HiCImpute: A Bayesian Hierarchical Model for Identifying Structural Zeros and Enhancing Single Cell Hi-C Data.

Qing Xie[1], Chenggong Han[1], Victor Jin[2], Shili Lin[1,3,4,*]

1 Interdisciplinary Ph.D. Program in Biostatistics.
2 Department of Molecular Medicine, University of Texas Health Science Center, San Antonio, TX 78229.
3 Department of Statistics,
4 Translational Data Analytics Institute, The Ohio State University, Columbus, OH 43210.

*** Address for correspondence:** **Shili Lin, PhD**
Department of Statistics
The Ohio State University
1958 Neil Avenue
Columbus, OH 43210-1247, USA
Tel: (614) 292-7404
Fax: (614) 292-2096
Email: shili@stat.osu.edu

Abstract

Single cell Hi-C techniques enable one to study cell to cell variability in chromatin interactions. However, single cell Hi-C (scHi-C) data suffer severely from sparsity, that is, the existence of excess zeros due to insufficient sequencing depth. Complicate things further is the fact that not all zeros are created equal, as some are due to loci truly not interacting because of the underlying biological mechanism (structural zeros), whereas others are indeed due to insufficient sequencing depth (sampling zeros), especially for loci that interact infrequently. Differentiating between structural zeros and sampling zeros is important since correct inference would improve downstream analyses such as clustering and discovery of subtypes. Nevertheless, distinguishing between these two types of zeros has received little attention in the single cell Hi-C literature, where the issue of sparsity has been addressed mainly as a data quality improvement problem. To fill this gap, in this paper, we propose HiCImpute, a Bayesian hierarchy model that goes beyond data quality improvement by also identifying observed zeros that are in fact structural zeros. HiCImpute takes spatial dependencies of scHi-C 2D data structure into account while also borrowing information from similar single cells and bulk data, when such are available. Through an extensive set of analyses of synthetic and real data, we demonstrate the ability of HiCImpute for identifying structural zeros with high sensitivity, and for accurate imputation of dropout values in sampling zeros. Downstream analyses using data improved from HiCImpute yielded much more accurate clustering of cell types compared to using observed data or data improved by several comparison methods. Most significantly, HiCImpute-improved data has led to the identification of subtypes within each of the excitatory neuronal cells of L4 and L5 in the prefrontal cortex.

Introduction

Understanding three-dimensional (3D) chromosome structures and chromatin interactions is essential for interpreting functions of the genome because the spatial organization of a genome plays an important role in gene regulation and maintenance of genome stability [1]. Biochemical methods such as high-throughput chromosome conformation capture coupled with next generation sequencing technology (e.g., Hi-C) provide genome-wide maps of contact frequencies, a proxy for how often any given pair of loci interact in the cell nucleus, the natural 3D space where the chromosomes reside [2]. Bulk Hi-C is an averaged snapshot of millions of cells with limited information on heterogeneity or variability between individual cells. In contrast, single-cell Hi-C (scHi-C) data enable one to construct whole genome structures for single cells, ascertain cell-to-cell variability, and cluster single cells. Such studies can lead to understanding of cell-population compositions and heterogeneity, and has the potential to identify and characterize rare cell populations or cell subtypes in a heterogeneous population [3].

Sparsity is one of the major difficulties in analyzing single cell data, and it is even more challenging for scHi-C data, as sparsity is an order of magnitude more severe compared to most of other types of single-cell data [4]. Since Hi-C data are represented as two-dimensional (2D) contact matrices, the coverage of scHi-C ($0.25 - 1\%$) is much smaller than that of single cell RNA-seq (scRNA-seq, $5 - 10\%$) [4]. A further complication is that, among observed zeros in an scHi-C contact matrix, some are true zeros (i.e. structural zeros - SZs) because the corresponding pairs do not interact with each other at all due to the underlying biological function, whereas others are sampling zeros (i.e., dropouts - DOs) as a result of low sequencing depth. Telling SZs and DOs apart is important as it would improve downstream analysis such as clustering and 3D structure recapitulation. For example, methods for reconstructing 3D structures have included a penalty term to position two loci in the 3D space as far as possible if they do not interact [5, 6]. If there is not sufficient sequencing depth, especially in single cells, and if observed zeros are not correctly identified as SZs and DOs, then, applying such a penalty can lead to an artificial separation of two loci that in fact have coordinated effects on certain biological functions.

Currently, the concepts of SZs and DOs are well understood and have received considerable attention in scRNA-seq research, with a number of methods developed to identify SZs and impute DOs. Several of the methods, including MAGIC [7], SAVER [8], scUnif [9], scImpute [10], MCImpute [11], and DrImpute [12], were evaluated and compared in a recent publication [13]. In contrast, the concepts of SZs and DOs have not been widely pursued in scHi-C research. In fact, although the issue of sparsity has been addressed, albeit still quite limited, in the scHi-C or bulk Hi-C literature, the focus has been on improving data quality, and little has been said about distinguishing between SZs and DOs [14]. Nevertheless, the need for imputing the zeros have been emphasized in several papers, which is treated as a necessary intermediate steps in these papers to improve data quality for answering various biological questions, including assessing data reproducibility, enhancing data resolution, constructing 3D structure, and clustering of single cells [4, 15–18].

Existing approaches for addressing sparsity to improve data quality all aim to "smooth" the data by borrowing information from neighbors, and they may be classified into three categories depending on the methodology used: (1) kernel smoothing, (2) random walks, and (3) convolutional neural network, with representatives in all categories provided in Supplementary Table S1. For kernel smoothing, the types of kernels that have been used in the literature are uniform kernels or 2D Gaussian kernels [16]. For example, HiCRep [15], which aims to assess the reproducibility of Hi-C data, applies a uniform kernel (or referred to as 2D mean filters in that paper) by replacing each entry in the 2D contact matrix with the mean count of all contacts in a

neighborhood. Another method, scHiCluster [4], has proposed the use of a method in its first step that may also be classified into this category: it uses a filter that is equivalent to taking the average of the genomic neighbors, although the filter may also incorporate different weights during imputation. While a uniform kernel (2D mean filter) takes the average of the genomic neighbors with equal weights, a 2D Gaussian kernel uses a weighted average of neighboring counts according to a 2D Gaussian distribution: the farther away a neighbor is from the entry that is being imputed, the smaller the weight. For instance, SCL [16] applies a 2D Gaussian function to impute scHi-C contact matrices before inferring the 3D chromosome structure.

Method referred to as random walks have also been proposed as a way to smooth out an observed 2D matrix for improving data quality [4, 17, 19, 20]. The idea of a "random walk" process is to borrow information from neighbors in a fashion different from the "neighborhood" idea in kernel smoothing. Any position that is on the same row or column as the entry being imputed (but not necessarily has to be a neighbor) will contribute to the "smoothed" count in each step of the random walk. In GenomeDISCO [17], it is found that taking three steps of the random walk would lead to the best results in the problems investigated therein. Another way to improve data quality is through applying convolutional neural network, a deep learning method commonly applied to analyzing imaging data; HiCPlus [18] and DeepHiC [21] are such supervised learning techniques for improving data quality.

Taking on the challenging problems of separating the zeros into structural zeros and sampling zeros, imputing those that are dropouts, and improving data quality more generally, in this paper, we develop HiCImpute, a Bayesian hierarchical model for single cell Hi-C data that borrows information from three sources (if available): neighborhood of a position in the 2D matrix, similar single cells, and bulk data. Through an extensive set of analyses of synthetic and real data, we evaluated the ability of HiCImpute for identifying structural zeros, its accuracy for imputing dropout values, and compare the performance with three existing methods for data quality improvement. We further evaluate downstream analyses using data improved from HiCImpute and the other methods to evaluate the improvement for cell type clustering and subtype discovery.

Results

Overview of HiCImpute

The overall goal of HiCImpute, a Bayesian hierarchical model for analyzing single cell Hi-C data, is to identify structural zeros with high sensitivity and to impute dropout values for the sampling zeros with great accuracy (Figure 1). The main idea relies on the introduction of an indicator variable denoting structural zero or otherwise, for which a statistical inference is made based on its posterior probability estimated using Markov chain Monte Carlo (MCMC) samples (see Methods). We further include additional information through hierarchical modeling and prior specifications by borrowing information from several sources such as neighborhood, similar single cells, and bulk data. A number of criteria for evaluating the performances of HiCImpute have been devised (See Methods). Briefly, the criteria include the proportion of true structural zeros (PTSZ) correctly identified to ascertain the power (sensitivity) for detecting true structural zeros, the proportion of true dropouts (PTDO) identified to gauge the ability for correct identification of dropout events (specificity), correlation (CIEZ and CIEA) and absolute errors (AEOA and AEOZ) for comparing between imputed values and underlying true values, and graphical tools (heatmap, ROC and AUC, SEVI and SOVI) for visualization of imputation accuracy. As part of the workflow, visualization of clustering and subtype discovery results will also be provided via t-SNP and K-means.

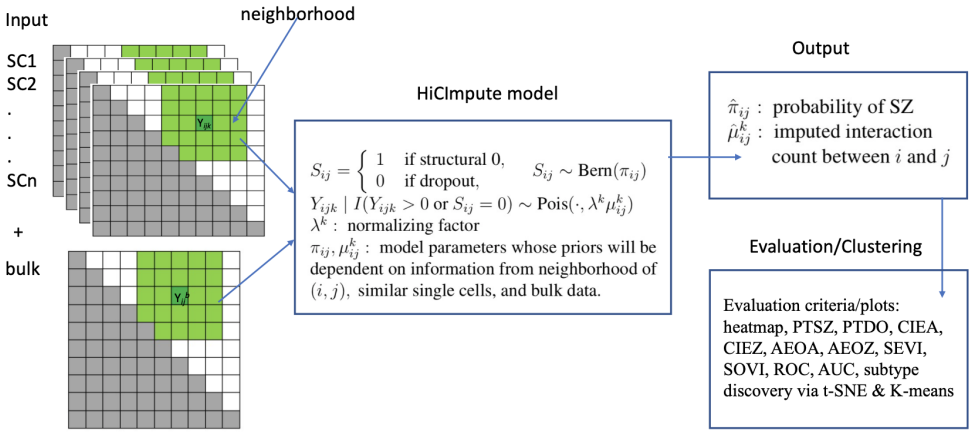


Figure 1. Schematic of the HiCImpute algorithm. Each green region on the left denotes the neighborhood. The indicator variable S_{ij} denotes whether an observed zero at the (i, j) position is a structural zero or not. The λ^k is related to the sequencing depth of single cell k and acts as a normalizing factor. Finally, the intensity parameter μ_{ij}^k is assumed to follow a common distribution across all similar single cells; shrinkage estimation with information from neighborhoods and bulk data will be obtained, which provides accommodations for potential overdispersion. PTSZ (proportion of true structural zeros correctly identified), PTDO (proportion of true dropouts correctly identified), AEOA (absolute error of all observed), AEOZ (absolute error of observed zeros), CIEA (correlation between imputed and expected for all observed), CIEZ (correlation between imputed and expected for observed zeros), SEVI (scatterplot of expected versus imputed), and SOVI (scatterplot of observed versus imputed), ROC (receiver operational characteristics), AUC (area under the curve).

HiCImpute greatly improves data quality

A major goal of imputing scHi-C data is to improve data quality for downstream analyses, including determination of cell identify, clustering, and subtype discovery [14–17]. In addition to HiCImpute, three existing methods that have been used to improve Hi-C data quality are also considered, so that their performance can be investigated and compared to HiCImpute: 2D mean filter (2DMF) in HiCRep [15], 2D Gaussian kernel (2DGK) in SCL [16], and random walk with 3 steps (RW3S) in GenomeDISCO [17]. These three particular methods were selected for comparison because of their well-characterized and known features in the statistics literature (2DMF and 2DGK) or because of their frequent use in this particular type of applications (RW3S).

We first simulated three "types" (T1, T2, T3) of single cells Hi-C data modeled after three K562 single cells data publicly available [22]. In addition to considering three cell types, a number of other parameters are also considered for a thorough investigation, including sequencing depth (7K, 4K, 2K) and the number of cells (10, 50, 100). Details of the simulation procedure is described in Methods. We first use heatmaps to visualize a 2D data matrix before and after the data quality improvement for each of the methods considered.It is clearly seen that for a T1 single cell at the 4K sequencing depth, HiCImpute was able to denoise and recover the underlying structure well (Figure 2a). On the other hand, whereas 2DMF and 2DGK oversmoothed the image (the main domain structures are still visible, though), RW3S completely lost the domain structure. The superior performance of HiCImpute can also be seen from the scatterplots of the expected versus the imputed (SEVI plots), where the imputed values are highly correlated with the expected, as the point cloud is distributed tightly around a straight line, including the observed zeros (Figure 2b). On the other hand, all three of the comparison methods have point clouds that follow a funnel shape, indicating much greater variability for larger counts; that is, the imputation becomes less accurate for larger counts. The shrinkage effect is expected (i.e. the imputed values are smaller than the expected counts due to smoothing), although the effect is much more pronounced with the comparison methods than with HiCImpute. Considering the aggregate performance for all single cells, we see that HiCImpute achieves better correlation between the imputed and expected counts, either for all observed values (CIEA) or only the observed zeros (CIEZ) compared to the other methods (Figure 2c). The absolute error for the observed zeros (AEOZ) or for all observed (AEOA) are much smaller compared to the other methods. The above observation for cell type T1 with sequencing depth at 4K holds to a large extend across cell types and number of cells (Supplementary Figures S1-Figures S4), although absolute errors for HiCImpute can be slightly larger than the comparison methods for setting with (low) sequencing depth, at 2K.

HiCImpute is highly sensitive for identifying structural zeros

A novel concept being explored in this paper for scHi-C is structural zeros and our ability to separate them from sampling zeros. The results discussed thus far (Figure 2b,c) provides some indirect assessment of the capability of HiCImpute; we now further provide direct evaluation and comparison with other methods. The results using the Bayes rule (Methods) show that HiCImpute has an extremely high sensitivity for detecting SZs. In fact, using the criterion of the proportion of true structural zeros (PTSZ) detected (i.e. the proportion of true underlying structural zeros being correctly declared as SZs – sensitivity), HiCImpute reaches the proportion of greater than 0.95 for all the situations considered (Figure 2d and Suplementary Figure S3). For the three comparison methods, an observed zero is identified as SZ if its imputed value is less

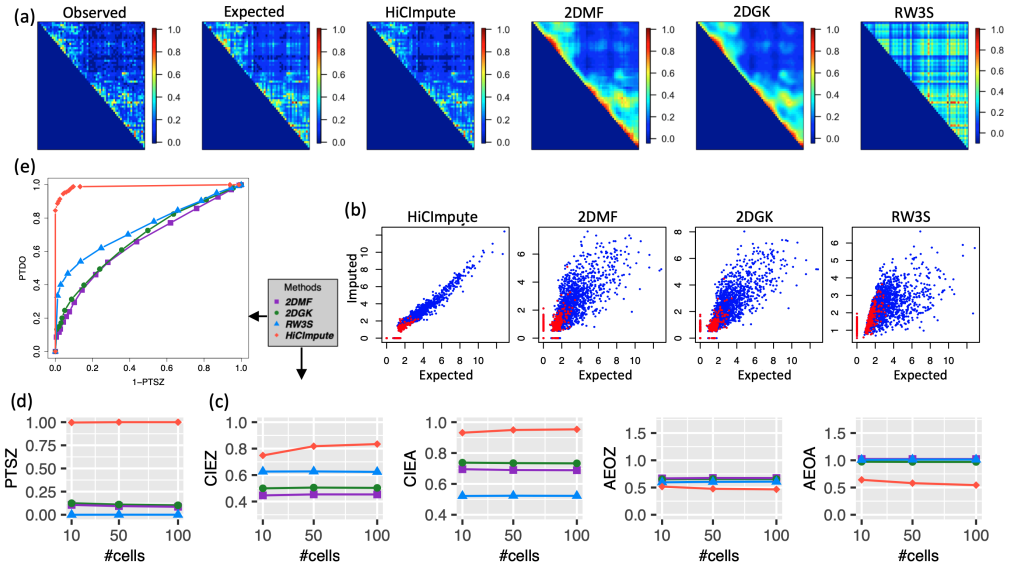


Figure 2. Comparison of results from HiCImpute for data quality improvement with 2DMF, 2DGK, and RW3S for T1 cells at 4K sequencing depth. Ordering of the subfigures is clockwise. (a) Heatmaps of the first single cell showing the observed and true (expected) 2D matrix images as well as the results from HiCImpute and three comparison methods; (b) Scatterplots of Extected Versus Imputed (SEVI plots) for HiCImpute and the comparison methods – the red dots represent the observed zeros, which contain both true SZs (expected = 0) and DOs; (c) aggregate results (over single cells) based on several evaluation criteria; (d) Proportion of true SZs correctly detection averaged over single cells; (e) ROC curves accounting for both PTSZ and PTDO (with AUC = 0.98 compared to 0.66, 0.68, and 0.74 for 2DMF, 2DGK, and RW3S, respectively).

than 0.5. This criterion, borrowed from the existing literature on scRNA-seq [7, 11], led to subpar performances: for T1 4K, less than 0.25 of the true SZs were detected. Although the PTSZ may reach over 0.75 when the sequencing depth is low (e.g. T2 2K), the value is typically low, at about 0.25 or less for most of the settings considered.

Since the three comparison methods only aim for data quality improvement, not for identifying structural zeros, their adaptation for this purpose with the threshold of 0.5 may be viewed as arbitrary. Therefore, we explore a range of threshold values and plot the performance as ROC curves (Figure 2e and Supplementary Figure S4). Once again, HiCImpute outperforms the other methods for this evaluation criterion. HiCImpute not only has larger sensitivity for detecting SZs, but also large r specificity for detecting DOs, with a much larger area under the curve (AUC). For HiCImpute, the AUC is 0.98 compared to 0.66, 0.68, and 0.74 for 2DMF, 2DGK, and RW3S, respectively.

HiCImpute identifies DOs and imputes them with high accuracy

Fixing the PTSZ at 0.95, we further examined and compared the performances of the methods. The reason that we chose to fix the threshold at this level is akin to controlling for the type II error at 0.05. Since the ability to identify SZs is critical for downstream analyses such as constructing 3D structures (as a penalty may be imposed based on SZs [23–25]), it is desirable to keep the proportion of failure to correctly identify the underlying structural zeros at a low level (e.g. 0.05). One can see from Table 1 that HiCImpute outperforms the other methods for correctly identifing the true

dropouts by a large margin across all three single cell types, sequencing depth, and sample sizes. For example, for T1 4K, the specificity, PTDO, for HiCImpute is at 95%; in contrast, even among the best of the three methods, RW3S, at most only 44% of the dropouts are correctly identified. In general, the specificity for HiCImpute is more than doubling that for a comparison method when the specificity for the method is below 50%. The accuracy of the imputed values for the DOs and the far superior performance of HiCImpute over the three smoothing methods are consistent with the plots discussed earlier (Figure 2e, Supplementary Figure S4).

Table 1. Mean (standard error) of the proportion of true dropouts (PTDO) correctly detected when the detection rate for the proportion of true structural zeros (PTSZ) is set to be 0.95.

Type	Sequence depth	#cells	HiCImpute	2DMF	2DGK	RW3S
T1	7k	10	0.98 (0.01)	0.29 (0.04)	0.31 (0.04)	0.50 (0.06)
		50	0.99 (0.01)	0.27 (0.05)	0.31 (0.05)	0.47 (0.07)
		100	0.99 (0.01)	0.27 (0.04)	0.30 (0.05)	0.46 (0.06)
	4k	10	0.95 (0.01)	0.21 (0.03)	0.24 (0.03)	0.43 (0.03)
		50	0.95 (0.01)	0.18 (0.03)	0.25 (0.03)	0.44 (0.03)
		100	0.95 (0.01)	0.19 (0.03)	0.26 (0.03)	0.44 (0.03)
	2k	10	0.95 (0.00)	0.39 (0.02)	0.45 (0.02)	0.55 (0.02)
		50	0.98 (0.00)	0.39 (0.02)	0.45 (0.02)	0.56 (0.03)
		100	0.98 (0.00)	0.39 (0.02)	0.44 (0.02)	0.56 (0.03)
T2	7k	10	0.60 (0.03)	0.08 (0.03)	0.10 (0.04)	0.26 (0.06)
		50	0.64 (0.04)	0.10 (0.04)	0.11 (0.04)	0.25 (0.05)
		100	0.63 (0.04)	0.10 (0.03)	0.11 (0.03)	0.26 (0.05)
	4k	10	0.89 (0.01)	0.30 (0.02)	0.34 (0.02)	0.63 (0.03)
		50	0.88 (0.01)	0.29 (0.02)	0.33 (0.02)	0.62 (0.03)
		100	0.88 (0.01)	0.29 (0.02)	0.33 (0.02)	0.62 (0.03)
	2k	10	0.93 (0.00)	0.39 (0.03)	0.43 (0.03)	0.76 (0.03)
		50	0.95 (0.00)	0.46 (0.02)	0.43 (0.02)	0.76 (0.02)
		100	0.96 (0.00)	0.39 (0.02)	0.43 (0.02)	0.76 (0.02)
T3	7k	10	0.67 (0.02)	0.07 (0.02)	0.08 (0.02)	0.32 (0.04)
		50	0.66 (0.03)	0.07 (0.02)	0.08 (0.02)	0.33 (0.05)
		100	0.67 (0.03)	0.06 (0.02)	0.08 (0.02)	0.32 (0.05)
	4k	10	0.91 (0.01)	0.09 (0.01)	0.10 (0.01)	0.54 (0.05)
		50	0.89 (0.01)	0.10 (0.01)	0.12 (0.01)	0.53 (0.03)
		100	0.89 (0.01)	0.09 (0.01)	0.12 (0.02)	0.54 (0.03)
	2k	10	0.96 (0.00)	0.18 (0.02)	0.19 (0.02)	0.56 (0.03)
		50	0.96 (0.00)	0.15 (0.01)	0.19 (0.01)	0.56 (0.03)
		100	0.95 (0.00)	0.18 (0.01)	0.19 (0.01)	0.56 (0.03)

Improved data lead to more accurate clustering of cells

We consider three real scHi-C datasets to demonstrate the improvement of cell type clustering after data improvement with HiCImpute and compare with the results using data improved by the three comparison methods: 2DMF, 2DGK, and WR3S.

The first scHi-C dataset (https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE117874) consists of 14 GM (lymphoblastoid) and 18 PBMC (peripheral blood mononuclear cells) [26]. Based on a sub-2D matrix of dimension 30×30 on chromosome 1 of the 32 SCs of the observed Hi-C data and using the K-means algorithm, there was one misclassification for the GM and 7 for PBMC (Table 2a). With the imputed data from scHiCBayes and the same sub-2D matrix, all GM cells were correctly classified, and there were only three misclassified PBMC cells, a fairly large improvement. On the other hand, using imputed data by 2DMF and 2DGK do not see any improvement, whereas the WR3S imputed data in fact led to more misclassifications on the GM and PBMC cells than using the observed data. The scatterplot of observed versus imputed (SOVI plot) shows that the imputed data from HiCImpute are highly correlated with the observed, whereas the other methods see widely scattered point clouds (Supplementary Figure S5) The correlation between the observed and the imouted are also seen to be much higher across all cells (Supplementary Figure S6).

The second Hi-C dataset (https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE80006) consists of two bulk K562 Hi-C data — one K562A (bulk A) and one K562B (bulk B) — and 19 scHi-C data of K562A and 15 K562B cells [22]. However, among the 34 single cells, only 10 has sequencing depth over 5K; for the remaining ones, most only have sequencing depth of 1K. Using hierarchical clustering, one can see that K562A and K562B cells are mixed together, and in fact, the group in the middle consists of the 10 cells that have sequencing depth of at least 5000, together with the two bulk data (Supplementary Figure S7). Considering only these 10 singles cells and clustering them using K-means based on the observed data led to one of the two K562A cells clustered with the eight K562B cells (Table 2b). On the other hand, clustering using improved data from HiCImpute corrected the misclassification, resulting in perfect separation of the K562A and K562B cells. In contrast, using data improved by 2DMF, 2DGK, or RW3S did not yield any improvement over the outcome from simply using the observed data. SOVI plots and correlations between observed and imputed further substantiate the superior performance of HiCImpute (Supplementary Figures S5 and S6).

Table 2. Clustering results for three single cell Hi-C datasets before and after data improved with four methods.

Dataset	type	K-means	Observed	HiCImpute	2DMF	2DGK	RW3S
(a) GSE117874	GM	C1	13	14	13	13	11
		C2	1	0	1	1	3
	PBMC	C1	7	3	7	7	8
		C2	11	15	11	11	10
(b) GSE80006	K562A	C1	1	2	1	1	1
		C2	1	0	1	1	1
	K562B	C1	0	0	0	0	0
		C2	8	8	8	8	8
(c) scm3C-seq	L4	C1	76	131	77	77	76
		C2	55	0	54	54	55
	L5	C1	105	0	105	104	105
		C2	75	180	75	76	75

The third scHi-C dataset (https://github.com/dixonla b/scm3C-seq) consists of prefrontal cortex cells of subtypes L4 (131 cells) and L5 (180 cells) [27]. It is known that there are 14 cell subtypes of the prefrontal cortex cells, including eight neuronal subtypes that were all clustered together based on the observed scHi-C data [27]. Among them are L4 and L5, two excitatory neuronal subtypes known to be located on different cortical layers. Our K-means analysis based on the observed L4 and L5 scHi-C data shows that these two subtypes are indeed mixed together (Table 2c), echoing the earlier finding [27]. Although the problem is much more challenging compared to the first two datasets given its size and the extremely mixed clustering results based on the observed data, using data improved by HiCImpute led to perfect separation of the two subtypes; whereas none of the data improved using the comparison methods yielded any improvement. SOVI plots of the observed versus the imputed values and the correlations across all cells painted the same picture as for the other two datasets on the superiority of scHi-C over the other methods (Supplementary Figures S5).

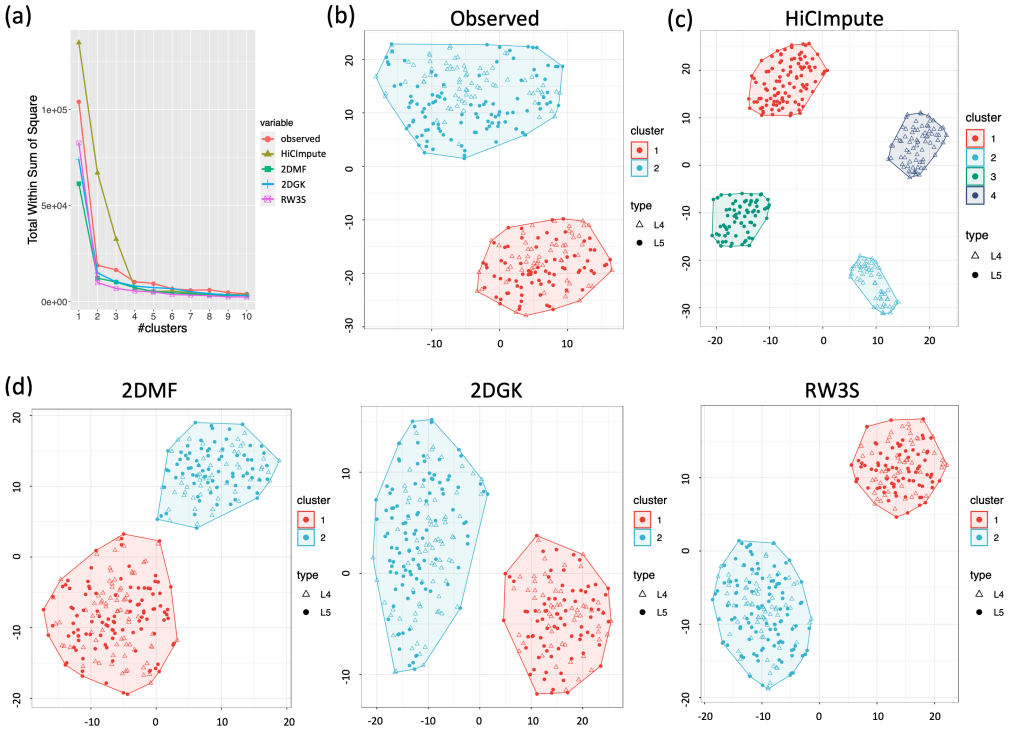


Figure 3. Comparison of results from HiCImpute, 2DMF, 2DGK, and RW3S via t-SNE visualization and clustering with K-means. (a) Plots of total within-cluster sum of squares versus number of clusters for K-means analysis; (b) t-SNE visualiation and K-means clustering boundaries based on observed data; (c) Same as (b) but based on HiCImpute-improved data; (d) Same as (b) but based on 2DMF, 2DGK, or RW3S-improved data.

Discovery of subtypes of L4 and L5

Cell to cell variability is a driving force behind the developments of single cell technologies [28]. Based on single cell RNA-seq data, subtypes of L4 and L5 have been discovered. For example, two L4 subtypes, Exc L4-5 FEZF2 SCN4B and Exc L4-6 FEZF2 IL26, were found to be highly distinctive as they occupied separate branches of a dendrogram [29]. On the other hand, the L4–IT–VISp–Rspo1 cells were shown to exhibit heterogeneity along the first principal component of scRNA-seq data [30]. Similarly, two subtypes of L5, Exc L5-6 THEMIS C1QL3 and L5-6 THEMIS DCSTAMP, were also found to be on two separate branches of a dendrogram [29], while there was also research that further classified L5 cells into L5a and L5b subtypes [30]. Other works have also found subclusters of excitatory neurons including L4 and L5 [31–33].

Inspired by the ample evidence in the literature that subtypes of L4 and L5 exist, we visualized the observed data and those improved by HiCImpute, 2DMF, 2DGK, and RW3S using t-SNE and then clustered using K-means. Based on the within-cluster sum of squares and visually inspecting the number of clusters where the "elbow" is identified (Figure 3a), we see that there are two clusters for the observed data (Figure 3b) and those improved with 2DMF, 2DGK, or RW3S (Figure 3d). On the other hand, for the data improved with HiCImpute, the plot clearly shows the existence of four clusters (Figure 3c). In fact, these four clusters are very well separated, with two of them consisting of purely L4 cells and two L5 cells. Using the adjusted rand index (ARI) [34], we further investigate the optimal number of clusters and the performance of clustering for the observed data and improved data with HiCImpute and the other methods. Based on the results (Supplementary Table S2), it is without a doubt that HiCImpute improves over the observed data and outperform the other methods. Most importantly, using data improved with HiCImpute, two subtypes each for L4 and L5 emerge, consistent with results in the literature. On the other hand, none of the data improved with the other methods led to the discovery of any subtypes for L4 or L5.

Visualization by a 2-way clustering heatmap using normalized and log-transformed HiCImpute-improved data for the 500 positions (on the 2D matrix) with the highest variation across all cells further substantiates the 3D structural differences between each of the two subtypes of L4 and L5 (Figure 4a), where the L4 cells were clustered into two subgroups, and the same for the L5 cells. Several genes that were found to be differentially expressed among subgroups of L4 and L5 in the literature [29] were also marked on the heatmap, where it can be seen that there are differential interaction intensities among the subtypes. To further elucidate the potential correspondence between differential gene expression and differential 3D interaction intensities among the subtypes of L4 or among those of L5, we combined all cells from each of the subtypes into four mega 2D matrices (L4T1, L4T2, L5T1, L5T2) and normalized them to the same total count and scaled them to be a value between 0 and 1. These 2D matrices displayed as heatmaps exhibit regions having differential interaction intensities. Zooming in on the region chr20:35,000,000-55,000,000, we can see that L5T1 has relatively lower intensities compared to its L5T2 counterparts, and furthermore, the latter appears to have some subtle domain structures that are missing in the former (Figure 4b). Interestingly, when we reproduce the mean RNAseq data in the same region for two subtypes discussed in the literature [29] as tracks in the UCSC genome browser (https://human-mtg-rna-hub.s3-us-west-2.amazonaws.com/HumanMTGRNAHub.html), the differences in the gene expression patterns are obviously (Figure 4c). Examples of other regions where there appear to be a correspondence between 3D structure differences and gene expression differences among subtypes of L4 or L5 are also provided (Supplementary Figure S8-Figure S10).

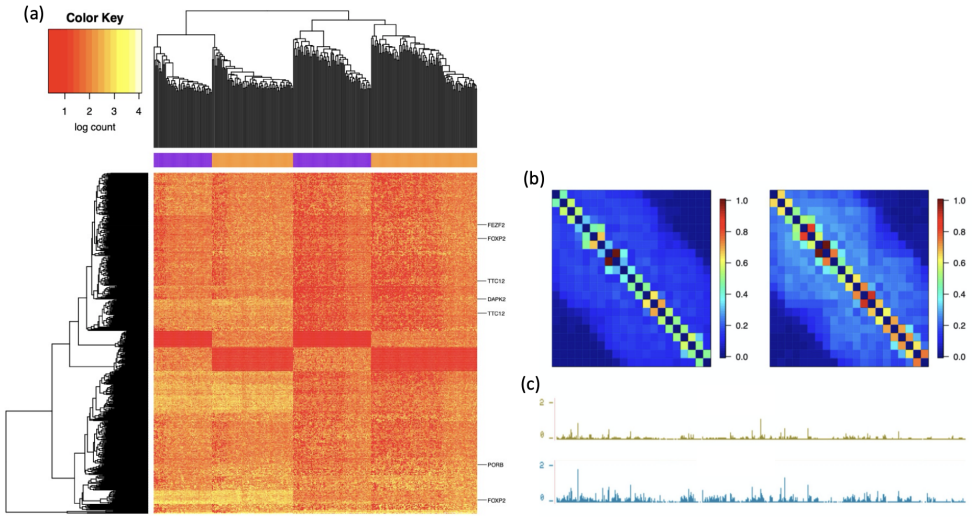


Figure 4. Correspondence between differential 3D structures and differential gene expressions among further subtypes. (a) Heatmap of 500 positions in the 2D interaction matrix with the largest variation among the cells (each row is a position and each column is a cell), with a 2-way clustering outcome placing the L4 cells into the two purple groups and the L5 cells into the two orange groups, and genes showing differential expression [29]indicated on the right edge of the heatmap; (b) Mega 2D matrices of normalized and scaled interaction intensities displayed as heatmaps, with the left for L5T1 and right for L5T2; (c) Mean gene expression for two L5 subtypes described in the literature [29].

Discussion

This paper introduces the concept of structural zeros in the context of 3D contacts, and explores the ability of HiCImpute for separating structural zeros from sampling zeros and the accuracy of imputing the dropouts. From both simulation and real data studies, we can see that HiCImpute has great ability of identifying structural zeros, and outperforms existing methods for its accuracy of imputing the contact counts of dropouts based on multiple criteria. This conclusion is based on outcomes from considering a number of factors, including the number of cells, sequencing depth, multiple cell types, and whether bulk data are available. The improved data from HiCImpute has greatly impacted downstream analysis. From the examples of clustering GM and PBMC cells, K562 cells, and prefrontal cortex cells, we have seen that data improved with HiCImpute led to more accurate clustering judging from known cell types. What is most exciting is the ability of HiCImpute for producing improved data that can lead to not only the separation of L4 and L5 of the prefrontal cortex cells, but also the discovery of two subtypes, each within L4 and L5, for the first time using scHi-C data. Given that the existence of further subtypes within each of these two excitatory neuronal subtypes has been documented in the literature using scRNA-seq data [29–33, 35], and given that our own analysis has found regions where there are differential expression and differential interactions between further subtypes, our results may be viewed as evidence for the potential of establishing the correspondence between scHi-C and scRNA and elucidating the ability of scHi-C data, when appropriately enhanced, for investigating cell-to-cell variability and uncovering hidden subpopulations and substructures.

The more accurate results do not come without a greater cost in computational time,

though. Our scHi-C method is implemented in C++ for computational efficiency since the algorithm based on Markov chain Monte Carlo is computationally intensive. The computational time for HiCImpute was in hours with hundred of cells for the L4/L5 prefrontal cortex data, compared to minutes with the other methods (Supplementary Table S3). Nevertheless, considering the time needed for collecting the samples and generating the data, this price to pay is completely justifiable, especially since biological insights are gained with the improved data from HiCImpute compared to the alternatives in the literature. Hours of computational time for the "truth" to be revealed is certainly worth the wait and the cost compared to the "truth" continued to be hidden. Nevertheless, effort will continue to be made to further improve the computational efficiency.

Methods

Bayesian Hierarchy Model

Suppose we have contact matrices for K Single Cells (SCs) and a bulk Hi-C dataset that is related to these SCs. Let Y_{ijk} and Y^b_{ij} represent the observed interaction frequencies between loci i and j $(i < j)$ for SC $k, k = 1, \cdots, K$, and the bulk data, respectively. Among those observed 0's, some are true 0s (i.e. structural zeros, SZs) since the two loci never interact with each other in this particular cell; whereas others are sampling zeros (i,e. dropouts, DOs) since they interact infrequently and thus dropout from the sample as their interaction is not observed due to insufficient sequencing depth. This zero-inflated problem is complicated since not all zeros are created equal, and our goal is to make statistical inferences to tease out those that are SZs from those that are DOs, and to imopute the values for the DOs.

Since Y_{ijk} is a count, its distribution can be reasonably modeled by a Poisson distribution, with additional hierarchical modeling to address potential overdispersion, leading to equivalency with a negative binomial model. Let $T_k = \sum_{i<j} Y_{ijk}$ denote the sequencing depth of SC k, and let μ^k_{ij} be the parameter representing the intensity of SC k if the SC is depth-normalized to a desired sequencing depth T, which may be the maximum sequencing depth among the SCs, that is, $T = \max\{T_k, k = 1, \cdots, K\}$, or may simply be an intended sequencing-depth level appropriate for downstream analysis, say 300,000, the level of the best K562 scHi-C data [22] . Then $\lambda^k = T_k/T$ is the proportionate sequencing depth of SC k relative to the intended one.

To distinguish the SZs from the DOs, we define an indicator variable S_{ij}, which equals to 1 if loci i and j do not interact, otherwise it is 0. That is, $S_{ij} \sim \text{Bernoulli}(\pi_{ij})$, where π_{ij} is the probability that pair i and j do not interact. Y_{ijk} therefore follows a mixture of a point-mass distribution at 0 and a Poisson distribution with mixing proportions π_{ij} and $1 - \pi_{ij}$, respectively. Hence,
$Y_{ijk} \mid I(Y_{ijk} > 0 \text{ or } S_{ij} = 0) \sim \text{Poisson}(\lambda^k \mu^k_{ij})$, where $I(\cdot)$ is the usual indicator function, and $\lambda^k \mu^k_{ij}$ is the intensity parameter for the non-normalized observed counts. We further let π_{ij} follow a Beta distribution and its mean is governed by the observed proportion of zeros across the SCs in that position. The idea is that if there is a large proportion of zeros at that position, it is more likely to be an SZ.

We allow for cell-to-cell variability by setting up an additional hierarchy to model μ^k_{ij} as follows: $\mu^k_{ij} \sim \text{Normal}^+(\mu_{ij}, \sigma^2_{ij})$, where Normal^+ is a truncated normal distribution on positive numbers, σ^2_{ij} is taken to be the standard deviation of nonzero counts in a neighborhood centered at (i, j), and μ_{ij} is further assumed to follow a Gamma distribution whose mean borrows information from both the bulk Hi-C and the neighborhood data across similar SCs. Specifically, let $Y^{(nSC)}_{ij} = \sum_k Y_{ijk} / \sum_k \lambda_k$, which is the weighted average of the "normalized" (to sequencing depth T) contacts

between i and j over the SCs with the weight proportional to the sequencing depth of each SC. Similarly, we let $T^b = \sum_{i<j} Y_{ij}^b$ and $Y_{ij}^{(nB)} = TY_{ij}^b/T^b$ be the sequencing depth of the bulk data and the count of the bulk data "normalized" to sequencing depth T, respectively. Then the mean of the Gamma distribution is set to be $\left(\sum_{(i,j)\in\Omega_2} Y_{ij}^{(nB)}/||\Omega_2||\right)\left(\sum_{(i,j)\in\Omega_1} Y_{ij}^{(nSC)}/(||\Omega_1||\bar{Y}^{(nSC)})\right)$, where Ω_1, Ω_2 are the neighborhoods for the SCs and the bulk data, respectively, $||\cdot||$ is the cardinality of the neighborhood and $\bar{Y}^{(nSC)}$ is the average of the $Y_{ij}^{(nSC)}$ over the SCs. Under this setting, we note that information from the SCs plays a modifying role by providing a weight factor to the information from the bulk data: if the average count in the neighborhood of the SCs is larger than the average count over the entire matrix, then the mean neighborhood count of the bulk data will be boosted otherwise it will be shrunk. Throughout all the data analysis, the neighborhood is taken to be the two immediate neighbors (if available) in all directions of a lattice (Figure 1).

Details on the prior specifications, the posterior distributions, Markov chain Monte Carlo (MCMC) sampling schemes, and convergence diagnostics are provided in the Supplementary Materials. Using samples generated by MCMC from the posterior distribution of π for a particular pair that have an observed zero count in an SC, we can make inference about whether the zero is a SZ or a DO. A natural decision based on the Bayes rule is to declare a zero for an SC to be a SZ if the corresponding π is estimated by the posterior sample mean to be greater than 0.5. However, to compare between HiCImpute and existing methods, as described in more details in the evaluation criteria below, we also set different thresholds to obtain an ROC curve.

For comparison with 2DMF, 2DGK, and RW3S in terms of PTSZ, we follow the recommendation in the scRNA-seq literature by labelling an observed zero to be a structural zero if the imputed count is less than 0.5 for each of the comparison methods [11]. We also vary the threshold to obtain an ROC curve separately for each of the three methods.

Performance evaluation criteria

To evaluate the performance of HiCImpute and to compare with other data quality improvement methods, including 2DMF, 2DGK, and RW3S, we consider the following novel criteria in addition to standard measures and plots, including the heatmap and t-SNE visualization tools [36], receiver operating characteristic (ROC) curves and area under the curve (AUC), K-means clustering algorithm, and the adjusted rand index for evaluating clustering results.

- PTSZ: Proportion of true structural zeros correctly identified. This is defined as the proportion of underlying structural zeros that are correctly identified as such by a method. Being able to separate structural zeros from sampling zeros is important for downstream analyses, especially for single cell classification to reveal cell sub-populations.
- PTDO: Proportion of true dropouts correctly identified. This is defined as the proportion of underlying sampling zeros (due to insufficient sequencing depth) that are correctly identified as such by a method. Similarly, being able to correctly identify dropouts is also critical for a number of downstream analyses.
- SEVI: Scatterplot of expected versus imputed. This serves as a visualization tool to directly assess whether dropouts are correctly recovered and accurately imputed for simulated data where the ground truth is known.
- SOVI: Scatterplot of observed versus imputed – applicable to real data for non-zero observed counts. This serves as a visualization tool to *indirectly* assess

whether the imputed values are sensible for the observed zeros by looking at the performance for observed non-zeros. For real data, whether an observed zero is a SZ or DO is unknown, and if it is a DO, the underlying expected non-zero value is also unknown. Nevertheless, the imputed values for the non-zero observed counts should not deviate wildly from the observed values even though some level of "smoothing" is applied.

- AEOA: Absolute errors for all observed data. This is defined as the absolute difference between the imputed and the expected for all observed data. This measure is to gage how well the imputed values can approximate its underlying true values.

- AEOZ: Absolute errors for observed zeros. Unlike AEOA that considers all observed, this measure only considers observed zeros. This measure provides a more focused evaluation on correct identification of structural zeros and the accuracy of the imputing dropout values.

Simulation studies and settings

To evaluate HiCImpute and compared with the three data quality improvement methods in the literature, we carried out an extensive simulation study for a total of over one hundred settings, including three types of single cells (T1, T2, T3, mimicking three K562 cells [22]), three sequencing depth in a 61×61 contact matrix on a segment of chromosome 19 (7k, 4k, and 2k), 3 sample sizes (10, 50, 100, representing the number of single cells), and 4 settings of SZs and DOs. The following describes the detailed simulation procedure to generate single cell data for each of the settings as well as bulk data.

- Step 1. Calculate the 3D distance matrix d where $d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$ at each pair of loci (i, j), $i < j$, where (x_i, y_i, z_i) represents the 3D coordinates for locus i of the 3D structure. For each of the three cells, its 3D structure was constructed using SIMBA3D [25] based on a K562 scHi-C data [22].

- Step 2. Use the following formula to generate the λ matrix following the literature [37]:

$$\log(\lambda_{ij}) = \alpha_0 + \alpha_1 \log d_{ij} + \beta_l \log(x_{l,i} x_{l,j}) + \beta_g \log(x_{g,i} x_{g,j}) + \beta_m log(x_{m,i} x_{m,j}),$$

  where α_1 is set to -1 to follow the typical biophysical model, α_0 is the scale parameter, and set to be 5.7, 6.3, and 6.8 for the three cell types, respectively. On the other hand, $x_{l,i}$, $x_{g,i}$, and $x_{m,i}$ are covariates generated from uniform distributions to mimic fragment length, GC content, and mappability score, respectively, and their coefficients, the β's, are all set to be 0.9.

- Step 3. Find the lower $\gamma\%$ quantile of the λ_{ij} as the threshold, for those $\lambda_{ij} <$ threshold, randomly select half of them to be candidates for structural zeros. Among these candidates, randomly select $\eta\%$ of them and set their new λ_{ij} value to be zero. These are the SZs across all SCs. In our simulation, we consider $\gamma = 10\%, 20\%$ and $\eta = 80\%, 50\%$, leading to 4 combinations. In the results presented in this paper, we only show those for $\gamma = 10\%$ and $\eta = 20\%$. Note that the results for the other three combinations led to the same conclusions qualitatively.

- Step 4. For the remaining $(1-\eta\%)$, they are randomly set to be SZ or not with equal probabilities when we simulate the contact count matrix for each single cell. For a particular single cell, the new λ_{ij} value is set to be zero if a position is selected to be SZ; otherwise, the λ_{ij} value is left unchanged in the original λ matrix. This leads to be a λ^* matrix for a specific single cell. Therefore, the SZs among the $(1-\eta\%)$ positions vary from SC to SC.

- Step 5. Simulate a 2D contact matrix for a SC using the λ^* matrix; the contact count at each position is generated based on a Poisson distribution with the corresponding value in the λ^* matrix as the intensity parameter. Note that the count is set to zero (SZ) if the corresponding value in the λ^* matrix is zero. Also note that a zero may still result even if the corresponding value is not zero, and these are DOs. This completes the simulation of one SC; the SZs and DOs vary from SC to SC.

- Repeat steps 4 and 5 for as many time as needed to obtain the desired number of SCs (sample size). We consider three sample sizes: 10, 50, and 100 SCs.

Finally, we created bulk data by combining the 2D contact matrices from 540 SCs equally divided among the three cell types (180 for each type).

Data Availability

The three real datasets analyzed are available at https://github.com/Queen0044/scHiC_data. The HiCImpute R package, together with the simulated data used in this study, are available on Github: https://github.com/Queen0044/HiCImpute.git.

References

1. N Naumova, M Imakaev, G Fudenberg, Y Zhan, BR Lajoie, LA Mirny, and J Dekker. Organization of the mitotic chromosome. *Science*, 342:948–953, 2013.

2. Elizabeth H Finn, Gianluca Pegoraro, Hugo B Brandão, Anne-Laure Valton, Marlies E Oomen, Job Dekker, Leonid Mirny, and Tom Misteli. Extensive heterogeneity and intrinsic variation in spatial genome organization. *Cell*, 176(6):1502–1515, 2019.

3. Vijay Ramani, Xinxian Deng, Ruolan Qiu, Choli Lee, Christine M Disteche, William S Noble, Jay Shendure, and Zhijun Duan. Sci-hi-c: a single-cell hi-c method for mapping 3d genome organization in large number of single cells. *Methods*, 2019.

4. Jingtian Zhou, Jianzhu Ma, Yusi Chen, Chuankai Cheng, Bokan Bao, Jian Peng, Terrence J Sejnowski, Jesse R Dixon, and Joseph R Ecker. Robust single-cell hi-c clustering by convolution-and random-walk–based imputation. *Proceedings of the National Academy of Sciences*, page 201901423, 2019.

5. Michael Rosenthal, Darshan Bryner, Fred Huffer, Shane Evans, Anuj Srivastava, and Nicola Neretti. Bayesian estimation of three-dimensional chromosomal structure from single-cell hi-c data. *Journal of Computational Biology*, 26(11):1191–1202, 2019.

6. ZhiZhuo Zhang, Guoliang Li, Kim-Chuan Toh, and Wing-Kin Sung. 3d chromosome modeling with semi-definite programming and hi-c data. *Journal of Computational Biology*, 20(11):831–846, 2013.

7. David van Dijk, Juozas Nainys, Roshan Sharma, Pooja Kathail, Ambrose J Carr, Kevin R Moon, Linas Mazutis, Guy Wolf, Smita Krishnaswamy, and Dana Pe'er. Magic: A diffusion-based imputation method reveals gene-gene interactions in single-cell rna-sequencing data. *BioRxiv*, page 111591, 2017.

8. Mo Huang, Jingshu Wang, Eduardo Torre, Hannah Dueck, Sydney Shaffer, Roberto Bonasio, John Murray, Arjun Raj, Mingyao Li, and Nancy R Zhang. Gene expression recovery for single cell rna sequencing. *bioRxiv*, page 138677, 2017.

9. Lingxue Zhu, Jing Lei, Bernie Devlin, and Kathryn Roeder. A unified statistical framework for single cell and bulk rna sequencing data. *The annals of applied statistics*, 12(1):609, 2018.

10. Wei Vivian Li and Jingyi Jessica Li. An accurate and robust imputation method scimpute for single-cell rna-seq data. *Nature communications*, 9(1):1–9, 2018.

11. Aanchal Mongia, Debarka Sengupta, and Angshul Majumdar. Mcimpute: Matrix completion based imputation for single cell rna-seq data. *Frontiers in genetics*, 10:9, 2019.

12. Wuming Gong, Il-Youp Kwak, Pruthvi Pota, Naoko Koyano-Nakagawa, and Daniel J Garry. Drimpute: imputing dropout events in single cell rna sequencing data. *BMC bioinformatics*, 19(1):220, 2018.

13. Lihua Zhang and Shihua Zhang. Comparison of computational methods for imputing single-cell rna-sequencing data. *IEEE/ACM transactions on computational biology and bioinformatics*, 2018.

14. Chenggong Han, Qing Xie, and Shili Lin. Are dropout imputation methods for scrna-seq effective for schi-c data? *Briefings in Bioinformatics*, 2020.

15. Tao Yang, Feipeng Zhang, Galip Gürkan Yardımcı, Fan Song, Ross C Hardison, William Stafford Noble, Feng Yue, and Qunhua Li. Hicrep: assessing the reproducibility of hi-c data using a stratum-adjusted correlation coefficient. *Genome research*, 27(11):1939–1949, 2017.

16. Hao Zhu and Zheng Wang. Scl: a lattice-based approach to infer 3d chromosome structures from single-cell hi-c data. *Bioinformatics*, 35(20):3981–3988, 2019.

17. Oana Ursu, Nathan Boley, Maryna Taranova, YX Rachel Wang, Galip Gurkan Yardimci, William Stafford Noble, and Anshul Kundaje. Genomedisco: A concordance score for chromosome conformation capture experiments using random walks on contact map graphs. *Bioinformatics*, 34(16):2701–2707, 2018.

18. Yan Zhang, Lin An, Jie Xu, Bo Zhang, W Jim Zheng, Ming Hu, Jijun Tang, and Feng Yue. Enhancing hi-c data resolution with deep convolutional neural network hicplus. *Nature communications*, 9(1):750, 2018.

19. Caiwei Zhen, Yuxian Wang, Lu Han, Jingyi Li, Jinghao Peng, Tao Wang, Jianye Hao, Xuequn Shang, Zhongyu Wei, and Jiajie Peng. A novel framework for single-cell hi-c clustering based on graph-convolution-based imputation and two-phase-based feature extraction. *bioRxiv*, 2021.

20. Miao Yu, Armen Abnousi, Yanxiao Zhang, Guoqiang Li, Lindsay Lee, Ziyin Chen, Rongxin Fang, Jia Wen, Quan Sun, Yun Li, et al. Snaphic: a computational pipeline to map chromatin contacts from single cell hi-c data. *bioRxiv*, 2020.

21. Hao Hong, Shuai Jiang, Hao Li, Guifang Du, Yu Sun, Huan Tao, Cheng Quan, Chenghui Zhao, Ruijiang Li, Wanying Li, et al. Deephic: A generative adversarial network for enhancing hi-c data resolution. *PLoS computational biology*, 16(2):e1007287, 2020.

22. Ilya M Flyamer, Johanna Gassler, Maxim Imakaev, Hugo B Brandão, Sergey V Ulianov, Nezar Abdennur, Sergey V Razin, Leonid A Mirny, and Kikuë Tachibana-Konwalski. Single-nucleus hi-c reveals unique chromatin reorganization at oocyte-to-zygote transition. *Nature*, 544(7648):110–114, 2017.

23. Guanghua Xiao, Xinlei Wang, and Arkady B Khodursky. Modeling three-dimensional chromosome structures using gene expression data. *Journal of the American Statistical Association*, 106(493):61–72, 2011.

24. ZhiZhuo Zhang, Guoliang Li, Kim-Chuan Toh, and Wing-Kin Sung. Inference of spatial organizations of chromosomes using semi-definite embedding approach and hi-c data. In *Annual international conference on research in computational molecular biology*, pages 317–332. Springer, 2013.

25. Michael Rosenthal, Darshan Bryner, Fred Huffer, Shane Evans, Anuj Srivastava, and Nicola Neretti. Bayesian estimation of three-dimensional chromosomal structure from single-cell hi-c data. *Journal of Computational Biology*, 2019.

26. Longzhi Tan, Dong Xing, Chi-Han Chang, Heng Li, and X Sunney Xie. Three-dimensional genome structures of single diploid human cells. *Science*, 361(6405):924–928, 2018.

27. Dong-Sung Lee, Chongyuan Luo, Jingtian Zhou, Sahaana Chandran, Angeline Rivkin, Anna Bartlett, Joseph R Nery, Conor Fitzpatrick, Carolyn O'Connor, Jesse R Dixon, et al. Simultaneous profiling of 3d genome structure and dna methylation in single human cells. *Nature methods*, 16(10):999–1006, 2019.

28. Xiaoning Tang, Yongmei Huang, Jinli Lei, Hui Luo, and Xiao Zhu. The single-cell sequencing: new developments and medical applications. *Cell & bioscience*, 9(1):1–9, 2019.

29. Rebecca D Hodge, Trygve E Bakken, Jeremy A Miller, Kimberly A Smith, Eliza R Barkan, Lucas T Graybuck, Jennie L Close, Brian Long, Nelson Johansen, Osnat Penn, et al. Conserved cell types with divergent features in human versus mouse cortex. *Nature*, 573(7772):61–68, 2019.

30. Bosiljka Tasic, Zizhen Yao, Lucas T Graybuck, Kimberly A Smith, Thuc Nghi Nguyen, Darren Bertagnolli, Jeff Goldy, Emma Garren, Michael N Economo, Sarada Viswanathan, et al. Shared and distinct transcriptomic cell types across neocortical areas. *Nature*, 563(7729):72–78, 2018.

31. Damon Polioudakis, Luis de la Torre-Ubieta, Justin Langerman, Andrew G Elkins, Xu Shi, Jason L Stein, Celine K Vuong, Susanne Nichterwitz, Melinda Gevorgian, Carli K Opland, et al. A single-cell transcriptomic atlas of human neocortical development during mid-gestation. *Neuron*, 103(5):785–801, 2019.

32. Russell J Ferland, Timothy J Cherry, Patricia O Preware, Edward E Morrisey, and Christopher A Walsh. Characterization of foxp2 and foxp1 mrna and protein in the developing and mature brain. *Journal of comparative Neurology*, 460(2):266–279, 2003.

33. Bradley J Molyneaux, Paola Arlotta, Joao RL Menezes, and Jeffrey D Macklis. Neuronal subtype specification in the cerebral cortex. *Nature reviews neuroscience*, 8(6):427–437, 2007.

34. Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of classification*, 2(1):193–218, 1985.

35. Bosiljka Tasic, Vilas Menon, Thuc Nghi Nguyen, Tae Kyung Kim, Tim Jarsky, Zizhen Yao, Boaz Levi, Lucas T Gray, Staci A Sorensen, Tim Dolbeare, et al. Adult mouse cortical cell taxonomy revealed by single cell transcriptomics. *Nature neuroscience*, 19(2):335–346, 2016.

36. Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

37. Jincheol Park and Shili Lin. Evaluation and comparison of methods for recapitulation of 3d spatial chromatin structures. *Briefings in bioinformatics*, 20(4):1205–1214, 2019.

38. Stephen P Brooks and Andrew Gelman. General methods for monitoring convergence of iterative simulations. *Journal of computational and graphical statistics*, 7(4):434–455, 1998.

39. Chong Chen, Changjing Wu, Linjie Wu, Yishu Wang, Minghua Deng, and Ruibin Xi. scrmd: Imputation for single cell rna-seq data via robust matrix decomposition. *bioRxiv*, page 459404, 2018.

40. Emily M Darrow, Miriam H Huntley, Olga Dudchenko, Elena K Stamenova, Neva C Durand, Zhuo Sun, Su-Chen Huang, Adrian L Sanborn, Ido Machol, Muhammad Shamim, et al. Deletion of dxz4 on the human inactive x chromosome alters higher-order genome architecture. *Proceedings of the National Academy of Sciences*, 113(31):E4504–E4512, 2016.

41. Elzo de Wit and Wouter De Laat. A decade of 3c technologies: insights into nuclear organization. *Genes & development*, 26(1):11–24, 2012.

42. Job Dekker. Gene regulation in the third dimension. *Science*, 319(5871):1793–1794, 2008.

43. Jesse R Dixon, Siddarth Selvaraj, Feng Yue, Audrey Kim, Yan Li, Yin Shen, Ming Hu, Jun S Liu, and Bing Ren. Topological domains in mammalian genomes identified by analysis of chromatin interactions. *Nature*, 485(7398):376, 2012.

44. Peter Fraser and Wendy Bickmore. Nuclear organization of the genome and the potential for gene regulation. *Nature*, 447(7143):413–417, 2007.

45. James Fraser, Iain Williamson, Wendy A Bickmore, and Josée Dostie. An overview of genome organization and how we got there: from fish to hi-c. *Microbiol. Mol. Biol. Rev.*, 79(3):347–372, 2015.

46. Andrew Gelman, Donald B Rubin, et al. Inference from iterative simulation using multiple sequences. *Statistical science*, 7(4):457–472, 1992.

47. Ming Hu, Ke Deng, Zhaohui Qin, Jesse Dixon, Siddarth Selvaraj, Jennifer Fang, Bing Ren, and Jun S Liu. Bayesian inference of spatial organizations of chromosomes. *PLoS computational biology*, 9(1):e1002893, 2013.

48. Daniel Hsu, Sham M Kakade, and Tong Zhang. Robust matrix decomposition with sparse corruptions. *IEEE Transactions on Information Theory*, 57(11):7221–7234, 2011.

49. Peter V Kharchenko, Lev Silberstein, and David T Scadden. Bayesian approach to single-cell differential expression analysis. *Nature methods*, 11(7):740, 2014.

50. Seungsoo Kim, Ivan Liachko, Donna G Brickner, Kate Cook, William S Noble, Jason H Brickner, Jay Shendure, and Maitreya J Dunham. The dynamic three-dimensional organization of the diploid yeast genome. *Elife*, 6:e23623, 2017.

51. Vladimir Yu Kiselev, Kristina Kirschner, Michael T Schaub, Tallulah Andrews, Andrew Yiu, Tamir Chandra, Kedar N Natarajan, Wolf Reik, Mauricio Barahona, Anthony R Green, et al. Sc3: consensus clustering of single-cell rna-seq data. *Nature methods*, 14(5):483–486, 2017.

52. Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

53. Annick Lesne, Julien Riposo, Paul Roger, Axel Cournac, and Julien Mozziconacci. 3d genome reconstruction from chromosomal contacts. *Nature methods*, 11(11):1141, 2014.

54. Erez Lieberman-Aiden, Nynke L Van Berkum, Louise Williams, Maxim Imakaev, Tobias Ragoczy, Agnes Telling, Ido Amit, Bryan R Lajoie, Peter J Sabo, Michael O Dorschner, et al. Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *science*, 326(5950):289–293, 2009.

55. Jie Liu, Dejun Lin, Galip Gürkan Yardımcı, and William Stafford Noble. Unsupervised embedding of single-cell hi-c data. *Bioinformatics*, 34(13):i96–i104, 2018.

56. Adriana Miele and Job Dekker. Long-range chromosomal interactions and gene regulation. *Molecular biosystems*, 4(11):1046–1057, 2008.

57. Tom Misteli. Spatial positioning: A new dimension in genome function. *Cell*, 119(2):153–156, 2004.

58. Tom Misteli. Beyond the sequence: cellular organization of genome function. *Cell*, 128(4):787–800, 2007.

59. Takashi Nagano, Yaniv Lubling, Tim J Stevens, Stefan Schoenfelder, Eitan Yaffe, Wendy Dean, Ernest D Laue, Amos Tanay, and Peter Fraser. Single-cell hi-c reveals cell-to-cell variability in chromosome structure. *Nature*, 502(7469):59–64, 2013.

60. Takashi Nagano, Yaniv Lubling, Eitan Yaffe, Steven W Wingett, Wendy Dean, Amos Tanay, and Peter Fraser. Single-cell hi-c for genome-wide detection of chromatin interactions that occur simultaneously in a single cell. *Nature protocols*, 10(12):1986, 2015.

61. Jonas Paulsen, Odin Gramstad, and Philippe Collas. Manifold based optimization for single-cell 3d genome reconstruction. *PLoS computational biology*, 11(8), 2015.

62. Tao Peng, Qin Zhu, Penghang Yin, and Kai Tan. Scrabble: single-cell rna-seq imputation constrained by bulk rna-seq data. *Genome biology*, 20(1):88, 2019.

63. Sandhya Prabhakaran, Elham Azizi, Ambrose Carr, and Dana Pe'er. Dirichlet process mixture model for correcting technical variation in single-cell gene expression data. In *International Conference on Machine Learning*, pages 1070–1079, 2016.

64. Suhas SP Rao, Miriam H Huntley, Neva C Durand, Elena K Stamenova, Ivan D Bochkov, James T Robinson, Adrian L Sanborn, Ido Machol, Arina D Omer, Eric S Lander, et al. A 3d map of the human genome at kilobase resolution reveals principles of chromatin looping. *Cell*, 159(7):1665–1680, 2014.

65. Satish Sati and Giacomo Cavalli. Chromosome conformation capture technologies and their impact in understanding genome function. *Chromosoma*, 126(1):33–44, 2017.

Supporting Information

Markov chain Monte Carlo Procedure

In the following, we provide the prior specifications, the posterior distributions, the Markov chain Monte Carlo (MCMC) sampling schemes, and convergence diagnostics. The notations are the same as those in the main paper and may not be reintroduced.

To distinguish the SZs from the DOs, we define an indicator variable S_{ij}, which equals to 1 if loci i and j do not interact; otherwise, it is 0. That is, $S_{ij} \sim \text{Bernoulli}(\pi_{ij})$, where π_{ij} is the probability that pair i and j do not interact. This probability, π_{ij}, is assumed to follow a Beta distribution with parameters a_{ij}^{β} and b_{ij}^{β}, and a_{ij}^{β} is further assumed to be uniformly distributed on (1,1000) to account for a large range of possible shapes. The mean of the Beta distribution, $\frac{a_{ij}^{\beta}}{a_{ij}^{\beta}+b_{ij}^{\beta}}$, is governed by the proportion of observed zeros at (i,j), denoted as $\hat{p}_{ij}$. Specifically, we define δ_{ij} to be a uniformly distributed variable that centers at $\hat{p}_{ij}$ with radius ϵ_1 (default is set to be 0.5), and we let $logit(\frac{a_{ij}^{\beta}}{a_{ij}^{\beta}+b_{ij}^{\beta}})$ follow a Normal distribution with mean and standard deviation being $logit(\delta_{ij})$ and σ_{δ}, respectively. That is, if we observe a large proportion of zeros for pair i and j, then it is more likely, a priori, that the pair is a structural zero.

We allow for cell-to-cell variability by setting up an additional hierarchy to model μ_{ij}^k as follows: $\mu_{ij}^k \sim \text{Normal}^+(\mu_{ij}, \sigma_{ij}^2)$, where Normal^+ is a truncated normal distribution on positive numbers, σ_{ij}^2 is taken to be the standard deviation of nonzero counts in a neighborhood centered at (i,j), and μ_{ij} is further assumed to follow a Gamma distribution with shape and scale parameters being α_{ij} and β_{ij}, respectively. Its mean, $\alpha_{ij}\beta_{ij}$, borrows information from both the bulk Hi-C and the neighborhood data across similar SCs, as already described in the main text. Further, α_{ij} is assumed to follow a Uniform distribution on (1,1000) to allow for a wide variety of shapes for the distribution.

To make inferences about the parameters, we devise a Markov chain Monte Carlo (MCMC) sampling procedure as follows. We first write the posterior distribution of Θ (a vector containing all parameters, including π_{ij}'s and μ_{ij}^k's, the main parameters of interest, as well as nuisance parameters):

$$
\begin{aligned}
P(\Theta|s,y) &\propto P(y|s,\Theta) \times P(s|\Theta) \times P(\Theta) \\
&\propto \prod_{(i,j,k):y_{ijk}>0} \frac{(\lambda^k \mu_{ij}^k)^{y_{ijk}} e^{-\lambda^k \mu_{ij}^k}}{y_{ijk}!} \prod_{(i,j,k):y_{ijk}=0} \left[e^{-\lambda^k \mu_{ij}^k}\right]^{1-s_{ij}} \\
&\times \prod_{i,j} [\pi_{ij}]^{\mathbb{1}_{\{s_{ij}=1\}}} [1-\pi_{ij}]^{\mathbb{1}_{\{s_{ij}=0\}}} \\
&\times \prod_{ij} \prod_{k} \frac{\phi(\frac{\mu_{ij}^k - \mu_{ij}}{\sigma_\mu})}{\sigma_\mu[1-\Phi(-\mu_{ij}/\sigma_\mu)]} \\
&\times \prod_{i,j} (\pi_{ij})^{a_{ij}^t - 1} (1-\pi_{ij})^{b_{ij}^t - 1} \frac{\Gamma(a_{ij}^t + b_{ij}^t)}{\Gamma(a_{ij}^t)\Gamma(b_{ij}^t)} \\
&\times \prod_{i,j} \frac{1}{1000-1} \mathbb{1}_{\{1 \le a_{ij}^t \le 1000\}} \\
&\times \prod_{i,j} exp\{-\frac{1}{2\sigma_\delta^2}(logit(\frac{a_{ij}}{a_{ij}+b_{ij}}) - logit(\delta_{ij}))^2\} \frac{1}{\frac{a_{ij}}{a_{ij}+b_{ij}}(1-\frac{a_{ij}}{a_{ij}+b_{ij}})} \\
&\times \prod_{i,j} \frac{1}{min\{\hat{p}+\epsilon_1, 1\} - max\{\hat{p}-\epsilon_1, 0\}} \mathbb{1}_{\{max\{\hat{p}-\epsilon_1,0\} \le \delta_{ij} \le min\{\hat{p}+\epsilon_1,1\}\}} \\
&\times \prod_{i,j} \frac{1}{\Gamma(\alpha_{ij}^t)(\beta_{ij}^t)^{\alpha_{ij}^t}} \mu_{ij}^{\alpha_{ij}^t - 1} e^{-\mu_{ij}/\beta_{ij}^t} \\
&\times \prod_{i,j} \frac{1}{1000-1} \mathbb{1}_{\{1 \le \alpha_{ij}^t \le 1000\}} \\
&\times \prod_{i,j} \frac{1}{(B_{ij}+\epsilon_2) - max\{0, B_{ij}-\epsilon_2\}} \mathbb{1}_{\{max\{0,B_{ij}-\epsilon_2\} \le \alpha_{ij}\beta_{ij} \le B_{ij}+\epsilon_2\}}
\end{aligned}
$$

To sample from the posterior distributions of the parameters in Θ, we use Metropolis-Hastings algorithms, and in particular the Gibbs sampler whenever the conditional distribution of a parameter is of a commonly known one. In the following, we briefly describe the updating schemes. We first note that Θ_{-g} denote the subvector of Θ that includes all the parameters except g.

- Update α_{ij}^t:

  Using the current α_{ij}^t, sample a candidate α_{ij}^{t*} from the proposal distribution $J_{\alpha_{ij}}(\alpha_{ij}^{t*}|\alpha_{ij}^t)$, a Uniform$(1, 1000)$ distribution, and calculate the ratio of the densities,

  $$r = \frac{p(\alpha_{ij}^{t*}|y, \Theta_{-\alpha_{ij}^t})}{p(\alpha_{ij}^t|y, \Theta_{-\alpha_{ij}^t})}$$

  where

  $$p(\alpha_{ij}^{t*}|y, \Theta_{-\alpha_{ij}^t}) \propto \frac{1}{\Gamma(\alpha_{ij}^t)(\beta_{ij}^t)^{\alpha_{ij}^t}} \mu_{ij}^{\alpha_{ij}^t - 1} e^{-\mu_{ij}/\beta_{ij}} \mathbb{1}_{\{0 \le \alpha_{ij} \le 1000\}} \mathbb{1}_{\{max\{0,B_{ij}-\epsilon_2\} \le \alpha_{ij}\beta_{ij} \le B_{ij}+\epsilon}$$

  Accept α_{ij}^{t*} with probability $min(r, 1)$.

- Update β_{ij}:

  Sample μ_{ij} from a Uniform$(\max\{0, B_{ij}-\epsilon_2\}, B_{ij}+\epsilon_2)$ distribution, and solve for β_{ij} using $\alpha_{ij}\beta_{ij} = \mu_{ij}$.

- Update μ_{ij}:

  Using the current μ_{ij}^t, sample a candidate μ_{ij}^{t*} from the proposal distribution $J_{\mu_{ij}}(\mu_{ij}^{t*}|\mu_{ij}^t)$, a $Normal^+(\mu_{ij}^t, 0.5)$ distribution, and calculate the ratio of the densities,

$$r = \frac{p(\mu_{ij}^{t*}|y, \Theta_{-\mu_{ij}^t})}{p(\mu_{ij}^t|y, \Theta_{-\mu_{ij}^t})}$$

  where

$$p(\mu_{ij}|y, \Theta_{-\mu_{ij}^t}) \propto \mu_{ij}^{\alpha_{ij}-1} e^{-\mu_{ij}/\beta_{ij}} \prod_{k=1}^{100} \frac{\phi(\mu_{ij}^k - \mu_{ij})/\sigma_\mu)}{\sigma_\mu[1 - \Phi(-\mu_{ij}/\sigma_\mu)]},$$

  and ϕ, Φ are the pdf and cdf of the standard normal distribution, respectively. Accept μ_{ij}^{t*} with probability $min(r, 1)$.

- Update $\mu_{ij}^k (k = 1, 2, \cdots, 100)$:

  Using the current μ_{ij}^k, sample a candidate μ_{ij}^{k*} from the proposal distribution $J_{\mu_{ij}^k}(\mu_{ij}^{k*}|\mu_{ij}^k)$, a $Normal^+(\mu_{ij}^k, 0.5)$ distribution, and calculate the ratio of the densities,

$$r = \frac{p(\mu_{ij}^{k*}|y, \Theta_{-\mu_{ij}^k})}{p(\mu_{ij}^k|y, \Theta_{-\mu_{ij}^k})},$$

  where

$$p(\mu_{ij}|y, \Theta_{-\mu_{ij}^k}) \propto \left[(\mu_{ij}^k)^{y_{ijk}} e^{-\lambda^k \mu_{ij}^k}\right]^{\mathbb{1}_{y_{ijk}>0}} \times \left[e^{-\lambda^k \mu_{ij}^k}\right]^{\mathbb{1}_{s_{ij}=0}\mathbb{1}_{y_{ijk}=0}} \phi(\frac{\mu_{ij}^k - \mu_{ij}}{\sigma_\mu})$$

  and ϕ is the pdf of the standard normal distribution. Accept μ_{ij}^{k*} with probability $min(r, 1)$.

- Update a_{ij}

  Using the current a_{ij}, sample a candidate a_{ij}^{t*} from the proposal distribution $J_{a_{ij}}(a_{ij}^{t*}|a_{ij})$, Uniform$(1, 1000)$, and calculate the ratio of the densities,

$$r = \frac{p(a_{ij}^{t*}|y, \Theta_{-a_{ij}})}{p(a_{ij}|y, \Theta_{-a_{ij}})}$$

  where

$$p(a_{ij}^{t*}|y, \Theta_{-a_{ij}}) \propto \pi_{ij}^{a_{ij}-1}(1 - \pi_{ij})^{b_{ij}-1} \frac{\Gamma(a_{ij} + b_{ij})}{\Gamma(a_{ij})} \mathbb{1}_{\{0 \le a_{ij} \le A_1\}}$$

  and

$$exp\{\frac{1}{2\sigma_\delta^2}(logit(\frac{a_{ij}}{a_{ij} + b_{ij}}) - logit(\delta_{ij}))^2\} \frac{1}{\frac{a_{ij}^\beta}{a_{ij}+b_{ij}}(1 - \frac{a_{ij}}{a_{ij}^\beta + b_{ij}})}.$$

  Accept a_{ij}^{t*} with probability $min(r, 1)$.

- Update δ_{ij}:

  Using the current δ_{ij}, sample a candidate δ_{ij}^{t*} from the proposal distribution $J_{\delta_{ij}}(\delta_{ij}^{t*}|\delta_{ij})$, a uniform$(max\{0, \hat{p}_{ij} - \epsilon_1\}, min\{\hat{p}_{ij} + \epsilon_1\})$ distribution, and calculate the ratio of the densities,

$$r = \frac{p(\delta_{ij}^{t*}|y, \Theta_{-\delta_{ij}})}{p(\delta_{ij}|y, \Theta_{-\delta_{ij}})}$$

where

$$p(logit(\delta_{ij})|B,S) \propto exp\{-\frac{1}{2\sigma_\delta^2}(logit(\frac{a_{ij}}{a_{ij}+b_{ij}})-logit(\delta_{ij}))^2\}\mathbb{1}_{\{max\{\hat{p}-\epsilon_1,0\}\leq\delta_{ij}\leq min\{\hat{p}+\epsilon_1,1\}}$$

Accept δ_{ij}^{t*} with probability $min(r,1)$.

- Update b_{ij}:
  Solve for b_{ij}, using $logit(\frac{a_{ij}}{a_{ij}+b_{ij}}) = logit(\delta_{ij})$.
- Update π_{ij}
  Sample π_{ij}^{t+1} from $Beta(\mathbb{1}_{\{s_{ij}=1\}}+a_{ij}, \mathbb{1}_{\{s_{ij}=0\}}+b_{ij})$ because

$$p(\pi_{ij}|B,S) \propto \pi_{ij}^{\mathbb{1}_{\{s_{ij}=1\}}+a_{ij}-1}(1-\pi_{ij})^{\mathbb{1}_{\{s_{ij}=0\}}+b_{ij}-1}.$$

- Update s_{ij}
  Sample s_{ij}^{t+1} from $Bernoulli(\frac{\pi_{ij}}{\pi_{ij}+(1-\pi_{ij})e^{-\sum_{k:y_{ijk}=0}\lambda^k\mu_{ij}^k}})$ because

$$p(s_{ij}|B,S) \propto [\pi_{ij}]^{s_{ij}}[1-\pi_{ij}]^{1-s_{ij}}[e^{-\sum_{k:y_{ijk}=0}\lambda^k\mu_{ij}^k}]^{1-s_{ij}}.$$

Convergence diagnostics. Trace plot, density plot, and cumulative mean plot were drawn to assess the performance of MCMC. An example cumulative mean plots for several parameters are provided to show that the chains converged property with stable estimates of the parameter values (Supplementary Figure S11). We also consider the Gelman–Rubin diagnostic by analyzing the difference between multiple Markov chains [38, 46]. Starting from different points, the three chains converged ultimately, and the scale reduction factors are all less than 1.1 for all the settings considered, indicating convergence. As an example, we show the trace plots and density plots for several parameters for the dataset with 10 single cells from T1 at 7K sequencing depth, which shows that the three chains are well mixed, consistent with the conclusion from considering the reduction factors (Figure S12). As a further evidence that our MCMC algorithm works well, we also provide an example to show that the autocorrelations for multiple parameters decay at a reasonable rate, as one would expect for a well-mixing chain (Figure S12).

Table S1. Partial list of existing methods for Hi-C data quality improvement.

Method	Goal	Hi-C Type	Category
HiCRep	Reproducibility	bulk	Kernel smoothing
SCL	3D Structure	single cells	Kernel smoothing
scHiCluster	Clustering	single cells	Kernel smoothing
scHiCluster	Clustering	single cells	Random walk
GenomeDISCO	Reproducibility	bulk	Random walk
SnapHi-C	Chromatin contacts	single cells	Random Walk
HiCPlus	Data Resolution	bulk	Neural network
DeepHiC	Data Resolution	bulk	Neural network

Table S2. K-means clustering results of L4 and L5 cells based on t-SNE embeded data. We considered 2-6 clusters for HiCImpute-improved data and 2-4 clusters for the rest since the results did not indicate any need for a greater number of clusters.

(a) Observed (a). ARI=-0.003.

cell type	1	2
L4	76	55
L5	105	75

(b) Observed (b). ARI=0.027.

cell type	1	2	3
L4	54	22	55
L5	41	64	75

(c) Observed (c). ARI=0.031.

cell type	1	2	3	4
L4	42	12	22	55
L5	22	51	32	75

(d) 2DMF (a). ARI=-0.003.

cell type	1	2
L4	77	54
L5	105	75

(e) 2DMF (b). ARI=-0.003.

cell type	1	2	3
L4	38	54	39
L5	47	74	59

(f) 2DMF (c). ARI=0.003.

cell type	1	2	3	4
L4	24	30	54	23
L5	25	32	74	49

(g) 2DGK (a). ARI=-0.003.

cell type	1	2
L4	77	54
L5	104	76

(h) 2DGK (b). ARI=-0.003.

cell type	1	2	3
L4	54	36	41
L5	76	47	57

(i) 2DGK (c). ARI=-0.004.

cell type	1	2	3	4
L4	36	41	31	23
L5	47	57	46	30

(j) RW3S (a). ARI=-0.003.

cell type	1	2
L4	76	55
L5	105	75

(k) RW3S (b). ARI=-0.004.

cell type	1	2	3
L4	37	39	55
L5	51	54	75

(l) RW3S (c). ARI=-0.004.

cell type	1	2	3	4
L4	29	26	39	37
L5	37	38	54	51

(m) HiCImpute (a). ARI=-0.002.

cell type	1	2
L4	76	55
L5	106	74

(n) HiCImpute (b). ARI=-0.002.

cell type	1	2	3
L4	55	0	76
L5	74	106	0

(o) HiCImpute (d). ARI=0.506.

cell type	1	2	3	4
L4	0	55	76	0
L5	74	0	0	106

(p) HiCImpute (d). ARI=0.392.

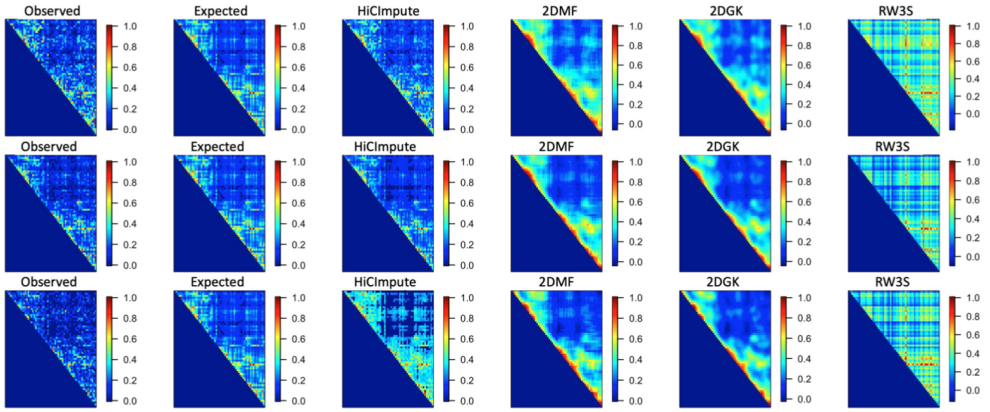
cell type	1	2	3	4	5
L4	0	55	0	0	76
L5	53	0	53	74	0

(q) HiCImpute (e). ARI=0.336.

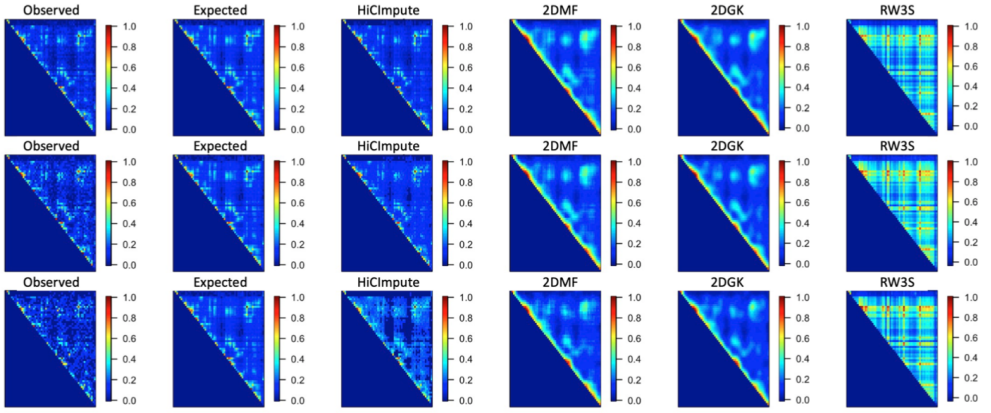
cell type	1	2	3	4	5	6
L4	55	32	0	44	0	0
L5	0	0	74	0	53	53

Table S3. Computation time comparison of packages on three real datasets.

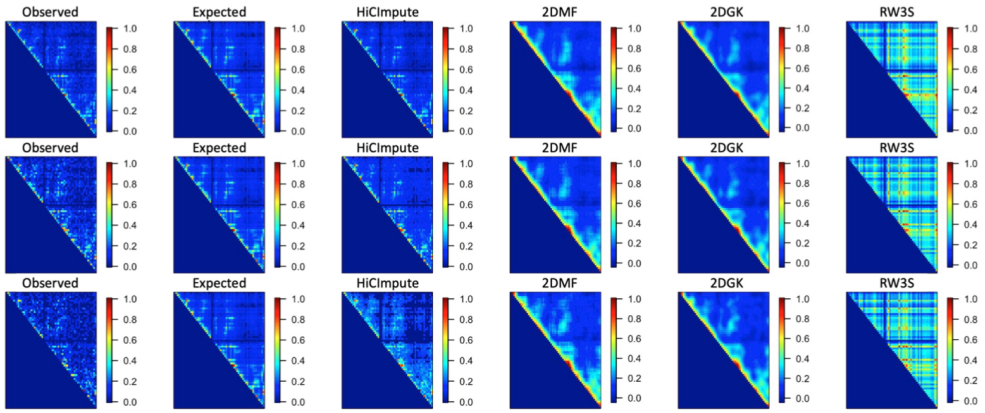
	HiCImpute	2DMF	2DGK	RW3S
GSE117874	6min	0.8s	1.5s	0.1s
GSE80006	2.5h	19s	15s	4s
scm3C-seq	17.3h	5min	4min	2min



(a) T1, 7k (top), 4k (middle), and 2k (bottom)

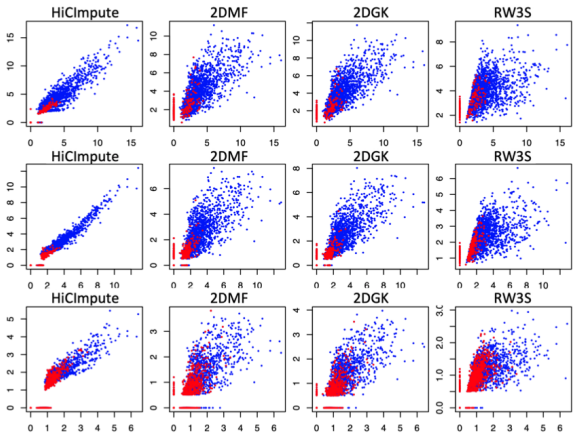


(b) T2, 7k (top), 4k (middle), and 2k (bottom)

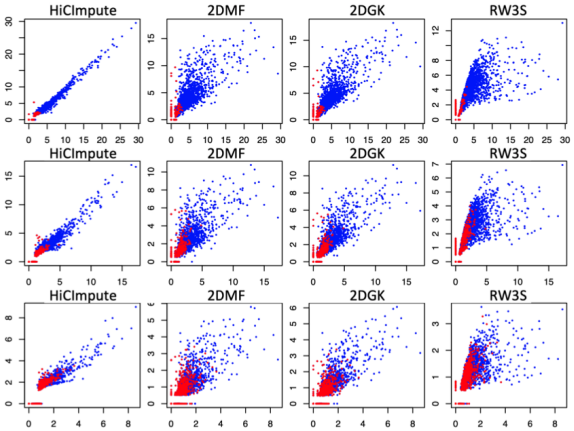


(c) T3, 7k (top), 4k (middle), and 2k (bottom)

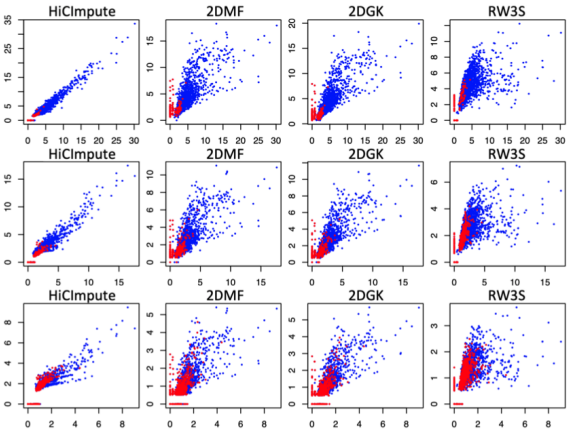
Figure S1. Heatmap showing the observed and true (expected) 2D matrix images as well as the results from HiCImpute, 2DMF, 2DGK, and RW3S for T1 (a), T2 (b), and T3 (c) cells at 7K (top), 4K (middle) and 2K (bottom) sequencing depth.



(a) T1, 7k (top), 4k (middle), and 2k (bottom)

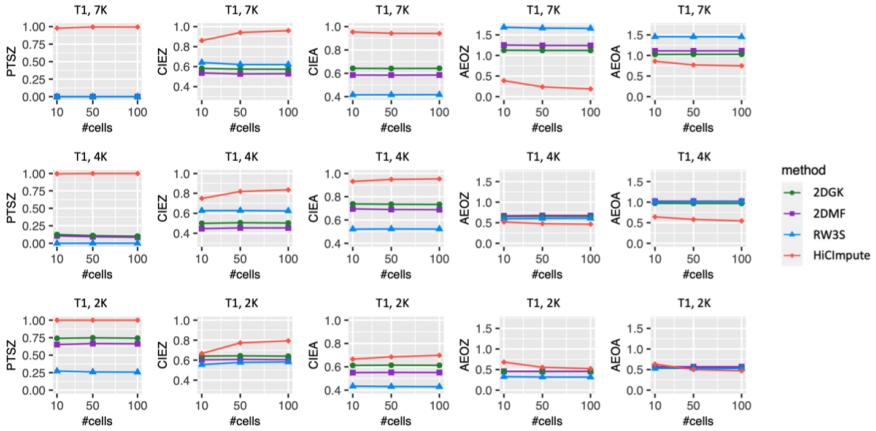


(b) T2, 7k (top), 4k (middle), and 2k (bottom)

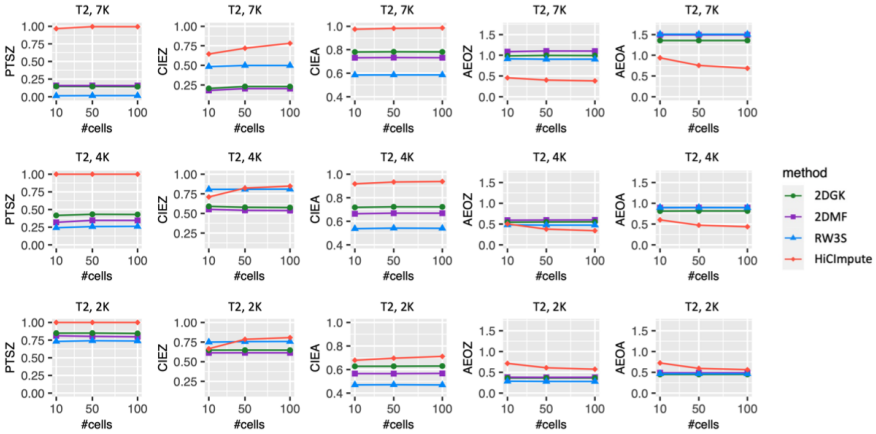


(c) T3, 7k (top), 4k (middle), and 2k (bottom)

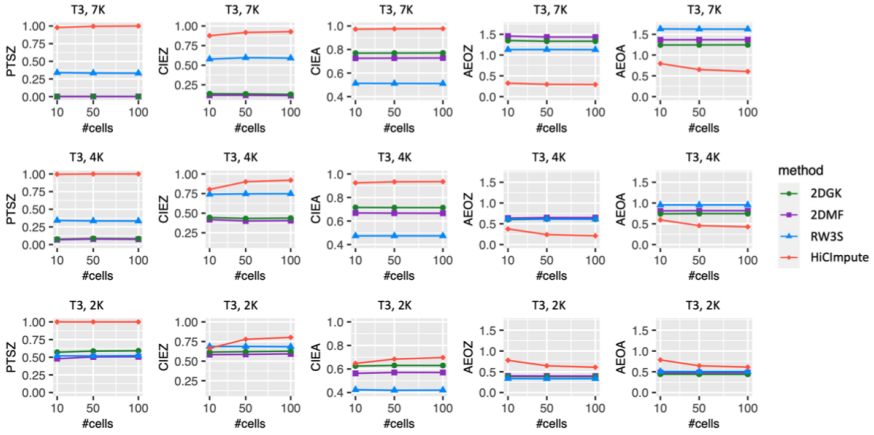
Figure S2. Scatterplots of Extected Versus Imputed (SEVI plots) for HiCImpute, 2DMF, 2DGK, and RW3S for T1 (a), T2 (b), and T3 (c) cells at 7K (top), 4K (middle) and 2K (bottom) sequencing depth – the red dots represent the observed zeros, which contain both true SZs and DOs.



(a) T1, 7k (top), 4k (middle), and 2k (bottom)



(b) T2, 7k (top), 4k (middle), and 2k (bottom)



(c) T3, 7k (top), 4k (middle), and 2k (bottom)

Figure S3. Aggregate results (over single cells) based on several evaluation criteria for T1 (a), T2 (b), and T3 (c) cells at 7K (top), 4K (middle) and 2K (bottom) sequencing depth.

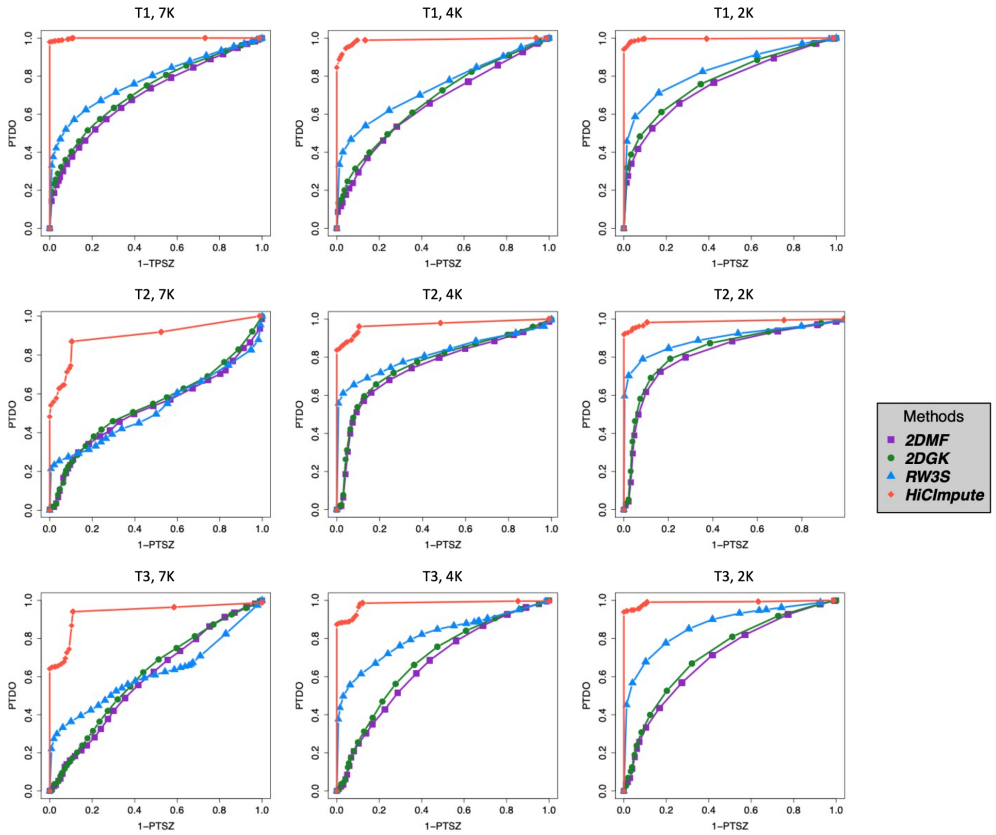


Figure S4. ROC curves accounting for both specificity and sensitivity of HiCImpute, 2DMF, 2DGK, and RW3S for T1 (row1), T2 (row2), and T3 (row 3) cells at 7K (column1), 4K (column2) and 2K (column 3) sequencing depth.

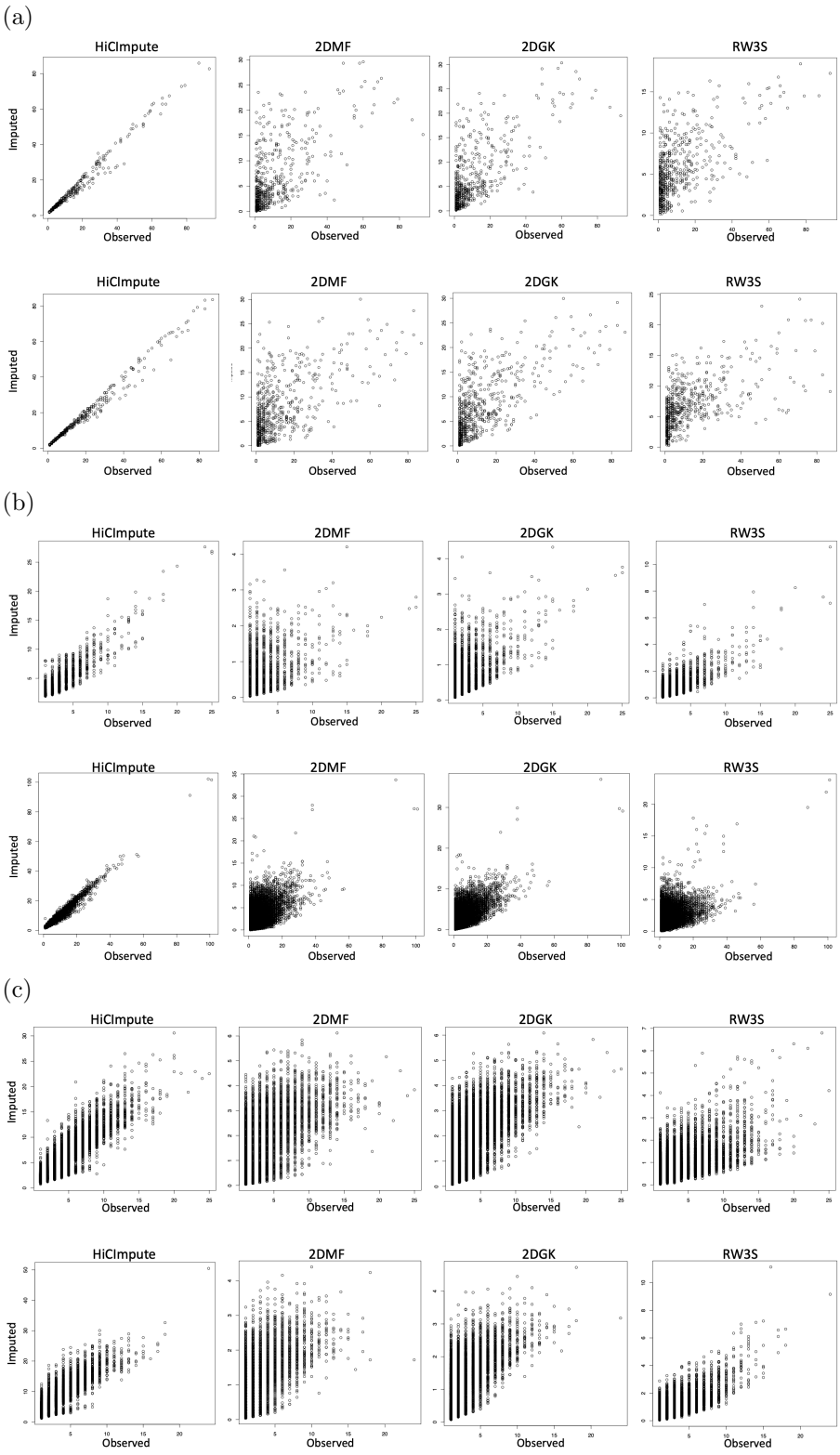


Figure S5. Scatterplots of observed versus imputed (SOVI plots) from HiCImpute, 2DMF, 2DGK, and RW3S. (a) GM (top row) and PMBC (bottom row); (b) K562A (top) and K562B (bottom); (c) L4 (top) and L5 (bottom).

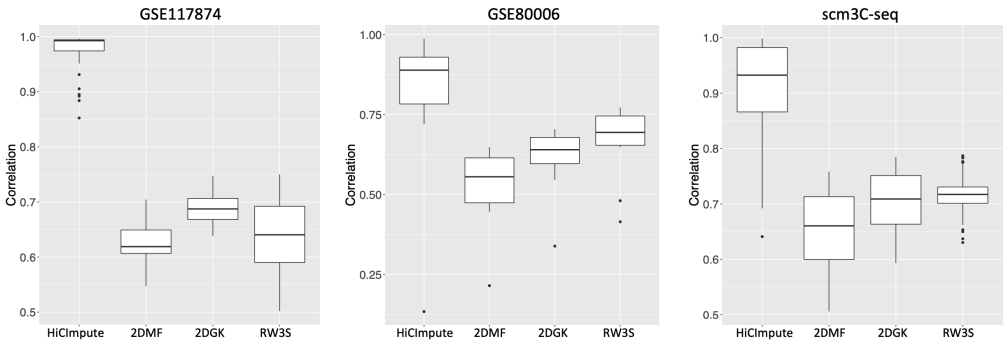


Figure S6. Boxplot of correlations between the observed and imputed from four methods for three datasets: GSE117874 (left), GSE80006 (middle), and scm3C-seq (right).

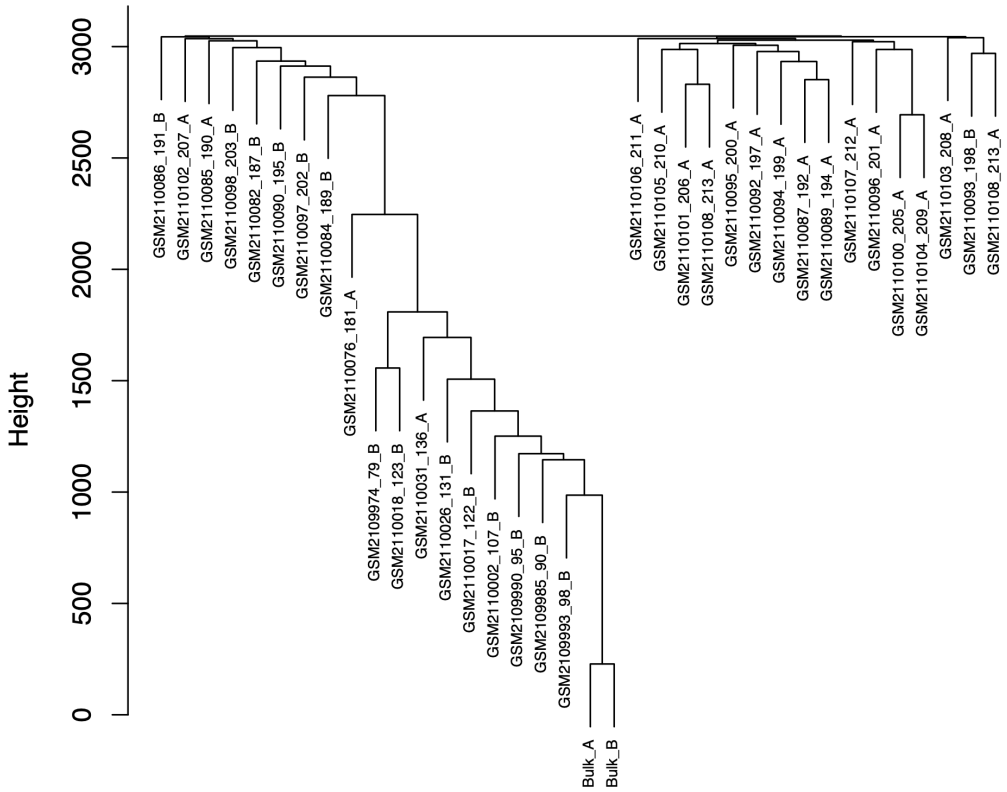


Figure S7. Dendrograms of 34 observed K562 single cells Hi-C data and two bulk datasets. The dendrogram was generated using the "complete" method.

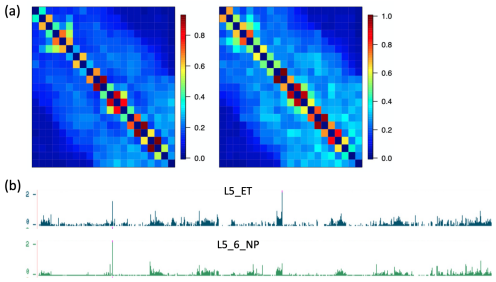


Figure S8. (a) Heatmaps of merged L4 subtype1 (left) and subtype2 (right) on chr8:127,000,000-147,000,000. (b) The mean RNAseq on chr8:127,000,000-147,000,000. The RNAseq plot is available in https://human-mtg-rna-hub.s3-us-west-2.amazonaws.com/HumanMTGRNAHub.html [29].

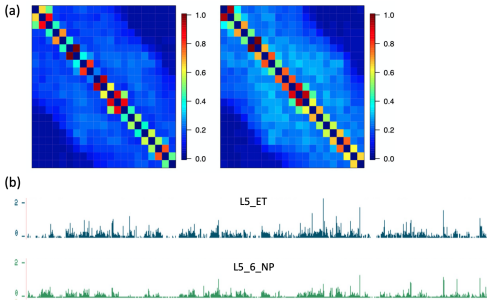


Figure S9. (a) Heatmaps of merged L4 subtype1 (left) and subtype2 (right) on chr11:105,000,000-125,000,000. (b) The mean RNAseq on chr11:105,000,000-125,000,000. The RNAseq plot is available in https://human-mtg-rna-hub.s3-us-west-2.amazonaws.com/HumanMTGRNAHub.html [29].

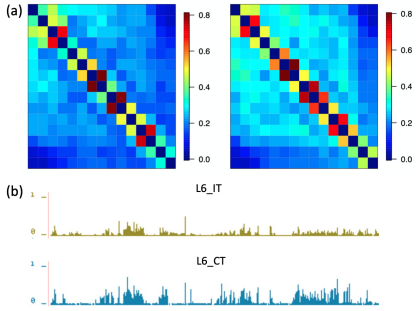


Figure S10. (a) Heatmap of merged L5 subtype1 (left) and subtype2 (right) on chr18:1,000,000-15,000,000. (b) The mean RNAseq on chr18:1,000,000-15,000,000. The RNAseq plot is available in https://human-mtg-rna-hub.s3-us-west-2.amazonaws.com/HumanMTGRNAHub.html [29].

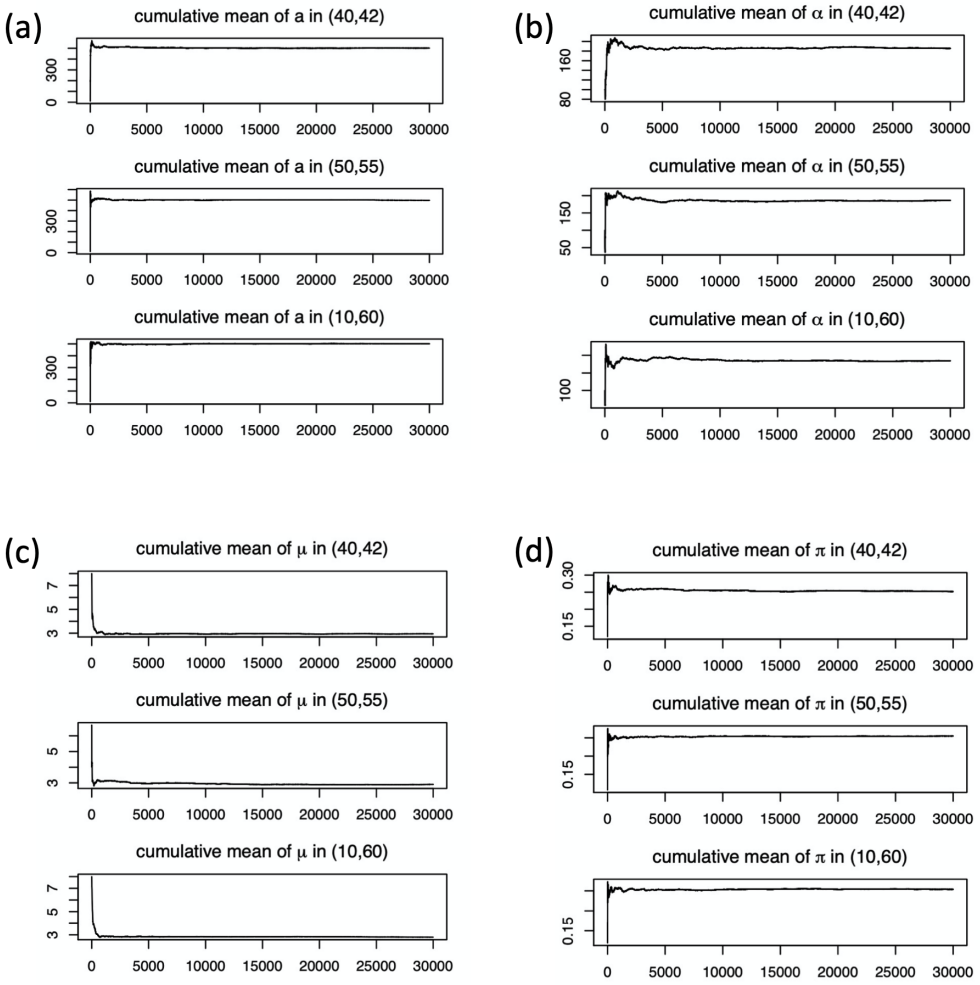


Figure S11. Cumulative mean plots of parameters a (a), α (b), μ (c) and π (d) at 3 positions of a dataset with 10 T1 cells at sequence depth 7k. Recall that a is the shape parameter of the Beta distribution that is the prior of π_{ij}; α is the shape parameter of Gamma distribution, which is the prior of μ_{ij}; μ is the mean of μ_{ij}^k; and π is the probability that the pair is a structural zero.

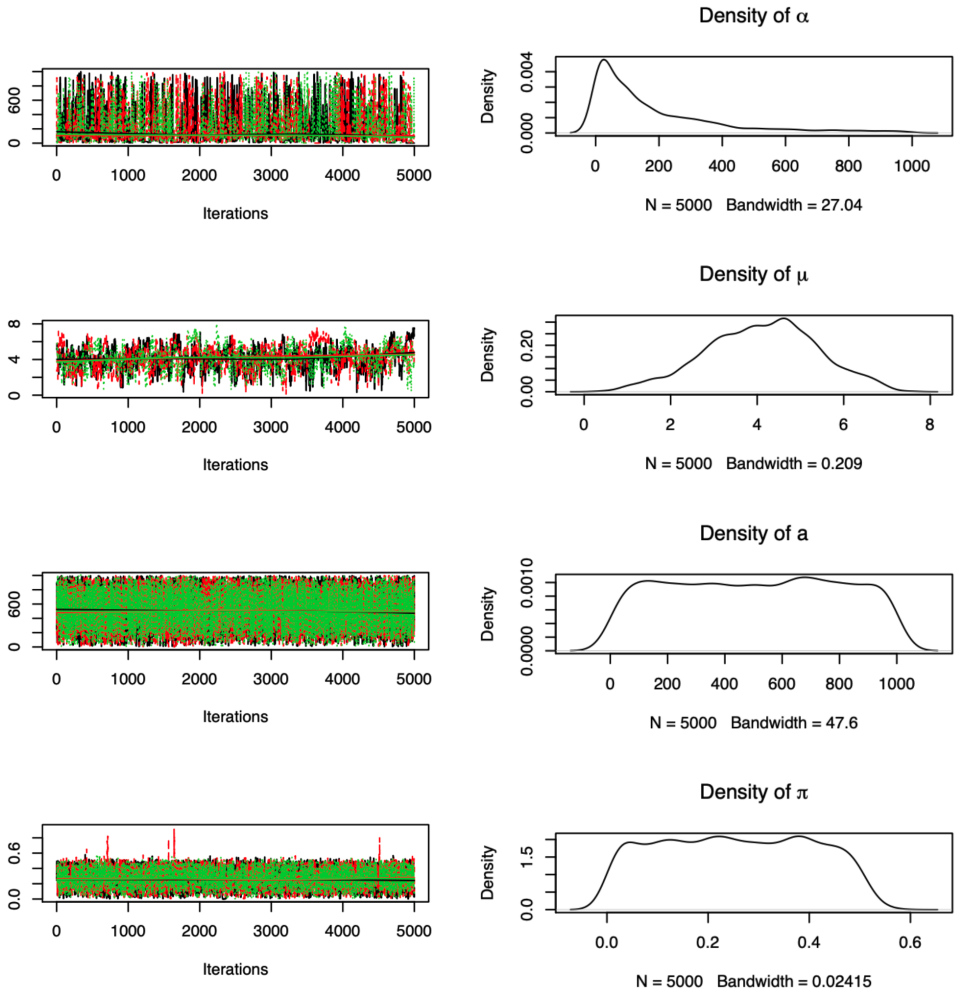


Figure S12. Trace plots of three chains starting from different points and the density of the parameters in the first chain for several parameters.

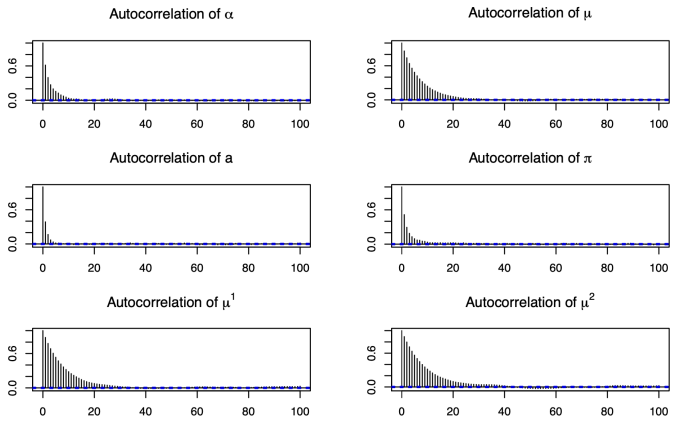


Figure S13. Autocorrelation plots for 6 parameters at position $(i, j) = (40, 42)$ for the simulated dataset with 10 T1 cells at 7K sequencing depth: μ is the overall expectation for all single cells, and μ^1 and μ^2 are the realizations in the first and second single cell, respectively.

Table S4. Potential scale reduction factors of 10 simulated T1 cells in sequence depth of 7k.

Parameters	Point estimate	Upper bound of C.I.
α	1.00	1.00
μ	1.02	1.01
a	1.00	1.00
π	1.00	1.00

Table S5. Raftery disgnosis of 10 T1 K562 simulated data, depth=7k, niter=30000

parameter	M (Burn-in)	N (Total)	Nmin (Lower bound)	I (Dependence factor)
α	39	42400	3746	11.30
μ_γ	2	3710	3746	0.99
β	2	3940	3746	1.05
μ	13	14054	3746	3.75
a	19	20429	3746	5.45
δ	38	40470	3746	10.80
b	15	18498	3746	4.94
π	39	43485	3746	11.60
s	4	60828	3746	16.20

Acknowledgements

This research is supported in part by a grant from the National Institute of Health R01GM114142. We thank Ms. Yongqi Liu for testing the software package.

Author Contributions

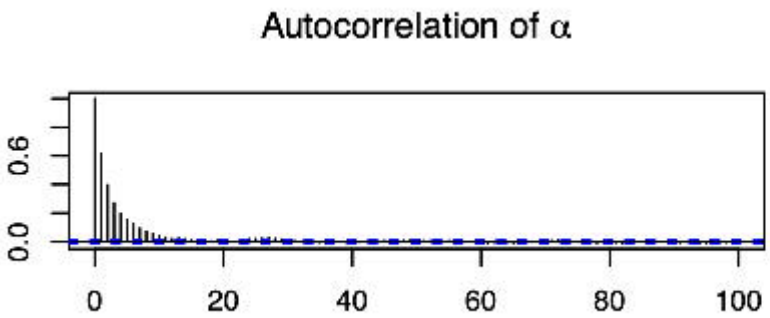
SL designed the study and supervised the project. QX conducted the research. CH contributed to the research. SL and QX wrote the manuscript. All authors contributed to the discussions and provided feedback.

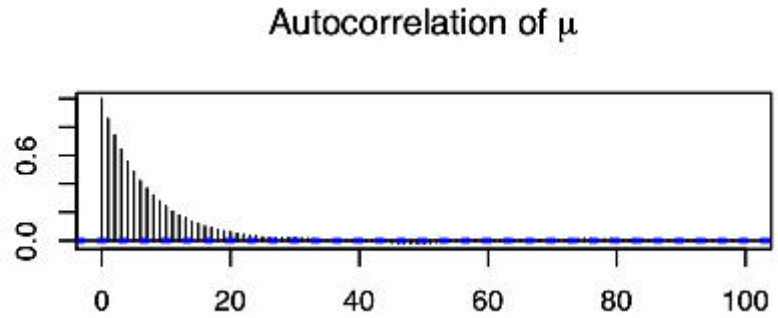
Competing interests

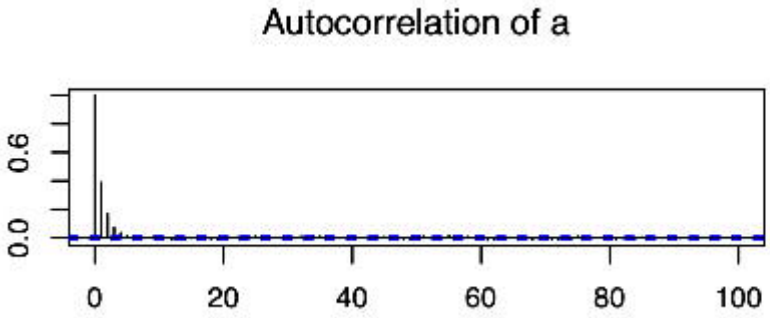
The authors declare no competing interests.

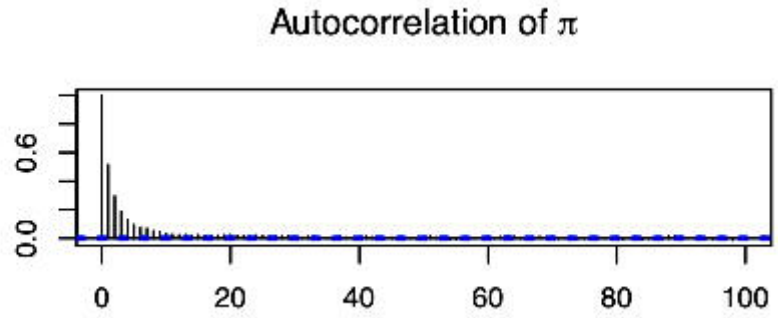
Additional information

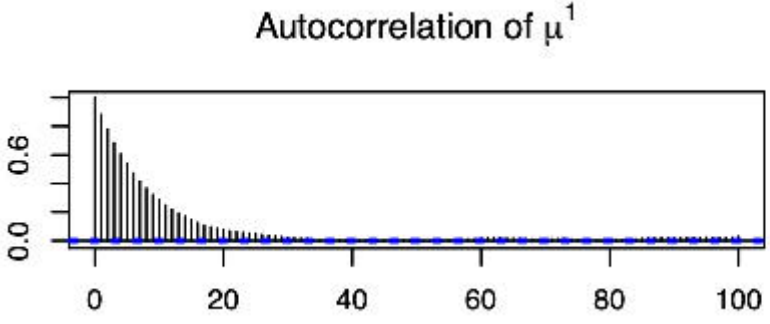
Correspondence and requests for materials should be addressed to SL.

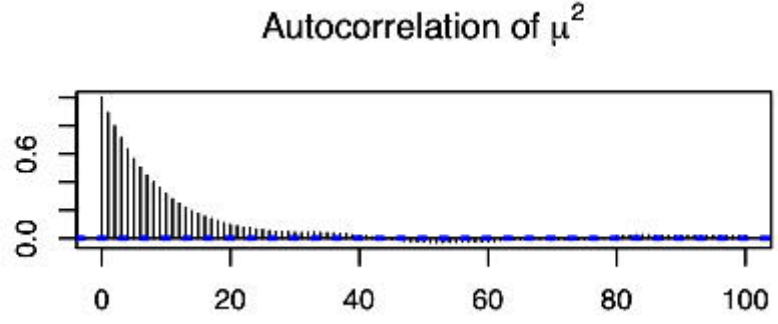

Autocorrelation of α
0.0
0.6
0
20
40
60
80
100

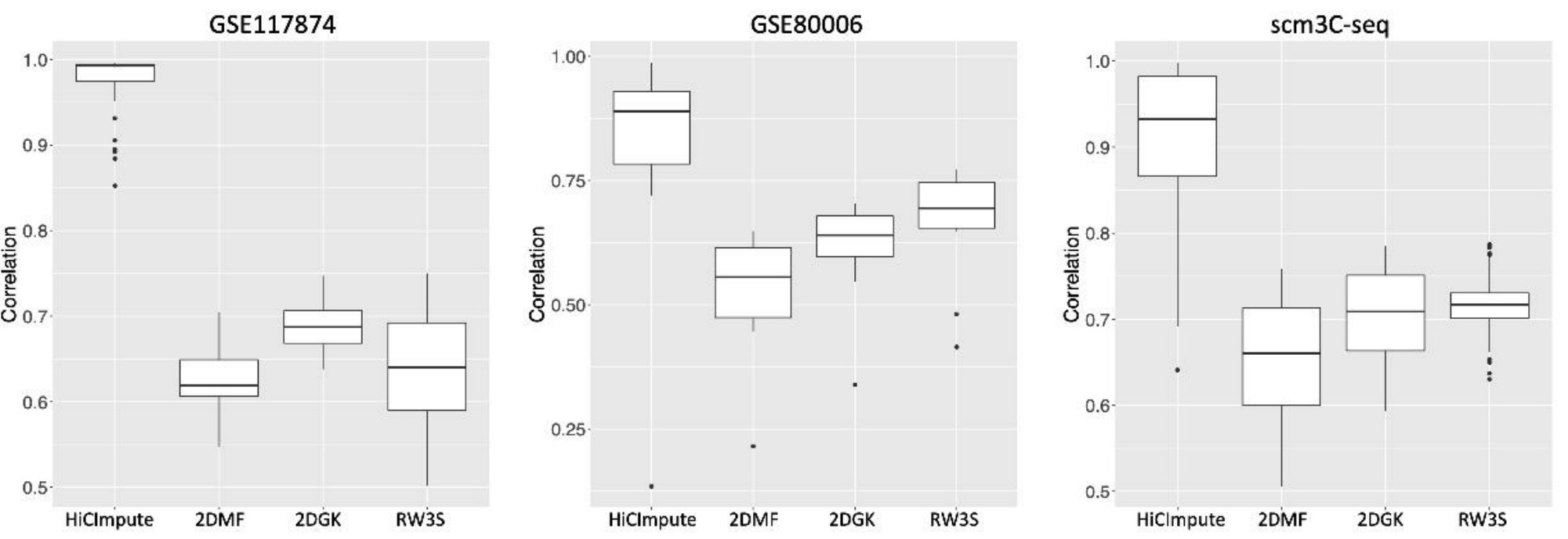

Autocorrelation of μ
0.0
0.6
0
20
40
60
80
100

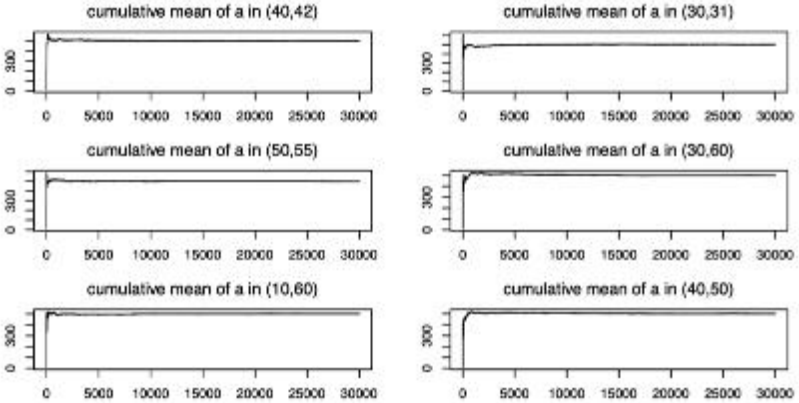

Autocorrelation of a
0.0
0.6
0
20
40
60
80
100


Autocorrelation of π
0.0
0.6
0
20
40
60
80
100

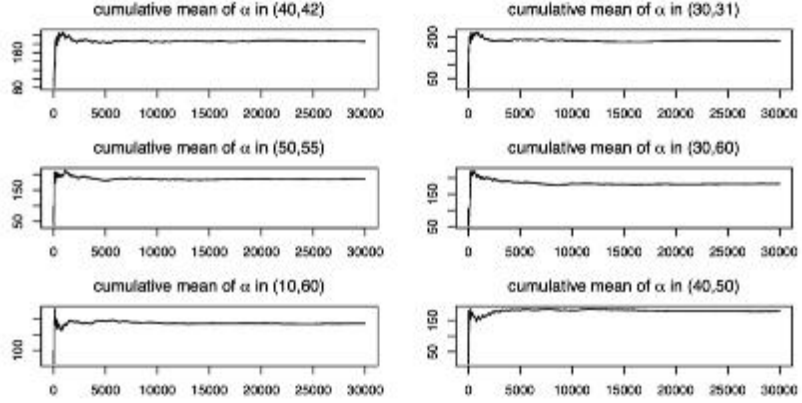

Autocorrelation of μ1
0.0
0.6
0
20
40
60
80
100


Autocorrelation of μ2
0.0
0.6
0
20
40
60
80
100

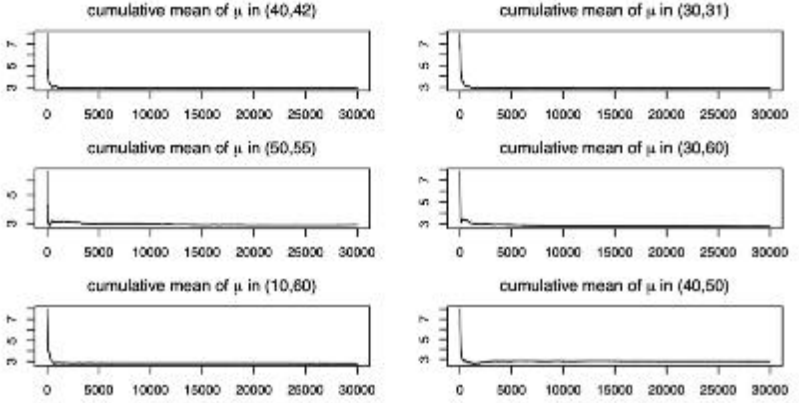

GSE117874
GSE80006
scm3C-seq
Correlation
1.0
0.9
0.8
0.7
0.6
0.5
0.75
0.50
0.25
HiCImpute
2DMF
2DGK
RW3S



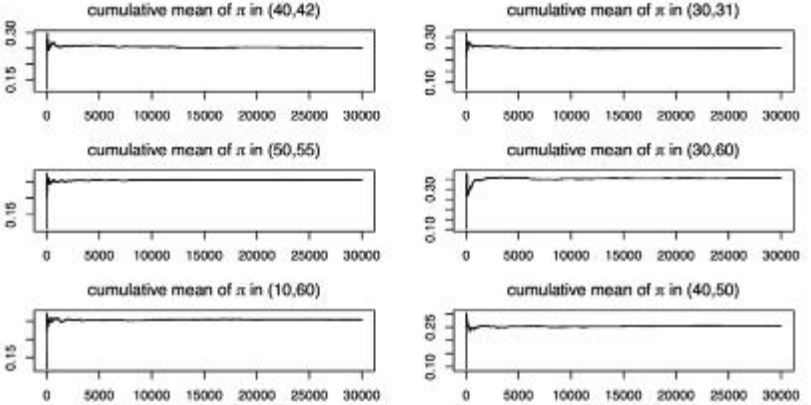
(a) a



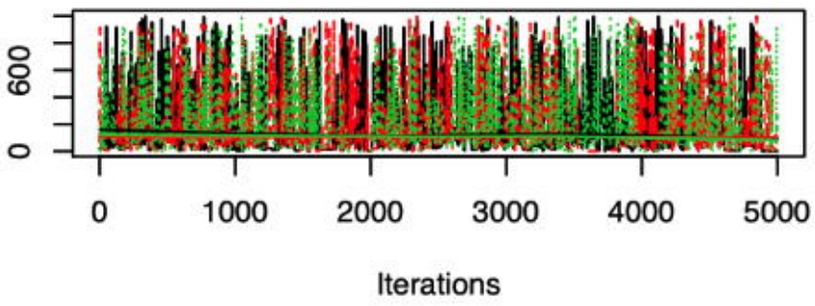
(b) α

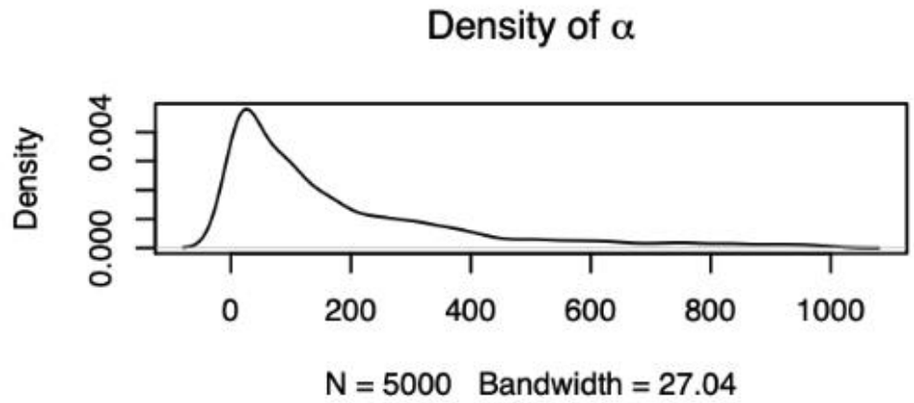


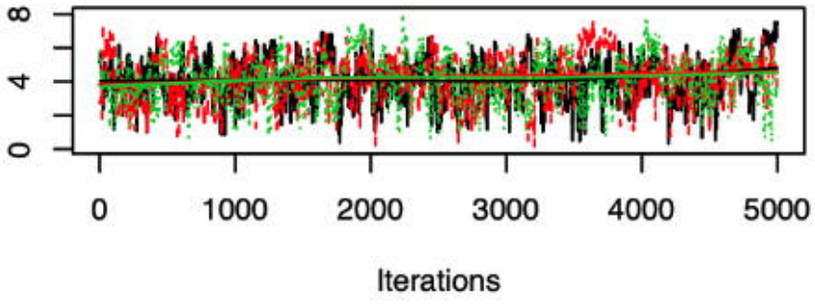
(c) μ

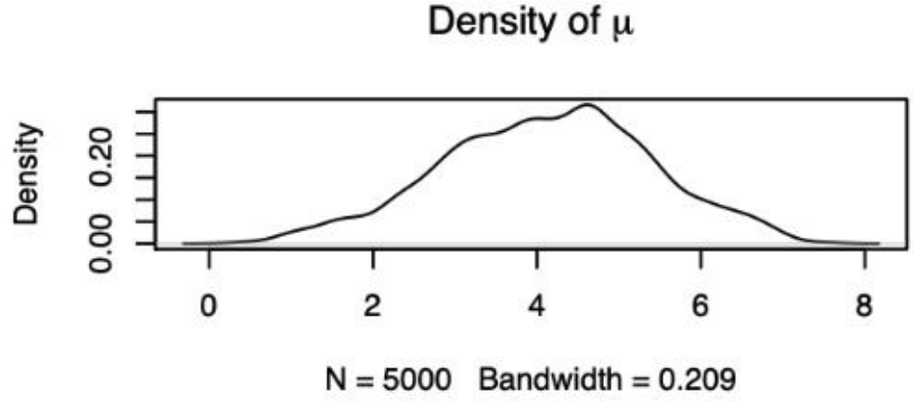


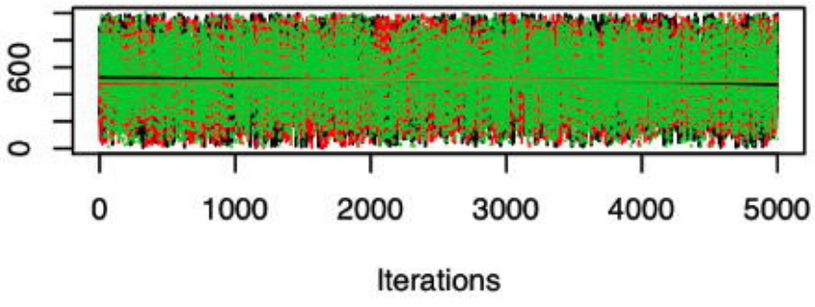
(d) π

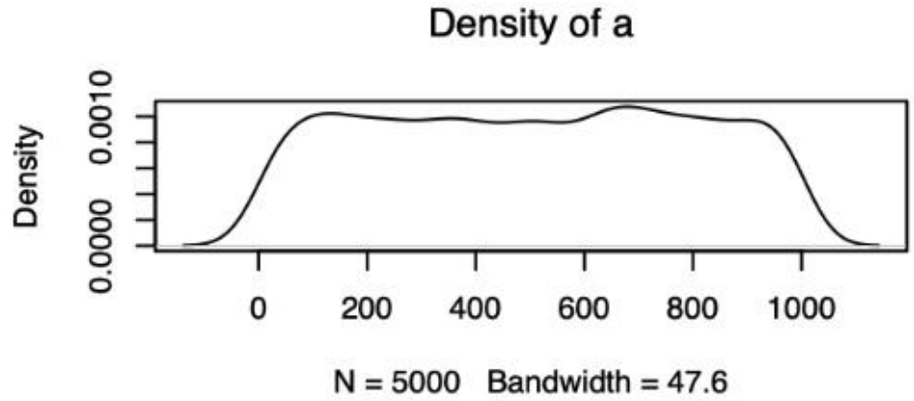

0
600
0
1000
2000
3000
4000
5000
Iterations

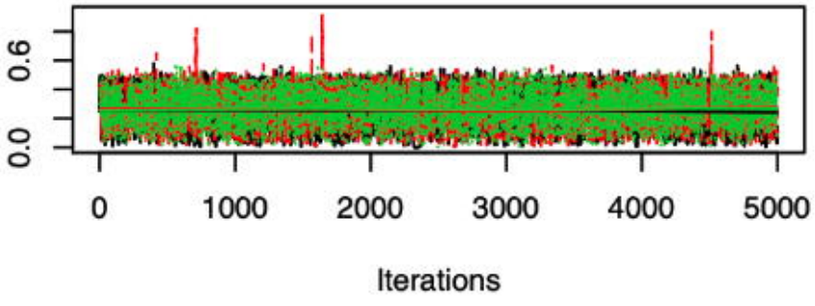

Density of α
Density
0.000
0.004
0
200
400
600
800
1000
N = 5000 Bandwidth = 27.04

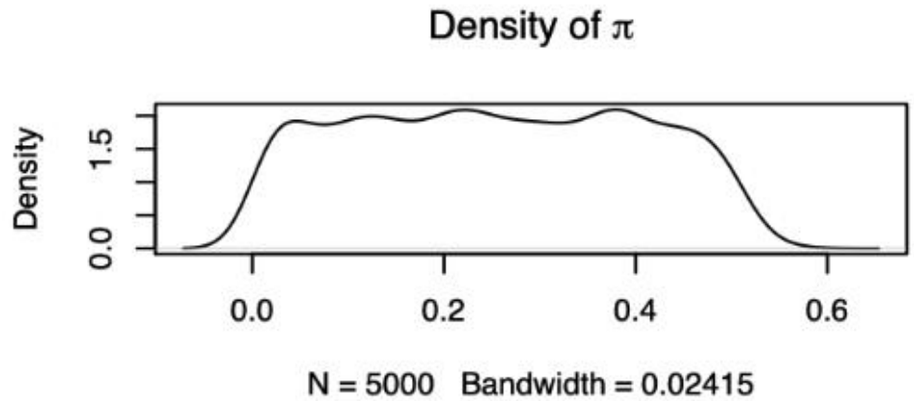

0
4
8
0
1000
2000
3000
4000
5000
Iterations

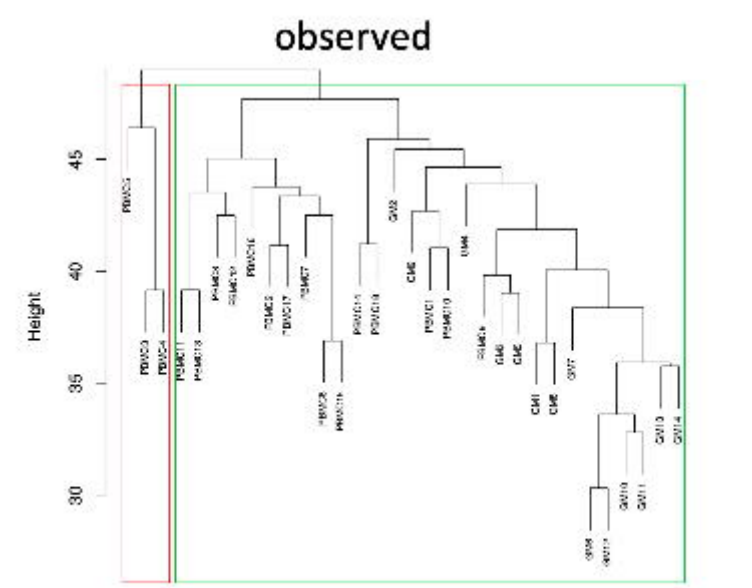

Density of μ
Density
0.00
0.20
0
2
4
6
8
N = 5000 Bandwidth = 0.209

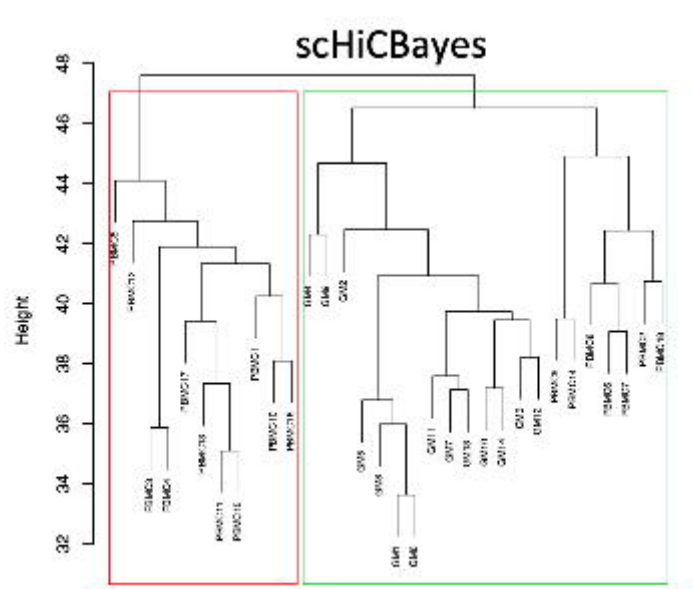

0
600
0
1000
2000
3000
4000
5000
Iterations

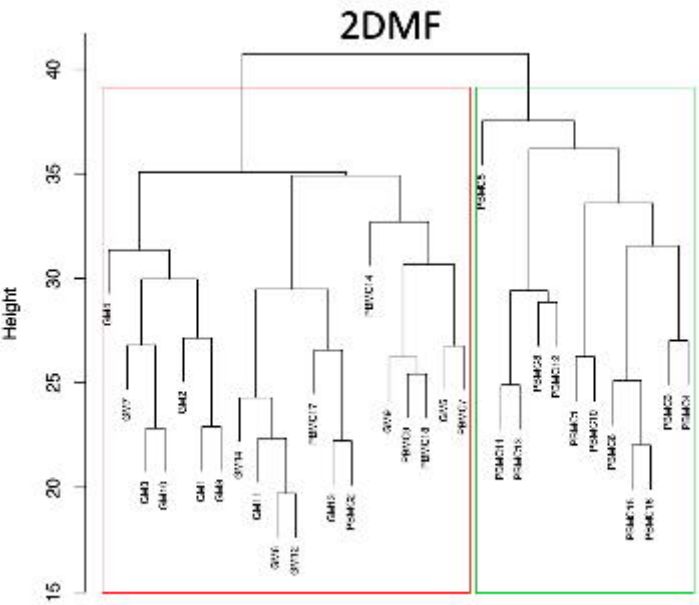

Density of a
Density
0.0000
0.0010
0
200
400
600
800
1000
N = 5000 Bandwidth = 47.6

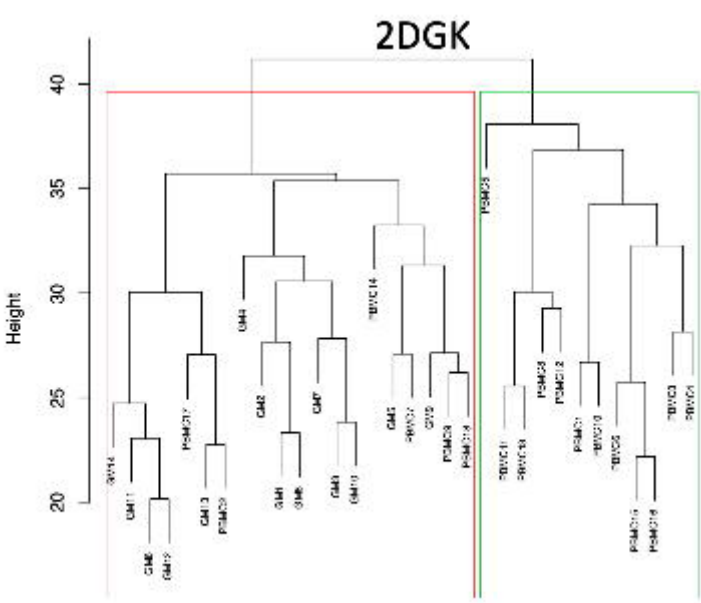

0.0
0.6
0
1000
2000
3000
4000
5000
Iterations

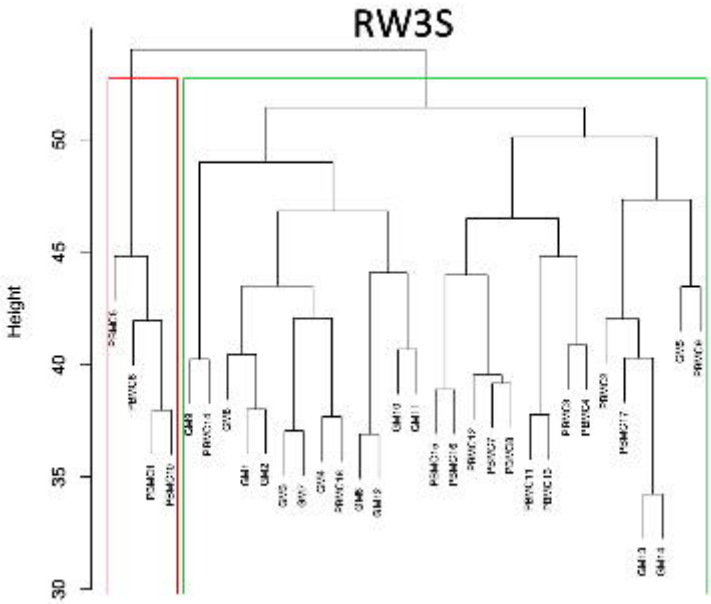

Density of π
Density
0.0
1.5
0.0
0.2
0.4
0.6
N = 5000 Bandwidth = 0.02415

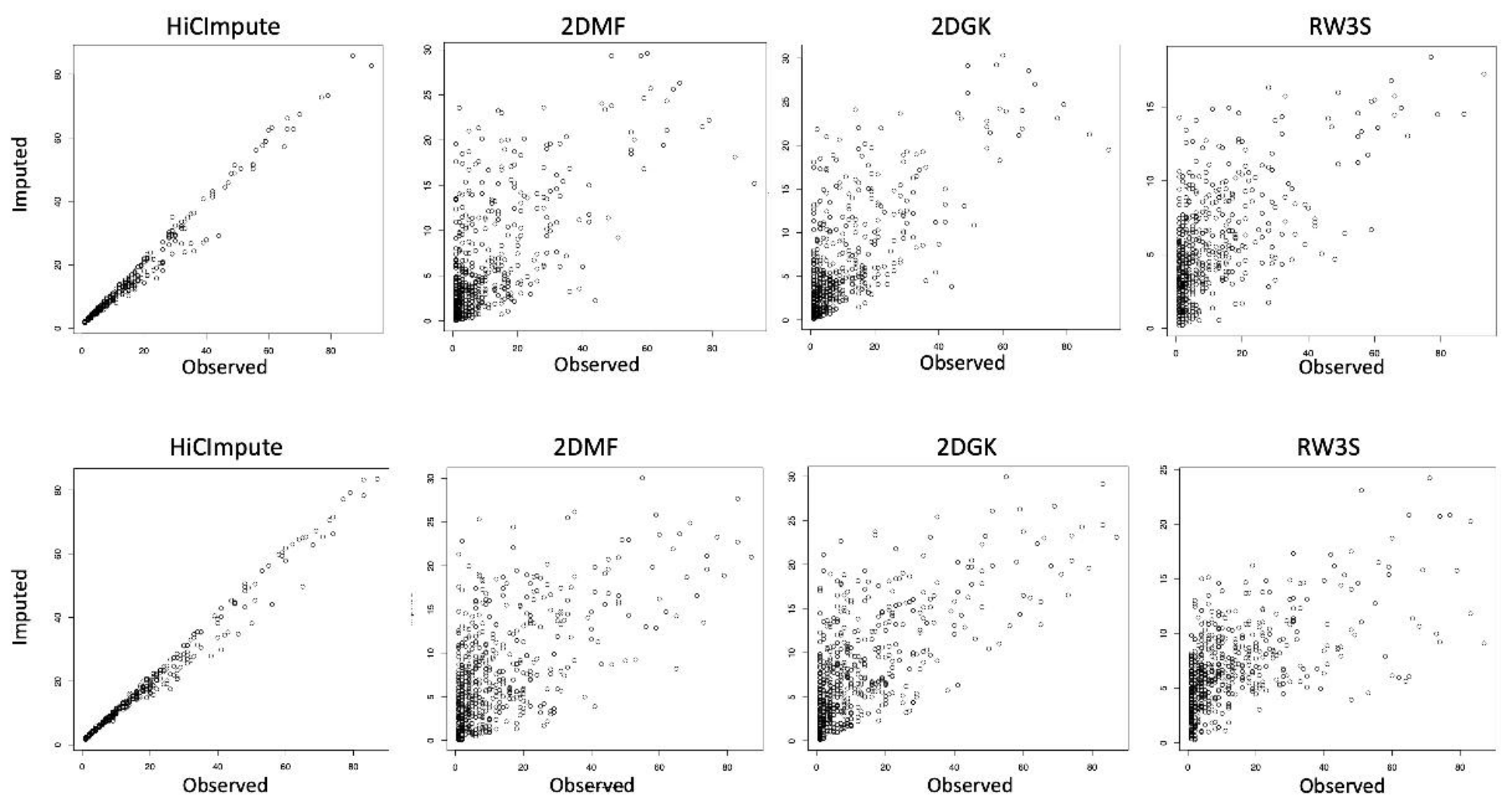

observed
Height

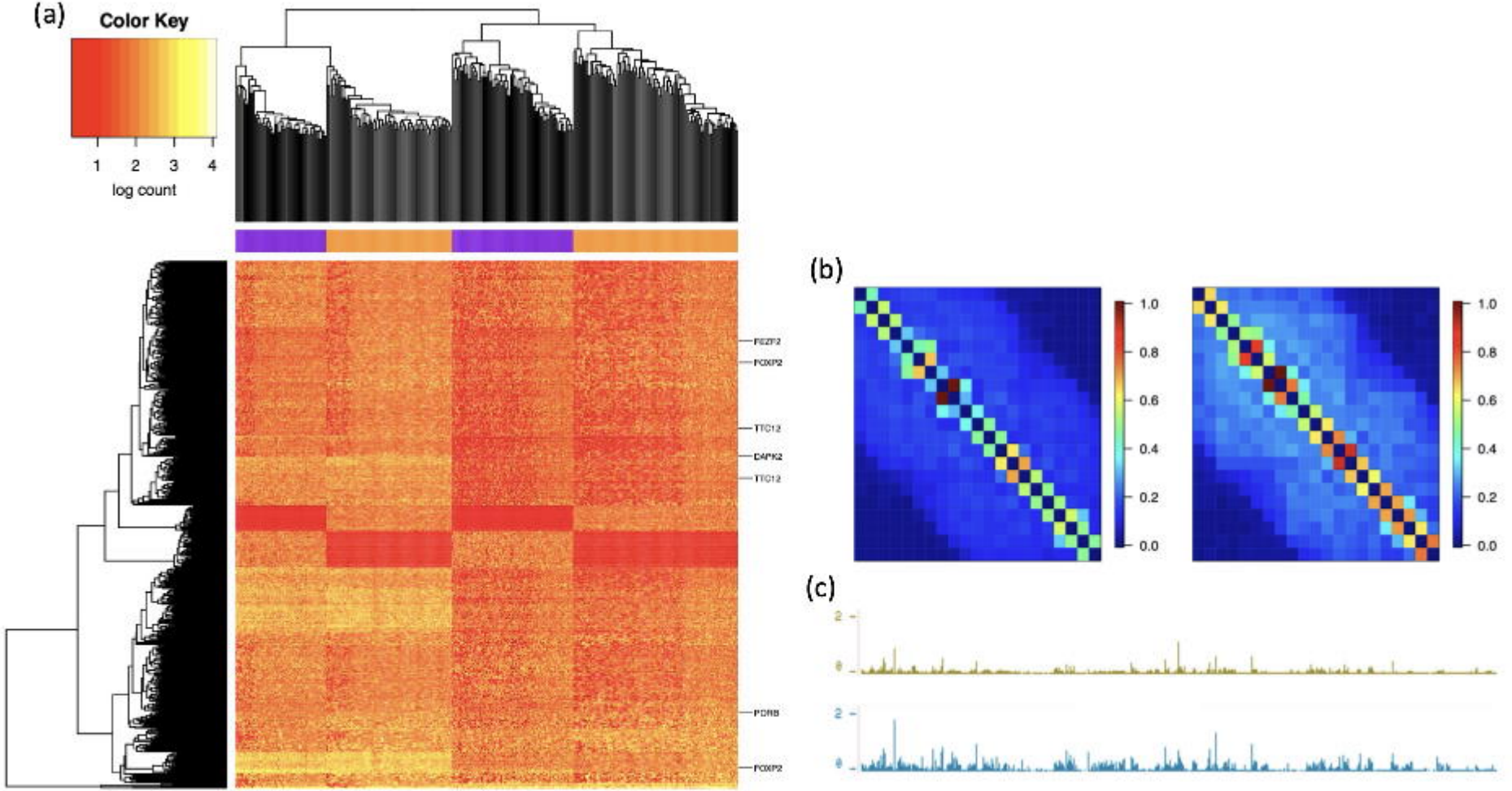

scHiCBayes
Height

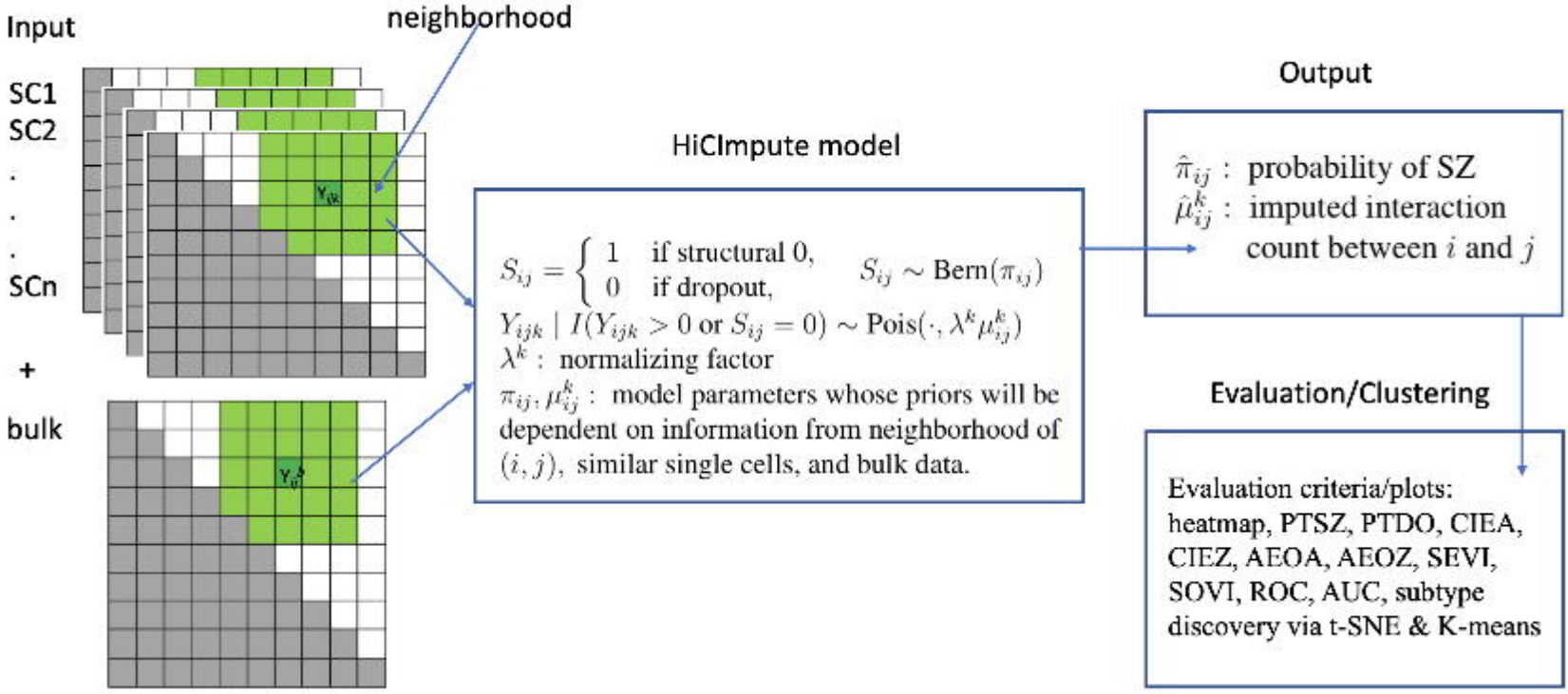

2DMF
Height

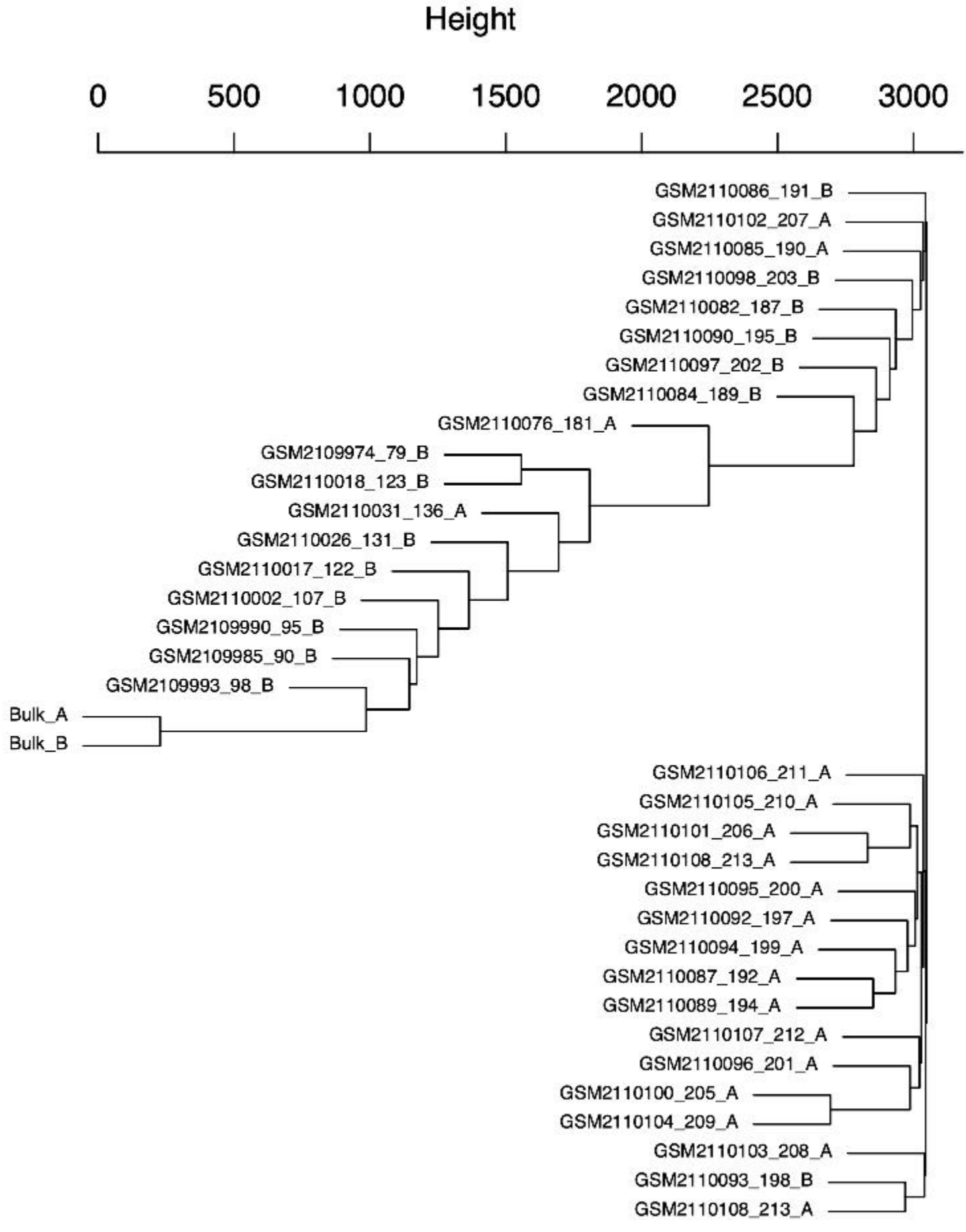

2DGK
Height

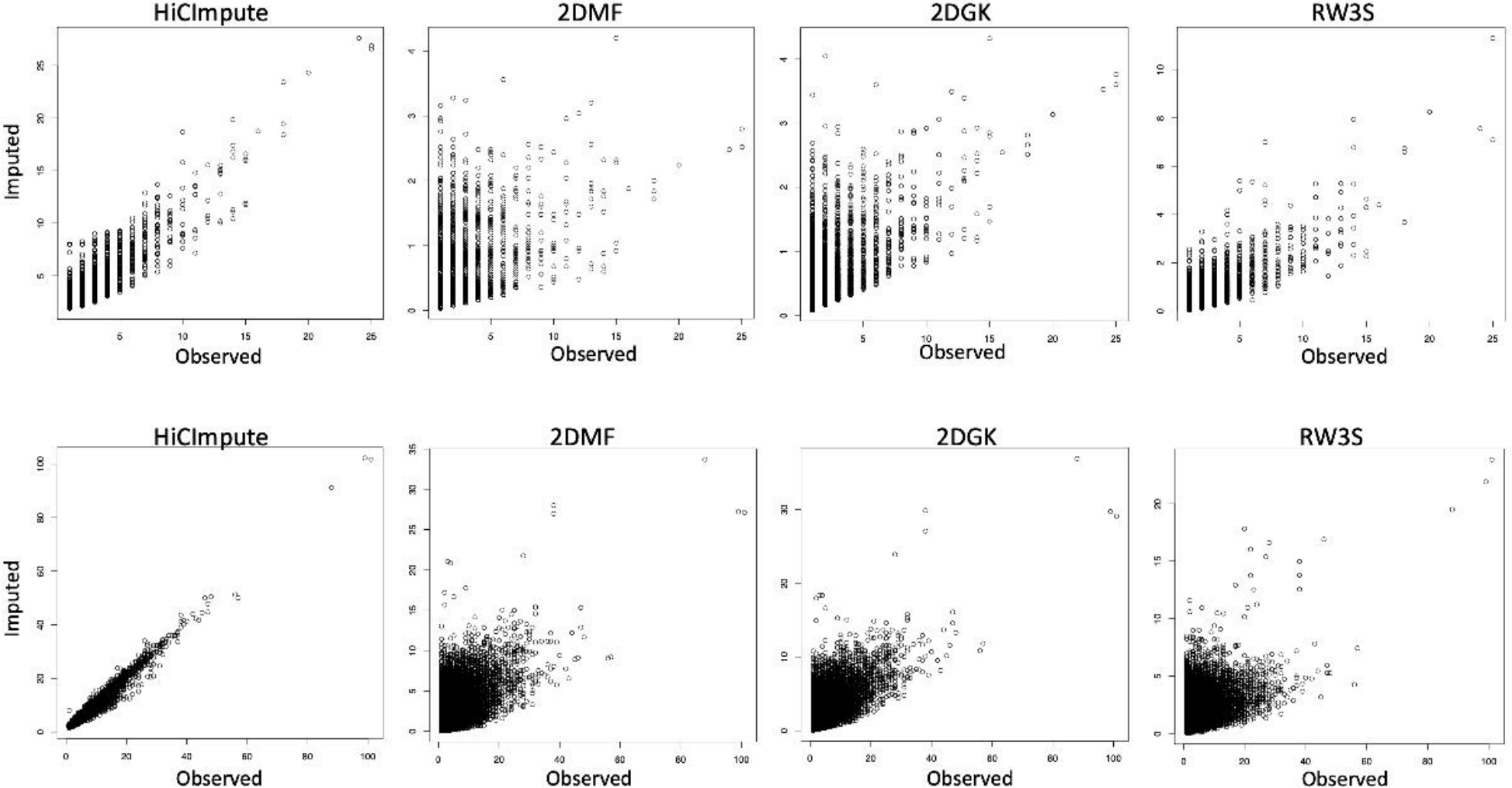

RW3S
Height

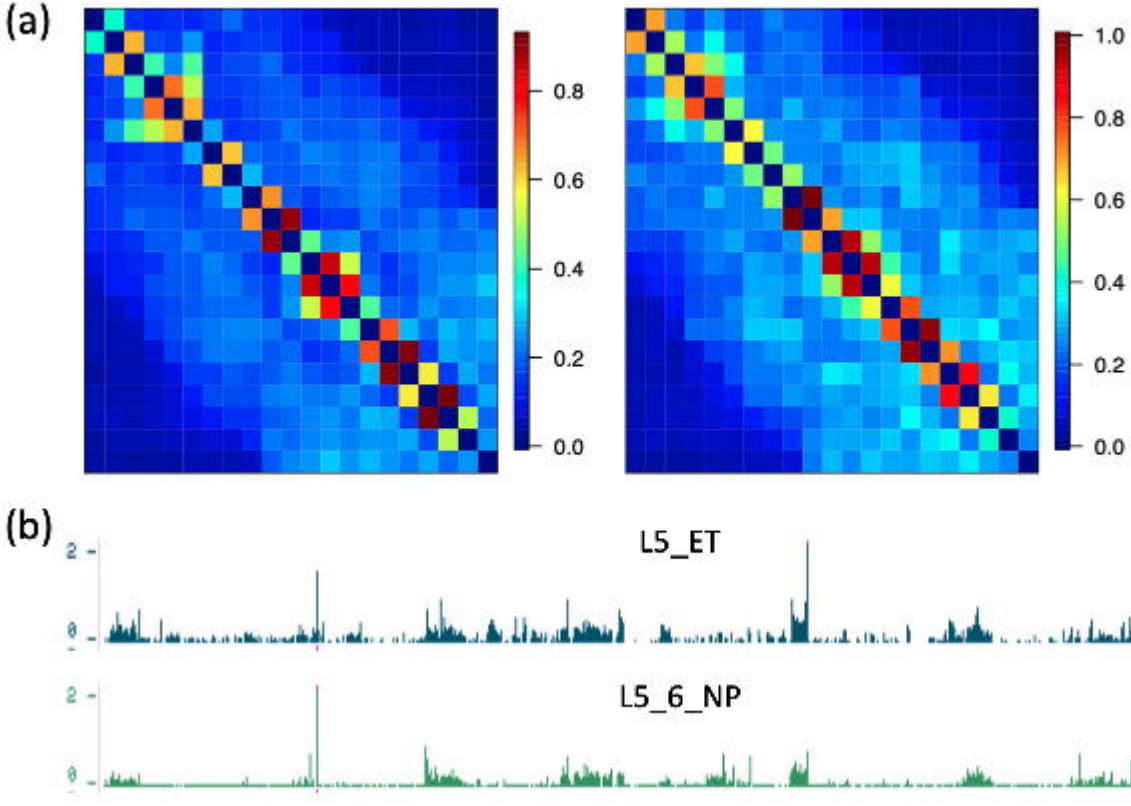

HiCImpute
2DMF
2DGK
RW3S
Imputed
Observed
HiCImpute
2DMF
2DGK
RW3S
Imputed
Observed

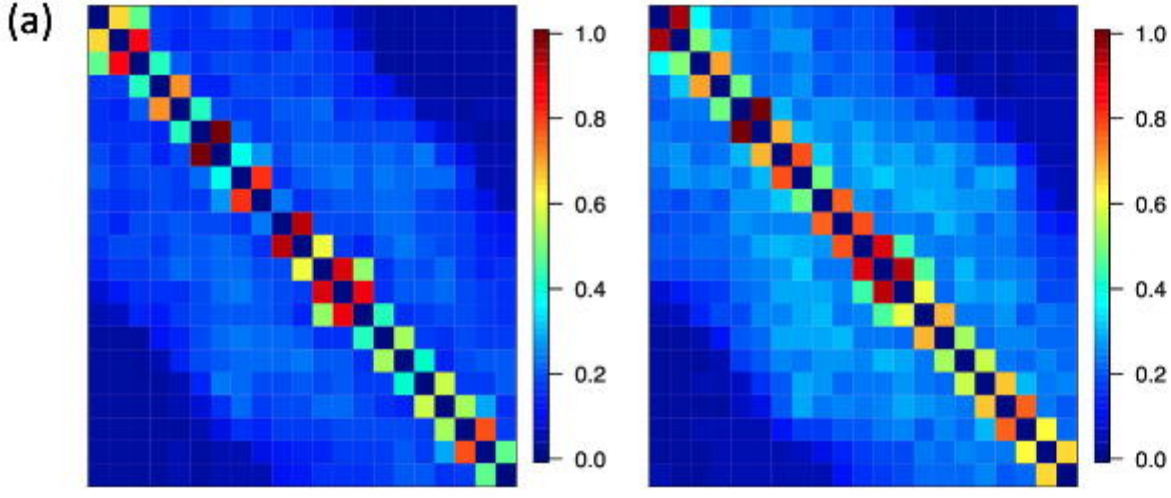

(a)
Color Key
1
2
3
4
log count
FEZF2
FOXP2
TTC12
DAPK2
TTC12
PDRG
FOXP2
(b)
1.0
0.8
0.6
0.4
0.2
0.0
(c)
2
0
2
0

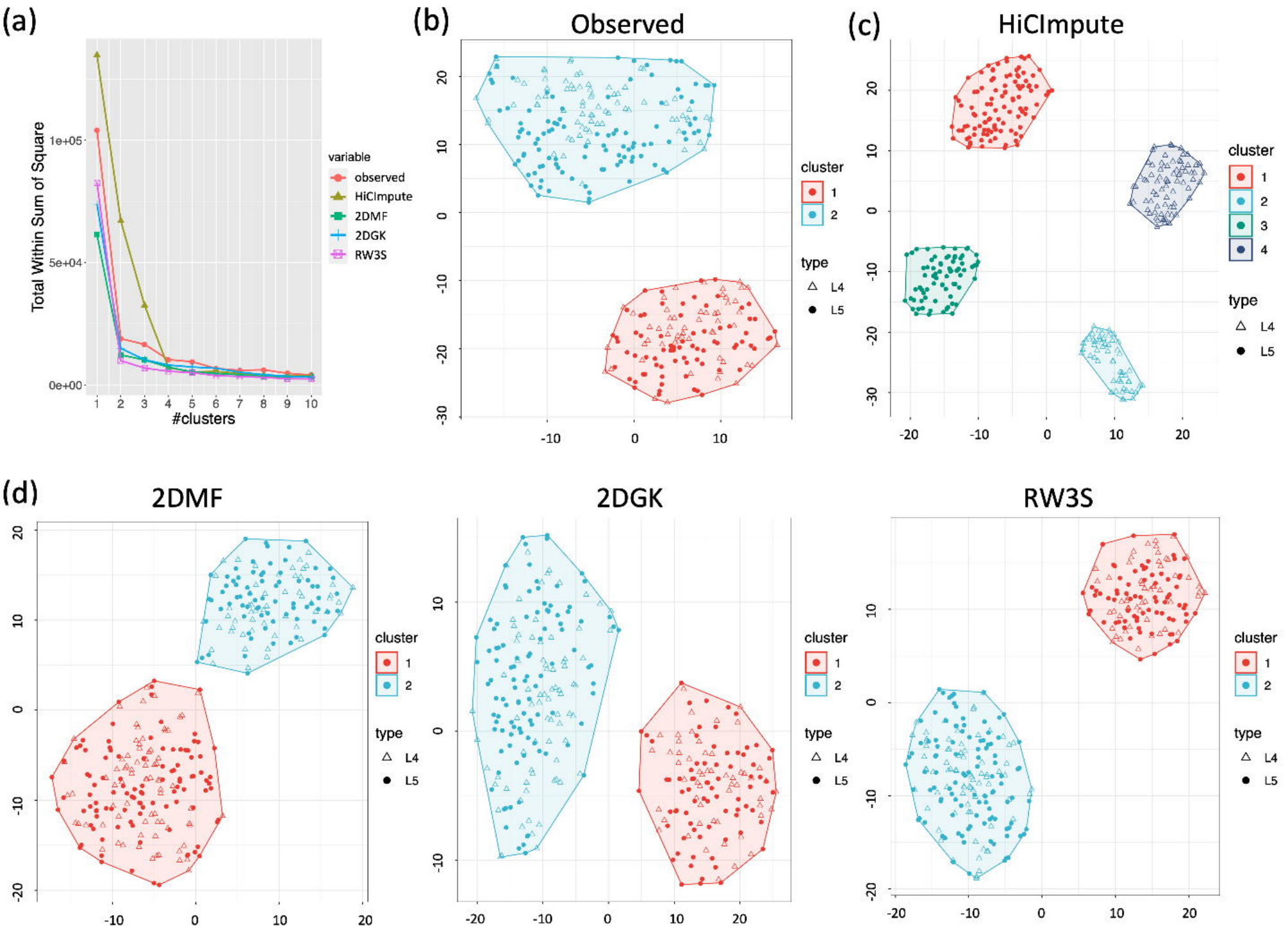

Input
neighborhood
SC1
SC2
.
.
.
SCn
+
bulk
HiCImpute model
$S_{ij} = \begin{cases} 1 & \text{if structural 0,} \\ 0 & \text{if dropout,} \end{cases} \quad S_{ij} \sim \text{Bern}(\pi_{ij})$
$Y_{ijk} \mid I(Y_{ijk} > 0 \text{ or } S_{ij} = 0) \sim \text{Pois}(\cdot, \lambda^k \mu_{ij}^k)$
λ^k : normalizing factor
π_{ij}, μ_{ij}^k : model parameters whose priors will be dependent on information from neighborhood of (i, j), similar single cells, and bulk data.
Output
$\hat{\pi}_{ij}$: probability of SZ
$\hat{\mu}_{ij}^k$: imputed interaction count between i and j
Evaluation/Clustering
Evaluation criteria/plots: heatmap, PTSZ, PTDO, CIEA, CIEZ, AEOA, AEOZ, SEVI, SOVI, ROC, AUC, subtype discovery via t-SNE & K-means

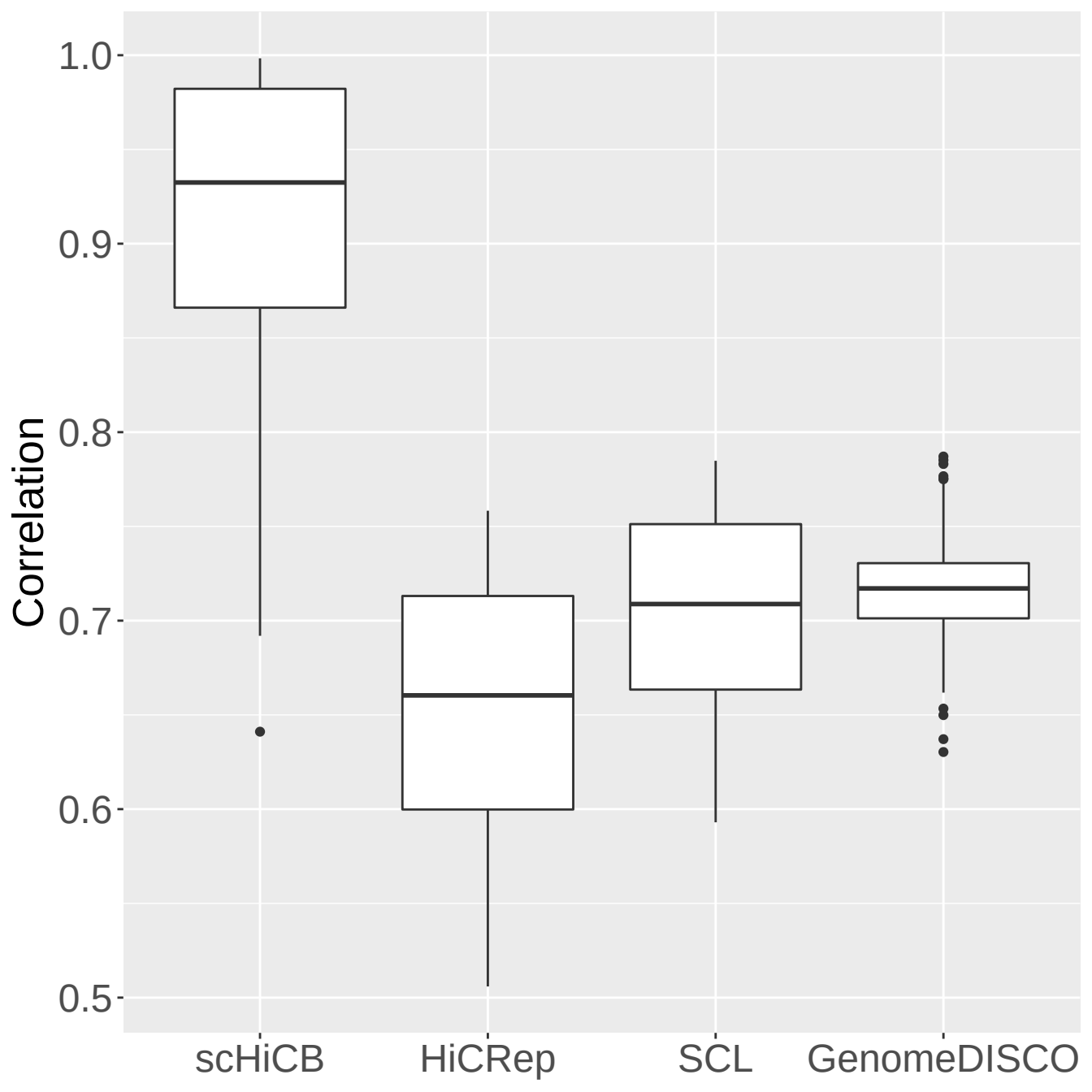

Height
0
500
1000
1500
2000
2500
3000
GSM2110086_191_B
GSM2110102_207_A
GSM2110085_190_A
GSM2110098_203_B
GSM2110082_187_B
GSM2110090_195_B
GSM2110097_202_B
GSM2110084_189_B
GSM2110076_181_A
GSM2109974_79_B
GSM2110018_123_B
GSM2110031_136_A
GSM2110026_131_B
GSM2110017_122_B
GSM2110002_107_B
GSM2109990_95_B
GSM2109985_90_B
GSM2109993_98_B
Bulk_A
Bulk_B
GSM2110106_211_A
GSM2110105_210_A
GSM2110101_206_A
GSM2110108_213_A
GSM2110095_200_A
GSM2110092_197_A
GSM2110094_199_A
GSM2110087_192_A
GSM2110089_194_A
GSM2110107_212_A
GSM2110096_201_A
GSM2110100_205_A
GSM2110104_209_A
GSM2110103_208_A
GSM2110093_198_B
GSM2110108_213_A

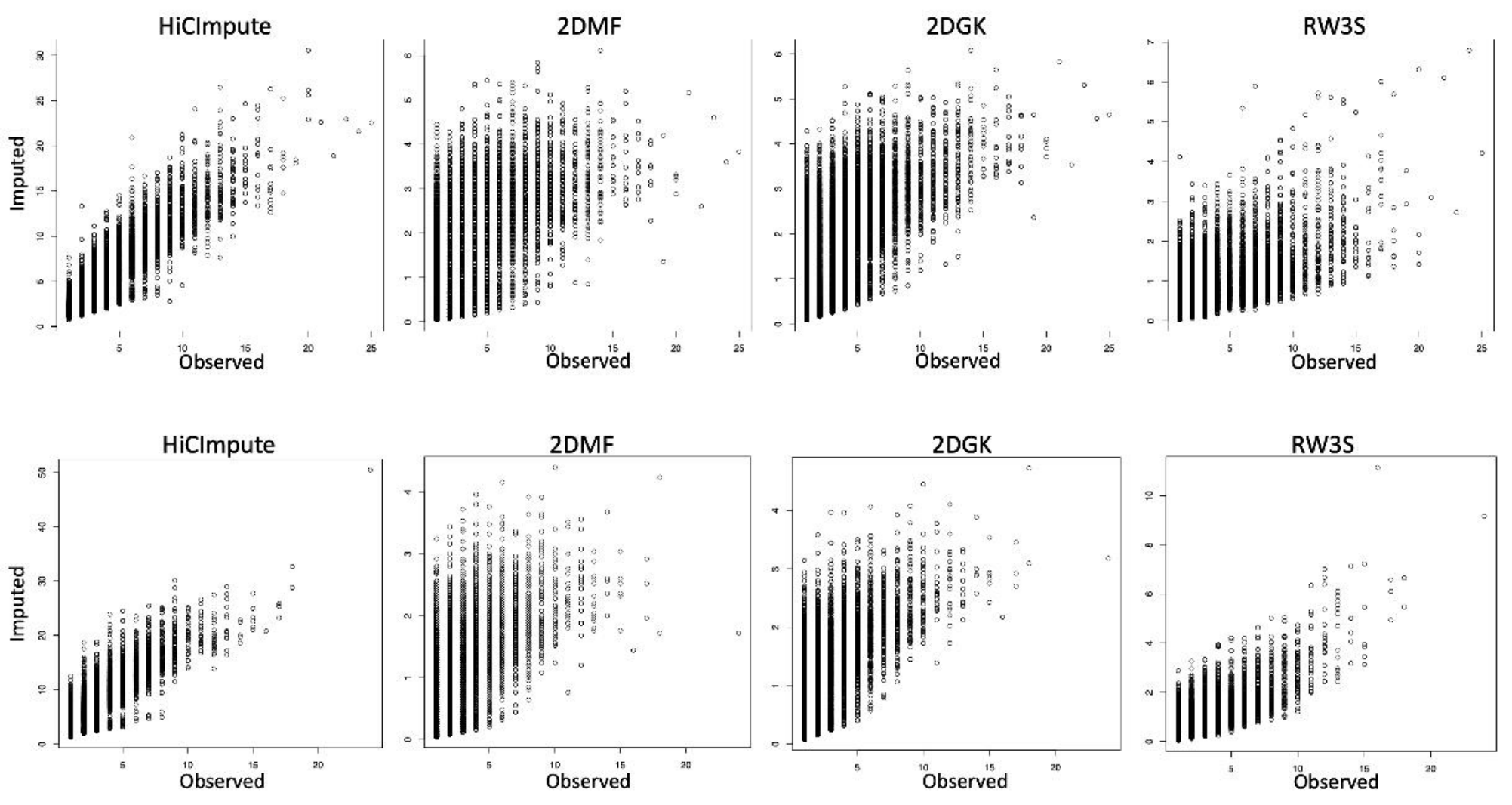

HiCImpute
2DMF
2DGK
RW3S
Imputed
Observed
HiCImpute
2DMF
2DGK
RW3S
Imputed
Observed

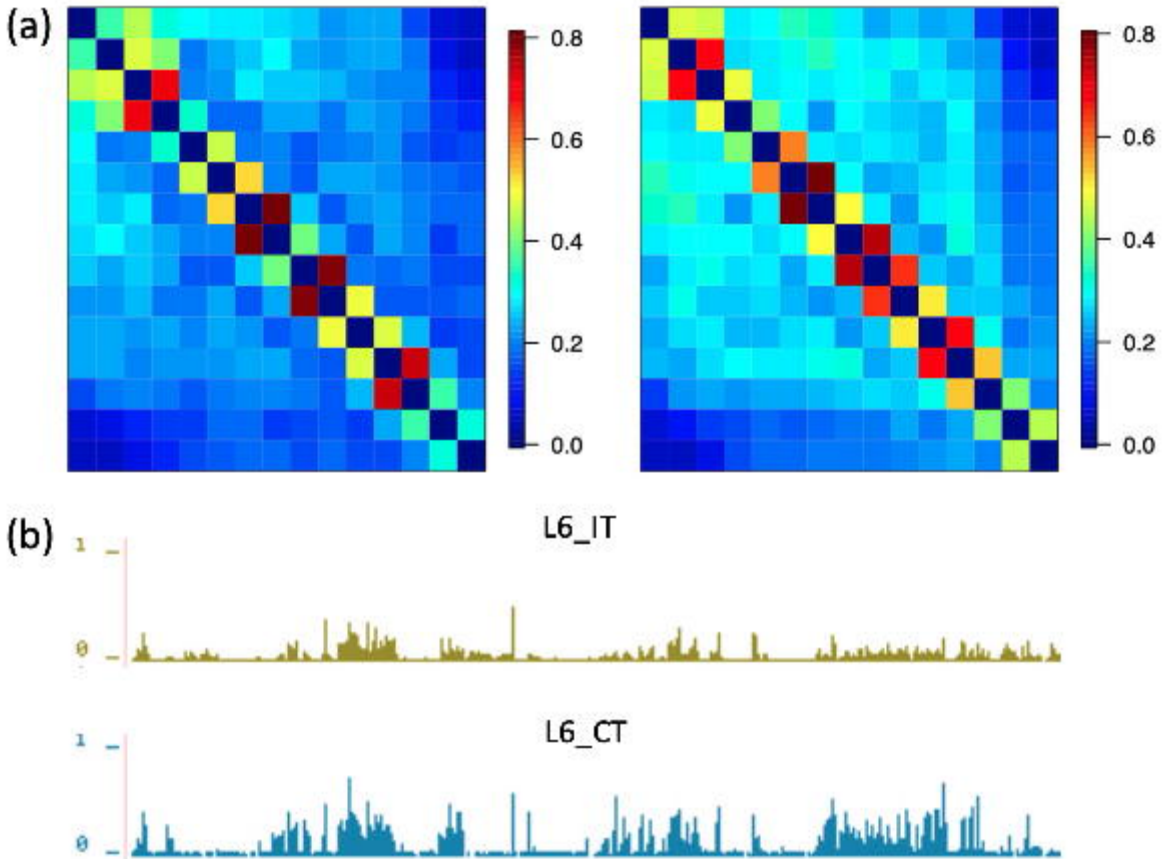

(a)
0.8
0.6
0.4
0.2
0.0
1.0
0.8
0.6
0.4
0.2
0.0
(b)
L5_ET
2
0
L5_6_NP
2
0


(a)
1.0
0.8
0.6
0.4
0.2
0.0
1.0
0.8
0.6
0.4
0.2
0.0
(b)
L5_ET
2
0
L5_6_NP
2
0

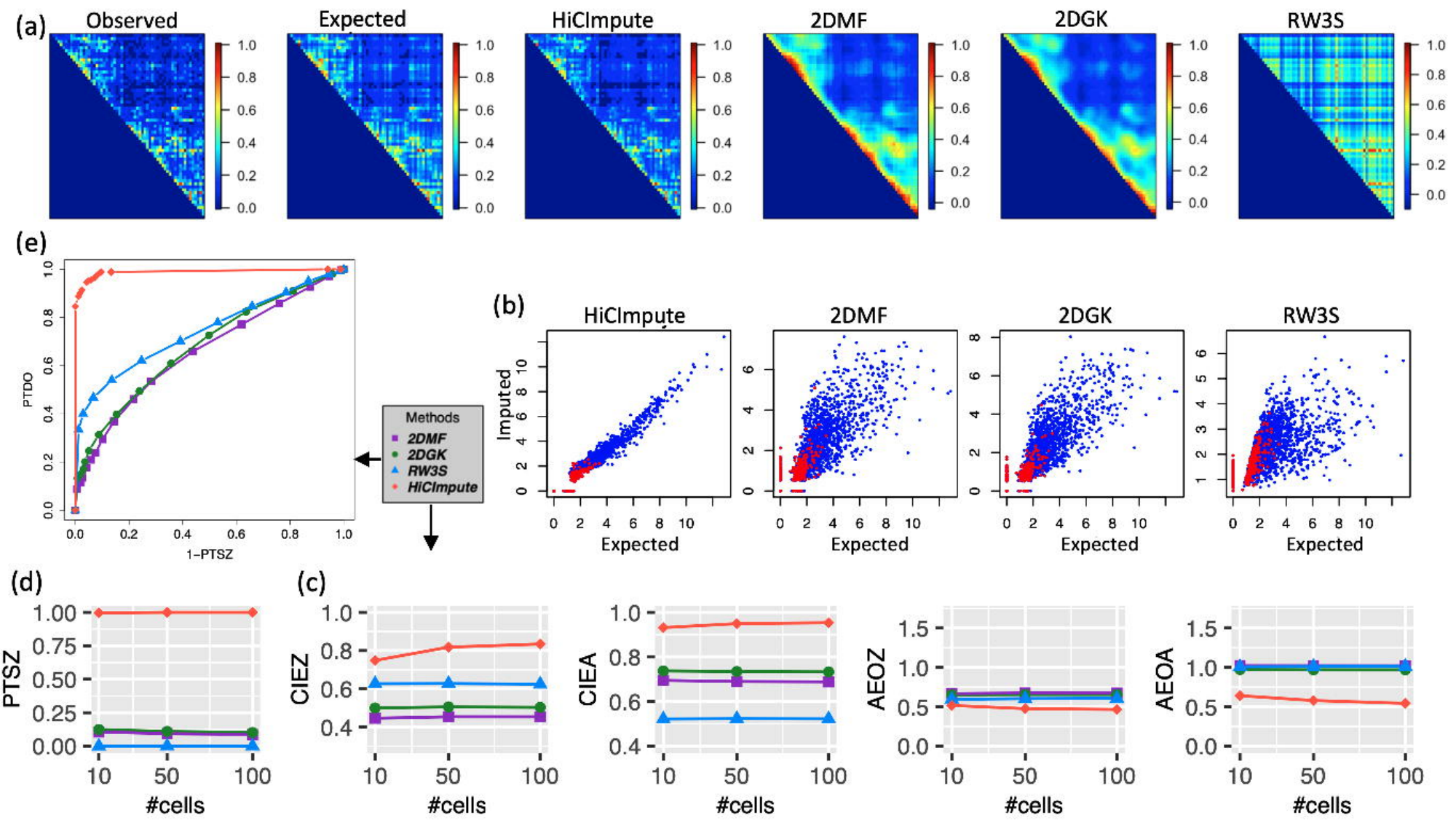

(a)
Total Within Sum of Square
1e+05
5e+04
0e+00
1 2 3 4 5 6 7 8 9 10
#clusters
variable
observed
HiCImpute
2DMF
2DGK
RW3S
(b)
Observed
cluster
1
2
type
L4
L5
(c)
HiCImpute
cluster
1
2
3
4
type
L4
L5
(d)
2DMF
2DGK
RW3S

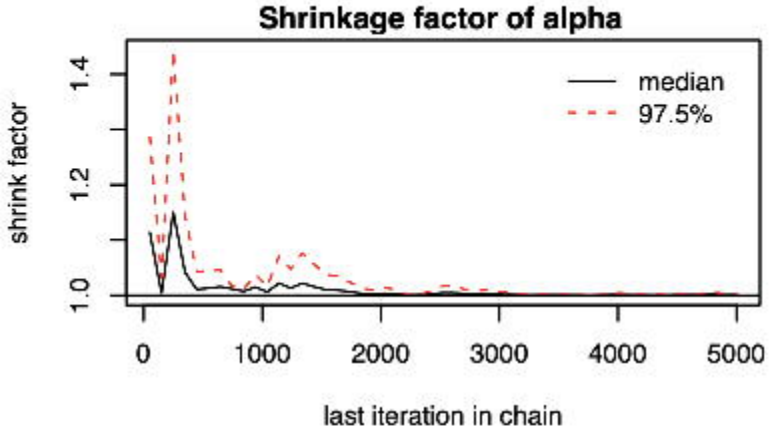

1.0
0.9
0.8
0.7
0.6
0.5
Correlation
scHiCB
HiCRep
SCL
GenomeDISCO

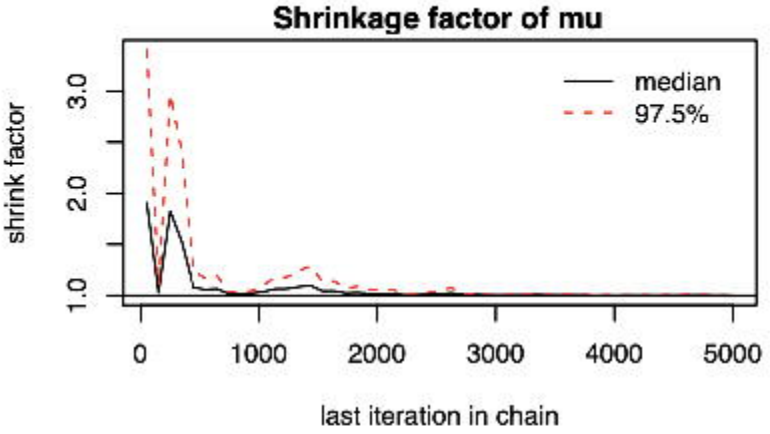

HiCImpute
2DMF
2DGK
RW3S
Imputed
Observed
HiCImpute
2DMF
2DGK
RW3S
Imputed
Observed

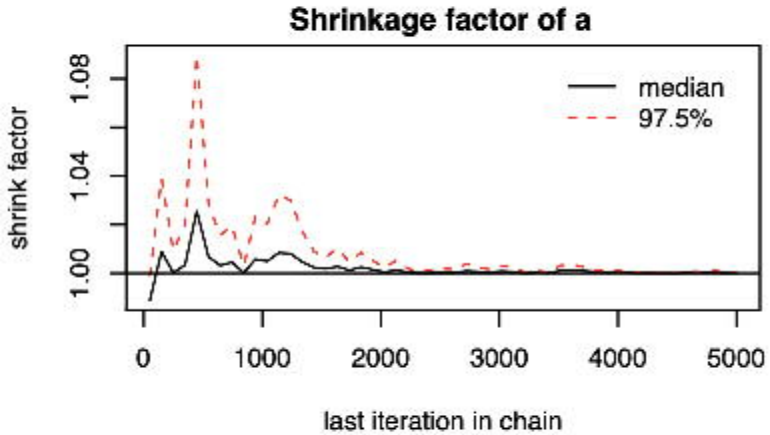

(a)
0.8
0.6
0.4
0.2
0.0
0.8
0.6
0.4
0.2
0.0
(b)
L6_IT
1
0
L6_CT
1
0

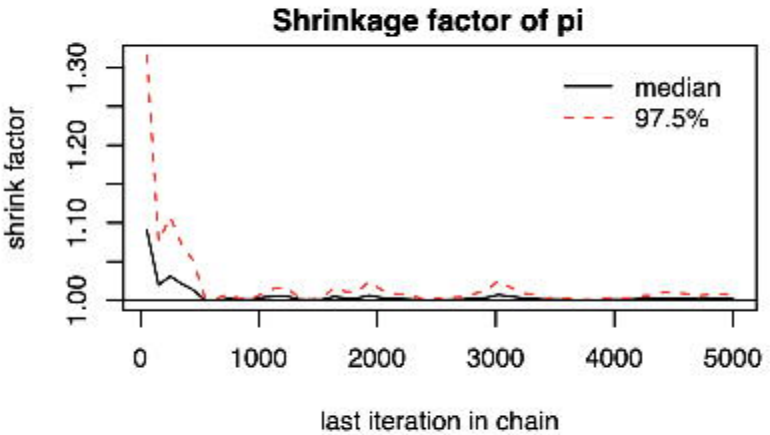

Methods
2DMF
2DGK
RW3S
HiCImpute

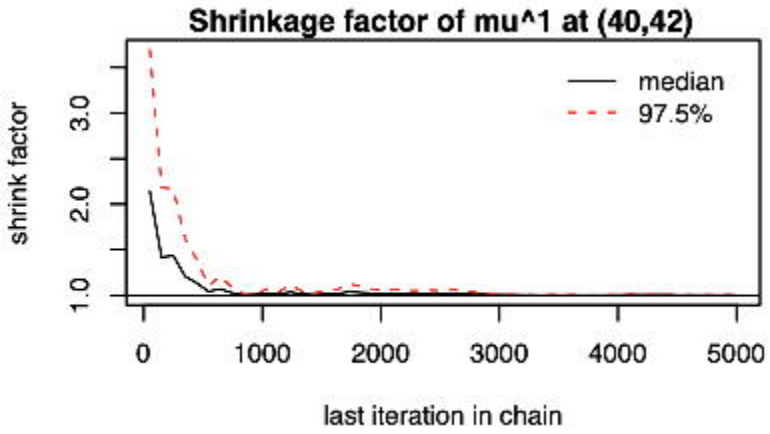

(a)
Observed
Expected
HiClmpute
2DMF
2DGK
RW3S
(e)
PTDO
1−PTSZ
Methods
2DMF
2DGK
RW3S
HiClmpute
(b)
HiClmpute
2DMF
2DGK
RW3S
Imputed
Expected
(d)
PTSZ
#cells
(c)
CIEZ
CIEA
AEOZ
AEOA
#cells

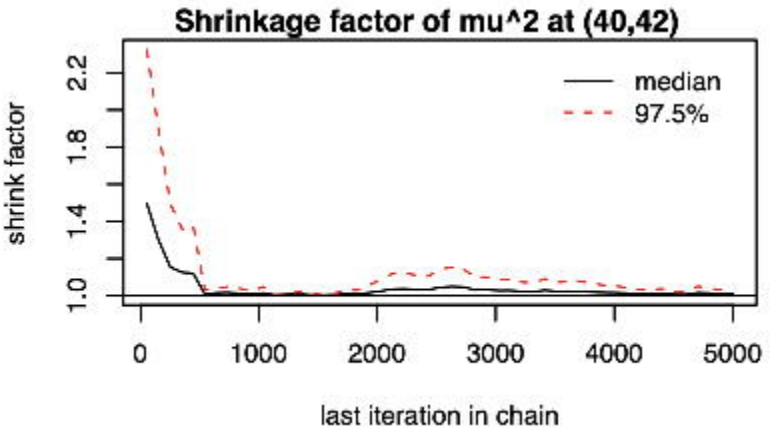

Shrinkage factor of alpha
median
97.5%
shrink factor
1.0
1.2
1.4
0
1000
2000
3000
4000
5000
last iteration in chain

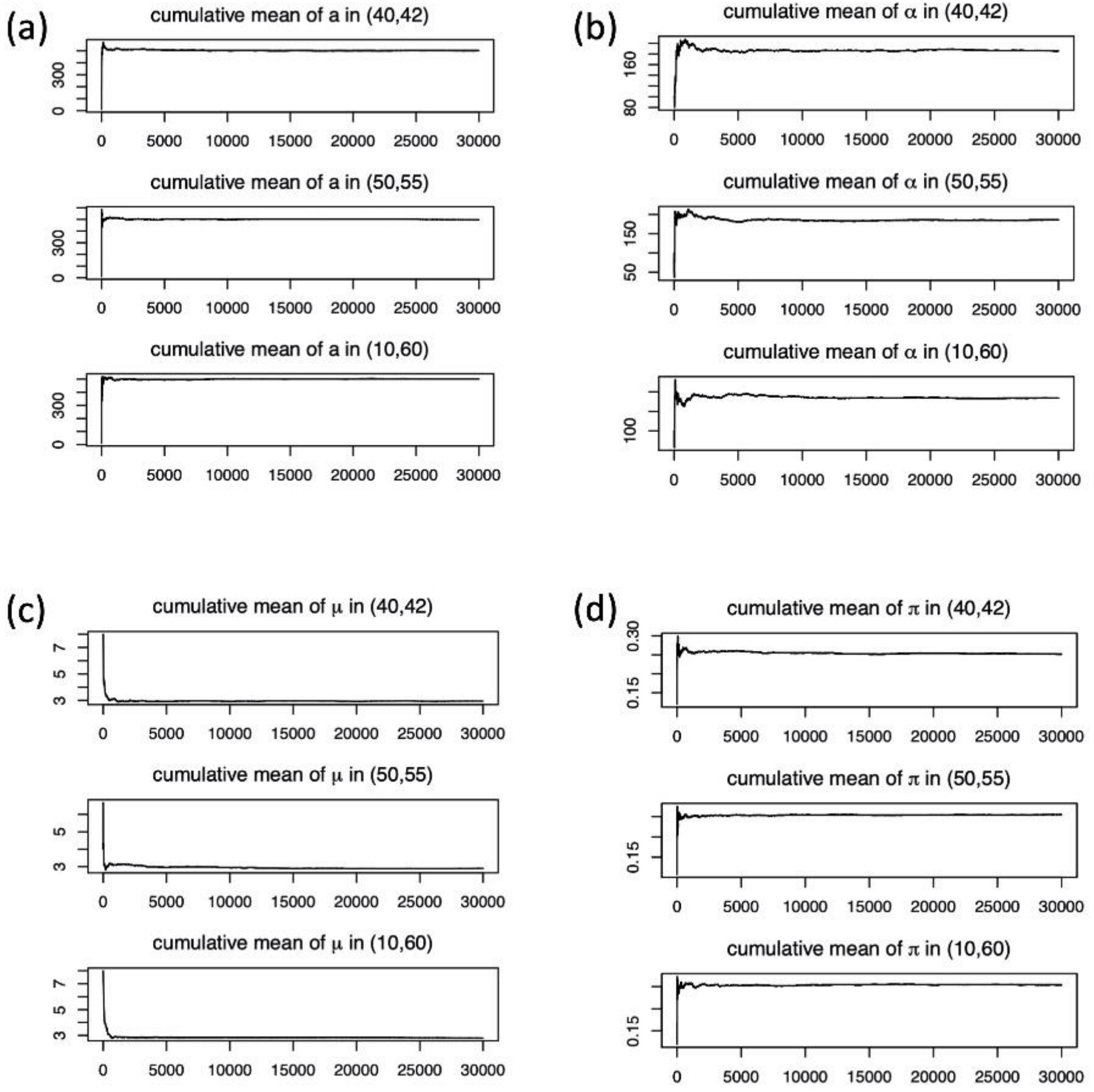

Shrinkage factor of mu
median
97.5%
shrink factor
1.0
2.0
3.0
0
1000
2000
3000
4000
5000
last iteration in chain

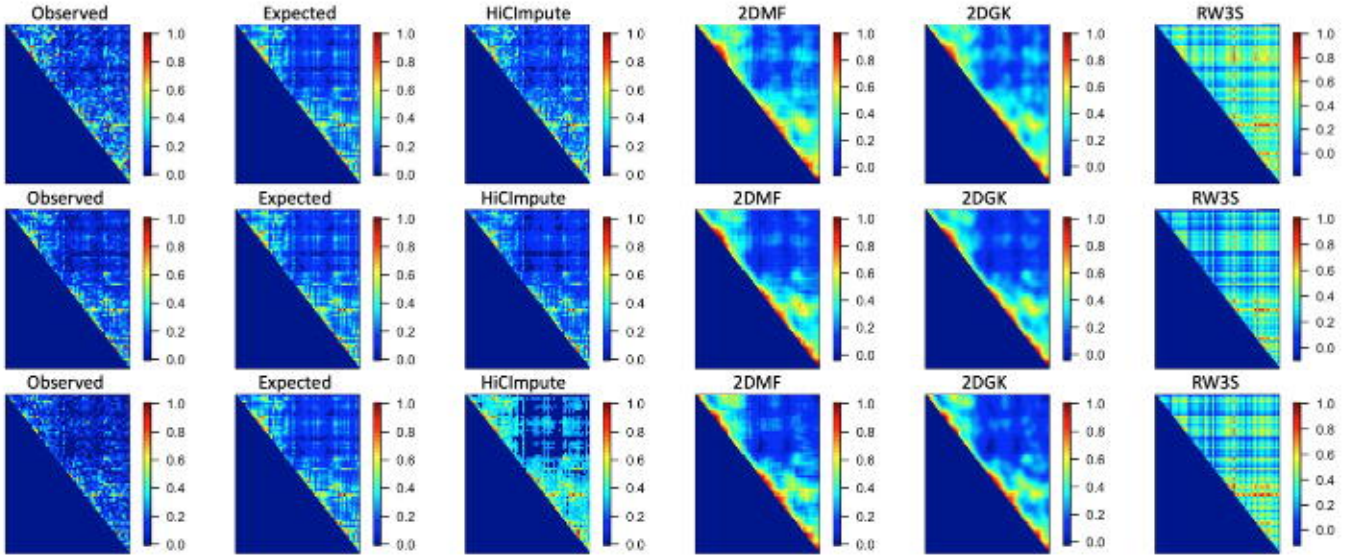

Shrinkage factor of a
median
97.5%
shrink factor
1.00
1.04
1.08
0
1000
2000
3000
4000
5000
last iteration in chain


Shrinkage factor of pi
median
97.5%
shrink factor
1.00
1.10
1.20
1.30
0
1000
2000
3000
4000
5000
last iteration in chain

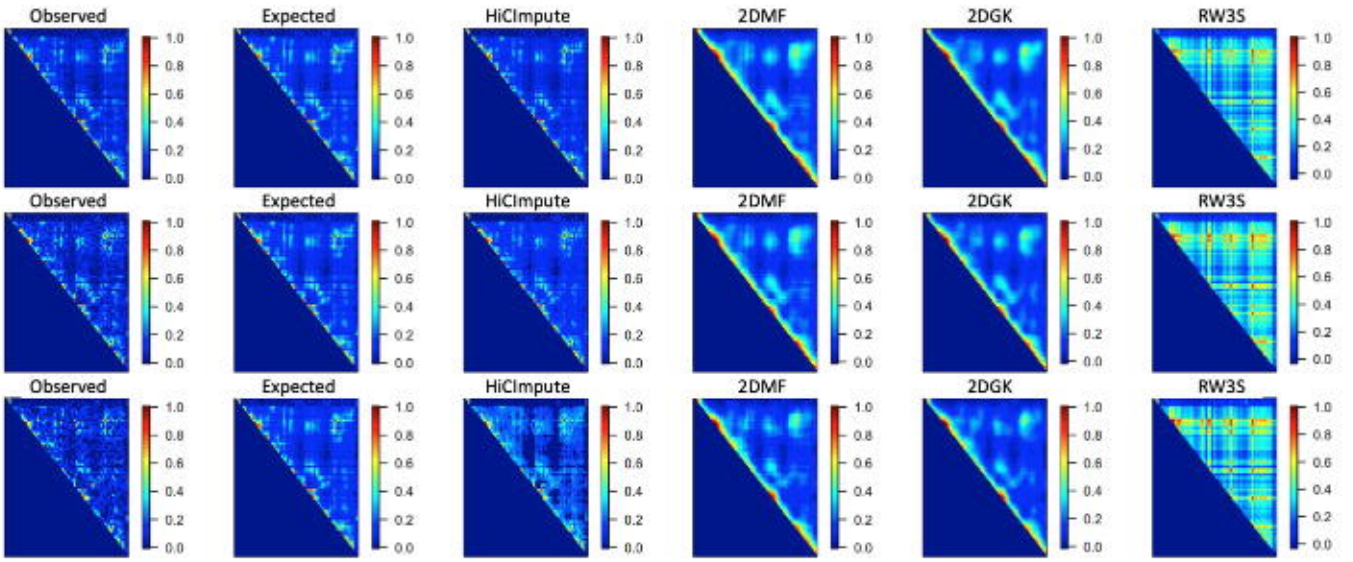

Shrinkage factor of mu^1 at (40,42)
median
97.5%
shrink factor
1.0
2.0
3.0
0
1000
2000
3000
4000
5000
last iteration in chain


Shrinkage factor of mu^2 at (40,42)
median
97.5%
shrink factor
1.0
1.4
1.8
2.2
0
1000
2000
3000
4000
5000
last iteration in chain

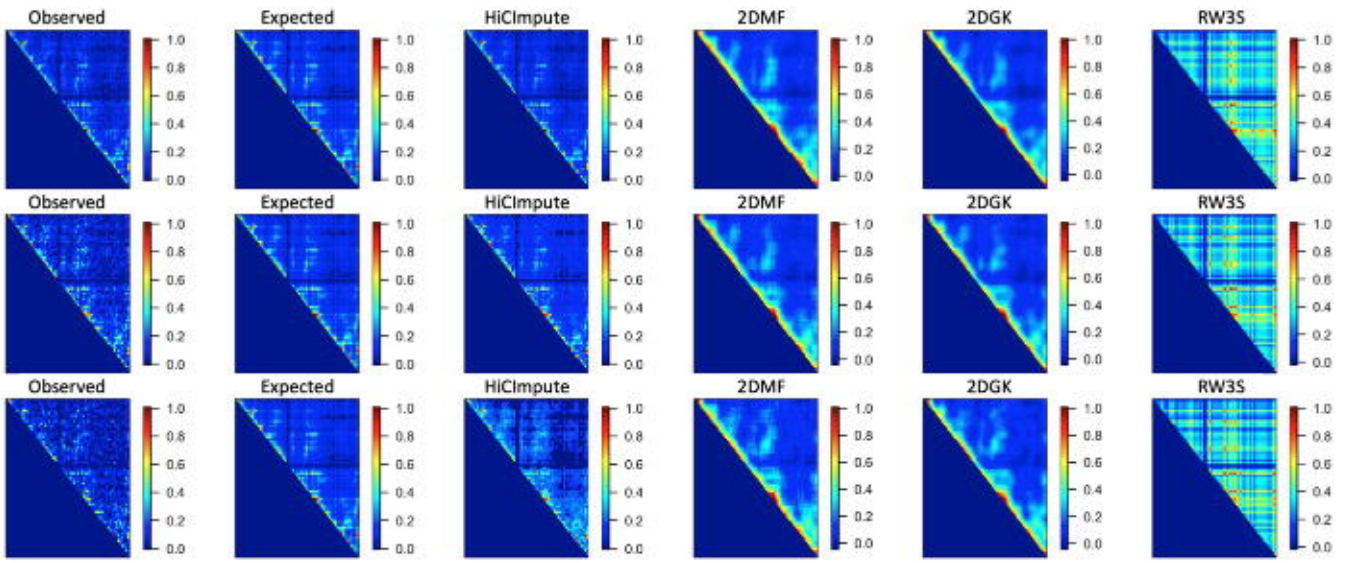

(a)
cumulative mean of a in (40,42)
0
300
0
5000
10000
15000
20000
25000
30000
cumulative mean of a in (50,55)
0
300
0
5000
10000
15000
20000
25000
30000
cumulative mean of a in (10,60)
0
300
0
5000
10000
15000
20000
25000
30000
(b)
cumulative mean of α in (40,42)
80
160
0
5000
10000
15000
20000
25000
30000
cumulative mean of α in (50,55)
50
150
0
5000
10000
15000
20000
25000
30000
cumulative mean of α in (10,60)
100
0
5000
10000
15000
20000
25000
30000
(c)
cumulative mean of μ in (40,42)
3
5
7
0
5000
10000
15000
20000
25000
30000
cumulative mean of μ in (50,55)
3
5
0
5000
10000
15000
20000
25000
30000
cumulative mean of μ in (10,60)
3
5
7
0
5000
10000
15000
20000
25000
30000
(d)
cumulative mean of π in (40,42)
0.15
0.30
0
5000
10000
15000
20000
25000
30000
cumulative mean of π in (50,55)
0.15
0
5000
10000
15000
20000
25000
30000
cumulative mean of π in (10,60)
0.15
0
5000
10000
15000
20000
25000
30000



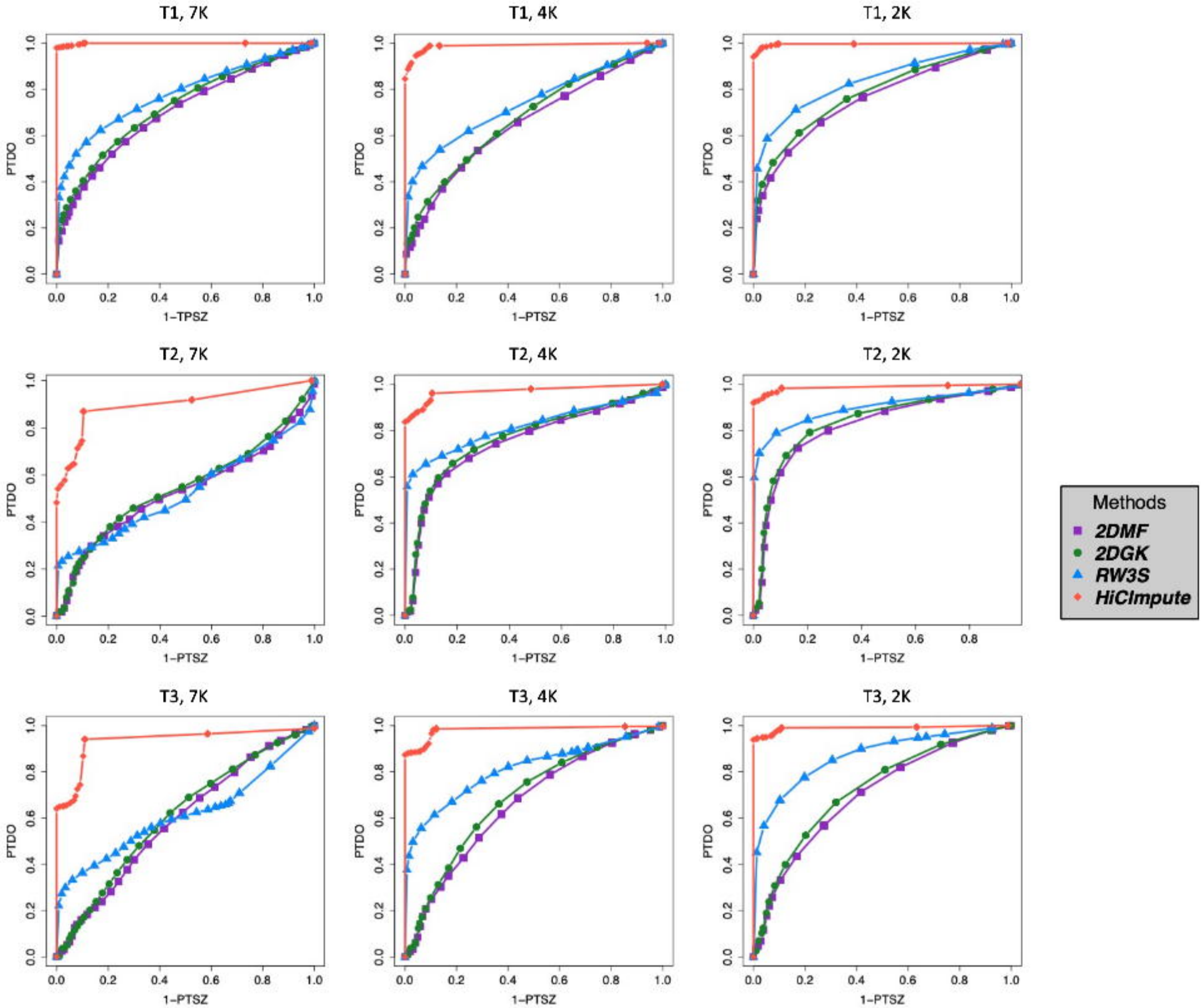
(a) T1, 7k (top), 4k (middle), and 2k (bottom)

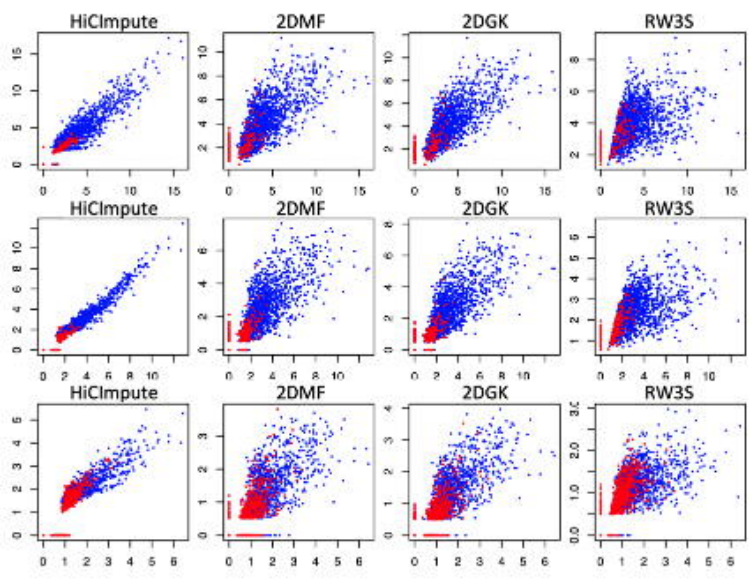


(b) T2, 7k (top), 4k (middle), and 2k (bottom)

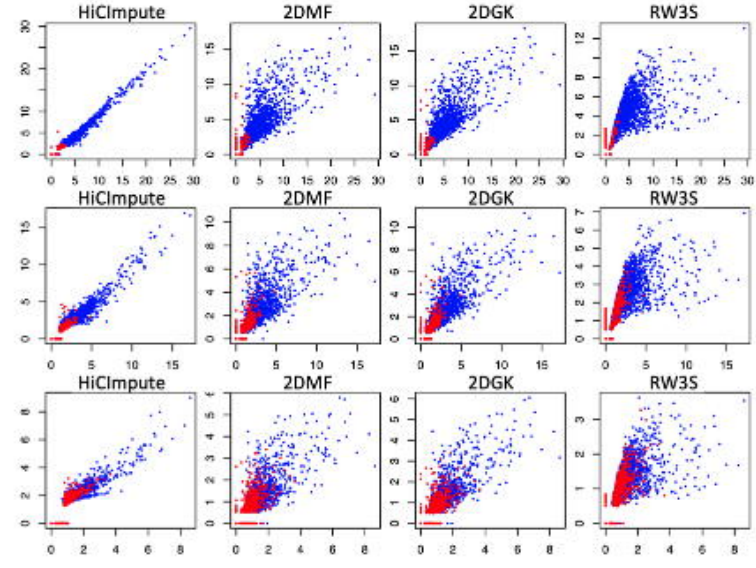


(c) T3, 7k (top), 4k (middle), and 2k (bottom)

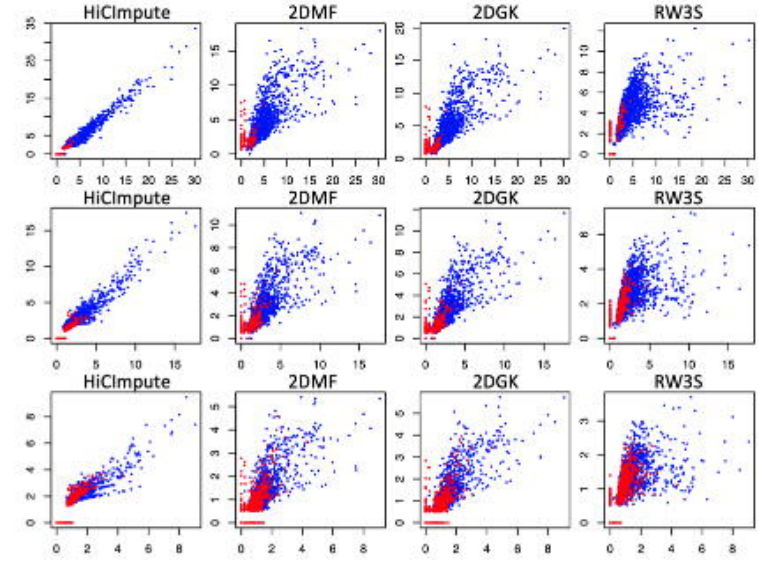

T1, 7K
T1, 4K
T1, 2K
T2, 7K
T2, 4K
T2, 2K
T3, 7K
T3, 4K
T3, 2K
PTDO
1−TPSZ
1−PTSZ
0.0
0.2
0.4
0.6
0.8
1.0
Methods
2DMF
2DGK
RW3S
HiCImpute



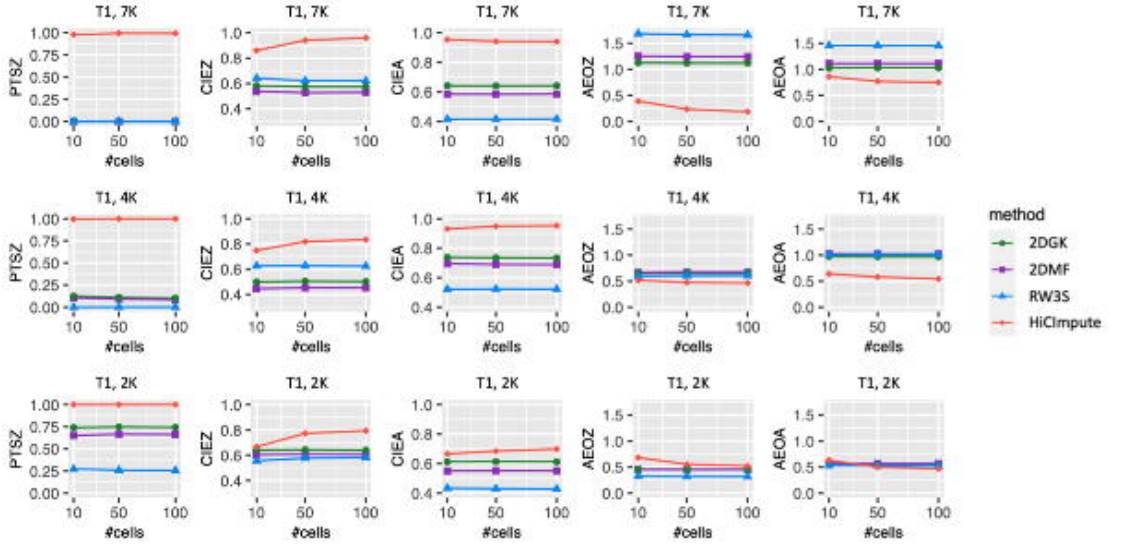
(a) T1, 7k (top), 4k (middle), and 2k (bottom)



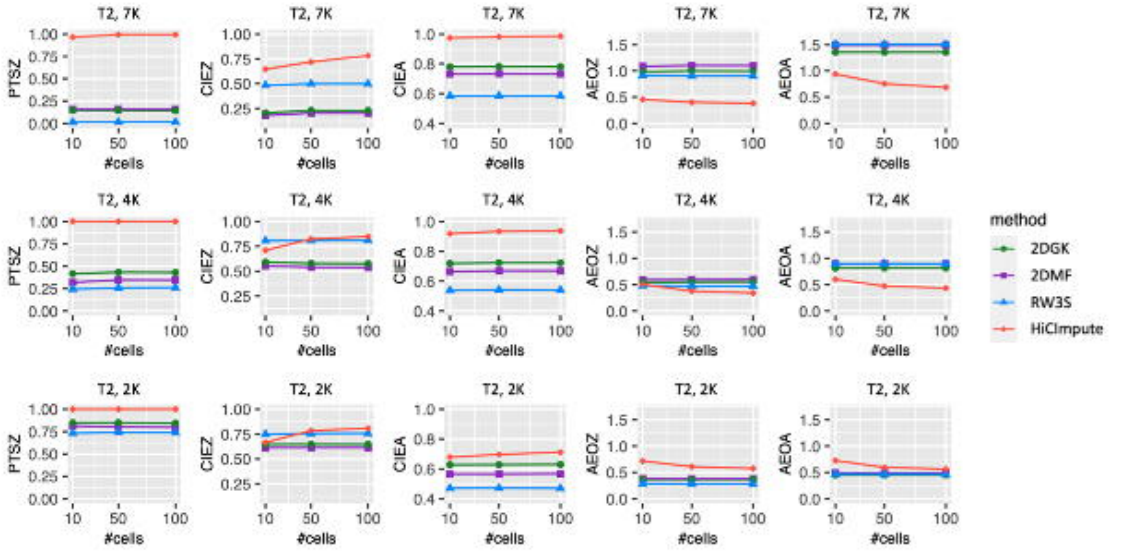
(b) T2, 7k (top), 4k (middle), and 2k (bottom)



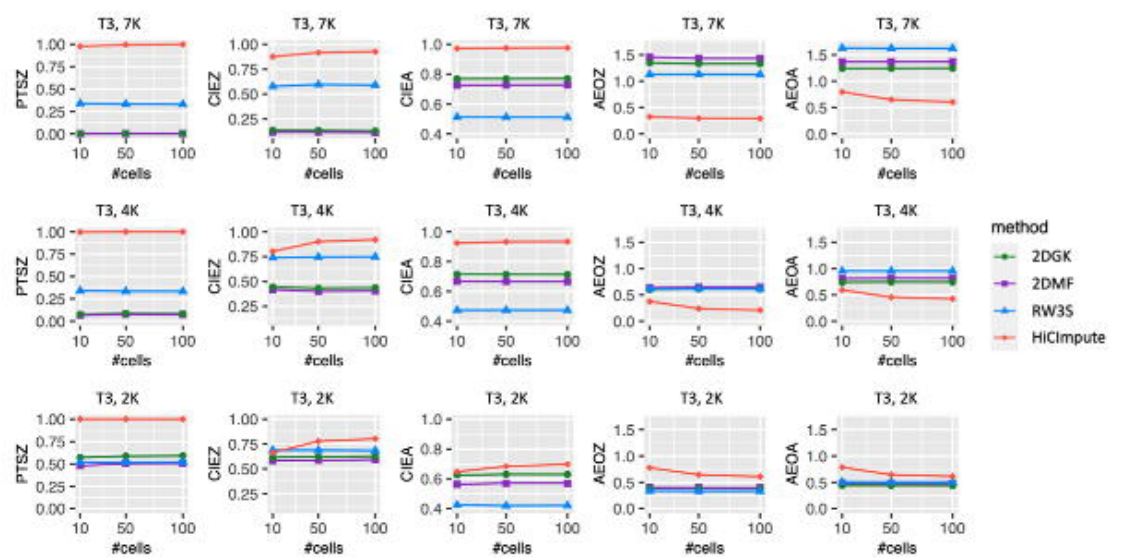
(c) T3, 7k (top), 4k (middle), and 2k (bottom)



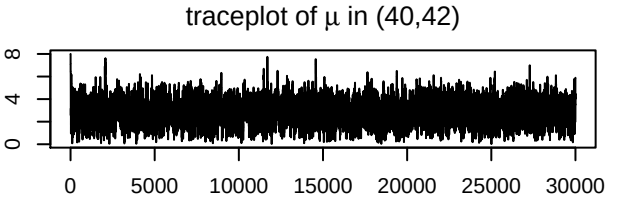
(a) T1, 7k (top), 4k (middle), and 2k (bottom)

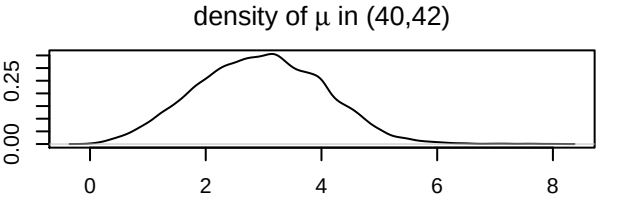


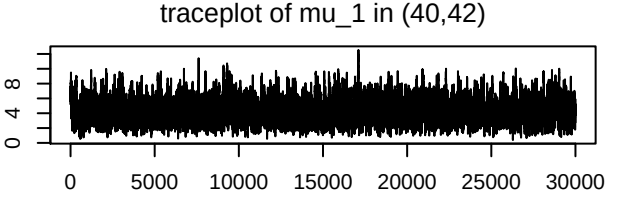
(b) T2, 7k (top), 4k (middle), and 2k (bottom)

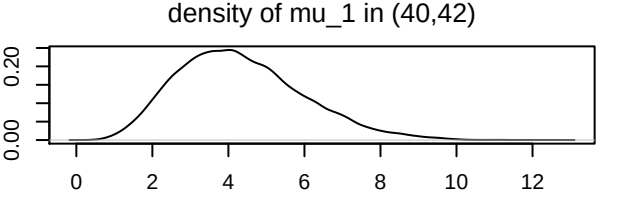


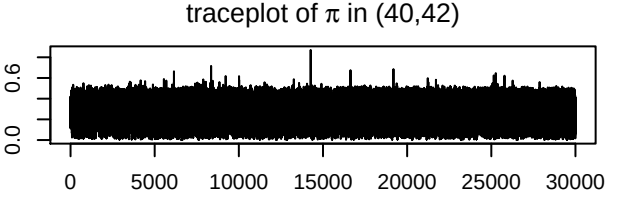
(c) T3, 7k (top), 4k (middle), and 2k (bottom)

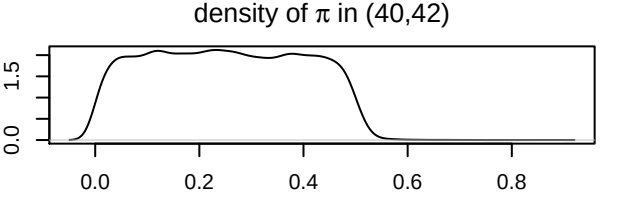

traceplot of μ in (40,42)
0
4
8
0
5000
10000
15000
20000
25000
30000

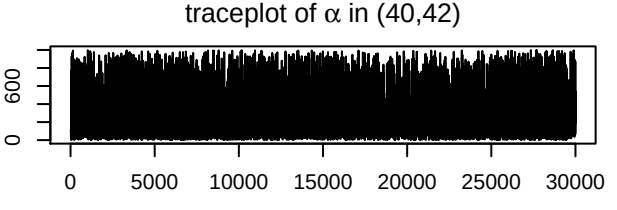

density of μ in (40,42)
0.00
0.25
0
2
4
6
8

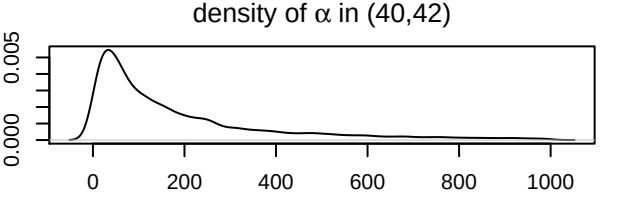

traceplot of mu_1 in (40,42)
0
4
8
0
5000
10000
15000
20000
25000
30000

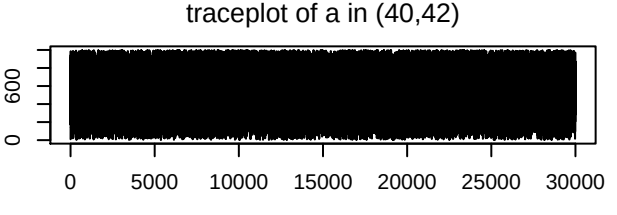

density of mu_1 in (40,42)
0.00
0.20
0
2
4
6
8
10
12

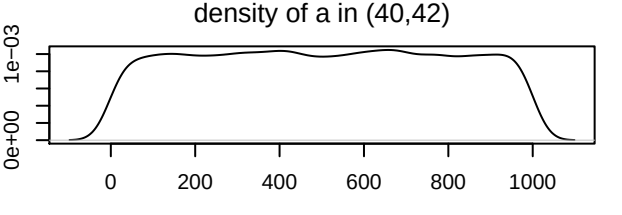

traceplot of π in (40,42)
0.0
0.6
0
5000
10000
15000
20000
25000
30000


density of π in (40,42)
0.0
1.5
0.0
0.2
0.4
0.6
0.8


traceplot of α in (40,42)
0
600
0
5000
10000
15000
20000
25000
30000


density of α in (40,42)
0.000
0.005
0
200
400
600
800
1000


traceplot of a in (40,42)
0
600
0
5000
10000
15000
20000
25000
30000


density of a in (40,42)
0e+00
1e−03
0
200
400
600
800
1000