University of Warsaw
Faculty of Mathematics, Informatics and Mechanics

Senbai Kang
Student no. Student No. 415790

Probabilistic graphical models for inferring tumor phylogeny and genomic variants from single cell DNA sequencing data

PhD dissertation
in COMPUTER SCIENCE

Supervisor:
Dr hab. Ewa Szczurek
Institute of Informatics

Warsaw, August 2023

Abstract

Recently, the swift advancements in single-cell DNA sequencing have enabled quantitative assessment of genetic content in individual cells, allowing downstream analyses at the single-cell level, such as variant calling and phylogenetic tree reconstruction. These tasks are particularly valuable in the study of tumor progression, where mutations accumulate in the course of evolution. Within this context, a plethora of models have been formulated. However, they often exhibit limitations, either by concentrating solely on a particular task, yielding a cascade of uncertainties through subsequent analyses, or only by partially leveraging the essential information intrinsic to the data. Hence, an imperative arises for single-cell DNA sequencing analysis to alleviate the propagation of errors throughout the analytical pipeline and to holistically harness the complete wealth of information embedded within the data. The overall aim of my thesis is to develop novel statistical methods to overcome these challenges by direct modeling of the noise and the signal in the data. Specifically, the first approach, called SIEVE, jointly performs variant calling and phylogenetic reconstruction from raw read counts of single-cell DNA sequencing by allowing for nucleotide substitutions. Subsequently, the second approach, called DelSIEVE, extends the capabilities beyond SIEVE by encompassing somatic deletions. In this way, DelSIEVE enhances the performance of variant identification. In summary, both SIEVE and DelSIEVE offer comprehensive models of single cell evolution and occurrence of genomic variants in single cells.

Keywords

intra-tumor heterogeneity, single-cell DNA sequencing, cell phylogeny reconstruction, somatic variant calling, single nucleotide variants, somatic deletions, finite-sites assumption, acquisition bias correction, statistical phylogenetic models, probabilistic graphical models, Markov chain Monte Carlo

Thesis domain (Socrates-Erasmus subject area codes)

11.3 Informatics, Computer Science

Subject classification

Applied computing → Life and medical sciences → Computational biology
Computing methodologies → Machine learning

Tytuł pracy w języku polskim

Probabilistyczne modele grafowe do wnioskowania o filogenezie nowotworów i wariantach genomowych z danych sekwencjonowania DNA pojedynczych komórek

Acknowledgments

Approaching the end of my PhD journey, it is imperative to acknowledge my utmost gratitude towards my supervisor, dr hab. Ewa Szczurek. Not only did Ewa bring me to the field of Computational Biology, but she also supported me continuously in every possible way. Her wise advice and incredible patience were indispensable to the completion of my work.

I am particularly grateful for the scientific collaboration with Dr. Jack Kuipers and Prof. Dr. Niko Beerenwinkel at ETH Zurich as well as Prof. Dr. David Posada at the University of Vigo. Their insightful contribution in the context of statistics and cancer biology greatly enriched and improved my work.

I would like to thank my collaborators and friends, Nico Borgsmüller from ETH Zurich and Monica Valecha from the University of Vigo for joining the ride. My special thanks also go to members in Ewa Szczurek's lab and ESRs in ITN-CONTRA for all the wonderful and memorable moments we shared together in this journey.

I sincerely acknowledge Prof. Dr. Oliver Y. Chén at the University of Lausanne as well as Dr. Bernat Bramon Mora at the Lausanne University Hospital for proof-reading and comments on the introduction to this thesis. The Polish version of the abstract of this thesis was prepared with the kind help of Agnieszka Geras.

Last but not least, I specially thank Zhixuan Li for her constant support and encouragement throughout the entire course of my PhD work, for sharing every piece of bitterness, as well as for celebrating together each modest stride and significant milestone.

The research in this thesis has received funding from
the European Union's Horizon 2020 research and innovation programme
under the Marie Skłodowska-Curie grant agreement No. 766030.

The work presented in this dissertation has been published in the following research paper and preprint:

Senbai Kang, Nico Borgsmüller, Monica Valecha, Jack Kuipers, Joao M. Alves, Sonia Prado-López, Débora Chantada, Niko Beerenwinkel, David Posada, and Ewa Szczurek. SIEVE: joint inference of single-nucleotide variants and cell phylogeny from single-cell DNA sequencing data. *Genome Biol* 23, 248 (2022). https://doi.org/10.1186/s13059-022-02813-9

Senbai Kang, Nico Borgsmüller, Monica Valecha, Magda Markowska, Jack Kuipers, Niko Beerenwinkel, David Posada, and Ewa Szczurek. Delsieve: joint inference of single-nucleotide variants, somatic deletions, and cell phylogeny from single-cell dna sequencing data. *bioRxiv*, (2023). https://doi.org/10.1101/2023.09.09.556903

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1. Basic cancer biology

Cancer, also known as cancerous or malignant tumor, is a complex and devastating disease that arises from the uncontrollable growth and proliferation of abnormal cells within the body. These abnormal cells are capable of infiltrating nearby tissues and spreading to other parts of the body through the bloodstream or lymphatic system, a phenomenon known as *metastasis* [4, 5]. Cancer can affect virtually any tissue or organ in the body and is a leading cause of morbidity and mortality worldwide, where metastasis is responsible for approximately 90% of cancer deaths [6]. In 2020, almost ten million cancer deaths occurred, with the top five cancer types leading to death being lung, colorectal, liver, stomach, and female breast cancers [7].

The development of cancer involves various biological mechanisms that facilitate the conversion of healthy cells to malignant ones. This transformation is often initiated by somatic mutations that disrupt the normal regulation of cell growth, division, and death [8, 9, 10, 11, 12, 13, 14, 15]. Somatic mutations can be divided into different types of DNA sequence changes, including point mutations (or single nucleotide variants, SNVs), where nucleotide substitutions occur at individual loci, small insertions and deletions with length < 50 basepairs, copy number aberrations (CNAs), where insertions and deletions occur at regions ranging in length between one kilobase and five megabases, and interchromosomal rearrangements, where a DNA segment is pruned and reattached at random to another segment [16, 17]. Somatic mutations can be acquired through exposure to carcinogens (e.g., tobacco smoke, certain chemicals, radiation) or can occur spontaneously during DNA replication [18, 16, 19].

Some somatic mutations, such as point mutations and insertions, can lead to the activation of oncogenes [20], which promote tumor proliferation. One of the most frequently mutated oncogenes among all cancer types is *KRAS*. For colorectal cancer (CRC), around 40% of patients harbor activated *KRAS* via point mutations [21]. The activation of *KRAS* leads to the disruption of the hydrolysis of guanosine triphosphate and/or the enhancement of nucleotide exchange. Consequently, the downstream signaling pathways are continuously activated, resulting in boosted tumor cell proliferation [22, 23].

Other somatic mutations can lead to cancer via different mechanisms. For instance, deletions can inactivate tumor suppressor genes [24], which normally restrict tumor growth [12, 25, 16, 26, 13, 15]. *TP53* is perhaps the most famous example of tumor suppressor genes, found in more than half of all sporadic cancers [27]. *TP53* is involved in the synthesis of the p53 protein, which plays a crucial role in preventing the development and progression of cancer. p53 monitors the integrity of the DNA and acts as a checkpoint during cell division.

When DNA damage or abnormalities are detected, p53 can initiate a series of responses, including cell cycle arrest, DNA repair, or triggering apoptosis (programmed cell death) to prevent the propagation of damaged cells [28, 27]. Thus, *TP53* mutations lead to abnormal activities of p53 proteins, impairing their ability to respond to DNA damage and to regulate essential cellular processes such as cell growth and differentiation.

Additionally, defects in DNA repair mechanisms can lead to the accumulation of somatic mutations, further fueling cancer development [29, 30, 31]. The sequential accumulation of somatic mutations indicates that cancer is an evolutionary process [10, 32] that leads to cell populations possessing highly heterogeneous genomic profiles, a complexity known as intra-tumor heterogeneity (ITH) [33, 34, 32, 35].

In the face of the diverse mechanisms of cancer, several therapy options for cancer have been developed, including surgery, radiation therapy, chemotherapy, immunotherapy, targeted therapy, and hormonal therapy [36, 37]. The choice of therapy depends on the type and stage of cancer, as well as the patient's overall health. Surgical removal of tumors aims to physically eliminate cancerous tissues, while radiation therapy uses high-energy rays to kill cancer cells or prevent their growth. Chemotherapy involves the use of drugs to destroy cancer cells, but it can also harm healthy cells at the same time. Immunotherapy has emerged as a promising approach that stimulates the immune system of the body to recognize and attack cancer cells specifically. Targeted therapies focus on specific molecules or signaling pathways that are essential for tumor growth and survival, while hormonal therapy is effective for hormone-sensitive cancers by blocking or reducing the influence of hormones that promote cancer growth. However, patients frequently suffer from drug resistance, treatment failure, cancer recurrence and poor prognosis, primarily due to the existence of ITH [38, 32, 39, 35]. Therefore, understanding and measuring ITH are of substantial importance to the diagnosis and clinical therapy of cancer.

1.2. Single-cell DNA sequencing

A straightforward approach to studying ITH is to directly examine the genome of cancer cells, thereby quantifying the inherent heterogeneity. The revolutionary development of first generation sequencing, also known as *Sanger sequencing* [40, 41], made it possible to accurately analyze the genetic composition of cells. Its application, however, is limited due to cumbersomeness, high cost ($1 per read), and low throughput (96 $\sim$ 384 DNA fragments per run) [42, 43, 44, 45]. To overcome these issues, *next generation sequencing* (NGS; or *massive parallel sequencing*) [46, 47] was developed, featuring high accuracy ($> 99.9\%$), high throughput (millions to billions of DNA fragments per run), and low cost ($0.02 per megabase) [42, 43, 45]. Both the first and the next generation sequencing require template amplification, potentially introducing errors, sequence-dependent biases and information loss [42, 43]. The *third generation sequencing*, or *long-read sequencing* [48, 49], aims to mitigate this problem by sequencing single molecules without amplification. The read length of the third generation sequencing is 10 $\sim$ 900 kilobases, much longer than that of the previous generations (up to 0.8 and 0.6 kilobases for the first and the next generation sequencing, respectively) [42, 43, 50], and thus facilitating *de novo* assembly, the identification of transcript isoforms, and the detection of structural variants [50, 51]. However, the third generation sequencing is currently less mature than the next generation sequencing, with higher cost (around $0.1 per megabase) and higher sequencing error rate [42, 43, 50].

Among the three sequencing technologies, NGS currently stands as the most cost-effective and extensively employed method. With NGS, the bulk DNA sequencing of cancer is typically conducted on cancer tissues, comprising millions of cells, where the genomes are fragmented into short pieces and subsequently sequenced together in mixtures, measuring genomic profiles in clones. Profiling the genome of single cells on the platform of NGS, the recently emerged *single-cell DNA sequencing* (scDNA-seq) improves the resolution of measuring ITH down to the single-cell level [52, 53, 54, 55]. Since the genetic material contained in a single cell is only 6pg, whole genome amplification (WGA) methods are developed in order to generate enough genetic material for later sequencing [56, 57, 58, 55, 59]. Some of these methods, such as degenerate oligonucleotide primed PCR (DOP-PCR) [60, 52, 61, 62], are PCR-based and offer high uniformity of sequencing coverage across the genome. This characteristic of the obtained sequencing data makes it suitable for CNA calling when coupled with single-cell whole genome sequencing (scWGS) [56, 57, 58, 55, 59]. Additionally, PCR-based WGA methods have the capability to amplify hundreds of cells in a single run, enabling high-throughput sequencing [59]. However, they often face a trade-off between covering a wide range of the genome and maintaining sufficient sequencing depth, which makes them a suboptimal choice for calling SNVs [56, 57, 55, 59]. Furthermore, it is worth mentioning that the PCR-based WGA methods employ thermostable polymerases, which have elevated error rates compared to thermolabile polymerases, introducing extra errors during the amplification [63].

The second type of WGA methods are isothermal-based, such as multiple displacement amplification (MDA), which uses isothermal random priming and extension with Φ29 polymerase [64, 65, 66, 67, 68]. Φ29 DNA polymerase exhibits two beneficial characteristics: strand displacement and high processivity. When the polymerase encounters newly synthesized DNA, it engages in strand displacement, leading to the creation of single-stranded DNA templates. These templates are then reprimed and extended, resulting in DNA amplification through an isothermal reaction. The high processivity of the Φ29 DNA polymerase enables the generation of DNA amplicons up to 10 kilobases in length, allowing for the retrieval of up to 75% of the human genome with deep sequencing, much higher than the PCR-based WGA methods [69], yet with lower throughput. In addition, the Φ29 DNA polymerase is highly reliable, introducing less errors during the amplification [56]. Nonetheless, MDA is susceptible to biases targeting genomic regions, leading to low uniformity of sequencing coverage across the genome (uneven coverage). Moreover, it can potentially result in allelic dropout (ADO), where one of the two alleles fails to be amplified during the process. Coupling the MDA method with scWGS, single-cell whole exome sequencing (scWES), and targeted sequencing, the resulting sequencing data is well-suited for calling SNVs but not for calling CNAs due to the difficulty in distinguishing genuine CNA events from amplification biases [56, 55, 59].

The last type of WGA methods combines isothermal with PCR, such as displacement DOP-PCR (or PicoPLEX) [70] and multiple annealing and looping-based amplification cycles (MALBAC) [71]. Both methods employ a restricted isothermal amplification, which is subsequently followed by PCR amplification of the generated amplicons from the isothermal step [70, 71]. Despite the relatively low throughput, this strategy allows for a balance between the uniformity and coverage of the genome. As a result, this category of WGA methods is suitable for calling both single SNV and CNA [71, 56, 59], although calling the latter may require additional data normalization [69].

In conclusion, each WGA method has distinct advantages and disadvantages, and none of them excels in every aspect. Therefore, the selection of the most suitable WGA method for scDNA-seq in any experiment relies on the specific research objectives and the questions being addressed.

The immediate outcome of scDNA-seq comprises raw sequencing files in FASTQ format, which can subsequently be aligned to reference genomes. Ultimately, for each cell at each site, this process yields raw read counts for each of the four nucleotides, where the nucleotide indicated by the reference genome is called the *reference nucleotide*, and the other three are called the *alternative nucleotides.* Furthermore, the corresponding sequencing coverage is computed by aggregating these four raw read counts.

The four raw read counts inherently mirror the underlying *genotype* of a cell at a specific site. However, their values experience fluctuations owing to the presence of aforementioned technical artifacts, including amplification errors, sequencing errors, ADOs, and uneven coverage. As a result, the challenge emerges to discern potential mutations from the noisy data. Moreover, somatic mutations can be discerned from germline mutations by analyzing healthy bulk DNA-seq samples collected from the same tissue as the individual cells.

1.3. Probabilistic graphical models

A probabilistic graphical model is the diagrammatic representation of probability distributions [72]. Such a model is composed of *nodes* (or *vertices*), each of which represents a random variable or a group of random variables, and *edges* (or *links*), each of which depicts the probabilistic relationships between the connected nodes. If the edges in a probabilistic graphical model are not directional, it is called a *undirected graphical model*, or a *Markov random field.* Otherwise it is called a *directed graphical model*, where the edges with arrows pointing from a *parent* node to a *child* node. The directional edges in such a model indicate dependencies between the parent and child nodes, represented by *local conditional probabilities.* Here, we focus on a specific type of directed graphical models, namely Bayesian networks, where no direct cycles exist in the graph. In other words, it is not possible in such Bayesian networks to follow a sequence of directed edges and return to the same node.

Nodes in Bayesian networks may have distinct representations, denoting different types of random variables. For instance, Figure 1.1 demonstrates three types of random variables. The shaded circle on the left, marked by y_n, representing the observed data, while the blank circles in the middle, marked by μ and τ, denoting unknown, *hidden* (or *latent*) random variables. For the rest of the nodes on the right, marked by μ_0, τ_0, α and β, they are fixed parameters, which sometimes are also represented by black dots in Bayesian networks. In addition, the rectangle surrounding y_n is called a *plate*, which represents N nodes of which only a single instance y_n is shown explicitly. Those nodes enclosed in a plate, such as $y_n, n = 1, \dots, N$, are independent and identically distributed.

A characteristic that is central to Bayesian networks is that the joint probability distribution of all nodes can be decomposed into the product of the local conditional distribution for each node conditional on its parents defined in the graph [72, 73, 74]. Mathematically speaking, for a Bayesian network with J nodes $x_j, j \in \{1, \dots, J\}$, the joint distribution of $\boldsymbol{x} = \{x_1, \dots, x_j, \dots, x_J\}$ is given by

$$P(\boldsymbol{x}) = \prod_{j=1}^{J} P\left(x_j \,|\, \mathrm{pa}(x_j)\right), \tag{1.1}$$

where $\mathrm{pa}(x_j)$ represents the set of parents of node x_j, and $P\left(x_j \,|\, \mathrm{pa}(x_j)\right)$ is a local conditional probability.

Following Equation (1.1), the joint distribution of all random variables in Figure 1.1 is

$$P(\boldsymbol{y}, \mu, \tau) = \prod_{n=1}^{N} P(y_n \,|\, \mu, \tau) P(\mu \,|\, \mu_0, \tau_0) P(\tau \,|\, \alpha, \beta), \tag{1.2}$$

where $\boldsymbol{y} = \{y_1, \dots, y_n, \dots, y_N\}$.

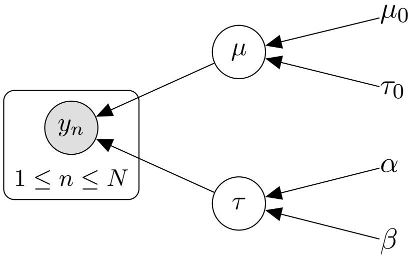


Figure 1.1: **An example of probabilistic graphical models.**

One of the most interesting tasks for graphical models is to estimate from the data the values of hidden random variables. For the graphical model shown in Figure 1.1, one would like to infer the values of hidden random variables μ and τ, as well as their associated uncertainties. Hence, according to Bayes' theorem and by plugging in Equation (1.2), the joint probability of μ and τ conditional on other variables (observed and fixed) writes

$$\begin{aligned} P(\mu, \tau \mid \boldsymbol{y}) &= \frac{P(\boldsymbol{y}, \mu, \tau)}{P(\boldsymbol{y})} \\ &= \frac{\prod_{n=1}^{N} P(y_n \mid \mu, \tau) P(\mu \mid \mu_0, \tau_0) P(\tau \mid \alpha, \beta)}{\iint P(\boldsymbol{y}, \mu, \tau) \, d\mu \, d\tau} \\ &= \frac{\prod_{n=1}^{N} P(y_n \mid \mu, \tau) P(\mu \mid \mu_0, \tau_0) P(\tau \mid \alpha, \beta)}{\iint \prod_{n=1}^{N} P(y_n \mid \mu, \tau) P(\mu \mid \mu_0, \tau_0) P(\tau \mid \alpha, \beta) \, d\mu \, d\tau}, \end{aligned} \tag{1.3}$$

where in the denominator μ and τ are marginalized out from the joint probability distribution to compute $P(\boldsymbol{y})$.

Probabilistic graphical models are naturally interpretable within the Bayesian framework. For the model defined in Figure 1.1 and Equation (1.3), it is straightforward to think $P(\mu, \tau \mid \boldsymbol{y}, \mu_0, \tau_0, \alpha, \beta)$ as the posterior probability of μ and τ, $\prod_{n=1}^{N} P(y_n \mid \mu, \tau)$ as the likelihood of the data $\boldsymbol{y}$ given μ and τ, as well as $P(\mu \mid \mu_0, \tau_0)$ and $P(\tau \mid \alpha, \beta)$ as the prior distribution for μ and τ, respectively. We may further assume specific probability distributions for the observed and hidden random variables, and adjust accordingly the number of fixed parameters for each prior and their values to manifest either informative or uninformative prior beliefs.

For some of Bayesian network models, computation of the marginal probability of a node or a group of nodes is possible using analytical formulas, which is enabled by *exact inference* algorithms. In general, exact inference is efficiently solved by the *sum-product* algorithm via local marginalization and message passing. Similarly, most probable state of each node that is marginalized out can be obtained through the *max-sum* algorithm [75, 76, 72, 73, 74]. However, some of the local probabilities do not allow marginalization that is computationally feasible. Specifically, for continuous hidden random variables, the integrations may lack closed-form analytical solutions, and the dimensionality of the parameter space along with the complexity of the integrand could hinder numerical integration. As for discrete hidden random variables, one needs to marginalize over all possible configurations of these variables, which may be exceedingly large, making exact calculations prohibitively expensive.

To estimate the values and uncertainties of μ and τ in our example Figure 1.1, one could theoretically use the sum-product algorithm to compute the marginal posterior for every hidden random variable. However, this approach may become infeasible in practice due to the computational intractability of the probability of the observed data $P(\boldsymbol{y})$, which serves as the denominator in Equation (1.3). To address this issue, approximation methods like variational inference and sampling techniques offer alternative solutions [72, 73, 74] (for sampling techniques, see also Section Markov chain Monte Carlo below).

A thorough description of probabilistic graphical models can be found in Koller *et al.* [73]. Throughout the rest of the thesis, I use "probabilistic graphical models" or "graphical models" to specifically denote Bayesian networks.

1.4. Statistical phylogenetic models

A phylogenetic tree or evolutionary tree is such a tree that depicts the evolutionary relationships among entities, such as species, based both on similarities and differences in their physical or genetic characteristics [77]. It is a common method in disciplines where evolution is of the concern, such as evolutionary biology [78, 79], epidemiology [80, 81], cancer [82, 83], and linguistics [84, 85].

A phylogenetic tree consists of nodes and branches (or edges). The nodes at the tips of the tree represent existing, observed entities, while the internal nodes represent extinct, unobserved entities. The branches serve as connections between nodes, with lengths representing phylogenetic times [86, 87] and calibrated either by time or evolutionary distance. Phylogenetic trees can be classified into different subtypes based on their characteristics. For example, a rooted phylogenetic tree contains an internal node, called root, that serves as the most recent common ancestor (MRCA) of all entities at the tips of the tree, providing a clear direction of evolutionary history, while an unrooted phylogenetic tree lacks such an internal node. By omitting the root, a rooted phylogenetic tree is converted to an unrooted one, and an unrooted phylogenetic tree can be rooted by incorporating an outgroup into the input data, where the root is the MRCA of the outgroup and other leaves.

Rooted and unrooted phylogenetic trees can exhibit two primary branching patterns: bifurcating and multifurcating. In a bifurcating rooted phylogenetic tree, each internal node has precisely two children, resulting in a binary branching pattern. Conversely, in a multifurcating rooted phylogenetic tree, at least one internal node possesses more than two children, leading to a non-binary branching pattern. Similarly, for the number of neighbors, a bifurcating rooted phylogenetic tree's internal nodes are connected to exactly three neighbors, whereas at least one internal node in a multifurcating rooted phylogenetic tree has more than three neighbors, forming a more complex network of relationships. In this thesis I will focus on the bifurcating rooted phylogenetic tree as it is the most common and biologically motivated form.

The number of possible phylogenetic trees increases exponentially with the number of tips. For example, for J existing nodes, there are in total $(2J-3)!! = (2J-3)(2J-5)\cdots 1$ distinct bifurcating rooted phylogenetic trees [86, 87]. Discovering the optimal rooted phylogenetic tree that best fits the data involves scoring and ranking tree topologies $\mathcal{T}$ along with their associated branch lengths $\boldsymbol{\beta}$. Numerous approaches with distinct scoring functions have been developed for this purpose, including distance-based techniques such as UPGMA (Unweighted Pair Group Method with Arithmetic Mean) and neighbor-joining, as well as parsimony-based methods [88, 86]. However, this thesis employs the statistical methods, which use likelihood functions to score the given $\mathcal{T}$ and $\boldsymbol{\beta}$. The likelihoods are computed considering a rooted phylogenetic tree as a Bayesian network (see Section Probabilistic graphical models) [88, 86, 87].

Typically, the data used for building phylogenetic trees is a set of J aligned sequences of the same length I, called *multiple alignment sequences*, denoted by $\mathcal{D} = [\boldsymbol{d}_1, \dots, \boldsymbol{d}_j, \dots, \boldsymbol{d}_J]$.

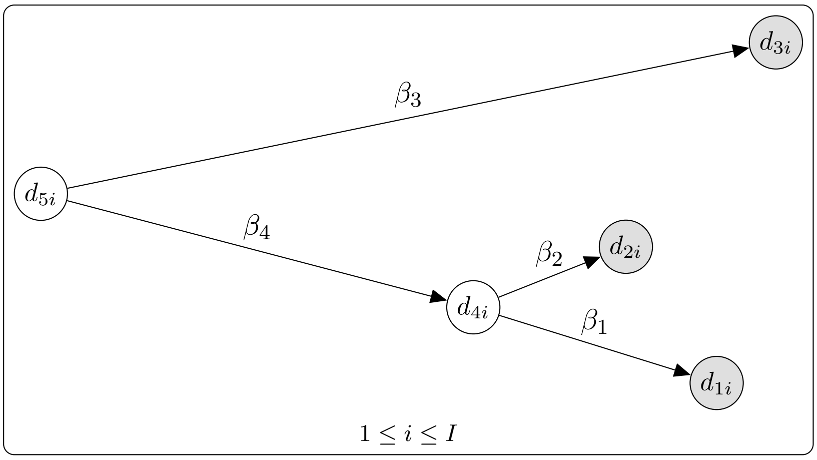


Figure 1.2: **An example of rooted phylogenetic trees.**

Each entry of a sequence represents a state from a fixed, finite set $G = \{g_1, \ldots, g_k, \ldots, g_K\}$. A common assumption made for $\mathcal{D}$ is that the positions or sites within the sequences are independent of one another. Taking as an example the rooted phylogenetic tree shown in Figure 1.2 where $J = 3$, the likelihood function can be expressed as follows:

$$\begin{aligned}
P(\mathcal{D} \,|\, \mathcal{T}, \boldsymbol{\beta}) &= P\left(\boldsymbol{d}_1, \boldsymbol{d}_2, \boldsymbol{d}_3 \,|\, \mathcal{T}, \boldsymbol{\beta}\right) \\
&= \prod_{i=1}^{I} P\left(d_{1i}, d_{2i}, d_{3i} \,|\, \mathcal{T}, \boldsymbol{\beta}\right) \\
&= \prod_{i=1}^{I} \sum_{d_{4i}, d_{5i}} P\left(d_{1i}, d_{2i}, d_{3i}, d_{4i}, d_{5i} \,|\, \mathcal{T}, \boldsymbol{\beta}\right) \\
&= \prod_{i=1}^{I} \sum_{d_{4i}, d_{5i}} \Big[P\left(d_{1i} \,|\, d_{4i}, \beta_1\right) P\left(d_{2i} \,|\, d_{4i}, \beta_2\right) P\left(d_{4i} \,|\, d_{5i}, \beta_4\right) \\
&\qquad\qquad P\left(d_{3i} \,|\, d_{5i}, \beta_3\right) P\left(d_{5i}\right) \Big],
\end{aligned} \tag{1.4}$$

where $\boldsymbol{\beta} = \{\beta_1, \beta_2, \beta_3, \beta_4\}$ with $\beta_j \in \mathbb{R}^+$ being the branch length associated with node d_{ji}, except for d_{5i}, which has no incoming branch. Since a rooted phylogenetic tree is essentially a Bayesian network, the joint probability of all nodes at site i, $P\left(d_{1i}, d_{2i}, d_{3i}, d_{4i}, d_{5i} \,|\, \mathcal{T}, \boldsymbol{\beta}\right)$, is decomposed according to the tree topology $\mathcal{T}$ in such a way that has been described in Section Probabilistic graphical models.

In Equation (1.4), the local conditional probability for d_{ji} depends on its parent and its associated branch length β_j, e.g., $P\left(d_{1i} \,|\, d_{4i}, \beta_1\right)$. Since $d_{ji} \in G$, where G is a set with K discrete elements, the local conditional probability can be represented with a K-by-K *transition probability matrix*

$$R(\beta) = \begin{pmatrix}
P(g_1 \,|\, g_1, \beta) & \cdots & P(g_k \,|\, g_1, \beta) & \cdots & P(g_K \,|\, g_1, \beta) \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
P(g_1 \,|\, g_k, \beta) & \cdots & P(g_k \,|\, g_k, \beta) & \cdots & P(g_K \,|\, g_k, \beta) \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
P(g_1 \,|\, g_K, \beta) & \cdots & P(g_k \,|\, g_K, \beta) & \cdots & P(g_K \,|\, g_K, \beta)
\end{pmatrix}$$

with initial condition $R(0) = I_K$, where I_K is the K-by-K identity matrix. Each row of $R(\beta)$

sums up to 1, indicating a continuous-time Markov chain, on top of which we further assume homogeneity, satisfying the Chapman–Kolmogorov equation [89, 90, 91]:

$$R(\beta + \beta_0) = R(\beta)R(\beta_0). \tag{1.5}$$

A probability distribution $\boldsymbol{\pi} = [\pi_1, \ldots, \pi_k, \ldots, \pi_K]^T, \pi_k \geq 0, \sum_{k=1}^{K} \pi_k = 1$ is called the *limiting distribution* of $R(\beta)$ when $\pi_k = \lim_{\beta \to \infty} P(g_k \,|\, g, \beta)$ for all $g_k, g \in G$. Moreover, a probability distribution $\boldsymbol{\psi} = [\psi_1, \ldots, \psi_k, \ldots, \psi_K]^T, \psi_k \geq 0, \sum_{k=1}^{K} \psi_k = 1$ is called the *stationary distribution*, also called *equilibrium* or *invariant*, of $R(\beta)$ when $\psi^T R(\beta) = \psi^T$ for all $\beta \geq 0$. For a given $R(\beta)$, if the underlying Markov chain is irreducible and aperiodic, then it is called *ergodic* and converges to equilibrium as $\beta \to \infty$, where $\boldsymbol{\psi}$ is unique and equals to $\boldsymbol{\pi}$ (see [89, 90, 91] for more details). When considering the marginal probability of the root node, such as $P(d_{5i})$ in Equation (1.4), it is generally handled in one of three ways: by setting it to the stationary distribution given that the underlying Markov chain is ergodic, by inferring its value from the available data, or by fixing it with a known distribution if such information is readily available.

Based on Equation (1.5), $R(\beta)$ can be written in terms of an *instantaneous transition rate matrix* Q using matrix exponentiation [88, 86, 87]:

$$R(\beta) = \exp(Q\beta), \tag{1.6}$$

where each row in Q sums up to 0. Q is defined in various forms across different disciplines. For instance, in modeling the evolutionary process of DNA, Q can be flexibly represented by a range of models, including JC69 [92], K80 [93], K81 [94], F81 [95], HKY85 [96], T92 [97], TN93 [98], GTR [99], as well as other models [83, 100]. These models exhibit various distinctions, encompassing factors such as the state space representation, which can be nucleotides or genotypes, the nature of the limiting distribution, which is fixed or inferred, and the rate matrix itself, reflecting different research focuses and underlying assumptions.

By further writing Equation (1.4) into

$$\begin{aligned} P(\mathcal{D} \,|\, \mathcal{T}, \boldsymbol{\beta}) = \prod_{i=1}^{I} \sum_{d_{5i}} & P\left(d_{3i} \,|\, d_{5i}, \beta_3\right) P\left(d_{5i}\right) \\ & \times \left[\sum_{d_{4i}} P\left(d_{1i} \,|\, d_{4i}, \beta_1\right) P\left(d_{2i} \,|\, d_{4i}, \beta_2\right) P\left(d_{4i} \,|\, d_{5i}, \beta_4\right) \right], \end{aligned} \tag{1.7}$$

the likelihood can be efficiently computed out using Felsenstein's pruning algorithm [95, 86], a simplified form of the sum-product algorithm for binary trees.

In summary, the use of statistical methods, which involve instantaneous transition rate matrices, is of substantial interest due to their ability to incorporate probabilistic models and analyze phylogenetic relationships through maximum likelihood or Bayesian frameworks [88, 86, 87]. These methods provide a powerful and flexible approach to understanding complex evolutionary patterns and are widely accepted in the field of phylogenetics, where many mature softwares have been developed and are commonly used, such as BEAST 2 [101], RAxML [102], and MrBayes [103]. I will refer to this approach as a "statistical phylogenetic model" in the rest of this thesis, which consists of an instantaneous transition rate matrix Q as well as a phylogenetic tree with topology $\mathcal{T}$ and branch lengths $\boldsymbol{\beta}$.

1.5. Markov chain Monte Carlo

As discussed in Section Probabilistic graphical models, it is infeasible to obtain the posterior distribution of hidden random variables through exact inference in most probabilistic graphical models of practical interest. To address this challenge, the use of approximation methods, particularly sampling techniques (commonly referred to as *Monte Carlo* methods), becomes imperative and is extensively adopted. The content presented in this section is closely in line with the textbook by Bishop *et al.* [72], but also integrates specific reconfigurations and supplementary insights drawn from various sources.

On the general level, the sampling techniques are applied in the context of evaluating the expectation of a function $f(\boldsymbol{Z})$, whose analytical form is intractable to obtain, with respect to a probability distribution $P(\boldsymbol{Z})$, where $\boldsymbol{Z}$ is a set of hidden random variables, either continuous or discrete. The expectation of $f(\boldsymbol{Z})$ writes

$$\mathbb{E}[f(\boldsymbol{Z})] = \int f(\boldsymbol{z}) P(\boldsymbol{z})\, d\boldsymbol{z} = S, \tag{1.8}$$

where $\boldsymbol{z}$ is a realization of $\boldsymbol{Z}$. By generating N independently and identically distributed samples $\boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_n, \ldots, \boldsymbol{Z}_N \sim P(\boldsymbol{Z})$, the sampling techniques approximate S with the mean of the samples

$$\hat{S}_N = \frac{1}{N} \sum_{n=1}^{N} f(\boldsymbol{Z}_n), \tag{1.9}$$

where $\mathbb{E}[\hat{S}_N] = S$, and the variance of $\hat{S}_N$ writes

$$\begin{aligned} \mathrm{Var}\left[\hat{S}_N\right] &= \frac{1}{N^2} \sum_{n=1}^{N} \mathrm{Var}[f(\boldsymbol{Z}_n)] \\ &= \frac{1}{N^2} \sum_{n=1}^{N} \mathbb{E}\left[f^2(\boldsymbol{Z}_n)\right] - \Big[\mathbb{E}[f(\boldsymbol{Z}_n)]\Big]^2 \\ &= \frac{1}{N^2} \sum_{n=1}^{N} \left[\int f^2(\boldsymbol{z}) P(\boldsymbol{z})\, d\boldsymbol{z} - S^2\right] \\ &= \frac{1}{N} \left[\int f^2(\boldsymbol{z}) P(\boldsymbol{z})\, d\boldsymbol{z} - S^2\right]. \end{aligned} \tag{1.10}$$

The probability distribution $P(\boldsymbol{Z})$ is commonly defined in relation to a one-to-one matched probabilistic graphical model, which may include observed random variables $\boldsymbol{X}$, such as Figure 1.1 and Equation (1.3) in Section Probabilistic graphical models, where sampling directly from the posterior distribution $P(\boldsymbol{Z} \mid \boldsymbol{X})$ is often challenging. However, evaluating $P(\boldsymbol{Z} \mid \boldsymbol{X})$ for any $\boldsymbol{z}$ up to a normalizing constant C is comparatively straightforward, which writes

$$P(\boldsymbol{z} \mid \boldsymbol{X} = \boldsymbol{x}) = \frac{1}{C} \widetilde{P}(\boldsymbol{z} \mid \boldsymbol{X} = \boldsymbol{x}), \tag{1.11}$$

where $\widetilde{P}(\boldsymbol{z} \mid \boldsymbol{X} = \boldsymbol{x})$ can easily be evaluated.

There are several sampling techniques available, which produce samples using distinct methodologies. One of the most powerful and widely used approaches is *Markov chain Monte Carlo* (MCMC) [104, 72, 74], known for its scalability to high-dimensional sample spaces while preserving computational efficiency. With the probability distribution for the initial state $\boldsymbol{Z}_0$ specified as $P(\boldsymbol{Z}_0)$, MCMC defines a *transition probability* conditional on the current state, denoted by $P(\boldsymbol{Z} \mid \boldsymbol{Z}_\tau)$, where $\boldsymbol{Z}_\tau$ is the sample at time τ. As a result, the collection of samples constitutes a discrete-time homogeneous Markov chain, where the transition probability distribution $P(\boldsymbol{Z}_{\tau+1} \mid \boldsymbol{Z}_\tau)$ is the same for all τ. Such a Markov chain is constructed specifically for the target distribution, $P(\boldsymbol{Z} \mid \boldsymbol{X})$, being its stationary distribution, which means that

$$P(\boldsymbol{Z}' \mid \boldsymbol{X}) = \sum_{\boldsymbol{Z}} P(\boldsymbol{Z} \mid \boldsymbol{X}) P(\boldsymbol{Z}' \mid \boldsymbol{Z}). \tag{1.12}$$

For a constructed Markov chain, a sufficient (but not necessary) condition for $P(\boldsymbol{Z} \mid \boldsymbol{X})$ being its stationary distribution is to make the Markov chain *time-reversible*, where the transition probability distribution $P(\boldsymbol{Z}' \mid \boldsymbol{Z})$ has a property called *detailed balance* [105, 72], writing

$$P(\boldsymbol{Z} \mid \boldsymbol{X}) P(\boldsymbol{Z}' \mid \boldsymbol{Z}) = P(\boldsymbol{Z}' \mid \boldsymbol{X}) P(\boldsymbol{Z} \mid \boldsymbol{Z}'). \tag{1.13}$$

Summing over $\boldsymbol{Z}$ on both sides of Equation (1.13) gives

$$\begin{aligned} \sum_{\boldsymbol{Z}} P(\boldsymbol{Z} \mid \boldsymbol{X}) P(\boldsymbol{Z}' \mid \boldsymbol{Z}) &= \sum_{\boldsymbol{Z}} P(\boldsymbol{Z}' \mid \boldsymbol{X}) P(\boldsymbol{Z} \mid \boldsymbol{Z}') \\ &= P(\boldsymbol{Z}' \mid \boldsymbol{X}) \sum_{\boldsymbol{Z}} P(\boldsymbol{Z} \mid \boldsymbol{Z}') \\ &= P(\boldsymbol{Z}' \mid \boldsymbol{X}), \end{aligned} \tag{1.14}$$

which shows that a transition probability distribution that adheres to the principle of detailed balance with respect to a distribution results in that distribution being stationary of the underlying Markov chain.

A time-reversible homogeneous Markov chain may have multiple stationary distributions. Therefore, further constraints need to be applied to guarantee the uniqueness of the stationary distribution for the Markov chain. By imposing ergodicity to the Markov chain, the distribution at time τ, $P(\boldsymbol{Z}_\tau)$, converge to equilibrium with unique stationary distribution $P(\boldsymbol{Z} \mid \boldsymbol{X})$ as $\tau \to \infty$, irrespective of the initial distribution $P(\boldsymbol{Z}_0)$. Research has demonstrated that a broad category of homogeneous Markov chains are indeed ergodic, requiring only modest limitations on the stationary distribution and transition probabilities [105].

To properly design a transition probability distribution that meets the requirement of detailed balance, a commonly adopted method within the MCMC framework is *Metropolis-Hastings algorithm* [106]. At time τ, the algorithm first proposes a candidate sample $\boldsymbol{Z}'$ conditional on the current state $\boldsymbol{Z}_\tau$ from a *proposal distribution* $q(\boldsymbol{Z}' \mid \boldsymbol{Z}_\tau)$. The choice of $q(\boldsymbol{Z}' \mid \boldsymbol{Z}_\tau)$ is made deliberately simple, enabling straightforward sampling. Subsequently, the candidate sample $\boldsymbol{Z}'$ is accepted according to an *acceptance probability*

$$\begin{aligned} A(\boldsymbol{Z}', \boldsymbol{Z}_\tau) &= \min\left\{1, \frac{P(\boldsymbol{Z}' \mid \boldsymbol{X})\, q(\boldsymbol{Z}_\tau \mid \boldsymbol{Z}')}{P(\boldsymbol{Z}_\tau \mid \boldsymbol{X})\, q(\boldsymbol{Z}' \mid \boldsymbol{Z}_\tau)}\right\} \\ &= \min\left\{1, \frac{\widetilde{P}(\boldsymbol{Z}' \mid \boldsymbol{X})\, q(\boldsymbol{Z}_\tau \mid \boldsymbol{Z}')}{\widetilde{P}(\boldsymbol{Z}_\tau \mid \boldsymbol{X})\, q(\boldsymbol{Z}' \mid \boldsymbol{Z}_\tau)}\right\}, \end{aligned} \tag{1.15}$$

where Equation (1.11) is plugged, leading to the fact that the normalizing constant C cancels out, and the term $q(\boldsymbol{Z}_\tau \mid \boldsymbol{Z}')/q(\boldsymbol{Z}' \mid \boldsymbol{Z}_\tau)$ is called the *Hastings ratio*. When the proposal distribution $q(\boldsymbol{Z} \mid \boldsymbol{Z}_\tau)$ is symmetric with respect to the current state $\boldsymbol{Z}_\tau$, the Hastings ratio is 1, and the Metropolis-Hastings algorithm reduces to the *Metropolis algorithm* [107].

The transition probability distribution $P(\boldsymbol{Z}' \mid \boldsymbol{Z}_\tau)$ defined by the Metropolis-Hastings algorithm is hence the product of the proposal distribution $q(\boldsymbol{Z}' \mid \boldsymbol{Z}_\tau)$ and the acceptance probability $A(\boldsymbol{Z}', \boldsymbol{Z}_\tau)$, namely $P(\boldsymbol{Z}' \mid \boldsymbol{Z}_\tau) = q(\boldsymbol{Z}' \mid \boldsymbol{Z}_\tau) A(\boldsymbol{Z}', \boldsymbol{Z}_\tau)$. It is straightforward to show that detailed balance holds for the Metropolis-Hastings algorithm using Equation (1.13):

$$
\begin{aligned}
P(\boldsymbol{Z}_\tau \,|\, \boldsymbol{X}) q(\boldsymbol{Z}' \,|\, \boldsymbol{Z}_\tau) A(\boldsymbol{Z}', \boldsymbol{Z}_\tau) &= \min\left\{P(\boldsymbol{Z}_\tau \,|\, \boldsymbol{X})\; q(\boldsymbol{Z}' \,|\, \boldsymbol{Z}_\tau), P(\boldsymbol{Z}' \,|\, \boldsymbol{X})\; q(\boldsymbol{Z}_\tau \,|\, \boldsymbol{Z}')\right\} \\
&= \min\left\{P(\boldsymbol{Z}' \,|\, \boldsymbol{X})\; q(\boldsymbol{Z}_\tau \,|\, \boldsymbol{Z}'), P(\boldsymbol{Z}_\tau \,|\, \boldsymbol{X})\; q(\boldsymbol{Z}' \,|\, \boldsymbol{Z}_\tau)\right\} \\
&= P(\boldsymbol{Z}' \,|\, \boldsymbol{X}) q(\boldsymbol{Z}_\tau \,|\, \boldsymbol{Z}') \\
&\quad \min\left\{1, \frac{P(\boldsymbol{Z}_\tau \,|\, \boldsymbol{X})\; q(\boldsymbol{Z}' \,|\, \boldsymbol{Z}_\tau)}{P(\boldsymbol{Z}' \,|\, \boldsymbol{X})\; q(\boldsymbol{Z}_\tau \,|\, \boldsymbol{Z}')}\right\} \\
&= P(\boldsymbol{Z}' \,|\, \boldsymbol{X}) q(\boldsymbol{Z}_\tau \,|\, \boldsymbol{Z}') A(\boldsymbol{Z}_\tau, \boldsymbol{Z}').
\end{aligned}
\tag{1.16}
$$

Based on Equation (1.15), upon acceptance of the candidate sample $\boldsymbol{Z}'$, the algorithm set $\boldsymbol{Z}_{\tau+1} = \boldsymbol{Z}'$, otherwise $\boldsymbol{Z}'$ is discarded, and $\boldsymbol{Z}_{\tau+1}$ is set to $\boldsymbol{Z}_\tau$. Subsequently, the algorithm draws another candidate sample from $q(\boldsymbol{Z} \,|\, \boldsymbol{Z}_{\tau+1})$ and repeats those aforementioned steps. Assuming that the algorithm runs for N iterations, it will eventually generate a sequence of $N+1$ samples, namely $\boldsymbol{Z}_0, \ldots, \boldsymbol{Z}_n, \ldots, \boldsymbol{Z}_N$, which are serially correlated.

Executing an MCMC chain properly entails careful consideration of several interconnected facets. One of them is to deal with the problem that the initial samples of the MCMC chain are not stationary. Though there is the theoretical guarantee that the distribution of $P(\boldsymbol{Z}_\tau)$ would converge to the stationary distribution as $\tau \to \infty$, the early samples are not guaranteed to be coming from that distribution. Indeed, the greater the initial disparity between the distribution $P(\boldsymbol{Z}_0)$ and the stationary distribution, the more extended the time required for the MCMC chain to run. A commonly employed strategy, known as *burn-in* or *warm-up*, involves retaining samples only after discarding the initial N' ones, where the Markov chain approximately converges to equilibrium starting from the $(N'+1)$-th sample [108]. Other strategies include configuring $\boldsymbol{Z}_0$ by selecting a representative sample from the stationary distribution derived from MCMC chains run preliminarily, thereby avoiding the burn-in phase [109].

Another consideration is to determine the convergence and termination of MCMC chains. Terminating the algorithm prematurely after convergence to equilibrium could lead to imprecise estimations, whereas unnecessarily prolonging the chain may marginally enhance estimations at the expense of wasting computational resources [108]. A pragmatic approach to ascertain the convergence and termination of MCMC chains involves employing diagnostic tools that analyze the collected samples. One of the most widely adopted tools is Gelman and Rubin's method [110], which determines convergence by comparing the between- and the within-chain variance using multiple MCMC chains with overdispersed starting states with respect to the target distribution, though it is at the risk of discarding too many samples [111, 108]. Recently, a modified version of Gelman and Rubin's method was proposed by running only one MCMC chain, where the between-chain variance is replaced by an estimator of the asymptotic variance for the sample mean, enabling to determine both the convergence and the termination [112]. To determine the termination of MCMC chains, one could keep monitoring the *effective sample size* (ESS), which is the number of independent samples equivalent to the correlated samples that are collected so far [113, 108]. The MCMC chain can be terminated once the estimated ESS reaches the prespecified threshold [114].

Upon terminating an MCMC chain, with the collected samples we may directly evaluate Equation (1.9) (following the exclusion of samples from the burn-in phase if applicable). Alternatively, a "thinning" approach can be adopted, wherein every mth sample is retained, primarily for computational reasons rather than statistical considerations [109]. An enormous amount of unthinned samples are typically highly autocorrelated, resulting in marginal benefits when all of them are utilized for assessing Equation (1.9). In comparison, the thinning approach conserves disk space and memory resources, while maintaining a reasonable level of statistical efficacy.

An additional aspect to consider is the selection of proposal distributions, as they play a significant role in influencing the overall effectiveness of the MCMC algorithm. A common choice for continuous state spaces is a Gaussian distribution centered at the current state, leaving its variance parameter adjustable. Large variance tends to result in bold proposals and subsequently low acceptance rates, which is the proportion of accepted proposals in all proposals, while small variance leads to minor changes to the value of the random variable, and hence high acceptance rates. In both cases, the MCMC chain explores the state space inefficiently, leading to long mixing time and slow convergence. An appropriate value for the variance of the proposal distribution is reflected by a proper acceptance rate, which results in well-mixed MCMC chains. It has been shown that the optimal acceptance rate is 0.234 for multivariate Gaussian proposal distributions [115], which have covariance matrices in replace of variances, and 0.44 for one-dimensional Gaussian proposal distributions [115, 116]. One may approach to the optimal acceptance rate manually by adjusting the value for variance through multiple short-run MCMC chains, although this method is laborious and difficult for high dimensions. An alternative approach is *adaptive MCMC*, where the algorithm automatically finds the desired values for the variances of the proposal distributions on the fly [117, 109]. This allows one to start with bold proposals to quickly move to the high probability region of the target distribution, followed by cautious proposals to meticulously explore that region. However, adjusting parameters of proposal distributions freely may destroy the ergodicity of the Markov chain. To ensure that the Markov property holds, a sufficient condition is *diminishing adaptation*, where the adaptation fades out gradually over time [118, 117].

In summary, MCMC is a robust and versatile method for the inference of intricate probabilistic graphical models via sampling from posteriors. MCMC is able to yield exact results given infinite computational resources and time, which, unfortunately, is unattainable in practice. Hence, MCMC is an approximation method.

1.6. Thesis focus and outline

1.6.1. Research challenges

Cancer is a lethal disease featuring ITH, a result of accumulated somatic mutations during the evolutionary process. With the advent of scDNA-seq, genetic materials are measured with the unprecedented resolution of individual cells, facilitating the understanding of ITH using cell phylogeny. However, this great advantage of scDNA-seq comes at a price, namely the elevated technical artifacts mainly originated from the WGA process, including ADOs, uneven coverage, amplification errors and sequencing errors. Thus, the analysis of scDNA-seq requires specific methods which take into account those technical artifacts.

Applying to scDNA-seq, various methods have been developed for either variant calling [119, 120, 121, 122, 123], phylogenetic reconstruction [124, 125, 126, 83, 127, 128, 129, 100], or both [130, 131]. Nonetheless, there still are numerous challenges to be cracked for these two tasks [55]. This thesis is dedicated to the resolution of the challenges outlined below.

Extensive exploitation of information from scDNA-seq

For each cell at each site, raw read counts for four nucleotides and the sequencing coverage from the aligned scDNA-seq data are available. Notably, many existing methods that utilize raw read counts as input tend to consider solely the sequencing coverage and the read count associated with the most representative alternative nucleotide, disregarding the remaining three raw read count values.

However, these overlooked read counts hold equal importance, as they encapsulate crucial insights into amplification and sequencing errors. Inclusion of raw read counts for all four nucleotides holds the potential to enhance the accuracy of error estimation, leading to more accurate identification of genotypes, and to facilitate the inference of the genotype where both alleles mutate to distinct alternative nucleotides.

Furthermore, the distribution of sequencing coverage is frequently overlooked. While this might be a sound choice for scDNA-seq data characterized by uniform coverage, it indicates a loss of valuable information in scDNA-seq data with uneven coverage, as often encountered in protocols utilizing isothermal-based methods for WGA. Incorporating the sequencing coverage distribution explicitly into a model holds the potential to significantly enhance the accuracy of ADO rate estimation. This enhancement, in turn, can greatly facilitate the accurate identification of genotypes.

In summary, one of the major challenges that I tackle in this thesis lies in harnessing the full potential of information within scDNA-seq data when employing raw read counts as input. This encompasses not only the incorporation of raw read counts for all four nucleotides but also the integration of sequencing coverage. While they provide information with respect to distinct types of technical artifacts, their collective utilization holds the promise of substantially enhancing the accuracy of genotype identification.

Variant calling and phylogenetic reconstruction as a joint task

Typically, variant calling and phylogenetic reconstruction are considered independent and sequential tasks, where the called variants are used to reconstruct cell phylogeny. Since the called variants are not absolutely precise, this workflow leads to the propagation of errors from variant calling to phylogenetic reconstruction.

To mitigate this problem, a more effective approach involves conducting variant calling and phylogenetic reconstruction as a joint task. This strategic integration capitalizes on the mutual exchange of information between the two processes, resulting in significantly enhanced accuracy. Indeed, there are models having made efforts towards this direction [130, 131].

However, these models adhere, strictly or partially, to the infinite-sites assumption (ISA), which posits that once a mutation arises, it persists indefinitely – a premise that is frequently violated in reality [132, 133]. Attempts to relax this assumption are often only partial and constrained. This situation leads to the failure of discerning crucial genotypes, including instances where both alleles are mutated.

Another noteworthy limitation of these models lies in their focus on the tree topology, whilst disregarding the invaluable information embedded in branch lengths. These branch lengths serve as crucial indicators of phylogenetic time spans between parent and child nodes. Hence, they hold crucial insights into the temporal dimension of phylogenetic relationships, and offer potential utility for downstream analyses.

The branch lengths are ideally measured by the number of mutations per genomic site. This measurement is most accurate when encompassing data from all sites across the entire genome to reconstruct the cell phylogeny. However, practical considerations necessitate the utilization of data from solely mutated sites, owing to computational efficiency concerns. Consequently, branch lengths are frequently measured by the number of mutations per mutated site, a practice that tends to result in overestimated branch lengths. This bias is known as *acquisition bias* [134, 135], which introduces the potential for biased tree topology inference [134]. It becomes imperative to address this bias in order to enhance the overall precision of cell phylogeny estimation. While methodologies to mitigate this bias have been proposed previously [134, 135], the integration of these techniques into the inference framework remains a complex endeavor.

To sum up, there are substantial benefits of combining variant calling with phylogenetic reconstruction in the analysis of scDNA-seq. Despite progress in this direction, formidable challenges persist and are considered in this thesis – accommodating the intricate realities of mutation dynamics beyond the scope of the ISA, whilst also attaining a holistic cell phylogeny that incorporates both tree topology and branch lengths corrected for acquisition bias.

Accurate identification of genotypes

The fact that scDNA-seq data is particularly noisy substantially hinders the accurate identification of genotypes. Consequently, currently available variant callers may suffer from unsatisfying recall and precision, as well as elevated false positive rate. To alleviate this problem, it is highly relevant to leverage the raw read counts from scDNA-seq to the fullest extent, which corresponds to the challenge of extensive exploitation of information from scDNA-seq.

Moreover, there is evident merit in sharing information across cells that are closely related. The relatedness may be defined in terms of evolutionary distances, which necessitates the utilization of cell phylogeny. This associates with the challenge of jointly calling variants and inferring cell phylogeny.

In summary, the precise determination of genotypes stands as a pivotal precursor to downstream analyses. Such analyses encompass the recognition of driver and passenger mutations, as well as the identification of oncogenes and tumor suppressor genes. In light of the inherent attributes of scDNA-seq, the accurate identification of genotypes remains an imposing and formidable task. This third research challenge of the thesis is intrinsically intertwined with the two aforementioned ones, and resolving the latter stands poised to offer insights into surmounting the former.

1.6.2. Thesis outline

This thesis is composed of two consecutive research projects during my PhD studies. Both projects address the challenges listed above in Section Research challenges, aiming overall at the joint effort of variant calling and cell phylogeny reconstruction. This is achieved by integrating statistical phylogenetic models into a dedicated probabilistic graphical model, where MCMC is subsequently employed for the inference.

SIEVE: joint inference of single-nucleotide variants and cell phylogeny from single-cell DNA sequencing data [1] (Chapter 2)

The first project presents SIEVE (SIngle-cell EVolution Explorer), a statistical method for the joint inference of SNVs and cell phylogeny under the finite-sites assumption (FSA) from scDNA-seq. SIEVE leverages raw read counts for all four nucleotides, incorporates the distribution of sequencing coverage, and corrects for the acquisition bias of branch lengths. SIEVE establishes a general, extensible framework by incorporating the flexible statistical phylogenetic models into a probabilistic graphical model. In our simulations, SIEVE outperforms other methods in phylogenetic reconstruction and variant calling accuracy, especially in the inference of homozygous variants. Applying SIEVE to three real datasets, one for triple-negative breast (TNBC), and two for colorectal cancer (CRC), we find that double mutant genotypes are rare in CRC but unexpectedly frequent in the TNBC samples.

DelSIEVE: joint inference of single-nucleotide variants, somatic deletions, and cell phylogeny from single-cell DNA sequencing data [136] (Chapter 3)

The second project presents DelSIEVE (somatic Deletions enabled SIngle-cell EVolution Explorer), a statistical method that builds upon SIEVE for the inference of somatic deletions in addition to SNVs and cell phylogeny from scDNA-seq. Thanks to the flexible framework established by SIEVE, this extension is straightforward by allowing for somatic deletions in the genotype state space. We prove in the comprehensive simulation study that, in the presence of somatic deletions, DelSIEVE exhibits outstanding performance with respect to identifying somatic deletions and SNVs, while performing comparatively well as SIEVE regarding cell phylogeny reconstruction. We further apply DelSIEVE to the same real datasets analyzed by SIEVE, where rare double mutant and somatic deletion genotypes are found in CRC samples. Intriguingly, for the TNBC sample we identify several somatic deletions, with less single and double mutant genotypes as compared to those previously reported by SIEVE.

1.7. Contribution and funding

1.7.1. Contribution

In both projects, I conceived the underlying statistical models under the supervision of Prof. Ewa Szczurek. I designed and implemented the models, performed simulation studies and data analyses, as well as generated all results using the new models. I wrote both manuscripts.

The outcomes of the thesis was greatly enriched by the invaluable contributions of our collaborators. They performed the CRC28 scDNA-seq experiment, pre-processed the scDNA-seq datasets, provided insightful input and feedback to the models and the analyses, and brought critical comments to the manuscripts.

1.7.2. Funding

The research has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 766030.

Chapter 2

SIEVE: joint inference of single-nucleotide variants and cell phylogeny from single-cell DNA sequencing data

2.1. Background

Intra-tumor heterogeneity is a consequence of accumulated somatic mutations during tumor evolution [137, 138] and the culprit of acquired resistance and relapse in clinical cancer therapy [38, 32, 35]. Phylogenetic inference is a powerful tool to understand the development of intra-tumor heterogeneity in time and space. Variant allele profiles derived from bulk sequencing data have typically been used to reconstruct the tumor phylogeny at the level of clones [139, 140, 82, 141, 142]. More recently, the development of single-cell DNA sequencing (scDNA-seq) [52, 54, 55] has enabled single-nucleotide variant (SNV) calling [119, 120, 130, 121, 122, 123] and phylogeny reconstruction [124, 125, 126, 83, 130, 127, 128, 129, 100] down to the single-cell level.

A statistical phylogenetic model is defined by an instantaneous transition rate matrix, a tree topology and tree branch lengths. Such a model defines a Markov process for the evolution of nucleotides or genotypes [86]. Studying the evolutionary process and estimating important parameters such as the branch lengths using statistical phylogenetic models has a long tradition, benefits from well established theory, and has many applications, such as interpreting temporal cell dynamics [143].

However, compared to statistical phylogenetic models, most methods for phylogeny reconstruction from scDNA-seq operate within a simpler modeling framework. First, although branch lengths are a critical part of a phylogenetic tree and reflect the real evolutionary distances among cells, they are often ignored. Those approaches that do infer branch lengths [83, 100] employ the data from the variant sites and ignore information from *background sites* (that have a wildtype genotype), which may lead to so-called acquisition bias and overestimated branch lengths [134, 135].

Moreover, variant calling and phylogenetic inference are commonly considered independent tasks. Variant calling is typically performed first, and phylogenetic inference is performed on the called variants. However, variant calling, particularly from scDNA-seq data, can be hampered by missing data and low coverage, potentially resulting in wrong calls that could mislead phylogenetic inference. A feasible strategy to alleviate this problem is to integrate tree reconstruction with variant calling [55], where phylogenetic information on cell ancestry is used to obtain more reliable variant calls. Recently developed methods for scDNA-seq data approach this strategy from different perspectives [130, 131]. However, those methods do not operate within the statistical phylogenetic framework, in particular do not infer branch lengths of the tree. Moreover, either they fully follow the infinite-sites assumption (ISA), which is often violated in real datasets [132, 133], or relax this assumption to only a limited extent. As a result, they may miss important events in the evolution of tumors. Thus, methods have not yet been developed which, employing statistical phylogenetic models under the finite-sites assumption (FSA), infer cell phylogeny from raw scDNA-seq data and simultaneously call variants.

To address this, we propose SIEVE (SIngle-cell EVolution Explorer), a statistical method that exploits raw read counts for all nucleotides from scDNA-seq to reconstruct the cell phylogeny and call variants based on the inferred phylogenetic relations among cells. To our knowledge, SIEVE is the first approach that employs a statistical phylogenetic model following FSA, where branch lengths, measured by the expected number of somatic mutations per site, are corrected for the acquisition bias using the data from the background sites, and simultaneously calls variants and allelic dropout (ADO) states from raw read counts data. SIEVE incorporates solutions tailored for scDNA-seq tumor data. First, it includes a trunk in the tree structure, representing the branch joining the healthy root to the most recent common ancestor (MRCA) of the subpopulation of the analyzed cells. As such, the model captures the early, important gene mutations, common for all cells in the trunk. Second, it employs a dedicated probabilistic model of the raw nucleotide read counts at the modeled sites, and discerns between single and double mutations at these sites. Thanks to its flexibility, the model is able to detect 12 different types of genotype transitions, corresponding to nine types of events in evolutionary history. Implemented and available as a package of BEAST 2 [101], a flexible and mature framework using Markov Chain Monte Carlo (MCMC) for statistical phylogenetic modeling, SIEVE allows for benefiting from other packages in this framework. Using simulated data, we assess the performance of our model in comparison to existing methods. To illustrate the functionality of SIEVE, we apply it to datasets from two patients with colorectal (CRC) and one with triple-negative breast cancer (TNBC).

2.2. Methods

SIEVE takes as input raw read count data at candidate SNV sites, accounting for the read counts for three alternative nucleotides and the total depth at each site (Figure 2.1a) and combines a statistical phylogenetic model with a probabilistic graphical model of the read counts, incorporating a Dirichlet-multinomial distribution of the nucleotide counts (Figure 2.1b). The statistical phylogenetic model allows for acquisition and loss of mutations on both maternal and paternal alleles (Figure 2.1c). It considers four possible genotypes, 0/0 (referred to as *wildtype*), 0/1 (*single mutant*), 1/1 (*double mutant*, where the two alternative nucleotides are the same) and $1/1'$ (*double mutant*, where the two alternative nucleotides are different). With these genotypes, SIEVE is able to discern 12 different types of genotype transitions, which can be categorized into nine types of mutation events, namely single mutation, coincident homozygous double mutation, coincident heterozygous double mutation, single back mutation, coincident double back mutation, homozygous single mutation addition, heterozygous single mutation addition, homozygous substitute single mutation, and heterozygous substitute single mutation (Table 2.1). Based on the inferred tree (Figure 2.1d), SIEVE calls the maximum likelihood somatic mutations (Figure 2.1e). With these calls and the recognized

Table 2.1: **12 types of genotype transitions that SIEVE is able to identify, with their interpretation as mutation events.** The genotype transitions correspond to possible changes of genotypes on a branch from the parent node to the child node. If any of these events occurs on independent branches of the phylogenetic tree, it is also considered as a parallel evolution event.

Genotype transition	Mutation event
$0/0 \to 0/1$	Single mutation
$0/0 \to 1/1$	Coincident homozygous double mutation
$0/0 \to 1/1'$	Coincident heterozygous double mutation
$0/1 \to 0/0$	Single back mutation
$1/1 \to 0/1$	Single back mutation
$1/1' \to 0/1$	Single back mutation
$1/1 \to 0/0$	Coincident double back mutation
$1/1' \to 0/0$	Coincident double back mutation
$0/1 \to 1/1$	Homozygous single mutation addition
$0/1 \to 1/1'$	Heterozygous single mutation addition
$1/1' \to 1/1$	Homozygous substitute single mutation
$1/1 \to 1/1'$	Heterozygous substitute single mutation

mutation events on the branches of the tree, we detect parallel evolution in the case when the same event re-occurs on independent branches of the tree. The tree contains a trunk joining the root representing a healthy cell with the most recent common ancestor (MRCA) of the modeled cells, representing the acquisition of clonal mutations at the initial stage of tumor progression. SIEVE leverages the noisy raw read counts to integrate genotype uncertainty into cell phylogeny inference. Benefiting from the inferred cell relationships, SIEVE is able to reliably infer the single-cell genotypes, especially for sites where only few reads are available.

2.2.1. SIEVE model

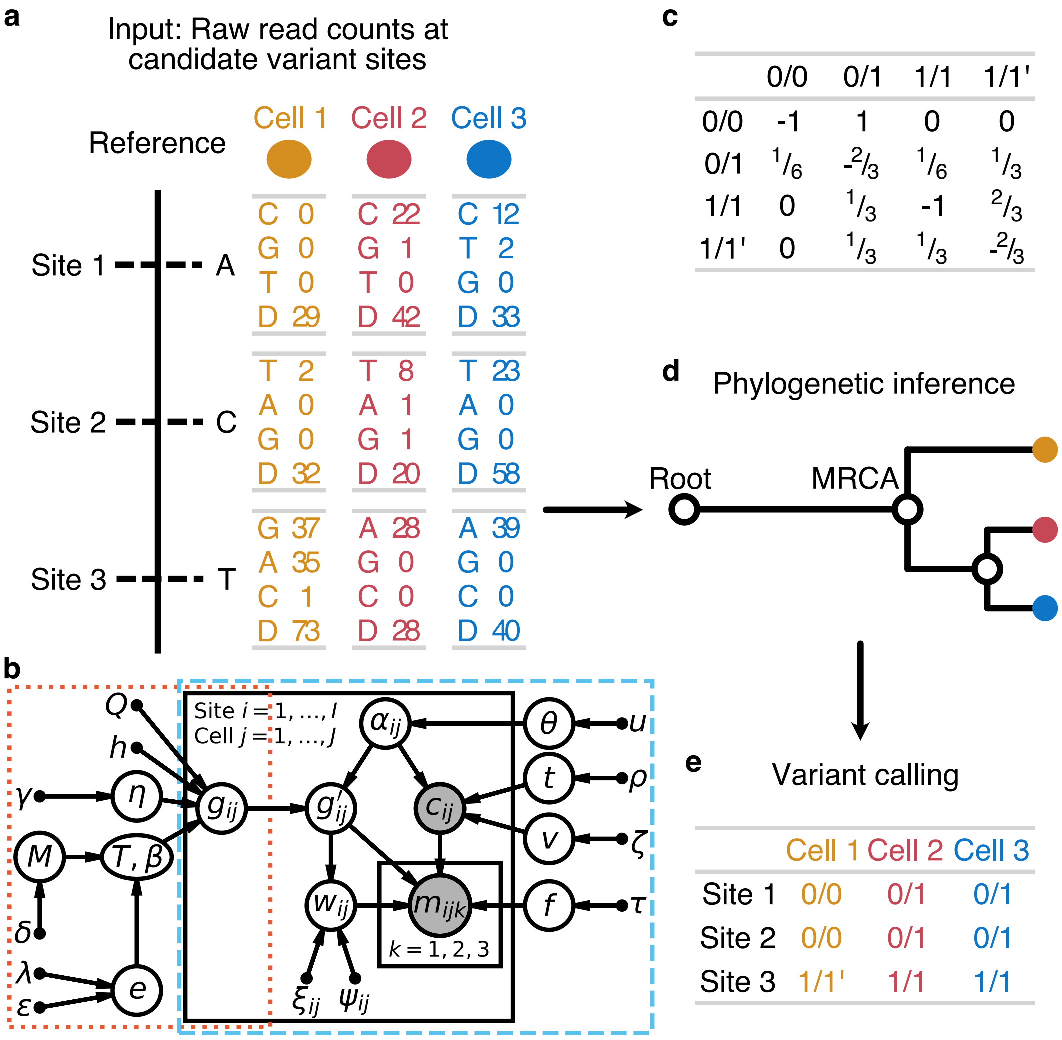
Input data

SIEVE takes as input raw read counts of all four nucleotides at candidate SNV sites (Figure 2.1a). Specifically, for cell $j \in \{1, \ldots, J\}$ at candidate SNV site $i \in \{1, \ldots, I\}$, the input data to SIEVE is in the form of $\mathcal{D}_{ij}^{(1)} = (\boldsymbol{m}_{ij}, c_{ij})$, where $\boldsymbol{m}_{ij} = \{m_{ijk} \,|\, k = 1, 2, 3\}$ corresponds to the read counts of three alternative nucleotides with values in descending order and c_{ij} to the sequencing coverage for cell j and site i. Candidate SNV sites are defined as statistically significant SNVs that could potentially occur in single cells (see Section Candidate site identification).

For scWGS and scWES datasets, raw read counts from I' background sites are denoted $\mathcal{D}^{(2)}$. The number of background sites is used to correct acquisition bias (see Section SIEVE likelihood). For datasets lacking background information (for instance, from targeted sequencing), SIEVE accepts a user-specified number of background sites only for acquisition bias correction.

Candidate site identification

To identify candidate variant sites, we employ a strategy similar to SCIPhI [130]. Specifically, a likelihood ratio test is conducted for SNV detection, but with a modification enabling to capture sites containing double mutant genotypes. To this end, the Beta-Binomial distribution



	0/0	0/1	1/1	1/1'
0/0	-1	1	0	0
0/1	$^1/_6$	$-^2/_3$	$^1/_6$	$^1/_3$
1/1	0	$^1/_3$	-1	$^2/_3$
1/1'	0	$^1/_3$	$^1/_3$	$-^2/_3$

	Cell 1	Cell 2	Cell 3
Site 1	0/0	0/1	0/1
Site 2	0/0	0/1	0/1
Site 3	1/1'	1/1	1/1

Figure 2.1: **Overview of the SIEVE model.** **a**, Input data to SIEVE at candidate SNV sites. For a specific cell at an SNV site, fed to SIEVE are the read counts for all nucleotides: reads of the three alternative nucleotides with values in descending order and the total coverage (denoted by D in **a**). **b**, Graphical representation of the SIEVE model. Bridged by g_{ij}, the genotype for site i in cell j, the orange dotted frame encloses the statistical phylogenetic model, and the blue dashed frame highlights the model of raw read counts. Shaded circle nodes represent observed variables, while unshaded circle nodes represent hidden random variables. Small filled circles correspond to fixed hyper parameters. Arrows denote local conditional probability distributions of child nodes given parent nodes. **c**, The transition rate matrix in the statistical phylogenetic model. During an infinitesimal time interval only one change is allowed to occur. **d**, The cell phylogeny inferred from the data with SIEVE. Not only is the tree topology crucial, but also the branch lengths. The root represents a normal cell, and the only direct child of the root is the most recent common ancestor (MRCA) of all cells. **e**, Variant calling given the inferred cell phylogeny. For further details see Section SIEVE model.

is fitted with free mean and overdispersion parameters at each site across all cells with non-zero variant read counts, and the corresponding likelihood is denoted L_1. Next, another constrained Beta-Binomial distribution is fitted using the same set of cells with fixed mean being 0.25 and free overdispersion, whose likelihood is denoted L_0. As a result, the test statistic $-2\log\frac{L_0}{L_1}$ asymptotically follows the χ^2 distribution with degrees of freedom being 1. The null hypothesis (H_0) is thus that the mean $= 0.25$, and the alternative hypothesis (H_1) is that the mean $\neq 0.25$. A site is classified as candidate variant when the corresponding p-value is larger than 0.05 or the fitted mean is larger than 0.25. This analysis is performed on tumor cells. Normal cells are additionally used to filter out germline mutations. This candidate site identification procedure is implemented in a tool named DataFilter.

The sites identified by DataFilter are referred to as ’candidate’ since they could sometimes be false discoveries due to technical errors in scDNA-seq. Moreover, the actual variant calling, i.e., determination of whether the variant occurs in each of the candidate sites in each cell is performed by SIEVE, and not DataFilter. Notably, all other methods that identify evolutionary trees, including CellPhy [100], SiFit [83], or SCIPhI [130], require an input either actual variants in each cell (CellPhy, SiFit) or the candidate variant sites (SCIPhI). The identification of these candidate sites is crucial for model performance, as it limits the number of sites where the variation may occur, which is much smaller compared to the full set of all possible sites.

Statistical phylogenetic model

The statistical phylogenetic model behind SIEVE includes an instantaneous transition rate matrix, which is defined by a continuous-time homogeneous Markov chain. We consider four possible genotypes $G = \{0/0, 0/1, 1/1, 1/1'\}$, where 0, 1, and $1'$ are used to denote the reference nucleotide, an alternative nucleotide, and a second alternative nucleotide which is different from that denoted by 1, respectively. The fundamental evolutionary events we consider are single mutations and single back mutations. The former happen when 0 mutates to 1, or 1 and $1'$ mutate to each other, while the latter occur when 1 or $1'$ mutates to 0. Hence, genotypes 0/0 and 0/1 represent wildtype and single mutant genotypes, respectively, whereas genotype 1/1 and $1/1'$ represent double mutant genotypes. We intentionally use the non-standard nomenclature of single and double mutants to discern important evolutionary events. In contrast, calling both 0/1 and 1/1’ a heterozygous mutation genotype would be more standard and correct, but would not differentiate between the genotype that has only a single allele changed with respect to the reference (0/1) from the genotype that has two alleles changed (1/1’). We only consider unphased genotypes, so we do not differentiate between 0/1 and 1/0 or between $1/1'$ and $1'/1$.

The joint conditional probability of all cells at SNV site i having genotype $g_{ij} \in G, j = 1, \ldots, J$ is determined according to the statistical phylogenetic model by

$$P\left(\boldsymbol{g}_i^{(L)} \,\middle|\, \mathcal{T}, \boldsymbol{\beta}, Q, h, \eta\right) = \sum_{\boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}} P\left(\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\} \,\middle|\, \mathcal{T}, \boldsymbol{\beta}, Q, h, \eta\right). \tag{2.1}$$

In Equation (2.1), $\boldsymbol{\beta}$ represents the branch lengths measured by the expected number of somatic mutations per site and Q is the instantaneous transition rate matrix of the Markov chain. $\mathcal{T}$ is the rooted binary tree topology, representing the genealogical relations among cells. We specifically require the root of $\mathcal{T}$ to have only one child, representing the most recent common ancestor (MRCA) of all cells. The branch between the root and the MRCA is the trunk of the cell phylogeny. The trunk is one of novelties of our approach, introduced to represent the accumulation of clonal mutations (shared among all cells) in the initial phase of tumor progression. Therefore, with J existing cells, labeled by $\{1, \dots, J\}$, as leaves, $\mathcal{T}$ has J internal hidden ancestor nodes, labeled by $\{J+1, \dots, 2J\}$, and $2J-1$ branches, whose lengths are kept in $\boldsymbol{\beta}$. The trunk is essential for $\mathcal{T}$ to assure that the root, labeled by $2J$, represents a normal ancestor cell even if the data only contains tumor cells. Hence the genotype of the root for SNV site i, denoted $g_{i(2J)}$, is fixed to 0/0. $\boldsymbol{g}_i^{(L)}$ represents the genotypes of J cells as leaves of $\mathcal{T}$, while $\boldsymbol{g}_i^{(A)}$ is the genotypes of all ancestor cells as internal nodes of $\mathcal{T}$. Note that we marginalize the genotypes of the ancestor nodes except for the root. We also consider among-site substitution rate variation following a discrete Gamma distribution with mean equal 1, parameterized by the number of rate categories h and shape η [144]. $\mathcal{T}, \boldsymbol{\beta}, \eta$ in Equation (2.1) are hidden variables, estimated using MCMC (see Section Posterior and MCMC), whereas h is a hyperparameter that is fixed (4 by default). Note that variant calling effectively corresponds to the determination of the values of the variables $\boldsymbol{g}_i^{(L)}$.

In the transition rate matrix Q (Figure 2.1c), each entry denotes a rate from one genotype to another during an infinitesimal time interval Δt. Note that at most one change is allowed to occur in Δt. For instance, the transition of 0/0 moving to 1/1 during Δt is impossible as two single somatic mutations are required; thus, the corresponding transition rate is 0. The transition rate from genotype 0/0 to 0/1 represents the somatic mutation rate and is set to 1. The back mutation rate is measured relatively to the somatic mutation rate and therefore is $^1/_3$.

With the genotype state space G defined, for a given branch length β, the underlying four-by-four transition probability matrix $R(\beta)$ of the Markov chain is represented using matrix exponentiation of the product of Q and β as $R(\beta) = \exp(Q\beta)$ [86].

Model of raw read counts

The probability of observing the input data $\mathcal{D}_{ij}$ for cell j at site i is factorized as

$$P(\mathcal{D}_{ij}) = P(\boldsymbol{m}_{ij} \,|\, c_{ij}) P(c_{ij}), \tag{2.2}$$

where the first component is the model of nucleotide read counts and the second the model of sequencing coverage.

Model of sequencing coverage. After single-cell whole-genome amplification (sc-WGA) some genomic regions are more represented than others. After scDNA-seq, this results in an uneven coverage along the genome, much more than in the case of bulk sequencing. Here, to model the sequencing coverage c in the presence of overdispersion, we employ a negative binomial distribution.

$$P(c \,|\, p, r) = \binom{c+r-1}{r-1} p^r (1-p)^c, \tag{2.3}$$

with parameters p and r. We reparameterize the distribution with $p = {}^{\mu}/_{\sigma^2}$ and $r = {}^{\mu^2}/_{\sigma^2 - \mu}$, where μ and σ^2 are the mean and the variance of the distribution of the sequencing coverage c, respectively.

Theoretically, each cell j at site i has its specific μ_{ij} and σ_{ij}^2 parameters, which, however, are impossible to be estimated freely. Hence, we make additional assumptions and pool the data for better estimates, adapting the approach of [145]. We assume that μ_{ij} and σ_{ij}^2 have the following forms, respectively:

$$\begin{aligned} \mu_{ij} &= \alpha_{ij} t s_j, \\ \sigma_{ij}^2 &= \mu_{ij} + \alpha_{ij}^2 v s_j^2. \end{aligned} \tag{2.4}$$

In Equation (2.4), t is the mean of allelic coverage (the expected coverage per allele) and v is the variance of allelic coverage. We estimate t and v with MCMC (see Section Posterior and MCMC). $\alpha_{ij} \in \{1, 2\}$ is a hidden random variable denoting the number of sequenced alleles for cell j at site i. According to the statistical phylogenetic model, both alleles are expected to be sequenced. However, due to the frequent occurrence of allelic dropout (ADO) during scWGA, there are cases where only one allele is amplified and therefore α_{ij} is 1. Equation (2.4) reflects the fact that the expected sequencing coverage and its raw variance are proportional to the number of sequenced alleles. Note that inferring the hidden variable α_{ij} corresponds to identifying occurrences of ADO events, and hence the ability of SIEVE to perform ADO calling. We denote the prior distribution of α_{ij}

$$\begin{cases} P(\alpha_{ij} = 1 \,|\, \theta) = \theta, \text{ if ADO occurs,} \\ P(\alpha_{ij} = 2 \,|\, \theta) = 1 - \theta, \text{ otherwise,} \end{cases} \tag{2.5}$$

where θ is a parameter corresponding to the the probability of ADO occurs, i.e., the ADO rate, which is estimated using MCMC.

In Equation (2.4), s_j is the size factor of cell j which makes sequencing coverage from different cells comparable and is estimated directly from the sequencing coverage using

$$\hat{s}_j = \underset{i:c_{ij} \neq 0}{\text{median}} \frac{c_{ij}}{\left(\prod_{\substack{j'=1 \\ c_{ij'} \neq 0}}^{J'} c_{ij'} \right)^{\frac{1}{J'}}}, \tag{2.6}$$

where J' is the number of cells with non-zero coverage at a site. By taking into account only the non-zero values, the estimate $\hat{s}_j$ is not affected by the missing data, which is prevalent in scDNA-seq.

Model of nucleotide read counts. We denote the genotype affected by ADO $g'_{ij} \in G \bigcup \{0/\text{-}, 1/\text{-}\}$, where 0/- and 1/- are the results of ADO occurring to g_{ij}. For instance, 0/- is caused either by 0 dropped out from 0/0 or by 1 dropped out from 0/1. Then the probability of g'_{ij} is denoted by

$$P\left(g'_{ij} \,|\, g_{ij}, \alpha_{ij}\right), \tag{2.7}$$

which is defined at length in Table 2.2.

Table 2.2: **Definition of the distribution of g'_{ij} conditional on g_{ij} and α_{ij} for SIEVE.**

g'_{ij}	g_{ij}	α_{ij}	$P(g'_{ij} \,\|\, g_{ij}, \alpha_{ij})$
0/0	0/0	2	1
0/-	0/0	1	1
0/1	0/1	2	1
1/1	1/1	2	1
1/-	1/1	1	1
1/1′	1/1′	2	1
1/-	1/1′	1	1
0/-	0/1	1	½
1/-	0/1	1	½
Others			0

We model the read counts of three alternative nucleotides $\boldsymbol{m}_{ij}$ given the sequencing coverage c_{ij} with a Dirichlet-multinomial distribution as

$$P(\boldsymbol{m}_{ij} \,|\, c_{ij}, \boldsymbol{a}_{ij}) = \frac{F(c_{ij}, a_{ij0})}{\prod_{k=1:m_{ijk}>0}^{3} F(m_{ijk}, a_{ijk}) F(c_{ij} - \sum_{k=1}^{3} m_{ijk}, a_{ij4})}, \tag{2.8}$$

with parameters $\boldsymbol{a}_{ij} = \{a_{ijk} \,|\, k = 1, \ldots, 4\}$ and $a_{ij0} = \sum_{k=1}^{4} a_{ijk}$. F is a function in the form of

$$F(x, y) = \begin{cases} xB(y, x), & \text{if } x > 0, \\ 1, & \text{otherwise}, \end{cases} \tag{2.9}$$

where B is the beta function. Note that $c_{ij} - \sum_{k=1}^{3} m_{ijk}$ is the read count of the reference nucleotide.

To improve the interpretation of Equation (2.8), we reparameterize it with $\boldsymbol{a}_{ij} = w_{ij}\boldsymbol{f_{ij}}$, where $\boldsymbol{f_{ij}} = \{f_{ijk} \,|\, k = 1, \ldots, 4\}, \sum_{k=1}^{4} f_{ijk} = 1$ is a vector of expected frequencies of each nucleotide and w_{ij} represents overdispersion. $\boldsymbol{f_{ij}}$ are categorical hidden variables dependent on g'_{ij}:

$$\boldsymbol{f}_{ij} = \begin{cases} \boldsymbol{f}_1 = \left(\frac{1}{3}f, \frac{1}{3}f, \frac{1}{3}f, 1 - f\right), & \text{if } g'_{ij} = 0/0 \text{ or } 0/\text{-}, \\ \boldsymbol{f}_2 = \left(\frac{1}{2} - \frac{1}{3}f, \frac{1}{3}f, \frac{1}{3}f, \frac{1}{2} - \frac{1}{3}f\right), & \text{if } g'_{ij} = 0/1, \\ \boldsymbol{f}_3 = \left(1 - f, \frac{1}{3}f, \frac{1}{3}f, \frac{1}{3}f\right), & \text{if } g'_{ij} = 1/1 \text{ or } 1/\text{-}, \\ \boldsymbol{f}_4 = \left(\frac{1}{2} - \frac{1}{3}f, \frac{1}{2} - \frac{1}{3}f, \frac{1}{3}f, \frac{1}{3}f\right), & \text{if } g'_{ij} = 1/1', \end{cases} \tag{2.10}$$

where f is the expected frequency of nucleotides whose existence is solely due to technical errors during sequencing. To be specific, f is defined as the effective sequencing error rate including amplification (where a nucleotide is wrongly amplified into another one during scWGA) and sequencing errors.

w_{ij} is also a categorical hidden variable dependent on g'_{ij}:

$$w_{ij} = \begin{cases} w_1, & \text{if } g'_{ij} = 0/0, 0/\text{-}, 1/1, \text{ or } 1/\text{-}, \\ w_2, & \text{if } g'_{ij} = 0/1 \text{ or } 1/1', \end{cases} \tag{2.11}$$

where w_1 is wild type overdispersion and w_2 is alternative overdispersion.

By plugging in Equations (2.10) and (2.11), Equation (2.8) is equivalently represented with

$$P(\boldsymbol{m}_{ij} | c_{ij}, g'_{ij}, f, w_{ij}) = \begin{cases} P_{0/0} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 0/0, \boldsymbol{f}_1, w_1\right), \\ P_{0/\text{-}} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 0/\text{-}, \boldsymbol{f}_1, w_1\right), \\ P_{0/1} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 0/1, \boldsymbol{f}_2, w_2\right), \\ P_{1/1} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 1/1, \boldsymbol{f}_3, w_1\right), \\ P_{1/\text{-}} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 1/\text{-}, \boldsymbol{f}_3, w_1\right), \\ P_{1/1'} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 1/1', \boldsymbol{f}_4, w_2\right). \end{cases} \tag{2.12}$$

Note that $P_{0/0}$ and $P_{0/\text{-}}$ share the same $\boldsymbol{f}$ and w_1, showing that the model of nucleotide read counts is not enough to discriminate 0/0 from 0/-, and so do $P_{1/1}$ and $P_{1/\text{-}}$. In such cases, incorporating the model of sequencing coverage helps resolve the entanglement.

To understand Equation (2.12), first take $P_{0/0}$ as an example. Theoretically, no alternative nucleotides are supposed to exist if no technical errors occur. Thus, any observations of any alternative nucleotides can only result from technical errors, and the expected frequency of the reference nucleotide is accordingly adjusted to $1 - f$. For another example $P_{0/1}$, say the reference nucleotide is A and the alternative nucleotide is C, and both their read count frequencies are supposed to be $^1/_2$ if no technical errors occur. For the other two alternative nucleotides, G and T, their observations could only result from technical errors, and both their frequencies are $^f/_3$. Moreover, either A or C may be sequenced as a different nucleotide (each with probability $1/2$). In the former case, the frequency of A decreases by $^f/_2$. In the latter case, if C is sequenced as A (with probability $^f/_3$) the frequency of A increases by $^1/_2 \times {}^f/_3$. Overall, the frequency of A decreases by $^f/_3$, resulting in $^1/_2 - {}^f/_3$.

f, w_1 and w_2 in Equation (2.12) are estimated with MCMC.

SIEVE likelihood

We denote the conditional variables in Equation (2.1) as $\Theta = \{\mathcal{T}, \boldsymbol{\beta}, Q, h, \eta\}$ and those in the model of raw read counts as $\Phi = \{t, v, \theta, f, w_1, w_2\}$. Given the input data $\mathcal{D}^{(1)}$ and $\mathcal{D}^{(2)}$, the log-likelihood of the SIEVE model is

$$\log \mathcal{L}(\Theta, \Phi) = \log \mathcal{L}^{(1)}(\Theta, \Phi) + \log \mathcal{L}^{(2)}(f, w_1), \tag{2.13}$$

where $\mathcal{L}^{(1)}$ is the tree likelihood corrected for acquisition bias computed from candidate SNV sites in $\mathcal{D}^{(1)}$, while $\mathcal{L}^{(2)}$ is the likelihood computed from background sites in $\mathcal{D}^{(2)}$, referred to as the background likelihood. Equation (2.13) does not contain $g_{ij}, g'_{ij}, \alpha_{ij}$ since they are marginalized out (see below).

Since we only use data from SNV sites to compute the tree likelihood, the tree branch lengths $\boldsymbol{\beta}$ are prone to be overestimated [134, 135]. The overestimation of $\boldsymbol{\beta}$ due to only using data from SNV sites is called acquisition bias, which is corrected in SIEVE according to [146]:

$$\log \mathcal{L}^{(1)} = \log P\left(\mathcal{D}^{(1)} \,\middle|\, \Theta, \Phi\right) + I' \log \left(\frac{1}{I} \sum_{i=1}^{I} C_i\right), \tag{2.14}$$

where the first component is the uncorrected tree log-likelihood for SNV sites, and C_i in the second component is the likelihood of SNV site i being invariant (see below). The regularization term $I' \log \left(\frac{1}{I} \sum_{i=1}^{I} C_i\right)$ renders SIEVE in favor of trees with short branch lengths where $\mathcal{L}^{(1)}$ is large due to the increasing averaged C.

To compute the uncorrected tree log-likelihood, we marginalize out α_{ij} and g'_{ij}:

$$
\begin{aligned}
P(\boldsymbol{m}_{ij}, c_{ij}|g_{ij}, \Phi) &= P(\boldsymbol{m}_{ij}, c_{ij}|g_{ij}, f, w_{ij}, t, v, \theta) \\
&= \sum_{\alpha_{ij}, g'_{ij}} P\left(\boldsymbol{m}_{ij}, c_{ij}, \alpha_{ij}, g'_{ij} \,\middle|\, g_{ij}, f, w_{ij}, t, v, \theta\right) \\
&= \sum_{\alpha_{ij}, g'_{ij}} P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij}, f, w_{ij}\right) P\left(g'_{ij} \,\middle|\, g_{ij}, \alpha_{ij}\right) \\
&\qquad \times P(c_{ij} \,|\, \alpha_{ij}, t, v) P(\alpha_{ij} \,|\, \theta) \\
&= \begin{cases}
P_{0/0} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta) \\
\qquad + P_{0/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot \theta, \text{ if } g_{ij} = 0/0, \\
P_{0/1} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta) \\
\qquad + \dfrac{1}{2}(P_{0/\text{-}} + P_{1/\text{-}}) \cdot P(c_{ij}|\alpha_{ij} = 1, t, v) \cdot \theta, \text{ if } g_{ij} = 0/1, \\
P_{1/1} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta) \\
\qquad + P_{1/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot \theta, \text{ if } g_{ij} = 1/1, \\
P_{1/1'} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta) \\
\qquad + P_{1/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot \theta, \text{ if } g_{ij} = 1/1',
\end{cases}
\end{aligned}
\tag{2.15}
$$

where $P_{0/0}, P_{0/\text{-}}, P_{0/1}, P_{1/1}, P_{1/\text{-}}, P_{1/1'}$ are defined in Equation (2.12) and $P\left(g'_{ij} \,\middle|\, g_{ij}, \alpha_{ij}\right)$ is defined in Equation (2.7). In the second line of Equation (2.15), the probability is factorized out according to Figure 2.1b.

To compute $\log P\left(\mathcal{D}^{(1)} \,\middle|\, \Theta, \Phi\right)$ in Equation (2.14), we assume that the SNV sites evolve independently and identically. By plugging Equations (2.1) and (2.15), $\log P\left(\mathcal{D}^{(1)} \,\middle|\, \Theta, \Phi\right)$ is denoted by

$$
\begin{aligned}
\log P\left(\mathcal{D}^{(1)} \,\middle|\, \Theta, \Phi\right) &= \sum_{i=1}^{I} \log \sum_{\boldsymbol{g}_i^{(L)}} P\left(\mathcal{D}_i^{(1)} \,\middle|\, \boldsymbol{g}_i^{(L)}, \Phi\right) \sum_{\boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}} P\left(\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\} \,\middle|\, \Theta\right) \\
&= \sum_{i=1}^{I} \log \sum_{\boldsymbol{g}_i^{(L)}} \left[\prod_{j=1}^{J} P(\boldsymbol{m}_{ij}, c_{ij} \,|\, g_{ij}, \Phi) \right. \\
&\qquad\qquad \left. \times \sum_{\boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}} P\left(\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\} \,\middle|\, \Theta\right) \right] \\
&= \sum_{i=1}^{I} \sum_{j=1}^{J} \log \sum_{\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}} \left[P(\boldsymbol{m}_{ij}, c_{ij} \,|\, g_{ij}, \Phi) \right. \\
&\qquad\qquad\qquad \left. \times P\left(\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\} \,\middle|\, \Theta\right) \right],
\end{aligned}
\tag{2.16}
$$

which is efficiently computed out by Felsenstein's pruning algorithm [95], with the extension of the model of raw read counts applied on leaves. Specifically, the Fenselstein's pruning algorithm is applied to an extended tree $\mathcal{T}$, where additional leaf nodes corresponding to the data are attached at the bottom of $\mathcal{T}$: for each node corresponding to genotype g_{ij} there is a leaf node added, corresponding to data $(\boldsymbol{m}_{ij}, c_{ij})$, and the transition probability between the genotype node and the leaf is given by Equation (2.15). For I candidate SNV sites, J cells and K genotype states in G (for SIEVE $K = 4$), the time complexity of Felsenstein's pruning algorithm is $\mathcal{O}(IJK^2)$.

C_i in Equation (2.14) is determined similarly to Equation (2.16) by computing the joint probability of observing the data $\mathcal{D}_i^{(1)}$ and $\boldsymbol{g}_i^{(L)} = 0/0$:

$$\begin{aligned}
C_i &= P\left(\mathcal{D}_i^{(1)}, \boldsymbol{g}_i^{(L)} = 0/0 \,\middle|\, \Theta, \Phi\right) \\
&= P\left(\mathcal{D}_i^{(1)} \,\middle|\, \boldsymbol{g}_i^{(L)} = 0/0, \Phi\right) \sum_{\boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}} P\left(\boldsymbol{g}_i^{(L)} = 0/0, \boldsymbol{g}_i^{(A)} \setminus \left\{g_{i(2J)}\right\} \,\middle|\, \Theta\right) \\
&= \prod_{j=1}^{J} P\left(\boldsymbol{m}_{ij}, c_{ij} \,|\, g_{ij} = 0/0, \Phi\right) \sum_{\boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}} P\left(\boldsymbol{g}_i^{(L)} = 0/0, \boldsymbol{g}_i^{(A)} \setminus \left\{g_{i(2J)}\right\} \,\middle|\, \Theta\right).
\end{aligned} \tag{2.17}$$

Formally, to compute the background likelihood, we should account for the fact that the background sites, similarly to the variant sites, also evolve under the phylogenetic model and involve similar computations as above. This, however, would result in a large additional computational burden due to the large number of background sites compared to the variant sites. Thus, to estimate the background log-likelihood efficiently, we make several simplifications and compute it only approximately. First, we assume that across I' background sites each cell has the same genotype $0/0$ and both alleles are covered. We further ignore the model of sequencing coverage and the tree log-likelihood in the computations. As a result, by employing an alternative expression of Dirichlet-multinomial distribution $\log \mathcal{L}^{(2)}$ is efficiently obtained as

$$\begin{aligned}
\log \mathcal{L}^{(2)}(f, w_1) &= \sum_{i=1}^{I'} \sum_{j=1}^{J} \log P_{0/0} \\
&= \sum_{i=1}^{I'} \sum_{j=1}^{J} \log \left[\frac{\Gamma(w_1)\Gamma(c_{ij}+1)}{\Gamma(c_{ij}+w_1)} \prod_{k=1}^{3} \frac{\Gamma(m_{ijk} + \frac{1}{3} f w_1)}{\Gamma(\frac{1}{3} f w_1)\Gamma(m_{ijk}+1)} \right. \\
&\qquad \left. \times \frac{\Gamma(c_{ij} - \sum_{k=1}^{3} m_{ijk} + (1-f)w_1)}{\Gamma((1-f)w_1)\Gamma(c_{ij} - \sum_{k=1}^{3} m_{ijk} + 1)} \right] \\
&= I'J \left[\log \Gamma(w_1) - 3 \log \Gamma\left(\frac{1}{3} f w_1\right) - \log \Gamma((1-f)w_1) \right] \\
&\quad + \sum_{c=1}^{\max(c_{ij})} N_c (\log \Gamma(c+1) - \log \Gamma(c+w_1)) \\
&\quad + \sum_{k=1}^{3} \sum_{m_k=1}^{\max(m_{ijk})} N_{m_k} \left(\log \Gamma\left(m_k + \frac{1}{3} f w_1\right) - \log \Gamma(m_k+1) \right) \\
&\quad + \sum_{c - \sum_{k=1}^{3} m_k = 1}^{\max(c_{ij} - \sum_{k=1}^{3} m_{ijk})} N_{c - \sum_{k=1}^{3} m_k} \left(\log \Gamma\left(c - \sum_{k=1}^{3} m_k + (1-f)w_1\right) \right. \\
&\qquad \left. - \log \Gamma\left(c - \sum_{k=1}^{3} m_k + 1\right) \right),
\end{aligned} \tag{2.18}$$

where $P_{0/0}$ is defined in Equation (2.12). N_c, N_{m_k} for $k = 1, 2, 3$, and $N_{c - \sum_{k=1}^{3} m_k}$ represent, across I' background sites and J cells, the unique occurrences of sequencing coverage c, of alternative nucleotide read counts m_1, m_2, m_3, and of reference nucleotide read counts $c - \sum_{k=1}^{3} m_k$, respectively. In Equation (2.18), some items, namely $\log\Gamma(c+1)$, $-\log\Gamma(m_k+1)$ for $k = 1, 2, 3$, and $-\log\Gamma(c - \sum_{k=1}^{3} m_k + 1)$, only depends on the data, which remain constants during MCMC. Therefore, they are ignored in the computation of background likelihood. It is clear that the background likelihood helps estimate f and w_1.

The time complexity of Equation (2.18) is $\mathcal{O}(c)$ with c being the number of unique values in the set of values representing sequencing coverage and read counts for all four nucleotides across all cells and background sites. Since IJK^2 is usually much larger than c, the overall time complexity of model likelihood is $\mathcal{O}(IJK^2)$.

Priors

To define priors for model parameters and for the tree coalescent, we employ the prior distributions defined in BEAST 2. We impose on $\mathcal{T}$ and $\boldsymbol{\beta}$ in Equation (2.1) a prior distribution following the Kingman coalescent process with an exponentially growing population. The tree prior is parameterized by scaled population size M and exponential growth rate q, and is denoted by

$$P(\mathcal{T}, \boldsymbol{\beta} \mid M, e), \tag{2.19}$$

whose analytical form is defined in [147]. M and e are hidden random variables and are estimated using MCMC. Note that, by default, M represents the number of time units, e.g., the number of years, and the mutation rate is measured by the number of mutations per time unit per site. Their product results in the unit of branch length, i.e., the number of mutations per site. Since scDNA-seq data usually does not contain temporal information as a result of collecting samples at the same time, it is impossible to differentiate M from the mutation rate. However, if the mutation rate is known, one could alternatively estimate a time-calibrated cell phylogeny.

As prior distributions, we assign to M

$$P(M \mid \delta) = \frac{1}{\delta}, \tag{2.20}$$

where δ is the current proposed value of M. Note that this is supposed to be normalized to define a proper probability distribution, but this form is sufficient to define a proper posterior (see Section Posterior and MCMC).

For e we choose

$$e \mid \lambda, \epsilon \sim \text{Laplace}(\lambda, \epsilon), \tag{2.21}$$

where we choose mean $\lambda = 10^{-3}$ and scale $\epsilon = 30.7$ (default in the BEAST 2 software). We choose an exponential distribution as the prior for η in Equation (2.1):

$$\eta \mid \gamma \sim \exp(\gamma), \tag{2.22}$$

where $\gamma = 1$.

For the model of sequencing coverage described in Equations (2.3) and (2.4), we set the prior for t within a large range of values with

$$t \mid \rho \sim \text{Uniform}(0, \rho), \tag{2.23}$$

where $\rho = 1000$, and the prior for v with

$$v \mid \zeta \sim \exp(\zeta), \tag{2.24}$$

where $\zeta = 25$. In terms of θ in Equation (2.5), it also has a uniform prior:

$$\theta \,|\, u \sim \text{Uniform}(0, u), \tag{2.25}$$

where $u = 1$.

For the model of nucleotide read counts described in Equations (2.10) to (2.12), we choose an exponential prior for f:

$$f \,|\, \tau \sim \exp(\tau), \tag{2.26}$$

where $\tau = 0.025$, and a log normal prior for both w_1 and w_2:

$$\begin{aligned} w_1 \,|\, \xi_1, \psi_1 &\sim \text{Log-Normal}(\xi_1, \psi_1), \\ w_2 \,|\, \xi_2, \psi_2 &\sim \text{Log-Normal}(\xi_2, \psi_2), \end{aligned} \tag{2.27}$$

where we choose for w_1 the log-transformed mean $\xi_1 = 3.9$ (150 for untransformed) and the standard deviation $\psi_1 = 1.5$, and for w_2 the log-transformed mean $\xi_2 = 0.9$ (10 for untransformed) and the standard deviation $\psi_2 = 1.7$. Specifically, the mean is log-transformed using

$$\xi_{\text{transformed}} = \log(\xi_{\text{untransformed}}) - \frac{\psi^2}{2}.$$

These specific values reflect our belief that w_1 is greater than w_2, and are chosen in such a way that both distributions cover a large range of possible values for w_1 and w_2.

Posterior and MCMC

With the model likelihood and priors defined, the posterior distribution of the unknown parameters is

$$\begin{aligned} & P\left(\mathcal{T}, \boldsymbol{\beta}, M, e, \eta, t, v, \theta, f, w_1, w_2 \,\middle|\, \mathcal{D}^{(1)}, \mathcal{D}^{(2)}\right) \\ =& \frac{1}{Z} P\left(\mathcal{D}^{(1)}, \mathcal{D}^{(2)} \,\middle|\, \mathcal{T}, \boldsymbol{\beta}, \eta, t, v, \theta, f, w_1, w_2\right) \\ & \times P(\mathcal{T}, \boldsymbol{\beta} \,|\, M, e) P(M \,|\, \delta) P(e \,|\, \lambda, \epsilon) P(\eta \,|\, \gamma) \\ & \times P(t \,|\, \rho) P(v \,|\, \zeta) P(\theta \,|\, u) P(f \,|\, \tau) \\ & \times P(w_1 \,|\, \xi_1, \psi_1) P(w_2 \,|\, \xi_2, \psi_2), \end{aligned} \tag{2.28}$$

where Z is a normalization constant, representing the probability of the observed data.

Since the posterior distribution does not have a closed-form analytical formula, we employ the MCMC algorithm with Metropolis-Hastings kernel to sample from the posterior distribution in Equation (2.28). Given the current state of the parameters q, we propose a new state q^* according to proposal distributions $P(q^*|q)$ that assure the reversibility and ergodicity of the Markov chain. With one parameter changed a time, q^* is accepted with probability

$$\min\left\{1, \frac{P\left(\mathcal{T}^*, \boldsymbol{\beta}^*, M^*, e^*, \eta^*, t^*, v^*, \theta^*, f^*, w_1^*, w_2^* \,\middle|\, \mathcal{D}^{(1)}, \mathcal{D}^{(2)}\right) P(q \,|\, q^*)}{P\left(\mathcal{T}, \boldsymbol{\beta}, M, e, \eta, t, v, \theta, f, w_1, w_2 \,\middle|\, \mathcal{D}^{(1)}, \mathcal{D}^{(2)}\right) P(q^* \,|\, q)}\right\}, \tag{2.29}$$

where the normalization constant Z cancels out after plugging in Equation (2.28).

For sampling the structure of the cell phylogeny, we take advantage of proposal distributions implemented in the BEAST 2 software [147] and modify them to make sure they are compatible with our tree topology, so that the sampled trees are binary and contain a trunk. Specifically, the tree branch lengths are changed by scaling the heights of the internal nodes. For tree topological exploration, we use the Wilson-Balding move to perform subtree pruning and regrafting. Specifically, a random node and half of its subtree is pruned and reattached to a random branch not belonging to the moved subtree. A subtree-slide move is also used, where a random node and half of its subtree slides either upwards or downwards along branches and cross at least one node. Both those two moves include changes to the lengths of some branches. The final type of move swaps two randomly selected subtrees.

For sampling unknown parameters, we perform either scaling operations or random Gaussian walks.

SIEVE runs with a two-stage sampling strategy. In the first stage the acquisition bias correction is switched off and all parameters are explored, while in the second stage the acquisition bias correction is turned on and parameters not affecting branch lengths are fixed with their estimates from the previous stage. This two-stage strategy proved to yield more accurate parameter and tree estimates than a strategy where both parameters and tree would be explored at once, with the acquisition bias correction enabled. Additionally, the initial tree in the second stage is set to the tree summarized from the first stage.

Variant calling, ADO calling, maximum likelihood gene annotation, and mutation event classification

During the sampling process $\boldsymbol{g}_i^{(L)}$, $\boldsymbol{g}_i^{(A)}$, g'_{ij} and α_{ij} (Equations (2.1), (2.15) and (2.16)) are hidden variables that are marginalized out. Therefore, to obtain estimates of these hidden variables, we infer their maximum likelihood configuration with the max-sum algorithm [72], using the maximum clade credibility tree [148] and parameters estimated from the MCMC posterior samples.

To be specific, by determining the maximum likelihood genotypes of the leaves ($\boldsymbol{g}_i^{(L)}$), we are able to call variants. By inferring the maximum likelihood g'_{ij} and α_{ij}, the ADO state is determined. Moreover, by computing the maximum likelihood genotypes of the internal nodes ($\boldsymbol{g}_i^{(A)}$), SIEVE maps mutations to specific tree branches.

Mutation events are classified into different categories based on the corresponding genotype transitions (Table 2.1). The single mutation ($0/0 \rightarrow 0/1$) happens when an allele of the wildtype is mutated. The coincident homozygous double mutation ($0/0 \rightarrow 1/1$) refers to the case when both alleles of the wildtype are mutated to the same alternative nucleotide, while the coincident heterozygous double mutation ($0/0 \rightarrow 1/1'$) refers to the case when both alleles of the wildtype are mutated to different alternative nucleotides. The single back mutation ($0/1 \rightarrow 0/0$, $1/1 \rightarrow 0/1$ and $1/1 \rightarrow 0/1$) happens when a mutated allele mutates back to the reference nucleotide, while the coincident double back mutation ($1/1 \rightarrow 0/0$ and $1/1' \rightarrow 0/0$) happens when both mutated alleles mutate back to the reference nucleotide. The homozygous single mutation addition ($0/1 \rightarrow 1/1$) refers to the case when the unmutated allele of the single mutant genotype mutates to the same alternative nucleotide as the mutated allele, while for the heterozygous single mutation addition ($0/1 \rightarrow 1/1'$) the unmutated allele mutates to an alternative nucleotide different from the mutated allele. For the homozygous substitute single mutation ($1/1' \rightarrow 1/1$), one of the mutated alleles mutates to the same alternative nucleotide as the other mutated allele, while for the heterozygous substitute single mutation ($1/1 \rightarrow 1/1'$) one of the mutated alleles mutates to another alternative nucleotide.

Summary of model assumptions

Taken together, SIEVE makes several assumptions about the evolutionary process behind the observed single cell data. First, the model assumes that the genome is diploid. This assumption stands behind most of our model equations. In order not to violate this model assumption, one should pre-process the data to exclude non-diploid regions. On the other hand, this comes with the cost of excluding sites in these regions. Leaving such sites introduces discrepancy with the assumption, but might give more statistical power for model inference. Thus, we leave this decision of excluding copy number altered regions as a preprocessing step to the user.

Another important assumption, made by most methods for phylogenetic reconstruction, is that the sites are independently affected by the mutational process. This assumption is key to computational performance, as it allows to factorize the model likelihood across the sites.

One more assumption made behind SIEVE is that the phylogenetic tree has a trunk, which connects a healthy cell as the root and its only child as the MRCA of all cells in the data. When there are only tumor cells in the data, the MRCA represents the first tumor cell founding the tumor tissue, and since many clonal mutations accumulate during the foundation process of tumor, the trunk is expected to be long. When both healthy and tumor cells are available, the MRCA is also a healthy cell, and since only very few, if any, mutations accumulate between two healthy cells, the trunk is expected to be short. The incorporation of the trunk comes in handy in practice not only because it can help to identify normal cells mixed with tumor cells, but also because an outgroup is not needed to root the tree.

Finally, SIEVE follows the finite sites assumption (FSA), which is both more general and more plausible than the infinite sites assumption (ISA). Events violating the ISA are expected biologically and probabilistically [132, 133]. It is important to note that per definition, SIEVE and other models that follow the FSA are well suited to model both cases (when ISA is violated and not). More specifically, the ISA is a special case of the FSA, so the models that follow the FSA also account for ISA.

Summary of evolutionary features accounted for by the model

In contrast to other models, SIEVE is able to identify 12 types of genotype transitions, corresponding to nine types of mutation events (Table 2.1). Moreover, when such events affecting the same site are detected on more than one branch, our model is able to detect parallel evolution. This is because SIEVE considers four genotype states (0/0, 0/1, 1/1, $1/1'$) and is based on the underlying Markov process model that follows the FSA. Among those nine mutation events, only one of them, namely the single mutation, corresponding to the transition from genotype state 0/0 to 0/1, is accounted for by models that follow the ISA. Moreover, SiFit, which follows the FSA but has a restricted genotype state space compared to SIEVE, is also unable to identify all 12 genotype transitions that are detectable by SIEVE.

Moreover, SIEVE's another feature is its compatibility with molecular clock models implemented in BEAST 2, including the strict, relaxed and random local molecular clock model [149, 150]. The use of these models opens the door for the estimation of divergence times (event timing) and substitution rates using sound statistical models.

Importantly, we separate these features from model assumptions, as these are properties that SIEVE supports in an unforced manner. For instance, SIEVE is able to identify 12 genotype transitions, but not all of them are necessarily to appear in the tree.

2.2.2. ScDNA-seq data simulator

In order to benchmark the performance of SIEVE against those of other published methods, we simulated scDNA-seq data by modifying CellCoal [151] (commit 594e063). In contrast to CellCoal, the sequencing coverage is generated according to Equations (2.3) to (2.6). Given the sequencing coverage, read counts are simulated with a multinomial distribution including errors. Input configuration follows the one described for CellCoal [151].

The simulator mimics both the biological evolution and the sequencing process. We first generated a binary genealogical cell lineage tree following the coalescent process assuming a strict molecular clock and created a reference genome where each site was initialized by the reference genotype with one of the four nucleotides. With a specific mutation rate, each site was evolved independently along the tree according to a rate matrix which contains ten diploid genotypes encoded with nucleotide pairs (Table 5.1). The rate matrix allows mutations and back mutations, where the probability of the latter is $^1/_3$ of the former. All simulated sites for which at least one cell has a non-reference genotype are considered as true SNV sites. Next, we added at most one ADO to cell j at site i according to the ADO rate. If ADO happens, the number of sequenced alleles α_{ij} drops from two to one. We recorded the true ADO states across cells for the SNV sites. Size factors for cells in Equation (2.4) were sampled from a normal distribution (mean $=1.2$, variance $=0.2$). Using the negative binomial distribution, we simulated the sequencing coverage with given t and v. Based on the ADO-affected genotype and sequencing coverage, the read count for each nucleotide was simulated using a multinomial distribution with a given amplification error rate and sequencing error rate.

2.2.3. Simulation design

We designed simulations to compare multiple methods in different aspects. The benchmarking framework was built using Snakemake [152].

Simulations only considering SNVs

We assumed that the tumor cell samples belonged to an exponentially growing population (growth rate $=10^{-4}$) with an effective population size of 10^4. The number of tumor cells was chosen to be either 40 or 100. We selected three mutation rates: 10^{-6}, 8×10^{-6}, and 3×10^{-5}. For different mutation rates, different total number of sites were chosen to result in around 1000 SNV sites for 100 cells (1.3×10^5 sites for 10^{-6}, 2×10^4 sites for 8×10^{-6}, and 6.5×10^3 sites for 3×10^{-5}), as well as between 250 to 1000 SNV sites for 40 cells (8×10^4 sites for 10^{-6}, 2×10^4 sites for 8×10^{-6}, and 5×10^3 sites for 3×10^{-5}). Additionally, we varied t and v in Equations (2.3) and (2.4) to simulate different *coverage qualities*. For high quality data, we chose high mean ($t=20$) and low variance ($v=2$) of allelic coverage. For medium quality data, we chose high mean ($t=20$) and medium variance ($v=10$). For low quality data, we chose low mean ($t=5$) and high variance ($v=20$), which was specifically created to mimic the CRC28 dataset.

Other important parameters in the simulation were fixed as follows: in Equation (2.5) $\theta=0.163$, in Equation (2.12) $w_1=100$ and $w_2=2.5$, and both amplification error rate and sequencing error rate were 10^{-3}, which resulted in the effective sequencing error rate $f\approx2\times10^{-3}$ in Equation (2.12).

We designed in total 18 simulation scenarios, each repeated 20 times.

Simulations considering both SNVs and CNAs

To add CNAs, we selected a set of datasets generated as described above, using the following parameters: 40 cells, medium mutation rate (8×10^{-6}) and medium coverage quality ($t=20, v=10$). Two levels of CNA prevalence were simulated: around $^1/_3$ or $^2/_3$ of all genomic sites. A site could contain CNAs occurring at an early or at a late stage during the evolutionary process with equal probabilities, and the corresponding number of CNAs was sampled in

$\{0, 1, 3, \ldots, 10\}$. For a site containing early stage CNAs, the probability of a cell carrying such events was sampled uniformly from the $[\,^2\!/_3, 1\,]$ interval, while for late stage CNAs the probability was sampled from the $(\,0, ^1\!/_3\,]$ interval. If a site in a cell was sampled to be affected by CNAs, a specific allele was selected for CNA with probability 0.5. To this end, if the sampled CNA value was 0, the read counts for the site and the cell was simply set to 0. Otherwise, we directly manipulated the simulated read counts of the chosen allele by multiplying the CNA value minus one, where the one CNA copy was retained for the other unchosen allele.

The simulated datasets after adding CNAs were stored in two versions: with or without genomic sites containing CNAs, both of which were used as input for all methods.

It is important to note that in these simulations, the CNAs were added independently of the phylogenetic structure. It is thus expected that we were simulating the most pessimistic scenario, as CNAs introducing bias in the data in the same way for phylogenetically related cells could in fact help with better phylogeny reconstruction.

2.2.4. Measurement of cell phylogeny accuracy and quality of variant calling

To assess the accuracy of the cell phylogeny reconstruction considering branch lengths, we computed the branch score (BS) distance from the inferred tree to the true tree [153]. For any two trees, this difference is computed as:

$$d_{BS} = \sqrt{\sum_i \left(l_{1i}^{(s)} - l_{2i}^{(s)}\right)^2 + \sum_i \left(l_{1i}^{(u)}\right)^2 + \sum_i \left(l_{2i}^{(u)}\right)^2}. \tag{2.30}$$

where $l_{ji}^{(s)}$ represents the length of a branch shared by both trees, and $l_{ji}^{(u)}$ represents the length of a branch i that is unique for tree j.

To assess the accuracy of the cell phylogeny reconstruction ignoring branch lengths we used the normalized Robinson-Foulds (RF) distance [154]:

$$d_{RF} = \frac{n_1^{(u)} + n_2^{(u)}}{n_1 + n_2}, \tag{2.31}$$

where n_j denotes the total number of branches in tree j, while $n_j^{(u)}$ represents the number branches exclusive of tree j.

Thus, BS distance and normalized RF distance values equal to 0 indicate a perfect tree reconstruction. For SIEVE and SiFit, we compute both normalized RF distance and BS distance in the rooted tree mode. For CellPhy, we compute these metrics in the unrooted tree mode as it infers an unrooted tree from data only containing tumor cells. Since SCIPhI reports a rooted tree without branch lengths, we can only compute the normalized RF distance. BS distance and normalized RF distance values were computed using the R package phangorn [155].

To evaluate the variant calling and ADO calling results, we computed precision, recall, F1 score and false positive rate (FPR). For variant calling, we separately compared the performance in calling the single mutant genotype and double mutant genotypes. In particular, when we evaluated the accuracy of single mutant genotype calling, any identification of double mutant genotypes whose true genotype is single mutant genotype was counted as a false negative. Moreover, we analyzed two different types of false positives in single mutant genotype calling. The first type corresponds to single mutation calls for sites where the true genotype is a wildtype genotype. The second type are single mutant calls for sites where the true genotype is a double mutant.

For SIEVE and Monovar, we computed the recall, precision, F1 score, and FPR for single mutant genotype calling and double mutant genotype calling. For SCIPhI, we only computed metrics for single mutant genotype calling as it does not call double mutant genotypes. Moreover, we evaluated the accuracy of calling ADO states only for SIEVE, as it is the only method that is able to call them.

2.2.5. Configurations of methods

For Monovar (commit 68fbb68), we used the true values of θ and f as priors for false negative rate and false positive rate and default values for other options.

For SCIPhI (commit 34975f7), we ran it with default options and 5×10^5 iterations.

To run CellPhy (commit 832f6c2) and SiFit (commit 9dc3774), we fed the required data with variants called by Monovar. For CellPhy, we piped the data in VCF format and initialized the tree search with three parsimonious trees. We instructed the tool to use a built-in rate matrix with ten genotypes (GT10), a stationary nucleotide frequency distribution learned from the data (FO), an error model applied to the leaves (E), and the Gamma model of site-wise substitution rate variation (G). For SiFit, we fed the input data as a ternary matrix and used the true values of θ and f as the prior for false negative rate and the estimated false positive rate, respectively. We ran it with 2×10^5 iterations.

On the simulated data, we ran SIEVE with a strict molecular clock model for 2×10^6 and 1.5×10^6 iterations for the first and the second sampling stage, respectively. On the real datasets, we used a log-normal relaxed molecular clock model to take into consideration branch-wise substitution rate variation. To achieve better mixed Markov chains, we employed an optimized relaxed clock model in [156] instead of the default one in BEAST 2.

Since more parameters are added when using the relaxed molecular clock model, we ran the analysis with 3×10^6 iterations for the first stage and 2.5×10^6 iterations for the second, respectively. Note that the parameters introduced by the relaxed molecular clock model are also explored in the second sampling stage. The SNVs were then annotated using Annovar (version 2020 Jun. 08) [157]. In the main text, the tree was plotted using ggtree [158] and the genotype heatmap was plotted using ComplexHeatmap [159].

2.2.6. Run time analysis

Repeated five times, we used a simulation scenario with the following parameters for run time analysis: medium mutation rate (8×10^{-6}) and medium coverage quality ($t = 20, v = 10$). SiFit and SCIPhI were run in the default, single-thread mode, while CellPhy and SIEVE were run in both single- and multi-thread mode, where different numbers of threads were provided to achieve their highest efficiency. SiFit, SCIPhI and the two stages of SIEVE were run for 10^6 iterations, respectively. With bootstrap applied, CellPhy was run with the default setting (a maximum of 1,000 replicates with a possible early-stopping). This analysis was performed on a server with 64 cores (AMD Ryzen Threadripper 3990X 64-Core Processor) and 256 GB memory.

2.3. Results

2.3.1. SIEVE accurately estimates tree topology and branch lengths.

We first evaluated the accuracy of SIEVE in inferring the simulated cell phylogeny with branch lengths using the BS distance [153] (Figure 2.2a). We compared to CellPhy [100] and

SiFit [83], which were fed with the variant calls from Monovar [119]. Here, we gave SiFit an advantage of setting the true positive error rate used in the simulation. Thanks to the acquisition bias correction, SIEVE reports branch lengths as expected number of somatic mutations per site, while CellPhy and SiFit per SNV site. SCIPhI [130] does not infer branch lengths, hence its BS distance could not be computed. SIEVE consistently outperformed CellPhy and SiFit, regardless of the number of cells, mutation rate and coverage quality. This may be because, in contrast to SIEVE, CellPhy and SiFit do not model raw reads and, importantly for the BS distance, do not correct the inferred branch lengths for acquisition bias. We also found that the BS distance of SIEVE had a negative nonlinear association with the number of background sites (Figure 5.1), explaining the relatively greater differences under higher mutation rates. These results proved the necessity for correcting the acquisition bias with enough background sites to obtain accurate branch lengths.

As the BS distance is dominated by the branch lengths, we further assessed SIEVE's accuracy in inferring the tree structure using the normalized RF distance [154]. Compared to CellPhy, SiFit and SCIPhI (Figure 2.2b), SIEVE was the most robust method to changes of mutation rate, number of cells and coverage quality. When the data hardly contained mutations violating the ISA (mutation rate being 10^{-6}, with less than 0.1% double mutant genotypes and at most 1% SNV sites with parallel mutations), all methods achieved a similar median RF distance (around 0.15-0.3). Since in contrast to SCIPhI, SIEVE, CellPhy and SiFit employ statistical phylogenetic models following FSA, this indicates that models following FSA are also applicable to data evolving under the ISA. SIEVE outperformed CellPhy and SiFit when the number of cells and the mutation rate increased. When the data clearly violated the ISA (mutation rates being 8×10^{-6} and 3×10^{-5}, with 0.02%-0.3% and 0.1%-1% double mutant genotypes, as well as 2%-8% and 10%-27% SNV sites with parallel mutations indicative of FSA, respectively), SCIPhI inferred reasonable tree topologies from datasets with a small number of cells (40). However, its performance dramatically dropped with 100 cells, especially when the data was of medium or high coverage quality. The behaviour of SCIPhI might be related to its estimation of ADO rate and single mutant genotype calling in these scenarios.

2.3.2. SIEVE accurately infers parameters in the model of raw read counts.

We next investigated the accuracy of parameter estimates, including *effective* sequencing error rate, ADO rate, and wildtype and alternative overdispersion (Figure 5.2). Here, the effective sequencing error rate (Figure 5.2a) takes into account both amplification and sequencing error rates in scDNA-seq. Wildtype and alternative overdispersion are parameters in the distribution of nucleotide read counts related to different genotypes. The former corresponds to genotype $0/0$ and $1/1$, while the latter to genotype $0/1$ and $1/1'$. SIEVE accurately inferred most parameters in all simulated scenarios regardless of the number of cells, mutation rate and coverage quality. Although SIEVE's accuracy of estimating ADO rate slightly decreased with the coverage quality, it still was the best among the competing methods. For data with medium and high coverage quality, 100 cells and higher mutation rates (8×10^{-6} and 3×10^{-5}), SCIPhI tended to overestimate ADO rates.

2.3.3. SIEVE accurately calls single and double mutations.

Next, we assessed SIEVE's performance in calling the single mutant genotype (Figure 2.2c,d, Figure 5.3a,b, Figure 5.4). As opposed to Monovar, recall for SIEVE and SCIPhI increased with the number of cells but was less sensitive to the coverage quality (Figure 2.2c). The recall

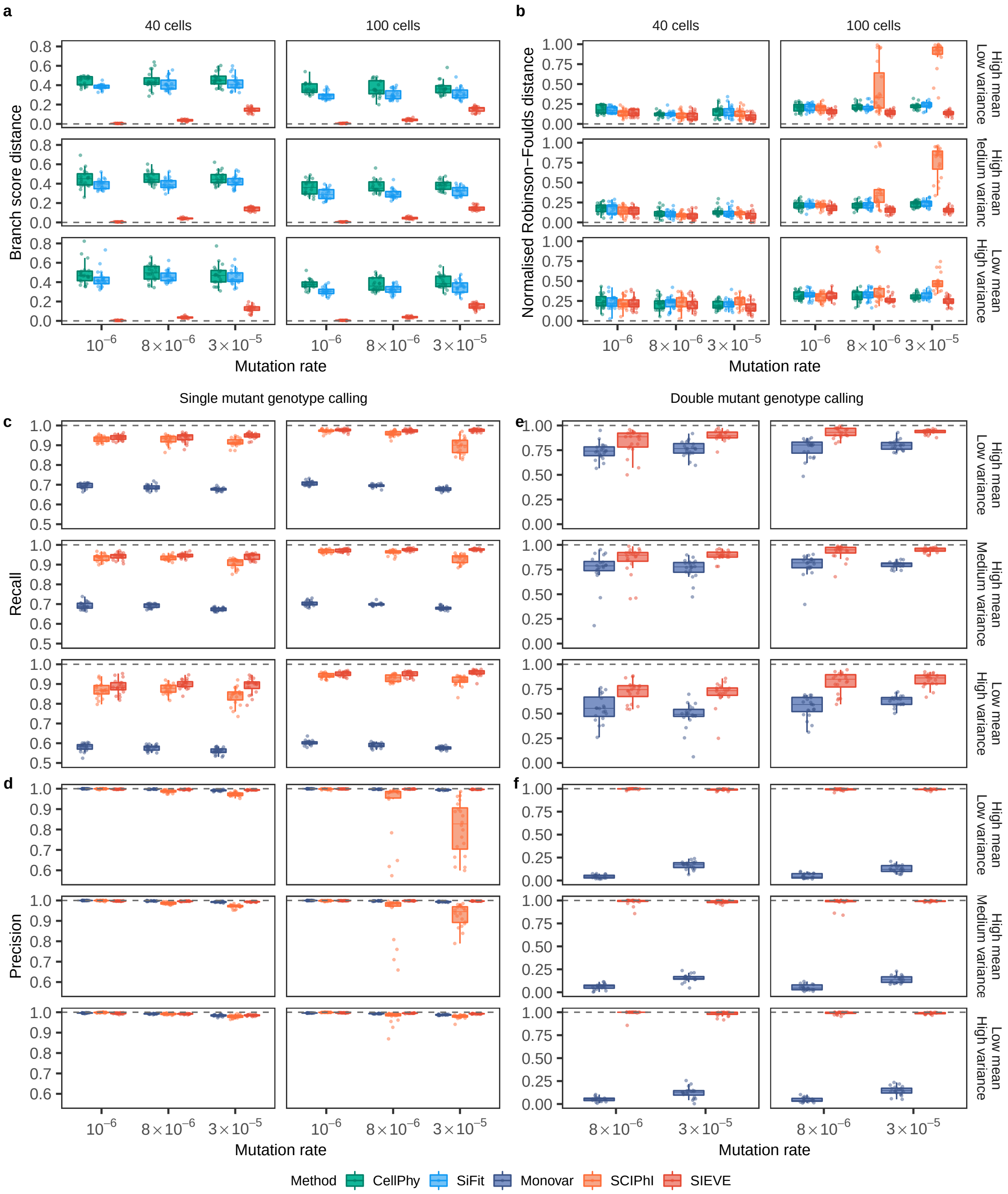


Figure 2.2: **Benchmarking result of the SIEVE model.** Varying are the number of tumor cells, mutation rate and coverage quality. Each simulation is repeated $n = 20$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. **a-b**, Box plots of the tree inference accuracy measured by the BS distance where the branch lengths are taken into account (**a**) and the normalised RF distance where only tree topology is considered (**b**). **c-d**, Box plots of the single mutant genotype calling results measured by the fraction of true positives respectively in the ground truth positives, i.e., the sum of true positives and false negatives, (recall, **c**) as well as in the predicted positives, i.e., the sum of true positives and false positives, (precision, **d**). **e-f**, Box plots of the double mutant genotype calling results measured by recall (**e**) and precision (**f**), where the variant calling results when mutation rate is 10^{-6} are omitted as very few double mutant genotypes are generated (less than 0.1%).

of SIEVE was higher than that of SCIPhI by 0.16%-18.55% and that of Monovar by 28.89%-71.74%. Unlike Monovar, both SIEVE and SCIPhI benefit from the information provided by cell phylogenies. We speculate that the advantage of SIEVE over SCIPhI stems from the use of raw read counts for all nucleotides, while SCIPhI only employs the sequencing coverage and the read count of the most prevalent alternative nucleotide.

Moreover, SIEVE and Monovar achieved comparable precision (Figure 2.2d) and false positive rates (Figure 5.3a) regardless of the number of cells, mutation rate and coverage quality. However, this did not hold for SCIPhI. By analysing the types of false positives among the predicted single mutant genotypes (Figure 5.4), we found that SCIPhI tended to miscall wildtype genotypes as single mutant genotype (i.e., 0/0 are called as 0/1) (Figure 5.4a). This occurred with high mutation rates (8×10^{-6} and 3×10^{-5}), especially in scenarios where SCIPhI inferred inaccurate trees (Figure 2.2b) and overestimated ADO rates (Figure 5.2b). The reason is twofold. First, the ISA upon which SCIPhI builds naturally limits its application to data following FSA. Second, under these scenarios, SCIPhI tends to mistake sites with no variant support for ADO events, and hence its high ADO rate. SIEVE avoids such mistakes by leveraging a model of sequencing coverage (see Section SIEVE model), thereby accounting for the related overdispersion and correctly estimating the ADO rate. We also noticed that when data clearly violated ISA, both Monovar and SCIPhI miscalled more double mutant genotypes as the single mutant genotype than SIEVE (Figure 5.4b).

We then focused on the results of double mutant genotype calling (Figure 2.2e,f, Figure 5.3c,d), where SCIPhI was excluded as it is unable to call such mutations. The recall of double mutant genotypes for SIEVE and Monovar increased with the number of cells and the coverage quality (Figure 2.2e), while SIEVE showed higher recall for such genotypes than Monovar. Moreover, SIEVE outperformed Monovar with high precision (almost 1, Figure 2.2f) and low false positive rate (almost 0, Figure 5.3c).

2.3.4. SIEVE accurately calls ADOs for data of adequate coverage quality.

We further assessed SIEVE's performance in ADO calling (Figure 5.5), where there are no published methods for us to compare with. When calling ADOs, SIEVE's performance was independent of the number of cells or mutation rate, but highly dependent on the coverage quality. The reason is that SIEVE calls ADOs by inferring the number of sequenced alleles, assuming it is proportional to the observed sequencing coverage. Consequently, for data with medium and high coverage quality the average F1 score of ADO calling was high (0.86 and 0.93, respectively), whereas for data with low coverage quality, which is typical for current scDNA-seq data, the ADO calling performance deteriorated, with average F1 score being only 0.10. Since the coverage quality of real data is low, we do not report ADO calling results for all real datasets analyzed below (Table 5.2).

2.3.5. SIEVE accurately infers cell phylogenies and calls variants in the presence of copy number aberrations (CNAs).

Both SIEVE and two compared methods, CellPhy and SCIPhI, work with the assumption that the genomes of the cells are diploid. SiFit allows deletions, thus considering copy number 1 or 2. Occurrences of CNAs change the copy number for some of the sites. Leaving such sites in the data introduces discrepancy with the assumption, but may give more statistical power for model inference. To investigate the degree to which the performance of SIEVE and other models is affected by CNAs, we considered simulation scenarios where both deletions and amplifications were added, by changing the copy number to any integer from the $[0, 10]$

interval that is different than 2. We varied the amount of genomic sites having CNAs in either small or large amount ($^1/_3$ or $^2/_3$ of all sites, respectively), and all methods were run both with CNA sites included and excluded from the input data.

The presence of CNAs had very little effect on the performance of inferring the simulated cell phylogeny with branch lengths by all evaluated methods. Indeed, the BS distances obtained by the methods were at a similar level, regardless of the presence of the CNAs and their amount (Figure 5.6a). In contrast, the presence of CNAs worsened the performance of all methods in the task of inferring the topology of phylogeny, as measured by the normalized RF distance. When the CNAs were present in a small amount, the normalized RF distance for all methods was only slightly increased, regardless of the inclusion of the CNA sites or their exclusion from the input data. In the case when the CNAs were present in a large amount, the normalized RF distance increased stronger and the methods visibly benefited from including CNA sites, as they suffered from insufficient information when the CNA sites were excluded (Figure 5.6b).

In terms of inferring the single mutant genotype, the recall and precision of SIEVE and SCIPhI were not affected much by the presence of CNA sites, regardless of their amount and inclusion or exclusion from the data. In contrast, these measures decreased for Monovar, deteriorating most strongly when CNAs were present in large amounts and included in the data (Figure 5.7a,b). The existence of CNA sites had little influence on the false positive rate of SCIPhI, and only slightly increased the false positive rates of Monovar and SIEVE (Figure 5.7c). The F1 score of SIEVE and SCIPhI were invariant to the CNA sites, while that of Monovar dropped proportionally to the amount of CNAs in the case when they were included in the data (Figure 5.7d). For inferring double mutant genotypes, adding CNAs had very little impact on the performance of both SIEVE and Monovar (Figure 5.8).

Overall, although assuming a diploid genome, SIEVE is robust to the existence of CNA sites in the input data for both inferring cell phylogeny and calling variants. For phylogeny inference using SIEVE it is rather desirable to potentially increase statistical power and include all sites in the data, even if they were affected by CNAs.

2.3.6. SIEVE achieves favourable run times and low memory usage in the default, multi-thread mode.

We further evaluated the run times and memory requirements of SIEVE and other approaches (Figure 5.9 and Table 5.3). While SIEVE in single thread mode was not competitive, it achieved stellar run time performance in the default, multi-thread mode. In particular, SIEVE outperformed other Bayesian methods and was similar in run time performance as compared to CellPhy, a model based on maximum likelihood inference and using bootstrap to estimate node support. With the increase of the number of both cells and sites, the run time of SIEVE in the multi-thread mode increased much slower compared to other methods. This indicates that SIEVE is scalable to large number of cells and sites. In terms of memory usage, all methods performed similarly well, except for SiFit, which required tremendous amounts of memory.

2.3.7. SIEVE inferred a phylogenetic tree and called variants for CRC cells.

We applied SIEVE to a new single-cell whole genome sequencing (scWGS) dataset, where 28 tumor cells were isolated from three primary tumor biopsies of a patient with CRC (CRC28; see Section Data description). We identified 8,470 candidate SNV sites and 1,163,335,103 background sites. To take into account branch-wise substitution rate variation, we employed a relaxed molecular clock model [156] (same for the following datasets). In the inferred maximum clade credibility (MCC) tree (Figure 2.3; see Figure 5.10 for the branch lengths), tumor cells grouped into three highly supported clades corresponding to the three biopsies. The average length of the branches was 4.2×10^{-7}. The estimated effective sequencing error and ADO rates were 7.6×10^{-4} and 0.20, respectively.

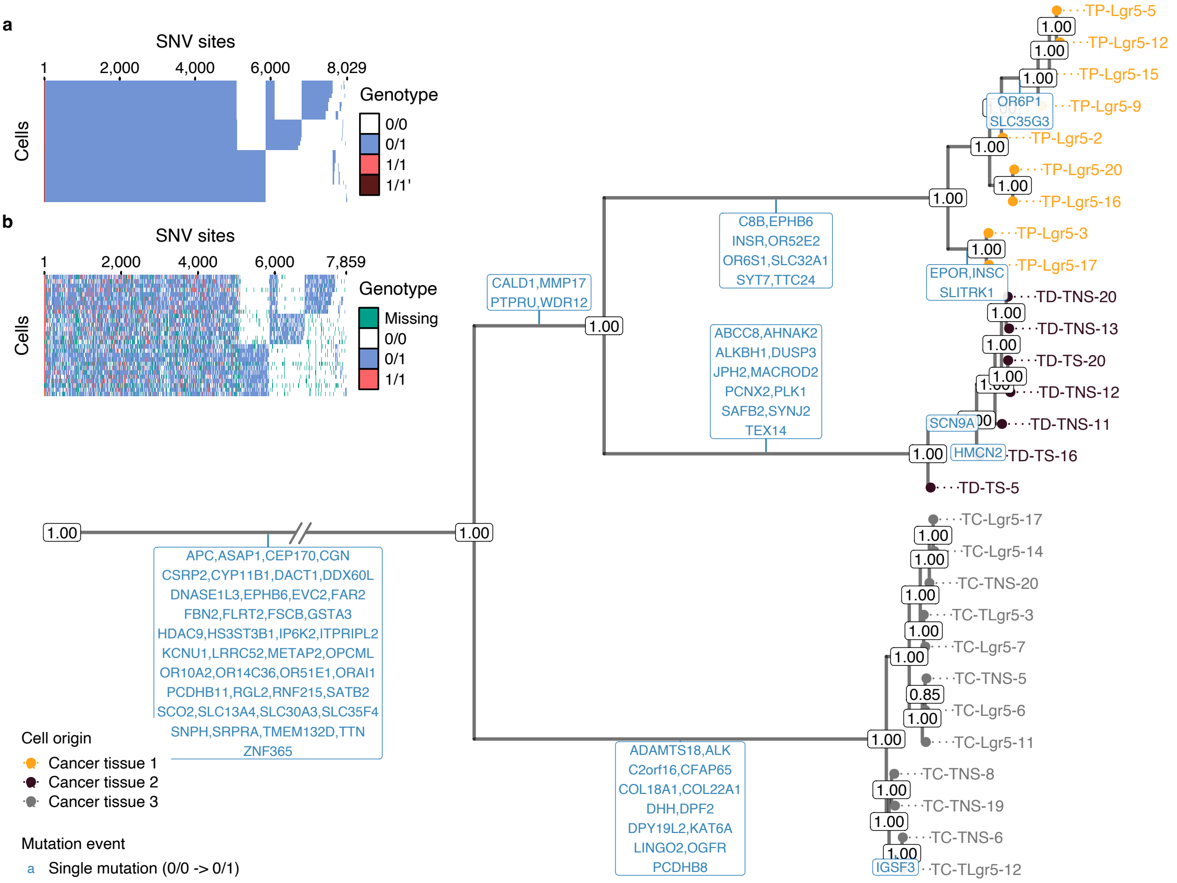


Figure 2.3: **Results of phylogenetic inference and variant calling for the CRC28 [1] dataset.** Shown is SIEVE's maximum clade credibility tree. The exceptionally long trunk has been folded (marked by slashes). Cells are colored according to the corresponding biopsies. The numbers at each node represent posterior probabilities (threshold $p > 0.5$). At each branch, depicted in blue are non-synonymous genes with CRC-related mutations. **a-b**, Variant calling heatmap for SIEVE (**a**) and Monovar (**b**). Listed in the legend are the categories of predicted genotypes by each method. Cells in the row are in the same order as that of leaves in the phylogenetic tree.

Among the trees obtained by other methods (Figure 5.11), the tree obtained by CellPhy was the most similar to the one by SIEVE and also the closest in terms of normalized RF and BS distance (Figure 5.12). Although all methods grouped tumor proximal (TP) cells identically as an independent subclone, SCIPhI and SiFit clustered tumor distal (TD) and tumor central (TC) cells distinctly. Both SIEVE and CellPhy agreed that TP and TD cells were closer than TC cells during the evolutionary history. The fact that the different biopsies form well-supported clades exposes a strong geographical clonal structure suggesting regular growth and limited cell migration. From the four compared models, only SIEVE and CellPhy reported node support values, giving clear intuitions about the confidence for each clade.

We mapped non-synonymous mutations to the internal branches, where only single mutations were found, indicating that the mutational process likely followed the ISA. Many mutations resided on the trunk (clonal mutations), including established CRC driver genes [160, 161], such as *APC*, as well as genes related to the metastatic progression of CRC [162, 163], such as *ASAP1* and *RGL2*. For all mapped genes, SIEVE identified only one type of mutation event, i.e., single mutations that correspond to the switch of the genotype from 0/0 to 0/1. The lack of other mutation events that are possible to identify using our model (see Table 2.1) indicates that for this sample the model did not detect any violations of the ISA.

SIEVE identified 8,029 SNV sites among the candidate SNV sites (Figure 2.3a), where most of the genotypes were single mutant and few were double mutant, including $1/1'$. The variant calling results of SIEVE and Monovar (Figure 2.3b) were overall similar. However, the calls from Monovar were clearly more noisy, with many missing entries and more double mutant genotypes, some of which might be false positives according to the simulation results. The proportion of genotypes called by SIEVE and Monovar were summarized in Table 5.4 (same for the following datasets).

2.3.8. SIEVE inferred a phylogenetic tree and called variants for TNBC cells.

We then applied SIEVE to a single-cell whole exome sequencing (scWES) dataset [2], containing 16 tumor cells collected from a patient with TNBC (TNBC16; see Section Data description). We identified 5,912 candidate SNV sites and 152,027,822 background sites. The estimated tree was supported by high posterior probabilities (Figure 2.4) with a relatively long trunk and short terminal branches (Figure 5.13). The average branch length was 4.6×10^{-6}. We estimated that the effective sequencing error rate was 8.2×10^{-4} and the ADO rate was 0.05.

SCIPhI and CellPhy returned trees that were similar in structure to the one obtained by SIEVE (Figure 5.14), where the tree inferred by SCIPhI was the closest to that inferred by SIEVE (Figure 5.15a) in terms of normalized RF distance and the one inferred by CellPhy was the closest in terms of the BS distance (Figure 5.15b). Finally, the tree obtained by SiFit was the least similar to all other methods.

While for the previous CRC28 dataset the events identified by SIEVE consisted solely of single mutations (transitions from 0/0 to 0/1 genotype), which are typically analyzed and often detected by other methods, the TNBC16 dataset is the showcase of SIEVE's ability to detect more diverse types of mutation events. By mapping non-synonymous mutations to the internal branches, we identified five different types of mutation events, including several violations of the ISA, such as back mutations and parallel mutations. These, apart from the standard single mutations, included 44 coincident homozygous double mutations (transitions from 0/0 to 1/1 genotype), nine homozygous single mutation additions (from 0/1 to 1/1 genotype), two parallel single mutations (from 0/0 to 0/1 genotype that occurred more than once in the tree), and seven single back mutations (from 0/1 to 0/0 genotype). Demeulemeester *et al.* [133] suggested that single back mutation events might occur due to retained mutability of the variant allele, thus making it likely to be mutated again. An alternative explanation for single back mutations could be an occurrence of a loss of heterozygosity. Other events violating the ISA might be due to mutational hotspots and hypermutable motifs [133]. As expected, most of the mutations, including single and double mutant genotypes, resided on the trunk, and some of them occurred in genes which were also reported in the original study [2], such as *TBX3*, *NOTCH2*, *NOTCH3* and *SETBP1*. In the original study, the evolutionary tree of SNV was reconstructed using hierarchical clustering. Unfortunately, clustering is not a phylogenetic method based on shared ancestry, and assumes ultrametricity (perfect clock). In contrast to

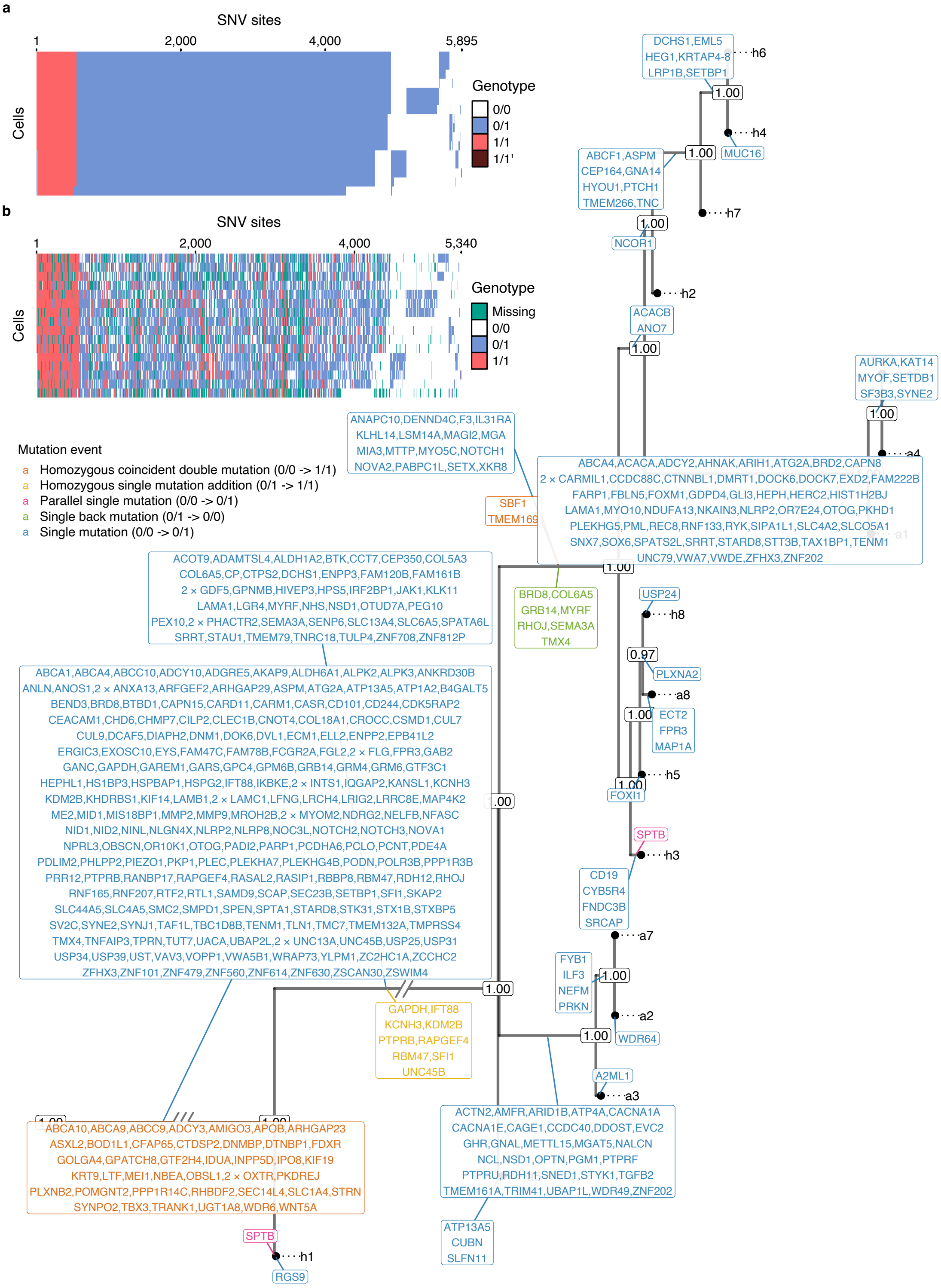


Figure 2.4: **Results of phylogenetic inference and variant calling for TNBC16 [2] dataset.** Shown is SIEVE's maximum clade credibility tree. Two exceptionally long branches are folded with the number of slashes proportional to the branch lengths. Tumor cell names are annotated to the leaves of the tree. The numbers at each node represent the posterior probabilities (threshold $p > 0.5$). At each branch, depicted in different colors are non-synonymous genes that are either TNBC-related single mutations (in blue) or other mutation events (in other colors). **a-b**, variant calling heatmap for SIEVE (**a**) and Monovar (**b**). Listed in the legend are the categories of predicted genotypes by each method. Cells in the row are in the same order as that of leaves in the phylogenetic tree.

hierarchical clustering, our approach gives more insights into the evolutionary history of the tumor. In particular, it infers the error rates, categorizes the types of the mutation events that occurred, and gives posterior estimates for the nodes (the node supports). The high support values (Figure 2.4) indicate that the tree inferred by SIEVE is highly plausible.

SIEVE identified 5,895 SNV sites (Figure 2.4a). In contrast to Monovar, SIEVE calls genotypes for all analyzed sites, including sites with missing data (Figure 2.4b).

2.3.9. SIEVE inferred a phylogenetic tree and called variants for CRC samples mixed with normal cells.

Finally, we applied SIEVE to another scWES dataset [3], which consisted of 35 tumor and normal cells as well as 13 adenomatous polyp cells from a patient with CRC (CRC0827 in [3]; referred to as CRC48 below; see Section Data description). The tumor cells came from two distinct anatomical locations (cancer tissue 1 and 2). We identified 707 candidate SNV sites as well as 119,486,190 background sites. From the inferred phylogenetic tree (Figures 2.5 and 5.16), we identified two tumor clades matching their anatomical locations and one clade for adenomatous polyp and normal cells. Nine cells collected from the tumor biopsies were clustered outside the tumor clades, suggesting that these were normal cells within the tumor biopsies, which was also pointed out in the original study. The average branch length of the inferred tree was 2.1×10^{-7}. We estimated that the effective sequencing error rate was 8.3×10^{-4} and the ADO rate was 0.10.

Other methods reported distinct trees (Figures 5.17 to 5.20), which might result from the relatively insufficient number of (candidate) variant sites as input. CellPhy, SCIPhI and SiFit were also able to distinguish the same set of normal cells from tumor cells. However, SiFit was unable to group tumor cells into two clades matching their anatomical locations as well as SIEVE.

From the non-synonymous mutations mapped to the branches, we observed unique subclonal mutations, including an established CRC driver mutation, *SYNE1* [161]. In addition to multiple single mutation events, we located two parallel single mutations (*CHD3* and *PLD2*), which evolved independently in adenomatous polyps and in tumor cells. Moreover, a mutated gene, *MLH3*, known being related to DNA mismatch repair [164], was found on the branch leading to the tumor subclone. This might be one of the reasons why this phylogenetic tree demonstrates a strong imbalance of branch lengths, with much longer branches found in the tumor subtree.

The variant calling results of SIEVE shared a similar but less noisy structure to those of Monovar (Figure 2.5a,b). We identified 678 SNV sites in total.

2.4. Discussion

Here we present a statistical approach for cell phylogeny inference and variant calling from scDNA-seq data. SIEVE leverages raw read counts to directly reconstruct cell phylogenies and then to reliably call single-cell variants. SIEVE tackles a considerably challenging problem, i.e., the propagation of errors in variant calling to the inference of cell phylogeny, by sharing information between these two tasks. Important characteristics of SIEVE include accounting for the FSA and correction for acquisition bias for tree branch lengths, which prevents from overfitting the phylogenetic model, and, finally, modeling the trunk of the evolutionary tree accommodating the events that are common for all cells.

Inferring mutation status accurately from highly noisy scDNA-seq data remains a demanding problem. A pivotal strength of SIEVE is its characteristic of using genotypes as

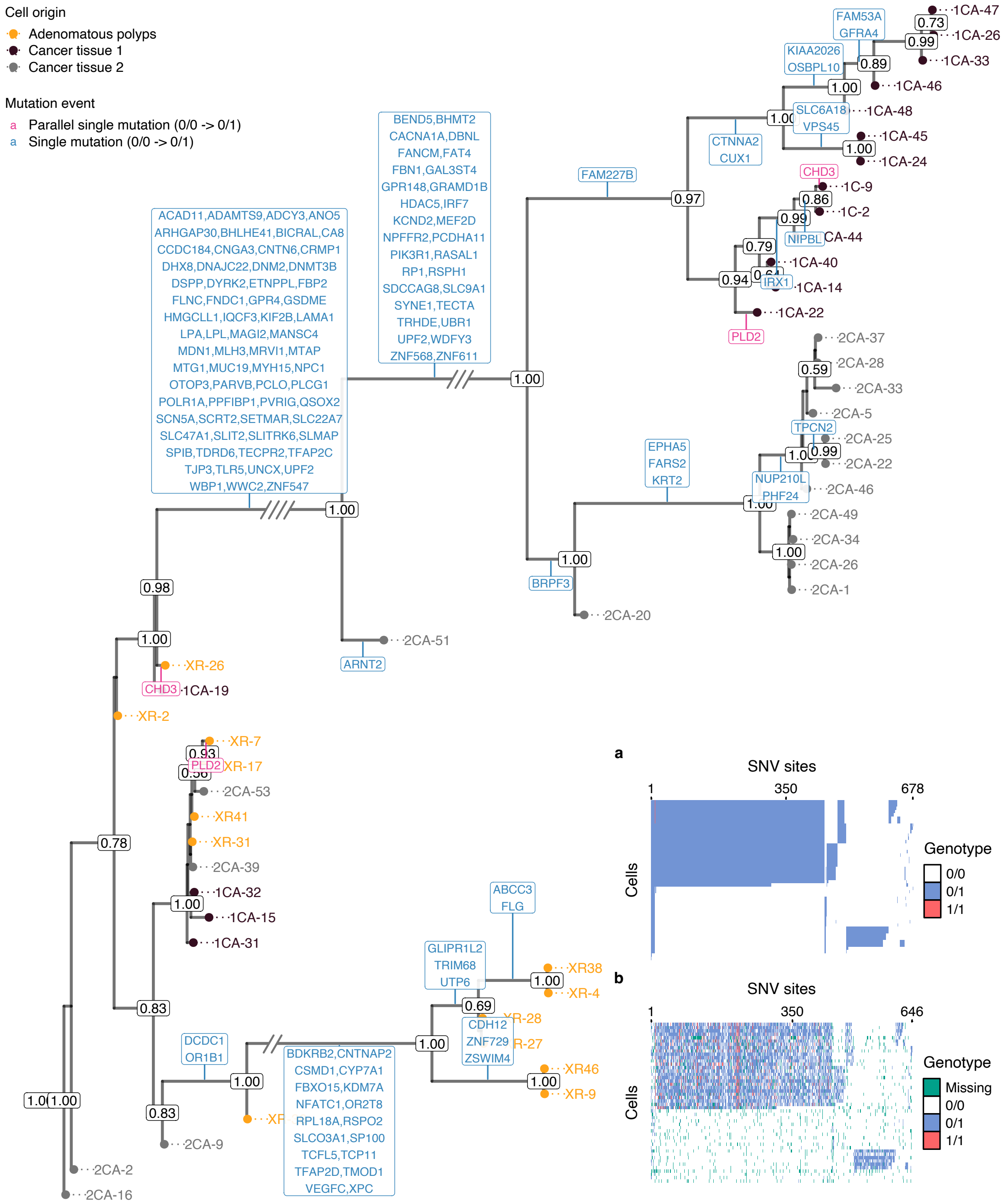


Figure 2.5: **Results of phylogenetic inference and variant calling for CRC48 [3] dataset.** Shown is SIEVE's maximum clade credibility tree. Three exceptionally long branches are folded with the number of slashes proportional to the branch lengths. Cell names are annotated to the leaves of the tree, colored by the corresponding biopsies. The numbers at each node represent the posterior probabilities (threshold $p > 0.5$). At each branch, non-synonymous genes are depicted in different colors including CRC-related single mutations in blue and parallel single mutations in pink. **a-b**, Variant calling heatmap for SIEVE (**a**) and Monovar (**b**). Listed in the legend are the categories of predicted genotypes by each method. Cells in the row are in the same order as that of leaves in the phylogenetic tree.

a bridge between tree inference and variant calling so that these tasks are united. SIEVE is able to reliably differentiate wildtype, single and double mutant genotypes. The benchmarking shows that SIEVE, regarding variant calling, outperforms methods which employ no cell relationships (Monovar) and which, despite accounting for such information, do not include an instantaneous transition rate matrix and branch lengths (SCIPhI). Regarding tree reconstruction, SIEVE is more robust than SCIPhI, which infers phylogenies following ISA from raw scDNA-seq data. It also outperforms methods that rely on variants called by other approaches as a pre-processing step, thereby likely being misled by wrongly inferred variants (CellPhy and SiFit). The high performance of SIEVE can also be attributed to the fact that it is the only model that performs acquisition bias correction, allowing for more accurate branch lengths, and models the distribution of sequencing coverage and accounting for its overdispersion. Finally, SIEVE is also able to reliably call ADOs given data of adequate coverage quality.

Although MCMC is employed in the inference, our results show that SIEVE is an efficient method regarding both run time and memory consumption in the default, multi-thread mode. It also has the potential of favourable scalability to large numbers of cells and sites, where the latter is particularly relevant to the inference of accurate cell phylogenies. Naturally, the more candidate variant sites are available, the more statistical power they confer.

Currently, SIEVE only considers SNVs and assumes a diploid genome. Further improvement could embrace small indels and CNAs to improve phylogenetic inference and variant calling, yet care must be taken to differentiate deletions during evolution from ADOs. Additionally, SIEVE only allows at most one ADO for each site and cell. Further extension could expand to locus dropout, which directly results in missing data.

We apply SIEVE to real scDNA-seq datasets harnessed from CRC and TNBC. SIEVE calls far fewer double mutant genotypes and gives more reliable mutation assignment than Monovar does, in line with the simulation results. We also notice that SIEVE identifies double mutant genotypes, which is rare in CRC but frequent in TNBC, indicating the noteworthy role such genotypes play in the evolution of different types of cancer. Future studies could be based on the phylogenetic tree and variants inferred by SIEVE to identify somatic mutations potentially related to the resistance and relapse in the clinical therapy of cancer. SIEVE can also be applied to targeted sequencing data, where a user-defined number of background sites could be specified for acquisition bias correction. Moreover, SIEVE's applicability is not restricted to cancer samples, and it can also be used to trace lineages of healthy cells.

In the real data analysis we utilize the relaxed molecular clock model implemented in BEAST 2. This shows one of the advantages of SIEVE being a package of BEAST 2, and the potential of exploiting the functionality of other BEAST 2 packages in our model.

Chapter 3

DelSIEVE: joint inference of single-nucleotide variants, somatic deletions, and cell phylogeny from single-cell DNA sequencing data

3.1. Background

Cancer is a genetic disease driven by somatic mutations in the evolutionary process [10, 12, 13, 14, 15], resulting in highly heterogeneous cell populations. One of the somatic mutations is single nucleotide variants (SNVs), which, through nucleotide substitutions, can activate oncogenes and thus promoting tumor proliferation, and can inactivate tumor suppressor genes, resulting in malfunctioned proteins. Another type of somatic mutations is *somatic deletions*, which can inactivate tumor suppressor genes by reducing the number of genomic copies through point deletions, small deletions and copy number aberrations (CNAs) [12, 25, 16, 26, 13, 15]. Phylogenetic inference is typically used to understand and quantify the underlying complexity, or intra-tumor heterogeneity (ITH) [33, 34, 32], which has substantial relevance in the clinical therapy and prognosis of cancer, especially against acquired resistance and relapse of tumor [38, 32, 35].

Previously, methods have been developed for bulk sequencing data to derive variant allele [165, 166, 167, 168, 169] and CNA profiles [170, 171, 172, 173] of clones, as well as to reconstruct tumor phylogeny [139, 140, 82, 141, 142]. Lately, the rapid development of single-cell DNA sequencing (scDNA-seq) technologies exhibit great potential for the analysis of ITH by profiling genetic materials with fine resolution of individual cells [52, 53, 54, 55]. However, despite the strengths, scDNA-seq suffers from a low signal-to-noise ratio, mainly due to the necessity of performing whole genome amplification (WGA) on the limited genetic material present in a single cell [56, 57, 58, 55, 59]. A popular WGA method is multiple displacement amplification (MDA) [64, 65, 66, 67, 68], which can generate a great amount of DNA copies efficiently without introducing many errors. However, MDA is prone to biases against genomic regions, leading to uneven coverage of the genome. Additionally, it may result in allelic dropout (ADO), where one of the two alleles fails to be amplified during the process. In some cases, the amplification of both alleles may fail, leading to locus dropout, which is a potential source of missing data. Such data is suitable for SNV calling, but not for CNA calling, as it is challenging to differentiate true CNA events from amplification biases [56, 55, 59].

Several methods calling SNVs from scDNA-seq have been proposed, which manage to increase statistical power in distinct aspects to account for specific errors. For instance, Monovar [119] pools single cells at each site together, while SCcaller [120], LiRA [122] and SCAN-SNV [121] leverage information on germline single nucleotide polymorphisms. The called SNVs can be used then as input for phylogenetic inference by other methods [124, 125, 126, 83, 127, 128, 129, 100], reconstructing the cell phylogeny with existing cells as leaves and extinct cells as internal nodes in the tree. To share more effectively information among individual cells and to reduce uncertainties introduced by variant callers in phylogenetic inference [86], SCIPhI [130] and SIEVE (previously developed by us) [1] jointly infer SNVs and cell phylogeny. SCIPhI considers a cell phylogeny without branch lengths under the infinite-sites assumption (ISA), which is reportedly often violated in reality [174, 132, 133]. In contrast, SIEVE models a cell phylogeny with branch lengths corrected for acquisition bias [134, 135] under the finite-sites assumption (FSA) within a statistical phylogenetic model, and models the sequencing coverage using a negative binomial distribution. Accounting for more information and providing a more flexible model to share information across cells, SIEVE outperforms SCIPhI in both SNV calling and cell phylogeny reconstruction [1].

One assumption of SIEVE's statistical phylogenetic model is that the genome remains diploid during the evolutionary process of the tumor, overlooking the possible occurrence of somatic deletions. Indeed, the inclusion and the accurate identification of somatic deletions for scDNA-seq remains a challenging problem. This difficulty arises because the sequencing data generated by somatic deletions bears a resemblance to and can be mistaken for ADOs or somatic back mutations. Nevertheless, to address this issue, innovative methods have explored the incorporation of a cell phylogeny, leveraging the idea that cells residing closely on the evolutionary tree share related information, while ADOs occur independently during the sequencing process. SCARLET [175] takes the first step in this direction by refining a copy number tree using read counts for SNVs with a loss-supported phylogeny model. SCIPhIN [131] considers somatic deletions, and allows for mutational losses and recurrent mutations on the cell phylogeny. However, both of them relax the ISA to only a limited extent, which might result in them missing other important events in the evolutionary process, such as double mutations (mutations affecting both alleles at a variant site). In addition, both SCARLET and SCIPhIN ignore the information conveyed by sequencing coverage. However, scDNA-seq data, particularly when coupled with MDA amplification method, is highly uneven across the genome. Therefore, deliberately disregarding the intricacies of sequencing coverage may result in substantial loss of the information embedded within the dataset.

We reasoned that utilizing the additional signal in coverage, combined with the information encoded in the raw read counts and phylogenetic similarities among cells, a model extending SIEVE could account for somatic deletions. Building upon this intuition, here we introduce DelSIEVE (somatic Deletions enabled SIngle-cell EVolution Explorer), a statistical phylogenetic model that includes all features of SIEVE, namely correcting branch lengths of the cell phylogeny for the acquisition bias, incorporating a trunk to model the establishment of the tumor clone, employing a Dirichlet-multinomial distribution to model the raw read counts for all nucleotides, as well as modeling the sequencing coverage using a negative binomial distribution, and extends them with the more versatile capacity of calling somatic deletions. DelSIEVE is capable of modeling locus dropout, where both alleles at a site are allowed to be dropped out during WGA. Importantly, it is the first model leveraging phylogenetic similarities among cells to tell apart the factual deletion genotypes from back mutations or technical artifacts such as ADO or locus dropout. By doing so, DelSIEVE is able to discern 28 types of genotype transitions, associated with 17 types of mutation events, much more than the 12 types of transitions that SIEVE can discern. DelSIEVE is available as a package of BEAST 2 [101] at https://github.com/szczurek-lab/DelSIEVE.

3.2. Methods

In the evolution of tumor, both SNVs and somatic deletions play important roles, leading to highly heterogeneous tumor populations. Assuming a diploid genome in a normal cell as the origin of tumor evolution, our DelSIEVE model performs joint inference of cell phylogeny from scDNA-seq and the resulting SNVs and somatic deletions in single cells.

3.2.1. DelSIEVE model

DelSIEVE takes as input raw read counts for all four nucleotides for cell $j \in \{1, \dots, J\}$ at candidate site $i \in \{1, \dots, I\}$ in the form of $\mathcal{D}_{ij}^{(1)} = (\boldsymbol{m}_{ij}, c_{ij})$, where $\boldsymbol{m}_{ij} = \{m_{ijk} \,|\, k = 1, 2, 3\}$ is the read counts of three alternative nucleotides with values in descending order and c_{ij} is the sequencing coverage (Figure 3.1a; see Section Candidate site identification for explanation of how candidate sites are identified). DelSIEVE also optionally takes raw read counts data $\mathcal{D}^{(2)}$ from I' background sites for acquisition bias correction. It is important to note that since DelSIEVE requires preselected candidate variant sites as input, it can only identify somatic deletions at those candidate sites.

The model first infers the cell phylogeny, followed by maximum likelihood estimation of the genotype state of each node in the tree (Figure 3.1a). The power of DelSIEVE lies in the elegantly devised probabilistic graphical model, where the hidden variable describing the genotype for site i in cell j, denoted g_{ij}, is used as the bridge between the statistical phylogenetic model and the model of raw read counts (Figure 3.1b).

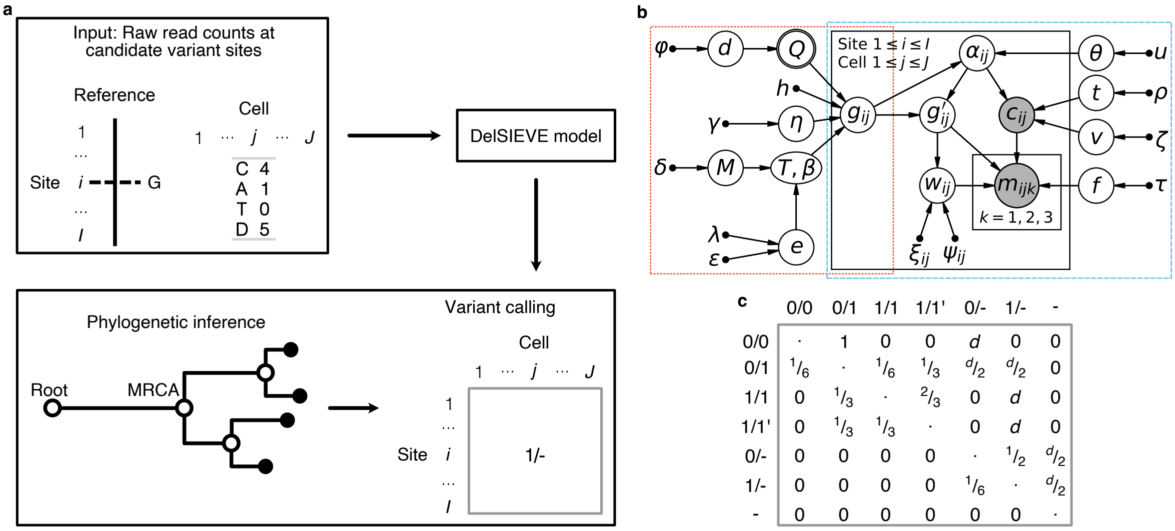
Statistical phylogenetic model

DelSIEVE expands the genotype state space defined in SIEVE: on top of $0/0$ (*wildtype*), $0/1$ (*single mutant*), $1/1$ (*double mutant*, where the two alternative nucleotides are the same) and $1/1'$ (*double mutant*, where the two alternative nucleotides are different), DelSIEVE additionally considers $0/$- (*reference-left single deletion*), $1/$- (*alternative-left single deletion*) and - (*double deletion*). Here, $0, 1, 1'$ and - represent the reference nucleotide, an alternative nucleotide, a second alternative nucleotide different from that denoted by 1, and deletions, respectively. The expanded genotype state space $G = \{0/0, 0/1, 1/1, 1/1', 0/\text{-}, 1/\text{-}, \text{-}\}$ enables the addition of somatic deletions as possible events in the statistical phylogenetic model (Figure 3.1c). Given the genotype state space G, DelSIEVE is able to discern 28 types of genotype transitions (16 more than SIEVE), which can be categorized into 17 types of mutation events (8 more than SIEVE; see Section Mutation event classification).

With the genotype state space G specified, we define the instantaneous transition rate matrix Q in Figure 3.1c, which is the key component to the statistical phylogenetic model. We set the somatic mutation rate to 1, where the relative measurements for back mutation rate and deletion rate are ${}^{1}\!/_{3}$ and d, respectively. Thus, Q is deterministic and depends on the hidden random variable corresponding to the relative deletion rate d:

$$P\left(Q \,|\, d\right) = 1. \tag{3.1}$$

Each entry in Q represents the transition rate from the genotype in the row to that in the column during an infinitesimal time Δt. Besides, each row in Q sums up to 0. The continuous-time homogeneous Markov chain underlying Q is time non-reversible and reducible. For instance, genotypes that have both alleles present can transition to genotypes with one or both alleles lost, but not vice versa. To be specific, genotypes $\{0/0, 0/1, 1/1, 1/1'\}$ and genotypes $\{0/\text{-}, 1/\text{-}\}$ form two ergodic, transient communicating classes, while genotype $\{\text{-}\}$ forms a



	0/0	0/1	1/1	1/1'	0/-	1/-	-
0/0	·	1	0	0	d	0	0
0/1	$^{1}/_{6}$	·	$^{1}/_{6}$	$^{1}/_{3}$	$^{d}/_{2}$	$^{d}/_{2}$	0
1/1	0	$^{1}/_{3}$	·	$^{2}/_{3}$	0	d	0
1/1'	0	$^{1}/_{3}$	$^{1}/_{3}$	·	0	d	0
0/-	0	0	0	0	·	$^{1}/_{2}$	$^{d}/_{2}$
1/-	0	0	0	0	$^{1}/_{6}$	·	$^{d}/_{2}$
-	0	0	0	0	0	0	·

Figure 3.1: **Overview of the DelSIEVE model.** **a** Analysis workflow of DelSIEVE with an example of input data. At candidate variate site $i \in \{1, \ldots, I\}$, the reference nucleotide is G. For cell $j \in \{1, \ldots, J\}$ at site i, observed are sequencing depth being 5 (marked by D) as well as read counts for nucleotide C being 4 and A being 1. Inferred first is the cell phylogeny from the input data by DelSIEVE. Based on the cell phylogeny, determined is the genotype state of each node in the tree through maximum likelihood estimation. For instance, $1/-$ is inferred as the genotype state of cell j at site i. **b** Probabilistic graphical model of DelSIEVE. The orange dotted frame shows the part corresponding to the the statistical phylogenetic model, and the blue dashed frame encloses the part corresponding to the model of raw read counts. Shaded circle nodes represent observed variables, while unshaded circle nodes represent hidden random variables. Nodes with double circles are deterministic random variables, meaning that they are readily fixed once the values of their parents are determined. Small filled circles correspond to fixed hyper parameters. Arrows denote local conditional probability distributions of child nodes given parent nodes. **c** Instantaneous transition rate matrix of the statistical phylogenetic model. The hidden random variable d is the deletion rate, measured relatively to the mutation rate. The elements in the diagonal of the matrix are denoted by dots, and have negative values equal to the sum of the other entries in the same row, ensuring that the sum of each row equals zero.

closed communicating class. As a result, the limiting distribution of the Markov chain exists, where the value corresponding to genotype - is 1, while the others are 0.

Based on the well-established theory of statistical phylogenetic models (see Section Statistical phylogenetic models), the joint conditional probability of the genotype states of all sequenced cells at site i, namely $\boldsymbol{g}_i^{(L)}$, is

$$P\left(\boldsymbol{g}_i^{(L)} \,\middle|\, \mathcal{T}, \boldsymbol{\beta}, Q, h, \eta\right) = \sum_{\boldsymbol{g}_i^{(A)} \setminus \left\{g_{i(2J)}\right\}} P\left(\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \left\{g_{i(2J)}\right\} \,\middle|\, \mathcal{T}, \boldsymbol{\beta}, Q, h, \eta\right). \tag{3.2}$$

Intuitively, this means that to compute the likelihood of the genotypes of the variant sites at the leaves, we marginalize out the genotypes at the ancestor nodes from the total likelihood. The variables in Equation (3.2) have the same meaning as in SIEVE (see Section SIEVE model). Briefly speaking, $\mathcal{T}$ is the rooted binary tree topology, whose root, representing a normal cell with diploid genome, has only one child, the MRCA of all sequenced cells. $\mathcal{T}$ has J existing, sequenced cells as leaves, whose genotypes are $\boldsymbol{g}_i^{(L)} = \left(g_{i1}, \ldots, g_{ij}, \ldots, g_{iJ}\right)^T$, where $g_{ij} \in G$. The J extinct, ancestor cells in $\mathcal{T}$ as internal nodes have genotypes $\boldsymbol{g}_i^{(A)} = \left(g_{i(J+1)}, \ldots, g_{ij}, \ldots, g_{i(2J)}\right)^T$, where $g_{ij} \setminus \left\{g_{i(2J)}\right\} \in G$ and $g_{i(2J)} = 0/0$. $\mathcal{T}$ also has $2J - 1$

branches, whose lengths $\boldsymbol{\beta} \in \mathbb{R}^{2J-1}$ represent the expected number of somatic mutations per site. h and η are the number of rate categories and shape, respectively, of a discrete Gamma distribution with mean equal 1 for modeling among-site substitution rate variation. Hidden random variables d in Equation (3.1) and $\mathcal{T}, \boldsymbol{\beta}, \eta$ in Equation (3.2) are estimated using MCMC, while the fixed hyperparameter h takes value 4 by default.

Given deletion rate d (and thus Q) and branch length β, the seven-by-seven transition probability matrix $R(\beta)$ is computed as $R(\beta) = \exp(Q\beta)$ [86].

Model of raw read counts

We factorize the probability of observing $\mathcal{D}_{ij}$ for cell j at site i into

$$P(\mathcal{D}_{ij}) = P(\boldsymbol{m}_{ij} \,|\, c_{ij}) P(c_{ij}), \tag{3.3}$$

where the former corresponds to the model of nucleotide read counts and the latter to the model of sequencing coverage.

Model of sequencing coverage. One of the major, yet often overlooked challenges in scDNA-seq is the highly uneven sequencing coverage. This happens because the genetic materials are amplified largely unequally during WGA. Similar to SIEVE, we employ a negative binomial distribution to capture the overdispersion existing in the sequencing coverage:

$$P(c \,|\, p, r) = \binom{c + r - 1}{r - 1} p^r (1 - p)^c, \tag{3.4}$$

where p and r are parameters. To improve interpretability, the distribution is reparameterized using mean μ and variance σ^2:

$$\begin{cases} p = \dfrac{\mu}{\sigma^2}, \\ r = \dfrac{\mu^2}{\sigma^2 - \mu}. \end{cases} \tag{3.5}$$

We assume that μ_{ij} and σ^2_{ij} have the same form as in SIEVE, namely

$$\begin{aligned} \mu_{ij} &= \alpha_{ij} t s_j, \\ \sigma^2_{ij} &= \mu_{ij} + \alpha^2_{ij} \nu s^2_j. \end{aligned} \tag{3.6}$$

Here, t and ν are the mean and the variance of allelic coverage, respectively. $\alpha_{ij} \in \{0, 1, 2\}$ represents the number of sequenced alleles. With the extended genotype state space G in the DelSIEVE model, the number of alleles possessed by a cell at a site can either be zero (corresponding to genotype state {-}), one (genotype states {0/-, 1/-}), or two ($\{0/0, 0/1, 1/1, 1/1'\}$). On top of that, the possible occurrence of ADOs during scWGA could alter the number of alleles possessed by a cell at a site. Here, we model two types of ADOs, single ADO and locus dropout.

The single ADO mode was previously proposed by us in SIEVE (see Section SIEVE model), where at most one ADO is allowed to happen to cell j at site i. For DelSIEVE, the corresponding prior distribution of α_{ij}, $P(\alpha_{ij} \,|\, g_{ij}, \theta)$, is defined in Table 3.1, where θ denotes the probability of the occurrence of single ADO when both alleles exist. One should consider the "Single ADO occurred" column as value of an additional hidden random variable corresponding to an ADO occurrence indicator, which will be marginalized out in the model. For example, the probability of an event of single ADO occurance when $g_{ij} = 0/$- equals $\theta/2$,

Table 3.1: **Definition of the distribution of α_{ij} conditional on g_{ij} and θ under single ADO mode for DelSIEVE.**

α_{ij}	g_{ij}	Single ADO occurred	$P(\alpha_{ij}\,|\,g_{ij},\theta)$
1	0/0	Yes	θ
2	0/0	No	$1-\theta$
1	0/1	Yes	θ
2	0/1	No	$1-\theta$
1	1/1	Yes	θ
2	1/1	No	$1-\theta$
1	1/1′	Yes	θ
2	1/1′	No	$1-\theta$
0	0/-	Yes	${}^{\theta}\!/_{2}$
1	0/-	No	$1-{}^{\theta}\!/_{2}$
0	1/-	Yes	${}^{\theta}\!/_{2}$
1	1/-	No	$1-{}^{\theta}\!/_{2}$
0	-	No	1
Others			0

because there is only one allele left to be dropped out. For genotype -, it is certain that single ADO has not occuredd as there is no allele existing.

To generalize DelSIEVE to model both ADO and locus dropout, we allow more than one allele to drop out. $P(\alpha_{ij}\,|\,g_{ij},\theta)$ is defined in Table 3.2, where θ represents the probability of an allele dropped out. We assume that the ADOs occur to each allele independently. For instance, when $g_{ij}=0/0$, the probability of $\alpha_{ij}=0$ is θ^2, happening only when both alleles drop out. For genotype 0/-, the sole allele drops out with probability θ, resulting in zero sequenced alleles.

s_j in Equation (3.6) is the size factor of cell j, which is estimated exactly in the same way as in SIEVE:

$$\hat{s}_j = \underset{i:c_{ij}\neq 0}{\operatorname{median}} \frac{c_{ij}}{\left(\prod_{\substack{j'=1\\c_{ij'}\neq 0}}^{J'} c_{ij'}\right)^{\frac{1}{J'}}}, \tag{3.7}$$

where J' is the number of cells with non-zero coverage at a site.

Model of nucleotide read counts. We showed before that the occurrence of ADOs could change the number of alleles possessed by cell j at site i. As a result, the genotype g_{ij} could change to the *ADO-affected genotype*, $g'_{ij}\in G$. The probability of g'_{ij} writes $P(g'_{ij}\,|\,g_{ij},\alpha_{ij})$, which is defined in Table 3.3 for the single ADO mode and in Table 3.4 for the locus dropout mode.

When $g'_{ij}\in G\setminus\{\text{-}\}$, we model $\boldsymbol{m}_{ij}$, the read counts of three alternative nucleotides, conditional on the sequencing coverage c_{ij} with a Dirichlet-multinomial distribution as

$$P(\boldsymbol{m}_{ij}\,|\,c_{ij},\boldsymbol{a}_{ij}) = \frac{F(c_{ij},a_{ij0})}{\prod_{k=1:m_{ijk}>0}^{3} F(m_{ijk},a_{ijk})F(c_{ij}-\sum_{k=1}^{3} m_{ijk},a_{ij4})}, \tag{3.8}$$

with parameters $\boldsymbol{a}_{ij}=\{a_{ijk}\,|\,k=1,\dots,4\}$ and $a_{ij0}=\sum_{k=1}^{4} a_{ijk}$. F is a function defined as

$$F(x,y) = \begin{cases} xB(y,x), & \text{if } x>0,\\ 1, & \text{otherwise,}\end{cases} \tag{3.9}$$

Table 3.2: **Definition of the distribution of α_{ij} conditional on g_{ij} and θ under locus dropout mode for DelSIEVE.**

α_{ij}	g_{ij}	Number of alleles dropped out	$P(\alpha_{ij} \mid g_{ij}, \theta)$
0	0/0	2	θ^2
1	0/0	1	$2\theta(1-\theta)$
2	0/0	0	$(1-\theta)^2$
0	0/1	2	θ^2
1	0/1	1	$2\theta(1-\theta)$
2	0/1	0	$(1-\theta)^2$
0	1/1	2	θ^2
1	1/1	1	$2\theta(1-\theta)$
2	1/1	0	$(1-\theta)^2$
0	1/1′	2	θ^2
1	1/1′	1	$2\theta(1-\theta)$
2	1/1′	0	$(1-\theta)^2$
0	0/-	1	θ
1	0/-	0	$1-\theta$
0	1/-	1	θ
1	1/-	0	$1-\theta$
0	-	0	1
		Others	0

where B is the beta function. Note that $c_{ij} - \sum_{k=1}^{3} m_{ijk}$ is the read count of the reference nucleotide.

Similar to SIEVE, we reparameterize Equation (3.8) by letting $\boldsymbol{a}_{ij} = w_{ij}\boldsymbol{f}_{ij}$. w_{ij} is related to the overdispersion. $\boldsymbol{f}_{ij} = \{f_{ijk} \mid k = 1, \ldots, 4\}$, $\sum_{k=1}^{4} f_{ijk} = 1$ is a vector of expected frequencies of each nucleotide, where the first three elements correspond to the three alternative nucleotides ordered decreasingly according to their read counts, and the last to the reference nucleotide. Depending on g'_{ij}, $\boldsymbol{f}_{ij}$ is given by

$$\boldsymbol{f}_{ij} = \begin{cases} \boldsymbol{f}_1 = \left(\frac{1}{3}f, \frac{1}{3}f, \frac{1}{3}f, 1-f\right), \text{ if } g'_{ij} = 0/0 \text{ or } 0/\text{-}, \\ \boldsymbol{f}_2 = \left(\frac{1}{2} - \frac{1}{3}f, \frac{1}{3}f, \frac{1}{3}f, \frac{1}{2} - \frac{1}{3}f\right), \text{ if } g'_{ij} = 0/1, \\ \boldsymbol{f}_3 = \left(1-f, \frac{1}{3}f, \frac{1}{3}f, \frac{1}{3}f\right), \text{ if } g'_{ij} = 1/1 \text{ or } 1/\text{-}, \\ \boldsymbol{f}_4 = \left(\frac{1}{2} - \frac{1}{3}f, \frac{1}{2} - \frac{1}{3}f, \frac{1}{3}f, \frac{1}{3}f\right), \text{ if } g'_{ij} = 1/1', \end{cases} \tag{3.10}$$

where f is the effective sequencing error rate, combining together amplification and sequencing errors.

The parameter w_{ij} also depends on g'_{ij}, where

$$w_{ij} = \begin{cases} w_1, \text{ if } g'_{ij} = 0/0, 0/\text{-}, 1/1, \text{ or } 1/\text{-}, \\ w_2, \text{ if } g'_{ij} = 0/1 \text{ or } 1/1', \end{cases} \tag{3.11}$$

and w_1 corresponds to wild type overdispersion and w_2 to alternative overdispersion.

Table 3.3: **Definition of the distribution of g'_{ij} conditional on g_{ij} and α_{ij} under single ADO mode for DelSIEVE.**

g'_{ij}	g_{ij}	α_{ij}	$P(g'_{ij} \mid g_{ij}, \alpha_{ij})$
0/0	0/0	2	1
0/-	0/0	1	1
0/1	0/1	2	1
0/-	0/1	1	$^1/_2$
1/-	0/1	1	$^1/_2$
1/1	1/1	2	1
1/-	1/1	1	1
1/1′	1/1′	2	1
1/-	1/1′	1	1
0/-	0/-	1	1
-	0/-	0	1
1/-	1/-	1	1
-	1/-	0	1
-	-	0	1
Others			0

By plugging Equations (3.10) and (3.11) into Equation (3.8), we have

$$P(\boldsymbol{m}_{ij}|c_{ij}, g'_{ij}, f, w_{ij}) = \begin{cases} P_{0/0} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 0/0, \boldsymbol{f}_1, w_1\right), \\ P_{0/\text{-}} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 0/\text{-}, \boldsymbol{f}_1, w_1\right), \\ P_{0/1} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 0/1, \boldsymbol{f}_2, w_2\right), \\ P_{1/1} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 1/1, \boldsymbol{f}_3, w_1\right), \\ P_{1/\text{-}} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 1/\text{-}, \boldsymbol{f}_3, w_1\right), \\ P_{1/1'} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 1/1', \boldsymbol{f}_4, w_2\right), \\ P_{\text{-}} = P(\boldsymbol{m}_{ij}|c_{ij}, g'_{ij} = \text{-}, f, w_{ij}) = 1, \end{cases} \tag{3.12}$$

where we additionally define $P(\boldsymbol{m}_{ij}|c_{ij}, g'_{ij} = \text{-}, f, w_{ij}) = 1$.

Although g_{ij} and g'_{ij} share the same genotype state space, it's important to note that some genotype states can arise from distinct evolutionary or technical events. For instance, genotype 1/- could be the outcome of evolutionary processes, where one allele was deleted while the other remained intact. Alternatively, it could also be a result of technical artifacts, where both alleles were initially present before scWGA, but one allele experienced dropout during the amplification process. The presence of multiple potential causes for genotypes, such as the genotype 1/-, introduces a significant challenge in disentangling their origins compared to methods like SIEVE, which predominantly attribute such genotypes to technical artifacts. However, an encouraging development is the integration of the statistical phylogenetic model and the model of sequencing coverage. This integration allows for a comprehensive analysis from both evolutionary and technical perspectives, thereby facilitating the disentanglement. By incorporating the statistical phylogenetic model, we gain insights into the evolutionary dynamics underlying genotype development, while the model of sequencing coverage provides valuable information about the technical nuances of the sequencing technique employed. This combined approach offers a more robust framework for disentangling the complex factors contributing to genotypic variations and enhancing our understanding of the underlying biological and technical processes involved.

Table 3.4: **Definition of the distribution of g'_{ij} conditional on g_{ij} and α_{ij} under locus dropout mode for DelSIEVE.**

g'_{ij}	g_{ij}	α_{ij}	$P(g'_{ij} \mid g_{ij}, \alpha_{ij})$
0/0	0/0	2	1
0/-	0/0	1	1
-	0/0	0	1
0/1	0/1	2	1
0/-	0/1	1	$^1/_2$
1/-	0/1	1	$^1/_2$
-	0/1	0	1
1/1	1/1	2	1
1/-	1/1	1	1
-	1/1	0	1
1/1'	1/1'	2	1
1/-	1/1'	1	1
-	1/1'	0	1
0/-	0/-	1	1
-	0/-	0	1
1/-	1/-	1	1
-	1/-	0	1
-	-	0	1
Others			0

DelSIEVE likelihood

Combining the statistical phylogenetic model and the model of raw read counts described above, we acquire the likelihood of DelSIEVE, denoted by

$$P\left(\mathcal{D}^{(1)}, \mathcal{D}^{(2)} \,\middle|\, \mathcal{T}, \boldsymbol{\beta}, Q, h, \eta, t, v, \theta, f, w_1, w_2\right). \tag{3.13}$$

To simplify notation, we denote some variables in the statistical phylogenetic model as $\Theta = \{\mathcal{T}, \boldsymbol{\beta}, Q, h, \eta\}$ and some in the model of raw read counts as $\Phi = \{t, v, \theta, f, w_1, w_2\}$. By taking the logarithm, Equation (3.13) is further writes

$$\log \mathcal{L}(\Theta, \Phi) = \log \mathcal{L}^{(1)}(\Theta, \Phi) + \log \mathcal{L}^{(2)}(\Theta, \Phi), \tag{3.14}$$

where $\mathcal{L}^{(1)}$ is the tree likelihood corrected for acquisition bias computed for candidate SNV sites in $\mathcal{D}^{(1)}$, while $\mathcal{L}^{(2)}$ is the likelihood computed for background sites in $\mathcal{D}^{(2)}$, referred to as the background likelihood.

Acquisition bias refers to the cases where the branch lengths of cell phylogenies are overestimated when only using data from SNV sites as input [134, 135]. Here, it is corrected similarly to SIEVE, following [146]:

$$\log \mathcal{L}^{(1)} = \log P\left(\mathcal{D}^{(1)} \,\middle|\, \Theta, \Phi\right) + I' \log \left(\frac{1}{I} \sum_{i=1}^{I} C_i\right), \tag{3.15}$$

where the first component is the uncorrected tree log-likelihood for SNV sites, and C_i in the second component is the likelihood of SNV site i being invariant (see below).

To compute $\log P\left(\mathcal{D}^{(1)}\,\middle|\,\Theta, \Phi\right)$ in Equation (3.15), we decompose it according to the probabilistic graphical model in Figure 3.1b. Assuming independent and identical evolution of each candidate variant site, $\log P\left(\mathcal{D}^{(1)}\,\middle|\,\Theta, \Phi\right)$ writes

$$\begin{aligned}
\log P\left(\mathcal{D}^{(1)}\,\middle|\,\Theta, \Phi\right) &= \sum_{i=1}^{I} \log \sum_{\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}} \Bigg[P\left(\mathcal{D}_i^{(1)}\,\middle|\,\boldsymbol{g}_i^{(L)}, \Phi\right) \\
&\qquad \times P\left(\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}\,\middle|\,\Theta\right) \Bigg] \\
&= \sum_{i=1}^{I} \log \sum_{\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}} \Bigg[\prod_{j=1}^{J} P\left(\boldsymbol{m}_{ij}, c_{ij}\,|\,g_{ij}, \Phi\right) \\
&\qquad \times P\left(\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}\,\middle|\,\Theta\right) \Bigg] \\
&= \sum_{i=1}^{I} \sum_{j=1}^{J} \log \sum_{\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}} \Bigg[P\left(\boldsymbol{m}_{ij}, c_{ij}\,|\,g_{ij}, \Phi\right) \\
&\qquad \times P\left(\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}\,\middle|\,\Theta\right) \Bigg],
\end{aligned} \tag{3.16}$$

where $P(\boldsymbol{m}_{ij}, c_{ij}\,|\,g_{ij}, \Phi)$, representing the model of raw read counts applied on the leaves of the phylogenetic tree, is similarly decomposed into

$$\begin{aligned}
P\left(\boldsymbol{m}_{ij}, c_{ij}\,|\,g_{ij}, \Phi\right) &= P\left(\boldsymbol{m}_{ij}, c_{ij}\,|\,g_{ij}, f, w_{ij}, t, v, \theta\right) \\
&= \sum_{\alpha_{ij}, g'_{ij}} P\left(\boldsymbol{m}_{ij}, c_{ij}, \alpha_{ij}, g'_{ij}\,\middle|\,g_{ij}, f, w_{ij}, t, v, \theta\right) \\
&= \sum_{\alpha_{ij}, g'_{ij}} \Big[P\left(\boldsymbol{m}_{ij}\,\middle|\,c_{ij}, g'_{ij}, f, w_{ij}\right) P\left(g'_{ij}\,\middle|\,g_{ij}, \alpha_{ij}\right) \\
&\qquad \times P\left(c_{ij}\,|\,\alpha_{ij}, t, v\right) P\left(\alpha_{ij}\,|\,g_{ij}, \theta\right) \Big].
\end{aligned} \tag{3.17}$$

$P(c_{ij}\,|\,\alpha_{ij}, t, v)$ in the above equation is defined through Equations (3.4) to (3.6), and $P(\boldsymbol{m}_{ij}\,|\,c_{ij}, g'_{ij}, f, w_{ij})$ is defined in Equation (3.12). Under the single ADO mode, $P(\alpha_{ij}\,|\,g_{ij}, \theta)$ and $P(g'_{ij}\,|\,g_{ij}, \alpha_{ij})$ are defined as shown in Table 3.1 and Table 3.3, respectively, while under the locus dropout mode in Table 3.2 and Table 3.4, respectively. As a result, Equation (3.17) takes distinct forms under different modes of modeling ADOs.

For the single ADO mode, Equation (3.17) is further represented as

$$P\left(\boldsymbol{m}_{ij}, c_{ij} \,|\, g_{ij}, \Phi\right) = \begin{cases} P_{0/0} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta) \\ \qquad + P_{0/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot \theta, \text{ if } g_{ij} = 0/0, \\ P_{0/1} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta) \\ \qquad + \dfrac{1}{2}(P_{0/\text{-}} + P_{1/\text{-}}) \cdot P(c_{ij}|\alpha_{ij} = 1, t, v) \cdot \theta, \text{ if } g_{ij} = 0/1, \\ P_{1/1} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta) \\ \qquad + P_{1/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot \theta, \text{ if } g_{ij} = 1/1, \\ P_{1/1'} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta) \\ \qquad + P_{1/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot \theta, \text{ if } g_{ij} = 1/1', \\ P_{0/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot (1-\dfrac{\theta}{2}) \\ \qquad + P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v) \cdot \dfrac{\theta}{2}, \text{ if } g_{ij} = 0/\text{-}, \\ P_{1/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot (1-\dfrac{\theta}{2}) \\ \qquad + P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v) \cdot \dfrac{\theta}{2}, \text{ if } g_{ij} = 1/\text{-}, \\ P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v), \text{ if } g_{ij} = \text{-}. \end{cases} \tag{3.18}$$

For the locus dropout mode, Equation (3.17) writes

$$P\left(\boldsymbol{m}_{ij}, c_{ij} \,|\, g_{ij}, \Phi\right) = \begin{cases} P_{0/0} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta)^2 \\ \qquad + P_{0/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot 2 \cdot \theta \cdot (1-\theta) \\ \qquad + P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v) \cdot \theta^2, \text{ if } g_{ij} = 0/0, \\ P_{0/1} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta)^2 \\ \qquad + (P_{0/\text{-}} + P_{1/\text{-}}) \cdot P(c_{ij}|\alpha_{ij} = 1, t, v) \cdot \theta \cdot (1-\theta) \\ \qquad + P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v) \cdot \theta^2, \text{ if } g_{ij} = 0/1, \\ P_{1/1} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta)^2 \\ \qquad + P_{1/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot 2 \cdot \theta \cdot (1-\theta) \\ \qquad + P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v) \cdot \theta^2, \text{ if } g_{ij} = 1/1, \\ P_{1/1'} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta)^2 \\ \qquad + P_{1/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot 2 \cdot \theta \cdot (1-\theta) \\ \qquad + P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v) \cdot \theta^2, \text{ if } g_{ij} = 1/1', \\ P_{0/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot (1-\theta) \\ \qquad + P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v) \cdot \theta, \text{ if } g_{ij} = 0/\text{-}, \\ P_{1/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot (1-\theta) \\ \qquad + P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v) \cdot \theta, \text{ if } g_{ij} = 1/\text{-}, \\ P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v), \text{ if } g_{ij} = \text{-}. \end{cases} \tag{3.19}$$

Equation (3.16) is computed efficiently using the Felsenstein's pruning algorithm [95]. For I candidate SNV sites, J cells and K genotype states in G (for DelSIEVE $K = 7$), the time complexity of the Felsenstein's pruning algorithm is $\mathcal{O}(IJK^2)$.

Since in the second component of Equation (3.15), C_i corresponds to the likelihood of candidate SNV site i being invariant, it is computed as the joint probability of $\mathcal{D}_i$ and $\boldsymbol{g}_i^{(L)} = 0/0$, writing

$$\begin{aligned}
C_i &= P\left(\mathcal{D}_i^{(1)}, \boldsymbol{g}_i^{(L)} = 0/0 \,\middle|\, \Theta, \Phi\right) \\
&= P\left(\mathcal{D}_i^{(1)} \,\middle|\, \boldsymbol{g}_i^{(L)} = 0/0, \Phi\right) \sum_{\boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}} P\left(\boldsymbol{g}_i^{(L)} = 0/0, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\} \,\middle|\, \Theta\right) \\
&= \prod_{j=1}^{J} P\left(\boldsymbol{m}_{ij}, c_{ij} \,|\, g_{ij} = 0/0, \Phi\right) \sum_{\boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}} P\left(\boldsymbol{g}_i^{(L)} = 0/0, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\} \,\middle|\, \Theta\right),
\end{aligned} \tag{3.20}$$

which is computed similarly to Equation (3.16), but with g_{ij} for $j = 1, \dots J$ fixed to $0/0$. In fact, C_i and $\log P\left(\mathcal{D}_i^{(1)} \,\middle|\, \Theta, \Phi\right)$ are computed simultaneously in the implementation for optimized efficiency.

To efficiently compute $\log \mathcal{L}^{(2)}$, the background likelihood in Equation (3.14), we make several simplifications similar to SIEVE. Specifically, we assume that each cell at each background site has the wildtype genotype with both alleles covered during scWGA. We also assume that $P(c_{ij} \,|\, \alpha_{ij}, t, v) = 1$ and $P\left(\boldsymbol{g}_i^{(L)} = 0/0, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\} \,\middle|\, \Theta\right) = 1$, thereby ignoring the model of sequencing coverage and the tree log-likelihood for the background sites i for $i = 1, \dots I'$. With an alternative form of the Dirichlet-multinomial distribution, $\log \mathcal{L}^{(2)}$ is approximately and efficiently computed by

$$\begin{aligned}
\log \mathcal{L}^{(2)}(f, w_1) &= \sum_{i=1}^{I'} \sum_{j=1}^{J} \log P_{0/0} \\
&= \sum_{i=1}^{I'} \sum_{j=1}^{J} \log \left[\frac{\Gamma(w_1)\Gamma(c_{ij}+1)}{\Gamma(c_{ij}+w_1)} \prod_{k=1}^{3} \frac{\Gamma(m_{ijk} + \frac{1}{3} f w_1)}{\Gamma(\frac{1}{3} f w_1)\Gamma(m_{ijk}+1)} \right. \\
&\qquad\qquad \left. \times \frac{\Gamma(c_{ij} - \sum_{k=1}^{3} m_{ijk} + (1-f)w_1)}{\Gamma((1-f)w_1)\Gamma(c_{ij} - \sum_{k=1}^{3} m_{ijk} + 1)} \right] \\
&= I'J \left[\log \Gamma(w_1) - 3 \log \Gamma\left(\frac{1}{3} f w_1\right) - \log \Gamma((1-f)w_1) \right] \\
&\quad + \sum_{c=1}^{\max(c_{ij})} N_c (\log \Gamma(c+1) - \log \Gamma(c + w_1)) \\
&\quad + \sum_{k=1}^{3} \sum_{m_k=1}^{\max(m_{ijk})} N_{m_k} \left(\log \Gamma\left(m_k + \frac{1}{3} f w_1\right) - \log \Gamma(m_k + 1) \right) \\
&\quad + \sum_{c - \sum_{k=1}^{3} m_k = 1}^{\max(c_{ij} - \sum_{k=1}^{3} m_{ijk})} N_{c - \sum_{k=1}^{3} m_k} \left(\log \Gamma\left(c - \sum_{k=1}^{3} m_k + (1-f)w_1\right) \right. \\
&\qquad\qquad \left. - \log \Gamma\left(c - \sum_{k=1}^{3} m_k + 1\right) \right),
\end{aligned} \tag{3.21}$$

where $P_{0/0}$ is defined in Equation (3.12). Across I' background sites and J cells, N_c, N_{m_k} for $k = 1, 2, 3$, and $N_{c - \sum_{k=1}^{3} m_k}$ represent the unique occurrences of sequencing coverage c, of alternative nucleotide read counts m_k for $k = 1, 2, 3$, and of reference nucleotide read counts $c - \sum_{k=1}^{3} m_k$, respectively. Some terms, namely $\log\Gamma(c+1)$, $-\log\Gamma(m_k+1)$ for $k = 1, 2, 3$, and $-\log\Gamma(c - \sum_{k=1}^{3} m_k + 1)$, are constants, and thus they are not updated in the MCMC iterations.

The time complexity of Equation (3.21) is $\mathcal{O}(c)$, where c is the number of unique values in the set of values representing sequencing coverage and read counts for all four nucleotides across I' background sites and J cells. Since generally $IJK^2 \gg c$, the overall time complexity of model likelihood is $\mathcal{O}(IJK^2)$. It is worth noting that given I candidate variant sites and J cells, the time complexity of DelSIEVE is around 1.8 times greater than that of SIEVE due to the expanded genotype state space.

Priors

Similar to SIEVE, we use prior distributions predefined and implemented in BEAST 2 for hidden random variables in the DelSIEVE model. For the cell phylogeny given by $\mathcal{T}$ and $\boldsymbol{\beta}$, we set a prior following the Kingman coalescent process with an exponentially growing population, denoted

$$P(\mathcal{T}, \boldsymbol{\beta} \,|\, M, e), \tag{3.22}$$

where M and e are hidden random variables, representing the scaled population size and the exponential growth rate, respectively. The analytical form of Equation (3.22) is defined at length in [147].

The default prior for M in BEAST 2 is

$$P(M \,|\, \delta) = \frac{1}{\delta}, \tag{3.23}$$

where δ is the current proposed value of M.

As for e, the default prior is

$$e \,|\, \lambda, \epsilon \sim \text{Laplace}(\lambda, \epsilon), \tag{3.24}$$

where the default values of the fixed parameters are mean $\lambda = 10^{-3}$ and scale $\epsilon = 30.7$.

For η in Equation (3.2), an exponential prior distribution is chosen:

$$\eta \,|\, \gamma \sim \exp(\gamma), \tag{3.25}$$

where $\gamma = 1$.

For the relative deletion rate d in Equation (3.1), a uniform prior distribution is used:

$$d \,|\, \varphi \sim \text{Uniform}(0, \varphi), \tag{3.26}$$

where $\varphi = 1$.

For the hidden random variables in the model of sequencing coverage in Equations (3.4) to (3.6), a weak prior is set for t:

$$t \,|\, \rho \sim \text{Uniform}(0, \rho), \tag{3.27}$$

where $\rho = 1000$, while the prior for v is

$$v \,|\, \zeta \sim \exp(\zeta), \tag{3.28}$$

where $\zeta = 25$.

For the ADO rate θ defined either under the single ADO (Table 3.1) or under the locus dropout mode (Table 3.2), we use an uninformative prior:

$$\theta \,|\, u \sim \text{Uniform}(0, u), \tag{3.29}$$

where $u = 1$.

Regarding the hidden random variables in the model of nucleotide read counts in Equations (3.8), (3.10) and (3.11), an exponential prior is set for f:

$$f \,|\, \tau \sim \exp(\tau), \tag{3.30}$$

where $\tau = 0.025$, and a log normal prior for both w_1 and w_2:

$$\begin{aligned} w_1 \,|\, \xi_1, \psi_1 &\sim \text{Log-Normal}(\xi_1, \psi_1), \\ w_2 \,|\, \xi_2, \psi_2 &\sim \text{Log-Normal}(\xi_2, \psi_2), \end{aligned} \tag{3.31}$$

where we choose for w_1 the log-transformed mean $\xi_1 = 3.9$ (150 for untransformed) and the standard deviation $\psi_1 = 1.5$, and for w_2 the log-transformed mean $\xi_2 = 0.9$ (10 for untransformed) and the standard deviation $\psi_2 = 1.7$. The mean is log-transformed using

$$\xi_{\text{transformed}} = \log(\xi_{\text{untransformed}}) - \frac{\psi^2}{2}.$$

These values of the fixed parameters in Equation (3.31) are chosen to cover a wide range of possible values for w_1 and w_2.

Posterior and MCMC

The posterior distribution of the hidden random variables writes

$$\begin{aligned} & P\left(\mathcal{T}, \boldsymbol{\beta}, M, e, \eta, d, t, v, \theta, f, w_1, w_2 \,\middle|\, \mathcal{D}^{(1)}, \mathcal{D}^{(2)}\right) \\ =& \frac{1}{Z} P\left(\mathcal{D}^{(1)}, \mathcal{D}^{(2)} \,\middle|\, \mathcal{T}, \boldsymbol{\beta}, Q, \eta, t, v, \theta, f, w_1, w_2\right) \\ & \times P(\mathcal{T}, \boldsymbol{\beta} \,|\, M, e) P(M \,|\, \delta) P(e \,|\, \lambda, \epsilon) \\ & \times P(\eta \,|\, \gamma) P(Q \,|\, d) P(d \,|\, \varphi) \\ & \times P(t \,|\, \rho) P(v \,|\, \zeta) P(\theta \,|\, u) P(f \,|\, \tau) \\ & \times P(w_1 \,|\, \xi_1, \psi_1) P(w_2 \,|\, \xi_2, \psi_2), \end{aligned} \tag{3.32}$$

where $Z = P(\mathcal{D}^{(1)}, \mathcal{D}^{(2)})$ is a normalization constant, and the likelihood of the model and priors for hidden random variables are defined in Section DelSIEVE likelihood and Section Priors, respectively. To simplify the notation, we denote the hidden random variables in Equation (3.32) as $\Lambda = \{\mathcal{T}, \boldsymbol{\beta}, M, e, \eta, d, t, v, \theta, f, w_1, w_2\}$.

Since Z in Equation (3.32) is intractable to calculate, we employ the MCMC algorithm with Metropolis-Hastings kernel to sample from the posterior distribution. In this algorithm, a new state of the hidden random variables Λ^* is proposed based on its current state Λ following a proposal distribution $q(\Lambda^* \,|\, \Lambda)$. $q(\Lambda^* \,|\, \Lambda)$ is designed to ensure the reversibility and ergodicity of the underlying Markov chain. For DelSIEVE, in each iteration, a new state of a randomly selected hidden variable is accepted with probability

$$\min\left\{1, \frac{P\left(\Lambda^* \,\middle|\, \mathcal{D}^{(1)}, \mathcal{D}^{(2)}\right) q(\Lambda \,|\, \Lambda^*)}{P\left(\Lambda \,\middle|\, \mathcal{D}^{(1)}, \mathcal{D}^{(2)}\right) q(\Lambda^* \,|\, \Lambda)}\right\}. \tag{3.33}$$

We employ exactly the same proposal distributions as we used in SIEVE, which are defined in BEAST 2. Briefly, regarding the branch lengths of the tree, the heights of the internal nodes are adjusted. For the tree topology, we use multiple moves, including subtree swapping, Wilson-Balding, and subtree sliding, where the last two moves also change branch lengths as a side effect. With respect to unknown parameters, scaling and random Gaussian walks are used. For detailed description of the aforementioned moves, refer to [147] and Section Posterior and MCMC in Chapter 2.

To achieve accurate parameter and tree estimates, DelSIEVE employs a two stage sampling strategy, similarly to SIEVE (see Section Posterior and MCMC in Chapter 2).

Variant calling, ADO calling and maximum likelihood gene annotation

In the efficient computation of model likelihood using Equations (3.16) and (3.17), we marginalize out some hidden random variables: $\boldsymbol{g}_i^{(L)}$, $\boldsymbol{g}_i^{(A)}$, g'_{ij} and α_{ij}. Hence, the direct results from the MCMC sampling process are the posterior distributions of cell phylogeny and other unknown hidden random variables. We obtain the estimates of those marginalized hidden random variables as a post processing step, similarly to SIEVE. Specifically, we use the max-sum algorithm [72], by fixing the maximum clade credibility tree [148] and parameters estimated from the MCMC posterior samples. As a result, the variants, ADO states, as well as the locations of mutated genes on the inferred cell phylogeny are determined by identifying the maximum likelihood states of $\boldsymbol{g}_i^{(L)}$, g'_{ij} and α_{ij}, as well as $\boldsymbol{g}_i^{(A)}$, respectively.

Mutation event classification

DelSIEVE is able to discern 28 types of genotype transitions, which are classified into 17 types of mutation events (Table 3.5). Each genotype transition is a combinatorial result of single mutations, single back mutations and single deletions. Single mutations happen when 0 mutates to 1, or 1 and 1′ mutate to each other. Single back mutations occur when 1 or 1′ mutates to 0. Single deletions happen when an existing allele is lost during evolution, namely 0 or 1 deleted.

Since DelSIEVE encompasses the genotype state space modeled by SIEVE, it is capable of discerning all genotype transitions that SIEVE can handle, namely the first 12 rows in Table 3.5 (for detailed explaination see Section Variant calling, ADO calling, maximum likelihood gene annotation, and mutation event classification in Chapter 2). Those mutation events that only DelSIEVE is able to discern are explained as follows.

The single deletion which is not loss of heterozygosity (LOH; related to genotype transitions 0/0 → 0/- and 1/1 → 1/-) takes place when one allele is deleted from genotypes in which both alleles originally contained the same nucleotide, while the single deletion which is LOH (0/1 → 0/-, 0/1 → 1/- and 1/1′ → 1/-) happens when one allele is deleted from genotypes in which both alleles originally had different nucleotides. The coincident deletion and mutation (0/0 → 1/-) refers to the case when one allele is deleted, and the other is mutated of the wildtype, while the coincident deletion and back mutation (1/1 → 0/- and 1/1′ → 0/-) happens when one allele is deleted, and the other is mutated back to the reference nucleotide. The single deletion mutation addition (0/- → 1/-) takes place when the only allele of the reference-left single deletion genotype is mutated to an alternative nucleotide, while the single deletion back mutation addition happens when the mutated allele of the alternative-left single deletion genotype is mutated back to the reference nucleotide. The single deletion addition (0/- → - and 1/- → -) refers to the case when the only allele is deleted of the reference- and

Table 3.5: **28 types of genotype transitions that DelSIEVE is able to identify, with their interpretation as mutation events.** The genotype transitions correspond to possible changes of genotypes on a branch from the parent node to the child node. If any of these events occurs on independent branches of the phylogenetic tree, it is also considered as a parallel evolution event. The first 12 genotype transitions are also identifiable with SIEVE. LOH in the table represents loss of heterozygosity.

Genotype transition	Mutation event	Identifiable solely by DelSIEVE
0/0 → 0/1	Single mutation	No
0/0 → 1/1	Coincident homozygous double mutation	No
0/0 → 1/1′	Coincident heterozygous double mutation	No
0/1 → 0/0	Single back mutation	No
1/1 → 0/1	Single back mutation	No
1/1′ → 0/1	Single back mutation	No
1/1 → 0/0	Coincident double back mutation	No
1/1′ → 0/0	Coincident double back mutation	No
0/1 → 1/1	Homozygous single mutation addition	No
0/1 → 1/1′	Heterozygous single mutation addition	No
1/1′ → 1/1	Homozygous substitute single mutation	No
1/1 → 1/1′	Heterozygous substitute single mutation	No
0/0 → 0/-	Single deletion (not LOH)	Yes
1/1 → 1/-	Single deletion (not LOH)	Yes
0/1 → 0/-	Single deletion (LOH)	Yes
0/1 → 1/-	Single deletion (LOH)	Yes
1/1′ → 1/-	Single deletion (LOH)	Yes
0/0 → 1/-	Coincident deletion and mutation	Yes
1/1 → 0/-	Coincident deletion and back mutation	Yes
1/1′ → 0/-	Coincident deletion and back mutation	Yes
0/- → 1/-	Single deletion mutation addition	Yes
1/- → 0/-	Single deletion back mutation addition	Yes
0/- → -	Single deletion addition	Yes
1/- → -	Single deletion addition	Yes
0/0 → -	Coincident double deletion	Yes
0/1 → -	Coincident double deletion	Yes
1/1 → -	Coincident double deletion	Yes
1/1′ → -	Coincident double deletion	Yes

alternative-left single deletion genotypes. Finally, for the coincident double deletion (0/0 → -, 0/1 → -, 1/1 → - and 1/1′ → -) both of the alleles existing before are deleted.

3.2.2. ScDNA-seq data simulator

We generated simulated data by modifying the simulator we had used in SIEVE. The first change we made was to expand the rate matrix, according to which each genomic site evolved along the tree (Table 5.5). The rate matrix contains 14 genotypes encoded with nucleotides, allowing for mutations, back mutations, and deletions. It has one parameter, deletion rate, which is measured relatively to the mutation rate. Another change was that we implemented the locus dropout mode to allow more than one ADO to occur at each site for each cell. The simulator takes the same input configuration as SIEVE does.

The simulation process was similar to that in SIEVE. Briefly, with a given number of cells, a binary cell lineage tree was first simulated following the coalescent process under the strict molecular clock. For a given number of genomic sites, each site was initialized by randomly selecting one of four nucleotides to have a reference genotype. Next, with a given mutation rate and a relative deletion rate, each site was evolved independently along the tree following the rate matrix defined in Table 5.5. A genomic site is considered as a true SNV site if at least one cell has a genotype that is not wildtype. ADOs were then added on top of the simulated genotypes under either single ADO or locus dropout mode, as long as there were existing alleles. We recorded the true ADO states for all cells at the true SNV sites. Size factors in Equation (3.7) were generated from a normal distribution with the mean $= 1.2$ and the variance $= 0.2$. The sequencing coverage was simulated using a negative binomial distribution following Equations (3.4) to (3.6). The read counts of each nucleotide were then generated following a multinomial distribution.

3.2.3. Simulation design

We designed a series of simulations to benchmark the performance of DelSIEVE. We reused and modified the benchmarking framework in SIEVE.

We assumed that 40 tumor cells were sampled from an exponentially growing population, whose growth rate and effective population size are 10^{-4} and 10^4, respectively. We used the same mutation rates as in SIEVE, namely 10^{-6}, 8×10^{-6} and 3×10^{-5}. We selected two levels of deletion rate relative to the mutation rate: 0.1 and 0.25.

For each mutation rate, we chose such number of genomic sites that DataFilter would produce a certain amount of candidate variant sites or background sites. For mutation rate 10^{-6}, we evolved 10^4 genomic sites to have around $400 \sim 700$ candidate variant sites. For mutation rate 8×10^{-6}, 10^4 genomic sites were chosen to have around 4×10^3 background sites. For mutation rate 3×10^{-5}, 1.2×10^5 genomic sites were chosen to have at least 2.5×10^3 background sites. For the higher mutation rates of 8×10^{-6} and 3×10^{-5}, the chosen numbers of genomic sites resulted in $> 5 \times 10^3$ and $> 1.1 \times 10^5$ true SNV sites, respectively. Due to the consideration of runtime efficiency, they were subsetted before piping to downstream methods.

To this end, we first computed a targeted number of true SNV sites n_{target} using

$$n_{\text{target}} = \min(700, \frac{n'}{5}),$$

where n' is the number of background sites. Next, we randomly selected n_{target} sites out of the true SNV sites. Together with the n' background sites, the selected n_{target} true SNV sites formed the new simulated data. This ensured that the number of true SNV sites in the final simulated data for different mutation rates were within the same range, and the ratio between the number of background sites and the true SNV sites was at least 5 for mutation rates being 8×10^{-6} and 3×10^{-5}.

We considered both single ADO and locus dropout mode. The ADO rate for the former was $\theta = 0.163$, and for the latter $\theta = 0.3$.

Similar to SIEVE, we had different combinations of t and v in Equations (3.4) to (3.6) for various coverage qualities. For simulated data referred to as high coverage quality, we used high mean ($t = 20$) and low variance ($v = 2$) of allelic coverage. For medium coverage quality data, we used high mean ($t = 20$) and medium variance ($v = 10$). For low coverage quality data, we fixed low mean ($t = 5$) and high variance ($v = 20$).

Other parameters were fixed when simulating the data. We set w_1 and w_2 in Equation (3.11) to 100 and 2.5, respectively. Moreover, we set both the amplification and sequencing error rate to 10^{-3}, and thus the effective sequencing error rate in Equation (3.10) was $f \approx 2 \times 10^{-3}$.

Overall, we designed 36 simulation scenarios, each repeated 10 times.

Furthermore, for each of those genotypes related to somatic deletions, we filtered out results if the proportion of simulated ground truth was less than 0.1%. We also excluded results from mutation rate being 10^{-6} as too few somatic deletions were generated (less than 0.3%, 0.7% and 0.005% for alternative-left single deletion, reference-left single deletion and double deletion, respectively). For the same reason, results were also excluded from double deletion for mutation rate being 8×10^{-6} (less than 0.2% generated).

For double mutant genotype, we excluded results when mutation rate was 10^{-6} as less than 0.2% of such genotype was generated.

3.2.4. Measurement of the quality of variant calling and cell phylogeny accuracy

For assessing the results of variant and ADO calling, standard performance measures such as precision, recall, F1 score, and false positive rate (FPR) were used. DelSIEVE, SIEVE, SCIPhIN and Monovar were evaluated using these measures in the task of single and double mutant genotype calling.

Both DelSIEVE and SCIPhIN identify somatic deletions at preselected candidate sites. Hence, we subsetted the true somatic deletions to those at the candidate variant sites when computing the metrics. This barely influenced the recall and F1 score for alternative-left single deletion, as majority of the sites containing such genotype were captured in the selection of the candidate variant sites. For reference-left single deletion and double deletion genotype, however, restricting to candidate sites would inevitably decrease recall and F1 score, as sites having solely those genotypes would be missed in the preselection.

To assess the accuracy of cell phylogeny reconstruction, we used the same measurements as in SIEVE, namely the BS distance [153] for both the tree topology and branch lengths, as well as the normalized RF distance [154] for the tree topology only (see Section Measurement of cell phylogeny accuracy and quality of variant calling in Chapter 2). For DelSIEVE, SIEVE and SiFit, we computed both the BS and the normalized RF distance in the rooted tree mode. For SCIPhIN, we only computed the normalized RF distance as it only infers a rooted tree without branch lengths. We used R package phangorn to compute BS and normalized RF distance [155].

3.2.5. Configurations of methods

For Monovar (commit 68fbb68), we used the true values of θ and f as priors for false negative rate and false positive rate and default values for other options.

For SCIPhIN (commit 27e5ca6), we gave it the true value of f to avoid estimating its mean error rate (option "wildMean"), and ran it with 10^6 iterations with zygosity learned (option "lz" set to 1). We also set the penalty of computing the loss (option "llp") and parallel score (option "lpp") to 30. The command line is as follows:

```
sciphin -l 1000000 --lz 1 --ll 1 --lp 1 --llp 30 --lpp 30 --ese 0 \
--wildMean 0.002
```

To run SiFit (commit 9dc3774), we fed the required data with variants called by Monovar as a ternary matrix. We used the true values of θ and f as the prior for false negative rate and the estimated false positive rate, respectively. We ran it with 2×10^5 iterations.

For SIEVE, originally it only supported single ADO mode. In this contribution, we additionally equipped it with the locus dropout mode, which is now available along with DelSIEVE.

On the simulated data, we configured a strict molecular clock model for DelSIEVE and SIEVE, both of which was then run for 2×10^6 and 1.5×10^6 iterations for the first and the second sampling stages, respectively. The deletion rate was also inferred in the second sampling stage as it is related to the branch lengths of the cell phylogeny. Both DelSIEVE and SIEVE were configured to match the ADO type employed during the simulation process. This ensured consistency between the simulation and analysis, allowing for accurate comparisons and evaluations of the methods' performance.

On the real datasets, we instead used a log-normal relaxed molecular clock model to account for branch-wise substitution rate variation for DelSIEVE. To obtain better mixed Markov chains, we used an optimized relaxed clock model [156] rather than the default one in BEAST 2. We increased the number of iterations for both stages to 4×10^6 and 3.5×10^6, respectively. Both the deletion rate and parameters introduced by the relaxed molecular clock model were explored in the second sampling stage. To reduce the uncertainties introduced by the model, DelSIEVE was run in single ADO mode.

To run Sequenza on the real datasets, we used the bam2seqz command in the sequenza-utils package to convert bam files for normal and tumor cells to the Sequenza file format, which was subsequently binned with the seqz_binning command, using a window size of 50. With this file as input, we used the sequneza.fit command from Sequenza v3.0.0 to estimate the ploidy.

The SNVs were annotated using Annovar (version 2020 Jun. 08) [157]. The cell phylogeny was plotted in R (version 4.2.3) [176] using ggtree [158], and the genotype heatmap was plotted using ComplexHeatmap [159]. Besides, the comparison of sequencing coverages reported by DelSIEVE and Sequenza was performed and plotted using ggstatsplot [177].

3.3. Results

3.3.1. DelSIEVE accurately called somatic deletions

First, we used simulated data to benchmark one of DelSIEVE's asset functionalities, namely calling somatic deletions (Methods; Section Simulation design). DelSIEVE's performance was benchmarked against SCIPhIN [131] (Figures 3.2, 5.21 and 5.22). Here, SCIPhIN was given an advantage by fixing its mean error rate to the true effective sequencing error rate used in the simulation. DelSIEVE and SCIPhIN were evaluated in the task of calling alternative- and reference-left deletions, while only DelSIEVE was evaluated in the task of calling double deletion genotype, as it is the only method to call such genotype.

For calling alternative- and reference-left single deletion, DelSIEVE overall outperformed SCIPhIN, regardless of the type of ADOs (single or locus dropout) used in the simulated data (Figure 3.2a, b, Figure 5.21a-d, Figure 5.22a, b). When the data was of medium or high coverage quality (with high mean and low or medium variance of coverage), DelSIEVE achieved F1 scores with medians ≥ 0.87 and ≥ 0.76 for alternative- and reference-left single deletions, respectively (Figure 3.2a, b). In contrast, SCIPhIN had F1 scores with medians $\leq$ 0.28 for alternative-left single deletion and ≤ 0.01 for reference-left single deletion. The related recall (Figure 5.21a, c) and precision (Figure 5.21b, d) also showed DelSIEVE's superiority. In

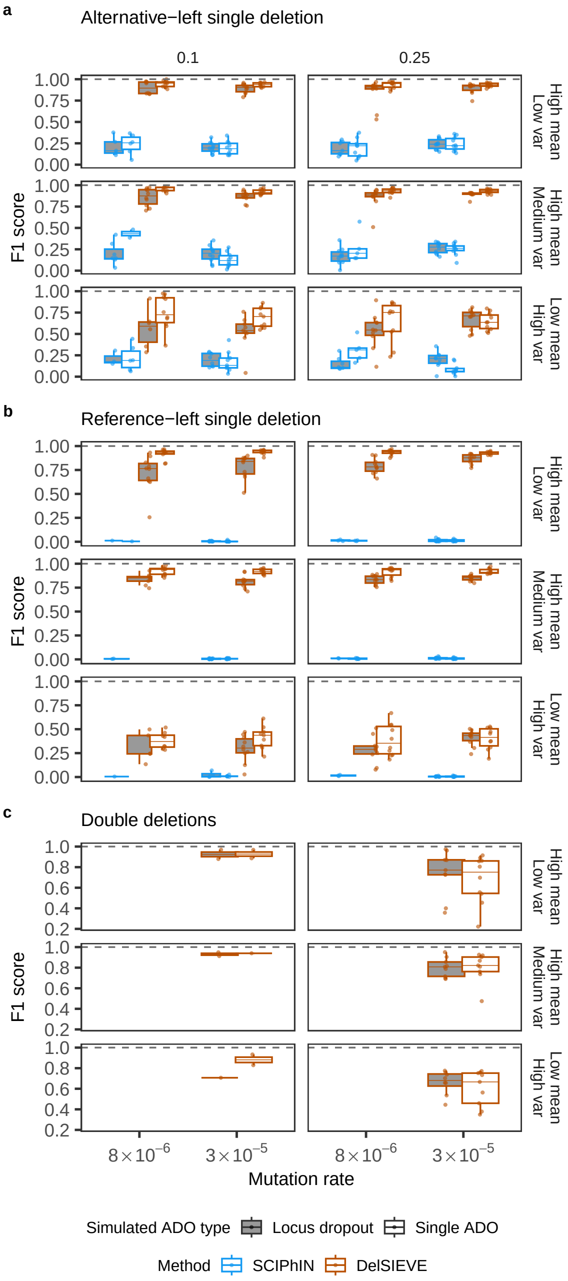


Figure 3.2: **F1 score for the benchmark of calling somatic deletions.** Varying are the mutation rate (the horizontal axis), the relative deletion rate (the vertical strip), the coverage quality (the horizontal strip) and the simulated ADO type (the shaded or blank boxes). Each simulation is repeated $n = 10$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. Data points were removed if the proportion of simulated ground truth was less than 0.1%. **a-c**, Box plots of the F1 score for calling alternative-left single deletion (**a**), reference-left single deletion (**b**), and double deletion (**c**). The results in **c** when mutation rate was 8×10^{-6} were omitted as very few double deletion were generated (less than 0.2%; see Section Simulation design).

particular, the high precision (≈ 1) and negligible FPR (≈ 0, see Figure 5.22a, b) of DelSIEVE indicate its high reliability in calling alternative- and reference-left single deletion genotypes.

When the data was of low coverage quality (low mean and high variance of coverage), the medians of F1 scores of DelSIEVE dropped to ≥ 0.55 and ≥ 0.29 for calling alternative- and reference-left single deletion genotypes, respectively, but still largely exceeded those of SCIPhIN (Figure 3.2a, b). The low quality of the data seemed to affect more the performance of DelSIEVE in calling reference-left single deletion compared to that in calling alternative-left single deletion (Figure 5.21a-d). This was expected since such low coverage provided very little information for calling reference-left single deletion. Furthermore, the FPR of DelSIEVE was still ≈ 0 for the low quality data.

We observed that the performance of DelSIEVE only slightly decreased when applied to data simulated under locus dropout mode, in comparison to the results obtained when it was applied to data simulated under single ADO mode. Given that DelSIEVE explicitly modeled the sequencing coverage, it was anticipated that data simulated under locus dropout mode would introduce additional uncertainties to the model.

DelSIEVE was the only method designed for explicitly calling double deletion genotype. Overall, in evaluation on simulated data, DelSIEVE obtained high medians of F1 scores ≥ 0.75 (Figure 3.2c). Its performance decreased as the relative deletion rate increased or the coverage quality of the data decreased (Figure 3.2c, Figure 5.21e, f), but the FPR kept at a negligible level (≈ 0; see Figure 5.22c).

3.3.2. DelSIEVE showed boosted performance in calling double mutant genotypes compared to SIEVE in the presence of somatic deletions.

We next assessed DelSIEVE's performance in calling single and double mutant genotypes against Monovar, SCIPhIN and SIEVE (Figures 3.3, 5.23 and 5.24). Regarding calling single mutant genotype, DelSIEVE and SIEVE performed comparatively well (minimum median F1 score of 0.9), and outperformed Monovar and SCIPhIN (minimum median F1 score 0.58 and 0.6, respectively; see Figure 3.3a). As mutation rate increased, the recall of both DelSIEVE and SIEVE slightly increased (Figure 5.23a), while the precision slightly decreased (Figure 5.23b), resulting in relatively constant F1 scores. In contrast, both Monovar and SCIPhIN experienced a decrease in both recall and precision as the mutation rate increased (Figure 5.23a, b). Consequently, their F1 scores declined, with SCIPhIN being more adversely affected compared to Monovar. Moreover, DelSIEVE and SIEVE had comparable recall (Figure 5.23a), while DelSIEVE showed higher precision (Figure 5.23b) and lower FPR (Figure 5.24a) than SIEVE did, especially when the mutation rate was high ($\geq 3 \times 10^{-5}$). We speculate that this might because SIEVE has to model the evident signal of somatic deletions as ADOs on top of single mutant genotype.

Additionally, as the mutation rate increased, the FPR of all methods also increased, with SCIPhIN exhibiting the most significant FPR increase (Figure 5.24a). It was noteworthy that, when the mutation rate was high ($\geq 3 \times 10^{-5}$), methods that incorporated cell phylogeny in variant calling, such as DelSIEVE, SIEVE and SCIPhIN, had slightly higher FPR in calling single mutant genotype compared to other methods, such as Monovar (Figure 5.24a). However, this loss was negligible compared to the advantage that SIEVE and DelSIEVE had over Monovar when precision, recall, and F1 were evaluated.

In the task of calling double mutant genotypes, SCIPhIN and Monovar obtained minimum median F1 scores 0.04 and 0.21, respectively, while SIEVE and DelSIEVE exhibited much higher performance with minimum median F1 scores 0.65 and 0.93, respectively (Figure 3.3b). More specifically, DelSIEVE and SIEVE had comparable recall (Figure 5.23c), but the former reached much higher precision than the latter (minimum medians 0.75 and 0.61, respectively; see Figure 5.23d). Again, this discrepancy in performance could be due to SIEVE's inclination

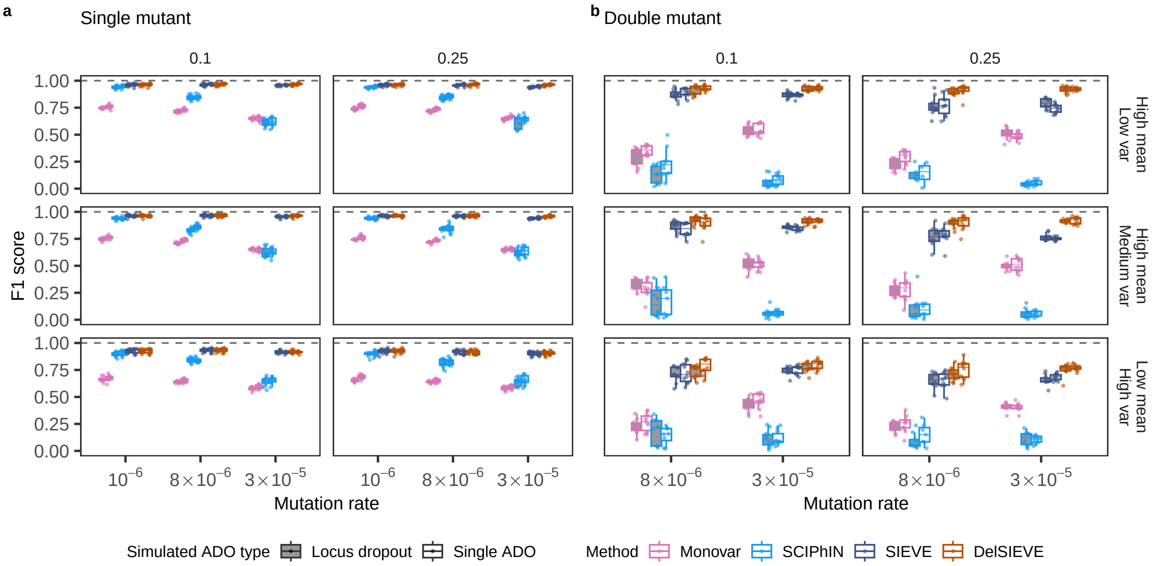


Figure 3.3: **F1 score for the benchmark of calling single and double mutant.** Varying are the mutation rate (the horizontal axis), the relative deletion rate (the vertical strip), the coverage quality (the horizontal strip) and the simulated ADO type (the shaded or blank boxes). Each simulation is repeated $n = 10$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. **a-b**, Box plots of the F1 score for calling single mutant (**a**) and double mutant (**b**). The results in **b** for mutation rate was 10^{-6} were omitted as too few double mutant were generated (less than 0.2%; see Section Simulation design).

to explain somatic deletions by modeling them as ADO events occurring within double mutant genotypes.

Besides, DelSIEVE had the lowest FPR (≈ 0) compared to other methods (Figure 5.24b). These findings highlighted the superior capability of DelSIEVE in accurately identifying double mutant genotypes in the presence of somatic deletions. On top of that, the slight advantage of Monovar over methods incorporating phylogeny observed for single mutant calling was not observed for double mutant calling. In contrast, in this task, Monovar had significantly elevated FPR compared to all other methods.

3.3.3. DelSIEVE outperformed SIEVE in calling ADOs on data with adequate coverage quality.

We then evaluated DelSIEVE's performance in calling single ADO and locus dropout against SIEVE (Figures 3.4, 5.25 and 5.26), which are the only two methods that can conduct these tasks. Though unsupported originally in SIEVE, locus dropout mode was implemented by us for the comparison (see Section Configurations of methods). The ADO type used during the simulation process was taken into consideration when configuring both DelSIEVE and SIEVE for analysis. As a result, the results of calling single ADO were accessible for data simulated under both single ADO and locus dropout modes. However, the results of calling locus dropout were only available for data simulated specifically under the locus dropout mode.

For calling single ADO, the performance of DelSIEVE and SIEVE were affected by the coverage quality of the data. When the data was of medium or high coverage quality, DelSIEVE

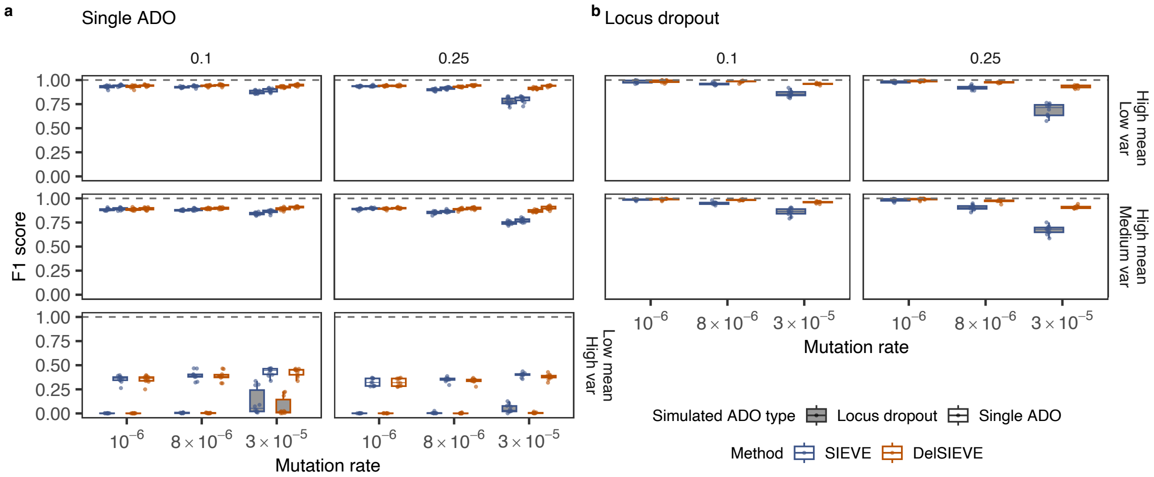


Figure 3.4: **F1 score for the benchmark of calling single ADO and locus dropout.** Varying are the mutation rate (the horizontal axis), the relative deletion rate (the vertical strip), the coverage quality (the horizontal strip) and the simulated ADO type (the shaded or blank boxes). Each simulation is repeated $n = 10$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. **a-b**, Box plots of the F1 score for calling single ADO (**a**) and locus dropout (**b**). The F1 score were unavailable in **b** when data was of low coverage quality due to unavailable precision.

reached a minimum median F1 score 0.9, higher than SIEVE (0.77; see Figure 3.4a). The performance of DelSIEVE remained consistent regardless of changes in the mutation rate and relative deletion rate, in contrast to SIEVE. This was anticipated because higher mutation and deletion rates resulted in an increased number of somatic deletions being generated. DelSIEVE was capable of differentiating somatic deletions from ADOs by incorporating them into the model. In contrast, SIEVE wrongly accounted for somatic deletions as ADOs occurring within single or double mutant genotypes. This behavior of SIEVE reduced the recall and precision, and increased FPR (Figure 5.25a, b, Figure 5.26a), similarly to its inferior performance in calling single and double mutant genotypes compared to DelSIEVE (see the previous section).

The performance of both DelSIEVE and SIEVE in calling single ADO declined when the data had low coverage quality (Figure 3.4a, Figure 5.25a, b, Figure 5.26a). This decrease in performance was further exacerbated when the data was simulated under the locus dropout mode, as compared to when it was simulated under the single ADO mode. The decrease in performance can be attributed to two primary factors. Firstly, data of low coverage quality contained more noise compared to that of higher coverage quality. The locus dropouts added even more noise on top of that. Secondly, the more complex model versions operating under the locus dropout mode inherently introduced more uncertainty to the results.

For calling locus dropout from data of medium or high coverage quality, DelSIEVE showed a minimum median F1 score of 0.91, higher than SIEVE did (0.68; see Figure 3.4b). Specifically, DelSIEVE and SIEVE were comparable in terms of recall (Figure 5.25c), but the former had a higher precision and lower FPR than the latter as the mutation rate and relative deletion rate increased (Figure 5.25d, Figure 5.26b). However, when the data was of low coverage quality, both methods reported no locus dropout, resulting in zero recall and FPR as well as unavailable precision and F1 score.

Since the quality of the real data resembles more that of low coverage quality, we decided to configure DelSIEVE under the single ADO mode to reduce the amount of uncertainties introduced.

3.3.4. DelSIEVE estimated cell phylogeny with comparable accuracy to SIEVE.

We further benchmarked DelSIEVE's performance in reconstructing the cell phylogeny against SiFit, SCIPhIN and SIEVE (Figure 5.27). To account for both tree structure and branch lengths in the evaluation, we used branch score (BS) distance as the metric. The results of SCIPhIN were excluded in the computation of BS score as it only reported the tree structure. Both DelSIEVE and SIEVE outperformed SiFit, showing the advantage of correcting the acquisition bias (Figure 5.27a). When the mutation rate was higher ($\geq 8 \times 10^{-6}$), DelSIEVE reported cell phylogenies with longer branch lengths than SIEVE and showed a bit larger BS score. This may be due to the fact that DelSIEVE, as a more complex model, with more considered genotypes, allowed more genotype transitions on the branches.

We then used the normalized RF distance as the metric, which only considered the tree structure. The performance of DelSIEVE and SIEVE in tree reconstruction was comparable in estimating the tree structure (maximum medium normalized RF distance 0.29 and 0.28, respectively), and was lower compared to SiFit (maximum median normalized RF distance 0.37) and SCIPhIN (0.33; see Figure 5.27b), especially when the mutation rate increased.

3.3.5. DelSIEVE reliably identified several somatic deletions in TNBC cells.

We applied DelSIEVE to real world scDNA-seq datasets analyzed previously in SIEVE with exactly the same input, configuring similarly a relaxed molecular clock model to account for branch-wise rate variation (see Section Configurations of methods). For scWES dataset TNBC16 [2], DelSIEVE reported a maximum clade credibility (MCC) cell phylogeny with a visually long trunk, supported by high posterior probabilities (Figures 3.5 and 5.28). The cell phylogeny was similar to that reported by SIEVE, with the normalized RF and the BS distances being 0.07 and 3.88×10^{-6}, respectively.

DelSIEVE identified the same types of mutation events reported by SIEVE, except for single back mutation. In terms of numbers, DelSIEVE explained the same data with less single mutations. Specifically, DelSIEVE identified 31 coincident homozygous double mutations (transitions from 0/0 to 1/1; 44 for SIEVE), eight homozygous single mutation additions (from 0/1 to 1/1; nine for SIEVE) and two parallel single mutations (from 0/0 to 0/1 that occurred more than once in the tree; same for SIEVE). SIEVE identified seven single back mutations (from 0/1 to 0/0; *BRD8*, *COL6A5*, *GRB14*, *MYRF*, *RHOJ*, *SEMA3A*, *TMX4*), narrating an evolutionary story of acquiring single mutations in these genes on the trunk of the tree, followed by losing them through single back mutations, resulting in these mutations possessed by only a subgroup of cells (a2, a3, a5 and a7). Reporting the same mutations in the same group of cells, DelSIEVE, however, narrated a more straightforward, parsimonious alternative, where cell a2, a3, a5 and a7 acquired these mutations directly from their most recent common ancestor.

In addition, DelSIEVE identified mutation events where somatic deletions were involved, including a large number of 245 coincident deletions and mutations (from 0/0 to 1/-), three single deletions which could be categorized as LOH (from 0/1 to 0/- or 1/-, or from $1/1'$ to 1/-), ten single deletions which were not LOH (from 0/0 to 0/-, or from 1/1 to 1/-), and finally ten single deletion mutation additions (from 0/- to 1/-). For instance, DelSIEVE inferred that gene *NEK1* and *NEK5*, which had been reported to be related to breast tumors [178],

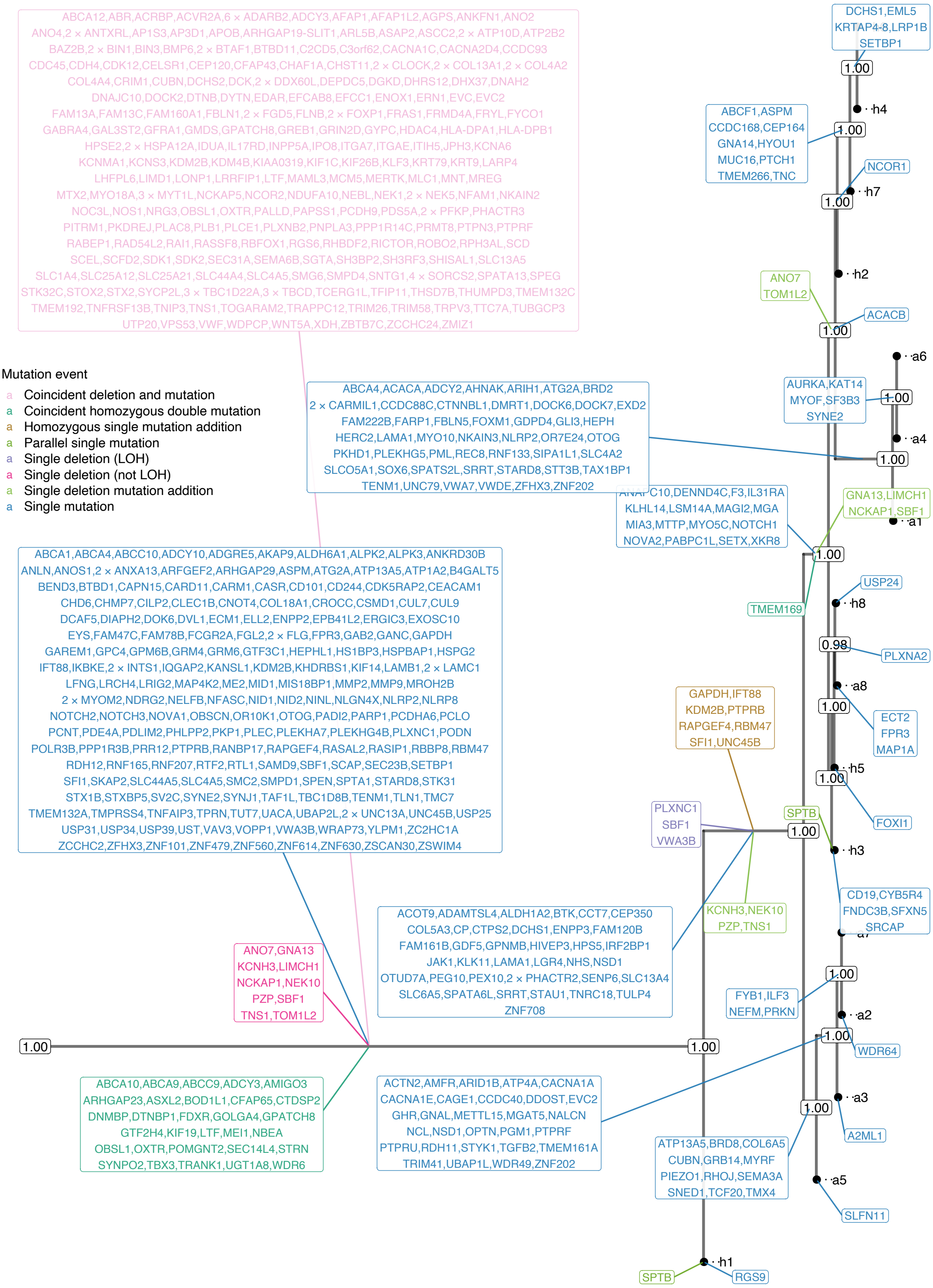

ABCA12,ABR,ACRBP,ACVR2A,6 × ADARB2,ADCY3,AFAP1,AFAP1L2,AGPS,ANKFN1,ANO2
ANO4,2 × ANTXRL,AP1S3,AP3D1,APOB,ARHGAP19-SLIT1,ARL5B,ASAP2,ASCC2,2 × ATP10D,ATP2B2
BAZ2B,2 × BIN1,BIN3,BMP6,2 × BTAF1,BTBD11,C2CD5,C3orf62,CACNA1C,CACNA2D4,CCDC93
CDC45,CDH4,CDK12,CELSR1,CEP120,CFAP43,CHAF1A,CHST11,2 × CLOCK,2 × COL13A1,2 × COL4A2
COL4A4,CRIM1,CUBN,DCHS2,DCK,2 × DDX60L,DEPDC5,DGKD,DHRS12,DHX37,DNAH2
DNAJC10,DOCK2,DTNB,DYTN,EDAR,EFCAB8,EFCC1,ENOX1,ERN1,EVC,EVC2
FAM13A,FAM13C,FAM160A1,FBLN1,2 × FGD5,FLNB,2 × FOXP1,FRAS1,FRMD4A,FRYL,FYCO1
GABRA4,GAL3ST2,GFRA1,GMDS,GPATCH8,GREB1,GRIN2D,GYPC,HDAC4,HLA-DPA1,HLA-DPB1
HPSE2,2 × HSPA12A,IDUA,IL17RD,INPP5A,IPO8,ITGA7,ITGAE,ITIH5,JPH3,KCNA6
KCNMA1,KCNS3,KDM2B,KDM4B,KIAA0319,KIF1C,KIF26B,KLF3,KRT79,KRT9,LARP4
LHFPL6,LIMD1,LONP1,LRRFIP1,LTF,MAML3,MCM5,MERTK,MLC1,MNT,MREG
MTX2,MYO18A,3 × MYT1L,NCKAP5,NCOR2,NDUFA10,NEBL,NEK1,2 × NEK5,NFAM1,NKAIN2
NOC3L,NOS1,NRG3,OBSL1,OXTR,PALLD,PAPSS1,PCDH9,PDS5A,2 × PFKP,PHACTR3
PITRM1,PKDREJ,PLAC8,PLB1,PLCE1,PLXNB2,PNPLA3,PPP1R14C,PRMT8,PTPN3,PTPRF
RABEP1,RAD54L2,RAI1,RASSF8,RBFOX1,RGS6,RHBDF2,RICTOR,ROBO2,RPH3AL,SCD
SCEL,SCFD2,SDK1,SDK2,SEC31A,SEMA6B,SGTA,SH3BP2,SH3RF3,SHISAL1,SLC13A5
SLC1A4,SLC25A12,SLC25A21,SLC44A4,SLC4A5,SMG6,SMPD4,SNTG1,4 × SORCS2,SPATA13,SPEG
STK32C,STOX2,STX2,SYCP2L,3 × TBC1D22A,3 × TBCD,TCERG1L,TFIP11,THSD7B,THUMPD3,TMEM132C
TMEM192,TNFRSF13B,TNIP3,TNS1,TOGARAM2,TRAPPC12,TRIM26,TRIM58,TRPV3,TTC7A,TUBGCP3
UTP20,VPS53,VWF,WDPCP,WNT5A,XDH,ZBTB7C,ZCCHC24,ZMIZ1
DCHS1,EML5
KRTAP4-8,LRP1B
SETBP1
1.00
h4
ABCF1,ASPM
CCDC168,CEP164
GNA14,HYOU1
MUC16,PTCH1
TMEM266,TNC
1.00
NCOR1
h7
1.00
h2
ANO7
TOM1L2
ACACB
1.00
a6
Mutation event
a Coincident deletion and mutation
a Coincident homozygous double mutation
a Homozygous single mutation addition
a Parallel single mutation
a Single deletion (LOH)
a Single deletion (not LOH)
a Single deletion mutation addition
a Single mutation
ABCA4,ACACA,ADCY2,AHNAK,ARIH1,ATG2A,BRD2
2 × CARMIL1,CCDC88C,CTNNBL1,DMRT1,DOCK6,DOCK7,EXD2
FAM222B,FARP1,FBLN5,FOXM1,GDPD4,GLI3,HEPH
HERC2,LAMA1,MYO10,NKAIN3,NLRP2,OR7E24,OTOG
PKHD1,PLEKHG5,PML,REC8,RNF133,SIPA1L1,SLC4A2
SLCO5A1,SOX6,SPATS2L,SRRT,STARD8,STT3B,TAX1BP1
TENM1,UNC79,VWA7,VWDE,ZFHX3,ZNF202
AURKA,KAT14
MYOF,SF3B3
SYNE2
1.00
a4
1.00
ANAPC10,DENND4C,F3,IL31RA
KLHL14,LSM14A,MAGI2,MGA
MIA3,MTTP,MYO5C,NOTCH1
NOVA2,PABPC1L,SETX,XKR8
GNA13,LIMCH1
NCKAP1,SBF1
a1
1.00
TMEM169
USP24
h8
0.98
PLXNA2
a8
1.00
ECT2
FPR3
MAP1A
h5
1.00
FOXI1
ABCA1,ABCA4,ABCC10,ADCY10,ADGRE5,AKAP9,ALDH6A1,ALPK2,ALPK3,ANKRD30B
ANLN,ANOS1,2 × ANXA13,ARFGEF2,ARHGAP29,ASPM,ATG2A,ATP13A5,ATP1A2,B4GALT5
BEND3,BTBD1,CAPN15,CARD11,CARM1,CASR,CD101,CD244,CDK5RAP2,CEACAM1
CHD6,CHMP7,CILP2,CLEC1B,CNOT4,COL18A1,CROCC,CSMD1,CUL7,CUL9
DCAF5,DIAPH2,DOK6,DVL1,ECM1,ELL2,ENPP2,EPB41L2,ERGIC3,EXOSC10
EYS,FAM47C,FAM78B,FCGR2A,FGL2,2 × FLG,FPR3,GAB2,GANC,GAPDH
GAREM1,GPC4,GPM6B,GRM4,GRM6,GTF3C1,HEPHL1,HS1BP3,HSPBAP1,HSPG2
IFT88,IKBKE,2 × INTS1,IQGAP2,KANSL1,KDM2B,KHDRBS1,KIF14,LAMB1,2 × LAMC1
LFNG,LRCH4,LRIG2,MAP4K2,ME2,MID1,MIS18BP1,MMP2,MMP9,MROH2B
2 × MYOM2,NDRG2,NELFB,NFASC,NID1,NID2,NINL,NLGN4X,NLRP2,NLRP8
NOTCH2,NOTCH3,NOVA1,OBSCN,OR10K1,OTOG,PADI2,PARP1,PCDHA6,PCLO
PCNT,PDE4A,PDLIM2,PHLPP2,PKP1,PLEC,PLEKHA7,PLEKHG4B,PLXNC1,PODN
POLR3B,PPP1R3B,PRR12,PTPRB,RANBP17,RAPGEF4,RASAL2,RASIP1,RBBP8,RBM47
RDH12,RNF165,RNF207,RTF2,RTL1,SAMD9,SBF1,SCAP,SEC23B,SETBP1
SFI1,SKAP2,SLC44A5,SLC4A5,SMC2,SMPD1,SPEN,SPTA1,STARD8,STK31
STX1B,STXBP5,SV2C,SYNE2,SYNJ1,TAF1L,TBC1D8B,TENM1,TLN1,TMC7
TMEM132A,TMPRSS4,TNFAIP3,TPRN,TUT7,UACA,UBAP2L,2 × UNC13A,UNC45B,USP25
USP31,USP34,USP39,UST,VAV3,VOPP1,VWA3B,WRAP73,YLPM1,ZC2HC1A
ZCCHC2,ZFHX3,ZNF101,ZNF479,ZNF560,ZNF614,ZNF630,ZSCAN30,ZSWIM4
GAPDH,IFT88
KDM2B,PTPRB
RAPGEF4,RBM47
SFI1,UNC45B
PLXNC1
SBF1
VWA3B
SPTB
1.00
h3
CD19,CYB5R4
FNDC3B,SFXN5
SRCAP
a7
ACOT9,ADAMTSL4,ALDH1A2,BTK,CCT7,CEP350
COL5A3,CP,CTPS2,DCHS1,ENPP3,FAM120B
FAM161B,GDF5,GPNMB,HIVEP3,HPS5,IRF2BP1
JAK1,KLK11,LAMA1,LGR4,NHS,NSD1
OTUD7A,PEG10,PEX10,2 × PHACTR2,SENP6,SLC13A4
SLC6A5,SPATA6L,SRRT,STAU1,TNRC18,TULP4
ZNF708
KCNH3,NEK10
PZP,TNS1
ANO7,GNA13
KCNH3,LIMCH1
NCKAP1,NEK10
PZP,SBF1
TNS1,TOM1L2
1.00
FYB1,ILF3
NEFM,PRKN
a2
1.00
1.00
WDR64
ABCA10,ABCA9,ABCC9,ADCY3,AMIGO3
ARHGAP23,ASXL2,BOD1L1,CFAP65,CTDSP2
DNMBP,DTNBP1,FDXR,GOLGA4,GPATCH8
GTF2H4,KIF19,LTF,MEI1,NBEA
OBSL1,OXTR,POMGNT2,SEC14L4,STRN
SYNPO2,TBX3,TRANK1,UGT1A8,WDR6
ACTN2,AMFR,ARID1B,ATP4A,CACNA1A
CACNA1E,CAGE1,CCDC40,DDOST,EVC2
GHR,GNAL,METTL15,MGAT5,NALCN
NCL,NSD1,OPTN,PGM1,PTPRF
PTPRU,RDH11,STYK1,TGFB2,TMEM161A
TRIM41,UBAP1L,WDR49,ZNF202
a3
1.00
A2ML1
ATP13A5,BRD8,COL6A5
CUBN,GRB14,MYRF
PIEZO1,RHOJ,SEMA3A
SNED1,TCF20,TMX4
a5
SLFN11
h1
SPTB
RGS9

Figure 3.5: **Results of phylogenetic inference for the TNBC16 dataset.** Shown is DelSIEVE's maximum clade credibility tree. Tumor cell names are annotated to the leaves of the tree. The numbers at each node represent the posterior probabilities (threshold $p > 0.5$). At each branch, depicted in different colors are non-synonymous genes that are either TNBC-related single mutations (in blue) or other mutation events (in other colors).

experienced both a deletion and a mutation on the trunk, resulting in all sequenced cells having genotype 1/-. Another gene, *LIMCH1*, known to be related to TNBC [179], had an allele deleted first on the trunk (genotype changed from 0/0 to 0/-), and then the left allele mutated for a subgroup of cells (genotype changed from 0/- to 1/-). The substantial amount of evolutionary events related to deletions highlights the importance of the extended functionality of DelSIEVE as compared to SIEVE.

In total, DelSIEVE identified 5,893 variant sites, close to 5,895 variant sites reported by SIEVE (Figure 3.6). Among the 683 sites inferred by DelSIEVE that contain somatic deletions (mostly 1/-; 11.6% of all variant sites), 377 were previously determined according to SIEVE to have double mutant genotypes and the remaining 306 to have single mutant genotype. This observation was in accordance with the simulation results, where SIEVE inclined to explaining somatic deletions as ADO events within single and double mutant genotyps to accommodate to the characteristics of the data, showing reliability to the results of DelSIEVE. The proportion of genotypes called by DelSIEVE and SIEVE were summarized in Table 5.6 (same for the following datasets).

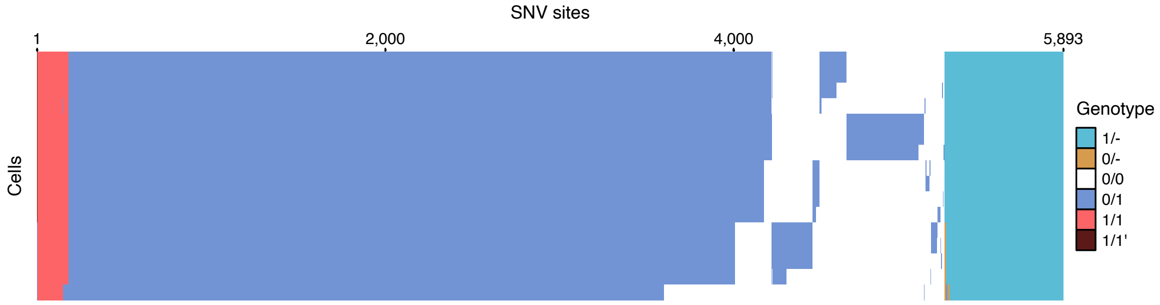


Figure 3.6: **Results of variant calling for the TNBC16 dataset.** Cells in the row are in the same order as that of leaves in the phylogenetic tree in Figure 3.5.

To further validate the ability of DelSIEVE to reliably call deletions, we inspected whether the sites identified as deleted displayed also a lower coverage than sites with neutral copy number. We next compared the strength of the coverage reduction effect on deleted sites to a dedicated copy number calling method, Sequenza [173] (Figure 3.7). The comparison was performed only for the sites shared between the input data of both methods, which, in this case, were all 5,912 candidate variant sites. Since Sequenza was designed to apply to bulk-seq data and only reported copy number (CN) at the clone (or subclone) level, we harmonized the resolution of DelSIEVE's results with Sequenza to ensure a fair comparison. To this end, we adjusted DelSIEVE to operate at the clone level as well. In other words, for this comparison, we considered all cells at a given site to contain somatic deletions if at least one cell indicated the presence of a deletion.

As expected, we observed that for DelSIEVE the mean value of sequencing coverages (denoted by $\hat{\mu}$ in Figure 3.7) in the group of sites with somatic deletions (3.95) was significantly lower compared to the mean for sites without somatic deletions (24.01, respectively), with effect size Cohen's d = 0.61. In contrast, the mean coverage for 44 sites identified as containing somatic deletions by Sequenza was 39.58, significantly larger than 21.56, the mean coverage for sites with amplifications (Cohen's d = 0.54), controverting Sequenza's copy number calls. Furthermore, a direct comparison revealed that sites identified as deleted by DelSIEVE showed much lower coverage levels than those identified as deleted by Sequenza (Cohen's d = 2.82). This indicates that DelSIEVE calls deletions more reliably than Sequenza.

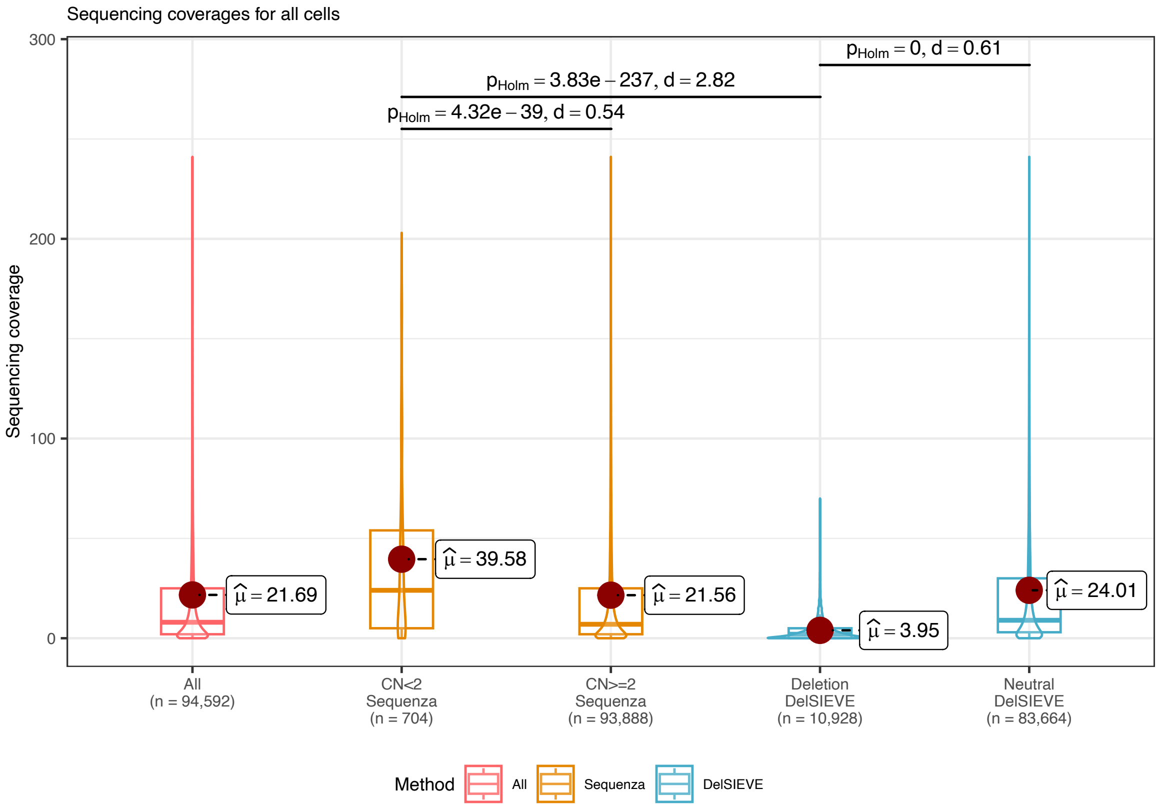


Figure 3.7: **Results of clone-wise sequencing coverage comparison for TNBC16 between DelSIEVE and Sequenza [173].** Compared were the sites shared between the input data of both methods. The resolution of variant calling was clone-wise in order to conduct a fair comparison. For Sequenza, sites were divided into two groups with copy number (CN) < 2 and ≥ 2, respectively. For DelSIEVE, sites were also divided into two groups, one with somatic deletions, the other copy neutral. Sequencing coverage across all cells at all sites were plotted for reference. In each group, the violin and the box plots matched the color of the method and showed the distribution of the sequencing coverage, while the burgundy dot denoted its mean value $\hat{\mu}$. The total number of dots in each group, which was the product of the number of cells (16) and the number of sites in each group, was marked with n on the horizontal axis. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. Within- and between-group comparisons were conducted between CN < 2 and ≥ 2 of Sequenza, between somatic deletions and copy neutral of DelSIEVE, and between CN < 2 of Sequenza and somatic deletions of DelSIEVE. For each comparison, shown were the p-value corrected by Holm–Bonferroni method and the absolute value of the effect size (Cohen's d).

3.3.6. DelSIEVE identified rare somatic mutations in CRC cells.

We then applied DelSIEVE to a scWGS dataset, CRC28 [1]. The estimated cell phylogeny was supported by high posterior probabilities with a long trunk (Figures 5.29 and 5.30), which was similar to that reported by SIEVE (the normalized RF and the BS distances were 0.08 and 8.03×10^{-7}, respectively). In particular, tumor proximal (TP) and tumor distal (TD) cells also formed a closer clade compared to tumor central (TC) cells in the tree reported by DelSIEVE. This suggested that, like SIEVE, DelSIEVE also inferred regular tumor growth and limited cell migration.

Similar to SIEVE, DelSIEVE annotated mutations of known CRC driver genes, for instance, *APC*, and of genes related to the metastatic progression of CRC, such as *ASAP1* and *RGL2* on the trunk of the tree. However, DelSIEVE identified more mutation events than SIEVE, including two coincident deletions and mutations, one single deletion which was not LOH, and one single deletion mutation addition. For example, DelSIEVE identified that *ACSL5*, potentially related to intestinal carcinogenesis [180], underwent a somatic deletion of one allele (genotype changed from 0/0 to 0/-) on the trunk and a mutation to the left allele (genotype changed from 0/- to 1/-) for the most recent common ancestor of TP and TD cells. Overall, DelSIEVE found very few mutation events that were not single mutations, indicating that single mutations dominated the evolutionary process of this sample.

DelSIEVE identified the same number of variant sites as SIEVE (8,029; see Figure 5.31), in which 13 sites contained somatic deletions (mostly 1/-; 0.16% of all variant sites). According to SIEVE, nine of those sites were inferred to have double mutant genotypes and four to have single mutant genotype. The contrasting results obtained by DelSIEVE, with multiple somatic deletions identified in TNBC16 but only few in CRC28, underscored an important feature of the method. While DelSIEVE employs a sophisticated modeling approach, it primarily relies on the data for the inference. In other words, the detection of somatic deletions was driven solely by the characteristics of the data itself and is not enforced by the model when the deletions are not there.

We further conducted a comparative analysis of the sequencing coverage between sites that were identified to contain somatic deletions and those that did not, using both DelSIEVE and Sequenza (Figures 5.32 to 5.34). Specifically, as CRC28 comprised tumor cells originating from distinct anatomical locations (denoted TP, TC, and TD cells), our comparison was conducted at the subclone resolution. This resolution represented the highest achievable level of detail that Sequenza could provide for this specific dataset, and we adjusted the resolution of DelSIEVE accordingly.

For TP cells (cancer tissue 1 in Figure 5.29; with nine cells) and TC cells (cancer tissue 3; with 12 cells), we could only inspect the results of DelSIEVE as there is no corresponding bulk sample for Sequenza. We observed noticeable differences of coverage between sites with and without somatic deletions called by DelSIEVE: for TP cells, the mean coverage $\hat{\mu} = 1.54$ for sites with somatic deletions was significantly lower than $\hat{\mu} = 6.37$ for sites without deletions Cohen's d = 0.59; Figure 5.32). This difference was also significant for the TC cells ($\hat{\mu} = 2.9$ for sites with somatic deletions, 10.26 for sites without, Cohen's d = 0.63; Figure 5.34).

For TD cells (cancer tissue 2; with seven cells), both DelSIEVE and Sequenza had lower $\hat{\mu}$ for sites containing somatic deletions compared to sites without deletions (Figure 5.33a). DelSIEVE exhibited a clear distinction, with a significantly lower $\hat{\mu}$ of 1.76 for sites with somatic deletions compared to 7.41 for sites without, resulting in Cohen's d = 0.5. Conversely, the difference in $\hat{\mu}$ was negligible for Sequenza, with values of 6.85 and 7.97 for sites with and without somatic deletions, respectively, resulting in Cohen's d = 0.1. Additionally, there was an evident difference in $\hat{\mu}$ between sites with somatic deletions identified by DelSIEVE and Sequenza, as indicated by a Cohen's d effect size of 0.5. These findings highlighted the divergent performance of DelSIEVE and Sequenza in calling somatic deletions for TD cells, where the results of the latter might not be reliable from the viewpoint of the conducted comparisons.

To further inspect the results from Sequenza, we visualized its reported CNs in TD cells across the entire genome (Figure 5.33b). The visualization clearly revealed that Sequenza inferred a substantial number of CNs other than 2 for each chromosome. Moreover, these CNs frequently exhibited fluctuations in their values, indicating that the method might be fitting to the noise rather than accurately capturing true CN states. These findings indicate that a significant portion of the CNs inferred from Sequenza could potentially be false positives.

3.3.7. DelSIEVE identified rare somatic mutations in CRC samples mixed with normal cells.

We finally analyzed another scWES dataset, CRC48 (CRC0827 in [3]). DelSIEVE pinpointed two tumor subclones, associated with their anatomical locations, each subclone containing exactly the same cells as in SIEVE (Figures 5.35 and 5.36). The rest of the cells collected from tumor biopsies were clustered together with cells from adenomatous polyps, suggesting that they might be normal cells residing inside cancer tissues, as pointed out by both the original study [3] and SIEVE. There were some distinctions between the cell phylogenies reported by DelSIEVE and SIEVE, with normalized RF and BS distances being 0.33 and 1.99×10^{-6}, respectively. This discrepancy is higher than observed for previous datasets, and might be due to the overall lower signal level in the data. Indeed, the CRC48 dataset has a substantially lower ratio between the number of candidate variant sites and the number of cells ($707/48 \approx 14.7$) compared to TNBC16 ($5912/16 = 369.5$) and CRC28 ($8470/28 = 302.5$).

DelSIEVE identified many single mutations on the branch leading to two tumor subclones, including a reported CRC driver mutation in gene *SYNE1* [161], as well as a mutation related to DNA mismatch repair, in gene *MLH3* [164], both of which were also identified on the same branch by SIEVE. Moreover, DelSIEVE found two parallel single mutations (*CHD3* and *PLD2*), which were also reported by SIEVE for the same cells. Furthermore, DelSIEVE identified only one site containing somatic deletions (among 679 variant sites, and only 0/-; see Figure 5.37), which was previously inferred by SIEVE to have single mutant genotype.

We conducted a comparative analysis of the site-wise sequencing coverage between sites that were identified to contain somatic deletions and those that did not, for cancer tissue 1 (Figure 5.38; with 17 cells) as well as cancer tissue 2 (Figure 5.39; with 18 cells). The comparisons were performed at the subclone resolution associated with the anatomic locations. Sites identified by DelSIEVE as containing somatic deletions showed much more pronounced mean coverage differences compared to sites without deletions, both for cancer tissue 1 (Cohen's d $= 0.4$) and for cancer tissue 2 (d $= 0.47$). These mean coverage differences between sites identified as deleted or not by Sequenza were negligible for both subclones (Cohen's d $= 0.06$ for cancer tissue 1; d $= 0.09$ for cancer tissue 2). Moreover, mean coverage was much lower for sites identified to carry somatic deletions by DelSIEVE than for sites identified as such by Sequenza (Cohen's d $= 0.46$ for cancer tissue 1; d $= 0.5$ for cancer tissue 2). For adenomatous polyps, DelSIEVE reported no somatic deletions, so we only compared the results of Sequenza (Figure 5.40). Countering the expected effect of deletions, we observed a higher mean coverage for the sites identified by Sequenza to have CN < 2 (37.97) than sites with CN ≥ 2 (35.76), though the difference was negligible (Cohen's d $= 0.03$). These findings again validated the deletion calls made by DelSIEVE and raised doubts about the CNs called by Sequenza in the context of the comparisons we performed regarding the sequencing coverages.

3.4. Discussion

We present DelSIEVE, a statistical method designed to jointly infer somatic deletions, SNVs, and the cell phylogeny from scDNA-seq data. Built upon SIEVE, which combines inference of SNVs and cell phylogeny, DelSIEVE takes a step forward by allowing for the occurrence of somatic deletions during the evolution of the tumor. In a nutshell, DelSIEVE features a statistical phylogenetic model with genotypes relating both to somatic deletions and to single and double mutants, a model of raw read counts allowing for both single ADO and locus dropout, a mechanism for acquisition bias correction for the branch lengths, and a trunk in the cell phylogeny for clonal mutations.

Somatic deletions often play an essential role in tumor evolution. Although our previous work, SIEVE, does account for the FSA in the statistical phylogenetic model, it only considers somatic mutations with nucleotide substitutions. Thus, it is not versatile enough to apply to data where somatic deletions are present. We have shown that for such data SIEVE tends to explain somatic deletions as a result of ADOs, with an inflated amount of single and double mutant genotypes inferred. The inclusion of somatic deletions in DelSIEVE fills this missing part in the puzzle. In particular, compared to SIEVE, DelSIEVE exhibits boosted performance in terms of calling double mutant genotypes, while performs similarly in estimating cell phylogeny and calling single mutant genotype.

The difficulty of identifying somatic deletions is mainly due to the similarity between the sequencing data resulting from somatic deletions and ADOs, as well as the uneven coverage inherent in scDNA-seq. Both DelSIEVE and SCIPhIN deconvolve somatic deletions from ADOs with the help of cell phylogeny. However, unlike SCIPhIN, DelSIEVE explicitly employs a statistical phylogenetic model allowing for both somatic deletions and double mutant genotypes, as well as a model of sequencing coverage using a negative binomial distribution. We have shown that DelSIEVE outperforms SCIPhIN in identifying somatic deletions, including alternative- (1/-) and reference-left single deletion (0/-), as well as in calling single and double mutant genotypes. Furthermore, DelSIEVE is the only method able to explicitly call double deletion genotype.

DelSIEVE and SIEVE are the only two methods being able to explicitly call ADOs, working under either single ADO or locus dropout mode. This task is daunting in a similar sense to calling somatic deletions. We have proved that DelSIEVE outperforms SIEVE regarding calling ADOs. However, the results are only reliable when the data is of adequate coverage quality, which is not given for real data yet. We anticipate that the coverage quality of future scDNA-seq data would be suitable for DelSIEVE to make reliable ADO inference.

Estimating cell phylogeny from scDNA-seq data is a crucial step as it lays the foundation for downstream analyses. Our previous research demonstrated the superiority of SIEVE over other methods, particularly in accurately estimating branch lengths. Building upon the success of SIEVE, our more sophisticated model, DelSIEVE, exhibits comparable performance in the precise estimation of cell phylogeny. Moreover, DelSIEVE surpasses SIEVE's functionality by discerning 17 types of mutation events, corresponding to 28 distinct types of genotype transitions. This expanded capability of mutation event identification makes DelSIEVE a valuable asset in unraveling complex genomic dynamics and understanding evolutionary relationships among cells. We believe that DelSIEVE will greatly benefit researchers in deciphering intricate cellular processes and furthering our understanding of genetic evolution.

For now, DelSIEVE demonstrates its proficiency in identifying somatic deletions, SNVs and ADO. One potential improvement would be to add the identification of small insertions and CNAs with CNs greater than two. Another limitation of DelSIEVE lies in the requirement for preselected input data using DataFilter. This step is limited to identifying candidate variant sites that specifically contain nucleotide substitutions. To address this limitation, a possible enhancement would be to enable DataFilter to preselect sites of tumor suppressor genes that are solely associated with somatic deletions. The inclusion of these sites, which are known to elevate the risk of tumor development, could further refine DelSIEVE's precision and clinical relevance in understanding tumorigenesis and potential therapeutic targets.

Despite these limitations, DelSIEVE proves to be already now one of the most sophisticated statistical phylogenetics models of its kind and extracts an unprecedented wealth of information on evolution of tumors from scDNA-data. We apply DelSIEVE to three real scDNA-seq datasets from TNBC and CRC samples, which were previously analyzed using SIEVE. DelSIEVE identifies rare somatic deletions and double mutant genotypes in the CRC samples, akin to the results of SIEVE. However, for the TNBC sample, DelSIEVE identifies multiple somatic deletions while revealing fewer single and double mutant genotypes compared to SIEVE, consistent with the benchmarking results. Additionally, we demonstrate the higher reliability of somatic deletions called by DelSIEVE than those by Sequenza. These results highlight the precision of DelSIEVE in reconstruction of the phylogenetic tree, as well its enhanced accuracy and effectiveness in identifying genotypes, which holds great potential for advancing our understanding of cancer biology and facilitating precision medicine approaches.

Chapter 4

Summary

This thesis formulates and implements a toolbox of statistical methods that aim to tackle prevailing obstacles in the analysis of scDNA-seq data.

The first statistical method, SIEVE, models raw read counts from scDNA-seq assuming that the sequencing coverage follows a negative binomial distribution, and the read counts of four nucleotides follow a Dirichlet-multinomial distribution. Important technical artifacts of scDNA-seq are considered, including single ADOs. To effectively harness information across individual cells, SIEVE employs a statistical phylogenetic model, with the evolutionary relationship among the cells reflected by the alteration of genotype states, which is modeled by a binary phylogenetic tree. The phylogenetic tree specifically contains a trunk to accommodate clonal mutations shared among all cells. In terms of genotype transitions, SIEVE allows for nucleotide substitutions under the FSA, wherein the number of alleles at a particular site remains constant. This enables the identification of crucial genotypes, such as double mutant.

On top of SIEVE, DelSIEVE further allows for somatic deletions, where the number of alleles reduces when somatic deletions occur. This largely enhances the applicability of DelSIEVE to cancer data, where somatic deletions play an important role. Moreover, DelSIEVE models locus dropout, a possible technical artifact that results in missing data. Both SIEVE and DelSIEVE are able to call ADO states from data of sufficient coverage quality. However, this requirement may not be met by the current scDNA-seq technology. Nevertheless, SIEVE and DelSIEVE holds the promise of accurately identifying ADO states from data of improved coverage quality in the future.

Central to SIEVE and DelSIEVE are the genotype states, acting as the connection between the statistical phylogenetic model and the model of raw read counts. Through the integration of genotype states as hidden random variables, these methods perform variant calling and phylogenetic inference simultaneously. This approach circumvents the potential accumulation of errors that could arise when treating variant calling and phylogenetic inference as separate tasks, ensuring a more accurate and coherent analysis.

From an alternative perspective, SIEVE establishes the foundation for a multifaceted framework adept at simultaneous variant calling and phylogenetic inference, and DelSIEVE emerges as a showcase of the inherent capacity of this framework to be broadened. Notably, this enhancement involves a sequential progression: first, the genotype state space is expanded to encompass somatic deletions; subsequently, the instantaneous transition rate matrix undergoes refinement to account for the expanded genotype state space. Furthermore, the model of nucleotide read counts within the model of raw read count is updated, encompassing the consideration of all possible genotype states. In fact, this process can be generalized for other enhancements, including scenarios like somatic insertions, wherein both small insertions and

CNAs would be taken into account.

It is important to emphasize that both SIEVE and DelSIEVE are associated with two distinct Markov chains. One of them is a continuous-time homogeneous Markov chain, which is defined by the instantaneous rate matrix within the statistical phylogenetic model. The construction of such rate matrices is subject to minimal constraints, requiring that each row sums up to 0. In this context, the genotype state space can give rise to multiple communicating classes, some of which might be transient or closed. Importantly, it's worth noting that this Markov chain might not be ergodic. The other one is a discrete-time homogeneous Markov chain, dynamically generated by the MCMC algorithm during the inference process. This chain must adhere to the property of ergodicity to ensure convergence. It's crucial to recognize that these two Markov chains are independent of each other, serving distinct roles within the framework of SIEVE and DelSIEVE.

Accurate variant calling and phylogenetic inference lay a solid foundation for the subsequent analyses. Further investigation into the called variants may facilitate the identification of novel oncogenes and tumor suppressor genes, as well as the driver and passenger mutations [14]. Moreover, one of the most promising applications of the cell phylogeny is to understand the evolutionary history of metastases [181]. In such cases, the input data typically contains individual cells sampled from multiple tumors at distinct anatomical sites within the same patient. Cell phylogeny inferred from such data may shed a light on mutations potentially associated with metastases and implicitly the starting points of metastases, which may be patient specific [182, 183, 143]. In particular, SIEVE and DelSIEVE yield cell phylogeny with branch lengths measured by the number of accumulated mutations per site, which comes convenient in the downstream analyses. Furthermore, novel statistical models could be proposed on top of SIEVE and DelSIEVE to explicitly model the evolutionary events related to metastases, namely the migrations among various anatomical sites.

BEAST 2 stands as a remarkably versatile and highly extensible framework dedicated to phylogenetic inference [101]. Both SIEVE and DelSIEVE have been implemented as packages within the BEAST2 ecosystem, thus enabling the efficient reconstruction of cell phylogenies from scDNA-seq. A plethora of packages for phylogenetic inference have been developed for BEAST 2, among which we have employed an optimized relaxed clock model [156] coupled with our models to account for branch-wise rate variation. This synergy underscores the immense potential for collaboration between our models and other packages within the BEAST 2 environment.

In summary, both SIEVE and DelSIEVE could further be expanded independently by additional extensions such as incorporation of insertions, application to metastasis analysis, or by exploring the supplementary packages of BEAST2. The already achieved functionality of our models and the large potential of their further extensions promise to unlock novel insights and catalyze innovative discoveries within the realm of single-cell genomics and phylogenetic reconstruction in the future.

Chapter 5

Supplementary material

5.1. Supplementary methods

5.1.1. Data description

CRC28

We isolated EpCAM+ cells from one normal and three tumoral regions (TP: tumor proximal; TC: tumor central; TD: tumor distal) from the patient with a BD FACSAria III cytometer. We successfully amplified the genomes of 28 tumor cells and 18 normal cells with Ampli1 (Silicon Biosystems) and built whole-genome sequencing libraries using the KAPA (Kapa Biosystems) library kit. Each library was sequenced at ≈6x on an Illumina Novaseq 6000 at the Spanish National Center of Genomic Analysis (CNAG-CR; `https://www.cnag.crg.eu/`) [1].

TNBC16

Extracted were 16 single tumor nuclei and 16 single normal nuclei for whole-exome sequencing. For each individual sorted nuclei, multiple-displacement-amplification was performed using the REPLI-G UltraFast Mini Kit (Qiagen, #150035). The protocol was modified by heating the lysed DNA at $65°C$ for 10 min and incubating the DNA with the Φ29 polymerase at $30°C$ for 80 min [2].

CRC48

96 single cells were isolated for whole exome sequencing, where the DNA materials were amplified with Qiagen REPLI-g Single Cell Kit. The protocol was modified by incubating the DNA with Φ29 polymerase at $30°C$ for exactly 120 min [3].

5.1.2. Data preprocessing

For the public TNBC16 [2] and CRC48 [3] datasets, we downloaded the raw sequencing reads from the SRA database in FASTQ format. For the three datasets (CRC28, TNBC16 and CRC48) We trimmed the Illumina adapter sequences using cutadapt (version 1.18) and mapped reads to the 1000G Reference Genome hs37d5 using BWA MEM (version 0.7.17). After de-duplication with Picard (version 2.18.14), we used GATK (version 3.7.0) for local realignment based on indel calls from the 1000G Phase 1 and the Mills and 1000G gold standard. Subsequently, we recalibrated the base scores using GATK (version 4.0.10) with polymorphisms from dbSNP (build 138) and indels from the 1000G Phase 1.

5.1.3. Code availability

All code in this thesis are freely accessible under a GNU General Public License v3.0 license.

Code used in Chapter 2

SIEVE is implemented in Java and is accessible at https://github.com/szczurek-lab/SIEVE. DataFilter for selecting candidate variant sites is available at https://github.com/szczurek-lab/DataFilter. The simulator is hosted at https://github.com/szczurek-lab/SIEVE_simulator, and the reproducible benchmarking framework is available at https://github.com/szczurek-lab/SIEVE_benchmark_pipeline. The scripts for generating all figures in this paper are hosted at https://github.com/szczurek-lab/SIEVE_analysis.

Code used in Chapter 3

DelSIEVE is implemented in Java and is accessible at https://github.com/szczurek-lab/DelSIEVE. The simulator is hosted at https://github.com/szczurek-lab/DelSIEVE_simulator. The reproducible benchmarking framework is available at https://github.com/szczurek-lab/DelSIEVE_benchmark_pipeline, and the scripts for generating all figures in this paper are hosted at https://github.com/szczurek-lab/DelSIEVE_analysis.

5.2. Supplementary figures of Chapter 2

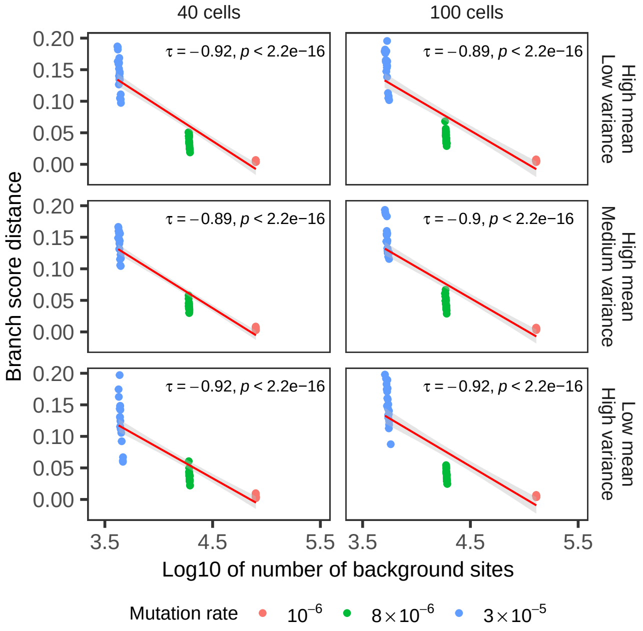


Figure 5.1: **Correlation plot of the BS distance against the number of background sites in log10 scale.** Varying are the number of cells and the coverage quality. BS distance data points are colored by the corresponding mutation rates. τ indicates the Kendall's correlation coefficient, which is invariant to the log transformation of the number of background sites. We choose 0.01 as the significance threshold.

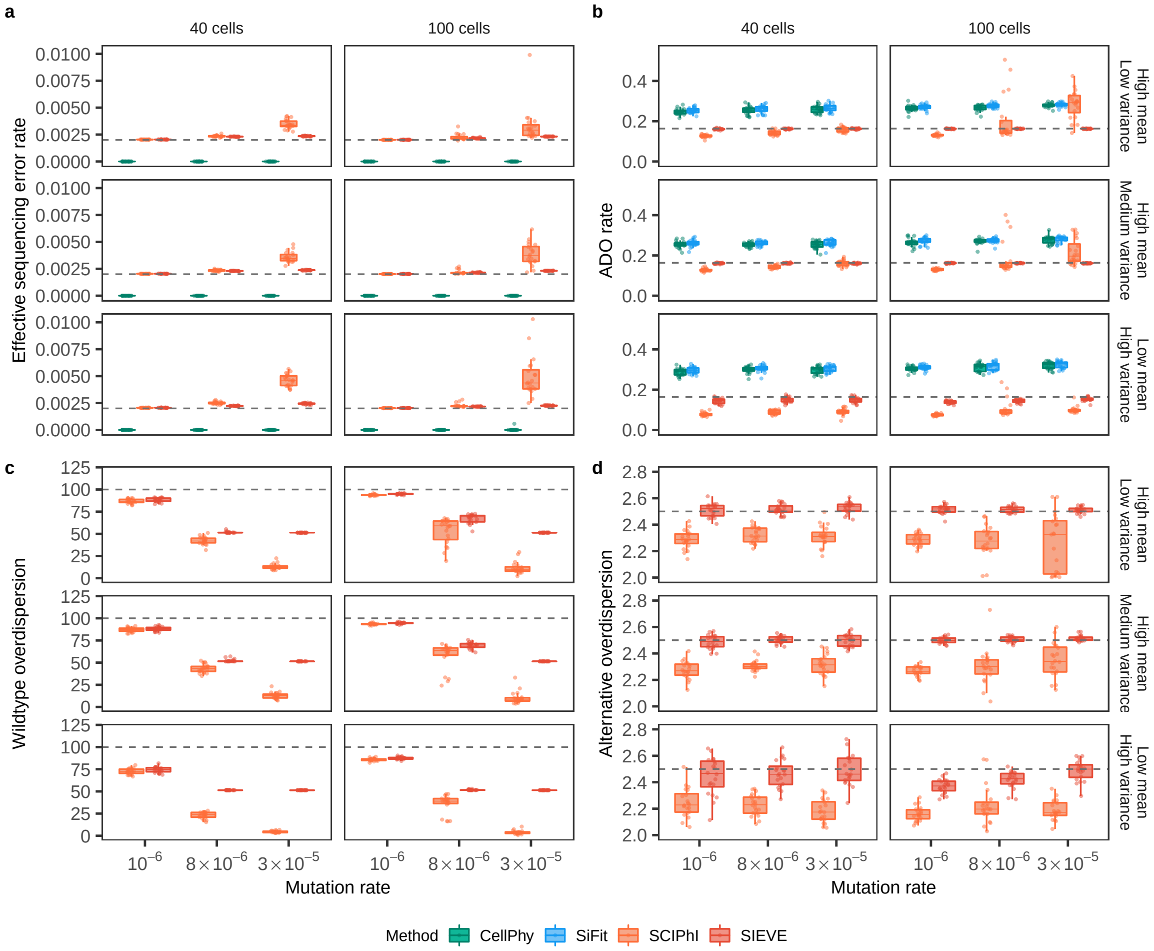


Figure 5.2: **Additional benchmarking results of the SIEVE model regarding parameter estimates.** Each simulation is repeated $n = 20$ times with each repetition denoted by colored dots. The gray dashed lines represent the ground truth used to generate the simulated data. **a-d**, Box plots of parameter estimation accuracy for four important parameters in the model of raw read counts (see Section SIEVE model): effective sequencing error rate (**a**), ADO rate (**b**), wildtype overdispersion (**c**) and alternative overdispersion (**d**).

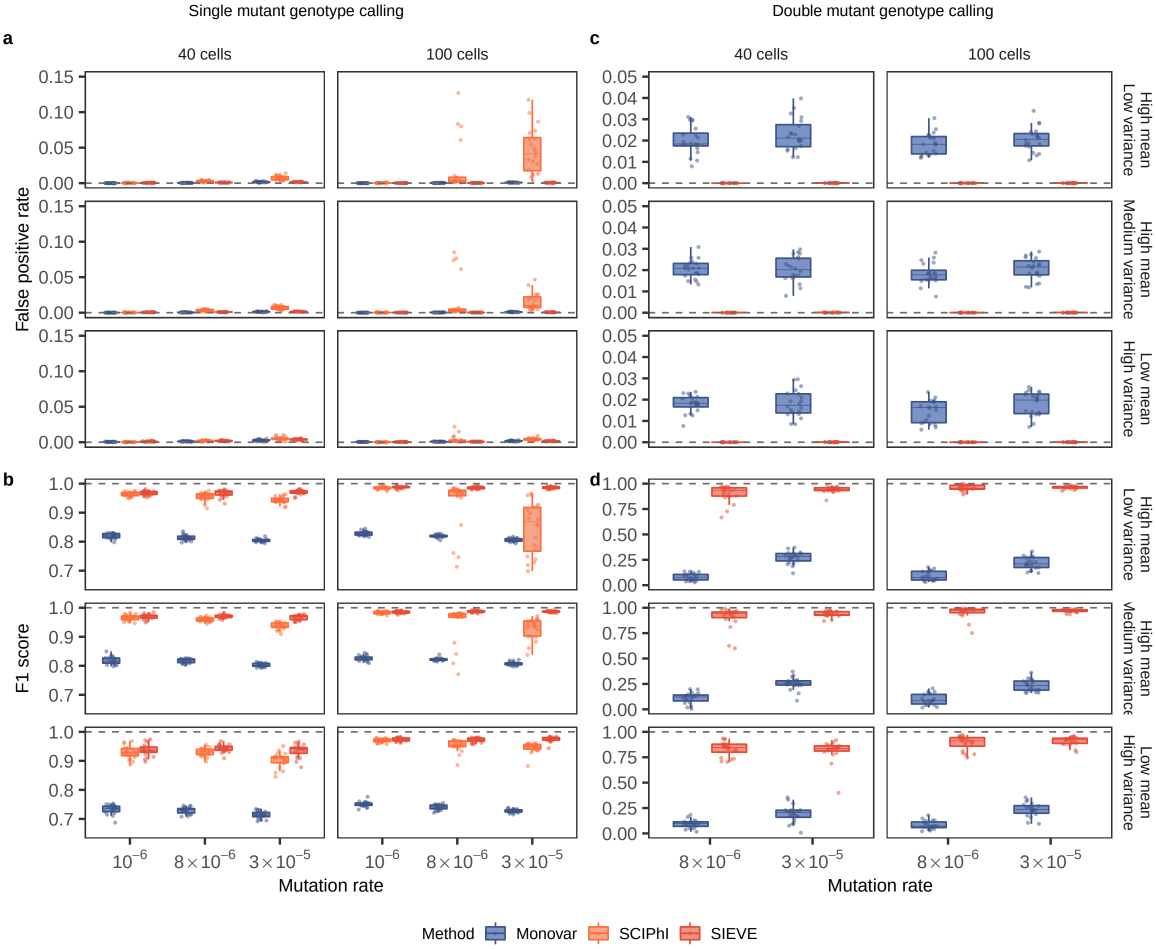


Figure 5.3: **Additional benchmarking results of the SIEVE model regarding variant calling.** Each simulation is repeated $n = 20$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. **a-b**, Box plots of the single mutant genotype calling results measured further by the fraction of false positives in the ground truth negatives, i.e., the sum of false positives and true negatives, (false positive rate, **a**) and the harmonic mean of recall and precision (F1 score, **b**). **c-d**, Box plots of the double mutant genotype calling results measured further by false positive rate (**c**) and F1 score (**d**), where the variant calling results when mutation rate is 10^{-6} are omitted as very few double mutant genotypes are generated (less than 0.1%).

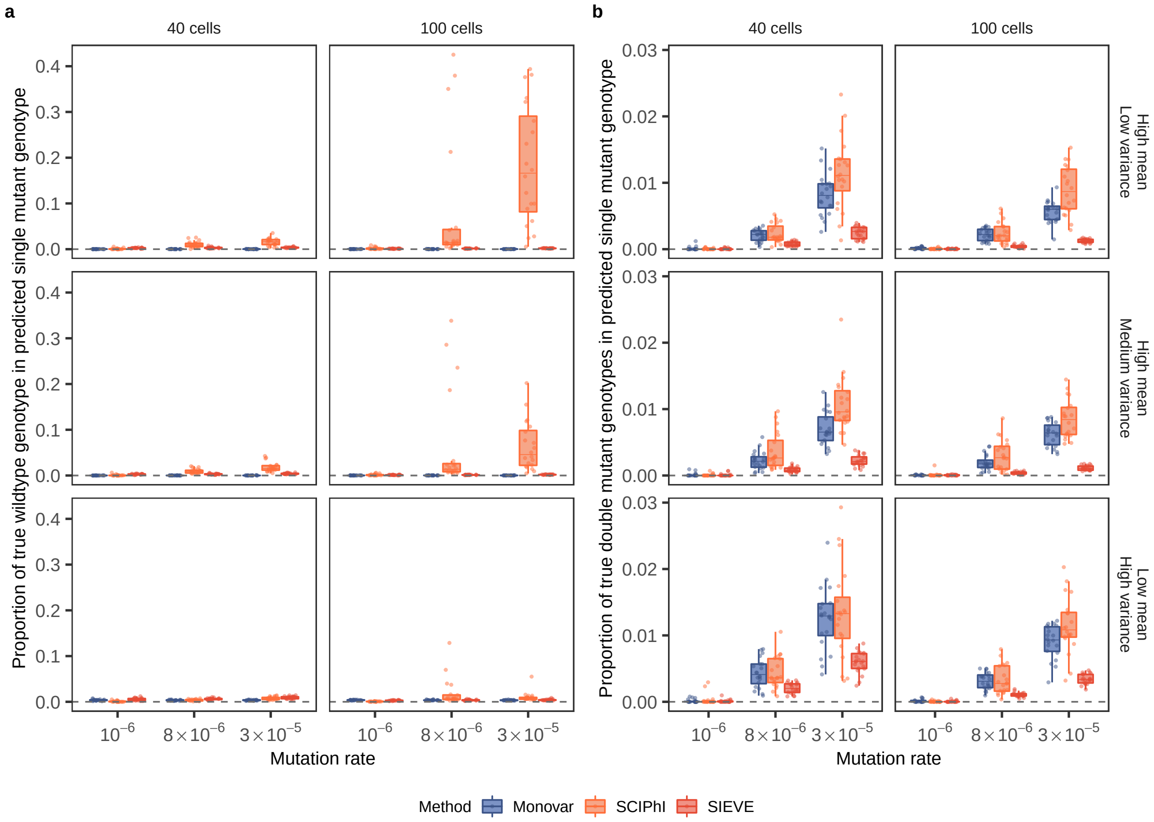


Figure 5.4: **Types of false positives in single mutant genotype calling.** The gray dashed lines represent the optimal proportions of each type. **a-b**, Box plots of the types of false positives in single mutant genotype calling, including the proportion of true wildtype (**a**) and true double mutant genotypes (**b**). For single mutant genotype calling, the sum of the precision, the proportion of true wildtype and the proportion of true double mutant genotypes is 1.

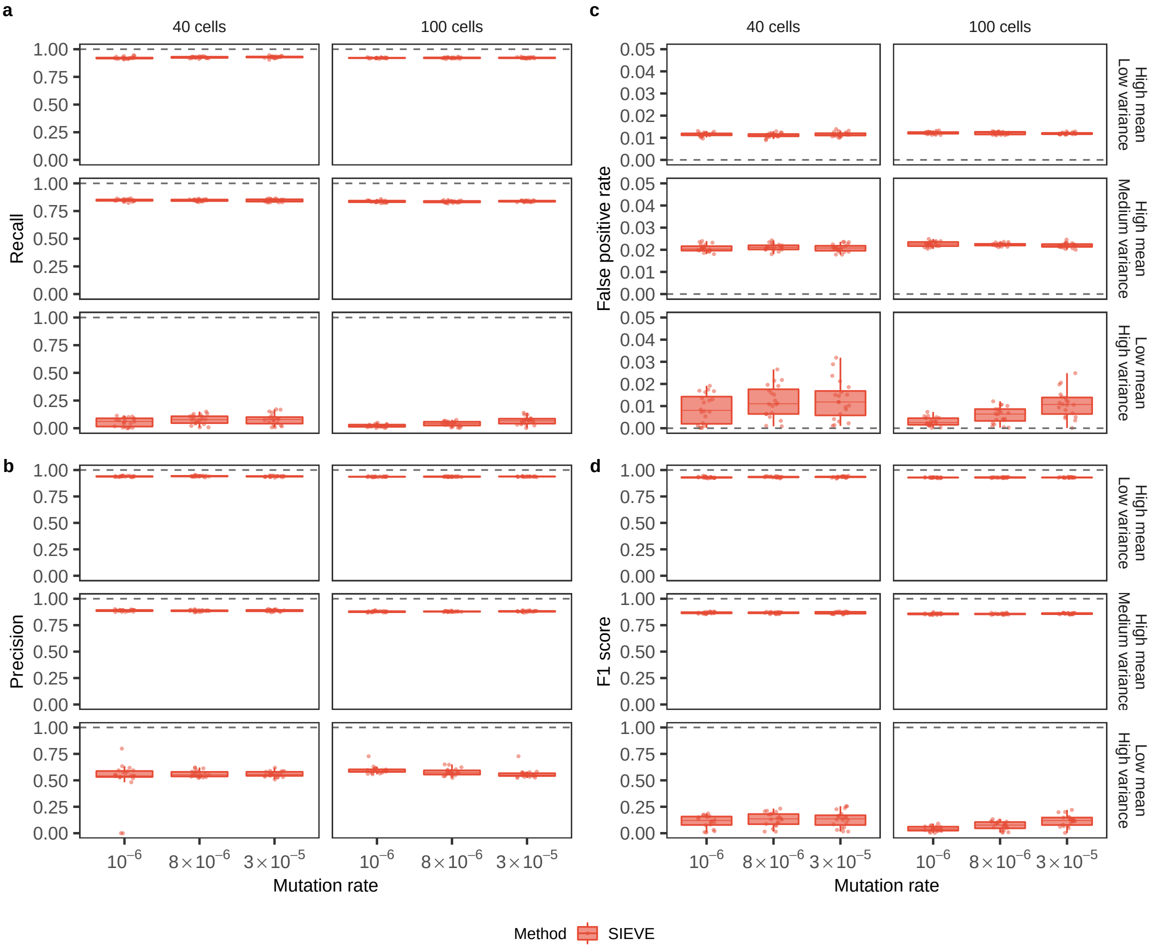


Figure 5.5: **Benchmarking results of the SIEVE model regarding ADO calling.** Each simulation is repeated $n = 20$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. **a-d**, Box plots of the ADO calling results measured in recall (**a**), precision (**b**), false positive rate (**c**) and F1 score (**d**).

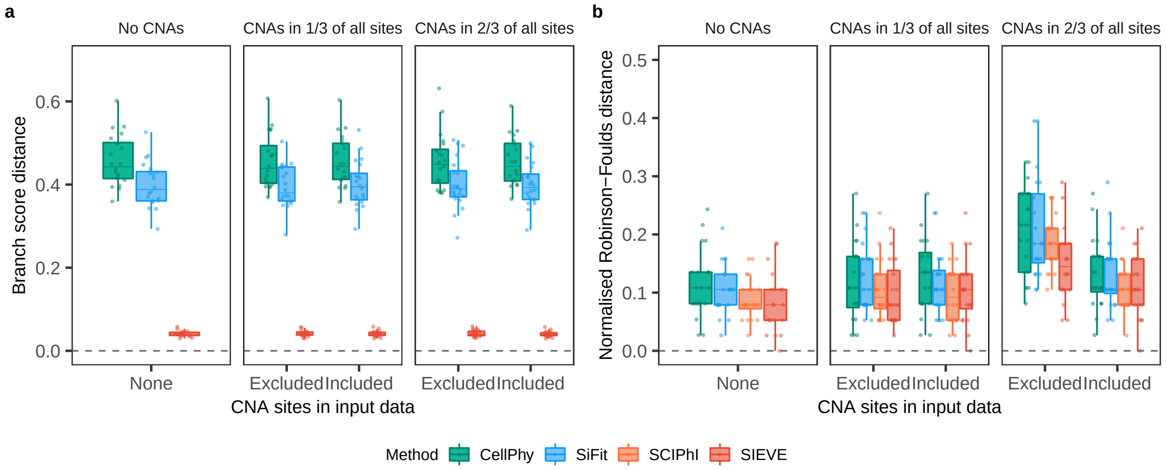


Figure 5.6: **Tree distance benchmarking results of the SIEVE model considering CNAs.** Varying are the prevalence of CNAs in all genomic sites and whether these CNA sites are included or not in the input data. Each simulation is repeated $n = 20$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. **a-b**, Box plots of the tree inference accuracy measured by the BS distance where the branch lengths are taken into account (**a**) and the normalised RF distance where only tree topology is considered (**b**).

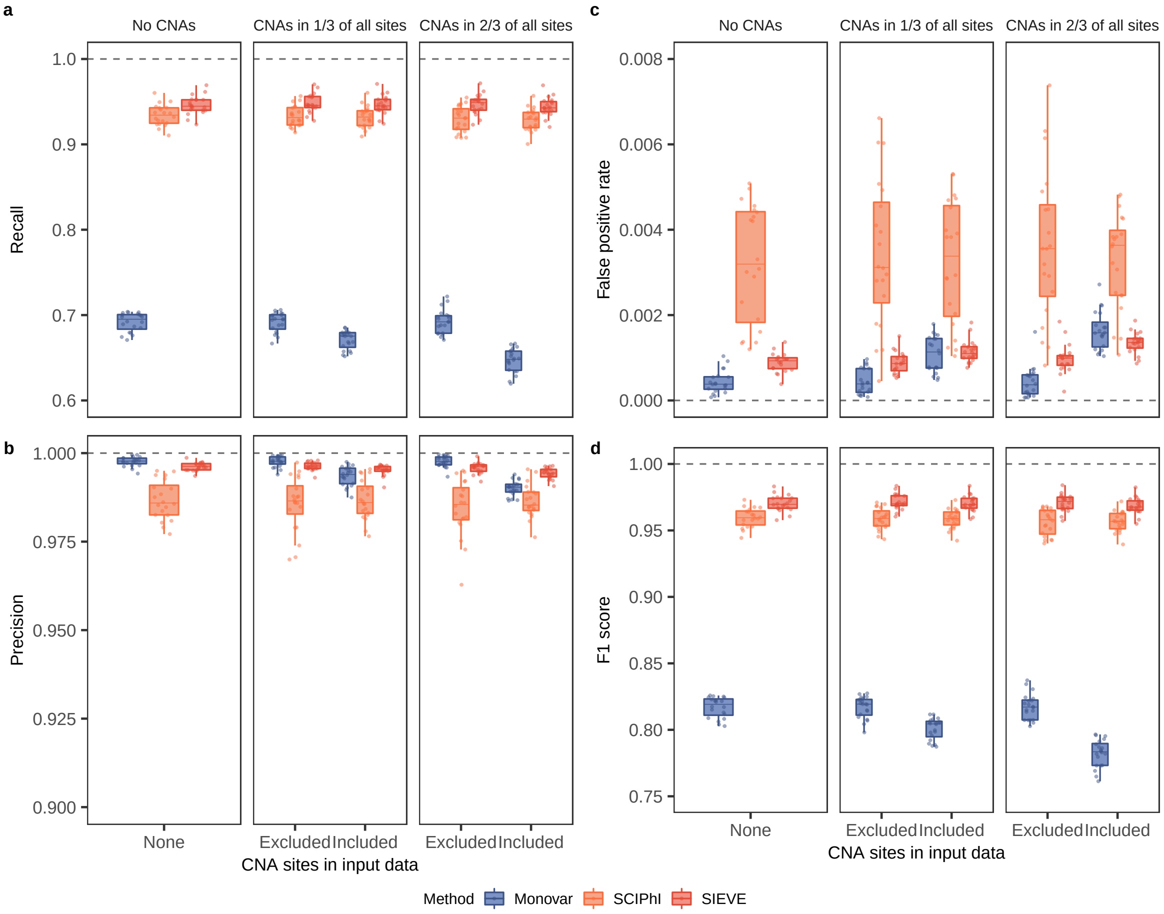


Figure 5.7: **Single mutant genotype calling results of the SIEVE model considering CNAs.** Varying are the prevalence of CNAs in all genomic sites and whether these CNA sites are included or not in the input data. Each simulation is repeated $n = 20$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. **a-d**, Box plots of the single mutant genotype calling results measured by recall (**a**), precision (**b**), false positive rate (**c**) and F1 score (**d**).

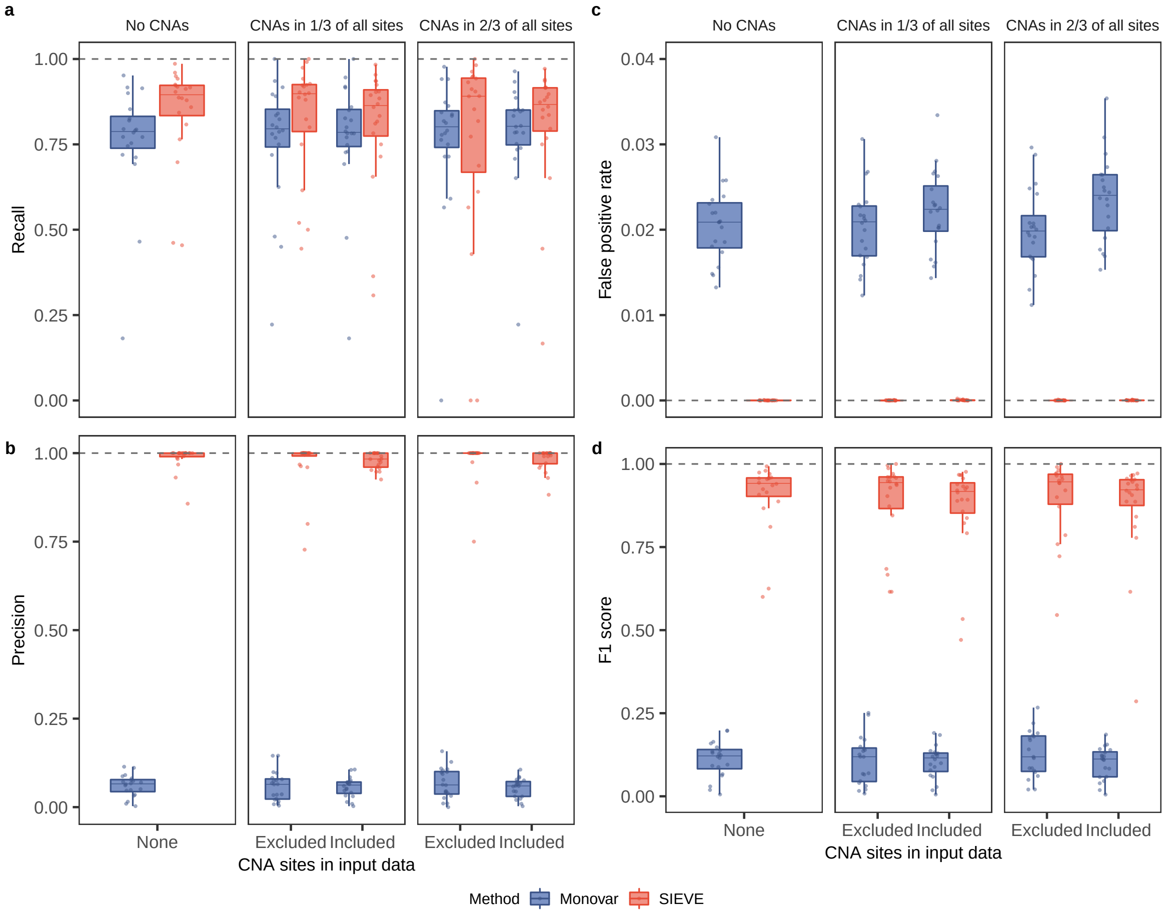


Figure 5.8: **Double mutant genotype calling results of the SIEVE model considering CNAs.** Varying are the prevalence of CNAs in all genomic sites and whether these CNA sites are included or not in the input data. Each simulation is repeated $n = 20$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. **a-d**, Box plots of the double mutant genotype calling results measured by recall (**a**), precision (**b**), false positive rate (**c**) and F1 score (**d**).

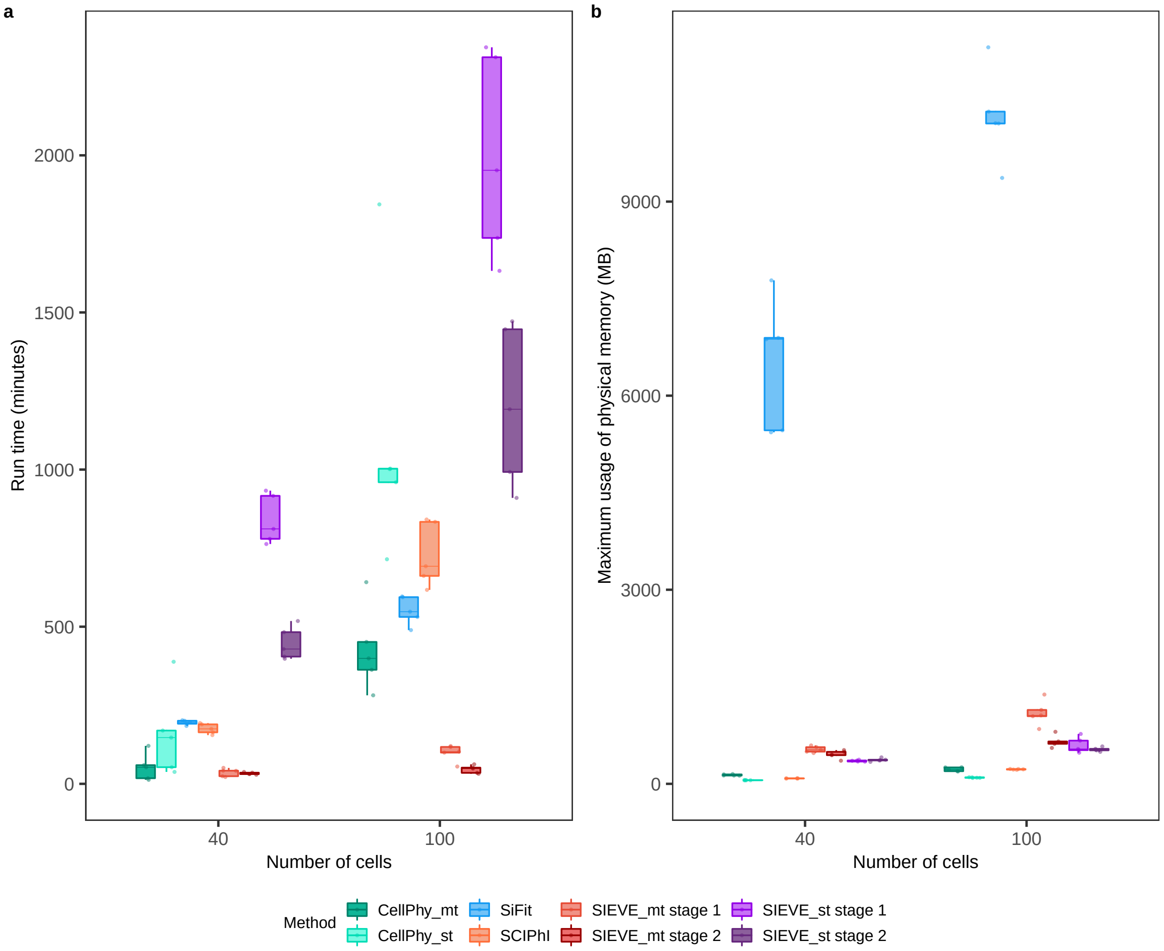


Figure 5.9: **Run time and memory usage evaluation.** Varying is the number of cells. Each simulation is repeated $n = 5$ times with each repetition denoted by colored dots. SiFit and SCIPhI were run under single-thread mode, while CellPhy and the two stages of SIEVE were run under both single- (CellPhy_st, SIEVE_st stage 1 and SIEVE_st stage 2) and multi-thread (CellPhy_mt and SIEVE_mt stage 1 and SIEVE_mt stage 2) mode. **a-b**, Box plots of efficiency benchmarking results of SIEVE with respect to run time in minutes (**a**) and maximum usage of physical memory in MB (**b**).

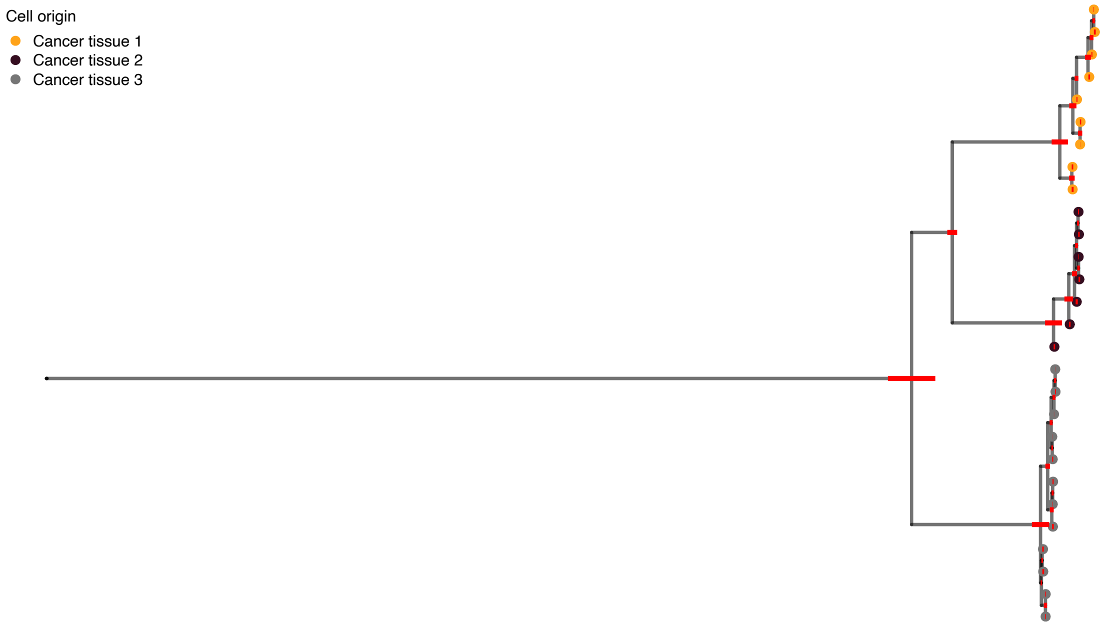


Figure 5.10: **Illustration of branch lengths of the phylogenetic tree inferred from CRC28 [1] by SIEVE.** Shown is exactly the same tree as in Fig. 3, except that cell names, subclone posterior probabilities and gene annotations are removed and no branches are folded. Red bars annotated to internal nodes except the root are the 95% highest posterior density (HPD) intervals of the corresponding branch lengths.

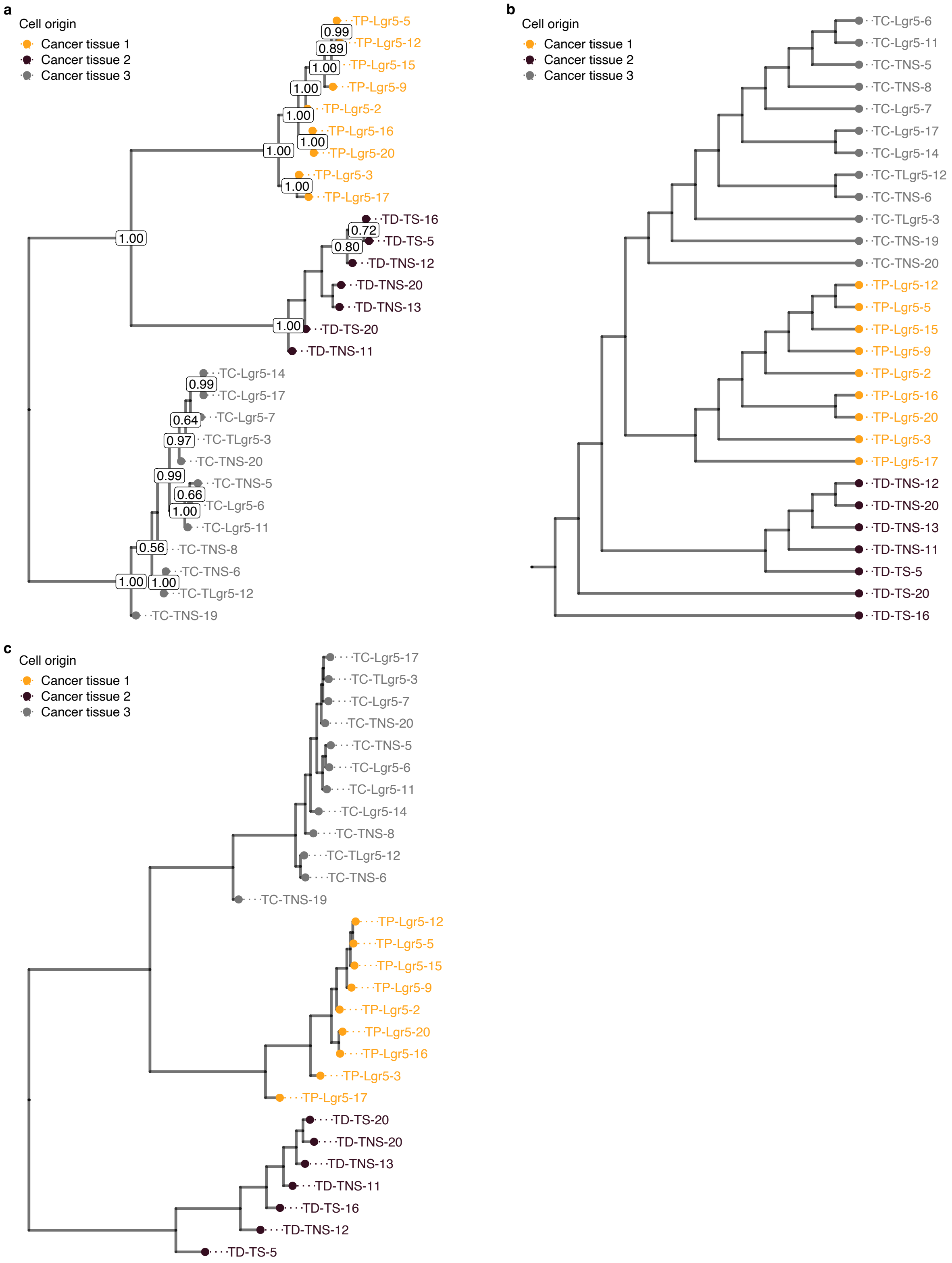


Figure 5.11: **Illustration of cell phylogenies inferred from CRC28 [1] by other methods. a-c**, Trees inferred by CellPhy (**a**), SCIPhI (**b**) and SiFit (**c**). CellPhy was run with bootstrap applied, thereby making node supports available. SCIPhI reported only tree topology, not branch lengths.

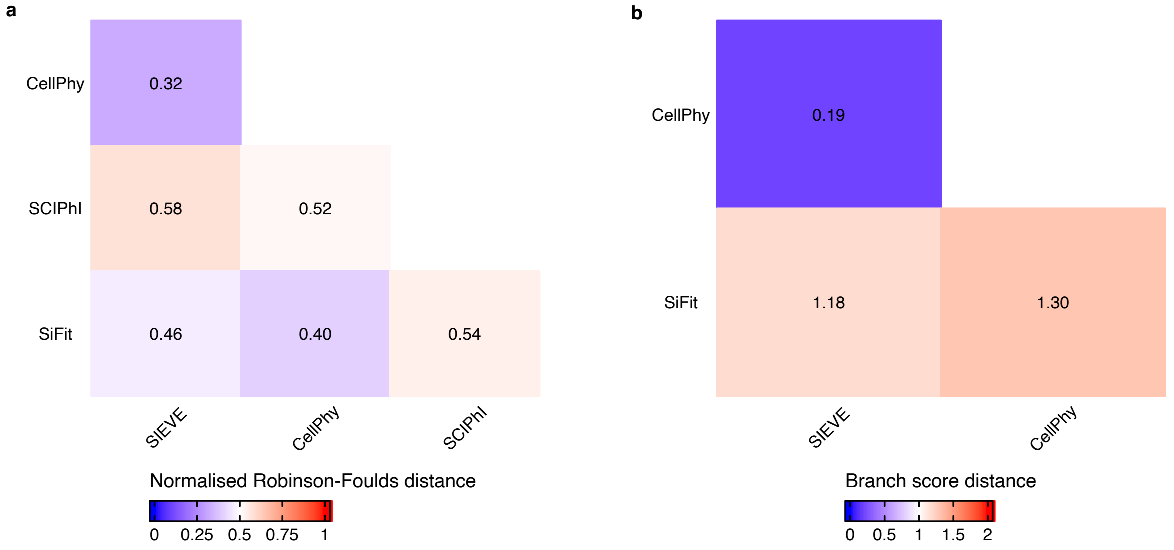


Figure 5.12: **Heatmaps for pairwise distances of phylogenetic trees inferred from CRC28 [1] by all methods. a-b**, Tree distances measured by normalised RF distance (**a**) and BS distance (**b**).

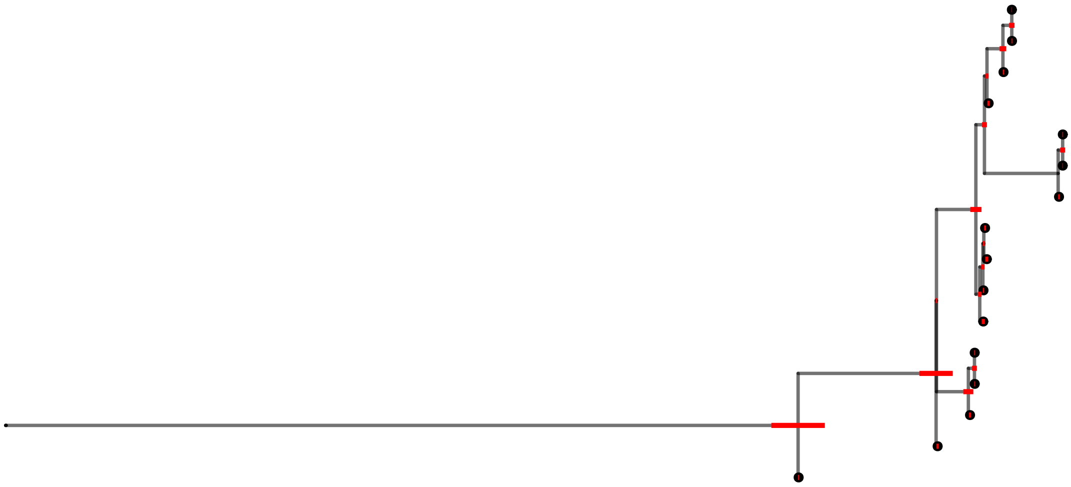


Figure 5.13: **Illustration of branch lengths of the phylogenetic tree inferred from TNBC16 [2] by SIEVE.** Shown is exactly the same tree as in Fig. 4, except that cell names, subclone posterior probabilities and gene annotations are removed and no branches are folded. Red bars annotated to internal nodes except the root are the 95% HPD intervals of the corresponding branch lengths.

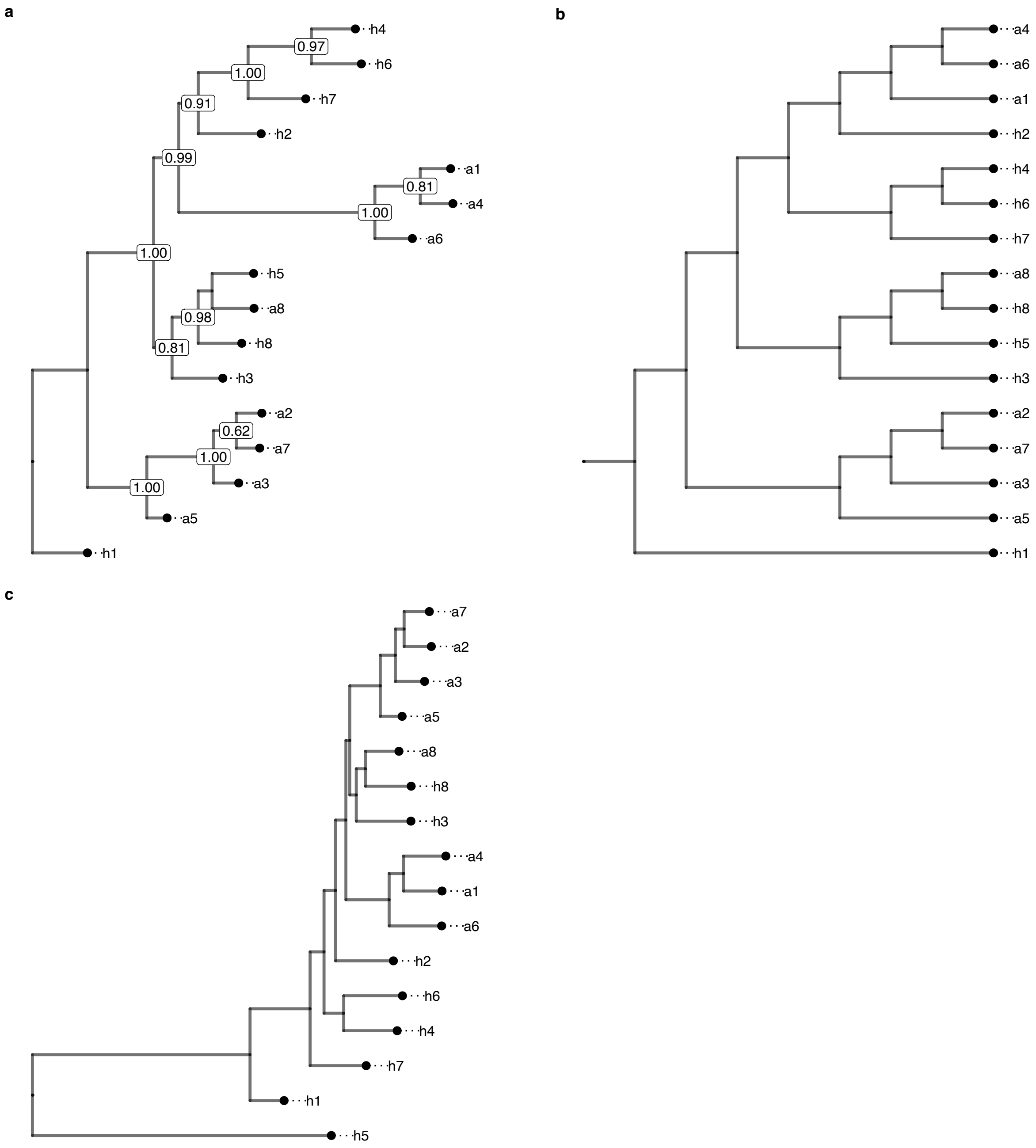


Figure 5.14: **Illustration of cell phylogenies inferred from TNBC16 [2] by other methods.** **a-c**, Trees inferred by CellPhy (**a**), SCIPhI (**b**) and SiFit (**c**). CellPhy was run with bootstrap applied, thereby making node supports available. SCIPhI reported only tree topology, not branch lengths.

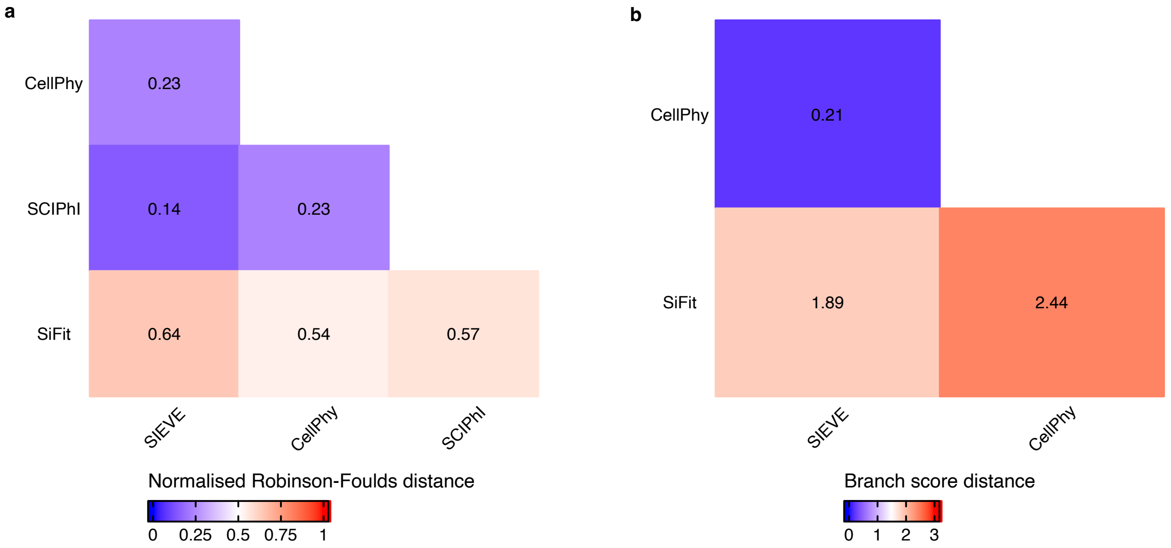


Figure 5.15: **Heatmaps for pairwise distances of phylogenetic trees inferred from TNBC16 [2] by all methods. a-b**, Tree distances measured by normalised RF distance (**a**) and BS distance (**b**).

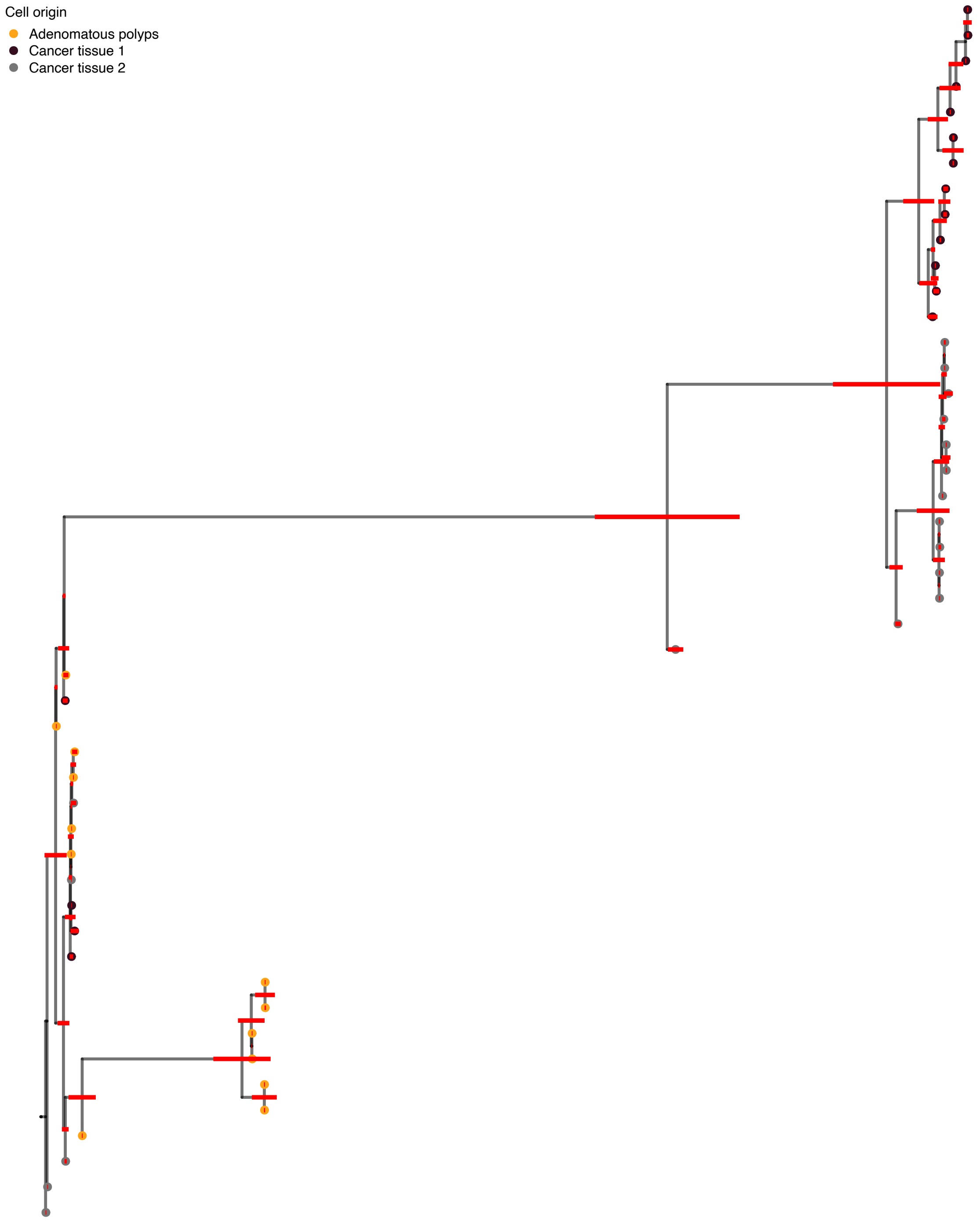


Figure 5.16: **Illustration of branch lengths of the phylogenetic tree inferred from CRC48 [3] by SIEVE.** Shown is exactly the same tree as in Figure 2.5, except that cell names, subclone posterior probabilities and gene annotations are removed and no branches are folded. Red bars annotated to internal nodes except the root are the 95% HPD intervals of the corresponding branch lengths.

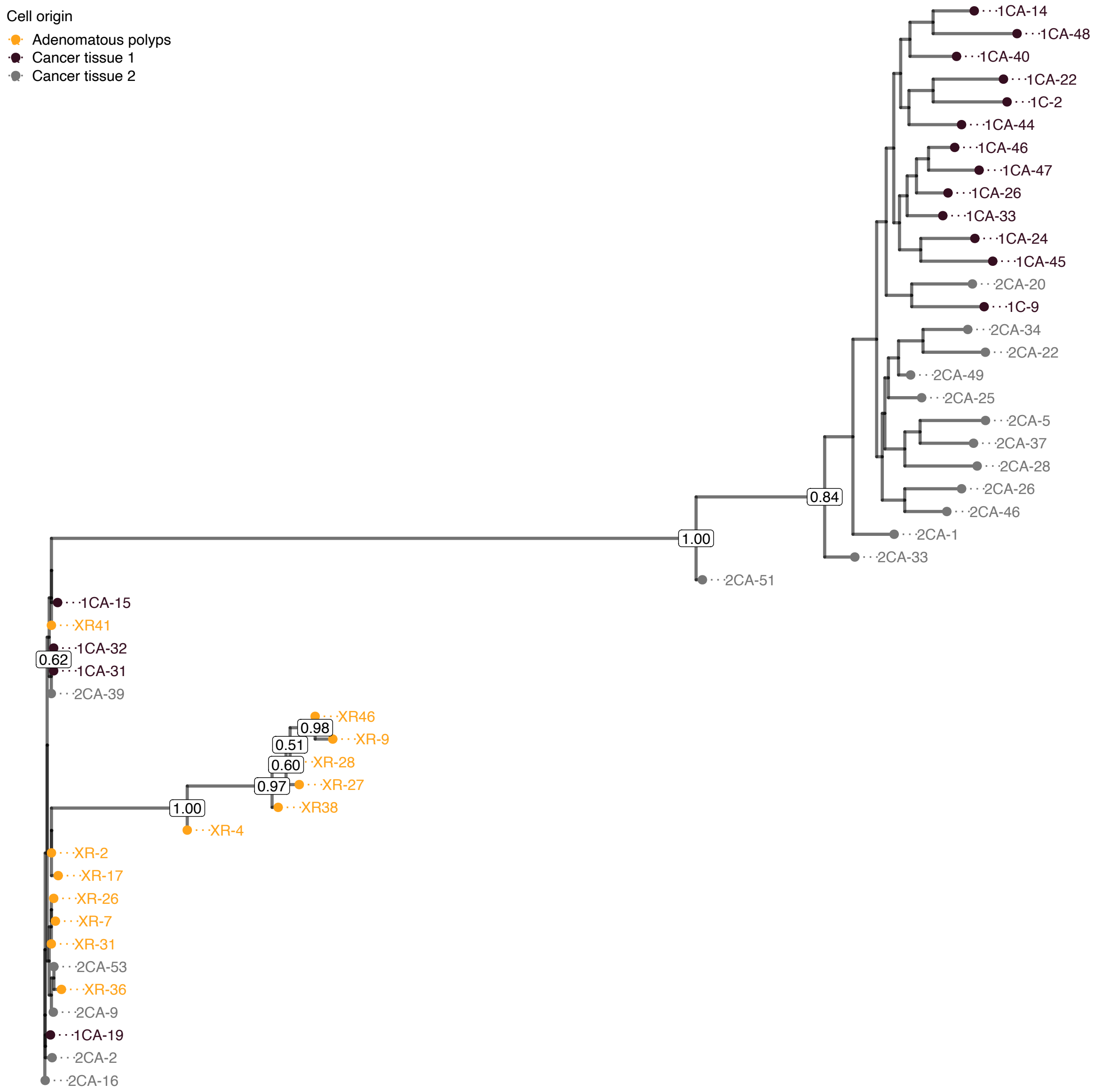


Figure 5.17: **Illustration of the cell phylogeny inferred from CRC48 [3] by CellPhy.** CellPhy was run with bootstrap applied, thereby making node supports available.

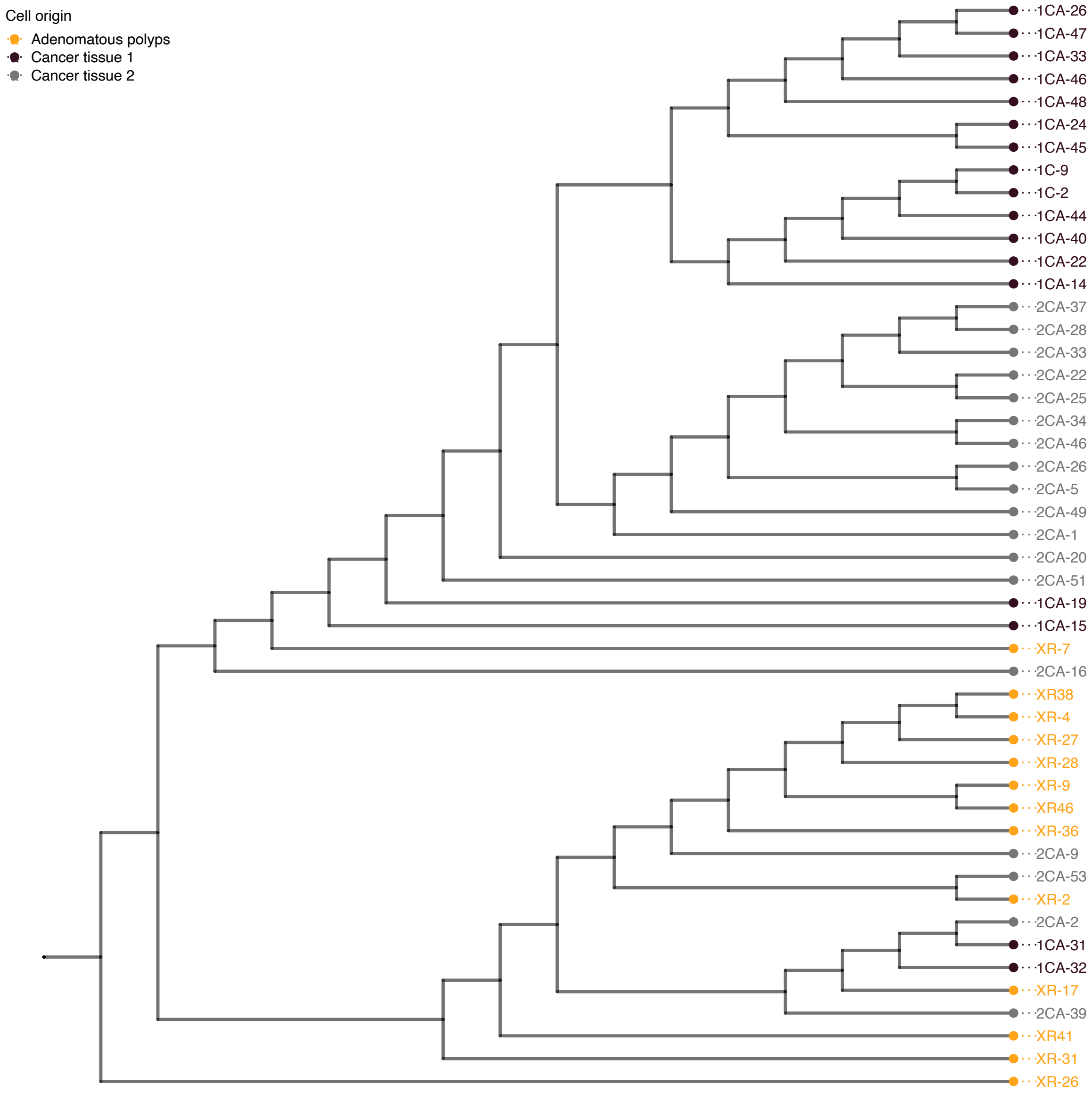


Figure 5.18: **Illustration of the cell phylogeny inferred from CRC48 [3] by SCIPhI.** SCIPhI reported only tree topology, not branch lengths.

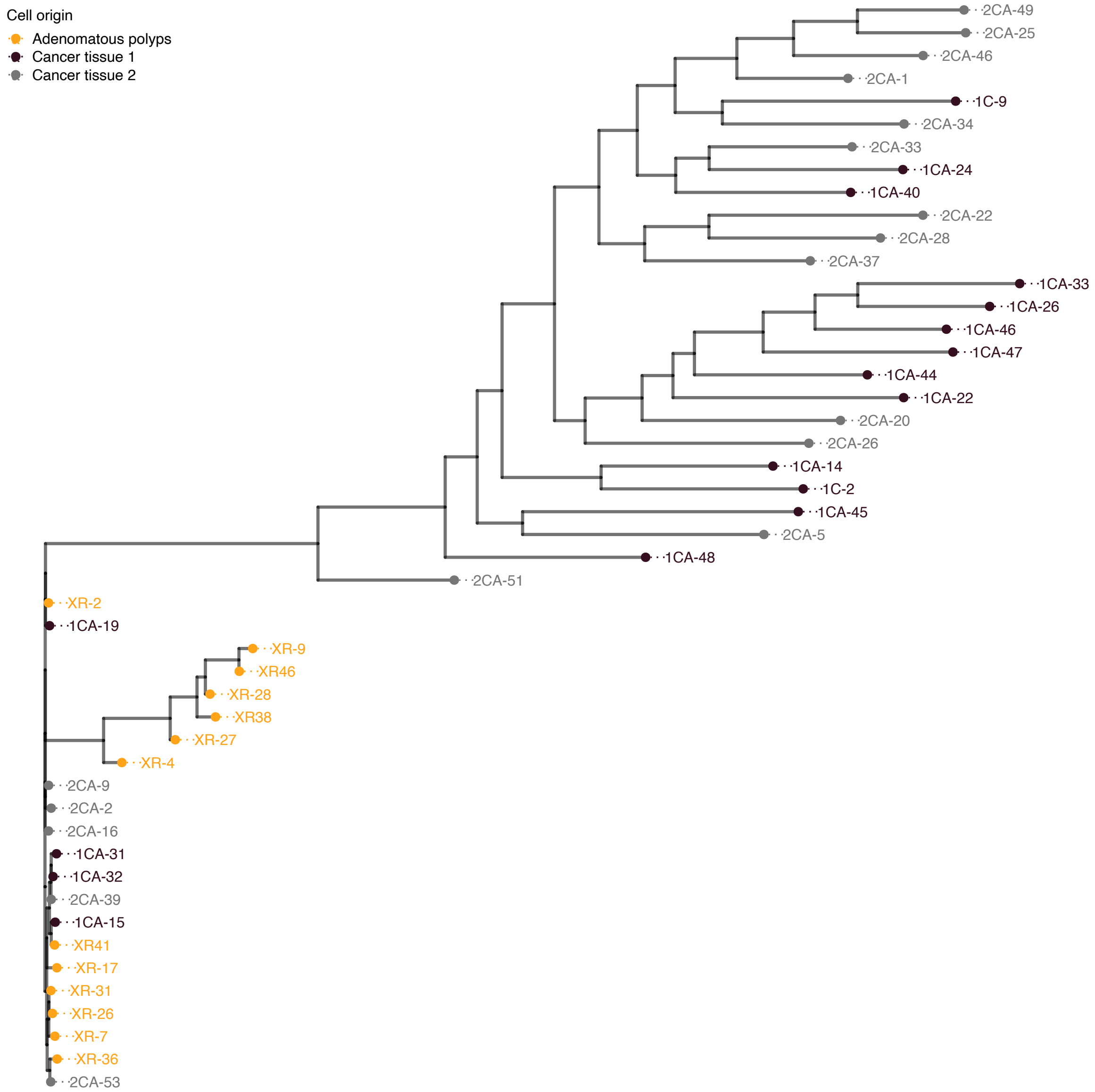


Figure 5.19: **Illustration of the cell phylogeny inferred from CRC48 [3] by SiFit.**

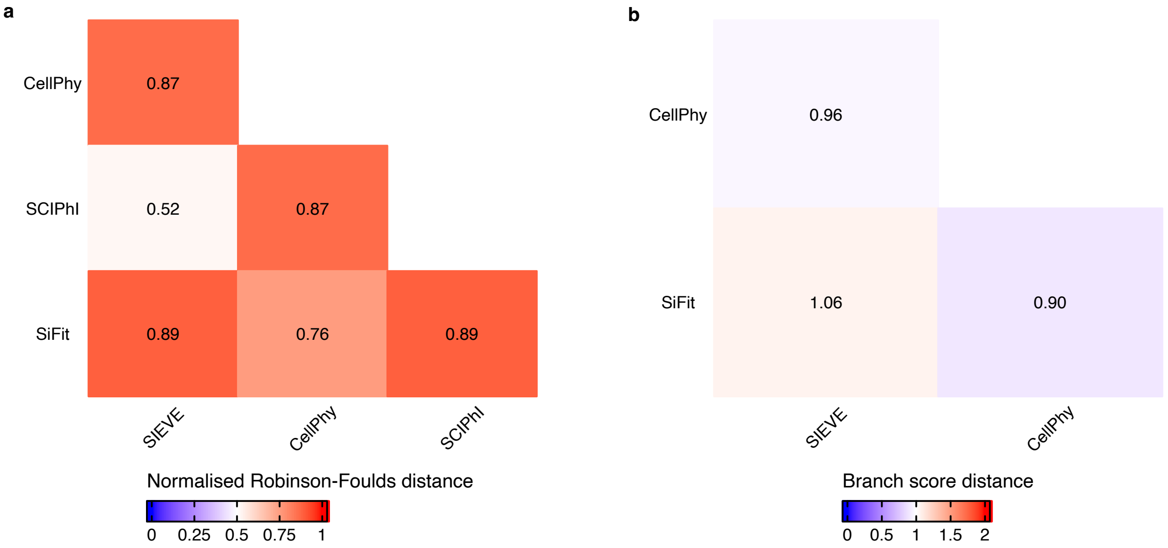


Figure 5.20: **Heatmaps for pairwise distances of phylogenetic trees inferred from CRC48 [3] by all methods. a-b**, Tree distances measured by normalised RF distance (**a**) and BS distance (**b**).

5.3. Supplementary tables of Chapter 2

Table 5.1: **Evolutionary rate matrix used in the simulator to generate the simulated data for SIEVE.** Genotypes are encoded with nucleotides rather than numbers. The diagonal elements are denoted by dots, and have negative values equal to the sum of the other entries in the same row, ensuring that the sum of each row equals zero.

	A/A	A/C	A/G	A/T	C/C	C/G	C/T	G/G	G/T	T/T
A/A	.	$^1/_3$	$^1/_3$	$^1/_3$	0	0	0	0	0	0
A/C	$^1/_6$	.	$^1/_6$	$^1/_6$	$^1/_6$	$^1/_6$	$^1/_6$	0	0	0
A/G	$^1/_6$	$^1/_6$	.	$^1/_6$	0	$^1/_6$	0	$^1/_6$	$^1/_6$	0
A/T	$^1/_6$	$^1/_6$	$^1/_6$	.	0	0	$^1/_6$	0	$^1/_6$	$^1/_6$
C/C	0	$^1/_3$	0	0	.	$^1/_3$	$^1/_3$	0	0	0
C/G	0	$^1/_6$	$^1/_6$	0	$^1/_6$	.	$^1/_6$	$^1/_6$	$^1/_6$	0
C/T	0	$^1/_6$	0	$^1/_6$	$^1/_6$	$^1/_6$	.	0	$^1/_6$	$^1/_6$
G/G	0	0	$^1/_3$	0	0	$^1/_3$	0	.	$^1/_3$	0
G/T	0	0	$^1/_6$	$^1/_6$	0	$^1/_6$	$^1/_6$	$^1/_6$	.	$^1/_6$
T/T	0	0	0	$^1/_3$	0	0	$^1/_3$	0	$^1/_3$	.

Table 5.2: **Inferred mean and variance of allelic coverage from SIEVE for real datasets.**

	Mean of allelic coverage t	Variance of allelic coverage v
CRC28	4.3	19.6
TNBC16	10.2	207.9
CRC48	19.4	635.6

Table 5.3: **The number of (candidate) variant sites and the corresponding number of threads used for CellPhy and SIEVE in the run-time benchmarking.** Different number of threads was chosen to achieve the highest efficiency for each method. In particular, when CellPhy was given as many threads as provided to SIEVE, its run time increased.

	No. of cells	No. of (candidate) variant sites	No. of threads
CellPhy	40	774 - 975	4 - 5
	100	932 - 1422	5 - 7
SIEVE	40	786 - 987	31 - 39
	100	961 - 1482	38 - 52

Table 5.4: **Summary of fractions of predicted genotypes by SIEVE and Monovar for three analyzed real datasets.** Entries marked with NA denote that the corresponding method does not call the specific genotype.

		Missing	0/0	0/1	1/1	$1/1'$
CRC28	SIEVE	NA	25.02%	74.64%	0.28%	0.06%
	Monovar	10.40%	38.09%	46.30%	5.21%	NA
TNBC16	SIEVE	NA	15.54%	75.11%	9.30%	0.05%
	Monovar	10.63%	32.92%	41.58%	14.87%	NA
CRC48	SIEVE	NA	59.48%	40.50%	0.02%	0
	Monovar	4.53%	69.41%	24.13%	1.93%	NA

5.4. Supplementary figures of Chapter 3

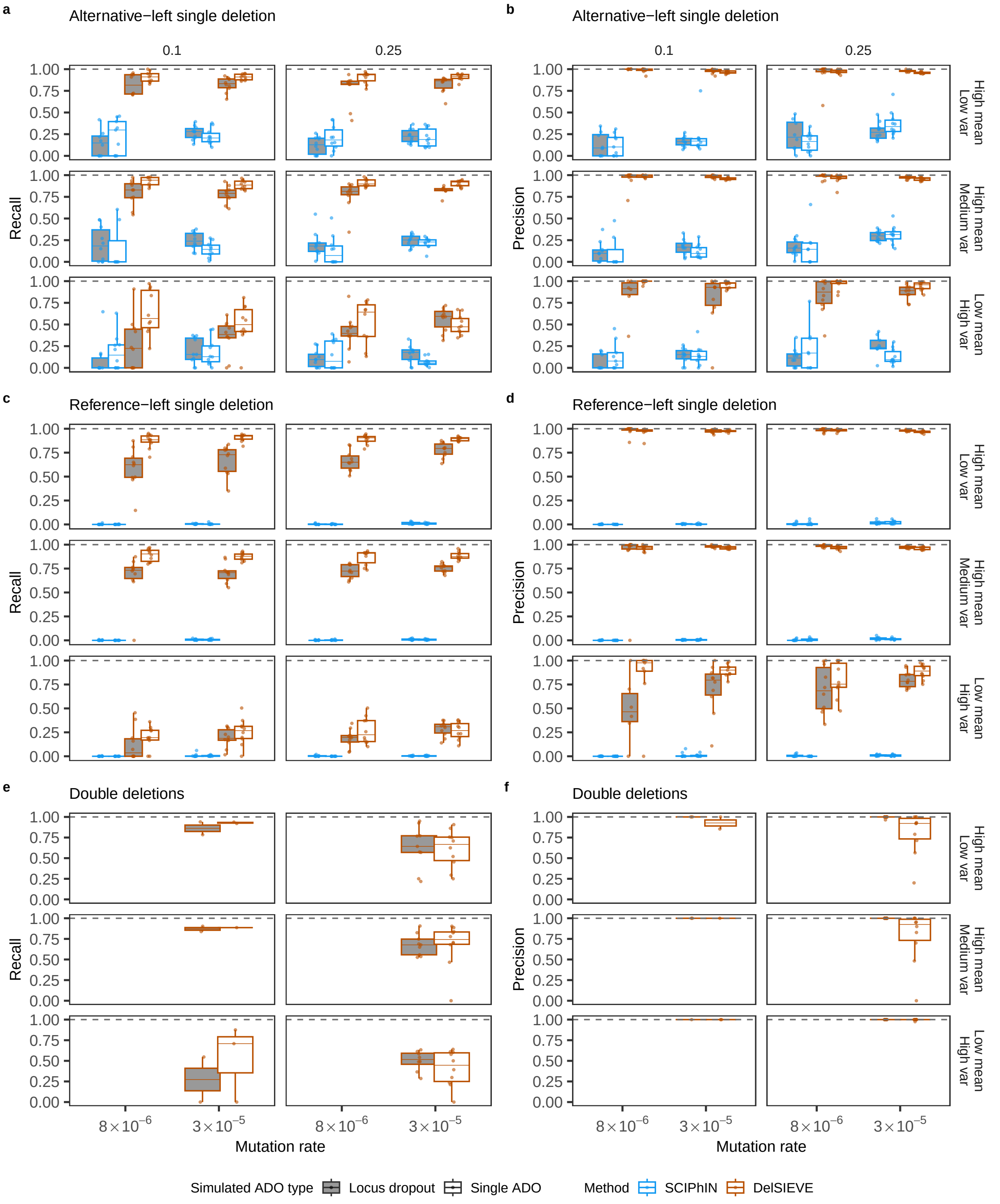


Figure 5.21: **Recall and precision for the benchmark of calling somatic deletions.** Varying are the mutation rate (the horizontal axis), the relative deletion rate (the vertical strip), the coverage quality (the horizontal strip) and the simulated ADO type (the shaded or blank boxes). Each simulation is repeated $n = 10$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. Data points were removed if the proportion of simulated ground truth was less than 0.1%. **a-b**, Box plots of the recall (**a**) and the precision (**b**) for calling alternative-left single deletion. **c-d**, Box plots of the recall (**c**) and the precision (**d**) for calling reference-left single deletion. **e-f**, Box plots of the recall (**e**) and the precision (**f**) for calling double deletion, where the results when mutation rate was 8×10^{-6} were omitted as very few double deletion were generated (less than 0.2%).

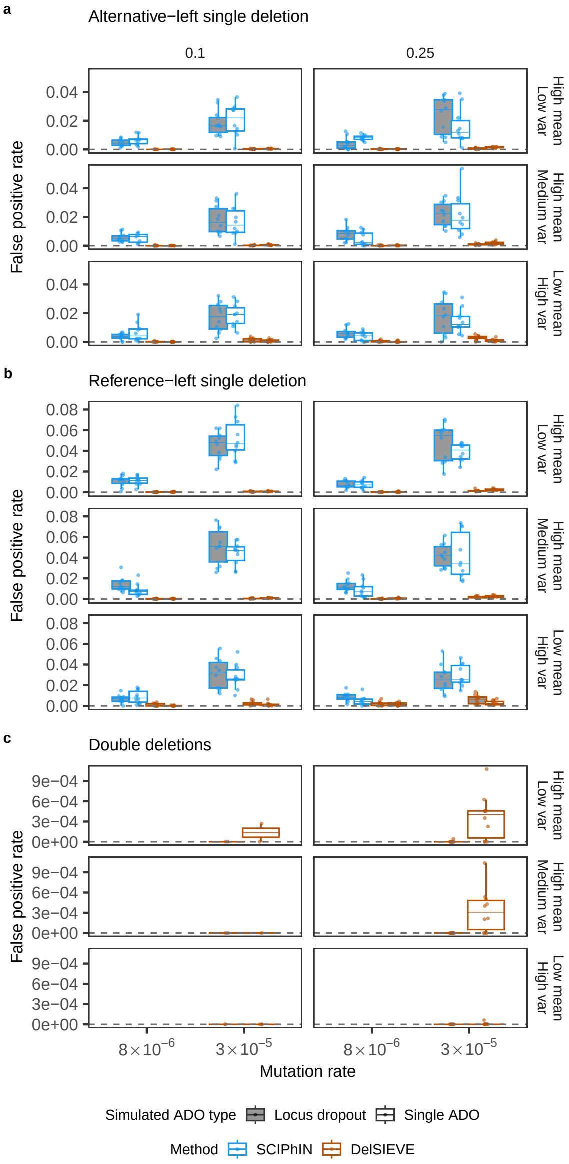


Figure 5.22: **False positive rate (FPR) for the benchmark of calling somatic deletions.** Varying are the mutation rate (the horizontal axis), the relative deletion rate (the vertical strip), the coverage quality (the horizontal strip) and the simulated ADO type (the shaded or blank boxes). Each simulation is repeated $n = 10$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. Data points were removed if the proportion of simulated ground truth was less than 0.1%. **a-c**, Box plots of the FPR for calling alternative-left single deletion (**a**), reference-left single deletion (**b**), and double deletion (**c**). The results in **c** when mutation rate was 8×10^{-6} were omitted as very few double deletion were generated (less than 0.2%).

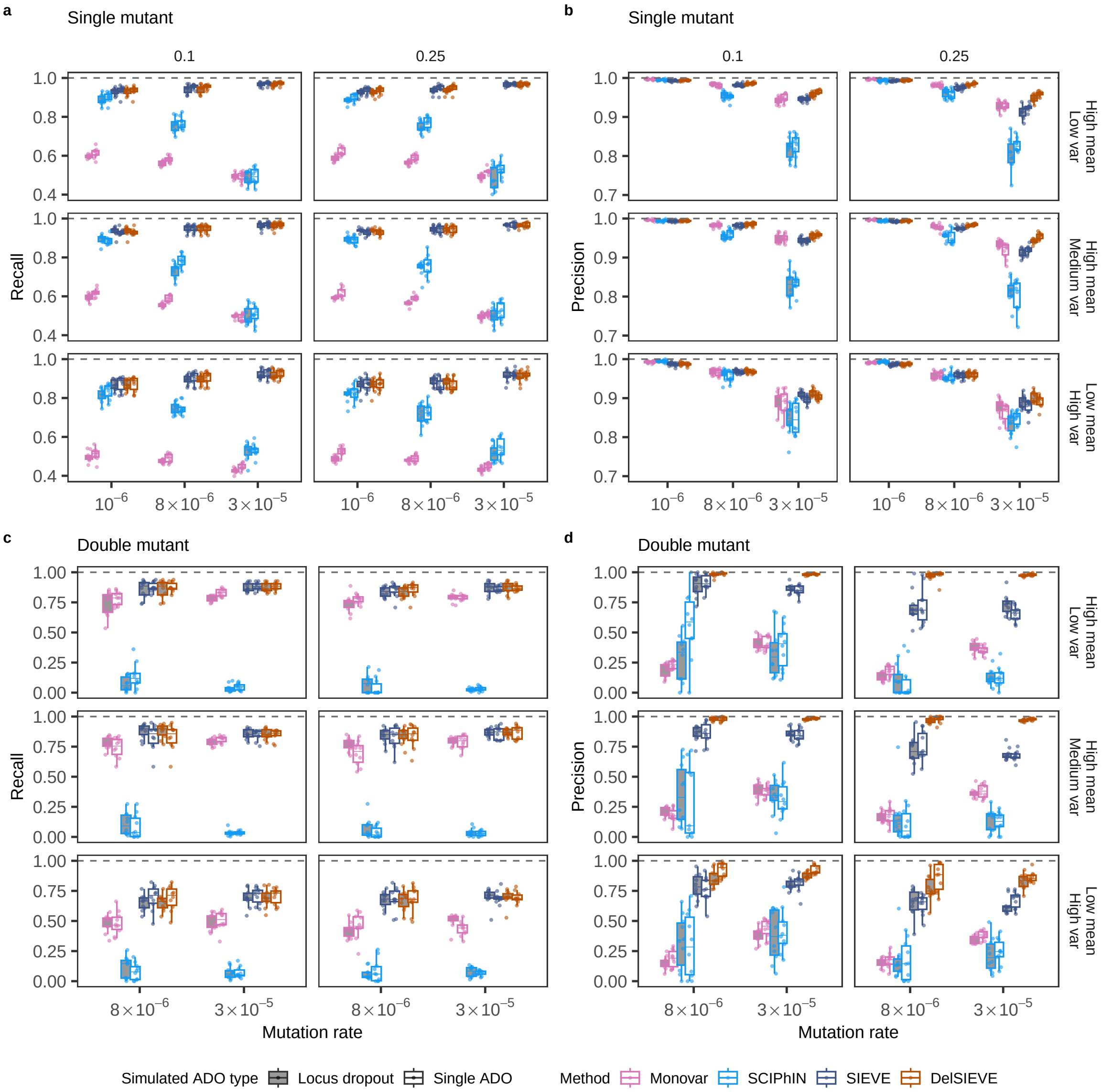


Figure 5.23: **Recall and precision for the benchmark of calling single and double mutant.** Varying are the mutation rate (the horizontal axis), the relative deletion rate (the vertical strip), the coverage quality (the horizontal strip) and the simulated ADO type (the shaded or blank boxes). Each simulation is repeated $n = 10$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. **a-b**, Box plots of the recall (**a**) and the precision (**b**) for calling single mutant. **c-d**, Box plots of the recall (**c**) and the precision (**d**) for calling double mutant, where the results when mutation rate was 10^{-6} were omitted as very few double mutant were generated (less than 0.2%).

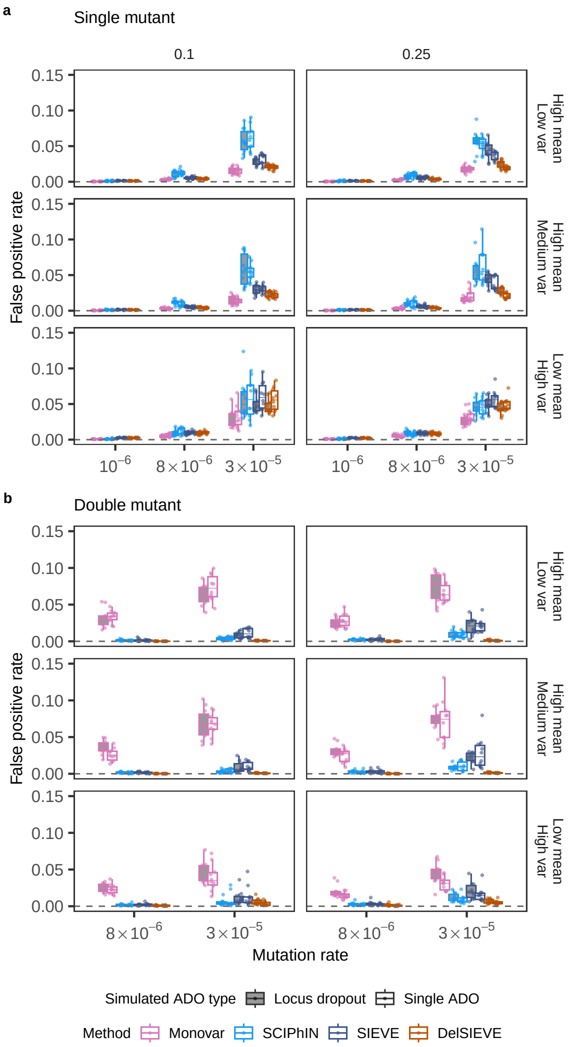


Figure 5.24: **False positive rate (FPR) for the benchmark of calling single and double mutant.** Varying are the mutation rate (the horizontal axis), the relative deletion rate (the vertical strip), the coverage quality (the horizontal strip) and the simulated ADO type (the shaded or blank boxes). Each simulation is repeated $n = 10$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. **a-b**, Box plots of the FPR for calling single mutant (**a**) and double mutant (**b**). The results in **b** when mutation rate was 10^{-6} were omitted as very few double mutant were generated (less than 0.2%).

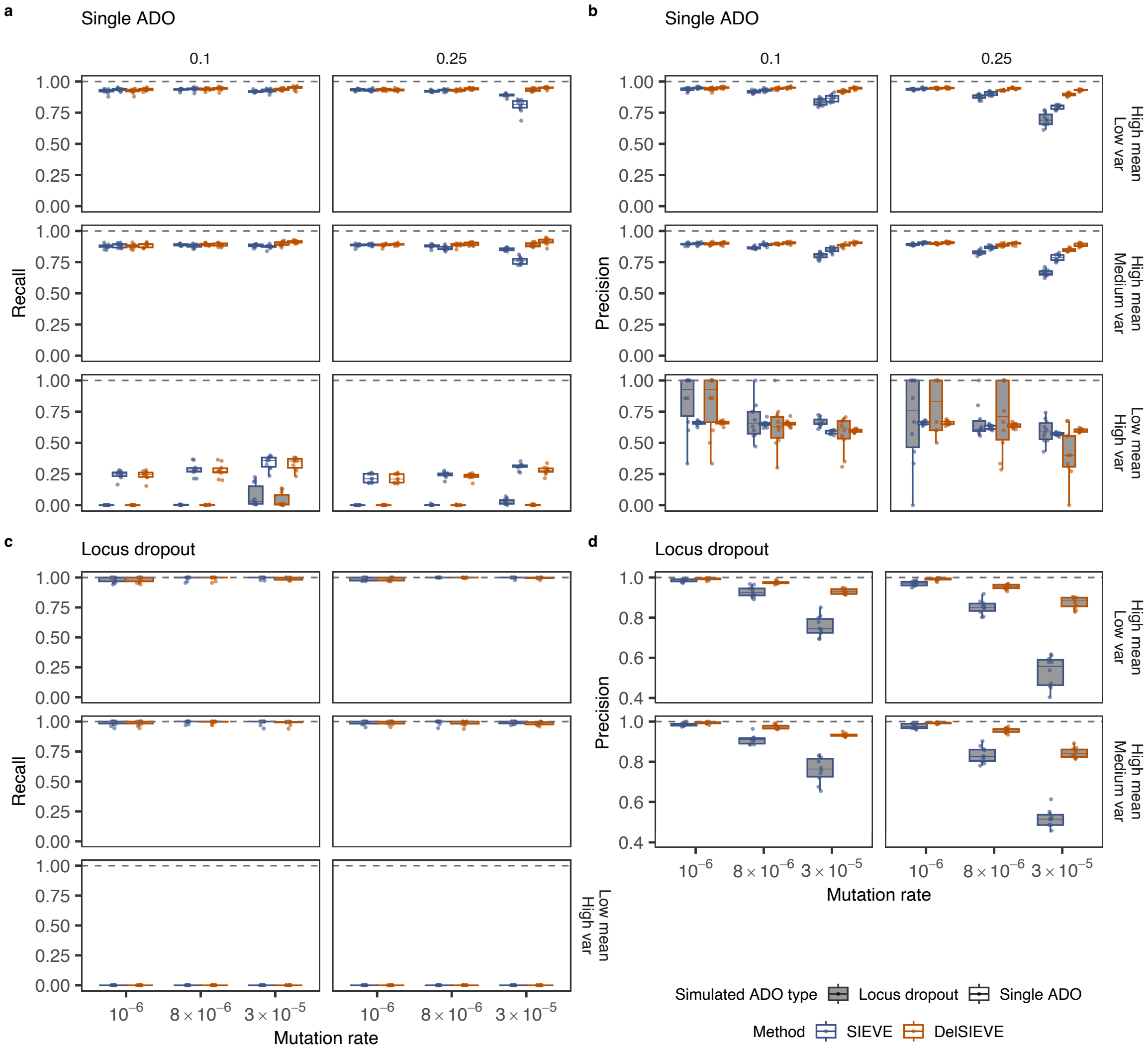


Figure 5.25: **Recall and precision for the benchmark of calling single ADO and locus dropout.** Varying are the mutation rate (the horizontal axis), the relative deletion rate (the vertical strip), the coverage quality (the horizontal strip) and the simulated ADO type (the shaded or blank boxes). Each simulation is repeated $n = 10$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. **a-b**, Box plots of the recall (**a**) and the precision (**b**) for calling single ADO. **c-d**, Box plots of the recall (**c**) and the precision (**d**) for calling locus dropout, where the precision were unavailable in **d** when data was of low coverage quality due to zero called locus dropout.

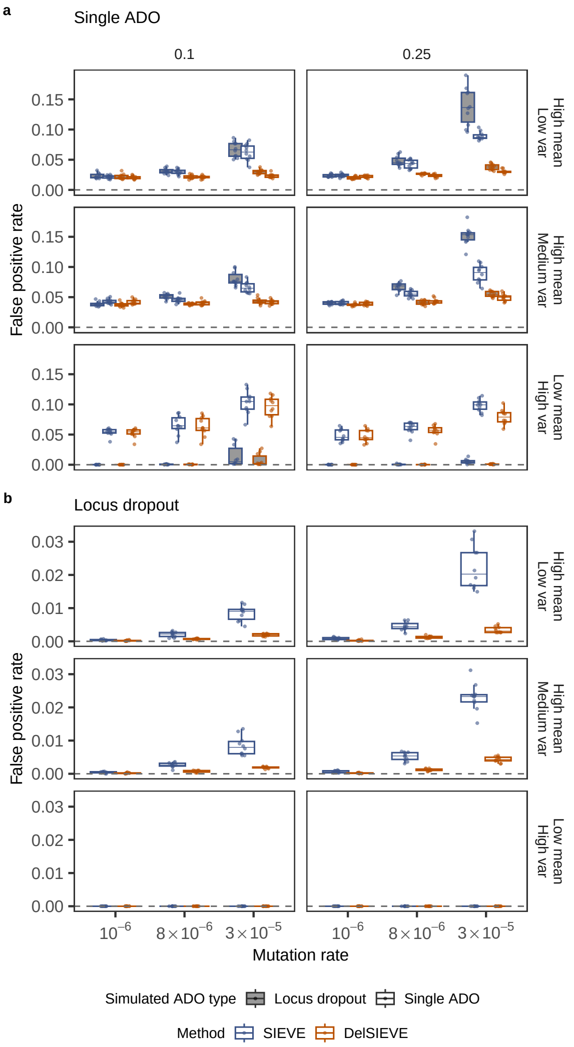


Figure 5.26: **False positive rate (FPR) for the benchmark of calling single ADO and locus dropout.** Varying are the mutation rate (the horizontal axis), the relative deletion rate (the vertical strip), the coverage quality (the horizontal strip) and the simulated ADO type (the shaded or blank boxes). Each simulation is repeated $n = 10$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. **a-b**, Box plots of the FPR for calling single ADO (**a**) and locus dropout (**b**).

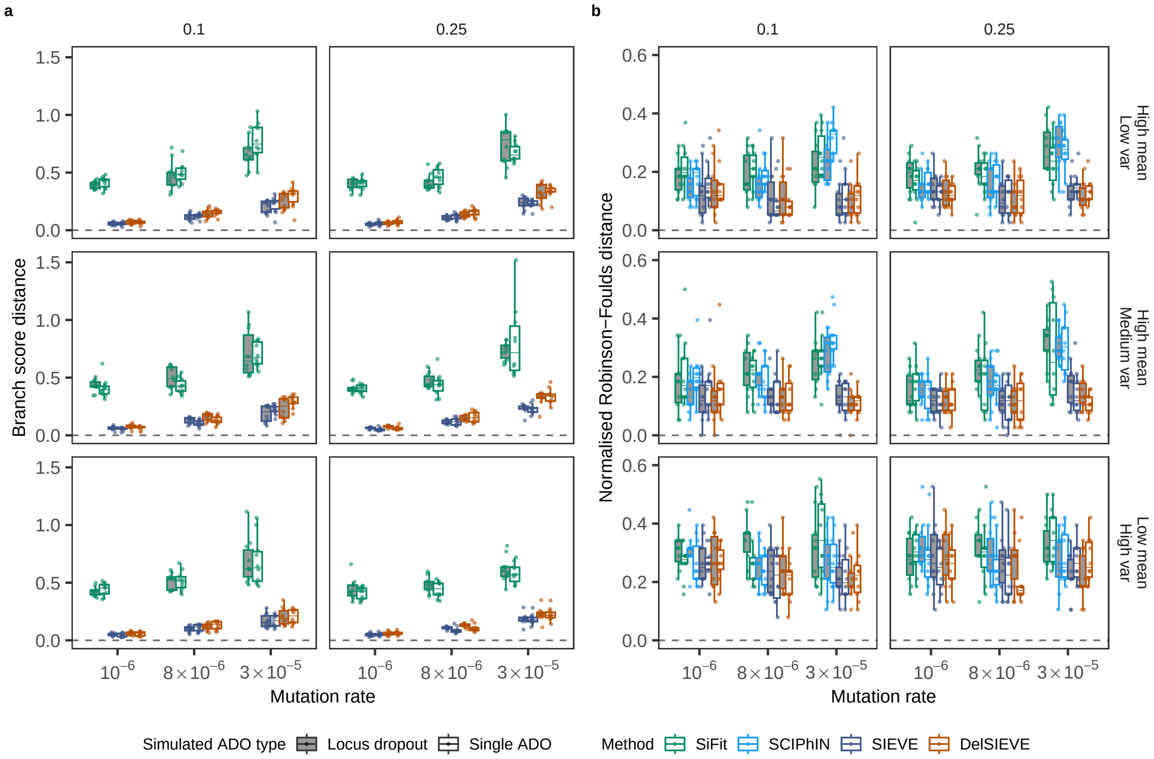


Figure 5.27: **Benchmark of tree inference accuracy.** Varying are the mutation rate (the horizontal axis), the relative deletion rate (the vertical strip), the coverage quality (the horizontal strip) and the simulated ADO type (the shaded or blank boxes). Each simulation is repeated $n = 10$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. **a-b**, Box plots of the BS distance where the branch lengths are taken into account (**a**) and the normalized RF distance where only tree topology is considered (**b**).

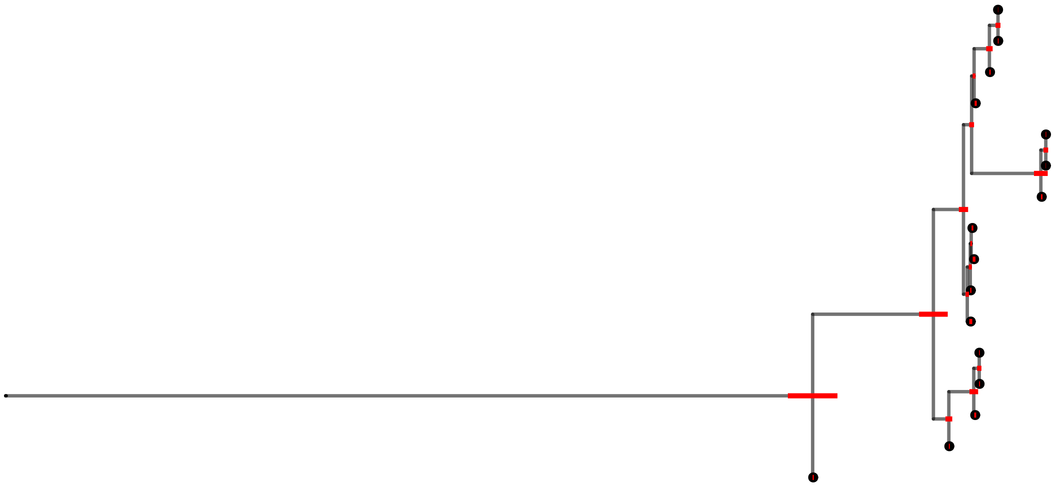


Figure 5.28: **Illustration of branch lengths of the phylogenetic tree inferred from TNBC16 [2] by DelSIEVE.** Shown is exactly the same tree as in Figure 3.5, except that cell names, subclone posterior probabilities and gene annotations are removed and no branches are folded. Red bars annotated to internal nodes except the root are the 95% HPD intervals of the corresponding branch lengths.

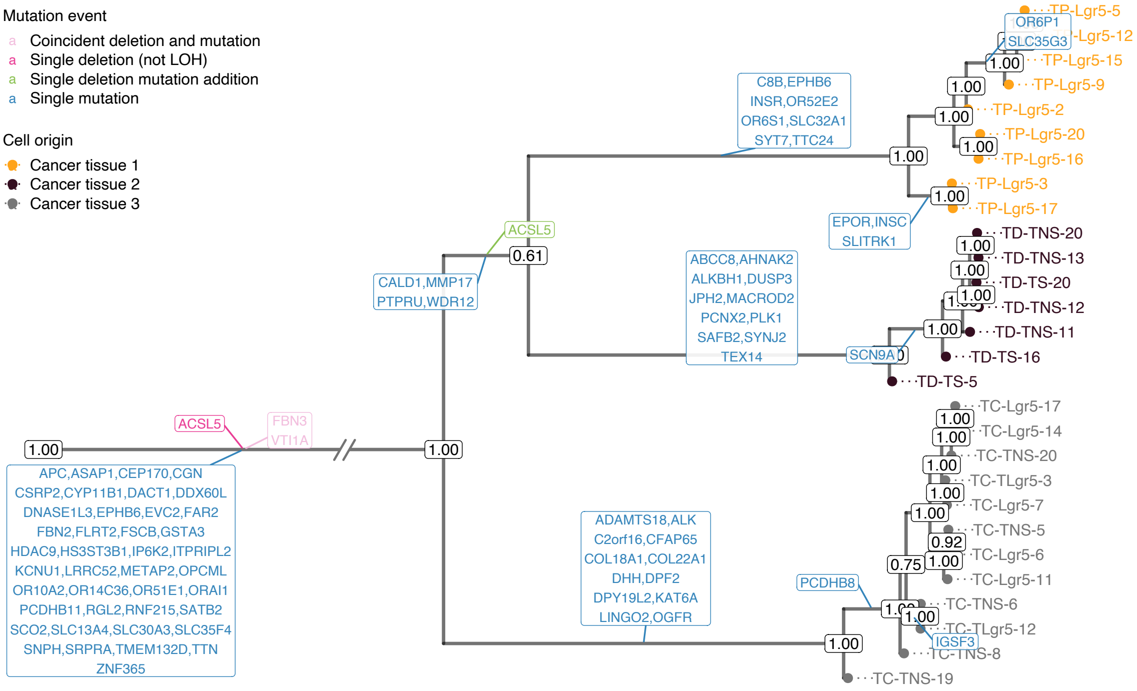


Figure 5.29: **Results of phylogenetic inference for the CRC28 [1] dataset.** Shown is DelSIEVE's maximum clade credibility tree. Tumor cell names are annotated to the leaves of the tree. The exceptionally long trunk has been folded (marked by slashes). Cells are colored according to the corresponding biopsies. The numbers at each node represent the posterior probabilities (threshold $p > 0.5$). At each branch, depicted in different colors are non-synonymous genes that are either CRC-related single mutations (in blue) or other mutation events (in other colors).

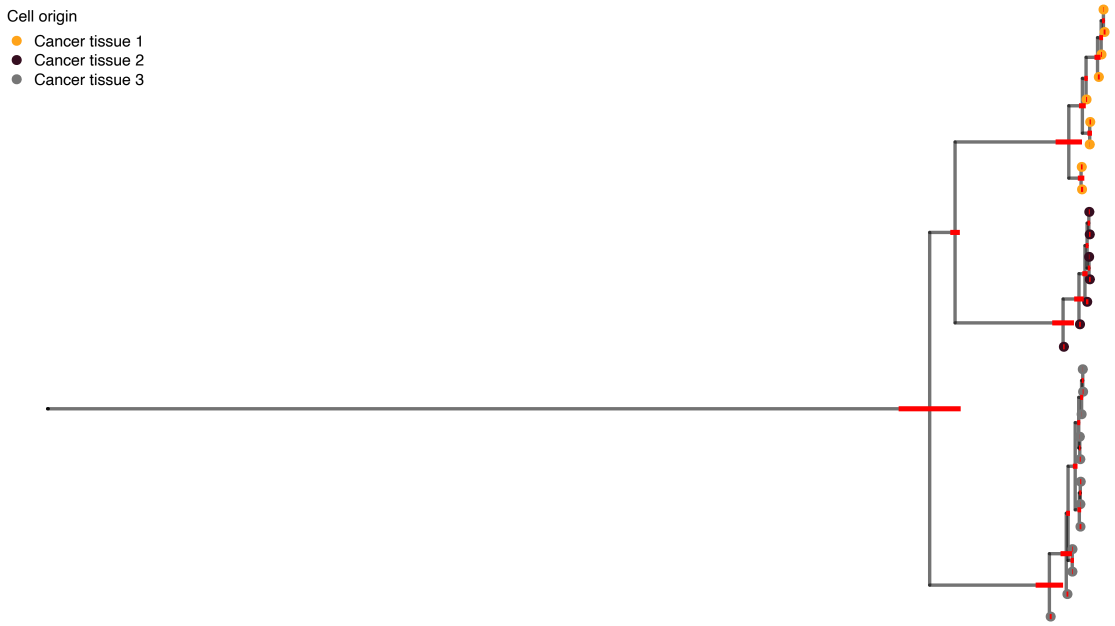


Figure 5.30: **Illustration of branch lengths of the phylogenetic tree inferred from CRC28 [1] by DelSIEVE.** Shown is exactly the same tree as in Figure 5.29, except that cell names, subclone posterior probabilities and gene annotations are removed and no branches are folded. Red bars annotated to internal nodes except the root are the 95% HPD intervals of the corresponding branch lengths.

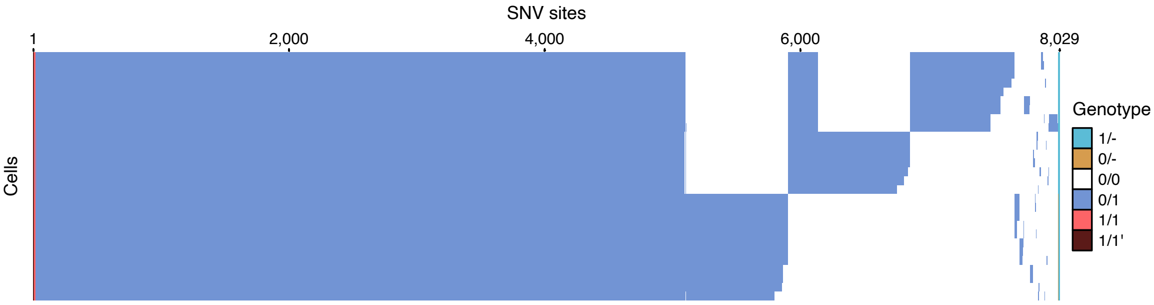


Figure 5.31: **Results of variant calling for the CRC28 [1] dataset.** Cells in the row are in the same order as that of leaves in the phylogenetic tree in Figure 5.29.

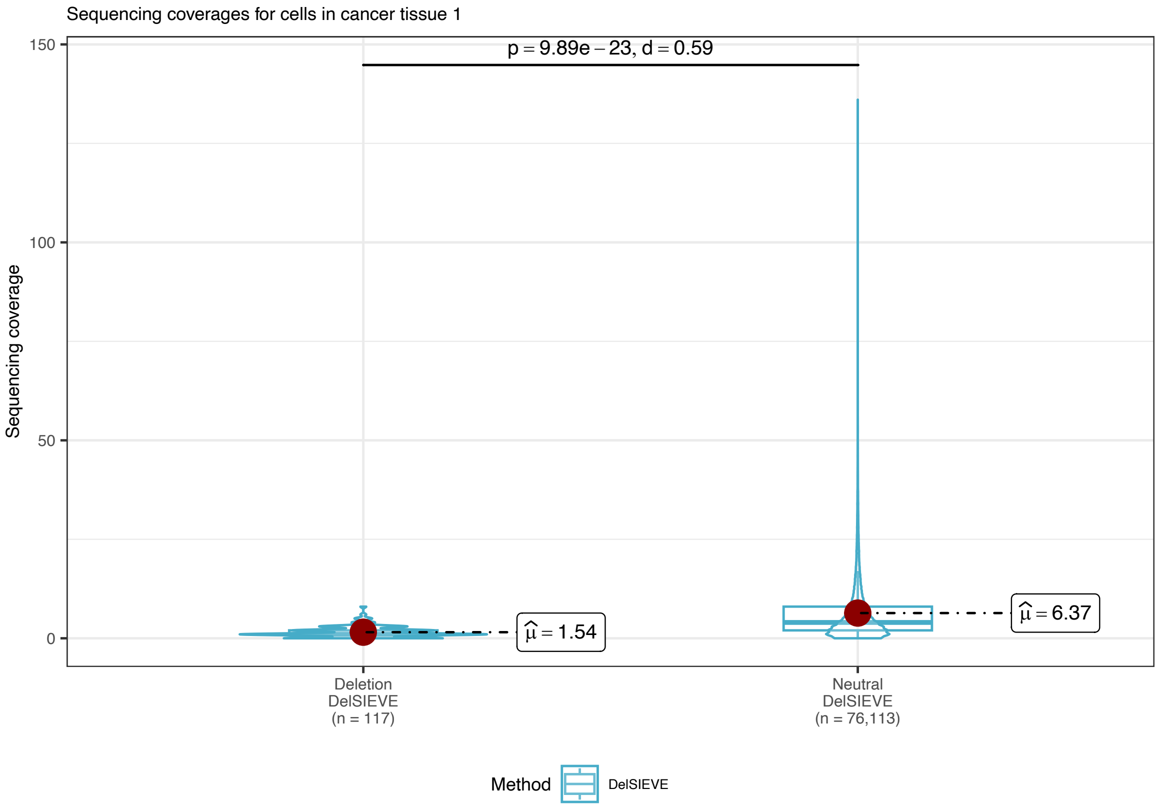


Figure 5.32: **Results of subclone-wise sequencing coverage comparison for TP cells in CRC28 [1].** For DelSIEVE, sites were divided into two groups, one with somatic deletions, the other copy neutral. In each group, the violin and the box plots matched the color of the method and showed the distribution of the sequencing coverage, while the burgundy dot denoted its mean value $\hat{\mu}$. The total number of dots in each group, which was the product of the number of cells (9) and the number of sites in each group, was marked with n on the horizontal axis. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. A within-group comparison was conducted between somatic deletions and copy neutral of DelSIEVE, where shown were the p-value corrected by Holm–Bonferroni method and the absolute value of the effect size (Cohen's d).

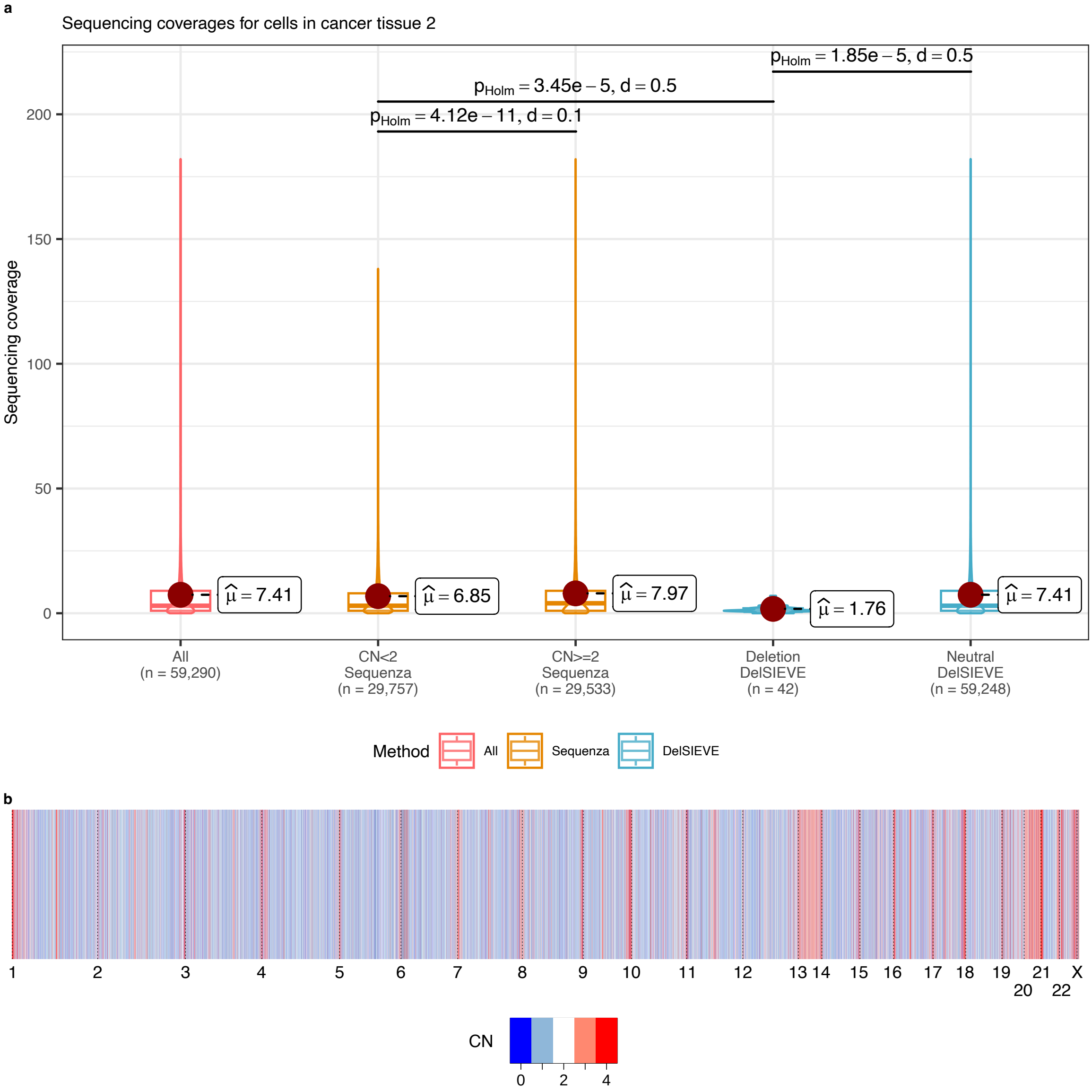


Figure 5.33: **Validation of somatic deletions called in TD cells for CRC28 [1].** **a**, Results of subclone-wise sequencing coverage comparison for TD cells in CRC28 between DelSIEVE and Sequenza. Compared were the sites shared between the input data of both methods. The resolution of variant calling was subclone-wise in order to conduct a fair comparison. For Sequenza, sites were divided into two groups with copy number (CN) < 2 and ≥ 2, respectively. For DelSIEVE, sites were also divided into two groups, one with somatic deletions, the other copy neutral. Sequencing coverage across all TD cells at all sites were plotted for reference. In each group, the violin and the box plots matched the color of the method and showed the distribution of the sequencing coverage, while the burgundy dot denoted its mean value $\hat{\mu}$. The total number of dots in each group, which was the product of the number of cells (7) and the number of sites in each group, was marked with n on the horizontal axis. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. Within- and between-group comparisons were conducted between CN < 2 and ≥ 2 of Sequenza, between somatic deletions and copy neutral of DelSIEVE, and between CN < 2 of Sequenza and somatic deletions of DelSIEVE. For each comparison, shown were the p-value corrected by Holm–Bonferroni method and the absolute value of the effect size (Cohen's d). **b**, plot of CNs called by Sequenza across the whole genome. Chromosome labels are aligned with the black dotted line in the plot, marking the start of the corresponding chromosome. CNs in the plot are colored according to the legend below.

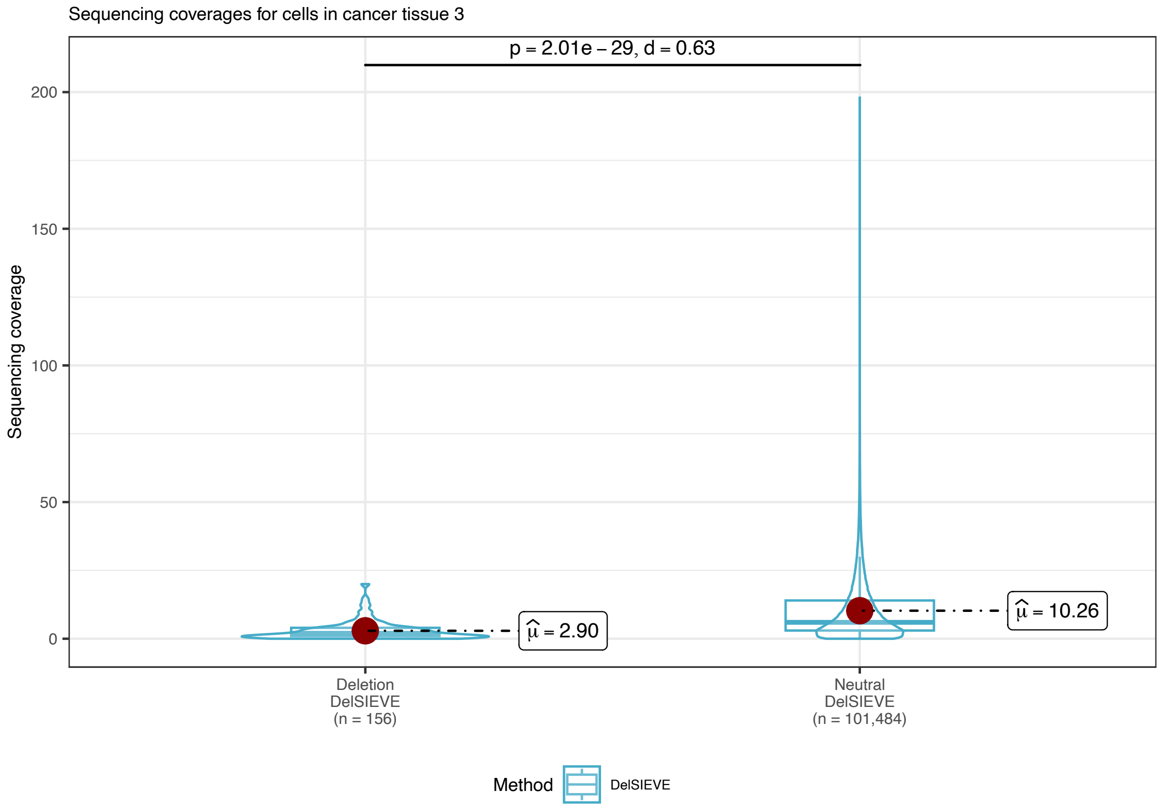


Figure 5.34: **Results of subclone-wise sequencing coverage comparison for TC cells in CRC28 [1].** For DelSIEVE, sites were divided into two groups, one with somatic deletions, the other copy neutral. In each group, the violin and the box plots matched the color of the method and showed the distribution of the sequencing coverage, while the burgundy dot denoted its mean value $\hat{\mu}$. The total number of dots in each group, which was the product of the number of cells (12) and the number of sites in each group, was marked with n on the horizontal axis. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. A within-group comparison was conducted between somatic deletions and copy neutral of DelSIEVE, where shown were the p-value corrected by Holm–Bonferroni method and the absolute value of the effect size (Cohen's d).

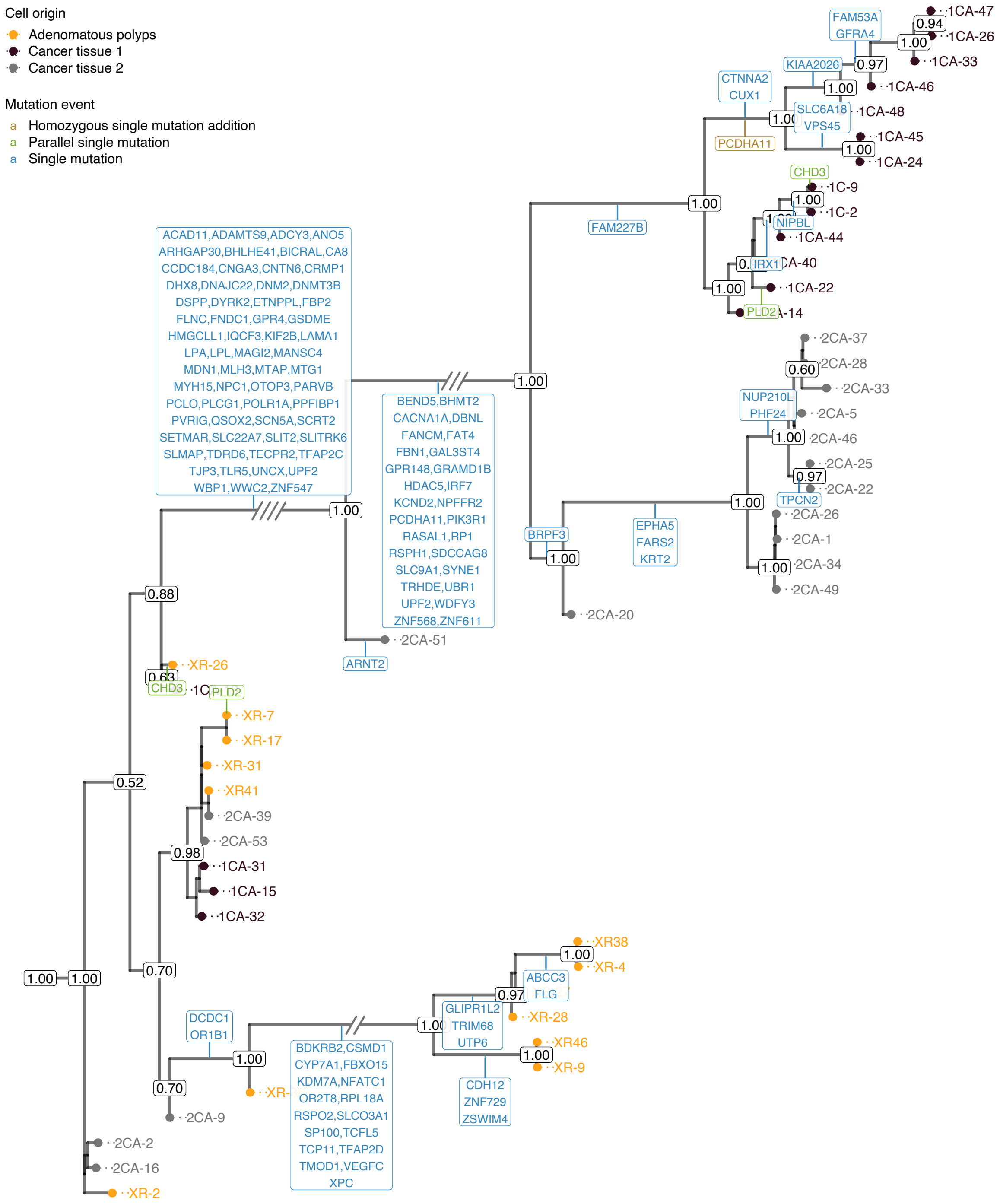


Figure 5.35: **Results of phylogenetic inference for the CRC48 [3] dataset.** Shown is DelSIEVE's maximum clade credibility tree. Tumor cell names are annotated to the leaves of the tree. Three exceptionally long branches are folded with the number of slashes proportional to the branch lengths. Cells are colored according to the corresponding biopsies. The numbers at each node represent the posterior probabilities (threshold $p > 0.5$). At each branch, depicted in different colors are non-synonymous genes that are either CRC-related single mutations (in blue) or other mutation events (in other colors).

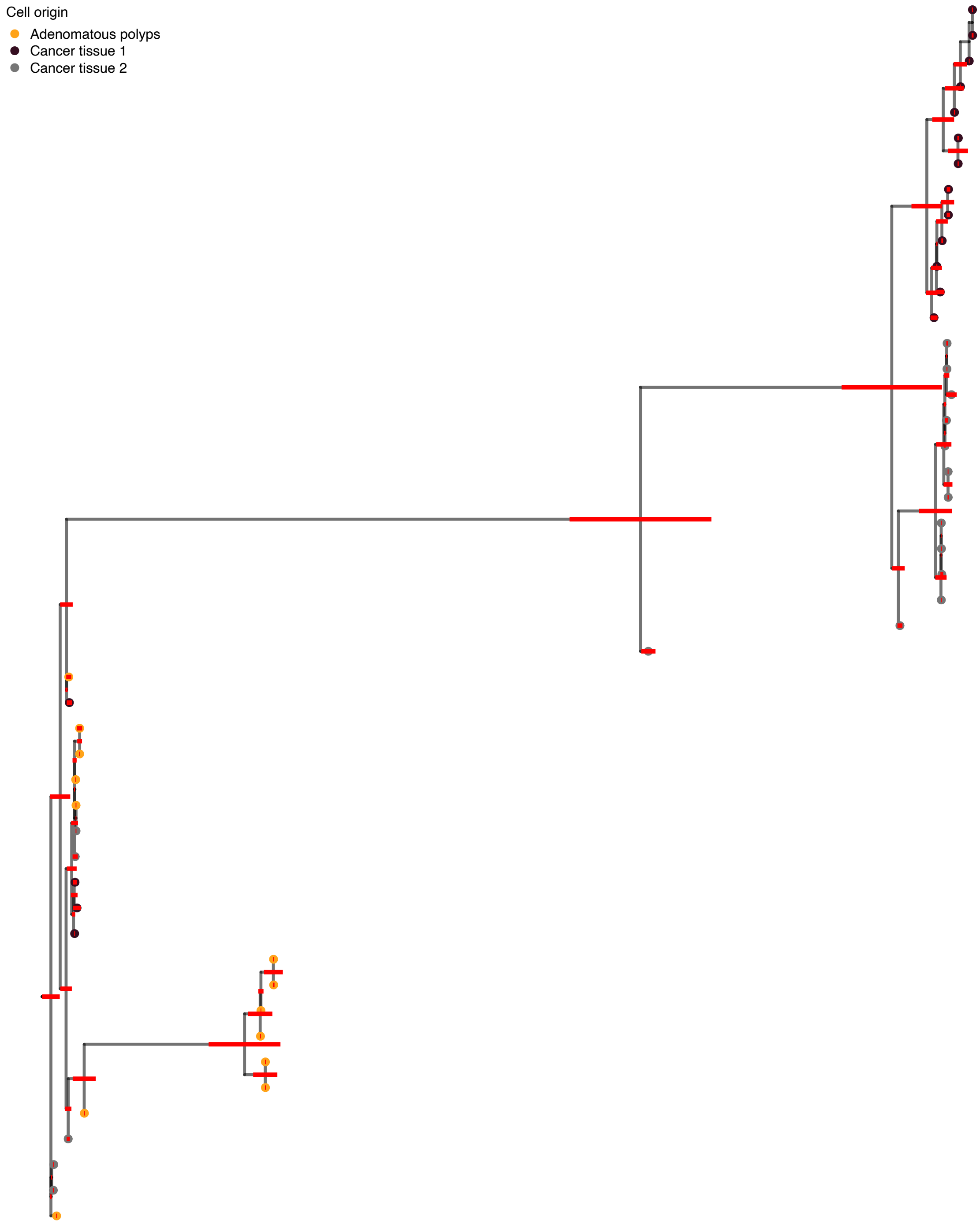


Figure 5.36: **Illustration of branch lengths of the phylogenetic tree inferred from CRC48 [3] by DelSIEVE.** Shown is exactly the same tree as in Figure 5.35, except that cell names, subclone posterior probabilities and gene annotations are removed and no branches are folded. Red bars annotated to internal nodes except the root are the 95% HPD intervals of the corresponding branch lengths.

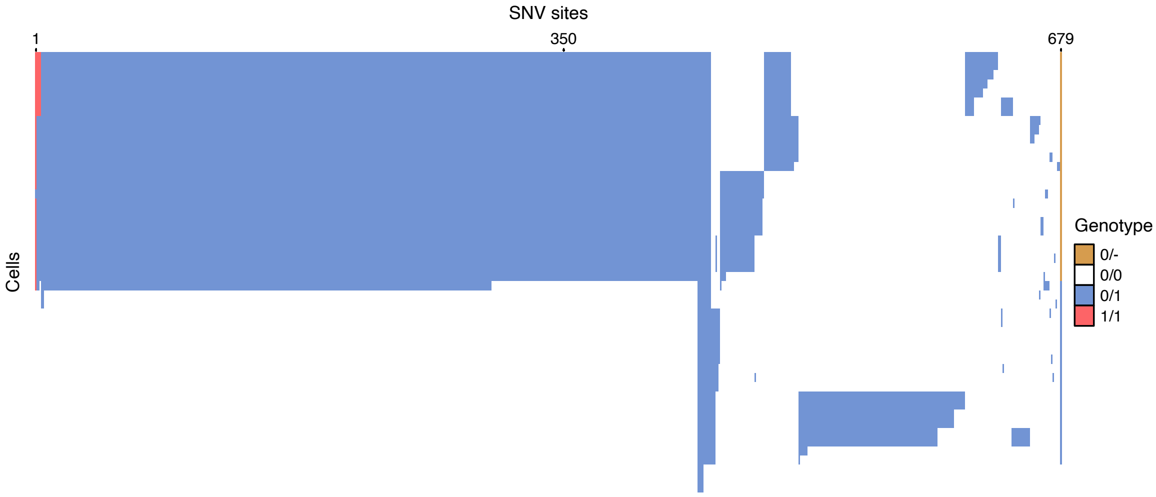


Figure 5.37: **Results of variant calling for the CRC48 [3] dataset.** Cells in the row are in the same order as that of leaves in the phylogenetic tree in Figure 5.35.

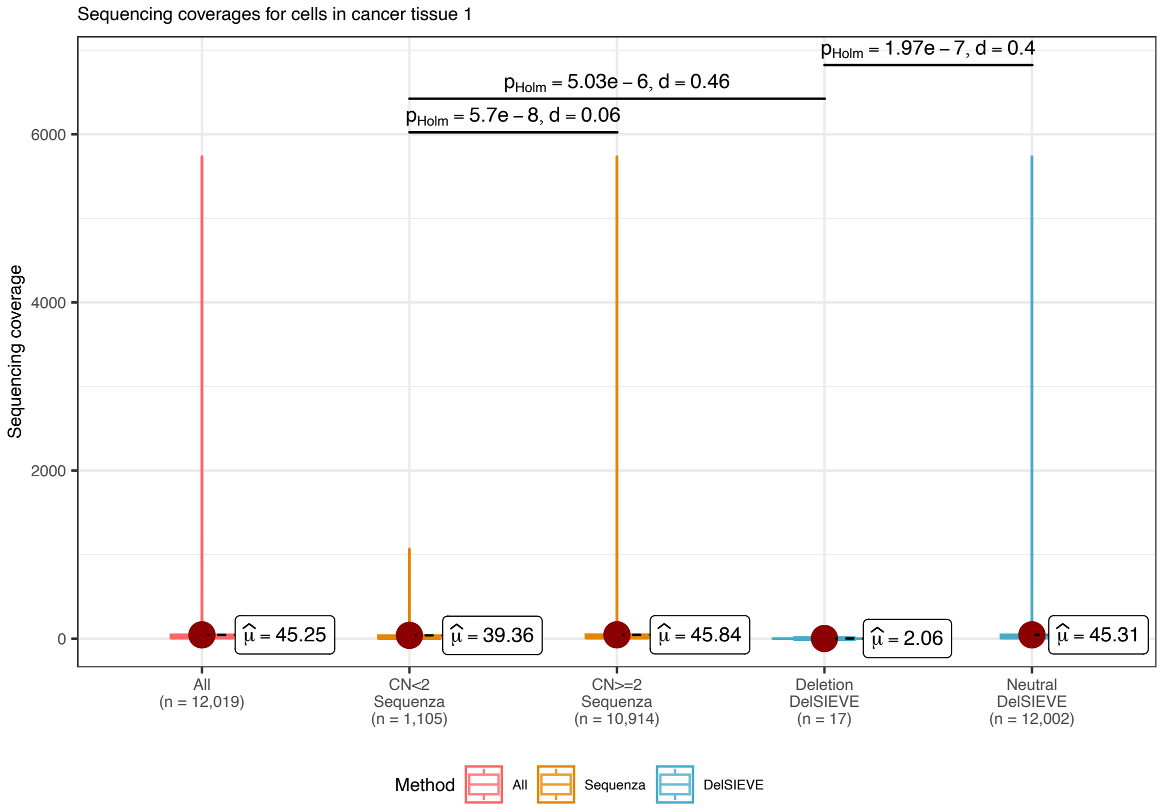


Figure 5.38: **Results of subclone-wise sequencing coverage comparison for cells in cancer tissue 1 of CRC48 [3] between DelSIEVE and Sequenza.** Compared were the sites shared between the input data of both methods. The resolution of variant calling was subclone-wise in order to conduct a fair comparison. For Sequenza, sites were divided into two groups with copy number (CN) < 2 and ≥ 2, respectively. For DelSIEVE, sites were divided into two groups, one with somatic deletions, the other copy neutral. Sequencing coverage across those cells in the subclone at all sites were plotted for reference. In each group, the violin and the box plots showed the distribution of the sequencing coverage, while the burgundy dot denoted its mean value $\hat{\mu}$. The total number of dots in each group, which was the product of the number of cells (17) and the number of sites in each group, was marked with n on the horizontal axis. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. Within- and between-group comparisons were conducted between CN < 2 and ≥ 2 of Sequenza, between somatic deletions and copy neutral of DelSIEVE, and between CN < 2 of Sequenza and somatic deletions of DelSIEVE. For each comparison, shown were the p-value corrected by Holm–Bonferroni method and the absolute value of the effect size (Cohen's d).

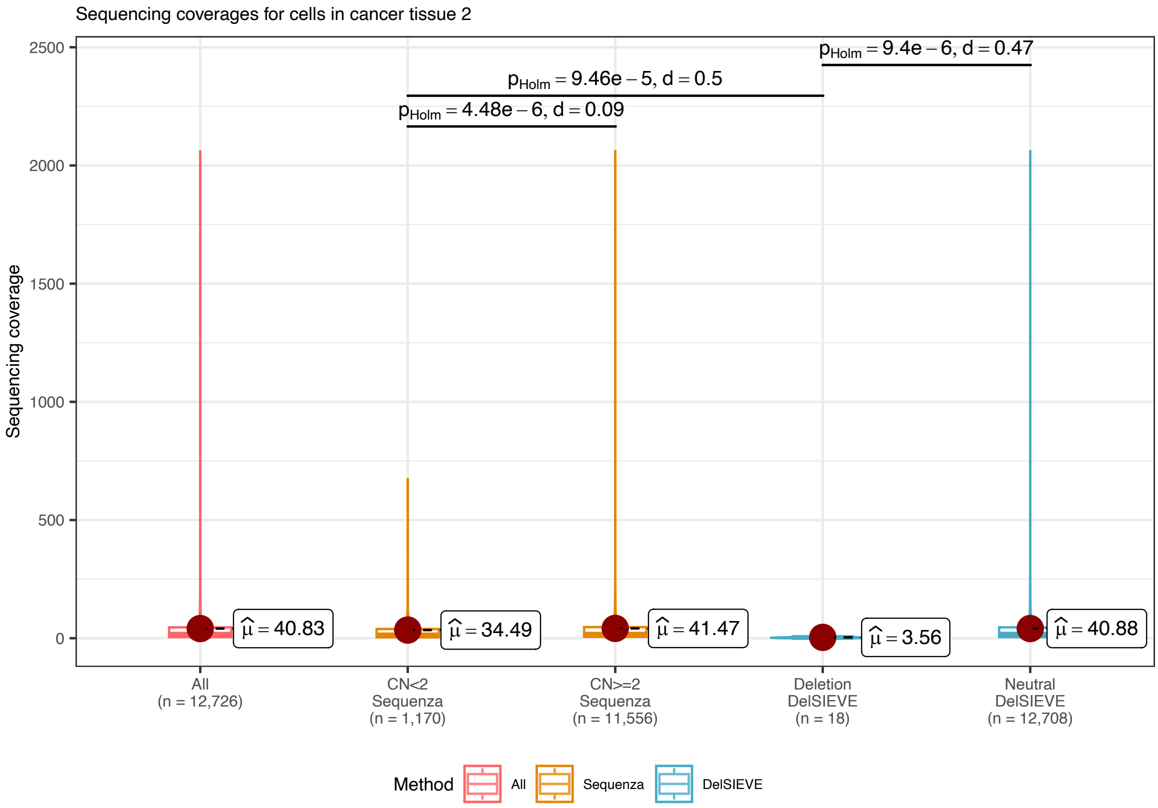


Figure 5.39: **Results of subclone-wise sequencing coverage comparison for cells in cancer tissue 2 of CRC48 [3] between DelSIEVE and Sequenza.** Compared were the sites shared between the input data of both methods. The resolution of variant calling was subclone-wise in order to conduct a fair comparison. For Sequenza, sites were divided into two groups with copy number (CN) < 2 and ≥ 2, respectively. For DelSIEVE, sites were also divided into two groups, one with somatic deletions, the other copy neutral. Sequencing coverage across those cells in the subclone at all sites were plotted for reference. In each group, the violin and the box plots showed matched the color of the method and the distribution of the sequencing coverage, while the burgundy dot denoted its mean value $\hat{\mu}$. The total number of dots in each group, which was the product of the number of cells (18) and the number of sites in each group, was marked with n on the horizontal axis. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. Within- and between-group comparisons were conducted between CN < 2 and ≥ 2 of Sequenza, between somatic deletions and copy neutral of DelSIEVE, and between CN < 2 of Sequenza and somatic deletions of DelSIEVE. For each comparison, shown were the p-value corrected by Holm–Bonferroni method and the absolute value of the effect size (Cohen's d).

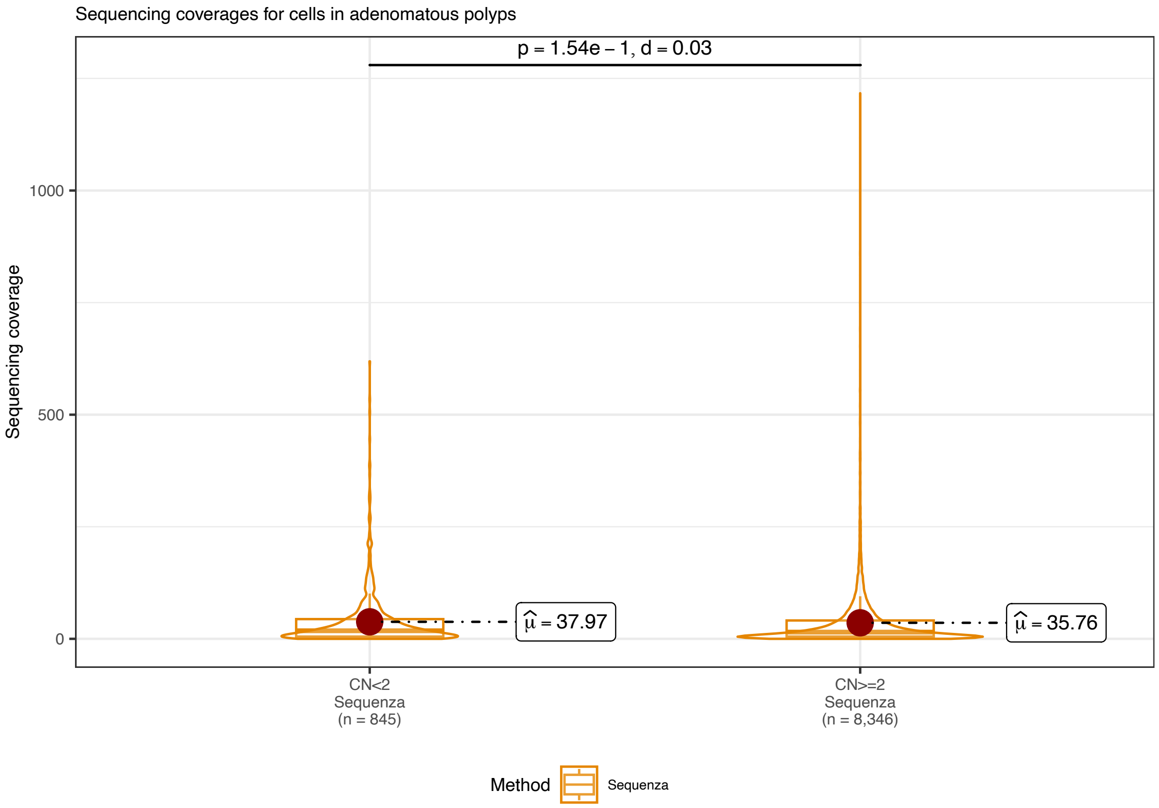


Figure 5.40: **Results of subclone-wise sequencing coverage comparison for adenomatous polyps cells in CRC48 [3].** For Sequenza, sites were divided into two groups with copy number (CN) < 2 and ≥ 2, respectively. In each group, the violin and the box plots matched the color of the method and showed the distribution of the sequencing coverage, while the burgundy dot denoted its mean value $\hat{\mu}$. The total number of dots in each group, which was the product of the number of cells (13) and the number of sites in each group, was marked with n on the horizontal axis. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. A within-group comparison was conducted between CN < 2 and CN ≥ 2 of Sequenza, where shown were the p-value corrected by Holm–Bonferroni method and the absolute value of the effect size (Cohen's d).

5.5. Supplementary tables of Chapter 3

Table 5.5: **Evolutionary rate matrix used in the simulator to generate the simulated data for DelSIEVE.** Genotypes are encoded with nucleotides rather than numbers. d is the deletion rate measured relatively to the mutation rate. The diagonal elements are denoted by dots, and have negative values equal to the sum of the other entries in the same row, ensuring that the sum of each row equals zero.

	A/A	A/C	A/G	A/T	C/C	C/G	C/T	G/G	G/T	T/T	A/-	C/-	G/-	T/-	-
A/A	.	$1/3$	$1/3$	$1/3$	0	0	0	0	0	0	d	0	0	0	0
A/C	$1/6$	.	$1/6$	$1/6$	$1/6$	$1/6$	$1/6$	0	0	0	$d/2$	$d/2$	0	0	0
A/G	$1/6$	$1/6$	.	$1/6$	0	$1/6$	0	$1/6$	$1/6$	0	$d/2$	0	$d/2$	0	0
A/T	$1/6$	$1/6$	$1/6$	.	0	0	$1/6$	0	$1/6$	$1/6$	$d/2$	0	0	$d/2$	0
C/C	0	$1/3$	0	0	.	$1/3$	$1/3$	0	0	0	0	d	0	0	0
C/G	0	$1/6$	$1/6$	0	$1/6$	.	$1/6$	$1/6$	$1/6$	0	0	$d/2$	$d/2$	0	0
C/T	0	$1/6$	0	$1/6$	$1/6$	$1/6$	.	0	$1/6$	$1/6$	0	$d/2$	0	$d/2$	0
G/G	0	0	$1/3$	0	0	$1/3$	0	.	$1/3$	0	0	0	d	0	0
G/T	0	0	$1/6$	$1/6$	0	$1/6$	$1/6$	$1/6$	.	$1/6$	0	0	$d/2$	$d/2$	0
T/T	0	0	0	$1/3$	0	0	$1/3$	0	$1/3$	.	0	0	0	d	0
A/-	0	0	0	0	0	0	0	0	0	0	.	$1/6$	$1/6$	$1/6$	$d/2$
C/-	0	0	0	0	0	0	0	0	0	0	$1/6$	.	$1/6$	$1/6$	$d/2$
G/-	0	0	0	0	0	0	0	0	0	0	$1/6$	$1/6$	.	$1/6$	$d/2$
T/-	0	0	0	0	0	0	0	0	0	0	$1/6$	$1/6$	$1/6$	.	$d/2$
-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.

Table 5.6: **Summary of fractions of predicted genotypes by DelSIEVE and SIEVE for three analyzed real datasets.** Entries marked with NA denote that the corresponding method does not call the specific genotype.

		-	1/-	0/-	0/0	0/1	1/1	1/1′
TNBC16	DelSIEVE	0	11.51%	0.07%	15.58%	69.82%	2.99%	0.03%
	SIEVE	NA	NA	NA	15.54%	75.11%	9.30%	0.05%
CRC28	DelSIEVE	0	0.15%	0.02%	25.02%	74.59%	0.16%	0.06%
	SIEVE	NA	NA	NA	25.02%	74.64%	0.28%	0.06%
CRC48	DelSIEVE	0	0	0.08%	59.61%	40.17%	0.14%	0
	SIEVE	NA	NA	NA	59.48%	40.50%	0.02%	0

Bibliography

[1] Senbai Kang, Nico Borgsmüller, Monica Valecha, Jack Kuipers, Joao M. Alves, Sonia Prado-López, Débora Chantada, Niko Beerenwinkel, David Posada, and Ewa Szczurek. Sieve: joint inference of single-nucleotide variants and cell phylogeny from single-cell dna sequencing data. *Genome Biology*, 23(1):248, Nov 2022.

[2] Yong Wang, Jill Waters, Marco L. Leung, Anna Unruh, Whijae Roh, Xiuqing Shi, Ken Chen, Paul Scheet, Selina Vattathil, Han Liang, Asha Multani, Hong Zhang, Rui Zhao, Franziska Michor, Funda Meric-Bernstam, and Nicholas E. Navin. Clonal evolution in breast cancer revealed by single nucleus genome sequencing. *Nature*, 512(7513):155–160, 8 2014.

[3] H Wu, XY Zhang, Z Hu, Q Hou, H Zhang, Y Li, S Li, J Yue, Z Jiang, SM Weissman, et al. Evolution and heterogeneity of non-hereditary colorectal cancer revealed by single-cell exome sequencing. *Oncogene*, 36(20):2857–2867, 2017.

[4] WE Damsky, N Theodosakis, and M2 Bosenberg. Melanoma metastasis: new concepts and evolving paradigms. *Oncogene*, 33(19):2413–2422, 2014.

[5] Xiangming Guan. Cancer metastases: challenges and opportunities. *Acta Pharmaceutica Sinica B*, 5(5):402–418, 2015.

[6] Thomas N Seyfried and Leanne C Huysentruyt. On the origin of cancer metastasis. *Critical Reviews™ in Oncogenesis*, 18(1-2), 2013.

[7] Hyuna Sung, Jacques Ferlay, Rebecca L Siegel, Mathieu Laversanne, Isabelle Soerjomataram, Ahmedin Jemal, and Freddie Bray. Global cancer statistics 2020: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians*, 71(3):209–249, 2021.

[8] Peter Armitage and Richard Doll. A two-stage theory of carcinogenesis in relation to the age distribution of human cancer. *British Journal of Cancer*, 11(2):161, 1957.

[9] Alfred G Knudson Jr. Mutation and cancer: statistical study of retinoblastoma. *Proceedings of the National Academy of Sciences*, 68(4):820–823, 1971.

[10] Peter C Nowell. The clonal evolution of tumor cell populations: Acquired genetic lability permits stepwise selection of variant sublines and underlies tumor progression. *Science*, 194(4260):23–28, 1976.

[11] Eric R Fearon and Bert Vogelstein. A genetic model for colorectal tumorigenesis. *Cell*, 61(5):759–767, 1990.

[12] Douglas Hanahan and Robert A Weinberg. The hallmarks of cancer. *Cell*, 100(1):57–70, 2000.

[13] Douglas Hanahan and Robert A Weinberg. Hallmarks of cancer: the next generation. *Cell*, 144(5):646–674, 2011.

[14] Bert Vogelstein, Nickolas Papadopoulos, Victor E Velculescu, Shibin Zhou, Luis A Diaz Jr, and Kenneth W Kinzler. Cancer genome landscapes. *Science*, 339(6127):1546–1558, 2013.

[15] Douglas Hanahan. Hallmarks of cancer: new dimensions. *Cancer Discovery*, 12(1):31–46, 2022.

[16] Michael R Stratton, Peter J Campbell, and P Andrew Futreal. The cancer genome. *Nature*, 458(7239):719–724, 2009.

[17] Jing Chen and Jun-tao Guo. Comparative assessments of indel annotations in healthy and cancer genomes with next-generation sequencing data. *BMC Medical Genomics*, 13:1–11, 2020.

[18] Rinne De Bont and Nik Van Larebeke. Endogenous dna damage in humans: a review of quantitative data. *Mutagenesis*, 19(3):169–185, 2004.

[19] Cristian Tomasetti and Bert Vogelstein. Variation in cancer risk among tissues can be explained by the number of stem cell divisions. *Science*, 347(6217):78–81, 2015.

[20] National Cancer Institute, Definition of oncogene. https://www.cancer.gov/publications/dictionaries/cancer-terms/def/oncogene. [Accessed July 23, 2023].

[21] Rodrigo Dienstmann, Kate Connor, Annette T Byrne, WH Fridman, D Lambrechts, A Sadanandam, L Trusolino, JHM Prehn, J Tabernero, and W Kolch. Precision therapy in ras mutant colorectal cancer. *Gastroenterology*, 158(4):806–811, 2020.

[22] Joshua H Cook, Giorgio EM Melloni, Doga C Gulhan, Peter J Park, and Kevin M Haigis. The origins and genetic interactions of kras mutations are allele-and tissue-specific. *Nature Communications*, 12(1):1808, 2021.

[23] Chunxiao Zhu, Xiaoqing Guan, Xinuo Zhang, Xin Luan, Zhengbo Song, Xiangdong Cheng, Weidong Zhang, and Jiang-Jiang Qin. Targeting kras mutant cancers: from druggable therapy to drug resistance. *Molecular Cancer*, 21(1):159, 2022.

[24] National Cancer Institute, Definition of tumor suppressor gene. https://www.cancer.gov/publications/dictionaries/genetics-dictionary/def/tumor-suppressor-gene. [Accessed July 23, 2023].

[25] Christopher Greenman, Philip Stephens, Raffaella Smith, Gillian L Dalgliesh, Christopher Hunter, Graham Bignell, Helen Davies, Jon Teague, Adam Butler, Claire Stevens, et al. Patterns of somatic mutation in human cancer genomes. *Nature*, 446(7132):153–158, 2007.

[26] Rameen Beroukhim, Craig H Mermel, Dale Porter, Guo Wei, Soumya Raychaudhuri, Jerry Donovan, Jordi Barretina, Jesse S Boehm, Jennifer Dobson, Mitsuyoshi Urashima, et al. The landscape of somatic copy-number alteration across human cancers. *Nature*, 463(7283):899–905, 2010.

[27] Colleen A Brady and Laura D Attardi. p53 at a glance. *Journal of Cell Science*, 123(15):2527–2532, 2010.

[28] Karen H Vousden and Carol Prives. Blinded by the light: the growing complexity of p53. *Cell*, 137(3):413–431, 2009.

[29] Philip C Hanawalt and Graciela Spivak. Transcription-coupled dna repair: two decades of progress and surprises. *Nature Reviews Molecular Cell Biology*, 9(12):958–970, 2008.

[30] Alessandro Torgovnick and Björn Schumacher. Dna repair mechanisms in cancer development and therapy. *Frontiers in Genetics*, 6:157, 2015.

[31] Vinod Tiwari and David M Wilson. Dna damage and associated dna repair defects in disease and premature aging. *The American Journal of Human Genetics*, 105(2):237–257, 2019.

[32] Nicholas McGranahan and Charles Swanton. Clonal heterogeneity and tumor evolution: Past, present, and the future. *Cell*, 168(4):613–628, 2017.

[33] Mel Greaves and Carlo C Maley. Clonal evolution in cancer. *Nature*, 481(7381):306–313, 2012.

[34] Lucy R Yates and Peter J Campbell. Evolution of the cancer genome. *Nature Reviews Genetics*, 13(11):795–806, 2012.

[35] Andriy Marusyk, Michalina Janiszewska, and Kornelia Polyak. Intratumor heterogeneity: The rosetta stone of therapy resistance. *Cancer Cell*, 37(4):471–484, 2020.

[36] Jacinta Abraham and John Staffurth. Hormonal therapy for cancer. *Medicine*, 44(1):30–33, 2016.

[37] Volker Schirrmacher. From chemotherapy to biological therapy: A review of novel concepts to reduce the side effects of systemic cancer treatment. *International Journal of Oncology*, 54(2):407–419, 2019.

[38] Rebecca A Burrell, Nicholas McGranahan, Jiri Bartek, and Charles Swanton. The causes and consequences of genetic heterogeneity in cancer evolution. *Nature*, 501(7467):338–345, 2013.

[39] Xuan Wang, Haiyun Zhang, and Xiaozhuo Chen. Drug resistance and combating drug resistance in cancer. *Cancer Drug Resistance*, 2(2):141, 2019.

[40] Frederick Sanger, Steven Nicklen, and Alan R Coulson. Dna sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences*, 74(12):5463–5467, 1977.

[41] Allan M Maxam and Walter Gilbert. A new method for sequencing dna. *Proceedings of the National Academy of Sciences*, 74(2):560–564, 1977.

[42] James M Heather and Benjamin Chain. The sequence of sequencers: The history of sequencing dna. *Genomics*, 107(1):1–8, 2016.

[43] Jay Shendure, Shankar Balasubramanian, George M Church, Walter Gilbert, Jane Rogers, Jeffery A Schloss, and Robert H Waterston. Dna sequencing at 40: past, present and future. *Nature*, 550(7676):345–353, 2017.

[44] Morteza Jalali, Francesca Yvonne Louise Saldanha, and Mehdi Jalali. *Basic science methods for clinical researchers*. Academic Press, 2017.

[45] Barton E Slatko, Andrew F Gardner, and Frederick M Ausubel. Overview of next-generation sequencing technologies. *Current Protocols in Molecular Biology*, 122(1):e59, 2018.

[46] Marcel Margulies, Michael Egholm, William E Altman, Said Attiya, Joel S Bader, Lisa A Bemben, Jan Berka, Michael S Braverman, Yi-Ju Chen, Zhoutao Chen, et al. Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, 437(7057):376–380, 2005.

[47] Jay Shendure, Gregory J Porreca, Nikos B Reppas, Xiaoxia Lin, John P McCutcheon, Abraham M Rosenbaum, Michael D Wang, Kun Zhang, Robi D Mitra, and George M Church. Accurate multiplex polony sequencing of an evolved bacterial genome. *Science*, 309(5741):1728–1732, 2005.

[48] Ido Braslavsky, Benedict Hebert, Emil Kartalov, and Stephen R Quake. Sequence information can be obtained from single dna molecules. *Proceedings of the National Academy of Sciences*, 100(7):3960–3964, 2003.

[49] Farzin Haque, Jinghong Li, Hai-Chen Wu, Xing-Jie Liang, and Peixuan Guo. Solid-state and biological nanopore for real-time sensing of single chemical and sequencing of dna. *Nano Today*, 8(1):56–74, 2013.

[50] Shanika L Amarasinghe, Shian Su, Xueyi Dong, Luke Zappia, Matthew E Ritchie, and Quentin Gouil. Opportunities and challenges in long-read sequencing data analysis. *Genome Biology*, 21(1):1–16, 2020.

[51] Cheng Yong Tham, Roberto Tirado-Magallanes, Yufen Goh, Melissa J Fullwood, Bryan TH Koh, Wilson Wang, Chin Hin Ng, Wee Joo Chng, Alexandre Thiery, Daniel G Tenen, et al. Nanovar: accurate characterization of patients' genomic structural variants using low-depth nanopore sequencing. *Genome Biology*, 21(1):1–15, 2020.

[52] Nicholas Navin, Jude Kendall, Jennifer Troge, Peter Andrews, Linda Rodgers, Jeanne McIndoo, Kerry Cook, Asya Stepansky, Dan Levy, Diane Esposito, et al. Tumour evolution inferred by single-cell sequencing. *Nature*, 472(7341):90–94, 2011.

[53] Nicholas E Navin. Cancer genomics: one cell at a time. *Genome Biology*, 15(8):1–13, 2014.

[54] Nicholas E Navin. The first five years of single-cell cancer genomics and beyond. *Genome Research*, 25(10):1499–1507, 2015.

[55] David Lähnemann, Johannes Köster, Ewa Szczurek, Davis J McCarthy, Stephanie C Hicks, Mark D Robinson, Catalina A Vallejos, Kieran R Campbell, Niko Beerenwinkel, Ahmed Mahfouz, et al. Eleven grand challenges in single-cell data science. *Genome Biology*, 21(1):1–35, 2020.

[56] Charles Gawad, Winston Koh, and Stephen R Quake. Single-cell genome sequencing: current state of the science. *Nature Reviews Genetics*, 17(3):175–188, 2016.

[57] Timour Baslan and James Hicks. Unravelling biology and shifting paradigms in cancer with single-cell sequencing. *Nature Reviews Cancer*, 17(9):557–569, 2017.

[58] Nuria Estévez-Gómez, Tamara Prieto, Amy Guillaumet-Adkins, Holger Heyn, Sonia Prado-López, and David Posada. Comparison of single-cell whole-genome amplification strategies. *bioRxiv*, 2018.

[59] Xian F Mallory, Mohammadamin Edrisi, Nicholas Navin, and Luay Nakhleh. Methods for copy number aberration detection from single-cell dna-sequencing data. *Genome Biology*, 21(1):1–22, 2020.

[60] Nigel P Carter, Charlotte E Bebb, Magnus Nordenskjo, Bruce AJ Ponder, Alan Tunnacliffe, et al. Degenerate oligonucleotide-primed pcr: general amplification of target dna by a single degenerate primer. *Genomics*, 13(3):718–725, 1992.

[61] Timour Baslan, Jude Kendall, Linda Rodgers, Hilary Cox, Mike Riggs, Asya Stepansky, Jennifer Troge, Kandasamy Ravi, Diane Esposito, Bv Lakshmi, et al. Genome-wide copy number analysis of single cells. *Nature Protocols*, 7(6):1024–1041, 2012.

[62] Timour Baslan, Jude Kendall, Brian Ward, Hilary Cox, Anthony Leotta, Linda Rodgers, Michael Riggs, Sean D'Italia, Guoli Sun, Mao Yong, et al. Optimizing sparse sequencing of single cells for highly multiplex copy number profiling. *Genome Research*, 25(5):714–724, 2015.

[63] Charles FA De Bourcy, Iwijn De Vlaminck, Jad N Kanbar, Jianbin Wang, Charles Gawad, and Stephen R Quake. A quantitative comparison of single-cell whole genome amplification methods. *PloS One*, 9(8):e105585, 2014.

[64] Frank B Dean, John R Nelson, Theresa L Giesler, and Roger S Lasken. Rapid amplification of plasmid and phage dna using phi29 dna polymerase and multiply-primed rolling circle amplification. *Genome Research*, 11(6):1095–1099, 2001.

[65] David Y Zhang, Margaret Brandwein, Terence Hsuih, and Hong Bo Li. Ramification amplification: a novel isothermal dna amplification method. *Molecular Diagnosis*, 6(2):141–150, 2001.

[66] Frank B Dean, Seiyu Hosono, Linhua Fang, Xiaohong Wu, A Fawad Faruqi, Patricia Bray-Ward, Zhenyu Sun, Qiuling Zong, Yuefen Du, Jing Du, et al. Comprehensive human genome amplification using multiple displacement amplification. *Proceedings of the National Academy of Sciences*, 99(8):5261–5266, 2002.

[67] Roger S Lasken. Genomic dna amplification by the multiple displacement amplification (mda) method. *Biochemical Society Transactions*, 37(2):450–453, 2009.

[68] Angel J Picher, Bettina Budeus, Oliver Wafzig, Carola Krüger, Sara García-Gómez, María I Martínez-Jiménez, Alberto Díaz-Talavera, Daniela Weber, Luis Blanco, and Armin Schneider. Trueprime is a novel method for whole-genome amplification from single cells based on tth primpol. *Nature Communications*, 7(1):13296, 2016.

[69] Timour Baslan and James Hicks. Single cell sequencing approaches for complex biological systems. *Current Opinion in Genetics & Development*, 26:59–65, 2014.

[70] John P Langmore. Rubicon genomics, inc. *Pharmacogenomics*, 3(4):557–560, 2002.

[71] Chenghang Zong, Sijia Lu, Alec R Chapman, and X Sunney Xie. Genome-wide detection of single-nucleotide and copy-number variations of a single human cell. *Science*, 338(6114):1622–1626, 2012.

[72] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*. Springer, 2006.

[73] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[74] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[75] Judea Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan kaufmann, 1988.

[76] Steffen L Lauritzen and David J Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society: Series B (Methodological)*, 50(2):157–194, 1988.

[77] Phylogenetic tree - wikipedia — en.wikipedia.org. https://en.wikipedia.org/wiki/Phylogenetic_tree. [Accessed August 2, 2023].

[78] Chuan-Chao Wang, Hui-Yuan Yeh, Alexander N Popov, Hu-Qin Zhang, Hirofumi Matsumura, Kendra Sirak, Olivia Cheronet, Alexey Kovalev, Nadin Rohland, Alexander M Kim, et al. Genomic insights into the formation of human populations in east asia. *Nature*, 591(7850):413–419, 2021.

[79] Aaron P Ragsdale, Timothy D Weaver, Elizabeth G Atkinson, Eileen G Hoal, Marlo Möller, Brenna M Henn, and Simon Gravel. A weakly structured stem for human origins in africa. *Nature*, pages 1–9, 2023.

[80] Edward C Holmes, Gytis Dudas, Andrew Rambaut, and Kristian G Andersen. The evolution of ebola virus: Insights from the 2013–2016 epidemic. *Nature*, 538(7624):193–200, 2016.

[81] Stephen W Attwood, Sarah C Hill, David M Aanensen, Thomas R Connor, and Oliver G Pybus. Phylogenetic and phylodynamic approaches to understanding and combating the early sars-cov-2 pandemic. *Nature Reviews Genetics*, 23(9):547–562, 2022.

[82] Andrew Roth, Jaswinder Khattra, Damian Yap, Adrian Wan, Emma Laks, Justina Biele, Gavin Ha, Samuel Aparicio, Alexandre Bouchard-Côté, and Sohrab P Shah. Pyclone: statistical inference of clonal population structure in cancer. *Nature Methods*, 11(4):396–398, 2014.

[83] Hamim Zafar, Anthony Tzen, Nicholas Navin, Ken Chen, and Luay Nakhleh. Sifit: inferring tumor trees from single-cell sequencing data under finite-sites models. *Genome Biology*, 18(1):1–20, 2017.

[84] Russell D Gray, Alexei J Drummond, and Simon J Greenhill. Language phylogenies reveal expansion pulses and pauses in pacific settlement. *Science*, 323(5913):479–483, 2009.

[85] Remco Bouckaert, Philippe Lemey, Michael Dunn, Simon J Greenhill, Alexander V Alekseyenko, Alexei J Drummond, Russell D Gray, Marc A Suchard, and Quentin D Atkinson. Mapping the origins and expansion of the indo-european language family. *Science*, 337(6097):957–960, 2012.

[86] Joseph Felsenstein. *Inferring phylogenies*, volume 2. Sinauer associates Sunderland, MA, 2004.

[87] Dirk Husmeier, Richard Dybowski, and Stephen Roberts. *Probabilistic modeling in bioinformatics and medical informatics.* Springer Science & Business Media, 2006.

[88] Richard Durbin, Sean R Eddy, Anders Krogh, and Graeme Mitchison. *Biological sequence analysis: probabilistic models of proteins and nucleic acids.* Cambridge university press, 1998.

[89] James R Norris. *Markov chains.* Cambridge university press, 1998.

[90] Athanasios Papoulis and S Unnikrishna Pillai. Probability, random variables, and stochastic processes, 2002.

[91] Geoffrey Grimmett and David Stirzaker. *Probability and random processes.* Oxford university press, 2020.

[92] TH Jukes and CR Cantor. Evolution of protein molecules. in 'mammalian protein metabolism'.(ed. hn munro.) pp. 21–132. *Academic Press, New York)*, 1:504–511, 1969.

[93] Motoo Kimura. A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. *Journal of Molecular Evolution*, 16:111–120, 1980.

[94] Motoo Kimura. Estimation of evolutionary distances between homologous nucleotide sequences. *Proceedings of the National Academy of Sciences*, 78(1):454–458, 1981.

[95] Joseph Felsenstein. Evolutionary trees from dna sequences: A maximum likelihood approach. *Journal of Molecular Evolution*, 17(6):368–376, 11 1981.

[96] Masami Hasegawa, Hirohisa Kishino, and Taka-aki Yano. Dating of the human-ape splitting by a molecular clock of mitochondrial dna. *Journal of Molecular Evolution*, 22:160–174, 1985.

[97] Koichiro Tamura. Estimation of the number of nucleotide substitutions when there are strong transition-transversion and g+ c-content biases. *Molecular Biology and Evolution*, 9(4):678–687, 1992.

[98] Koichiro Tamura and Masatoshi Nei. Estimation of the number of nucleotide substitutions in the control region of mitochondrial dna in humans and chimpanzees. *Molecular Biology and Evolution*, 10(3):512–526, 1993.

[99] Simon Tavaré. Some probabilistic and statistical problems on the analysis of dna sequence. *Lecture of Mathematics for Life Science*, 17:57, 1986.

[100] Alexey Kozlov, Joao M Alves, Alexandros Stamatakis, and David Posada. Cellphy: accurate and fast probabilistic inference of single-cell phylogenies from scdna-seq data. *Genome Biology*, 23(1):1–30, 2022.

[101] Remco Bouckaert, Timothy G. Vaughan, Joëlle Barido-Sottani, Sebastián Duchêne, Mathieu Fourment, Alexandra Gavryushkina, Joseph Heled, Graham Jones, Denise Kühnert, Nicola De Maio, Michael Matschiner, Fábio K. Mendes, Nicola F. Müller, Huw A. Ogilvie, Louis du Plessis, Alex Popinga, Andrew Rambaut, David Rasmussen,

Igor Siveroni, Marc A. Suchard, Chieh-Hsi Wu, Dong Xie, Chi Zhang, Tanja Stadler, and Alexei J. Drummond. Beast 2.5: An advanced software platform for bayesian evolutionary analysis. *PLOS Computational Biology*, 15(4):1–28, 04 2019.

[102] Alexandros Stamatakis. Raxml version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics*, 30(9):1312–1313, 2014.

[103] Fredrik Ronquist, Maxim Teslenko, Paul Van Der Mark, Daniel L Ayres, Aaron Darling, Sebastian Höhna, Bret Larget, Liang Liu, Marc A Suchard, and John P Huelsenbeck. Mrbayes 3.2: efficient bayesian phylogenetic inference and model choice across a large model space. *Systematic Biology*, 61(3):539–542, 2012.

[104] Nicholas Metropolis and Stanislaw Ulam. The monte carlo method. *Journal of the American Statistical Association*, 44(247):335–341, 1949.

[105] Radford M Neal. Probabilistic inference using markov chain monte carlo methods. 1993.

[106] W Keith Hastings. Monte carlo sampling methods using markov chains and their applications. 1970.

[107] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953.

[108] Vivekananda Roy. Convergence diagnostics for markov chain monte carlo. *Annual Review of Statistics and Its Application*, 7:387–412, 2020.

[109] Steve Brooks, Andrew Gelman, Galin Jones, and Xiao-Li Meng. *Handbook of markov chain monte carlo*. CRC press, 2011.

[110] Andrew Gelman and Donald B Rubin. Inference from iterative simulation using multiple sequences. *Statistical Science*, 7(4):457–472, 1992.

[111] Mary Kathryn Cowles and Bradley P Carlin. Markov chain monte carlo convergence diagnostics: a comparative review. *Journal of the American Statistical Association*, 91(434):883–904, 1996.

[112] D Vats and C Knudson. Revisiting the gelman-rubin diagnostic. arxiv, 2018.

[113] Christian P Robert, George Casella, and George Casella. *Monte Carlo statistical methods*, volume 2. Springer, 1999.

[114] Dootika Vats, James M Flegal, and Galin L Jones. Multivariate output analysis for markov chain monte carlo. *Biometrika*, 106(2):321–337, 2019.

[115] Andrew Gelman, Walter R Gilks, and Gareth O Roberts. Weak convergence and optimal scaling of random walk metropolis algorithms. *The Annals of Applied Probability*, 7(1):110–120, 1997.

[116] Gareth O Roberts and Jeffrey S Rosenthal. Optimal scaling for various metropolis-hastings algorithms. *Statistical Science*, 16(4):351–367, 2001.

[117] Gareth O Roberts and Jeffrey S Rosenthal. Examples of adaptive mcmc. *Journal of Computational and Graphical Statistics*, 18(2):349–367, 2009.

[118] Gareth O Roberts and Jeffrey S Rosenthal. Coupling and ergodicity of adaptive markov chain monte carlo algorithms. *Journal of Applied Probability*, 44(2):458–475, 2007.

[119] Hamim Zafar, Yong Wang, Luay Nakhleh, Nicholas Navin, and Ken Chen. Monovar: single-nucleotide variant detection in single cells. *Nature Methods*, 13(6):505–507, 2016.

[120] Xiao Dong, Lei Zhang, Brandon Milholland, Moonsook Lee, Alexander Y Maslov, Tao Wang, and Jan Vijg. Accurate identification of single-nucleotide variants in whole-genome-amplified single cells. *Nature Methods*, 14(5):491–493, 2017.

[121] Lovelace J Luquette, Craig L Bohrson, Max A Sherman, and Peter J Park. Identification of somatic mutations in single cell dna-seq using a spatial model of allelic imbalance. *Nature Communications*, 10(1):1–14, 2019.

[122] Craig L Bohrson, Alison R Barton, Michael A Lodato, Rachel E Rodin, Lovelace J Luquette, Vinay V Viswanadham, Doga C Gulhan, Isidro Cortés-Ciriano, Maxwell A Sherman, Minseok Kwon, et al. Linked-read analysis identifies mutations in single-cell dna-sequencing data. *Nature Genetics*, 51(4):749–754, 2019.

[123] David Lähnemann, Johannes Köster, Ute Fischer, Arndt Borkhardt, Alice C McHardy, and Alexander Schönhuth. Accurate and scalable variant calling from single cell dna sequencing data with prosolo. *Nature Communications*, 12(1):1–11, 2021.

[124] Ke Yuan, Thomas Sakoparnig, Florian Markowetz, and Niko Beerenwinkel. Bitphylogeny: a probabilistic framework for reconstructing intra-tumor phylogenies. *Genome Biology*, 16(1):1–16, 2015.

[125] Edith M Ross and Florian Markowetz. Onconem: inferring tumor evolution from single-cell sequencing data. *Genome Biology*, 17(1):1–14, 2016.

[126] Katharina Jahn, Jack Kuipers, and Niko Beerenwinkel. Tree inference for single-cell data. *Genome Biology*, 17(1):1–17, 2016.

[127] Salem Malikic, Katharina Jahn, Jack Kuipers, S Cenk Sahinalp, and Niko Beerenwinkel. Integrative inference of subclonal tumour evolution from single-cell and bulk sequencing data. *Nature Communications*, 10(1):1–12, 2019.

[128] Alexey M Kozlov, Diego Darriba, Tomáš Flouri, Benoit Morel, and Alexandros Stamatakis. Raxml-ng: a fast, scalable and user-friendly tool for maximum likelihood phylogenetic inference. *Bioinformatics*, 35(21):4453–4455, 2019.

[129] Hamim Zafar, Nicholas Navin, Ken Chen, and Luay Nakhleh. Siclonefit: Bayesian inference of population structure, genotype, and phylogeny of tumor clones from single-cell genome sequencing data. *Genome Research*, 29(11):1847–1859, 2019.

[130] Jochen Singer, Jack Kuipers, Katharina Jahn, and Niko Beerenwinkel. Single-cell mutation identification via phylogenetic inference. *Nature Communications*, 9(1):5144, 12 2018.

[131] Jack Kuipers, Jochen Singer, and Niko Beerenwinkel. Single-cell mutation calling and phylogenetic tree reconstruction with loss and recurrence. *Bioinformatics*, 08 2022. btac577.

[132] Jack Kuipers, Katharina Jahn, Benjamin J Raphael, and Niko Beerenwinkel. Single-cell sequencing data reveal widespread recurrence and loss of mutational hits in the life histories of tumors. *Genome Research*, 27(11):1885–1894, 2017.

[133] Jonas Demeulemeester, Stefan C. Dentro, Moritz Gerstung, and Peter Van Loo. Biallelic mutations in cancer genomes reveal local mutational determinants. *Nature Genetics*, 54(2):128–133, Feb 2022.

[134] Paul O. Lewis. A Likelihood Approach to Estimating Phylogeny from Discrete Morphological Character Data. *Systematic Biology*, 50(6):913–925, 11 2001.

[135] Adam D. Leaché, Barbara L. Banbury, Joseph Felsenstein, Adrián nieto-Montes de Oca, and Alexandros Stamatakis. Short Tree, Long Tree, Right Tree, Wrong Tree: New Acquisition Bias Corrections for Inferring SNP Phylogenies. *Systematic Biology*, 64(6):1032–1047, 07 2015.

[136] Senbai Kang, Nico Borgsmüller, Monica Valecha, Magda Markowska, Jack Kuipers, Niko Beerenwinkel, David Posada, and Ewa Szczurek. Delsieve: joint inference of single-nucleotide variants, somatic deletions, and cell phylogeny from single-cell dna sequencing data. *bioRxiv*, 2023.

[137] Mel Greaves. Evolutionary Determinants of Cancer. *Cancer Discovery*, 5(8):806–820, 08 2015.

[138] Stefan C. Dentro, Ignaty Leshchiner, Kerstin Haase, Maxime Tarabichi, Jeff Wintersinger, Amit G. Deshwar, Kaixian Yu, Yulia Rubanova, Geoff Macintyre, Jonas Demeulemeester, Ignacio Vázquez-García, Kortine Kleinheinz, Dimitri G. Livitz, Salem Malikic, Nilgun Donmez, Subhajit Sengupta, Pavana Anur, Clemency Jolly, Marek Cmero, Daniel Rosebrock, Steven E. Schumacher, Yu Fan, Matthew Fittall, Ruben M. Drews, Xiaotong Yao, Thomas B.K. Watkins, Juhee Lee, Matthias Schlesner, Hongtu Zhu, David J. Adams, Nicholas McGranahan, Charles Swanton, Gad Getz, Paul C. Boutros, Marcin Imielinski, Rameen Beroukhim, S. Cenk Sahinalp, Yuan Ji, Martin Peifer, Inigo Martincorena, Florian Markowetz, Ville Mustonen, Ke Yuan, Moritz Gerstung, Paul T. Spellman, Wenyi Wang, Quaid D. Morris, David C. Wedge, Peter Van Loo, Stefan C. Dentro, Ignaty Leshchiner, Moritz Gerstung, Clemency Jolly, Kerstin Haase, Maxime Tarabichi, Jeff Wintersinger, Amit G. Deshwar, Kaixian Yu, Santiago Gonzalez, Yulia Rubanova, Geoff Macintyre, Jonas Demeulemeester, David J. Adams, Pavana Anur, Rameen Beroukhim, Paul C. Boutros, David D. Bowtell, Peter J. Campbell, Shaolong Cao, Elizabeth L. Christie, Marek Cmero, Yupeng Cun, Kevin J. Dawson, Nilgun Donmez, Ruben M. Drews, Roland Eils, Yu Fan, Matthew Fittall, Dale W. Garsed, Gad Getz, Gavin Ha, Marcin Imielinski, Lara Jerman, Yuan Ji, Kortine Kleinheinz, Juhee Lee, Henry Lee-Six, Dimitri G. Livitz, Salem Malikic, Florian Markowetz, Inigo Martincorena, Thomas J. Mitchell, Ville Mustonen, Layla Oesper, Martin Peifer, Myron Peto, Benjamin J. Raphael, Daniel Rosebrock, S. Cenk Sahinalp, Adriana Salcedo, Matthias Schlesner, Steven E. Schumacher, Subhajit Sengupta, Ruian Shi, Seung Jun Shin, Lincoln D. Stein, Oliver Spiro, Ignacio Vázquez-García, Shankar Vembu, David A. Wheeler, Tsun-Po Yang, Xiaotong Yao, Ke Yuan, Hongtu Zhu, Wenyi Wang, Quaid D. Morris, Paul T. Spellman, David C. Wedge, and Peter Van Loo. Characterizing genetic intra-tumor heterogeneity across 2,658 human cancer genomes. *Cell*, 184(8):2239–2254.e39, 2021.

[139] Moritz Gerstung, Christian Beisel, Markus Rechsteiner, Peter Wild, Peter Schraml, Holger Moch, and Niko Beerenwinkel. Reliable detection of subclonal single-nucleotide variants in tumour cell populations. *Nature Communications*, 3(1):1–8, 2012.

[140] Sohrab P Shah, Andrew Roth, Rodrigo Goya, Arusha Oloumi, Gavin Ha, Yongjun Zhao, Gulisa Turashvili, Jiarui Ding, Kane Tse, Gholamreza Haffari, et al. The clonal and mutational evolution spectrum of primary triple-negative breast cancers. *Nature*, 486(7403):395–399, 2012.

[141] Gavin Ha, Andrew Roth, Jaswinder Khattra, Julie Ho, Damian Yap, Leah M Prentice, Nataliya Melnyk, Andrew McPherson, Ali Bashashati, Emma Laks, et al. Titan: inference of copy number architectures in clonal cell populations from tumor whole-genome sequence data. *Genome Research*, 24(11):1881–1893, 2014.

[142] Amit G Deshwar, Shankar Vembu, Christina K Yung, Gun Ho Jang, Lincoln Stein, and Quaid Morris. Phylowgs: reconstructing subclonal composition and evolution from whole-genome sequencing of tumors. *Genome Biology*, 16(1):1–20, 2015.

[143] Tanja Stadler, Oliver G Pybus, and Michael PH Stumpf. Phylodynamics for cell biologists. *Science*, 371(6526):eaah6266, 2021.

[144] Ziheng Yang. Among-site rate variation and its impact on phylogenetic analyses. *Trends in Ecology & Evolution*, 11(9):367–372, 1996.

[145] Simon Anders and Wolfgang Huber. Differential expression analysis for sequence count data. *Genome Biology*, 11(10):R106, 10 2010.

[146] Joseph Felsenstein. Phylogenies from restriction sites: a maximum-likelihood approach. *Evolution*, 46(1):159–173, 1992.

[147] Alexei J. Drummond, Geoff K. Nicholls, Allen G. Rodrigo, and Wiremu Solomon. Estimating mutation parameters, population history and genealogy simultaneously from temporally spaced sequence data. *Genetics*, 161(3):1307–1320, 2002.

[148] Joseph E O'Reilly and Philip CJ Donoghue. The efficacy of consensus tree methods for summarizing phylogenetic relationships from a posterior sample of trees estimated from morphological data. *Systematic Biology*, 67(2):354–362, 2018.

[149] Alexei J Drummond, Simon Y. W Ho, Matthew J Phillips, and Andrew Rambaut. Relaxed phylogenetics and dating with confidence. *PLOS Biology*, 4(5):null, 03 2006.

[150] Alexei J. Drummond and Marc A. Suchard. Bayesian random local clocks, or one rate to rule them all. *BMC Biology*, 8(1):114, Aug 2010.

[151] David Posada. CellCoal: Coalescent Simulation of Single-Cell Sequencing Samples. *Molecular Biology and Evolution*, 37(5):1535–1542, 02 2020.

[152] Johannes Köster and Sven Rahmann. Snakemake—a scalable bioinformatics workflow engine. *Bioinformatics*, 28(19):2520–2522, 08 2012.

[153] M K Kuhner and J Felsenstein. A simulation comparison of phylogeny algorithms under equal and unequal evolutionary rates. *Molecular Biology and Evolution*, 11(3):459–468, 05 1994.

[154] D.F. Robinson and L.R. Foulds. Comparison of phylogenetic trees. *Mathematical Biosciences*, 53(1):131–147, 1981.

[155] Klaus Schliep, Alastair J. Potts, David A. Morrison, and Guido W. Grimm. Intertwining phylogenetic trees and networks. *Methods in Ecology and Evolution*, 8(10):1212–1220, 2017.

[156] Jordan Douglas, Rong Zhang, and Remco Bouckaert. Adaptive dating and fast proposals: Revisiting the phylogenetic relaxed clock model. *PLOS Computational Biology*, 17(2):1–30, 02 2021.

[157] Kai Wang, Mingyao Li, and Hakon Hakonarson. ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data. *Nucleic Acids Research*, 38(16):e164–e164, 07 2010.

[158] Guangchuang Yu, David K. Smith, Huachen Zhu, Yi Guan, and Tommy Tsan-Yuk Lam. ggtree: an r package for visualization and annotation of phylogenetic trees with their covariates and other associated data. *Methods in Ecology and Evolution*, 8(1):28–36, 2017.

[159] Zuguang Gu, Roland Eils, and Matthias Schlesner. Complex heatmaps reveal patterns and correlations in multidimensional genomic data. *Bioinformatics*, 32(18):2847–2849, 05 2016.

[160] Dongdong Huang, Wenjie Sun, Yuwei Zhou, Peiwei Li, Fang Chen, Hanwen Chen, Dajing Xia, Enping Xu, Maode Lai, Yihua Wu, et al. Mutations of key driver genes in colorectal cancer progression and metastasis. *Cancer and Metastasis Reviews*, 37(1):173–187, 2018.

[161] Hans Raskov, Jacob H Søby, Jesper Troelsen, Rasmus D Bojesen, and Ismail Gögenur. Driver gene mutations and epigenetics in colorectal cancer. *Annals of Surgery*, 271(1):75–85, 2020.

[162] T. Müller, U. Stein, A. Poletti, L. Garzia, M. Rothley, D. Plaumann, W. Thiele, M. Bauer, A. Galasso, P. Schlag, M. Pankratz, M. Zollo, and J. P. Sleeman. Asap1 promotes tumor cell motility and invasiveness, stimulates metastasis formation in vivo, and correlates with poor survival in colorectal cancer patients. *Oncogene*, 29(16):2393–2403, Apr 2010.

[163] Meng-Shun Sun, Lan-Ting Yuan, Chia-Hao Kuei, Hui-Yu Lin, Yen-Lin Chen, Hui-Wen Chiu, and Yuan-Feng Lin. Rgl2 drives the metastatic progression of colorectal cancer via preventing the protein degradation of β-catenin and kras. *Cancers*, 13(8), 2021.

[164] Alan D. D'Andrea. 4 - dna repair pathways and human cancer. In John Mendelsohn, Joe W. Gray, Peter M. Howley, Mark A. Israel, and Craig B. Thompson, editors, *The Molecular Basis of Cancer (Fourth Edition)*, pages 47–66.e2. W.B. Saunders, Philadelphia, fourth edition edition, 2015.

[165] Kristian Cibulskis, Michael S Lawrence, Scott L Carter, Andrey Sivachenko, David Jaffe, Carrie Sougnez, Stacey Gabriel, Matthew Meyerson, Eric S Lander, and Gad Getz. Sensitive detection of somatic point mutations in impure and heterogeneous cancer samples. *Nature Biotechnology*, 31(3):213–219, 2013.

[166] David Jones, Keiran M Raine, Helen Davies, Patrick S Tarpey, Adam P Butler, Jon W Teague, Serena Nik-Zainal, and Peter J Campbell. cgpcavemanwrapper: simple execution of caveman in order to detect somatic single nucleotide variants in ngs data. *Current Protocols in Bioinformatics*, 56(1):15–10, 2016.

[167] Yu Fan, Liu Xi, Daniel ST Hughes, Jianjun Zhang, Jianhua Zhang, P Andrew Futreal, David A Wheeler, and Wenyi Wang. Muse: accounting for tumor heterogeneity using a sample-specific error model improves sensitivity and specificity in mutation calling from sequencing data. *Genome Biology*, 17(1):1–11, 2016.

[168] Zhongwu Lai, Aleksandra Markovets, Miika Ahdesmaki, Brad Chapman, Oliver Hofmann, Robert McEwen, Justin Johnson, Brian Dougherty, J Carl Barrett, and Jonathan R Dry. Vardict: a novel and versatile variant caller for next-generation sequencing in cancer research. *Nucleic Acids Research*, 44(11):e108–e108, 2016.

[169] Sangtae Kim, Konrad Scheffler, Aaron L Halpern, Mitchell A Bekritsky, Eunho Noh, Morten Källberg, Xiaoyu Chen, Yeonbin Kim, Doruk Beyter, Peter Krusche, et al. Strelka2: fast and accurate calling of germline and somatic variants. *Nature Methods*, 15(8):591–594, 2018.

[170] Daniel C Koboldt, Qunyuan Zhang, David E Larson, Dong Shen, Michael D McLellan, Ling Lin, Christopher A Miller, Elaine R Mardis, Li Ding, and Richard K Wilson. Varscan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome Research*, 22(3):568–576, 2012.

[171] Gavin Ha, Andrew Roth, Daniel Lai, Ali Bashashati, Jiarui Ding, Rodrigo Goya, Ryan Giuliany, Jamie Rosner, Arusha Oloumi, Karey Shumansky, et al. Integrative analysis of genome-wide loss of heterozygosity and monoallelic expression at nucleotide resolution reveals disrupted pathways in triple-negative breast cancer. *Genome Research*, 22(10):1995–2007, 2012.

[172] Lei Bao, Minya Pu, and Karen Messer. Abscn-seq: a statistical method to estimate tumor purity, ploidy and absolute copy numbers from next-generation sequencing data. *Bioinformatics*, 30(8):1056–1063, 2014.

[173] Francesco Favero, Tejal Joshi, Andrea Marion Marquard, Nicolai Juul Birkbak, Marcin Krzystanek, Qiyuan Li, Z Szallasi, and Aron Charles Eklund. Sequenza: allele-specific copy number and mutation profiles from tumor sequencing data. *Annals of Oncology*, 26(1):64–70, 2015.

[174] Andrew McPherson, Andrew Roth, Emma Laks, Tehmina Masud, Ali Bashashati, Allen W Zhang, Gavin Ha, Justina Biele, Damian Yap, Adrian Wan, et al. Divergent modes of clonal spread and intraperitoneal mixing in high-grade serous ovarian cancer. *Nature Genetics*, 48(7):758–767, 2016.

[175] Gryte Satas, Simone Zaccaria, Geoffrey Mon, and Benjamin J Raphael. Scarlet: single-cell tumor phylogeny inference with copy-number constrained mutation losses. *Cell Systems*, 10(4):323–332, 2020.

[176] R Core Team. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria, 2023.

[177] Indrajeet Patil. Visualizations with statistical details: The 'ggstatsplot' approach. *Journal of Open Source Software*, 6(61):3167, 2021.

[178] Wen-Liang Gao, Lei Niu, Wei-Ling Chen, Yong-Qu Zhang, and Wen-He Huang. Integrative analysis of the expression levels and prognostic values for nek family members in breast cancer. *Frontiers in Genetics*, 13:798170, 2022.

[179] Simone Bersini, Nikki K Lytle, Roberta Schulte, Ling Huang, Geoffrey M Wahl, and Martin W Hetzer. Nup93 regulates breast tumor growth by modulating cell proliferation and actin cytoskeleton remodeling. *Life Science Alliance*, 3(1), 2020.

[180] Christina Klaus, Ursula Schneider, Christian Hedberg, Anke K Schütz, Jürgen Bernhagen, Herbert Waldmann, Nikolaus Gassler, and Elke Kaemmerer. Modulating effects of acyl-coa synthetase 5-derived mitochondrial wnt2b palmitoylation on intestinal wnt activity. *World Journal of Gastroenterology: WJG*, 20(40):14855, 2014.

[181] Woo Suk Hong, Max Shpak, and Jeffrey P Townsend. Inferring the origin of metastases from cancer phylogenies. *Cancer Research*, 75(19):4021–4025, 2015.

[182] Marco Gerlinger, Andrew J Rowan, Stuart Horswell, James Larkin, David Endesfelder, Eva Gronroos, Pierre Martinez, Nicholas Matthews, Aengus Stewart, Patrick Tarpey, et al. Intratumor heterogeneity and branched evolution revealed by multiregion sequencing. *New England Journal of Medicine*, 366(10):883–892, 2012.

[183] Zheng Hu, Jie Ding, Zhicheng Ma, Ruping Sun, Jose A Seoane, J Scott Shaffer, Carlos J Suarez, Anna S Berghoff, Chiara Cremolini, Alfredo Falcone, et al. Quantitative evidence for early metastatic seeding in colorectal cancer. *Nature Genetics*, 51(7):1113–1122, 2019.