

1 **Deep generative modeling and clustering of single cell Hi-C data**

2 Qiao Liu^{1,†}, Wanwen Zeng^{2,†}, Wei Zhang³, Sicheng Wang⁴, Hongyang Chen⁵, Rui Jiang^{6,*}, Mu
3 Zhou^{7,*} and Shaoting Zhang^{8,*}

4 ¹ Department of Statistics, Stanford University, Stanford, CA 94305, USA;

5 ² College of Software, Nankai University, Tianjin 300071, China;

6 ³ Department of Biomedical Engineering, School of Control Science and Engineering, Shandong
7 University, Jinan, Shandong 250061, China;

8 ⁴ Department of Computer Science and Engineering, University of California San Diego, La Jolla,
9 CA 92093, USA;

10 ⁵ The Research Center for Intelligent Network, Zhejiang Lab, Hangzhou 311121, China;

11 ⁶ Ministry of Education Key Laboratory of Bioinformatics, Research Department of Bioinformatics
12 at the Beijing National Research Center for Information Science and Technology, Center for
13 Synthetic and Systems Biology, Department of Automation, Tsinghua University, Beijing 100084,
14 China;

15 ⁷ SenseBrain Research, San Jose, CA 95131, USA;

16 ⁸ Shanghai Artificial Intelligence Laboratory, Shanghai 200240, China

17

18

19 [†] These authors contributed equally.

20

21 * Corresponding authors:

22 ruijiang@tsinghua.edu.cn; muzhou@sensebrain.site; zhangshaoting@pjlab.org.cn

23

24 **Abstract**

25 **Deciphering 3D genome conformation is important for understanding gene regulation and**
26 **cellular function at a spatial level. The recent advances of single cell Hi-C technologies have**
27 **enabled the profiling of the 3D architecture of DNA within individual cell, which allows us to**
28 **study the cell-to-cell variability of 3D chromatin organization. Computational approaches are**
29 **in urgent need to comprehensively analyze the sparse and heterogeneous single cell Hi-C data.**
30 **Here, we proposed scDEC-Hi-C, a new framework for single cell Hi-C analysis with deep**
31 **generative neural networks. scDEC-Hi-C outperforms existing methods in terms of single cell**
32 **Hi-C data clustering and imputation. Moreover, the generative power of scDEC-Hi-C could**
33 **help unveil the heterogeneity of chromatin architecture across different cell types. We expect**
34 **that scDEC-Hi-C could shed light on deepening our understanding of the complex**
35 **mechanism underlying the formation of chromatin contacts. scDEC-Hi-C is freely available**
36 **at <https://github.com/kimmo1019/scDEC-Hi-C>.**

37 **Keywords:** single cell · 3D genome · deep learning · unsupervised learning.

38

39 **Key points**

- 40 • scDEC-Hi-C provides an end-to-end framework based on autoencoder and deep generative
41 model to comprehensively analyze single cell Hi-C data, including low-dimensional
42 embedding and clustering.
- 43 • Through a series of experiments including single cell Hi-C data clustering and structural
44 difference identification, scDEC-Hi-C demonstrates superior performance over existing
45 methods.
- 46 • In the downstream analysis of chromatin loops from single cell Hi-C data, scDEC-Hi-C is
47 capable of significantly enhancing the ability for identifying single cell chromatin loops by
48 data imputation.

49 **Introduction**

50 The rapid development in single-cell technologies enables us to reliably measure the genomic,
51 transcriptomic and epigenomic features of a particular cellular context at single-cell resolution [1-4].
52 These powerful technologies provide scientists with the opportunity to study the unique patterns of
53 cell type specificity and gene regulation. One fundamental question regarding the abundant single
54 cell data is how to distinguish different cell types in a heterogeneous cell population based on the
55 measured molecular signatures. A variety of computational approaches have been developed to
56 decipher the heterogeneity across cell types based on transcriptome, methylome, and chromatin
57 accessibility [5-11].

58 The majority of the current single-cell assays, such as RNA sequencing (scRNA-seq) and
59 transposase-accessible chromatin using sequencing (scATAC-seq), ignore the spatial information
60 of the genome, such as 3D chromatin structure, which plays an important role in genome functions,
61 including gene transcription and DNA replication [12-14]. The emerging single cell Hi-C
62 technologies bridge this gap by measuring the 3D chromatin structures in individual cells, which
63 have the potential to comprehensively reveal the diverse genome functions underlying the unique
64 genome structure [15-19].

65 Several computational methods have been proposed for the single cell Hi-C data analysis. For
66 example, scHiCluster [20] introduced a random walk-based strategy for data imputation and used
67 PCA for embedding. HiCRep/MDS [21] used multi-dimensional scaling (MDS) for learning a low-
68 dimensional embedding. Higashi [22] is a recent method that utilized hypergraph representation
69 learning for single cell imputation and embedding. However, all these methods require an
70 additional clustering approach (e.g., K-means) for identifying cell types. In addition, choosing the
71 most appropriate clustering approach is sometimes difficult as it is hard for a single clustering

72 approach to perform the best across different datasets. Moreover, modeling the generation process
73 of ultra-sparse single cell Hi-C data could help us better understand the formulation of 3D
74 chromatin conformation, which was ignored by most previous methods.

75 To overcome the above mentioned limitations, we developed scDEC-Hi-C, a comprehensive
76 end-to-end unsupervised learning framework for single cell Hi-C data embedding, clustering, and
77 generation by deep generative neural networks. Unlike existing methods that treat embedding and
78 clustering as two separated tasks, our approach enables simultaneously learning the low-
79 dimensional embeddings of single cell Hi-C data and clustering the single cell Hi-C data by neural
80 network in an unsupervised manner. From systematical experiments, scDEC-Hi-C demonstrates
81 superiority in various tasks, including clustering the cell types, data imputation for quality
82 enhancement, as well as data generation given a desired cell type. To the best of our knowledge,
83 scDEC-Hi-C is the first computational framework that integrates the data embedding and clustering
84 intrinsically for the single cell Hi-C data analysis.

85 **Results**

86 **Overview of scDEC-Hi-C**

87 scDEC-Hi-C consists of two major computational modules, including a convolutional autoencoder
88 module for chromosome-wise representation learning and a deep generative module for cell-wise
89 representation learning and clustering (Fig 1). The autoencoder module aims at extracting the low-
90 dimensional features for each chromosome within a cell. Then the chromosome-wise features are
91 transformed to cell-wise features through a chromosome readout function. We chose global
92 concatenation for the readout function as default. The cell-wise generative model is adopted from
93 our previous work scDEC [23] where G and H networks aim at bidirectional transformation
94 between the m-dimensional latent space and n-dimensional representer space. Note that the latent

95 variables \mathbf{z} follows a standard Gaussian distribution $N(\mathbf{0}, \mathbf{I})$ and \mathbf{c} follows a category distribution
96 $\text{Cat}(K, \mathbf{w})$, which is parameterized by the number of clusters K and the weight \mathbf{w} . G network takes
97 \mathbf{z} and \mathbf{c} as inputs and D_x network was used for matching the distribution of cell-wise representation
98 \mathbf{x} and G network output $\tilde{\mathbf{x}}$ through adversarial training. Similarly, H network and D_z network also
99 work in an adversarial manner where H network could learn the latent representation ($\tilde{\mathbf{z}}$) and infer
100 the cluster ($\tilde{\mathbf{c}}$) simultaneously. The detailed model architecture and training strategy can be found in
101 Supplementary Table 1.

102 **scDEC-Hi-C is capable of identifying cell heterogeneity**

103 A fundamental problem in single cell Hi-C data analysis is to identify different cell types in
104 heterogeneous cell populations. To evaluate the performance of scDEC-Hi-C on this task, we
105 adopted two commonly used benchmark datasets here and systematically compared scDEC-Hi-C to
106 three baseline methods (see Methods for data preprocessing and Supplementary Table 2). Three
107 metrics, including NMI, ARI, and Homogeneity, were introduced for measuring the performance in
108 this unsupervised learning task in order to quantify the ability for distinguishing different cell types
109 in the single cell Hi-C datasets (see Methods). Note that all baseline methods are only able to learn
110 the embedding for each single cell and require additional clustering methods (e.g, K-means) while
111 scDEC-Hi-C simultaneously learns cell embeddings and assigns clustering labels to each cell.
112 scDEC-Hi-C is capable of learning embeddings which could separate cells from different cell types
113 with a relatively larger margin than other baseline methods (Fig.2A-B). It is worth mentioning that
114 scDEC-Hi-C exhibits superior performance on Ramani dataset [17] by outperforming other
115 methods with an ARI of 0.845, compared to 0.826 of Higashi, 0.795 of scHiCluster, and 0.785 of
116 HiCRep/MDS (Fig. 2C). In the second Dip-C dataset [24] where only annotated labels were
117 available, we treat the annotated labels as surrogate ground truth labels. All methods demonstrate

118 significantly lower clustering performance than the Ramani dataset with ground truth label.
119 Specifically, scDEC-Hi-C demonstrates slightly lower performance than Higashi (Fig. 2C). In the
120 readout module in scDEC-Hi-C, the information coming from each chromosome was aggregated.
121 Thus it is worthy to evaluate the contribution of each chromosome. The experimental results show
122 that chromosome 11 contributed the most in Ramani dataset and scDEC-Hi-C consistently
123 outperformed Higashi in 18 chromosomes out of 24 (Fig. 2D). To further investigate the effect of
124 sequencing depth on the clustering performance, we randomly dropout the sequencing reads with
125 different rate. scDEC-Hi-C consistently outperforms all other baseline methods at different dropout
126 rates (Fig. 2E).

127 **scDEC-Hi-C enables the identification of structural differences**

128 In single cell Hi-C data analysis, one fundamental question to ask is whether cell type specificity is
129 revealed by the structural difference regions in Hi-C contacts. The cell type specificity in single cell
130 data, such as single cell RNA-seq and single cell ATAC-seq, can be clearly revealed by marker
131 genes or differential peaks [25]. In bulk Hi-C data, it has been validated that cell type specificity is
132 highly associated with the dynamic chromatin loops within topologically associating domains
133 (TADs) [26, 27]. Therefore, it is worthwhile to investigate whether the structural differences also
134 exist in single cell Hi-C data. To explore this, we used the autoencoder from the first stage of
135 scDEC-Hi-C model as an approach for scDEC-Hi-C imputation. In brief, we segmented Hi-C
136 contact matrix of each chromosome per cell into non-overlapping square patches within the range
137 of 1Mbp. We then treated the output of decoder as the imputed single cell Hi-C data (see Methods).
138 We designed extensive experiments to evaluate whether the imputed single cell Hi-C data could
139 reveal more biological insights than the raw data. We aggregated single cells of K562 and
140 GM12878 cell lines from Ramani dataset and then merged them as the aggregated Hi-C data. In the

141 meanwhile, we also downloaded the bulk Hi-C data from GM12878 and K562 cell lines as ground
142 truth for validation. From the Hi-C profile of a genomic region (chr9: 132.9M-134.9M), K562 and
143 GM12878 have significantly different Hi-C contacts map the difference is also emphasized by the
144 imputed single cell Hi-C data (Fig. 3A). Specifically, the chromatin structural boundaries marked
145 by the rectangle is much clearer by imputed data than the raw data, which demonstrates the power
146 and effectiveness of scDEC-Hi-C in enhancing the resolution of chromatin structural boundaries. It
147 is also noticeable that the chromatin structural boundaries revealed by bulk Hi-C data have a larger
148 consistency with imputed single cell data than the raw single cell data. To further investigate the
149 regulatory landscape of this genomic region, we downloaded both RNA-seq and histone
150 modification data from ENCODE database [28] and visualized them with the help of WashU
151 Epigenome Browser [29]. It can be seen that both RNA-seq signal and H3K4me1 marker are more
152 enriched in K562 cell line than GM12878 cell line in the bounded region (Fig. 3B), which indicates
153 a strong activity of regulatory elements such as enhancer in K562. Next, we designed quantitative
154 experiments to verify whether the resolution of single Hi-C data could be improved by scDEC-Hi-
155 C model. Taking the bulk K562 Hi-C data as ground truth, we calculated the Pearson's correlation
156 of Hi-C interactions of different distances between ground truth and imputed data. It is seen that the
157 interactions at a larger distance are more difficult to impute (Fig. 3C). The correlation between bulk
158 Hi-C data and raw single cell Hi-C data is less than 0.25 while the single cell Hi-C data imputed by
159 scDEC-Hi-C and scHiCluster are much higher than the baseline. scDEC-Hi-C consistently
160 outperforms scHiCluster at different distances ranging from 0 to 1Mb. To sum up, scDEC-Hi-C
161 enables improving the identification of structural boundaries which further helps us study the
162 chromatin structure difference across diverse cell types.

163 **scDEC-Hi-C enhances the discovery of chromatin loops**

164 Chromatin loops are defined as a pair of genomic regions that are brought into spatial proximity,
165 which can be inferred from bulk Hi-C data. Chromatin loops have been proved to be highly
166 relevant to gene regulation, cell fates and functions. We then intended to explore whether the
167 chromatin loops can also be identified within single cell Hi-C data. Similarly, we merged single
168 cell Hi-C data of K562 and GM12878 cell lines, respectively. In the meanwhile, we also
169 downloaded the corresponding bulk Hi-C data for comparison. We applied Fit-Hi-C [30], a
170 computational tool for calling chromatin loops from Hi-C data, to bulk Hi-C data and imputed
171 single cell Hi-C data by scDEC-Hi-C, respectively. There are 6478 chromatin loops in GM12878
172 cell line while 732 (11.3%) chromatin loops are also discovered in imputed single cell Hi-C data
173 (Fig. 4A). scDEC-Hi-C additionally identified 294 chromatin loops which are not contained in the
174 bulk Hi-C chromatin loops. Note that only 196 chromatin loops can be identified from raw single
175 cell Hi-C data and scDEC-Hi-C significantly improves the precision from 1.4% to 11.3% by
176 imputation (Supplementary Figure 1). We visualized the chromatin loops in a genomic region
177 (chr3:118.2M-120.2M) of bulk Hi-C chromatin loops versus either raw single cell Hi-C data (Fig.
178 4B) or imputed single cell Hi-C data (Fig. 4C). In K562 cell line, 12.0% of the chromatin loops
179 from bulk Hi-C data can be also recovered by imputed single cell Hi-C data and 72.5% of the
180 chromatin loops from imputed single cell Hi-C data are also contained in bulk chromatin loops (Fig.
181 4D). In the same genomic region, imputed single cell Hi-C data contains three chromatin loops
182 while two of them were consistent with bulk Hi-C chromatin loops (Fig. 4F) while the raw single
183 cell Hi-C data only has one false chromatin loop (Fig. 4E). To conclude, scDEC-Hi-C is able to
184 promote the identification of chromatin loops from Hi-C data.

185 **Ablation analysis**

186 To systematically evaluate the robustness of scDEC-Hi-C, we designed the following ablation
187 studies. We used Ramani dataset for the ablation studies. First, we removed the cell-wise scDEC
188 module and only kept the chromosome-wise convolutional autoencoder module. We directly used
189 K-means for clustering the features from concatenated autoencoder features. The ARI, NMI and
190 Homogeneity decreases by 6.2%, 7.2%, and 7.1%, respectively. Second, we trained the
191 chromosome-wise autoencoder model first and then fixed the weights in the autoencoder and
192 trained the cell-wise scDEC module. Without joint training of the multi-stage modules, the
193 performance also decreases by 2.4% of ARI, 2.8% of NMI and 2.5% of Homogeneity. The model
194 ablation studies demonstrate the significant contribution of both multi-stage model and joint
195 training strategy.

196

197 **Table 1.** Model ablation studies. The standard deviation of the metric was calculated based on five
198 runs.

	ARI	NMI	Homogeneity
scDEC-Hi-C	0.845±0.010	0.867±0.012	0.819±0.009
scDEC-Hi-C w/o scDEC	0.783±0.009	0.795±0.015	0.748±0.014
scDEC-Hi-C w/o joint training	0.821±0.013	0.839±0.017	0.794±0.013

199

200 **Conclusion and discussion**

201 In this study, we proposed scDEC-Hi-C, a computational tool for comprehensive single cell Hi-C
202 data analysis using deep generative neural network. Unlike previous works that treat dimension
203 reduction and clustering of the single cell Hi-C data as two separated and independent tasks,
204 scDEC-Hi-C intrinsically integrates the task of learning a low-dimensional representation and

205 clustering the single cell by designing a two-stage multi-scale framework, which is composed of a
206 chromosome-wise autoencoder and a cell-wise symmetric GAN model. During the training, the
207 multi-scale models are simultaneously optimized and the results of embedding and clustering are
208 benefitting each other. Based on a series of experiments, scDEC-Hi-C achieves superior or
209 competitive performance compared to state-of-the-art baseline methods. For the downstream
210 analysis, scDEC-Hi-C model demonstrated the excellent ability of imputing the sparse and noisy
211 single cell Hi-C data, which facilitates the identification of chromatin structural differences and
212 chromatin loops. Besides, scDEC-Hi-C also shows the superior power in generating the Hi-C
213 profile of different cell types, which has been confirmed to be consistently with the cell type label
214 (Supplementary Figure 2).

215 We also provide several directions for further improving our work. First, the inter-chromosomal
216 interactions, which were ignored by existing methods and scDEC-Hi-C, have been proved to
217 regulate gene expression [31]. Second, incorporating multi-omics data, including functional
218 genomic regulatory annotation data [32, 33] and pharmaceutical interaction data [34, 35], could
219 potentially improve the performance. Third, it is worthwhile for applying scDEC-Hi-C to other
220 different types of 3D genome interaction data such as HiChIP [36].

221 With scDEC-Hi-C, researchers can perform single cell Hi-C experiments of the cell types or tissues
222 with interest. Then one can simultaneously perform unsupervised learning analysis on single cell
223 Hi-C data and uncover biological findings through the imputation and generation power. We expect
224 that scDEC-Hi-C can help unveil the single cell regulation mechanism in 3D genome.

225 **Methods**

226 **Data preprocessing**

227 For Ramani dataset, we filtered cells with less than 5000 contacts. Then we collected 624 cells for
228 Ramani dataset. For Dip-C dataset, we used the same QC strategy from the original paper[24] and
229 collected 1954 annotated cells across 14 cell types. The details of datasets were summarized in
230 Supplementary Table 2. The raw Hi-C contact matrices were log-transformed and then resized by
231 spline interpolation so that the Hi-C contact matrix of each chromosome was represented as a 50 by
232 50 matrix. Then we applied a mean filtering and random walk as suggested by scHiCluster [20].
233 The chromosome-wise module encodes each chromosome into a 50-dimensional vector and then
234 concatenated across all chromosomes. The cell-wise module further learns a low-dimensional
235 representation of a cell with dimension of latent variable \mathbf{z} set to 10. The embedding of each cell
236 was based on the concatenation of reconstructed $\tilde{\mathbf{z}}$ and $\tilde{\mathbf{c}}$ (before softmax).

237 **Adversarial training in scDEC-Hi-C model**

238 The scDEC-Hi-C is multi-scale unsupervised learning model derived from our previous works
239 Roundtrip and scDEC [23, 37] with extensive modifications. scDEC-Hi-C mainly contains a
240 chromosome-wise module convolutional autoencoder (CAE) [38] and a cell-wise model scDEC.
241 The CAE module aims at mapping scHi-C data from the original data space to a representer space,
242 which significantly reduced the data dimension. Specifically, the CAE module takes the single cell
243 Hi-C interaction of each chromosome as a training instance and each intra-chromosomal interaction
244 matrix will be encoded to a fixed dimension vector through encoder. The embedding vectors for
245 intra-chromosomal interaction matrices within each cell are concatenated in the representer space to
246 obtain a fused embedding. The training of the CAE can be formulated as

$$\mathcal{L}_{AE} = \mathbb{E}[\|\mathbf{x}^{chr} - D(E(\mathbf{x}^{chr}))\|_F^2]$$

247 where \mathbf{x}^{chr} denotes an intra-chromosomal Hi-C interaction matrix and $E(\cdot)$, $D(\cdot)$ denote the
 248 encoder and decoder in the CAE module, respectively. The chromosome-wise features $E(\mathbf{x}^{chr})$ of
 249 each chromosome were concatenated to obtain the cell-wise representation by

$$\mathbf{x} = \text{Concat}(E(\mathbf{x}^{chr1}), \dots, E(\mathbf{x}^{chrX}))$$

250 The scDEC module takes cell-wise fused embedding in the representer space as input and learns
 251 the low-dimension embedding of a cell in the latent space and clusters the cells simultaneously.
 252 scDEC module is composed of a pair of two GAN models. For the forward GAN model, a pair of
 253 latent variables \mathbf{z} and \mathbf{c} are sampled from a Gaussian distribution and a Categorical distribution,
 254 respectively. The categorical distribution is updated through an adaptive mechanism
 255 (Supplementary Table 3). G network is used for conditionally generating fake data $\{\tilde{\mathbf{x}}_i\}_{i=1}^N$ that have
 256 a similar distribution to the real data $\{\mathbf{x}_i\}_{i=1}^N$ in the representer space while the discriminator
 257 network D_x tries to discern true data from generated samples in the representer space. In the
 258 backward GAN model, the function H and the discriminator D_z aim at transforming the data from
 259 representer space to the latent space. Discriminators can be considered as binary classifiers where
 260 any input data point will be asserted to be positive or negative. Besides, we used WGAN-GP [39]
 261 as the architecture for the pair of GAN models where the gradient penalties of discriminators were
 262 considered as additional loss terms. We then define the objective loss functions of the above four
 263 networks (G, H, D_x and D_z) in the training process as

$$\begin{cases} \mathcal{L}_{GAN}(G) = - \mathbb{E}_{\mathbf{z} \sim p(\mathbf{z}), \mathbf{c} \sim \text{Cat}(K, \mathbf{w})} [D_x(G(\mathbf{z}, \mathbf{c}))] \\ \mathcal{L}_{GAN}(D_x) = - \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} [D_x(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p(\mathbf{z}), \mathbf{c} \sim \text{Cat}(K, \mathbf{w})} [D_x(G(\mathbf{z}, \mathbf{c}))] + \lambda \mathbb{E}_{\hat{\mathbf{x}} \sim \hat{p}(\hat{\mathbf{x}})} [(\|\nabla_{\hat{\mathbf{x}}} D_x(\hat{\mathbf{x}})\|_2 - 1)^2] \\ \mathcal{L}_{GAN}(H) = - \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} [D_z(H(\mathbf{x}))] \\ \mathcal{L}_{GAN}(D_z) = - \mathbb{E}_{\mathbf{z} \sim p(\mathbf{z})} [D_z(\mathbf{z})] + \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} [D_z(H(\mathbf{x}))] + \lambda \mathbb{E}_{\bar{\mathbf{z}} \sim \bar{p}(\bar{\mathbf{z}})} [(\|\nabla_{\bar{\mathbf{z}}} D_z(\bar{\mathbf{z}})\|_2 - 1)^2] \end{cases}$$

264 where $p(\mathbf{z})$ and $\text{Cat}(K, \mathbf{w})$ denote the distribution of the continuous variable and discrete variable
265 in the latent space. In practice, sampling \mathbf{x} from $p(\mathbf{x})$ can be regarded as a process of randomly
266 sampling from *i.i.d* data in the representer space with replacement. $\hat{p}(\hat{\mathbf{x}})$ and $\bar{p}(\bar{\mathbf{z}})$ denote a
267 uniformly sampling from the straight line between a pair of points sampled from true data and
268 generated data in the representer and latent space, respectively. λ is a penalty coefficient which is
269 set to 10 in all experiments.

270 **Roundtrip loss**

271 During the training process, we also intend to minimize the roundtrip loss [37] which is defined as
272 $\rho((\mathbf{z}, \mathbf{c}), H(G(\mathbf{z}, \mathbf{c})))$ and $\rho(\mathbf{x}, G(H(\mathbf{x})))$ where \mathbf{z} and \mathbf{c} are sampled from $p(\mathbf{z})$ and $\text{Cat}(K, \mathbf{w})$,
273 respectively. The basic principle for this loss is to minimize the distance when a data point goes
274 through a roundtrip transformation between two different data domains. Specifically, we applied l_2
275 loss to the continuous part in roundtrip loss and cross entropy loss to the discrete part in roundtrip
276 loss. We further denoted the roundtrip loss as

$$\mathcal{L}_{RT}(G, H) = \alpha \|\mathbf{x} - G(H(\mathbf{x}))\|_2^2 + \alpha \|\mathbf{z} - H_z(G(\mathbf{z}, \mathbf{c}))\|_2^2 + \beta CE(\mathbf{c}, H_c(G(\mathbf{z}, \mathbf{c})))$$

277 where α and β are two coefficients and are both set to 10 in the experiments. $H_z(\cdot)$ and $H_c(\cdot)$
278 denote the continuous and discrete part of $H(\cdot)$, respectively. $CE(\cdot)$ represents the cross-entropy
279 function. The idea of roundtrip loss which exploits transitivity for regularizing structured data has
280 also been used in previous works [40, 41].

281 **Joint training**

282 Combining the adversarial training loss and roundtrip loss together, we can get the full training loss
283 for the scDEC module as $\mathcal{L}(G, H) = \mathcal{L}_{GAN}(G) + \mathcal{L}_{GAN}(H) + \mathcal{L}_{RT}(G, H)$ and $\mathcal{L}(D_x, D_z) =$
284 $\mathcal{L}_{GAN}(D_x) + \mathcal{L}_{GAN}(D_z)$, respectively. We iteratively updated the weight parameters in two

285 generative models (G and H) and the two discriminative models (D_x and D_z), respectively. Thus,
286 the training of scDEC module can be represented as

$$G^*, D_x^*, H^*, D_z^* = \begin{cases} \arg \min_{G, H} \mathcal{L}(G, H) \\ \arg \min_{D_x, D_z} \mathcal{L}(D_x, D_z) \end{cases}$$

287 To further achieve joint training of CAE and scDEC modules, we first pretrained the CAE module
288 for 100 epochs. Then we updated the parameters of CAE and scDEC iteratively. The Adam
289 optimizer [42] with a learning rate of 2×10^{-4} was used for optimizing the parameters in neural
290 networks. The whole training process is illustrated in Supplementary Table 4 in detail.

291 **Data imputation by scDEC-Hi-C model**

292 We use the chromosome-wise model autoencoder for data imputation. Specifically, the
293 reconstructed Hi-C map from the decoder was regarded as the imputed single cell Hi-C data. We
294 used the same strategy in [13] for Hi-C matrices extraction.

295 **Data generation by scDEC-Hi-C model**

296 We generate the intermediate cell state (embeddings) of single cell Hi-C data by interpolating the
297 latent indicator \mathbf{c} of two “neighboring” cell types. Assume that two cell types correspond to the
298 latent indicator \mathbf{c}_1 and \mathbf{c}_2 , respectively. The generated single cell Hi-C profile can be represented as
299 $G(\mathbf{z}, \hat{\mathbf{c}})$ where $\hat{\mathbf{c}} = \alpha \mathbf{c}_1 + (1 - \alpha) \mathbf{c}_2$. Note that the α is the coefficient from 0 to 1 and \mathbf{z} is sampled
300 from a standard Gaussian distribution.

301 **Network architecture in scDEC-Hi-C**

302 For the CAE module, the encoder contains four convolutional layers and two fully connected layers
303 while the decoder consists of two fully connected layers and four transposed convolutional layers
304 for reconstructing the Hi-C interaction matrices. For the scDEC module. The G network contains
305 ten fully connected layers and each hidden layer has 512 nodes while the H network contains ten

306 fully-connected layers and each hidden layer has 256 nodes. D_x and D_z both contain two fully
 307 connected layers and 256 nodes in the hidden layer. Note that batch normalization [43] was used in
 308 discriminator networks.

309 **Updating the Category distribution**

310 The probability parameter \mathbf{w} in the Category distribution $\text{Cat}(K, \mathbf{w})$ is adaptively updated every
 311 200 batches of data based on the inferred cluster label (Supplementary Table 3).

312 **Evaluation metrics for clustering**

313 We compared different methods for clustering according to three commonly used metrics,
 314 normalized mutual information (NMI) [44], adjusted Rand index (ARI) [45] and Homogeneity [46].
 315 Assuming that U and V are true label assignment and predicted label assignment given n
 316 observation data points, which have C_U and C_V clusters in total, respectively. NMI is then
 317 calculated as

$$\text{NMI} = \frac{\sum_{p=1}^{C_U} \sum_{q=1}^{C_V} |U_p \cap V_q| \log \frac{n |U_p \cap V_q|}{|U_p| \times |V_q|}}{\max \left(-\sum_{p=1}^{C_U} |U_p| \log \frac{|U_p|}{n}, -\sum_{q=1}^{C_V} |V_q| \log \frac{|V_q|}{n} \right)}$$

318 The Rand index [47] is a measure of agreement between two cluster assignments while ARI
 319 corrects lacking a constant value when the cluster assignments are selected randomly. We define
 320 the following four quantities: 1) n_1 : number of pairs of two objects in the same groups in both U
 321 and V , 2) n_2 : number of pairs of two objects in different groups in both U and V , 3) n_3 : number of
 322 pairs of two objects in the same group of U but different group in V , 4) n_4 : number of pairs of two
 323 objects in the same group of V but different group in U . Then ARI is calculated by

$$\text{ARI} = \frac{\binom{n}{2} (n_1 + n_4) - [(n_1 + n_2)(n_1 + n_3) + (n_3 + n_4)(n_2 + n_4)]}{\binom{n}{2} - [(n_1 + n_2)(n_1 + n_3) + (n_3 + n_4)(n_2 + n_4)]}$$

324 Homogeneity is calculated by $\text{Homo} = 1 - \frac{H(U|V)}{H(U)}$, where

$$\begin{cases} H(U|V) = - \sum_{p=1}^{C_U} \sum_{q=1}^{C_V} \frac{|U_p \cap V_q|}{n} \log \frac{|U_p \cap V_q|}{\sum_{q=1}^{C_V} |U_p \cap V_q|} \\ H(U) = - \sum_{p=1}^{C_U} \frac{\sum_{q=1}^{C_V} |U_p \cap V_q|}{C_U} \log \frac{\sum_{q=1}^{C_V} |U_p \cap V_q|}{C_U} \end{cases}$$

325 **Baseline methods**

326 We compared scDEC-Hi-C to three comparison methods in our study. scHiCluster is a PCA-based
327 method that could be used for imputing and clustering scHi-C data. scHiCluster was implemented
328 from <https://github.com/zhoujt1994/scHiCluster> and the default parameters were used.
329 HiCRep/MDS used multidimensional scaling to embed scHi-C data into two dimension and was
330 implemented from <https://github.com/liu-bioinfo-lab/scHiCTools>. Higashi is a hypergraph
331 representation learning framework for embedding scHi-C data. We downloaded Higashi from
332 <https://github.com/ma-compbio/Higashi> and implemented using the default parameters.

333

334 **Data availability**

335 Three datasets were used in this study. scHi-C dataset of four human cell lines (GM12878, HAP1,
336 HeLa and K562) was collected from Ramani et al (GEO: GSE84920). scHi-C dataset of mouse
337 brain development was collected from Tan et al (GEO: GSE162511). Note that the first dataset has
338 ground truth cluster label for each cell. The latter dataset only contains annotated labels, which
339 were used as surrogate labels in the clustering experiments.

340 **Code availability**

341 scDEC-Hi-C is an open-source software based on the TensorFlow library [48], which can be
342 downloaded from <https://github.com/kimmol019/scDEC-Hi-C>.

343 **Acknowledgement**

344 The authors thank the anonymous reviewers for their valuable and constructive suggestions.

345 **Author contributions statement**

346 Q.L., R.J., M.Z. and S.T.Z. conceived the study. Q.L. designed and implemented scDEC-Hi-C. Q.L.
347 and W.W.Z. performed data analysis. Q.L. and W.W.Z. interpreted the results. Q.L. wrote the
348 manuscript. Other authors provided editorial support. All authors read and approved the final
349 version of the manuscript.

350 **Funding**

351 This work was supported by National Natural Science Foundation of China (No. 62003178). Key
352 Research Project of Zhejiang Lab (No. 2022PI0AC01). The National Key Research and
353 Development Program of China grant no. 2021YFF1200902, the National Natural Science
354 Foundation of China grants nos. 61873141, and a grant from the Guoqiang Institute, Tsinghua
355 University.

356 **Competing interests**

357 The authors declare no competing interests.

358

359 **Reference**

- 360
- 361 1. Buenrostro JD, Wu B, Litzenburger UM et al. Single-cell chromatin accessibility reveals
362 principles of regulatory variation, *Nature* 2015;523:486-490.
 - 363 2. Gawad C, Koh W, Quake SR. Single-cell genome sequencing: current state of the science,
364 *Nat Rev Genet* 2016;17:175-188.
 - 365 3. Shapiro E, Biezuner T, Linnarsson S. Single-cell sequencing-based technologies will
366 revolutionize whole-organism science, *Nat Rev Genet* 2013;14:618-630.
 - 367 4. Stoeckius M, Hafemeister C, Stephenson W et al. Simultaneous epitope and transcriptome
368 measurement in single cells, *Nat Methods* 2017;14:865-868.
 - 369 5. Stuart T, Butler A, Hoffman P et al. Comprehensive integration of single-cell data, *Cell*
370 2019;177:1888-1902. e1821.
 - 371 6. Butler A, Hoffman P, Smibert P et al. Integrating single-cell transcriptomic data across
372 different conditions, technologies, and species, *Nat Biotechnol* 2018;36:411-420.
 - 373 7. Liu Q, Xia F, Yin Q et al. Chromatin accessibility prediction via a hybrid deep
374 convolutional neural network, *Bioinformatics* 2018;34:732-738.
 - 375 8. Duren Z, Chang F, Naqing F et al. Regulatory analysis of single cell multiome gene
376 expression and chromatin accessibility data with scREG, *Genome Biol* 2022;23:114.
 - 377 9. Yin Q, Wu M, Liu Q et al. DeepHistone: a deep learning approach to predicting histone
378 modifications, *BMC genomics* 2019;20:11-23.
 - 379 10. Liu Q, Hua K, Zhang X et al. DeepCAGE: Incorporating transcription factors in genome-
380 wide prediction of chromatin accessibility, *Genomics Proteomics Bioinformatics* 2022.
 - 381 11. Yin Q, Liu Q, Fu Z et al. scGraph: a graph neural network-based approach to automatically
382 identify cell types, *Bioinformatics* 2022.
 - 383 12. Rao SS, Huntley MH, Durand NC et al. A 3D map of the human genome at kilobase
384 resolution reveals principles of chromatin looping, *Cell* 2014;159:1665-1680.
 - 385 13. Liu Q, Lv H, Jiang R. hicGAN infers super resolution Hi-C data with generative adversarial
386 networks, *Bioinformatics* 2019;35:i99-i107.
 - 387 14. Marchal C, Sima J, Gilbert DM. Control of DNA replication timing in the 3D genome, *Nat*
388 *Rev Mol Cell Biol* 2019;20:721-737.
 - 389 15. Nagano T, Lubling Y, Stevens TJ et al. Single-cell Hi-C reveals cell-to-cell variability in
390 chromosome structure, *Nature* 2013;502:59-64.
 - 391 16. Flyamer IM, Gassler J, Imakaev M et al. Single-nucleus Hi-C reveals unique chromatin
392 reorganization at oocyte-to-zygote transition, *Nature* 2017;544:110-114.
 - 393 17. Ramani V, Deng X, Qiu R et al. Massively multiplex single-cell Hi-C, *Nat Methods*
394 2017;14:263-266.
 - 395 18. Stevens TJ, Lando D, Basu S et al. 3D structures of individual mammalian genomes studied
396 by single-cell Hi-C, *Nature* 2017;544:59-64.
 - 397 19. Tan L, Xing D, Chang CH et al. Three-dimensional genome structures of single diploid
398 human cells, *Science* 2018;361:924-928.
 - 399 20. Zhou J, Ma J, Chen Y et al. Robust single-cell Hi-C clustering by convolution- and random-
400 walk-based imputation, *Proc Natl Acad Sci U S A* 2019;116:14011-14018.
 - 401 21. Liu J, Lin D, Yardimci GG et al. Unsupervised embedding of single-cell Hi-C data,
402 *Bioinformatics* 2018;34:i96-i104.
 - 403 22. Zhang R, Zhou T, Ma J. Multiscale and integrative single-cell Hi-C analysis with Higashi,
404 *Nat Biotechnol* 2022;40:254-261.

- 405 23. Liu Q, Chen S, Jiang R et al. Simultaneous deep generative modeling and clustering of
406 single cell genomic data, *Nat Mach Intell* 2021;3:536-544.
- 407 24. Tan L, Ma W, Wu H et al. Changes in genome architecture and transcriptional dynamics
408 progress independently of sensory experience during post-natal brain development, *Cell*
409 2021;184:741-758 e717.
- 410 25. Hao Y, Hao S, Andersen-Nissen E et al. Integrated analysis of multimodal single-cell data,
411 *Cell* 2021;184:3573-3587 e3529.
- 412 26. Ramirez F, Bhardwaj V, Arrigoni L et al. High-resolution TADs reveal DNA sequences
413 underlying genome organization in flies, *Nat Commun* 2018;9:189.
- 414 27. Wang R, Chen F, Chen Q et al. MyoD is a 3D genome structure organizer for muscle cell
415 identity, *Nature communications* 2022;13:205.
- 416 28. Consortium EP. An integrated encyclopedia of DNA elements in the human genome,
417 *Nature* 2012;489:57-74.
- 418 29. Li D, Hsu S, Purushotham D et al. WashU Epigenome Browser update 2019, *Nucleic Acids*
419 *Res* 2019;47:W158-W165.
- 420 30. Ay F, Bailey TL, Noble WS. Statistical confidence estimation for Hi-C data reveals
421 regulatory chromatin contacts, *Genome research* 2014;24:999-1011.
- 422 31. Xiong K, Ma J. Revealing Hi-C subcompartments by imputing inter-chromosomal
423 chromatin interactions, *Nat Commun* 2019;10:5069.
- 424 32. Zeng W, Chen S, Cui X et al. SilencerDB: a comprehensive database of silencers, *Nucleic*
425 *acids research* 2021;49:D221-D228.
- 426 33. Chen S, Liu Q, Cui X et al. OpenAnnotate: a web server to annotate the chromatin
427 accessibility of genomic regions, *Nucleic Acids Res* 2021;49:W483-W490.
- 428 34. Xu C, Liu Q, Huang M et al. Reinforced molecular optimization with neighborhood-
429 controlled grammars, *Advances in neural information processing systems* 2020;33:8366-8377.
- 430 35. Liu Q, Hu Z, Jiang R et al. DeepCDR: a hybrid graph convolutional network for predicting
431 cancer drug response, *Bioinformatics* 2020;36:i911-i918.
- 432 36. Mumbach MR, Rubin AJ, Flynn RA et al. HiChIP: efficient and sensitive analysis of
433 protein-directed genome architecture, *Nat Methods* 2016;13:919-922.
- 434 37. Liu Q, Xu J, Jiang R et al. Density estimation using deep generative neural networks,
435 *Proceedings of the National Academy of Sciences* 2021;118:e2101344118.
- 436 38. Masci J, Meier U, Cireşan D et al. Stacked convolutional auto-encoders for hierarchical
437 feature extraction. In: *International conference on artificial neural networks*. 2011, p. 52-59.
438 Springer.
- 439 39. Gulrajani I, Ahmed F, Arjovsky M et al. Improved training of wasserstein gans. In:
440 *Advances in neural information processing systems*. 2017, p. 5767-5777.
- 441 40. Yi Z, Zhang H, Tan P et al. Dualgan: Unsupervised dual learning for image-to-image
442 translation. In: *Proceedings of the IEEE international conference on computer vision*. 2017, p.
443 2849-2857.
- 444 41. Zhu J-Y, Park T, Isola P et al. Unpaired image-to-image translation using cycle-consistent
445 adversarial networks. In: *Proceedings of the IEEE international conference on computer vision*.
446 2017, p. 2223-2232.
- 447 42. Kingma DP, Ba J. Adam: A method for stochastic optimization, *arXiv preprint*
448 *arXiv:1412.6980* 2014.
- 449 43. Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing
450 internal covariate shift, *arXiv preprint arXiv:1502.03167* 2015.

- 451 44. Strehl A, Ghosh J. Cluster ensembles---a knowledge reuse framework for combining
452 multiple partitions, *Journal of Machine Learning Research* 2002;3:583-617.
- 453 45. Hubert L, Arabie P. Comparing partitions, *Journal of classification* 1985;2:193-218.
- 454 46. Rosenberg A, Hirschberg J. V-measure: A conditional entropy-based external cluster
455 evaluation measure. In: *Proceedings of the 2007 joint conference on empirical methods in natural*
456 *language processing and computational natural language learning (EMNLP-CoNLL)*. 2007, p. 410-
457 420.
- 458 47. Rand WM. Objective criteria for the evaluation of clustering methods, *Journal of the*
459 *American Statistical association* 1971;66:846-850.
- 460 48. Abadi M, Barham P, Chen J et al. Tensorflow: A system for large-scale machine learning.
461 In: *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*.
462 2016, p. 265-283.
- 463

464

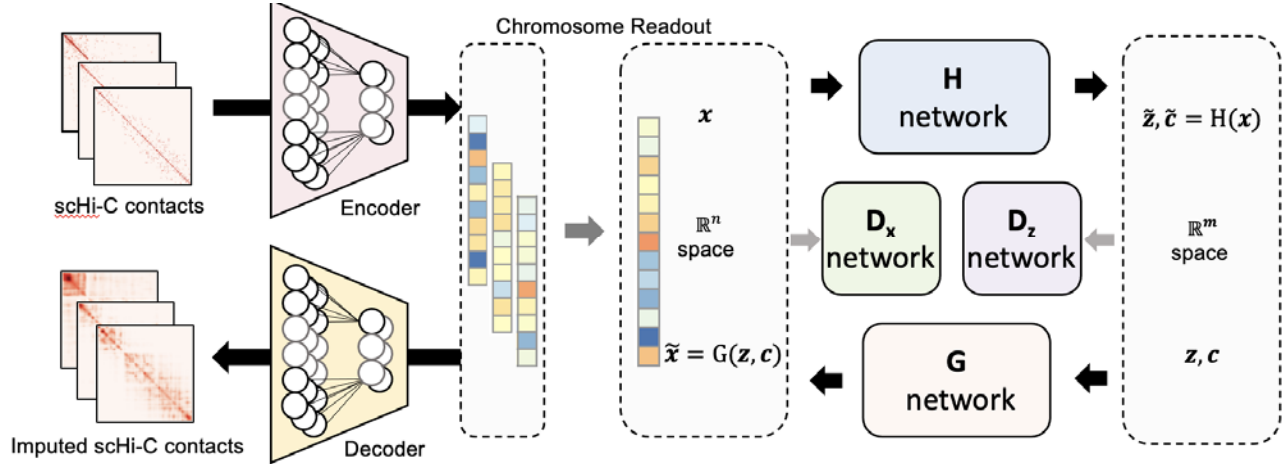


Figure1. The overview of the proposed scDEC-Hi-C model. scDEC-Hi-C is a multi-scale model which contains a chromosome-wise convolutional autoencoder (CAE) and a cell-wise single cell deep embedding and clustering model. The intra-chromosome single-cell Hi-C contacts matrices are first fed to a CAE for dimension reduction and latent feature extraction. Then the chromosome readout (e.g., concatenation) is applied to get the cell-wise representation. The cell-wise deep generative neural networks can further learn a low dimensional representation of a cell and cluster each cell simultaneously. In the latent space, latent variables z and c sampled from a Gaussian distribution and a Category distribution respectively, are fed to the H network. The H network has two outputs of which one corresponds to the latent embedding (\tilde{z}) and one corresponds to the estimated cluster label (\tilde{c}). The D_x and D_z discriminator networks are used for adversarial training.

465

466

467

468

469

470

471

472

473

474

475

476

477

478

479

480

481

482

484
485

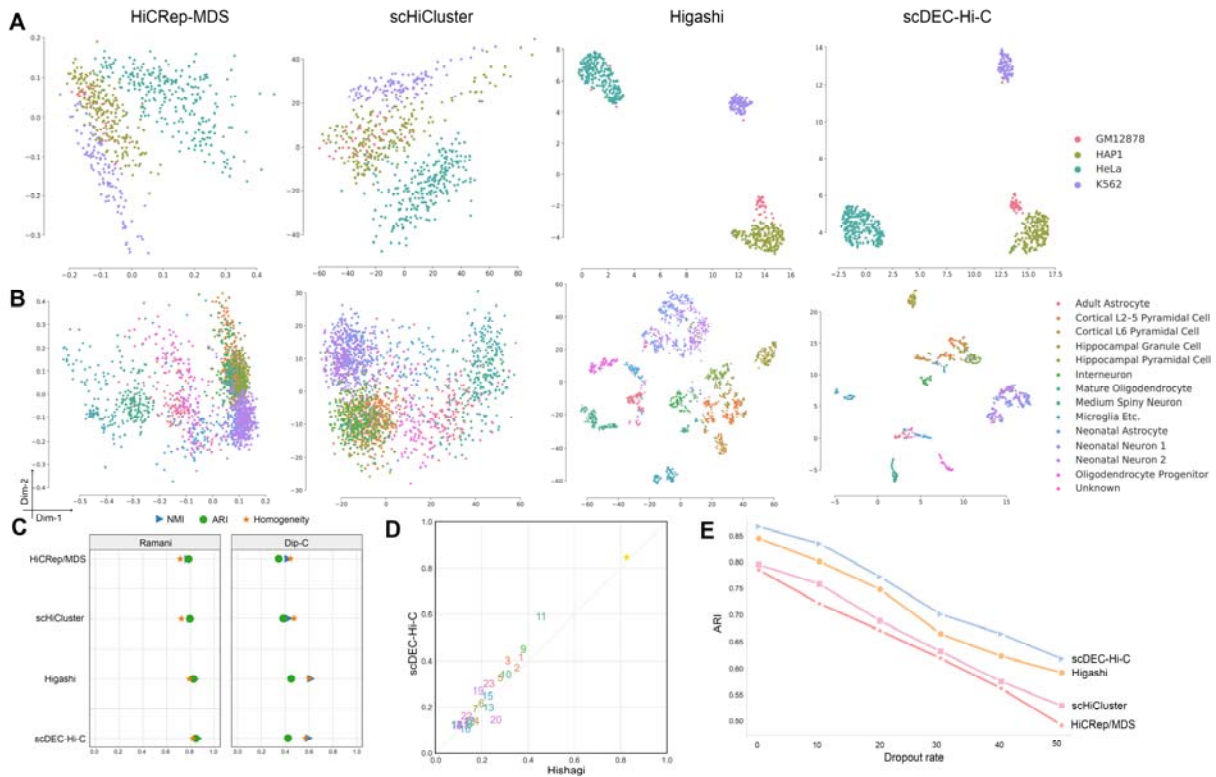


Figure2. The performance of scDEC-Hi-C method and baseline methods on single cell Hi-C datasets. (A) The embeddings visualization of Ramani dataset across four methods. (B) The embeddings visualization of Dip-C dataset across four methods. (C) The clustering performance in terms of NMI, ARI and Homogeneity of four methods across two datasets. (D) The performance of scDEC-Hi-C and baseline methods under different dropout rate on Ramani dataset.

486
487
488
489
490
491
492
493
494
495
496
497
498
499

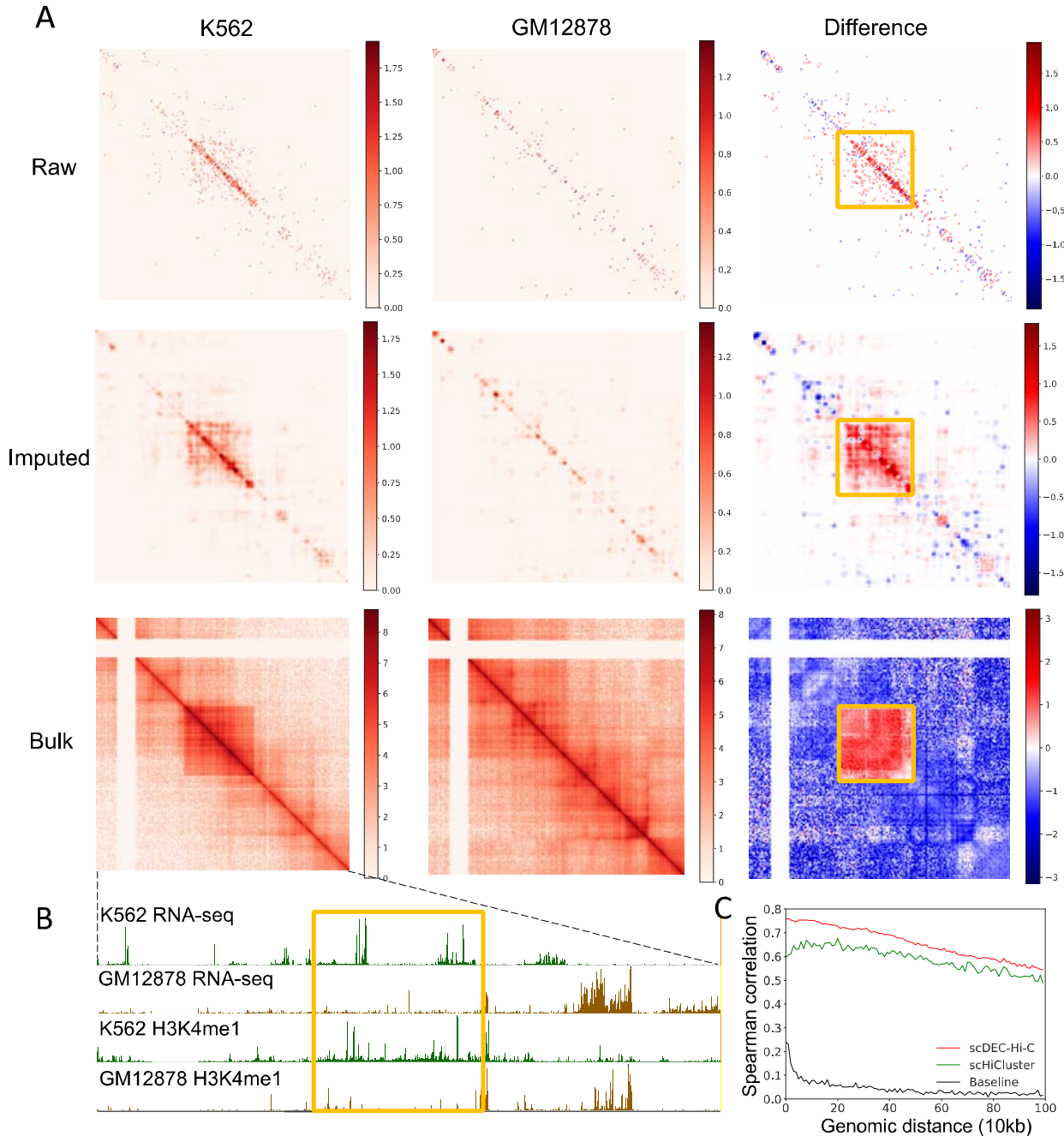


Figure 3. The imputation results of scDEC-Hi-C method. (A) The first row denotes merged single cell Hi-C profile of 40 cells of a genomic region (chr9: 132.9M-134.9M) across two diverse cell lines. The middle row denotes the corresponding imputed single cell Hi-C profile with scDEC-Hi-C. The third row denotes the corresponding bulk Hi-C profile of the two cell lines. The differences of the Hi-C profile from two cell lines are illustrated. (B) Genome annotation including RNA-seq and H3K4me1 histone marker across two cell lines of the same genomic region. (C) The Spearman correlation between bulk K562 Hi-C data and aggregated single cell Hi-C data after imputation by scDEC-Hi-C (red) and scHiCluster (green). The baseline (black) denotes the Spearman correlation between bulk K562 Hi-C data and aggregated single cell Hi-C data without imputation.

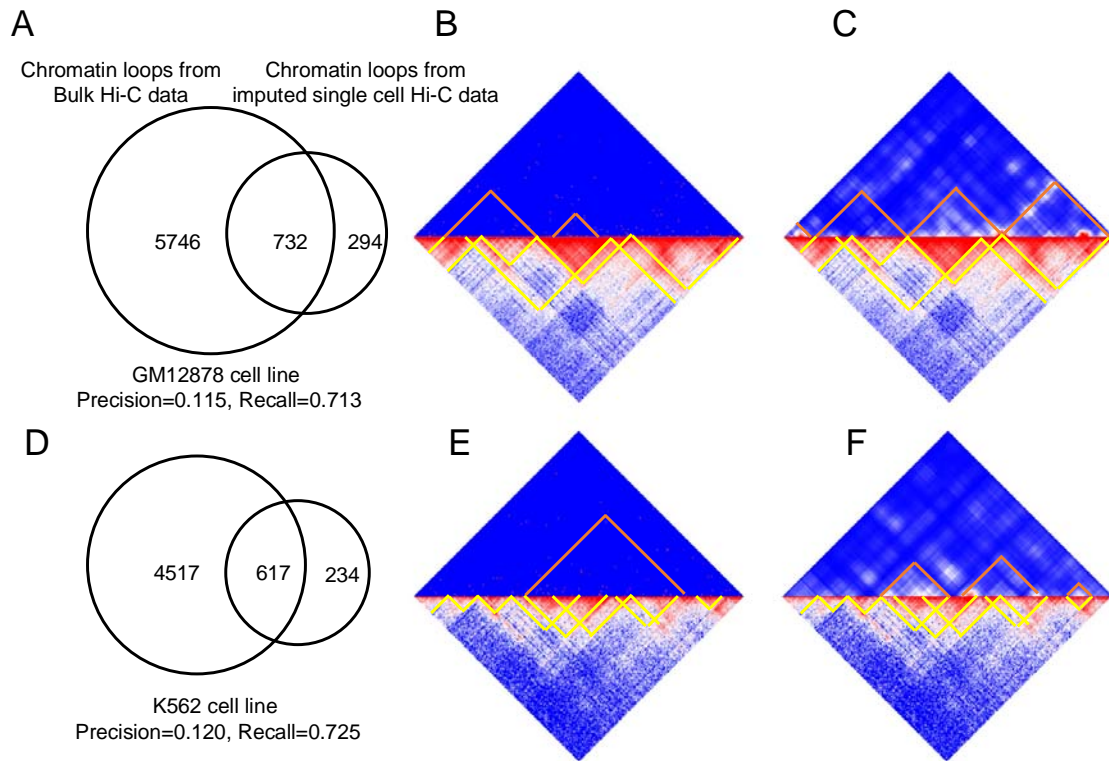


Figure 4. scDEC-Hi-C facilitates the identification of chromatin loops. (A) The Venn plot of chromatin loops from bulk Hi-C data and single-cell Hi-C data imputed by scDEC-Hi-C in GM12878 cell line. (B) The chromatin loops from raw single cell Hi-C data versus chromatin loops from bulk Hi-C data of a GM12878 cell line genomic region (chr3:118.2M-120.2M). (C) The chromatin loops from imputed single cell Hi-C data versus chromatin loops from bulk Hi-C data in GM12878 cell line of the same genomic region. (D) The Venn plot of chromatin loops from bulk Hi-C data and single-cell Hi-C data imputed by scDEC-Hi-C in K562 cell line. (E) The chromatin loops from raw single cell Hi-C data versus chromatin loops from bulk Hi-C data of a K562 cell line genomic region (chr3:118.2M-120.2M). (F) The chromatin loops from imputed single cell Hi-C data versus chromatin loops from bulk Hi-C data in GM12878 cell line of the same genomic region.