PMD Uncovers Widespread Cell-State Erasure by scRNAseq Batch Correction Methods

Scott R Tyler,[1,*] Supinda Bunyavanich,[1,2,*] and Eric E Schadt[1,3,*]

[1]Department of Genetics and Genomic Sciences, Icahn School of Medicine at Mount Sinai, New York, NY, USA.

[2]Division of Allergy and Immunology, Department of Pediatrics, Icahn School of Medicine at Mount Sinai, New York, NY, USA.

[3]Sema4, a Mount Sinai venture, Stamford CT, 06902, US

*Corresponding authors

Abstract

Single cell RNAseq (scRNAseq) batches range from technical replicates to multi-tissue atlases, thus requiring robust batch correction methods that operate effectively across this similarity spectrum. Currently, no metrics allow for full benchmarking across this spectrum, resulting in benchmarks that quantify *removal* of batch effects without quantifying *preservation* of real batch differences. Here, we address these gaps with a new statistical metric [Percent Maximum Difference (PMD)] that linearly quantifies batch similarity, and simulations generating cells from mixtures of distinct gene expression programs (cell-lineages/-types/-states). Using 690 real-world and 672 simulated integrations (7.2e6 cells total) we compared 7 batch integration approaches across the spectrum of similarity with batch-confounded gene expression. Count downsampling appeared the most robust, while others left residual batch effects or produced over-merged datasets. We further released open-source PMD and downsampling packages, with the latter capable of downsampling an organism atlas (245,389 cells) in tens of minutes on a standard computer.

In bulk RNAseq experiments requiring several sequencing batches, samples are randomized across batches to minimize the effects of batch on any biologic group. In most single cell RNAseq (scRNAseq) experiments, however, biologic replicates are often processed and sequenced in their own batch making biologic variance intrinsically confounded by batch effects. While cell hashing enables co-processing of single cells from different biologic samples[1], this approach, decreases sensitivity through read competition with transcripts and requires simultaneous sample preparation, which can be difficult or impossible with clinical research or time-course experimental designs.

scRNAseq experimental batches can range in similarity from replicates to multi-tissue atlases[2, 3] where batches have little-to-no overlap in composition. An ideal batch correction method would be robust across a broad range of batch similarities, from replicates to divergent batches. Because one cannot know *a priori* which sources of variation are biologic or technical, methods should not rely on assumptions of the overall nature of the input datasets and where they lie on this similarity spectrum.

Two major classes of scRNAseq batch correction/normalization approaches currently exist. One classof approaches normalizes counts or distributions within cells (referred to as depth normalization), while the second performs some method of latent-dimension mapping, identifying similarly behaving components across batches. Methods of the latter category include Multiple Canonical Correlation Analysis (MCCA),[4] linear model correction of fuzzy-clustering results on Principal Component Analysis (PCA) reduced matrices,[5] Non-Negative Matrix Factorization (NMF),[6] and graph network embeddings, among others[7, 8]. These approaches have loss functions that increase overlap between datasets or maximize concordant components.

While it is important to account for technical variation by batch, it is also important to retain true biologic variation across batches. A cell's -lineage, -type, and -state determine different cellular expression programs; real variation in cell-state can be driven by many

sources, including serum/media batches, dissociation/trituration techniques,[9] and the amount of time cells were kept on ice or room-temperature during processing[10]. Despite this complexity of mixed gene expression programs, many scRNAseq simulation approaches apply a more simplistic approach for simulating cell clusters, with clusters simulated from a single source of differentially expressed genes[11].

Assessments of batch correction algorithms often judge the ability of the algorithms to integrate batches from different technologies, quantifying *only* the ability to *remove differences* between batches[4, 12]. While important for cross-technology/species meta-analyses of highly similar datasets, a bench biologist performing a treatment/control or disease/non-diseased experiment will typically perform assays on a single platform, without *a priori* knowledge of the ground-truth similarity of replicates, batches, or treatments. Enabled by our creation of a new statistic [Percent Maximum Difference (PMD)], here, we assess the ability of normalization/batch correction algorithms'[4-6, 13, 14] to *remove* batch specific variation while *preserving* biologically real, batch confounded variation. Our findings point towards UMI downsampling as the most robust method, with all other approaches tested either leaving residual batch effects, or erasing batch confounded real variance resulting in the erroneous merging of disparate cell-types/-states.

Results

Efficacy with identical sub-sampled cells

In many published post-processing datasets, the range in total unique molecular identifier (UMI) counts observed in individual cells varies by three orders of magnitude, and across datasets, an order of magnitude difference in the distribution of total UMI per cell is also typical (**Extended Data Fig. 1**)[15]. We therefore devised an approach to quantify batch integration performance when batches *only* vary in the total observed counts. The transcriptome of each observed cell represents a sub-sample of the cell's original transcriptome (**Fig. 1a**)[15] - we refer to these observed cells as "parent cells." We sub-sampled the observed dataset to 50% of the original sampling depth (**Fig. 1a,b**), to create sub-sampled "child cells." If correction algorithms effectively remove the sampling depth effect, each parent cell should co-cluster with its child cell, given that each cell-pair is a sub-sampling of the same ground truth transcriptome from the original cell (**Fig. 1c**). A parent-child cell co-cluster ratio (percent-same) of 1.0 indicates that all child cells clustered with their corresponding parent cells, whereas a match ratio of 0.0 indicates that none of the child cells clustered with their corresponding parent cells (See "percent-same" calculation; **Fig. 1a,b**).

We used this approach on a dataset of 1,000 neurons,[16] expecting latent-dimension mapping methods to perform the best (high percent-same), given that they have explicit loss functions to maximize overlap. Indeed, Harmony, a latent-dimension mapping approach, showed the greatest percent-same, although other latent-dimension mapping methods showed lower percent-same scores compared to downsampling (**Fig. 1d,e**). Notably, adding a third orthogonal dataset (homeostatic intestine) showed similar results (**Box 1b**; **Extended Data Fig. 2**).

Efficacy with highly similar biological datasets

Biologic interpretation of scRNAseq typically builds on clustering results, it is therefore important to understand and quantify the effect of batch correction on clustering results. However, batch correction algorithms may leave differing levels of residual variance in the final integrated dataset, resulting in differing numbers of clusters; yet with real-world datasets the ground truth number of clusters is unknown. A useful metric comparing batch similarity must therefore be invariant to the number of clusters found. Unlike the "percent-same" metric, real-world integration tasks integrate different cells without parent-child pairs, necessitating a different metric. In contrast to kBET, which makes the assumption of equivalent and interchangeable batches[17], we sought a metric that does not assume equivalent batches and precisely quantifies batch similarity. We therefore created the "percent maximum difference" (PMD) metric/test that quantifies the overall similarity in cluster composition across batches. PMD is provably invariant to the number of clusters found when relative overlap in cluster composition is preserved, operates linearly across the spectrum of batch similarity, is unaffected by batch size differences or overall number of cells, and does not require that batches be similar, filling a crucial gap in the field for benchmarking scRNAseq batch correction assessment (**Extended Data Fig. 3**).[17] These properties are not present in any other statistical metric benchmarked here including χ^2, χ^2 -log10(p-value), or Cramer's V (**Extended Data Fig. 8**; see **Methods** for proof, benchmark, and detailed characterization of PMD). PMD yields a single quantitative representation on a scale of 0-1 of overall batch dissimilarity (1), or similarity (0). PMD=1 for batches with no overlapping clusters, while PMD≈0 for two batches from the same single cell suspension (see **Extended Data Fig. 3b,c** for a visual representation and calculation details). To make the PMD approach broadly useful to the community, we created and released the open source 'PercentMaxDiff' R package (https://github.com/scottyler89/PercentMaxDiff).

Using PMD, we quantified batch similarity after correction, when cells came from the same single cell suspension, where PMD is expected to be low/near-zero (**Box 1c:** 1k and 10k neurons)[16, 18]. Similar to our results from the "percent-same" figures of merit, we found that

Harmony followed by UMI downsampling, then Liger provided the greatest batch similarity (measured by lowest PMD), while other approaches yielded batches whose cluster composition was more dissimilar than no correction indicating residual or magnified batch effects (**Extended Data Fig. 4**).

Efficacy with dissimilar biologic datasets

To test performance on the opposite side of the similarity/dissimilarity spectrum, something not addressed previously in comparing batch correction approaches, we utilized datasets with no expected overlap in cluster composition: brain[16] and intestine[19] (**Box 1d**; **Fig. 2a,b**; expected PMD=1). Given that depth normalization approaches do not have explicit loss-functions to increase overlap, they are expected to yield integrations with higher PMD(≈1). Indeed, all normalization approaches indicated no overlap in cluster composition (PMD=1), while latent-dimension mapping approaches showed up to 40.2% (1-PMD) overlap between the brain and intestine (**Fig. 2c**).

To quantify neuronal/intestinal cell mixing, we calculated the percent of clusters that contained cells that expressed *NEUROD2* (a neuronal transcription factor), and cells that expressed *FABP1* (an intestinal enterocyte marker gene), as these cells can be reasonably assumed to be erroneously co-clustered. The latent-dimension mapping methods did indeed harbor mixed neuron/intestinal epithelial clusters (>30% of clusters mixed in full datasets), while depth normalization approaches did not (**Extended Data Fig. 5a-c**).

Thus far we have shown that our implementation of UMI downsampling and Harmony perform consistently well when the input batches are expected to be identical or extremely similar. However, when datasets share no biologic overlap, depth normalization methods (including downsampling) preserve these differences, while latent-dimension mapping approaches overestimate the level of overlap between batches, resulting in disparate cell populations being merged. The most realistic scenario for experimenters however, is that

batches will have unknown overlap, where the truth lies somewhere between the two extremes of replicates and complete divergence.

Results with biologic datasets of uncertain overlap

We created two biologic figures-of-merit that reflect the frequent reality of uncertain overlap. The first benchmark is the integration of peripheral blood mononuclear cells (PBMCs)[20] integrated with cells from the brain,[16] expecting them to be mostly, but perhaps not entirely, different (**Box 1d**); the premise being that some of the brain's blood may have been left behind during the perfusion process (**Extended Data Fig. 6a,b**).

Indeed, all methods found some overlap between these datasets, but the degree of overlap differed widely. UMI downsampling, showed the least level of overlap (1.7% similar (1-PMD); PMD=0.983) followed by no correction and normalization approaches; latent-dimension mapping approaches, however, found roughly 6.5-7 times that overlap (11.1-11.9% similar (1-PMD)) (**Extended Data Fig. 6c**). All methods found a cluster that was shared between the two datasets representing the highly similar microglia/macrophage cells (**Extended Data Fig. 6d,e**).[21] Many latent-dimension mapping approaches, however, placed PBMCs into the same clusters as neuronal populations, and in some cases B-cells co-clustered with neurons (**Extended Data Fig. 6e**). These data indicate that with dissimilar datasets with some small overlap, depth normalization methods correctly preserve these differences, while latent-dimension mapping methods push datasets together.

We next examined homeostatic intestine and irradiated intestine (**Box 1e**; **Fig. 3a,b**)[19], expecting some overlap, although given the tissues are in dramatically different cell-states, we did not have reasonable expectations for the percent overlap. We saw a notably large range in PMD (**Fig. 3c**), ranging from a PMD of 99.1% with no-correction to a PMD of 20.2% (Harmony) (**Fig. 3c**). These results demonstrate that the choice of batch correction method can drastically alter the biologic conclusions of an experiment with batches of divergent cell-states.

Discrete sample types vs continuously divergent sample types

A remaining question is the impact of integrating samples with many replicates that originate from discrete sample-types or integrating samples that exist across a continuum of changing cell-states and cell-type abundances. To address the former, we integrated cortical and substantia nigra samples, each with replicates (**Fig. 4a-c**)[22]. Pairwise sample-sample PMD heatmaps show clear separation of cortical from substantia nigra samples across all methods. In all cases, samples from the same brain location were more similar to each other compared to across sample types (**Fig. 4d**), indicating that under this scenario the choice of integration approach may have minimal impact on biologic conclusions. However, using endometrium over the course of the menstruation cycle[23] to model continuously changing cell states and cell-type abundances, depth normalization approaches were shown to be relatively consistent with each other identifying these samples as largely dissimilar, with changing cell states and cell-type abundances. On the other hand, latent dimension mapping approaches dramatically altered the pattern of changes over time (**Fig. 4e**). Similar to cell-state erasure in homeostatic and irradiated intestine, latent dimension mapping approaches increased sample-sample similarity (lower PMD), but also changed the pattern of stage-to-stage similarity in a manner unique to each of these methods (**Fig. 4f,g**).

Synthetic dataset benchmarking with simulated cell-types/-states

The disparate results from algorithms integrating homeostatic and irradiated intestine, as well as divergent results with samples over a continuum of similarity (endometrium throughout menstruation cycle), highlight the need for simulations that can untangle the effect of mixed expression programs corresponding to cell-lineage -type and/or -state. We therefore simulated a mixture model of negative binomial distributed gene expression profiles,[11, 24] to test the effect of

batch confounded gene-expression program mixtures, where batches differ in sequencing depth and are variably confounded with biology (See **Methods** for details).

We hypothesized that algorithm discordance based on the homeostatic and irradiated intestine datasets may stem from batch correction algorithms maintaining different levels of biologically real, but batch confounded, sources of variation. To test this hypothesis, we simulated six cell-types, with a cell-state optionally added, some of which appear in a batch confounded manner (**Fig. 5a**). Overall, we simulated 8 different scenarios in which cell-states were added in different arrangements across batches, in combinations with varying levels of clusters appearing in both batches, or in a batch specific manner; this totaled 32 simulation scenarios, each simulated in triplicate (96 batch integration tasks total) (**Extended Data Fig. 7**). Testing these 7 dataset integration approaches sums to 672 scRNAseq analyses, representing 9.8e5 synthetic cells. To analyze these simulations, we employed 9 different quantitative metrics including those that measure global cluster accuracy after correction, the ability of algorithms to successfully remove the batch effect, and the efficacy of algorithms in maintaining multiple sources of biologic variance, including when confounded by batch (**Box 1f**; **Extended Data Table 1**).

Simulation results: overall batch correction accuracy by simulation

In our figures of merit utilizing real-datasets, we relied on assumptions of dataset similarity based on the biologic underpinnings of the datasets. With simulation however, we can calculate the ground truth PMD based on known clusters. We therefore used several metrics comparing the observed PMD in clustering results after integration relative to the ground truth PMD.

A method is more accurate the closer it's observed PMD is to the ground truth PMD [abs($PMD_{observed}$-$PMD_{ground\text{-}truth}$) should be lower, closer to zero]; overall, normalization methods yielded PMD values closer to ground truth than latent-dimension mapping

approaches(F=42.05,*P*=1.52e-43 main effects 1-way ANOVA; TukeyHSD:*P*<1e-3, normalization-methods lower compared to latent-dimensional mapping methods) (**Fig. 5b**). Similarly, log-ratio ln($PMD_{observed}/PMD_{ground\text{-}truth}$) should be near to zero; downsampling outperformed other methods (F=36.52,*P*= 3.68e-38 main effects,1-way ANOVA; TukeyHSD:*P*<0.006 downsampling-vs-all; downsampling closest to zero, and significantly different form other methods; **Fig. 5c**). Similar results were seen with $PMD_{observed}$-$PMD_{ground\text{-}truth}$ (F=37.8,P=1.56e-39 main effects, 1-way ANOVA; **Fig. 5d**). With $PMD_{observed}$-$PMD_{ground\text{-}truth}$, UMI downsampling was centered around zero, while others showed directional bias, making batches more similar (over-merging clusters: $PMD_{observed}$-$PMD_{ground\text{-}truth}$<0) or different (leaving residual batch effect: $PMD_{observed}$-$PMD_{ground\text{-}truth}$>0) compared to the ground truth (**Fig. 5c,d**). Other methods may therefore systematically result in differences in overall batch similarity, while UMI downsampling does not in these simulations.

We next sought to quantify whether batch correction algorithms merged two discrete ground-truth populations into a single cluster using the cluster purity metric. Fitting with our biologic figures of merit, latent-dimension mapping approaches tended to merge clusters when they were derived not only from a single cell-type source of variance, but were synthesized from a mixture of cell-types and -states (*P*<1e-7; 1-way ANOVA/TukeyHSD, **Fig. 5e**).

Relative mutual information measures clustering accuracy by quantifying the amount of information and structure in the clustering results relative to ground truth clusters. By this metric, normalization approaches significantly outperformed latent-dimension mapping approaches (F=246.6, *P*=2.13e-165 main effects 1-way ANOVA; TukeyHSD:*P*<1e-7 normalization vs latent-dimension mapping; **Fig. 5f**).

Flipping the cluster purity equation (here called ‘reverse purity’) allows one to quantify if a single ground truth population was erroneously split into several groups. We refined this metric to quantify specifically the cellular populations that were split based on batch (see **Methods**). This metric found a notable range in batch removal efficacy (F=41.9, *P*=2.81e-40, 1-

way ANOVA); downsampling and Harmony showed the highest reverse-purity across batch, almost universally removing the batch effect; these methods were not significantly different from each other (*P*=1.00, 1-way ANOVA/TukeyHSD post-hoc, **Fig. 5g**).

Simulation results: cell-type and -state specific errors in batch correction by simulation

We next sought to directly answer the question: do integration methods inappropriately erase batch confounded cell-state information? To this end, we included simulations that had a base cell-type (with no cell-state added) in batch 1, while in batch 2, we simulated the same base cell-type, but with an added cell-state. With clusters identified as the mixture of all expression programs (cell-lineage, cell-type, cell-state), these two clusters should be identified as separate. We therefore quantified the purity of these populations, to measure if clusters are inappropriately merged relative to the ground truth. Indeed, normalization approaches showed high purity in this context, while latent-dimension mapping approaches did not, indicating that normalization approaches preserve cell-state differences across batch (F=205.9, *P*=2.49e-78 main effects; *P*<1e-7 in all post-hoc comparisons; 1-way ANOVA/TukeyHSD; **Fig. 5h**).

Lastly, we also quantified whether algorithms merged two different base cell-types, when they both expressed high levels of the same “state” genes. A biologic analogue would be two different base cell-types appearing in a batch confounded manner, but both having a high replication signal, being merged into a single population after correction. Normalization methods and Liger showed high purity in this context while Suerat3-MCCA and Harmony did not (F=73.95, *P*=8.31e-38 main effects; *P*<1e-7 in all post-hoc comparisons; 1-way ANOVA/TukeyHSD; **Fig. 5i**). These results mirror our biologic findings that latent-dimension mapping approaches may over-merge discrete cellular populations, which should be either split by cell-type (**Fig. 2, Extended Data Fig. 5**) or cell-state (**Fig. 3, Extended Data Fig. 6**).

Combined biologic and synthetic benchmark results

Overall, we found that UMI downsampling was the most robust method across all biologic and synthetic datasets–ranking the highest on average (**Fig. 4j**). Notably, if we examine the minimum rank across all benchmarks, all other methods tested here had low rank-performance in at least one category, suggesting that other algorithms can under-perform in specific contexts.

A Scalable Implementation of UMI downsampling

Having shown that downsampling appears to be the most robust method across biologic datasets and synthetic simulations with adverse cell-lineage/type/state confounding factors, we sought to enable its application to large datasets. We therefore used the recently released senescent Tabula Muris datasets to benchmark the speed and scalability of our pip installable python package 'bio_pyminer_norm' for UMI downsampling[23].

While 40,000 Bone marrow cells, held in memory using sparse matrix format, was downsampled in 150 seconds, it is also important for algorithms to scale beyond what can be held in memory. We therefore created a new implementation that downsampled the senescent Tabula Muris dataset (245,389 cells),[23] in hdf5 format, using less than 24 Gb of memory, on a spinning platter hard drive, with 12 processes, in 75 minutes. Tutorials using these and smaller datasets are available on the repository website: https://bitbucket.org/scottyler892/pyminer_norm.

Discussion

Batch correction algorithms have primarily, or exclusively, focused on the ability to *remove* differences between batches, without quantifying *retention* of real biologic variation. Here, we introduced PMD, a statistical metric/test that precisely quantifies the degree of overlap in cluster composition across batches; PMD is provably invariant to the number of clusters, is linear with similarity, robust to dataset size, or differences in batch sizes (**Extended Data Fig.**

8). These properties, for the first time, enabled the quantitative assessment of normalization/batch correction algorithms across the full spectrum of similarity. Furthermore, PMD can be used outside the context of benchmarking; for example providing a simple metric to quantify the overlap in cluster composition across patients could even be used to generate a patient-level pseudo-time like measure to plot disease progression throughout a cohort.

With biologic datasets, one cannot know *a priori*, which sources of variance are truly technical or biologic; therefore, a robust method of batch correction is needed that operates well across the spectrum from identical batches, to biologically real batch confounded variance. Our results from 6.2e6 cells from 690 real-datasets integrations, and 9.8e5 cells from 672 simulated integrations, suggest that the simple approach of UMI downsampling is the most robust of those tested here. Furthermore, our results indicate that latent-dimension mapping approaches can erase batch confounded biologically meaningful gene expression, and over-merge disparate cell populations. Highlighted by the fact that clusters composed of both intestinal epithelial cells and neurons were frequently observed when integrating intestinal and brain datasets, clustering results downstream of latent-dimension mapping should be considered as best matches across batch rather than truly similar populations.

Given that the selection of batch correction algorithm will determine the degree of overlap between datasets, this can be viewed as a manually selected hyperparameter that allows the user to directly determine the results. While downsampling has hyperparameters that will alter sensitivity to rare cell-types and low expression transcripts–it does not appear to strip batch-imbalanced biologically real variance, therefore making it a safe approach for UMI/count based technologies. While other depth normalization approaches are similar in this regard many others left residual batch effect both with biological and simulated data.

We have primarily focused on the impact of normalization/batch correction on clustering results, given that the majority of scRNAseq analyses rely and build on this important step. Until other approaches are found that are robust across the spectrum of batch similarities, our

primary recommendations are therefore to use a downsampled dataset for at least the clustering step; if latent-dimension mapping approaches are employed however, comparison of post-integration batch similarity by PMD with results from a downsampled version of the dataset can provide a useful check for concordance. Once clusters have been identified however, downstream analyses may be better served by other approaches with enhanced sensitivity, so long as cell-cell statistical independence is preserved without data leakage–a requirement for all frequentist statistics. Importantly however, latent dimension mapping approaches remain the only viable option when integrating across sequencing technologies or across species.

Lastly, we provided both the PMD statistics R package, and the python package for UMI downsampling that is highly parallelizable and can scale to hundreds of thousands of cells, without access to high performance computing; these packages should be broadly useful to the bioinformatics community.

Acknowledgements

Support for this work was provided by T32CA078207, K99HG011270, R01AI118833, U19AI136053, and R01AI147028.

Author Contributions

SRT conceived of and performed all analysis and wrote manuscript. EES wrote proof of PMD's cluster invariance property. SB and EES guided research and edited manuscript.

Competing Interests Statement

Authors declare no conflicts of interest.

Methods

Biologic datasets: 10x Genomics datasets

All datasets, baring those of intestinal origin were obtained from Chromium's publicly available datasets.

PBMC 5' gene expression:
(https://cf.10xgenomics.com/samples/cell-vdj/3.0.0/vdj_v1_mm_c57bl6_pbmc_5gex/vdj_v1_mm_c57bl6_pbmc_5gex_filtered_feature_bc_matrix.h5)[20]

1k neurons:
(https://cf.10xgenomics.com/samples/cell-exp/3.0.0/neuron_1k_v3/neuron_1k_v3_filtered_feature_bc_matrix.h5)[16]

10k neurons:
(https://cf.10xgenomics.com/samples/cell-exp/3.0.0/neuron_10k_v3/neuron_10k_v3_filtered_feature_bc_matrix.h5)[18]

Other biologic datasets

The intestinal datasets, which were aligned to the Ensembl reference, as with those from Chromium, were previously published.[19] Several datasets from the human cortical and substantia nigra[22] and the endometrium were also previously released[23]. However, all raw input datasets used here are re-distributed in the benchmarking repository for ease of replication.

Global dataset downsampling

For the benchmarks that included sub-sampled "child cells," this 50% transcript sub-sampling of the dataset was created using the pyminer_norm package (available by pip: python3 -m pip install bio-pyminer-norm) using the command line call:

```
python3 -m pyminer_norm.random_downsample_whole_dataset -i <dataset.tsv> -o
<dataset_ds50.tsv> -percent 0.5
```

Implementation of normalization approaches

No normalization

For no normalization, the datasets are simply concatenated by genes with no other modifications.

UMI downsampling

We created a new python package that is pip installable called bio_pyminer_norm to provide a parallelized and scalable implementation of UMI downsampling. As shown in **Extended Data Fig. 1**, this approach requires the user to determine lower (and optionally upper) bounds of the sum of the UMI counts and total number of observed genes for which cells will be included or discarded. Next, each cell has its transcriptome simulated, shuffled and an equal number of transcripts randomly sampled. Unless otherwise specified, this is downsampled to the lower cutoff value for total sum UMI.

Scater-RLE

Following guidance from the Scater repository, we implemented this normalization approach using the following call: scater::normalize(exprs, exprs_values = "counts", return_log = TRUE, log_exprs_offset = NULL, centre_size_factors = TRUE, preserve_zeroes = FALSE).[13]

Combat-seq Negative Binomial

Combat-seq with Negative binomial distribution is distributed in the sva-devel package. Following author suggested tutorial, we used the following implementation: ComBat_seq(exprs, as.character(batch), group=NULL, full_mod=FALSE).[14]

Seurat-v3 Multi-CCA

Following the SeuratV3 website tutorial on integration, we first normalized and processed each dataset using: NormalizeData then FindVariableFeatures. Next, we used the functions: FindIntegrationAnchors and IntegrateData to complete the MCCA data integration in SeuratV3.[4]

Harmony

To use Harmony, we followed the author guidelines available on the repository website, first normalizing counts to counts per thousand, then log2 transforming the matrix, next genes were min-max scaled to range between 0 and 1. We used the following call: HarmonyMatrix(exprs, batch_table, "batch", do_pca = TRUE). As noted in the methods section on clustering, because this method does not yield a corrected expression matrix, but rather only the lower dimensional embedding, this embedding is used for clustering without additional feature selection.[5]

Liger

Following the Liger repository tutorial, we used the createLiger function on the input list of datasets to be normalized; next we utilized the liger::normalize function, followed by the selectGenes, and scaleNotCenter functions.

As per author guidance on their repository tutorial, we used the Liger suggestK function to generate a K vs median KL divergence curve. They authors suggest using the ‘elbow-rule’ of this curve, however, given that there is no actual relationship between the units of number of clusters and KL-divergence, we implement a K selection method that sums the area under that curve (AUC), and select the K that first reaches 95% of this AUC. This K was used for the

optimizeALS function. Lastly, we used the quantileAlignSNF function to complete Liger based batch correction.[6]

Similar to Harmony, Liger does not yield a corrected expression matrix, but rather, it yields a latent-dimension representation of the merged datasets. We therefore again forewent feature selection, and used this representation for clustering directly.

PMD metric definition

Definitions

- $\mathbf{O}_{actual}$: The observed contingency table $\mathbf{O}_{actual}$, is tabulated with dimensions *m* by *n* where *m* = number of identified clusters, *n* = number of batches.
- $\mathbf{E}_{actual}$: is the expected matrix, under the hypothetical scenario where the distribution of clusters is the same across batches. This *m* by *n* matrix is tabulated in the same manner as the "expected matrix" from a χ^2 test-of-independence.
- $\mathbf{O}_{max}$: A *n* by *n* matrix, whose eye is populated by each corresponding batch's number of cells. For each batch (*i*) with *i* = 0..*n*, $\mathbf{O}_{max}[\, i,i \,]$ = batch_size[*i*]; while all other entries in the matrix are 0. This represents the theoretic maximum level of asymmetry across batches, under the hypothetical that each batch was comprised of a single cluster that contained all of that batch's cells.
- $\mathbf{E}_{max}$: Similar to the calculation of $\mathbf{E}_{actual}$ with respect to $\mathbf{O}_{actual}$, this *n* x *n* "expected matrix", but as calculated from $\mathbf{O}_{max}$.

Using the observed and expected contingency matrices (similar to χ^2 test-of-independence), we found that the sum of the absolute value of the difference between expected and observed matrices were equivalent, regardless of the size of *m*, the number of clusters observed within each batch ($\Sigma|\mathbf{O}_{actual} - \mathbf{E}_{actual}|$), so long as relative cluster composition by batch

was conserved (**Extended Data Fig. 3a**). A proof that the PMD metric has the property of invariance to the number of clusters is provided in a subsequent section.

This equivalence property allowed the creation of a hypothetical contingency table, in which the entirety of each dataset fell into a single cluster that was specific for this batch ($\mathbf{O}_{max}$). $\mathbf{O}_{max}$ can be thought of as, what a contingency matrix would look like, given no overlap in cluster composition by batch (given the input batch sizes), with only a single cluster per batch. Thus creating a hypothetical contingency matrix that maximizes the possible asymmetry by batch, yet mirrors batch sizes of the input datasets. Because there is no dependence on the number of clusters within batch on the outcome of $\Sigma|\mathbf{O} - \mathbf{E}|$ (**Extended Data Fig. 3a**), we can calculate this for the maximum asymmetric hypothetical contingency matrix ($\Sigma|\mathbf{O}_{max} - \mathbf{E}_{max}|$) as well, and take the ratio of the actual difference relative to maximum possible (**Extended Data Fig. 3b**):

$$\text{Raw PMD} = \Sigma|\mathbf{O}_{actual} - \mathbf{E}_{actual}| \,/\, \Sigma|\mathbf{O}_{max} - \mathbf{E}_{max}|$$

This yields the Raw PMD metric, that is bounded between 0 and 1, that quantifies how similar cluster composition is across batches relative to the version of the input that maximizes the possible asymmetry by batch. A result that equals 0 indicates that the input ($\mathbf{O}_{actual}$) showed batches where cluster abundances were *exactly* equivalent ($\mathbf{O}_{actual} \approx \mathbf{E}_{actual}$; **Extended Data Fig. 3c**). Note that because of random Poisson sampling, it is likely that a real-world dataset would only *approach* zero, but that there would be fluctuation above zero even when there was no real difference – simply attributable to random sampling.

When no clusters appear in more than one batch, a dataset would maximize the Raw PMD value at 1 (**Extended Data Fig. 3d**). Because of the previously mentioned equivalency of $\Sigma|\mathbf{O} - \mathbf{E}|$ regardless of the number of clusters, $\Sigma|\mathbf{O}_{actual} - \mathbf{E}_{actual}|$ will be equivalent to $\Sigma|\mathbf{O}_{max} - \mathbf{E}_{max}|$, thus making the ratio equal to 1. Unlike the lower bound however, Poisson sampling under this scenario is not expected to yield a distribution because all clusters have a 0 percent chance of appearing in multiple batches.

Characterization of the Raw PMD metric and creation of the final PMD metric

To characterize the linearity and other properties of the Raw PMD function (**Extended Data Fig. 3**), we simulated ten iterations, each with two batches containing 10 clusters each, with variable numbers of cells in each batch as noted in the legend of **Extended Data Fig. 3e**. In combination with varying the number of cells per batch, we also incrementally changed the amount of overlap in the two batches. This was done by decreasing the number clusters that were in batch 1 that also appeared in batch 2. In all simulations, each batch still maintained 10 clusters, but progressively increasing the number of clusters that were specific to a given dataset. The most dissimilar scenario therefore contained a total of 20 clusters (2 batches, 10 clusters each that were all batch specific). The most similar scenario contained a total of 10 clusters (2 batches, 10 clusters each that were all shared across both batches evenly). This characterization is implemented in the do_full_pmd_characterization() function in the PercentMaxDiff R package that we released (https://github.com/scottyler89/PercentMaxDiff).

In all simulated cases, the relationship between the number of non-shared clusters was linear with the Raw PMD value (**Extended Data Fig. 3e**) as calculated with the equation in **Extended Data Fig. 3b**. Interestingly however, the slope and intercept of these lines appeared to depend on the number of cells in the datasets. We plotted histograms (**Extended Data Fig. 3f**) to characterize the distributions of Raw PMD at the intercept of each line in **Extended Data Fig. 3e**; this showed that when there was no actual difference in underlying cluster abundance (2 batches, 10 clusters total, all with equal relative abundance), the distributions of Raw PMD appeared to be Poisson distributed. This fits with the underlying discrete sampling errors that generated the cluster by batch contingency tables.

Importantly, the peak of these distributions (lambda parameter of the Poisson function) appeared to be dependent on the input dataset sizes, shown in different colors in **Extended Data Fig. 3f**. This indicates that the Raw PMD function, while being bounded between 0 and 1 and equivalent under *idealized* circumstances regardless of number of cells or clusters, it is

affected by random Poisson sampling. This is particularly true when underpowered, as noted by the higher Raw PMD when comparing two batches with only 100 cells each, but 10 cell types. At this point, the background noise from Poisson sampling increases the observed Raw PMD, even when the underlying distributions in cluster abundance across batch are equivalent (**Extended Data Fig. 3f**).

An ideal metric would allow all input datasets, regardless of number of clusters or number of cells, to scale linearly with the degree of overlap between the input datasets. We therefore sought to better characterize any dependence of the Raw PMD on 1) the number of clusters and 2) the number of cells. Specifically, for each dataset in these simulations, we performed Poisson sampling Monte Carlo simulations under the hypothetical that there are no differences in relative abundance of clusters by batch (as with the Y-intercept of the lines shown in **Extended Data Fig. 3e**). This allows an emprical observation of the input dataset's Y-intercept under the "unpatterned by batch" scenario, while also capturing the effect of the input sample sizes and number of clusters given that these are maintained in the simulations.

This is similar to calculating the expected matrix, as with a χ^2 test-of-independence, but through simulation that matches the underlying cluster abundances and number of cells rather than just calculating the *idealized* expected matrix; this allowed the characterization of how Poisson sampling alters the Raw PMD when the underlying cluster distributions by batch are equivalent, but matching the input batch sizes. Using this background, we calculated the observed Raw PMD for each Monte Carlo simulation, under the hypothetical of no-pattern by batch, and fit the above described Poisson distributions, calculating the lambda parameter using the fitdistr function from the R library MASS.

Using 100 null datasets of each given input in the prior simulation (**Extended Data Fig. 3e**), we observed that the lambda parameter distributions showed a clear dependence on both the total number of clusters and the size of the input datasets **Extended Data Fig. 3g**. This can be intuitively understood as increasing the chance of random Poisson sampling away from

perfect equivalence between batches, simply due to random sampling. Notably, this lambda parameter background is the intercept of the lines shown in **Extended Data Fig. 3e**. Furthermore, due to the overall linearity of the shown lines, this intercept can be used to calculate the slope of the lines in **Extended Data Fig. 3e** as well, thus allowing a modification of the Raw PMD, that incorporates and corrects for the background distributions caused by random Poisson sampling after fitting the distribution and calculating the best fit lambda parameter (**Extended Data Fig. 3h**):

PMD = (Raw PMD – lambda) / (1 – lambda)

Indeed, applying this correction to Raw PMD, allowed the creation of the final PMD metric which shows no dependence on input size of the batches or total number of clusters (**Extended Data Fig. 3i**). Thus, the final PMD metric has an upper bound equal to 1, whose lower distribution is centered around 0, and scales 1-to-1 linearly with increasing batch asymmetry.

Proof of cluster invariance with PMD equation

To prove whether the PMD is invariant to the number of clusters, let $C_{i,j}$ denote the clusters from $i = 1..n$ detected (*n* clusters in total) across the batches B_j for $j = 1..m$ (m batches in total). Let $n(C_{i,j})$ denote the number of cells in cluster $C_{i,j}$, and $n(B_j) = \sum_i n(C_{i,j})$ (sum of the cell counts across all clusters identified in batch *j* – number of cells in the batch). Thus, the total number of cells over all batches and clusters is given by:

$$N = \sum_{i=1}^{n} \sum_{j=1}^{m} n(C_{i,j}) = \sum_{j=1}^{m} n(B_j)$$

With this nomenclature, the PMD can be written as:

$$PMD = \frac{\sum_{j=1}^{m} \sum_{i=1}^{n} \left| n(C_{i,j}) - E(C_{i,j}) \right|}{\sum_{j=1}^{m} \sum_{k=1}^{m} \left| n(B_{j,k}) - \frac{n(B_k)}{N} n(B_k) \right|}$$

$$= \frac{\sum_{j=1}^{m}\sum_{i=1}^{n}\left|n(C_{i,j}) - \frac{n(B_j)}{N}n(C_i)\right|}{\sum_{j=1}^{m}\sum_{k=1}^{m}\left|n(B_{j,k}) - \frac{n(B_k)}{N}n(B_k)\right|},$$

where $E(C_{i,j})$ is the expected cell count for batch *j* in cluster *i* under the assumption of no association between batch and cell counts in the clusters, $n(C_i) = \sum_{j=1}^{m} C_{i,j}$ (number of cells in cluster *i* across all batches), and $n(B_{j,k}) = I_{j,k} \cdot n(B_j)$ with $I_{j,k} = 1$ when $j = k$ and *0* otherwise.

For the denominator of the *PMD* all of the terms are independent of the clusters and so by definition is invariant with respect to the number of clusters. To show the numerator is also invariant to the number of clusters any given algorithm uses to represent a cluster identified by other algorithms under the condition that the proportion of cells from the different batches is preserved, without loss of generality, suppose that for cluster C_i we subdivide it into *l* subclusters:

$$C_{i_1}, \dots, C_{i_l}$$

with the condition

$$\frac{n(C_{i_k,j})}{\sum_{j=1}^{m} n(C_{i_k,j})} = \frac{n(C_{i_k,j})}{n(C_{i_k})} = \frac{n(C_{i,j})}{n(C_i)},$$

for $j = 1..m$ and $k = 1..l$. We then have that

$$n(C_{i_k,j}) = n(C_{i_k})\frac{n(C_{i,j})}{n(C_i)}.$$

It then follows that

$$\sum_{k=1}^{l}\left|n(C_{i_k,j}) - \frac{n(B_j)}{N}n(C_{i_k})\right| = \sum_{k=1}^{l}\left|n(C_{i_k})\frac{n(C_{i,j})}{n(C_i)} - \frac{n(B_j)}{N}n(C_{i_k})\right|$$

$$= \sum_{k=1}^{l}\left|\frac{n(C_{i_k})}{n(C_i)}\left(n(C_{i,j}) - \frac{n(B_j)}{N}n(C_i)\right)\right|$$

$$= \frac{\sum_{k=1}^{l} n(C_{i_k})}{n(C_i)}\sum_{k=1}^{l}\left|\left(n(C_{i,j}) - \frac{n(B_j)}{N}n(C_i)\right)\right|$$

$$= \sum_{k=1}^{l} \left| \left(n(C_{i,j}) - \frac{n(B_j)}{N} n(C_i) \right) \right|,$$

since $\sum_{k=1}^{l} n(C_{i_k}) = n(C_i)$. Thus we have that regardless of the number of clusters that subdivide any given cluster, so long as the proportion of cell counts across batches in the subdivided clusters is conserved, the *PMD* score will not change. Note also that subclusters in this context does not necessarily imply or require any hierarchical lineage relationships between clusters, but rather are simply discussing potential subdivisions of a dataset.

Characterization of PMD, χ^2 statistic, χ^2 statistic -log10(p-value), and Cramer's V at the extremes of similarity and difference between batches

Cramer's V statistic is a previously published approach that builds on χ^2 to create a correlation-like metric based on counts of discrete classes in a manner similar to PMD.[25] A prior report indicated that this statistic demonstrated a high level of correctable bias, especially with contingency tables of dimension greater than 2x2, as will usually be the case in a cluster x batch contingency table.[26] However, in initial testing, we found that the bias corrected Cramer's V was still non-linear across the spectrum of similarity.

To demonstrate specific scenarios in which the PMD metric provides an advantage over other metrics such as the χ^2 statistic, its -log10(p-value), and bias corrected Cramer's V (as implemented in rcompanion[27]), we designed a series of comparisons of all four metrics under different scenarios. An ideal metric would have all of the following characteristics:

1. **Robustness to differences in power**: An ideal metric would be unaffected by the total number of cells in the batches. For example, two datasets of equal size (1,000 cells each) that were sampled from the same distribution of cell clusters, should be identified as equivalent to each other. The returned metric should not be different if these datasets

were 5,000 cells each, but equivalently identical. This allows direct interpretation on the degree of overlap between batches, regardless of their size.

2. **Robustness to differences in batch size**: An ideal metric for comparing how similar or different two batches are to each other in terms of their cluster composition, would be robust to differences in the size of two batches. For example, in the datasets used here, the dataset of 1,000 brain cells integrated with 10,000 brain cells should yield the same result as an integration of 1,000 brain cells with another dataset of 1,000 brain cells – in both cases, the desired metric would indicate that the batches come from the same cluster composition – unaffected by differences in batch sizes.
3. **Invariance to the number of clusters found within a given batch**: Batch correction algorithms are expected to leave different levels of within-batch variance as well as across batch variance, yet, with real-world datasets, we do not know the ground truth number of clusters. However, when integrating datasets that came from the same single cell suspension, we know that cluster composition should be the same. Additionally, we can make reasonable assumptions that intestinal epithelial cells should not be found in the same clusters as neuronal cells. However, we cannot claim to know the ground truth number of clusters within these datasets. To be able to benchmark batch correction algorithms, which leave different levels of within-batch variance, without making assumptions on the number of clusters, a metric is required that would report the degree to which cluster composition was similar across batches, while still allowing the number of clusters within a batch to vary, as this is determined by unbiased algorithms. For example, if two batch correction algorithms were used, on intestine and brain datasets, and both found no batch overlap, but algorithm one found three brain derived clusters, while the other found 4 brain derived cell clusters, an ideal metric would indicate that in both cases, the methods were equivalent in yielding no batch overlap. As formally

proven above, PMD is invariant to the number of clusters found, however, we will also show this with Monte Carlo simulations (**Extended Data Fig. 8**).

4. **Linearity across the spectrum of batch similarity:** For two batch correction algorithms to be fairly compared to each other, there must be a consistent mathematical relationship between the degree to which batches are similar/dissimilar, and the reported metric. In the particular case where the relationship between the percentage of overlap in cluster composition and reported metric is linear, the difference between having 50% and 60% of cells derived from the same cluster composition would be the same difference in the reported metric as 80% to 90% of cells coming from shared cluster composition. This gives immediately interpretable results.

We therefore sought to benchmark PMD against the more traditional metrics such as χ^2 statistic, its -log10(p-value), and the less frequently used χ^2 statistic derived bias corrected Cramer's V, under different idealized or adverse situations to assess for all of these properties. In each case, we calculated these metrics for a simulation of integrating two different batches, with differing numbers of clusters, and differing degrees of overlap between batches.

Importantly, what it means to have a given percentage overlap in cluster composition can come in two different modes. The first mode, and our first simulation paradigm (**Extended Data Fig. 8a**), is by sharing different numbers of clusters. For example, with two batches of four equally abundant clusters each could share no clusters (eight clusters total, four for each batch). They could share one out of the four clusters that appear in each batch (25% similar composition, 75% non-overlapping composition), etc. The second mode by which two batches can have varying degrees of overlap in cluster composition, is by sharing all, or some clusters, but having different relative abundance of the more-shared or less-shared clusters (**Extended Data Fig. 8b**). In our implementation of this second paradigm, we allocated a single cluster to batch 1, and vary the relative abundance of this cluster in batch 2, while batch 2's remaining

clusters are unique to it. In the case of 0% overlap, 0% of the second batch's cells were allocated to cluster #1 (the cluster that harbors all of the cells from batch 1), and 100% of batch2's cells are allocated to clusters number 2 to n with equal probability. In the case of 25% overlap, 25% of batch 2's cells are allocated to the first cluster, while the remaining 75% are distributed evenly across the remaining batch 2 specific clusters. Similarly, in the case of 50% overlap, 50% of batch 2 is allocated to cluster #1, and the remaining 50% of cells are allocated evenly to clusters unique to batch2, etc. Note that in this second simulation paradigm, when there is 100% overlap between batches 1 and 2, each cell from batch 2 has a 100% probability of being allocated to cluster #1, thus being equivalent to having only a single cluster across both batches.

We utilized both of these simulation paradigms (**Extended Data Fig. 8a,b**) to benchmark PMD, the χ^2 test-of-independence statistic, its corresponding -log10(p-value), and bias corrected Cramer's V to evaluate each of these metric's adherence to the above listed desirable traits of an ideal metric for comparing cluster composition after batch correction, integration and unsupervised clustering.

For our first Monte Carlo simulation in this comparison, we created two fully equivalent batches, with equal abundance (1000 cells each), and equal probabilities of sampling each cluster. We further simulated equivalent batches that harbored differing number of clusters (still all equivalent across batch, such that there was no pattern of clusters across batch). As expected, PMD was robust to changes in dimensionality of the input matrix; however, with increasing number of clusters, the χ^2 statistic rose linearly the number of clusters in the batches (**Extended Data Fig. 8c**), thus demonstrating that the χ^2 statistic gives linear background signal when batches are equivalent, increasing with the number of clusters in each batch. Similarly, Cramer's V showed a slight trend upwards. This is likely a phenomenon similar to the Raw PMD metric with a loss of power due to random sampling. Note that this pattern goes away when

there are only a single cluster simulated (**Extended Data Fig. 8d**). Additionally, in the special case of one cluster, Cramer's V cannot compute as it requires ≥2x2 contingency table.

Next, we performed a Monte Carlo simulation in which batches were completely orthogonal in their cluster composition, with no overlap; still simulating a variable number of clusters within each batch, but with no overlap (**Extended Data Fig. 8a**; 100% different), again equivalent in batch sizes. As expected, because of the invariance property, PMD was robust to different numbers of clusters, when there was no overlap in cluster composition between batches (**Extended Data Fig. 8e**). The χ^2 statistic however, lacked the invariance property, particularly when moving from 1 cluster per batch to 4 clusters per batch, but then plateaued. The -log10(p-value of the χ^2 test-of-independence however bottomed out with the lowest possible p-value due to floating capacity errors at -log10(p-value) = ~308. Interestingly, Cramer's V was negatively correlated with the number of clusters despite batches being completely different in each circumstance. The same pattern of results was observed under the second simulated paradigm (**Extended Data Fig. 8f**).

We next assessed the effect of differences in batch sizes, where one batch was larger than the other. When cluster composition was the same, PMD was centered around zero, regardless of the number of clusters or differences in batch size (**Extended Data Fig. 8g**), whereas the χ^2 statistic was highly significant with one cluster, but dropped to non-significant with more than one cluster. The degree to which the one cluster results were significant by the χ^2 statistic was dependent on the dataset sizes as shown by the difference between dotted and solid lines (**Extended Data Fig. 8g**). Similarly to **Extended Data Fig. 8c** Cramer's V showed some level of background above zero likely due to sampling error, as with Raw PMD.

Under the second simulation paradigm, where the number of clusters was only expanded in the second batch, this simulation was equivalent to having only a single cluster in total that held all cells from both batches. Consistent with the prior simulation, PMD was 0 (indicating no difference), while the χ^2 statistic and its corresponding p-values indicated highly

significant differences by batch in a manner dependent on batch sizes (**Extended Data Fig. 8h**). Again, in the special case of a single cluster present, Cramer's V fails to compute.

We next assessed a scenario similar to the above (**Extended Data Fig. 8g,h**), but now with no overlap in cluster composition. PMD gave consistent results with PMD=1 (completely different), regardless of input batch sizes (**Extended Data Fig. 8i**). The χ^2 statistic however was fully dependent on the size of the input datasets. This indicates that differences in cell-filtering through variable QC pipelines, or other sources will directly impact the ability to use the χ^2 statistic or its -log10(p-value) for comparisons. Cramer's V was dependent on both the sizes of the datasets and the number of clusters, despite the composition of clusters across batches being completely different in all cases. Furthermore, the invariance property of PMD and robustness to input dataset sizes demonstrate that it can be used, even to compare completely different dataset integrations, such as brain-heart concordance, relative to intestine-PMBC concordance. The same results were seen under the second simulation paradigm (**Extended Data Fig. 8j**).

Quantifying the linearity of PMD, χ^2 statistic, and χ^2 statistic -log10(p-value), and Cramer's V across the spectrum of similarity

We sought to quantify the linearity of these metrics, at intermediate levels of batch similarity, rather than only at the extreme poles (completely different and identical cluster composition). Linearity is important because it will enable the satisfaction of parametric frequentist statistics assumptions, that the difference between 0 and 0.5 be equivalent to the difference between 0.5 and 1.0. In other words, the property of linearity in relation to the underlying overlap in cluster composition should produce Gaussian distributions in the estimates of overlap when the underlying ground truth overlap is the same.

To test the property of linearity across the spectrum of similarity, we first created simulations in which batches were of equal size, with differing levels of overlap as determined

by presence or absence of shared clusters (**Extended Data Fig. 8a**). PMD gave consistent results, in which the PMD readout matched the simulated percentage of clusters that appeared in a batch specific manner (**Extended Data Fig. 8k**). For example, when 25% of clusters appeared in a batch specific manner, PMD was centered around 0.25, etc (**Extended Data Fig. 8k**, upper left panel). This concordance manifested in full linearity across the spectrum of overlap under this simulation paradigm for PMD (**Extended Data Fig. 8k**, lower left panel). Interestingly, this pattern held true for the χ^2 statistic as well (**Extended Data Fig. 8k**, middle-left panels). However, the χ^2 -log10(p-value) was non-linear, and was inversely proportional to the number of clusters (**Extended Data Fig. 8k**, middle-right panels). Similarly, Cramer's V was non-linear across the spectrum of similarity (**Extended Data Fig. 8k**, right panels). Importantly, however, these patterns changed under our second simulation paradigm.

Using the second simulation paradigm in which the percent overlap was determined by the relative abundance of a single potentially shared cluster, we found that again PMD was robust to the number of clusters and linear across the spectrum of batch similarity (**Extended Data Fig. 8l**, left panels). The χ^2 statistic however showed an exponential pattern across the spectrum of similarity that was consistent regardless of the number of clusters (**Extended Data Fig. 8l**, middle-left panels) and the -log10(p-values) were slightly non-linear as well, in a manner dependent on the number of clusters (**Extended Data Fig. 8l**, middle-right lower panel). Under these circumstances, Cramer's V was largely linear (**Extended Data Fig. 8l**, right lower panel), however, as previously mentioned, the special case of one cluster fails to compute.

Next, we simulated a range of batch similarity, with identical batch sizes, but testing the effect of power under two scenarios – integrating two batches with 5,000 cells each (solid lines), compared to integrating two batches with 1,000 cells each (dashed lines). PMD was linear across the spectrum of similarity and robust to different levels of power under both simulation paradigms (**Extended Data Fig. 8m,n** left panels). Interestingly, the χ^2 statistic was again linear within the two simulated batch sizes (two batches, 5,000 cells each and two batches 1,000 cells

each), however, the slope of these lines was different, in which the simulated integrations with a larger number of cells had a greater slope compared to the smaller simulations (**Extended Data Fig. 8m**, lower middle-left panel). With the χ^2 test-of-independence -log10(p-value), we observed floating point errors, as the p-value approaches zero, that resulted in non-linearity with dataset similarity in the larger dataset (**Extended Data Fig. 8m**, middle-right panels). Cramer's V was robust to dataset sizes, but was non-linear across the spectrum of similarity (**Extended Data Fig. 8m**, right panels). Interestingly, under the second simulation paradigm, the χ^2 statistic was exponential in relation to batch similarity, instead of being linear when batches as with the first simulation paradigm (**Extended Data Fig. 8n**, lower middle panel). Under the second simulation paradigm, Cramer's V was robust to dataset size and linear, because batch sizes were identical, but failed under the special case of one cluster (**Extended Data Fig. 8n**, right panels). PMD, was again robust to dataset size, number of clusters, and linear across the spectrum of similarity (**Extended Data Fig. 8n**, left panels).

Lastly, we performed the same simulations as in (**Extended Data Fig. 8m,n**, but with datasets that were of different sizes within each integration (solid lines: batch1:1,000 cells, batch2:5,000 cells; dashed lines: batch1:500 cells, batch2:1,000 cells). PMD was again robust to differences in batch size, differences in power (as indicated by overlapping solid and dashed lines), was unaffected by the number of clusters, and operated linearly with batch similarity under both simulation paradigms, (**Extended Data Fig. 8o,p**, left panels). The χ^2 statistic and its -log10(p-value) operated linearly only under the first simulation paradigm, but in a manner that was dependent on the number of clusters as well as power, while again under the second simulation paradigm, the χ^2 statistic and its -log10(p-value) were non-linear with batch similarity, dependent on the number of cells, the difference in batch size, and in the case of -log10(p-value), decreased with increasing number of clusters (**Extended Data Fig. 8o,p**, middle-left and middle-right panels). Under the first simulation paradigm, Cramer's V was non-linear with percent of clusters that were unique to a given batch, but was robust to different dataset sizes

(**Extended Data Fig. 8o**, right panels). Under the second simulation paradigm, Cramer's V was both non-linear and dependent on dataset sizes under this simulation paradigm (**Extended Data Fig. 8p**, right panels). Surprisingly, the pattern of non-linearity in Cramer's V was different under the two simulation paradigms (**Extended Data Fig. 8o,p**, right-lower panels).

Overall, these results demonstrate that PMD is 1) invariant to the number of clusters, 2) robust to the size of input datasets, 3) robust to differences between batch sizes, and 4) operates linearly across the spectrum of similarity in batches when simulated as presence/absence of shared clusters across batches (first simulation paradigm) or relative abundance of a shared cluster (second simulation paradigm). The χ^2 statistic, its -log10(p-value), and Cramer's V did not consistently share these properties robustly to all simulated circumstances.

scRNAseq simulations

Simulating cell-types

Using Splatter,[11] we simulated 1500 cells with 10,000 genes using the parameters: "de.downProb"=.25, "de.prob"=.75, "de.facScale"=.75, "bcv.common"=.75. All other parameters were left at their default levels. Each cell was uniformly randomly assigned to one of 6 clusters, and to either batch 1 or batch 2, depending on the details of the simulation as noted in **Extended Data Fig. 7**.

Simulating cell-states

To simulate a cell-state module, a single 'cluster' was simulated with the same dimensions as the original dataset. All cells were then paired with a cell-state transcriptome. 25% of genes were then considered to be members of the cell-state. A gaussian weighting vector was generated using rnorm * 1/50 to shrink variance, and shifted to center around 0.25. Each gene belonging to the cell-state was assigned to a weight from this distribution. A linear

mixture of the cell transcriptome and states is then created to form the final cell, using weights for the mixture similar to above, mixing cell-type and state with random weights either centered at 33.3% state or a half-Gaussian whose lower bound was zero for cells that were being injected with the cell-state or not respectively.

Clustering

Feature selection prior to clustering

Automated highly variable gene selection, and clustering was applied equivalently to all methods that yield corrected transcriptomes, as previously reported by PyMINEr[28]. The mean variance relationship was fit by a lowess locally weighted regression curve. Residuals were calculated, and those genes whose residual was ≥2 standard deviations above the loess fit curve were selected. Harmony and Liger do not yield corrected transcriptomes, but rather latent-dimension representations of the data; therefore, these latent-dimension representations are used directly for clustering without feature selection.

Clustering

Clustering is performed on a local affinity matrix calculated as follows:

- Calculate the symmetric pairwise Spearman correlation matrix of each cell against all other cells
- Calculate the squared Euclidean distance of all cells to all other cells using the Spearman correlation matrix as input.
- For each cell, mask the 95% most dissimilar cells (or all but the 200 closest cells, whichever gives the lower number of connections), as well as the diagonal, self-distance=0, thus creating a local distance matrix.
- For all remaining cells that were not masked, the inverse Squared Euclidean distances were linearly normalized between 0 and 1 within each cell. This creates a local weighted

distance matrix, which is then converted to a weighted graph, treating the local distance matrix as a weighted adjacency matrix, which is then subjected to Louvain modularity based clustering.

This process is fully automated in PyMINEr,[28] hyperparameter free, and implemented equally to all batch correction methods, using the -louvain_clust argument. Louvain modularity was used via the python-louvain package.

Clustering performance metrics

The “percent-same” metric that was implemented in some biologic benchmarks is calculated by taking the pairs of cells from the original dataset, and its downsampled counterpart, and quantifying the number of pairs that were placed into the same cluster (numerator), relative to the total number of pairs (denominator). This metric only quantifies cells that are present in both datasets; if a parent or child cell was removed by a pipeline for quality control purposes, they were not counted given that it would not have a matching cell-pair to assess if the cluster was the same.

Mutual information was calculated using the mi.empirical function from the entropy R package. Relative mutual information is the ratio of the observed mutual information to the theoretic maximum mutual information, if the results were 100% accurate relative to the ground truth clusters.

Purity was calculated using the purity function from the NMF package in R.[29] The purity function takes in two arguments, first: the cluster results, and second: the ground truth clusters. Purity is a metric that takes the method’s clusters, identifies its plurality ground truth cluster, then returns the percent of all cells that fall into a method’s cluster whose plurality is the same ground truth cluster. This particularly penalizes for merging ground truth clusters together.

Reversing the arguments of the purity function does the opposite however, we therefore use reverse purity to quantify splitting ground truth clusters (reverse purity).

The “true percent max difference” is reported in **Extended Data Table 1**; this metric calculates the PMD based on the ground truth known clusters in the simulated batches. This metric is used in combination with the percent maximum difference calculated using the observed clustering results for each method. The log ratio percent maximum difference is $\ln(PMD_{observed}/PMD_{ground\text{-}truth})$. Difference in percent maximum difference is $PMD_{observed}-PMD_{ground\text{-}truth}$. Absolute difference in percent maximum difference is $|PMD_{observed}-PMD_{ground\text{-}truth}|$.

The “state merged with base purity across batch” is the typical purity calculation, but first filtered to include only cells from clusters for which the base cluster appears both with and without the state added, thus penalizing specifically for merging clusters that belong to the same base cluster with and without a state. This quantifies an algorithm’s propensity for erasing batch confounded ‘states’ or expression programs.

The “state merged with different base purity across batch” quantification is the purity function applied specifically to the filtered subset of ground truth clusters with the cell-state added that did not belong to the same base “cell-type.”

The “split by batch reverse purity” metric is the reverse purity metric but selectively applied to the subset of cells that belonged to ground truth clusters that appeared in both batches, whereas cell clusters that appeared only in a single batch were removed.

Relative abundance by batch as noted by stacked bar charts in **Extended Data Figs. 5,6** were calculated by first normalizing each dataset by percentage of the dataset that falls into the given cluster; this allows each dataset to appear as equivalent, normalizing for differences in total number of cells in the given datasets.

UMI downsampling of senescent Tabula Muris datasets

The bone marrow dataset was used as a large dataset (40,000 cells) that could still be held in memory in sparse format. This dataset was normalized using the function pyminer_norm.downsample.downsample.

The full senescent tabula muris dataset (245,389 cells) was too large to hold in memory (on a machine with 32 Gb); we therefore also implemented a completely out of memory implementation of UMI downsampling: pyminer_norm.downsample.downsample_out_of_memory. See tutorial at: https://bitbucket.org/scottyler892/pyminer_norm for usage details for these two examples and others.

Clock time was used for measuring the duration of the downsampling process.

Software versions

All R analyses were performed in R version 3.6.0 (2019-04-26) -- "Planting of a Tree". R packages require for analysis were: splatter: 1.8.0, scater: 1.12.2, Seurat: 3.1.2, DESeq2: 1.24.0, sva: 3.33.2, irlba: 2.3.3, liger: 0.4.2, harmony: 1.0, BiocParallel: 1.18.1, ggplot2: 3.3.0, reshape2: 1.4.3, pals: 1.6, tidyverse: 1.3.0, stringr: 1.4.0, scales: 1.1.0, entropy: 1.2.1, NMF: 0.22.0. Python was version 3.7.1.

Data Availability

All dataset inputs used in this manuscript are distributed in the data folder of the benchmark repository: https://bitbucket.org/scottyler892/sc_norm_bench_v1

Code Availability

All code for this benchmark is freely available at the following repository:

https://bitbucket.org/scottyler892/sc_norm_bench

The UMI downsampling package we created is available by pip installation:

python3 -m pip install bio-pyminer-norm

The UMI downsampling package repository, as well as 10x genomics dataset and tabula muris dataset tutorials is located here:

https://bitbucket.org/scottyler892/pyminer_norm/

The R implementation of PMD and associated functions/statistical tests are available here:

https://github.com/scottyler89/PercentMaxDiff

and can be installed via the devtools package:

devtools::install_github('scottyler89/PercentMaxDiff')

References

1. Stoeckius, M. et al. Cell Hashing with barcoded antibodies enables multiplexing and doublet detection for single cell genomics. *Genome Biology* **19**, 224 (2018).
2. Schaum, N. et al. Single-cell transcriptomics of 20 mouse organs creates a Tabula Muris. *Nature* **562**, 367-372 (2018).
3. Han, L. et al. Single-cell atlas of a non-human primate reveals new pathogenic mechanisms of COVID-19. *bioRxiv*, 2020.2004.2010.022103 (2020).
4. Stuart, T. et al. Comprehensive Integration of Single-Cell Data. *Cell* **177**, 1888-1902.e1821 (2019).
5. Korsunsky, I. et al. Fast, sensitive and accurate integration of single-cell data with Harmony. *Nature Methods* **16**, 1289-1296 (2019).
6. Welch, J.D. et al. Single-Cell Multi-omic Integration Compares and Contrasts Features of Brain Cell Identity. *Cell* **177**, 1873-1887.e1817 (2019).
7. Lopez, R., Regier, J., Cole, M.B., Jordan, M.I. & Yosef, N. Deep generative modeling for single-cell transcriptomics. *Nature methods* **15**, 1053-1058 (2018).
8. Polański, K. et al. BBKNN: fast batch alignment of single cell transcriptomes. *Bioinformatics* **36**, 964-965 (2019).
9. Bella, E.D. & Stoddart, M.J. Cell detachment rapidly induces changes in noncoding RNA expression in human mesenchymal stromal cells. *BioTechniques* **67**, 286-293 (2019).
10. O'Flanagan, C.H. et al. Dissociation of solid tumor tissues with cold active protease for single-cell RNA-seq minimizes conserved collagenase-associated stress responses. *Genome Biology* **20**, 210 (2019).
11. Zappia, L., Phipson, B. & Oshlack, A. Splatter: simulation of single-cell RNA sequencing data. *Genome Biology* **18**, 174 (2017).
12. Luecken, M. et al. Benchmarking atlas-level data integration in single-cell genomics. *bioRxiv*, 2020.2005.2022.111161 (2020).
13. 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).
14. Zhang, Y., Parmigiani, G. & Johnson, W.E. ComBat-Seq: batch effect adjustment for RNA-Seq count data. *bioRxiv*, 2020.2001.2013.904730 (2020).
15. Townes, F.W., Hicks, S.C., Aryee, M.J. & Irizarry, R.A. Feature selection and dimension reduction for single-cell RNA-Seq based on a multinomial model. *Genome Biology* **20**, 295 (2019).
16. Genomics, x., Vol. 2020 (https://support.10xgenomics.com/single-cell-gene-expression/datasets/3.0.0/neuron_1k_v3; 2019).
17. Büttner, M., Miao, Z., Wolf, F.A., Teichmann, S.A. & Theis, F.J. A test metric for assessing single-cell RNA-seq batch correction. *Nature methods* **16**, 43-49 (2019).
18. Genomics, x., Vol. 2020 (https://support.10xgenomics.com/single-cell-gene-expression/datasets/3.0.0/neuron_10k_v3?; 2019).
19. Ayyaz, A. et al. Single-cell transcriptomes of the regenerating intestine reveal a revival stem cell. *Nature* **569**, 121-125 (2019).
20. Genomics, x., Vol. 2020 (https://support.10xgenomics.com/single-cell-vdj/datasets/3.0.0/vdj_v1_mm_c57bl6_pbmc_5gex; 2019).
21. Ginhoux, F. et al. Fate Mapping Analysis Reveals That Adult Microglia Derive from Primitive Macrophages. *Science* **330**, 841-845 (2010).
22. Agarwal, D. et al. A single-cell atlas of the human substantia nigra reveals cell-specific pathways associated with neurological disorders. *Nature Communications* **11**, 4183 (2020).

23. Almanzar, N. et al. A single-cell transcriptomic atlas characterizes ageing tissues in the mouse. *Nature* **583**, 590-595 (2020).
24. Svensson, V. Droplet scRNA-seq is not zero-inflated. *Nature Biotechnology* **38**, 147-150 (2020).
25. Cramir, H. Mathematical methods of statistics. *Princeton U. Press, Princeton* **500** (1946).
26. Bergsma, W. A bias-correction for Cramér's V and Tschuprow's T. *Journal of the Korean Statistical Society* **42**, 323-328 (2013).
27. Mangiafico, S. Package 'rcompanion'.
28. Tyler, S.R. et al. PyMINEr Finds Gene and Autocrine-Paracrine Networks from Human Islet scRNA-Seq. *Cell Reports* **26**, 1951-1964.e1958 (2019).
29. Gaujoux, R. & Seoighe, C. A flexible R package for nonnegative matrix factorization. *BMC Bioinformatics* **11**, 367 (2010).

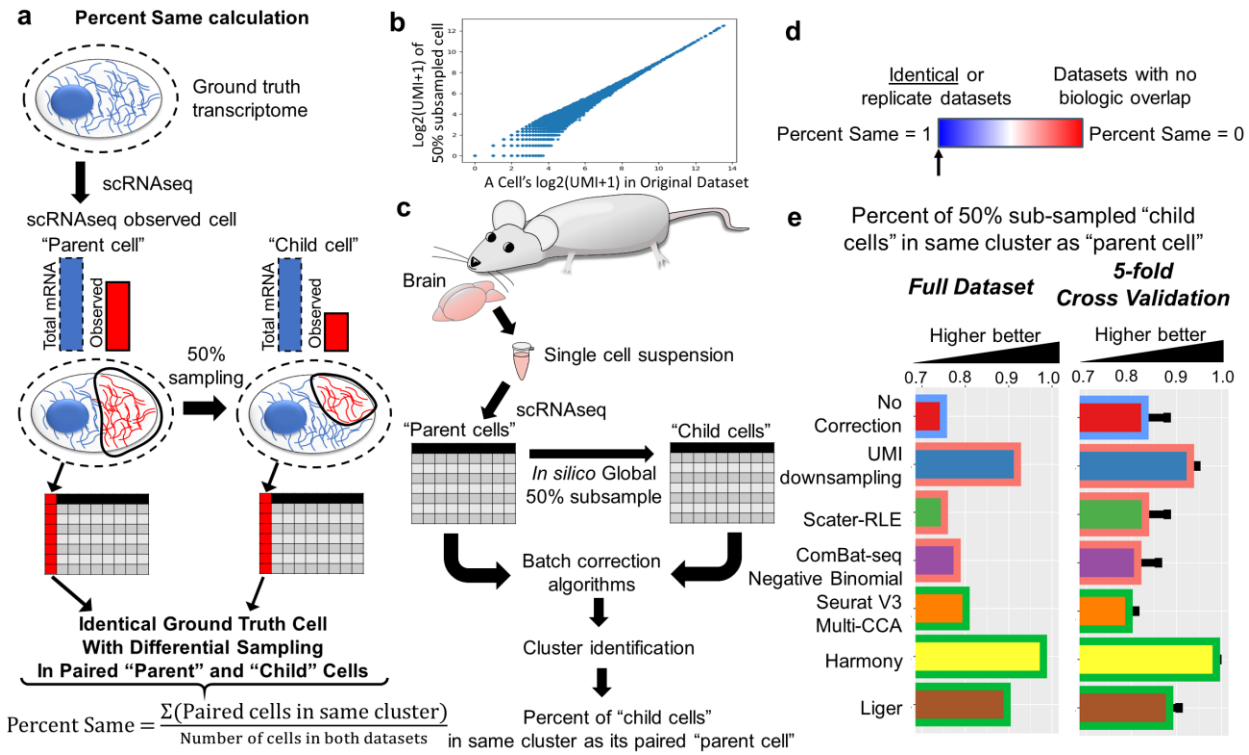


Figure 1: Performance in integrating identical datasets with differences only in UMI depth. a, Example schematic of the same cell with different transcript capture efficiency leading to different observed total UMI. **b**, Scatter plot showing a brain scRNAseq dataset that has been globally downsampled at random. Each point is a transcript's log2(UMI+1) measure in the parental cell (X-axis) and downsampled cell (Y-axis). **c**, Pipeline overview for assessing batch correction algorithm accuracy in the scenario where scRNAseq was performed on a mouse brain sample, and the dataset was downsampled at random as in **b** creating an identical version of the dataset that differs only in the total observed UMI, which in real-world datasets can range by orders of magnitude (**Extended Data Fig. 1b**). **d**, This scenario represents the greatest possible similarity between two datasets, where percent-same should equal 1. **e**, A bar chart displaying the percentage of downsampled cells that were classified as belonging to the same cluster as its parental cell. A higher score indicates better performance, with 1 indicating a perfect score, and zero indicating the worst possible score. Blue border is for no-correction, peach border is for depth normalization approaches, and green border is for latent-dimension

mapping approaches. Five-fold cross-validation of this integration task show highly concordant results (means +/- SD).

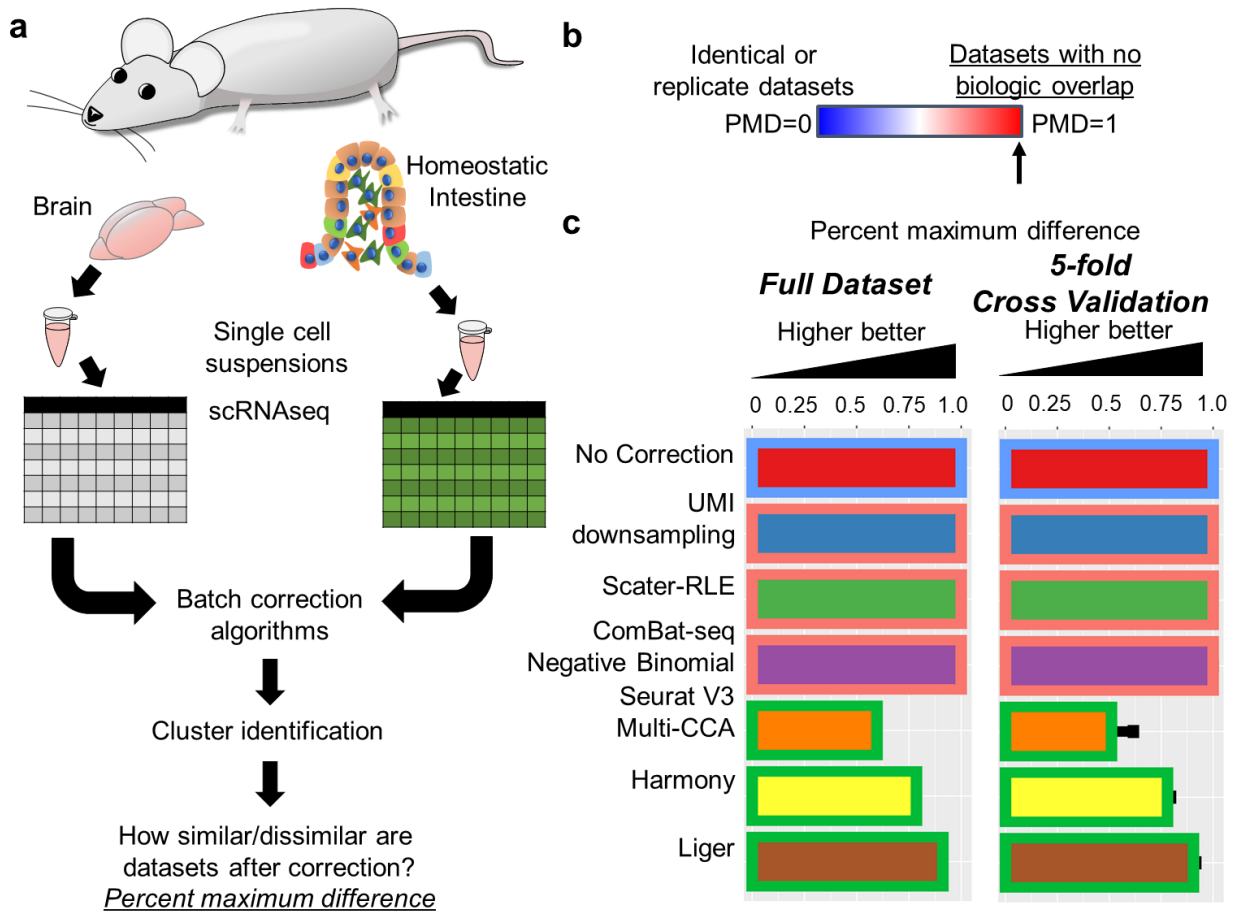


Figure 2: Batch correction algorithm performance when samples are expected to have no overlap. a, A schematic representation of the dataset integration task in which there is little to no overlap in cell-types expected – brain and intestine samples – quantified by the percent maximum difference (PMD). **b**, These datasets are not expected to share any clusters, therefore having an expected PMD≈1. **c**, A bar chart displaying the PMD between the input brain and intestine datasets. Because the datasets are expected to have no overlap in cell-types, a higher value indicates better performance. Depth normalization approaches and no correction showed no overlap between these datasets (PMD=1), whereas latent-dimension mapping approaches showed only between 60-90 percent difference relative to the maximum possible. Blue border indicates no correction, peach border represents depth normalization approaches, and green

borders represent latent-dimension mapping approaches. Five-fold cross-validation of this integration task show highly concordant results (means +/- SD).

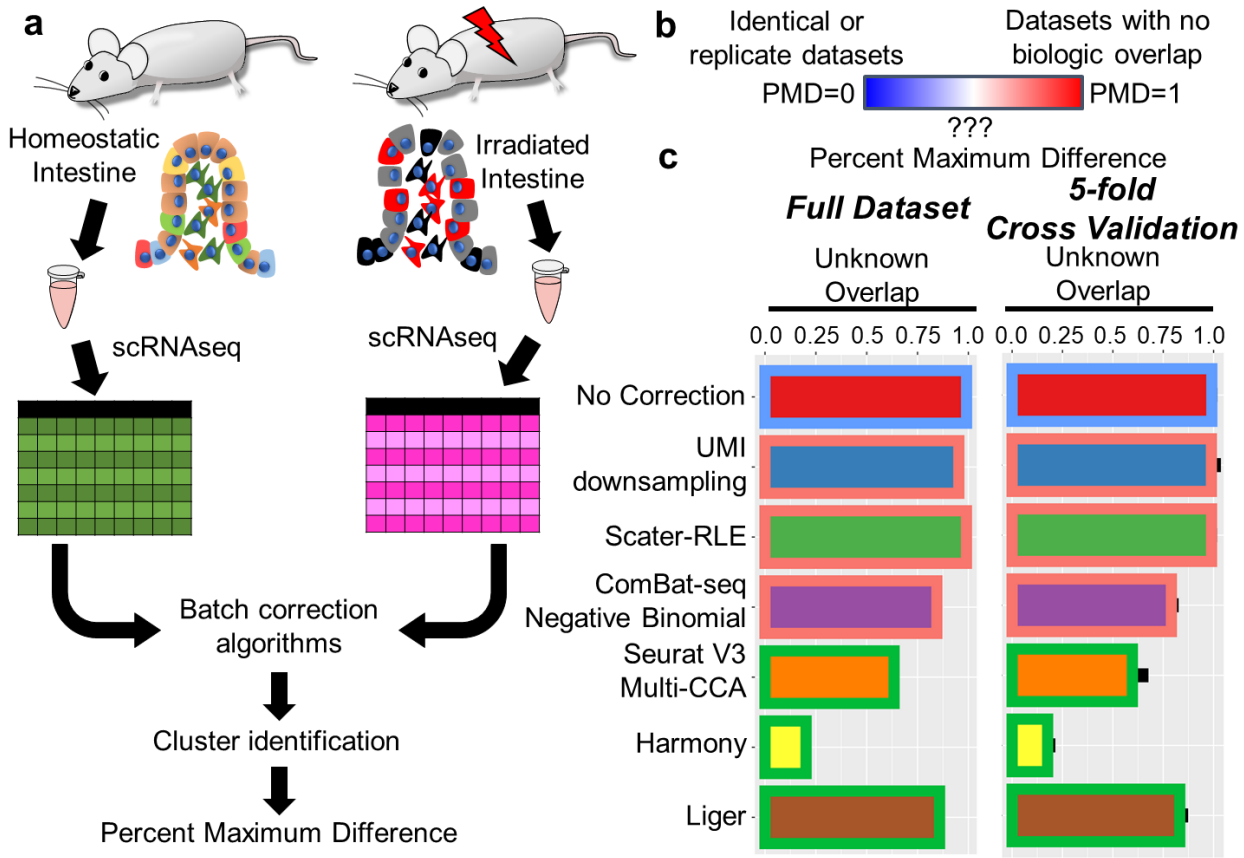


Figure 3: Widespread disagreement on dataset similarity based on batch correction algorithms. a, Using datasets from homeostatic and irradiated mouse intestines, we created a batch correction task with no clear expectation for similarity in clustering results[19]. **b**, Large scale transcriptional responses related to replication and cell death are expected, but the original base cell-types are the same, it was therefore unclear *a priori* how similar these datasets would be by PMD. **c**, Overall clustering results show a wide range of overall dataset similarity ranging from nearly completely different without batch correction and with depth normalization approaches to mostly the same (20.2% maximum difference) with Harmony. Five-fold cross-validation of this integration task show highly concordant results (means +/- SD).

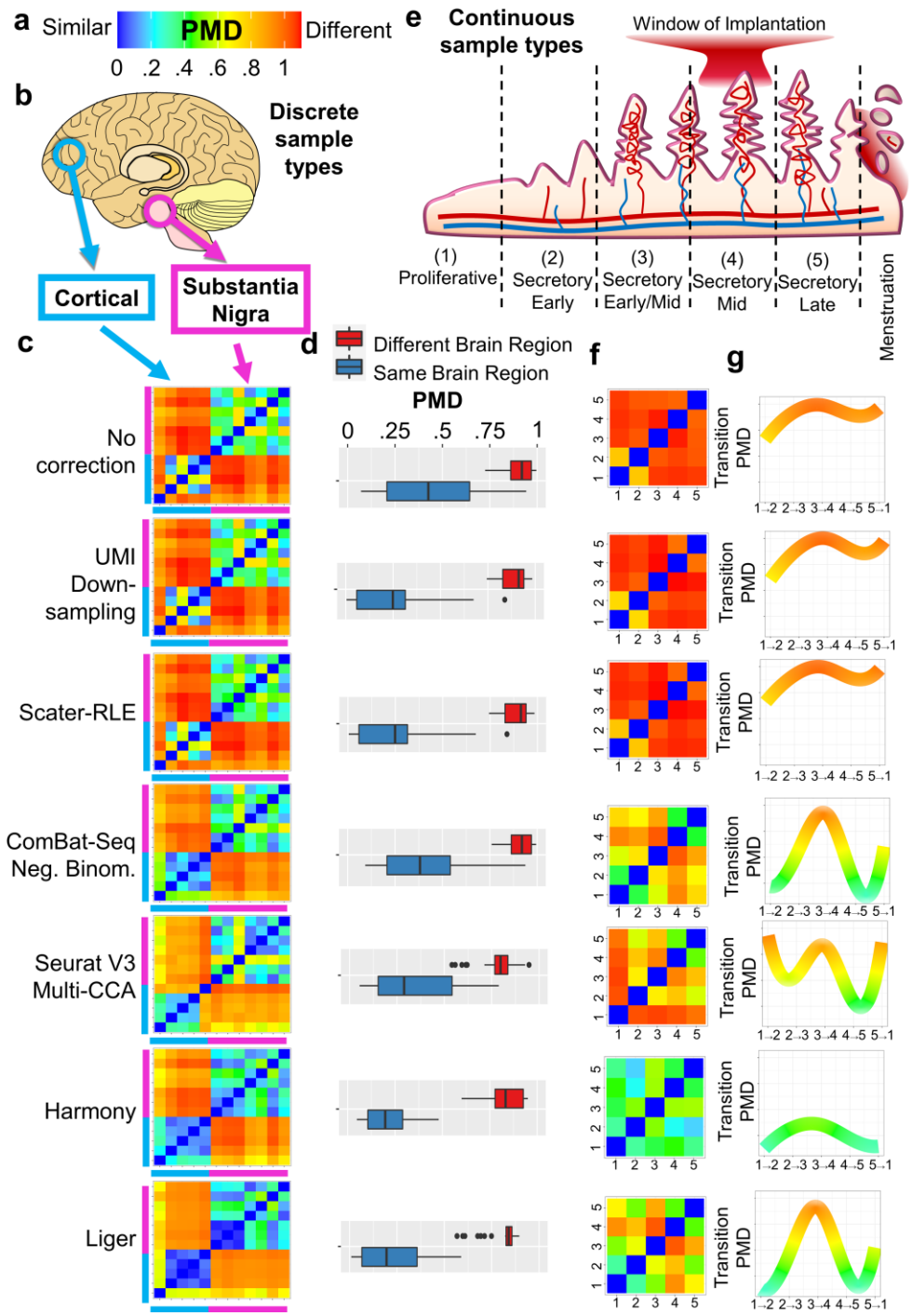


Figure 4: Batch correction of discrete or continuous sample types. a, For panels **b**,**c**,**f**,**g** PMD is color-coded with shorter-wavelengths corresponding to more similar samples. **b**, Samples from human cortical tissue were integrated with samples from the substantia nigra; these areas of the brain are largely comprised of the same cell types, but in very different

abundance. **c**, Heatmaps of the pairwise PMD across all batches show clear separation of the two sample-types in all cases, for all batch-correction methods. **d**, Boxplots of within-sample type, and across-sample-type PMD show similar results across all batch correction methods. **e**, Samples from the endometrium across the different stages of estrus were integrated to assess batch correction algorithm performance when samples exist along a continuum of cell-state and cell-type abundance changes. **e**, Pairwise heatmaps of each stage's PMD against all other stages. **f**, The PMD moving along the sequential transitions throughout the menstrual cycle; higher PMD indicates that the given transition involved greater remodeling.

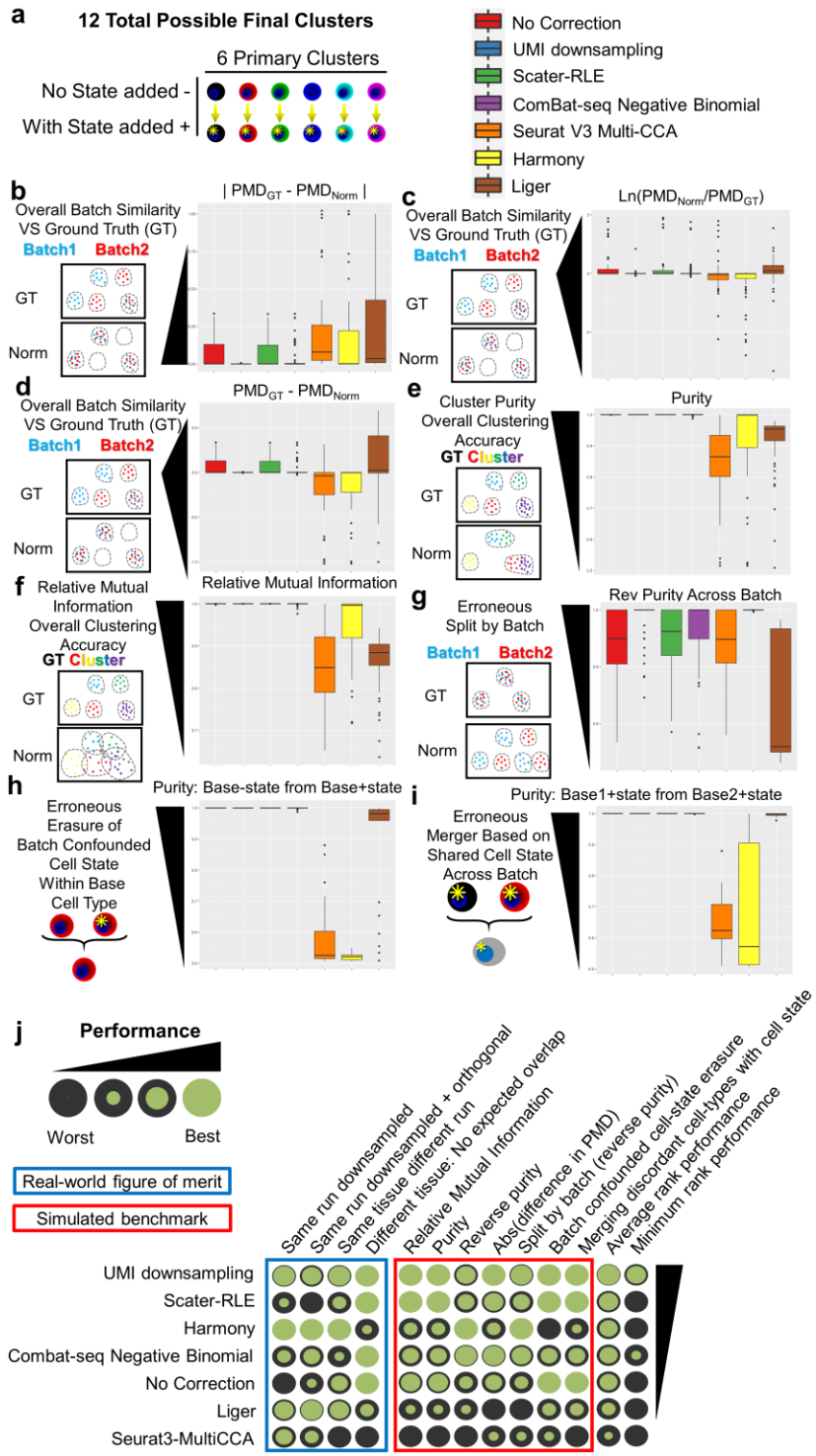
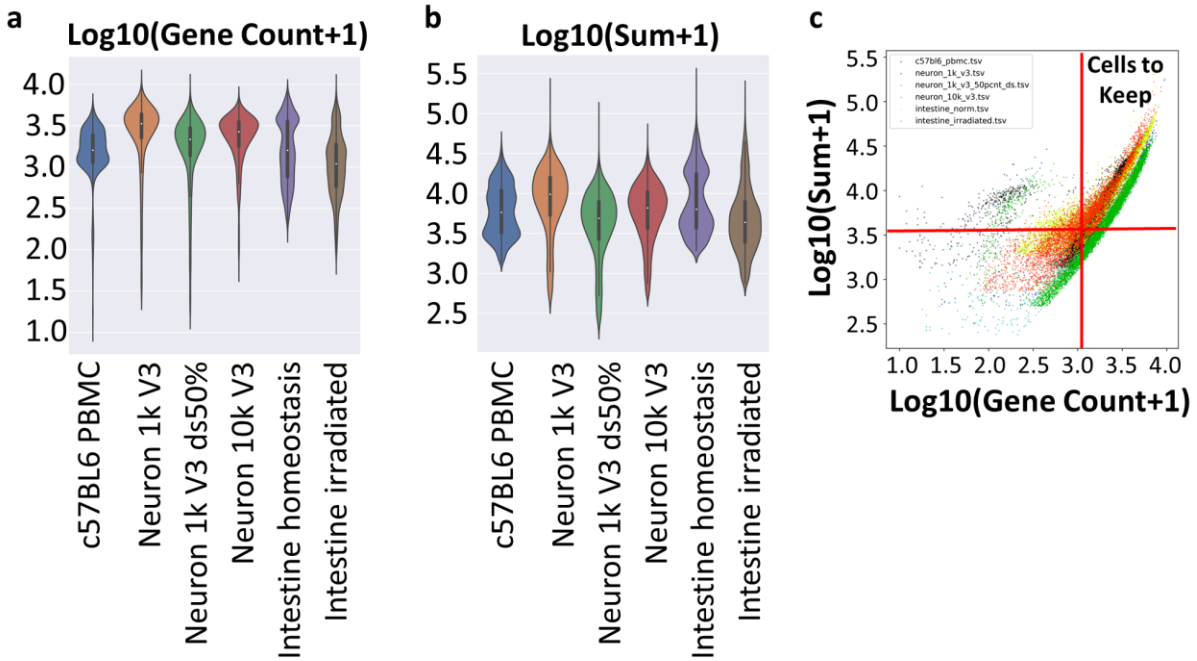
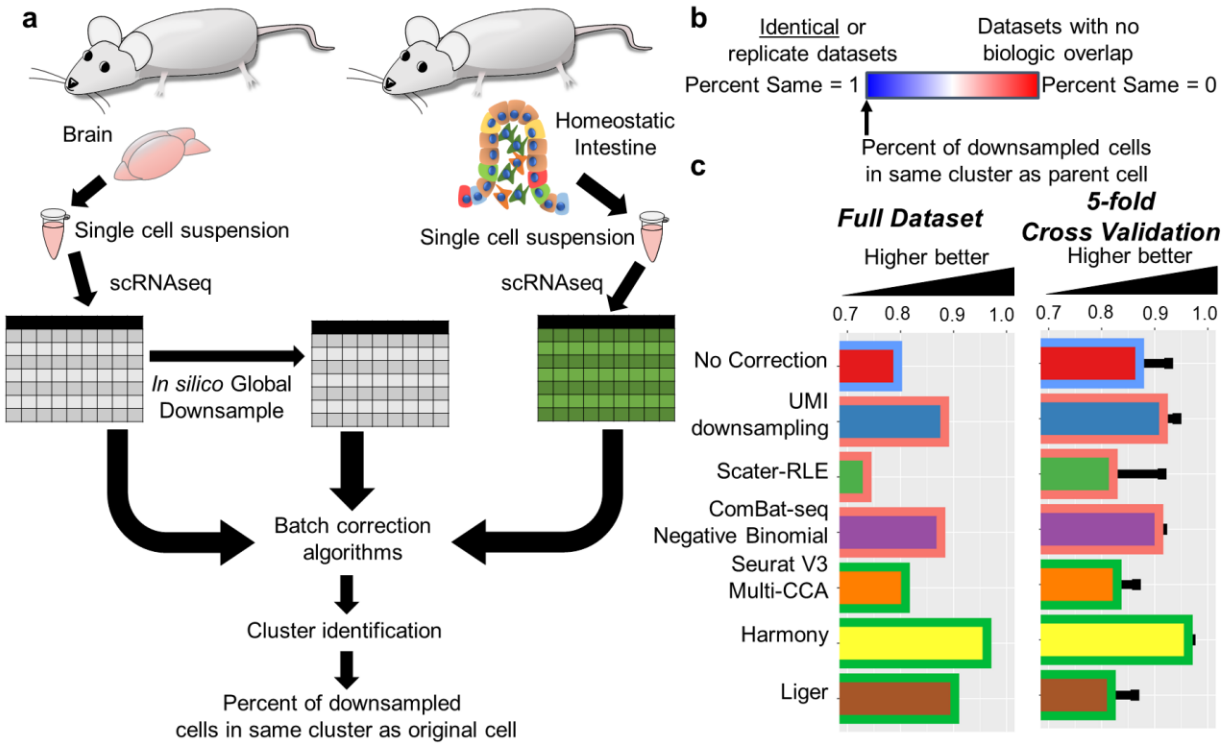


Figure 5: Simulating a mixture of cell-types/states shows that downsampling is the most robust approach across 32 different simulation conditions. a, Twelve possible clusters

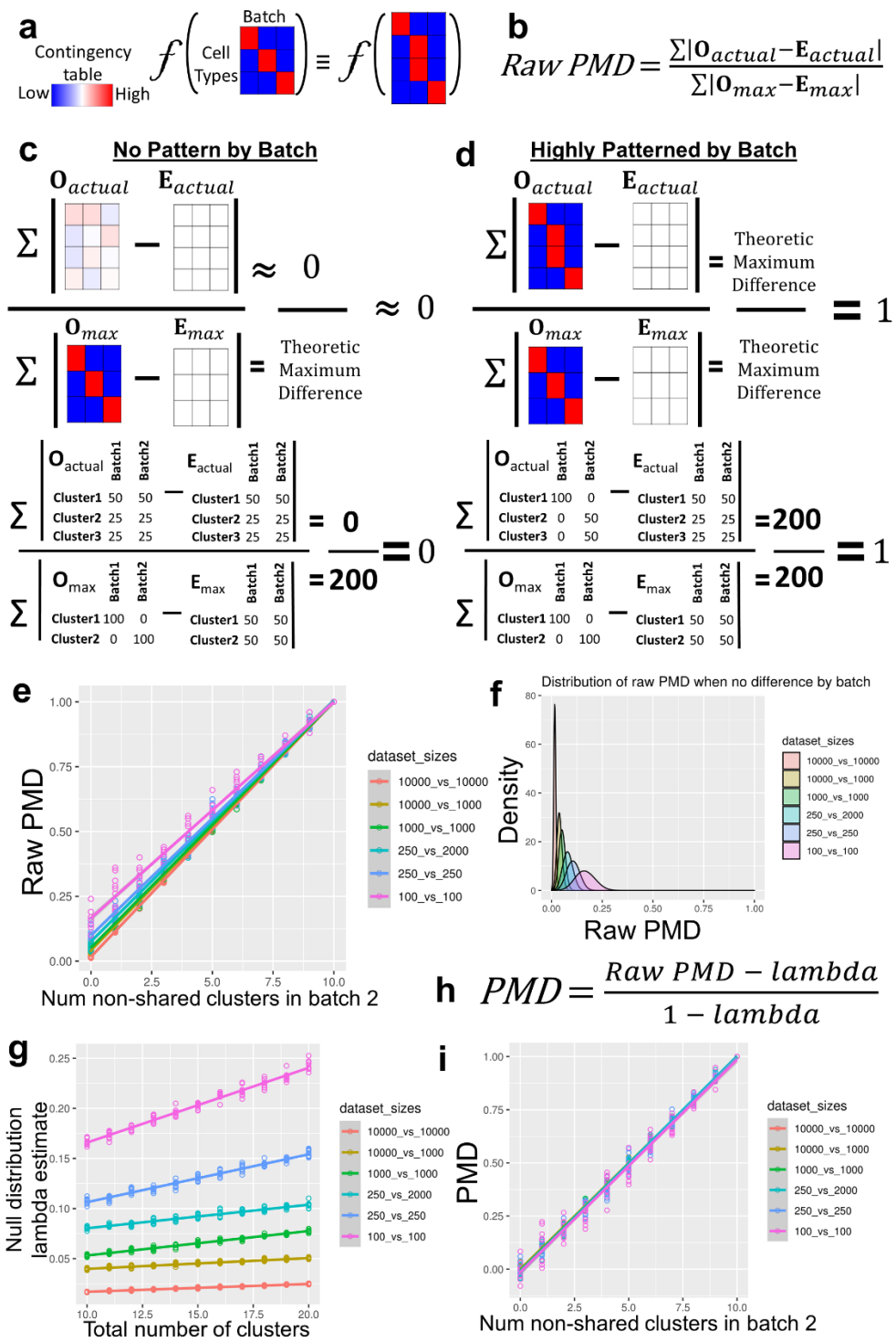
were used in simulations; they were generated from six primary clusters, with a cell-state optionally added (**Extended Data Fig. 7**). **b-i,** Boxplots and schematics illustrating the possible failure-mode are shown for six metrics. **b**, The absolute difference in ground truth (GT) observed PMD indicates how different the dataset similarity is after batch correction compared to the known ground truth. **c**, Similar to **b**, but the log-ratio PMD $\ln(PMD_{observed} / PMD_{ground\text{-}truth})$. Higher indicates that an algorithm tends to leave residual batch effects (batches appear more different than they are in ground truth). Lower indicates that an algorithm tends to make batches more similar than they are in the ground truth. **d**, Difference between the ground truth and observed PMD ($PMD_{ground\text{-}truth} - PMD_{observed}$); interpretation is the same as **c**. **e**, Purity measures the quality of clustering results, penalizing when separate ground truth clusters are erroneously merged together. **f**, Mutual information quantifies overall clustering accuracy using ground truth known cluster identities as reference. **g**, Reverse purity quantifies cluster splitting; when subset for clusters that only appear in both batches, this metric penalizes cluster splitting specifically attributable to the batch effect. **h**, Cluster purity on the subsets of clusters created from the same base cluster with and without the added cell-state penalizes erasing cell-state differences across batch. **i**, Purity on the subset of clusters that share a cell-state, but belong to different base cell-types penalizes merging clusters based on a similar cell-state across batches, but generated from different base cell-types. **j**, Each tested method (row) is sorted by their overall average rank performance among real-world and synthetic figures of merit. Each method's performance for a given metric (columns) is denoted by rank performance such that metrics are comparable to each other. Larger green circles denote better performance rank, while smaller indicate worse performance. A blue box bounds the real-world dataset based results, while a red box outlines the metrics from synthetic datasets. In both cases, only metrics for which the truth is reasonably known were included.



Extended Data Figure 1: The distributions of observed genes and total UMI per cell in the real-world datasets used. a, A violin plot displaying the large range in number of genes (on a log scale) detected in individual cells. **b**, A violin plot showing the large range of total UMI observed at the cell level across different scRNAseq datasets, and even within a single dataset uniformly spanning at least two orders of magnitude, but up to three. **c**, A scatter plot of all cells from each dataset (color coded) showing the relationship between the observed number of genes (X-axis) and the total UMI (Y-axis) observed in each cell. Using our implementation of UMI downsampling, lower range cutoffs are selected by the user to accommodate different tolerances for sensitivity and number of cells included - as indicated by the red lines as an example. Note that many datasets, if not filtered as such, can include sequenced droplets with thousands of UMI coming from only tens of genes.

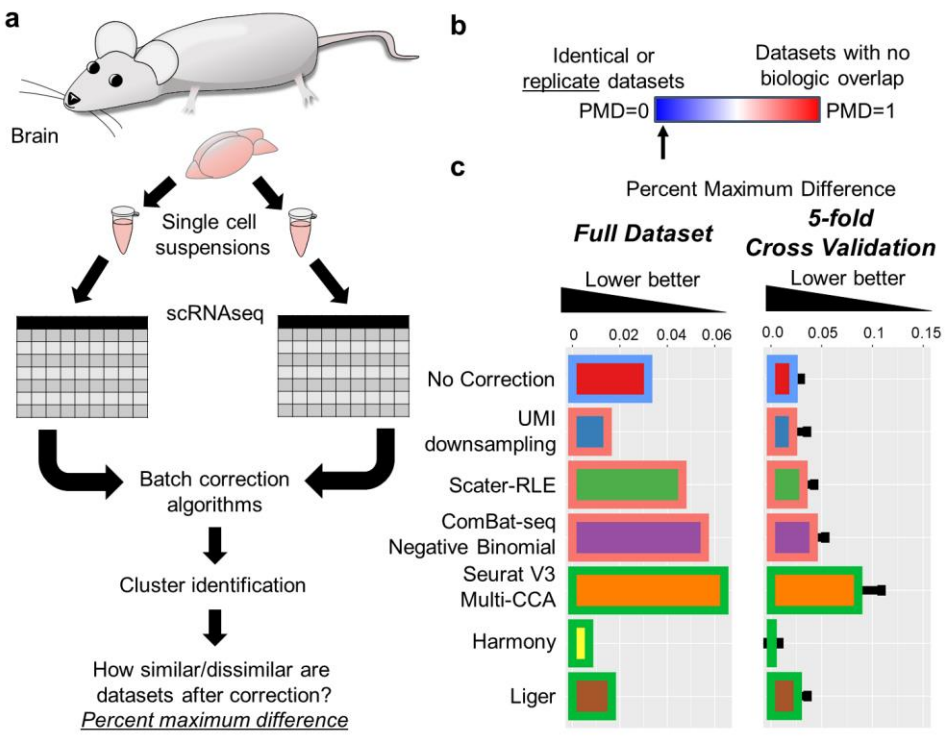


Extended Data Figure 2: Performance when integrating identical datasets with differences only in UMI depth with the addition of an orthogonal dataset. a, The overall experimental and pipeline schematic of this figure of merit. The same brain datasets were used as in **Fig. 1**, but also integrated was a homeostatic intestine dataset. **b**, Because the only assessment made here was the percent of downsampled cells that were identified as belonging to the same cluster as their parent cell, this benchmark was again on the side of extreme similarity in which these cells should be identified as exactly the same. **c**, Bar charts of the percent-same metric show concordant results with those observed without the addition of the orthogonal intestine (**Fig. 1e)**. Results represent either the full dataset, or the mean+/-SD of five-fold cross-validated splits of the data.

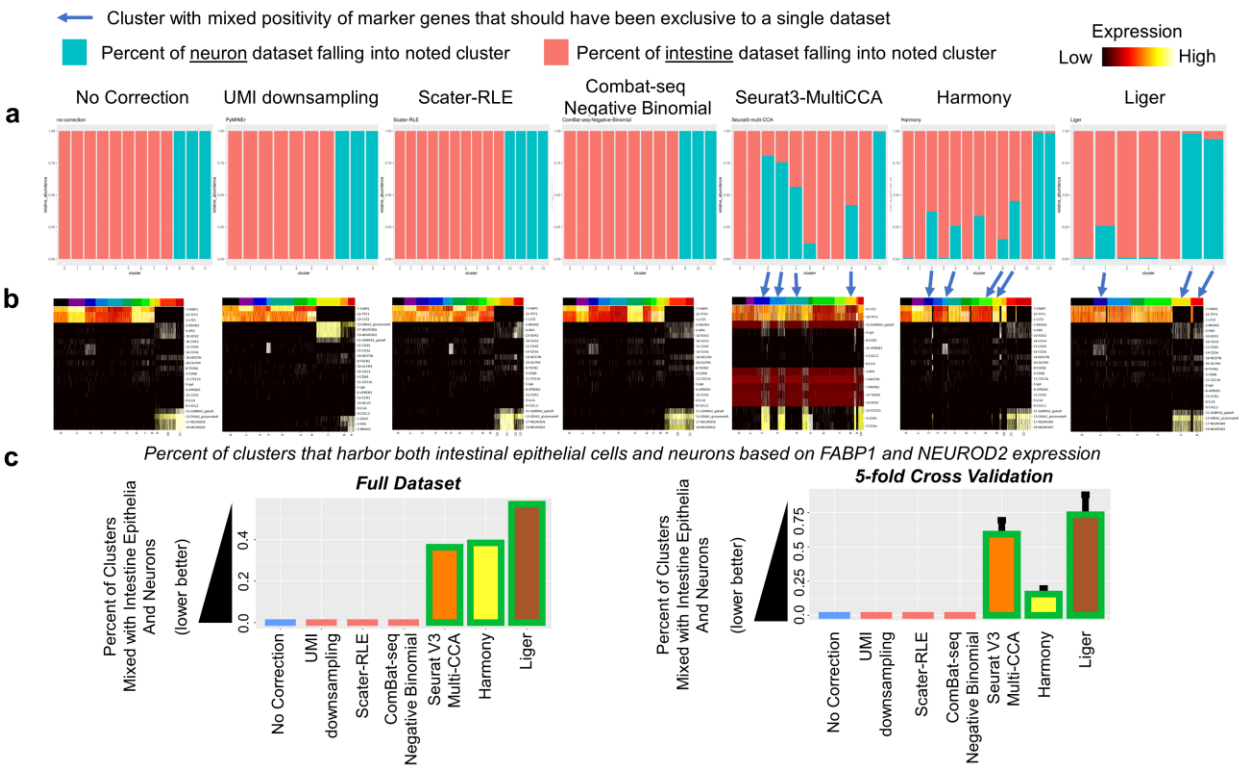


Extended Data Figure 3: Percent Maximum Difference calculation. a, A schematic showing that an ideal metric quantifying differences in batches should be invariant to the number of clusters found within each batch if relative overlap across batches is unchanged. **b**, The

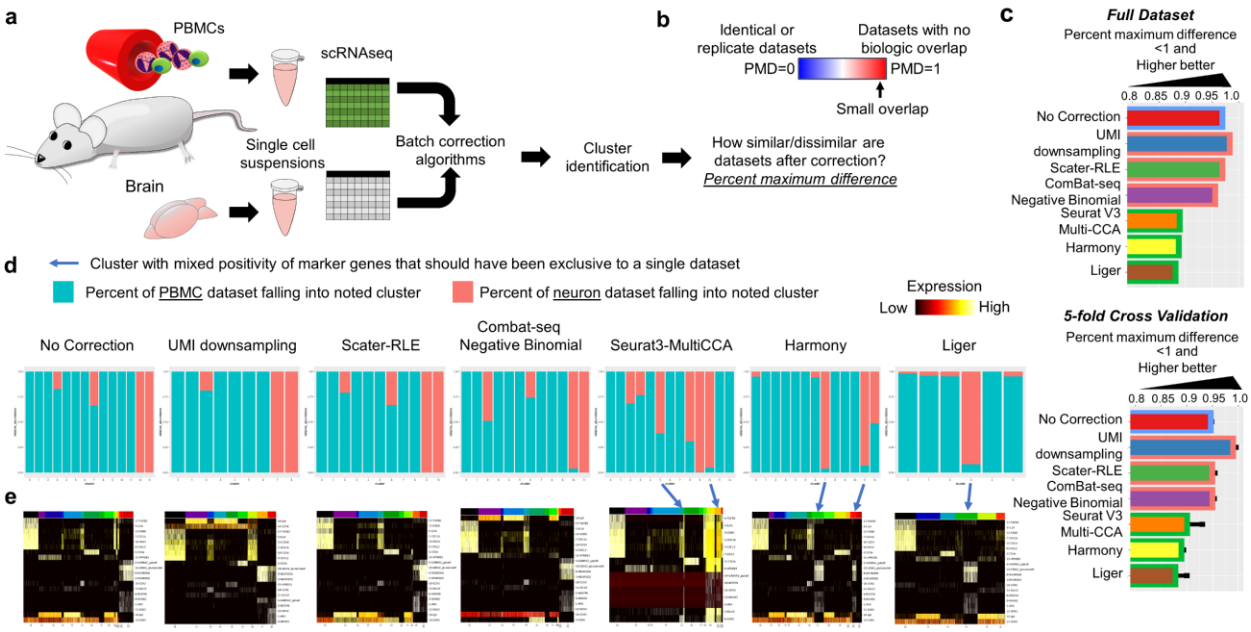
equation for the Raw Percent Maximum Difference (Raw PMD) (before empiric power correction). **c**, A visual example of the Raw PMD calculation where there is very little asymmetry across batches. **d**, In contrast to the scenario in **c**, when the actual clustering results are highly asymmetric by batch, the numerator and denominator are equivalent and this ratio equals one. **e**, Raw PMD (y-axis) plotted against the number of clusters that are specific for a given batch (x-axis: higher is more batch asymmetry); simulations with different batch sizes are noted by color. **f**, Histograms of the intercepts of the lines in **e** revealed Poisson distributions, whose lambda parameter (peak of the distribution) differed based on the number of cells simulated (colors). **g**, Lambda parameter estimates (y-axis) of bootstrap shuffled null versions of the simulations shown in **e**, varied with the number of cells (color) and degree of overlap (x-axis), indicating that Raw PMD can be corrected by measuring the lambda parameter. **h**, The final PMD calculation, adjusted for power with the lambda parameter of the null distribution corrects for the intercept (numerator) and slope (denominator). **i**, After adjustment, PMD (y-axis), is linear with the degree of overlap between batches (as noted by number of batch specific clusters, x-axis), and equivalent regardless of the size of these batches (color).



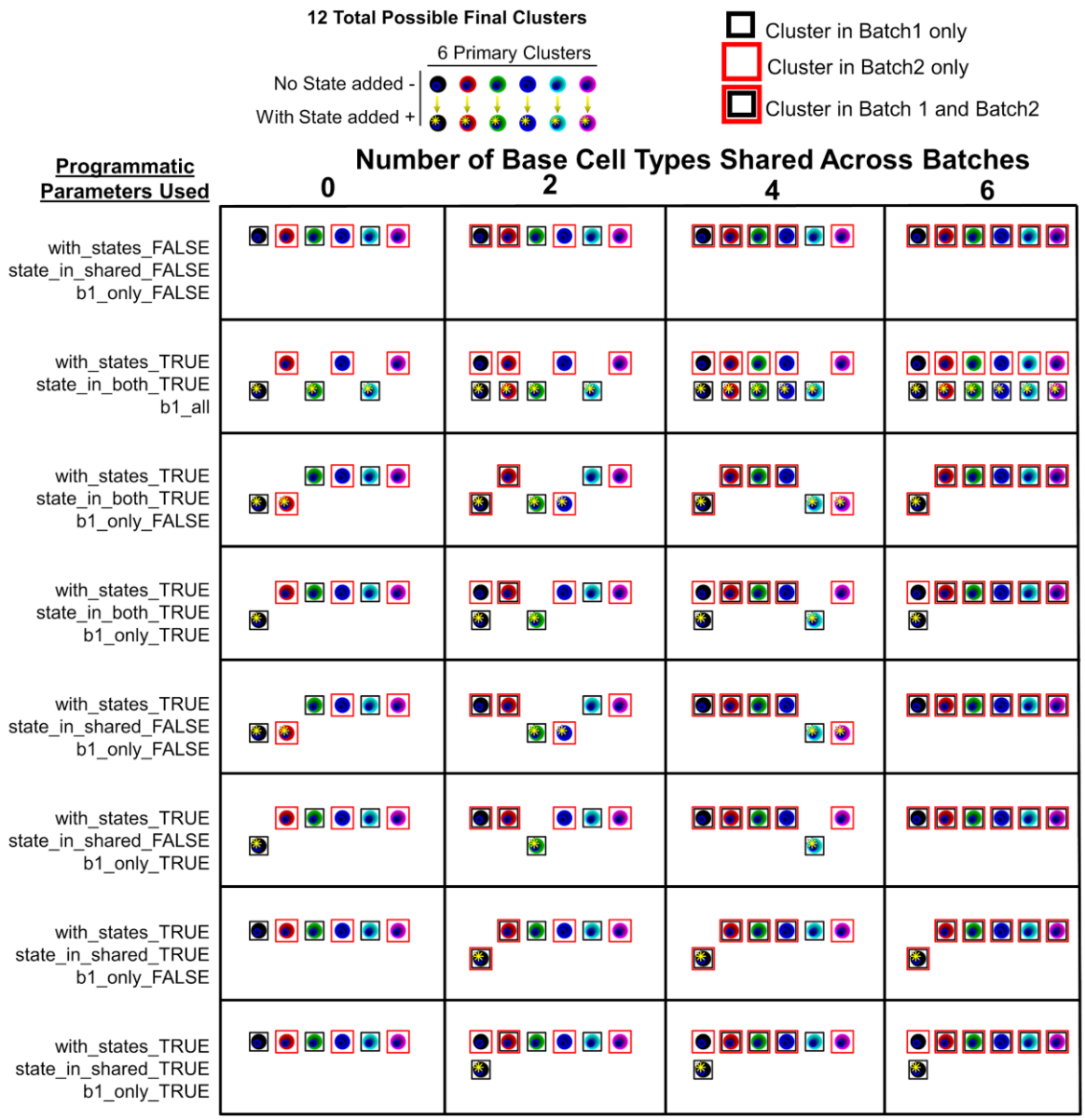
Extended Data Figure 4: Batch correction algorithm performance on integrating two batches from the same biologic sample. a, A schematic representation of the experimental and analytic pipeline for this data integration example. **b**, Given that each scRNAseq dataset was generated from the same single cell suspension (1k neurons and 10k neurons from 10x genomics V3 chemistry) we expect the resulting distribution of cell clusters to be very similar to each other, thus yielding a very low percent maximum difference. **c**, A bar chart showing the percent maximum difference between datasets; results show that Harmony and UMI downsampling were the top two performers respectively in this task given that very little difference is expected between these replicate datasets. Results from the full datasets as well as 5-fold cross validation splits are shown (means+/-SD).



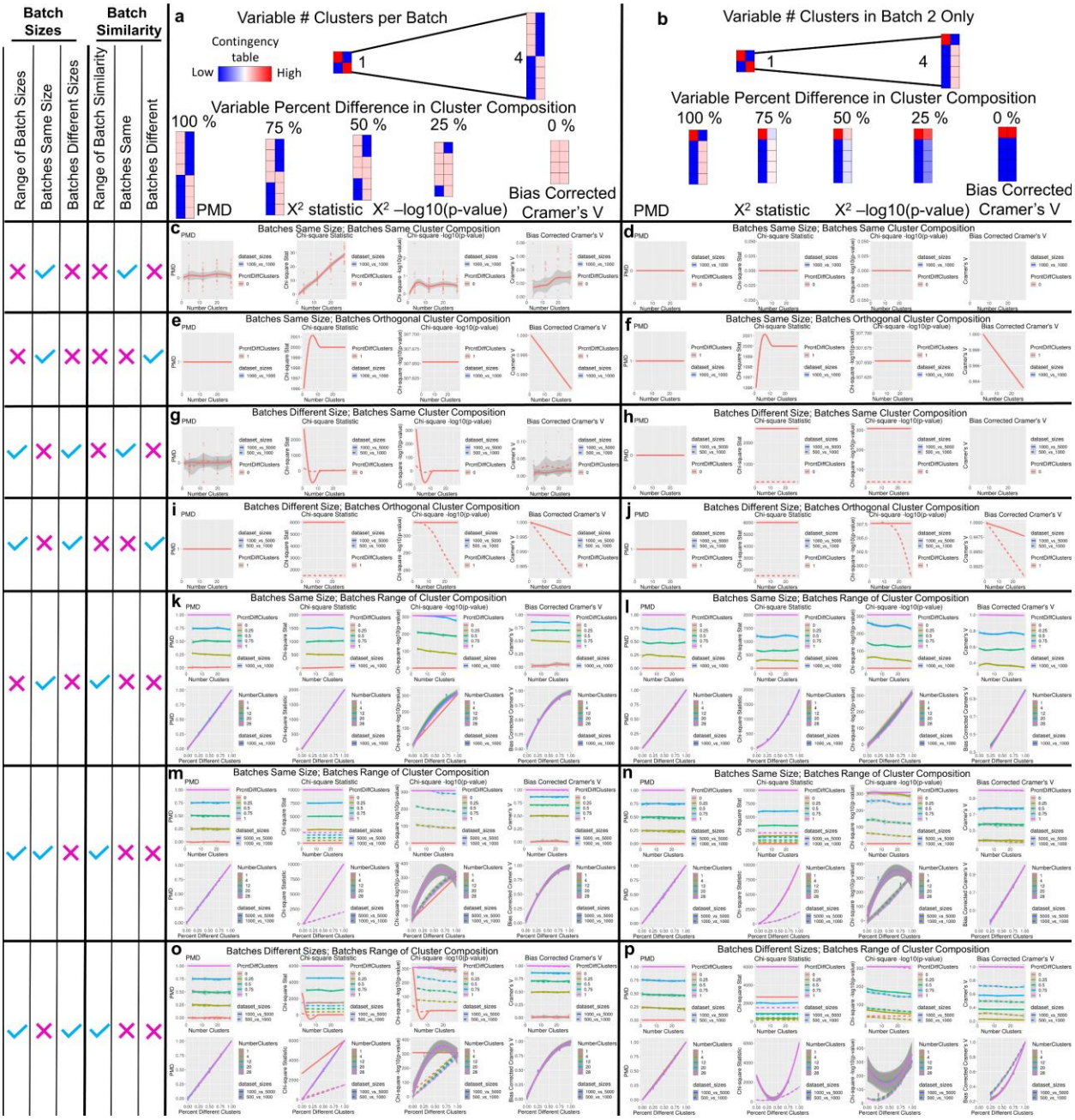
Extended Data Figure 5: Latent-dimension mapping approaches merge biologically real batch confounded sources of variance using brain and intestinal scRNAseq. a, Relative abundance of clusters within each full dataset by different methods. **b**, Heatmaps show manually curated marker genes for several intestinal and neuronal markers that are expected to segregate populations well. Blue arrows indicate clusters with mixed marker positivity across datasets. **c**, Quantifications of the percent of clusters that contain cells that are positive for *FABP1* (an intestinal epithelial marker), alongside cells that are positive for *NEUROD2* (a neuron marker), with positivity defined as a UMI count >5. Results are shown for the full dataset as well as 5-fold cross validated splits showing mean +/- standard deviation.



Extended Data Figure 6: Latent-dimension mapping approaches merge biologically real batch confounded sources of variance using PBMCs and brain scRNAseq. a, scRNAseq from peripheral blood mononuclear cells (PBMCs) and a brain sample were batch corrected and integrated. **b**, PMD was expected to be near, but less than 1. **c**, While all methods found some overlap, depth normalization and no-normalization found that batches were more dissimilar than latent-dimension mapping approaches. On the top panel, PMD is shown for the full dataset, while the bottom panel shows PMD results based on five-fold cross validation splits of each input dataset (mean+/-SD). **d**, Relative abundance of clusters within each dataset by different methods. Displayed are normalized relative proportions of each dataset. **e**, Heatmaps show manually curated marker genes for several intestinal and neuronal markers that are expected to segregate populations well. Blue arrows indicate clusters with mixed marker positivity across datasets.



Extended Data Figure 7: All simulated scenarios. We simulated 8 different parameter combinations (rows), with differing degrees of overlap in base cell-type presence in both or a single batch (columns). In all combinations, this generated 32 simulation combinations that were repeated in triplicate for 96 different scRNAseq batch correction and integration tasks; note that some of these scenarios converged to similar schematics. The noted simulation parameters are also reported in **Extended Data Table 1**.



Extended Data Figure 8: Characteristics of PMD, χ^2 test-of-independence statistic, χ^2 -log10(p-value), and Cramer's V. a,b, Schematic of two different simulation paradigms to vary batch similarity. **a**, Batches were comprised of the same number of clusters, but varied by the number of clusters that were batch specific clusters. **b**, A second simulation paradigm where batch one had one cluster containing its cells (cluster #1), batch two varied in cluster number; the percentage overlap was determined by varying the relative abundance of cluster #1 in batch

two, while equally partitioning the rest of cells to other clusters. **c,e,g,l,k,m,o,** The left column of results demonstrate behavior of each metric (further subdivided by columns) under the first simulation paradigm described in panel **a**. **d,f,h,j,l,n,p**, The right column of results demonstrate metric behavior under the second simulation paradigm described in panel **b**. **c,d**, PMD, χ^2, χ^2's -log10(p-value), and bias corrected Cramer's V (y-axes) for the scenario in which batches have varying number of clusters (x-axis), batches contain the same number of cells, and batches share the same cluster composition. **e**,**f**, Batches are fully orthogonal, sharing no overlap in cluster composition, but batches contain the same number of cells with varying number of clusters (x-axis). **g**,**h**, Batches are identical, but with different batch sizes (indicated by solid & dashed lines). Note that Cramer's V does not compute due to small contingency table size. **i**,**j**, Batches are different sizes (solid and dashed lines), and orthogonal cluster composition. **k-p**, (top row) Panels depicting the relationship between the simulated number of clusters (x-axis), and the corresponding metric (y-axis). Color indicates the ground truth batch overlap. (Bottom row) x-axis is the ground truth percent differences between batches, while colors indicate the number of clusters. In all panels, solid or dotted lines correspond to the indicated dataset sizes. **k**,**l**, Batches are the same size, over a range of cluster compositions. PMD is linear across the spectrum of similarity, regardless of simulation paradigm, while this is untrue for χ^2 or its -log10(p-value). **m**,**n**, Similar to **k,l**, Batches are the same size, over a range of similarities, but now simulated with two different batch sizes (B1 & B2:1000 or 5000 cells each) to determine the effect of different batch sizes (solid or dotted lines). **o**,**p**, Batch sizes are each different, with variable scales of difference shown by solid or dotted lines when batches are of different size from each other.

	Comparison Type	Comparison Description	Experimental Setup	Figure of Merit
a	Experimentally identical	Cells of a single tissue compared to subsamples of the transcriptomes of the same cells	Subsample from source “parent” cells to create “child” cells with all cells of the same tissue type	Cluster across all parent and child cells and compute the percentage of times child cells cluster with parent cells (**Fig. 1**)
b	Experimentally identical in context of highly dissimilar	As above (**Box 1a**), but in the context of unrelated cell types	Subsample from source “parent” cells to create child cells where these cells are of the same type, but mixed with unrelated cell types	Same as above **Box 1a** (**ED Fig. 2**)
c	Experimentally highly similar	Cells from the same single cell suspension	scRNA seq data generated on multiple isolates of the same sample	Percent maximum difference where methods that minimize the PMD are preferred (**ED Fig. 4**)
d	Experimentally highly dissimilar	Cells of one type compared to cells of an unrelated type	Independent runs of scRNAseq data generated on disparate tissues expected to have little or no overlap in cellular composition	Percent maximum difference where methods that maximize the PMD are preferred (**Fig. 2, ED Fig. 5,6**)
e	Experimentally batch confounded cell-state and cell-type abundance.	Cells of a given batch are of similar cell-type, but in different cell-states, and cell-type abundance	scRNAseq data generated on cells derived from normal intestine and cell derived from irradiated intestine, or differences in endometrium throughout menstruation.	Percent maximum difference (**Fig. 3,4**)
f	Simulated mixtures of similar and dissimilar cells	Cells of the same and different types compared for similar and dissimilar groupings with and without batch confounded cell-state	Different cell-types/states simulated across different batches to assess clustering results after batch correction compared to ground truth clusters	Various including batch similarity compared to ground truth, overall clustering accuracy, extent of misclustering by batch (**Fig. 5**)

Box 1: A brief overview of the figures of merit based on real biological datasets, and synthetic datasets, indicating whether they are testing how approaches perform on similar or dissimilar datasets and whether or not cell state information is included in the figure of merit or synthetic benchmark.

Extended Data Table 1: All results for simulation scenarios, and metrics used for quantifying batch correction algorithm performance are provided.