nature methods

ANALYSIS

https://doi.org/10.1038/s41592-018-0254-1

A test metric for assessing single-cell RNA-seq batch correction

Maren Büttner[1,6], Zhichao Miao[2,3,6], F. Alexander Wolf[1], Sarah A. Teichmann[2,3,4*] and Fabian J. Theis[1,5*]

Single-cell transcriptomics is a versatile tool for exploring heterogeneous cell populations, but as with all genomics experiments, batch effects can hamper data integration and interpretation. The success of batch-effect correction is often evaluated by visual inspection of low-dimensional embeddings, which are inherently imprecise. Here we present a user-friendly, robust and sensitive *k*-nearest-neighbor batch-effect test (kBET; https://github.com/theislab/kBET) for quantification of batch effects. We used kBET to assess commonly used batch-regression and normalization approaches, and to quantify the extent to which they remove batch effects while preserving biological variability. We also demonstrate the application of kBET to data from peripheral blood mononuclear cells (PBMCs) from healthy donors to distinguish cell-type-specific inter-individual variability from changes in relative proportions of cell populations. This has important implications for future data-integration efforts, central to projects such as the Human Cell Atlas.

The term "batch effect" is commonly used to describe technical variation that emerges when samples are handled in distinct batches. This situation usually occurs if one repeats an experiment with biologically equivalent cells, such as from different patients with the same disease, or technically equivalent cells, such as identically cultured cells sequenced on subsequent days (Fig. 1a). Biological and technical variation both contribute substantially to total variability in single-cell RNA-sequencing (scRNA-seq) data. In an experiment with a balanced design, biological and technical variation can be readily distinguished (Fig. 1b). In contrast, a confounded design groups cells of the same condition into the same sequencing runs, and thus separates biologically distinct cells into separate handling and sequencing experiments. This confounds biological and technical variability. If a balanced experimental design is not possible, for example, because of the chip design, an alternative strategy is to generate several technical replicates per biological condition[1].

Accounting for technical factors in an scRNA-seq dataset is a key step in the preprocessing workflow after cell and gene filtering[2] and affects the selection of potentially interesting genes. The choice of 'interesting' genes removes noise from the data but also defines the potential outcome of the data analysis. Furthermore, differences between replicates in scRNA-seq data can arise from different sequencing depths: fewer genes are detected at shallow sequencing depths[3–5].

Various methods have been proposed to remove or reduce cell-specific bias and batch effects in single-cell data while preserving biological variability, ranging from linear regression models such as ComBat[6] to nonlinear models such as Seurat's canonical correlation analysis (CCA)[7] and projection of mutual nearest neighbors (MNNs)[8]. In addition, differential test frameworks like MAST[4], DESeq2[9] and limma[10] include the batch effect as a covariate in the model design (Supplementary Table 1 provides an overview of single-cell normalization and batch-correction methods). We stress that differential testing with batch correction and the creation of a batch-corrected data matrix for downstream analyses such as clustering are distinct tasks in scRNA-seq data analysis; here we focus on the latter.

Given the wide variety of available strategies, we sought to determine which methods remove batch effects and preserve biological variation best. Current approaches to detect batch effects involve exploratory data analysis, that is, visual inspection of low-dimensional embeddings, such as principal component analysis (PCA). Data visualization is highly recommended as a first step and provides important insight on the distribution of data. However, the results are subjective and lack a quantitative measure for robust comparisons, especially if used to evaluate biases across many samples or methods. Thus, we recommend an additional quantitative step after exploratory data analysis.

Here we propose kBET as a method to quantify batch effects in scRNA-seq data. Intuitively, a replicated experiment is well mixed if a subset of neighboring samples (e.g., single-cell transcriptomic data points) has the same distribution of batch labels as the full dataset (Fig. 1c). In contrast, a repetition of the experiment with some bias is expected to yield a skewed distribution of batch labels across the dataset (Fig. 1d). kBET uses a χ^2-based test for random neighborhoods of fixed size to determine whether they are well mixed, followed by averaging of the binary test results to return an overall rejection rate. This result is easy to interpret: low rejection rates imply well-mixed replicates.

In this study, we applied kBET to the analysis of four mouse single-cell datasets from studies using microwell-plate-based and droplet-based methods (100–3,000 cells per batch) and assessed the performance and accuracy of 11 normalization and 7 batch-effect regression approaches (Fig. 1e). Batch correction based on log(counts + 1), log(counts per million (CPM) + 1) or scran pooling,

[1]Helmholtz Zentrum München–German Research Center for Environmental Health, Institute of Computational Biology, Neuherberg, Germany. [2]European Molecular Biology Laboratory, European Bioinformatics Institute (EMBL-EBI), Wellcome Genome Campus, Cambridge, UK. [3]Wellcome Sanger Institute, Wellcome Genome Campus, Cambridge, UK. [4]Department of Physics, Cavendish Laboratory, University of Cambridge, Cambridge, UK. [5]Department of Mathematics, Technische Universität München, Munich, Germany. [6]These authors contributed equally: Maren Büttner, Zhichao Miao. *e-mail: st9@sanger.ac.uk; fabian.theis@helmholtz-muenchen.de

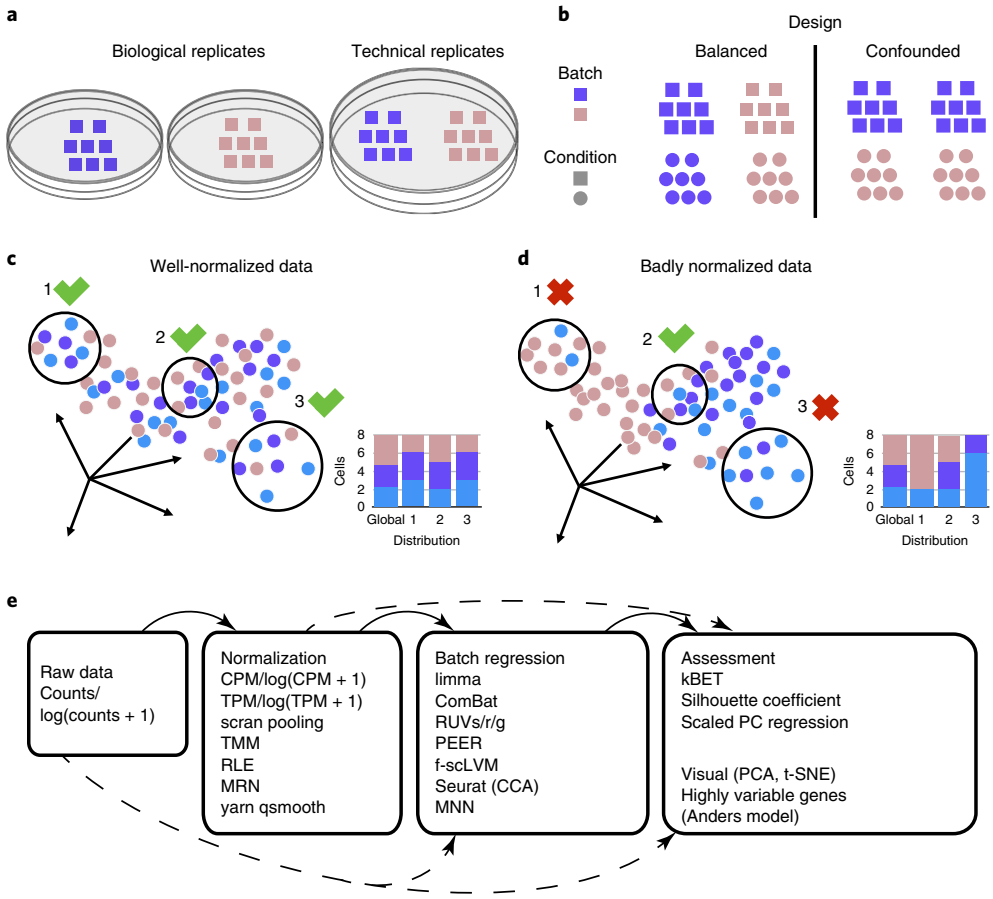

Fig. 1 | Batch types and the concept of kBET. Estimating the batch effect in single-cell RNA-seq data. **a**, Biological and technical replicates have different origins. Technical replicates are derived from the same biological samples (cell cultures in this case), whereas biological replicates are independent samples. **b**, Experimental designs. A balanced design allows one to separate technical and biological sources of variation, whereas a confounded design mixes the two. **c**,**d**, The kBET concept. **c**, In a dataset with replicates and no batch effects, the proportions of the batch labels in any neighborhood do not differ from the global distribution. **d**, In a replicated dataset with batch effects, data points from respective batches tend to cluster with their 'peers', and batch label proportions differ considerably between arbitrarily chosen neighborhoods. **e**, Overview of normalization and batch-regression methods, as well as assessment approaches.

together with ComBat or limma regression, reduced the batch effect while preserving biological structure across all datasets (Table 1). Finally, we explored the potential of kBET to assess the integration of separate studies, and determined that kBET also allows one to study inter-individual variability in complex human tissue data.

Results

kBET outperforms other batch-effect detection methods. We evaluated the performance of kBET on simulated data (500 samples ('cells') with 1,000 'genes' each) for which 1%, 10% or 20% of mean gene expression was varied in a second batch (Methods). With appropriate scaling, the expected mean expression remained unchanged. A second batch with 1% biased genes overlapped well with the first batch, yielding a low rejection rate (Fig. 2a). In contrast, a second batch with 20% biased genes separated from the first batch, so that samples were surrounded by samples from the same batch, thus yielding a high rejection rate (Fig. 2b).

Table 1 | Best overall normalization and batch-correction methods

Dataset	Klein et al.[12]	Kolodziejczyk et al.[14]			Mouse early embryo
		2i	a2i	LIF	
Sequencing technique	inDrop	Smart-seq2	Smart-seq		
Normalization	log(counts + 1)/scran pooling	log(CPM + 1)	scran pooling	TMM/log(CPM + 1)	log(counts + 1)
Batch correction	ComBat	ComBat	ComBat	limma/ComBat	ComBat

The ranking of batch-correction strategies is based on kBET, retained HVGs and FPRs for data from Klein et al.[12] and Kolodziejczyk et al.[14]. For mouse early embryonic development data integration, the ranking is based on both kBET and silhouette. TMM, trimmed mean of *M* values.

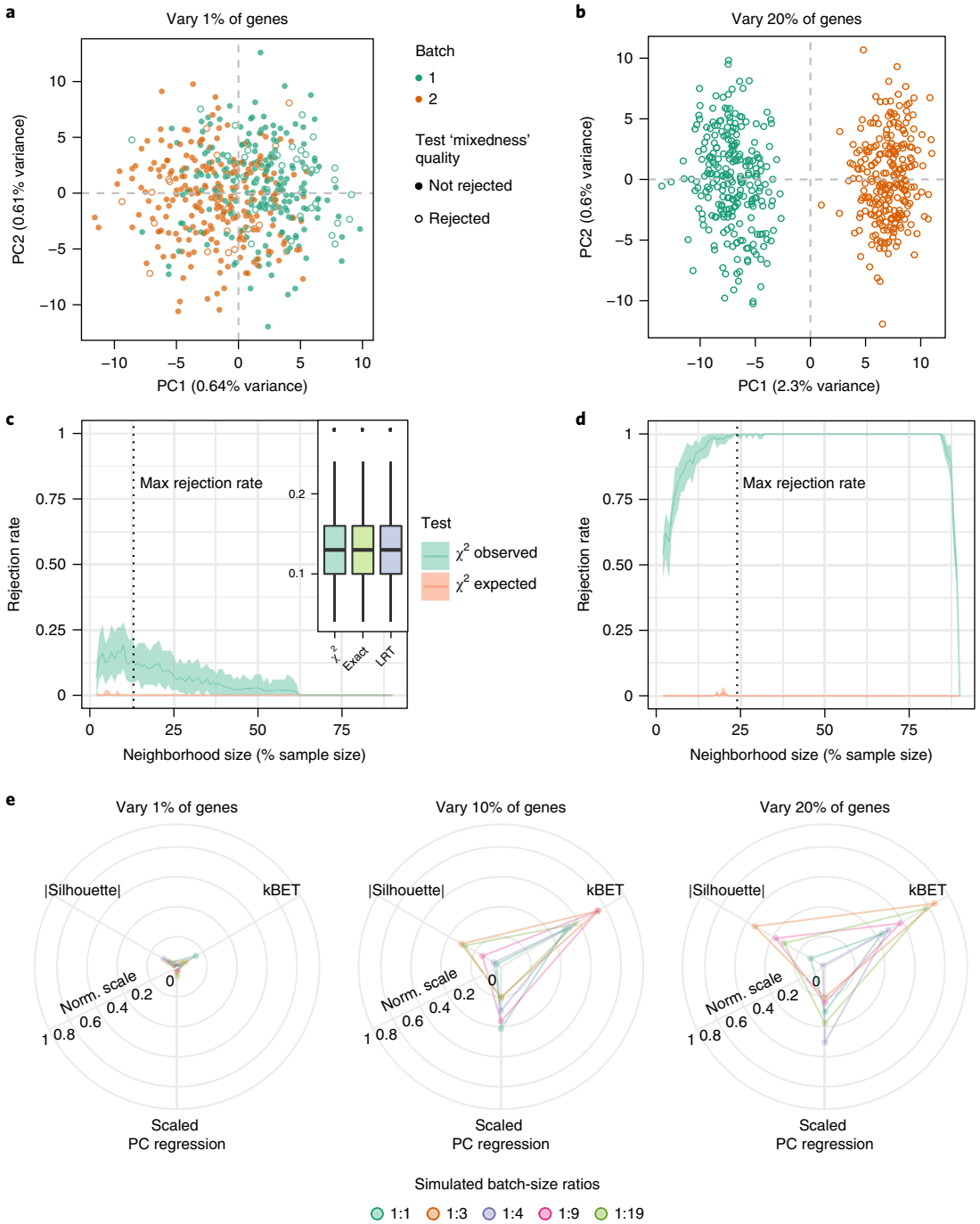

Fig. 2 | kBET is more responsive than other batch tests on simulated data. Simulation results for 1,000 genes and 500 cells. Batch effect is modeled as a fraction of shifted gene means in the second batch. **a**,**b**, PCA plots of log-scaled data for two equally sized batches with 1% (**a**) and 20% (**b**) varied mean gene expression levels (Methods). **c**,**d**, kBET mean rejection rates depend on neighborhood size for the data in **a** and **b**, respectively. The dashed vertical lines indicate the optimal neighborhood size for batch-effect detection, that is, where the rejection rate is maximal. Shaded areas represent the 95th percentile of $n = 100$ repeated kBET runs. In each run, the number of tested neighborhoods was 10% of the sample size (i.e., 50 cells). The inset in **c** shows a box plot of likelihood ratio test (LRT) results and exact test results for $n = 100$ runs. Center lines indicate means, lower and upper hinges correspond to the 25th and 75th percentiles, respectively, whiskers extend to 1.5 times the interquartile range, and individual data points represent outliers. **e**, Comparison of kBET to other batch-effect tests on a normalized scale: scaled variance explained by batch ("scaled PC regression"; FDR < 0.05) and absolute silhouette. We simulated several batch sizes to assess the effect of unequal batch sizes. kBET detected the degree of bias most effectively.

kBET uses a Pearson's χ^2-based test for random neighborhoods of fixed size k and averages the binary test result. Use of the likelihood ratio test or the exact multinomial test as the underlying hypothesis test (Methods, Supplementary Note 1 and ref. [11]) yielded very similar results (Fig. 2c (inset) and Supplementary Fig. 1). The neighborhood size k is an important factor for the hypothesis test in kBET. For smaller values of k, the rejection rate is lower in general[11]. As soon as the neighborhood size k for each test became larger than the size of a single batch, we observed a decrease in the rejection rate. This was due to the decreasing number of possible

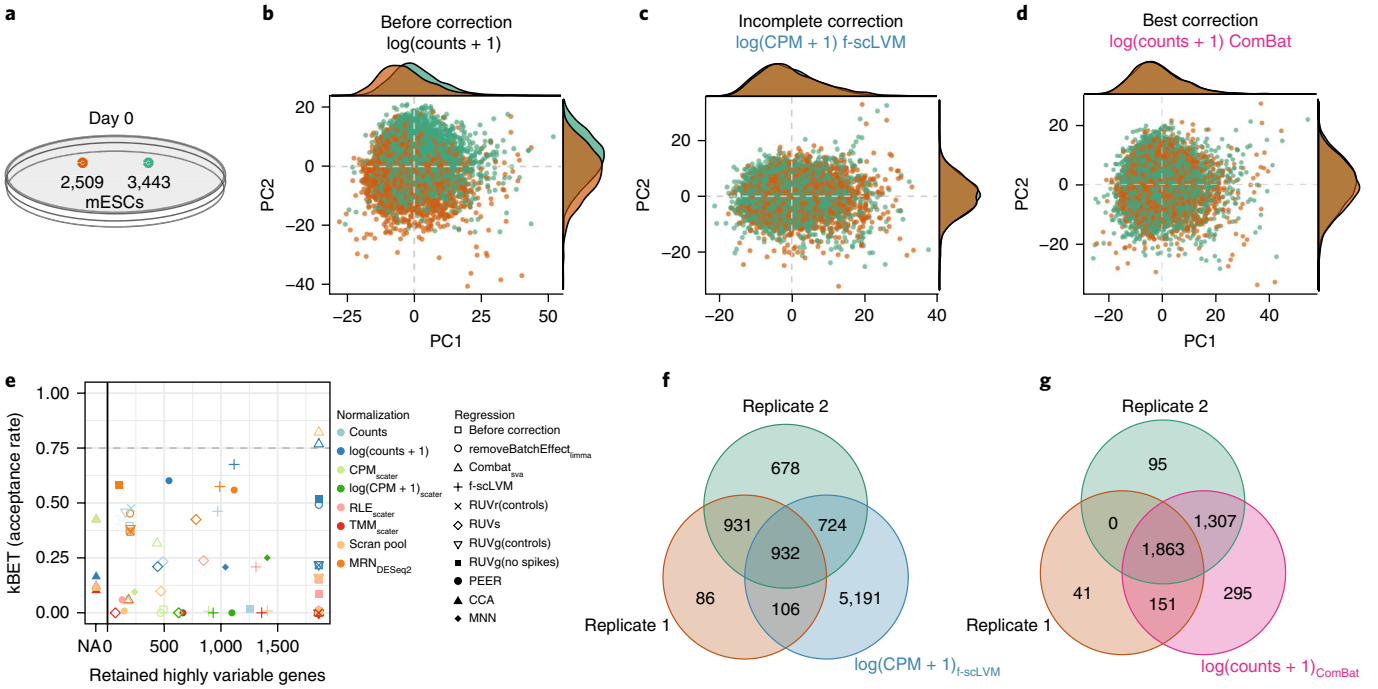

Fig. 3 | ComBat provides the best correction on mESC inDrop technical replicates. a, The inDrop protocol provides a large unique-molecular-identifier-count dataset with two technical replicates. **b**–**d**, PCA plots showing log-normalized counts (**b**), a biology-removing batch removal (f-scLVM on log-transformed CPM; **c**) and a biology-preserving batch removal (ComBat on log-transformed counts; **d**). Density plots on the axes show the frequency of replicates along the PCs. On the basis of visual inspection, the approaches in **c** and **d** appear to work equally well. **e**, Percentage of retained HVGs versus the mean acceptance rate (1 – rejection rate, from $n = 100$ kBET runs) for all combinations of normalizations and batch-regression approaches. Seurat's CCA alignment batch-corrects data only in a latent space generated by manifold learning, and thus we could not compute HVGs for it. **f**,**g**, HVGs per replicate before correction and for the whole dataset after batch correction. HVGs in each replicate are computed on log(counts + 1) values. f-scLVM (**f**) retained 932 HVGs but had a high false positive rate, whereas ComBat (**g**) captured all HVGs with a low false positive rate.

choices of batch labels; the 'local' batch-label distribution became more similar to the global batch-label distribution (Fig. 2c,d). For neighborhood sizes somewhere between exceedingly small and large, the average rejection rate became maximal. The maximum value indicated the presence of a batch effect (Supplementary Note 1), which we used for quantification.

We investigated kBET's ability to detect batch effects with alternative measures: the absolute average silhouette width ('silhouette') and the scaled variance of the top 50 principal components (PCs) that correlate significantly with the batch effect (false discovery rate (FDR) < 0.05, 'scaled PC regression'; Methods, Supplementary Note 2). All three measures operate on a normalized scale (0, no effect; 1, strong effect). As test cases, we varied an increasing fraction of gene expression means (Fig. 2e) and simulated different gene dropouts or added noise to the means (Supplementary Note 3 and Supplementary Figs. 2 and 3). Further, we simulated different batch sizes ranging from equal size (1:1) to substantial size imbalance (1:19). kBET's rejection rate increased most in response to the degree of bias compared with the silhouette: silhouettes showed little difference between 10% and 20% varied genes, where kBET clearly discriminated the effect. Scaled PC regression increased with the degree of batch effect, but also returned a significant result when only 1% of genes were varied. kBET performed well when only a few data points were biased by batch, as it still revealed a substantial bias in size-imbalanced batches. Overall, kBET was the most sensitive and robust measure of batch effects in this comparison.

kBET accurately captures batch effects. Batch effects originate from different sources, as is evident in comparisons of technical replicates. We investigated the mouse embryonic stem cell (mESC) LIF cultures of Klein et al.[12], which were generated via the inDrop protocol. The authors provided two technical replicates in samples of the day 0 culture (Fig. 3a), which offered an ideal case for batch-correction assessment. Prior to batch correction, we visualized the data, and observed a clear inter-batch difference as the shift of the technical replicates in the top two PCs (Fig. 3b). We compared all combinations of normalization and batch-correction strategies, and here we illustrate major performance differences in f-scLVM-corrected log(CPM + 1) values (Fig. 3c) and ComBat-corrected log(counts + 1) values (Fig. 3d) in terms of the top two PCs. However, in addition to the top 2 PCs, we found that the next 13 PCs also had a significant correlation with the batch covariate in the f-scLVM case (FDR < 0.05). Thus, the batch effect was not corrected but became unnoticeable with visualization. For ComBat-corrected log(counts + 1) values, none of the PCs correlated significantly with the batch covariate. kBET revealed that ComBat applied to log(counts + 1) or scran normalization removed batch effects best (*y*-axis in Fig. 3e; "acceptance rate" is the reverted 'rejection rate'), in contrast to the incomplete batch-correction performance of f-scLVM. The PCA plot shows only the batch effect of the first two PCs, whereas kBET effectively quantified subtler batch effects.

Distinguishing batch effects from biological variability. Preservation of biological heterogeneity is the second challenge of batch correction; without it the optimal batch correction would remove all variance, setting each sample to the same constant. We assessed biologically relevant heterogeneity by evaluating highly variable genes (HVGs)[13] before and after correction. Before correction, we considered only HVGs present in all replicates separately; this is the conservative, batch-free set of HVGs that we compared

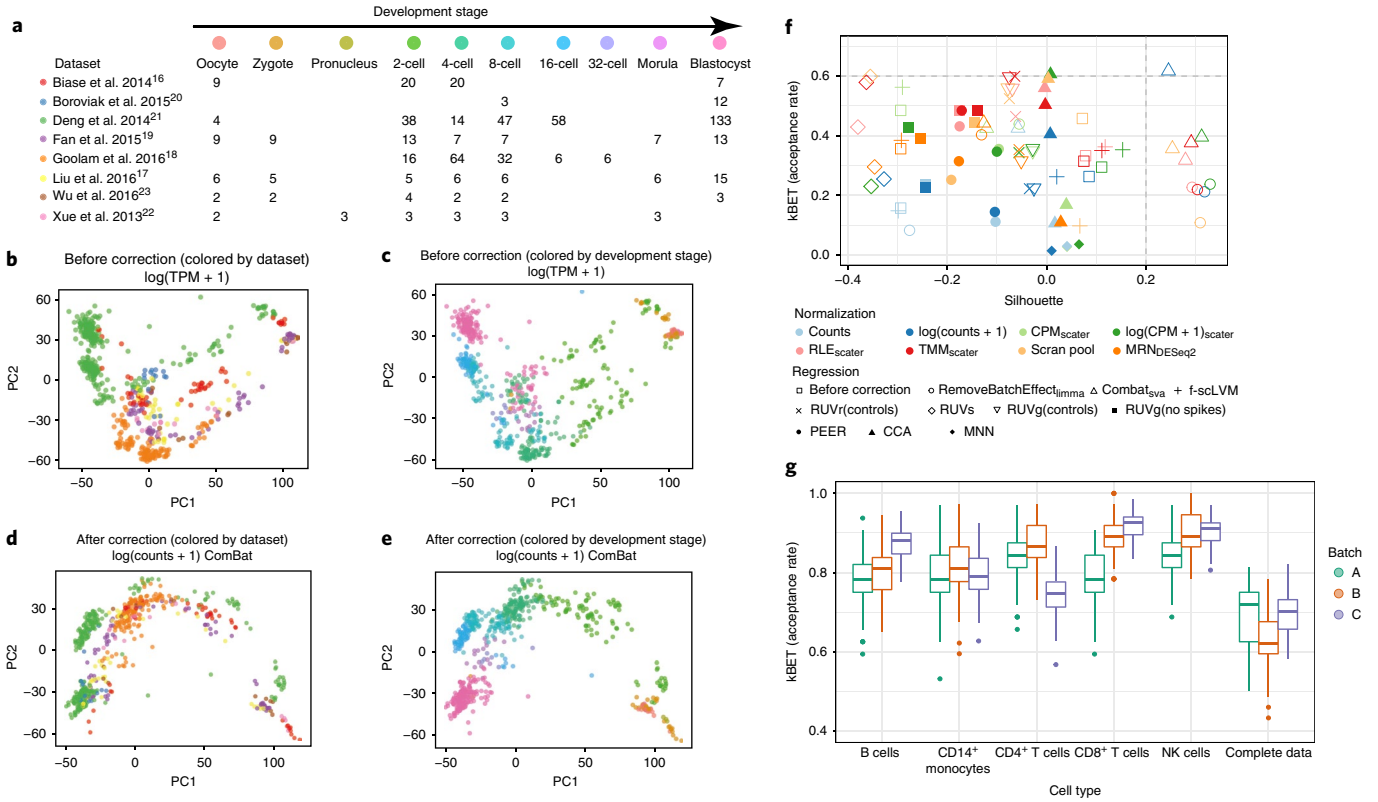

Fig. 4 | kBET assesses data-integration quality and inter-individual variability. a, Overview of eight early-mouse-development datasets. **b**,**c**, PCA plots of log(TPM + 1) normalized expression data, color-coded by dataset (**b**) and by developmental stage (**c**). Cells of the same developmental stage largely failed to cluster together across studies. **d**,**e**, PCA plots of log(counts + 1) normalized expression data after batch correction by limma, color-coded by dataset (**d**) and by developmental stage (**e**). Cells of the same developmental stage were better aligned after correction. **f**, Silhouette of embryonic development versus average kBET acceptance rate (weighted per developmental stage; $n = 100$ kBET runs per cell type) reveals that ComBat applied to log(counts + 1) data provides good mixing of cells at the same developmental stage from different studies. A high kBET acceptance rate and high silhouette indicate the best separation of developmental stages. **g**, PBMC data from eight unrelated individuals processed in three experiments (batches) on a Chromium 10x Genomics device, with donor cell identity assigned with demuxlet[26]. kBET mean acceptance rates (1 − rejection rate, from $n = 100$ kBET runs) are shown for each experiment, with individual donor identity used as a batch variable. For fair comparison, we equalized the number of cells in each batch by downsampling abundant cell types and the complete dataset. kBET yielded lower acceptance rates when we used the complete dataset (and neglected variation in cell-type frequencies), whereas acceptance rates were higher for the respective cell types. Center lines indicate means, lower and upper hinges correspond to the 25th and 75th percentiles, respectively, whiskers extend to 1.5 times the interquartile range, and individual data points represent outliers. TPM, transcripts per million.

to the set of HVGs after batch correction (HVG_{corr}). In total, we evaluated the fraction of retained HVGs after correction (Methods and Fig. 3e–g).

To complement the concept of retained HVGs, batch correction should not introduce additional variability in the data. Thus, the difference between sets of HVGs before and after correction is a proxy for false discoveries, which we used to compute a false positive rate (FPR; Methods). Here the two technical replicates shared 1,863 batch-free HVGs ($HVG_{batch\text{-}free}$), and more than 700 HVGs resided in either of the replicates (Fig. 3f,g).

After correction by f-scLVM, we retained half of $HVG_{batch\text{-}free}$ and discovered more than 5,000 HVGs in the whole dataset (Fig. 3f and Supplementary Fig. 4a,b), which explains f-scLVM's minimal kBET acceptance rate (Fig. 3e). When we computed the FPR on the basis of log(CPM + 1) normalized data, we found an FPR of 27% (Supplementary Fig. 4c). We obtained the best result for the combination of log(counts + 1) and ComBat (Fig. 3d): all $HVG_{batch\text{-}free}$ were kept after batch correction, and only 295 HVGs were caused by batch correction (8% FPR; Fig. 3g).

In conclusion, batch correction may confound observations massively and mask the biological signal completely. The current 'best' batch-correction strategies still leave part of the batch effect in the data (Fig. 3e–g). This explains the increase in the total amount of HVGs after correction (Supplementary Fig. 4b) and in the FPR (Supplementary Fig. 4c). Both silhouette and PC regression showed little discrimination between most correction strategies (Supplementary Fig. 4d,e), whereas kBET resolved them in detail (Fig. 3e and Supplementary Fig. 4d,e).

kBET guides best practices for plate-based scRNA-seq. Next, we examined mESCs cultured in three different media (2i, a2i and LIF)[14] and sequenced with the Smart-seq2/C1 protocol (Supplementary Fig. 5a). These datasets are rather similar in terms of heterogeneity, but the biological origin of the heterogeneity is different in each culture condition (compare with ref. [14]). We obtained well-mixed data for all datasets with log(CPM + 1) normalization and batch correction with ComBat (Supplementary Fig. 5b,c). Nevertheless, we observed performance differences across culture conditions that were independent of the number of batches (Supplementary Note 4).

Beyond replicates: dataset integration across studies. With the explosion of scRNA-seq data in recent years[15], there is a need for a

comprehensive strategy of data integration. It is more challenging to correct batch effects between studies than it is to do so within the same study, especially if cell types vary between studies. Here we benchmarked batch-correction performance on eight Smart-seq-based datasets[16–23] that profile mouse development from oocyte to blastocyst (Fig. 4a and Methods).

We remapped the reads to the same reference transcriptome with Salmon[24] to reduce quantification biases[25]. Notably, even different versions of Salmon resulted in different degrees of batch effect (Supplementary Note 5). Batch effects before correction were quite obvious even in PCA (Fig. 4b,c): data from Biase et al.[16] and Deng et al.[21] deviated substantially from the other data in our analysis (average acceptance rate of 16%). Consequently, cells are more likely to cluster by study than by embryonic stage. Also, clustering by study is partly explained by library size (Supplementary Fig. 6). Nevertheless, we achieved acceptable batch-effect correction. We obtained the best results with ComBat on log(counts + 1) values (Fig. 4d,e), with an average acceptance rate of 62% (Supplementary Table 2).

A meaningful integration maintains the correct trajectory of developmental stages, while the same cell types from different studies mingle. Thus, we assessed the batch effect of each developmental stage on the basis of averaged kBET results (a high acceptance rate implied good mixing) and monitored the developmental progression by silhouette (higher silhouette values reflected good separation of stages) (Fig. 4f). Before correction, the developmental stages separated weakly (silhouette of 0.08 for log(counts + 1)), and correction with linear methods such as limma and ComBat yielded distinct clustering by stage. Only ComBat achieved good mixing of study batches. Notably, PC1 corresponded to the real developmental time of the cells.

Although Seurat's CCA alignment was among the top-performing methods and yielded the second best kBET result for log(CPM + 1) data, a silhouette of approximately 0 indicated overcorrection. MNN yielded a low acceptance rate and improved clustering by cell types only for counts, possibly because of low sample numbers.

This example illustrates how batch-effect correction tools play a key role in data integration and provide effective separation of the biological signal from complex technical variations. For future data-integration efforts with more complex data structures and less prior knowledge about cell types, the community needs more sophisticated batch-correction methods that model nested batch structures and several batch variables.

kBET detects inter-individual variability in PBMC data. To estimate pure biological variability with kBET, we studied a pooled dataset of human PBMCs from eight unrelated individuals for which donor identity was reconstructed for each cell with demuxlet[26]. Pooling removes technical variation between individuals. Clustering and *t*-distributed stochastic neighbor embedding visualization revealed several cell types (Supplementary Fig. 7a) and significant variation in cell-type frequency between individuals (Supplementary Fig. 7b). Note that all samples were distributed across three independent experiments (batch A, individuals 1–4; batch B, individuals 5–8; and batch C, all individuals), and cell type frequencies were very similar between batches, thus excluding sampling bias.

We applied kBET to estimate inter-individual variability in all these experimental batches. kBET detected considerable variation within a cell type even after accounting for frequency shifts (Fig. 4g and Supplementary Fig. 7c). We found acceptance rates of ~0.75–0.9 for each cell type, versus 0.62–0.72 for complete data. Thus variation in aggregates such as bulk RNA-seq data is driven not only by single-cell expression differences, but also by variation in cell population sizes. kBET offers a sensitive and unbiased way to estimate inter-individual variability among cells of the same type.

Discussion

Batch effects in scRNA-seq data can have severe effects on downstream analysis if not properly accounted for. Moreover, they have a substantial random noise component that stems mostly from technical experimental factors. kBET introduces a nonlinear measure for scRNA-seq batch effects, which we used to evaluate batch-correction methods. In the simplest possible case—technical replicates that were otherwise homogeneous—ComBat corrected the data and preserved the underlying biological properties (Supplementary Table 2). On biological replicates with greater batch-to-batch variability, such as two independent cultures of the same cell type, ComBat again performed well, because of its regularization for low sample numbers. A study in which ComBat was used on complex tissue data reported similar results[27].

Many methods such as ComBat and RUV[28] were designed to correct bulk expression data but can be applied to scRNA-seq data. Although scRNA-seq data reflect cell-to-cell variability, they are much sparser because of stochastic gene expression and dropouts, which is not accounted for by batch-effect correction approaches for bulk data. A mere mean shift and variance stabilization would not take into account a batch-to-batch difference that solely addressed dropout rates (Supplementary Note 6). Also, dropout and cellular detection rates[4,5] are closely correlated to library size (Supplementary Fig. 8 and ref. [29]). The single-cell-specific approaches model stochastic expression and dropout explicitly[30] or implicitly[31]. As zeros in gene expression comprise both biological and technical variation, several approaches aim to impute dropout to retain biological information[32–35].

For complex tissue data, CCA[7] and MNN[8] provide generalized, nonlinear modeling approaches to align similar populations. In contrast to ComBat, both methods are independent of variations in population density[7,8]. Although CCA and MNN did not outperform linear methods in the small-scale examples we tested, they have potential in future large-scale data integration. Moreover, with thousands of measured cells per dataset, optimal memory usage and efficient implementation (Supplementary Note 7 and ref. [36]) will be as important as accurate correction for confounders.

kBET is a powerful tool for comparing batch-effect correction schemes, as it allows the study of high-dimensional data without prior assumptions regarding statistical properties. Analysis tasks such as clustering into cell types and ordering of cells by pseudotime[37] rely on batch-effect-corrected data. kBET's assumption of equivalent and interchangeable batches is simple, but the translation into balanced experimental design is challenging. For complex experimental setups such as time series, collecting and sequencing all cells at all time points together is the only way to prevent confounding with both technical and biological variation between samples. The demuxlet[26] approach allows inter-individual variability to be assessed quantitatively without technical confounding, and kBET's heterogeneity statistics are a useful measure for biological variability across individuals.

In the worst case, batch-effect correction may fail completely if data lack a minimum level of quality. By quantifying batch effects with kBET before and after correction, we were able to detect poor-quality correction and poor-quality data. On the basis of the kBET result showing that overall variation is driven by differences in cluster proportions, we would prefer to sequence more cells from fewer donors for complex samples (in contrast to prior statements[38]). We expect this discussion to have serious repercussions for decisions regarding experimental design in emerging single-cell expression atlases such as the Human Cell Atlas[39] and the Mouse Cell Atlas[40].

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/s41592-018-0254-1.

Received: 6 October 2017; Accepted: 31 October 2018;
Published online: 20 December 2018

References

1. Tung, P.-Y. et al. Batch effects and the effective design of single-cell gene expression studies. *Sci. Rep.* **7**, 39921 (2017).
2. Lun, A. T. L., McCarthy, D. J. & Marioni, J. C. A step-by-step workflow for low-level analysis of single-cell RNA-seq data with Bioconductor. *F1000Res.* **5**, 2122 (2016).
3. Heimberg, G., Bhatnagar, R., El-Samad, H. & Thomson, M. Low dimensionality in gene expression data enables the accurate extraction of transcriptional programs from shallow sequencing. *Cell Syst.* **2**, 239–250 (2016).
4. 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).
5. Hicks, S. C., Townes, F. W., Teng, M. & Irizarry, R. A. Missing data and technical variability in single-cell RNA-sequencing experiments. *Biostatistics* **19**, 562–578 (2018).
6. Johnson, W. E., Li, C. & Rabinovic, A. Adjusting batch effects in microarray expression data using empirical Bayes methods. *Biostatistics* **8**, 118–127 (2007).
7. Butler, A., Hoffman, P., Smibert, P., Papalexi, E. & Satija, R. Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nat. Biotechnol.* **36**, 411–420 (2018).
8. 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).
9. 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).
10. Ritchie, M. E. et al. limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Res.* **43**, e47 (2015).
11. Cressie, N. & Timothy, R. C. Pearson's χ^2 and the loglikelihood ratio statistic G^2: a comparative review. *Int. Stat. Rev.* **57**, 19–43 (1989).
12. Klein, A. M. et al. Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. *Cell* **161**, 1187–1201 (2015).
13. Brennecke, P. et al. Accounting for technical noise in single-cell RNA-seq experiments. *Nat. Methods* **10**, 1093–1095 (2013).
14. Kolodziejczyk, A. A. et al. Single cell RNA-sequencing of pluripotent states unlocks modular transcriptional variation. *Cell Stem Cell* **17**, 471–485 (2015).
15. Angerer, P. et al. Single cells make big data: new challenges and opportunities in transcriptomics. *Curr. Opin. Syst. Biol.* **4**, 85–91 (2017).
16. Biase, F. H., Cao, X. & Zhong, S. Cell fate inclination within 2-cell and 4-cell mouse embryos revealed by single-cell RNA sequencing. *Genome Res.* **24**, 1787–1796 (2014).
17. Liu, W. et al. Identification of key factors conquering developmental arrest of somatic cell cloned embryos by combining embryo biopsy and single-cell sequencing. *Cell Discov.* **2**, 16010 (2016).
18. Goolam, M. et al. Heterogeneity in Oct4 and Sox2 targets biases cell fate in 4-cell mouse embryos. *Cell* **165**, 61–74 (2016).
19. Fan, X. et al. Single-cell RNA-seq transcriptome analysis of linear and circular RNAs in mouse preimplantation embryos. *Genome Biol.* **16**, 148 (2015).
20. Boroviak, T. et al. Lineage-specific profiling delineates the emergence and progression of naive pluripotency in mammalian embryogenesis. *Dev. Cell* **35**, 366–382 (2015).
21. Deng, Q., Ramsköld, D., Reinius, B. & Sandberg, R. Single-cell RNA-seq reveals dynamic, random monoallelic gene expression in mammalian cells. *Science* **343**, 193–196 (2014).
22. Xue, Z. et al. Genetic programs in human and mouse early embryos revealed by single-cell RNA sequencing. *Nature* **500**, 593–597 (2013).
23. Wu, J. et al. The landscape of accessible chromatin in mammalian preimplantation embryos. *Nature* **534**, 652–657 (2016).
24. Patro, R., Duggal, G., Love, M. I., Irizarry, R. A. & Kingsford, C. Salmon provides fast and bias-aware quantification of transcript expression. *Nat. Methods* **14**, 417–419 (2017).
25. Teng, M. et al. A benchmark for RNA-seq quantification pipelines. *Genome Biol.* **17**, 74 (2016).
26. Kang, H. M. et al. Multiplexed droplet single-cell RNA-sequencing using natural genetic variation. *Nat. Biotechnol.* **36**, 89–94 (2018).
27. Liu, Q. et al. Quantitative assessment of cell population diversity in single-cell landscapes. *bioRxiv* Preprint at https://www.biorxiv.org/content/early/2018/05/30/333393 (2018).
28. Risso, D., Ngai, J., Speed, T. P. & Dudoit, S. Normalization of RNA-seq data using factor analysis of control genes or samples. *Nat. Biotechnol.* **32**, 896–902 (2014).
29. Bacher, R. et al. SCnorm: robust normalization of single-cell RNA-seq data. *Nat. Methods* **14**, 584–586 (2017).
30. 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).
31. Buettner, F., Pratanwanich, N., McCarthy, D. J., Marioni, J. C. & Stegle, O. f-scLVM: scalable and versatile factor analysis for single-cell RNA-seq. *Genome Biol.* **18**, 212 (2017).
32. Li, W. V. & Li, J. J. An accurate and robust imputation method scImpute for single-cell RNA-seq data. *Nat. Commun.* **9**, 997 (2018).
33. 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).
34. Huang, M. et al. SAVER: gene expression recovery for single-cell RNA sequencing. *Nat. Methods* **15**, 539–542 (2018).
35. van Dijk, D. et al. Recovering gene interactions from single-cell data using data diffusion. *Cell* **174**, 716–729 (2018).
36. Wolf, F. A., Angerer, P. & Theis, F. J. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biol.* **19**, 15 (2018).
37. Saelens, W., Cannoodt, R., Todorov, H. & Saeys, Y. A comparison of single-cell trajectory inference methods: towards more accurate and robust tools. *bioRxiv* Preprint at https://www.biorxiv.org/content/early/2018/03/05/276907 (2018).
38. Bhaduri, A., Nowakowski, T. J., Pollen, A. A. & Kriegstein, A. R. Saturating single-cell datasets. *bioRxiv* Preprint at https://www.biorxiv.org/content/early/2017/11/12/218370 (2017).
39. Regev, A. et al. The Human Cell Atlas. *eLife* **6**, e27041 (2017).
40. *Tabula Muris* Consortium. Single-cell transcriptomic characterization of 20 organs and tissues from individual mice creates a *Tabula Muris*. *bioRxiv* Preprint at https://www.biorxiv.org/content/early/2018/03/29/237446 (2018).
52. Aken, B. L. et al. Ensembl 2017. *Nucleic Acids Res.* **45**, D635–D642 (2017).

Acknowledgements

We thank A. Böttcher for motivating this study, and T. Illicic for carrying out pilot analyses. We thank in particular M. Subramaniam and J. Ye (UCSF) for the PBMC data. We are grateful to the members of the Teichmann and Theis labs for valuable discussions and comments on the manuscript. M.B. is supported by a DFG Fellowship through the Graduate School of Quantitative Biosciences Munich (QBM). Z.M. is supported by a Single Cell Gene Expression Atlas grant from the Wellcome Trust (nr. 108437/Z/15/Z). F.A.W. acknowledges support by the Helmholtz Postdoc Programme, Initiative and Networking Fund of the Helmholtz Association. F.J.T. acknowledges financial support by the German Science Foundation (SFB 1243 and Graduate School QBM) and by the Bavarian government (BioSysNet). This collaboration was supported by a Helmholtz International Fellow Award to S.A.T.

Author contributions

M.B. developed, tested and validated the method; prepared and analyzed the data; and wrote the paper. Z.M. prepared and analyzed the data and wrote the paper. F.A.W. assisted with method development and manuscript writing. S.A.T. and F.J.T. oversaw the research, designed the method validation and wrote the paper.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information is available for this paper at https://doi.org/10.1038/s41592-018-0254-1.

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

Correspondence and requests for materials should be addressed to S.A.T. or F.J.T.

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

Methods

kBET: *k*-nearest-neighbor batch estimation. Let the full gene expression dataset $D = \{x_1, \ldots, x_n\}$, where $x_j \in R^g$ and $X \in R^{n \times g}$ is the corresponding gene expression data matrix with n samples and g genes. In an scRNA-seq dataset X, each sample has meta-information such as cell type, FACS gate, and the batch i that it was measured in.

The batch variable i has l categories such that n_i denotes the number of samples in batch i, $f_i = n_i/n$ is the global fraction of samples in batch i, and $\nu = (n_1, \ldots, n_l)$ is the batch configuration of all samples. Also, we define $\tilde{f}_i$ as the local fraction of samples in batch i in some subset $N \subset D$. In particular, we consider subsets of k nearest neighbors.

Then we formulate the null hypothesis of data being 'well mixed', that is, the absence of a batch effect, as

$$\tilde{f}_i = f_i \;\; \forall\, i \in \{1, \ldots, l\} \forall \text{ for subsets } N \subset D$$

In order to statistically test the null hypothesis, we define a neighborhood subset $N_j = x_j \cup \{x_s |\; s \text{ among } k-1 \text{ nearest neighbors of } j\}$. Nearest neighbors are computed with the cover-tree algorithm (FNN R package). To optimize computation efficiency, we precompute the first 50 eigenvectors of the largest eigenvalues with the svd function and use the reduced dataset to find nearest neighbors.

Let n_{ji}^k denote the number of cells in batch i that are in subset j of size k. Testing the null hypothesis involves two steps:

1. We test the null hypothesis in each subset N_j of a given sequence of subsets. In each subset N_j, this amounts to testing whether the distribution of n_{ji}^k with respect to i matches the distribution under the null hypothesis.
2. We summarize the result of the sequence of tests by computing the average rejection rate S over all tests—a test statistic for the whole dataset. Hence, testing whether S exceeds a significance threshold allows for rejection of the null hypothesis for the whole dataset.

Note that by carrying out these two steps, we go beyond a standard test for homogeneity of subsets of a given dataset.

χ^2-based test. In the limit of high values of k, n_{ji}^k is Gaussian-distributed with respect to i. A test for small values of k is provided as an exact test (Supplementary Note 1). Then, we can use Pearson's χ^2 test, the test statistic of which is

$$\kappa_j^k = \sum_{i=1}^{l} \frac{(n_{ji}^k - f_i \times k)^2}{f_i \times k} \sim \chi^2_{l-1}$$

where χ^2_{l-1} denotes the χ^2 distribution with $l-1$ degrees of freedom. The P value for each κ_j^k is computed as

$$P_j^k = 1 - F_{l-1}(\kappa_j^k)$$

where $F_{l-1}(x)$ denotes the cumulative distribution function of the χ^2 distribution with $l-1$ degrees of freedom.

Principal component regression. PCA is an orthogonal transformation of the data matrix into a set of linearly uncorrelated variables. The PCs correspond to the eigenvectors of the covariance matrix of the data and are ordered by the explained variance of the data. If a batch effect is present in the data, it contributes to the variance, and a corresponding batch covariate correlates significantly with some of the PCs. As the set of PCs is uncorrelated, regression of the batch covariate B (with l categories defined in the kBET model) and the ith PC returns the coefficient of determination as an approximation of the variance explained by B in each PC (PC regression, similar to ref. [41]). Overall, the total contribution of the batch effect to the variance in the data may be approximated by

$$\mathrm{Var}(C|B) = \sum_{i=1}^{G} \mathrm{Var}(C|\mathrm{PC}_i) \cdot R^2(\mathrm{PC}_i|B)$$

where $\mathrm{Var}(C|\mathrm{PC}_i)$ is the variance of C explained by the ith PC. However, using a linear regression model enables us to evaluate the significance of $R^2(\mathrm{PC}_i|B)$. For the case of two batches, the significance test equals a univariate, two-tailed t-test on the loadings of each PC split by batch covariate. For more than two batches, the univariate t-test can be generalized to a one-way analysis of variance, for which the test measure is F-distributed. We use this approach to compute the fraction of significantly correlated PCs (default, top 50 PCs; in the case of CCA, the top 10 PCs). P values were adjusted to FDR < 0.05.

However, as the number of features (genes) increases, the largest and smallest eigenvalues of the sample covariance matrix converge[42]. Consequently, $\mathrm{Var}(C|B)$ decreases with the number of features as well, and because of the high dimensionality of scRNA-seq data, batch effects defined by explained variance are difficult to interpret.

Therefore, we use the sum of explained variance of all PCs with significant $R^2(\mathrm{PC}_i|B)$ scaled by the variance explained by the top 50 PCs as a proxy for the batch effect:

$$\mathrm{Var}(C|B)_{\mathrm{scaled}} \approx \frac{\sum_{i \in W} \mathrm{Var}(C|\mathrm{PC}_i)}{\sum_{i=1}^{N} \mathrm{Var}(C|\mathrm{PC}_i)}$$

where W is the index set of all N top PCs that are significantly correlated with the batch covariate (Supplementary Note 2).

Silhouette. The calculation of a silhouette aims to determine whether a particular clustering has minimized within-cluster dissimilarity and maximized inter-cluster dissimilarity[43]. Let us assume that there is a given clustering into more than one cluster. For each sample i, the silhouette width is defined as follows.

Let $a(i)$ be the average dissimilarity between i and all other data points of its cluster A. If i is the only observation in this cluster, set $s(i) := 0$. For all other clusters $C \neq A$, let $d(i, C)$ be the average dissimilarity of i to all samples of C. There is some cluster B whose dissimilarity $d(i, B)$ is minimal: $b(i) := \min_C d(i, C)$, which is the 'neighboring' cluster to sample i. Then, the silhouette width $s(i)$ is defined as the scaled difference of average dissimilarity within a cluster and the average dissimilarity to its neighboring cluster:

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

Finally, the mean of all silhouette widths $s(i)$ gives the silhouette s from which we display its absolute value (Fig. 2). We adapted the calculation from the scone R package[28].

The silhouette width $s(i)$ ranges from –1 to 1, with $s(i) \to 1$ if two clusters are separate and $s(i) \to -1$ if two clusters overlap but have dissimilar variance. If $s(i) \to 0$, both clusters have roughly the same structure. Thus, we use the absolute value $|s|$ as an indicator for the presence or absence of batch effects.

Computation of highly variable genes. To determine whether a batch-correction method is overcorrecting, we check the number of HVGs before and after batch correction (which was not possible for Seurat's batch correction, as it does not return a batch-corrected data matrix). In the Anders (Brennecke)[13] model implemented in the M3Drop[44] package ('BrenneckeGetVariableGenes' function), the relation of the squared coefficient of variation (CV^2) and mean for each gene follows a Gamma model, $CV^2 \sim (\alpha_1/\mu) + \alpha_0$. CV^2 decreases with increasing mean gene expression. The slope parameter α_1 and offset α_0 are estimated by nonlinear least-squares fit. A gene is considered as highly variable if its CV^2 is higher than expected from its mean, that is, if it is above the model fit curve in a plot of mean CV^2.

To define a batch-free gene set before batch correction, we fit the Anders (Brennecke) model to each batch separately and intersect the corresponding sets of HVGs. Let l be the number of batches and a_i be the set of HVGs for batch i. We define

$$\mathrm{HVG}_{\mathrm{batch-free}} = \bigcap_{i=1}^{l} a_i$$

as the set of HVGs present in each of the batches in a dataset.

More specifically, we consider the fact that HVGs depend on the type of normalization[45]. Then, the reference set of HVGs consists of all genes that are highly variable in all batches with log(counts + 1) normalization. After batch correction, we compute HVGs for the whole corrected dataset ($\mathrm{HVG}_{\mathrm{corr}}$). Ideally, we would retain all $\mathrm{HVG}_{\mathrm{batch\text{-}free}}$ after batch correction. We define the fraction of retained batch-free HVGs as

$$p_{\mathrm{retained}} = \frac{|\mathrm{HVG}_{\mathrm{batch-free}} \cap \mathrm{HVG}_{\mathrm{corr}}|}{|\mathrm{HVG}_{\mathrm{batch-free}}|}$$

to determine whether the biological signal in the data is preserved after batch correction.

False positive rate for highly variable genes. We quantify the number of HVGs caused by the batch effect as an FPR. In contrast to the fraction of retained HVGs, we define the FPR by the fraction of HVGs that are found in the whole dataset but not in any of the batches. More formally, let

- a denote the set of HVGs in the complete dataset, and
- a_i denote the set of HVGs in batch i.

Then, the FPR is

$$\mathrm{FPR} = 1 - \frac{|\bigcup_{i=1}^{l} (a \cap a_i)\,|}{|a|}$$

Data normalization. Data normalization methods account for the sequencing depth as a size factor and normalize the expression data to the same comparable level. The normalization methods used are summarized in Supplementary Table 1.

Briefly, we used the following: (1) CPM based on the library size; (2) relative log expression; (3) trimmed mean of *M* values; (4) scran size factor[46]; (5) qsmooth from the YARN package[47]; (6) transcripts per million, derived from the mapping by Salmon[24] (version 0.8.2); and (7) mean ratio normalization, which uses size factors from the DESeq2 package[9].

Batch regression. Methodologically, the recent batch-regression approaches either require the assignment of batches as input or assess bias in the data independently of batch information. In this paper, we compare five established batch-regression methods (details in Supplementary Table 1): (1) limma, which uses a linear regression model to remove batch effects (we used the 'removeBatcheffect' function from the limma package[10]); (2) the ComBat model[6] function from the sva package[48], which is a linear regression model based on empirical Bayes methods; (3) the f-scLVM model, a factor-analysis-based latent variable model whereby, after model training, the batch-effect-related factors are removed with the 'regressOut' function implemented in the fscLVM package[31]; (4) PEER, based on factor analysis[49]; and (5) RUVs, RUVr and RUVg from the RUVseq package[28], which remove unwanted variance according to replicate samples, residuals and control genes, respectively. We derived control genes using the edgeR package[50] and used the top 400 constant genes as control genes. The model parameter *k* in RUVseq and PEER indicates the number of hidden factors correlated with the variance. We tested several values from 1 to 7 and 25% of the sample size. We also investigated (6) Seurat's batch effect correction, based on CCA and dynamic time warping[7], and (7) MNN[8], which uses cosine similarity as internal normalization. Methods 1–3, 6 and 7 require batch information for correction; methods 4 and 5 assess general bias in the data.

Simulated data. We used two different models for simulating scRNA-seq data and batch effects. The first model is based on a zero-inflated negative binomial distribution for count data similar to that in ref. [46]. The second approach uses the Splat model of the R framework Splatter[51] (Supplementary Note 3).

A zero-inflated negative binomial model. We modeled the number of transcripts per gene and per cell as count data that followed the negative binomial distribution with zero inflation to account for dispersion and sparsity caused by dropouts. Mean expression levels for each gene were sampled from the beta-distribution (with appropriate scaling):

$$\mu \sim \mathrm{Beta}(a, b) \cdot c$$

with parameters $a = 2$, $b = 5$ and $c = 100$. The dropout probability for each simulated gene $j \in \{1, \ldots, G\}$ in batch $i \in \{1, 2\}$ was modeled by the logistic (sigmoid) function $p_{ij} = \mathrm{sigm}(-(\beta_0 + \beta_{1,i}\mu_{ij}))$, where we chose $\beta_0 = -1.5$ and $\beta_{1,i} = 1/\mathrm{median}(\mu_i)$. Every sample was drawn from $s_{ij} \sim NB(\mu_{ij}, \theta|\mathrm{Ber}(p_{ij}))$, where $\theta = 1$ and Ber is the Bernoulli distribution.

Batch-effect strength is modeled as an increasing fraction of affected genes. With the parameters for the first batch set up, the mean expression levels of the second batch μ_2 are subject to different degrees of variation. We multiply 1%, 10% and 20% of the mean expression levels μ by a gamma-distributed random variable $\gamma \sim \Gamma(\alpha, \beta)$ and $\alpha = \beta = 1$:

$$\mu_{2,j} = \begin{cases} \mu_{1,j} \cdot \gamma & \text{when } j \in \{1, \ldots, h \cdot G\} \\ \mu_{1,j} & \text{otherwise} \end{cases}$$

where $h \in \{1\%, 10\%, 20\%\}$ and G is the number of genes in the dataset.
The gamma distribution with the chosen parameters has a mean and variance equal to 1 such that the expected value of the sampled mean expression levels stays unchanged. In addition, we varied the sample size of the two batches: in each simulation, we sampled 500 instances with 1,000 genes each, with the size ratio of the batches being

$$r \in \left\{1, \frac{1}{3}, \frac{1}{4}, \frac{1}{9}, \frac{1}{19}\right\}$$

This means equally sized batches contained 250 samples each, and batches with $r = 1/9$ had 450 and 50 samples, respectively.

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

Data availability

We applied the batch estimates to several scRNA-seq datasets. In the inDrop publication, the droplet-based sequencing was demonstrated on mESCs growing on LIF+ medium and two additional technical replicates[12]. In our analysis, we used two replicates that consisted of 5,952 cells from two batches and 11,308 genes with at least 2 cells having more than 4 unique molecular identifier (UMI) reads per cell. Data were downloaded as UMI-filtered read count matrices from accession GSE65525.
Kolodziejczyk et al.[14] explored heterogeneity in mESCs cultured with three different media (2i, a2i and LIF+) on full-length sequenced transcripts (Smart-seq). The three conditions included 219, 123 and 207 cells in 4, 2 and 3 batches, respectively. The mESC data sequenced with full-length Smart-seq[14] were downloaded from ENA (project ID PRJEB6455) as FASTQ files and mapped to an Ensembl[52] mouse transcriptome (GRCm38.p5.87, equivalent to UCSC mm10) with Salmon[24]. Cells were quality-controlled according to data derived from the Espresso database (http://www.ebi.ac.uk/teichmann-srv/espresso/).
Further, scRNA-seq has been widely applied in explorations of mouse embryonic development. To test the performance of batch correction for data integration, we collected single-cell RNA-seq data of mouse early embryonic development from eight different studies[16–23], consisting of 56, 49, 124, 65, 15, 294, 17 and 15 cells, respectively. The early embryonic development data used have the following accession IDs: E-GEOD-57249, E-GEOD-70605, E-MTAB-3321, GSE53386, E-MTAB-2958, E-GEOD-45719, E-GEOD-44183 and E-GEOD-66582. All studies applied Smart-seq-based protocols for scRNA-seq. All FASTQ files were mapped to an Ensembl[52] mouse transcriptome (version GRCm38.p5.87) with Salmon[24] (version 0.8.2; *k*-mer = 21 to tolerate different read lengths). Here we considered the studies as batches while omitting the flowcell batches. We continued our analysis without further gene filtering or quality control.
Kang et al.[26] studied genetic variation among PBMCs from eight individuals as a replacement for cell barcoding in droplet-based sequencing (10x Genomics). From that study, we used three experimental runs: 3,514 and 4,106 cells from four healthy donors each, and 5,832 cells from these eight healthy donors. Human PBMC data[26] can be provided by the authors upon request. Count matrices are available under accession number GSE96583.

References

41. McCarthy, D. J., Campbell, K. R., Lun, A. T. L. & Wills, Q. F. Scater: pre-processing, quality control, normalization and visualization of single-cell RNA-seq data in R. *Bioinformatics* **33**, 1179–1186 (2017).
42. Baik, J. & Silverstein, J. W. Eigenvalues of large sample covariance matrices of spiked population models. *J. Multivariate Anal.* **97**, 1382–1408 (2006).
43. Rousseeuw, P. J. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* **20**, 53–65 (1987).
44. Andrews, T. S. & Hemberg, M. Dropout-based feature selection for scRNASeq. *bioRxiv* Preprint at https://www.biorxiv.org/content/early/2018/05/17/065094 (2018).
45. 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).
46. Lun, A. T. L., Bach, K. & Marioni, J. C. Pooling across cells to normalize single-cell RNA sequencing data with many zero counts. *Genome Biol.* **17**, 75 (2016).
47. Paulson, J. N. et al. Tissue-aware RNA-Seq processing and normalization for heterogeneous and sparse data. *BMC Bioinformatics* **18**, 437 (2017).
48. Leek, J. T., Johnson, W. E., Parker, H. S., Jaffe, A. E. & Storey, J. D. The sva package for removing batch effects and other unwanted variation in high-throughput experiments. *Bioinformatics* **28**, 882–883 (2012).
49. Stegle, O., Parts, L., Piipari, M., Winn, J. & Durbin, R. Using probabilistic estimation of expression residuals (PEER) to obtain increased power and interpretability of gene expression analyses. *Nat. Protoc.* **7**, 500–507 (2012).
50. 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).
51. Zappia, L., Phipson, B. & Oshlack, A. Splatter: simulation of single-cell RNA sequencing data. *Genome Biol.* **18**, 174 (2017).

nature research

Corresponding author(s): Fabian J. Theis

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	To simulate data, we used custom code and the R package 'splatter' available on Bioconductor and Github.
Data analysis	For analysis, we use the custom R package 'kBET' available on Github (and given as zip-file).

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

The mESC data sequenced with inDrop11 were downloaded as UMI-filtered read count matrices with accession number GSE65525.

The mESC data sequenced with full length SMART-seq23 were downloaded from ENA (project id: PRJEB6455) as fastq files and mapped to Ensembl56 mouse transcriptome (GRCm38.p5.87, equivalent to UCSC mm10) with Salmon33. Cells were quality controlled according to data derived from the Espresso database (http://www.ebi.ac.uk/teichmann-srv/espresso/).
Early embryonic development data were derived from several studies25–32 with accession ids: E-GEOD-57249, E-GEOD-70605, E-MTAB-3321, GSE53386, EMTAB-2958, E-GEOD-45719, E-GEOD-44183 and E-GEOD-66582. All studies applied SMARTseq-based protocols for single-cell RNA-seq. All fastq files were mapped to Ensembl56 mouse transcriptome (version GRCm38.p5.87) with Salmon33 (version 0.8.2, kmer = 21 to tolerate different read length). Here, we only consider the studies as batches while omitting the flowcell batches. We continued our analysis without further gene filtering or quality control.
Human PBMC data35 are provided upon request by the authors. Also, count matrices are available with accession number GSE96583.

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	For simulated data, we chose sample size according to common single-cell RNAseq data sample size. For actual single-cell RNAseq data, we used published datasets.
Data exclusions	From published datasets, we selected only those parts which contained several replicates.
Replication	From published datasets, we selected only those parts which contained several replicates.
Randomization	Does not apply for the manuscript as we worked with published data.
Blinding	For kBET, we use the batch information to estimate the impact of the batch effect. As part of kBET, we developped a null model where we permute the batch effect labels of the cells. Expected and observed average rejection rates are compared. Blinding would correspond to the permutation of batch labels in our case.

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