# scRNA-seq Workshop Part 1

## Applied Bioinformatics for Genomics

JENNIFER A. FOLTZ, PHD

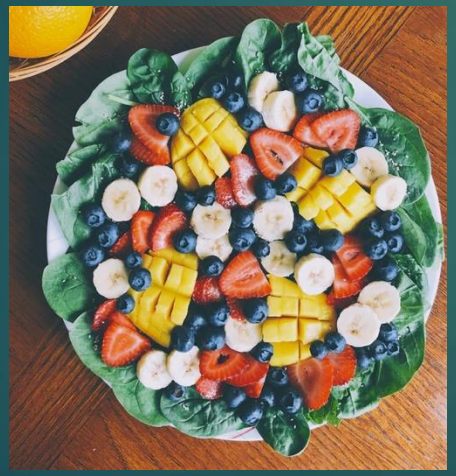
ASSISTANT PROFESSOR, SECTION OF COMPUTATIONAL BIOLOGY

JENNIFER.A.FOLTZ@WUSTL.EDU

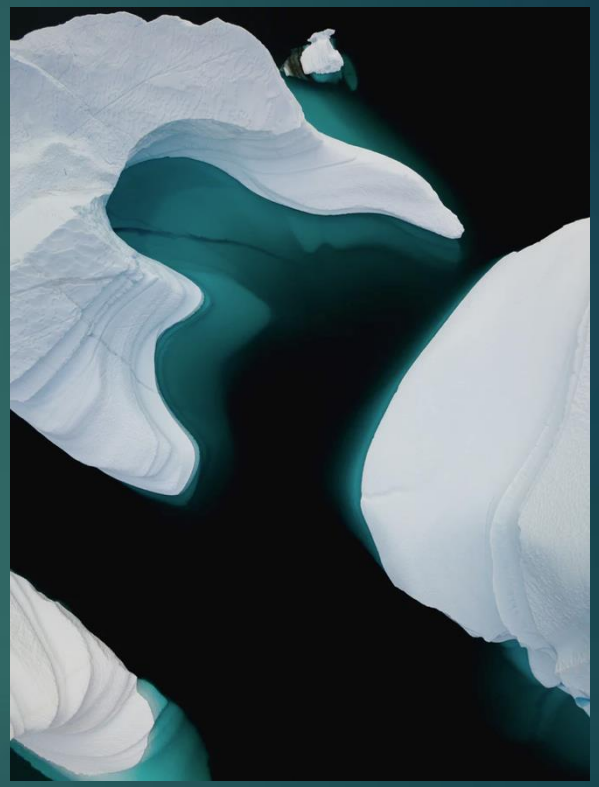

With content contributions from Dr. Petti

# scRNAseq versus Bulk RNA-seq

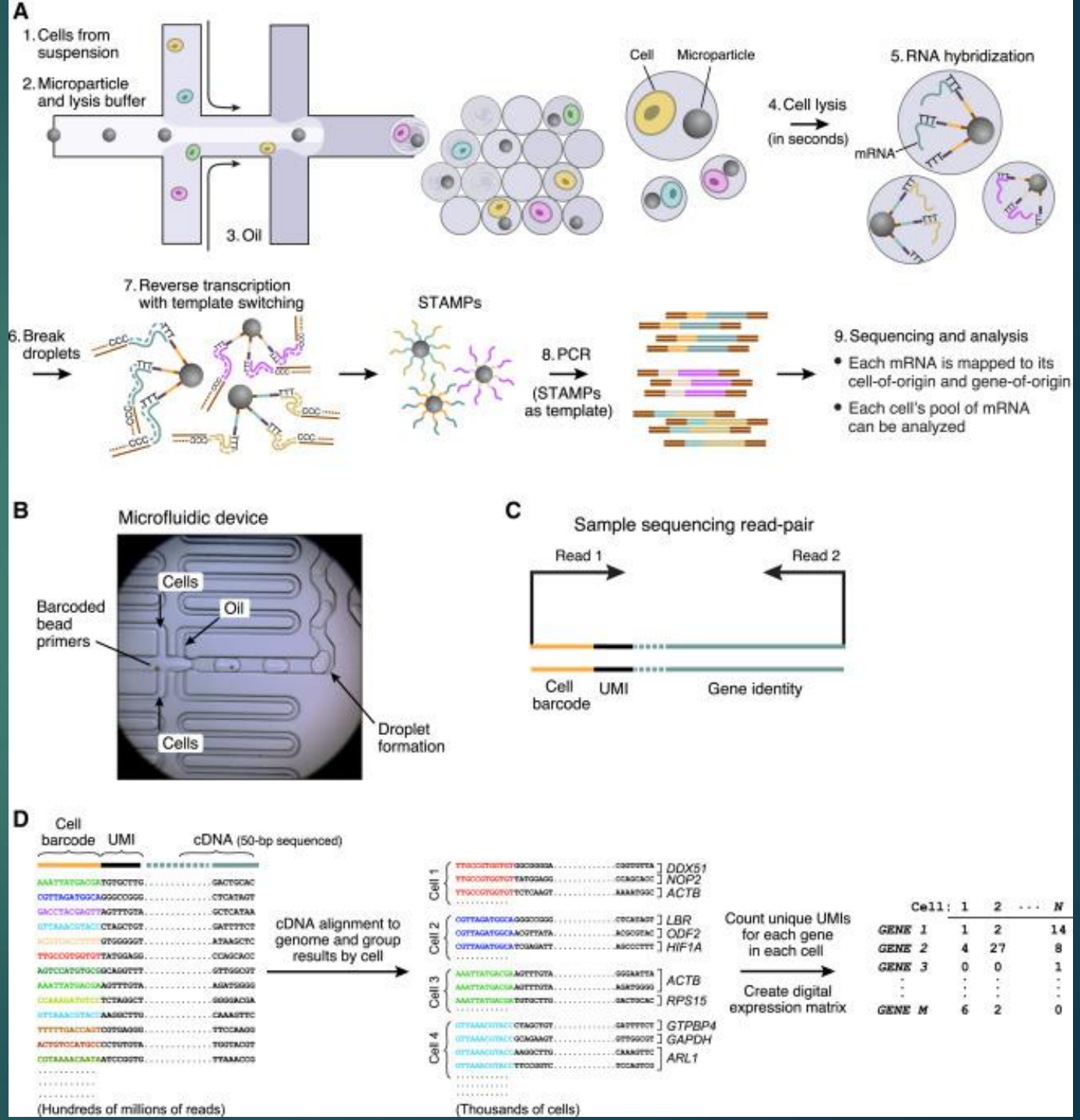
Bulk RNA-seq averages across the population



scRNA-seq reports per-cell expression and enables computational “sorting”

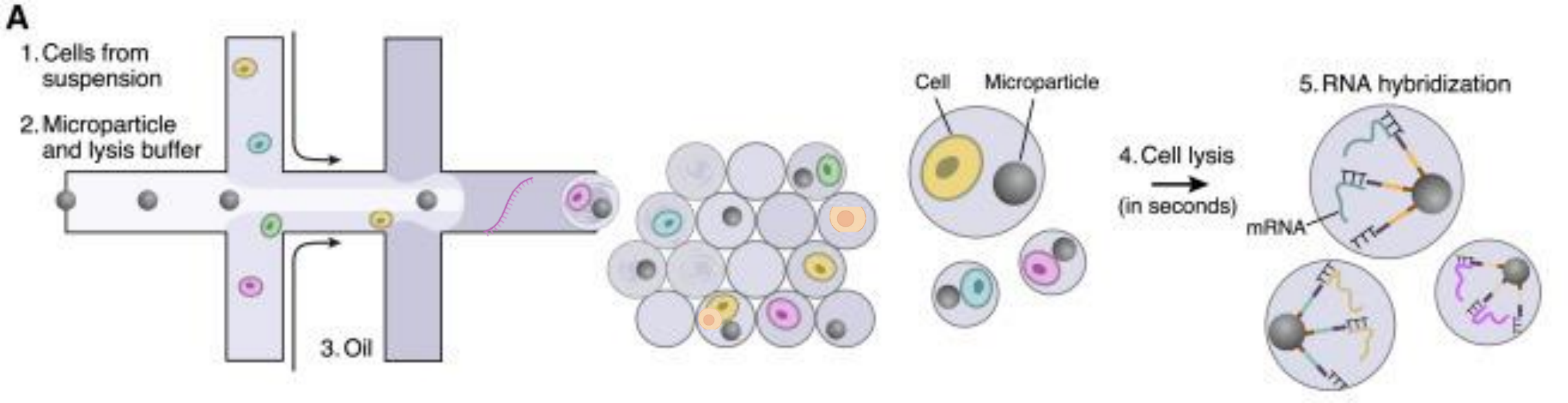


# Droplet based scRNA-seq: The Good




*Mackowski et al, 2015, Cell*

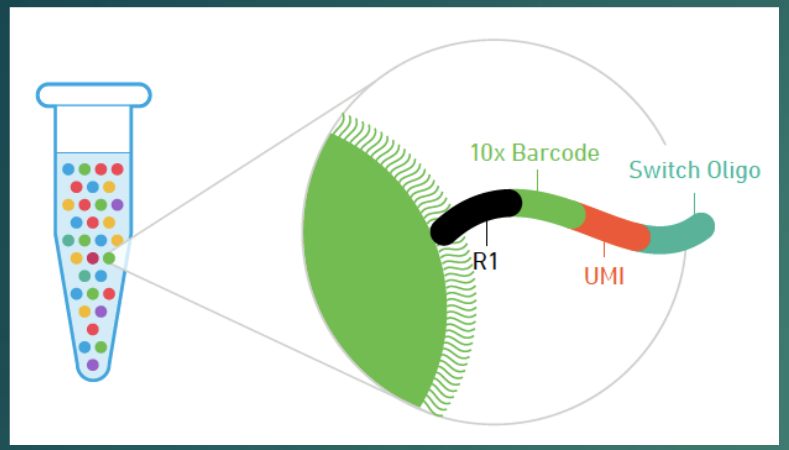
# Droplet based scRNA-seq: The Ugly



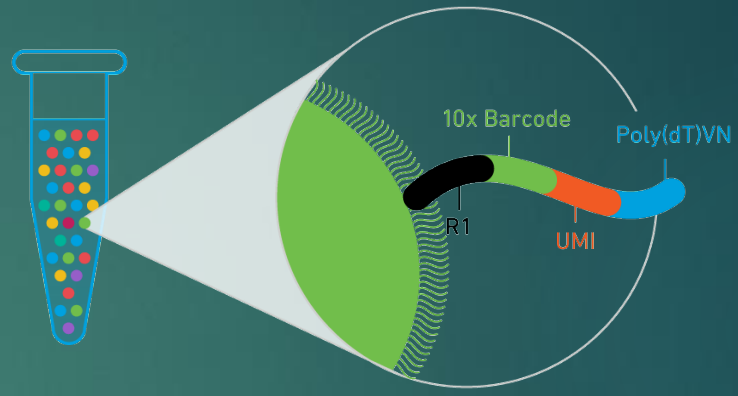

*Mackowski et al, 2015, Cell; modified with biorender.com; 10x genomics*

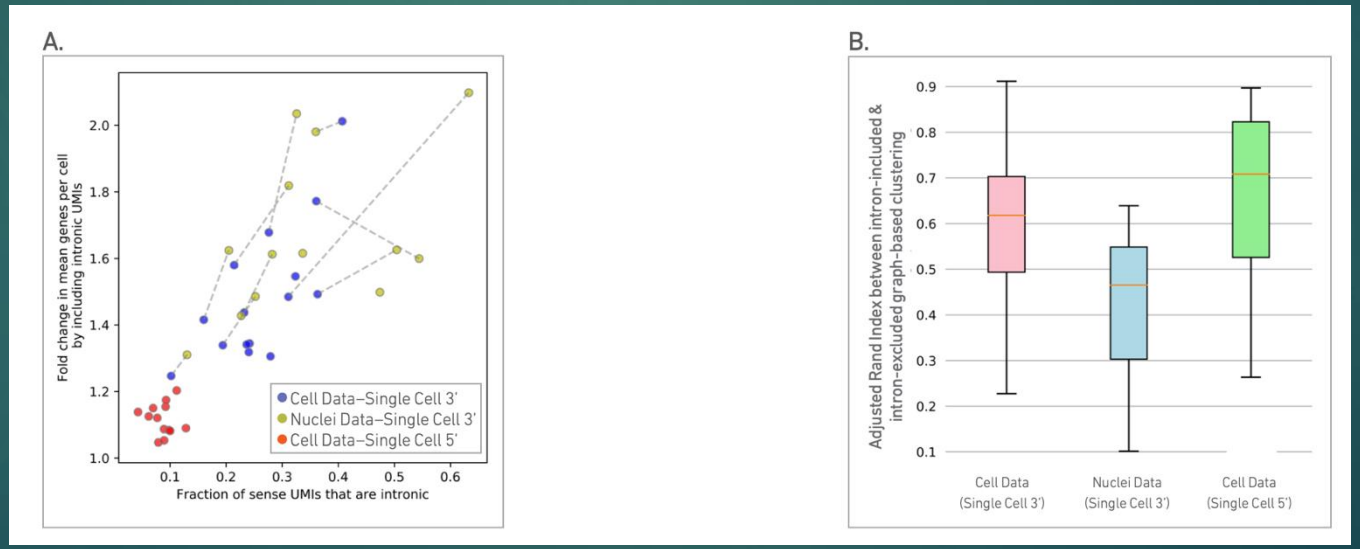
# 5’ vs. 3’ Chemistry

5’ bead




3’ bead



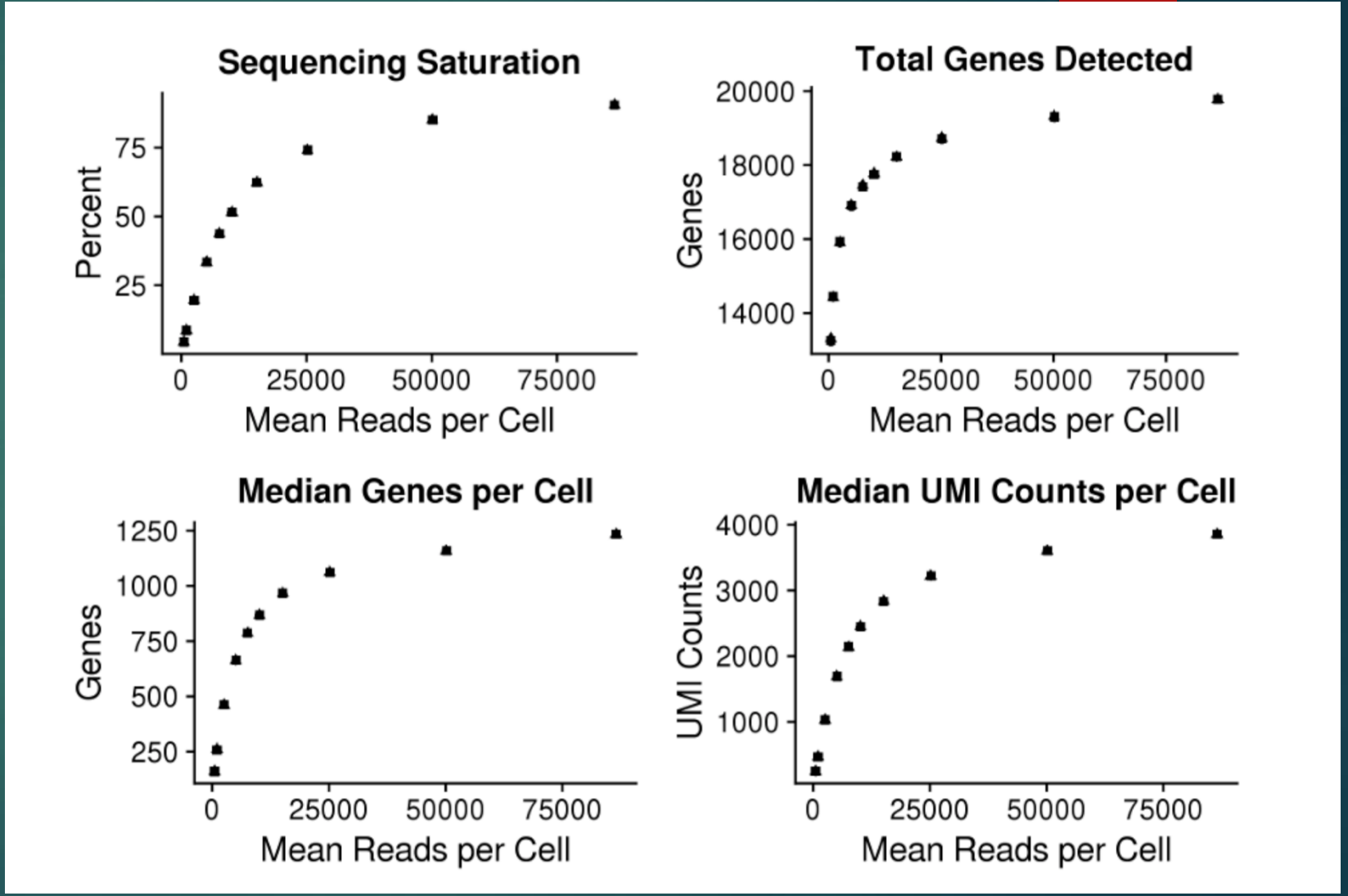

And more! https://teichlab.github.io/scg_lib_structs/

# How deeply do you need to sequence?

**Recommendations (reads/cell):**

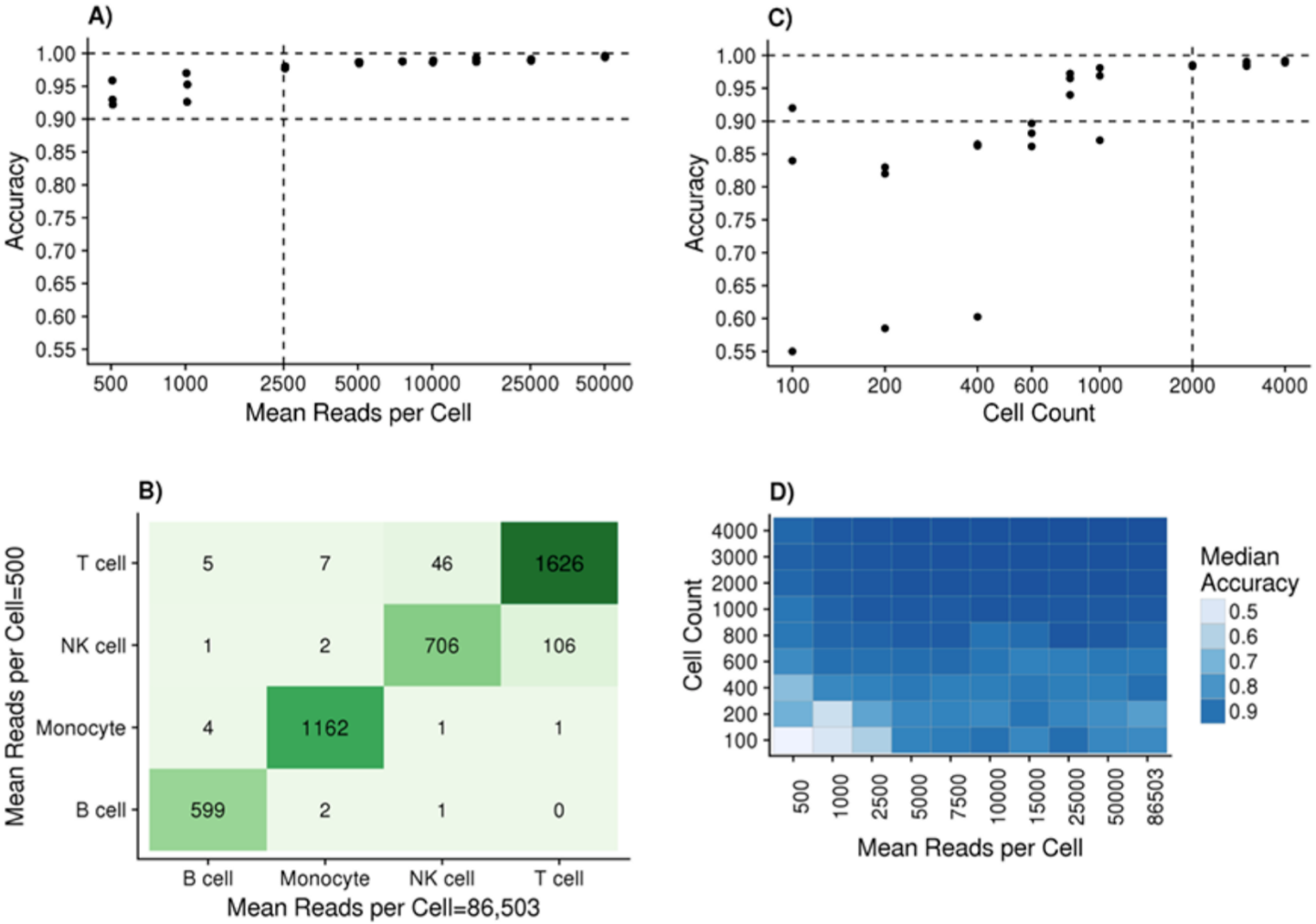
3’ - V3: 20K
5’ - 20K
5’ with variant discovery - 200K
5’ V(D)J - 5K
5’ Cell Surface Protein- 5K
Fixed RNA: 10K

General Rule:
Achieve 90% saturation




https://www.10xgenomics.com/support/single-cell-gene-expression/documentation/steps/sample-prep/resolving-cell-types-as-a-function-of-read-depth-and-cell-number

# How deeply do you need to sequence?



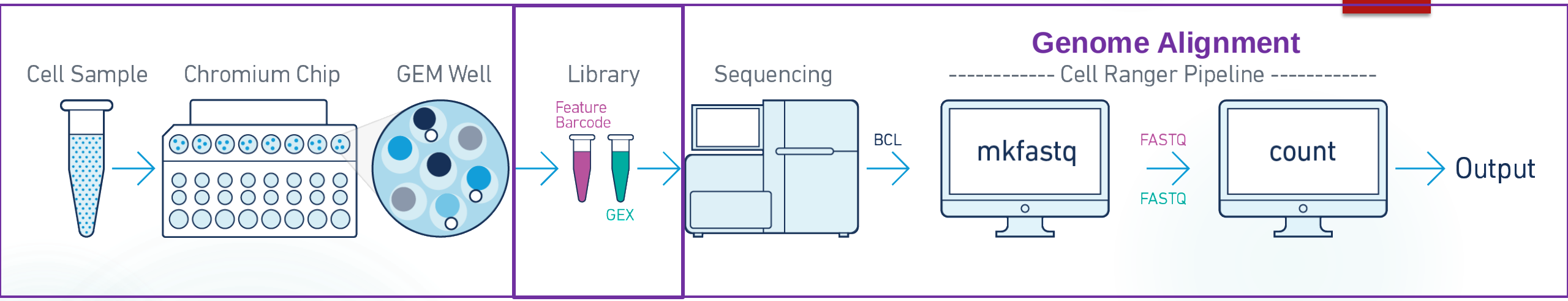

https://www.10xgenomics.com/support/single-cell-gene-expression/documentation/steps/sample-prep/resolving-cell-types-as-a-function-of-read-depth-and-cell-number

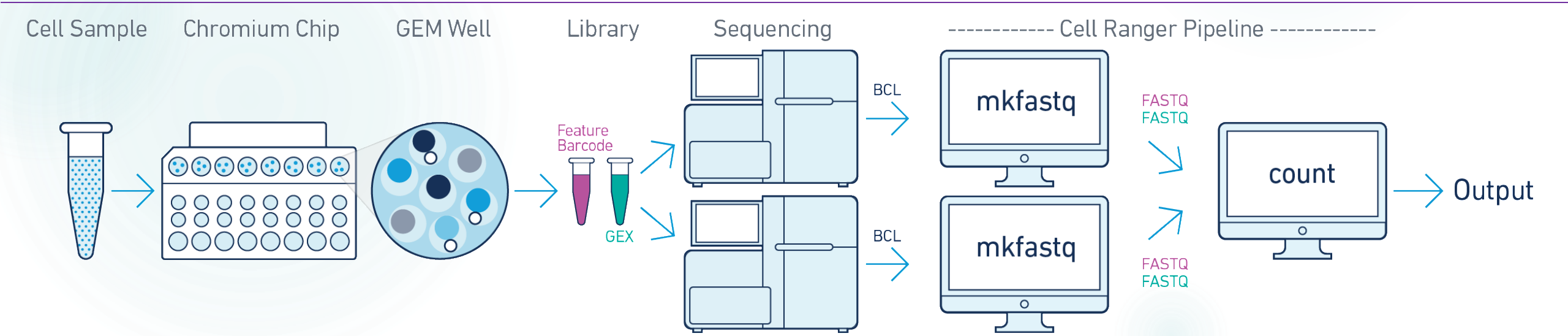
# Sample Preparation: scRNA-seq

- To sort or not to sort, that is the question?
  - Factors to consider:
    - Do you have a population of interest in your sample or is this purely exploratory?
      - The number of cells present in your population of interest- too low prevalence is impossible to find
      - Keep in mind the doublet rate for all sample submissions: ~8% when targeting 10,000 cells- this goes up with number of cells targeted
    - Will sorting interfere with downstream antibody staining?
    - What is the expected viability?

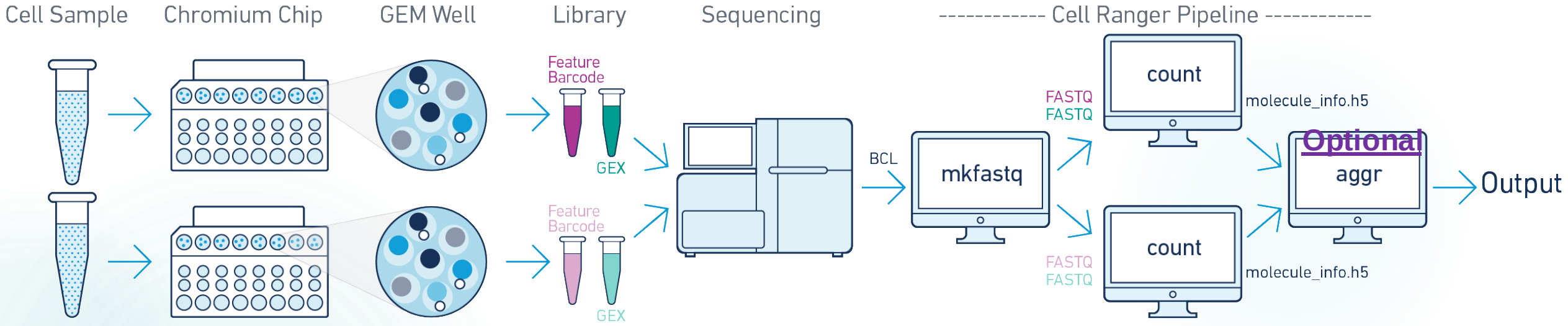
    - https://satijalab.org/howmanycells/

# Sample Processing: MGI and Beyond



# Sample Processing: Multiple Samples

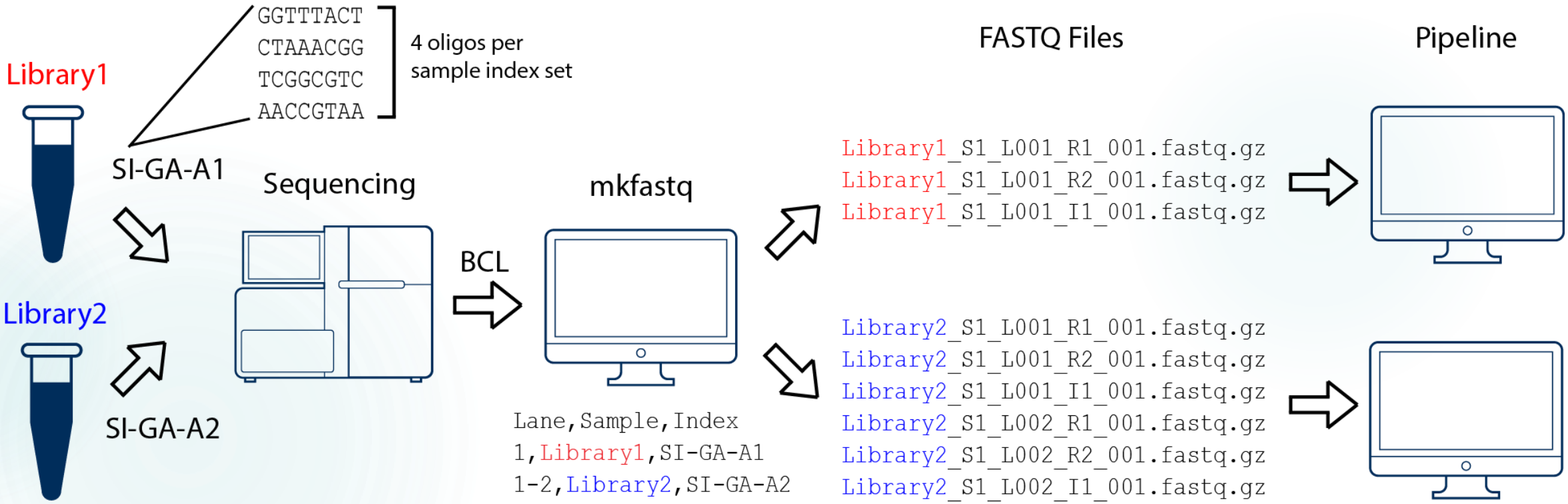



- Two Options:
  - Aggregate samples in cellRanger
    - Randomly downsamples reads across samples to be equivalent (default setting-can be turned off)
    - Allows you to visualize multiple samples in the cloupe interface
    - Will need to have run Cellranger on individual samples first with exact same feature set
    - Barcodes used to identify cells will be appended with a number (e.g. -2, -3) that corresponds to order of aggregation
  - Merge samples/aggregate in R (or other packages)
    - Maintains original sequencing depth and tries to normalize sequencing by cells

# CellRanger Important Info

- Genome Selection:
  - Human/mouse
  - Special Features- e.g. CARs
- Alignment for gene, protein (antibodies/hashtags), TCR/BCR, & ATAC fragments
- Output Key Files:
  - Filtered_feature_bc_matrices: input into most single-cell packages
    - In contrast to raw_feature_bc_matrices: this has all cell barcodes not just ones identified to be present in your data
    - Essentially a feature by cell matrix
  - Web_summary.html: user friendly visualization of QC metrics for cellranger
  - .cloupe file: point and click visual interface for limited single-cell analysis and visualization of data:
    - Outputs a umap/tsne of data, allows for visualization of genes and proteins, & subclustering of data, along with differential expression & graphs
    - Great way to explore the data and identify any potential problems with output quickly

# Cellranger Step 0: Sample Demultiplexing




https://support.10xgenom /latest

# CellRanger Web_summary

Summary | Analysis

**10,950**
Estimated Number of Cells

**67,137**
Mean Reads per Cell

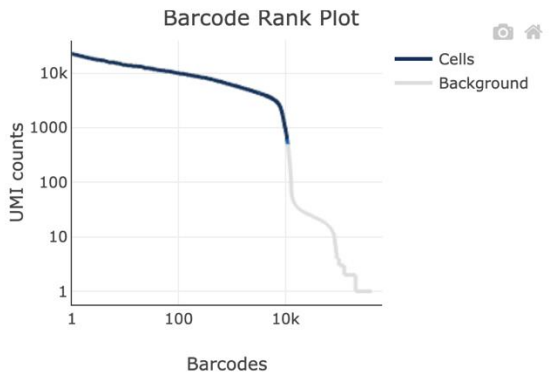
## Sequencing

| | |
|---|---|
| Number of Reads | 735,154,190 |
| Number of Short Reads Skipped | 0 |
| Valid Barcodes | 86.1% |
| Valid UMIs | 99.4% |
| Sequencing Saturation | 88.2% |
| Q30 Bases in Barcode | 93.4% |
| Q30 Bases in RNA Read | 88.8% |
| Q30 Bases in RNA Read 2 | 85.6% |
| Q30 Bases in UMI | 92.7% |

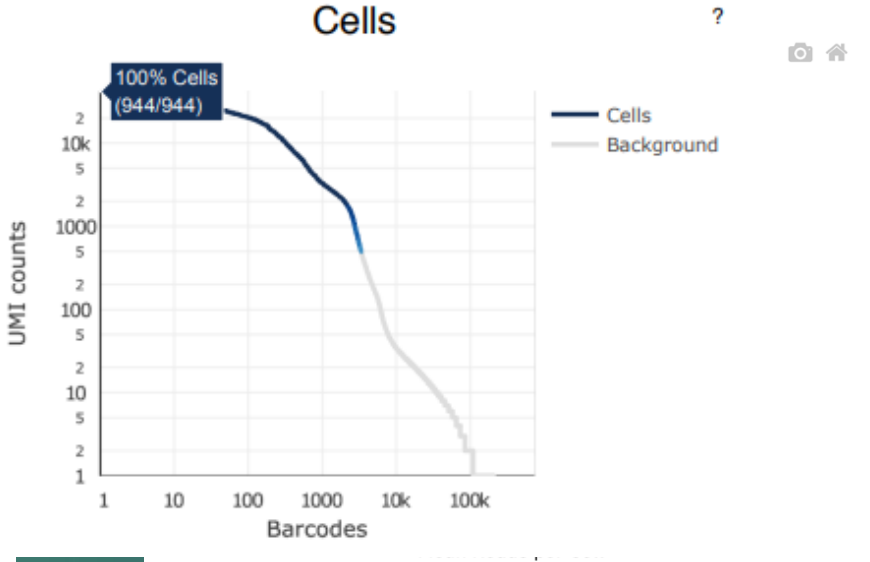
## Mapping

| | |
|---|---|
| Reads Mapped to Genome | 87.9% |
| Reads Mapped to Genome (GRCh38) | 87.1% |
| Reads Mapped to Genome (mm10) | 0.8% |
| Reads Mapped Confidently to Genome | 71.5% |

## Cells



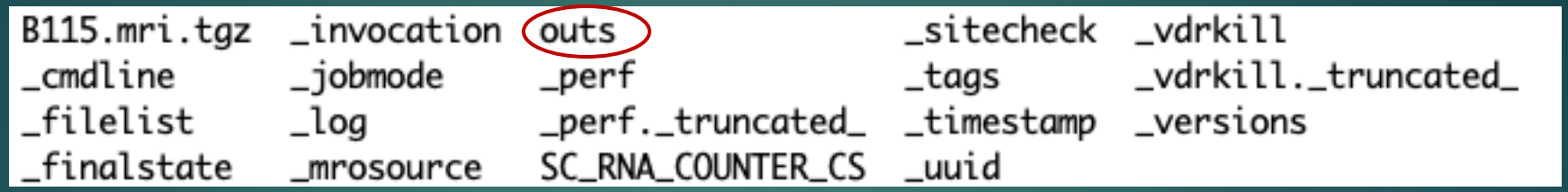

| | |
|---|---|
| Estimated Number of Cells | 10,950 |
| Estimated Number of Cells (GRCh38) | 10,856 |
| Estimated Number of Cells (mm10) | 1,784 |
| Fraction Reads in Cells | 95.5% |
| Fraction Reads in Cells (GRCh38) | 95.6% |
| Fraction Reads in Cells (mm10) | 40.6% |
| Mean Reads per Cell | 67,137 |
| Median Genes per Cell (GRCh38) | 1,590 |
| Median Genes per Cell (mm10) | 2 |
| Total Genes Detected (GRCh38) | 19,641 |
| Total Genes Detected (mm10) | 62 |
| Median UMI Counts per Cell (GRCh38) | 3,438 |
| Median UMI Counts per Cell (mm10) | 162 |




## Sample

| | |
|---|---|
| Sample ID | Donor |
| Sample Description | |
| Chemistry | Single Cell 5' PE |
| Include introns | False |
| Reference Path | …ata-cellranger-GRCh38-and-mm10-3.1.0 |
| Transcriptome | GRCh38_and_mm10-3.1.0 |
| Pipeline Version | cellranger-6.0.0 |

# Cellranger output files



| | | | | |
|---|---|---|---|---|
| B115.mri.tgz | _invocation | outs | _sitecheck | _vdrkill |
| _cmdline | _jobmode | _perf | _tags | _vdrkill._truncated_ |
| _filelist | _log | _perf._truncated_ | _timestamp | _versions |
| _finalstate | _mrosource | SC_RNA_COUNTER_CS | _uuid | |

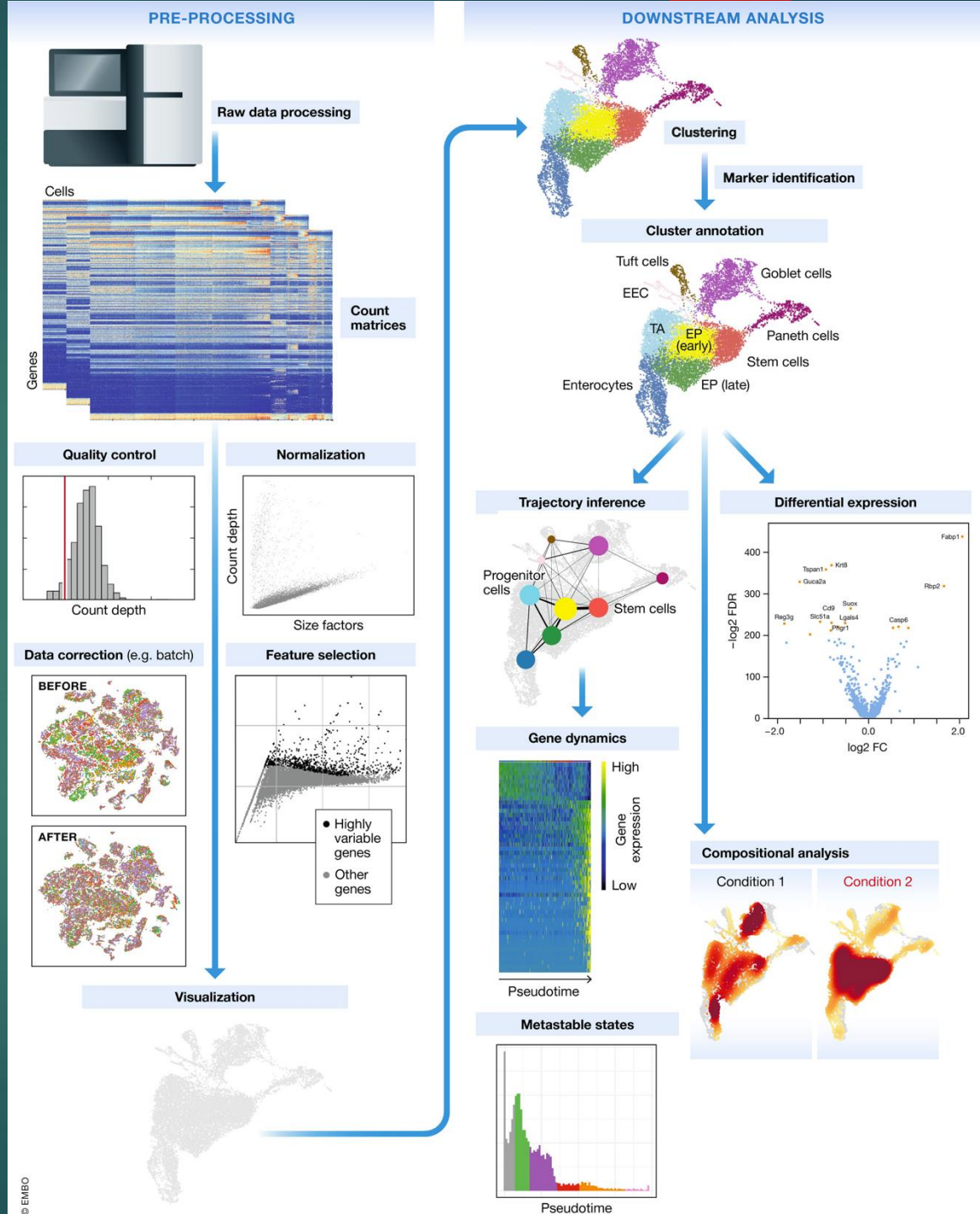
Analysis:
clustering - flat file clustering results
diffexp – DEGs for each cluster
pca – details about each principal component, projections, etc
tsne – coordinates of each cell in t-SNE plot
cloupe.cloupe – input to loupe browser for interactive analysis
**filtered_feature_bc_matrix**
barcodes.tsv.gz
features.tsv.gz
matrix.mtx.gz
filtered_feature_bc_matrix.h5

**metrics_summary.csv** – flat file QC information
molecule_info.h5
possorted_genome_bam.bam
possorted_genome_bam.bam.bai
raw_feature_bc_matrix – not filtered for cell-associated barcodes
raw_feature_bc_matrix.h5 – not filtered for cell-associated barcodes
**web_summary.html** – QC information and minimal interactive analysis

# Possible reasons for low quality

| Metric | Human |
| --- | --- |
| **Estimated Cells** | Low viability, lysed cells |
| **Target Reads/Cell** | Rarely problematic |
| **% Transcriptome mapping** | Wrong transcriptome, low sequence quality |
| **% Antisense Reads** | Wrong chemistry, low sequence quality |
| **Fraction reads in cells** | Lysed cells, extracellular RNA |

# Sample Processing Post-Sequencing




*Luecken et al., 2019*

# Methods Galore

*Number of single cell tools ~ Number of single cell studies ~ 500 (https://www.scrna-tools.org/)*

Point and Click:
Loupe Browse (free)
Partek Flow ($$$)
SeqGeq ($$$)
kana (free)

General-purpose:
***Seurat V5***
***Scanpy***
Monocle V3
Scran
scater
LIGER (NMF)
scAlign
Scanorama
Kallisto bustools (replaces cellranger)

Predicting the future:
RNAvelocity
Dynamo
CellRank
scVelo
Monocle 3 (R)
Slingshot (R)
PAGA (Python)
pCreode (Python)
STREAM
Palantir
CytoTrace (R/Python)

Large data sets:
scSVA
SAUCIE

Cell type assignment:
SingleR
CellHarmony
scPred
Moana
Garnet
Capybara

Mutation Detection:
CONICSmat (CNV)
HoneyBadger (CNV, LOH)
cb_sniffer (SNVs, Indels)
Vartrix (SNVs, Indels)

*Modified from Dr. Allegra Petti*

# Analysis governed by two main principles

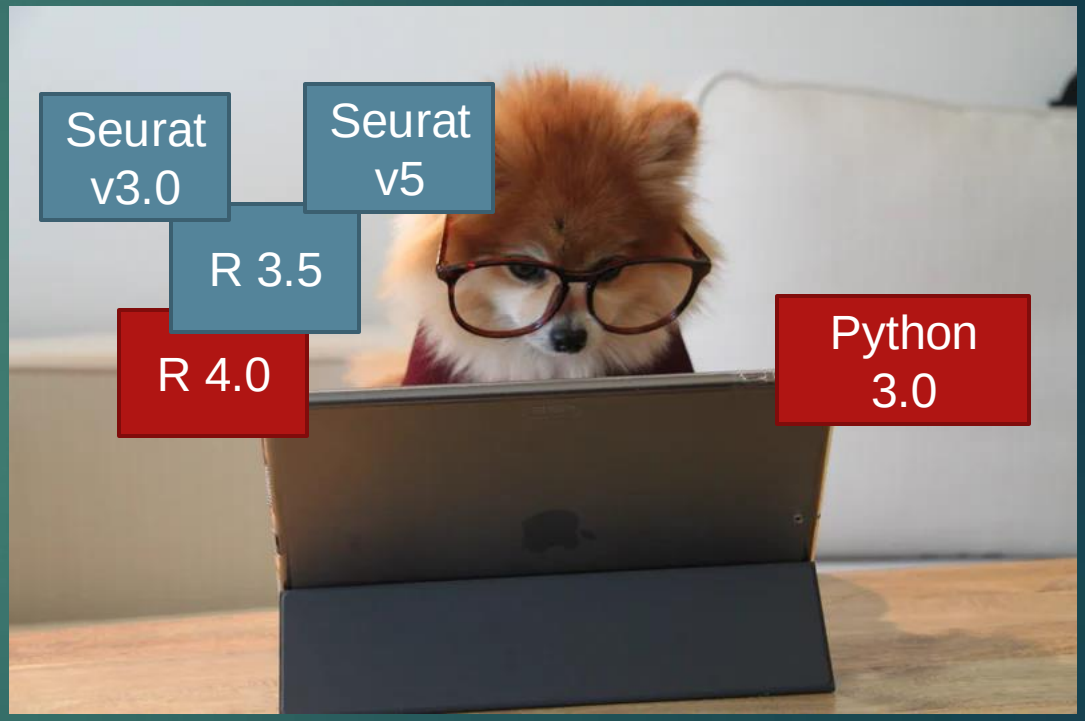
1. Single-cell RNA-seq data is very high-dimensional
2. And very sparse:
   Fraction of transcripts captured per cell:
   10x 3’ V2: 14-15%
   10x 3’ V3: 30-32%

Additional Quirks:

- Transcripts encoding ribosomal proteins can comprise 30-50% of reads
- Top 100 transcripts often comprise ~50% of reads
- Low sensitivity
- Works best for cell type classification, more subtle signatures may get lost
- Results may favor highly expressed genes (e.g. *VIM*)

# Seurat Analysis Package

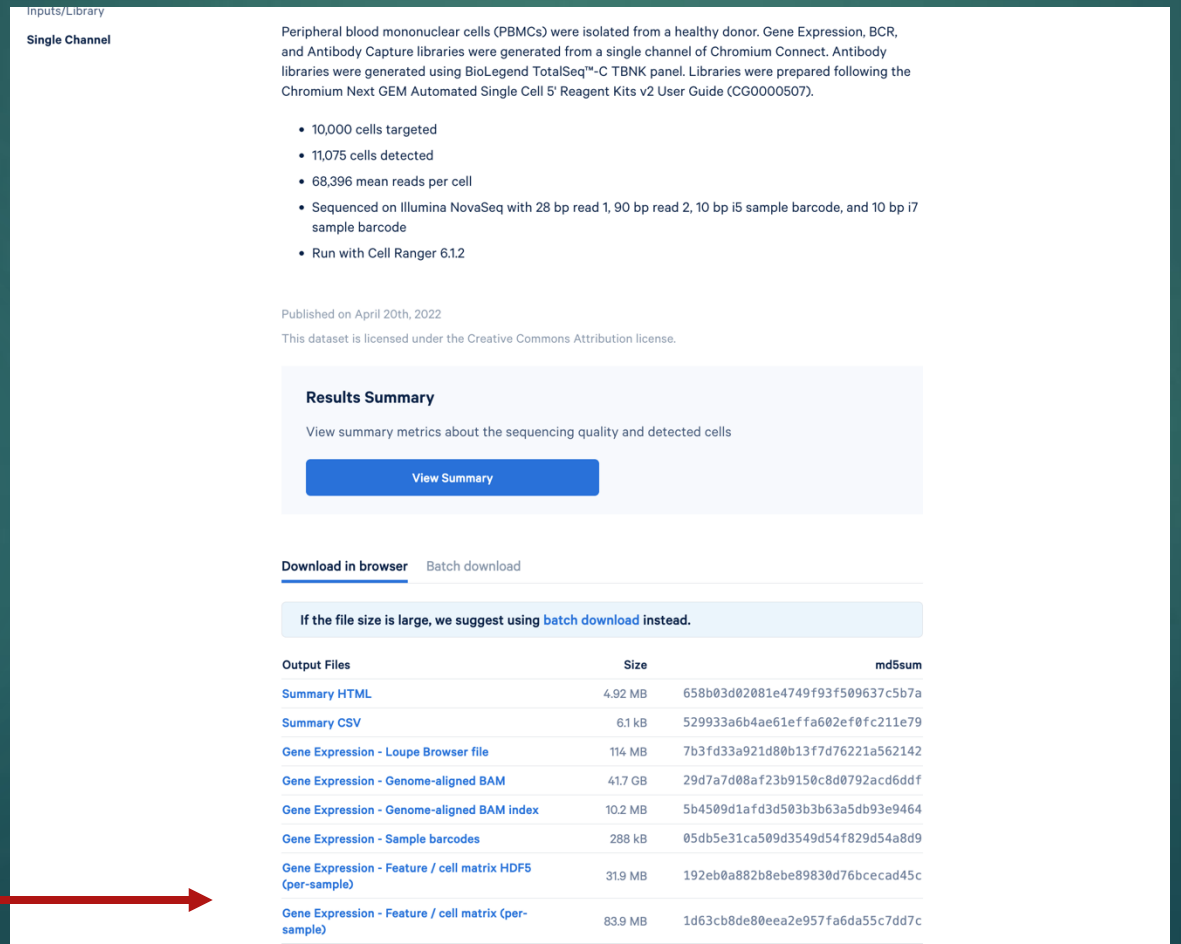
- Most popular R language based package for analyzing scRNAseq, CITE-seq, hashtag, CRISPR-single cell data
- Essentially provides convenient wrappers around available methods for normalization, dimensional reduction, visualization, differential expression
- Available for windows & mac, however, I recommend dockers:


- Ability to run multiple versions, packages on the same system without uninstalling/re-installing.
- Can share the same ”computer system” with colleagues to improve reproducibility

# Practice Dataset

https://www.10xgenomics.com/resources/datasets/integrated-gex-totalseq-c-and-bcr-analysis-of-chromium-connect-generated-library-from-10k-human-pbmcs-2-standard



Inputs/Library

**Single Channel**

Peripheral blood mononuclear cells (PBMCs) were isolated from a healthy donor. Gene Expression, BCR, and Antibody Capture libraries were generated from a single channel of Chromium Connect. Antibody libraries were generated using BioLegend TotalSeq™-C TBNK panel. Libraries were prepared following the Chromium Next GEM Automated Single Cell 5' Reagent Kits v2 User Guide (CG0000507).

- 10,000 cells targeted
- 11,075 cells detected
- 68,396 mean reads per cell
- Sequenced on Illumina NovaSeq with 28 bp read 1, 90 bp read 2, 10 bp i5 sample barcode, and 10 bp i7 sample barcode
- Run with Cell Ranger 6.1.2

Published on April 20th, 2022

This dataset is licensed under the Creative Commons Attribution license.

**Results Summary**

View summary metrics about the sequencing quality and detected cells

View Summary

**Download in browser** Batch download

If the file size is large, we suggest using batch download instead.

| Output Files | Size | md5sum |
|---|---|---|
| Summary HTML | 4.92 MB | 658b03d02081e4749f93f509637c5b7a |
| Summary CSV | 6.1 kB | 529933a6b4ae61effa602ef0fc211e79 |
| Gene Expression - Loupe Browser file | 114 MB | 7b3fd33a921d80b13f7d76221a562142 |
| Gene Expression - Genome-aligned BAM | 41.7 GB | 29d7a7d08af23b9150c8d0792acd6ddf |
| Gene Expression - Genome-aligned BAM index | 10.2 MB | 5b4509d1afd3d503b3b63a5db93e9464 |
| Gene Expression - Sample barcodes | 288 kB | 05db5e31ca509d3549d54f829d54a8d9 |
| Gene Expression - Feature / cell matrix HDF5 (per-sample) | 31.9 MB | 192eb0a882b8ebe89830d76bcecad45c |
| Gene Expression - Feature / cell matrix (per-sample) | 83.9 MB | 1d63cb8de80eea2e957fa6da55c7dd7c |

# Seurat-v5 Basics

1. In R terminal, type:
   ?FunctionName
   example: ?FindAllMarkers

2. Code available on github, e.g.:
   https://github.com/satijalab/seurat/blob/master/man/FindClusters.Rd

object.data <- Read10X("outs/**filtered_feature_bc_matrix**/")

- File path must have EXACTLY filtered_feature_bc_matrix Or with newer versions, can be sample_feature_bc_matrix
- This reads in a large matrix into R- if you have multiple data modalities, there will be a matrix for each modality (e.g. Gene, antibodies)
- These can be accessed via the $ e.g. object$`Gene Expression`- Note that R needs the `` to handle the spaces in your assay

object <- CreateSeuratObject(object.data, project = ”your project name")

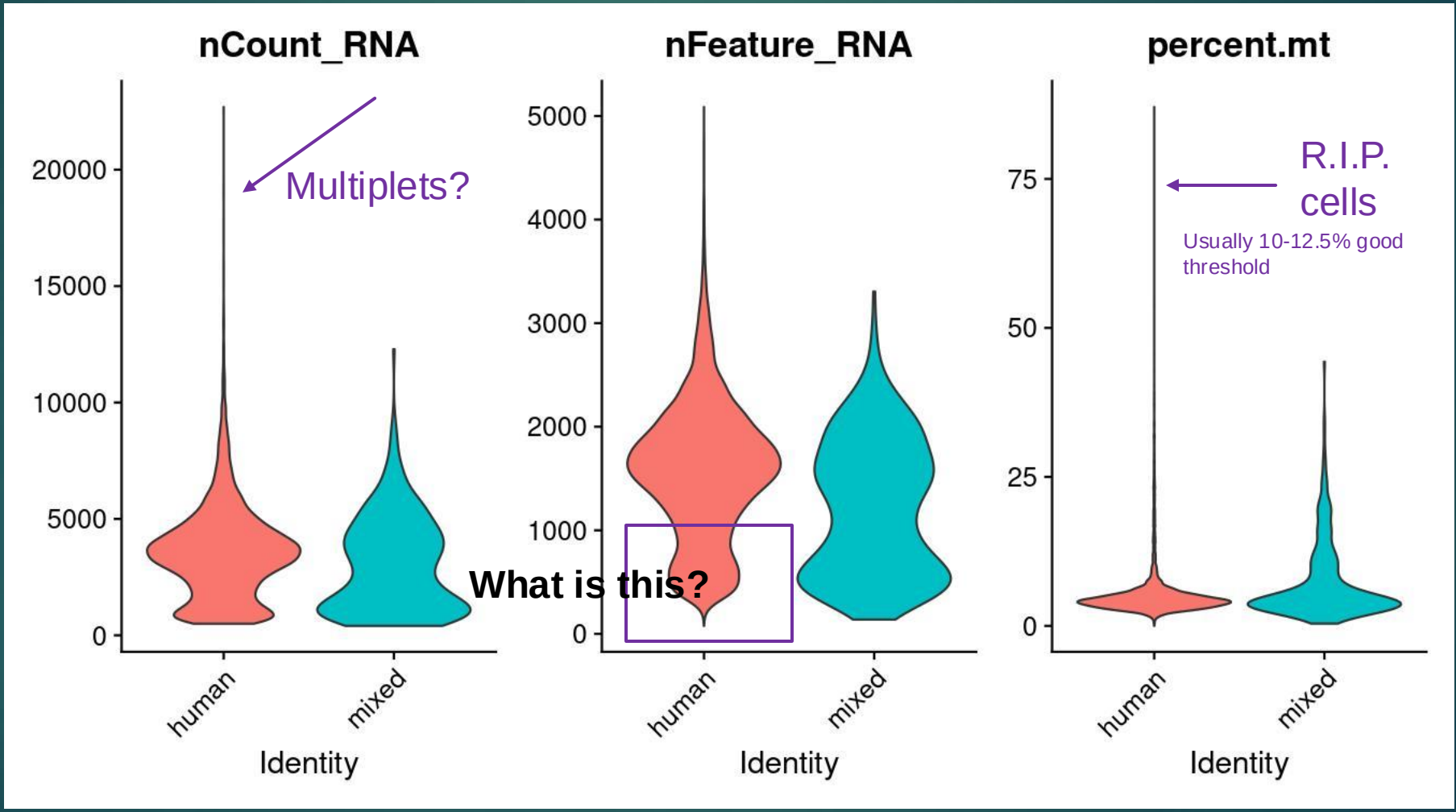
- If you have multiple data modalities you will need to replace object.data with object.data$`Gene Expression`
- You can remove features here- **how does this align with your research question & analysis and downstream algorithms?**

object[["percent.mt"]] = PercentageFeatureSet(object, pattern = "^MT-", assay = "RNA")

- In human genome, all mitochondrial genes are prefixed with “MT”, and exclusion of dead cells is an important step in QC so we want to understand what percentage of counts in each cell are due to mitochondrial genes
- This tells R to look for genes that **start** with MT- note this is for human and mouse genes are named differently
- Some people will also look at ribosomal features

# Seurat-v5

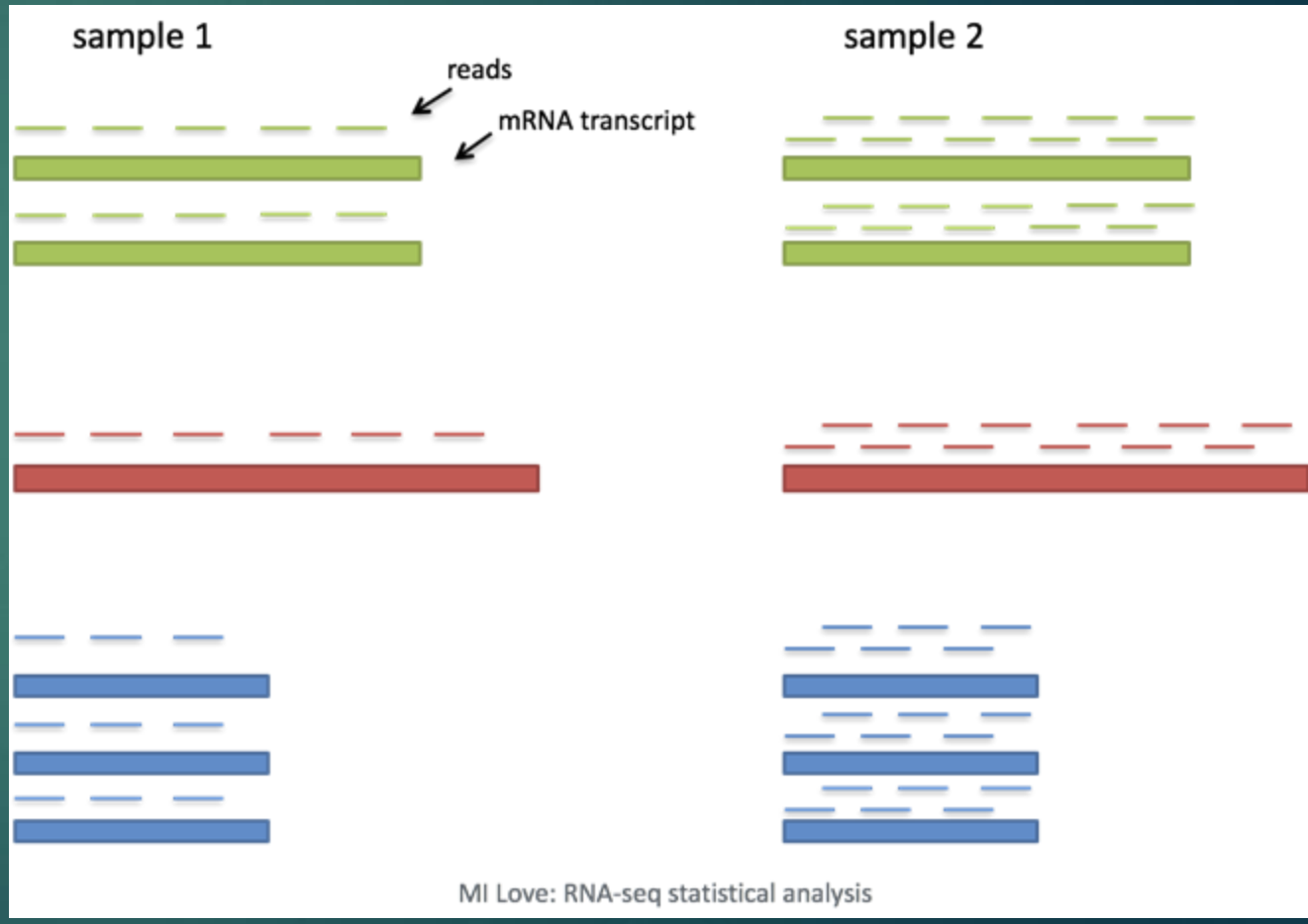
VlnPlot(object, features = c("nCount_RNA", "nFeature_RNA", "percent.mt"), ncol =3, pt.size = 0)




object <- subset(object, nFeature_RNA > 200 & percent.mt < 10 & nFeature_RNA < 5000 & nCount_RNA < 15000)

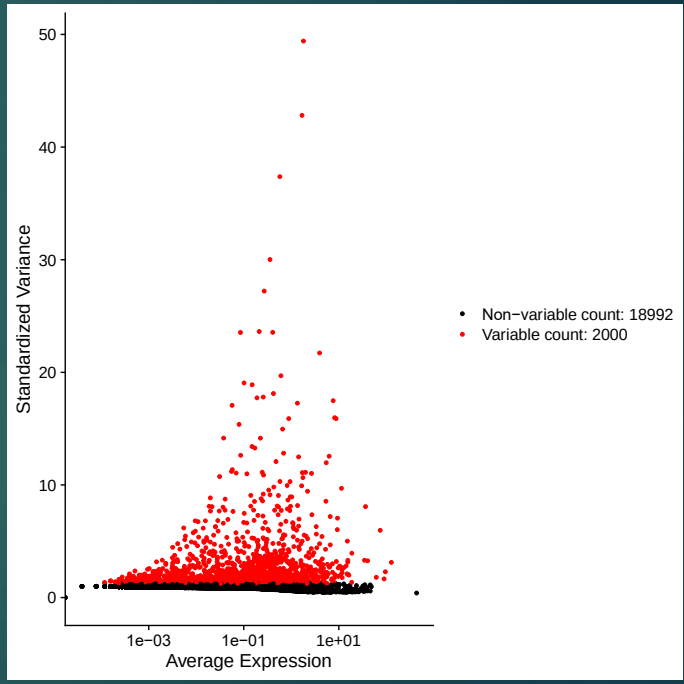
# Normalize, scale, control for unwanted variation

- **Goal: remove technical effects while preserving biological variation**
- Sequencing depth of a cell: total UMI (nCounts)
- Even in the same experiment, different cells have very different sequencing depths
- Expression level of a gene in a cell is proportional to the sequencing depth of the cell, unless the data is normalized to sequencing depth
- Older normalization approach(es) scale every gene in the cell by the same factor: the sequencing depth
- Normalize all cells to equal levels; HOWEVER, this assumes all cells are the same size
  - **NormalizeData**:
    - Divide by total counts in each cell
    - Scale to fixed counts (default is $1x10^4$ use $1x10^6$ for CPM)
    - Add 1
    - natural log



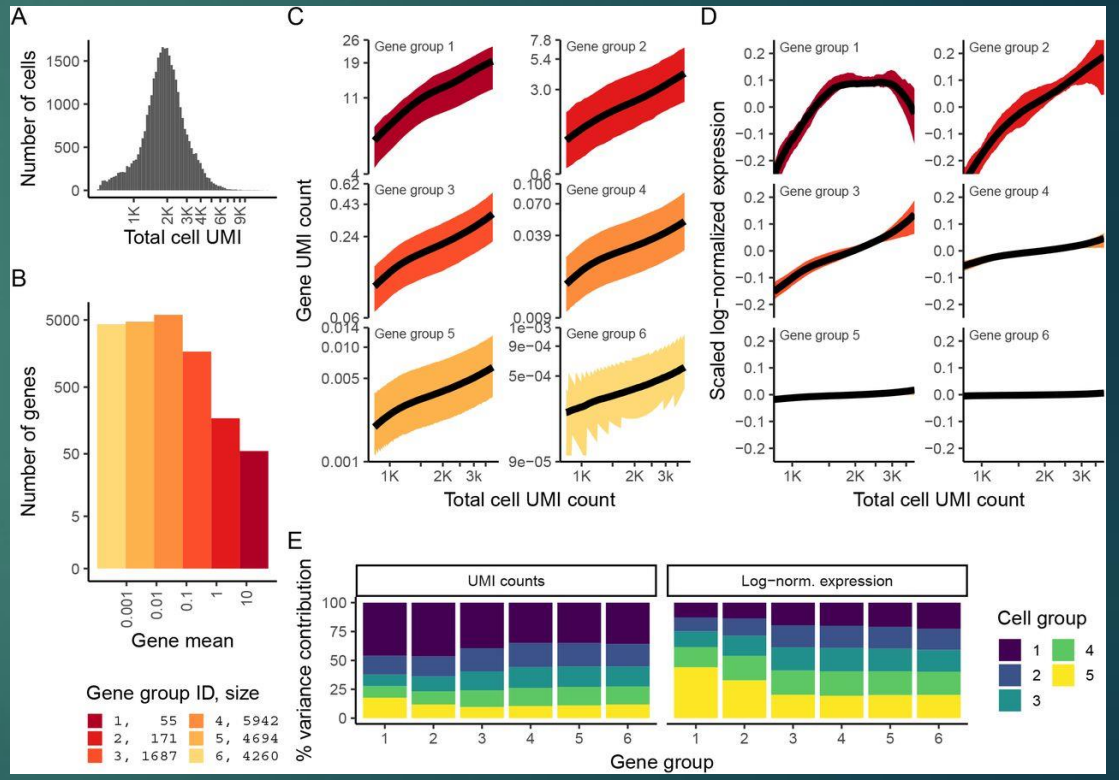
# FindVariableFeatures

- **FindVariableFeatures:** use a variance stabilizing transformation
- vst: Variance-Stabilizing Transformation
  - For each gene, plot log(Var) vs log(mean)
  - Use loess linear regression to calculate expected variance based on observed mean
  - Standardize the expression of each gene so that its variance matches the calculated expected variance
  - Plot standardized variance vs log(mean) and choose outliers
  - Choose 2000-3000 outliers
- mean.var.plot
- Dispersion
- **ScaleData:** subtract mean, divide by standard deviation, remove unwanted signal using multiple regression



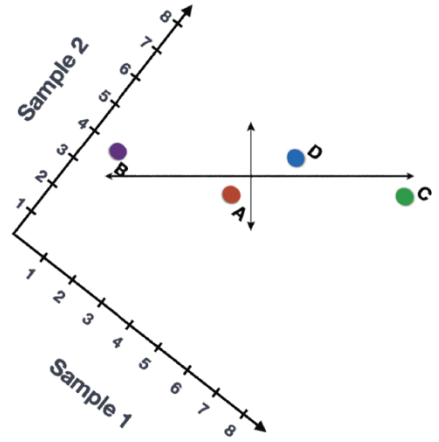
# Regularized negative binomial normalization with Seurat's SCTransform function

- Variance stabilizing transformation used in FindVariableFeatures affects different genes differently
- Highly and lowly expressed genes (B, C) are affected differently by scaling factor (D)
- Variance related to sequencing depth, and traditional normalization unevenly changes contribution of each gene to overall variance
- Solution: Use negative binomial regression (with parameters derived from groups of genes with similar expression) to remove impact of sequencing depth. Seurat function SCTransform.




Hafemeister and Satija. bioRxiv 2019

# Selecting Principal Components



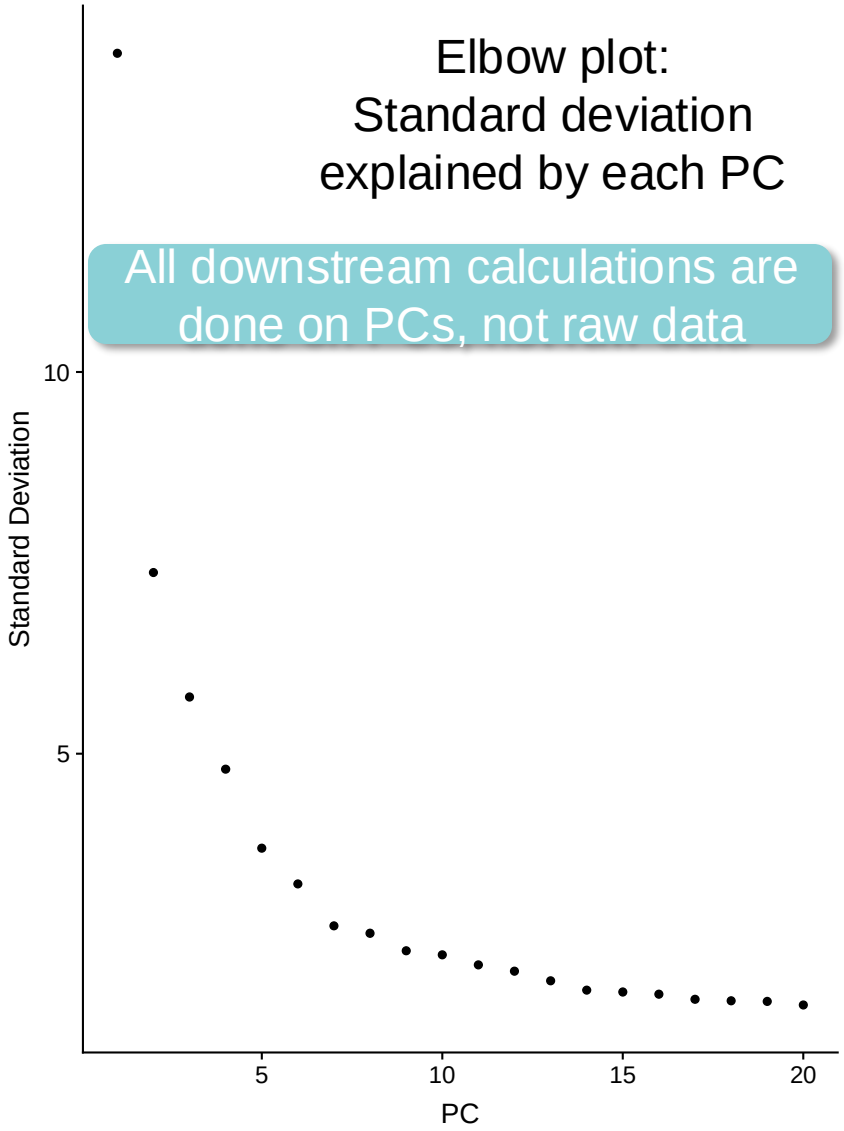

| | Sample 1 | Sample 2 | Influence on PC1 | Influence on PC2 |
|---|---|---|---|---|
| Gene A | 4 | 5 | -2 | 0.5 |
| Gene B | 1 | 4 | -10 | 1 |
| Gene C | 8 | 8 | 8 | -5 |
| Gene D | 5 | 7 | 1 | 6 |

- Retain and plot key information about each principal component (PC):
  - Percentage of standard deviation explained
  - P-value (obtained from bootstrapping “Jackstraw”)
  - Plot gene expression heatmaps for each of the top ~12 principal components
- Choose Principal Components:
  - Purpose: choose relative importance of minor expression signatures
  - Discontinuity in elbow plot
  - All PCs that explain >= x% of SD (e.g. 2%)
  - P-value from JackStraw analysis < $1x10^{-n}$ (e.g. $1x10^{-100}$)
  - Clarity of PC heatmaps

Excellent explanation: https://hbctraining.github.io/scRNA-seq_online/lessons/05_theory_of_PCA.html

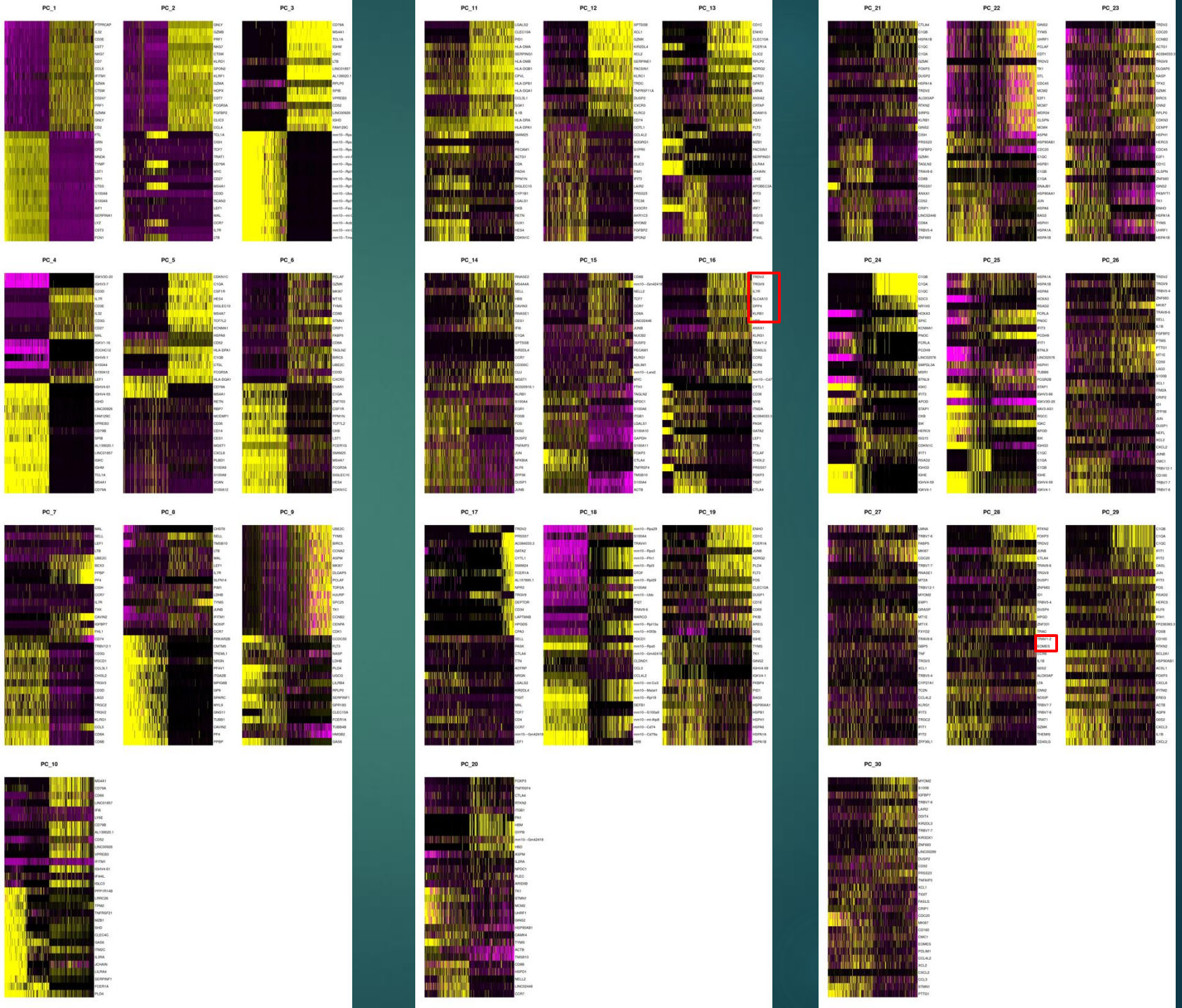
Elbow plot:
Standard deviation explained by each PC

All downstream calculations are done on PCs, not raw data

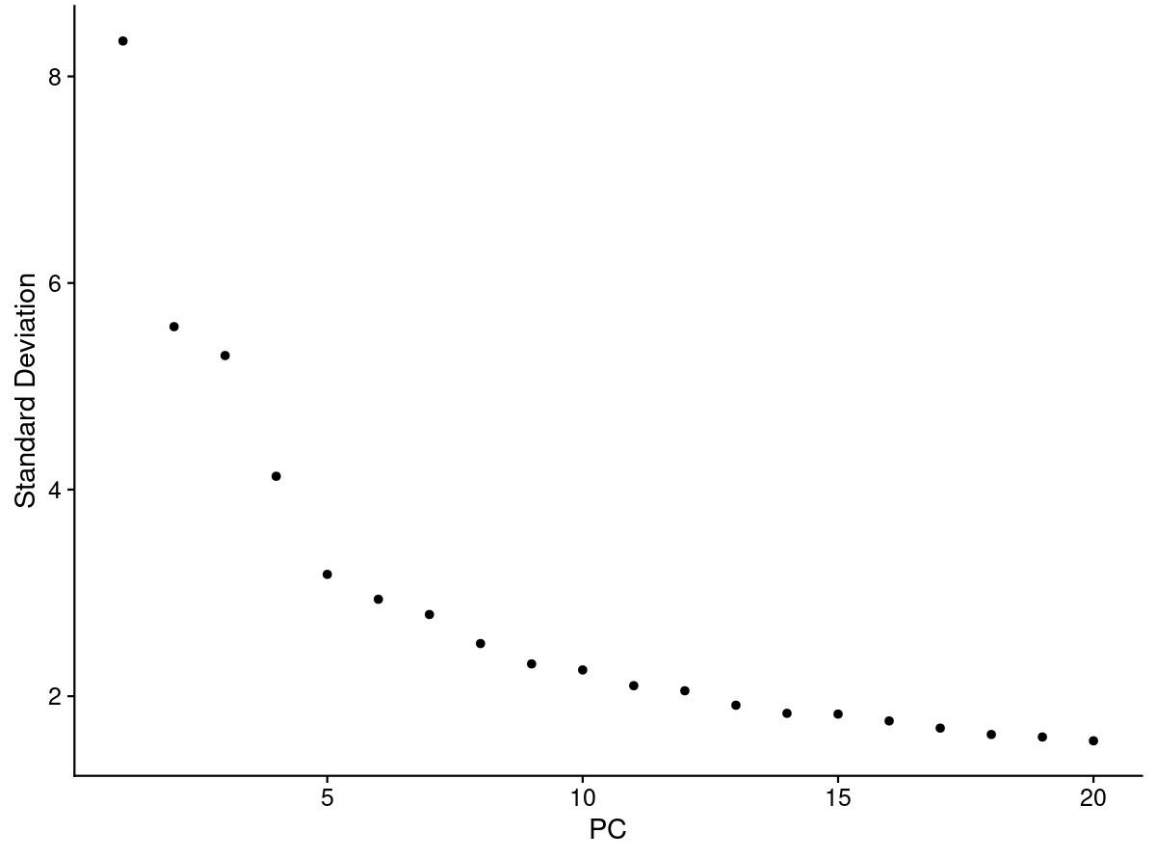


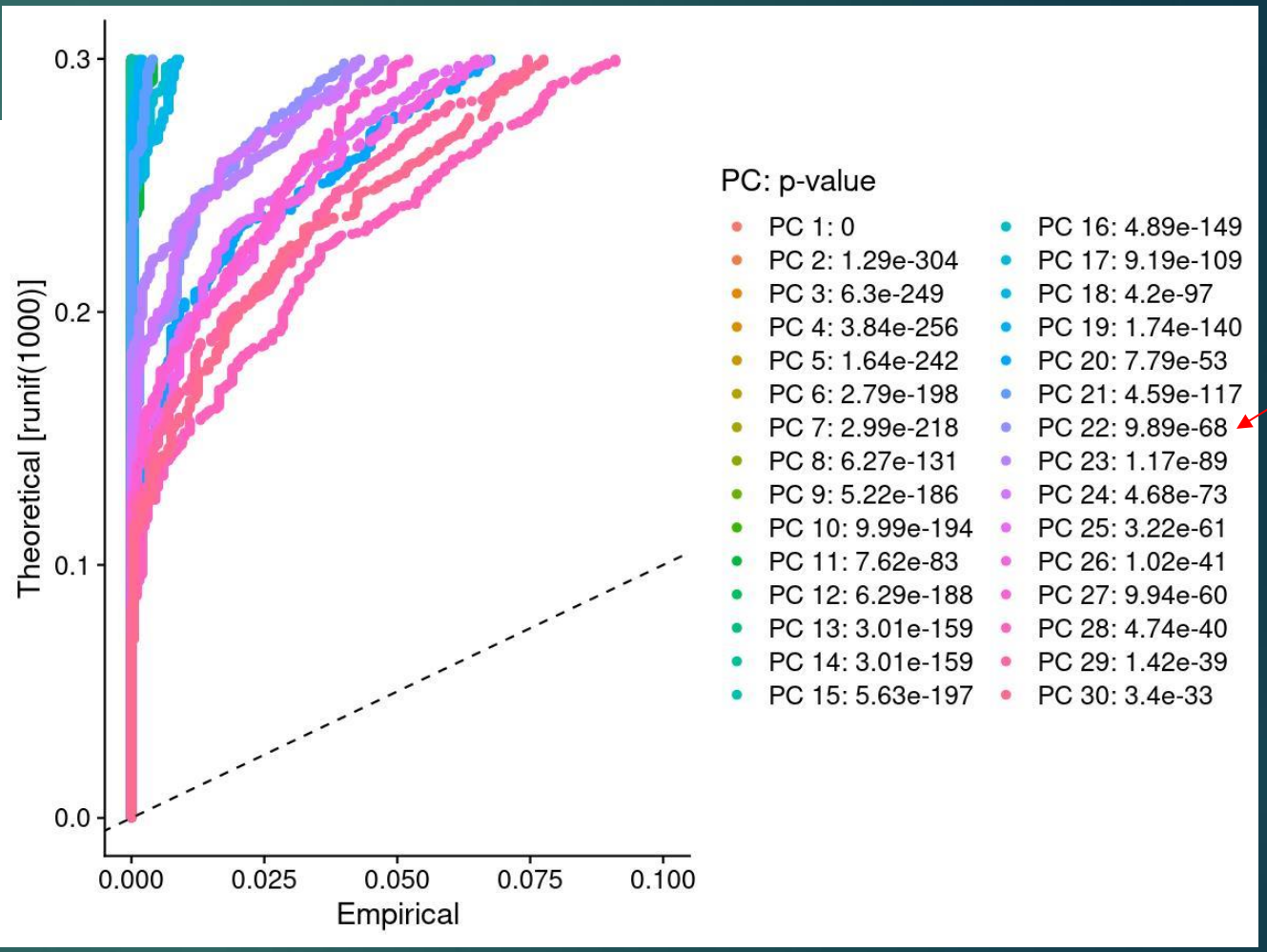
# Choosing PCs

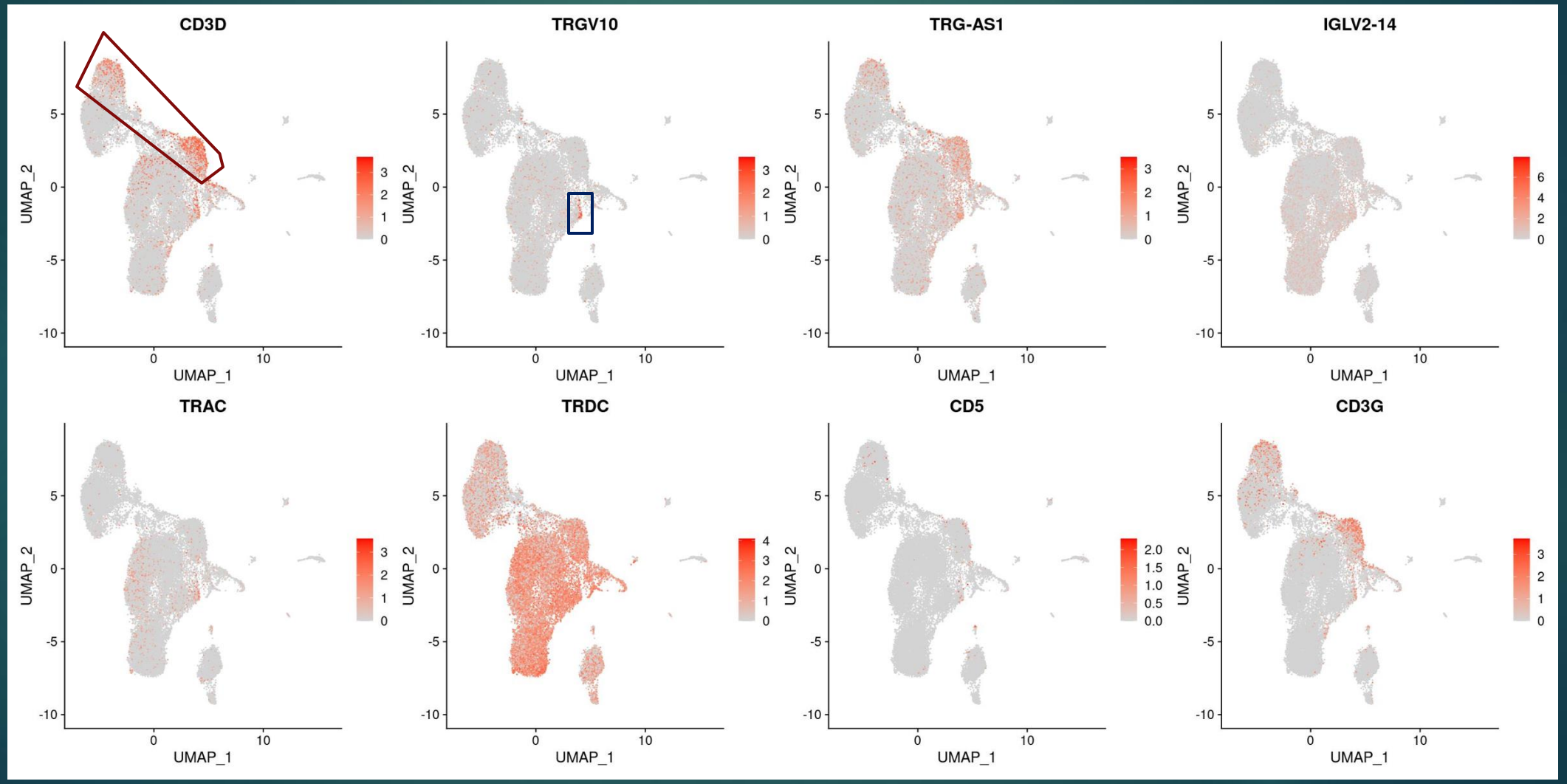
Dim Heatmaps



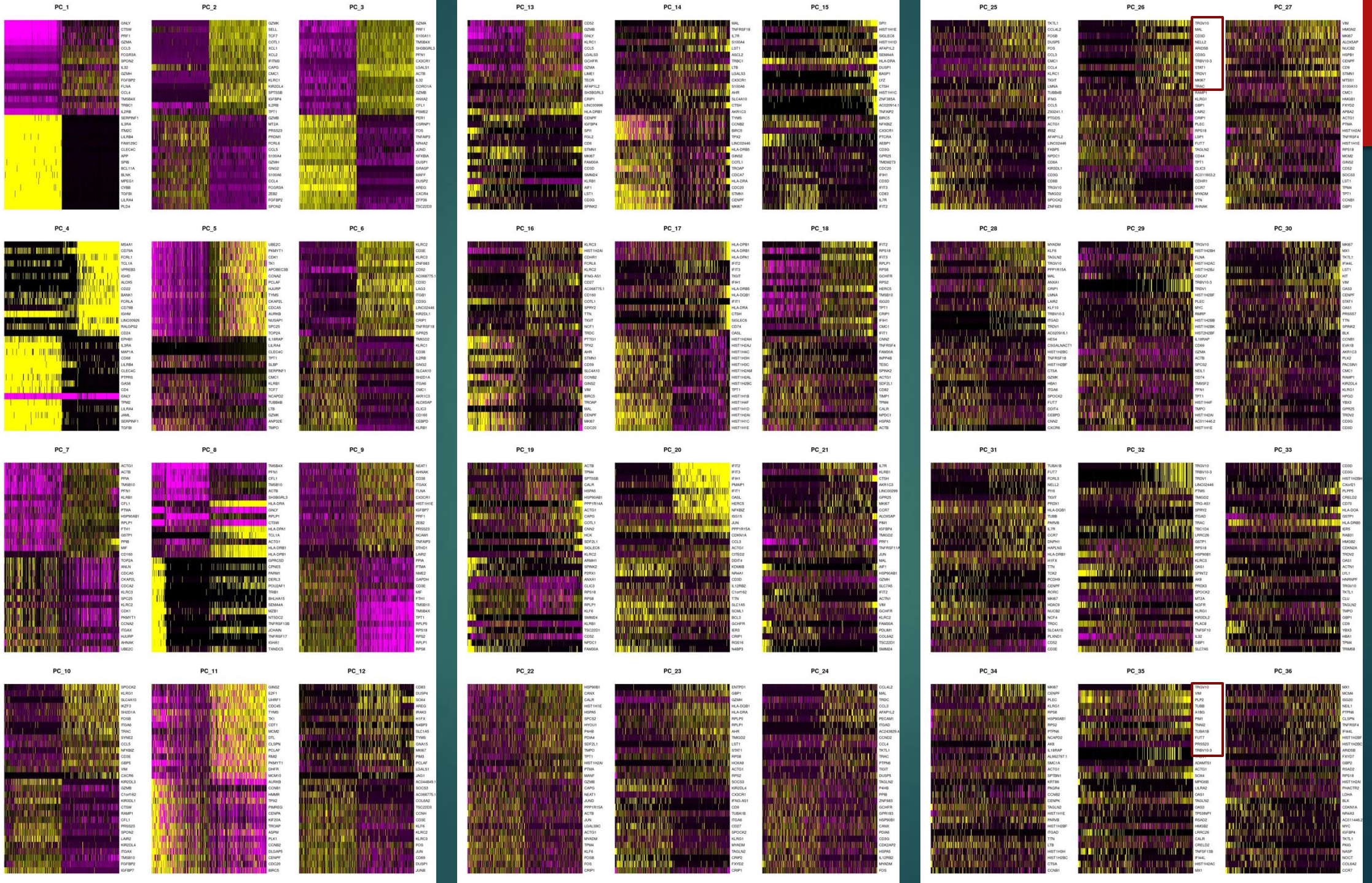
# PC Demo


Standard Deviation
8
6
4
2
5
10
15
20
PC

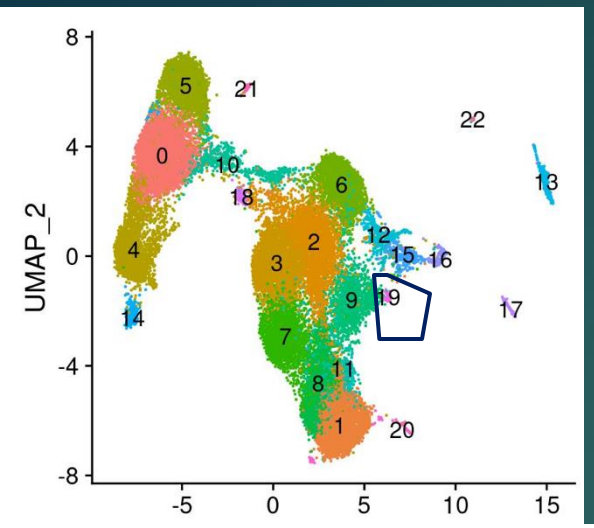

Theoretical [runif(1000)]
0.3
0.2
0.1
0.0
0.000
0.025
0.050
0.075
0.100
Empirical
PC: p-value
PC 1: 0
PC 2: 1.29e-304
PC 3: 6.3e-249
PC 4: 3.84e-256
PC 5: 1.64e-242
PC 6: 2.79e-198
PC 7: 2.99e-218
PC 8: 6.27e-131
PC 9: 5.22e-186
PC 10: 9.99e-194
PC 11: 7.62e-83
PC 12: 6.29e-188
PC 13: 3.01e-159
PC 14: 3.01e-159
PC 15: 5.63e-197
PC 16: 4.89e-149
PC 17: 9.19e-109
PC 18: 4.2e-97
PC 19: 1.74e-140
PC 20: 7.79e-53
PC 21: 4.59e-117
PC 22: 9.89e-68
PC 23: 1.17e-89
PC 24: 4.68e-73
PC 25: 3.22e-61
PC 26: 1.02e-41
PC 27: 9.94e-60
PC 28: 4.74e-40
PC 29: 1.42e-39
PC 30: 3.4e-33



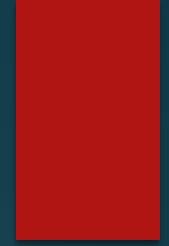

With various cluster resolutions (up to 2.5), the best I could do was get 25% of the cluster to be TRGV10+

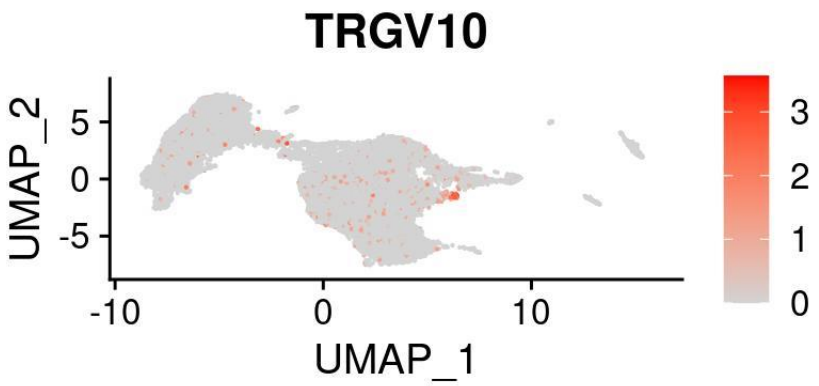

PC_1
PC_2
PC_3
PC_4
PC_5
PC_6
PC_7
PC_8
PC_9
PC_10
PC_11
PC_12
PC_13
PC_14
PC_15
PC_16
PC_17
PC_18
PC_19
PC_20
PC_21
PC_22
PC_23
PC_24
PC_25
PC_26
PC_27
PC_28
PC_29
PC_30
PC_31
PC_32
PC_33
PC_34
PC_35
PC_36

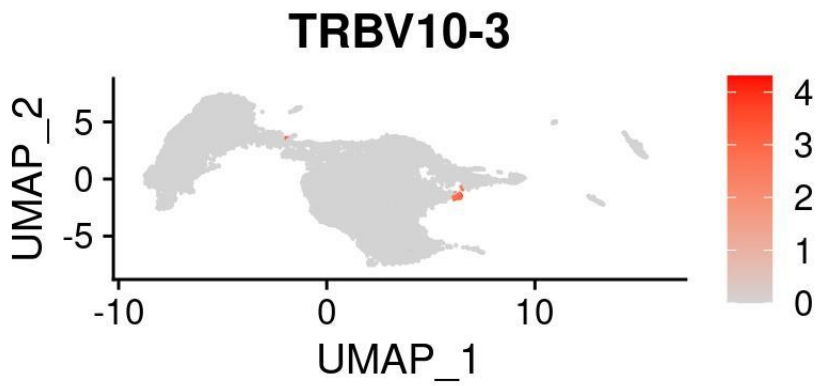
1.0 res

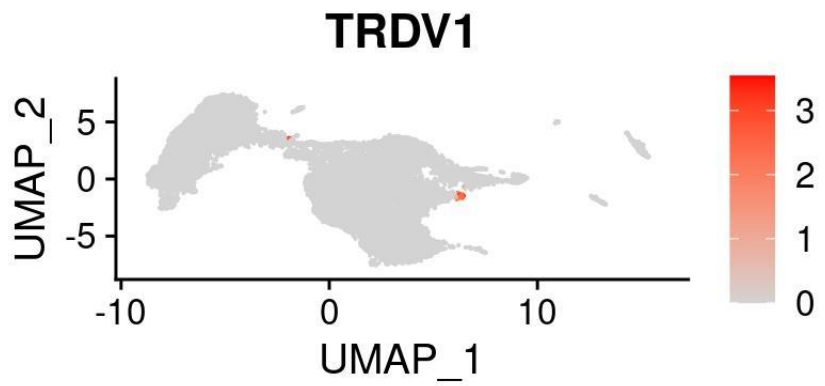

UMAP_2
UMAP_1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22

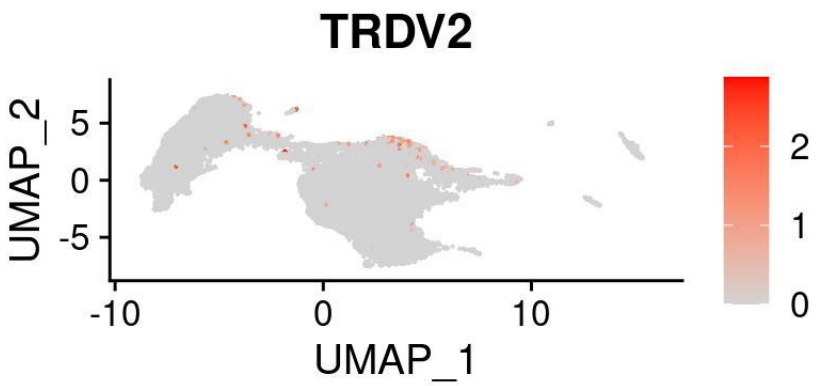
PC35

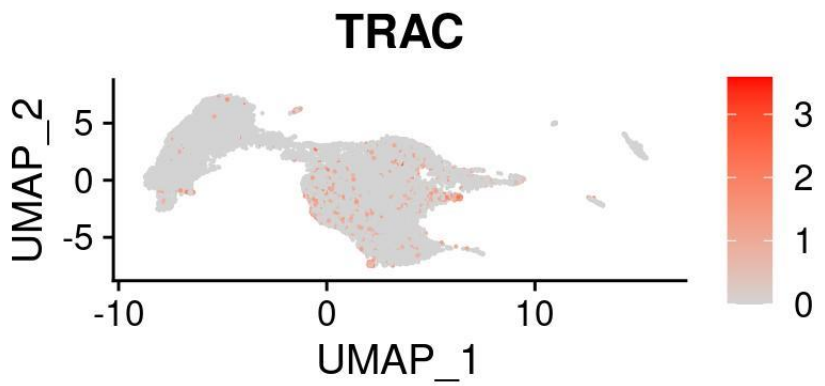


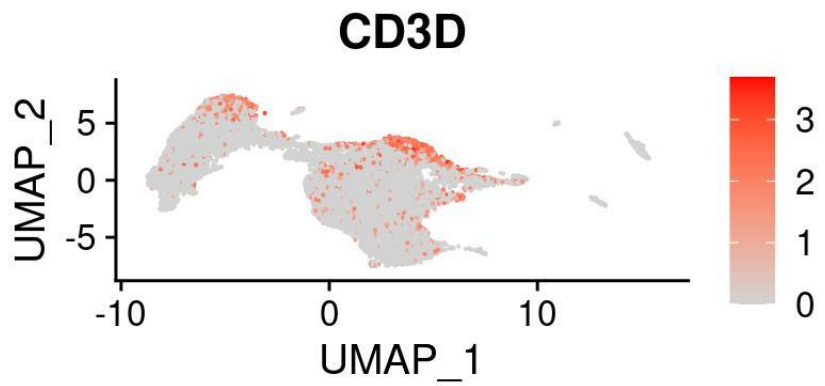

TRGV10
UMAP_2
UMAP_1


TRBV10-3
UMAP_2
UMAP_1


TRDV1
UMAP_2
UMAP_1


TRDV2
UMAP_2
UMAP_1

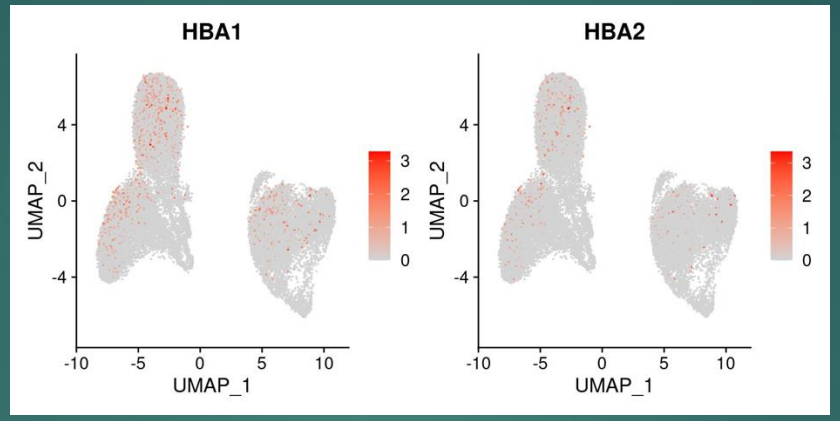

TRAC
UMAP_2
UMAP_1


CD3D
UMAP_2
UMAP_1

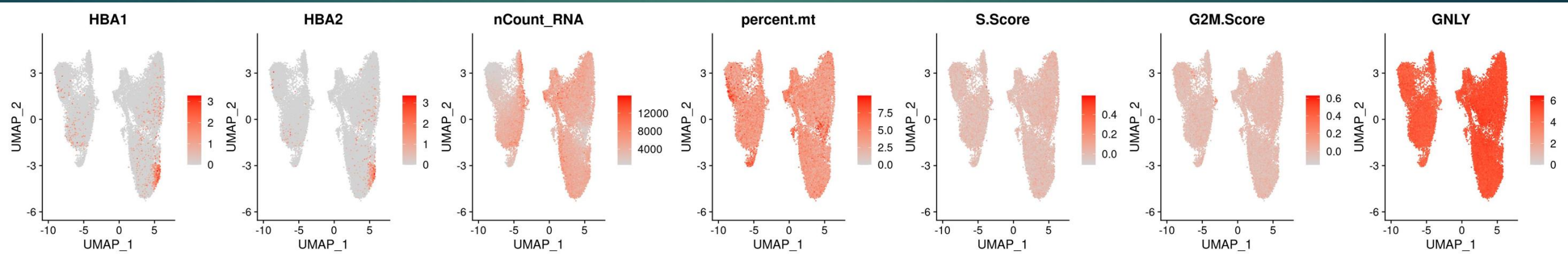
# Is more always better?

## NK Cells

PC14




PC50



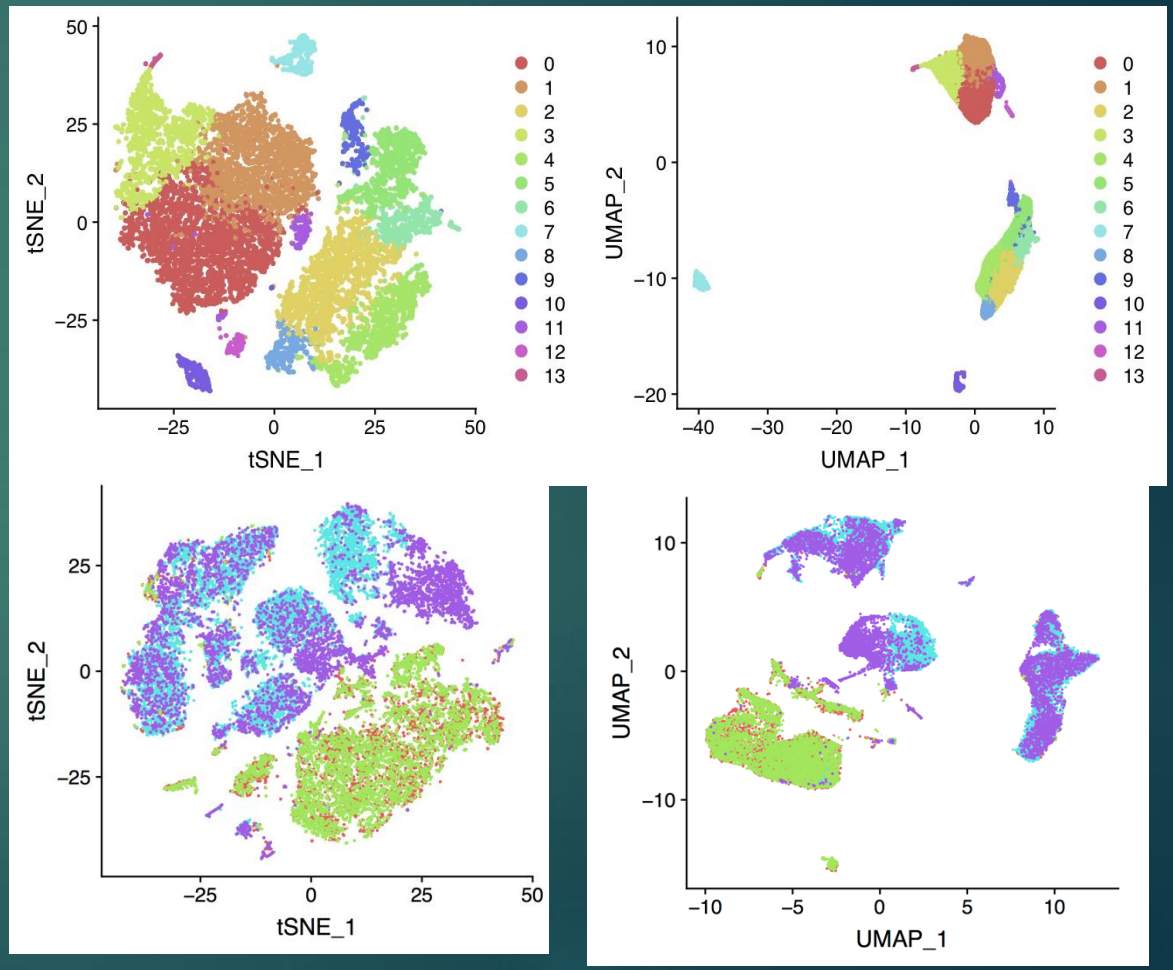
# Plotting using t-SNE/UMAP

t-SNE = *t*-distributed Stochastic Neighbor Embedding
UMAP = Uniform Manifold Approximation and Projection

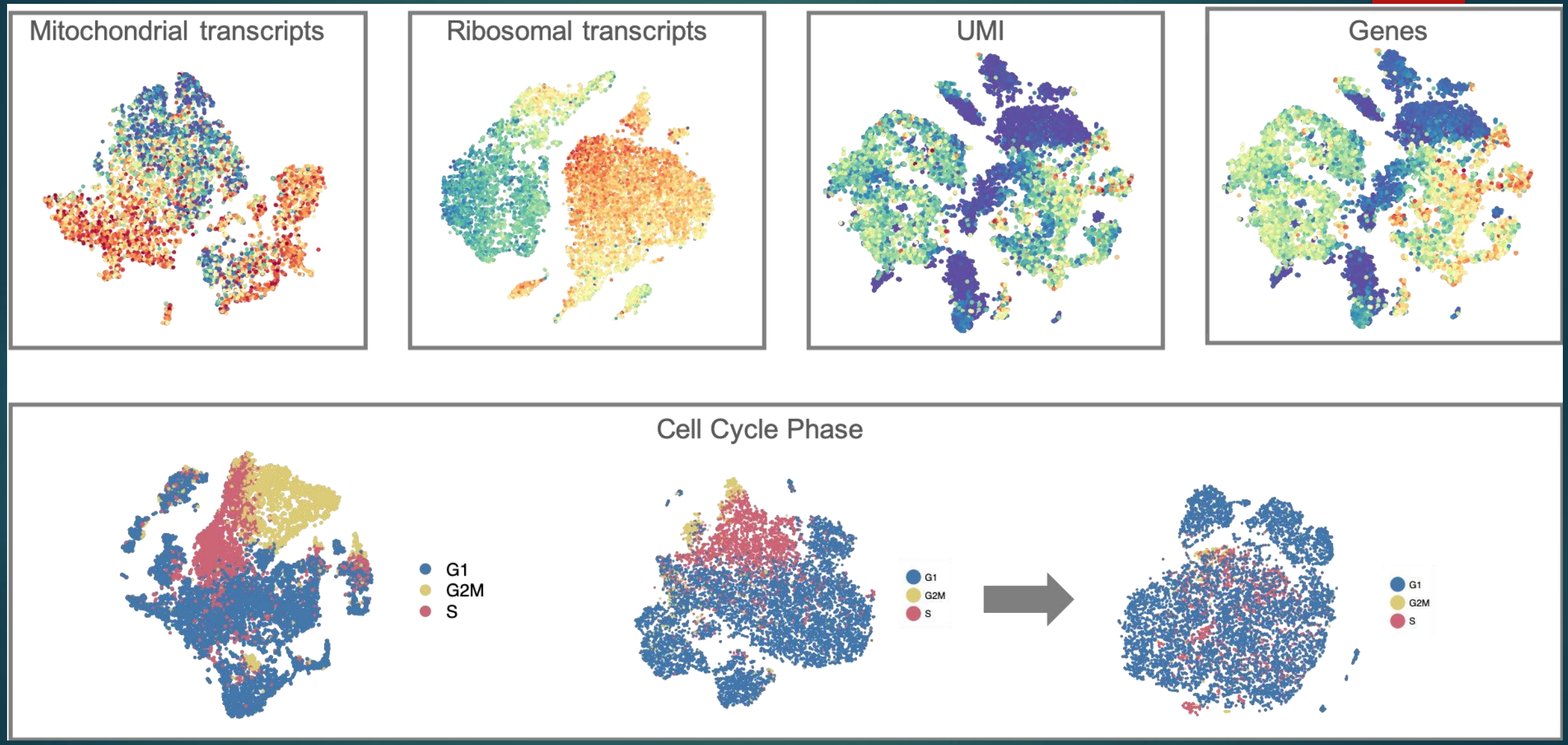
**Goal: Embed high-dimensional data in low-dimensional space**

End product: 2D plot where cells are positioned near each other if they have similar gene expression profiles. “Units” are relative and data-dependent.

- Expression “distances” between points (ie cells) in high-dimensional space are modeled using a gaussian distribution.
- Operates in “PCA space”
- “clusters” are not clusters. This is not clustering.
- t-SNE preserves local structure only.
- UMAP preserves local AND global structure.
- Implication: In UMAP, distances between points *and* clusters are ***more*** interpretable in terms of expression distances/similarity.

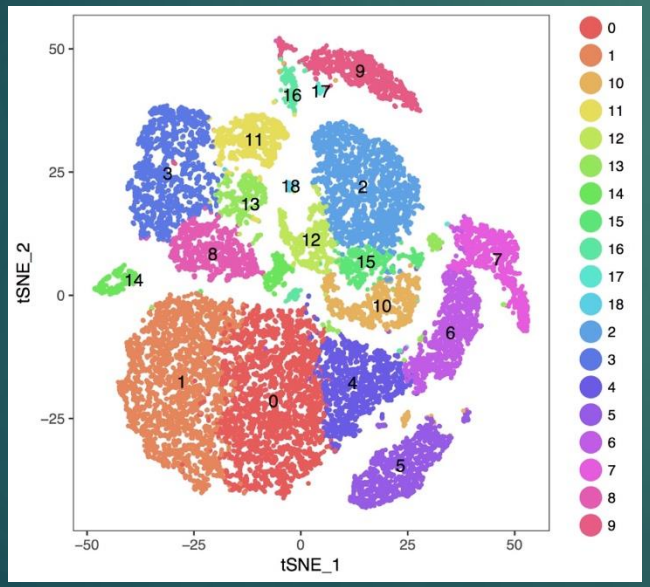


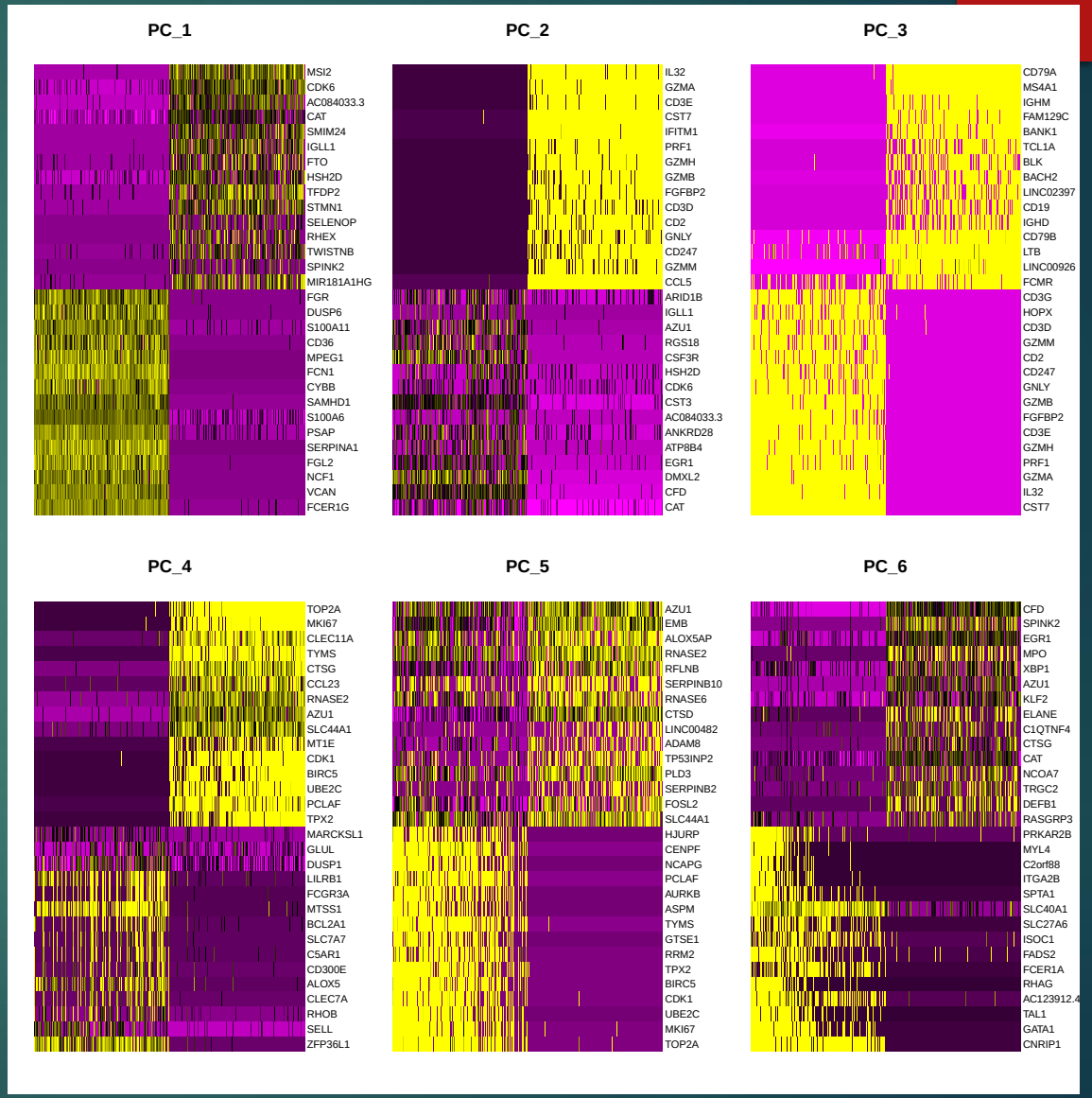
# Interpreting the t-SNE/UMAP, Part I: Potentially misleading sources of variation



# Interpreting the t-SNE/UMAP, Part II: Systematic analysis of variation

- What is driving the t-SNE/UMAP layout?
- Find genes that vary:
  - Principal components
  - Individual cluster-specific genes
- Examine across clusters/t-SNE/UMAP
- **http://bioconductor.org/books/3.15/OSCA.basic/dimensionality-reduction.html**

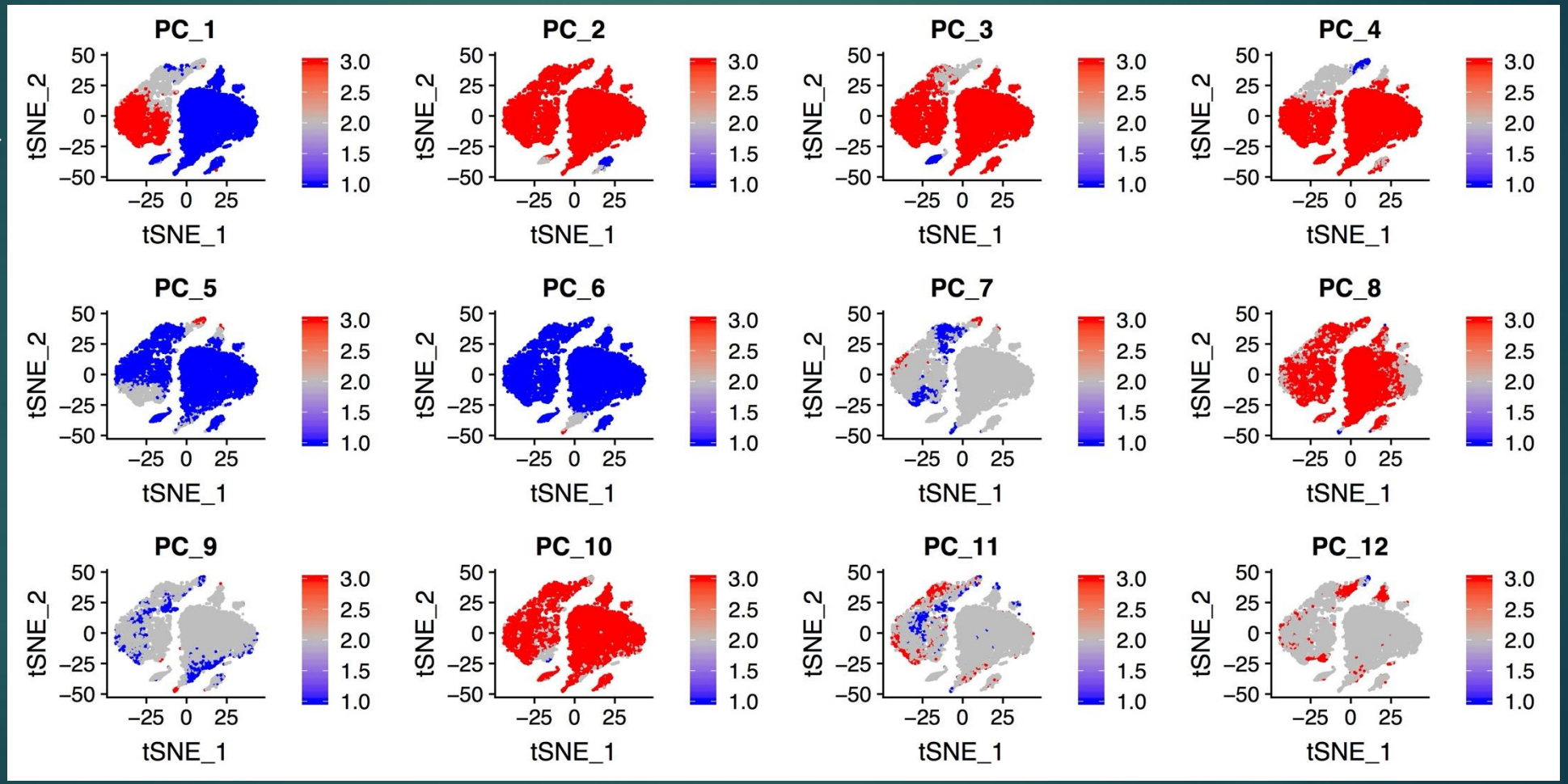


# Part II, cont'd: Visualizing sources of variation

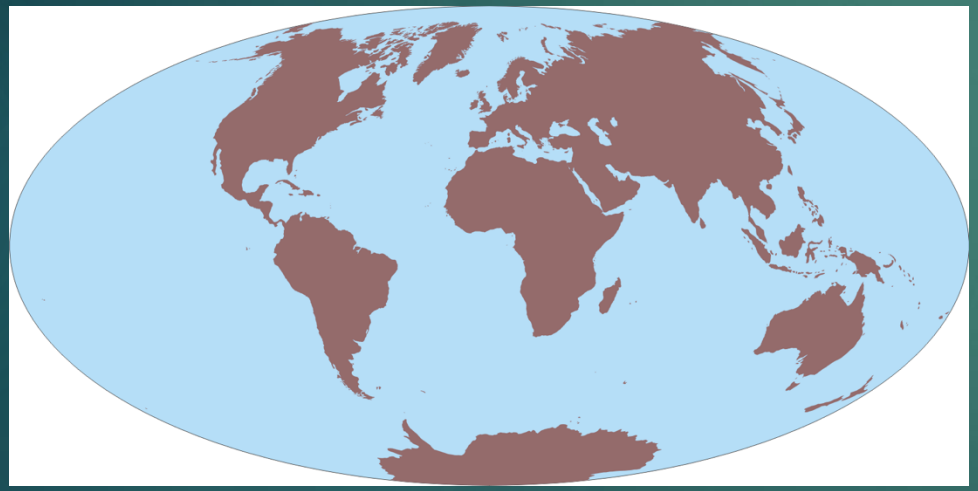
Small but distinct cell clusters may contribute heavily to the layout (may need to be removed)

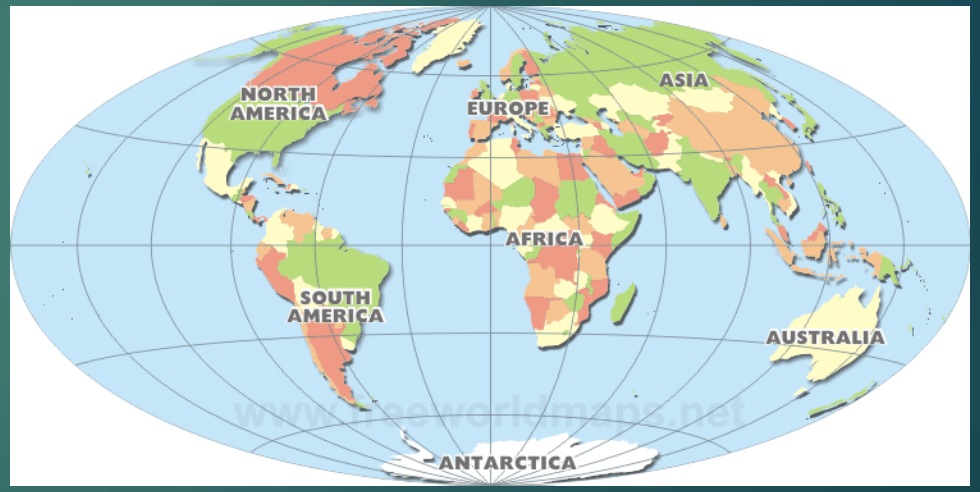
PC #1 captures the biggest source of variation



# 2-D layout vs. Clustering

- tSNE and UMAP reflects natural organization of data by approximating high-dimensional relationships in low-dimensional space
- Clustering imposes structure by assigning cells to non-overlapping groups based on relative expression similarity





# Clustering 101

- Clustering largely helps with being able to run stats
  - Very few DE packages out there that can work without calling clusters
    - Single cell haystack
  - Multiple ways to go about these:
    - K clustering- you decide how many clusters (k) you want the algorithm to decide
    - You decide everything: e.g. lasso clustering, circling cells you want to group together
    - Threshold clustering: by expression of a gene or subset of genes:
      - E.g. subset(object, CD3D <0 & NCAM-1 > 0) for NK cells (just an example!)
      - Challenging due to the noise of the data- do you subset on raw counts, normalized data, etc? What about cells that should be positive but aren't?
    - "unsupervised" graph-based clustering- can be Louvain, leiden, smart local moving-different methods called under the hood to get to the same goal
      - Users specifies a "resolution"- higher = more clusters, lower = less clusters: usually higher needed with more cells and/or more heterogeneity; and the inverse is also true

# Coming to terms with your clusters

- ”unsupervised” graph-based clustering- can be Louvain, leiden, different methods called under the hood to get to the same goal
  - Users specifies a “resolution”- higher = more clusters, lower = less clusters: usually higher needed with more cells and/or more heterogeneity; and the inverse is also true
  - You can always have more or less clusters:
  - Justification:

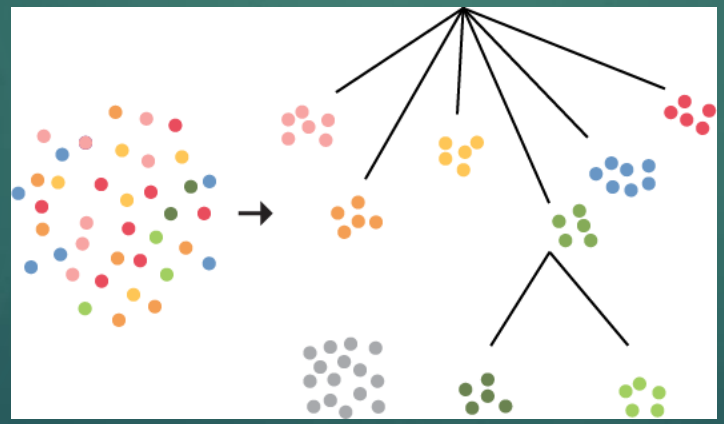
## High-Dimensional Single-Cell Analysis Identifies Organ-Specific Signatures and Conserved NK Cell Subsets in Humans and Mice


### Abstract

Natural killer (NK) cells are innate lymphoid cells (ILCs) involved in antimicrobial and antitumoral responses. Several NK cell subsets have been reported in humans and mice, but their heterogeneity across organs and species remains poorly characterized. We assessed the diversity of human and mouse NK cells by single-cell RNA sequencing on thousands of individual cells isolated from spleen and blood. Unbiased transcriptional clustering revealed two distinct signatures differentiating between splenic and blood NK cells. This analysis at single-cell resolution identified three subpopulations in mouse spleen and four in human spleen, and two subsets each in mouse and human blood. A comparison of transcriptomic profiles within and between species highlighted the similarity of the two major subsets, NK1 and NK2, across organs and species. This unbiased approach provides insight into the biology of NK cells and establishes a rationale for the translation of mouse studies to human physiology and disease.

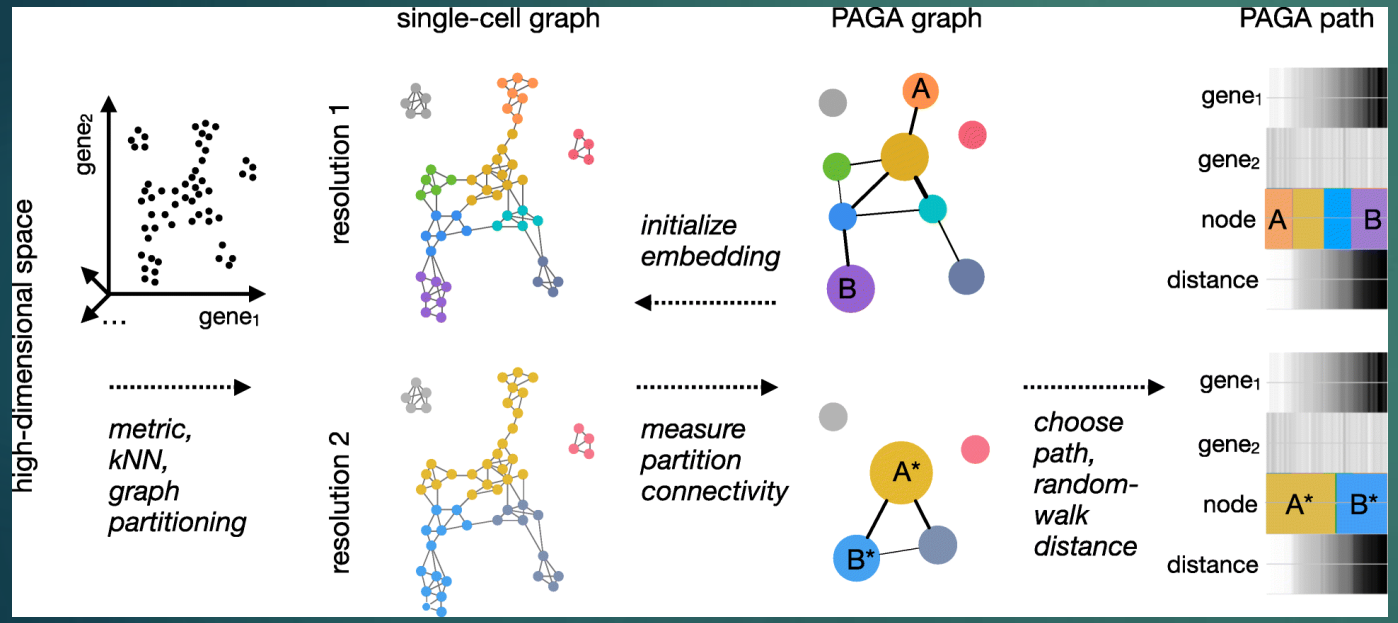
# Cluster Assignments Continued

- Variety of packages to assign cells to a cell type for you (e.g. a reference)
  - Ex. SingleR, Garnett, Seurat TransferData
  - Some care about your clustering while others are cluster agnostic
  - Cluster ”extend” decreases noise
  - Cluster agnostic pro is no need to cluster prior to using the reference



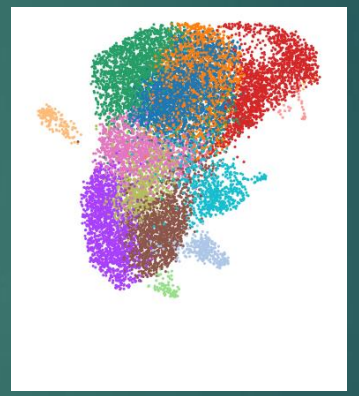
garnett

# Partition-Associated Graph Abstraction (PAGA)

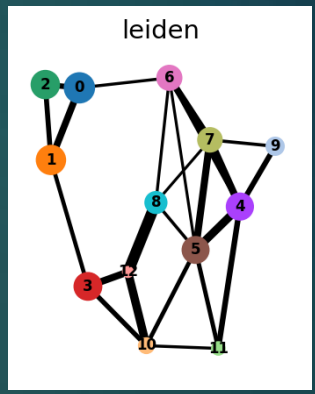



Example:

leiden



PAGA








Available in R through: https://github.com/zhanghao-njmu/SCP

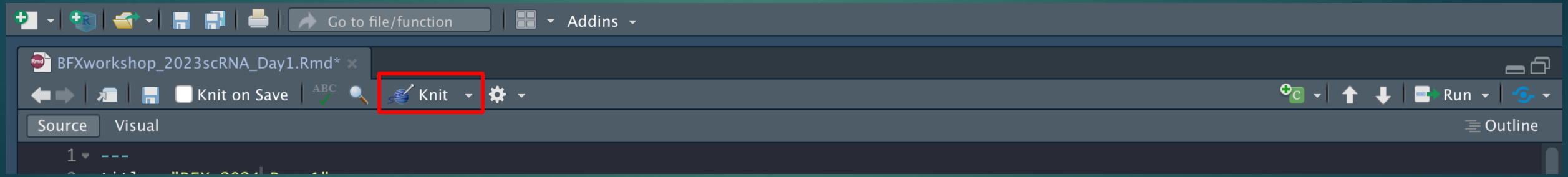
Wolf, F.A., Hamey, F.K., Plass, M. *et al.* PAGA: graph abstraction reconciles clustering with trajectory inference through a topology preserving map of single cells. *Genome Biol* **20**, 59 (2019). https://doi.org/10.1186/s13059-019-1663-x

# Resources

- https://rnabio.org/module-08-scrna/0008/02/01/scRNA/
- http://bioconductor.org/books/release/OSCA/
- https://hbctraining.github.io/scRNA-seq_online/lessons/01_intro_to_scRNA-seq.html
- PCA: https://www.youtube.com/watch?v=FgakZw6K1QQ&t=0s
- https://www.sc-best-practices.org/preamble.html

# Homework

- Run Rmarkdown document for this week
  - 1st download the dataset specified earlier
  - Change the directory path in the R markdown document to where the file is located on your computer
  - Read through the script, noting you only need to change the top variables if you'd like to (recommended to gain experience in how they can affect your data)
  - Then Knit to create a html print out of your work!




Next week: Differential expression, batch correction, & handling CITE-seq data