Single Cell Transcriptomics

Analysis of Next-Generation Sequencing Data

Friederike Dündar

Applied Bioinformatics Core

Slides at https://bit.ly/2T3sjRg[1]

March 17, 2020

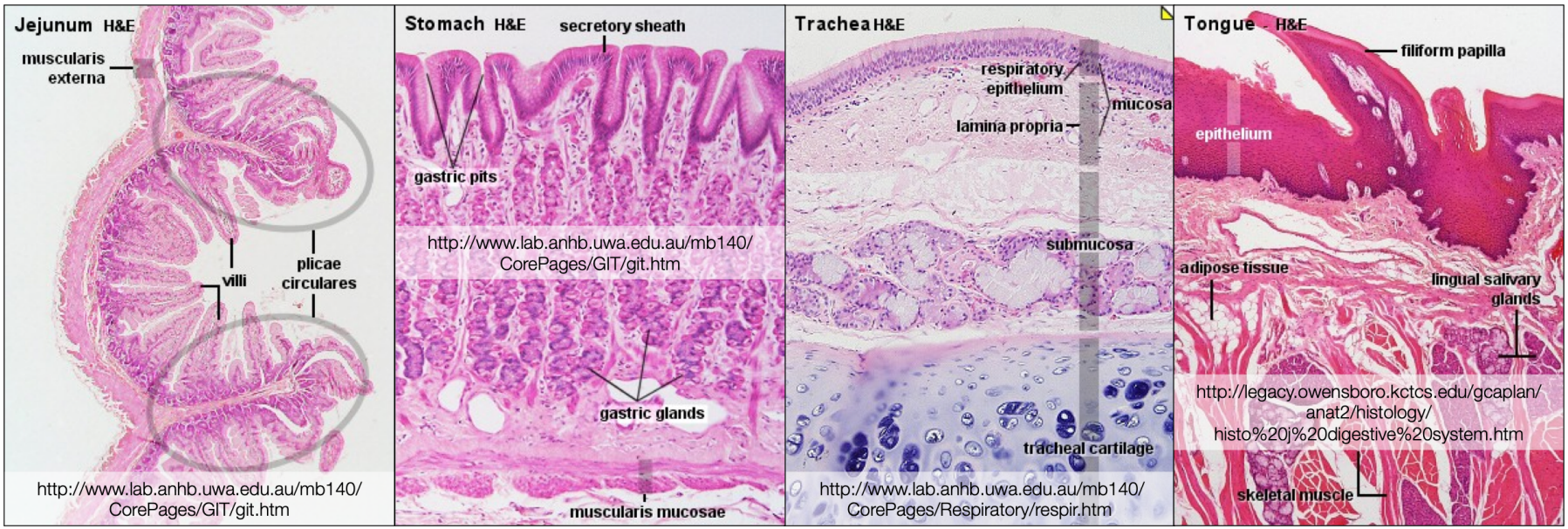
[1] https://physiology.med.cornell.edu/faculty/skrabanek/lab/angsd/schedule_2020/

1. Why measure single cells?
2. How to sequence the transcriptome of single cells?
3. Processing of scRNA-seq data
4. How to draw biologically meaningful insights from scRNA-seq?
5. Conclusions
6. References

Why measure single cells?

Bulk RNA-seq returns the average expression of an entire cell population.

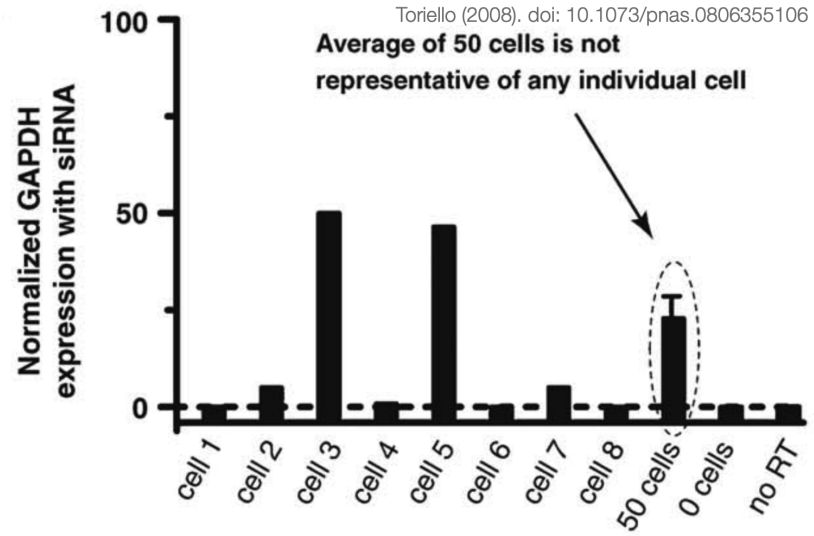
1. Tissues/organs[2] are usually made up of **very different** types of cells that are often **difficult to separate** prior to the experiment.
   - endothelial cells, osteocytes, myocytes, neurons, lymphocytes, macrophages, erythrocytes, oocytes, alveolar cells, chondrocytes, ...
   - stem cells, secreting cells, metabolizing cells, pacemaker cells, ...

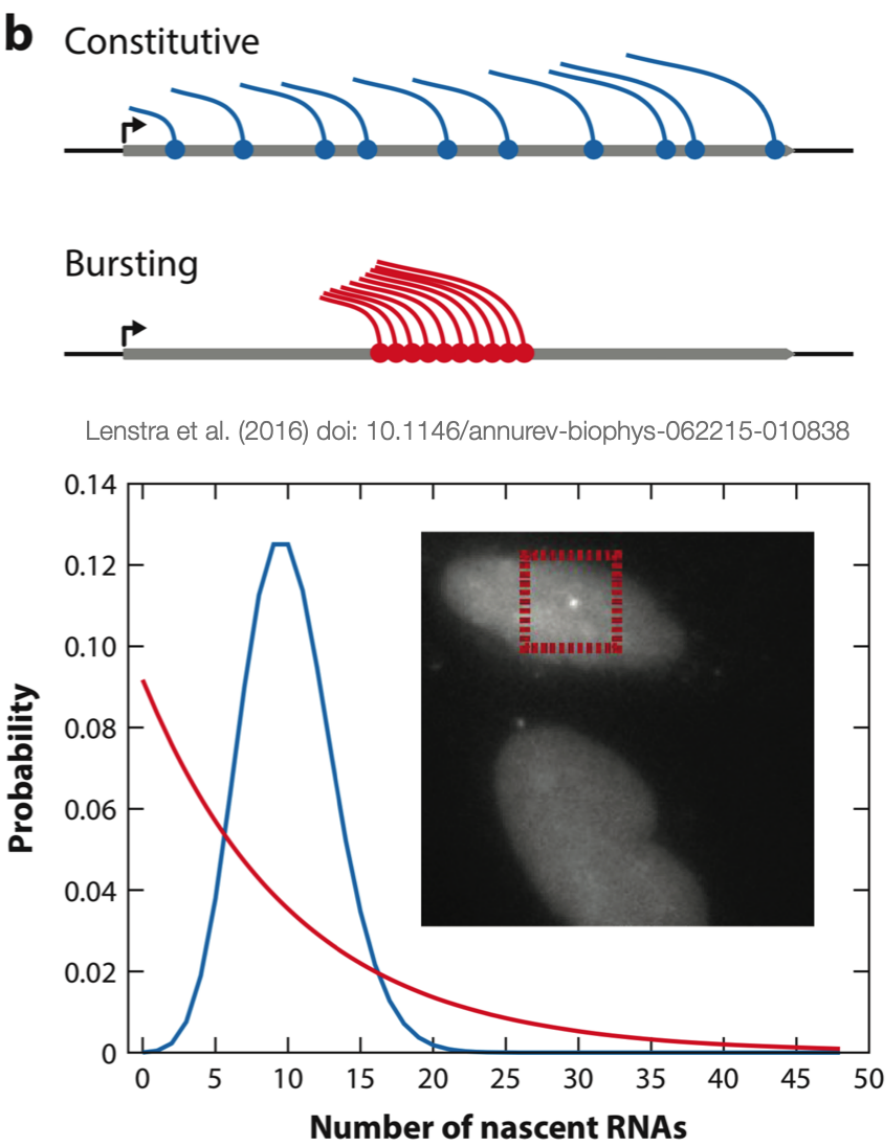


[2] Many solid tumors, too.

Why is bulk RNA-seq not enough?

2. Even very similar cells/clonal cell cultures display **heterogeneity at the molecular level** when interrogated at a defined time point.
   - cell cycle, age, exposure to environmental stimuli/stress, metabolic state





Why is bulk RNA-seq not enough?

The average behavior measured in millions of cells does not necessarily reflect the behavior in individual cells

In theory, we should therefore apply single-cell approaches to **all** studies of cells because **transcription** is, fundamentally, a **stochastic** process and mammalian cells are known to have non-continuous, **bursting** transcription, which inherently leads to variable cellular states.

Why is bulk RNA-seq not enough?

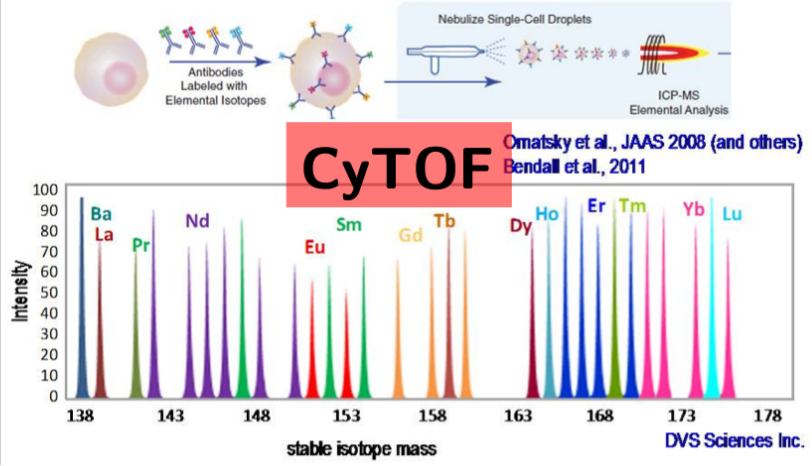
In practice, most scRNA-seq studies published to date deal with the higher-level complexities of organs and tissues:

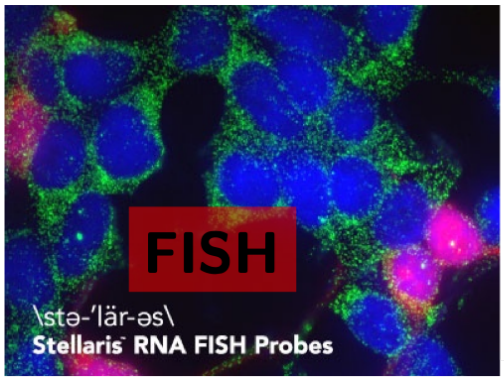
- characterizing **developmental** processes
  - traditionally hampered by extremely low cell numbers
- cell type catalogues of **entire organs** or very **heterogeneous tissues**
  - pancreas, brain, liver, lung, retina
- **immune cell** studies
  - often coupled with single-cell clonotyping
  - helps distinguish numerous activation states of T/B cells
- **tumor** studies
  - so far, mostly distinguishing between malignant and physiological cells (e.g. infiltrating immune cells)

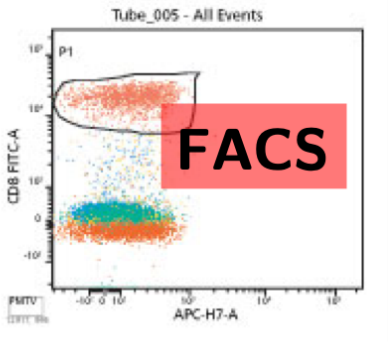
"Single-cell analyses are needed to fully understand the cellular specificity and complexity of tissue microenvironments."

"Traditional" single-cell methods

Microscopy and **cytometry** have been used for decades to understand properties of single cells. The major limitations have been **throughput** and the number of **features** that could be assessed simultaneously.







	FACS	CyTOF	qPCR
Cell capture method	Laser	Mass cytometry	Micropipettes
Number of cells per experiment	Millions	Millions	300–1,000
Cost	$0.05 per cell	$35 per cell	$1 per cell
Sensitivity	Up to 17 markers	Up to 40 markers	10–30 genes per cell

Papalexi & Satija (2018) doi: 10.1038/nri.2017.76

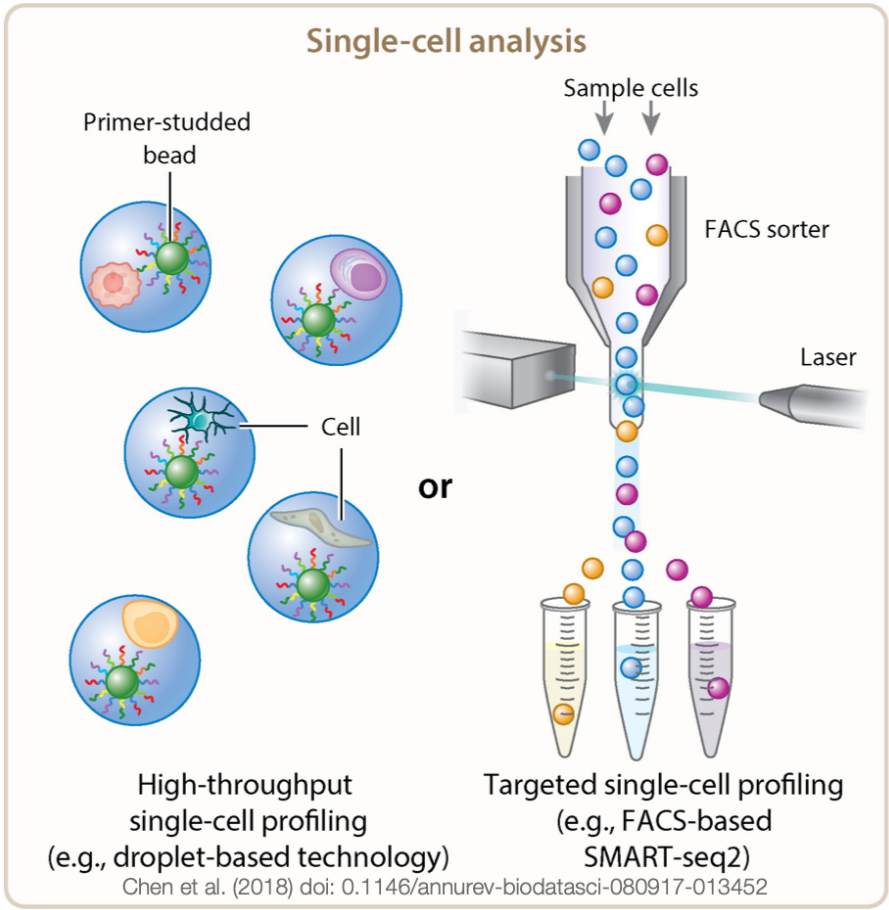
How to sequence the transcriptome of single cells?

From bulk to single cell transcriptomes

The main **challenges**:

- automated **cell isolation**
  - FACS vs. microfluidics
- untargeted whole transcriptome **amplification**
  - required input: 0.1–1 µg total RNA
  - [RNA] per cell: 0.1–50 pg (!)
- **parallel processing**
  - individual cell lysis & RT carried out in wells (<100 cells), microchambers (Fluidigm chip), nanochambers, or droplets (>10,000s cells)

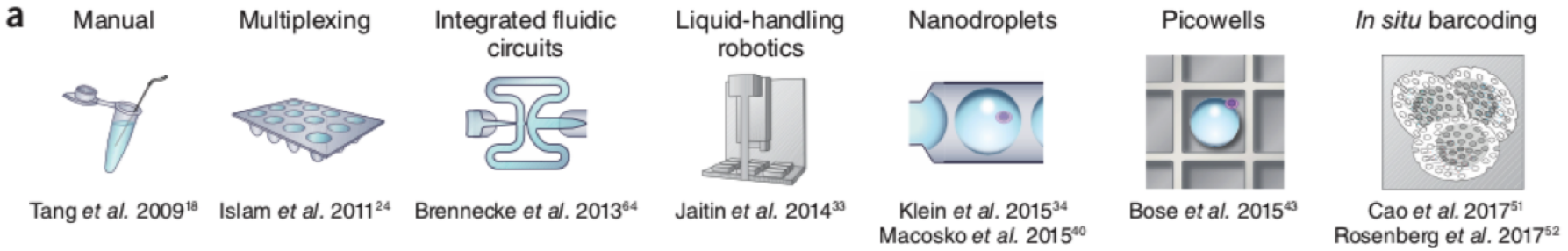
Details: Saliba et al. [2014] & Chen et al. [2018].

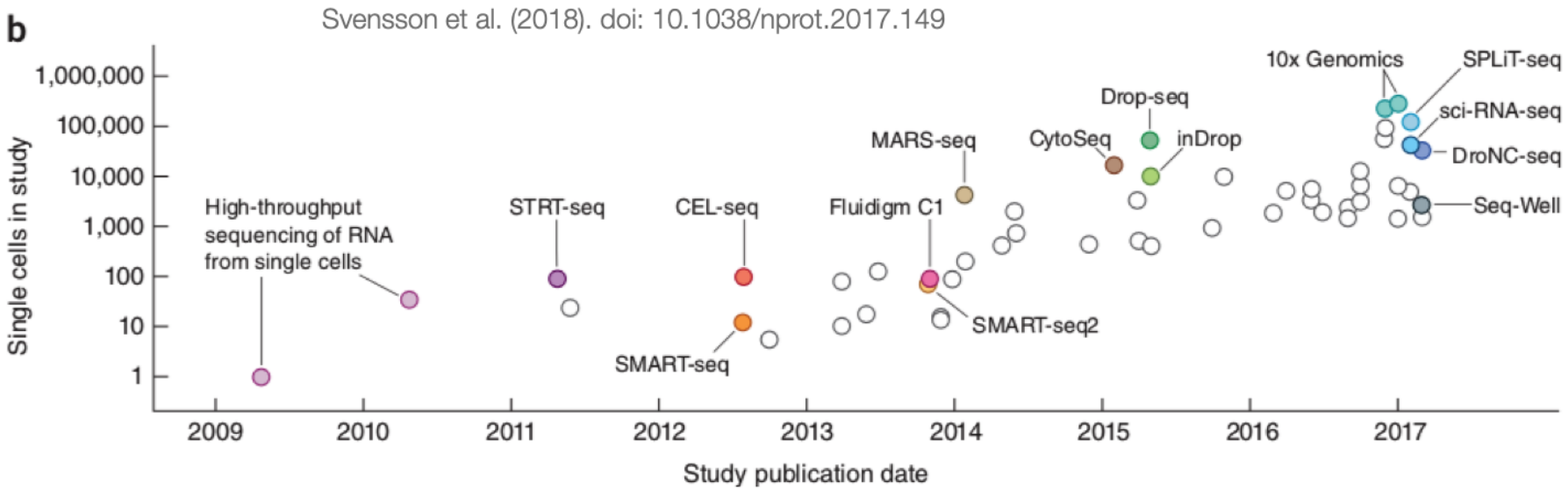


Major **innovations**:

- microfluidics
- random cell capture

Numerous solutions have been proposed in the past decade

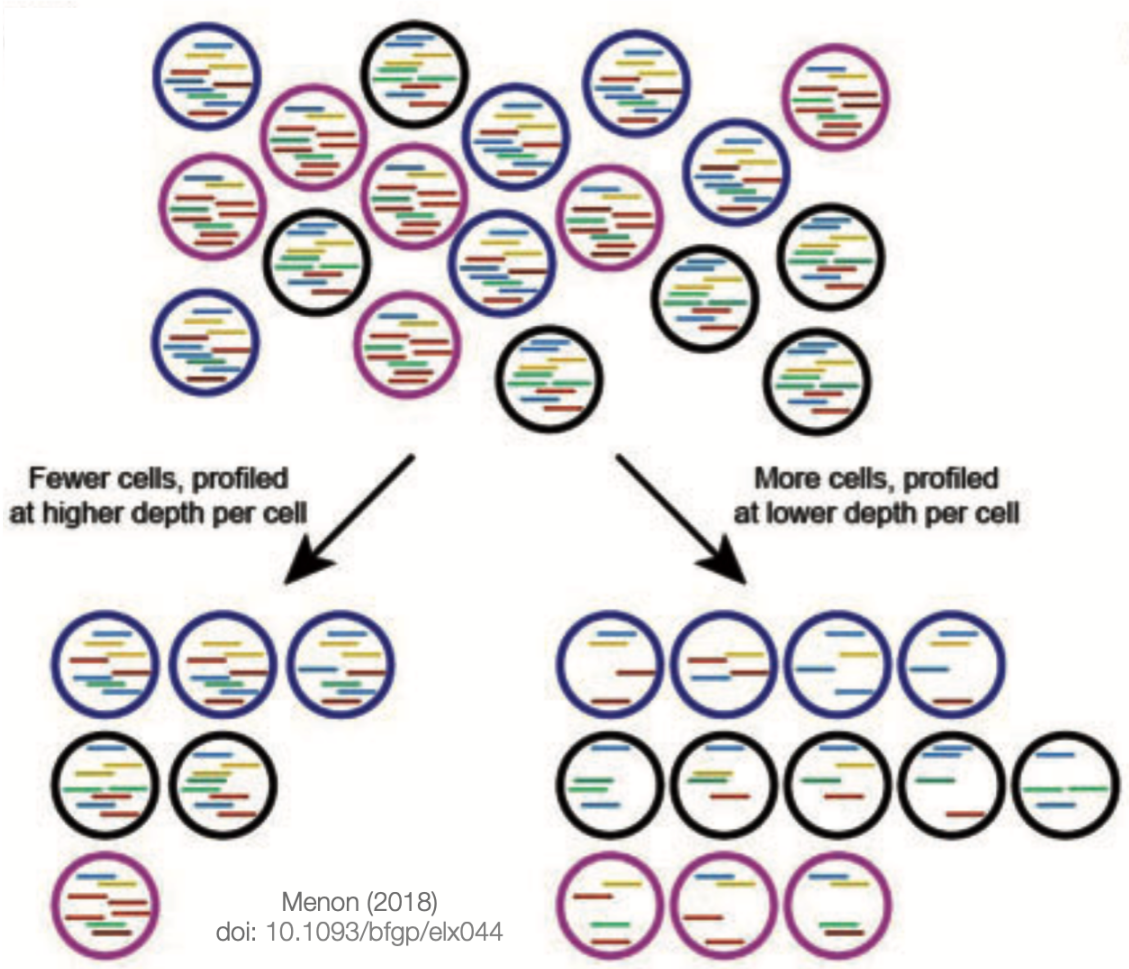




100s cells thanks to **multiplexing**, ca. 1,000 cells thanks to **fluidics**, 10,000s cells thanks to random cell captures techniques with **nanodroplets** and picowells, 100K cells thanks to ***in situ* barcoding**

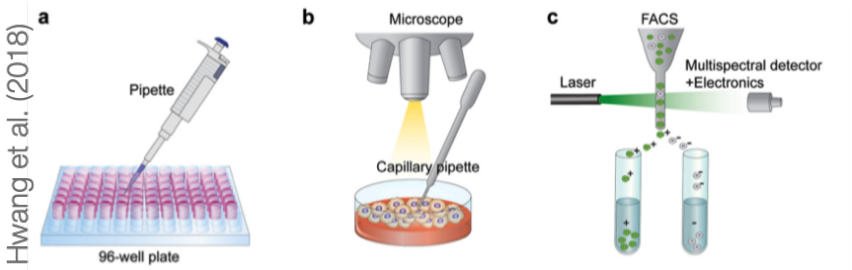
Sensitivity vs. quantity

Given a fixed population of cells and a total number of reads available, reads can either be used to sequence **fewer cells more deeply** or to sequence **more cells at a shallower depth**.



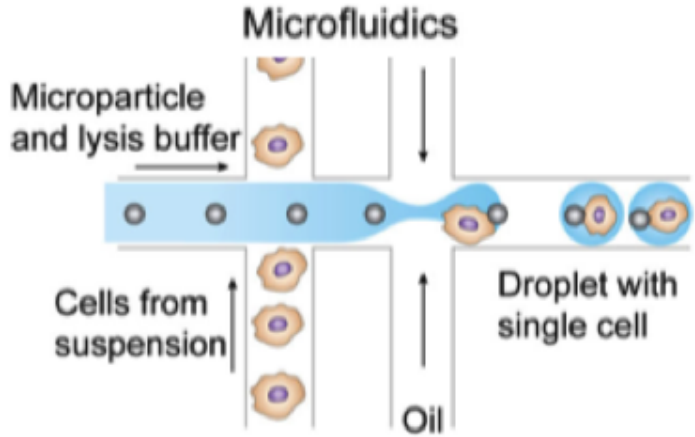
Sensitivity vs. quantity

"manual" cell isolation
e.g. SMART-seq, CEL-seq2
low-throughput



- labor intensive and costly (every cell gets its own library prep!)
- 100s cells ⇒ 100K - 4mio reads per (!) cell
- Smart-seq allows for full-length transcripts
- CEL-seq2 enables great gene diversity and reliably picks up even weakly expressed genes [Mereu et al., 2019]

Droplet-based cell isolation
e.g. inDrop, 10X Chromium
high-throughput



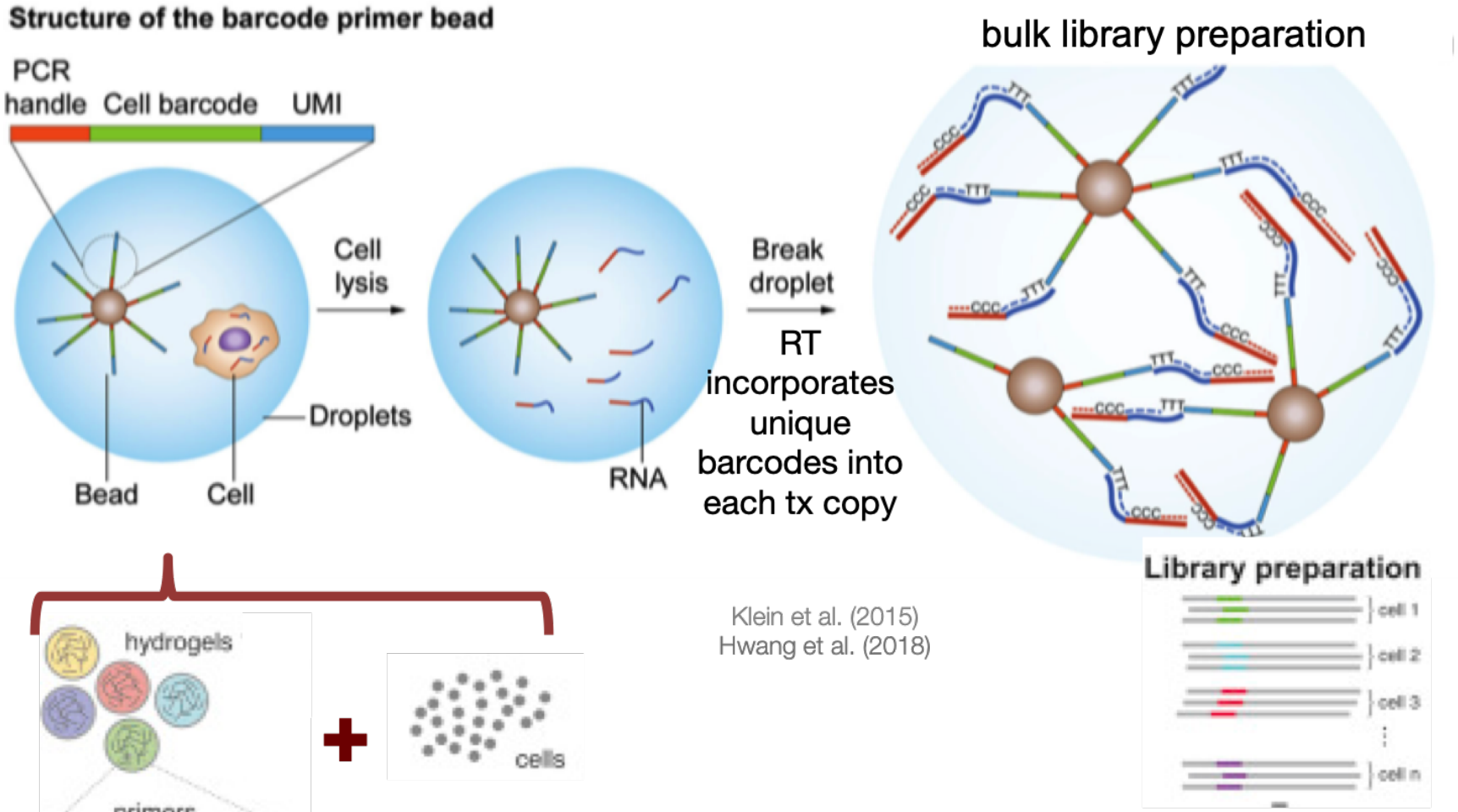
- can be automated
- 1,000s-10,000s of cells ⇒ 20K - 200K reads per cell
- usually 3' end counting only
- strand information is preserved

The most popular scRNA-seq methods

	SMART-seq2	CEL-seq2	STRT-seq	Quartz-seq2	MARS-seq	Drop-seq	inDrop	Chromium	Seq-Well	sci-RNA-seq	SPLiT-seq
Single-cell isolation	FACS, microfluidics	FACS, microfluidics	FACS, microfluidics, nanowells	FACS	FACS	Droplet	Droplet	Droplet	Nanowells	Not needed	Not needed
Second strand synthesis	TSO	RNase H and DNA pol I	TSO	PolyA tailing and primer ligation	RNase H and DNA pol I	TSO	RNase H and DNA pol I	TSO	TSO	RNase H and DNA pol I	TSO
Full-length cDNA synthesis?	Yes	No	Yes	Yes	No	Yes	No	Yes	Yes	No	Yes
Barcode addition	Library PCR with barcoded primers	Barcoded RT primers	Barcoded TSOs	Barcoded RT primers	Barcoded RT primers	Barcoded RT primers	Barcoded RT primers	Barcoded RT primers	Barcoded RT primers	Barcoded RT primers and library PCR with barcoded primers	Ligation of barcoded RT primers
Pooling before library?	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Library amplification	PCR	In vitro transcription	PCR	PCR	In vitro transcription	PCR	In vitro transcription	PCR	PCR	PCR	PCR
Gene coverage	Full-length	3'	5'	3'	3'	3'	3'	3'	3'	3'	3'
Number of cells per assay											

Chen et al. (2018) doi: 0.1146/annurev-biodatasci-080917-013452

Droplet-based sequencing

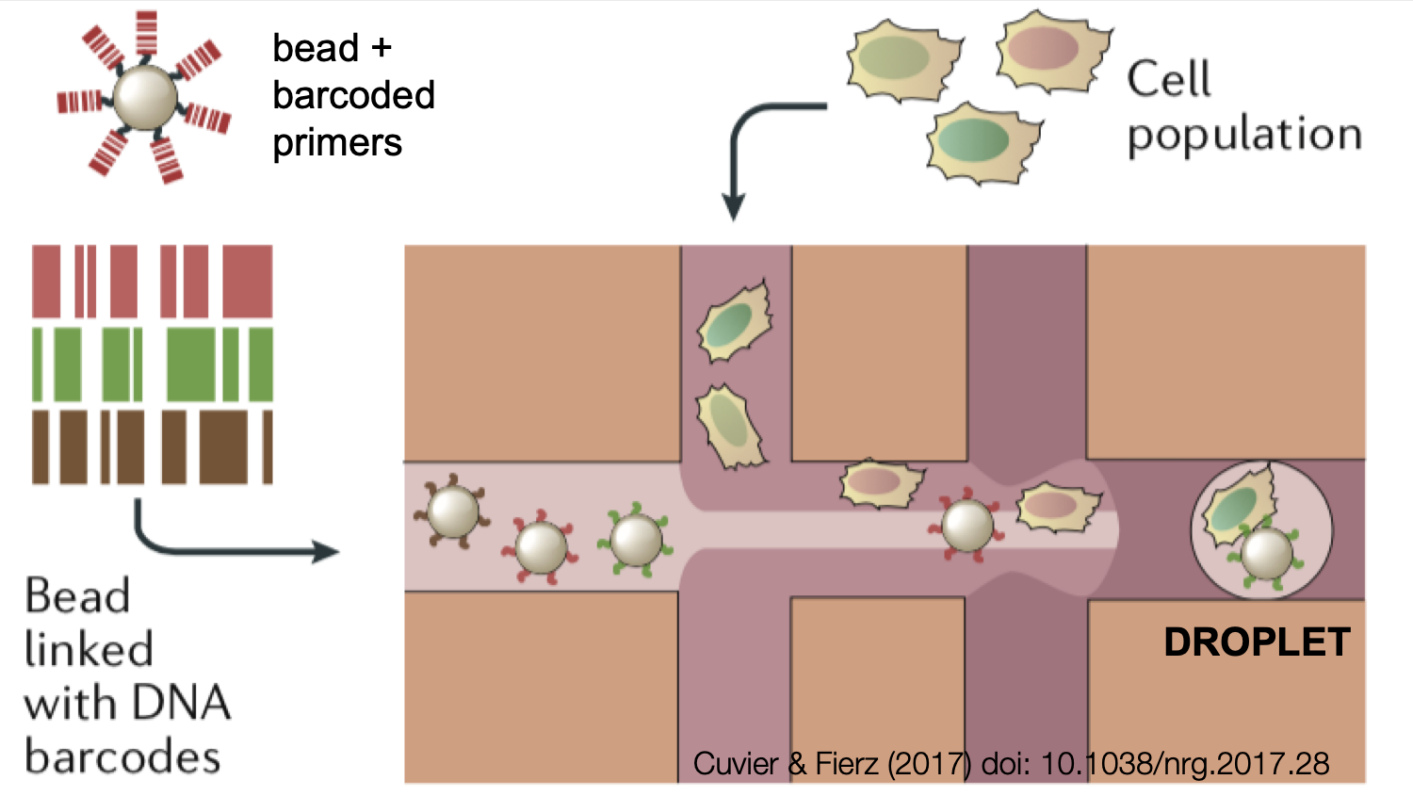


Klein et al. (2015)
Hwang et al. (2018)

Droplet-based sequencing

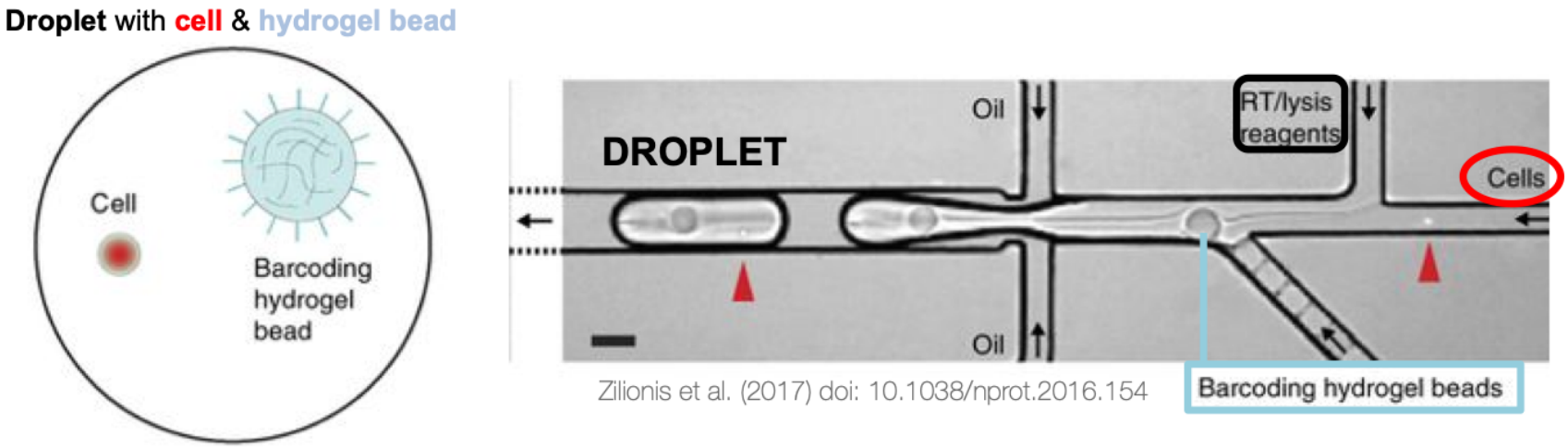
1. Droplet generation

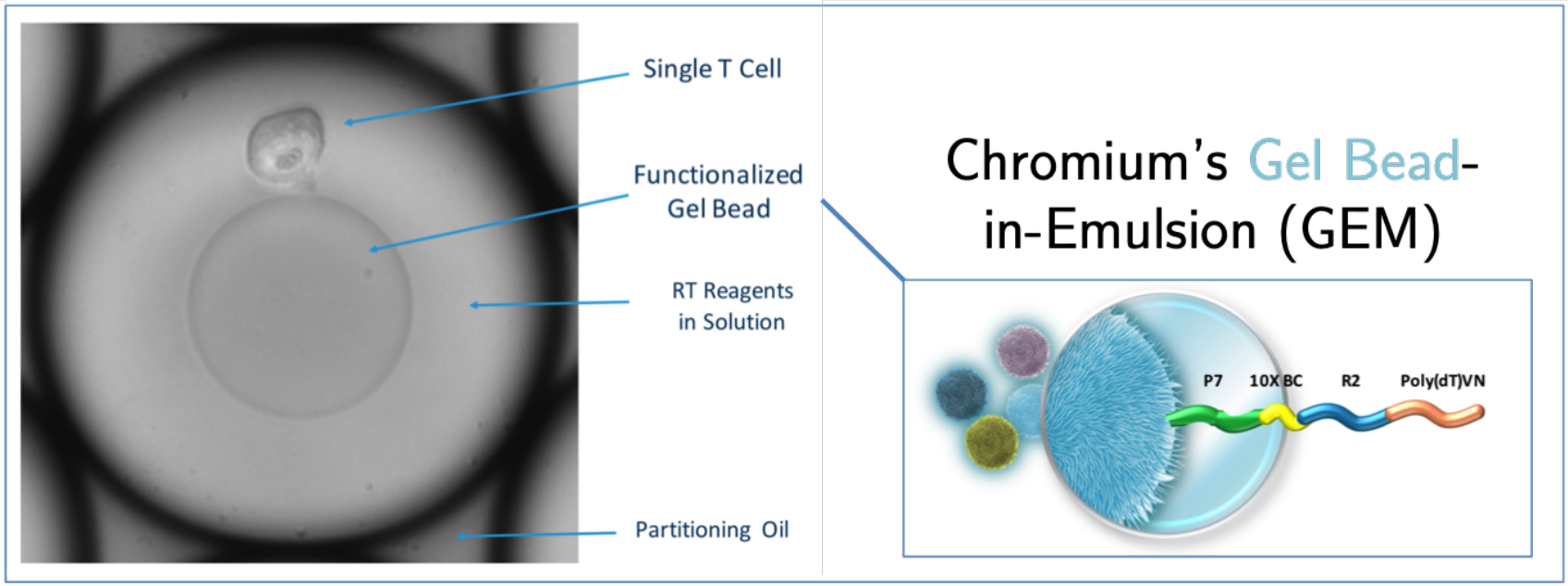
Using microfluidics, individual cells are captured together with a large set of (barcoded) poly(dT) primers (that are attached to hydrogel beads for the purpose of delivery).



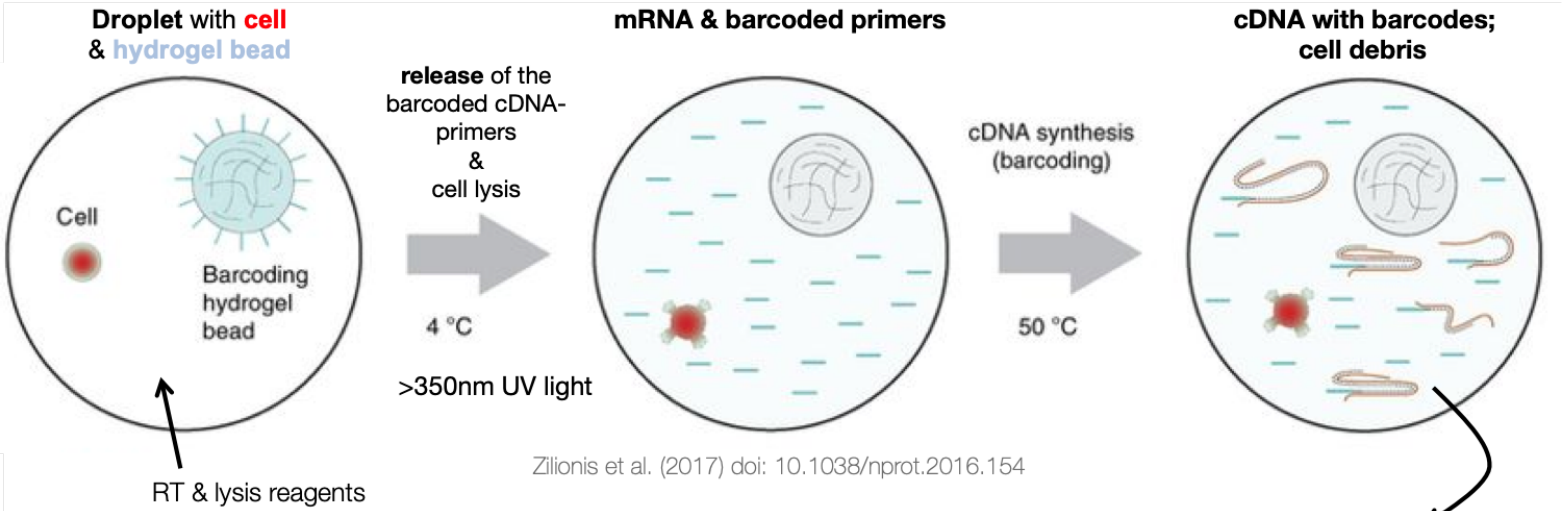
The final droplet contains **cell** + **primers** + **reagents** for cell lysis and RT.

Droplet-based sequencing: droplet content

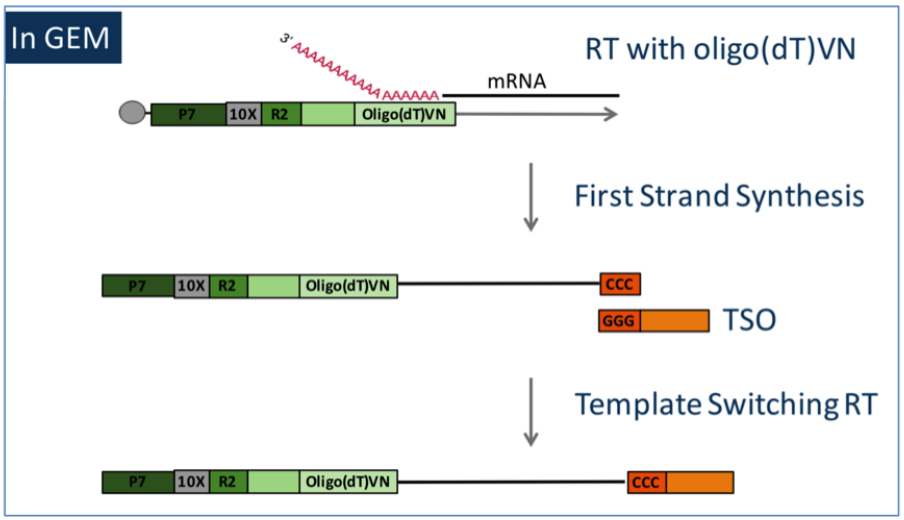




Droplet-based sequencing: capture & barcoding of mRNA transcripts



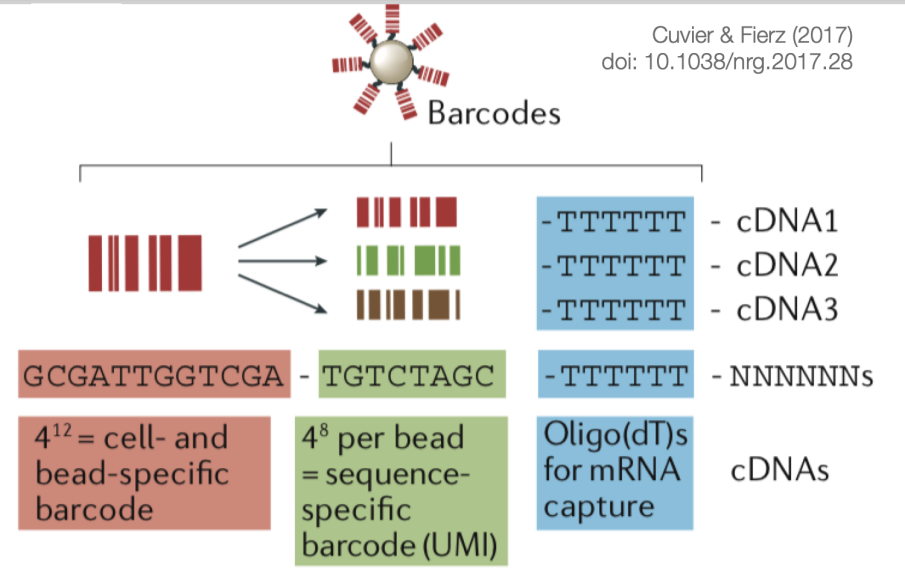
Zilionis et al. (2017) doi: 10.1038/nprot.2016.154



Droplet-based sequencing: Barcode details

ex.: inDrop (most similar to the commercial 10X Chromium)

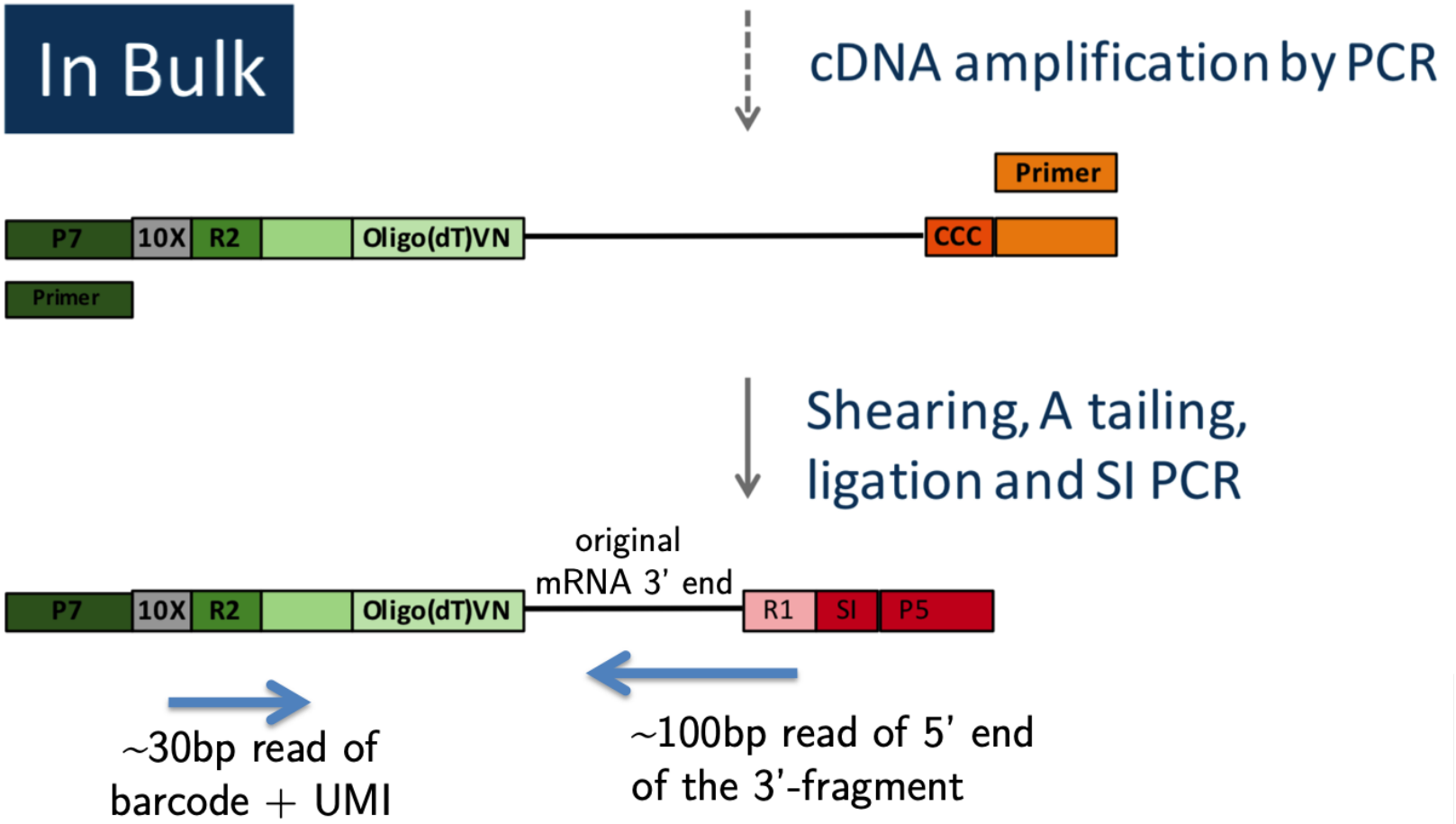
1. bead-specific **barcode** (⇒ cell)
2. primer-specific unique molecular identifier (**UMI**) (⇒ individual transcripts!)
3. (Illumina adapters)
4. **oligo(dT)** for poly(A)-mRNA capture



Barcode diversity can be increased through multiple rounds of oligo-additions (see [Zilionis et al., 2017] for details).

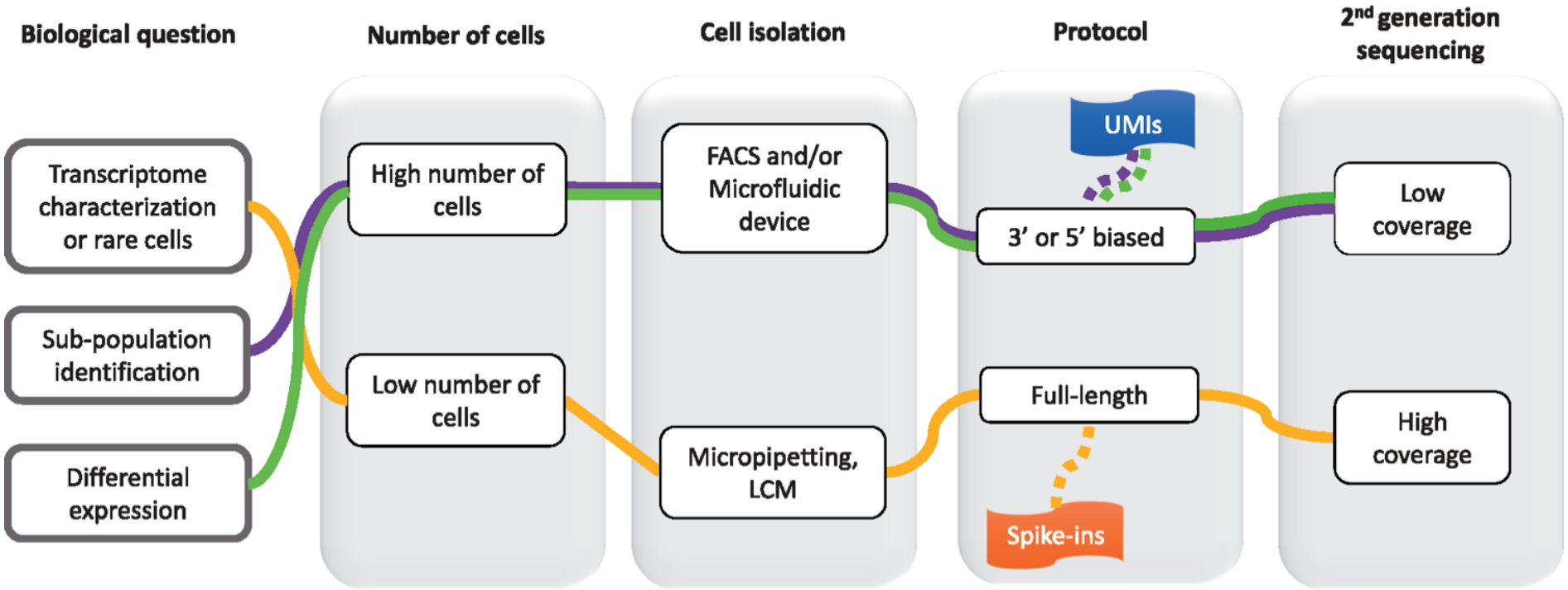
Droplet-based sequencing: Library preparation

getting the cDNA ready for Illumina-based sequencing



paired-end sequencing is a must, but the first read is typically much shorter since you do not want to run into the poly(A)-tail (why?)

How to choose between different scRNA-seq platforms



Dal Molin (2018). doi: 10.1093/bib/bby007

See Chen et al. [2018], Svensson et al. [2018], Ziegenhain et al. [2017], Zhang et al. [2019] for good overviews and reviews of different platforms.

All of these methods *dissociate* the tissues, i.e. spatial information is lost and mRNA levels may also reflect the stress induced by the protocol.

The ideal single-cell transcriptomics method

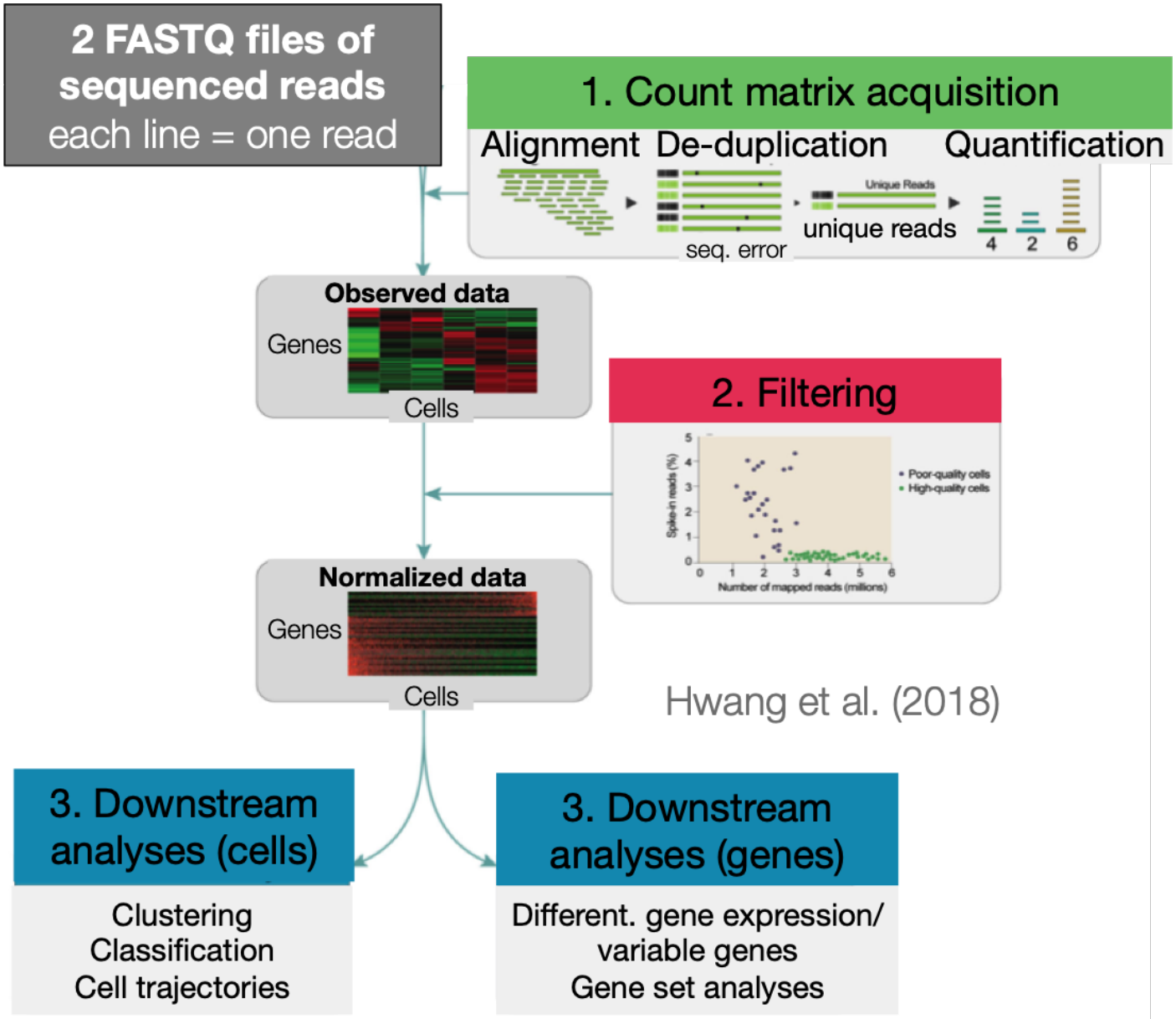
From Beltrame et al. [2019]:

Feature	Smart-Seq2	10X Chromium
Universal in terms of cell size, type and state.	not yet	not yet
In situ measurements.	not yet	not yet
No minimum input of number of cells to be assayed.	🙂	☹
Every cell is assayed, i.e. 100% capture rate.	🙂	😐
Every transcript in every cell is detected, i.e. 100% sensitivity.	☹	☹
Every transcript is identified by its full-length sequence.	🙂	☹
Transcripts are assigned correctly to cells, e.g. no doublets.	🙂	😐
Additional multimodal measurements.	not yet	**V(D)J; spatial info**
Cost effective per cell.	☹	🙂
Easy to use.	😐	😐
Open source.	🙂	☹

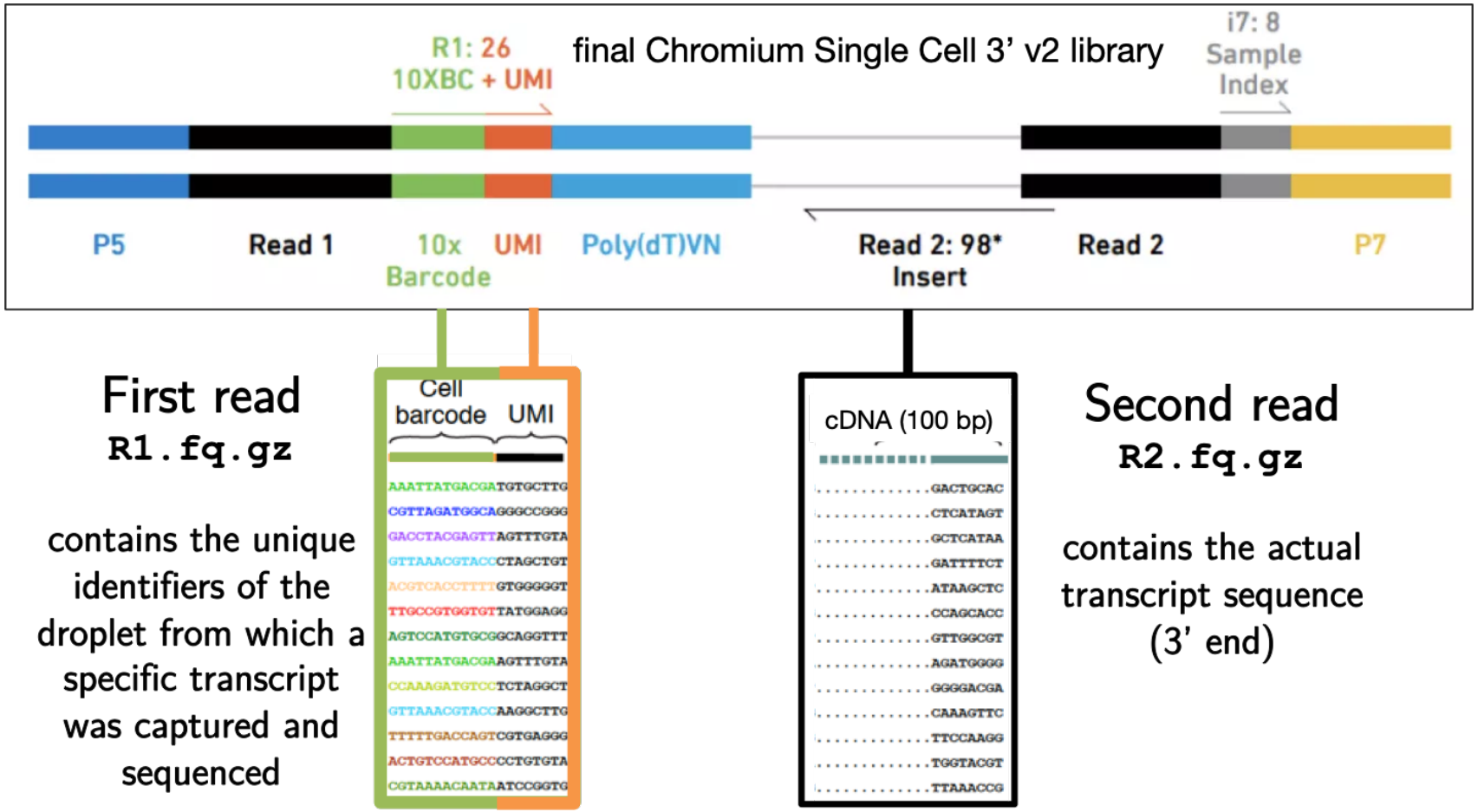
Obviously, the optimal solution does not exist. Pick the one that matches your needs most closely. See, e.g. Mereu et al. [2019] and Ding et al. [2019] for benchmarking studies of different platforms.

Processing of scRNA-seq data

Processing overview



1. Count matrix acquisition: 2 FASTQ files per sample



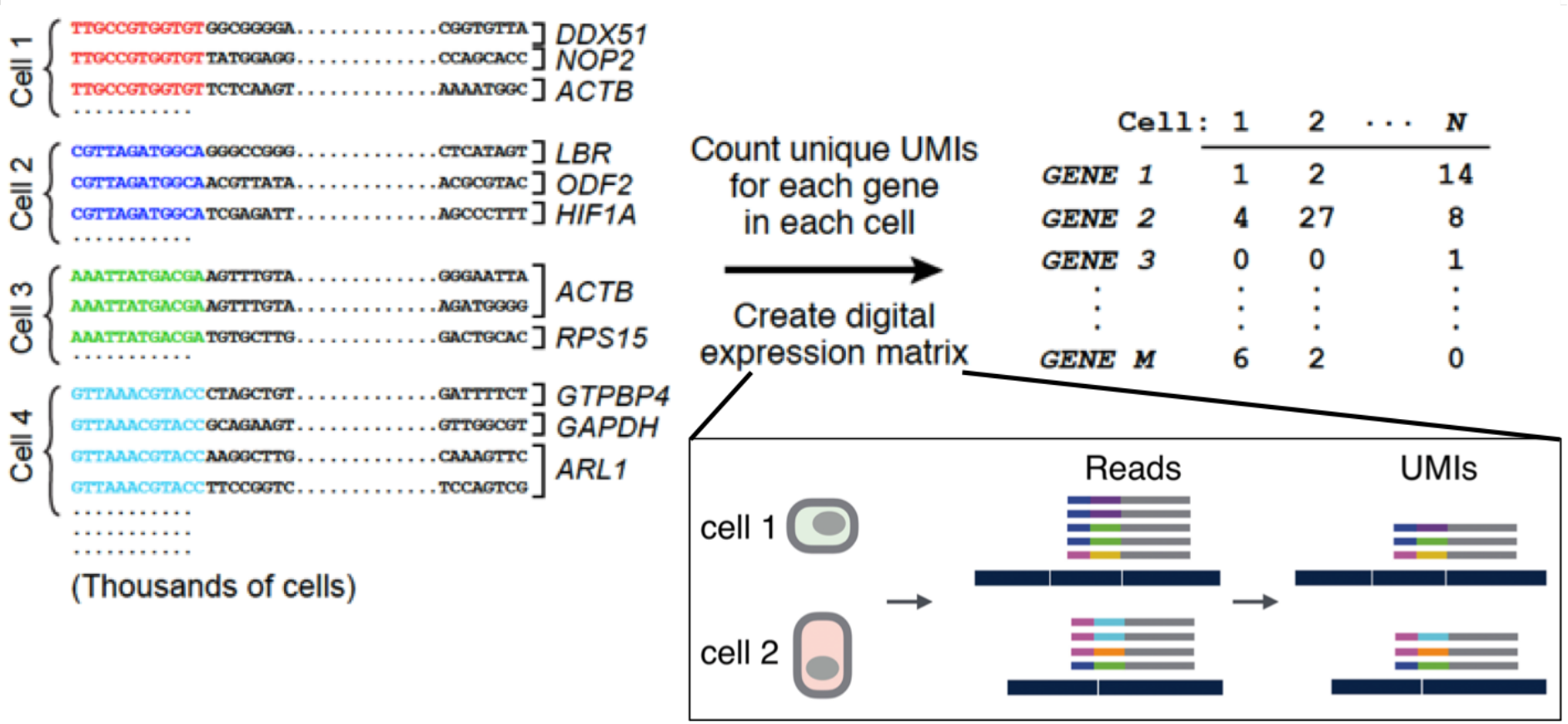
First read
`R1.fq.gz`

contains the unique identifiers of the droplet from which a specific transcript was captured and sequenced

Second read
`R2.fq.gz`

contains the actual transcript sequence (3' end)

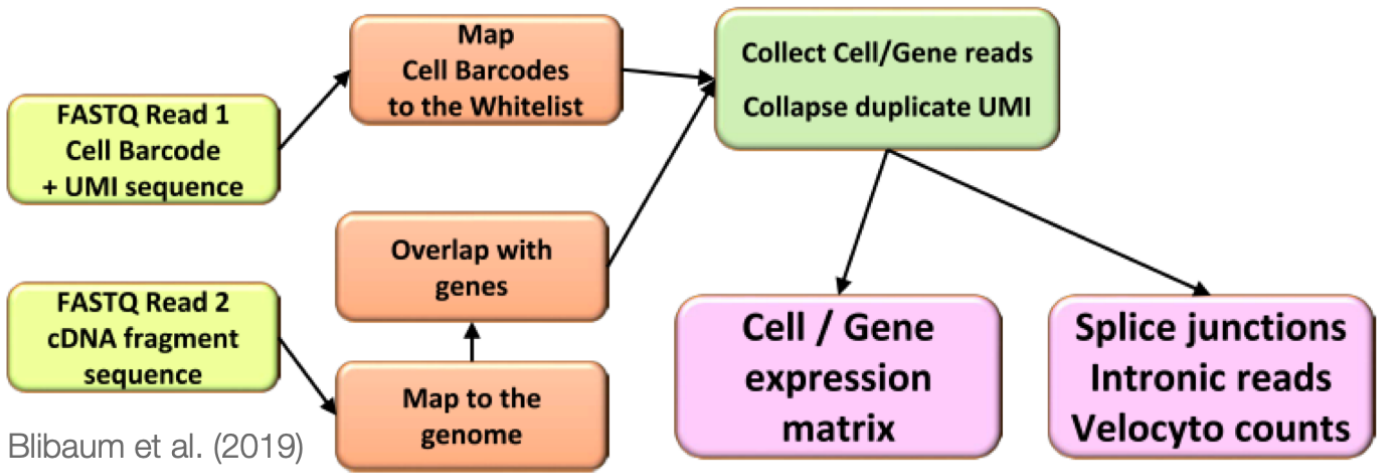
1. Count matrix acquisition



UMIs are tremendously helpful in being able to ignore amplification bias: only one UMI count is kept

1. Software for Count matrix acquisition

- **`STARsolo`** provides the same functionalities as the **`CellRanger`** pipeline from 10X Genomics, but allows for greater flexibility and speed [Blibaum et al., 2019]

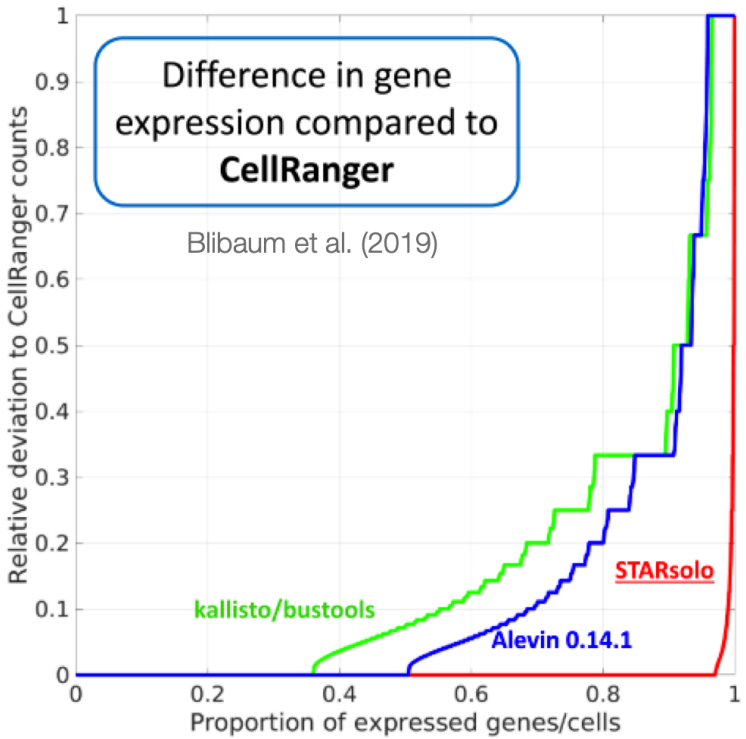


- python-based wrappers: **`scumi`** [Ding et al., 2019] ; R-wrappers: **`scPipe`** [Tian et al., 2018]
- pseudo-alignment based tools: **`alevin`** [Srivastava et al., 2019] or **`kallisto/bustools`** [Melsted et al., 2019]

1. Processing raw reads: software for count matrix generation

As always, your choices matter.

- `CellRanger` uses a custom-filtered subset of the GENCODE annotation
- most applications will only report reads overlapping with unique exons
- different tools handle ambiguous reads and to intron definitions differently



2. Quality controls of the count matrix

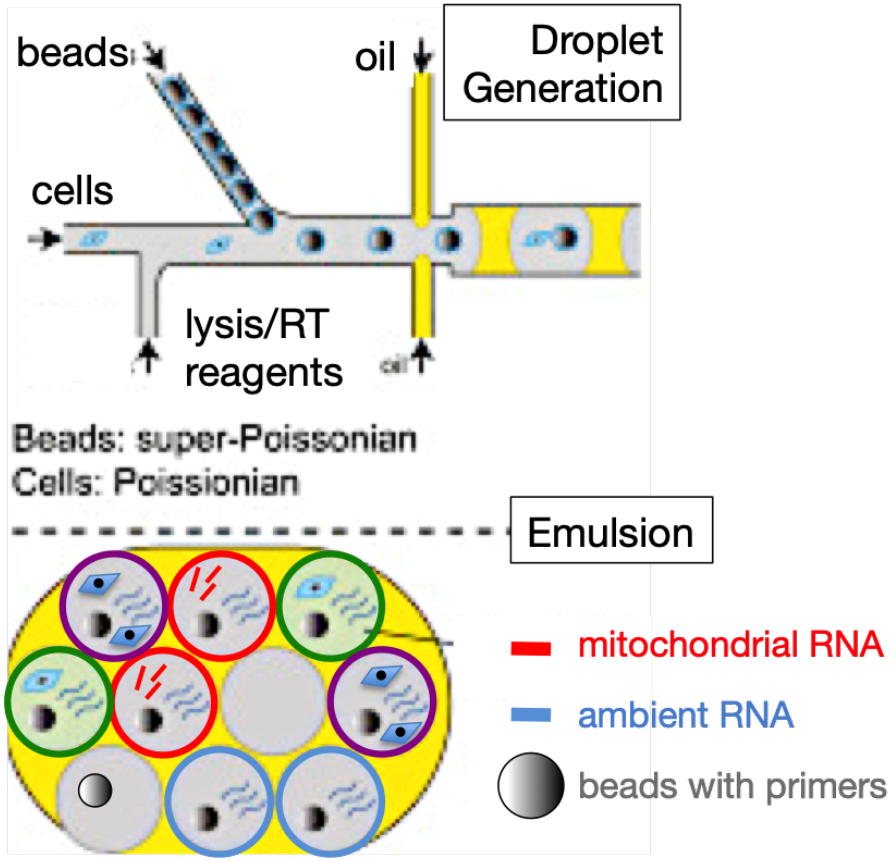
GOALS:

- get a feeling for how well the experiment worked
  - how many cells were captured?
  - how deeply was each cell sequenced? (= cell-specific library sizes)
  - how many individual transcripts were captured per cell?
- identify columns that contain the transcriptomes of real, single cells
- identify genes that may reflect contaminants (e.g. they are unexpectedly present in all cells)

2. Quality controls of the count matrix: **Cells**

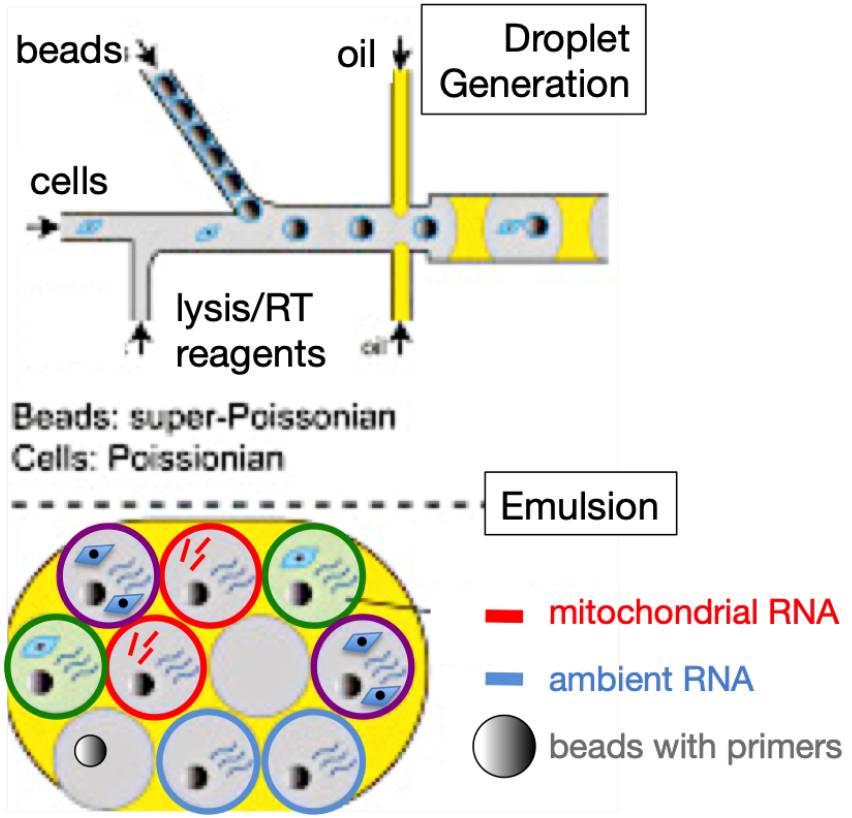
In droplet-based sequencing, many of the issues that we look for in the **cell-based QC** are related to the fact that **we cannot be sure how many cells a droplet contained before library preparation**.

The less healthy and separatable the cells were, the worse these issues will get.

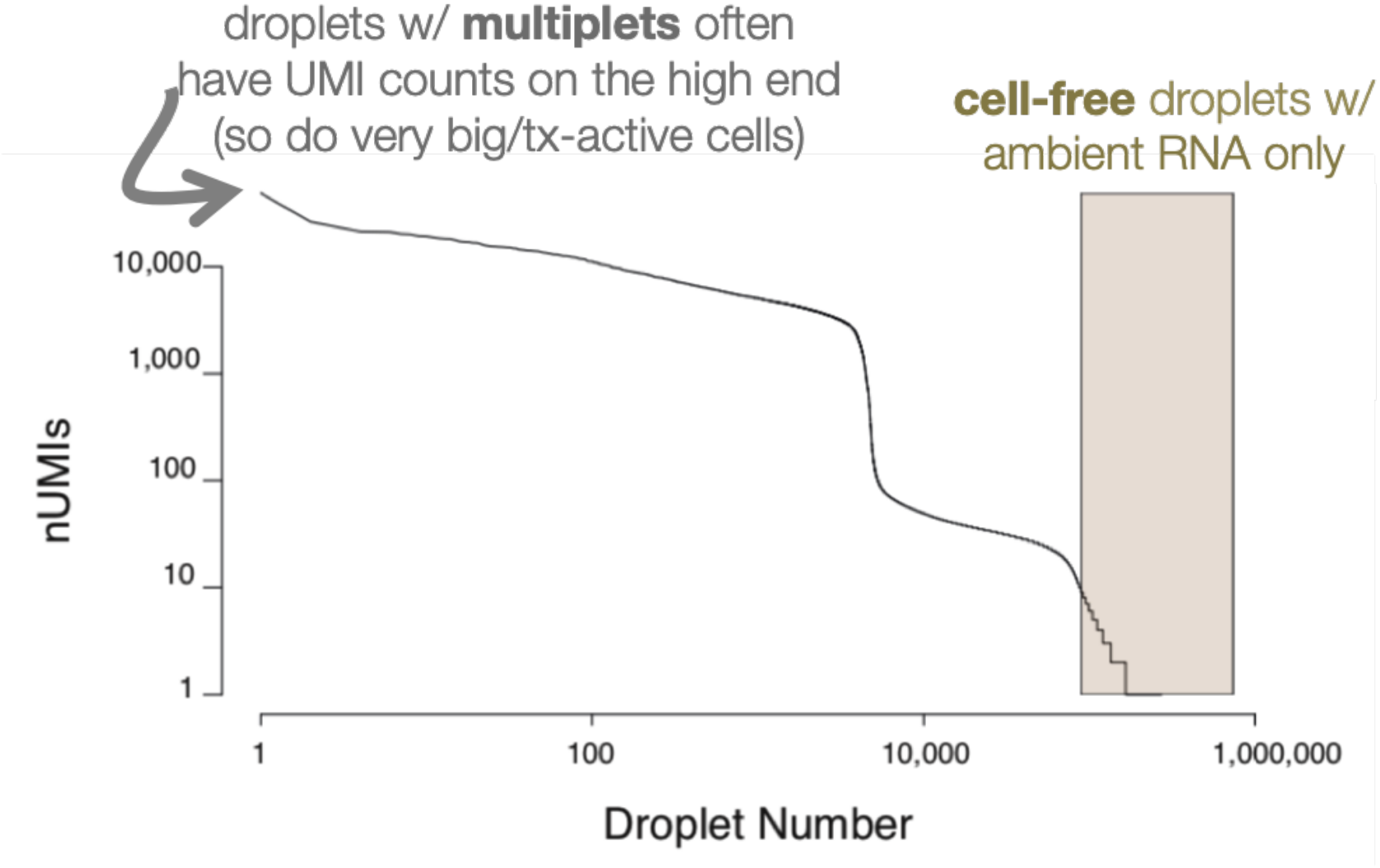


2. Quality controls of the count matrix: **Cells**

# cells	Observation & Consequences
Zero	should have low UMI numbers representing **ambient** RNA
1 intact cell	optimal outcome; number of transcripts should be a function of the overall abundance of transcripts of the original cell
1 dying cell	apoptosis ⇒ membrane permeabilization & mRNA degradation (⇒ cytoplasmic mRNA loss & overabundance of RNA protected within **mitochondria**
Multiple cells	resulting transcriptome for a single barcode will be a random sample from **all** the cells (usually around 5% of the droplets!)



2. Quality controls of the count matrix: Deciding which droplets represent the cells of interest



2. Quality controls of the count matrix: `CellRanger`'s stanard output

Estimated Number of Cells
11,769

Mean Reads per Cell	Median Genes per Cell
54,286	1,906

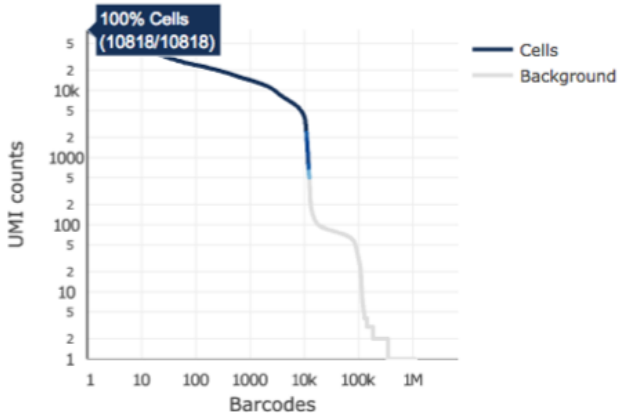
Sequencing

Number of Reads	638,901,019
Valid Barcodes	97.4%
Sequencing Saturation	68.2%
Q30 Bases in Barcode	93.7%
Q30 Bases in RNA Read	90.1%
Q30 Bases in Sample Index	90.1%
Q30 Bases in UMI	92.4%

Mapping

Reads Mapped to Genome	95.5%
Reads Mapped Confidently to Genome	92.5%
Reads Mapped Confidently to Intergenic Regions	5.0%
Reads Mapped Confidently to Intronic Regions	34.7%
Reads Mapped Confidently to Exonic Regions	52.7%
Reads Mapped Confidently to Transcriptome	49.7%
Reads Mapped Antisense to Gene	1.3%

Cells



Estimated Number of Cells	11,769
Fraction Reads in Cells	95.1%
Mean Reads per Cell	54,286
Median Genes per Cell	1,906
Total Genes Detected	23,036
Median UMI Counts per Cell	6,521

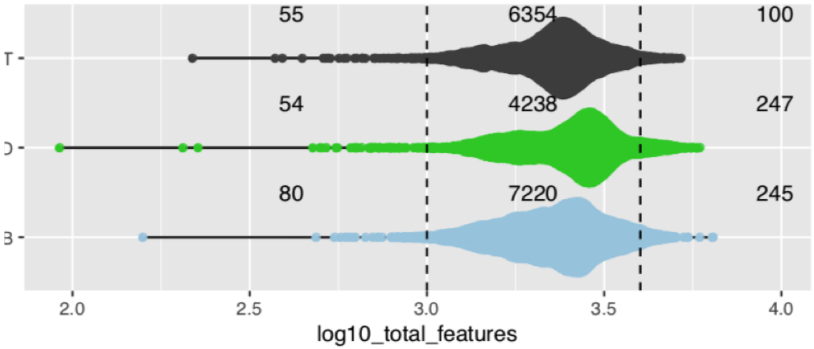
Sample

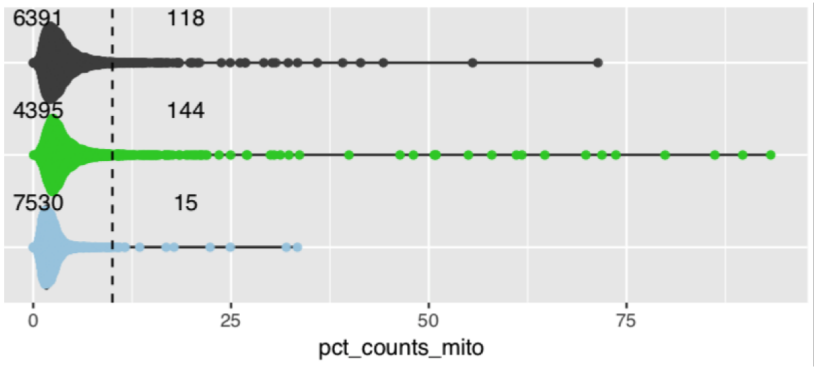
Name	pbmc_10k_v3
Description	Peripheral blood mononuclear cells (PBMCs) from a healthy donor
Transcriptome	GRCh38
Chemistry	Single Cell 3' v3
Cell Ranger Version	3.0.0

https://support.10xgenomics.com/img/single-cell-gex/web-summary-gex-3.0a.png

2. Quality controls of the count matrix: filtering cells

The exclusion of cells should always be done in conjunction with visual inspection of the diagnostic plots. Different sample types[3] will yield different distributions and will come with different expectations, too.





- remove cells with very **low UMI** counts
- remove cells with very **few genes**
- remove cells with very **high mitochrondrial** content
- last year: cells with very **high UMI** counts and genes were often removed because they were suspected to be multiplets – **more recent approach**: use an established package to flag potential doublets (e.g. `scds` [Bais and Kostka, 2019])

[3] uniformly or differently sized cells, metabolically active vs. quiescent etc.

2. Quality controls of the count matrix: example code

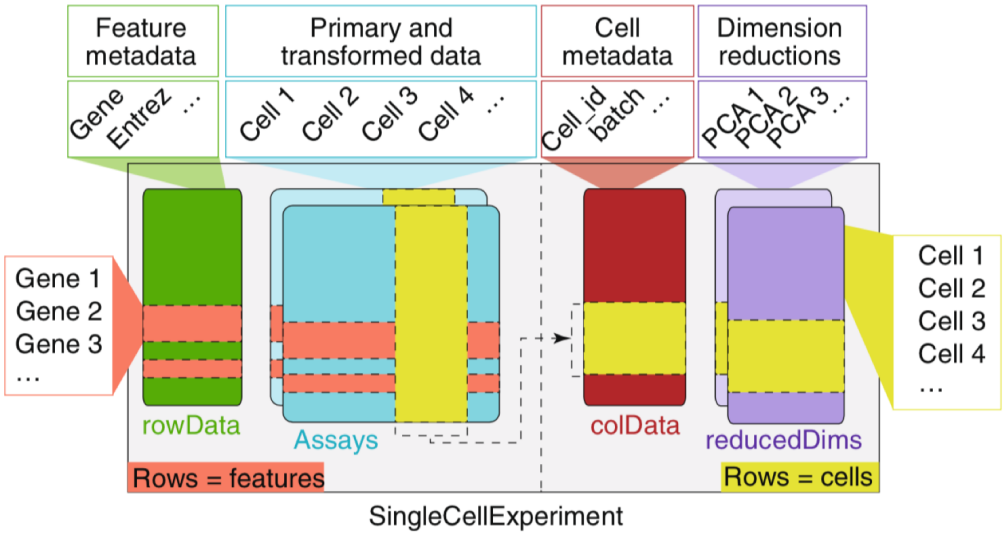
Taken from https://osca.bioconductor.org/quality-control.html.

```
## Load toy example
library(SingleCellExperiment); library(scater); library(AnnotationHub)
sce.416b <- scRNAseq::LunSpikeInData(which="416b")
sce.416b
```

```
## class: SingleCellExperiment
## dim: 46604 192
## metadata(0):
## assays(1): counts
## rownames(46604): ENSMUSG00000102693 ENSMUSG00000064842 ...
##   ENSMUSG00000095742 CBFB-MYH11-mcherry
## rowData names(1): Length
## colnames(192): SLX-9555.N701_S502.C89V9ANXX.s_1.r_1
##   SLX-9555.N701_S503.C89V9ANXX.s_1.r_1 ...
##   SLX-11312.N712_S508.H5H5YBBXX.s_8.r_1
##   SLX-11312.N712_S517.H5H5YBBXX.s_8.r_1
## colData names(9): Source Name cell line ... spike-in addition block
## reducedDimNames(0):
## spikeNames(0):
## altExpNames(2): ERCC SIRV
```

SingleCellExperiment object

- container derived from bioconductor's `SummarizedExperiment`
- **expression values** stored in `matrices` (rows = features/genes, columns = cells) ($\rightarrow$ `assay`)
- **metadata** about the features and cells stored in separate `DataFrames`
  - feature metadata: e.g. gene names, number of cells with non-zero expression, ... ($\rightarrow$ `rowData`)
  - cell metadata: e.g. sample, classification, # UMI, ... ($\rightarrow$ `colData`)
- **cell coordinates** obtained from dimensionality reductions ($\rightarrow$ `reducedDim`)



2. Quality controls of the count matrix: example code cont'd

```
## identifying the mitochondrial transcripts
ens.mm.v97 <- AnnotationHub()[["AH73905"]]
location <- mapIds(ens.mm.v97, keys=rownames(sce.416b),
  keytype="GENEID", column="SEQNAME")
```

```
## Warning: Unable to map 563 of 46604 requested IDs.
```

```
is.mito <- which(location=="MT")

## calculate QC metrics
qc.df <- scater::perCellQCMetrics(sce.416b, subsets=list(Mito=is.mito))
names(qc.df)
```

```
##  [1] "sum"                 "detected"            "percent_top_50"
##  [4] "percent_top_100"     "percent_top_200"     "percent_top_500"
##  [7] "subsets_Mito_sum"    "subsets_Mito_detected" "subsets_Mito_percent"
## [10] "altexps_ERCC_sum"    "altexps_ERCC_detected" "altexps_ERCC_percent"
## [13] "altexps_SIRV_sum"    "altexps_SIRV_detected" "altexps_SIRV_percent"
## [16] "total"
```

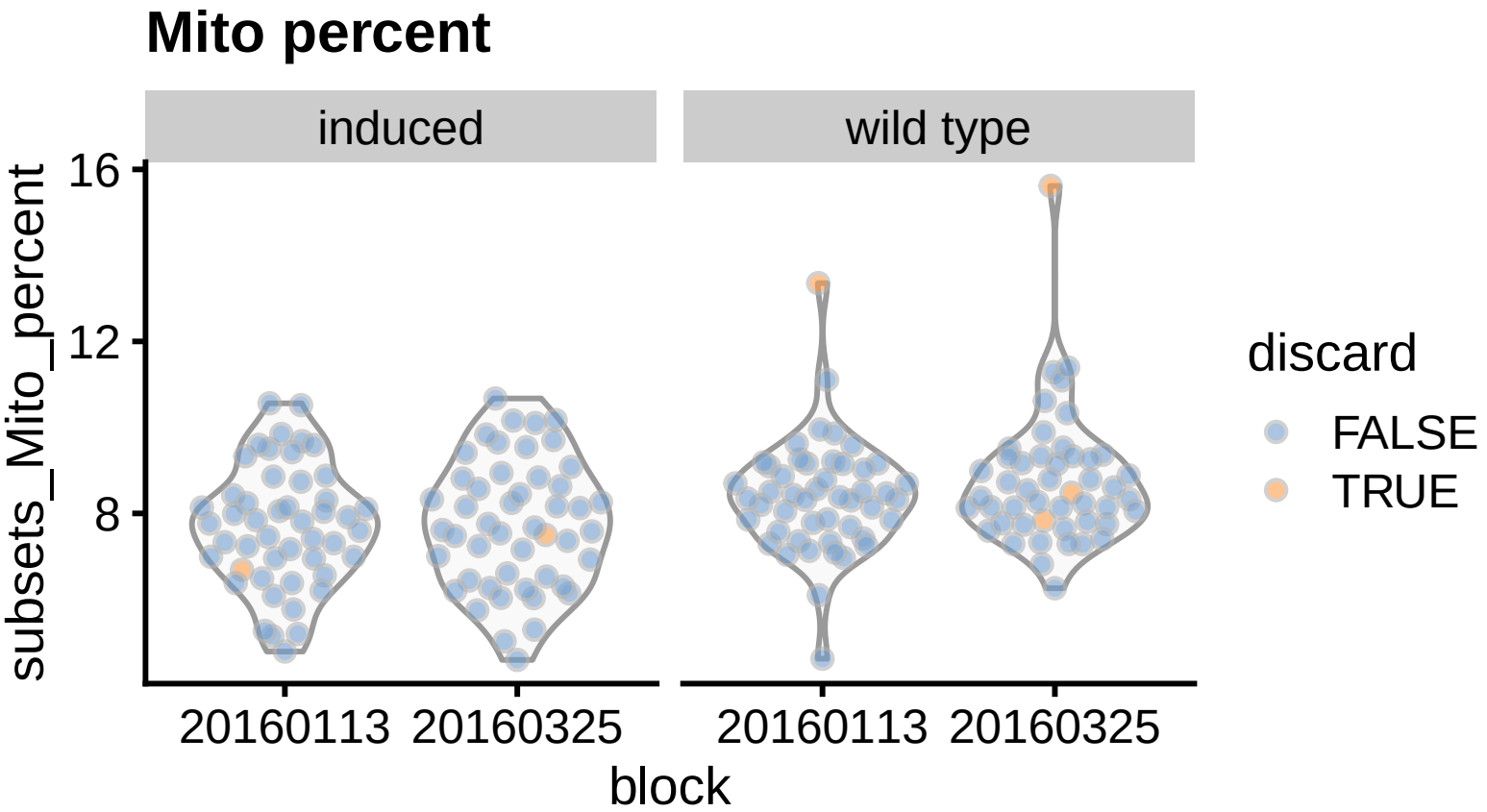
2. Quality controls of the count matrix: example code cont'd (2)

scater metric	Meaning
`sum`	sum of counts for each cell (= **library sizes**)
`detected`	number of features above `detection.limit` (default: 0 → number of **genes with non-zero expression** per cell)

```
## determine outliers for each metric
reasons <- quickPerCellQC(qc.df, percent_subsets=c("subsets_Mito_percent",
    "altexps_ERCC_percent"))
sce.416b$discard <- reasons$discard

## add QC results to colData of the SCE
colData(sce.416b) <- cbind(colData(sce.416b), qc.df)
sce.416b$block <- factor(sce.416b$block)
sce.416b$phenotype <- ifelse(grepl("induced", sce.416b$phenotype),
  "induced", "wild type")
```

2. Quality controls of the count matrix: example code cont'd (3)

```
## make plot
plotColData(sce.416b, x="block", y="subsets_Mito_percent",  colour_by="discard",
  other_fields="phenotype") + facet_wrap(~phenotype) + ggtitle("Mito percent")
```

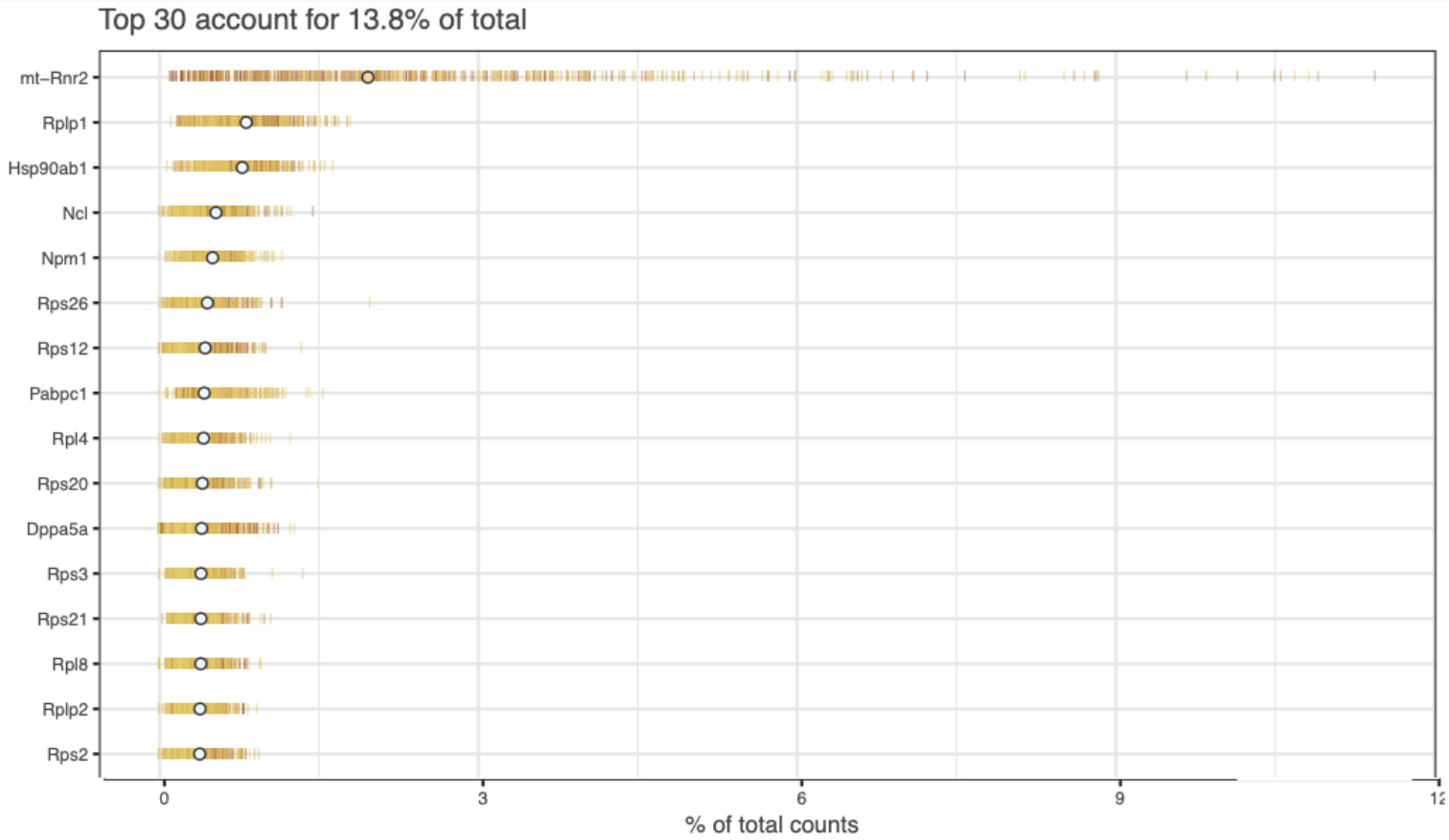


More plots & details:https://osca.bioconductor.org/quality-control.html#quality-control-plots

2. Quality controls of the count matrix: Assessing **genes**

The most strongly expressed genes should encompass ribosomal *proteins* and other housekeeping genes and ideally some of the typical marker genes known for your sample type.

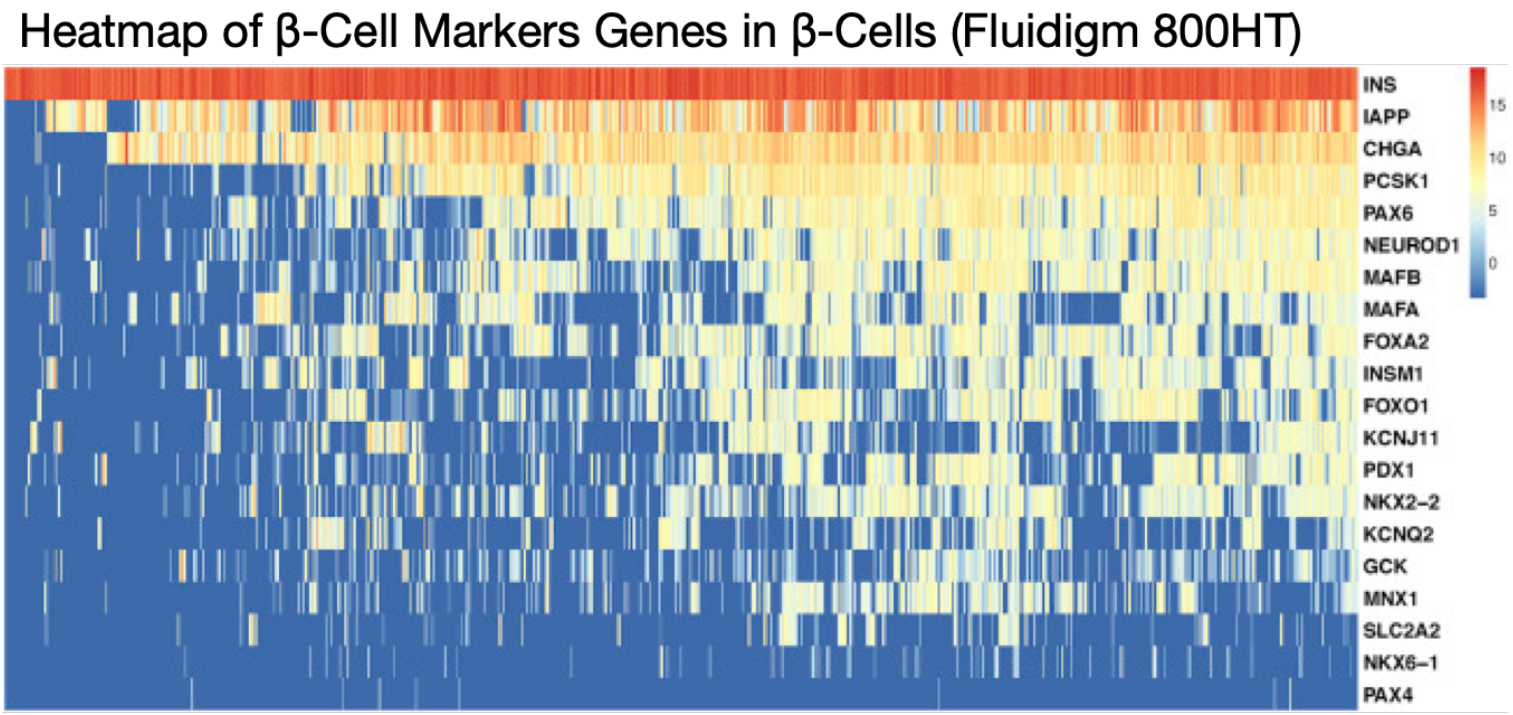
```
plotHighestExprs(example_sce, exprs_values = "counts")
```



2. Quality controls of the count matrix: Assessing **genes**

dropouts = undetected transcripts

- false negatives
- nearly impossible to distinguish from true negatives
- very common and not restricted to lowly expressed genes

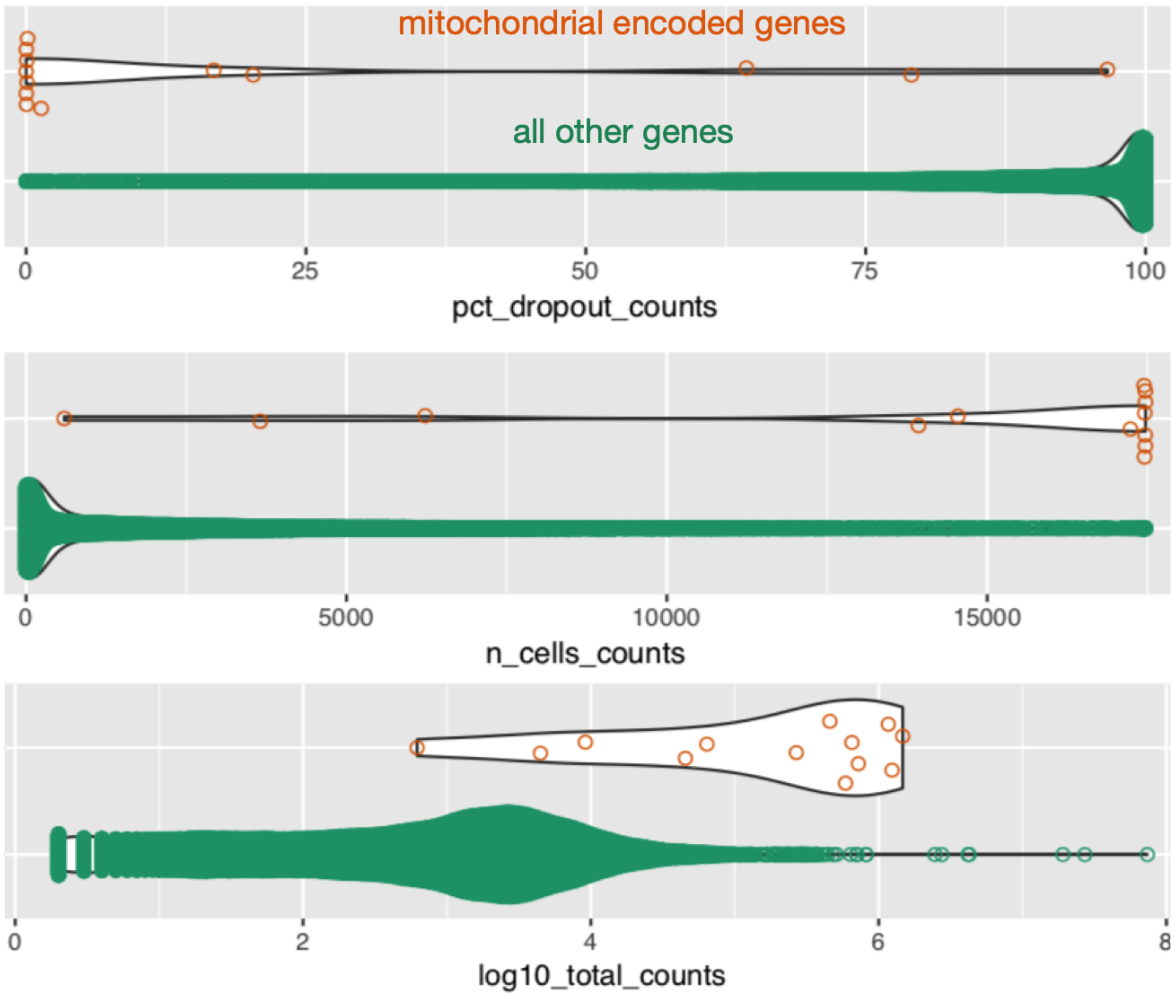


Wang & Kaestner (2018) doi: 10.1016/j.cmet.2018.11.016

All genes shown here are *known* to be expressed in pancreatic β cells.

2. Quality controls of the count matrix: Assessing **genes**

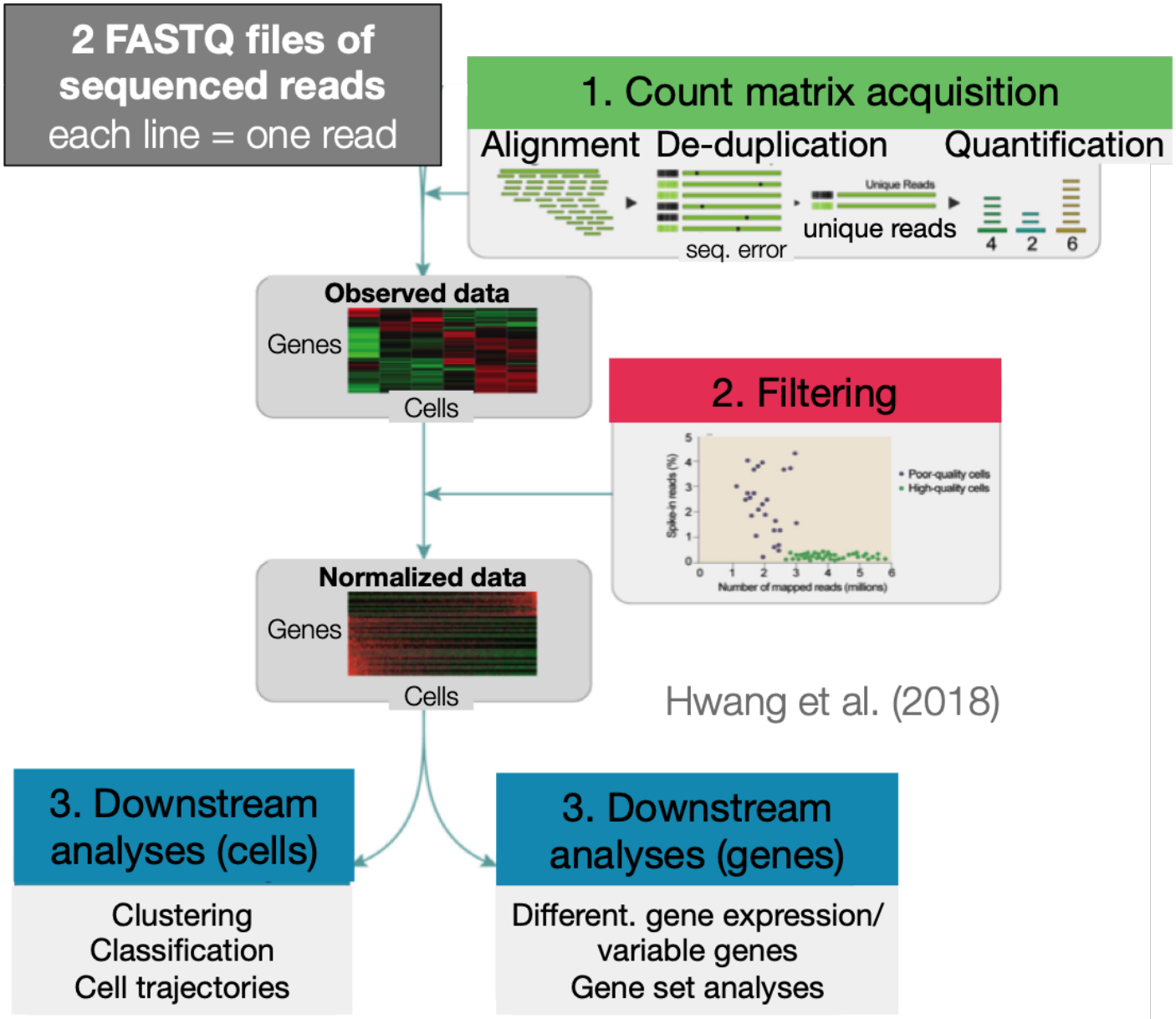
Gene dropouts are VERY COMMON and NOT restricted to lowly expressed genes!



Currently, scRNA-seq is **not a transcriptome-wide** method; it is a technique that will return a **sample** of a cell's transcriptome! [Andrews and Hemberg, 2018]

It is often benefitial to remove genes with **extremely low** capture rates because they can distort downstream analyses. Identify possibly **contaminating transcripts** (see `SoupX` [Young and Behjati, 2020] or `DecontX` [Yang et al., 2019]).

Processing overview



3. Normalization

. . . aims to reduce **systematic** differences in read counts.

Typical factors that influence downstream analyses are:

- **number of UMI/genes** within a cell – not just for technical reasons, this also correlates with cell size and general RNA content of a cell!
- biological factors: **cell cycle** status, cell size
- technical **batch effects** such as time of preparation, experimenter, sequencing lane/machine/day

Technical noise affecting the cell-wide profiles is difficult to estimate because every single cell (of every experiment) is considered a biological replicate.

For **biological confounders**, it's almost impossible to find a consensus of whether to ignore them or not.

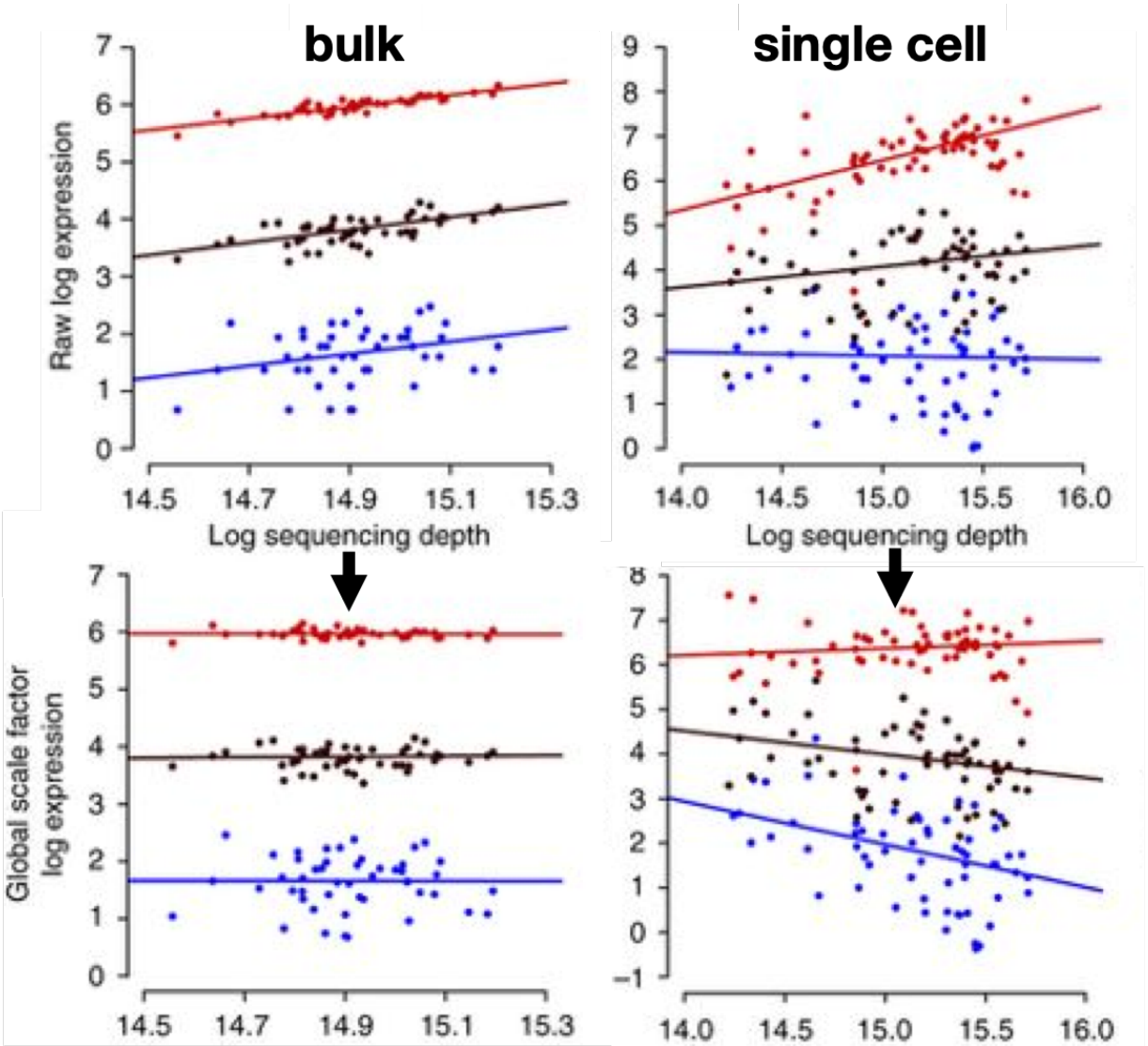
3. Normalization assumptions

The one factor everyone can agree on that definitely needs to be adjusted is the **difference in library sizes** for individual cells.

From Hafemeister and Satija [2019]:

1. Normalized expression level of a gene should not correlate with the total sequencing depth of a cell.
2. The variance of a normalized gene (across cells) should primarily reflect biological heterogeneity, independent of gene abundance or sequencing depth.

3. Normalization: effect of global scale factor

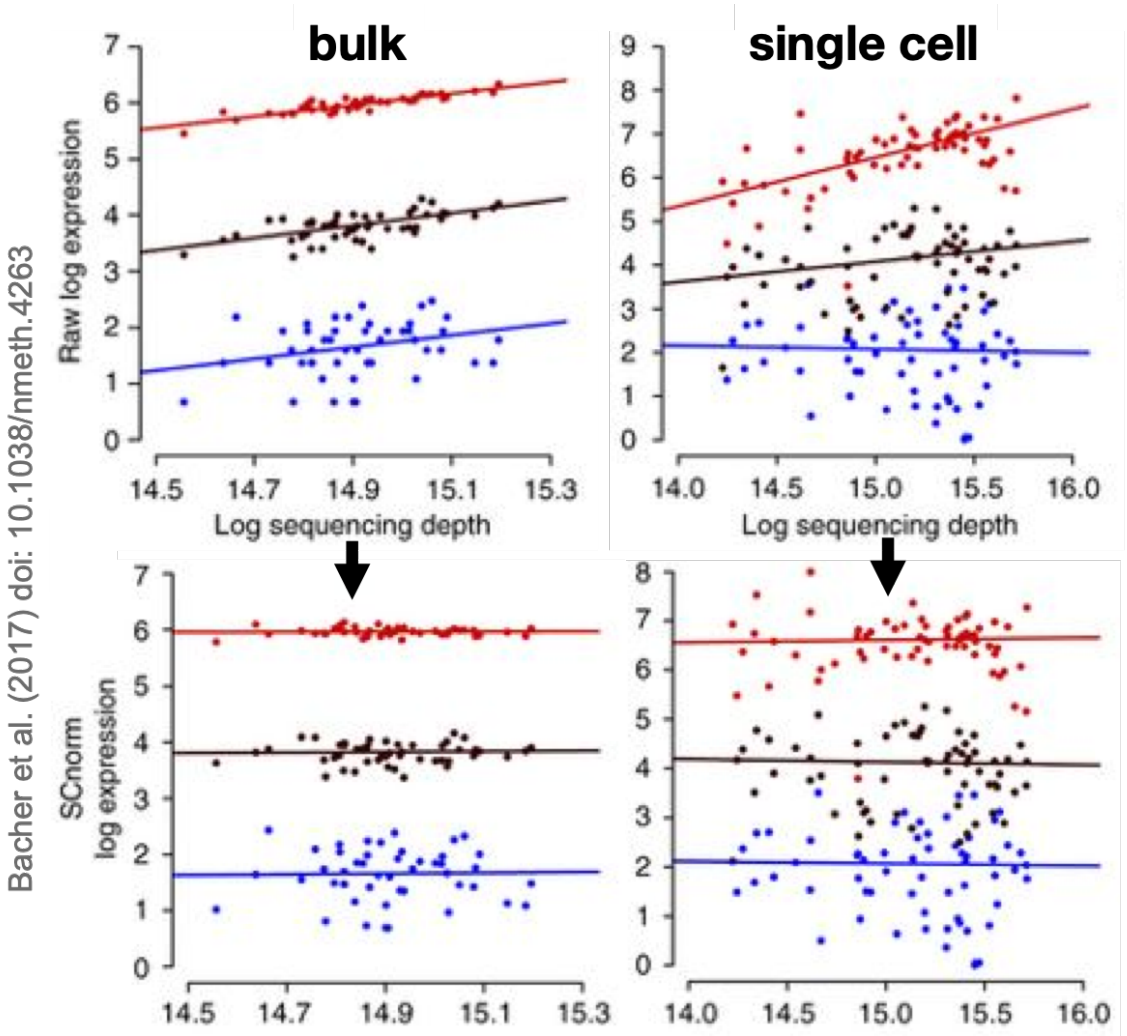


Bacher et al. (2017) doi: 10.1038/nmeth.4263

scRNA-seq shows systematic variation between transcript-specific expression & sequencing depth ("**count-depth relationship**")

Global scale factor works well for bulk RNA-seq, but less so for scRNA-seq

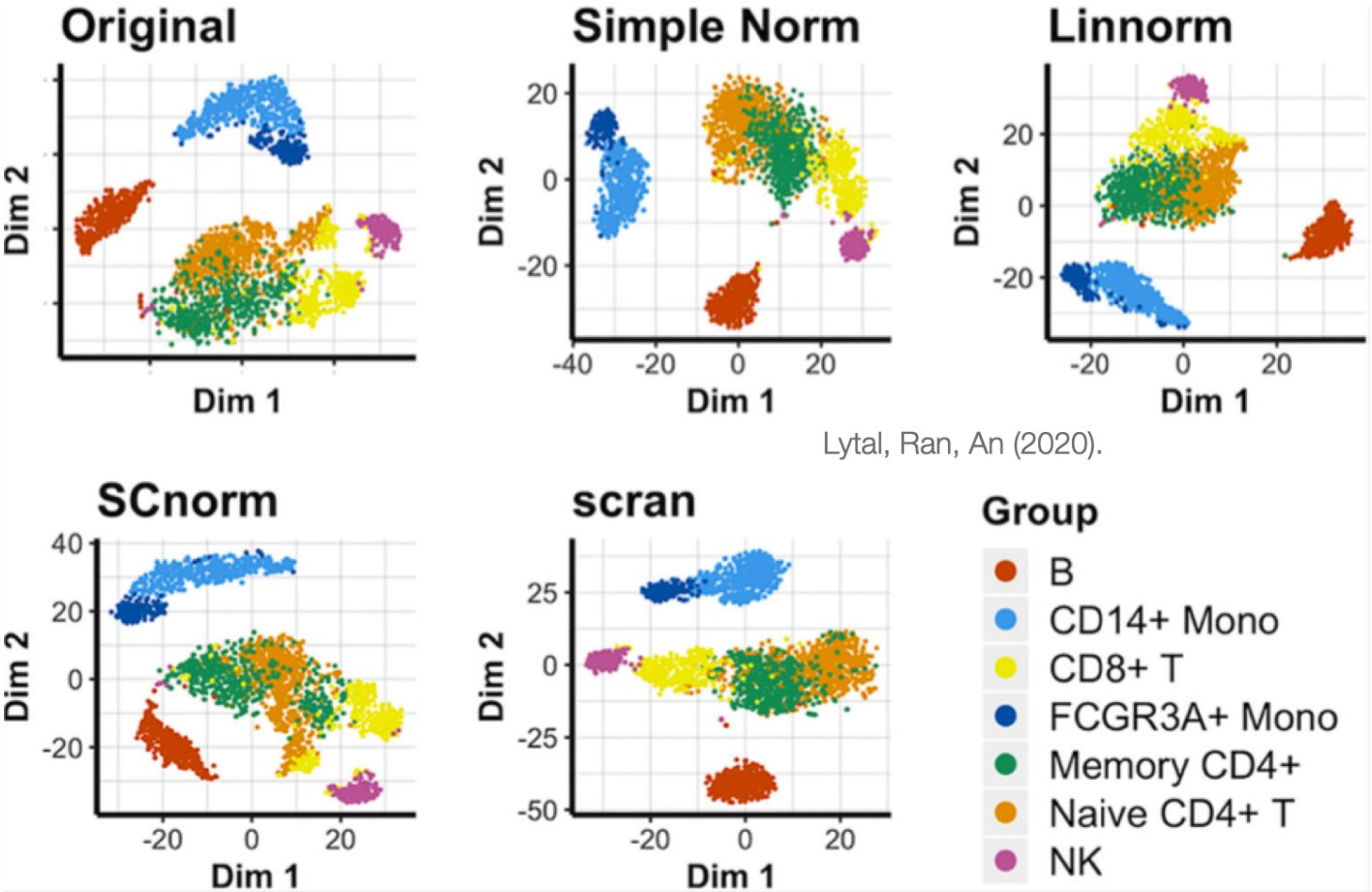
3. Normalization: applying different scale factors for different groups of genes



scNorm caclulates different scale factors for different groups of genes (grouping based on count-depth-relationships)

3. Normalization: effect on dim.reduction & clustering

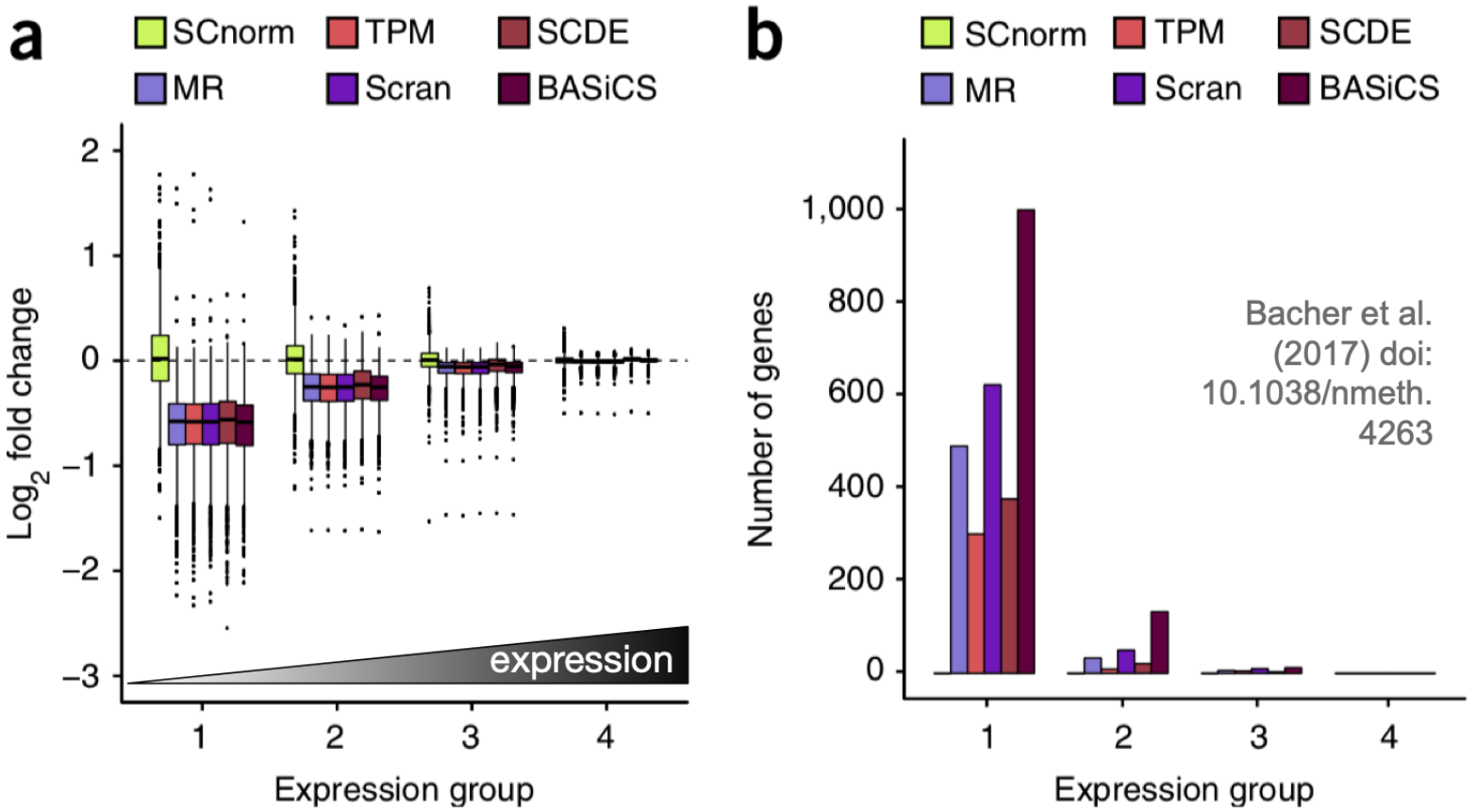
Normalization accuracy is not supremely important for exploratory analyses, i.e. simple size-factor normalization is often "good enough" [Amezquita et al., 2020, Germain et al., 2020].



Lytal, Ran, An (2020).

3. Normalization: effect on logFC and marker gene detection

Normalization is **extremely important** for marker gene detection and every gene-wise comparison.



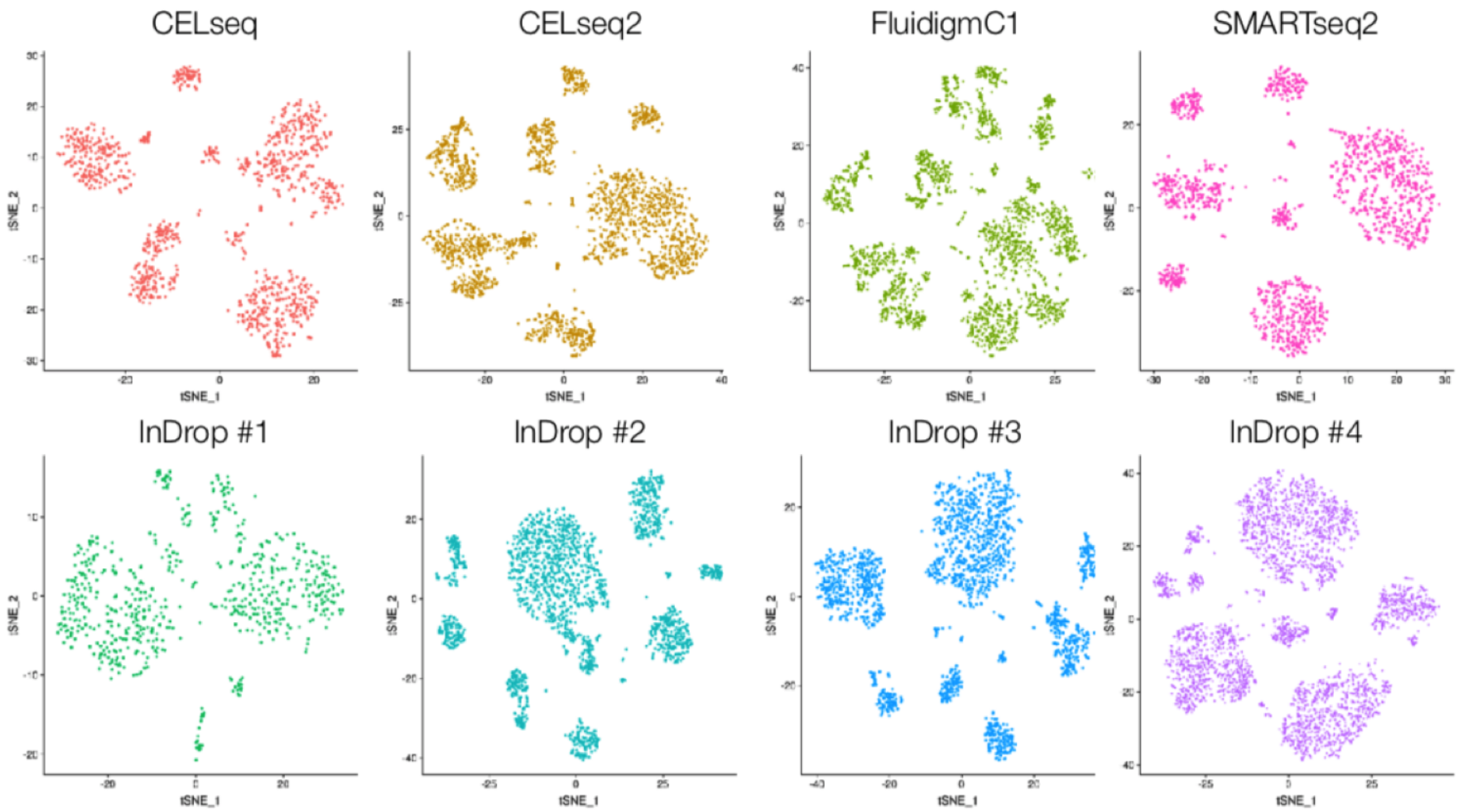
3. Normalization with `scTransform`

The same observations that Bacher et al. [2017] described for low-throughput, non-UMI-based data sets hold true for the `scTransform` method by Hafemeister and Satija [2019], which was specifically developed for droplet-based data.

1. GLM is used to **fit model** parameters (neg. binom) for each gene using **sequencing depth as a covariate**.
2. Resulting parameters are **regularized** based on a gene's average expression (variance adjustment).
3. 2nd round of NB regression, this time **constraining the parameter estimates** to the limits found in (2).
4. seq. depth **normalized and variance-stabilized** expression values: *Pearson residuals* = *residuals*/*SE*

`scTransform` is the method of choice for **droplet- and UMI-based** data.

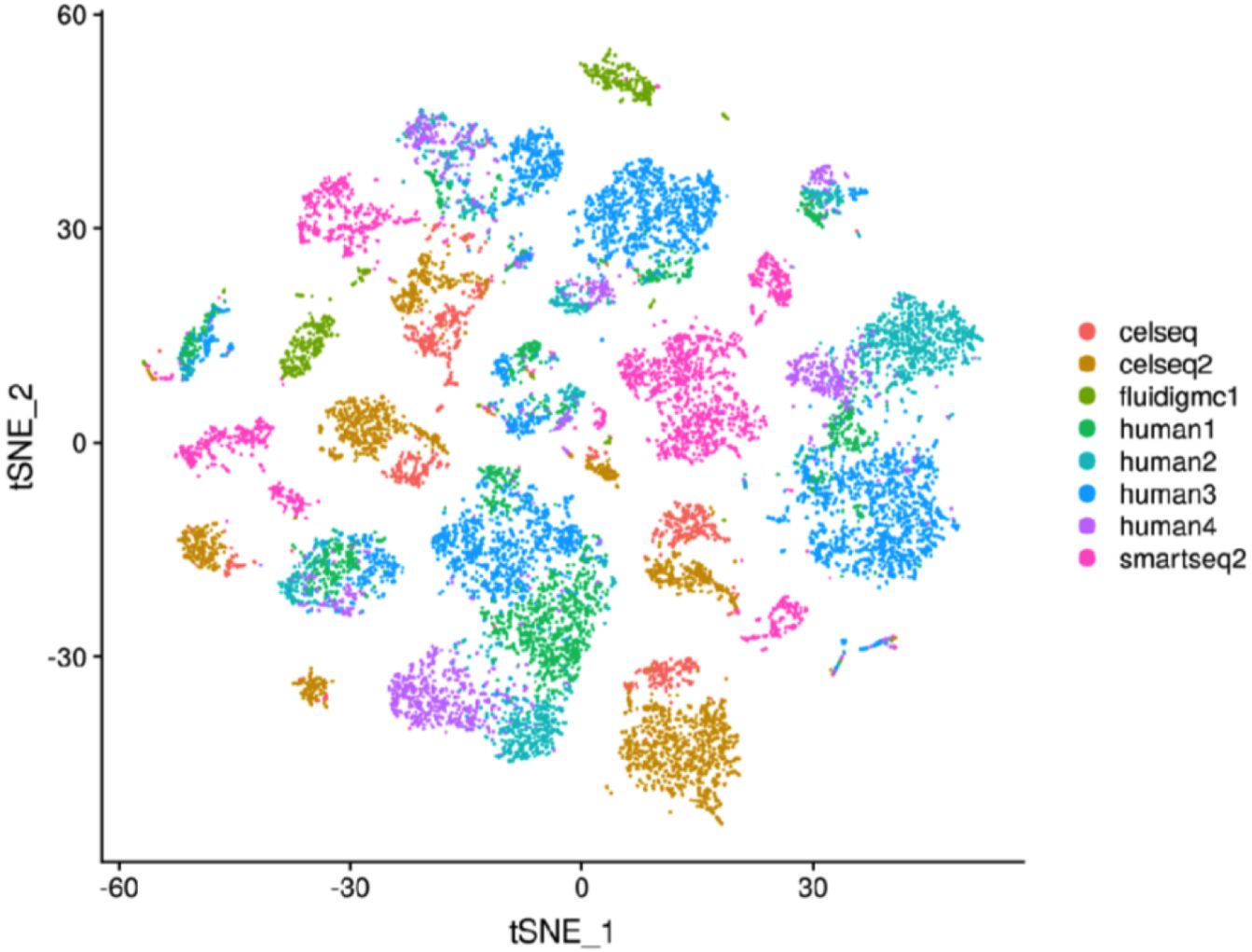
4. Batch correction and data integration



Data from Baron et al. 2016, Cell Syst.; Lawlor et al. 2017, Genome Res.; Grün et al 2016, Cell Stem Cell; Muraro et al. 2016 Cell Syst.

images courtesy Tim Stuart

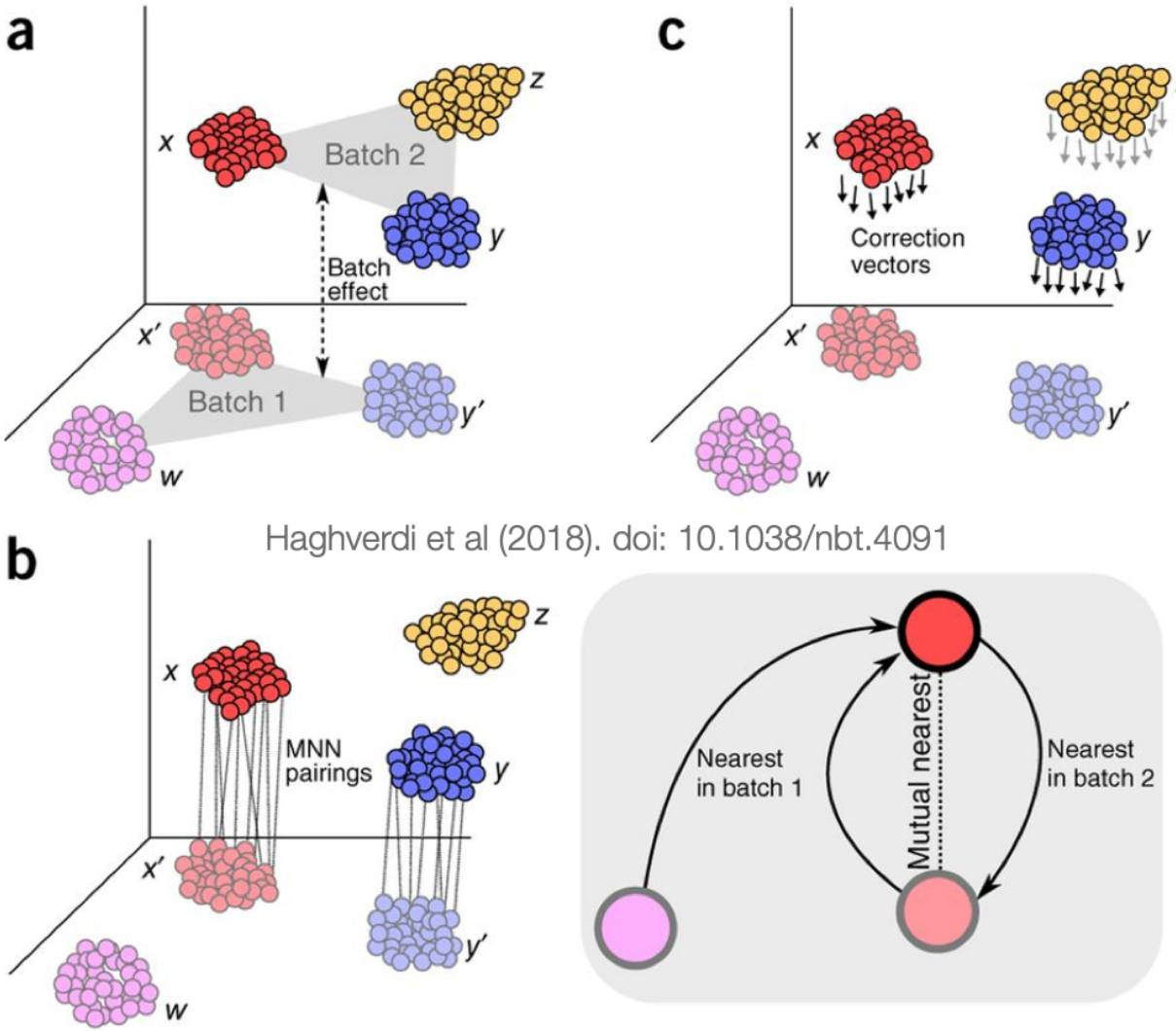
4. Batch correction for integrative analyses



All samples were derived from pancreas.

Merging all samples into one matrix without additional batch correction will lead to artificial clusters.

4. Batch correction for integrative analyses: MNN



Haghverdi et al (2018). doi: 10.1038/nbt.4091

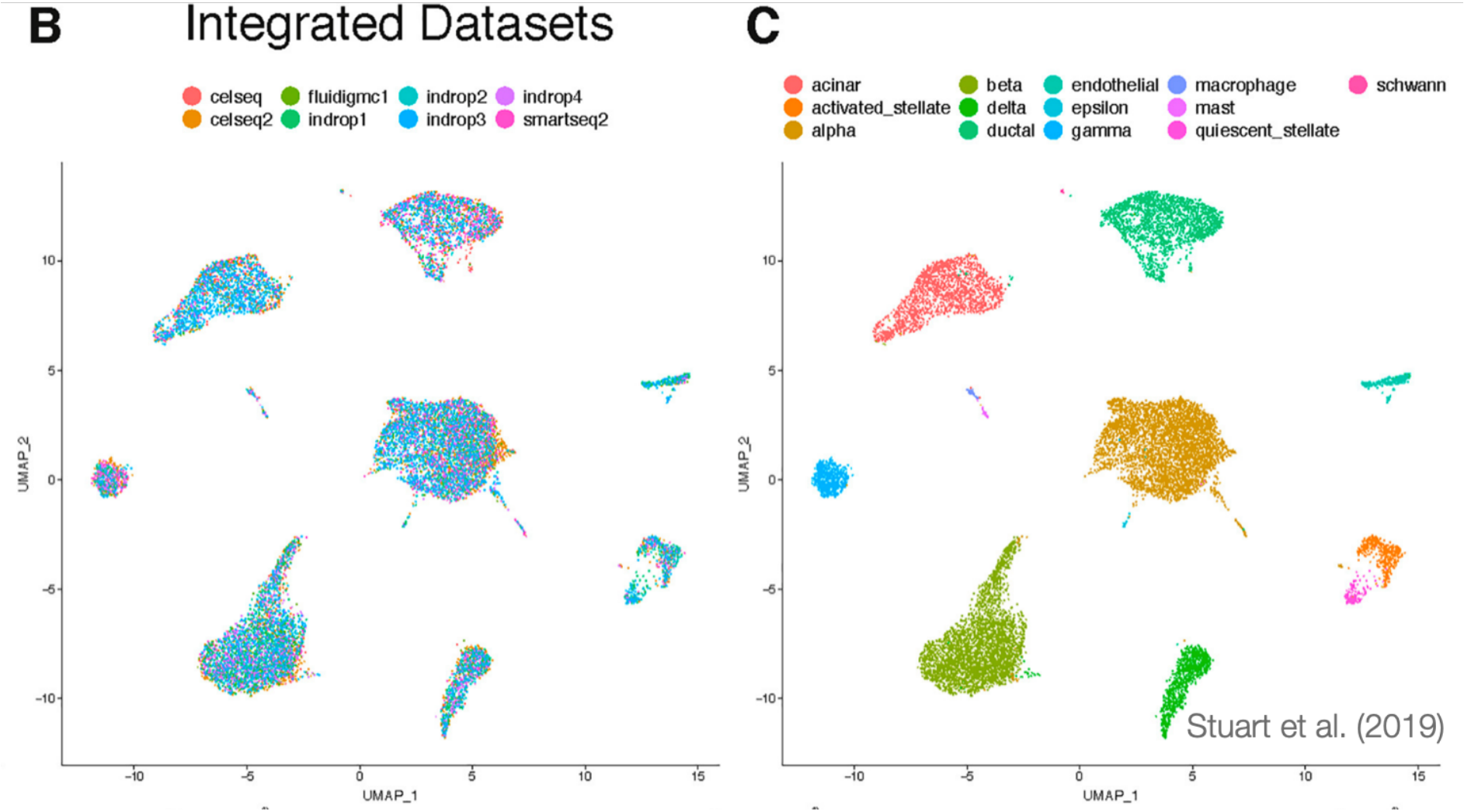
1. Mutual Nearest Neighbors = most similar cells across batches

2. mean difference between cells in an MNN pair ~ batch effect

3. correction vector applied to the expression values = batch correction

The CCA-based integration implemented in `Seurat` is similar in spirit [Stuart et al., 2019].

4. Batch correction for integrative analyses



After addressing the batch effect of "experiment", the clustering reveals the different cell types.

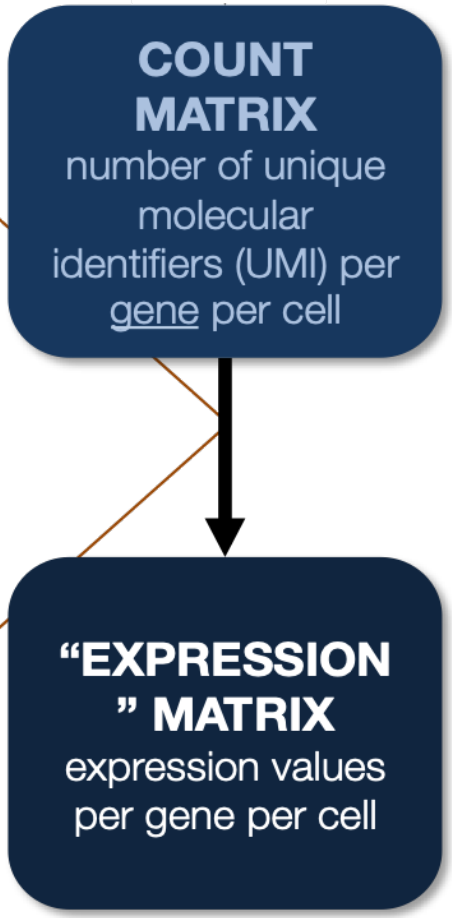
Summary of basic count matrix processing steps

Filtering cells
- require certain # UMI and genes per cell
- remove cells with high mitochondrial content

Filtering genes
- require minimal detection threshold for individual genes

Adjusting for different library sizes (N) per cell
- e.g. scTransform (Hafemeister 2019), scater (Lun 2016)

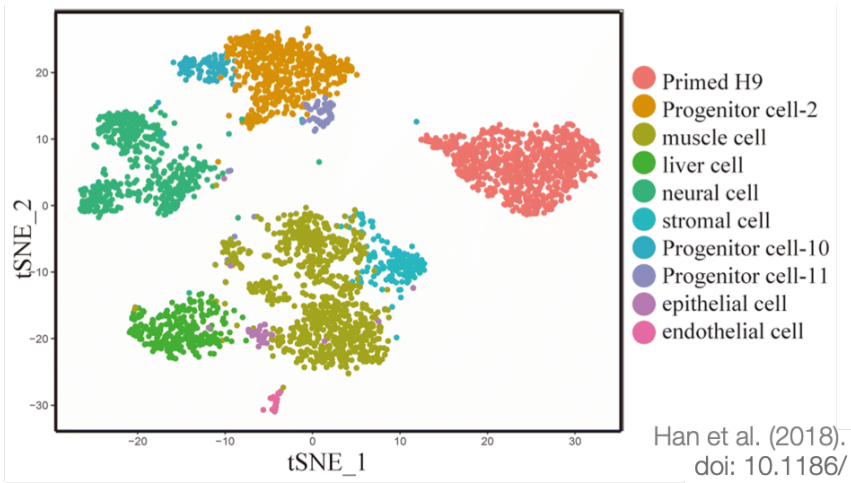
Possibly batch effect removal/sample alignment
- e.g. MNNcorrect (Haghverdi 2018), Seurat v3 (Stuart 2018)



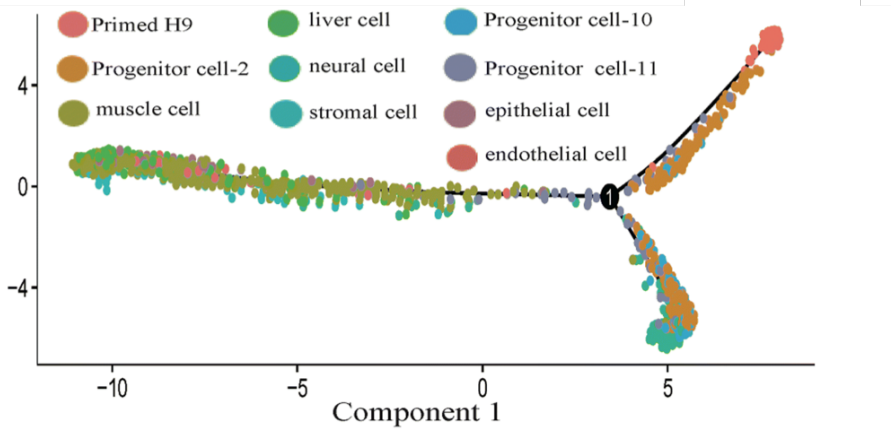
How to draw biologically meaningful insights from scRNA-seq?

Identifying cell types and/or cell states of interest

- visualizations of **dimensionality reduction**
  - PCA, tSNE, UMAP, Diffusion Maps
- **clustering**
  - k-means, hierarchical clustering, graph-based community detection
- **marker gene identification**
  - DGE detection between clusters of interest, followed by GO term & pathway enrichment analyses
- **trajectory inference**
  - inferring developmental timeline/ordering

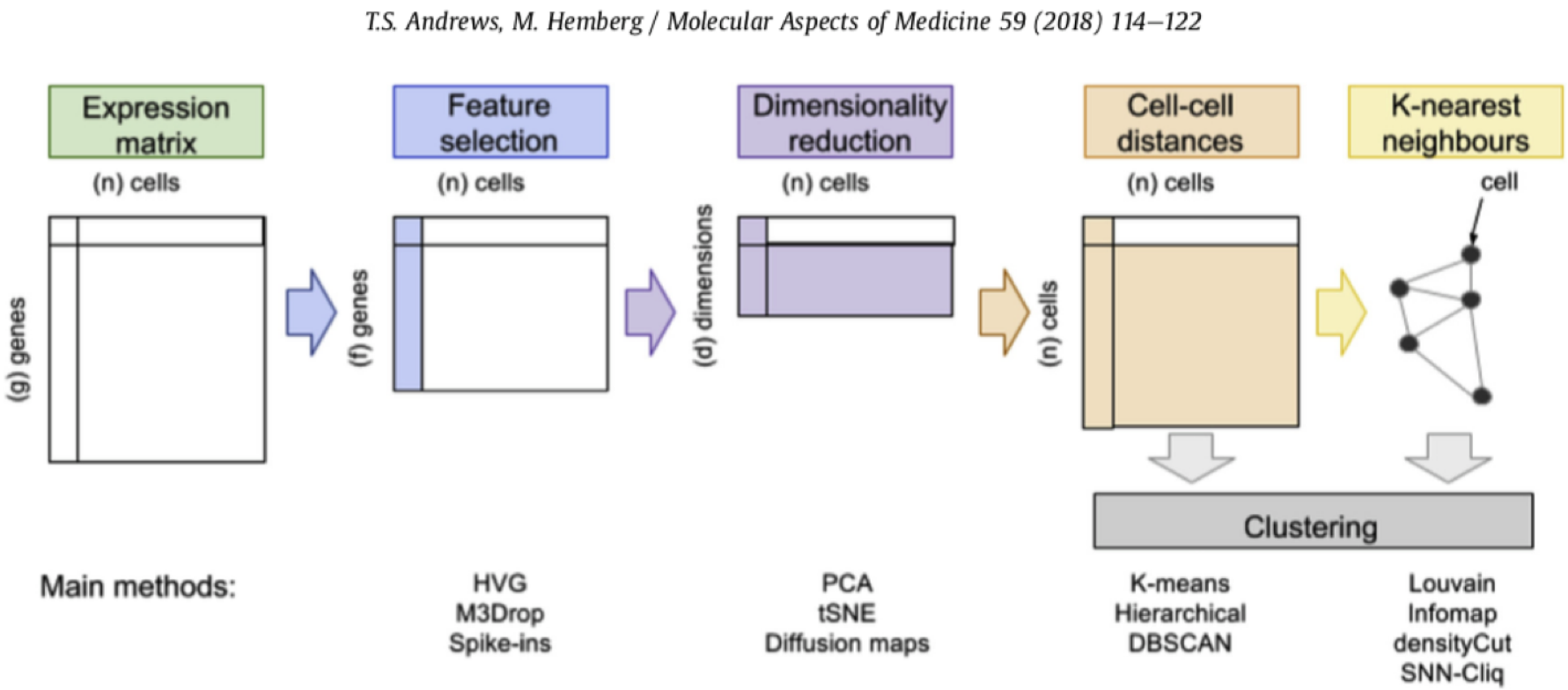


Han et al. (2018). doi: 10.1186/s13059-018-1426-0



There is no consensus on how to define "cell type" [Clevers et al., 2017]!

Common workflow for identifying clusters



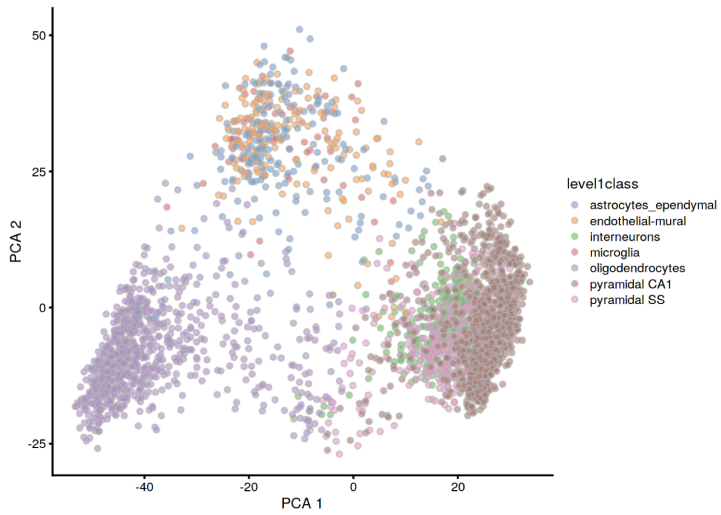
Feature selection – example code

Many more details here: https://osca.bioconductor.org/feature-selection.html

```
## see https://osca.bioconductor.org/feature-selection.html for
## how sce.pbmc was generated
library(scran);library(magrittr)

## model the gene-wise variance trying to separate technical from biological var.
## also allows for blocking on batch factors etc.
dec.pbmc <- modelGeneVar(sce.pbmc)
## extract the top 10% of genes w/ supposedly highest biological components
chosen <- getTopHVGs(dec.pbmc, prop=0.1)
```

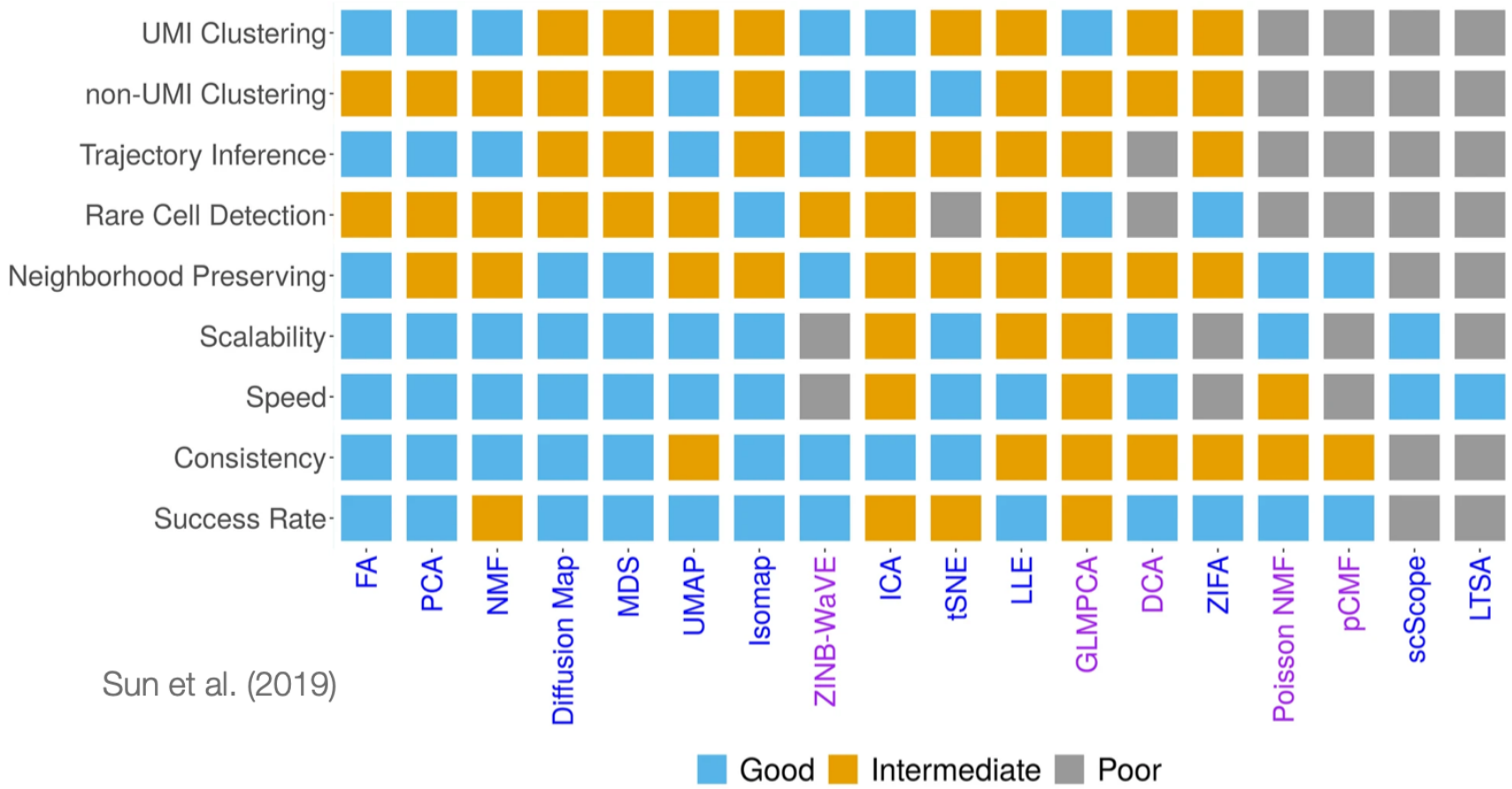
Dimensionality reduction:

```
set.seed(123)
sce.pbmc <- runPCA(sce.pbmc,
  subset_row=chosen)
## accessing PCA coordinates:
reducedDim(sce.pbmc, "pca")
plotReducedDim(sce.zeisel, dimred="PCA",
  colour_by="level1class")
```



1. Dimensionality reduction methods

Common goal: extract vectors that capture the majority of the **biologically meaningful** variation, ideally in (way) fewer than 20,000 features.

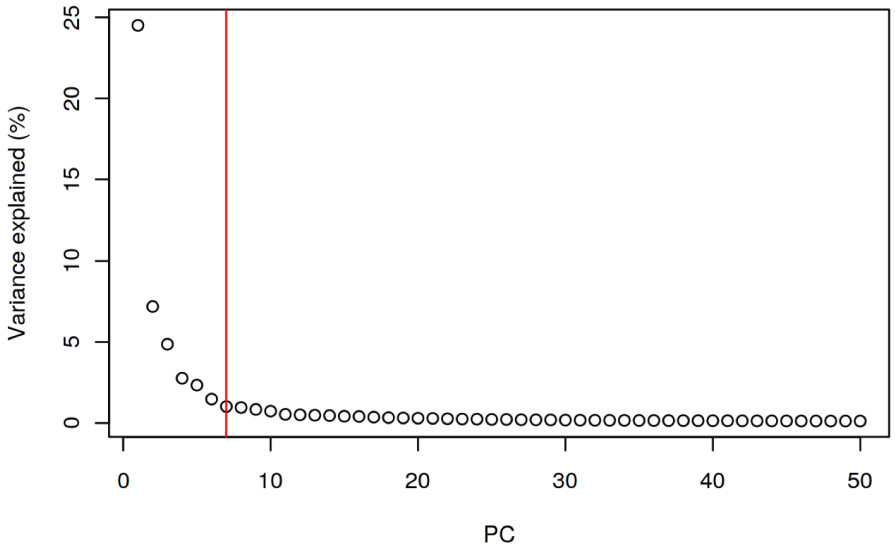


PCA, UMAP, and diffusion maps are the most commonly used methods.

1. Dimensionality reduction techniques

Scree plots are often used to decide how many PCs should be kept for downstream analyses. They display the fraction of the variance that is explained by each PC.

```
percent.var <- attr(reducedDim(sce.zeisel), "percentVar")
chosen.elbow <- PCAtools::findElbowPoint(percent.var)
plot(percent.var, xlab="PC", ylab="Variance explained (%)")
abline(v=chosen.elbow, col="red")
## retain selected number of PCs
reducedDim(sce.zeisel, "PCA") <- reducedDim(sce.zeisel, "PCA")[,1:20]
```



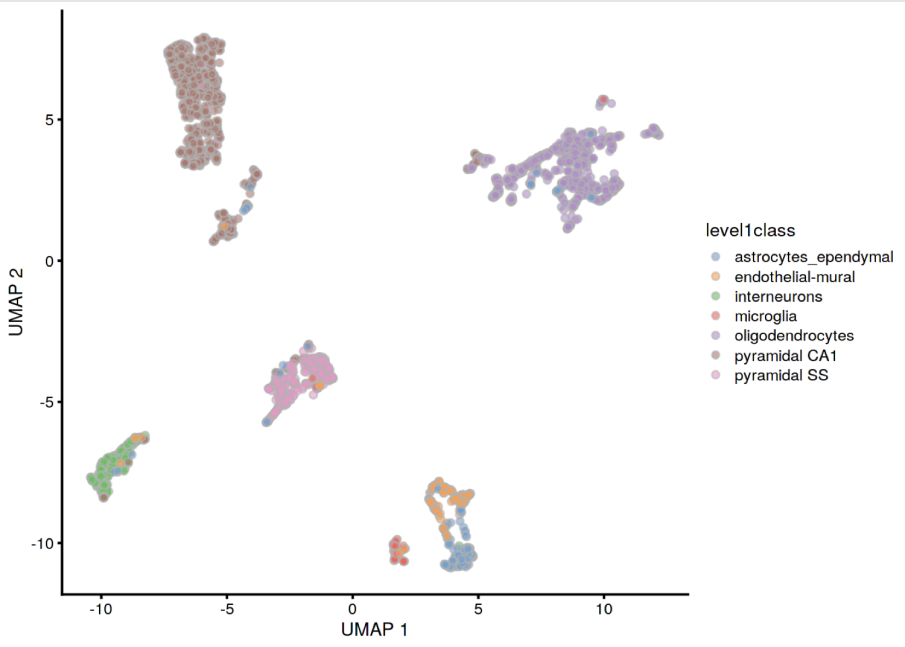
Typically, we retain about 15-35 dimensions.

1. Dimensionality reduction techniques

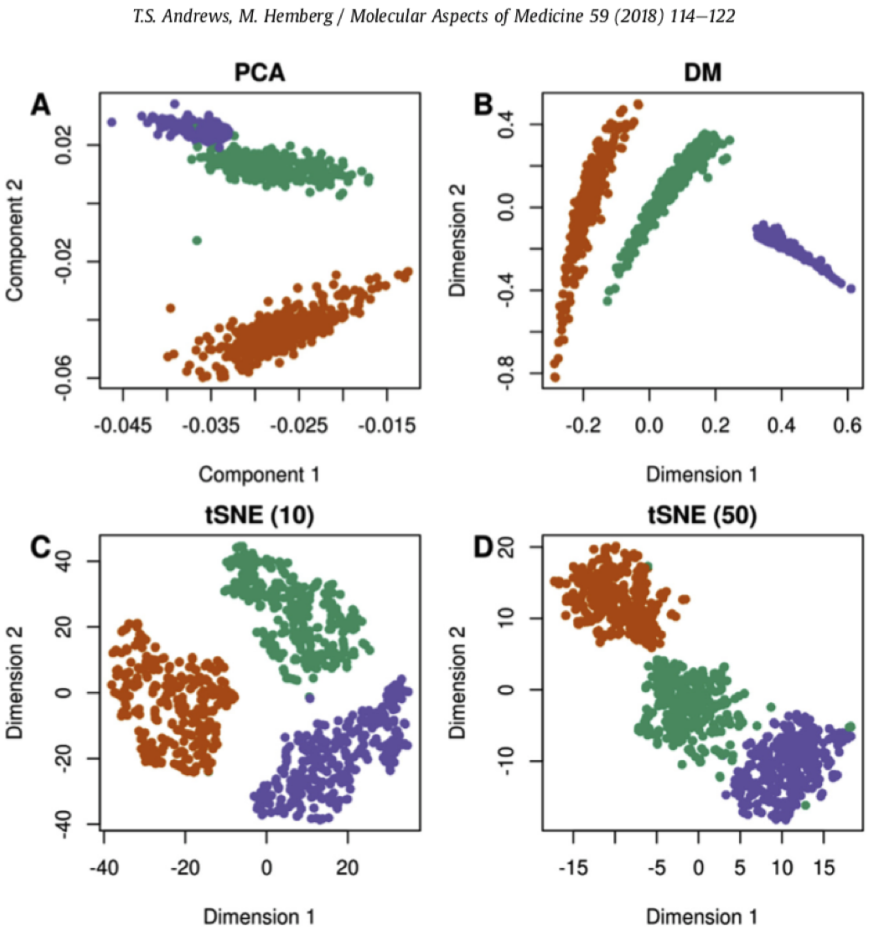
UMAP: Uniform manifold approximation and projection

- non-linear dimensionality reduction, fairly similar to t-SNE [Van Der Maaten et al., 2008, Becht et al., 2019]
- tries to find a lower-dimensional representation that preserves relationships between neighbors
- is typically performed not on the PCA-reduced space, not the full matrix

```
## define that the PCA coordinates
### should be used for UMAP
sce.zeisel <- runUMAP(sce.zeisel,
  dimred="PCA")

## UMAP typically only returns
## 2 dimensions (for reasons of
## computation time)
plotReducedDim(sce.zeisel, dimred="UMAP",
  colour_by="level1class")
```



1. Dimensionality reduction techniques – summary



- PCA preserves variance
- diffusion map finds non-linear trajectory (better for continuous data)
- tSNE and UMAP highlight clustering structure, i.e. local neighborhoods

2. Clustering methods implemented for scRNA-seq

"empirically define groups of cells with similar expression profiles"

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[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[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

DBSCAN, density-based spatial clustering of applications with noise; PCA, principal component analysis; scRNA-seq, single-cell RNA sequencing.

Kiselev (2019). doi: 10.1038/s41576-018-0088-9

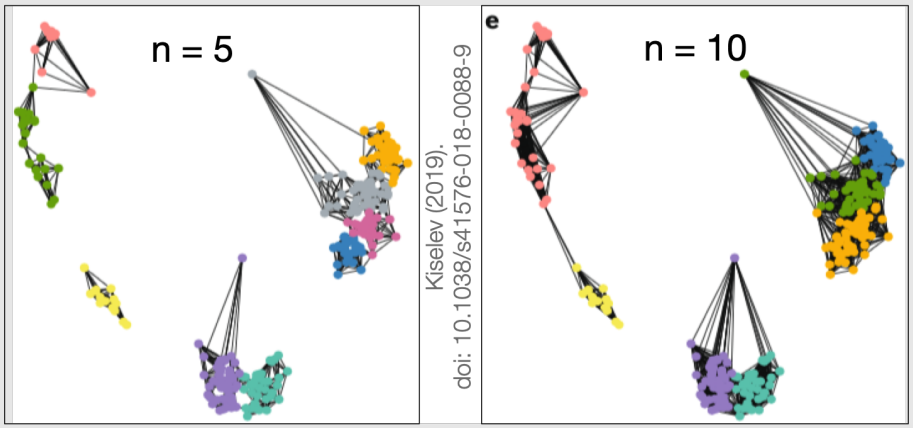
For assessments of the different clustering techniques for scRNA-seq data, see Freytag et al. [2018], Duò et al. [2018], Menon [2018].

No size fits all, but Seurat's graph-based clustering works reasonably well for high-throughput, droplet-based approaches.

2. Clustering

Graph clustering/community detection

- clusters = groups of **nodes** that are densely connected
- density is a user-specified parameter
- works well on many (>1000) cells



1. select the top x PCs that capture the majority of the *gene* signatures
2. construct a graph where nodes = *cells*, edges = similarity measures (based on PCs)
3. for every cell, identify its k-nearest-neighbours (**SNN** graph), i.e. every cell::neighbor pair gets a weight that captures the similarity of the two cells' neighborhoods (that consist of k NN each!)
4. use the iterative Louvain community detection method to identify groups of nodes that are densely connected

See Andrews and Hemberg [2018] and Kiselev et al. [2019] for details for the clustering techniques.

2. Clustering

Graph clustering/community detection

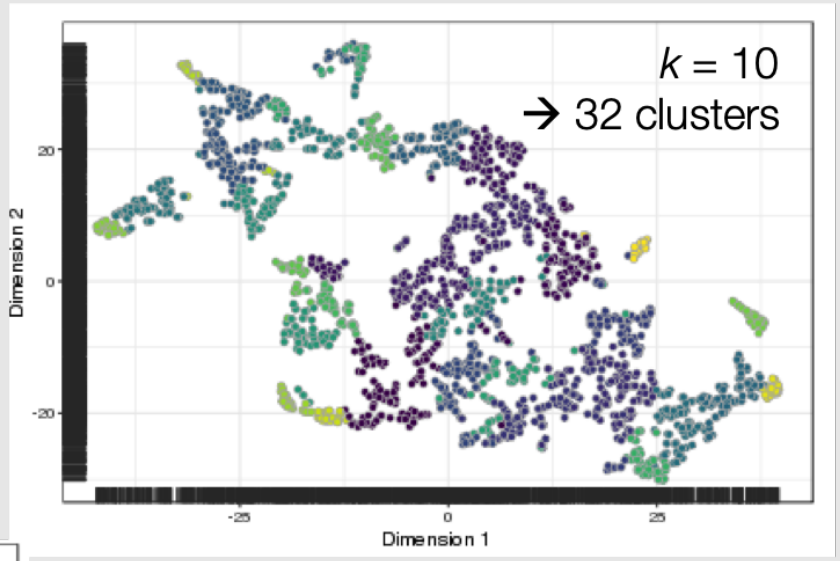
Dimensionality reduction with **PCA**

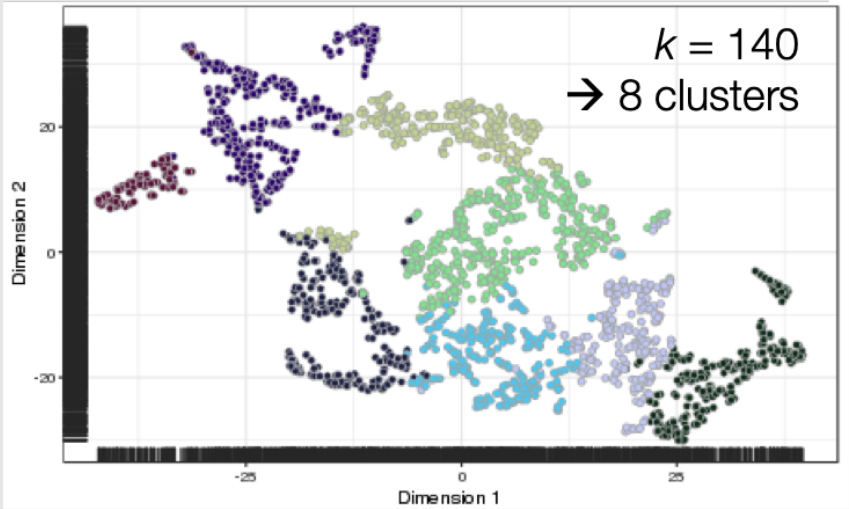
↓

pairwise cell-cell distances based on shared **nearest neighbors**

↓

identifying **communities** of densely connected cells (Louvain)





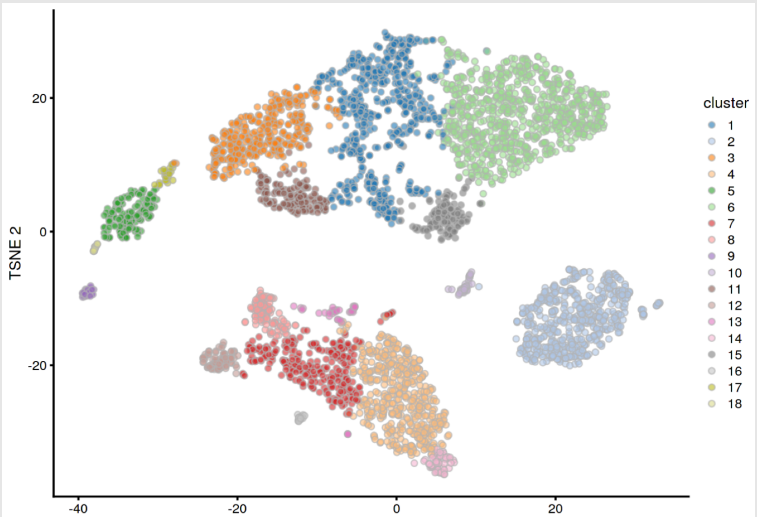
- How many PCs for the graph construction?
- **How many (k) neighbors for the initial graph?**
- How many iterations of the Louvain algorithm?

2. Clustering

Seurat-style clustering with bioconductor

See https://osca.bioconductor.org/clustering.html for many more details.

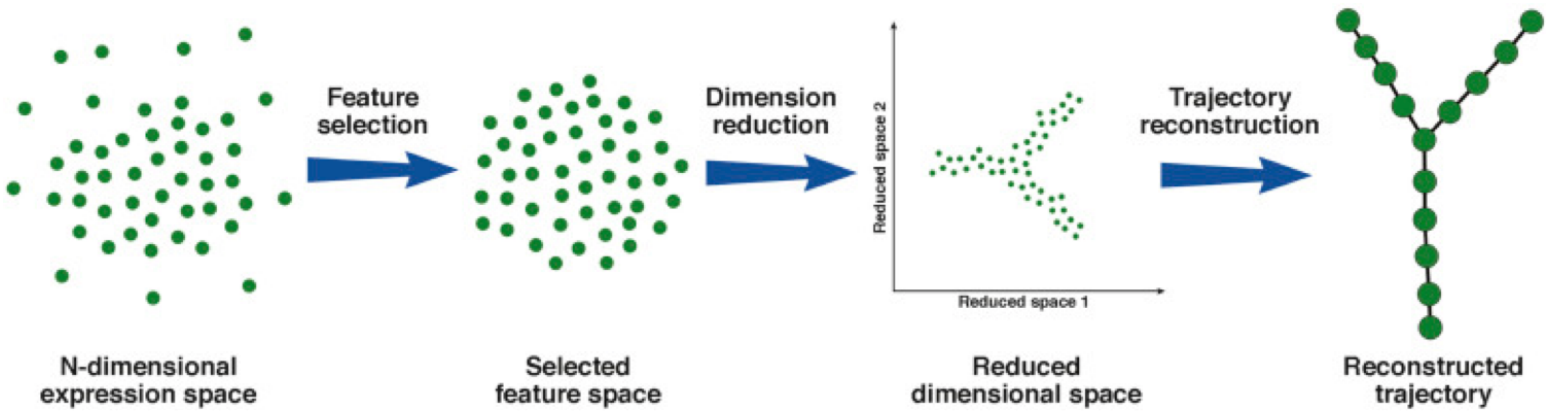
```
library(scran)
g <- buildSNNGraph(sce.pbmc, k=10, use.dimred = 'PCA')
clust <- igraph::cluster_walktrap(g)$membership
## higher resolution (fewer neighbors)
g.5 <- buildSNNGraph(sce.pbmc, k=5, use.dimred = 'PCA')

## store cluster info in SCE
sce.pbmc$cluster <- factor(clust)
plotReducedDim(sce.pbmc, "TSNE", colour_by="cluster")
```



4. Trajectory inference

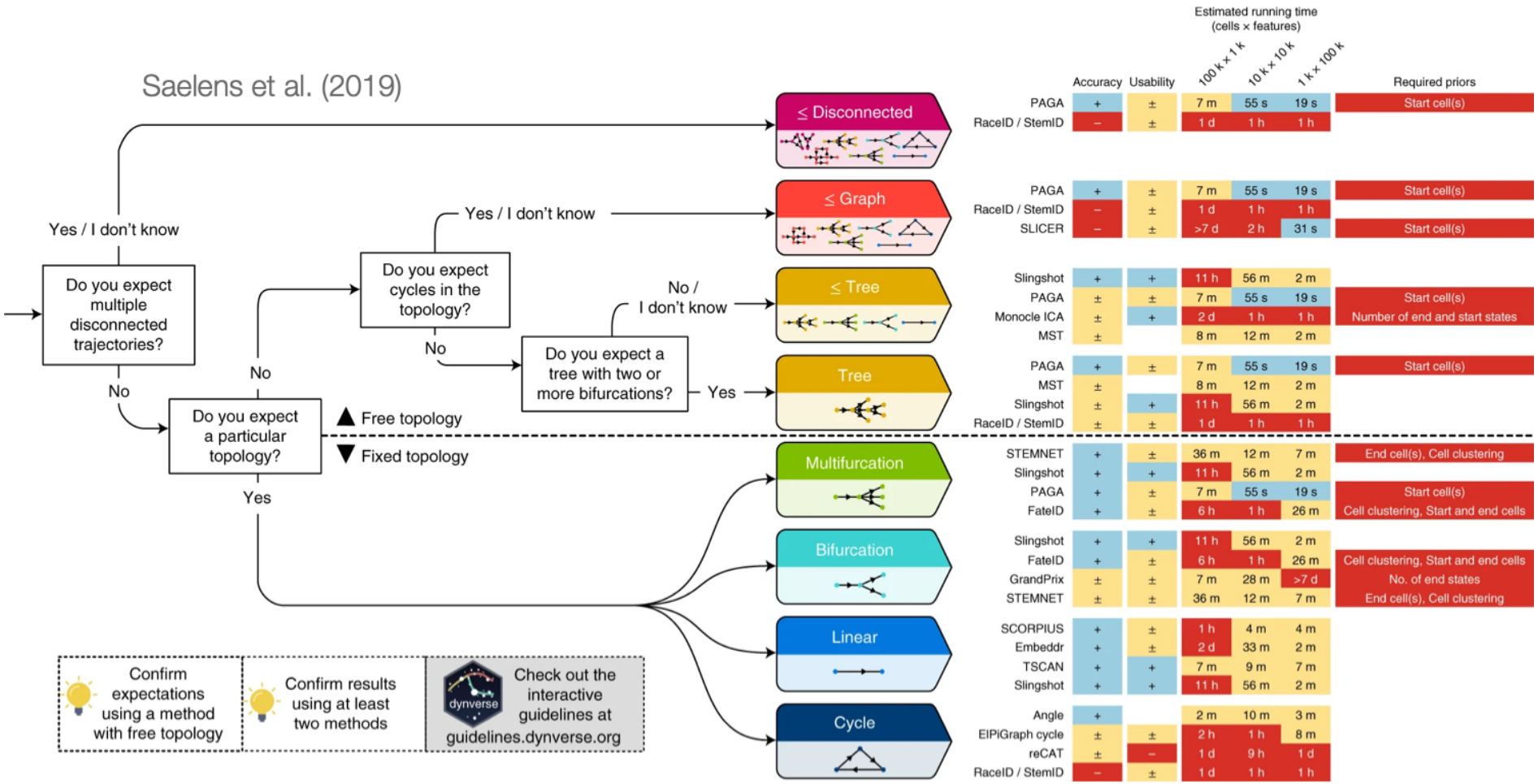
= ordering cells along a pseudotime trajectory where pseudotime is calculated based on expression similarities of neighboring cells

- many different topologies are, theoretically, possible, but most methods focus on inferring **linear trajectories** or limit themselves to less complex topologies
- can handle non-linear processes; more appropriate than clustering for **continuous** data along a **trajectory**
- pseudotime != real time [4]; the direction of the order is often reversed, too
- absolutely depends on **cells representing the transitional states** to be present in the data!



[4] A longer branch can simply reflect a lineage with more cells.

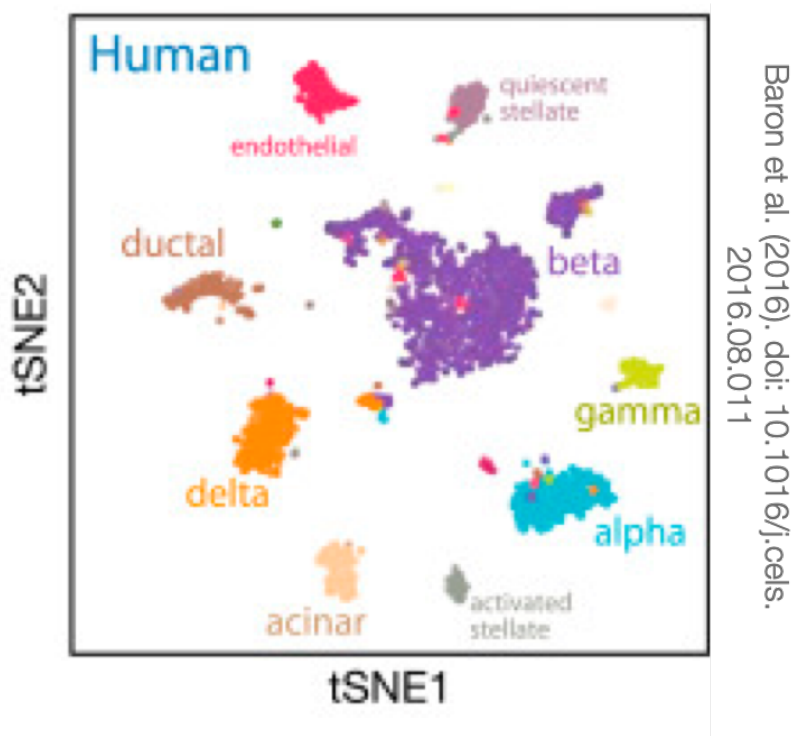
4. Trajectory techniques – how to choose



`slingshot` works reasonably well and is available via BioC [Street et al., 2018]; do read the excellent benchmark paper of trajectory inference methods by Saelens et al. [2019].

Getting some feeling for replicability and biological significance of CELL TYPES/POPULATIONS

- repeated runs (incl. different tools) of clusterings etc. will only give you an idea of the **technical** robustness of your parameter choices
- cell types may be compared across different species
- *known* marker genes may give some insights into significance of individual clusters



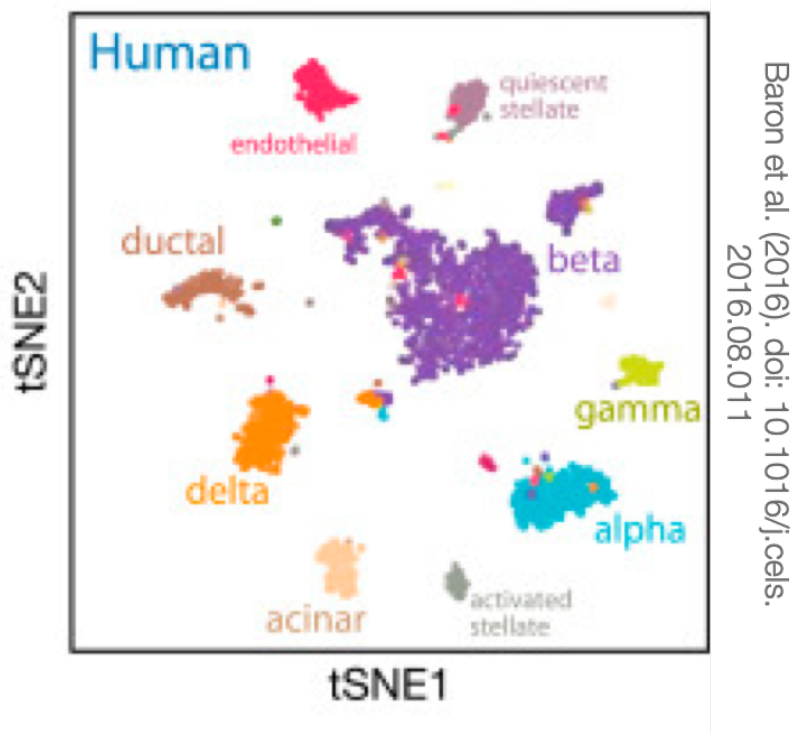
If cell types differ by few genes, we will not pick them up!

Getting some feeling for replicability and biological significance of MARKER GENES

Typical cell identity signals are robust & low-dimensional! [Crow and Gillis, 2018, Heimberg et al., 2016]

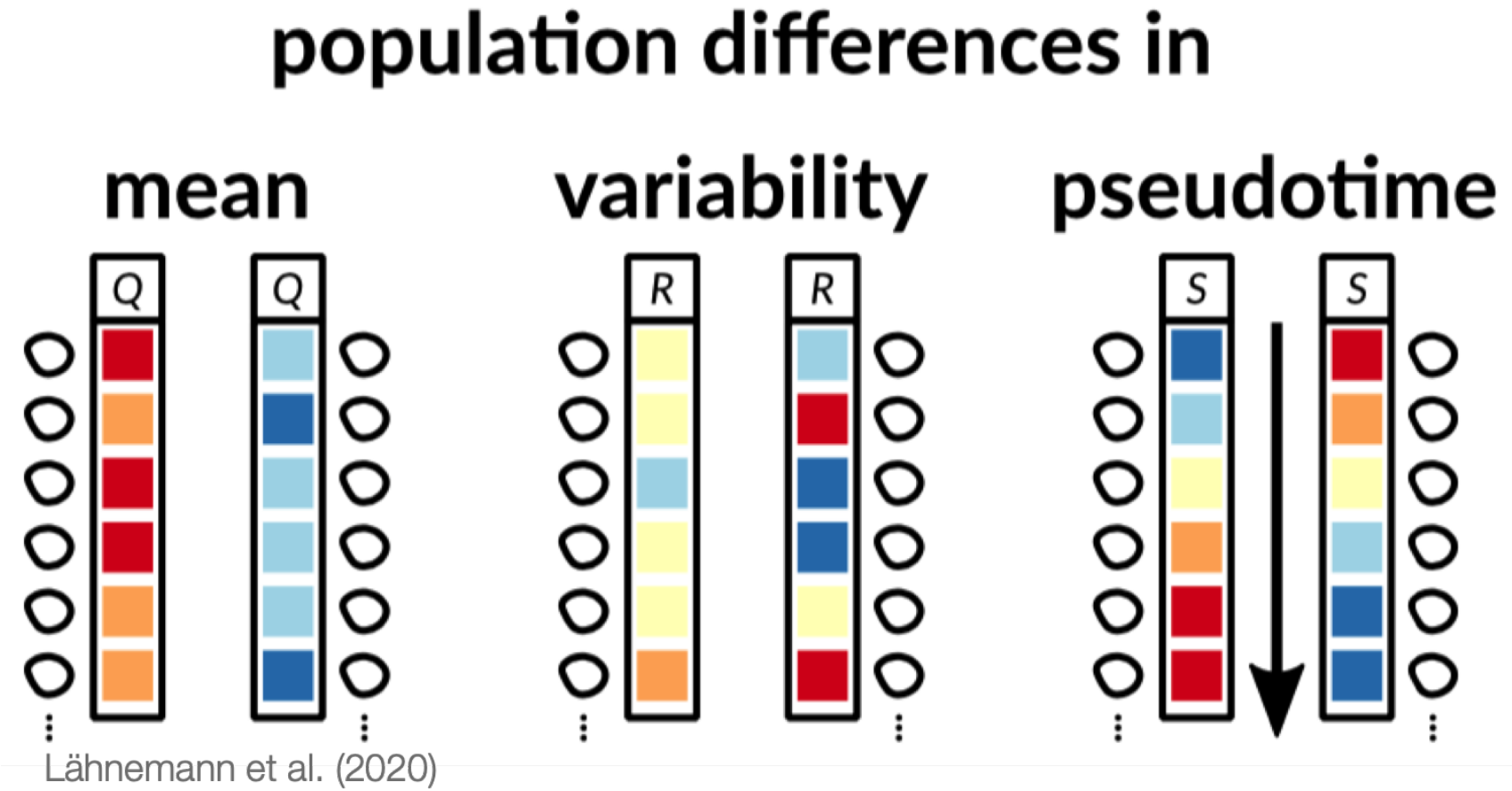
- ca. 100 genes: distinguish glia vs. neurons (1st PC)
- ca. 1,000 genes: distinguish neuron subtypes (PC1-3)

The genes you identify as "markers" may just have highly correlated expression patterns with the true drivers of the cell identity.



Novel marker gene identifications must be followed up by **additional experiments**.

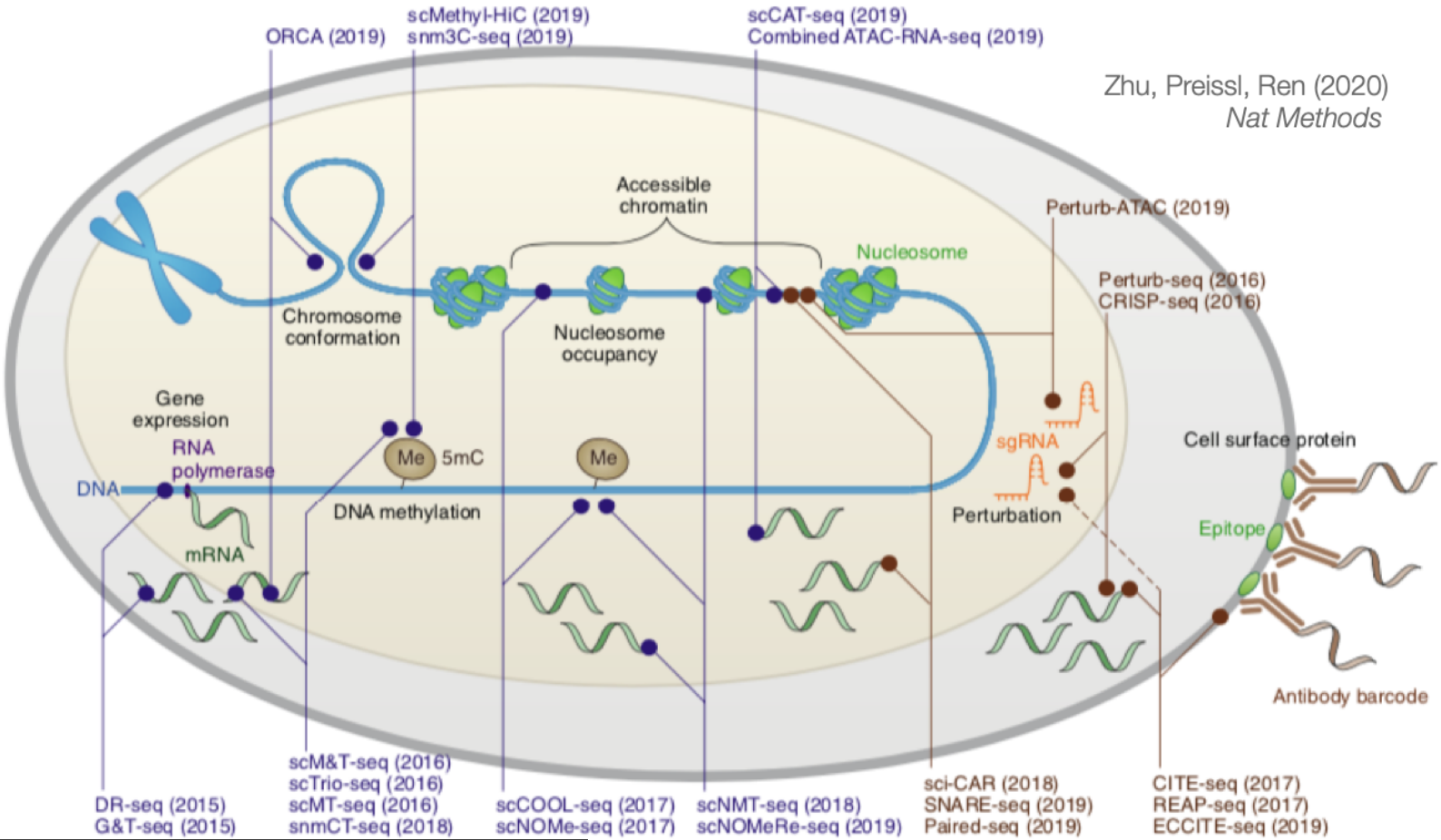
Biologically meaningful differences can arise from different gene expression properties



Different types of patterns of interest require different tests/analyses.

Moving forward: multimodal single-cell measurements

Method of the year 2019: **simultaenous** measurement of 2 or more modalities[5] from the **same** cell. See Zhu et al. [2020] and Eisenstein [2020] for details.



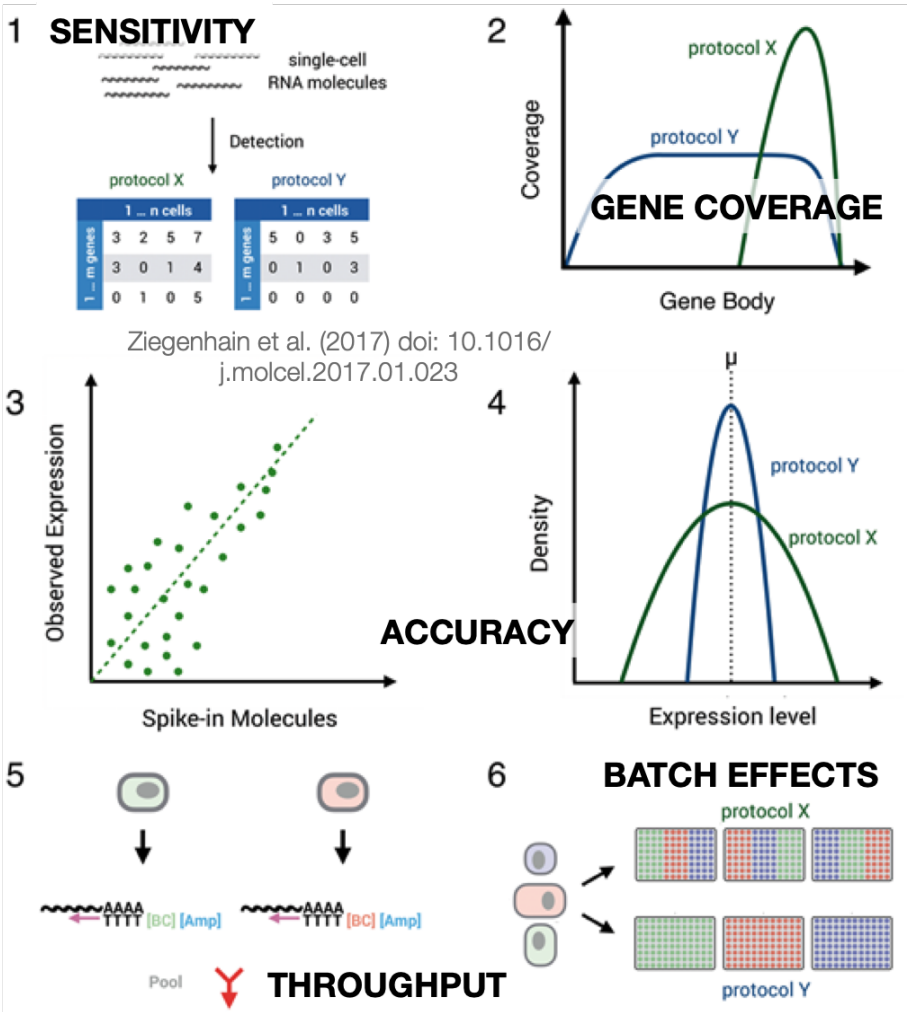
[5] → transcriptome, proteome, epigenetic components

Conclusions

Summary of typical processing steps

1. **Filtering**
   - doublets, empty droplets, droplets of remnant/dead cells
   - too rarely captures features
2. **Normalization** (and possibly integration)
   - `scTransform`
   - MNN or CCA
3. **Feature selection**
   - e.g. most variably expressed genes
4. **Dimensionality reduction**
   - obtain a subspace where distances between the cells are more reliable than in the full matrix
   - PCA, tSNE, UMAP
5. **Clustering and cell annotation**
   - e.g. via hand-picked marker genes or via automated methods such as `SingleR`

Every scRNA-seq technique has unique pros & cons



Decision will depend on:

- sample availability
- experimental question
- access to the method
- possibly previously published studies

Limits of scRNA-seq

- Technical challenges
  - **sensitivity** is still low
  - **costs** are still somewhat prohibitive
- Numerous sources of **cell-to-cell variability**
  - cell cycle
  - cell size
  - transcription bursts
  - stress during isolation
- **Analysis methods** are in their infancy!

Have a rationale!

What is your **hypothesis**? How are you going to distinguish transient from permanent effects? Do you have a way of obtaining some idea of the "ground truth"?

When NOT to use scRNA-seq (yet?)

- fairly **homogeneous populations**, true interest is in identifying the main effect of a treatment/condition/genotype...
- **complex experimental designs** (e.g., many experimental variables)
- genes of interest are known to be **lowly expressed**/subtly changing

Beware!

If you are interested in **individual *genes***, scRNA-seq should **not** be your first choice.

See Lafzi et al. [2018] for lots of practical advice before planning your own scRNA-seq experiment!

Examples of publicly available scRNA-seq data collections

Consortia-style efforts:

- Tabula muris, Human Cell Atlas, Allen Brain Map
- Single Cell Expression Atlas

Repositories for **published data sets** (providing processed data):

- Single Cell Portal (Broad Institute) – processed by the individual groups themselves
- Conquer – uniformly processed samples, includes QC reports! [Soneson and Robinson, 2018]
- **`scRNAseq` package** $\rightarrow$ allows you to load diverse data sets directly as SCE objects into your R workspace

Interactive visualization tools

Cakir (2020)	ASAP	Bbrowser	cellxgene	Granatum	iSEE	Loom viewer	Loupe Cell Browser	SCope	scSVA	scVI	Single Cell Explorer	SPRING	UCSC Cell Browser
Web Interface	✓		✓		✓	✓		✓	✓		✓	✓	✓
Interactivity	✓	✓	✓		✓		✓	✓	✓		✓	✓	✓
Docker	✓		✓		✓			✓	✓		✓		
Cloud Support				✓				✓	✓			✓	
Loom					✓	✓		✓	✓	✓	✓		
h5ad		✓	✓						✓	✓	✓		✓
SCE					✓								
Seurat		✓									✓		✓
csv/txt	✓	✓		✓					✓	✓	✓	✓	✓
Platform	Java/ R	Desktop	Python	R	R	Python	Desktop	Python	R	Python	Python	Python	Python

Cakir et al. (2020)	cellxgene	iSEE	Loom-viewer	scSVA	SCope	Single Cell Explorer	UCSC Cell Browser
Ease of cell selection	✓	✓		✓	✓	✓	✓
Zoom in/out	✓	✓		✓	✓		✓
Multiple embeddings	✓	✓	✓		✓		✓
Highlight gene expression	✓	✓		✓	✓	✓	✓
Highlight metadata	✓	✓	✓	✓		✓	✓
Extra analysis	✓	✓		✓		✓	
Web page loads fast	✓		✓	✓	✓	✓	✓

Coding resources

- "Orchestrating Single-Cell Analysis with **Bioconductor**" [Amezquita et al., 2020]: https://osca.bioconductor.org/
- **Seurat** vignettes: https://satijalab.org/seurat/vignettes.html
- Hemberg Lab/Kiselev Lab course (BioC & Seurat): https://scrnaseq-course.cog.sanger.ac.uk/website/index.html

References

Figures taken from the following publications:

[Amezquita et al., 2020, Andrews and Hemberg, 2018, Bacher et al., 2017, Baron et al., 2016, Beltrame et al., 2019, Blibaum et al., 2019, Cakir et al., 2020, Chen et al., 2018, Cuvier and Fierz, 2017, Dal Molin and Di Camillo, 2018, Haghverdi et al., 2018, Han et al., 2018, Hwang et al., 2018, Lähnemann et al., 2020, L. Lun et al., 2016, Kiselev et al., 2019, Klein et al., 2015, Lytal et al., 2020, Menon, 2018, Papalexi and Satija, 2018, Zhu et al., 2020, Saelens et al., 2019, Soneson and Robinson, 2018, Stuart et al., 2019, Sun et al., 2019, Svensson et al., 2018, Wang and Kaestner, 2018, Zhang et al., 2019, Ziegenhain et al., 2017, Zilionis et al., 2017]

Robert A. Amezquita, Aaron T.L. Lun, Etienne Becht, Vince J. Carey, Lindsay N. Carpp, Ludwig Geistlinger, Federico Marini, Kevin Rue-Albrecht, Davide Risso, Charlotte Soneson, Levi Waldron, Hervé Pagès, Mike L. Smith, Wolfgang Huber, Martin Morgan, Raphael Gottardo, and Stephanie C. Hicks. Orchestrating single-cell analysis with Bioconductor. *Nature Methods*, 17:137–145, 2020. doi: 10.1038/s41592-019-0654-x.

Tallulah S. Andrews and Martin Hemberg. Identifying cell populations with scRNASeq. *Molecular Aspects of Medicine*, 2018. doi: 10.1016/j.mam.2017.07.002.

Rhonda Bacher, Li Fang Chu, Ning Leng, Audrey P. Gasch, James A. Thomson, Ron M. Stewart, Michael Newton, and Christina Kendziorski. SCnorm: Robust normalization of single-cell RNA-seq data. *Nature Methods*, 2017. doi: 10.1038/nmeth.4263.

Abha S Bais and Dennis Kostka. scds: computational annotation of doublets in single-cell RNA sequencing data. *Bioinformatics*, 2019. doi: 10.1093/bioinformatics/btz698.

Maayan Baron, Adrian Veres, Samuel L. Wolock, Aubrey L. Faust, Renaud Gaujoux, Amedeo Vetere, Jennifer Hyoje Ryu, Bridget K. Wagner, Shai S. Shen-Orr, Allon M. Klein, Douglas A. Melton, and Itai Yanai. A Single-Cell Transcriptomic Map of the Human and Mouse Pancreas Reveals Inter- and Intra-cell Population Structure. *Cell Systems*, 3(4): 346–360, 2016. doi: 10.1016/j.cels.2016.08.011.

Etienne Becht, Leland McInnes, John Healy, Charles Antoine Dutertre, Immanuel W.H. Kwok, Lai Guan Ng, Florent Ginhoux, and Evan W. Newell. Dimensionality reduction for visualizing single-cell data using UMAP. *Nature Biotechnology*, 2019. doi: 10.1038/nbt.4314.

Eduardo Beltrame, Jase Gehring, Valentine Svensson, Dongyi Lu, Jialong Jiang, Matt Thomson, and Lior Pachter. Introduction to single-cell rna-seq technologies. 2 2019. doi: 10.6084/m9.figshare.7704659.v1. URL https://figshare.com/articles/Introduction_to_single-cell_RNA-seq_technologies/7704659.

Ash Blibaum, Jonathan Werner, and Alexander Dobin. STARsolo : single-cell RNA-seq analyses beyond gene expression . In *Genome Informatics*. F1000Research, 2019. doi: 10.7490/f1000research.1117634.1.

Batuhan Cakir, Martin Prete, Ni Huang, Stijn van Dongen, Pnar Pir, and Vladimir Yu. Kiselev. Comparison of visualisation tools for single-cell RNAseq data. *bioRxiv*, 2020. doi: 10.1101/2020.01.24.918342.

Xi Chen, Sarah A. Teichmann, and Kerstin B. Meyer. From Tissues to Cell Types and Back: Single-Cell Gene Expression Analysis of Tissue Architecture. *Annual Review of Biomedical Data Science*, 2018. doi: 10.1146/annurev-biodatasci-080917-013452.

Hand Clevers, Susanne Rafelski, Michael Elowitz, Allon Klein, Jay Shendure, Cole Trapnell, Ed Lein, Emma Lundberg, Matthias Uhlen, Martinez AriasAlfonso, Joshua Sanes, Paul Blainey, James Eberwine, Junhyong Kim, and Christopher Love. What Is Your Conceptual Definition of "Cell Type" in the Context of a Mature Organism? *Cell Systems*, 2017. ISSN 24054712. doi: 10.1016/j.cels.2017.03.006.

Megan Crow and Jesse Gillis. Co-expression in Single-Cell Analysis: Saving Grace or Original Sin? *Trends in Genetics*, 2018. doi: 10.1016/j.tig.2018.07.007.

Olivier Cuvier and Beat Fierz. Dynamic chromatin technologies: From individual molecules to epigenomic regulation in cells. *Nature Reviews Genetics*, 2017. doi: 10.1038/nrg.2017.28.

Alessandra Dal Molin and Barbara Di Camillo. How to design a single-cell RNA-sequencing experiment: pitfalls, challenges and perspectives. *Briefings in Bioinformatics*, 2018. doi: 10.1093/bib/bby007.

Jiarui Ding, Xian Adiconis, Sean K Simmons, Monika S Kowalczyk, Cynthia C Hession, Nemanja D Marjanovic, Travis K Hughes, Marc H Wadsworth, Tyler Burks, Lan T Nguyen, John Y H Kwon, Boaz Barak, William Ge, Amanda J Kedaigle, Shaina Carroll, Shuqiang Li, Nir Hacohen, Orit Rozenblatt-Rosen, Alex K Shalek, Alexandra-Chloé Villani, Aviv Regev, and Joshua Z Levin. Systematic comparative analysis of single cell RNA-sequencing methods. *bioRxiv*, 2019. doi: 10.1101/632216.

Angelo Duò, Mark D. Robinson, and Charlotte Soneson. A systematic performance evaluation of clustering methods for single-cell RNA-seq data. *F1000Research*, 2018. doi: 10.12688/f1000research.15666.1.

Michael Eisenstein. The secret life of cells. *Nature Methods*, 2020. doi: 10.1038/s41592-019-0698-y.

Saskia Freytag, Luyi Tian, Ingrid Lönnstedt, Milica Ng, and Melanie Bahlo. Comparison of clustering tools in R for medium-sized 10x Genomics single-cell RNA-sequencing data. *F1000Research*, 2018. doi: 10.12688/f1000research.15809.1.

Pierre-Luc Germain, Anthony Sonrel, and Mark D Robinson. pipeComp, a general framework for the evaluation of computational pipelines, reveals performant single-cell RNA-seq preprocessing tools. *bioRxiv*, 2020. doi: 10.1101/2020.02.02.930578.

Christoph Hafemeister and Rahul Satija. Normalization and variance stabilization of single-cell RNA-seq data using regularized negative binomial regression. *Genome Biology*, (20), 2019. doi: 10.1186/s13059-019-1874-1.

Laleh Haghverdi, Aaron T.L. Lun, Michael D. Morgan, and John C. Marioni. Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. *Nature Biotechnology*, 2018. doi: 10.1038/nbt.4091.

Xiaoping Han, Haide Chen, Daosheng Huang, Huidong Chen, Lijiang Fei, Chen Cheng, He Huang, Guo Cheng Yuan, and Guoji Guo. Mapping human pluripotent stem cell differentiation pathways using high throughput single-cell RNA-sequencing. *Genome Biology*, 2018. doi: 10.1186/s13059-018-1426-0.

Graham Heimberg, Rajat Bhatnagar, Hana El-Samad, and Matt Thomson. Low Dimensionality in Gene Expression Data Enables the Accurate Extraction of Transcriptional Programs from Shallow Sequencing. *Cell Systems*, 2016. doi: 10.1016/j.cels.2016.04.001.

Byungjin Hwang, Ji Hyun Lee, and Duhee Bang. Single-cell RNA sequencing technologies and bioinformatics pipelines. *Experimental and Molecular Medicine*, 50(96), 2018. doi: 10.1038/s12276-018-0071-8.

Vladimir Yu Kiselev, Tallulah S. Andrews, and Martin Hemberg. Challenges in unsupervised clustering of single-cell RNA-seq data. *Nature Reviews Genetics*, 2019. doi: 10.1038/s41576-018-0088-9.

Allon M. Klein, Linas Mazutis, Ilke Akartuna, Naren Tallapragada, Adrian Veres, Victor Li, Leonid Peshkin, David A. Weitz, and Marc W. Kirschner. Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. *Cell*, 2015. doi: 10.1016/j.cell.2015.04.044.

Aaron T. L. Lun, Karsten Bach, and John C. Marioni. Pooling across cells to normalize single-cell RNA sequencing data with many zero counts. *Genome Biology*, 17(1):75, 2016. doi: 10.1186/s13059-016-0947-7.

Atefeh Lafzi, Catia Moutinho, Simone Picelli, and Holger Heyn. Tutorial: guidelines for the experimental design of single-cell RNA sequencing studies. *Nature Protocols*, 13(December), 2018. doi: 10.1038/s41596-018-0073-y.

David Lähnemann, Johannes Köster, Ewa Szczurek, Davis J. McCarthy, Stephanie C. Hicks, Mark D. Robinson, Catalina A. Vallejos, Kieran R. Campbell, Niko Beerenwinkel, Ahmed Mahfouz, Luca Pinello, Pavel Skums, Alexandros Stamatakis, Camille Stephan Otto Attolini, Samuel Aparicio, Jasmijn Baaijens, Marleen Balvert, Buys de Barbanson, Antonio Cappuccio, Giacomo Corleone, Bas E. Dutilh, Maria Florescu, Victor Guryev, Rens Holmer, Katharina Jahn, Thamar Jessurun Lobo, Emma M. Keizer, Indu Khatri, Szymon M. Kielbasa, Jan O. Korbel, Alexey M. Kozlov, Tzu Hao Kuo, Boudewijn P.F. Lelieveldt, Ion I. Mandoiu, John C. Marioni, Tobias Marschall, Felix Mölder, Amir Niknejad, Lukasz Raczkowski, Marcel Reinders, Jeroen de Ridder, Antoine Emmanuel Saliba, Antonios Somarakis, Oliver Stegle, Fabian J. Theis, Huan Yang, Alex Zelikovsky, Alice C. McHardy, Benjamin J. Raphael, Sohrab P. Shah, and Alexander Schönhuth. Eleven grand challenges in single-cell data science. *Genome Biology*, 21, 2020. ISSN 1474760X. doi: 10.1186/s13059-020-1926-6.

Nicholas Lytal, Di Ran, and Lingling An. Normalization Methods on Single-Cell RNA-seq Data: An Empirical Survey. *Frontiers in Genetics*, 11, 2020. doi: 10.3389/fgene.2020.00041.

Páll Melsted, Vasilis Ntranos, Lior Pachter, and Inanc Birol. The barcode, UMI, set format and BUStools. *Bioinformatics*, 2019. ISSN 14602059. doi: 10.1093/bioinformatics/btz279.

Vilas Menon. Clustering single cells: A review of approaches on high-and low-depth single-cell RNA-seq data. *Briefings in Functional Genomics*, 2018. doi: 10.1093/bfgp/elx044.

Elisabetta Mereu, Atefeh Lafzi, Catia Moutinho, Christoph Ziegenhain, Davis J. MacCarthy, Adrian Alvarez, Eduard Batlle, Sagar, Dominic Grün, Julia K. Lau, Stéphane C. Boutet, Chad Sanada, Aik Ooi, Robert C. Jones, Kelly Kaihara, Chris Brampton, Yasha Talaga, Yohei Sasagawa, Kaori Tanaka, Tetsutaro Hayashi, Itoshi Nikaido, Cornelius Fischer, Sascha Sauer, Timo Trefzer, Christian Conrad, Xian Adiconis, Lan T. Nguyen, Aviv Regev, Joshua Z. Levin, Swati Parekh, Aleksandar Janjic, Lucas E. Wange, Johannes W. Bagnoli, Wolfgang Enard, Marta Gut, Rickard Sandberg, Ivo Gut, Oliver Stegle, and Holger Heyn. Benchmarking Single-Cell RNA Sequencing Protocols for Cell Atlas Projects. *bioRxiv*, 2019. doi: 10.1101/630087.

Efthymia Papalexi and Rahul Satija. Single-cell RNA sequencing to explore immune cell heterogeneity. *Nature Reviews Immunology*, 2018. doi: 10.1038/nri.2017.76.

Wouter Saelens, Robrecht Cannoodt, Helena Todorov, and Yvan Saeys. A comparison of single-cell trajectory inference methods. *Nature Biotechnology*, 2019. doi: 10.1038/s41587-019-0071-9.

Antoine Emmanuel Saliba, Alexander J. Westermann, Stanislaw A. Gorski, and Jörg Vogel. Single-cell RNA-seq: Advances and future challenges. *Nucleic Acids Research*, 2014. doi: 10.1093/nar/gku555.

Charlotte Soneson and Mark D. Robinson. Bias, robustness and scalability in single-cell differential expression analysis. *Nature Methods*, 2018. doi: 10.1038/nmeth.4612.

Avi Srivastava, Laraib Malik, Tom Smith, Ian Sudbery, and Rob Patro. Alevin efficiently estimates accurate gene abundances from dscRNA-seq data. *Genome Biology*, 2019. ISSN 1474760X. doi: 10.1186/s13059-019-1670-y.

Kelly Street, Davide Risso, Russell B. Fletcher, Diya Das, John Ngai, Nir Yosef, Elizabeth Purdom, and Sandrine Dudoit. Slingshot: Cell lineage and pseudotime inference for single-cell transcriptomics. *BMC Genomics*, 2018. doi: 10.1186/s12864-018-4772-0.

Tim Stuart, Andrew Butler, Paul Hoffman, Christoph Hafemeister, Efthymia Papalexi, William M. Mauck, Yuhan Hao, Marlon Stoeckius, Peter Smibert, and Rahul Satija. Comprehensive Integration of Single-Cell Data. *Cell*, 2019. doi: 10.1016/j.cell.2019.05.031.

Shiquan Sun, Jiaqiang Zhu, Ying Ma, and Xiang Zhou. Accuracy, robustness and scalability of dimensionality reduction methods for single-cell RNA-seq analysis. *Genome Biology*, 2019. doi: 10.1186/s13059-019-1898-6.

Valentine Svensson, Roser Vento-Tormo, and Sarah A. Teichmann. Exponential scaling of single-cell RNA-seq in the past decade. *Nature Protocols*, 2018. doi: 10.1038/nprot.2017.149.

Luyi Tian, Shian Su, Xueyi Dong, Daniela Amann-Zalcenstein, Christine Biben, Azadeh Seidi, Douglas J. Hilton, Shalin H. Naik, and Matthew E. Ritchie. scPipe: A flexible R/Bioconductor preprocessing pipeline for single-cell RNA-sequencing data. *PLoS Computational Biology*, 2018. doi: 10.1371/journal.pcbi.1006361.

Laurens Van Der Maaten, Geoffrey Hinton, and Geoffrey Hinton van der Maaten. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008. doi: 10.1007/s10479-011-0841-3.

Yue J. Wang and Klaus H. Kaestner. Single-Cell RNA-Seq of the Pancreatic Islets—a Promise Not yet Fulfilled? *Cell Metabolism*, 2018. ISSN 15504131. doi: 10.1016/j.cmet.2018.11.016.

Shiyi Yang, Sean E. Corbett, Yusuke Koga, Zhe Wang, W. Evan Johnson, Masanao Yajima, and Joshua D. Campbell. Decontamination of ambient RNA in single-cell RNA-seq with DecontX. *bioRxiv*, 2019. doi: 10.1101/704015.

Matthew D Young and Sam Behjati. SoupX removes ambient RNA contamination from droplet based single cell RNA sequencing data. *GigaScience*, 2020. doi: 10.1101/303727.

Xiannian Zhang, Tianqi Li, Feng Liu, Yaqi Chen, Jiacheng Yao, Zeyao Li, Yanyi Huang, and Jianbin Wang. Comparative Analysis of Droplet-Based Ultra-High-Throughput Single-Cell RNA-Seq Systems. *Molecular Cell*, 2019. doi: 10.1016/j.molcel.2018.10.020.

Chenxu Zhu, Sebastian Preissl, and Bing Ren. Single-cell multimodal omics: the power of many. *Nature Methods*, 2020. doi: 10.1038/s41592-019-0691-5.

Christoph Ziegenhain, Beate Vieth, Swati Parekh, Björn Reinius, Amy Guillaumet-Adkins, Martha Smets, Heinrich Leonhardt, Holger Heyn, Ines Hellmann, and Wolfgang Enard. Comparative Analysis of Single-Cell RNA Sequencing Methods. *Molecular Cell*, 2017. doi: 10.1016/j.molcel.2017.01.023.

Rapolas Zilionis, Juozas Nainys, Adrian Veres, Virginia Savova, David Zemmour, Allon M. Klein, and Linas Mazutis. Single-cell barcoding and sequencing using droplet microfluidics. *Nature Protocols*, 2017. doi: 10.1038/nprot.2016.154.