SHARP: Single-cell RNA-seq Hyper-fast and Accurate Processing via Ensemble Random Projection

Shibiao Wan[1,2], Junil Kim[1,2,3] and Kyoung Jae Won[1,2,3*]

[1]Institute for Diabetes, Obesity and Metabolism, Perelman School of Medicine, University of Pennsylvania, Philadelphia, 19104, Pennsylvania, USA

[2]Department of Genetics, University of Pennsylvania, Philadelphia, 19104, Pennsylvania, USA

[3]Biotech Research and Innovation Centre (BRIC), University of Copenhagen, Copenhagen, Denmark.

To process large-scale single-cell RNA-sequencing (scRNA-seq) data effectively without excessive distortion during dimension reduction, we present SHARP, an ensemble random projection-based algorithm which is scalable to clustering 1.3 million cells. Comprehensive benchmarking tests demonstrate that SHARP outperforms existing methods in terms of speed and accuracy, especially for the large-size datasets (>40,000 cells) where SHARP is at least 20 times faster.

By enabling transcriptomic profiling at the individual-cell level, scRNA-seq has been widely applied in various domains of biology and medicine to characterize novel (or rare) cell types and detect intra-population heterogeneity [1]. The amount of scRNA-seq data in public domain has increased explosively due to technological development and the efforts to obtain large-scale transcriptomic profiling of cells [2]. Computational algorithms to process and analyze large-scale high-dimensional single-cell data are essential. Dimension reduction using principal component

* Corresponding author. E-mail: kyoung.won@bric.ku.dk

analysis (PCA) [3] or independent component analysis (ICA) [4] has been successfully applied to process and to visualize high dimensional scRNA-seq data. However, it requires considerable time to obtain principal or independent components as the number of cells increases. It is also notable that dimension reduction decreases processing time at the cost of losing original cell-to-cell distances. For instance, t-distributed stochastic neighbor embedding (tSNE) [5] effectively visualizes multi-dimensional data into a reduced-dimensional space. However, tSNE distorts the distance between cells for its visualization.

To effectively handle very large-scale scRNA-seq data without excessive distortion of cell-to-cell distances, we developed SHARP (https://github.com/shibiaowan/SHARP), a hyper-fast clustering algorithm based on ensemble random projection (RP) (Fig. 1a and Online Methods). RP [6] projects the original D-dimensional data into a d-dimensional ($d << D$) subspace, using a $d \times D$-dimensional random matrix $\mathbf{R}$ whose elements conform to a distribution with zero mean and unit variance. Notably, RP preserves cell-to-cell distances even in a much lower-dimensional space and is robust to missing values, which provides a well-suited condition for clustering high-dimensional scRNA-seq data (Online Methods). SHARP dramatically reduced the running cost for clustering while showing far enhanced clustering performance especially for large-size scRNA-seq datasets. Of note, SHARP is capable of clustering 1.3 million cells, which was not possible to other clustering approaches we tested.

SHARP employed a divide-and-conquer strategy followed by RP to accommodate effective processing of large-scale scRNA-seq data (Fig. 1a and Online Methods). In short, the scRNA-seq

data is divided into small blocks. RP followed by a hierarchical clustering algorithm is applied to each block. To merge individual RP-based clustering results, a weighted-ensemble clustering approach is used. Finally, a similarity-based ensemble clustering approach is to integrate clustering results of each block (Fig. 1a and Online Method).

Clustering involves extensive use of computational resources in calculating distances and/or dimension reduction. For scRNA-seq data with N cells and D genes, a simple hierarchical clustering requires time complexity of $O(N^2D)$ [7] to calculate distance between cells. The fastest version of tSNE combined with the k-means algorithm requires $O(DN\log(N))$ [5]. A simple PCA requires $O(ND \cdot \min(N, D))$ for data reduction [6]. By contrast, the time complexity for SHARP is only $O(N\log(N)\sqrt{D})$ (Online Methods). Employing a divide-and-conquer strategy and RP, SHARP can cluster unlimited number of cells theoretically. Besides, contrary to existing R-based clustering packages which can not even upload scRNA-seq data with more than 1 million cells, SHARP can effectively perform clustering on 1.3 million cells (Online Methods).

We performed comprehensive benchmarking of SHARP and other existing scRNA-seq clustering algorithms including SC3 [8], SIMLR [9], hierarchical clustering and tSNE combined with k-means (Fig. 1b-c and Supplementary Note 1) using 17 publicly available scRNA-seq datasets whose cell number ranges from 124 to 1.3 million cells [10-20] (Supplementary Table 1). To evaluate clustering performance, we used adjusted random index (ARI) [21] as well as running time.

The benchmarking tests demonstrate dramatic cost reduction of SHARP (Fig. 1b). Reflecting the

theoretical running costs, the two classical algorithms (tSNE + kMeans, hierarchical clustering) manifested exponential increase in their processing time as the number of cells increased (Fig. 1b). SC3 [8] and SIMLR [9] showed better performance than the classical clustering approaches, but they still required a considerable amount of time for clustering. The computing cost of SHARP was substantially lower than other clustering algorithms. Remarkably, the required computing cost of SHARP rose roughly linearly even with the very large size of the datasets. For the cells with larger than 40,000, SHARP ran at least 20 times faster than SC3 [8] and SIMLR [9].

Notably, SHARP clustered the scRNA-seq with 1.3 million cells in 42 minutes when using a multi-core system (Fig. 1b). Due to the data loading problem (and potential exhaustive memory use), we could not show the running time of other approaches for 1.3 million cells. When using a multi-core system on the Montoro dataset [17] with 66,265 cells, SHARP ran remarkably faster than SC3 and SIMLR (Supplementary Figure 1 and Supplementary Note 2). We expect far superior performance of SHARP against its competitors.

In parallel, we compared the clustering performance using the pre-defined cell types for each datasets (Supplementary Table 1). For almost all datasets we tested, SHARP showed better performances (Fig. 1c). It is notable that the performance became generally quite worse for large datasets (>40,000 single cells). In contrast, SHARP showed an ARI larger than 0.7 regardless of the size of the datasets, demonstrating its robustness.

To explain the robust clustering performance of SHARP, we investigate the degree of distortion

caused by dimension reduction by comparing the correlation of cell-to-cell distances after reducing dimension using SHARP, PCA and tSNE. Reflecting the property of RP, SHARP showed almost perfect similarities in cell-to-cell distance with correlation coefficient > 0.94 even in a dimensional space which is 74 times lower (from 20862 to 279) than the original one (Fig. 2a, Supplementary Figure 2 and Supplementary Note 3). Cell-to-cell distances were deteriorated when dimension reduction was performed to the same number of dimension using PCA (Fig. 2a). tSNE, an algorithm to visualize high-dimensional data into 2 or 3 dimensional space, showed a poor correlation (Fig. 2a).

scRNA-seq suffers a high frequency of dropouts where many of the true expressions are not captured. To evaluate the robustness of SHARP against dropouts, we tested SHARP while increasing dropout rates in a scRNA-seq dataset (Fig. 2b). We found that both SHARP and SC3 are robust to the added dropouts. The performance of SIMLR, even though it was better than SC3 when there were no added dropouts, became worse when the dropout rates were increased over 5%.

To visualize how well SHARP separate scRNA-seq data, we showed the results of SHARP in the two-dimensional space (Online Methods). Compared to original tSNE, SHARP showed a better separation of the cells (Fig. 2c and Supplementary Figures 3a-d). For instance, the neuron cells (purple) in the Darmanis dataset [11] became well separated in SHARP (Fig. 2c). Also, the ductal cells (cyan) in the Enge dataset [15] were better clustered using SHARP (Fig. 2c). SHARP also provides heat maps for the clustering results. For instance, the heat map for the Enge dataset [15]

clearly showed the cell types in pancreas including α (GCG), β (INS), acinar (PRSS1) and δ (SST) cells (Supplementary Figure 4).

SHARP provides an opportunity to study the million-cell-level dataset. Previous analysis on the scRNA-seq data with 1,306,127 cells from embryonic mice brains [20] was performed using the *k*-means and a graph clustering (equivalent to kernel k-means) algorithms [20]. However, *k*-means cannot identify the optimal number of clusters and it depends on the initial seeds for clustering. Using SHARP, we identified a total of 244 clusters from this datasets (17 clusters with more than 1,000 cells) (Supplementary Table S2). Fig. 2d showed the top 4 clusters among them. Gene ontology (GO) analysis (Supplementary Table S2) show that Cluster 2 is associated with dendrites and Cluster 3 is with axon. We also identified a cluster (Cluster 8) enriched for the genes associated with "non-motile cilium assembly", which is important for brain development and function [22] and an immune cells with high IL4 expression (Cluster 14).

SHARP employed ensemble average to combine the results of several runs of RP. We found that the ensemble strategy provides the robustness in clustering (Supplementary Figure 5 and Supplementary Note 4). SHARP was robust to the ensemble size when the size is larger than 5 (Supplementary Figures 6-7 and Supplementary Note 5). Moreover, SHARP's performance was not highly affected by the size of the block when the size is larger than 1,000 cells (Supplementary Figure 8 and Supplementary Note 5). SHARP is also roughly insensitive to the degree of dimension reduction (Supplementary Figure 9 and Supplementary Note 5). Besides clustering, the

property of RP to preserve cell-to-cell distance in the reduced dimension will be useful for other applications for scRNA-seq data.

References

1. Potter, S.S., *Single-cell RNA sequencing for the study of development, physiology and disease.* Nat Rev Nephrol, 2018. **14**(8): p. 479-492.
2. Han, X., et al., *Mapping the Mouse Cell Atlas by Microwell-Seq.* Cell, 2018. **172**(5): p. 1091-1107.
3. Jolliffe, I.T., *Principal component analysis and factor analysis*, in *Principal component analysis*. 1986, Springer. p. 115-128.
4. Hyvarinen, A. and E. Oja, *Independent component analysis: algorithms and applications.* Neural Netw, 2000. **13**(4-5): p. 411-30.
5. Van Der Maaten, L., *Accelerating t-SNE using tree-based algorithms.* The Journal of Machine Learning Research 2014. **15**(1): 3221-3245.
6. Bingham, E. and H. Mannila. *Random projection in dimensionality reduction: applications to image and text data*. in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. 2001. ACM.
7. Murtagh, F, and Legendre P., *Ward's hierarchical agglomerative clustering method: which algorithms implement Ward's criterion?*. Journal of classification 2014. **31**(3): 274-295.
8. Kiselev, V.Y., et al., *SC3: consensus clustering of single-cell RNA-seq data.* Nat Methods, 2017. **14**(5): p. 483-486.
9. Wang, B., et al., *Visualization and analysis of single-cell RNA-seq data by kernel-based similarity learning.* Nat Methods, 2017. **14**(4): p. 414-416.
10. Goolam, M., et al., *Heterogeneity in Oct4 and Sox2 Targets Biases Cell Fate in 4-Cell Mouse Embryos.* Cell, 2016. **165**(1): p. 61-74.
11. Darmanis, S., et al., *A survey of human brain transcriptome diversity at the single cell level.* Proceedings of the National Academy of Sciences of the United States of America, 2015. **112**(23): p. 7285-7290.
12. Wang, Y.J., et al., *Single-Cell Transcriptomics of the Human Endocrine Pancreas.* Diabetes, 2016. **65**(10): p. 3028-38.
13. Kolodziejczyk, A.A., et al., *Single Cell RNA-Sequencing of Pluripotent States Unlocks Modular Transcriptional Variation.* Cell Stem Cell, 2015. **17**(4): p. 471-85.
14. Baron, M., et al., *A Single-Cell Transcriptomic Map of the Human and Mouse Pancreas Reveals Inter- and Intra-cell Population Structure.* Cell Syst, 2016. **3**(4): p. 346-360 e4.
15. Enge, M., et al., *Single-Cell Analysis of Human Pancreas Reveals Transcriptional Signatures of Aging and Somatic Mutation Patterns.* Cell, 2017. **171**(2): p. 321-330 e14.

16. Klein, A.M., et al., *Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells.* Cell, 2015. **161**(5): p. 1187-1201.
17. *Montoro, D.T., et al. A revised airway epithelial hierarchy includes CFTR-expressing ionocytes.* Nature, 2018. **560**(7718): p. 319-324.
18. Park, J., et al., *Single-cell transcriptomics of the mouse kidney reveals potential cellular targets of kidney disease.* Science, 2018. **360**(6390): p. 758-763.
19. Macosko, E. Z., et al. Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. Cell, 2015. **161**(5): p. 1202-1214.
20. [https://support.10xgenomics.com/single-cell-gene-expression/datasets/1.3.0/1M_neurons]
21. Hubert, L. and P. Arabie, *Comparing partitions.* Journal of Classification, 1985. **2**(1): p. 193-218.
22. Guemez-Gamboa, A., et al. *Primary cilia in the developing and mature brain*. Neuron, 2014. **82**(3): p. 511-521.

Software availability. An R package for SHARP has been developed and is available at https://github.com/shibiaowan/SHARP.

Acknowledgements

This work is supported by NIH grant (R01 DK106027).

Authors' contributions

SW and KJW conceived and designed the experiments. SW performed the experiments and analyzed the data. SW, JK and KJW participated in writing the paper. The manuscript was approved by all authors.

Competing interests

The authors declare no competing financial interests.

Online Methods

The framework of SHARP. SHARP accepts gene expression data arranged in a matrix $\mathbf{M} \in \mathscr{R}^{D \times N}$, where each of the D rows corresponds to a gene (or transcript) and each of the N columns corresponds to a single cell. The type of input data can be either fragments/reads per kilo base per million mapped reads (FPKM/RPKM), counts per million mapped reads (CPM), transcripts per million (TPM) or unique molecule identifiers (UMI). For consistency, FPKM/RPKM values are converted into TPM values and UMI values are converted into CPM values. SHARP processes scRNA-seq data in 4 interconnected steps: (1) data partition, (2) RP based clustering, (3) weighted ensemble clustering and (4) similarity-based meta-clustering.

Data partition. For a large-scale dataset SHARP performs data partition using a divide-and-conquer strategy. SHARP divides scRNA-seq data $\mathbf{M} \in \mathscr{R}^{D \times N}$ into B blocks, where each block may contain different numbers of cells (i.e., $N_1, \ldots, N_B$, where $\sum_{i=1}^{B} N_i = N$). To avoid bias during data partition, we randomly permuted the original single-cell data before partitioning. In practice, SHARP roughly equally divides $\mathbf{M}$ and allows users to assign the base number of single cells in each block (e.g., n). In this case, $B = \lceil N/n \rceil$, where $\lceil x \rceil$ is the minimum integer no less than x. The numbers of single cells $\{N_b\}_{b=1}^{B}$ in each block are as follows:

1. If $B = 1, N_b = N,$ where $b = B = 1$;

2. If $B = 2, N_b = \begin{cases} \lfloor N/2 \rfloor, & \text{where } b = \{B-1\} \\ \lceil N/2 \rceil, & \text{where } b = \{B\} \end{cases}$;

3. If $B \geq 3$, $N_b = \begin{cases} n, & \text{where } b = \{1,\ldots,B-2\} \\ \lfloor (N - n(B-2))/2 \rfloor, & \text{where } b = \{B-1\} \\ \lceil (N - n(B-2))/2 \rceil, & \text{where } b = \{B\} \end{cases}$,

This enables SHARP to maximize the usage of local computational resources and avoid memory overflow while minimizing the negative impact from imbalanced numbers of data for each block.

It should be noted that R lacks 64-bit integers support and a scRNA-seq data matrix with >1 million cells whose number of elements is usually significantly larger than $2^{31} - 1$ can not be directly loaded. The divide-and-conquer strategy enables SHARP to upload more than 1 million cells.

Random Projection (RP). RP is a group of simple yet powerful dimension-reduction technique. It is based on the Johnson-Lindenstrauss lemma [23] given below:

Lemma 1. Given $\epsilon > 0$, a set $\mathscr{X}$ of N points in $\mathscr{R}^T$, and a positive integer $d \geq d_0 = \mathscr{O}(\log N/\epsilon^2)$, there exists $f: \mathscr{R}^T \rightarrow \mathscr{R}^d$ such that

$(1-\epsilon) \parallel \mathbf{u} - \mathbf{v} \parallel^2 \leq \parallel \mathbf{f}(\mathbf{u}) - \mathbf{f}(\mathbf{v}) \parallel^2 \leq (1+\epsilon) \parallel \mathbf{u} - \mathbf{v} \parallel^2$

for all $\mathbf{u}, \mathbf{v} \in \mathscr{X}$.

The lemma suggests that if points in a high-dimensional space are projected onto a randomly selected subspace of suitable dimension, the distances between the points are approximately preserved [24].

Specifically, the original D-dimensional data is projected onto a d-dimensional subspace, using a

random matrix whose column are unit length, i.e.,

$$\mathbf{P} = \frac{1}{\sqrt{\mathbf{d}}}\mathbf{RM} \in \mathscr{R}^{d\times N}, \mathbf{M} \in \mathscr{R}^{D\times N}, \mathbf{R} \in \mathscr{R}^{d\times D}$$

As long as the elements of **R** conforms to any distributions with zero mean and unit variance, **R** gives a mapping that satisfies the Johnson-Lindenstrauss lemma.

Choice of random matrix R. To reduce the computational complexity, we adopted a very sparse RP proposed in [25] where the elements of **R** (i.e., $r_{i,j}$) is defined as:

$$r_{i,j} = \sqrt{s}\begin{cases} 1 & \text{with probability } \frac{1}{2s}, \\ 0 & \text{with probability } 1-\frac{1}{s}, \text{ where } i = \{1,\ldots,d\}, j = \{1,\ldots,D\}. \\ -1 & \text{with probability } \frac{1}{2s}, \end{cases}$$

As suggested in [25], we selected $s = \sqrt{D}$.

Choice of the subspace dimension *d*. To balance between maintaining robust performance and yielding a solution as parsimonious as possible, we selected $d = \log_2(N)/\epsilon^2$, where $\epsilon \in (0,1]$ as suggested by Johnson-Lindenstrauss lemma.

Ensemble RP. After RP, pairwise correlation coefficients between each pair of single cells were calculated using the dimension-reduced feature matrix. An agglomerative hierarchical clustering (hclust) with the "ward.D" [26] method was used to cluster the correlation-based distance matrix.

The performance of an individual RP-based clustering is volatile. To obtain a robust performance, ensemble of several runs of RPs has been used. Specifically, we first applied RP K times to obtain K RP-based dimension-reduced feature matrices and then further K distance matrices. Each of the K matrices was clustered by a "ward.D"-based hclust. As a result, K different clustering results were obtained, each from a RP-based distance matrix, which would be combined by a weighted-based meta-clustering (wMetaC) algorithm [27] detailed in the next step.

Weighted-based meta-clustering (wMetaC). Compared to the traditional cluster-based similarity partitioning algorithm (CSPA) [28] which treat each instance and each cluster equally important, wMetaC assigns different weights to different instances (or instance pairs) and different clusters to improve the clustering performance. wMetaC includes four steps: (1) calculating cell weights; (2) calculating weighted cluster-to-cluster pairwise similarity; (3) clustering on weighted cluster-based similarity matrix and (4) determining final results by a voting scheme. Note that wMetaC was applied to each block of single cells.

Specifically, for calculating cell weights, similar to the first several steps in CSPA, we first converted the individual RP-based clustering results into a co-location similarity matrix **S**, whose element $s_{i,j}$ represents the similarity between the i-th and j-th single cells. Then, based on the idea that the weight for each pair of single cells is determined by the degree of consistency of the co-location clustering results of these two single cells, we converted the similarity matrix **S** to the weight matrix **W** according to the following equation:

$$w_{i,j} = s_{i,j}(1 - s_{i,j}),$$

where $w_{i,j}$ is the element in the i-th row and the j-th column of W. It is easy to see that when $s_{i,j} = 1$ (i.e., the i-th cell and the j-th cell are with 100% probability in the same cluster) or $s_{i,j} = 0$ (i.e., the i-th cell and the j-th cell are with 0% probability in the same cluster), $w_{i,j}$ reaches the minimum at 0; when $s_{i,j} = 0.5$ (i.e., the co-location probability of the i-th cell and the j-th cell in the same cluster is 0.5 whereas the probability of them in different clusters is also 0.5, which means this is the most difficult-to-cluster case), $w_{i,j}$ reaches the maximum at 0.25. In other words, zero weight is assigned to those most "easy-to-cluster" pairs of single cells and the highest cell-to-cell weight is assigned for the most "difficult-to-cluster" pairs. Then, a weight associated with each cell was calculated as the accumulation of all the cell-to-cell weights related with the corresponding cell. To calculate the weighted cluster-to-cluster similarity, we first noted that the size of the similarity matrix is $|C| \times |C|$, where C is the union set of all the clusters obtained in each individual RP-based clustering results in the previous step, and $|\cdot|$ is the cardinality of a set. Then, for any two clusters, their similarity is determined by the sum of weights of their overlapped elements (i.e., cells) divided by that of their combined ones. Then, in the third step (i.e., meta-clustering), we used a hierarchical clustering with "ward.D" to cluster the obtained similarity matrix. After clustering, we understood which cluster in the 1-st RP-based clustering corresponds to which cluster(s) in the 2-nd, 3-rd,…, K-th RP-based clustering. Then, in the final step, we reorganized the K RP-based clustering results according to the result in the third step, and then we used a voting scheme to determine the final clustering results. These procedures were repeated for each of the B blocks.

Similarity-based meta-clustering (sMetaC). To integrate the clustering results of the B blocks obtained by wMetaC, we proposed a similarity-based meta-clustering (sMetaC) approach which is similar to wMetaC. The major differences between wMetaC and sMetaC: (1) the cluster-to-cluster pairwise similarity of the former is calculated based on co-location weights of single cells in each cluster, whereas that of the latter is calculated based on the mean of the cell-to-cell correlation coefficients; (2) the individual clustering results of the former actually correspond to the same block of single cells but in different lower-dimensional space, whereas those of the latter correspond to different blocks of single cells; (3) the former requires a voting scheme to integrate K individual clustering results, whereas the latter does not and it just needs to reorganize the clusters to make clusters consistent across blocks.

Determining the optimal number of clusters. SHARP determines the optimal number of clusters by using three criteria which are based on internal evaluations of the clustering results (Supplementary Note 6).

Time complexity analysis. SHARP includes 4 steps for clustering: (1) data partition; (2) RP; (3) wMetaC; (4) sMetaC. For a scRNA-seq data matrix $\mathbf{M} \in \mathscr{R}^{D \times N}$, SHARP first divides the data into B blocks, the b-th block with N_b single cells. According to the "data partition" section, $N_b \leq n$, where n is a fixed user-defined parameter enabling that one application of RP-based clustering runs sufficiently fast. Our analysis (Supplementary Figure 13) shows that $n = 1500$ or 2000 is a good balance between performance and speed in our case. Then, for each block, one run of RP requires time complexity of $O(nd\sqrt{D})$ [23], where $d = \lceil \log(N)/\epsilon^2 \rceil \ll D$. Note here, d is calculated

based on N rather than n for dimension-reduction consistency across blocks. Practically, in the 13 reported scRNA-seq datasets, the dimension can be reduced by 42~238 times (i.e., D/d), depending on the number of single cells and the number of genes. SHARP requires several (i.e., K) runs of RPs (with complexity of $O(Knd\sqrt{D})$). Subsequently, SHARP uses a hierarchical clustering (hclust) with "ward.D" for each of the K RPs, thus with overall time complexity of $O(K(nd\sqrt{D}+n^2))$ (Note that the time complexity of hclust in R package is $O(n^2)$ [7]). Later, wMetaC, essentially a hclust for the individual predicted clusters (also without loss of generality, suppose the number of clusters for each RP-based clustering is equally C_1, where $C_1 \ll n$), was applied to each block (in total, time complexity of $O(K(nd\sqrt{D}+n^2)+(KC_1)^2)$). Finally, SHARP integrated the results of all blocks by proposing a method called sMetaC, whose time complexity is similar to wMetaC except the number of instances is different (similarly, we can suppose the number of clusters in each block is equally C_2, where $C_2 \ll n$). In this case, the total time complexity is $O(B[K(nd\sqrt{D}+n^2)+(KC_1)^2]+(BC_2)^2)$. Practically, K, B, C_1 and C_2 are very small, thus the time complexity of SHARP can be written as $O(KN(d\sqrt{D}+n))$. Because n is fixed across different datasets, $d=\lceil \log(N)/\epsilon^2 \rceil$ and D is usually larger than 10,000, thus $d\sqrt{D} > n$ and therefore the time complexity of SHARP is essentially $O(N\log(N)\sqrt{D})$.

On the other hand, among the compared state-of-the-art methods, tSNE plus k-means is arguably the fastest. Theoretically, tSNE requires $O(DN\log(N))$ for dimension reduction to 2D or 3D space [5]. For tSNE plus k-means for clustering, the time complexity is $O(DN\log(N)+2Nki)$, where k is the number of clusters and i is the number of iterations. Thus, the total time complexity for tSNE + kMeans is $O(DN\log(N))$.

All the tests except for the 1.3-million-cell dataset were performed using a single core on an Intel Xeon CPU E5-2699 v4 @ 2.20GHz system with 500GB memory. To run 1.3 million cells, we used 16 cores on the same system.

Visualization. For visualization, SHARP uses a weighted combination of the dimension-reduced feature matrix and the cell-to-cluster matrix derived from the clustering results. The former matrix is obtained by three steps: 1) applying K runs of RPs for each block of the large-scale scRNA-seq data; 2) combining these block-wise matrices to obtain K RP-based dimension-reduced matrices and 3) averaging these K matrices into one ensemble matrix. For the latter matrix, we constructed a $N \times pC$ matrix, where N is the number of single cells and pC is the predicted number of clusters. If the i-the single cell is predicted to be in the j-th cluster, then the element of the i-th row and j-th column is 1, otherwise 0. Subsequently, these two matrices were combined with different weights to formulate the visualization matrix, which is the input matrix of tSNE for visualization.

Based on the clustering results, SHARP can further detect genes for each cluster. We adopted a method similar to SC3 except three points: (1) Besides p-value and area under receiver operating curve (AUROC), SHARP uses two more criteria to select marker genes, namely cluster-mean fold change (FC) and expression sparsity (i.e., the percentage of expressions across all cells); (2) SHARP uses an adaptive threshold instead of a hard-threshold (i.e., p-value < 0.01 and AUROC > 0.85); (3) SHARP uses a parallelization way to calculate all of the criteria mentioned above.

References

23. Johnson, W.B. and J. Lindenstrauss, *Extensions of Lipschitz mappings into a Hilbert space.* Contemporary mathematics, 1984. **26**(189-206): p. 1.
24. Frankl, P. and H. Maehara, *The Johnson-Lindenstrauss Lemma and the Sphericity of Some Graphs.* Journal of Combinatorial Theory Series B, 1988. **44**(3): p. 355-362.
25. Li, P., T.J. Hastie, and K.W. Church. *Very sparse random projections*. in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2006. ACM.
26. Ward, J.H., *Hierarchical Grouping to Optimize an Objective Function.* Journal of the American Statistical Association, 1963. **58**(301): p. 236-&.
27. Ren, Y.Z., et al., *Weighted-object ensemble clustering: methods and analysis.* Knowledge and Information Systems, 2017. **51**(2): p. 661-689.
28. Strehl, A. and J. Ghosh, *Cluster ensembles---a knowledge reuse framework for combining multiple partitions.* Journal of machine learning research, 2002. **3**(1): p. 583-617.

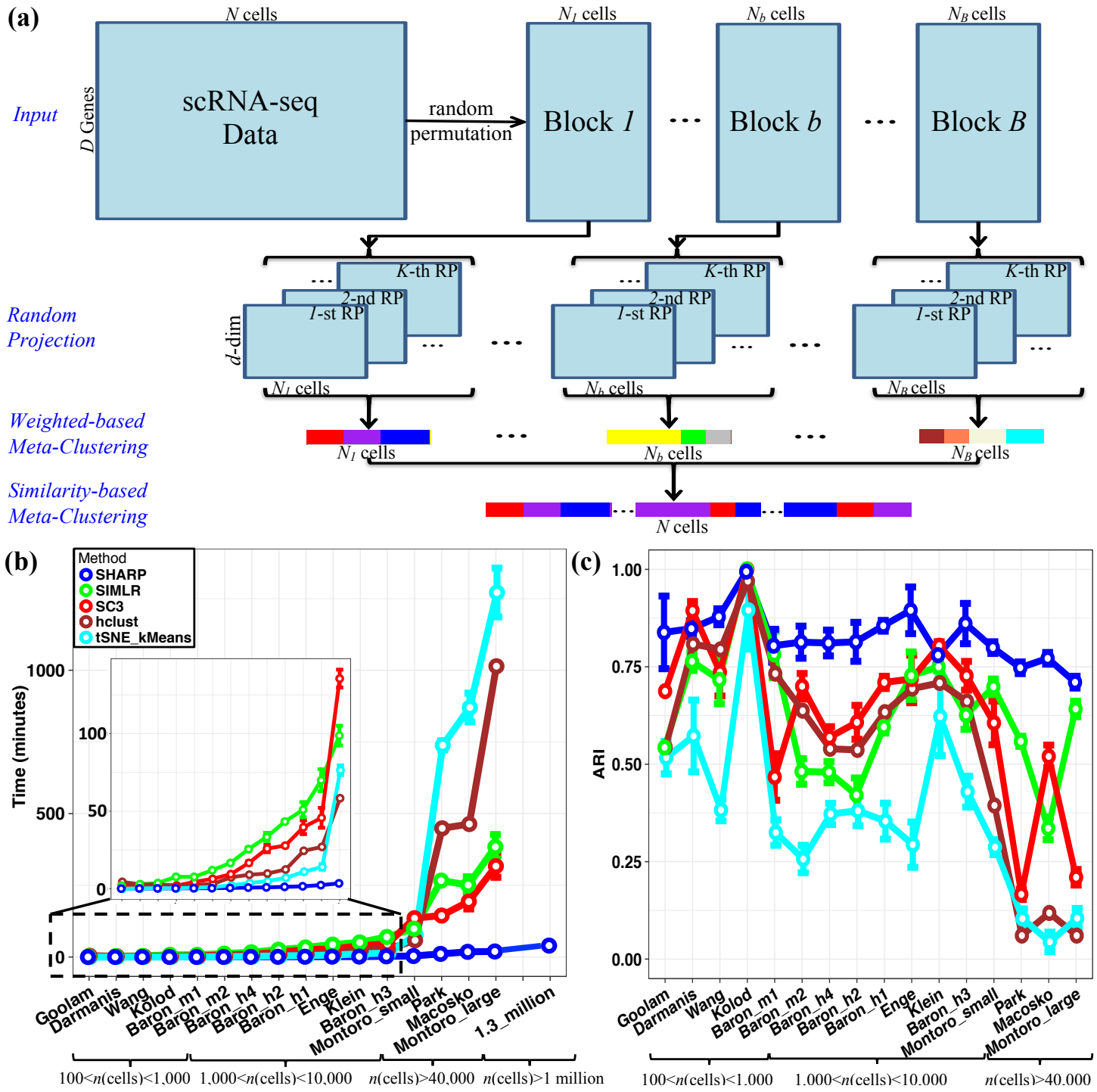

Figure 1 The framework of SHARP. (a) SHARP has 4 steps for clustering: divide-and-conquer, random projection (RP), weighted-based meta clustering, and similarity-based meta-clustering. (b) Running time and (c) clustering performance (adjusted Rand index, ARI [21]) of SHARP in 17 single-cell RNA-seq datasets [10-20] with numbers of single cells ranging from 124 to 1.3 million. Among them, for the dataset containing 1.3 million single cells, only the running time was provided due to lack of ground-truth clustering labels. All of the results for SHARP were based on 100 realizations of SHARP on each dataset. All the tests except for the 1.3-million-cell dataset were performed using a single core on an Intel Xeon CPU E5-2699 v4 @ 2.20GHz system with 500GB memory. To run 1.3 million cells, we used 16 cores on the same system.

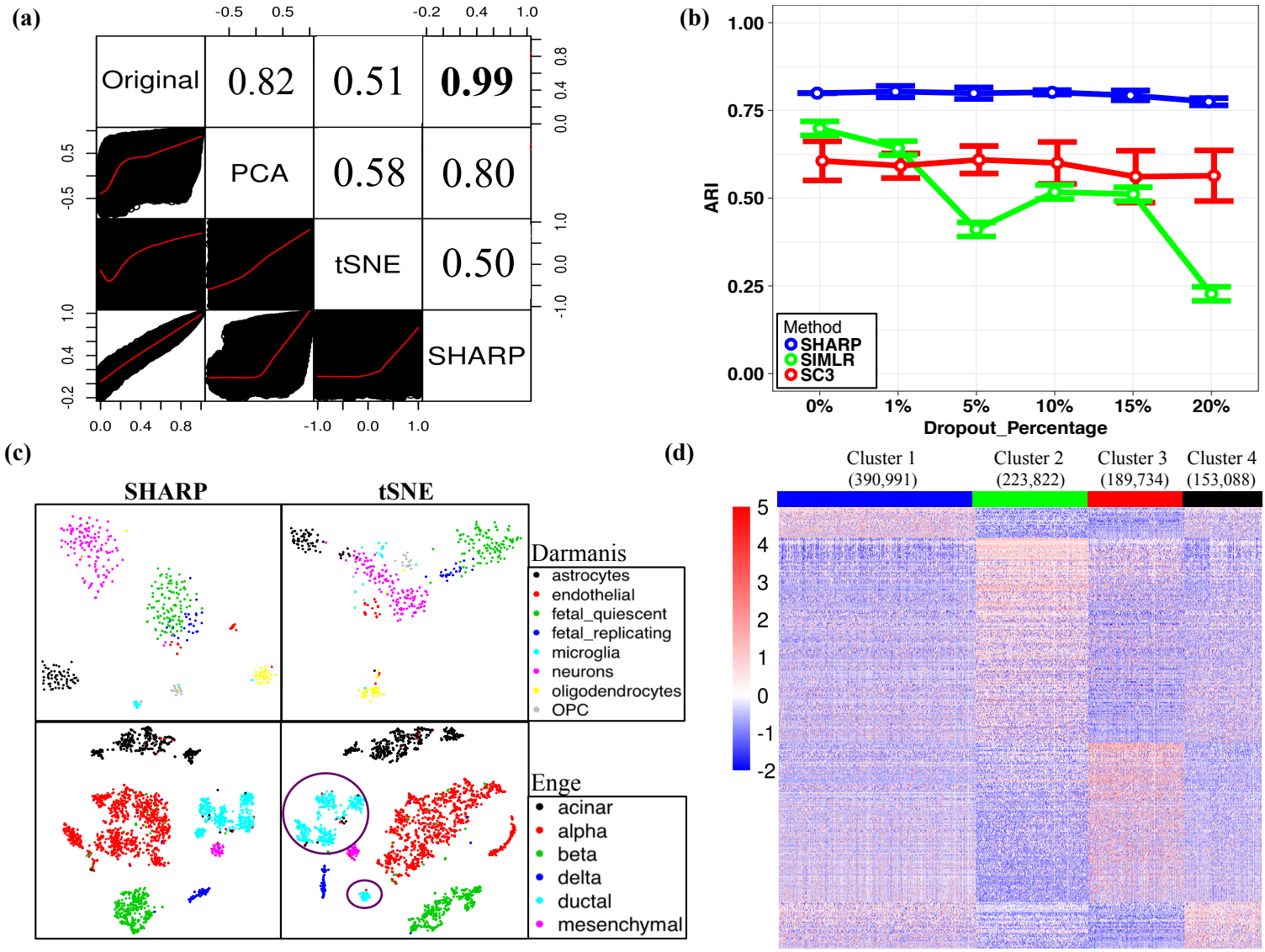

Figure 2 (a) Cell-to-cell distance preservation in SHARP space comparing with that in tSNE and PCA for the Enge [15] dataset. The lower triangular part shows the scatter plots of the cell-to-cell distances, whereas the upper triangular part shows the Pearson correlation coefficient (PCC) of the corresponding two spaces. The higher the PCC, the better the cell-to-cell distance preserves in the dimension-reduced space. (b) SHARP is robust to the dropout events on the Montoro_small [17] dataset when compared with SIMLR and SC3. The dropout percentage is calculated based on the nonzero gene expressions across cells. (c) Visualization capabilities of SHARP compared with that of tSNE in the Darmanis [11] and Enge [15] datasets. The cyan points of the Enge dataset determined by tSNE indicates that tSNE clusters them in two distant groups, whereas SHARP can correctly cluster them into one compact cluster. (d) Cluster-specific marker gene expressions of the top 4 major clusters for the 1.3 million single cells [20] by SHARP. The total number of clusters predicted by SHARP is 244. The number in brackets represents the number of single cells in the corresponding cluster. The clustering time is around 42 minutes.