scReadSim: a single-cell RNA-seq and ATAC-seq read simulator

Guanao Yan [1] and Jingyi Jessica Li [1,2,3,4,*]

Abstract

Benchmarking single-cell RNA-seq (scRNA-seq) and single-cell ATAC-seq (scATAC-seq) computational tools demands simulators to generate realistic sequencing reads. However, none of the few read simulators aim to mimic real data. To fill this gap, we propose scReadSim, a single-cell RNA-seq and ATAC-seq read simulator that allows user-specified ground truths and generates synthetic data (in FASTQ and BAM formats) by mimicking real data. We show that scReadSim's synthetic data resemble real data (sci-ATAC-seq and 10x Multiome) at both read and count levels. scReadSim's benchmark applications show that cellranger is a preferred scRNA-seq UMI deduplication tool, and HMMRATAC and MACS3 achieve top performance in scATAC-seq peak calling.

[1] Department of Statistics, University of California, Los Angeles, CA 90095-1554
[2] Department of Human Genetics, University of California, Los Angeles, CA 90095-7088
[3] Department of Computational Medicine, University of California, Los Angeles, CA 90095-1766
[4] Department of Biostatistics, University of California, Los Angeles, CA 90095-1772
[*] To whom correspondence should be addressed. Email: jli@stat.ucla.edu

Introduction

The rapid development of single-cell sequencing technologies enables the characterization of single cells' genomic, epigenomic, and transcriptomic features [1]. More than a thousand computational tools have been developed to analyze single-cell sequencing data [2], necessitating third-party benchmarking of computational tools. Realistic simulators are essential for fair benchmarking, because they can generate synthetic data that mimic real data and contain ground truths for evaluating computational tools.

Although many simulators have been developed for the two most popular single-cell sequencing technologies—single-cell RNA sequencing (scRNA-seq) and single-cell Assay for Transposase-Accessible Chromatin using sequencing (scATAC-seq), most of the existing simulators do not generate sequencing reads but instead directly simulate read counts in genes [3–7] or genomic regions [8]. (See [9] for a comprehensive review of scRNA-seq count simulators.) As such, these count simulators cannot be used to benchmark read-level bioinformatics tools that process reads stored in FASTQ or BAM files (Fig. 1), such as unique molecular identifier (UMI) deduplication tools for scRNA-seq data [10–12] and peak-calling tools for scATAC-seq data [13–16]. Benchmarking these tools require read simulators.

For scRNA-seq, minnow [17] and Dropify [18] are the only two read simulators to our knowledge. Note that only minnow has a software package publicly available; minnow requires a UMI count matrix as input but does not learn from real scRNA-seq reads. Moreover, minnow generates scRNA-seq reads only from annotated genes and transcripts, respectively. It thus cannot resemble real scRNA-seq data in the inter-genes (i.e., intergenic regions) that might be transcribed. Also, since the verification results for minnow were only at the UMI count level [17], it remains unclear how well the synthetic reads generated by minnow resemble real reads.

For scATAC-seq, SCAN-ATAC-Sim [19] is the only read simulator to our knowledge. Like minnow, SCAN-ATAC-Sim does not learn from real scATAC-seq reads. Instead, SCAN-ATAC-Sim generates scATAC-seq reads from bulk ATAC-seq reads under simplistic assumptions: different genomic regions (i.e., peaks called from bulk ATAC-seq reads) are treated as independent, and only up to 2 reads are allowed in each genomic region in every cell, regardless of the region length. In short, SCAN-ATAC-Sim is not designed as a simulator to generate synthetic scATAC-seq data by mimicking real data.

Motivated by the limitations of the above read simulators, we propose scReadSim as a scRNA-seq and scATAC-seq read simulator that generates realistic synthetic data by mimicking real data. scReadSim inputs a scRNA-seq or scATAC-seq BAM file and outputs synthetic reads in the FASTQ and BAM formats (Figs. 2–3). We verify that scReadSim generates synthetic scRNA-seq and scATAC-seq data that resemble real data at both the read and count levels. Moreover, as a flexible simulator, scReadSim enables users to specify open chromatin regions for generating synthetic scATAC-seq reads, and it also allows the cell number and sequencing depth to vary for the synthetic data.

Among scReadSim's myriad uses, we use two exemplar applications to highlight scReadSim's utility as a benchmarking tool. First, scReadSim mimics real scRNA-seq data by first generating realistic UMI counts and then simulating reads. Hence, the synthetic UMI count matrix—an intermediate output of scReadSim—serves as the ground truth for benchmarking scRNA-seq UMI deduplication tools such as UMI-tools [10], cellranger [11], and Alevin [12], which all process reads (in FASTQ) into a UMI count matrix. Our benchmarking demonstrates that cellranger is preferred for its greater accuracy and higher computational efficiency. Second, for scATAC-seq, scReadSim provides ground-truth peaks and non-peaks by learning from user-specified trustworthy peaks and non-peaks or calling trustworthy peaks and non-peaks from real data. The ground-truth peaks provided by scReadSim allow benchmarking peak-calling tools such as MACS3 [13], HOMER [14], HMMRATAC [15], and SEACR [16]. Our benchmarking results show that HMMRATAC and MACS3 are the top performers.

Results

scReadSim generates realistic synthetic scRNA-seq reads

To verify scReadSim's ability to generate realistic scRNA-seq reads, we deploy scReadSim to the 10x single-cell Multiome dataset (the RNA-seq modality only, see "Availability of data and materials") and compare scReadSim's synthetic data with the real data at two levels: UMI counts and read sequences.

At the UMI count level, scReadSim uses the count simulator scDesign2 [6] to generate UMI counts. The current version of scReadSim assumes that cells belong to discrete cell types, but scReadSim is adaptive to scDesign3 [20], which generalizes scDesign2 by allowing cells to follow continuous trajectories or have spatial locations; note that scDesign2 is equivalent to scDesign3 when cells are in dicrete types. Since scDesign2 has been verified for generating realistic scRNA-seq counts [9], we only conduct a confirmation study by showing that scReadSim's synthetic UMI counts resemble real UMI counts in three aspects: (1) the distributions of six summary statistics, including four gene-level statistics (mean, variance, coefficient of variance, and zero proportion) and two cell-level statistics (zero proportion and cell library size) (Fig. S1A); (2) correlations among top expressed genes (Fig. S1B); (3) cells' UMAP embeddings (Fig. S1C).

At the read sequence level, we evaluate the resemblance of scReadSim's synthetic reads to real reads in three aspects. First, scReadSim's synthetic reads preserve the k-mer spectra, which describe the distribution of k-mers' frequencies (i.e., numbers of occurrences) in reads. Fig. 4A shows similar k-mer spectra in synthetic reads and real reads. Second, scReadSim can introduce substitution errors to imitate those observed in real reads. Fig. 4B shows that synthetic reads mimic real reads in terms of the substitution error rate per base call. Third, after read alignment, the pseudo-bulk read coverage of synthetic reads mimics that of real reads (Fig. 4C).

scReadSim generates realistic synthetic scATAC-seq reads

In scATAC-seq read generation, scReadSim provides ground-truth peaks and non-peaks for benchmarking purposes, by learning from user-specified or scReadSim-identified (under stringent criteria) peaks and non-peaks that are deemed trustworthy in real data (Fig. 3; Methods). In generating synthetic scATAC-seq data, scReadSim sets trustworthy peaks and non-peaks as ground-truth peaks and non-peaks, respectively. Since real data contain gray areas (i.e., regions complementary to trustworthy peaks and non-peaks) whose read coverages are between those of trustworthy peaks or non-peaks, scReadSim converts these gray areas to ground-truth non-peaks in the synthetic data generation (Methods). We verify that scReadSim generates realistic synthetic scATAC-seq reads at two levels: read counts and read sequences, by mimicking two real datasets: 10x single-cell Multiome dataset (the ATAC modality only) and sci-ATAC-seq (see "Availability of data and materials").

At the read count level, we verify that the synthetic scATAC-seq count matrix mimics the real count matrix in four aspects: (1) the distributions of six summary statistics, including four peak-level statistics (mean, variance, coefficient of variance, and zero proportion) and two cell-level statistics (zero proportion and cell library size) (Figs. S2A and S3A; note that scReadSim preserves the distributions of peak-level statistics better than those of cell-level statistics because scDesign2, the internal count simulator used in scReadSim, was designed to preserve peaks' count distributions and treat cells as random observations instead of fixed features [6]); (2) the Reads Per Kilobase Million (RPKM) distributions of synthetic reads resembling those of real reads in the trustworthy peaks and non-peaks (Figs. S2B and S3B); (3) correlations among top open peaks (Figs. S2C and S3C); (4) cells' UMAP embeddings (Figs. S2D and S3D).

At the read sequence level, we evaluate the resemblance of scReadSim's synthetic scATAC-seq reads to real reads in five aspects. First, synthetic reads mimic real reads in terms of k-mer spectra (Figs. 5A and S4A). Second, synthetic reads preserve the distribution of fragment sizes in real reads (Figs. 5B and S4B); it is known that the distribution of fragment sizes results from the pattern of chromatin accessibility and is thus meaningful [21]. Third, synthetic reads maintain the substitution error rate per base call in real reads (Figs. 5C and S4C). Fourth, the peak calling method MACS3 achieves consistent peak calling results using real and synthetic scATAC-seq data (Figs. 6A–B and S5A–B; note that the consistency is higher for the 10x single-cell Multiome dataset (the ATAC-seq modality only), which is less sparse than the sci-ATAC-seq dataset). Fifth, after read alignment, the read coverage of synthetic reads mimics that of real reads in the trustworthy peaks and non-peaks (Figs. 6C and S5C); in particular, the synthetic read coverage of the ground-truth non-peaks converted from gray areas mimics that of the nearby trustworthy non-peaks.

scReadSim also allows users to design the ground-truth peaks and non-peaks arbitrarily in the synthetic scATAC-seq data (Fig. 7). We show that the synthetic scATAC-seq data with user-designed ground-truth peaks and non-peaks mimic the trustworthy peaks and non-peaks in real data, thus providing reliable ground truths for benchmarking computational tools that process scATAC-seq reads (Fig. S6).

scReadSim allows varying cell numbers and sequencing depths

scReadSim flexibly allows users to vary cell numbers and sequencing depths to generate synthetic data. For UMI-based scRNA-seq data, scReadSim defines the sequencing depth as the number of UMIs; for non-UMI-based scRNA-seq data and scATAC-seq data, scReadSim defines the sequencing depth as the number of sequencing reads in a BAM file (with duplicated reads resulted from Polymerase Chain Reaction (PCR) removed).

For demonstration purposes, we deploy scReadSim to a 10x single-cell Multiome dataset (the RNA-seq modality only, see "Availability of data and materials") and vary the cell number or the sequencing depth (Fig. 8). Fig. 8A corresponds to the synthetic datasets scReadSim generates with a fixed sequencing depth but increasing cell numbers $1{,}219$ $(0.25\times)$, $2{,}438$ $(0.5\times)$, $4{,}877$ $(1\times$, the original cell number), $9{,}754$ $(2\times)$, and $19{,}508$ $(4\times)$. For a fair visual comparison, we downsample every synthetic dataset to $1{,}219$ cells $(0.25\times)$ in Fig. 8A, which shows that the read coverage of $1{,}219$ synthetic cells decreases as the synthetic cell number increases, an expected phenomenon reflecting the trade-off between the cell number and the per-cell sequencing depth given a fixed total sequencing depth.

Fig. 8B corresponds to the synthetic datasets scReadSim generates with a fixed cell number but increasing sequencing depths $1.2M$ $(0.25\times)$, $2.5M$ $(0.5\times)$, $5.0M$ $(1\times$, the original sequencing depth), $10.1M$ $(2\times)$, and $20.1M$ $(4\times)$. Fig. 8B shows the trend of the increasing read coverage as the sequencing depth increases.

Note that although scReadSim allows users to specify the sequencing depth arbitrarily, users must consider the biological limits so that the specified sequencing depth is reasonable (e.g., the total number of mRNA transcripts is the upper bound on the number of UMIs).

Application 1: Benchmarking UMI deduplication tools

For UMI-based scRNA-seq data, UMI deduplication tools were developed to quantify gene expression levels from scRNA-seq reads, some of which may come from the same RNA molecule. UMI deduplication tools input scRNA-seq reads (containing UMIs and cell barcodes) and output a gene-by-cell UMI count matrix. scReadSim can be used for benchmarking UMI deduplication tools because it provides the ground-truth UMI count matrix in its synthetic scRNA-seq read generation process. Hence, we deploy scReadSim to the 10x single-cell Multiome dataset (the RNA-seq modality only, see "Availability of data and materials") and use the synthetic reads to benchmark popular UMI deduplication tools: cellranger [11], UMI-tools [10], and Alevin [12]. Since Alevin is a transcript-level deduplication tool but scReadSim generates gene-level scRNA-seq reads, we focus on cellranger and UMI-tools in the following result comparison. We only include Alevin's result in Figs. S7 and S8.

Our benchmark study shows that cellranger achieves better performance than UMI-tools in three aspects. First, cellranger is computationally more efficient than UMI-tools on all synthetic datasets (with varying cell numbers and sequencing depths) (Fig. 9A). As expected, cellranger

and UMI-tools both take longer to run when the sequencing depth increases, and both methods' running time is unaffected by the cell number when the sequencing depth is fixed.

Second, the UMI count matrix output by cellranger agrees better with the ground-truth UMI count matrix in terms of (1) Pearson correlation and Kendall's tau, both gene-wise (the correlation across cells per gene) and cell-wise (the correlation across genes per cell) (Fig. 9B); and (2) the distributions of six summary statistics, including four gene-level statistics (mean, variance, coefficient of variance, and zero proportion) and two cell-level statistics (zero proportion and cell library size) (Fig. 9C).

Third, UMAP visualization shows that, compared to UMI-tools, cellranger outputs a UMI count matrix that is more similar to the ground-truth UMI count matrix (Pearson correlation 0.984 vs. UMI-tools' 0.665) and better preserves the separation among cell clusters (Fig. 9D).

Application 2: Benchmarking scATAC-seq peak-calling tools

For benchmarking scATAC-seq peak-calling tools, scReadSim provides ground-truth peaks and non-peaks by learning from user-specified trustworthy peaks and non-peaks or calling trustworthy peaks and non-peaks from real data. Here we use scReadSim's synthetic scATAC-seq data (pseudo-bulk), which contain designed grouth-truth peaks and non-peaks (see Methods "Benchmark of peak-calling tools") and mimic the sci-ATAC-seq dataset (see "Availability of data and materials"), to benchmark four popular peak-calling tools: (1) MACS3, designed for identifying transcript factor binding sites from ChIP-seq data; (2) SEACR, designed for CUT&RUN data; (3) HOMER, designed for ChIP-seq data but also widely used for ATAC-seq data; and (4) HMMRATAC, designed for ATAC-seq data.

Our benchmark results indicate that MACS3 and HMMRATAC have the best overall performance among the four peak-calling tools. In particular, the peaks called by MACS3 best resemble the ground-truth peaks in terms of read coverage, while HMMRATAC achieves the best balance between precision and recall.

First, we plot the RPKM distributions in ground-truth peaks and non-peaks, as well as in the identified peaks and non-peaks by each peak-calling tool (Fig. 10A). The distributions indicate that MACS3 identifies the peaks that are most similar to the ground-truth peaks, followed by HMMRATAC, HOMER, and SEACR. Specifically, the peaks called by HMMRATAC and SEACR are much longer than those by MACS and HOMER: the average peak lengths are 1543.0 bp and 1746.9 bp for HMMRATAC and SEACR, respectively, in contrast to 490.4 bp and 431.6 bp for MACS3 and HOMER, respectively. Possibly due to the design for CUT&RUN data, SEACR identifies the genomic regions with non-zero read coverages as peaks. However, the sci-ATAC-seq pseudo-bulk data used here contain many trustworthy non-peaks with non-zero read coverages, so most of the peaks called by SEACR contain so few reads that these peaks' RPKMs are close to 0.

Second, all four tools detect most of the ground-truth peaks, under the assumption that a ground-truth peak is considered as correctly identified if a minimum length of the peak (i.e., half of the length of the shortest peak) is overlapped by an identified peak (Figs. 10B Left, 10C–

D, and S9). Under the default settings of the four tools, the recall rates (i.e., the percentage of ground-truth peaks identified correctly) are all high: 86.8% for MACS3, 83.9% for SEACR, 92.4% for HOMER, and 84.4% for HMMRATAC.

Third, HMMRATAC achieves the highest precision rate of 90.3% (i.e., the percentage of identified peaks that are correct), followed by 76.5% for MACS3, 56.3% for HOMER, and 7.8% for SEACR. SEACR's low precision rate is possibly due to its design, leading to a vast number of identified peaks ($34{,}834$ peaks vs. $3{,}678$ peaks of MACS3, $5{,}314$ peaks of HOMER, and $3{,}029$ peaks of HMMRATAC) (Figs. 10B Right). Combining the precision and recall rates, the F1 scores indicate that HMMRATAC (87.3%) outperforms MACS3 (81.3%), HOMER (70.0%), and SEACR (14.3%).

Discussion

In this article, we propose scReadSim, a single-cell RNA-seq and single-cell ATAC-seq read simulator that generates realistic synthetic data by mimicking real data. Trained on real data, scReadSim generates synthetic sequencing reads in BAM or FASTQ formats. Particularly, scReadSim has three characteristics.

1. As a single-cell omics simulator, scReadSim mimics both scRNA-seq and scATAC-seq data. The synthetic data generated by scReadSim contains ground truths, including UMI counts for scRNA-seq and open chromatin regions for scATAC-seq. In particular, scReadSim allows users to design ground-truth open chromatin regions for scATAC-seq data generation.

2. As a read simulator, scReadSim generates sequencing reads instead of a count matrix. The synthetic reads are output in FASTQ and BAM formats, allowing the benchmark of read-level bioinformatics tools.

3. As a versatile simulator, scReadSim allows the cell number and sequencing depth to vary.

Deploying scReadSim to a sci-ATAC-seq dataset [22] and a 10x single-cell Multiome dataset [23] (including both single-cell gene expression and chromatin accessibility data) (see "Availability of data and materials"), we verify that the synthetic data generated by scReadSim resembles the real data in terms of both read counts and read sequences (Figs. 4–6 and Supplementary Figs. S1–S5). To confirm the flexibility of scReadSim, we show that scReadSim enables users to design ground-truth peaks for generating synthetic scATAC-seq reads (Figs. 7 and S6) and vary the cell number and sequencing depth for synthetic scRNA-seq and scATAC-seq data (Fig. 8).

Among scReadSim's versatile uses, we use two exemplar applications to demonstrate scReadSim's utility as a benchmarking tool. First, scReadSim provides a ground-truth UMI count matrix for benchmarking scRNA-seq UMI deduplication tools such as cellranger and UMI-tools. We find that cellranger is preferred over UMI-tools for greater accuracy and higher computational efficiency (Fig. 9). Second, scReadSim provides ground-truth peaks and non-peaks for generating scATAC-seq sequencing reads. We benchmark four peak-calling tools (MACS3, SEACR, HOMER, and

HMMRATAC) on the synthetic scATAC-seq reads generated by scReadSim (Figs. 10 and S9): the peaks called by MACS3 best resemble the ground-truth peaks in terms of the read coverage, and HMMRATAC achieves the best balance between precision and recall.

Although the current version of scReadSim uses scDesign2 [6] as the internal count simulator (assuming that cells are from discrete cell types), scReadSim is easily adaptive to the new count simulator scDesign3 [20] that allows cells to come from continuous trajectories and have spatial locations. Moreover, scReadSim can be used for evaluating the robustness of read-level bioinformatics tools because scReadSim allows the generation of sequencing reads with varying levels of noise injection (e.g., read positions and coverages).

As a future direction, we will generalize scReadSim to generate scRNA-seq short and long reads with ground-truth RNA isoform abundances, a functionality unavailable in existing scRNA-seq simulators.

Methods

The scReadSim method

The input of scReadSim is a BAM file storing the aligned sequencing reads from scRNA-seq or scATAC-seq. As an overview, scReadSim consists of three major steps: genome segmentation, synthetic count generation, and synthetic read generation.

The first step segments the genome so that each genomic region (segment) is defined as a feature. Specifically, for scRNA-seq, scReadSim requires users to input a gene annotation file to segregate the genome into genes and inter-genes; for scATAC-seq, scReadSim requires trustworthy peaks and non-peaks (if not user-specified (optional), scReadSim by default uses MACS3 with stringent criteria to call trustworthy peaks (q-value ≤ 0.01) and non-peaks (q-value > 0.1) from the input BAM file) and defines the complementary genomic regions as gray areas. Next, scReadSim summarizes scRNA-seq reads in the input BAM file into a gene-by-cell UMI count matrix and an inter-gene-by-cell UMI count matrix, or scReadSim summarizes scATAC-seq reads in the input BAM file into a peak-by-cell count matrix and a non-peak-by-cell count matrix. We use a general term “feature-by-cell count matrix” to refer to each count matrix in the following text.

In the second step—synthetic count generation, scReadSim trains the count simulator scDesign2 [6] (if the cells belong to distinct clusters; otherwise, scDesign3 [20] can be used if the cells follow continuous trajectories) on the feature-by-cell count matrices to generate the corresponding synthetic count matrices. For scATAC-seq, scReadSim converts gray areas into non-peaks before generating the corresponding synthetic count matrix. In summary, scReadSim generates two synthetic count matrices—gene-by-cell and inter-gene-by-cell matrices—for scRNA-seq; scReadSim generates three synthetic count matrices—peak-by-cell, non-peak-by-cell, and non-peak-by-cell (converted from gray areas) matrices—for scATAC-seq.

The third step generates synthetic reads based on the synthetic count matrices, the input BAM file, and the reference genome. The synthetic reads are output in FASTQ and BAM formats. Figs. 2–3 illustrate the scReadSim workflows for scRNA-seq and scATAC-seq, respectively.

scReadSim for scRNA-seq

The procedure below is for simulating data from UMI-based scRNA-seq technologies such as 10x Chromium [24] and Drop-seq [25]. For non-UMI-based scRNA-seq technologies such as Smart-seq2 [26], the simulation procedure follows the next subsection "scReadSim for scATAC-seq with user-designed open chromatin regions" with user-designed open chromatin regions replaced by genes.

1. Pre-processing real scRNA-seq data to obtain UMI-count matrices

   To pre-process real scRNA-seq data for training, scReadSim requires a BAM file (containing scRNA-seq reads in cells) and a gene annotation file (in GTF format). Based on the gene coordinates in the annotation file, scReadSim segregates the reference genome into two sets of features: genes and inter-genes. Then scReadSim counts the number of UMIs whose reads overlap with each gene in every cell to construct a gene-by-cell UMI-count matrix, and similarly, an inter-gene-by-cell UMI-count matrix. The two matrices contain the same cells in the same column order.

2. Synthetic UMI-count matrix generation

   To generate the gene- and inter-gene-by-cell UMI-count matrices for synthetic cells, scReadSim trains the count simulator scDesign2 [6] on the pre-processed gene- and inter-gene-by-cell UMI-count matrices from real data, respectively. Note that scReadSim chooses scDesign2 because scDesign2 can generate realistic synthetic counts by capturing gene-gene correlations [9]; however, scReadSim is flexible to accommodate any scRNA-seq count simulator.

   Specifically, scDesign2 involves two steps: (1) model fitting on the real data count matrix and (2) generation of a synthetic count matrix by sampling synthetic cells from the fitted model. Hence, scReadSim uses scDesign2 to generate synthetic gene-by-cell and inter-gene-by-cell count matrices from the real gene-by-cell and inter-gene-by-cell count matrices, respectively. Since scDesign2 requires cells to be in discrete cell types, scReadSim addresses this requirement by pre-clustering the cells using the gene-by-cell count matrix and Louvain clustering (in the Seurat R package [27]) if cells do not have pre-defined cell type labels. Note that scReadSim can also use the updated count simulator scDesign3 [20] to generate synthetic count matrices if cells are from continuous trajectories instead of discrete cell types.

   In summary, scReadSim generates the synthetic gene-by-cell and inter-gene-by-cell UMI-count matrices in this step.

3. Synthetic read generation

   To generate the scRNA-seq reads for synthetic cells, scReadSim uses:

   - the synthetic gene-by-cell and inter-gene-by-cell UMI-count matrices,
   - the coordinates of the genes and inter-genes,
   - the input BAM file,
   - the reference genome,
   - the user-specified read length (optional; default 90 nt).

   In UMI-based scRNA-seq data, every paired-end read has only one end (referred to as read 2) containing RNA sequence information, while the other end (referred to as read 1) contains cell barcode and UMI information [17, 28]. The generation of these two ends is described below (Fig. 2).

   (a) Generation of read 2 (RNA sequence)

   For simplicity, we use a "read" to refer to a read 2. For each gene or inter-gene i in each synthetic cell, scReadSim (1) extracts the corresponding UMI count c_u (i.e., the number of synthetic UMIs) from the synthetic feature-by-cell UMI-count matrices; (2) summarizes the read count of each UMI from the real reads that overlap with this feature and obtains the empirical distribution $\widehat{F}_i$ of the read count per UMI; (3) samples the number of synthetic reads c for each synthetic UMI according to the empirical distribution $\widehat{F}_i$; and (4) for each synthetic UMI with read count c,

   i. samples with replacement a real UMI from the real UMIs with read counts equal to c;
   ii. samples (with replacement) c real reads from those that are associated with the sampled real UMI;
   iii. generates c synthetic reads from the c real reads, the read length, and the reference genome;
   iv. assigns the c synthetic reads to the synthetic UMI.

   Specifically, in (4), scReadSim converts the c real reads' 5' positions in the reference genome into the c synthetic reads' 5' positions (after adding random shifts, detailed in the next paragraph); then scReadSim finds the c synthetic reads' 3' positions in the reference genome based on the read length (e.g., if a synthetic read has the 5' position x and the read length l, then its 3' position is $x+l-1$). Next, given every synthetic read's 5' and 3' positions, and strand information (preserved from real reads), scReadSim extracts the read sequence from the reference genome sequence.

   Regarding the random shifts mentioned in the above paragraph, scReadSim adds a random shift (sampled uniformly from -4 to 4 nt) to each real read's 5' position to obtain a synthetic read's 5' position. The purpose of doing so is to resolve duplicate

5' positions due to sampling with replacement. To make the synthetic reads more realistic, scReadSim also allows the introduction of substitution errors, as described in subsection "Substitution errors introduced into synthetic reads."

(b) Generation of read 1 (cell barcode and UMI)

Read 1 is a concatenated sequence string of a UMI and a cell barcode. Accordingly, scReadSim concatenates a randomly generated cell barcode and a UMI to form a synthetic read 1, which will then be paired with a synthetic read 2 generated above. Below we describe the generation of a cell barcode and a UMI for each synthetic read 1, whose length is specified as 26 nt (including a 16-nt cell barcode and a 10-nt UMI).

Specifically, for each synthetic cell, scReadSim generates a 16-nt cell barcode by randomly sampling A, C, G, and T with replacement for 16 times and assigns this cell barcode to all synthetic reads belonging to the synthetic cell. To generate a UMI for each gene or inter-gene in each synthetic cell, scReadSim (1) extracts the corresponding UMI count c_u (i.e., the number of UMIs) from the synthetic feature-by-cell UMI-count matrices; (2) generates c_u 10-nt UMIs, with each UMI created by randomly sampling A, C, G, and T with replacement for 10 times; and (3) assigns the c_u UMIs to the synthetic read 2 based on the read-UMI relationship generated above. Note that the lengths of cell barcodes and UMIs can be user-specified (default lengths are 16 and 10 nt, respectively).

In summary, scReadSim outputs paired-end synthetic reads in two FASTQ files for read 1 and read 2 separately. Regarding the Phred quality scores in the FASTQ files, the read 2 FASTQ file uses the quality score 37 for reference sequences and the quality score 24 for erroneous substitutions, while the read 1 FASTQ file uses the quality score 37 for all sequences. Optionally, scReadSim can output a BAM file, which contains the mapped reads from the two FASTQ files to the reference genome.

scReadSim for scATAC-seq

1. Pre-processing

To pre-process real scATAC-seq data for training, scReadSim requires a BAM file (containing scATAC-seq reads in cells) and users' trustworthy peaks and non-peaks in the input BAM file. Alternatively, if users do not specify trustworthy peaks and non-peaks, scReadSim first deploys the peak-calling tool MACS3 [13] to identify the trustworthy peaks using a stringent rule (by setting a q-value threshold 0.01) and the trustworthy non-peaks using a loose rule (by setting a q-value threshold 0.1; the non-peaks are complementary to the peaks called under this threshold).

Based on the trustworthy peaks and non-peaks, scReadSim defines "gray areas" as the genomic regions complementary to the peaks and non-peaks, and such gray areas represent

the chromatin regions that cannot be confidently classified as peaks or non-peaks. In summary, scReadSim segments the reference genome into three sets of features: peaks, non-peaks, and gray areas (Fig. 3). Then scReadSim counts the number of reads overlapping each peak in every cell to construct a peak-by-cell count matrix. scReadSim also generates a non-peak-by-cell count matrix similarly. The two matrices contain the same cells in the same column order.

2. Synthetic count matrix generation

   In scATAC-seq read generation, scReadSim provides ground-truth peaks and non-peaks for benchmarking purposes by learning from user-specified trustworthy peaks and non-peaks or calling trustworthy peaks and non-peaks from real data. To generate the peak-by-cell and non-peak-by-cell count matrices for synthetic cells, scReadSim trains the count simulator scDesign2 on the pre-processed peak-by-cell and non-peak-by-cell count matrices from real data. Next, scReadSim converts the gray areas into non-peaks (so that the peaks can be regarded as "ground-truth peaks") and constructs a synthetic count matrix based on the gray areas' lengths, trustworthy non-peaks' lengths, and the previous synthetic non-peak-by-cell count matrix.

   Specifically, scReadSim generates gray areas' synthetic read counts based on the trustworthy non-peaks' synthetic read counts so that the gray areas' synthetic read coverage mimics the non-peaks' synthetic read coverage. For each gray area,

   (a) scReadSim first finds the closest non-peak on the 5' of the gray area and extracts the non-peak's synthetic count vector (the non-peak's corresponding row) from the synthetic non-peak-by-cell count matrix;

   (b) scReadSim then randomly masks (sets to 0) the count entries of the non-peak's synthetic count vector with a probability equal to $\max\left(1 - \frac{\text{length of gray area}}{\text{length of non-peak}},\ 0\right)$ and assigns the masked count vector as the gray area's synthetic count vector.

   This random masking essentially scales the non-peak's synthetic read coverage according to the ratio of the gray area's length and the non-peak's length so that the gray area's synthetic read coverage mimics that of its nearby non-peak.

3. Synthetic read generation

   To generate the scATAC-seq reads for synthetic cells, scReadSim uses

   - the synthetic peak-by-cell, non-peak-by-cell count matrices,
   - the coordinates of the peaks and non-peaks,
   - the input BAM file,
   - the reference genome,
   - the user-specified read length (optional; default 50 nt).

As scATAC-seq usually generates paired-end reads, scReadSim accordingly generates synthetic paired-end reads. For clarity, we use a "read" to denote one end in a paired-end read. For each peak or non-peak in each synthetic cell, scReadSim (1) extracts the corresponding count c (i.e., the number of synthetic reads) from the synthetic feature-by-cell (feature refers to peak or non-peak) count matrices; (2) samples (with replacement) $\lceil c/2 \rceil$ (rounding $c/2$ to the least integer greater than or equal to $c/2$) real reads that overlap this feature in the BAM file; (3) finds the mates of (i.e., real reads in pairs with) the $\lceil c/2 \rceil$ sampled real reads from the BAM file; and (4) generates $\lceil c/2 \rceil$ synthetic read pairs from the $\lceil c/2 \rceil$ real read pairs, the read length, and the reference genome.

Specifically, in (4), scReadSim follows a similar procedure to the one in [subsection "scReadSim for scRNA-seq" $\rightarrow$ "3. Synthetic read generation" $\rightarrow$ "(a) Generation of read 2 (RNA sequence)" $\rightarrow$ "step (4) iii. generates c synthetic reads from the c real reads, the read length, and the reference genome"]. The only difference is the accommodation of paired-end reads. In the random shifting step to avoid duplicate reads, scReadSim adds the same random shift to a pair of reads' 5' positions to keep the fragment size unchanged after the shift.

Typically, a scATAC-seq BAM file stores cell barcodes as additional information. To generate a cell barcode (e.g., 16-nt long) for each synthetic cell, scReadSim randomly samples A, C, G, and T with replacement 16 times, and assigns this cell barcode to all synthetic reads belonging to the synthetic cell.

scReadSim for scATAC-seq with user-designed open chromatin regions

1. Pre-processing

   Optionally, scReadSim allows users to design ground-truth open chromatin regions and then generates synthetic scATAC-seq reads accordingly. When users take this option, scReadSim requires users to input the BAM file, the trustworthy peaks and non-peaks of the BAM file, and the user-designed ground-truth peaks (Fig. 11). Given the user-designed ground-truth peaks, scReadSim specifies the complementary regions to the ground-truth peaks as the ground-truth non-peaks. In summary, scReadSim defines two sets of peaks and non-peaks: (1) the set of "trustworthy peaks and non-peaks" based on the user-specified (or scReadSim-defined) trustworthy peaks and non-peaks and (2) the set of "ground-truth peaks and non-peaks" based on the user-designed ground-truth peaks.

   For the "trustworthy peaks and non-peaks" feature set, scReadSim counts the number of reads overlapping each trustworthy peak or non-peak in every cell to construct a trustworthy-peak-by-cell count matrix and a trustworthy-non-peak-by-cell count matrix. The two matrices contain the same cells in the same column order.

   For the "ground-truth peaks and non-peaks" feature set, since the specified ground-truth peaks may not correspond to real peaks in the BAM file, scReadSim constructs the ground-truth peaks' counts by mapping the trustworthy peaks to the "most similar" ground-truth

peaks in terms of region length. Below are the detailed construction steps.

(a) For every ground-truth peak, scReadSim defines a corresponding trustworthy peak as follows. First, scReadSim finds the ground-truth peak's 50 most similar trustworthy peaks (in the same chromosome) in terms of lengths (Fig. 12A). Second, scReadSim chooses the trustworthy peak with the largest ratio of read count by peak length.

(b) To construct the ground-truth-peak-by-cell count matrix, for every ground-truth peak, scReadSim adds the corresponding trustworthy peak's row in the trustworthy-peak-by-cell count matrix to the ground-truth-peak-by-cell count matrix (Fig. 12B). This constructed ground-truth-peak-by-cell count matrix has the same column order as the trustworthy-peak-by-cell count matrix.

The ground-truth-non-peak-by-cell matrix is constructed similarly, where every ground-truth non-peak is mapped to a trustworthy non-peak. Note that for each ground-truth non-peak, scReadSim first finds the 50 most similar trustworthy non-peaks in terms of lengths, and then it chooses the trustworthy non-peak with the smallest ratio of read count by peak length.

The ground-truth-peak-by-cell and ground-truth-non-peak-by-cell count matrices, as well as the maps from ground-truth features to trustworthy features, will be used in the next steps.

2. Synthetic count matrix generation

   Similar to the synthetic count matrix generation for scATAC-seq without user-designed ground-truth peaks, scReadSim trains scDesign2 on the ground-truth-peak-by-cell and ground-truth-non-peak-by-cell count matrices (constructed from step "1. Pre-processing") to generate the ground-truth-peak-by-cell and ground-truth-non-peak-by-cell count matrices for synthetic cells.

3. Synthetic read generation

   To generate synthetic scATAC-seq reads from the synthetic ground-truth-peak-by-cell and ground-truth-non-peak-by-cell count matrices, for each ground-truth feature (peak or non-peak) in every synthetic cell, scReadSim (1) extracts the corresponding count c (i.e., the number of synthetic reads) from the synthetic ground-truth-feature-by-cell count matrices; (2) samples (with replacement) $\lceil c/2 \rceil$ real reads in the BAM file that overlap the ground-truth feature's corresponding trustworthy feature; (3) obtain the mates of the $\lceil c/2 \rceil$ sampled real reads; and (4) generates $\lceil c/2 \rceil$ synthetic read pairs from the $\lceil c/2 \rceil$ real read pairs, the read length, the signed distance between the ground-truth feature and its corresponding real feature, and the reference genome.

   Specifically, in (4), scReadSim converts the c real reads' 5' positions in the reference genome to the c synthetic reads' 5' positions after two adjustments: first, shifting real reads from the trustworthy feature to the ground-truth feature based on the signed distance between the two features; second, adding a random shift to each real read pair's 5' positions. In

greater detail, in the first adjustment step, assuming that the signed distance between the trustworthy and ground-truth features is $+d$, if a real read has a 5' position x, then the synthetic read's 5' position is $x + d$. This adjustment ensures that the synthetic reads are located in the ground-truth feature as how the real reads are located in the real feature. The second adjustment (random shift addition) is to avoid duplicate reads due to sampling with replacement. Specifically, scReadSim adds the same random shift to a pair of reads' 5' positions to keep the fragment size unchanged after the shift.

Output in FASTQ or BAM formats

To output the synthetic reads (excluding the UMI-based scRNA-seq reads 1, which contain cell barcodes and UMIs) in a FASTQ or BAM file, scReadSim first records the coordinates of the synthetic reads in a BED file. Then scReadSim uses the bedtools [29] `getfasta` function to extract every synthetic read's sequence from the reference genome and store the synthetic reads' sequences in a FASTA file, which is then converted to a FASTQ file by the seqtk software [30]. For paired-end reads, scReadSim would generate two FASTQ files: one for reads 1 and the other for reads 2. Users may choose FASTQ as the output format of scReadSim, or they may additionally output a BAM file by aligning the synthetic reads to the reference genome by the alignment software bowtie2 [31].

Substitution errors introduced into synthetic reads

To mimic the sequencing error rates observed in the actual Illumina sequencing reads, scReadSim introduces substitution errors into the synthetic reads. Specifically, scReadSim first uses the software fgbio [32] to calculate every position i's average error rate (i.e., the probability that the position is sequenced as a wrong nucleotide, denoted by p_i) and three substitution error rates (i.e., the probabilities that the position is wrongly sequenced as the three other nucleotides; the sum of the three probabilities is p_i) based on all real reads mapped to that position. Then for each synthetic read, scReadSim decides if every base call, say at position i, is erroneous by sampling from Bernoulli (p_i). If the base call is decided to be erroneous (i.e., the Bernoulli sample is 1), a substitution nucleotide is randomly sampled from the other three nucleotides with probabilities proportional to the three substitution error rates at position i. Specifically, for paired-end sequencing technologies like scATAC-seq, scReadSim introduces substitution errors separately for the two mates in a pair.

Data analysis

Count-level comparison between synthetic and real data

To verify that scReadSim generates synthetic data that mimics the real data regarding the count level, we compare synthetic and real gene-by-cell UMI count matrices through the following as-

pects: summary statistics, correlation among top-expressed genes (or peaks), and visualization of 2D embeddings. For scATAC-seq, in addition to these three aspects used in scRNA-seq, we compare the RPKM distributions between peak-by-cell read count matrices.

- **Summary statistics.** We utilize the following summary statistics to describe the marginal distributions of the count matrix entries: for every feature, we calculate the mean, variance, coefficient of variance, and zero proportion across cells; for every cell, we calculate the library size and zero proportion across features.
- **Feature-feature correlation.** To illustrate that scReadSim preserves the correlation among the genes (or peaks), we select 100 top-expressed features (genes for scRNA-seq; peaks for scATAC-seq) from the real count matrix and use a heatmap to show the correlation among these features for both real and synthetic count matrices.
- **Visualization of 2D embeddings.** We utilize principal component analysis (PCA) and uniform manifold approximation and projection (UMAP) [33, 34] to obtain the low-dimensional embedding of the combined real and synthetic count matrices: we first use PCA to obtain the 30 principal components and then deploy UMAP to obtain the 2D embeddings. We label the cells with the dataset sources (synthetic cells generated by scReadSim or real cells). We also report the median of the integration local inverse Simpson's index (miLISI) proposed by [35] to quantify the mixing levels of synthetic and real cells (see Subsection "Evaluation metrics"). The miLISI value (ranging from 1 to 2) will be close to 2 if a perfect mixture exists between the synthetic and real cells.
- **RPKM.** We use the quantity RPKM (see Subsection "Evaluation metrics") to summarize the read coverage over the whole reference genome. Specifically, we report the RPKM quantities separately for the peaks and non-peaks.

Sequence-level comparison between synthetic and real scRNA-seq data

We deploy scReadSim onto the 10x single-cell Multiome dataset (the RNA-seq modality only, see "Availability of data and materials") by inputting the reference genome, the BAM file, and the gene annotation GTF file (Fig. 2). Lastly, scReadSim generates synthetic scRNA-seq reads in the BAM format.

To show that the scReadSim's synthetic scRNA-seq data mimics real data under the read sequence level, we compare the real and synthetic BAM files through the k-mer spectrum, error rate per base, and genome browser visualization.

- **K-mer spectrum.** K-mer spectrum measures the occurrences of substrings of length k within the read sequences. We use the tool Jellyfish [36] with the default settings to obtain the 11-mer, 21-mer, 31-mer, and 41-mer spectra in the synthetic and real BAM files.

- **Error rates per base.** We calculate and compare the substitution error rate for every base in the real and synthetic reads. The substitution error rate per base within reads are obtained using the software fgbio [32] with the function `ErrorRateByReadPosition` by setting the option `collapse` as false.

- **Genome browser visualization.** We use the IGV genome browser [37] to visualize the real and synthetic BAM files. This visualization depicts how the pooled single-cell reads align with the reference genome and illustrates the bulk-level similarity of read coverage between the two BAM files.

Sequence-level comparison between synthetic and real scATAC-seq data

We deploy scReadSim onto the sci-ATAC-seq and 10x single-cell Multiome dataset (the ATAC-seq modality only, see "Availability of data and materials") by inputting the reference genome, the BAM file, and trustworthy peaks and non-peaks (Fig. 3). The trustworthy peaks are identified using MACS3 from the input BAM file with a stringent rule (by setting q-value parameter as `-q` 0.01). To obtain the trustworthy non-peaks, we first deploy MACS3 onto the input BAM file with a less stringent rule (by setting q-value parameter as `-q` 0.1), then we take the inter-peaks as the trustworthy non-peaks. Lastly, scReadSim generates synthetic scATAC-seq reads in the BAM format.

To justify that scReadSim simulates realistic scATAC-seq data under the read sequence level, we compare the real and synthetic BAM files through the k-mer spectrum, error rate per base, genome browser visualization (see Subsection "Sequence-level comparison between synthetic and real scRNA-seq data"), and, additionally, the fragment size distribution. We further verify the realism of scATAC-seq data by deploying the peak caller MACS3 onto both real and synthetic BAM files and comparing the peak calling results.

- **Fragment-size distribution.** For paired-end sequencing reads, the fragment-size distribution measures the distribution of the distances between two ends of the read pairs. We use the function `bamPEFragmentSize` of the software deeptools [38] with the default settings to obtain the fragment distributions of real and synthetic scATAC-seq reads.

- **Peak calling comparison.** We deploy the peak-calling tool MACS3 with the same setting (by setting q-value parameter as `-q` 0.01) onto the real and synthetic BAM files and compare the peak calling results using a Venn diagram. In addition, to show the ROC (receiver operating characteristic) curve and Precision-Recall curve, we implement MACS3 onto synthetic reads with a less stringent criterion to obtain more synthetic peaks for comparison (by setting q-value parameter as `-q` 0.5). Treating the peaks called from real data as truth, each peak identified from synthetic data is determined to be a true peak if it overlaps with a true peak over more than half of the minimal true peaks' length. Specifically, we gradually change the thresholds (such as p-values of peaks output by MACS3) of peak-calling tools to vary the number of peaks identified by the tools.

Benchmark of UMI deduplication tools

We deploy scReadSim onto the 10x single-cell Multiome dataset (the RNA-seq modality only, see "Availability of data and materials") to generate the synthetic scRNA-seq data. Then we use the simulated synthetic scRNA-seq reads to benchmark the two most-popular gene-level UMI deduplication tools: UMI-tools and cellranger. By taking the synthetic gene-by-cell UMI-count matrix output by scReadSim as the ground truth, we deploy each method onto the synthetic scRNA-seq reads to obtain the deduplicated UMI count matrix. Note that these deduplication tools may identify a different number of cells; hence, we only preserve those cells that are identified in all methods. Before measuring the similarity between the generated UMI count matrices and the ground-truth UMI count matrix, we filter the genes that have zero expression level across all cells in at least one method. We compare the generated UMI count matrices with the ground-truth count matrix in the following three aspects:

1. First, we calculate the correlation marginally between the ground-truth UMI count matrix and deduplicated UMI count matrices, gene-wise and cell-wise. The association measurements include Pearson correlation and Kendall's tau.

2. Second, we compare the UMI count matrices through six summary statistics, including four gene-level statistics (mean, variance, coefficient of variance, and zero proportion) and two cell-level statistics (zero proportion and cell library size).

3. Third, we use PCA and UMAP for dimension reduction and visualization to examine whether the deduplicated UMI count matrices preserve the same single-cell pattern from the ground truth.

In addition, we measure the time complexity by deploying each method onto the synthetic data with varying cell numbers: 1219 ($0.25\times$), 2438 ($0.5\times$), 4877 ($1\times$, the original cell number), 9754 ($2\times$), 19508 ($4\times$); and varying sequencing depth $1.2M$ ($0.25\times$), $2.5M$ ($0.5\times$), $5.0M$ ($1\times$, the original sequencing depth), $10.1M$ ($2\times$), $20.1M$ ($4\times$). The analysis is implemented on a server equipped with 256x Intel® Xeon Phi™ CPU 7210 at 1.30GHz.

Benchmark of peak-calling tools

To generate synthetic scATAC-seq reads with specified ground-truth open chromatin regions, we input scReadSim with a sci-ATAC-seq BAM file, lists of trustworthy peaks and non-peaks identified using peak-calling tool MACS3, and a list of arbitrarily designed ground-truth peaks (Fig. 11). Specifically, the trustworthy peaks and non-peaks are determined using MACS3 from the real BAM file (see Subsection "Sequence-level comparison between synthetic and real scATAC-seq data"). For the ground-truth peak list, we truncate each gene transcript region from its transcription starting site (TSS) with a random length uniformly sampled from $[250, 550]$ base pairs. Then these truncated gene transcript regions are further combined into longer regions if they intersect with

each other before input into scReadSim. Lastly, scReadSim generates a synthetic scATAC-seq dataset with these designed ground-truth peaks in a BAM format.

To benchmark the most popular peak-calling tools: MACS3, HOMER, SEACR, and HMM-RATAC, we deploy these four peak-calling tools with the default settings onto the synthetic scATAC-seq BAM file with the specified ground-truth peaks. Then we compare the peak-calling results with the ground-truth peaks, throughout the following three aspects:

- First, we calculate the synthetic reads' RPKM quantities for the ground-truth peaks and the peaks identified by the four peak-calling tools. The RPKM quantities are also reported for the non-peaks.
- Second, treating the ground-truth peaks as truth, we plot the ROC and precision-recall curves by changing the number of discoveries for each peak-calling tool. A peak identified by the peak-calling tool is determined to be a true peak if it overlaps with one ground-truth peak over half of the minimal ground-truth peak length.
- Third, we directly compare these peak calling results through the Venn diagram and upset plot using the Python package Intervene [39].

Evaluation metrics

Reads Per Kilobase Million (RPKM). The RPKM value of a feature is defined as

$$\text{RPKM} = \frac{\text{count per feature} \times 1{,}000{,}000}{\text{total count of reads} \times \text{feature length in kilobase}}.$$

miLISI. The integration local inverse Simpson's Index (iLISI) [35] for each cell measures the effective number of datasets in a neighborhood of the cell. A cell surrounded by neighboring cells from a single dataset has an iLISI value equal to 1; otherwise, if the cell has a neighborhood with an equal number of cells from two datasets, the iLISI value would be 2. Hence, the median iLISI (miLISI) across all cells measures the median mixing level of the two datasets, and a miLISI close to 2 means that the two datasets have a perfect mixing.

Availability of data and materials

Datasets

- **10x Genomics**: The 10x Genomics single-cell multiome dataset includes ATAC and gene expression data from the embryonic mouse brain tissue [23]. The raw BAM files are downloaded from `https://www.10xgenomics.com/resources/datasets/fresh-embryonic-e-18-mouse-brain-5-k-1-standard-2-0-0`.

- **sci-ATAC-seq**: The sci-ATAC-seq dataset measures the chromatin accessibility in 17 samples spanning 13 tissues in 8-week old mice [22]. The raw BAM file is downloaded from `https://atlas.gs.washington.edu/mouse-atac/`. Data from the tissue "Bone Marrow 62016" are used for analysis.

Software and code

The `scReadSim` Python package is available at `https://github.com/JSB-UCLA/scReadSim`. To run scReadSim, the following dependent software packages are needed.

- **MACS3**: Version 3.0.0a7 (`https://github.com/macs3-project/MACS`).
- **samtools**: Version 1.12 (with htslib 1.12) (`http://www.htslib.org/`).
- **bedtools**: Version 2.29.1 (`https://bedtools.readthedocs.io/en/latest/`).
- **seqtk**: Version 1.3-r117-dirty (`https://github.com/lh3/seqtk`).
- **bowtie2**: Version 2.3.4.1 (`http://bowtie-bio.sourceforge.net/bowtie2/index.shtml`).
- **Seurat**: R package Seurat Version 4.0.6.
- **fgbio**: Version 2.0.1 (`http://fulcrumgenomics.github.io/fgbio/`).

In addition, the following software packages are used for the analyses in this work.

Dimension reduction methods

- **UMAP**: Function `umap` from R package umap Version 0.2.7.0.
- **PCA**: Function `irlba` from R package irlba 2.3.5.

UMI deduplication tools

- **UMI-tools**: Version 1.1.2 (`https://github.com/CGATOxford/UMI-tools`).
- **Alevin**: Software Alevin (integrated into the software salmon Version 1.8.0) (`https://salmon.readthedocs.io/en/latest/alevin.html`).
- **cellranger**: Version 7.0.0 (`https://support.10xgenomics.com/single-cell-gene-expression/software/pipelines/latest/what-is-cell-ranger`).

Peak-calling tools

- **MACS3**: Version 3.0.0a7 (https://github.com/macs3-project/MACS).
- **SEACR**: Version 1.3 (https://github.com/FredHutch/SEACR).
- **HOMER**: Software HOMER Version 4.11 (http://homer.ucsd.edu/homer/index.html).
- **HMMRATAC**: Version 1.2.10 (https://github.com/LiuLabUB/HMMRATAC).

Other tools

- **Jellyfish**: Version 2.3.0 (https://github.com/gmarcais/Jellyfish).
- **Intervene**: Python package Intervene Version 0.41.0 (https://intervene.readthedocs.io/en/latest/install.html).

Competing interests

The authors declare no competing interests.

Acknowledgements

The authors appreciate the comments and feedback from the members of the Junction of Statistics and Biology at UCLA (http://jsb.ucla.edu).

Funding

This work was supported by the following grants: National Science Foundation DBI-1846216 and DMS-2113754, NIH/NIGMS R01GM120507 and R35GM140888, Johnson Johnson WiSTEM2D Award, Sloan Research Fellowship, UCLA David Geffen School of Medicine W.M. Keck Foundation Junior Faculty Award, and Chan-Zuckerberg Initiative Single-Cell Biology Data Insights [Silicon Valley Community Foundation Grant Number: 2022-249355] (to J.J.L.).

References

[1] Yukie Kashima, Yoshitaka Sakamoto, Keiya Kaneko, Masahide Seki, Yutaka Suzuki, and Ayako Suzuki. Single-cell sequencing techniques from individual to multiomics analyses. *Experimental & Molecular Medicine*, 52(9):1419–1427, 2020.

[2] Luke Zappia and Fabian J Theis. Over 1000 tools reveal trends in the single-cell rna-seq analysis landscape. *Genome biology*, 22(1):1–18, 2021.

[3] Luke Zappia, Belinda Phipson, and Alicia Oshlack. Splatter: simulation of single-cell rna sequencing data. *Genome biology*, 18(1):1–15, 2017.

[4] Davide Risso, Fanny Perraudeau, Svetlana Gribkova, Sandrine Dudoit, and Jean-Philippe Vert. A general and flexible method for signal extraction from single-cell rna-seq data. *Nature communications*, 9(1):1–17, 2018.

[5] Wei Vivian Li and Jingyi Jessica Li. A statistical simulator scdesign for rational scrna-seq experimental design. *Bioinformatics*, 35(14):i41–i50, 2019.

[6] Tianyi Sun, Dongyuan Song, Wei Vivian Li, and Jingyi Jessica Li. scDesign2: a transparent simulator that generates high-fidelity single-cell gene expression count data with gene correlations captured. *Genome biology*, 22(1):1–37, 2021.

[7] Jinjin Tian, Jiebiao Wang, and Kathryn Roeder. ESCO: Single cell expression simulation incorporating gene co-expression. *Bioinformatics*, 37(16):2374–2381, 2021.

[8] Zeinab Navidi, Lin Zhang, and Bo Wang. simATAC: a single-cell ATAC-seq simulation framework. *Genome biology*, 22(1):1–16, 2021.

[9] Helena L Crowell, Sarah X Morillo Leonardo, Charlotte Soneson, and Mark D Robinson. Built on sand: the shaky foundations of simulating single-cell rna sequencing data. *bioRxiv*, 2021.

[10] Tom Smith, Andreas Heger, and Ian Sudbery. UMI-tools: Modeling sequencing errors in Unique Molecular Identifiers to improve quantification accuracy. *Genome Research*, 27(3): 491–499, 2017.

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

[12] Avi Srivastava, Laraib Malik, Tom Smith, Ian Sudbery, and Rob Patro. Alevin efficiently estimates accurate gene abundances from dscrna-seq data. *Genome biology*, 20(1):1–16, 2019.

[13] Yong Zhang, Tao Liu, Clifford A Meyer, Jérôme Eeckhoute, David S Johnson, Bradley E Bernstein, Chad Nusbaum, Richard M Myers, Myles Brown, Wei Li, et al. Model-based analysis of chip-seq (macs). *Genome biology*, 9(9):1–9, 2008.

[14] Sven Heinz, Christopher Benner, Nathanael Spann, Eric Bertolino, Yin C Lin, Peter Laslo, Jason X Cheng, Cornelis Murre, Harinder Singh, and Christopher K Glass. Simple combinations of lineage-determining transcription factors prime cis-regulatory elements required for macrophage and b cell identities. *Molecular cell*, 38(4):576–589, 2010.

[15] Evan D Tarbell and Tao Liu. HMMRATAC: a Hidden Markov ModeleR for ATAC-seq. *Nucleic acids research*, 47(16):e91–e91, 2019.

[16] Michael P Meers, Dan Tenenbaum, and Steven Henikoff. Peak calling by sparse enrichment analysis for cut&run chromatin profiling. *Epigenetics & chromatin*, 12(1):1–11, 2019.

[17] Hirak Sarkar, Avi Srivastava, and Rob Patro. Minnow: A principled framework for rapid simulation of dscRNA-seq data at the read level. *Bioinformatics*, 35(14):i136–i144, 2019.

[18] Wenshan Li and Wei Wang. Read simulator for single cell RNA sequencing. *UCLA Electronic Theses and Dissertations*, 2019.

[19] Zhanlin Chen, Jing Zhang, Jason Liu, Zixuan Zhang, Jiangqi Zhu, Donghoon Lee, Min Xu, and Mark Gerstein. SCAN-ATAC-Sim: a scalable and efficient method for simulating single-cell ATAC-seq data from bulk-tissue experiments. *Bioinformatics*, 37(12):1756–1758, 2021.

[20] Dongyuan Song, Qingyang Wang, Guanao Yan, Tianyang Liu, and Jingyi Jessica Li. A unified framework of realistic in silico data generation and statistical model inference for single-cell and spatial omics. *bioRxiv*, pages 2022–09, 2022.

[21] Feng Yan, David R Powell, David J Curtis, and Nicholas C Wong. From reads to insight: a hitchhiker's guide to atac-seq data analysis. *Genome biology*, 21:1–16, 2020.

[22] Darren A Cusanovich, Andrew J Hill, Delasa Aghamirzaie, Riza M Daza, Hannah A Pliner, Joel B Berletch, Galina N Filippova, Xingfan Huang, Lena Christiansen, William S DeWitt, et al. A single-cell atlas of in vivo mammalian chromatin accessibility. *Cell*, 174(5):1309–1324, 2018.

[23] 10xGenomics. Fresh embryonic e18 mouse brain (5k), single cell multiome atac + gene expression dataset by cell ranger arc 2.0.0, 2019. `https://www.10xgenomics.com/resources/datasets/fresh-embryonic-e-18-mouse-brain-5-k-1-standard-2-0-0`.

[24] Grace X. Y. Zheng, Jessica M. Terry, Phillip Belgrader, Paul Ryvkin, Zachary W. Bent, Ryan Wilson, Solongo B. Ziraldo, Tobias D. Wheeler, Geoff P. McDermott, Junjie Zhu, Mark T. Gregory, Joe Shuga, Luz Montesclaros, Jason G. Underwood, Donald A. Masquelier, Stefanie Y. Nishimura, Michael Schnall-Levin, Paul W. Wyatt, Christopher M. Hindson, Rajiv

Bharadwaj, Alexander Wong, Kevin D. Ness, Lan W. Beppu, H. Joachim Deeg, Christopher McFarland, Keith R. Loeb, William J. Valente, Nolan G. Ericson, Emily A. Stevens, Jerald P. Radich, Tarjei S. Mikkelsen, Benjamin J. Hindson, and Jason H. Bielas. Massively parallel digital transcriptional profiling of single cells. *Nature Communications*, 8(1):14049, 2017.

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

[26] Simone Picelli, Omid R Faridani, Åsa K Björklund, Gösta Winberg, Sven Sagasser, and Rickard Sandberg. Full-length rna-seq from single cells using smart-seq2. *Nature protocols*, 9(1):171–181, 2014.

[27] Yuhan Hao, Stephanie Hao, Erica Andersen-Nissen, William M Mauck III, Shiwei Zheng, Andrew Butler, Maddie J Lee, Aaron J Wilk, Charlotte Darby, Michael Zager, et al. Integrated analysis of multimodal single-cell data. *Cell*, 184(13):3573–3587, 2021.

[28] Robert Salomon, Dominik Kaczorowski, Fatima Valdes-Mora, Robert E. Nordon, Adrian Neild, Nona Farbehi, Nenad Bartonicek, and David Gallego-Ortega. Droplet-based single cell RNAseq tools: A practical guide. *Lab on a Chip*, 19(10):1706–1727, 2019.

[29] Aaron R Quinlan and Ira M Hall. Bedtools: a flexible suite of utilities for comparing genomic features. *Bioinformatics*, 26(6):841–842, 2010.

[30] Wei Shen, Shuai Le, Yan Li, and Fuquan Hu. Seqkit: a cross-platform and ultrafast toolkit for fasta/q file manipulation. *PloS one*, 11(10):e0163962, 2016.

[31] Ben Langmead and Steven L Salzberg. Fast gapped-read alignment with bowtie 2. *Nature methods*, 9(4):357–359, 2012.

[32] Tim Fennell and Nils Homer. fgbio. `https://github.com/fulcrumgenomics/fgbio`, 2019.

[33] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[34] Etienne Becht, Leland McInnes, John Healy, Charles-Antoine Dutertre, Immanuel WH Kwok, Lai Guan Ng, Florent Ginhoux, and Evan W Newell. Dimensionality reduction for visualizing single-cell data using umap. *Nature biotechnology*, 37(1):38–44, 2019.

[35] Ilya Korsunsky, Nghia Millard, Jean Fan, Kamil Slowikowski, Fan Zhang, Kevin Wei, Yuriy Baglaenko, Michael Brenner, Po-ru Loh, and Soumya Raychaudhuri. Fast, sensitive and accurate integration of single-cell data with harmony. *Nature methods*, 16(12):1289–1296, 2019.

[36] G Marcais and C Kingsford. Jellyfish: A fast k-mer counter. *Tutorialis e Manuais*, 1:1–8, 2012.

[37] James T Robinson, Helga Thorvaldsdóttir, Wendy Winckler, Mitchell Guttman, Eric S Lander, Gad Getz, and Jill P Mesirov. Integrative genomics viewer. *Nature biotechnology*, 29(1): 24–26, 2011.

[38] Fidel Ramírez, Devon P Ryan, Björn Grüning, Vivek Bhardwaj, Fabian Kilpert, Andreas S Richter, Steffen Heyne, Friederike Dündar, and Thomas Manke. deeptools2: a next generation web server for deep-sequencing data analysis. *Nucleic acids research*, 44(W1): W160–W165, 2016.

[39] Tiziano Flati, Silvia Gioiosa, Giovanni Chillemi, Andrea Mele, Alberto Oliverio, Cecilia Manironi, Arianna Rinaldi, and Tiziana Castrignanò. A gene expression atlas for different kinds of stress in the mouse brain. *Scientific Data*, 7(1):1–18, 2020.

Figures

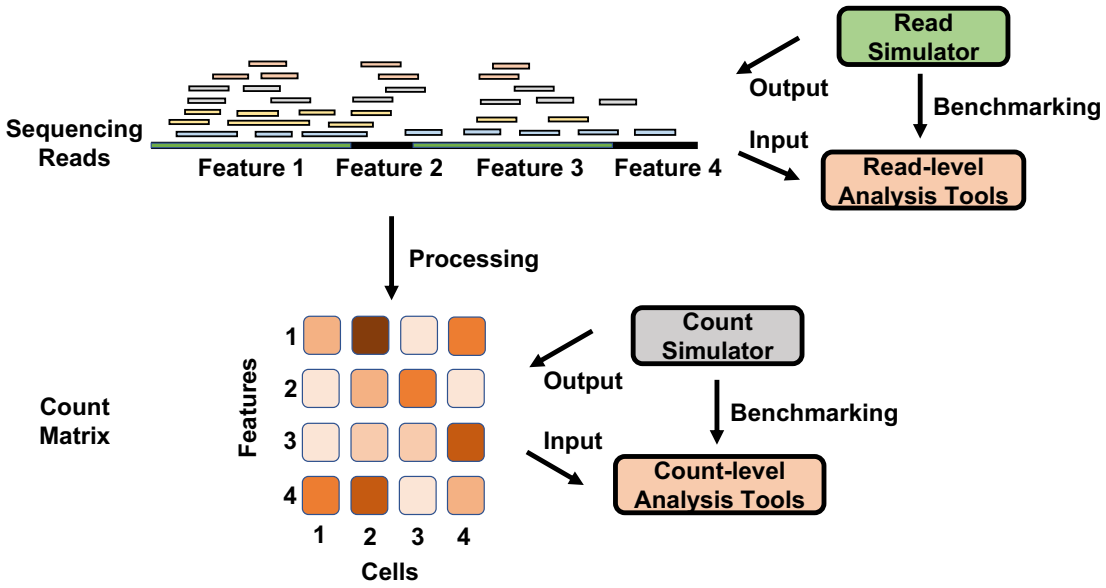

Figure 1: Relationships between single-cell sequencing data analysis tools and simulators. Read-level analysis tools process sequencing reads directly, while count-level analysis tools input a processed sequencing read count matrix. To benchmark count-level analysis tools, researchers need count simulators that generate synthetic sequencing read counts. However, count simulators cannot provide ground truths for benchmarking read-level analysis tools.

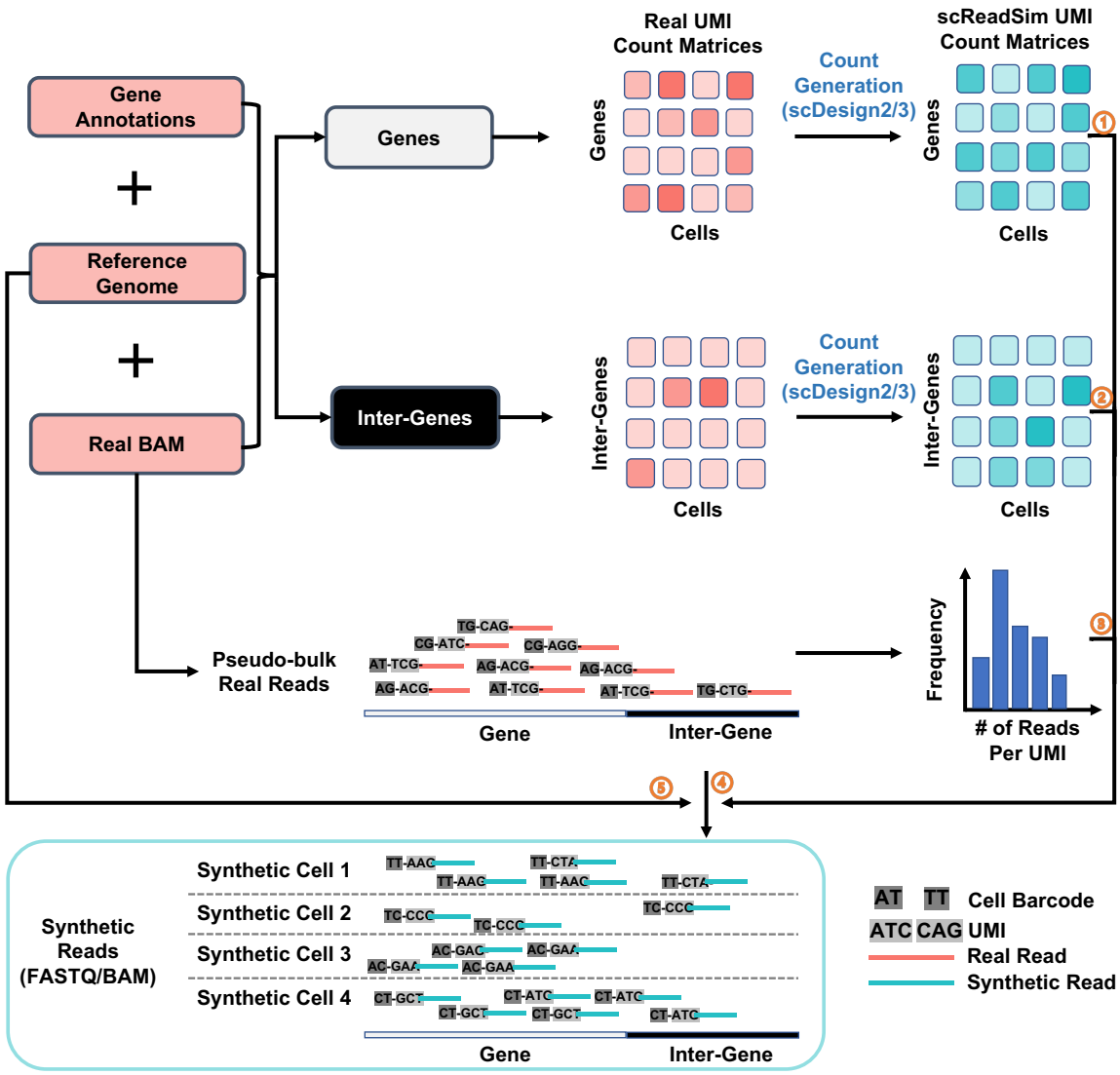

Figure 2: Workflow of scReadSim for generating scRNA-seq reads. scReadSim requires users to input a real BAM file, the corresponding reference genome, and a gene annotation GTF file. Based on the input, scReadSim segregates the reference genome into genes and inter-genes (i.e., intergenic regions). Based on the genes and inter-genes, scReadSim summarizes scRNA-seq reads in the input BAM file into a gene-by-cell UMI count matrix and an inter-gene-by-cell UMI count matrix. Then scReadSim trains the count simulator scDesign2 [6] (if the cells belong to distinct clusters; otherwise, scDesign3 [20] can be used if the cells follow continuous trajectories) on the two UMI count matrices to generate the corresponding synthetic UMI count matrices ("ground truths" for benchmarking UMI deduplication tools). Last, scReadSim generates synthetic scRNA-seq reads based on the synthetic UMI count matrices (① and ②), the input BAM file (③ and ④), and the reference genome (⑤). The synthetic reads are output in FASTQ and BAM formats.

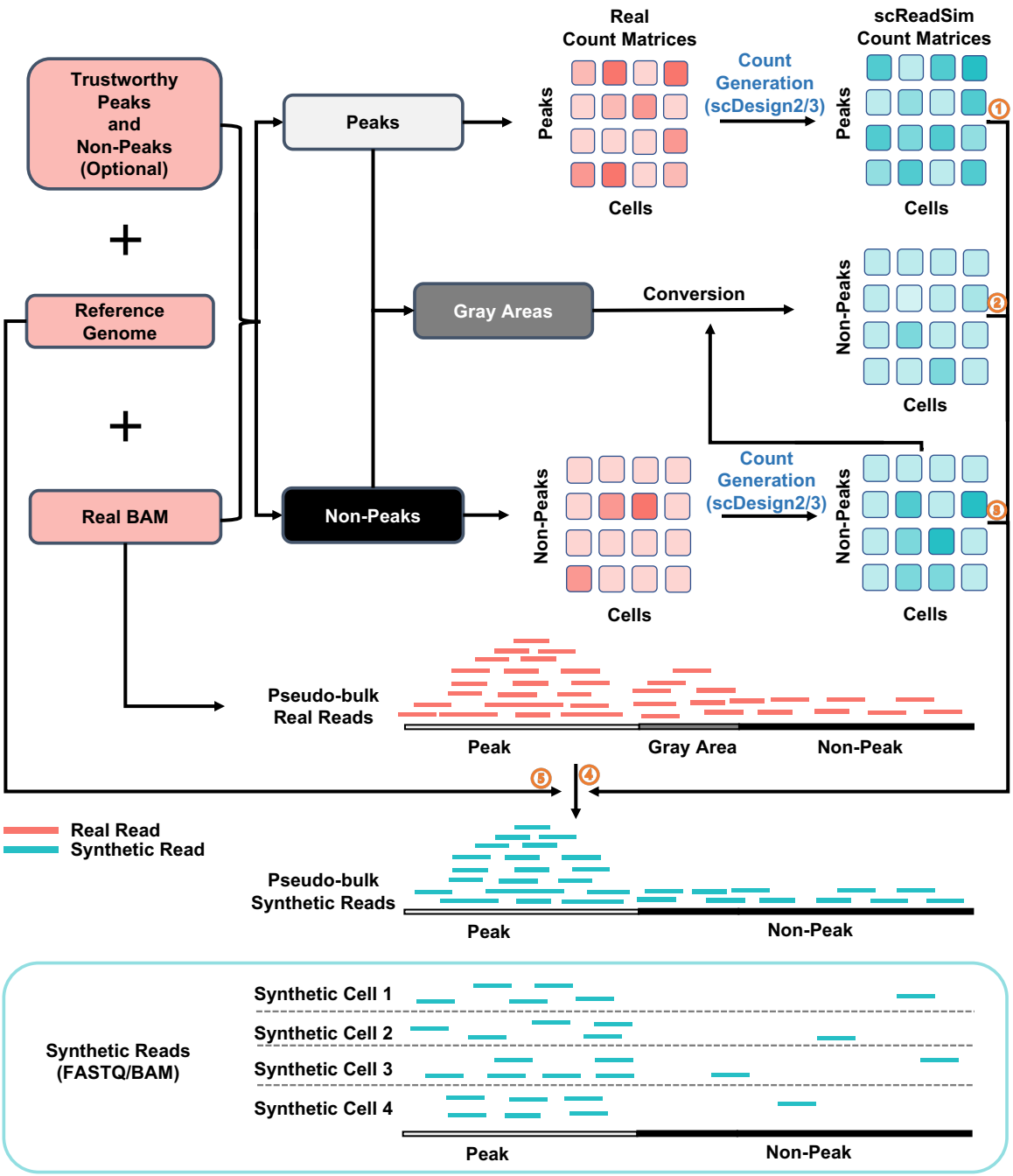

Figure 3: Workflow of scReadSim for generating scATAC-seq reads. The input includes a BAM file, the corresponding reference genome, and optionally, users' trustworthy peaks and non-peaks in the input BAM file; if users do not input trustworthy peaks and non-peaks, scReadSim by default uses MACS3 [13] with stringent criteria to call trustworthy peaks (called at q-value 0.01) and non-peaks (complementary to the peaks called at q-value 0.1) from the input BAM file. Then scReadSim defines gray areas as the genomic regions complementary to the peaks and non-peaks, and scReadSim summarizes scATAC-seq reads in the input BAM file into a peak-by-cell count matrix and a non-peak-by-cell count matrix. Next, scReadSim trains the count simulator scDesign2 [6] (if the cells belong to distinct clusters; otherwise, scDesign3 [20] can be used if the cells follow continuous trajectories) on the two count matrices to generate the corresponding synthetic count matrices for the peaks and non-peaks. Further, scReadSim converts the gray areas into non-peaks (so that the peaks can be regarded as "ground-truth peaks") and constructs a synthetic count matrix based on the gray areas' lengths and the already-generated synthetic non-peak-by-cell count matrix. Last, scReadSim generates synthetic reads based on the three synthetic count matrices (①, ②, and ③), the input BAM file (④), and the reference genome (⑤). The synthetic reads are output in FASTQ and BAM formats.

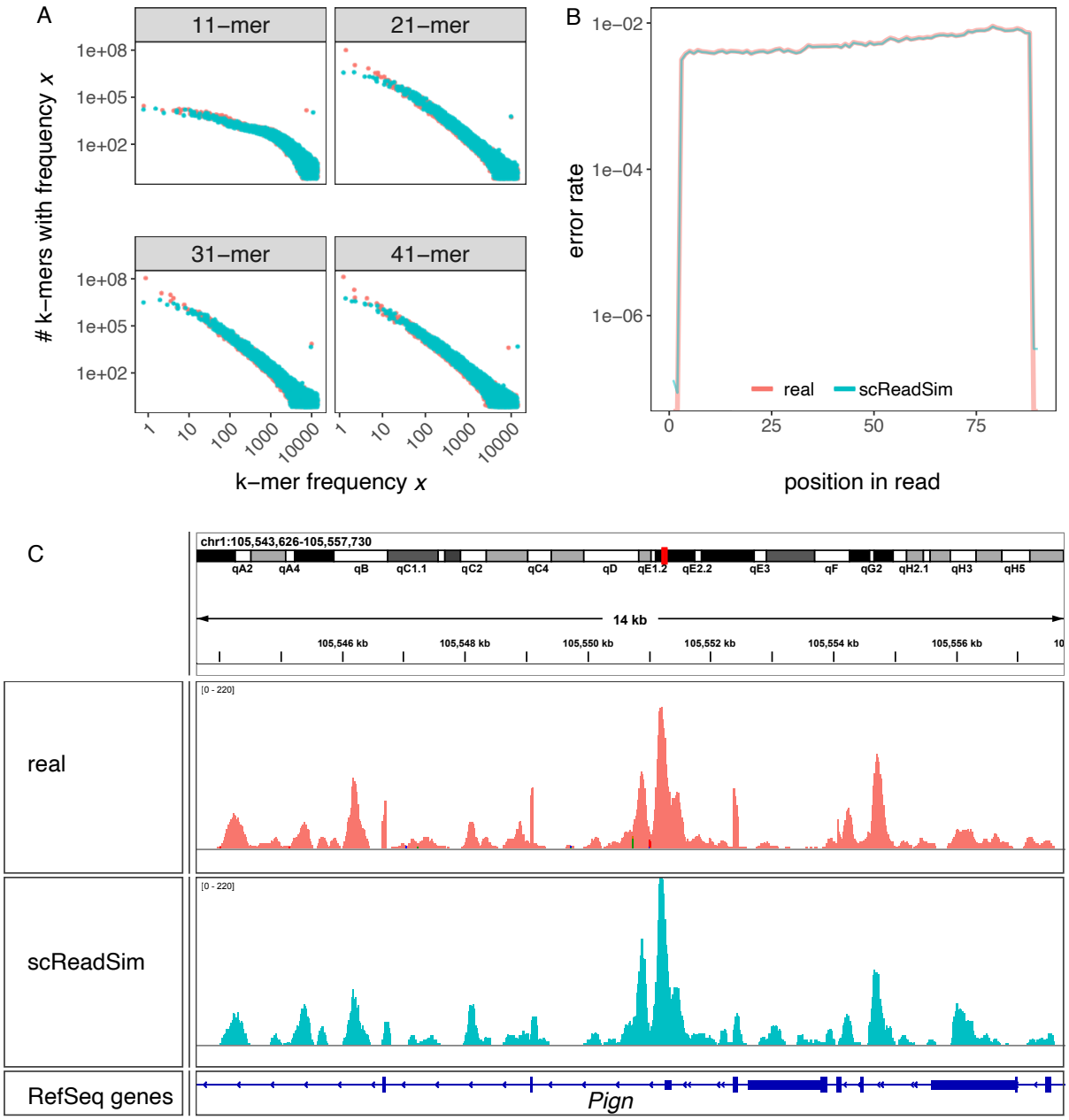

Figure 4: scReadSim's synthetic data resemble real 10x single-cell Multiome dataset (the RNA-seq modality only) [23] at the read sequence level. **A** Comparison of the k-mer spectra between scReadSim's synthetic data and real data. The x-axis refers to the occurrence frequency of a specific k-mer, and the y-axis represents the number of unique k-mers with this frequency. Both the x-axis and y-axis are in the $\log_{10}$ scale. **B** Comparison of the error rate per base call within a read between scReadSim's synthetic data and real data. The x-axis represents the positions of bases within a read, and the y-axis refers to the substitution error rate at each position. **C** Read coverage comparison of the real and synthetic BAM files in the IGV genome browser [37].

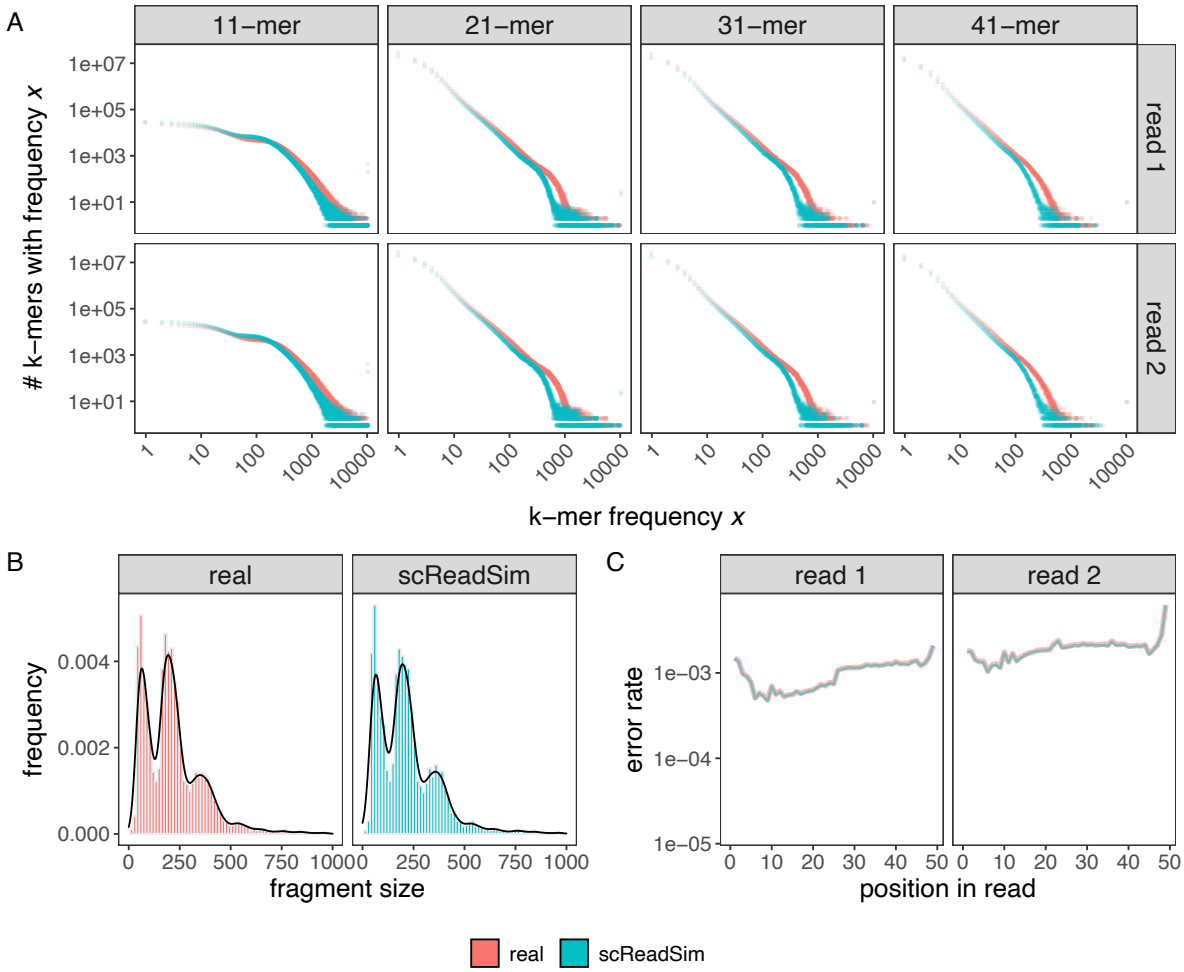

Figure 5: scReadSim's synthetic data resemble real 10x single-cell Multiome dataset (the ATAC-seq modality only) [23] at the read sequence level. **A** Comparison of the k-mer spectra between scReadSim's synthetic data and real data. The x-axis refers to the occurrence frequency of a specific k-mer, and the y-axis represents the number of unique k-mers with this frequency. Both the x-axis and y-axis are in the $\log_{10}$ scale. **B** Comparison of the fragment-size distributions between scReadSim's synthetic data and real data. **C** Comparison of the error rate per base call within a read between scReadSim's synthetic data and real data. The x-axis represents the positions of bases within a read, and the y-axis refers to the substitution error rate at each position.

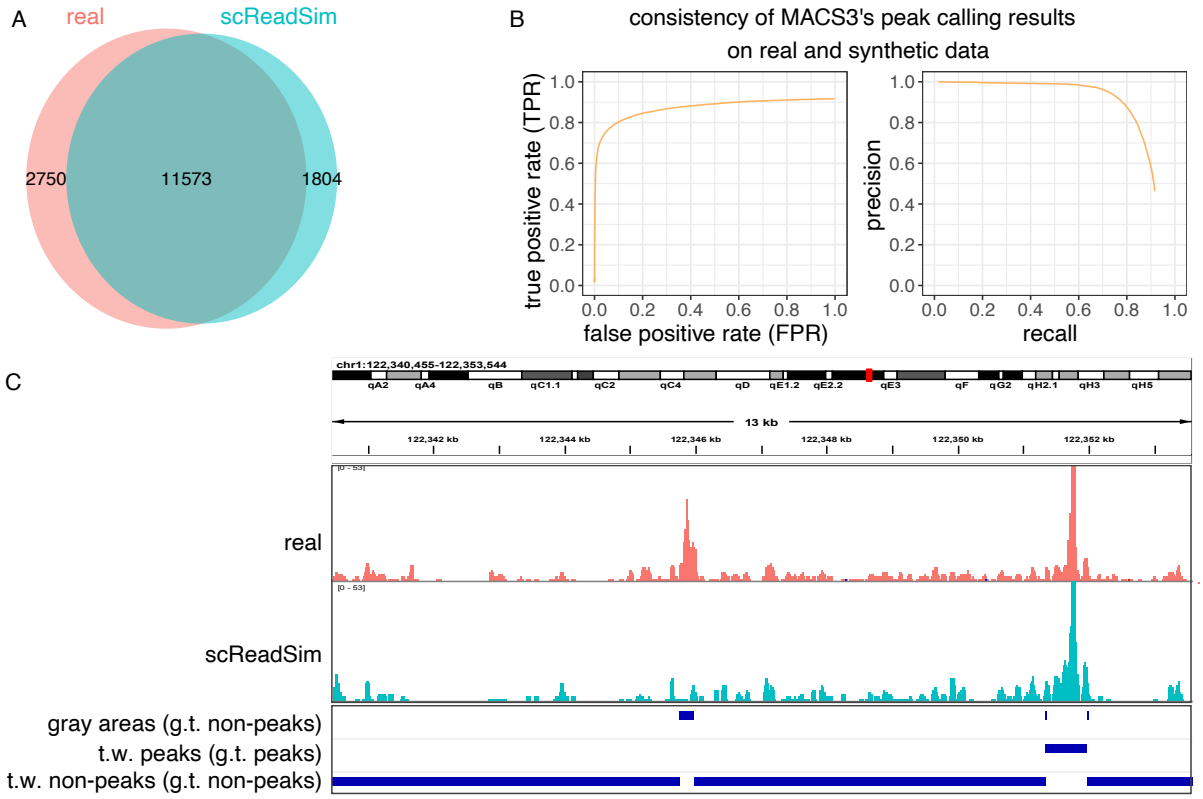

Figure 6: scReadSim's synthetic data resemble real 10x single-cell Multiome dataset (the ATAC-seq modality only) [23] in terms of open chromatin regions at the pseudo-bulk level. **A** A Venn diagram of MACS3's called peaks from scReadSim's synthetic data and real data. A peak called from real data (referred to as a "real peak") is considered to overlap with a peak called from synthetic data (referred to as a "synthetic peak") if the real peak has at least 101 bp (half of the shortest real peak's length) overlapped by the synthetic peak. **B** True positive rate (TPR) vs. false positive rate (FPR) (left) and precision vs. recall (right) for evaluating the synthetic peaks in (A) by treating the real peaks in (A) as truths. The TPR, equivalent to the recall, is the proportion of real peaks that have at least 101 bp overlapped by a synthetic peak; the FPR is the proportion of real non-peaks (the regions complementary to the real peaks) that have at least 101 bp overlapped by a synthetic peak; the precision is the proportion of synthetic peaks that overlap at least 101 bp of a real peak. **C** Read coverage comparison of the real and synthetic BAM files in the IGV genome browser [37]. Gray areas, trustworthy (t.w.) peaks, and trustworthy non-peaks are user-specified or scReadSim-identified from real data. scReadSim converts gray areas to ground-truth (g.t.) non-peaks, and it maintains trustworthy peaks and non-peaks as ground-truth peaks and non-peaks, respectively, in synthetic data.

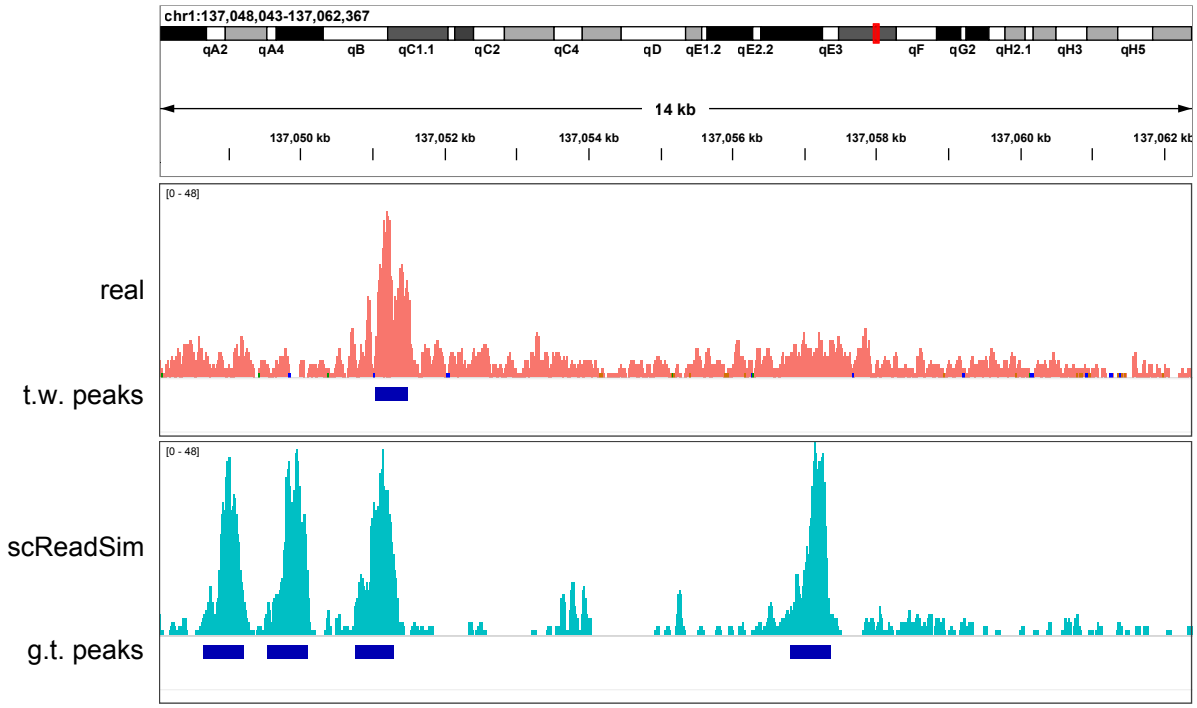

Figure 7: IGV genome browser [37] visualization of scReadSim's synthetic sci-ATAC-seq reads with user-designed ground-truth (g.t.) peaks (bottom track). From the real sci-ATAC-seq reads (top track) [22], the trustworhy (t.w.) peaks are called by MACS3 and used to train scReadSim.

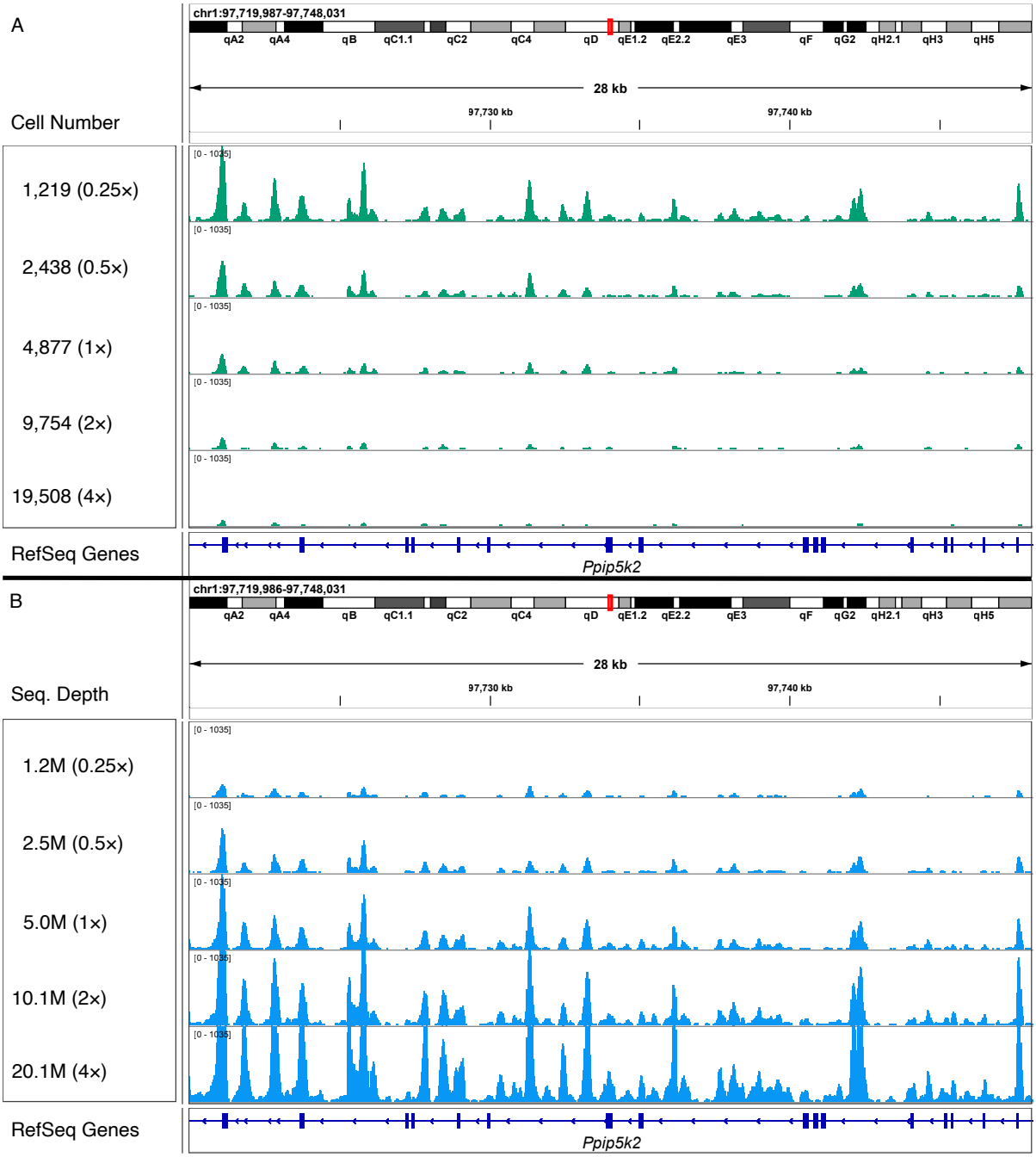

Figure 8: scReadSim allows varying cell number and sequencing depth as input parameters, with the corresponding synthetic data visualized in the IGV genome browser [37]. **A** Varying numbers of synthetic cells. The track names indicate the multiplication factors: '$1\times$' specifies the cell number in the real data; '$4\times$' specifies 4 times the real cell number. All five synthetic datasets are downsampled to $1{,}219$ cells for a fair comparison. **B** Varying sequencing depths in synthetic data. The track names indicate the multiplication factors: '$1\times$' specifies the sequencing depth in the real data; '$4\times$' specifies 4 times the real sequencing depth. For better visualization, the genome browser's track height is set to $1{,}035$ for all 10 tracks.

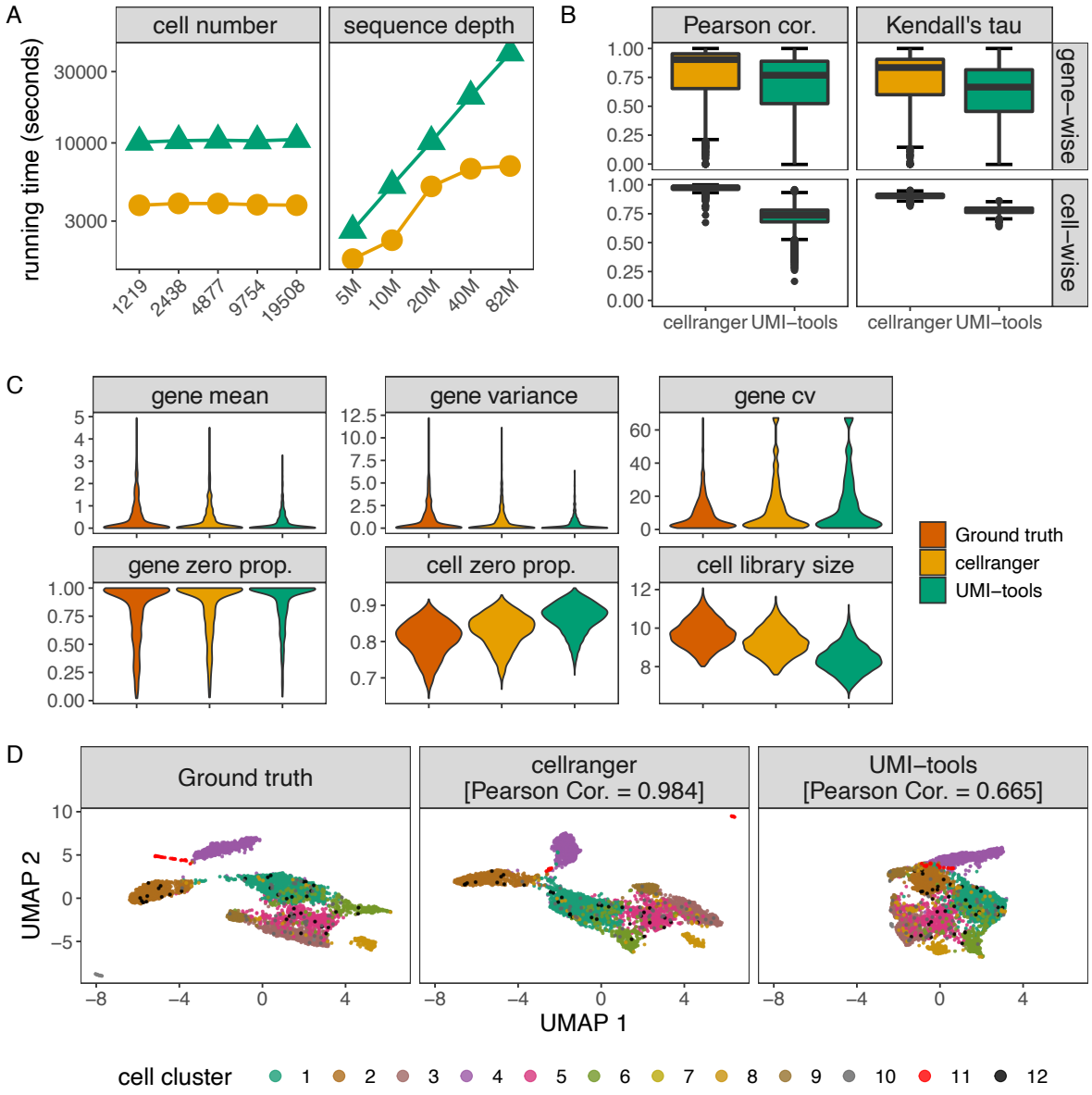

Figure 9: Benchmark of UMI deduplication tools using scReadSim's synthetic scRNA-seq reads. The ground-truth UMI count matrix is the gene-by-cell UMI count matrix generated by scReadSim. Two deduplication tools are considered: cellranger and UMI-tools. **A** Time usage of deduplication tools on synthetic datasets with varying cell numbers or sequencing depths. The y-axis indicates the time lapse (in seconds), and the x-axis reflects the number of synthetic cells or the total number of synthetic reads (sequencing depth). **B** Cell-wise and gene-wise correlations (Pearson correlation and Kendell's tau) between the ground-truth UMI count matrix and each deduplication tool's output UMI count matrix. **C** Summary statistics of the UMI count matrices at the gene level (mean, variance, coefficient of variance (cv), zero proportion) and the cell level (zero proportion and cell library size). **D** UMAP visualization of the UMI count matrices. Cells are labeled by the cell clusters used as the ground truths in scReadSim. The Pearson correlation ("Pearson Cor.") is calculated between the ground-truth UMI count matrix and each deduplication tool's output UMI count matrix (after expanding each matrix into a vector, the Perason correlation is calculated between the two vectors); the larger the correlation, the greater the similarity between the two matrices.

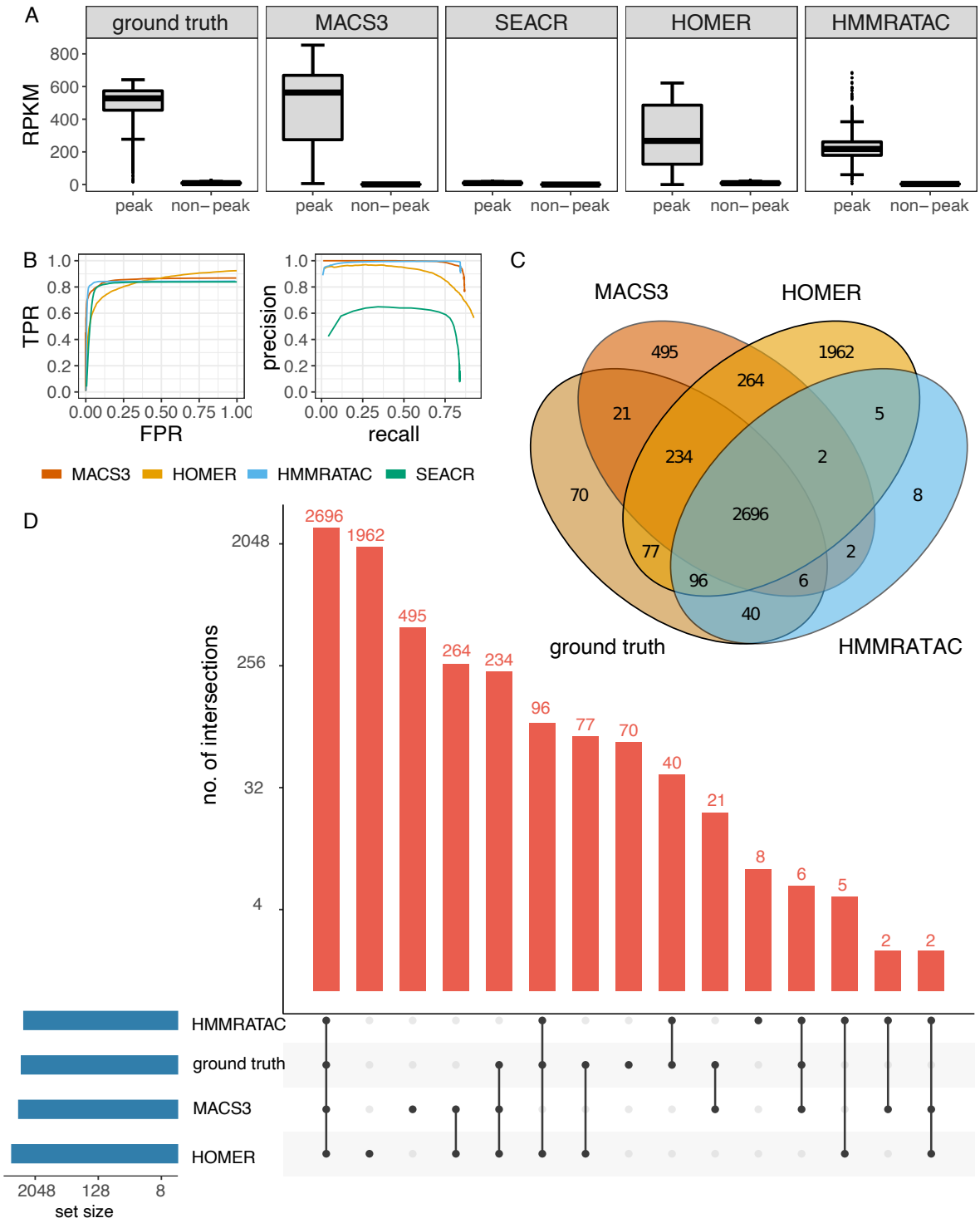

Figure 10: Benchmark of peak-calling tools (MACS3, SEACR, HOMER, and HMMRATAC) using scReadSim's synthetic scATAC-seq reads from designed ground-truth peaks and non-peaks (see Methods "Benchmark of peak-calling tools"). **A** RPKM value distributions in peak and non-peak regions in each set of peak calling results: for "User Input," peaks refer to the user-designed open chromatin regions; for each peak-calling tool, peaks are called by the method. **B** True positive rate (TPR) vs. false positive rate (FPR) curves (left) and precision vs. recall curves (right) using user-designed open chromatin regions as the truths. A called peak is considered true if it overlaps at least 225 bp of a user-designed open chromatin region, where 225 is half of the shortest open chromatin region. The Venn diagram (**C**) and upset plot (**D**) of the user-designed open chromatin regions and the called peaks by MACS3, HOMER, and HMMRATAC. See Fig. S9 for the comparison including SEACR.

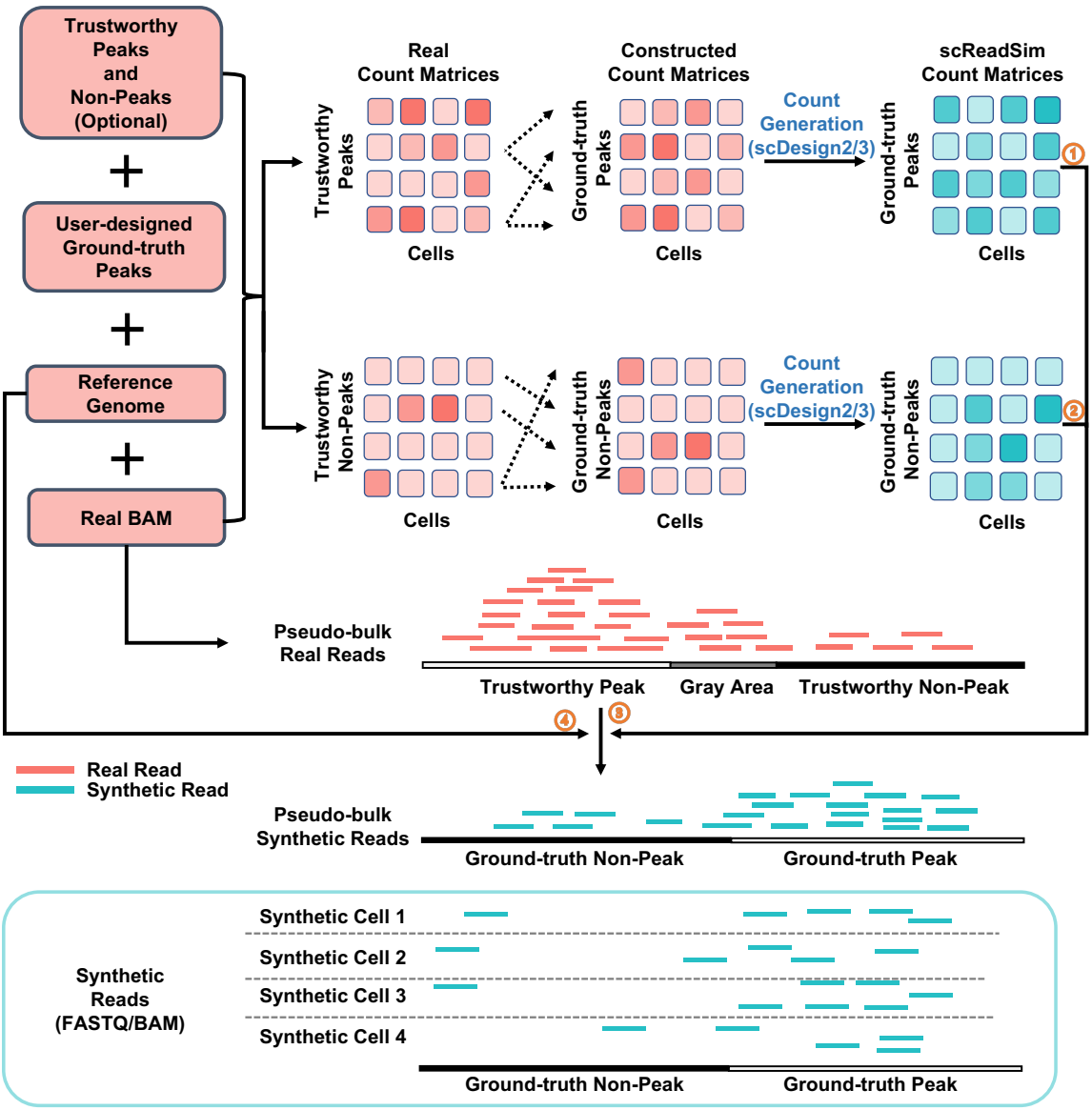

Figure 11: Workflow of scReadSim for generating scATAC-seq reads with user-designed open chromatin regions (i.e., ground-truth peaks). The input includes a BAM file, the corresponding reference genome, trustworthy peak and non-peak lists, and a ground-truth peak list that the synthetic scATAC-seq reads will be generated accordingly. Specifically, if users do not specify the trustworthy peaks and non-peaks, scReadSim by default uses MACS3 [13] with stringent criteria to call trustworthy peaks (q-value ≤ 0.01) and non-peaks (q-value > 0.1) from the input BAM file. Given ground-truth peaks, scReadSim defines the inter-peaks as the ground-truth non-peaks. Based on trustworthy peaks and non-peaks, scReadSim summarizes scATAC-seq reads in the input BAM file into a trustworthy-peak-by-cell count matrix and a trustworthy-non-peak-by-cell count matrix. Next, scReadSim defines a mapping between trustworthy and ground-truth peaks and constructs a ground-truth-peak-by-cell count matrix by extracting the corresponding trustworthy-peak-by-cell count matrix's entries. Similarly, scReadSim constructs a ground-truth-non-peak-by-cell count matrix from the non-trustworthy-peak-by-cell count matrix. Further, scReadSim trains the count simulator scDesign2 [6] (if the cells belong to distinct clusters; otherwise, scDesign3 [20] can be used if the cells follow continuous trajectories) on the ground-truth-peak- and ground-truth-non-peak-by-cell count matrices to generate the corresponding synthetic count matrices for the ground-truth peaks and non-peaks. Last, scReadSim generates synthetic reads based on the synthetic count matrices (① and ②), the input BAM file (③), and the reference genome (④). The synthetic reads are output in FASTQ and BAM format.

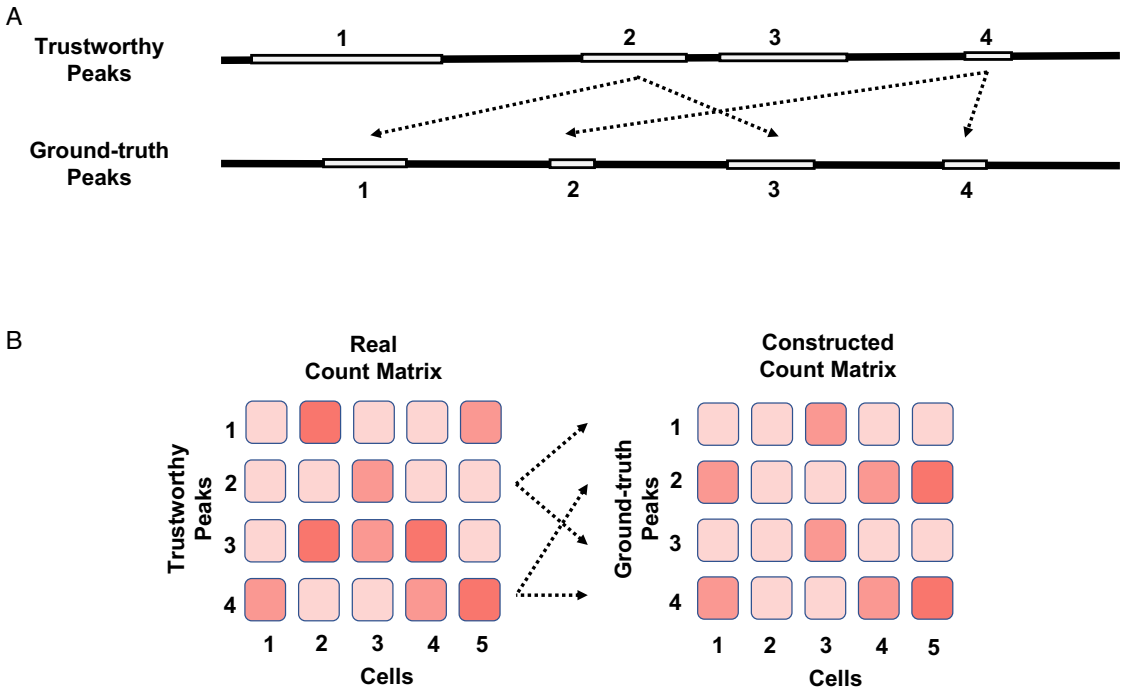

Figure 12: The mechanism of scReadSim's data generation with user-designed open chromatin regions (i.e., ground-truth peaks). **A** The mapping between trustworthy peaks and ground-truth peaks. For every ground-truth peak, scReadSim finds its most similar trustworthy peak (in the same chromosome) in terms of length. **B** The constructed count matrix for ground-truth peaks. For every ground-truth peak, scReadSim adds the corresponding trustworthy peak's row in the trustworthy-peak-by-cell count matrix to the ground-truth-peak-by-cell count matrix.

Supplementary Figures

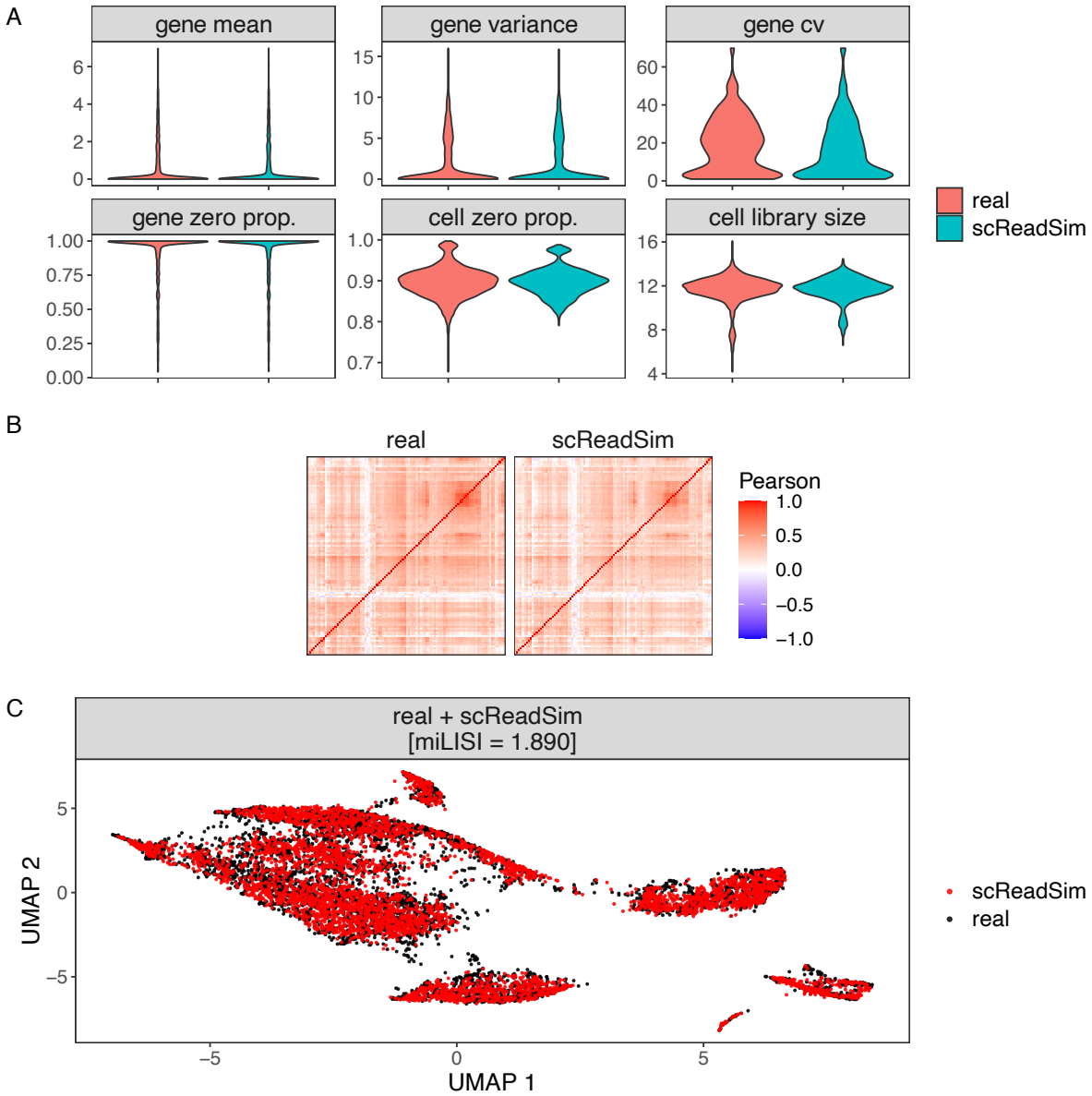

Figure S1: scReadSim's synthetic data resemble real 10x single-cell Multiome dataset (the RNA-seq modality only) [23] at the UMI count level. **A** Summary statistics of the UMI count matrix at the gene level (mean, variance, coefficient of variance (cv), and zero proportion) and the cell level (zero proportion and cell library size). **B** Correlations among the 100 top-expressed genes in the synthetic and real count matrices. The top-expressed genes are defined based on the real count matrix. **C** UMAP visualization of the pooled real and synthetic cells. miLISI measures the similarity of real and synthetic cells in the UMAP space: the miLISI value ranges between 1 and 2, with 2 indicating a perfect mixing of real and synthetic cells.

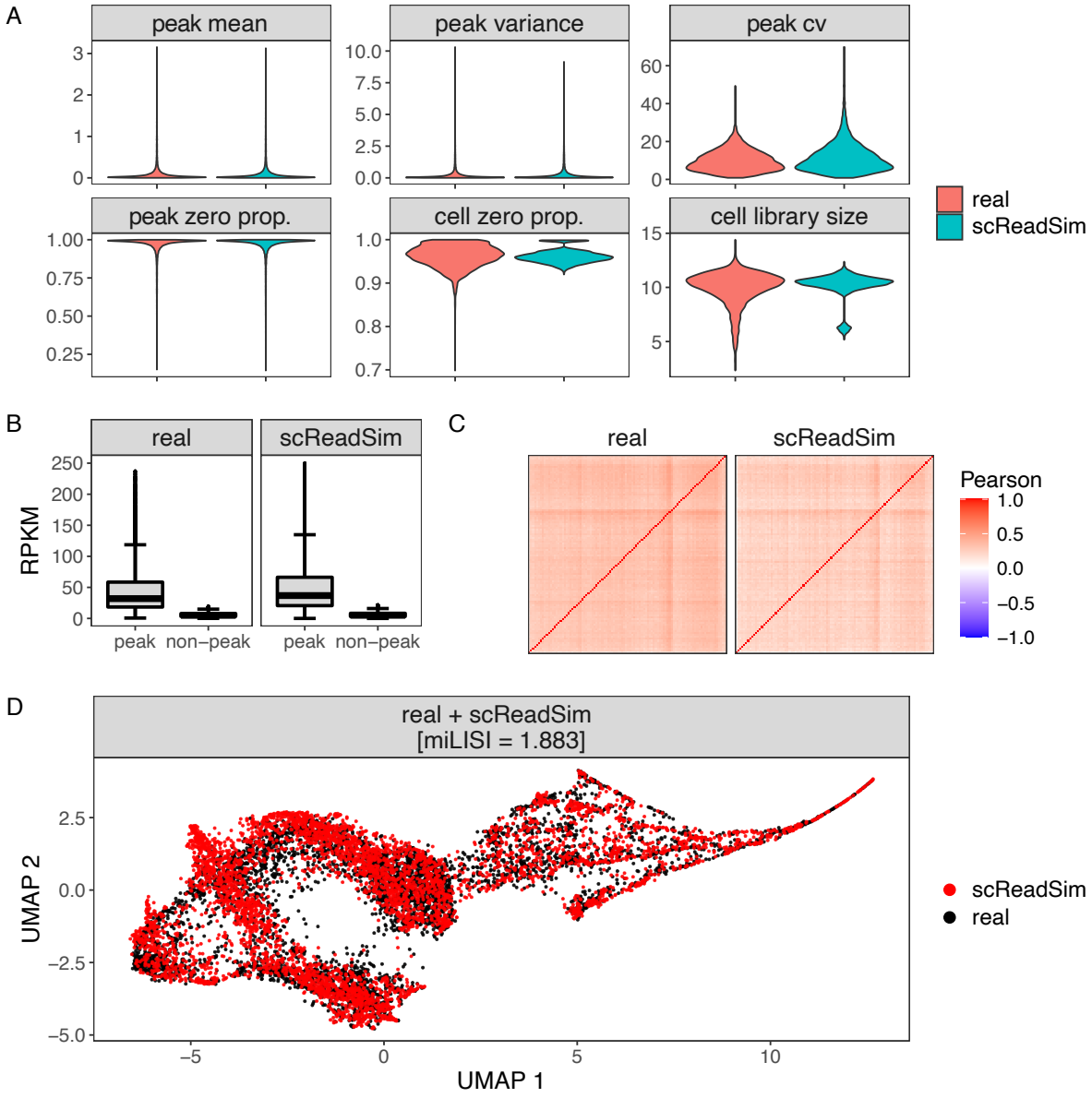

Figure S2: scReadSim's synthetic data resemble real 10x single-cell Multiome dataset (the ATAC-seq modality only) [23] at the count level. **A** Summary statistics of the count matrix at the peak level (mean, variance, coefficient of variance (cv), zero proportion) and the cell level (zero proportion and cell library size). **B** Comparison of the RPKM (Reads Per Kilobase Million) value distributions in peak and non-peak regions between scReadSim's synthetic data and real data. Peaks and non-peaks are obtained using MACS3 from the real BAM file, see Methods "Count-level comparison between synthetic and real data". **C** Correlation among 100 top peaks in the synthetic and real count matrices. The top-open peaks are defined based on the real count matrix. **D** UMAP visualization of the pooled real and synthetic cells. miLISI measures the similarity of real and synthetic cells in the UMAP space: the miLISI value ranges between 1 and 2, with 2 indicating a perfect mixing of real and synthetic cells.

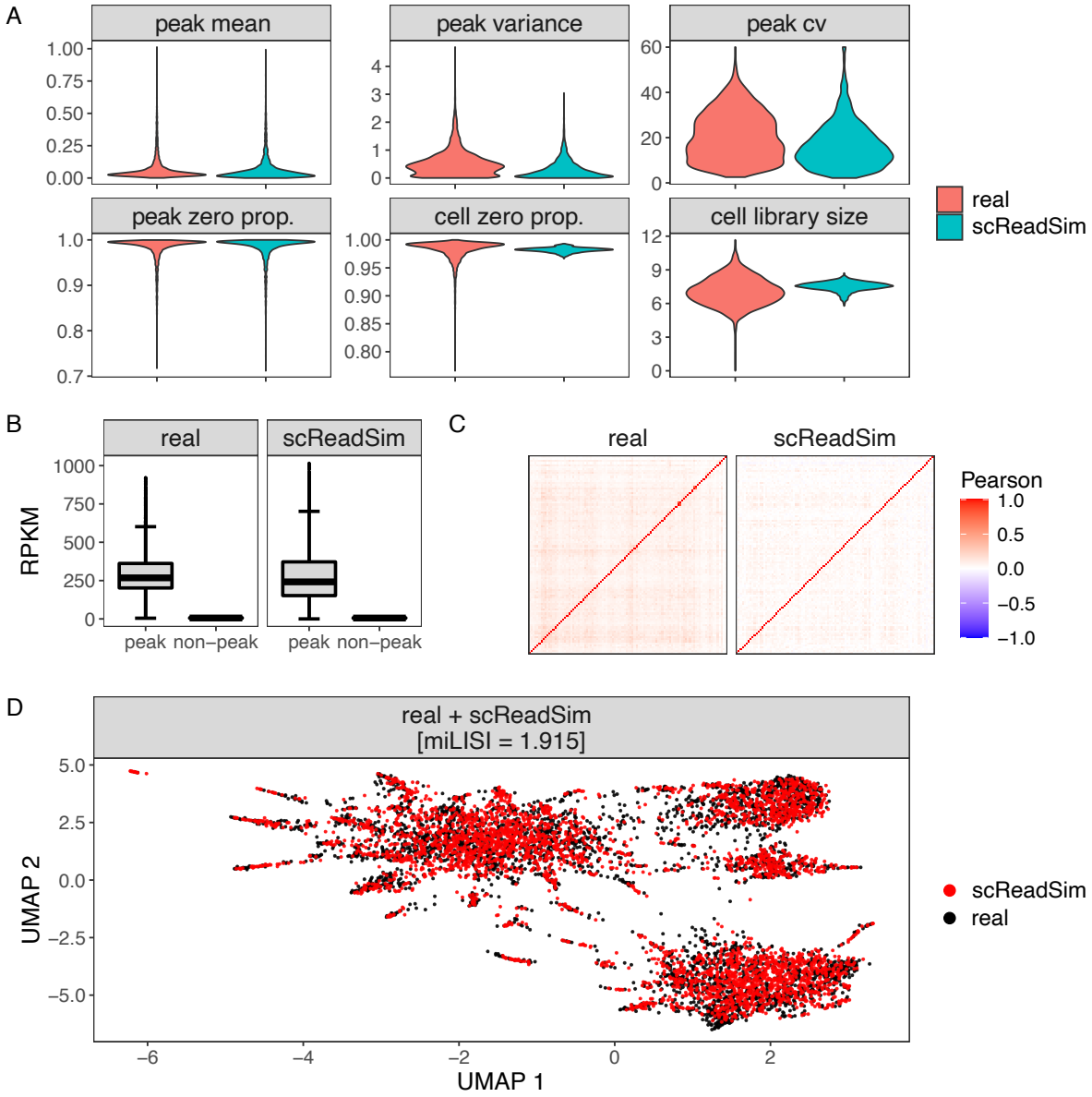

Figure S3: scReadSim's synthetic data resemble real sci-ATAC-seq data [22] at the count level. **A** Summary statistics of the count matrix at the peak level (mean, variance, coefficient of variance (cv), zero proportion) and the cell level (zero proportion and cell library size). **B** Comparison of the RPKM (Reads Per Kilobase Million) value distributions in peak and non-peak regions between scReadSim's synthetic data and real data. Peaks and non-peaks are obtained using MACS3 from the real BAM file, see Methods "Count-level comparison between synthetic and real data". **C** Correlation among 100 top peaks in the synthetic and real count matrices. The top-open peaks are defined based on the real count matrix. **D** UMAP visualization of the pooled real and synthetic cells. miLISI measures the similarity of real and synthetic cells in the UMAP space: the miLISI value ranges between 1 and 2, with 2 indicating a perfect mixing of real and synthetic cells.

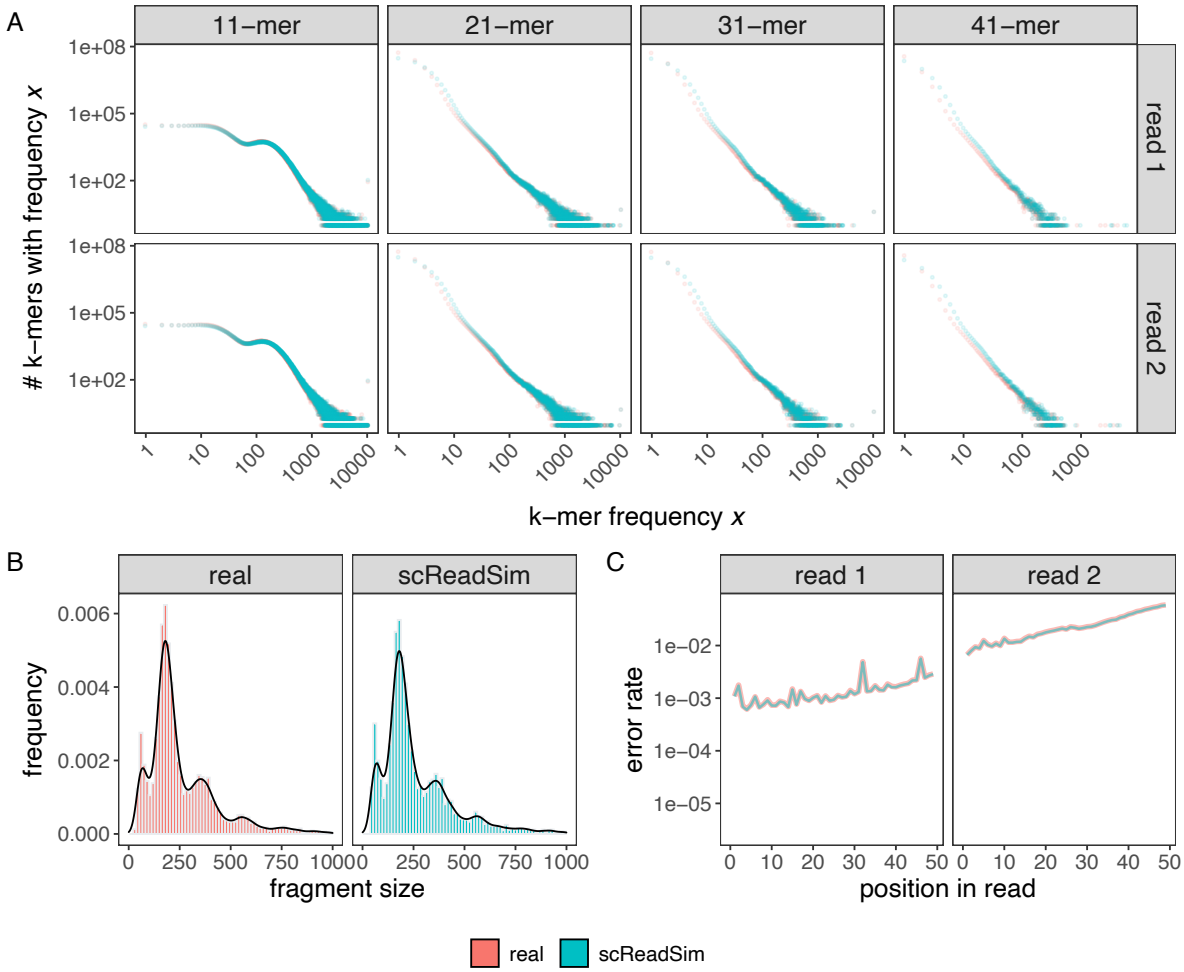

Figure S4: scReadSim's synthetic data resemble real sci-ATAC-seq data [22] at the read sequence level. **A** Comparison of the k-mer spectra between scReadSim's synthetic data and real data. The x-axis refers to the occurrence frequency of a specific k-mer, and the y-axis represents the number of unique k-mers with this frequency. Both the x-axis and y-axis are in the $\log_{10}$ scale. **B** Comparison of the fragment-size distributions between scReadSim's synthetic data and real data. **C** Comparison of the error rate per base call within a read between scReadSim's synthetic data and real data. The x-axis represents the positions of bases within a read, and the y-axis refers to the substitution error rate at each position.

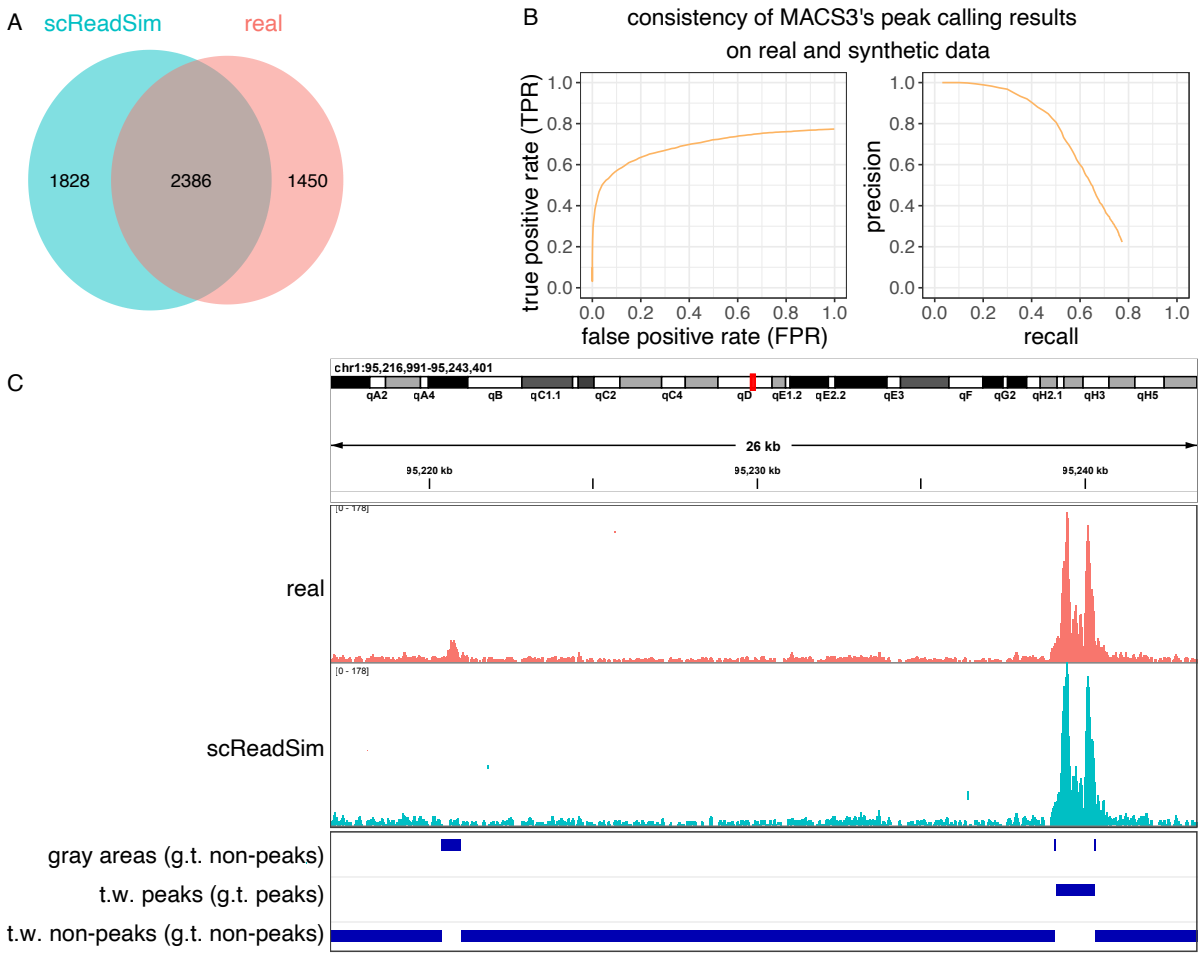

Figure S5: scReadSim's synthetic data resemble real sci-ATAC-seq data [22] in terms of open chromatin regions at the pseudo-bulk level. **A** A Venn diagram of MACS3's called peaks from scReadSim's synthetic data and real data. A peak called from real data (referred to as a "real peak") is considered to overlap with a peak called from synthetic data (referred to as a "synthetic peak") if the real peak has at least 135 bp (half of the shortest real peak's length) overlapped by the synthetic peak. **B** True positive rate (TPR) vs. false positive rate (FPR) (left) and precision vs. recall (right) for evaluating the synthetic peaks in (A) by treating the real peaks in (A) as truths. The TPR, equivalent to the recall, is the proportion of real peaks that have at least 135 bp overlapped by a synthetic peak; the FPR is the proportion of real non-peaks (the regions complementary to the real peaks) that have at least 135 bp overlapped by a synthetic peak; the precision is the proportion of synthetic peaks that overlap at least 135 bp of a real peak. **C** Read coverage comparison of the real and synthetic BAM files in the IGV genome browser. Gray areas, trustworthy (t.w.) peaks, and t.w. non-peaks are user-specified or scReadSim-identified from real data. scReadSim converts gray areas to ground-truth (g.t.) non-peaks, and it maintains trustworthy peaks and non-peaks as ground-truth peaks and non-peaks, respectively, in synthetic data.

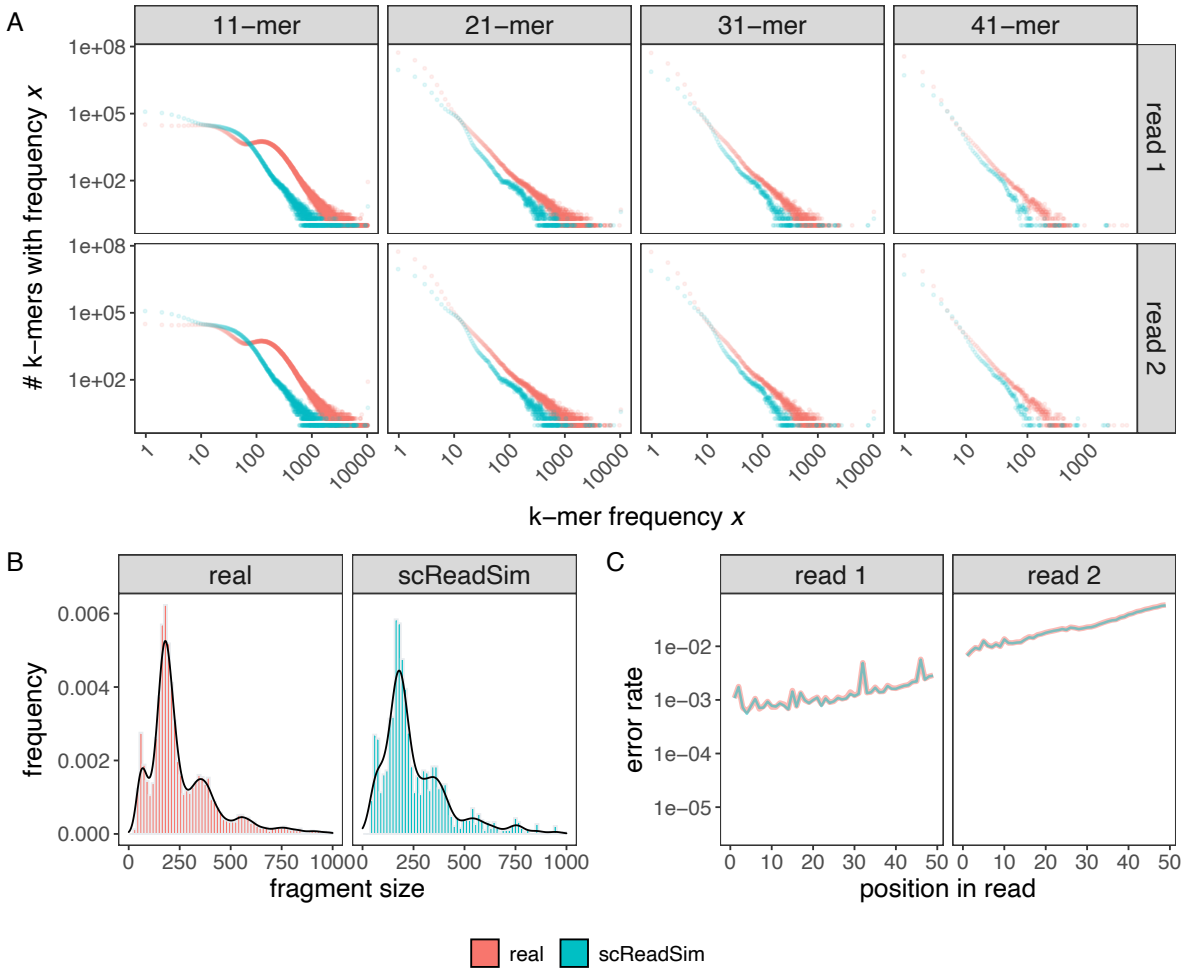

Figure S6: scReadSim's synthetic data with user-designed open chromatin regions resemble real sci-ATAC-seq data [22] at the read sequence level. **A** Comparison of the k-mer spectra between scReadSim's synthetic data and real data. The x-axis refers to the occurrence frequency of a specific k-mer, and the y-axis represents the number of unique k-mers with this frequency. Both the x-axis and y-axis are in the $\log_{10}$ scale. **B** Comparison of the fragment-size distributions between scReadSim's synthetic data and real data. **C** Comparison of the error rate per base call within a read between scReadSim's synthetic data and real data. The line plot's x-axis represents the positions of bases within a read, and the y-axis refers to the substitution error rate corresponding to each position.

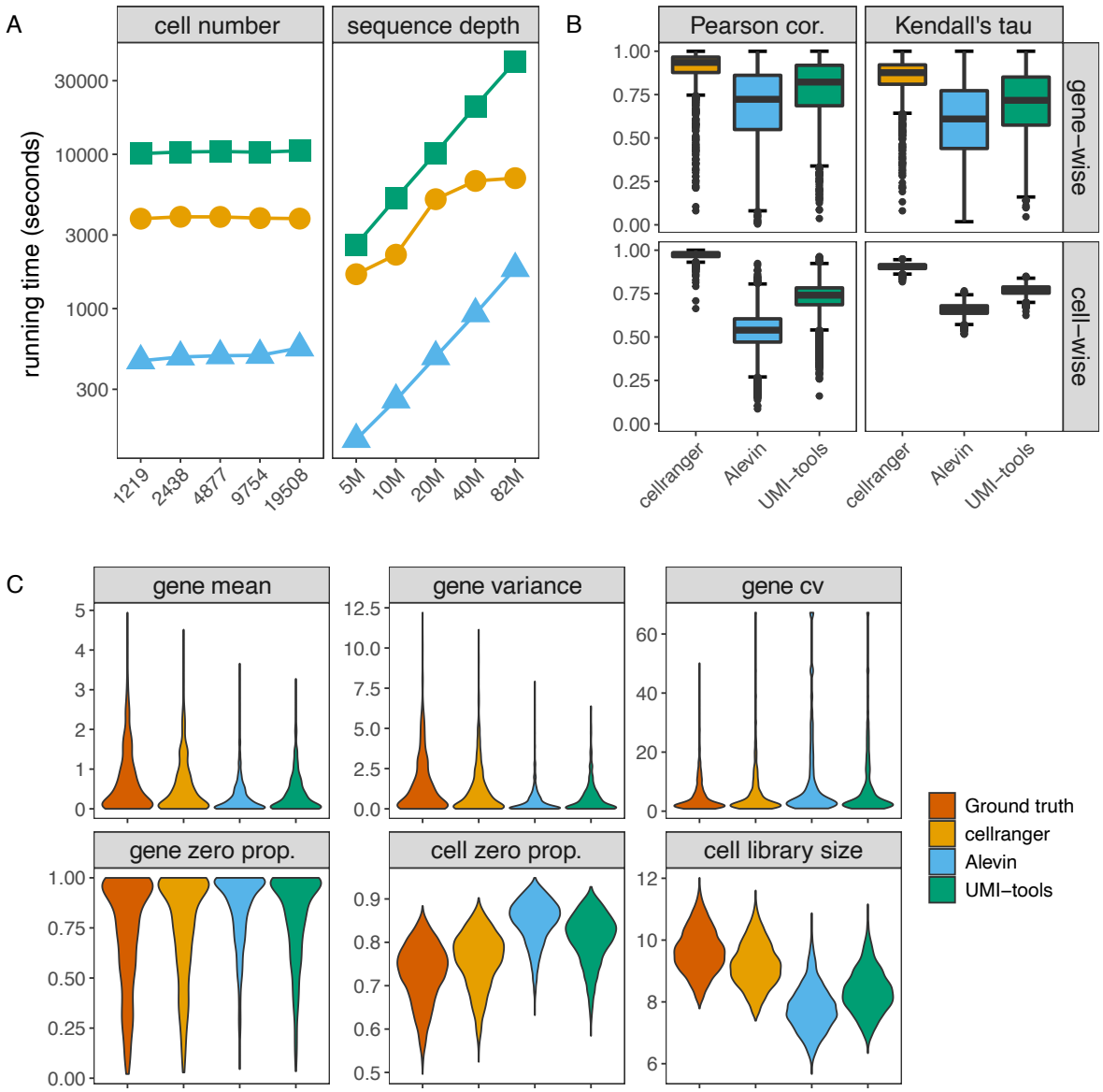

Figure S7: Benchmark of UMI deduplication tools using scReadSim's synthetic scRNA-seq reads. The ground-truth UMI count matrix is the gene-by-cell UMI count matrix generated by scReadSim. Three deduplication tools are considered: cellranger, Alevin, and UMI-tools. **A** Time usage of deduplication tools on synthetic datasets with varying cell numbers or sequencing depths. The y-axis indicates the time lapse (in seconds), and the x-axis reflects the number of synthetic cells or the total number of synthetic reads (sequencing depth). **B** Cell-wise and gene-wise correlations (Pearson correlation and Kendell's tau) between the ground-truth UMI count matrix and each deduplication tool's output UMI count matrix. **C** Summary statistics of the UMI count matrices at the gene level (mean, variance, coefficient of variance (cv), and zero proportion) and the cell level (zero proportion and cell library size).

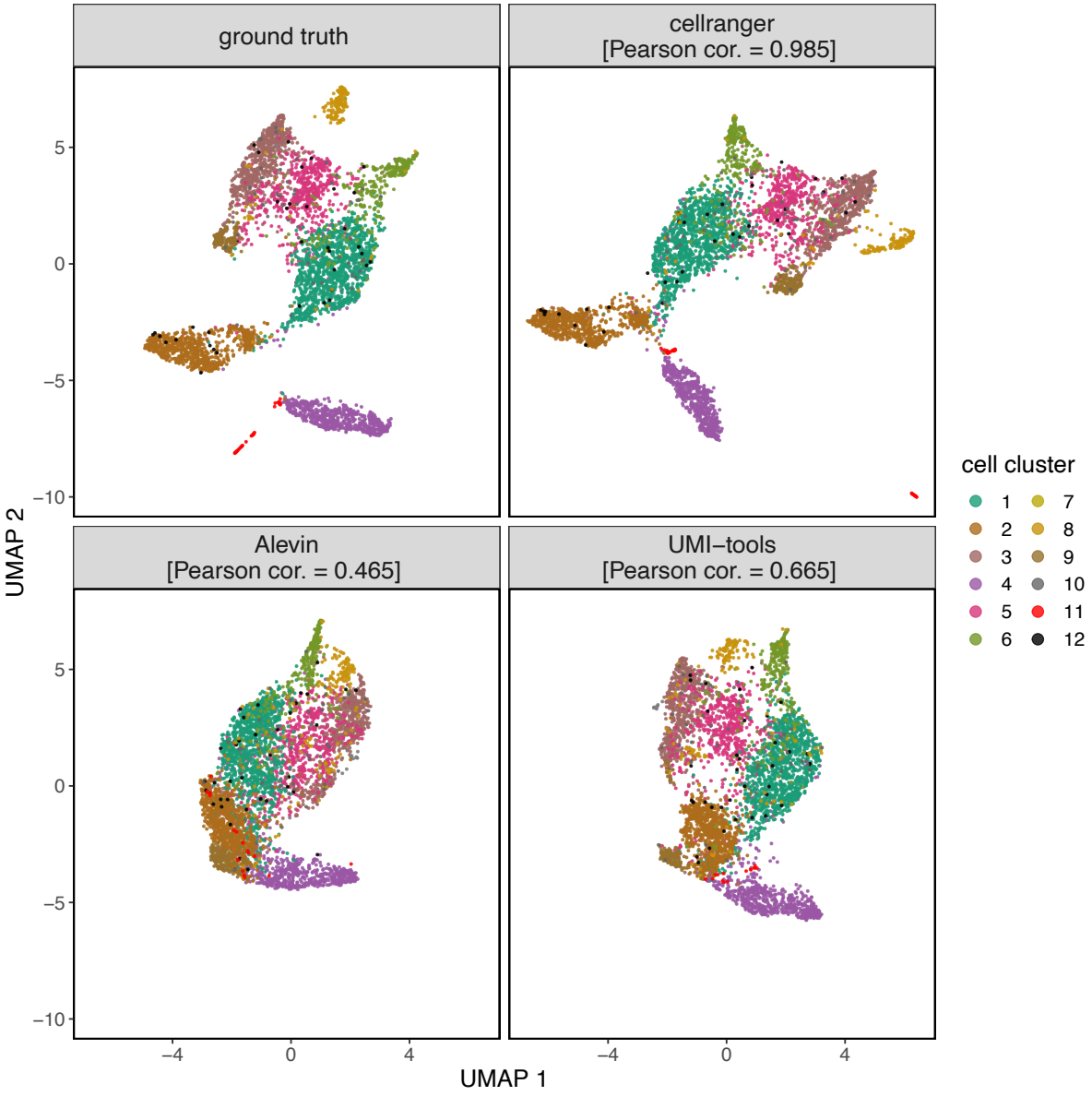

Figure S8: UMAP visualization of the UMI count matrices. Cells are labeled by the cell clusters used as the ground truths in scReadSim. The Pearson correlation ("Pearson Cor.") is calculated between the ground-truth UMI count matrix and each deduplication tool's output UMI count matrix (after expanding each matrix into a vector, the Perason correlation is calculated between the two vectors); the larger the correlation, the greater the similarity between the two matrices.

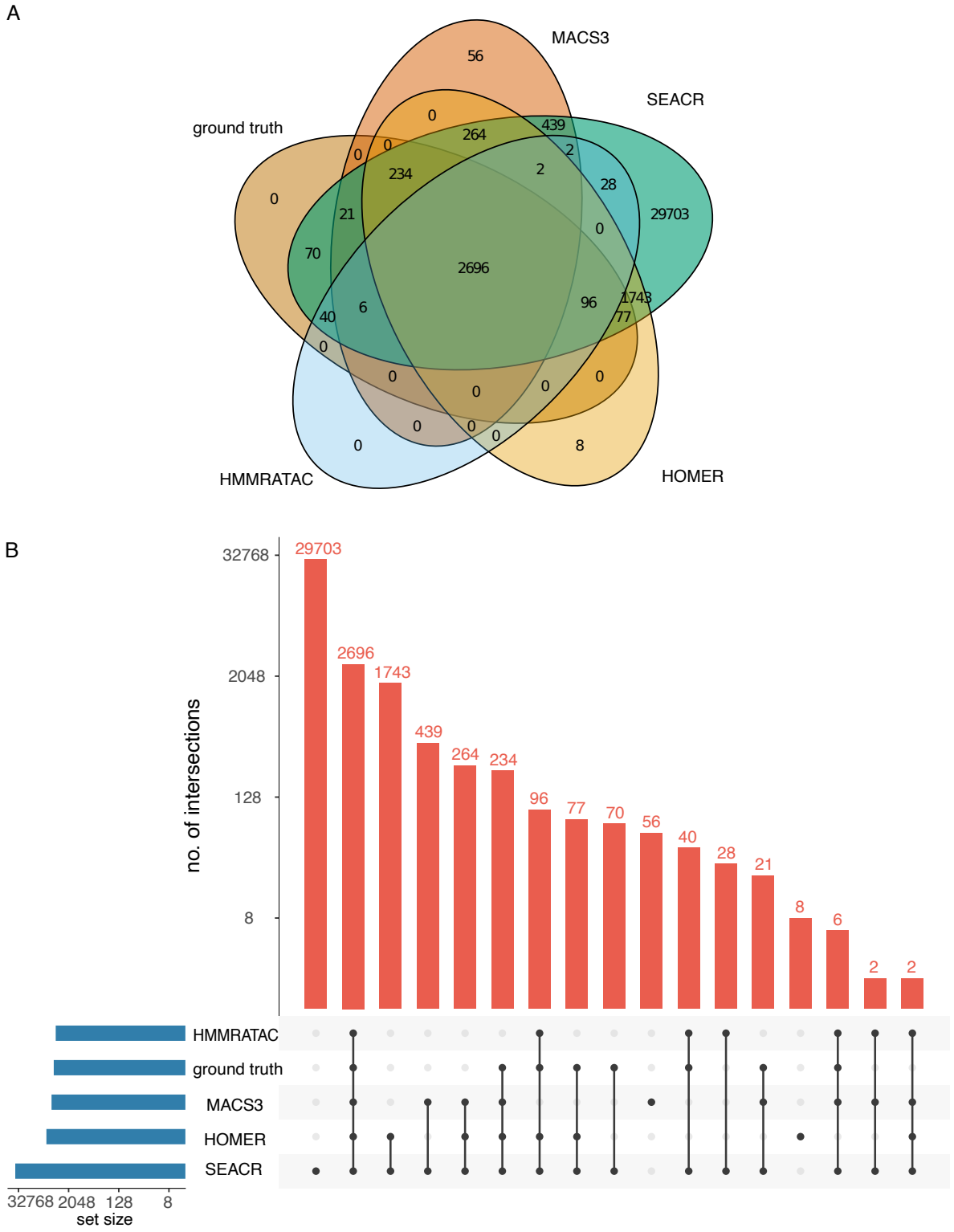

Figure S9: Benchmark of peak-calling tools (MACS3, SEACR, HOMER, and HMMRATAC) using scReadSim's synthetic scATAC-seq reads from user-designed open chromatin regions. The Venn diagram (**A**) and upset plot (**B**) of the user-designed open chromatin regions and the called peaks by MACS3, SEACR, HOMER, and HMMRATAC.