Statistical analysis for scRNAseq data

Cathy Maugis-Rabusseau

cathy.maugis@insa-toulouse.fr

INSTITUT de MATHEMATIQUES de TOULOUSE

INSA TOULOUSE

Plan

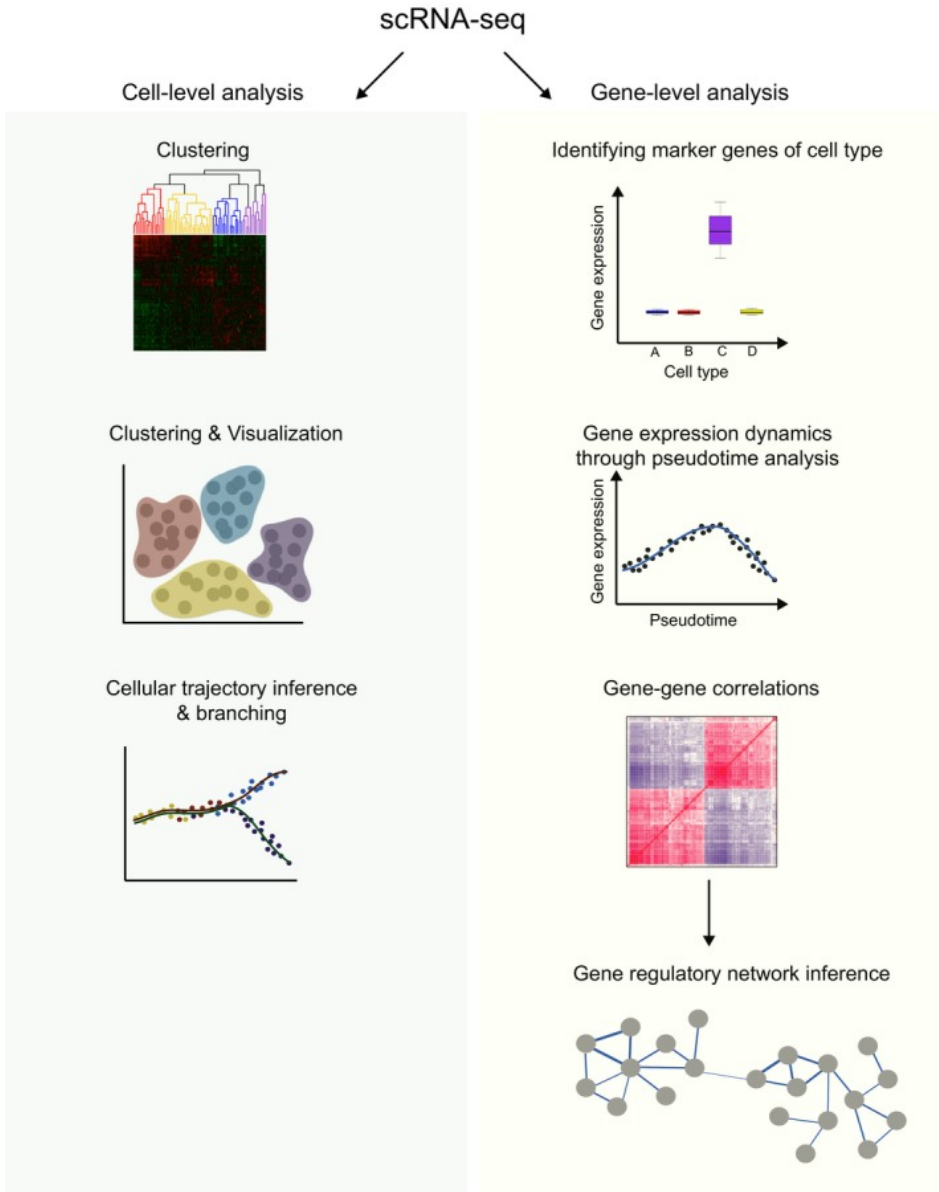
1. Introduction
2. Feature selection / extraction
3. Dimension reduction
4. Single cell clustering
5. Pseudotime analysis
6. Differential analysis

scRNA-seq data

- n cells, G genes: $n \leq G$ or $n \approx G \Longrightarrow$ high dimensionality
- Measures: $x_{ij} =$ expression of the gene j for the cell $i \quad \in \mathbb{N}$
- Technical and biological noise
- High variability
- Zero-inflated data $\Longrightarrow$ "sparsity"
  ($\geq$ 80% of zeros per raw, dropouts)

Biological questions

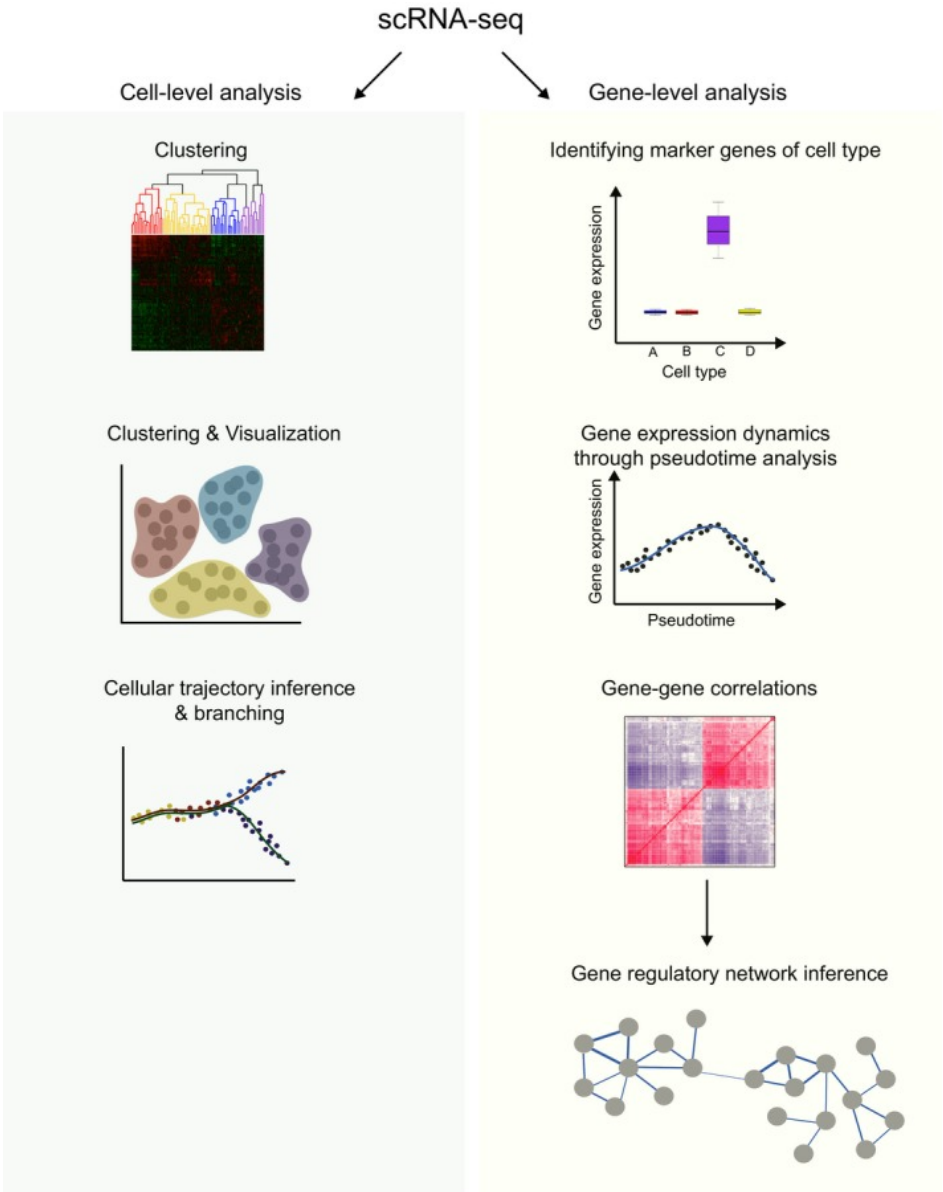
- Are there distinct subpopulations of cells?
- For each cell type, what are the marker genes?
- How visualize the cells?
- Are there continuums of differentiation / activation cell states?
- ...



Rostom et al, FEBS 2017

Statistical analysis

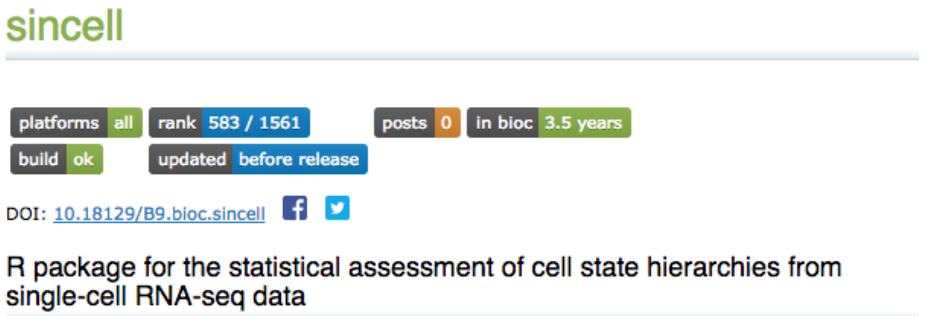
- Clustering of cells
- Variable (gene) selection in learning or differential analysis (hypothesis testing)
- Reduction dimension
- Network inference
- ...



Rostom et al, FEBS 2017

Some bio-info-stat. pipelines/workflows

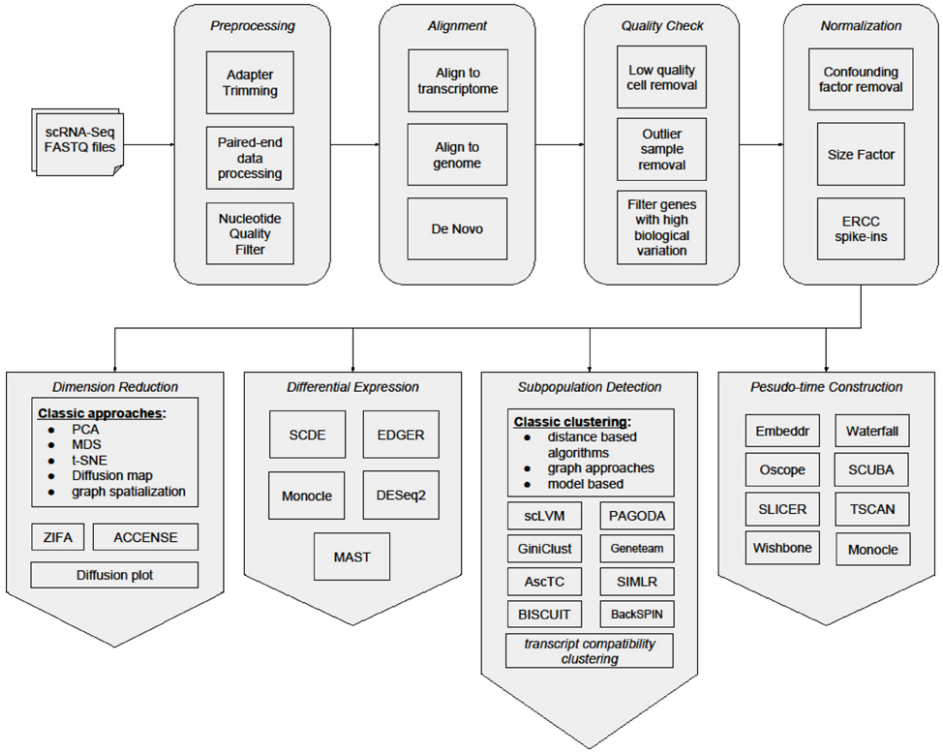
- [Juliá et al., 2015] Sincell (Bioconductor/R package)



https://bioconductor.org/packages/release/bioc/html/sincell.html

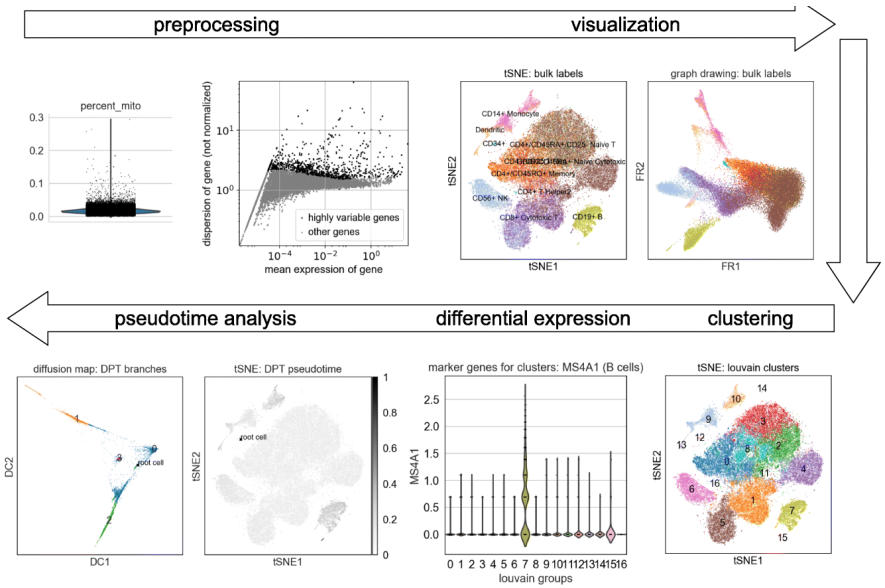
Some bio-info-stat. pipelines/workflows

- [Juliá et al., 2015] Sincell (Bioconductor/R package)
- [Poirion et al., 2016]



Some bio-info-stat. pipelines/workflows

- [Juliá et al., 2015] Sincell (Bioconductor/R package)
- [Poirion et al., 2016]
- [Wolf et al., 2018] SCANPY



https://github.com/theislab/Scanpy

Some bio-info-stat. pipelines/workflows

- [Juliá et al., 2015] Sincell (Bioconductor/R package)
- [Poirion et al., 2016]
- [Wolf et al., 2018] SCANPY
- [Guo et al., 2015] SINCERA:

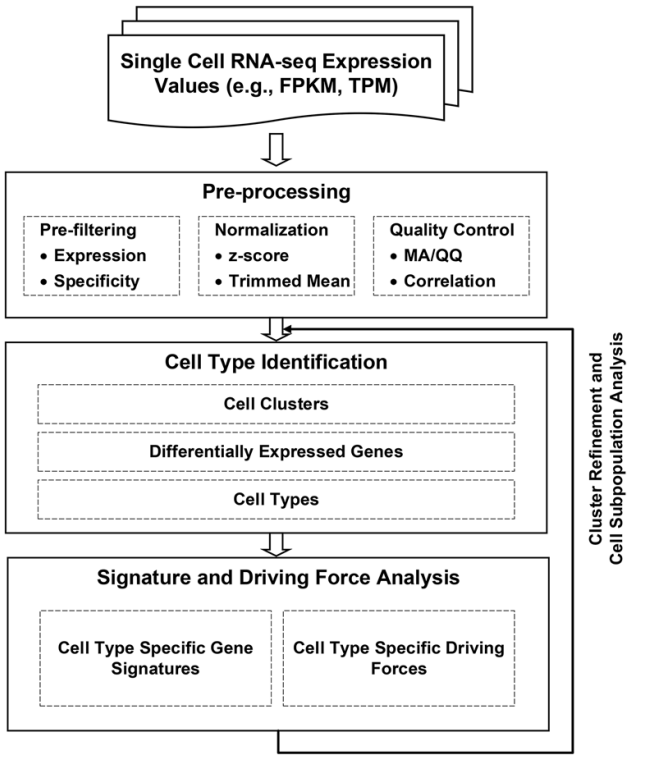


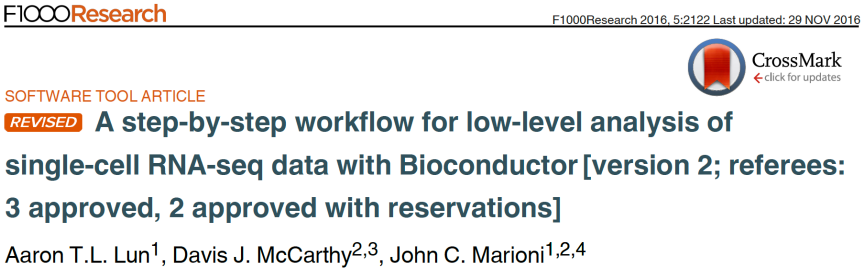
Fig 1. Schematic Workflow. The analytic pipeline consists of three main components: pre-processing, cell type identification, and cell type specific gene signature and driving force identification.

https://github.com/xu-lab/SINCERA

https://research.cchmc.org/pbge/sincera.html

Some bio-info-stat. pipelines/workflows

- [Juliá et al., 2015] Sincell (Bioconductor/R package)
- [Poirion et al., 2016]
- [Wolf et al., 2018] SCANPY
- [Guo et al., 2015] SINCERA:
- [Lun et al., 2016] Workflow Package : simpleSingleCell

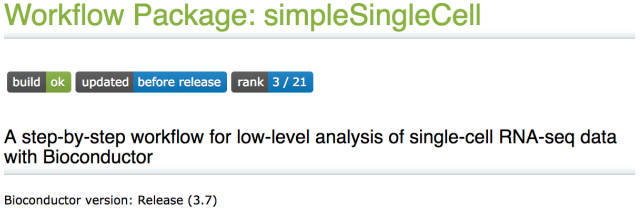

F1000Research F1000Research 2016, 5:2122 Last updated: 29 NOV 2016

CrossMark click for updates

SOFTWARE TOOL ARTICLE

REVISED A step-by-step workflow for low-level analysis of single-cell RNA-seq data with Bioconductor [version 2; referees: 3 approved, 2 approved with reservations]

Aaron T.L. Lun[1], Davis J. McCarthy[2,3], John C. Marioni[1,2,4]


Workflow Package: simpleSingleCell

build ok updated before release rank 3 / 21

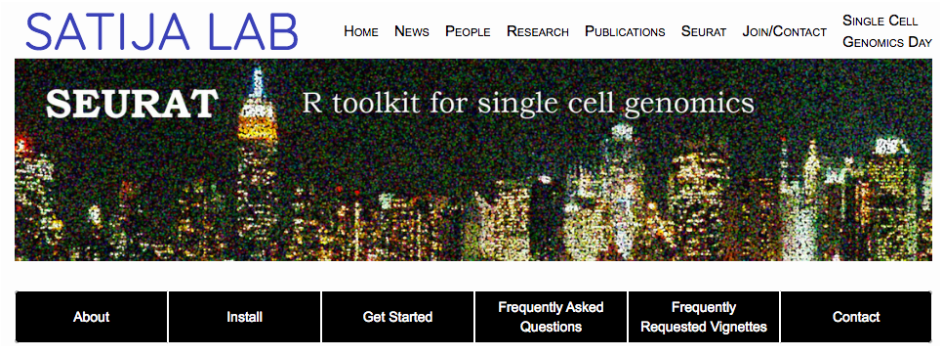
A step-by-step workflow for low-level analysis of single-cell RNA-seq data with Bioconductor

Bioconductor version: Release (3.7)

https://bioconductor.org/packages/release/workflows/html/simpleSingleCell.html

Some bio-info-stat. pipelines/workflows

- [Juliá et al., 2015] Sincell (Bioconductor/R package)
- [Poirion et al., 2016]
- [Wolf et al., 2018] SCANPY
- [Guo et al., 2015] SINCERA:
- [Lun et al., 2016] Workflow Package : simpleSingleCell
- [Satija et al., 2015] SEURAT:



https://satijalab.org/seurat/

- ...

Plan

1 Introduction

2 **Feature selection / extraction**

3 Dimension reduction

4 Single cell clustering

5 Pseudotime analysis

6 Differential analysis

Feature (gene) extraction

- Simple filtering criteria : see e.g [Lun et al., 2016],[Soneson and Robinson, 2018]
  Filtering of lowly expressed genes:
  - genes expressed in $< \tau\%$ of cells
  - genes with a mean average of expression $< \tau$

- Dropout-based feature selection M3Drop, [Andrews and Hemberg, 2018]

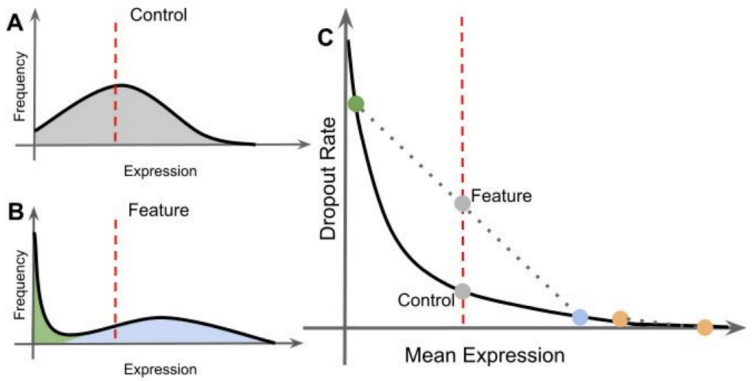


Figure 1: Differentially expressed genes exhibit bimodal expression which increases the dropout rate relative to the mean expression. (**A & B**) Genes with the same mean expression (dashed red line), but (A) is expressed evenly across cells, whereas (B) is highly expressed in some cells (blue) and lowly expressed in others (green). (**C**)This leads to a surplus of dropouts since mean and dropout rate average linearly (dotted line) whereas the expectation (black line) is non-linear. Orange points indicate a gene with very high expression where differential expression leads to only a small increase in dropout-rate.

- Based on the Michaelis-Menten function

$$P_{dropout} = 1 - \frac{S}{K_M + S}$$

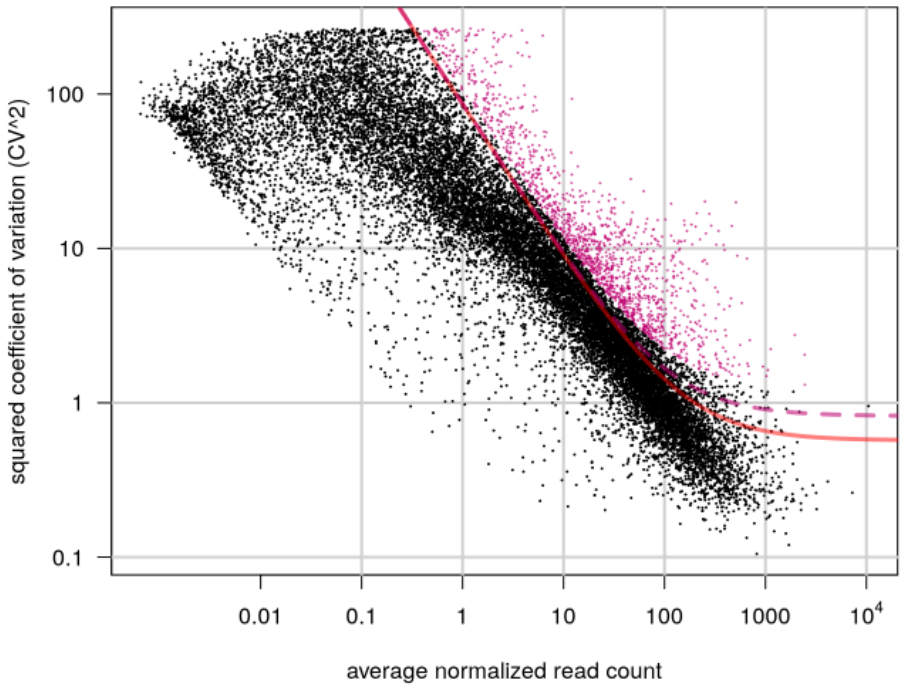
where

$$S = \text{ mean expression}$$
$$P_{dropout} = \text{ dropout rate}$$

- MLE to obtain the global K_M across all genes

Highly Variable Genes (HVG)

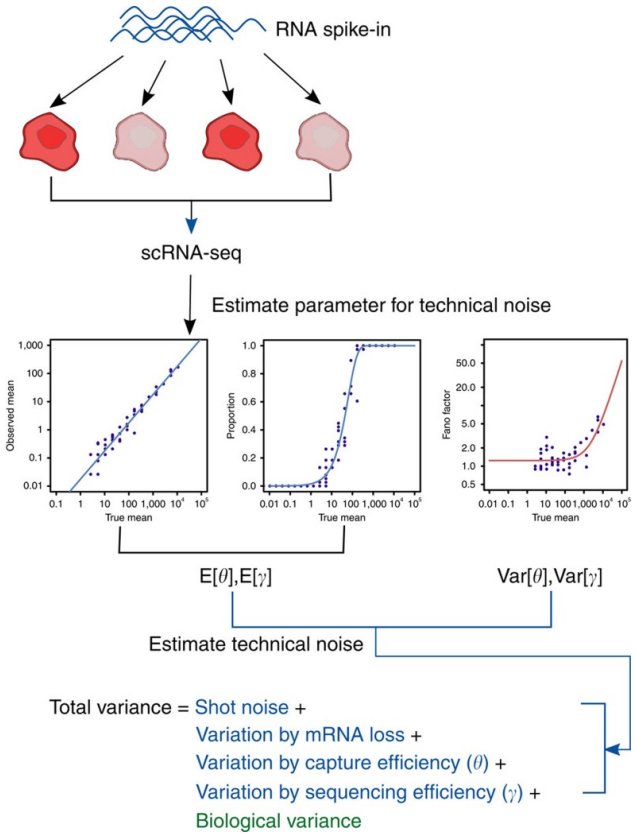
- [Brennecke et al., 2013]
  - Fits a quadratic model (gamma generalized linear model) to the relationship between mean expression and the coefficient of variation squared (CV2)
  - χ^2 test is used to find genes signif. above the curve
  - Implemented in M3Drop package



Highly Variable Genes (HVG)

- [Brennecke et al., 2013]
- [Kim et al., 2015]

Uses spike-ins to estimate parameters related to technical variance and estimates gene-specific biological variability by substracting the estimated technical variance from the total variance.



Highly Variable Genes (HVG)

- [Brennecke et al., 2013]
- [Kim et al., 2015]
- [Vallejos et al., 2015]

BASiCS = Bayesian Analysis of Single-Cell Sequencing Data
Models spike-ins and endogenous genes simultaneously as two Poisson-Gamma hierarchical models

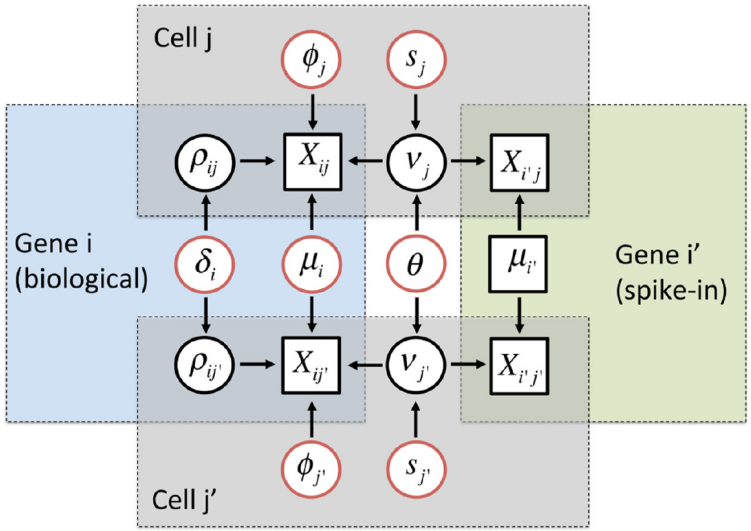


Fig 2. Graphical representation of the hierarchical model implemented in BASiCS. Diagram based on the expression counts of 2 genes (i: biological and i': technical) at 2 cells (j and j'). Squared and circular nodes denote known observed quantities (observed expression counts and added number of spike-in mRNA molecules) and unknown elements, respectively. Whereas black circular nodes represent the random effects that play an intermediate role in our hierarchical structure, red circular nodes relate to unknown model parameters in the top layer of hierarchy in our model. Blue, green and grey areas highlight elements that are shared within a biological gene, technical gene or cell, respectively. BASiCS treats cell-specific normalising constants (ϕ_j's and s_j's) as model parameters, and estimates them by combining information across all genes. Unexplained technical noise is quantified via a single hyper-parameter θ, borrowing information across all genes and cells. Finally, BASiCS quantifies biological cell-to-cell variability via gene-specific hyper-parameters δ_i, borrowing information across all cells.

Highly correlated genes

- Gene-gene correlation:
  - Calculate the gene-gene correlation matrix $\rho = (\rho_{ij})_{i,j=1,\dots,G}$
  - Evaluate the correlation magnitude for each gene : $\tilde{\rho}_i = \max_j |\rho_{ij}|$
  - Take the top few thousand genes having the highest correlation magnitude
- PCA loadings: Select the genes with high PCA loadings
- ...
- Non adapted for batch effects

Plan

1 Introduction

2 Feature selection / extraction

3 **Dimension reduction**

4 Single cell clustering

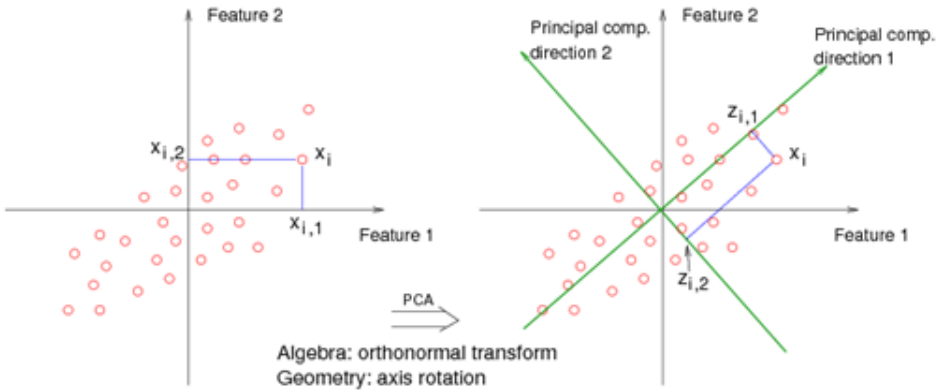
5 Pseudotime analysis

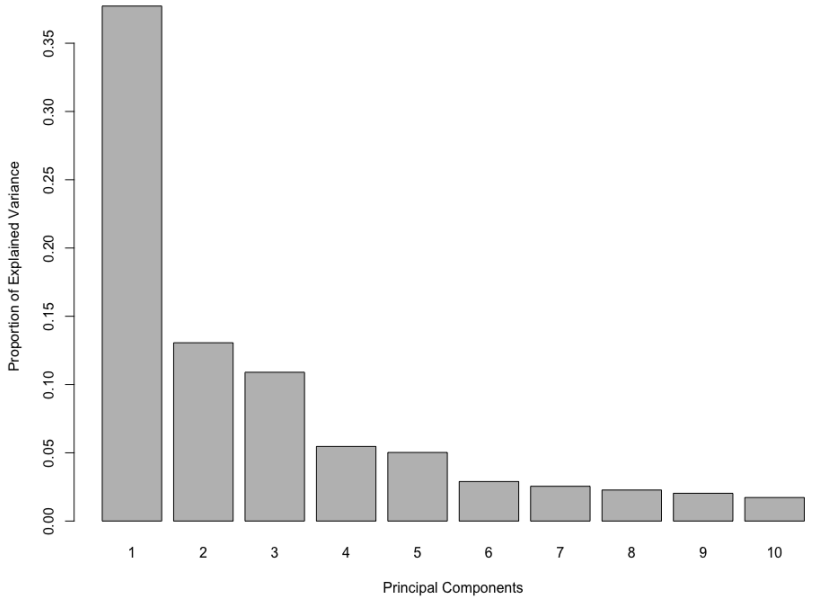
6 Differential analysis

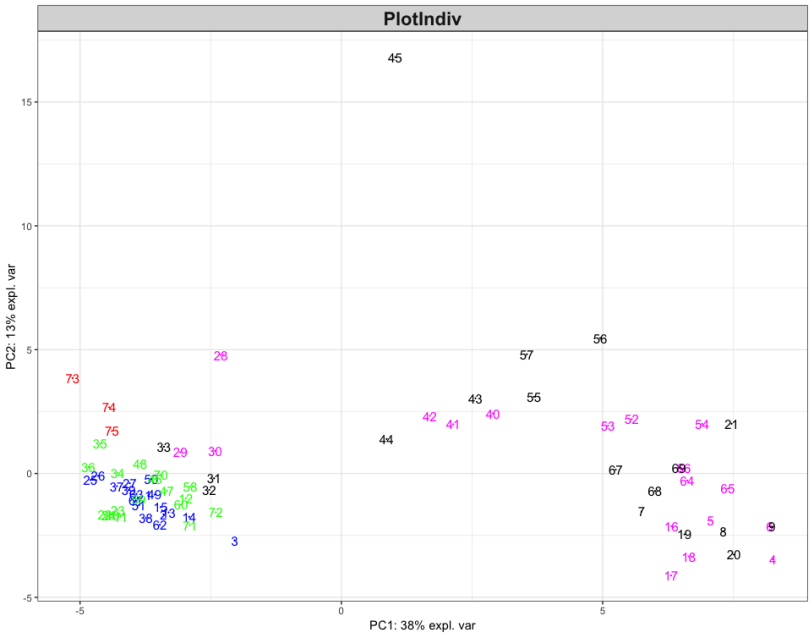
Objectives

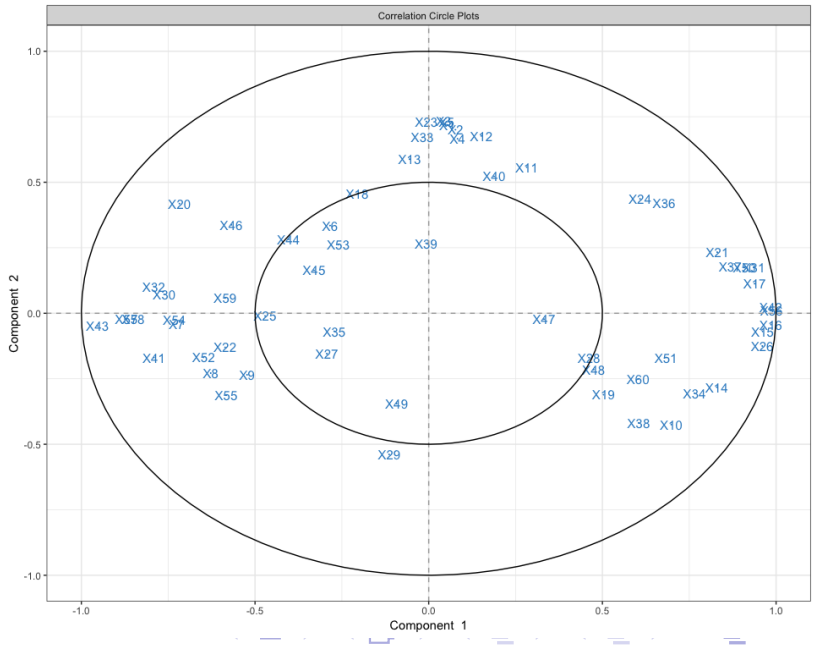
- Minimize curse of dimensionality
- Allow visualization
- Reduce computational time
- ....
- But attention to the interpretations after!

Principal component analysis (PCA)









Principal component analysis (PCA)

- Diagonalization of the covariance (or correlation) matrix
- Linear transformations:
  meta-variables = linear combinations of the genes
- Capture the dimensions with higher variance
- Fast deterministic procedure
- Sparse-PCA : PCA + gene selection

Extensions of PCA for scRNAseq data

- [Pierson and Yau, 2015] : ZIFA (Zero Inflated Factor Analysis)
  - Deals with the large number of zero-values in scRNASeq data
  - Relationship between the dropout rate p_0 and the mean level of non-zero expression (log read count) μ:

$$p_0 = \exp(-\lambda\mu^2)$$

  - ZIFA adopts a latent variable model and uses an EM algorithm for the parameter estimation
  - Python software : https://github.com/epierson9/ZIFA

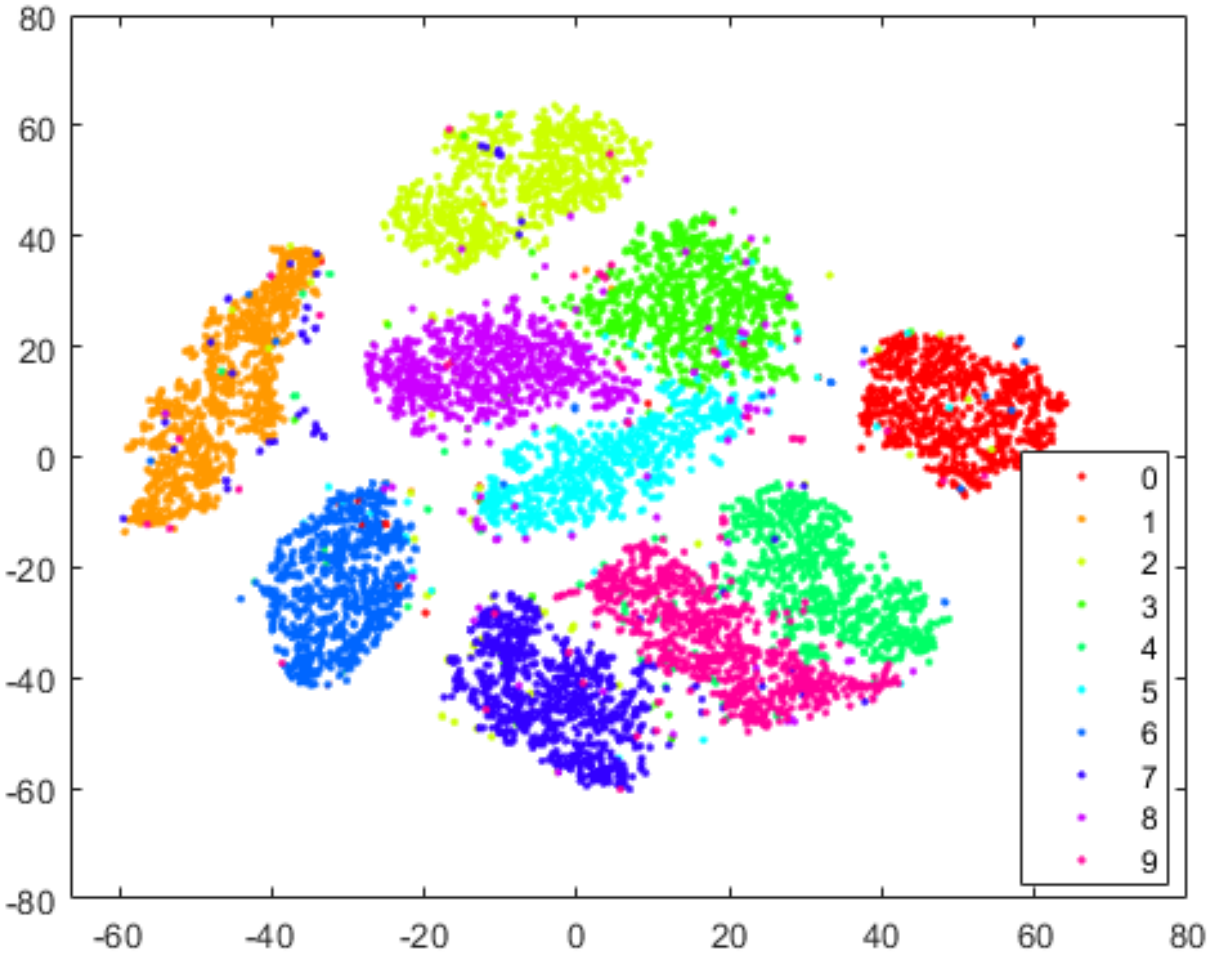
- [Risso et al., 2017] : ZINB-WaVE
  = Zero-Inflated Negative Binomial Model for RNA-Seq Data
  a method similar to PCA based on a zero- inflated negative binomial model instead of a Gaussian model

  https://bioconductor.org/packages/release/bioc/html/zinbwave.html

Extensions of PCA

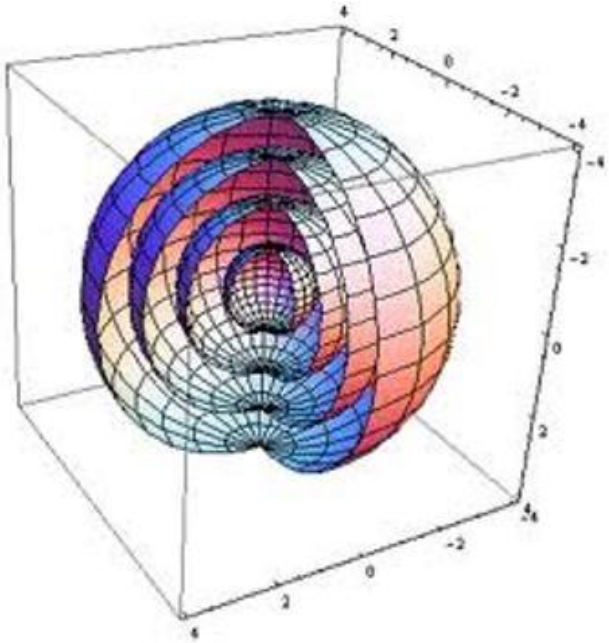
- [Lin et al., 2017] CIDR (https://github.com/VCCRI/CIDR)
  1. Preliminary, $\log(x_{ij} + 1)$
  2. Identification of dropout candidates.
     (CIDR finds a sample-dependent threshold that separates the zero peak from the rest of the expression distribution for each cell)
  3. Estimation of the relationship between dropout rate and gene expression levels
     (non-linear least-squares regression to fit a decreasing logistic function to the data)
  4. Calculation of dissimilarity between the imputed gene expression profiles for every pairs of single cells
  5. PCoA using the CIDR dissimilarity matrix
  6. Clustering (CAH) using the first few principal coordinates

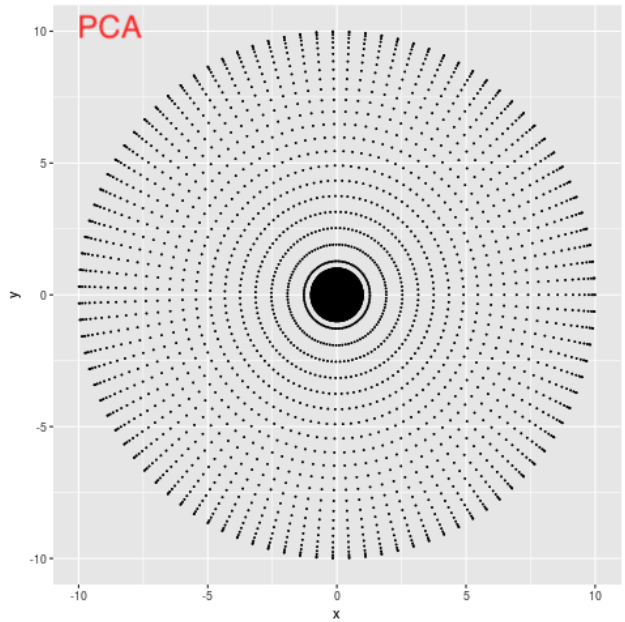
Example of t-SNE plot



t-SNE

- Reduce a dataset to 2 dimensions
- Non-linear dimension reduction technique
- Want to preserve the neighborhood
- "Don’t interpret distances in t-SNE plots"

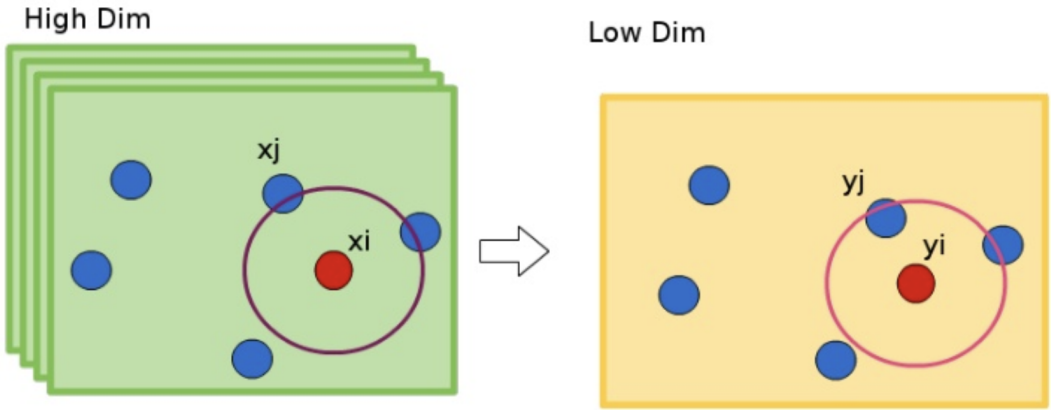




https://constantamateur.github.io/2018-01-02-tSNE/

t-SNE

- Reduce a dataset to 2 dimensions
- Non-linear dimension reduction technique
- Want to preserve the neighborhood
- "Don't interpret distances in t-SNE plots"



- INPUT : $\mathcal{X} = (x_1, \ldots, x_n)$ with $x_i \in \mathbb{R}^G$ (High dimensional data)
- OUTPUT: $\mathcal{Y} = (y_1, \ldots, y_n)$ with $y_i \in \mathbb{R}^2$ (Low dimensional data)

Stochastic Neighbor Embedding (SNE)

Converting the high-dimensional Euclidian distances into conditional probabilities (=similarities)

- Similarity of points in high-dimension

$$p_{j|i} = \frac{\exp(-\|x_i - x_j\|^2/2\sigma_i^2)}{\sum_{k\neq i}\exp(-\|x_i - x_k\|^2/2\sigma_i^2)} \qquad \text{Gaussian distrib.}$$

- Similarity of points in low dimension

$$q_{j|i} = \frac{\exp(-\|y_i - y_j\|^2)}{\sum_{k\neq i}\exp(-\|y_i - y_k\|^2)} \qquad \text{Gaussian distrib.}$$

- Cost function: Kullback-Leibler divergence

$$C(\mathcal{Y}) = \sum_i KL(P_i|Q_i) = \sum_i \sum_j p_{j|i} \ln\left(\frac{p_{j|i}}{q_{j|i}}\right)$$

- Minimize the cost function using gradient descent

Symmetric SNE

- Pairwise similarities:

$$p_{ij} = \frac{\exp(-\|x_i - x_j\|^2/2\sigma^2)}{\sum_{k \neq l} \exp(-\|x_k - x_l\|^2/2\sigma^2)} \qquad \text{Gaussian distrib.}$$

$$q_{ij} = \frac{\exp(-\|y_i - y_j\|^2)}{\sum_{k \neq l} \exp(-\|y_k - y_l\|^2)} \qquad \text{Gaussian distrib.}$$

- In practice,

$$p_{ij} = \frac{p_{i|j} + p_{j|i}}{2n}$$

  + perplexity (effective nb of neighbors) $Perp(P_i) = 2^{H(P_i)}$ (link to σ_i, $H(P_i)$ = Shannon entropy)

- Minimize the cost function

$$C(\mathcal{Y}) = \sum_i \sum_j p_{ij} \ln \left(\frac{p_{ij}}{q_{ij}} \right)$$

t-SNE

- Use Student (heavy tail) distribution than Gaussian in low-dimensional space:

$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq l}(1 + \|y_k - y_l\|^2)^{-1}}$$

- Cost function

$$C(\mathcal{Y}) = \sum_i \sum_j p_{ij} \ln \left(\frac{p_{ij}}{q_{ij}} \right)$$

  - Large p_{ij} modeled by small q_{ij}: large penalty
  - Small p_{ij} modeled by large q_{ij}: small penalty
  - t-SNE: mainly preserves local similarity structure of the data

t-SNE

Algorithm 1: Simple version of t-Distributed Stochastic Neighbor Embedding.

Data: data set $\mathcal{X} = \{x_1, x_2, ..., x_n\}$,
cost function parameters: perplexity $Perp$,
optimization parameters: number of iterations T, learning rate η, momentum $\alpha(t)$.
Result: low-dimensional data representation $\mathcal{Y}^{(T)} = \{y_1, y_2, ..., y_n\}$.
begin
  compute pairwise affinities $p_{j|i}$ with perplexity $Perp$ (using Equation 1)
  set $p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n}$
  sample initial solution $\mathcal{Y}^{(0)} = \{y_1, y_2, ..., y_n\}$ from $\mathcal{N}(0, 10^{-4}I)$
  for t=1 **to** T **do**
    compute low-dimensional affinities q_{ij} (using Equation 4)
    compute gradient $\frac{\delta C}{\delta \mathcal{Y}}$ (using Equation 5)
    set $\mathcal{Y}^{(t)} = \mathcal{Y}^{(t-1)} + \eta \frac{\delta C}{\delta \mathcal{Y}} + \alpha(t) \left(\mathcal{Y}^{(t-1)} - \mathcal{Y}^{(t-2)}\right)$
  end
end

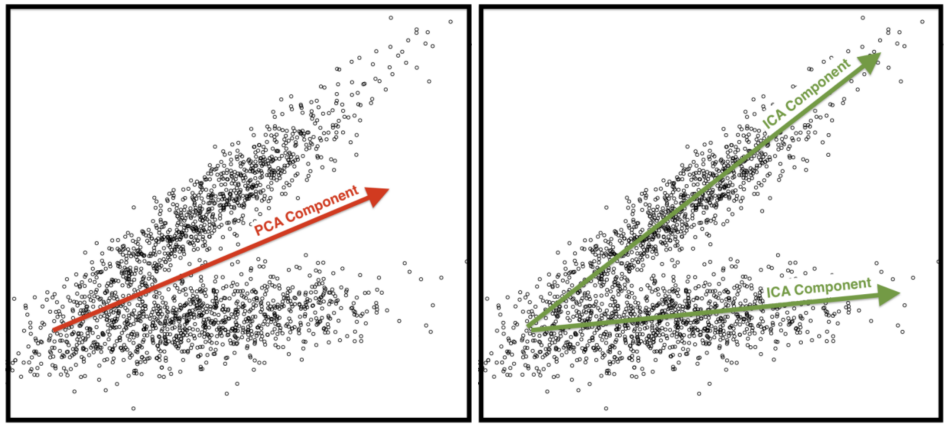
- In practice, t-SNE is used on the first components of PCA to reduce the time calculation
- t-SNE is a stochastic algorithm
- t-SNE is implemented in various programming languages (see https://lvdmaaten.github.io/tsne/)

Other methods

- ICA = Independent Component Analysis

  Computational method for separating a multivariate signal into additive subcomponents.

  This is done by assuming that the subcomponents are non-Gaussian signals and that they are statistically independent from each other.



- Diffusion maps

  Computes a family of embeddings of a data set into Euclidean space whose coordinates can be computed from the eigenvectors and eigenvalues of a diffusion operator on the data.

- ...

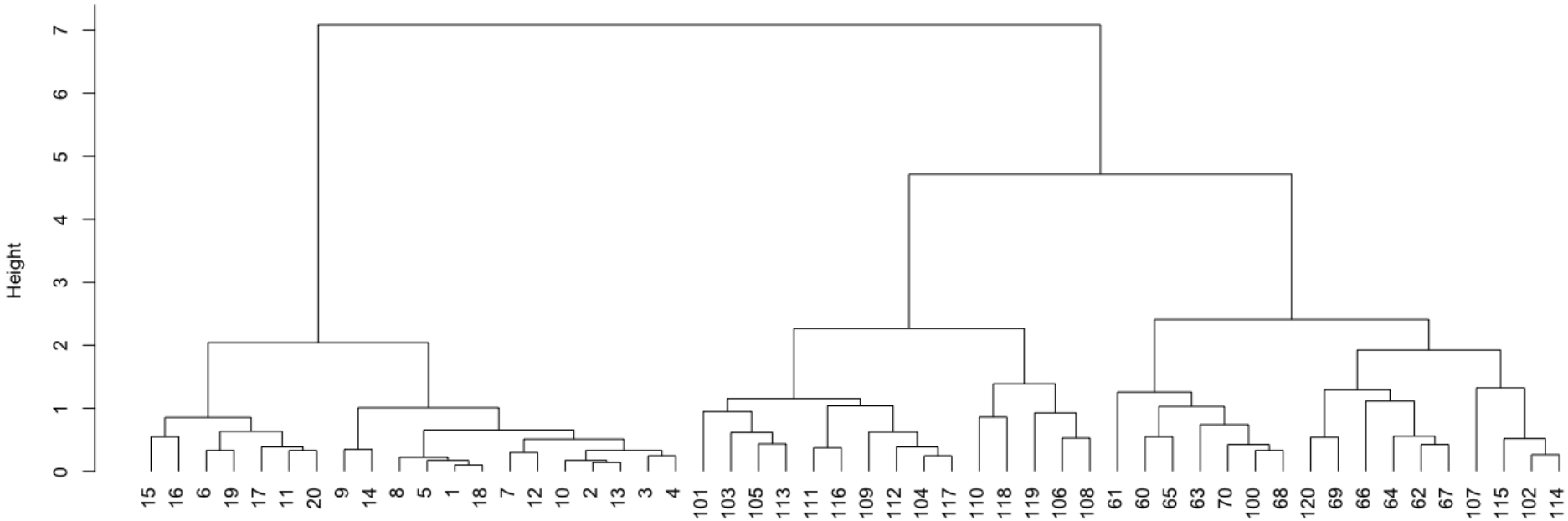
Plan

1. Introduction
2. Feature selection / extraction
3. Dimension reduction
4. **Single cell clustering**
5. Pseudotime analysis
6. Differential analysis

What is clustering?

- Goal : organizing cells into groups whose members are similar in some way
- Fundamental question : What is meant by "similar cells"?
- The number of clusters is unknown
- Typical clustering methods :
  - Hierarchical clustering (CAH),
  - Kmeans clustering,
  - Graph-based clustering
  - Model-based clustering
  - ...

Hierarchical clustering

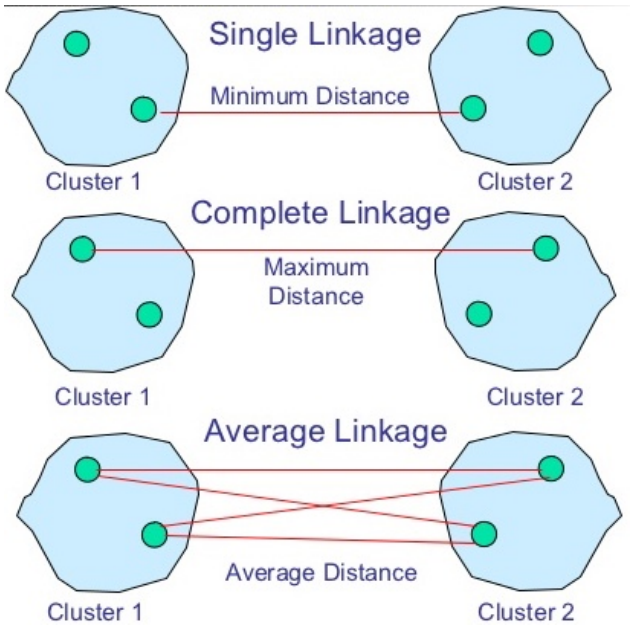


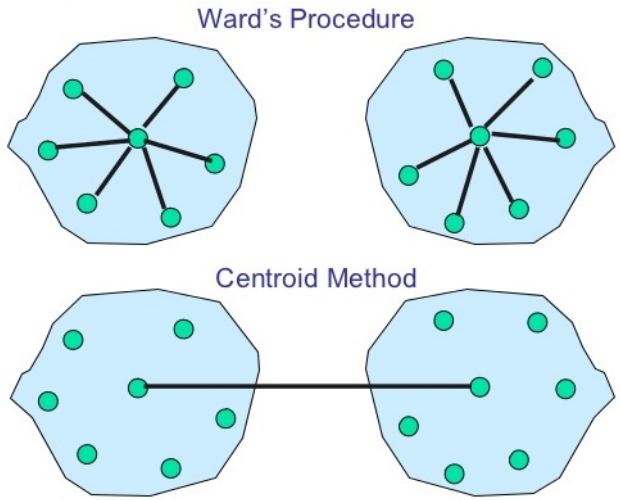
Hierarchical clustering

- Choose a (dis)-similarity between data points and a linkage criterion (similarity between clusters)
- Agglomerative : starts with all data points as individual clusters and joins the most similar ones in a bottom-up approach
- Divisive : starts with all data points in one large cluster and splits it into two at each step (a top-down approach)
- A dendrogram representing the decisions at each merge/division of clusters

Hierarchical clustering

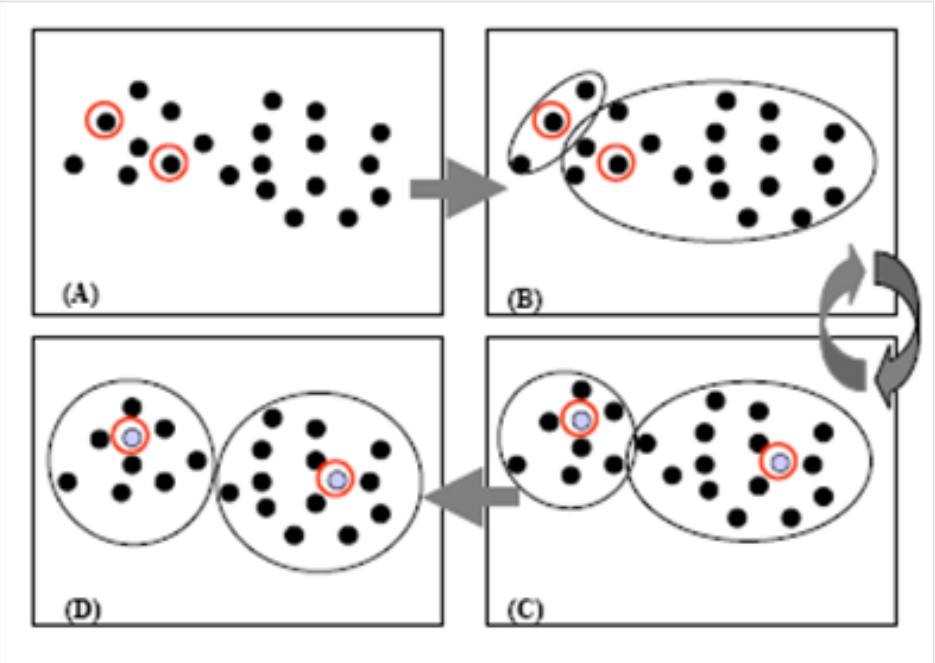
- (dis)-similarity between data points :
  - Euclidian distance
  - 1-correlation, 1-correlation^2, ...
  - Dissimilarity between the imputed gene expression profiles (CIDR),
  - ....
- Linkage criteria : Ward, complete linkage, average linkage, ...





K-means

1. Starts with random selection of cluster centers
2. Assigns each data points to the nearest cluster
3. Calculates the new cluster centers
4. Repeats steps 2-3 until no more changes occur



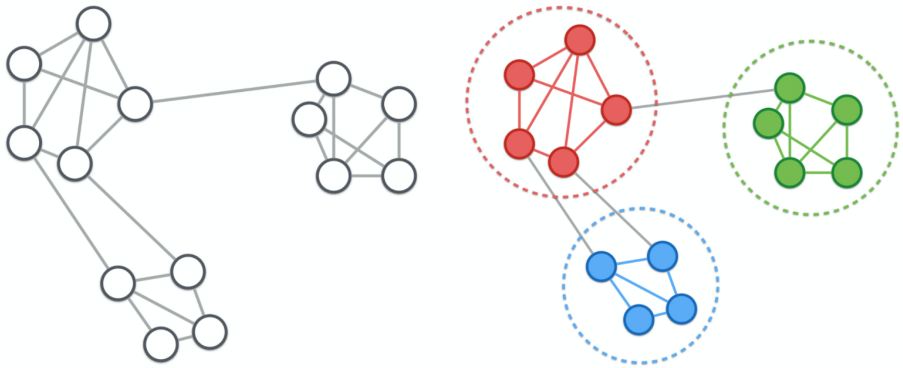
- Require a distance between data points
- Fuzzy Kmeans, ...

Graph-based clustering

- Objects (cells) are represented as nodes
- Assign a weight to each branch between two nodes x and $\tilde{x}$

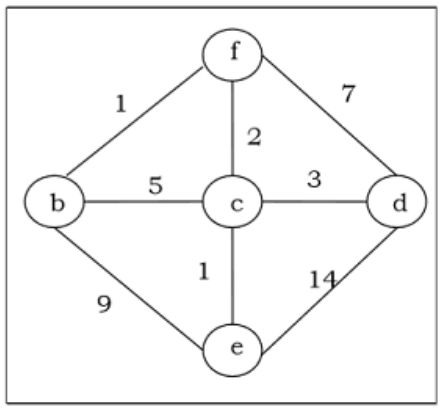
$$w(x, \tilde{x}) = distance(x, \tilde{x})$$

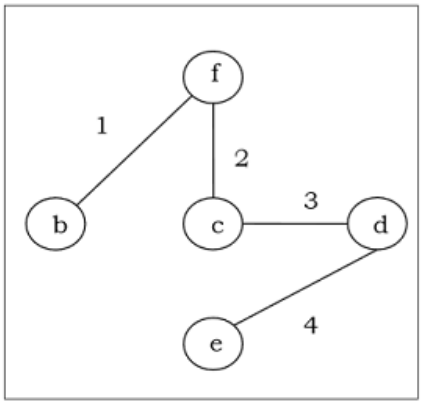
- Clustering



Graph-based clustering

- Minimal Spanning Tree (MST)
  = a connected sub-graph with minimal weight that contains all nodes and has no cycle
  (ex: Prim's algo, Kruskal's algo)





- Different strategies to delete some branchs

Model-based clustering

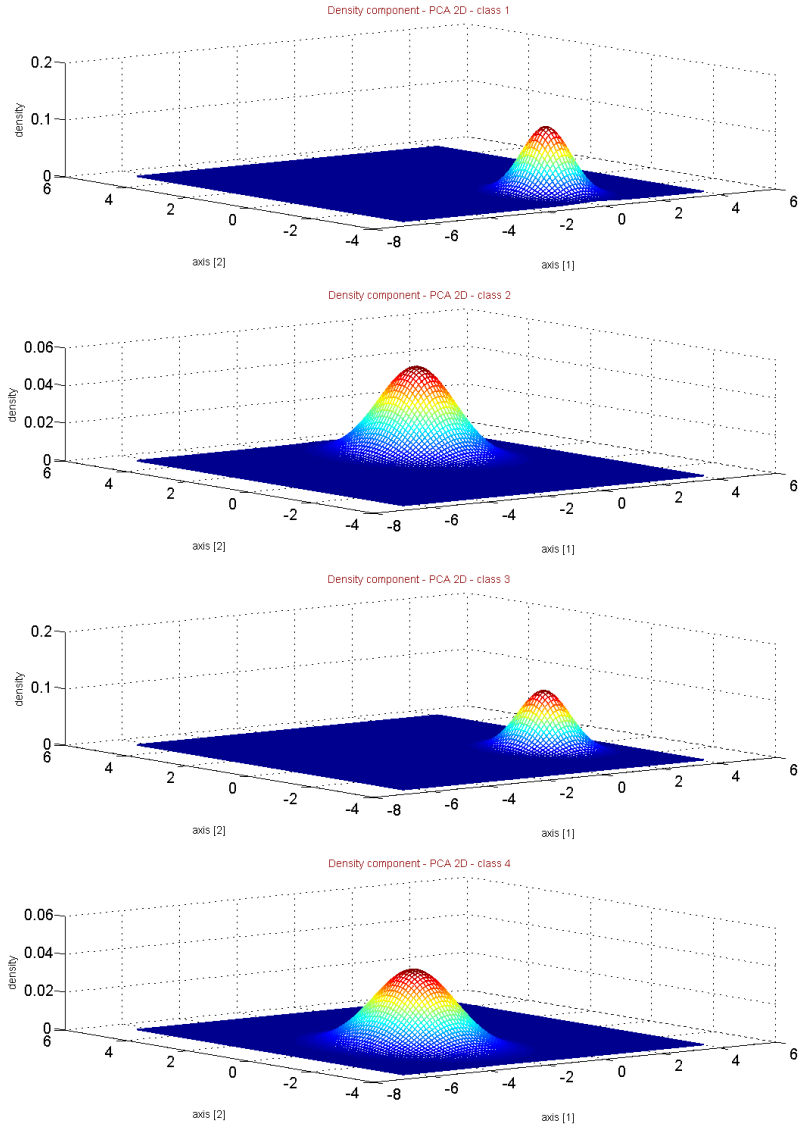
- Data are distributed from an unknown distribution s
- s is estimated by a finite mixture:
  - Data are organized into K subpopulations
  - Each subpopulation is distributed from $f_k(.|\alpha_k)$

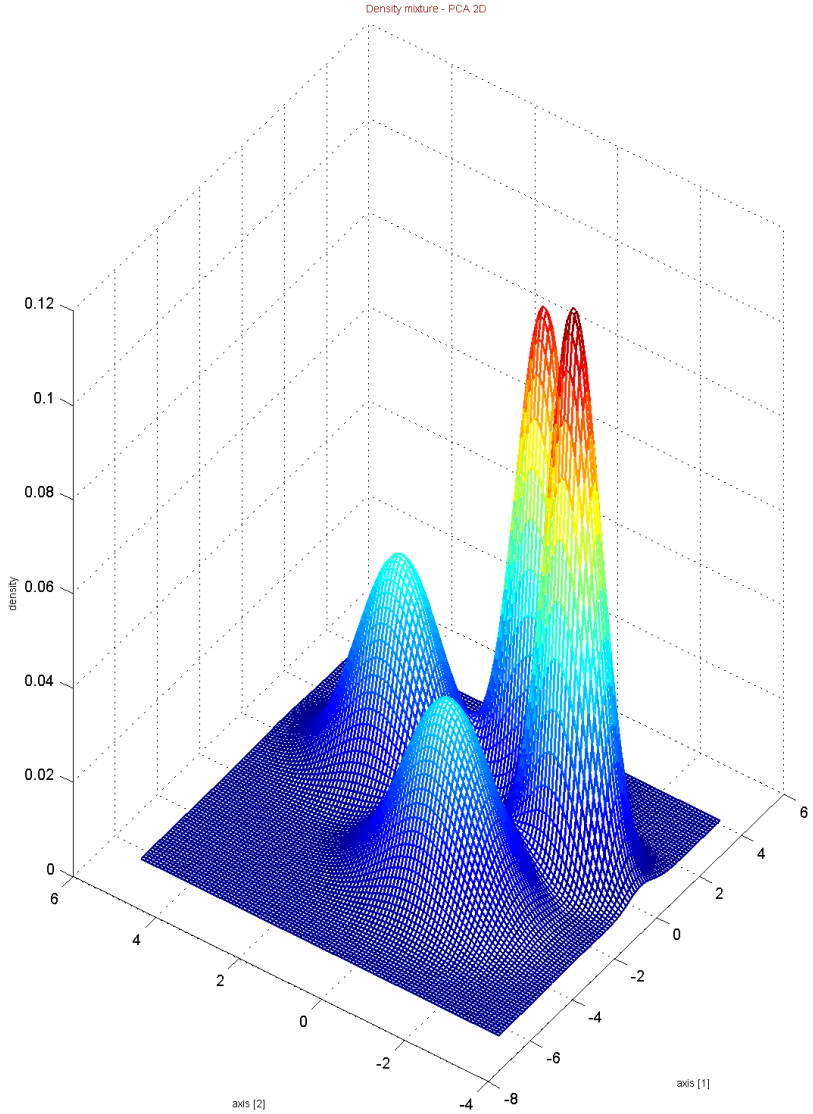
  $\Rightarrow$ Thus the population is distributed from a mixture of these subdistributions

$$f(.|\theta_K) = \sum_{k=1}^{K} \pi_k f_k(.|\alpha_k) \text{ with } (\pi_1, \ldots, \pi_K) \in]0,1[^K, \sum_{k=1}^{K} \pi_k = 1$$

Parameter vector: $\theta_K = (\pi_1, \ldots, \pi_K, \alpha_1, \ldots, \alpha_K)$

Model-based clustering





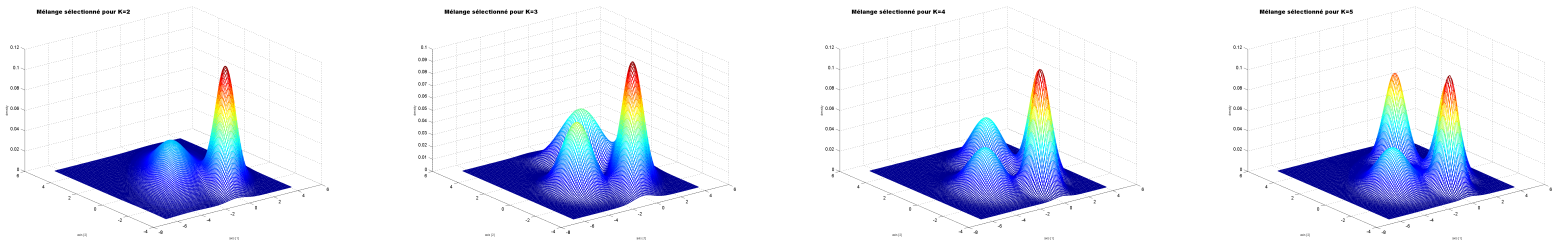
Model-based clustering

- Model collection:

$$\forall K \in \mathbb{N}^\star,\ \mathcal{S}_K = \left\{ x \in \mathbb{R}^p \mapsto f(x|\theta_K) = \sum_{k=1}^{K} \pi_k f_k(.|\alpha_k) \right\}$$

$\Rightarrow$ Choice of the model collection

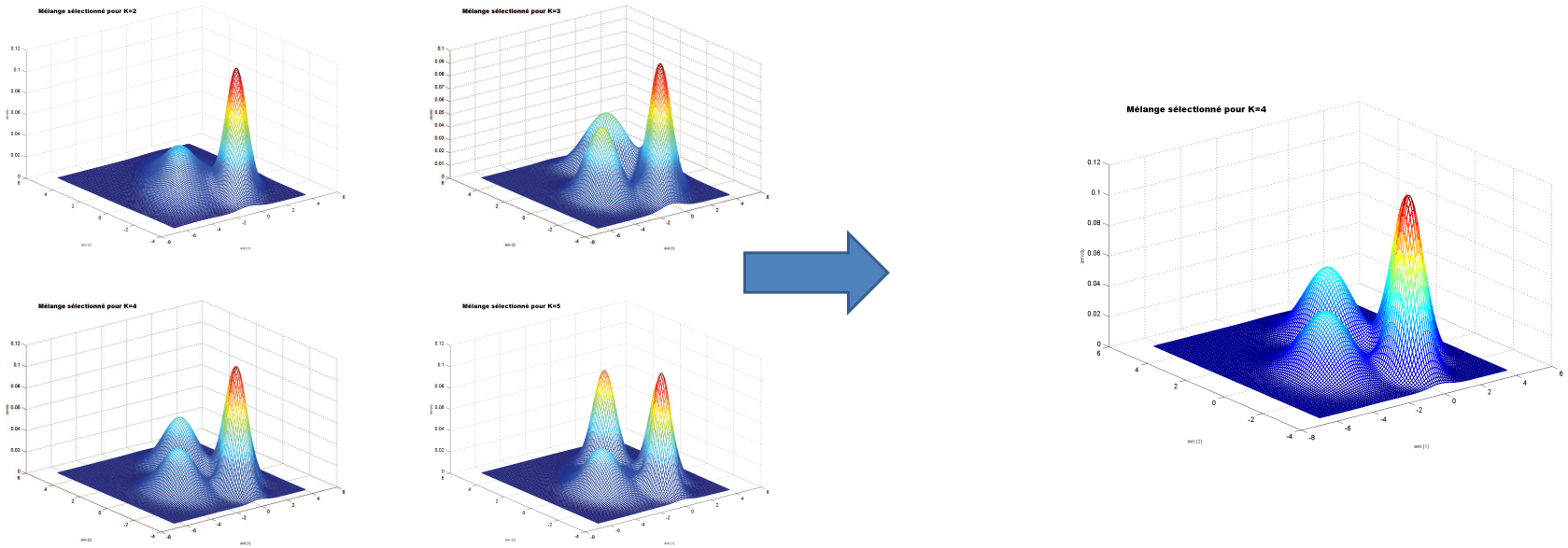
- For each model $\mathcal{S}_K$, the mixture is determined which best fits the data: $f(.|\hat{\theta}_K)$



$\Rightarrow$ Need a parameter estimation algorithm ($\hat{\theta}_K$)

Model-based clustering

- Choose the "best" mixture among $f(.|\hat{\theta}_2), f(.|\hat{\theta}_3), \ldots, f(.|\hat{\theta}_{K_{\max}})$



$\Rightarrow$ Need a model selection criterion to determine $\hat{K}$ (and $f(.|\hat{\theta}_{\hat{K}})$).

- Clustering : Maximum A Posteriori (MAP) rule
  A cell is assigned in the cluster for which it has the highest probability of belonging.

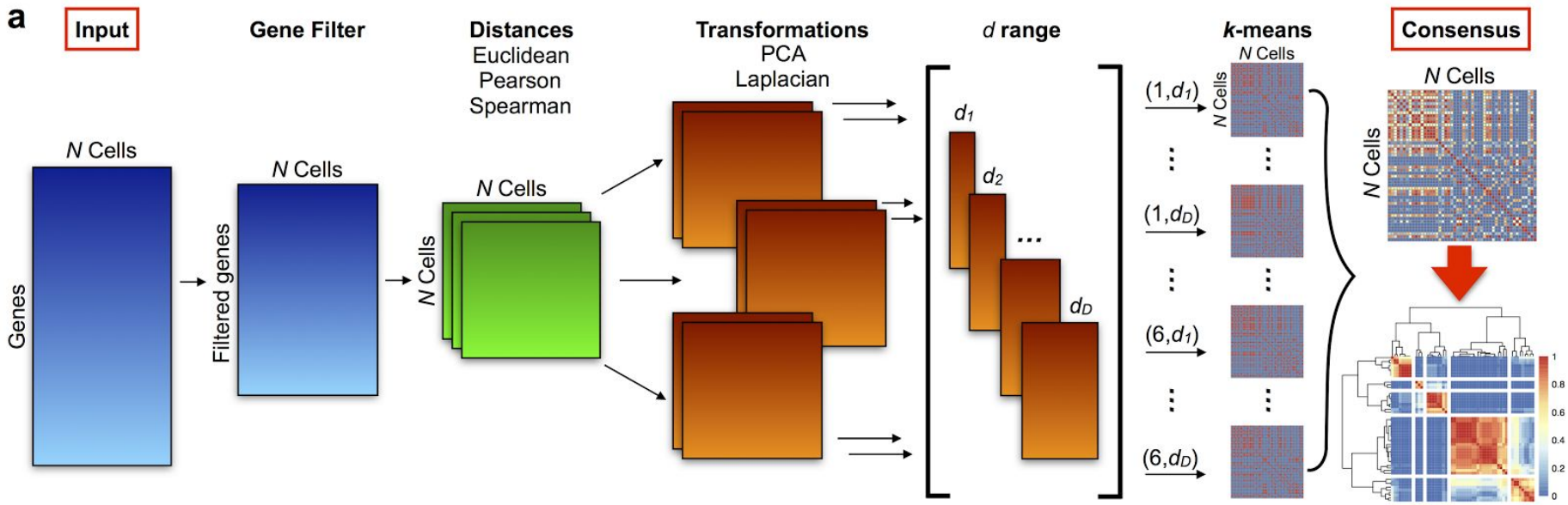
Some clustering softwares

Software	Description	Ref
SC3	Kmeans on different dimensions of PCA (on log-counts) + CAH on consensus matrix https://bioconductor.org/packages/release/bioc/html/SC3.html	[Kiselev et al., 2017]
pcaReduce	Combines PCA on log-counts, k-means and "iterative" hierarchical clustering https://github.com/JustinaZ/pcaReduce	[Žurauskienė and Yau, 2016]
SINCERA	CAH using Pearson corr. and average linkage on z-score (prelim. data transf.) https://research.cchmc.org/pbge/sincera.html	[Guo et al., 2015]
SNN-Cliq	Shared Nearest Neighbours graph + clique method http://bioinfo.uncc.edu/SNNCliq/	[Xu and Su, 2015]
Seurat	PCA + Nearest Neighbor graph clustering https://github.com/satijalab/seurat	[Butler et al., 2018]
CIDR	PCA (diss. CIDR on counts, dropouts imputation) + CAH https://github.com/VCCRI/CIDR	[Lin et al., 2017]
SAFE	Ensemble clustering using SC3, CIDR, Seurat et t-SNE Kmeans https://github.com/yycunc/SAFEclustering	[Yang et al., 2017]
BISCUIT	Dropout imputation and clustering. mixtures of multivariate log-normal and Bayesian inference https://github.com/sandhya212/BISCUIT_SingleCell_IMM_ICML_2016	[Prabhakaran et al., 2016]
SIMLR	Kernel-based similarity learning (S) + t-SNE on S + Kmeans https://bioconductor.org/packages/release/bioc/html/SIMLR.html	[Wang et al., 2017]
RaceID	Kmedoids clustering on similarity matrix of Pearson's correlation coeff. https://cran.r-project.org/web/packages/RaceID/	[Grün et al., 2015]
backSpin	Biclustering based on sorting points into neighborhoods (SPIN) https://github.com/linnarsson-lab/BackSPIN	[Zeisel et al., 2015]

▪ ▪ ▪

Reviews : [Duò et al., 2018], [Freytag et al., 2018]

SC3



[Kiselev et al., 2017]

https://bioconductor.org/packages/release/bioc/html/SC3.html

pcaReduce

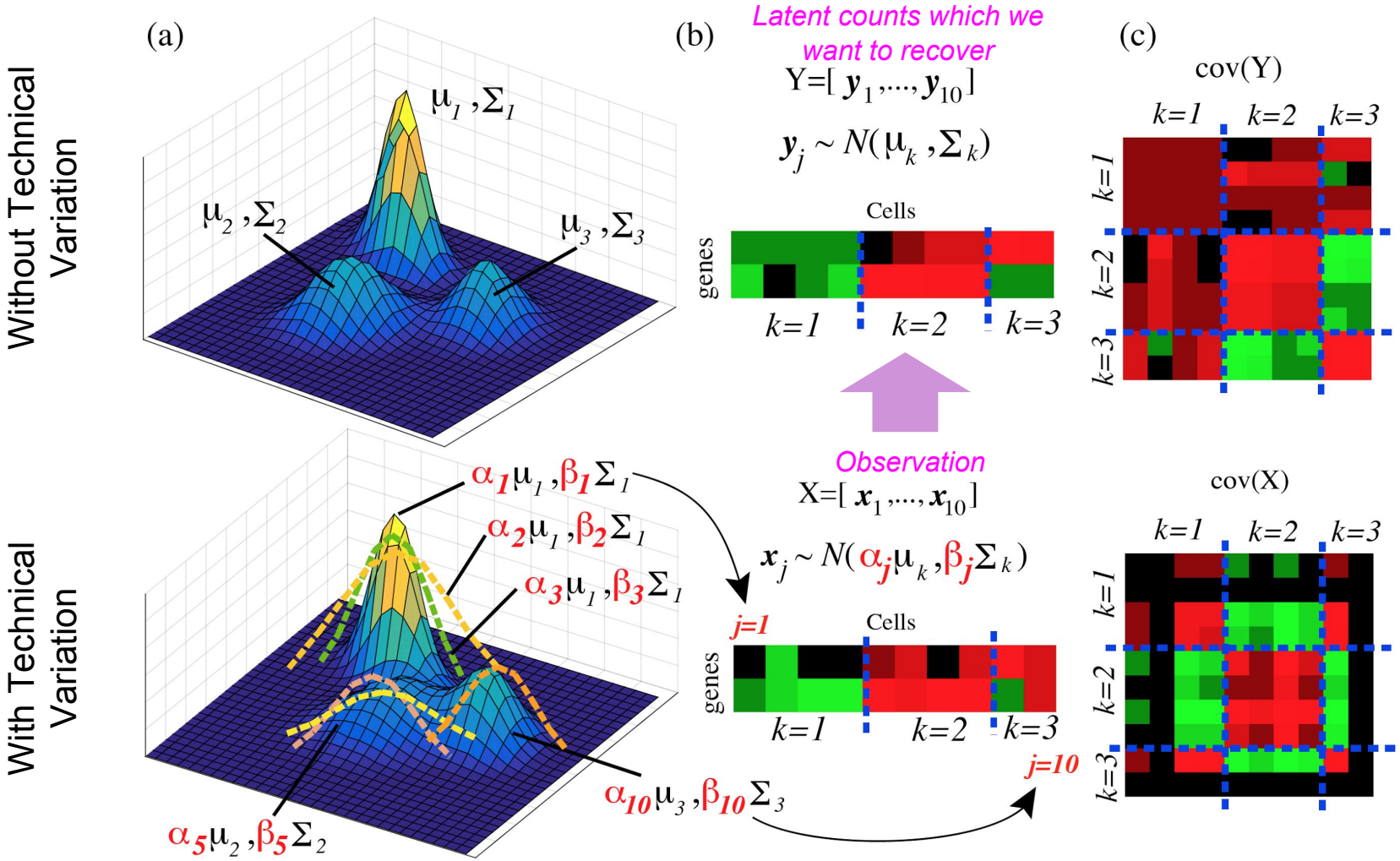
Algorithm 1: The *pcaReduce* algorithm. Here $\mathbf{X}_{n\times d}$ is a gene expression matrix with n cells (given in rows) and d genes (in columns); q is the number of dimensions – effectively this refers to the number of levels in the hierarchy; Y is a score matrix, which is the output of PCA algorithm; μ_{ij} and Σ_{ij} definition are given in Eq. (1); (i) and (ii) denote two different merging settings: (i) merging is based on largest probability $P(i,j)$ value; (ii) merging is based on sampling according to $P(i,j)$ distribution.

Input: $\mathbf{X}_{n\times d}$ and q ;
Output: a collection of q clusterings;
$Y \longleftarrow \texttt{PCA}(X_{n\times d}, \text{dim}{=}q)$;
$(\mu, \Sigma) \longleftarrow \texttt{kmeans}(Y,\ K = q+1)$;
$Q \longleftarrow q-1$;
for $r = 1, \ldots, Q$ **do**
  for *all possible pairs* (i,j) *in* $\{1,\ldots,K\} \times \{1,\ldots,K\}$ **do**
    $P(i,j) \longleftarrow p\left(Y_i \cup Y_j | \mu_{ij}, \Sigma_{ij}\right)$;
  end for
  (i) either choose pair (i,j) with largest probability $P(i,j)$ and merge clusters i and j;
  (ii) or sample a pair (i,j) with probability $P(i,j)$ and merge clusters i and j;
  $q \longleftarrow q-1$;
  $Y \leftarrow Y_{n\times q}$ (i.e. remove last dimension);
  update (μ, Σ);
  $K \leftarrow K-1$;
end for

[Žurauskienė and Yau, 2016]

https://github.com/JustinaZ/pcaReduce

BISCUIT

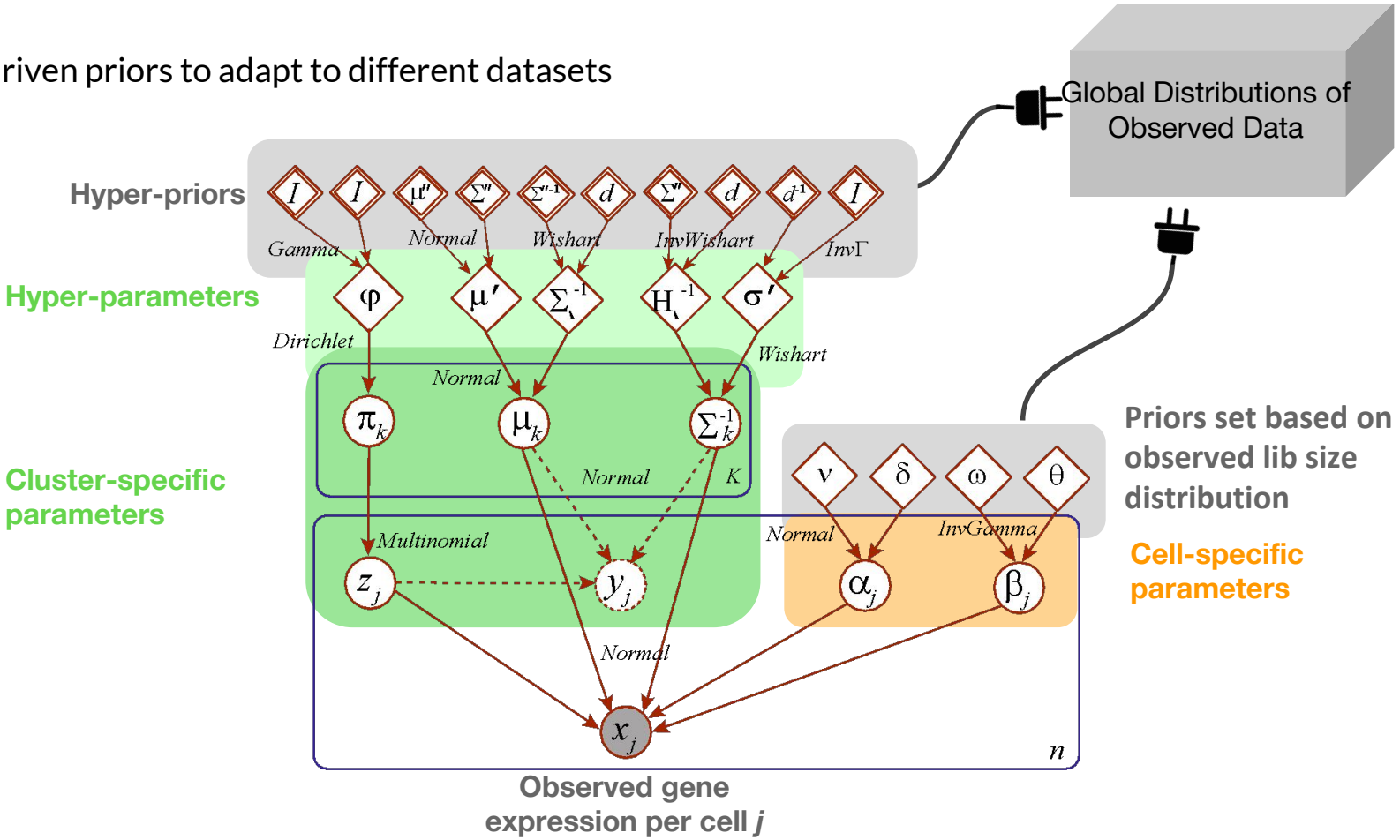


[Prabhakaran et al., 2016]

https://github.com/sandhya212/BISCUIT_SingleCell_IMM_ICML_2016

BISCUIT

Data-driven priors to adapt to different datasets



[Prabhakaran et al., 2016]

https://github.com/sandhya212/BISCUIT_SingleCell_IMM_ICML_2016

SAFE

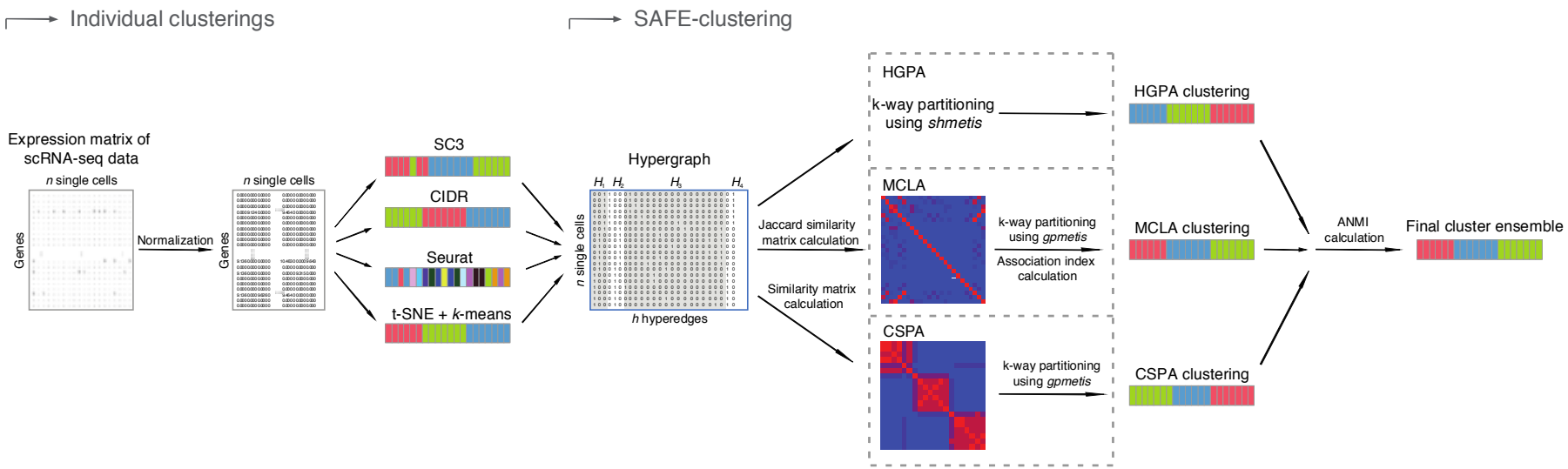


Figure 1. Overview of SAFE-clustering. Log-transformed expression matrix of scRNA-seq data are first clustered using four state-of-the-art methods, SC3, CIDR, Seurat and t-SNE + *k*-means; and then individual solutions are combined using one of the three hypergraph-based partitioning algorithms: hypergraph partitioning algorithm (HGPA), meta-cluster algorithm (MCLA) and cluster-based similarity partitioning algorithm (CSPA) to produce consensus clustering.

[Yang et al., 2017]

https://github.com/yycunc/SAFEclustering

Remarks

- What clustering method to choose?
- What is meant by "similar cells"?
- How to choose the number of clusters?
- Consistency between several methods, consensus of clusterings
- Is it important to cluster all the cells? Fuzzy clustering versions
- Ideas: biclustering, clustering + gene selection, ...

Plan

1 Introduction

2 Feature selection / extraction

3 Dimension reduction

4 Single cell clustering

5 **Pseudotime analysis**

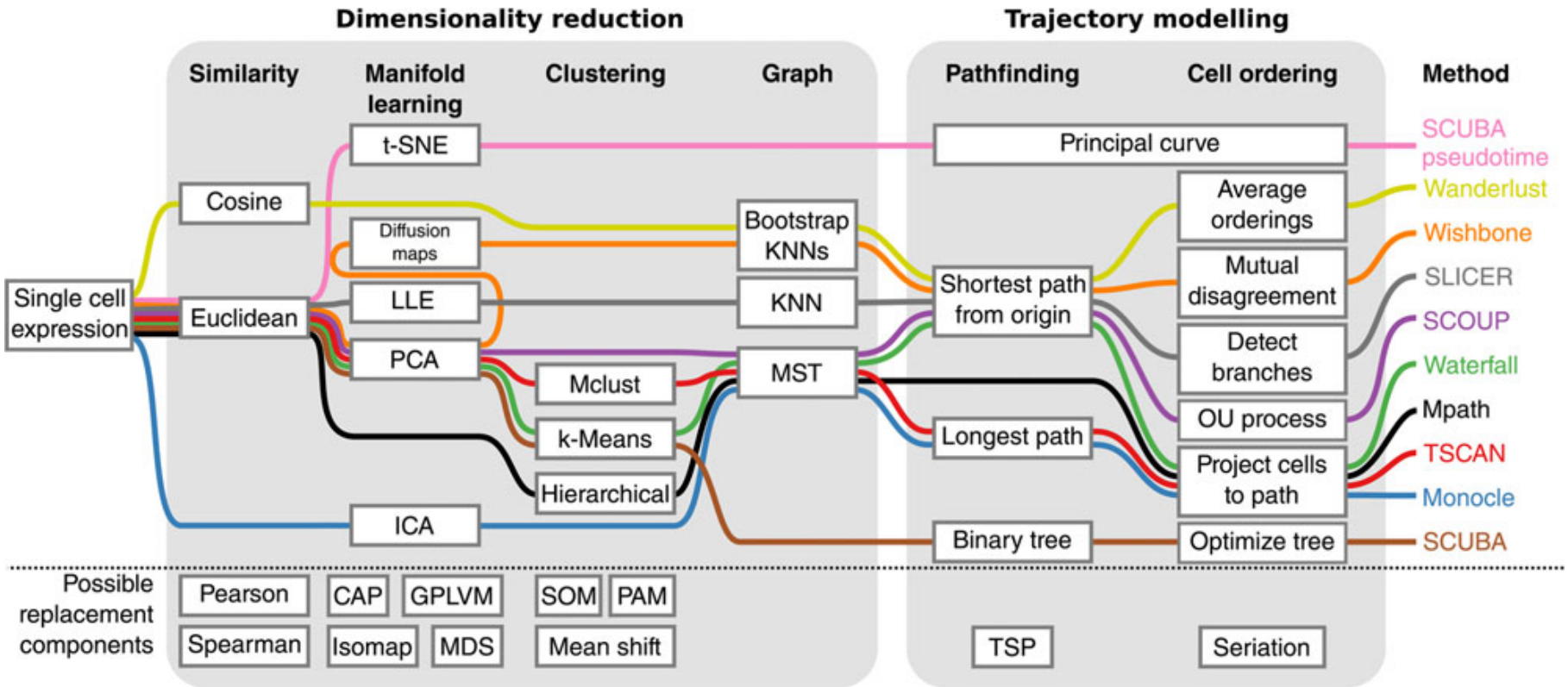
6 Differential analysis

Pseudotime analysis

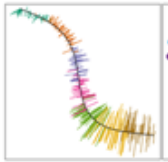
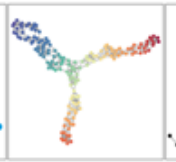
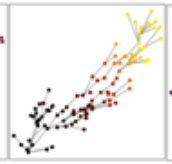
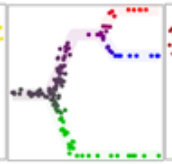
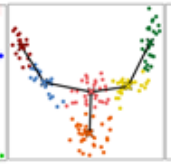
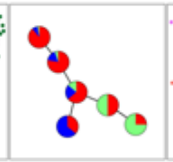
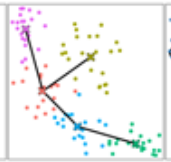
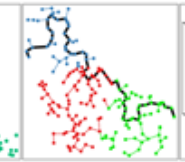
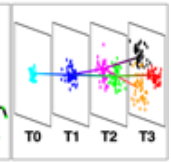
- Trajectory inference aims to reconstruct a cellular dynamic process
- 2 main steps:
  - Dimensionality reduction step: PCA, t-SNE, ... or graph-based techniques .... or clustering methods
  - Trajectory modelling step
- Reviews:
  - [Cannoodt et al., 2016]: comparison of 10 methods
  - [Saelens et al., 2018]: comparison of 29 (among 59 existing) methods

`https://github.com/dynverse/dynverse`

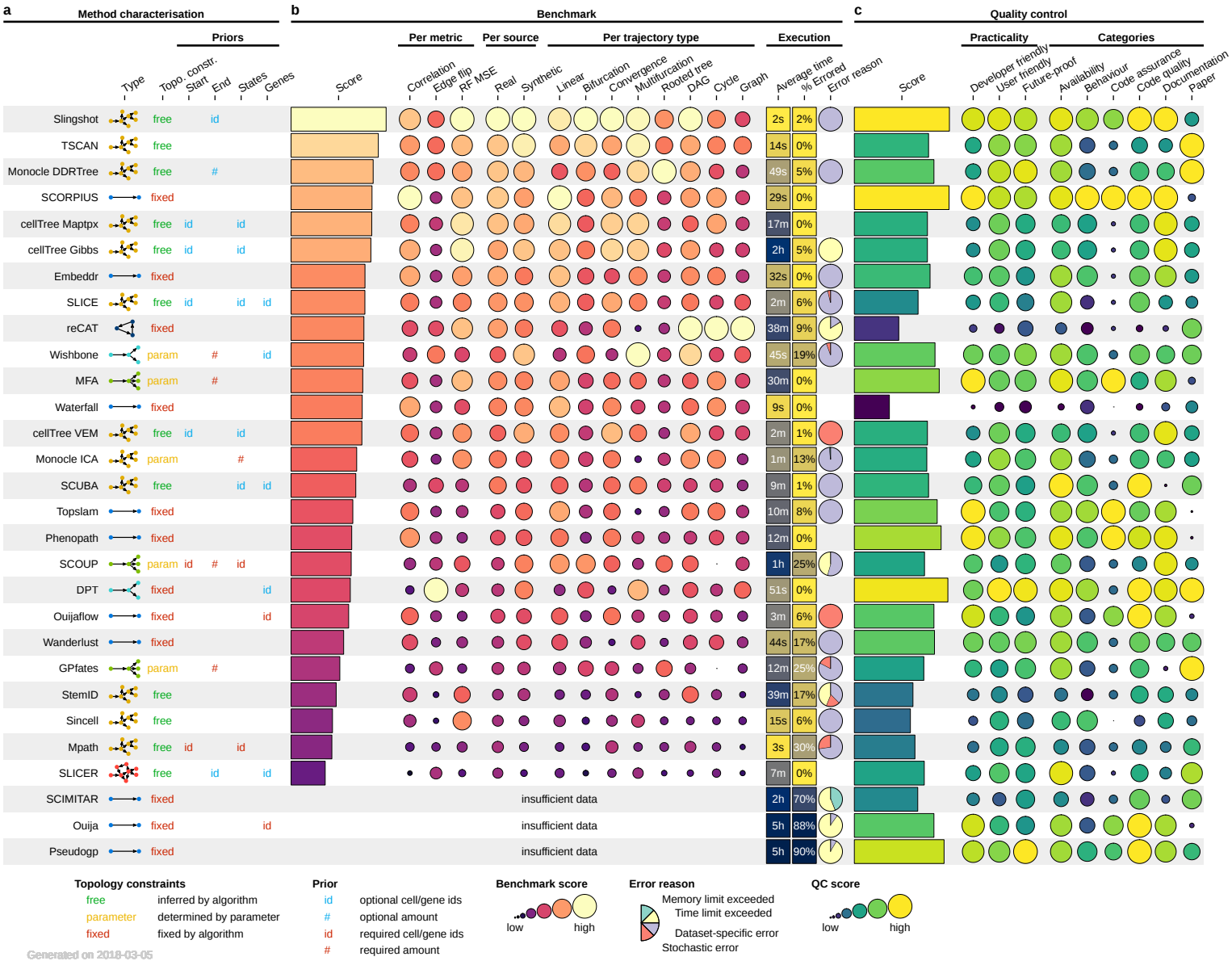
Pseudotime analysis



Pseudotime analysis

Method	SCUBA pseudotime	Wanderlust	Wishbone	SLICER	SCOUP	Waterfall	Mpath	TSCAN	Monocle	SCUBA
Visual abstract										
Structure	Linear	Linear	Single bifurcation	Branching	Branching	Linear	Branching	Linear	Branching	Branching
Robustness strategy	Principal curves	Ensemble, starting cell	Ensemble, starting cell	Starting cell	Starting population	Clustering of cells	Clustering of cells using external labelling	Clustering of cells	Differential expression	Simple model
Extra input requirements	None	Starting cell	Starting cell	Starting cell	Starting population	None	Time points	None	Time points	Time points
Unbiased	+	±	±	±	±	+	−	+	−	−
Scalability w.r.t. cells	−	−	±	±	−	±	+	+	−	±
Scalability w.r.t. genes	+	+	+	+	−	+	±	±	±	+
Code and documentation	−	±	+	±	+	±	+	+	+	±
Parameter ease-of-use	+	+	+	+	−	±	−	+	+	+

Pseudotime analysis



Plan

1. Introduction
2. Feature selection / extraction
3. Dimension reduction
4. Single cell clustering
5. Pseudotime analysis
6. **Differential analysis**

Methods for differential analysis

Review: [Soneson and Robinson, 2018]

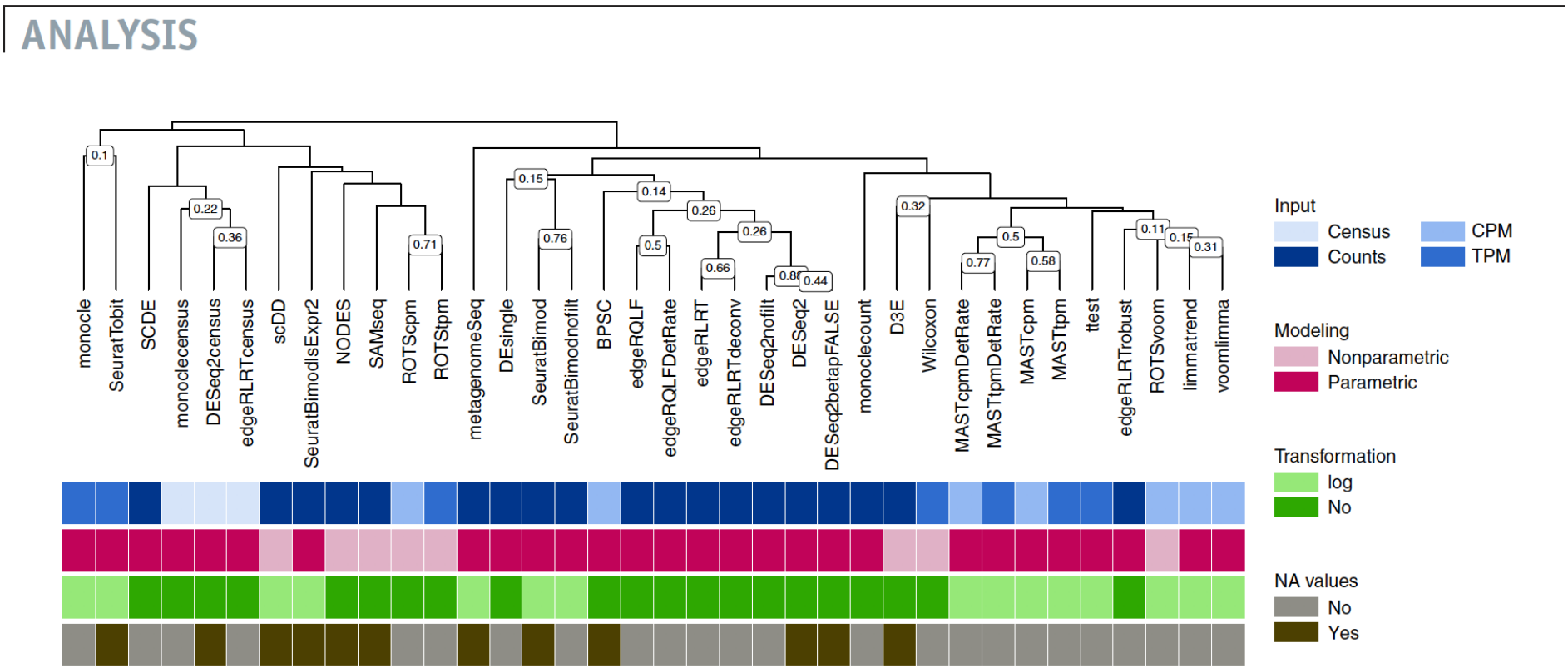


Figure 3 | Average similarities between gene rankings obtained by the evaluated DE methods. The dendrogram was obtained by complete-linkage hierarchical clustering based on the matrix of average AUCC values across all data sets. The labels of the internal nodes represent their stability across data sets (fraction of instances where they are observed). Only nodes with stability scores of at least 0.1 are labeled. Colored boxes represent method characteristics.

Methods for differential analysis

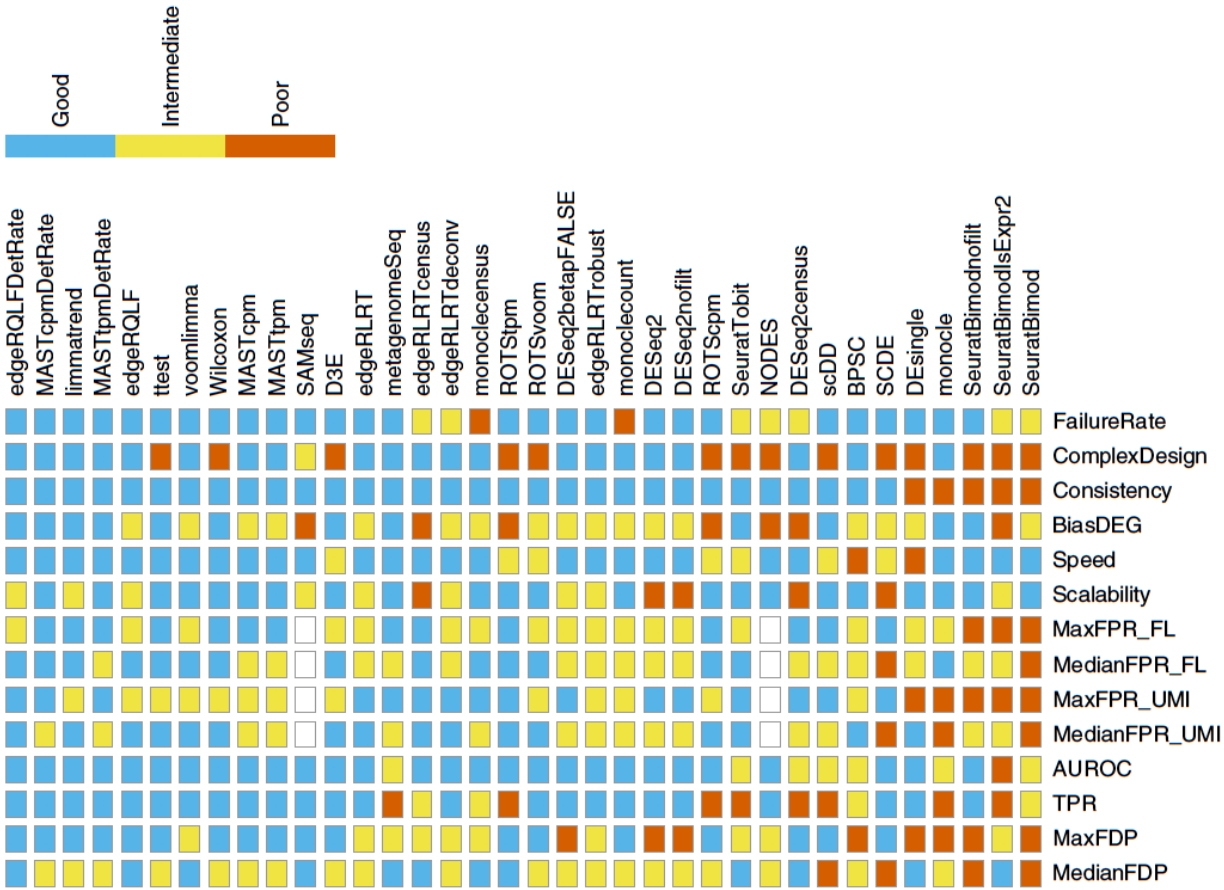


Figure 5 | Summary of DE method performance across all major evaluation criteria. Criteria and cutoff values for performance categories are available in the Online Methods. Methods are ranked by their average performance across the criteria, with the numerical encoding good = 2, intermediate = 1, poor = 0. NODES and SAMseq do not return nominal *P* values and were therefore not evaluated in terms of the FPR.

Some other interesting resources

- List of software packages for single-cell data analysis, including RNA-seq: https://github.com/seandavi/awesome-single-cell
- Course of Hemberg's lab:


Analysis of single cell RNA-seq data

Vladimir Kiselev (wikiselev), Tallulah Andrews, Jennifer Westoby (Jenni_Westoby), Davis McCarthy (davisjmcc), Maren Büttner (marenbuettner) and Martin Hemberg (m_hemberg)

2018-05-29

https://hemberg-lab.github.io/scRNA.seq.course/index.html

References I


Andrews, T. S. and Hemberg, M. (2018).
Identifying cell populations with scrnaseq.
Molecular aspects of medicine, 59:114–122.


Brennecke, P., Anders, S., Kim, J. K., Kołodziejczyk, A. A., Zhang, X., Proserpio, V., Baying, B., Benes, V., Teichmann, S. A., Marioni, J. C., et al. (2013).
Accounting for technical noise in single-cell rna-seq experiments.
Nature methods, 10(11):1093.


Butler, A., Hoffman, P., Smibert, P., Papalexi, E., and Satija, R. (2018).
Integrating single-cell transcriptomic data across different conditions, technologies, and species.
Nature biotechnology, 36(5):411.


Cannoodt, R., Saelens, W., and Saeys, Y. (2016).
Computational methods for trajectory inference from single-cell transcriptomics.
European journal of immunology, 46(11):2496–2506.


Duò, A., Robinson, M. D., and Soneson, C. (2018).
A systematic performance evaluation of clustering methods for single-cell rna-seq data.
F1000Research, 7.


Freytag, S., Tian, L., Lönnstedt, I., Ng, M., and Bahlo, M. (2018).
Comparison of clustering tools in r for medium-sized 10x genomics single-cell rna-sequencing data.
F1000Research, 7.


Grün, D., Lyubimova, A., Kester, L., Wiebrands, K., Basak, O., Sasaki, N., Clevers, H., and van Oudenaarden, A. (2015).
Single-cell messenger rna sequencing reveals rare intestinal cell types.
Nature, 525(7568):251.


Guo, M., Wang, H., Potter, S. S., Whitsett, J. A., and Xu, Y. (2015).
Sincera: a pipeline for single-cell rna-seq profiling analysis.
PLoS computational biology, 11(11):e1004575.

References II

Juliá, M., Telenti, A., and Rausell, A. (2015).

Sincell: an r/bioconductor package for statistical assessment of cell-state hierarchies from single-cell rna-seq.
Bioinformatics, 31(20):3380–3382.

Kim, J. K., Kolodziejczyk, A. A., Ilicic, T., Teichmann, S. A., and Marioni, J. C. (2015).

Characterizing noise structure in single-cell rna-seq distinguishes genuine from technical stochastic allelic expression.
Nature communications, 6:8687.

Kiselev, V. Y., Kirschner, K., Schaub, M. T., Andrews, T., Yiu, A., Chandra, T., Natarajan, K. N., Reik, W., Barahona, M., Green, A. R., and Hemberg, M. (2017).

Sc3: consensus clustering of single-cell rna-seq data.
Nature Methods, 14:483 EP –.

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., McCarthy, D. J., and Marioni, J. C. (2016).

A step-by-step workflow for low-level analysis of single-cell rna-seq data with bioconductor.
F1000Research, 5.

Pierson, E. and Yau, C. (2015).

Zifa: Dimensionality reduction for zero-inflated single-cell gene expression analysis.
Genome biology, 16(1):241.

Poirion, O. B., Zhu, X., Ching, T., and Garmire, L. (2016).

Single-cell transcriptomics bioinformatics and computational challenges.
Frontiers in genetics, 7:163.

References III

Prabhakaran, S., Azizi, E., Carr, A., and Pe'er, D. (2016).

Dirichlet process mixture model for correcting technical variation in single-cell gene expression data.
In *International Conference on Machine Learning*, pages 1070–1079.

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.

Saelens, W., Cannoodt, R., Todorov, H., and Saeys, Y. (2018).

A comparison of single-cell trajectory inference methods: towards more accurate and robust tools.
bioRxiv, page 276907.

Satija, R., Farrell, J. A., Gennert, D., Schier, A. F., and Regev, A. (2015).

Spatial reconstruction of single-cell gene expression data.
Nature biotechnology, 33(5):495.

Soneson, C. and Robinson, M. (2018).

Bias, robustness and scability in single-cell differential expression analysis.
Nature Methods, 15:255–261.

Vallejos, C., Marioni, J., and Richardson, S. (2015).

Basics: Bayesian analysis of single-cell sequencing data.
PLOS COMPUTATIONAL BIOLOGY, 11.

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.

Wolf, F. A., Angerer, P., and Theis, F. J. (2018).

Scanpy: large-scale single-cell gene expression data analysis.
Genome biology, 19(1):15.

References IV

Xu, C. and Su, Z. (2015).

Identification of cell types from single-cell transcriptomes using a novel clustering method.
Bioinformatics, 31(12):1974–1980.

Yang, Y., Huh, R., Culpepper, H. W., Lin, Y., Love, M. I., and Li, Y. (2017).

SAFE-clustering: Single-cell aggregated (from ensemble) clustering for single-cell RNA-seq data.
bioRxiv.

Zeisel, A., Muñoz-Manchado, A. B., Codeluppi, S., Lönnerberg, P., La Manno, G., Juréus, A., Marques, S., Munguba, H., He, L., Betsholtz, C., et al. (2015).

Cell types in the mouse cortex and hippocampus revealed by single-cell rna-seq.
Science, 347(6226):1138–1142.

Žurauskienė, J. and Yau, C. (2016).

pcareduce: hierarchical clustering of single cell transcriptional profiles.
BMC Bioinformatics, 17.