scMerge leverages factor analysis, stable expression, and pseudoreplication to merge multiple single-cell RNA-seq datasets

Yingxin Lin[a], Shila Ghazanfar[a,b,1], Kevin Y. X. Wang[a,1], Johann A. Gagnon-Bartsch[c], Kitty K. Lo[a], Xianbin Su[d,e], Ze-Guang Han[d,e], John T. Ormerod[a], Terence P. Speed[f,g], Pengyi Yang[a,b,2], and Jean Yee Hwa Yang[a,b,2]

[a]School of Mathematics and Statistics, University of Sydney, Sydney, NSW 2006, Australia; [b]Charles Perkins Centre, University of Sydney, Sydney, NSW 2006, Australia; [c]Department of Statistics, University of Michigan, Ann Arbor, MI 48109; [d]Key Laboratory of Systems Biomedicine, Ministry of Education, Shanghai Center for Systems Biomedicine, Shanghai Jiao Tong University, Shanghai 200240, China; [e]Collaborative Innovation Center of Systems Biomedicine, Shanghai Center for Systems Biomedicine, Shanghai Jiao Tong University, Shanghai 200240, China; [f]Bioinformatics Division, Walter and Eliza Hall Institute of Medical Research, Parkville, VIC 3052, Australia; and [g]Department of Mathematics and Statistics, University of Melbourne, Melbourne, VIC 3010, Australia

Edited by Wing Hung Wong, Stanford University, Stanford, CA, and approved April 2, 2019 (received for review November 26, 2018)

Concerted examination of multiple collections of single-cell RNA sequencing (RNA-seq) data promises further biological insights that cannot be uncovered with individual datasets. Here we present scMerge, an algorithm that integrates multiple single-cell RNA-seq datasets using factor analysis of stably expressed genes and pseudoreplicates across datasets. Using a large collection of public datasets, we benchmark scMerge against published methods and demonstrate that it consistently provides improved cell type separation by removing unwanted factors; scMerge can also enhance biological discovery through robust data integration, which we show through the inference of development trajectory in a liver dataset collection.

single-cell RNA-seq data | data integration | factor analysis | normalization | pseudoreplications

Single-cell transcriptome profiling by next generation sequencing (scRNA-seq) has enabled unprecedented resolution in studying cell identity, heterogeneity, and differentiation trajectories in various biological systems (1). Comprehensive characterization of large collections of scRNA-seq datasets can provide a more holistic understanding of the underlying biological processes which may not be achievable from analyzing each dataset independently. However, the integration of multiple scRNA-seq datasets remains a challenge due to prevailing technical effects associated with experiments performed across multiple batches, conditions, and organisms. Here, we present scMerge, an algorithm that corrects for batch effects within an experiment as well as removing dataset-specific effects across collections of datasets, and subsequently enables integrative biological analysis of multiple scRNA-seq datasets.

While normalization methods such as SCnorm (2), scran (3), mnnCorrect (4), and ComBat (5) can be applied for combining multiple scRNA-seq datasets, they are either not specifically designed for adjusting batch effects or are primarily designed in the context of removing batch effects within a single experiment. Alternatively, data integration methods such as Seurat (6) and the fast version of mnnCorrect (fastMNN) (4) generate dimension-reduced datasets where individual genes cannot be examined for downstream analysis such as differential expression (DE)-based marker identification or pseudotime trajectory estimation. While the zero-inflated negative binomial model (ZINB-WaVE) (7) also produces a dimension-reduced dataset, it enables "subtraction" from the full data for downstream analysis. However, it is shown to suffer in scaling to a larger number of cells. Moreover, methods such as ComBat implicitly assume that the batches or datasets to be integrated contain similar proportions of particular cell types. As previously described (4), this assumption can lead to incorrectly normalized data, especially when particular batches or datasets have markedly different proportions of cell types, e.g., as a result of fluorescence-activated cell sorting applied to a set of samples; mnnCorrect addresses this by estimating a set of "mutual nearest neighbors," a mapping of individual cells between batches or datasets, but it can be unstable due to the selection of individual pairs of cells, as opposed to the more robust selection of pairs of cell clusters.

Results

scMerge. To enable effective integration of multiple scRNA-seq datasets, scMerge leverages factor analysis of single-cell stably expressed genes (scSEGs) and pseudoreplicates identified across datasets. The three key components are summarized as follows (Fig. 1*A* and *SI Appendix*, Fig. S1): (*i*)the identification of scSEGs via a Gamma–Gaussian mixture model (8) for use as "negative controls" for estimating unwanted factors; (*ii*) the construction of pseudoreplicates to estimate the effects of unwanted factors;

Significance

Single-cell RNA-sequencing (scRNA-seq) profiling has exploded in recent years and enabled new biological knowledge to be discovered at the single-cell level. Successful and flexible integration of scRNA-Seq datasets from multiple sources promises to be an effective avenue to obtain further biological insights. This study presents a comprehensive approach to integration for scRNA-seq data analysis. It addresses the challenges involved in successful integration of scRNA-seq datasets by using the knowledge of genes that appear not to change across all samples and a robust algorithm to infer pseudoreplicates between datasets. This information is then consolidated into a single-factor model that enables tailored incorporation of prior knowledge. The effectiveness of scMerge is demonstrated by extensive comparison with other approaches.

Author contributions: S.G., K.Y.X.W., P.Y., and J.Y.H.Y. designed research; Y.L., S.G., K.Y.X.W., J.A.G.-B., J.T.O., T.P.S., P.Y., and J.Y.H.Y. performed research; Y.L., K.Y.X.W., J.A.G.-B., and J.T.O. contributed new reagents/analytic tools; Y.L., S.G., K.Y.X.W., J.A.G.-B., X.S., Z.-G.H., T.P.S., P.Y., and J.Y.H.Y. analyzed data; Y.L., S.G., K.Y.X.W., K.K.L., T.P.S., P.Y., and J.Y.H.Y. wrote the paper; and Y.L., K.K.L., and J.Y.H.Y. processed and curated all data.

The authors declare no conflict of interest.

This article is a PNAS Direct Submission.

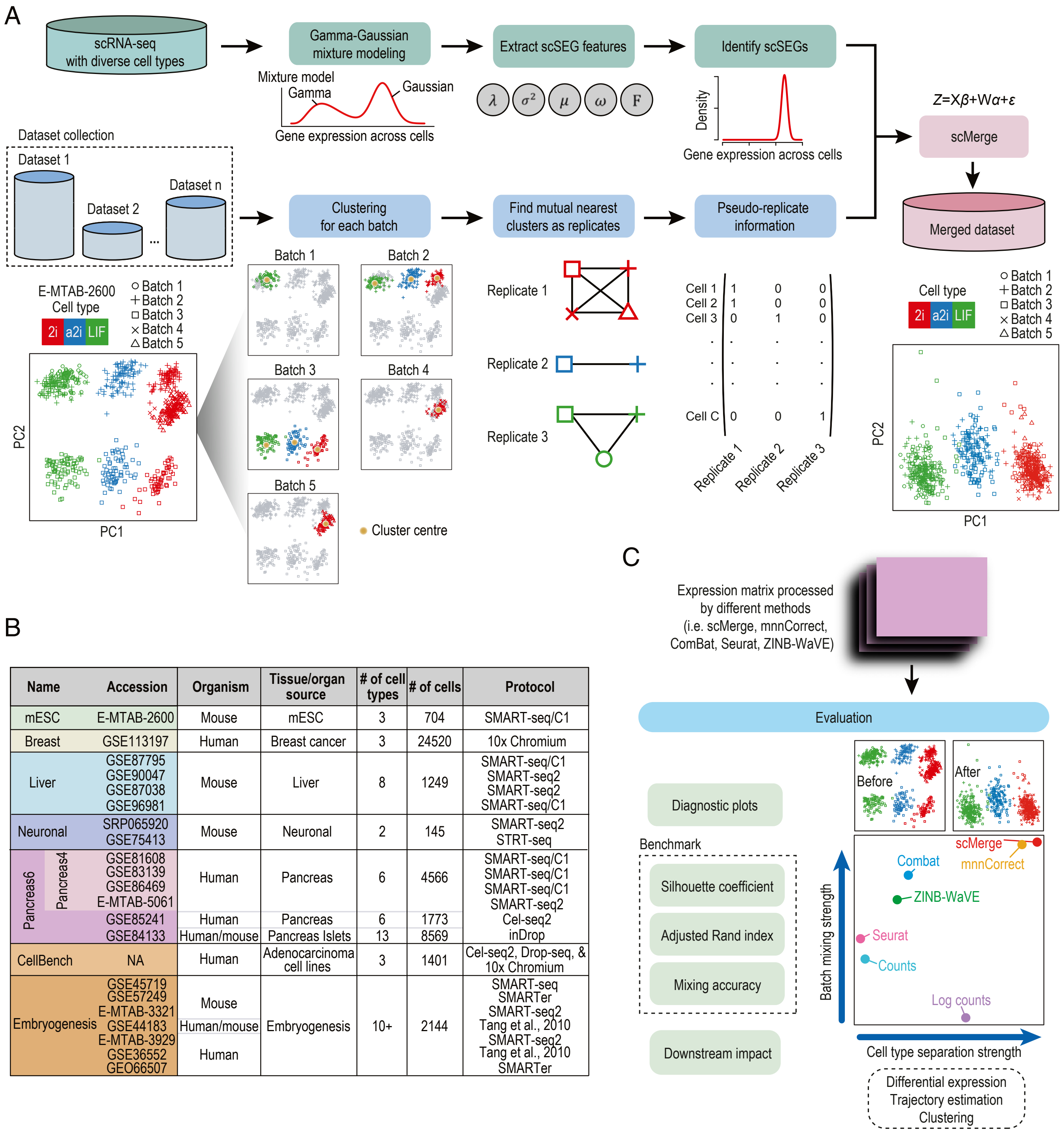
Data deposition: The R package scMerge is available on the Github repository (https://sydneybiox.github.io/scMerge).

[1] S.G. and K.Y.X.W. contributed equally to this work.

[2] To whom correspondence may be addressed. Email: jean.yang@sydney.edu.au or pengyi.yang@sydney.edu.au.

This article contains supporting information online at www.pnas.org/lookup/suppl/doi:10.1073/pnas.1820006116/-/DCSupplemental.

Published online April 26, 2019.



Name	Accession	Organism	Tissue/organ source	# of cell types	# of cells	Protocol
mESC	E-MTAB-2600	Mouse	mESC	3	704	SMART-seq/C1
Breast	GSE113197	Human	Breast cancer	3	24520	10x Chromium
Liver	GSE87795 GSE90047 GSE87038 GSE96981	Mouse	Liver	8	1249	SMART-seq/C1 SMART-seq2 SMART-seq2 SMART-seq/C1
Neuronal	SRP065920 GSE75413	Mouse	Neuronal	2	145	SMART-seq2 STRT-seq
Pancreas6 Pancreas4	GSE81608 GSE83139 GSE86469 E-MTAB-5061	Human	Pancreas	6	4566	SMART-seq/C1 SMART-seq/C1 SMART-seq/C1 SMART-seq2
	GSE85241	Human	Pancreas	6	1773	Cel-seq2
	GSE84133	Human/mouse	Pancreas Islets	13	8569	inDrop
CellBench	NA	Human	Adenocarcinoma cell lines	3	1401	Cel-seq2, Drop-seq, & 10x Chromium
Embryogenesis	GSE45719 GSE57249 E-MTAB-3321	Mouse	Embryogenesis	10+	2144	SMART-seq SMARTer SMART-seq2
	GSE44183	Human/mouse				Tang et al., 2010
	E-MTAB-3929 GSE36552 GEO66507	Human				SMART-seq2 Tang et al., 2010 SMARTer

Fig. 1. (*A*) First, scSEGs are identified using a reference dataset with diverse cell types, to be used as negative control genes. Second, for a given data collection with multiple datasets, clustering is performed per dataset, and MNCs are identified across datasets. Selected cells from these clusters are then identified as pseudoreplicates, to be treated as replicates in the factor analysis step. Factor analysis is performed with the negative control genes and pseudoreplicates, resulting in a single merged dataset. (*B*) Summary of 14 datasets comprising seven data collections used in this study. (*C*) Summary of evaluation strategies for merged datasets using diagnostic plots, indices comparison with known cell type labels, and further downstream impacts.

and (*iii*) the adjustment of datasets for unwanted variation using a fastRUVIII model.

We propose an analytical framework for deriving a list of scSEGs based on an SEG index from scRNA-seq data. This is achieved by ranking genes based on four characteristics extracted from scRNA-seq data which we termed "SEG features." The proposed approach was applied to two large-scale high-resolution scRNA-seq datasets generated from early human and mouse development (9, 10) to identify genes that are stably expressed across a wide range of cell types and developmental stages. The broad coverage of these two datasets, from as early as zygotes to mature blastocysts that represent distinctive tissue precursors including trophectoderm, primitive endoderm, and epiblast (11), provides a suitable starting point for deriving SEGs in human and mouse.

Assuming that, for a given set of batches of datasets, there is at least one cell type in common across different batches, we can consider these cells of the same type in different batches as pseudoreplicates. The identification of pseudoreplicates from multiple sets of scRNA-seq data can be performed using an unsupervised or a semisupervised approach.

To remove the unwanted variation across multiple datasets and batches, we developed and incorporated a fast version of RUVIII (fastRUVIII) (12) in scMerge.

Details of these components are included in *Materials and Methods*. In essence, scMerge takes gene expression matrices from a collection of datasets and a list of negative control genes whose expressions are expected to be relatively constant across these datasets. The final output is a single normalized and batch-corrected gene expression matrix with all input matrices merged and ready for further downstream analysis. Biological knowledge such as cell type information can easily be incorporated into scMerge to further improve performance.

SEGs Are Better Negative Controls for Integration. Using a mixture modeling approach (8), our algorithm defines scSEGs (1,076 for human and 826 for mouse) that are characterized by low variability and wide range of expression (Fig. 2*A*). Expression of scSEGs show minimal association with cell types and developmental stages compared with previously identified housekeeping genes from bulk transcriptome data (bHK) (13, 14) or random subsets of genes (n = 1,076; Fig. 2*B*). We found that, in cases where batch information is unknown and thus pseudoreplicates cannot be identified, using scSEGs as negative control genes results in better integration of data (Fig. 2*C* and *SI Appendix*, Fig. S2), compared with bHK genes. Consistent with this, we found that using scSEGs as negative controls also results in better integration of data (F1 scores; see *Materials and Methods*) than using External RNA Controls Consortium (ERCC) spike-in controls

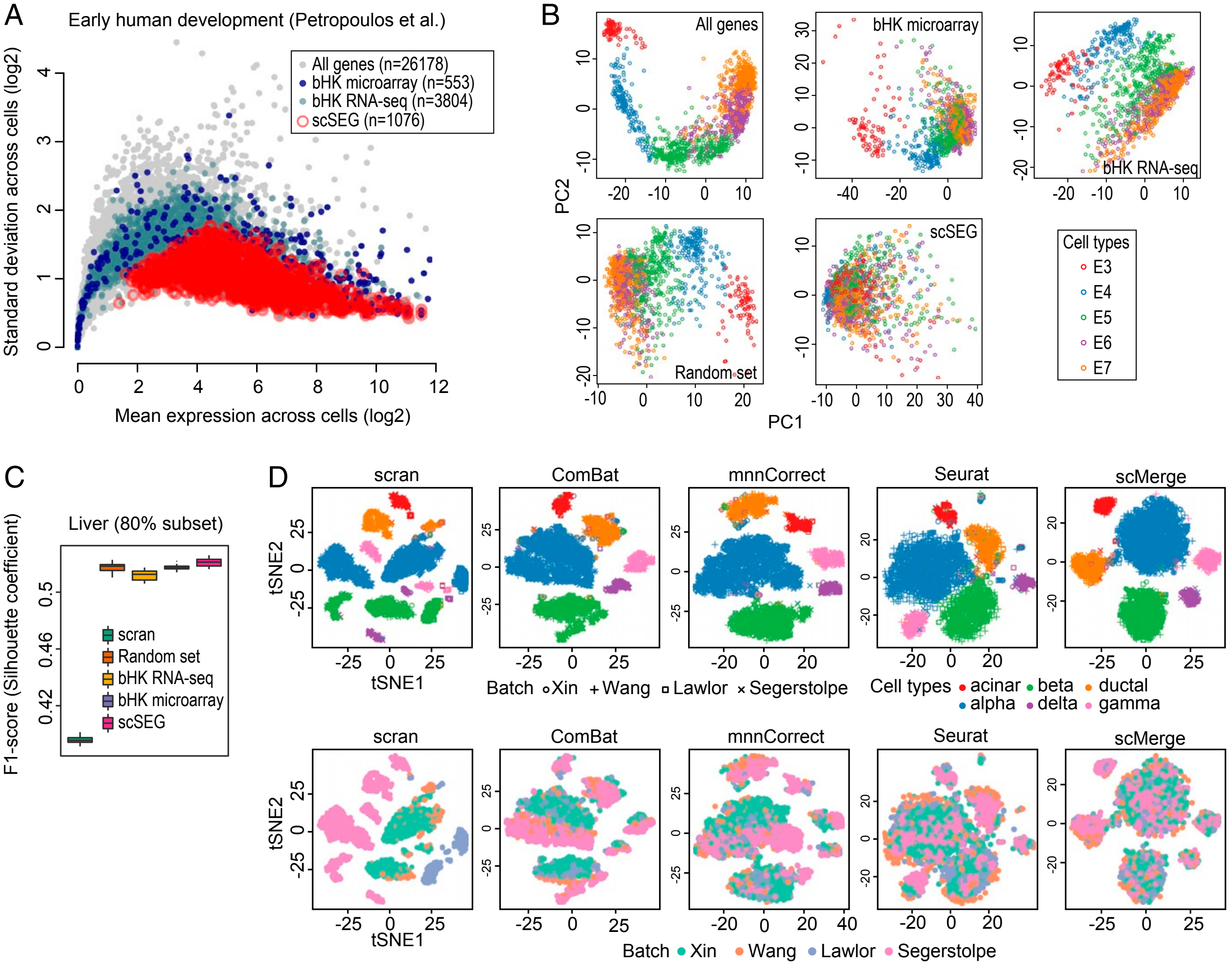


Fig. 2. (*A*) Scatter plot showing mean expression (x axis) and SD (y axis) on log scale of each gene (gray circles) across profiled single cells. Open red circles represent scSEGs derived from human in this study, whereas dark and light blue solid circles represent housekeeping genes defined previously using bulk microarray (bHK microarray) (13) and RNA-Seq (bHK RNA-Seq) (14) data. (*B*) A panel of PCA plots based on all genes or different subsets of genes including bHK microarray, bHK RNA-Seq data, scSEGs, and a random selection of genes, for the Petropoulos et al. (9) data. (*C*) Boxplots comparing the effects of different types of negative controls for Liver data collection. The y axis represents the F1 score of Silhouette coefficients between cell type mixing and (1 − datasets mixing), where higher values are desired. Stratified sampling is performed to randomly subset 80% of cells from the datasets, repeated 10 times to examine stability. (*D*) A 2 × 5 panel of tSNE plots of the Pancreas4 data collection using the output from scran, ComBat, mnnCorrect, Seurat, and scMerge (using scSEGs as negative controls). *Top* is color-coded by cell types, and *Bottom* is color-coded by the distinct Pancreas datasets.

(*SI Appendix*, Fig. S3), potentially due to the exogenous nature of ERCC probes. Conceptually, the choice of scSEGs will have a greater effect when integrating heterogeneous datasets with large differences between cells and a high proportion of highly variable genes (HVGs), and, as a result, appropriate selection of negative control genes has a large influence on the normalization results.

Visual Diagnostic Assessment Demonstrates Effective Removal of Unwanted Variation. To assess the performance of scMerge for integrating multiple scRNA-seq datasets, we collated 14 publicly available scRNA-seq datasets and grouped them into seven distinct and diverse data collections where each characterizes a broad biological system (Fig. 1*B* and *SI Appendix*, Table S1). Each data collection varies across key characteristics, including number of datasets, sequencing platforms, species, and cell type compositions (Fig. 1*B*). We compared scMerge to other approaches, including scran (3), mnnCorrect (4), ComBat (5), Seurat (6), and ZINB-WaVE (7), for normalizing and merging each data collection. The performance of each method was evaluated using multiple criteria, including visual inspection of diagnostic plots, Silhouette coefficients, adjusted Rand indices (ARIs), and downstream biological impact (Fig. 1*C* and *Materials and Methods*).

We find that scMerge effectively removes batch and dataset-specific effects across a wide range of biological systems, including a collection of human pancreatic scRNA-seq datasets. Visual inspection of t-distributed stochastic neighbor embedding (tSNE) plots (Fig. 2*D*), and similarly for principal component analysis (PCA) plots (*SI Appendix*, Fig. S4), shows that, unlike other methods, scMerge clearly separates acinar and ductal cells. Additionally, in scMerge-processed datasets, cell type information explains a higher percentage of “wanted” variation than “unwanted” variation (15) (*SI Appendix*, Fig. S5).

scMerge Outperforms Existing Integration Methods. In general, we found that scMerge performed favorably in terms of maintaining strong biological signal and reducing unwanted variation such as batch and/or data-specific noise in seven data collections (Fig. 3*A* and *SI Appendix*, Figs. S6–S12). The merge results for these datasets are also available on our website. Our evaluation metrics capture the trade-off between these two broad objectives; scMerge manages the trade-off between separating cell types and merging batches well (Fig. 3*A*) across multiple data collections in comparison with other methods. Summarizing these two quantities into a single F1 score, we find that scMerge has better performance (*SI Appendix*, Table S2), despite the choice of Silhouette coefficient or ARI as the comparison metric (*SI Appendix*, Fig. S13*A*). A direct contribution of improved data integration is its impact on downstream analyses. Using the identification of differentially expressed (DE) genes as a performance measure, we found that, in comparison with other methods, data integrated by scMerge led to comparable number of DE genes among different cell types but significantly fewer DE genes across batches (*SI Appendix*, Fig. S13*B*). Together, these results suggest that scMerge is able to maintain or even enhance biological signals while effectively removing unwanted variation.

Effective Removal of Unwanted Variation Improves Cell Trajectory Estimation. To illustrate the capability of scMerge to enable further downstream analyses, we studied the integrated expression matrices of the Liver data collection and examined the stability of cell trajectory reconstruction when faced with incomplete data. We reconstructed cell trajectories, using Monocle 2 (16), of hepatoblasts, hepatocytes, and cholangiocytes for both the full Liver data collection and for a subset of the original Liver data collection, where cells corresponding to the E17.5 time point of GSE90047 (17) were removed. We find that the trajectory associated with scMerge is most consistent with the full Liver data collection (Fig. 3*B*) and agrees with current literature (18) (*SI Appendix*, Fig. S14), while other methods tended to generate extraneous branches with the subset of the Liver data collection.

Cross-Species Integration Confirms Similarity of Human and Mouse Embryonic Development. Finally, we illustrate the potential of scMerge in facilitating fine-grained annotation of cell types during early human and mouse development by integrating the Embryogenesis data collection: seven datasets that profiled human (9, 19–21) and mouse (10, 22, 23) embryogenesis at various stages ranging from zygotes to late blastocysts (Fig. 1*B*). A semisupervised version of scMerge (*Materials and Methods*) was used to normalize and merge the seven datasets by matching cells from 2-cell, 4-cell, 8-cell, and 16-cell stages across these datasets. In the merged dataset, we can accurately annotate epiblast, primitive endoderm, and trophectoderm within blastocysts in the human data using 12 known marker genes. This is confirmed by the high concordance (ARI = 1) of clustering results and cell types annotated by the previous study (24).

Similarly, the merged data also provide finer annotation of the mouse cells from blastocysts where no cell type annotation is available from their original studies (10, 22, 23). This is achieved by training on the merged dataset a Support Vector Machine that learns from the annotated human blastocyst cells (19, 21) and classifying mouse blastocyst cells. Using this approach, we further classified 92 mouse blastocyst cells into trophectoderm (while the rest of the 54 cells remain unassigned) (*Materials and Methods*). This suggests that the majority of mouse blastocysts can be annotated as trophectoderm. This is consistent with current knowledge that cells strongly associated with trophectoderm lineage span across the estimated pseudotime, whereas epiblast and primitive endoderm occurs only in later pseudotime (Fig. 3*C* and *SI Appendix*, Fig. S15). This illustrates how scMerge can be used to match different cell types across species based on their relationship in the dimension-reduced space. In addition to clear time course separation of cells from zygotes to late blastocysts (Fig. 3*D*), we observe clear overlap of many cells from human and mouse blastocysts and inner cell mass regions across the pseudotime trajectory, confirming current understanding of a high similarity between early human and mouse embryonic development and lineage specification.

Discussion

The integration of multiple scRNA-seq datasets remains a great methodological challenge due to technical effects associated with batches, platforms, and species. In this paper, we proposed a comprehensive integration approach for scRNA-seq data. We are able to illustrate the applicability of our scMerge methods on a diverse collection of datasets with varying levels of integration difficulty. In all cases, we have shown that our method, scMerge, performs similarly to or outperforms state-of-the-art methods. Furthermore, our method permits tailored incorporation of prior knowledge via a semisupervised approach, a feature not provided by other approaches, and, as such, provides a practical platform for scientists.

We also introduced an analytic method for identifying scSEGs. While scSEGs defined in this study are used as default negative controls in scMerge, scMerge also accepts other negative control genes defined from alternative sources, using different approaches, or provided by users. Examples include (*i*) housekeeping genes defined from bulk microarray (13), (*ii*) housekeeping genes defined from bulk RNA-seq (14), (*iii*) external spike-in sequences (e.g., ERCC), and (*iv*) empirical scSEGs based on the above identification procedure excluding the F statistics in the index calculation when cell type information is unknown.

As with most algorithms, scMerge requires multiple input parameters, including ones that require tuning. However, we

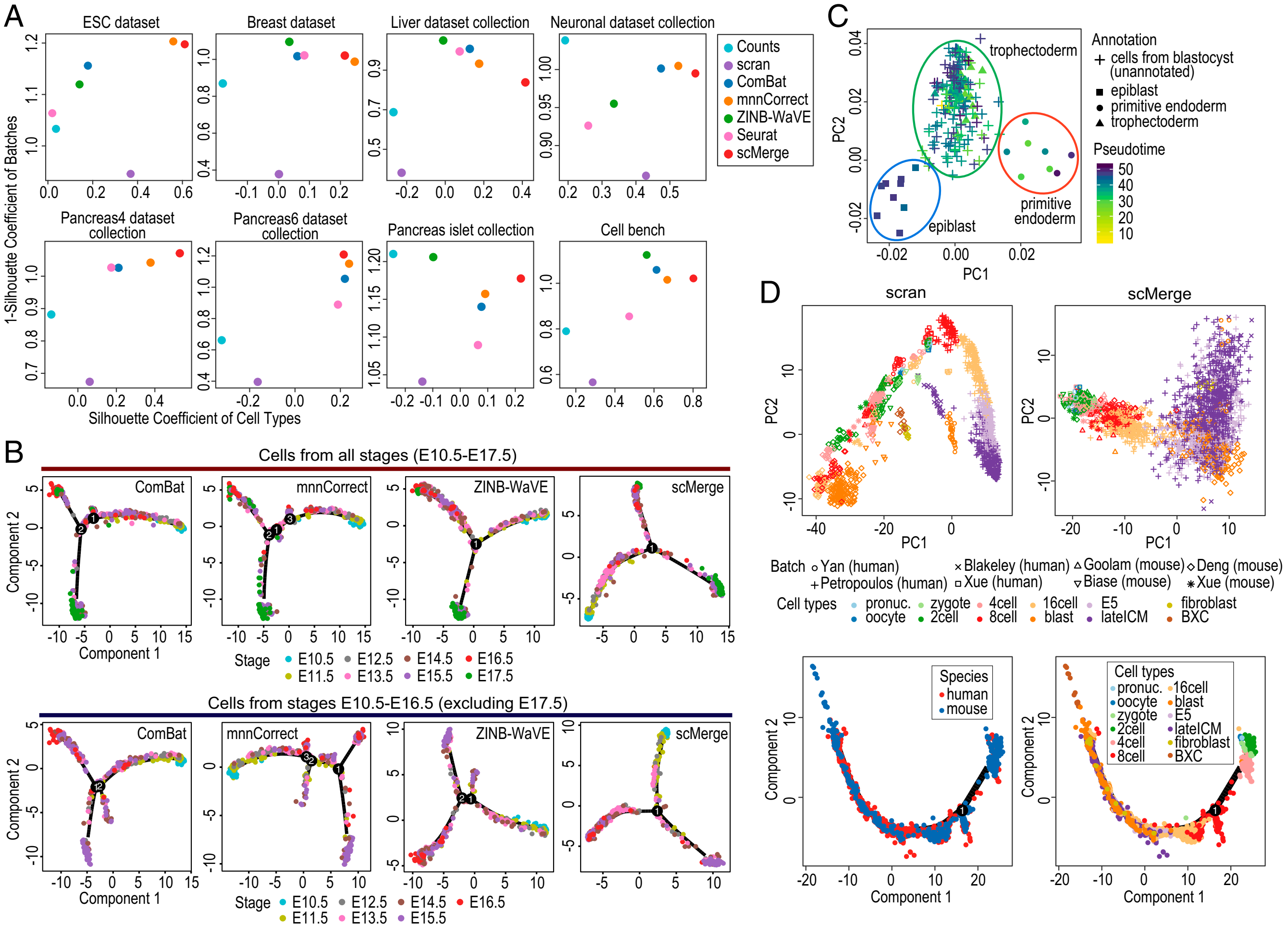


Fig. 3. (*A*) A 2 × 4 panel of scatter plots of Silhouette coefficients for no normalization (Counts), scran, ComBat, mnnCorrect, ZINB-WaVE, Seurat, and scMerge (using scSEGs as negative controls). The *x* axes denote the Silhouette coefficient of cell types, and *y* axes denote the 1 − Silhouette coefficient of batch effects, where desirable outcomes are in the top right-hand corner. (*B*) A 2 × 4 panel of pseudotime trajectories demonstrating the stability of scMerge. *Top* displays the trajectories from Monocle 2 using hepatoblasts, hepatocytes, and cholangiocytes from all time points in the Liver data collection, and *Bottom* displays the trajectories from Monocle 2 with the Liver data collection with time point E17.5 removed. (*C*) A PCA plot of blastocyst cells from dataset generated by Blakeley et al. (21). Blastocyst cells from all other datasets [including Yan et al. (19), Biase et al. (23), Goolam et al. (22), and Deng et al. (10)] are projected on the same PCA plot. Cells are color-coded by pseudotime. (*D*) A 2 × 2 panel of PCA plots and pseudotime trajectories of the Embryogenesis data collection following scMerge (using scSEGs as negative controls). *Top Left* is PCA plot of the ESC data collection for scran, color-coded by developmental time point. *Top Right* is PCA plot of the ESC data collection for scMerge, color-coded by developmental time point. *Bottom* are the pseudotime trajectory of ESC data collection after scMerge, color-coded by (*Left*) developmental time point and (*Right*) species.

found that the algorithm's performance was not sensitive to realistic variation in certain key parameters. Firstly, the vector of the numbers of clusters, which is presumed to be the number of cell types in the batch or the expected number of clusters in each batch, is robust to overestimation (*SI Appendix*, Fig. S17*A*). Secondly, the default setting of the number of factors of unwanted variation (i.e., ref. 20) is robust to different datasets and insensitive to overestimation (*SI Appendix*, Fig. S17 *B* and *C*).

A fastRUVIII was implemented in scMerge to improve the computational efficiency of RUVIII for large-scale scRNA-seq data. It uses randomized singular value decomposition (RSVD) (25), a fast probabilistic algorithm implemented as the *rsvd()* function in rsvd (26). The computational time of scMerge using these two different SVD methods on the Pancreas-4 data is shown in *SI Appendix*, Fig. S16. This dataset consists of 23,699 genes and 4,566 cells, and the RSVD method is able to reduce the computational time from a standard SVD by fivefold (*SI Appendix*, Fig. S16*A*). By evaluating the gene-wise correlation between the full decomposition and the RSVD approximations under various proportion parameters, we show that the RSVD approximations are typically of very high quality (*SI Appendix*, Fig. S16*B*). We found, empirically, that an RSVD proportion of 0.01 typically allows for a balance between the computational performance and numerical accuracy. We will be able to achieve further speed improvement with parallelization of the pseudoreplicates algorithm.

Benchmarking computational tools should not be limited to only one approach, as most methods aim to balance between multiple competing criteria (compare the well-known “bias and variance” trade-off). Given that any specific benchmarking model will have its own strengths and weaknesses and therefore provides a different “view” of the process, we use (*i*) a collection of diagnostic plots, (*ii*) quantitative metrics, and (*iii*) perturbation of real data to examine the performance and utilities of scMerge. Here, we visualize and evaluate the results of various integration methods using PCA plots, tSNE plots, relative log expression (RLE) plots, and density plots of percentage of variance explained; all of these have illustrated the consistency of scMerge for noise removal. Furthermore, the utility and robustness of scMerge for improving cell trajectory estimation is demonstrated through the analysis of both the Liver data collection and cross-species embryonic stem cells (ESCs) data collection. Other downstream analytics such as clustering for cell type discovery and prediction are also possible. It is worth noting, however, for identification of marker genes using various DE algorithms, that histograms of P values should be used to diagnose and ensure that unintended reduction of the variation between cells of a given two cell types has no significant impact.

In summary, scMerge enables integrative analysis of multiple scRNA-seq data in a holistic manner. While several other state-of-the-art methods can be tuned and tweaked for combining scRNA-seq datasets, scMerge performs consistently better in terms of cell type separation and unwanted factor removal with its specific design for a broad scenario of scRNA-seq data integration. The downstream impact of scMerge is further demonstrated through integrating large collections of embryogenesis datasets across human and mouse and subsequently annotating cell types in development across different species.

Code Availability

The R package scMerge is available on the Github repository https://sydneybiox.github.io/scMerge.

Materials and Methods

scMerge. The main inputs to scMerge consist of a list of SEGs (either predefined or identified from current datasets), preferably with gene-wise standardized log transformation [e.g., log2(cpm + 1)] of multiple datasets that are to be normalized and merged into a single dataset. scMerge utilizes the RUVIII procedure to adjust the data by identifying pseudoreplicates and using the SEGs as negative control genes, with the number of unwanted variation factors (k_{RUVIII}) defaulted to 20.

Gamma–Gaussian Mixture Model-Based Stably Expressed Features. To characterize gene expression patterns from an scRNA-seq dataset, we use a Gamma–Gaussian mixture model (8) to fit gene expression values across individual cells. Specifically, nonzero expression values x_i [on log2 of the fragments per kilobase of transcript per million mapped reads (log2FPKM) scale] of gene i across cells are modeled by a mixture of distributions consisting of a Gamma component, corresponding to cells in which the gene is expressed at a low level, and a Gaussian component, corresponding to those in which the gene is expressed at a high level. The joint density function of the mixture model for the ith gene is defined as follows:

$$f(x_i;\alpha_i,\beta_i,\mu_i,\sigma_i^2,\lambda_i)=\lambda_i\frac{\beta_i^{\alpha_i}}{\Gamma(\alpha_i)}x_i^{\alpha_i-1}e^{-\beta_i x_i}+(1-\lambda_i)\frac{1}{\sigma_i\sqrt{2\pi}}e^{-\frac{(x_i-\mu_i)^2}{2\sigma_i^2}},$$

where α_i and β_i denote the shape and rate parameters of the Gamma component, μ_i and σ_i^2 denote the mean and variance of the Gaussian component, and $0 \leq \lambda_i \leq 1$ is the mixing proportion indicating the proportion of cells in the Gamma component. The mixture model parameters can be estimated using the Expectation-Maximization algorithm. In our Gamma–Gaussian mixture model setting, genes with a low mixing proportion (λ_i) and a small variance (σ_i^2) with respect to the Gaussian component suggest, respectively, that a unimodal and an invariant expression pattern across the profiled single cells are therefore more likely to be SEGs.

We also consider the fraction ω_i of zeros for gene i across all cells as one of the characteristics of stable gene expression. There are a number of reasons why the measured expression level for a given gene and cell may be zero, including technical dropout due to failure to amplify the RNA from a small amount of starting material (27), stochastic expression patterns (28), and, of course, if no transcription is occurring for that gene. Therefore, a desired characteristic of SEGs is a relatively small ω value (i.e., low proportion of zeros) observed in scRNA-seq data, since we expect these genes to be stably expressed in all cells. One confounding factor is that SEGs with low expression levels may have higher proportions of zeros than SEGs with high expression levels simply due to technical dropout events, as opposed to the underlying biology. Our approach to account for this confounding factor is to take into consideration average expression level μ_i in the Gaussian component of gene i as follows:

$$\omega_i^{*}=\sqrt{\omega_i\cdot\frac{\mu_i-\min(\mu_i)}{\max(\mu_i)-\min(\mu_i)}},$$

where the min and max are over all genes. This is because we anticipate more dropout events for SEGs with lower expression levels compared with SEGs with higher expression levels. Since both ω and minmax normalized μ range from 0 to 1, we take the square root of their multiplication to obtain the mean value. Ideally, the genes with small values of these three stably expressed features derived from the estimated mixture model correspond to SEGs.

F Statistic for Equivalent Expression Across Predefined Experimental Conditions. We use an F statistic as another stably expressed feature to select for genes which have similar average gene expression levels across different predefined groups of experimental replicates, cell types, tissues, and individuals. Specifically, our F statistic is the one used in one-way analysis of variance, namely,

$$F=\frac{\left(\sum_{k=1}^{P}n_k(\bar{x}_{k.}-\bar{x}_{..})^2\right)/(P-1)}{\left(\sum_{k=1}^{P}\sum_{l=1}^{n_k}(\bar{x}_{kl}-\bar{x}_{k.})^2\right)/(N-P)},$$

for N cells across P groups with n_k cells in the kth group, with dots denoting group means across the group index k and sample index l. This F statistic quantifies departure from the ideal scenario of equal means across groups, and so we would expect to observe a small F statistic associated with the experimental conditions for SEGs. This F statistic forms our fourth stably expressed feature, where we set the predefined class label as the associated experimental condition when available.

SEG Index. Genes with small λ, σ^2, $\omega\star$, and F statistic are more likely to be SEGs. We refer to these four quantities as stably expressed features. By combining these four stably expressed features, we define a stably expressed index (SEG index) for each gene. Specifically, we first ranked genes in increasing order with respect to λ, σ^2, ω^*, and F statistics, respectively. Next, we rescaled the ranks of each of the four stably expressed features to lie between 0 and 1, and then defined the SEG index for each gene as the average of its scaled ranks across all four stably expressed features. Thus, SEGs can be selected by adjusting the SEG index threshold and can be subsequently validated using a panel of evaluation matrices. Importantly, genes can also be ranked in terms of their degree of evidence toward characteristics of SEGs.

scSEG List. We derived separate lists for human and mouse by computing the rank percentiles of SEG index as well as the four stably expressed features. Genes with an SEG index rank percentile above 80 as well as a reversed rank percentile above 60 for each of the four stably expressed features were included in the scSEG gene list. Using this approach, we identified 1,076 and 830 human scSEG (h-scSEG) and mouse scSEG (m-scSEG) genes, respectively, given in Dataset S1.

Pseudoreplicates. For an unsupervised approach to the identification of pseudoreplicates, we propose the following stepwise procedure.

Step 1: Identify HVGs as the union of all batch-specific HVGs, where these are defined as genes that are expressed in more than 10% of the cells using the Brennecke method implemented in the package *M3Drop* (29). Within each batch, we perform K-means clustering on the top 10 principal components (PCs) based on the HVGs, where the number of clusters ($k_{Cluster}$) is the presumed number of cell types.

Step 2: Identify mutual nearest clusters (MNCs) from the batches. We start from the identification of each pair of batches. We use Pearson correlation as the dissimilarity metric (30) and define the distance between two clusters as the median distance based on the common HVGs between pairs of cells. Let d_{ab} denote the distance between cell a of cluster A and cell b of cluster B, which can be calculated as

$$d_{ab}=1-\frac{\sum_{i=1}^{G}(Y_{ia}-\bar{Y}_{.a})(Y_{ib}-\bar{Y}_{.b})}{\sqrt{\sum_{i=1}^{G}(Y_{ia}-\bar{Y}_{.a})^2}\sqrt{\sum_{i=1}^{G}(Y_{ib}-\bar{Y}_{.b})^2}},$$

where Y_{ij} denotes the log-transformed gene expression of gene i in cell j, with $i=1,\ldots,G$. Then the distance between cluster A and cluster B, d_{AB}, is defined as

$$d_{AB} = \text{median}(d_{ab}).$$

For any two batches within the dataset, we calculate all of the pairwise distances between the clusters in the two batches. If each of a pair of clusters considers the other as the nearest cluster, we regard them as MNCs. Note that this procedure is not recommended for pairs of batches where both batches only have one cluster, since the estimated clusters from each of the two batches will always consider each other as MNCs. For any batch with only one estimated cluster, the algorithm only identifies MNCs from the batches with at least two clusters. The same procedure is repeated for every pair of batches, which provides us with a list of MNC pairs.

Step 3: Identify MNC subgraphs. We generate a graph with each cluster as a node and each MNC pair as an edge. The graph may be partitioned into subgraphs by the fast greedy modularity optimization (31) algorithm implemented in the R package *igraph* to find dense subgraphs. The clusters within the same subgraph are therefore considered as an MNC subgraph. For the clusters that do not belong to any subgraph, we consider these clusters as their own MNC subgraph, containing just the single cluster.

Step 4: Identify pseudoreplicates from MNC subgraphs. For each of the clusters in each MNC subgraph, we identify the top 50% set (by default) of cells closest to the cluster centroid, by calculating the Euclidean distance (across the HVG genes) of the cells from the centroid of the assigned cluster. The sets of cells identified in this way are classified as pseudoreplicates, resulting in sets of pseudoreplicate cells equal to the number of MNC subgraphs.

Step 5: A binary replicate matrix is then created from the pseudoreplicates constructed as above, with rows of the matrix corresponding to cells and columns corresponding to sets of pseudoreplicates, equal to the sum of the number of MNC subgraphs and the number of cells that were not classed as pseudoreplicates. Note that each cell is assigned to one and only one set of pseudoreplicates, i.e., each row contains only one positive “1” value.

For the semisupervised approach, where we have a priori information, we first perform the same identification of pseudoreplicates as described above. If the cell type information is known, we can revise the pseudoreplicate replicate matrix by merging the a priori cell types into one pseudoreplicate. Moreover, if some other factors of interest are known, such as developmental stage, the pseudoreplicates identified here are further split according to the known factors of interest.

fastRUVIII. To remove the unwanted variation across multiple datasets and batches, we developed and incorporated a fast version of RUVIII (12) model in scMerge; fastRUVIII extends on R package *ruv* by speeding up the computational speed by up to fivefold from its original implementation. Specifically, fastRUVIII uses the gene-wise standardized data as an input instead of the log-transformed data, and, as such, the overall mean and variance of genes have similar values. Let Y_{ibc} be the size factor normalized then log-transformed expression value of gene i in cell c within batch b, where $i = 1, \ldots, G$; $b = 1, \ldots, B$; $c = 1, \ldots, C_b$, where C_b indicates the number of cells in batch b. Let C be the total number of cells in dataset, with

$$C = \sum_b C_b.$$

The standardized data Z_{ibc} can be calculated as

$$Z_{ibc} = \frac{Y_{ibc} - Y_{i..}}{s_i},$$

where $Y_{i..}$ is the average expression of gene i across cells and batches calculated by

$$Y_{i..} = \frac{1}{C} \sum_{bc} Y_{ibc};$$

s_i is the corresponding SD of gene i, calculated by

$$s_i^2 = \frac{1}{(C - B)} \sum_{bc} \left(Y_{ibc} - \frac{1}{C_b} \sum_b Y_{ibc} \right)^2.$$

The standardized data $Z_{C \times G}$ can be fitted to the model underlying the RUVIII model, which is formulated as

$$Z_{C \times G} = X_{C \times p} \beta_{p \times G} + W_{C \times k} \alpha_{k \times G} + \epsilon_{C \times G},$$

where X is the matrix of factor of interest; p is the number of factors of interest; W is the unobserved design matrix corresponding to the unwanted factors; k is the linear dimension of the unwanted factors, which is unknown; and ϵ denotes the random error. RUVIII adjusts the data Z in three steps, as follows: (*i*)Calculate the residuals of Z with respect to pseudoreplicates identified from the dataset, and estimate α using randomized SVD on these residuals; estimate W using the negative control subset of α; multiply the estimates of α and W, and then subtract this product from the data.

The final adjusted data $\hat{Y}_{ibc}$ is calculated by

$$\hat{Y}_{ibc} = s_i \times \hat{Z}_{ibc} + Y_{i..}.$$

Note that, to improve the computational efficiency of RUVIII in large-scale scRNA-seq data, fastRUVIII uses RSVD (25), a fast probabilistic algorithm implemented in the *rsvd()* function in rsvd (26).

The speed-up is achieved via the use of a proportion parameter to perform a lower-dimension approximation.

Benchmark Data and Data Processing. A summary of the following seven datasets or data collections is provided in Fig. 1*B* and *SI Appendix*, Table S1. Datasets refer to single sets of data, which can contain multiple batches, while data collections refer to sets of multiple distinct datasets that may come from different experiments, protocols, and species.

i) The mouse ESC (mESC) dataset generated by Kolodziejczyk et al. (32) includes the single-cell RNA-sequencing of cells cultured in three different conditions (serum, 2i and a2i) from five batches. The raw count data were downloaded from https://www.ebi.ac.uk/teichmann-srv/espresso/, which includes the cell culture labels and batch information.

ii) The Breast cancer dataset is a single-cell dataset profiling the transcriptomes of 25,790 primary human breast epithelial cells isolated from reduction mammoplasties of seven individuals (33). The data contain three main cell types with four different batches. The raw count data were downloaded from National Center for Biotechnology Information (NCBI) Gene Expression Omnibus (GEO) accession number GSE113197. The cell type labels were inferred using the method of the original publication.

iii) For the Liver data collection, raw count matrices were downloaded from NCBI GEO accession number GSE90047 (17) and GSE87038 (34). Raw fastq files were obtained from GEO accession number GSE87795 (35) and GSE96981 (36) and processed based on our scRNA-Seq pipeline, described in *Data Processing*. Note that, for GSE87795, we included the 389 cells from embryonic day (E)11.5 to E16.5 that were assigned with cell labels in Su et al. (35). Given that we have prior knowledge of time point, we used a semisupervised approach to identify the pseudoreplicates of the Liver dataset collection, where cells of the four liver datasets are from different fetal mouse liver developmental stages (E9.5 to E17.5). To identify the pseudoreplicates corresponding to the hepatoblasts, hepatocytes, and cholangiocytes, we used three known markers of hepatoblasts and cholangiocytes, Alb, Afp, and Epcam, to guide the scMerge algorithm. The mean expression of these markers was calculated for each set of pseudoreplicates, and sets of pseudoreplicates were then classed into groups with high or low expression of the marker genes via k-means clustering. In the sets of pseudoreplicates that are highly expressed, the markers are further split according to the developmental stages.

iv) For the Olfactory neuronal data collection, raw sequencing reads were downloaded from GEO, the Sequence Read Archive (SRA) for GSE75413 (37) and SRP065920 (38), respectively. Gene expression count matrices were generated from raw fastq files using our scRNA-seq processing pipeline described in *Data Processing*. We defined the maturity of neuronal cells using Omp and Gap43 as marker genes.

v) For the Pancreas data collection (including Pancreas Islet dataset), raw count matrices of the four pancreas datasets were downloaded from GEO and European Bioinformatics Institute (EBI) accession numbers GSE86469 (39), E-MTAB-5061 (40), GSE85241 (41), and GSE84133 (42). For pancreas datasets GSE81608 (43) and GSE83139 (44), we downloaded the raw fastq files from the NCBI SRA using the fastq-dump utility. We then mapped the fastq files to the hg38 human genome reference and the gencode v26 transcriptome reference using the Spliced Transcripts Alignment to a Reference (STAR) aligner (version 2.5.3a) (45). The resulting mapped read files were then converted to bam and sorted and indexed using Samtools (46), and read counts were obtained using the HTSeq software (47). The cells with less than 300,000 reads or that expressed less than 3,000 genes were removed for the downstream analysis. We included the cell types that exist in at least two datasets, which are acinar, alpha, beta, delta, ductal, and gamma. In benchmarking “Pancreas 4,” we integrated the datasets GSE81608, GSE83139, GSE86469, and E-MTAB-5061 from Smart-seq/Smart-seq2. In benchmarking “Pancreas 6,” we further integrated the datasets from two other protocols, GSE85241 (CEL-seq2) and cells from human in GSE84133 (inDrop). In benchmarking "Pancreas islets," we integrated the data from human and mouse of GSE84133.

vi) The Cell Bench dataset is a benchmarking dataset created using three human lung adenocarcinoma cell lines, HCC827, H1975, and H2228, which were cultured separately, and the same batch was processed in three different ways (48). We considered the three different cell lines as different "cell types" and combined the data generated from three different protocols: CEL-seq2, Drop-seq with Dolomite equipment, and Drop-seq with 10X Chromium. The data were downloaded directly from https://github.com/LuyiTian/CellBench_data/.

vii) For the ESC data collection, we collected four human and four mouse datasets profiling ESCs. The processed count data or FPKM data with accession ID numbers GSE45719 (10), GSE57249 (23), GSE53386 (49), GSE44183 (20), GSE36552 (19), and GEO66507 (21) were downloaded from the GEO, while E-MTAB-3321 (22) and E-MTAB-3929 (9) were downloaded from the EBI ArrayExpress website.

Data processing. For the scRNA-seq processing pipeline, all fastq files from mouse tissues were mapped to the mm10 reference and the gencode.vM14 transcriptome supplemented with ERCC sequences using the STAR aligner (version 2.5.3a) (45). The resulting mapped read files were then converted to bam and sorted and indexed using Samtools (46), and read counts were obtained using the HTSeq software (47).

For all datasets described in *Benchmark Data and Data Processing*, only cells that passed the quality control of the original publication were included. We first performed size factor standardization to the raw count matrices for each batch/dataset using the *normalize()* function in the R package *scater* (15). We then filtered genes such that genes were expressed (nonzero value) in at least 1% of cells per batch/dataset, as well as expressed in at least 1% across the entire set of batches/datasets. To combine the data from human and mouse, we identified and matched all homologous gene pairs. Cells were labeled by the cell type information provided by the original publications. When integrating datasets from protocols with vastly different sequencing depths due to different platforms (e.g., Drop-seq and SMART-seq) as was the case for the Pancreas6, Cell Bench dataset, and ESC data collection, we performed cosine standardization as a preprocessing step before performing scMerge, which is calculated by

$$\tilde{Y}_{ij} = \frac{Y_{ij}}{\sqrt{\sum_{i=1}^{G} Y_{ij}^2}},$$

where Y_{ij} denotes the log-transformed gene expression of gene i in cell j.

Evaluation metrics.

Silhouette coefficient. To assess the extent to which the gene expression data are grouped based on the batch effect as opposed to biological signal, we used the silhouette coefficient in a similar manner to scone and kBET (50, 51). Let $a(j)$ denote the average Euclidean distance over the first three PCs of the expression matrix between the cell j and all other cells in the same group to which cell j is assigned. Let $b(j)$ be the minimum average distance between the cell j and cells in all other groups. The silhouette coefficient of cell j is calculated as

$$s(j) = \frac{b(j) - a(j)}{\max\{a(j), b(j)\}} \in [-1, 1], j = 1, \ldots, C.$$

A value of $s(j)$ close to 1 indicates that the cell j is appropriately assigned in the group. The average silhouette width across all of the cells is calculated by

$$s = \frac{1}{C}\sum_{j=1}^{C} \text{sil}(j).$$

We calculated the average silhouette width of the expression data using two different groupings: (*i*) grouping based on the batch as batch silhouette coefficient (s_{batch}) and (*ii*) grouping based on known cell types as the cell type silhouette coefficient ($s_{cellTypes}$). Ideally, the batch effect corrected expression matrix has a small s_{batch}, which indicates the cells are not grouped by batch, and a large $s_{cellTypes}$, which suggests the preservation of biological signal. To summarize these two evaluation measures, we calculated the harmonic mean of these two silhouette coefficients following transformation and scaling to $[0, 1]$, called the F1 score, given by

$$F1_{sil} = \cdot \frac{(1 - s'_{batch}) \cdot s'_{cellTypes}}{1 - s'_{batch} + s'_{cellTypes}} \in [0, 1],$$

where $s' = (s + 1)/2$. An expression matrix with either a strong batch effect or low biological signal will have a low F1 score, while a high F1 score of an expression matrix suggests the successful removal of batch effects and the preservation of biological variation of interest.

ARI. To assess the clustering analysis performance of the adjusted gene expression matrix, we used ARI to evaluate the concordance of clustering results with respect to the cell type labels and the batch, denoted respectively as $ARI_{cellTypes}$ and ARI_{batch}. Considering the cells are partitioned into different classes with respect to cell type labels or batch, let a be the number of pairs of cells partitioned into the same class by a clustering method, b be the number of pairs of cells partitioned into the same cluster but in fact belong to different classes, c be the number of pairs of cells partitioned into different clusters but belong to the same class, and d be the number of pairs of cells from different classes partitioned into different clusters. Then the ARI is calculated as

$$\text{ARI} = \frac{2(ad - bc)}{(a + b)(b + d) + (a + c)(c + d)}.$$

A high $ARI_{cellTypes}$ indicates a high concordance between the clustering result and known cell type information, whereas a high ARI_{batch} indicates the clustering result is dominated by the batch effect. In a similar manner to the silhouette coefficients, we scaled the ARI values into the range of $[0, 1]$ as ARI', and then summarized the them into an F1 score as follows:

$$F1_{ARI} = 2 \cdot \frac{(1 - ARI'_{batch}) \cdot ARI'_{cellTypes}}{1 - ARI'_{batch} + ARI'_{cellTypes}} \in [0, 1].$$

Diagnostic plots. To assess whether the batch correction method or the choice of parameters of scMerge is suitable for a certain dataset, we have provided three kinds of diagnostic plots to visualize and evaluate the results of the batch correction methods: (*i*) PCA or tSNE plots, (*ii*) RLE plots, and (*iii*) density plots of the percentage of variance explained by wanted and unwanted variation factors.

i) PCA plots were generated using the union of HVGs identified from each batch. For tSNE plots, the *Rtsne* package was used for the first 10 PCs of the corrected gene expression matrices with default parameter settings (perplexity 30) for the uncorrected log-transformed normalized data, and batch-corrected expression data with mnnCorrect, ComBat, ZINB-WaVE, and scMerge. For Seurat, we generated the tSNE plots based on 20 canonical vectors.

ii) RLE plots are a useful tool to visualize unwanted variation (52). RLE plots are boxplots of RLE for each cell, calculated as $Y_{ij} - \text{med}(Y_{i.})$, where $\text{med}(Y_{i.}) = \text{median}\{Y_{ij} : j = 1, 2, \ldots\}$, and Y_{ij} is the log expression value of gene i in cell j. If the cell type information is known, we can also generate multiple RLE plots per cell type. If the unwanted variation is removed, the cells from different batches should have a similar distribution, and the medians of the boxplots should be close to zero.

iii) To evaluate the association of wanted and unwanted variation as well as technical variables (total number of counts, total number of features, and percentage of ERCC counts for the datasets with ERCC spike-ins controls), we calculated the coefficient of determination (R^2) for a linear regression model for each gene, with technical, batch (unwanted variation), and cell type (wanted variation) variables. Following the batch effect removal, the percentage of variance explained by the wanted variation should ideally be greater than the unwanted variation.

Evaluation and Assessment.

Benchmarking methods. In our benchmarking, (*i*) for mnnCorrect, we first performed per-batch scaling normalization using *multibatch()* and then ran *mnnCorrect()* with the default setting using HVGs to identify mutual nearest neighbors, which is implemented in scran package version 1.9.12. (*ii*) For ComBat, we used the batch information to create a design matrix to correct the batch effect, implemented in the sva package version 3.29.0, and used the log-transformed size factor normalized data as the input of the algorithm. (*iii*) For Seurat (version 2.3.4), we normalized the data using *NormalizeData()*, scaled the data using *ScaleData()*, and identified the variable genes using *FindVariableGenes()* within each batch. We then performed a canonical correlation analysis with 20 canonical vectors using the union of variable genes from each batch. (*iv*) For ZINB-WaVE (version 1.3.3), we set the number of latent factors as $K = 2$ for mESC, Olfactory neuronal, Liver, and cellBench data collections, and as $K = 10$ for other data collections. Batch information was included as sample-level covariates and the normalized matrix obtained. Note that ZINB-WaVE only takes integer count matrices as input, and so could not be used for the Pancreas data collection analysis, as GSE86469 was provided as noninteger expected raw counts values from GEO.

Evaluation metrics for scSEG. To evaluate the scSEG, we performed repeated stratified subsampling for the Pancreas dataset collection (four human pancreas datasets) and the Liver dataset collection (four mouse liver datasets). In each stratified subsampling, we randomly selected 80% or 20% (Fig. 2*C* and *SI Appendix*, Fig. S2) of the cells. Then we performed RUVg (53) on the subsample using different negative control gene lists [human-scSEG or mouse-scSEG, bulkHK (bHK) RNA-seq, bHK Microarray, and random subset of genes, equal in number to the scSEG]. We evaluated choices of scSEGs using the F1 score of silhouette coefficients as described above.

Evaluation approaches.

Diagnostic level. All diagnostic plots were generated and assessed for all methods that returned the normalized expression matrix across all data collections.

F1 measures across all metrics. For each data collection and each comparison method, we calculated all metrics. This generated $F1_{sil}$ and $F1_{ARI}$, which were used to compare all methods across all data collections.

Liver cell pseudotemporal trajectory reconstruction. We reconstructed the dynamic trajectory of hepatoblasts, hepatocytes, and cholangiocytes in the Liver dataset collection using Monocle 2 (54) by using the HVG as ordering genes, where the DDRTree (discriminative dimensionality reduction trees) method was used as the dimensionality reduction method. To evaluate the robustness of the adjustment methods, we removed the cells at E17.5 from GSE90047 and evaluated the change in trajectories among the different batch correction methods.

DE analysis. We performed DE analysis on the mESC data as well as on hepatocytes and cholangiocytes of the Liver dataset collection. We evaluated the performance by comparing the number of DE genes between two cell types and the number of DE genes within one cell type and between two batches where no DE genes are expected. DE genes are called using limma-trend (55). The significance threshold was set to absolute log fold change greater than 2 and adjusted *P* value less than 0.05.

ESC data collection and data analytics method. We integrated seven human and mouse ESC datasets using semisupervised scMerge algorithm where we matched developmental time points for 2-cell, 4-cell, 8-cell, and 16-cell stages between human and mouse. We selected to match these developmental stages because only after 16-cell stages does lineage segregation appear in both human and mouse (24). Dynamic cell trajectory of all cells were reconstructed using Monocle 2 (54) with DDRTree as the dimension reduction method. Ordering genes were defined by DE genes in at least three pairwise comparisons of developmental time points ("pronucleus, oocyte, zygote" vs 2-cell, 2-cell vs 4-cell, 4-cell vs 8-cell, and 8-cell vs 16-cell) based on moderated *t* tests in limma (55) (with adjusted *P* value $<$ 0.001). We then refined subtype annotation of blast cells. The previous study (24) further refined subtype annotation of human blastocyst cells into epiblast, primitive endoderm, and trophectoderm. Fig. 3*D* illustrates that using 12 known markers, NANOG, SOX2, KLF17, TDGF1, PDGFRA, GATA6, GATA4, SOX17, GATA3, GATA2, KRT18, and TEAD346, we can accurately annotate human blastocyst cells from Yan et al. (19) onto the first two PCs of blastocyst cells from another data set [Blakeley et al. (21)] (*SI Appendix*, Fig. S15*B*). Mouse blastocyst cells were further refined by projecting these cells [from Goolam et al. (22), Biase et al. (23), and Deng et al. (10)] on the same two PCs (*SI Appendix*, Fig. S15*B*), where we observed that most of the blastocyst cells overlay on the trophectoderm cells from human. We further classified the mouse blastocyst cells using Support Vector Machines using *svm()* in e1071 package (56) by training on the primitive endoderm, and trophectoderm cells from Yan et al. (19) and Blakeley et al. (21). The cells with prediction probability higher than 0.8 were assigned to the corresponding cell type, and otherwise were classified as "unassigned" (Dataset S2).

ACKNOWLEDGMENTS. We thank all of our colleagues, particularly at The University of Sydney, School of Mathematics and Statistics, for their support and intellectual engagement. We also thank Mark Greenaway and Dr. Garth Tarr for their assistance with package development. The following sources of funding for each author, and for the manuscript preparation, are gratefully acknowledged: Australian Research Council Discovery Project Grant DP170100654 (to J.Y.H.Y., J.T.O., and P.Y.); Discovery Early Career Researcher Award DE170100759 (to P.Y.); Australia National Health and Medical Research Council Career Developmental Fellowship APP1111338 (to J.Y.H.Y.); the Judith and David Coffey Life Laboratory at the Charles Perkins Center, The University of Sydney (S.G.); Australian Postgraduate Award (to K.Y.X.W.); Research Training Program Tuition Fee Offset and Stipend Scholarship (to Y.L.); and NHMRC Program Grant 1054618 (to T.P.S.). J.A.G.-B. was supported by the National Science Foundation under Grant DMS-1646108. Additional funding was provided by Shanghai Jiao Tong University and The University of Sydney Systems Biomedicine and Translational Medicine Fund (Grant AF6260003 to Y.L., X.S., Z.-G.H., P.Y., and J.Y.H.Y.). The funding source had no role in the study design; in the collection, analysis, and interpretation of data; in the writing of the manuscript; and in the decision to submit the manuscript for publication.

1. Adhemar Jaitin D, et al. (2014) Massively parallel single-cell RNA-seq for marker-free decomposition of tissues into cell types. *Science* 343:776–779.
2. Bacher R, et al. (2017) Scnorm: Robust normalization of single-cell RNA-seq data. *Nat Methods* 14:584–586.
3. Lun ATL, McCarthy DJ, Marioni JC (2016) A step-by-step workflow for low-level analysis of single-cell RNA-seq data with bioconductor. *F1000Research* 5:2122.
4. Haghverdi L, Lun ATL, Morgan MD, Marioni JC (2018) Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. *Nat Biotechnol* 36:421–427.
5. Johnson WE, Li C, Rabinovic A (2007) Adjusting batch effects in microarray expression data using empirical bayes methods. *Biostatistics* 8:118–127.
6. Butler A, Hoffman P, Smibert P, Papalexi E, Satija R (2018) Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nat Biotechnol* 36:411–420.
7. Risso D, Perraudeau F, Gribkova S, Dudoit S, Vert J-P (2018) A general and flexible method for signal extraction from single-cell RNA-seq data. *Nat Commun* 9:284.
8. Ghazanfar S, Bisogni AJ, Ormerod JT, Lin DM, Yang JYH (2016) Integrated single cell data analysis reveals cell specific networks and novel coactivation markers. *BMC Syst Biol* 10:127.
9. Petropoulos S, et al. (2016) Single-cell RNA-seq reveals lineage and x chromosome dynamics in human preimplantation embryos. *Cell* 165:1012–1026.
10. Deng Q, Ramsköld D, Reinius B, Sandberg R (2014) Single-cell RNA-seq reveals dynamic, random monoallelic gene expression in mammalian cells. *Science* 343:193–196.
11. Cockburn K, Rossant J (2010) Making the blastocyst: Lessons from the mouse. *J Clin Invest* 120:995–1003.
12. Gagnon-Bartsch JA, Speed TP (2012) Using control genes to correct for unwanted variation in microarray data. *Biostatistics* 13:539–552.
13. Eisenberg E, Levanon EY (2003) Human housekeeping genes are compact. *Trends Genet* 19:362–365.
14. Eisenberg E, Levanon EY (2013) Human housekeeping genes, revisited. *Trends Genet* 29:569–574.
15. McCarthy DJ, Campbell KR, Lun ATL, Wills QF (2017) Scater: Pre-processing, quality control, normalization and visualization of single-cell RNA-seq data in R. *Bioinformatics* 33:1179–1186.
16. Qiu X, et al. (2017) Reversed graph embedding resolves complex single-cell trajectories. *Nat Methods* 14:979–982.
17. Yang L, et al. (2017) A single-cell transcriptomic analysis reveals precise pathways and regulatory mechanisms underlying hepatoblast differentiation. *Hepatology* 66: 1387–1401.
18. Müsch A (2018) From a common progenitor to distinct liver epithelial phenotypes. *Curr Opin Cel Biol* 54:18–23.
19. Yan L, et al. (2013) Single-cell RNA-seq profiling of human preimplantation embryos and embryonic stem cells. *Nat Struct Mol Biol* 20:1131–1139.
20. Xue Z, et al. (2013) Genetic programs in human and mouse early embryos revealed by single-cell RNA sequencing. *Nature* 500:593–597.
21. Blakeley P, et al. (2015) Defining the three cell lineages of the human blastocyst by single-cell RNA-seq. *Development* 142:3613.
22. Goolam M, et al. (2016) Heterogeneity in oct4 and sox2 targets biases cell fate in 4-cell mouse embryos. *Cell* 165:61–74.
23. Biase F, Cao X, Zhong S (2014) Cell fate inclination within 2-cell and 4-cell mouse embryos revealed by single-cell RNA sequencing. *Genome Res* 24:1787–1796.
24. Stirparo GG, et al. (2018) Integrated analysis of single-cell embryo data yields a unified transcriptome signature for the human preimplantation epiblast. *Development* 145:dev158501.
25. Nathan H, Martinsson P-G, Tropp JA (2011) Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Rev* 53:217–288.
26. Erichson NB, Voronin S, Brunton SL, Kutz JN (2016) Randomized matrix decompositions using R. arXiv:1608.02148.
27. Kharchenko PV, Silberstein L, Scadden DT (2014) Bayesian approach to single-cell differential expression analysis. *Nat Methods* 11:740–742.
28. Suter DM, et al. (2011) Mammalian genes are transcribed with widely different bursting kinetics. *Science* 332:472–474.
29. Brennecke P, et al. (2013) Accounting for technical noise in single-cell RNA-seq experiments. *Nat Methods* 10:1093–1095.
30. Kim T, et al. (August 22, 2018) Impact of similarity metrics on single-cell RNA-seq data clustering. *Brief Bioinf*, 10.1093/bib/bby076.
31. Clauset A, Newman MEJ, Moore C (2004) Finding community structure in very large networks. *Phys Rev E* 70:066111.
32. Kolodziejczyk AA, et al. (2015) Single cell RNA-sequencing of pluripotent states unlocks modular transcriptional variation. *Cell Stem Cell* 17:471–485.
33. Nguyen QH, et al. (2018) Profiling human breast epithelial cells using single cell RNA sequencing identifies cell diversity. *Nat Commun* 9:2028.

34. Dong J, et al. (2018) Single-cell RNA-seq analysis unveils a prevalent epithelial/mesenchymal hybrid state during mouse organogenesis. *Genome Biol* 19:31.
35. Su X, et al. (2017) Single-cell RNA-seq analysis reveals dynamic trajectories during mouse liver development. *BMC Genomics* 18:946.
36. Camp JG, et al. (2017) Multilineage communication regulates human liver bud development from pluripotency. *Nature* 546:533–538.
37. Hanchate NK, et al. (2015) Single-cell transcriptomics reveals receptor transformations during olfactory neurogenesis. *Science* 350:1251–1255.
38. Tan L, Li Q, Xie XS (2015) Olfactory sensory neurons transiently express multiple olfactory receptors during development. *Mol Syst Biol* 11:844.
39. Lawlor N, et al. (2017) Single-cell transcriptomes identify human islet cell signatures and reveal cell-type–specific expression changes in type 2 diabetes. *Genome Res* 27:208–222.
40. Segerstolpe Å, et al. (2016) Single-cell transcriptome profiling of human pancreatic islets in health and type 2 diabetes. *Cell Metab* 24:593–607.
41. Muraro MJ, et al. (2016) A single-cell transcriptome atlas of the human pancreas. *Cell Syst* 3:385–394.
42. Baron M, et al. (2016) A single-cell transcriptomic map of the human and mouse pancreas reveals inter-and intra-cell population structure. *Cell Syst* 3:346–360.
43. Xin Y, et al. (2016) RNA sequencing of single human islet cells reveals type 2 diabetes genes. *Cell Metab* 24:608–615.
44. Wang YJ, et al. (2016) Single cell transcriptomics of the human endocrine pancreas. *Diabetes* 65:3028–3038.
45. Dobin A, et al. (2013) STAR: Ultrafast universal RNA-seq aligner. *Bioinformatics* 29: 15–21.
46. Li H, et al. (2009) The sequence alignment/map format and samtools. *Bioinformatics* 25:2078–2079.
47. Anders S, Theodor Pyl P, Huber W (2015) Htseq—A python framework to work with high-throughput sequencing data. *Bioinformatics* 31:166–169.
48. Tian L, et al. (2018) scRNA-seq mixology: Towards better benchmarking of single cell RNA-seq protocols and analysis methods. bioRxiv, p 433102.
49. Fan X, et al. (2015) Single-cell RNA-seq transcriptome analysis of linear and circular RNAs in mouse preimplantation embryos. *Genome Biol* 16:148.
50. Cole MB, et al. (2017) Performance assessment and selection of normalization procedures for single-cell RNA-seq. bioRxiv, p 235382.
51. Buttner M, Miao Z, Wolf A, Teichmann SA, Theis FJ (2017) Assessment of batch-correction methods for scRNA-seq data with a new test metric. bioRxiv, p 200345.
52. Gandolfo LC, Speed TP (2018) Rle plots: Visualizing unwanted variation in high dimensional data. *PLoS One* 13:e0191629.
53. Risso D, Ngai J, Speed TP, Dudoit S (2014) Normalization of RNA-seq data using factor analysis of control genes or samples. *Nat Biotechnol* 32:896–902.
54. Trapnell C, et al. (2014) The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nat Biotechnol* 32: 381–386.
55. Ritchie ME, et al. (2015) Limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Res* 43:e47–e47.
56. Meyer D, Dimitriadou E, Hornik K, Weingessel A, LeischF (2019) e1071: Misc Functions of the Department of Statistics, Probability, R package, version 1.7-0.1. Available at https://cran.r-project.org/web/packages/e1071/index.html. Accessed April 12, 2019.