Genome Biology

METHODOLOGY

Open Access

Phylogenetic tree inference from single-cell RNA sequencing data with SCITE-RNA

Norio Zimmermann[1,2†], Xiaoyu Sun[1,2†], Joanna Hård[1,2,3], Jack Kuipers[1,2*] and Niko Beerenwinkel[1,2*]

†Norio Zimmermann and Xiaoyu Sun contributed equally to this work.

*Correspondence: jack.kuipers@bsse.ethz.ch; niko.beerenwinkel@bsse.ethz.ch

[1] Department of Biosystems Science and Engineering, ETH Zurich, Basel 4056, Switzerland
[2] SIB Swiss Institute of Bioinformatics, Basel 4056, Switzerland
[3] KTH Royal Institute of Technology, Stockholm 100 44, Sweden

Abstract

We present SCITE-RNA, a novel phylogenetic tree inference method designed for single-cell RNA sequencing data which takes reference and alternative read counts of single-nucleotide variants as input. Our approach uses a maximum-likelihood random-scan greedy search that alternates between cell lineage tree and mutation tree representations to escape local optima until convergence is achieved in both. We demonstrate superior performance on simulated data compared to existing methods. Furthermore, we show its applicability to cancer single-cell RNA sequencing data, where it allows us to link evolutionary trajectories of cells to their gene expression profiles.

Keywords: Phylogenetic tree inference, Tumor evolution, Single-cell RNA sequencing

Background

Recent advancements in sequencing technologies have led to an increase in the availability of single-cell sequencing data and the development of new computational tools [10]. In contrast to conventional methods such as bulk sequencing, single-cell sequencing technologies provide higher resolution for analyzing the heterogeneity of cell populations. In single-cell RNA sequencing (scRNA-seq), amplified complementary DNA (cDNA) obtained by reverse transcription of RNA is sequenced [25]. Since the transcriptome is transcribed from the genome, scRNA-seq can indirectly provide genetic information about the expressed genomic regions, along with providing insights into the gene expression profiles of cells [10]. It is therefore, in principle, also informative about the evolutionary relationships among cells. Phylogenetic inference aims to reconstruct these evolutionary histories from observed heritable traits, such as genetic variations.

For single-cell DNA sequencing data, which directly measures genetic variation in the DNA, various phylogenetic tree inference methods have been developed [12, 15, 16, 18–20, 31–33]. However, its adoption remains limited due to high costs [21]. More commonly used are bulk DNA sequencing, as well as scRNA-seq [21]. scRNA-seq technologies include full-length protocols such as Smart-seq2 [25], and

high-throughput 3'-end methods [39] as offered by 10x Genomics. Full-length methods typically offer greater sensitivity for gene detection, with lower technical noise, higher coverage and fewer dropout events compared to 3'-end protocols, which additionally suffer from 3'-end coverage biases [13, 37]. Nonetheless, 3'-end methods are advantageous for detecting rare cell populations due to their higher cell throughput [37].

Most scRNA-seq-based phylogenetic tree reconstruction methods rely on single-nucleotide variants (SNVs), and therefore benefit from the aforementioned advantages of full-length technologies like Smart-seq2 compared to 3'-end methods. scRNA-seq tree inference methods include DENDRO [40], SClineager [23], PhylinSic [22], Canopy2 [38], and PhylEx [14]. Both DENDRO and SClineager use alternative and reference allele read counts of transcribed SNVs as input. DENDRO then computes a pairwise genetic divergence matrix between cells and clusters the cells based on this matrix into clones [40]. SClineager aims to overcome drop-out by using a Bayesian hierarchical model with Markov Chain Monte Carlo (MCMC) sampling to infer the true variant allele frequencies of the cells. These are then used to build clonal trees by hierarchical clustering methods [23]. PhylinSic calls SNVs and infers a nucleotide sequence from the read counts of the mutated loci before tree inference. Then it uses BEAST2 [1] models to infer phylogenetic cell lineage trees [22]. Both PhylEx and Canopy2 integrate bulk DNA data with scRNA-seq data to reconstruct clonal trees. Phylex [14] uses a tree-structured stick breaking process and Canopy2 [38] a Bayesian hierarchical model with MCMC sampling.

The existing methods have several limitations. SClineager and DENDRO rely on clustering cells into a limited number of clones, which may lack resolution and not accurately reflect the true phylogenetic structure. By calling genotypes from reference and alternative read counts before reconstructing the phylogeny, PhylinSic does not fully account for the read count information during tree inference. PhylEx's and Canopy2's use of bulk DNA data poses limitations when only scRNA-seq data is available. Another common problem of tree optimization methods that explore the tree space, such as PhylinSic, PhylEx, and Canopy2, is their tendency to become trapped in local optima, which might be difficult to escape from.

In this paper, we present **S**ingle-**C**ell **I**nference of **T**umor **E**volution from **RNA** sequencing data (SCITE-RNA), a tree inference method for scRNA-seq data. SCITE-RNA takes reference and alternative read counts of SNVs called from scRNA-seq data, selects potentially mutated loci, and reconstructs a phylogenetic tree of the sequenced cells. In the inference we greedily optimize over cell lineage and mutation trees, by alternating between these two tree spaces. This approach is fast and has the ability to escape local optima, because a simple tree move in the mutation space, such as reattaching a subtree at a new location, can lead to major changes in the structure of the cell lineage tree and vice versa. Using SCITE-RNA we address some of the limitations of existing approaches: (1) SCITE-RNA does not rely on additional data sources like bulk DNA sequencing, (2) it does not assume a certain number of clones in advance, (3) it specifically addresses the problem of getting trapped in local optima, and (4) it can estimate important parameters, such as the dropout probability, directly from the data.

We show superior performance of SCITE-RNA on simulated data for difficult tree reconstruction problems involving a large number of clones. Additionally, we demonstrate its ability to reconstruct plausible phylogenies on several real cancer scRNA-seq datasets.

Results

We first present a short overview of the SCITE-RNA method, followed by a comparative evaluation on both simulated and real scRNA-seq datasets. We benchmark our model against existing phylogenetic tree inference approaches for scRNA-seq data.

Method overview

SCITE-RNA (Fig. 1) takes as input reference and alternative read counts from SNVs across individual cells. For each cell–mutation pair, it computes a likelihood based on a model of scRNA-seq data involving the observed read counts, the inferred genotype and model parameters such as the dropout rate. Using these likelihoods, the algorithm obtains the likelihood of any given mutation or cell lineage tree. For cell lineage trees, nodes represent individual cells and the binary tree structure captures the evolutionary relationships between them. For mutation trees, nodes correspond to mutations, and the tree encodes the order in which mutations occurred. We do not model branch lengths for either tree type. SCITE-RNA employs a greedy optimization strategy that iteratively explores the tree spaces via random pruning and reattachment operations. Since this process is stochastic, the inferred tree and its likelihood can vary from run to run. After each optimization round, the tree is converted to its alternative representation, either from a cell lineage tree to a mutation tree or vice versa (Fig. 2). This iterative process of tree optimization and space switching is continued until the likelihood reaches an optimum in both spaces. Finally, based on the inferred genotypes obtained

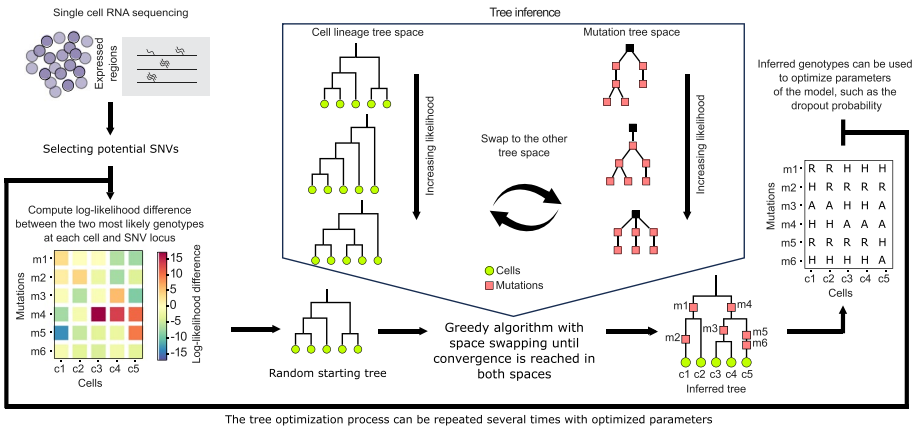

Fig. 1 SCITE-RNA method overview. First, RNA of single cells is sequenced, followed by the selection of potentially mutated loci. We then obtain a likelihood for the read counts of each cell–mutation pair given the inferred genotypes and hence a joint likelihood for any given cell lineage tree or mutation tree. To find the maximum likelihood tree, we explore the tree space greedily with simple tree moves. Once a set of moves is completed, the tree is converted into the alternative space, where the exploration continues. This process is repeated until the tree hits a local optimum in both spaces. Using the genotypes inferred by the final tree, we optimize the model parameters to better fit a specific dataset and continue with the second round of tree optimization

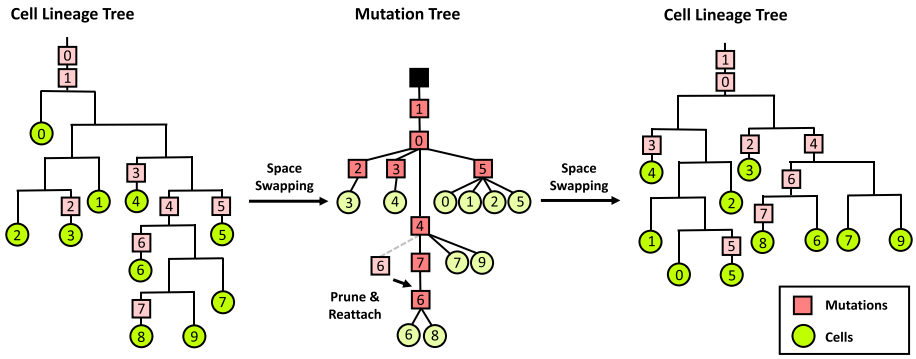

Fig. 2 Swapping between tree spaces. The cell lineage tree on the left has been optimized by pruning and reattaching subtrees of cells. It is converted into a mutation tree representation (middle). Mutations placed on the same edge are assigned a random order during the tree space swapping. A simple move in the mutation tree space, for example, moving a single mutation (No. 6) to a new location, increases the likelihood of the tree. Swapping back to the cell lineage tree representation, the simple move in the mutation tree, combined with random sampling among equivalent tree conversions, has resulted in significant changes to the cell lineage tree

from the optimized tree, SCITE-RNA refines model parameters such as the dropout rate and overdispersion parameters. They influence the probability of observing specific reference and alternative read counts given a genotype and can vary across datasets or even across individual SNVs. The parameter optimization thus also enables the model to better account for RNA-specific phenomena, such as monoallelic expression and imbalanced biallelic expression of different genes. The updated parameters are then used in a second round of tree inference since the optimal tree structure may depend on the model parameters. The output of SCITE-RNA composes both the tumor single-cell phylogeny and the inferred genotypes (Fig. 1). A complete description of the methodology can be found in the Methods section.

Performance evaluation on simulated data

Comparison of tree space optimization strategies

To compare optimization in different tree spaces (Fig. 3), we conducted a simulation study using three distinct configurations of cells and SNVs: more SNVs (100 cells, 500 SNVs), balanced (500 cells, 500 SNVs), and more cells (500 cells, 100 SNVs). We simulate 100 random binary cell lineage trees with randomly placed mutations in each. Tree optimization is compared across four approaches: optimizing only in cell lineage tree space (c), only in mutation tree space (m), and alternating between the two, starting in either cell tree space (cm) or mutation tree space (mc).

We compare the likelihood of the optimal trees to that of the ground truth trees, normalized by the number of cells and SNVs. Values greater than 0 indicate that the inferred trees have a higher likelihood than the ground truth trees, which is possible due to finite sample noise.

As shown in Fig. 3, SCITE-RNA finds high-likelihood trees for all approaches, with the exception of solely optimizing in the mutation tree space (m) for larger numbers of SNVs. The benefit of switching between tree spaces becomes more pronounced when

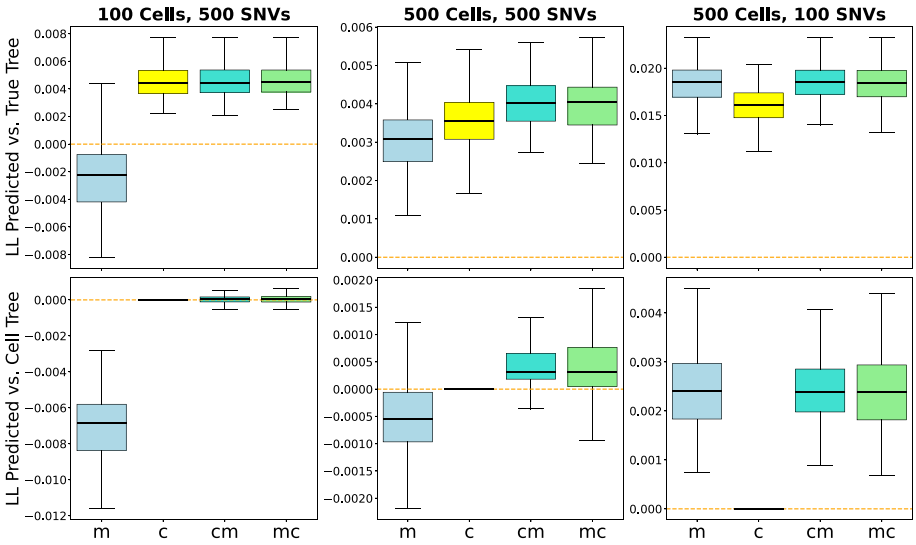

Fig. 3 Comparison of tree optimization strategies across different configurations of cells and SNVs. The upper row shows the normalized log-likelihood (LL) difference between the optimized trees and the ground truth trees. Values above 0 indicate, that the optimized trees have a higher likelihood than the true trees. The lower row shows the normalized log-likelihoods after subtracting the log-likelihoods of the trees optimized solely in the cell lineage tree space. This highlights the differences in achieved optimal log-likelihood for individual runs. Tree optimization is compared across four approaches: optimizing only in cell lineage tree space (c), only in mutation tree space (m), and alternating between the two, starting with either cell tree space (cm) or mutation tree space (mc)

the number of cells is similar to the number of SNVs. When the number of SNVs is larger than the number of cells, optimizing the cell tree space is more effective due to its smaller size. The opposite, but less pronounced effect is observed, when the number of SNVs is smaller than the number of cells.

For an equal number of cells and SNVs, we observe that the likelihood, when switching between tree spaces, is usually higher compared to optimizing in a single tree space. When averaging across multiple runs, tree inference with space switching consistently performs at least as well as, and often superior to, the best results obtained by optimization in either tree space alone. This trend holds robustly across all tested combinations of cell and mutation counts.

Comparison with existing methods

We compare the performance of SCITE-RNA to DENDRO [40], SClineager [23] and PhylinSic [22] on simulated data. Like SCITE-RNA, both SClineager and DENDRO work with reference and alternative allele counts derived from scRNA-seq data. To run PhylinSic, we slightly adapt their pipeline to call genotypes and infer a maximum clade credibility tree on the simulated reference and alternative read counts (see Additional file 1: Section A.1 for more details).

The evaluation focuses on the accuracy of predicted genotypes and the similarity of the inferred tree structure to the true tree. We quantify these aspects by calculating the mean absolute error of the predicted variant allele frequencies, the Path Difference between the ground truth and predicted trees (adapted from [34]; see Additional file 1: Section A.2 for further details), and the normalized Robinson-Foulds distance [28]. We

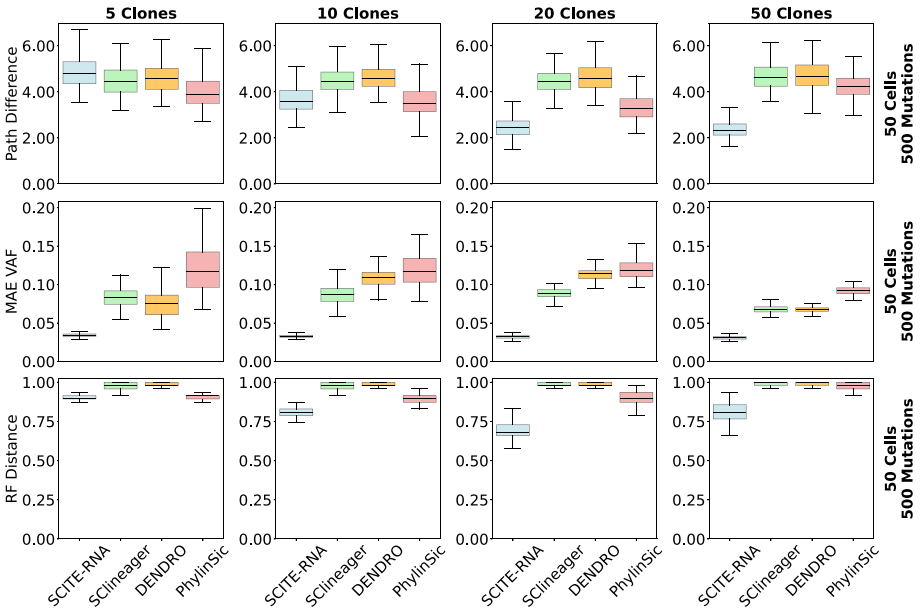

Fig. 4 Comparison of SCITE-RNA to DENDRO, SClineager and PhylinSic based on the Path Difference, the mean absolute error of the predicted variant allele frequencies (MAE VAF) and the normalized Robinson-Foulds (RF) Distance. The datasets simulate a variable number of clones with an increasing clonal complexity from left to right. SCITE-RNA outperforms the other methods in tree reconstruction for datasets with a higher number of clones and demonstrates superior accuracy in inferring true variant allele frequencies across all settings

simulate trees with 50 cells and 500 SNVs, along with a variable number of clones, representing cells with the same genotype (Fig. 4). We chose a relatively small number of cells and large number of SNVs, as this configuration is similar to the real datasets we explore below. We generate random cell lineage trees with 5, 10, 20, and 50 clones. For each number of clones, we simulate 100 trees with reference and alternative read counts for each cell–locus pair (see Additional file 1: Section A.1 for more details).

As shown in Fig. 4, SCITE-RNA clearly and consistently outperforms the other methods in inferring the genotypes in all settings, while it reconstructs more accurate tree structures in most settings. In contrast, the genotype calling and smoothing procedure of PhylinSic results in poor genotype predictions on our simulated datasets. The Robinson-Foulds distance shows a benefit of using SCITE-RNA in most settings, only for five clones it is comparable to PhylinSic. The benefit of using SCITE-RNA for tree inference increases with increased clonal complexity. However, for very small numbers of clones, the Path Difference to the true tree is slightly smaller for DENDRO, SClineager and PhylinSic. This worse performance for small numbers of clones is because SCITE-RNA assumes that SNVs/cells are independently placed on the tree to factorize the likelihood, an assumption that does not hold when only a few clones are present. Combined with the noisy nature of the data, this leads to overfitting in the SCITE-RNA trees, as our model does not impose any constraints on the number of clones.

As SClineager and DENDRO use clustering to generate the tree structure, we also provide a comparison of our method using the same hierarchical clustering approach based on the predicted variant allele frequencies (see Additional file 1: Section A.3). With clustering, SCITE-RNA outperforms SClineager and DENDRO even for small numbers of

clones, in terms of both tree reconstruction and cell-to-clone assignment (Additional file 1: Fig. S1). For larger numbers of clones, however, SCITE-RNA without clustering achieves a larger improvement in performance over DENDRO and SClineager compared to the clustered approach (Additional file 1: Fig. S2).

Another approach to reduce overfitting a particular dataset is to generate a consensus tree from a set of bootstrapped trees using DendroPy [35] (see Additional file 1: Section A.3). This improves both tree reconstruction and genotype prediction, especially at low clonal complexity (Additional file 1: Fig. S2). For moderate and higher number of clones, the consensus tree outperforms the clustered tree, though the improvement over standard SCITE-RNA is modest. The runtime, however, increases linearly with the number of bootstrap samples, as a tradeoff for the possible improvements in accuracy. We apply this approach on the smaller real datasets as well, as it yields more stable results than individual runs (see Application to cancer data section). For larger trees, the low number and small size of matching clades (Additional file 1: Fig. S3), makes the building of maximum clade credibility trees from bootstrap samples difficult.

When varying simulation parameters and including additional mutation events like homoplasy and copy number alterations we observe a slight deterioration in performance for SCITE-RNA for higher levels of noise, lower coverage and a larger fraction of loci affected by additional mutation events. The impact of copy number alterations and homoplasy on the tree reconstruction remains limited, while genotype inference is more sensitive to these perturbations. However, SCITE-RNA consistently outperforms the alternative methods across all tested settings (see Additional file 1: Section A.4 and Figs. S4-S6).

SCITE-RNA maintains its superior performance on larger datasets with 1 million cell–mutation entries in the following configurations: 5,000 cells and 200 mutations, 2,000 cells and 500 mutations, 1,000 cells and 1,000 mutations, 500 cells and 2,000 mutations, and 200 cells and 5,000 mutations (see Additional file 1: Section A.5 and Fig. S7).

Regarding runtime, PhylinSic requires the most computational time in most settings, taking about 540 seconds on average to run the tree inference for 100 cells and 100 mutations, followed by SClineager with about 50 seconds for the same dataset (see Additional file 1: Fig. S8). DENDRO is the fastest finishing in less than a second even for datasets with a couple hundred cells and mutations. On similar datasets it takes a few seconds to run SCITE-RNA with two optimization rounds. On large datasets with thousands of cells or mutations, SCITE-RNA is still the second fastest model with a runtime of about an hour for a dataset with 1,000 cells and 1,000 mutations, while SClineager and PhylinSic both take more than a day to finish. SCITE-RNA therefore offers the best overall performance, at a relatively low computational cost compared to models using MCMC such as SClineager and PhylinSic.

Application to cancer data

We applied our model to two scRNA-seq datasets from multiple myeloma patients, with identifiers MM16 and MM34 [8, 30]. The MM34 dataset includes 127 cells collected via bone marrow biopsy from the primary tumor and metastatic cells obtained from an ascites punctuation after two months of unsuccessful treatment [8, 30]. The MM16 dataset contains 46 cells sampled from a bone marrow biopsy at diagnosis and again

six months later during a minimal residual disease state following chemotherapy. Additionally, we demonstrate the scalability of our approach by applying it to a glioblastoma dataset (BT_S2), which contains 1,169 cells collected from both the tumor core and peritumoral tissue [5].

After preprocessing and filtering using STAR for read alignment [6] and HaplotypeCaller for variant calling [26], we retained 1,015 SNVs in MM16, 1,528 SNVs in MM34, and 969 SNVs in BT_S2 for phylogenetic tree inference (see Additional file 1: Section A.6).

We ran two rounds of tree inference and parameter optimization, from which we use the results of the second round. More information on the parameter optimization can be found in Additional file 1: Section A.7. In addition we generated consensus trees from 1,000 bootstrap samples for the multiple myeloma datasets to assess uncertainty in the tree structure (see Additional file 1: Section A.3).

In the MM16 dataset, our inferred trees (Fig. 5) show a partial separation between pre-treatment and post-treatment cells, with some post-treatment cells (in blue) appearing more closely related to the primary tumor cells (in red). This observation aligns with findings from the original study [8], which identified five post-treatment cells as surviving tumor cells based on copy number alterations. In contrast, DENDRO separates the two samples into two almost completely distinct branches, which does not align with five of the post-treatment cells indeed being tumor cells similar to the pre-treatment cells. SClineager and PhylinSic predict more intermixing of pre- and post-treatment cells but also do not reflect the expected five cells that should appear more closely related to the pre-treatment cells. When comparing automatically assigned clones to the classification of post-treatment and primary tumor cells, SCITE-RNA and DENDRO both achieve the highest Adjusted Rand Index (ARI) scores [11, 27] of 0.473 (see Additional file 1: Section A.3 and Fig. S9).

For patient MM34, our method identifies a clear split between primary and metastatic tumor cells, with metastasis-derived cells forming a distinct subtree (Fig. 6). The consensus tree topology supports a linear evolutionary model, consistent with the interpretation of the authors of the original study [8]. This expected separation between primary and metastatic tumor cells is similarly revealed by PhylinSic, but not captured as clearly in the tree inferred by DENDRO and even less so in the tree inferred by SClineager. This is also reflected in the ARI scores for automatic clone assignment, where SCITE-RNA and its consensus tree achieve the highest values of 0.938 and 0.969, respectively (see Additional file 1: Section A.3 and Fig. S10).

For BT_S2 we use the provided cell type annotations, which were derived in the original study by clustering gene expression data [5]. The authors used copy number profiles estimated from the scRNA-seq data to validate the split between tumor cells and other cell types [5]. In the inferred SCITE-RNA tree, neoplastic cells indeed predominantly form a distinct branch containing 85% of all the tumor cells with 93% purity, while the remaining parts of the tree contain 95% of healthy cells with a purity of 89% (Fig. 7). Neither SClineager nor PhylinSic reveal plausible phylogenetic relationships among the cell types and instead show a lot of intermixing. Although DENDRO clusters cells into distinct cell types, neoplastic cells are dispersed across multiple branches.

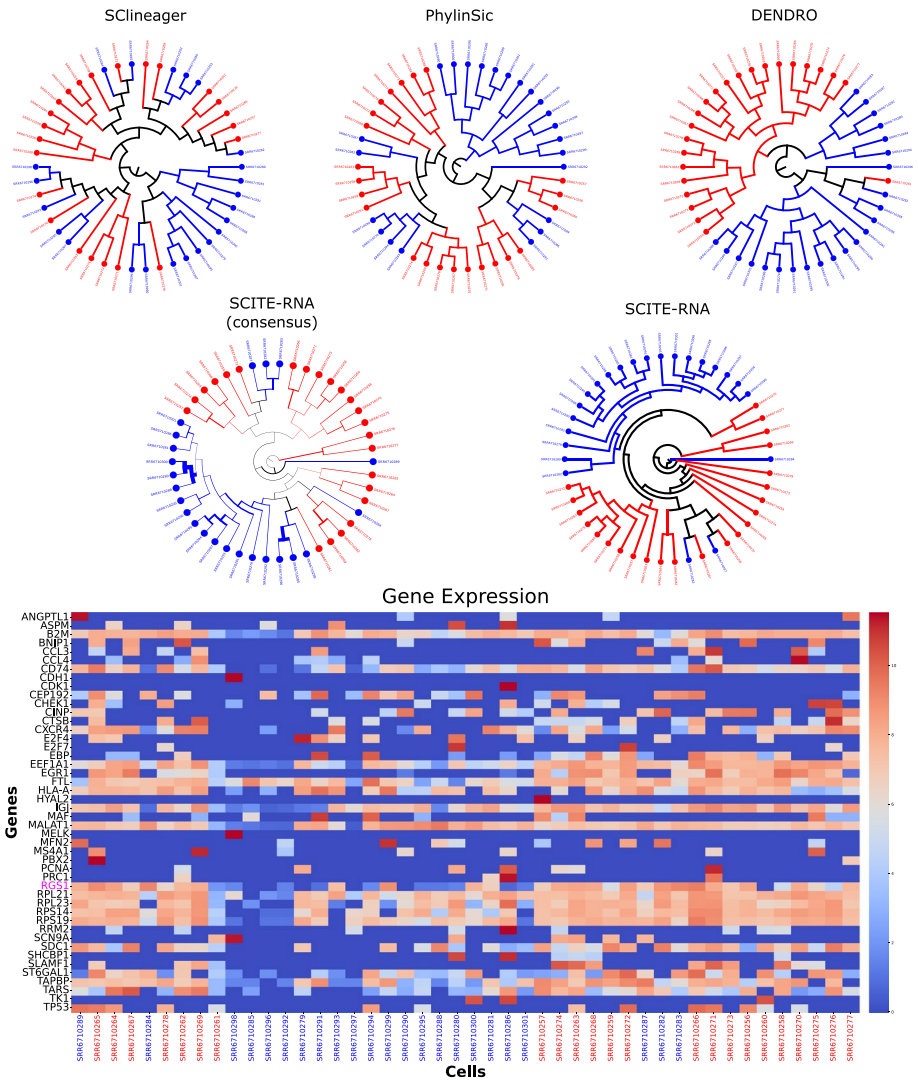

Fig. 5 Application of SCITE-RNA, DENDRO, PhylinSic and SClineager on multiple myeloma dataset MM16. A consensus tree was generated from 1,000 bootstrap samples. Red nodes represent primary tumor cells and blue nodes indicate post-treatment cells. The thickness of the splits in the consensus tree reflects the support for each split. Below the trees, we show the expression levels of selected genes, with columns ordered according to the circular leaf order of the SCITE-RNA consensus tree. The gene *RGS1* is highlighted in pink

The observed read counts show patterns of missing data and varying read depths, which differ among cell types, likely due to variations in copy number and gene expression (Fig. 7). To assess the informativeness of read depth, we performed hierarchical clustering using Ward's method based on correlation distance and assigned cells to clones by applying flat clustering to the linkage matrix using SciPy's `fcluster` [36]. For assigning cells to the seven known cell types, hierarchical clustering achieved the highest ARI score of 0.445, followed by 0.414 for DENDRO and 0.205 for SCITE-RNA. In contrast, when assessing the binary split into healthy and cancer cells, SCITE-RNA outperformed the other methods with an ARI score of 0.615, compared to 0.315 for hierarchical clustering and 0.202 for DENDRO. Both SClineager and PhylinSic yielded ARI scores around 0 (see Additional file 1: Section A.3 and Fig. S11).

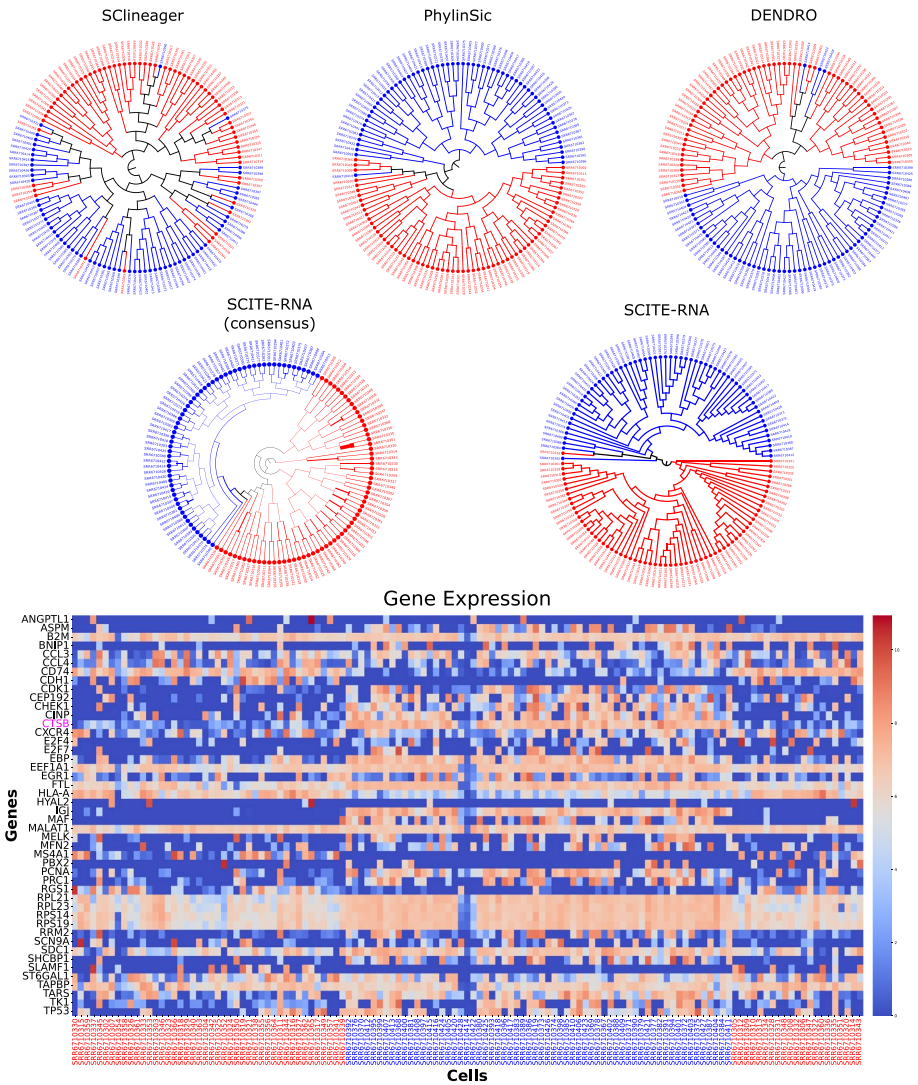

Fig. 6 Application of SCITE-RNA, DENDRO, PhylinSic and SClineager on multiple myeloma dataset MM34. A consensus tree was generated from 1,000 bootstrap samples. Red nodes correspond to primary tumor cells, while blue nodes indicate metastasis cells. The thickness of the splits in the consensus tree reflects the support for each split. Below the trees, we show the expression levels of selected genes, with columns ordered according to the circular leaf order of the consensus SCITE-RNA tree. The gene *CTSB* is highlighted in pink

In addition to the tree structure, scRNA-seq data enables us to explore gene expression patterns across the inferred phylogenies (Figs. 5, 6, and 7). We visualize the expression of several genes highlighted by Fan et al. [8] for their potential relevance to multiple myeloma, ordering cells left to right via depth-first traversal of the SCITE-RNA tree. The expression patterns seem to reflect the branching structure of the tree, despite some cells originating from different samples. They also provide additional biological support: In MM16, for instance, the gene *RGS1* (regulator of G-protein signaling 1, highlighted in Fig. 5), which is associated with poor prognosis in multiple myeloma [29], is markedly more expressed in the branches closely related to the pre-treatment cells. In MM34, *CTSB* (Cathepsin B, highlighted in Fig. 6), whose overexpression is known to be

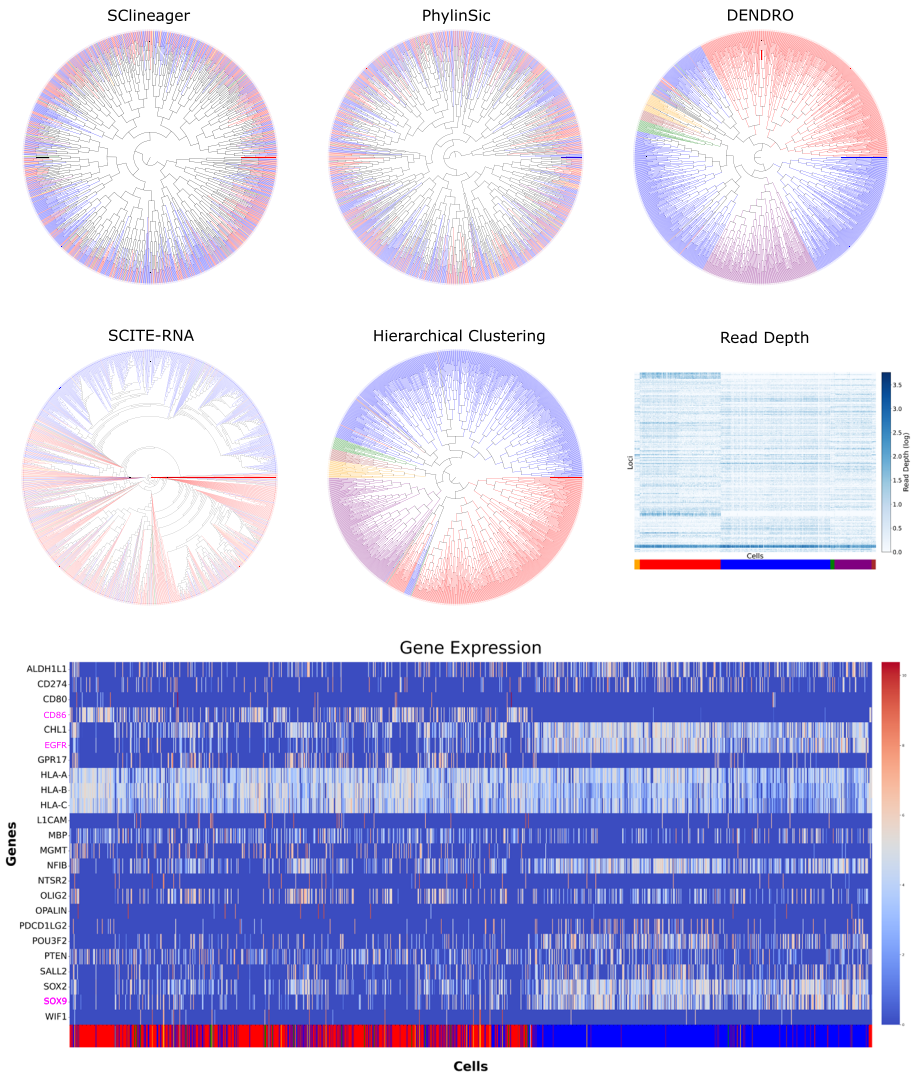

Fig. 7 Application of SCITE-RNA, DENDRO, PhylinSic, SClineager and hierarchical clustering on glioblastoma dataset BT_S2. Blue nodes represent neoplastic cells, red immune cells, purple oligodendrocyte progenitor cells, green neurons, orange astrocytes, brown oligodendrocytes and pink vascular cells. Below the trees, we show the expression levels of selected genes, with columns ordered according to the circular leaf order of the SCITE-RNA tree. The genes *EGFR*, *SOX9* and *CD86* are highlighted in pink. In the second row we show the read depths per locus ordered by cell type, which are indicated by the same color as the nodes

associated with various cancers [24], shows substantially higher expression in the metastasis branch compared to primary tumor cells. For BT_S2, we highlight genes identified in [5] and, for example, confirm their finding that *EGFR* (Epidermal Growth Factor Receptor) and *SOX9* (SRY-box transcription factor 9) are upregulated in the cancer branch, whereas *CD86* (Cluster of Differentiation 86) is downregulated.

Discussion

We have proposed SCITE-RNA for inferring phylogenies from scRNA-seq data using SNVs and demonstrated that it outperforms existing approaches on simulated datasets (Fig. 4). Two alternative methods, DENDRO [40] and SClineager [23], focus on identifying a small number of cell clones and may miss more fine-grained biological

heterogeneity that SCITE-RNA can capture. Although PhylinSic [22] shows slightly better performance than DENDRO and SClineager, it is computationally slower and still generally underperforms compared to SCITE-RNA.

To further validate our method, we applied SCITE-RNA to two multiple myeloma and one glioblastoma dataset. The inferred phylogenies demonstrate more biologically plausible evolutionary relationships among cells from different samples compared to those generated by DENDRO, PhylinSic and SClineager (Figs. 5, 6 and 7). Our PhylinSic results for datasets MM16 and MM34 differ slightly from the original publication [22]. Likely reasons are differences in the variant calling and filtering as well as stochasticity in the tree inference. For example, in the MM34 dataset, PhylinSic's original inferred tree placed two primary tumor cells among the metastasis cells. In contrast, our PhylinSic analysis identified one metastasis cell, which had the highest proportion of missing data, as more closely related to the primary tumor cells, suggesting a potential misclassification of this cell (Fig. 6).

In the glioblastoma dataset BT_S2, our method achieved the best separation between neoplastic and healthy cells into two distinct branches, though it did not distinguish between specific healthy cell types (Fig. 7). This separation suggests substantial differences in mutational profiles between cancerous and noncancerous cells, but weaker differences among healthy cell types. We further demonstrate that read depth, which is influenced by gene expression, copy number alterations, and potential noise or batch effects, carries additional information that can aid in subdividing healthy cells into distinct types during tree inference. Importantly, the use of scRNA-seq data allows for the mapping of gene expression values onto the tree structure. This not only provides support for the inferred trees but also potentially enables the study of gene expression and cellular functions at a clonal level.

Inferring trees with hundreds or even thousands of cells and mutations requires a fast tree optimization algorithm. To address this, we employ random-scan greedy search and space-swapping, where moves in each space are easy to compute efficiently, even though they might be equivalent to complex and computationally expensive moves in the other space (Fig. 2). Our results show that alternating between cell and mutation tree spaces yields superior results, particularly when both mutation tree optimization and cell lineage optimization are similarly effective (Fig. 3). While maximum-likelihood phylogenetic tree inference has been shown to be NP-hard [3], for many problems in phylogenetic inference it has been established that the complexity is (super)-exponential either in the number of cells or in the number of mutations, depending on the algorithm used [9]. By switching spaces we may be benefiting implicitly from whichever is the lower complexity, and there may be interesting theoretical and complexity questions arising from considering algorithms like ours that work in both spaces. In practice, even if optimization in one of the two spaces is suboptimal, jointly optimizing across both spaces is, generally, at least as effective as the better-performing individual space. Consequently, tree space switching is more robust regardless of the ratio of number of cells to SNVs. The tree likelihood of the inferred trees using tree space swapping was generally higher compared to the ground truth tree (Fig. 3). This highlights that the algorithm is achieving its objective of optimizing the trees well.

Nevertheless, overfitting remains a concern, particularly with noisy data. One cause is that placing each mutation at its individually optimal position in the tree may not yield a globally optimal solution. To mitigate this risk, an approach could be to penalize the number of edges with mutations for cell lineage trees and attachment points of cells in the mutation tree space to reduce the number of inferred clones. Alternatively, bootstrapping the mutations and then generating a consensus tree can reduce overfitting and allow the assessment of uncertainty in the tree structure. However, the runtime increases linearly with the number of bootstrap samples and the consensus tree depends on the presence of shared clades across bootstrap replicates. Due to low coverage and high noise, shared clades are often too rare and too small to build a reliable consensus tree, particularly for very large datasets. Alternative methods of building consensus trees from high uncertainty bootstrap trees are an interesting direction for future research. Possible solutions include removing cells with particularly variable placement across bootstrap trees or clustering cells into metacells before performing tree inference. Using metacells could not only likely reduce noise in the tree structure but also improve the algorithm's efficiency. A faster option to simplify trees is to use hierarchical clustering to group the cells into a smaller number of clones based on the predicted genotypes. We saw that both clustering and bootstrapping can offer improvements in performance for trees with few clones, and bootstrapping also for phylogenies with larger number of clones.

Adopting an MCMC-based approach could enable posterior inference over trees and model parameters. However, currently MCMC based approaches like PhylinSic or SClineager are often limited to small or medium-sized datasets, or to analyses focusing on a reduced number of clones. Tree-space switching could be integrated by proposing moves in both tree representations and converting to one space for posterior evaluation. Space switching could thus serve as a mechanism for constructing computationally efficient tree proposals that traverse regions of tree space that are otherwise hard to reach using standard moves in a single space representation.

The estimation of model parameters requires accurate genotype predictions. Variant allele frequencies that are very low or high are more likely to be assigned homozygous genotypes, as these maximize the likelihood for individual cases. However, in the presence of allelic dropout, a notable proportion of these could be heterozygous, leading to an overestimation of homozygous genotypes by shifting the inferred mutation location in the cell lineage tree. Conversely, real datasets may contain loci with both reference and alternative genotypes due to multiple mutations affecting the same locus, violating our model assumptions. As the model only compares the two most likely genotypes, some cells with two mutations might be inferred as heterozygous, potentially overestimating the number of heterozygous genotypes. Furthermore, our method does not incorporate copy number alterations or gene expression in the tree optimization itself, both of which might also be obtained from scRNA-seq data, and offer interesting directions for further phylogenetic developments.

Conclusions

Alternating between tree spaces for optimization is efficient and could be extended to other modalities, such as scDNA-seq data. The combination of fast optimization and data-driven estimation of model parameters, enables the use of scRNA-seq data not only for gene expression analysis but also for reconstructing the evolutionary history of cells, allowing us to augment expression-based analyses with phylogenetic information.

Methods

We first outline the model assumptions and then describe the tree inference procedure.

Model assumptions

The model is based on the following assumptions: (1) Each locus has exactly two copies in each cell. (2) Each locus has two possible alleles, called the reference and the alternative allele. (3) Mutations always occur at individual loci and convert one of the two copies into the other allele. (4) At most one mutation can occur at any given locus (infinite sites assumption [17]). (5) Observed read counts per cell and locus are independent given the genotypes.

These assumptions allow for our method to be computationally efficient. However, the model does not account for more complex forms of genetic variation, such as copy number alterations or structural variants. Additionally, the infinite sites assumption can be violated in cases of recurrent mutations or back-mutations, especially in larger datasets or over long timescales.

We opted for these simplifications because they not only enhance the computational efficiency and thus allow for the optimization of large trees, but also reduce the risk of overfitting to sparse and noisy single-cell RNA-seq data by lowering model complexity.

Under these assumptions, each cell contains either two reference alleles, two alternative alleles, or one reference allele and one alternative allele at any locus. These three genotypes are labeled **0** (reference-only), **1** (heterozygous), and **2** (alternative-only), respectively. Furthermore, there are only four types of transitions: **01** (i.e., **0** to **1**), **12**, **10**, and **21**. A direct conversion between genotypes **0** and **2** is not possible because it would require two mutations.

SCITE-RNA takes candidate SNVs from scRNA-seq data as input in the form of a $N \times M$ matrix $\mathbf{D}$ of observations, where N is the number of cells and M the number of loci. An entry $D_{ij} = (r_{ij}, a_{ij})$ in $\mathbf{D}$ consists of the read count of the reference, r_{ij}, and of the alternative allele, a_{ij}, at locus j in cell i. As in [32], we treat the coverage $c_{ij} = r_{ij} + a_{ij}$ as given and use beta-binomial distributions for D_{ij} when the genotype $G_{ij} \in \{\mathbf{0}, \mathbf{1}, \mathbf{2}\}$ is given.

For homozygous genotypes, the read counts are modeled as

$$P(D_{ij} \mid G_{ij} = \mathbf{0}) = \text{BetaBin}(r_{ij} \mid c_{ij}, \alpha_{\mathbf{0}}, \beta_{\mathbf{0}}),$$

$$P(D_{ij} \mid G_{ij} = \mathbf{2}) = \text{BetaBin}(r_{ij} \mid c_{ij}, \alpha_{\mathbf{2}}, \beta_{\mathbf{2}})$$

where α and β are parameterized by the expected variant allele frequency of the alternative allele $f = \alpha/(\alpha + \beta)$ and overdispersion $\omega = \alpha + \beta$, which increases with decreasing variance. These parameters depend on the underlying genotype, as we detail later.

In the heterozygous case, we use a mixture of beta-binomial distributions to model allele-specific dropout. Specifically, we assume that when dropout occurs, the observed read counts follow the homozygous distribution of the remaining allele. Let δ denote the dropout probability. Then we set

$$P(D_{ij} \mid G_{ij} = \mathbf{1}) = (1-\delta)\,\mathrm{BetaBin}(r_{ij} \mid c_{ij}, \alpha_{\mathbf{1}}, \beta_{\mathbf{1}}) + \frac{\delta}{2}P(D_{ij} \mid G_{ij} = \mathbf{0}) + \frac{\delta}{2}P(D_{ij} \mid G_{ij} = \mathbf{2})$$

Since the observations per cell and locus are independent given the genotypes, the likelihood of $\mathbf{D}$ factorizes as the product over all cells and loci as

$$P(\mathbf{D} \mid \mathbf{G}) = \prod_{i,j} P(D_{ij} \mid G_{ij}) \tag{1}$$

To estimate the parameters α and β, or equivalently f and ω, which are dependent on the genotype, we consider the case $G_{ij} = \mathbf{0}$. Since in this case only reference alleles are present, all reads except the erroneous ones support that allele. Errors might arise during reverse transcription, PCR amplification, or during sequencing, and we assume these error processes affect both alleles symmetrically. Consequently, the expected variant allele frequency for the homozygous reference genotype should satisfy

$$f_{\mathbf{0}} = 1 - f_{\mathbf{2}}$$

In the heterozygous case ($G_{ij} = \mathbf{1}$), the two alleles are present in equal amounts, so their expected read counts are also equal, hence $f_{\mathbf{1}} = \frac{1}{2}$.

To estimate the values of $\omega_{\mathbf{0}} = \omega_{\mathbf{2}}$ and $\omega_{\mathbf{1}}$, we note that in the heterozygous case, both biological and technical factors introduce additional variance to the expected variant allele frequency f. Reverse transcription may fail to capture one of the two alleles, and amplification and expression levels may differ between the two alleles. In such cases, the ratio between reference and alternative alleles is no longer 1:1 at sequencing time. This variability is accounted for by choosing a smaller value for $\omega_{\mathbf{1}}$.

To infer the expected variant allele frequency $f_{\mathbf{0}}$, dropout δ, and the overdispersion parameters $\omega_{\mathbf{0}}$ and $\omega_{\mathbf{1}}$ from real scRNA-seq data, we adopt an iterative approach. In a first round, we perform tree inference using a predefined set of parameters. Based on the resulting genotype assignments, we optimize model parameters to better reflect the characteristics of the specific dataset. In simulations we show that the model parameters are indeed approximately recovered by our method (see Additional file 1: Fig. S12 and Additional file 1: Section A.8 for details). Since allelic expression can be imbalanced or even monoallelic in single cells, both $\omega_{\mathbf{1}}$ and the dropout probability can vary across SNVs and consequently we optimize SNV-specific parameter values using the bounded limited-memory BFGS (Broyden–Fletcher–Goldfarb–Shanno) algorithm [2] (see Additional file 1: Section A.7 for more details). These updated parameters of the expected variant allele frequency $f_{\mathbf{0}}$, dropout δ, and

the overdispersion parameters ω_0 and ω_1 serve as refined parameters for a second round of tree inference. By default, we perform two rounds of alternating tree reconstruction and parameter estimation, with the goal to improve model fit to the data and the quality of the inferred trees, while keeping the computational costs low. Additional rounds of tree inference and parameter optimization have only marginal effects on the inferred parameters and the tree metrics remain largely stable (see Additional file 1: Figs. S13 and S14).

Tree inference

A cell phylogeny can be represented as a cell lineage tree, where mutations are placed on the edges to show where they occur, or as a mutation tree, illustrating the tumor's mutational history, with cells linked to specific mutations (Fig. 2). Tracing back to the root then identifies all mutations present in a given cell. Given a tree in either form and the attached mutations or cells, we can deduce the corresponding genotype profile $\mathbf{G}$, which allows us to obtain the likelihood of that tree by multiplying the likelihoods of the data given the genotype for each cell–locus pair using Eq. 1. This mapping is not injective, as the same genotype profile can result from different trees, but we can group trees that produce the same genotype profile into equivalence classes. Then there is a one-to-one correspondence between classes in the two tree spaces and it is easy to convert a tree into a random instance of the same class (which therefore has the same likelihood) in the other space. As an example in Fig. 2, consider the cell lineage tree with 2 mutation events occurring on the same edge at the root. If we convert this cell tree into a mutation tree, any permutation of these mutations (e.g. (0,1) or (1,0)) is valid and thus one mutation order is randomly chosen.

Before tree inference, potentially mutated loci are selected, and the most likely genotypes before and after the mutation are determined (see Additional file 1: Section A.9 for details). This allows us to compute the log-likelihood differences between those two genotypes for each cell–locus pair.

The goal of the tree inference algorithm is to find the tree with the highest likelihood. Due to the enormous sizes of the tree spaces, it is impractical to test every possible tree. Instead, the algorithm starts with a random tree and gradually optimizes it using a greedy approach. At each step, the algorithm is given a starting tree from which it explores the local tree neighborhood by performing tree operations (Fig. 1). During cell lineage tree optimization, nodes are selected in a random order. Then the respective subtrees consisting of the chosen node and its descendants are pruned and reattached at their optimal location. If there are several positions with equal likelihood, one is chosen randomly. This step is repeated until every node, including internal nodes, has been processed.

The same standard pruning and reattaching of subtrees is performed for mutation trees. Since they often contain linear chains of mutations, which are harder to optimize by switching subtrees, we added an optional second move for smaller mutation trees. In this move, individual mutations with only a single child are pruned and reinserted at their optimal position. This helps optimally rearranging linear chains of mutations. For cancer datasets with several thousand potential SNVs, we focused on pruning and reattaching of subtrees to keep computational costs low.

For each tree, the algorithm calculates the highest likelihood achievable by attaching mutations to edges (for cell lineage trees) or cells to nodes (for mutation trees). Since the observations per cell and locus are assumed to be independent, we only need to determine the maximum-likelihood attachment location of each individual cell or mutation separately. We use established tree traversals (e.g. [12]) to lower complexity, which leads to our method scaling well with the number of cells and mutations (see Additional file 1: Section A.10 for details).

It is very common for greedy algorithms to get stuck in local optima. However, we optimize the tree by alternating between two different tree spaces, and a local optimum in one space does not necessarily correspond to a local optimum in the other space. Hence, the tree space swapping is a way to potentially escape from local optima (Fig. 3). The process of converting the tree into one possible representation in the other space and continuing the exploration there repeats until the likelihood in both spaces stops improving.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s13059-026-04123-w.

Additional file 1: Supplementary material. Supplementary figures, simulation details, supporting analyses, and additional methods.

Acknowledgements
Not applicable.

Peer review information
Claudia Feng was the primary editor of this article and managed its editorial process and peer review in collaboration with the rest of the editorial team. The peer-review history is available in the online version of this article.

Authors' contributions
X.S., N.Z. and J.K. developed the method and contributed to data analysis. J.H. contributed to data analysis. J.K. and N.B. supervised the project. All authors contributed to the manuscript preparation.

Authors' information
Niko Beerenwinkel: Bluesky @cbg.ethz.ch.

Funding
Part of this work was funded by the EC Horizon 2020 program (project OLISSIPO, No. 951970) and the Swiss National Science Foundation (project No. 310030_179518 and 320030_236168).

Data availability
The multiple myeloma datasets are publicly available from the NCBI Gene-Expression Omnibus (GEO) under accession number GSE110499 [7], and the glioblastoma dataset is available under accession number GSE84465 [4].
The code is published on GitHub https://github.com/cbg-ethz/SCITE-RNA [42] under the GNU General Public License version 3 and archived on Zenodo https://doi.org/10.5281/zenodo.20306664 [41] under the same license.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
JK is an Editorial Board Member for Genome Biology but was not involved in the editorial process of this manuscript. All other authors have no competing interests to declare.

Received: 5 September 2025 Accepted: 21 May 2026
Published online: 15 June 2026

References

1. Bouckaert R, Vaughan TG, Barido-Sottani J, Duchêne S, Fourment M, Gavryushkina A, et al. BEAST 2.5: An advanced software platform for Bayesian evolutionary analysis. PLoS Comput Biol. 2019;15. https://doi.org/10.1371/journal.pcbi.1006650.
2. Byrd RH, Lu P, Nocedal J, Zhu C. A limited memory algorithm for bound constrained optimization. SIAM J Sci Comput. 1995;16:1190–208. https://doi.org/10.1137/0916069.
3. Chor B, Tuller T. Maximum likelihood of evolutionary trees is hard. vol. 3500; 2005. p. 296–310. https://doi.org/10.1007/11415770_23.
4. Darmanis S, Quake S. Single-cell RNAseq analysis of diffuse neoplastic infiltrating cells at the migrating front of human glioblastoma. Datasets. Gene Expression Omnibus. 2017. https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE84465. Accessed on 20 May 2026.
5. Darmanis S, Sloan SA, Croote D, Mignardi M, Chernikova S, Samghababi P, et al. Single-cell RNA-seq analysis of infiltrating neoplastic cells at the migrating front of human glioblastoma. Cell Rep. 2017;21:1399–410. https://doi.org/10.1016/j.celrep.2017.10.030.
6. Dobin A, Davis CA, Schlesinger F, Drenkow J, Zaleski C, Jha S, et al. STAR: ultrafast universal RNA-seq aligner. Bioinformatics. 2013;29:15–21. https://doi.org/10.1093/bioinformatics/bts635.
7. Fan J, Kharchenko P, Ryu D, Lee H, Kim K, Park W. Single cell RNA sequencing of multiple myeloma II. Datasets. Gene Expression Omnibus. 2018. https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE110499. Accessed on 20 May 2026.
8. Fan J, Lee HO, Lee S, Ryu DE, Lee S, Xue C, et al. Linking transcriptional and genetic tumor heterogeneity through allele analysis of single-cell RNA-seq data. Genome Res. 2018;28:1217–27. https://doi.org/10.1101/gr.228080.117.
9. Gramm J, Nickelsen A, Tantau T. In: Keith JM, editor. Fixed-parameter algorithms in phylogenetics. Totowa: Humana Press; 2008. p. 507–535. https://doi.org/10.1007/978-1-60327-159-2_24.
10. Heumos L, Schaar AC, Lance C, Litinetskaya A, Drost F, Zappia L, et al. Best practices for single-cell analysis across modalities. Nat Rev Genet. 2023;24:550–72. https://doi.org/10.1038/s41576-023-00586-w.
11. Hubert L, Arabie P. Comparing partitions. J Classif. 1985;2:193–218. https://doi.org/10.1007/BF01908075.
12. Jahn K, Kuipers J, Beerenwinkel N. Tree inference for single-cell data. Genome Biol. 2016;17:86. https://doi.org/10.1186/s13059-016-0936-x.
13. Jovic D, Liang X, Zeng H, Lin L, Xu F, Luo Y. Single-cell RNA sequencing technologies and applications: A brief overview. Clin Transl Med. 2022;12:e694. https://doi.org/10.1002/ctm2.694.
14. Jun S, Toosi H, Mold J, Engblom C, Chen X, O'Flanagan C, et al. Reconstructing clonal tree for phylo-phenotypic characterization of cancer using single-cell transcriptomics. Nat Commun. 2023;22. https://doi.org/10.1038/s41467-023-36202-y.
15. Kang S, Borgsmüller N, Valecha M, Kuipers J, Alves JM, Prado-López S, et al. SIEVE: joint inference of single-nucleotide variants and cell phylogeny from single-cell DNA sequencing data. Genome Biol. 2022;23:248. https://doi.org/10.1186/s13059-022-02813-9.
16. Kang S, Borgsmüller N, Valecha M, Markowska M, Kuipers J, Beerenwinkel N, et al. DelSIEVE: cell phylogeny modeling of single nucleotide variants and deletions from single-cell DNA sequencing data. Genome Biol. 2025;26:255. https://doi.org/10.1186/s13059-025-03738-9.
17. Kimura M. The number of heterozygous nucleotide sites maintained in a finite population due to steady flux of mutations. Genetics. 1969;61:893–903. https://doi.org/10.1093/genetics/61.4.893.
18. Koptagel H, Jun SH, Hård J, Lagergren J. Scuphr: A probabilistic framework for cell lineage tree reconstruction. PLoS Comput Biol. 2024;20(5). https://doi.org/10.1371/journal.pcbi.1012094.
19. Kozlov A, Alves JM, Stamatakis A, Posada D. CellPhy: accurate and fast probabilistic inference of single-cell phylogenies from scDNA-seq data. Genome Biol. 2022;23:37. https://doi.org/10.1186/s13059-021-02583-w.
20. Kuipers J, Tuncel MA, Ferreira PF, Jahn K, Beerenwinkel N. Single-cell copy number calling and event history reconstruction. Bioinformatics. 2025;41. https://doi.org/10.1093/bioinformatics/btaf072.
21. Lei Y, Tang R, Xu J, Wang W, Zhang B, Liu J, et al. Applications of single-cell sequencing in cancer research: progress and perspectives. J Hematol Oncol. 2021;14:91. https://doi.org/10.1186/s13045-021-01105-2.
22. Liu X, Griffiths JI, Bishara I, et al. Phylogenetic inference from single-cell RNA-seq data. Sci Rep. 2023;13. https://doi.org/10.1038/s41598-023-39995-6.
23. Lu T, Park S, Zhu J, et al. Overcoming expressional drop-outs in lineage reconstruction from single-cell RNA-sequencing data. Cell Rep. 2021;34. https://doi.org/10.1016/j.celrep.2020.108589.
24. Ma K, Chen X, Liu W, Chen S, Yang C, Yang J. CTSB is a negative prognostic biomarker and therapeutic target associated with immune cells infiltration and immunosuppression in gliomas. Sci Rep. 2022;12:4295. https://doi.org/10.1038/s41598-022-08346-2.
25. Picelli S, Faridani O, Björklund A, et al. Full-length RNA-seq from single cells using Smart-seq2. Nat Protoc. 2014;9:171–81. https://doi.org/10.1038/nprot.2014.006.
26. Poplin R, Ruano-Rubio V, DePristo MA, Fennell TJ, Carneiro MO, Van der Auwera GA, et al. Scaling accurate genetic variant discovery to tens of thousands of samples. bioRxiv. 2017. https://doi.org/10.1101/201178.
27. Rand WM. Objective criteria for the evaluation of clustering methods. J Am Stat Assoc. 1971;66:846–50. https://doi.org/10.1080/01621459.1971.10482356.
28. Robinson DF, Foulds LR. Comparison of phylogenetic trees. Math Biosci. 1981;53:131–47. https://doi.org/10.1016/0025-5564(81)90043-2.
29. Roh J, Shin SJ, Lee AN, Yoon DH, Suh C, Park CJ, et al. RGS1 expression is associated with poor prognosis in multiple myeloma. J Clin Pathol. 2017;70:202–207. https://doi.org/10.1136/jclinpath-2016-203713.
30. Ryu D, Kim SJ, Hong Y, Jo A, Kim N, Kim HJ, et al. Alterations in the transcriptional programs of myeloma cells and the microenvironment during extramedullary progression affect proliferation and immune evasion. Clinical Cancer Research. 2020;26:935–944. Epub 2019 Sep 26. Erratum in: Clin Cancer Res. 2020 Sep 15;26(18):5049. https://doi.org/10.1158/1078-0432.CCR-19-0694.

31. Sashittal P, Zhang H, Iacobuzio-Donahue C, Raphael BJ. ConDoR: tumor phylogeny inference with a copy-number constrained mutation loss model. Genome Biol. 2023;24. https://doi.org/10.1186/s13059-023-03106-5.
32. Singer J, Kuipers J, Jahn K, et al. Single-cell mutation identification via phylogenetic inference. Nat Commun. 2018;9. https://doi.org/10.1038/s41467-018-07627-7.
33. Sollier E, Kuipers, J K Takahashi, et al. COMPASS: joint copy number and mutation phylogeny reconstruction from amplicon single-cell sequencing data. Nat Commun. 2023;14. https://doi.org/10.1038/s41467-023-40378-8.
34. Steel M, Penny D. Distributions of tree comparison metrics-some new results. Systematic Biology - SYST BIOL. 1993;06(42):126–41. https://doi.org/10.1093/sysbio/42.2.126.
35. Sukumaran J, Holder M. DendroPy: a Python library for phylogenetic computing. Bioinformatics (Oxford England). 2010;06(26):1569–71. https://doi.org/10.1093/bioinformatics/btq228.
36. Virtanen P, Gommers R, Oliphant TE, et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. Nature Methods. 2020;261–72. https://doi.org/10.1038/s41592-019-0686-2.
37. Wang X, He Y, Zhang Q, Ren X, Zhang Z. Direct comparative analyses of 10X Genomics Chromium and Smart-Seq2. Genomics Proteomics Bioinformatics. 2021;19:253–66. https://doi.org/10.1016/j.gpb.2020.02.005.
38. Weideman AMK, Wang R, Ibrahim JG, Jiang Y. Canopy2: Tumor phylogeny inference by bulk DNA and single-cell RNA sequencing. Stat Biosci. 2026;18:68–110. https://doi.org/10.1007/s12561-024-09466-1.
39. Zheng GXY, Terry JM, Belgrader P, Ryvkin P, Bent ZW, Wilson R, et al. Massively parallel digital transcriptional profiling of single cells. Nat Commun. 2017;8:14049. https://doi.org/10.1038/ncomms14049.
40. Zhou Z, Xu B, Minn A, et al. DENDRO: genetic heterogeneity profiling and subclone detection by single-cell RNA sequencing. Genome Biol. 2020;21. https://doi.org/10.1186/s13059-019-1922-x.
41. Zimmermann N, Sun X, Hård J, Kuipers J, Beerenwinkel N. Phylogenetic tree inference from single-cell RNA sequencing data with SCITE-RNA. Zenodo. 2026. https://doi.org/10.5281/zenodo.20306664.
42. Zimmermann N, Sun X, Hård J, Kuipers J, Beerenwinkel N. SCITE-RNA. Github. 2026. https://github.com/cbg-ethz/SCITE-RNA. Accessed on 20 May 2026.

Publisher's Note