rcCAE: a convolutional autoencoder based method for detecting tumor clones and copy number alterations from single-cell DNA sequencing data

Zhenhua Yu *[1,2], Furui Liu[1], Fangyuan Shi[1,2] and Fang Du[1,2]

[1] *School of Information Engineering, Ningxia University*
[2] *Collaborative Innovation Center for Ningxia Big Data and Artificial Intelligence Co-founded by Ningxia Municipality and Ministry of Education, Ningxia University*

Abstract

Intra-tumor heterogeneity (ITH) is one of the major confounding factors that result in cancer relapse, and deciphering ITH along with copy number alterations (CNAs) is essential for personalized therapy. Single-cell DNA sequencing (scDNA-seq) now enables profiling of single-cell CNAs and thus aids in high-resolution inference of ITH. Here, we introduce an integrated framework called rcCAE, to accurately infer cell subpopulations and single-cell CNAs. A convolutional autoencoder (CAE) is employed in rcCAE to learn latent representation of the cells as well as distill copy number information from noisy read counts data. This unsupervised representation learning via the CAE model makes it convenient to accurately cluster cells over the low-dimensional latent space, and detect single-cell CNAs from improved read counts signal. Extensive performance evaluation on simulated datasets shows rcCAE outperforms existing CNA calling methods, and is more effective in inferring clonal architecture. Furthermore, evaluation of rcCAE on two real datasets demonstrate it is able to provide more refined clonal structure, of which some details are lost in clonal inference based on single-cell absolute copy numbers.

Keywords— Autoencoder, Hidden Markov model, Copy number alteration, Single-cell sequencing

1 Background

The evolutionary history of cancer disease is often depicted as a phylogenetic tree that elucidates the linage relationship between tumor clones [1, 2, 3, 4, 5, 6]. The emergence of each tumor clone is accompanied with a set of newly acquired genomic mutations such as single-nucleotide variation (SNV) and copy number alteration (CNA), and this genetic intra-tumor heterogeneity (ITH) has been one of the major confounding factors that contribute to therapy resistance and cancer relapse [1, 2, 6]. Without a full understanding of ITH, cancer treatment may often target only major clones whereas the low-prevalence clones may gain a growth advantage after therapy, resulting in cancer relapse [7]. Comprehensively charactering ITH is crucial to understand cancer progression and thus help designing personalized treatment [8, 9, 10]. This requires accurate detection of genomic mutations of each tumor clone as well as inference of the phylogenetic tree.

Single-cell DNA sequencing (scDNA-seq) makes it convenient to profile genomes of single cells and thus enables elucidation of ITH based on single-cell SNVs or CNAs [11]. In scDNA-seq, whole-genome amplification (WGA) is employed to amplify the DNA of isolated single cells to required amount of genetic

*Corresponding author: zhyu@nxu.edu.cn

material for sequencing. Typical WGA methods include degenerate oligonucleotide-primed PCR (DOP-PCR) [12, 13], multiple displacement amplification (MDA) [14, 15] and multiple annealing and looping-based amplification cycles (MALBAC) [16]. These methods are all based on exponential amplification and tend to introduce amplification bias and errors into the produced sequences, which makes the sequenced reads error-prone and read count fluctuates extensively across genomic regions. This gives rise to a challenge of how to minimize the effects of these non-biological factors on mutation calling. A few methods have been specifically developed to detect SNVs from scDNA-seq data by considering confounding factors such as allele dropout (ADO) and false positives [17, 18, 19, 20, 21, 22, 23]. For CNA detection, the data generated from DOP-PCR is suitable for calling CNAs but this protocol is limited in its sequencing depth and coverage uniformity. The low sequencing coverage and amplification bias make the distribution of read counts over-dispersed, and this poses a critical challenge of accurately deciphering CNAs from noisy read counts.

The general pipeline to call CNAs from scDNA-seq data mainly consists of three steps: 1) divide the genome into consecutive bins and count reads within each bin (the size of bins can be fixed or variable); 2) preprocess read counts to correct bias caused by GC-content and mappability, remove outlier bins having extremely high or low read counts, and exclude cells that have significantly uneven read counts or are low in sequencing coverage; 3) identify breakpoints that partition the genome into segments (any two adjacent segments have different copy numbers), and call absolute copy number for each segment [24]. To date, a few methods have been specifically proposed to detect single-cell CNAs from scDNA-seq data [25, 26, 27, 28, 29, 30, 31, 32]. These methods can be divided into two categories: detect single-cell CNAs with or without normal cells serving as negative controls. Ginkgo [25] is the first CNA calling method specifically designed for scDNA-seq data. It provides a web platform to detect single-cell CNAs using a circular binary segmentation (CBS) [33] algorithm, and also constructs a tumor phylogenetic tree based on the inferred single-cell copy-number profiles. SCNV [26] first identifies normal cells and pools them together as the composite control, then employs a bin-free segmentation approach to estimate copy numbers. It requires at least 20 normal cells to generate the control, and may be not applicable to datasets containing few normal cells. Similarly, SCOPE [27] uses a generalized likelihood ratio test to jointly segment all cells, and leverages a modified Bayesian information criterion (BIC) [34] to determine the optimal number of segments. It automatically identifies normal cells based on calculation of the Gini coefficient and uses them to remove outlier cells. SCICoNE [28] is developed to simultaneously infer single-cell CNAs and a CNA mutation tree. One disadvantage of SCICoNE is that it treats breakpoint detection and CNA analysis as two separate procedures, which may disseminate errors made during breakpoint detection to downstream CNA analysis. SCYN [29] relies on SCOPE to prepare and preprocess read counts, thereafter employs a dynamic programming algorithm to find the optimal segmentation results by maximizing a simplified modified BIC. The authors also introduce SeCNV [32] to perform cross-sample segmentation on normalized read counts. Recently, a method called SCONCE [31] is proposed to call single-cell CNAs using a hidden Markov model (HMM), and it requires normal cells as negative controls to reduce false CNA calls. Despite to the fact that these methods perform well on their respective test datasets, there are still two problems that are not well addressed in these methods: 1) how to improve the quality of noisy and extensively fluctuated read counts without normal cells as negative controls, such that single-cell CNAs can still be accurately detected; 2) how to infer cell subpopulation composition from the raw read counts rather than the CNA analysis results derived from the read counts, thus obviate propagation of errors made in CNA analysis to clonal structure inference. These two problems are intertwined with each other, therefore integrated frameworks for simultaneously inferring single-cell CNAs and cell subpopulations from raw read counts are sorely required for scDNA-seq data.

Deep generative models such as variational autoencoder (VAE) [35, 36] and generative adversarial network [37, 38] have shown great success in learning representations from single-cell sequencing data [39, 40, 41, 42, 43, 44]. For instance, learning low-dimensional latent representation of single cells using VAE models promises efficient and accurate clustering of single cells [40]. The intuition behind these methods is that observed genetic heterogeneity among single cells actually results from some biological processes that are related to tumor evolution and could be modeled as latent variables in the autoencoders [42]. Inspired by this conception, we develop a novel computational method called rcCAE for analyzing single-cell read

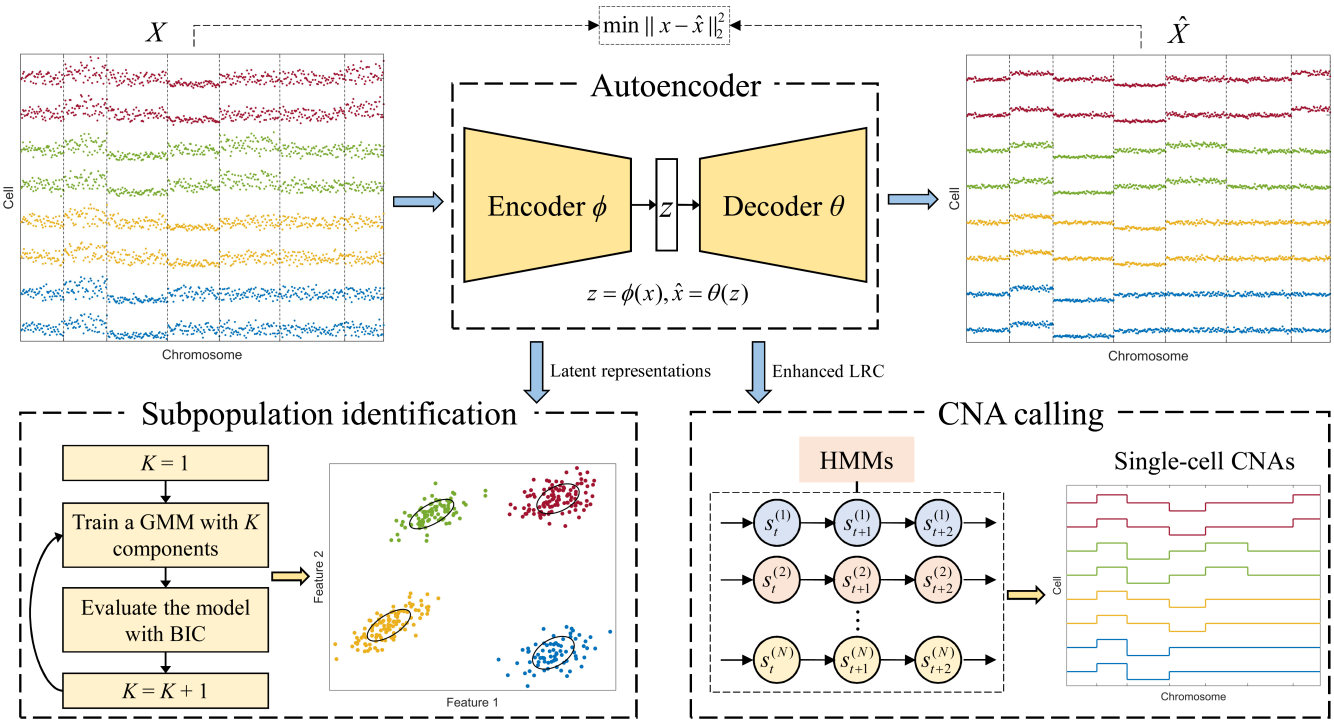

Figure 1: The workflow of rcCAE. rcCAE takes the log2 transformed read counts (LRC) of all cells as inputs, and yields cell subpopulation composition as well as single-cell copy number segments as outputs. With a convolutional autoencoder for unsupervised representation learning, rcCAE projects the LRC data into a latent space where cells are efficiently clustered into distinct subpopulations, and effectively distill copy number information from noisy LRC data. A Gaussian mixture model is employed to identify cell subpopulations based on latent representation of cells, and the number of subpopulations is determined by Bayesian information criterion. For calling CNAs for each cell, rcCAE employs a hidden Markov model to jointly segment the genome and infer absolute copy number for each segment.

counts based on an autoencoder framework (as shown in Figure 1). rcCAE uses a convolutional encoder network to project normalized single-cell read counts into a low-dimensional latent space where the cells are clustered into distinct subpopulations through a Gaussian mixture model (GMM), and leverages a convolutional decoder network to recover the read counts from learned latent representations. The data reconstruction process can be treated as a refinement of the original read counts by attenuating the effects of non-biological confounding factors, thus makes the recovered data being higher in signal-to-noise ratio. We then apply a novel hidden Markov model (HMM) on the recovered data to simultaneously segment the genome and infer absolute copy numbers for each cell. To the best of our knowledge, rcCAE is the first method that applies deep learning to single-cell read counts for CNA calling. We comprehensively evaluate rcCAE on both simulated and real datasets to verify its high scalability on different-sized scDNA-seq datasets, and also compare rcCAE to the state-of-the-art (SOTA) single-cell CNA detection methods. The results showcase our method can accurately characterize cell subpopulation composition in latent space, and is more effective than the competitors in calling absolute copy numbers of single cells.

2 Results and discussion

2.1 The workflow of rcCAE

The framework of rcCAE mainly consists of four modules: 1) a preprocessing module to fetch read counts from the BAM files containing read alignments of sequenced cells, as well as perform filtering and normalization of the read counts; 2) a convolutional autoencoder (CAE) to learn latent representations of single cells and enhance the quality of read counts; 3) a GMM based clustering model to find cell

subpopulations based on the latent representations; and 4) an HMM to predict copy number segments of each cell from the improved read counts. Specifically, encoder of the CAE projects read counts of each cell into a low-dimensional space and the decoder recovers the read counts from the latent space. Given the learned latent features, the GMM model coupled with BIC for model selection is employed to cluster cells into distinct subpopulations. The recovered read counts from the CAE are then analyzed using the HMM to detect CNAs of each cell.

2.1.1 Obtain and preprocess read counts

We get read counts of each cell from sequencing data by first dividing the whole genome into equal-sized bins and then counting the reads overlapping with each bin. GC-content and mappability score of each bin are also calculated using an in-house tool to filter outlier bins and correct bias of read counts. These data filtering and correction steps include: 1) exclude "bad" bins according to GC-content percentage and mappability score; 2) perform library size normalization for each cell to make read counts comparable among cells; 3) exclude bins that have extremely high or low read counts as these bins are considered as confounding outliers; 4) remove cells that have high Gini coefficients calculated based on the normalized read counts; 5) correct read counts bias induced by GC-content for each cell. The resulted normalized read counts of N cells at M bins are denoted by a $N \times M$ matrix R, and we take log2 of R_i to represent the i-th cell. The log2 transformed read counts (LRC) of all cells are denoted by a matrix X. More methodological details about the data preprocessing are provided in section 4.1.

2.1.2 Learn latent representations of single cells

As each x (one row of the X) is a high-dimensional feature vector (e.g. $\sim$5000 bins for human genome if the bin size is 500kb) and often confounded by extensive signal fluctuation, directly applying conventional clustering methods such as K-means and hierarchical clustering on original data may suffer from high computational complexity and low clustering accuracy. Learning latent representations of single cells and performing clustering on the low-dimensional latent data may promise higher performance as demonstrated on single-cell RNA sequencing (scRNA-seq) data [40, 41]. To achieve this, we assume observed genetic heterogeneity among single cells results from some biological processes that could be modeled as latent variables, and there is a function ϕ that maps each x to its latent representation z, i.e. $z = \phi(x)$, while the observed x could be reconstructed by applying a function θ to the latent z, i.e. $\hat{x} = \theta(z)$. Here $\hat{x}$ could be treated as a refined version of x via effective elimination of technically confounding factors. As copy numbers along the genome are local patterns and convolutional kernels are highly effective in learning local features, we therefore use convolutional neural networks to process the LRC data. Specifically, a convolutional autoencoder is employed to reduce the dimensionality and simultaneously enhance the quality of X. The ϕ and θ are implemented as the encoder and decoder of the CAE model, respectively. The encoder ϕ outputs low-dimensional representation z of each input x and the decoder θ outputs a denoised version of x, therefore clustering of the cells can be applied on the reduced data Z and detection of single-cell copy numbers can be applied on $\hat{X}$. The CAE model is trained by minimizing the reconstruction loss, and more details are given in section 4.2.

2.1.3 Infer cell subpopulations and single-cell CNAs

We assume z follow a mixture of multivariate normal distributions, and employ a Gaussian mixture model to cluster cells as adopted in [42]. To find the best number of clusters, denoted by K, we measure BIC of each model associated with a specified value of K, and select the model that has the minimum BIC as the best solution (more details are given in section 4.3).

For detecting CNAs of each cell, we employ an HMM to simultaneously segment and identify absolute copy numbers of the inferred segments. Hidden states of the HMM are integer copy numbers that change from 0 to 10 (the maximum copy number can also be manually specified). We assume the observed LRC follows a normal distribution under each hidden state, and use expectation maximization (EM) algorithm to estimate the model parameters. To determine the ploidy of each cell, baseline shift of LRC

data is considered in the HMM to recognize the most abundant copy number states. After the algorithm converges, the copy number of each bin is deduced from the state that has the maximum posterior probability. We also propose a post-processing step to recalibrate the copy number of highly amplified regions with copy number of >10. More details about the CNAs detection are provided in section 4.4.

2.2 rcCAE shows better performance than the state-of-the-arts on simulated data

2.2.1 Simulation of scDNA-seq datasets

We simulate various scDNA-seq datasets to evaluate the performance of rcCAE. Similar to the simulation procedure used in SimSCSnTree [45], we first generate a clonal tree where each node except root represents a tumor clone and each edge is labeled with newly acquired CNAs by the child node, then sample cells from the tree following the approach adopted in OncoNEM [4]. Given the CNAs of each cell, SCSsim [46] software is used to generate the genome sequence of each cell and simulate reads (sequencing coverage is set to 0.03). The number of tumor clones is set to 15 and the number of cells is set to 500 on all simulations. The size of simulated CNAs changes from 3Mb to 20Mb, and the simulation is conducted on chromosomes 1, 2 and 3 of hg19 reference. Produced reads are aligned to the reference using BWA [47] tool under default parameters, and BAM files are generated using SAMtools [48]. We generate datasets with different tumor ploidy (Simulation A: near-diploidy, Simulation B: near-triploidy and Simulation C: near-tetraploidy), and the average number of simulated CNAs is 18 for diploid, 35 for triploid and 36 for tetraploid samples. For each tumor ploidy, the simulation is repeated 5 times, and this results in 15 datasets for comprehensive assessment of rcCAE. More details about dataset simulation are provided in section 4.5.

2.2.2 Performance evaluation

We compare rcCAE to five state-of-the-art CNA calling methods including SCOPE [27], SCYN [29], SCICoNE [28], SeCNV [32] and SCONCE [31] on the simulated datasets. All these methods use read counts to call single-cell CNAs and perform well on their respective test datasets. We exclude CHISEL [30] from performance comparison as it additionally requires allele frequency for estimating copy numbers. To measure copy number estimation accuracy, we make a comparison between the predicted copy number and the ground truth for each bin, and calculate the accuracy as the proportion of bins whose copy numbers are accurately predicted. In addition, we also assess ploidy estimation performance of each method by comparing the predicted average copy number (ACN) with the ground truth value. For inferring tumor clones, we use adjusted rand index (ARI) to measure the clustering accuracy. The detailed description about the performance assessment and parameter configuration for the investigated methods are provided in Supplementary Methods (Additional file 1).

When running rcCAE on the simulated datasets, we set bin size to 200kb and the number of epochs for training the CAE to 200 on all simulated datasets as training loss changes little after 200 epochs as shown in Figure S1 (Additional file 1). Example results of rcCAE on three simulated datasets are given in Figures S2-S4. It is observed that rcCAE significantly enhances the quality of LRC by decreasing the variance of LRC in each copy number state. This improvement benefits from the adopted encoding-decoding distillation procedure that makes it convenient to distinguish between different copy number states and between distinct tumor clones. As the encoder of the CAE is encouraged to disentangle biologically related factors from the over-dispersed LRC data, the decoder of the CAE is able to reconstruct the LRC by effectively eliminating technically confounded factors. Compared to the over-dispersed original LRC signal, the distribution of reconstructed LRC is highly concentrated near the expected mean value under each copy number state, which suggests naive normal distributions could be employed to model the data. These results demonstrate the proposed CAE framework is useful to enhance the detection accuracy of single-cell CNAs and tumor clones.

2.2.3 Comparison to other methods

We first examine ploidy estimation accuracy of the investigated methods. Difference between the real and predicted ACNs (denoted as ΔACN) is calculated for each cell to indicate the ploidy estimation accuracy. The results in Figure 2 show rcCAE yields better predictions than the SOTAs across different simulations, especially on near-tetraploid datasets. When dealing with near-diploid datasets (Simulation A), SCICoNE overestimates the ploidy and predicts all cells to be near-triploid (median ΔACN is -1.02), while other methods perform well with median ΔACN close to 0. On near-triploid datasets (Simulation B), SCOPE, SCYN and SeCNV perform similarly and accurately estimate the ploidy for most of the samples. SCICoNE tends to overestimate the ploidy (median ΔACN is -0.21) and SCONCE erroneously identifies majority of the cells to be near-tetraploid (median ΔACN is 1). By comparison, our method produces accurate inferences of the ploidy for all cells, the mean and variance of ΔACN are 0.0048 and 1.1×10^{-4}, respectively. On near-tetraploid datasets (Simulation C), all of the existing methods tend to underestimate the ploidy, the median ΔACN of SCOPE, SCYN, SCICoNE, SeCNV and SCONCE are 1.87, 1.87, 0.69, 1.82 and 1.961, respectively, indicating they fail to distinguish between different copy number states on most tetraploid samples. Benefiting from the distillation procedure applied on the LRC, our method is able to assign correct ploidy to each cell, therefore achieves better performance than the competitors and yields median ΔACN close to 0. Figure S5 gives a detailed view of the ploidy estimation results of all methods, and the results show rcCAE' predictions are in concordance with the ground truth.

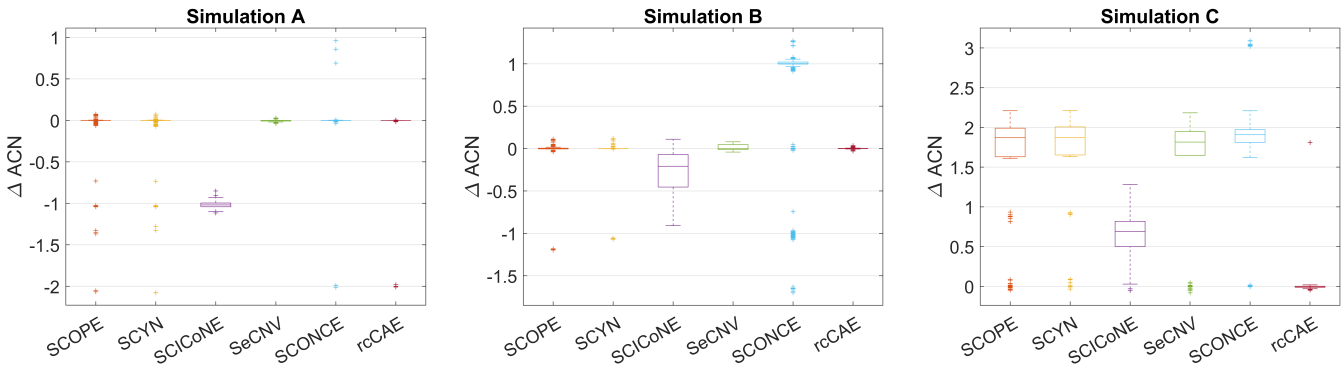

Figure 2: Ploidy estimation accuracy of the investigated methods. Average copy number (ACN) is calculated for each cell, and the difference between real and predicted ACNs (denoted as ΔACN) is used to indicate ploidy estimation accuracy of each method. Results on simulated near-diploid datasets (Simulation A), near-triploid datasets (Simulation B) and near-tetraploid datasets (Simulation C) are analyzed. Each boxplot is generated based on all cells included in the corresponding simulation. rcCAE generally performs better than other methods, especially on the near-tetraploid datasets where all of the existing methods tend to underestimate the ploidy.

We also assess how well rcCAE and other methods perform in estimating absolute copy numbers (as shown in Figure 3). The per-cell copy number accuracy is calculated as the proportion of bins whose copy numbers are accurately detected. To make a fair comparison between different methods, the common bins covered by the results of all methods are used for evaluation on each dataset. The average number of common bins is 2509 for Simulation A, 2636 for Simulation B and 2166 for Simulation C. On near-diploid datasets (Simulation A), SCOPE, SCYN, SeCNV, SCONCE and rcCAE reaches >0.998 median accuracy, while SCICoNE fails to identify absolute copy numbers (median accuracy is 0.016) due to overestimation of the ploidy. Similar performance is observed for SCOPE, SCYN, SeCNV and rcCAE on near-triploid datasets. By comparison, SCICoNE yields median accuracy of 0.698 and SCONCE performs poorly as it overestimates the ploidy and therefore there is an overall offset between the predicted and real copy numbers. On near-tetraploid datasets, the existing methods suffer from degraded performance due to underestimation of the ploidy, and rcCAE surpasses the competitors with detection accuracy of 0.993. As ploidy acts as a scaling factor when assigning copy numbers to different genomic regions, the accuracy of ploidy inference has a heavy effect on CNA calling, and our method shows better robustness in both

aspects. In addition, we measure the accuracy of each method in identifying each copy number state, and the results are shown in Figure S6.

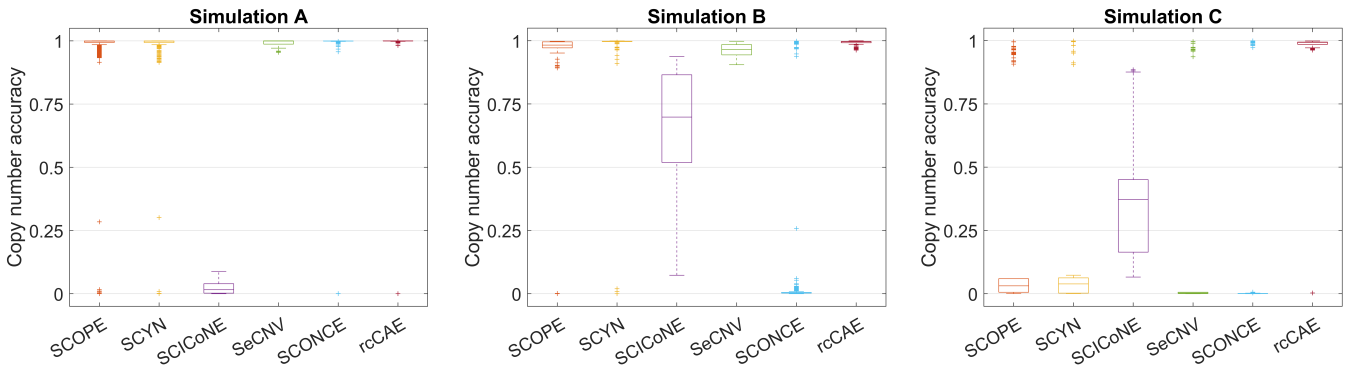

Figure 3: Copy number estimation accuracy of the investigated methods. For each cell, copy number accuracy is calculated as the proportion of bins whose copy numbers are correctly predicted. For each dataset, the bins used for evaluation are the common bins covered by the results of all methods. Results on simulated near-diploid datasets (Simulation A), near-triploid datasets (Simulation B) and near-tetraploid datasets (Simulation C) are compared between the methods. Each boxplot is generated based on all cells included in the corresponding simulation. As rcCAE accurately identifies the most abundant copy numbers for most of the cells, it generally yields better results than other methods especially on the near-tetraploid datasets.

We proceed to evaluate the performance of rcCAE in identifying tumor clones based on the learned latent representations. Two conventional dimensionality reduction methods including PCA and t-SNE are adopted for comparison by clustering the projected data using the same GMM. More details about how t-SNE is used are provided in the Supplementary Methods. An example of the clustering results is given in Figure S7 that shows rcCAE is more effective in disentangling distinct cell subpopulations. The results also indicate distinct clusters are clearly separated in the latent space, which suggests our method effectively captures the difference between CNAs of tumor clones and therefore makes it convenient to distinguish between tumor clones in the low-dimensional space. Figure S8 gives ARI scores of all methods, and suggests our method outperforms PCA and t-SNE across different tumor ploidy. rcCAE exactly decipher clonal composition on 12 out of 15 datasets, and performs acceptedly well on the remaining 3 datasets (ARI > 0.94).

Finally, to check the robustness of rcCAE' performance against data size, we apply it to simulated datasets where the number of cells is in $\{50, 75, 100, 125, 150\}$. The datasets are generated from a near-diploid dataset in Simulation A. Formally, we first randomly select 5 out of 16 subpopulations, then sample cells from each selected subpopulation with the probability proportional to subpopulation size. This process is repeated 10 times to construct a total of 50 datasets for evaluation. The cell clustering, copy number as well as ploidy estimation performance are shown in Figure 4. The results suggest our method is able to deliver accurate predictions of copy number and ploidy across different data sizes, and performs acceptedly well in inferring subpopulation composition when the number of cells is larger than 50. An example of copy number estimation results on a dataset having 75 cells is given in Figure S9. Taken together, we recommend using rcCAE on datasets containing >50 cells to obtain reliable prediction results.

2.2.4 The effects of hyper-parameters

The size of convolutional kernels and dimension of latent space are two important hyper-parameters defining the structure of the CAE. To examine the effects of kernel size (denoted as k) and latent dimension (denoted as d) on the performance of rcCAE, we compare cell clustering and CNA calling results for different configurations of k and d. Specifically, values in $\{3, 5, 7, 9, 11\}$ and $\{2, 3, 4, 5, 6\}$ are tested for k and d, respectively. Copy number estimation results (Figure S10) imply rcCAE performs consistently well

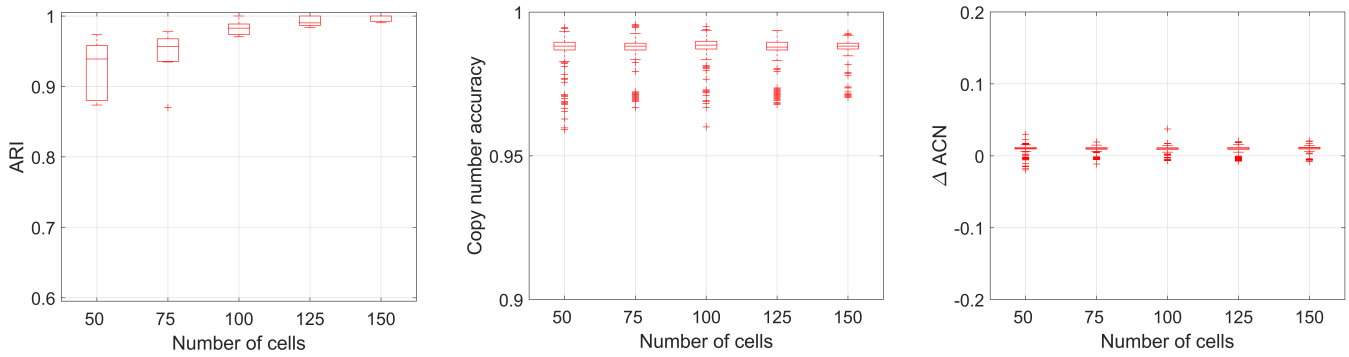

Figure 4: Performance evaluation on different-sized datasets. Values in {50, 75, 100, 125, 150} are tested for the number of cells, and the data are generated from a near-diploid dataset in the Simulation A. The cell clustering, copy number as well as ploidy estimation accuracy are measured with respected to the number of cells, and the results indicate rcCAE could deliver reliable prediction results on datasets containing >50 cells.

in identifying single-cell CNAs regardless of different network structures defined by k and d. This high robustness to network structures benefits from rcCAE's effective distillation of copy number information from noisy LRC data and accurate ploidy estimations (Figure S11). By comparison, the hyper-parameters k and d have significant effects on clustering results (Figure S12). Generally, setting latent dimension to 3 or 4 gives more accurate identifications of cell subpopulations, and median-sized kernels tend to deliver more stable clustering results. To make a good tradeoff between inference accuracy and efficiency, we therefore set k to 7 and d to 3 on all experiments.

2.3 Evaluation of rcCAE on a breast cancer dataset

We apply rcCAE to infer tumor clonal structure from a high grade (III) and triple negative breast ductal carcinoma dataset [13]. This dataset (T10) consists of 100 single cells and is selected to investigate primary tumor growth. The SRA files of all cells are downloaded from NCBI SRA under accession number SRA018951, and converted into FASTQ files using SRA Toolkit. The reads are then aligned to human hg19 reference using BWA tool under default parameters. A previous study [13] has clustered the cells into 4 subpopulations (D+P, H, AA and AB) based on integer copy number profiles, and each subpopulation consists of 47, 24, 25 and 4 cells, respectively. The D+P subpopulation comprises mainly diploid (D) cells and also some pseudodiploid (P) cells that undergo varying degrees of genomic variations. The H subpopulation contains hypodiploid cells that are characterized by broad copy number deletions on most of the chromosomes, and presents an average ploidy of 1.7. The aneuploid subpopulations AA and AB are highly clonal with extensive copy number gains across the whole genome (the average ploidy is 3.0 for AA and 3.3 for AB). Given the highly different copy number profiles between the subpopulations, we assess if rcCAE can decipher the same clonal structure from single-cell raw read counts data.

Read counts data are obtained for every 500kb bins. By applying quality control procedures to the read counts, we get a data matrix with size of 100×5120. The number of epochs for training the CAE is set to 200 (as suggested by the results in Figure S13). Our method clusters the cells into 5 subpopulations each with 45, 24, 25, 5 and 1 cells, respectively (Figure 5 and Figure S14). The clustering results show the subpopulations are clearly separated in the latent space, implying there are significant differences between the copy number profiles of these subpopulations. Cluster 1 consists of predominantly diploid cells that present copy neutral status across nearly the whole genome, and aligns well with the D+P subpopulation. Cells from cluster 2 are characterized by hemizygous deletions spanning chromosomes 1p, 3p, 4q, 5q, 8p, 10q, 13q, 14, 16q and 17p, this results in an average ploidy of 1.7, and the results are in high concordance with the H subpopulation. In addition, the number of cells belonging to cluster 2 is equal to that of the H subpopulation, indicating our method precisely identifies this tumor clone. Cluster 3 is comprised of aneuploid cells that exhibit extensive copy number gains on all chromosomes and hemizygous deletions on chromosomes 4-7. The average ploidy of cells in cluster 3 is 3.1, and the

number of cells is consistent with that of the AA subpopulation, these results demonstrate our method exactly recovers the AA subpopulation. Cluster 4 contains 5 cells of which 4 cells show highly similar copy number profiles. Compared to cluster 3, cluster 4 shows copy number gains on most of the chromosomes and similar hemizygous deletions on chromosomes 4-7, which suggests this cluster may be from the AB subpopulation. Our method additionally finds another cluster 5 formed by only one cell that is featured by hemizygous deletions on chromosomes 2, 3q, 8, 9 and 16. This cell is significantly different from the cells of cluster 2 in the copy number profiles (Figure S14), therefore is classified into a separate cluster by our method. By directly analyzing the raw read counts data, rcCAE is able to provide more refined clonal structure of which some details are lost in clonal inference based on single-cell CNAs.

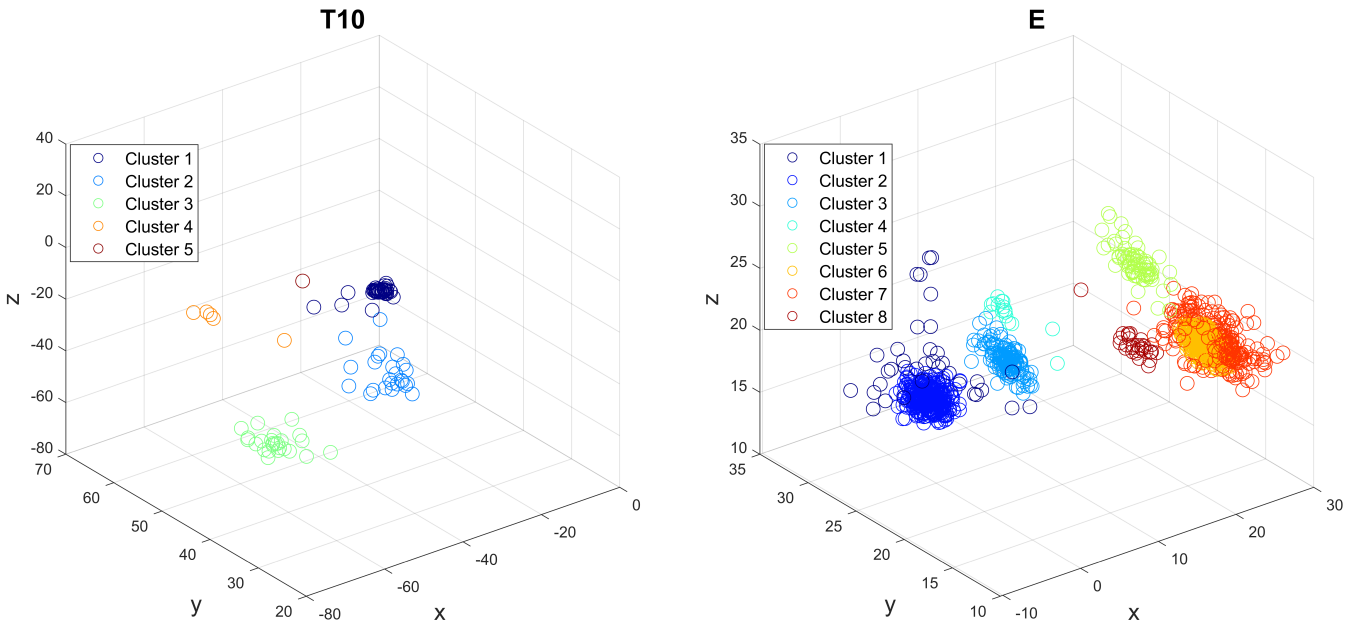

Figure 5: Clustering results of rcCAE on real datasets. On the breast cancer dataset (T10) consisting of 100 cells, rcCAE clusters the cells into 5 subpopulations (labeled as 1-5) each with 45, 24, 25, 5 and 1 cells, respectively (left subfigure). On the 10X Genomics dataset, rcCAE clusters the cells into 8 subpopulations (labeled as 1-8) each with 37, 351, 152, 16, 75, 552, 234 and 29 cells, respectively (right subfigure). The results show these subpopulations are clearly separated in the latent space, implying they present significant different copy number profiles.

2.4 Evaluation of rcCAE on a 10X Genomics dataset

To further verify the performance of rcCAE on large datasets containing thousands of cells, we apply rcCAE to a 10X Genomics dataset freely available at https://support.10xgenomics.com/single-cell-dna/datasets/1.0.0/breast_tissue_E_2k. We download the merged BAM file containing read alignments of 2053 cells and fetch read counts according to barcodes of the cells. A previous study has assigned 1446/2053 of the cells to 6 clusters including one diploid subpopulation (containing 388 cells) and 5 aneuploid tumor clones (each with 168, 58, 20, 782 and 30 cells, respectively) using CHISEL algorithm [30]. The remaining cells are with S-phase of the cell cycle, low sequencing coverage or doublets, therefore they are unclassified [30]. The clustering results are obtained by joint analysis of read counts and allele frequency data. Tumor clones 2 and 3 show highly consistent copy number profiles across the whole genome but different allele-specific copy numbers on chromosome 2, therefore we merge the two clones into a same cluster as they are indistinguishable from each other solely based on the read counts. We now label the new clusters with I-V (each with 388, 168, 78, 782 and 30 cells, respectively). We employ rcCAE to infer clonal structure from this dataset and check if our method can yield similar results as CHISEL by only using read counts data.

The bin size is set to 500kb, and the number of epochs for training the CAE is set to 50 since the training loss changes little after 50 epochs as shown in Figure S15. rcCAE assigns the cells into 8 clusters

(labeled as 1-8) each with 37, 351, 152, 16, 75, 552, 234 and 29 cells, respectively (Figure 5). We take one cell from each cluster to make a comparison between the original and enhanced LRC data, and the results show our method significantly improves the data quality (Figure 6). By investigating the inferred copy numbers of clusters 1 and 2, we find they are diploid subpopulations having highly similar copy number profiles and the total number of cells from them is equal to that of CHISEL cluster I, which suggests rcCAE subdivides the cells of CHISEL cluster I into two sets that are in close proximity in the latent space (Figure 5). Similar results are observed for cluster 3 and 4 (labeled as clone A) that they exhibit consistent copy numbers across the whole genome with average ploidy of 3.55 and align well with the CHISEL cluster II. The main difference between cluster 5 (labeled as clone B) and clone A is that clone B has copy number of 4 on chromosomes 2-3 and copy number of 3 on chromosome 8 while clone A' corresponding copy numbers are 3 and 4 respectively. The results also imply clone B has very similar copy number profiles with the CHISEL cluster III, and rcCAE effectively deciphers this clone by identifying 75/78 of the cells. Clusters 6 and 7 (labeled as clone C) also show quite similar copy number profiles and have copy number of 3 on chromosome 2 when compared to clone B. In addition, 786 cells in clone C completely encompasses the 782 cells from the CHISEL cluster IV, which suggests clone C aligns well with the CHISEL cluster IV. Cluster 8 represents a minor clone (labeled as clone D) that contains 29 cells, and the main difference of clone D from clone B is that it has copy number of 3 on chromosomes 2-3. Based on comparison of the copy number profiles and clone size, we found clone D is in concordance with the CHISEL cluster V. Taken together, rcCAE may be sensitive to LRC changes among the cells belonging to same cluster and tends to generate over-segmented clustering results, however it is still able to accurately recover the copy number profiles of single cells by exploiting strong reconstruction ability of the CAE model.

3 Conclusions

Comprehensively characterizing intra-tumor heterogeneity is essential for understanding tumor evolution and specifying personalized cancer treatments. Single-cell DNA sequencing now provides an unprecedented view of ITH at single-cell resolution. Specifically, inference of ITH based on single-cell copy number profiles has emerged as an important paradigm of scDNA-seq data analysis, and this gives rise to a challenge of how to accurately estimate single-cell copy numbers from highly fluctuated read counts. In this paper, we develop rcCAE, an autoencoder based method for jointly inferring ITH and single-cell copy numbers from scDNA-seq data. The approach leverages a convolutional autoencoder to learn latent representations of single cells as well as enhance the quality of read counts, thus provides an integrated framework that enables efficient characterization of underlying cell subpopulations in a low-dimensional latent space, and accurate detection of single-cell copy numbers from improved read counts by effectively eliminating non-biological confounding factors. As learned representations make the cells be placed into distinct clusters that are clearly separated in latent space, rcCAE simply employs a Gaussian mixture model to infer clonal composition. Compared to ITH inference based on integer single-cell copy numbers, rcCAE directly deciphers ITH from original read counts, which avoids potential error propagation from copy number analysis to ITH inference. For estimating copy number segments for each cell, rcCAE uses a novel hidden Markov model to quantitatively characterize the relationship between observed read counts and underlying copy number, and the HMM explicitly represents tumor ploidy as a key parameter in the emission models to find the solution that best explains the observed read counts. rcCAE is the first method that borrows both the strong representation learning ability of autoencoder and good explanation capability of conventional statistical methods for analyzing single-cell read counts, thus opens a new paradigm for CNA-based ITH analysis.

We simulate various scDNA-seq datasets by adjusting controlling factors of data generation such as tumor ploidy and number of cells. Evaluations on the simulated datasets demonstrate our proposed rcCAE is highly effective in learning latent patterns, thus provides accurate inference of cell subpopulation composition and distillation of copy number information from highly confounded read counts. With a simulation study, we also estimate the minimum number of cells that is required by rcCAE to obtain reliable ITH inferences, and such information could be helpful in the design of upstream scDNA-seq

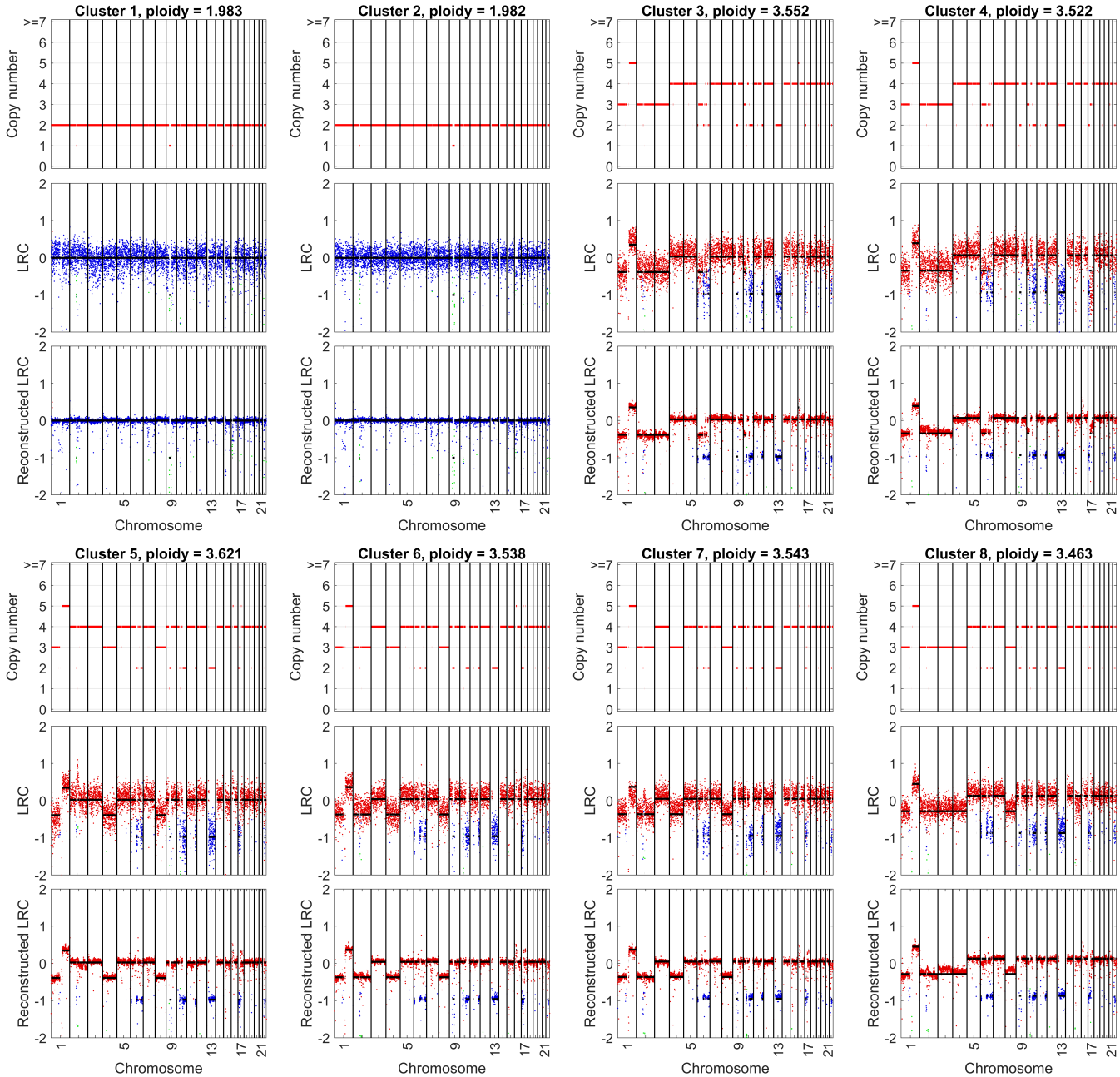

Figure 6: Copy number estimation results of rcCAE on the 10X Genomics dataset. Clusters 1 and 2 are diploid subpopulations having highly similar copy number profiles. Clusters 3 and 4 showcase consistent copy numbers across the whole genome with average ploidy of 3.55 and may come from a same clone (labeled as clone A). Cluster 5 represents another clone (labeled as clone B) that has copy number of 4 on chromosomes 2-3 and copy number of 3 on chromosome 8. Clusters 6 and 7 show quite similar copy number profiles and have copy number of 3 on chromosome 2, which suggests they derive from a same clone (labeled as clone C). Compared to clusters 1-7, cluster 8 represents a minor clone (labeled as clone D), and it has copy number of 3 on chromosomes 2-3. The results also imply the quality of LCR data is significantly enhanced by rcCAE's encoding-decoding framework, thus single-cell CNAs could be accurately detected.

experiments. Using rcCAE, we also elucidate the tumor clonal structure and single-cell copy numbers from a triple negative breast ductal carcinoma dataset and a 10X Genomics dataset. rcCAE yields highly similar partitions of the cells and clonal copy number profiles as reported in existing studies, while provides more delicate clonal structures of which some details are lost in ITH analysis based on single-cell absolute copy numbers.

rcCAE differs from existing approaches in two aspects: 1) instead of inferring ITH based on single-

cell copy number profiles, rcCAE deciphers ITH by learning latent representations of cells from read counts, obviating the need for correcting errors made in CNA calling when analyzing ITH, and thus could give more robust characterization of ITH; and 2) our method has a unique feature of enhancing read counts quality under an unsupervised representation learning, and dose not need to identify normal cells from input data or require matched normal cells to serve as controls. rcCAE implements these two modules with a convolutional autoencoder of which the model parameters are optimized by minimizing the reconstruction loss, this encourages the encoder to recognize local patterns in read counts as well as combine them to effectively represent single cells in a latent space, and the decoder to disentangle actual copy number states from technical factors mixed in read counts. With the cell clustering and CNA detection results of rcCAE, we can use consensus of single-cell copy numbers in each tumor clone to represent clonal copy number profiles, thus aid in fast reconstruction of tumor phylogeny by treating each clone as a unit.

Further improvements in single-cell CNA calling could be achieved by cross-cell segmentation of read counts. First, cells in same clone have a common set of breakpoints, and incorporating this information into CNA detection could aid in better identifying breakpoints. By exploiting cell-to-cluster labels provided by rcCAE, jointly detecting copy number segments of cells in a same clone is a feasible direction to further improve rcCAE's performance. Second, cells from different clones may share same breakpoints since tumor evolves by accumulating mutations [49], and cross-cell segmentation could attenuate the effects of read counts fluctuation and thus deliver better segmentation results. A more challenging task is to perform cross-cell segmentation along with identification of absolute copy numbers. Existing methods that are not based on HMMs often require a post-processing procedure to estimate absolute copy number of each segment after obtaining segmentation results. Despite to the fact that HMM-based methods can provide segmentation as well as absolute copy numbers for a single cell, how to apply HMM to multiple cells is still an open question, and a major challenge is that hidden space size may grow exponentially with the number of cells. Taken together, cross-cell segmentation with simultaneous identification of absolute copy numbers is a promising direction for future study of ITH inference from raw read counts data.

4 Methods

4.1 Obtain and preprocess read counts

Given per-cell BAM files or a merged BAM file containing read alignments of all sequenced cells, we divide the genome into fix-sized bins and obtain read count of a bin by counting reads whose starting positions are within the bin. When a merged BAM file as supported by 10X Genomics is provided, the barcodes of cells need to be specified for identifying the origin of each read alignment. We use BamTools [50] to manipulate the BAM files and fetch information of each read alignment. The fetched read counts are represented by a $N \times M$ matrix, here N denotes the number of cells and M is the number of bins. As GC-content and mappability differs across genomic regions and have unneglectable effects on read counts, we also calculate GC percent and mappability score of each bin to correct read counts bias as well as exclude outlier bins that have extreme values of GC percent or mappability. Specifically, bins that have mappability score of $\geq$0.9 or GC percent of $\leq$10% are excluded as these bins may pose as confounding outliers in downstream analysis [51]. To make read counts comparable among cells, we measure library size as the sum of read counts across all bins for each cell, and apply library size normalization on each cell by dividing read count of each bin with the mean read count of all bins. There are some "bad" bins that locate at highly repetitive or mappable genomic regions, therefore we further exclude bins with mean value of the normalized read counts being in the lower or upper 1% quantile. Considering cells may end up with low sequencing coverage or high amplification bias, we calculate the Gini coefficient for each cell based on the normalized read counts, and eliminate cells having $\geq$0.3 Gini coefficient. These cells present high level of heterogeneity/fluctuation in the read counts and thus shall not be considered in detection of cell subpopulations. Finally, correction for GC-content bias using median normalization approach [52] is applied on each cell to improve the quality of read counts. For computational convenience, we calculate the log2 transformed read counts of all cells, here denoted by a matrix X, for downstream analysis of cell

subpopulation composition and single-cell CNAs.

4.2 Learn latent representations and enhance the quality of LRC

Unlike previous methods that require normal cells as negative controls, we aim to directly infer cell subpopulations and simultaneously detect single-cell CNAs without normalizing LRC using baseline normal cells. This is much challenging due to the high dimensionality and low signal-to-noise ratio of the LRC data. Typically, there are ∼5000 bins on human genome if bin size is set to 500kb, and read counts are often confounded by extensive signal fluctuation, which makes conventional clustering methods not applicable to addressing this problem. Inspired by recent success of deep generative models in learning representations of cells from scRNA-seq data [40, 41], we introduce a convolutional autoencoder based approach to represent single cells in a latent space where cells could be efficiently clustered into distinct subpopulations. Formally, for each x (one row of the X), we use the encoder ϕ of the CAE model to get its latent representation $z = \phi(x)$, and recover x through the decoder network θ of the CAE, i.e. $\hat{x} = \theta(z)$. Here the latent z could be assumed to indicate underlying biological processes that result in the observed cell and are related to mutational events shared by other cells derived from the same tumor clone. By borrowing nonlinear feature learning capability of the encoder, it is expected to effectively eliminate technically confounded factors from the LRC data and thus deliver better representations of the cells. The data reconstruction could be treated as a refinement process that distills copy number information from original LRC data, therefore make it convenient to accurately estimate copy numbers from the recovered LRC.

We train the CAE model to minimize the reconstruction loss $\mathcal{L}(x, \hat{x}; \phi, \theta)$ which is measured using the ℓ_2-norm:

$$\mathcal{L}(x, \hat{x}; \phi, \theta) = \frac{1}{2M} \|x - \hat{x}\|_2^2 \tag{1}$$

During each training iteration, the mean loss is calculated for a batch of samples, and conventional gradient decent algorithm is employed to update the network parameters. After the model training is completed, the latent representations and recovered LRC data of all cells are denoted as Z and $\hat{X}$, respectively.

4.3 Infer cell subpopulations

Given the latent representations Z of all cells, we employ a GMM based clustering approach coupled with Bayesian information criterion for model selection to find cell subpopulations. We select the model that has the minimum BIC as the best solution. Specifically, the value of number of clusters K increases from 1 to a predefined maximum value, and the evaluation terminates if the minimum BIC has remained unchanged at least 10 times. This procedure finally divides the cells into distinct subpopulations, and cells belonging to the same subpopulation have highly similar copy number profiles across whole genome.

4.4 Detect single-cell CNAs

To jointly segment the genome and infer absolute copy numbers for each cell, we employ an HMM to analyze the enhanced LRC data. The HMM defines 11 hidden states each of which represents a copy number that ranges from 0 to 10. Here we use $\hat{X}_{ij}$ to denote the LRC of the i-th cell in the j-th bin. As the variance of the data has been significantly decreased after processing with the CAE model (as shown in Figures 6 and S2-S4), we therefore assume $\hat{X}_{ij}$ follows a normal distribution under hidden state s:

$$f(\hat{X}_{ij}|s, \sigma_i, o_i) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{(\hat{X}_{ij} - \mu_{is})^2}{2\sigma_i^2}\right) \tag{2}$$

with the mean μ_{is} defined as:

$$\mu_{is} = \log_2\left(\frac{c_s}{2}\right) + o_i \tag{3}$$

where o_i denotes baseline shift of the LRC due to change of ploidy, σ_i is a global parameter representing the standard variance of LRC, and c_s represents copy number in state s. We use expectation maximization algorithm to estimate the model parameters. The expected partial log-likelihood is formulated as:

$$E(LL_i) = \sum_{j=1}^{M} \sum_{s=1}^{S} \gamma_{js} \log \left(f(\hat{X}_{ij} | s, \sigma_i, o_i) \right) \tag{4}$$

where the posterior probability γ_{js} that the j-th bin is in state s is calculated using the forward-backward algorithm [53], and S denotes the number of hidden states. During the n-th iteration of the EM algorithm, we update parameter o_i with the following formula:

$$o_i^{(n)} = \frac{\sum_{j=1}^{M} \sum_{s=1}^{S} \gamma_{js}^{(n)} \left(\hat{X}_{ij} - \log_2 \left(\frac{c_s}{2} \right) \right)}{\sum_{j=1}^{M} \sum_{s=1}^{S} \gamma_{js}^{(n)}} \tag{5}$$

and update σ_i by:

$$\sigma_i^{(n)} = \sqrt{\frac{\sum_{j=1}^{M} \sum_{s=1}^{S} \gamma_{js}^{(n)} \left(\hat{X}_{ij} - \log_2 \left(\frac{c_s}{2} \right) - o_i^{(n)} \right)^2}{\sum_{j=1}^{M} \sum_{s=1}^{S} \gamma_{js}^{(n)}}} \tag{6}$$

We adopt a grid search of the initial value of parameter o_i, and initialize σ_i to the standard variance of LRC of the i-th cell. After the algorithm converges, the copy number of each bin is deduced from the state that has the maximum posterior probability. We also propose a post-processing step to recalibrate the copy number of highly amplified regions with copy number of >10, and more details are provided in the Supplementary Methods. Given single-cell copy numbers, a consensus-based approach could be used to deduce copy number states of each subpopulation, thereafter a clonal tree depicting lineage relationship between clones could be efficiently reconstructed.

4.5 Simulate scDNA-seq data

Simulation of scDNA-seq data consists of three steps: (1) mimic a clonal tree that depicts lineage relationship between tumor clones; (2) construct genome sequence of each cell given simulated CNAs; (3) generate sequencing reads of each cell according to parameters such as sequencing coverage and read length. Each node except root of the clonal tree represents a tumor clone, and each edge is labeled with new CNAs acquired by the child node. The clonal tree is constructed as follows: (1) attach the first clone to root, and iteratively attach remaining clones to randomly selected non-root nodes of the tree; (2) generate CNAs (size of the CNAs ranges from 3Mb to 20Mb) and assign them to edges of the tree, the number of CNAs assigned to each edge is proportional to weight of the edge (the edge that connects root and the first clone is given a higher weight of 3, and others are given the same weight of 1). To emulate different tumor ploidy p, we first assign 60% out of the CNAs having copy number of p to the first edge of the tree, then process remaining CNAs according to the weights of the edges; (3) uniformly attach 20% of all cells to the nodes of the tree, and iteratively add remaining cells to the tree. The probability of selecting a node as the attachment point is proportional to current size of the node, which is defined as the number of cells already added to the node, and this promises generation of different-sized subpopulations including major and minor clones. Given the simulated clonal tree, the CNAs of each cell can be obtained by accumulating the mutations along the path from root to the cell. We employ SCSsim [46] software to construct the genome sequence and generate sequencing reads (sequencing coverage is set to 0.03) of each cell. FASTQ files are processed using BWA [47] with default parameters to generate read alignments, and BAM files are generated using SAMtools [48].

4.6 Implementation

The network structure of the CAE is defined as follows: the encoder consists of three 1-D convolutional layers each with 128, 64 and 32 kernels, respectively, the latent layer is a 3-dimensional fully connected

layer, and the decoder consists of a fully connected layer and three transposed 1-D convolutional layers each with 64, 128 and 1 kernels, respectively. Rectified Linear unit (ReLU) activation function is used in all convolutional layers except the last layer of the decoder, and kernel size is set to 7. We use 'Adam' algorithm to train the network with learning rate of 0.0001, and set batch size to 64. The number of epochs is specified per dataset. All experiments are conducted on a server with 1 RTX 2080 Ti GPU, 128GB RAM and 64 CPU cores. The data preprocessing, CAE and clustering modules are implemented in python, the single-cell CNA calling algorithm is implemented in MATLAB.

4.7 Software availability

The source code and detailed documentation of rcCAE are available at https://github.com/zhyu-lab/rccae.

References

[1] Greaves, M., Maley, C.C.: Clonal evolution in cancer. Nature **481**(7381), 306–313 (2012)

[2] Burrell, R.A., McGranahan, N., Bartek, J., Swanton, C.: The causes and consequences of genetic heterogeneity in cancer evolution. Nature **501**(7467), 338–345 (2013)

[3] Yuan, K., Sakoparnig, T., Markowetz, F., Beerenwinkel, N.: BitPhylogeny: a probabilistic framework for reconstructing intra-tumor phylogenies. Genome Biol **16**(1), 36 (2015)

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

[5] El-Kebir, M.: SPhyR: tumor phylogeny estimation from single-cell sequencing data under loss and error. Bioinformatics **34**(17), 671–679 (2018)

[6] Turajlic, S., Sottoriva, A., Graham, T., Swanton, C.: Resolving genetic heterogeneity in cancer. Nature Reviews Genetics **20**(7), 404–416 (2019)

[7] McGranahan, N., Swanton, C.: Clonal heterogeneity and tumor evolution: past, present, and the future. Cell **168**(4), 613–628 (2017)

[8] Oesper, L., Mahmoody, A., Raphael, B.J.: THetA: inferring intra-tumor heterogeneity from high-throughput DNA sequencing data. Genome Biol **14**(7), 80 (2013)

[9] Alizadeh, A.A., Aranda, V., Bardelli, A., Blanpain, C., Bock, C., Borowski, C., Caldas, C., Califano, A., Doherty, M., Elsner, M., *et al.*: Toward understanding and exploiting tumor heterogeneity. Nature medicine **21**(8), 846–853 (2015)

[10] Yu, Z., Li, A., Wang, M.: CloneCNA: detecting subclonal somatic copy number alterations in heterogeneous tumor samples from whole-exome sequencing data. BMC Bioinformatics **17**, 310 (2016)

[11] Gawad, C., Koh, W., Quake, S.R.: Single-cell genome sequencing: current state of the science. Nature Reviews Genetics **17**(3), 175–188 (2016)

[12] Carter, N.P., Bebb, C.E., Nordenskjo, M., Ponder, B.A., Tunnacliffe, A., *et al.*: Degenerate oligonucleotide-primed pcr: general amplification of target dna by a single degenerate primer. Genomics **13**(3), 718–725 (1992)

[13] Navin, N., Kendall, J., Troge, J., Andrews, P., Rodgers, L., McIndoo, J., Cook, K., Stepansky, A., Levy, D., Esposito, D., *et al.*: Tumour evolution inferred by single-cell sequencing. Nature **472**(7341), 90–94 (2011)

[14] Dean, F.B., Hosono, S., Fang, L., Wu, X., Faruqi, A.F., Bray-Ward, P., Sun, Z., Zong, Q., Du, Y., Du, J., *et al.*: Comprehensive human genome amplification using multiple displacement amplification. Proceedings of the National Academy of Sciences **99**(8), 5261–5266 (2002)

[15] Leung, K., Klaus, A., Lin, B.K., Laks, E., Biele, J., Lai, D., Bashashati, A., Huang, Y.-F., Aniba, R., Moksa, M., *et al.*: Robust high-performance nanoliter-volume single-cell multiple displacement amplification on planar substrates. Proceedings of the National Academy of Sciences **113**(30), 8484–8489 (2016)

[16] Zong, C., Lu, S., Chapman, A.R., Xie, X.S.: Genome-wide detection of single-nucleotide and copy-number variations of a single human cell. Science **338**(6114), 1622–1626 (2012)

[17] Zafar, H., Wang, Y., Nakhleh, L., Navin, N., Chen, K.: Monovar: single-nucleotide variant detection in single cells. Nature methods **13**(6), 505–507 (2016)

[18] Dong, X., Zhang, L., Milholland, B., Lee, M., Maslov, A.Y., Wang, T., Vijg, J.: Accurate identification of single-nucleotide variants in whole-genome-amplified single cells. Nature methods **14**(5), 491–493 (2017)

[19] Singer, J., Kuipers, J., Jahn, K., Beerenwinkel, N.: Single-cell mutation identification via phylogenetic inference. Nature communications **9**(1), 1–8 (2018)

[20] Hård, J., Al Hakim, E., Kindblom, M., Björklund, Å.K., Sennblad, B., Demirci, I., Paterlini, M., Reu, P., Borgström, E., Ståhl, P.L., *et al.*: Conbase: a software for unsupervised discovery of clonal somatic mutations in single cells through read phasing. Genome biology **20**(1), 1–18 (2019)

[21] Luquette, L.J., Bohrson, C.L., Sherman, M.A., Park, P.J.: Identification of somatic mutations in single cell dna-seq using a spatial model of allelic imbalance. Nature communications **10**(1), 1–14 (2019)

[22] Lähnemann, D., Köster, J., Fischer, U., Borkhardt, A., McHardy, A.C., Schönhuth, A.: Accurate and scalable variant calling from single cell dna sequencing data with prosolo. Nature communications **12**(1), 1–11 (2021)

[23] Tu, K., Lu, K., Zhang, Q., Huang, W., Xie, D.: Accurate single-cell genotyping utilizing information from the local genome territory. Nucleic acids research **49**(10), 57–57 (2021)

[24] Mallory, X.F., Edrisi, M., Navin, N., Nakhleh, L.: Methods for copy number aberration detection from single-cell dna-sequencing data. Genome biology **21**(1), 1–22 (2020)

[25] Garvin, T., Aboukhalil, R., Kendall, J., Baslan, T., Atwal, G.S., Hicks, J., Wigler, M., Schatz, M.C.: Interactive analysis and assessment of single-cell copy-number variations. Nature methods **12**(11), 1058–1060 (2015)

[26] Wang, X., Chen, H., Zhang, N.R.: Dna copy number profiling using single-cell sequencing. Briefings in bioinformatics **19**(5), 731–736 (2018)

[27] Wang, R., Lin, D.-Y., Jiang, Y.: Scope: a normalization and copy-number estimation method for single-cell dna sequencing. Cell systems **10**(5), 445–452 (2020)

[28] Kuipers, J., Tuncel, M.A., Ferreira, P., Jahn, K., Beerenwinkel, N.: Single-cell copy number calling and event history reconstruction. bioRxiv (2020). doi:10.1101/2020.04.28.065755

[29] Feng, X., Chen, L., Qing, Y., Li, R., Li, C., Li, S.C.: Scyn: single cell cnv profiling method using dynamic programming. BMC genomics **22**(5), 1–13 (2021)

[30] Zaccaria, S., Raphael, B.J.: Characterizing allele-and haplotype-specific copy numbers in single cells with chisel. Nature biotechnology **39**(2), 207–214 (2021)

[31] Hui, S., Nielsen, R.: Sconce: a method for profiling copy number alterations in cancer evolution using single-cell whole genome sequencing. Bioinformatics **38**(7), 1801–1808 (2022)

[32] Ruohan, W., Yuwei, Z., Mengbo, W., Xikang, F., Jianping, W., Shuai Cheng, L.: Resolving single-cell copy number profiling for large datasets. Briefings in Bioinformatics **23**(4), 264 (2022)

[33] Olshen, A.B., Venkatraman, E.S., Lucito, R., Wigler, M.: Circular binary segmentation for the analysis of array-based dna copy number data. Biostatistics **5**(4), 557–572 (2004)

[34] Zhang, N.R., Siegmund, D.O.: Model selection for high-dimensional, multi-sequence change-point problems. Statistica Sinica, 1507–1538 (2012)

[35] Kingma, D.P., Welling, M.: Auto-Encoding Variational Bayes. arXiv (2013). doi:10.48550/ARXIV.1312.6114

[36] Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., Lerchner, A.: beta-VAE: Learning basic visual concepts with a constrained variational framework. In: International Conference on Learning Representations (2017)

[37] Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., Bharath, A.A.: Generative adversarial networks: An overview. IEEE signal processing magazine **35**(1), 53–65 (2018)

[38] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial networks. Communications of the ACM **63**(11), 139–144 (2020)

[39] Wang, D., Gu, J.: Vasc: dimension reduction and visualization of single-cell rna-seq data by deep variational autoencoder. Genomics, proteomics & bioinformatics **16**(5), 320–331 (2018)

[40] Grønbech, C.H., Vording, M.F., Timshel, P.N., Sønderby, C.K., Pers, T.H., Winther, O.: scvae: variational auto-encoders for single-cell gene expression data. Bioinformatics **36**(16), 4415–4422 (2020)

[41] Svensson, V., Gayoso, A., Yosef, N., Pachter, L.: Interpretable factor models of single-cell rna-seq via variational autoencoders. Bioinformatics **36**(11), 3418–3421 (2020)

[42] Rashid, S., Shah, S., Bar-Joseph, Z., Pandya, R.: Dhaka: variational autoencoder for unmasking tumor heterogeneity from single cell genomic data. Bioinformatics **37**(11), 1535–1543 (2021)

[43] Mitra, R., MacLean, A.L.: Rvagene: generative modeling of gene expression time series data. Bioinformatics **37**(19), 3252–3262 (2021)

[44] Li, G., Fu, S., Wang, S., Zhu, C., Duan, B., Tang, C., Chen, X., Chuai, G., Wang, P., Liu, Q.: A deep generative model for multi-view profiling of single-cell rna-seq and atac-seq data. Genome biology **23**(1), 1–23 (2022)

[45] Mallory, X.F., Nakhleh, L.: Simscsntree: a simulator of single-cell dna sequencing data. Bioinformatics **38**(10), 2912–2914 (2022)

[46] Yu, Z., Du, F., Sun, X., Li, A.: Scssim: an integrated tool for simulating single-cell genome sequencing data. Bioinformatics **36**(4), 1281–1282 (2020)

[47] Li, H., Durbin, R.: Fast and accurate short read alignment with burrows–wheeler transform. bioinformatics **25**(14), 1754–1760 (2009)

[48] Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., Durbin, R.: The sequence alignment/map format and samtools. Bioinformatics **25**(16), 2078–2079 (2009)

[49] Swanton, C.: Intratumor heterogeneity: evolution through space and time. Cancer research **72**(19), 4875–4882 (2012)

[50] Barnett, D.W., Garrison, E.K., Quinlan, A.R., Strömberg, M.P., Marth, G.T.: Bamtools: a c++ api and toolkit for analyzing and managing bam files. Bioinformatics **27**(12), 1691–1692 (2011)

[51] Ha, G., Roth, A., Lai, D., Bashashati, A., Ding, J., Goya, R., Giuliany, R., Rosner, J., Oloumi, A., Shumansky, K., *et al.*: Integrative analysis of genome-wide loss of heterozygosity and monoallelic expression at nucleotide resolution reveals disrupted pathways in triple-negative breast cancer. Genome research **22**(10), 1995–2007 (2012)

[52] Yu, Z., Liu, Y., Shen, Y., Wang, M., Li, A.: Climat: accurate detection of copy number alteration and loss of heterozygosity in impure and aneuploid tumor samples using whole-genome sequencing data. Bioinformatics **30**(18), 2576–2583 (2014)

[53] Rabiner, L.R.: A tutorial on hidden markov models and selected applications in speech recognition. Proceedings of the IEEE **77**(2), 257–286 (1989)