Single Cell RNA-seq Analysis

Ahmed Mahfouz
Department of Human Genetics, Leiden University Medical Center
Leiden Computational Biology Center
Pattern Recognition and Bioinformatics, TU Delft

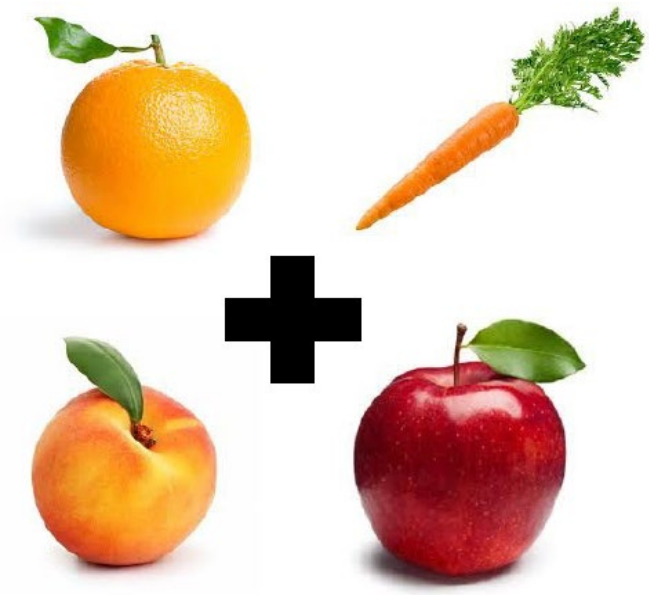
@ahmedElkoussy

Leiden
Computational Biology Center

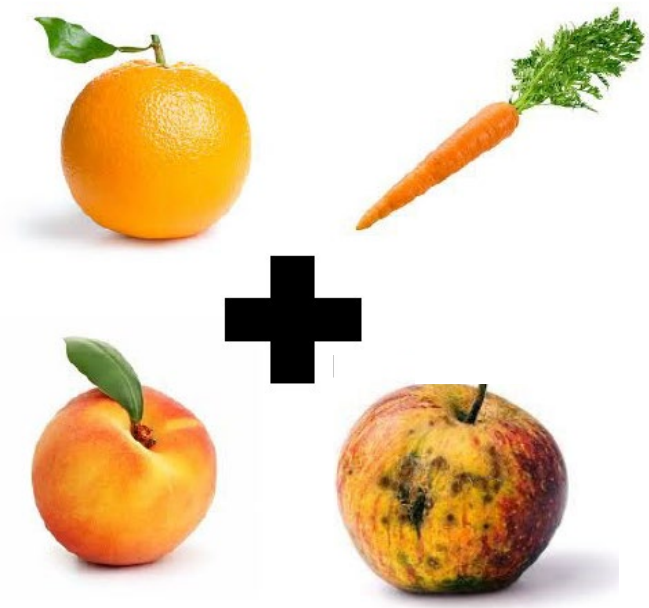
LUMC

TU Delft

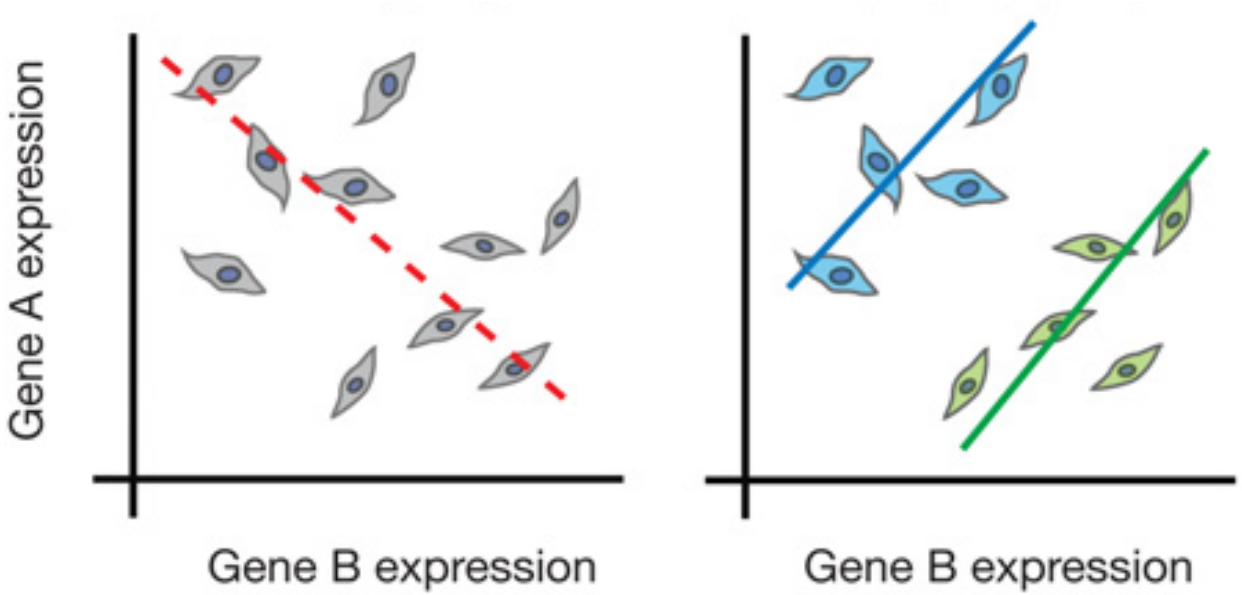
Why single cells?



Why single cells?



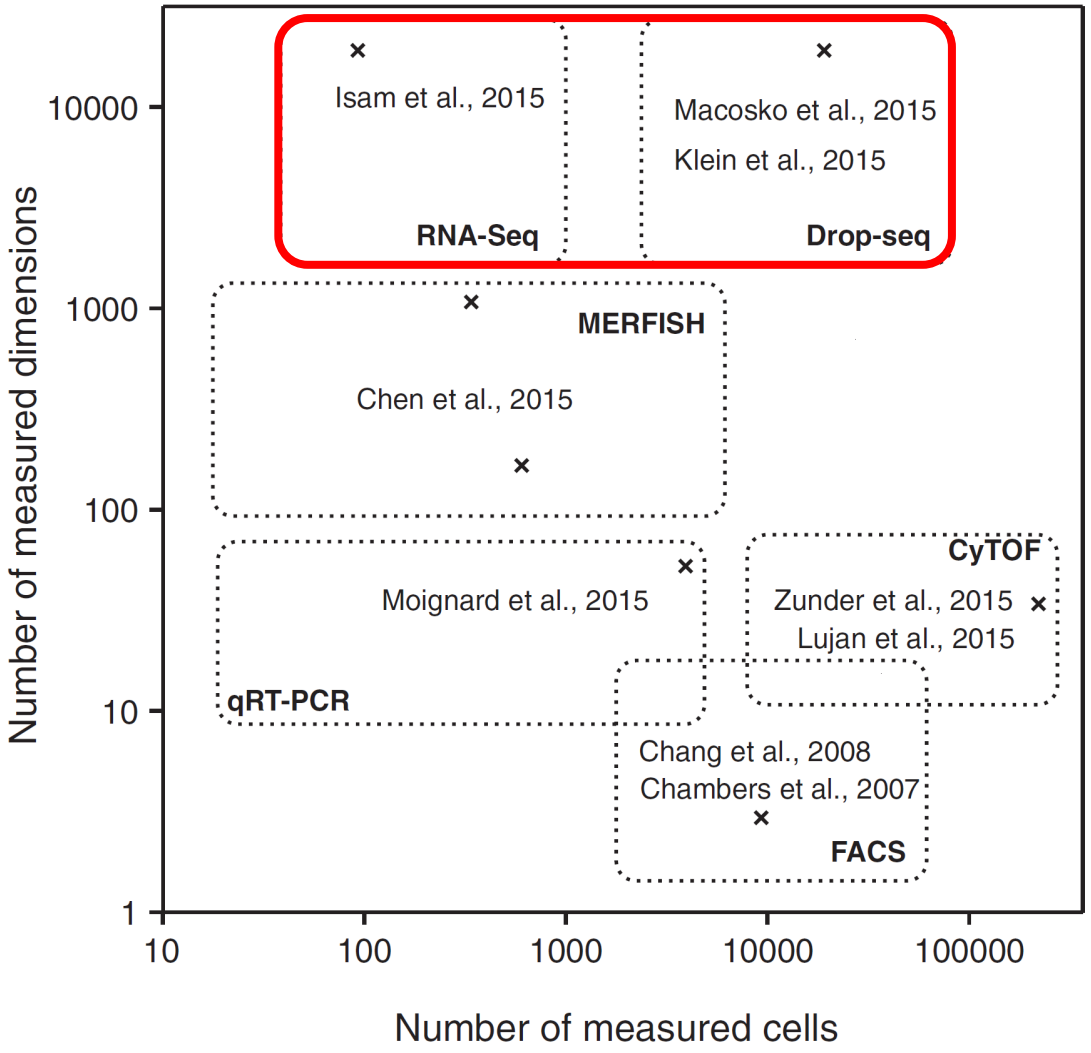
Simpson's Paradox



Simpson's Paradox describes the misleading effects that arise when averaging signals from multiple individuals.

Trapnell (Genome Research 2015)

How can we study single cells?



Every method has it's pros and cons.

Marr et al. *Curr Op Biotech* 2016

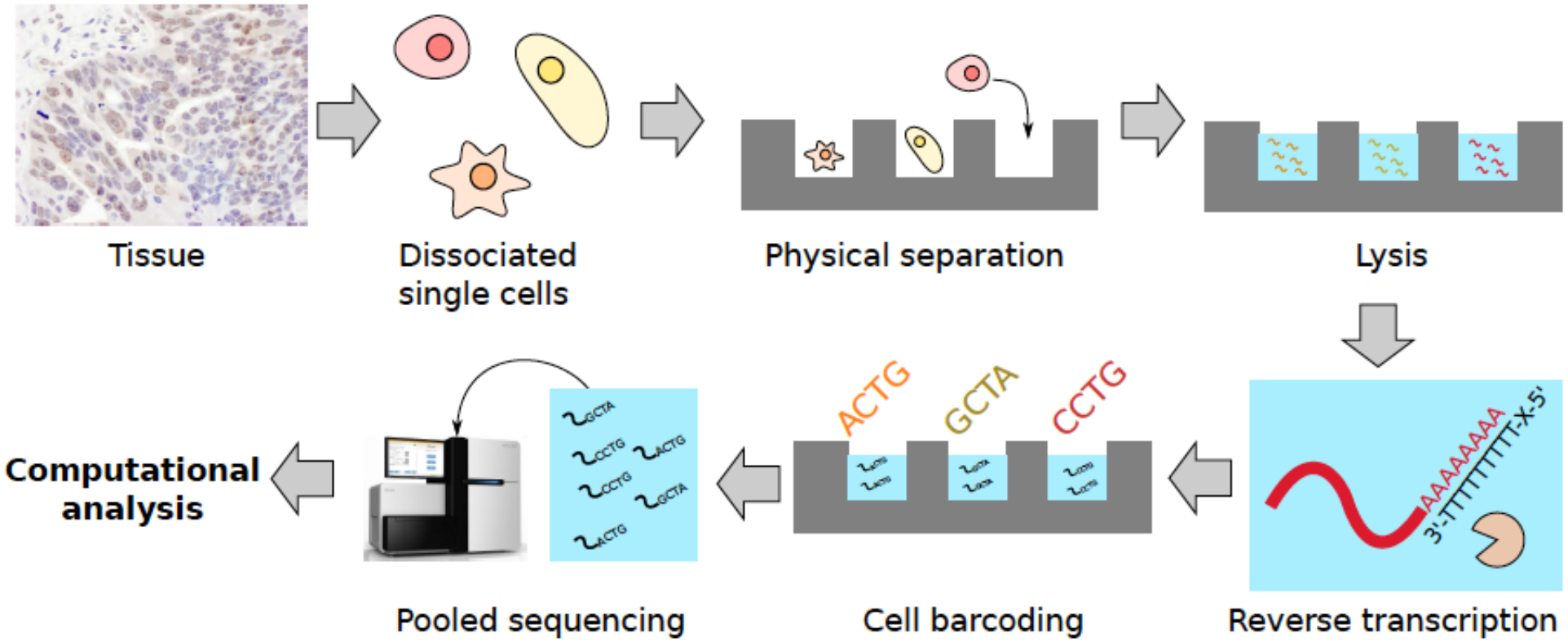
Goals of today

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

Agenda

09:00 – 10:00	**Lecture part 1:** quality control, normalization, batch correction, feature selection
10:00 – 10:15	---*Break*---
10:15 – 12:00	**Practical part 1:** quality control, normalization, batch correction, feature selection
12:00 – 13:00	---*Lunch*---
13:00 – 14:00	**Lecture part 2:** dimensionality reduction, cell type identification
14:15 – 16:00	**Practical part 2:** dimensionality reduction, cell type identification
16:00 – 17:00	**Lecture part 3:** research examples

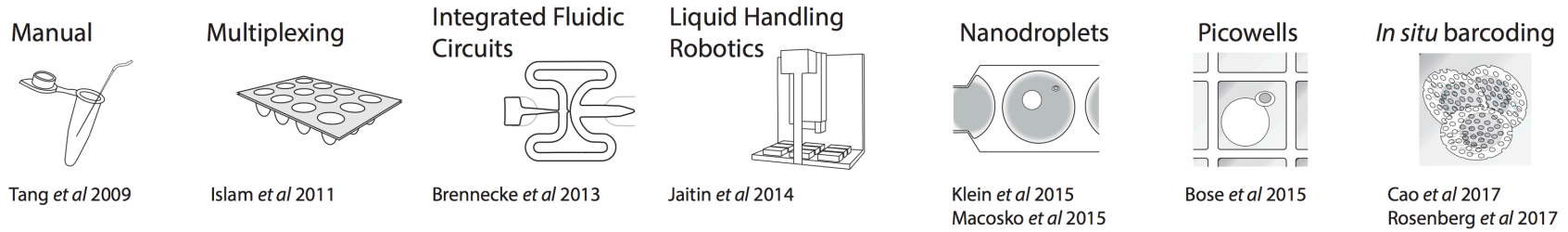
Single cell RNA-sequencing (scRNA-seq)

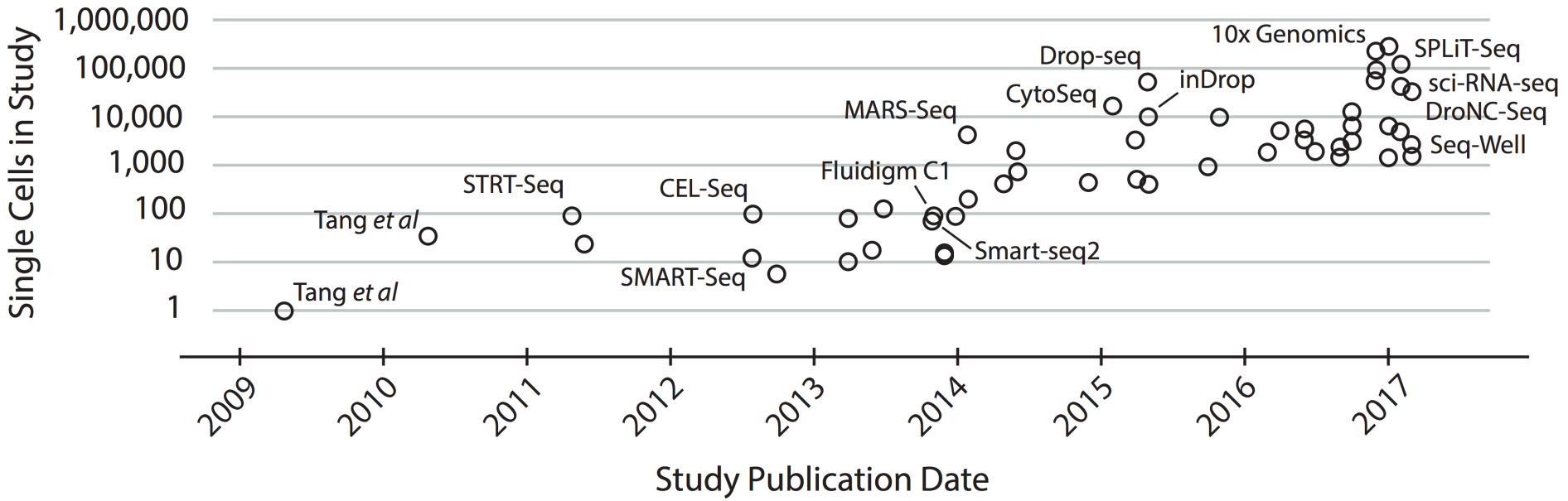


Aaron Lun (Marioni Group, EMBL-EBI)

scRNA-seq Protocols

Number of cells

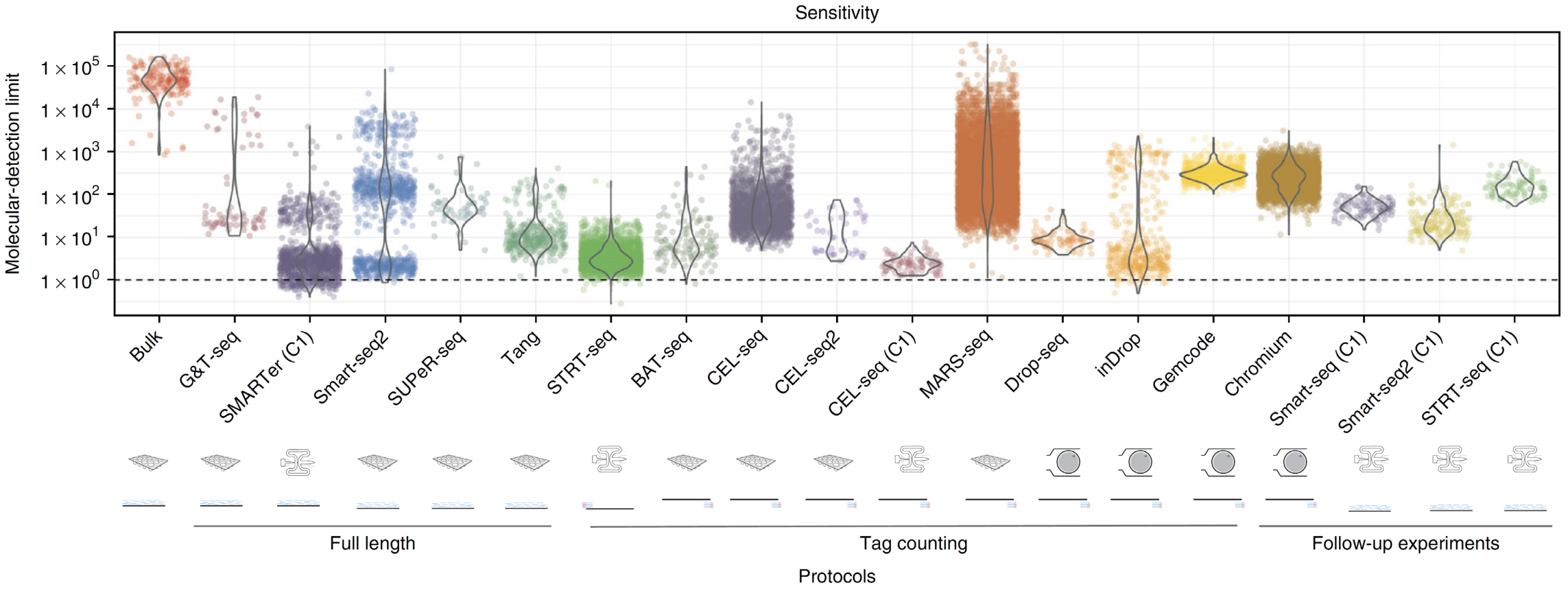




Svensson et al. Nature Methods 2017

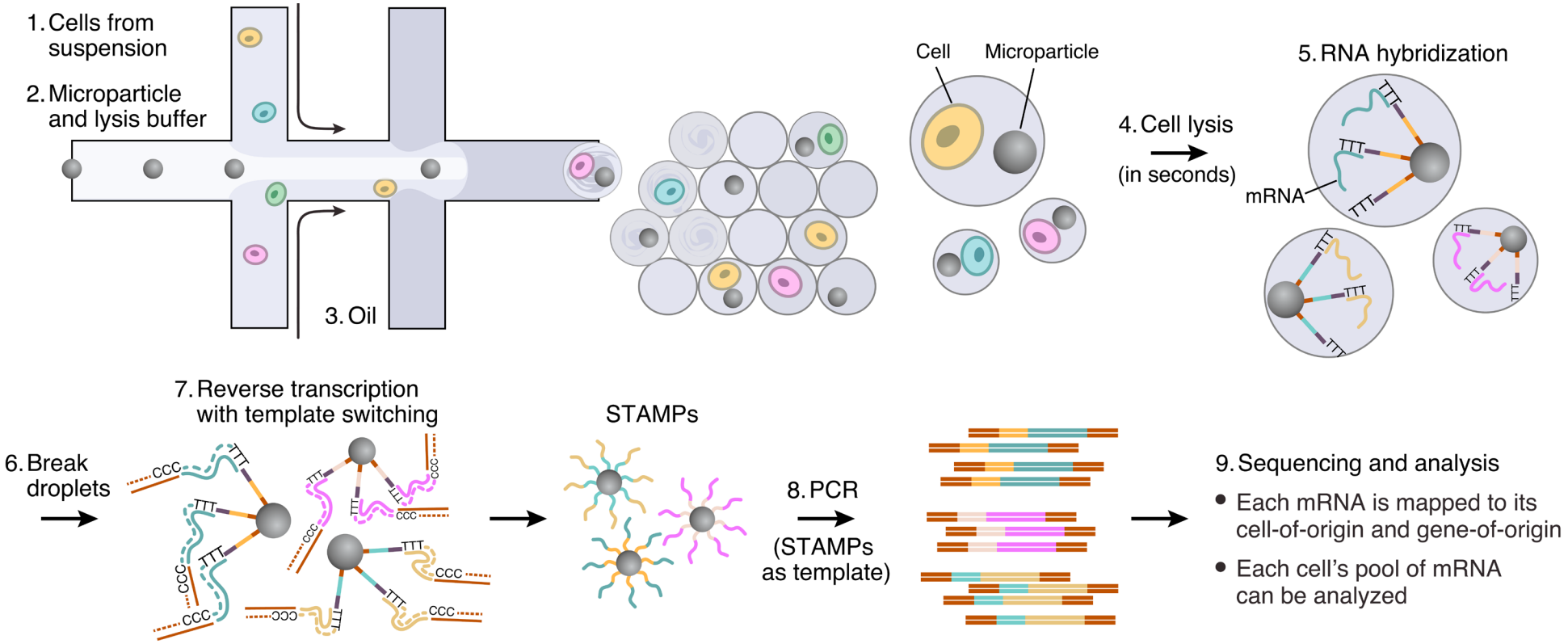
scRNA-seq Protocols

Sensitivity

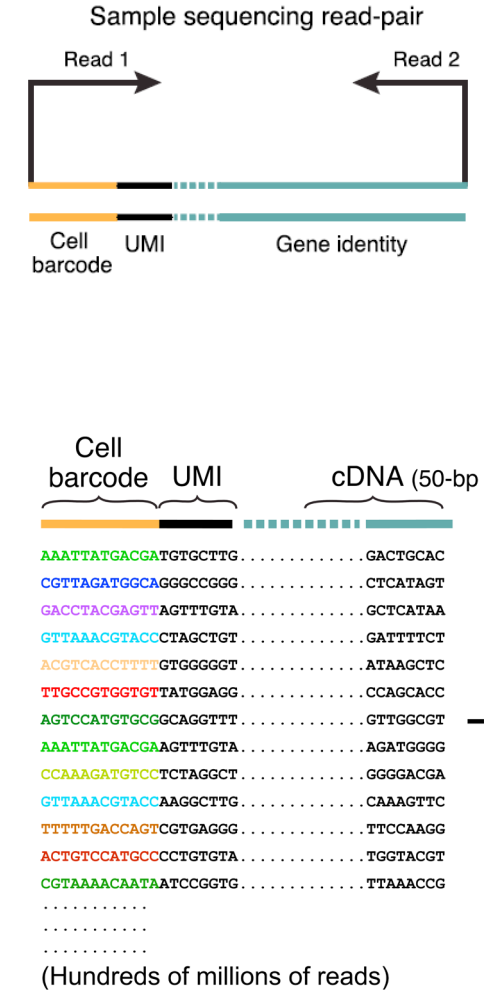


Svensson et al. Nature Methods 2017

Drop-seq



STAMPs: Single-cell transcriptomes attached to microparticles

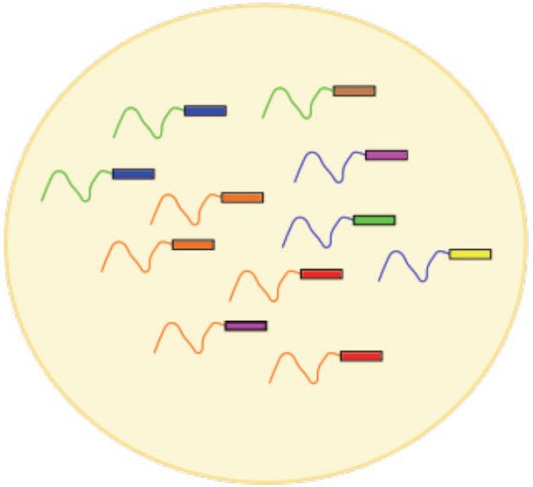


Macosko et al., Cell 2015

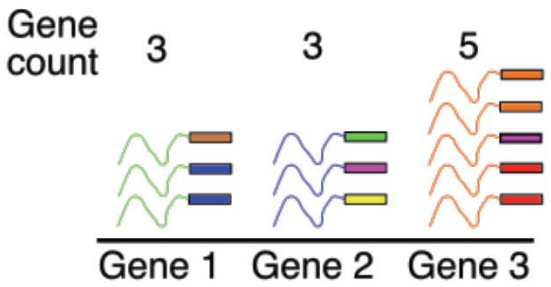
Unique Molecular Identifiers (UMIs)

- 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.

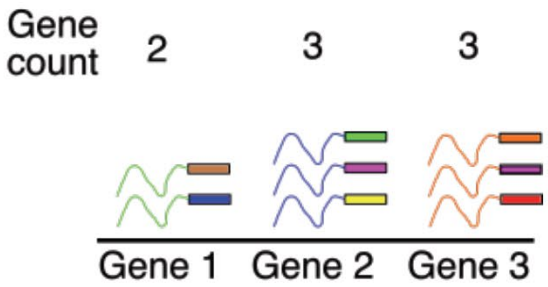
Sequenced fragments from an individual cell



Pre de-duplication

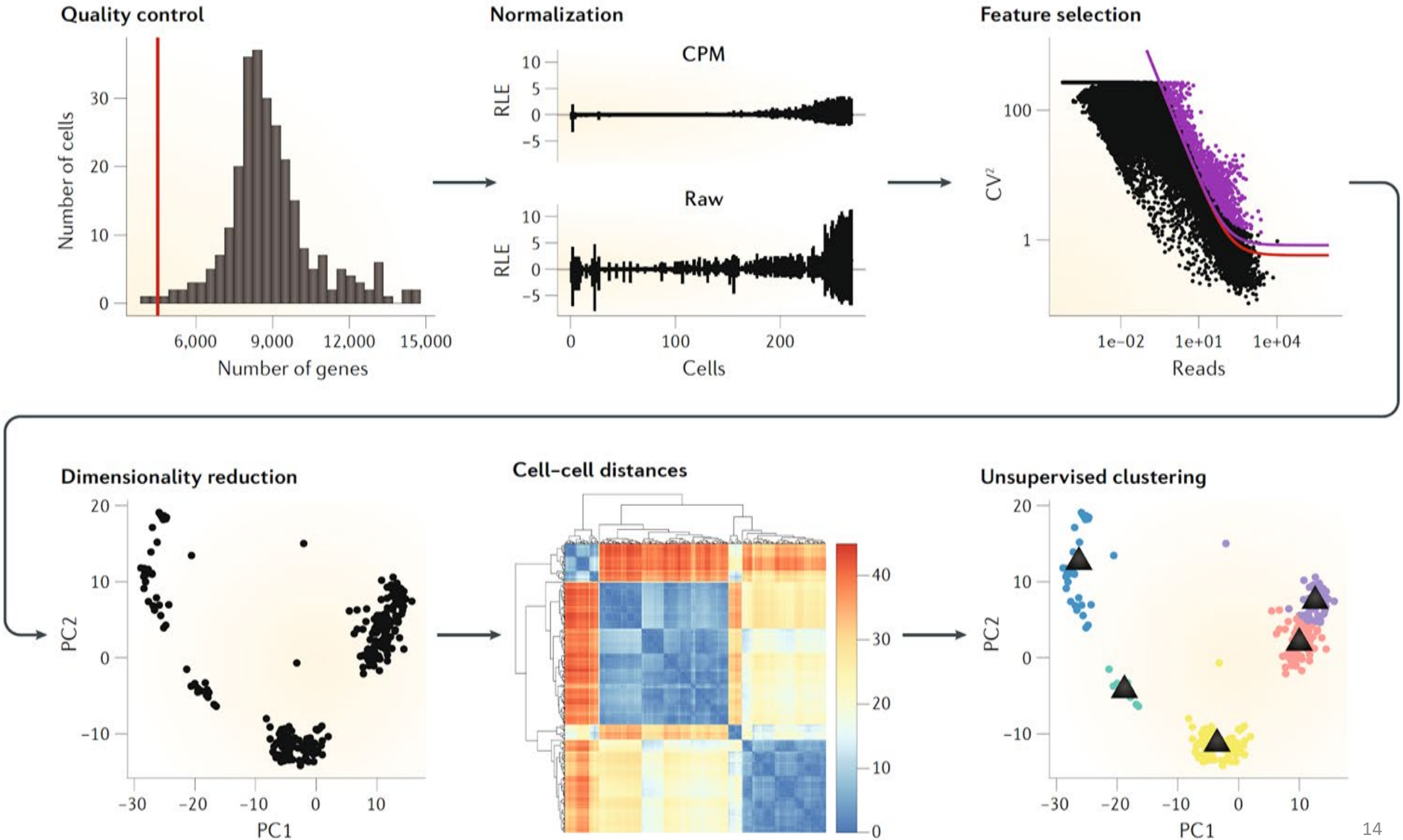


Post de-duplication

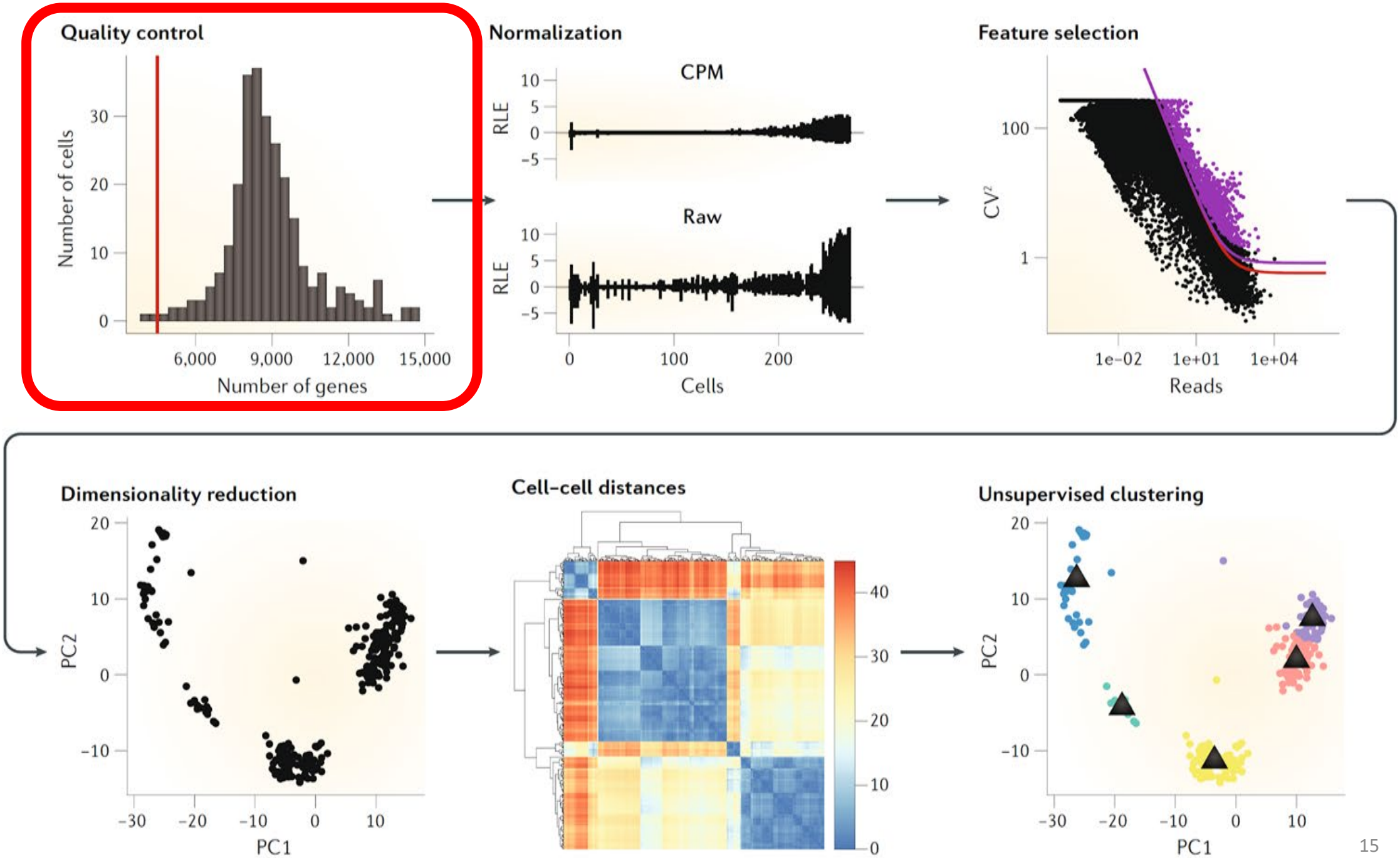


scRNA-seq Data Analysis

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



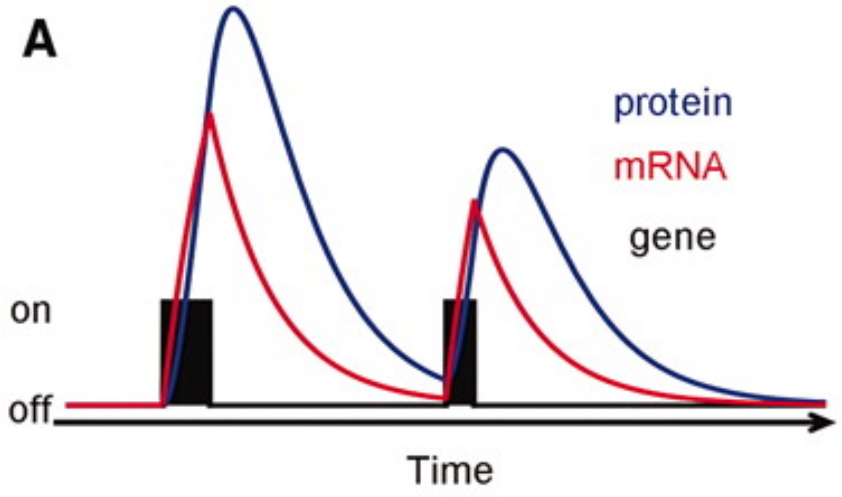
Kiselev et al. (Nature review Genetics 2019)

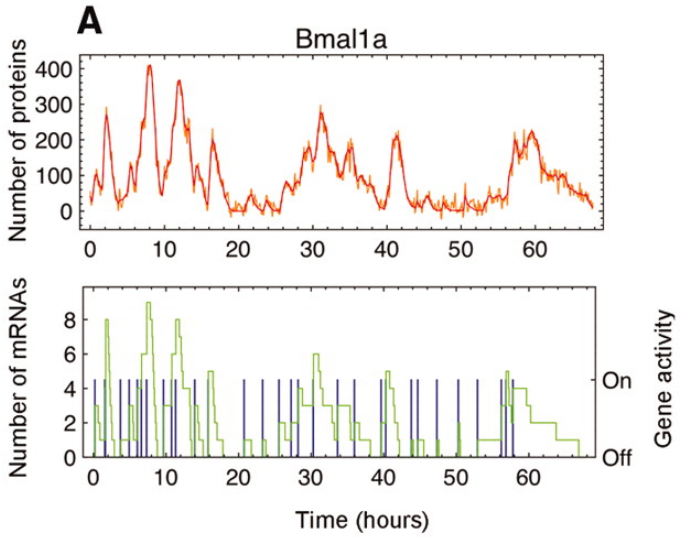


Kiselev et al. (Nature review Genetics 2019)

Transcriptional bursting

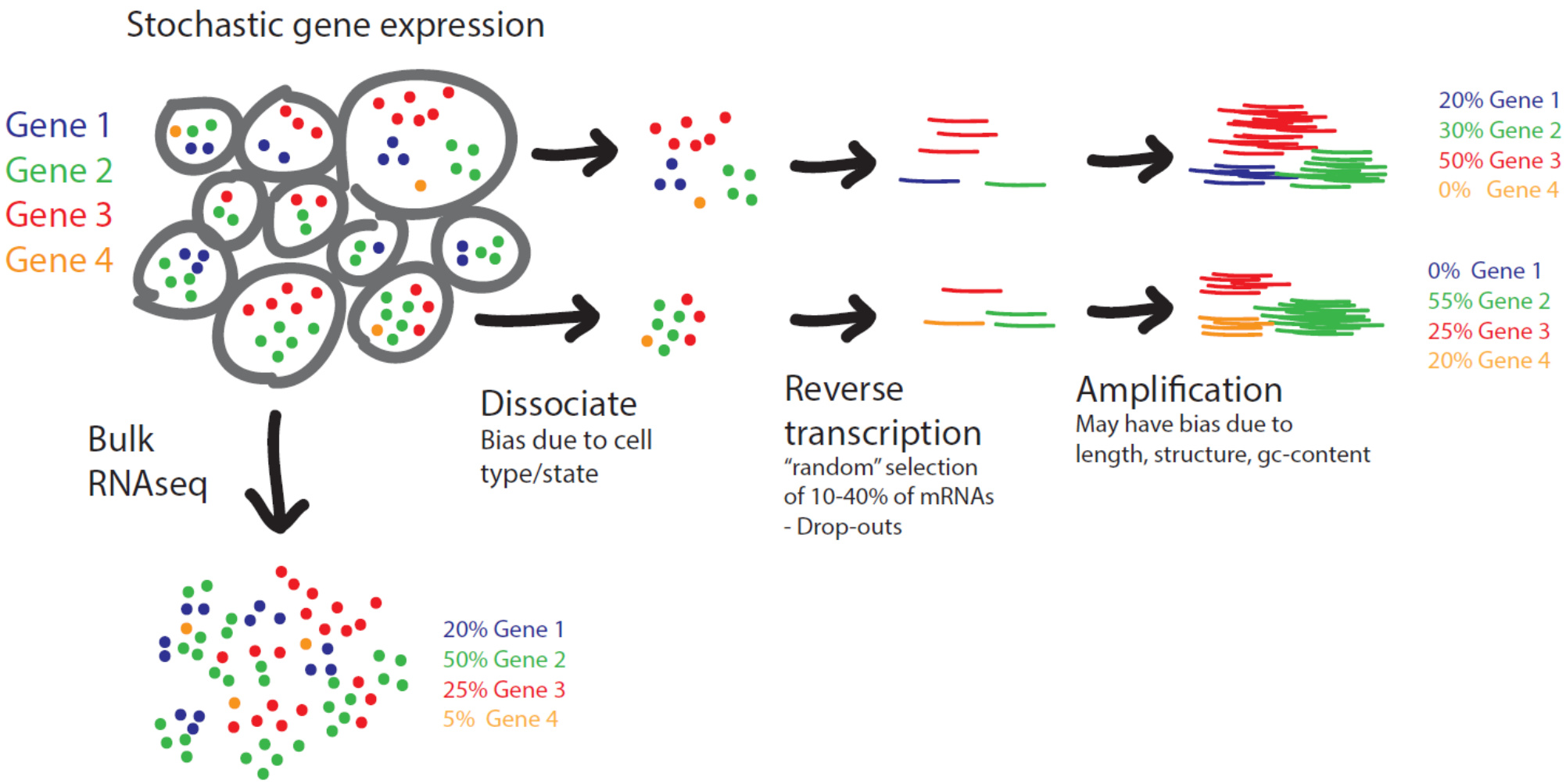
- Burst frequency and size is correlated with mRNA abundance
- Many TFs have low mean expression (and low burst frequency) and will only be detected in a fraction of the cells





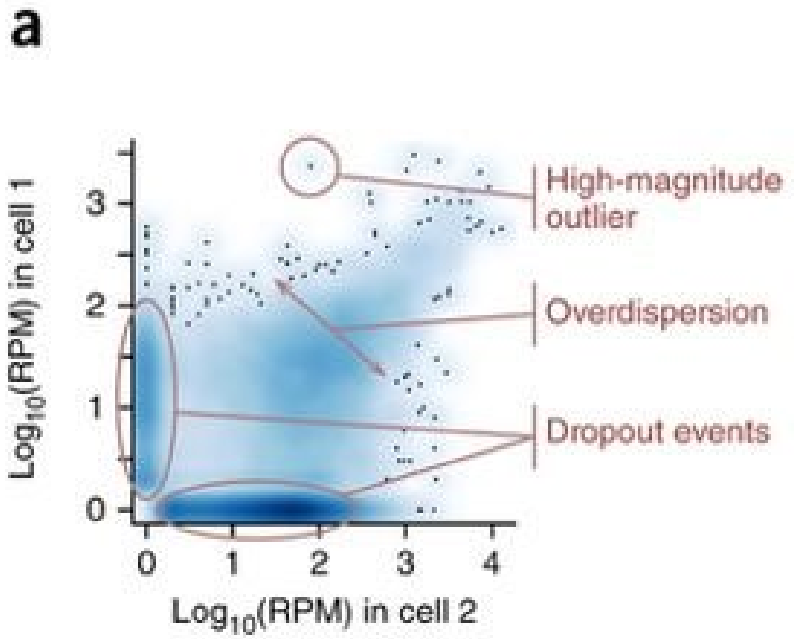
Suter et al. (Science 2011)

Bursting, drop-outs and amplification bias



Problems compared to bulk RNA-seq

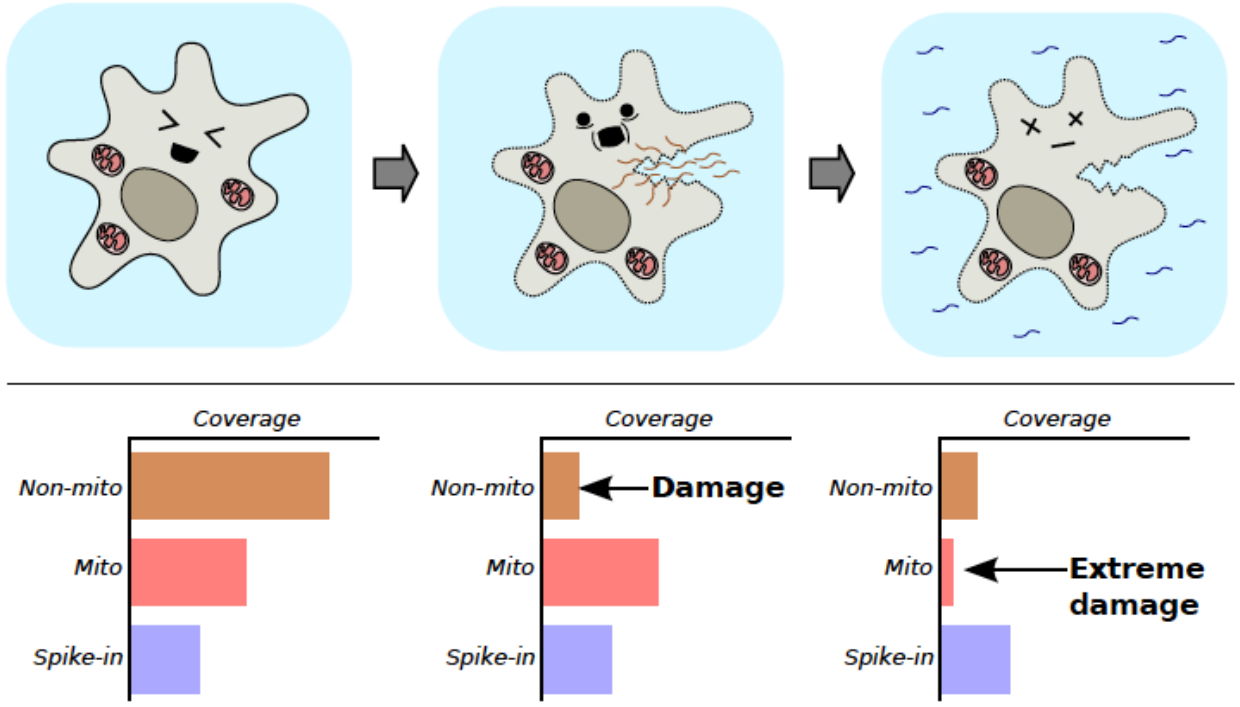
- Amplification bias
- Drop-out rates
- Transcriptional bursting
- Background noise
- Bias due to cell-cycle, cell size and other factors
- Often clear batch effects



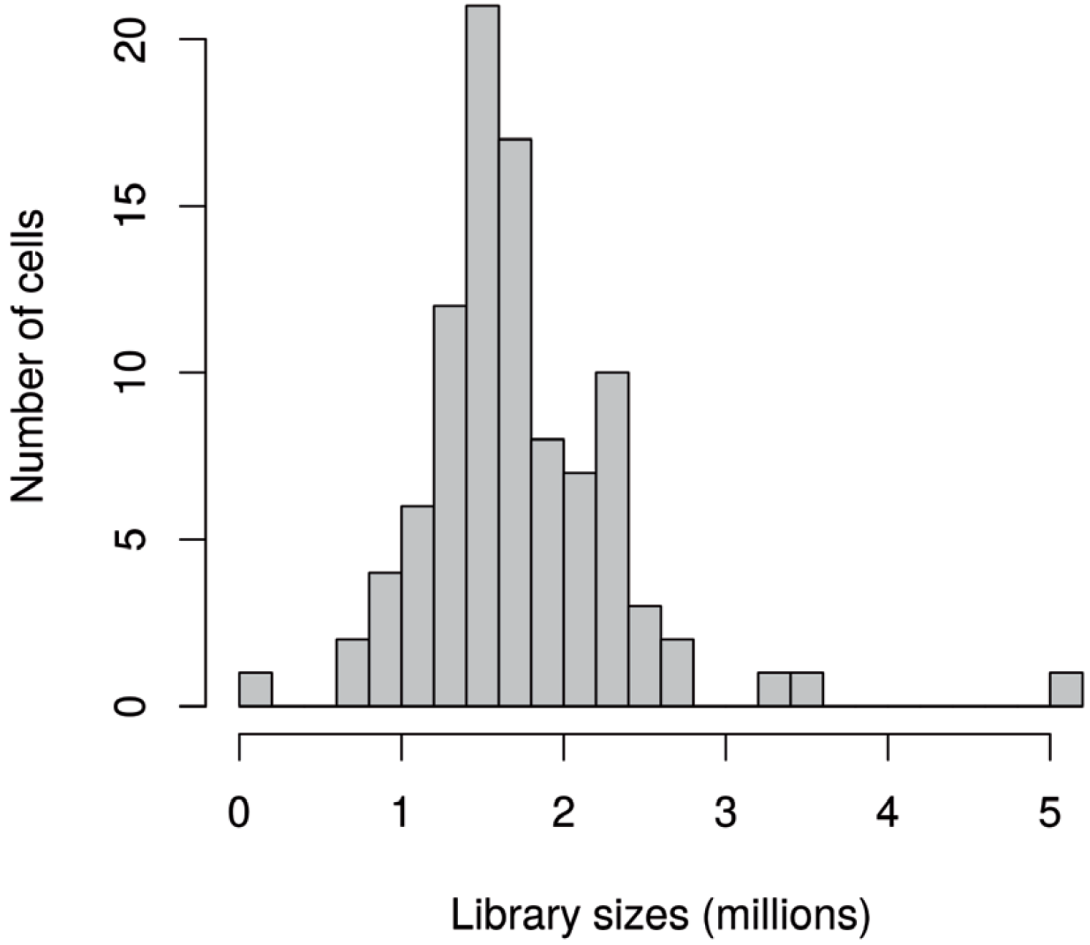
Karchenko et al. (Nature Methods 2014)

Quality control of cells

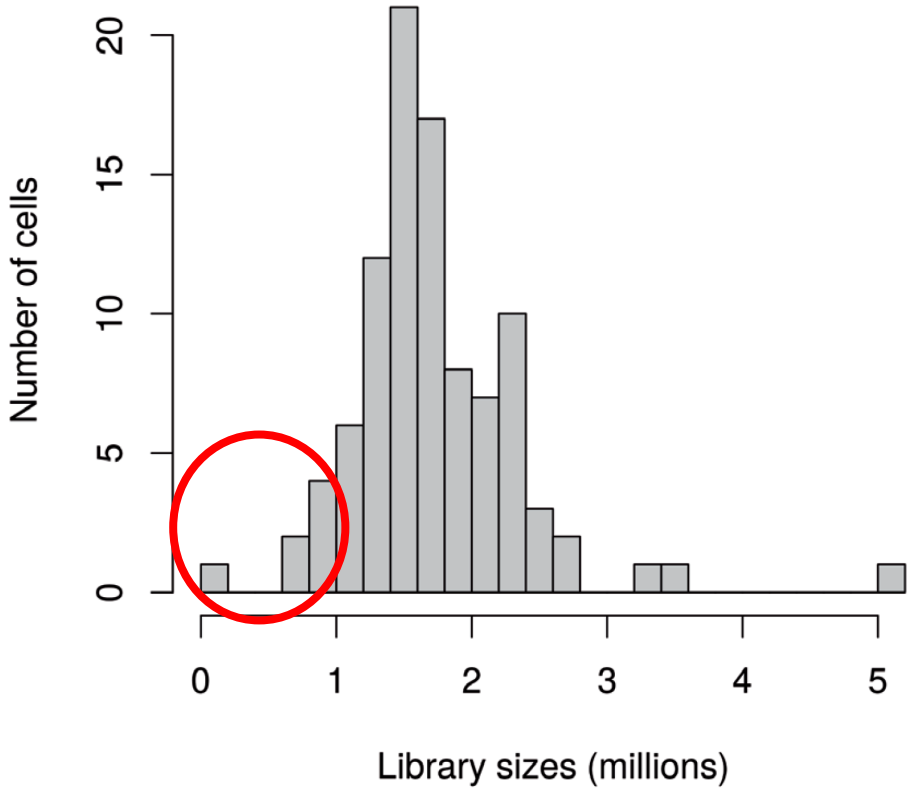
- Low sequencing depth
- Low numbers of expressed genes (i.e. any nonzero count)
- High spike-in (if present) or mitochondrial content



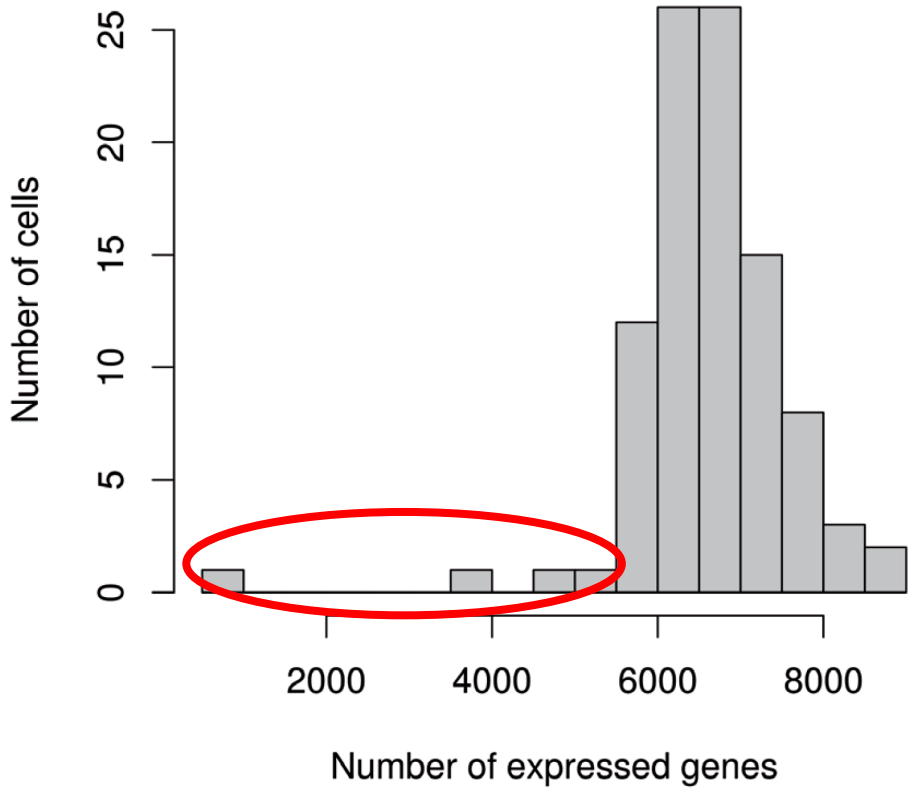
Select low quality cells



Quality control of cells

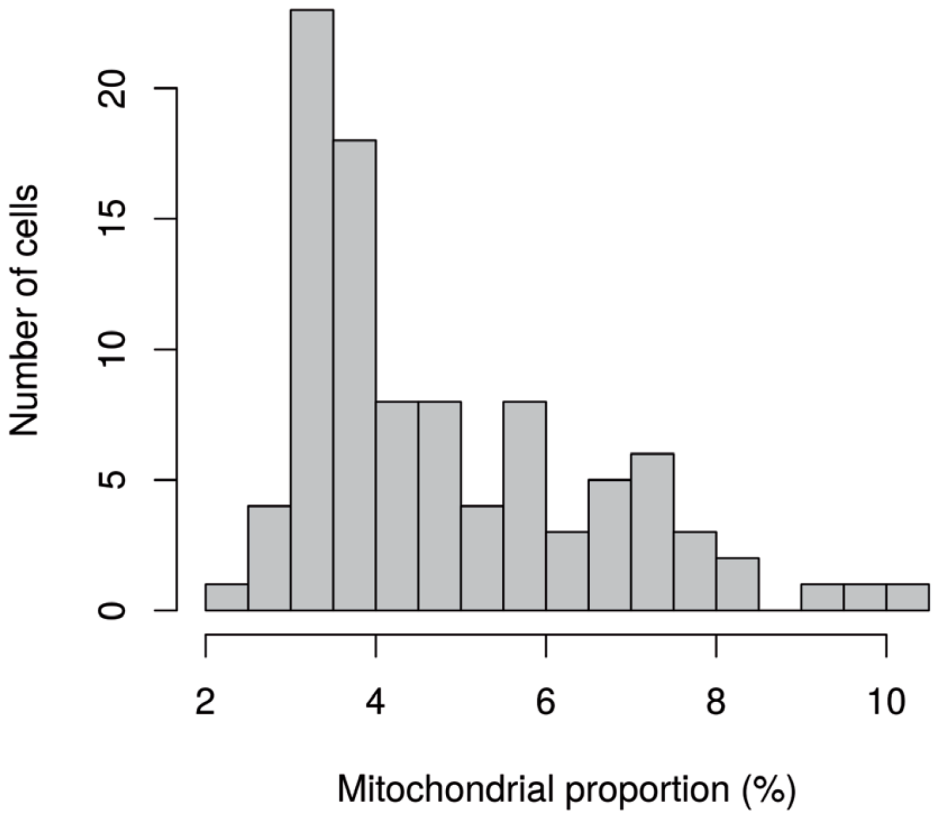


RNA has not been efficiently captured during library preparation

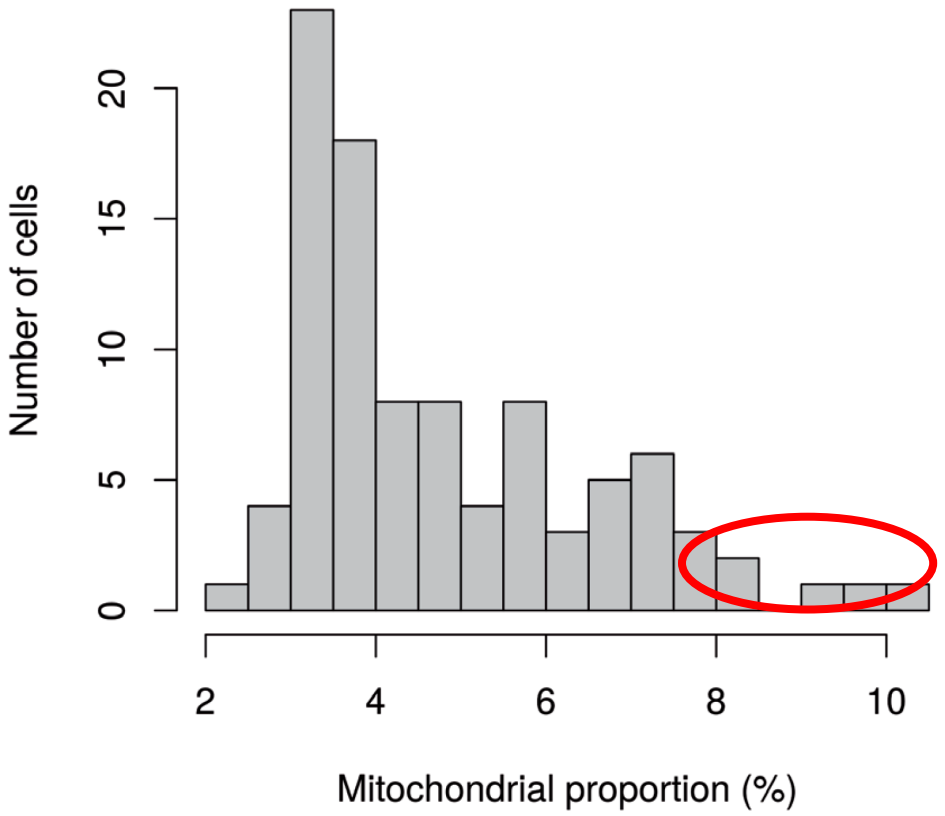


Diverse transcript population not captured

Select low quality cells



Quality control of cells (2)

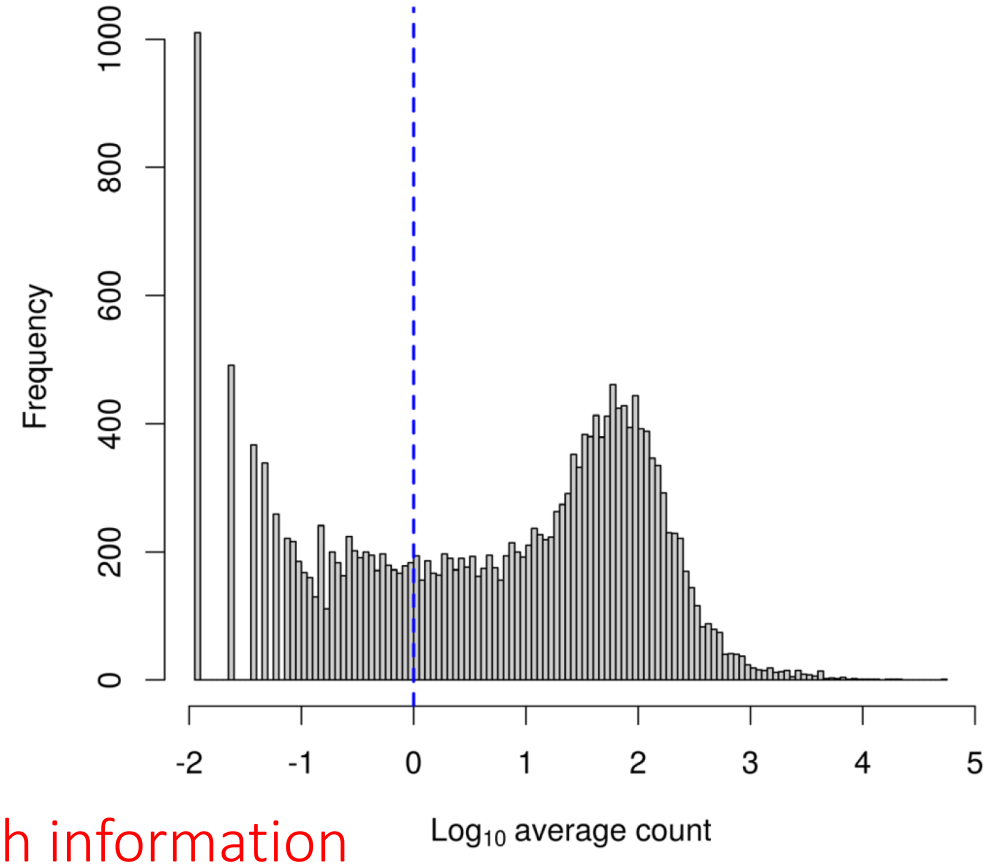


Possibly because of increased apoptosis
and/or loss of cytoplasmic RNA from lysed cells

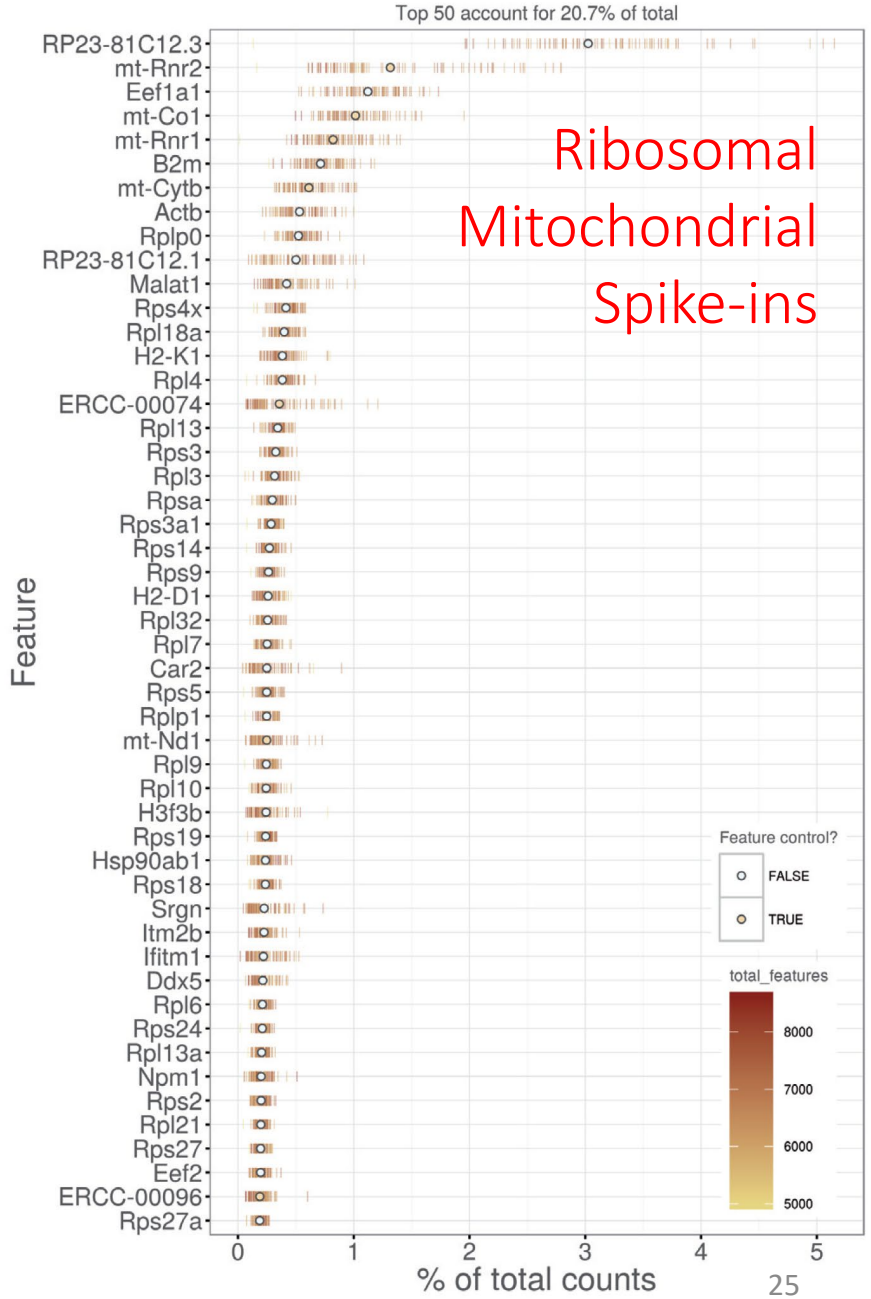
Deciding on cutoffs for filtering

- Do you have a homogeneous population of cells with similar sizes?
- Is it possible that you will remove cells from a smaller cell type?
- Examine PCA/tSNE/UMAP before and after filtering and make a judgment on whether to remove more or less cells.

Quality control of genes



Not enough information for reliable statistical inference



Ribosomal
Mitochondrial
Spike-ins

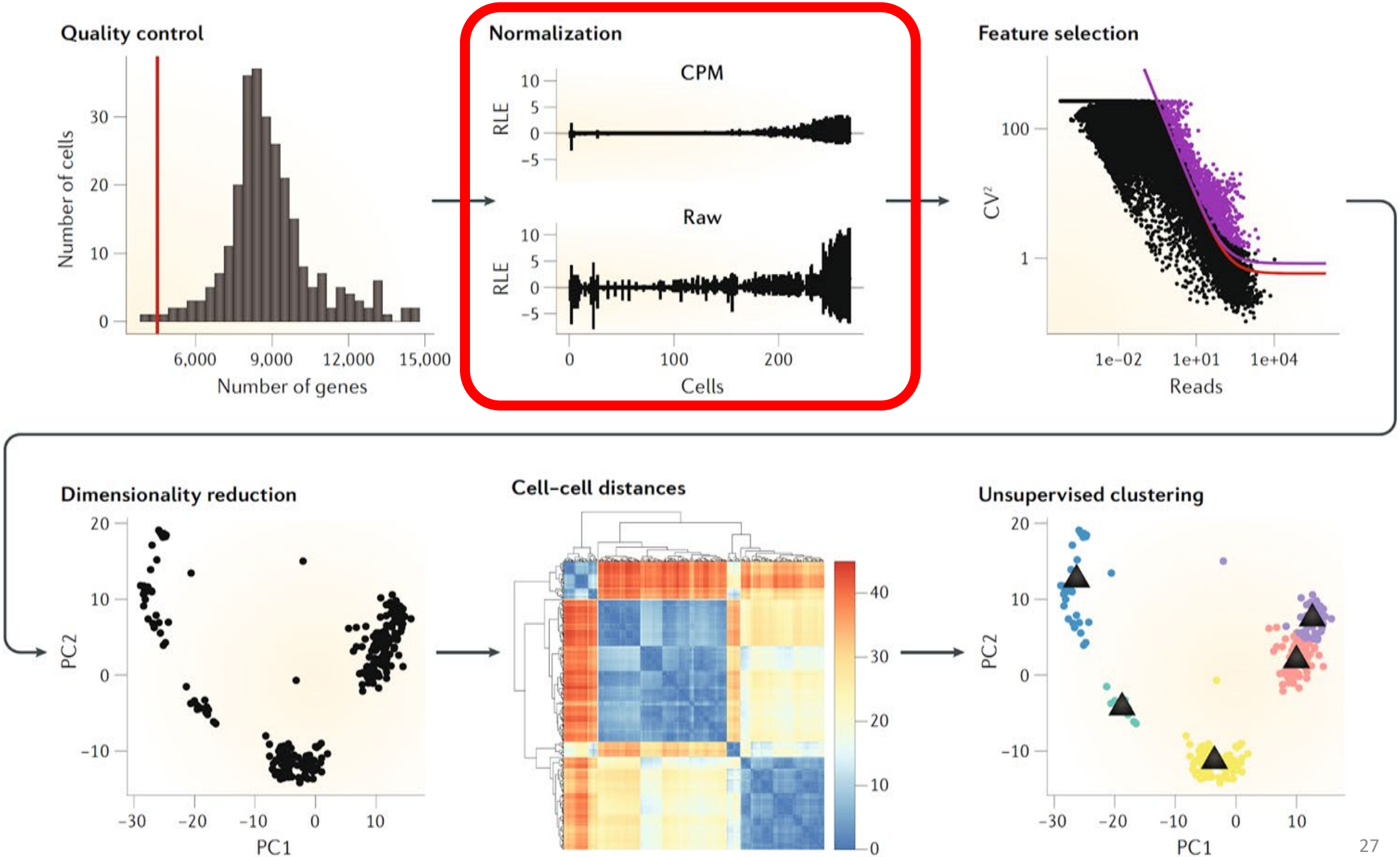
QC (pitfalls and recommendations)

- 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.

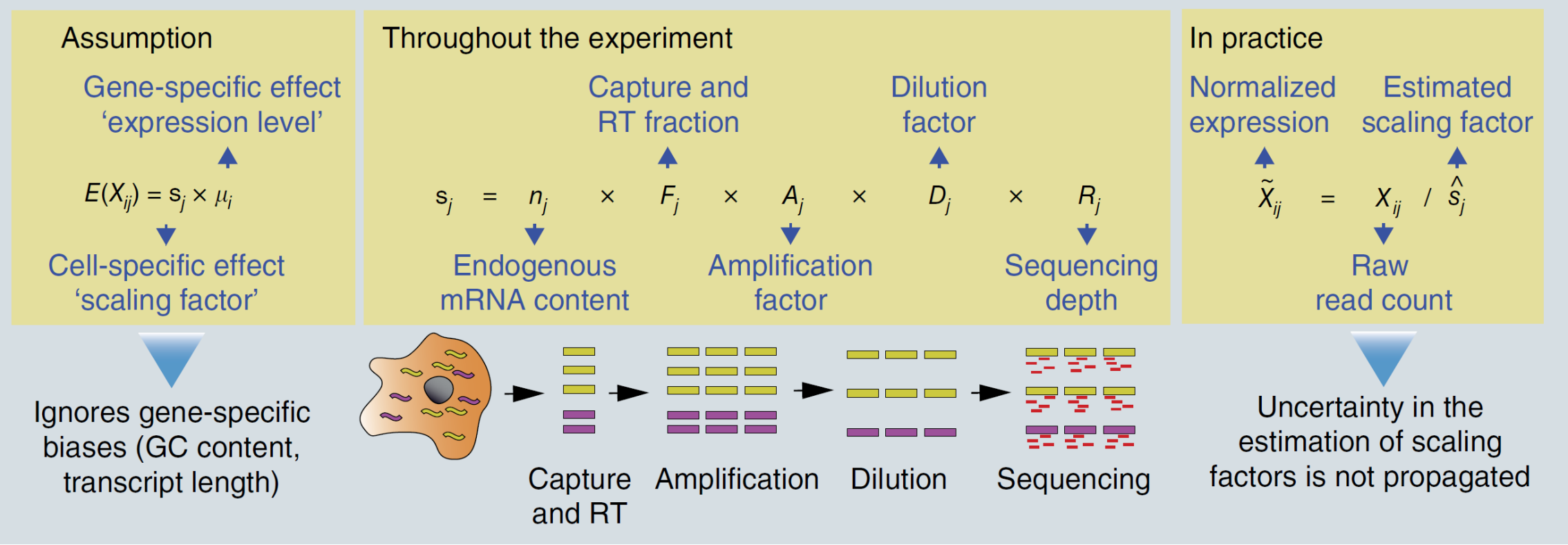
- If the distribution of QC covariates differ between samples, QC thresholds should be determined separately for each sample to account for sample quality differences as in Plasschaert et al (2018).

Luecken and Theis (MSB 2019)



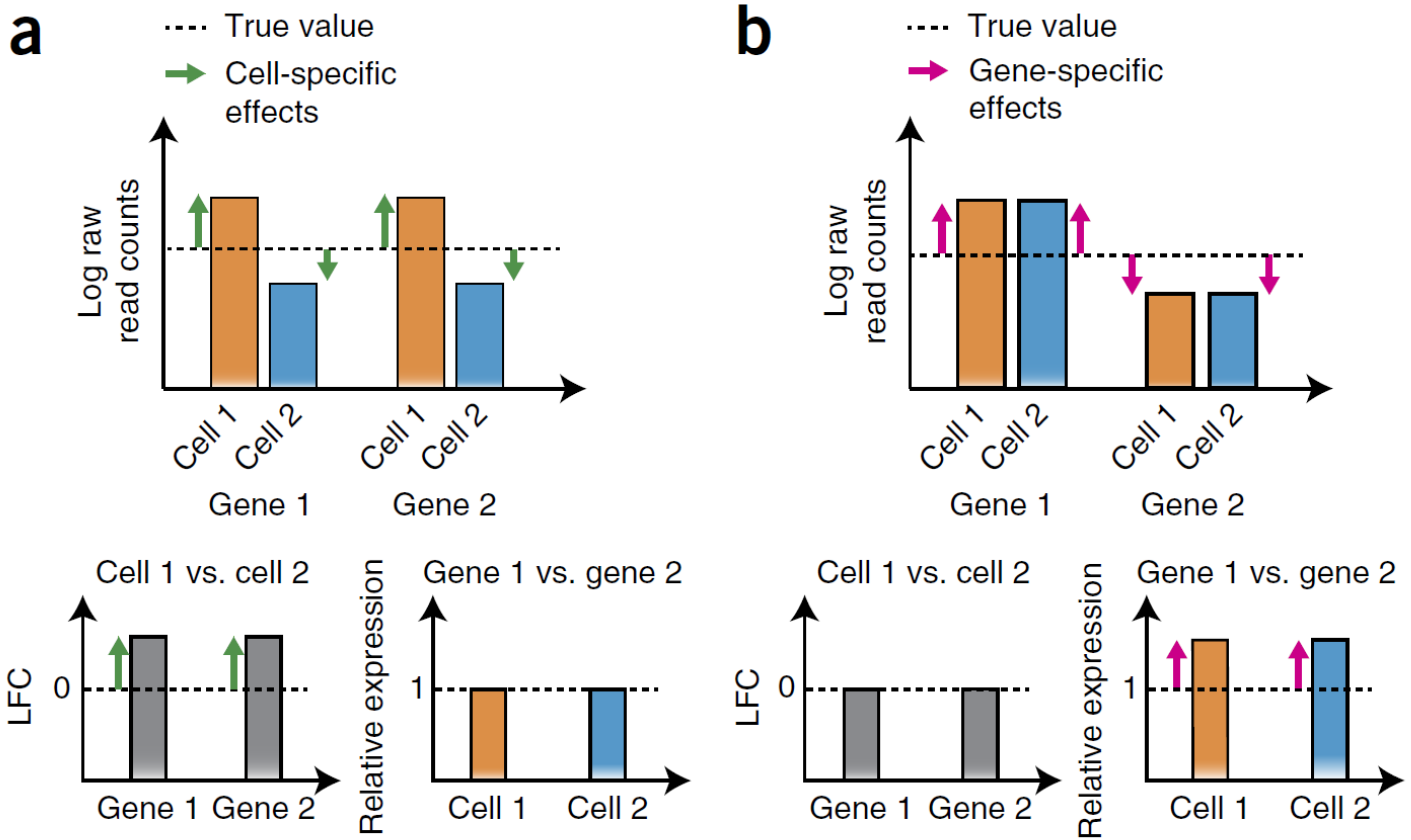
Kiselev et al. (Nature review Genetics 2019)

Normalization



Vallejos et al. (Nat Methods 2017)

Cell- and gene-specific effects in RNA-seq experiments



Vallejos et al. (Nat Methods 2017)

Which effects are removed by UMIs?

c

	Cell-specific effects	Gene-specific effects	Not removed by UMIs
Sequencing depth	✓		✓
Amplification	✓	✓	
Capture and RT efficiency	✓	✓	✓
Gene length		✓	
GC content	✓	✓	✓
mRNA content	✓		✓

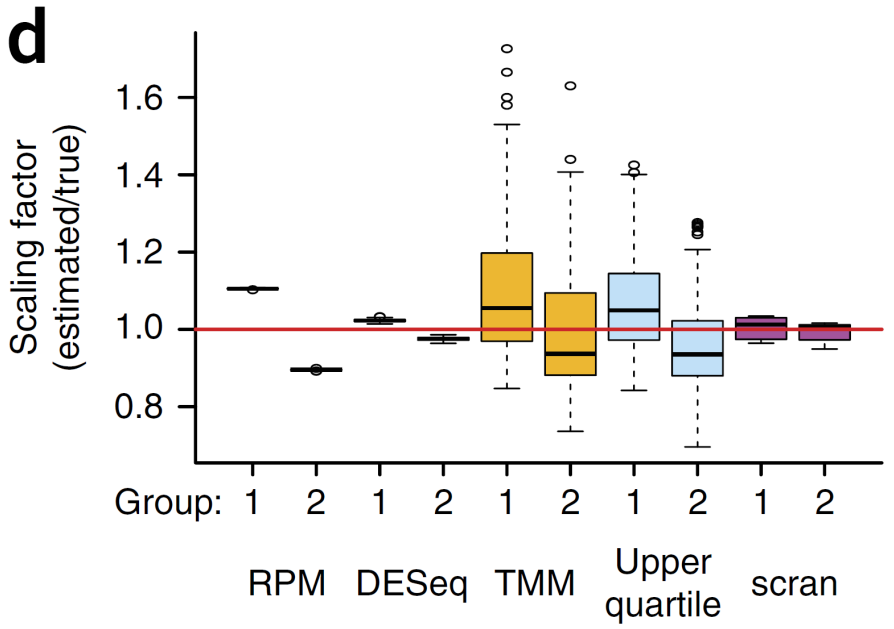
Normalization

- 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

Vallejos et al. (Nat Methods 2017)

What is different from bulk RNA-seq?

- Noise
  - Low mRNA content per cell
  - Variable mRNA capture
  - Variable sequencing depth
- Different cell types in the same sample
- Bulk RNA-seq normalization methods (FPKM, CPM, TPM, upperquartile) are based on per-gene statistics –> not suitable for zero-inflated data



Vallejos et al. (Nat Methods 2017)

Normalization methods

1. Size factor scaling methods
   - Log-normalization

2. Probabilistic methods
   - scTransform (Hafemeister & Satija Genome Biol 2019)
   - ZINB-WaVE (Risso et al. Nature Comm 2018)

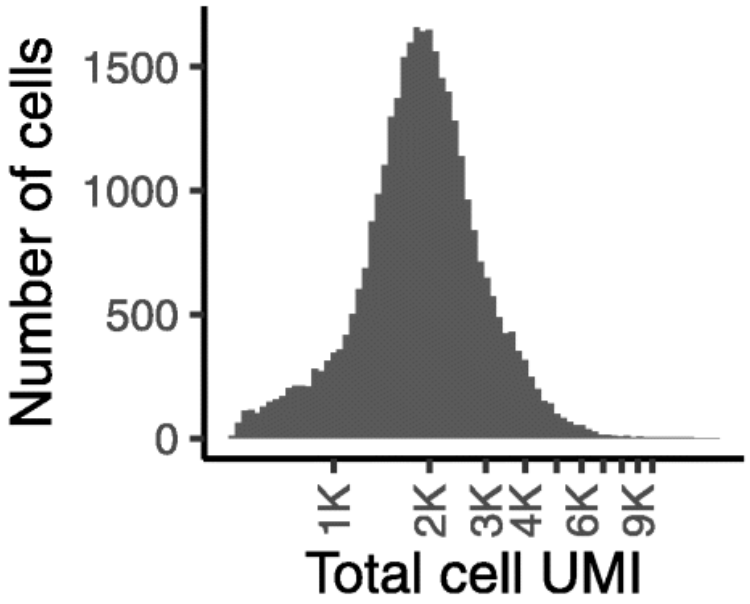
Log-normalization

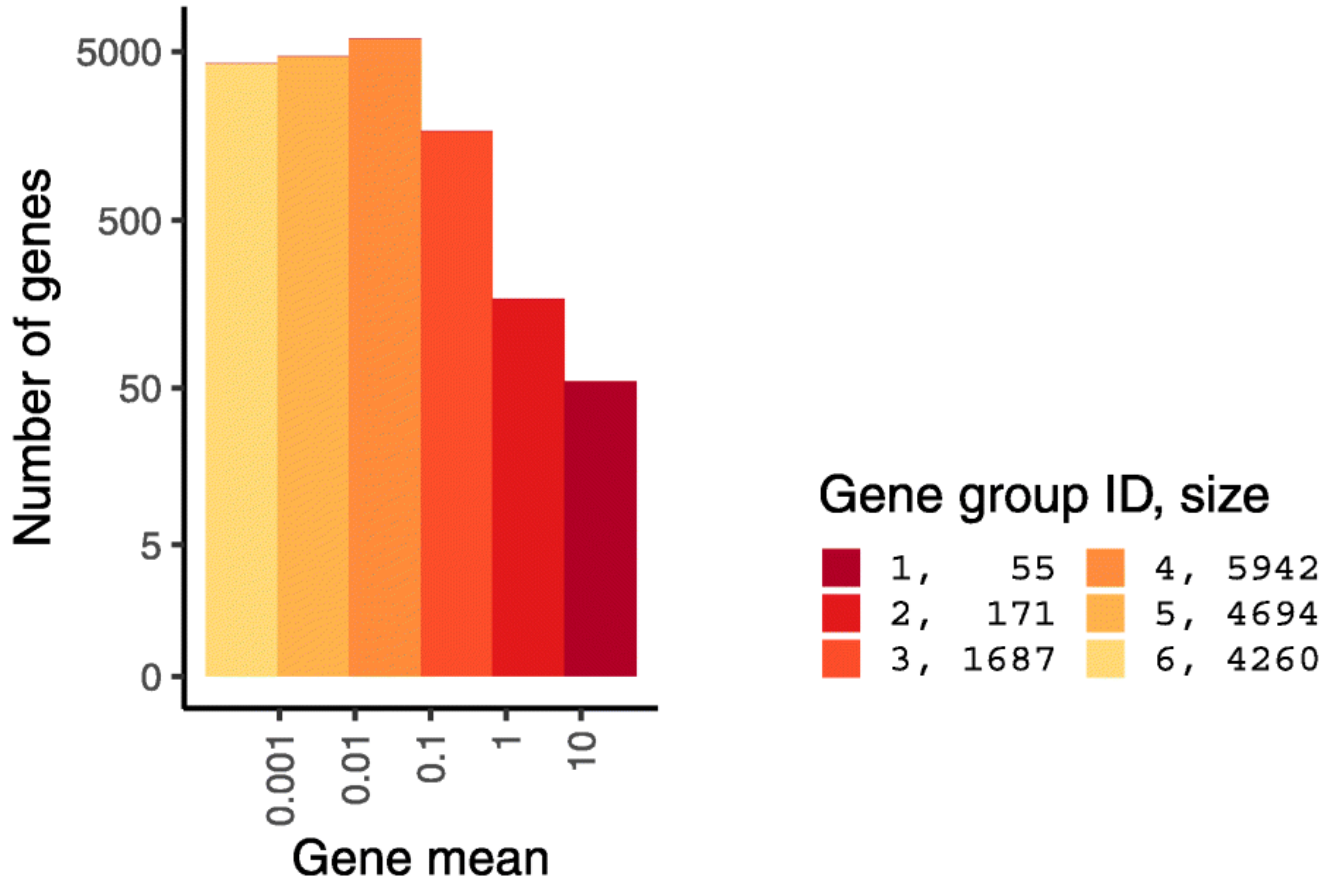
$$Y_{ij} = log_e((\frac{X_{ij}}{\sum_i X_{ij}} \times 10{,}000) + 1)$$

- Simplest and most commonly used normalization strategy
- Divide all counts for each cell by a cell-specific scaling factor (i.e. size factor)
- Assumes that any cell-specific bias (e.g., in capture or amplification efficiency) affects all genes equally via scaling of the expected mean count for that cell
- Modified CPM normalization
- Seurat, scanpy, 10X Cell Ranger: log-normalization

Does log-normalization (scaling) work?

33,148 PBMCs, 10x Genomics
16,809 genes detected ≥ 5 cells

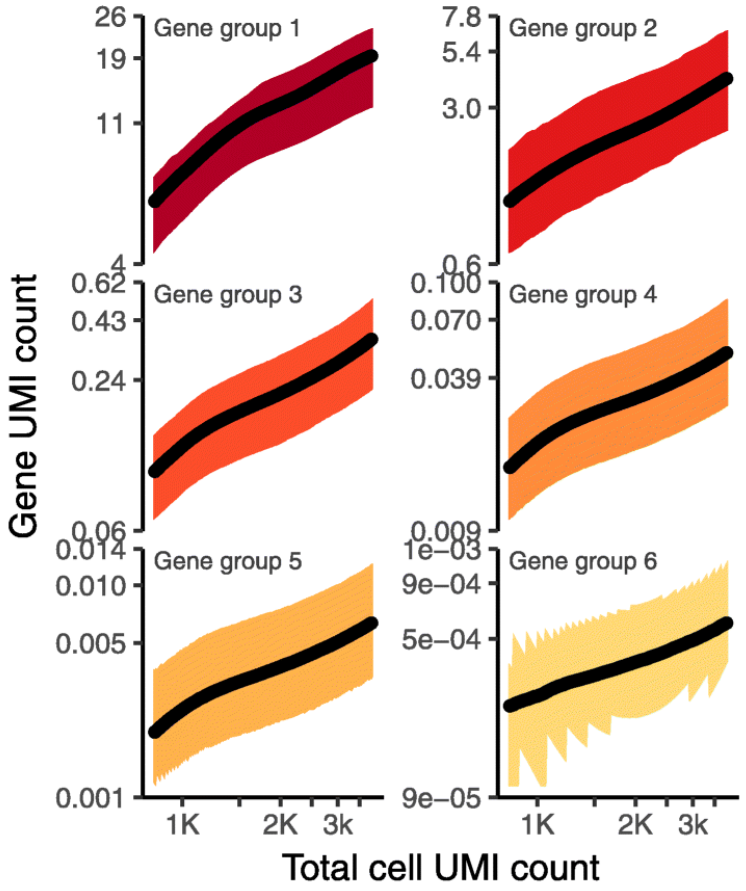




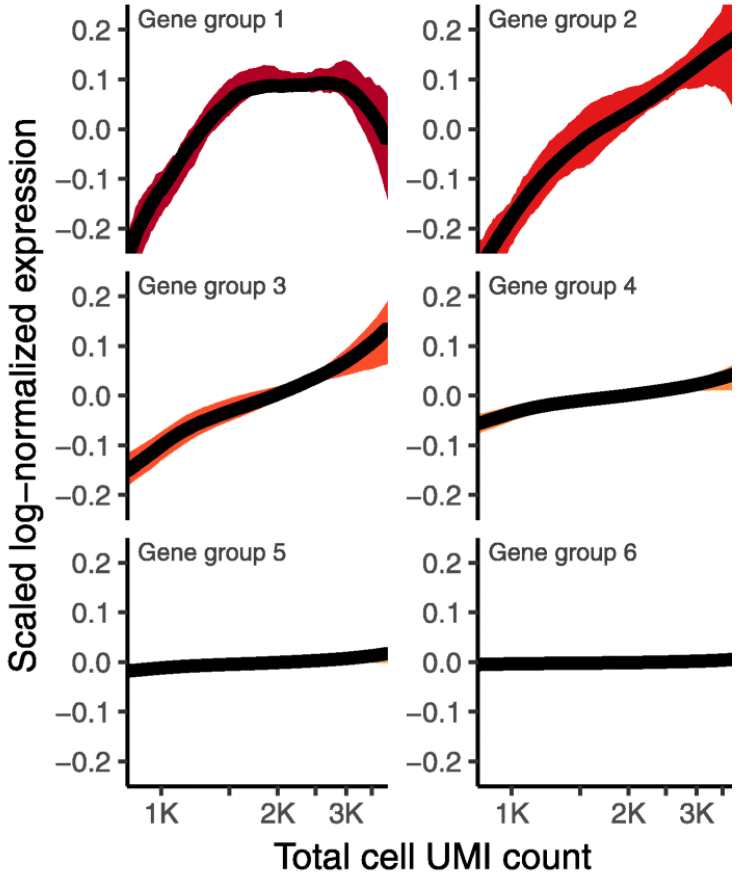
Hafemeister & Satija (Genome Biol 2019)

Does log-normalization (scaling) work?

Before normalization



After normalization



Hafemeister & Satija (Genome Biol 2019)

Modeling scRNAseq data

- Model the UMI counts for a given gene using a generalized linear model

$$\log\big(\mathbb{E}(x_i)\big) = \beta_0 + \beta_1 \log_{10} m + e_i$$

x_i: vector of UMI counts assigned to gene i

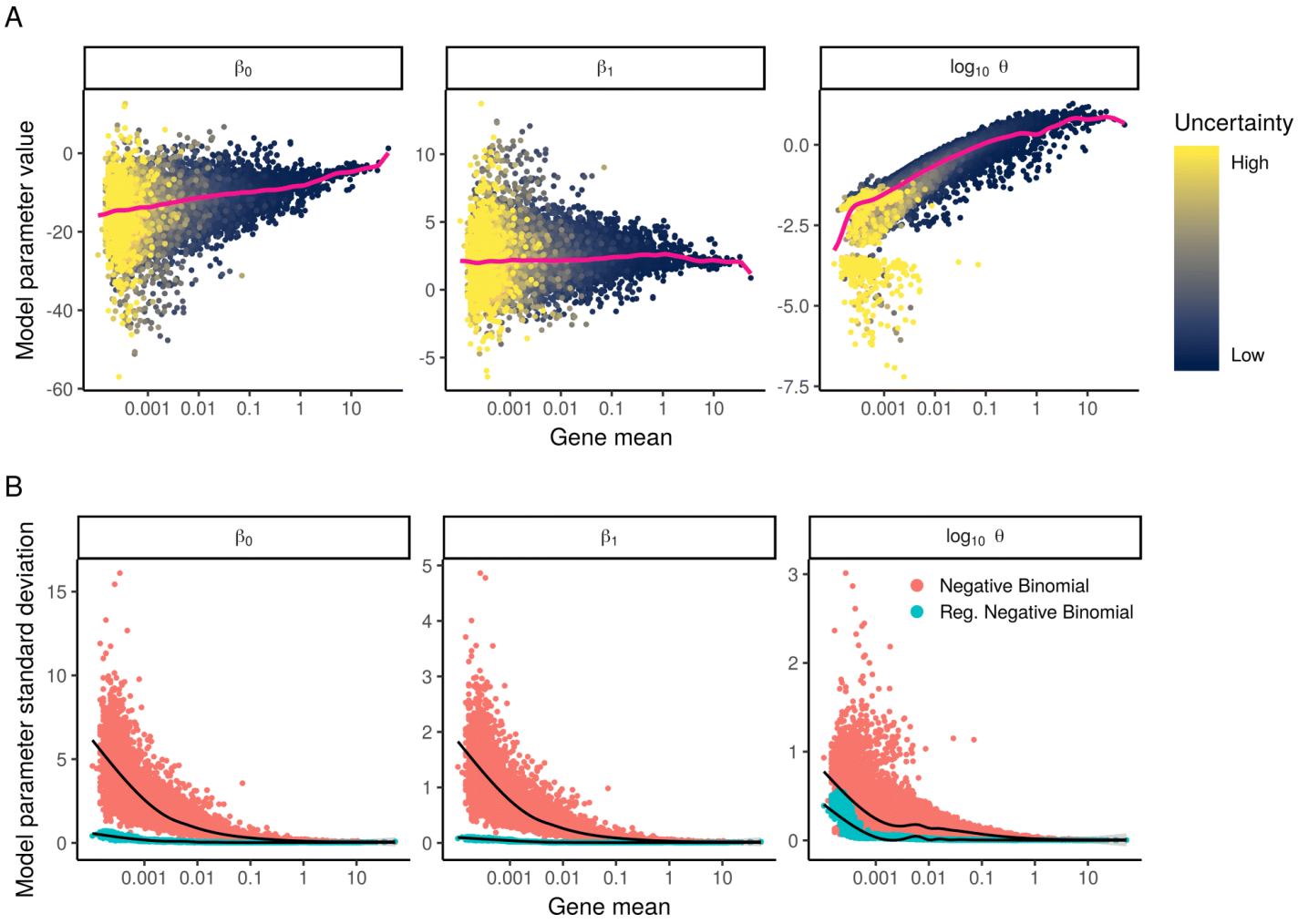
m: vector of molecules assigned to the cells, i.e., $m_j = \sum_i x_{ij}$

e_i: negative binomial (NB) error distribution, parameterized with mean μ and variance $\mu + \frac{\mu^2}{\sigma}$

Hafemeister & Satija (Genome Biol 2019)

Modeling scRNAseq data

- BUT, modeling each gene separately results in overfitting
- Solution: regularize all model parameters, including the NB dispersion parameter θ, by sharing information across genes



Hafemeister & Satija (Genome Biol 2019)

Modeling scRNAseq data

scTransform: Regularized negative binomial regression

Step1: fit independent regression models per gene

Step2: exploit the relationship of model parameter value
gene mean to learn global trends in the data (kernel regr
with a normal kernel)

Step3: use the regularized regression parameters to trans
UMI counts into Pearson residuals:

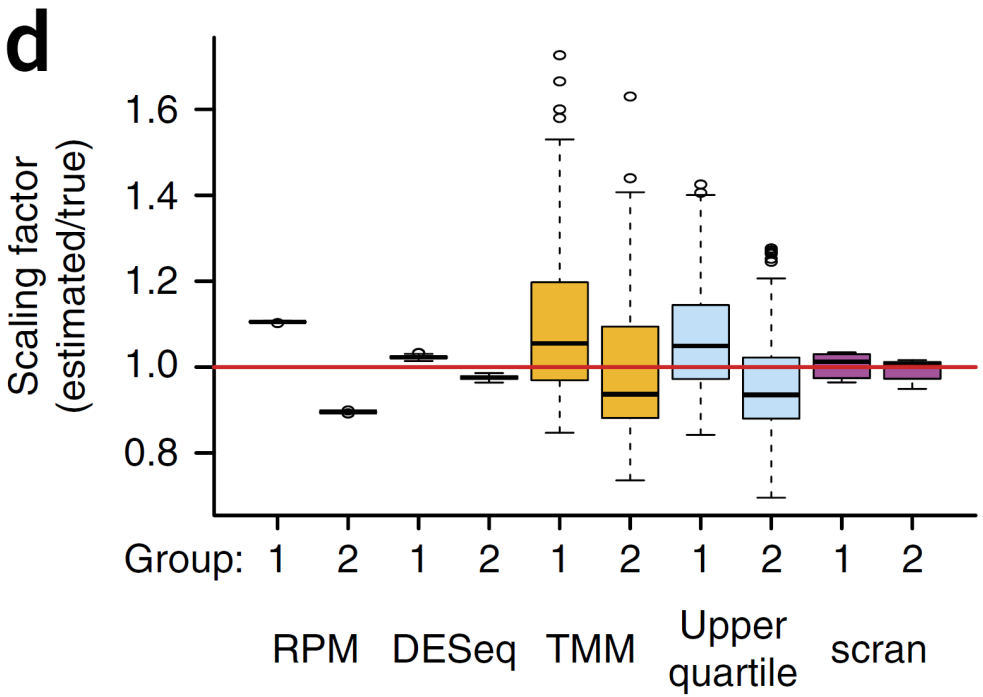
$$z_{ij} = \frac{x_{ij} - \mu_{ij}}{\sigma_{ij}},$$

$$\mu_{ij} = \exp\left(\beta_{0_i} + \beta_{1_i} \log_{10} m_j\right),$$

$$\sigma_{ij} = \sqrt{\mu_{ij} + \frac{\mu_{ij}^2}{\theta_i}},$$

Normalization (5)

- Bulk RNA-based methods: FPKM, CPM, TPM, upperquartile *(NOT APPROPRIATE)*
- Log normalization (Seurat)
- Negative binomial (Monocole)
- Zero-inflated negative binomial (ZINB) models
- scTransform (regularized NB regression)
- …



Performance Assessment and Selection of Normalization Procedures for Single-Cell RNA-Seq
Cole et al, Cell Systems 2019

Vallejos et al. (Nat Methods 2017)

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.

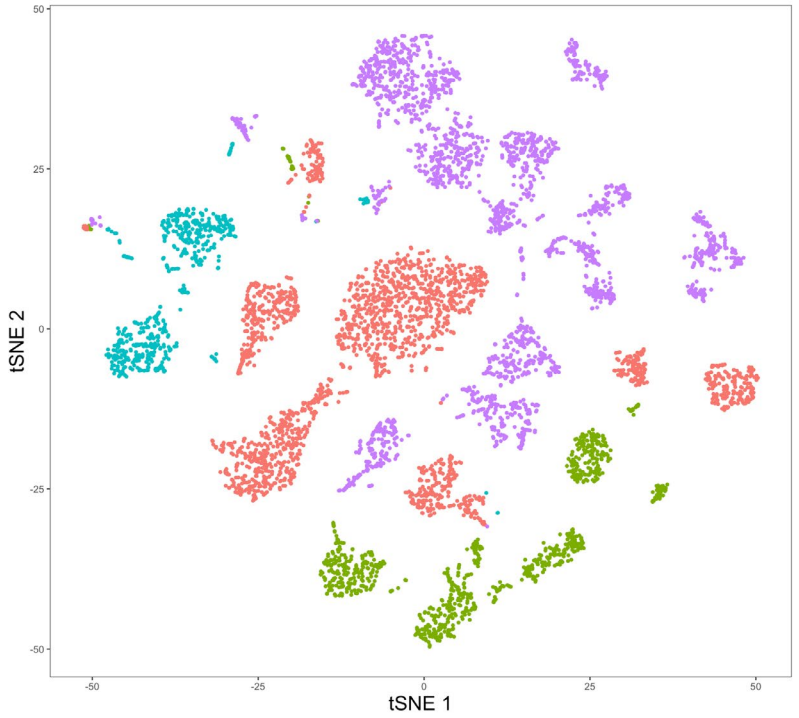
- There is no consensus on scaling genes to 0 mean and unit variance. We prefer not to scale gene expression.

- Normalized data should be log(x+1)-transformed for use with downstream analysis methods that assume data are normally distributed.

Luecken and Theis (MSB 2019)

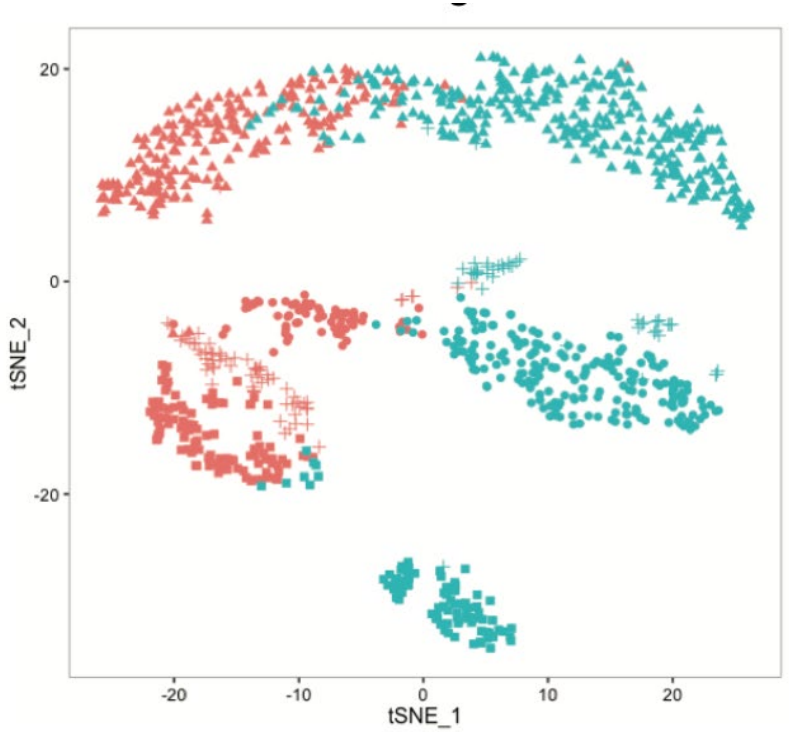
Batch correction

Why integrate?



Donor 1
Donor 2
Donor 3
Donor 4

Same tissue from different donors



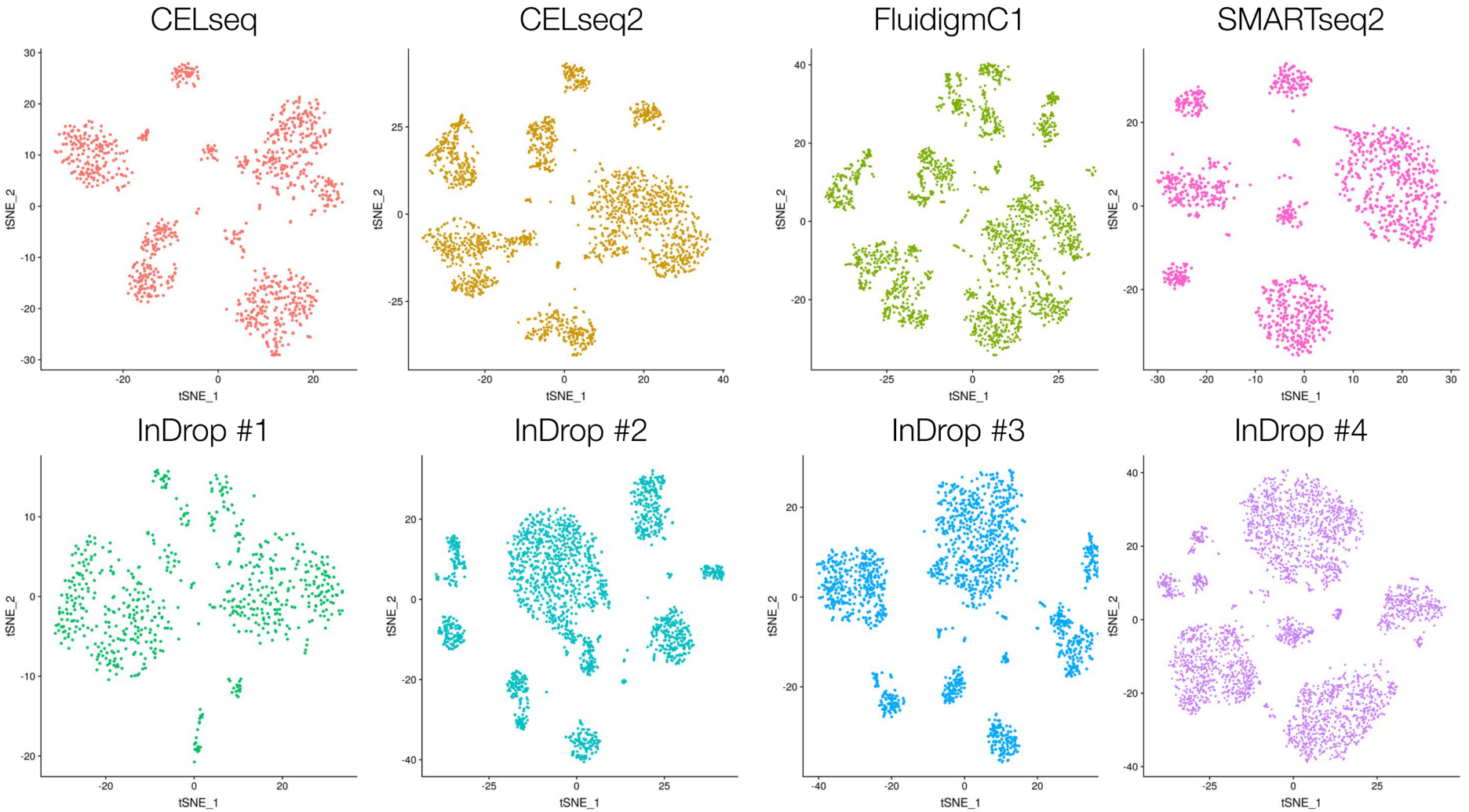
Sick
Healthy

▲ Cell Type 1
■ Cell Type 2
● Cell Type 3

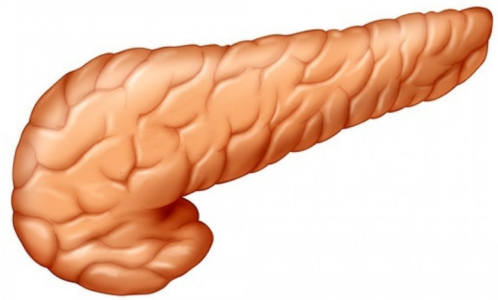
Cross condition comparisons

Building a cell atlas

8 maps of the human pancreas



Human pancreas



Baron et al. 2016, *Cell Syst.*

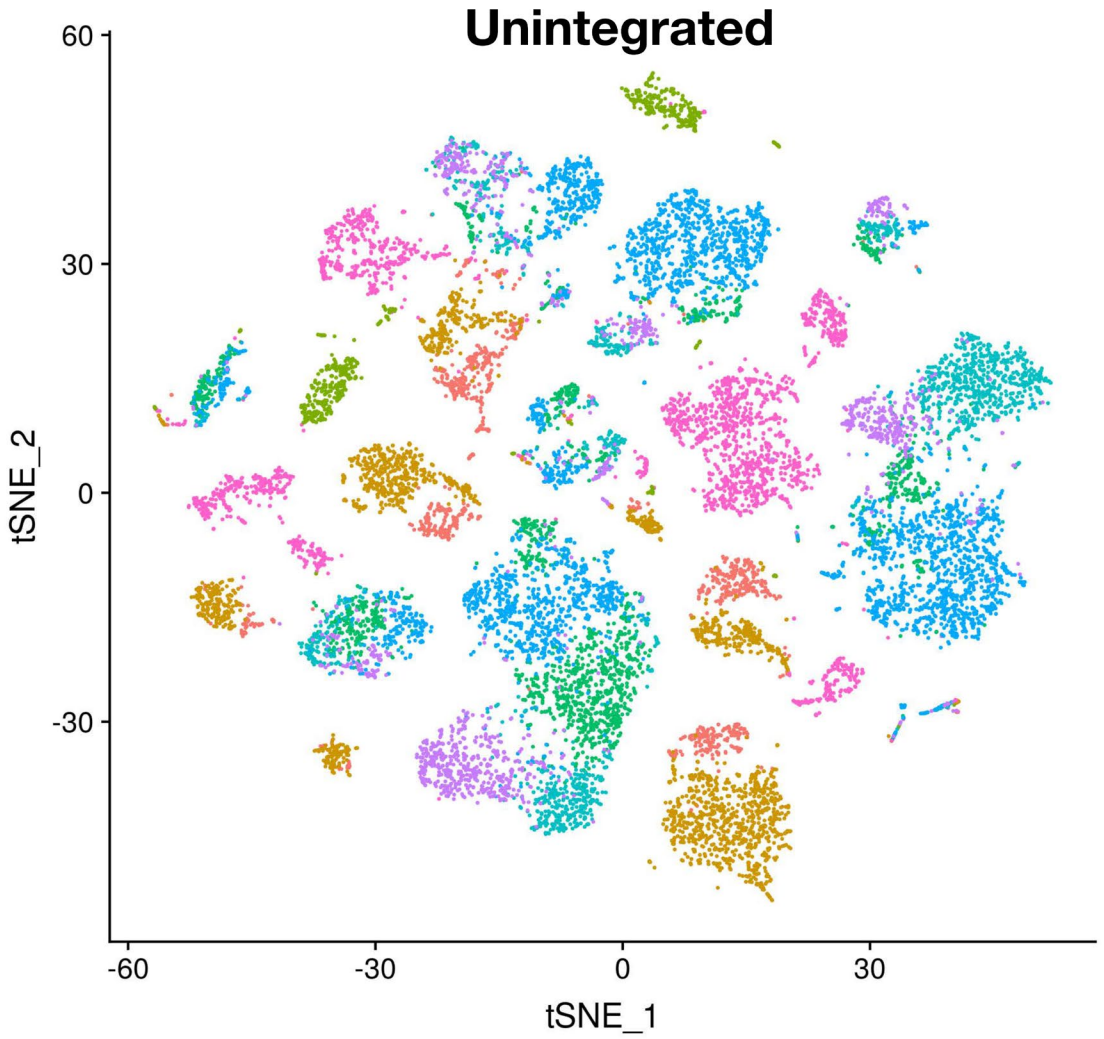
Lawlor et al. 2017, *Genome Res.*

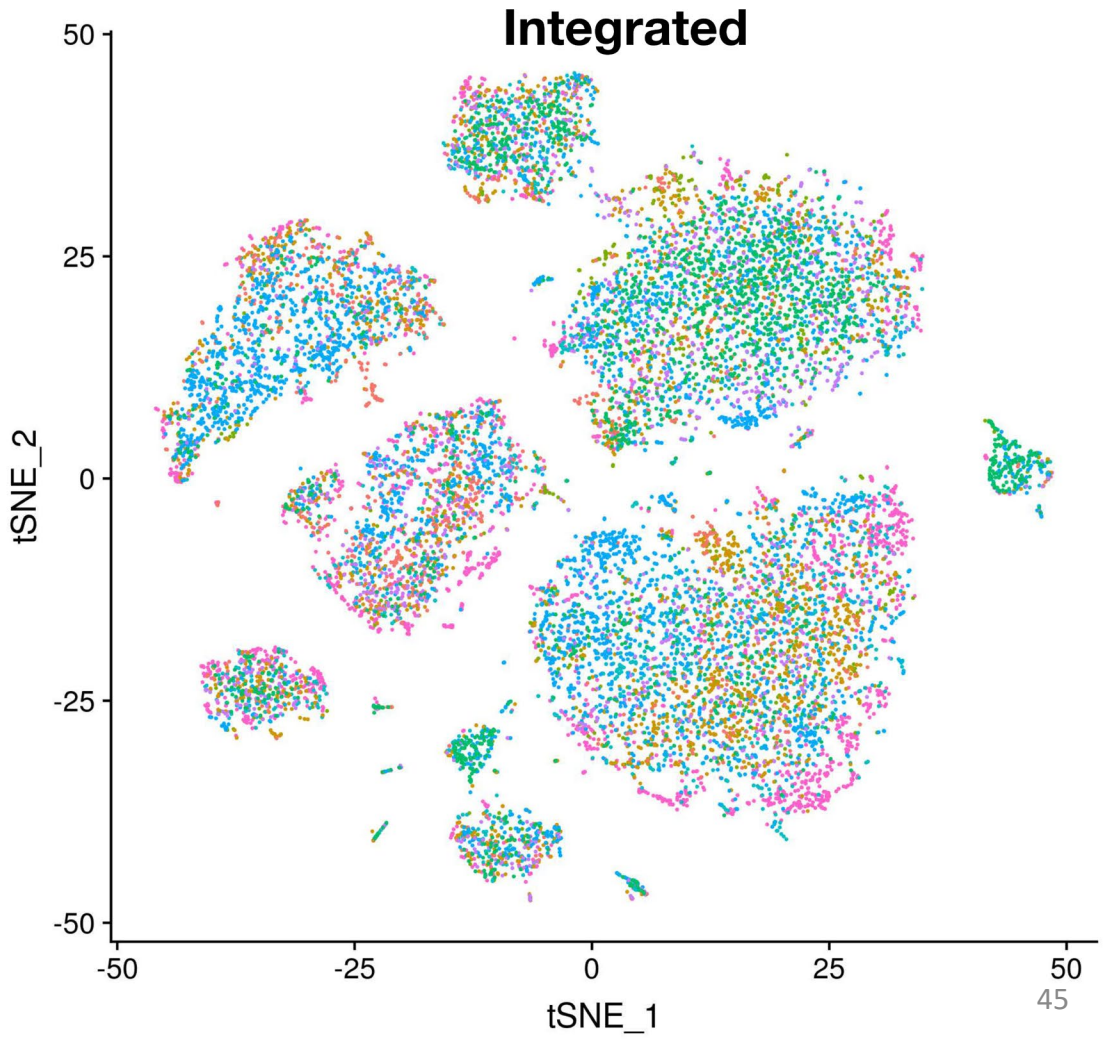
Grun et al. 2016, *Cell Stem Cell*

Muraro et al. 2016, *Cell Syst.*

Building a cell atlas

8 maps of the human pancreas

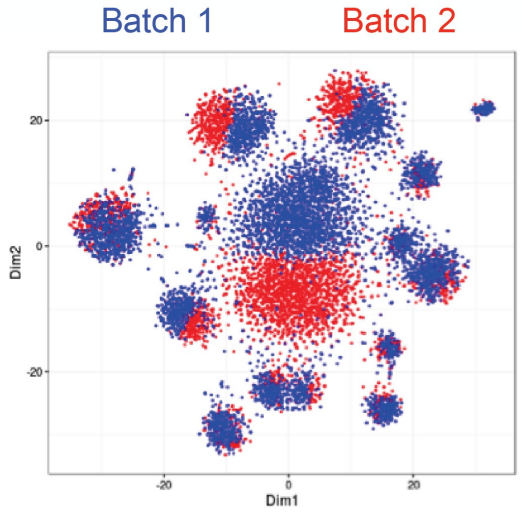




Confounders and batch effects

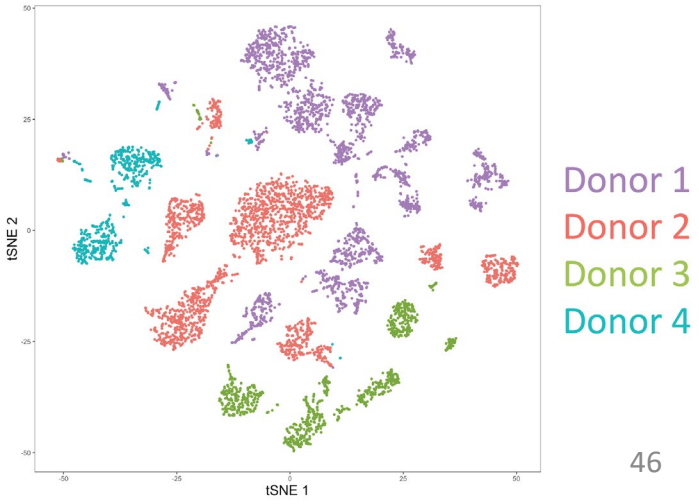
1. Technical variability
   - Changes in sample quality/processing
   - Library prep or sequencing technology
   - ‘Experimental reality’

Technical ‘batch effects’ confound downstream analysis



2. Biological variability
   - Patient differences
   - Environmental/genetic perturbation
   - Evolution! (cross-species analysis)

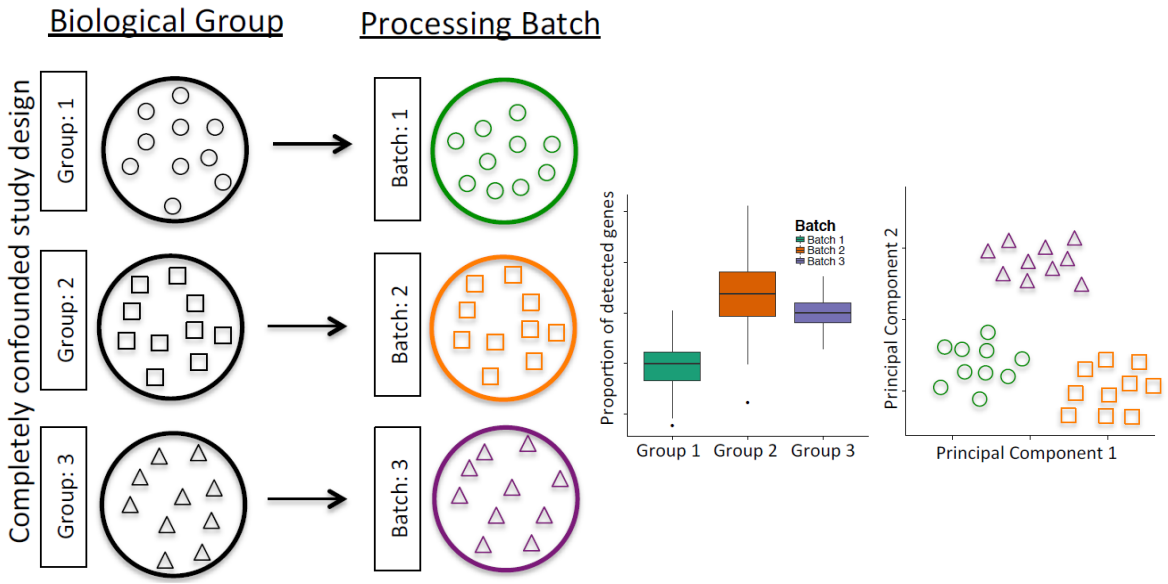
Biological ‘batch effects’ confound comparisons of scRNA-seq data



Shaham et al. (Bioinformatics 2017)

Confounders and batch effects

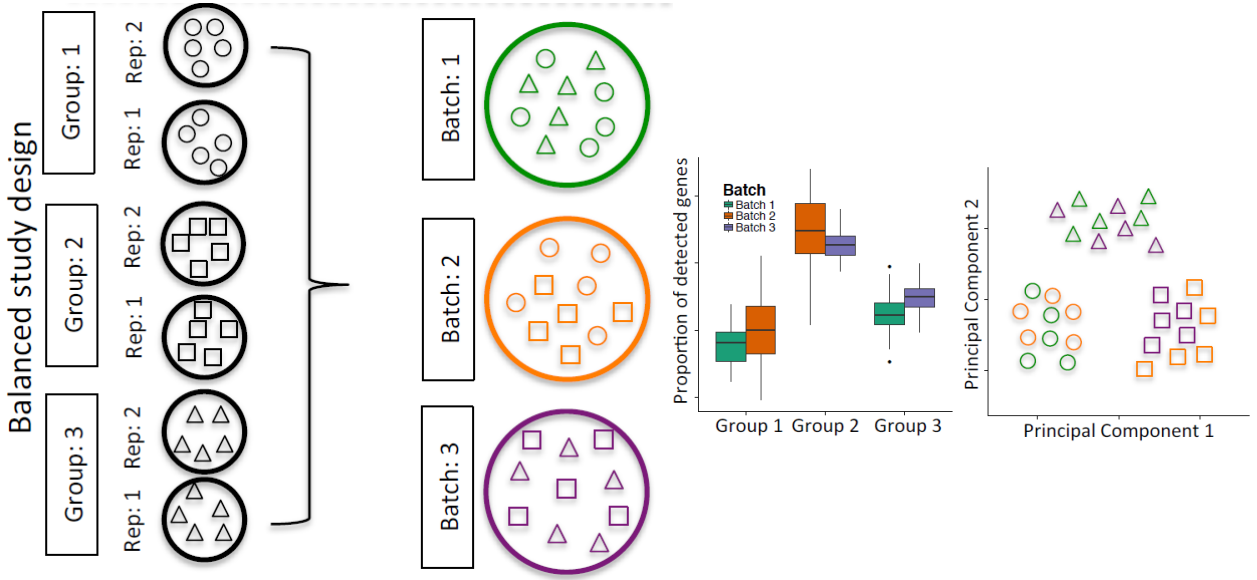
Confounded design



Don’t design your experiment like this!!!

Hicks et al. (bioRxiv 2015)

Not confounded design



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

Normalization vs Batch correction

- **Normalization:** occurs regardless of the batch structure and only considers technical biases.

- **Batch correction:** only occurs across batches and must consider both technical biases and biological differences.

- *Technical biases:* tend to affect genes in a similar manner, or at least in a manner related to their biophysical properties (e.g., length, GC content).

- *Biological differences:* highly unpredictable.

Batch correction methods

- Many good options have been developed for bulk RNA-seq data:
  - RUVseq() or svaseq()
  - Linear models with e.g. removeBatchEffect() in limma or scater
  - ComBat() in sva
  - …

- But bulk RNA-seq methods make modelling assumptions that are likely to be violated in scRNAseq data
  - The composition of cell populations are either known or the same across batches
  - Batch effect is additive: batch-induced fold-change in expression is the same across different cell subpopulations for any given gene

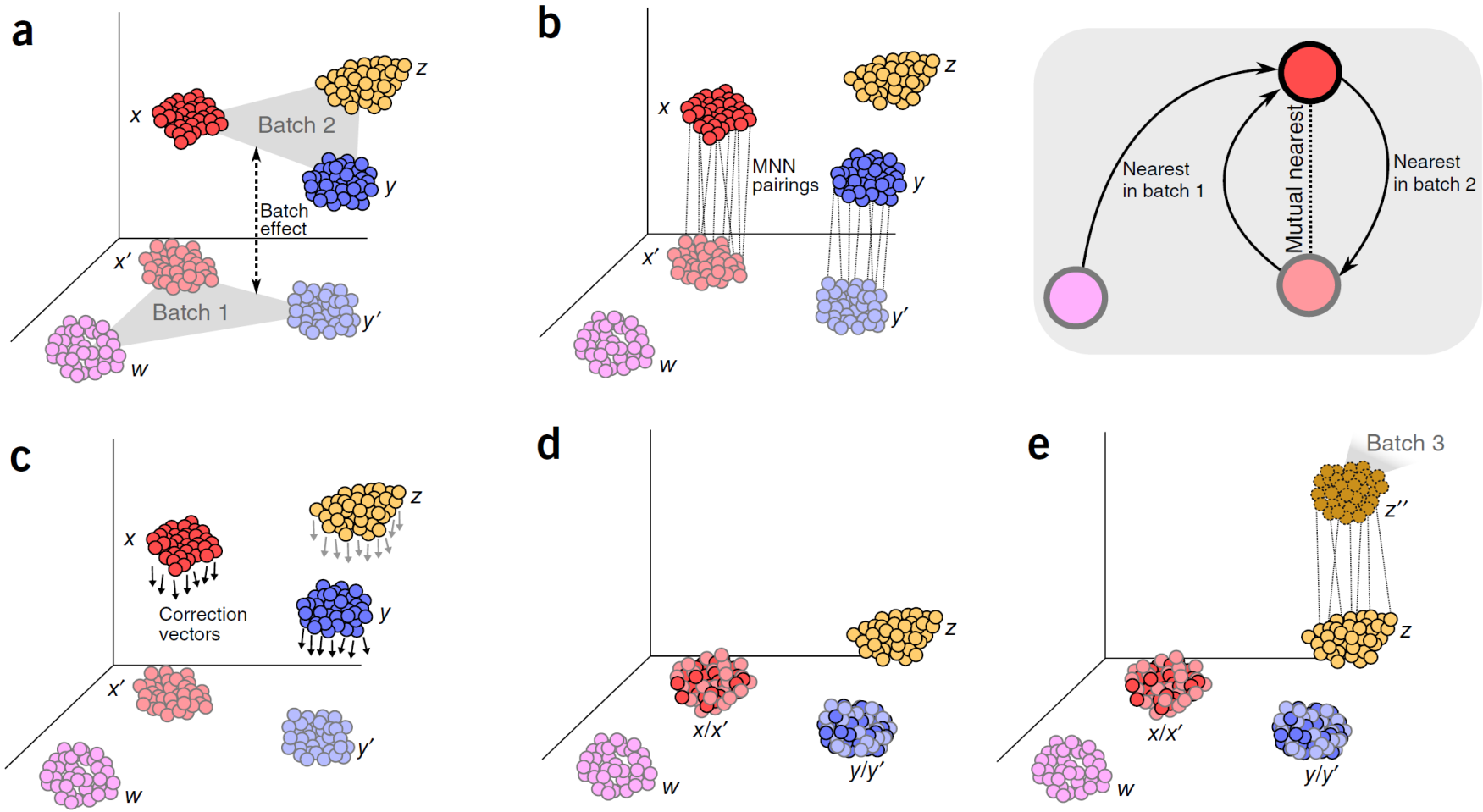
Batch correction methods

- 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

Mutual Nearest Neighbors (MNN)

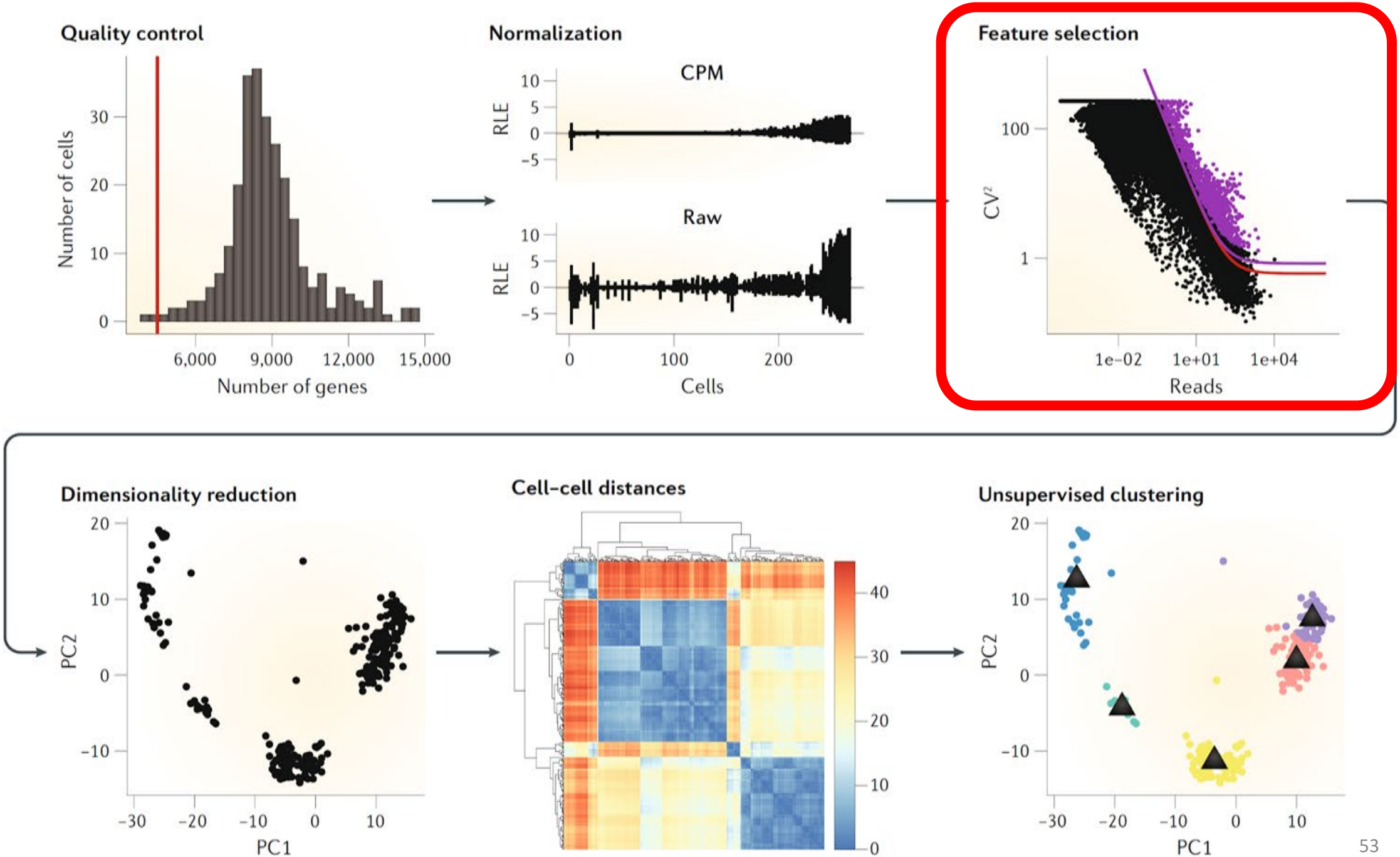


Haghverdi et al., (Nature Biotech 2017)

Using the corrected values

- Batch correction facilitates cell-based analysis of population heterogeneity in a consistent manner across batches.
  - No need to identify mappings between separate clusterings
  - Increased number of cells allows for greater resolution of population structure

- BUT…
- It is not recommended to use the corrected expression values for gene-based analyses (e.g. differential expression)
- Arbitrary correction algorithms are not obliged to preserve the magnitude (or even direction) of differences in per-gene expression when attempting to align multiple batches



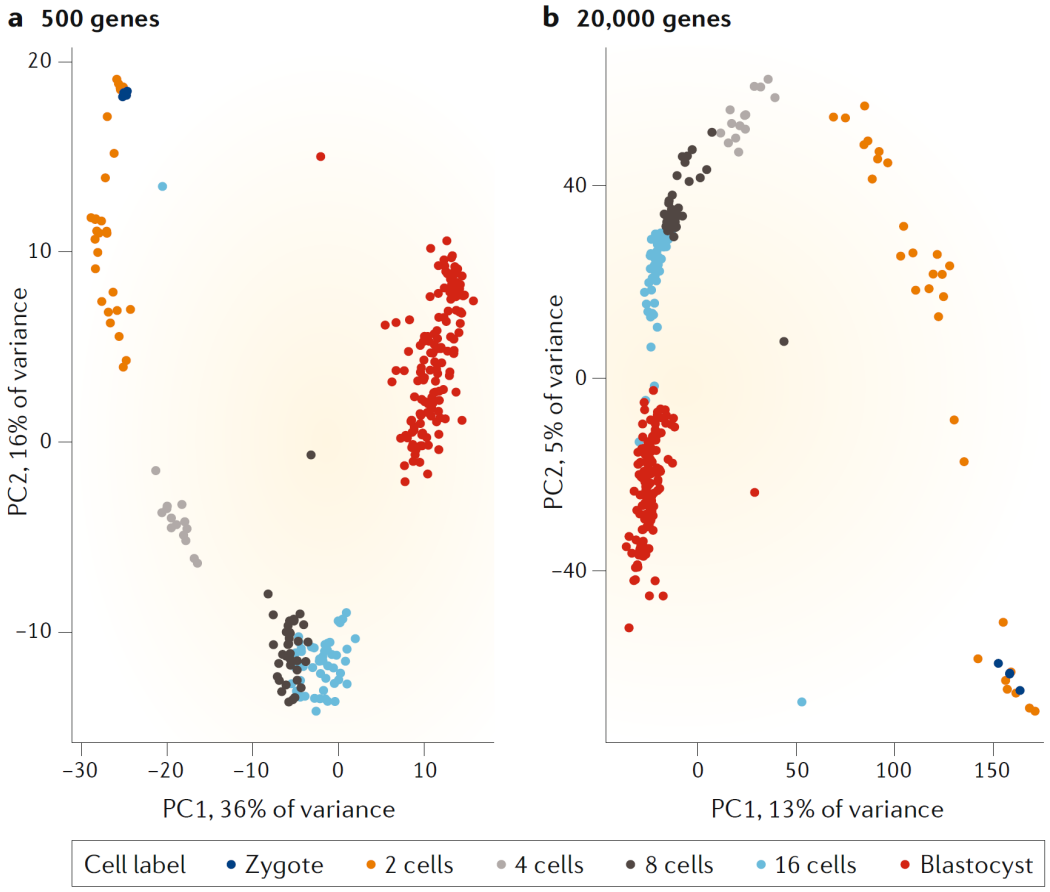
Kiselev et al. (Nature review Genetics 2019)

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



Kiselev et al. (https://doi.org/10.1038/s41576-018-0088-9)

Feature selection

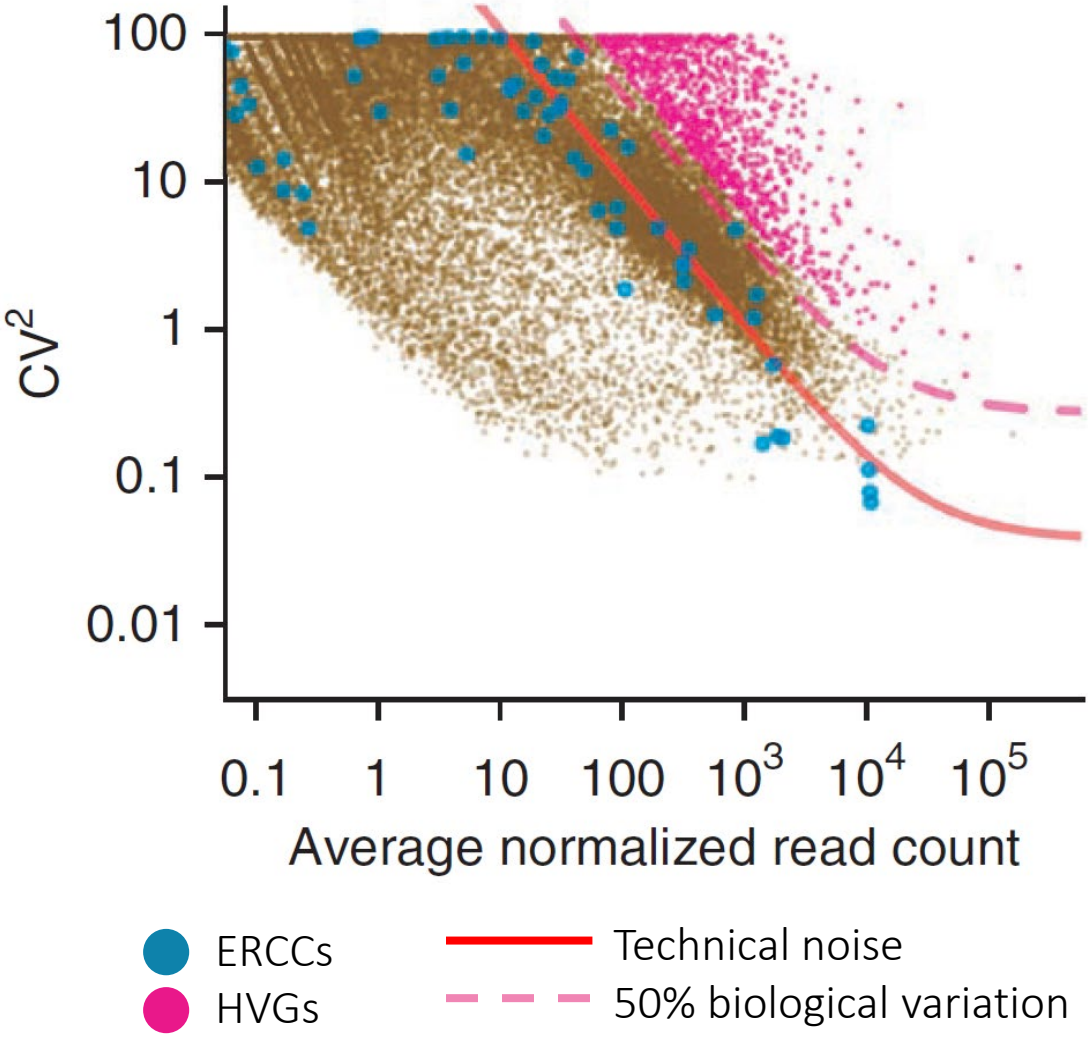
Highly Variable Genes (HVG)

- $CV = \frac{var}{mean} = \frac{\sigma}{\mu}$

- Fit a gamma generalized linear model

- No ERCCs?

  -> estimate technical noise based on all genes



Brennecke et al. (https://doi.org/10.1038/nmeth.2645)

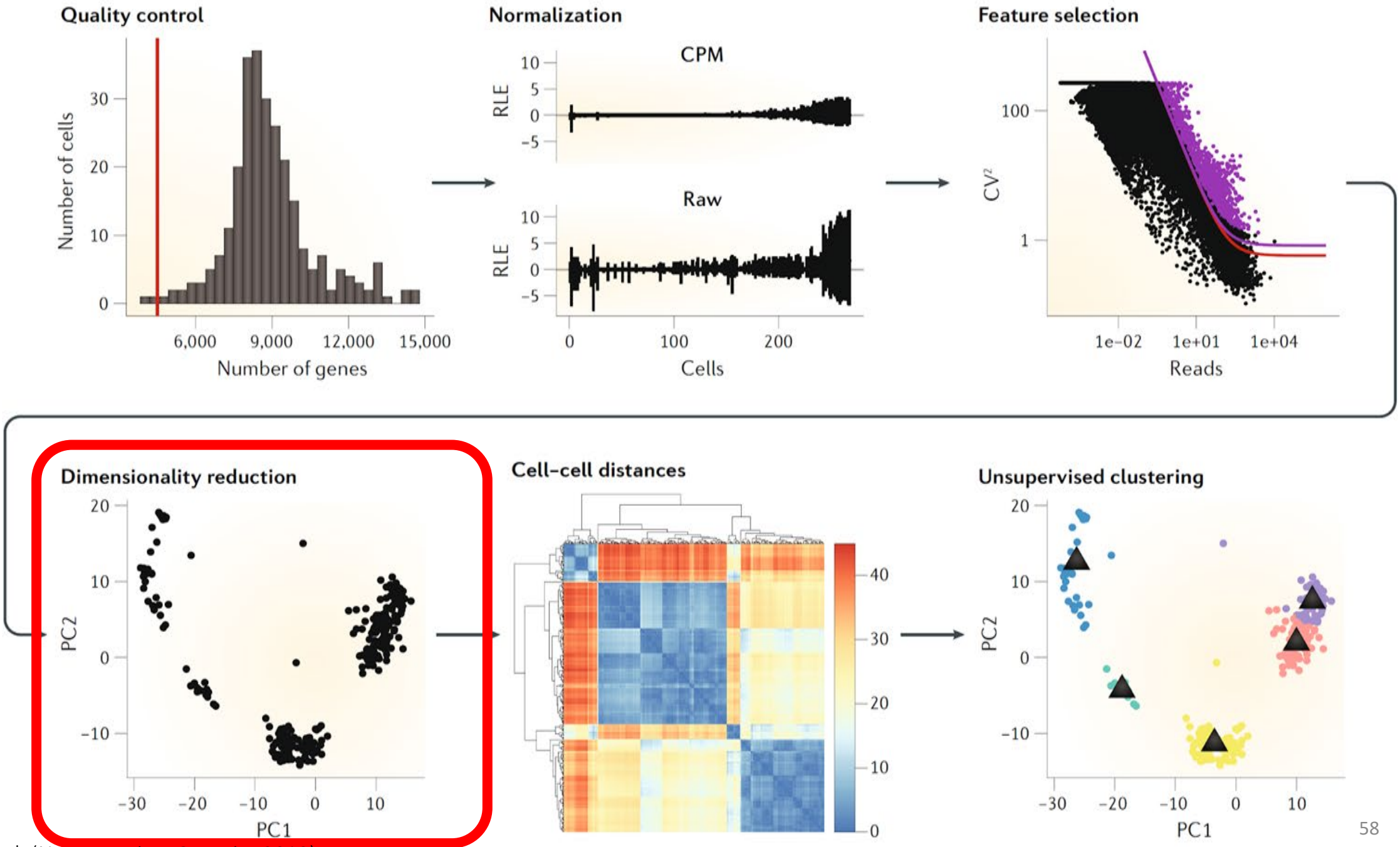
Feature Selection (pitfalls and recommendations)

- 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.

Luecken and Theis (MSB 2019)

End part 1

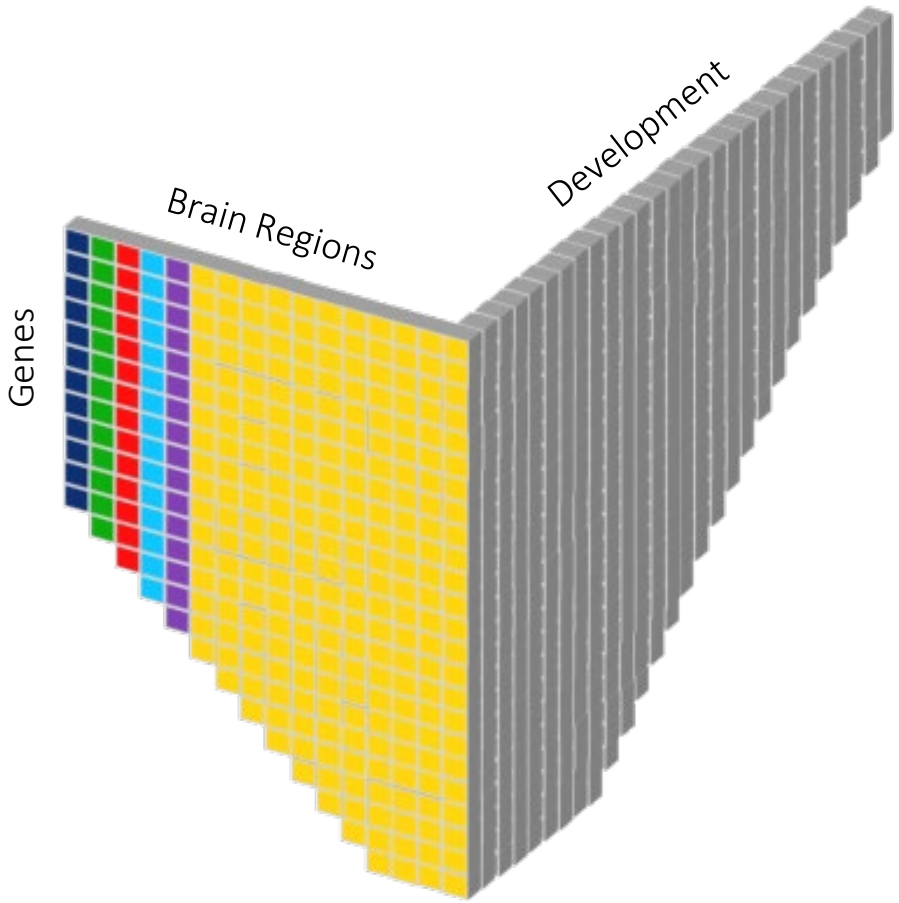


Kiselev et al. (Nature review Genetics 2019)

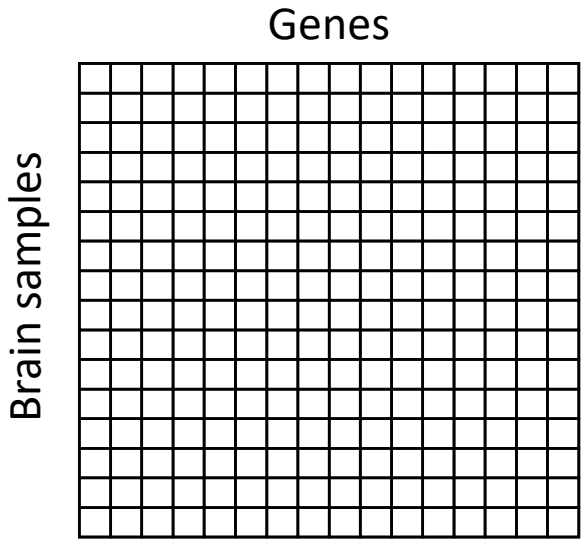
High dimensional data

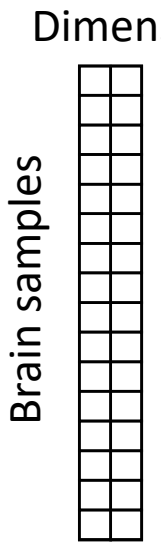
- We have huge amounts of complex data (many samples x many genes)

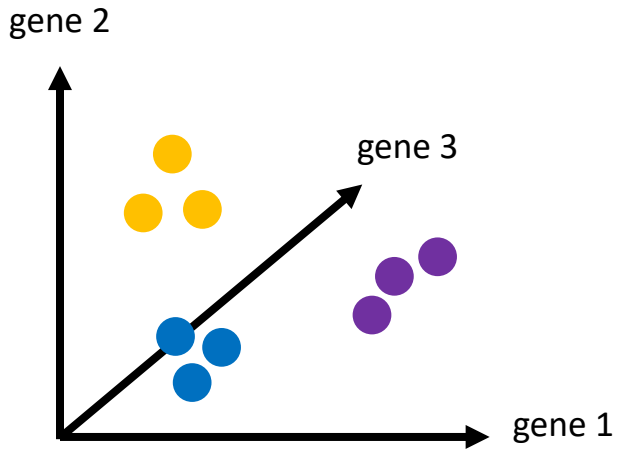
- We want to reduce complexity for analysis

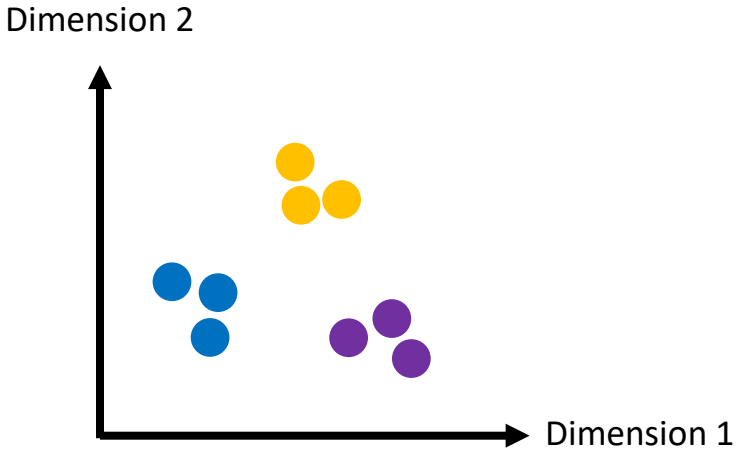


Dimensionality reduction









Why Dimensionality reduction?

- Simplify complexity, so it becomes easier to work with
  - “Remove” redundancies in the data
  - Identify the most relevant information (find and filter noise)
  - Reduce computational time for downstream procedures e.g. clustering

- Visualization

Dimensionality reduction

Matrix factorization

Graph-based

Auto-encoders

PCA	linear		
ICA	linear		
MDS	non-linear		
Sparce NNMF	non-linear	2010	https://pdfs.semanticscholar.org/664d/40258f12ad28ed0b7d4 c272935ad72a150db.pdf
cPCA	non-linear	2018	https://doi.org/10.1038/s41467-018-04608-8
ZIFA	non-linear	2015	https://doi.org/10.1186/s13059-015-0805-z
ZINB-WaVE	non-linear	2018	https://doi.org/10.1038/s41467-017-02554-5
Diffusion maps	non-linear	2005	https://doi.org/10.1073/pnas.0500334102
Isomap	non-linear	2000	10.1126/science.290.5500.2319
t-SNE	non-linear	2008	https://lvdmaaten.github.io/publications/papers/JMLR_2008.pdf
- BH t-SNE	non-linear	2014	https://lvdmaaten.github.io/publications/papers/JMLR_2014.pdf
- Flt-SNE	non-linear	2017	arXiv:1712.09005
LargeVis	non-linear	2018	arXiv:1602.00370
UMAP	non-linear	2018	arXiv:1802.03426
PHATE	non-linear	2017	https://www.biorxiv.org/content/biorxiv/early/2018/06/28/120378.full.pdf
scvis	non-linear	2018	https://doi.org/10.1038/s41467-018-04368-5
VASC	non-linear	2018	https://doi.org/10.1016/j.gpb.2018.08.003

Dimensionality reduction

Matrix factorization

Graph-based

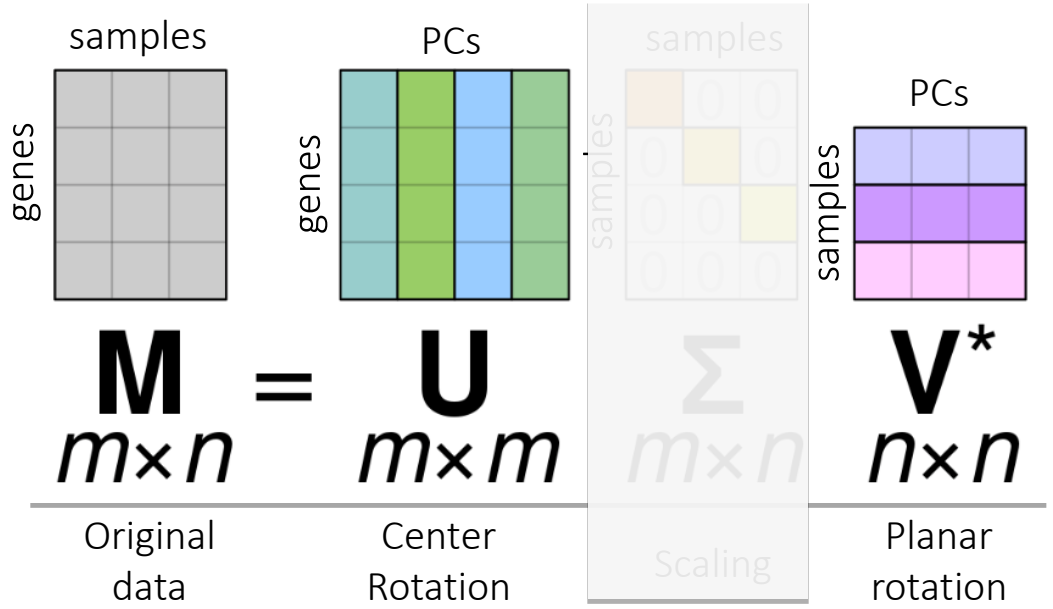
Auto-encoders

PCA	linear		
ICA	linear		
MDS	non-linear		
Sparce NNMF	non-linear	2010	https://pdfs.semanticscholar.org/664d/40258f12ad28ed0b7d4c272935ad72a150db.pdf
cPCA	non-linear	2018	https://doi.org/10.1038/s41467-018-04608-8
ZIFA	non-linear	2015	https://doi.org/10.1186/s13059-015-0805-z
ZINB-WaVE	non-linear	2018	https://doi.org/10.1038/s41467-017-02554-5
Diffusion maps	non-linear	2005	https://doi.org/10.1073/pnas.0500334102
Isomap	non-linear	2000	10.1126/science.290.5500.2319
t-SNE	non-linear	2008	https://lvdmaaten.github.io/publications/papers/JMLR_2008.pdf
- BH t-SNE	non-linear	2014	https://lvdmaaten.github.io/publications/papers/JMLR_2014.pdf
- Flt-SNE	non-linear	2017	arXiv:1712.09005
LargeVis	non-linear	2018	arXiv:1602.00370
UMAP	non-linear	2018	arXiv:1802.03426
PHATE	non-linear	2017	https://www.biorxiv.org/content/biorxiv/early/2018/06/28/120378.full.pdf
scvis	non-linear	2018	https://doi.org/10.1038/s41467-018-04368-5
VASC	non-linear	2018	https://doi.org/10.1016/j.gpb.2018.08.003

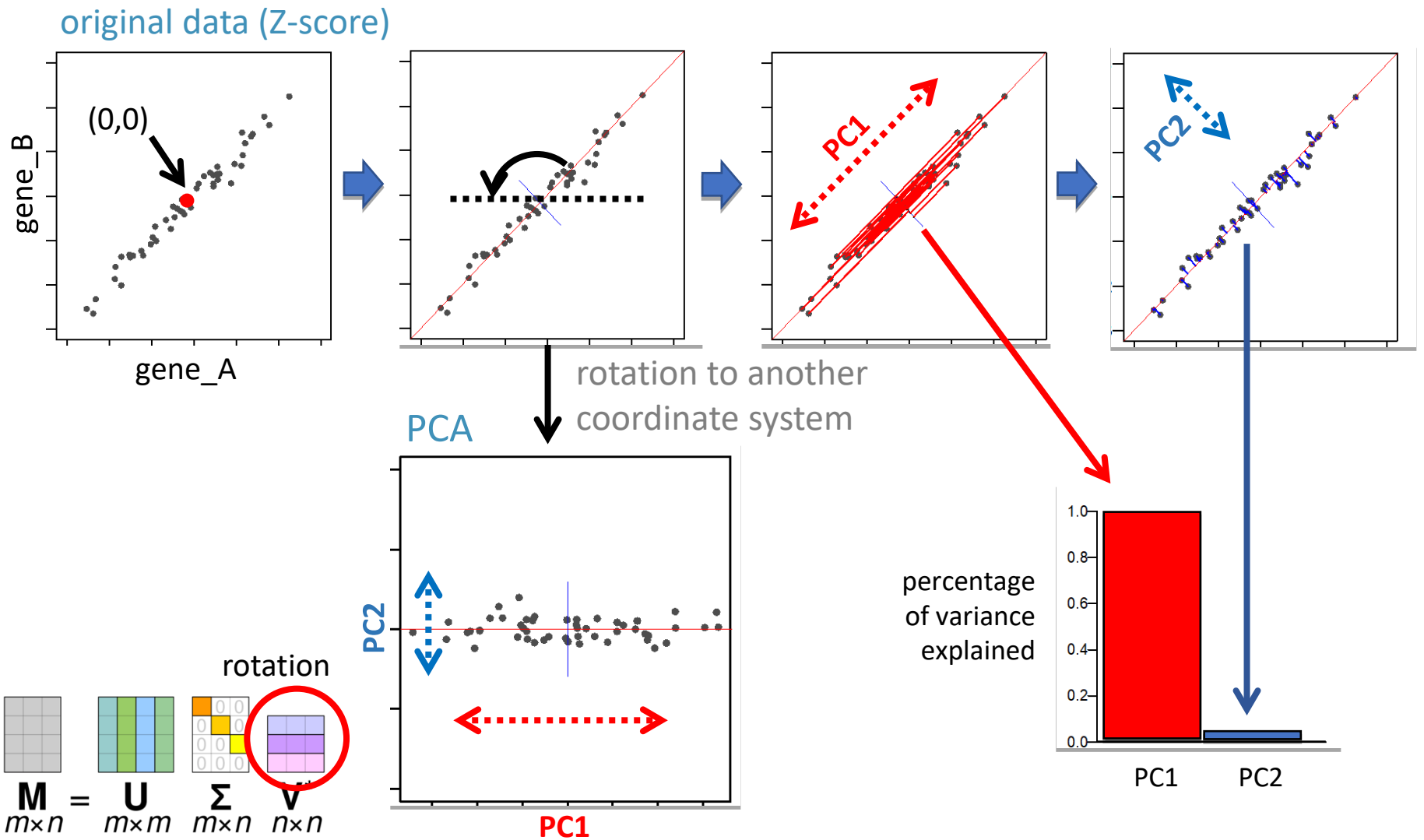
Principle Component Analysis (PCA)

- It is a LINEAR algebraic method of dimensionality reduction.

- It is a case inside Singular Value Decomposition (SVD) method (data compression)
  - Any matrix can be decomposed as a multiplication of other matrices (Matrix Factorization).

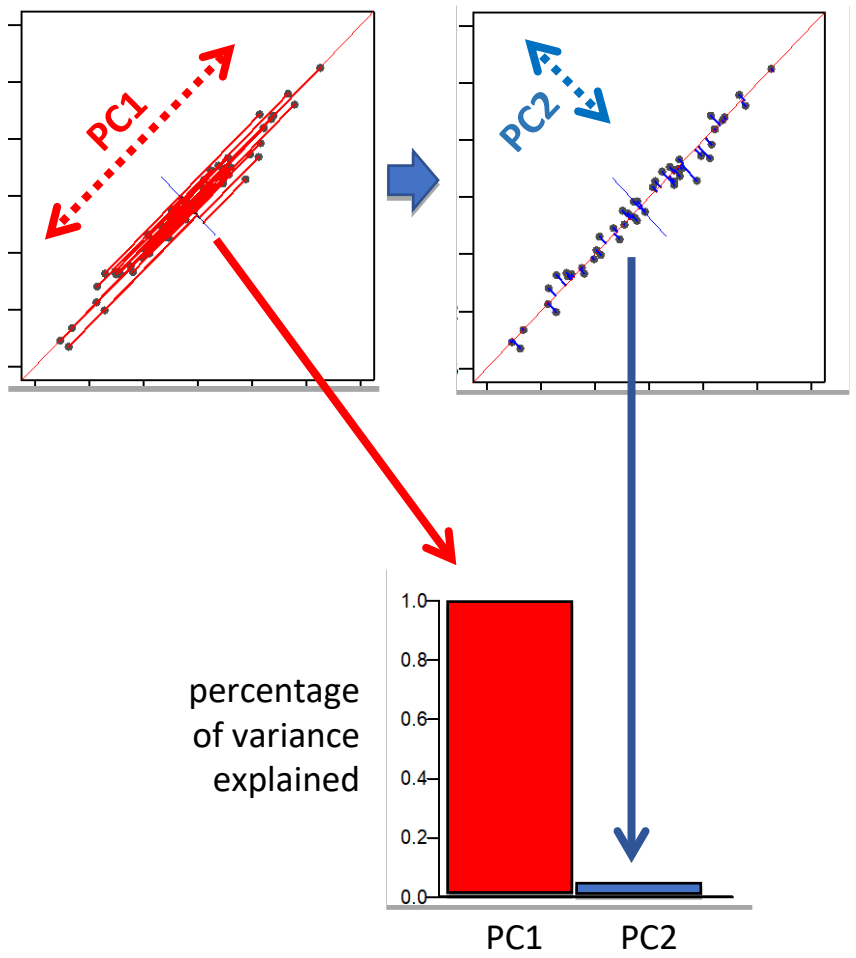


Principle Component Analysis (PCA)



Principle Component Analysis (PCA)

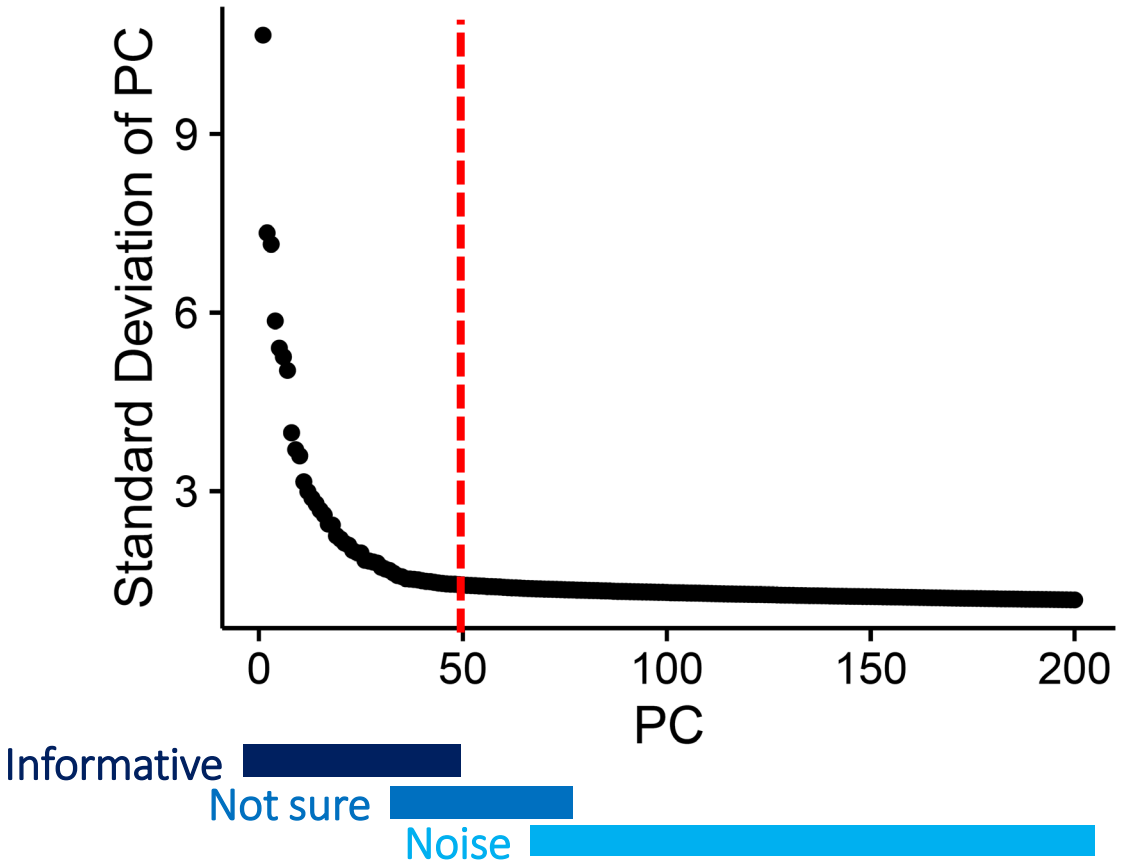
- PC1 explains >98% of the variance
- 1 PC thus represents 2 genes very well
  - "Removing" redundancy
- PC2 is nearly insignificant in this example
  - Could be disregarded



Principle Component Analysis (PCA)

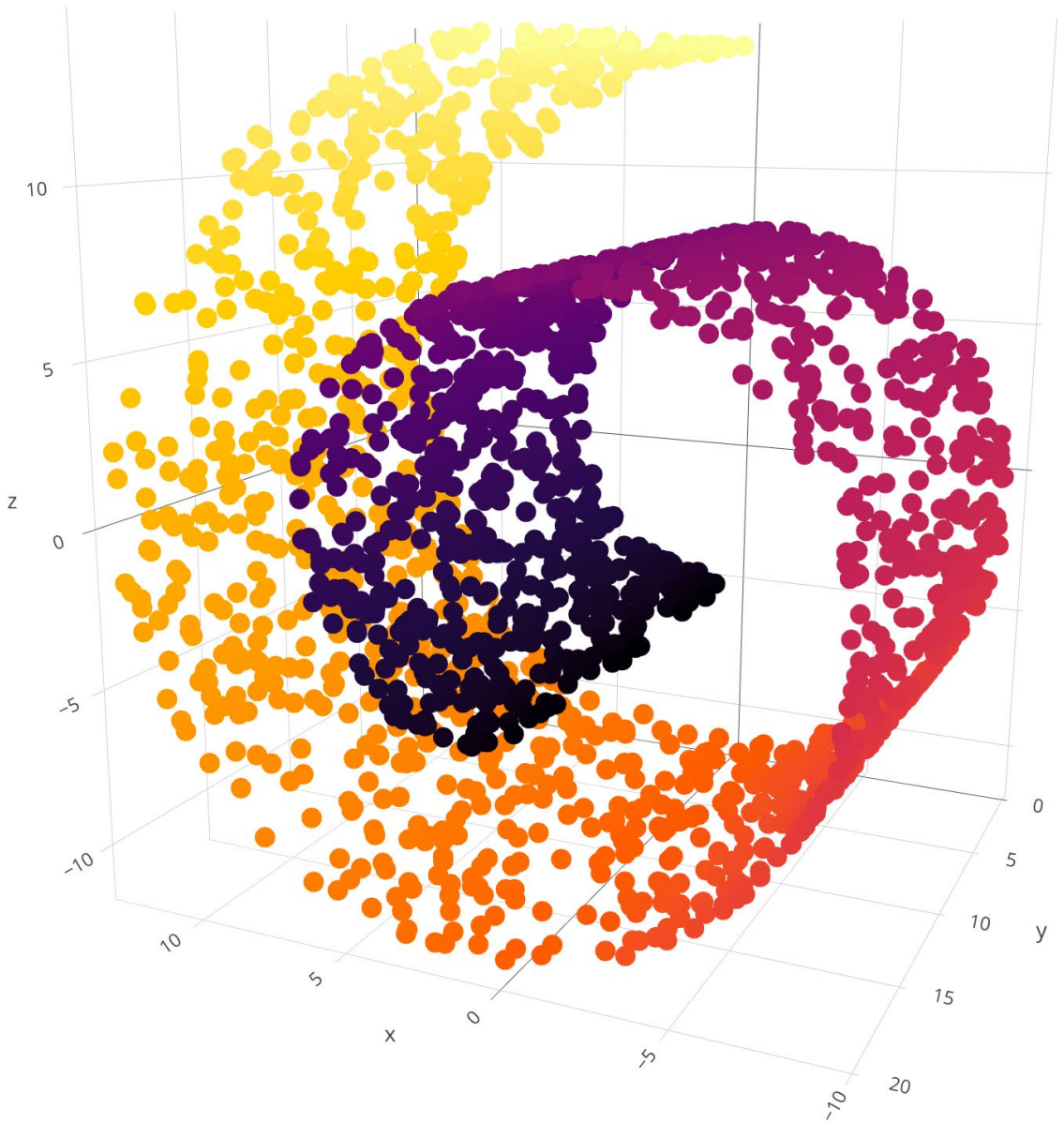
- In reality…

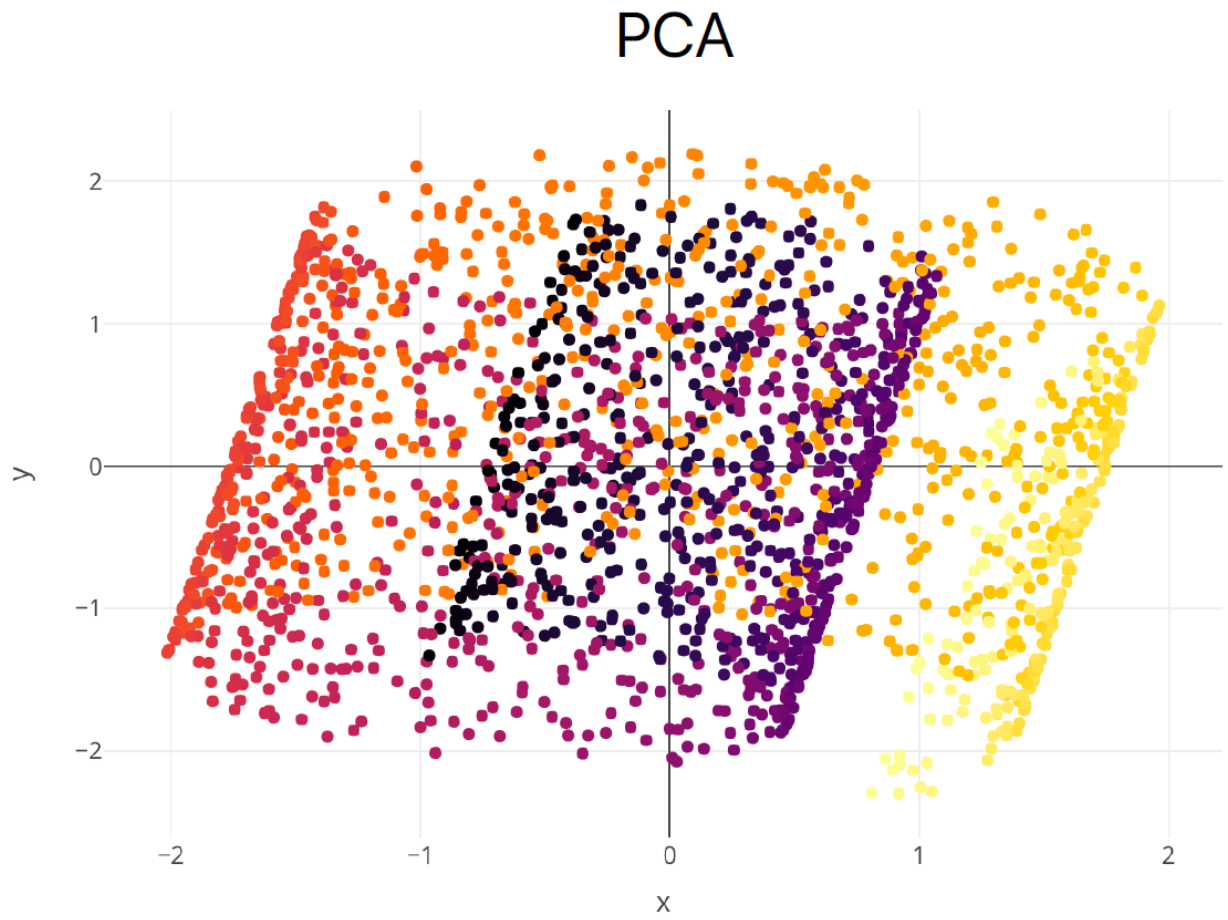
Scree/elbow plot

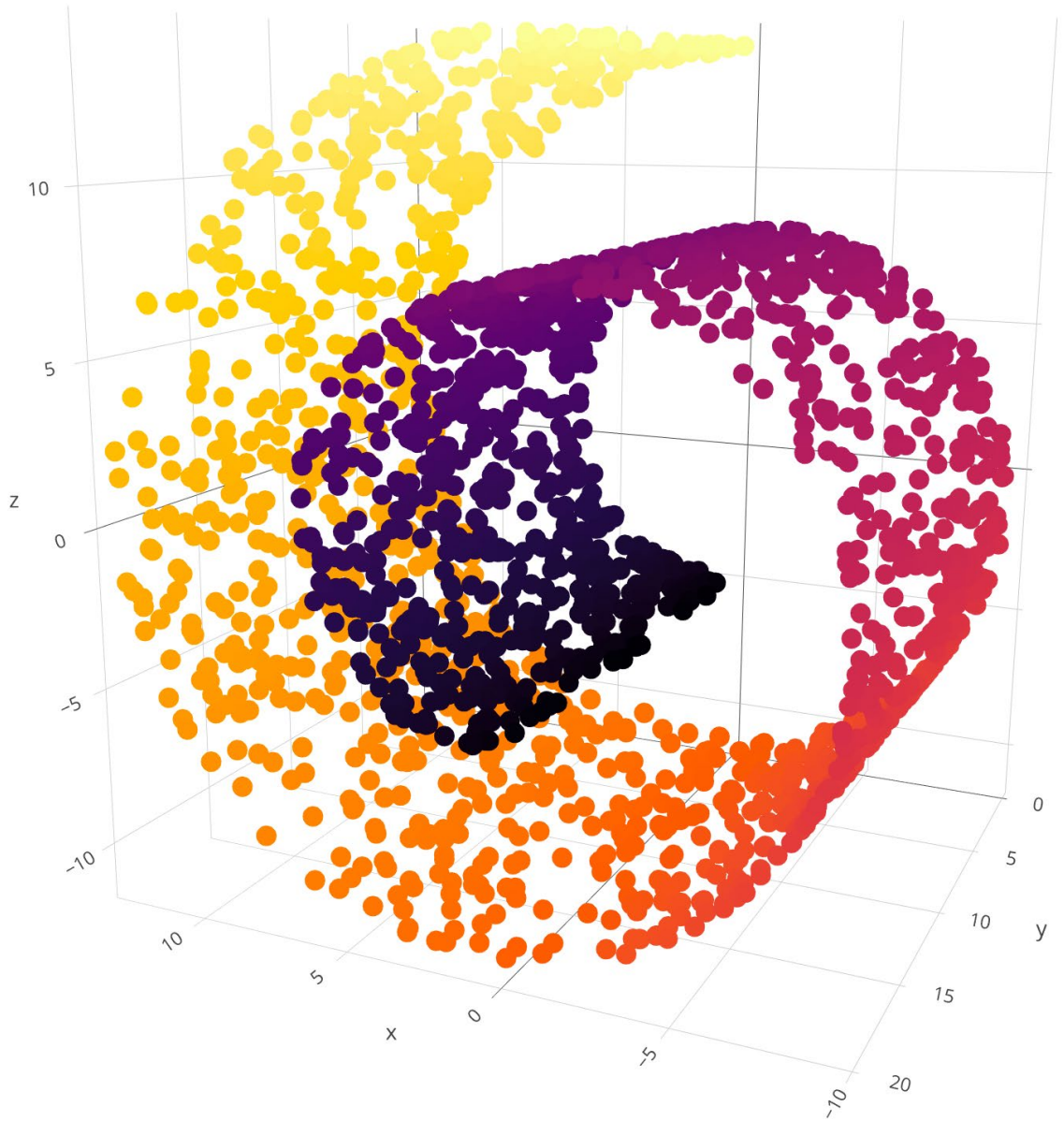


Principle Component Analysis (PCA)

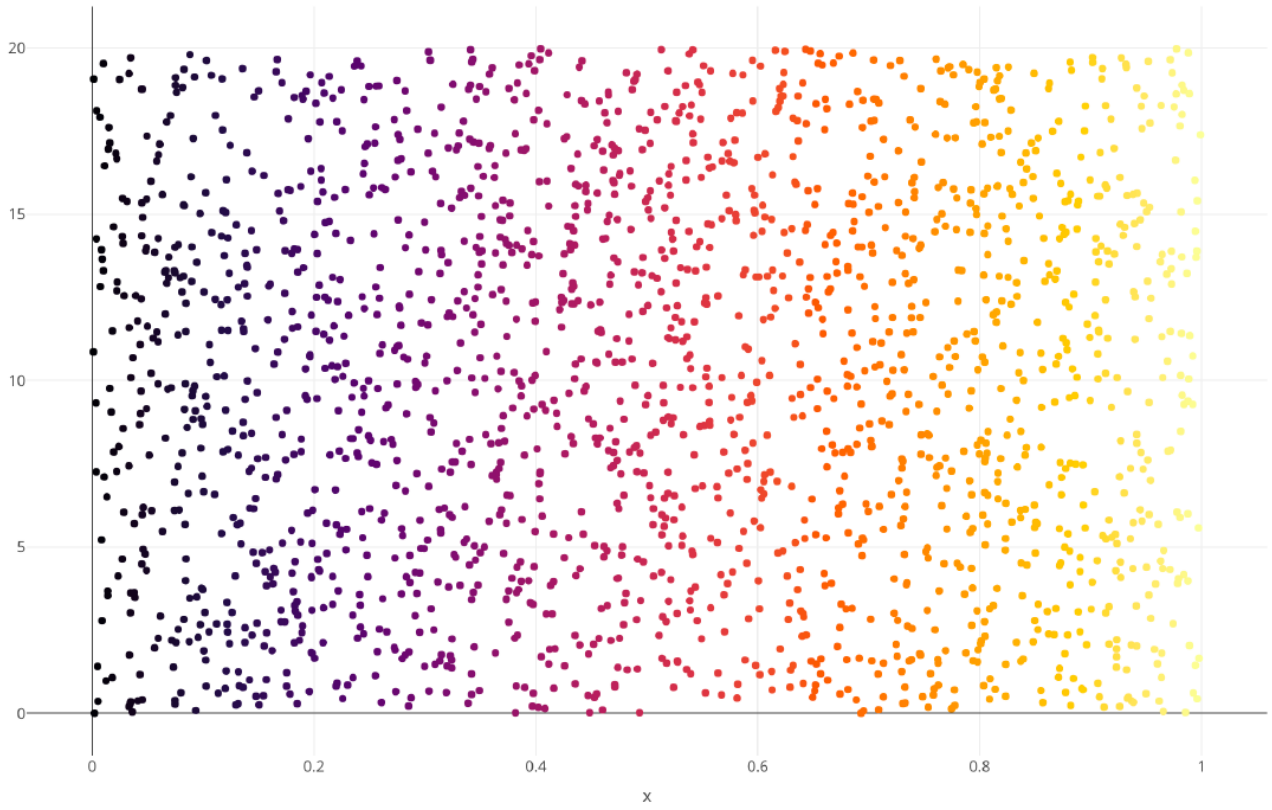
- LINEAR method of dimensionality reduction
- The TOP principal components contain higher variance from the data
- Can be used as FILTERING, by selecting only the top significant PCs
- *It is an interpretable/parametric dimensionality reduction*


10
5
z
0
−5
−10
10
5
x
0
−5
−10
0
5
10
y
15
20


PCA
2
1
y
0
−1
−2
−2
−1
0
1
2
x



Better

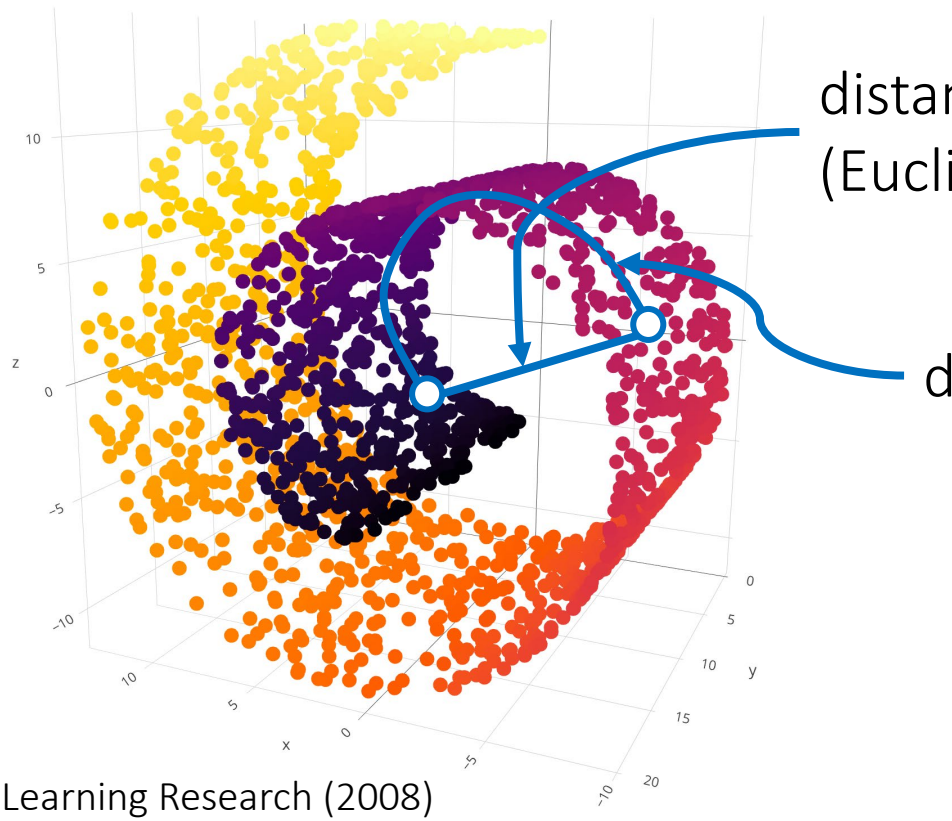


t-SNE

t-distributed stochastic neighborhood embedding

- It is a graph-based NON-LINEAR dimensionality reduction

Manifold



distance in PCA space
(Euclidean distance)

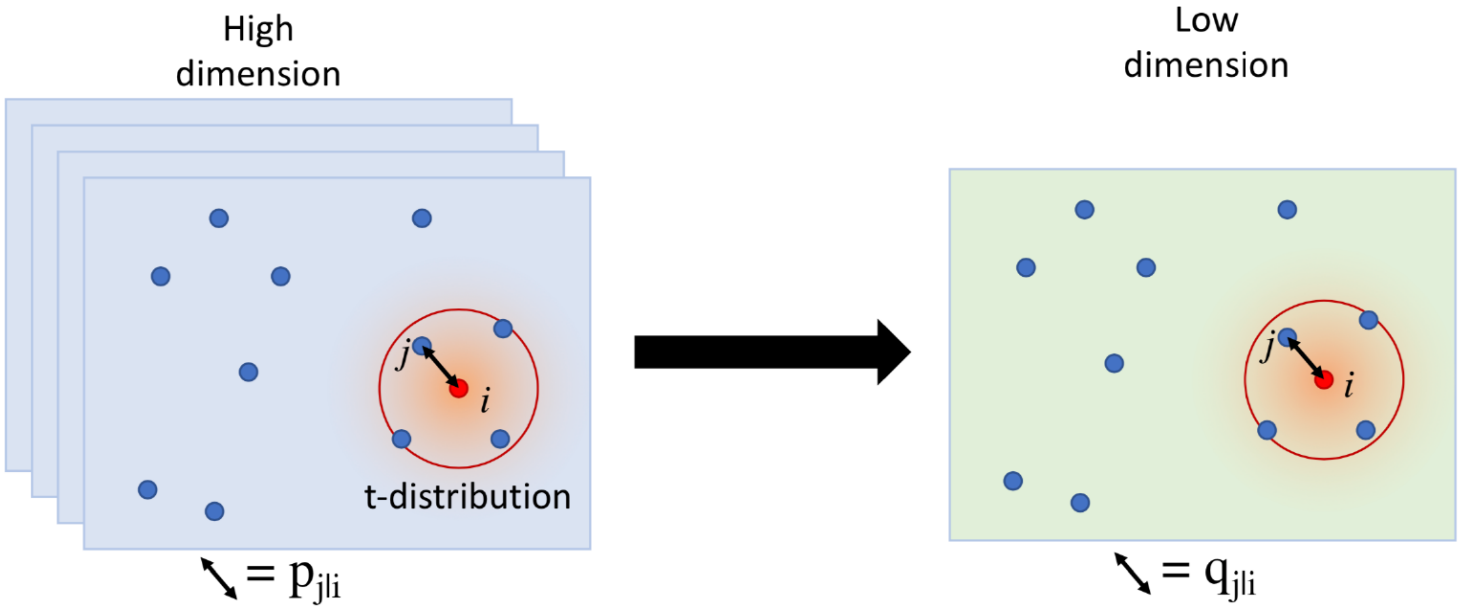
distance in tSNE space

t-SNE calculates the distances based on the distance to the neighbor cell

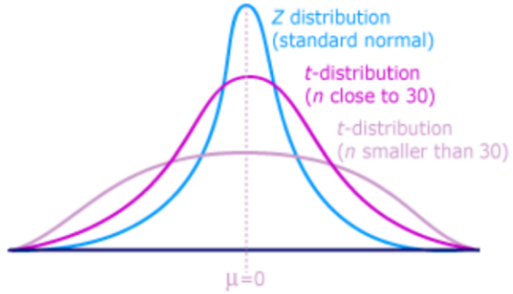
van der Maaten and Hinton, Journal of Machine Learning Research (2008)

t-SNE

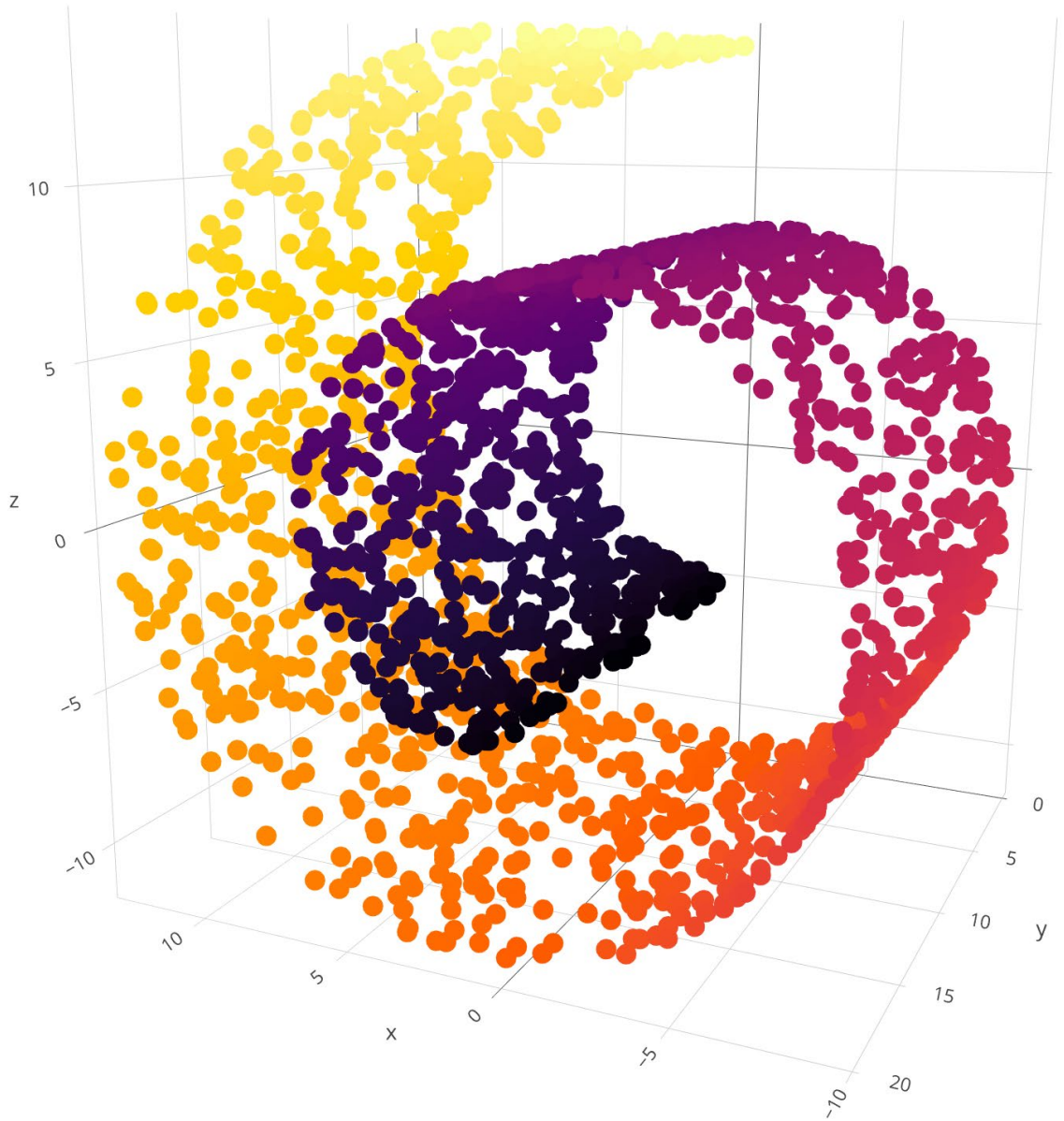
t-distributed stochastic neighborhood embedding

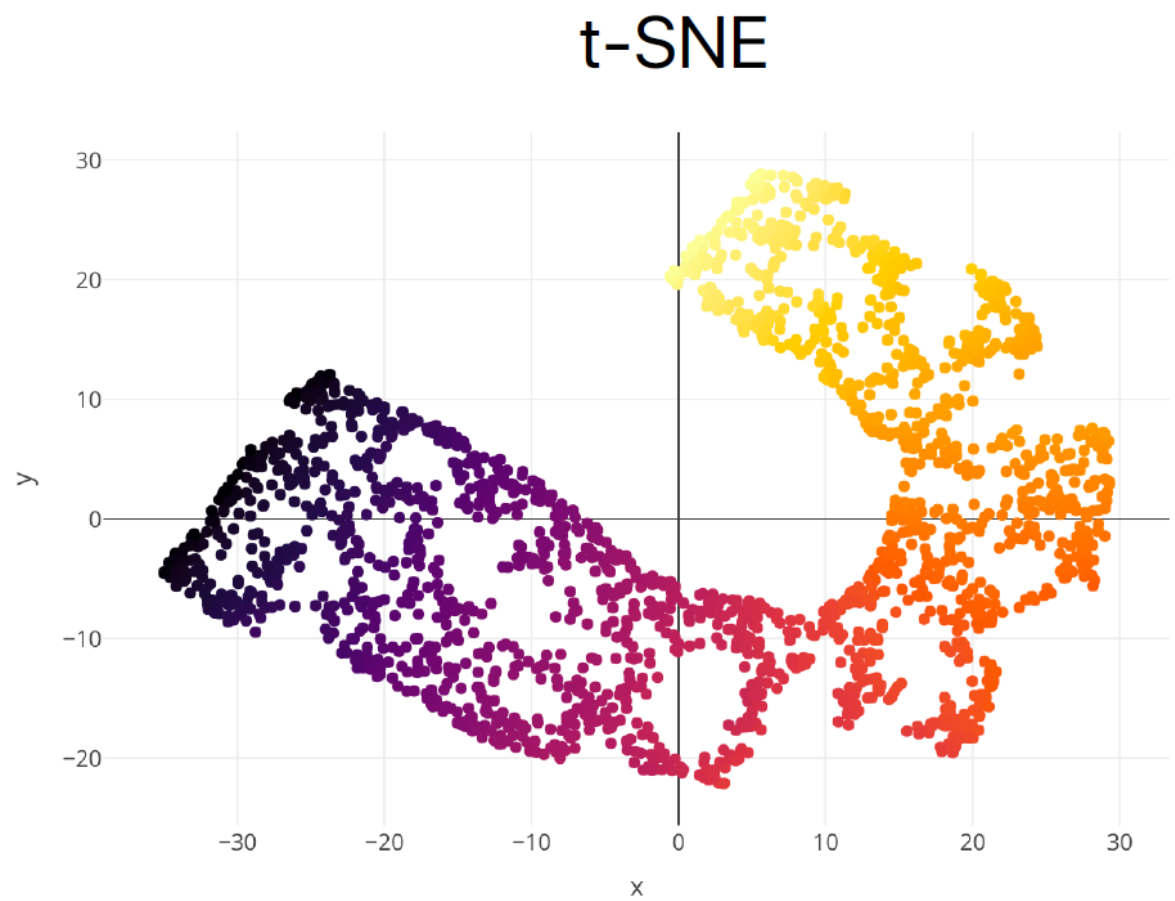


p_{jli} and q_{jli} measure the conditional probability that a point i would pick point j as it's nearest neighbor, in high (p) and low (q) dimensional space respectively.



van der Maaten and Hinton, Journal of Machine Learning Research (2008)


10
5
z
0
−5
−10
10
5
0
−5
−10
x
0
5
10
15
20
y


t-SNE
30
20
10
0
−10
−20
y
−30
−20
−10
0
10
20
30
x

t-SNE

t-distributed stochastic neighborhood embedding

- NON-LINEAR method of dimensionality reduction
- It is the current GOLD-STANDARD method in single cell data (including scRNA-seq)
- Can be run from the top PCs (e.g.: PC1 to PC10)

Problems

- It does not learn an explicit function to map new points
- It's cost function is not convex – This means that the optimal t-SNE cannot be computed
- Many hyper-parameters need to be defined empirically (dataset-specific)

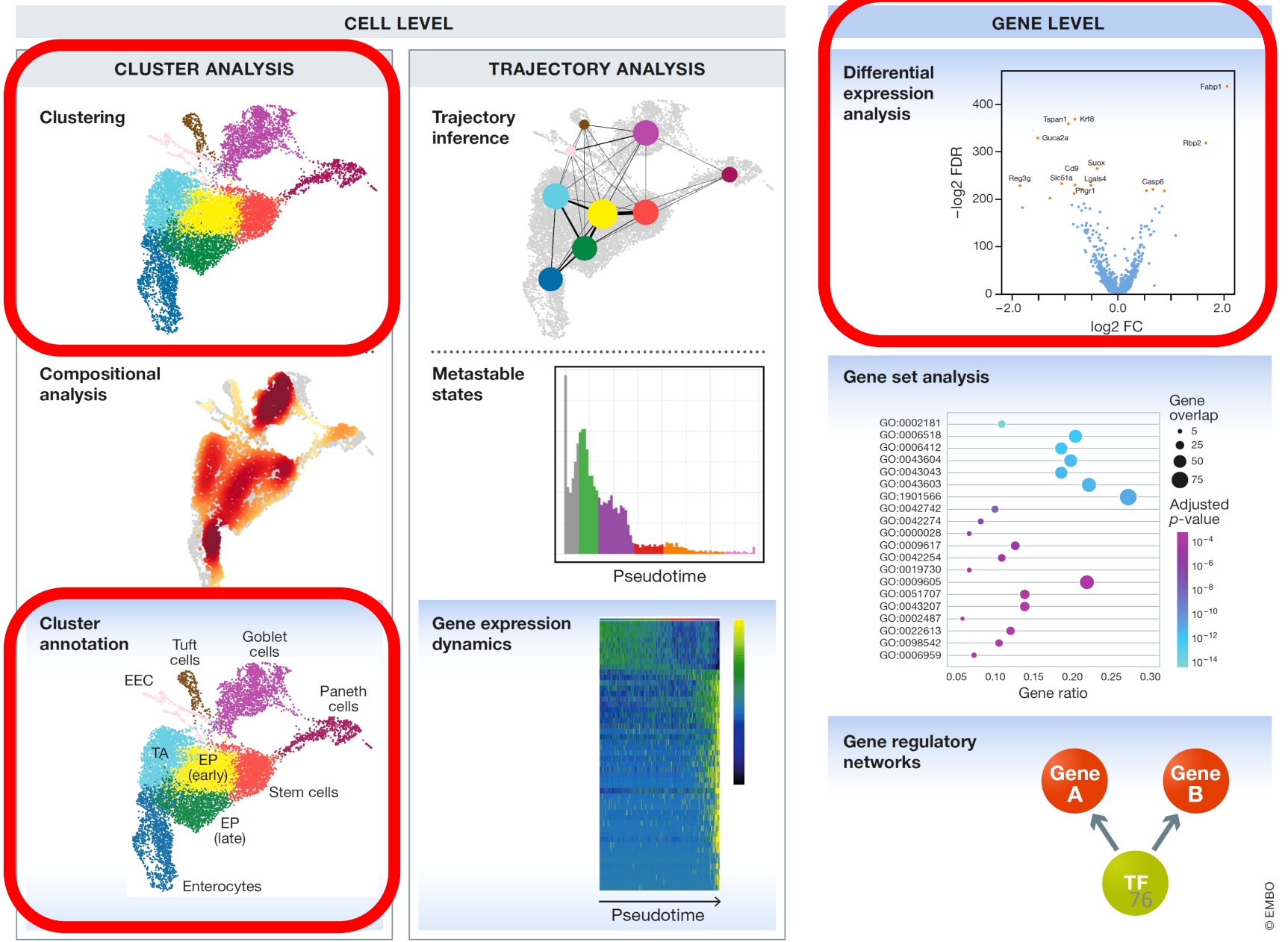
Dimensionality Reduction

UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction

Leland McInnes and John Healy
Tutte Institute for Mathematics and Computing
leland.mcinnes@gmail.com jchealy@gmail.com

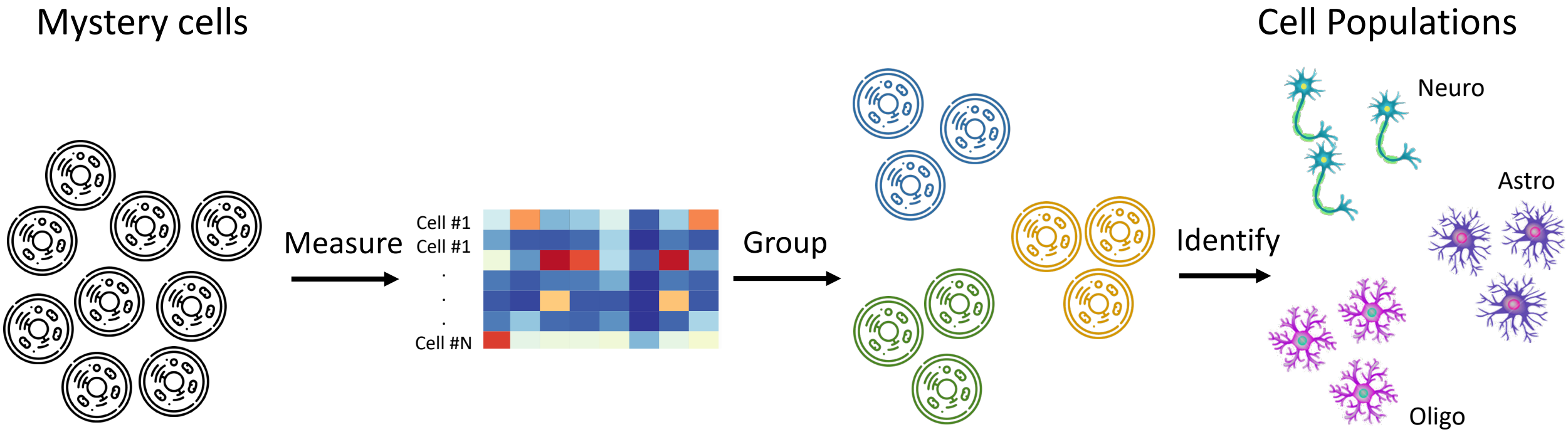
February 13, 2018

scRNA-seq Downstream Analysis

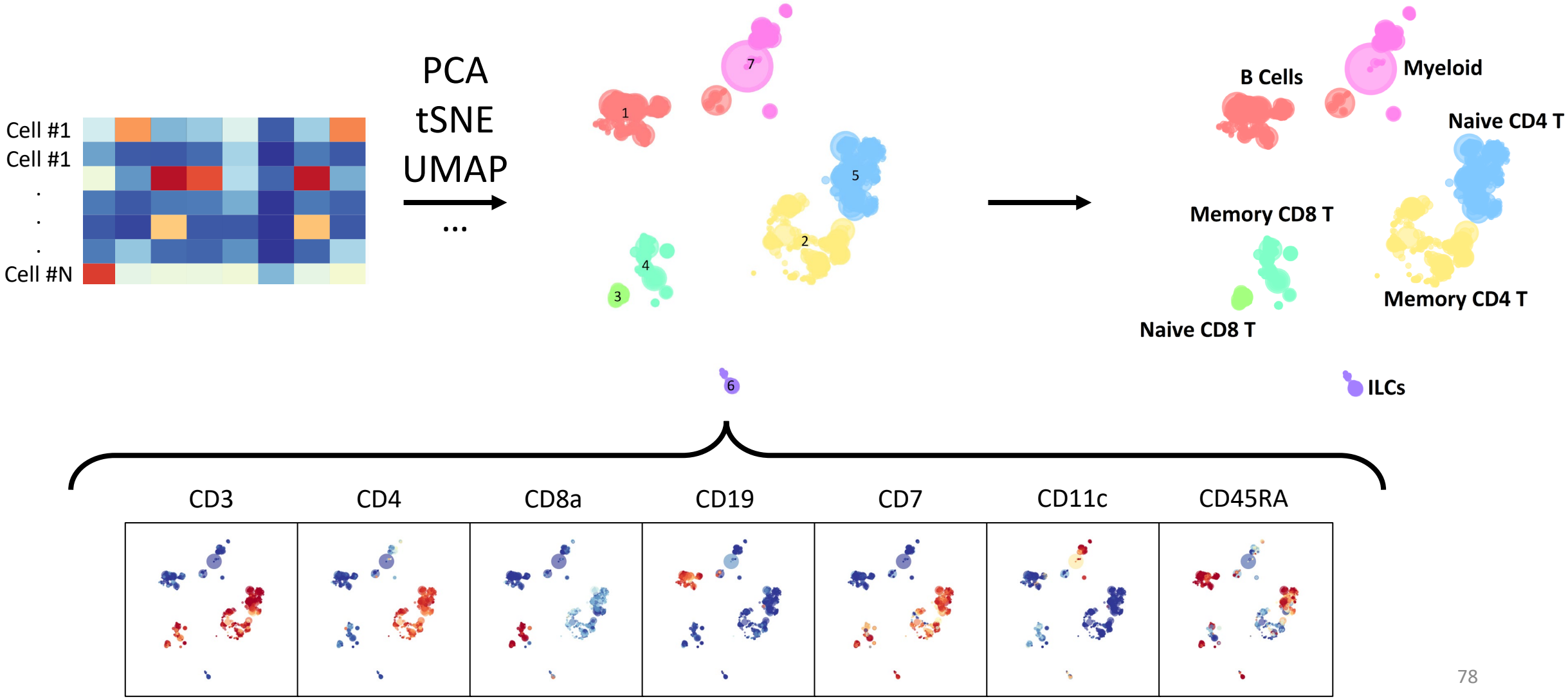


Luecken and Theis (MSB 2019)

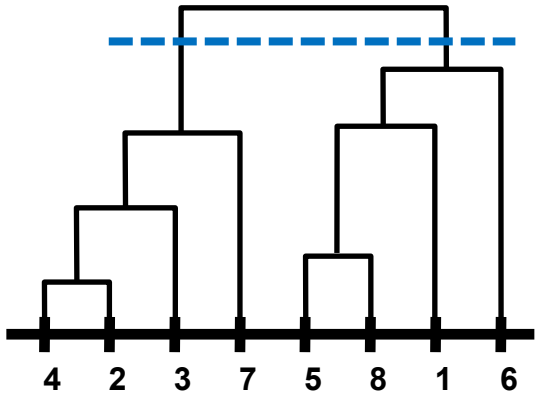
How can we identify cell populations?



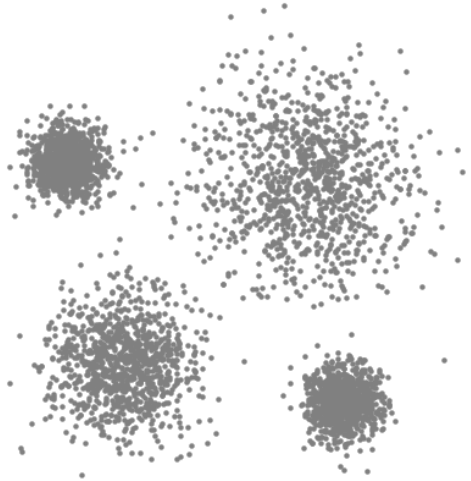
Unsupervised approach



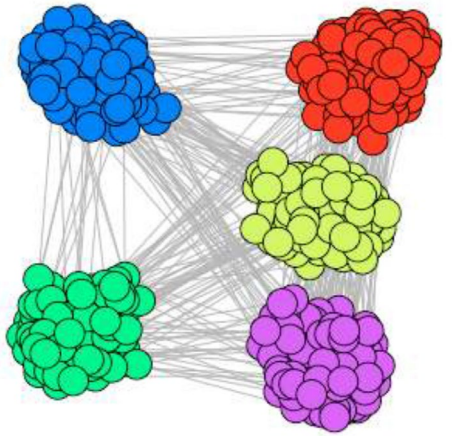
Many clustering approaches



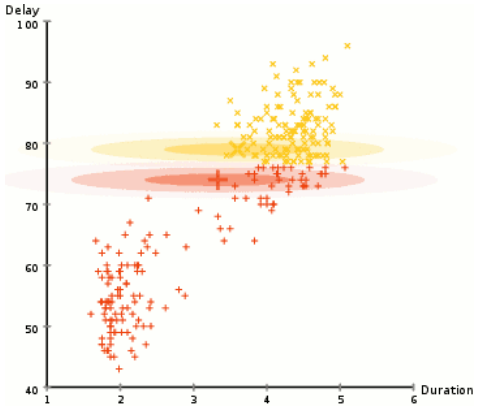
Hierarchical Clustering



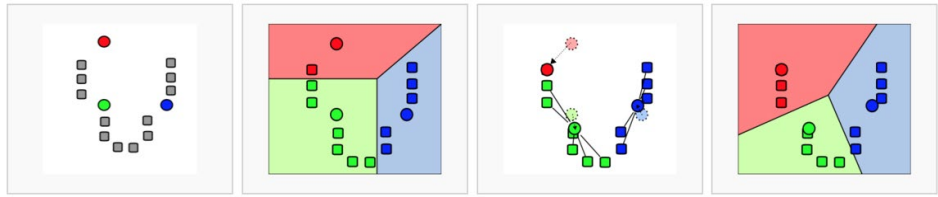
Mean shift clustering



Graph-based clustering



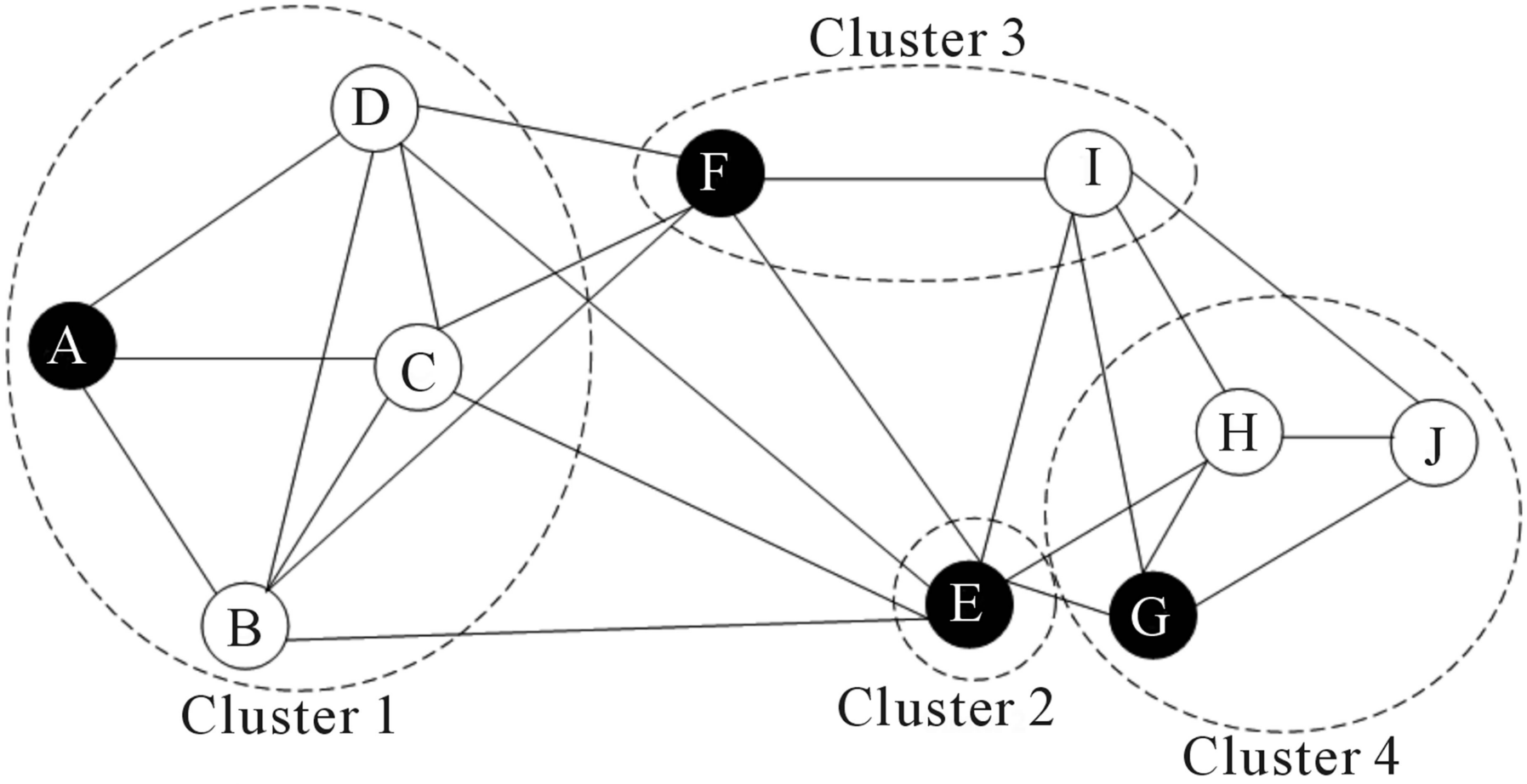
Gaussian mixture modeling



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.*

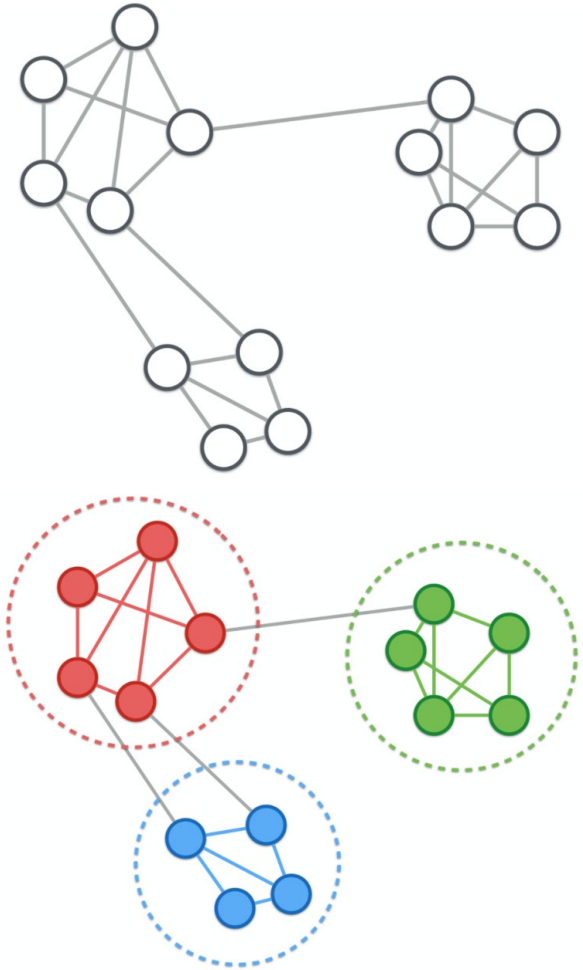
- **Shared Nearest Neighbor (SNN) graph**

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)

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

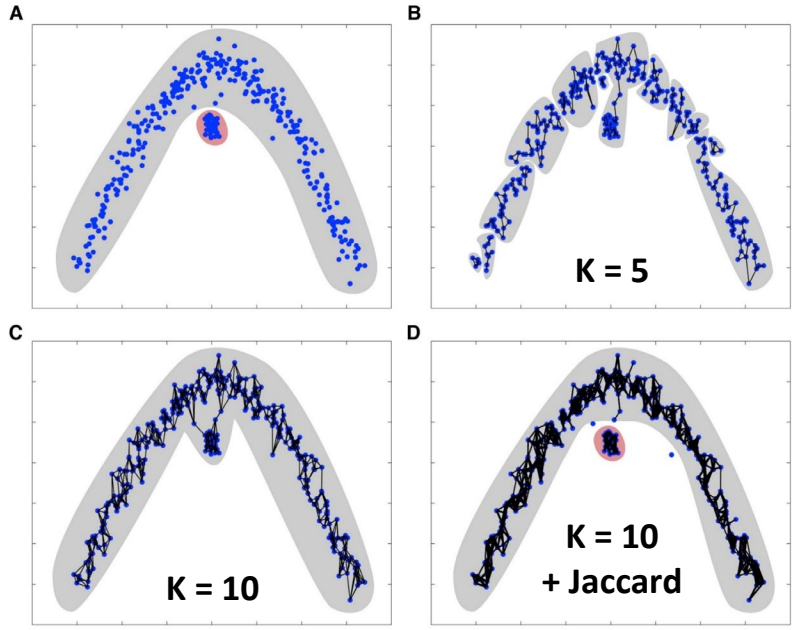
Kiselev et al. (https://doi.org/10.1038/s41576-018-0088-9)

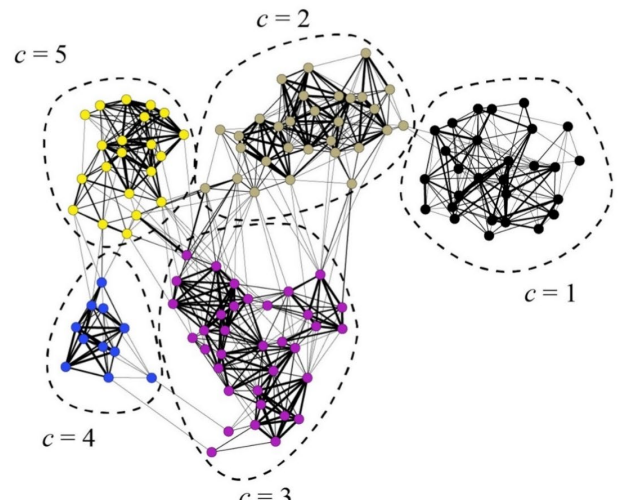
Seurat

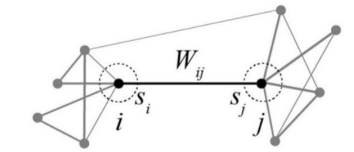
1) Construct KNN (k-nearest neighbor) graph based on the Euclidean distance in PCA space.

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/Leiden algorithm)



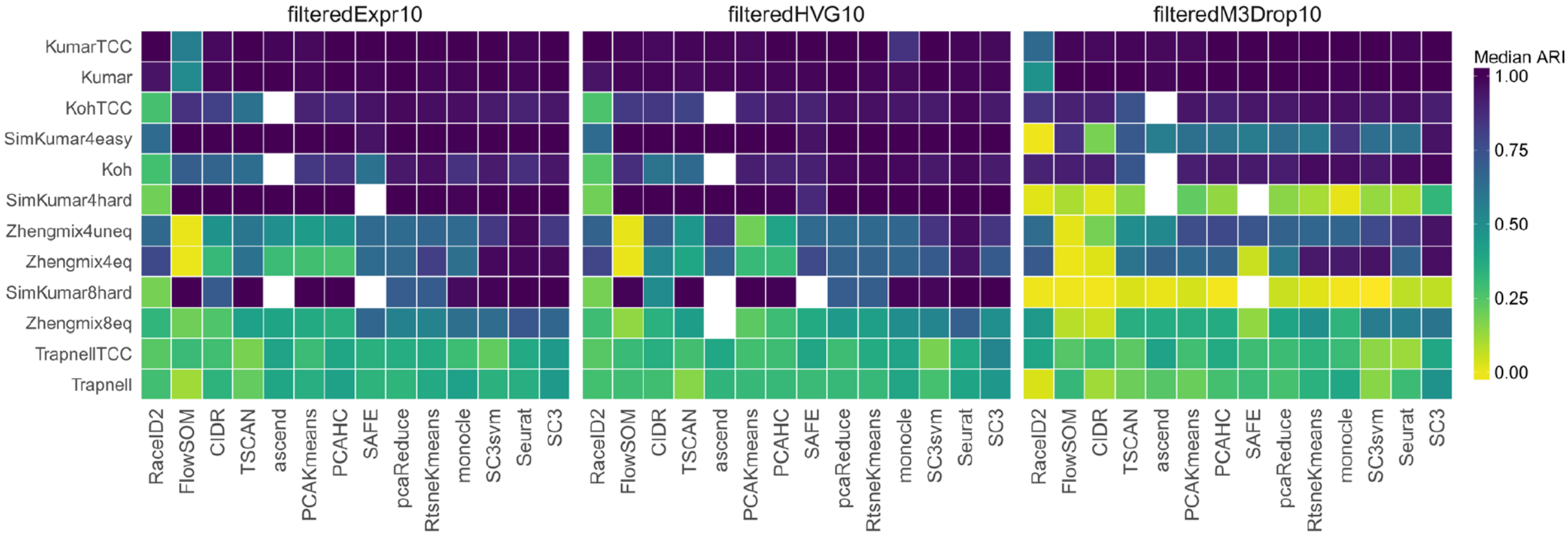




$$Q = \frac{1}{2m}\sum_{i,j}\left[W_{ij} - \frac{s_i s_j}{2m}\right]\delta(c_i, c_j)$$

Xu and Su (https://doi.org/10.1093/bioinformatics/btv088)
Levine et al. (https://doi.org/10.1016/j.cell.2015.05.047)

Benchmarking scRNA-seq clustering methods



Duò et al. (https://doi.org/10.12688/f1000research.15666.2)

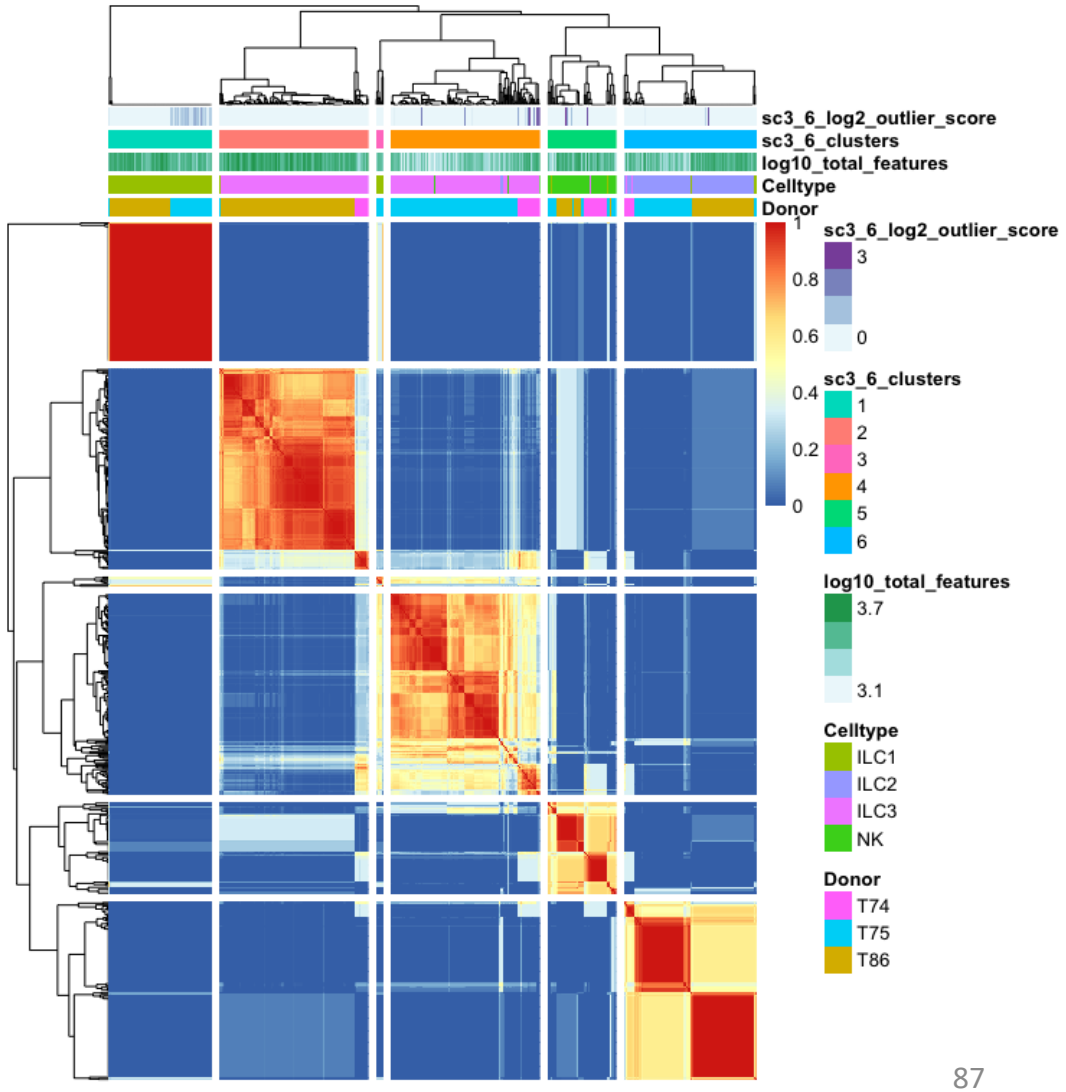
How many clusters do you really have?

- 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

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)

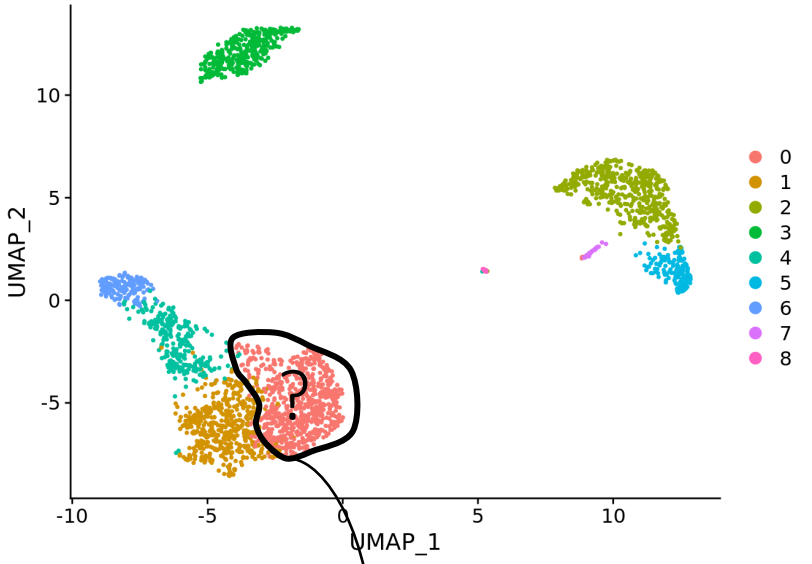
- CellMarker (http://biocc.hrbmu.edu.cn/CellMarker/)
  - Human: 13,605 cell markers of 467cell types in 158 tissues
  - Mouse: 9,148 cell makers of 389 cell types in 81 tissues
  - Zhang et al. (https://doi.org/10.1093/nar/gky900)

Challenges in clustering

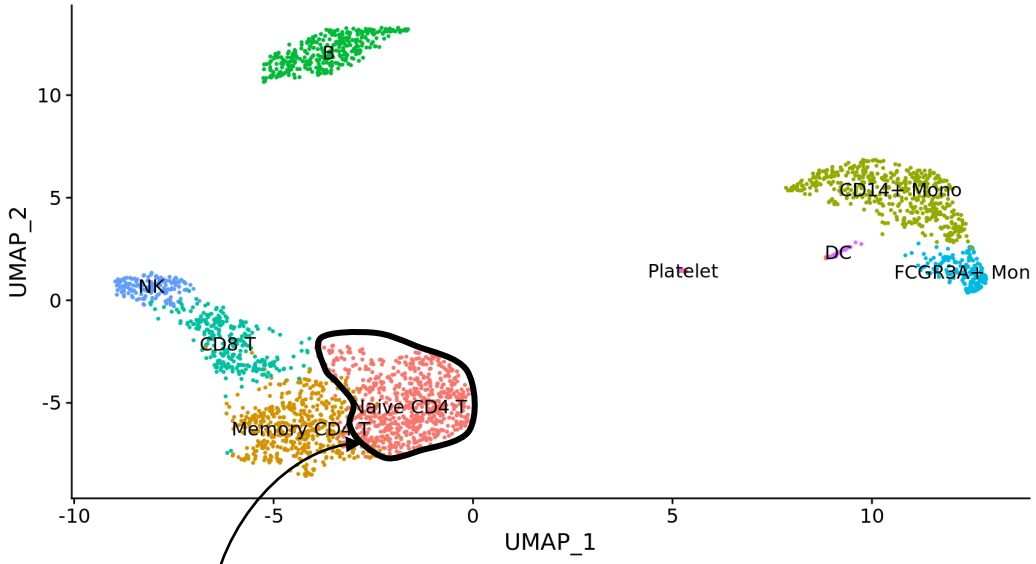
- What is a cell type?
- What is the number of clusters *k*?
- **Scalability**: in the last few years the number of cells in scRNA-seq experiments has grown by several orders of magnitude from ~10^2 to ~10^6

DE for cluster annotation

Unannotated clusters



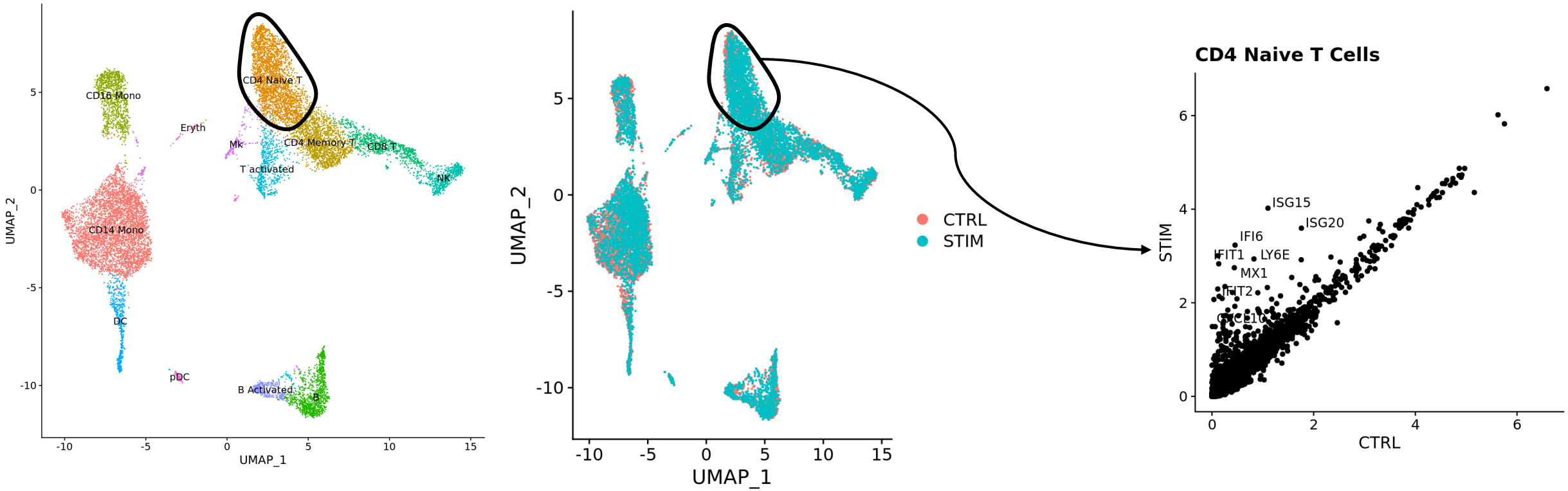
Annotated clusters



Compare *Cluster 0* to all other cells

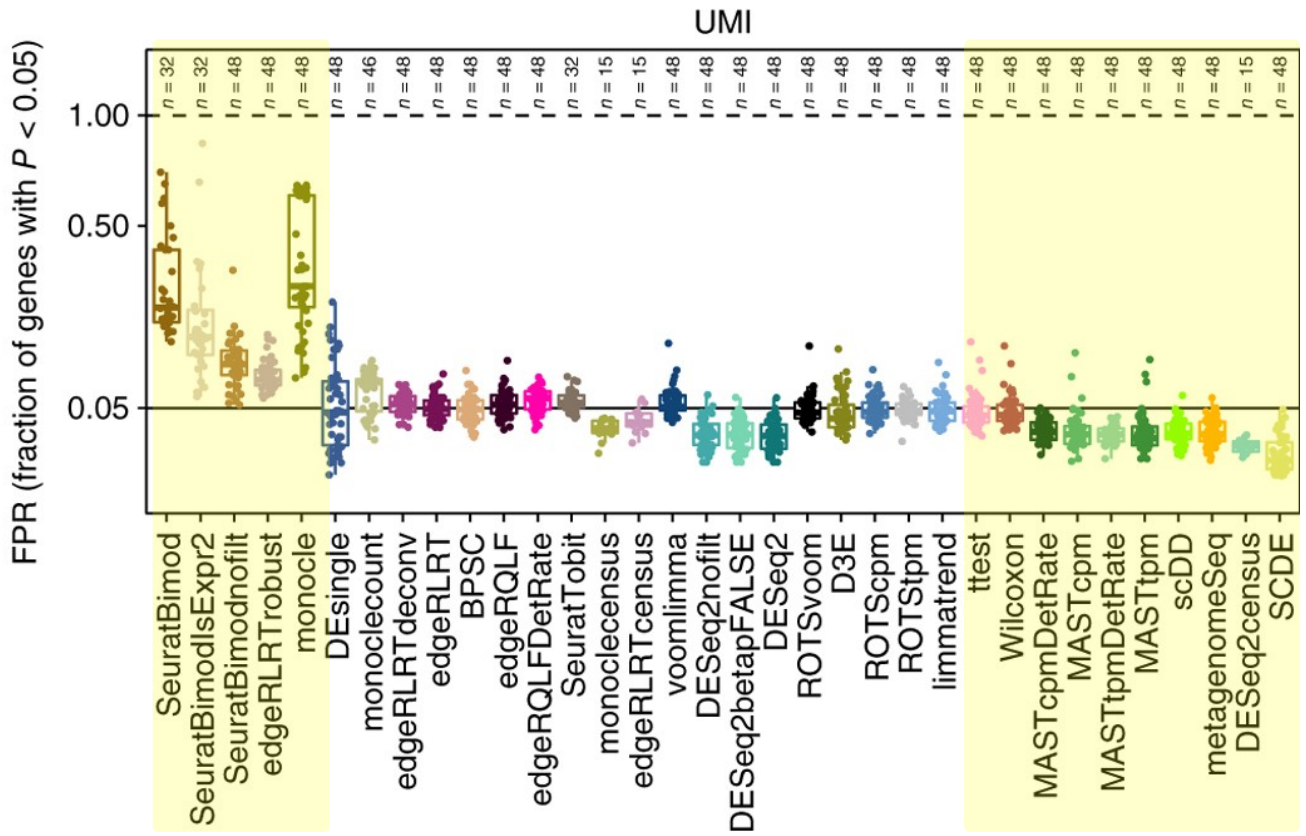
IL7R
CCR7

DE for comparing conditions



Comparing different methods

- Benchmark study (Soneson & Robinson, Nature Methods 2018)
- Overall, MAST, Wilcoxon, t-test outperformed other methods



Non-parametric tests

- Forget about modeling the data (it seems difficult), let's use a non-parametric test.
  - Svensson, *Droplet scRNA-seq is not zero-inflated*, Nature Biotechnology 2020
- No assumption that expression values follow any particular distribution
- Expression values are (generally) converted to ranks and test whether the distribution of ranks for one group are significantly different from the distribution of ranks for the other group.
- Assumption: distributions have the same shape in both groups

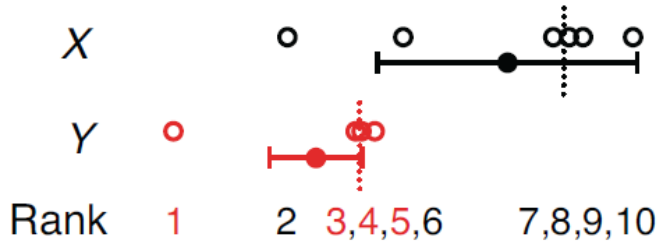
Wilcoxon rank-sum test aka Mann-Whitney U test

- H_0: median$_1$ = median$_2$
- Start by ranking all values
- Calculate the test statistic:

$$U = W - \frac{n_Y(n_Y+1)}{2}$$

sum of ranks in the smaller-sized sample

The lowest possible rank in the sample with the lower ranks



$$W = 1 + 3 + 4 + 5 = 13$$
$$U' = W - n_Y(n_Y + 1)/2$$
$$= 13 - 10$$
$$= 3$$

For cases in which both samples are larger than 10, the distribution of *U* is approximately normal

Example from: Krzywinski M, Altman N (2014) Nat Methods 11:467–469.

That must be the solution to everything?

- Not really…
- Wilcoxon rank sum test is not as powerful as parametric tests, i.e. it requires more data points to detect the same effects
- Might fail to deal with a large number of tied values, such as the case for zeros in single-cell RNA-seq expression data.

Single-cell DE in practice

Seurat

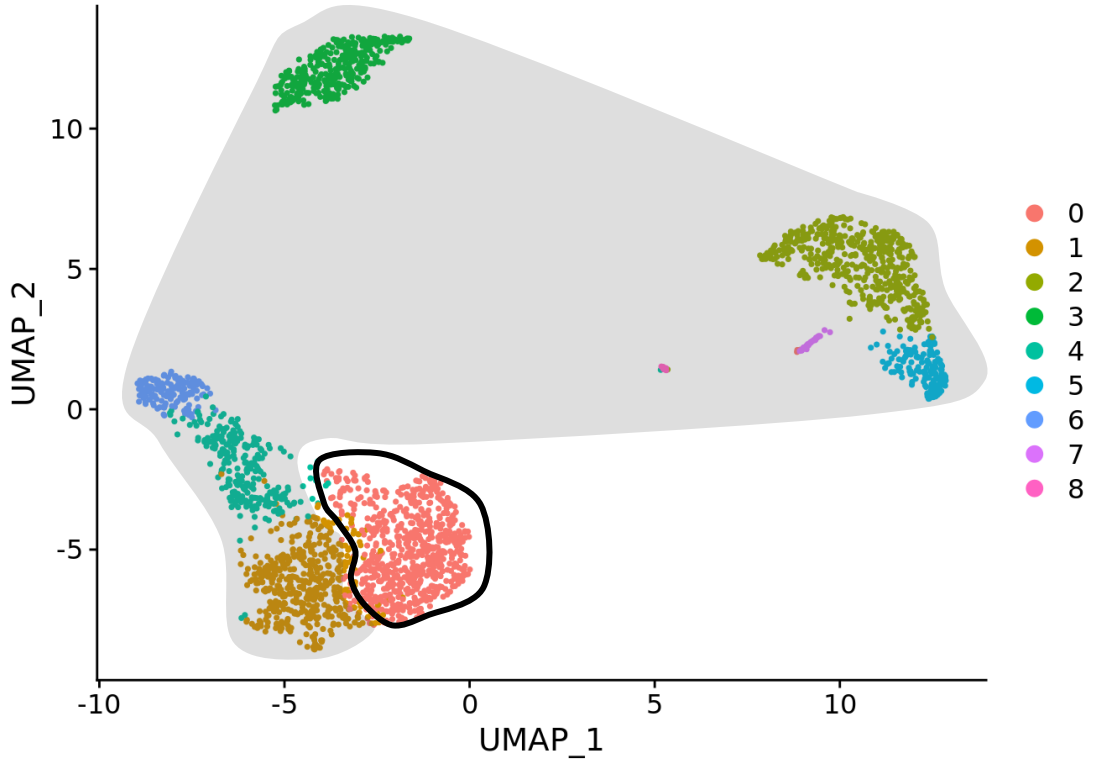
- "wilcox" : Wilcoxon rank sum test (default)
- "bimod" : Likelihood-ratio test for single cell feature expression, (McDavid et al., Bioinformatics, 2013)
- "roc" : Standard AUC classifier
- "t" : Student's t-test
- "poisson" : Likelihood ratio test assuming an underlying negative binomial distribution. Use only for UMI-based datasets
- "negbinom" : Likelihood ratio test assuming an underlying negative binomial distribution. Use only for UMI-based datasets
- "LR" : Uses a logistic regression framework to determine differentially expressed genes. Constructs a logistic regression model predicting group membership based on each feature individually and compares this to a null model with a likelihood ratio test.
- "MAST" : GLM-framework that treates cellular detection rate as a covariate (Finak et al, Genome Biology, 2015) (Installation instructions)
- "DESeq2" : DE based on a model using the negative binomial distribution (Love et al, Genome Biology, 2014) (Installation instructions)

https://satijalab.org/seurat/v3.1/de_vignette.html

Identifying cluster markers

- Approach 1: one-vs-all (default is Seurat)

- Limitations:
  - Sensitive to the population composition (one dominant population can drive marker selection for every other cluster)

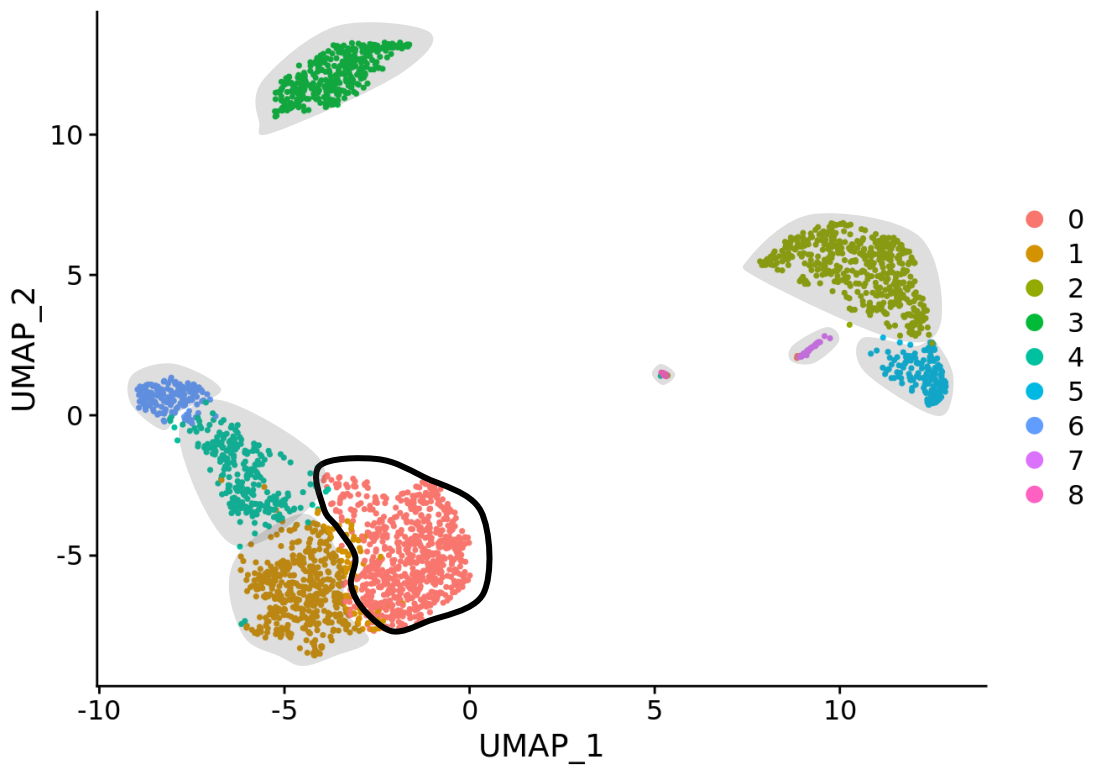


Identifying cluster markers

- Approach 2: multiple pairwise comparisons (default in scran)

- Strategies to combine results:
  - Prioritize genes significant in ***any*** pairwise comparison -> focuses on combinations of genes that (together) drive separation of a cluster from the others
  - Prioritize genes significant in ***all*** pairwise comparisons -> explicitly favors genes that are uniquely expressed in a cluster (too stringent)

- Limitations:
  - How to combine and report results?
  - Slow



Additional (practical) considerations

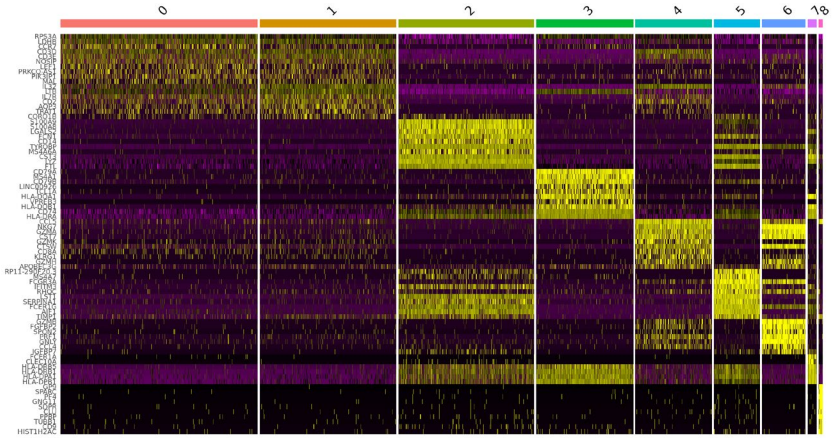
- Focus on ***positive*** markers only
  - It is difficult to interpret and experimentally validate the absence of expression

- Focus on genes with ***large effect size*** (log fold-change, LFC)
  - More biologically interesting markers (e.g. possible to validate with qPCR)
  - Faster testing (in Seurat)

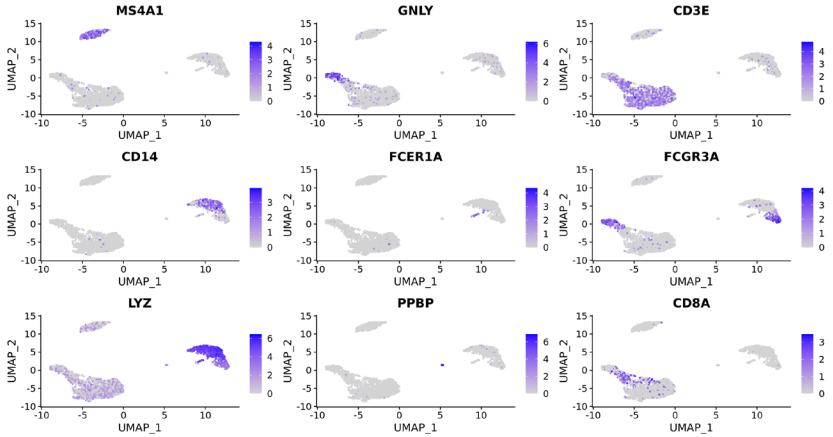
- Filter genes that are very infrequently detected in either group of cells
  - Seurat: `min.pct, logfc.threshold, min.diff.pct, max.cells.per.ident`

Check the identified markers

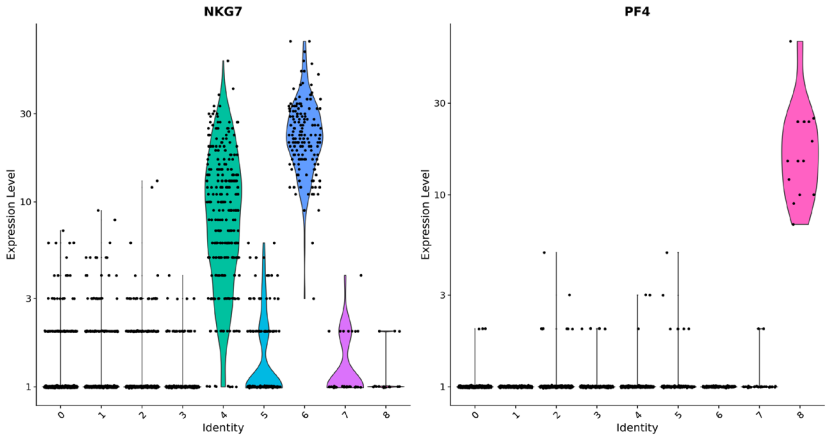
Heatmap



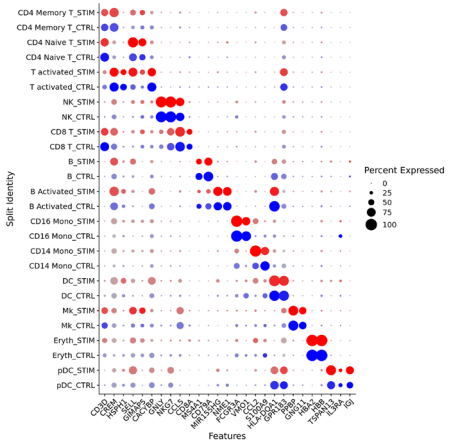
Overlap on tSNE/UMAP



Violinplot



Dotplot



End Part 2

Summary of today

- Overview of single-cell assays/platforms/protocols
- Quality control
- Normalization
- Data integration
- Feature selection
- Dimensionality reduction
- Cell type identification

There is more to it…

- Constructing the cell x gene matrix
- Trajectory inference
- Single cell regulatory networks
- Imputation
- Single cell multi-omics
- Sample multiplexing
- Single cell isoform sequencing
- Cell lineage + scRNA-seq
- Spatial transcriptomics
- …

Single cell analysis course

- MGC/BioSB Single cell analysis course

https://github.com/LeidenCBC/MGC-BioSB-SingleCellAnalysis2020

Useful Resources

- Best practices in single cell RNA-seq analysis (Luecken & Theis, MSB 2019)

https://www.embopress.org/doi/pdf/10.15252/msb.20188746

- Orchestrating Single-Cell Analysis with Bioconductor

https://osca.bioconductor.org/

- Single Cell Course (Martin Hemberg Lab, Welcome 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/SingleCellGenomcisDay18_References.pdf?dl=1

Thank You!

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