Flexible Experimental Designs for Valid Single-cell RNA-sequencing Experiments Allowing Batch Effects Correction

Fangda Song, Ga Ming Angus Chan and Yingying Wei*

Department of Statistics, The Chinese University of Hong Kong, Hong Kong SAR, China

Abstract

Despite their widespread applications, single-cell RNA-sequencing (scRNA-seq) experiments are still plagued by batch effects and dropout events. Although the completely randomized experimental design has frequently been advocated to control for batch effects, it is rarely implemented in real applications due to time and budget constraints. Here, we mathematically prove that under two more flexible and realistic experimental designs—the "reference panel" and the "chain-type" designs—true biological variability can also be separated from batch effects. We develop **B**atch effects correction with **U**nknown **S**ubtypes for scRNA-**seq** data (BUSseq), which is an interpretable Bayesian hierarchical model that closely follows the data-generating mechanism of scRNA-seq experiments. BUSseq can simultaneously correct batch effects, cluster cell types, impute missing data caused by dropout events, and detect differentially expressed genes without requiring a preliminary normalization step. We demonstrate that BUSseq outperforms existing methods with simulated and real data.

Keywords: Batch effects; Experimental design; Single-cell RNA-seq experiments; Model-based clustering; Integrative analysis

*Correspondence should be addressed to Yingying Wei (yweicuhk@gmail.com)

Introduction

Single-cell RNA-sequencing (scRNA-seq) technologies enable the measurement of the transcriptome of individual cells, which provides unprecedented opportunities to discover cell types and understand cellular heterogeneity [1]. However, like the other high-throughput technologies [2–4], scRNA-seq experiments can suffer from severe batch effects [5]. Moreover, compared to bulk RNA-seq data, scRNA-seq data can have an excessive number of zeros that result from dropout events—that is, the expressions of some genes are not detected even though they are actually expressed in the cell due to amplification failure prior to sequencing [6]. Consequently, despite the widespread adoption of scRNA-seq experiments, the design of a valid scRNA-seq experiment that allows the batch effects to be removed, the biological cell types to be discovered, and the missing data to be imputed remains an open problem.

One of the major tasks of scRNA-seq experiments is to identify cell types for a population of cells [1]. The cell type of each individual cell is unknown and is often the target of inference. Classic batch effects correction methods, such as Combat [7] and SVA [8, 9], are designed for bulk experiments and require knowledge of the subtype information of each sample a prior. For scRNA-seq data, this subtype information corresponds to the cell type of each individual cell. Clearly, these methods are thus infeasible for scRNA-seq data. Alternatively, if one has knowledge of a set of control genes whose expression levels are constant across cell types, then it is possible to apply RUV [10, 11]. However, selecting control genes is often difficult for scRNA-seq experiments.

To identify unknown subtypes, MetaSparseKmeans [12] jointly clusters samples across batches. Unfortunately, MetaSparseKmeans requires all subtypes to be present in each batch. Suppose that we conduct scRNA-seq experiments for blood samples from a healthy individual and a leukemia patient, one person per batch. Although we can anticipate that the two batches will share T cells and B cells, we do not expect that the healthy individual will have cancer cells as the leukemia patient. Therefore, MetaSparseKmeans is too restrictive for many scRNA-seq experiments.

The mutual-nearest-neighbor (MNN) based approaches, including MNN [13] and Scanorama [14], allow each batch to contain some but not all cell types. However, these methods require batch effects to be almost orthogonal to the biological subspaces and much smaller than

the biological variations between different cell types [13]. These are strong assumptions and cannot be validated at the design stage of the experiments. Seurat [15, 16], LIGER [17] and scMerge [18] attempt to identify shared variations across batches by low-dimensional embeddings and treat them as shared cell types. However, they may mistake the technical artifacts as the biological variability of interest if some batches share certain technical noises, for example when each patient is measured by several batches. To handle severe batch effects for microarray data, Luo and Wei [19] developed BUS to simultaneously cluster samples across multiple batches and correct batch effects. However, none of the above methods considers features unique to scRNA-seq data, such as the count nature of the data, over-dispersion [20], dropout events [6], or cell-specific size factors [21]. ZIFA [22] and ZINB-WaVE [23] incorporate dropout events into the factor model, whereas scVI [24] and SAVER-X [25] couple the modeling of dropout events with neural networks. However, as is the case with the other state-of-the-art methods, these papers do not discuss the designs of scRNA-seq experiments under which their methods are applicable.

Nevertheless, it is crucial to understand the conditions under which biological variability can be separated from technical artifacts. Obviously, for completely confounded designs—for example one in which batch 1 measures cell type 1 and 2, whereas batch 2 measures cell type 3 and 4—no method is applicable.

Here, we propose Batch effects correction with Unknown Subtypes for scRNA-seq data (BUSseq), an interpretable hierarchical model that simultaneously corrects batch effects, clusters cell types, and takes care of the count data nature, the overdispersion, the dropout events, and the cell-specific size factors of scRNA-seq data. We mathematically prove that it is legitimate to conduct scRNA-seq experiments under not only the commonly advocated completely randomized design [1, 5, 26, 27], in which each batch measures all cell types, but also the "reference panel" design and the "chain-type" design, which allow some cell types to be missing from some batches. Furthermore, we demonstrate that BUSseq outperforms the existing approaches in both simulation data and real applications. The theoretical results answer the question about when we can integrate multiple scRNA-seq datasets and analyze them jointly. We envision that the proposed experimental designs will be able to guide biomedical researchers and help them to design better scRNA-seq experiments.

Results

BUSseq is an interpretable hierarchical model for scRNA-seq

We develop a hierarchical model BUSseq that closely mimics the data generating procedure of scRNA-seq experiments (**Fig. 1a** and **Supplementary Fig. 1**). Given that we have measured B batches of cells each with a sample size of n_b, let us denote the underlying gene expression level of gene g in cell i of batch b as X_{big}. X_{big} follows a negative binomial distribution with mean expression level μ_{big} and a gene-specific and batch-specific overdispersion parameter ϕ_{bg}. The mean expression level is determined by the cell type W_{bi} with the cell type effect β_{gk}, the log-scale baseline expression level α_g, the location batch effect ν_{bg}, and the cell-specific size factor δ_{bi}. The cell-specific size factor δ_{bi} characterizes the impact of cell size, library size and sequencing depth. It is of note that the cell type W_{bi} of each individual cell is unknown and is our target of inference. Therefore, we assume that a cell on batch b comes from cell type k with probability $P(W_{bi} = k) = \pi_{bk}$ and the proportions of cell types $(\pi_{b1}, \cdots, \pi_{bK})$ vary among batches.

Unfortunately, it is not always possible to observe the expression level X_{big}. Without dropout ($Z_{big} = 0$), we can directly observe $Y_{big} = X_{big}$. However, if a dropout event occurs ($Z_{big} = 1$), then we observe $Y_{big} = 0$ instead of X_{big}. It has been noted that highly expressed genes are less-likely to suffer from dropout events [6]. We thus model the dependence of the dropout rate $P(Z_{big} = 1|X_{big})$ on the expression level using a logistic regression with batch-specific intercept γ_{b0} and odds ratio γ_{b1}.

Noteworthy, BUSseq includes the negative binomial distribution without zero inflation as a special case. When all cells are from a single cell type and the cell-specific size factor δ_{bi} is estimated a priori according to spike-in genes, BUSseq can reduce to a form similar to BASiCS [20].

We only observe Y_{big} for all cells in the B batches and the total G genes. We conduct statistical inference under the Bayesian framework and develop a Markov chain Monte Carlo (MCMC) algorithm [28]. Based on the parameter estimates, we can learn the cell type for each individual cell, impute the missing underlying expression levels X_{big} for dropout events, and identify genes that are differentially expressed among cell types. Moreover, our algorithm can

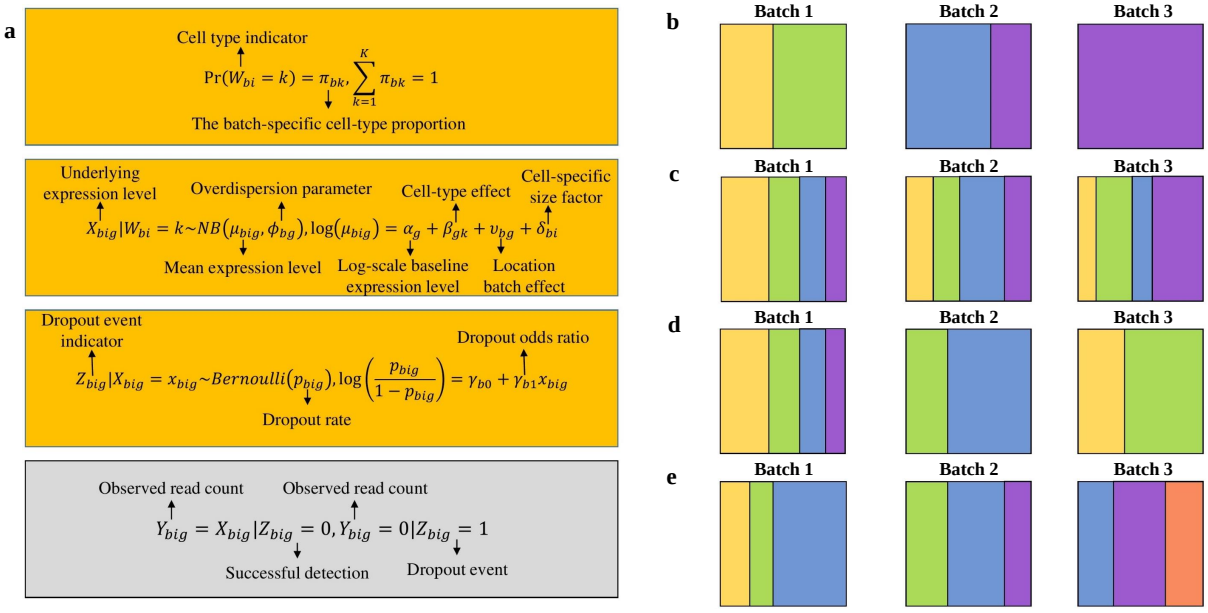

Figure 1: Illustration of the BUSseq model and various types of experimental designs. **(a)** The hierarchical structure of the BUSseq model. Only Y_{big} in the grey rectangle is observed. **(b)** A confounded design that contains three batches. Each polychrome rectangle represents one batch of scRNA-seq data with genes in rows and cells in columns; and each color indicates a cell type. Batch 1 assays cells from cell types 1 and 2; batch 2 profiles cells from cell types 3 and 4; and batch 3 only contains cells from cell type 4. **(c)** The complete setting design. Each batch assays cells from all of the four cell types, although the cellular compositions vary across batches. **(d)** The reference panel design. Batch 1 contains cells from all of the cell types, and all the other batches have at least two cell types. **(e)** The chain-type design. Every two consecutive batches share two cell types. Batch 1 and Batch 2 share cell types 2 and 3; Batch 2 and Batch 3 share cell types 3 and 4 (see also **Supplementary Figs. 1 and 2**).

automatically detect the total number of cell types K that exists in the dataset according to the Bayesian information criterion (BIC) [29]. BUSseq also provides a batch-effect corrected version of count data, which can be used for downstream analysis as if all of the data were measured in a single batch. Details are in Methods and Supplementary Notes.

Valid experimental designs for scRNA-seq experiments

If a study design is completely confounded, as shown in **Fig. 1b**, then no method can separate biological variability from technical artifacts, because different combinations of batch-effect and cell-type-effect values can lead to the same probabilistic distribution for the observed data, which in statistics is termed a *non-identifiable* model. Formally, a model is

said to be *identifiable* if each probability distribution can arise from only one set of parameter values [30]. Statistical inference is impossible for non-identifiable models because two sets of distinct parameter values can give rise to the same probabilistic function. We prove that the BUSseq model is identifiable under conditions that are very easily met in reality. It is thus applicable to a wide range of experimental designs.

For the "complete setting," in which each batch measures all of the cell types (**Fig. 1c** and Theorem 1 in Methods), BUSseq is identifiable as long as: (I) the odds ratio γ_{b1}s in the logistic regressions for the dropout rates are negative for all of the batches, (II) every two cell types have more than one differentially expressed gene, and (III) the ratios of mean expression levels between two cell types $(\frac{exp(\beta_{1k})}{exp(\beta_{1\tilde{k}})}, \cdots, \frac{exp(\beta_{Gk})}{exp(\beta_{G\tilde{k}})})$ are different for each cell-type pair $(k, \tilde{k})$ (see Theorem 1 in Methods). Condition (I) requires that the highly expressed genes are less likely to have dropout events, which is routinely observed for scRNA-seq data [6]. Condition (II) always holds in reality. Because scRNA-seq experiments measure the whole transcriptome of a cell, condition (III) is also always met in real data. For example, if there exists one gene g such that for any two distinct cell-type pairs (k_1, k_2) and (k_3, k_4) their mean expression levels ratios $\frac{exp(\beta_{gk_1})}{exp(\beta_{gk_2})}$ and $\frac{exp(\beta_{gk_3})}{exp(\beta_{gk_4})}$ are not the same, then condition (III) is already satisfied.

The commonly advocated completely randomized experimental design falls into the "complete setting" design, whereas the latter further relaxes the assumption implied by the former that the cell-type proportions are almost the same for all batches. The identical composition of the cell population within each batch is a crucial requirement for traditional batch effects correction methods developed for bulk experiments such as Combat [13]. In contrast, BUSseq is not limited to this balanced design constraint and is applicable to not only the completely randomized design but also the general complete setting design.

Ideally, we would wish to adopt completely randomized experimental designs. However, in reality, it is always very challenging to implement complete randomization due to time and budget constraints. For example, when we recruit patients sequentially, we often have to conduct scRNA-seq experiments patient-by-patient rather than randomize the cells from all of the patients to each batch, and the patients may not have the same set of cell types. Fortunately, we can prove that BUSseq also applies to two sets of flexible experimental designs, which allow cell types to be measured in only some but not all of the batches.

Assuming that conditions (I)-(III) are satisfied, if there exists one batch that contains cells from all cell types and the other batches have at least two cell types (**Fig. 1d**), then BUSseq can tease out the batch effects and identify the true biological variability (see Theorem 2 in Methods). We call this setting the "reference panel design."

Sometimes, it can still be difficult to obtain a reference batch that collects all cell types. In this case, we can turn to the chain-type design, which requires every two consecutive batches to share two cell types (**Fig. 1e**). Under the chain-type design, given that conditions (I)-(III) hold, BUSseq is also identifiable and can estimate the parameters well (see Theorem 3 in Methods).

A special case of the chain-type design is when two common cell types are shared by all of the batches, which is frequently encountered in real applications. For instance, when blood samples are assayed, even if we perform scRNA-seq experiment patient-by-patient with one patient per batch, we know a priori that each batch will contain at least both T cells and B cells, thus satisfying the requirement of the chain-type design.

The key insight is that despite batch effects, differences between cell types remain constant across batches. The differences between a pair of cell types allow us to distinguish batch effects from biological variability for those batches that measure both cell types. Once batch effects have been identified, we can conduct joint clustering across batches with batch effects adjusted. In fact, BUSseq can separate batch effects from cell type effects under more general designs beyond the easily understood and commonly encountered reference panel design and chain-type design. If we regard each batch as a node in a graph and connect two nodes with an edge if the two batches share at least two cell types, then BUSseq is identifiable as long as the resulting graph is connected (see **Supplementary Fig. 2** and Theorem 4 in Methods).

For scRNA-seq data, dropout rate depends on the underlying expression levels. Such missing data mechanism is called missing not at random (MNAR) in statistics. It is very challenging to establish identifiability for MNAR. Miao et al. [31] showed that for many cases even when both the outcome distribution and the missing data mechanism have parametric forms, the model can be nonidentifiable. However, fortunately, despite the dropout events and the cell-specific size factors, by creating a set of functions similar to the probability generating function, we proved Theorems 1-4 (see their proofs in Supplementary Notes). The reference panel design, the chain-type design and the connected design liberalize researchers

from the ideal but often unrealistic requirement of the completely randomized design.

BUSseq accurately estimates the parameters and imputes the missing data

We first evaluate the performance of BUSseq via a simulation study. We simulate a dataset with four batches and a total of five cell types under the chain-type design (**Fig. 2a-d** and Theorem 3). Every two consecutive batches share at least two cell types, but none of the batches contains all of the cell types. The sample sizes for each batch are $(n_1, n_2, n_3, n_4) = (300, 300, 200, 200)$, and there are a total of 3,000 genes. In real datasets, batch effects are often much larger than the cell type effects (**Fig. 3a**) and not orthogonal to the cell type effects (**Supplementary Fig. 3**). In the simulation study, we choose the magnitude of the batch effects, cell type effects, the dropout rates, and the cell-specific size factors to mimic real data scenarios (**Fig. 3a**). The simulated observed data suffer from severe batch effects and dropout events (**Fig. 2d** and **Fig. 3c**). The dropout rates for the four batches are $26.79\%, 24.53\%, 28.36\%$ and 31.29%, with the corresponding total zero proportions given by $44.13\%, 48.85\%, 53.07\%$ and 61.38%.

BUSseq correctly identifies the presence of five cell types among the cells (**Fig. 2e**). Moreover, despite the dropout events, BUSseq accurately estimates the cell type effects β_{gk}s (**Fig. 2a and f**), the batch effects ν_{bg}s (**Fig. 2b and g**), and the cell-specific size factors δ_{bi}s (**Fig. 2j**). When controlling the Bayesian False Discovery Rate (FDR) at 0.05 [32, 33], we identify all intrinsic genes that differentiate cell types with the true FDR being 0.02 (Methods).

In the simulation study, we know the underlying expression levels X_{big}s. Therefore, we can compare them with our inferred expression levels $\widehat{X}_{big}$s based the observed data Y_{big}s which are subject to dropout events. **Fig. 2h** demonstrates that BUSseq can learn the underlying expression levels well. This success arises because BUSseq uses an integrative model to borrow strengths both across genes and across cells from all batches. As a result, BUSseq can achieve accurate estimation and imputation despite the dropout events.

Combat offers a version of data that have been adjusted for batch effects [7]. Here, we also provide batch-effects-corrected count data based on quantile matching (Methods). The

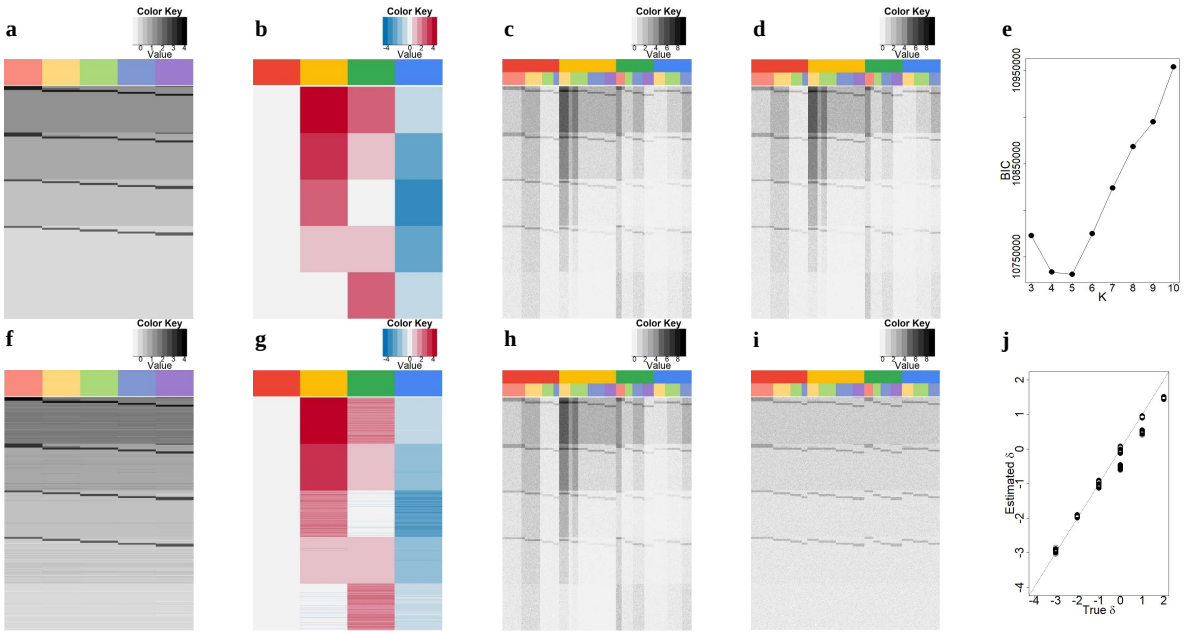

Figure 2: Patterns of the simulation study. **(a)** True log-scale mean expression levels for each cell type $\boldsymbol{\alpha} + \boldsymbol{\beta}$. Each row represents a gene, and each column corresponds to a cell type. The intrinsic genes that are differentially expressed between cell types can have high, medium high, median low or low expression levels. **(b)** True batch effects. Each row represents a gene, and each column corresponds to a batch. **(c)** True underlying expression levels $\boldsymbol{X}$. Each row represents a gene, and each column corresponds to a cell. The upper color bar indicates the batches, and the lower color bar represents the cell types. There are a total of 3,000 genes. The sample sizes for each batch are 300, 300, 200 and 200, respectively. **(d)** The simulated observed data $\boldsymbol{Y}$. The overall dropout rate is 27.3%, whereas the overall zero rate is 50.8%. **(e)** The BIC plot. The BIC attains the minimum at $K = 5$, identifying the true cell type number. **(f)** The estimated log-scale mean expression levels for each cell type $\hat{\boldsymbol{\alpha}} + \hat{\boldsymbol{\beta}}$. **(g)** Estimated batch effects. **(h)** Imputed expression levels $\widehat{\boldsymbol{X}}$. **(i)** Corrected count data $\widetilde{\boldsymbol{X}}$ grouped by batches. **(j)** Scatter plot of the estimated versus the true cell-specific size factors.

adjusted count data no longer suffer from batch effects and dropout events, and they even do not need further cell-specific normalization (Fig. **2i**). Therefore, they can be treated as if measured in a single batch for downstream analysis.

BUSseq outperforms existing methods in simulation study

We benchmarked BUSseq with the state-of-the-art methods for batch effects correction for scRNA-seq data—LIGER [17], MNN [13], Scanorama [14], scVI [24], Seurat [16] and ZINB-WaVE [23]. The adjusted Rand index (ARI) measures the consistency between two clustering results and is between zero and one, a higher value indicating better consistency

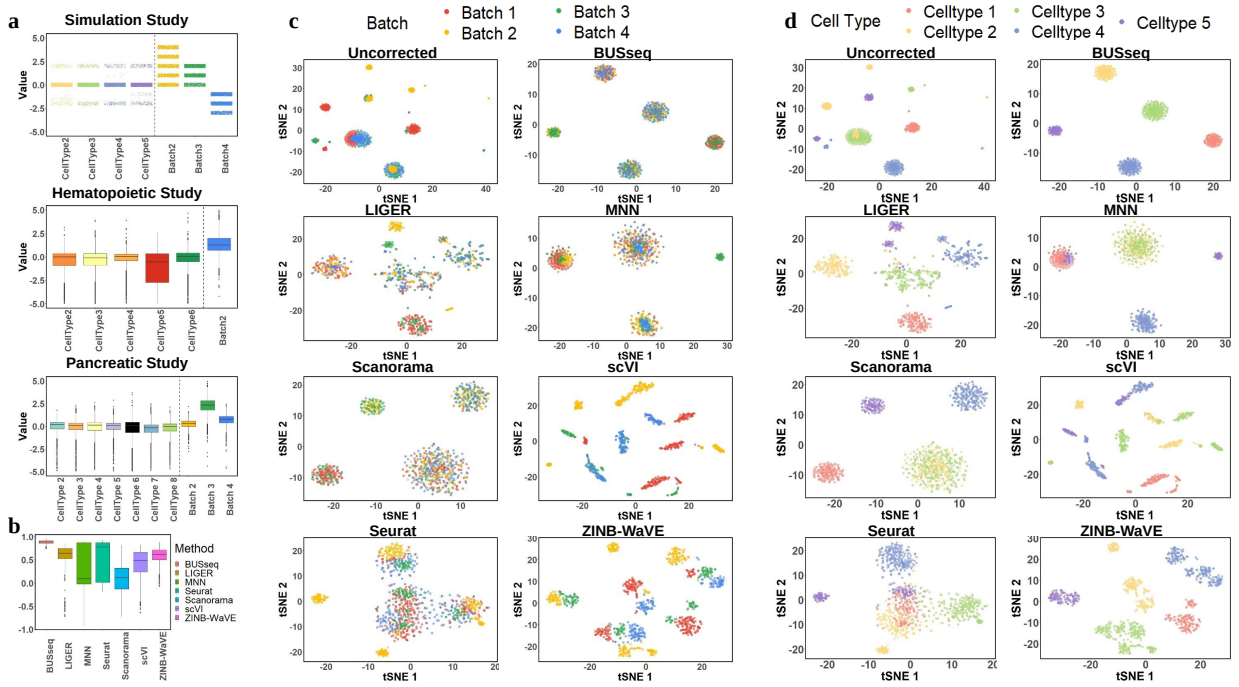

Figure 3: Comparison of batch effects correction methods in the simulation study. **(a)** Comparison of the magnitude of cell type effects and batch effects in the simulation study and the two real applications. The subpanel for the simulation study jitters around the assumed values for $\boldsymbol{\beta}$ and $\boldsymbol{\nu}$. The boxplots show the distributions of the estimated cell type effects $\widehat{\boldsymbol{\beta}}$ and batch effects $\widehat{\boldsymbol{\nu}}$ by BUSseq in the two real studies. The magnitude of the batch effects and cell type effects in the simulation study were chosen to mimic the real data scenarios. **(b)** The boxplots of silhouette coefficients for all compared methods. **(c)** T-distributed Stochastic Neighbor Embedding (t-SNE) plots colored by batch for each compared method. **(d)** t-SNE plots colored by true cell type labels for each compared method.

(Supplementary Notes). The ARI between the inferred cell types $\widehat{W}_{bi}$s by BUSseq and the true underlying cell types W_{bi}s is one. Thus, BUSseq can perfectly recover the true cell type of each cell. In comparison, we apply each of the compared methods to the dataset and then perform their own clustering approaches (Supplementary Notes). The ARI is able to compare the consistency of two clustering results even if the numbers of clusters differ, therefore, we choose the number of cell types by the default approach of each method rather than set it to a common number. The resulting ARIs are 0.837 for LIGER, 0.654 for MNN, 0.521 for Scanorama, 0.480 for scVI, 0.632 for Seurat and 0.571 for ZINB-WaVE. Moreover, the t-SNE plots (**Fig. 3c and d**) show that only BUSseq can perfectly cluster the cells by cell types rather than batches. We also calculated the silhouette score for each cell for each compared method (Supplementary Notes). A high silhouette score indicates that the cell is well matched to its own cluster and separated from neighboring clusters. **Fig. 3b** shows that BUSseq gives the best segregated clusters.

BUSseq outperforms existing methods on hematopoietic data

We re-analyzed the two hematopoietic datasets previously studied by Haghverdi et al.[13], one profiled by the SMART-seq2 protocol for a population of hematopoietic stem and progenitor cells (HSPC) from 12-week-old female mice [34] and another assayed by the massively parallel single-cell RNA-sequencing (MARS-seq) protocol for myeloid progenitors from 6- to 8-week-old female mice [35]. Although the two datasets were generated in two different laboratories (**Fig. 4a**), both datasets have cell-type label for each cell that is annotated according to the expression levels of marker genes [13, 35] from fluorescence-activated cell sorting (FACS) (Methods).

In order to compare BUSseq with existing methods, we compute the ARI between the clustering of each method and the FACS labels. The resulting ARIs are 0.582 for BUSseq, 0.307 for LIGER, 0.575 for MNN, 0.518 for Scanorama, 0.197 for scVI, 0.266 for Seurat and 0.348 for ZINB-WaVE. BUSseq thus outperforms all of the other methods in being consistent with FACS labeling. BUSseq also has silhouette coefficients that are comparable to those of MNN, which are better than those of all the other methods (**Supplementary Fig. 4**). Furthermore, t-SNE plots confirm that BUSseq performs the best in segregating cells into different cell types (**Fig. 4b**).

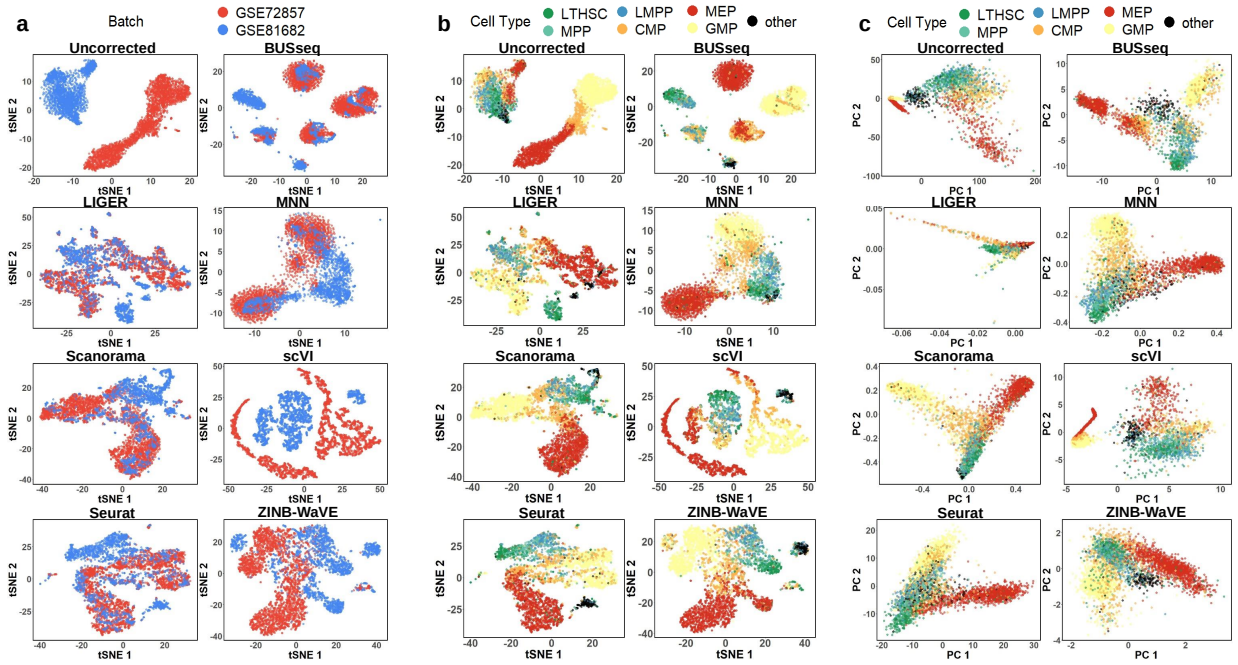

Figure 4: t-SNE and Principal Component Analysis (PCA) plots for the hematopoietic data. **(a)** t-SNE plots colored by batch. **(b)** t-SNE plots colored by FACS cell type labels. (c) PCA plots colored by FACS cell type labels.

Specifically, BUSseq learns 6 cell types from the dataset. According to the FACS labels (Methods), Cluster 2, Cluster 5, and Cluster 6 correspond to the common myeloid progenitors (CMP), megakaryocyte-erythrocyte progenitors (MEP) and granulocyte-monocyte progenitors (GMP), respectively (**Fig. 4c** and **Fig. 5a-c**). Cluster 1 is composed of long-term hematopoietic stem and progenitor cells (LTHSC) and multi-potent progenitors (MPP). These are cells from the early stage of differentiation. Cluster 4 consists of a mixture of MEP and CMP, while Cluster 3 is dominated by cells labeled as "other". Comparison between the subpanel for BUSseq in **Fig. 4c** and **Fig. 5b** indicates that Cluster 4 are cells from an intermediate cell type between CMP and MEP. In particular, according to **Fig. 5e**, the marker genes *Apoe* and *Gata2* are highly expressed in Cluster 4 but not in CMP (Cluster 2) and MEP (Cluster 6), and the marker gene *Ctse* is expressed in MEP (Cluster 6) but not in Cluster 4 and CMP (Cluster 2). Therefore, cells in Cluster 4 do form a unique group with distinct expression patterns. This intermediate cell stage between CMP and GMP is missed by all of the other methods considered. Moreover, we find that well known B-cell lineage genes [36], *Ebf1*, *Vpreb1*, *Vpreb3*, and *Igll1*, are highly expressed in Cluster 3, but not in the other clusters (**Fig. 5c and e**). To identify Cluster 3, which is dominated by cells labeled as

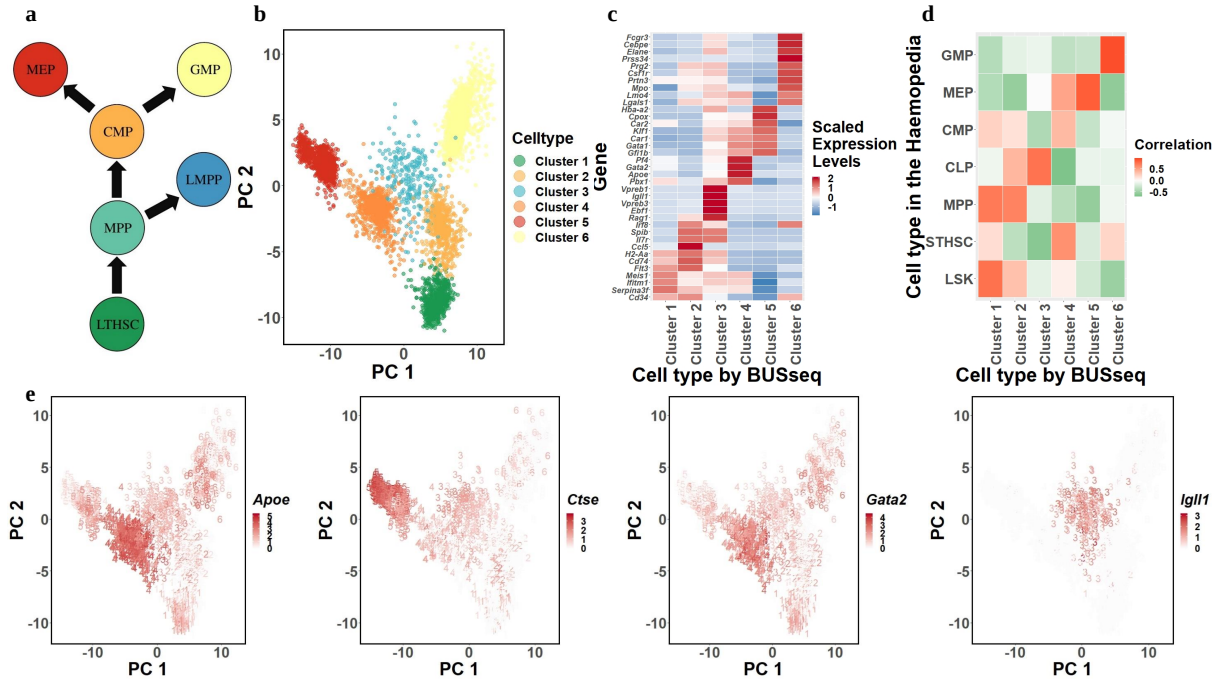

Figure 5: BUSseq preserves the hematopoietic stem and progenitor cells (HSPC) differentiation trajectories. **(a)** The diagram of HSPC differentiation trajectories. **(b)** The PCA plot of the corrected count matrix from BUSseq colored according to the estimated cell types by BUSseq. **(c)** The heatmap of scaled expression levels of key genes for HSPC. **(d)** The heatmap of correlation between gene expression profiles of each cell type inferred by BUSseq and those in the Haemopedia RNA-seq datasets. **(e)** The expression levels of four marker genes, *Apoe*, *Gata2*, *Ctse* and *Igll1*, shown in the PCA plots of corrected count data by BUSseq, respectively. The digit labels denote the corresponding clusters identified by BUSseq.

"other" by Nestorowa et al. [34], we map the mean expression profile of each cluster learned by BUSseq to the Haemopedia RNA-seq dataset [37]. It turns out that Cluster 3 aligns well to common lymphoid progenitors (CLP) that give rise to T-lineage cells, B-lineage cells and natural killer cells (**Fig. 5d**). Therefore, Cluster 3 represents cells that differentiate from lymphoid-primed multipotent progenitors (LMPP) cells [35]. Once again, all the other methods fail to identify these cells as a separate group.

Thus, although BUSseq does not assume any temporal ordering between cell types, it is able to preserve the differentiation trajectories (**Fig. 5a and b**); although BUSseq assumes that each cell belongs to one cell type rather than conducts semisoft clustering [38], it is capable of capturing the subtle changes across cell types and within a cell type due to continuous processes such as development and differentiation.

We further inspect the functions of the intrinsic genes that distinguish different cell types.

BUSseq detects 1419 intrinsic genes at the Bayesian FDR cutoff of 0.05 (Methods). The gene set enrichment analysis [39] shows that 51 KEGG pathways [40] are enriched among the intrinsic genes (p-values < 0.05) (Supplementary Notes). The highest ranked pathway is the Hematopoietic Cell Lineage Pathway, which corresponds to the exact biological process studied in the two datasets. Among the remaining 50 pathways, thirteen are related to the immune system, and another nine are associated with cell growth and differentiation (**Supplementary Table 1**). Therefore, the pathway analysis demonstrates that BUSseq is able to capture the underlying true biological variability, even if the batch effects are severe, as shown in **Fig. 3a** and **Fig. 4a**.

BUSseq outperforms existing method on pancreas data

We further studied the four scRNA-seq datasets of human pancreas cells analyzed in Haghverdi et al. [13], two profiled by CEL-seq2 protocol [41, 42] and two assayed by SMART-seq2 protocol [42, 43]. These cells were isolated from deceased organ donors with and without type 2 diabetes. We obtained 7,095 cells after quality control (Supplementary Notes) and treated each dataset as a batch following Haghverdi et al. [13].

For the two datasets profiled by the SMART-seq2 protocol, Segerstolpe et al. [43] and Lawlor et al. [42] provide cell-type labels; for the other two datasets assayed by the CEL-seq2 protocol, Haghverdi et al. [13] provide the cell-type labels based on the marker genes in the original publications [41, 42]. We can thus compare the clustering results from each batch effects correction method with the labeled cell types (**Fig. 6a and b**).

The pancreas is highly heterogeneous and consists of two major categories of cells: islet cells and non-islet cells. Islet cells include alpha, beta, gamma, and delta cells, while non-islet cells include acinar and ductal cells. BUSseq identifies a total of eight cell types: five for islet cells, two for non-islet cells and one for the labeled "other" cells. Specifically, the five islet cell types identified by BUSseq correspond to three groups of alpha cells, a group of beta cells, and a group of delta and gamma cells. The two non-islet cell types identified by BUSseq correspond exactly to the acinar and ductal cells. Compared to all of the other methods, BUSseq gives the best separation between islet and non-islet cells, as well as the best segregation within islet cells. In particular, the median silhouette coefficient by BUSseq is higher than that of any other method (**Fig. 6c**).

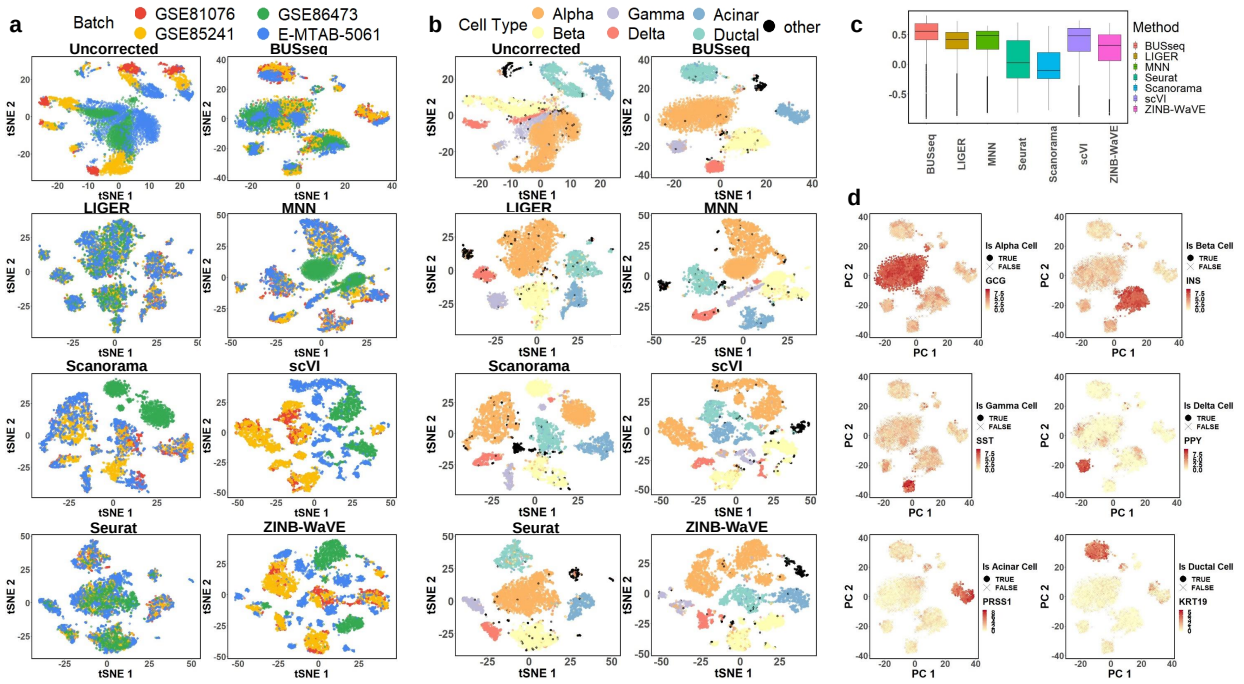

Figure 6: t-SNE plots for the pancreas data. **(a)** t-SNE plots colored by batch. **(b)** t-SNE plot colored by FACS cell type labels. **(c)** The boxplot of silhouette coefficients for all of the compared methods. **(d)** The expression levels of six marker genes, *GCG* for alpha cells, *INS* for beta cells, *SST* for gamma cells, *PPY* for delta cells, *PRSS1* for acinar cells, and *KRT19* for ductal cells, shown in the t-SNE plot of the corrected count data of BUSseq, respectively.

The ARIs of all methods are 0.608 for BUSseq, 0.542 for LIGER, 0.279 for MNN, 0.527 for Scanorama, 0.282 for scVI, 0.287 for Seurat and 0.380 for ZINB-WaVE. Thus, BUSseq outperforms all of the other methods in being consistent with the cell-type labels according to marker genes. In **Fig. 6d**, the locally high expression levels of marker genes for each cell type show that BUSseq correctly clusters cells according to their biological cell types.

BUSseq identifies 426 intrinsic genes at the Bayesian FDR cutoff of 0.05 (Methods). We conducted the gene set enrichment analysis [39] with the KEGG pathways [40] (Supplementary Notes). There are 14 enriched pathways (p-values < 0.05). Among them, three are diabetes pathways; two are pancreatic and insulin secretion pathways; and another two pathways are related to metabolism (**Supplementary Table 2**). Recall that the four datasets assayed pancreas cells from type 2 diabetes and healthy individuals, therefore, the pathway analysis once again confirms that BUSseq provides biologically and clinically valid cell typing.

Discussion

For the completely randomized experimental design, it seems that "everyone is talking, but no one is listening." Due to time and budget constraints, it is always difficult to implement a completely randomized design in practice. Consequently, researchers often pretend to be blind to the issue when carrying out their scRNA-seq experiments. In this paper, we mathematically prove and empirically show that under the more realistic reference panel and chain-type designs, batch effects can also be adjusted for scRNA-seq experiments. We hope that our results will alarm researchers of confounded experimental designs and encourage them to implement valid designs for scRNA-seq experiments in real applications.

BUSseq provides one-stop services. In contrast, most existing methods are multi-stage approaches—clustering can only be performed after the batch effects have been corrected and the differential expressed genes can only be called after the cells have been clustered. The major issue with multi-stage methods is that uncertainties in the previous stages are often ignored. For instance, when cells have been first clustered into different cell types and then differential gene expression identification is conducted, the clustering results are taken as if they were the underlying truth. As the clustering results may be prone to errors in practice, this can lead to false positives and false negatives. In contrast, BUSseq simultaneously corrects batch effects, clusters cell types, imputes missing data, and identifies intrinsic genes that differentiate cell types. BUSseq thus accounts for all uncertainties and fully exploits the information embedded in the data. As a result, BUSseq is able to capture subtler changes between cell types, such as the cluster corresponding to LMPP lineage that is missed by all the state-of-the-art methods.

BUSseq employs MCMC for statistical inference. As MCMC algorithms not only provide point estimates but also explore the entire posterior distributions and hence allow the users to quantify the uncertainty of estimates, they are famous for heavy computation load. However, fortunately, the computational complexity of BUSseq is $O(\sum_{b=1}^{B} n_b GK)$, which is both linear in the number of genes G and in the total number of cells $\sum_{b=1}^{B} n_b$. Moreover, most steps of the MCMC algorithm for BUSseq are parallelizable. We implement a parallel multi-core-CPU version and a parallel GPU version of the algorithm, respectively. Running the GPU version of the algorithm with a single core of an Intel Xeon Gold 6132 Processor and one NVIDIA

Tesla P100 GPU took 0.35, 1.15, 1.5 hours for the simulation, the hematopoietic and the human pancreas data, respectively (**Supplementary Table 3**). Compared to the time for preparing samples and conducting the scRNA-seq experiments, the computation time of BUSseq is affordable and worthwhile for the accuracy.

Practical and valid experimental designs are urgently required for scRNA-seq experiments. We envision that the flexible reference panel and the chain-type designs will be widely adopted in scRNA-seq experiments and BUSseq will greatly facilitate the analysis of scRNA-seq data.

Acknowledgments

This work was supported by the Hong Kong Ph.D. Fellowship PF15-17417 and the General Research Funds 14306417 and 14305319 from the Hong Kong Research Grants Council of the Hong Kong Special Administrative Region of the People's Republic of China and Direct Grants from the Research Committee of the Chinese University of Hong Kong. We acknowledge Dr. Xiangyu Luo for helpful comments on an early version of our paper.

Author Contributions

FD.S developed the method and the proof, implemented the algorithm, prepared the software package, analyzed the data, and wrote the paper. GM.C. implemented the algorithm and analyzed the data. YY.W. conceived and supervised the study, developed the method and the proof, and wrote the paper.

Competing Interests

The authors declare no competing financial interests.

References

[1] Bacher, R. & Kendziorski, C. Design and computational analysis of single-cell RNA-sequencing experiments. *Genome Biology* **17**, 63 (2016).

[2] Irizarry, R. A. *et al.* Multiple-laboratory comparison of microarray platforms. *Nature Methods* **2**, 345–350 (2005).

[3] Leek, J. T. *et al.* Tackling the widespread and critical impact of batch effects in high-throughput data. *Nature Reviews Genetics* **11**, 733–739 (2010).

[4] Taub, M. A., Corrada Bravo, H. & Irizarry, R. A. Overcoming bias and systematic errors in next generation sequencing data. *Genome Medicine* **2**, 87 (2010).

[5] Hicks, S. C., Townes, F. W., Teng, M. & Irizarry, R. A. Missing data and technical variability in single-cell RNA-sequencing experiments. *Biostatistics* **19**, 562–578 (2018).

[6] Kharchenko, P. V., Silberstein, L. & Scadden, D. T. Bayesian approach to single-cell differential expression analysis. *Nature Methods* **11**, 740 (2014).

[7] Johnson, W. E., Li, C. & Rabinovic, A. Adjusting batch effects in microarray expression data using empirical Bayes methods. *Biostatistics* **8**, 118–127 (2007).

[8] Leek, J. T. & Storey, J. D. Capturing heterogeneity in gene expression studies by surrogate variable analysis. *PLoS Genetics* **3**, e161 (2007).

[9] Leek, J. T. Svaseq: removing batch effects and other unwanted noise from sequencing data. *Nucleic Acids Research* **42**, e161–e161 (2014).

[10] Risso, D., Ngai, J., Speed, T. P. & Dudoit, S. Normalization of RNA-seq data using factor analysis of control genes or samples. *Nature Biotechnology* **32**, 896 (2014).

[11] Jacob, L., Gagnon-Bartsch, J. A. & Speed, T. P. Correcting gene expression data when neither the unwanted variation nor the factor of interest are observed. *Biostatistics* **17**, 16–28 (2015).

[12] Huo, Z., Ding, Y., Liu, S., Oesterreich, S. & Tseng, G. Meta-analytic framework for sparse k-means to identify disease subtypes in multiple transcriptomic studies. *Journal of the American Statistical Association* **111**, 27–42 (2016).

[13] Haghverdi, L., Lun, A. T., Morgan, M. D. & Marioni, J. C. Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. *Nature Biotechnology* **36**, 421 (2018).

[14] Hie, B., Bryson, B. & Berger, B. Efficient integration of heterogeneous single-cell transcriptomes using Scanorama. *Nature Biotechnology* **37**, 685 (2019).

[15] Butler, A., Hoffman, P., Smibert, P., Papalexi, E. & Satija, R. Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nature Biotechnology* **36**, 411 (2018).

[16] Stuart, T. *et al.* Comprehensive integration of single-cell data. *Cell* **177**, 1888–1902.e21 (2019).

[17] Welch, J. D. *et al.* Single-cell multi-omic integration compares and contrasts features of

brain cell identity. *Cell* **177**, 1873–1887.e17 (2019).

[18] Lin, Y. *et al.* scmerge leverages factor analysis, stable expression, and pseudoreplication to merge multiple single-cell rna-seq datasets. *Proceedings of the National Academy of Sciences* **116**, 9775–9784 (2019).

[19] Luo, X. & Wei, Y. Batch effects correction with unknown subtypes. *Journal of the American Statistical Association* **114**, 581–594 (2019).

[20] Vallejos, C. A., Marioni, J. C. & Richardson, S. BASiCS: Bayesian analysis of single-cell sequencing data. *PLoS Computational Biology* **11**, e1004333 (2015).

[21] Wang, J. *et al.* Gene expression distribution deconvolution in single-cell RNA sequencing. *Proceedings of the National Academy of Sciences of the United States of America* **115**, E6437–E6446 (2018).

[22] Pierson, E. & Yau, C. Zifa: Dimensionality reduction for zero-inflated single-cell gene expression analysis. *Genome Biology* **16**, 241 (2015).

[23] Risso, D., Perraudeau, F., Gribkova, S., Dudoit, S. & Vert, J.-P. A general and flexible method for signal extraction from single-cell RNA-seq data. *Nature Communications* **9**, 284 (2018).

[24] Lopez, R., Regier, J., Cole, M. B., Jordan, M. I. & Yosef, N. Deep generative modeling for single-cell transcriptomics. *Nature Methods* **15**, 1053 (2018).

[25] Wang, J. *et al.* Data denoising with transfer learning in single-cell transcriptomics. *Nature Methods* **16**, 875–878 (2019).

[26] Baran-Gale, J., Chandra, T. & Kirschner, K. Experimental design for single-cell RNA sequencing. *Briefings in Functional Genomics* **17** (2017).

[27] Dal, M. A. & Di, C. B. How to design a single-cell RNA-sequencing experiment: pitfalls, challenges and perspectives. *Briefings in Bioinformatics* (2018).

[28] Robert, C. & Casella, G. *Monte Carlo Statistical Methods* (Springer Science & Business Media, 2013).

[29] Schwarz, G. *et al.* Estimating the dimension of a model. *The Annals of Statistics* **6**, 461–464 (1978).

[30] Casella, G. & Berger, R. L. *Statistical Inference*, vol. 2 (Duxbury Pacific Grove, CA, 2002).

[31] Miao, W., Ding, P. & Geng, Z. Identifiability of normal and normal mixture models with nonignorable missing data. *Journal of the American Statistical Association* **111**, 1673–1683 (2016).

[32] Newton, M. A., Noueiry, A., Sarkar, D. & Ahlquist, P. Detecting differential gene expression with a semiparametric hierarchical mixture method. *Biostatistics* **5**, 155–176

(2004).

[33] Peterson, C., Stingo, F. C. & Vannucci, M. Bayesian inference of multiple Gaussian graphical models. *Journal of the American Statistical Association* **110**, 159–174 (2015).

[34] Nestorowa, S. *et al.* A single cell resolution map of mouse haematopoietic stem and progenitor cell differentiation. *Blood* **128**, e20–31 (2016).

[35] Paul, F. *et al.* Transcriptional heterogeneity and lineage commitment in myeloid progenitors. *Cell* **163**, 1663–1677 (2015).

[36] Herman, J. S., Grün, D. *et al.* FateID infers cell fate bias in multipotent progenitors from single-cell RNA-seq data. *Nature Methods* **15**, 379 (2018).

[37] Choi, J. *et al.* Haemopedia RNA-seq: a database of gene expression during haematopoiesis in mice and humans. *Nucleic Acids Research* **47**, D780–D785 (2018).

[38] Zhu, L., Lei, J., Klei, L., Devlin, B. & Roeder, K. Semisoft clustering of single-cell data. *Proceedings of the National Academy of Sciences* **116**, 466–471 (2019).

[39] Huang, D. W., Sherman, B. T. & Lempicki, R. A. Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. *Nature Protocols* **4**, 44 (2009).

[40] Kanehisa, M. & Goto, S. KEGG: kyoto encyclopedia of genes and genomes. *Nucleic Acids Research* **28**, 27–30 (2000).

[41] Grün, D. *et al.* De novo prediction of stem cell identity using single-cell transcriptome data. *Cell Stem Cell* **19**, 266–277 (2016).

[42] Lawlor, N. *et al.* Single-cell transcriptomes identify human islet cell signatures and reveal cell-type-specific expression changes in type 2 diabetes. *Genome Research* **27**, 208–222 (2017).

[43] Segerstolpe, Å. *et al.* Single-cell transcriptome profiling of human pancreatic islets in health and type 2 diabetes. *Cell Metabolism* **24**, 593–607 (2016).

[44] George, E. I. & McCulloch, R. E. Variable selection via Gibbs sampling. *Journal of the American Statistical Association* **88**, 881–889 (1993).

[45] Robinson, M. D. & Oshlack, A. A scaling normalization method for differential expression analysis of RNA-seq data. *Genome Biology* **11**, R25 (2010).

[46] Muraro, M. J. *et al.* A single-cell transcriptome atlas of the human pancreas. *Cell Systems* **3**, 385–394 (2016).

Methods

BUSseq model

The hierarchical model of BUSseq can be summarized as:

$$Pr(W_{bi} = k) = \pi_{bk}, \sum_{k=1}^{K} \pi_{bk} = 1;$$

$$X_{big}|W_{bi} = k \sim NB(\mu_{big}, \phi_{bg}), log(\mu_{big}) = \alpha_g + \beta_{gk} + \nu_{bg} + \delta_{bi};$$

$$Z_{big}|X_{big} = x_{big} \sim Bernoulli(p_{big}), log(\frac{p_{big}}{1 - p_{big}}) = \gamma_{b0} + \gamma_{b1} x_{big};$$

$$Y_{big} = X_{big}|Z_{big} = 0, Y_{big} = 0|Z_{big} = 1.$$

Collectively, $\boldsymbol{Y} = \{Y_{big}\}_{b=1,\cdots,B;i=1,\cdots,n_b}^{g=1,\cdots,G}$ are the observed data; the underlying expression levels $\boldsymbol{X} = \{X_{big}\}_{b=1,\cdots,B;i=1,\cdots,n_b}^{g=1,\cdots,G}$, the dropout indicators $\boldsymbol{Z} = \{Z_{big}\}_{b=1,\cdots,B;i=1,\cdots,n_b}^{g=1,\cdots,G}$ and the cell type indicators $\boldsymbol{W} = \{W_{bi}\}_{b=1,\cdots,B;i=1,\cdots,n_b}$ are all missing data; the log-scale baseline gene expression levels $\boldsymbol{\alpha} = \{\alpha_g\}_{g=1,\cdots,G}$, the cell type effects $\boldsymbol{\beta} = \{\beta_{gk}\}_{k=2,\cdots,K}^{g=1,\cdots,G}$, the location batch effects $\boldsymbol{\nu} = \{\nu_{bg}\}_{b=2,\cdots,B}^{g=1,\cdots,G}$, the overdispersion parameters $\boldsymbol{\phi} = \{\phi_{bg}\}_{b=1,\cdots,B}^{g=1,\cdots,G}$, the cell-specific size factors $\boldsymbol{\Delta} = \{\delta_{bi}\}_{b=1,\cdots,B}^{i=2,\cdots,n_b}$, the dropout parameters $\boldsymbol{\Gamma} = \{\gamma_{b0}, \gamma_{b1}\}^{b=1,\cdots,B}$ and the cell compositions $\boldsymbol{\pi} = \{\pi_{bk}\}_{b=1,\cdots,B}^{k=1,\cdots,K}$ are the parameters. Without loss of generality, for model identifiability, we assume that the first batch is the reference batch measured without batch effects with $\nu_{1g} = 0$ for every gene and the first cell type is the baseline cell type with $\beta_{g1} = 0$ for every gene. Similarly, we take the cell-specific size factor $\delta_{b1} = 0$ for the first cell of each batch. We gather all the parameters as $\boldsymbol{\Theta} = \{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\nu}, \boldsymbol{\phi}, \boldsymbol{\Delta}, \boldsymbol{\Gamma}, \boldsymbol{\pi}\}$.

Experimental designs

By creating a set of functions similar to the probability generating function, we prove that BUSseq is identifiable, in other words, if two sets of parameters are different, then their probability distribution functions for the observed data are different, for not only the "complete setting" but also the "reference panel" and the "chain-type" designs (see the proofs in the Supplementary Notes).

Theorem 1. *(The Complete Setting)*
If $\pi_{bk} > 0$ for every batch b and cell type k, given that (I) $\gamma_{b1} < 0$ for every b, (II) for any two cell types k_1 and k_2, there exist at least two differentially expressed genes g_1 and g_2— $\beta_{g_1k_1} \neq \beta_{g_1k_2}$ and $\beta_{g_2k_1} \neq \beta_{g_2k_2}$, and (III) for any two distinct cell-type pairs $(k_1, k_2) \neq (k_3, k_4)$, their differences in cell-type effects are not the same $\boldsymbol{\beta}_{k_1} - \boldsymbol{\beta}_{k_2} \neq \boldsymbol{\beta}_{k_3} - \boldsymbol{\beta}_{k_4}$, then BUSseq is identifiable (up to label switching) in the sense that $L_o(\boldsymbol{\Theta}|\boldsymbol{y}) = L_o(\boldsymbol{\Theta}^*|\boldsymbol{y})$ for any $\boldsymbol{y}$ implies

that $\pi_{bk} = \pi^*_{b\rho(k)}, (\gamma_{b0}, \gamma_{b1}) = (\gamma^*_{b0}, \gamma^*_{b1}), \alpha_g + \beta_{gk} = \alpha^*_g + \beta^*_{g\rho(k)}, \nu_{gb} = \nu^*_{gb}, \delta_{bi} = \delta^*_{bi}$ *and* $\phi_{bg} = \phi^*_{bg}$ *for every gene* g *and batch* b*, where* ρ *is a permutation of* $\{1, 2, \cdots, K\}$.

In the following, we denote the cell types that are present in batch b as C_b and count the number of cell types existing in batch b as $K_b = |C_b|$.

Theorem 2. *(The Reference Panel Design)*
If there are a total of K *cell types* $\cup_{b=1}^{B} C_b = \{1, 2, \cdots, K\}$, $K_b \geq 2$ *for every batch* b*, and there exists a batch* $\tilde{b}$ *such that it contains all of the cell types* $C_{\tilde{b}} = \{1, 2, \cdots, K\}$*, then given that conditions (I)-(III) hold, BUSseq is identifiable (up to label switching).*

Theorem 3. *(The Chain-type Design)*
If there are a total of K *cell types* $\cup_{b=1}^{B} C_b = \{1, 2, \cdots, K\}$ *and every two consecutive batches share at least two cell types* $|C_b \cap C_{b-1}| \geq 2$ *for all* $b \geq 2$*, then given that conditions (I)-(III) hold, BUSseq is identifiable (up to label switching).*

Therefore, even for the "reference panel" and "chain-type" designs that do not assay all cell types in each batch, batch effects can be removed; cell types can be clustered; and missing data due to dropout events can be imputed. Both the reference panel design and the chain-type design belong to the more general connected design.

Theorem 4. *(The Connected Design)*
We define a batch graph $G = (V, E)$*. Each node* $b \in V$ *represents a batch. There is an edge* $e \in E$ *between two nodes* b_1 *and* b_2 *if and only if batches* b_1 *and* b_2 *share at least two cell types. If the batch graph is connected and conditions (I)-(III) hold, then BUSseq is identifiable (up to label switching).*

Statistical inference

We conduct the statistical inference under the Bayesian framework. We assign independent priors to each component of $\boldsymbol{\Theta}$ as follows: $\boldsymbol{\pi}_b = (\pi_{b1}, \cdots, \pi_{bK}) \sim Dirichlet(\xi, \cdots, \xi), 1 \leq b \leq B; \gamma_{b0} \sim N(0, \sigma^2_{z0}), 1 \leq b \leq B; -\gamma_{b1} \sim Gamma(a_\gamma, b_\gamma), 1 \leq b \leq B; \alpha_g \sim N(m_a, \sigma^2_a), 1 \leq g \leq G; \nu_{bg} \sim N(m_c, \sigma^2_c), 2 \leq b \leq B, g = 1, \cdots, G; \delta_{bi} \sim N(m_d, \sigma^2_d), 1 \leq b \leq B, 2 \leq i \leq n_b; \phi_{bg} \sim Gamma(\kappa, \tau), 1 \leq b \leq B, 1 \leq g \leq G.$

We are interested in detecting genes that differentiate cell types. Therefore, we impose a spike-and-slab prior [44] using a normal mixture to the cell-type effect β_{gk}. The spike component concentrates on zero with a small variance $\tau^2_{\beta 0}$, whereas the slab component tends to deviate from zero, thus having a larger variance $\tau^2_{\beta 1}$. We introduce another latent variable L_{gk} to indicate which component β_{gk} comes from. $L_{gk} = 0$ if gene g is not differentially expressed between cell type k and cell type one, and $L_{gk} = 1$, otherwise. We further define

$D_g = \sum_{k=2}^{K} L_{gk}$. If $D_g > 0$, then the expression level of gene g does not stay the same across cell types. Following Huo et al. [12], we call such genes intrinsic genes, which differentiate cell types. To control for multiple hypothesis testing, we let $L_{gk} \sim Bernoulli(p)$ and assign a conjugate prior $Beta(a_p, b_p)$ to p. We set $\tau_{\beta 1}$ to a large number and let $\tau_{\beta 0}$ follow an inverse-gamma prior $Inv-Gamma(a_\tau, b_\tau)$ with a small prior mean.

We develop an MCMC algorithm to sample from the posterior distribution (Supplementary Notes). After the burn-in period, we take the mean of the posterior samples to estimate $\boldsymbol{\gamma}_b, \alpha_g, \beta_{gk}, \nu_{bg}, \delta_{bi}$ and ϕ_{bg} and use the mode of posterior samples of W_{bi} to infer the cell type for each cell.

When inferring the differential expression indicator L_{gk}, we control the Bayesian false discovery rate (FDR) [32] defined as

$$FDR(\kappa) = \frac{\sum_{g=1}^{G}\sum_{k=2}^{K} \xi_{gk} I(\xi_{gk} \le \kappa)}{\sum_{g=1}^{G}\sum_{k=2}^{K} I(\xi_{gk} \le \kappa)},$$

where $\xi_{gk} = Pr(L_{gk} = 0|\boldsymbol{y})$ is the posterior marginal probability that gene g is not differentially expressed between cell type k and cell type one, which can be estimated by the T posterior samples $L_{gk}^{(t)}$s collected after the burn-in period as $\frac{1}{T}\sum_{t=1}^{T}(1 - L_{gk}^{(t)})$. Given a control level α such as 0.1, we search for the largest $\kappa_0 \le 0.5$ such that the estimated $\widehat{FDR}(\kappa)$ based on $\widehat{\xi}_{gk}$s is smaller than α and declare $\widehat{L}_{gk} = 1$ if $\widehat{\xi}_{gk} \le \kappa_0$. The upper bound 0.5 for κ_0 prevents us from calling differentially expressed genes with small posterior probability $Pr(L_{gk} = 1|\boldsymbol{y})$. Consequently, we identify the genes with $\widehat{D}_g = \sum_{k=2}^{K} \widehat{L}_{gk} > 0$ as the intrinsic genes. We set $\alpha = 0.05$ in both the simulation study and the real applications.

BUSseq allows the user to input the total number of cell types K according to prior knowledge. When K is unknown, BUSseq selects the number of cell types $\widehat{K}$ such that it achieves the minimum BIC (Supplementary Notes).

Batch-effects-corrected values

To facilitate further downstream analysis, we also provide a version of count data $\widetilde{\boldsymbol{X}} = \{\widetilde{X}_{big}\}_{b=1,\cdots,B;i=1,\cdots,n_b}^{g=1,\cdots,G}$ for which the batch effects are removed and the biological variability is retained. We develop a quantile matching approach based on inverse sampling. Specifically, given the fitted model and the inferred underlying expression level $\widehat{x}_{big}$, we first sample u_{big} from $Unif[F_{NB}(\widehat{x}_{big}-1; exp(\widehat{\alpha}_g+\widehat{\beta}_{g\hat{w}_{bi}}+\widehat{\nu}_{bg}+\widehat{\delta}_{bi}), \widehat{\phi}_{bg}), F_{NB}(\widehat{x}_{big}; exp(\widehat{\alpha}_g+\widehat{\beta}_{g\hat{w}_{bi}}+\widehat{\nu}_{bg}+\widehat{\delta}_{bi}), \widehat{\phi}_{bg})]$ where $Unif[a, b]$ denotes the uniform distribution on the interval $[a, b]$ and $F_{NB}(\cdot; \mu, r)$ denotes the cumulative distribution function of a negative binomial distribution with mean μ and overdispersion parameter r. Next, we calculate the u_{big}^{th} quantile of $NB(exp(\widehat{\alpha}_g + \widehat{\beta}_{g\hat{w}_{bi}}), \widehat{\phi}_{1g})$ as the corrected value $\widetilde{x}_{big}$.

The corrected data $\widetilde{\boldsymbol{X}}$ are not only protected from batch effects but also impute the missing data due to dropout events. Moreover, further cell-specific normalization is not needed. Meanwhile, the biological variability is retained thanks to the quantile transformation and sampling step. Therefore, we can directly perform downstream analysis on $\widetilde{\boldsymbol{X}}$.

Assignment of FACS cell type labels to learned clusters

In the two real data examples, we first identify the cell type of each individual cell according to FACS labeling. Then, for each cluster learned by BUSseq, we calculate the proportion of labeled cell types. If a cell type accounts for more than one-third of the cells in a given cluster, we assign this cell type to the cluster. Although a cluster may be assigned more than one cell type, most identified clusters by BUSseq are dominated by only one cell type.

Mapping clusters to Haemopedia

Haemopedia is a database of gene expression profiles from diverse types of hematopoietic cells [37]. It collected flow sorted cell populations from healthy mice.

To understand Cluster 3 learned by BUSseq for the hematopoietic data, which is dominated by cells classified as "other" according to the FACS labeling, we mapped the cluster means learned by BUSseq to the Haemopedia RNA-seq dataset.

We first applied TMM normalization [45] to all the samples in the Haemopedia RNA-seq dataset. Then, we extracted 7 types of hematopoietic stem and progenitor cells from Haemopedia, including Lin$^-$Sca-1$^+$c-Kit$^+$ (LSK) cells, short-term hematopoietic stem cells (STHSC), MPP, CLP, CMP, MEP and GMP. Each selected cell type had two RNA-seq samples in Haemopedia, so we averaged over the two replicates for each cell type. Further, we added one to the normalized expression levels as a pseudo read count to handle genes with zero read count and log-transformed the data. Finally, we scaled the data across the 7 cell types for each gene. To be comparable, we transformed the cluster mean learned by BUSseq as $m_{gk} = log(1 + exp(\alpha_g + \beta_{gk}))$ for gene g in the cluster k and scaled m_{gk} across all cell types as well. Finally, we calculated the correlation between the cluster means inferred by BUSseq and the reference expression profiles in Haemopedia for 37 marker genes. The 37 marker genes were retrieved from Paul et al. [35] (31 maker genes for HSPC) and Herman et al. [36] (6 maker genes for LMPP).

Software availability

The C++ source code of the parallel multi-core-CPU version of BUSseq is available on GitHub `https://github.com/songfd2018/BUSseq-1.0`, and the CUDA C source code of the GPU

version of BUSseq is available on GitHub `https://github.com/Anguscgm/BUSseq_gpu`. All codes for producing results and figures in this manuscript are also available on GitHub (`https://github.com/songfd2018/BUSseq-1.0_implementation`).

Data availability

The published data sets used in this manuscript are available through the following accession numbers: SMART-seq2 platform hematopoietic data with GEO GSE81682 by Nestorowa et al. [34]; MARS-seq platform hematopoietic data with GEO GSE72857 by Paul et al. [35]; CEL-seq platform pancreas data with GEO GSE81076 by Grün et al. [41]; CEL-seq2 platform pancreas data with GEO GSE85241 by Muraro et al. [46]; SMART-seq2 platform pancreas data with GEO GSE86473 by Lawlor et al. [42]; and SMART-seq2 platform pancreas data with ArrayExpress E-MTAB-5061 by Segerstolpe et al. [43].

The parameter settings for the simulation study and the simulated data are available on GitHub (`https://github.com/songfd2018/BUSseq-1.0_implementation`).