He *et al. Genome Medicine* (2026) 18:79
https://doi.org/10.1186/s13073-026-01648-4

Genome Medicine

RESEARCH Open Access

Single-cell omics data-driven decoding of tumor clonal evolution through reinforcement learning

Qiushun He[1†], Zhilong Zhang[2,3†], Yuanmei Wang[4,5†], Xinghao Du[2,3], Chenxi Xie[1], Yueyuxiao Yang[1], Huanming Yang[4,5*], Yang Yu[2,3,6*] and Meng Yang[1*]

Abstract

Background Tumor evolution is driven by substantial cellular heterogeneity, yet its reconstruction remains challenging with conventional bulk sequencing approaches. Single-cell transcriptomic data provide opportunities to study clonal diversity and evolutionary dynamics, but accurate inference of clonal structure and lineage relationships from these data remains difficult.

Methods We developed single-cell reinforcement learning (RL) for evolution modeling (scRevol), an RL-based model for inferring tumor evolution from single-cell RNA sequencing (scRNA-seq) data. Using copy number variation (CNV) profiles inferred from scRNA-seq data, scRevol employs a label assignment learning strategy to generate informative embeddings, identify clonal populations, and reconstruct evolutionary trajectories. We evaluated scRevol using simulated datasets, lineage tracing data, and ovarian cancer scRNA-seq datasets.

Results In simulated datasets, scRevol showed robust intra-cluster coherence and accurately recovered lineage topology across varying levels of clonal complexity and noise. Compared with clustering baselines and existing methods for single-cell tumor evolution analysis, tscRevol achieved strong agreement with ground truth. In lineage tracing data, scRevol identified clonal groups associated with metastatic potential and revealed substantial metastatic heterogeneity. In ovarian cancer datasets, scRevol resolved subclonal structures across primary and metastatic lesions and associated inferred clones with distinct transcriptional and pathway-level features.

Conclusions These results support scRevol as a practical framework for reconstructing tumor evolution from single-cell transcriptomic data and for characterizing clonal architecture and subclonal diversity.

†Qiushun He, Zhilong Zhang and Yuanmei Wang contributed equally to this work.

*Correspondence:
Huanming Yang
yanghm@genomics.cn
Yang Yu
yuy@nju.edu.cn
Meng Yang
yangmeng1@mgi-tech.com
[1]MGI Tech, Building 11, Beishan Industrial Zone, Yantian District, Shenzhen 518083, China
[2]National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China
[3]School of Artificial Intelligence, Nanjing University, Nanjing 210023, China
[4]BGI-Shenzhen, Shenzhen 518083, China
[5]HIM-BGI Omics Center, Hangzhou Institute of Medicine (HIM), Chinese Academy of Sciences, 150 Dongfang Street, Qiantang District, Hangzhou 310016, China
[6]Computer Science & Technology Building, 163 Xianlin Road, Qixia District, Nanjing 210023, Jiangsu Province, China

Background

Tumor evolution is a dynamic and multifaceted process characterized by the continuous emergence of genetic variants, clonal diversification, and selective pressures that drive cancer progression and therapy resistance [1, 2]. Clones exhibit profound heterogeneity, varying in their genetic composition both within a single tumor (intra-tumor) and among different tumors (inter-tumor). Understanding the evolutionary process is crucial for developing effective cancer treatment strategies. The study of tumor evolution, particularly the identification and tracking of repeated evolutionary patterns, is pivotal for precision oncology [3, 4].

Computational approaches for reconstructing tumor evolution have been developed across different sequencing modalities and can be broadly grouped into three categories. The first class of algorithms is based on bulk sequencing data, where DNA or RNA from both cancerous and normal cells are sequenced together. For instance, PyClone is renowned for its Bayesian clustering approach to accurately infer clonal genotypes, though it can be computationally intensive and may struggle with low-depth sequencing data [5]. ClonEvol focuses on visualizing the clonal evolution of tumors and provides intuitive graphical representations, but its reliance on pre-determined clusters can limit its flexibility [6]. Models based on bulk sequencing data typically rely on cancer cell fraction, single-nucleotide variants (SNVs), and/or copy number variations (CNVs). They are limited by the need for high sequencing depth and tumor purity, and may also result in the loss of allele-specific information [6, 7]. An alternative approach involves single-cell DNA sequencing, which offers a more detailed view of clonal evolution [8–11]. SCITE and OncoNEM are specifically designed for constructing phylogenetic trees from single-cell DNA data [8, 9]. SCITE, a Markov Chain Monte Carlo algorithm, constructs a maximum likelihood tree from imperfect single-cell sequencing genotypes, illustrating the sequence of mutations in tumor evolution. OncoNEM is a likelihood-based method that uses a heuristic search to identify the maximum likelihood clonal tree, which represents the evolutionary relationships among subpopulations. It clusters cells into clones and infers unobserved populations to enhance likelihood. Single-cell DNA sequencing methods offer a promising alternative to bulk sequencing for studying tumor evolution. However, they are limited by low throughput and reduced genome coverage [12, 13]. The third class of algorithms is based on scRNA-seq data. These methods are either clustering algorithms (Clonalscope [14], STARCH [15]) or methods used exclusively for inferring tumor evolution (SCEVAN [16]). Most existing approaches address clonal assignment and evolutionary inference with separate modules, and often make strong modeling assumptions (e.g., infinite-sites) or impose demanding data requirements (e.g., high purity). As more scRNA-seq data is generated, effectively utilizing this data to combine clustering and phylogenetic tree construction for inferring the clonal evolution tree of tumors will greatly aid in understanding the evolutionary processes of tumors.

Reinforcement learning has emerged as a powerful paradigm in machine learning [17], yielding impressive outcomes in robotics [18], autonomous driving [19], navigation [20], and large language models [21]. By iteratively receiving feedback from an environment, an RL agent aims to discover an optimal policy that maximizes cumulative rewards [22]. The application of RL to tumor evolution has recently gained traction, offering a data-driven alternative to overcome the limitations of traditional methods [23, 24]. For example, Azer et al. integrated single-cell DNA sequencing with neural networks and RL to infer tumor phylogeny, demonstrating resilience to noise and scalability across different cell and mutation counts [23]. In single-cell contexts, however, SNV calling is particularly vulnerable to allelic dropout and coverage variation, whereas RNA-inferred CNV profiles tend to yield more stable signals for defining malignant clones and subclones. Building on this rationale, we introduce **s**ingle-**c**ell **r**einforcement learning for **evol**ution modeling (scRevol), which leverages CNVs derived from scRNA-seq data to (i) derive clonal assignments and (ii) infer evolutionary relationships in a sequential pipeline. Through an RL-based training process, scRevol infers clonal structure and corresponding evolutionary relationships, offering a practical framework for modeling tumor progression.

Methods

Framework of scRevol

The scRevol algorithm integrates RL and graph-based methods into a cohesive three-step pipeline designed to achieve robust clustering and accurate tumor evolution tree reconstruction.

In the first step, the unsupervised clustering problem is reformulated as a decision-making process modeled as a Markov Decision Process (MDP). This framework enables the application of the Proximal Policy Optimization (PPO) algorithm, an RL approach known for its data efficiency and training stability, to solve an MDP for optimizing latent embeddings [25]. The process begins with the input CNV matrix D, an empty buffer B, an initialized policy model π_θ represented by a neural network, a reward function R, and a clustering method C. Here, CNV matrix D is represented as a $\mathrm{c} \times \mathrm{m}$ matrix, where c denotes the number of cells and m denotes the number of the CNV features across genomic regions or genes. The training is conducted iteratively over a predefined number of steps. The policy model is implemented as a neural network comprising one hidden layer of size 512×512 and an output layer, with ReLU activation functions. During

training, a batch of size 100 is updated at each step using the Adam optimizer with a learning rate of 0.0001.

During each iteration i, the policy model computes the parameters of a latent normal distribution N_i, from which a set of (noted as n) latent embeddings z ($\mathrm{c} \times \mathrm{k}$, where k is a predefined hyperparameter) is sampled. A normal distribution is adopted because it provides a smooth latent space that stabilizes optimization and facilitates efficient gradient-based learning. These embeddings compress the CNV matrix and reduce noise, making them suitable for downstream clustering. The clustering method C is then applied to group the latent embeddings, producing cluster labels l. The quality of clustering is evaluated by the reward function R, which assigns reward r based on clustering metrics. In the following experiments, the silhouette score is employed as the specific reward metric. The generated tuples $(\mathrm{z}, \mathrm{l}, \mathrm{r})$, containing the sampled embeddings, cluster labels, and corresponding rewards, are stored in the buffer B. To update the policy, we optimize a PPO objective using the clustering reward associated with each sampled embedding. This requires translating the scalar clustering score into a learning signal for policy updates while keeping successive updates conservative as the policy changes. We therefore compute an advantage term for each embedding to capture its relative quality within the current batch. We also compute an importance ratio that compares how likely the same embedding is under the current policy versus the previous policy. These quantities are then used to form the PPO clipped surrogate objective described below. Formally, the advantage Adv is defined as:

$$Adv_{z_j} = r_j - \frac{1}{n}\sum_{k=1}^{n} r_k \tag{1}$$

We next compute the importance ratio δ , defined as the likelihood ratio of sampling the same embedding under the current distribution N_i versus the previous distribution N_{old}:

$$\delta = \frac{\log P(z_j|N_i)}{\log P(z_j|N_{old})} \tag{2}$$

To ensure optimization stability, PPO updates the policy π_θ by maximizing a clipped surrogate objective function, using Adv and δ:

$$\begin{aligned} &\mathcal{L}(\theta) \leftarrow E_j \in \{1, \ldots, n\} \\ &[min(Adv \cdot \delta,\ Adv \cdot clip(\delta,\ 1-\epsilon,\ 1+\epsilon))] \end{aligned} \tag{3}$$

This iterative process continues until the policy π_θ converges, producing latent embeddings with improved clustering performance and reduced noise.

In the second step, the trained policy model is used to sample optimized latent embeddings from the input CNV matrix D. These embeddings are passed through the clustering method C to generate final cluster labels L. The refined labels provide a more accurate structural representation of the data, capturing underlying biological patterns and heterogeneity. These cluster labels serve as input for the final stage of the pipeline.

The third step focuses on reconstructing the tumor evolution tree using the minimum spanning tree (MST) algorithm. Cluster centers are computed by averaging the CNV data points within each cluster, as defined by the labels L. Pairwise distances between cluster centers are then calculated using a predefined metric such as the Euclidean distance:

$$Distance(\mathrm{C}_i, \mathrm{C}_j) = \parallel \mathrm{C}_i - \mathrm{C}_j \parallel_2 \tag{4}$$

where C_i and C_j are the cluster centers. These distances are used to construct a complete graph, where edges are iteratively added in ascending order of distance to form the MST while ensuring no cycles are introduced. This process results in a tree structure that spans all clusters and minimizes the total edge length.

Clustering CNVs via RL

Predicting latent embeddings for CNV matrices is a challenging task that requires overcoming noise and disturbances while obtaining a robust representation in the latent space that is suitable for subsequent tree-building techniques. Traditional methods often fail to address the noise adequately or rely on linear approaches, leading to suboptimal performance and limiting the potential for effective tree construction. To tackle this issue, we reformulate the problem of predicting latent embeddings as a sequential decision-making process, modeling it as an MDP. An MDP describes a system where the state evolves over time, with transitions influenced by the actions taken [26]. By adopting this MDP framework, we can systematically explore the dynamic interaction between latent embeddings and their mathematical properties.

To solve the MDP, we optimize the policy $\pi\ (\mathrm{a}|s)$ that determines the action a given the state s. The goal is to maximize the expected cumulative reward over a trajectory, defined as:

$$V(\pi) = \mathrm{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) | s0 \sim \rho(\cdot), a_t \sim \pi(\cdot|s_t), s_{t+1} \sim P(s_{t+1}|s_t, a_t)\right] \tag{5}$$

where $\gamma \in [0,1)$ is the discount factor, s_t and a_t denote the state and action at timestep t, and the expectation is taken over trajectories induced by the policy π. Pseudocode for the method is given below.

Input: Raw data D, empty buffer B, latent model π_θ, reward function R, clustering method C, clip parameter ϵ, number of iterations K.
for $k = 1, \ldots, K$ **do**
  Sample latent embedding $z \sim \pi_\theta(D)$
  Obtain cluster labels $l \leftarrow C(z)$
  Compute reward $r \leftarrow R(z)$
  Store (z, l, r) in buffer B
end for
Extract all stored samples from buffer: $(z_B, l_B, r_B) \leftarrow B$
for $k = 1, \ldots, K$ **do**
  Compute advantage estimate: Adv $\leftarrow r_B - \mathbb{E}[r_B]$
  Compute importance weight: $\delta \leftarrow \frac{\pi_\theta(z_B)}{\pi_{\theta_{\text{old}}}(z_B)}$
  Compute clipped surrogate loss: $\mathcal{L}(\theta) \leftarrow \mathbb{E}\left[\min(\text{Adv} \cdot \delta, \text{Adv} \cdot \text{clip}(\delta, 1-\epsilon, 1+\epsilon))\right]$
  Update π_θ using gradient ascent on $\mathcal{L}(\theta)$
end for
Output: Optimized latent model π_θ

Algorithm 1. PPO for Latent Embedding

Enhancing training efficiency and embedding robustness

Supervised initialization for speed-up

Training from scratch often involves extensive interactions and prolonged training times. To mitigate this, the model is initialized by approximating the dimensionality reduction results of principal component analysis (PCA). This initialization accelerates convergence and enhances stability during training, providing a more efficient starting point for the learning process.

Preventing reward hacking with Kullback-Leibler (KL) constraint

During experimentation, we observed a disconnect between optimizing reward signals and achieving high-quality clustering results. Specifically, higher rewards did not always correspond to better latent embeddings, indicating the occurrence of reward hacking. To address this, PCA results are used as a baseline under normal conditions, and a KL divergence constraint is introduced between the PCA-initialized policy π_{θ_0} and the target policy π_θ. The KL constraint is defined as:

$$D_{KL}(\pi_\theta \parallel \pi_{\theta_0}) = \mathrm{E}_{z\sim\pi_\theta}\left[\log\frac{\pi_\theta(z)}{\pi_{\theta_0}(z)}\right] \qquad (6)$$

where π_θ (z) and π_{θ_0}(z) represent the probabilities of generating the latent embedding z under the target and reference policies, respectively. This constraint ensures that the latent embeddings remain aligned with the structure captured by PCA, while still optimizing the reward. By penalizing excessive divergence, the model is guided to stay within the desired solution space, improving the quality and robustness of the embeddings.

Generation of simulated data

The copy-number matrix was the primary input for all downstream analyses. We first generated single-cell CNV profiles with CNAsim [27], with minor modifications to enhance clone selection and tree pruning, reducing the frequency of excessive events. Key parameters included the number of clones, set to values of 2, 3, 4, 5, or 6, and error rates ranging from 0.25 to 0.4 (0.25, 0.3, 0.35, and 0.4). The parameter *placement-param* was varied from 1 to 5. Configurations were randomly combined to ensure diverse dataset generation. Certain parameters remained constant across all simulations: the number of cells was fixed at 500, the normal fraction was set to 0.1, *cn-copy-param* was assigned a value of 0.7, and both *region-length* and *min-cn-length* were fixed at 1,000,000. These settings yielded a total of 499 unique datasets, providing robust input for benchmarking and analysis.

A practical complication is that the methods we benchmarked do not operate on the same data modality. Clonalscope and SCEVAN are designed to take an scRNA-seq count matrix as input and contain their own CNV-calling step, whereas scRevol and the generic clustering baselines (K-means, Leiden, Ward's hierarchical clustering, neighbor joining) assume that a cell-by-segment CNV matrix is already available. If we had evaluated all methods only on the simulated CNV matrices, the CNV-based methods would have been tested one step closer to the ground truth, while the RNA-based methods would still have to start from expression data that we had intentionally made more realistic in the CNV-to-RNA simulation step (library-size variation, dropout and expression outliers; Additional file 1: Fig. S1). In that situation, an observed performance gap may arise primarily from this additional generative stage, during which noise and distributional shifts are introduced to mimic scRNA-seq data, rather than from the clonal inference algorithms themselves.

To remove this asymmetry and to reflect the common real-world scenario in which only scRNA-seq is available, we therefore converted the simulated CNV profiles

into matched RNA-like matrices and required all RNA-input methods to start from this shared expression space. Concretely, region-level CNVs were projected to genes via a predefined region-to-gene annotation and aggregated per cell; RNA counts were then simulated on top of these gene-level CNVs using a gamma/dosage scheme with library-size variation, rare outliers and dropout, following Bai et al. [28]. The resulting matrices can be used directly by RNA-based CNV callers (for example, inferCNV) and by methods that internally reconstruct CNVs (Clonalscope, SCEVAN).

In parallel, we retained the CNV branch with controlled noise levels to assess how methods that work natively on CNV matrices, such as scRevol and the clustering baselines, respond to different error rates in the absence of a CNV-inference step. During the RNA-based benchmark, a small number of runs did not finish for external tools. Specifically, SCEVAN failed on 3 simulated datasets due to a mismatch in the subclone-name assignment step, and inferCNV failed on 11 datasets at the sequencing-depth normalization stage. To ensure that all methods were evaluated on exactly the same inputs, we therefore restricted the RNA-based comparison to the 485 simulations that completed successfully for every tool. By contrast, in the CNV-only branch all 499 CNAsim-generated CNV matrices were processed without errors.

Processing of lineage tracing data

We obtained lineage tracing data (GSE161363) [29] from the NCBI GEO database, focusing on the M5k subset (5,000 engineered cells) and the LM0 subset (pre-implantation cells). The experimental system utilized a human *KRAS*-mutant lung adenocarcinoma line (A549 cells), without inclusion of normal cells in the original study. To infer CNVs, pre-implantation cells from LM0 were used as the reference to investigate post-implantation clonal evolution.

Cells from LM0 were filtered to ensure the inclusion of those with typical expression profiles and relevance to the experimental focus. First, cells with *LG_Present == 'Engrafted'* were selected, representing lineage groups successfully engrafted after transplantation. Additional filtering was performed based on quality metrics: cells with mitochondrial gene percentages (*pct_counts_mt*) below 5% were retained, and those within the central, densely populated ranges of total counts (17,500 – 27,500) and detected genes (4,000–5,000) were selected, as visualized by SCANPY's *pl.violin* [30]. From the filtered dataset, 50 cells were randomly chosen to serve as the reference group. For M5k, the dataset originally contained approximately 100 clonal groups. We prioritized groups based on size (100-1,000 cells) and metastatic site diversity. Selected groups included those with diverse metastatic behaviors, such as CP017 (clone 17), CP026, and CP030; restricted metastasis, such as CP029, which was confined to the left lung; and specific site preferences, such as CP022 to the right lung E, CP027 to mediastinum 1, and CP039 to the right lung W. This resulted in 1,281 cells comprising the case group. The reference and case groups were combined to create a dataset of 1,331 cells. Genes expressed in fewer than three cells were excluded, leaving 15,472 genes for downstream analysis. The inferCNV tool was subsequently used to identify CNVs from the single-cell RNA sequencing data [31].

Preprocessing of ovarian cancer scRNA-seq data

scRNA-seq data for ovarian cancer were obtained from the VIB-KU Leuven Center for Cancer Biology [32]. This dataset encompasses samples across three clinical conditions, ranging from normal ovaries to early and advanced disease stages, as well as cases from primary tumor sites, metastatic lesions, and chemo-sensitive or chemo-resistant phenotypes (see Availability of data and materials). To ensure data quality, stringent quality control (QC) measures were implemented. Cells were filtered based on QC criteria, including abnormal numbers of detected genes (minGene > 500, maxGene < 6000), unique molecular identifiers (UMIs) counts (minUMI > 500, maxUMI < 30000), or percentages of mitochondrial gene expression (< 15%). Genes detected in fewer than three cells were excluded. Samples with poor overall quality, as indicated by elevated mitochondrial gene expression or atypical cell type proportions, were also excluded. Data preprocessing and integration were performed using Seurat (v4.3.0.1) [33]. After identifying the top 2000 highly variable genes, PCA was conducted, and the first 30 principal components (PCs) were retained for further analysis. Initial clustering was achieved using the shared nearest neighbor (SNN) modularity optimization algorithm. To account for batch effects arising from multiple data sources, Harmony (v1.2.0) [34] was applied to the PCA space, followed by nonlinear dimensionality reduction using uniform manifold approximation and projection (UMAP). Unsupervised clustering was performed with a resolution parameter of 0.1, resulting in the identification of distinct cell populations.

Malignant epithelial cells were identified and distinguished from other cell types across all samples, resulting in the identification of eight major cell clusters: non-epithelial cells and epithelial cells (KRT18, EPCAM, CD24). Batch correction was conducted using Harmony to minimize inter-sample variation and ensure consistent cell type annotation. inferCNV was subsequently utilized to identify CNVs, with the selection of non-epithelial cells as the reference group to confirm that the detected CNVs were specific to tumor cells and not artifacts of sequencing or data processing. A cutoff parameter of 0.1 was applied during analysis. CNV scores were computed by

aggregating the CNV levels of cells within each subcluster, providing a quantitative measure for comparative analysis.

Baseline methods

We benchmarked general-purpose clustering methods (K-means, Leiden) alongside single-cell CNV-specific tools (e.g., SCEVAN and Clonalscope [14, 16]). All baselines were run on the same preprocessed cell-by-locus CNV profiles.

General-purpose clustering baselines

We evaluated four commonly used partitioning strategies: K-means, Leiden, Ward's hierarchical clustering and neighbor joining (NJ). For K-means, the number of clusters K was chosen by the elbow criterion. Leiden clustering was run in SCANPY with PCA preprocessing under default settings unless otherwise noted [30]. Ward's hierarchical clustering was performed on Euclidean distances using Ward's linkage, and the cut level was chosen to maximise the silhouette score. For neighbor joining, we first constructed an NJ tree from the pairwise distance matrix and then obtained a partition by cutting the longest internal branches; the number of groups was likewise selected by silhouette optimisation. As these algorithms return only cluster labels rather than an explicit evolutionary structure, we reconstructed a clone-level lineage from the cluster centroids. By default we used the same MST procedure as in scRevol, and we also implemented NJ and Unweighted Pair Group Method with Arithmetic Mean (UPGMA) as alternative tree-building strategies.

Single-cell CNV-specific baselines

Clonalscope and SCEVAN are purpose-built for single-cell, CNV-based subclonal analysis. We followed the official tutorials with default parameters. As these tools also return clusters without a native tree, we applied the same MST reconstruction to obtain a clone-level lineage prior to evaluation.

Benchmark metrics

We adapted metrics from the ICGC-TCGA DREAM Somatic Mutation Calling Tumor Heterogeneity and Evolution Challenge (Salcedo et al. [35]), originally developed for bulk samples, and tailored them to single-cell analyses. For clarity, we rename the selected metrics (SC1B, SC2, and SC3) while preserving their computational intent, and report their correspondence throughout the paper. Specifically, we use these metrics to evaluate agreement in the number of clones, pairwise co-assignment consistency, and agreement of lineage topology, respectively.

Lineage-count Concordance (LCC; adapted from SC1B)

Predict the number of lineages in the sample:

$$LCC = \frac{L-d+1}{L+1} \tag{7}$$

where $\mathrm{L} \geq 1$ is the true number of subclonal lineages and $\mathrm{d} = \min(|\mathrm{k} - \mathrm{L}|, \mathrm{L}+1)$ is the absolute difference between the predicted number k and the true number L.

Pairwise Co-clustering Divergence (PCD; adapted from SC2)

Infer the lineage assignment of each CNV. This metric evaluates the alignment of CNV events within clusters by calculating the mean of the Area Under the Precision-Recall curve (AUPR) and the Adjusted Jensen-Shannon Divergence (AJSD) between the predicted co-clustering matrix CCM^{Pr} and the true co-clustering matrix CCM^{Tr}, given by:

$$PCD = \frac{AUPR(CCM^{Tr}, CCM^{Pr}) + (1 - AJSD(CCM^{Tr}, CCM^{Pr}))}{2} \tag{8}$$

The AJSD term is used to normalize the original AJSD value to align with the same direction and scale as AUPR. The co-clustering matrix CCM is an $\mathrm{n} \times \mathrm{n}$ matrix, where CCM_{ij} represents the probability that mutation i is in the same subclone as mutation j, with $0 \leq \mathrm{CCM}_{ij} \leq 1$. A perfect clustering is represented by setting $CCM_{ij} = 1$ if mutations i and j belong to the same lineage, and $CCM_{ij} = 0$ otherwise.

Topological Concordance (TC; adapted from SC3)

This metric assesses the similarity between the true and predicted matrices using the Pearson correlation coefficient (PCC):

$$TC = PCC = \frac{Cov(C,K)}{\sigma_C \sigma_K} \tag{9}$$

where $\mathrm{Cov(C, K)}$ is the covariance between the vectorized versions of the true matrix C and the predicted matrix K, and σ_C and σ_K are their standard deviations. The matrices C and K are constructed by horizontally concatenating the following matrices for both the true and predicted versions: CCM (co-clustering matrix), ADM (ancestral relationship matrix, where $ADM_{ij} = 1$ if mutation i is the ancestor of mutation j, and 0 otherwise), ADM^T (the transpose of ADM), CM (cousin matrix, computed as: $\mathrm{CM}_{ij} = 1 - \mathrm{CCM}_{ij} - \mathrm{ADM}_{ij} - \mathrm{ADM}_{ji}$).

To further enhance the evaluation of models, we incorporated the adjusted rand index (ARI) as an additional metric. The ARI compares the model's clustering assignments to true labels, providing a measure of clustering accuracy that is particularly valuable in single-cell data, where precise detection of genetic variations is essential.

In addition to ARI, we also employed normalized mutual information (NMI) and V-measure to assess the concordance between known cell-clone labels and the cell clusters identified by the algorithms.

This comprehensive approach allows for a detailed evaluation of how well the models delineate complex genetic landscapes and track evolutionary histories in single-cell sequencing data, offering valuable insights into each algorithm's ability to navigate the intricacies of tumor biology.

Feature selection and consistency evaluation

To identify relevant features for clone populations, we performed feature selection based on cell-clone grouping information, utilizing frequency metrics and mutual information gain. Since the features represent CNVs, we simplified the CNV values for each feature to enhance clarity and ensure consistency in feature selection. Specifically, CN = 2 was categorized as "normal", values greater than 2 were classified as "gain", and values less than 2 as "loss". For each feature, the frequencies of loss, normal, and gain were then calculated across clusters.

In the simulation experiments, where known CNV events were deliberately introduced, we selected features with a frequency greater than 0.5 as defining characteristics of each cluster. This criterion assumes that the CNV categories occurring most frequently (i.e., in more than 50% of the cells) best represent the defining features of the cluster. To evaluate the consistency of the predicted features, we addressed potential discrepancies between the true and predicted clone counts. The Hungarian optimization algorithm was employed to match the predicted clone groups with the true clone groups, ensuring the most accurate alignment between predicted and actual clone labels [36]. After establishing the optimal matching, we used the Jaccard index to assess the consistency of CNV features, quantifying the degree to which the predicted clones exhibited the same CNV events as those introduced in the simulation.

In lineage tracking experiments, we employed the same approach, calculating both the frequency and mutual information gain for each feature. To identify key features for defining clone groups, we set a threshold at the 95th percentile of the mutual information gain, selecting features above this threshold as the defining characteristics for partitioning the clones. Enrichment analysis was then performed on the selected features using the Enrichr API from the GSEAPY package, with entries from the Gene Ontology 2023 database [37]. In this step, we used the pooled gene set of selected CNV associated features, without separating genes into gain and loss lists, since CNV direction can vary across clone groups for the same gene and our goal here is to capture overall differences in CNV patterns across clones. Only terms with an adjusted p-value < 0.05 were retained for interpretation. Simultaneously, we conducted differential gene expression analysis based on the original scRNA-seq data using SCANPY, with groups inferred by scRevol [30].

Local Neighbourhood Consistency (LNC)

To assess lineage consistency without imposing discrete tree-derived ground truth, we treated the hybrid-prior lineage tree from the original study as a weak reference and quantified how well RNA-derived clusters respect local tree structure. For each lineage (as a phylo tree), we retained tips with non-missing cluster labels; if edge lengths were unavailable, they were set to 1. Cophenetic distances were computed, and for each tip we identified its k nearest neighbours (default $k = 10;\ \text{if}\ n < k + 1,\ k = n - 1$). The LNC score is the mean fraction of each tip's neighbours sharing its cluster label (lineages with < 5 evaluable tips were excluded).

To calibrate LNC, cluster labels were permuted $B = 1000$ times to obtain a null distribution. The normalized enrichment score (LNC_NES) is $(\mathrm{LNC} - \mu_0)/\sigma_0$, where μ_0 and σ_0 are the null mean and standard deviation; we also report a one-sided permutation p-value. For each method, lineage-level LNC_NES values were summarized across lineages by the median and interquartile range (IQR).

Single-cell pseudotime analysis

We constructed single-cell pseudotime trajectories using the "monocle2" R package (v2.30.0) [38]. The analysis involved dimensionality reduction with DDRTree and the identification of developmental paths through a MST, allowing the assignment of pseudotimes to reflect cellular progression.

Gene set functional analysis

Gene set functional analysis was executed using the R packages "clusterProfiler" and "GSVA" [39, 40]. By harnessing the Kyoto Encyclopedia of Genes and Genomes (KEGG) database, an unsupervised gene set enrichment analysis was performed to calculate enrichment scores for each gene set within each sample.

Differential expression analysis

Clonal information for each cell was mapped onto the RNA expression matrix. Differential gene expression analysis for each clone was conducted using the R package "DESeq2" [41]. Genes exhibiting an absolute log2FoldChange greater than 0.5 and an adjusted p-value less than 0.05 were considered as differentially expressed.

Sensitivity analysis

We carried out a series of ablation experiments to examine four design choices in our framework: the number of samples per update, the clipping ratio in PPO learning step, the strategy used to reconstruct a tree, and the choice of reward function. Performance was quantified by LCC, PCD, and TC as reported in Additional file 2: Table S1-S4.

Number of samples per update

Varying the batch size from 50 to 150 mainly affected the stability of the learned policy (Additional file 2: Table S1). Using 150 samples per update gave the highest lineage-count accuracy (LCC = 0.865 ± 0.111), showing that larger batches help the model recover the correct number of clones. However, our default setting of 100 samples per update achieved the best topological agreement (TC = 0.313 ± 0.201) and the highest pairwise consistency (PCD = 0.906 ± 0.084), indicating that moderate batches provide cleaner gradients without over-stretching the update step. The smallest batch [50] kept PCD high (0.904 ± 0.009) but yielded the lowest TC (0.251 ± 0.207), reflecting noisier tree structures. We therefore used 100 samples per update in all main experiments.

PPO clip ratio

The clipping ratio controls how far the new policy is allowed to move from the previous one at each update. As shown in Additional file 2: Table S2, 0.2 offered the best overall balance: it preserved high PCD (0.906 ± 0.084) and reached the highest TC (0.313 ± 0.201). A tighter clip (0.1) slightly improved pairwise agreement (PCD = 0.911 ± 0.080) but did not translate into better topology (TC = 0.300 ± 0.254), suggesting slower exploration. A looser clip (0.3) clearly degraded topology (TC = 0.179 ± 0.198) while not improving LCC or PCD. We therefore fixed the PPO clip ratio to 0.2.

Tree-construction strategies

Because several baselines only output cluster labels, we examined three post-hoc strategies to turn these labels into a lineage: MST (our default), NJ and UPGMA (Additional file 2: Table S3). All three retained high pairwise consistency (PCD ≈ 0.90), but MST produced the most faithful topology (TC = 0.313 ± 0.201) while keeping LCC comparable (0.788 ± 0.282). NJ showed similar PCD (0.901 ± 0.082) but a lower TC (0.261 ± 0.168), and UPGMA slightly improved LCC (0.806 ± 0.262) at the cost of a marked drop in TC (0.177 ± 0.177), consistent with its stronger ultrametric assumption. These results support MST as the default, with NJ/UPGMA retained as alternatives.

Reward function

We finally compared different unsupervised criteria as reward signals (Additional file 2: Table S4). Using the silhouette coefficient led to a balanced improvement on all three metrics (LCC = 0.788 ± 0.282, PCD = 0.906 ± 0.084, TC = 0.313 ± 0.201), indicating that a joint consideration of intra-cluster compactness and inter-cluster separation is most aligned with the target clonal structure. In contrast, the Davies-Bouldin index produced noticeably lower LCC (0.572 ± 0.267) and PCD (0.756 ± 0.093), while the Calinski-Harabasz index improved LCC (0.823 ± 0.221) but failed to preserve topology (TC = 0.122 ± 0.213). We therefore adopted the silhouette-based reward in all reported runs.

Runtime analysis

For the runtime comparison we distinguished between CNV-only workflows and RNA-input methods. For scRevol and the general clustering baselines (K-means, Leiden, Ward's hierarchical clustering and NJ), we measured only the step that takes an already simulated CNV matrix and produces cluster/subclone labels. This reflects the scenario in which CNV profiles have been obtained beforehand. In this setting, simple clustering methods were, as expected, the fastest, and scRevol incurred a higher but still practical cost (Additional file 2: Table S5).

Clonalscope and SCEVAN, in contrast, were executed in their standard end-to-end mode, which also performs CNV inference from RNA. That component alone typically takes several hours using inferCNV's cell-mode, and it is not included in the timings of the CNV-only methods. Their runtimes are therefore reported primarily to indicate the overall cost of RNA-input pipelines, rather than for a head-to-head comparison with scRevol or the clustering baselines. Within this framing, scRevol adds overhead but does not significantly compromise practical usability.

Results

Tumor evolution reconstruction framework

As shown in Fig. 1, our framework reconstructs tumor evolutionary trajectories from CNV matrices through a stepwise process involving clustering and evolutionary tree construction. To address noise in CNV data, the workflow is divided into two stages: clustering identifies representative cell populations, while tree construction infers the evolutionary relationships among these populations. We demonstrate the framework's effectiveness through benchmarking on simulated datasets, validation with lineage tracing data to capture metastatic heterogeneity, and integrative functional and pathway analyses to gauge the reliability of the reconstructed evolutionary trajectories.

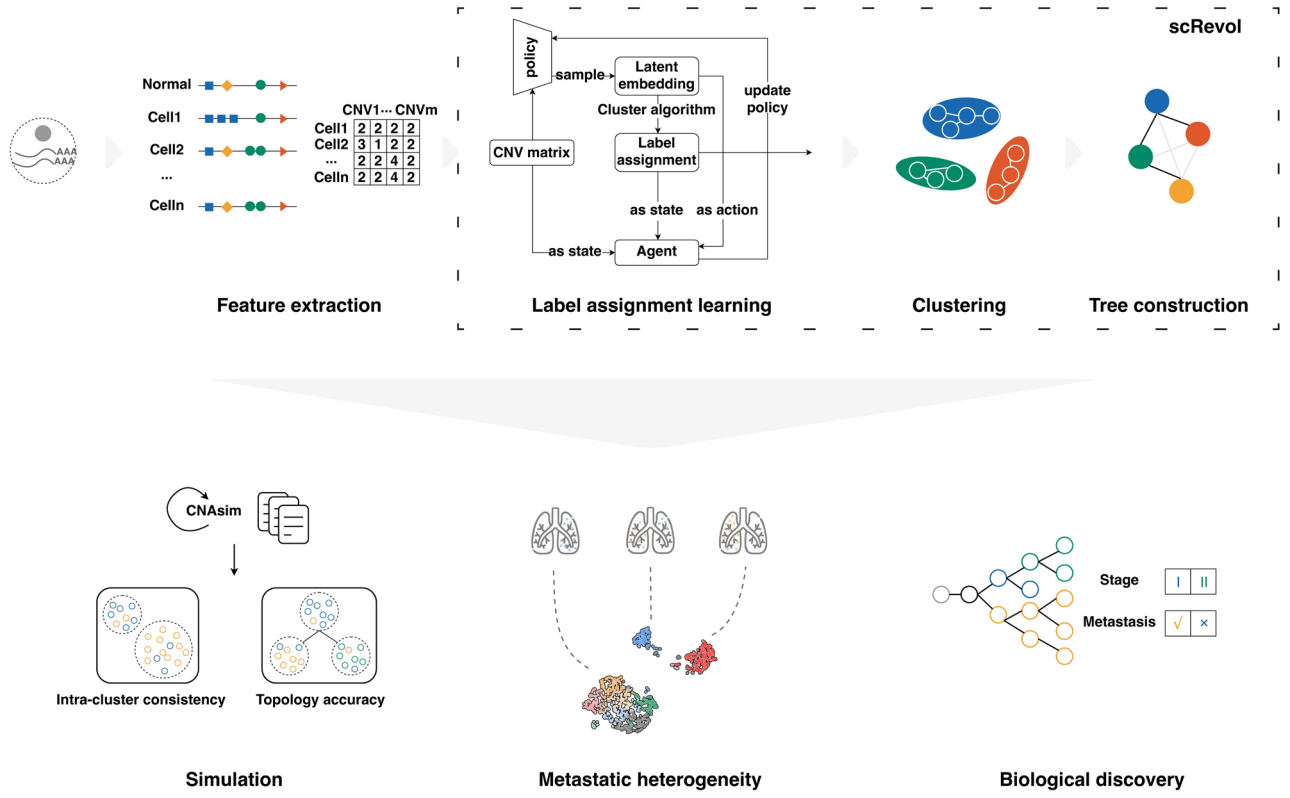

Fig. 1 Workflow of scRevol for tumor evolution modeling and biological insights. The top part illustrates the scRevol framework, including label assignment learning, clustering, and tree construction. The bottom part shows its application in tumor evolution

A key component of this framework is the label assignment learning module, which employs an RL-based approach to refine clustering. The clustering task is formulated as a MDP, where the state consists of the CNV matrix together with the evolving cluster label assignments, and actions correspond to latent embeddings that are iteratively optimized by the policy. Once an action (i.e., a latent embedding) is specified, a clustering method is applied to the embedding space to reassign cluster labels, thereby yielding a new label assignment that constitutes the next state in the MDP. In our implementation, label reassignment is performed with the Leiden algorithm. Clustering quality is quantified through a reward function, with the silhouette score used as the default reward signal in this study, and the resulting feedback guides subsequent policy updates. Using the PPO algorithm, the module progressively refines the embeddings and transitions through successive states, ultimately producing clusters that are both accurate and biologically coherent. These optimized clusters form the foundation for constructing evolutionary trees via the MST algorithm, enabling efficient and biologically consistent reconstruction of tumor progression trajectories.

Performance evaluation of scRevol

We first evaluated scRevol in an RNA-based setting that mirrors our simulation workflow (Additional file 1: Fig. S1). Clonal structures generated by CNAsim were converted into RNA-like matrices to reflect typical scRNA-seq properties, including library-size heterogeneity, sparsity, and measurement noise (Methods). We compared scRevol with general clustering baselines (Leiden, K-means, Ward's hierarchical clustering, NJ) and with single-cell tumour-evolution methods (Clonalscope, SCEVAN) across datasets spanning multiple levels of clonal complexity. Clustering accuracy was assessed with NMI, ARI, V-measure, and LCC (Fig. 2A and B). Overall, scRevol showed stable performance and was frequently among the top performers, particularly on metrics that require both correct cell assignment and recovery of the true clone number. The main deviation occurred for two-clone datasets, where LCC was lower (Fig. 2B), likely because the Leiden-based initialization at the default resolution tends to split very small lineage structures and LCC penalizes such discrepancies more strongly when the true clone count is small. Utilizing frequency-filtered features and the Hungarian optimization algorithm

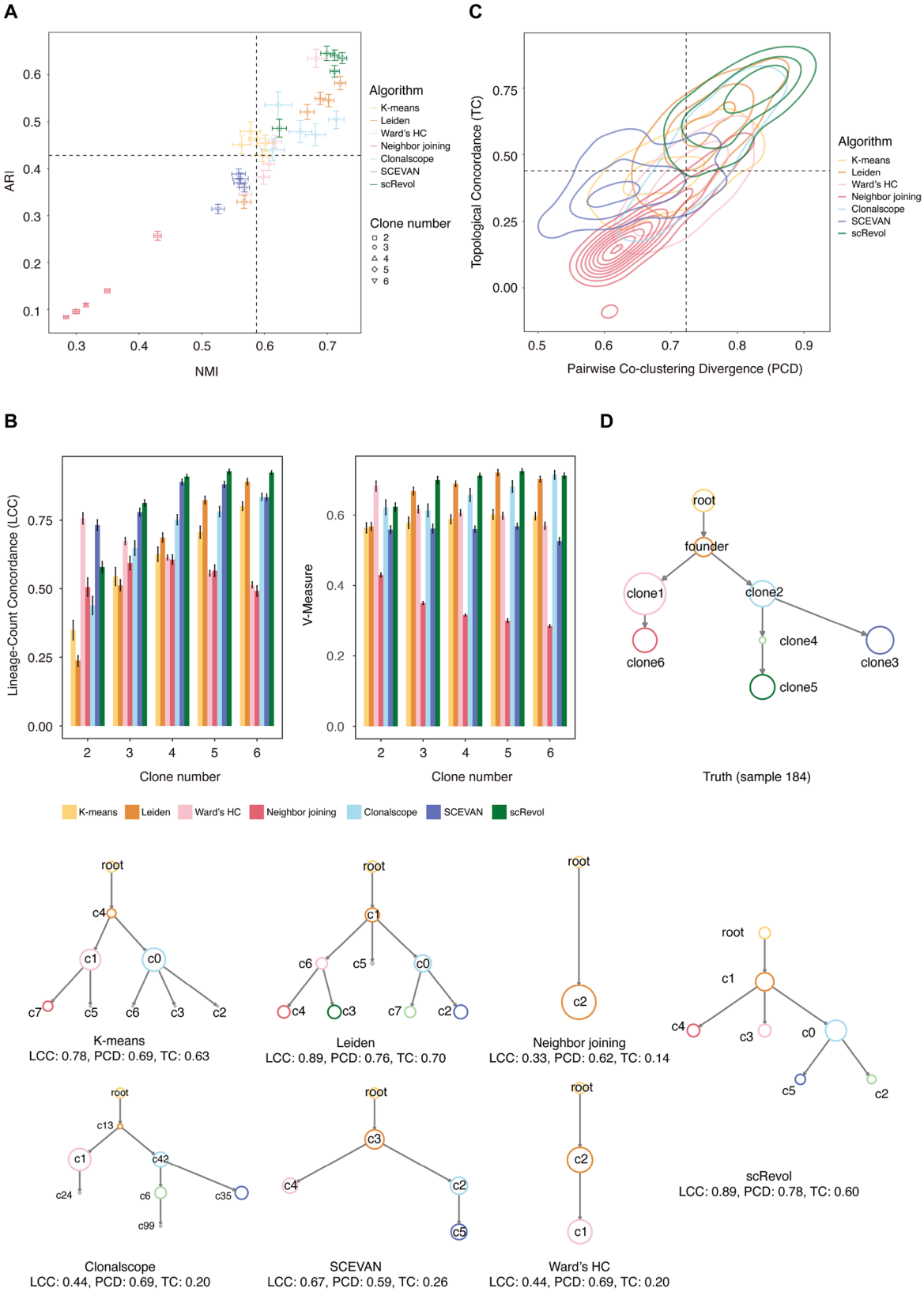

Fig. 2 (See legend on next page.)

(See figure on previous page.)

Fig. 2 Performance evaluation on simulated data. **A** Mean ARI and NMI values for seven algorithms across 485 simulated datasets. The x-axis denotes the mean NMI, and the y-axis denotes the mean ARI. Dashed lines represent the mean performance across all algorithms. Error bars indicate the standard error (s.e.) calculated from 485 datasets, with values presented as mean ± s.e. The optimal error rate for each algorithm is highlighted. **B** Comparison of algorithm performance as measured by LCC and V-measure metrics. Bars represent the mean values, with error bars indicating the standard error (mean ± s.e.) calculated across 485 datasets. **C** Density plots showing the PCD and TC scores across algorithms. Higher PCD values indicate greater consistency among clones, and higher TC values reflect closer alignment with the true clonal tree topology. **D** Reconstructed clonal tree for sample 184. The top panel shows the ground-truth clonal relationships; the bottom panel shows predictions from seven methods. Node labels indicate clone identifiers; circle area reflects clone size; panel annotations report method-level LCC, PCD and TC

(detailed in the Methods), we calculated the feature space for each cluster and measured feature similarity using the Jaccard index. As shown in Additional file 1: Fig. S2, scRevol effectively identifies subclonal populations and captures associated features. To examine whether clustering quality translated to evolutionary structure, we quantified clonal consistency with PCD and assessed lineage topology with TC, which captures both ancestor-descendant and non-ancestral relationships (Methods). In the PCD-TC space, scRevol occupied the high-agreement region more consistently than alternatives (Fig. 2C). For qualitative inspection, two randomly chosen samples are shown (Fig. 2D and Additional file 1: Fig. S3). In sample 184, scRevol recovered the true clone count (LCC = 0.89) and attained the highest PCD (0.78), with a moderate TC (0.60). In sample 300, scRevol achieved LCC = 0.67, PCD = 0.82, and TC = 0.76, placing it within the top range across methods. These examples indicate that scRevol reconstructs clonal topology and relative clone sizes with high agreement to the ground truth.

Because the RNA branch introduces additional noise and sparsity relative to the simulated CNV matrices, we also benchmarked only methods that operate directly on CNV input, specifically scRevol and the clustering baselines, using CNV data with controlled noise levels. Across four noise levels (0.25–0.40), scRevol generally achieved the highest ARI and NMI (Additional file 1: Fig. S4). In the TC-PCD space, it remained in the high-agreement region, whereas Ward's hierarchical clustering and NJ were consistently located at lower TC/PCD values (Additional file 1: Fig. S5). LCC and V-measure showed the same trend, with scRevol more reliably preserving the true clone count and cell-clone assignments at higher noise levels (Additional file 1: Fig. S6). These CNV-only experiments indicate that the observed gains arise from the RL-guided clustering and tree-search procedure rather than the RNA-to-CNV conversion.

Ablation studies on PPO batch size, clipping ratio, tree-construction strategy (MST, NJ, UPGMA) and reward design (silhouette, DBI, CHI) identified the default setting (100 samples per update, clip = 0.2, MST, silhouette) as near-optimal across LCC, PCD and TC (Additional file 2: Table S1-S4; Methods). Runtime analysis indicated only a moderate overhead relative to lightweight baselines (Additional file 2: Table S5).

In summary, within the scope of these simulations, scRevol accurately recovers cell-clone structure, reproduces lineage topology and remains stable under added noise, offering a practical balance between accuracy and computational cost for clonal clustering and lineage reconstruction.

scRevol identifies clonal groups and elucidates metastatic heterogeneity

We evaluated scRevol on lineage-tracing data from Quinn et al., focusing on its ability to recover clonal structure [29]. In the original study, the authors utilized single-cell lineage tracing to investigate metastatic rates and the driving factors behind metastasis, revealing significant heterogeneity. Based on cell counts and metastatic status, we selected seven lineage groups along with pre-implantation cells for analysis (details in Methods). As shown in Fig. 3A, scRevol identified 10 distinct clonal groups, five of which exhibited strong associations with specific lineages (e.g., c0 with CP017, c3 with CP026). Statistical analysis using Chi-square (Chi^2 = 4012, p-value < 0.01) and Cramér's V (0.66) confirmed these associations. The TreeMetRate metric, derived from the original study, indicated that clonal groups c0 and c3 exhibited significant metastatic potential. We visualized clonal groups and lineages on a standard PCA-UMAP embedding (Additional file 1: Fig. S7B). Baseline methods showed modest agreement with lineage labels (Leiden ARI = 0.396; K-means ARI = 0.344; Clonalscope ARI = 0.284; SCEVAN ARI = 0.219). In comparison, the scRevol embedding delineated the pre-implantation lineage (c8) and the highly metastatic groups (c0, c3) more clearly, yielding ARI of 0.443 and a silhouette score of 0.447 (Fig. 3B and Additional file 1: Fig. S7A).

To further investigate the relationship between clonal groups and genetic features, we performed feature selection using information gain and frequency thresholds (95% information gain cutoff, 297 genes). As shown in the clone-wise kernel density estimate plots (Fig. 3B) and the global landscape (Additional file 1: Fig. S8 and Fig. S9), clonal groups c0 and c3 were distinct from others, with most CNV features exhibiting normal levels, and only a small proportion showing gain or loss. In contrast, many other clonal groups show a higher prevalence of copy number gains. This pattern is consistent with the CNV heatmap in Fig. 3C and is more readily seen when

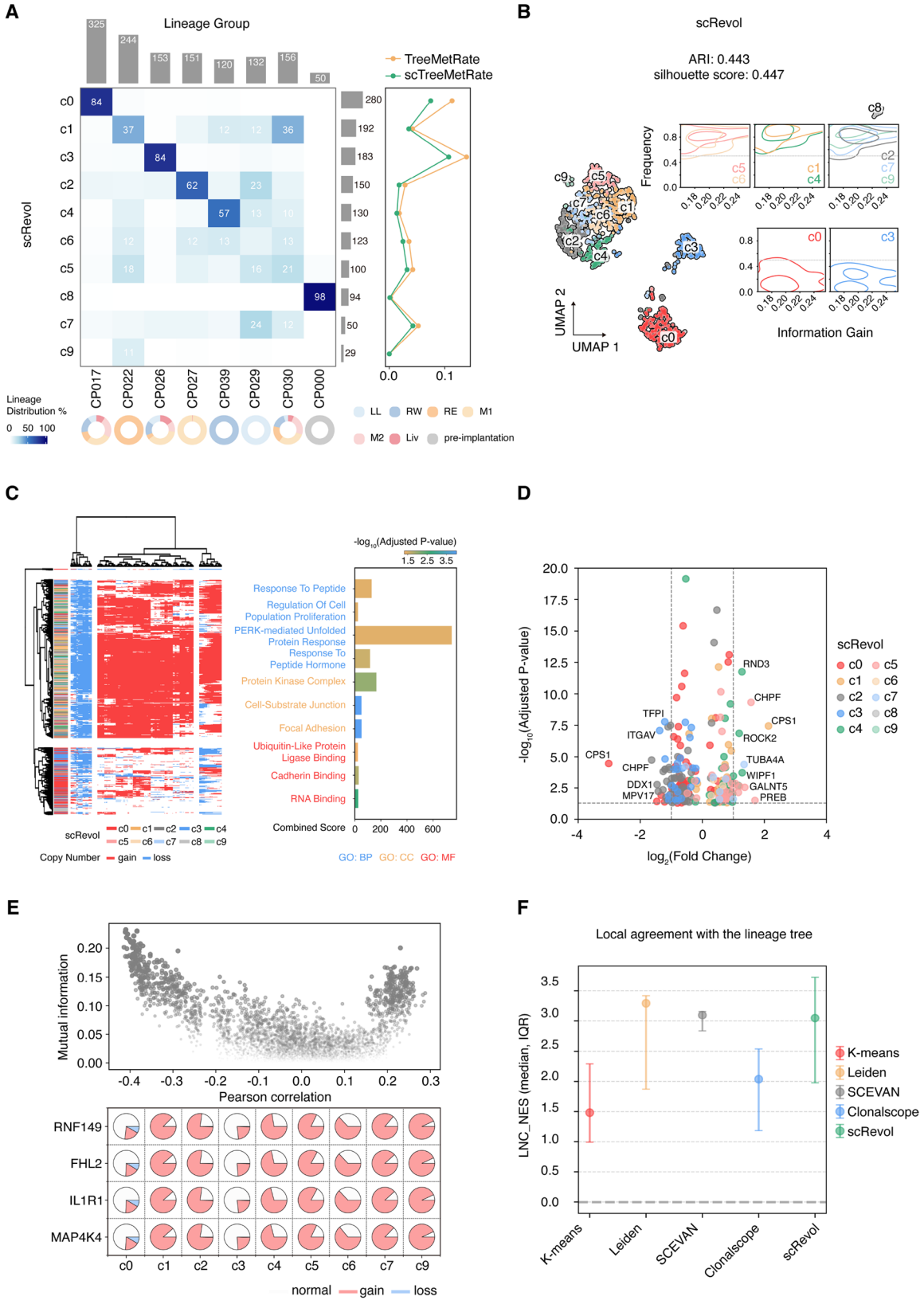

Fig. 3 (See legend on next page.)

(See figure on previous page.)

Fig. 3 Clonal group characterization and metastatic heterogeneity across lineages. **A** Heatmap showing the relationship between clonal groups (rows) and lineage groups (columns). Lineage percentages are annotated within the heatmap, and lineage distributions are visualized as donut plots below. The line chart on the right displays TreeMetRate and scTreeMetRate metrics for each clonal group. Data are presented as mean values. **B** scRevol embedding and clone-specific feature enrichment. Left: embedding generated by scRevol, coloured by inferred clonal groups (ARI and silhouette score shown above). Right: clone-wise KDE plots in the information gain-frequency plane; dashed lines mark the fixed thresholds, and contours highlight feature density in the selected regions (upper right: high information gain and high frequency; lower right: high information gain and low frequency). **C** Heatmap of discretized CNV states (loss, normal, gain) across 297 selected genes, with cells annotated by inferred clonal groups. Functional enrichment was performed on the pooled set of selected CNV-associated genes, combining genes that show gain or loss across clones. The bar chart on the right highlights pathway significance based on adjusted p-values and combined scores. **D** Volcano plot of differential gene expression across clonal groups. Top-ranking genes are highlighted based on significance and fold change. The x-axis represents $\log_2$(fold change), and the y-axis represents -$\log_{10}$(adjusted p-value). **E** Association between gene-level CNV and metastatic potential. Each point in the scatter plot (top) represents one gene (CNV feature) and is positioned by its Pearson correlation and mutual information with TreeMetRate. The bottom panel shows, for the four top-ranked genes, the distribution of CNV states (normal, gain, loss) within each inferred clonal group. **F** Tree-locality across methods (hybrid prior). Median LNC_NES by method; error bars show the interquartile range. Scores were computed on hybrid-prior lineage trees. The dashed line denotes the null expectation (NES = 0). Methods: K-means (red), Leiden (orange), SCEVAN (grey), Clonalscope (blue), and scRevol (green)

cells are ordered by inferred clonal assignment (Additional file 1: Fig. S10), which allows more direct comparison of CNV profiles across clonal groups. To assess the functional relevance of these genetic features, we conducted gene set enrichment analysis (see Methods, adjusted p-value < 0.05). Among the pathways identified, cell-substrate junctions and focal adhesion were the most significant based on their adjusted p-values. Additionally, the pathway with the highest combined score was PERK-mediated unfolded protein response, all of which are associated with metastasis [42, 43].

We then integrated gene expression to examine clone-specific traits. Using the Wilcoxon rank-sum test, we analyzed the differential expression of the 297 selected genes across clonal groups. As shown in Fig. 3D, clonal group c0 exhibited low expression of *CPS1*, while c1 showed high expression. This aligns with recent work by Chen et al., which suggests that suppression of *CPS1* (as observed in c0) may enhance metastatic potential, while overexpression (as in c1) promotes proliferation [44]. Additionally, clonal group c4 showed high expression of *RND3* and *ROCK2*, while c3 exhibited decreased expression of *TFPI* and *ITGAV*, all of which have been implicated in metastasis [45–48]. To confirm the relevance of these genes to metastasis, we calculated Pearson correlation coefficients and mutual information between each gene and the metastatic metric (TreeMetRate). In Fig. 3E (top), each point represents one gene and is positioned by its Pearson correlation and mutual information with TreeMetRate. We identified 121 metastasis-associated genes (absolute correlation > 0.3, mutual information > 0.15), of which 120 were among the 297 selected genes. The four genes (*MAP4K4*, *IL1R1*, *FHL2*, and *RNF149)* showing the strongest correlations had low copy number variation frequencies in clonal groups c0 and c3. In combination with the previous analyses, the large number of metastasis-associated genes suggests that each clonal group exhibits a complex metastatic pattern, which contributes to the metastatic heterogeneity observed across different lineages.

Having established global associations across lineages, we next examined whether RNA-derived clones are consistent with the recorder tree within individual lineages. This analysis is motivated by the intuition that recorder-adjacent leaves should, on average, share a clone more often than distant leaves. Lineage-wise panels show that clone assignments from all methods only partially align with the same hybrid-prior tree (Additional file 1: Fig. S11). Even with fixed tip order and the original annotations for metastatic rate (scTreeMetRate) and anatomical location, adjacent leaves often differ in rate or tissue, indicating that the tree does not define a unique subclonal partition. Consequently, the recorder tree is not appropriate as a hard ground truth. We therefore treat it as a weak reference and assess tree locality using a permutation-normalized local neighbourhood consistency score (LNC_NES). Across the selected lineages, scRevol achieved consistently high LNC_NES, matching or exceeding baseline methods (Fig. 3F). This indicates that RNA-only clusters from scRevol align well with local neighborhoods on the recorder tree, supporting lineage-consistent structure without relying on hard tree-derived labels.

Collectively, these analyses show that scRevol captures clonal structure and metastatic heterogeneity, links clone assignments to gene-level CNV and expression signals, and preserves local phylogenetic neighborhoods in a weak-supervision setting.

CNV profiling reveals tumor differentiation and subclonal evolution

Tumor tissues exhibit substantial heterogeneity within their complex microenvironment, posing challenges for distinguishing tumor cells from normal cells [49]. To evaluate scRevol's capability in this context, we analyzed rigorously controlled single-cell transcriptomic datasets [32]. We inferred CNV profiles for individual cells using inferCNV [31], which then served as input for scRevol. UMAP visualization revealed distinct clusters corresponding to tumor and normal cells, with tumor clone

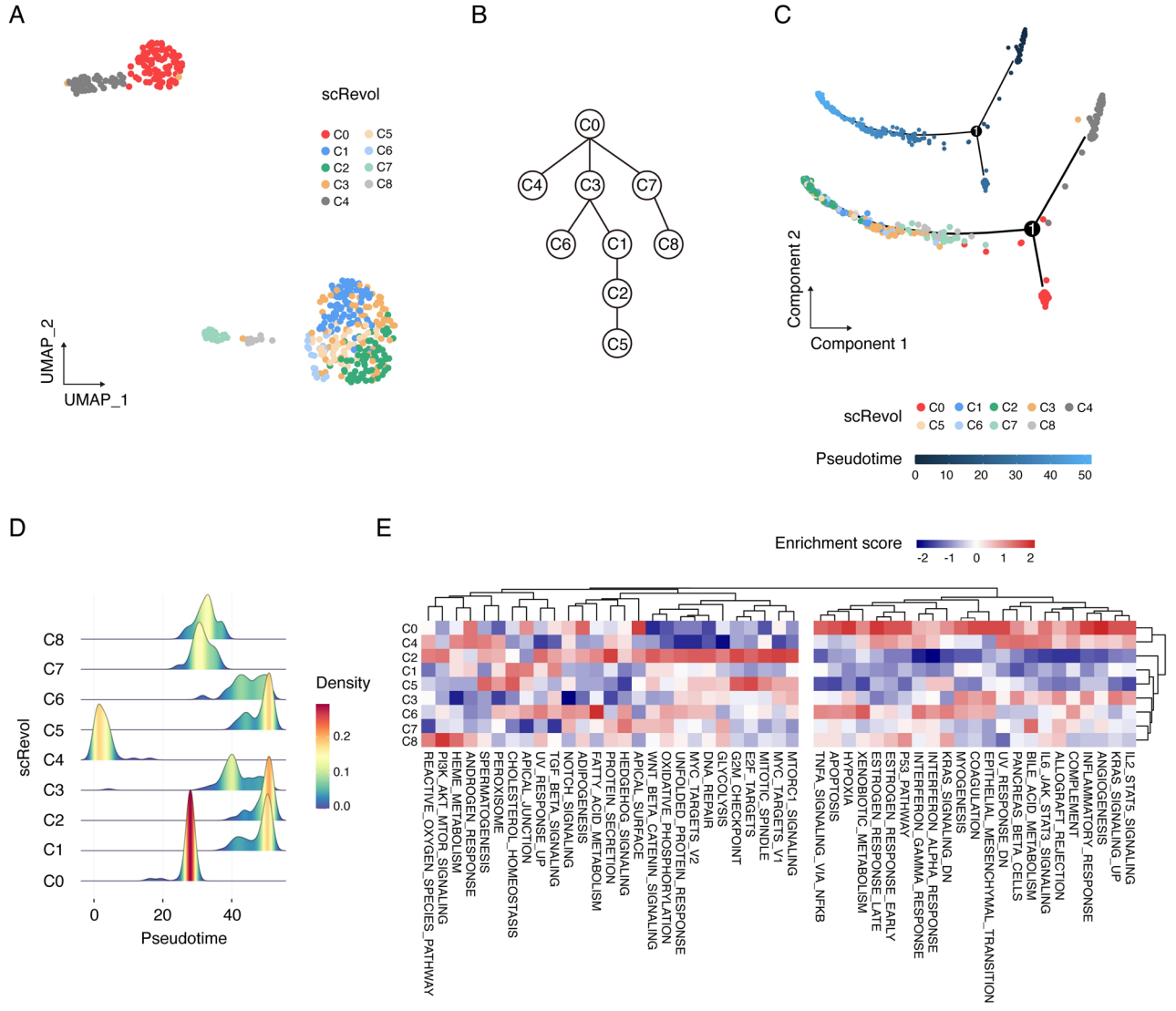

Fig. 4 Delineating clonal evolution and functional dynamics via scRevol analysis. Distinction between normal and tumor cells. **A** UMAP visualization depicting cellular distribution, based on embeddings and clones identified by scRevol using the SOL1307 dataset. **B** Topological structure of the clonal evolution tree, as inferred by scRevol. **C** Left: UMAP plot illustrating the pseudotime trajectory of clones. Right: Distribution plot of the pseudotime trajectory across various stages. **D** Mountain plot depicting the pseudotime trajectory of clones. **E** Gene Set Variation Analysis (GSVA) results, displaying enrichment scores for multiple gene sets across different clones

purities exceeding 0.96 under the defined analysis framework, as shown in Additional file 1: Fig. S12.

To further explore intratumoral heterogeneity, characterized by the coexistence of multiple subclones within a tumor, we re-clustered cells from the SOL1307 dataset after excluding populations containing over 80% normal cells. This refined analysis facilitated the construction of a clonal evolution tree, illustrating hierarchical relationships among tumor subclones (Fig. 4A and B). We next performed pseudotime analysis on the original expression profiles to relate CNV-defined subclones to continuous transcriptional trajectories. Along the inferred pseudotime axis, cells assigned to different subclones occupied distinct regions, with C4 enriched toward earlier positions and C0 and additional subclones distributed toward later positions (Fig. 4C and D). In line with the bifurcating structure of the CNV-based clonal tree, the pseudotime distributions were consistent with a separation of trajectories toward C4 on one branch and toward C3 and C7 on another. Similarly, cells from C3 tended to occupy earlier pseudotime positions than those from C6 and C1, and C2 tended to precede C5 along the same axis. These patterns support the consistency of scRevol's reconstructed trajectories. Functional analyses identified distinct biological processes associated with each subclone. Subclone C0 was enriched in interferon and inflammatory response pathways, indicative of immune cell infiltration during early tumor development.

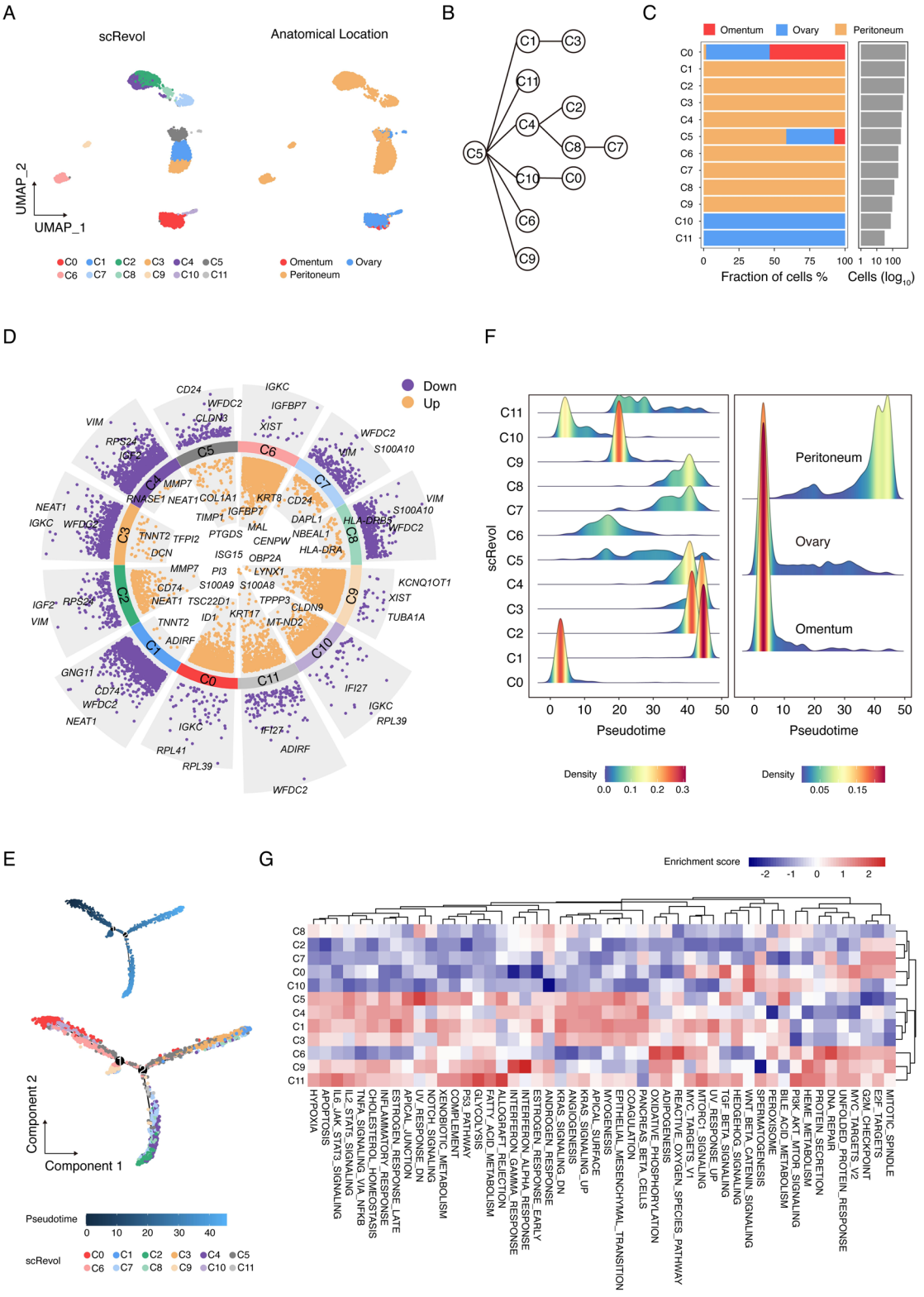

Fig. 5 (See legend on next page.)

(See figure on previous page.)

Fig. 5 scRevol-inferred clonal dynamics and functional profiles in HGSOC. **A** UMAP visualization of clone distribution across various anatomical sites, as inferred from the dataset using scRevol. **B** Topological structure of the inferred clonal evolution tree. **C** Distribution of clones across different anatomical sites, including the omentum, ovary, and peritoneum. **D** Circular plot illustrating differentially expressed genes for each clone. The top three upregulated (yellow) and bottom three downregulated (purple) genes are annotated. **E** UMAP plot depicting the pseudotime trajectory of clones, which are divided into three major branches. **F** Mountain plot depicting the pseudotime trajectory of clones, with color gradients representing density (red for high and blue for low). **G** Heatmap presenting GSVA results, used to determine the biological functions associated with each clone

In contrast, subclone C2 was linked to DNA repair and G2M checkpoint pathways, suggesting increased proliferative potential and malignancy as tumor evolution progressed (Fig. 4E).

For context, we additionally provide clonal tree visualizations derived from SCEVAN and Clonalscope on the same dataset in Additional file 1: Fig. S13. We also performed a side-by-side comparison with SCEVAN and Clonalscope focusing on a biologically interpretable signal. As shown in Additional file 1: Fig. S14, scRevol yields a clone enriched for *BRCA1/BRCA2*-high cells. We used *BRCA1/BRCA2* as canonical homologous-recombination genes with established relevance in ovarian cancer; high expression provides a concise, function-linked readout of coherence. Under the same preprocessing and thresholds, SCEVAN and Clonalscope distribute *BRCA1/BRCA2*-high cells more diffusely across multiple clusters, with no single cluster showing comparable enrichment.

Together, these results indicate that scRevol resolves subclonal structure in a manner consistent with trajectory-based ordering, while recovering a functionally interpretable clone marked by *BRCA1/BRCA2*-high expression in this dataset.

Revealing subclonal structures across primary and metastatic lesions

Metastatic lesions originate from specific clonal populations within the primary tumor, acquiring genetic and phenotypic adaptations to survive and proliferate in distant microenvironments. Tracing these metastatic-initiating clones provides insights into tumor progression and metastasis [50–53]. To investigate this, single-cell transcriptomic datasets from ovarian cancer, encompassing both primary and metastatic lesions, were analyzed. As depicted in Fig. 5A and B, data from multiple anatomical locations, including the ovary, omentum, and peritoneum, were classified into distinct tumor clones using scRevol. Notably, subclone C5 was identified across all three sites, suggesting a common origin in the primary lesion followed by dissemination to metastatic locations (Fig. 5C). In addition to this shared subclone, unique subclones were detected in specific microenvironments. Subclones C1, C2, C3, C4, C6, C7, C8, and C9 were exclusive to the peritoneum, while subclones C10 and C11 were specific to the omentum. Differential gene expression analysis highlighted molecular features of these subclones, with the most highly upregulated and downregulated genes annotated for each, as illustrated in Fig. 5D. Compared to C5, these unique subclones emerged later in the evolutionary timeline, reflecting the dynamic nature of tumor evolution and the development of tissue-specific clonal structures.

Pseudotime analysis placed C5 near the central region of the inferred trajectory, with cells from C0 and C10 tending to occupy earlier positions and other subclones distributed toward later regions, as shown in Fig. 5E and F. Tumor cells from the ovarian primary site were positioned earlier along the pseudotime axis than those from the peritoneum.Together, the CNV-defined subclones and their distribution along pseudotime provide a coherent view of multi-site progression across primary and metastatic lesions. Functional analysis in Fig. 5G biological distinctions among subclones. Subclones C5, C4, and C3 were enriched in interferon and inflammatory response pathways, indicating active immune cell infiltration during the early stages of tumor evolution. By contrast, subclones C0 and C10 were associated with TGFβ and WNT signaling, consistent with immune suppression in later stages.

In summary, scRevol provides a framework for analyzing multi-site tumor evolution. It identifies subclones shared between primary and metastatic lesions and detects tissue-specific subclone populations, offering insights into tumor progression and metastasis.

Discussion

In this study, we present scRevol, a computational framework designed to reconstruct tumor evolutionary trajectories and assess clonal heterogeneity by integrating reinforcement learning-based clustering with evolutionary tree construction. Analyses of both simulated and clinical datasets underscore the method's capacity to resolve clonal structures from CNV data and to infer biologically meaningful trajectories. Specifically, in the lineage tracing dataset, scRevol identified clonal groups closely associated with high metastatic potential, in agreement with the original lineage-tracing observations and offering additional insights into the genetic profiles of these clones. Additionally, analyses of ovarian cancer datasets revealed the progression of metastatic clones across distinct anatomical sites, providing a more nuanced view of clonal relationships and metastatic potential. Together, these results highlight scRevol's

promise as a tool for unraveling tumor progression and metastatic heterogeneity.

Despite these encouraging outcomes, several areas for future improvement merit consideration. One key challenge lies in the limited use of lineage tracing data for benchmarking. Lineage tracing, which integrates CRISPR-induced edits with scRNA-seq, provides temporal lineage trajectories alongside transcriptomic profiles from the same cells. Although scRevol relies on scRNA-seq data for CNV inference and modeling, lineage tracing datasets offer a distinct advantage for benchmarking clonal evolution models because of their relatively high temporal resolution. Unlike simulated datasets, lineage tracing captures the in vivo dynamics of tumor evolution and serves as a ground truth for evolutionary reconstructions. However, discrepancies emerge due to differences in temporal resolution: CRISPR-induced edits often accumulate more rapidly than CNV events, leading to inconsistencies between lineage trajectories and CNV-based models. For example, subclones originating from the same progenitor cell may proliferate extensively without acquiring detectable CNV changes, complicating dataset integration for evaluation purposes. While preliminary attempts to address these challenges have been made, the study did not fully exploit the potential of lineage tracing data, which could have added substantial credibility to the results. Further methodological advancements are needed to resolve these integration challenges and harness lineage tracing data effectively for benchmarking clonal evolution models.

A promising avenue for future work involves the development of biologically informed reward functions. While unsupervised metrics such as the silhouette score provide a useful signal of cluster compactness and separation, incorporating prior biological knowledge into the reward could guide the policy toward solutions that are more biologically coherent and easier to interpret. Further, an end-to-end formulation that jointly optimizes cluster assignments and tree structure may reduce redundancy and improve both computational efficiency and biological fidelity. In the current implementation, the tree is reconstructed from distances between clone-centroid CNV profiles to retain interpretability. A natural extension is to explore whether the learned latent embedding can provide an alternative notion of clone-to-clone proximity, and whether embedding-based distances improve robustness or biological coherence when integrated into such an end-to-end objective.

Conclusions

Collectively, our results present scRevol as an RL-based framework that integrates CNV-guided clustering with evolutionary tree reconstruction to infer tumor trajectories from single-cell transcriptomic data. Across simulated datasets, scRevol recovered clonal composition and lineage topology with stable performance under varying noise conditions, and in lineage-tracing and ovarian cancer cohorts it delineated clones that reflect metastatic heterogeneity, subclonal diversification and site-specific functional programs. By jointly analyzing CNV profiles, gene expression and pathway-level signatures, the framework offers a practical means to resolve clonal architecture and approximate progression trajectories in diverse tumor settings. Further refinement, including deeper integration of lineage-tracing information and biologically informed reward functions, will be important to extend its scope and interpretability. Within these bounds, scRevol provides a useful addition to computational approaches for studying tumor evolution and clonal dynamics at single-cell resolution.

Abbreviations

SNVs	Single-nucleotide variants
CNVs	Copy number variations
UMAP	Uniform manifold approximation and projection
RL	Reinforcement learning
scRNA-seq	Single-cell RNA sequencing
MDP	Markov decision process
PPO	Proximal policy optimization
KL	Kullback-Leibler
MST	Minimum spanning tree
Ward's HC	Ward's hierarchical clustering
NJ	Neighbor joining
UPGMA	Unweighted pair group method with arithmetic mean
LCC	Lineage-count concordance
PCD	Pairwise co-clustering divergence
TC	Topological concordance
ARI	Adjusted rand index
NMI	Normalized mutual information
IQR	Interquartile range
PCA	Principal component analysis
PCs	Principal components
SNN	Shared nearest neighbor

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s13073-026-01648-4.

Supplementary Material 1: Supplementary figures. Supplementary figures supporting the benchmarking analysis, lineage-tracing analysis, and ovarian cancer case study. Format: DOCX (.docx).

Supplementary Material 2: Supplementary tables. Supplementary tables containing ablation results, runtime analysis, and other supporting quantitative results. Format: DOCX (.docx).

Acknowledgements

Not applicable.

Authors' contributions

M.Y., Y.Y. and H.Y. conceived and designed the study; Z.Z., X.D. and C.X. designed, implemented, and optimized the model; Q.H. and Y.W. performed data analysis; Q.H., Z.Z., Y.W. and X.D. drafted the manuscript. All the authors contributed significantly to data interpretation and editing of the manuscript. All authors reviewed and approved the final manuscript.

Funding

Not applicable.

Data availability

The simulated benchmark dataset generated and analyzed in this study is available in Figshare (https://figshare.com/articles/dataset/Simulated_Data_for_Benchmark/30615932) [54]. The single-cell lineage dataset analyzed in this study is available from the Gene Expression Omnibus (GEO) under accession number GSE161363, including samples GSM4905334 (5k-TS), GSM4905335 (5k-RNA), and GSM4905342 (LM0-RNA) (https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE161363) [29]. The single-cell RNA-seq dataset used in Figs. 4 and 5 was obtained from the Diether Lambrechts laboratory resource for high-grade serous tubo-ovarian cancer (https://lambrechtslab.sites.vib.be/en/high-grade-serous-tubo-ovarian-cancer-refined-single-cell-rna-sequencing-specific-cell-subtypes) [32]. Figure 4 used sample SOL1307, whereas Fig. 5 used a larger set of samples from the same resource, including SOL1303, SOL1306, SOL1307, SOL003, SOL006, SOL008, SOL012, and SOL016. This dataset comprised 9,693 cells derived from omentum, ovary, and peritoneum across stages I, III, and IV, with all samples classified as HGSOC. The source code for scRevol is available at GitHub: https://github.com/melobio/scRevol/tree/main [55].

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare no competing interests.

Received: 21 April 2025 / Accepted: 2 April 2026
Published online: 18 April 2026

References

1. Dentro SC, Leshchiner I, Haase K, Tarabichi M, Wintersinger J, Deshwar AG, et al. Characterizing genetic intra-tumor heterogeneity across 2,658 human cancer genomes. Cell. 2021;184(8):2239–54.
2. van de Donk NWCJ, Pawlyn C, Yong KL. Multiple myeloma. Lancet. 2021;397(10272):410–27.
3. Denisenko E, de Kock L, Tan A, Beasley AB, Beilin M, Jones ME, et al. Spatial transcriptomics reveals discrete tumour microenvironments and autocrine loops within ovarian cancer subclones. Nat Commun. 2024;15(1):2860.
4. Tirier SM, Mallm J-P, Steiger S, Poos AM, Awwad MHS, Giesen N, et al. Subclone-specific microenvironmental impact and drug response in refractory multiple myeloma revealed by single-cell transcriptomics. Nat Commun. 2021;12(1):6960.
5. Roth A, Khattra J, Yap D, Wan A, Laks E, Biele J, et al. Pyclone: statistical inference of clonal population structure in cancer. Nat Methods. 2014;11(4):396–8.
6. Dang HX, White BS, Foltz SM, Miller CA, Luo J, Fields RC, et al. Clonevol: clonal ordering and visualization in cancer sequencing. Ann Oncol. 2017;28(12):3076–82.
7. Williams MJ, Werner B, Heide T, Curtis C, Barnes CP, Sottoriva A, et al. Quantification of subclonal selection in cancer from bulk sequencing data. Nat Genet. 2018;50(6):895–903.
8. Jahn K, Kuipers J, Beerenwinkel N. Tree inference for single-cell data. Genome Biol. 2016;17(1):86.
9. Ross EM, Markowetz F. Onconem: inferring tumor evolution from single-cell sequencing data. Genome Biol. 2016;17:1–14.
10. Zafar H, Tzen A, Navin N, Chen K, Nakhleh L. Sifit: inferring tumor trees from single-cell sequencing data under finite-sites models. Genome Biol. 2017;18:1–20.
11. Wang F, Wang Q, Mohanty V, Liang S, Dou J, Han J, et al. MEDALT: single-cell copy number lineage tracing enabling gene discovery. Genome Biol. 2021;22:1–22.
12. Guo L, Yi X, Chen L, Zhang T, Guo H, Chen Z, et al. Single-cell DNA sequencing reveals punctuated and gradual clonal evolution in hepatocellular carcinoma. Gastroenterology. 2022;162(1):238–52.
13. Mallory XF, Edrisi M, Navin N, Nakhleh L. Methods for copy number aberration detection from single-cell DNA-sequencing data. Genome Biol. 2020;21:1–22.
14. Wu C-Y, Rong J, Sathe A, Hess PR, Lau BT, Grimes SM et al. Cancer subclone detection based on DNA copy number in single-cell and spatial omic sequencing data. Nat Methods. 2025;22(9).
15. Elyanow R, Zeira R, Land M, Raphael BJ. Starch: copy number and clone inference from spatial transcriptomics data. Phys Biol. 2021;18(3):035001.
16. De Falco A, Caruso F, Su X-D, Iavarone A, Ceccarelli M. A variational algorithm to detect the clonal copy number substructure of tumors from scrna-seq data. Nat Commun. 2023;14(1):1074.
17. Li Y. Deep Reinforcement Learning: an overview. arXiv preprint arXiv:1701.07274. 2017.
18. Margolis GB, Yang G, Paigwar K, Chen T, Agrawal P. Rapid locomotion via reinforcement learning. Int J Robot Res. 2024;43(4):572–87.
19. Kiran BR, Sobh I, Talpaert V, Mannion P, Al Sallab AA, Yogamani S, et al. Deep reinforcement learning for autonomous driving: A survey. IEEE Trans Intell Transp Syst. 2021;23(6):4909–26.
20. Zhu Y, Mottaghi R, Kolve E, Lim JJ, Gupta A, Fei-Fei L, et al. editors. Target-driven visual navigation in indoor scenes using deep reinforcement learning. 2017: IEEE.
21. Ouyang L, Wu J, Jiang X, Almeida D, Wainwright C, Mishkin P, et al. Training language models to follow instructions with human feedback. Adv Neural Inf Process Syst. 2022;35:27730–44.
22. Sutton RS, Barto AG. Reinforcement learning: an introduction. Robotica. 1999;17(2):229–35.
23. Azer ES, Ebrahimabadi MH, Malikić S, Khardon R, Sahinalp SC. Tumor phylogeny topology inference via deep learning. Iscience. 2020;23(11):101655.
24. Ivanovic S, El-Kebir M. Modeling and predicting cancer clonal evolution with reinforcement learning. Genome Res. 2023;33(7):1078–88.
25. Schulman J, Wolski F, Dhariwal P, Radford A, Klimov O. Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347. 2017.
26. Puterman ML. Markov decision processes: discrete stochastic dynamic programming. Wiley; 2014.
27. Weiner S, Bansal MS. CNAsim: improved simulation of single-cell copy number profiles and DNA-seq data from tumors. Bioinformatics. 2023;39(7):btad434.
28. Bai X, Duren Z, Wan L, Xia LC. Joint inference of clonal structure using single-cell genome and transcriptome sequencing data. NAR Genomics Bioinform. 2024;6(1):lqae017.
29. Quinn JJ, Jones MG, Okimoto RA, Nanjo S, Chan MM, Yosef N, et al. Single-cell lineages reveal the rates, routes, and drivers of metastasis in cancer xenografts. Science. 2021;371(6532):eabc1944.
30. Wolf FA, Angerer P, Theis FJ. SCANPY: large-scale single-cell gene expression data analysis. Genome Biol. 2018;19:1–5.
31. Tickle TTI, Georgescu C, Brown M, Haas B. inferCNV of the Trinity CTAT Project. 2019. https://github.com/broadinstitute/inferCNV.
32. Olbrecht S, Busschaert P, Qian J, Vanderstichele A, Loverix L, Van Gorp T, et al. High-grade serous tubo-ovarian cancer refined with single-cell RNA sequencing: specific cell subtypes influence survival and determine molecular subtype classification. Genome Med. 2021;13:1–30.
33. Hao Y, Hao S, Andersen-Nissen E, Mauck WM, Zheng S, Butler A, et al. Integrated analysis of multimodal single-cell data. Cell. 2021;184(13):3573–87.
34. Korsunsky I, Millard N, Fan J, Slowikowski K, Zhang F, Wei K, et al. Fast, sensitive and accurate integration of single-cell data with Harmony. Nat Methods. 2019;16(12):1289–96.
35. Salcedo A, Tarabichi M, Buchanan A, Espiritu SMG, Zhang H, Zhu K et al. Crowd-sourced benchmarking of single-sample tumor subclonal reconstruction. Nat Biotechnol. 2025;43(4):581–92.
36. Kuhn HW. The hungarian method for the assignment problem. Nav Res Logist Q. 1955;2(1–2):83–97.
37. Fang Z, Liu X, Peltz G. Gseapy: a comprehensive package for performing gene set enrichment analysis in Python. Bioinformatics. 2023;39(1):btac757.
38. Qiu X, Mao Q, Tang Y, Wang L, Chawla R, Pliner HA, et al. Reversed graph embedding resolves complex single-cell trajectories. Nat Methods. 2017;14(10):979–82.
39. Yu G, Wang L-G, Han Y, He Q-Y. Clusterprofiler: an R package for comparing biological themes among gene clusters. OMICS. 2012;16(5):284–7.
40. Hänzelmann S, Castelo R, Guinney J. GSVA: gene set variation analysis for microarray and RNA-seq data. BMC Bioinformatics. 2013;14:1–15.
41. Love MI, Huber W, Anders S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biol. 2014;15:1–21.
42. Yayan J, Franke K-J, Berger M, Windisch W, Rasche K. Adhesion, metastasis, and inhibition of cancer cells: a comprehensive review. Mol Biol Rep. 2024;51(1):165.

43. Maas NL, Diehl JA. Molecular pathways: the PERKs and pitfalls of targeting the unfolded protein response in cancer. Clin Cancer Res. 2015;21(4):675–9.
44. Chen S, Tang Q, Hu M, Song S, Wu X, Zhou Y, et al. Loss of carbamoyl phosphate synthetase 1 potentiates hepatocellular carcinoma metastasis by reducing aspartate level. Adv Sci. 2024;11(45):2402703.
45. Paysan L, Piquet L, Saltel F, Moreau V. Rnd3 in cancer: a review of the evidence for tumor promoter or suppressor. Mol Cancer Res. 2016;14(11):1033–44.
46. Du Y, Lu S, Ge J, Long D, Wen C, Tan S, et al. ROCK2 disturbs MKP1 expression to promote invasion and metastasis in hepatocellular carcinoma. Am J Cancer Res. 2020;10(3):884.
47. Wang J, Xiao J, Wen D, Wu X, Mao Z, Zhang J, et al. Endothelial cell-anchored tissue factor pathway inhibitor regulates tumor metastasis to the lung in mice. Mol Carcinog. 2016;55(5):882–96.
48. Tang Y-L, Li G-S, Li D-M, Tang D, Huang J-Z, Feng H, et al. The clinical significance of integrin subunit alpha V in cancers: from small cell lung carcinoma to pan-cancer. BMC Pulm Med. 2022;22(1):300.
49. Dohmen J, Baranovskii A, Ronen J, Uyar B, Franke V, Akalin A. Identifying tumor cells at the single-cell level using machine learning. Genome Biol. 2022;23(1):123.
50. Zheng X, Wang X, Cheng X, Liu Z, Yin Y, Li X, et al. Single-cell analyses implicate ascites in remodeling the ecosystems of primary and metastatic tumors in ovarian cancer. Nat Cancer. 2023;4(8):1138–56.
51. Vázquez-García I, Uhlitz F, Ceglia N, Lim JLP, Wu M, Mohibullah N, et al. Ovarian cancer mutational processes drive site-specific immune evasion. Nature. 2022;612(7941):778–86.
52. Khan SU, Fatima K, Malik F, Kalkavan H, Wani A. Cancer metastasis: molecular mechanisms and clinical perspectives. Pharmacol Ther. 2023;250:108522.
53. Rogiers A, Lobon I, Spain L, Turajlic S. The genetic evolution of metastasis. Cancer Res. 2022;82(10):1849–57.
54. He Q. Simulated data for benchmark. Figshare. 2025. https://doi.org/10.6084/m9.figshare.30615932.
55. Chenxi Xie ZZ, Xinghao Du Q, He Y, Wang, scRevol. GitHub. 2025. https://github.com/melobio/scRevol/tree/main.

Publisher's note