure 4.9) was used for the parameter estimation. The output of SPADE_estimate() includes

(1) GeneID: Gene index.

(2) theta_Gau: Optimal length-scale hyperparameter in kernel function for each gene.

(3) Lik_Gau: Maximum log likelihood computed for each gene.

```
Est <- SPADE_estimate(expr_data=data_norm, info=info)

head(Est)

##   GeneID theta_Gau   Lik_Gau
## 1      1  11.22829 -91.34812
## 2      2  32.89034 -87.74524
## 3      3 130.70081 -83.34457
## 4      4  29.62392 -77.91255
## 5      5  44.91683 -67.52400
## 6      6  58.66151 -63.95050
```

Figure 4.9 Parameter estimation function in the SPADE package

Testing

After the optimal length-scale hyperparameter was estimated, *P*-value for each gene was computed based on a quadratic score statistic with the Davies method. SPADE_test() R function (Figure 4.10) was used for computing a *P*-value for each gene.

```
Test_res <- SPADE_test(object=data_norm, location=info, para=Est)

Test_res[c(1, 230),]

##     geneid         Q    Pvalue Adjust.Pvalue
## 1     Tal1  65.77883 0.4434016     0.5390583
## 230   lyve 487.63637 0.0000000     0.0000000

sum(Test_res$Adjust.Pvalue < 0.05)

## [1] 38
```

Figure 4.10 Test and output in the SPADE package to identify SV genes within groups

Output data

The output of SPADE_test() includes

(1) geneid: Gene name.

(2) Q: A score statistic used to calculate the *P* value. Q follows a mixture of independent chi-square distributions with mixing weights that depend on the eigenvalues of the kernel matrix.

(3) Pvalue: Based on a chi-square mixture distribution and Q value, an exact method based on the Davies method is utilized to compute *P*-values for genes to define SV genes.

(4) Adjusted.Pvalue: *P*-values across all genes are adjusted with the Benjamini and Hochberg method to correct for occurrence of false positives.

4.2.2 Identifying spatially variable genes between groups

SPADE was the first method to investigate SV genes between groups with spatial transcriptomic data. The identification of SV genes between groups was essential for understanding the changes of spatial patterns with different treatment conditions or different time phases. We illustrated the application of the SPADE R package to identify SV genes between groups with four components, including Input data, Normalization, Parameter estimation and testing, and Output data.

Input data

To illustrate how SPADE can be applied to identify SV genes between groups. A real spatial transcriptomic dataset (Section 3.2.4) with axolotl telencephalon (i.e., ARRISTA) was provided in the package. Two different post injury stages (i.e., 2DPI, 5DPI) were included in the dataset. The example data (Figure 4.11) contain three genes with 1,188 spots and 938 spots in the 2DPI group and 5DPI group, respectively.

```
data(D2_data)
data(D2_info)
data(D5_data)
data(D5_info)
dim(D2_data)

## [1]    3 1188

dim(D2_info)

## [1] 1188    2

dim(D5_data)

## [1]   3 938

dim(D5_info)

## [1] 938   2
```

Figure 4.11 Examples data in the SPADE package for between-groups SV gene identification

Normalization

Still, the two-step normalization strategy was implemented for read counts data from each group to transform original data into continuous data. Still, SPADE_norm() R function (Figure 4.12) was used for the normalization.

```
D2_norm <- SPADE_norm(readcounts=as.matrix(D2_data), info=D2_info)
D5_norm <- SPADE_norm(readcounts=as.matrix(D5_data), info=D5_info)
```

Figure 4.12 Normalization in the SPADE package for between-groups SV gene identification

Parameter estimation and testing

SPADE identified SV genes between groups based on a crossed likelihood-ratio test in spatial transcriptomic data. SPADE first estimated the optimal hyperparameter for kernel matrix in each group, respectively. Thus, for each gene, the log likelihood in each group was easily calculated using its optimal kernel matrix. Then we exchanged the estimated hyperparameters to compute the log likelihoods for both groups, and compared them to their original optimal log likelihoods. The likelihood ratio test statistic was calculated to identify SV genes with *P*-values computed using F test with degree freedom of one. SPADE_DE() R function (Figure 4.13) was utilized for the SV gene identification with two groups. I have shown patterns of these three genes below from the 2DPI (above) and the 5DPI (below) group (Figure 4.14).

Output data

The output of SPADE_DE() function includes:

(1) geneid: Gene names.

```
res <- SPADE_DE(D2_norm, D5_norm, D2_info, D5_info)

## NO. Gene =  1
## NO. Gene =  2
## NO. Gene =  3

res

##               geneid theta_Gau1 theta_Gau2   logLik11  logLik21
logLik10
## 1 AMEX60DDU001010113   49.52012   70.40230   345.6043  343.4853
314.6583
## 2 AMEX60DDU001038720   49.63787   81.30095 -1407.1231 -574.7335 -
1407.1231
## 3 AMEX60DDU001022818   49.63787   50.10993   143.7086  329.6725
143.7047
##    logLik20         Diff       Pvalue Adjust.Pvalue
## 1  337.0464 74.769707700 5.289511e-18  1.586853e-17
## 2 -588.6931 27.919286739 1.264827e-07  1.897241e-07
## 3  329.6725  0.007881462 9.292587e-01  9.292587e-01
```

Figure 4.13 Testing procedure in the SPADE package for between-groups SV gene identification

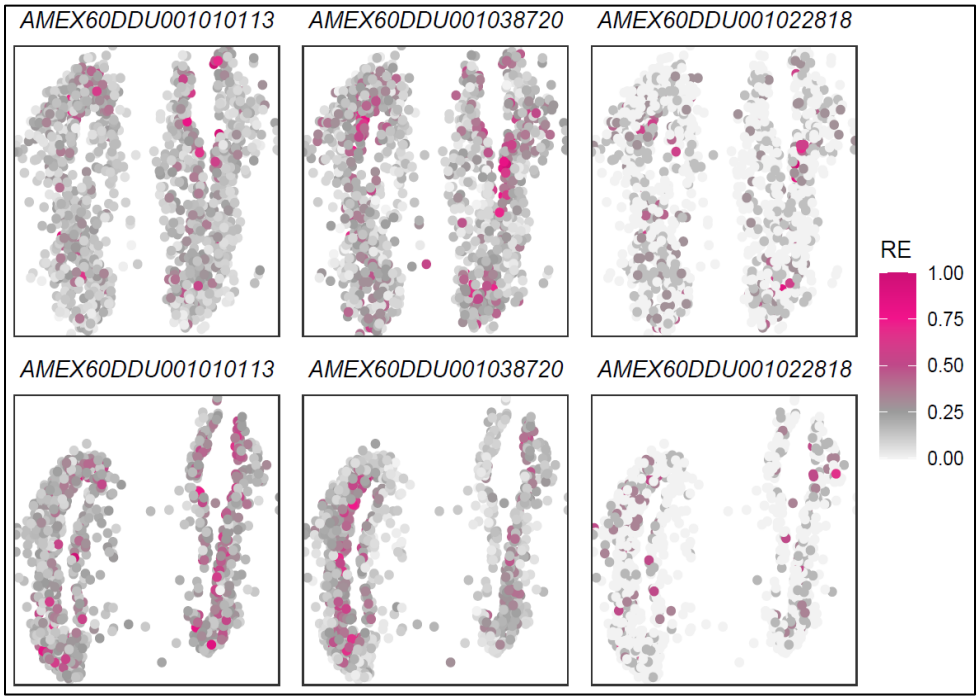

Figure 4.14 Spatial pattern in the example data for between-groups SV gene identification

(2) theta_Gau1: Optimal length-scale hyperparameter estimated for group 1.

(3) theta_Gau2: Optimal length-scale hyperparameter estimated for group 2.

(4) logLik11: Log likelihood calculated for group 1 using parameters from itself.

(5) logLik21: Log likelihood calculated for group 2 using parameters from itself.

(6) logLik10: Log likelihood calculated for group 1 using parameters from group 2.

(7) logLik20: Log likelihood calculated for group 2 using parameters from group 1.

(8) Diff: Log likelihood ratio statistic calculated with 2*(logLik11+logLik21-logLik10-logLik20).

(9) Pvalue: *P*-values computed using F test with degree freedom of one.

(10) Adjust.Pvalue: *P*-values across all genes are adjusted with the Benjamini and Hochberg method to correct for occurrence of false positives.

Chapter 5: Conclusions and future perspectives

The technological advances in SCS have enabled us to explore the genomic and transcriptomic heterogeneity among individual cells (Wen and Tang, 2018). With the increasing data generated from these technologies, a growing number of analysis methods have been developed to explore various biomedical fields such as cancer biology (Zhang *et al.*, 2021), neuroscience (Close *et al.*, 2021), and immunology (Cao *et al.*, 2020). However, methods to analyze SCS data are still in their early stages. This dissertation presents two novel methods for the statistical analysis of SCS data.

First, we introduced FLCNA, a subclone clustering method for scDNA-seq data. FLCNA utilizes a Gaussian mixture model combined with a fused lasso penalty term to cluster subclone and simultaneously detect CNAs. Extensive simulations demonstrated the superior performance of FLCNA in clustering subclones and detecting CNAs with scDNA seq data, particularly in data with a large proportion of shared CNAs. Using a real dataset of breast cancer, we observed clearly distinct cluster patterns based on genomic variants. We also implicated various critical pathways related to breast cancer etiology by matching the identified CNAs from FLCNA to significant genes identified using traditional GWAS studies.

Second, we developed SPADE, a spatial pattern and differential expression analysis method based on a GPR model with a Gaussian kernel to identify SV genes in spatial transcriptomic data. SPADE estimates the optimal hyperparameter value for each gene to

improve the accuracy of identifying SV genes. In addition to identifying genes with differential expression patterns within groups, SPADE provides a framework for detecting SV genes between different treatment conditions or time phrases. Numerous simulation studies and real data analyses have shown that SPADE is powerful for identifying SV genes in spatial transcriptomic data.

In summary, our FLCNA method provides an alternative tool for accurate subclone clustering in SCS data instead of using the traditional two-step protocol (i.e., CNA detection, subclone clustering), which has the drawback of generating false-positive CNAs in the first copy number profiling step. With the emerging spatial transcriptomic technologies, the SPADE method provides a new option to investigate spatial variable patterns within or between groups. R packages have been generated for both methods to facilitate their application. These methods will provide great insights into the understanding of the intra-tumor heterogeneity, tissue landscapes, tumor progression, and therapy outcome.

In the future, our method for subclone and CNA detection has the potential to be used as a normalization step for copy number variation (CNV) detection in other types of data (e.g., haploid species), especially those lacking reference samples. In studies about CNV detection without references, typical ways to find references are choosing a few samples with smaller variation of intensities across markers (e.g., SCOPE (Wang *et al.*, 2020)) or using the other samples from the same batch (e.g., SCCNV (Dong *et al.*, 2020)) for normalization. However, these preselected samples are not always true references and many common CNVs might be neutralized while using the whole genome of these "reference" samples for normalization. To address this issue, our FLCNA method can be

used to generate bin-based references in the normalization step, ultimately improving the performance of CNV detection. Specifically, we can utilize CNA estimation results from FLCNA to select reference bins for each bin instead of using the whole genome as reference. However, it might be challenging to consider this strategy in data with no apparent subclone patterns since FLCNA was developed to identify subclones in SCS data.

Second, besides scDNA-seq data, more types of data (e.g., scRNA-seq, protein, spatial transcriptomics) from the same sample are expected to be provided for the same cohort studies in the future. Thus, our subclone clustering method can be extended by integrating omics data in the model to provide more accurate subclone clustering results. For example, cellular location information from the spatial transcriptomics data will contribute to the clustering of subclone in SCS data as adjacent cells are more likely to belong to the same subclone (Elyanow *et al.*, 2021). One of the challenges is to incorporate the coordinates of cells into the model of clustering with SCS data. The GPR model with Gaussian kernel utilized in our SPADE method might be helpful. Another issue is that different types of data might provide inconsistent subclone clustering results, which makes it challenging for integrative analysis or results interpretation.

Finally, our SPADE method in Chapter 3 can also be extended to consider other imaging data or temporal information in the model for identifying SV genes with spatial transcriptomic data. The matched histopathological images are always generated together with the sequencing-based spatial transcriptomics data for the same tissue sample, which can provide additional cellular information to enhance the gene expression patterns (Shan *et al.*, 2022). The integration of matched transcriptomics and image data is expected to

improve the accuracy of SV identification. However, this integrative analysis remains challenging in modeling and computation due to the different formats, dimensions, and noise level of these data. In addition to spatial heterogeneity, the gene expression in developmental tissue (e.g., embryonic heart) exhibit high levels of temporal variation (Velten *et al.*, 2022). The inclusion of temporal information for investigating expression patterns uncovered the dynamic changes of molecular events during development process in tissues (Chou *et al.*, 2016). Still, the complexity in space, time, and gene expression for modelling posed a great challenge in extracting significant genes from the data.

References

Al-Sukhni,W. *et al.* (2012) Identification of germline genomic copy number variation in familial pancreatic cancer. *Hum. Genet.*, **131**, 1481–1494.

Almet,A.A. *et al.* (2021) The landscape of cell-cell communication through single-cell transcriptomics. *Curr. Opin. Syst. Biol.*, **26**, 12–23.

Anderson,D. (1974) Algorithms for minimization without derivatives. *IEEE Trans. Automat. Contr.*, **19**, 632–633.

ANSCOMBE,F.J. (1948) THE TRANSFORMATION OF POISSON, BINOMIAL AND NEGATIVE-BINOMIAL DATA. *Biometrika*, **35**, 246–254.

Asadollahi,R. *et al.* (2014) The clinical significance of small copy number variants in neurodevelopmental disorders. *J. Med. Genet.*, **51**, 677–688.

Atta,L. and Fan,J. (2021) Computational challenges and opportunities in spatially resolved transcriptomic data analysis. *Nat. Commun.*, **12**, 5283.

Baslan,T. *et al.* (2020) Novel insights into breast cancer copy number genetic heterogeneity revealed by single-cell genome sequencing. *Elife*, **9**.

Baslan,T. and Hicks,J. (2017) Unravelling biology and shifting paradigms in cancer with single-cell sequencing. *Nat. Rev. Cancer*, **17**, 557–569.

Bridges,C.C. (1966) Hierarchical Cluster Analysis. *Psychol. Rep.*, **18**, 851–854.

Burandt,E. *et al.* (2013) Prognostic relevance of AIB1 (NCoA3) amplification and overexpression in breast cancer. *Breast Cancer Res. Treat.*, **137**, 745–753.

Cable,D.M. *et al.* (2022) Robust decomposition of cell type mixtures in spatial transcriptomics. *Nat. Biotechnol.*, **40**, 517–526.

Cai,X. *et al.* (2014) Single-cell, genome-wide sequencing identifies clonal somatic copy-number variation in the human brain. *Cell Rep.*, **8**, 1280–1289.

Cao,Y. *et al.* (2020) Single-cell RNA Sequencing in Immunology. *Curr. Genomics*, **21**, 564–575.

Cariati,F. *et al.* (2019) Dissecting Intra-Tumor Heterogeneity by the Analysis of Copy Number Variations in Single Cells: The Neuroblastoma Case Study. *Int. J. Mol. Sci.*, **20**.

Castellani,C.A. *et al.* (2014) Copy number variation distribution in six monozygotic twin pairs discordant for schizophrenia. *Twin Res. Hum. Genet. Off. J. Int. Soc. Twin Stud.*, **17**, 108–120.

Chen,A. *et al.* (2022) Spatiotemporal transcriptomic atlas of mouse organogenesis using DNA nanoball-patterned arrays. *Cell*, **185**, 1777-1792.e21.

Chen,Z. *et al.* (2020) RobustClone: a robust PCA method for tumor clone and evolution inference from single-cell sequencing data. *Bioinformatics*, **36**, 3299–3306.

Chou,S.-J. *et al.* (2016) Analysis of spatial-temporal gene expression patterns reveals dynamics and regionalization in developing mouse brain. *Sci. Rep.*, **6**, 19274.

Close,J.L. *et al.* (2021) Spatially resolved transcriptomics in neuroscience. *Nat. Methods*, **18**, 23–25.

D'Aurizio,R. *et al.* (2016) Enhanced copy number variants detection from whole-exome sequencing data using EXCAVATOR2. *Nucleic Acids Res.*, **44**, e154–e154.

Dagogo-Jack,I. and Shaw,A.T. (2018) Tumour heterogeneity and resistance to cancer therapies. *Nat. Rev. Clin. Oncol.*, **15**, 81–94.

Dai,Z. and Liu,P. (2021) High copy number variations, particular transcription factors, and low immunity contribute to the stemness of prostate cancer cells. *J. Transl. Med.*, **19**, 206.

Davis-Marcisak,E.F. *et al.* (2019) Differential Variation Analysis Enables Detection of Tumor Heterogeneity Using Single-Cell RNA-Sequencing Data. *Cancer Res.*, **79**, 5102–5112.

Dempster,A.P. *et al.* (1977) Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. Ser. B*, **39**, 1–22.

Deshwar,A.G. *et al.* (2015) PhyloWGS: reconstructing subclonal composition and evolution from whole-genome sequencing of tumors. *Genome Biol.*, **16**, 35.

van Dijk,D. *et al.* (2018) Recovering Gene Interactions from Single-Cell Data Using Data Diffusion. *Cell*, **174**, 716-729.e27.

Ding,J. *et al.* (2022) Temporal modelling using single-cell transcriptomics. *Nat. Rev. Genet.*, **23**, 355–368.

Dokmanic,I. *et al.* (2015) Euclidean Distance Matrices: Essential theory, algorithms, and applications. *IEEE Signal Process. Mag.*, **32**, 12–30.

Dong,X. *et al.* (2020) SCCNV: A Software Tool for Identifying Copy Number Variation From Single-Cell Whole-Genome Sequencing. *Front. Genet.*, **11**, 505441.

Dries,R. *et al.* (2021) Giotto: a toolbox for integrative analysis and visualization of spatial expression data. *Genome Biol.*, **22**, 78.

Edsgärd,D. *et al.* (2018) Identification of spatial expression trends in single-cell gene expression data. *Nat. Methods*, **15**, 339–342.

Elosua-Bayes,M. *et al.* (2021) SPOTlight: seeded NMF regression to deconvolute spatial transcriptomics spots with single-cell transcriptomes. *Nucleic Acids Res.*, **49**, e50.

Elyanow,R. *et al.* (2021) STARCH: copy number and clone inference from spatial transcriptomics data. *Phys. Biol.*, **18**, 35001.

Evrony,G.D. *et al.* (2021) Applications of Single-Cell DNA Sequencing. *Annu. Rev. Genomics Hum. Genet.*, **22**, 171–197.

Fan,J. and Li,R. (2001) Variable Selection via Nonconcave Penalized Likelihood and its Oracle Properties. *J. Am. Stat. Assoc.*, **96**, 1348–1360.

Fawkner-Corbett,D. *et al.* (2021) Spatiotemporal analysis of human intestinal development at single-cell resolution. *Cell*, **184**, 810-826.e23.

Ferronika,P. *et al.* (2017) Copy number alterations assessed at the single-cell level revealed mono- and polyclonal seeding patterns of distant metastasis in a small-cell lung cancer patient. *Ann. Oncol. Off. J. Eur. Soc. Med. Oncol.*, **28**, 1668–1670.

Finak,G. *et al.* (2015) MAST: A flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell RNA sequencing data. *Genome Biol.*, **16**, 1–12.

Fischer,K. and Pflugfelder,G.O. (2015) Putative Breast Cancer Driver Mutations in TBX3 Cause Impaired Transcriptional Repression. *Front. Oncol.*, **5**, 244.

Foulkes,W.D. *et al.* (2010) Triple-negative breast cancer. *N. Engl. J. Med.*, **363**, 1938–1948.

Gao,R. *et al.* (2016) Punctuated copy number evolution and clonal stasis in triple-negative breast cancer. *Nat. Genet.*, **48**, 1119–1130.

Garvin,T. *et al.* (2015) Interactive analysis and assessment of single-cell copy-number variations. *Nat. Methods*, **12**, 1058–1060.

Gerlinger,M. *et al.* (2012) Intratumor heterogeneity and branched evolution revealed by multiregion sequencing. *N. Engl. J. Med.*, **366**, 883–892.

Gómez-Rubio,V. (2016) Spatial Point Patterns: Methodology and Applications with R. *J. Stat. Software, B. Rev.*, **75**, 1–6.

Grün,D. and van Oudenaarden,A. (2015) Design and Analysis of Single-Cell Sequencing Experiments. *Cell*, **163**, 799–810.

Guo,J. *et al.* (2010) Pairwise variable selection for high-dimensional model-based clustering. *Biometrics*, **66**, 793–804.

Ha,G. *et al.* (2012) Integrative analysis of genome-wide loss of heterozygosity and monoallelic expression at nucleotide resolution reveals disrupted pathways in triple-negative breast cancer. *Genome Res.*, **22**, 1995–2007.

Ha,G. *et al.* (2014) TITAN: inference of copy number architectures in clonal cell populations from tumor whole-genome sequence data. *Genome Res.*, **24**, 1881–1893.

Haynes,W. (2013) Benjamini--Hochberg Method. In, Dubitzky,W. *et al.* (eds), *Encyclopedia of Systems Biology*. Springer New York, New York, NY, p. 78.

Hou,Y. *et al.* (2016) Single-cell triple omics sequencing reveals genetic, epigenetic, and transcriptomic heterogeneity in hepatocellular carcinomas. *Cell Res.*, **26**, 304–319.

Hu,P. *et al.* (2016) Single Cell Isolation and Analysis. *Front. cell Dev. Biol.*, **4**, 116.

James and others,M. (1967) Some methods for classification and analysis of multivariate observations. *Proc. fifth Berkeley Symp. Math. Stat. Probab.*, **1**, 281–297.

Ji,A.L. *et al.* (2020) Multimodal Analysis of Composition and Spatial Architecture in Human Squamous Cell Carcinoma. *Cell*, **182**, 497-514.e22.

Jiang,Y. *et al.* (2016) Assessing intratumor heterogeneity and tracking longitudinal and spatial clonal evolutionary history by next-generation sequencing. *Proc. Natl. Acad. Sci. U. S. A.*, **113**, E5528-37.

Kanehisa,M. *et al.* (2016) KEGG as a reference resource for gene and protein annotation. *Nucleic Acids Res.*, **44**, D457-62.

Kim,C. *et al.* (2018) Chemoresistance Evolution in Triple-Negative Breast Cancer Delineated by Single-Cell Sequencing. *Cell*, **173**, 879-893.e13.

Lähnemann,D. *et al.* (2020) Eleven grand challenges in single-cell data science. *Genome Biol.*, **21**, 31.

Laks,E. *et al.* (2018) Resource: Scalable whole genome sequencing of 40,000 single cells identifies stochastic aneuploidies, genome replication states and clonal repertoires. *bioRxiv*.

Lasken,R.S. (2007) Single-cell genomic sequencing using Multiple Displacement Amplification. *Curr. Opin. Microbiol.*, **10**, 510–516.

Law,C.W. *et al.* (2014) voom: precision weights unlock linear model analysis tools for RNA-seq read counts. *Genome Biol.*, **15**, R29.

Leinonen,R. *et al.* (2011) The sequence read archive. *Nucleic Acids Res.*, **39**, D19–D21.

Li,B. and Li,J.Z. (2014) A general framework for analyzing tumor subclonality using SNP array and DNA sequencing data. *Genome Biol.*, **15**, 473.

Li,Y. *et al.* (2009) Spatial Linear Mixed Models with Covariate Measurement Errors. *Stat. Sin.*, **19**, 1077–1093.

Liu,B. *et al.* (2012) A functional copy-number variation in MAPKAPK2 predicts risk and prognosis of lung cancer. *Am. J. Hum. Genet.*, **91**, 384–390.

Liu,F. *et al.* (2016) Biomarkers for EMT and MET in breast cancer: An update. *Oncol. Lett.*, **12**, 4869–4876.

Love,M.I. *et al.* (2014) Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biol.*, **15**, 550.

Magi,A. *et al.* (2013) EXCAVATOR: detecting copy number variants from whole-exome sequencing data. *Genome Biol.*, **14**, R120.

Mallory,X.F. *et al.* (2020) Methods for copy number aberration detection from single-cell DNA-sequencing data. *Genome Biol.*, **21**, 208.

Maniatis,S. *et al.* (2021) Spatially resolved transcriptomics and its applications in cancer. *Curr. Opin. Genet. Dev.*, **66**, 70–77.

McCarthy,D.J. *et al.* (2012) Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation. *Nucleic Acids Res.*, **40**, 4288–4297.

Method of the Year 2020: spatially resolved transcriptomics. (2021) *Nat. Methods*, **18**, 1.

Miao,Z. *et al.* (2018) DEsingle for detecting three types of differential expression in single-cell RNA-seq data. *Bioinformatics*, **34**, 3223–3224.

Miles,L.A. *et al.* (2020) Single-cell mutation analysis of clonal evolution in myeloid malignancies. *Nature*, **587**, 477–482.

Miller,B.F. *et al.* (2021) Characterizing spatial gene expression heterogeneity in spatially resolved single-cell transcriptomic data with nonuniform cellular densities. *Genome Res.*, **31**, 1843–1855.

Moffitt,J.R. *et al.* (2018) Molecular, spatial, and functional single-cell profiling of the hypothalamic preoptic region. *Science*, **362**.

Morita,K. *et al.* (2020) Clonal evolution of acute myeloid leukemia revealed by high-throughput single-cell genomics. *Nat. Commun.*, **11**, 5327.

Moschopoulos,P.G. and Canada,W.B. (1984) The distribution function of a linear combination of chi-squares. *Comput. Math. with Appl.*, **10**, 383–386.

Mroz,E.A. and Rocco,J.W. (2013) MATH, a novel measure of intratumor genetic heterogeneity, is high in poor-outcome classes of head and neck squamous cell carcinoma. *Oral Oncol.*, **49**, 211–215.

Navin,N. *et al.* (2011) Tumour evolution inferred by single-cell sequencing. *Nature*, **472**, 90–94.

Navin,N.E. (2014) Cancer genomics: one cell at a time. *Genome Biol.*, **15**, 452.

Navin,N.E. (2015) The first five years of single-cell cancer genomics and beyond. *Genome Res.*, **25**, 1499–1507.

Oesper,L. *et al.* (2013) THetA: inferring intra-tumor heterogeneity from high-throughput DNA sequencing data. *Genome Biol.*, **14**, R80.

Park,J. *et al.* (2021) Cell segmentation-free inference of cell types from in situ transcriptomics data. *Nat. Commun.*, **12**, 3545.

Partel,G. and Wählby,C. (2021) Spage2vec: Unsupervised representation of localized spatial gene expression signatures. *FEBS J.*, **288**, 1859–1870.

Plackett,R.L. (1983) Karl Pearson and the Chi-Squared Test. *Int. Stat. Rev. / Rev. Int. Stat.*, **51**, 59–72.

Polyak,K. (2014) Tumor heterogeneity confounds and illuminates: a case for Darwinian tumor evolution. *Nat. Med.*, **20**, 344–346.

Rasmussen,C.E. and Williams,C.K.I. (2006) Gaussian Processes for Machine Learning The MIT Press.

Ren,X. *et al.* (2018) Understanding tumor ecosystems by single-cell sequencing: promises and limitations. *Genome Biol.*, **19**, 211.

Risso,D. *et al.* (2018) A general and flexible method for signal extraction from single-cell RNA-seq data. *Nat. Commun.*, **9**, 284.

Robinson,M.D. *et al.* (2010) edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, **26**, 139–140.

Rojas,C. and Wahlberg,B. (2014) On change point detection using the fused lasso method.

Ross,E.M. and Markowetz,F. (2016) OncoNEM: inferring tumor evolution from single-cell sequencing data. *Genome Biol.*, **17**, 69.

Runcie,D.E. and Crawford,L. (2019) Fast and flexible linear mixed models for genome-wide genetics. *PLOS Genet.*, **15**, 1–24.

Satija,R. *et al.* (2015) Spatial reconstruction of single-cell gene expression data. *Nat. Biotechnol.*, **33**, 495–502.

Seryakov,A. *et al.* (2021) RNA Sequencing for Personalized Treatment of Metastatic Leiomyosarcoma: Case Report. *Front. Oncol.*, **11**, 666001.

Shah,S. *et al.* (2016) In Situ Transcription Profiling of Single Cells Reveals Spatial

Organization of Cells in the Mouse Hippocampus. *Neuron*, **92**, 342–357.

Shah,S.P. *et al.* (2006) Integrating copy number polymorphisms into array CGH analysis using a robust HMM. *Bioinformatics*, **22**, e431-9.

Shan,Y. *et al.* (2022) TIST: Transcriptome and Histopathological Image Integrative Analysis for Spatial Transcriptomics. *Genomics. Proteomics Bioinformatics*.

Shang,L. and Zhou,X. (2022) Spatially aware dimension reduction for spatial transcriptomics. *Nat. Commun.*, **13**, 7203.

Shannon,P. *et al.* (2003) Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res.*, **13**, 2498–2504.

Shi,S. *et al.* (2020) Expression profile of Toll-like receptors in human breast cancer. *Mol. Med. Rep.*, **21**, 786–794.

Sollis,E. *et al.* (2023) The NHGRI-EBI GWAS Catalog: knowledgebase and deposition resource. *Nucleic Acids Res.*, **51**, D977–D985.

Ståhl,P.L. *et al.* (2016) Visualization and analysis of gene expression in tissue sections by spatial transcriptomics. *Science*, **353**, 78–82.

Stanta,G. and Bonin,S. (2018) Overview on Clinical Relevance of Intra-Tumor Heterogeneity. *Front. Med.*, **5**, 85.

Stickels,R.R. *et al.* (2021) Highly sensitive spatial transcriptomics at near-cellular resolution with Slide-seqV2. *Nat. Biotechnol.*, **39**, 313–319.

Subramani,R. *et al.* (2017) Role of Growth Hormone in Breast Cancer. *Endocrinology*, **158**, 1543–1555.

Subramanian,A. *et al.* (2005) Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proc. Natl. Acad. Sci. U. S. A.*, **102**, 15545–15550.

Sun,S. *et al.* (2020) Statistical analysis of spatial expression patterns for spatially resolved transcriptomic studies. *Nat. Methods*, **17**, 193–200.

Svensson,V. *et al.* (2018) SpatialDE: identification of spatially variable genes. *Nat. Methods*, **15**, 343–346.

Tang,X. *et al.* (2019) The single-cell sequencing: new developments and medical applications. *Cell Biosci.*, **9**, 53.

Tepe,B. *et al.* (2018) Single-Cell RNA-Seq of Mouse Olfactory Bulb Reveals Cellular Heterogeneity and Activity-Dependent Molecular Census of Adult-Born Neurons. *Cell Rep.*, **25**, 2689-2703.e3.

Thrane,K. *et al.* (2018) Spatially Resolved Transcriptomics Enables Dissection of

Genetic Heterogeneity in Stage III Cutaneous Malignant Melanoma. *Cancer Res.*, **78**, 5970–5979.

Tibshirani,R. *et al.* (2005) Sparsity and smoothness via the fused lasso. *J. R. Stat. Soc. Ser. B (Statistical Methodol.*, **67**, 91–108.

Tokumaru,Y. *et al.* (2020) KRAS signaling enriched triple negative breast cancer is associated with favorable tumor immune microenvironment and better survival. *Am. J. Cancer Res.*, **10**, 897–907.

Ton,J.-F. *et al.* (2018) Spatial mapping with Gaussian processes and nonstationary Fourier features. *Spat. Stat.*, **28**, 59–78.

Trapnell,C. *et al.* (2014) The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nat. Biotechnol.*, **32**, 381–386.

Velten,B. *et al.* (2022) Identifying temporal and spatial patterns of variation from multimodal data using MEFISTO. *Nat. Methods*, **19**, 179–186.

Vinh,N.X. *et al.* (2009) Information Theoretic Measures for Clusterings Comparison: Is a Correction for Chance Necessary? In, *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09. Association for Computing Machinery, New York, NY, USA, pp. 1073–1080.

Vogelstein,B. *et al.* (2013) Cancer genome landscapes. *Science*, **339**, 1546–1558.

Wang,K. *et al.* (2007) PennCNV: an integrated hidden Markov model designed for high-resolution copy number variation detection in whole-genome SNP genotyping data. *Genome Res.*, **17**, 1665–1674.

Wang,R. *et al.* (2020) SCOPE: A Normalization and Copy-Number Estimation Method for Single-Cell DNA Sequencing. *Cell Syst.*, **10**, 445-452.e6.

Wang,X. *et al.* (2018) DNA copy number profiling using single-cell sequencing. *Brief. Bioinform.*, **19**, 731–736.

Wang,X. *et al.* (2015) Kernel methods for large-scale genomic data analysis. *Brief. Bioinform.*, **16**, 183–192.

Wang,Y. *et al.* (2014) Clonal evolution in breast cancer revealed by single nucleus genome sequencing. *Nature*, **512**, 155–160.

Weber,L.L. *et al.* (2021) doubletD: detecting doublets in single-cell DNA sequencing data. *Bioinformatics*, **37**, i214–i221.

Wei,X. *et al.* (2022) Single-cell Stereo-seq reveals induced progenitor cells involved in axolotl brain regeneration. *Science*, **377**, eabp9444.

Wen,L. and Tang,F. (2018) Boosting the power of single-cell analysis. *Nat. Biotechnol.*, **36**, 408–409.

Wu,M.C. *et al.* (2011) Rare-variant association testing for sequencing data with the sequence kernel association test. *Am. J. Hum. Genet.*, **89**, 82–93.

Xia,H. *et al.* (2015) A novel framework for analyzing somatic copy number aberrations and tumor subclones for paired heterogeneous tumor samples. *Biomed. Mater. Eng.*, **26 Suppl 1**, S1845-53.

Xiao,F. *et al.* (2019) An accurate and powerful method for copy number variation detection. *Bioinformatics*, **35**, 2891–2898.

Yachida,S. *et al.* (2010) Distant metastasis occurs late during the genetic evolution of pancreatic cancer. *Nature*, **467**, 1114–1117.

Yu,Z. *et al.* (2022) SCClone: Accurate Clustering of Tumor Single-Cell DNA Sequencing Data. *Front. Genet.*, **13**, 823941.

Zaccaria,S. and Raphael,B.J. (2021) Characterizing allele- and haplotype-specific copy numbers in single cells with CHISEL. *Nat. Biotechnol.*, **39**, 207–214.

Zafar,H. *et al.* (2019) SiCloneFit: Bayesian inference of population structure, genotype, and phylogeny of tumor clones from single-cell genome sequencing data. *Genome Res.*, **29**, 1847–1859.

Zappia,L. *et al.* (2017) Splatter: Simulation of single-cell RNA sequencing data. *Genome Biol.*, **18**, 174.

Zhang,Jianjun *et al.* (2014) Intratumor heterogeneity in localized lung adenocarcinomas delineated by multiregion sequencing. *Science*, **346**, 256–259.

Zhang,Y. *et al.* (2021) Single-cell RNA sequencing in cancer research. *J. Exp. Clin. Cancer Res.*, **40**, 81.

Zhao,J. *et al.* (2015) Death-associated protein kinase 1 promotes growth of p53-mutant cancers. *J. Clin. Invest.*, **125**, 2707–2720.

Zhuang,X. (2021) Spatially resolved single-cell genomics and transcriptomics by imaging. *Nat. Methods*, **18**, 18–22.

Zou,H. (2006) The Adaptive Lasso and Its Oracle Properties. *J. Am. Stat. Assoc.*, **101**, 1418–1429.

Zucker,M.R. *et al.* (2019) Inferring clonal heterogeneity in cancer using SNP arrays and whole genome sequencing. *Bioinformatics*, **35**, 2924–2931.

Appendix A: A statistical learning method for simultaneous copy number estimation and subclone clustering with single cell sequencing data

A.1 Parameter estimation using expectation–maximization algorithm

In FLCNA, the parameter set $\boldsymbol{\Theta} = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}, \pi_k\}_{k=1}^{K}$ are estimated using an expectation–maximization (EM) algorithm. let $\Delta_{j,k}$ be an indicator function of the hidden cluster information for $\boldsymbol{x}_j$, $\Delta_{j,k}= 1$ if $\boldsymbol{x}_j$ is from the k-th cluster, and $\Delta_{j,k}= 0$ otherwise. Assuming $\Delta_{j,k}$ is unobserved, the penalized log-likelihood function for the complete data will be given by

$$Q_P(\boldsymbol{\Theta}) = \sum_{j=1}^{N}\sum_{k=1}^{K}\Delta_{j,k}\{\log(\pi_k) + \log f_k(\boldsymbol{x}_j;\ \boldsymbol{\mu}_k, \boldsymbol{\Sigma})\}$$

$$- \lambda \sum_{k=1}^{K}\sum_{i=1}^{P-1}\tau_{i,\,i+1}^{(k)}\left|\mu_{i,k} - \mu_{i+1,k}\right|. \quad (1)$$

With Eq. (1), the parameter set $\boldsymbol{\Theta} = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}, \pi_k\}_{k=1}^{K}$ can be estimated with the EM algorithm by the following iterative procedure. The EM algorithm iterates between E-step and M-step, and produces a sequence of estimates $\widehat{\boldsymbol{\Theta}}^{(t)}$, $t = 0, 1, 2, ...$

1) Initialization: We first estimate the starting values $\widehat{\boldsymbol{\Theta}}^{(0)} = \{\widehat{\boldsymbol{\mu}}_k^{(0)}, \widehat{\boldsymbol{\Sigma}}^{(0)}, \hat{\pi}_k^{(0)}\}_{k=1}^{K}$ using model without penalty (λ=0).

2) Iteration:

E-step:

We start with the E-step given the current parameter estimates $\widehat{\boldsymbol{\Theta}}^{(t)}$. In this step, we calculate the probability for sample j belongs to k -th cluster with

$$\hat{\Delta}_{j,k}^{(t+1)} = E\left(\Delta_{j,k} \middle| \boldsymbol{X}, \widehat{\boldsymbol{\Theta}}^{(t)}\right) = \Pr\left(\Delta_{j,k} = 1 \middle| \boldsymbol{X}, \widehat{\boldsymbol{\Theta}}^{(t)}\right)$$

$$= \frac{\hat{\pi}_k^{(t)} f_k(\boldsymbol{x}_j; \widehat{\boldsymbol{\mu}}_k^{(t)}, \widehat{\boldsymbol{\Sigma}}^{(t)})}{\sum_{k'=1}^{K} \hat{\pi}_{k'}^{(t)} f_k(\boldsymbol{x}_j; \widehat{\boldsymbol{\mu}}_{k'}^{(t)}, \widehat{\boldsymbol{\Sigma}}^{(t)})}, \qquad (2)$$

where the numerator is the density for j-th sample to be clustered into k-th cluster, and the denominator is the sum of densities for j-th sample to be clustered into K different clusters. Then Eq. (2) will be plugged it into the Eq. (1) about $Q_P(\boldsymbol{\Theta})$ to estimate other parameters, including the cluster "weight" π_k, the variance for i-th marker σ_i^2 and cluster mean $\boldsymbol{\mu}$.

M-Step:

Given $\hat{\Delta}_{j,k}^{(t+1)}$ and $\widehat{\boldsymbol{\Theta}}^{(t)}$, the goal of M-step is to update parameter set $\widehat{\boldsymbol{\Theta}}^{(t+1)}$ by maximizing the log-likelihood function $Q_P\left(\boldsymbol{\Theta}, \widehat{\boldsymbol{\Theta}}^{(t)}\right)$. Specifically, the estimate of "weights" $\pi_k's$ can be easily updated by taking the first derivative of $Q_P\left(\boldsymbol{\Theta}, \widehat{\boldsymbol{\Theta}}^{(t)}\right)$ w.r.t. π_k with

$$\frac{\partial Q_P}{\partial \pi_k} = 0 \rightarrow \hat{\pi}_k^{(t+1)}$$

$$= \frac{1}{N} \sum_{j=1}^{N} \hat{\Delta}_{j,k}^{(t+1)}. \qquad (3)$$

Given $\hat{\Delta}_{j,k}^{(t+1)}$, $\hat{\pi}_k^{(t+1)}$ and $\hat{\mu}_{i,k}^{(t)}$, we can update the estimate of variance for i-th marker σ_i^2 by taking the first derivative of $Q_P\left(\boldsymbol{\Theta}, \widehat{\boldsymbol{\Theta}}^{(t)}\right)$ w.r.t. σ_i^2 with

$$\frac{\partial Q_P}{\partial \sigma_i^2} = 0 \rightarrow \left(\hat{\sigma}_i^{(t+1)}\right)^2 = \frac{1}{N}\sum_{j=1}^{N}\sum_{k=1}^{K}\hat{\Delta}_{j,k}^{(t+1)}\left(x_{i,j} - \hat{\mu}_{i,k}^{(t)}\right)^2, 1 \le j \le p. \quad (4)$$

Given $\hat{\Delta}_{j,k}^{(t+1)}$, $\hat{\pi}_k^{(t+1)}$ and $\hat{\sigma}_i^{(t+1)}$, according to Eq. (1), after some transformation, we can update the estimates of mean values $\widehat{\boldsymbol{\mu}}^{(t+1)}$ with

$$\widehat{\boldsymbol{\mu}}^{(t+1)} = \underset{\boldsymbol{\mu}}{\operatorname{argmin}} \frac{1}{2}\sum_{j=1}^{N}\sum_{k=1}^{K}\left\{\hat{\Delta}_{j,k}^{(t+1)}\sum_{i=1}^{p}\frac{\left(x_{i,j} - \mu_{i,k}\right)^2}{\left(\hat{\sigma}_i^{(t)}\right)^2}\right\} + \lambda\sum_{k=1}^{K}\sum_{i}^{P-1}\tau_{i,i+1}^{(k)}\left|\mu_{i,k} - \mu_{i+1,k}\right|. \quad (5)$$

Eq. (5) cannot be solved directly with close form, but $\widehat{\boldsymbol{\mu}}^{(t+1)}$ can be estimated using a local quadratic approximation (LQA) algorithm, which will be discussed in detail next.

A.2 Estimation of $\widehat{\boldsymbol{\mu}}^{(t+1)}$ using local quadratic approximation

According to LQA, we can approximate

$$\left|\mu_{i,k}^{(s+1)} - \mu_{i+1,k}^{(s+1)}\right| \approx \frac{\left(\mu_{i,k}^{(s+1)} - \mu_{i+1,k}^{(s+1)}\right)^2}{2\left|\hat{\mu}_{i,k}^{(s)} - \hat{\mu}_{i+1,k}^{(s)}\right|} + \frac{1}{2}\left|\hat{\mu}_{i,k}^{(s)} - \hat{\mu}_{i+1,k}^{(s)}\right|, \quad (6)$$

where s is the iteration index used to denote iterations of the LQA within the M-step (different from iteration index t in the EM algorithm), and $\widehat{\boldsymbol{\mu}}^{(s)}$ are the estimates from the

previous iteration. Thus, the minimization problem in Eq. (5) has been converted into a generalized quadratic problem which has close form solution. Notably, Eq. (5) can be decomposed into *K* separate minimization problems. For example, for each k, we can solve (iteratively over s)

$$\min_{\boldsymbol{\mu}_k^{(s+1)}} \frac{1}{2}\sum_{j=1}^{N}\left\{\hat{\Delta}_{j,\,k}^{(t+1)}\sum_{i=1}^{p}\frac{\left(x_{i,\,j}-\hat{\mu}_{i,k}^{(s+1)}\right)^2}{\left(\hat{\sigma}_i^{(t)}\right)^2}\right\} + \lambda\sum_{i=1}^{P-1}\tau_{i,\,i+1}^{(k)}\frac{\left(\mu_{i,k}^{(s+1)}-\mu_{i+1,k}^{(s+1)}\right)^2}{2\left|\hat{\mu}_{i,k}^{(s)}-\hat{\mu}_{i+1,k}^{(s)}\right|}, \qquad (7)$$

with close form. To solve Eq. (7), we need to transfer it into matrix form first. Let

- $\hat{\Delta}_k^{(t+1)}=\left(\hat{\Delta}_{1,\,k}^{(t+1)},\ldots,\hat{\Delta}_{N,\,k}^{(t+1)}\right)^T$ be the estimated latent variable for the k-cluster from E-step in the EM algorithm.
- $\boldsymbol{J}_{N\times 1}=(1,\ldots,1)^T$ is a matrix with all elements to be 1.
- $\tilde{\boldsymbol{\mu}}_k=\left(\tilde{\mu}_{1,\,k},\ldots,\tilde{\mu}_{P,\,k}\right)^T$ is the pre-defined mean vector for the k-th cluster where $\tilde{\mu}_{i,k}$ is estimated from the model without any penalization ($\lambda=0$).
- $\hat{\boldsymbol{\mu}}_k^{(s)}=\left(\hat{\mu}_{1,\,k}^{(s)},\ldots,\hat{\mu}_{P,\,k}^{(s)}\right)^T$ is the estimate of mean vector for the k-th cluster from previous iteration in the EM algorithm.
- $\hat{\boldsymbol{\mu}}_k^{(s+1)}=\left(\mu_{1,\,k}^{(s+1)},\ldots,\mu_{P,\,k}^{(s+1)}\right)^T$ is the estimate of our interest which is the mean vector for the k-th cluster.

- $\boldsymbol{D} = \begin{pmatrix} -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & \ldots & 0 & 0 \\ \vdots & & & & & \\ 0 & 0 & 0 & \ldots & -1 & 1 \end{pmatrix}_{(P-1)\times P}$ is a matrix introduced to calculate the difference of mean values for each pair of consecutive markers in a cluster.

Then Eq. (1) can also be given with

$$\boldsymbol{G}\left(\boldsymbol{\mu}_k^{(s+1)}\right) = \left(\hat{\Delta}_k^{(t+1)}\right)^T \left(\boldsymbol{X} - \boldsymbol{\mu}_k^{(s+1)}\boldsymbol{J}^T\right)^2 \boldsymbol{\Sigma}^{-1}\boldsymbol{J} + \lambda \boldsymbol{D}^{\mathrm{T}}\left(\mathrm{diag}(\boldsymbol{C})\right)^2 \boldsymbol{D}\left(\boldsymbol{\mu}_k^{(s+1)}\right)^2, \qquad (8)$$

where $\boldsymbol{C} = \left[\mathrm{abs}(\boldsymbol{D}\tilde{\boldsymbol{\mu}}_k) \odot \mathrm{abs}\left(\boldsymbol{D}\hat{\boldsymbol{\mu}}_{\boldsymbol{k}}^{(\boldsymbol{s})}\right)\right]^{-1/2}$.

Thus, we can easily find the solution for the quadratic equation of $\boldsymbol{G}\left(\boldsymbol{\mu}_k^{(s+1)}\right)$ with respect to $\boldsymbol{\mu}_k^{(s+1)}$,

$$\hat{\boldsymbol{\mu}}_k^{(s+1)} = \mathrm{argmin}(\boldsymbol{G}) = \left(\hat{\Delta}_k^{(t+1)}\right)^T \boldsymbol{X}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\left[\left(\hat{\Delta}_k^{(t+1)}\right)^T \boldsymbol{J}\boldsymbol{\Sigma}^{-1} + \lambda \boldsymbol{D}^{\mathrm{T}}(\mathrm{diag}(\boldsymbol{C}))^2\boldsymbol{D}\right]^{-1}.$$

A.3 Model selection

There are two hyperparameters to be pre-defined in the FLCNA method, including the number of clusters K and the tuning parameter λ. To find the optimal values of K and λ, we use a Bayesian information criterion (BIC), defined by

$$\mathrm{BIC}(K,\lambda) = -2\sum_{j=1}^{N} \log\left\{\sum_{k=1}^{K} \hat{\pi}_k f_k\left(\boldsymbol{x}_j; \hat{\boldsymbol{\mu}}_k, \hat{\boldsymbol{\Sigma}}\right)\right\} + d\,\log N. \qquad (9)$$

The degrees of freedom $d = K - 1 + P + e(\hat{\boldsymbol{\mu}})$, where $e(\hat{\boldsymbol{\mu}})$ is the number of distinct nonzero elements in $\hat{\boldsymbol{\mu}}$, and was used to adjust the number of breakpoints in degree of

freedom. For each pair of parameter values (K, λ), the clustering model with smallest BIC value is selected as the optimal model and the corresponding parameters are estimated.

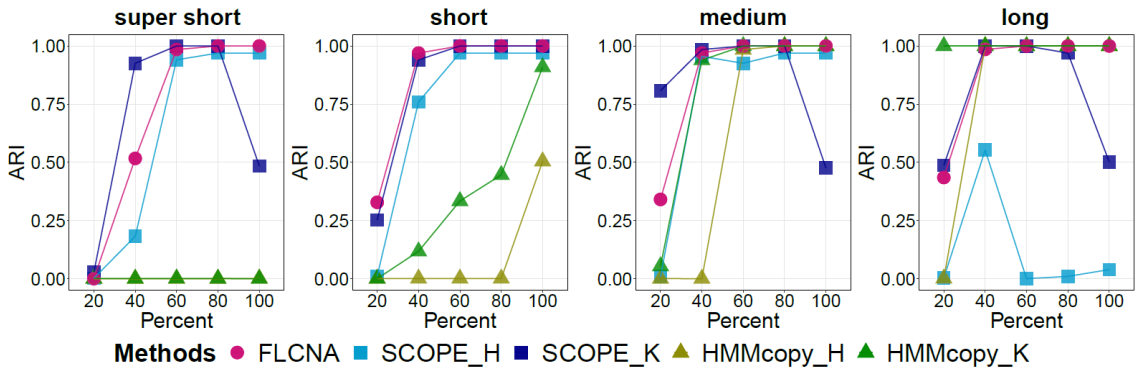

Figure A.1 Assessment of FLCNA using simulation data with three clusters and mixed CNA states. Clustering results from FLCNA were compared to existing methods (i.e., SCOPE and HMMcopy) coupled with different clustering methods. For each of three clusters, we added signals of 50 CNA segments to the background signals with varied lengths (super short: 2~5 markers, short: 5~10 markers, medium: 10~20 markers, and long: 20~35 markers) and varied CNA proportions (20%, 40%, 60%, 80%, 100%), respectively. Signals of mixed CNA states (i.e., Del.d, Del.s, Norm, Dup.s and Dup.d) were spiked in. ARI: Adjusted Rand Index; SCOPE_H: SCOPE_Hierarchical; SCOPE_K: SCOPE_K-means; HMMcopy_H: HMMcopy_Hierarchical; HMMcopy_K: HMMcopy_K-means.

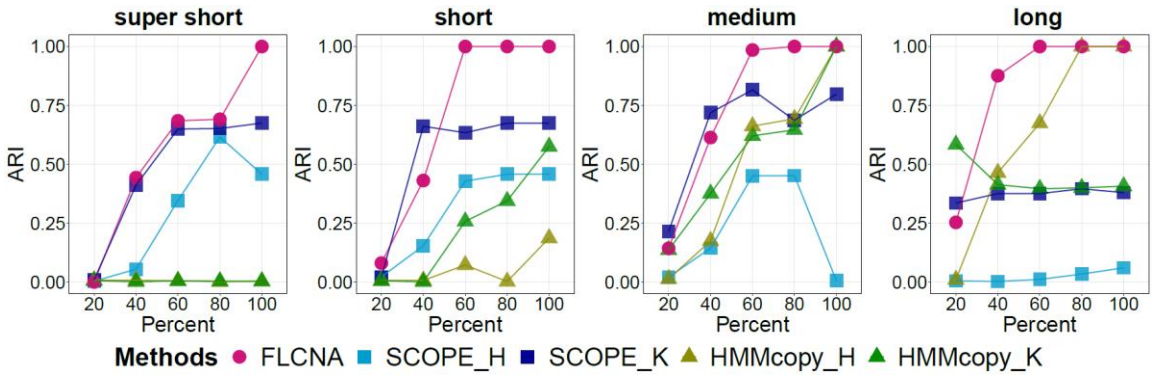

Figure A.2 Assessment of FLCNA using simulation data with five clusters, varied numbers of CNAs and mixed CNA states. Clustering results from FLCNA were compared to existing methods (i.e., SCOPE and HMMcopy) coupled with different clustering methods. For each of five clusters, we added signals of varied numbers of CNA segments (20~80) to the background signals with varied lengths (super short: 2~5 markers, short: 5~10 markers, medium: 10~20 markers, and long: 20~35 markers) and varied CNA proportions (20%, 40%, 60%, 80%, 100%), respectively. Signals of mixed CNA states (i.e., Del.d, Del.s, Norm, Dup.s and Dup.d) were spiked in. ARI: Adjusted Rand Index; SCOPE_H: SCOPE_Hierarchical; SCOPE_K: SCOPE_K-means; HMMcopy_H: HMMcopy_Hierarchical; HMMcopy_K: HMMcopy_K-means.

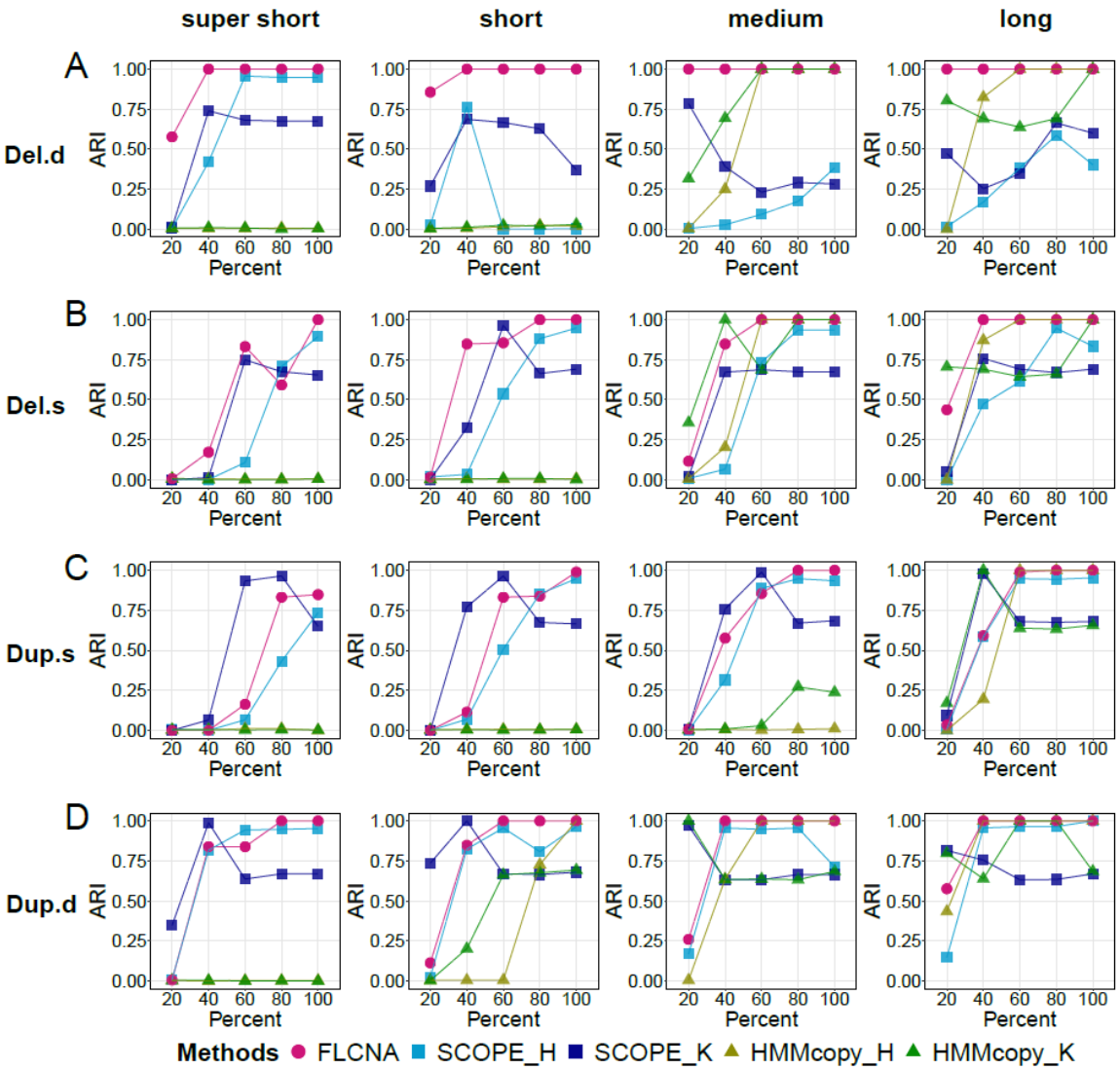

Figure A.3 Assessment of FLCNA using simulation data with five clusters and a single type of CNA state. Clustering results from FLCNA were compared to existing methods (i.e., SCOPE and HMMcopy) coupled with different clustering methods. For each of five clusters, we added signals of 50 CNA segments to the background signals with varied lengths (super short: 2~5 markers, short: 5~10 markers, medium: 10~20 markers, and long: 20~35 markers) and varied CNA proportions (20%, 40%, 60%, 80%, 100%), respectively. Signals of Del.d (A), Del.s (B), Dup.s (C) and Dup.d (D) were spiked in separately. ARI: Adjusted Rand Index; Del.d: Deletion of double copies; Del.s: Deletion of a single copy; Dup.s: Duplication of a single copy; Dup.d: Duplication of double copies; SCOPE_H: SCOPE_Hierarchical; SCOPE_K: SCOPE_K-means; HMMcopy_H: HMMcopy_Hierarchical; HMMcopy_K: HMMcopy_K-means.

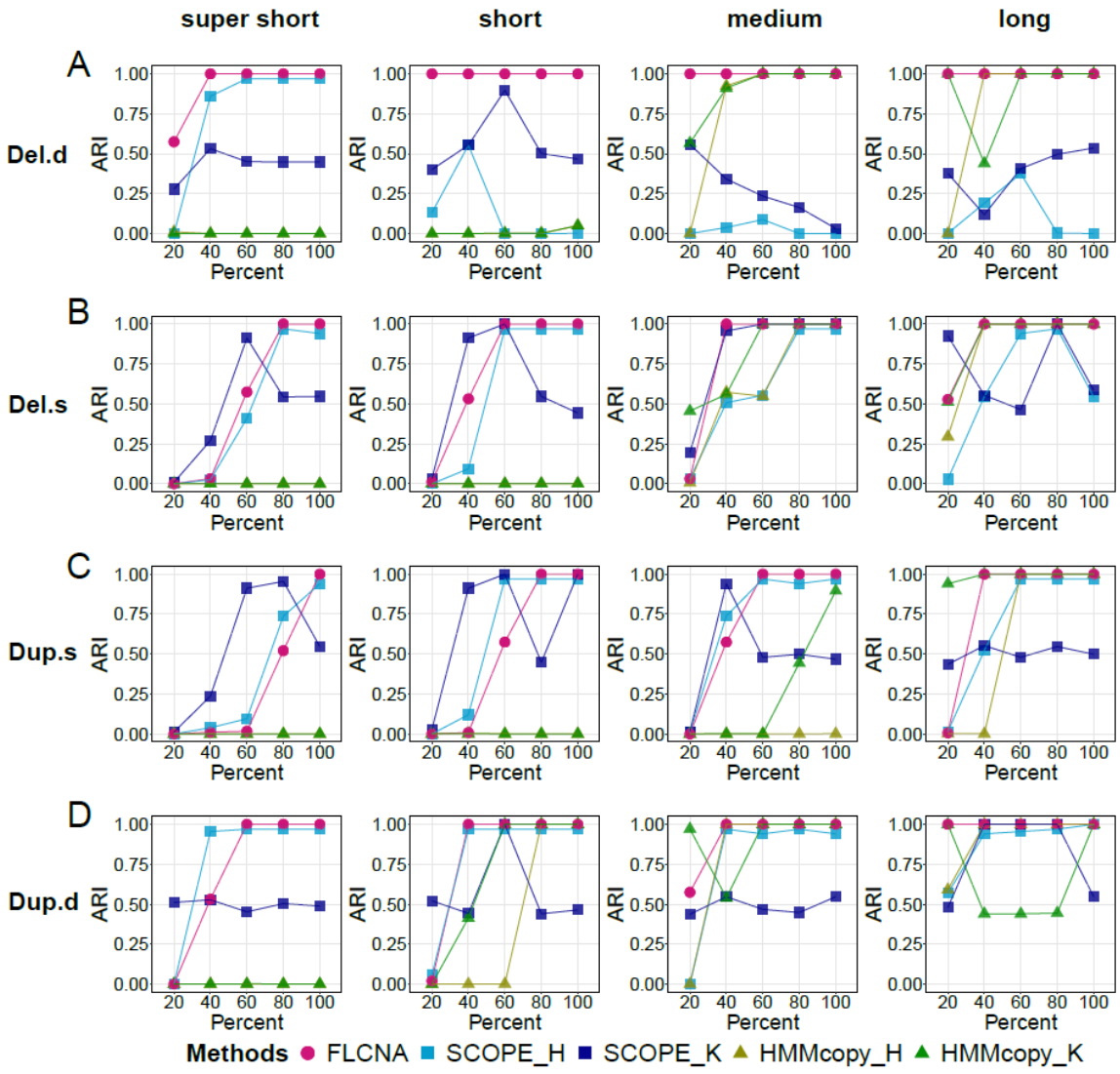

Figure A.4 Assessment of FLCNA using simulation data with three clusters and a single type of CNA state. Clustering results from FLCNA were compared to existing methods (i.e., SCOPE and HMMcopy) coupled with different clustering methods. For each of three clusters, we added signals of 50 CNA segments to the background signals with varied lengths (super short: 2~5 markers, short: 5~10 markers, medium: 10~20 markers, and long: 20~35 markers) and varied CNA proportions (20%, 40%, 60%, 80%, 100%), respectively. Signals of Del.d (A), Del.s (B), Dup.s (C) and Dup.d (D) were spiked in separately. ARI: Adjusted Rand Index; Del.d: Deletion of double copies; Del.s: Deletion of a single copy; Dup.s: Duplication of a single copy; Dup.d: Duplication of double copies; SCOPE_H: SCOPE_Hierarchical; SCOPE_K: SCOPE_K-means; HMMcopy_H: HMMcopy_Hierarchical; HMMcopy_K: HMMcopy_K-means.

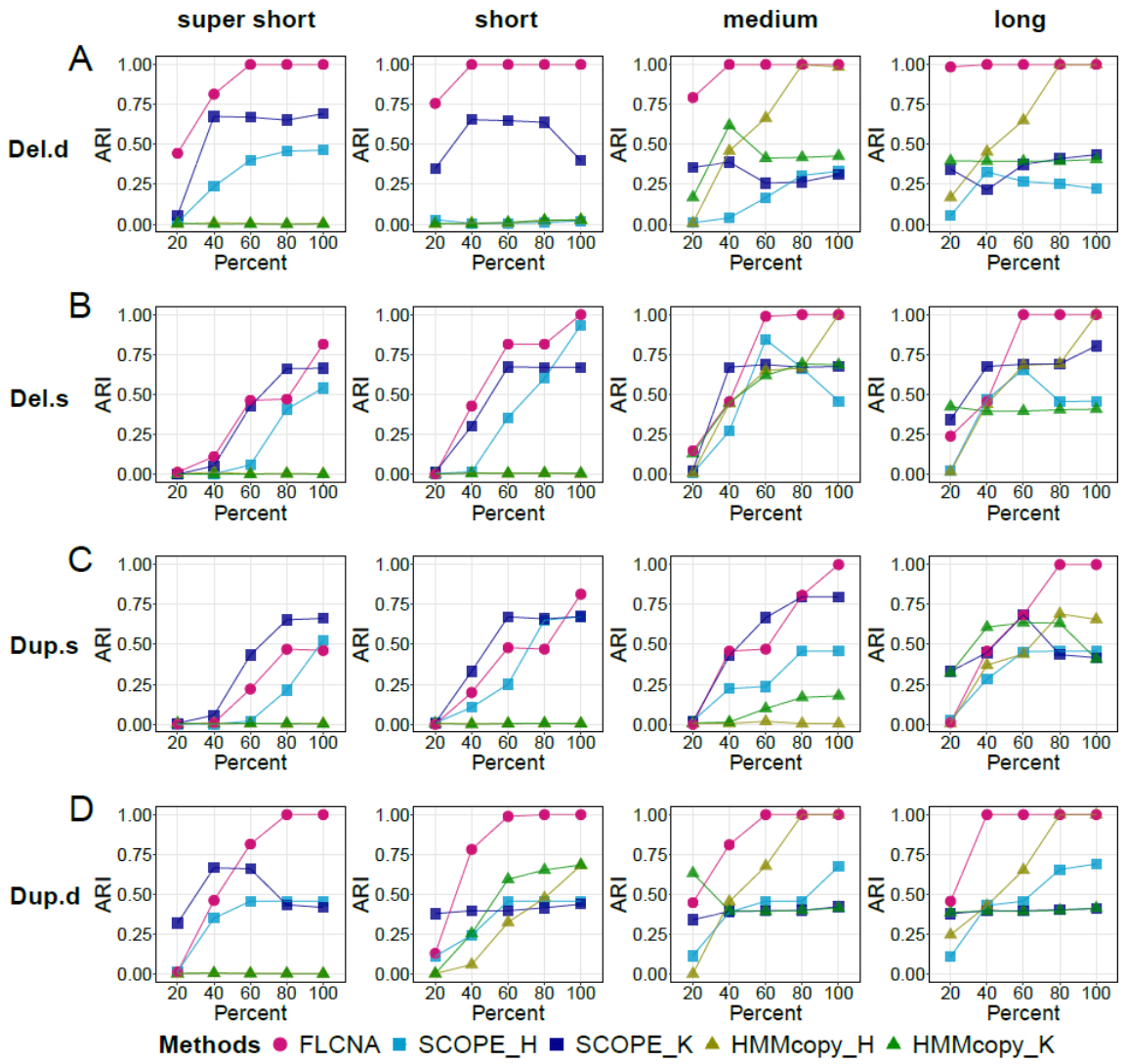

Figure A.5 Supplementary Figure A.5 Assessment of FLCNA using simulation data with five clusters, varied numbers of CNAs and a single type of CNA state. Clustering results from FLCNA were compared to existing methods (i.e., SCOPE and HMMcopy) coupled with different clustering methods. For each of five clusters, we added signals of varied numbers of CNA segments (20~80) to the background signals with varied lengths (super short: 2~5 markers, short: 5~10 markers, medium: 10~20 markers, and long: 20~35 markers) and varied CNA proportions (20%, 40%, 60%, 80%, 100%), respectively. Signals of Del.d (A), Del.s (B), Dup.s (C) and Dup.d (D) were spiked in separately. ARI: Adjusted Rand Index; Del.d: Deletion of double copies; Del.s: Deletion of a single copy; Dup.s: Duplication of a single copy; Dup.d: Duplication of double copies; SCOPE_H: SCOPE_Hierarchical; SCOPE_K: SCOPE_K-means; HMMcopy_H: HMMcopy_Hierarchical; HMMcopy_K: HMMcopy_K-means.

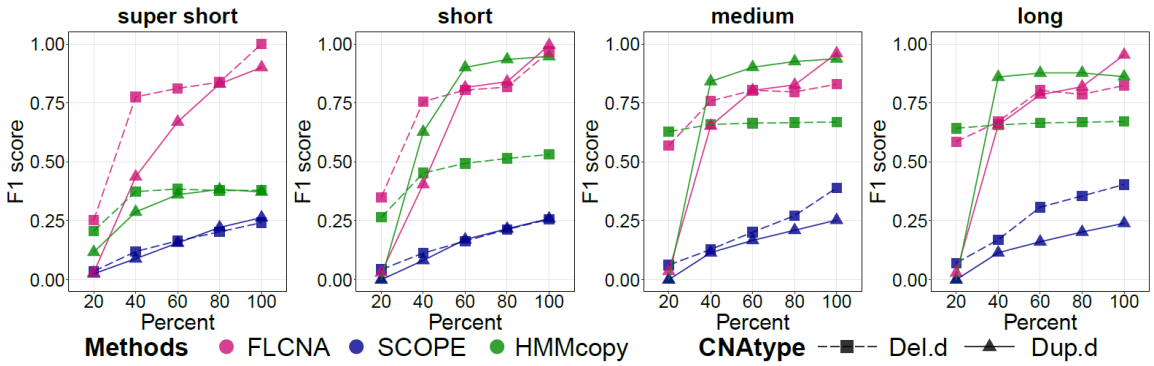

Figure A.6 Assessment of FLCNA to detect CNAs using simulation data with five clusters and aberration of double copies. CNA calls were generated by FLCNA, SCOPE and HMMcopy, respectively. For each of five clusters, we added signals of 50 CNA segments to the background signals with varied lengths (super short: 2~5 markers, short: 5~10 markers, medium: 10~20 markers, and long: 20~35 markers) and varied CNA proportions (20%, 40%, 60%, 80%, 100%), respectively. Deletion of double copies (Del.d) and duplication of double copies (Dup.d) were spiked in separately. *F*1 score was utilized to evaluate the performance of CNA detection for each method.

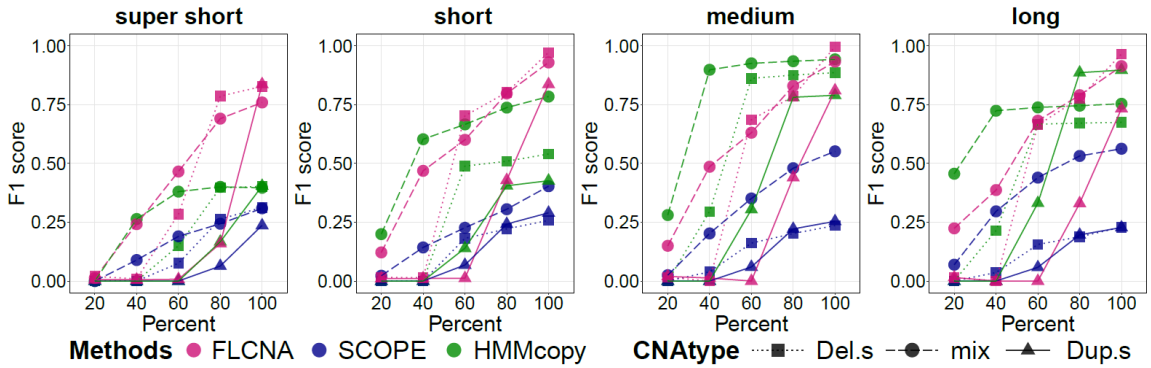

Figure A.7 Assessment of FLCNA to detect CNAs using simulation data with three clusters. CNA calls were generated by FLCNA, SCOPE and HMMcopy, respectively. For each of three clusters, we added signals of 50 CNA segments to the background signals with varied lengths (super short: 2~5 markers, short: 5~10 markers, medium: 10~20 markers, and long: 20~35 markers) and varied CNA proportions (20%, 40%, 60%, 80%, 100%), respectively. Deletion of a single copy (Del.s), mixed CNA states (mix) and duplication of a single copy (Dup.s) were spiked in separately. *F*1 score was utilized to evaluate the performance of CNA detection for each method.

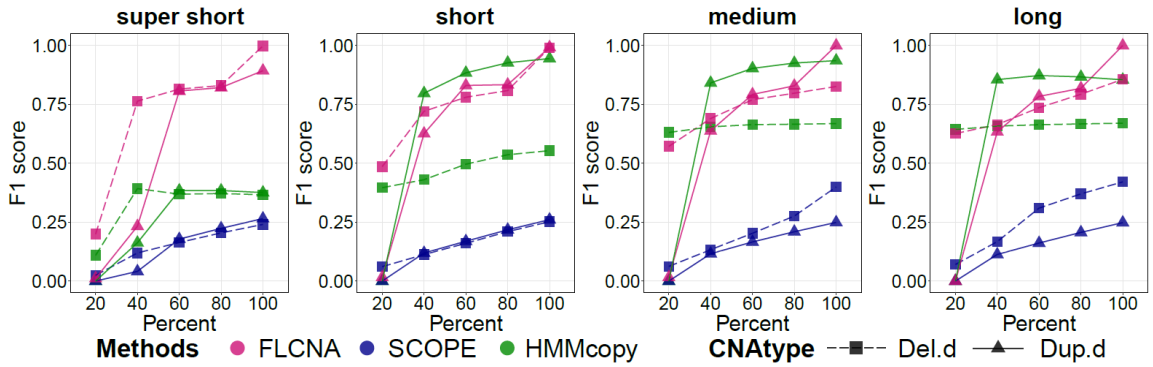

Figure A.8 Assessment of FLCNA to detect CNAs using simulation data with three clusters and aberration of double copies. CNA calls were generated by FLCNA, SCOPE and HMMcopy, respectively. For each of three clusters, we added signals of 50 CNA segments to the background signals with varied lengths (super short: 2~5 markers, short: 5~10 markers, medium: 10~20 markers, and long: 20~35 markers) and varied CNA proportions (20%, 40%, 60%, 80%, 100%), respectively. Deletion of double copies (Del.d) and duplication of double copies (Dup.d) were spiked in separately. *F*1 score was utilized to evaluate the performance of CNA detection for each method.

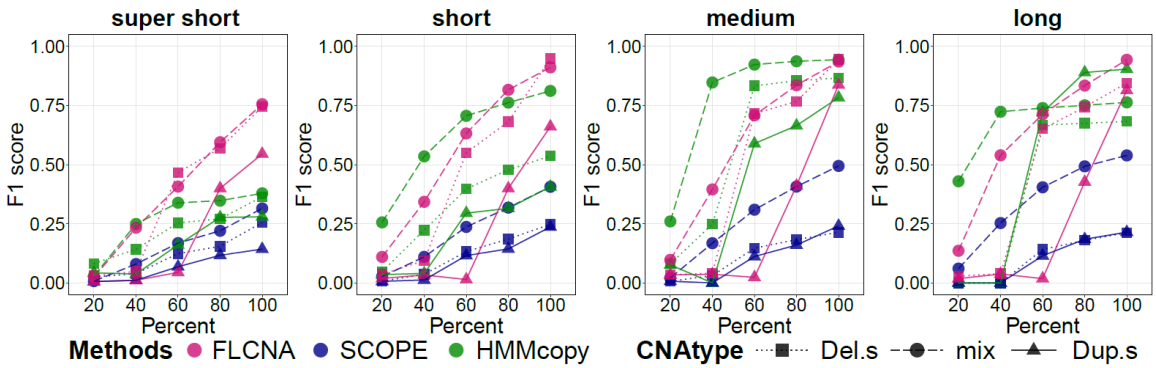

Figure A.9 Assessment of FLCNA to detect CNAs using simulation data with five clusters and varied numbers of CNAs. CNA calls were generated by FLCNA, SCOPE and HMMcopy, respectively. For each of five clusters, we added signals of varied numbers of CNA segments (20~80) to the background signals with varied lengths (super short: 2~5 markers, short: 5~10 markers, medium: 10~20 markers, and long: 20~35 markers) and varied CNA proportions (20%, 40%, 60%, 80%, 100%), respectively. Deletion of a single copy (Del.s), mixed CNA states (mix) and duplication of a single copy (Dup.s) were spiked in separately. *F*1 score was utilized to evaluate the performance of CNA detection for each method.

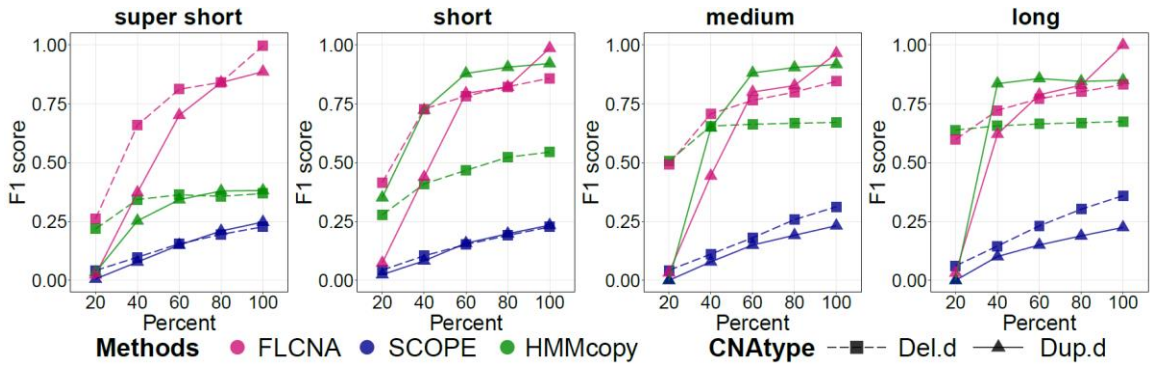

Figure A.10 Assessment of FLCNA to detect CNAs using simulation data with five clusters, varied numbers of CNAs and aberration of double copies. CNA calls were generated by FLCNA, SCOPE and HMMcopy, respectively. For each of five clusters, we added signals of varied numbers of CNA segments (20~80) to the background signals with varied lengths (super short: 2~5 markers, short: 5~10 markers, medium: 10~20 markers, and long: 20~35 markers) and varied CNA proportions (20%, 40%, 60%, 80%, 100%), respectively. Deletion of double copies (Del.d) and duplication of double copies (Dup.d) were spiked in separately. *F*1 score was utilized to evaluate the performance of CNA detection for each method.

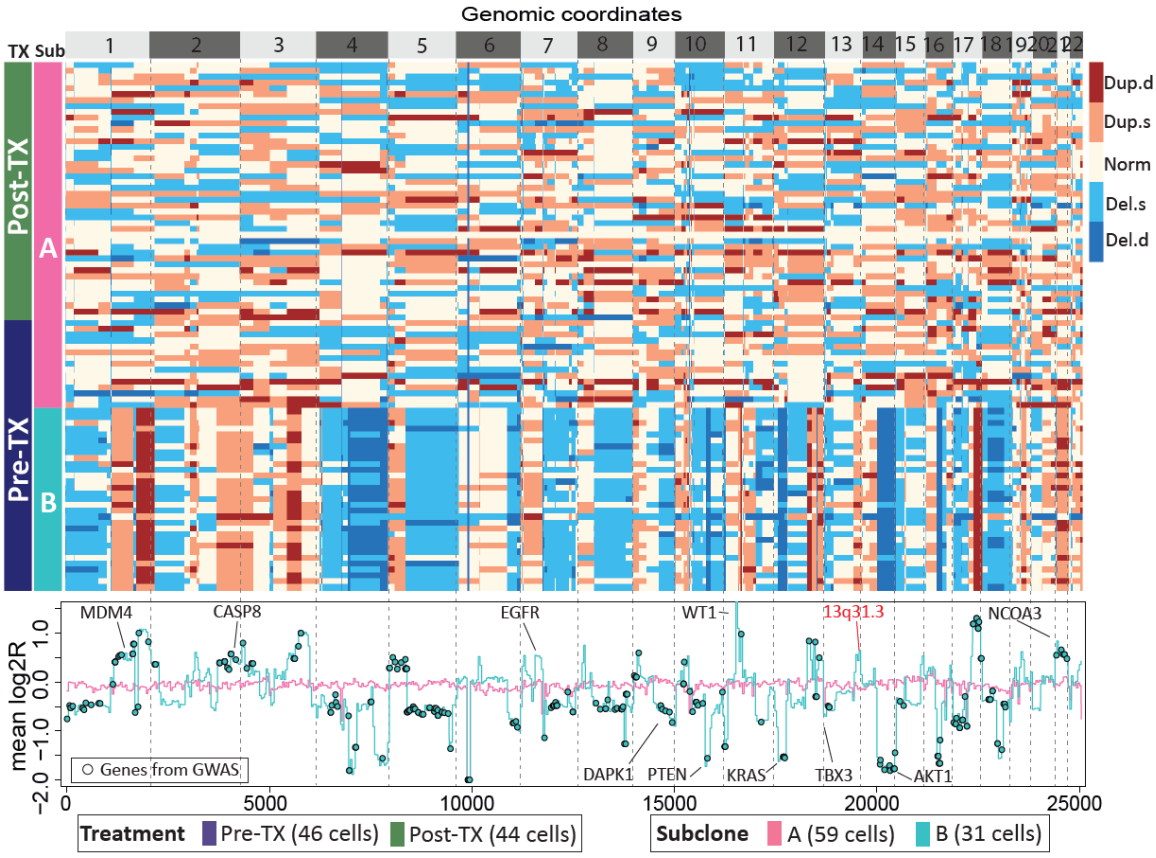

Figure A.11 Subclone clustering of KTN129 patient using FLCNA. Cell clusters and copy number profile with different CNA states (Del.d, Del.s, Norm, Dup.s and Dup.d) were generated using FLCNA. Mean log2R were provided for each cluster. Shared CNAs identified using FLCNA were matched to significant genes from genome-wide association studies (GWAS) in the NHGRI-EBI GWAS Catalog. Del.d: Deletion of double copies; Del.s: Deletion of a single copy; Norm: Normal/diploid; Dup.s: Duplication of a single copy; Dup.d: Duplication of double copies; log2R: Logarithm transformation of ratio between normalized read counts and its sample specific mean; Pre-TX: pre-treatment; Post-TX: post-treatment.

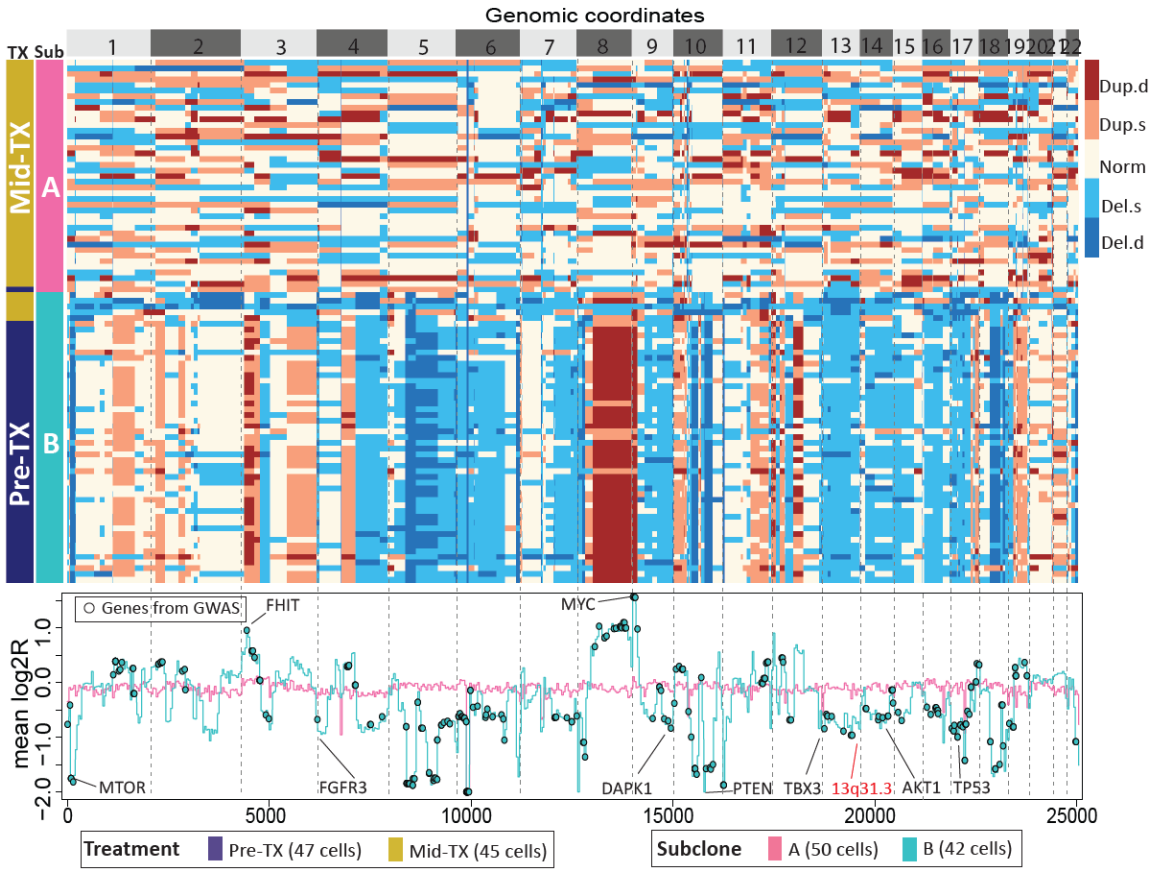

Figure A.12 Subclone clustering of KTN302 patient using FLCNA. Cell clusters and copy number profile with different CNA states (Del.d, Del.s, Norm, Dup.s and Dup.d) were generated using FLCNA. Mean log2R were provided for each cluster. Shared CNAs identified using FLCNA were matched to significant genes from genome-wide association studies (GWAS) in the NHGRI-EBI GWAS Catalog. Del.d: Deletion of double copies; Del.s: Deletion of a single copy; Norm: Normal/diploid; Dup.s: Duplication of a single copy; Dup.d: Duplication of double copies; log2R: Logarithm transformation of ratio between normalized read counts and its sample specific mean; Pre-TX: pre-treatment; Mid-TX: mid-treatment.

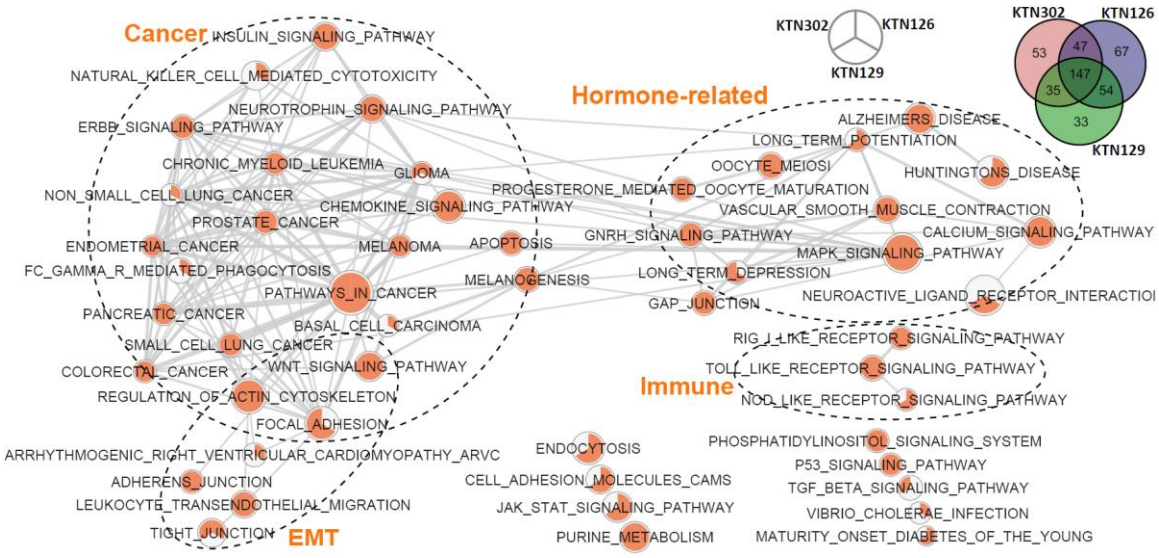

Figure A.13 Gene expression networks in the TNBC dataset. The shared CNAs identified using FLCNA were mapped into significant genes from the genome-wide association studies (GWAS) with breast cancer. These matched genes were utilized for KEGG pathway enrichment analysis for three patients (i.e., KTN126, KTN129, KTN302). Each node in network is a pie plot showing three patients. Node size corresponds to the number of genes within the pathway. Colors inner the node correspond to the index whether this pathway is identified in this patient. Edge weight corresponds to the number of genes found in both connected pathways. Venn diagrams show the distribution of genes from GWAS which were also detected from above three patients. EMT: epithelial-mesenchymal transition.

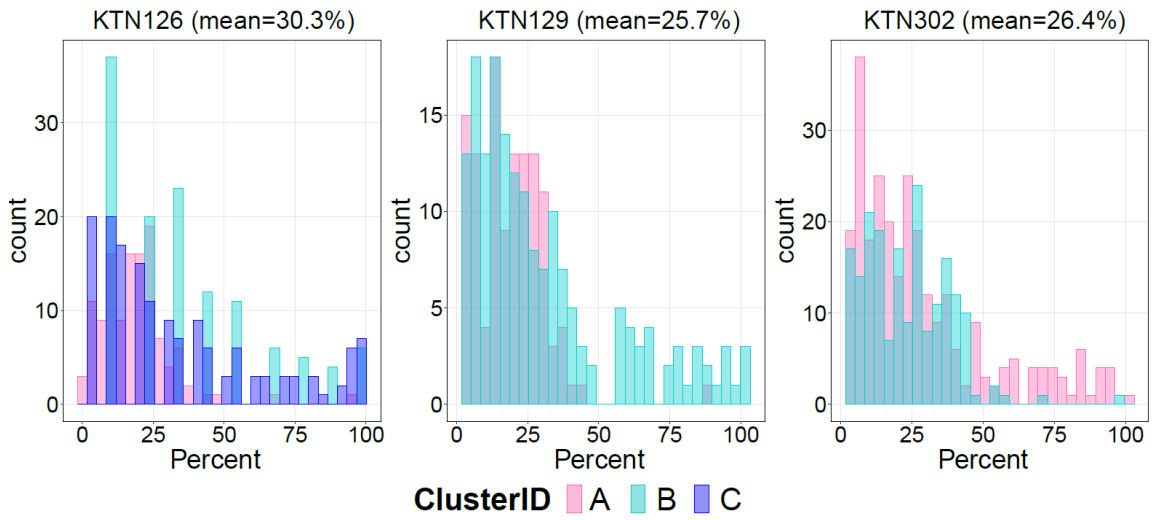

Figure A.14 Distribution of shared percentage for CNAs detected using FLCNA in the TNBC dataset. CNAs were identified from the TNBC dataset with three patients (KTN126, KTN129, KTN302) using the FLCNA method.

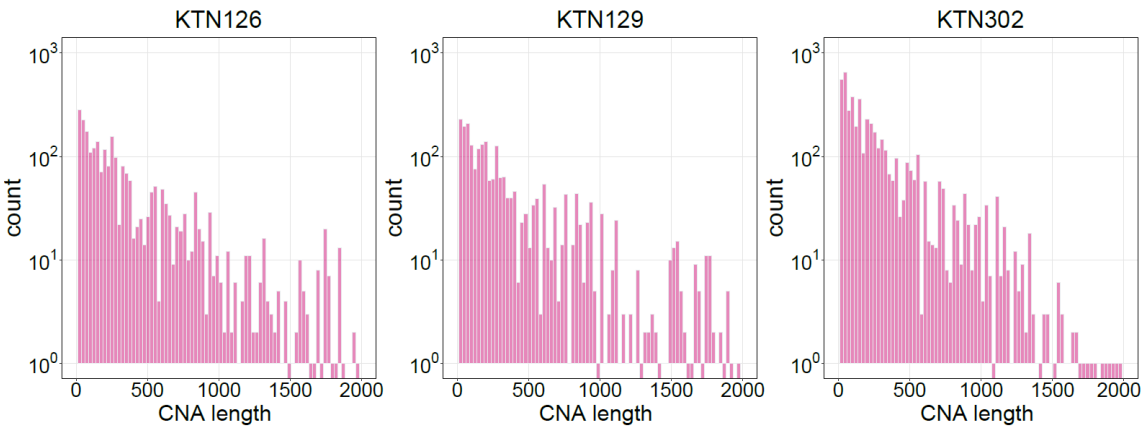

Figure A.15 Distribution of CNAs detected using FLCNA in the TNBC dataset. CNAs were identified from the TNBC dataset with three patients (KTN126, KTN129, KTN302) using the FLCNA method.

Table A.1 Computational time of different CNA detection methods with scDNA-seq data. A high-performance cluster with 8 cores and 12GB RAM was used for CNA detection with KTN126 patient in the THBC dataset.

Methods	**Time (hours)**
FLCNA	1.20
SCOPE	10.5
HMMcopy	0.15

Appendix B: Spatial pattern and differential expression analysis with spatial transcriptomic data

B.1 Estimation of the optimal hyperparameter θ_i

In our main content, to better understand the construction of variance for the expression data $\boldsymbol{y}_i$, we express the total covariance as $\boldsymbol{\Sigma}_i = \tau_{i1}\boldsymbol{K}_i + \tau_{i2}\mathbf{I}$, which includes τ_{i1} and τ_{i2} to measure variance explained by spatial pattern and random noise, respectively. However, to simplify the derivation process for parameter estimation, we factor out a scaling variance factor τ_i from $\boldsymbol{\Sigma}_i$, and redefine it as $\boldsymbol{\Sigma}_i = \tau_i(\boldsymbol{K}_i + \varphi_i\mathbf{I})$, where φ_i is a variance ratio factor with $\varphi_i = {}^{\tau_{i2}}/_{\tau_i}$. For simplicity of the presentation, we ignore the gene notation for all variables in the model in the following text (e.g., using $\boldsymbol{y}$ instead of $\boldsymbol{y}_i$).

$$LL(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\Theta}) = -\frac{N}{2}\log(2\cdot\pi) - \frac{1}{2}\log(|\boldsymbol{\Sigma}|) - \frac{1}{2}(\boldsymbol{y} - \boldsymbol{\mu}\cdot 1)^T(\boldsymbol{\Sigma})^{-1}(\boldsymbol{y} - \boldsymbol{\mu}\cdot 1),$$

where $\boldsymbol{\mu}$ is the mean gene expression level with $\boldsymbol{\mu} = \boldsymbol{X}^T\boldsymbol{\beta}$. Then we factor the kernel covariance matrix through spectral decomposition with $\boldsymbol{K} = \mathbf{V}\mathbf{S}\mathbf{V}^T$ to speed up the calculation of likelihood. Thus the total variance can also be expressed as $\boldsymbol{\Sigma} = \tau\cdot\mathbf{V}(\mathbf{S} + \varphi\cdot\mathbf{I})\mathbf{V}^T$, given that eigenvectors are orthogonal, $\mathbf{V}\mathbf{V}^T = \mathbf{I}$. To estimate the mean expression level $\boldsymbol{\mu}$ and scaling variance τ, we take the first derivative of $LL(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\Theta})$ w.r.t. $\boldsymbol{\mu}$ and τ, respectively.

$$\frac{\partial LL(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\Theta})}{\partial \boldsymbol{\mu}} = 0 \rightarrow \widehat{\boldsymbol{\mu}} = \frac{(\mathbf{V}^T\mathbf{1})^T(\mathbf{S}+\varphi\cdot\mathbf{I})^{-\mathbf{1}}(\mathbf{V}^T\boldsymbol{y})}{(\mathbf{V}^T\mathbf{1})^T(\mathbf{S}+\varphi\cdot\mathbf{I})^{-\mathbf{1}}(\mathbf{V}^T\mathbf{1})}$$

$$\frac{\partial LL(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\Theta})}{\partial \tau} = 0 \rightarrow \hat{\tau} = \frac{1}{N}\cdot\frac{(\mathbf{V}^T\boldsymbol{y}-\mathbf{V}^T\mathbf{1}\cdot\widehat{\boldsymbol{\mu}})^T(\mathbf{V}^T\boldsymbol{y}-\mathbf{V}^T\mathbf{1}\cdot\widehat{\boldsymbol{\mu}})}{(\mathbf{S}+\varphi\cdot\mathbf{I})}$$

We found that the estimate of mean expression level $\widehat{\boldsymbol{\mu}}$ and scaling variance $\hat{\tau}$ can both be expressed as a function depending only on φ, separately. Taking $\widehat{\boldsymbol{\mu}}$ and $\hat{\tau}$ back to the log likelihood function $LL(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\Theta})$, we can get an updated log likelihood function depending only on kernel hyperparameter θ and variance ratio factor φ. Thus, the optimal length-scale hyperparameter $\hat{\theta}$ and optimal $\hat{\varphi}$ can be simultaneously estimated by maximizing the log likelihood function $LL(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\Theta})$. $\hat{\theta}$ will be used in the hypothesis test for identifying spatially variable genes.

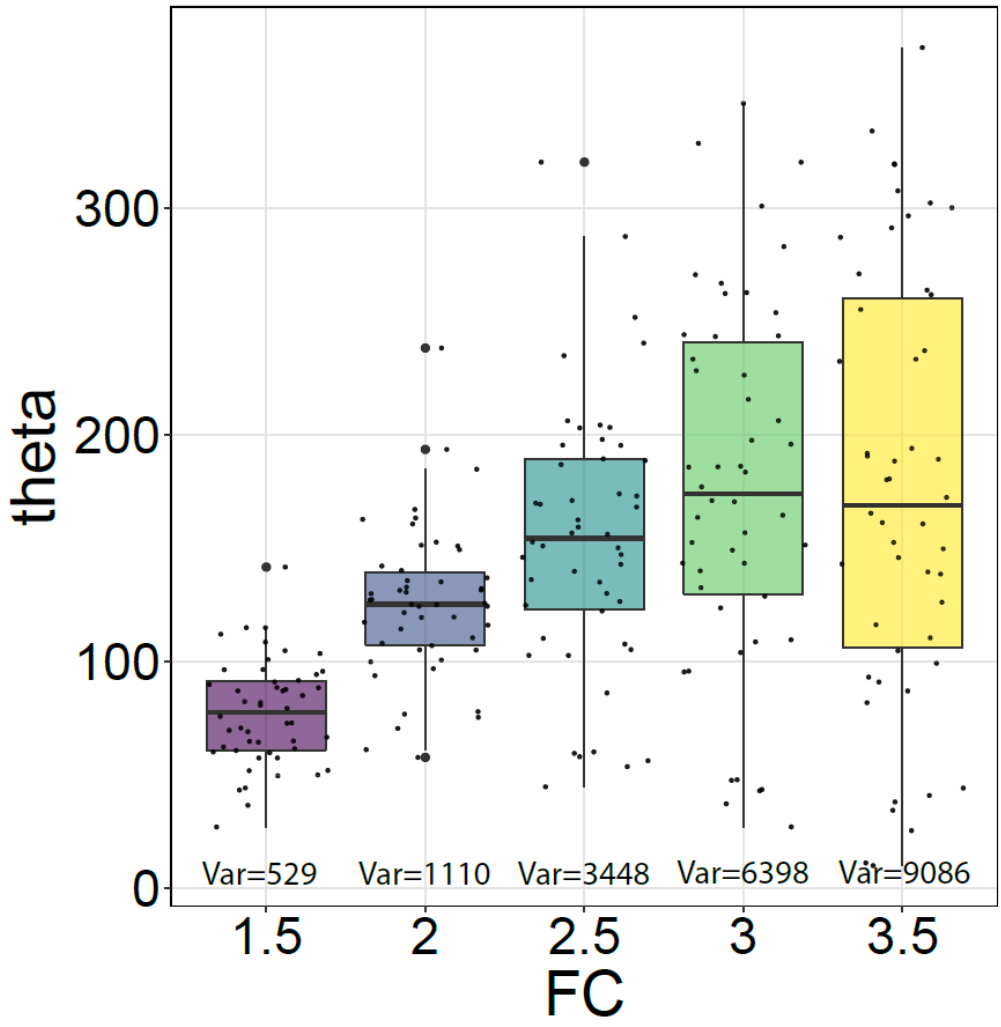

Figure B.1 Distribution of hyperparameter values estimated using SPADE in the simulation data with the MERFISH dataset. Length-scale hyperparameter values for the pre-defined 50 marker genes were estimated using the SPADE method in the simulation data with the MERFISH data. Var: Variance.

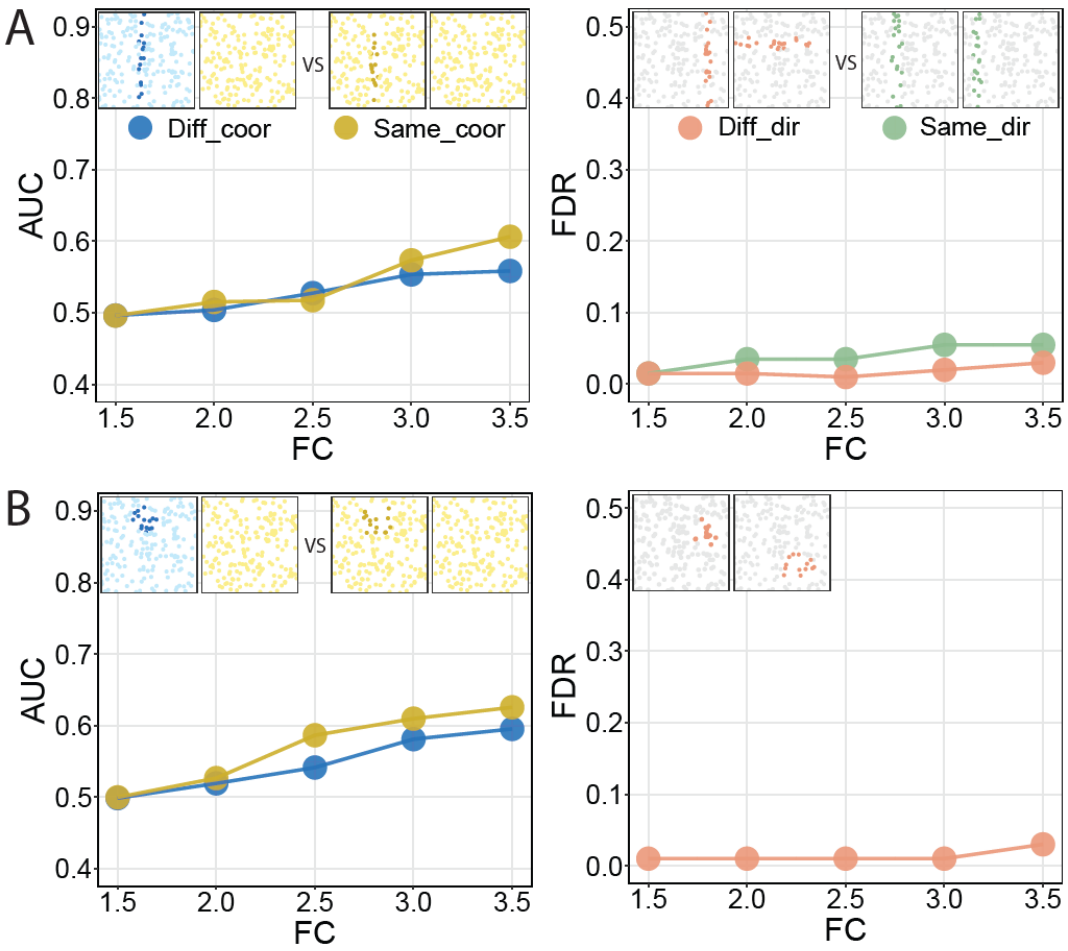

Figure B.2 Assessment of the robustness for SPADE to identify spatially variable genes between groups with marked spots proportion of 10%. We also evaluated the effects of spot coordinates and pattern directions to the performance of SPADE, separately. Specifically, to evaluate how coordinates of spots affect SPADE to identify SV genes between groups, we compared the performance of SPADE in groups with same coordinates to groups with different coordinates. AUC was utilized to evaluate the effect of coordinates to SPADE. We also compared SPADE in groups with the same spatial pattern directions to those with different directions. The false discovery rate (FDR) was used to assess the type I error of SPADE. Both hotspots (A)and streak (B) patterns were considered in simulation studies. Diff_coor: different coordinates; Same_coor: same coordinates; Diff_dir: different pattern directions; Same_dir: Same pattern directions.

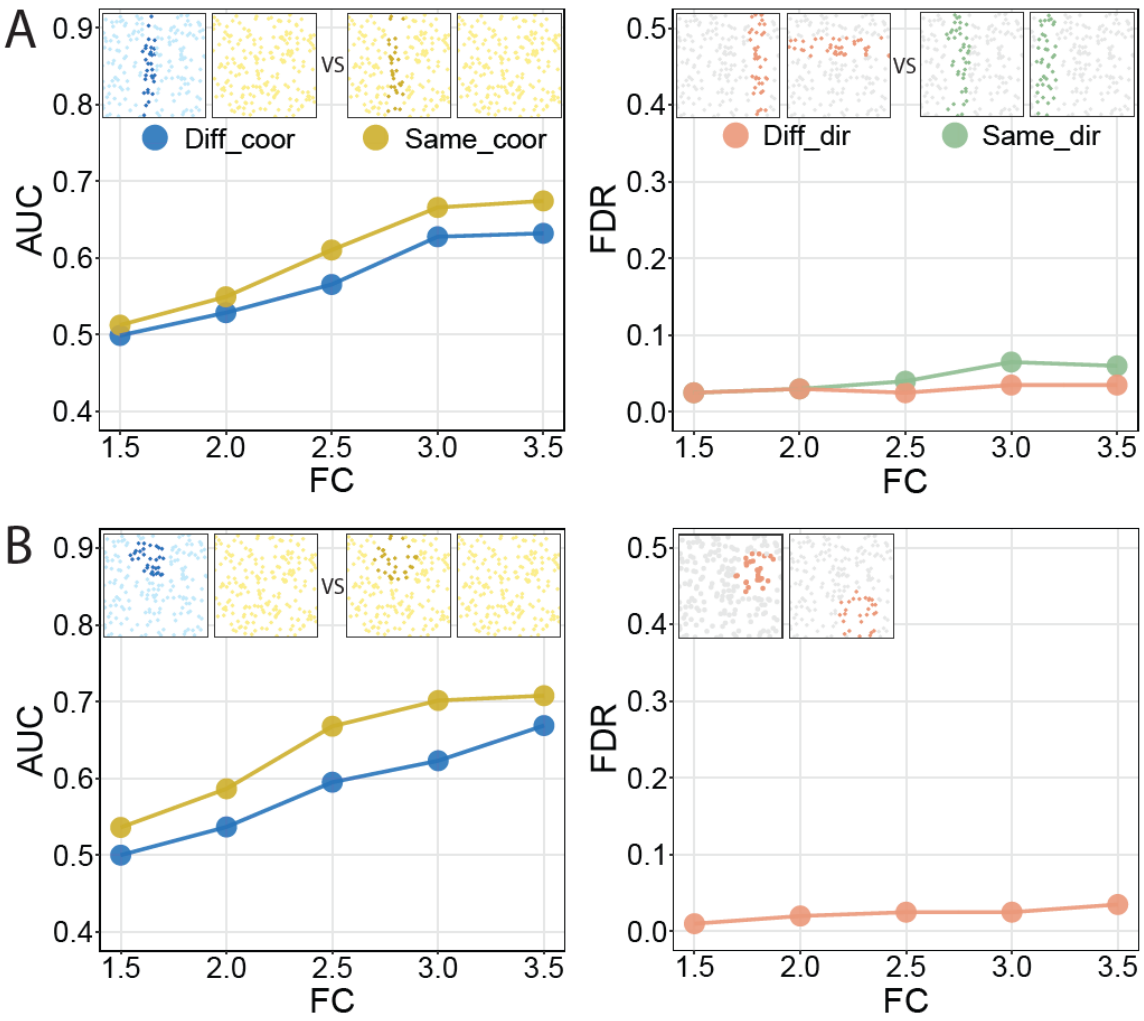

Figure B.3 Assessment of the robustness for SPADE to identify spatially variable genes between groups with marked spots proportion of 20%. We also evaluated the effects of spot coordinates and pattern directions to the performance of SPADE, separately. Specifically, to evaluate how coordinates of spots affect SPADE to identify SV genes between groups, we compared the performance of SPADE in groups with same coordinates to groups with different coordinates. AUC was utilized to evaluate the effect of coordinates to SPADE. We also compared SPADE in groups with the same spatial pattern directions to those with different directions. The false discovery rate (FDR) was used to assess the type I error of SPADE. Both hotspots (A)and streak (B) patterns were considered in simulation studies. Diff_coor: different coordinates; Same_coor: same coordinates; Diff_dir: different pattern directions; Same_dir: Same pattern directions.

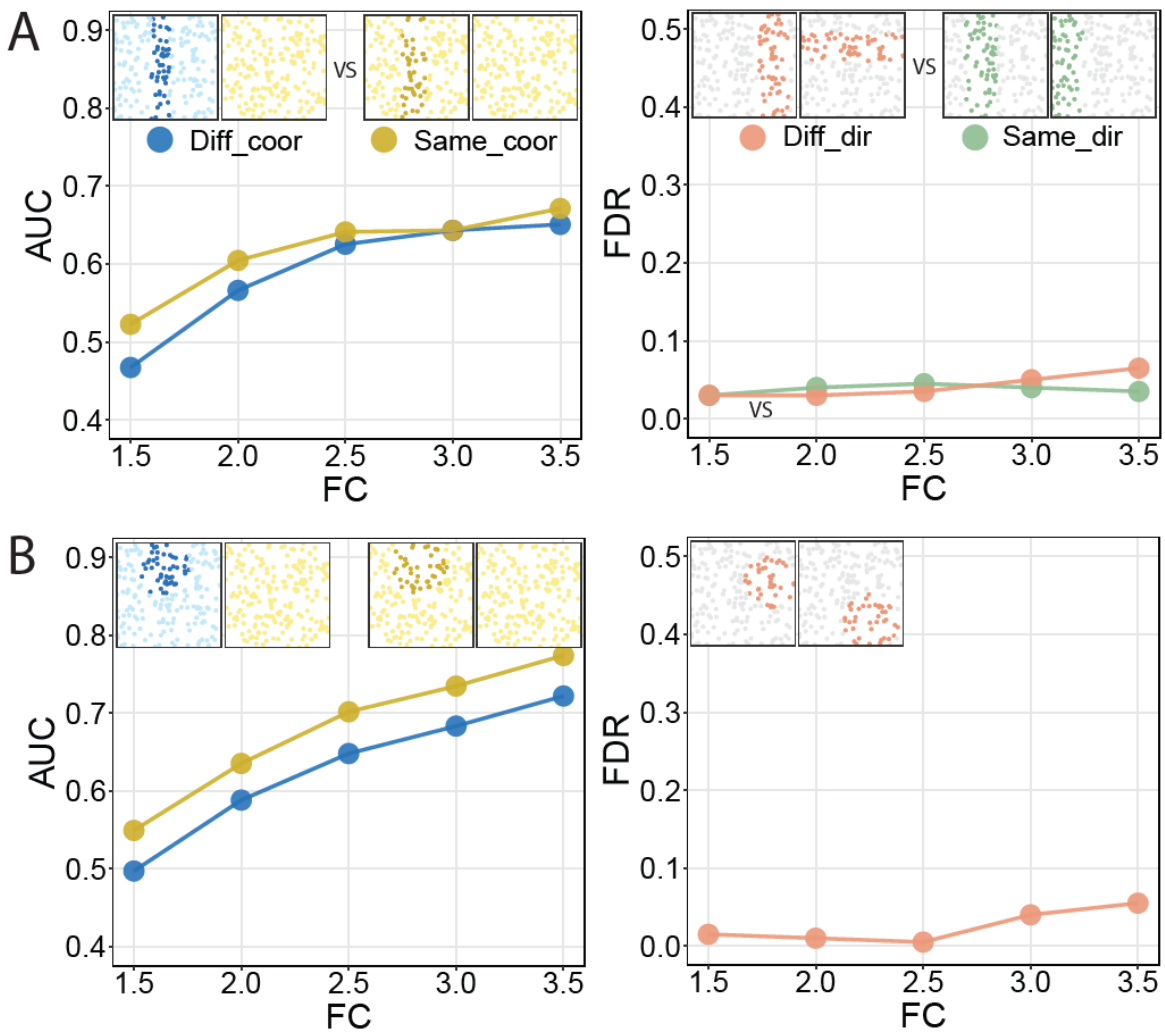

Figure B.4 Assessment of the robustness for SPADE to identify spatially variable genes between groups with marked spots proportion of 30%. We also evaluated the effects of spot coordinates and pattern directions to the performance of SPADE, separately. Specifically, to evaluate how coordinates of spots affect SPADE to identify SV genes between groups, we compared the performance of SPADE in groups with same coordinates to groups with different coordinates. AUC was utilized to evaluate the effect of coordinates to SPADE. We also compared SPADE in groups with the same spatial pattern directions to those with different directions. The false discovery rate (FDR) was used to assess the type I error of SPADE. Both hotspots (A)and streak (B) patterns were considered in simulation studies. Diff_coor: different coordinates; Same_coor: same coordinates; Diff_dir: different pattern directions; Same_dir: Same pattern directions.

Table B.1 Computational time of different methods with SeqFISH dataset. A high-performance cluster with 12GB RAM was used to identify spatially varible genes using SeqFISH dataset.

Methods	**Time (mins)**
SPADE	1.16
SPARK	1.08
SpatialDE	1.68
MERINGUE	0.01