SMNN: Batch Effect Correction for Single-cell RNA-seq data via Supervised Mutual Nearest Neighbor Detection

Yuchen Yang[1*], Gang Li[2*], Huijun Qian[2], Kirk C. Wilhelmsen[1,3], Yin Shen[4,5], and Yun Li[1,6,7,†]

[1]Department of Genetics, [2]Statistics and Operations Research, [6]Biostatistics, [7]Computer Science, [3]Renaissance Computing Institute, University of North Carolina, Chapel Hill, NC 27599, USA. [4]Institute for Human Genetics, [5]Department of Neurology, University of California, San Francisco, San Francisco, CA 94143, USA

[*] Equal contribution

Abstract

An ever-increasing deluge of single-cell RNA-sequencing (scRNA-seq) data has been generated, often involving different time points, laboratories or sequencing protocols. Batch effect correction has been recognized to be indispensable when integrating scRNA-seq data from multiple batches. A recent study proposed an effective correction method based on mutual nearest neighbors (MNN) across batches. However, the proposed MNN method is unsupervised in that it ignores cluster label information of single cells. Such cluster or cell type label information can further improve effectiveness of batch effect correction, particularly under realistic scenarios where true biological differences are not orthogonal to batch effect. Under this motivation, we propose SMNN which performs supervised mutual nearest neighbor detection for batch effect correction of scRNA-seq data. Our SMNN either takes cluster/cell-type label information as input, or, in the absence of such information, infers cell types by performing clustering of scRNA-seq data. It then detects mutual nearest neighbors within matched cell types and corrects batch effect accordingly. Our extensive evaluations in simulated and real datasets show that SMNN provides improved merging within the corresponding cell types across batches, leading to reduced differentiation across batches over MNN. Furthermore, SMNN retains more cell type-specific features after correction. Differentially expressed genes (DEGs) identified between cell types after SMNN correction are biologically more relevant, and the DEG true positive rates improve by up to 841%. SMNN is implemented in R, and freely available at https://yunliweb.its.unc.edu/SMNN/ and https://github.com/yycunc/SMNNcorrect.

Author summary

The presence of batch effects poses grand challenges to integrative analysis of scRNA-seq data from multiple resources. One powerful tool MNN corrects batch effect of scRNA-seq data based on mutual nearest neighbors across batches. However, this method makes a critical assumption that batch effect is orthogonal to true biological differences. This assumption in practice can easily be violated. When that happens, MNN suffers from biases introduced by wrongly matched pairs of cells. To overcome this shortcoming, here we present a new method, SMNN, which performs supervised mutual nearest neighbor detection for batch effect correction. We benchmark the performance of SMNN using both simulations and real data, and demonstrate that, compared to MNN, our SMNN can better mix cells of the same type/state across batches. More importantly, SMNN can more effectively retain

biologically relevant features, and thereof provide improved cell type clustering and enhanced power for detecting differentially expressed genes (DEGs) between different cell types.

Introduction

An ever-increasing amount of single cell RNA-sequencing (scRNA-seq) data has been generated as scRNA-seq technologies mature and sequencing costs continue dropping. However, large scale scRNA-seq data, for example, those profiling tens of thousands to millions of cells (such as the Human Cell Atlas Project [1], almost inevitably involve multiple batches across time points, laboratories, or experimental protocols. The presence of batch effect renders joint analysis across batches challenging [2,3]. Batch effect, or systematic differences in gene expression profiles across batches, not only can obscure the true underlying biology, but also may lead to spurious findings. Thus, batch effect correction, which aims to mitigate the discrepancies across batches, is crucial and deemed indispensable for the analysis of scRNA-seq data across batches [4].

Because of its importance, a number of batch effects correction methods has been recently proposed and implemented. Most of these methods, including limma [5], ComBat [6], and svaseq [7], are regression-based. Among them, limma and ComBat explicitly model known batch effect as a blocking term. Because of the regression framework adopted, standard statistical approaches to estimate the regression coefficients corresponding to the blocking term can be conveniently employed. In contrast, svaseq is often used to detect underlying unknown factors of variation, for instance, unrecorded differences in the experimental protocols. svaseq first identifies these unknown factors as surrogate variables and subsequently corrects them. For these regression-based methods, once the regression coefficients are estimated or the unknown factors are identified, one can then regress out these batch effects accordingly, obtaining residuals that will serve as the batch-effect corrected expression matrix for further analyses. These methods have become standard practice in the analysis of bulk RNA-seq data. However, when it comes to scRNA-seq data, one key underlying assumption behind these methods, that the cell composition within each batch is identical, might not hold. Consequently, estimates of the coefficients might be inaccurate. As a matter of fact, when applied to scRNA-seq data, the corrected results derived from these methods widely adopted for bulk RNA-seq data might be even inferior to raw data without no correction, in some extreme cases [8].

To address the heterogeneity and high dimensionality of complex data, several dimension-reduction approaches have been adopted. An incomplete list of these strategies includes principal component analysis (PCA), autoencoder, or force-based methods such as t-distributed stochastic neighbor embedding (t-SNE) [9]. Through those dimension reduction techniques, one can project new data onto the reference dataset using a set of landmarks from the reference [8,10,11,12] to remove batch effects between any new dataset and the reference dataset. Such projection methods require the reference batch contains all the cell types across batches. As one example, Spitzer *et al.* [11] employed force-based dimension reduction and showed that leveraging a few landmark cell types from bone marrow (the most appropriate tissue in that it provides the most complete coverage of immune cell types) allowed mapping and comparing immune cells across different tissues and species. When applied to scRNA-seq data, however, these methods suffer when cells from a new batch fall out of the space inferred from the reference. Furthermore, determining the dimensionality of the low dimensional manifolds is still an open and challenging problem. To address the limitations of existing methods, a recent study proposed a new method, MNN correction, which leverages information of mutual nearest neighbors across batches. MNN has demonstrated superior performance over alternative methods [8]. MNN makes a critical assumption that true biological differences are orthogonal to batch effect. For this assumption to hold, variation from batch effect is required, at the minimum, to be much smaller than that from biological effect. Under this assumption, MNN finds, across batches, mutual nearest neighboring cells for each cell to be corrected, and then computes batch-effect correction vectors under a Gaussian kernel. However, this orthogonality assumption might not hold in real data, particularly given that different batches may easily differ in many aspects, including samples used, single cell capture method, or library preparation approach. Under non-orthogonal scenarios, MNN will not be optimal using its global (ignoring cell type information) nearest neighbor search strategy, leading to undesired correction results. For example, under the scenario depicted in **Fig 1b**, MNN leads to cluster 1 (C1) and cluster 2 (C2) mis-corrected due to mismatching single cells in the two clusters/cell-types across batches.

To address the above issue, here we present SMNN, a supervised version of MNN that incorporates cell type information. SMNN performs nearest neighbor searching within the same cell type, instead of global searching ignoring cell type labels (**Fig 1a**). Cell type information, when unknown *a priori*, can be inferred via clustering methods [13,14,15,16] .

Results

SMNN Framework

The motivation of our SMNN lies in the potential of single-cell cluster information to improve the accuracy of nearest neighbor (NN) identification. A preliminary clustering within each batch before any correction can provide knowledge regarding cell composition within each scRNA-seq dataset, which serves to encode the cellular correspondence across batches (**Fig 1a**). With this clustering information, we can refine the nearest neighbor searching space within a certain population of cells that are of the same or similar cell type(s) or state(s) in all the batches.

SMNN takes a natural two-step approach to leverage cell type label information for enhanced batch effect correction (**Fig 1c and S1 text**). First, it takes the expression matrices across multiple batches as input, and performs clustering separately for each batch. Specifically, in this first step, SMNN uses Seurat v. 3.0 [17] where dimension reduction is conducted via principal component analysis (PCA) to the default of 20 PCs, and then graph-based clustering follows on the dimension-reduced data with *resolution* parameter of 0.9 [18]. Obtaining an accurate matching of the cluster labels across batches is of paramount importance for subsequent nearest neighbor detection. SMNN requires users to specify a list of marker genes and their corresponding cell type labels to match clusters/cell types across batches. We later refer to this cell type or cluster matching as cluster harmonization across batches. Because not all cell types are necessarily shared across batches, and no prior knowledge exists regarding the exact composition of cell types in each batch, SMNN allows users to take discretion in terms of the marker genes to include, representing the cell types that, they believe, are shared across batches. Based on the marker gene information, a harmonized label is assigned to *every* cluster identified across all the batches according to two criteria: the percentage of cells in a cluster expressing a certain marker gene and the average gene expression levels across all the cells in the cluster. After harmonization, cluster labels are comparable across batches. This completes step 1 of SMNN. Note that this entire clustering step can be bypassed by feeding SMNN cluster labels that are consistent or comparable across batches.

With the harmonized cluster or cell type label information obtained in the first step, SMNN, in the second step, searches mutual nearest neighbors only within each matched cell type between the first batch (which serves as the reference batch) and any of the other batches (the current batch), and performs batch effect correction accordingly. Compared to MNN, where the mutual nearest neighbors are searched globally, SMNN identifies each pair of

neighbors from the same cell population or state, which can provide more accurate information for downstream correction. Then, following MNN, SMNN computes batch effect correction vector for each identified pair of cells and calculates the cell-specific correction vectors by exploiting a Gaussian kernel to obtain a weighted average across all the pair-specific vectors of their mutual nearest neighbors. Each cell's correction vector is further scaled according to the cell's location in the space defined by the correction vector, and standardized according to quantiles across batches, in order to eliminate the "kissing effects" phenomenon. "Kissing effects" refer to the phenomenon that naïve batch effect correction brings only the surfaces of two point-clouds in contact, rather than fully merging them [8]. At the end of the second step, SMNN returns the batch-effect corrected expression matrix for each batch, as well as the information regarding nearest neighbors between the reference batch and the current batch under correction. This step is carried out for every batch except the reference batch so that all batches are corrected to the same reference batch in the end.

Simulation results

Since MNN has been shown to excel alternative methods [4,8], we here focus on comparing our SMNN with MNN. We first compared the performance of SMNN to MNN in simulated data. In our simulations, SMNN demonstrates superior performance over MNN under both orthogonal and non-orthogonal scenarios (**Fig 2 and 3 and S1-3 Fig**). We show t-SNE plot for each cell type before and after MNN and SMNN correction under both the orthogonal and non-orthogonal scenarios. Under orthogonality, the two batches partially overlapped in the t-SNE plot before correction, suggesting that the variation due to batch effect was indeed much smaller than that due to biological effect. Both MNN and SMNN successfully mixed single cells from two batches (**S2 Fig**). However, for cell types 1 and 3, there were still some cells from the second batch left unmixed with those from the first batch after MNN correction (**S2a and c Fig**). Under the non-orthogonal scenario, the differences between two batches were more pronounced before correction, and SMNN apparently outperformed MNN (**S3 Fig**), especially in cell type 1 (**S3a Fig**). Moreover, we also computed Frobenius norm distance [19] for each cell between its simulated true profile before introducing batch effects and after SMNN and MNN correction. The results showed an apparently reduced deviation from the truth after SMNN correction than MNN (**Fig 3**). These results suggest that SMNN provides improved batch effect correction over MNN under both orthogonal and non-orthogonal scenarios.

Real data results

For performance evaluation in real data, we first carried out batch effect correction on two hematopoietic datasets using SMNN and MNN, respectively. **Fig 4a-c** shows t-SNE plot before and after correction. Notably, both SMNN and MNN substantially mitigated discrepancy between the two datasets. Comparatively, SMNN better mixed cells of the same cell type across batches (**S4 Fig**), especially for CMP and MEP cells, which were wrongly corrected by MNN due to sub-optimal nearest neighbor search ignoring cell type information (**S5 Fig**). We also compared the distance for the cells between batch 1 and 2, and found that, compared to data before correction, both MNN and SMNN reduced the Euclidean distance between the two batches (**S6 Fig**). Moreover, SMNN further decreased the distance by up to 8.2% than MNN (2.8%, 4.3% and 8.2% for cells of type CMP, MEP and GMP, respectively). Regarding the overall variance in the two batches, our results show that, SMNN reduced the overall variance by up to 4.8% on top of MNN corrected results (**Fig 4d-f**). These results suggest improved batch effect correction by SMNN.

SMNN identifies differentially expressed genes that are biologically relevant

We then compared the DEGs among different cell types identified by SMNN and MNN. After correction, in the merged hematopoietic dataset, 1012 and 1145 up-regulated DEGs were identified in CMP cells by SMNN and MNN, respectively, when compared to GMP cells, while 926 and 1108 down-regulated DEGs were identified by the two methods, respectively (**Fig 5a and S7a Fig**). Of them, 736 up-regulated and 842 down-regulated DEGs were shared between SMNN and MNN corrected data. GO enrichment analysis showed that, the DEGs detected only by SMNN were overrepresented in GO terms related to blood coagulation and hemostasis, such as platelet activation and aggregation, hemostasis, coagulation and regulation of wound healing (**Fig 5b**). Similar DEG detection was carried out to detect genes differentially expressed between CMP and MEP cells. 181 SMNN-specific DEGs were identified out of the 594 up-regulated DEGs in CMP cells when compared to MEP cells (**Fig 5c**), and they were found to be enriched for GO terms involved in immune cell proliferation and differentiation, including regulation of leukocyte proliferation, differentiation and migration, myeloid cell differentiation and mononuclear cell proliferation (**Fig 5d**). Lastly, genes identified by SMNN to be up-regulated in GMP when compared to MEP cells, were found to be involved in immune processes; whereas up-regulated genes in MEP over GMP were enriched in blood

coagulation (**S7e-h Fig**). These cell-function-relevant SMNN-specific DEGs indicate SMNN can maintain some cell features that are missed by MNN after correction.

In addition, we considered two sets of "working truth": first, DEGs identified in uncorrected batch 1; second DEGs identified in batch 2, and we compared SMNN and MNN results to both sets of working truth. The results showed that, in both comparisons (one comparison for each set of working truth), fewer DEGs were observed in SMNN-corrected batch 2, but higher TPR in each of the three cell types than those in MNN results. When compared to the uncorrected batch 1, 3.6% - 841% improvements were observed in SMNN results than MNN (**Fig 6 and S8 and S9 Fig**). Similarly, SMNN increased the TPR by 6.2% - 54.0% on top of MNN when compared to uncorrected batch 2 (**S10-12 Fig**). Such an improvement in the accuracy of DEG identification indicates that higher amount of information regarding cell structure was retained after SMNN correction than MNN.

We also identified DEGs between T cells and B cells in the merged PBMC and T cell datasets after SMNN and MNN correction, respectively. Compared to B cells, 3213 and 4180 up-regulated DEGs were identified in T cells by SMNN and MNN, respectively, 2203 of which were shared between the two methods (**S13a Fig**). GO enrichment analysis showed that, the SMNN-specific DEGs were significantly enriched for GO terms relevant to the processes of immune signal recognition and T cell activation, such as T cell receptor signaling pathway, innate immune response−activating signal transduction, cytoplasmic pattern recognition receptor signaling pathway and regulation of autophagy (**S13b Fig**). In B cells, 5422 and 3462 were found to be up-regulated after SMNN and MNN correction, where 2765 were SMNN-specific (**S13c Fig**). These genes were overrepresented in GO terms involved in protein synthesis and transport, including translational elongation and termination, ER to Golgi vesicle−mediated transport, vesicle organization and Golgi vesicle budding (**S13d Fig**). These results again suggest that SMNN more accurately retains or rescues cell features after correction.

SMNN more accurately identifies cell clusters

Finally, we examined the ability to differentiate cell types after SMNN and MNN correction in three datasets (**S1 Table**). In all three real datasets, ARI after SMNN correction showed 7.6 - 42.3% improvements over that of MNN (**Fig 7**), suggesting that SMNN correction more effectively recovers cell-type specific features.

Discussion

In this study, we present SMNN, a batch effect correction method for scRNA-seq data via supervised mutual nearest neighbor detection. Our work is built on the recently developed method MNN, which has showed advantages in batch effect correction than existing alternative methods. On top of MNN, our SMNN relaxes a strong assumption that underlies MNN: that the biological differentiations are orthogonal to batch effects [8]. When this fundamental assumption is violated, especially under the realistic scenario that the two batches are rather different, MNN tends to err when searching nearest neighbors for cells belonging to the same biological cell type across batches. Our SMNN, in contrast, explicitly considers cell type label information to perform supervised mutual nearest neighbor matching, thus empowered to extract only desired neighbors from the same cell type.

A notable feature of our SMNN is that it can detect and match the corresponding cell populations across batches with the help of feature markers provided by users. SMNN performs clustering within each batch before merging across batches, which can reveal basic data structure, i.e. cell composition and proportions of contributing cell types, without any adverse impact due to batch effects. Cells of each cluster are labeled by leveraging their average expression levels of certain marker(s), thus enabling us to limit the mutual nearest neighbor detection within a smaller search space (i.e., only among cells of the same or similar cell type or status). This supervised approach eliminates the correction biases incurred by pairs of cells wrongly matched across cell types. We benchmarked SMNN together with MNN on both simulated and three published scRNA datasets. Our results clearly show the advantages of SMNN in terms removing batch effects. For example, our results for the hematopoietic datasets show that SMNN better mixed cells of all the three cell types across the two batches (**Fig 4a-c**), and reduced the differentiation between the two batches by up to 8.2% on top of MNN corrected results (**Fig 4d-f and S6 Fig**), demonstrating that our SMNN method can more effectively mitigate batch effect.

More importantly, the wrongly matched cell pairs may wipe out the distinguishing features of cell types. This is mainly because, for a pair of cells from two different cell types, the true biological differentiations between them would be considered as technical biases and subsequently removed in the correction process. Compared to MNN, SMNN also appears to more accurately recover cell-type specific features: clustering accuracy using SMNN-corrected data increases substantially in all the three real datasets (by 7.6 to 42.3% when measured by ARI) (**Fig 7**). Furthermore, we observe power enhancement in detecting DEGs between different cell types in the data after SMNN correction than MNN (**Fig 5 and 6 and S7-12 Fig**). Specifically, the true positive rates of the DEGs identified by SMNN were improved by up to 841% and 54.0% than those by MNN when compared to the two set of

working truth, respectively (**Fig 6c and d and S8-12 Fig**). Moreover, GO term enrichment results show that, the up-regulated DEGs identified only in SMNN-corrected GMP and MEP cells were involved in immune process and blood coagulation, respectively (**S7f and h Fig**), which accurately reflect the major features of these two cell types [20]. Similarly, DEGs identified between T and B cells after SMNN correction are also biologically more relevant than those identified after MNN correction (**S13b and c Fig**). These results suggest that SMNN can eliminate the overcorrection between different cell types and thus maintains more biological features in corrected data than MNN. Efficient removal of batch effects at reduced cost of biological information loss, manifested by SMNN in our extensive simulated and real data evaluations, empowers valid and more powerful downstream analysis.

In summary, extensive simulation and real data benchmarking suggest that our SMNN can not only better rescue biological features and thereof provide improved cluster results, but also facilitate the identification of biologically relevant DEGs. Therefore, we anticipate that our SMNN is valuable for integrated analysis of multiple scRNA-seq datasets, accelerating genetic studies involving single-cell dynamics.

Materials and methods

Simulation Framework

We simulated two scenarios, orthogonal and non-orthogonal, to compare the performance of MNN and SMNN. The difference between the two scenarios lies in the directions of the true underlying batch effect vectors with respect to those of the biological effects.

Baseline simulation

Our baseline simulation framework, similar to that adopted in the MNN paper, contains two steps:

Firstly, different batches of data are independently generated from a Gaussian mixture model to represent a low dimensional biological space, with each component in the mixture corresponding to one cell type. Specifically, we consider two batches with gene expression matrix X_k and Y_l, each follows a three-component Gaussian mixture model in a three-dimensional space, representing the low (here three) dimensional biological space.

$$X_k \sim \sum_{i=1}^{3} w_{1i} N(\mu_{1i}, I_3), with \sum_{i=1}^{3} w_{1i} = 1, and \;\; w_{11}, w_{12}, w_{13} \geq 0, for\; k = 1,2, \dots, n_1 \qquad (1)$$

$$Y_l \sim \sum_{j=1}^{3} w_{2j} N(\mu_{2j}, I_3), with \sum_{j=1}^{3} w_{2j} = 1, and\ \ w_{21}, w_{22}, w_{23} \geq 0, for\ l = 1,2, \dots, n_2 \quad (2)$$

where μ_{1i} is the vector specifying cell-type specific means for the three cell types in the first batch, reflecting the biological effect; similarly for μ_{2j}; n_1 and n_2 is the total number of cells in the first and second batch, respectively; w_{1i} and w_{2j} are the different mixing coefficients for the three cell types in the two batches; and I_3 is the three dimensional identity matrix with diagonal entries as ones and the rest entries as zeros. In our simulations, we set $n_1 = 1000, n_2 = 1100$ and

$$(w_{11}, w_{12}, w_{13}) = (0.3, 0.5, 0.2) \quad (3)$$

$$(w_{21}, w_{22}, w_{23}) = (0.25, 0.5, 0.25) \quad (4)$$

Secondly, we project the low dimensional data with batch effect to the high dimensional gene expression space. We map both datasets to G = 50 dimensions by linear transformation using the same random Gaussian matrix $\boldsymbol{P}$, to simulate high-dimensional gene expression profiles.

$$\widetilde{X_k} = \boldsymbol{P}X_k, for\ k = 1,2, \dots, n_1 \quad (5)$$

$$\widetilde{Y_l} = \boldsymbol{P}Y_l, for\ l = 1,2, \dots, n_2 \quad (6)$$

Here $\boldsymbol{P}$ is a G×3 Gaussian random matrix with each entry simulated from the standard normal distribution.

Introduction of batch effects

In the MNN paper, batch effects are directly introduced in the high dimensional gene expression space. Specifically, a Gaussian random vector $b = (b_1, b_2, \dots, b_G)^T$ is simulated and added to the second dataset via the following:

$$X_{Observed,k} = \widetilde{X_k} + \varepsilon_{1,k}, for\ k = 1,2, \dots, n_1 \quad (7)$$

$$Y_{Observed,l} = \widetilde{Y_l} + b + \varepsilon_{2,l}, for\ l = 1,2, \dots, n_2 \quad (8)$$

where $\widetilde{X_k}$ and $\widetilde{Y_l}$ are projected high-dimensional gene expression profiles; $\varepsilon_{1,k}$ and $\varepsilon_{2,l}$ are independent random noises added to the expression of each "gene" for each cell in the two batches.

In our simulations, we adopt a different approach: we introduce batch effects in the low dimensional biological space. Specifically, we simulate a bias vector $c = (c_1, c_2, c_3)^T$ in the biological space:

$$X_{Observed,k} = \widetilde{X_k} + \varepsilon_{1,k} = \boldsymbol{P}X_k + \varepsilon_{1,k}, for\ k = 1,2, \dots, n_1 \quad (9)$$

$$Y_{Observed,l} = \widetilde{Y_{SMNN,l}} + \varepsilon_{2,l} = \boldsymbol{P}(Y_l + c) + \varepsilon_{2,l} = \boldsymbol{P}Y_l + Pc + \varepsilon_{2,l}, for\ l = 1,2,\dots,n_2 \quad (10)$$

Comparing our simulation framework with that employed in the MNN paper, we would like to note the following:

1) For any vector c, there is a corresponding vector $b = \boldsymbol{P}c$ given a fixed projection matrix $\boldsymbol{P}$. This implies that our approach generates data that are special cases of those from MNN. In particular, since $(b)_l = (\boldsymbol{P}c)_l = \sum_{i=1}^{G} P_{li}c_i \sim N(0, \sum_{i=1}^{G} {c_i}^2)$, if $\sum_{i=1}^{G} {c_i}^2 = 1$, our approach becomes equivalent to generating a standard Gaussian random vector.
2) Our formulation allows flexible modeling of the biological effects and batch effects in the same low dimensional biological space. Specifically,

$$\mu_{2i} = \mu_{1i} + c, for\ i = 1,2,3. \quad (11)$$

Note that $(\mu_{1j} - \mu_{1i})c = 0, for\ i \neq j \in \{1,2,3\}$ in the orthogonal case and $(\mu_{1j} - \mu_{1i})c \neq 0, for\ i \neq j \in \{1,2,3\}$ in the non-orthogonal case.

In summary, our simulation framework, allowing flexible manipulation of biological and batch effects in the same low dimensional space, is effectively a special case of that adopted in the MNN paper.

The two scenarios

As aforementioned, we consider two scenarios, orthogonal case and non-orthogonal case. Orthogonality is defined in the sense that biological differences (that is, mean difference between any two clusters/cell-types), are orthogonal to those from batch effects.

Leveraging the simulation framework described before, we simulate two scenarios via the following:

1) In the orthogonal case, we set $c = (0,0,2)^T$
   a. $\mu_{11} = (5,0,0)^T, \mu_{12} = (0,0,0)^T, \mu_{13} = (0,5,0)^T$
   b. $\mu_{21} = (5,0,2)^T, \mu_{22} = (0,0,2)^T, \mu_{23} = (0,5,2)^T$
2) In the non-orthogonal case, we set $c = (0,5,2)^T$
   a. $\mu_{11} = (5,0,0)^T, \mu_{12} = (0,0,0)^T, \mu_{13} = (0,5,0)^T$
   b. $\mu_{21} = (5,5,2)^T, \mu_{22} = (0,5,2)^T, \mu_{23} = (0,10,2)^T$

Performance evaluation

MNN and SMNN share the goal to correct batch effects. Mathematically, using the notations introduced in baseline simulation, the goal translates into de-biasing vector c (which would be effectively reduced to b in the orthogonal

case). Without loss of generality and following MNN, we treat the first batch as the reference and correct the second batch $\{Y_{Observed,l}: l = 1, \dots, n_2\}$ to the first batch $\{X_{Observed,k}: k = 1, \dots, n_1\}$. Denote the corrected values from MNN and SMNN as $\{\widehat{Y_{MNN,l}}: l = 1, \dots, n_2\}$ and $\{\widehat{Y_{SMNN,l}}: l = 1, \dots, n_2\}$, respectively.

To measure the performance of the two correction methods, we utilize the Frobenius norm [19] to define the loss function:

$$L(\hat{Y}, \tilde{Y}) = \left\|\tilde{\boldsymbol{Y}} - \hat{Y}\right\|_F = \sqrt{\sum_{l=1}^{n_2} \left\|\tilde{Y}_l - \hat{Y}_l\right\|^2} = \sqrt{\sum_{l=1}^{n_2}\sum_{g=1}^{G} \left|\widehat{Y_{l,g}} - \widehat{Y_{l,g}}\right|^2} \quad (12)$$

where $\tilde{\boldsymbol{Y}} = [\tilde{Y}_1, \dots, \tilde{Y}_k, \dots, \widetilde{Y_{n_2}}]$, $\hat{Y} = [\hat{Y}_1, \dots, \hat{Y}_k, \dots, \widehat{Y_{n_2}}]$. Note that $\tilde{\boldsymbol{Y}}$ is the simulated true profiles introduced in equations (**Error! Reference source not found.**) and (**Error! Reference source not found.Error! Reference source not found.Error! Reference source not found.Error! Reference source not found.**) before batch effects, and noises are introduced in equations (**Error! Reference source not found.**) and (**Error! Reference source not found.**). Since MNN conducts cosine normalization to the input and the output, we use cosine-normalized $\tilde{\boldsymbol{Y}}$ when calculating the above loss function.

Real data benchmarking

To assess the performance of SMNN in real data, we applied both SMNN and MNN to two hematopoietic scRNA-seq datasets, generated using different sequencing platforms, MARs-seq and SMART-seq2 (**S1 Table**) [10,21]. The first batch produced by MARs-seq consists of 1920 cells of six major cell types, and the second batch generated by SMART-seq2 contains 2730 of three cell types, where three cell types, common myeloid progenitor (CMP) cells, granulocyte-monocyte progenitors (GMP) cells and megakaryocyte-erythrocyte progenitor (MEP) cells, are shared between these two batches (here the two datasets). Batch effect correction was carried out using both MNN and SMNN, following the default instructions. Cell type labels were fed to SMNN directly according to the annotation from the original papers. To better compare the performance between MNN and SMNN, only the three cell types shared between the two batches were extracted for our downstream analyses. The corrected results of all the three cell types together, as well as for each of them separately, were visualized by t-SNE using *Rtsne* function from *Rtsne* package [9,22]. In order to qualify the mixture of single cells using both batch correction methods, we calculated: 1)

the distance for the cells within each cell type in batch 2 to the centroid of the corresponding cell group in batch 1; and 2) the overall variance in the two batches.

To measure the separation of cell types after correction, we additionally attempted to detect differentially expressed genes (DEGs) between different cell types in both SMNN and MNN corrected datasets. The corrected expression matrices of the two batches were merged and DEGs were detected by Seurat using Wilcoxon rank sum test. Genes with an adjusted p-value < 0.01 were considered as differentially expressed. Gene ontology (GO) enrichment analysis was performed for the DEGs exclusively identified by SMNN using *clusterProfiler* [23]. Because there is no ground truth for DEGs, we further identified DEGs between different cell types within corrected batch 2 and then compared them to those identified in uncorrected batch 1 and uncorrected batch 2, which supposedly are not affected by the choice of batch effect correction method. True positive rate (TPR) was computed for each comparison.

Additionally, we also performed batch effect correction on another two tissues/cell lines, pancreas [24,25] human peripheral blood mononuclear cells (PBMCs) [26], again using both SMNN and MNN. Single cell clustering was applied to batch-effects corrected gene expression matrices following the pipeline described in MNN paper. Cell type labels before correction were considered as ground truth and Adjusted Rand Index (ARI) [27] was employed to measure the clustering similarity before and after correction:

$$ARI(L_q, L_s) = \frac{\sum_{q,s}\binom{n_{qs}}{2} - \left[\sum_q \binom{n_q}{2}\sum_s \binom{n_s}{2}\right] / \binom{n}{2}}{\frac{1}{2}\left[\sum_q \binom{n_q}{2} + \sum_s \binom{n_s}{2}\right] - \left[\sum_q \binom{n_q}{2}\sum_s \binom{n_s}{2}\right] / \binom{n}{2}} \quad (13)$$

where n_e and n_t are the single cell numbers in cluster q and s, respectively; n_{qs} is the number of single cells shared between clusters q and s; and n is the total number of single cells. ARI ranges from 0 to 1, where a higher value represents a higher level of similarity between the query and subject clusters.

Acknowledgements

This research was supported by the National Institute of Health grant R01 HL129132 (awarded to YL).

Author Contributions

Conceptualization: Yuchen Yang, Gang Li, Yun Li.

Formal analysis: Yuchen Yang, Gang Li, Huijun Qian.

Funding acquisition: Yun Li.

Methodology: Yuchen Yang, Gang Li.

Software: Yuchen Yang, Gang Li.

Supervision: Yun Li.

Visualization: Yuchen Yang, Gang Li, Huijun Qian.

Writing – original draft: Yuchen Yang, Gang Li.

Writing – review & editing: Yun Li, Kirk C. Wilhelmsen, Yin Shen.

REFERENCES

1. Rozenblatt-Rosen O, Stubbington MJ, Regev A, Teichmann SA (2017) The human cell atlas: from vision to reality. Nature News 550: 451.
2. Stegle O, Teichmann SA, Marioni JC (2015) Computational and analytical challenges in single-cell transcriptomics. Nature Reviews Genetics 16: 133.
3. Chen M, Zhou X (2017) Controlling for confounding effects in single cell RNA sequencing studies using both control and target genes. Scientific reports 7: 13587.
4. Stuart T, Satija R (2019) Integrative single-cell analysis. Nature Reviews Genetics: 1.
5. Smyth GK (2005) Limma: linear models for microarray data. Bioinformatics and computational biology solutions using R and Bioconductor: Springer. pp. 397-420.
6. Johnson WE, Li C, Rabinovic A (2007) Adjusting batch effects in microarray expression data using empirical Bayes methods. Biostatistics 8: 118-127.
7. Leek JT (2014) Svaseq: removing batch effects and other unwanted noise from sequencing data. Nucleic acids research 42: e161-e161.
8. Haghverdi L, Lun AT, Morgan MD, Marioni JC (2018) Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. Nature biotechnology 36: 421.
9. Van Der Maaten L (2014) Accelerating t-SNE using tree-based algorithms. The Journal of Machine Learning Research 15: 3221-3245.
10. Nestorowa S, Hamey FK, Sala BP, Diamanti E, Shepherd M, et al. (2016) A single-cell resolution map of mouse hematopoietic stem and progenitor cell differentiation. Blood 128: e20-e31.
11. Spitzer MH, Gherardini PF, Fragiadakis GK, Bhattacharya N, Yuan RT, et al. (2015) An interactive reference framework for modeling a dynamic immune system. Science 349: 1259425.
12. Stuart T, Butler A, Hoffman P, Hafemeister C, Papalexi E, et al. (2019) Comprehensive Integration of Single-Cell Data. Cell 177: 1888-1902 e1821.
13. Duò A, Robinson MD, Soneson C (2018) A systematic performance evaluation of clustering methods for single-cell RNA-seq data. F1000Research 7.
14. Kiselev VY, Andrews TS, Hemberg M (2019) Challenges in unsupervised clustering of single-cell RNA-seq data. Nature Reviews Genetics: 1.
15. Zhu L, Lei J, Klei L, Devlin B, Roeder K (2019) Semisoft clustering of single-cell data. Proceedings of the National Academy of Sciences 116: 466-471.
16. Sun Z, Chen L, Xin H, Jiang Y, Huang Q, et al. (2019) A Bayesian mixture model for clustering droplet-based single-cell transcriptomic data from population studies. Nature communications 10: 1649.
17. Butler A, Hoffman P, Smibert P, Papalexi E, Satija R (2018) Integrating single-cell transcriptomic data across different conditions, technologies, and species. Nature biotechnology 36: 411.

18. Yang Y, Huh R, Culpepper HW, Lin Y, Love MI, et al. (2019) SAFE-clustering: Single-cell aggregated (from ensemble) clustering for single-cell RNA-seq data. Bioinformatics 35: 1269-1277.
19. Van Loan CF, Golub GH (1983) Matrix computations: Johns Hopkins University Press.
20. Lieu YK, Reddy EP (2012) Impaired adult myeloid progenitor CMP and GMP cell function in conditional c-myb-knockout mice. Cell Cycle 11: 3504-3512.
21. Paul F, Arkin Ya, Giladi A, Jaitin DA, Kenigsberg E, et al. (2015) Transcriptional heterogeneity and lineage commitment in myeloid progenitors. Cell 163: 1663-1677.
22. Maaten Lvd, Hinton G (2008) Visualizing data using t-SNE. Journal of machine learning research 9: 2579-2605.
23. Yu G, Wang L-G, Han Y, He Q-Y (2012) clusterProfiler: an R package for comparing biological themes among gene clusters. Omics: a journal of integrative biology 16: 284-287.
24. Grün D, Muraro MJ, Boisset J-C, Wiebrands K, Lyubimova A, et al. (2016) De novo prediction of stem cell identity using single-cell transcriptome data. Cell stem cell 19: 266-277.
25. Muraro MJ, Dharmadhikari G, Grün D, Groen N, Dielen T, et al. (2016) A single-cell transcriptome atlas of the human pancreas. Cell systems 3: 385-394. e383.
26. Zheng GX, Terry JM, Belgrader P, Ryvkin P, Bent ZW, et al. (2017) Massively parallel digital transcriptional profiling of single cells. Nature communications 8: 14049.
27. Hubert L, Arabie P (1985) Comparing partitions. Journal of classification 2: 193-218.

Figure Legend

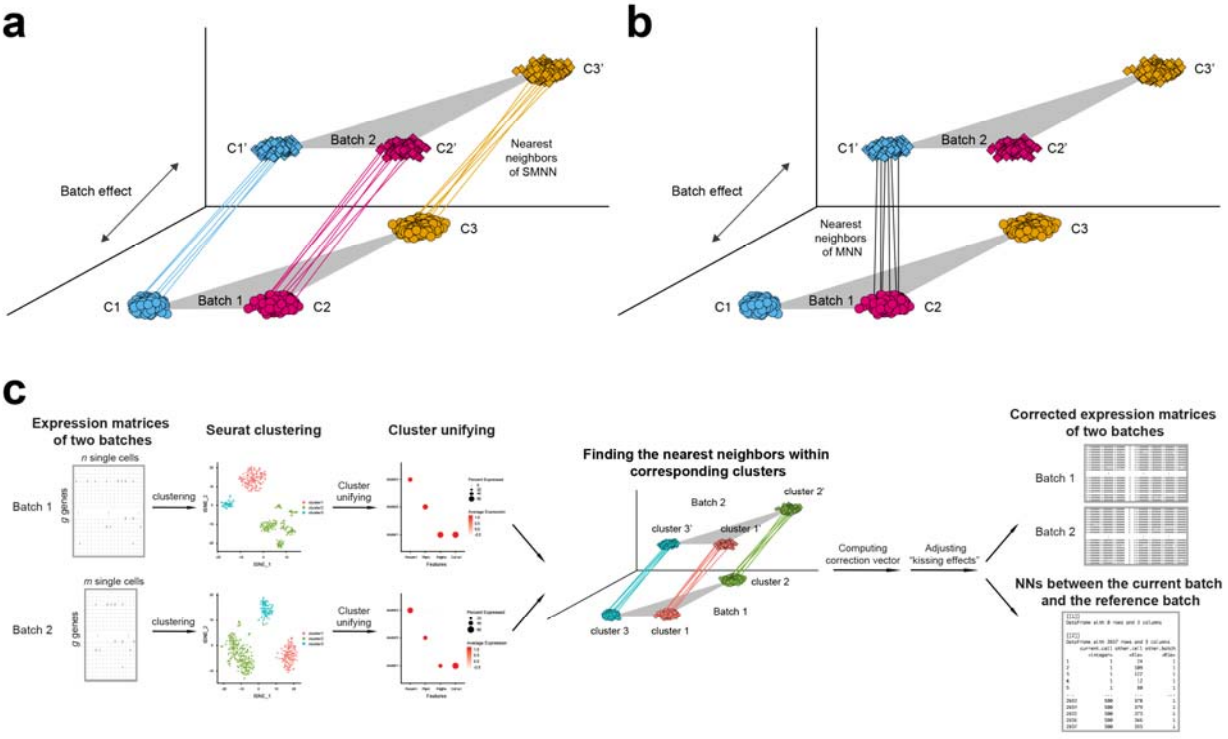


Fig 1. Overview of SMNN. Schematics for detecting mutual nearest neighbors between two batches under a non-orthogonal scenario **(a)** in SMNN; and **(b) in** MNN. **(c)** Workflow of SMNN. Single cell clustering is first performed within each batch using Seurat; and then SMNN takes user-specified marker gene information for each cell type to match clusters/cell types across batches. With the clustering and cluster-specific marker gene information, SMNN searches mutual nearest neighbors within each cell type and performs batch effect correction accordingly.

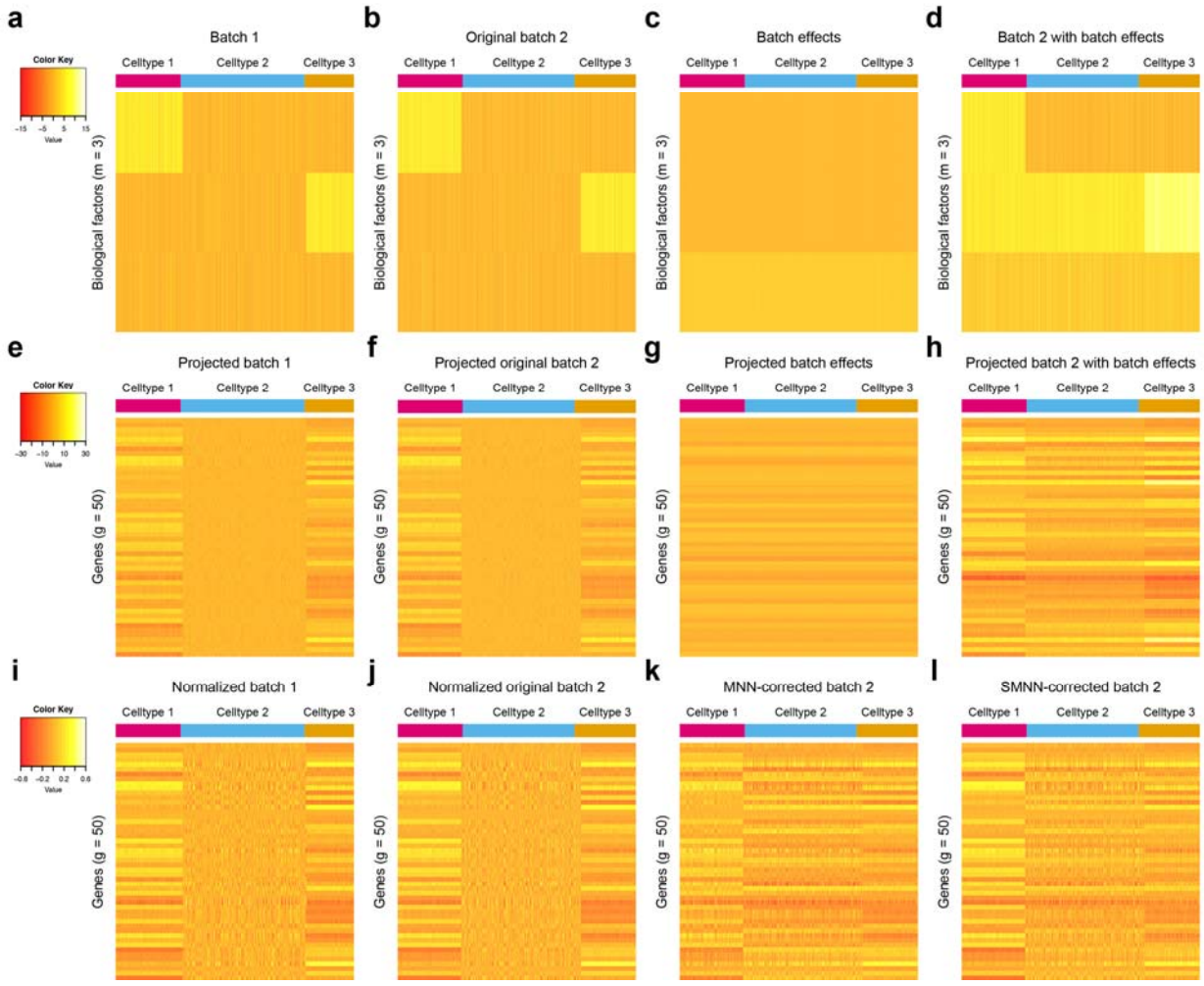


Fig 2. Heatmap of gene expression matrices for simulated data under non-orthogonal scenario. (a), **(b)**, **(c)** and **(d)** show the 3-dimensional biological space with rows of each heatmap representing biological factors and columns corresponding to single cells. **(e)**, **(f)**, **(g)** and **(h)** show the high dimensional gene expression profiles with rows corresponding to genes and columns again representing single cells. **(a)**, **(e)** and **(i)** correspond to the batch 1, and **(b)**, **(f)** and **(j)** correspond to batch 2. **(c)** and **(g)** provide a visualization for the direction of batch effects in low-dimension biological space and high-dimension gene expression spaces, respectively. **(d)** and **(h)**, sum of **(b)** and **(c)** and sum of **(f)** and **(g)** respectively, are "observed" data for cells in batch 2 in low and high dimensional space respectively. **(i)** and **(j)** are the cosine-normalized data for batch 1 and original batch 2. Note "original" is in the sense that no batch effects have been introduced to the data yet. **(k)** and **(l)** are the MNN and SMNN corrected results, respectively.

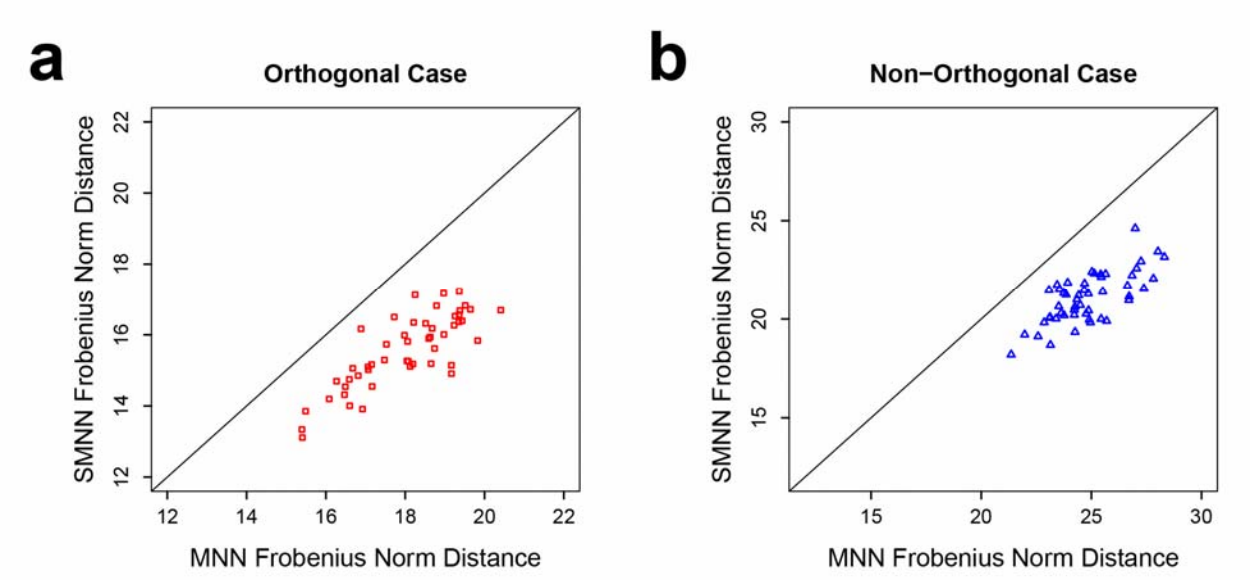


Fig 3. Frobenius norm distance between two batches after SMNN and MNN correction in simulation data under orthogonal (left) and non-orthogonal scenarios (right).

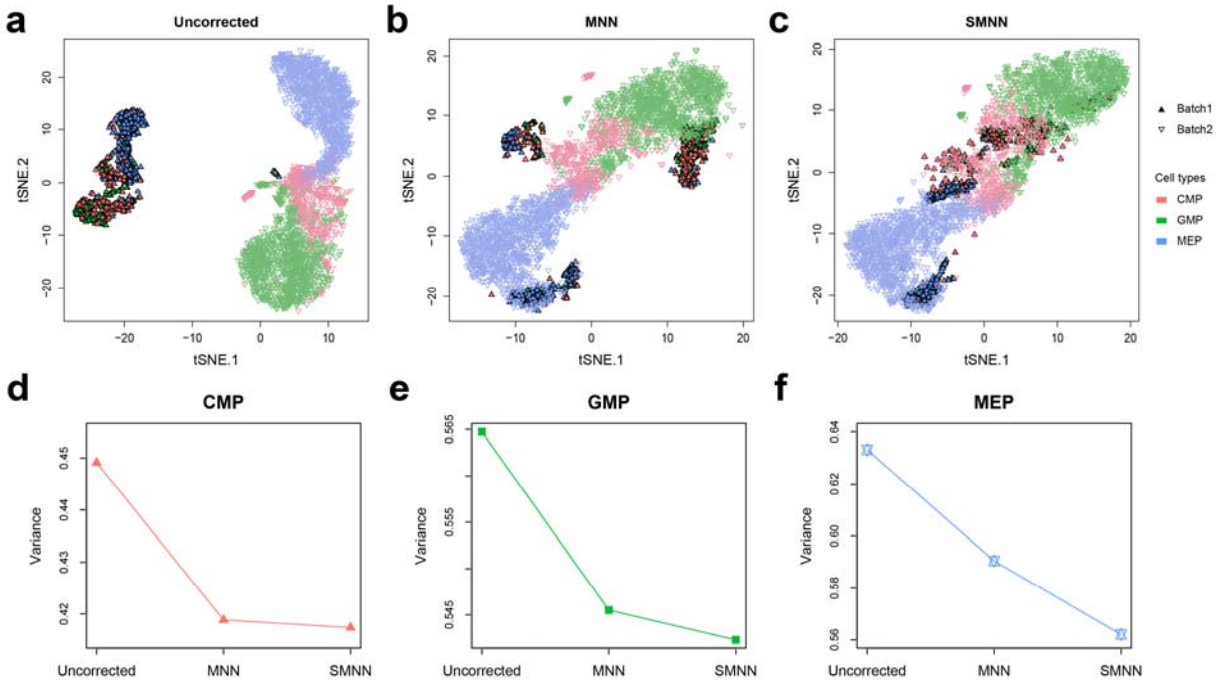


Fig 4. Performance comparison between SMNN and MNN in two hematopoietic datasets. (**a-c**) t-SNE plots for two hematopoietic datasets before and after correction with SMNN and MNN. Solid and inverted triangle represent the first and second batch, respectively; and different cell types are shown in different colors. (**d-f**) Variance comparisons for the three different cell types: CMP (**d**), GMP (**e**) and MEP (**f**), in merged data by pooling batch 1 with different versions of batch 2. Specifically, we show the following three versions of batch 2 data: original observed (uncorrected), MNN-corrected (MNN) and SMNN corrected (SMNN). The SMNN corrected version resulted in variances slightly (for CMP and GMP cells) or substantially (for MEP cells) smaller than those from the MNN corrected version, suggesting improved mixing of cells across batches.

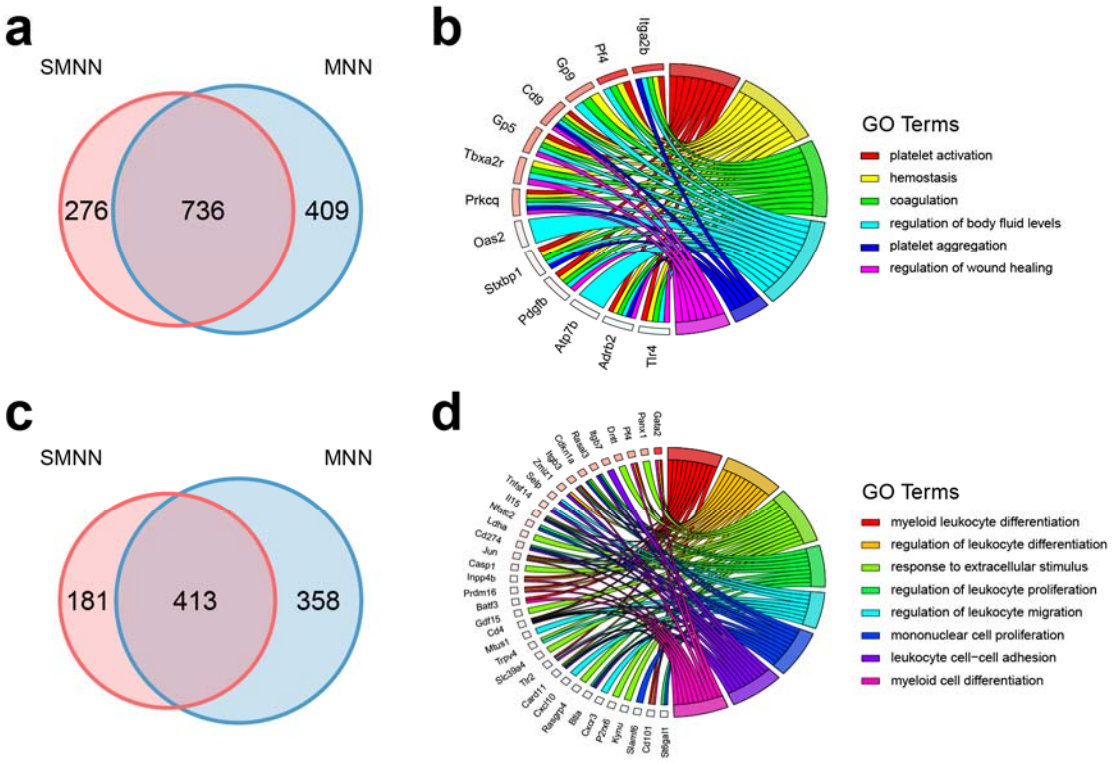


Fig 5. Comparison of differentially expressed genes (DEGs), identified in the merged dataset by pooling batch 1 data with batch 2 data after SMNN and MNN correction. (**a**) Overlap of DEGs up-regulated in CMP over GMP after SMNN and MNN correction. (**b**) Feature enriched GO terms and the corresponding DEGs up-regulated in CMP over GMP. (**c**) Overlap of DEGs up-regulated in CMP over MEP after SMNN and MNN correction. (**d**) Feature enriched GO terms and the corresponding DEGs up-regulated in CMP over MEP.

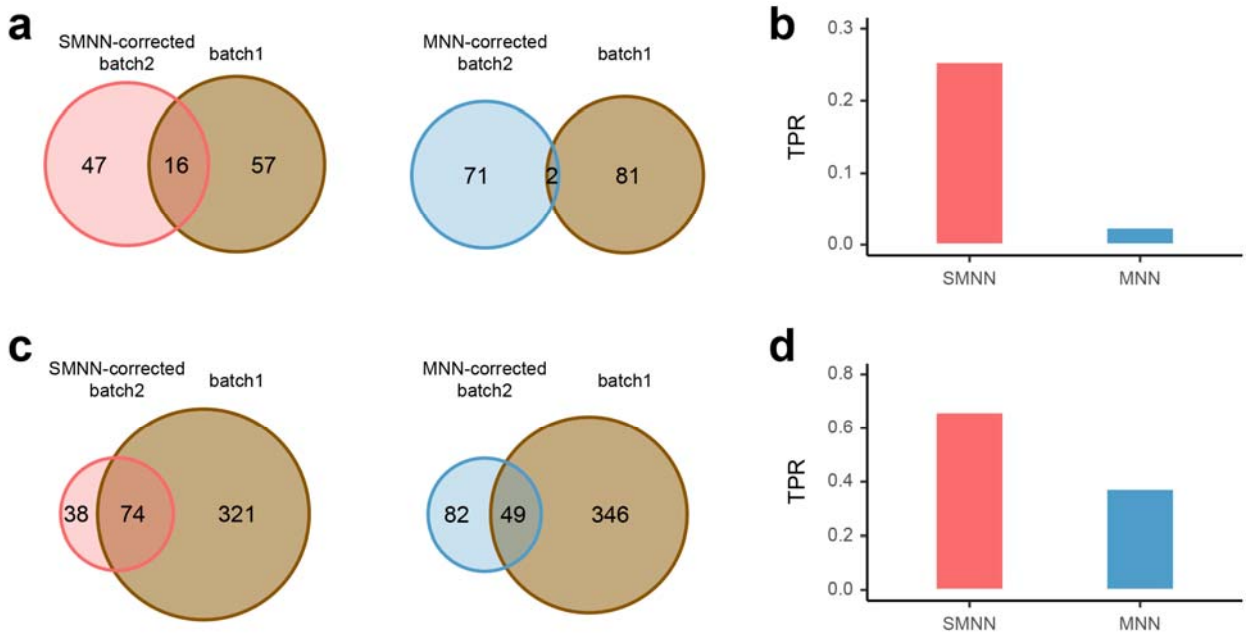
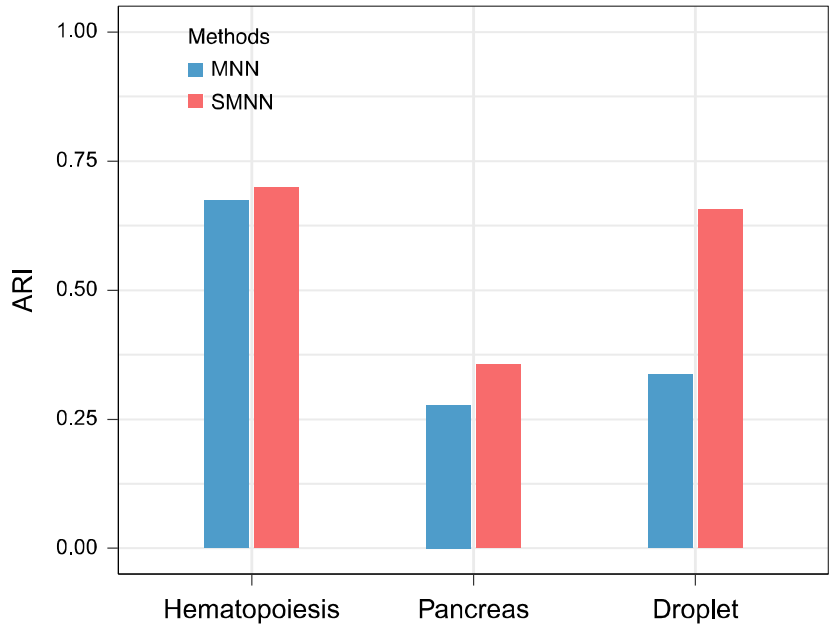


Fig 6. Reproducibility of DEGs (between CMP and GMP), identified in uncorrected batch 1 and in SMNN or MNN-corrected batch 2. (**a**) Reproducibility of DEGs up-regulated in CMP over GMP, detected in batch 1, versus SMNN (left) or MNN-corrected (right) batch 2. (**b**) True positive rate (TPR) of the DEGs (between CMP and GMP) identified in batch 2 after SMNN and MNN correction. (**c**) Reproducibility of DEGs up-regulated in GMP over CMP, identified in the uncorrected batch 1, and in SMNN (left) or MNN-corrected (right) batch 2. (**d**) TPR of the DEGs up-regulated in GMP over CMP identified in batch 2 after SMNN and MNN correction.



459

460 **Fig 7. Clustering accuracy in three datasets after batch effect correction.** Adjusted Rand Index (ARI) is

461 employed to measure the similarity between clustering results before and after batch effect correction.