Contents

1 Introduction

Cells are one of the fundamental units of life. They show an immense complexity and diversity. Their identity and function is determined by environmental stimuli, the physical environment, the cell cycle and neighbouring cells (Wagner et al., 2016). Only recently has it been possible to investigate the transcriptome of a single cell. Single-cell RNA sequencing (scRNA-seq) was first published by Tang et al. (2009). This method addresses new biological issues, such as the identification of rare cell populations, and allows us to measure the frequency of cell types in tissues, characterise differences in similar cell types and investigate the heterogeneity of cell states or cell lineages (Andrews and Hemberg, 2017).

A typical scRNA-seq workflow consists of the isolation of single cells, the extraction of RNA, cDNA library preparation, and the amplification and sequencing of the libraries. A wide variety of scRNA-seq protocols exists, differing in throughput, full transcript or 3' sequencing, costs and automatisation. Small-scale protocols are standard PCR plate-based methods or methods in which cell isolation and library preparation are combined into one protocol. A typical small-scale method is the PCR plate-based SMART-seq2 (Picelli et al., 2013). Libraries are full-transcript sequenced using a standard Illumina sequencing approach. Typically, hundreds of cells are processed and ERCC is used for normalisation. On the other side of the spectrum are droplet-based methods such as Drop-seq and 10xChromium, which allow the processing of thousands of cells.

The differences between scRNA-seq and bulk experiments are the lower sequencing depth (100,000–5 million reads per cell), higher variability and more outliers. scRNA-seq data suffers from technical noise, batch effects and low capture efficiency. Batch effects occur when different biological conditions are processed in different batches, making the deconvolution of technical noise and biological effect impossible. This should be avoided by an appropriate experimental design that allows for the statistical deconvolution between unwanted and wanted variation. In scRNA-seq the single experimental unit is the cell, making it not always possible to use this approach. Different cells in an experiment may need different sample processing, or their biological differences may affect the downstream analyses.

The starting amounts of the library preparation can be as low as ten picogrammes of total RNA (Picelli et al., 2013). Two main issues are arising due to the low starting amount are overamplification and low capture efficiency. Low and moderate expressed genes are not captured during the reverse transcription, which leads to dropouts of genes and a zero-inflated gene expression. scRNA-seq data has an excess of zero counts, which can be split into systematic, semi-systematic and stochastic zeros (Lun et al., 2016a). Systematic zeros and semi-systematic zeros come from genes that are silent in a subpopulation or across all cells, respectively. Stochastic zeros are zero counts that have been obtained due to sampling. They affect genes with a count distribution near zero and have to be dealt with the normalisation steps.

To deal with technical noise Unique Molecular Identifiers (UMI) or spike-in from the External RNA Control Consortium (ERCC) are used. UMI are short random barcodes attached to the single-stranded cDNA in the reverse transcriptase process. By counting the unique UMI reads aligned to the genome an estimated tag count is obtained. Spike-in are added before amplification. Under the assumption that the amplification of the endogenous and exogenous RNA is similar, they can be used for library size normalisation and to remove technical noise.

In general, scRNA-seq experiments consist of high-dimensional data. High-dimensional data suffers from the curse of dimensionality (Wagner et al., 2016). Distances in high-dimensional data become unstable and subpopulations cannot be separated (Andrews and Hemberg, 2017). Additionally, computational requirements are high. Reduction of the dimension is made by two approaches. Using linear or non-linear projections of the data from the original high-dimensional to a lower-dimensional space, or by feature selection, in which uninformative genes are removed.

The detection of new cell types is one of the most common aims for scRNA-seq data. Lately,

a broad spectrum of clustering methods have been specifically developed for the clustering of single cells. The aim of this study is to evaluate clustering methods for scRNA-seq data.

2 Methods

2.1 Dimension reduction

All methods require a dimension reduction step before clustering. Commonly used methods are either Principal Component Analysis (PCA) or t-distributed Stochastic Neighbour Embedding (tSNE). PCA finds a rotation of the original data in which the newly obtained first coordinates have the highest possible variance, the second coordinates the second-greatest variance etc. Practically, PCA is computed by spectral decomposition of the correlation matrix. Dimension reduction is achieved either by selecting only the first few PCs whose eigenvalues are less than average, or by determineing the number graphically by plotting the eigenvalues. PCA is deterministic and relatively fast but restricted to linear spaces.

In contrast, tSNE is a non-linear mapping (Van Der Maaten, 2013). Stochastic neighbour embedding (SNE) transforms Euclidean distances to conditional probabilities $p_{j|i}$. That is the probability of x_j is the nearest neighbour of x_i under a Gaussian centred at x_i. The low-dimensional counterpart $q_{i|j}$ is similar with a Gaussian centered at y_i and variance of $1/\sqrt{(2)}$. SNE minimises the divergence between $p_{j|i}$ and $q_{j|i}$ using the Kullback-Leiber divergence. tSNE implements a Student's t-distribution for the low dimensional space and a symmetric version of the cost function to simplify optimisation and to overcome the crowding problem. In tSNE, the cost function uses the joint probabilities p_{ij} and q_{ij} instead of conditional probabilities. To deal with large datasets, the Barnes-Hut implementation uses random walks on the nearest neighbour network with a PCA step to reduce the dimensionality of the high-dimensional data. tSNE is stochastic, depends on a perplexity parameter and distances between clusters are not preserved.

2.2 Clustering methods

Identifying unknown cell populations is one of the main uses of scRNA-seq data (Andrews and Hemberg, 2017). Eleven clustering methods have been evaluated for this study. These methods and algorithms can be roughly classified into three groups: K-means, graphclustering and hierarchical-based clustering. SC3, SIMLR, Linnorm and RaceID use k-means in different fashions, while pcaReduce and CIDR are based on hierarchical clustering. The graph-based methods are SNN-Cliq and Seurat. An overview of the methods is given in Table ...

K-means K-means clustering finds a predefined number of centres k and cell assignments, such that their within-group sum of squares is minimised (Hartigan and Wong, 1979). k cluster centres are randomly assigned. Each data point is then assigned to the nearest centre using Euclidean distances. The centres are then recomputed using the average of the data points that are assigned to each of the k centres. This procedure is iterated until the algorithm converges. The assigning of the centres is random. Also, it's not guaranteed to find the global minimum. The drawbacks of the method are that it assumes spherical cluster and the sensitivity to scaling.

pcaReduce pcaReduce uses k-means clustering to find the number of clusters in the reduced dimension given by PCA (Yau et al., 2016). The main assumption is that large classes of cells are contained in low-dimension PC representation and more refined subsets of these cells types are contained in higher-dimensional PC representations. Given a gene expression matrix, the clustering algorithm starts with a k-means clustering on the PCA projections $Y_{n\times q}$ with $q+1$ clusters. Where n are the cells and q are

the number of PCs. The number of initial clusters k is typically around 30, guaranteeing that most cell types are captured. For all pairs of clusters, the joint probabilities are computed. Two clusters are merged by selecting the pair with the highest joint probability or by sampling proportionally by the joint probabilities. The number of clusters is now decreased to $K-1$. Next, the PC with the lowest variance is deleted and a k-means clustering with $K-2$ centres is performed. This process is repeated until only one single cluster remains. When using pcaReduce q cluster partitions with $k-1$ clusters are obtained.

SC3 Implemented in the SC3 method is a gene- and cell-filtering, as well as a log transformation step of the expression matrix (Kiselev et al., 2017). The filtered expression matrix is then used to compute Euclidean, Pearson and Spearman dissimilarity measures. By PCA or Laplacian graphs a lower-dimensional representation of the data is obtained. K-means clustering is then performed on the different dimensions. Next, a consensus matrix of the different clustering results is computed. The consensus matrix is a binary similarity matrix with entry one if two cells belong to the same cluster and zero otherwise. The consensus matrix is obtained by averaging the individual clustering. The last step is a hierarchical clustering step with complete linkage. The cluster is inferred by the k level of hierarchy, where k is supplied by the user. To reduce runtime SC3 changes the clustering method when supplied with more than 5'000 cells. Randomly selected cells are then used for the clustering approach described above. These subpopulations are then used to train a support vector machine to infer the remaining cells.

SNNcliq SNNcliq computes a shared nearest-neighbour-graph based on the high-dimensional data (Xu and Su, 2015). The nodes are the data points and the weighted edges are the similarities between the data points. Cells are defined as a cluster if they have a defined number of edges between them, forming a "clique".

A similarity matrix using Euclidean or other similarity measures is then computed. Using this similarity matrix, the k-nearest-neighbours (KNN) for each data point are listed. The number of nearest neighbours has to be supplied by the user. Edges between data points are assigned if they share at least one KNN. The weights of the edges are defined by a function of the number of nearest neighbours and their respective ranks. Identification of clusters is made by finding quasi-cliques associated with each node and merging them to unique clusters by the use of a greedy algorithm. A node induces a subgraph, which consists of all neighbour nodes and edges. For each node, a local degree is computed and a node is removed from the subgraph if its degree is lower thana given threshold which is proportional to the size of the clique. The threshold is supplied by the user and is typically set to 0.7. Next, the degrees between the nodes are recomputed and the process is repeated until no more nodes can be removed. A subgraph is assigned to a quasi-clique if it contains more than three nodes. To reduce redundancy, quasi-cliques that are completely included in other cliques are removed as well. Clusters are then identified by merging the quasi-cliques. For each pair, an overlapping rate is computed. If it exceeds a predefined threshold m the sub graphs are merged. Merging in different orders leads to different results, so pairs with larger sizes are prioritized.

SIMLR Most clustering methods rely on standard similarity metrics like Euclidean distances (Wang et al., 2017). SIMLR uses a weighted function of multiple kernels to compute a distance matrix. The assumptions is that the matrix has a block-diagonal structure, where the blocks represent the clusters c. The kernels are Gaussian kernels with a range of hyperparameters defining the variance of each kernel. The similarities are then used for data visualisation with tSNE or clustering using k-means on the latent space representations of the similarities.

CIDR Clustering through Imputation and Dimensionality Reduction (CIDR) takes the high dropout rate in scRNA-seq data into account (Lin et al., 2017). The method splits the squared Euclidean distance into three terms. These consist of one term in which both genes k for the cell pairs i an j are non-zero, a second term in which one gene is zero and a third where both are zero. The authors state that only the cases where one gene is zero have a strong influence on the distances and the subsequent dimension reduction and clustering. To reduce the dropout-induced zero inflation, the method imputes the third term by its expected value given the distribution of the dropouts. The algorithm works basically in five steps: (i) Find features that are dropout candidates. That is genes that show an expression level below a threshold T. (ii) Find the empirical dropout probability $\hat{P}(u)$ using the whole data set. (iii) Calculate the dissimilarity using Euclidean distances together with pairwise a imputation process. Features that fall below the threshold T are imputed using a weighting function. The weighting is based on the probability of being a dropout. (iv) Perform dimension reduction using PCA on the imputed distance matrix. (v) Perform hierarchical clustering using the first few PCs. The number of PCs can be determined by several methods. Here, we use an implemented variation of the scree method.

Seurat Seurat uses raw counts, where filtering is both done gene- and cell-wise. A user-specified threshold for the minimum number of expressed features per cell and the minimum number of gene-wise expression per cell. Scaling, log transformation and normalisation of the counts is done with a scale factor of 10000, a log2 transformation.... In the second gene-filtering step, low-variance genes are filtered out. Clustering is done using a PCA-latent space representation, which is used to compute the Jaccard distances. The Louvain algorithm then is used for clustering.

and a smart local moving algorithm (SLM). Here a resolution parameter defines the number of clusters.

Linnorm Linnorm is a normalisation and transformation method for count data (Yip et al., 2017). The main assumption is that a homogeneously expressed gene-set exists. Using this gene subset, and by ignoring zero counts, the normalisation and transformation parameters are calculated. After normalisation, the expression values should show both homoscedasticity and normality. Linnorm includes functions for subpopulation analysis by t-SNE or PCA dimension reduction and subsequent k-means or hierarchical clustering.

First, the values are scaled according to library size. Low count genes and genes that show high technical noise are filtered out. By default, genes showing a non-zero expression in at least 75 % of the cells are retained. Note that this threshold is set to maintain at least three non-zero cells per gene, in order to calculate the skewness of the gene distributions. By gradually increasing this threshold only genes which show a negative correlation between the mean and the standard deviation (SD) is assured. A locally weighted scatterplot smoothing (LOWESS) curve is fitted on the mean versus SD relationship. The SD is scaled and outliers based on the SD are removed. Next, genes that show a high skewness are filtered out. The data is then transformed using a modified log transformation.

TSCAN TSCAN uses a pseudo-time algorithm for cell ordering (Ji and Ji, 2015). PCA dimension reduction on the preprocessed gene expression data is performed. Preprocessing is done by log transformation and by adding a pseudo-count. Low expressed genes are filtered out based on the zero-proportion and their covariance. Clustering is done by model-based clustering. The travelling salesman problem (TSP) is then solved by a minimum spanning tree. The user can then define the start/end point through the available biological information and compute a pseudo-time ordering score.

ZINB-WaVE The method models the counts as an a zero-inflated negative binomial (ZINB) model that accounts for the zero-inflated, over-dispersed nature of scRNAseq count data(Risso et al., 2017). The model allows for the adjustment of gene- and cell-level confounders. Based on the preprocessing steps genes with less than five counts per feature are filtered out. Additionally, it is recommended to use only high variable genes.

2.3 Datasets

The datasets Kumar, Trapnell and Koh were downloaded from the *conquer* repository `http://imlspenticton.uzh.ch:3838/conquer`. The Zheng data was downloaded from the 10xChromium repository: `https://support.10xgenomics.com/single-cell-gene-expression/datasets`. For the Kumar, Trapnell and Koh data count-scale, length-scaled Transcripts Per kilobase Million (TPM) are used. These are on a count-scale and are independent from the average length of the transcript (Soneson et al., 2015). The Zheng data consists of UMI counts.

Kumar et al. (2014) The Kumar dataset consists of *Dgcr*8-knockout and V6.5 variotypes from mouse embryonic stem cells (mESCs). Cells were cultured on a serum with a Leukaemia Inhibitory Factor (LIF) or under Erk and GSK3 signalling inhibition (2Li). The authors investigated the expression of pluripotency factors and their involvement in the heterogeneity of pluripotent stem cells. Sample preparation and whole transcriptome amplification was done using a Fluidigm C1 system and following a SMARTer protocol. Sequencing was done using the Illumina system with paired-end reads. Sequencing depth was 1 million reads per cell.

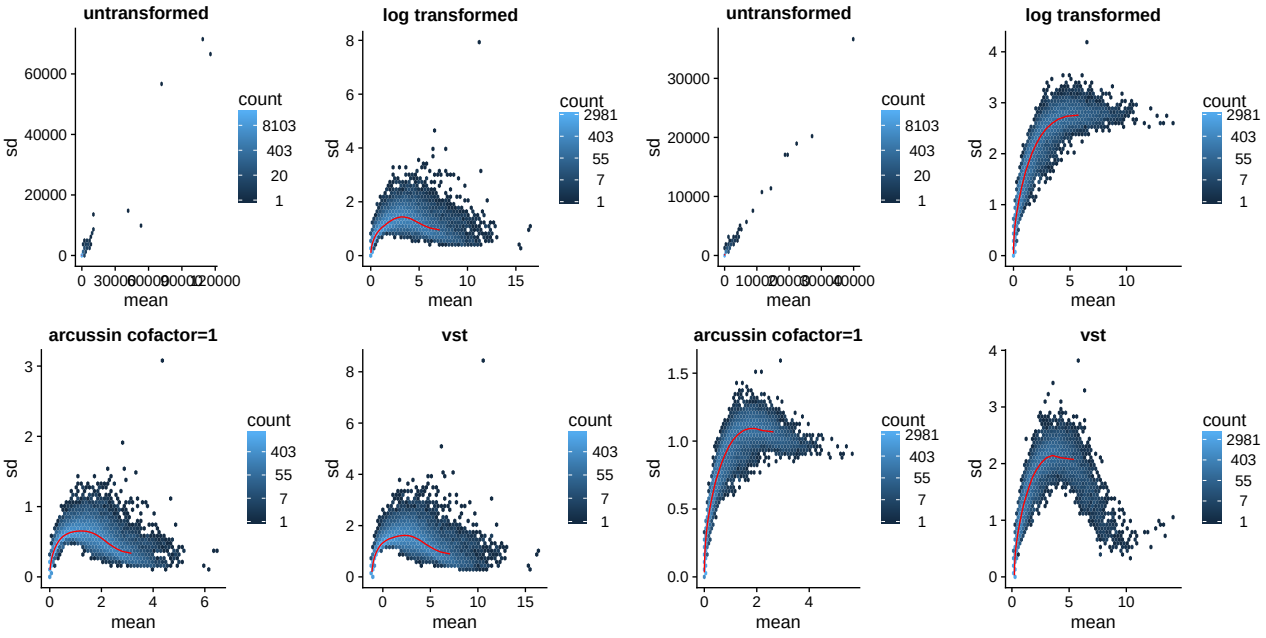
Trapnell et al. (2014) Trapnell et al. (2014) used human skeletal muscle myoblast cells to investigate temporal differentiation. Cells were expanded under high–mitogen conditions. Differentiation was induced by switching to low-serum medium. Cells were captured before switching to low-serum medium (T0), after 24 h (T24) and 48h (T48). Between 49 and 77 cells were isolated at each time point and used for single mRNA-Seq library preparation by the use of a Fluidigm C1 system and following a SMARTer protocol. Libraries were sequenced with paired-end sequencing on a HiSeq 2500 (Illumina) platform. Sequencing depth was 4 million reads per library. The authors excluded libraries that contained fewer than 1 million reads.

Koh et al. (2016) H7 human embryonic stem cells (hESCs) were used to study human mesoderm development. Starting from undifferentiated stem cells, several differentiation stages, sorted by time point and further refined by fluorescence-activated cell sorting (FACS) were isolated. Finally, nine different cell lines were obtained: undifferentiated H7 hESCs (H7hESC), anterior primitive streak populations (APS), mid primitive streak populations (MPS), lateral mesoderm (D2LtM), FACS-purified DLL1+ paraxial mesoderm populations (DLL1pPXM), early somite progenitor populations (ESMT), PDGFR+ sclerotome populations (Sclrtm) and two different dermomyotome populations (D5CntrlDRmmtm). In total, ten different cell types were then sequenced on a Fluidigm C1 system and following a SMARTer protocol. Libraries were sequenced through paired-end sequencing on a HiSeq 2500 (Illumina) platform. Sequencing depth was 1 – 2 million reads per cell.

Zheng et al. (2017) FACS-purified fresh peripheral blood mononuclear cells (PBMCs) were used to assess the performance of the 10xChromium system. Sample preparation and library construction were done with a 10xChromium system. The libraries were sequenced using an Illumina system. For this study, the datasets for CD19+B, CD8+CD45RA+ naive cytotoxic, CD14+ monocytes and CD4+/CD25+ regulatory T cells were used to construct an artificial population. From each library, 200 cells were sampled before beeing merged to obtain a single expression matrix.

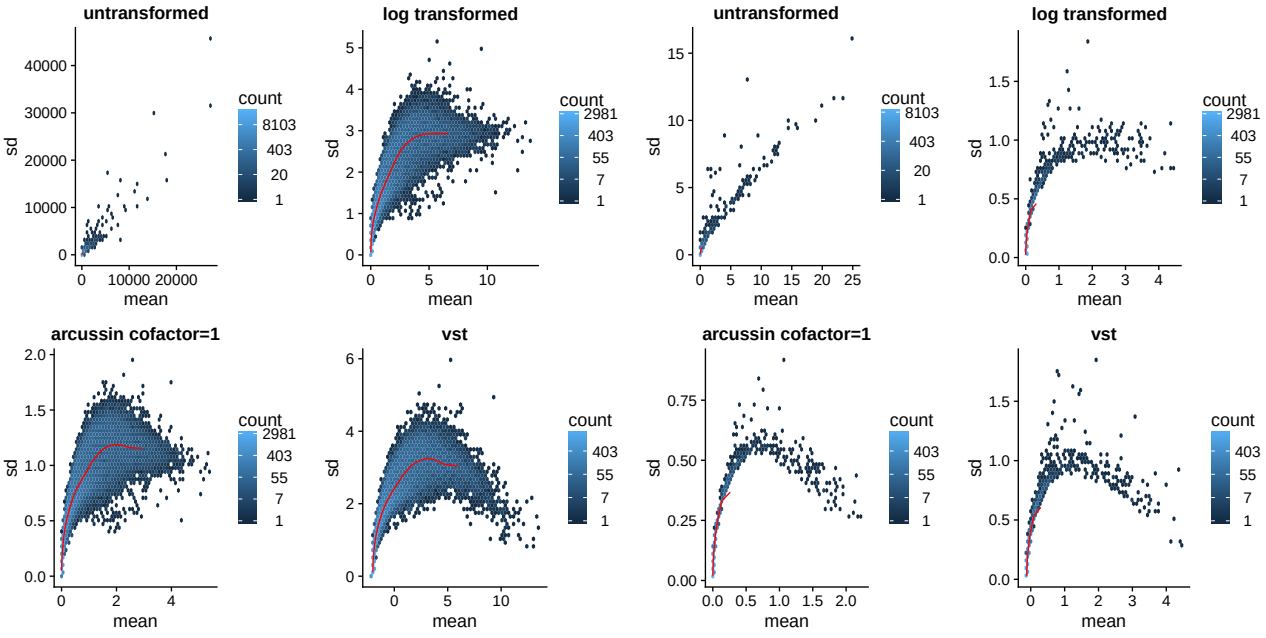
Simulated datasets Using the Splatter package, expression data were simulated (Oshlack et al., 2017). Parameters for the simulation were estimated from a subpopulation of the Kumar dataset. Embryonic stem cell variotypes V6.5 with signalling inhibition and LIF were used for the estimation. SimDataKumar consists of 500 cells with four subpopulations. The fractions per subpopulations were 0.1, 0.15, 0.5 and 0.25 of the total cell population. The probability that a gene is differentially

expressed is 0.05, 0.1, 0.2 and 0.4 in the four different groups. Similarly, SimDataKumar2 consists of four subgroups with fractions of 0.2, 0.15, 0.4 and 0.25 of a total of 500 cells. The fractions of differentially-expressed genes were lower, with a probability of 0.01, 0.05, 0.05 and 0.08. The spike-in RNA is excluded before the parameter estimation.



(a) Kumar dataset

(b) Koh dataset



(c) Trapnell dataset

(d) Zheng dataset

Figure 1: Shown is the genewise standart deviation versus the mean for the datasets Kumar (a), Koh(b), Trapnell(c) and Zheng(d). PLotted are the untransformed and the log, arcus sinus and VST-transformed data.

2.4 Data transformation and normalisation

Data transformation RNA-seq data may suffer from heteroscedasticity and skewness (Zwiener et al., 2014). Genes with higher mean have on average a higher variance across cells leading to unequal variances between different genes. To handle this property different transformation were considered. Namely, a binary logarithmic transformation with a pseudo-count of one, arcus sinus transformations and a variance-stabilising transformation (VST) from the DESeq package (Huber et al., 2002). Log transformations will have an impact on extreme values. However, they will not address the problem of heteroscedasticity. Arcus sinus transformation should deal with extreme values and equalise the variances. After transformation, the mean and the variances should be independent. VST addresses the problem of extreme values and unequal variances across genes. After such transformation, the mean and the variances of the genes should be independent. For the study, a binary logarithmic transformation plus a pseudo-count of one is used. The mean-SD dependence for different transformations is shown in Figure 1.

Filtering and normalisation The quality control of the data sets follows Lun et al. (2016b). In the first step, genes that are not expressed in any cell (systematic–zeros) are removed in order to reduce the size of the expression matrix. To find potential outliers, PCA can be used on the phenotype characteristic of each cell can be used (Figure 7, 8, 9, 10; a). Cells were filtered based on the library size and the total number of genes. Cells with log10 library sizes that are more than three median absolute deviations (MADs) below the median log-library size were filtered out (Figure 2, 3, 4 and 5). The same filter was used with respect to the total number of genes per cell. For the Kumar and the Zheng dataset, ERCCs and MT counts were available. Cells with large proportions of ERCC or mitochondrial RNA are seen as low-quality cells. In the Kumar dataset, cells with an ERCC proportion above three MADs are as well removed. The same filter was used for mitochondrial RNA in the Zheng data.

The metadata for the Trapnell dataset contained information about the cell quality. In this dataset, cells that were marked as debris and any single libraries consisting of more than one cell were filtered out. After filtering, 531 cells in the Koh dataset, 246 in the Kumar dataset and 222 in the Trapnell dataset were retained. The filtering was less strict in the Koh dataset compared to the original analysis where they retained 498 cells.

Low-abundance genes influence the mean-variance trend. Here low-abundance genes are filtered by their average counts (see Figures 7, 8, 9 and 10; d). For the Kumar, Trapnell, simDataKumar and Zheng data genes with average counts less than one are removed. The Zheng data set had a shallower sequencing depth. A different filter is used, and features which are not expressed in at least two cells are excluded. To find batch effects a linear model regressing the PC values against the total features was used (Lun et al., 2016b).

Another examination of the technical variation was done using the marginal variances(Lun et al., 2016b). For that, a linear model with the expression values per gene as response variables and a chosen explanatory variable is fitted. The correlation coefficient can then be seen as the marginal explained variance for the explanatory variables.

A wide variety of normalisation methods exist based on bulk RNA methods. These methods are usually not designed for dealing with the zero-inflated nature of scRNA-seq dataLun et al. (2016a). Methods for normalisation of scRNA-seq data are based on spike-ins or RNA counts. Spike-in RNA is added before the library preparation. Any changes in the spike-in coverage are assumed to be due to technical factors. The normalisation is done by scaling the counts to level the spike-in. However, this approach is not feasible as none or only a limited number of spike-in counts were present in the datasets.

Here, normalisation through pooled cells is used, where the problem of excess zero counts is reduced by the pooling of multiple cells Lun et al. (2016a). The normalisation procedure can briefly be described as follows: (i) Different pools of cells are defined. (ii) The expression values are summed across the cell pools. (iii) The cell pool is normalised against an average of the summed expression values. (iv) This step is repeated several times to construct a linear system. The summed count size is then used to estimate the corrected size factor. The size factors for the pooled cells are then deconvoluted" into cell-based factors.

2.5 Optimal number of clusters

Methods to determine the optimal number of clusters are subjective methods as elbow or silhouette plots. In the Elbow plots, the within-cluster sum of square is plotted against a range of clusters. The silhouette plot is a standardized measure of distances between each point inside and outside of the respective cluster. Less subjective is the gap statistic. Here the log within sum of squares is compared to its expectation. The null distribution is expected to be uniformly distributed, it is not clear if this is correct for high dimensional data (Tibshirani et al., 2001). Other possible methods are the calinsky criterion, hierarchical clustering.... Here the optimal number of clusters is determined by Elbow plots,

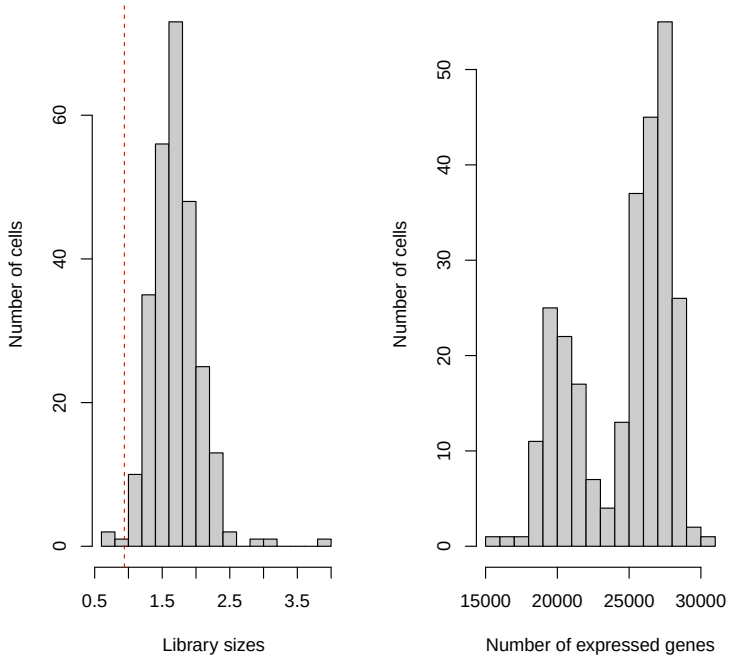


Figure 2: Histogram of Kumar 2014.

clustering is based on kmeans clustering and the within-cluster sum of square thereof. The elbow plots suggest three clusters for the Kumar dataset, 2 - 5 in the Trapnell data, 3 in in the Zhengmix data (see Figure **??**). The optimal number of clusters is unclear for the Koh data set. Minimization of within sum of squares was also done in the tSNE latent space with 30 dimensions. Here the optimal number of clusters are 3,3, 4 and 6 to 8 in the Kumar, Trapnell, Zheng and the Koh data sets, respectively.

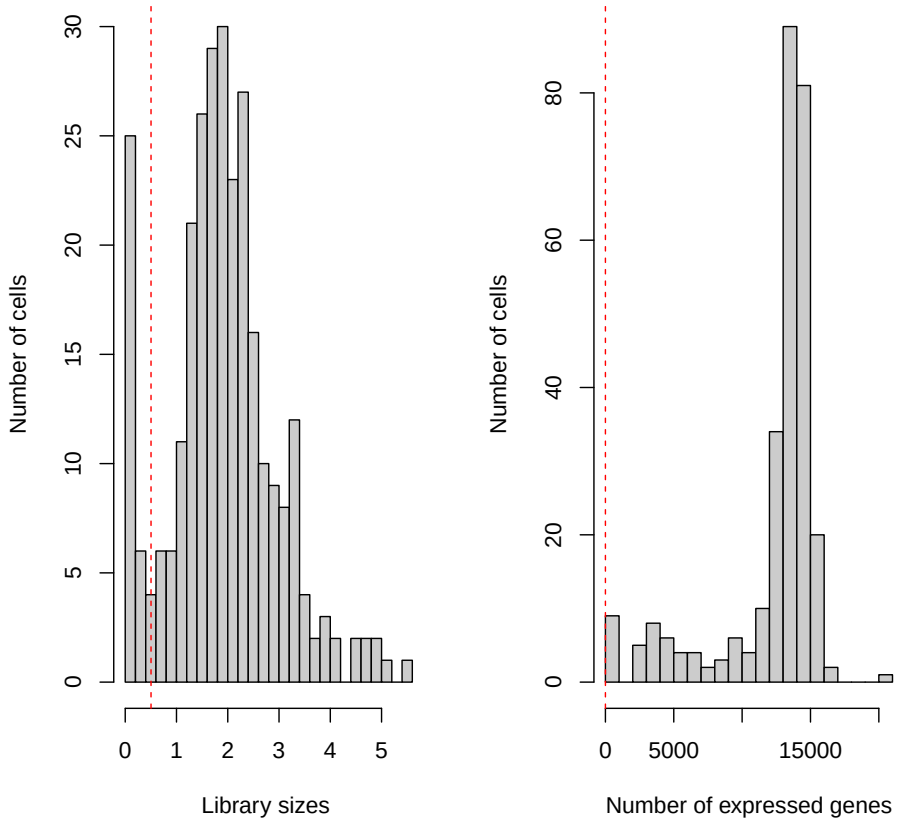


Figure 3: Histogram of Trapnell 2014.

3 Method

3.1 Evaluation of the clustering methods

The clustering methods were run with the annotaded number of clusters given by the authors or the truth. Additionally, under default mode and with a maximized ARI. The clustering methods were run under the default mode, with cell- and gene-wise filtered data sets and using smoothed data. Additionally, the methods were run under a range of number of clusters. The run parameters in the default mode were given either by the default setting of the packages or by examples in the different package vignettes. If the method was able to detect the number of subpopulations, the auto detection function was used to infer the number of cluster. For some methods(which) the number of clusters had to be provided. Then, the number of cluster given by the cell annotation provided by the authors were used. For the evaluation with filtered data sets filtering steps were excluded. As in the previous evaluation the inputs were counts, log transformed counts and normalized counts. Evaluation with non filtered data sets was done as described before, except that the filtering steps are now included in the methods.

Depending on the required inputs counts , log transformed counts or normalized counts from the filtered data sets were used. Any filtering steps done by the methods were excluded in this analysis.

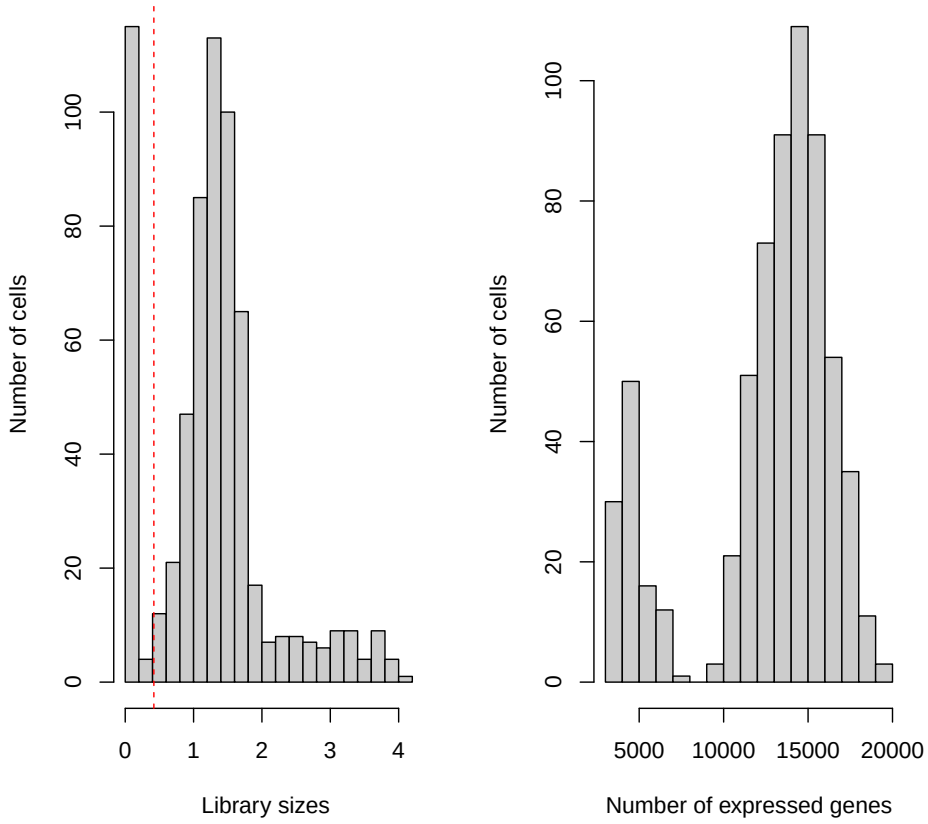


Figure 4: Histogram of Koh 2016.

An overview of the different method settings is given by table ... In a further analysis the clustering methods were tested for different values for the number of clusters k. Seurat and dbscan do not allow the setting of the number of cluster. Hence, the number of KNN in Seurat and the size of the neighborhood ϵ in dbscan was selected.

3.2 Parameter settings

For most of the methods the number of cluster is the most important parameter. Other important settings were the kNN, the number of latent space dimensions used for the clustering algorithms or the settings of the filtering and normalization steps. Here different parameter settings were used to find the optimal settings for each method. First, the methods were run under default mode. Many users will likely run the methods without any fine tuning of the parameters and it was seen important to provide results under this setting. Note that only the methods PCAreduce, SC3, Linnorm, RACEID, TSCAN are completely unsupervised and no parameters have to be provided. For SEURAT and dbscan the number PCs or the number off the kNN have to be defined. CIDR, RtSNEkmeans, SIMLR need the specification of the number of clusters k. For these three methods the parameters setting is equal to the settings described in the next section. Although its possible to run the methods unsupervised, for many of the methods a fine tuning of the parameters is highly recommended. Next, methods were run with a the set ground truth of numbers of sub populations. For many of the methods k is a input parameter. However, the methods dbscan and SEURAT allow the setting of k only indirect trough

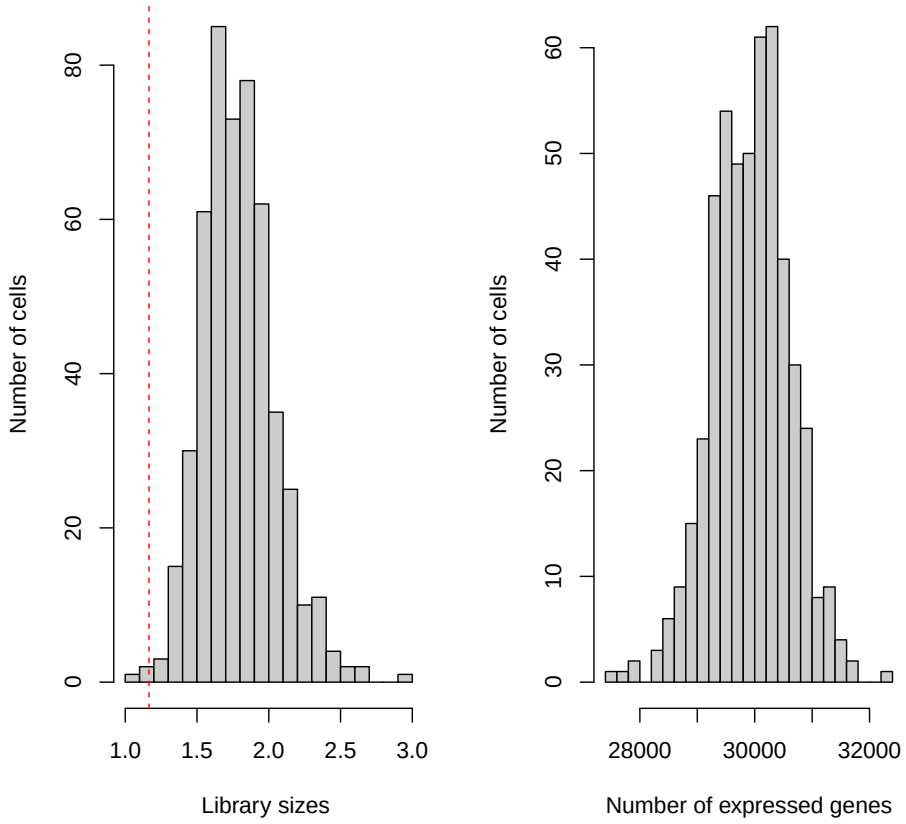


Figure 5: Histogram of simDataKumar.

kNN or a resolution parameter. To evaluate the methods for the best possible partitions of cells , given the ground truth and evaluated by the use of the ARI score the methods were run under a range of parameters. Whenever possible, the range was chosen such that a clear maximum for the ARI score could be found. A full listing of the parameter settings for each method and run mode is provided in table .... Next, a brief overview for the chosen parameter setting and the rationale behind it is given.

RtSNEkmeans To reduce run time the Barnes-Hut tSNE implementation is used. Perplexity was set to 30 for all data sets. We note that the value of the perplexity can give different tSNE representations, however here the default setting was chosen. tSNE is performed on the first 30 dimensions in the in the PCA latent space .

pcaReduce For PCAreduce the range of clusters cannot be specified, instead the number of dimension q in the PCA latent space are to be specified. The results are $q-1$ different clustering solutions, with $k-2$ clusters. For all data sets 30 dimensions were chosen and evaluation was based on the respective number of clusters in the subsequent analysis. The method is based on kmeans clustering and has to be run several times for stable results. Here 50 samples were chosen. Merging of clusters was done by the default sampling proportional to the joint probabilities.

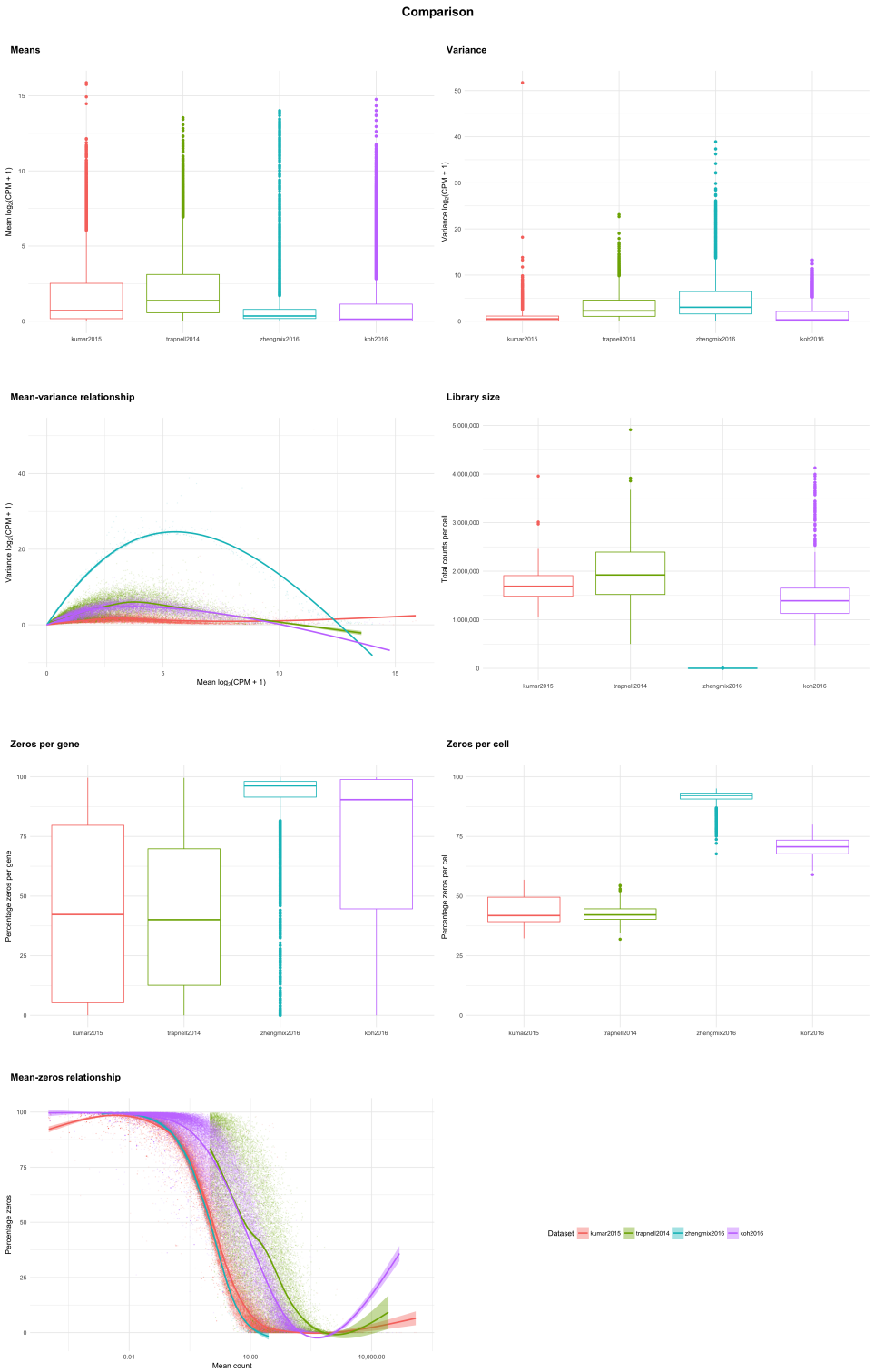


Figure 6: Comparison between the data sets. Based on compare function of Splatter.

SC3 A gene filtering step is implemented in the method. Filtering steps were included when run with the unfiltered data sets, otherwise the filtering step is excluded. Based on the dropout distribution genes below and above the 10th and 90th percentile are filtered out before clustering. This seems reasonable for the Kumar, simDataKumar and Koh data sets. However, for the Koh and Zhengmix

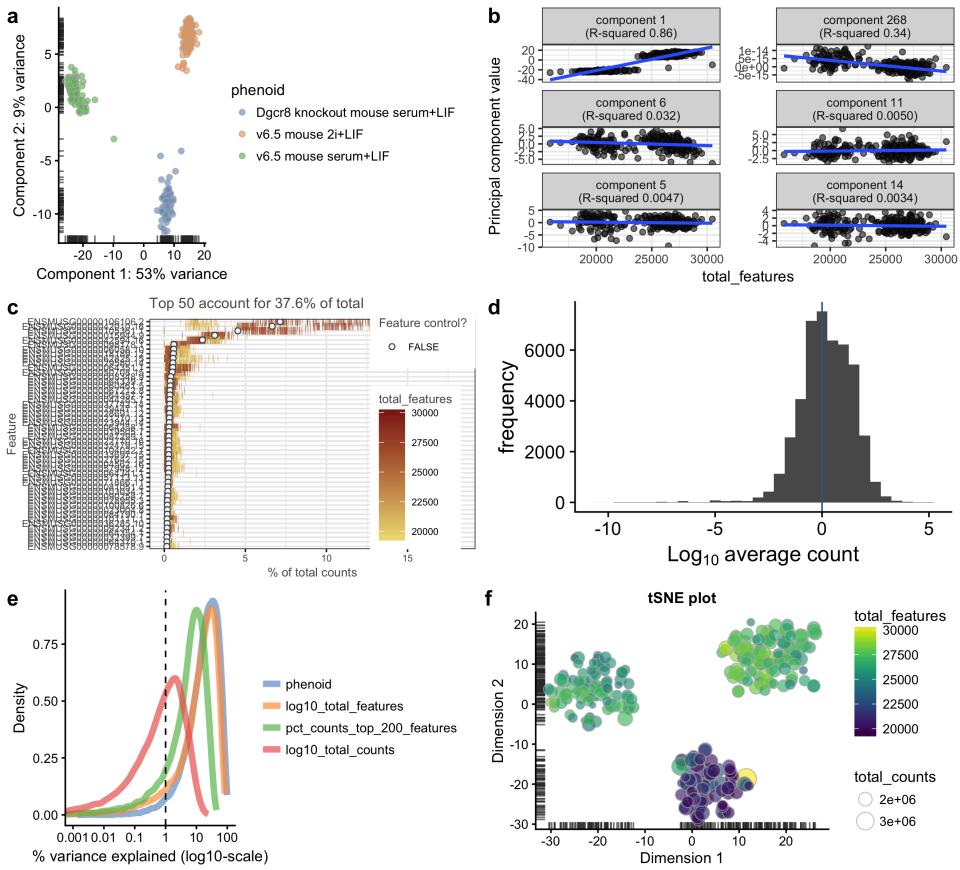


Figure 7: QC summary of Kumar 2015.

data set the upper threshold is set to the 99th percentile. SC3 can run under semi- or unsupervised mode. For the default mode a range of cluster is given, and the number of sub populations is automatically inferred by the method. Otherwise the number of annotated number of sub population is provided.

SNNCliq The connectivity of the quasi-cliques was set to the default value 0.7. Like wise the merging threshold parameter was set to the default of 0.5. The method was run with normalized, filtered data and the number of clusters was set to a range from 3 to 10 in all data sets. SNNclique works on different distance metrics, here the default euclidean distances are used.

SIMLR The tuning parameter k was set to the default value of 10 on all runs. The parameter number of clusters c is set accordingly to the run mode.

Seurat Implemented in the method are normalization and a gene filtering steps. Filtering criteria are the minimal number of gene expression in the cells and the number of total features per cell. The default setting is zero for both parameters, removing the filtering step. This setting is also chosen when the method is run on the filtered data sets. For the unfiltered data genes which are expressed in less than 5 cells in the Kumar , Trapnell, Koh and simDataKumar were filtered out. For the Zhengmix data the threshold is set to one, according to the filtering used in the QC and normalization steps. The default log normalization is used, currently the only option. Scale factor for cell-level normalization was set to 10000. As a default no explanatory variables were chosen to be regressed out. The experimental batch would be a natural choice as a covariate, but as the sub populations and

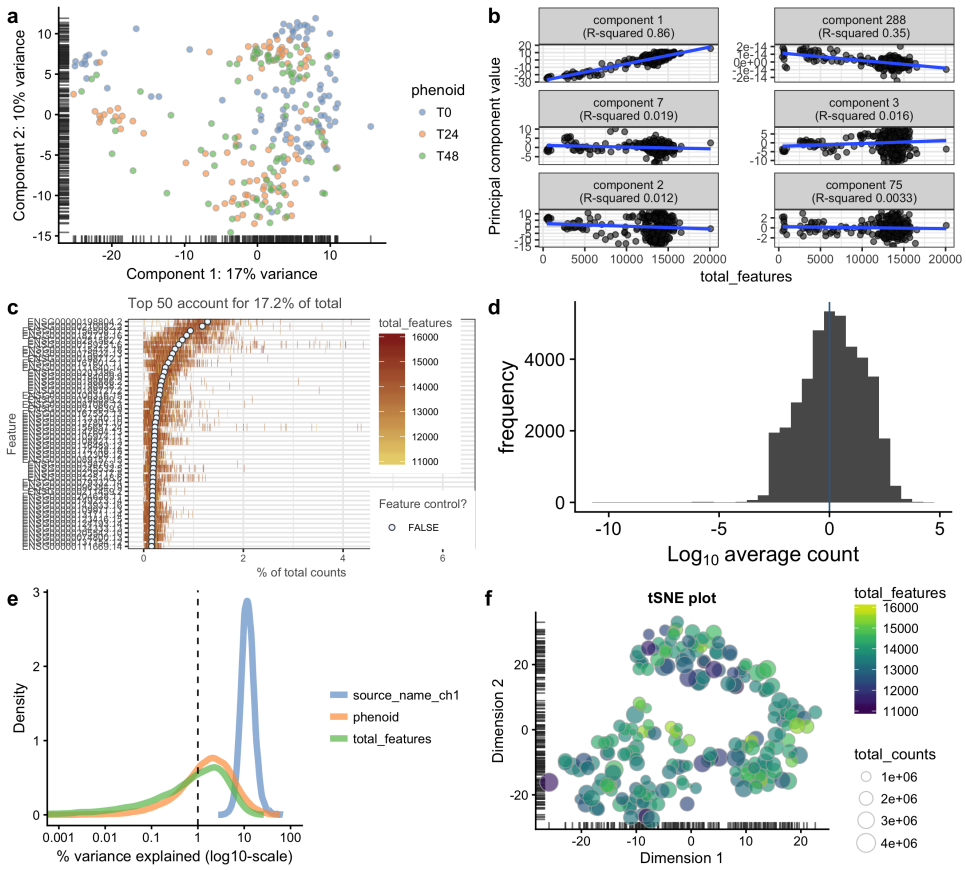


Figure 8: QC summary of Trapnell 2014.

the batches are identical for the Kumar and Trapnell data it was chosen not include it. The clustering parameters to be defined were a resolution parameter and the number of PCs for the clustering. The resolution parameter was set to the default value of 0.8 . The number of PC was determined by the methods recommended by the authors. Using scree plots and a jackknife permutation test (more exact) to determine the number of principal components. 9, 12, 10 and 15 principal components were used for the Kumar, Trapnell, Zheng and Koh dataset, respectively. 10 percent of cells were used for the number of neighbors in the k-nearest neighbor algorithm. A range of kNN fractions was used to find the the optimal value for the kNN parameter. A range from 0.5 % to 40 % of the total number of cells is used to infer the optimal number for the kNN parameter.

dbscan To choose appropriate parameters for the size of neighborhood epsilon and the minimum number of points in neighborhood the k-nearest neighbor distance is used. As a initial number of neighbors 10 percent of cells is used. then for each point the k-NN distance is computed and plot by increasing order. The chosen values for the distances are 280, 410, 35 and 380 for the Kumar, Trapnell, Zheng and Koh data sets, respectively. The default value for the minimum number of points is 5. For each of the dataset a range of epsilon around the theoretical optimum was chosen to optimize the clustering results.

CIDR CIDR uses three parameter settings; the number of clusters, the number of PCs (nPCs) and the method for hierarchical clustering. By default Ward distances are used in the hierarchical clustering. CIDR is able to infer the number of from 2 to n clusters. By default n is set to $nPC*2+2$. When not run in the default mode we choose the nPC according to a variation of the scree-plot and

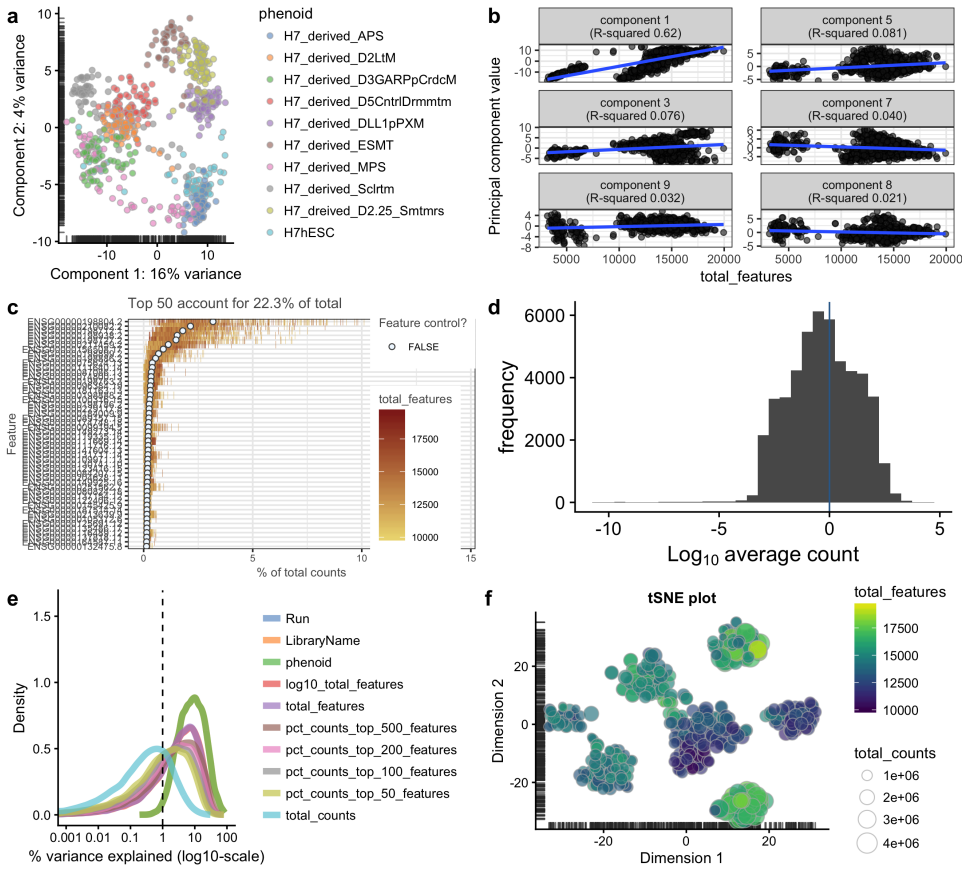


Figure 9: QC summary of Koh 2016.

set the number of cluster accordingly to the respective data set. The number of used PCs for the data sets Kumar, Trapnell, Koh , Zhengmix and simDataKumar are 5, 10, 8, 8 and 3 , respectively.

TSCAN The expression matrix is by the default log2 + 1 transformed. In all settings this transformation is used. Next, in a filtering step a gene wise threshold for the minimum expression value and a lower bound for the minimum retained fraction of genes that show expression greater than the threshold. As default the minimum expression is set 1 for the transformed counts, and the minimum fraction is 0.5. The minimum fraction had is set to 0.1 for Zhengmix due to the low sequencing depth of this dataset. For all run methods PCA is performed by default. By default the method infers clusters from a range of 2 to 9 clusters. If run semi supervised the respective number of clusters is given. By default "ellipsoidal, varying volume, shape, and orientation" is used for the model.

Linnorm The filtering thresholds are set to the default in all runs, except the Zhengmix data. For this data set the minimum non-zero expression is to a proportion of 0.1. In the default unsupervised mode the tSNE clustering is performed with k from 2 to 20. In the other runs the respective k per dataset is supplied.

3.3 Evaluation metrics

One evaluation criteria was the Hubert - Arabje Adjusted Rand Index (ARI) for comparing two partitions. The measure is adjusted for chance and 0 if there's no agreement between pairs and 1 if there is full agreement between pairs. The other criteria is the F1 score. It is the weighted average

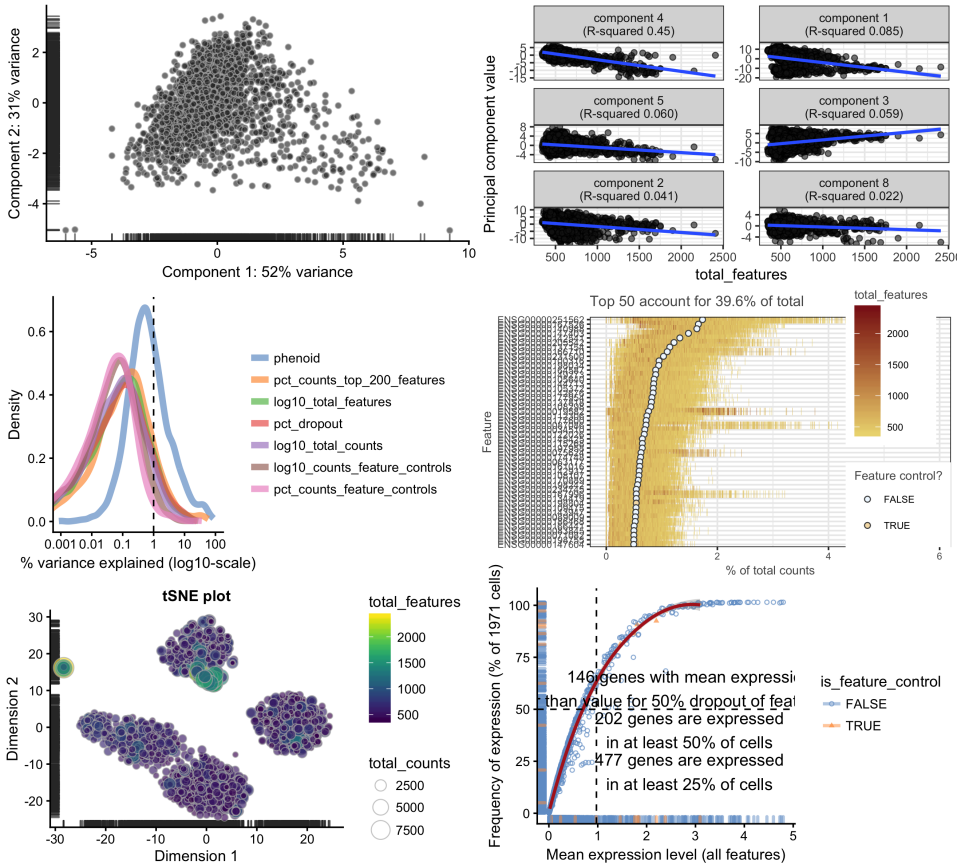


Figure 10: QC summary of Zheng 2016.

mean between precision and recall. With weights defined by the inverse of the precision and recall. F1 scores can take on values between 0 and 1.The predicted clusters and the "ground truth" were match by the Hungarian algorithm. Some of the clustering methods are unsupervised and the partitions does not need to have the same sizes (non-bipartite). This causes problems with Hungarian algorithm. As a solution the assignment matrix is augmented with dummy columns with the maximum matrix value as its entries.

```
## Error in `align<-.xtable`(`*tmp*`, value = switch(1 + is.null(align), :  "align" must
have length equal to 5 ( ncol(x) + 1 )
```

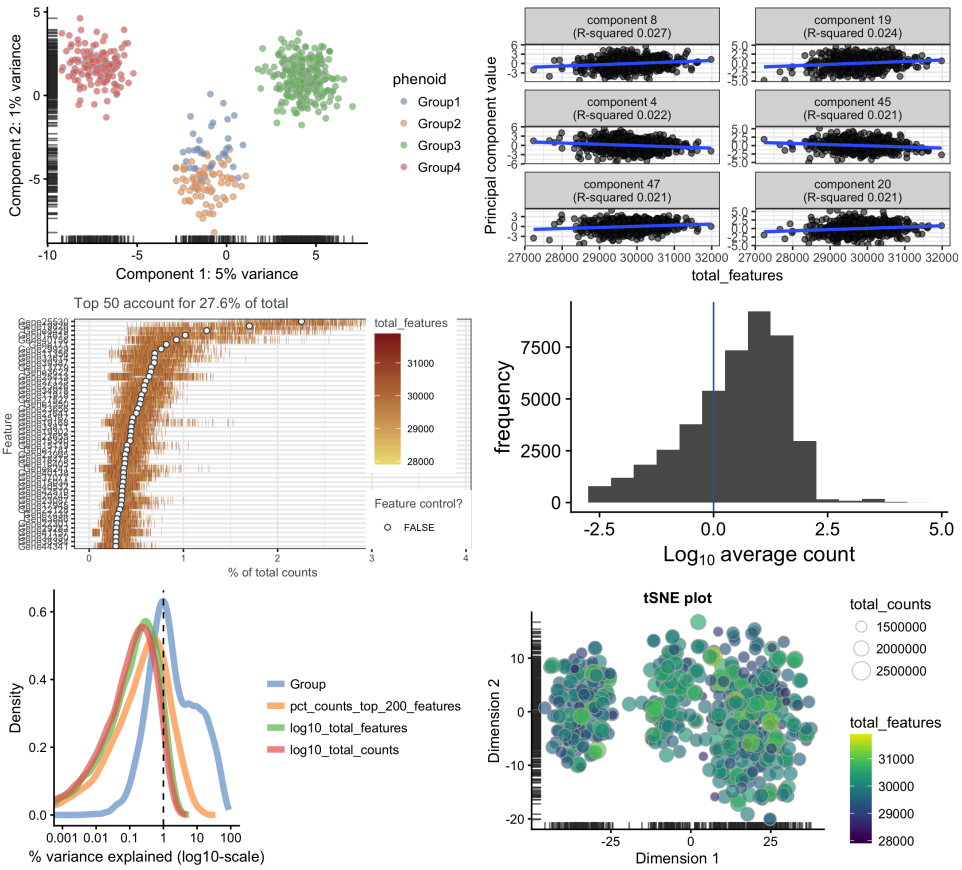


Figure 11: QC summary of simDataKumar.

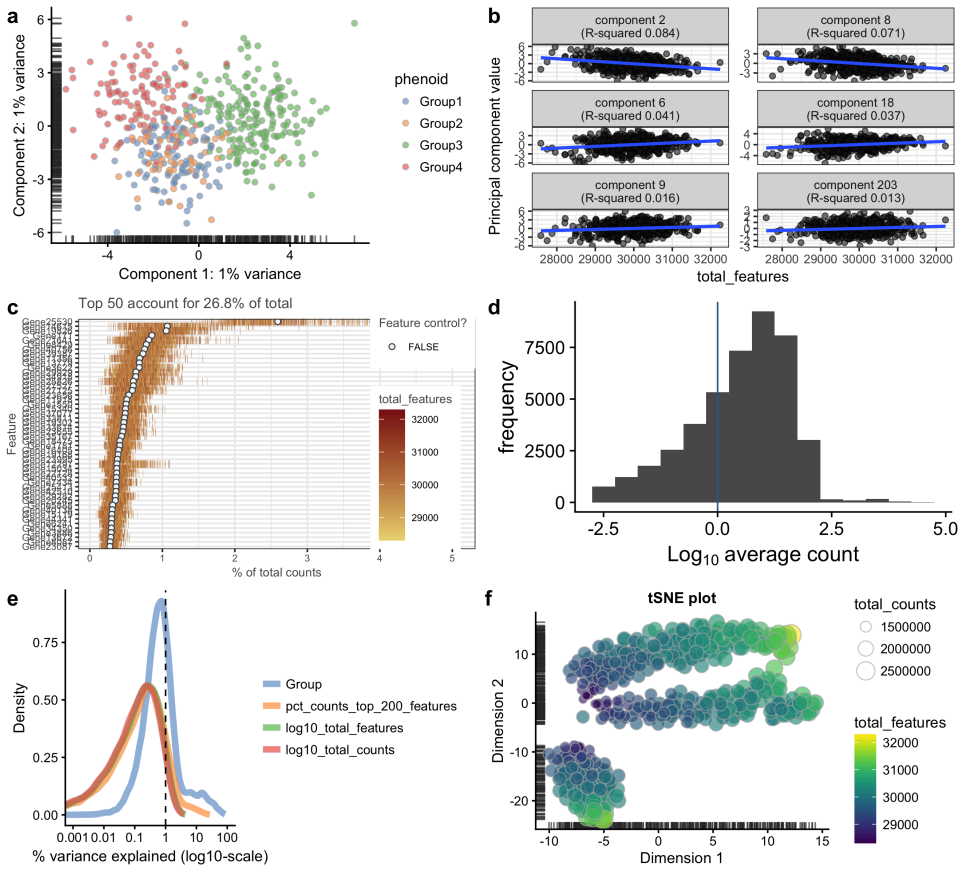


Figure 12: QC summary of simDataKumar2.

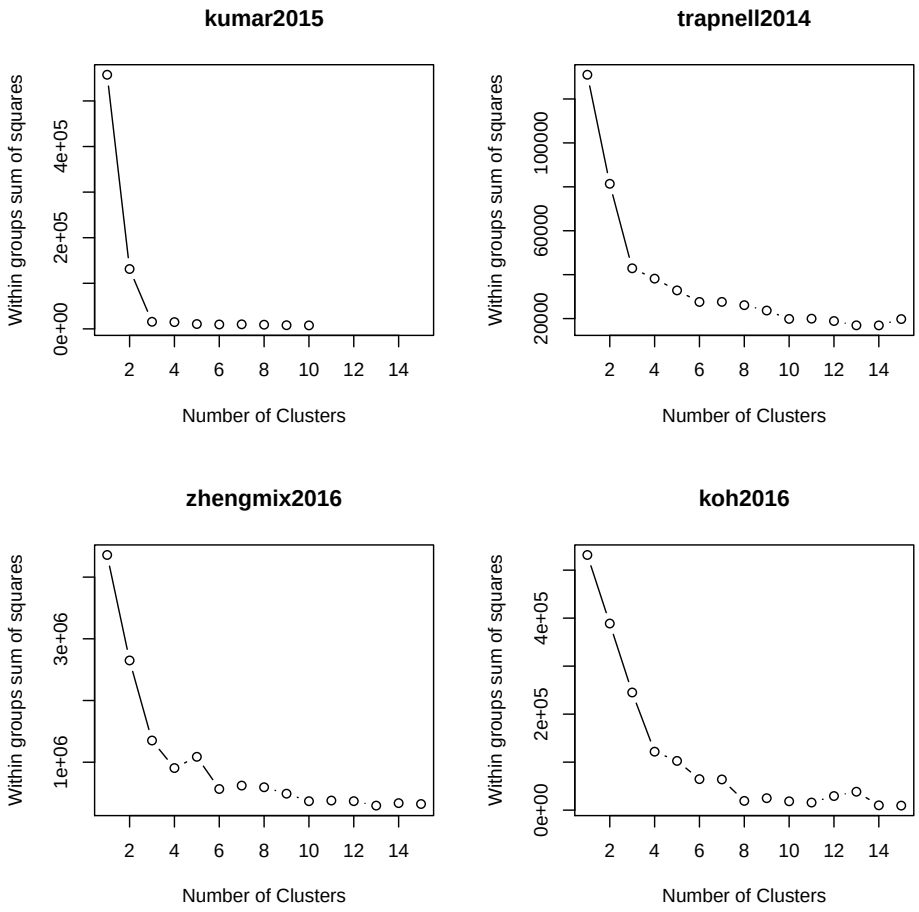


Figure 13: Optimal number of clusters by minimizing within sum of squares based on the latent space of tSNE (30 dimensions)

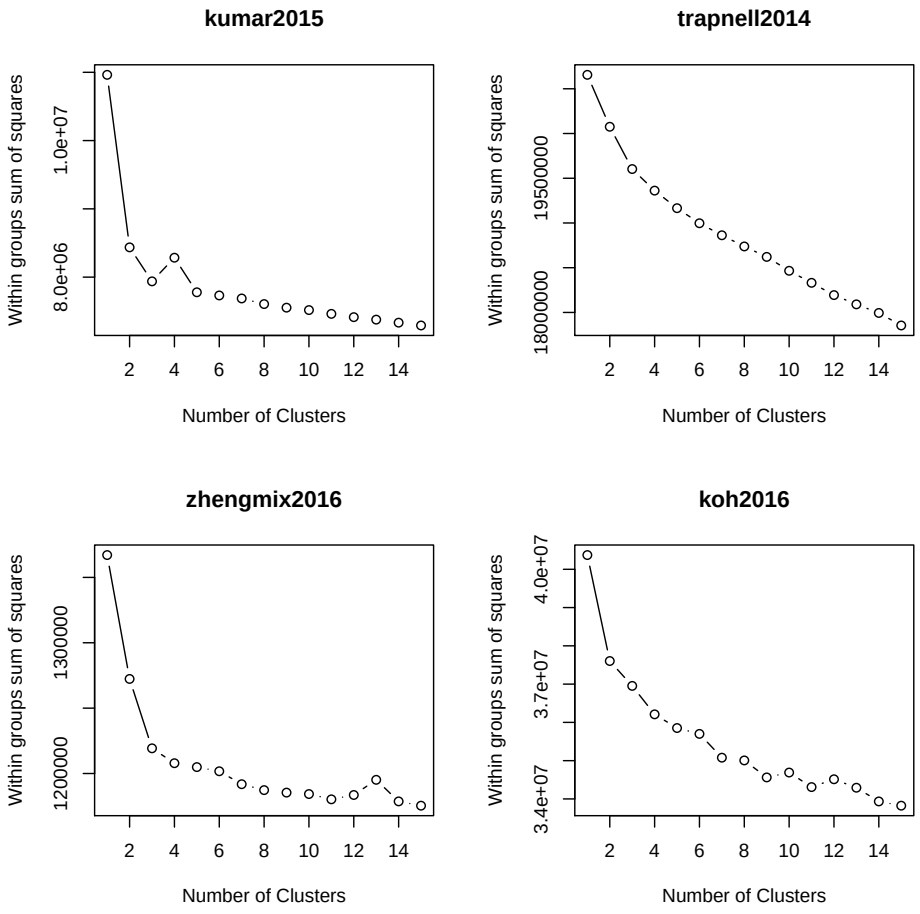


Figure 14: Optimal number of clusters by within sum of squares based on the full dimensions

Method	Description	dimension reduction	clustering	zero inflation	normalization	unsupervised
tSNEkmeans	tSNE dimension reduction and kmeans clustering	tSNE	kmeans	no	no	no
pcaReduce	PCA dimension reduction and kmeans clustering through an iterative process. Step wise merging of cluster by joint probabilities and reducing the number of dimension by PC with lowest variance	PCA	kmeans, hierarchical clustering	no	no	
SC3	PCA dimension reduction or Laplacian graph. Kmeans clustering on different dimensions. Hierarchical clustering on consensus matrix obtained by kmeans.	PCA	repeated kmeans, hierarchical clustering on similarity matrix of kmeans results	no	no	yes
SNN-cliq	Shared nearest neighbor graph based on similarities. Clustering through forming of cliques and subsequent merging.	graph based	merging of cliques	no	no	
dbscan	Density based clustering	none	density based clustering	no	no	yes
SIMLR		tSNE	kmeans	yes	no	yes
CIDR	PCA dimension reduction based on zero imputed similarities. Hierarchical clustering on a number of PC determined by variation of scree method.	PCA on imputed distances	hierarchical clustering	yes	no	yes
Seurat v1.4	Nearest neighbor graph based on PCA latent space	HVG and PCA		no	yes	yes

	cellfiltering	genefiltering	normalization	autodetect	expressionvalues
tSNEkmeans	no	no	no	no	normcounts
pcaReduce	no	no	no	no	normcounts
SC3	no	(yes)	no	yes	normcounts
SNNCliq	no	no	no	no	normcounts
dbscan	no	no	no	no	normcounts
SIMLR	no	no	(yes)	no	normcounts
CIDR	no	no	no	yes	normcounts
Seurat	(yes)	yes	yes	no	counts
TSCAN	no	yes	yes	no	counts
ZINBWaVEkmeans	no	no	yes	no	counts
RACEID	yes	yes	yes	yes	counts
Linnorm	no	(yes)	yes	no	counts

Table 1: Overview of filtering and normalization steps by method

4 Results

Transformation and normalisation In Kumar similar amount of the variance is explained by the total number of genes, the proportion of ERCC and the phenotype. This indicates that the data set is heavily influenced by batch effects. The same holds in the Trapnell data, but on a lower scale. Variance in Koh data is primarily influenced by the phenotype and to a lesser extent by the total number of genes and the top 200 features. Zheng is primarily dominated by the biological variation with the other explanatory factors contribution only marginally to the total variances. (Different methods exist to normalize RNA-seq data like TMM normalization, DEseq normalization and by library size. However, none of these methods are designed to deal with the zero-inflated nature of scRNA-seq dataLun et al. (2016a). Another often used approach for scRAN-seq data is the normalization by spike-inn. This approach is not feasible as no or only a limited number of spike-inn counts were present in the data sets.) The data still shows heteroscedasticity. There are only subtle differences between the arcus sinus and log transformations in all the datasets. The Compared with the Arcus sinus transformation.

Clustering The annotated number of clusters in the Kumar, Trapnell and Koh data sets are 3, 3 and 10, respectively. In this data sets the true cell population is given by the authors annotation and it was chosen as the ground truth. For the simulated data set simDataKumar the number of sub populations are set to 4. The Zheng dataset contained 4 merged cell sub populations. For comparison the within sum of squares (WSS) on a range of kmeans clustering was computed. Similar results were obtained, with exception of the Koh data where the elbow plots suggest 8 clusters.

For the methods cidr, pcaReduce, tSNE and kmeans, SC3 , TSCAN , linnorm and SIMLR a range of number of clusters were used. For Seurat a range in k-NN and for dbscan different neighborhood sizes were picked. The Adjusted Rand Index (ARI) is used as a evaluation metric, with the ground truth set by the cell annotation or the given ground truth.

In the Kumar dataset most of the methods show a clear maximum ARI score with three clusters. An exception is pcaReduce where either 3 and 4 clusters achieved a high score. SC3 has its maximum with 3 clusters, but higher numbers of clusters reach as well relatively high scores (see Figure 24; a). (In SNNClique almost half of all the clustering labels are in disagreement with the annotated cell labeling.) CIDR, pcaReduce, tSNEkmeans show a maximum score with three clusters in the Trapnell dataset. Whereas for SIMLR and SC3 the optimal number of clusters is 4 and 2 , respectively . In the Zheng data CIDR, Linnorm, pcaReduce, simlr, tSNEkmeans and SC3 had a maximum score value with 4 clusters. SC3 reached similar high values for 4 to 7 clusters. In the Koh data pcaReduce, RtSNEkmeans, SC3, SIMLR, cidr and tscan have their maximum at 10, 12, 11 , 8 , 8 and 10 , respectively. Note that the the maximum is not so clear. Except for CIDR, which had a low overall score, all methods have high values on 8 to 12 clusters. In the simulated data set only CIDR and SIMLR had its maximum with 4 clusters, whereas Linnorm, pcaReduce, RtSNEkmeans and TSCAN have a maximum at 3.

Clustering with the annotated number of cluster, under default mode and with unfiltered data sets All methods were run with the number of cluster set to the annotation given by the authors or by the set ground truth. For graph based methods were the neighborhood size had to be define 10 percent of the total cells were used as the neighborhood size. Others parameters like the number of PC's in Seurat etc. were chosen according to recommendations given by the method manual. For this clustering the filtered dataset was used. The partitioning of the clusters were then evaluated using the ARI. Figure 19 shows the Linnorm and pcaReduce performed bad in the simple Kumar dataset. (why) No high scores were achieved in the relatively difficult Trapnell data, with SC3 , RtSNE and Linnorm as the best performing methods. Overall, pcaReduce had the lowest and SC3 the highest ARI scores in the filtered data sets. When run under the default method the methods performed similarly. An

exception is Linnorm which has higher scores in the Koh, Kumar and simulated data set. Figure 23 shows the differences in the ARI scores between the filtered and unfiltered data sets. Here Seurat, SIMLR and TSCAN perform better in the Koh data. Better scores were also achieved in the Trapnell data by SIMLR, RtSNEkmeans Linnorm and CIDR. Surprisingly RtSNEkmeans had lower scores in the simulated data. As well as pcaReduce in the Kumar and Zhengmix and Linnorm in the Koh, Kumar and simulated data. SC3, Seurat, SIMLR, SIMLRlargescale, CIDR and TSCAN performed best with the filtered data sets. Linnorm, pcaReduce, raceID in the filtered dataset. RtSNEkmeans performed similarly under default settings and in the unfiltered data sets.

F1 scores for the Kumar dataset are shown in Figure **??**. Most methods are able to assign the correct partitions the three sub populations. However, Linnorm only detected two clusters. pcaReduce is able to correctly partition cluster number three but fails in the sub populations two and three. In the Trapnell data cell population 1 had the highest score. For population 2 and 3 the scores were lower for all methods. Koh had the highest number cluster, here nine cluster were annotated by the authors. The majority of the methods failed to assign cells to sub population 6 (which is that and why).

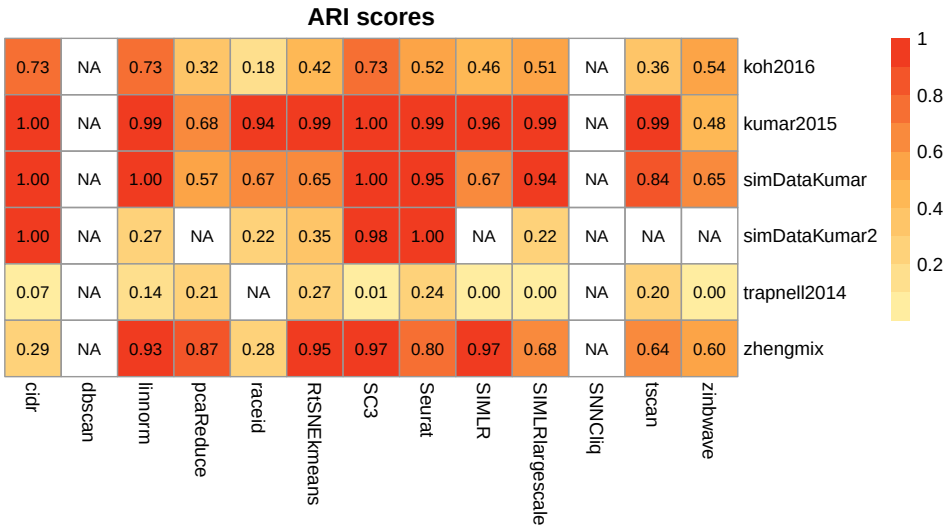


Figure 15: ARI scores with unfiltered data sets. The number of clusters is determined by the authors annotation.

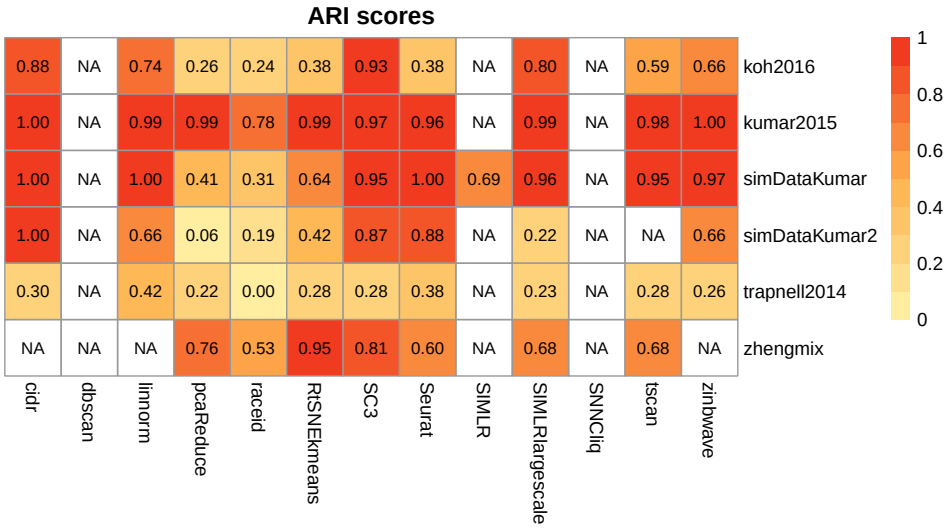


Figure 16: ARI scores with default setting.

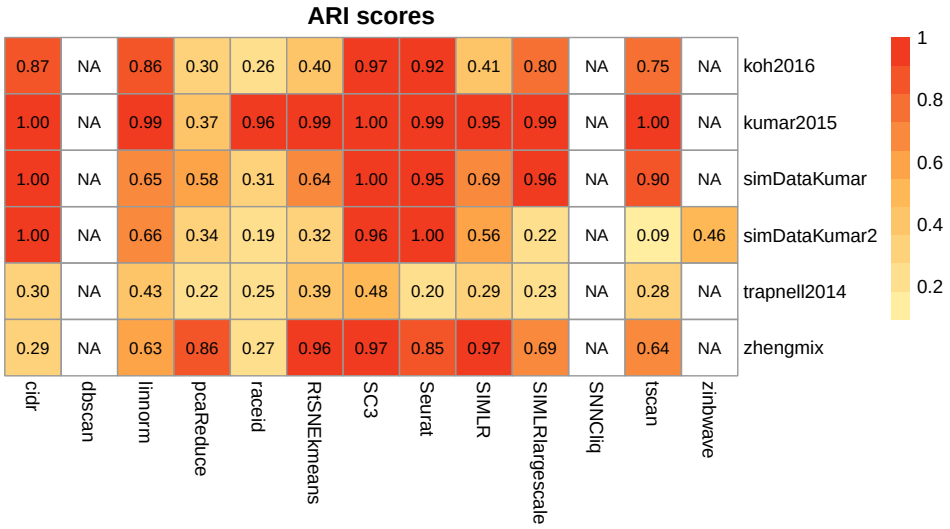


Figure 17: ARI scores with filtered data sets. The number of clusters is determined by the authors annotation.

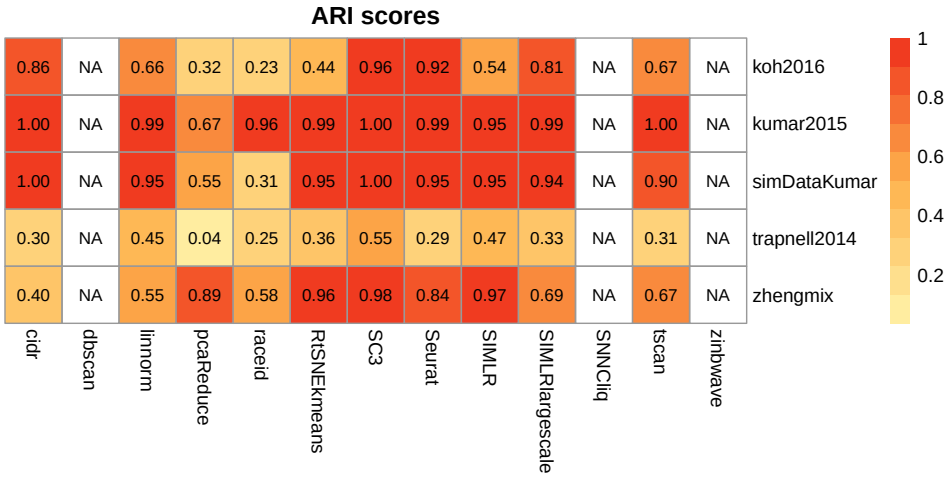


Figure 18: ARI scores with filtered data sets. A optimal k is chosen such that the ARI score is maximized.

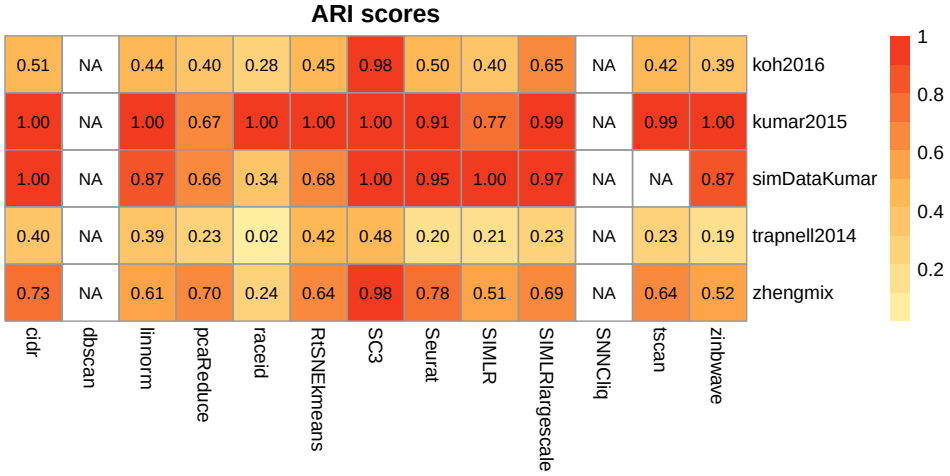


Figure 19: ARI scores with smoothed data sets.

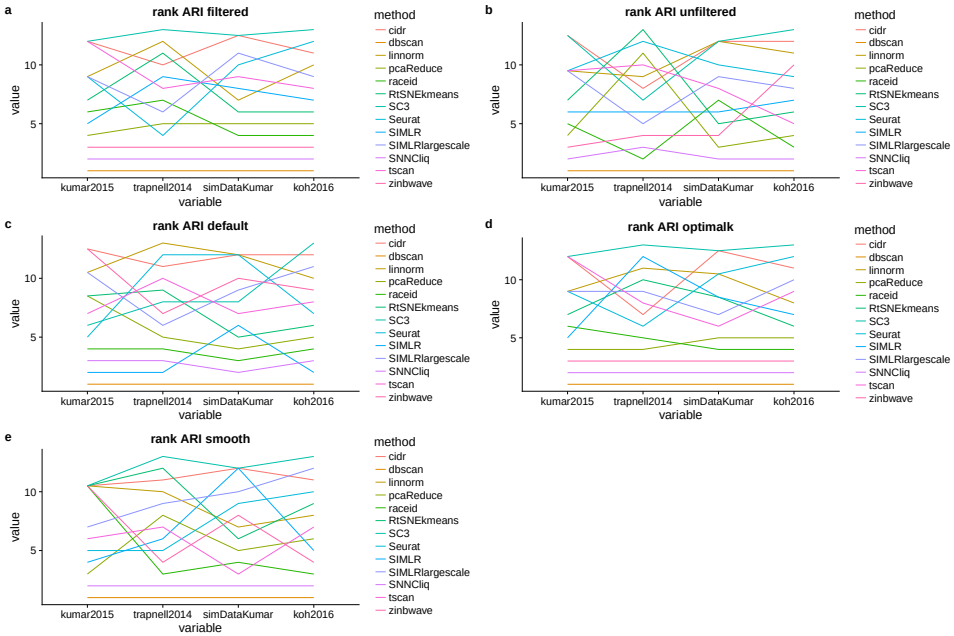


Figure 20: Ranks of ARI scores.

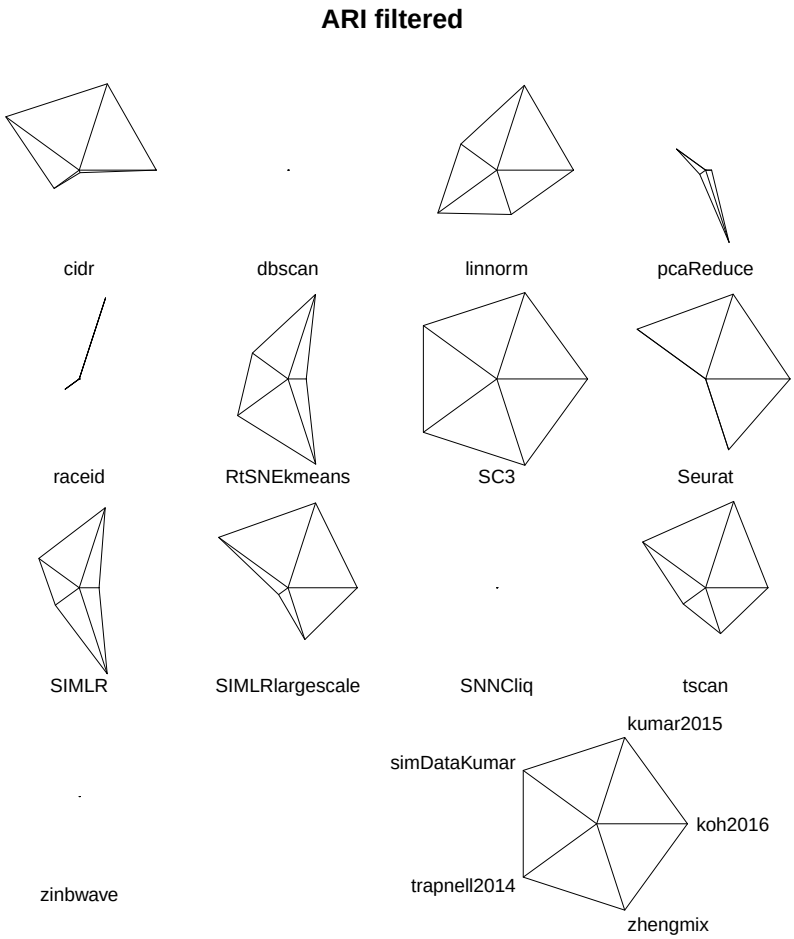


Figure 21: Starplot of ARI scores.

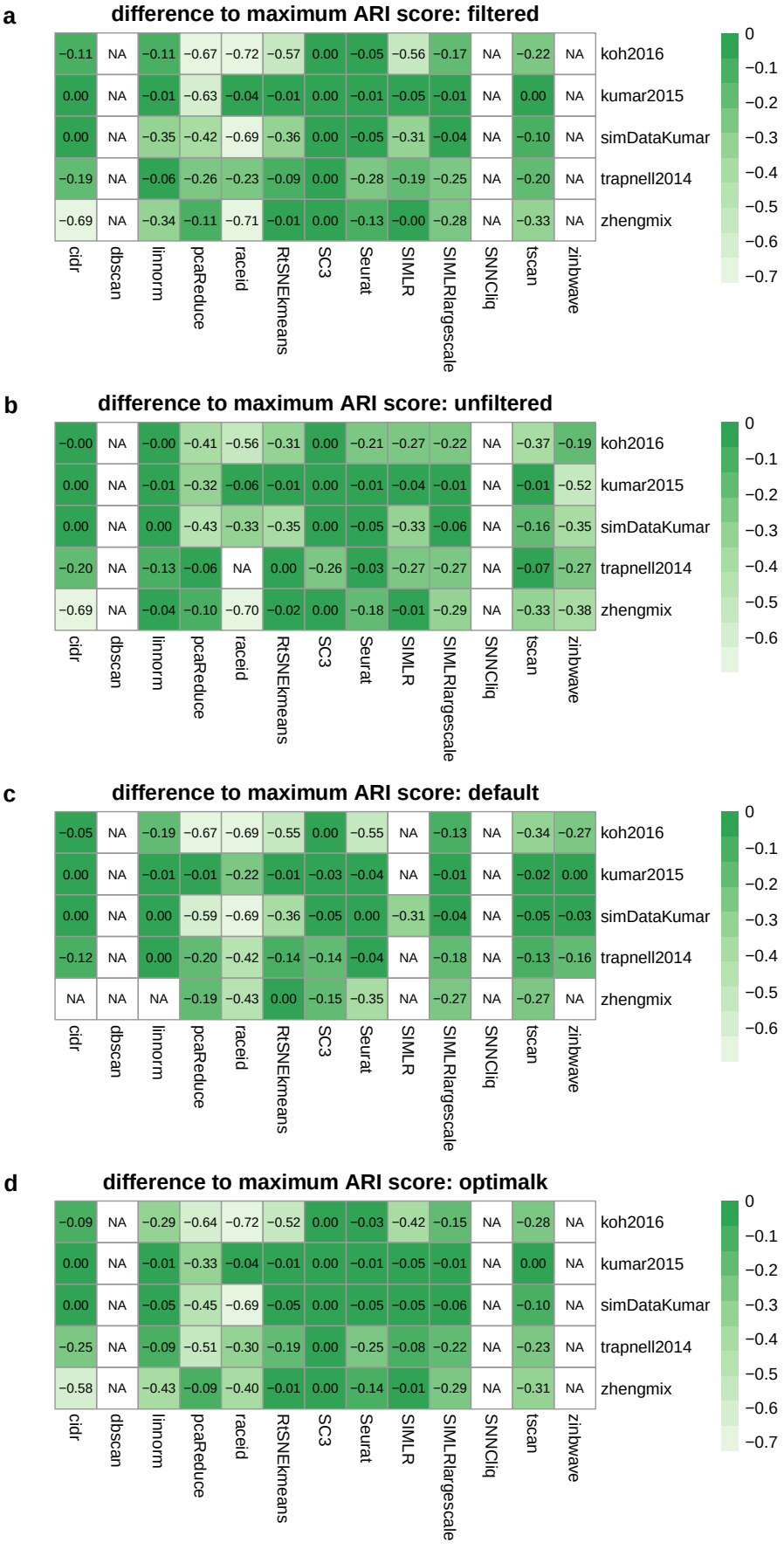


Figure 22: Differences between maximum ARI scores.

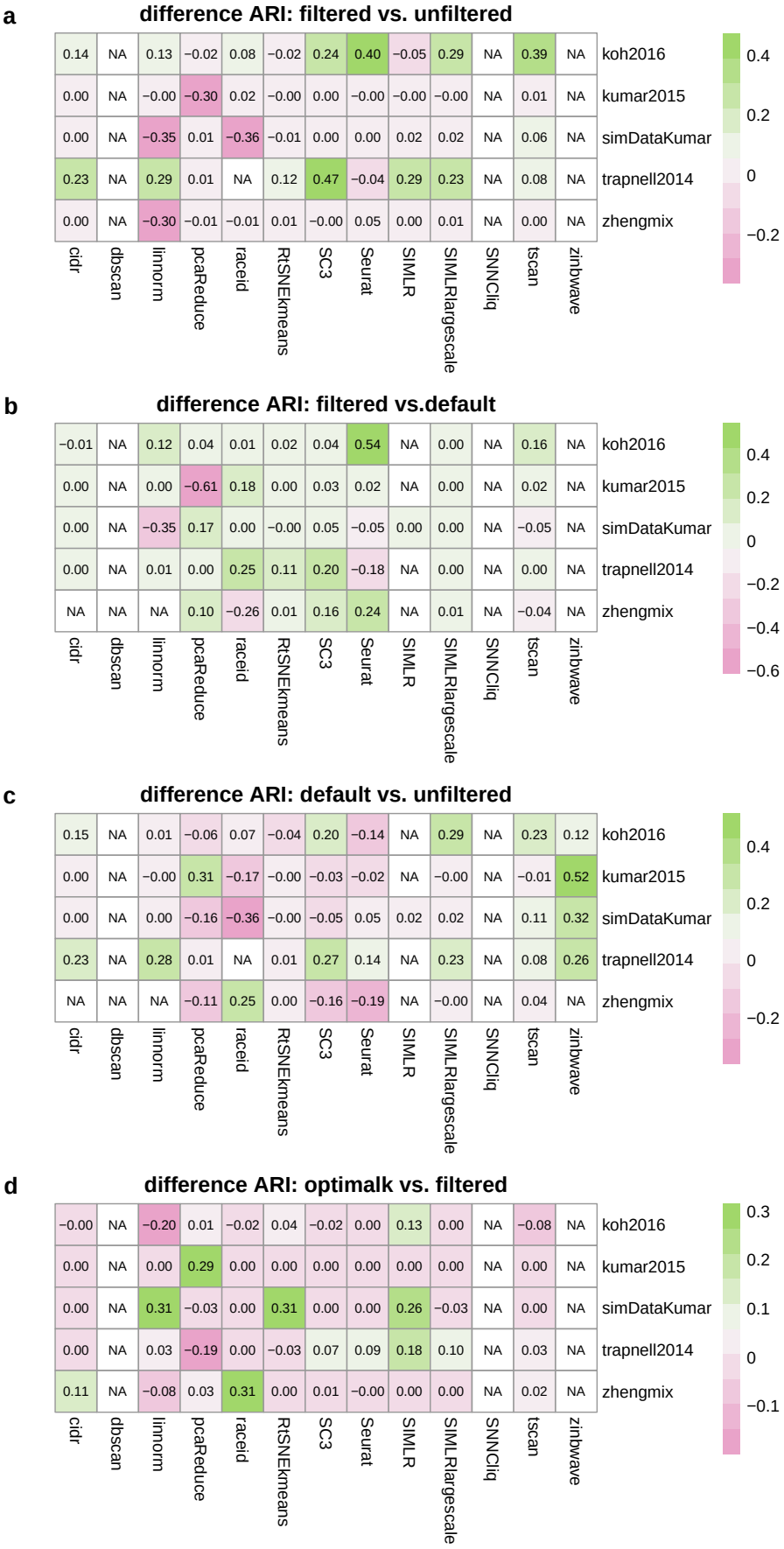


Figure 23: Differences in ARI scores between data sets.

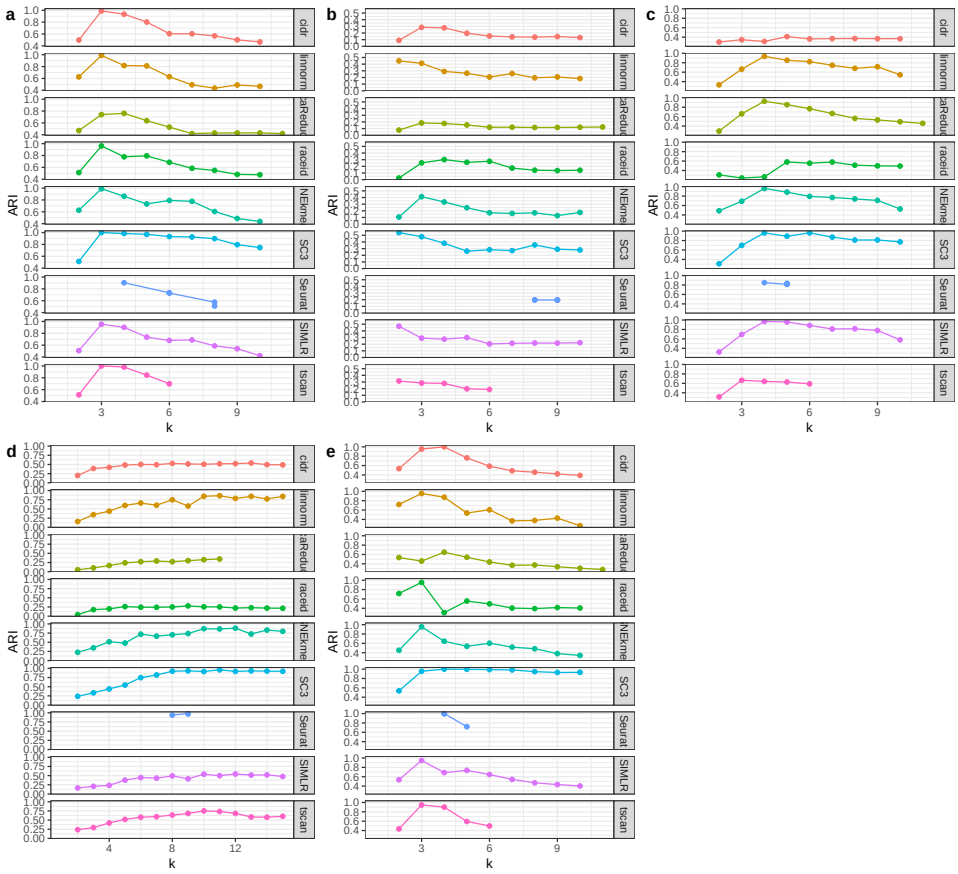


Figure 24: ARI scores for range of parameters for the data sets Kumar (a), Trapnell (b), Zhengmix (c), Koh (d) and simDatakKumar (e). Shown are the methods where the number of cluster could be defined. Shown are the methods...

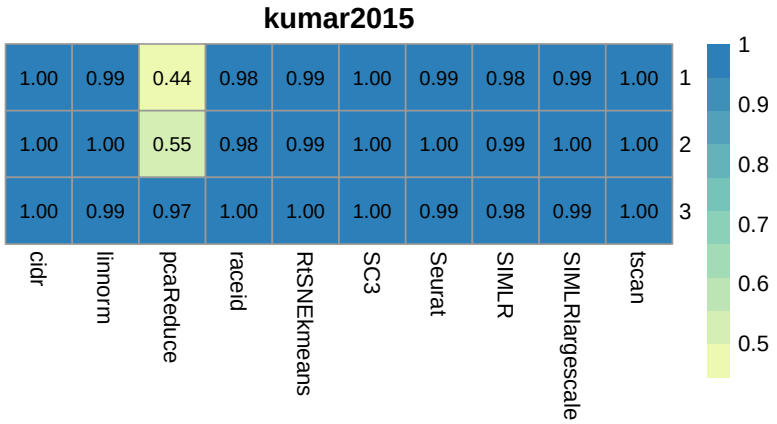


Figure 25: F1 scores for the filtered Kumar dataset

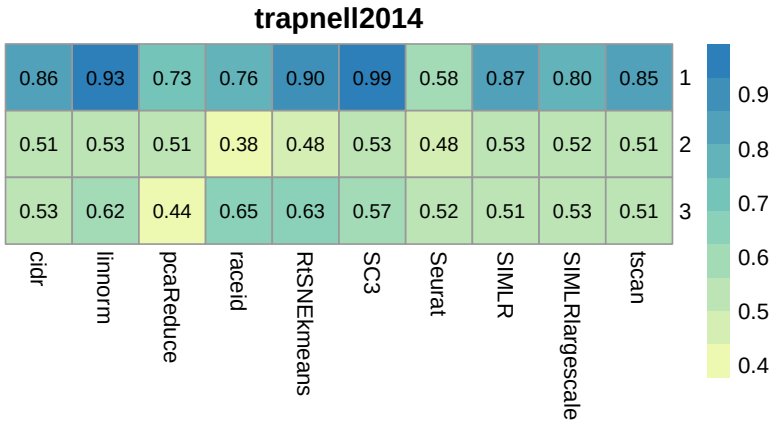


Figure 26: F1 scores for the filtered Trapnell dataset

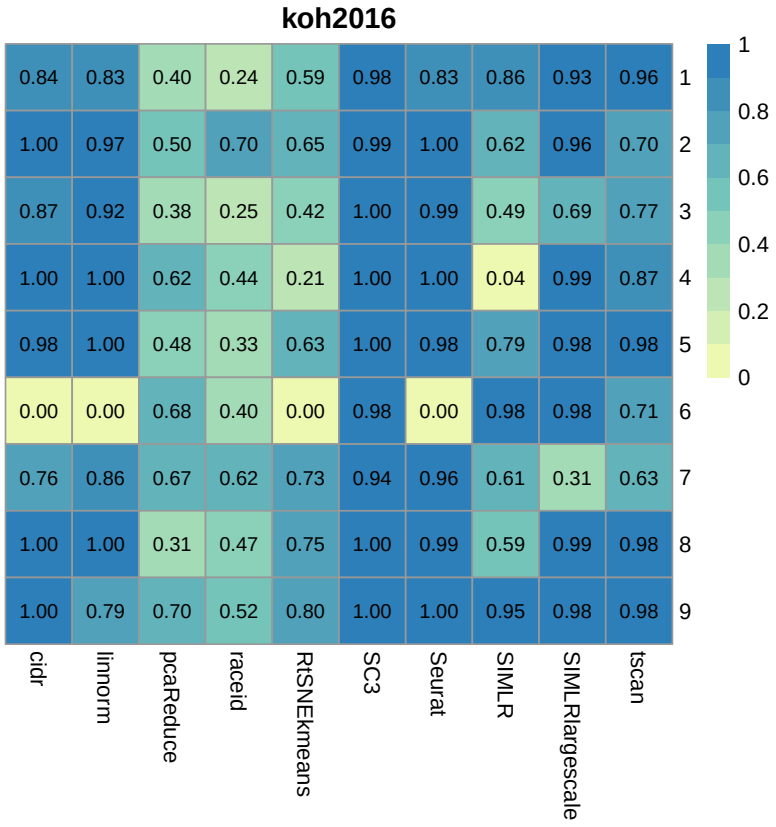


Figure 27: F1 scores for the filtered Koh dataset

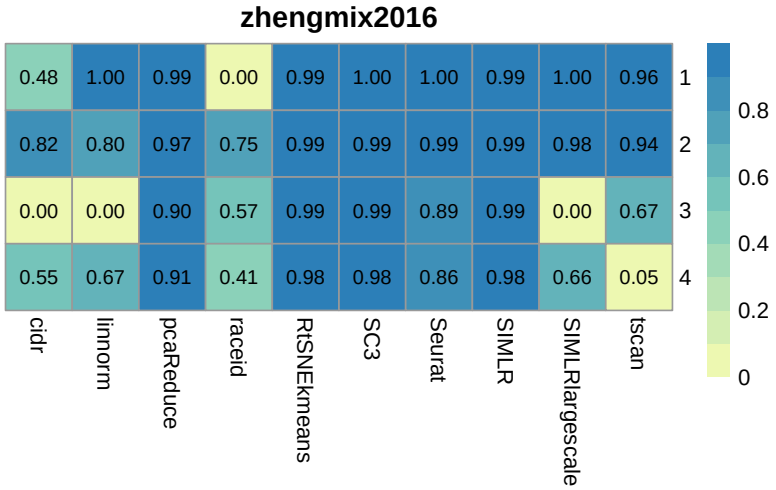


Figure 28: F1 scores for the filtered Zheng mix dataset

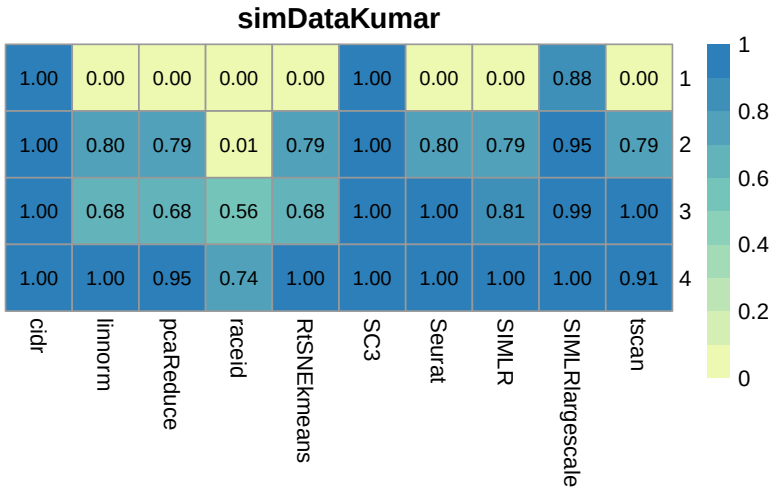


Figure 29: F1 scores for the filtered simDataKumar.

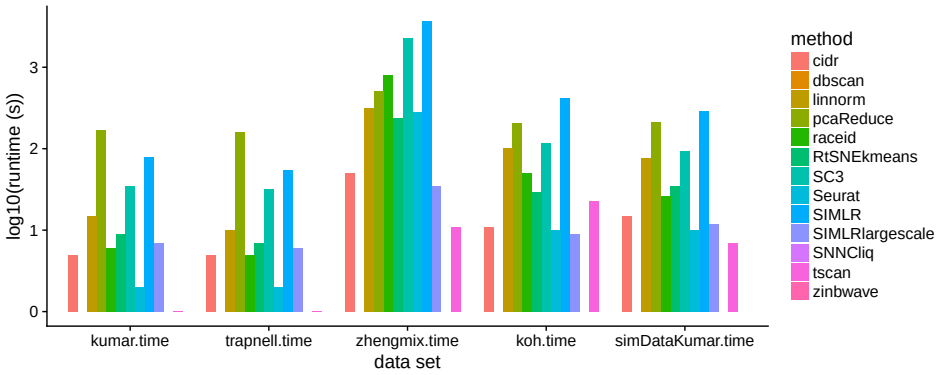


Figure 30: Runtime for the methods on the data sets Kumar etc.

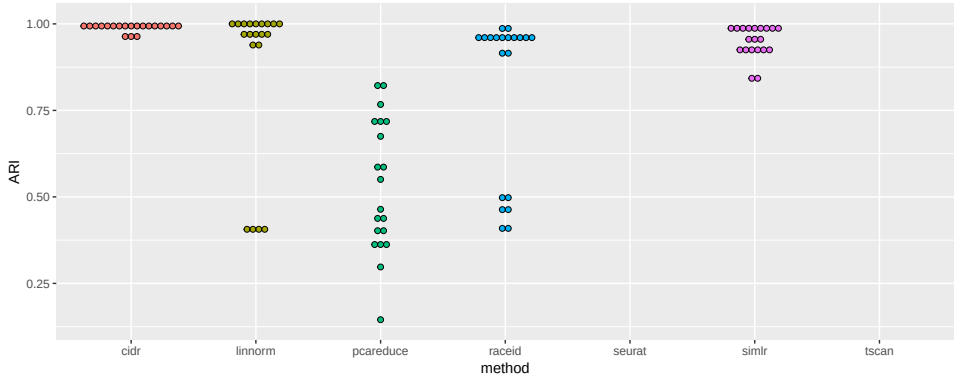


Figure 31: Stability analysis results with (20) bootstrap samples for the Kumar dataset.

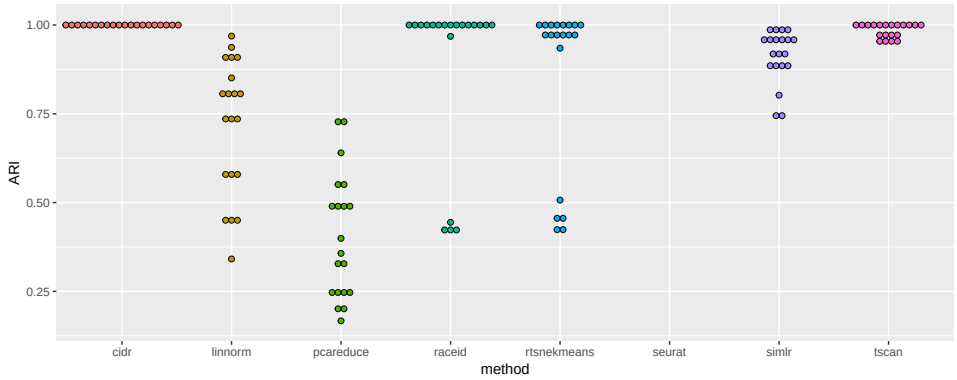


Figure 32: Stability analysis results with 20 subsamples (n=100) for the Kumar dataset.

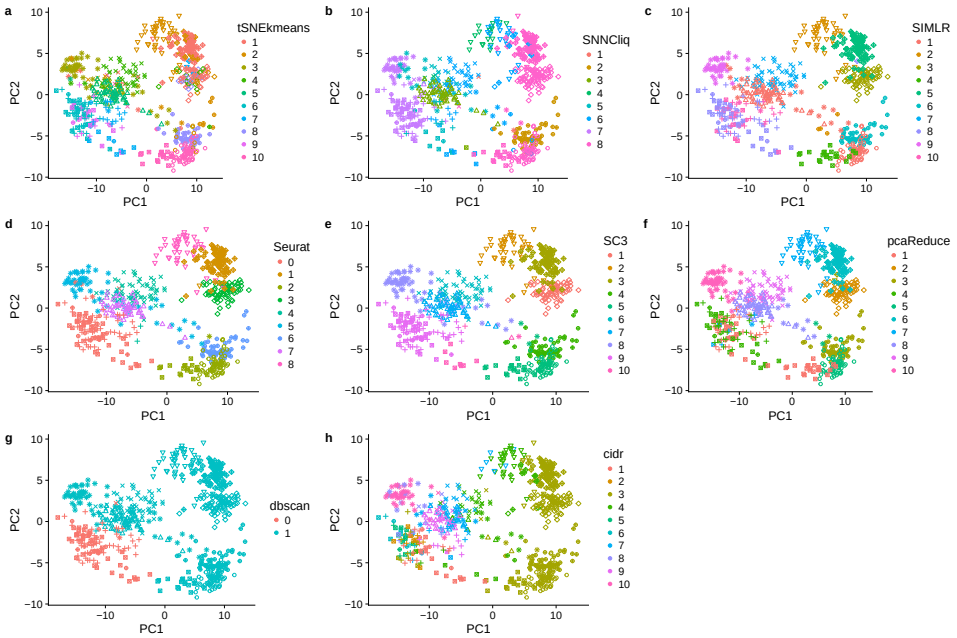


Figure 33: Clusters koh2016 on PC representations.

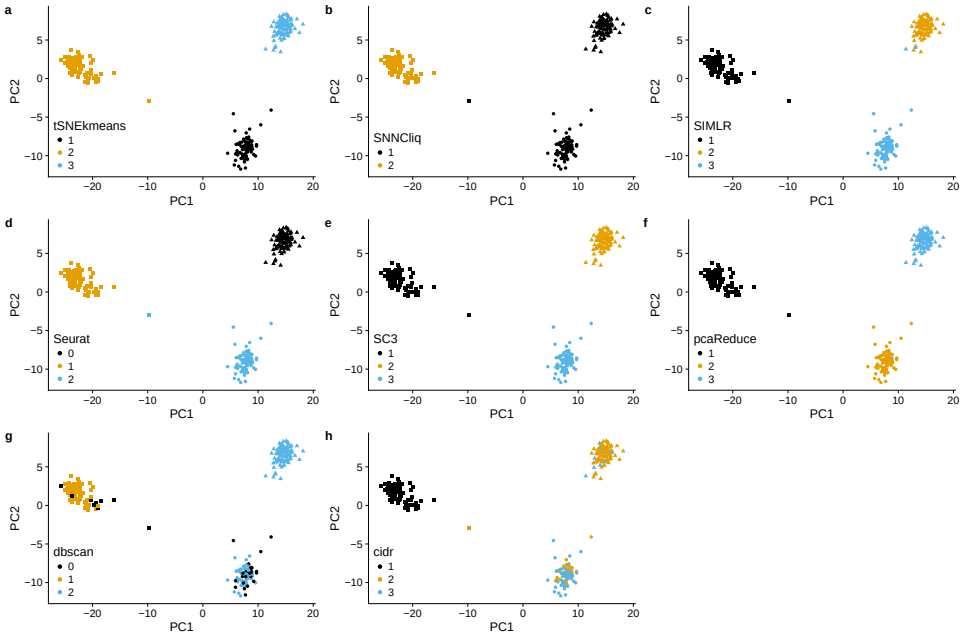


Figure 34: Clusters Kumar 2015 on PC representations.

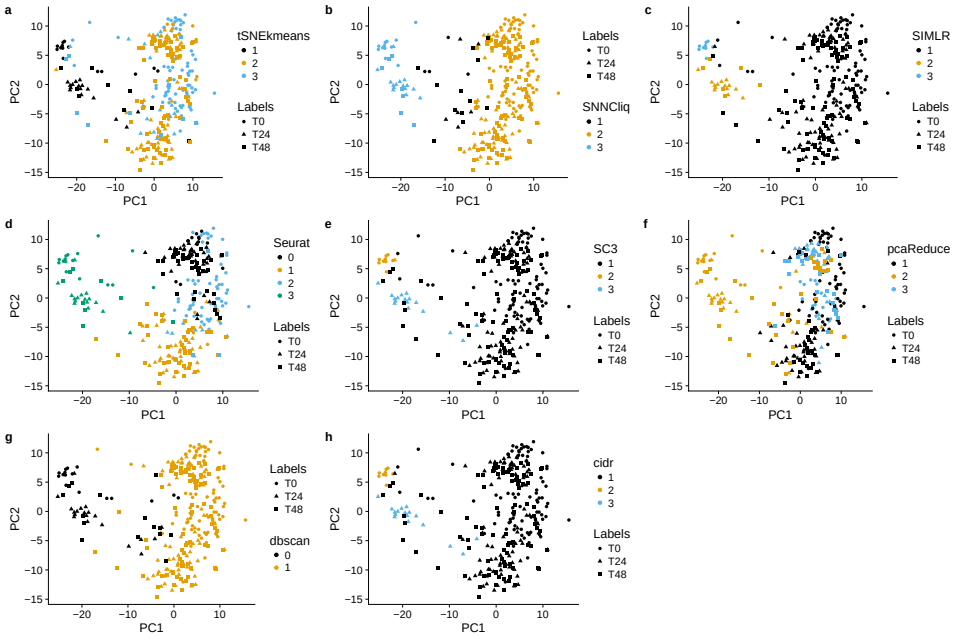


Figure 35: Clusters Trapnell 2014 on PC representations.

	method	parameters	default	annotated k	highest ARI (from
1	cidr	n	NULL	NULL	
2	cidr	nCluster	NULL	k	
3	cidr	nPC	4	4	
4	cidr	cMethod	wardD2	wardD2	
5	dbscan	eps	user	user	
6	dbscan	minPts (kNN)	5	0.1	
7	linnorm	minZeroPortion	0.75	0.75 , (0.25 for Zheng)	
8	linnorm	num_PC	3	3	
9	linnorm	num_Center	1 to 20	k	
10	RaceID	min.total	3000 (200 for Zheng)	3000 (200 for Zheng)	
11	RaceID	max.expr	Inf	Inf	
12	RaceID	min.expr	5	5	
13	RaceID	minnumber	1	1	
14	RaceID	do.gap	TRUE	FALSE	
15	RaceID	clustrnr	20	0	
16	RaceID	cln	0	k	
17	Rtsnekmeans	k	k	k	
18	Rtsnekmeans	Perplexity	30	30	
19	Rtsnekmeans	initial_dims	50	30	
20	SIMLRlargescale	c	k	k	
21	SIMLRlargescale	k	10	10	
22	SIMLRlargescale	kk	100	100	
23	SIMLRlargescale	normalize	FALSE	FALSE	
24	SIMLR	c	k	k	
25	SIMLR	k	10	10	
26	SIMLR	kk	100	100	
27	SIMLR	normalize	FALSE	TRUE	
28	SIMLR	no.dim			
29	tscan	minexpr_percent	0.1 to 0.5	0.1 to 0.5	
30	tscan	clusternum	2 to 20	k	
31	SC3	ks	2 to 10	elbowplot	
32	SC3	k_estimator	TRUE	FALSE	
33	SEURAT	k.param	30	10 percent	
34	SEURAT	dims.use	NULL	screeplot	
35	SEURAT	reduction.type	pca	pca	
36	SEURAT	resolution	0.8	0.8	
37	pcaReduce	q	30	30	
38	pcaReduce	nbt	100	100	

References

Andrews, T. S. and Hemberg, M. (2017). Identifying cell populations with scrnaseq. *Molecular aspects of medicine*.

Hartigan, J. A. and Wong, M. A. (1979). Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108.

Huber, W., Von Heydebreck, A., Sültmann, H., Poustka, A., and Vingron, M. (2002). Variance stabilization applied to microarray data calibration and to the quantification of differential expression. *Bioinformatics*, 18(suppl_1):S96–S104.

Ji, Z. and Ji, H. (2015). Tscan: Tools for single-cell analysis.

Kiselev, V. Y., Kirschner, K., Schaub, M. T., Andrews, T., Yiu, A., Chandra, T., Natarajan, K. N., Reik, W., Barahona, M., Green, A. R., et al. (2017). Sc3: consensus clustering of single-cell rna-seq data. *Nature methods*.

Koh, P. W., Sinha, R., Barkal, A. A., Morganti, R. M., Chen, A., Weissman, I. L., Ang, L. T., Kundaje, A., and Loh, K. M. (2016). An atlas of transcriptional, chromatin accessibility, and surface marker changes in human mesoderm development. *Scientific data*, 3:160109.

Kumar, R. M., Cahan, P., Shalek, A. K., Satija, R., DaleyKeyser, A. J., Li, H., Zhang, J., Pardee, K., Gennert, D., Trombetta, J. J., et al. (2014). Deconstructing transcriptional heterogeneity in pluripotent stem cells. *Nature*, 516(7529):56–61.

Lin, P., Troup, M., and Ho, J. W. (2017). Cidr: Ultrafast and accurate clustering through imputation for single-cell rna-seq data. *Genome biology*, 18(1):59.

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

Lun, A. T., McCarthy, D. J., and Marioni, J. C. (2016b). A step-by-step workflow for low-level analysis of single-cell rna-seq data with bioconductor. *F1000Research*, 5.

Oshlack, A., Phipson, B., and Zappia, L. (2017). Splatter: simulation of single-cell rna sequencing data. *Genome biology*, 18(1):174.

Picelli, S., Björklund, Å. K., Faridani, O. R., Sagasser, S., Winberg, G., and Sandberg, R. (2013). Smart-seq2 for sensitive full-length transcriptome profiling in single cells. *Nature methods*, 10(11):1096–1098.

Risso, D., Perraudeau, F., Gribkova, S., Dudoit, S., and Vert, J.-P. (2017). Zinb-wave: A general and flexible method for signal extraction from single-cell rna-seq data. *bioRxiv*, page 125112.

Soneson, C., Love, M. I., and Robinson, M. D. (2015). Differential analyses for rna-seq: transcript-level estimates improve gene-level inferences. *F1000Research*, 4.

Tang, F., Barbacioru, C., Wang, Y., Nordman, E., Lee, C., Xu, N., Wang, X., Bodeau, J., Tuch, B. B., Siddiqui, A., et al. (2009). mrna-seq whole-transcriptome analysis of a single cell. *Nature methods*, 6(5):377–382.

Tibshirani, R., Walther, G., and Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423.

Trapnell, C., Cacchiarelli, D., Grimsby, J., Pokharel, P., Li, S., Morse, M., Lennon, N. J., Livak, K. J., Mikkelsen, T. S., and Rinn, J. L. (2014). The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nature biotechnology*, 32(4):381–386.

Van Der Maaten, L. (2013). Barnes-hut-sne. *arXiv preprint arXiv:1301.3342*.

Wagner, A., Regev, A., and Yosef, N. (2016). Revealing the vectors of cellular identity with single-cell genomics. *Nature biotechnology*, 34(11):1145–1160.

Wang, B., Zhu, J., Pierson, E., Ramazzotti, D., and Batzoglou, S. (2017). Visualization and analysis of single-cell rna-seq data by kernel-based similarity learning. *Nature Methods*, 14(4):414–416.

Xu, C. and Su, Z. (2015). Identification of cell types from single-cell transcriptomes using a novel clustering method. *Bioinformatics*, 31(12):1974–1980.

Yau, C. et al. (2016). pcareduce: hierarchical clustering of single cell transcriptional profiles. *BMC bioinformatics*, 17(1):140.

Yip, S. H., Wang, P., Kocher, J.-P. A., Sham, P. C., and Wang, J. (2017). Linnorm: improved statistical analysis for single cell rna-seq expression data. *Nucleic acids research*.

Zheng, G. X., Terry, J. M., Belgrader, P., Ryvkin, P., Bent, Z. W., Wilson, R., Ziraldo, S. B., Wheeler, T. D., McDermott, G. P., Zhu, J., et al. (2017). Massively parallel digital transcriptional profiling of single cells. *Nature communications*, 8:14049.

Zwiener, I., Frisch, B., and Binder, H. (2014). Transforming rna-seq data to improve the performance of prognostic gene signatures. *PloS one*, 9(1):e85150.