

# Single Cell RNA-seq Clustering

---

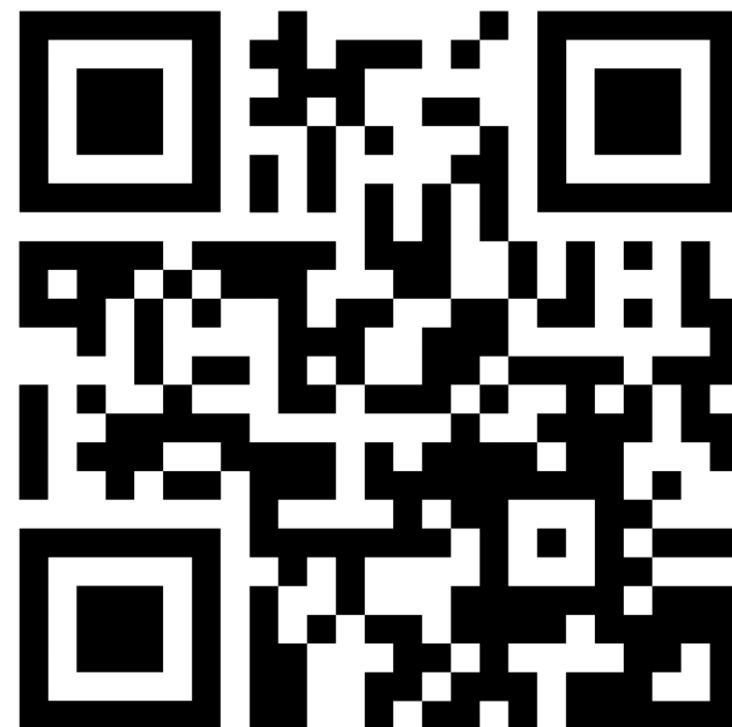
Ahmed Mahfouz

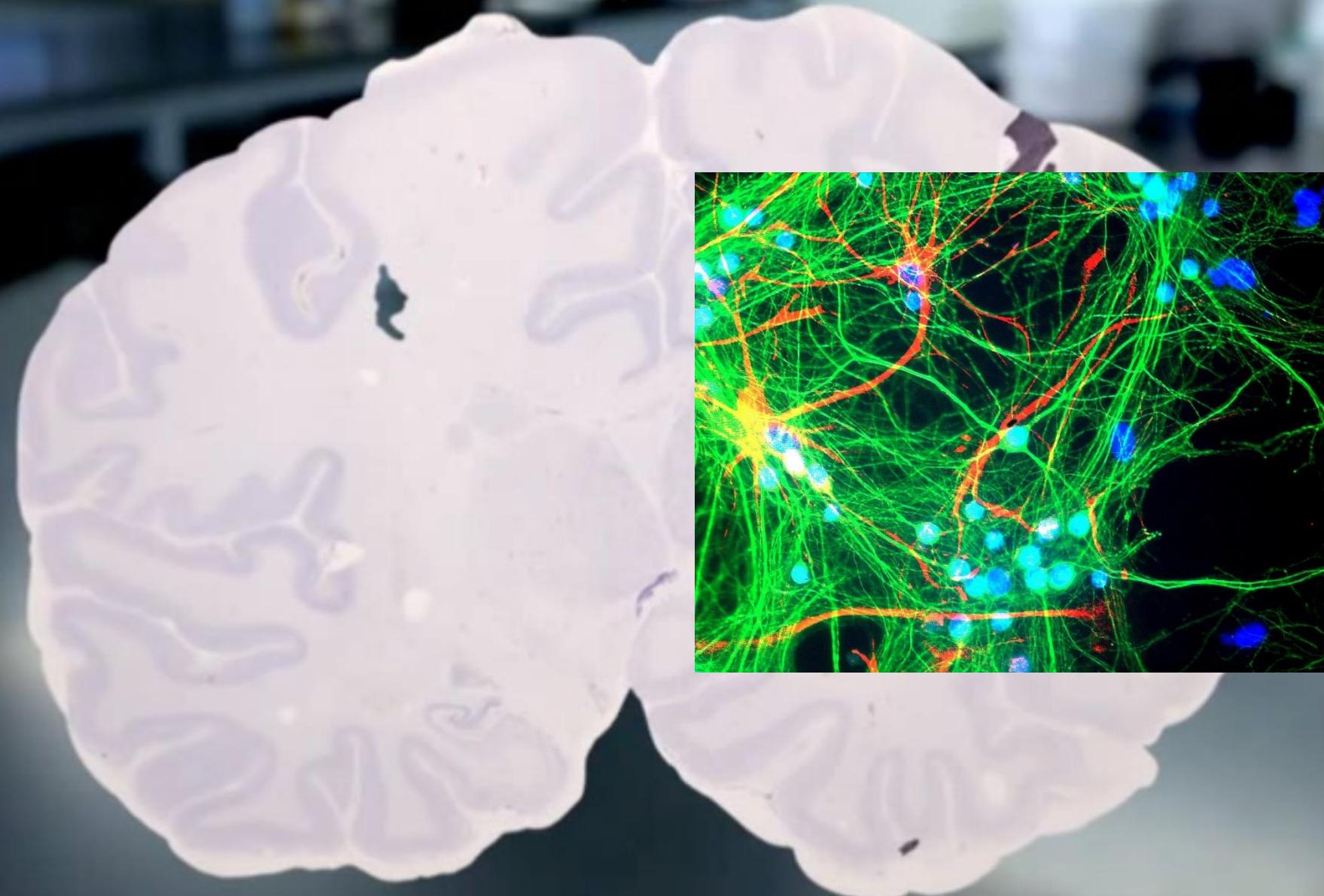
Leiden Computational Biology Center, LUMC  
Delft Bioinformatics Lab, TU Delft

BioSB Statistics for Omics – 28 June 2019

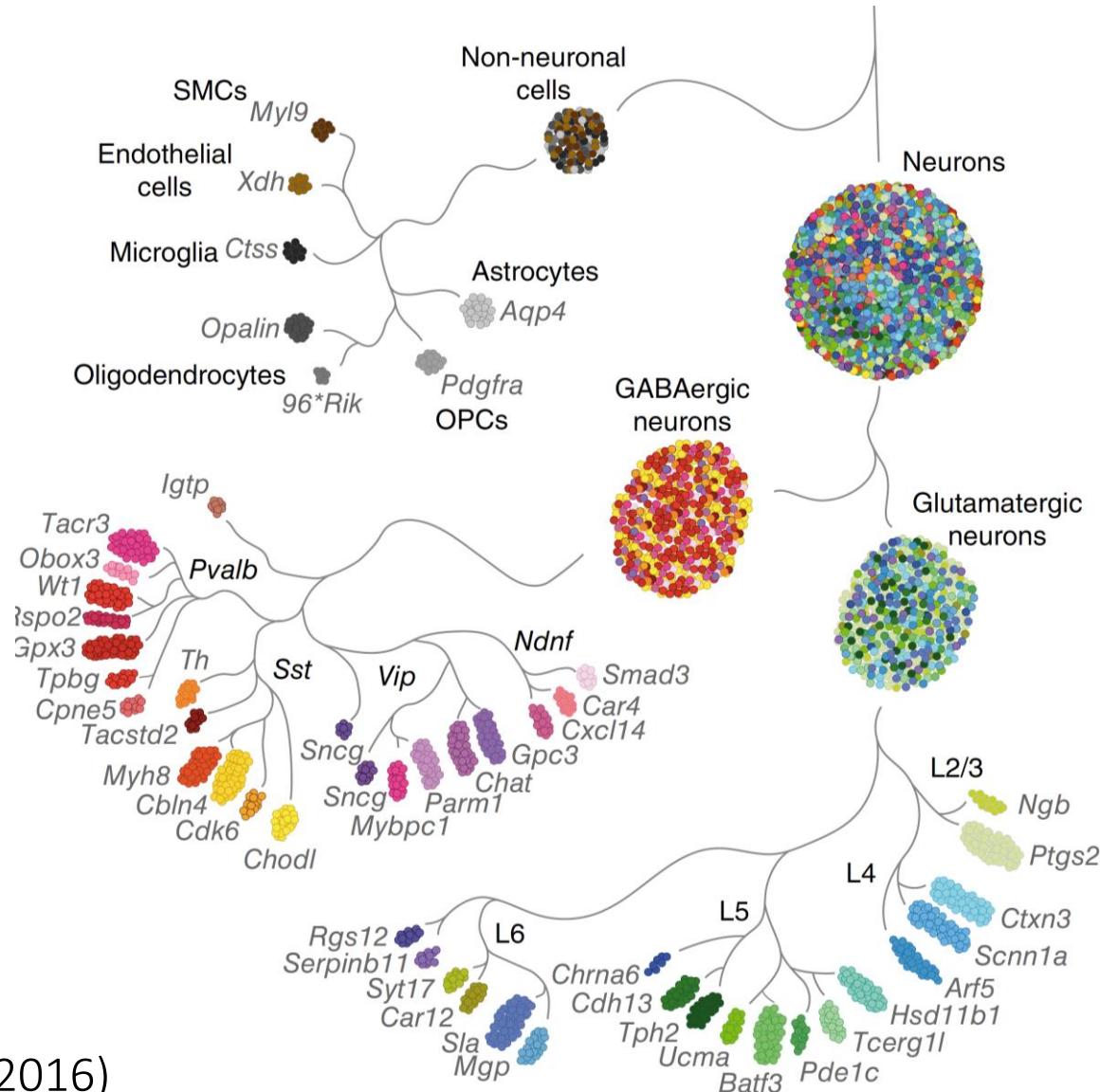
# Session materials

<https://github.com/ahmedmahfouz/BioSB-Statistics-for-Omics-2019>





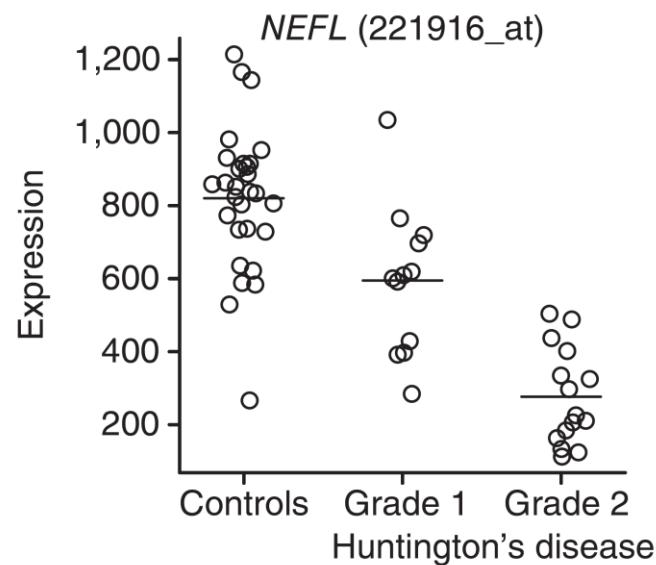
# Cell types in the brain



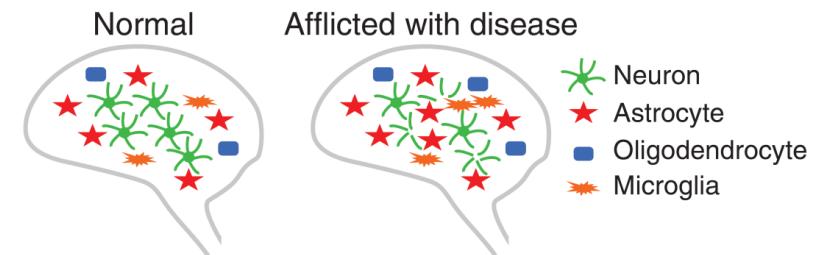
# Cellular heterogeneity in the brain

*Human Molecular Genetics*, 2006, Vol. 15, No. 6 965–977  
doi:10.1093/hmg/ddl013  
Advance Access published on February 8, 2006

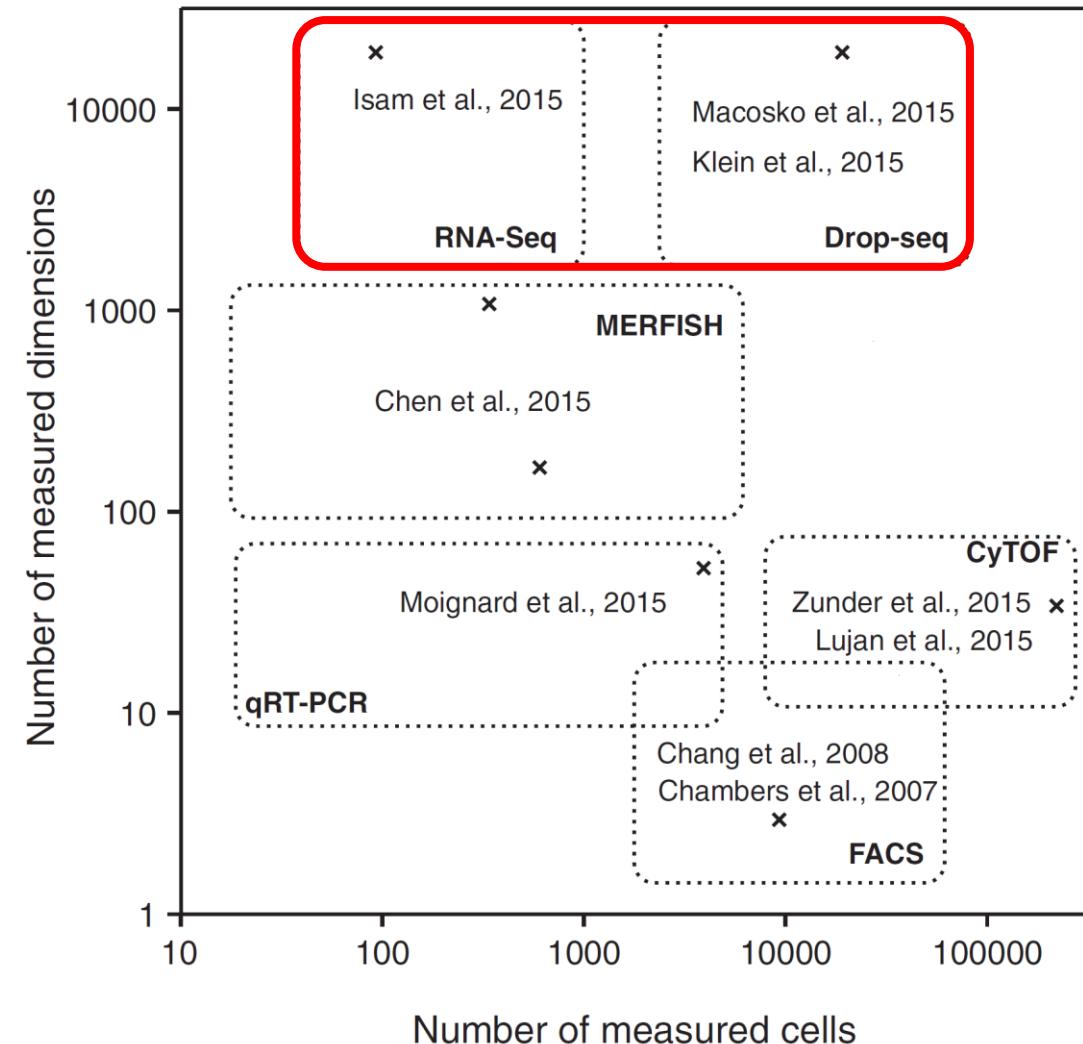
## Regional and cellular gene expression changes in human Huntington's disease brain



- 1) Decreased number of neurons?
- 2) Decreased expression?
- 3) Both?



# How can we study single cells?

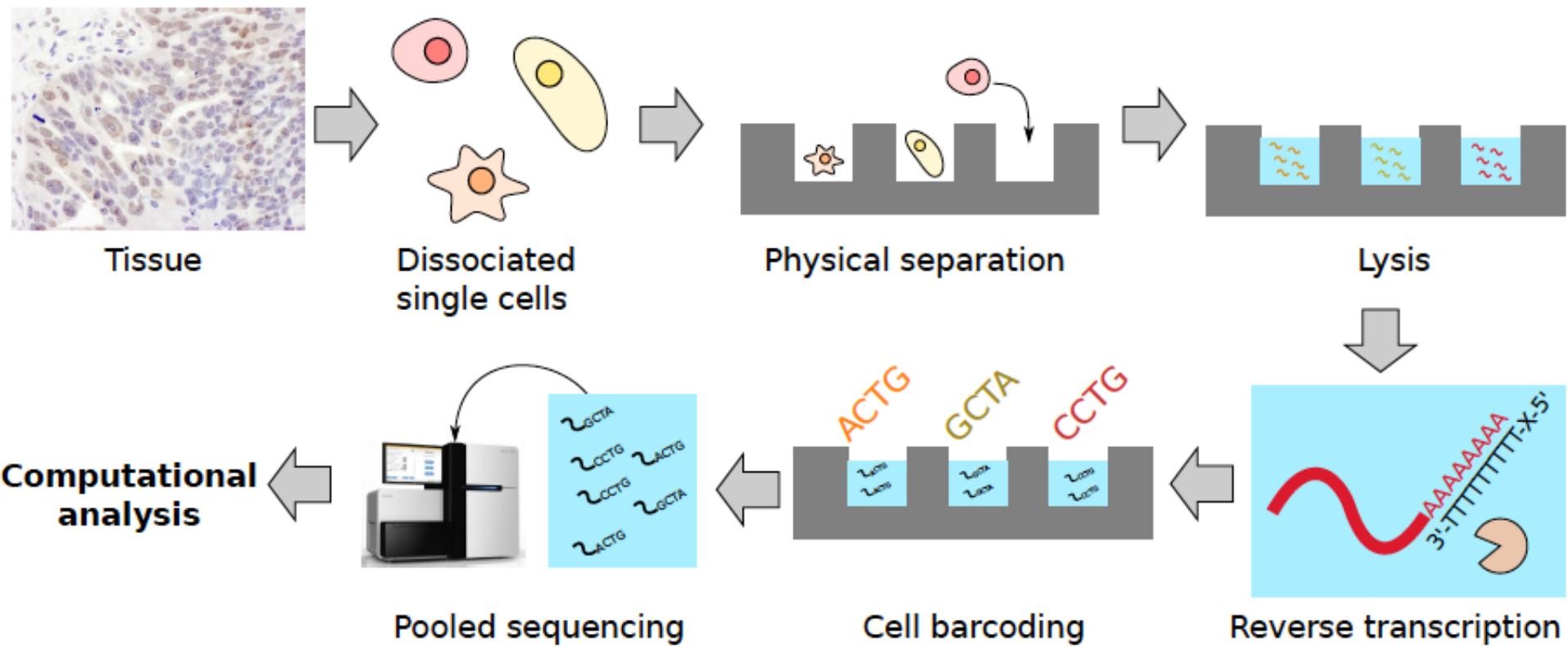


Every method has  
it's pros and cons.

# Goals of today

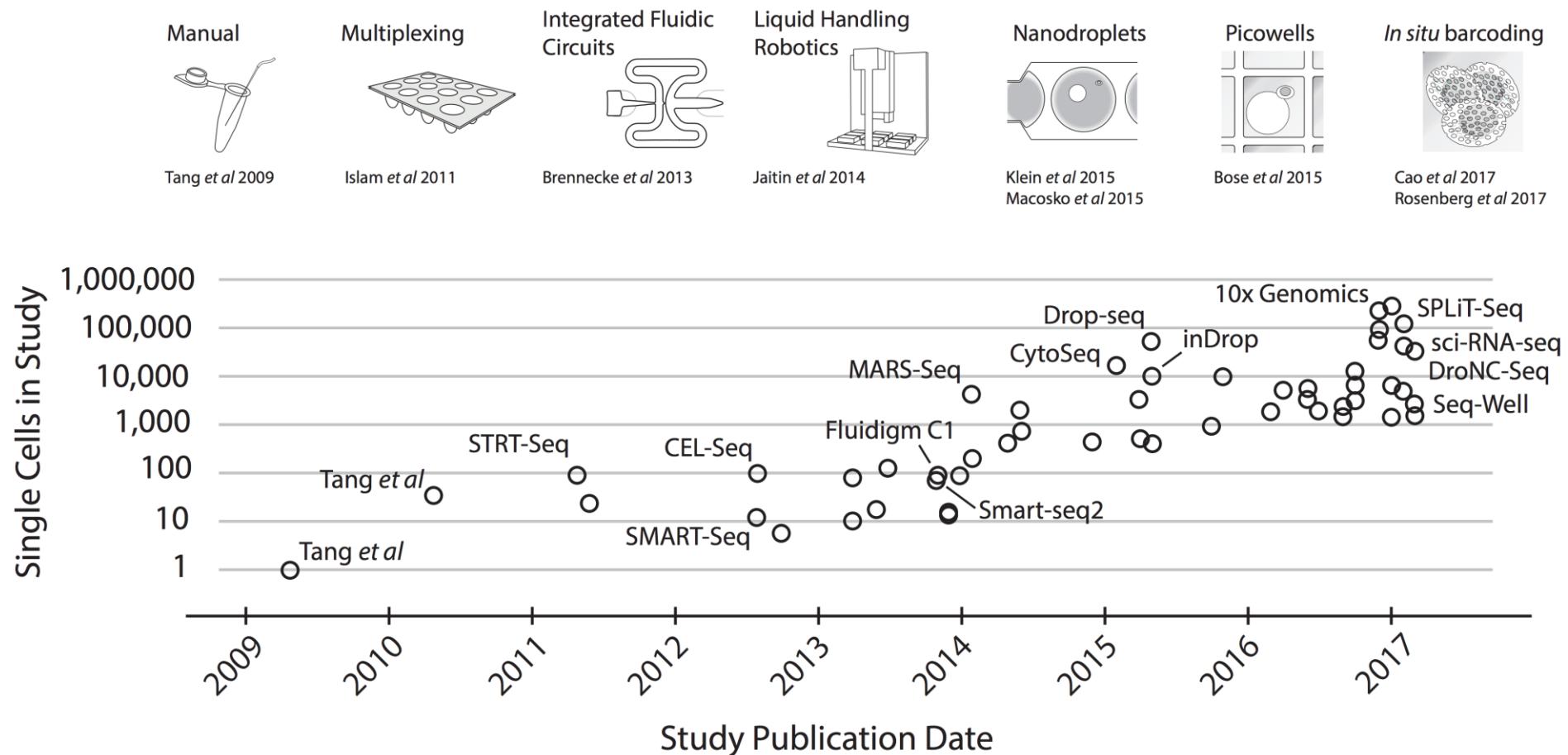
- Introduce single cell RNA-sequencing (scRNA-seq)
- Outline analysis workflow
- Go through (some) steps and discuss best practices

# Single cell RNA-sequencing (scRNA-seq)



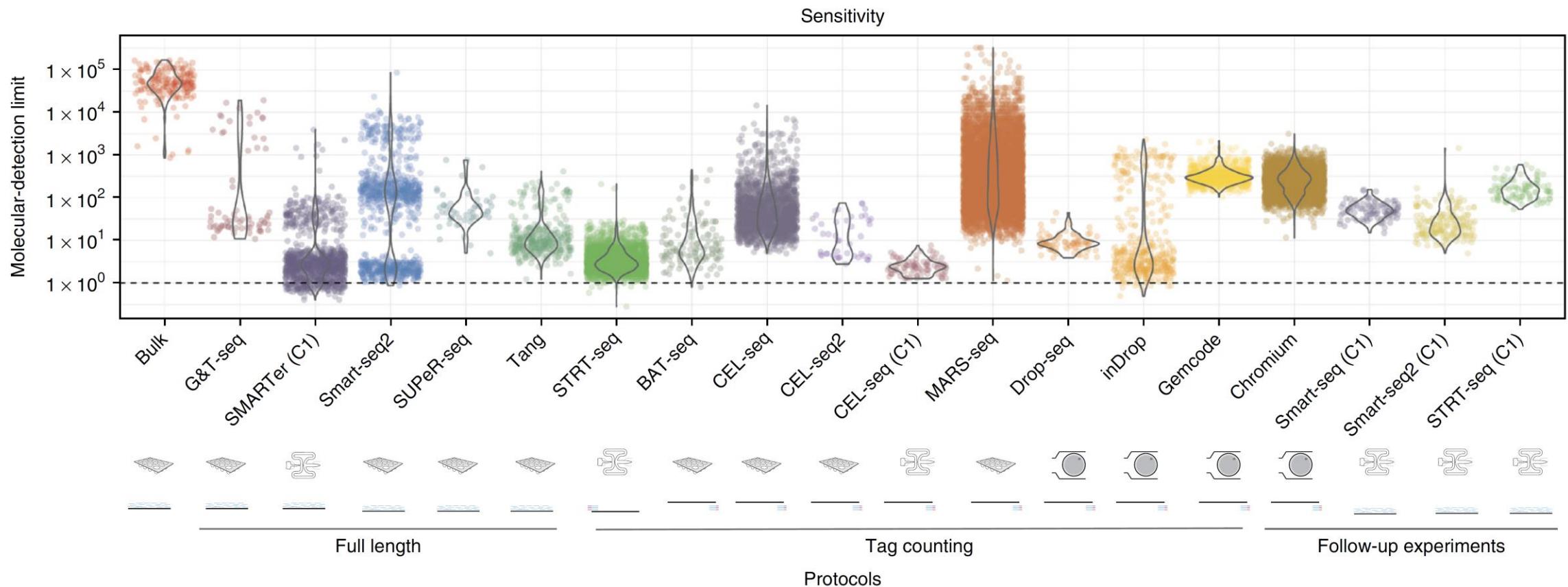
# scRNA-seq Protocols

*Number of cells*

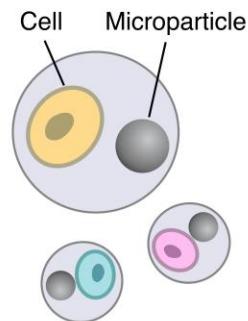
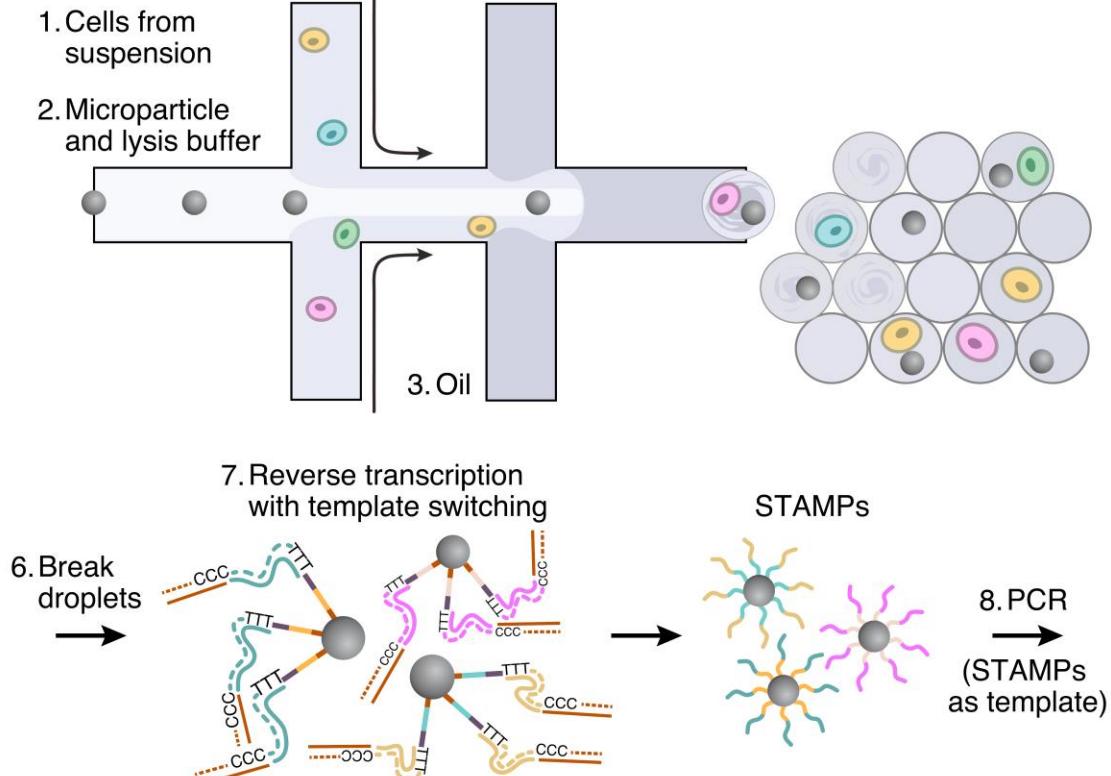


# scRNA-seq Protocols

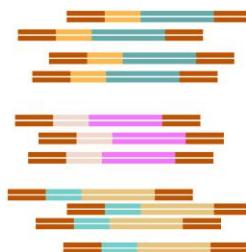
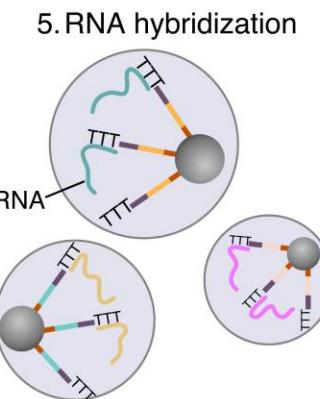
## *Sensitivity*



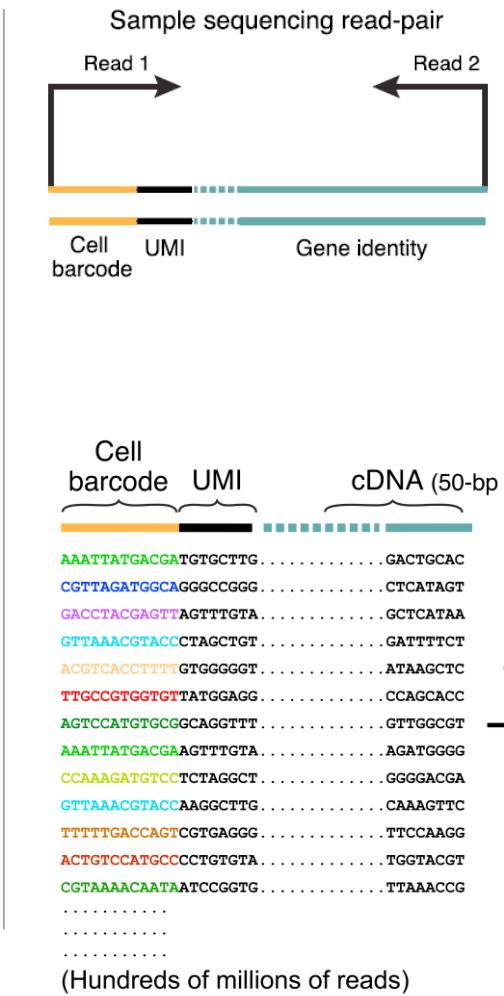
# Drop-seq



4. Cell lysis  
(in seconds)



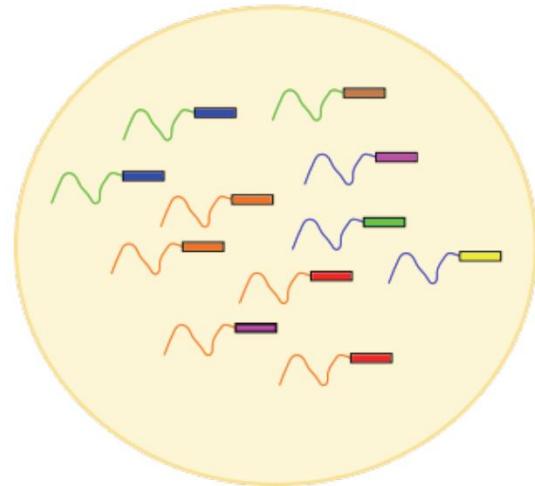
9. Sequencing and analysis
- Each mRNA is mapped to its cell-of-origin and gene-of-origin
  - Each cell's pool of mRNA can be analyzed



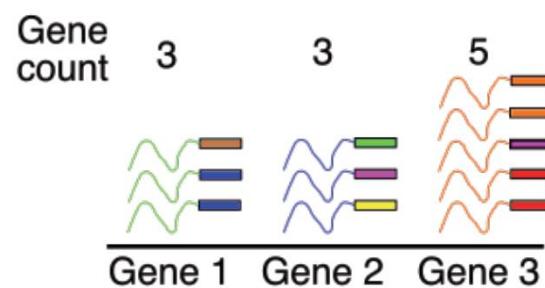
# Unique Molecular Identifiers (UMIs)

- Unique molecular identifiers give (almost) exact molecule counts in sequencing experiments.
- They reduce the amplification noise by allowing (almost) complete de-duplication of sequenced fragments.

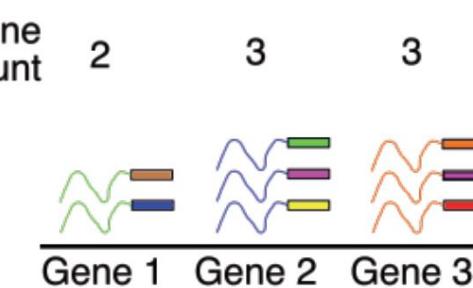
Sequenced fragments from an individual cell



Pre  
de-duplication



Post  
de-duplication

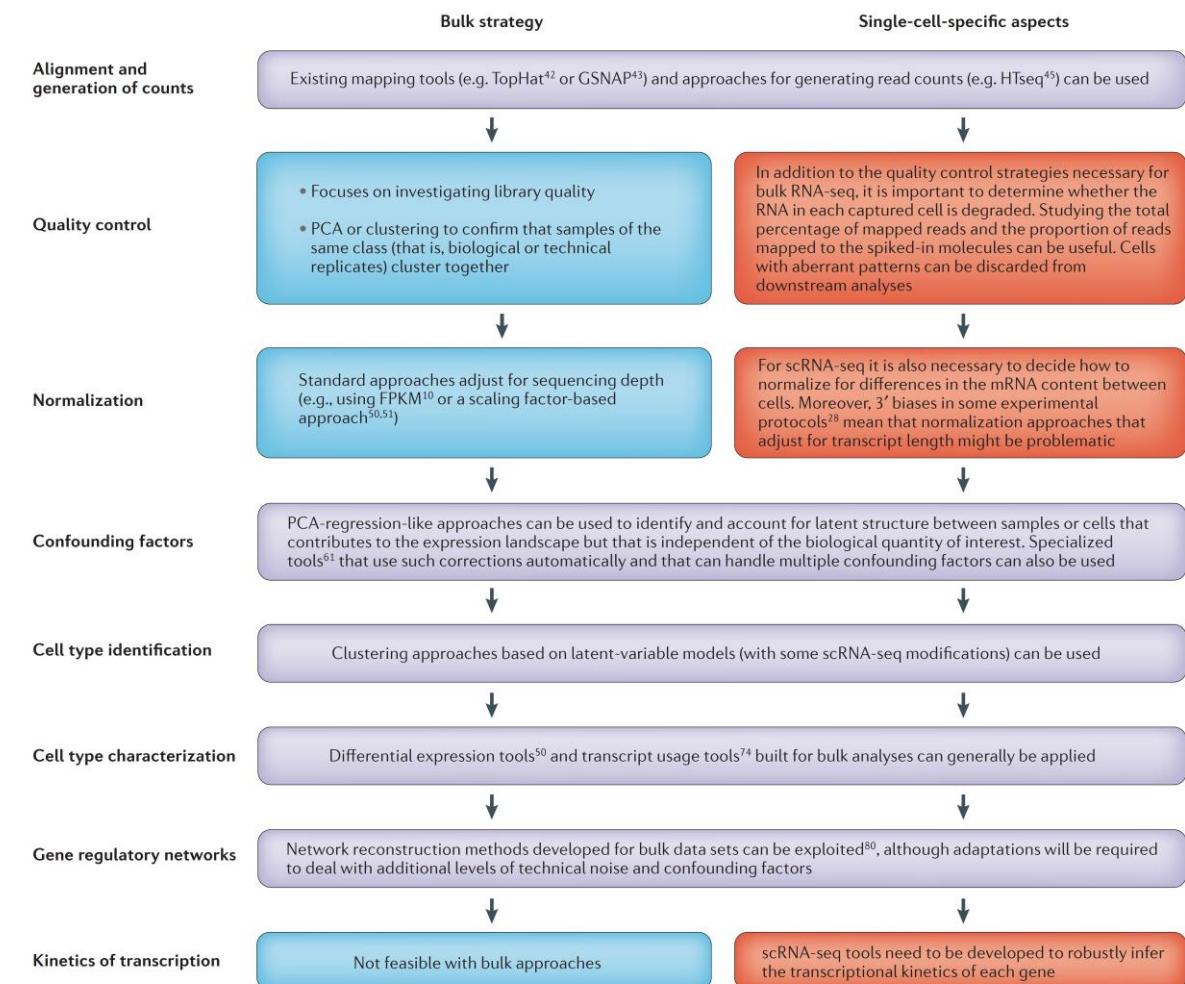


# scRNA-seq Data Analysis

Our goal is to derive/extract real biology from technically noisy data.

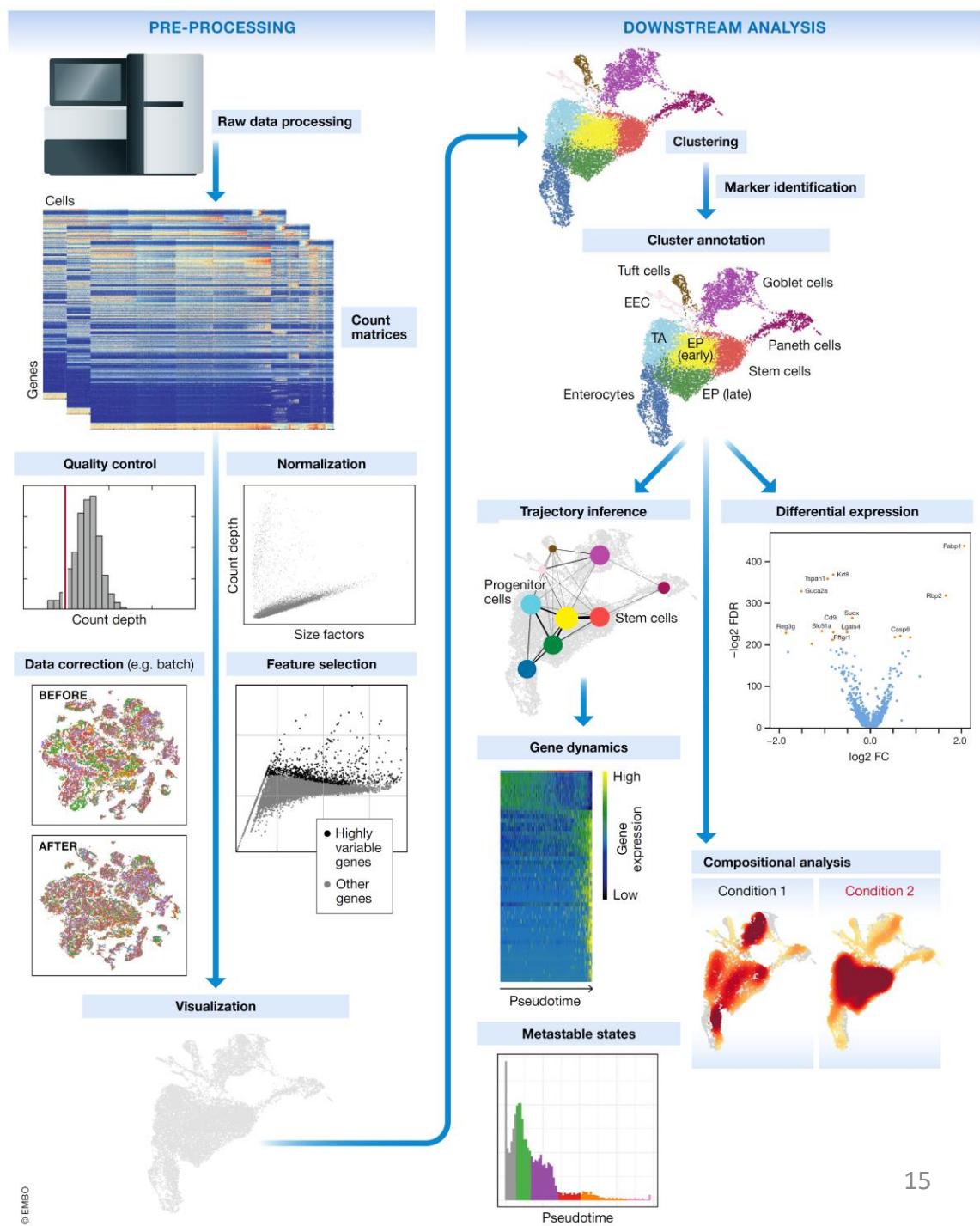
# What is different from bulk?

- The main sources of discrepancy between the libraries are:
  - **Amplification** (up to 1 million fold)
  - **Gene ‘dropouts’**
- In both cases the discrepancies are introduced due to low starting amounts of transcripts since the RNA comes from one cell only.



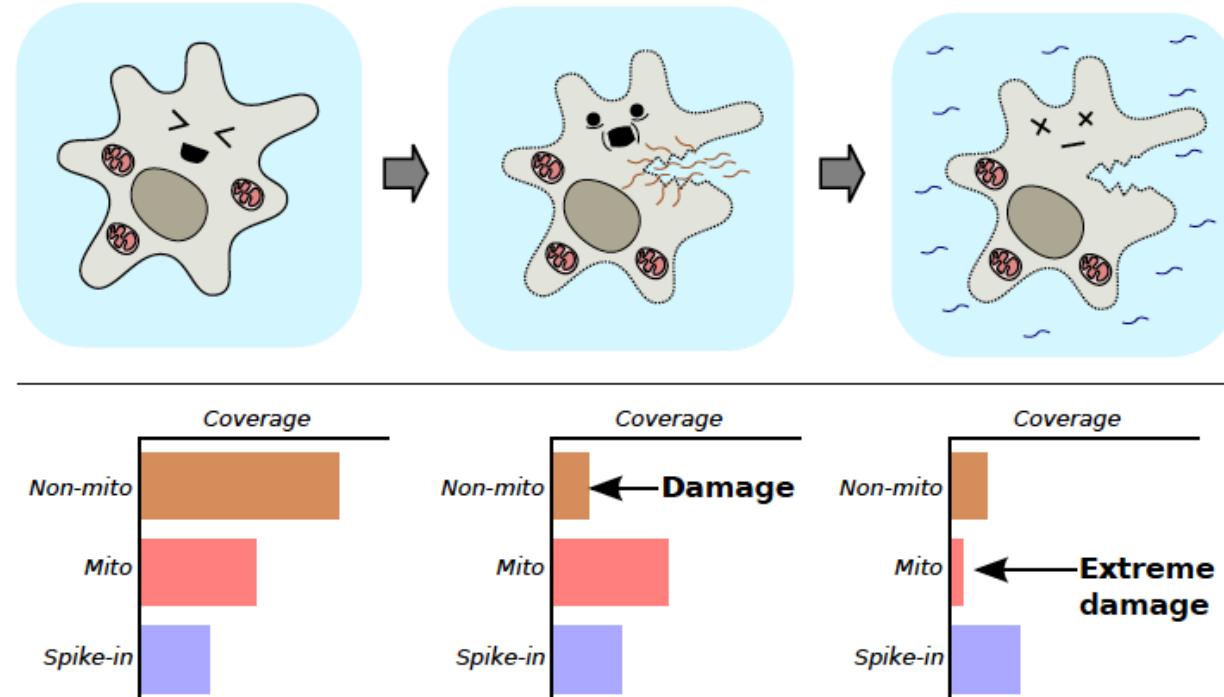
# scRNA-seq Data Analysis

- Preprocessing:
  - Reads to count matrix
  - Quality control (QC)
  - Normalization
  - Batch correction
  - Feature selection
- Downstream
  - Cell type identification (clustering/classification)
  - Trajectory inference
  - Differential expression
  - Compositional analysis
  - Co-expression network analysis



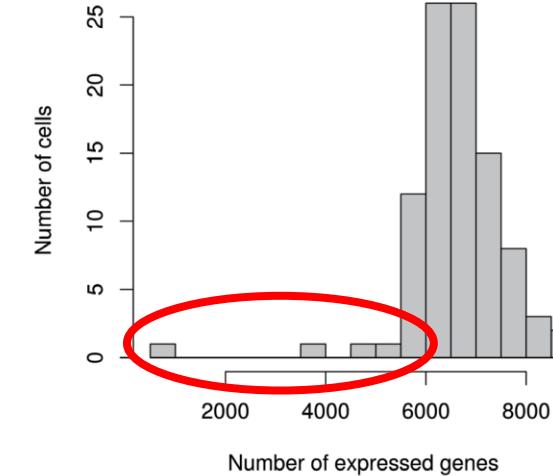
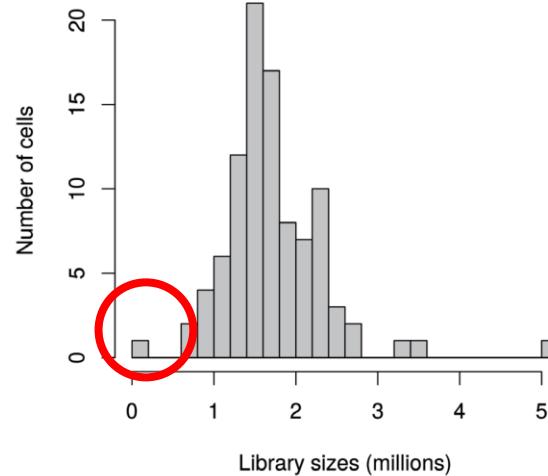
# Quality control of cells (1)

- Low sequencing depth
- Low numbers of expressed genes (i.e. any nonzero count)
- High spike-in (if present) or mitochondrial content



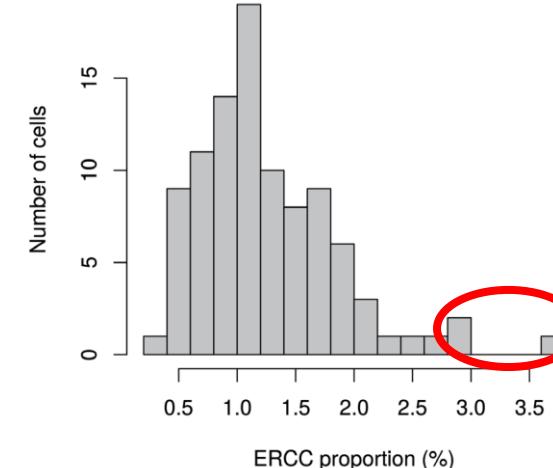
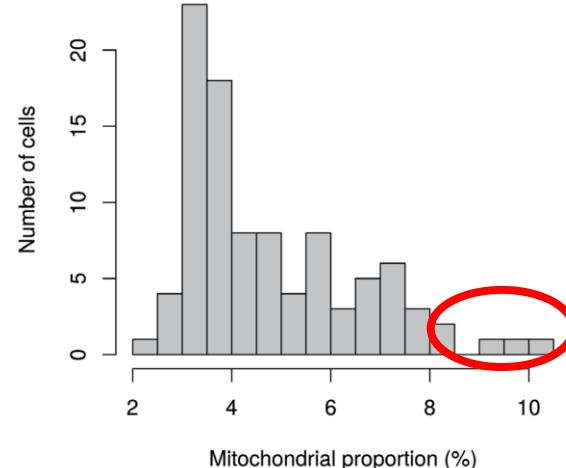
# Quality control of cells (2)

RNA has not been  
efficiently captured  
during library  
preparation

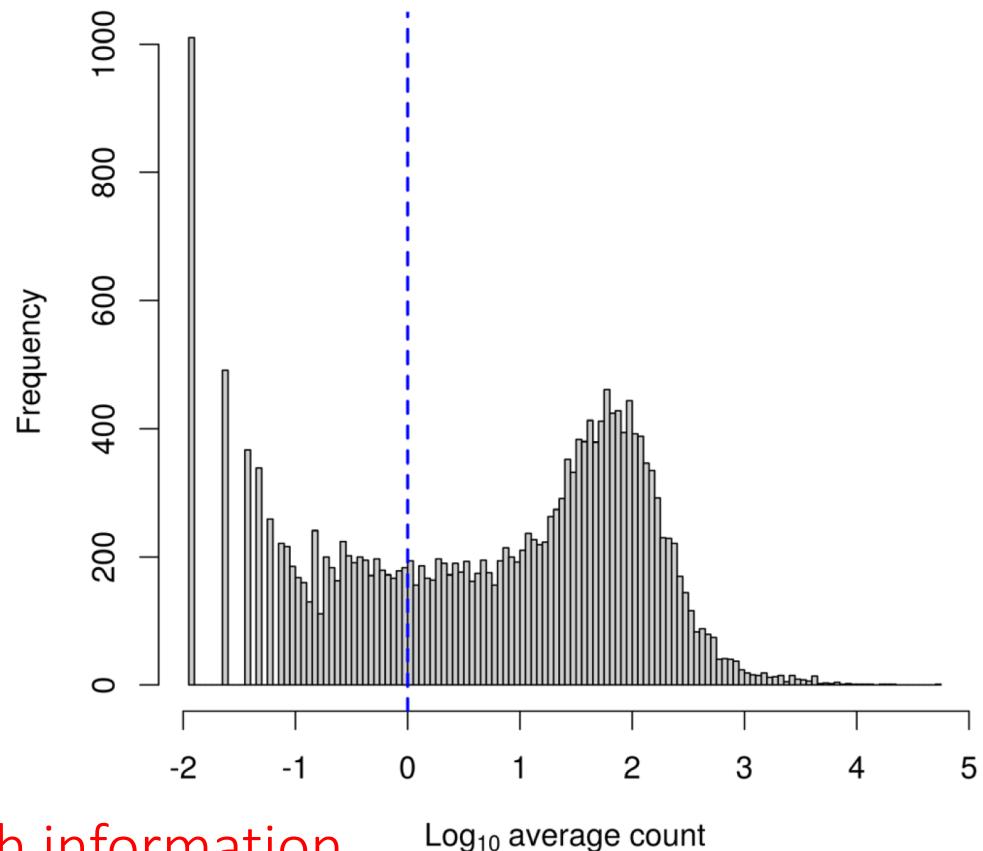


Diverse transcript  
population not  
captured

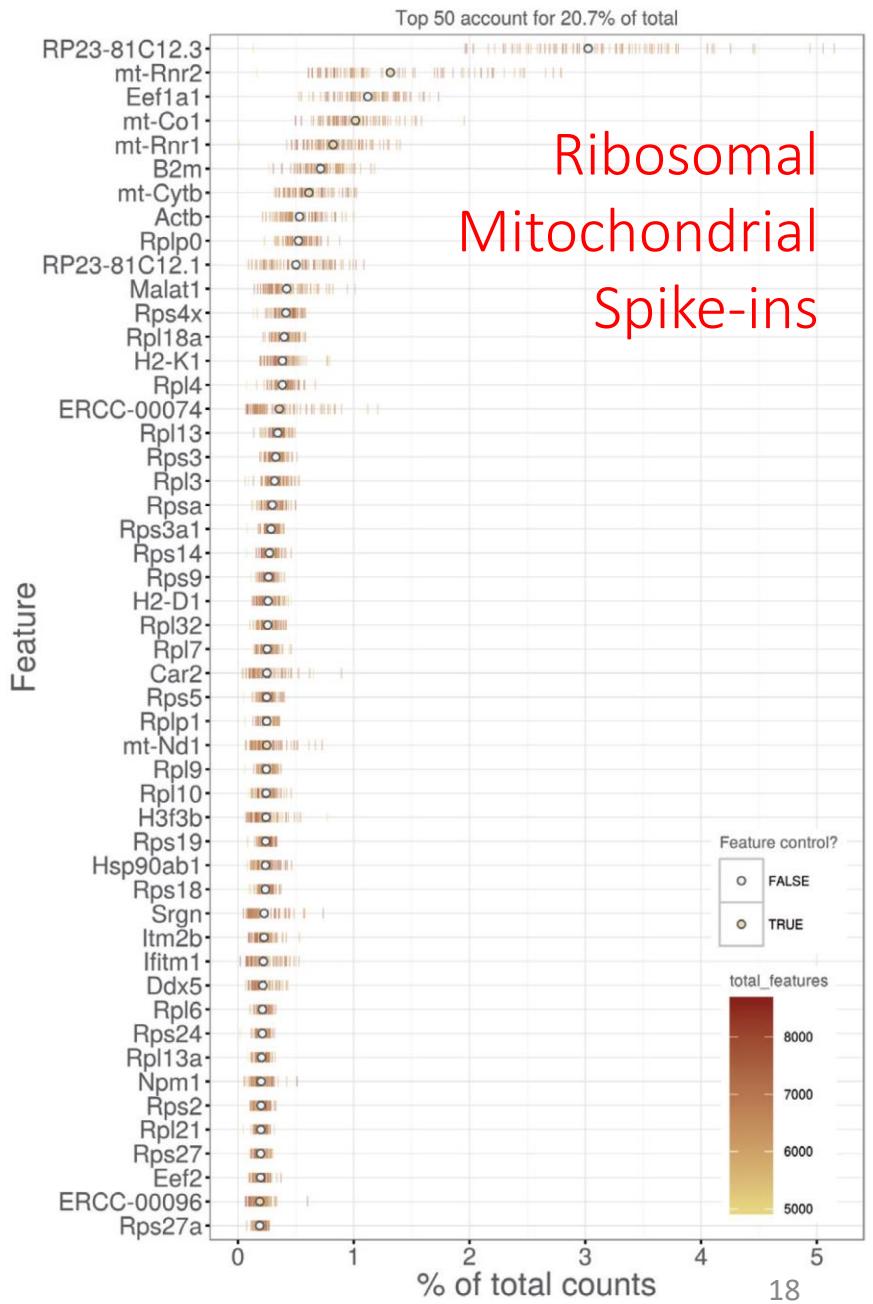
Possibly because of  
increased apoptosis  
and/or loss of cytoplasmic  
RNA from lysed cells



# Quality control of genes



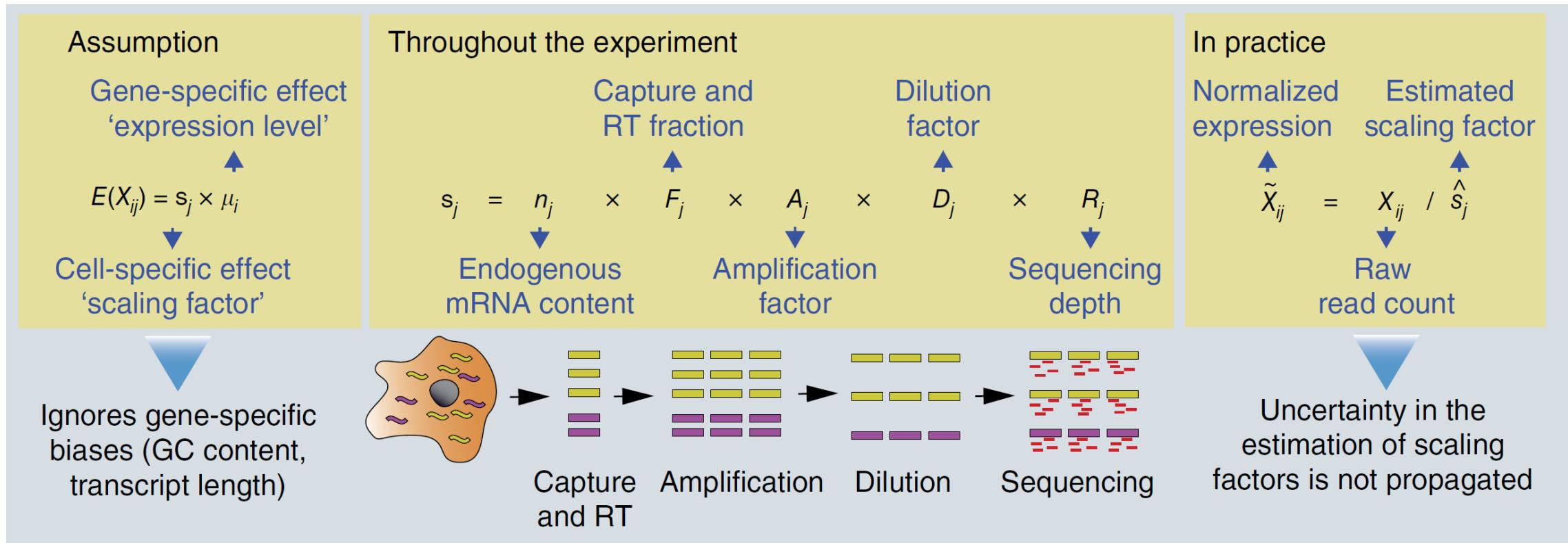
Not enough information  
for reliable statistical  
inference



# QC (pitfalls and recommendations)

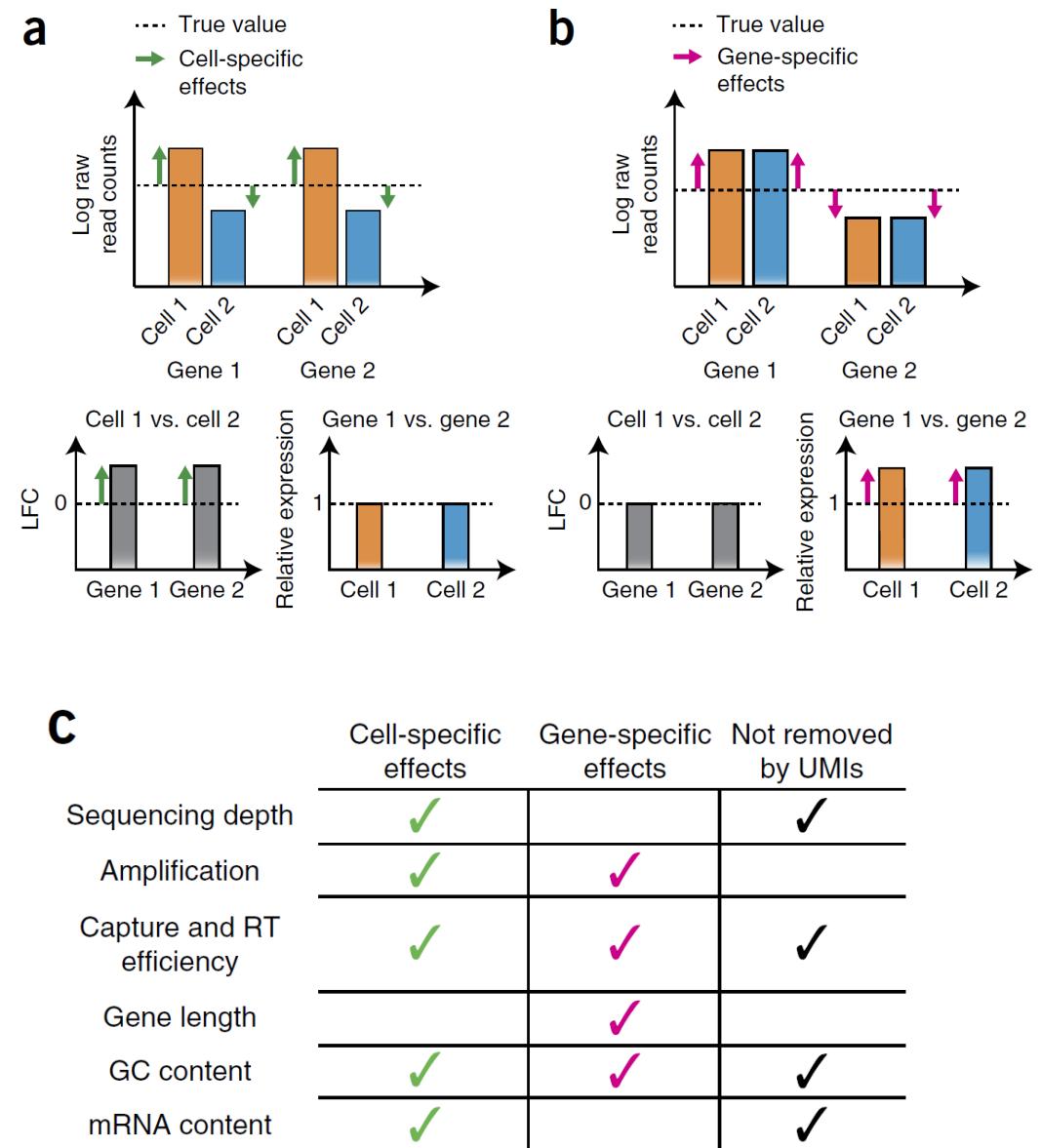
- Perform QC by finding outlier peaks in the number of genes, the count depth and the fraction of mitochondrial reads. Consider these covariates jointly instead of separately.
- Be as permissive of QC thresholding as possible, and revisit QC if downstream clustering cannot be interpreted.
- If the distribution of QC covariates differ between samples, QC thresholds should be determined separately for each sample to account for sample quality differences as in Plasschaert et al (2018).

# Normalization (1)



# Normalization (2)

- The aim is bring all cells onto the same distribution to remove biases
- We want to preserve biological variability, not introduce new technical variation
- Primary source of bias is sequencing depth – scale down counts accordingly
- Need a method that is robust to sparsity and composition bias
  - TMM & DESeq size factors are not!

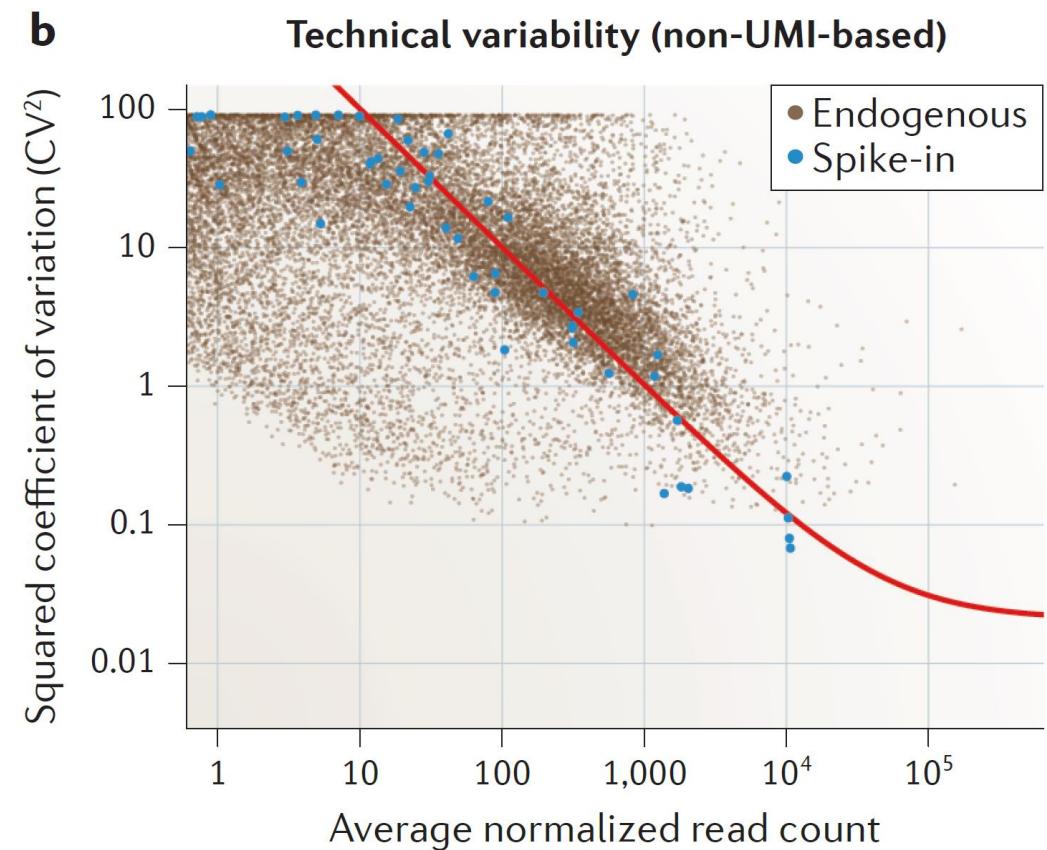


# Normalization (3)

## *Using spike-ins*

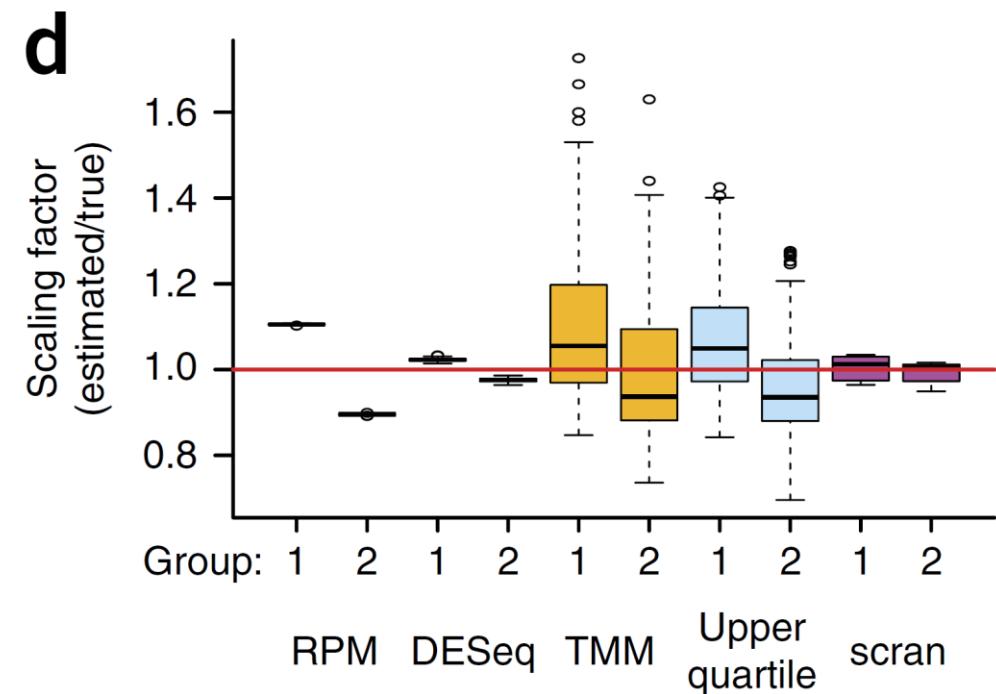
### Caveats:

- The same quantity of spike-in RNA may not be consistently added to each sample
- Synthetic spike-in transcripts may not behave in the same manner as endogenous transcripts
- Not easily incorporated in all scRNA-seq protocols (not in droplet-based)



# Normalization (5)

- Bulk RNA-based methods: FPKM, CPM, TPM, upperquartile
- Log normalization (Seurat)
- Negative binomial (Monocle)
- Downsampling (RaceID)
- ...



# Normalization (pitfalls and recommendations)

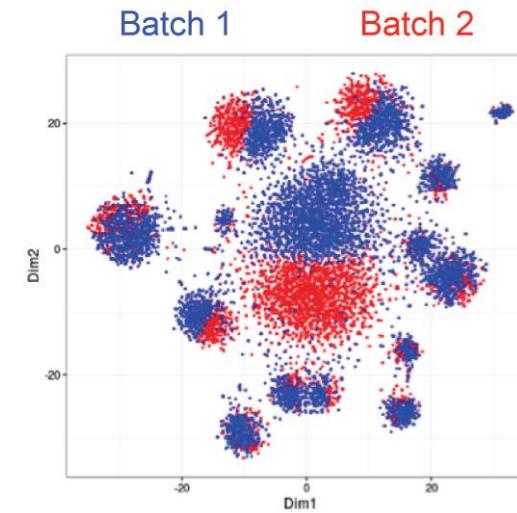
- We recommend scran for normalization of non-full-length datasets. An alternative is to evaluate normalization approaches via scone especially for plate-based datasets. Full-length scRNA-seq protocols can be corrected for gene length using bulk methods.
- There is no consensus on scaling genes to 0 mean and unit variance. We prefer not to scale gene expression.
- Normalized data should be  $\log(x+1)$ -transformed for use with downstream analysis methods that assume data are normally distributed.

# Confounders and batch effects (1)

## 1. Technical variability

- Changes in sample quality/processing
- Library prep or sequencing technology
- ‘Experimental reality’

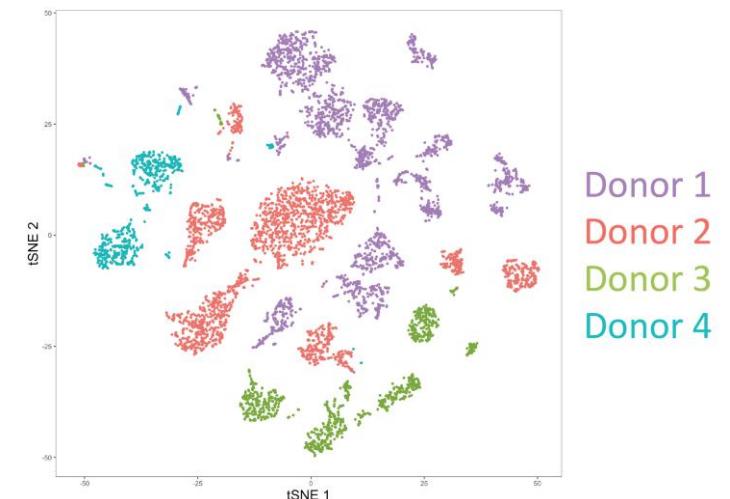
Technical ‘batch effects’ confound downstream analysis



## 1. Biological variability

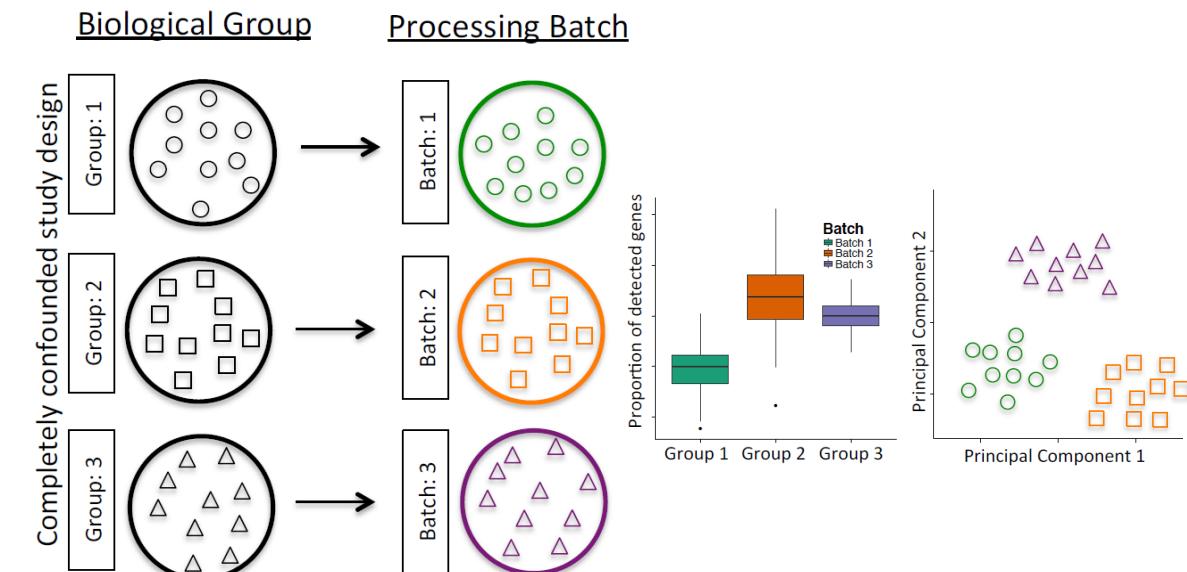
- Patient differences
- Environmental/genetic perturbation
- Evolution! (cross-species analysis)

Biological ‘batch effects’ confound comparisons of scRNA-seq data



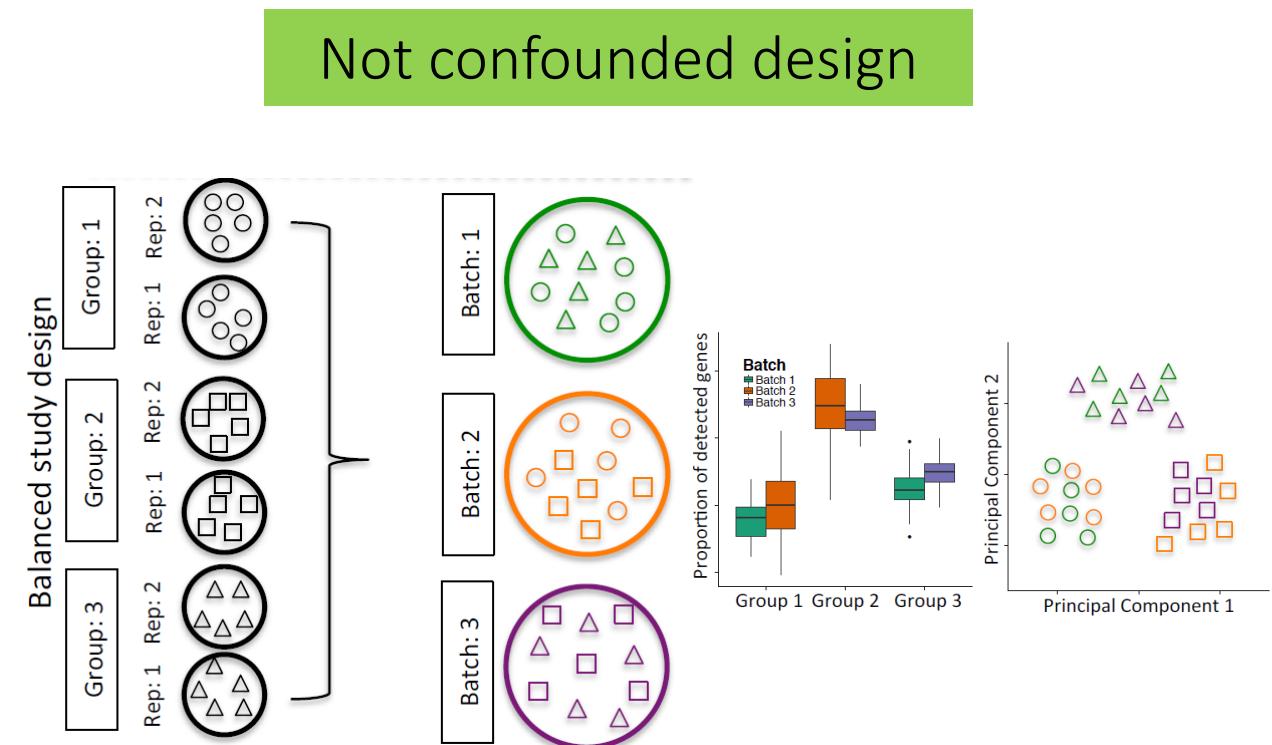
# Confounders and batch effects (2)

## Confounded design



Don't design your experiment like this!!!

## Not confounded design



Good experimental design *does not remove batch effects*, it prevents them from biasing your results.

# Batch correction methods

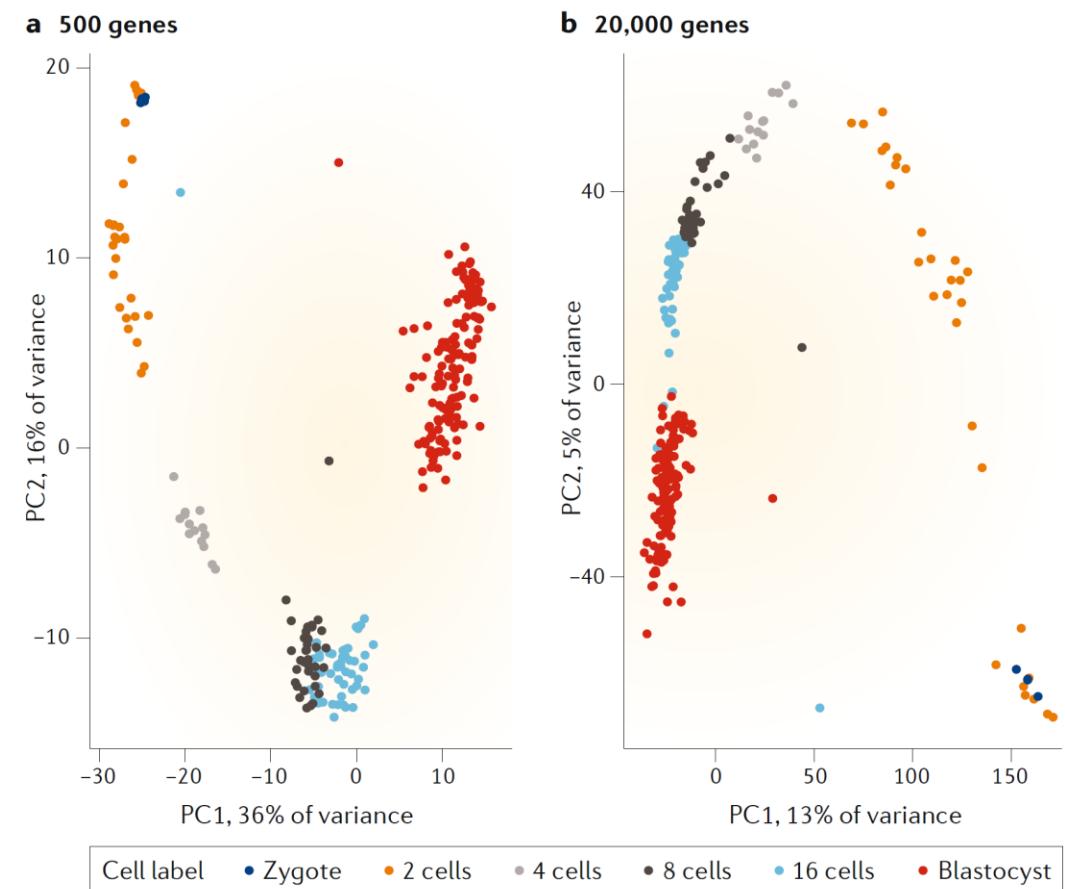
- MNNcorrect (<https://doi.org/10.1038/nbt.4091>)
- CCA + anchors (Seurat v3) (<https://doi.org/10.1101/460147>)
- CCA + dynamic time warping (Seurat v2) (<https://doi.org/10.1038/nbt.4096>)
- LIGER (<https://doi.org/10.1101/459891>)
- Harmony (<https://doi.org/10.1101/461954>)
- Conos (<https://doi.org/10.1101/460246>)
- Scanorama (<https://doi.org/10.1101/371179>)
- scMerge (<https://doi.org/10.1073/pnas.1820006116>)
- ...

**Two broad strategies:**

- Joint dimension reduction
- Graph-based joint clustering

# Feature selection

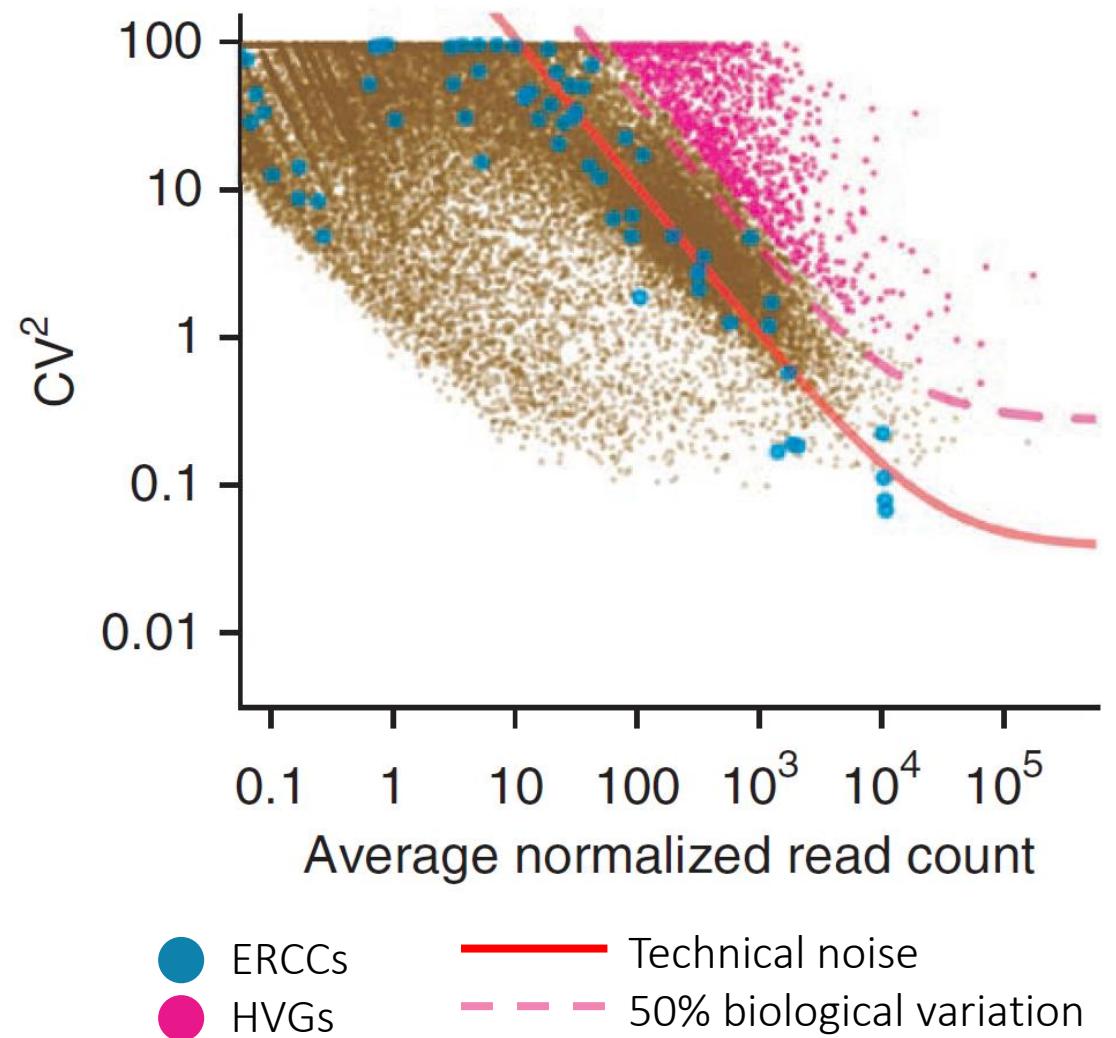
- Curse of dimensionality:  
More features (genes) -> smaller distances between samples (cells)
- Remove genes which only exhibit technical noise
  - Increase the signal:noise ratio
  - Reduce the computational complexity



# Feature selection

## Highly Variable Genes (HVG)

- $CV = \frac{var}{mean} = \frac{\sigma}{\mu}$
- Fit a gamma generalized linear model
- No ERCCs?
  - > estimate technical noise based on all genes



# Feature selection

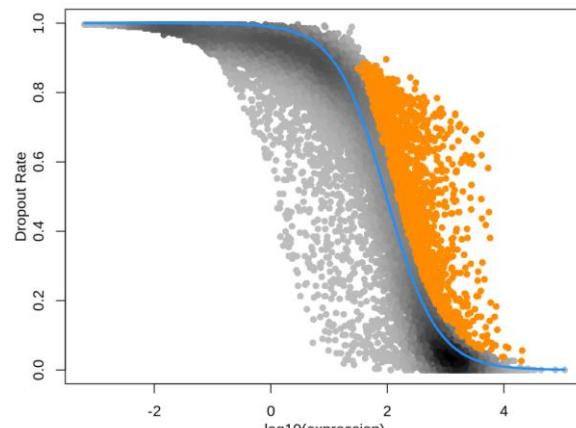
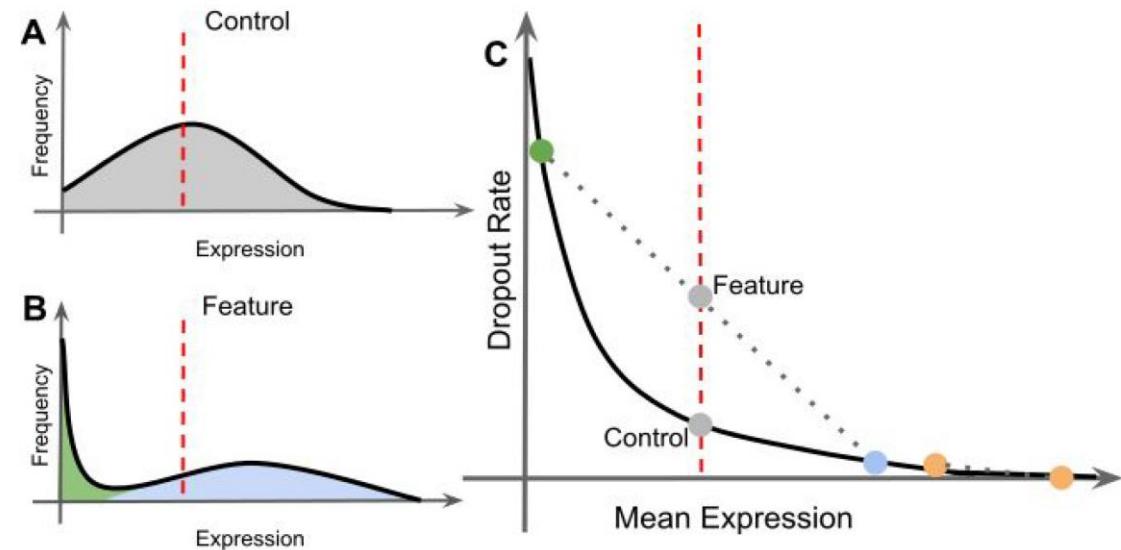
## M3Drop: Dropout-based feature selection

- Reverse transcription is an enzyme reaction thus can be modelled using the Michaelis-Menten equation:

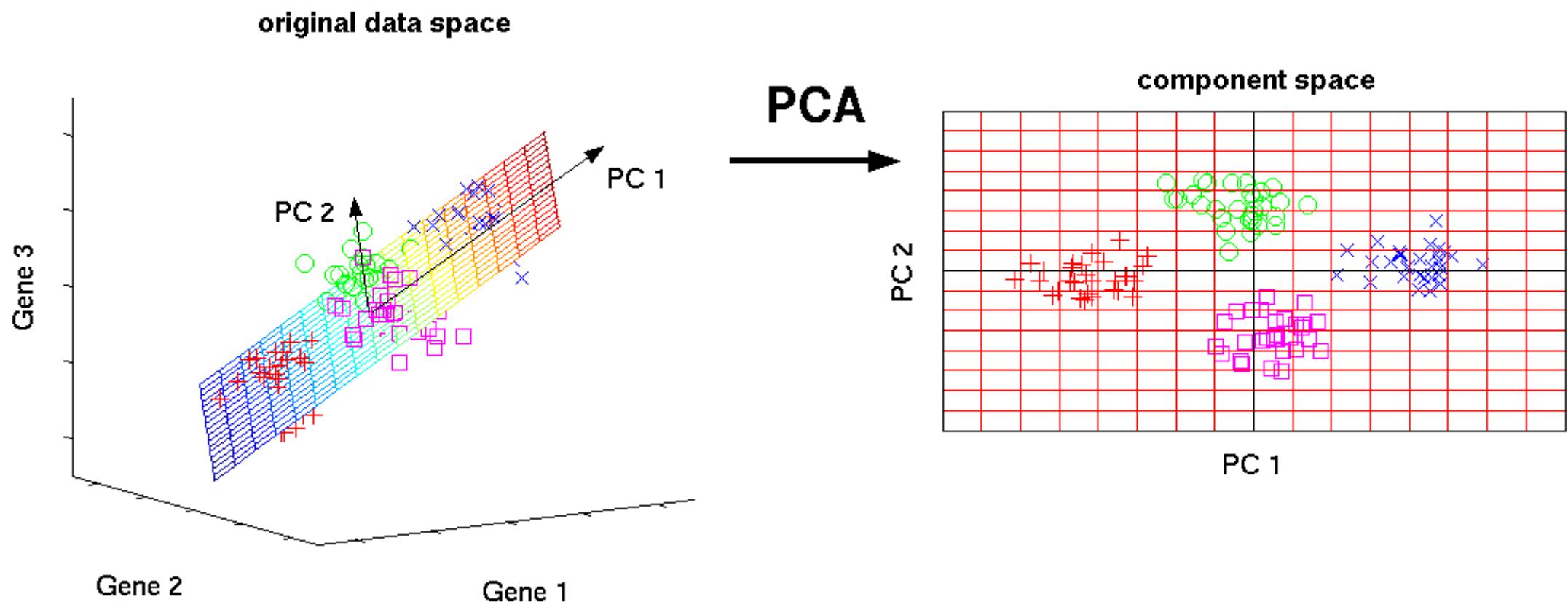
$$P_{dropout} = 1 - \frac{S}{K_M + S}$$

$S$ : average expression

$K_M$ : Michaelis-Menten constant



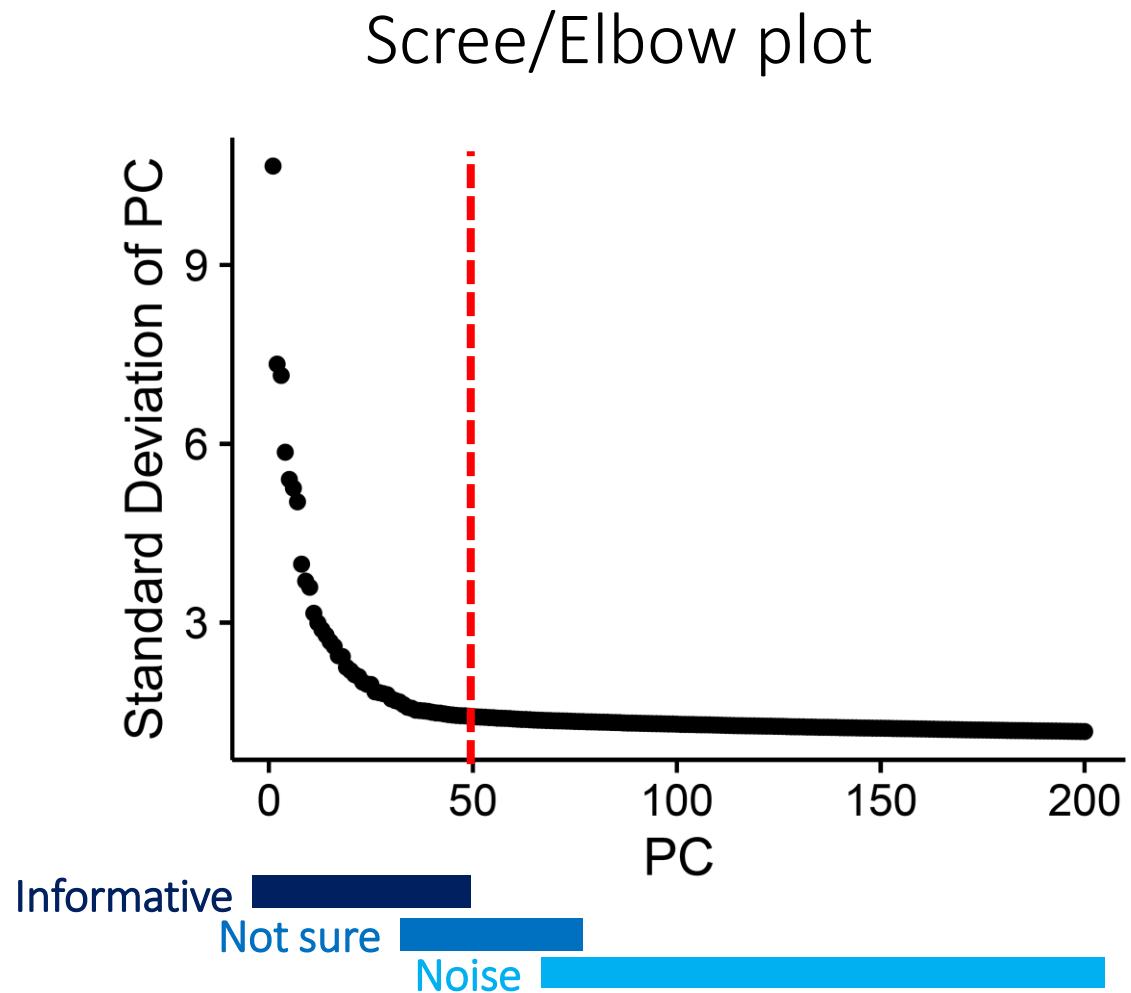
# PCA: Principal Component Analysis



# Feature selection

## Selecting principal components

- To overcome the extensive technical noise in scRNA-seq data, it is common to cluster cells based on their PCA scores
- Each PC represents a ‘metagene’ that (linearly) combines information across a correlated gene set



# Feature Selection (pitfalls and recommendations)

- We recommend selecting between 1,000 and 5,000 highly variable genes depending on dataset complexity.
- Feature selection methods that use gene expression means and variances cannot be used when gene expression values have been normalized to zero mean and unit variance, or when residuals from model fitting are used as normalized expression values. Thus, one must consider what pre-processing to perform before selecting HVGs.

# Dimensionality reduction (1)

Matrix  
factorization

Graph-based

Auto-encoders

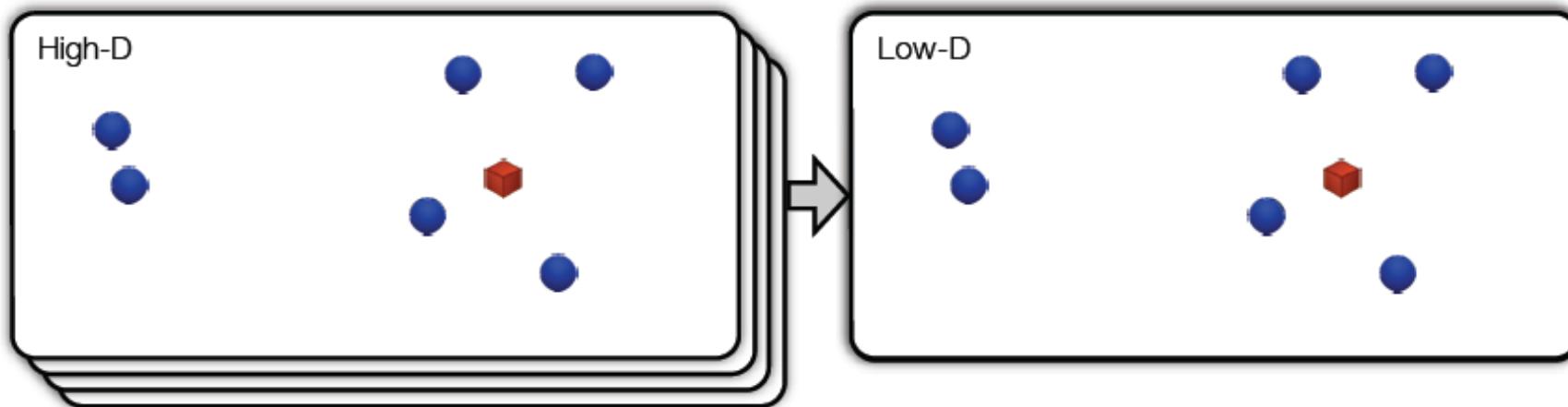
PCA	linear		
ICA	linear		
MDS	non-linear		
Sparce NNMF	non-linear	2010	<a href="https://pdfs.semanticscholar.org/664d/40258f12ad28ed0b7d4_c272935ad72a150db.pdf">https://pdfs.semanticscholar.org/664d/40258f12ad28ed0b7d4_c272935ad72a150db.pdf</a>
cPCA	non-linear	2018	<a href="https://doi.org/10.1038/s41467-018-04608-8">https://doi.org/10.1038/s41467-018-04608-8</a>
ZIFA	non-linear	2015	<a href="https://doi.org/10.1186/s13059-015-0805-z">https://doi.org/10.1186/s13059-015-0805-z</a>
ZINB-WaVE	non-linear	2018	<a href="https://doi.org/10.1038/s41467-017-02554-5">https://doi.org/10.1038/s41467-017-02554-5</a>

Diffusion maps	non-linear	2005	<a href="https://doi.org/10.1073/pnas.0500334102">https://doi.org/10.1073/pnas.0500334102</a>
Isomap	non-linear	2000	<a href="https://doi.org/10.1126/science.290.5500.2319">https://doi.org/10.1126/science.290.5500.2319</a>
t-SNE	non-linear	2008	<a href="https://lvdmaaten.github.io/publications/papers/JMLR_2008.pdf">https://lvdmaaten.github.io/publications/papers/JMLR_2008.pdf</a>
- BH t-SNE	non-linear	2014	<a href="https://lvdmaaten.github.io/publications/papers/JMLR_2014.pdf">https://lvdmaaten.github.io/publications/papers/JMLR_2014.pdf</a>
- Flt-SNE	non-linear	2017	<a href="https://arxiv.org/abs/1712.09005">arXiv:1712.09005</a>
LargeVis	non-linear	2018	<a href="https://arxiv.org/abs/1602.00370">arXiv:1602.00370</a>
UMAP	non-linear	2018	<a href="https://arxiv.org/abs/1802.03426">arXiv:1802.03426</a>
PHATE	non-linear	2017	<a href="https://www.biorxiv.org/content/biorxiv/early/2018/06/28/120378.full.pdf">https://www.biorxiv.org/content/biorxiv/early/2018/06/28/120378.full.pdf</a>

scvis	non-linear	2018	<a href="https://doi.org/10.1038/s41467-018-04368-5">https://doi.org/10.1038/s41467-018-04368-5</a>
VASC	non-linear	2018	<a href="https://doi.org/10.1016/j.gpb.2018.08.003">https://doi.org/10.1016/j.gpb.2018.08.003</a>

# Dimensionality Reduction (2)

- t-SNE: t-distributed stochastic neighborhood embedding

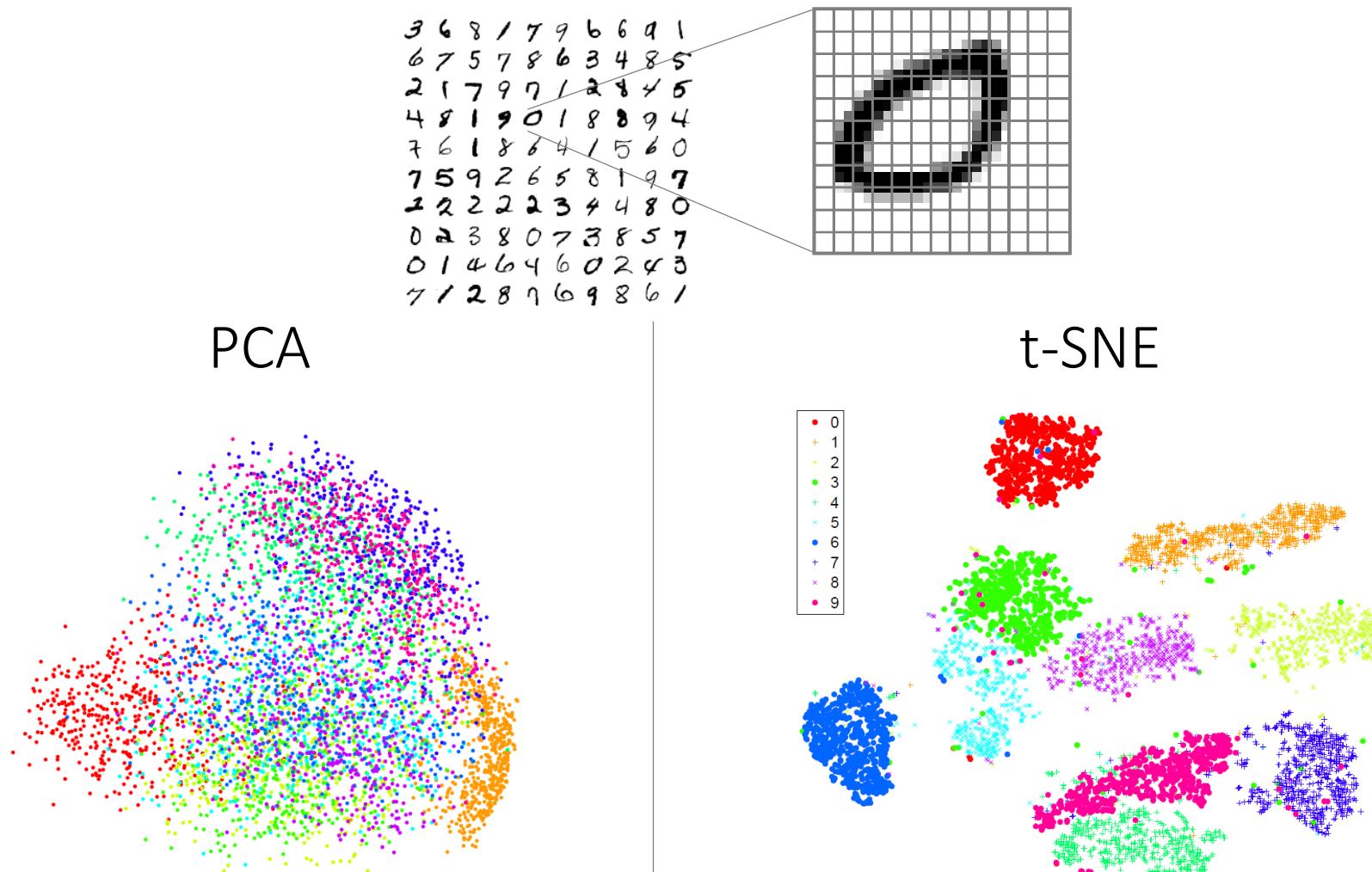


$$p_{ij} = \frac{\exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma^2)}{\sum_{k \neq l} \exp(-\|\mathbf{x}_k - \mathbf{x}_l\|^2 / 2\sigma^2)}$$

$$q_{ij} = \frac{(1 + \|\mathbf{y}_i - \mathbf{y}_j\|^2)^{-1}}{\sum_{k \neq l} (1 + \|\mathbf{y}_k - \mathbf{y}_l\|^2)^{-1}}$$

$$KL(P||Q) = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

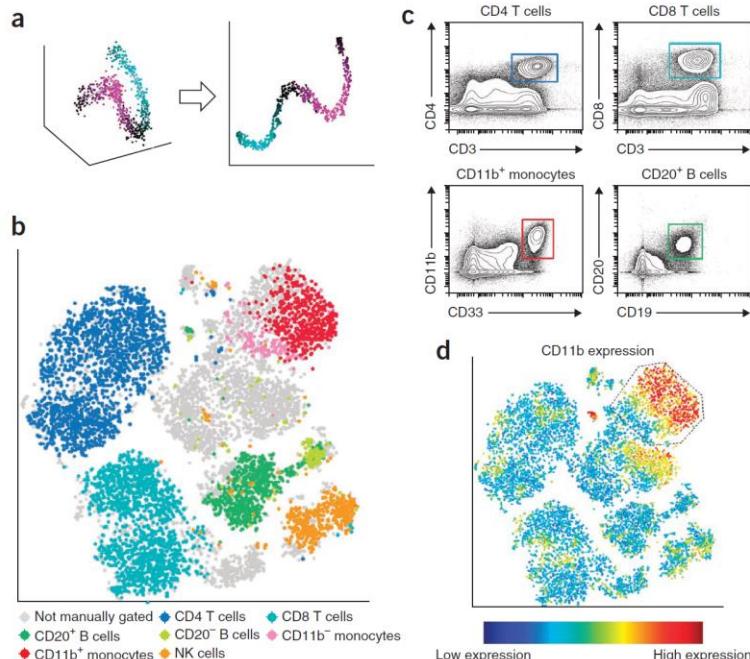
# Dimensionality Reduction (3)



# Dimensionality Reduction (4)

viSNE enables visualization of high dimensional single-cell data and reveals phenotypic heterogeneity of leukemia

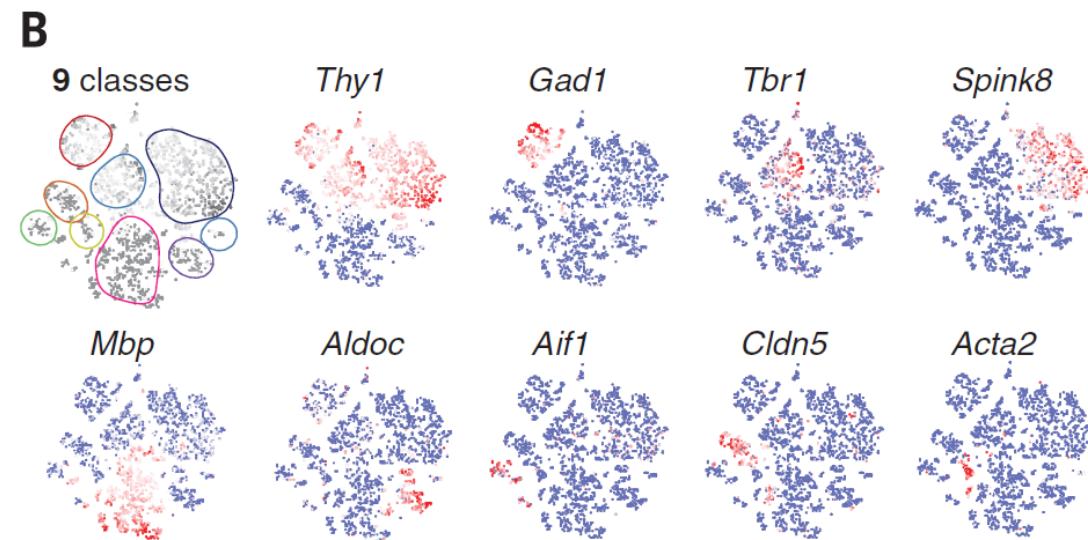
El-ad David Amir<sup>1</sup>, Kara L Davis<sup>2,3</sup>, Michelle D Tadmor<sup>1,3</sup>, Erin F Simonds<sup>2,3</sup>, Jacob H Levine<sup>1,3</sup>, Sean C Bendall<sup>2,3</sup>, Daniel K Shenfeld<sup>1,3</sup>, Smita Krishnaswamy<sup>1</sup>, Garry P Nolan<sup>2,4</sup> & Dana Pe'er<sup>1,4</sup>



Amir et al., Nature Biotechnology 2013

## Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq

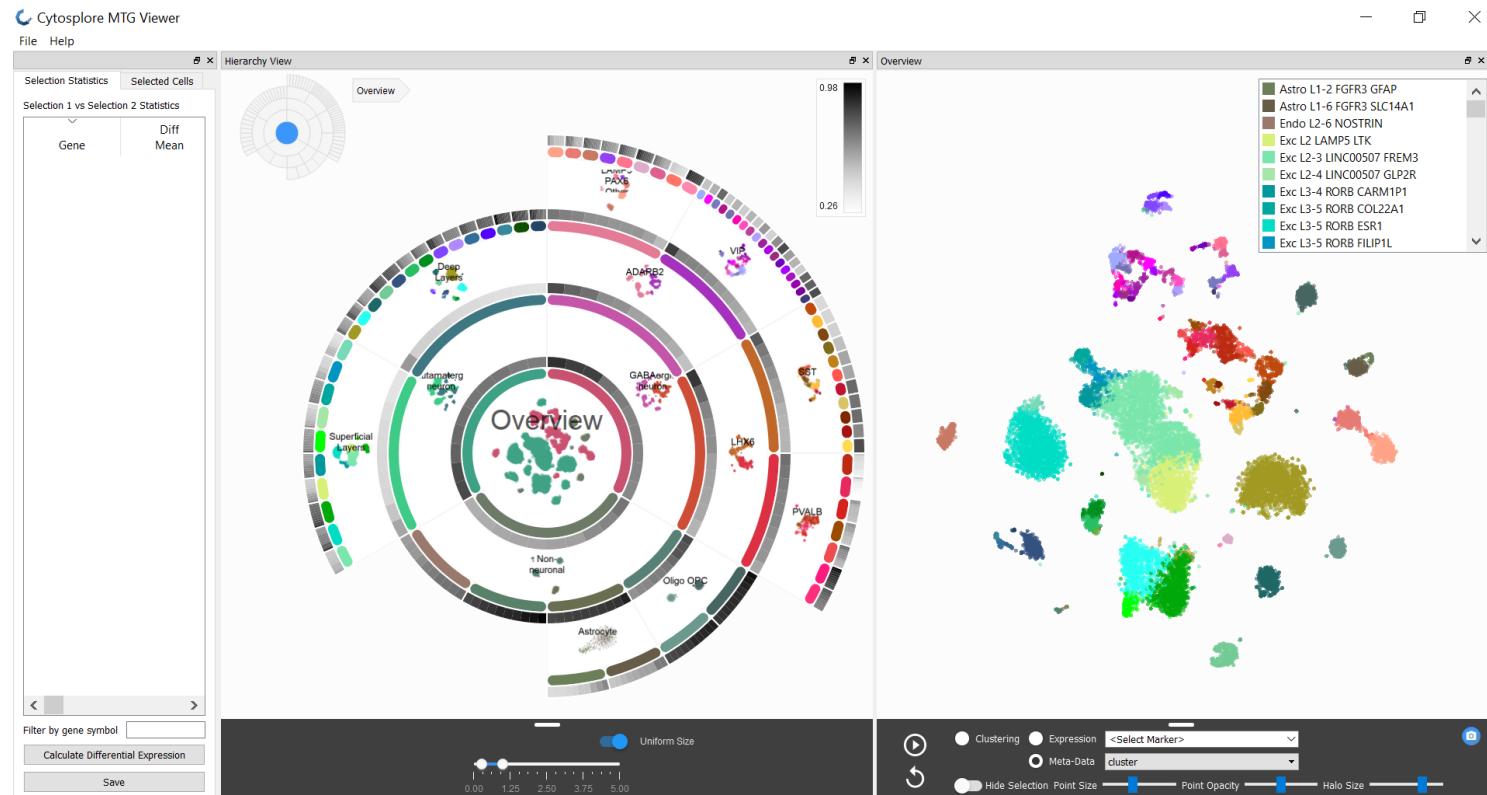
Amit Zeisel,<sup>1\*</sup> Ana B. Muñoz-Manchado,<sup>1\*</sup> Simone Codeluppi,<sup>1</sup> Peter Lönnberg,<sup>1</sup> Gioele La Manno,<sup>1</sup> Anna Juréus,<sup>1</sup> Sueli Marques,<sup>1</sup> Hermany Munguba,<sup>1</sup> Liqun He,<sup>2</sup> Christer Betsholtz,<sup>2,3</sup> Charlotte Rojny,<sup>4</sup> Gonçalo Castelo-Branco,<sup>1</sup> Jens Hjerling-Leffler,<sup>1†</sup> Sten Linnarsson<sup>1‡</sup>



Zeisel et al., Nature Biotechnology 2013<sup>45</sup>

# Dimensionality Reduction (5)

- CytoSplore: high performance single cell transcriptome visualizations  
<https://viewer.cytosplore.org>



# Real-time tSNE

<https://ai.googleblog.com/2018/06/realtime-tsne-visualizations-with.html>

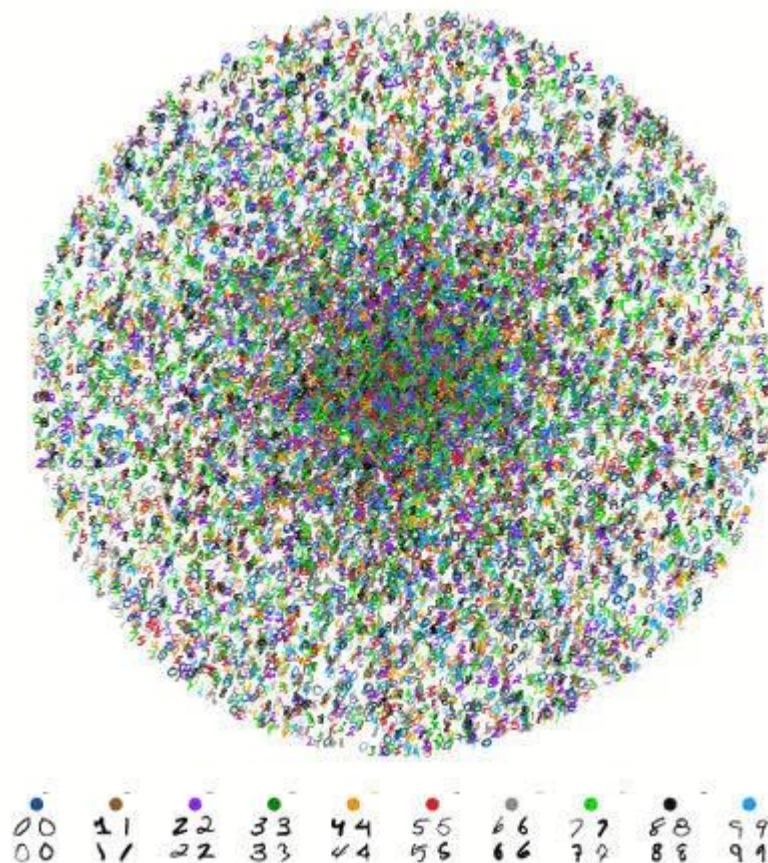
The screenshot shows a blog post on the Google AI Blog. The header includes the Google AI logo and the text "The latest news from Google AI". The main title is "Realtime tSNE Visualizations with TensorFlow.js" and the date is "Thursday, June 7, 2018". The author is listed as "Posted by Nicola Pezzotti, Software Engineering Intern, Google Zürich". The content discusses the t-distributed Stochastic Neighbor Embedding (tSNE) algorithm, its use in interpreting deep neural network outputs, and its limitations for large datasets. It highlights the use of TensorFlow.js for real-time visualizations.

Realtime tSNE Visualizations with TensorFlow.js

Thursday, June 7, 2018

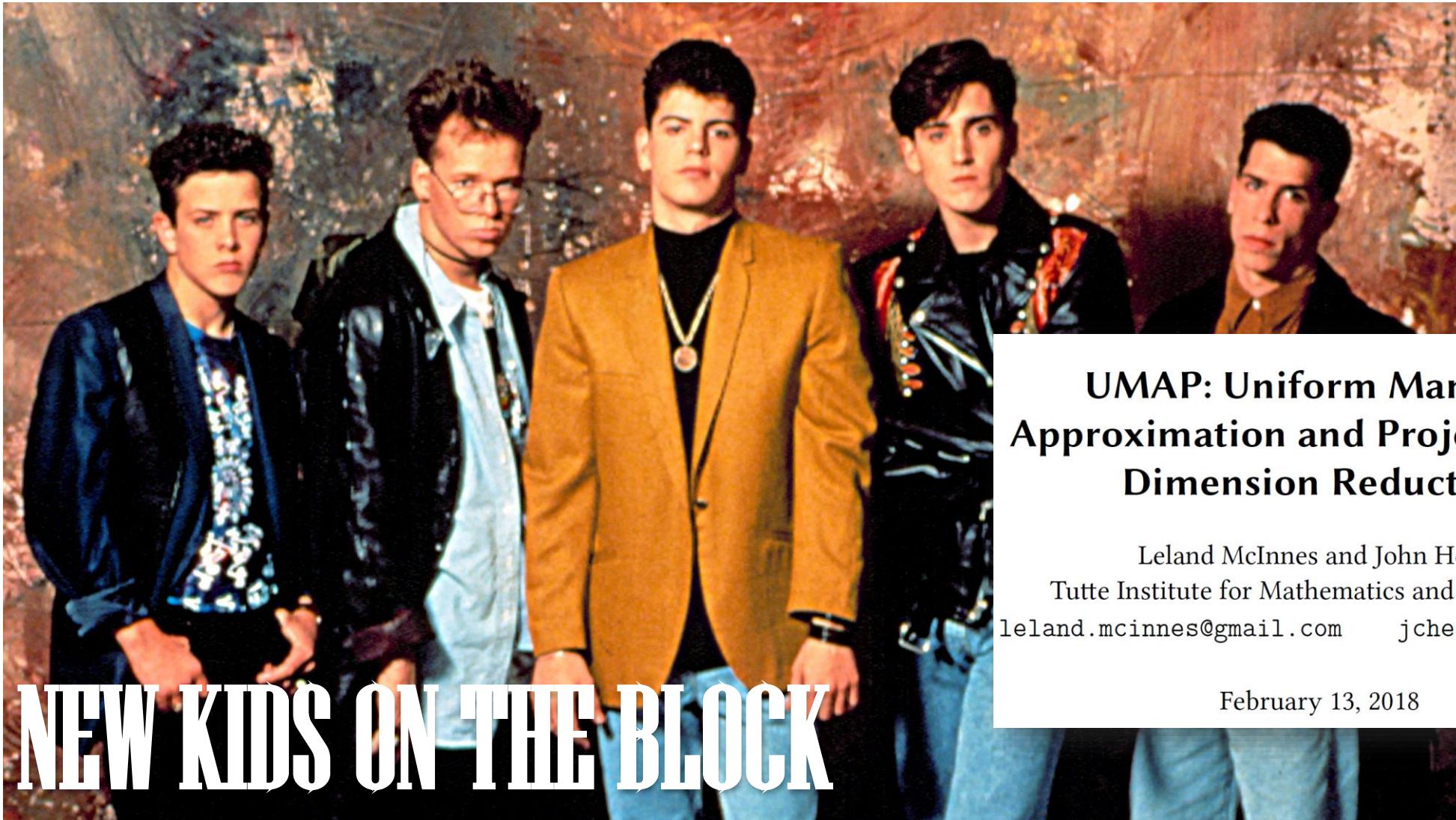
Posted by Nicola Pezzotti, Software Engineering Intern, Google Zürich

In recent years, the [t-distributed Stochastic Neighbor Embedding](#) (tSNE) algorithm has become one of the most used and insightful techniques for exploratory data analysis of high-dimensional data. Used to interpret deep neural network outputs in tools such as the [TensorFlow Embedding Projector](#) and [TensorBoard](#), a powerful feature of tSNE is that it reveals clusters of high-dimensional data points at different scales while requiring only minimal tuning of its parameters. Despite these advantages, the computational complexity of the tSNE algorithm limits its application to relatively small datasets. While several evolutions of tSNE have been developed to address this issue (mainly focusing on the scalability of the similarity computations between data points), they have so far not been enough to provide a truly interactive experience when visualizing the evolution of the tSNE embedding for large datasets.



<https://nicola17.github.io/tfjs-tsne-demo/>

# Dimensionality Reduction (5)



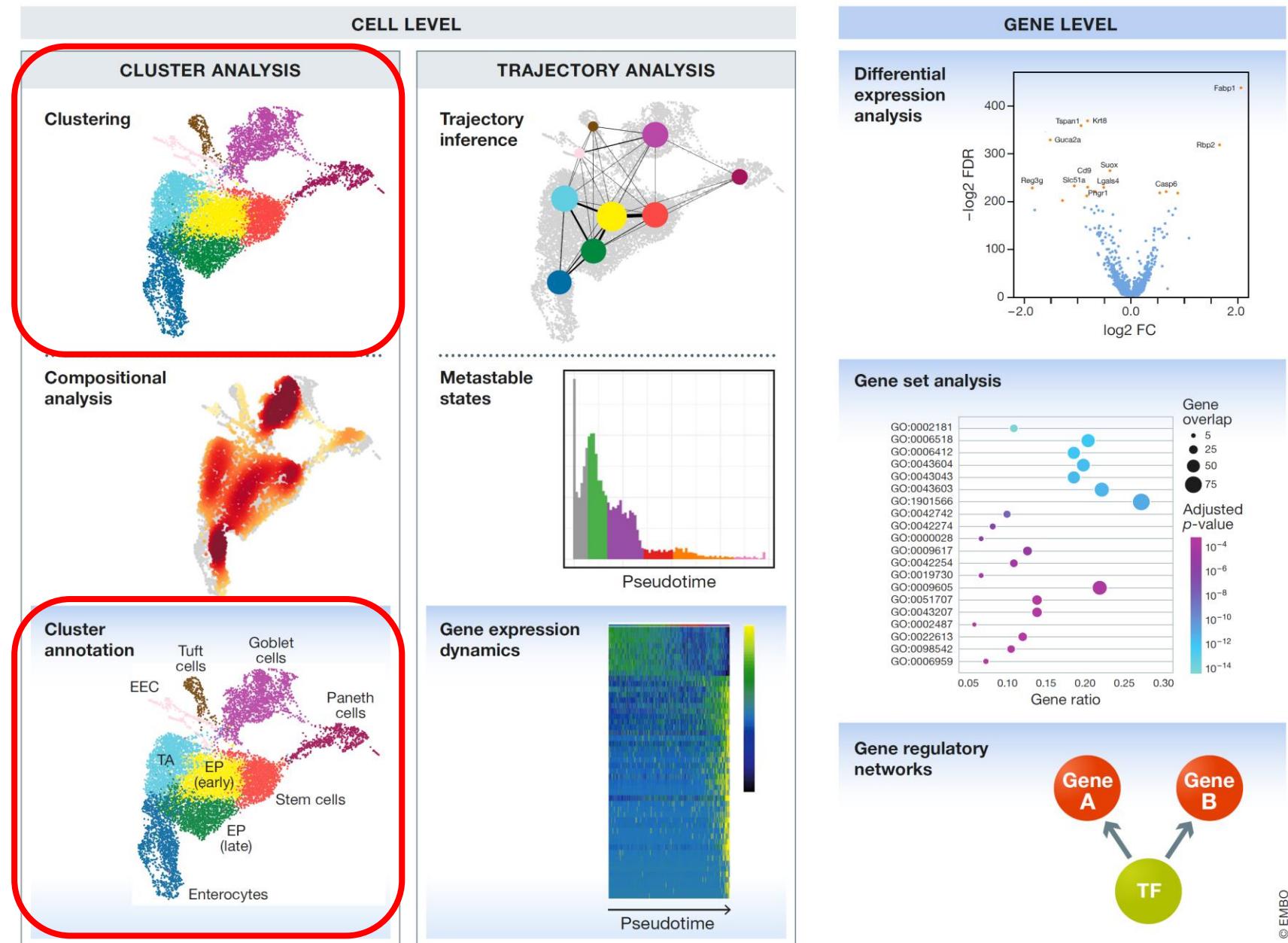
**NEW KIDS ON THE BLOCK**

**UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction**

Leland McInnes and John Healy  
Tutte Institute for Mathematics and Computing  
[leland.mcinnes@gmail.com](mailto:leland.mcinnes@gmail.com)    [jchealy@gmail.com](mailto:jchealy@gmail.com)

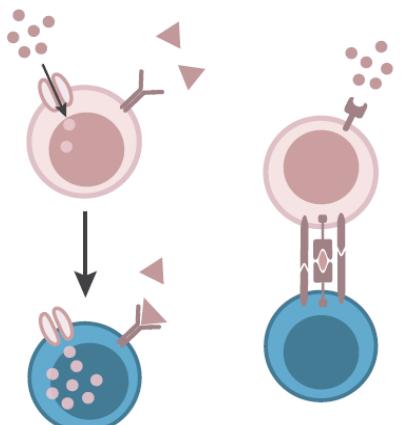
February 13, 2018

# scRNA-seq Downstream Analysis

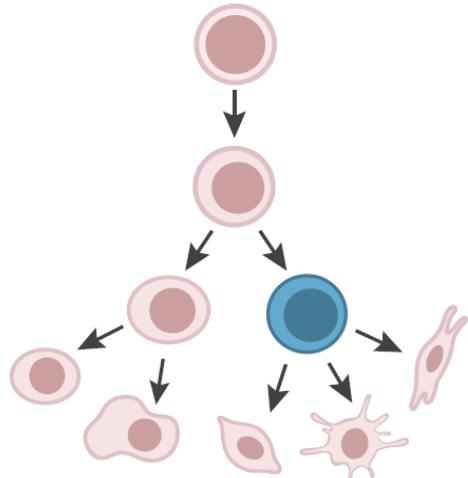


# Cell Identity

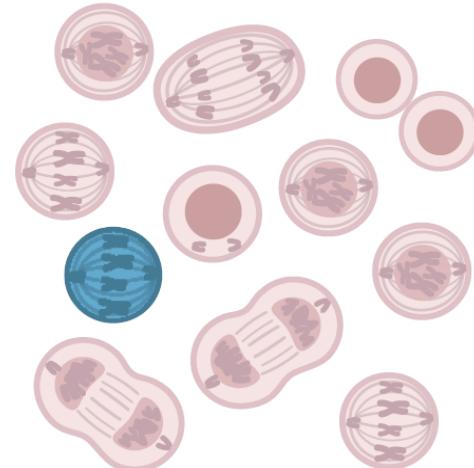
Environmental stimuli



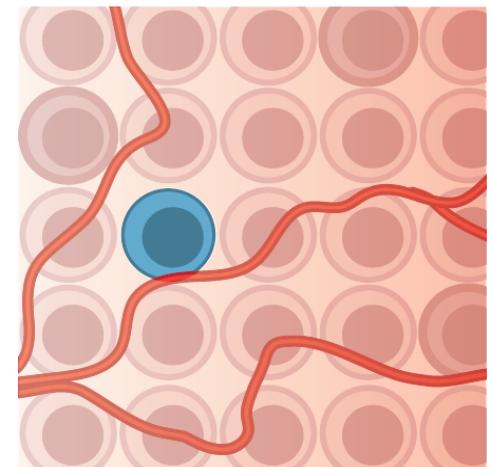
Cell development



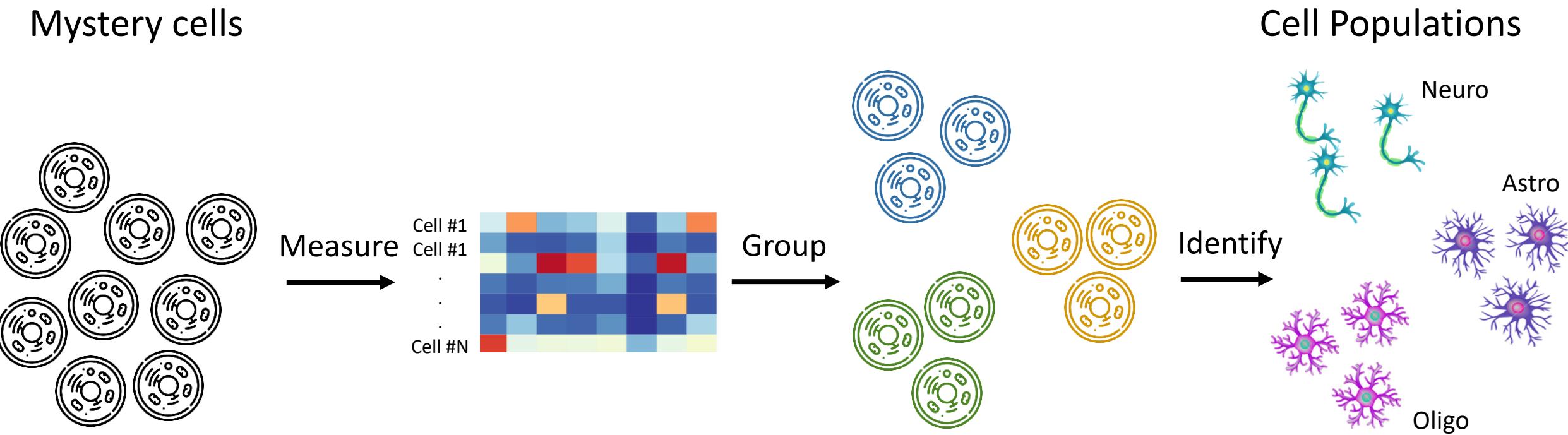
Cell cycle



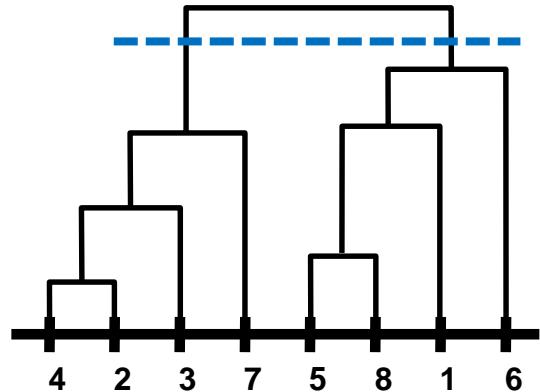
Spatial context



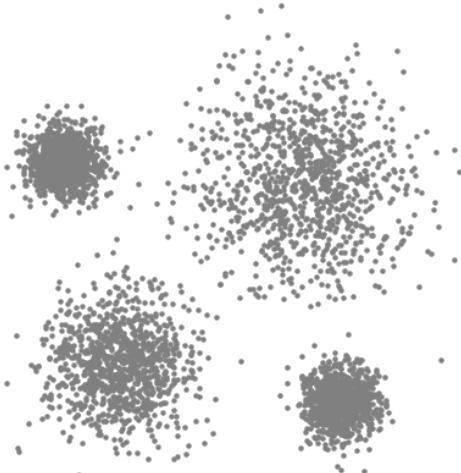
# How can we identify cell populations?



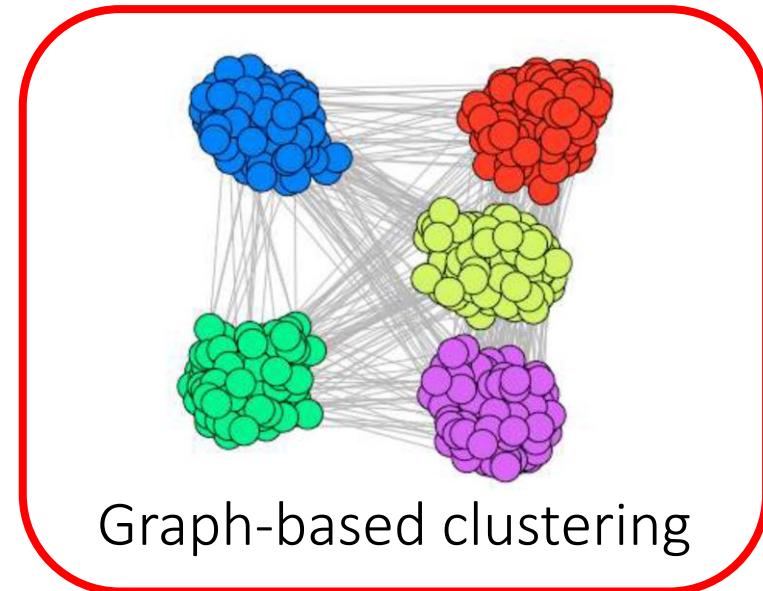
# Many clustering approaches



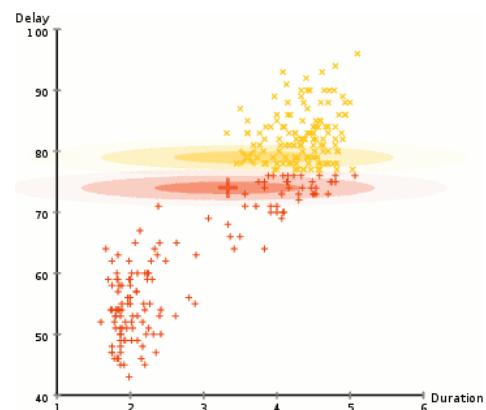
Hierarchical Clustering



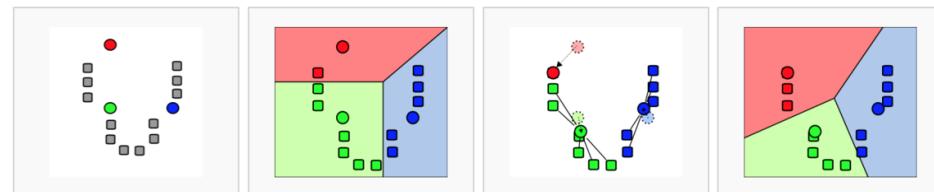
Mean shift clustering



Graph-based clustering



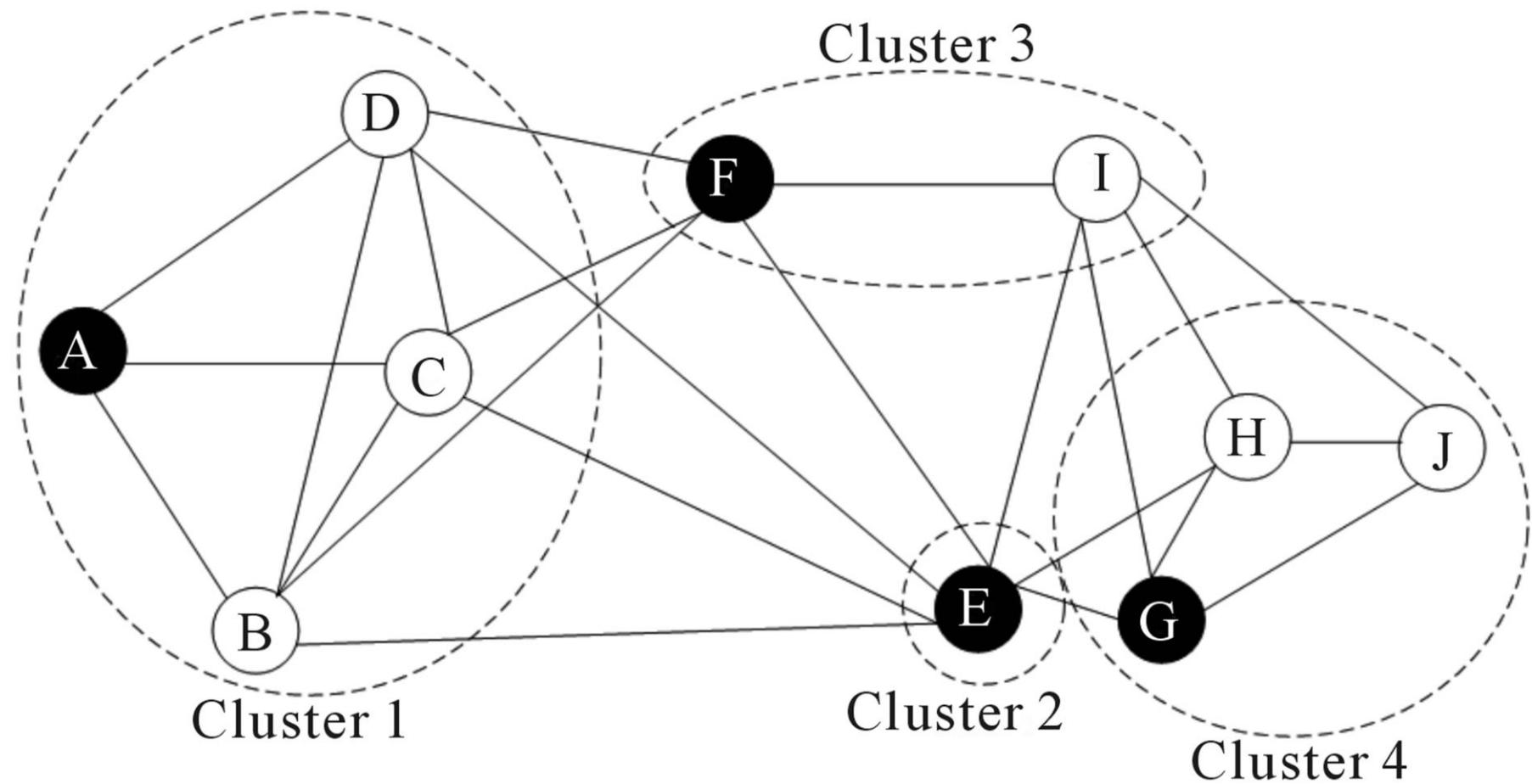
Gaussian mixture modeling



k-means clustering

# Graph-based clustering

Nodes -> cells  
Edges -> similarity



# Graph Types

- **k-Nearest Neighbor (kNN) graph**

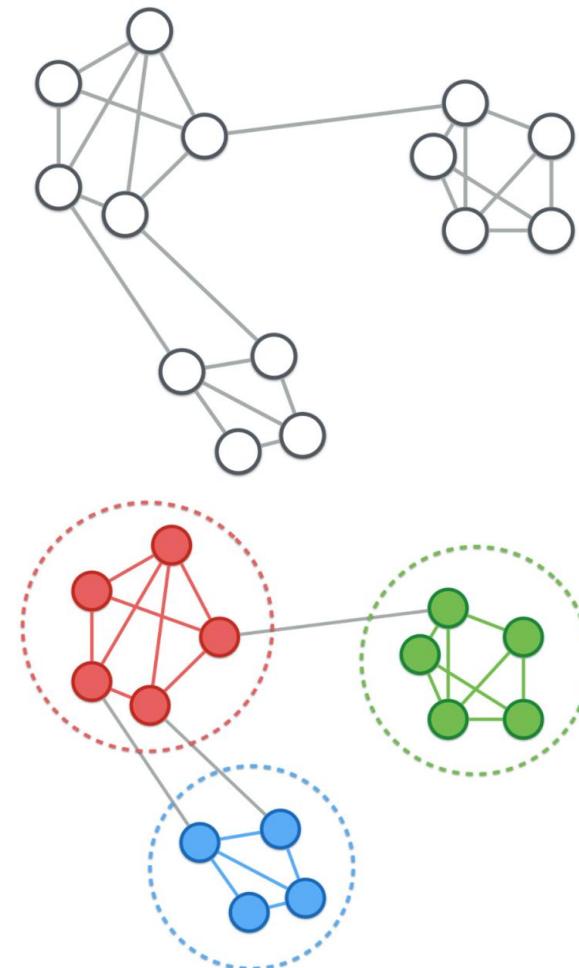
A graph in which two vertices  $p$  and  $q$  are connected by an edge, if the distance between  $p$  and  $q$  is among the  $k$ -th smallest distances from  $p$  to other objects from  $P$ .

- **Shared Nearest Neighbor (SNN) graph**

A graph in which weights define proximity, or similarity between two nodes in terms of the number of neighbors (i.e., directly connected nodes) they have in common.

# Graph clustering (Community detection)

- **Communities (clusters):** groups of nodes with higher probability of being connected to each other than to members of other groups
- **Community detection:** find a group (community) of nodes with more edges inside the group than edges linking nodes of the group with the rest of the graph.
- Algorithms for community detection:
  - Spectral clustering
  - Louvain
  - Markov clustering
  - ...

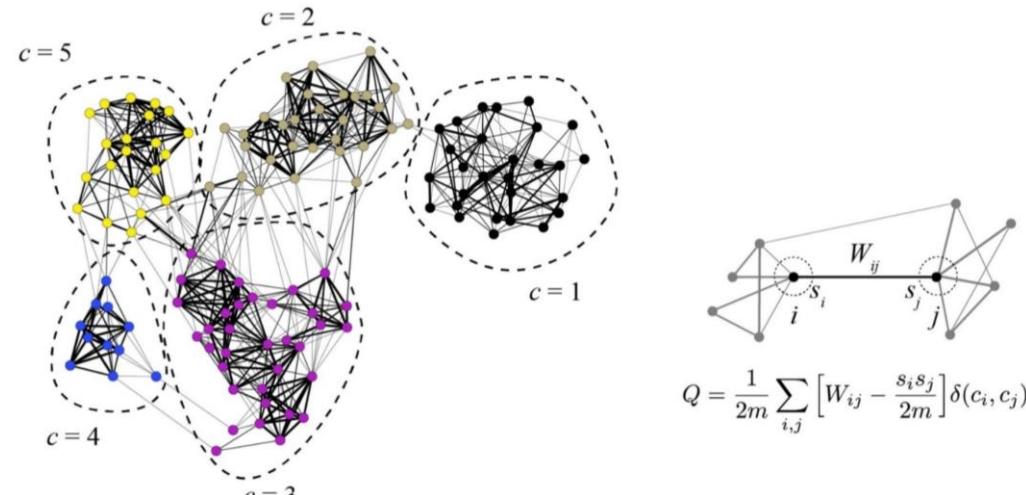
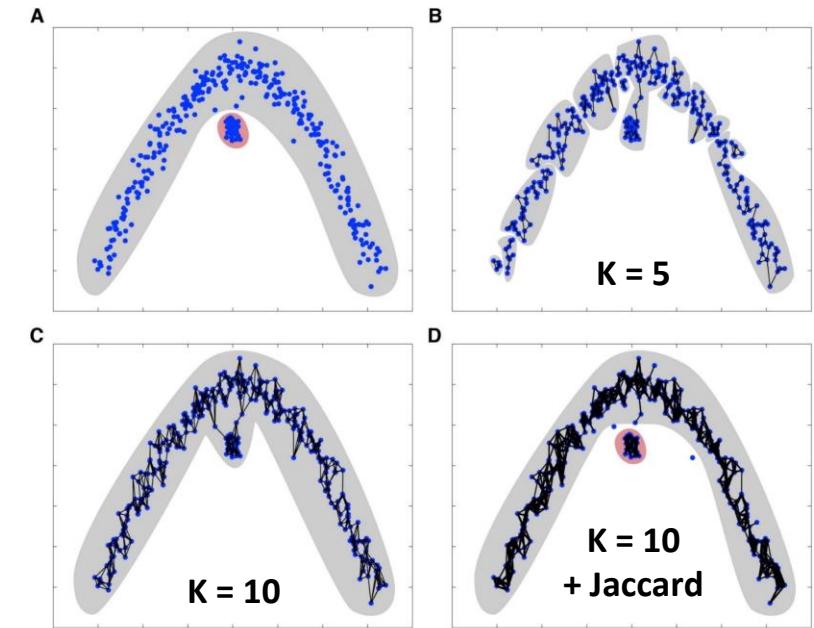


# scRNA-seq clustering methods

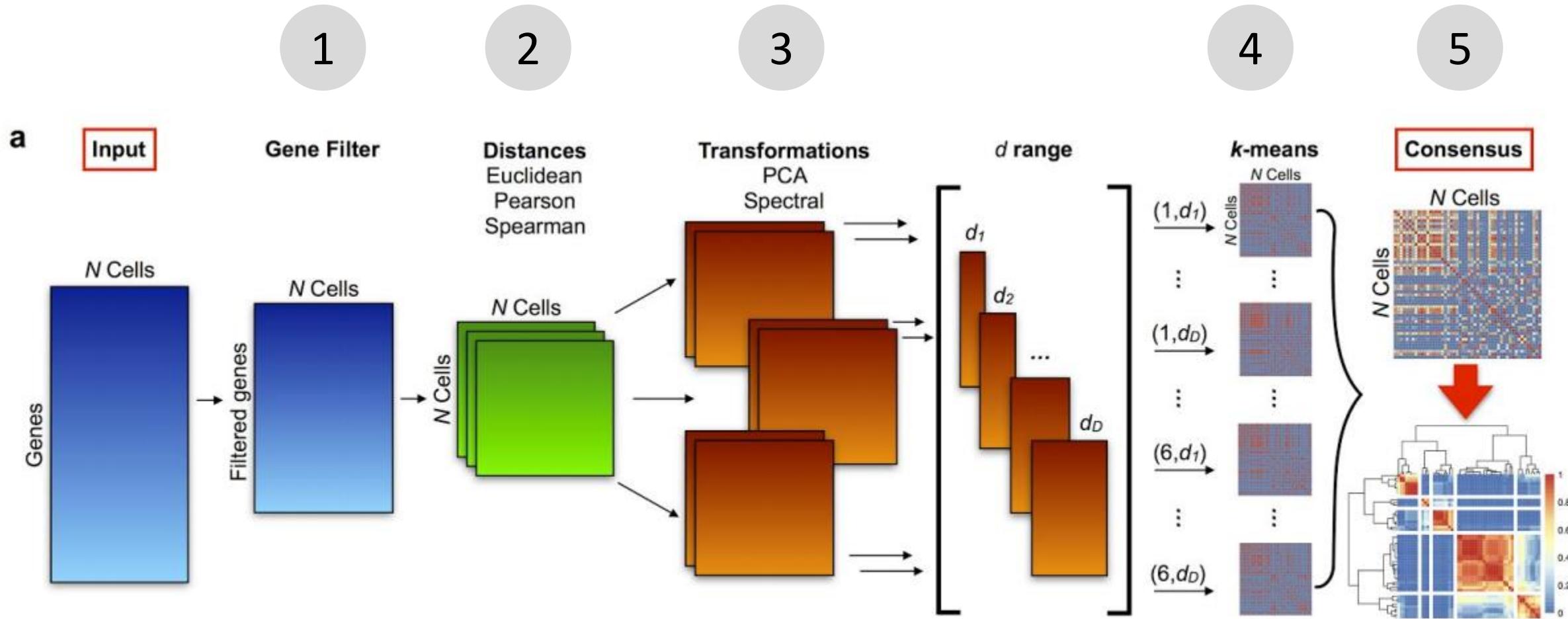
Name	Year	Method type	Strengths	Limitations
scanpy <sup>4</sup>	2018	PCA+graph-based	Very scalable	May not be accurate for small data sets
Seurat (latest) <sup>3</sup>	2016			
PhenoGraph <sup>32</sup>	2015			
SC3 (REF. <sup>22</sup> )	2017	PCA+k-means	High accuracy through consensus, provides estimation of k	High complexity, not scalable
SIMLR <sup>24</sup>	2017	Data-driven dimensionality reduction+k-means	Concurrent training of the distance metric improves sensitivity in noisy data sets	Adjusting the distance metric to make cells fit the clusters may artificially inflate quality measures
CIDR <sup>25</sup>	2017	PCA+hierarchical	Implicitly imputes dropouts when calculating distances	
GiniClust <sup>75</sup>	2016	DBSCAN	Sensitive to rare cell types	Not effective for the detection of large clusters
pcaReduce <sup>27</sup>	2016	PCA+k-means+hierarchical	Provides hierarchy of solutions	Very stochastic, does not provide a stable result
Tasic et al. <sup>28</sup>	2016	PCA+hierarchical	Cross validation used to perform fuzzy clustering	High complexity, no software package available
TSCAN <sup>41</sup>	2016	PCA+Gaussian mixture model	Combines clustering and pseudotime analysis	Assumes clusters follow multivariate normal distribution
mpath <sup>45</sup>	2016	Hierarchical	Combines clustering and pseudotime analysis	Uses empirically defined thresholds and a priori knowledge
BackSPIN <sup>26</sup>	2015	Biclustering (hierarchical)	Multiple rounds of feature selection improve clustering resolution	Tends to over-partition the data
RaceID <sup>23</sup> , RaceID2 (REF. <sup>115</sup> ), RaceID3	2015	k-Means	Detects rare cell types, provides estimation of k	Performs poorly when there are no rare cell types
SINCERA <sup>5</sup>	2015	Hierarchical	Method is intuitively easy to understand	Simple hierarchical clustering is used, may not be appropriate for very noisy data
SNN-Cliq <sup>80</sup>	2015	Graph-based	Provides estimation of k	High complexity, not scalable

# Seurat

- 1) Construct KNN (k-nearest neighbor) graph based on the Euclidean distance in PCA space.
- 2) Refine the edge weights between any two cells based on the shared overlap in their local neighborhoods (Jaccard distance).
- 3) Cluster cells by optimizing for modularity (Louvain algorithm)



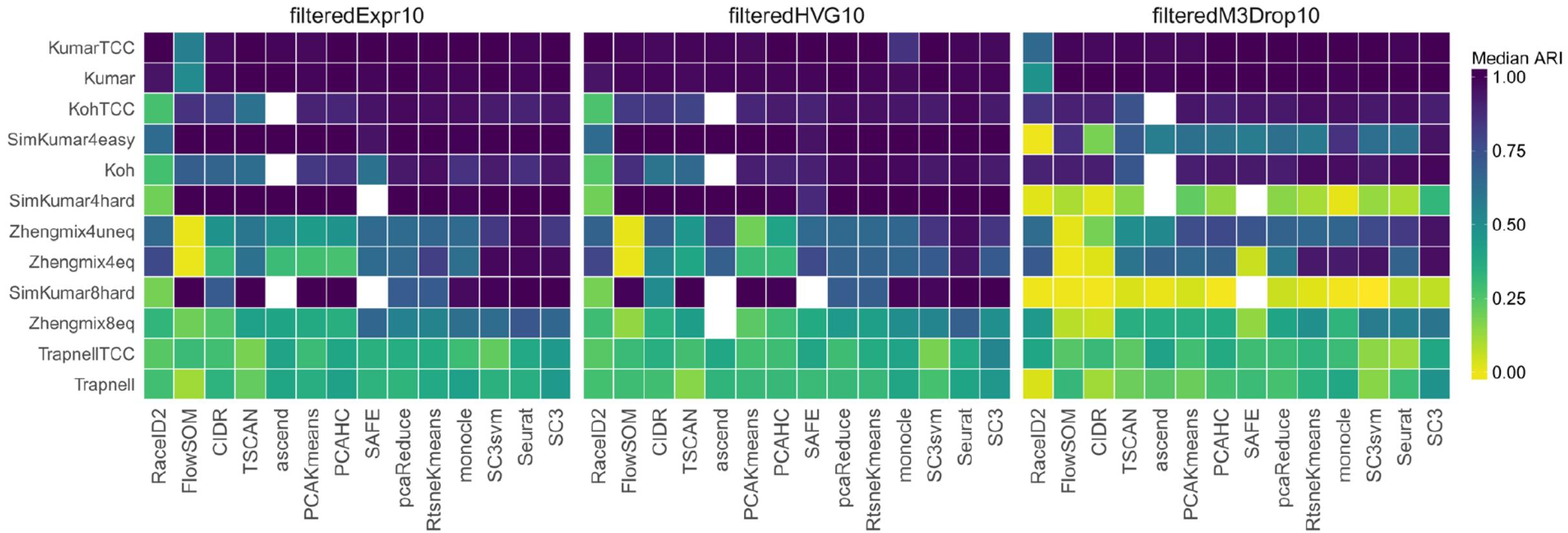
# Single Cell Consensus Clustering – SC3



# Single Cell Consensus Clustering – SC3

- 1) Gene filtering – rare and ubiquitous genes
- 2) Distance matrices (DM) – Euclidean, Spearman, Pearson
- 3) Transformation of DM with PCA or Laplacian
- 4) K-means clustering with first  $d$  eigenvectors
- 5) Consensus clustering – distance 1/0 for cells in same/different clusters -> hierarchical clustering on average distances.

# Benchmarking scRNA-seq clustering methods



# Comparing different clusterings

- Adjusted Rand Index (ARI)

Given a set  $S$  of  $n$  elements, and two groupings or partitions (e.g. clusterings) of these elements  $X = \{X_1, X_2, \dots, X_r\}$  and  $Y = \{Y_1, Y_2, \dots, Y_s\}$

Confusion matrix/contingency table

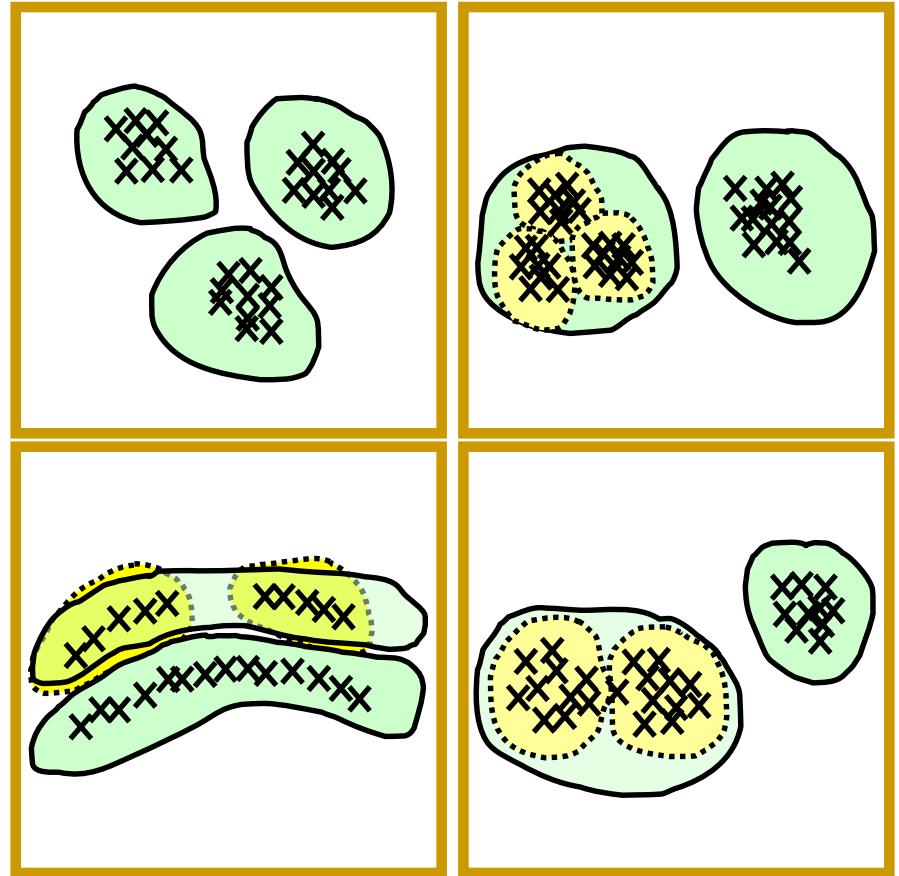
$X \setminus Y$	$Y_1$	$Y_2$	$\dots$	$Y_s$	Sums
$X_1$	$n_{11}$	$n_{12}$	$\dots$	$n_{1s}$	$a_1$
$X_2$	$n_{21}$	$n_{22}$	$\dots$	$n_{2s}$	$a_2$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
$X_r$	$n_{r1}$	$n_{r2}$	$\dots$	$n_{rs}$	$a_r$
Sums	$b_1$	$b_2$	$\dots$	$b_s$	

$$n_{ij} = |X_i \cap Y_j|$$

$$\widehat{ARI} = \frac{\text{Adjusted Index}}{\text{Max Index}} = \frac{\overbrace{\sum_{ij} \binom{n_{ij}}{2} - [\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}] / \binom{n}{2}}^{\text{Index}}}{\underbrace{\frac{1}{2} [\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}]}_{\text{Expected Index}} - \overbrace{[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}] / \binom{n}{2}}^{\text{Expected Index}}}$$

# Clustering is subjective!

- Principle choices
  - Similarity measure
  - Algorithm
- Different choice leads to different results
  - Subjectivity becomes reality
- Cluster process
  - Validate, interpret (generate hypothesis), repeat steps

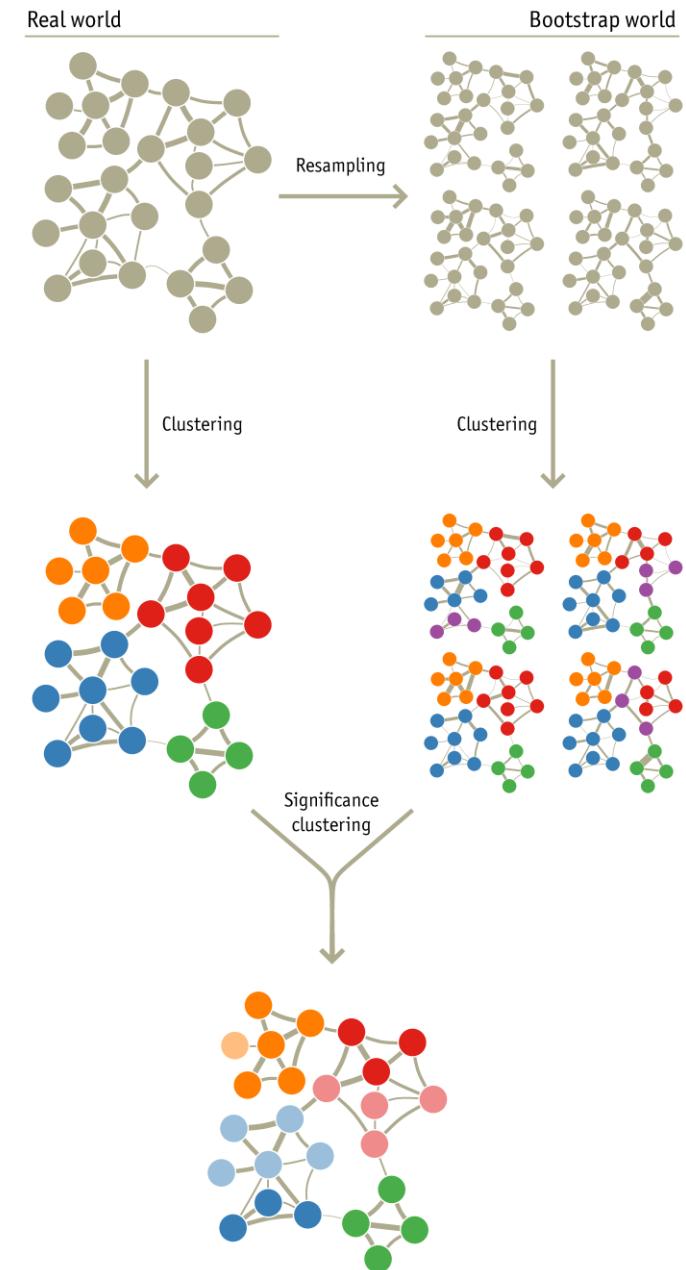


# How many clusters do you really have?

- It is hard to know when to stop clustering – you can always split the cells more times.
- Can use:
  - Do you get any/many significant DE genes from the next split?
  - Some tools have automated predictions for number of clusters – may not always be biologically relevant

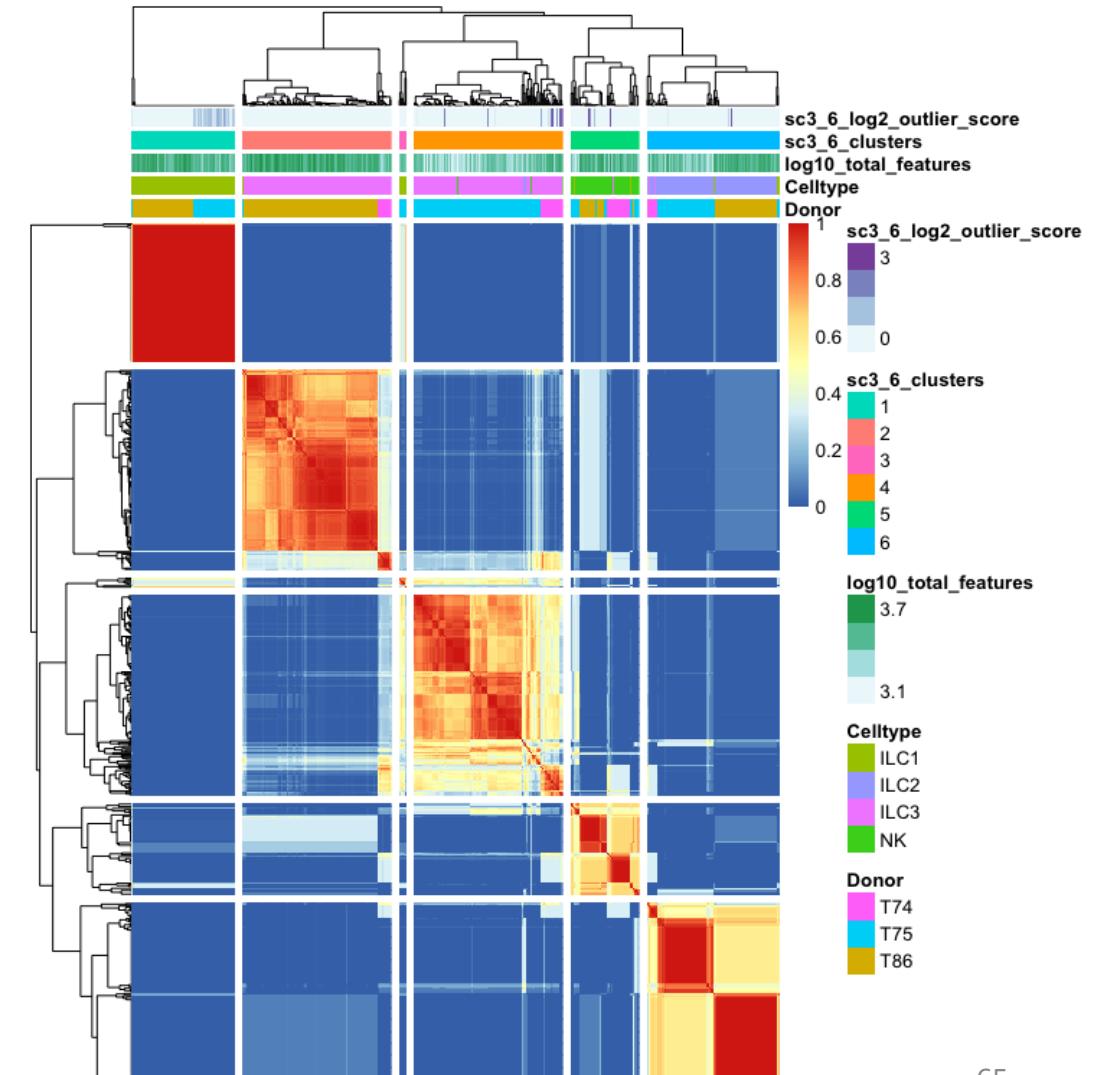
# Bootstrapping

- How confident can you be that the clusters you see are real?
- You can always take a random set of cells from the same cell type and manage to split them into clusters.



# Always check QC data

- Is what your splitting mainly related to batches, qc-measures (especially detected genes)?



# From clusters to cell identities

- Using lists of DE genes and prior knowledge of the biology
- Using lists of DE genes and comparing to other scRNAseq data or sorted cell populations

# Databases with celltype gene signatures

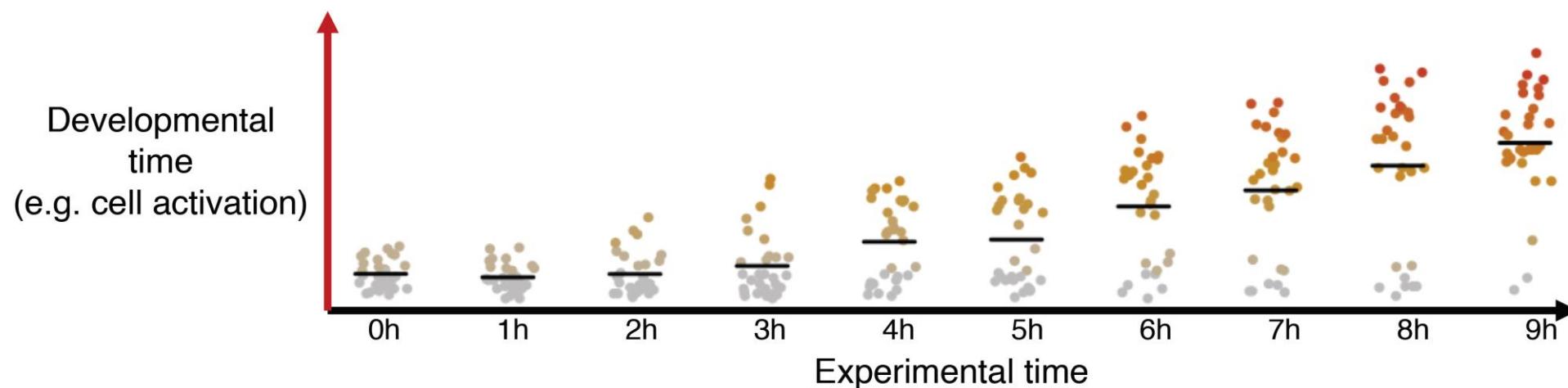
- PanglaoDB (<https://panglaodb.se/>)
  - Human: 295 samples, 72 tissues, 1.1 M cells
  - Mouse: 976 samples, 173 tissues, 4 M cells
  - Franzén et al (<https://doi.org/10.1093/database/baz046>)
- CellMarker (<http://biocc.hrbmu.edu.cn/CellMarker/>)
  - Human: 13,605 cell markers of 467 cell types in 158 tissues
  - Mouse: 9,148 cell makers of 389 cell types in 81 tissues
  - Zhang et al. (<https://doi.org/10.1093/nar/gky900>)

# Challenges in clustering

- What is a cell type?
- What is the number of clusters  $k$ ?
- **Scalability:** in the last few years the number of cells in scRNA-seq experiments has grown by several orders of magnitude from  $\sim 10^2$  to  $\sim 10^6$

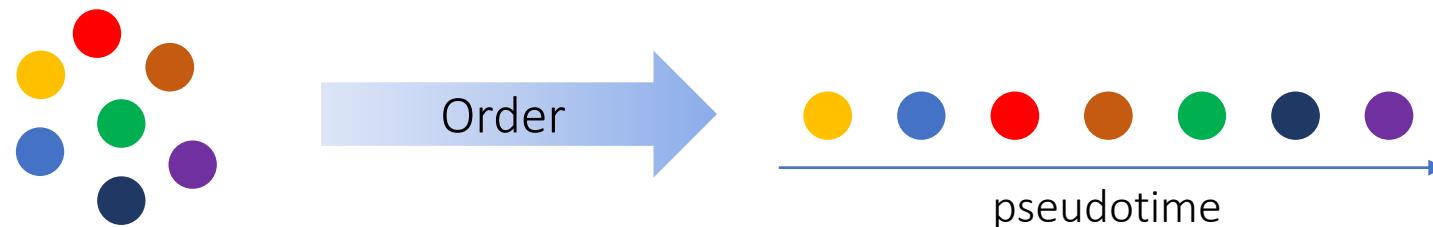
# What is trajectory inference / pseudotime?

- Cells that differentiate display a continuous spectrum of states
- Individual cells will differentiate in an unsynchronized manner
  - > Each cell is a snapshot of differentiation time



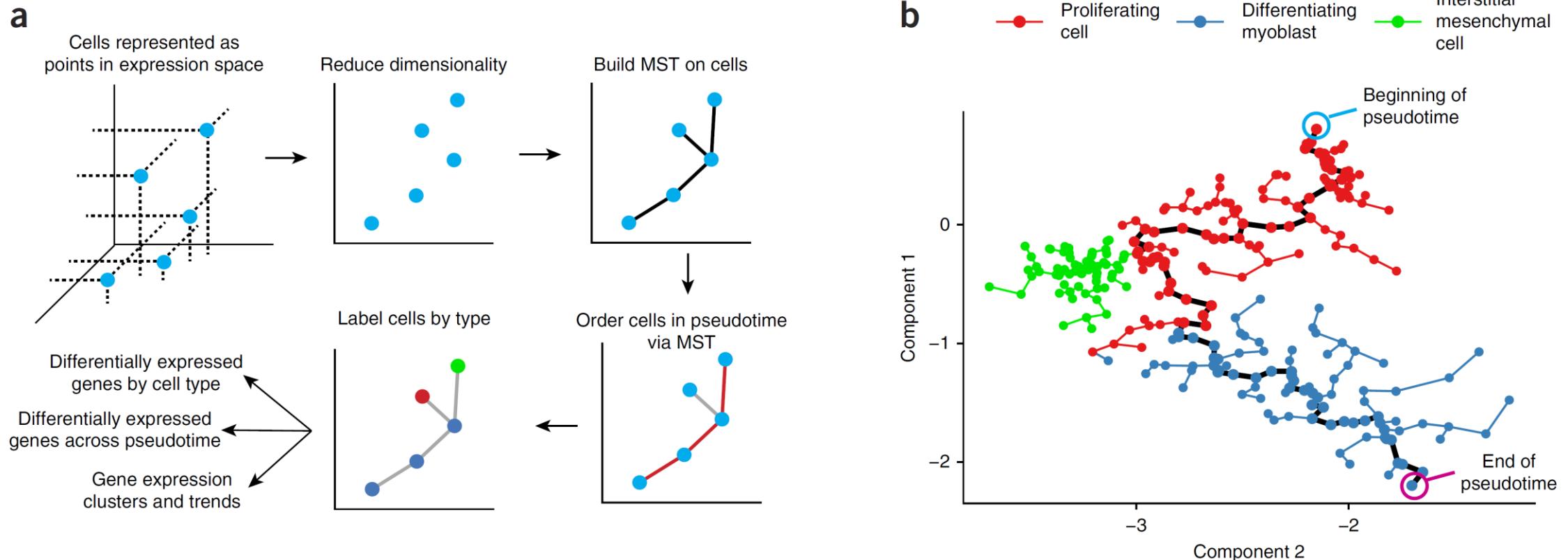
# Trajectory Inference

- Pseudotime: artificial measure of a cell's progression through some process (e.g. differentiation) from scRNA-seq snapshot data
- Key assumptions:
  - continuity of transcriptome changes
  - presence of all intermediate cell stages

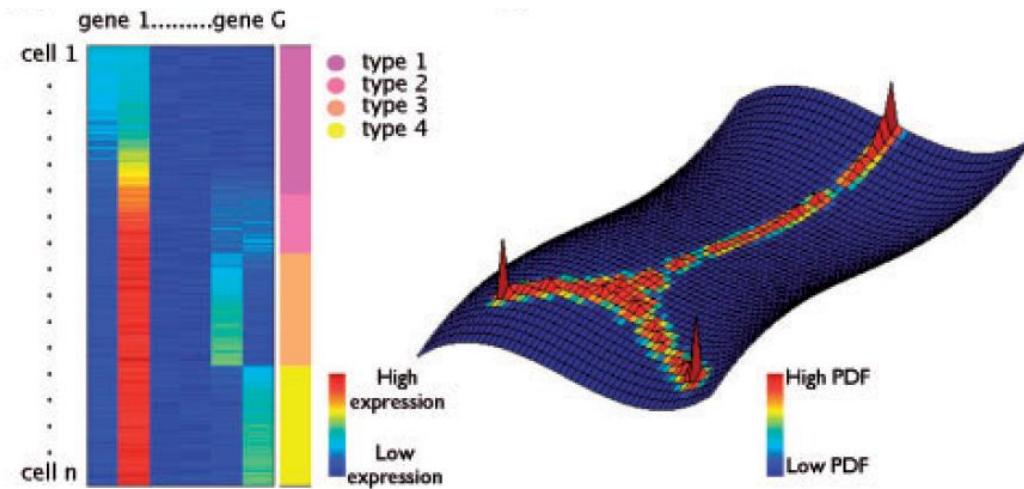


# Monocle

(v1 based on minimum spanning trees)

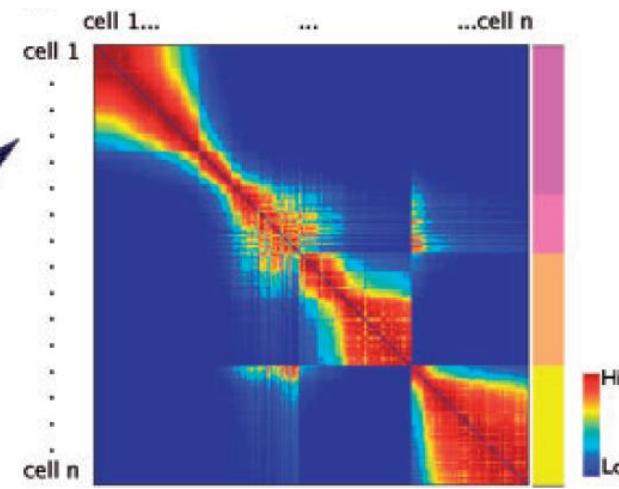


# Diffusion Maps

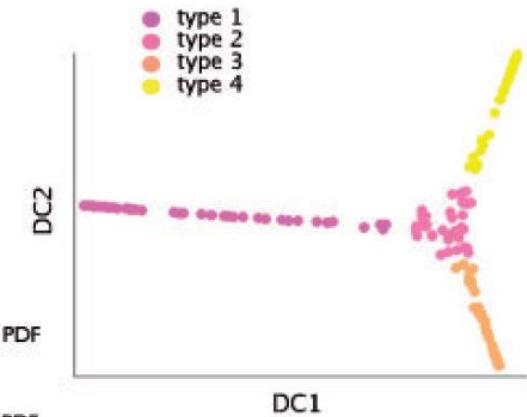


Cell X Gene matrix

Representation of each cell by a Gaussian in the  $G$ -dimensional gene space

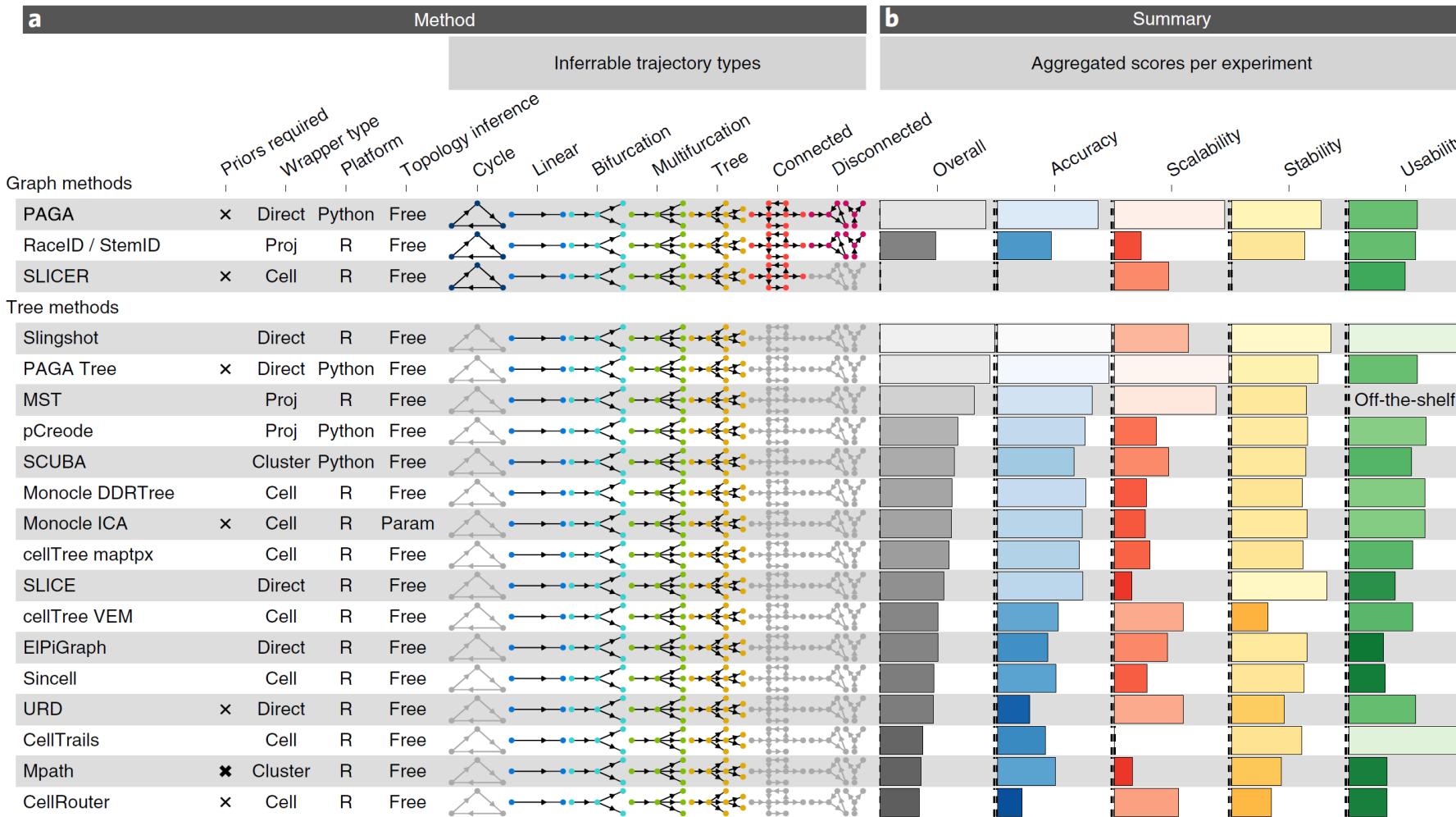


Cell X Cell Markovian transition probability matrix

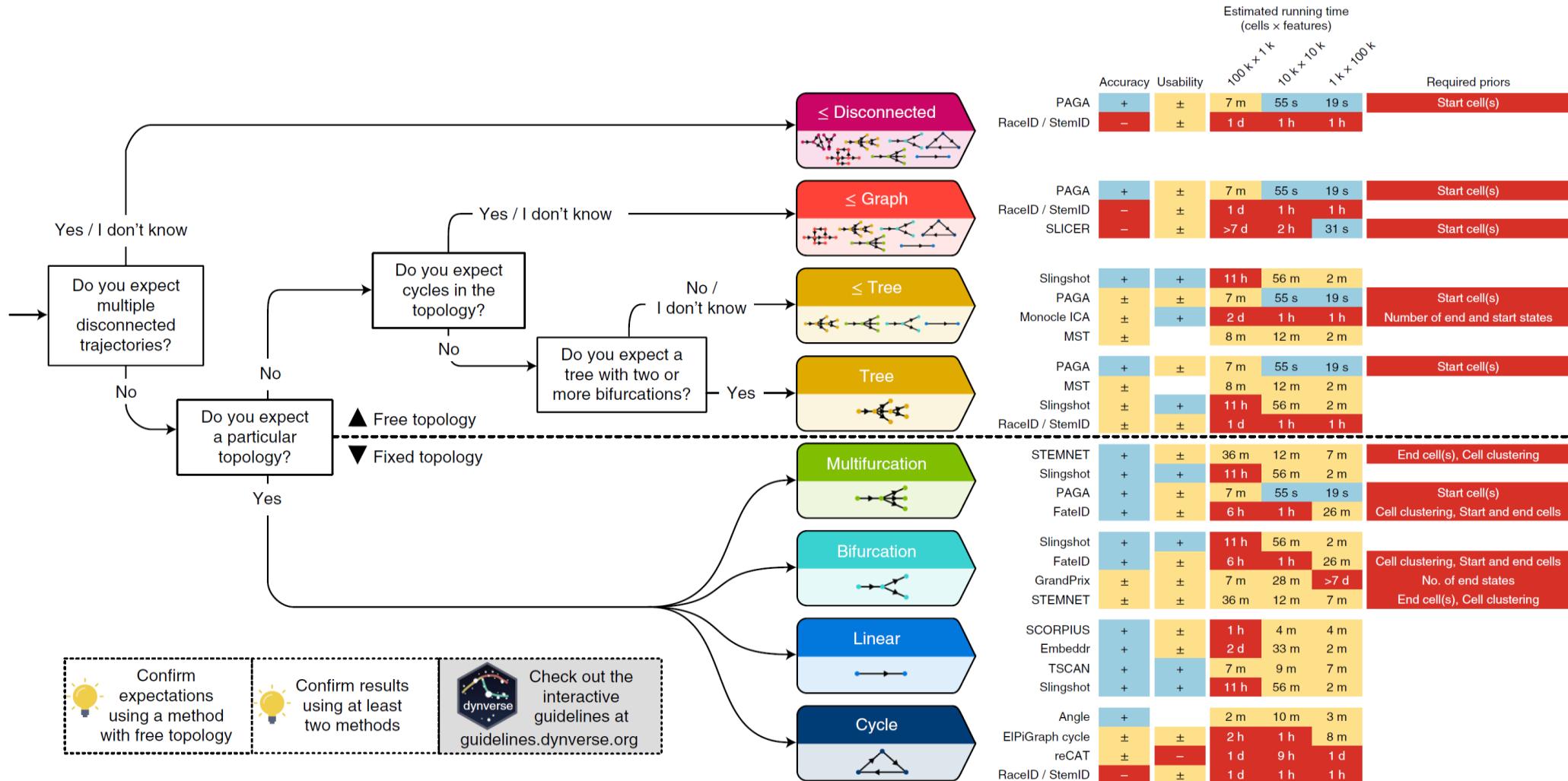


Data embedding on the first two eigenvectors of the Markovian transition matrix

# Trajectory inference methods



# Trajectory inference methods



# Summary

- Brief introduction to single cell RNA-sequencing
- Analysis workflow:
  - Quality control
  - Normalization
  - Feature selection
  - Dimensionality reduction
  - Clustering
  - Trajectory inference

# There is more to it...

- Constructing the cell x gene matrix
- Batch correction methods
- Automatic cell type identification (classification methods)
- Single cell regulatory networks
- Imputation
- Single cell multi-omics
- Sample multiplexing
- Single cell isoform sequencing
- Cell lineage + scRNA-seq
- Spatial transcriptomics
- ...

# Useful Resources

- Orchestrating Single-Cell Analysis with Bioconductor

<https://osca.bioconductor.org/>

- Single Cell Course (Martin Hemberg Lab, Wellcome Trust Sanger):

<http://hemberg-lab.github.io/scRNA.seq.course>

- Aaron Lun's single cell workflow (very detailed):

<https://www.bioconductor.org/packages/release/workflows/html/simpleSingleCell.html>

- GitHub: Awesome Single Cell

<https://github.com/seandavi/awesome-single-cell>

- Recent developments in single cell genomics

[https://www.dropbox.com/s/woya6ffgq8a3pkw/SingleCellGenomicsDay18\\_References.pdf?dl=1](https://www.dropbox.com/s/woya6ffgq8a3pkw/SingleCellGenomicsDay18_References.pdf?dl=1)

# Thank You

-  a.mahfouz@lumc.nl
-  <https://www.lcbc.nl/>
-  @ahmedElkoussy

## ANNOUNCEMENT

MGC course ' Single Cell analysis '

Date: 14 – 18 October, 2019

Location: Erasmus MC

Registration: <https://forms.lumc.nl/lumc2/RegistrationSingleCell>

[More information](#)

