Single cell RNA seq

Vasilis Ntranos
UCSF

BMS 225A — Lecture 03 — October 15, 2025

Today's session

Week 3: Lecture 3

- Intro to single-cell RNA-seq
  - purpose and usecases
  - overview of the technology

- Computational workflow
  - from sequencing reads to a gene expression matix
  - sources of error and quality control
  - overview of key steps in the analysis
    - normalization, dimensionality reduction, clustering, etc.

- Single cell RNA seq analysis with **scanpy**
  - go over the main scanpy functions
  - we'll use these in the workshop

Resources (coding)

Up to now

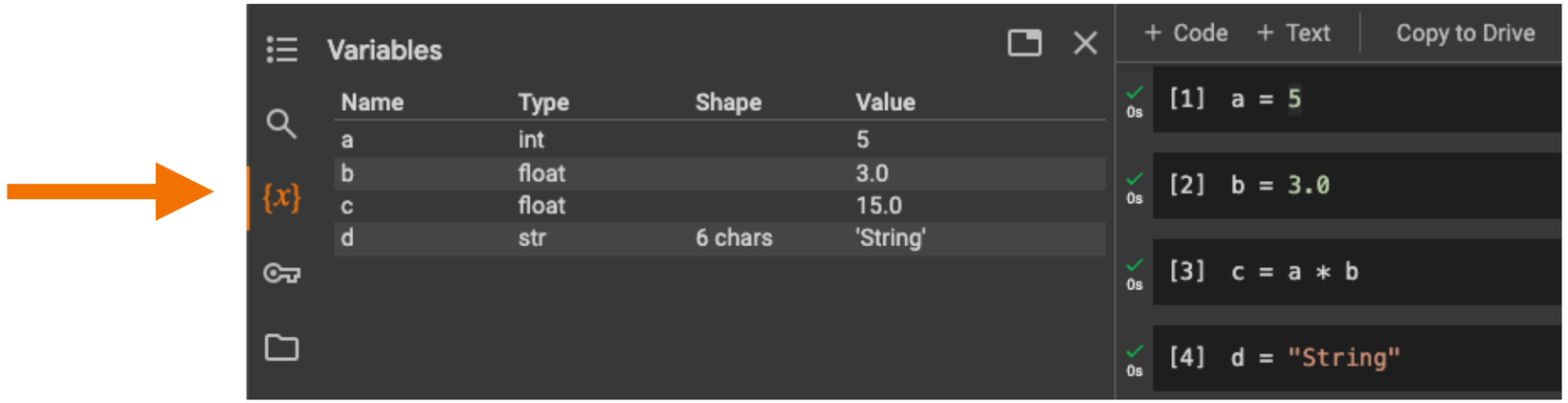
1. **Python**
   - Book: Automate the Boring Stuff with Python (Chapters 1 - 5) https://automatetheboringstuff.com/
     - Video Tutorial: Python Basics https://www.youtube.com/watch?v=rfscVS0vtbw

2. **Pandas**
   - intro tutorials https://pandas.pydata.org/docs/getting_started/intro_tutorials/index.html
     - user guide https://pandas.pydata.org/docs/user_guide/index.html
     - comparison with R https://pandas.pydata.org/docs/getting_started/comparison/comparison_with_r.html
   - ✦ Video Tutorial: Pandas: https://www.youtube.com/playlist?list=PL-osiE80TeTsWmV9i9c58mdDCSskIFdDS

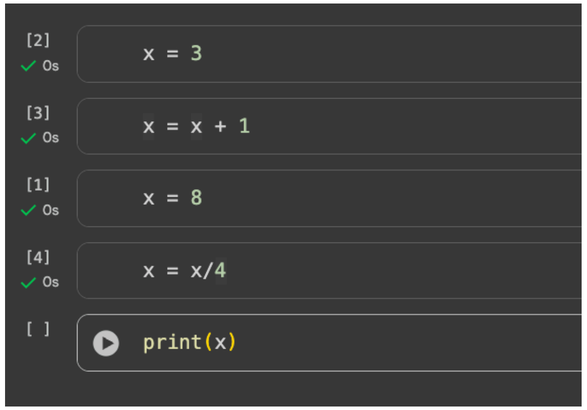
3. **BMS 225A Colab Notebooks**
   - Python Basics https://gist.github.com/vasilisNt/438d6e8c1b187800e56f219e211e0950
   - Pandas Practice https://gist.github.com/vasilisNt/814f5aabdfcb8d23f46cecf149f9397a
   - Pandas Workshop https://gist.github.com/vasilisNt/29ca0d9ca5418e8bb6b11dfcce8c0285

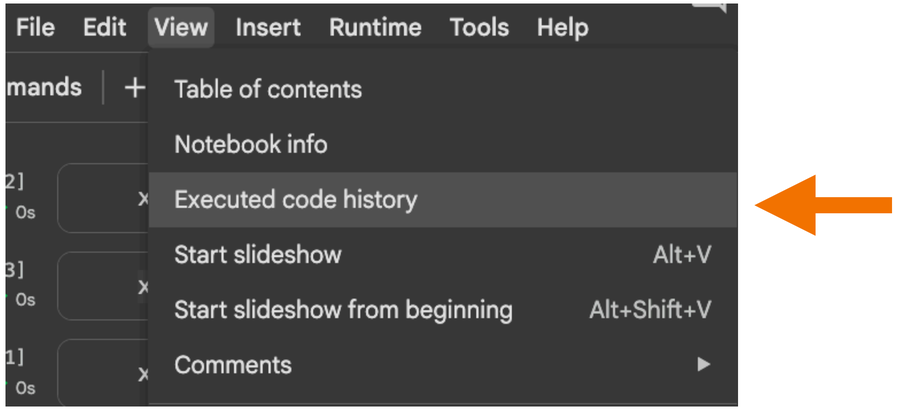
Colab notebook tips

How can I see the variables I created on my Google Colab session?



How can I see the the order of executed commands?





Inside a Pandas DataFrame

DataFrame:

	patient_id	age	gender	height	weight	smoke	alco	active
0	43005	15375	Male	169	68.0	False	False	False
1	76878	21712	Female	182	91.0	True	False	True
2	68983	22699	Male	166	72.0	False	False	False
3	61210	21163	Female	172	65.0	True	True	True
4	68948	19650	Male	168	72.0	False	False	True

stored in a variable called patient_data

patient_data

variables (attributes)

values

43005	15375	Male	169	68.0	False	False	False
76878	21712	Female	182	91.0	True	False	True
68983	22699	Male	166	72.0	False	False	False
61210	21163	Female	172	65.0	True	True	True
68948	19650	Male	168	72.0	False	False	True

index

0
1
2
3
4

columns

patient_id	age	gender	height	weight	smoke	alco	active

functions (methods)

reset_index( ) rename( ) drop_duplicates( )

sum( ) mean( ) pivot_table( ) …

access everything with a .

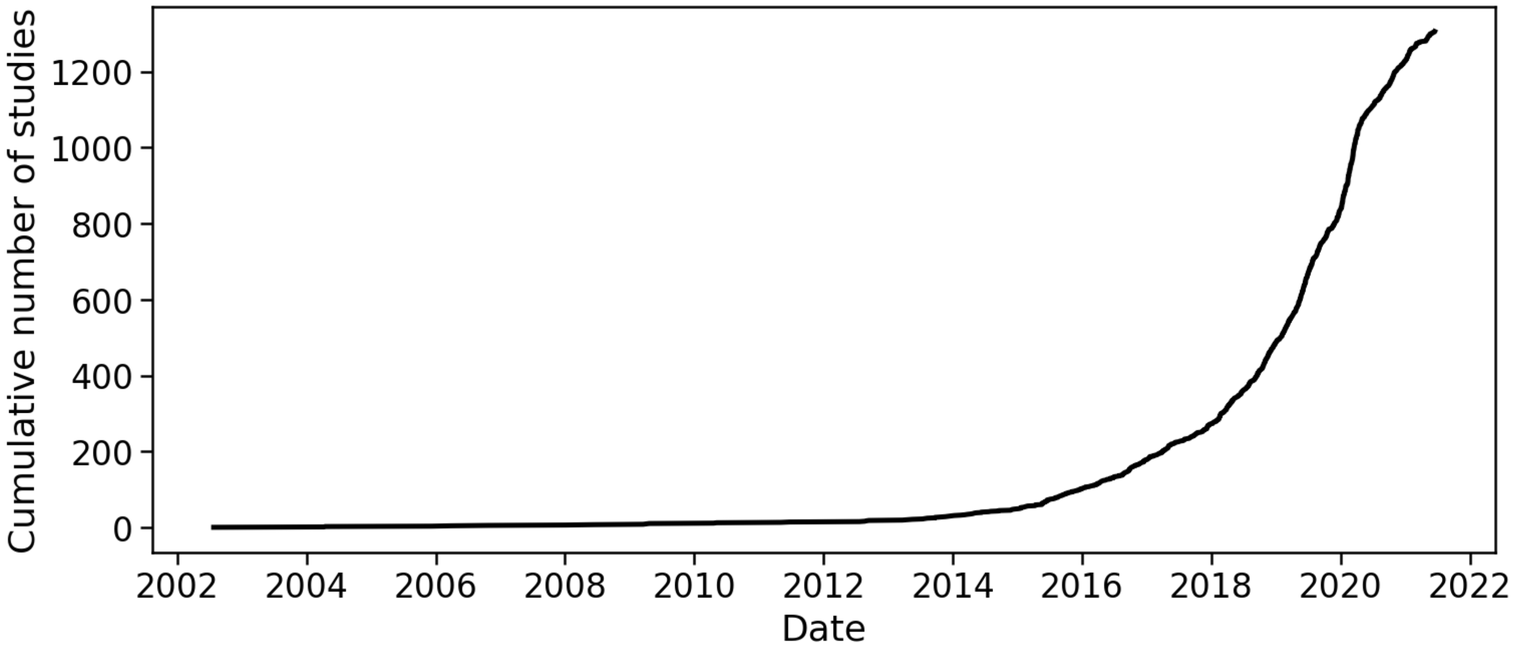
patient_data.columns

patient_data.values

patient_data.drop_duplicates( )

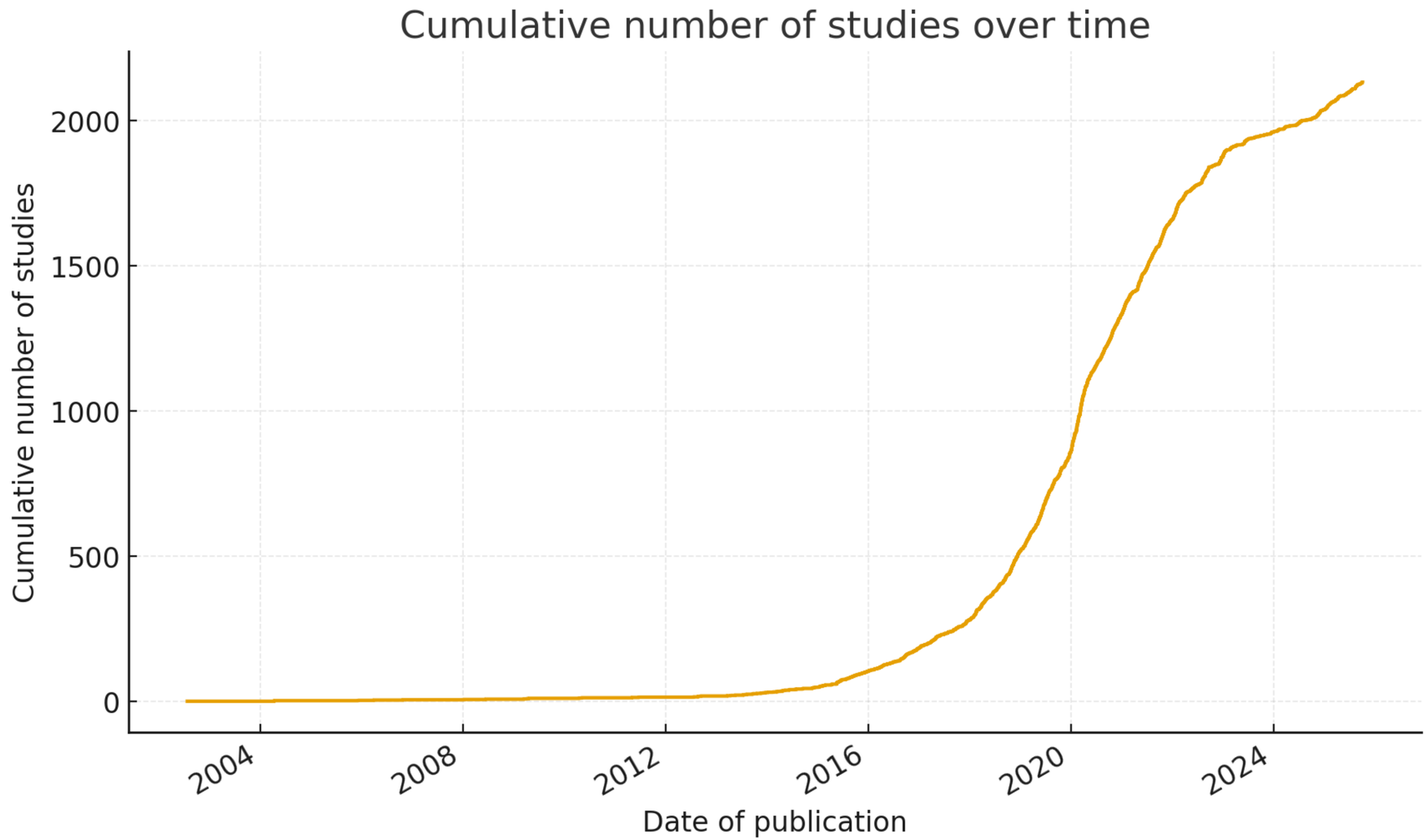
patient_data.reset_index( )

Single cell rna seq



Svensson, et al., A curated database reveals trends in single-cell transcriptomics, *Database*, 2020

https://github.com/pachterlab/kallistobustools/blob/master/docs/tutorials/scRNA-seq_intro/python/scRNA-seq_intro.ipynb

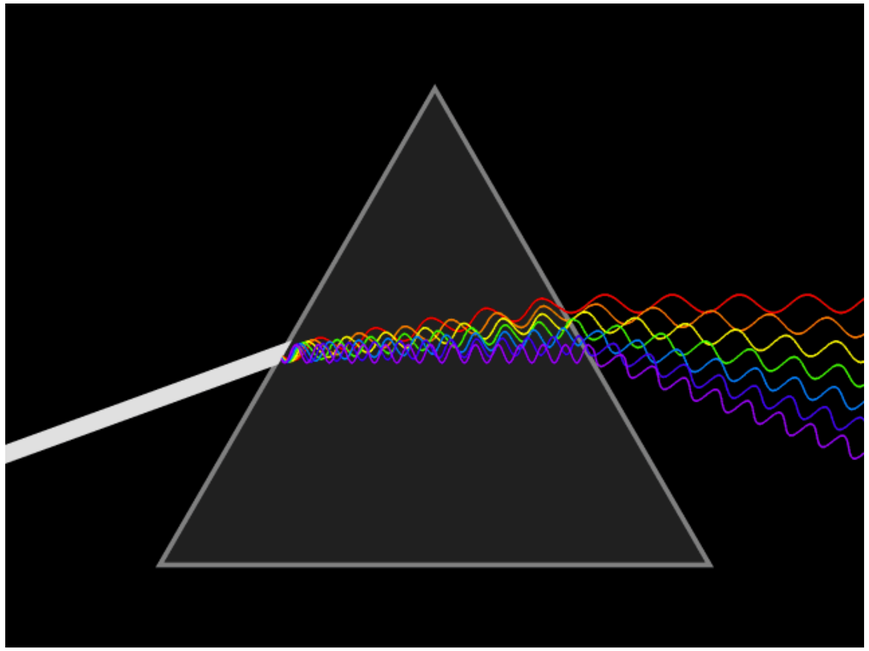


Svensson, et al., A curated database reveals trends in single-cell transcriptomics, *Database*, 2020

https://github.com/pachterlab/kallistobustools/blob/master/docs/tutorials/scRNA-seq_intro/python/scRNA-seq_intro.ipynb

The purpose of single-cell RNA-seq

- Decompose tissue or organ expression into its constituent parts
- Identify the different types of cells that comprise tissues and organs
- Examine the molecular biology of cells via their expression signatures
- Determine differentiation trajectories of cells
- Develop expression biomarkers for disease states

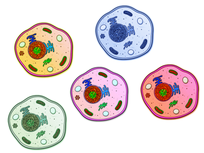


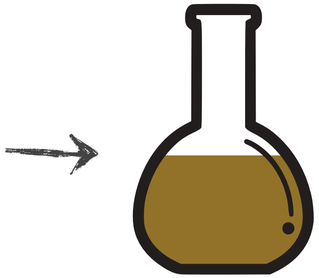
https://github.com/pachterlab/BI-BE-CS-183-2023/

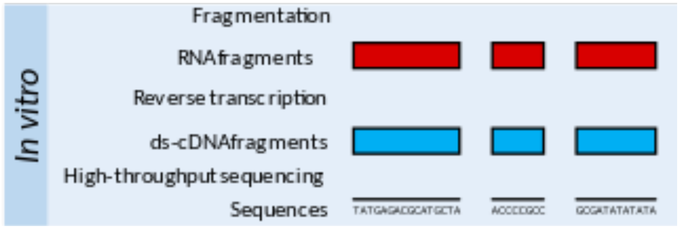
Introduction to Computational Biology and Bioinformatics

(Bulk) RNA-Seq

- Overview

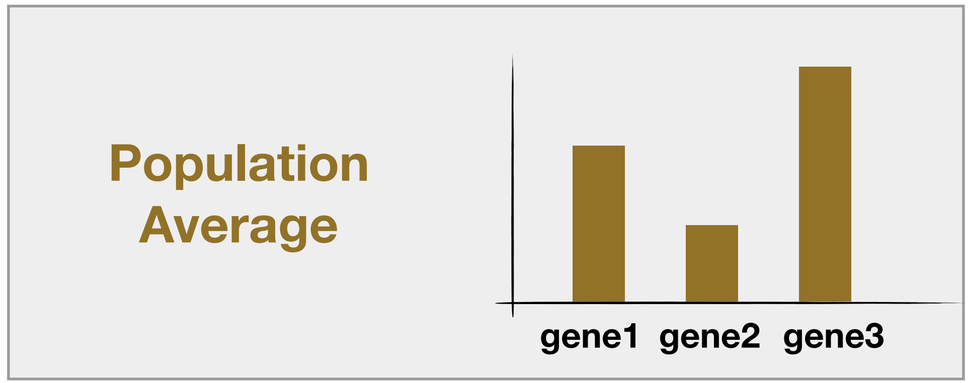






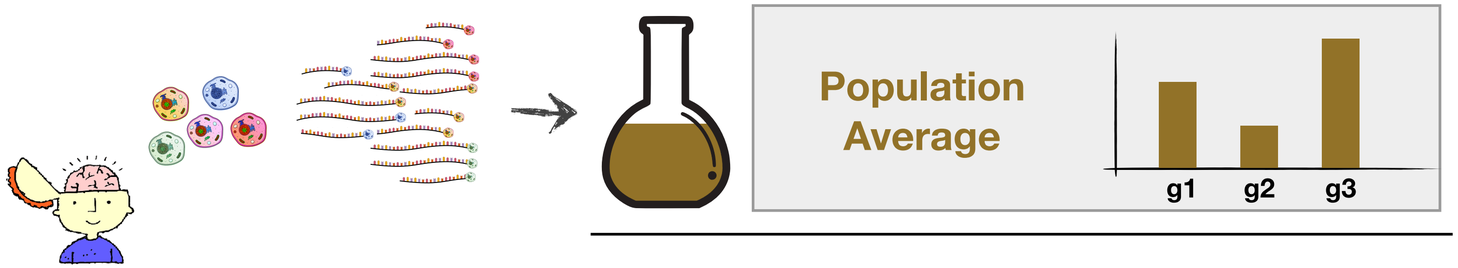
```
>
  read_alignment

  quantification
```

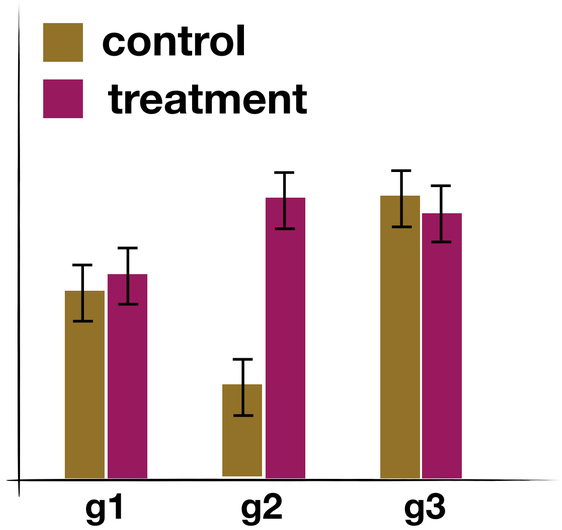


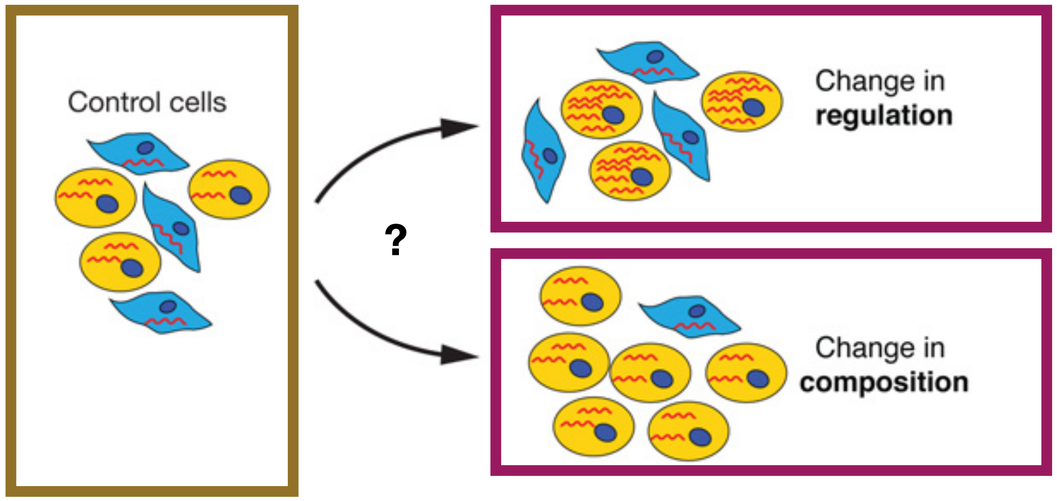
From bulk to single cell measurements

- Overview



- Limitations of bulk measurements

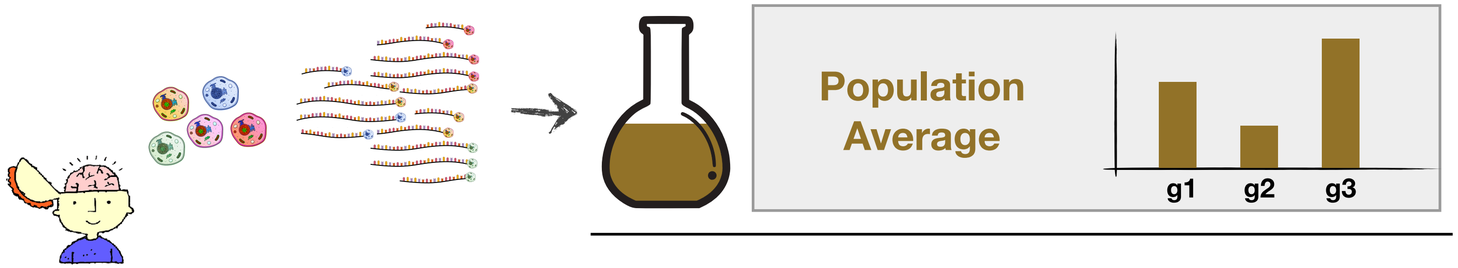




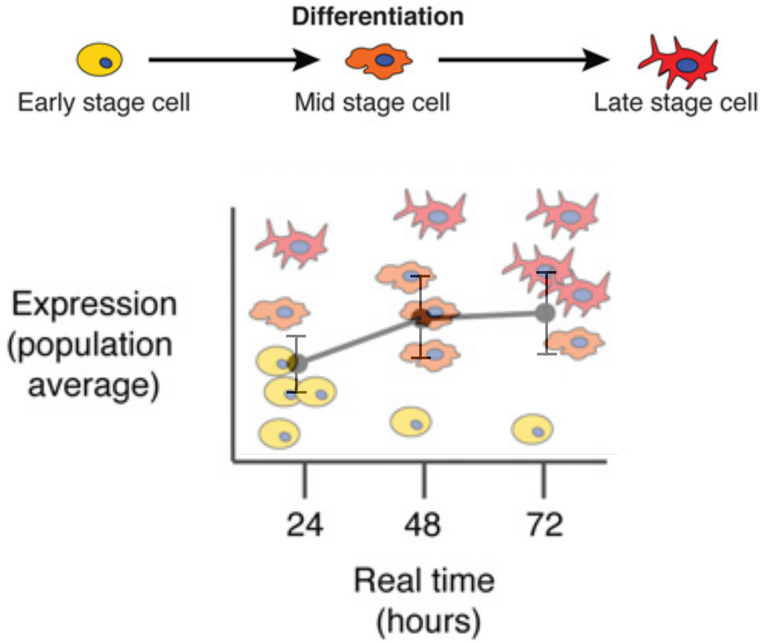
Trapnell, et al., 2015 10.1101/gr.190595.115

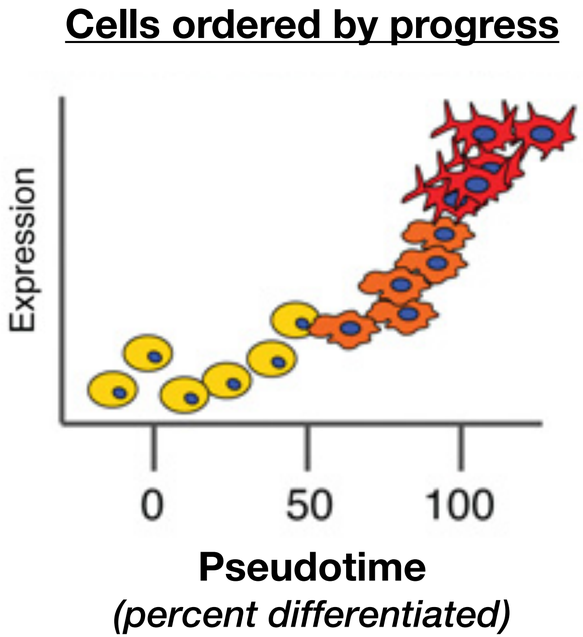
From bulk to single cell measurements

- Overview



- Limitations of bulk measurements



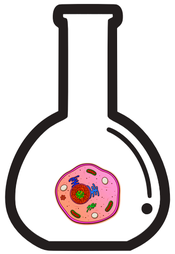


Trapnell, et al., 2015 10.1101/gr.190595.115

From bulk to single cell measurements

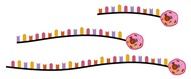
- Overview

- **Single-Cell** RNA-Seq (Tang *et al*. 2009)

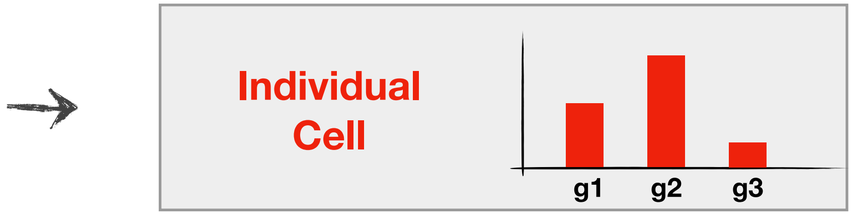


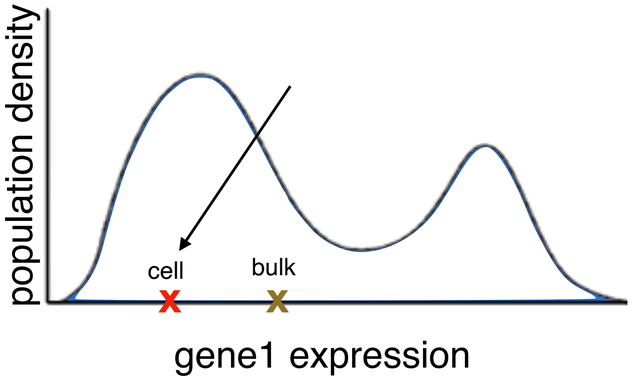
one cell

- **chemistry challenge**: small amount of starting material



- one cell is (obviously) not enough

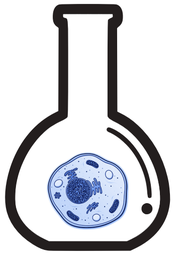




From bulk to single cell measurements

- Overview

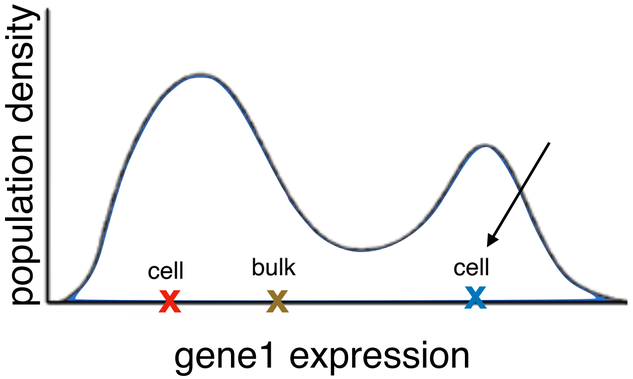
- **Single-Cell** RNA-Seq (Tang *et al*. 2009)



one cell

- **chemistry challenge**: small amount of starting material

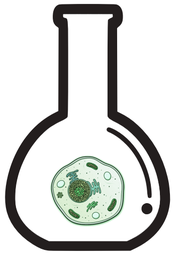
- one cell is (obviously) not enough



From bulk to single cell measurements

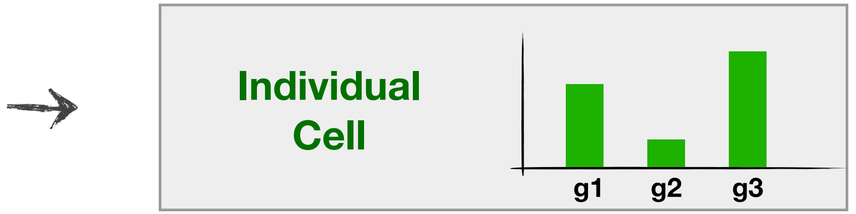
- Overview

- **Single-Cell** RNA-Seq (Tang *et al*. 2009)



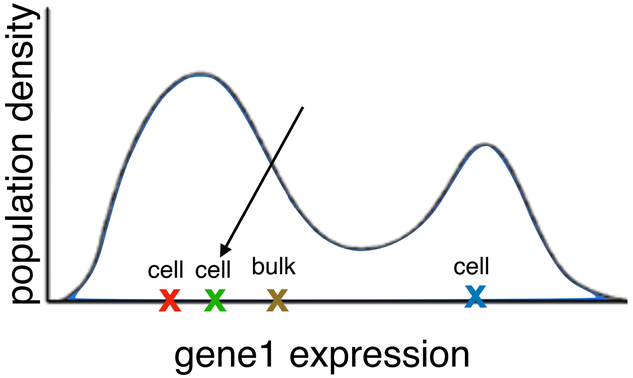
one cell

- chemistry challenge: small amount of starting material



- one cell is (obviously) not enough

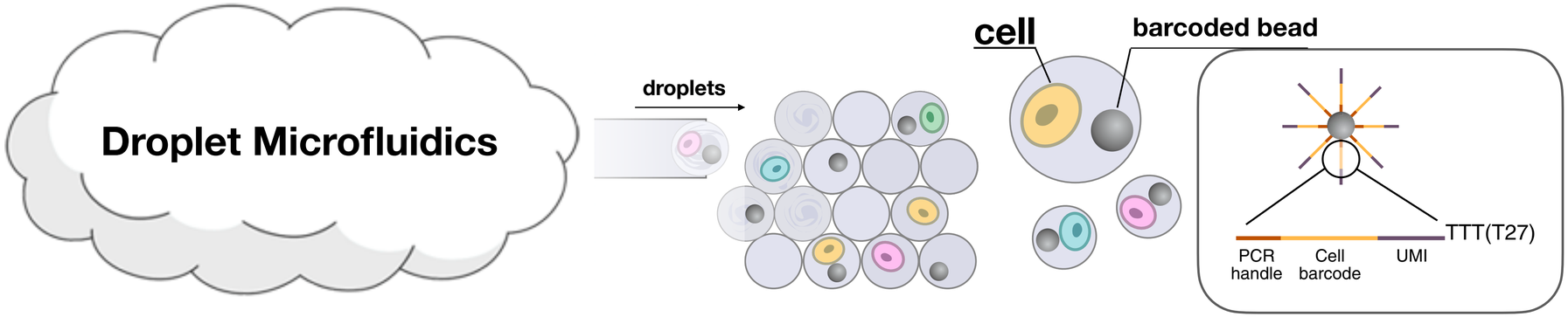
time consuming
very high cost per cell



Single-cell RNA Seq | Droplet microfluidics

Droplet microfluidics

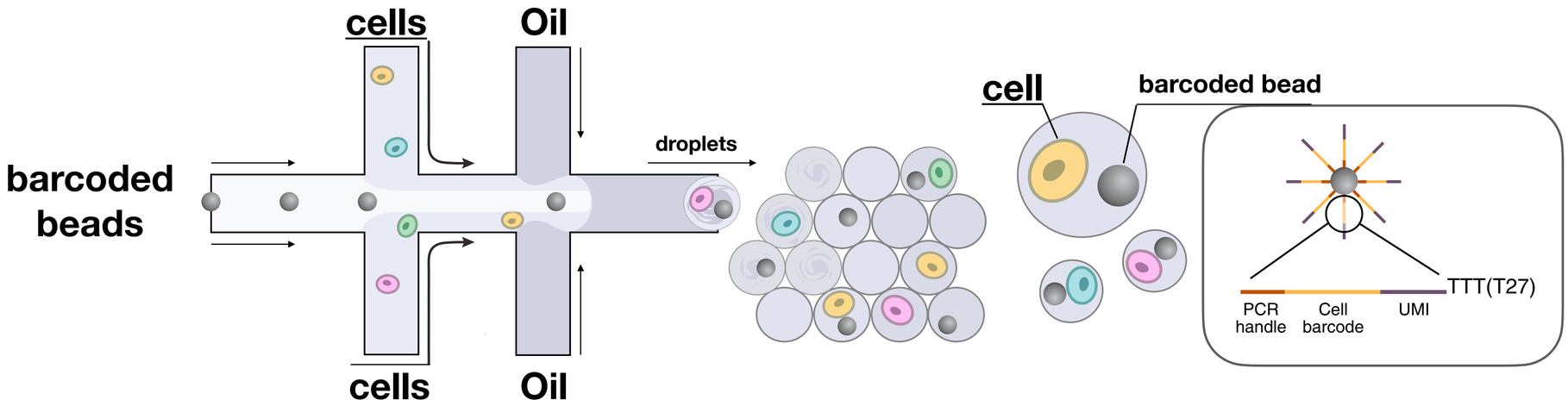
- Drop-Seq (Makosko et al., 2015)
- InDrop (Klein et al., 2015)
- 10x Genomics (Zheng et al., 2017)
  - **Scale up** number of cells (capture cells inside nanoliter droplets)
    - key ingredients: **cell barcodes** and **unique molecular identifiers (UMIs)** [Islam et al., 2012]



Single-cell RNA Seq | Droplet microfluidics

Droplet microfluidics

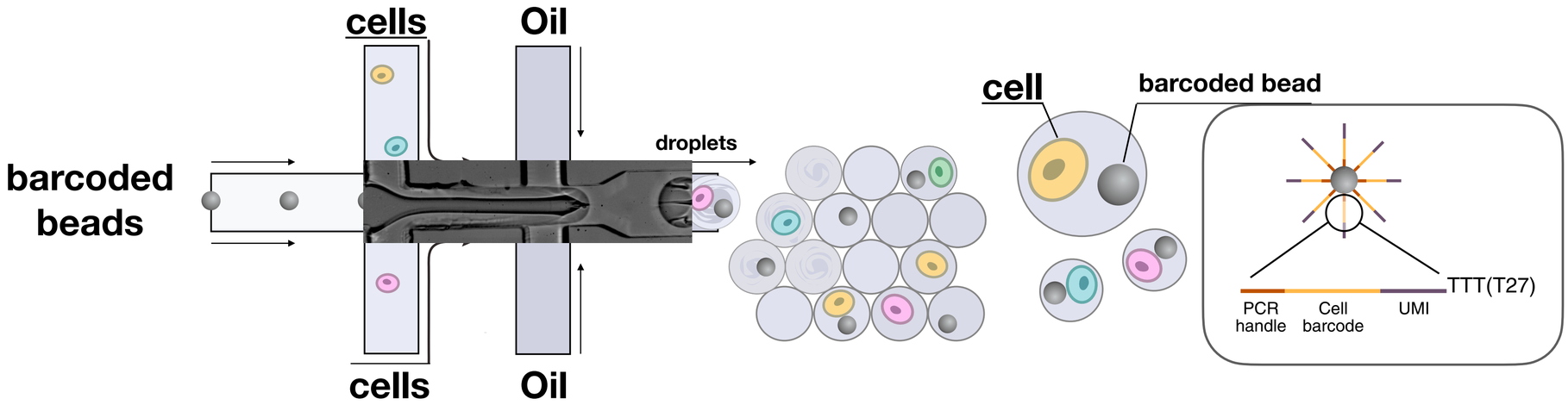
- Drop-Seq (Makosko et al., 2015)
- InDrop (Klein et al., 2015)
- 10x Genomics (Zheng et al., 2017)
  - **Scale up** number of cells (capture cells inside nanoliter droplets)
    - key ingredients: **cell barcodes** and **unique molecular identifiers (UMIs)** [Islam et al., 2012]



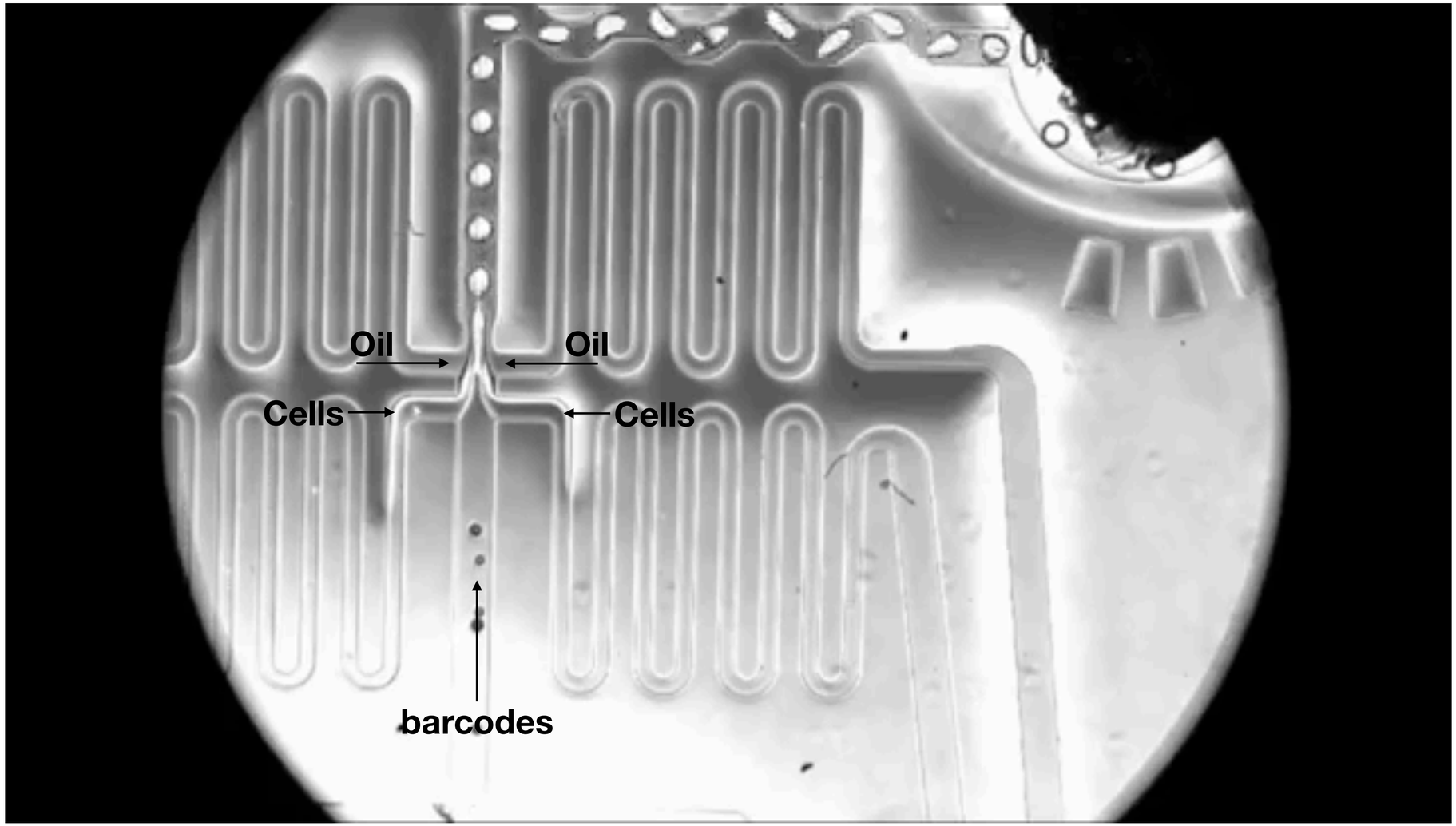
Single-cell RNA Seq | Droplet microfluidics

Droplet microfluidics

- Drop-Seq (Makosko et al., 2015)
- InDrop (Klein et al., 2015)
- 10x Genomics (Zheng et al., 2017)
  - **Scale up** number of cells (capture cells inside nanoliter droplets)
    - key ingredients: **cell barcodes** and **unique molecular identifiers (UMIs)** [Islam et al., 2012]

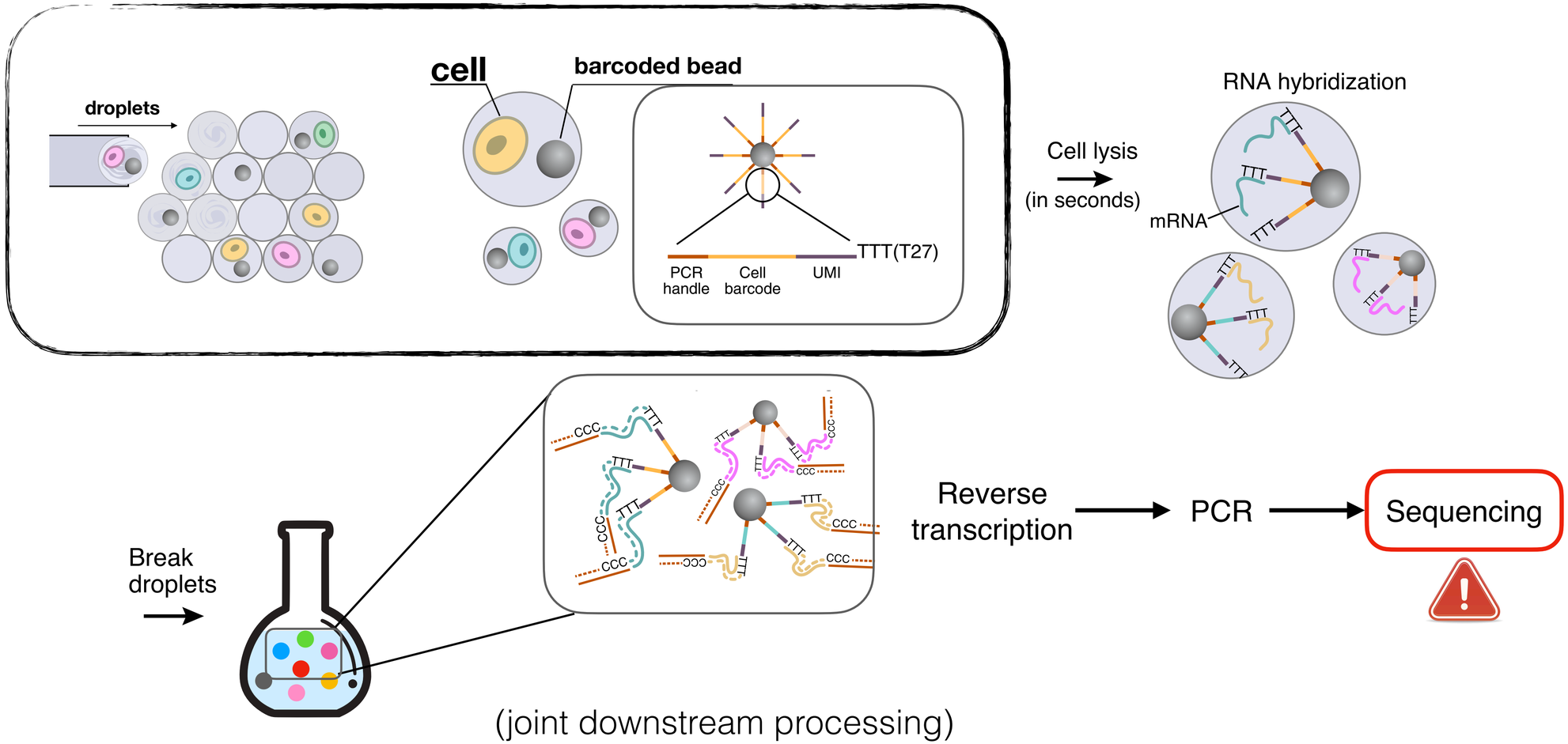


Single-cell RNA Seq | Droplet microfluidics



Single-cell RNA Seq | Droplet microfluidics

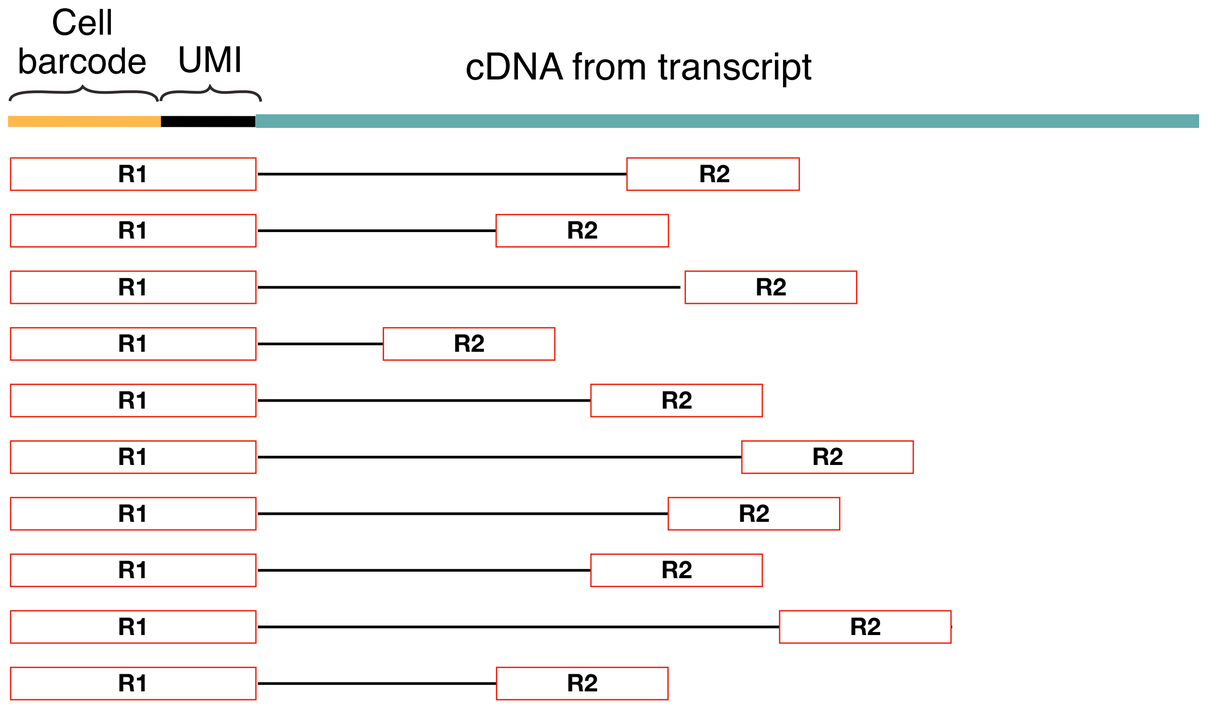
What happens next?



Single-cell RNA Seq | Sequencing

- Sequencing:

Paired-end reads
(3'end-bias)

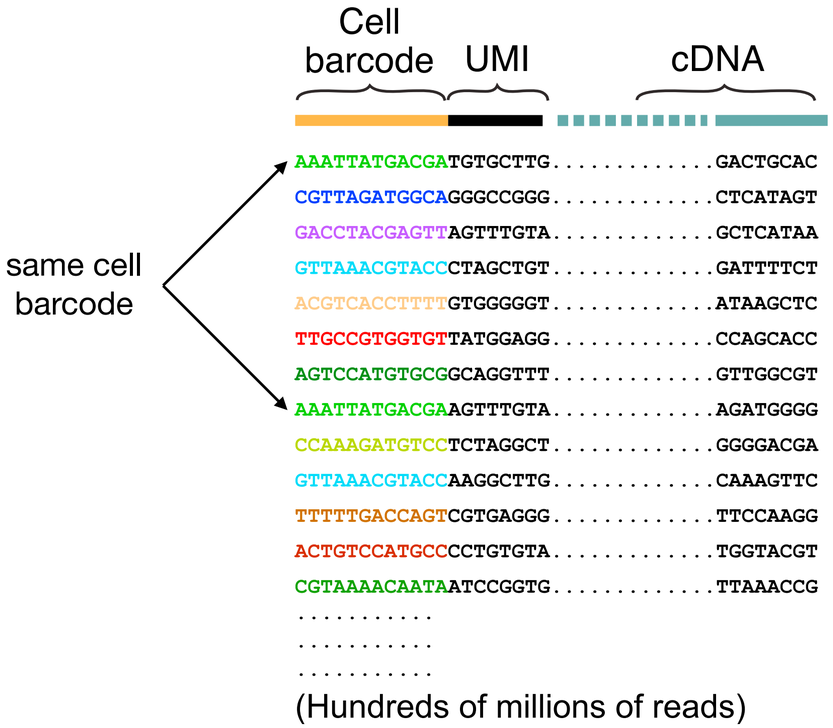


Single-cell RNA Seq | Sequencing

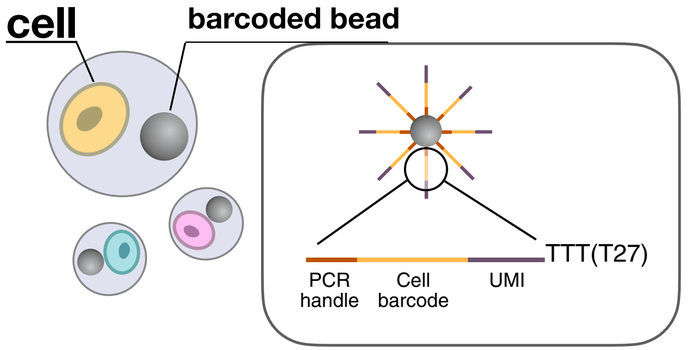
- Sequencing:

Paired-end reads
(3'end-bias)

Single-cell RNA Seq | Computational Workflow



Goal:
Trace reads back to individual cells



Single-cell RNA Seq | Computational Workflow

- This is computationally intensive!
  - Software that takes us from sequenced reads (fastq files) to raw count data:
    - Cell Ranger (10x Genomics)
    - kallisto/BUStools (orders of magnitude faster, all single cell technologies)
    - salmon/Alevin (same)

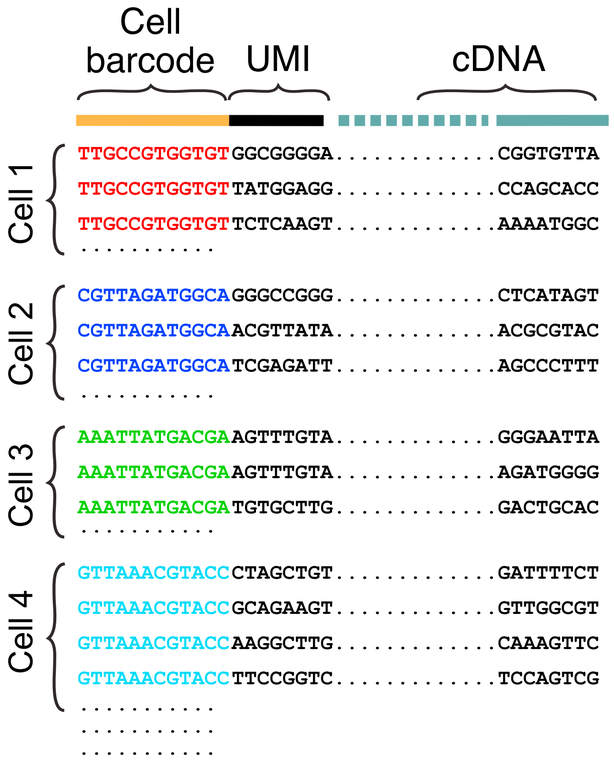
Cell Ranger (unpublished): https://support.10xgenomics.com/single-cell-gene-expression/software/pipelines/latest/using/tutorial_ct

Melsted, P., Booeshaghi, A.S., Liu, L. *et al.* Modular, efficient and constant-memory single-cell RNA-seq preprocessing. *Nat Biotechnol* **39**, 813–818 (2021). https://doi.org/10.1038/s41587-021-00870-2
https://www.kallistobus.tools/

He, D., Zakeri, M., Sarkar, H. *et al.* Alevin-fry unlocks rapid, accurate and memory-frugal quantification of single-cell RNA-seq data. *Nat Methods* **19**, 316–322 (2022). https://doi.org/10.1038/s41592-022-01408-
https://salmon.readthedocs.io/en/latest/alevin.html

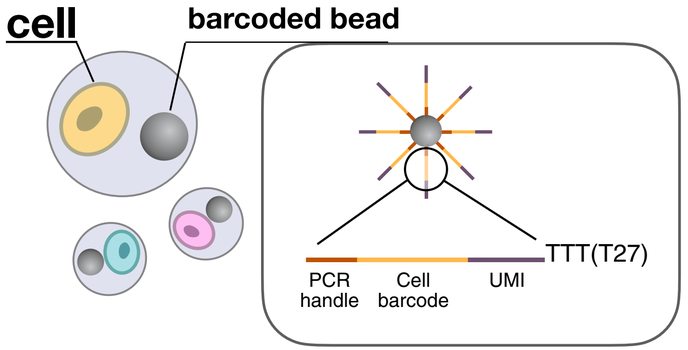
Single-cell RNA Seq | Computational Workflow

- key steps:



(Thousands of cells)

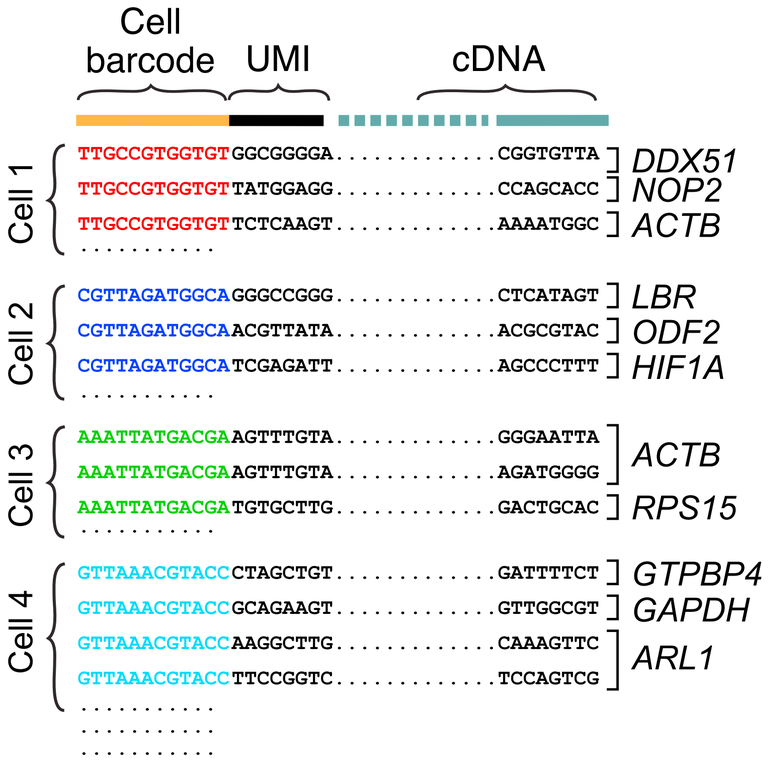
Goal:
Trace reads back to individual cells



1. Group by cell barcode

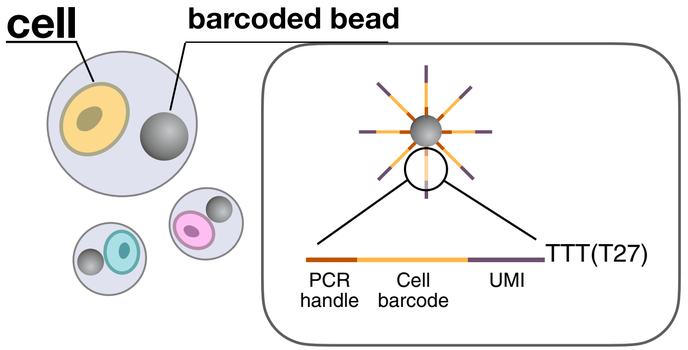
Single-cell RNA Seq | Computational Workflow

- key steps:



(Thousands of cells)

Goal:
Trace reads back to individual cells

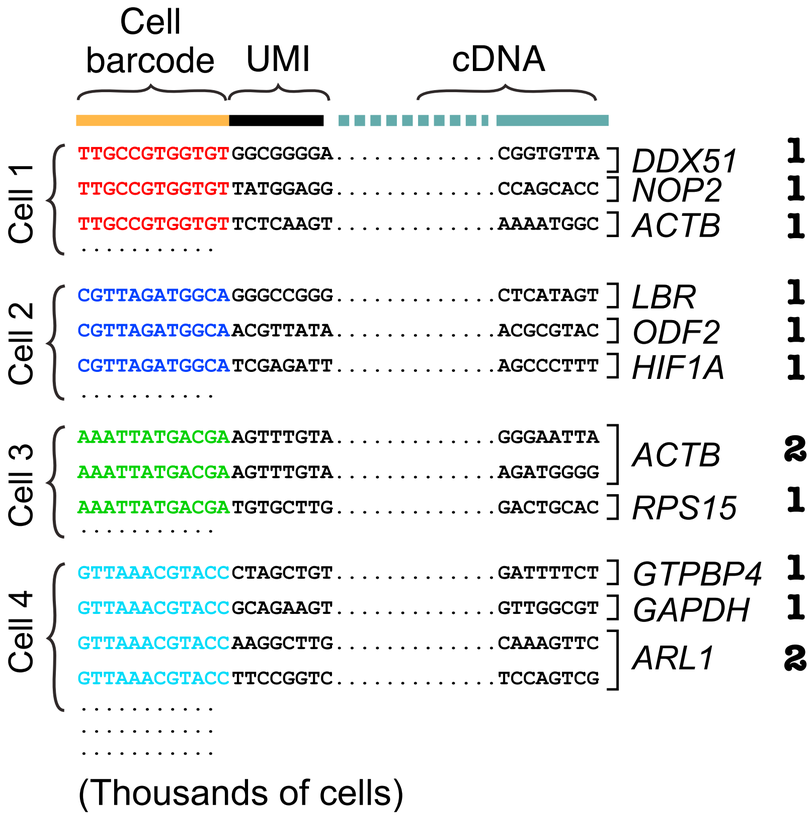


1. Group by cell barcode

2. Align cDNA reads (transcript identity)

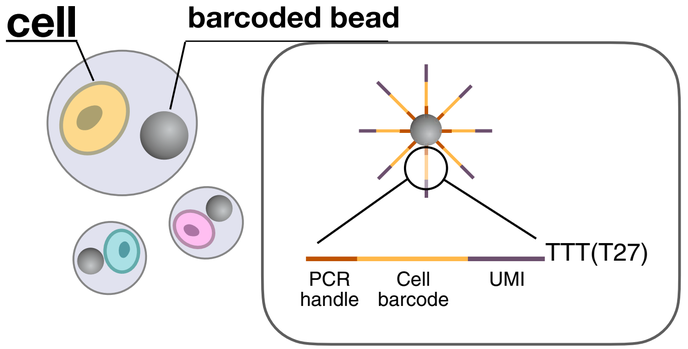
Single-cell RNA Seq | Computational Workflow

- key steps:



(Thousands of cells)

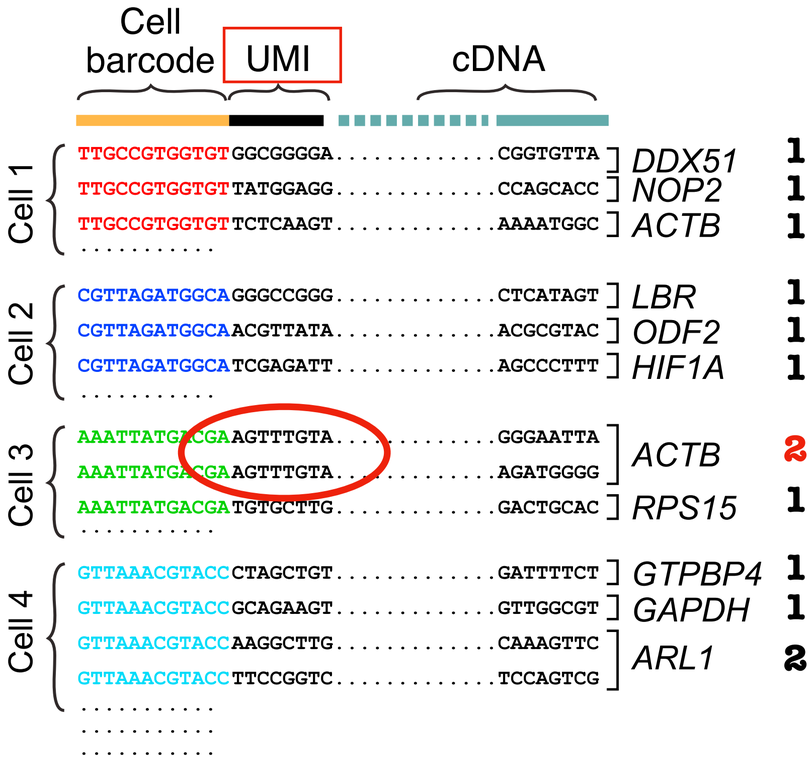
Goal:
Trace reads back to individual cells



1. Group by cell barcode
2. Align cDNA reads (transcript identity)
3. Count reads per cell per gene

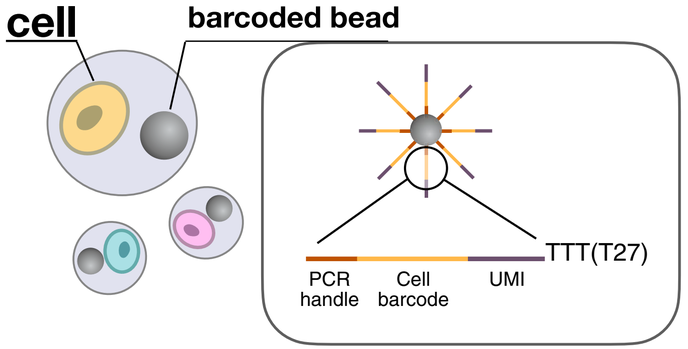
Single-cell RNA Seq | Computational Workflow

- key steps:



(Thousands of cells)

Goal:
Trace reads back to individual cells



1. Group by cell barcode

2. Align cDNA reads (transcript identity)

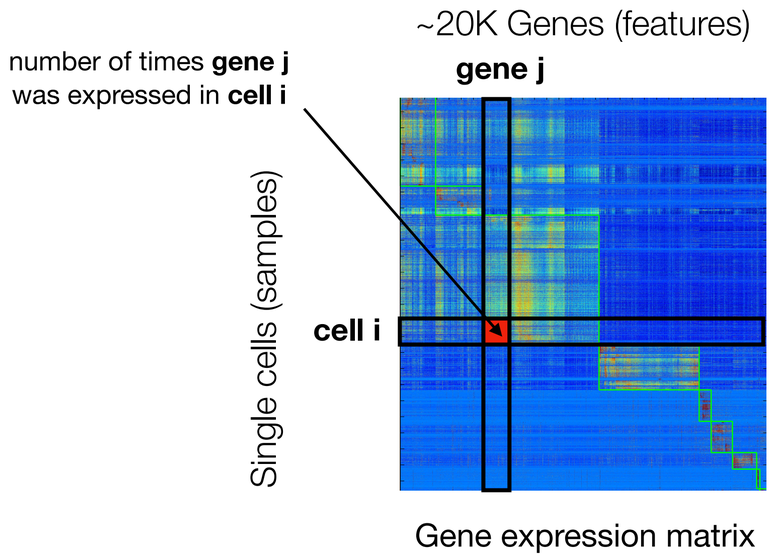
unique molecules

3. Count ~~reads~~ per cell per gene

Single-cell RNA Seq | Computational Workflow

Final output

gene expression matrix



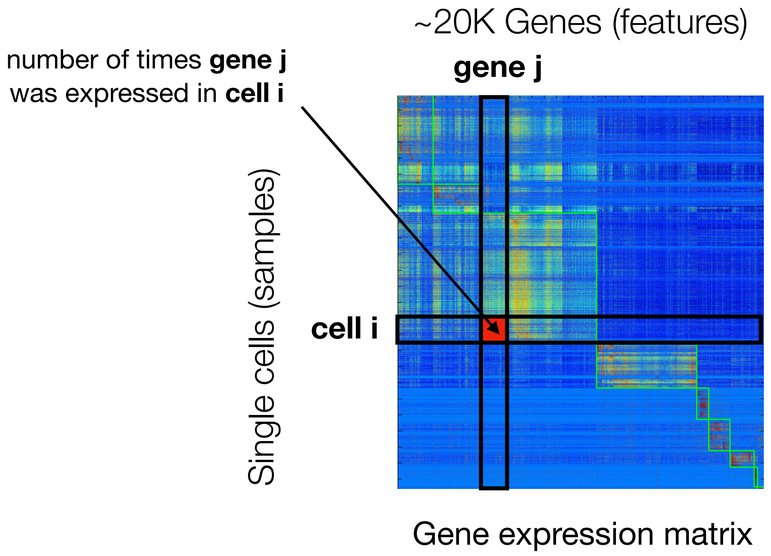
What can go wrong?

Single-cell RNA Seq | Computational Workflow

raw counts / input

~~**Final output**~~

gene expression matrix



Next Steps

Pre-processing and Quality Control

Normalization & Transformation

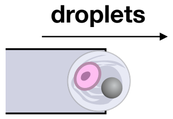
Feature Selection

Dimensionality Reduction & Visualization

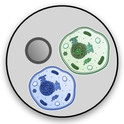
Clustering and Cell type annotation

...

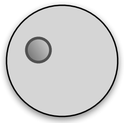
Single-cell RNA Seq | Pre-processing and Quality Control



Single Cell



Doublet



Empty Droplet

Leaky Droplet

Damaged/Dying cell

Variable mRNA capture efficiency

Library not sequenced to saturation
Sequencing errors in cell barcodes
cDNA fragment maps to multiple genes

Cell Ranger Summary

Estimated Number of Cells
11,769

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

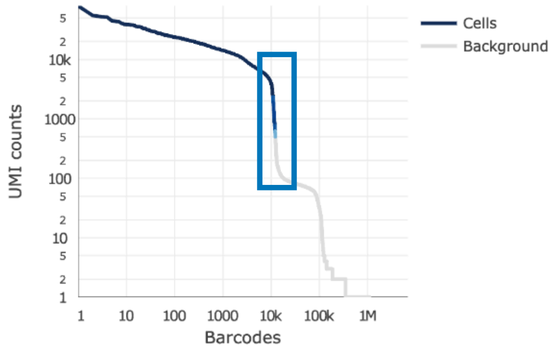
Sequencing

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

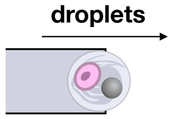


Estimated Number of Cells	11,769
Fraction Reads in Cells	95.1%
Mean Reads per Cell	54,286
Median Genes per Cell	1,906
Total Genes Detected	23,036
Median UMI Counts per Cell	6,521

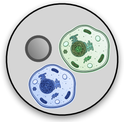
Sample

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

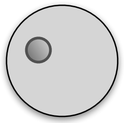
Single-cell RNA Seq | Pre-processing and Quality Control



Single Cell



Doublet



Empty Droplet

Leaky Droplet

Damaged/Dying cell

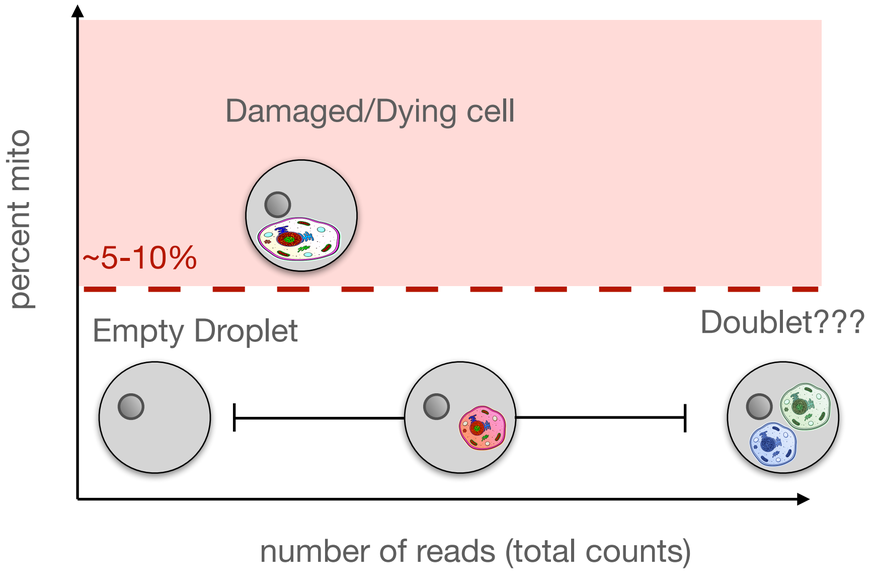
Variable mRNA capture efficiency

Library not sequenced to saturation
Sequencing errors in cell barcodes
cDNA fragment maps to multiple genes

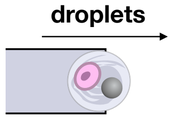
Commonly used QC metrics

For each **cell:**

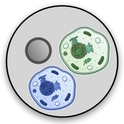
- Number of reads (total counts)
- Number of genes detected
- Percentage of mitochondrial reads



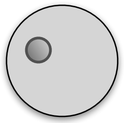
Single-cell RNA Seq | Pre-processing and Quality Control



Single Cell



Doublet



Empty Droplet

Leaky Droplet

Damaged/Dying cell

Variable mRNA capture efficiency

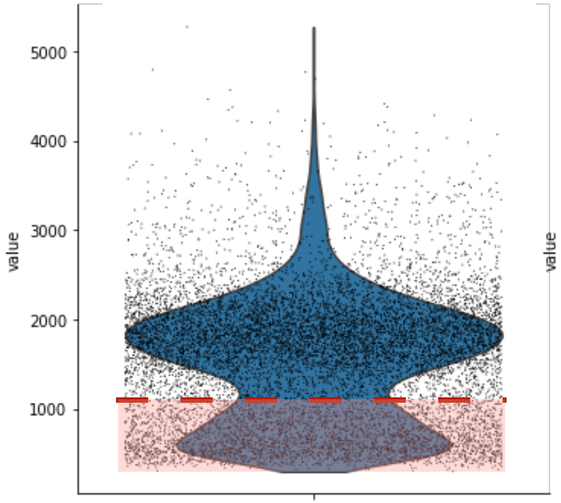
Library not sequenced to saturation
Sequencing errors in cell barcodes
cDNA fragment maps to multiple genes

Commonly used QC metrics

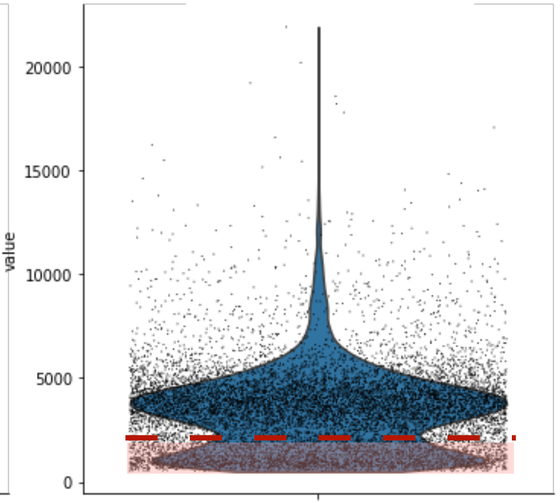
For each **cell:**

- Number of reads (total counts)
- Number of genes detected
- Percentage of mitochondrial reads

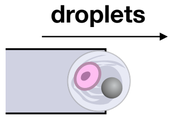
number of genes



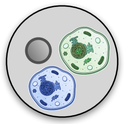
total counts



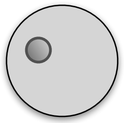
Single-cell RNA Seq | Pre-processing and Quality Control



Single Cell



Doublet



Empty Droplet

Leaky Droplet

Damaged/Dying cell

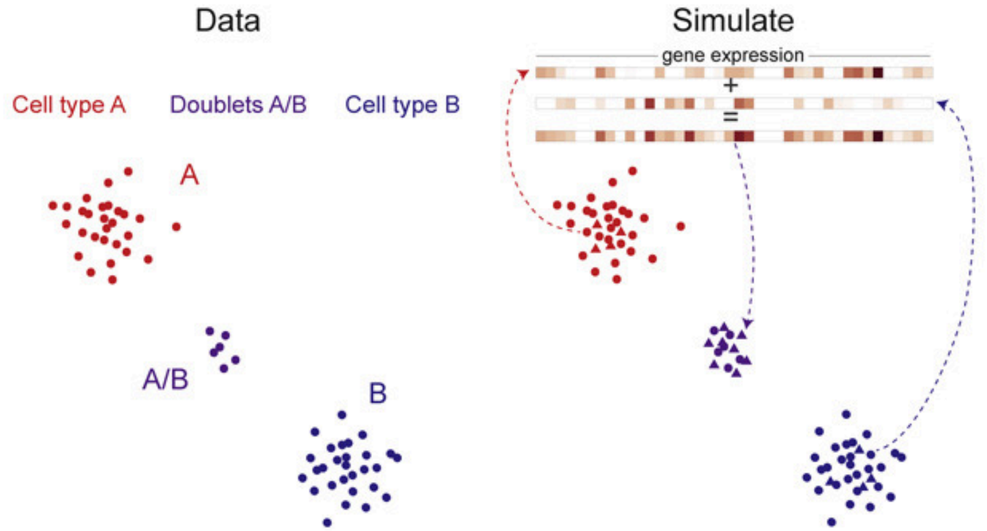
Variable mRNA capture efficiency

Library not sequenced to saturation
Sequencing errors in cell barcodes
cDNA fragment maps to multiple genes

Doublet Detection

- Heterotypic/homotypic doublets
- Biological doublets (!)
- Multiplets (rare)

We can detect heterotypic doublets using the fact that their transcriptome is the **sum** of two individual cell transcriptomes (usually via simulation)



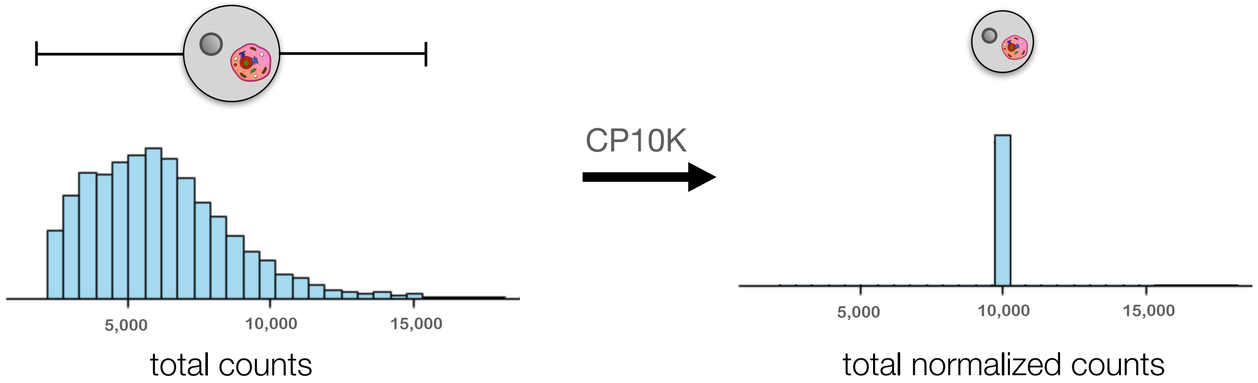
Wolock, et al. "**Scrublet**: computational identification of cell doublets in single-cell transcriptomic data." *Cell systems* (2019).

McGinnis, et al. "**DoubletFinder**: doublet detection in single-cell RNA sequencing data using artificial nearest neighbors." *Cell systems* (2019).

Bernstein, et al. "**Solo**: doublet identification in single-cell RNA-seq via semi-supervised deep learning." *Cell systems* (2020).

Single-cell RNA Seq | Normalization & Transformation

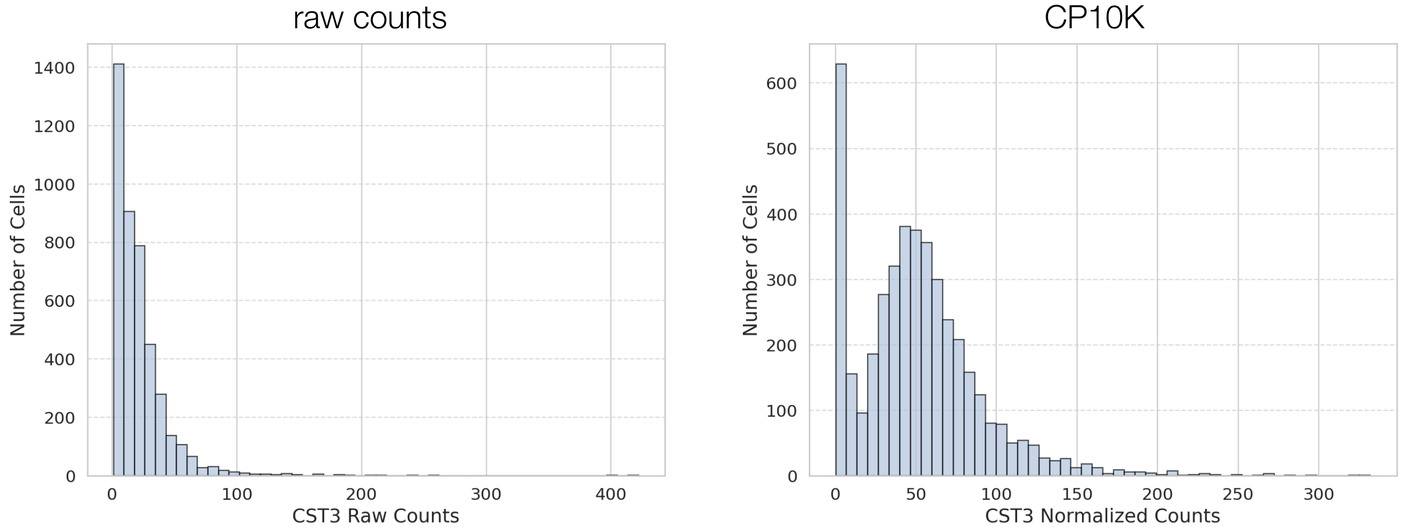
- CP10k normalization (Counts Per 10K)



The CP10K normalized counts answer the following question:

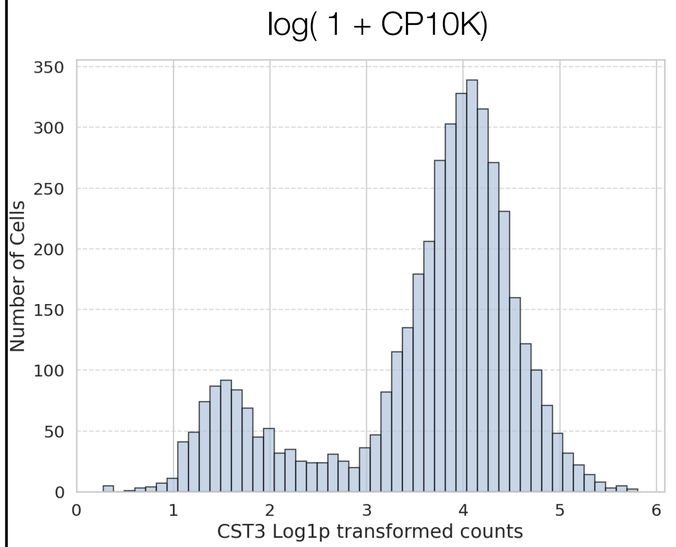
"How many mRNA molecules would you expect to get if you sampled 10K from that cell?"

- Histogram of **CST3** expression (highly expressed in Monocytes and DCs):



Logarithmize

- variance stabilization transformation
- closer to a normal distribution than counts

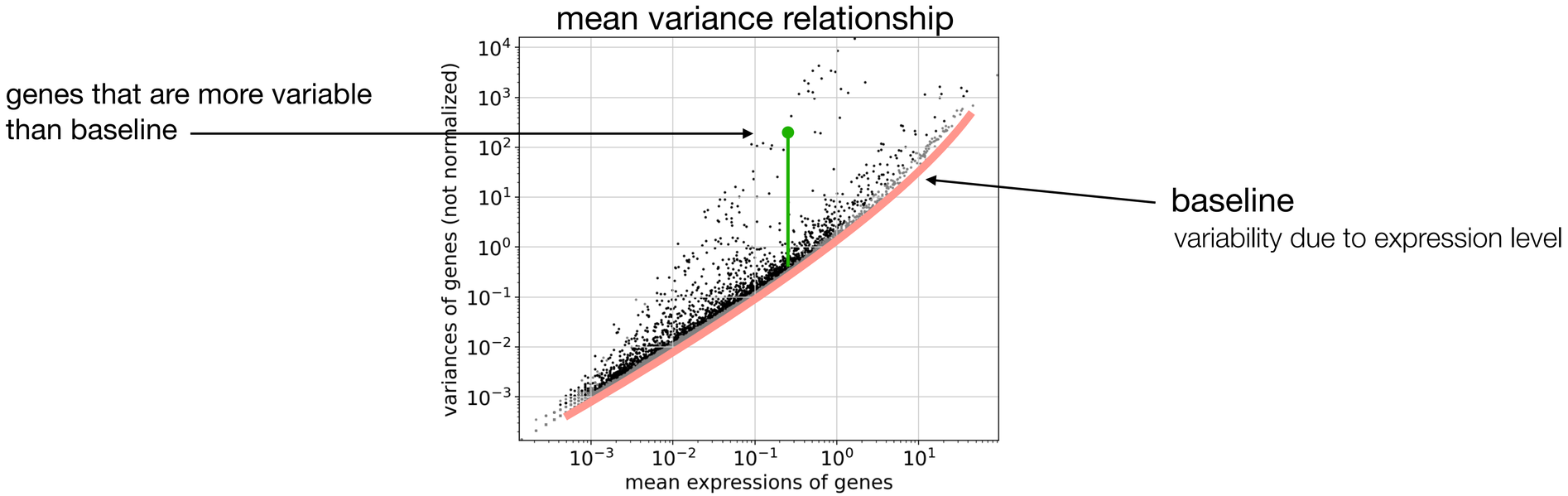


Single-cell RNA Seq | Feature Selection

- Selecting “interesting” genes for downstream analysis
  - Most genes in the dataset will not be biologically relevant for the experiment
  - We want to select genes that are “variable”, meaning their expression levels change significantly across different cell types, treatment conditions. etc
  - This can be thought of as an intermediate “dimensionality reduction” for the data (20K genes -> 5K genes)

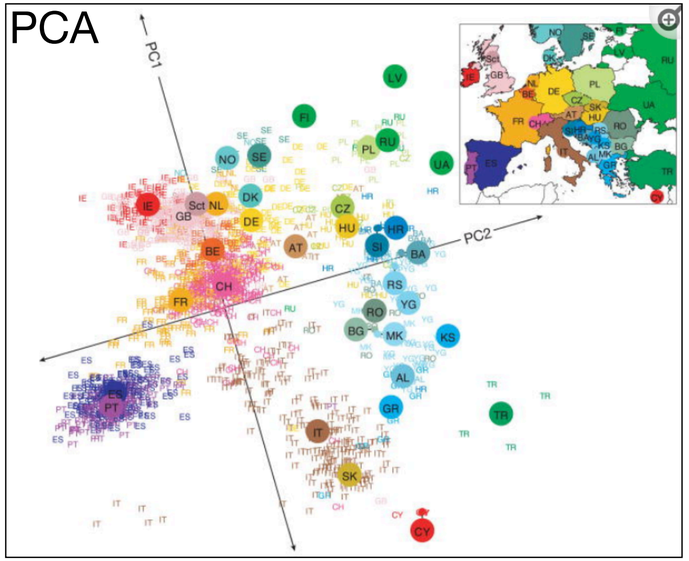
★ Highly variable genes are **not** the genes with the highest variance

- Property of count data (not only gene counts):
  - variance depends on the mean!
  - i.e., the higher the gene is expressed on average the more variable it will be in the collected data



Single-cell RNA Seq | Dimensionality Reduction & Visualization

- We want to select “interesting” dimensions in the data
  - Although single cell data are “high-dimensional”, gene expression is highly coordinated (regulation, coexpression, gene networks, pathways)
  - intrinsic dimensionality of the data is likely much smaller
- Principal Component Analysis (PCA)
  - Linear dimentionality reduction, well understood
  - interpretable, maximizes variance of the projected data
  - typically used to get ~50 dimensions that capture most of the variability



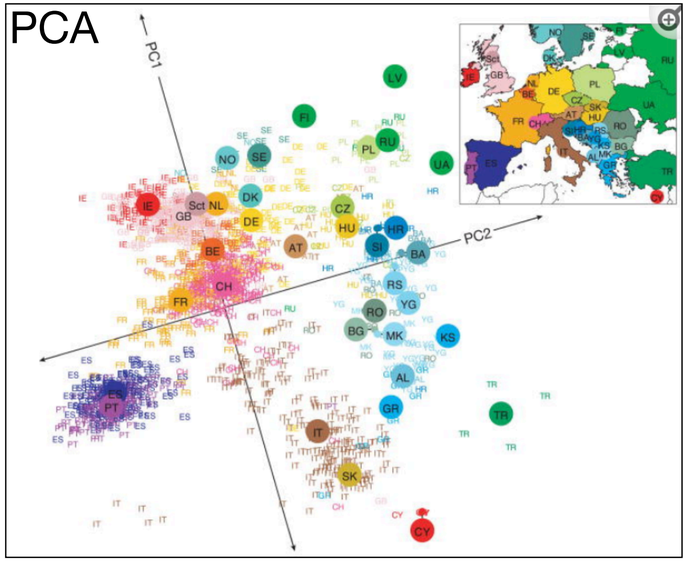
Novembre, *et al.* Genes mirror geography within Europe. *Nature* (2008). https://doi.org/10.1038/nature07331

Single-cell RNA Seq | Dimensionality Reduction & Visualization

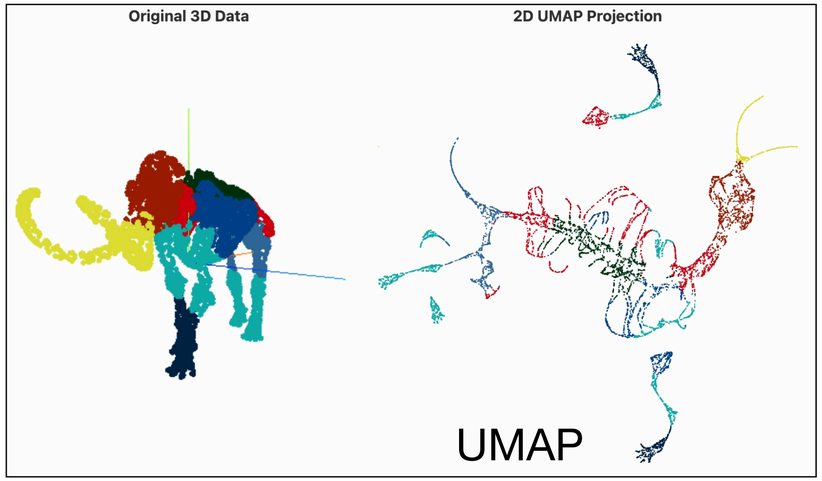
- We want to select “interesting” dimensions in the data
  - Although single cell data are “high-dimensional”, gene expression is highly coordinated (regulation, coexpression, gene networks, pathways)
  - intrinsic dimensionality of the data is likely much smaller

- Principal Component Analysis (PCA)
  - Linear dimensionality reduction, well understood
  - interpretable, maximizes variance of the projected data
  - typically used to get ~50 dimensions that capture most of the variability

- Non-linear dimensionality reduction (t-SNE, UMAP)
  - usually the last step to go from 50 dimensions to a 2D plot
  - good for high-level visualization, and data exploration
  - extreme information loss — not interpretable
  - See **interactive examples** in https://pair-code.github.io/understanding-umap/



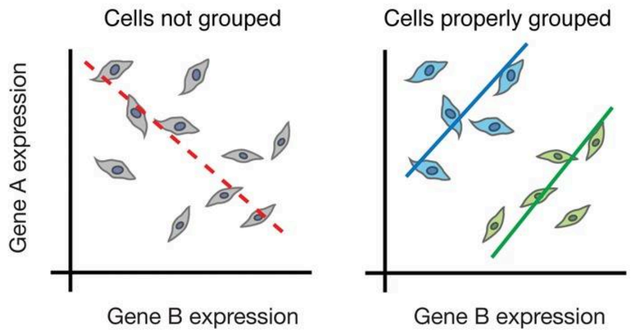
Novembre, *et al.* Genes mirror geography within Europe. *Nature* (2008). https://doi.org/10.1038/nature07331



https://pair-code.github.io/understanding-umap/

Single-cell RNA Seq | Clustering and Cell type annotation

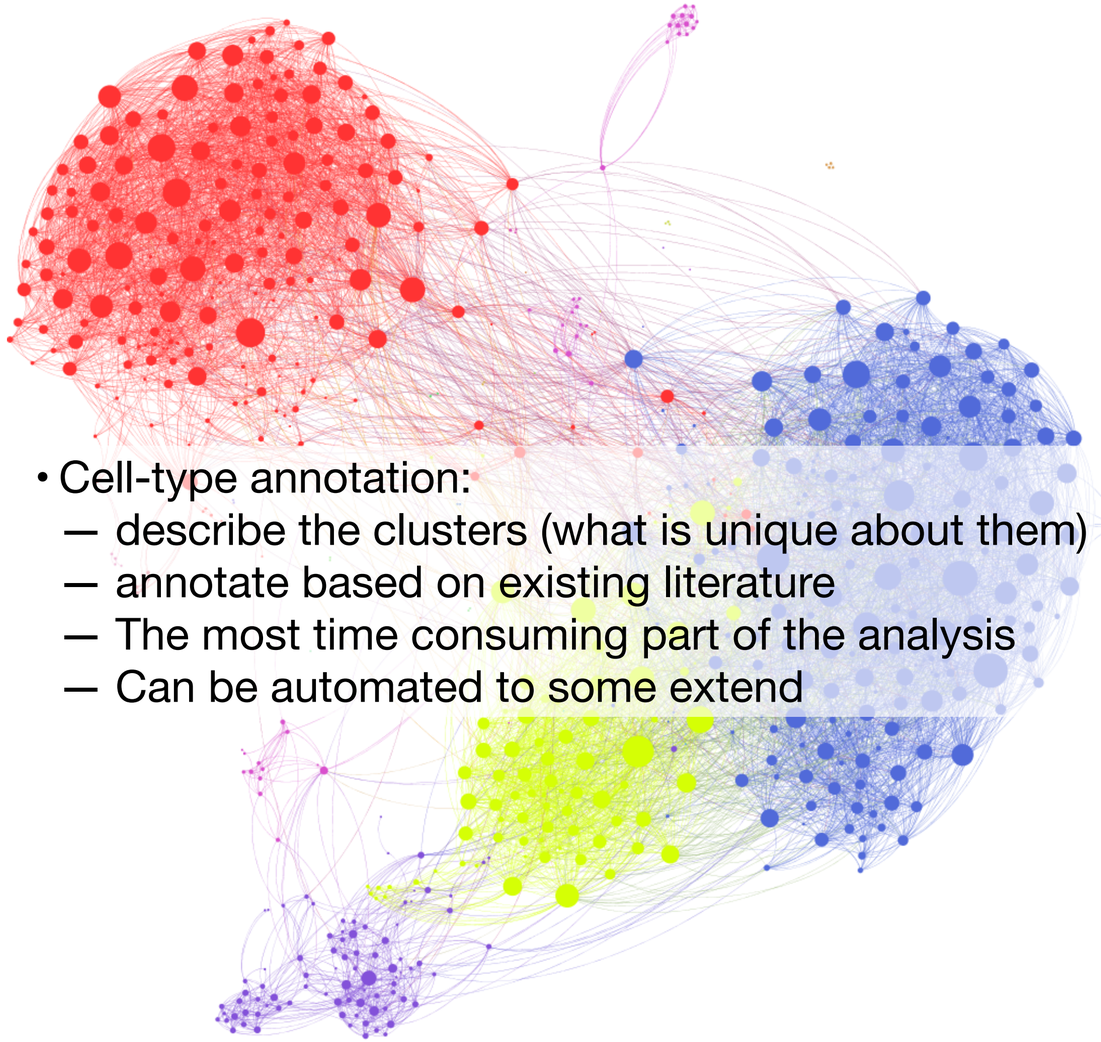
- We want to find **groups of similar cells** in the data
  - Similarity based on gene expression / transcriptional state
  - Proper grouping is very important (Simpson's paradox)



- Two key steps:

1. Neighbor graph construction
   — each cell is connected to its k=15 closest neighbors

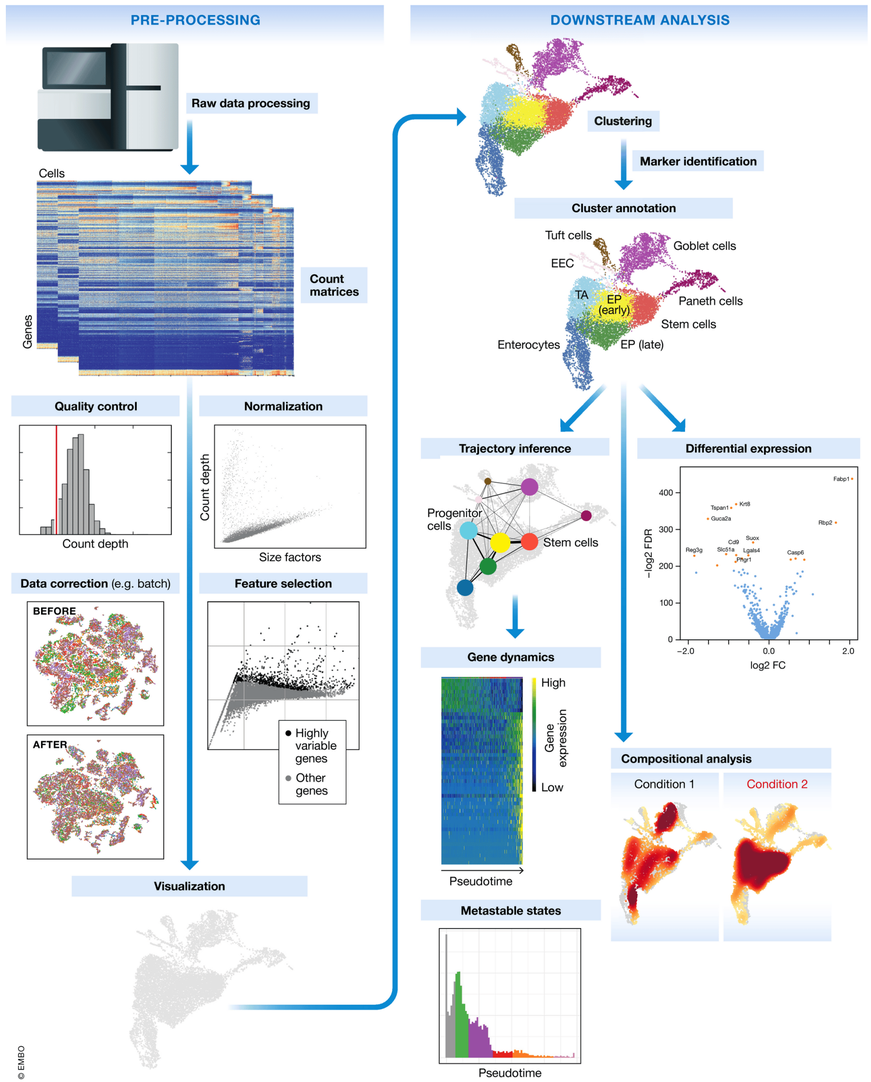
2. Community detection
   —groups of cells that are well-connected
   —same algorithms used to find communities in social networks



- Cell-type annotation:
  — describe the clusters (what is unique about them)
  — annotate based on existing literature
  — The most time consuming part of the analysis
  — Can be automated to some extend

Trapnell, et al., 2015 10.1101/gr.190595.115

Single-cell RNA Seq | Overview of the analysis



Summary of key steps

1. Data Pre-processing & Quality Control (QC)
- **Read Alignment & Count Matrix Generation:** Map sequencing reads to a reference genome, quantify gene expression levels, and create a cell by gene matrix, where each entry represents the expression level of a specific gene in a single cell.
- **Quality Control:** Remove low-quality cells and genes:
  - Low Read Count: Filter cells with very low reads.
  - Few Genes Detected: Remove cells with few genes detected.
  - Mitochondrial Reads: Discard cells with high mitochondrial content.
  - Doublets: Remove droplets with multiple cells.

2. Normalization & Transformation
- **Normalization:** Correct for technical variations in sequencing depth to ensure that differences in gene expression reflect biological variation
- **Transformation:** Apply transformations such as log1p to stabilize variance across expression levels and reduce skewness.

3. Feature Selection
- **Highly Variable Genes:** Identify and retain genes that show high variability across cells, as these genes are more likely to capture biologically meaningful differences.

4. Dimensionality Reduction & Visualization
- **Dimensionality Reduction:** Reduce features while retaining key information (e.g., PCA, t-SNE, UMAP).
- **Visualization:** Plot reduced data to visualize relationships and groupings among cells

5. Batch Effect Correction
- **Data Integration Methods:** Adjust for batch effects that may arise from technical differences between sequencing runs or sample processing, using methods like Combat or Harmony.

6. Clustering & Cell Type Annotation
- **Clustering:** Group cells with similar expression profiles.
- **Cell type annotation:**
  - Marker Genes: Identify cluster-specific genes.
  - Literature Comparison: Compare markers to known cell types.
  - Reference Mapping: Use reference atlases for annotation.

7. Downstream Analyses
- **Differential Expression:** Identify genes that are differentially expressed between clusters or treatment conditions.
- **Trajectory Inference:** Order cells along a pseudotime trajectory to model dynamic processes (e.g., differentiation).
- **Compositional analysis:** Identify changes in cell-type composition between conditions (healthy vs disease)

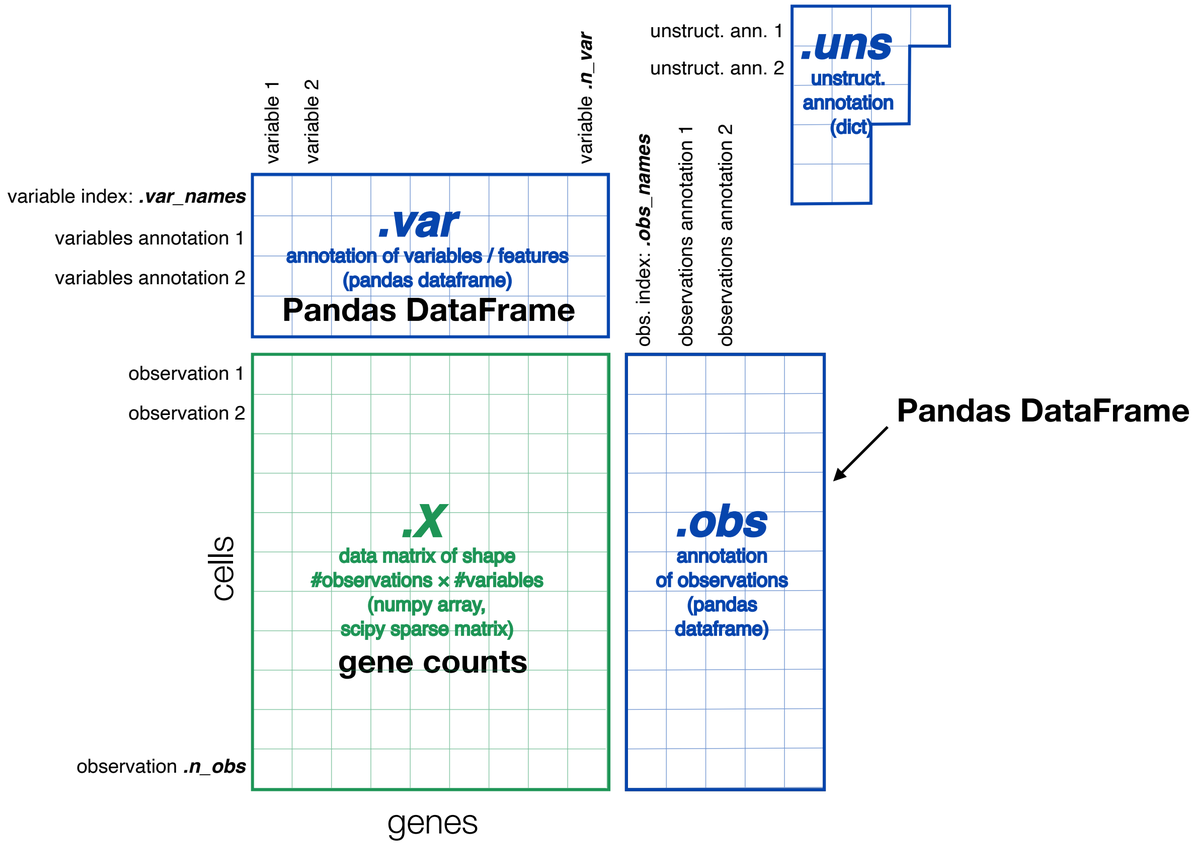
https://www.sc-best-practices.org/preamble.html

Luecken, Malte D., and Fabian J. Theis. "Current best practices in single-cell RNA-seq analysis: a tutorial." *Molecular systems biology* 15.6 (2019): e8746.

Single-cell analysis using scanpy

- Scanpy is a python library specifically designed for analyzing single-cell gene expression data. It provides **functions** for both data analysis and visualization.
- We will be working with **AnnData** objects that are designed to hold single cell data
- Inside an AnnData object:

```
adata.X
adata.obs
adata.var
adata.var_names
adata[filtering_condition]
adata.layers["layer_name"]
```



During the workshop:

We will import Scanpy, load the data, and calculate quality control metrics.	`import scanpy as sc`
	`sc.read_h5ad('blood_RNA_raw_counts.h5ad')`
	`sc.pp.calculate_qc_metrics(adata,…)`

Next, we'll visualize these metrics with a violin plot and filter out low-quality genes and cells based on identified thresholds.	`sc.pl.violin(adata,…)`
	`sc.pp.filter_genes(adata,…)`
	`adata = adata[ qc_filtering_condition ]`

Using Scrublet, we'll identify and remove doublets	`sc.pp.scrublet(adata,…)`
Then we will normalize and log-transform the data.	`sc.pp.normalize_total(adata,…)` `sc.pp.log1p(adata)`
We'll then identify highly variable genes and perform dimensionality reduction with PCA.	`sc.pp.highly_variable_genes(adata,…)` `sc.tl.pca(adata)`
Then we will correct batch effects using Harmony	`sc.external.pp.harmony_integrate(adata,…)`

After correcting batch effects, we'll build a nearest-neighbors graph, compute UMAP coordinates for visualization, and apply the Leiden algorithm to cluster the cells.	`sc.pp.neighbors(adata,…)`
	`sc.tl.umap(adata,…)`
	`sc.tl.leiden(adata,…)`

Finally, we'll visualize these clusters on a UMAP plot.	`sc.pl.umap(adata,…)`

During the workshop:

- Load Data

```
adata = sc.read_h5ad('blood_RNA_raw_counts.h5ad') # AnnData object
```

- Quality Control

```
adata.var["mt"] = adata.var_names.str.startswith("MT-")

sc.pp.calculate_qc_metrics(adata, qc_vars=["mt"], inplace=True)

sc.pp.filter_genes(adata,min_cells=5)

sc.pl.violin(
    adata,
    ["n_genes_by_counts", "total_counts", "pct_counts_mt"],
    jitter=0.4,
    multi_panel=True,)

adata = adata[ adata.obs['pct_counts_mt']< 12.5 ]
adata = adata[ adata.obs['total_counts']> 500 ]
```

During the workshop:

- Doublet Detection

```
sc.pp.scrublet(adata, batch_key="LIBRARY_ID")

adata = adata[ adata.obs['predicted_doublet']!=True ]
```

- Normalization and log-Transformation

```
adata.layers["counts"] = adata.X.copy()

sc.pp.normalize_total(adata, target_sum=10000) # CP10k normalization

sc.pp.log1p(adata) # Logarithmize the data
```

- Highly variable genes (feature selection)

```
sc.pp.highly_variable_genes(adata, n_top_genes = 5000,
                            flavor='seurat_v3',
                            batch_key='LIBRARY_ID',
                            layer='counts')
```

During the workshop:

- Dimensionality reduction with PCA

```
sc.tl.pca(adata, n_comps=50)
```

```
sc.pl.pca(adata,color='LIBRARY_ID',size=5)
```

- Batch effect correction with harmony

```
sc.external.pp.harmony_integrate(adata,key='LIBRARY_ID')
```

- Nearest-neighbor graph, clustering and UMAP projection

```
sc.pp.neighbors(adata, use_rep='X_pca_harmony', n_neighbors=15, random_state=42)
```

```
sc.tl.umap(adata, min_dist=0.5, spread=1.0)
```

```
sc.tl.leiden(adata, resolution=1,key_added='leiden_clusters')
```

- UMAP visualization

```
sc.pl.umap(adata,color='leiden_clusters')
```