RUV-III-NB: Normalization of single cell RNA-seq Data

Agus Salim [1,2,3,4,5*], Ramyar Molania [2], Jianan Wang [2,6],
Alysha De Livera [1,7], Rachel Thijssen [8] Terence P. Speed [2,3*]

[1] Melbourne School of Population and Global Health, University of Melbourne VIC 3053
[2] Bioinformatics Division, Walter and Eliza Hall Institute of Medical Research, Parkville VIC 3052
[3] School of Mathematics and Statistics, University of Melbourne VIC 3010
[4] Baker Heart and Diabetes Institute, Melbourne VIC 3004
[5] Department of Mathematics and Statistics, La Trobe University VIC 3086
[6] Department of Medical Biology, University of Melbourne VIC 3010
[7] School of Science, RMIT University, Melbourne VIC 3000
[8] Blood Cells and Blood Cancer Division, Walter and Eliza Hall Institute of Medical Research, Parkville VIC 3052, Australia

[*] To whom correspondence should be addressed.
November 6, 2021

Abstract

Motivation: Despite numerous methodological advances, the normalization of single cell RNA-seq (scRNA-seq) data remains a challenging task. The performance of different methods can vary greatly across datasets. Part of the reason for this is the different kinds of unwanted variation, including library size, batch and cell cycle effects, and the association of these with the biology embodied in the cells. A normalization method that does not explicitly take into account cell biology risks removing some of the signal of interest. Here we propose RUV-III-NB, a statistical method that can be used to adjust counts for library size and batch effects. The method uses the concept of pseudo-replicates to ensure that relevant features of the unwanted variation are only inferred from cells with the same biology and return adjusted sequencing count as output.
Results: Using five publicly available datasets that encompass different technological platforms, kinds of biology and levels of association between biology and unwanted variation, we show that RUV-III-NB manages to remove library size and batch effects, strengthen biological signals, improve differential expression analyses, and lead to results exhibiting greater concordance with independent datasets of the same kind. The performance of RUV-III-NB is consistent across the five datasets and is not sensitive to the number of factors assumed to contribute to the unwanted variation. It also shows promise for removing other kinds of unwanted variation such as platform effects. **Availability:** The method is implemented as a publicly available R package available from `https://github.com/limfuxing/ruvIIInb`.
Contact: salim.a@unimelb.edu.au, terry@wehi.edu.au

Supplementary information: Online Supplementary Methods and Figures online.

Introduction

Single-cell RNA-seq (scRNA-seq) technologies have gained popularity over the last few years as more and more studies interrogate transcriptomes at the single cell level. Just as with other omics data, scRNA-seq data inevitably contains unwanted variation which can compromise downstream analyses if left unaddressed. As in the case with bulk RNA-seq data, library size is the major source of unwanted variation in scRNA-seq data and consequently, removing library size effects is the first priority in preprocessing scRNA-seq data. The successful removal of library size effects is crucial for the validity of downstream analyses such as clustering, cell-type annotation, differential expression and trajectory analyses. Several studies (Lun *et al.*, 2016; Vallejos *et al.*, 2017; Bacher *et al.*, 2017; Hafemeister and Satija, 2019) have found that the bulk RNA-seq procedures for removing library size effects do not work well for scRNA-seq data. This is because the relationship between gene expression and library size in scRNA-seq data is typically complex and gene-specific, a feature of the data that has necessitated the development of methods using gene-specific scaling factors (Bacher *et al.*, 2017; Hafemeister and Satija, 2019; Brown *et al.*, 2021), as opposed to methods that use global scaling factors e.g. (Robinson and Oshlack, 2010; Lun *et al.*, 2016). In addition to library size effects, scRNA-seq data can exhibit batch effects (Ziegenhain *et al.*, 2017) due to variation between cell counts *within* a study (e.g. due to plate-to-plate variation) and variation between cell counts *across* studies (e.g. due to platform and sample preparation variation). In this paper, we concentrate on dealing with the first, although we show that our method has the potential to perform *data integration* by adjusting for library size and batch effects across studies.

Like Vallejos *et al.* (2017), in this paper we will use the term 'normalization' to refer to a procedure that attempts to remove all kinds of unwanted variation and not only that due to library size. One of the key challenges when performing normalization is to remove the right kind and amount of variation. Removing the wrong or too much variation risks removing biology, especially if biological variation is associated with unwanted variation. Most methods that adjust scRNA-seq data for batch effects (Haghverdi *et al.*, 2018; Korsunsky *et al.*, 2019; Lin *et al.*, 2019b; Stuart *et al.*, 2019) proceed in two steps: library size effects are removed first, and then batch effects are removed from data that has been adjusted for library size. This approach is reasonable if there is little or no association between library size, batch and biology, but when there are such associations, its effectiveness may be reduced. For example, when different cell-types have quite different library size distributions, the first step may adjust the data too aggressively and remove library size differences arising as differences between cell-types. ZINB-WaVE (Risso *et al.*, 2018) can be used to perform simultaneous adjustment for library size and batch effects. However, it requires that the batches are known a priori, and its adjustment is carried out without considering the possibility that library size, biology and batch may be associated. In this paper, we propose RUV-III-NB that simultaneously adjusts scRNA-seq gene counts for library size and within study batch differences. As with RUV-III (Molania *et al.*, 2019) which inspired this work we do not assume that batch details are known, but seek to use *replicates* and *negative control* genes to capture and adjust for the unwanted variation. Negative controls are genes whose variation is (largely) unwanted and not of biological interest, while we necessarily modify our notion of replicates, for the gene expression levels in single cells cannot be measured in replicate. To ensure that the right kind and amount of variation is removed from gene counts we estimate the effect of unwanted variation on these counts using suitably defined *pseudo-replicates* of cells or *pseudo-cells* that have the same biology, and we propose strategies to define pseudo-replicates. Using five real datasets, we compare RUV-III-NB to several popular methods for normalizing scRNA-seq data and demonstrate its ability to retain biological signals when biology and unwanted variation are associated.

The organization of this paper is as follows. In Section 2, we introduce the statistical model underlying RUV-III-NB and the strategies we propose for defining pseudo-replicates. In Section 3, we benchmark RUV-III-NB against other approaches using five publicly available datasets. We then close with a Discussion in Section 4.

RUV-III-NB: Model and framework

RUV-III-NB takes raw sequencing counts as input and models the counts y_{gc} for genes g and cells c, as independent Negative Binomial (NB), $y_{gc} \sim NB(\mu_{gc}, \psi_g)$ or Zero-Inflated Negative Binomial (ZINB) random variables, $g = 1, \dots G, c = 1, \dots N$. Here we will only discuss the NB model for UMI data and leave the ZINB model for read count data to the Supplementary Methods section. Without loss of generality, we further assume there are m groups among the N cells with the same underlying biology within and different underlying biology across groups. We will refer to these groups as pseudo-replicate sets, that is, sets of cells whose members will be regarded as replicates for the purposes of normalization. Let $\boldsymbol{y}_g = (y_{g1}, y_{g2}, \dots, y_{gN})^T$ be the vector of counts for gene g and $\boldsymbol{\mu}_g$ be its vector of mean (i.e. expected value) parameters under the NB model. We use a generalized linear model with log link function to relate these mean parameters to the unobserved unwanted factor levels captured by the matrix $\mathbf{W}$ while the biology of interest will be embodied in the matrix $\mathbf{M}$, these being related by

$$\log \boldsymbol{\mu}_g = \zeta_g \mathbf{1} + \mathbf{M}\boldsymbol{\beta}_g + \boldsymbol{W}\boldsymbol{\alpha}_g, \tag{1}$$

where $\boldsymbol{M}(N \times m)$ is the pseudo-replicate design matrix with $\boldsymbol{M}(i,j) = 1$ if the ith cell is part of the jth pseudo-replicate set and 0 otherwise, $\boldsymbol{\beta}_g(m \times 1) \sim N(0, \lambda_\beta^{-1}\boldsymbol{I}_m)$ is the vector of biological parameters, with unique values for each of the m replicate sets, $\mathbf{W}(N \times k)$ is the unobserved matrix of k-dimensional unwanted factor levels and $\boldsymbol{\alpha}_g(k \times 1) \sim N(\boldsymbol{\alpha}_\mu, \lambda_\alpha^{-1}\boldsymbol{I}_k)$ is the vector of regression coefficient associated with the unwanted factors, and finally ζ_g is the location parameter for gene g after adjusting for unwanted factors, $g = 1, \ldots G$. In our application, we found that setting $\lambda_\alpha = 0.01$ and $\lambda_b = 16$ yield good results.

For a given number k of unwanted factors we use a double-loop iteratively re-weighted least squares (IRLS) algorithm, where in the inner loop, given current estimates of the dispersion parameters, we estimate the parameters of the loglinear model above, including the unobserved unwanted factor levels $\mathbf{W}$ (see Supplementary Methods for details), and once convergence is achieved there, we update the dispersion parameters in the outer loop. Two important constructs enable the algorithm to estimate the unobserved unwanted factor levels and their gene-specific effects on the sequencing count. These two constructs are the pseudo-replicate design matrix $\mathbf{M}$ and the set of negative control genes.

The pseudo-replicate design matrix $\mathbf{M}$ plays an important role for estimating the effect of the unwanted factors on the data (Molania *et al.*, 2019; Jacob *et al.*, 2015). This effect is represented by $\boldsymbol{\alpha}_g$ and in RUV-III-NB it is estimated by projecting the current IRLS working vector onto the orthogonal complement of the subspace spanned by the columns of $\mathbf{M}$ and, given the estimate of $\boldsymbol{\alpha}_g$, we use the set of *negative control* genes to estimate the unobserved unwanted factor levels $\mathbf{W}$. As stated above, negative controls are genes whose variation is (largely) unwanted and not of biological interest, (Gagnon-Bartsch and Speed, 2012), i.e, $\boldsymbol{\beta}_g \approx 0$ for all negative control genes g. The model for these genes thus reduces to

$$\log \boldsymbol{\mu}_g \approx \zeta_g \mathbf{1} + \boldsymbol{W}\boldsymbol{\alpha}_g,$$

We recommend the use of single-cell housekeeping genes (Lin *et al.*, 2019a) as the negative controls but users can (and may need to) devise their own negative control set. The important property of such genes is that they are affected by the same sources of unwanted variation as the other genes, and that their variation is not related to the biology of interest in the study.

Strategies for defining pseudo-replicate sets

To identify the effects of the unwanted variation on the gene counts, the RUV-III-NB algorithm requires users to specify one or more sets of cells with relatively homogeneous biology, and these are called *pseudo-replicate* sets. In cases where the biological factor of interest for each cell is known, e.g when different treatments are compared across the same cell type, or when two or more cell lines are being compared, then cells with the same level of the biological factor of interest can be declared to be a pseudo-replicate set . There will be situations where the biology of interest is not known a priori at the single cell level. For example, it is often the case that cell type information is unavailable in advance, especially for droplet-based technologies. For such situations we will outline some strategies that can be used to define pseudo-replicate sets.

Single batch

When the data comes from a single batch, users can cluster the cells into distinct biologically homogeneous sets of cells. The clustering could be done using the log (normalized count + 1) where the scaling factor for normalization is calculated using `computeSumFactors` function in `scran` package (Lun *et al.*, 2016). For clustering we recommend the use of a graph-based method such as the Louvain algorithm (Blondel *et al.*, 2008). Cells allocated to the same cluster can then be considered to form a pseudo-replicate set.

Multiple batches

When the data comes multiple batches, we need to match clusters containing cells with similar biology located in different batches. We recommend that users use the `scReplicate` function in the Bioconductor package `scMerge` (Lin *et al.*, 2019b) for this purpose. This function takes log(normalized count + 1) as input and performs K-means clustering for each batch separately followed by identification of clusters in different batches that are mutual nearest clusters (Lin *et al.*, 2019b). Once these mutual nearest clusters (MNC) are identified, cells from the same MNC can be considered to form a pseudo-replicate set.

Strengthening pseudo-replicate sets using pseudo-cells

Even when pseudo-replicate sets can be defined by clustering, the clustering may at times be imprecise, with considerable biological heterogeneity across cells in the same cluster. Thus declaring all such cells to be a pseudo-replicate set may risk removing some of the biological signal of interest. To address this issue, we introduce the idea of basing pseudo-replicate sets on *pseudo-cells*.

Pseudo-cells: single batch

Within a single batch and biology, we suppose that the major source of unwanted variation is library size, and that other intra-batch variation (e.g., well-to-well variation within a plate) is minimal. The idea is to form pseudo-replicates of pseudo-cells that have been constructed to have as much variation as possible in their library size while keeping their biology as homogeneous as possible, more homogeneous than we might see in actual single cells in a pseudo-replicate set. Suppose we have identified m pseudo-replicate sets using either known single cell biology or the strategy that we have just described above. For each of the pseudo-replicate sets, we form pseudo-cells that represent the pseudo-replicate set using the following pool-and-divide strategy:

1. Assign each cell to one of the $J = 10$ pools based on its library size, where pool j contains n_j cells, $j = 1 \dots J$.

2. Pooling: Let $\mathbf{Y}_j$ be the matrix of counts for cells belonging to pool $j = 1, 2, \dots J$, where rows corresponds to genes and columns corresponds to cells. We aggregate the counts for these cells by forming row totals of $\mathbf{Y}_j$ and denote the vector containing these row totals by $\mathbf{s}_j$ with components $s_{gj} = \sum_{c \in poolj} y_{gc}$.

3. Dividing: For each gene g, we randomly sample a count z_{gj} from the pool-aggregated counts, where $z_{gj} \mid s_{gj} \sim \text{Binomial}(s_{gj}, p = 1/n_j)$ where s_{gj} is the aggregated count for gene g in pool j consisting of n_j cells. This step is formally equivalent to randomly dividing the aggregated counts for the pool into those for n_j pseudo-cells and choosing one of the pseudo-cells at random. The hope is that the pseudo-cell so defined will exhibit average and so stabler biology in its gene counts, while concentrating the unwanted variation in the pool, here library size.

4. We thus obtain counts $\mathbf{z}_j = (z_{1j}, z_{2j}, \dots, z_{Gj})^T$ for the pseudo-cell that represents pool j.

5. We repeat steps 1-4 for all J pools and declare the J pseudo-cells so defined to be a pseudo-replicate set.

6. Finally, we carry out steps 1-5 above for the other pseudo-replicate sets, at the end of which we will have m pseudo-replicate sets each containing J pseudo-cells.

It can be shown that the counts assigned to these pseudo-cells will still have the quadratic mean-variance relationship typical of negative binomial random variables (see Supplementary Methods). The difference between these pseudo-cells and the real cells lies in the overdispersion parameter. For the same gene, the overdispersion parameter for pseudo-cells will be smaller, reflecting the reduced variability resulting from the pool-and-divide strategy. To incorporate this feature of pseudo-cells into the RUV-III-NB fitting process, we simply treat them as additional cells whose dispersion parameters are estimated separately from those of the real cells.

Pseudo-cells: multiple batches

When there are multiple batches, the procedure for forming pseudo-cells just described needs to follow the stratification of our cells into sets of MNC. Then we construct pseudo-cells for each of the clusters that makes up an MNC. For example, suppose we have $b = 3$ batches A, B and C and we identified two clusters for each batch with the following MNC: (A_1, B_2, C_2) and (A_2, B_1, C_1) where A_1 refers to the first cluster in batch A, etc. The procedure for forming the pseudo-cells would then be as follows:

1. Start with the first MNC (A_1, B_2, C_2)
2. Assign each cell in A_1 into one of the J pools based on its library size, where pool j contains n_j cells.
3. Pooling: Let $\mathbf{Y}_j$ be the matrix of counts for cells belonging to pool j where rows correspond to genes and columns corresponds to cells. Aggregate the gene counts in these cells by forming the row totals of $\mathbf{Y}_j$ and denote this new vector by s_{gj}.
4. Dividing: For each gene g, we randomly sample a count z_{jg} from the pool-aggregated counts, where $z_{jg} \sim \text{Binomial}(s_{jg}, p = 1/n_j)$ where s_{jg} is the pool-aggregated count for gene g. As above, this step is equivalent to randomly dividing the aggregated counts for the pool into those for n_j pseudo-cells and choosing one of the pseudo-cells randomly.
5. We thus obtain $\mathbf{z}_j = (z_{1j}, z_{2j}, \ldots, z_{Gj})$ as the count data for pseudo-cell that represent pool j.
6. Repeat steps 2-5 for cells in B_2 and C_2.
7. Declare all the pseudo-cells formed in step 2-6 above to be a pseudo-replicate set.
8. Go to step 1 and repeat steps 2-6 for the second MNC (A_2, B_1, C_1)

When this procedure is completed, we will have as many pseudo-replicate sets as we have MNC sets and each pseudo-replicate set is made up of $b \times J$ pseudo-cells.

Adjusted counts

Once we obtain the estimates of unwanted factors $\hat{\boldsymbol{W}}$ and their effects $\hat{\boldsymbol{\alpha}}_g$, we remove their effects from the raw data. RUV-III-NB provides two forms of adjusted data. These adjusted data can be used as input to downstream analyses such as clustering, trajectory and differential expression analyses.

- Pearson residuals:

  $$\frac{y_{gc} - \hat{\mu}_{gc}}{\sqrt{\hat{\mu}_{gc} + \hat{\mu}_{gc}^2 \hat{\psi}_g^2}}$$

  where $\hat{\mu}_{gc} = \exp(\hat{\zeta}_g + \hat{\boldsymbol{w}}_{\boldsymbol{c}}^{\boldsymbol{T}} \hat{\boldsymbol{\alpha}}_g)$.

  When $k = 1$ and $\hat{\mathbf{W}}$ is approximately equal to log library size (up to a scaling factor), these Pearson residuals agree with those of (Hafemeister and Satija, 2019). When $k > 1$ and some columns of $\mathbf{W}$ reflect batch effects, these Pearson residuals will also adjust for unwanted variation other than library size, such as batch effects.

- Log of percentile-invariant adjusted count (PAC):

$$\log(F^{-1}(r_{gc}; \mu_{gc} = \exp(\hat{\zeta}_g + \boldsymbol{m_c^T} \boldsymbol{\hat{\beta}_g} + \boldsymbol{\bar{w}^T} \boldsymbol{\hat{\alpha}_g}), \hat{\psi}_g) + 1)$$

where $r_{gc} \sim U(a_{gc}, b_{gc})$ and

$$\begin{aligned} a_{gc} &= F(y_{gc}; \mu_{gc} = \exp(\hat{\zeta}_g + \boldsymbol{m_c^T} \boldsymbol{\hat{\beta}_g} + \boldsymbol{\hat{w}_c}^{\boldsymbol{T}} \boldsymbol{\hat{\alpha}_g}, \hat{\psi}_g)) \\ b_{gc} &= F(y_{gc} + 1; \mu_{gc} = \exp(\hat{\zeta}_g + \boldsymbol{m_c^T} \boldsymbol{\hat{\beta}_g} + \boldsymbol{\hat{w}_c}^{\boldsymbol{T}} \boldsymbol{\hat{\alpha}_g}, \hat{\psi}_g)) \end{aligned}$$

where $F(.)$ is the negative binomial c.d.f and $F^{-1}(.)$ its inverse, $\boldsymbol{m}_c$ is the cth row of the matrix $\boldsymbol{M}$, $\boldsymbol{\hat{w}_c}$ the cth row of the matrix $\boldsymbol{\hat{W}}$ and $\boldsymbol{\bar{w}}$ is vector of entries equal to the average level $N^{-1} \sum_{c=1}^{N} \boldsymbol{\hat{w}_c}$ of unwanted variation. Here $U(a, b)$ denoted a random variable uniformly distributed over the interval (a, b).

The intuition behind this adjustment is as follows. We first obtain the percentiles of the observed counts under the fitted NB model, where the mean value parameter includes terms for unwanted variation. Since negative binomials are discrete distributions, percentiles can only be determined up to an interval. To come up with an estimate of a percentile for practical use, we simply select a uniformly distributed random value from this interval in a manner suggested in (Dunn and Smyth, 1996). We then find the corresponding count for that estimated percentile under a different NB model, namely one where the mean parameter is free from unwanted variation, i.e. where $\boldsymbol{\hat{w}_c^T} \boldsymbol{\hat{\alpha}_g}$ is replaced by $\boldsymbol{\bar{w}^T} \boldsymbol{\hat{\alpha}_g}$. We then add 1 and *log*. Our definition of percentile-invariant adjusted count explicitly derives the counts as percentiles of a full NB distribution and in this regard it is similar to that in Zhang *et al.* (2020) who proposed this approach to obtain batch-corrected bulk RNA-seq data. Their adjustment was only applied to non-zero counts, and left the zero counts intact. That was not expected to pose significant problems for bulk RNA-seq data where zero counts are relatively scarce, but because zero counts are very prominent in scRNA-seq data, we broaden their approach and also adjust zero counts. On the other hand, scTransform's corrected count (Hafemeister and Satija, 2019) is calculated by taking away from the observed count the difference between the predicted counts at the observed and at the average log library size, followed by rounding to avoid non-integer values.

Benchmarking Datasets

To benchmark our methods against others, we use the following five datasets that encompass different technological platforms, illustrate different strategies for identifying pseudo-replicates and pose different challenges for normalization due to association between different unwanted factors and biology (Table 1).

Prior to normalization all datasets were subjected to quality control checks using Bioconductors's `scater` package (McCarthy *et al.*, 2017) to remove low quality cells. Low abundance genes were also removed and additional parameters for each method were set to their default.

- Non-Small Cell Lung Cancer cells (NSCLC): The dataset was generated using 10x and is freely available from the 10x Genomics website (www.10xgenomics.com). The sequencing was done in one batch, so there are no batch effects, but the cells are a mixture of cells with larger size such as epithelial cells and smaller cells such as T cells. The challenge here is to normalize when library size is associated with the biology, namely, cell-type. After QC, there were 10,0019 genes and 6,622 cells.

- Cell line: 10x technology was used to sequence cells in three batches. One batch contained only the Jurkat cell line, another contained only the 293T cell line, while the third batch contains 50-50 mixture of both cell lines. Data were downloaded from 10x Genomics website. After QC, there were 7,943 genes and 9,027 cells.

- Chronic lymphocytic leukemia (CLL): This in-house dataset was generated using the CelSeq2 technology as part of a study investigating the transcriptomic signature of Venetoclax resistance. The cells were pre-sorted so that the vast majority are B-cells and were treated with dimethyl sulfoxide (DMSO) as well as single treatment (TRT) and combination treatments (TRT+) for one week, before being sequenced on six different plates. In addition to this, a small number of cells from the Granta cell line were included on each plate. After QC, there were 11,470 genes and 1,644 cells. The dataset is included as `CLLdata` object in the `ruvIIInb` R package.

- Gaublomme: Th17 cells derived under a non-pathogenic condition (TGF-β1+IL-6, unsorted: 130 cells from 2 batches and TGF-β1+IL-6; sorted for IL-17A/GFP+: 151 cells from 3 batches) and a pathogenic condition (Il-1β1+IL-6+IL-23, sorted for IL-17A/GFP+: 139 cells from 2 batches) were sequenced using the SMARTseq technology (Gaublomme *et al.*, 2015). After QC, there were 7,590 genes and 337 cells.

- Pancreas: Human pancreatic islet cells from two different studies. (Baron *et al.*, 2016) used the inDrop technology to sequence the cells, while (Muraro *et al.*, 2016) used the CELSeq2 technology. Datasets were downloaded from https://hemberg-lab.github.io/scRNA.seq.datasets/human/pancreas/. After QC, there were 11,542 genes and 10.687 cells.

Benchmarking Methods

For the NSCLC dataset where there is no batch effect and the only task is removing library size effects, we compared RUV-III-NB with the following methods: scran (Lun *et al.*, 2016), scTransform (Hafemeister and Satija, 2019), ZINBwave (Risso *et al.*, 2018) and Dino (Brown *et al.*, 2021). For the other datasets where batch effects are present, we compare RUV-III-NB to the following batch correction methods: mnnCorrect and fastMNN (Haghverdi *et al.*, 2018), Seurat3 (Stuart *et al.*, 2019) coupled with scTransform normalization, ZINBwave (Risso *et al.*, 2018) and scMerge (Lin *et al.*, 2019b). These methods have been selected because all of them return the adjusted gene level counts required to calculate the benchmarking metrics (see below). This is in contrast with other methods such as Harmony (Korsunsky *et al.*, 2019), where the adjusted data is only available as principal components (PC). Some of the methods produce multiple versions of normalized data and in Supp. Table 1, we provide details on which normalized version we used for calculating the various metrics in our benchmarking exercise.

We use the following criteria for assessing the performance of the different methods:

- **Genewise correlations between the adjusted counts and log library size:** We expect a good normalization to remove any association between gene expression levels and log library size, especially when cells with the same biology are considered.

- **R^2 between log library size and the leading PC:** For each cell-type, the coefficient of determination (R^2) when regressing log library size on the leading PCs should be as low as possible. This is because we believe that a good normalization should reduce the association between the adjusted data and log library size, so within a group of cells with similar biology, the leading PCs should contain little information about library size.

- **Silhouette statistics for clustering by batch:** When batch effects are reduced, we should expect a lower degree of clustering by batch, within a set of cells with homogeneous biology.

- **Silhouette statistics for clustering by biology:** When batch effects are removed, we expect biological signals to be strengthened and lead to better clustering by biology.

  For all methods, with the exception of Seurat3-Integrated, to calculate silhouette, we used the first ten PC calculated using genes whose normalized expression variance lies in the top 50%. For Seurat3-Integrated, we use all anchor features for calculating PC. The number of anchor features is typically 2000, much less than the half of the total number of genes. When biological factors of interest are available from the dataset, these are used to calculate silhouette scores. Otherwise, we use the Bioconductor package `SingleR` (Aran *et al.*, 2019) to estimate the cell types. PC were derived using the R package `irlba` (Baglama *et al.*, 2019).

- **Differential expression (DE):** When comparing cells of the same cell-type across batches or smaller vs larger library size, a good normalization should *decrease* the proportion of differentially expressed genes (DEG). When comparing cells across different biologies, a good normalization should increase the concordance between the results found with the current and those of an independent study, as measured by the number of DEG.

- **Canonical correlation between relative log expression (RLE) plot medians and interquartile ranges (IQR) and log library size and batch variables:** With good normalization, we expect that within a set of cells with homogeneous biology, the RLE plot summary statistics have little association with unwanted factors.Because the RLE calculation requires subtracting log of gene-specific median expression (Gandolfo and Speed, 2018), the RLE plots were only conducted for genes with non-zero median expression.

'Gold-standard' DE genes

We compare the concordance of differentially-expressed genes (DEG) obtained from the different methods to the following 'gold standard' DEG:

- Celline: 'Gold standard' DEG in this case were derived by comparing Jurkat and 293t cells from batch 3, which has cells from both cell lines. The assumption is that cells assayed in the same batch will exhibit similar batch effects that will, to some extent, cancel when we compare cells of different types within batches. The DE analysis was performed using the Kruskal-Wallis test on the log(scran-normalized data + 1).

- Gaublomme: 'Gold standard' DEG here were derived from an external dataset. We downloaded the raw Affymetrix CEL files from the GEO website (ID: GSE39820). The microarray data were normalized using the `GCRMA` package version 2.58.0 and DE analysis comparing non-pathogenic (TGF-β1+IL-6) vs pathogenic (Il-1β1+IL-6+IL-23) microarray samples was performed using the `limma` package (Ritchie *et al.*, 2015).

- Pancreas: 'Gold standard' DEG here were also derived from an external dataset. Normalized Agilent microarray expression data were downloaded from `https://www.omicsdi.org/dataset/arrayexpress-repository/E-MTAB-465` and DE analysis comparing Alpha vs Beta cells was performed using `limma`.

Results

RUV-III-NB normalizes scRNA-seq data without removing biology

For the NSCLC study, library size is correlated with cell-type (Fig. 1A). The first two PC from the scran-normalized data manages to cluster Epithelial cells away from the immune cells (Fig. 1B), while RUV-III-NB logPAC further splits Monocytes off from the other immune cells (Fig. 1C) as does Dino, to an extent (Supp. Fig. 1). When we look within each cell-type, the normalized data from both RUV-III-NB, Dino and scTransform-Pearson have median correlations with log library size closest to zero (Fig. 1D), with RUV-III-NB (with $K = 1$) having correlation density the most tightly concentrated around zero. When this correlation is calculated across ALL cell-types, biological signals are involved and as expected, RUV-III-NB maintains higher correlation with log library size compared to scTransform-Pearson. In terms of principal components (PC), only RUV-III-NB, scTransform-Pearson and to an extent, Dino (Supp. Fig. 1), managed to reduce the correlation with log library size for homogeneous cells, with scTransform-Pearson having slightly lower correlation, especially for minor PC (Fig. 1E), but this comes at the expense of lower silhouette score for clustering by cell-type, compared with RUV-III-NB and Dino (Fig. 1F). This indicates that scTransform-Pearson removes some biology at the same time as removing library size effects, while RUV-III-NB and Dino performs normalization with minimal removal of biology, achieving the best biological silhouette score.

RUV-III-NB removes batch effects while preserving biology

For the other studies, batches are a major source of unwanted variation after library size. For these studies, UMI counts are available for all the studies except Gaublomme, which uses the SMART-Seq2 technology. The cell line (Fig. 2), CLL (Fig. 3) and Gaublomme (Fig. 4) studies all have at least one cell-type unavailable in some batches, thus creating an association between batch and biology. In addition to this, in the CLL, Gaublomme and Pancreatic (Fig. 5) studies there is a considerable difference in library size distributions across batches, creating an association between library size and batch. We summarize the characteristics of these studies in Table 1.

For the cell line study, after scran normalization, the leading PC still exhibit library size (Fig. 2A) and batch effects (Fig. 2B). RUV-III-NB removes the batch effects from the leading PC (Fig 2C) and so does scMerge (Supp. Fig. 2). MNNCorrect, Seurat3-Pearson, Seurat3-log corrected and ZINB-WaVE do not remove batch effects, while fastMNN and Seurat3-Integrated remove batch effects but also remove biology (Supp. Fig. 2). Within each cell type, RUV-III-NB normalized data has the lowest correlations with log library size for homogeneous cells (Fig. 2D), and so does scMerge when the number of unwanted factors (K) is ≥ 4. The performance of RUV-III-NB is not sensitive to the choice of K parameter once this parameter is set high enough. We saw this robustness across K in all of the studies in this paper.

Compared with the other methods, RUV-III-NB consistently has the highest biological silhouette Fig. 2E) and the lowest batch silhouette (Fig. 2F) and its performance is unaffected by the choice of K. Overall the performance of scMerge is most similar to RUV-III-NB, which is not surprising as both methods are based on the RUV-III algorithm (Molania *et al.*, 2019). Their difference is that scMerge uses a linear model on log-transformed count data while RUV-III-NB uses a generalized linear model directly on the counts. The 'approximate' linear model used by scMerge should be satisfactory for highly-abundant genes but it is expected to be less accurate for moderate to low abundance genes.

For the CLL study, the leading PC from the scran-normalized data clusters cells treated with DMSO away from the others (Fig. 3A). However, there are clear batch effects, especially among the DMSO-treated and Granta cells (Fig. 3B) which RUV-III-NB reduces, see (Fig. 3C). RUV-III-NB, scMerge, Seurat3-Pearson and Seurat3-Integrated all reduce the R^2 between PC and normalized data (Fig. 3D), but only RUV-III-NB, scMerge and Seurat3-Pearson improve their biological silhouettes when compared to scran-normalized data (Fig. 3E). scMerge's ability to improve the biological silhouette appears to be more sensitive to overestimation of K. RUV-III-NB, scMerge and Seurat3-Integrated reduce the batch silhouette relative to scran-normalized data, especially for the Granta cells (Fig. 3F). The performance of RUV-III-NB in terms of these metrics is stable when K varies, while scMerge seems to remove some biology when K becomes larger.

In the Gaublomme study, the first PC from scran-normalized data shows clustering by pathogenicity but batch effects are clearly visible in second PC (Fig. 4A). RUV-III-NB manages to remove the batch effects (Fig. 4B) as does scMerge, Seurat3-log corrected and fastMNN (Supp. Fig. 3). All methods, with the exception of MNNCorrect reduce the correlation between the leading PC and log library size (Fig. 4C) and also all methods reduce the clustering by batch (Fig 4E). scMerge, fastMNN and to an extent RUV-III-NB and MNNCorrect increase the biological silhouette while normalizing the data (Fig. 4D). The performance of RUV-III-NB in terms of these metrics appear to be stable for $K \geq 3$.

In the Pancreatic study, the leading PC of the scran-normalized data clearly shows that cells of the same type are split by the 'batch' variable, which in this case is the original study from which they came (Supp. Fig 4). RUV-III-NB, scMerge and Seurat3-Integrated integrate the different studies so that cells of the same type are clustered together (Supp. Fig 4). RUV-III-NB, scMerge and Seurat3-Pearson consistently manage to reduce the correlation between normalized data and log library size for homogeneous cell types (Fig. 5A). In terms of R^2 between leading PC and log library size, scMerge and Seurat3-Integrated have the lowest association with log library size, followed by RUV-III-NB (Fig. 5B). Seurat3-Integrated is the most successful in improving the biological silhouette, followed by RUV-III-NB and scMerge (Fig. 5C) and it also has the lowest silhouette for clustering by batch, followed by scMerge and RUV-III-NB (Fig. 5D).

In terms of relative log expression (RLE), the RUV-III-NB normalized data also has lower correlations with log library size and batch effects (Supp. Fig 5).

RUV-III-NB accommodates size heterogeneity within a cell type

For cells of the same type in the NSCLC and Pancreas studies, RUV-III-NB normalized data show a slightly higher correlation between the leading PC and log library size, when compared with scTransform-Pearson (Fig. 1E) and scMerge and Seurat3-Integrated (Fig. 5B). It is worth noting that these two studies have UMI counts, and so library size corresponds more closely to the number of molecules inside a cell, and hence cell size. We investigated whether the higher correlation between log library size and PC exhibited by RUV-III-NB is biologically meaningful. Towards this end, for the NSCLC data, we performed DE analysis comparing monocytes with smaller ($<$ median) vs larger ($\geq$ median) library size. The results show that RUV-III-NB has the smallest number of DE genes (Fig. 6A) and a KEGG pathway analysis found that only DEG from RUV-III-NB logPAC and scTransform-log corrected are significantly enriched with terms from the phagosome pathway (Supp. Fig. 6). This is consistent with Wang *et al.* (1992) who reported that larger monocytes have increased phagocytic activity. For the pancreatic study, we carried out a similar analysis in comparing beta cells with above and below median library sizes from the batch corresponding to Baron *et al.* (2016)'s experiment. Sasaki *et al.* (2021) reported that patients with type II diabetes have reduced size beta cells. We found that that only the DEG using counts adjusted by RUV-III-NB were significantly enriched in terms from the insulin resistance pathway (Supp. Fig. 7). We concluded that the higher correlation between log library size and PC exhibited by RUV-III-NB is biologically-meaningful and is a consequence of size-related heterogeneity. This conclusion is also strengthened by the fact that for the Gaublomme study, which does not have UMI counts (Fig. 4), the PC of RUV-III-NB normalized data do not show a higher correlation with log library size. This is likely because library size does not correspond closely to cell size for non-UMI data, making it less meaningful biologically. The cell line study (Fig. 2) has UMI counts but we expect cells from the same cell line to be very homogeneous, and as expected the PC of RUV-III-NB normalized data within a particular cell-type do not show any association with log library size.

RUV-III-NB reduces false discoveries and improves concordance with 'gold standard' DEG

For datasets with multiple batches, we carried out a DE analysis comparing cells of the same type across batches. The DE analysis was performed using the Kruskal-Wallis test on the counts adjusted by the different methods. The proportion of DEG was then estimated from the p-values using Storey's method

(Storey and Tibshirani, 2003). The results show that for the Gaublomme study, scMerge and Seurat3 have the highest proportion of null (i.e. non-DE) genes, followed by RUV-III-NB (Fig. 4F), but it is important to note that the success of Seurat3 comes at the expense of removing biological signals (Fig. 4D). Meanwhile, for studies with multiple batches where UMI counts are available (Supp. Fig. 8,9 and 10), RUV-III-NB clearly has the highest proportion of null genes and hence the lowest proportion of DEG when comparing the same cell type across batches.

For the cell line, Gaublomme and Pancreatic studies, we also compared the concordance of DEG based on counts normalized by the different methods with the 'gold standard' DEG. For the cell line study, the DEG are for the 293t vs Jurkat cell comparison, for the Gaublomme study we compare pathogenic vs sorted non-pathogenic Th-17 cells, while for the Pancreatic study we compare alpha and beta cells. We found that for the cell line and Gaublomme studies where batch is associated with biology, RUV-III-NB has the best concordance (Figs. 6A-B), while for the Pancreatic study (Fig. 6C) where batch and biology are not associated, none of the batch-effect removal methods improve on scran normalization, with RUV-III-NB ranking second after Seurat3 with log-corrected counts.

RUV-III-NB: Overall Performance

In Supp. Figs 11-15, we provide Wheel diagrams that summarize the overall performance of the different methods according to the benchmarking criteria. Each sector corresponds to a particular method and each segment within a sector to a particular criterion, with longer segments indicating better performance according to that criterion. RUV-III-NB achieves the best overall performance for all UMI datasets except the Pancreatic study. Two aspects where RUV-III-NB appears to have a distinct advantage over the other methods are its performance on the DE and RLE criteria. For the Pancreatic dataset, the Seurat3 methods have better biological and technical silhouettes but RUV-III-NB has better DE and RLE characteristics. For the read count data (Fig. 14), RUV-III-NB's performance is comparable to MNNCorrect and slightly lower than scMerge and fastMNN.

Sensitivity Analyses

The RUV-III-NB algorithm require users to specify the negative control gene set and the number of unwanted factors. Using the cell line dataset, we investigate the sensitivity of the key performance metrics against these parameters. We use five different strategies to identify the negative control gene set and varying K from 1 to 20. Supp. Fig. 16A demonstrate that for four negative control gene sets, including set 2 that uses the default single-cell housekeeping genes, the R^2 between log library size and leading principal component of normalized data is relatively robust when K is increased and thus potentially overestimated.

Set 4, in which the negative control gene set was identified as non-DEG from the batch with two cell lines (batch 3), is the only one where the R^2 is affected by overestimation of K. In terms of average batch (Supp Fig. 16B) and biological silhouette width (Supp Fig. 16C), its performance is quite similar across different negative control gene sets, for $K \geq 2$.

Discussion

Single-cell RNA-seq offers us an unparalleled opportunity to advance our understanding of the transcriptome at single cell level. However, scRNA-seq data contains significant amounts of unwanted variation that, when left unaddressed, may compromise downstream analyses. Most methods for removing unwanted variation from scRNA-seq data implicitly assume that the unwanted factors are at worst weakly associated with the biological signals of interest. In this paper, we have proposed RUV-III-NB, a statistical method for normalizing scRNA-seq data which does not make this assumption. The method adjusts for unwanted variation using pseudo-replicate sets, which should ensure that it does not remove too much biology when biology and unwanted variation are associated. Using publicly available data from five studies we show this to be the case. We also show that for clustering-based analyses RUV-III-NB successfully adjusts counts for library size and batch variation and improves the biological signals.

RUV-III-NB can be used for both UMI and read count data but its improvement relative to other methods is especially evident for sparse UMI count data. When using RUV-III-NB users need to specify the number of unwanted factors in the data (K) and the set of negative control genes. We have shown that RUV-III-NB performance is relatively robust to overestimation of K and the choice of negative control gene sets. While RUV-III-NB is developed primarily to remove within-study batch effects, it can also be used to integrate datasets from different studies where platform difference is a major source of unwanted variation. Using the pancreatic islet data, we have shown that the performance of RUV-III-NB for data integration purposes is quite competitive .

A unique feature of RUV-III-NB is that it returns a sequencing count after adjusting for the unwanted variation. We call this the *percentile-invariant adjusted count (PAC)* and it can be used to perform downstream analyses such as differential expression (DE), cell-type annotation and pseudotime analyses. In this paper, we have shown that when used for DE analysis, it delivers good control of false discoveries and improved power to detect 'gold standard' DE genes. In the vignette that accompanies the R package, we also demonstrated how the corrected count can be used to perform cell-type annotation.

RUV-III-NB has acceptable computing time when run in High-Performance Computing (HPC) environment. For the examples used in this paper, the running time on HPC environment with 15 cores and 120 Gb total RAM (8Gb RAM per core), ranges from approximately 120 minutes for the CLL dataset with around 1,650 cells to around 280 minutes for the Pancreatic dataset with more than 10,000 cells (Fig. 17). The running time does not increase as we increase the number of unwanted factors (K) and it is approximately a square root, rather than a linear function of the number of cells.

Acknowledgement

The authors would like to thank Jean Yang, Yingxin Lin and Xiangnan Xu for their suggestions that have improved the quality of this manuscript.

References

Aran, D., Looney, A. P., Liu, L., Wu, E., Fong, V., Hsu, A., Chak, S., Naikawadi, R. P., Wolters, P. J., Abate, A. R., Butte, A. J., and Bhattacharya, M. (2019). Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage. *Nat. Immunol.*, **20**, 163–172.

Bacher, R., Chu, L.-F., Leng, N., Gasch, A. P., Thomson, J. A., Stewart, R. M., Newton, M., and Kendziorski, C. (2017). Scnorm: robust normalization of single-cell rna-seq data. *Nature Methods*, **14**, 584.

Baglama, J., Reichel, L., and Lewis, B. W. (2019). *irlba: Fast Truncated Singular Value Decomposition and Principal Components Analysis for Large Dense and Sparse Matrices*. R package version 2.3.3.

Baron, M., Veres, A., Wolock, S. L., Faust, A. L., Gaujoux, R., Vetere, A., Ryu, J. H., Wagner, B. K., Shen-Orr, S. S., Klein, A. M., Melton, D. A., and Yanai, I. (2016). A single-cell transcriptomic map of the human and mouse pancreas reveals inter- and intra-cell population structure. *Cell Systems*, **3**(4), 346–360.e4.

Blondel, V. D., Guillaume, J.-L., Lambiotte, R., and Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, **2008**(10), P10008.

Brown, J., Ni, Z., Mohanty, C., Bacher, R., and Kendziorski, C. (2021). Normalization by distributional resampling of high throughput single-cell RNA-sequencing data. *Bioinformatics*. btab450.

Dunn, P. K. and Smyth, G. K. (1996). Randomized quantile residuals. *Journal of Computational and Graphical Statistics*, **5**(3), 236.

Gagnon-Bartsch, J. A. and Speed, T. P. (2012). Using control genes to correct for unwanted variation in microarray data. *Biostatistics*, **13**(3), 539–552.

Gandolfo, L. C. and Speed, T. P. (2018). RLE plots: Visualizing unwanted variation in high dimensional data. *PLoS One*, **13**(2), e0191629.

Gaublomme, J. T., Yosef, N., Lee, Y., Gertner, R. S., Yang, L. V., Wu, C., Pandolfi, P. P., Mak, T., Satija, R., Shalek, A. K., Kuchroo, V. K., Park, H., and Regev, A. (2015). Single-Cell Genomics Unveils Critical Regulators of Th17 Cell Pathogenicity. *Cell*, **163**(6), 1400–1412.

Hafemeister, C. and Satija, R. (2019). Normalization and variance stabilization of single-cell rna-seq data using regularized negative binomial regression. *Genome Biology*, **20**(1), 296.

Haghverdi, L., Lun, A. T. L., Morgan, M. D., and Marioni, J. C. (2018). Batch effects in single-cell rna-sequencing data are corrected by matching mutual nearest neighbors. *Nature Biotechnology*, **36**(5), 421–427.

Jacob, L., Gagnon-Bartsch, J. A., and Speed, T. P. (2015). Correcting gene expression data when neither the unwanted variation nor the factor of interest are observed. *Biostatistics*, **17**(1), 16–28.

Korsunsky, I., Millard, N., Fan, J., Slowikowski, K., Zhang, F., Wei, K., Baglaenko, Y., Brenner, M., Loh, P.-r., and Raychaudhuri, S. (2019). Fast, sensitive and accurate integration of single-cell data with harmony. *Nature Methods*, **16**(12), 1289–1296.

Lin, Y., Ghazanfar, S., Strbenac, D., Wang, A., Patrick, E., Lin, D. M., Speed, T., Yang, J. Y. H., and Yang, P. (2019a). Evaluating stably expressed genes in single cells. *GigaScience*, **8**(9).

Lin, Y., Ghazanfar, S., Wang, K. Y. X., Gagnon-Bartsch, J. A., Lo, K. K., Su, X., Han, Z.-G., Ormerod, J. T., Speed, T. P., Yang, P., and Yang, J. Y. H. (2019b). scmerge leverages factor analysis, stable expression, and pseudoreplication to merge multiple single-cell rna-seq datasets. *Proceedings of the National Academy of Sciences*, **116**(20), 9775–9784.

Lun, A. T., Bach, K., and Marioni, J. C. (2016). Pooling across cells to normalize single-cell rna sequencing data with many zero counts. *Genome Biology*, **17**(1), 75.

McCarthy, D. J., Campbell, K. R., Lun, A. T., and Wills, Q. F. (2017). Scater: pre-processing, quality control, normalization and visualization of single-cell RNA-seq data in R. *Bioinformatics*, **33**(8), 1179–1186.

Molania, R., Gagnon-Bartsch, J. A., Dobrovic, A., and Speed, T. P. (2019). A new normalization for Nanostring nCounter gene expression data. *Nucleic Acids Research*, **47**(12), 6073–6083.

Muraro, M. J., Dharmadhikari, G., Grün, D., Groen, N., Dielen, T., Jansen, E., van Gurp, L., Engelse, M. A., Carlotti, F., de Koning, E. J., and van Oudenaarden, A. (2016). A Single-Cell Transcriptome Atlas of the Human Pancreas. *Cell Syst*, **3**(4), 385–394.

Risso, D., Perraudeau, F., Gribkova, S., Dudoit, S., and Vert, J.-P. (2018). A general and flexible method for signal extraction from single-cell rna-seq data. *Nature Communications*, **9**(1), 284.

Ritchie, M. E., Phipson, B., Wu, D., Hu, Y., Law, C. W., Shi, W., and Smyth, G. K. (2015). limma powers differential expression analyses for rna-sequencing and microarray studies. *Nucleic Acids Research*, **43**(7), e47–e47.

Robinson, M. D. and Oshlack, A. (2010). A scaling normalization method for differential expression analysis of rna-seq data. *Genome Biology*, **11**(3), R25.

Sasaki, H., Saisho, Y., Inaishi, J., Watanabe, Y., Tsuchiya, T., Makio, M., Sato, M., Nishikawa, M., Kitago, M., Yamada, T., and Itoh, H. (2021). Reduced beta cell number rather than size is a major contributor to beta cell loss in type 2 diabetes. *Diabetologia*, **64**(8), 1816–1821.

Storey, J. D. and Tibshirani, R. (2003). Statistical significance for genomewide studies. *Proceedings of the National Academy of Sciences*, **100**(16), 9440–9445.

Stuart, T., Butler, A., Hoffman, P., Hafemeister, C., Papalexi, E., Mauck, W. M., Hao, Y., Stoeckius, M., Smibert, P., and Satija, R. (2019). Comprehensive integration of single-cell data. *Cell*, **177**(7), 1888–1902.e21.

Vallejos, C. A., Risso, D., Scialdone, A., Dudoit, S., and Marioni, J. C. (2017). Normalizing single-cell rna sequencing data: challenges and opportunities. *Nature methods*, **14**(6), 565–571.

Wang, S. Y., Mak, K. L., Chen, L. Y., Chou, M. P., and Ho, C. K. (1992). Heterogeneity of human blood monocyte: two subpopulations with different sizes, phenotypes and functions. *Immunology*, **77**(2), 298–303.

Zhang, Y., Parmigiani, G., and Johnson, W. E. (2020). ComBat-seq: batch effect adjustment for RNA-seq count data. *NAR Genomics and Bioinformatics*, **2**(3).

Ziegenhain, C., Vieth, B., Parekh, S., Reinius, B., Guillaumet-Adkins, A., Smets, M., Leonhardt, H., Heyn, H., Hellmann, I., and Enard, W. (2017). Comparative analysis of single-cell rna sequencing methods. *Molecular Cell*, **65**(4), 631–643.e4.

Table 1: Characteristics of datasets used for benchmarking

Study	Platform	UMI	Unwanted Fac.	Biological Fac.	Pseudo-Reps strategy	LS × Batch	LS × Bio	Batch × Bio
NSCLC	10x	Yes	LS	Cell-type	Pseudocells	No	Yes	No
Cell line	10x	Yes	LS, batch	Cell lines	MNCs from scMerge::scReplicate	No	No	Yes
CLL	Celseq2	Yes	LS, plate	Treatment	Granta+pseudocells	Yes	Yes	Yes
Gaublomme	SmartSeq2	No	LS, batch	Pathogenicity	Actual cell type	Yes	No	Yes
Pancreatic	inDrop,CelSeq2	Yes	LS, batch	cell-type	MNCs from scMerge::scReplicate	Yes	No	No

LS = library size, LS×Batch = presence of LS and batch association, LS×Bio = Presence of LS and biological factor association, Batch×Bio = Presence of batch and biological factor association

Supp. Table 1: Data Assays Used for Calculating Benchmarking Metrics

Methods	Genewise Corr	Silhouette&PC	DE	RLE
fastMNN	reconstructed	reconstructed	reconstructed	reconstructed
MNNCorrect	corrected	corrected	corrected	corrected
scMerge	scMerge	scMerge	scMerge	scMerge
sctransfom-Pearson	y	y	y	y
sctransfom-log corrected	log(umi_corrected+1)	log(umi_corrected+1)	log(umi_corrected+1)	log(umi_corrected+1)
Seurat3-Pearson	scale.data@SCT	scale.data@SCT	scale.data@SCT	scale.data@SCT
Seurat3-log corrected	log(count@SCT+1)	log(count@SCT+1)	log(count@SCT+1)	log(count@SCT+1)
Seurat3-Integrated	NA	scale.data@Integrated	NA	NA
RUV-III-NB	logPAC	logPAC	logPAC	logPAC
ZINB-Wave	normalizedValues	normalizedValues	normalizedValues	normalizedValues

Genewise Corr: Correlation between normalized gene expression and a factor e.g log library size, calculated gene-by-gene.
PC: Principal Component
DE: Differential Expression Analysis.
RLE = Relative Log of Normalized Expression.
PAC = Percentile adjusted count

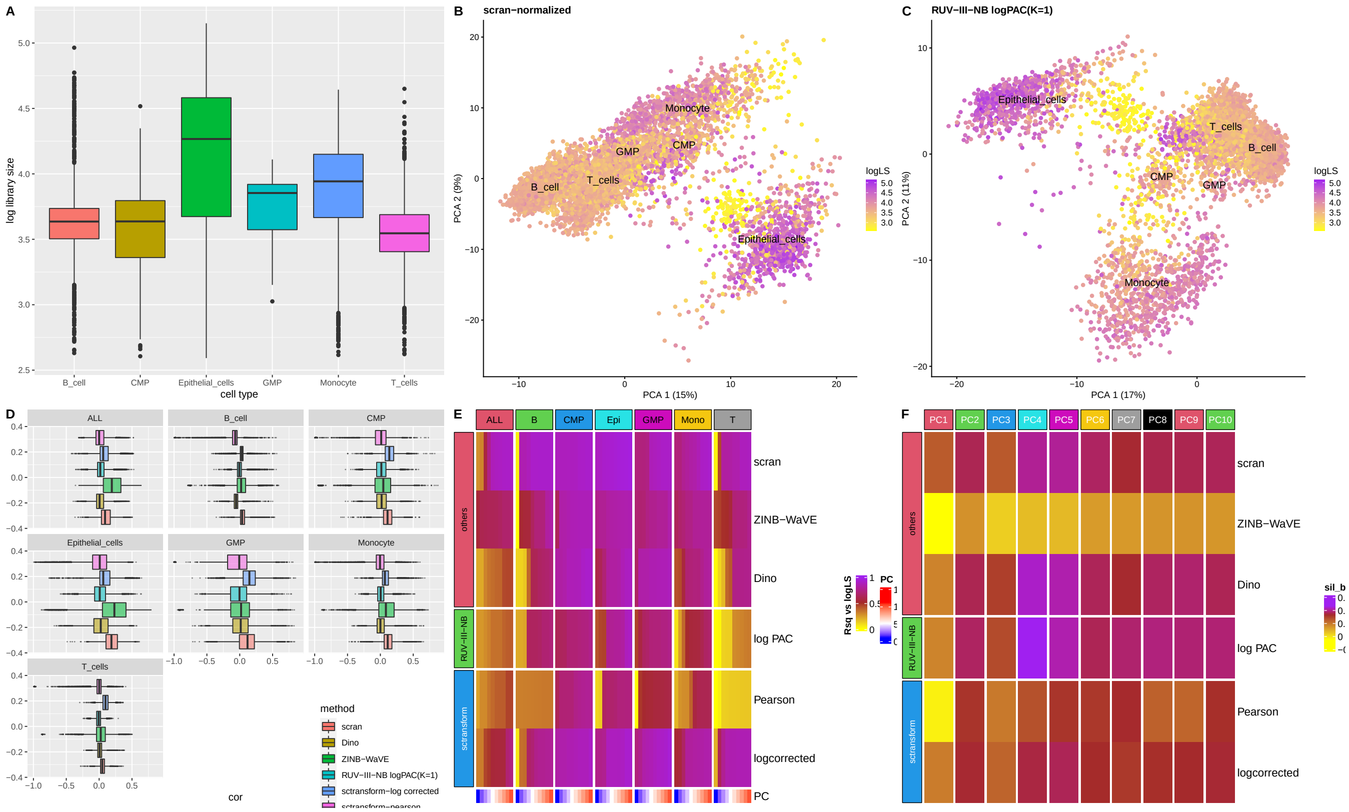


Fig. 1: NSCLC study. (A) Log library size distributions by cell type. These show that library size is biologically meaningful. (B) PC of scran-normalized data. Colour refers to log library size. (C) PC of RUV-III-NB log percentile adjusted count (PAC). These show that Monocytes are better separated from the rest of the cells. (D) Density plots of Spearman correlations between log library size and normalized data for ALL cells together and each cell type separately. RUV-III-NB has the most concentrated density around zero for ALL cells and for most of the other cell types. (E) Heatmap of R-squared between logLS and PC of normalized data. RUV-III-NB and sctransform-Pearson have the lowest correlation, while RUV-III-NB still retains some of the size-related heterogeneity within a cell type. (F) Biological silhouette. RUV-III-NB is the only method that improves the biological silhouette over scran normalization.

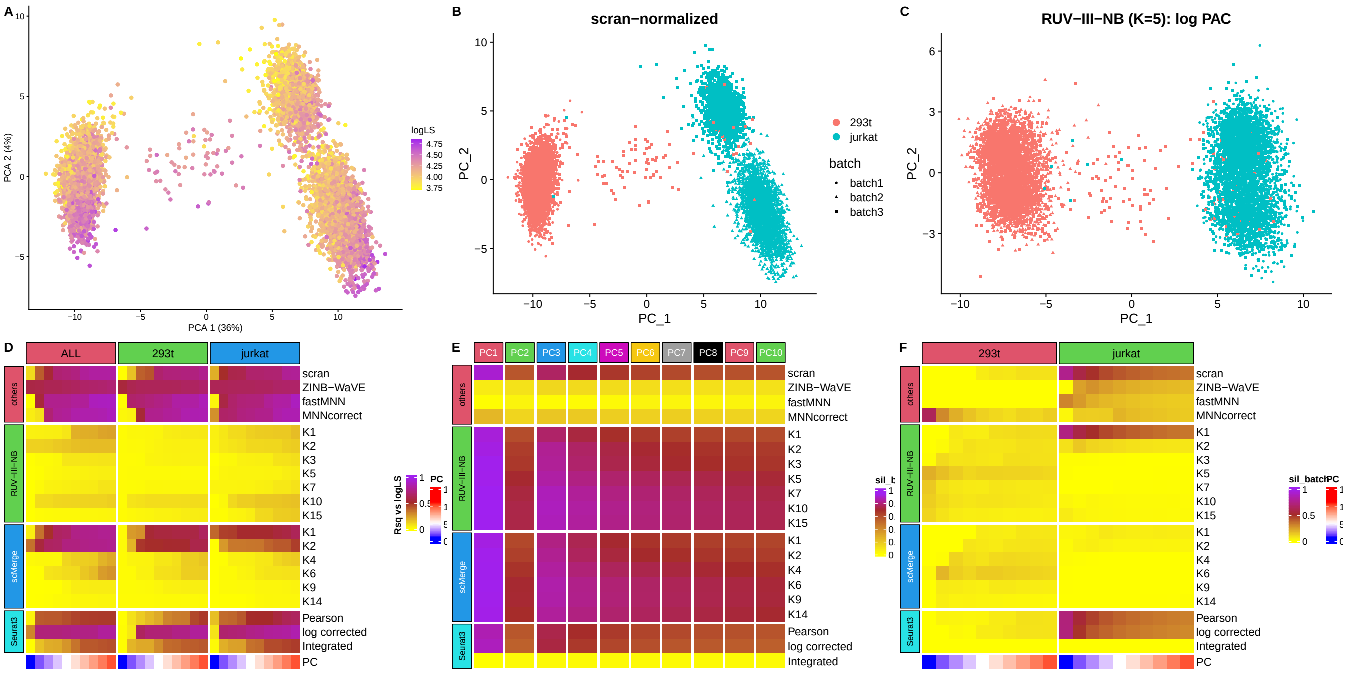


Fig. 2: Cell line study. (A) The first two PC of scran-normalized data. Colour refers to log library size. The library size effect is clearly visible in the PC. (B) PC of scran-normalized data. Colour refers to cell type. Batch effects are visible for the Jurkat cells. (C) PC of RUV-III-NB log percentile adjusted counts (PAC). Clustering by cell type is clearly visible with batch effects removed. (D) Heatmap of R-squared between logLS and PCs of normalized data. For the Jurkat cells, scran and Seurat3 failed to remove the correlation, while MNNcorrect introduces a correlation in 293t cells. RUV-III-NB and scMerge remove correlation for both cell types. (E) Average biological silhouette score. RUV-III-NB and scMerge improve the biological signal and increasing the number of unwanted factors beyond a certain point only slightly degrades performance. (F) Non-biological silhouette. RUV-III-NB and scMerge successfully reduce the non-biological signal within each cell type.

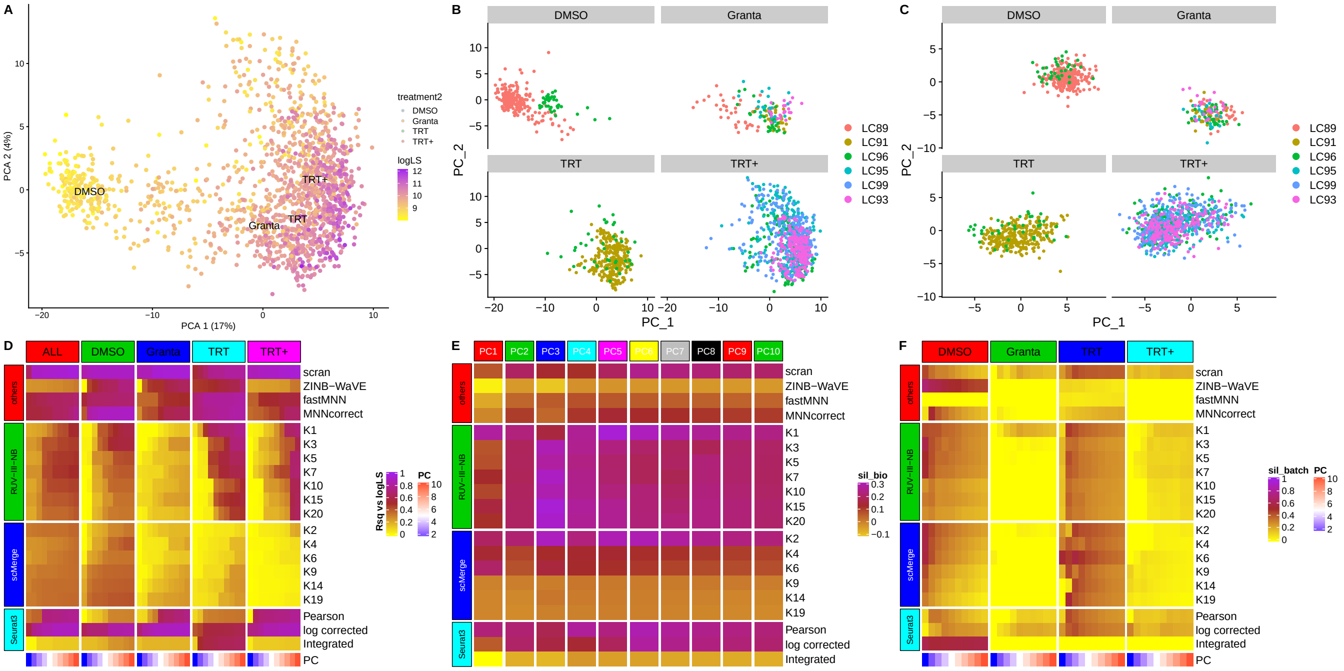


Fig. 3: CLL study.(A) The first two PC of scran-normalized data. Colour refers to log library size. The shallower sequencing of the DMSO-treated cells is clearly visible. (B) PC of scran-normalized data faceted by cell type. Colours refers to plate identity. The plate effects among DMSO and Granta cells are clearly visible. (C) PC of RUV-III-NB log PAC ($K = 20$) faceted by cell type. Colours refers to plate identity. No plate effects are visible. (D) Heatmap of R-squared between logLS and PC of normalized data. scMerge has the lowest correlation, followed by RUV-III-NB. (E) Biological silhouette scores. RUV-III-NB, scMerge and Seurat3 improve the biological silhouette score compared to scran, but scMerge's performance is sensitive to the choice of K. (F) Non-biological silhouette for each cell-type. RUV-III-NB and scMerge have the lowest silhouette score, followed closely by Seurat3.

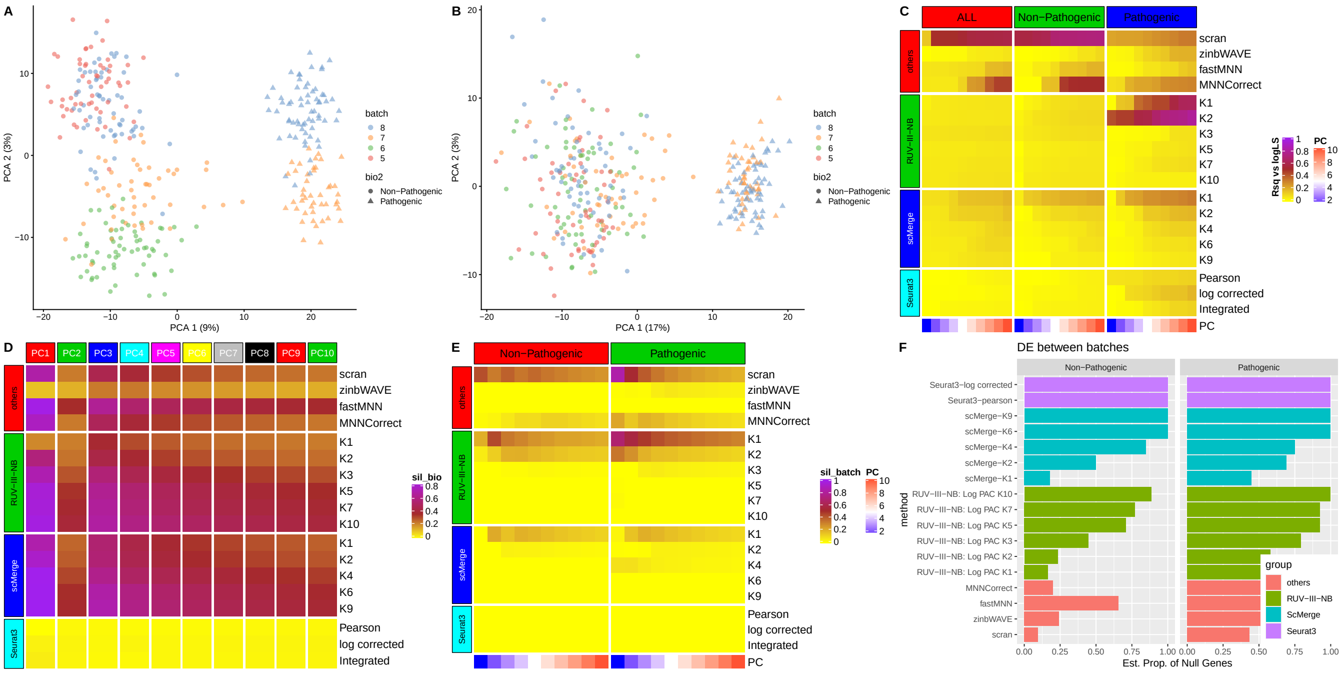


Fig. 4: Gaublomme study. (A) The first PC of scran-normalized data clearly shows the separation of pathogenic from non-pathogenic cells, but batch effects dominate the second PC. (B) PC of RUV-III-NB Pearson residuals ($K = 5$). Batch effect are no longer visible. (C) Heatmap of R-squared between logLS and PC of normalized data. All methods, except MNNCorrect reduce the correlation with logLS when compared to scran-normalized data. (D) Biological silhouette scores. RUV-III-NB improves the biological silhouette scores when compared to scran-normalization, followed by scMerge. The other methods remove biological signal in the process of removing unwanted variation. (E) Non-biological silhouette scores for each cell-type. All methods reduce the non-biological signal when compared with scran normalization. (F) Estimated proportion of non-DEG when comparing the same cell types across batches. Seurat3 and scMerge has the highest proportion of non DEG, followed by RUV-III-NB.

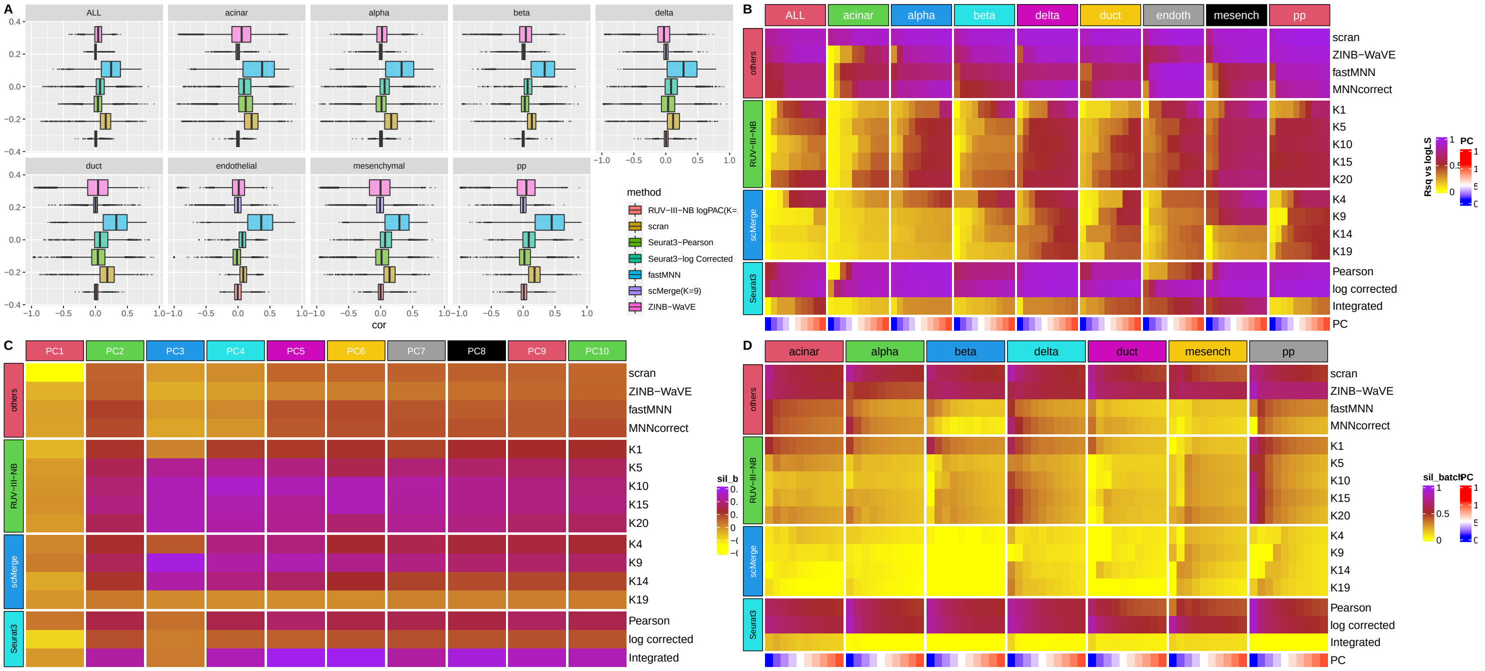


Fig. 5: Pancreas study. (A) Densities of Spearman correlations between log library size and normalized data for ALL and each cell type. RUV-III-NB has the most concentrated density around zero, followed by scMerge. (B) Heatmap of R-squared between logLS and PCs of normalized data. RUV-III-NB and scMerge have the lowest correlation, with RUV-III-NB still retaining some of the size-related heterogeneity within a cell type. (C) Biological silhouette score. Seurat3-Integrated has the best biological silhouette score, followed by RUV-III-NB (D) Non-biological silhouette scores for each cell-type. scMerge has the lowest silhouette, followed by RUV-III-NB. RUV-III-NB has a higher silhouette because it maintains some of the size-related heterogeneity within cell type.

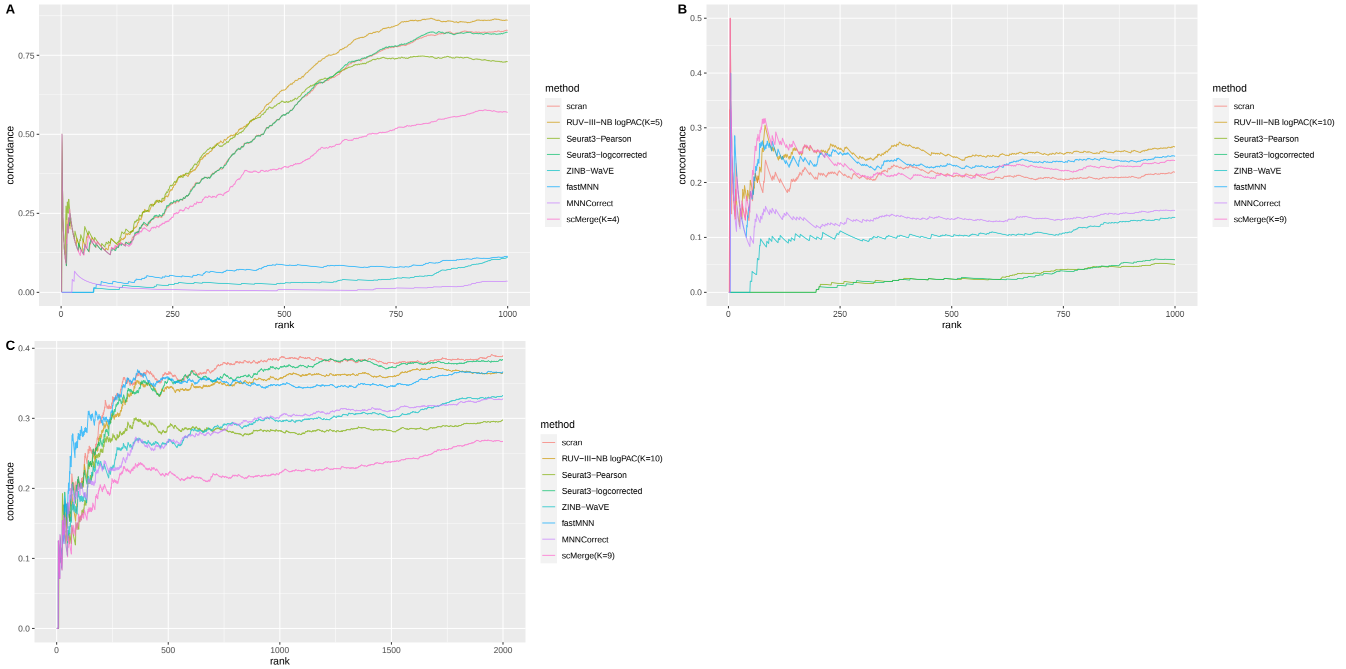
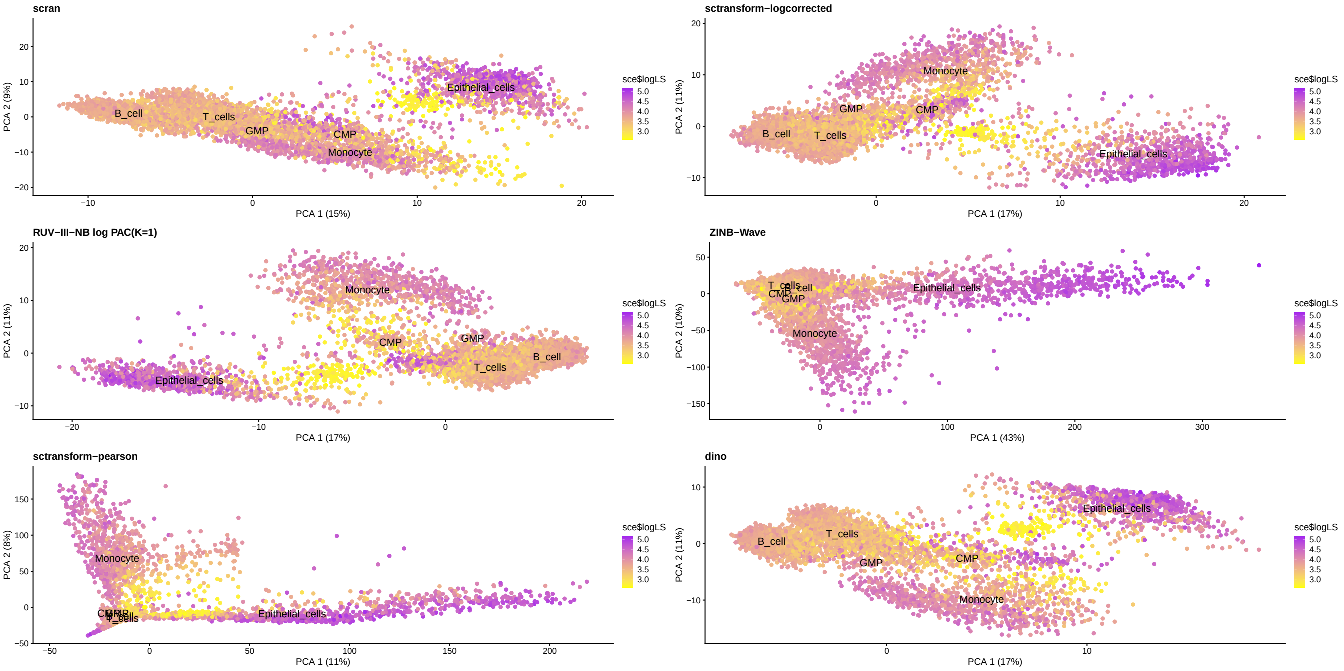
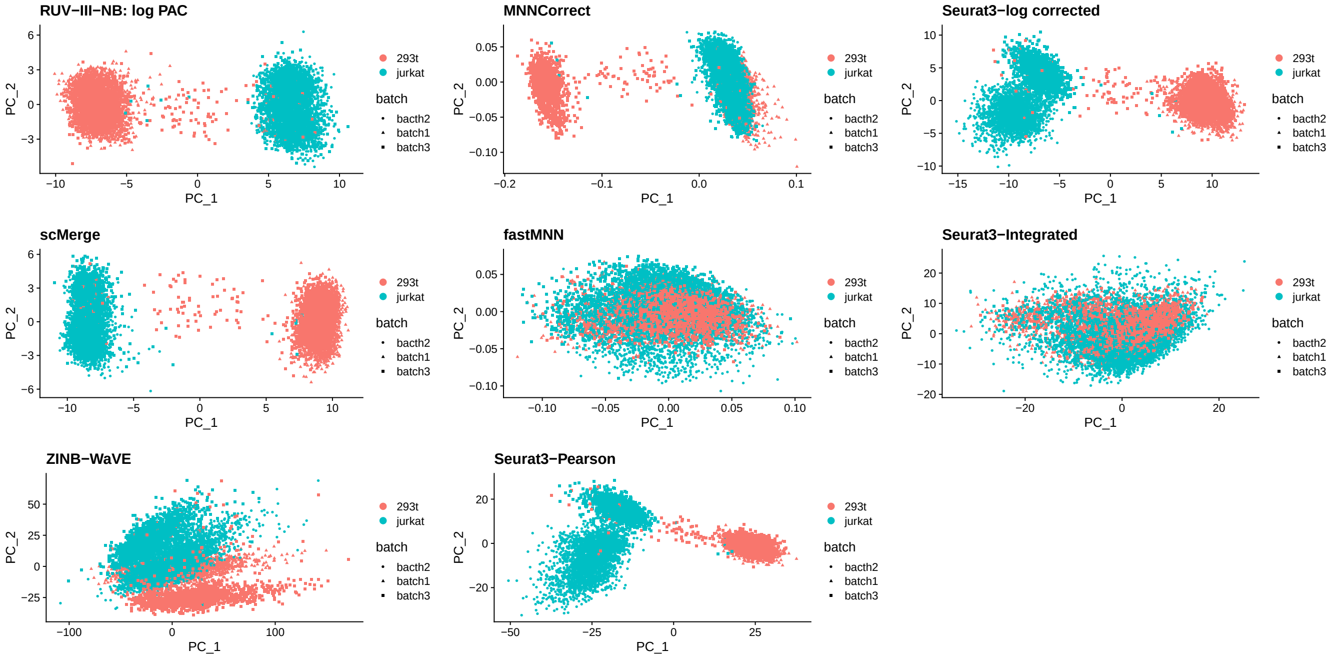


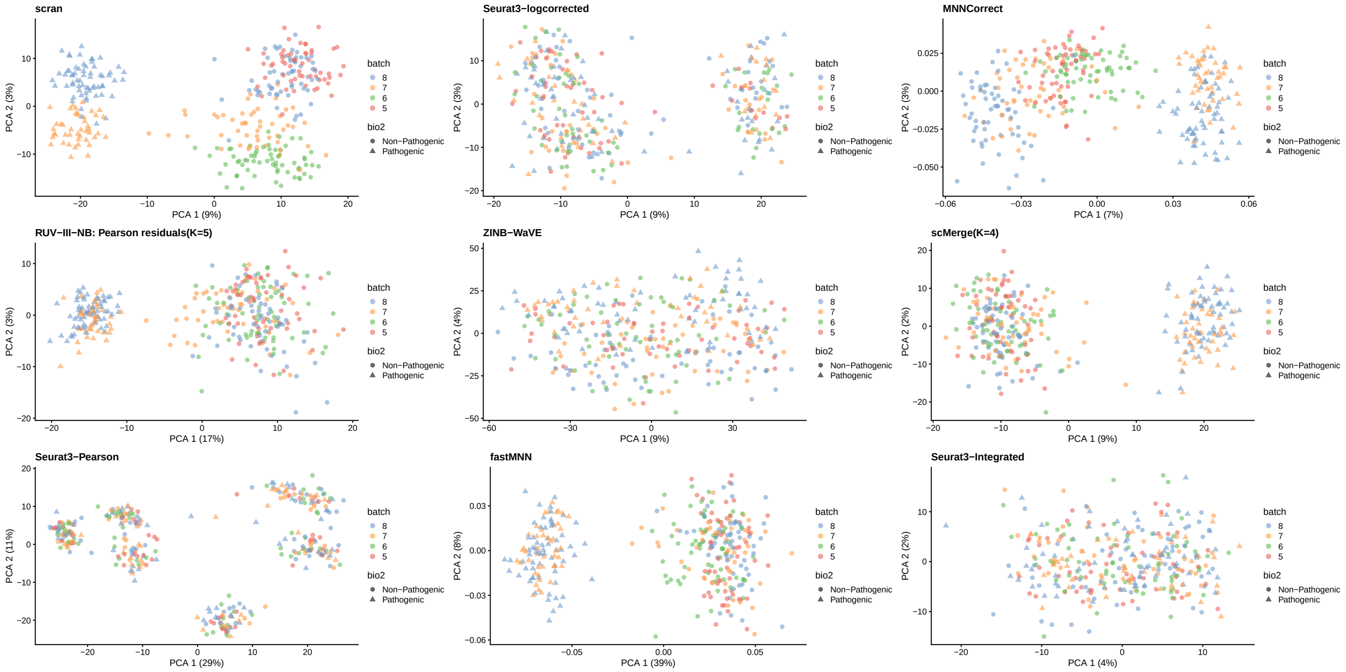
Fig. 6: Concordance of DEG. (A) Jurkat cells in the cell line study. RUV-III-NB has the best concordance, followed by Seurat3. (B) Pathogenic vs Sorted Non-Pathogenic cells in the Gaublomme study. RUV-III-NB has the best concordance followed by fastMNN and scMerge. (C) Alpha vs Beta cells in the Pancreatic study. scran has the best concordance followed by Seurat3 and RUV-III-NB.



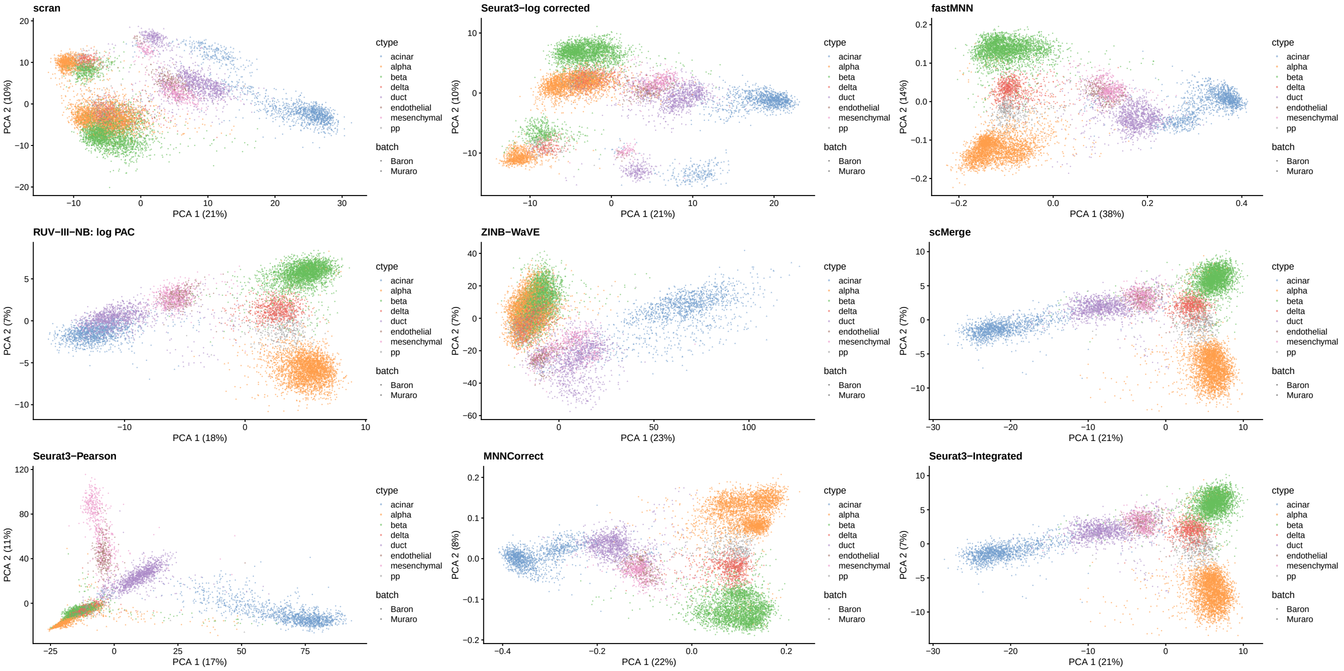
Supp. Fig. 1: NSCLC study. Leading PC of the differently normalized data. Coloured by log library size.



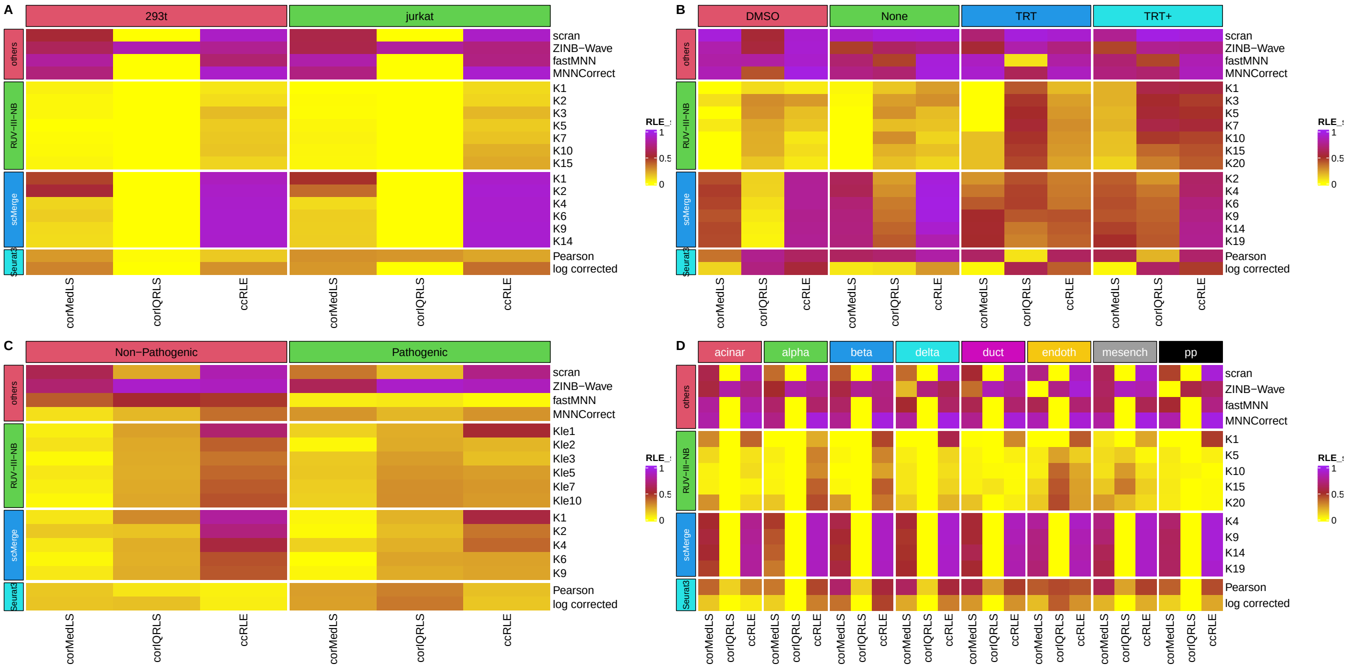
Supp. Fig. 2: Cell line study. Leading PC of the differently normalized data. Coloured by cell types.



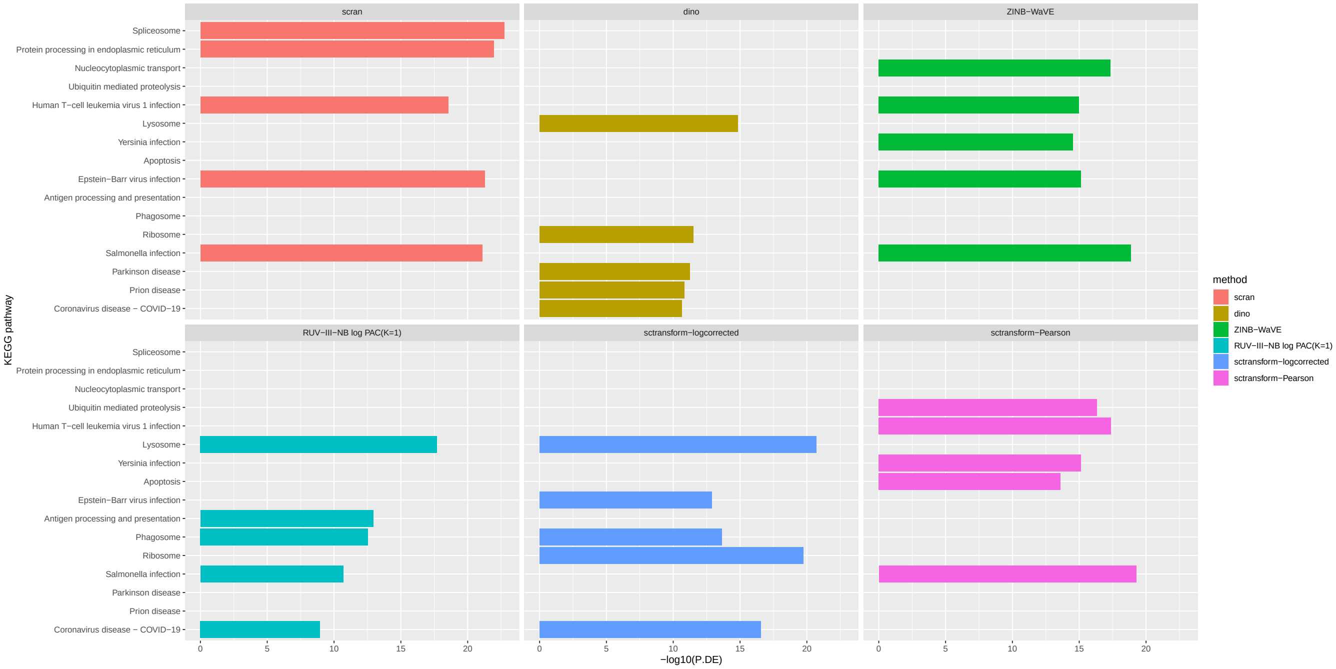
Supp. Fig. 3: Gaublomme study. Leading PC of the differently normalized data. Coloured by batches, while shape refers to cell type.



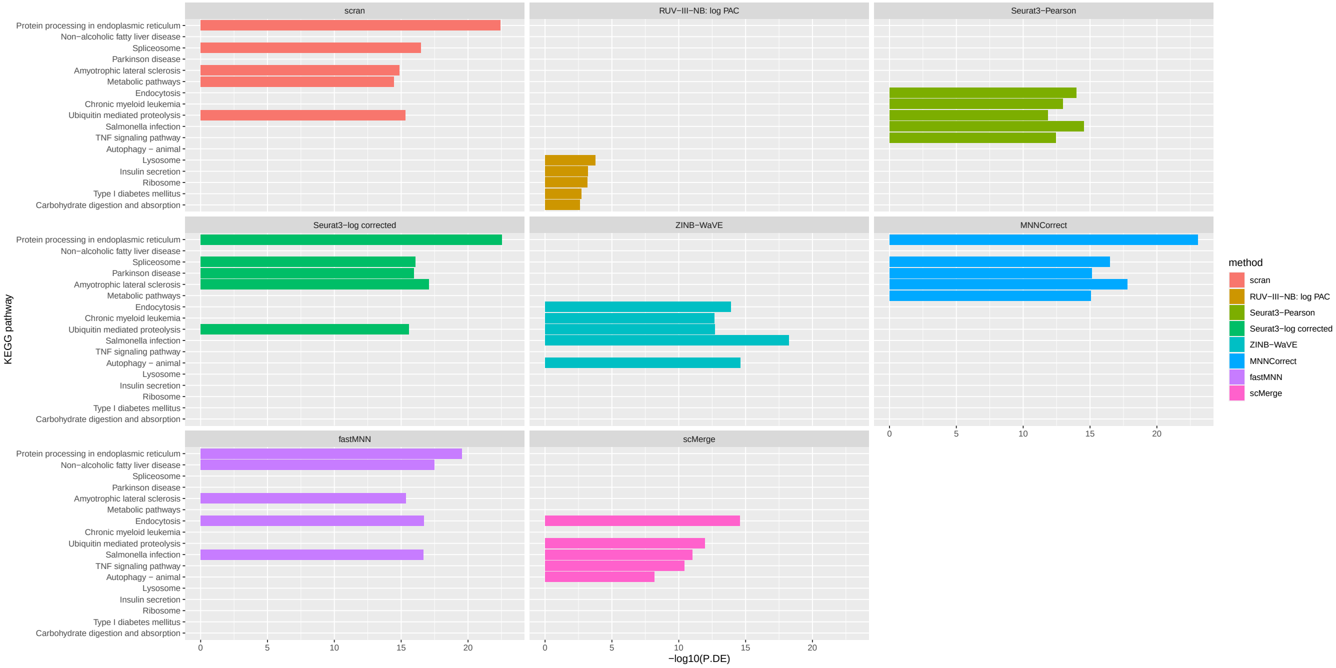
Supp. Fig. 4: Pancreatic study. Leading PC of differently normalized data. Coloured by cell type, while shape refers to batch.



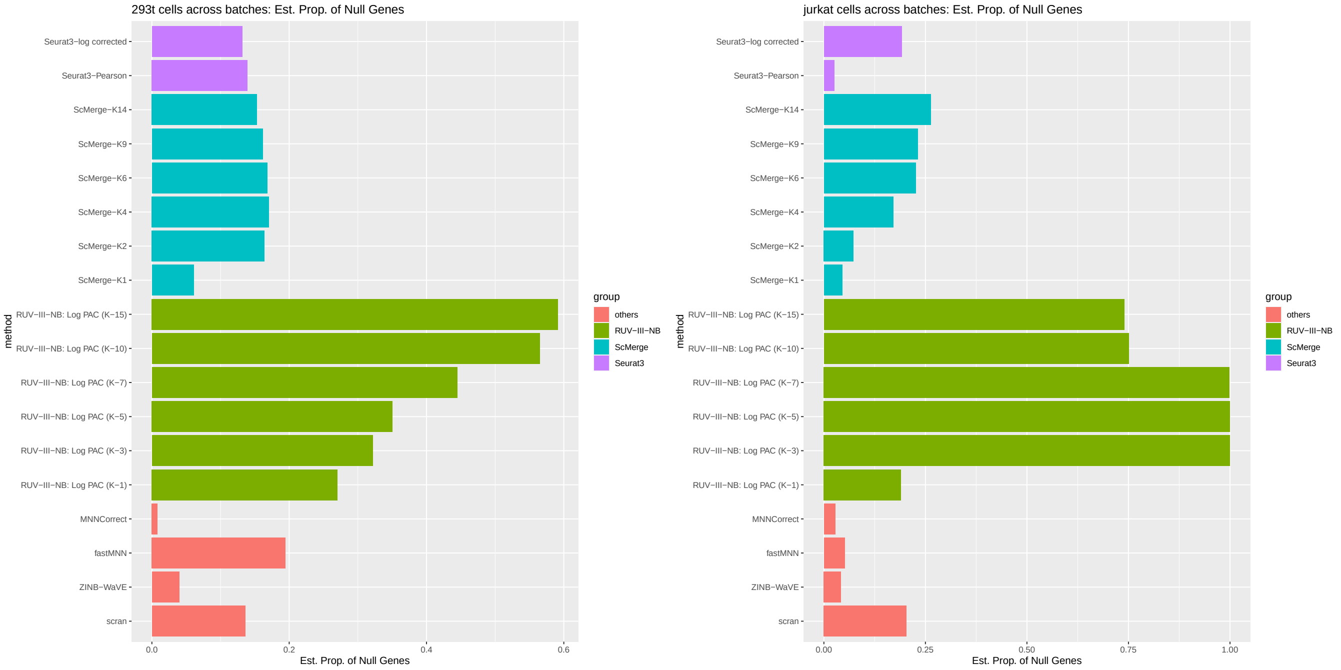
Supp. Fig. 5: Heatmap of correlations between RLE plot medians and log library sizes (first column), correlations between RLE plot IQR and log library sizes (second column) and canonical correlation between (RLE plot median, RLE plot IQR) and (log LS , batch variables) (third column), all stratified by cell type. (A) Cell line study. (B) CLL study. (C) Gaublomme study. (D) Pancreatic study.



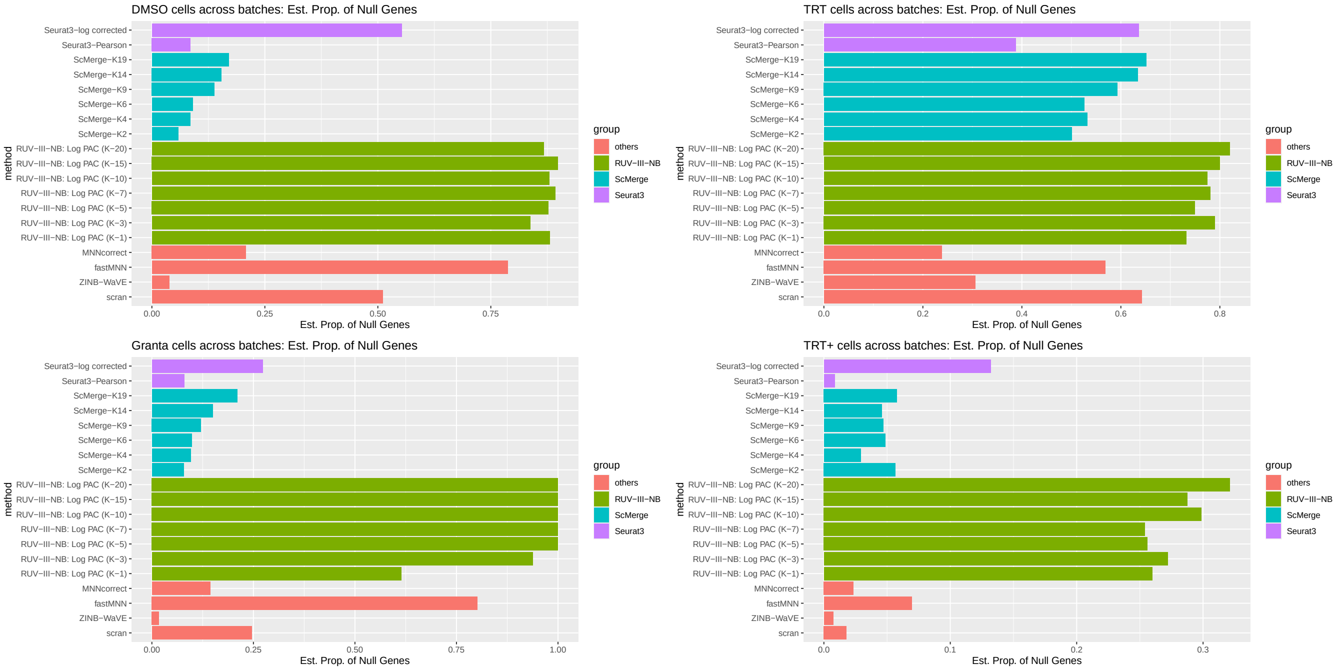
Supp. Fig. 6: Top 5 KEGG Pathways among DEG calculated from normalized data from the NSCLC study: larger vs smaller Monocytes.



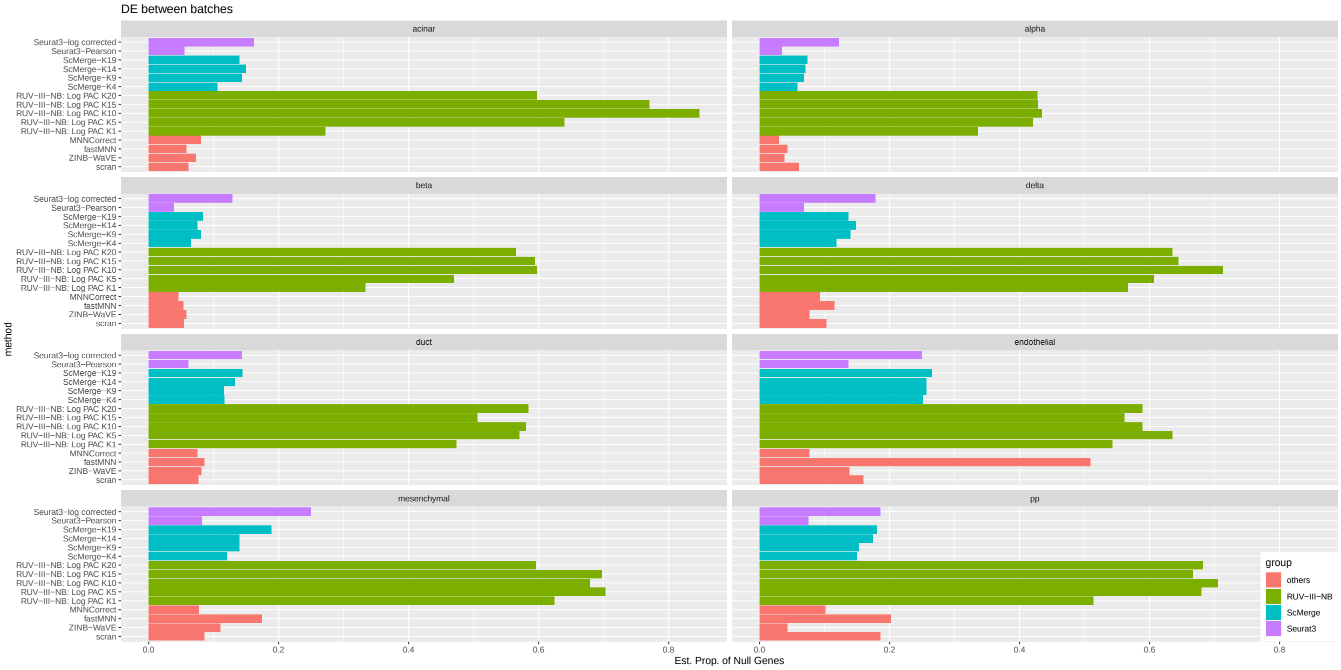
Supp. Fig. 7: Top 5 KEGG Pathways among DEG calculated from normalized data from the Pancreatic study: larger vs smaller Beta cells



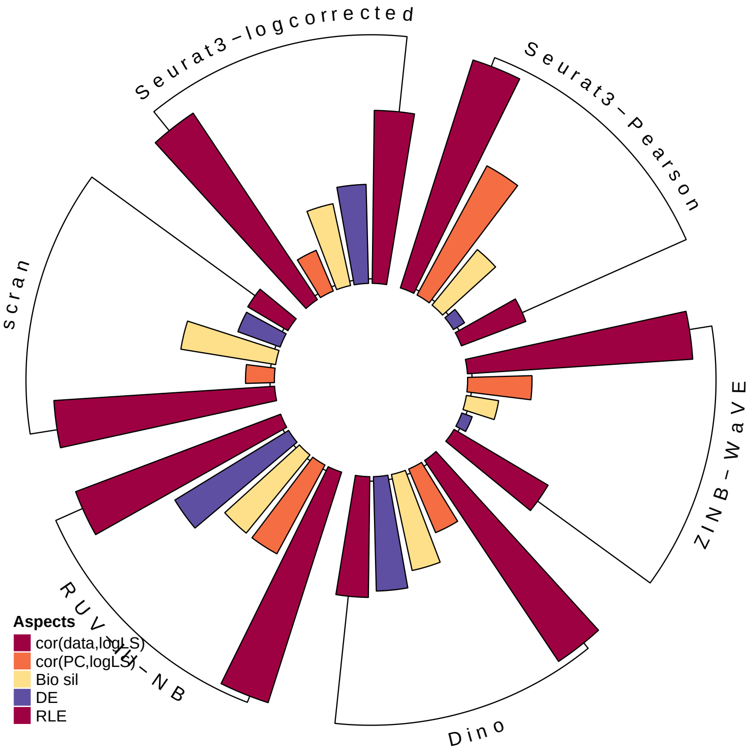
Supp. Fig. 8: Celline study. Estimated proportion of non-DE (i.e. null) genes when comparing the same cell type across batches. When batch effects are completely removed, the proportion of null genes is 1.



Supp. Fig. 9: CLL study. Estimated proportion of non-DE (i.e. null) genes when comparing the same cell type across batches. When batch effects are completely removed, the proportion of null genes is 1.

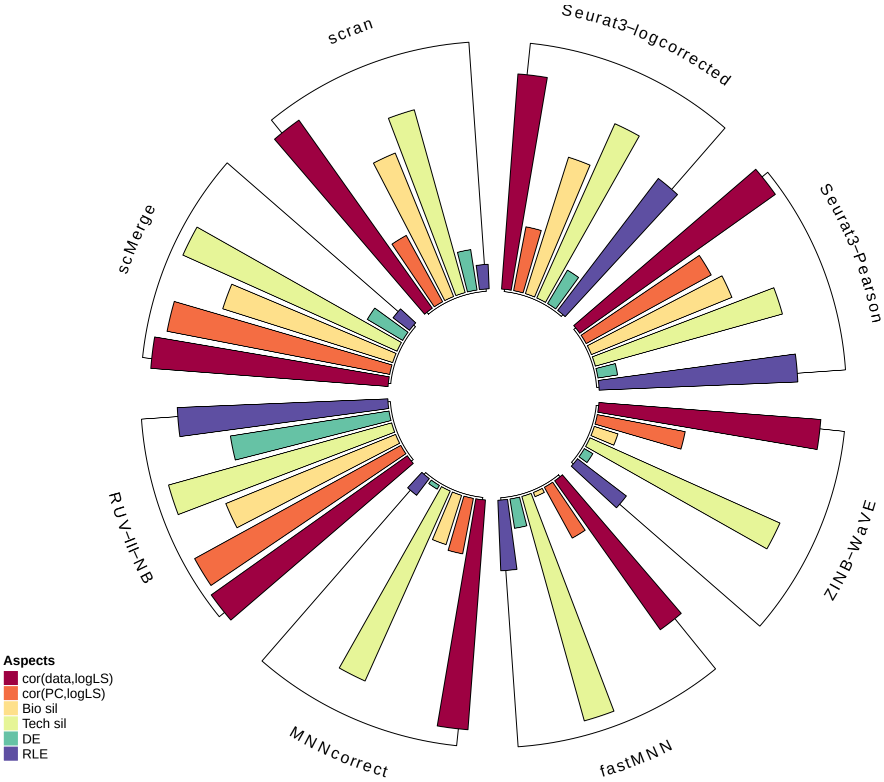


Supp. Fig. 10: Pancreatic study. Estimated proportion of non-DE (i.e. null) genes when comparing the same cell type across batches. When batch effects are completely removed, the proportion of null genes is 1.



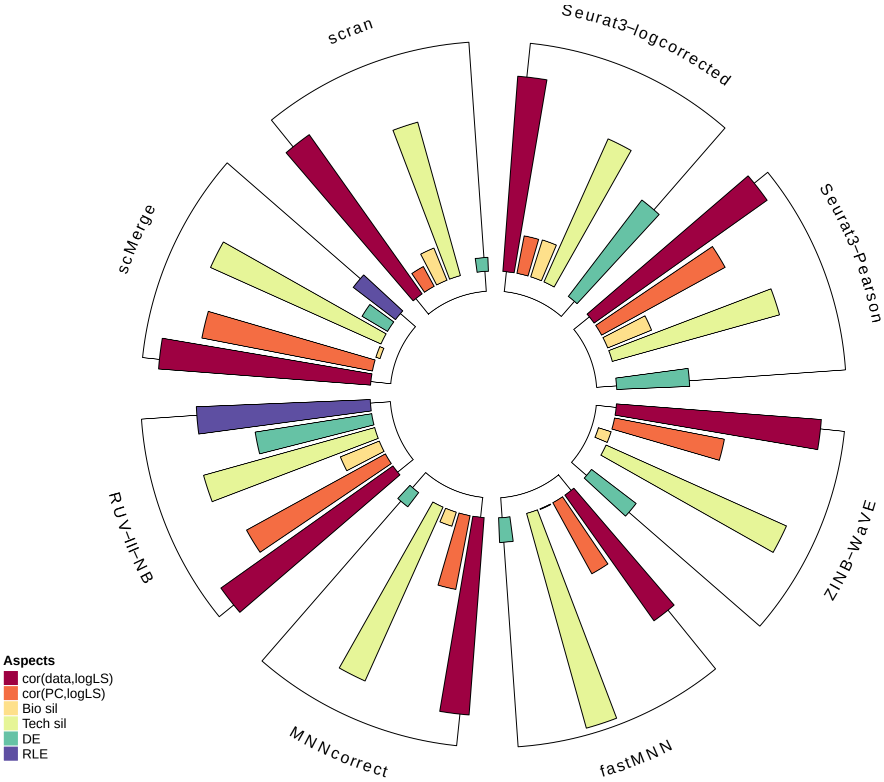
Supp. Fig. 11: Overall performance in the NSCLC study. Sectors correspond to methods and length of a segment inside each sector corresponds to level of performance with respect to a specific aspect. For assessment metrics where lower indicates better performance such as batch (technical) silhouette and correlation between RLE characteristics and unwanted factors, the length of the segment is calculated as 1-metric. All metrics, except for biological silhouette, are calculated as an average of within cell-type statistics

.



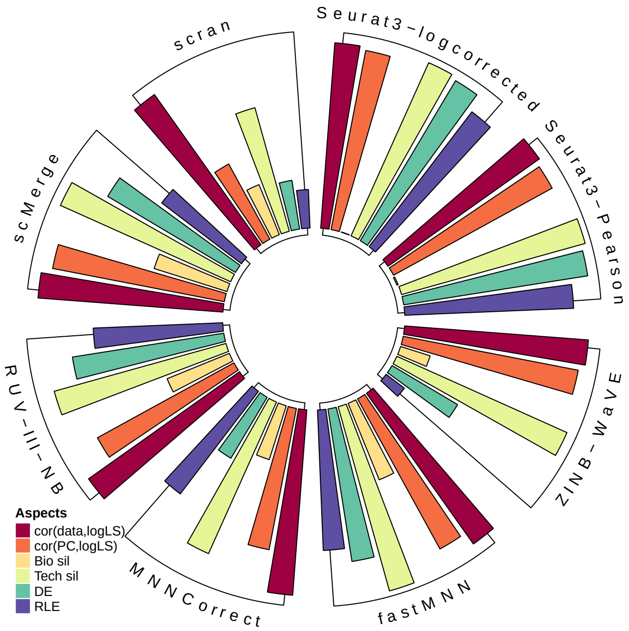
Supp. Fig. 12: Overall performance in the Cell line study. Sectors correspond to methods and length of a segment inside each sector corresponds to level of performance with respect to a specific aspect. For assessment metrics where lower indicates better performance such as batch (technical) silhouette and correlation between RLE characteristics and unwanted factors, the length of the segment is calculated as 1-metric. All metrics, except for biological silhouette, are calculated as an average of within cell-type statistics

.



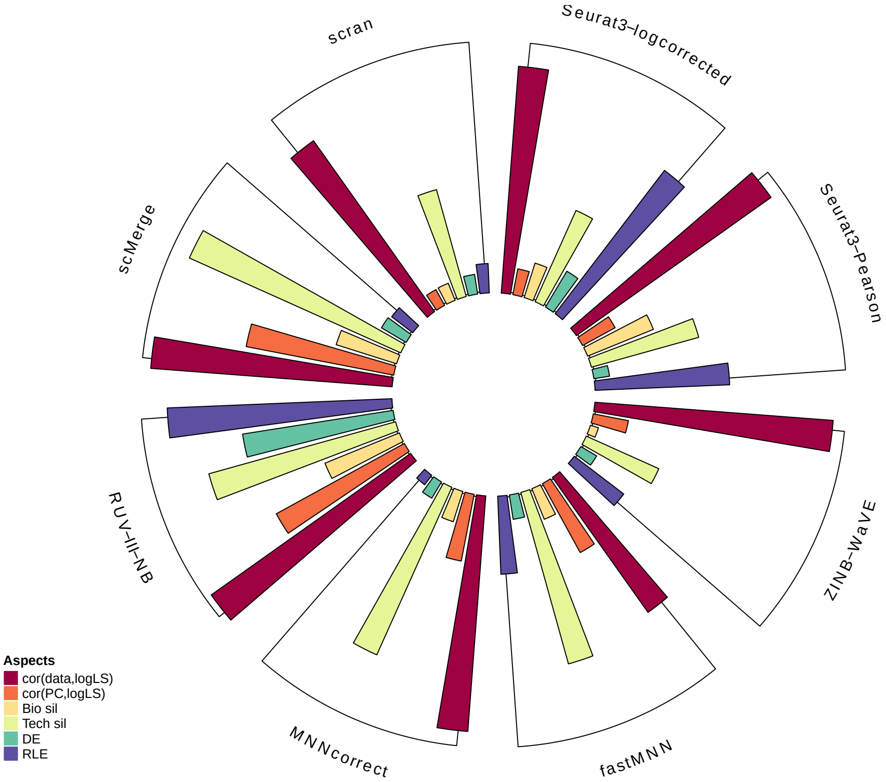
Supp. Fig. 13: Overall performance in the CLL study. Sectors correspond to methods and length of a segment inside each sector corresponds to level of performance with respect to a specific aspect. For assessment metrics where lower indicates better performance such as batch (technical) silhouette and correlation between RLE characteristics and unwanted factors, the length of the segment is calculated as 1-metric. All metrics, except for biological silhouette, are calculated as an average of within cell-type statistics

.



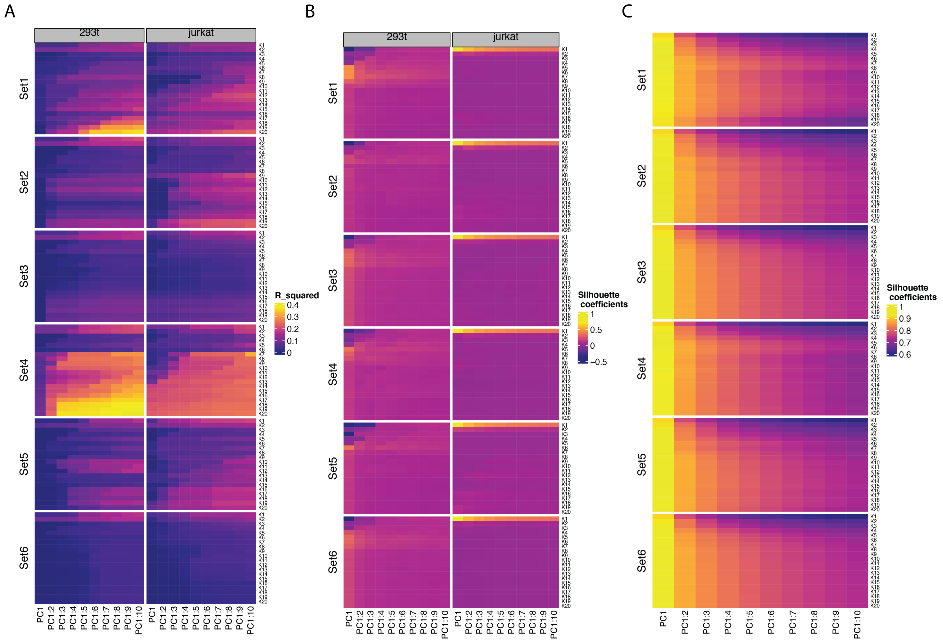
Supp. Fig. 14: Overall performance in the Gaublomme study. Sectors correspond to methods and length of a segment inside each sector corresponds to level of performance with respect to a specific aspect. For assessment metrics where lower indicates better performance such as batch (technical) silhouette and correlation between RLE characteristics and unwanted factors, the length of the segment is calculated as 1-metric. All metrics, except for biological silhouette, are calculated as an average of within cell-type statistics

.



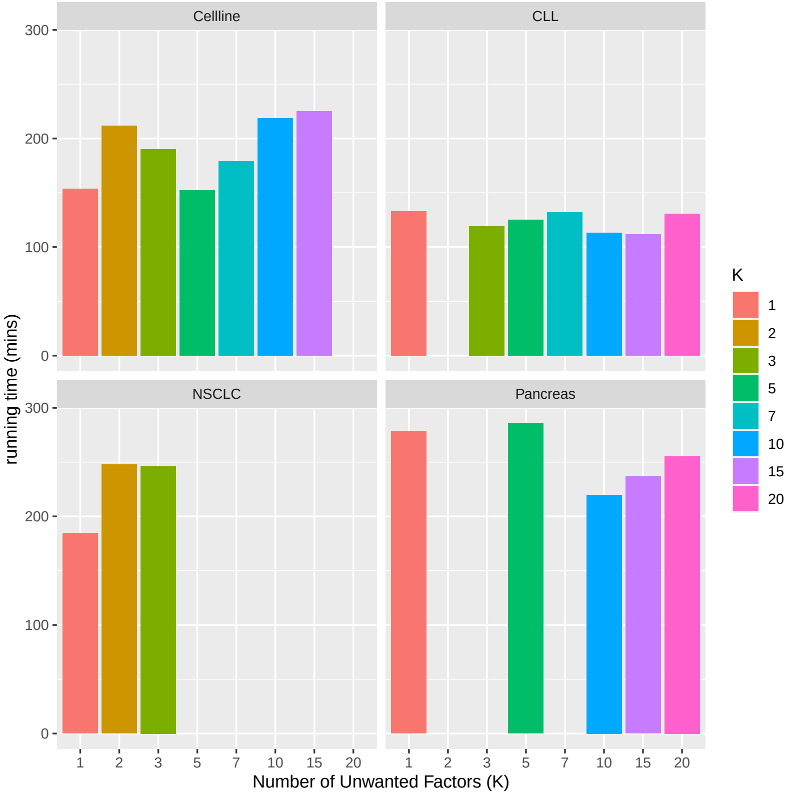
Supp. Fig. 15: Overall performance in the Pancreatic study. Sectors correspond to methods and length of a segment inside each sector corresponds to level of performance with respect to a specific aspect. For assessment metrics where lower indicates better performance such as batch (technical) silhouette and correlation between RLE characteristics and unwanted factors, the length of the segment is calculated as 1-metric. All metrics, except for biological silhouette, are calculated as an average of within cell-type statistics

.



Supp. Fig. 16: Sensitivity of RUV-III-NB performance metrics across different negative control sets and numbers of unwanted factors (K) using the Cell line dataset. (A) R^2 between log library and cumulative principal components (PC). (B) Average Batch Silhouette width. (C) Average Biological Silhouette width. Set1: all genes are negative controls. Set2: scRNA-seq housekeeping genes from the scMerge package. Set3 : use scran::modelGeneVar function to select 1000 genes with the highest technical variance in batch with two cell lines (batch 3). Set4: DE analysis between cell lines in batch 3, followed by selecting 1000 genes the highest p-values. Set5: use scMerge::scSEGIndex function to select the 1000 genes with the most stable expression from batch 3. Set6: use genes with biological absolute log fold-change ≤ 0.05 (biological log fold-change is from DE analysis between the cell lines in batch 3) and technical absolute log fold-change ≥ 2(technical log fold-change is from DE analysis between cell lines in batch 1 and 2)

.



Supp. Fig. 17: RUV-III-NB running time for various datasets on HPC environment with 15 cores and 120 Gb total RAM (8Gb RAM per core)