nature methods

ARTICLES

https://doi.org/10.1038/s41592-018-0229-2

Deep generative modeling for single-cell transcriptomics

Romain Lopez[1], Jeffrey Regier[1], Michael B. Cole[2], Michael I. Jordan[1,3] and Nir Yosef[1,4,5*]

Single-cell transcriptome measurements can reveal unexplored biological diversity, but they suffer from technical noise and bias that must be modeled to account for the resulting uncertainty in downstream analyses. Here we introduce single-cell variational inference (scVI), a ready-to-use scalable framework for the probabilistic representation and analysis of gene expression in single cells (https://github.com/YosefLab/scVI). scVI uses stochastic optimization and deep neural networks to aggregate information across similar cells and genes and to approximate the distributions that underlie observed expression values, while accounting for batch effects and limited sensitivity. We used scVI for a range of fundamental analysis tasks including batch correction, visualization, clustering, and differential expression, and achieved high accuracy for each task.

Single-cell RNA sequencing (scRNA-seq) is a powerful tool that is beginning to make important contributions to diverse research areas such as development[1], autoimmunity[2], and cancer[3]. The interpretation of scRNA-seq data remains challenging, however, as it is confounded by nuisance factors such as limited[4] and variable[5] sensitivity, batch effects[6], and transcriptional noise[7]. Several recent studies modeled scRNA-seq bias and uncertainty by fitting a probabilistic model for each gene measurement in each cell, which represents the data in a lower and potentially less noisy dimension[8–10]. Once these models have been fit, they can be used for various tasks such as clustering[11], imputation[12], and differential expression analysis[13].

Although these methods have provided new insights into the biological variation between cells, they assume that a generalized linear model can be used to accurately map onto a low-dimensional manifold underlying the data, which is not necessarily justified. Also, different models are currently used for different tasks, whereas the application of a single distributional model to a range of downstream tasks would help to ensure consistency and interpretability. Finally, most existing methods cannot be applied to more than tens of thousands of cells, but recent datasets include hundreds of thousands of cells or more[14].

To address these limitations, we developed scVI, a fully probabilistic approach for the normalization and analysis of scRNA-seq data. scVI is based on a hierarchical Bayesian model[15] with conditional distributions specified by deep neural networks, which can be trained very efficiently even for very large datasets. The transcriptome of each cell is encoded through a nonlinear transformation into a low-dimensional latent vector of normal random variables. This latent representation is then decoded by another nonlinear transformation to generate a posterior estimate of the distributional parameters of each gene in each cell. The transformation assumes a zero-inflated negative binomial distribution, which accounts for the observed overdispersion and limited sensitivity[10,16,17].

Several recent papers have also demonstrated the utility of neural networks for embedding scRNA-seq datasets in a scalable manner[18–21]. scVI stands out from these as the only method that explicitly models the two key nuisance factors in scRNA-seq data (library size[8,22] and batch effects[10,23]) and the only readily available solution for a range of analysis tasks using the same generative model (Methods, Supplementary Note 1, Supplementary Table 1). To demonstrate its flexibility, we carried out batch removal, normalization, dimensionality reduction, clustering, and differential expression. We show here that for each of these tasks, scVI compared favorably to current state-of-the-art methods.

Results

The scVI model. We modeled the observed expression x_{ng} of each gene g in each cell n as a sample drawn from a zero-inflated negative binomial (ZINB) distribution $p(x_{ng} \mid z_n, s_n, \ell_n)$ conditioned on the batch annotation s_n of each cell (if available), as well as two additional, unobserved random variables[10,16,17] (Methods). The first variable, ℓ_n, is a one-dimensional Gaussian that represents nuisance variation due to differences in capture efficiency and sequencing depth, and serves as a cell-specific scaling factor. The second variable, z_n, is a low-dimensional vector of Gaussians (set here to ten dimensions; Supplementary Fig. 1) representing the remaining variation, which should better reflect biological differences between cells[24]. We used it to represent each cell as a point in a low-dimensional latent space that served for visualization and clustering. In the scVI model, a neural network maps the latent variables to the parameters of the ZINB distribution (Fig. 1a, neural networks 5 and 6). This mapping goes through intermediate values ρ_g^n, which provide a batch-corrected, normalized estimate of the percentage of transcripts in each cell n that originate from each gene g. We used these estimates for differential expression analysis and its scaled version (multiplying ρ_g^n by the estimated library size ℓ_n) for imputation. We derived an approximation for the posterior distribution of the latent variables $q(z_n, \log \ell_n | x_n, s_n)$ by training another neural network using variational inference and a scalable stochastic optimization procedure[25–27] (Fig. 1a, neural networks 1–4).

Model evaluation. We evaluated scVI along with a set of benchmark methods for probabilistic modeling and imputation of scRNA-seq data using a collection of published datasets spanning a range of technical and biological characteristics (Supplementary Table 2

[1]Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, Berkeley, CA, USA. [2]Department of Physics, University of California, Berkeley, Berkeley, CA, USA. [3]Department of Statistics, University of California, Berkeley, Berkeley, CA, USA. [4]Ragon Institute of MGH, MIT, and Harvard, Cambridge, MA, USA. [5]Chan Zuckerberg BioHub, San Francisco, CA, USA. *e-mail: niryosef@berkeley.edu

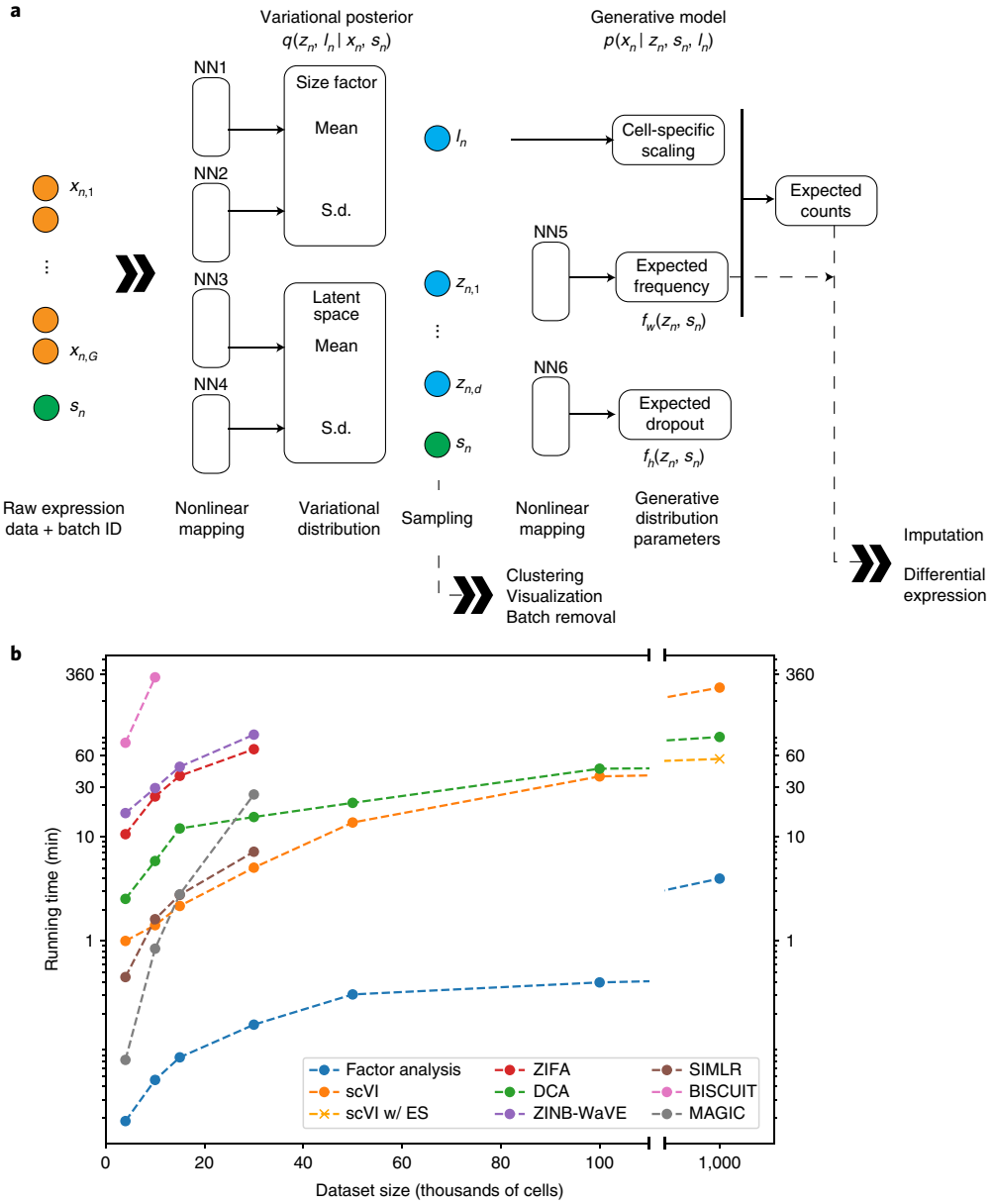

Fig. 1 | Overview of scVI. Given a gene expression matrix with batch annotations as input, scVI learns a nonlinear embedding of the cells that can be used for multiple analysis tasks. **a**, The neural networks used to compute the embedding and the distribution of gene expression. NN, neural network. f_w and f_h are functional representations of NN5 and NN6, respectively. **b**, Running times for fitting models on the BRAIN-LARGE data with a set of 720 genes and increasing input sizes subsampled randomly from the complete dataset. Algorithms were tested on a machine with one eight-core Intel i7-6820HQ CPU addressing 32 GB RAM, and one NVIDIA Tesla K80 (GK210GL) GPU addressing 24 GB RAM. Basic matrix factorization with FA acted as a control. For the 1-million-cell dataset, we report the results of scVI with and without early stopping (ES).

and Methods). To assess the scalability of training, we randomly subsampled a dataset of 1.3 million mouse brain cells[28] (BRAIN-LARGE). To facilitate comparison to state-of-the-art algorithms for probabilistic modeling and dimensionality reduction of single-cell data[8–12], which may be less scalable, we limited this analysis to the 720 genes with the largest s.d. across all cells (Fig. 1b). We found that most methods were capable of processing up to 50,000 cells before running out of memory (using 32 GB RAM). In contrast, scVI was generally faster and scaled to 1 million cells, thanks to its reliance on a fixed number of cells at each iteration of iterative stochastic optimization (Methods). We observed similar scalability with DCA[20], a denoising autoencoder that also uses stochastic optimization. Notably, as the dataset size approached 1 million cells, fewer training iterations (or epochs) were needed, and thus heuristics for stopping the learning process may save time. Indeed, standard scVI, which uses a fixed number of epochs, was slower than DCA, which uses the stopping heuristic by default, but scVI's early-stopping option greatly enhanced its speed (it trains in under 1 h) without affecting data fit (Supplementary Fig. 2).

Next, we evaluated the extent to which the methods fit the data by assessing their ability to accurately impute missing values. On five datasets of different sizes (BRAIN-LARGE[28], CORTEX[29], PBMC[30], RETINA[31], and HEMATO[32]; 3–27,000 cells; Supplementary Table 2), we set 9% of nonzero entries (chosen randomly (Supplementary Figs. 3 and 4) or with a preference for low values (Supplementary Figs. 5 and 6)) to zero and tested the ability of each method to

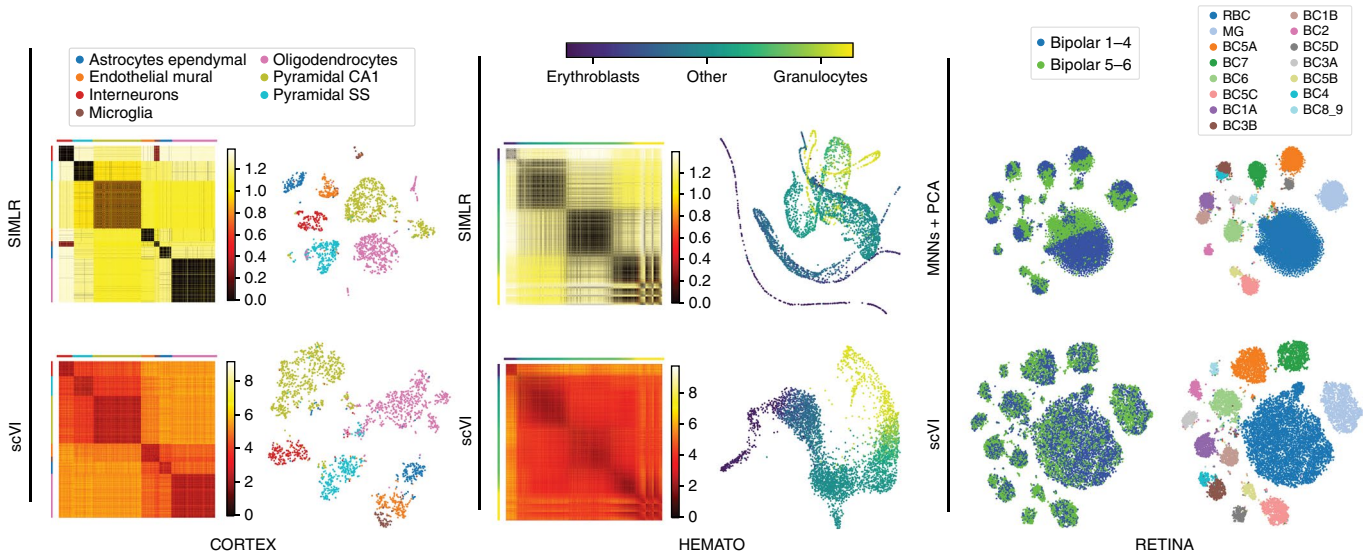

Fig. 2 | Biological signal is retained in the scVI latent space. scVI was applied to three datasets (CORTEX, $n = 3{,}005$ cells; HEMATO, $n = 4{,}016$ cells; and RETINA, $n = 27{,}499$ cells). CORTEX and HEMATO showed distance matrices in the latent space and 2D cell embeddings for scVI and SIMLR. Distance matrix scales are in relative units from low to high similarity over the range of values in the entire matrix; cells are grouped using labels provided in the original studies. CORTEX cell subsets were ordered by hierarchical clustering as in the original study. The embedding plot layout was determined by *t*-distributed stochastic neighbor embedding (t-SNE) (CORTEX) or a five-nearest-neighbors graph visualized with a Fruchterman–Reingold force-directed algorithm (HEMATO) (see Supplementary Fig. 10d for original SIMLR embedding). The color-coding is the same for embeddings and distance matrices. For RETINA, scVI is compared with principal component analysis followed by the mutual nearest neighbors method. t-SNE on the latent space provides embeddings. Left, cells are color-coded by batch. Right, cells are color-coded by subpopulation annotations from the original study[31].

recover the values. In most cases, methods based on a ZINB distribution—namely, scVI, DCA, and ZINB-WaVE[10] (when it scales to the dataset size)—performed better than methods that use alternative distributions[8,12] (e.g., log normal in ZIFA[9]), thus supporting the suitability of ZINB for current scRNA-seq datasets. In one important exception, scVI was outperformed by MAGIC[12] (which imputes by means of propagation in a cell–cell similarity graph) on the HEMATO hematopoietic differentiation dataset, which includes fewer cells (4,016) than genes (7,397). In such cases, scVI is expected to underfit the data, potentially leading to worse imputation accuracy. However, restriction of the analysis to the top 700 variable genes improved imputation (Supplementary Fig. 3c).

As an additional evaluation of model fit, we tested the likelihood of data that were held out during training, and obtained results in agreement with those of the synthetic dropout test (Supplementary Fig. 7, Supplementary Table 3). Furthermore, scVI, like ZIFA and factor analysis (FA), can also be used to generate unseen data via sampling of the latent space. As evidence of the validity of this procedure, we sampled from the posterior distribution given the perturbed training data, and observed that the samples were largely consistent with the unperturbed data (Supplementary Fig. 8).

Capturing biological structure in latent space. We next evaluated the extent to which the latent space inferred by scVI reflects biological variability between cells. One way to assess this is to rely on prior stratification of the cells into biologically meaningful subpopulations, which is normally done by unsupervised clustering followed by manual inspection and annotation[29,30]. We evaluated accuracy with respect to these stratifications (available for the CORTEX and PBMC datasets) by applying *k*-means clustering on the latent space and testing for overlap with the annotated subpopulations (using the same *k* as in the annotated data), or by comparing the proximity of cells in the same subpopulation to the proximity of cells from different subpopulations (Methods). A dataset that included single-cell protein measurements in addition to mRNA (CBMC)[33] served as an alternative benchmark, by allowing us to evaluate the extent to which the similarity between cells in the mRNA latent space resembled their similarity at the protein level (Methods).

In these tests, scVI grouped cells that were from the same annotated subpopulation or that expressed similar proteins, and it compared favorably to other methods that aim to infer a biologically meaningful latent space (ZIFA, ZINB-WaVE, DCA, and FA; Supplementary Fig. 9). Notably, a simpler version of scVI that does not explicitly model library size did not perform as well as the standard scVI, thus supporting our modeling choice.

Next, we benchmarked scVI with SIMLR[11], a method that couples clustering with learning of a cell–cell similarity matrix and a respective low-dimensional (latent) representation. SIMLR outperformed scVI by providing a tighter representation of the computationally annotated subpopulations. This result was expected, as SIMLR explicitly aims to produce a tight representation in a target number of clusters; however, a possible consequence of this action is that SIMLR may not capture higher-resolution structural properties of the cell–cell similarity map. Indeed, in the protein-versus-mRNA test, scVI and DCA performed best, albeit by a small margin (Supplementary Fig. 9c). scVI also more accurately captured hierarchical structure among cell subsets, such as was reported for cortical cells (CORTEX);[29] cells from related subpopulations tended to be closer to each other in scVI's latent space (Supplementary Fig. 9e–g). Another important case is when variation is continuous rather than discrete, as reported for differentiating hematopoietic cells (HEMATO)[32]. SIMLR identified several discrete clusters and did not reflect the continuous nature of this system as well as scVI or PCA did (Fig. 2, Supplementary Fig. 10). Finally, the data may be almost entirely dominated by noise and lack structure. On a noisy dataset that we generated by sampling at random from a vector of ZINB distributions, SIMLR erroneously reported 11 distinct clusters, which were not perceived by other methods (Supplementary Fig. 11). Altogether, these results suggest that the latent space of

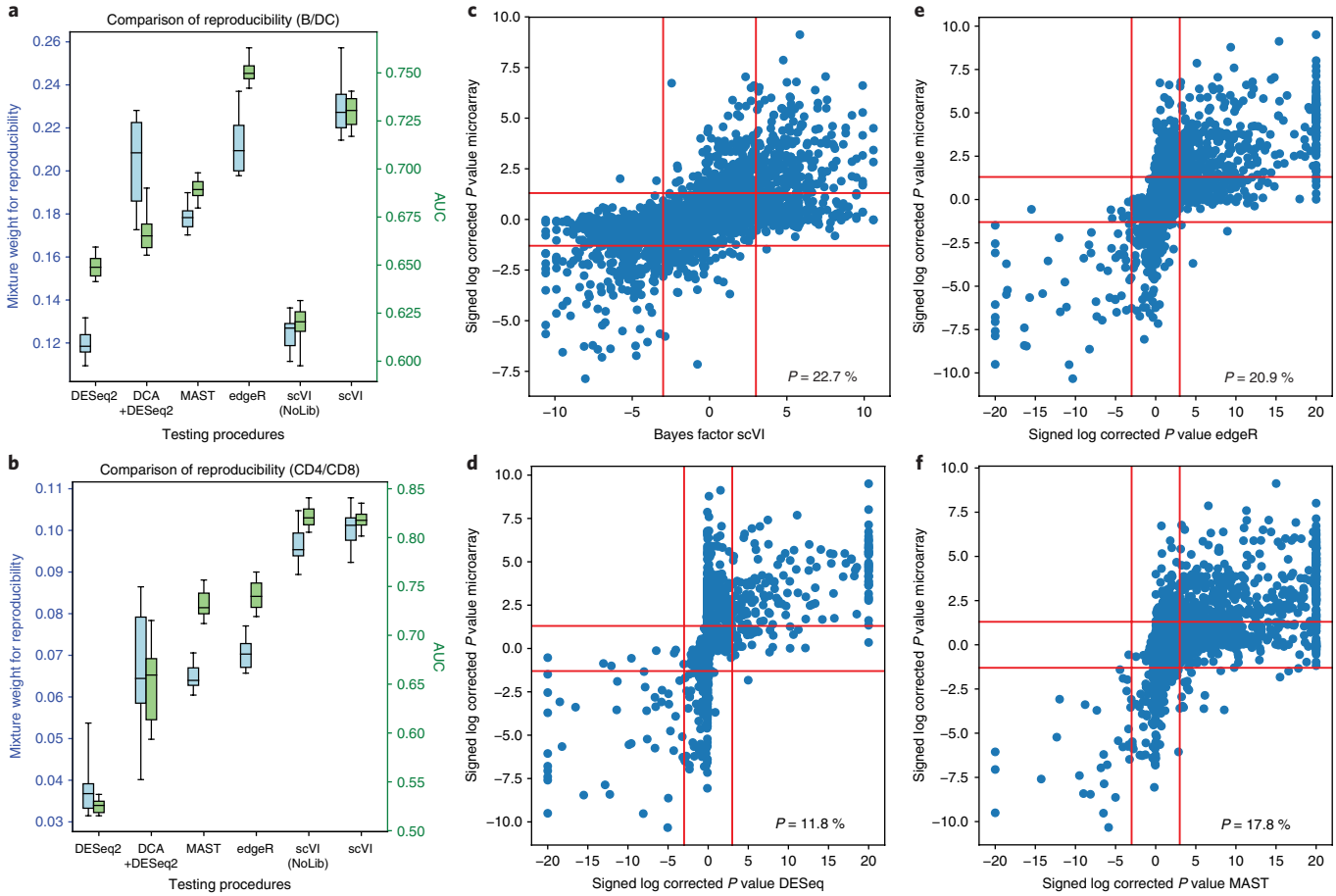

Fig. 3 | Benchmarking of differential expression analysis. Performance was evaluated on the PBMC dataset ($n = 12{,}039$ cells) on the basis of consistency with published bulk data. **a**,**b**, Comparison of B cells and dendritic cells (**a**) and of $CD4^+$ and $CD8^+$ T cells (**b**) evaluated for consistency with the IDR[41] framework (blue) and using AUROC (green). scVI (NoLib) refers to a simpler version of scVI that does not include the cell-specific scaling factor. The range of values was derived from subsampling of 100 cells from each cluster $n = 20$ times to determine robustness. Box plots indicate the median (center lines), interquantile range (hinges), and 5th to 95th percentiles (whiskers). **c**–**f**, Significance levels of differential expression between B cells and dendritic cells. Points represent individual genes ($n = 3{,}346$). Bayes factors or BH-corrected *P* values on scRNA-seq data are compared with bulk microarray-based BH-corrected *P* values. Horizontal lines denote the significance threshold of 0.05 for corrected *P* values. Vertical lines denote the significance threshold for the Bayes factor of scVI (**c**) or 0.05 for corrected *P* values for DESeq2 (**d**), edgeR (**e**), and MAST (**f**). We also report the median mixture weight for reproducibility *p* (higher values are better).

scVI is flexible and describes the data well, even when the data do not fit in a simple structure of discrete cell states.

Accounting for technical variability. scVI provides a parametric distribution designed to decouple biological signal from the effects of sample-level categorical nuisance factors such as batch annotations and variation in sequencing depth. To evaluate the capacity of scVI to correct batch effects, we used a mouse retinal bipolar neuron dataset consisting of two batches (RETINA). We defined an entropy measure to evaluate the mixing of cells from different batches in any local neighborhood of the latent space (abstracted using a *k*-nearest-neighbor graph; Methods). In this dataset, scVI aligned the batches considerably better than ComBat[34] (which uses linear models and empirical Bayes shrinkage) and a recent method based on matching of mutual nearest neighbors[35], while still maintaining a tight representation of preannotated subpopulations (Fig. 2, Supplementary Figs. 9d and 12). Algorithms that do not account for batch effects in their models provided poor mixing of batches, as expected. Specifically, although SIMLR and DCA were capable of clustering the cells well within each batch, the respective clusters from each batch remained largely separated. We obtained similar results when we applied a simplified version of scVI with no batch variable, thus supporting our modeling choice.

Turning to confounding due to variation in sequencing depth, we found, as expected, that in relatively homogeneous populations the library size factor inferred by scVI (ℓ_n) strongly correlated with the observed depth per cell (for example, in a subpopulation of peripheral blood mononuclear cells (PBMCs); Supplementary Fig. 13a). A related technical issue is low sensitivity due to limited mRNA capture efficiency and (to a lesser extent) sequencing depth, which exacerbates the number of zero entries and can distort similarity among homogeneous cells. We found that most zero entries could be explained by the negative binomial component (Supplementary Fig. 14a,b) rather than the 'inflation' of additional unexplained Bernoulli-distributed zeros. Consistently, the occurrence of zero entries largely agreed with a process of random sampling of genes from each cell, in a manner proportional to their expected frequency (as inferred in the matrix ρ of our model, which is proportional to the negative binomial mean) and with no additional bias (Supplementary Fig. 13b and Supplementary Note 2). Indeed, we found that zero probabilities from the negative binomial distribution correlated more with cell-specific quality factors related

to library size (e.g., number of reads per unique molecular identifier), whereas zero probabilities from the Bernoulli correlated more with quality factors indicative of alignment errors (using subpopulations of BRAIN-SMALL cortical cells or PBMCs; Supplementary Figs. 13c,d and 14c,d), possibly because of contamination or mRNA degradation. Taken together, these results corroborate the idea that most zeros, at least in the datasets explored here, can be explained by low (or zero) 'biological' abundance of the respective transcript, exacerbated by limited sampling.

Differential expression. With its probabilistic representation of the data, scVI provides a natural way of performing various types of hypothesis testing, while intrinsically controlling for nuisance factors. In the case of differential expression between two sets of cells, one can use the model to approximate the posterior probability of the alternative hypotheses (genes are different) and that of the null hypotheses through repeated sampling from the variational distribution, thus obtaining a low variance estimate of their ratio (i.e., Bayes factor[36,37]; Methods).

To evaluate scVI in comparison with other differential expression methods[13,17,20,38], we used a dataset of 12,039 PBMCs from a healthy human donor (PBMC) and undertook comparisons between B cell and dendritic cell clusters, and between $CD4^+$ and $CD8^+$ T cell clusters. Bulk-level comparative analysis of similar cell subsets served as a gold standard[39,40]. For evaluation, we first defined genes as true positives (Benjamini–Hochberg (BH)-adjusted P value < 0.05 in bulk data) and then calculated the area under the receiver operating characteristic curve (AUROC) on the basis of the Bayes factor for scVI or BH-corrected P value for the other methods. Because the definition of true positives requires a somewhat arbitrary threshold, we also used a second score that evaluated the reproducibility of gene ranking (bulk reference versus single cell, considering all genes), using the irreproducible discovery rate (IDR)[41]. scVI had the highest AUROC in the T cell comparison, whereas edgeR outperformed scVI by a smaller margin in the comparison of B cells versus dendritic cells. scVI performed best with respect to IDR in both comparisons (Fig. 3, Supplementary Fig. 15a–e). We noted that the use of DCA followed by DESeq2 constituted a solid improvement over the direct application of DESeq2, which was designed for bulk data, thus supporting the need for single-cell-adapted models. Furthermore, a simpler variant of scVI that does not include the library size factor performed extremely poorly in the comparison of B cells versus dendritic cells, thus supporting the usefulness of explicit inclusion of library size normalization in the model.

Discussion

scVI was designed to address an important need in the rapidly evolving field of single-cell transcriptomics—namely, to account for measurement uncertainty and bias in tertiary analysis tasks through a common, scalable statistical model. As a result, it provides a computationally efficient and 'all-inclusive' tool that couples low-dimensional probabilistic representation of gene expression data with downstream analysis capabilities, comparing favorably to state-of-the-art methods in each of a range of tasks, including batch-effect correction, imputation, clustering, and differential expression.

scVI takes raw count data as input and includes an effective normalization procedure that is integrated into its model. First, it learns a cell-specific scaling factor as a hidden variable, with the objective of maximizing the likelihood of the data[8,10,22], which is more justifiable than a posteriori correction of the observed counts[5]. Second, scVI explicitly accounts for batch annotations, via a mild assumption of conditional independence. We demonstrated that both of these components are essential for the method's performance.

The scVI deep learning architecture is built on several canonical building blocks such as nonlinearities, regularization, and mean-field approximation to the posterior[25] (Methods). The exploration of other, possibly better, architectures[42] and procedures for parameter and hyperparameter tuning[43] might in some instances provide a better model fit and more suitable approximate inference. Notably, because our procedure has a random component and optimizes a nonconvex objective function, it might give alternative results with different initializations. To address this, we demonstrated the stability of scVI in terms of its objective function, as well as imputation and clustering (Supplementary Fig. 1). A related issue is that if there are few observations (cells) for each gene, the prior and the inductive bias of the neural network might keep scVI from fitting the data closely. Indeed, gene prefiltering may be warranted in cases where there are fewer cells than genes. A complementary approach would make use of techniques such as Bayesian shrinkage[17] or regularization and second-order optimization[10]. However, we were able to show that for a range of datasets of varying sizes, scVI fit the data well and captured relevant biological diversity between cells.

Because it provides a general probabilistic representation of gene expression, scVI could enable other forms of scRNA-seq analysis not explored in this study, such as lineage inference[1] and cell-state annotation[7,44]. Furthermore, because it requires only the latent space and the model specification (which both have a low memory footprint) to generate any data point (cell × gene) of interest, scVI can be used as an effective baseline for scalable and interactive visualization tools[45–47]. Finally, scVI can be extended to merge multiple datasets from a given tissue while integrating prior biological annotations of cell types. We therefore expect this work to be of immediate interest, especially where dataset harmonization needs to be scalable and conducive to various forms of downstream analysis[14].

Online content

Any methods, additional references, Nature Research reporting summaries, source data, statements of data availability and associated accession codes are available at https://doi.org/10.1038/s41592-018-0229-2.

Received: 30 March 2018; Accepted: 26 October 2018;
Published online: 30 November 2018

References

1. Semrau, S. et al. Dynamics of lineage commitment revealed by single-cell transcriptomics of differentiating embryonic stem cells. *Nat. Commun.* **8**, 1096 (2017).
2. Gaublomme, J. T. et al. Single-cell genomics unveils critical regulators of Th17 cell pathogenicity. *Cell* **163**, 1400–1412 (2015).
3. Patel, A. P. et al. Single-cell RNA-seq highlights intratumoral heterogeneity in primary glioblastoma. *Science* **344**, 1396–1401 (2014).
4. Kharchenko, P. V., Silberstein, L. & Scadden, D. T. Bayesian approach to single-cell differential expression analysis. *Nat. Methods* **11**, 740–742 (2014).
5. Vallejos, C. A., Risso, D., Scialdone, A., Dudoit, S. & Marioni, J. C. Normalizing single-cell RNA sequencing data: challenges and opportunities. *Nat. Methods* **14**, 565–571 (2017).
6. Shaham, U. et al. Removal of batch effects using distribution-matching residual networks. *Bioinformatics* **33**, 2539–2546 (2017).
7. Wagner, A., Regev, A. & Yosef, N. Revealing the vectors of cellular identity with single-cell genomics. *Nat. Biotechnol.* **34**, 1145–1160 (2016).
8. Prabhakaran, S., Azizi, E., Carr, A. & Pe'er, D. Dirichlet process mixture model for correcting technical variation in single-cell gene expression data. *PMLR* **48**, 1070–1079 (2016).
9. Pierson, E. & Yau, C. ZIFA: dimensionality reduction for zero-inflated single-cell gene expression analysis. *Genome. Biol.* **16**, 241 (2015).
10. 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. *Nat. Commun.* **9**, 284 (2018).
11. Wang, B., Zhu, J., Pierson, E., Ramazzotti, D. & Batzoglou, S. Visualization and analysis of single-cell RNA-seq data by kernel-based similarity learning. *Nat. Methods* **14**, 414–416 (2017).
12. van Dijk, D. et al. MAGIC: a diffusion-based imputation method reveals gene-gene interactions in single-cell RNA-sequencing data. *bioRxiv* Preprint at https://www.biorxiv.org/content/early/2017/02/25/111591 (2017).

13. Finak, G. et al. MAST: a flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell RNA sequencing data. *Genome. Biol.* **16**, 278 (2015).
14. Regev, A. et al. The Human Cell Atlas. *eLife* **6**, e27041 (2017).
15. Gelman, A. & Hill, J. *Data Analysis Using Regression and Multilevel/Hierarchical Models* (Cambridge University Press, New York, 2007).
16. Grün, D., Kester, L. & van Oudenaarden, A. Validation of noise models for single-cell transcriptomics. *Nat. Methods* **11**, 637–640 (2014).
17. Love, M. I., Huber, W. & Anders, S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome. Biol.* **15**, 550 (2014).
18. Ding, J., Condon, A. & Shah, S. P. Interpretable dimensionality reduction of single cell transcriptome data with deep generative models. *Nat. Commun.* **9**, 2002 (2018).
19. Wang, D. & Gu, J. VASC: dimension reduction and visualization of single cell RNA sequencing data by deep variational autoencoder. *bioRxiv* Preprint at https://www.biorxiv.org/content/early/2017/10/06/199315 (2017).
20. Eraslan, G., Simon, L. M., Mircea, M., Mueller, N. S. & Theis, F. J. Single cell RNA-seq denoising using a deep count autoencoder. *bioRxiv* Preprint at https://www.biorxiv.org/content/early/2018/04/13/300681 (2018).
21. Grønbech, C. H. et al. scVAE: variational auto-encoders for single-cell gene expression data. *bioRxiv* Preprint at https://www.biorxiv.org/content/early/2018/05/16/318295 (2018).
22. Vallejos, C. A., Marioni, J. C. & Richardson, S. BASiCS: Bayesian analysis of single-cell sequencing data. *PLoS Comput. Biol.* **11**, e1004333 (2015).
23. Cole, M. B. et al. Performance assessment and selection of normalization procedures for single-cell RNA-seq. *bioRxiv* Preprint at https://www.biorxiv.org/content/early/2018/05/18/235382 (2017).
24. Louizos, C., Swersky, K., Li, Y., Welling, M. & Zemel, R. The variational fair autoencoder. Oral presentation at the International Conference on Learning Representations, San Juan, Puerto Rico, 2–4 May 2016.
25. Kingma, D. P. & Welling, M. Auto-encoding variational Bayes. Oral presentation at the International Conference on Learning Representations, Banff, Alberta, Canada, 14–16 April 2014.
26. Blei, D. M., Kucukelbir, A. & McAuliffe, J. D. Variational inference: a review for statisticians. *J. Am. Stat. Assoc.* **112**, 859–877 (2017).
27. Sønderby, C. K., Raiko, T., Maaløe, L., Sønderby, S. K. & Winther, O. Ladder variational autoencoders. In *Advances in Neural Information Processing Systems* (eds Lee, D. D. et al.) 3738–3746 (NIPS Foundation, La Jolla, CA, 2016).
28. 10x Genomics. Support: single cell gene expression datasets. *10x Genomics* https://support.10xgenomics.com/single-cell-gene-expression/datasets (2017).
29. Zeisel, A. et al. Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science* **347**, 1138–1142 (2015).
30. Zheng, G. X. Y. et al. Massively parallel digital transcriptional profiling of single cells. *Nat. Commun.* **8**, 14049 (2017).
31. Shekhar, K. et al. Comprehensive classification of retinal bipolar neurons by single-cell transcriptomics. *Cell* **166**, 1308–1323 (2016).
32. Tusi, B. K. et al. Population snapshots predict early haematopoietic and erythroid hierarchies. *Nature* **555**, 54–60 (2018).
33. Stoeckius, M. et al. Simultaneous epitope and transcriptome measurement in single cells. *Nat. Methods* **14**, 865–868 (2017).
34. Johnson, W. E., Li, C. & Rabinovic, A. Adjusting batch effects in microarray expression data using empirical Bayes methods. *Biostatistics* **8**, 118–127 (2007).
35. Haghverdi, L., Lun, A. T. L., Morgan, M. D. & Marioni, J. C. Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. *Nat. Biotechnol.* **36**, 421–427 (2018).
36. Kass, R. E. & Raftery, A. E. Bayes factors. *J. Am. Stat. Assoc.* **90**, 773–795 (1995).
37. Held, L. & Ott, M. On p-values and Bayes factors. *Annu. Rev. Stat. Appl.* **5**, 393–419 (2018).
38. Robinson, M. D., McCarthy, D. J. & Smyth, G. K. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics* **26**, 139–140 (2010).
39. Nakaya, H. I. et al. Systems biology of vaccination for seasonal influenza in humans. *Nat. Immunol.* **12**, 786–795 (2011).
40. Görgün, G., Holderried, T. A. W., Zahrieh, D., Neuberg, D. & Gribben, J. G. Chronic lymphocytic leukemia cells induce changes in gene expression of CD4 and CD8 T cells. *J. Clin. Invest.* **115**, 1797–1805 (2005).
41. Li, Q., Brown, J. B., Huang, H. & Bickel, P. J. Measuring reproducibility of high-throughput experiments. *Ann. Appl. Stat.* **5**, 1752–1779 (2011).
42. Zoph, B. & Le, Q. Neural architecture search with reinforcement learning. Oral presentation at the International Conference on Learning Representations, Toulon, France, 24–26 April 2017.
43. Bergstra, J. S., Bardenet, R., Bengio, Y. & Kégl, B. Algorithms for hyper-parameter optimization. In *Advances in Neural Information Processing Systems 24* (eds Shawe-Taylor, J. et al.) 2546–2554 (NIPS Foundation, La Jolla, CA, 2011).
44. Tanay, A. & Regev, A. Scaling single-cell genomics from phenomenology to mechanism. *Nature* **541**, 331–338 (2017).
45. DeTomaso, D. & Yosef, N. FastProject: a tool for low-dimensional analysis of single-cell RNA-Seq data. *BMC Bioinformatics* **17**, 315 (2016).
46. Fan, J. et al. Characterizing transcriptional heterogeneity through pathway and gene set overdispersion analysis. *Nat. Methods* **13**, 241–244 (2016).
47. Wolf, F. A., Angerer, P. & Theis, F. J. SCANPY: large-scale single-cell gene expression data analysis. *Genome. Biol.* **19**, 15 (2018).

Acknowledgements

N.Y. and R.L. were supported by NIH–NIAID (grant U19 AI090023). We thank A. Klein, S. Dudoit, and J. Listgarten for helpful discussions.

Author contributions

R.L., J.R., and N.Y. conceived the statistical model. R.L. developed the software. R.L. and M.B.C. applied the software to real data analysis. R.L., J.R., N.Y., and M.I.J. wrote the manuscript. N.Y. and M.I.J. supervised the work.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information is available for this paper at https://doi.org/10.1038/s41592-018-0229-2.

Reprints and permissions information is available at www.nature.com/reprints.

Correspondence and requests for materials should be addressed to N.Y.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Methods

The scVI probabilistic model. First, we present in more detail the generative process for scVI. Each expression value x_{ng} is drawn independently through the following process:

$$\begin{aligned} z_n &\sim \text{Normal}(0, I) \\ \ell_n &\sim \text{log normal}(\ell_\mu, \ell_\sigma^2) \\ \rho_n &= f_w(z_n, s_n) \\ w_{ng} &\sim \text{Gamma}(\rho_n^g, \theta) \\ y_{ng} &\sim \text{Poisson}(\ell_n w_{ng}) \\ h_{ng} &\sim \text{Bernoulli}(f_h^g(z_n, s_n)) \\ x_{ng} &= \begin{cases} y_{ng} & \text{if } h_{ng} = 0 \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

A standard multivariate normal prior for z is commonly used in variational autoencoders because it can be reparameterized in a differentiable way into any arbitrary multivariate Gaussian random variable[25], which turns out to be extremely convenient in the inference process.

B denotes the number of batches and $\ell_\mu, \ell_\sigma \in \mathbb{R}_+^B$ parameterize the prior for the scaling factor (on a log scale). ℓ_μ, ℓ_σ are set to be the empirical mean and variance of the log-library size per batch. We note that the random variable ℓ_n is not the log-library size (scaling the sampled observation) itself but a scaling factor that is expected to correlate strongly with log-library size (hence the choice of the parameters). The parameter $\theta \in \mathbb{R}_+^G$ denotes a gene-specific inverse dispersion, estimated via variational Bayesian inference.

f_w and f_h are neural networks that map the latent space and batch annotation back to the full dimension of all genes: $\mathbb{R}^d \times \{0,1\}^B \to \mathbb{R}^G$. We use superscript annotation (for example, $f_w^g(z_n, s_n)$) to refer to a single entry that corresponds to a specific gene g. Neural network f_w is constrained during the inference to encode the mean proportion of transcripts expressed across all genes by the use of a softmax activation at the last layer. Namely, for each cell n, the sum of $f_w^g(z_n, s_n)$ values over all genes g is 1. Neural network f_h encodes whether a particular entry has dropped out owing to technical effects[9,10]. These intermediate vectors can therefore be interpreted as expected frequencies. Importantly, we note that neural networks allow users to go beyond the generalized linear model framework and provide a more flexible model of gene expression. All neural networks use dropout regularization and batch normalization. Each network has one, two, or three fully connected layers, with 128 or 256 nodes each. Details are provided in Supplementary Table 2. The activation functions between two hidden layers are all ReLU. We use a standard link function to parameterize the distribution parameters (exponential, logarithmic, or softmax). Weights for some layers are shared between f_w and f_h.

Fast inference via stochastic optimization. The posterior distribution combines the prior knowledge with information acquired from the data matrix X. We cannot directly apply a Bayes rule to determine the posterior because the denominator (the marginal distribution, integrated over the latent variables) $p(x_n|s_n)$ is intractable. Inference over the whole graphical model is not needed. We can integrate out the latent variables w_{ng}, h_{ng}, and y_{ng} because $p(x_{ng}|z_n, \ell_n, s_n)$ has a closed-form density. Notably, the distribution $p(x_{ng}|z_n, s_n, \ell_n)$ is a ZINB[16] with mean $\ell_n \rho_n^g$, gene-specific dispersion θ^g and zero-inflation probability $f_h^g(z_n, s_n)$ (Supplementary Note 3). We discuss the numerical stability and parameterization of the ZINB distribution in Supplementary Note 4. Having simplified our model, we use variational inference[26] to approximate the posterior $p(z_n, \ell_n|x_n, s_n)$. Our variational distribution $q(z_n, \ell_n|x_n, s_n)$ is mean-field:

$$q(z_n, \ell_n|x_n, s_n) = q(z_n|x_n, s_n)\, q(\ell_n|x_n, s_n)$$

The variational distribution $q(z_n|x_n, s_n)$ is chosen to be Gaussian with a diagonal covariance matrix, mean, and covariance given by an encoder network applied to (x_n, s_n), as in ref. [25]. The variational distribution $q(\ell_n|x_n, s_n)$ is chosen to be log-normal with the scalar mean and variance also given by an encoder network applied to (x_n, s_n). The variational lower bound is

$$\begin{aligned} \log p(x|s) \geq\ & \mathbb{E}_{q(z,l|x,s)} \log p(x|z,l,s) \\ & - D_{\text{KL}}(q(z|x,s)\|p(z)) \\ & - D_{\text{KL}}(q(l|x,s)\|p(l)) \end{aligned} \tag{1}$$

In this objective function, the dispersion parameters θ_g for each gene are treated as global variables to be optimized in a variational Bayesian inference fashion.

To optimize the lower bound, we use the analytic expression for $p(x|z,l,s)$ and use analytic expressions for the Kullback–Leibler divergences. We use the reparameterization trick to compute low-variance Monte Carlo estimates of the expectations' gradients. Analytic closed-form for the Kullback–Leibler divergence and the reparameterization trick are possible only on certain distributions that multivariate Gaussians are a part of[25]. The reparameterization trick is a specific sampling scheme from the variational distribution, which makes our objective function stochastic. Remarkably, this sampling step coupled with neural network approximation to the posterior is what makes it possible to go beyond restrictive 'conditional conjugacy' properties often needed for sampling or variational inference. This allows us to efficiently carry out inference with arbitrary models, including those with conditional distributions specified by neural networks[25].

A second level of stochasticity comes from subsampling from the training set (possible because the cells are identically independently distributed when conditioned on the latent variables). We then have an online optimization procedure that can handle massive datasets, used by both scVI and other methods that exploit neural networks[18–21]. At each iteration, we focus only on a small subset of the data randomly sampled ($M = 128$ data points) and do not need to go through the entire dataset. Therefore, there is no need to store the entire dataset in memory. Because the number of genes is in practice limited to a few tens of thousands, these mini-batches of cells can be handled easily by a GPU. Now, our objective function is continuous and end-to-end differentiable, which allows us to use automatic differentiation operators.

Throughout the paper, we use Adam (a first-order stochastic optimizer) with $\varepsilon = 0.01$. As indicated in ref. [27], we use deterministic warm-up and batch normalization during learning to learn an expressive model. A complete list of hyperparameters is provided in Supplementary Table 2. The hyperparameters were chosen via a small grid search that maximized held-out log likelihood—a common practice for training deep generative models. One of the strengths of scVI is that there are only three dataset-specific hyperparameters to set (learning rate, number of layers, and layer width). We optimize the objective function until convergence—usually between 120 and 250 epochs, where each epoch is a complete pass through the dataset (we note that bigger datasets require fewer epochs). For the larger subset of the BRAIN-LARGE dataset, we also ran with the early-stopping criterion: the algorithm stopped after 12 consecutive epochs with no improvement on the validation loss.

Because the encoder network $q(z|x,s)$ might still produce output correlated with the bath s, one could use in principle a maximum mean discrepancy (MMD)-based penalty as in ref. [24] to correct the variational distribution. For this paper, however, we did not explicitly enforce the MMD penalty and simply retained the conditional independence property, which has been shown to be sufficiently efficient. This may be useful with other datasets, though it explicitly assumes that the exact same biological signal is present in the datasets.

Bayesian differential expression. For each gene g and pair of cells (z_a, z_b) with observed gene expression (x_a, x_b) and batch ID (s_a, s_b), we can formulate two mutually exclusive hypotheses:

$$\mathcal{H}_1^g := \mathbb{E}_s f_w^g(z_a, s) > \mathbb{E}_s f_w^g(z_b, s) \text{ versus } \mathcal{H}_2^g := \mathbb{E}_s f_w^g(z_a, s) \leq \mathbb{E}_s f_w^g(z_b, s)$$

where the expectation $\mathbb{E}_s$ is taken with the empirical frequencies. Notably, we propose a hypothesis testing that does not calibrate the data to one batch but will find genes that are consistently differentially expressed. Evaluation of which hypothesis is more probable amounts to evaluation of a Bayes factor[37] (Bayesian generalization of the P value). Its sign indicates which of $\mathcal{H}_1^g$ and $\mathcal{H}_2^g$ is more likely. Its magnitude is a significance level, and throughout the paper, we consider a Bayes factor as strong evidence in favor of a hypothesis if $|K| > 3$ (ref. [36]) (equivalent to an odds ratio of $\exp(3) \approx 20$).

$$K = \log_e \frac{p(\mathcal{H}_1^g|x_a, x_b)}{p(\mathcal{H}_2^g|x_a, x_b)}$$

where the posterior of these models can be approximated via the variational distribution

$$p(\mathcal{H}_1^g|x_a, x_b) \approx \sum_s \iint_{z_a\, z_b} p(f_w^g(z_x, s) \leq f_w^g(z_x, s))\, p(s)\, dq(z_a|x_a)\, dq(z_b|x_b)$$

where $p(s)$ designates the relative abundance of cells in batch s and all of the measures are low-dimensional, so we can use naive Monte Carlo to compute these integrals. We can then use a Bayes factor for the test.

Because we assume that all the cells are independent, we can average the Bayes factors across a large set of randomly sampled cell pairs, one from each subpopulation. The average factor will provide an estimate of whether cells from one subpopulation tend to express g at a higher frequency.

We demonstrated the robustness of our method by repeating the entire evaluation process and comparing the results (Fig. 3a,b). We also ensured that our Bayes factors were well calibrated by running the differential expression analysis across cells from the same cluster and making sure no genes reached the significance threshold (Supplementary Fig. 15f).

Modeling choices. In this section, we consider the extent to which each of a sequence of modeling choices in the design of scVI contributes to its performance. As a baseline approach, consider normalizing scRNA-seq data as in the literature[9]

and reducing the dimensionality of the data by using a variational autoencoder with a Gaussian prior and a Gaussian conditional probability.

One way to enhance a model is to change the Gaussian conditional probability to one of the many available count distributions, such as ZINB, negative binomial (NB), Poisson, and others. Recent work by Eraslan et al. using simulated data shows that when the dropout effect drives the signal-to-noise ratio to a less favorable regime, a denoising autoencoder with mean squared error (i.e., Gaussian conditional likelihood) cannot recover cell types from expression data, whereas an autoencoder with ZINB conditional likelihood can[20]. This result points to the importance of at least modeling the sparsity of the data and is in agreement with previous contributions[9,10].

The next question is which count distribution to use. In scVI we have chosen to use the ZINB, a choice motivated by published literature (for example, ref. [10]). First, the choice of negative binomial is common with RNA-seq data, as they are overdispersed[17]. Furthermore, under some assumptions this distribution captures the steady-state form of the canonical two-state promoter-activation model[16]. Finally, recent work by Grønbech et al[21]. proposes an analysis based on Bayesian model selection (held-out log-likelihood as in this paper). In that analysis, the NB and ZINB distributions stand out with similarly high scores. We demonstrate that the addition of a zero-inflation (Bernoulli) component is important for explaining a subset of the zero values in the data (Supplementary Fig. 14) and that it captures important aspects of technical variability that are not captured by the NB component (Supplementary Fig. 13).

To enhance the model further, we added terms to account for library size as a nuisance factor, which can be considered as a Bayesian approach to normalization as in refs [8,22]. We showed how this contributes to our model by increasing clustering scores and differential expression analysis accuracy on the PBMC dataset.

As a further enhancement, we designed the generative model to explain data from different experimental batches. This is not a trivial task, as a substantial covariate shift may exist between the observed transcript measurements. We showed how this modification to our model is crucial when dealing with batch effects in subsection on the RETINA dataset.

Datasets and preprocessing. Below we describe all of the datasets and the preprocessing steps used in the current work. We focused on relatively large datasets (3,000 cells or more) with unique molecular identifiers, thus providing enough information during training and avoiding the problem of overcounting due to amplification. An asterisk after the dataset name indicates that we used it as an auxiliary dataset; these datasets were used not for general benchmarking but rather to support specific points presented in the paper. The only case where we subsampled the data multiple times was that of the BRAIN-LARGE dataset. However, we simply used one instance of it to report all possible scores (further details are presented in Supplementary Table 2).

CORTEX. The Mouse Cortex Cells dataset from ref. [29] contains 3,005 mouse cortex cells and gold-standard labels for seven distinct cell types. Each cell type corresponds to a cluster to recover (Supplementary Table 4). We retained the top 558 genes ordered by variance as in ref. [8].

PBMC. We considered scRNA-seq data from two batches of PBMCs from a healthy donor (4,000 and 8,000 PBMCs, respectively)[30]. We derived quality control metrics using the cellrangerRkit R package (v. 1.1.0). Quality metrics were extracted from Cell Ranger throughout the molecule-specific information file. After filtering as in ref. [23], we extracted 12,039 cells with 10,310 sampled genes and generated biologically meaningful clusters with the software Seurat (Supplementary Table 5). We then filtered genes that we could not match with the bulk data used for differential expression, which resulted in $g = 3{,}346$.

BRAIN-LARGE. This dataset contains 1.3 million brain cells from 10x Genomics[28]. We randomly shuffled the data to get a subset of 1 million cells and ordered genes by variance to retain first 10,000 and then 720 sampled variable genes. This dataset was then sampled multiple times in cells for the runtime and goodness-of-fit analysis. We report imputation scores for the 10,000 cells and 720 gene samples only.

RETINA. After filtering according to the original pipeline, the dataset of bipolar cells from ref. [31] contained 27,499 cells and 13,166 genes from two batches. We used the cluster annotation from 15 cell types from the author. We also extracted their normalized data with ComBat and used it for benchmarking.

HEMATO. This dataset with continuous gene expression variations from hematopoietic progenitor cells[32] contains 4,016 cells and 7,397 genes. We removed the library *basal-bm1*, which was of poor quality, on the basis of the authors' recommendation. We used their population balance analysis result as a potential function for differentiation.

CBMC.* This dataset includes 8,617 cord blood mononuclear cells[33] profiled using 10×, along with 13 well-characterized mononuclear antibodies for each cell. We kept the top 600 genes by variance.

BRAIN-SMALL.* This dataset, which consists of 9,128 mouse brain cells profiled using 10x[28], was used as a complement to PBMC for our study of zero abundance and quality control metric correlation with our generative posterior parameters. We derived quality control metrics by using the cellrangerRkit R package (v. 1.1.0). Quality metrics were extracted from Cell Ranger throughout the molecule-specific information file. We kept the top 3,000 genes by variance. We used the clusters provided by Cell Ranger for the correlation analysis of zero probabilities.

Statistics. *Differential expression for bulk datasets.* Specifically, we assembled a set of genes that are differentially expressed between human B cells and dendritic cells (microarrays; $n = 10$ in each group[39]; GSE29618) and between CD4$^+$ and CD8$^+$ T cells (microarrays; $n = 12$ in each group[40]; GSE8835). For GSE29618, we first loaded bulk human expression array data using the GEOquery package, selecting all B cell and myeloid dendritic cell samples from the baseline ("Day0") time point. We retained all expression features described by exactly one gene symbol and regressed the expression of these expression measures on cell-type covariate (B cell versus myeloid dendritic cell) using lmFit linear modeling in limma. *P* values were derived from empirical Bayes moderated *t*-tests for differences between the two cell types, using eBayes in limma. We conducted an identical study on GSE8835 for the CD4$^+$ and CD8$^+$ T cell comparison. These *P* values were then corrected via the standard BH procedure.

Differential expression for scRNA-seq datasets. We used the packages as detailed above. The *P* values were then corrected via the standard BH procedure.

Capturing technical variability. We computed the average probability of zero from the NB distribution and from the Bernoulli across all genes for a particular cell. We tested for a correlation between these cell-specific zero probabilities and cell-specific quality control metrics by using a Pearson-correlation test.

Evaluation. We describe below how we computed the metrics used in the study. Further details of the algorithms used for benchmarking in this work are provided in Supplementary Note 5.

Log-likelihood on held-out data. We provide a multivariate metric of goodness of fit on the data in Supplementary Note 6.

Corrupting the datasets for imputation benchmarking. In this study we used two different approaches to measure the robustness of algorithms to noise in the data:

- Uniform zero introduction: we randomly selected 10% of the nonzero entries and multiplied the entry n with a Ber(0.9) random variable.
- Binomial data corruption: we randomly selected 10% of the matrix and replaced an entry n with a Bin(n, 0.2) random variable.

Accuracy of imputing missing data. As imputation is tantamount to replacing missing data by its mean conditioned on being observed, we used the median $\mathbb{L}_1$ distance between the original dataset and the imputed values for corrupted entries only. For MAGIC, we used the output of the associated software. For BISCUIT, we used the imputed counts. For ZIFA, we used the mean of the generative distribution conditioned on the nonzero event (mean of the factor analysis part) that we projected back into count space. For scVI and ZINB-WaVE, we used the mean of the NB distribution.

Silhouette width. The silhouette width requires either a similarity matrix or a latent space. We can define a silhouette score for each sample i with

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

where $a(i)$ is the average distance from i to all data points in the same cluster c_i, and $b(i)$ is the lowest average distance from i to all data points in the same cluster c among all clusters c. Clusters can be replaced with batches if one is estimating the silhouette width to assess batch effects[23].

Clustering metrics. The following metrics require clustering and not simply a similarity matrix. For these, we use k-means clustering on the given latent space of dimension 10 with $T = 200$ random initializations to achieve a stable score.

Adjusted Rand index. This index requires clustering. For most indexes,

$$\mathrm{ARI} = \frac{\sum_{ij}\binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right]\Big/\binom{n}{2}}{(1/2)\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \left[\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right]\Big/\binom{n}{2}}$$

where n_{ij}, a_i, and b_j are values from the contingency table.

Normalized mutual information.

$$\mathrm{NMI} = \frac{I(P;T)}{\sqrt{\mathbb{H}(P)\mathbb{H}(T)}}$$

where P,T designates empirical categorical distributions for the predicted and real clustering, I is the mutual entropy, and $\mathbb{H}$ is the Shannon entropy.

Entropy of batch mixing. Fix a similarity matrix for the cells and take U to be a uniform random variable on the population of cells. Take B_U as the empirical frequencies for the 50 nearest neighbors of cell U being a in batch b. Report the entropy of this categorical variable and average over $T = 100$ values of U.

Protein abundance/mRNA expression. Take the similarity matrix for the normalized protein abundance (centered log-ratio transformation; see ref. [33]). Compute a 100-nearest-neighbors graph. Fix a similarity matrix for the cells and compute a 100-nearest-neighbors graph. Report the Spearman correlation of the flattened matrices and the fold enrichment.

Let A be the set of edges in the protein nearest neighbors (NN) graph, B be the set of edges in the cell NN graph, and C be the entire set of possible edges. The fold enrichment is defined as

$$\frac{|A \cap B| \times |C|}{|A|\,|B|}$$

Differential expression metrics. We used 100 cells from each cluster. In scVI, we draw 200 samples from the variational posterior; subsampling ensures that our results are stable.

Area under the curve. We assign each gene a label of differentially expressed (DE) or non-DE on the basis of its P value from the reference data (genes with BH-corrected P values < 0.05 are positive, and the rest are negative); then we use these labels to compute the AUROC.

Irreproducible discovery rate. The IDR is computed with the corresponding R package. We adjust the prior for the mixture weight to be the fraction of genes detected in the microarray data.

Reporting Summary. Further information on research design is available in the Nature Research Reporting Summary linked to this article.

Software availability. An open-source software implementation of scVI is available on Github (https://github.com/YosefLab/scVI). All code for the reproduction of results and figures in this article has been deposited at https://zenodo.org/badge/latestdoi/125294792 and is included as Supplementary Software.

Data availability

All of the datasets analyzed in this paper are public and can be referenced at https://github.com/romain-lopez/scVI-reproducibility.

nature research

Corresponding author(s): Nir Yosef

Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Research policies, see Authors & Referees and the Editorial Policy Checklist.

Statistical parameters

When statistical analyses are reported, confirm that the following items are present in the relevant location (e.g. figure legend, table legend, main text, or Methods section).

n/a	Confirmed	
☐	☒	The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☐	☒	An indication of whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☐	☒	The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☐	☒	A description of all covariates tested
☐	☒	A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☐	☒	A full description of the statistics including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☐	☒	For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☐	☒	For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☐	☒	For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☐	☒	Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated
☐	☒	Clearly defined error bars *State explicitly what error bars represent (e.g. SD, SE, CI)*

Our web collection on statistics for biologists may be useful.

Software and code

Policy information about availability of computer code

Data collection	No software was used to collect the data.
Data analysis	https://github.com/romain-lopez/scVI-reproducibility (version 0.1) Python packages: scikit-learn v0.19.0 R packages: IDR v1.2 (cran)

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors/reviewers upon request. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Research guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

All of the datasets analyzed in this manuscript are public and referenced at https://github.com/romain-lopez/scVI-reproducibility

Field-specific reporting

Please select the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/authors/policies/ReportingSummary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	No experiments in study
Data exclusions	No experiments in study
Replication	No experiments in study
Randomization	No experiments in study
Blinding	No experiments in study

Reporting for specific materials, systems and methods

Materials & experimental systems

n/a	Involved in the study
☒	☐ Unique biological materials
☒	☐ Antibodies
☒	☐ Eukaryotic cell lines
☒	☐ Palaeontology
☒	☐ Animals and other organisms
☒	☐ Human research participants

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging