corral: Single-cell RNA-seq dimension reduction, batch integration, and visualization with correspondence analysis

Lauren L. Hsu[1,2], Aedín C. Culhane[1,2,3,*]

[1] Department of Biostatistics, Harvard TH Chan School of Public Health, Boston MA

[2] Department of Data Science, Dana-Farber Cancer Institute, Boston MA

[3] School of Medicine, University of Limerick, Ireland

* Corresponding author

Keywords: dimension reduction, single cell, scRNAseq, batch integration, gene expression, principal component analysis, correspondence analysis, matrix factorization, biplot

Abstract

Effective dimension reduction is an essential step in analysis of single cell RNA-seq (scRNAseq) count data, which are high-dimensional, sparse, and noisy. Principal component analysis (PCA) is widely used in analytical pipelines, and since PCA requires continuous data, it is often coupled with log-transformation in scRNAseq applications. However, log-transformation of scRNAseq counts distorts the data, and can obscure meaningful variation. We describe correspondence analysis (CA) for dimension reduction of scRNAseq data, which is a performant alternative to PCA. Designed for use with counts, CA is based on decomposition of a chi-squared residual matrix and does not require log-transformation of scRNAseq counts. We extend beyond standard CA (decomposition of Pearson residuals computed on the contingency table) and propose variations of CA, including an alternative chi-squared statistic, that address overdispersion and high sparsity in scRNAseq data. The performance of five variations of CA and standard CA are benchmarked on 10 datasets and compared to glmPCA. CA variations are fast, scalable, and outperform standard CA and glmPCA, to compute embeddings with more performant or comparable clustering accuracy in 8 out of 9 datasets. Of the variations we considered, we found that CA using the Freeman-Tukey chi-squared residual was most performant overall in scRNAseq data. Our analyses also showed that variance stabilizing transformations applied in conjunction with standard CA (using Pearson residuals) and the use of "power deflation" smoothing both improve performance in downstream clustering tasks, as compared to standard CA alone. CA has advantages including visual illustration of associations between genes and cell populations in a "CA biplot," and easy extension to multi-table analysis enabling integrative dimension reduction. We introduce *corralm*, a CA-based method for multi-table batch integration of scRNAseq data in shared latent space, and we propose a new approach for assessing batch integration. We implement CA for scRNAseq in the *corral* R/Bioconductor package (https://www.bioconductor.org/packages/corral) interfaces directly with widely used single cell classes in Bioconductor, allowing for easy integration into scRNAseq pipelines.

Introduction

Single cell mRNA sequencing (scRNAseq) simultaneously measures the transcript levels of genes in thousands of individual cells, providing a window into the transcriptional and functional diversity of cells in a tissue or experiment. Because of the size and complexity of the data, the datasets are orders of magnitude greater than those encountered when analyzing "bulk" RNAseq data from tissue samples. While such fine resolution data have the potential to reveal new biological findings, scRNAseq data exhibit sparsity, noisiness, and technical artefacts beyond those seen for bulk RNA samples (1,2), necessitating scRNAseq specific pre-processing and normalization (3,4). Typically scRNAseq analysis includes the use of dimension reduction, as it attenuates noise and ensures computational tractability, but the choice of method considerably influences downstream analyses, results, and conclusions (3,5).

Selecting an appropriate dimension reduction method is important; an effective method finds a representation of the data that minimizes noise and redundancy and uncovers meaningful signals that reveal latent structures and patterns within the data (6,7). When defined from scRNAseq data, reduced dimension embedding representations are most useful when they preserve meaningful, biologically relevant variation; are robust, meaning that the decomposition of new but similar observations

consistently yields a similar embedding space; and generalize and transfer to new data, enabling new observations arising from similar biological processes to be projected into the same latent space.

Single cell RNAseq counts are generally modeled as multinomially distributed, and are often approximated as negative binomial or Poisson counts (2), reflecting the fact that the data are neither continuous nor approximately Gaussian. As such, use of principal component analysis (PCA) requires that discrete and sparse scRNAseq count data be transformed prior to dimension reduction with this method (6). PCA is a linear dimension reduction method that obtains a low-dimensional data representation along orthogonal linear axes such that the proportion of variance accounted on each axis is maximized in Euclidean space (4,8–11). Because PCA is most suitable for continuous data that is approximately normally distributed, it may exhibit artefacts when applied to data with gradients or non-continuous data (such as counts); one such artefact, called the "arch" or "horseshoe" effect, has been found to occur when PCA is applied to scRNAseq data without log-transformation (4,6,12). So, in practice, and despite known issues with applying log-transformation to scRNAseq count data (2,13,14), most single cell workflows begin with a log(x+1) transformation of the counts matrix, and then use PCA to decompose the resulting "logcounts" data (3). The use of logcounts has poor theoretical justification, and in some cases may obscure meaningful variation (2,14), but the resulting reduced dimension embeddings of the data from PCA are nonetheless used in scRNAseq clustering, trajectory analysis, and cell type classification (3). Although other dimension reduction approaches tailored for scRNAseq have been proposed (2,15–17), PCA remains the most widely used method largely due to its simplicity, speed, and computational efficiency; in a comparison of 18 dimension reduction methods, PCA ranked highly when accuracy and performance in downstream analysis were considered with computational scalability (18).

Classical matrix factorization methods, including PCA, can be considered instances of the general duality diagram approach proposed by Benzécri and the French school of multivariate statistics in the 1970's (8,19–23) which pivots focus from the matrix as columns of fixed variables to the matrix as an operator between inner product spaces, unifying classical multivariate methods like PCA with modern kernel methods into the same framework (8,21). Another matrix factorization method that emerged in the duality diagram framework is correspondence analysis (CA), a fast dimension reduction method appropriate for non-negative, count-based data and can identify relationships between categorical data types that is popular among ecologists for analyzing species-by-site abundance count matrices (8,24). In practice, PCA is often computed by singular value decomposition (SVD) of column centered or Z-score normalized data (Figure 1A) (4,25) and CA is computed by SVD of the Pearson residuals to reveal the row-column associations that deviate from expectation (26). The principal components in CA partition the co-dependence between the rows and columns such that a higher weight indicates a stronger dependence or association between row and column; for scRNAseq data, CA principal components can identify co-dependence between gene expression counts and particular cells. From this perspective, the main difference is the space into which the data are transformed then decomposed. Whereas PCA operates in Euclidean data space to partition the variance, CA partitions the total contingency chi-square table along linear additive components (27). CA has a long tradition in diverse settings and disciplines, including linguistics, business and marketing research, and archaeology (26,28), where it is applied to large, sparse count data. CA has also been applied in bioinformatics to perform codon usage analysis (29,30); to analyze microarray transcriptomics data (31); to integrate GO labels with microarray data (32); and to analyze metagenomic and microbiome data (33). In the *made4* R/Bioconductor package, Culhane et al.

implemented CA for microarray and bulk RNA-seq data (34–36). We now discuss its application to scRNAseq analysis.

Focusing on the issues of log-transforming scRNAseq counts when applying PCA, Townes et al. (2019), Hafemeister & Satija (2019), and Lause et al. (2021) have presented various approaches to scRNAseq analysis based on Pearson residual normalization as an alternative to distortive log-transformation. Townes et al. (2019) proposed glmPCA, a generalization of PCA that minimizes deviance rather than mean squared error (MSE) and accommodates non-canonical link functions, and that can be approximated with PCA of Pearson or deviance residuals (2). Lause et al. proposed analytic Pearson residual normalization (14), extending work from Hafemeister & Satija, who used a regression-based approach to computing Pearson residuals (13). Lause et al. cited our open-source Bioconductor workshops which describe CA; the relationships among CA, PCA, and SVD; and their application in scRNAseq data as support that glmPCA from Townes et al., (2019), scTransform from Hafemeister & Satija (2019) and their approach are CA or closely approximate CA (14,37). However, CA, which can be computed by SVD on the standardized Pearson residuals, may not be the most appropriate approach when there is overdispersion in the contingency table (41, 42).

CA has a rich history and several variations and extensions of CA have been developed for different research contexts across a wide range of fields. We benchmark performance of standard CA with Pearson residuals and three additional CA approaches (with five specific variations) that address the issue of overdispersion, including substitution with an alternative chi-squared statistic that is better adapted to counts data exhibiting high sparsity and overdispersion. We compare the five variations of CA that address data overdispersion with standard CA and glmPCA (2) in scRNAseq analysis. Because cell clustering and characterization is a key part of most scRNAseq workflows, we set as the goal of the benchmarking task to find embedding representations that facilitate identifying and annotating complex populations of cells. We show that the CA biplot allows a geometric interpretation of features and objects in the same space, which in turn facilitates efficient exploratory data analysis and cluster interpretation. We implemented CA for scRNAseq in *corral*, an R/Bioconductor package that interfaces directly with Bioconductor classes (such as *SingleCellExperiment*). The *corral* package is designed for computational scalability and is fast and performant compared to PCA and other dimension reduction methods, including glmPCA. Because of its design and implementation, switching from PCA to CA with *corral* is achieved through a simple pipeline substitution, and improves dimension reduction of scRNAseq datasets.

Results & discussion

Correspondence analysis: Count-based dimension reduction

Correspondence analysis (CA) casts scRNAseq read counts in a contingency table analysis framework and in its canonical form can be conceptualized as a two-step procedure (graphically outlined in Figure 1A; detailed in **Methods**). The count matrix is transformed to a matrix of Pearson chi-squared residuals, which is then factorized with singular value decomposition (SVD).

CA analysis of scRNAseq does not require, but is compatible with, log-transformed read counts (logcounts). PCA, which has been widely used, requires data transformation, and is therefore generally

applied to logcounts data, despite the fact that log-transforming scRNAseq counts has been shown to distort latent space representation such that the first dimension is driven by individual cell sparsity, or the number of features with zero observed counts ("zero fraction") (2). Since we propose CA as a more suitable alternative to PCA for finding cell embeddings, we compared CA to the widely used correlation-based PCA (4).

We applied both CA and PCA to a synthetic scRNAseq benchmarking data set (on both counts and logcounts) obtained by sequencing pseudo-cell mixtures of mRNA from eight distinct groups (38). Plots of the first two principal components for both PCA and CA are shown in Figure 1B. Although the first PCA component separated cells into groups to an extent, the second component merely found a gradient within the groups. In contrast, CA clearly clustered and separated all groups within two components. Similarly, results in purified PBMCs (Zhengmix4eq benchmarking dataset) demonstrated that CA can be applied directly to counts or to logcounts and still achieve good clustering and separation, whereas PCA on counts produces an "arch" or "horseshoe effect," which arises from the presence of a latent sequential ordering or gradient in the data (12,25). PCA on logcounts performed similarly to CA on either counts or logcounts.

We quantitatively compared the performance of the four pipeline configurations presented in Figure 1C (CA and PCA on both counts and logcounts) on six reference benchmark datasets—the three scRNAseq datasets from the SCMixology study that varied the technology for sequencing a known mixture of three cell lines (38) and the three Zhengmix PBMC datasets (39,40). (All benchmark datasets are listed in the Benchmarking section of **Methods**.) Cluster recovery based on the annotated cell types in the study was assessed using Adjusted Rand Index (ARI), which assesses similarity between two sets of data partitions (Figure 1D). In all comparisons, CA outperforms or matches PCA's performance (highest ARI per datasets marked with orange circle). Consistent with Figure 1C, CA is robust when applied to either counts or logcounts data, obviating the need for log-transformation and therefore avoiding its associated issues.

Overdispersion in CA

One well-known aspect of CA is that it can be influenced by "rare objects" or outliers (41). Due to high underlying heterogeneity of gene expression within and between various cell types, scRNAseq data often include biologically "real" outliers as opposed to artefacts due to noisy data. For example, professional secretory cells have a distinct biological profile often driven by extraordinarily high production of one or two proteins, such as insulin production by pancreatic islet cells or immunoglobin production by immune cells. Similarly, senescent or quiescent cells may have a different gene expression profile compared to rapidly dividing cells in culture or high-grade tumor cells. We investigated adjustments to CA that address overdispersion and sensitivity to rare objects with low overall counts. Six CA methods (standard CA and five adaptations of CA to address overdispersion) were applied to nine datasets, including the three Zhengmix human PBMC benchmarking datasets, as well as cells from human pancreas, human brain, and *Xenopus* tail (detailed in Table 1). Clustering performance on cell embedding representations generated from each was compared and benchmarked in reference to glmPCA (2) (see **Methods** – Benchmarking for details).

The five adaptations of CA broadly fall into three general approaches (Figure 2A). The first class of approaches we considered was to explicitly apply a variance stabilizing transformation to the count matrix prior to computing Pearson residuals. Lause et al. (2021) discussed variance stabilizing transformation as compared to Pearson residual normalization, though in their study did not combine variance stabilization and Pearson residual normalization prior to matrix decomposition. They reported that the degree of correction from variance stabilizing transformation alone was insufficient for scRNAseq data in their pipeline configuration and found that only normalizing with analytic Pearson residuals was more effective than only applying variance stabilization (14). Given that scRNA-seq counts are often approximated as Poisson-distributed, we considered three variance stabilizing transformations that are typically applied to count data. These three square-root based transformations all stem from the original observation by R.A. Fisher that performing an arccosine transformation on the square root of multinomial probabilities yields approximately normally distributed angles on a hypersphere (42). The first was square root transformation of count data (Row 3 of Figure 2A), which has been used to correct overdispersion in Poisson counts (43). The second is Anscombe's variance stabilizing count transformation (Row 4 of Figure 2A), originally proposed in 1948 for use with Poisson, binomial, and negative binomial data (44). Third, we used the Freeman-Tukey's variance stabilizing count transformation (Row 5 of Figure 2A), originally proposed in 1950, also for Poisson and other count data (45).

Our results indicate that when variance stabilization is combined with classical CA, performance is superior to CA alone. Variance stabilization of counts prior to computing Pearson residuals provided great gains in downstream clustering with increases in ARI of 0.4 in two studies (Zhengmix4uneq, Aztekin *Xenopus* tail); square-root transformation prior to CA increases the ARI in 7 datasets, while transformation to Anscombe counts or Freeman Tukey counts increased the ARI clustering performance in every dataset when compared to standard CA (with no variance stabilization of counts prior to computing Pearson residuals). Indeed, Anscombe's variance stabilizing count transformation achieves the highest observed ARI in 1 of 9 test datasets (pancreas: Lawlor) and Freeman-Tukey's variance stabilizing count transformation had best overall performance in 3 of 9 datasets (Zhengmix4uneq; pancreas: Muraro, Lawlor). Although the square root count transformation did not outperform the other two transformations in any of the comparisons, its Adjusted Rand Index (ARI) was within 0.05 of other two transformation in 7 of 9 datasets. Furthermore, in the pancreas datasets, variance stabilizing count transformation coupled with classical CA yielded the highest ARI of tested methods, outperforming glmPCA.

The second variation we considered is the use of "power deflation" as a data smoothing method. Power deflation handles extreme outliers in the chi-squared residual matrix by raising all values to a power, α, prior to performing SVD, while preserving sign (Bottom row of Figure 2A). Conceptually, this transformation is similar to the Tukey ladder transformation (46), and has a smoothing effect on the matrix of chi-squared distances, reducing the impact of outlying values while preserving the ordering of values. The range of α is technically between 0 to 1, where a value of 0.5 is a square root transformation. To achieve a "soft" smoothing effect, we considered $\alpha \in [0.9,\ 0.98]$ (data not shown) and present results for $\alpha = 0.9$ in Figure 2. We found that in all nine datasets, this power deflation smoothing approach performed comparably to, or better than, standard CA, although its impact on CA performance was less than variance stabilizing count transformation.

Finally, we considered alternative chi-squared statistics that are better-suited to count data with high levels of sparsity and overdispersion. CA of the Freeman-Tukey residuals (CA-FT) has been applied to

archaeological site data, where it exhibited a variance-stabilizing effect and outperformed classical CA (SVD of the Pearson residuals), in the analysis of sparse, over-dispersed artifact data (counts of archeological artifacts by site) (45,47,48). Both Pearson residuals and Freeman-Tukey residuals are members of the Cressie-Read family of power divergence statistics for testing goodness-of-fit in multinomially-distributed count data, and when squared, both residuals are chi-square distributed random variables (47,49). We found that CA-FT is well-suited for scRNAseq counts (Row 2 of Figure 2A) as it outperformed CA of the Pearson residuals in all nine datasets and its performance was comparable to (ARI within 0.02) or superior to glmPCA in 8 out of 9 benchmarking datasets. In most datasets CA-FT also had higher or comparable clustering accuracy (ARI) to Pearson residual CA with variance stabilizing transformation. CA-FT achieved the highest ARI overall in 6 out of 9 datasets.

Unlike standard CA (Pearson residuals), we observed little benefit to combining CA-FT with variance stabilizing transformation (square root, Anscombe, or Freeman-Tukey) (Figure S1). Variance stabilizing transformation prior to CA-FT actually reduced its performance in 5 out of 9 datasets, and in the Muraro pancreas dataset, decreased the ARI from 0.75 (CA-FT) to less than 0.65 (CA-FT with square root transformation). CA-FT with square root transformation only improved downstream clustering of the Darmanis brain dataset from ARI of 0.65 (CA-FT) to 0.73 (CA-FT with square root transformation). In all datasets except for the Darmanis brain, power CA-FT with deflation data smoothing had greater performance that CA-FT with variance stabilizing transformations, however, overall, it provided no or only marginal increases in performance of CA-FT. Results show that while the performance of standard CA of the Pearson residuals improves dramatically with variance stabilizing transformation, CA-FT adjusts for and is appropriate to be used with overdispersed data.

In dimension reduction of scRNAseq, component selection can greatly influence downstream cell clustering analysis. Therefore, we considered clustering performance as a function of the number of components selected (Figures 2C, S2). The ability to recover “known” clusters (measured with ARI between clustering output and the previously identified and published cell types) was higher for the simpler mixtures of known, purified cell types (i.e., Zhengmix datasets) and lower for the complex tissues (i.e., Brain, Pancreas, and *Xenopus* tail). We interpret this to mean that the higher order components were capturing variation helpful for separating the larger number of less well-defined cell types. In comparing performance, we found an association between the number of components and the complexity of the clustering task; the inclusion of more components generally increases the ARI in more complex tissue. In contrast, we saw there was little additional gain and for some a reduction in ARI when the number of components was increased for the Zhengmix datasets. The Zhengmix datasets comprise combinations of distinct PBMC cell types that were sorted and purified prior to sequencing. In simple datasets, including additional components beyond those that sufficiently capture the biological variance may add stochastic, technical, or systematic noise in the system. Benchmarking of methods using the maximum ARI and the ranking of methods was robust to the number of components; CA-FT was consistently the most performant approach whether the first 30 (Figure S2) or first 50 (Figure 2B) components were included in downstream clustering.

CA, CA-FT, and its variations generate a deterministic result that is stably reproduced. In contrast, glmPCA is not deterministic, and therefore results may vary when the method is rerun on the same dataset (Figures 2C, S3). For reproducibility, we tested ten random seed initiations of glmPCA (colored in gray; Figure 2C), which revealed that glmPCA results are consistent for simpler datasets but in other datasets,

such as the *Xenopus* tail dataset, its performance varies dramatically between iterations. In the Lawlor pancreas dataset, one of the tested seeds failed, suggesting that results were somewhat dependent on finding a “lucky seed”. In simpler datasets, such Zhengmix, all methods generated high ARI scores and glmPCA results had consistency between individual runs (Figure 2C). However, there was greater variance in glmPCA performance with increasing data complexity. For each dataset, we present the average of the maximum ARI achieved in each of 10 runs of glmPCA.

CA variations that are adapted for overdispersion are more performant in downstream clustering than standard CA or glmPCA (Figure 2B). Of the approaches we considered, CA-FT was most performant, and outperformed standard CA with the variance stabilization transformation or related power deflation approach.

Geometric interpretation of cell and feature embeddings

The CA biplot provides a natural framework for cluster interpretation, highlighting biologically meaningful relationships among gene expression patterns and cell populations, and may be extended to guide feature selection. Every transformed count (residual) in a CA matrix has an intuitive interpretation, as it is the chi square test statistic for strength of association between a particular row (expression of a gene) and column (cell). This means that the CA matrix captures the strongest associations between gene expression and cell type, highlighting functional contrasts by individual cells and by subpopulations of cells. Biplots visualize associations between features and objects, or in this case, genes and cells. Rather than examining the feature and object embeddings individually, the biplot places both sets of embeddings on the same axes, revealing not only the associations that may exist among either rows or columns separately, but also between particular rows and columns (6,50). The distance from the origin indicates the magnitude of association; the angular rotation distance (cosine similarity) reflects similarity of the objects (or features) to each other, or association between objects and features.

To illustrate the power of the CA biplot we performed a standard correspondence analysis of the Zhengmix8 PBMC benchmarking dataset, then plotted the first two dimensions of the resulting cell and gene embeddings (Figure 3). The 20 genes with highest weight by L2 norm (Euclidean distance from the origin) in the first two dimensions are colored blue, with a corresponding gene label. Cell populations are colored by cell type. The biplot highlights genes that have strong associations with and may discriminate between particular cell populations. For example, natural killer (NK) cells constitutively express granulysin, encoded by the gene GNLY, and although they are not the only cellular producers of granulysin, GNLY expression in other cells such as cytotoxic T-cell populations is driven by immune activation (51). In the CA biplot, we see that GNLY has a high weight in PC2 (indicated by distance from origin) and has a similar angular rotation as the NK cell population (high cosine similarity). Correspondingly, the inset ridge plots in Figure 3 showing histograms of log expression in each cell population confirm it is highly expressed specifically in the NK cell population.

Calcium-binding proteins S100A8 and S100A9, also known as MRP8 and MRP14 respectively, are members of the S100 family that are constitutively expressed in monocytes and neutrophils (52,53). Correspondingly, in the CA biplot in Figure 3, the expression of both genes is strongly associated with the monocyte population (same direction, large magnitude), which is consistent with the distribution of S100A8 expression among cell populations, illustrated in the inset plot. Similarly, LYZ encodes for lysozyme, a molecule highly secreted by monocytes (54). As expected, reflecting the elevated differential

expression of the gene among the monocyte population shown in the inset, the gene's location in low-dimensional embedding space relative to the cell population is close in angle, while also being far from the origin.

The biplot in Figure 3 also informs about genes that are highly and differentially expressed in multiple cell populations: TYROBP encodes for a signaling adaptor protein, also called KARAP/DAP12, which was initially identified as a wiring component in NK anti-viral and anti-tumoral function. TREM-1, a KARAP/DAP12-associated surface protein, amplifies monocyte, macrophage, and granulocyte activation by cytokines and chemokines following LPS stimulation (55). While other lymphoid and myeloid cells may express TYROBP, it has predominantly been observed in NK, monocytes/macrophages, and dendritic cells, which is consistent with the enriched expression levels in the expected cell types: NK and monocytes. The gene is projected between these cell population and expression ridge plots confirm that it exhibits elevated expression specifically in NK and monocyte cell populations.

CD74 is part of the MHC class II complex which is consistent with both its biplot positioning and expression plot: angularly, it lies closest to the B cell population, but is also rotated slightly towards the monocyte population (56). Correspondingly, although expression of CD74 at low levels is seen in cells of all types, it is most elevated in B cells and with increased expression in some monocytes. Similarly, the HLA-DRA gene encodes the alpha chain of the HLA-DR protein, which is a cell-surface receptor in the MHC class II complex (57). As both B cells and monocytes are professional antigen presenting cells that require all the machinery of the MHC class II complex, these genes are important for function of both cell types, and the positions of both genes in the biplot are angled between the most relevant cell types, providing a biologically meaningful summary of associations between genes and cell sub-populations.

The CA biplot facilitates unified analysis of cell and gene embeddings, which can inform cluster interpretation, in addition to serving as a basis for integrating with and extending other methods, such as gene set enrichment analysis and projection of supplementary data into a shared latent space.

Batch integration with corralm

The need to integrate multiple datasets is an important motivator for the development and continued refinement of CA (10,35,58). Our multi-table adaptation of CA, implemented as *corralm*, operates on the indexed residuals and includes scaling for library size and can align cells from different datasets. We compared *corralm*'s performance with widely used batch integration methods (Figure 4), including Harmony and Seurat 3, both of which performed well in a recent benchmarking study examining 14 batch-effect correction methods (59). Each method was applied to batch integration of the SCMixology dataset, which is a benchmarking dataset comprising scRNAseq profiles from a mixture of three cell lines (H2228, H1975, and HCC827), obtained in three batches using different library preparation platforms (Dropseq, Celseq2, and 10X) (38). The objective is to find a low-dimensional representation of the data that aligns the data across batches while preserving distinct cell line clusters.

The plots of the first two principal components of the reduced dimension representation show that *corralm*, Harmony adapted with *corralm* embeddings, and SCTransform successfully integrate batches while preserving cell line clusters (Figure 4A, rows 1, 3, 4). In contrast, the published version of Harmony (which uses PCA embeddings) fails at both data integration and cluster detection on these same data (Figure 4A, row 2). Qualitatively, SCTransform appears to exhibit the best alignment by batch and tightest

clusters by cell-type, but its run-time is an order of magnitude slower than *corralm* and Harmony with *corralm*. SCTransform runs in approximately 1 minute 25 seconds, while *corralm* and Harmony with *corralm* run in approximately 7 seconds for the equivalent task, allocated a single core of a 2019 MacBook Pro laptop (with 2.3 GHz 8-core Intel Core i9; 64 GB memory). However, it should be noted that the SCMixology dataset is relatively small (1,401 cells), but at scale, this difference in run-time would significantly impact the overall speed of a pipeline, thus demonstrating an advantage of *corralm* and Harmony with *corralm*.

Cluster evaluation measures such as ARI detect whether clusters can be re-identified, but do not directly quantify how well datasets are integrated in their low dimensional embedding representations. We propose a new batch integration metric, scaled variance (SV; *scaled_var* in the *corral* package), for quantitatively assessing batch integration when integrating datasets with similar cell populations and distributions across batches, presented in Figure 4B and described in detail in **Methods**. For each dimension of each embedding, we compute the variance of the subset of observations from each batch and scale by the overall variance in that dimension as a measure of under- or over-dispersion of the subset's embeddings in that dimension. For example, in the SCMixology benchmarking dataset, biologically identical samples (mixture of three cell lines) were assayed using three library preparation methods (Dropseq, Celseq2, and 10X), with each batch expected to have the same distribution of cells. SV values closer to one indicate better integration (more similarity in dispersion) in a given dimension by batch. Consistent with the plots in Figure 4A, the SV plots (Figure 4B) showed that SCTransform had the best integration, with all SV points very close to one. Similarly, *corralm* and Harmony with *corralm* also showed good batch integration, and both outperform Harmony with PCA, which had SV values far from one.

Computational performance of corral's CA implementation

The *corral* implementation of CA leverages the fast, approximate, partial SVD method from the *irlba* package in R (60); even when allocated just a single core on a 2019 MacBook Pro laptop (with 2.3 GHz 8-core Intel Core i9; 64 GB memory), *corral* runs in under a minute for a dataset of 1,500 features and over 20,000 cells (top 50 components). Figure 5A shows that for the analogous task, glmPCA takes a little over an hour, and that across a range of dataset sizes (1,500 features), the glmPCA run-time increases rapidly with the number of cells, while CA (*corral*) scales much more favorably (Figure 5). As faster and more efficient implementations of SVD are developed, run-time and/or memory use may be further reduced by modularly incorporating these approaches into the *corral* pipeline. Classic CA and the variations we considered are not sparse implementations, and computational performance may be further enhanced with adaptations for sparsity. Since CA has similar computational requirements to PCA, it is a simple pipeline substitution to replace PCA with CA in a workflow.

Conclusion

Correspondence analysis (CA) is a statistical technique with a rich theoretical foundation that was first proposed and mathematically characterized nearly a century ago (61) and which has continued to be developed and extended. CA has been periodically "rediscovered" in a variety of disciplines (20,28,62–64) and most recently in the field of scRNAseq analysis: several groups have suggested Pearson residual-based normalization prior to matrix decomposition with PCA (2,13,14), a process that is conceptually similar to standard CA—apart from differences in how residuals are computed, one additional distinction

is in PCA's additional Z-score normalization step (4) after computing Pearson residuals, as opposed to directly decomposing the residual matrix with SVD.

We compared the performance of CA and five CA variations of CA that address overdispersion in scRNAseq data and benchmarked these against glmPCA (2), which is a popular method in the field. Of the variations we considered, we found that CA of the Freeman-Tukey chi-squared residuals (CA-FT) was most performant overall in scRNAseq data. Our analyses also showed that variance stabilizing transformations applied in conjunction standard CA (with Pearson residuals) and the use of "power deflation" smoothing both improve performance in downstream clustering tasks, as compared to standard CA alone. Therefore, for dimension reduction of scRNAseq data, we recommend using CA-FT or, incorporating variance-stabilization, smoothing, or both if necessary when using standard CA (of the Pearson residuals).

Given the impact of dimension reduction on downstream analyses of scRNAseq data, it is important to continue to study the effects of normalization, scaling, and similarity metrics on the overall performance of the dimension reduction process. The performance of dimension reduction approaches depends the variance structure, noise, and other characteristics of a dataset; we find that as has been reported elsewhere (18), the performance of these methods varies depending on the characteristics of individual datasets. This underscores the importance of including process quality controls and tests to ensure that methods perform consistently and are selected appropriately for a given dataset and application. Benchmarking studies are limited by lack of robust reference datasets that reflect the depth of complexity and nuance in actual biological research contexts. To date, most high-quality, "ground-truth" benchmarking datasets are derived from simple, *in vitro,* synthetic or "pseudo"-cell mixtures, derived from pools of distinct cell types which neither reflect either the true diversity of cell types found in human tissues or properties of real-world research data. In biological tissue, parameters like number of "true" clusters are unknown *a priori* and depend heavily on the specific research question and context. A complementary approach is to consider benchmarking datasets obtained by sequencing complex tissue samples, although these datasets also have their own disadvantages; cells derived from tissue in such studies are largely assigned identities based on one analytical method (and for one particular set of study objectives) without a way of independently validating the assignments—and therefore set an unreliable standard for future benchmarking studies of other methods which can never outperform the method used for initial assignment. As the field advances with respect to systematic benchmarking frameworks for complex datasets in different contexts, we will be better equipped to test the merits of each approach and identify optimal approaches based on data characteristics.

The analysis we present here is limited by the annotations of our benchmarking datasets. With the exception of the SCMixology and Zhengmix datasets (which both comprise well-defined cell clusters and are artificially simpler than data from complex tissues), the datasets we analyzed did not have independently validated cell type annotations and so our assessment of performance is limited by the original cell type assignments. Even if a given method better distinguishes important sub-populations or a rare cell type from clustering, these advantages may not be reflected in the ARI, and the method would actually receive a small penalty for differences from "reference." Furthermore, given the complexity of and subjectivity inherent in cell cluster annotation, researchers may call different cell populations or clusters even on the same dataset, depending upon the research objectives. The diversity of research questions and data challenges that arise in single cell biology necessitate the breadth of statistical and

computational approaches; it is unlikely that a “one size fits all” pipeline or tool that will be superior in all analytical tasks. That said, the robust conceptual framework for CA and its empirical performance advantages over PCA argue for its incorporation in scRNAseq analyses.

We implemented CA, CA-FT, and other variations that adjust for overdispersion for scRNAseq in the R/Bioconductor package *corral*, enabling its integration into commonly used analytical pipelines. Moreover, detailed documentation, tutorials, and vignettes are available (3,37). We conclude with ideas for future development—CA, especially when situated within the broader duality diagram framework, can serve as both a platform for and rich source of further methods adaptation and development. By simultaneously visualizing both cell and gene embeddings, the CA biplot emphasizes the row-column duality inherent in these data and facilitates joint analysis of genes and cells. Furthermore, the unified approach to analysis of gene and cell embeddings and a natural framework to extend and/or integrate with other approaches, including gene set enrichment analysis, supervised decomposition, and projection of supplementary data into a shared latent space—for example, with a similar approach as used previously in *mogsa* and *omicade4* (10,34,36). Cell and gene embeddings can be used as matrix operators to project supplementary data into shared latent space, which facilitates approaches for multi-modal and multi-batch integration, as well as fast approximation methods. Matrix projection via multiplication is fast and scalable, even for very large datasets, and in future extensions, can be used as the basis for fast, approximate dimension reduction approaches based on decomposing a small, representative subset of the data and then projecting into the space the full matrix. Especially as advances in library preparation methods enable analysis of ever-larger numbers of individual cells, computational considerations are critical in selecting analytical methods and designing scRNAseq pipelines.

Methods

Standard correspondence analysis on a single table

Similar to many other matrix factorization methods, correspondence analysis comprises two main steps: a data transformation routine (see also Figure 1A), and a matrix decomposition operation (such as SVD or eigen analysis). In applying "standard" CA to scRNAseq count data, we use SVD to decompose Pearson residuals of gene-by-cell expression count matrix, where the residual quantifies the difference between the observed and the expected data. In this case, the expected value is the product of the row and column weight from the original count matrix. A positive residual, indicating that the observed value (count) for that feature/gene and cell pair is higher than expected, suggests an association or co-dependency; correspondingly, a negative residual shows a lower value than expected, suggesting indicating a negative association between the expression of a gene expression and a cell subpopulation. When squared, the residuals are chi-squared distributed random variables, and their sum of squares comprises a chi-squared goodness-of-fit test statistic with (n-1)(m-1) degrees of freedom (47,65).

Correspondence analysis is a dual scaling along the rows and the columns of each count matrix.
CA applied to scRNAseq count data proceeds through the following two discrete steps:

1. Transformation from counts to standardized residuals. Suppose $\mathbf{X}$ is an $m \times n$ matrix with n cells (indexed on j) in the columns and m features (indexed on i) in the rows, comprising observations x_{ij}. The abundance p_{ij}, the weight of the ith row $p_{i.}$, and the weight of the jth column $p_{.j}$ for a given observation x_{ij} are:

$$p_{ij} = \frac{x_{ij}}{N} \quad ; \quad N = \sum_{i=1}^{m} \sum_{j=1}^{n} x_{ij}$$

$$p_{i.} = \text{row weights} = \sum_{j=1}^{n} p_{ij} \qquad p_{.j} = \text{col weights} = \sum_{i=1}^{m} p_{ij}$$

The expected abundance for observation x_{ij} is $p_{i.}\, p_{.j}$ and is what we would expect to see in a cell assuming there is no relationship between a row and column. The standardized (Pearson) residuals $r_{p;ij}$ are the difference between the observed and expected, and can be computed:

$$r_{p;ij} = \frac{\text{observed} - \text{expected}}{\sqrt{\text{expected}}} = \frac{p_{ij} - p_{i.}\, p_{.j}}{\sqrt{p_{i.}\, p_{.j}}}$$

This transformation is equivalent to the computation applied in contingency table analysis of categorical data measuring the strength of association between elements in a row and a column. It yields a matrix $\mathbf{M_S}$ where the sum of the distances of the points to their centroid ("total inertia") is the chi-squared statistic of the matrix (26,28). As a result of this transformation $\mathbf{M_S}$ is centered and should appear more Gaussian, and therefore is appropriate input for SVD.

2. Matrix decomposition. $\mathbf{M_S}$ is decomposed using singular value decomposition (SVD) to find left singular matrix $\mathbf{U}$, diagonal matrix of singular values $\mathbf{D}$, and right singular matrix $\mathbf{V}$ such that:

$$\mathbf{M_S} = \mathbf{UDV^T}$$

and

$$\mathbf{V^TV} = \mathbf{U^TU} = \mathbf{I}$$

The resulting $\mathbf{U}$ matrix can be either used as an embedding directly, with each column representing a dimension in the new latent space, or coordinate scores can be computed. Standard coordinate scores are given by dividing the $\mathbf{U}$ and $\mathbf{V}$ matrices by the vectors of row weights and column weights, respectively. Principal coordinate scores are given by multiplying the standard coordinate scores by the vector of diagonal values of the matrix $\mathbf{D}$. The principal coordinate scores differ from the standard coordinate scores by a scalar on each dimension, and both reflect the ordination scores of the features and cells (41). Unlike in PCA, where differences in embeddings approximate Euclidean distances, correspondence analysis decomposes the overall chi-squared statistic. The value of the underlying chi-squared statistic is high when there is an association between a row-column pair of the table.

Variations of CA

We considered five variations of CA to address overdispersion in scRNAseq counts (also summarized graphically in Figure 2A).

1. **CA with Freeman-Tukey chi-squared residuals.** Instead of computing the Pearson residuals described above, the residuals are computed:

$$r_{f;ij} = \sqrt{p_{ij}} + \sqrt{p_{ij} + \frac{1}{N}} - \sqrt{4p_{i.}\, p_{.j} + \frac{1}{N}}$$

   The matrix of these residual values is then decomposed with SVD as described in Step 2 above.
2. **CA with variance stabilizing transform: Square root.** The square root of the matrix of counts $\mathbf{X}$ is computed before performing the residual transformation.
3. **CA with variance stabilizing transform: Anscombe.** Each element x_{ij} of the matrix of counts $\mathbf{X}$ is transformed to $x_{ij}^* = 2\sqrt{x_{ij} + \frac{3}{8}}$. The residual transformation is computed on the variance-stabilized counts matrix $\mathbf{X}^*$.
4. **CA with variance stabilizing transform: Freeman-Tukey.** Each element x_{ij} of the matrix of counts $\mathbf{X}$ is transformed to $x_{ij}^* = \sqrt{x_{ij}} + \sqrt{x_{ij} + 1}$. The residual transformation is computed on the variance-stabilized counts matrix $\mathbf{X}^*$.
5. **CA with power deflation.** After performing the Pearson residual transformation, each value in the matrix of residuals is transformed to a power of $\alpha \in (0,1)$, while preserving the sign. Each element r_{ij} in the residual matrix is transformed to $r_{ij}^* = \mathrm{sgn}(\mathrm{r_{ij}})\, |r_{ij}|^{\alpha}$. We recommend selecting $\alpha \in [0.9, 0.99]$ for a "soft" smoothing effect, presenting results for $\alpha = 0.9$.

corralm: multi-table adaptation of correspondence analysis

The adaptation of correspondence analysis for to the integration of multiple tables is similar to the method for single tables with additional matrix concatenation operations. When integrating datasets, we employ indexed residuals, by dividing the standardized residuals by the square root of expected proportion to reduce the influence of column with larger masses (library depth), which is a known source of batch effect in scRNAseq studies. Indexed residuals have a straightforward interpretation for example a value of 0.5

indicated that the observed value is 50% higher than the expected value. A value of -0.5 indicated that the observed value is 50% less likely than expected to have a gene-cell association than expected.

1. Match tables and select features. Identify the intersection of features across the k matrices to be integrated, and subset the tables for only those m^* features. While in these analyses we focus on batch integration and therefore match on features, the tables can either be matched by features, for integration across batches, or by cells, for multi-modal integration across 'omic types.

2. Transformation from counts to indexed residuals. Given each table with n cells and m^* features, the row weight $p_{i.}$, column weight $p_{.j}$, and abundance p_{ij} for each observation are computed as described above for standard CA. The indexed residuals r_{ij} can be computed:

$$r_{ij} = \frac{\text{observed} - \text{expected}}{\text{expected}} = \frac{p_{ij} - p_{i.}p_{.j}}{p_{i.}p_{.j}}$$

   Each table is scaled separately, so as to preserve the internal structure of each dataset.

3. Concatenate matrices. The transformed matrices of indexed residuals are then concatenated along the matching features to form a new matrix $\mathbf{M_C}$ which has m^* features and the total number of cells in the k matrices (i.e., sum of n across k).

4. Matrix decomposition. Singular value decomposition (SVD) is applied to the concatenated matrix of indexed residuals $\mathbf{M_C}$ to find left singular matrix $\mathbf{U}$, diagonal matrix of singular values $\mathbf{D}$, and right singular matrix $\mathbf{V}$ such that:

$$\mathbf{M_S} = \mathbf{UDV^T}$$

   and

$$\mathbf{V^TV} = \mathbf{U^TU} = \mathbf{I}$$

   The columns of the $\mathbf{U}$ matrix then serve as the embeddings generated by this procedure, and the cells correspond to their indices in the concatenated matrix $\mathbf{M_C}$.

Scaled variance plot

When integrating embedding representations across batches, measures for cluster evaluation are effective for assessing group compactness and recovery of cell populations via clustering. However, they do not directly assess how well dataset embeddings are integrated across batches. To focus specifically on batch integration, we developed and applied a heuristic scaled variance metric, which captures the relative dispersion of each batch with respect to the entire dataset. The scaled variance of component dimension d^* for the subset of observations in batch b^*, $SV_{b^*,d}$, is computed with:

$$SV_{b^*,d} = \frac{\mathrm{Var}(\mathbf{E_{b=b^*,d=d^*}})}{\mathrm{Var}(\mathbf{E_{d=d^*}})}$$

where $\mathbf{E}$ is the matrix of embeddings, and b indexes the rows (observations by batch) while d indexes the columns to indicate which component dimension to evaluate. When the datasets are well integrated, SV values for each batch are close to 1, indicating that each batch has similar dispersion as compared to the entire embedding. This metric is appropriate when the types of cells represented in different datasets are expected to be similar but cannot account for situations where the expected distribution of cell types (and therefore, embeddings) is fundamentally different between batches.

Benchmarking

We considered the ten scRNA-seq benchmarking datasets shown in Table 1. The reduced dimension embeddings from each method were clustered using nearest neighbor graph clustering, as implemented in the *bluster* package's default NNGraph parameter set. Performance on the clustering task was assessed with Adjusted Rand Index (ARI) (66), using as "ground truth" the cell type labels from the original datasets. Results shown in Figure 2C are from clustering using different numbers of PCs. Results shown in Figure 2B are computed by taking the maximum across all the tested PCs from Figure 2C, and for glmPCA, the value shown is the average of the maxima achieved by each seed (ten seeds tested in total). Datasets (detailed below) were acquired from three R/Bioconductor data packages: CellBench, DuoClustering2018, and scRNAseq.

Name	Tissue / species	Number of cells	Number of classes	Comments	R/Bioconductor data package	Citation
SC Mixology	Human adenocarcinoma cell lines	1,401	3	Synthetic mixture of 3 cell lines (HCC827, H1975 and H2228) sequenced using 3 library preparation technologies (10X, Celseq, Dropseq)	*CellBench*	(38)
Zhengmix4eq	Human PBMC	3,994	4	Mixture of 4 purified PBMC cell types in approximately equal proportions	*DuoClustering2018*	(39,40)
Zhengmix4uneq		6,498	4	Mixture of 4 purified PBMC cell types in unbalanced proportions		
Zhengmix8eq		3,994	8	Mixture of 8 purified PBMC cell types in approximately equal proportions		
Baron	Human pancreas	8,569	14	Normal human pancreas	*scRNAseq*	(67)
Muraro		3,009	11			(68)
Lawlor		638	8			(69)
Chen	Mouse brain	14,437	47			(70)
Darmanis	Human brain	466	9			(71)
Aztekin	*Xenopus* tail	13,199	32			(72)

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of data and materials

Code and documentation are available in the *corral* R/Bioconductor package:
https://www.bioconductor.org/packages/corral
R code to reproduce the figures and analysis in this manuscript is available on Github at:
https://github.com/laurenhsu1/corral_manuscript
A tutorial describing different implementations of PCA and CA, including *corral*, is available at:
https://aedin.github.io/PCAworkshop

The datasets used in these analyses are detailed in the Benchmarking section of **Methods**, including citations and where the data can be accessed directly through R data packages.

Competing interests

The authors declare that they have no competing interests.

Funding

This project has been made possible in part by grant number CZF2019-002443 from the Chan Zuckerberg Initiative DAF, an advised fund of Silicon Valley Community Foundation, of which ACC is a grantee. LH is funded in part by the NIH NIGMS Biostatistics Training Grant Program (Predoctoral training grant T32GM135117).

Authors' contributions

LH and ACC wrote the manuscript and conceptualized the methods presented. ACC wrote the Bioconductor workshop vignette on CA. LH wrote R package *corral* and code to perform analyses, created figures.

Acknowledgements

We are grateful for helpful discussions with Prof. John Quackenbush and his lab at Harvard TH Chan School of Public Health, and with Bioconductor colleagues funded by the Chan Zuckerberg Initiative seed network program.

Figures

Figure 1

Fig 1. Correspondence Analysis (CA) is a count-based alternative to PCA that is robust for use with raw and log-normalized counts

A. Graphical overview of scRNAseq dimension reduction with matrix factorization

Step 1
Transform count matrix to residuals (raw, log-normalized, or otherwise pre-processed)

n cells, *m genes*, **X**

Standard CA: Pearson residuals

1. Convert matrix from counts (**x**) to abundances (**p**)

$$p_{ij} = \frac{x_{ij}}{N} \qquad N = \sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij}$$

2. Compute standardized Pearson residuals (**r_p**)

$$r_{p;ij} = \frac{p_{ij} - p_{i.}\, p_{.j}}{\sqrt{p_{i.}\, p_{.j}}} = \frac{\text{observed} - \text{expected}}{\sqrt{\text{expected}}} \qquad p_{i.} = \text{row weights} = \sum_{i=1}^{m} p_{ij} \qquad p_{.j} = \text{col weights} = \sum_{j=1}^{n} p_{ij}$$

Principal Component Analysis (PCA): Z-score residuals

Center and scale normalized counts matrix (by features; columns)

$$r_{z;ij} = \frac{x_{ij} - \mu_i}{\sigma_i} \qquad \mu_i = \text{mean of gene i} \qquad \sigma_i = \text{standard deviation of gene i}$$

Step 2
Factor the transformed residual matrix using singular value decomposition (SVD)

n cells, *m genes*, **R** = *k PCs*, *m genes*, *k singular values*, *n cells*, *k PCs*

X* = **U** **D** **V^T**

Step 3
Visualize embeddings directly as a component plot or biplot; use embeddings in downstream analysis

- Compute Standard and/or Principal coordinates
- Use UMAP or t-SNE to summarize CA results in 2 dimensions

corral dim2 / **corral dim1**

B. Comparison between decomposition of Z-score vs. Chi-squared transformed synthetic benchmarking mixture

PCA | corral; Dim1, Dim2; mix

C. CA is robust for use with raw and log-transformed counts

Counts | *Log counts*; **CA**, **PCA**; corral1, corral2, pca1, pca2

Cell type: b.cells, cd14.monocytes, naive.cytotoxic, regulatory.t

D. Comparison of NNGraph clustering by pre-processing pipeline

counts | logcounts; Dataset: scmix_10x (0.61, 0.47 | 0.54, 0.53), scmix_celseq (0.9, 0.61 | 0.74, 0.83), scmix_dropseq (0.68, 0.68 | 0.7, 0.67), zhengmix4eq (0.78, 0.62 | 0.83, 0.83), zhengmix4uneq (0.47, 0.39 | 0.51, 0.45), zhengmix8eq (0.65, 0.27 | 0.51, 0.46); ARI; method: corral, pca

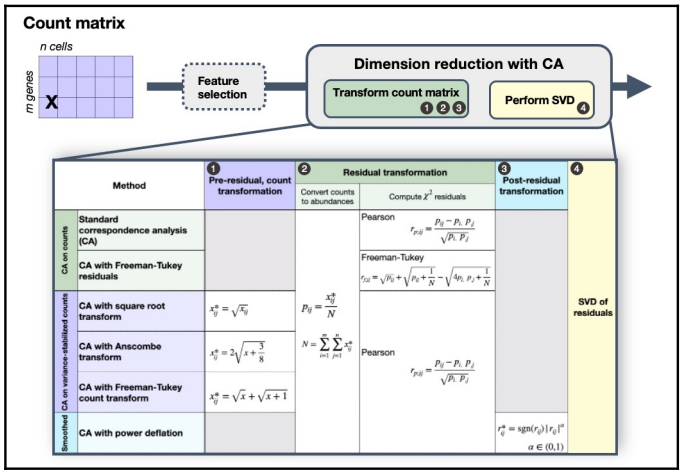
Figure 1. Correspondence Analysis (CA) is an alternative to PCA for count data that is robust for use with raw and log-normalized counts. **A**. Graphical overview of steps for dimension reduction with matrix factorization, including standard (classical) CA and PCA for reference. Standard CA and PCA can be computed with singular value decomposition (SVD) of the Pearson or the Z-score residuals, respectively. **B.** Plots show the first two components generated from PCA (on logcounts; left) and from CA (*corral* on counts; right) applied to a synthetic benchmarking mRNA mixture with 8 groups (data distributed in the CellBench R package; adapted from (3)). "Cells" are colored by group. CA resolves the groups into

clusters, whereas standard PCA is driven by a gradient in the second component and fails to resolve the groups. **C.** Plots show the first two components generated by CA (*corral*; top row) and PCA (bottom row) on both counts (left column) and logcounts (right column) of the Zhengmix4eq dataset, which comprises approximately 4,000 purified PBMCs in approximately equal mixtures. Cells are colored by type. CA is robust for use with counts or logcounts, whereas PCA on counts results in a Guttman, horseshoe, or arch effect. **D.** CA (green) and PCA (purple) were applied to counts (left column) and logcounts (right column) from six benchmarking datasets (SCMixology, distributed in the CellBench R package, and Zhengmix datasets, distributed in the DuoClustering R package). Embeddings from all approaches were used as input for NNGraph clustering, with performance in recovering published clusters assessed using Adjusted Rand Index (ARI). CA consistently meets or exceeds performance of PCA. An orange circle marks the highest ARI achieved in each dataset.

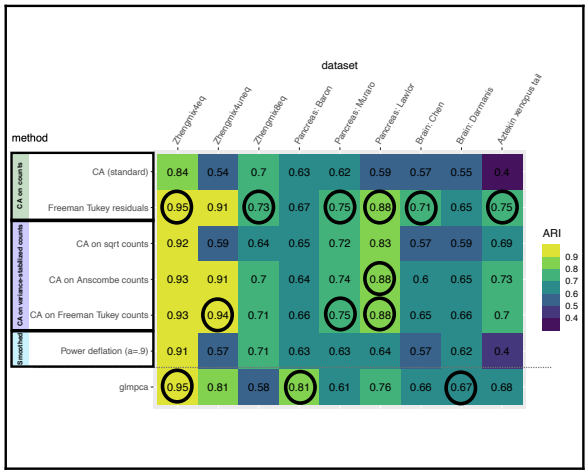
Figure 2

Fig 2. CA can be adapted to address overdispersion in count data

A. Variations of CA



B. Comparison of NNGraph clustering ARI by dimension reduction method



C. Comparison of NNGraph clustering ARI by dimension reduction method and number of included components

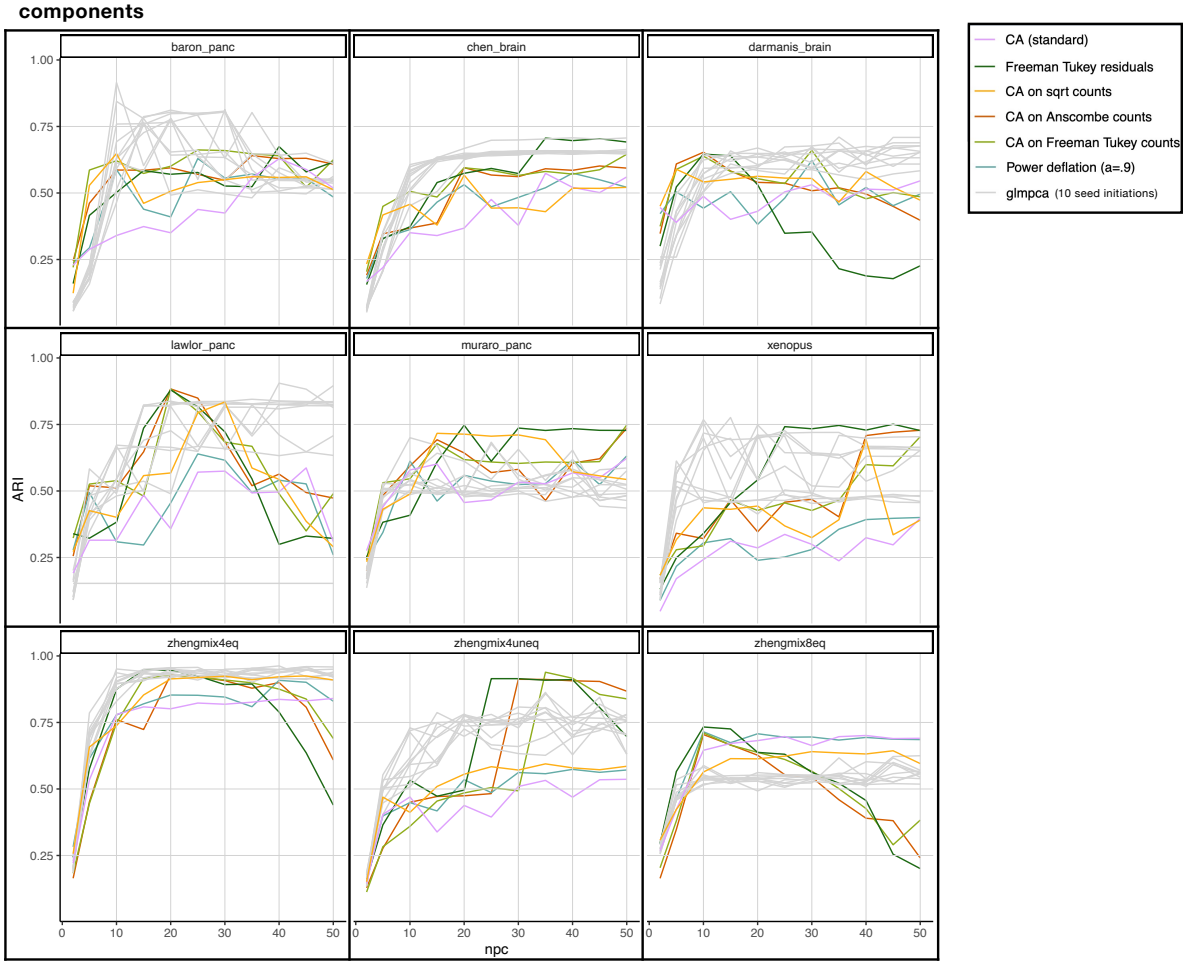


Figure 2. CA adaptations to address overdispersion in count data. **A.** Table summarizing the standard CA procedure and five adaptations to address overdispersion. The first set (row 1 and 2) include methods

that involve no transformations apart from computing chi-squared residuals. The second set (rows 3-5) feature variance stabilizing transformations performed on counts prior to standard CA. The third approach (row 6) smooths the chi-squared residual matrix with a minor "power deflation" prior to decomposition with SVD. **B.** Table of NNGraph cluster recovery performance achieved by each method (rows), in nine datasets (columns), reporting the maximum ARI selected across a range of PCs (full results of ARI by PC shown in Figure 2**C**), with ARI from ten runs of glmPCA were averaged prior to selecting the maximum. Highest ARI (to two decimal places) in each dataset is circled, and the cell clusters in the original datasets are used as the reference groupings. Freeman-Tukey residuals exhibit the best overall performance, with the highest ARI in 6 of the 9 datasets. **C.** Plot of ARI by number of components in each of nine datasets (same as **B**), colored by method. Results for glmPCA (gray) include ten seeds.

Figure 3

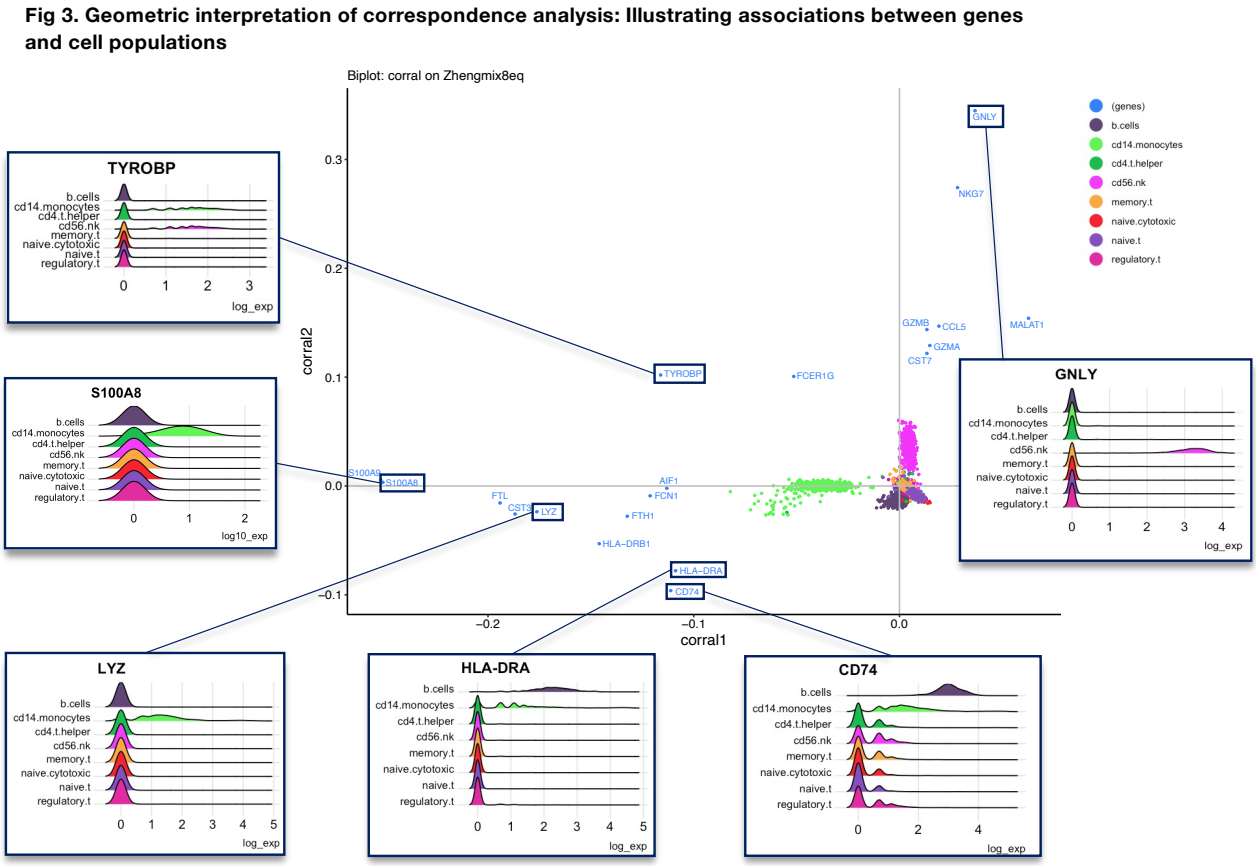
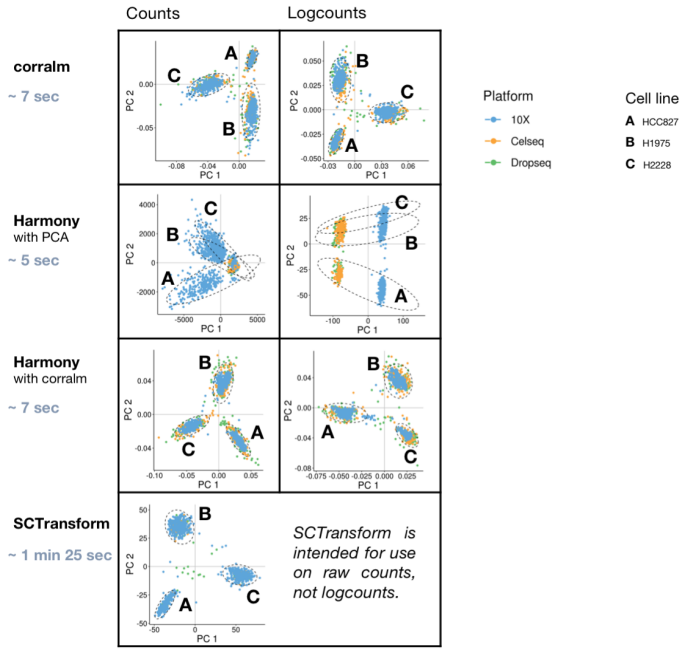


Figure 3. Geometric interpretation of correspondence analysis: Illustrating associations between genes and cell populations. Biplot of the first two dimensions of CA in the Zhengmix8 dataset. The eight cell populations are colored by type, while genes are labeled and colored in blue. The top twenty genes by weight (furthest from the origin in the first two components) are shown. Six biologically significant genes are highlighted, and ridge plots illustrate their log-expression: GNLY is highly expressed in NK cells, whereas TYROBP is highly expressed in both NK and CD14 monocytes. LYZ and S100A8 are both highly expressed, monocyte-specific genes. Both CD74 and HLA-DRA are highly expressed in B cells, and moderately expressed in monocytes, as shown in the respective ridgeplots.

Figure 4

Fig 4. corralm, a multi-table adaptation of CA, integrates count matrices across batches by finding a shared, low-dimensional latent space

A. Batch integration methods
Top 2 components, with approx. run-time for each method



B. Scaled variance by batch
Values closer to 1 indicate similar dispersion by dataset (i.e., better integration)

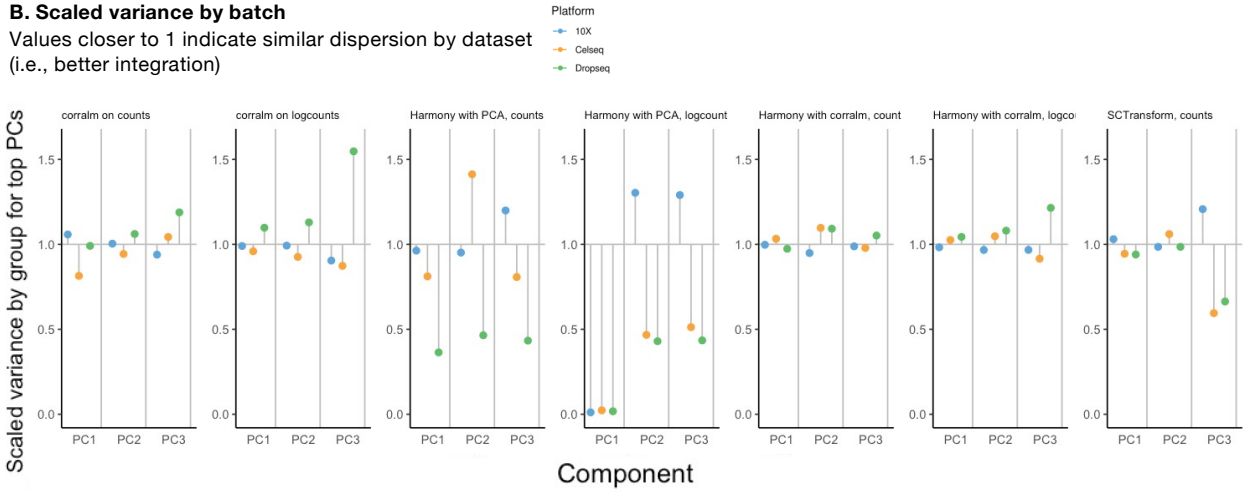


Figure 4. The corralm multi-table adaptation of CA integrates count matrices across batches by finding a shared, low-dimensional latent space. **A.** Comparison of seven integration workflows on the SCMixology benchmarking dataset (comprising mixtures of three cell lines: H2228, H1975, and HCC827 that were each used with three library preparation protocols—Dropseq, Celseq2, and 10X—followed by Illumina sequencing) The first column shows results on counts, and the second column shows logcounts. corralm is both fast and performant and can be combined with methods such as Harmony (the 3rd row) to further improve performance. **B**. Scaled variance (SV) of the batches representing the three SCMixology

library preparation platforms, computed on the first three components of counts and logcounts presented in Figure 4A, colored by batch. SV close to 1 indicate that embeddings exhibit similar distribution across batches. corralm, Harmony with corralm, and SCTransform exhibit good batch alignment, while Harmony with PCA shows values far from 1, suggesting that the embeddings were not successfully integrated across batches.

Figure 5

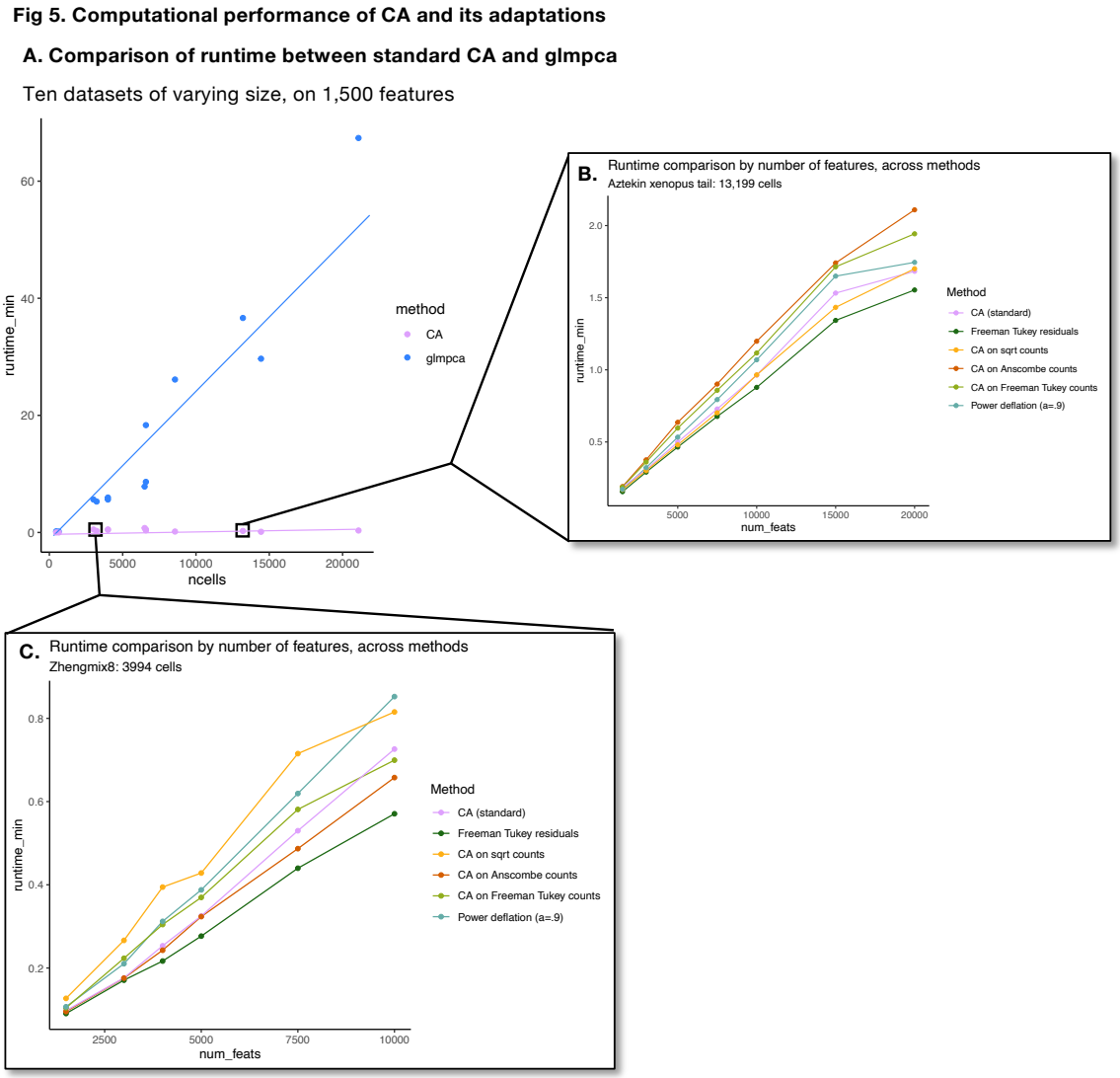


Figure 5. Computational performance of CA and its adaptations. **A**. Plot comparing runtime for standard CA and glmPCA on ten datasets, selecting down to 1,500 features in each. Standard CA consistently runs in under a minute, even for datasets with over 20,000 cells, while glmPCA scales less favorably and requires over an hour for the equivalent input matrix (1,500 features x ~22,000 cells). **B**. Plot comparing runtime with increasing number of features in the Aztekin *Xenopus* tail dataset, across the CA adaptation methods. Since they use similar routines, their runtimes are fairly similar. **C**. Plot comparing runtime with increasing number of features in the Zhengmix8 dataset, across the CA adaptation methods. In both **B** and **C**, it is notable that even with an order of magnitude more features, CA and its adaptations run in a fraction of the time glmPCA takes.

References

1. Hicks SC, Townes FW, Teng M, Irizarry RA. Missing data and technical variability in single-cell RNA-sequencing experiments. Biostatistics. 2018 Oct 1;19(4):562–78.

2. Townes FW. Feature selection and dimension reduction for single-cell RNA-seq based on a multinomial model. 2019;16.

3. Amezquita RA, Lun ATL, Becht E, Carey VJ, Carpp LN, Geistlinger L, et al. Orchestrating single-cell analysis with Bioconductor. Nat Methods. 2020 Feb 2;17(2):137–45.

4. Hsu LL, Culhane AC. Impact of Data Preprocessing on Integrative Matrix Factorization of Single Cell Data. Front Oncol. 2020 Jun 23;10:973.

5. Kiselev VY, Andrews TS, Hemberg M. Challenges in unsupervised clustering of single-cell RNA-seq data. Nat Rev Genet. 2019 May;20(5):273–82.

6. Nguyen LH, Holmes S. Ten quick tips for effective dimensionality reduction. Ouellette F, editor. PLOS Comput Biol. 2019 Jun 20;15(6):e1006907.

7. Stein-O'Brien GL, Arora R, Culhane AC, Favorov AV, Garmire LX, Greene CS, et al. Enter the Matrix: Factorization Uncovers Knowledge from Omics. Trends Genet. 2018 Oct;34(10):790–805.

8. Holmes S. Multivariate data analysis: The French way. In: Institute of Mathematical Statistics Collections [Internet]. Beachwood, Ohio, USA: Institute of Mathematical Statistics; 2008 [cited 2021 Oct 11]. p. 219–33. Available from: http://projecteuclid.org/euclid.imsc/1207580085

9. Hotelling H. Relations Between Two Sets of Variates. Biometrika. 1936 Dec;28(3/4):321.

10. Meng C, Zeleznik OA, Thallinger GG, Kuster B, Gholami AM, Culhane AC. Dimension reduction techniques for the integrative analysis of multi-omics data. Brief Bioinform. 2016 Jul;17(4):628–41.

11. Pearson K. On lines and planes of closest fit to systems of points in space. Lond Edinb Dublin Philos Mag J Sci. 1901 Nov;2(11):559–72.

12. Diaconis P, Goel S, Holmes S. Horseshoes in multidimensional scaling and local kernel methods. Ann Appl Stat [Internet]. 2008 Sep 1 [cited 2021 Oct 11];2(3). Available from: https://projecteuclid.org/journals/annals-of-applied-statistics/volume-2/issue-3/Horseshoes-in-multidimensional-scaling-and-local-kernel-methods/10.1214/08-AOAS165.full

13. Hafemeister C. Normalization and variance stabilization of single-cell RNA-seq data using regularized negative binomial regression. 2019;15.

14. Lause J, Berens P, Kobak D. Analytic Pearson residuals for normalization of single-cell RNA-seq UMI data. Genome Biol. 2021 Dec;22(1):258.

15. Durif G, Modolo L, Mold JE, Lambert-Lacroix S, Picard F. Probabilistic count matrix factorization for single cell expression data analysis. Birol I, editor. Bioinformatics. 2019 Oct 15;35(20):4011–9.

16. Pierson E, Yau C. ZIFA: Dimensionality reduction for zero-inflated single-cell gene expression analysis. Genome Biol. 2015 Dec;16(1):241.

17. Risso D, Perraudeau F, Gribkova S, Dudoit S, Vert J-P. A general and flexible method for signal extraction from single-cell RNA-seq data. Nat Commun. 2018 Dec;9(1):284.

18. Sun S. Accuracy, robustness and scalability of dimensionality reduction methods for single-cell RNA-seq analysis. 2019;21.

19. Benzécri J-P. Problèmes statistiques et méthodes géométriques. Cah Anal Données. 1978;3(2):131–46.

20. Benzécri J-P, others. L'analyse des données. Vol. 2. Dunod Paris; 1973.

21. De la Cruz O, Holmes S. The duality diagram in data analysis: Examples of modern applications. Ann Appl Stat [Internet]. 2011 Dec 1 [cited 2021 Sep 26];5(4). Available from: https://projecteuclid.org/journals/annals-of-applied-statistics/volume-5/issue-4/The-duality-diagram-in-data-analysis--Examples-of-modern/10.1214/10-AOAS408.full

22. Escoufier Y. The duality diagram: A means of better practical applications. In: Legendre P, Legendre L, editors. Developments in Numerical Ecology. Berlin: Springer; 1987.

23. Escoufier Y. Operator related to a data matrix: a survey. In: Rizzi A, Vichi M, editors. Compstat 2006 - Proceedings in Computational Statistics [Internet]. Heidelberg: Physica-Verlag HD; 2006 [cited 2021 Oct 11]. p. 285–97. Available from: http://link.springer.com/10.1007/978-3-7908-1709-6_22

24. Legendre P, Legendre L. Numerical ecology. Third English edition. Amsterdam: Elsevier; 2012. 990 p. (Developments in environmental modelling).

25. Holmes S, Huber W. Modern statistics for modern biology. Cambridge, United Kingdom: Cambridge university press; 2019.

26. Greenacre MJ. Correspondence analysis: Correspondence analysis. Wiley Interdiscip Rev Comput Stat. 2010 Sep;2(5):613–9.

27. Digby PGN, Kempton RA. Multivariate Analysis of Ecological Communities [Internet]. Dordrecht: Springer Netherlands; 1987 [cited 2021 Oct 11]. Available from: https://doi.org/10.1007/978-94-009-3135-0

28. Greenacre MJ. Theory and applications of correspondence analysis. London ; Orlando, Fla: Academic Press; 1984. 364 p.

29. Grantham R, Gautier C, Gouy M, Mercier R, Pavé A. Codon catalog usage and the genome hypothesis. Nucleic Acids Res. 1980;8(1):197–197.

30. Perriere G. Use and misuse of correspondence analysis in codon usage studies. Nucleic Acids Res. 2002 Oct 15;30(20):4548–55.

31. Fellenberg K, Hauser NC, Brors B, Neutzner A, Hoheisel JD, Vingron M. Correspondence analysis applied to microarray data. Proc Natl Acad Sci. 2001 Sep 11;98(19):10781–6.

32. Busold CH, Winter S, Hauser N, Bauer A, Dippon J, Hoheisel JD, et al. Integration of GO annotations in Correspondence Analysis: facilitating the interpretation of microarray data. Bioinformatics. 2005 May 15;21(10):2424–9.

33. McMurdie PJ, Holmes S. phyloseq: An R Package for Reproducible Interactive Analysis and Graphics of Microbiome Census Data. PLOS ONE. 2013;8(4):11.

34. Culhane AC, Perriere G, Considine EC, Cotter TG, Higgins DG. Between-group analysis of microarray data. Bioinformatics. 2002 Dec 1;18(12):1600–8.

35. Culhane AC, Perrière G, Higgins DG. Cross-platform comparison and visualisation of gene expression data using co-inertia analysis. BMC Bioinformatics. 2003;15.

36. Meng C, Basunia A, Peters B, Gholami AM, Kuster B, Culhane AC. MOGSA: Integrative Single Sample Gene-set Analysis of Multiple Omics Data. Mol Cell Proteomics. 2019 Aug;18(8):S153–68.

37. Culhane AC, Hsu LL. Dimension Reduction for Beginners: Hitchhiker's Guide to Matrix Factorization and PCA [Internet]. 2019. Available from: https://aedin.github.io/PCAworkshop/index.html

38. Tian L, Dong X, Freytag S, Lê Cao K-A, Su S, JalalAbadi A, et al. Benchmarking single cell RNA-sequencing analysis pipelines using mixture control experiments. Nat Methods. 2019 Jun;16(6):479–87.

39. Zheng GXY, Terry JM, Belgrader P, Ryvkin P, Bent ZW, Wilson R, et al. Massively parallel digital transcriptional profiling of single cells. Nat Commun. 2017 Apr;8(1):14049.

40. Duò A, Robinson MD, Soneson C. A systematic performance evaluation of clustering methods for single-cell RNA-seq data. F1000Research. 2020 Nov 16;7:1141.

41. Greenacre M. The contributions of rare objects in correspondence analysis. 2013;94(1):9.

42. Mosteller F, Tukey JW. The Uses and Usefulness of Binomial Probability Paper. J Am Stat Assoc. 1949 Jun;44(246):174–212.

43. Bartlett MS. The Use of Transformations. Biometrics. 1947 Mar;3(1):39.

44. Anscombe FJ. THE TRANSFORMATION OF POISSON, BINOMIAL AND NEGATIVE-BINOMIAL DATA. Biometrika. 1948;35(3–4):246–54.

45. Freeman MF, Tukey JW. Transformations Related to the Angular and the Square Root. Ann Math Stat. 1950 Dec;21(4):607–11.

46. Tukey JW. Exploratory data analysis. Reading, Mass: Addison-Wesley Pub. Co; 1977. 688 p. (Addison-Wesley series in behavioral science).

47. Beh EJ, Lombardo R, Alberti G. Correspondence analysis and the Freeman–Tukey statistic: A study of archaeological data. Comput Stat Data Anal. 2018;15.

48. Plackett RL, Bishop YMM, Fienberg SE, Holland PW. Discrete Multivariate Analysis: Theory and Practice. J R Stat Soc Ser Gen. 1976;139(3):402.

49. Cressie N, Read TRC. Multinomial Goodness-Of-Fit Tests. J R Stat Soc Ser B Methodol. 1984 Jul;46(3):440–64.

50. Greenacre M. Contribution Biplots. J Comput Graph Stat. 2013 Jan;22(1):107–22.

51. Krensky AM, Clayberger C. Biology and clinical relevance of granulysin. Tissue Antigens. 2009 Mar;73(3):193–8.

52. Gonzalez LL, Garrie K, Turner MD. Role of S100 proteins in health and disease. Biochim Biophys Acta BBA - Mol Cell Res. 2020 Jun;1867(6):118677.

53. Wang S, Song R, Wang Z, Jing Z, Wang S, Ma J. S100A8/A9 in Inflammation. Front Immunol. 2018 Jun 11;9:1298.

54. Gordon S, Plüddemann A, Martinez Estrada F. Macrophage heterogeneity in tissues: phenotypic diversity and functions. Immunol Rev. 2014 Nov;262(1):36–55.

55. Tomasello E, Vivier E. KARAP/DAP12/TYROBP: three names and a multiplicity of biological functions. Eur J Immunol. 2005 Jun;35(6):1670–7.

56. Su H, Na N, Zhang X, Zhao Y. The biological function and significance of CD74 in immune diseases. Inflamm Res. 2017 Mar;66(3):209–16.

57. Matern BM, Olieslagers TI, Voorter CEM, Groeneweg M, Tilanus MGJ. Insights into the polymorphism in HLA-DRA and its evolutionary relationship with HLA haplotypes. HLA. 2020 Feb;95(2):117–27.

58. Doledec S, Chessel D. Co-inertia analysis: an alternative method for studying species-environment relationships. Freshw Biol. 1994 Jun;31(3):277–94.

59. Tran HTN. A benchmark of batch-effect correction methods for single-cell RNA sequencing data. 2020;32.

60. Baglama J, Reichel L. Augmented Implicitly Restarted Lanczos Bidiagonalization Methods. SIAM J Sci Comput. 2005 Jan;27(1):19–42.

61. Hirschfeld HO. A Connection between Correlation and Contingency. Math Proc Camb Philos Soc. 1935 Oct;31(4):520–4.

62. Abdi H, Valentin D. Multiple Correspondence Analysis. Encycl Meas Stat. 2007 Jan;

63. Beh EJ, Lombardo R. A GENEALOGY OF CORRESPONDENCE ANALYSIS: A GENEALOGY OF CORRESPONDENCE ANALYSIS. Aust N Z J Stat. 2012 Jun;54(2):137–68.

64. Hill MO. Correspondence Analysis: A Neglected Multivariate Method. Appl Stat. 1974;23(3):340.

65. Pearson K. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. Lond Edinb Dublin Philos Mag J Sci. 1900 Jul;50(302):157–75.

66. Hubert L, Arabie P. Comparing partitions. J Classif. 1985 Dec;2(1):193–218.

67. Baron M, Veres A, Wolock SL, Faust AL, Gaujoux R, Vetere A, et al. A Single-Cell Transcriptomic Map of the Human and Mouse Pancreas Reveals Inter- and Intra-cell Population Structure. Cell Syst. 2016 Oct;3(4):346-360.e4.

68. Muraro MJ, Dharmadhikari G, Grün D, Groen N, Dielen T, Jansen E, et al. A Single-Cell Transcriptome Atlas of the Human Pancreas. Cell Syst. 2016 Oct;3(4):385-394.e3.

69. Lawlor N, George J, Bolisetty M, Kursawe R, Sun L, Sivakamasundari V, et al. Single-cell transcriptomes identify human islet cell signatures and reveal cell-type–specific expression changes in type 2 diabetes. Genome Res. 2017 Feb;27(2):208–22.

70. Chen R, Wu X, Jiang L, Zhang Y. Single-Cell RNA-Seq Reveals Hypothalamic Cell Diversity. Cell Rep. 2017 Mar;18(13):3227–41.

71. Darmanis S, Sloan SA, Zhang Y, Enge M, Caneda C, Shuer LM, et al. A survey of human brain transcriptome diversity at the single cell level. 2015;6.

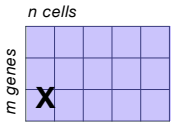
72. Aztekin C, Hiscock TW, Marioni JC, Gurdon JB, Simons BD, Jullien J. Identification of a regeneration-organizing cell in the *Xenopus* tail. Science. 2019 May 17;364(6441):653–8.

Fig 1. Correspondence Analysis (CA) is a count-based alternative to PCA that is robust for use with raw and log-normalized counts

A. Graphical overview of scRNAseq dimension reduction with matrix factorization

Step 1

Transform count matrix to residuals (raw, log-normalized, or otherwise pre-processed)



Standard CA: Pearson residuals

1. Convert matrix from counts (**x**) to abundances (**p**)

$$p_{ij} = \frac{x_{ij}}{N} \qquad N = \sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij}$$

2. Compute standardized Pearson residuals (**r_p**)

$$r_{p;ij} = \frac{p_{ij} - p_{i.}\, p_{.j}}{\sqrt{p_{i.}\, p_{.j}}} = \frac{\text{observed} - \text{expected}}{\sqrt{\text{expected}}}$$

$p_{i.}$ = row weights = $\sum_{i=1}^{m} p_{ij}$

$p_{.j}$ = col weights = $\sum_{j=1}^{n} p_{ij}$

Principal Component Analysis (PCA): Z-score residuals

Center and scale normalized counts matrix (by features; columns)

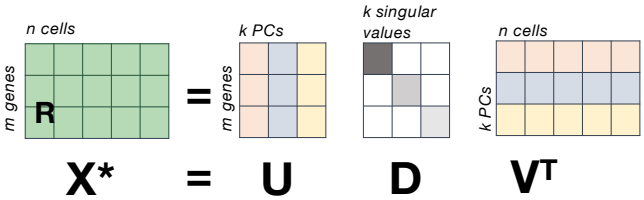
$$r_{z;ij} = \frac{x_{ij} - \mu_i}{\sigma_i}$$

μ_i = mean of gene i

σ_i = standard deviation of gene i

Step 2

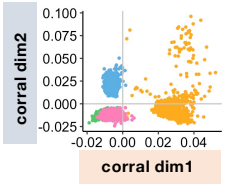
Factor the transformed residual matrix using singular value decomposition (SVD)



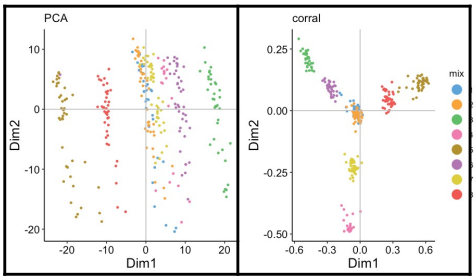
Step 3

Visualize embeddings directly as a component plot or biplot; use embeddings in downstream analysis

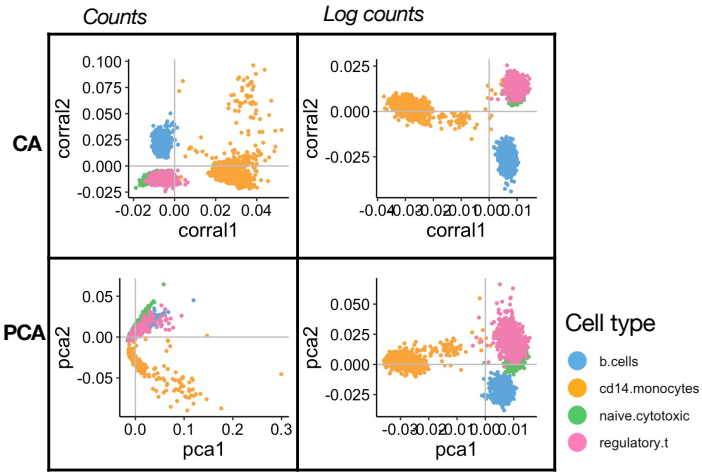
- Compute Standard and/or Principal coordinates
- Use UMAP or t-SNE to summarize CA results in 2 dimensions



B. Comparison between decomposition of Z-score vs. Chi-squared transformed synthetic benchmarking mixture



C. CA is robust for use with raw and log-transformed counts



D. Comparison of NNGraph clustering by pre-processing pipeline

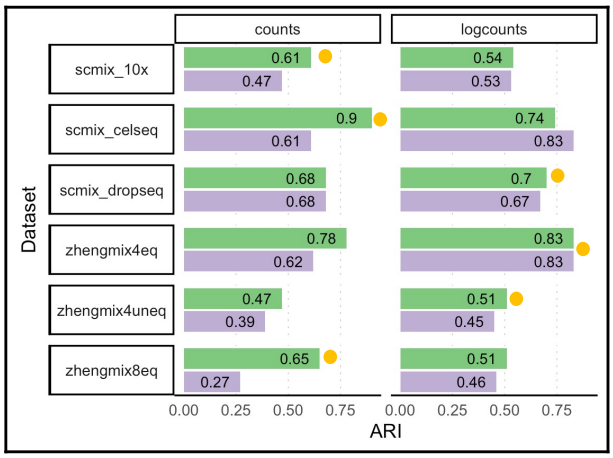
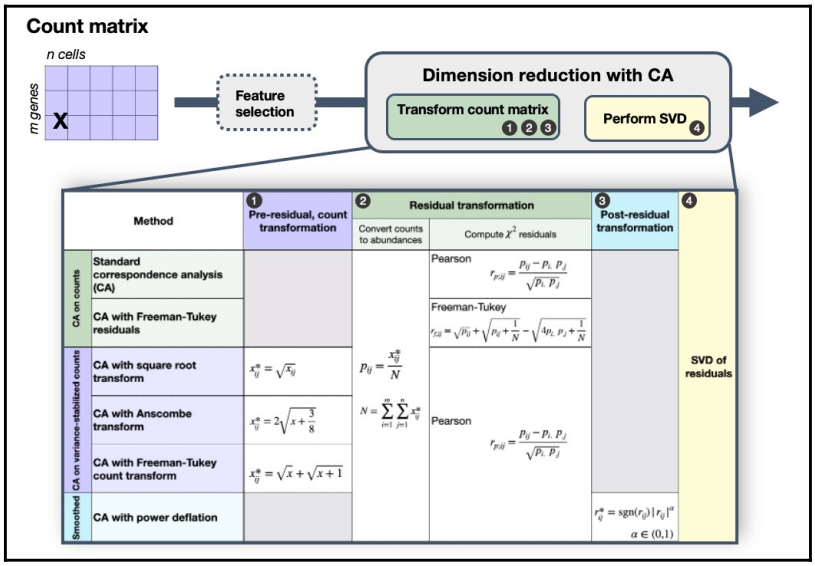
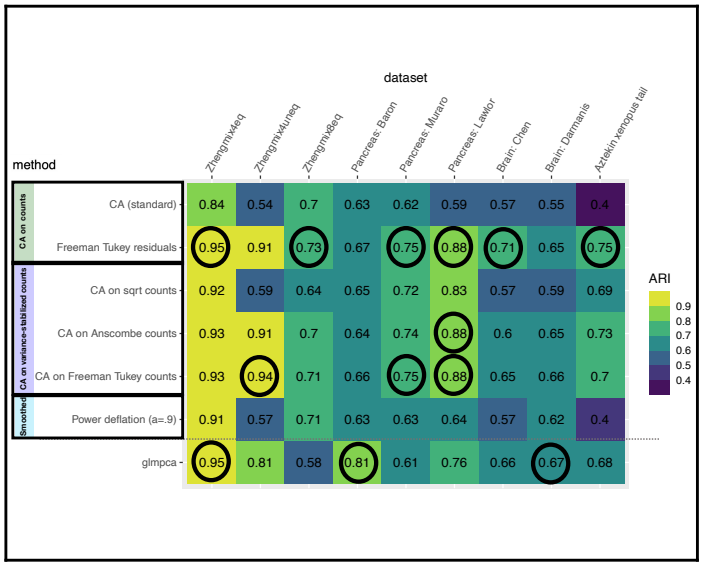


Fig 2. CA can be adapted to address overdispersion in count data

A. Variations of CA



B. Comparison of NNGraph clustering ARI by dimension reduction method



C. Comparison of NNGraph clustering ARI by dimension reduction method and number of included components

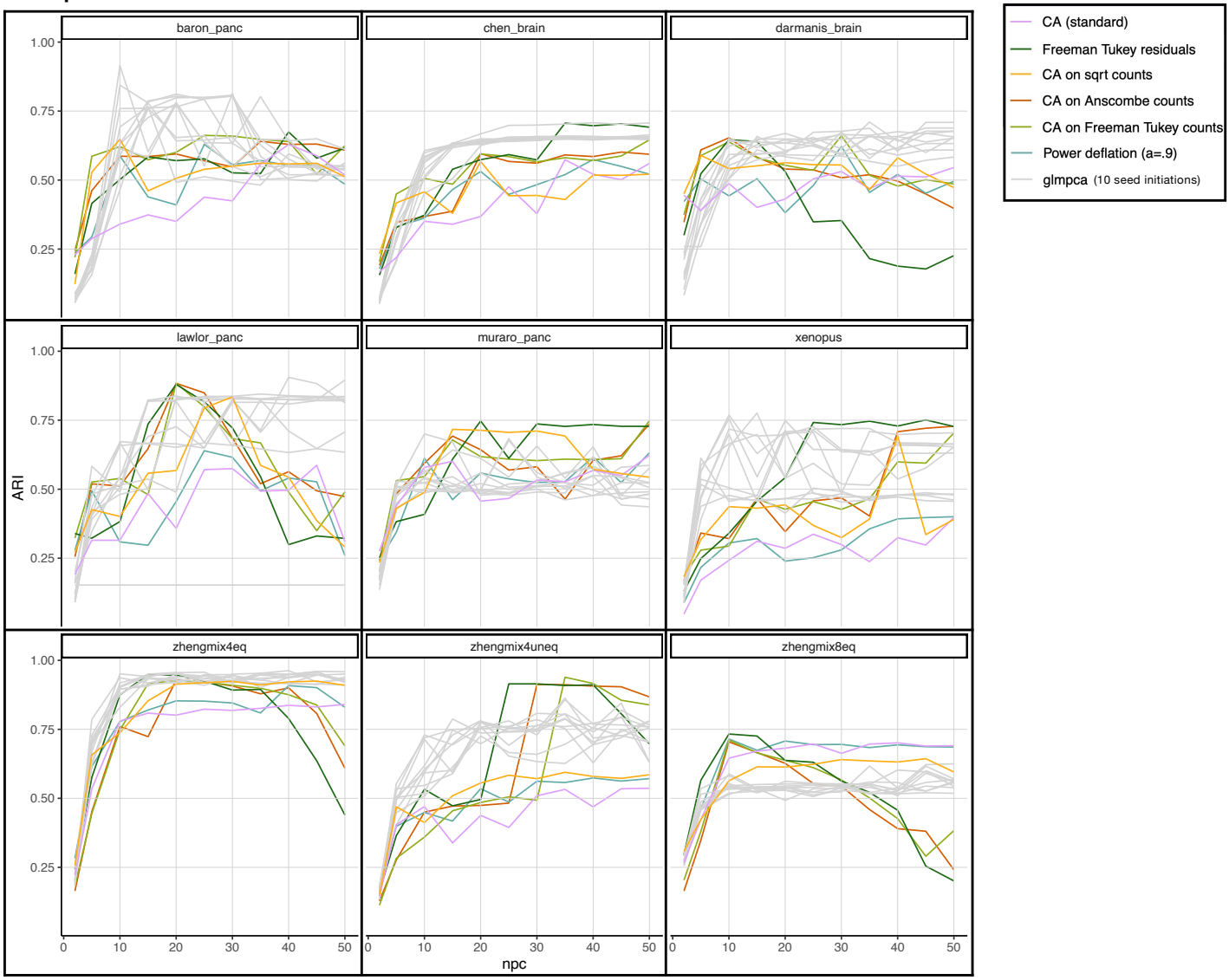


Fig 3. Geometric interpretation of correspondence analysis: Illustrating associations between genes and cell populations

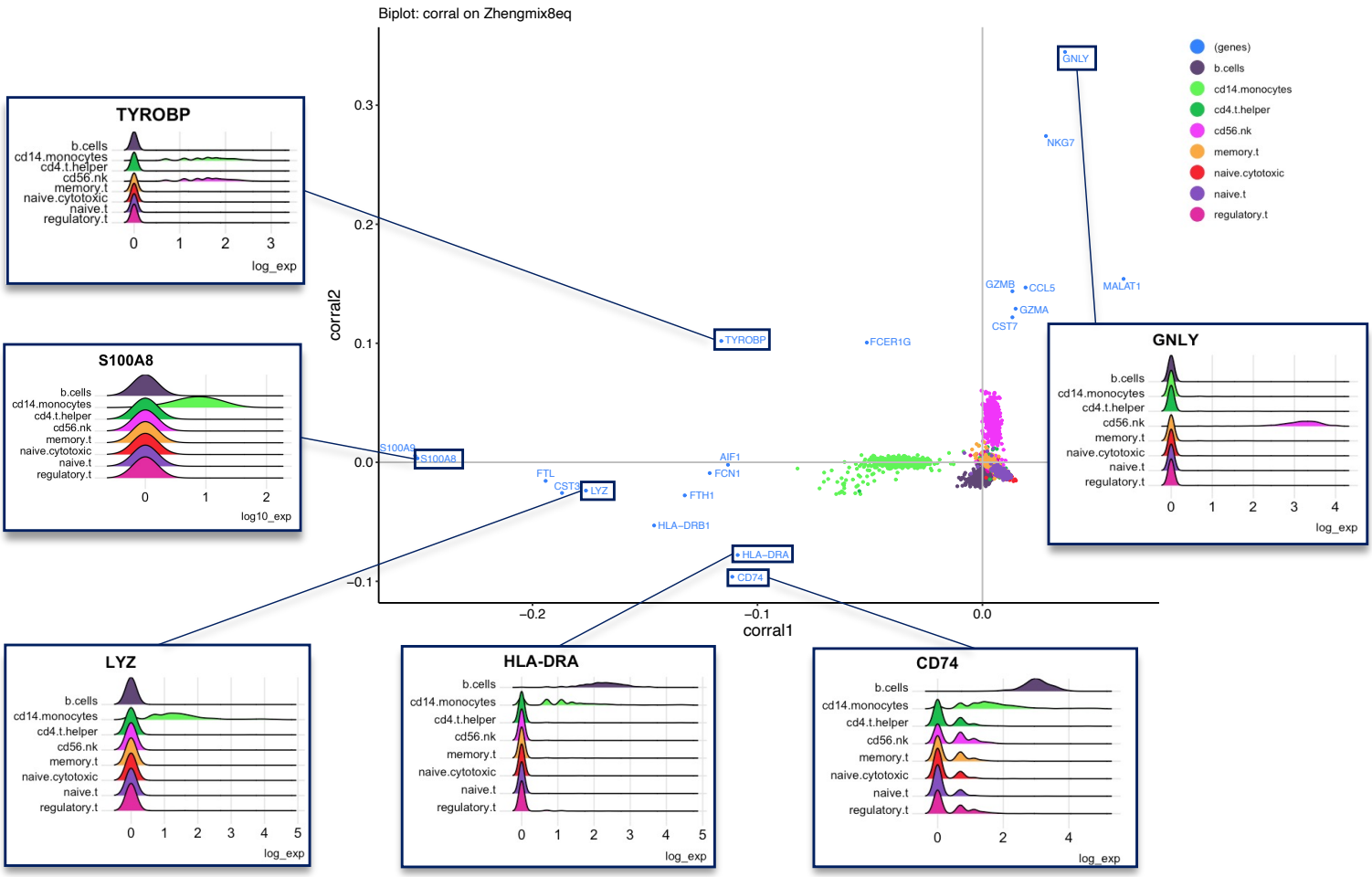
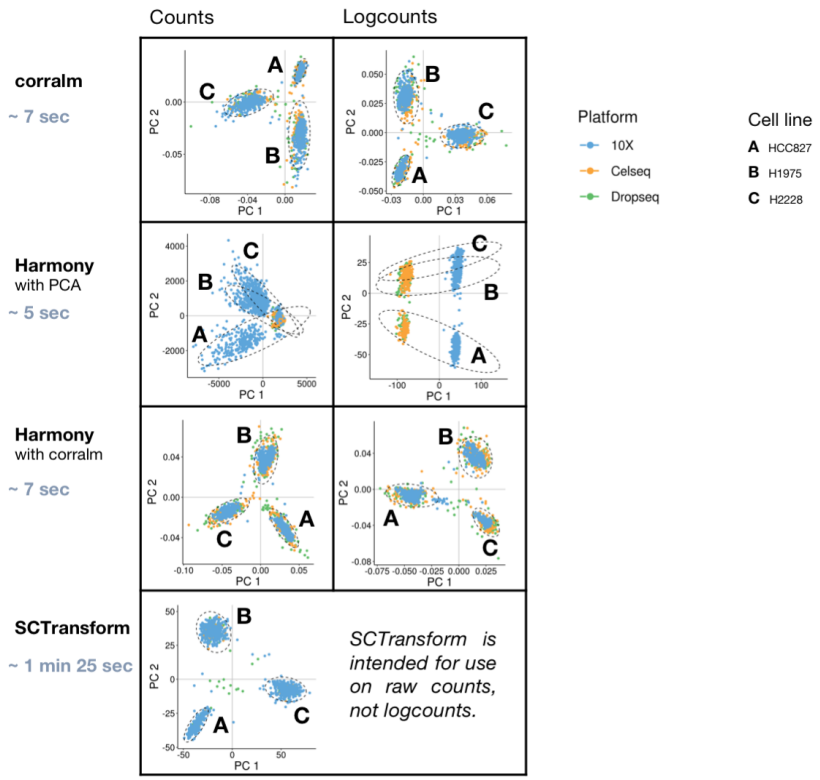


Fig 4. corralm, a multi-table adaptation of CA, integrates count matrices across batches by finding a shared, low-dimensional latent space

A. Batch integration methods

Top 2 components, with approx. run-time for each method



B. Scaled variance by batch

Values closer to 1 indicate similar dispersion by dataset (i.e., better integration)

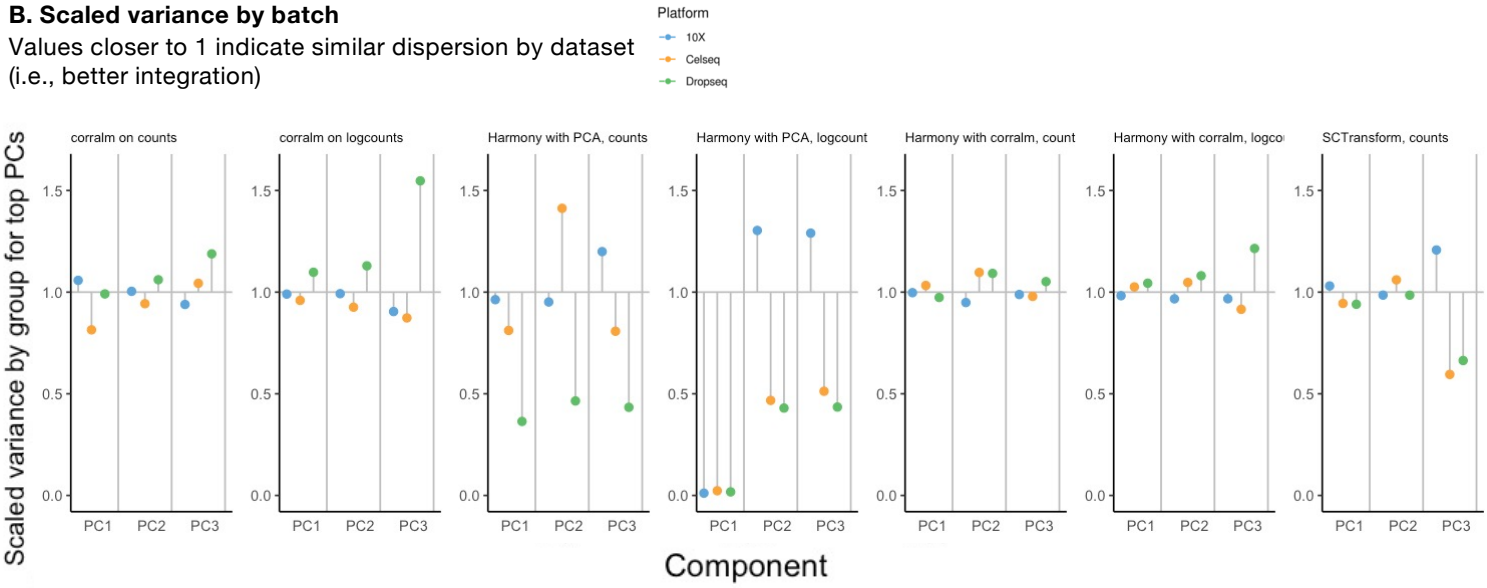
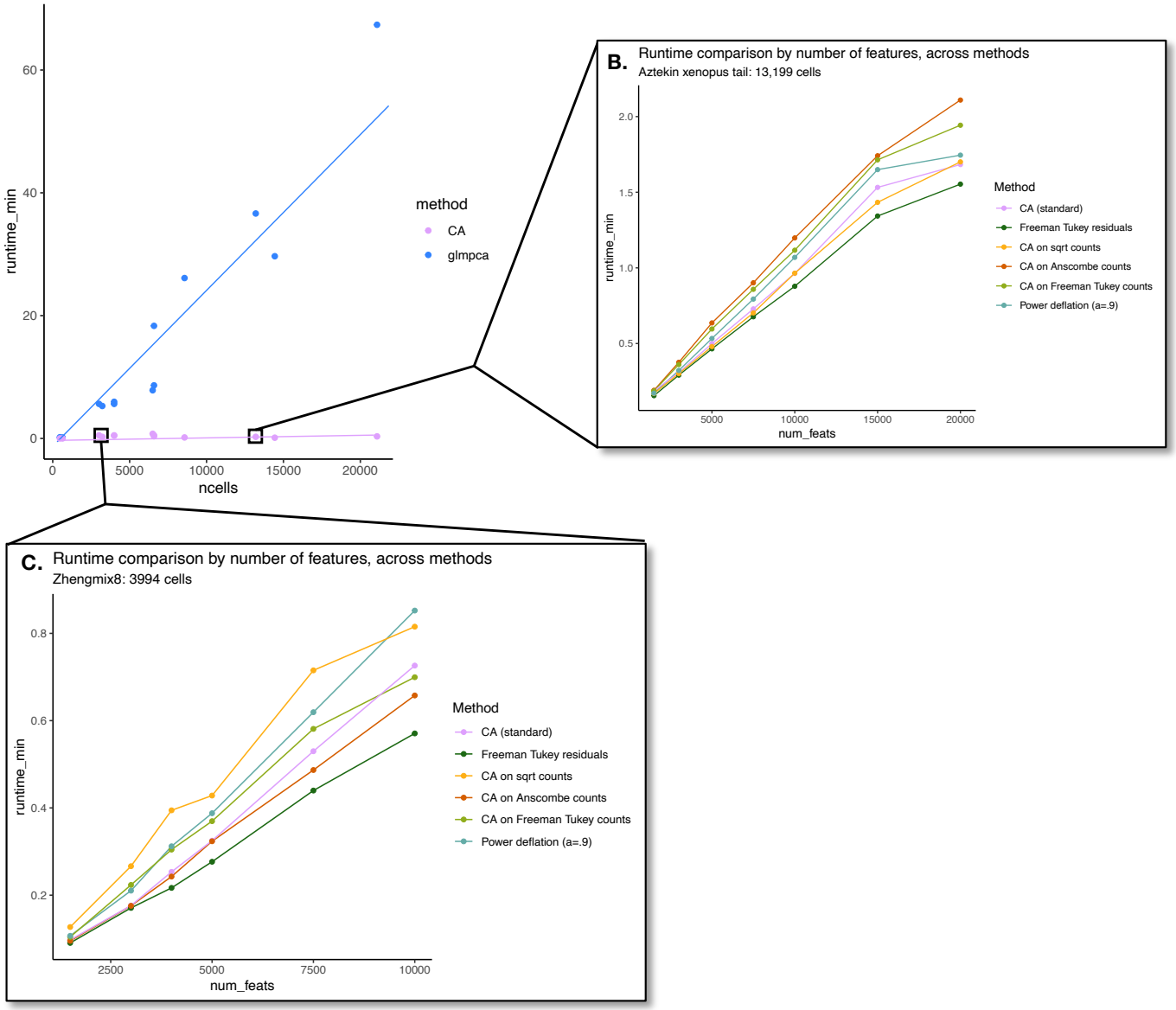


Fig 5. Computational performance of CA and its adaptations

A. Comparison of runtime between standard CA and glmpca

Ten datasets of varying size, on 1,500 features