scHiCTools: a computational toolbox for analyzing single cell Hi-C data

Fan Feng, Jie Liu

June 2019

1 Introduction

Single-cell Hi-C sequencing (scHi-C) technology (Nagano et al., 2017) allows us to understand chromatin organization dynamics and cell-to-cell heterogeneity, and connects many important genome research areas, including gene regulation and epigenomics. However, interpretation of scHi-C data exposes intrinsic data analysis challenges, such as the fact that Hi-C data are essential two-dimensional pairwise measures rather than one dimensional measures as RNA-seq data and ATAC-seq data, and practical data analysis challenges, such as sparsity of contact maps, batch effect, and sequencing noise.

Previously, similarity measures for comparing Hi-C contact matrices mostly focus on bulk Hi-C data (Yardımcı et al., 2019). These methods evaluate how likely two bulk Hi-C experiments are generated from the same biological sample. In a recent work (Liu et al., 2018), these reproducibility methods have been applied on single cell Hi-C data to evaluate similarity among n single cells, and coupled with multidimensional scaling (MDS) to project these n single cells into a lower dimensional Euclidean space. Among these methods, HiCRep (Yang et al., 2017) yielded the best performance, but its $O(n^2)$ computational complexity makes it impractical when the number of cells is large. In our scHiCTools, we implemented a faster version of HiCRep, together with another Hi-C similarity measure named Selfish (Ardakany et al., 2019), and a new inner product approach which provides a more efficient way of embedding scHi-C data. All of the three approaches have $O(n)$ computational complexity. We demonstrated that the new inner product approach runs faster than original HiCRep, and produces comparably accurate projection. To deal with the sparsity in scHi-C data, three smoothing approaches were implemented, including linear convolution, random walk, and network enhancing (Wang et al., 2018). Among the three, linear convolution appeared to be most effective for smoothing sparse datasets. Our open source toolbox, scHiCTools, as the first toolbox of such kind, can be useful for analyzing scHi-C data.

2 Methods

Three embedding approaches are implemented in scHiCTools. The first approach is a faster implementation of original HiCRep (Yang et al., 2017). Original HiCRep calculates m stratum-adjusted correlation coefficients (SCCs) of the m strata near the diagonal of two contact maps, and then uses weighted sum to aggregate them into one score. It is equivalent to finding a feature vector for each contact map and then computing the inner product among the feature vectors (Supplementary Note 1). This simplification reduces HiCRep's computation complexity from $O(n^2)$ to $O(n)$, and we name it **fastHiCRep**, which is implemented in our toolbox. Alternatively, we can further simplify fastHiCRep by directly setting the concatenated z-normalized strata as feature vectors (Supplementary Note 1). With the feature vectors, an inner product is then calculated to obtain the similarity matrix of a group of cells. We name this second approach **InnerProduct**. In the end, a dimension reduction method, Multidimensional Scaling (MDS), is used to get a lower-dimensional embedding of each cell. The third embedding approach **Selfish** (Ardakany et al., 2019) was recently proposed for bulk Hi-C comparative analysis. It first uses a sliding window to obtain a number of square regions along the diagonal of the contact map, and then counts overall contact numbers in each region. Then, it generates a one-hot "fingerprint matrix" for each contact map based on pairwise comparison of these reads. Gaussian kernels over the fingerprint matrices are calculated as similarities among the cells.

Our toolbox scHiCTools includes three smoothing approaches. **Linear convolution** is based on a 2D filters (a.k.a., convolution kernels) with equal values in every position, which can be viewed as smoothing over nearby bins in Hi-C contact maps. For example, original HiCRep uses a parameter h to describe a $(2h+1)\times(2h+1)$ kernel, i.e. $h = 1$ indicating a 3×3 kernel with each element equals $\frac{1}{9}$. Because this approach is similar to

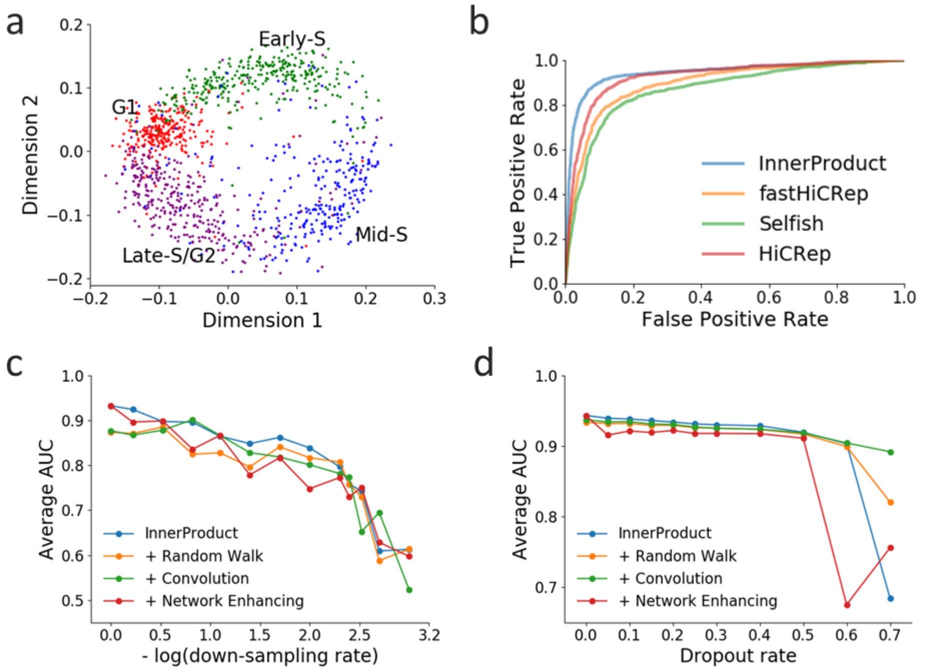

Figure 1: Benchmarking experiment results. (a) The embedding of single cells in a cell cycle study (Nagano et al., 2017). (b) Evaluating the three embedding methods with a cell-cycle phasing task by average ROCs. (c) Smoothing methods do not perform well when all positions in Hi-C maps are randomly downsampled. The x-axis is the negative logarithm of sampling rates; y-axis is the average AUCs from ROC curves. (d) Linear convolution improves the performance of embedding when the dropped out rate is high.

reducing resolution, it is believed to be effective when contact maps are sparse. **Random walk** is a stochastic process updating the elements of the input matrix W by $W' = W \cdot B$, in which $B_{ij} = \frac{W_{ij}}{\sum_i W_{ij}}$. In **network enhancing** (Wang et al., 2018), a special random walk is used to increase gaps between leading eigenvalues of a doubly stochastic contact matrix, which makes the partition of contact maps more prominent, enhancing the boundaries for topologically associated domains (TADs).

3 Results

We benchmarked the projection performance and run time of these methods on a recent scHi-C dataset (Nagano et al., 2017), exactly following the evaluation procedure in a recent work (Liu et al., 2018) (Supplementary Note 2). We had following observations.

InnerProduct produced satisfactory projection. InnerProduct produced satisfactory projection of the single cells (Fig. 1a), achieving an average area under the ROC curve (AUC) of 0.943, which was as good as original HiCRep reported in the recent work (Liu et al., 2018). The AUCs from fastHiCRep and Selfish were relatively lower (Fig. 1b). Implemented fastHiCRep did not perform as well as the original HiCRep in this task, which might due to their subtle difference (Supplementary Note 3).

All the three embedding methods are efficient. The run time of the three methods was compared in Supplementary Table 1. Overall, the three embedding methods were efficient. For embedding 800 cells, all three methods finished within minutes up to an hour. Given the fact that all of the three embedding approaches have $O(n)$ computation complexity, they can scale up very well for a large number of cells. FastHiCRep was slightly slower than InnerProduct, which was slower than Selfish under the default parameters. Note that run time of these approaches depends on parameter settings, which is further discussed in Supplementary Note 4.

Linear convolution smoothing and random walk improves projection at high dropout rates. We applied two sparsification methods on the scHi-C dataset (Nagano et al., 2017), and applied InnerProduct together with the three smoothing approaches, and evaluated the projection performance (see Supplementary Note 5 for additional details). The first sparsification method was used to randomly reduce 40% $\sim$ 99.9% of the contacts for all positions (reducing the contact number from ∼200,000 to ∼500 in each cell). The second one was used to discard contacts from 5% ∼ 60% genomic loci (to simulate dropouts in sequencing data). It was observed that under the second sparsification method, linear convolution and random walk showed some consistent improvement. Linear convolution increased projection accuracy more effectively at higher dropout rates. However, none of the three improved the projection performance when the first sparsification was used.

4 Implementation of scHiCTools

Our scHiCTools is implemented in Python. The source code is available and maintained at https://github.com/liu-bioinfo-lab/scHiCTools. The toolbox is quite easy to use. Users can choose different input formats, including .hic files, sparse matrices in text files, and customized formats. For customized formats, some simple additional information such as reference genome and deliminators is required. Users can also choose the resolution of the contact maps, any of the three smoothing methods (linear convolution, random walk, and network enhancing), any of the three embedding methods (fastHiCRep, Inner Product and Selfish), different ways of aggregating similarity across different chromosomes (taking mean or median), which chromosome(s) to use, and how many strata to use (Supplementary Note 6). In the future, we will keep updating the toolbox with new scHi-C analysis algorithms.

References

A. R. Ardakany, F. Ay, and S. Lonardi. Selfish: Discovery of differential chromatin interactions via a self-similarity measure. *bioRxiv*, 2019.

J. Liu, D. Lin, G. Yardimci, and W. S. Noble. Unsupervised embedding of single-cell Hi-C data. *Bioinformatics*, 34:96–104, 2018.

T. Nagano, Y. Lubling, C. Várnai, C. Dudley, W. Leung, Y. Baran, N. M. Cohen, S. Wingett, P. Fraser, and A. Tanay. Cell-cycle dynamics of chromosomal organization at single-cell resolution. *Nature*, 547:61–67, 2017.

B. Wang, A. Pourshafeie, M. Zitnik, J. Zhu, C. D. Bustamante, S. Batzoglou, and J. Leskovec. Network enhancement as a general method to denoise weighted biological networks. *Nature Communications*, 9(1): 3108, 2018.

T. Yang, F. Zhang, G. G. Y. mcı, F. Song, R. C. Hardison, W. S. Noble, F. Yue, and Q. Li. HiCRep: assessing the reproducibility of Hi-C data using a stratum-adjusted correlation coefficient. *Genome Research*, 27(11): 1939–1949, 2017.

G. G. Yardımcı, H. Ozadam, M. E. Sauria, O. Ursu, K.-K. Yan, T. Yang, A. Chakraborty, A. Kaul, B. R. Lajoie, F. Song, et al. Measuring the reproducibility and quality of Hi-C data. *Genome Biology*, 20(1):57, 2019.