Introduction to scRNA-seq

Stem Cell Network Workshop

David Cook
Postdoctoral Fellow, Wrana Lab
Lunenfeld-Tanenbaum Research Institute
@DavidPCook | david.cook@uottawa.ca

Design

1. Why single cells?
2. Single-cell platforms
3. Sample preparation
4. Design considerations
   - Cost
   - How deep to sequence?
   - How many cells?
   - How many samples?

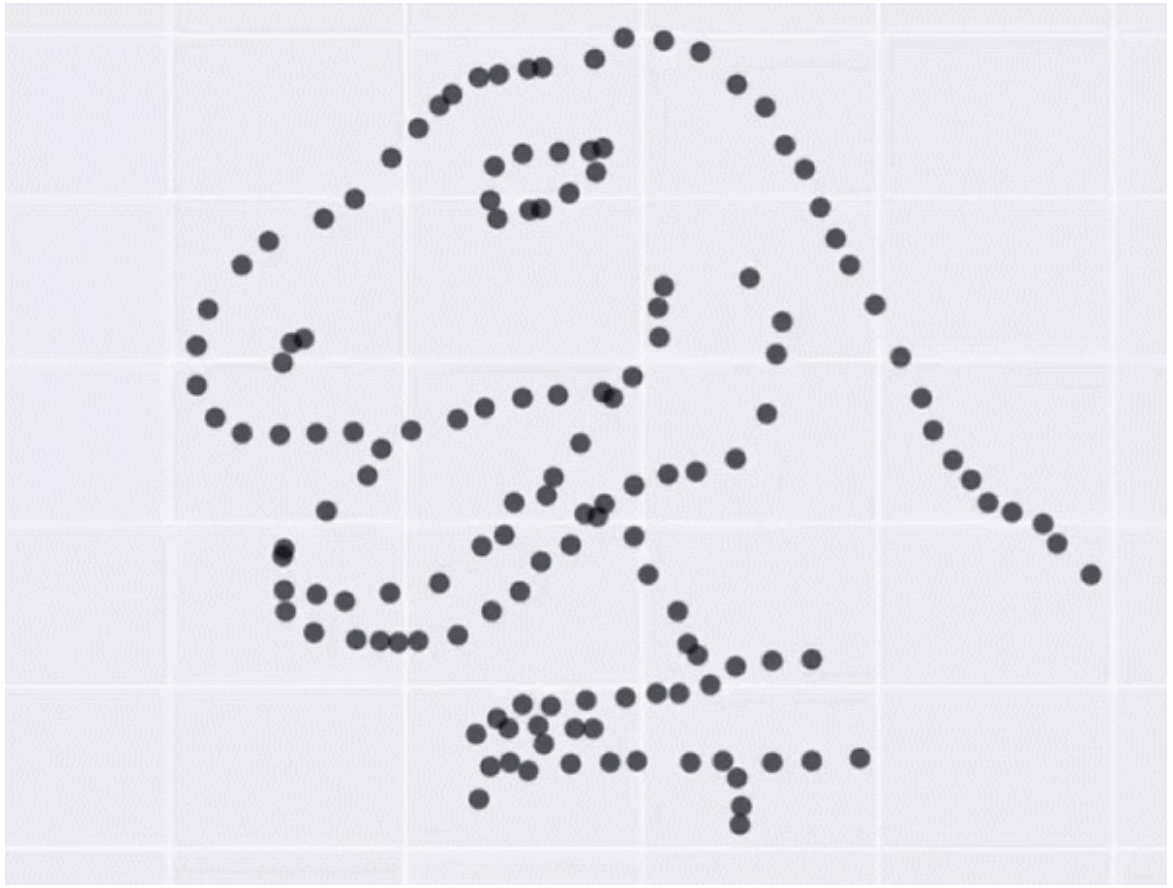
Analysis

1. Choosing your ecosystem
2. Standard preprocessing
   - Quality Control
   - Normalization
   - Feature selection
   - Dimensionality reduction
   - Clustering & Annotation
3. To integrate or to not integrate?
4. Downstream analysis – biological priors help!
   - Differential expression
   - Trajectory inference
   - Signalling / transcription factor inference
   - Cell-cell communication
5. Discussion

Design

Why single cells?

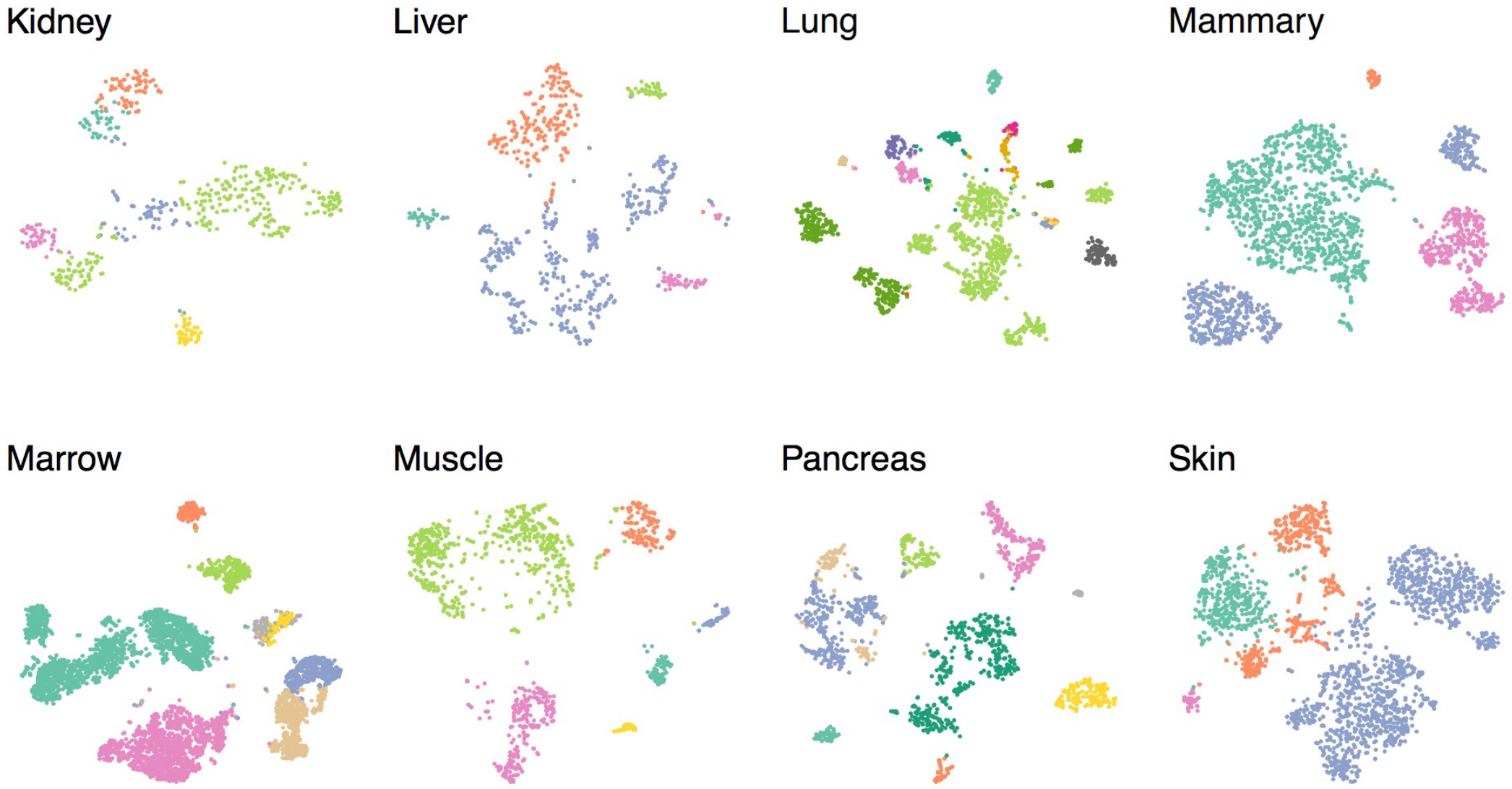
Accuracy—average measurements from complex distributions are meaningless



X Mean:	54.26
Y Mean:	47.83
X SD:	16.76
Y SD:	26.93
Corr:	-0.06

Matejka and Fitzmaurice (Autodesk Research, Toronto)

Biological systems are complex – Tissue Heterogeneity

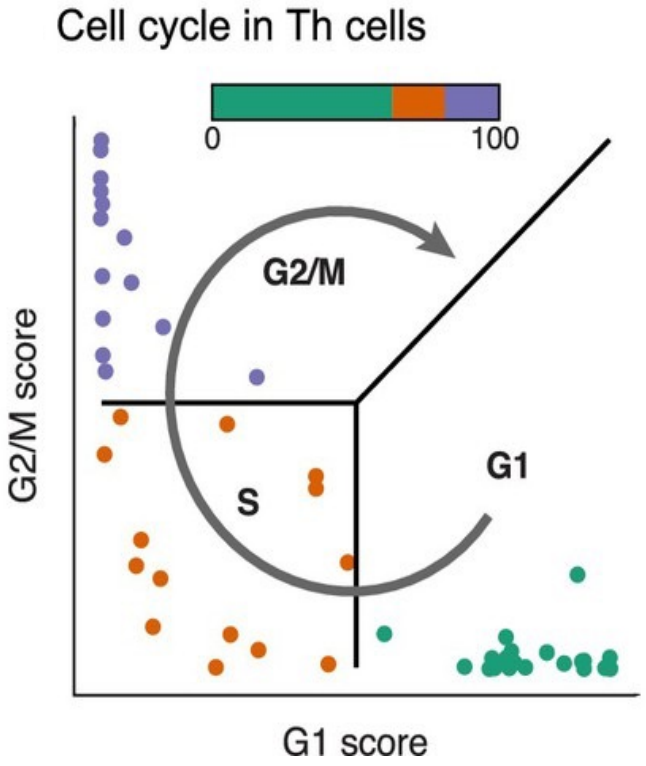


The Tabula Muris Consortium, *BioRxiv,* 2017

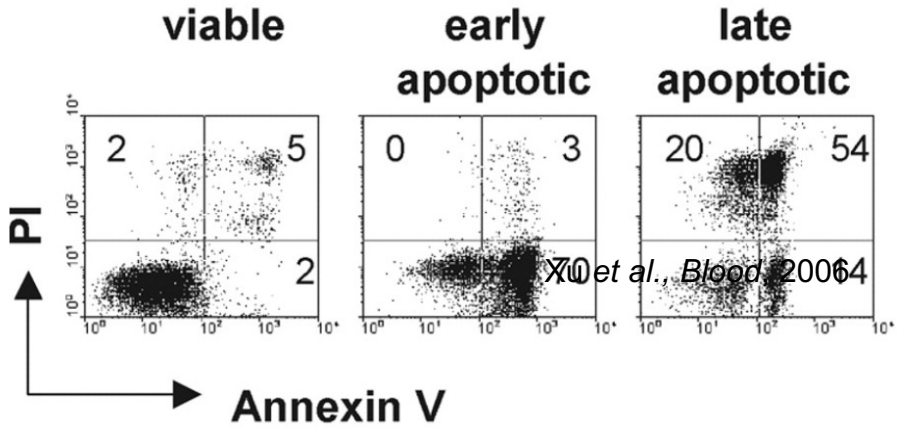
Biological systems are complex – Cellular Heterogeneity

Cell Cycle

The Human Cell Atlas, *eLife,* 2017



Apoptosis/Senescence

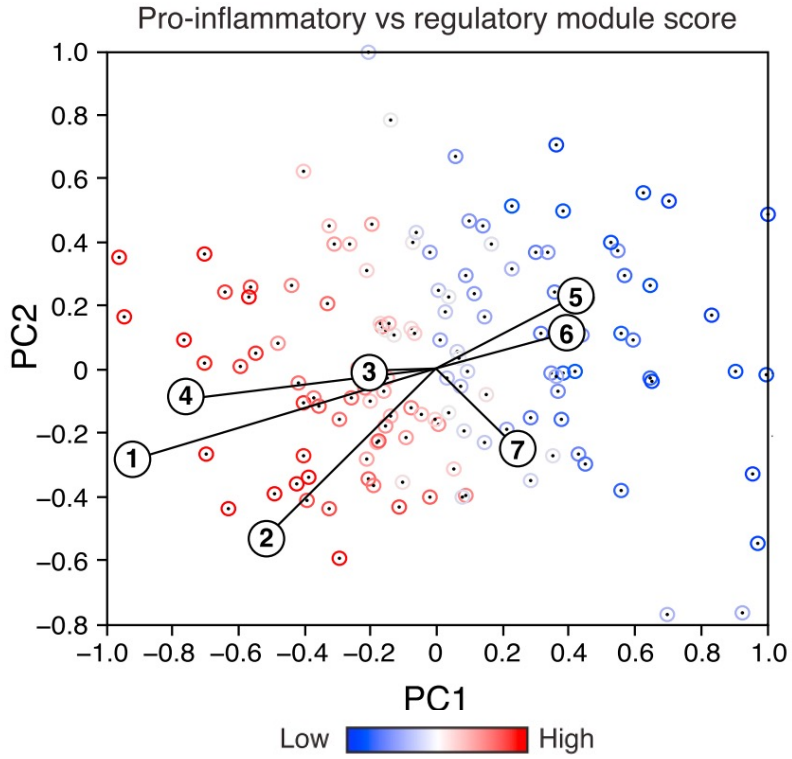


Xu *et al.*, *Blood* 2006

Phenotypic spectrum

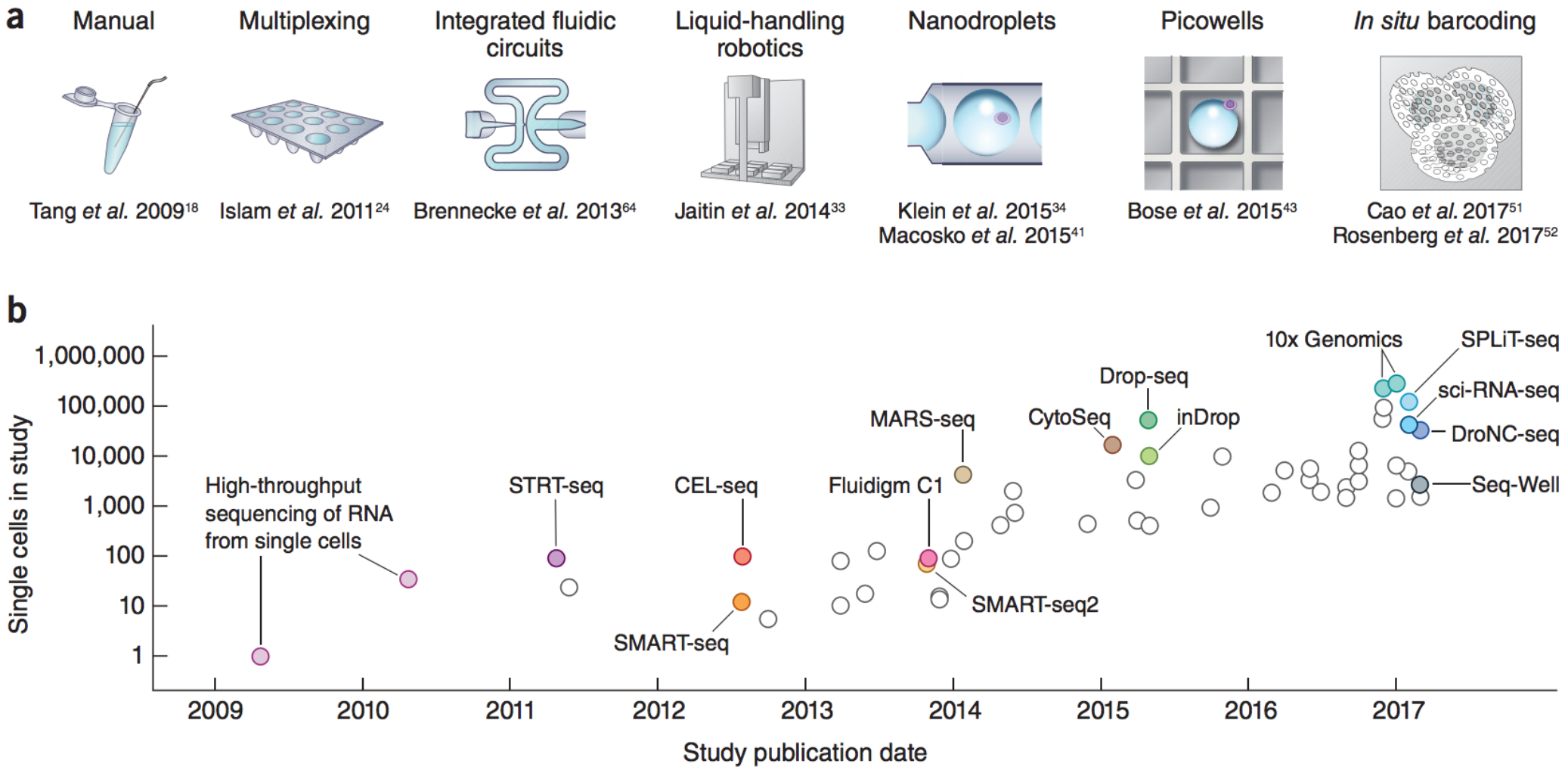
(eg Th17 T-cells)

Gaublomme *et al.*, *Cell,* 2015



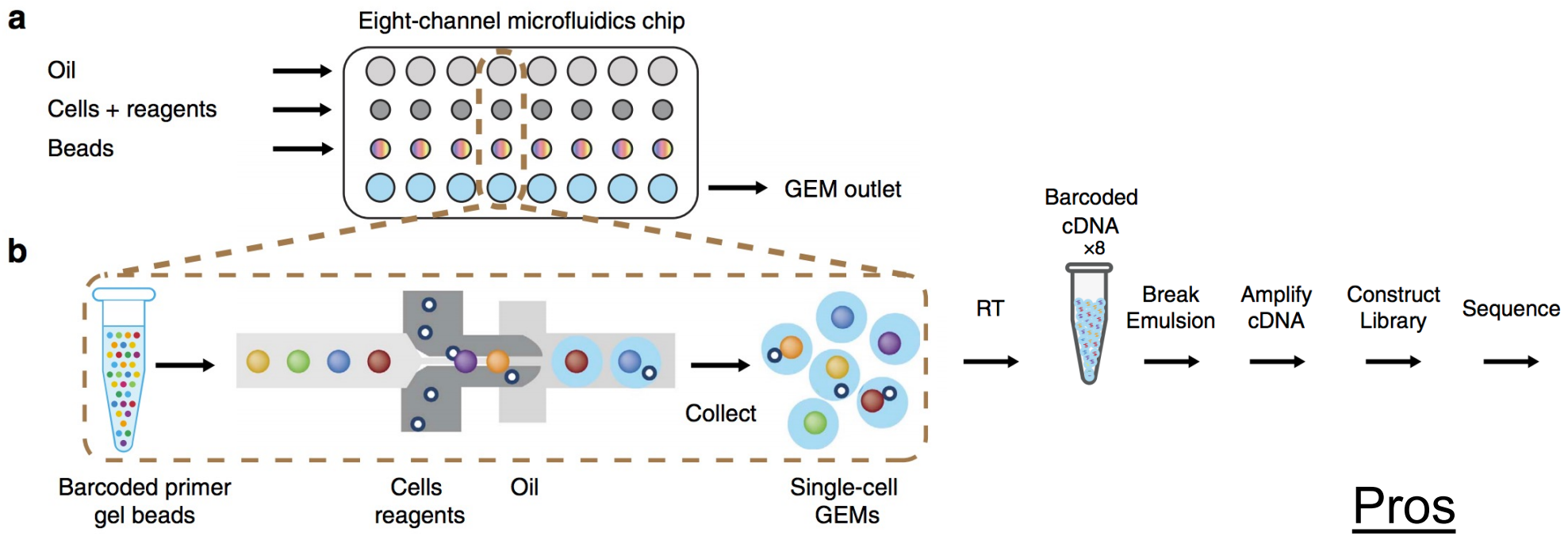
Single Cell Platforms

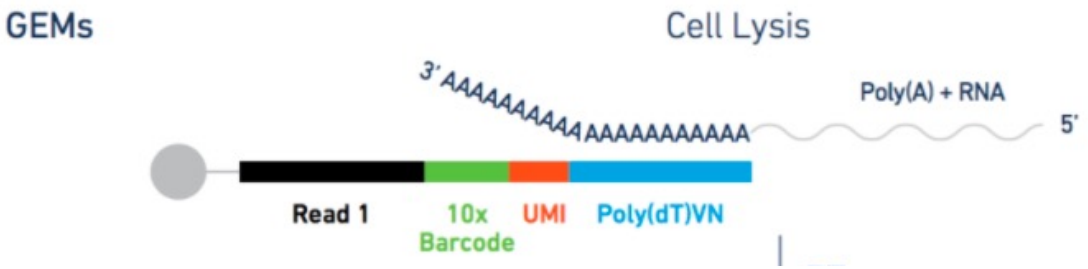
Exponential increase in throughput



Svennson *et al.*, *Nature Protocols,* 2018

Main platforms – Droplet methods (eg. 10x Genomics Chromium, Drop-Seq)





Pros

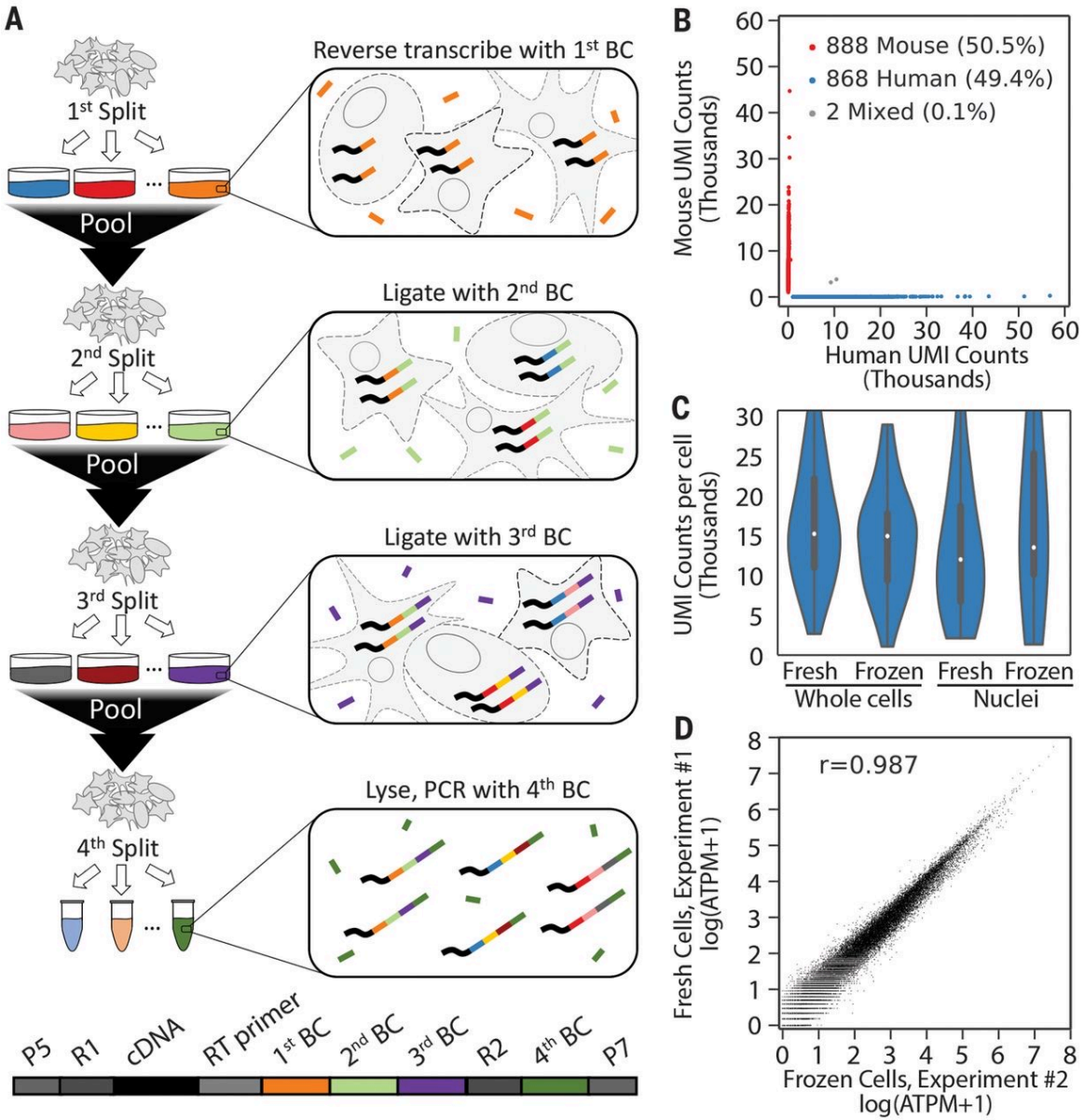
Very high throughput
Up to 8 unique samples per run
System cost relatively low

Cons

Limited customizability

Zheng *et al., Nature Comm,* 2017

Main platforms – Combinatorial indexing (eg. Parse Biosciences, sciRNA-seq)



Pros

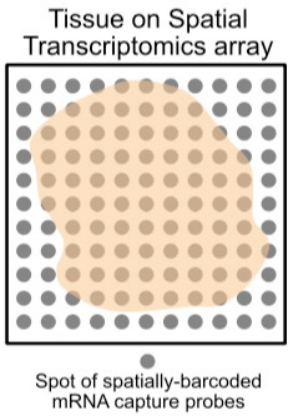
No hardware/microfluidics needed
More customization
Scalable
Requires fixation

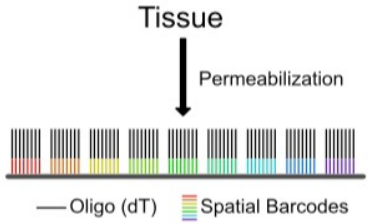
Cons

Much more sample handling than droplet methods
Requires fixation

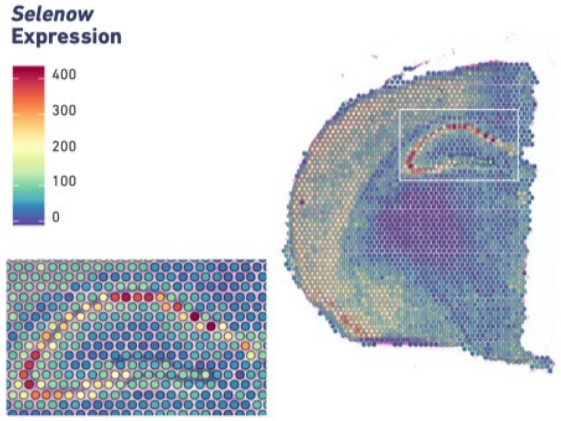
Main platforms – Spatial

Capture Sites - 10x Genomics Visium & SlideSeq





55um diameter capture sites



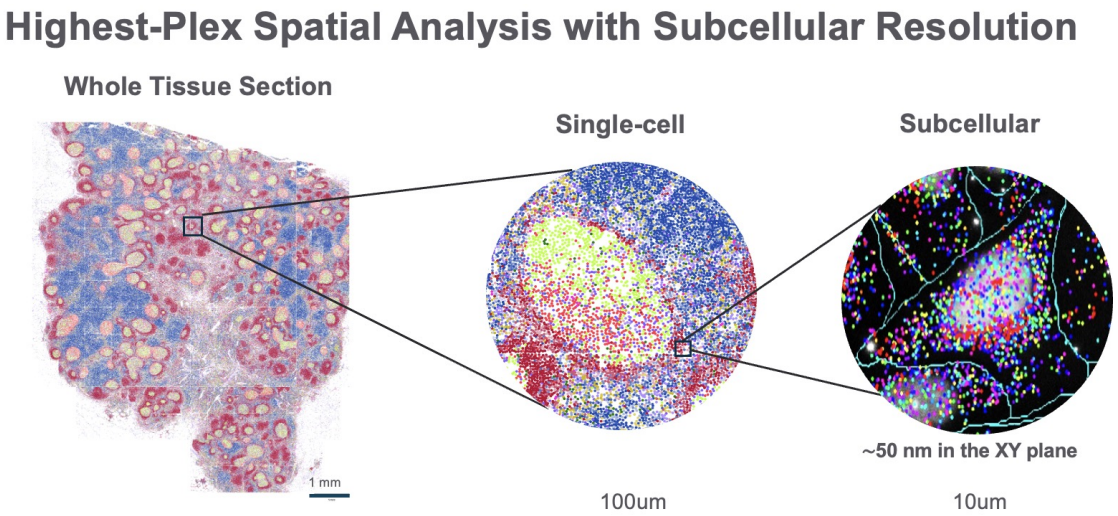
Pros

Whole transcriptome (polyA capture)
Straight forward library prep
Compatible with H&E-stained slides
FFPE or FF

Cons

Limited resolution (not single cell—yet)
Limited throughput

In Situ Hybridization - Nanostring CosMx & 10x Genomics Xenium



Pros

Subcellular resolution
Can be combined with antibody panels
Cost per slide, not tissue (TMA compatible)

Cons

Hardware cost $$$
Not yet whole transcriptome (only hundreds to 1k genes)

Experimental design considerations

Approximate cost breakdown for 10x Genomics experiment

Consumables	**Costs ($CAD)**
10x Reagents (Beads, enzyme, etc)	$3000/sample (max throughput = ~10,000 cells/sample)
Microfluidics chip	$320 for an 8-sample chip (consumable; no re-use)

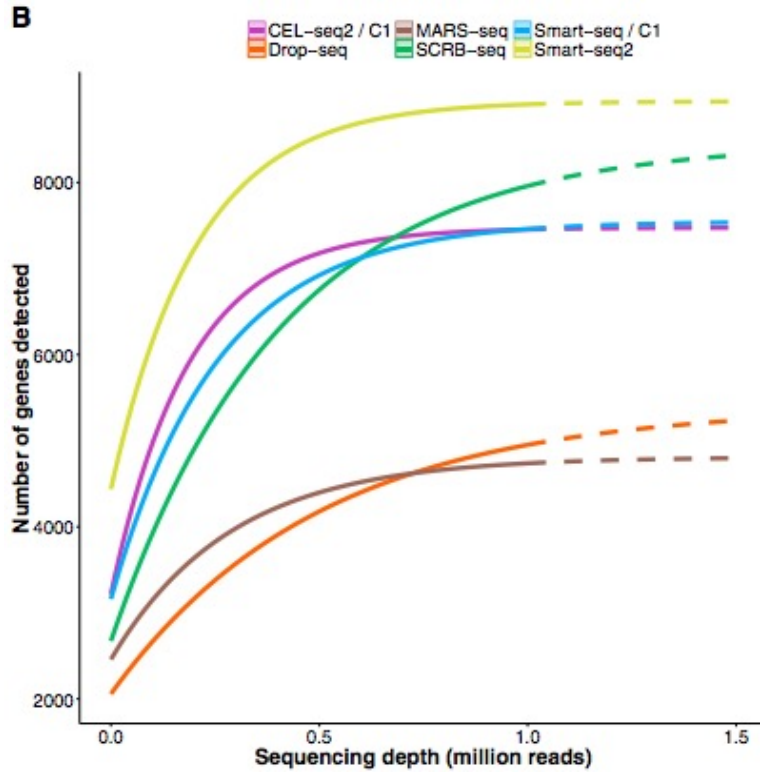
Sequencing options

NovaSeq S2/S4 PE100 Lane (~2.4B reads = ~96k cells)	$6000
NextSeq500 75-cycle (~400M reads) = ~16k cells	$3200

Multiplexing methods (eg. TotalSeq Cell Hashing, MULTI-seq, CellPlex) allow you to divide cell yields among pooled samples
(eg. 1x reagents = $3000 per 10k cells / 3 samples = 3000 cells per sample)

How deep to sequence?

of genes saturates around 1 million reads



Ziegenhain *et al., Molecular Cell,* 2017

But we don't necessarily need to detect everything in every cell!

General Rule: "…when the number of individual genes required to answer a given biological question is small, then greater transcriptome coverage is more important than analyzing large number of cells." Torre *et al., Cell Systems,* 2018

10x Genomics Recommendation

20-25k reads/cell

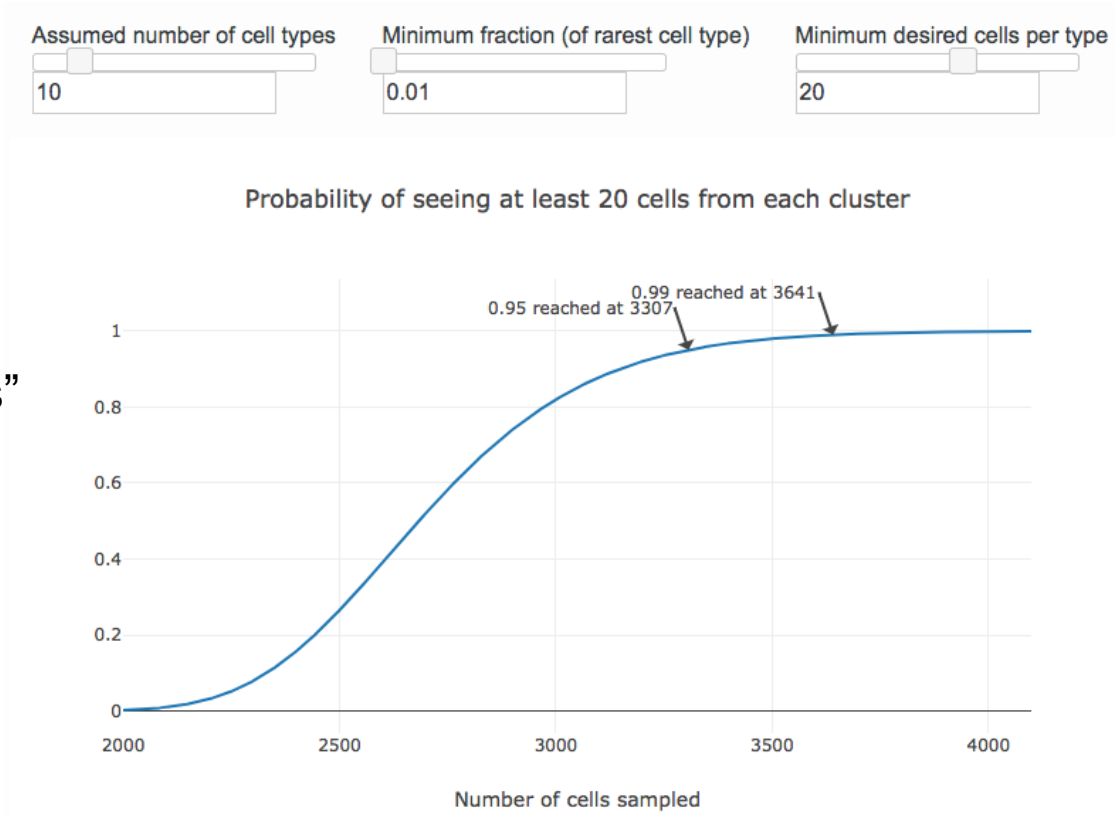
How many cells?

Two main things to consider

1) How many cell types are there?
2) What is the proportion of the rarest cell type you're interested in?

10x Genomics currently allows for each run to yield anywhere from ~500-10,000 cells

Satija Lab "How Many Cells" power calculator
https://satijalab.org/howmanycells



Caution: Do you actually know how many cell types are there? What about cell states?
The current trend in the field seems more focused in increasing cell #

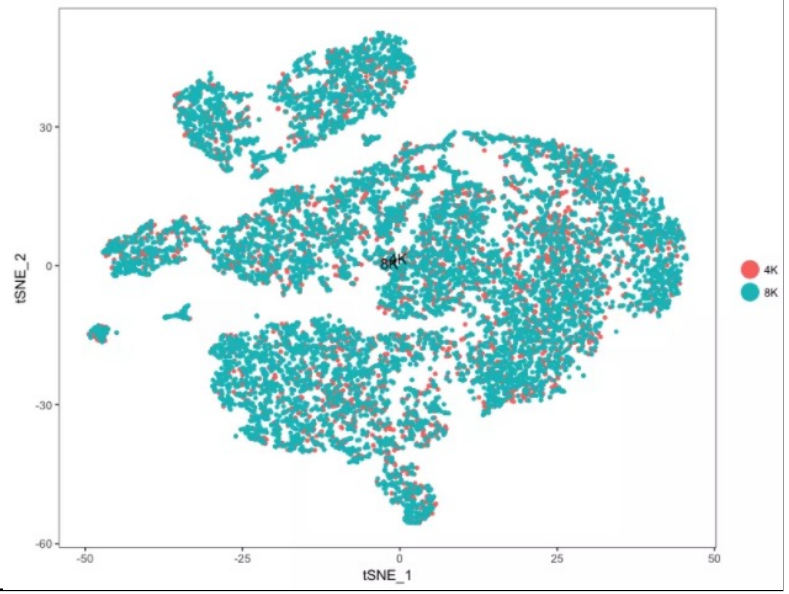
What about replicates?

It kind of depends on the purpose of the experiment

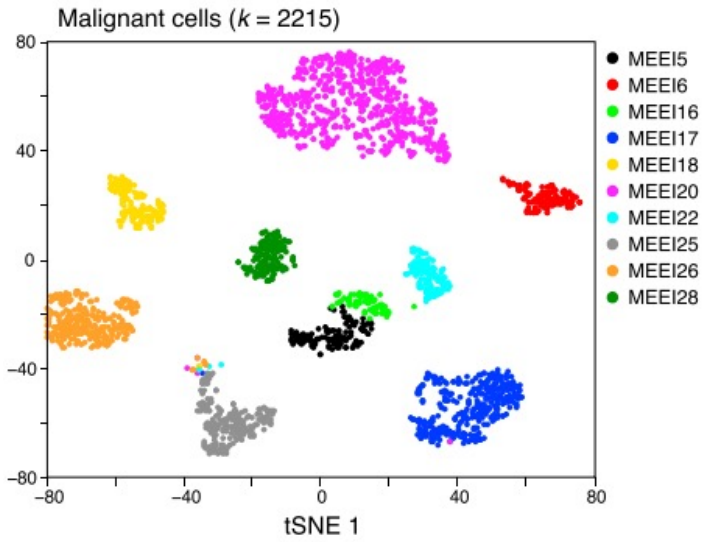
Replicates are always good, but currently, each replicate is quite expensive. They are becoming much more common than they were a couple years ago, but many still do no/few replicates and validate with other methods

- Maximize replicates if you can
- Pooling unlabelled replicates (eg. multiple mice) doesn't really solve the problem – multiplex instead!
- Minimize sample prep batches as much as possible. Never confound experimental variable with batch (controls collected day X, treated collected day Y)

Technical replicate of PBMCs has near-perfect overlap



Cancer cells differ dramatically between patients



Analysis

Choosing your ecosystem

scRNA-seq analysis involves management of many independent parts (eg. raw/normalized expression matrices, cell metadata, cell embeddings, etc). Various “ecosystems’ have been developed to hold on to all these pieces for you and provide convenient functionality to operate on them

Seurat

Language: R
Tutorial: Link
Pros:

- Natively provides most functionality you would need
- Very user-friendly for beginners
- Most popular ecosystem. Large community of users

Cons:

- Development led by a single lab
- Most functionality siloed within single package
- Sometimes challenging to interact with tools outside the ecosystem

SingleCellExperiment

Language: R
Tutorial: Link
Pros:

- Easy interaction with the many R packages through Bioconductor
- Ecosystem developed by some very reputably statisticians in the genomics field

Cons:

- Common code less intuitive than Seurat
- Dependency on diverse packages introduces complexity in standard workfows

scanpy

Language: Python
Tutorial: Link
Pros:

- Most functionality needed built directly into scanpy
- Data can interact directly with some of the most sophisticated tools
- Python is faster than R

Cons:

- Python much less friendly to users without coding background
- Documentation of python packages often less extensive than R packages

Standard Preprocessing

Standard Preprocessing - Quality Control

Goal

Evaluate the general quality of our data and remove any poor quality (technical or biological) cells that could negatively affect downstream analysis

Three common metrics

1. **Number of measured transcripts per cell**
   Dependent on sequencing depth (reads per cell), transcriptome complexity, and technical variables like lysis/RT efficiency.

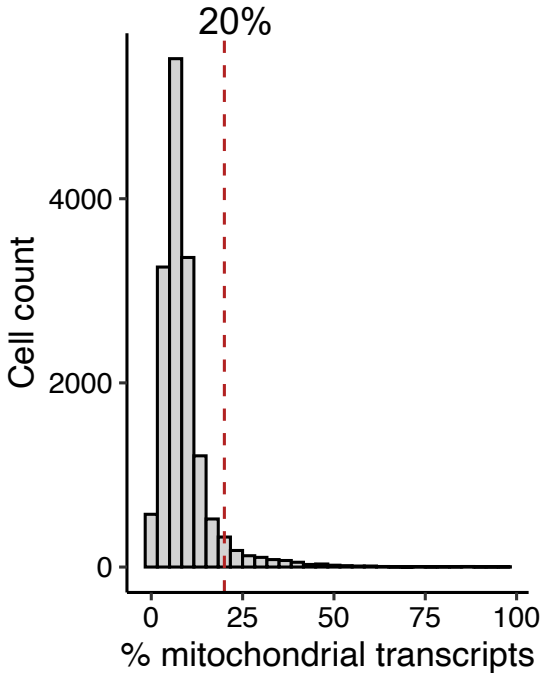
2. **Number of unique genes detected per cell**
   Typically correlated well with (1).

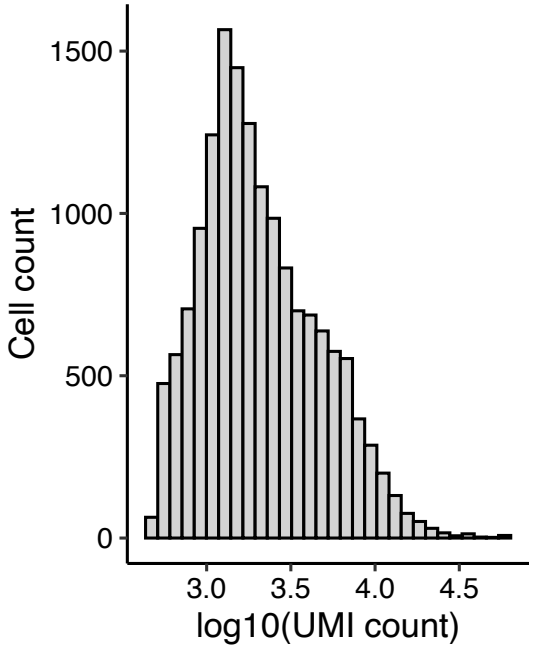
3. **Percentage of transcripts per cell associated with mitochondrial genes**
   High percentages associated with dying cells as membrane integrity and cytoplasmic RNAs are lost, enriching for mitochondrial transcripts

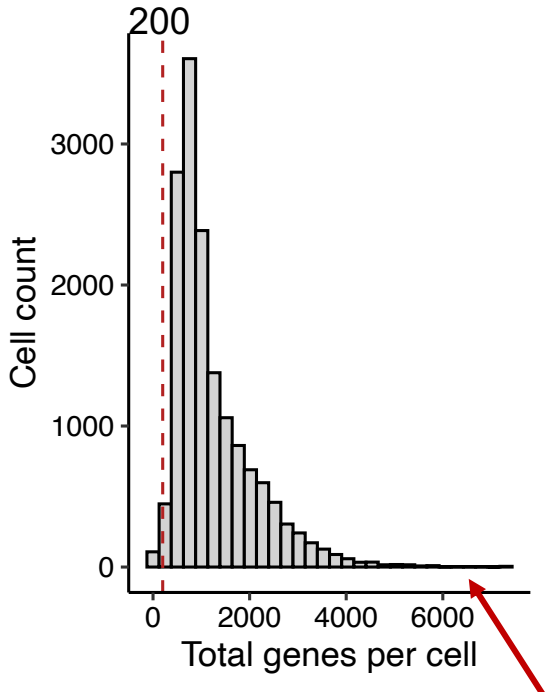
Standard Preprocessing - Quality Control

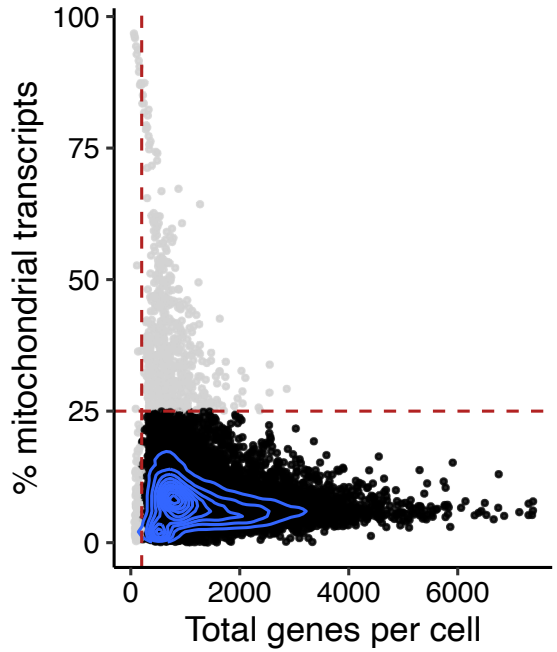
Most tutorials use PBMC data, which is often extremely clean. Many real samples require filtering to remove dead/poor quality cells. Visualize your data!

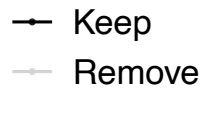
When in doubt, default to **not** removing something. You can always see how a certain population affects downstream analysis and then decide if you need to revisit filtering it out.











Upper-end thresholds used to be common, assuming they were removing doublets. Can lead to biased removal of specific high-complexity cell types

Standard Preprocessing - Quality Control

Doublet removal

With many scRNA-seq experiments pushing to the upper end of the advertised 10k cell yield, a non-trivial percentage of data points will be from >1 cell in a single droplet

Multiplet Rate (%)	# of Cells Recovered Single Cell 3' v3.1 (Standard)
~0.8%	~1000
~1.5%	~2000
~2.3%	~3,000
~3.0%	~4,000
~3.8%	~5,000
~4.6%	~6,000
~5.3%	~7,000
~6.1%	~8,000
~6.8%	~9,000
~8.0%	~10,000

May not seem like a lot, but in a 10k cell dataset, some relevant clusters will be more rare than this

Solution #1 – Experimental

Multiplex multiple samples that each contain a sample barcode that can be sequenced (eg. CellPlex, MULTI-seq, Cell Hashing). Datapoints positive for >1 barcode would reflect a doublet. These can be identified and remove from downstream analysis The number of doublets that can be identified is dependent to the number of samples pooled

Solution #2 – Computational

Various methods exist that simulate doublets by aggregating expression profiles of distinct clusters and learning what mixed expression profiles.

Examples

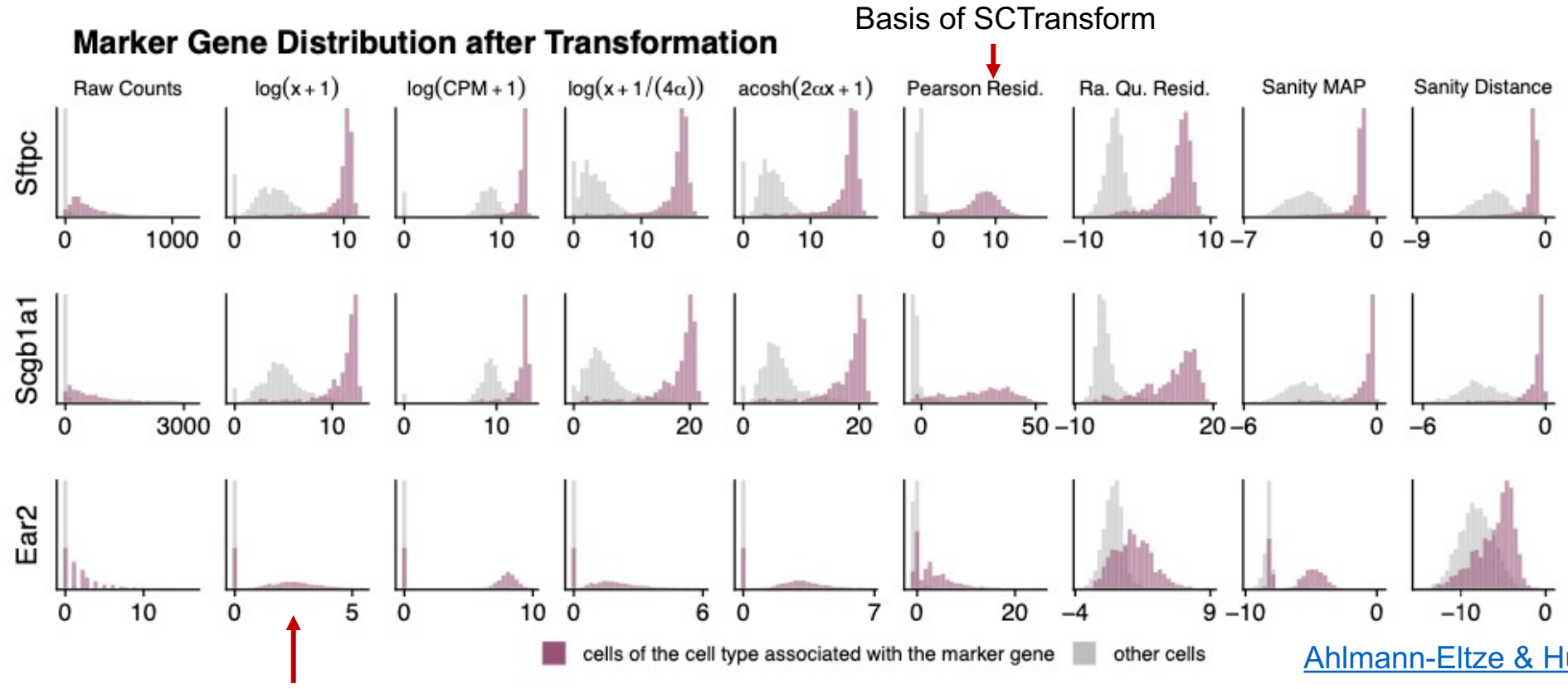
scDblFinder
DoubletFinder
Scrublet

Benchmark paper (2021): PMID: 33338399

Standard Processing - Normalization

Goal

- Correct for depth differences between cells (ie. differences in the # of measured transcripts). Often involves scaling counts to some constant denominator (often 10,000 for scRNA-seq)
- Account for variance differences (heteroscedasticity) between genes



Ahlmann-Eltze & Huber, *bioRxiv*, 2022

If relevant annotations were known for cells at this point, normalized data could be used directly for differential expression

Many single-cell workflows, however, require that we generate low dimensional representations of our data to better evaluate cell-cell similarities in the data

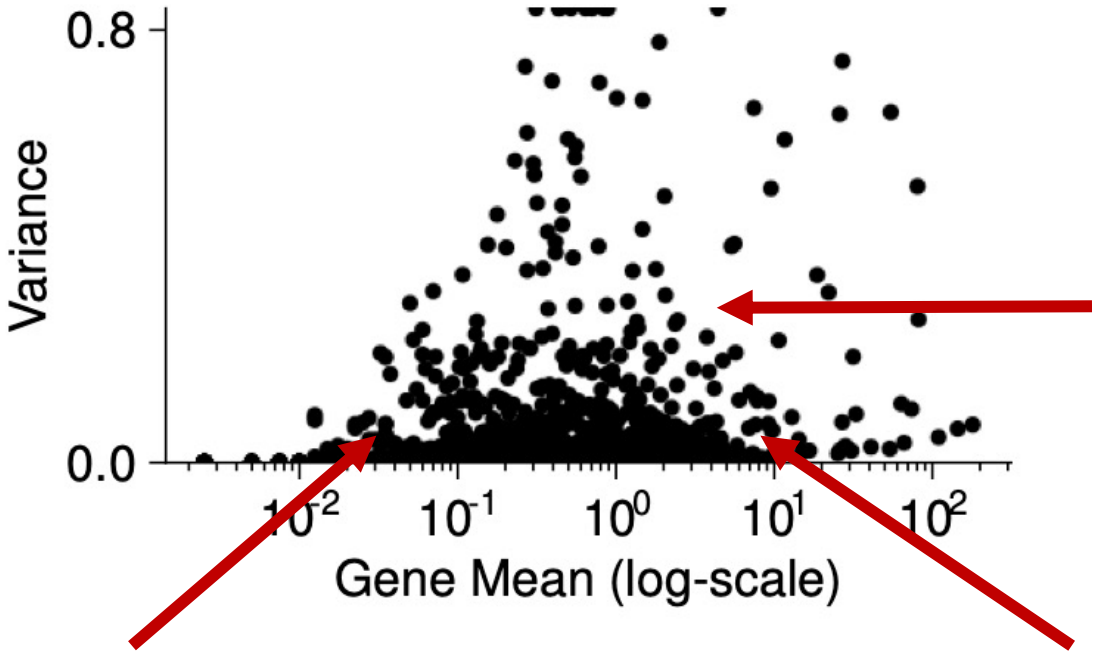
Standard Preprocessing - Feature Selection

Goal

Identify genes that tell us the most about the data structure, eliminate genes that provide little information and/or introduce noise into the downstream analysis

Approach

Most often involves selecting the top n (often 2000-3000) genes with the highest variance across cells



Informative: highly variable genes (eg. cell type markers)

Uninformative: lowly expressed and/or rare detection

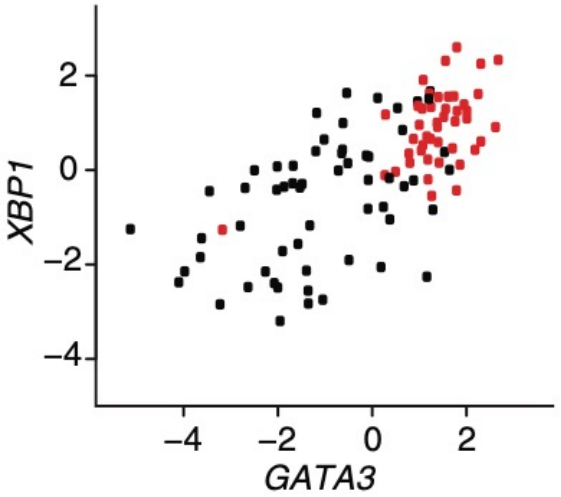
Uninformative: highly expressed, but ubiquitous (eg. “housekeeping”)

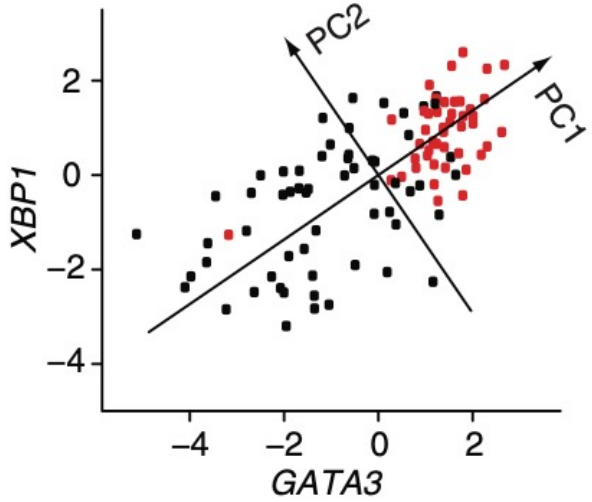
Ahlmann-Eltze & Huber, *bioRxiv*, 2022

Standard Preprocessing - Principal Component Analysis (PCA)

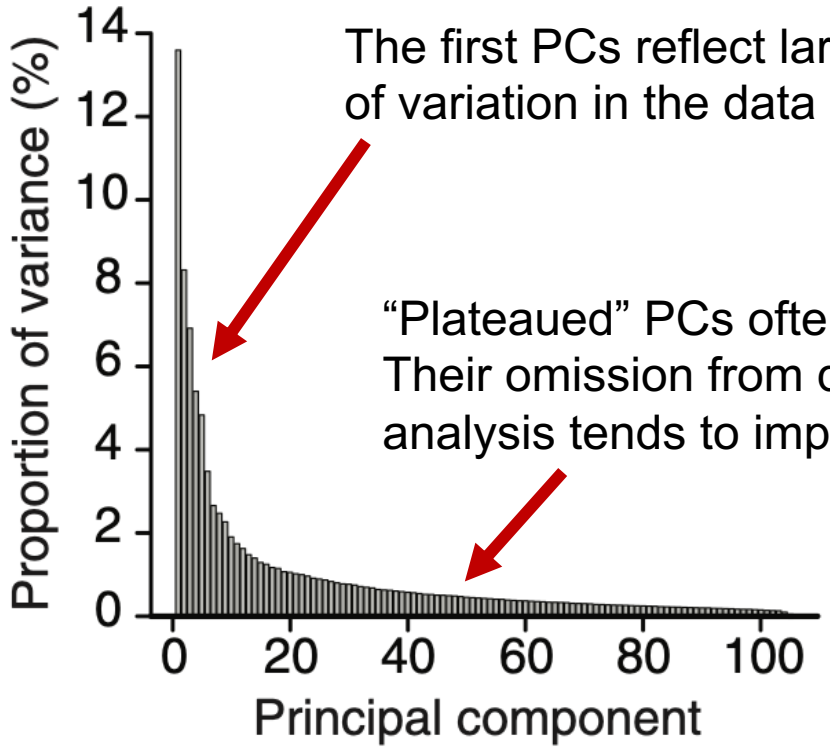
Goal

Represent a large percentage of the variation present in the 2000-3000 dimensions (ie. gene) of data with a much smaller number (~20-30) of components by summarizing co-expression patterns





Ringnér, *Nat Biotech*, 2008



The first PCs reflect largest sources of variation in the data

"Plateaued" PCs often reflect noise. Their omission from downstream analysis tends to improve signal

The first 20-30 principal components (PCs) are a low dimensional representation of the data the serves as an input for many downstream analyses, including clustering, UMAP, trajectory inference

Standard Preprocessing - Non-linear dimensionality reduction

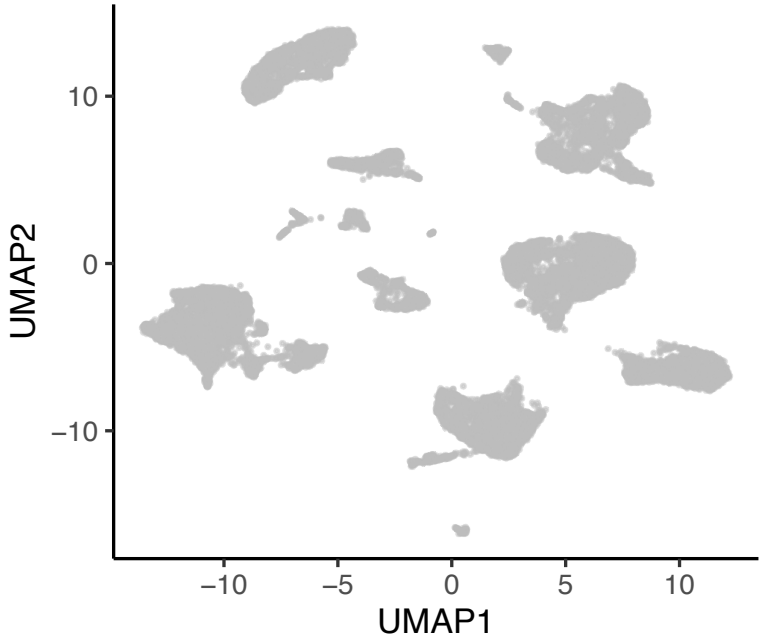
Goal

Typically—to represent high dimensional data in two dimensions to allow for effective visualization

Approach

Often involves applying algorithms that evaluate cell-cell similarity in higher dimension space (often PCA space) and then try to position data points in 2D in a way that best matches(big oversimplification but you get my point)

17,157 cells from mouse intestine



How to (mis)read UMAP – LINK

- Embedding highly dependent on hyperparameters
- Cluster sizes in a UMAP plot mean nothing
- Distances between clusters might not mean anything
- Random noise doesn't always look random

UMAP IS NOT A CLUSTERING ALGORITHM

Alternatives embedding methods

Graph embeddings (eg. SPRING), tSNE, Diffusion map, PHATE

Standard Preprocessing - Clustering

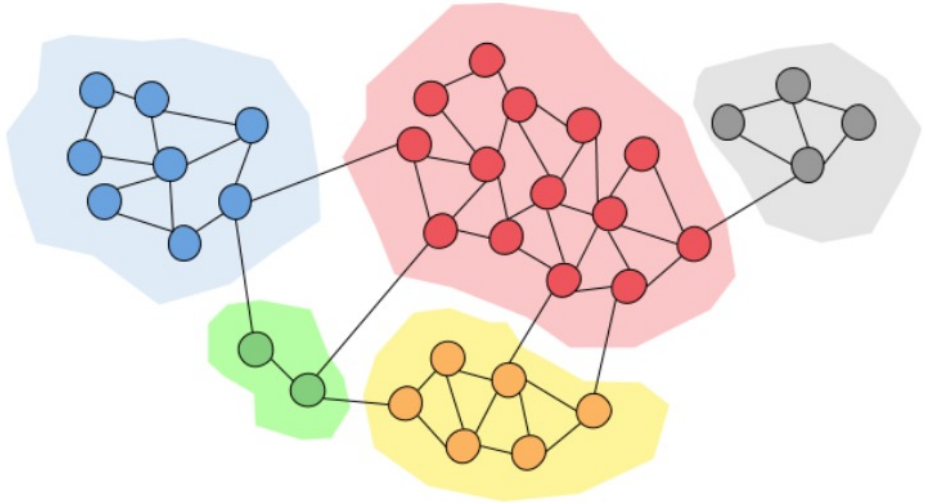
Goal

Define discrete groups of similar cells, often aiming to identify distinct cell types in the data

Approach

Most single-cell clustering workflows involve graph-based clustering:

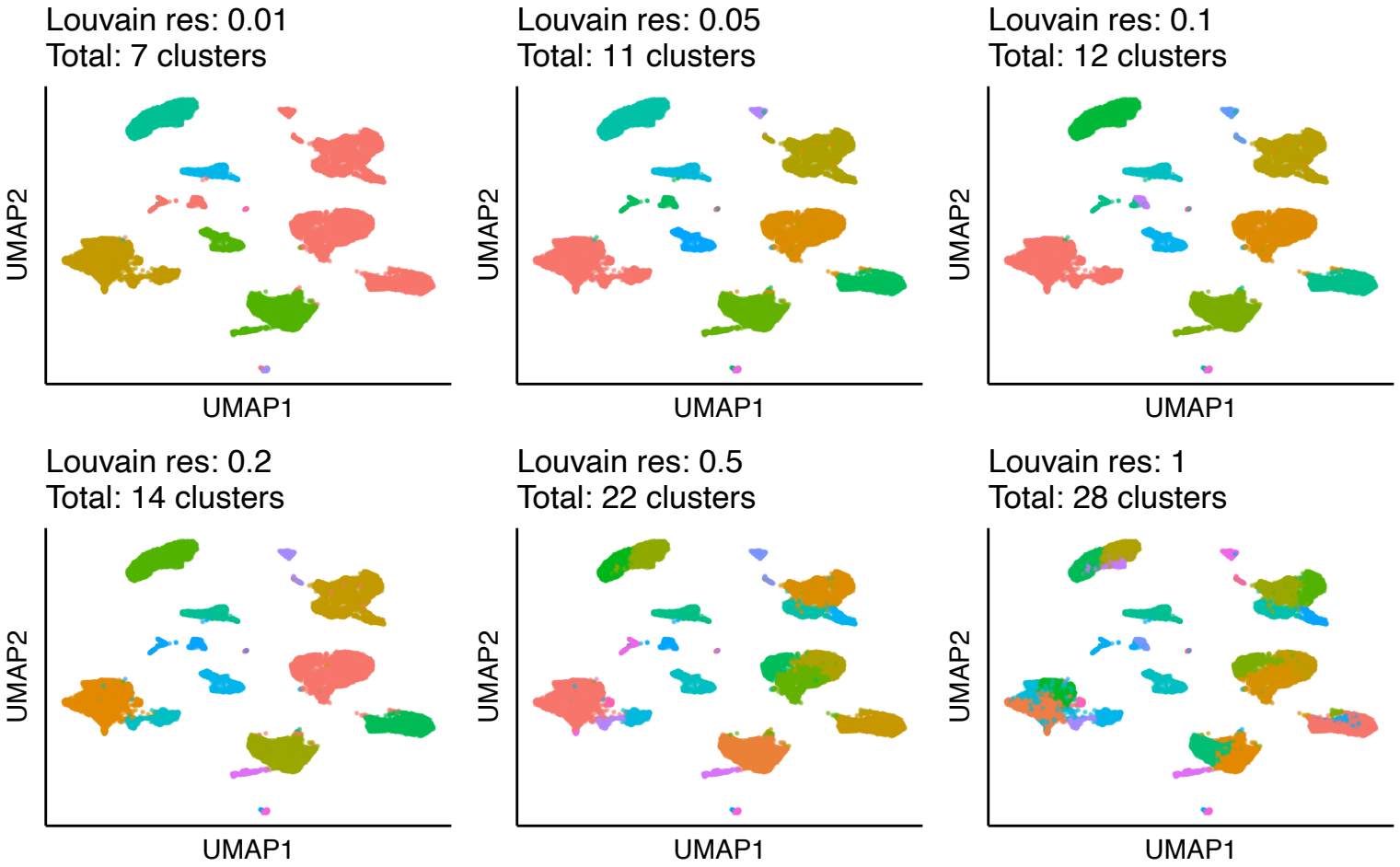
1. Build a nearest-neighbor graph of your data from PCA space. Each node is a cell, and each node is connected to its *k* most similar cells
2. Apply a “community detection” algorithm (eg. Louvain or Leiden) on the graph to define cluster boundaries, optimizing for high connectivity within boundaries, minimal connectivity between communities



Standard Preprocessing - Clustering

Resolution

A user-defined parameter specifying the granularity of the clustering



Methods exist to evaluate clustering (eg. silhouette width). **Quantitatively "best" resolution is not necessarily relevant biologically**

Advice

Choose a resolution that matches the "resolution" of your study

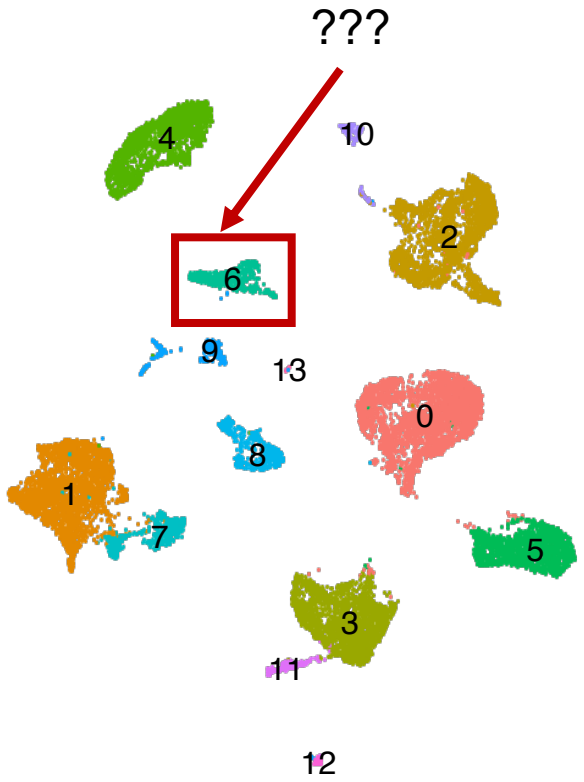
Approach clustering with biological priors. Eg. Does it seem reasonable that there are 10 different fibroblast populations in your tissue?

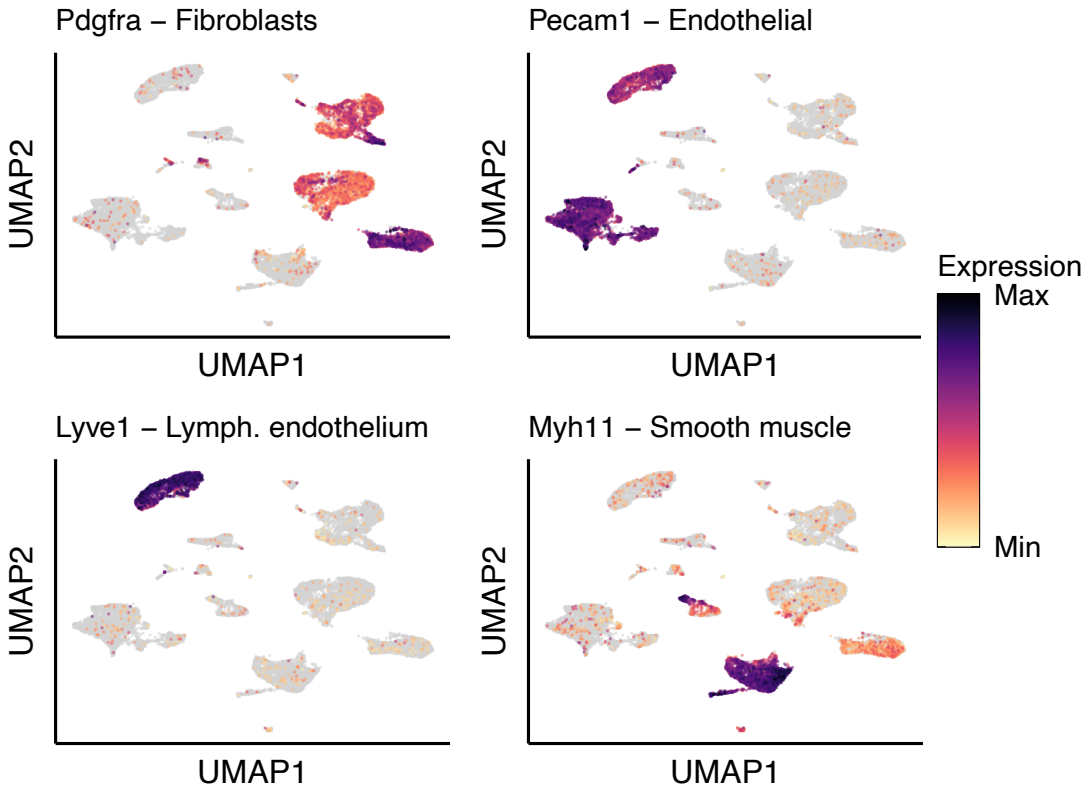
If dealing with a complex sample, consider a first-pass of low-resolution clustering, then subset major populations and re-cluster (avoid over-clustering

Cell Type Annotation - Manual

Goal

Map literature-defined cell type markers to clusters





Differential expression to identify cluster 6 markers: eg.

Prnp – Expressed in CNS

Plp1 – Myelin protein

Apoe – KO affects enteric glial cells

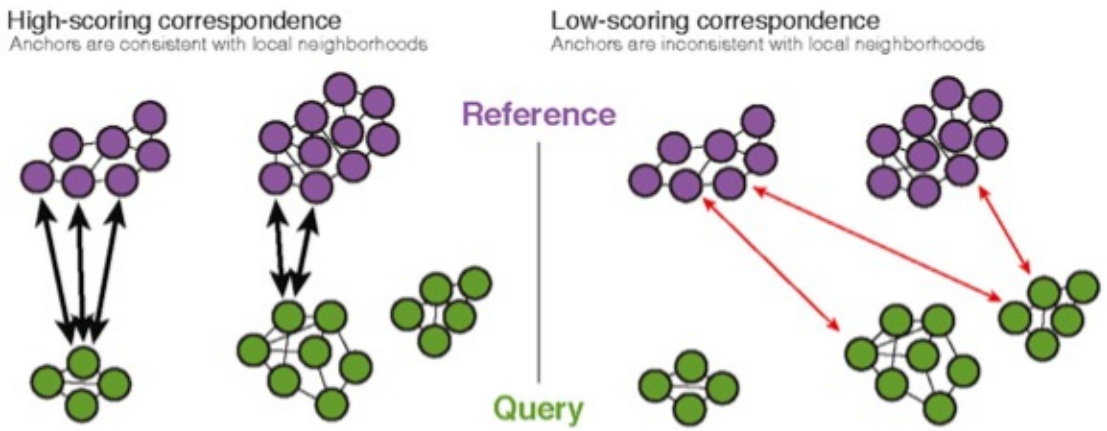
Cluster 6 = enteric glial

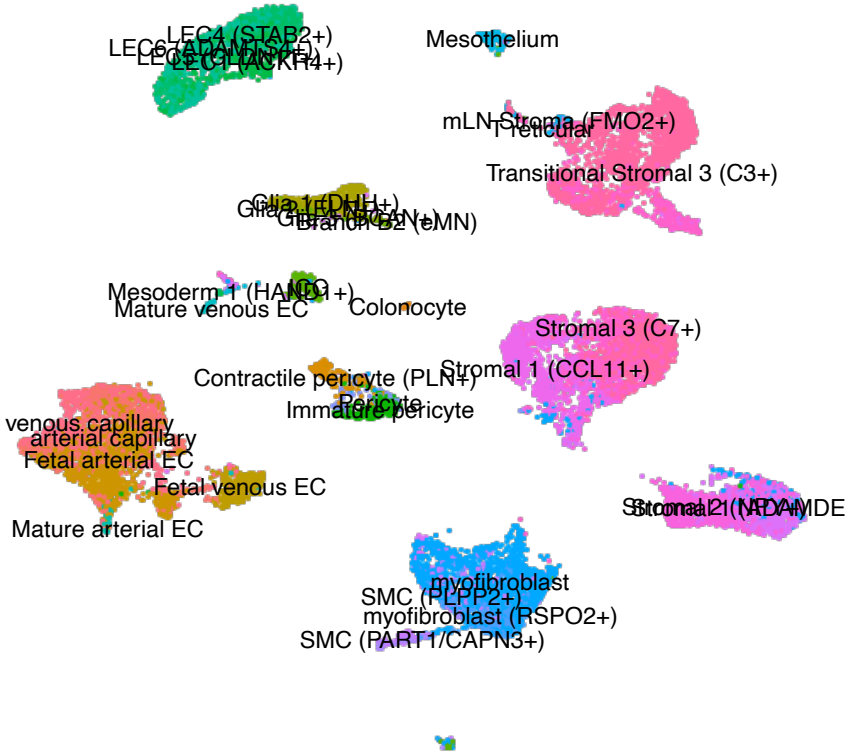
Cell Type Annotation - Automated

Goal

Use a well-annotated/vetted reference dataset to predict the identity of each cell in your data

Approach 1 – Using a reference scRNA-seq dataset
(eg. Seurat's label transfer method)





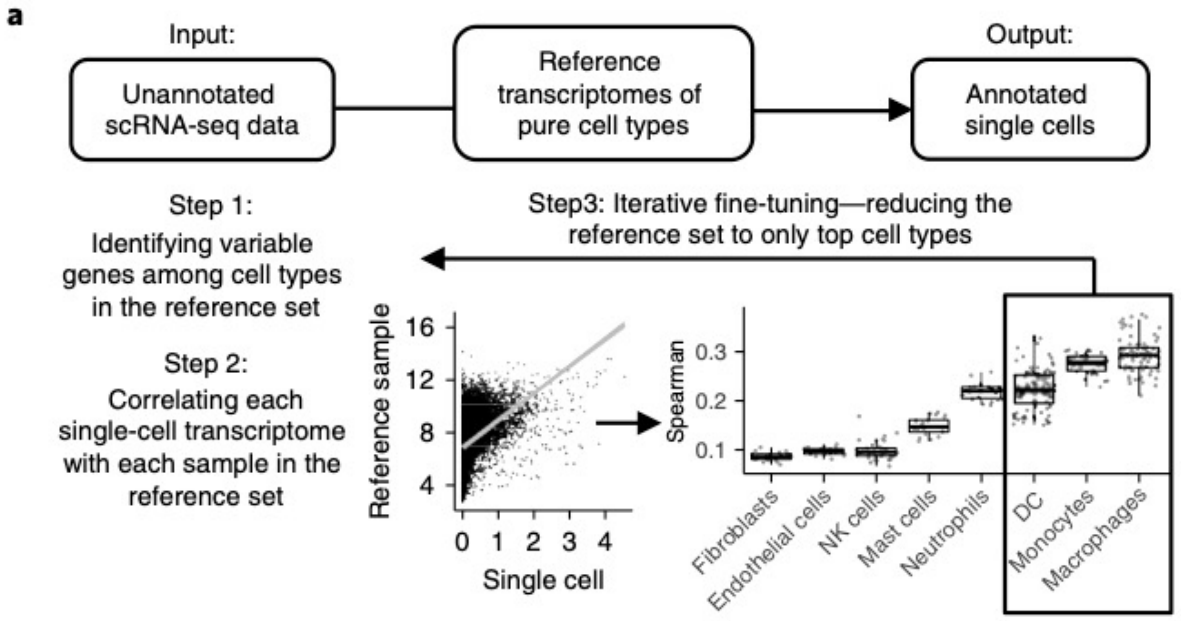
Automated annotation using gut cell atlas from the Human Cell Atlas

Cell Type Annotation - Automated

Goal

Use a well-annotated/vetted reference dataset to predict the identity of each cell in your data

Approach 2 – Using references of "pure" expression patterns
(eg. Methods: SingleR, CellAssign; Signature Database: celldex)



SingleR: Aran et al., Nat Imm, 2019

Challenges

Most automated methods annotate at the level of individual cells. Depending on the reference/query datasets, this can lead to occasional low confidence labels. Eg. cell type mismatch in query/reference samples can lead to inappropriate labels.

Advice

- Interpret annotations with biological priors
- Sometimes annotations can be cleaned up by applying a "majority rules" label to all cells belonging to a cluster
- Many reference atlases contain high-resolution cell type labels (eg. 5 different fibroblast subtypes). Just as with clustering, don't hesitate to aggregate common labels into a lower resolution grouping (eg. "Fibroblasts") if that is more appropriate for your study

To integrate or to not integrate

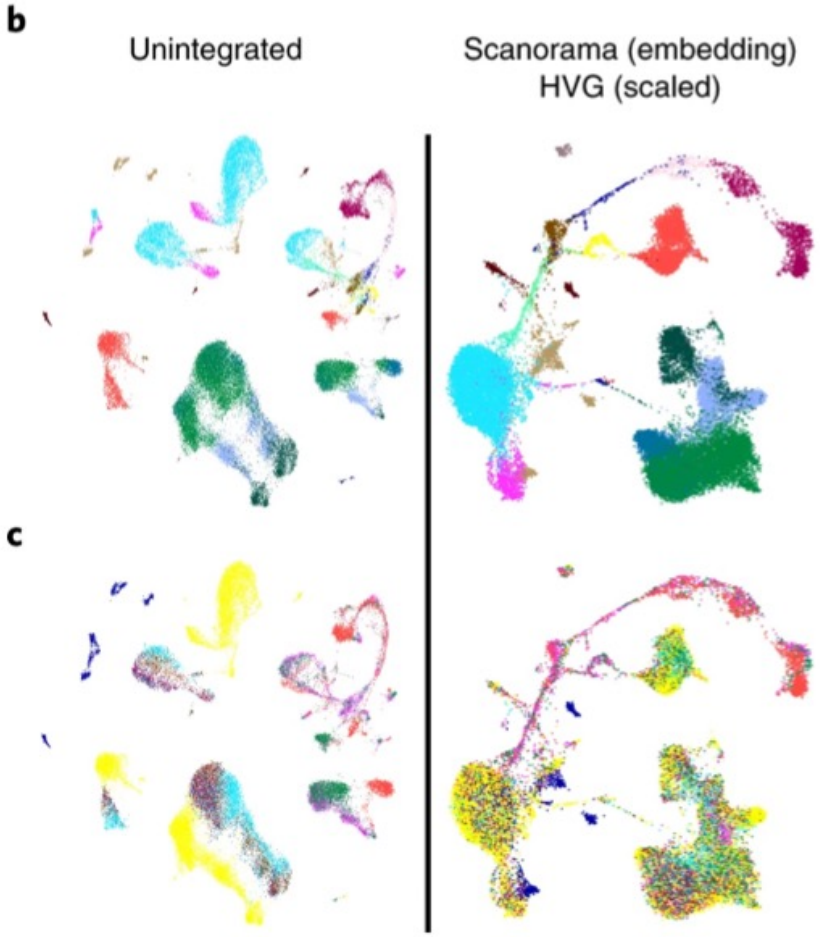
Integration

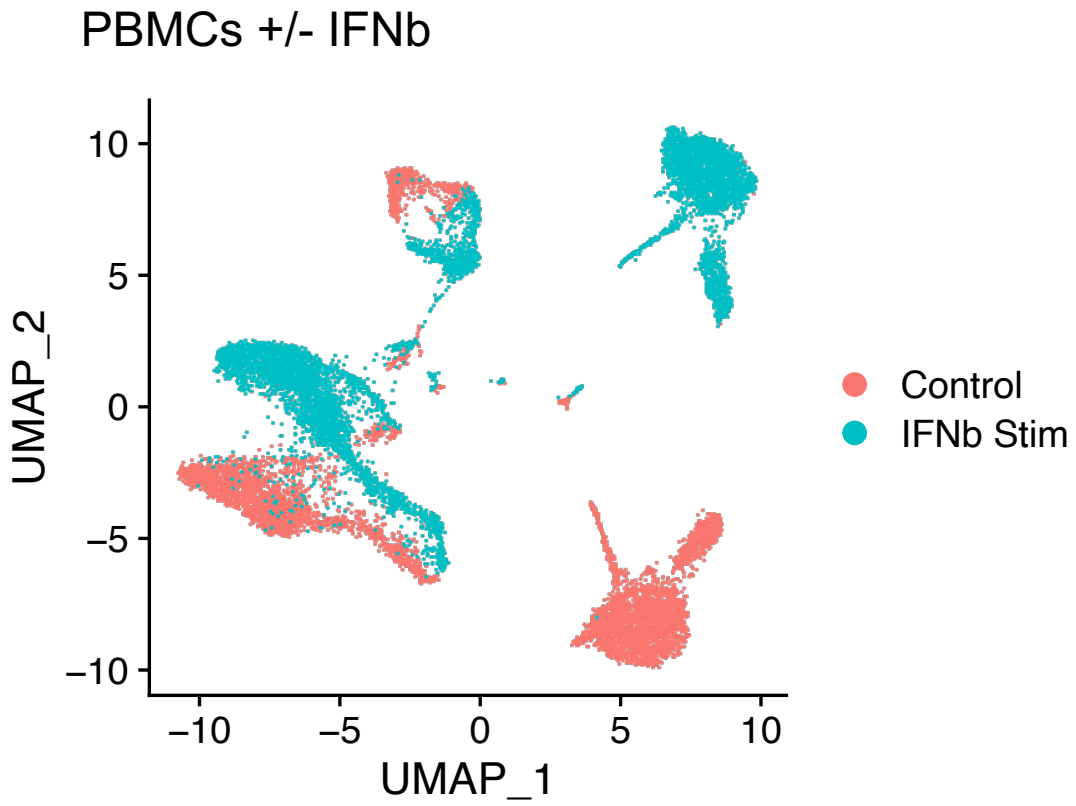
Both experimental (eg. treatment, genotype, etc) and technical (eg. batch, technology) variables can introduce gene expression or measurement differences that lead to divergence of populations in low dimensional embeddings

Colored by:

Cell type

Batch



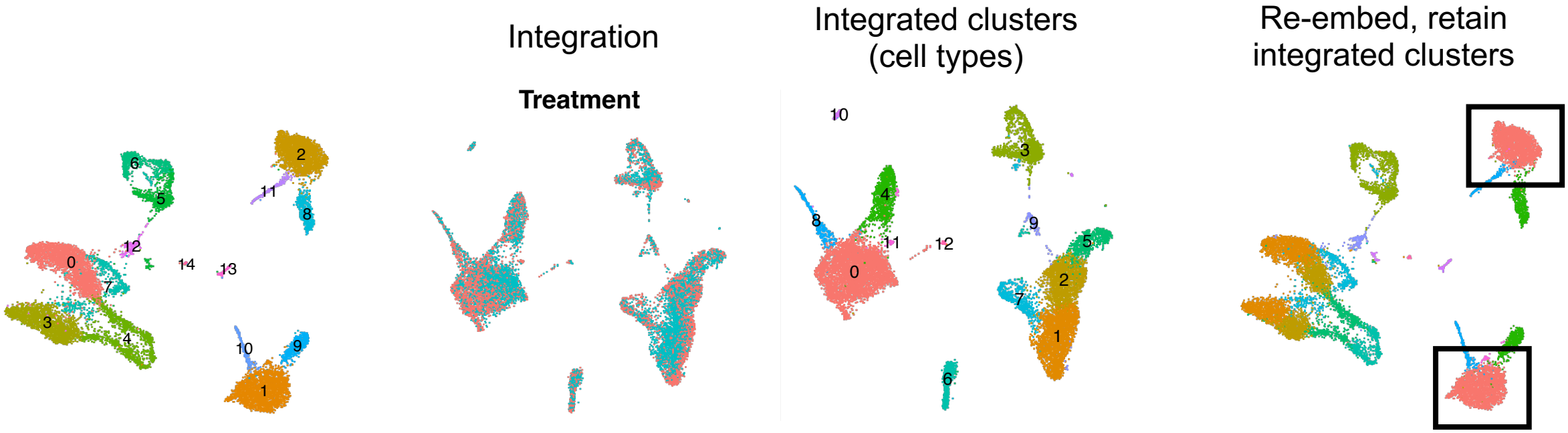


[Luecken et al., Nat Methods, 2021]

Integration

Integration allows for the correction/removal of some variable **for the purpose of aligning samples for low dimensional embedding and clustering**. Eg. clusters represent cell types regardless of experimental condition/batch

Integration is not performed for the purpose of correcting expression values for differential expression or exploration!!! If needed, experimental batch should be included as a covariate in DGE regression models

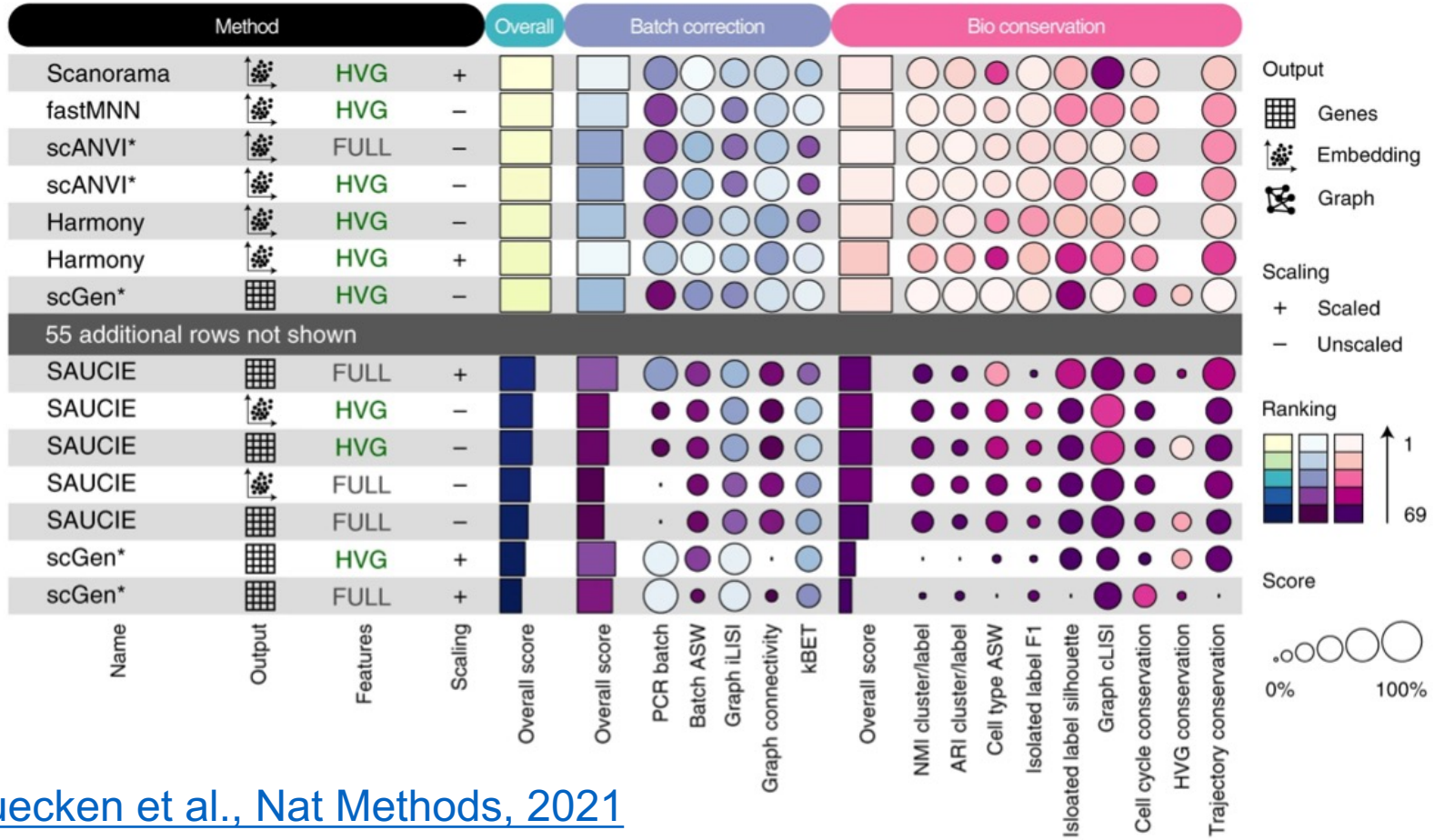


Differential expression between samples within same cluster using uncorrected expression values

Integration

Integration methods must balance bioconservation and batch removal. We don't want to overcorrect and remove differences associated with cell type composition (eg. forcing a novel population into another cell type) or relevant phenotypic variability (eg. cell cycle state, differentiation status)

Benchmark of 68 method+preprocessing combinations:



Luecken et al., Nat Methods, 2021

Downstream Analysis

Differential Expression

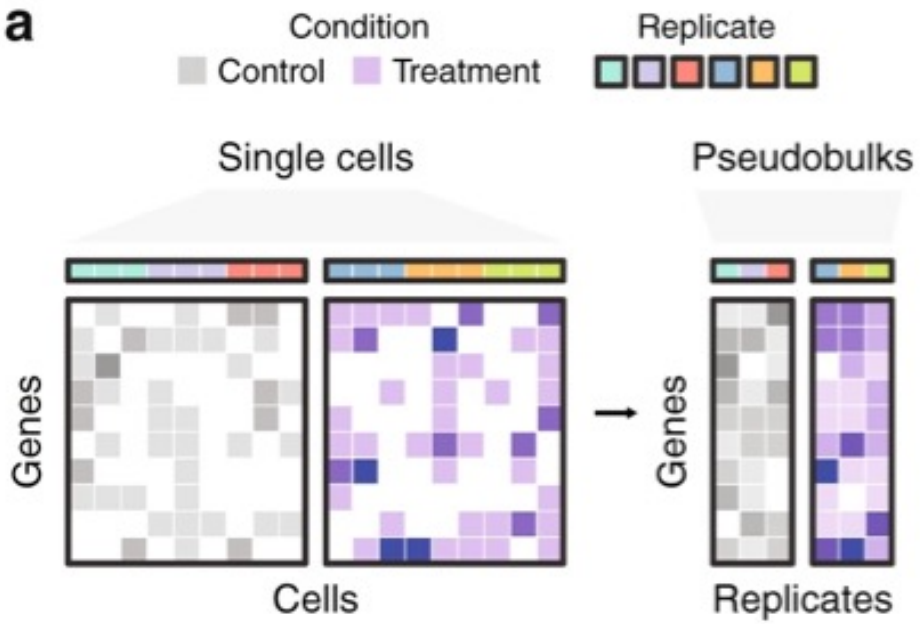
Differential Expression

Goal

Use statistical approaches to evaluate expression differences between defined groups of cells

Best Practice

If independent biological replicates have been used, “pseudobulk” expression profiles for each group (eg. cluster) can be generated for each sample and tools developed for bulk RNA seq (eg. DESeq, edgeR) can be used to test for differential expression



To dig into this and the comparison with methods that use single-cell measurements, see Squair et al., Nat Comm, 2021

If you are in the stages of planning experiments and are not comfortable with the cost of including replicates, consider the possibility of using multiplexing strategies to pool samples for processing. Eg. CellPlex, MULTI-seq, Cell Hashing, SNP-based multiplexing

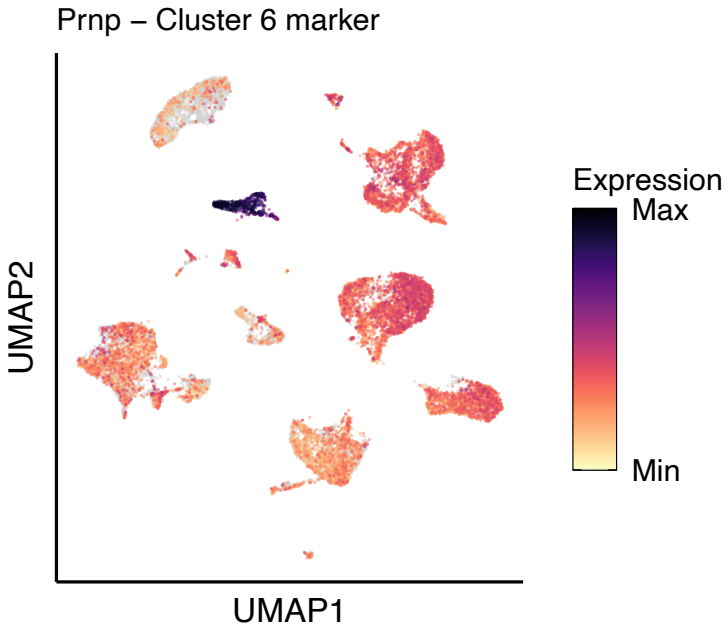
Differential Expression

Common Practice

Treat each cell as an independent replicate and perform a non-parametric test (eg. Wilcoxon rank sum test) between groups of cells

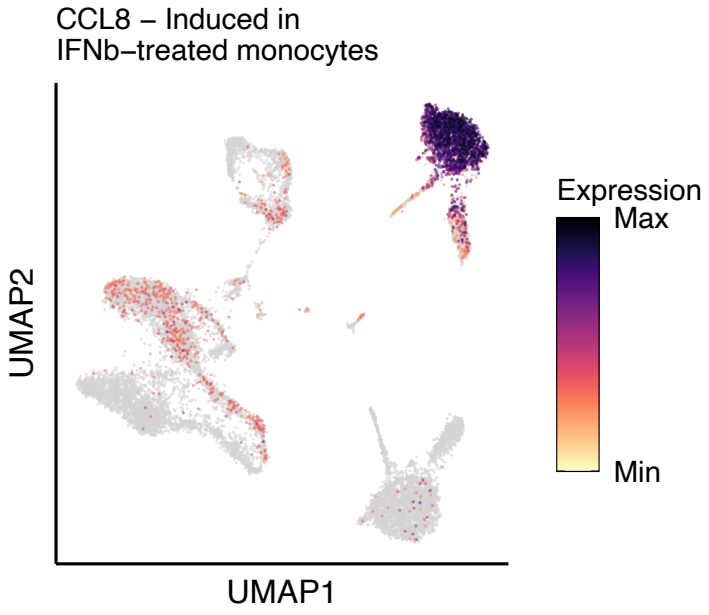
Marker Identification

Often test ”one-versus-rest”.
Eg. Cluster 6 vs. all other clusters



Differential expression between conditions

Test within cluster, between samples
(integrated clusters make this more straight forward)



Note: Current approaches aren’t very good at evaluating *specificity*. I expect to see this change over time

Downstream Analysis

Trajectory Inference

Trajectory Inference

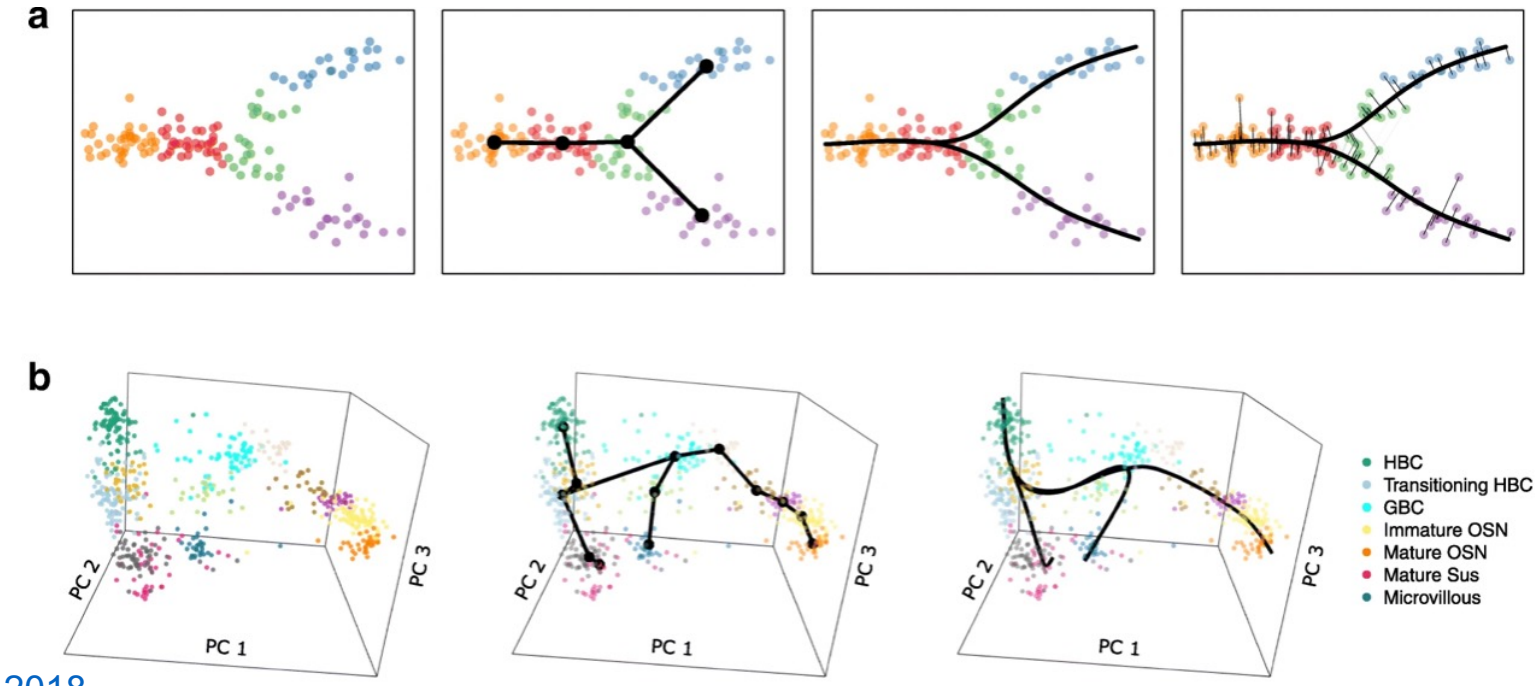
Goal

To infer a continuous phenotypic trajectory from single-cell data

Approach

Given asynchrony/heterogeneity associated with phenotypic dynamics, individual cells may be at different points along a continuum at the time of scRNA-seq processing.

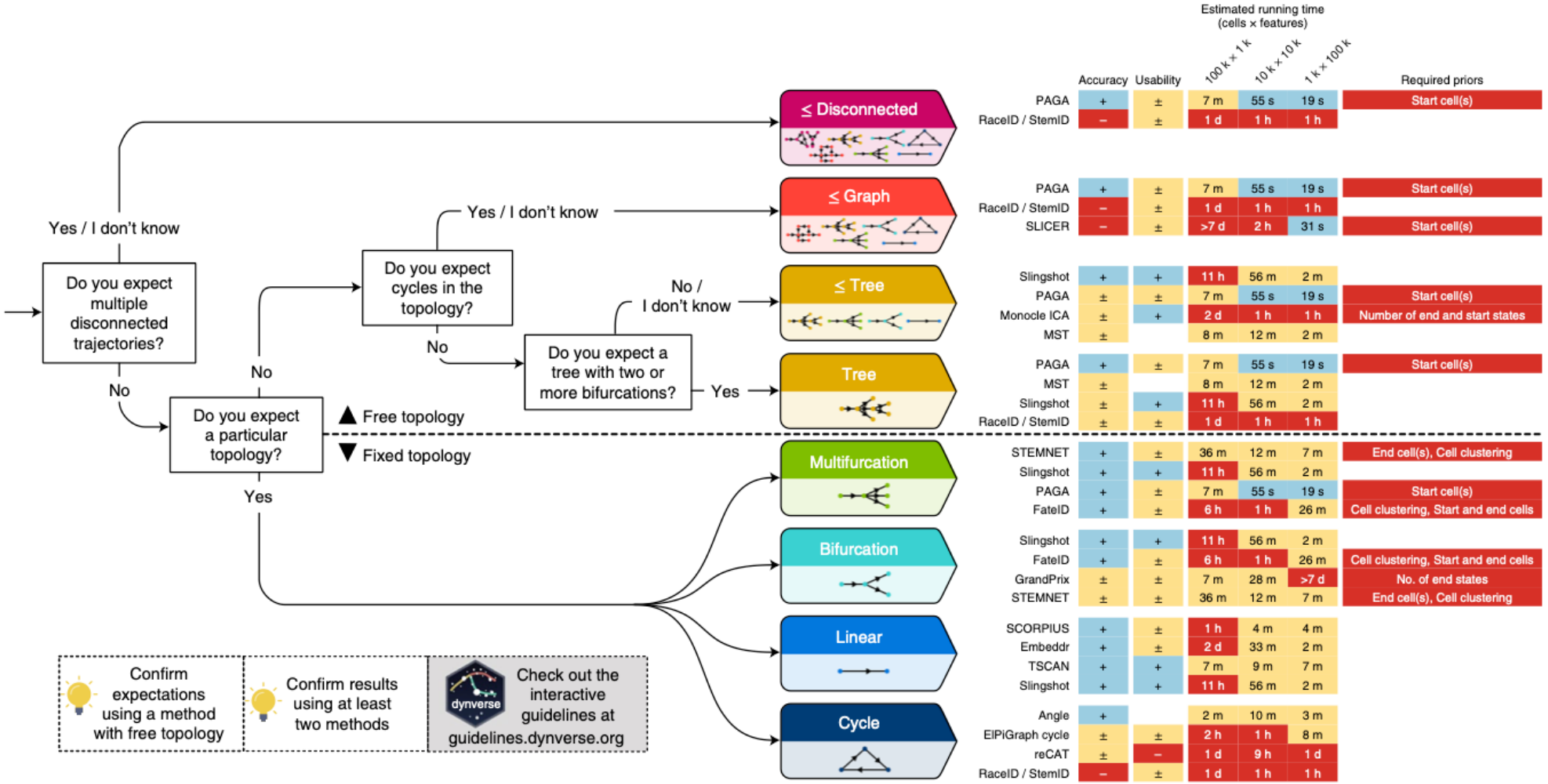
Trajectories are usually defined as an optimal path ”through” the data (eg. diffusion along a nearest-neighbor graph, a principal curve through some low dimensional embedding, etc)



Street et al., BMC Genomics, 2018

Trajectory Inference

Benchmark of trajectory inference methods
[Saelens et al., Nat Biotech, 2019]

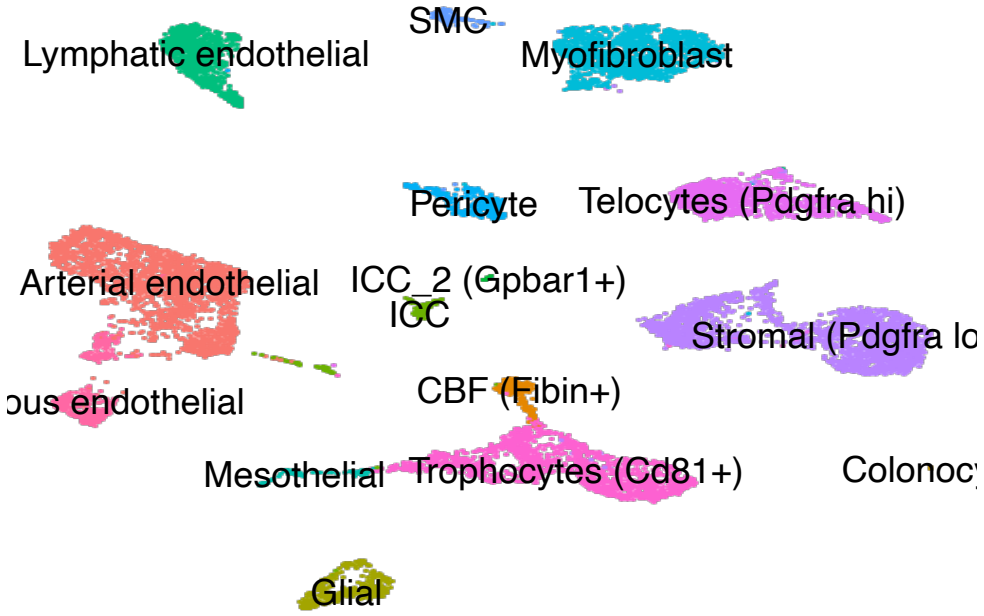


Trajectory Inference

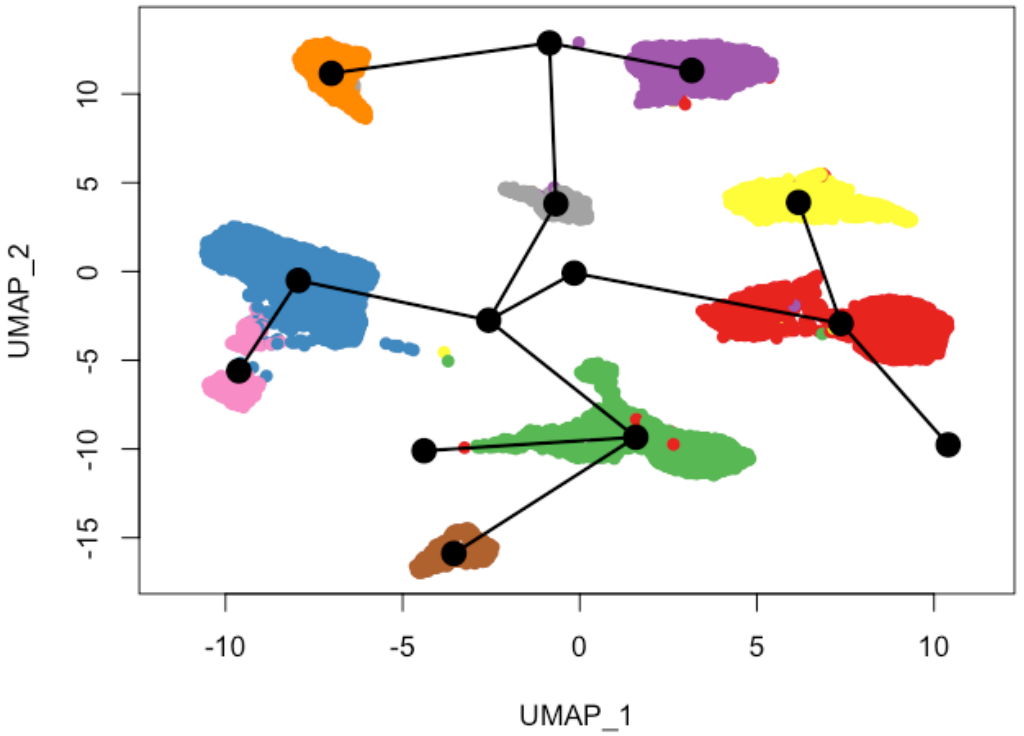
WARNING: Many of these tools will give the “optimal” trajectory given the input data, regardless of whether or not a trajectory makes sense in that data

These tools are not to discover **IF** a trajectory exists, they build trajectories **assuming** a trajectory exists

Whole intestinal mesenchyme



Trajectory inference w/ Slingshot



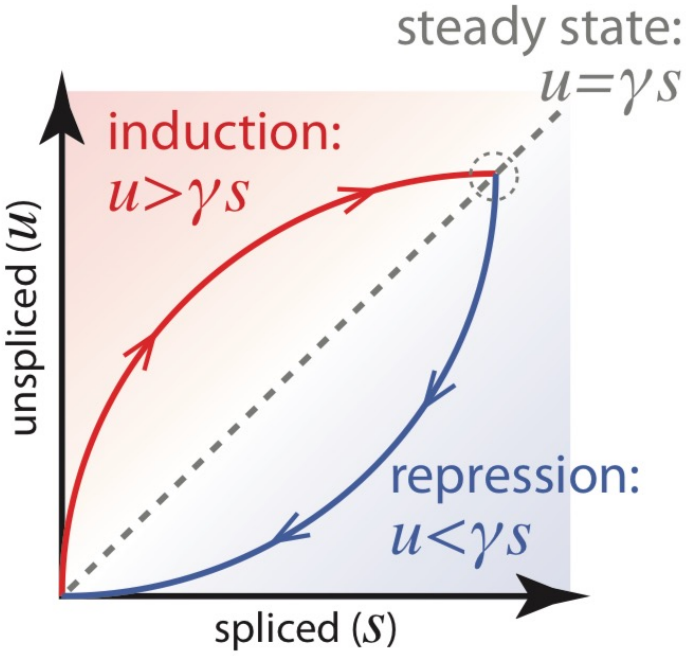
Do not blindly apply these tools and take the results to be some ground truth

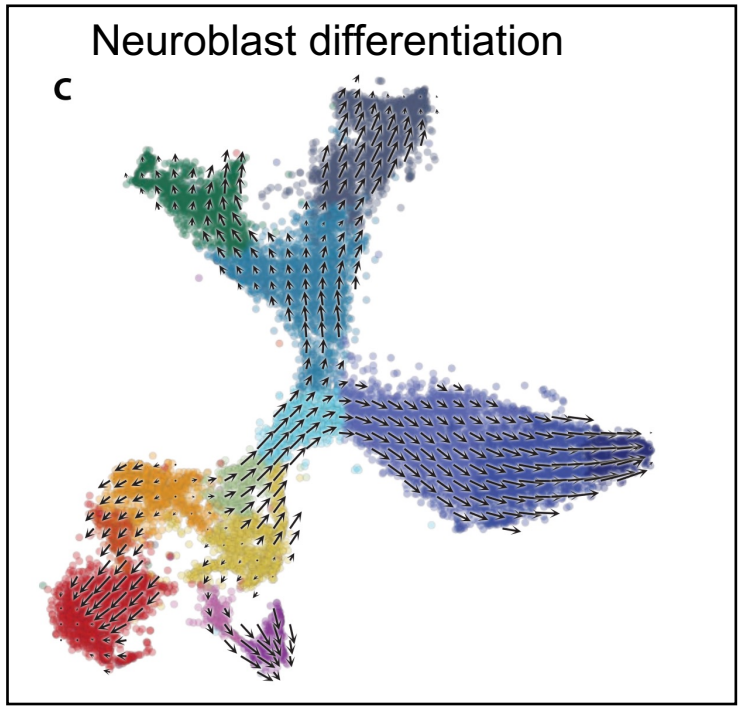
Trajectory inference with RNA velocity

RNA velocity of single cells

Gioele La Manno[1,2], Ruslan Soldatov[3], Amit Zeisel[1,2], Emelie Braun[1,2], Hannah Hochgerner[1,2], Viktor Petukhov[3,4], Katja Lidschreiber[5], Maria E. Kastriti[6], Peter Lönnerberg[1,2], Alessandro Furlan[1], Jean Fan[3], Lars E. Borm[1,2], Zehua Liu[3], David van Bruggen[1], Jimin Guo[3], Xiaoling He[7], Roger Barker[7], Erik Sundström[8], Gonçalo Castelo-Branco[1], Patrick Cramer[5,9], Igor Adameyko[6], Sten Linnarsson[1,2]* & Peter V. Kharchenko[3,10]*

Nature, 2018





IMPORTANT: Inferences are limited to a timeframe determined by splicing kinetics (on the scale of hours)

IMPORTANT: Velocity estimates are heavily dependent on parameter selection (see supplemental text to original paper)

IMPORTANT: Remember that UMAP embeddings are are distorted spaces. You cannot project a velocity vector linearly across the embedding and assume that is the target destination

For a highly detailed exploration into details of RNA velocity, see Gorin et al., PLoS Comp Bio, 2022

Downstream Analysis

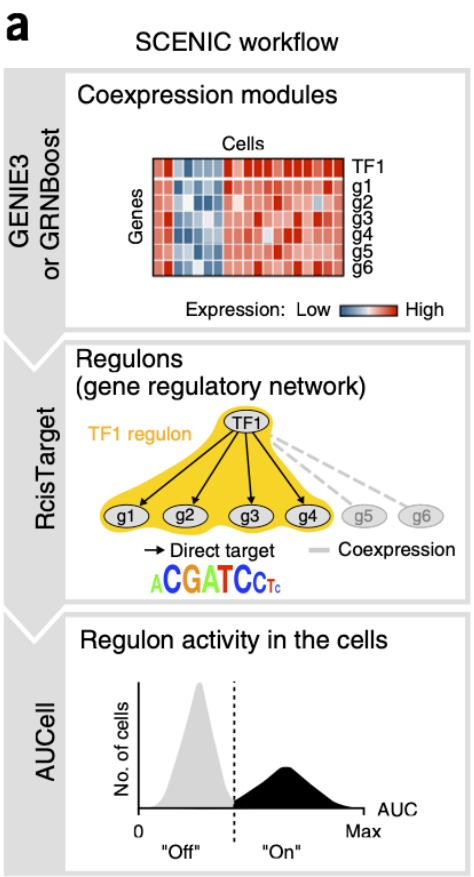
Signalling and transcription factor activity inference

Signalling inference

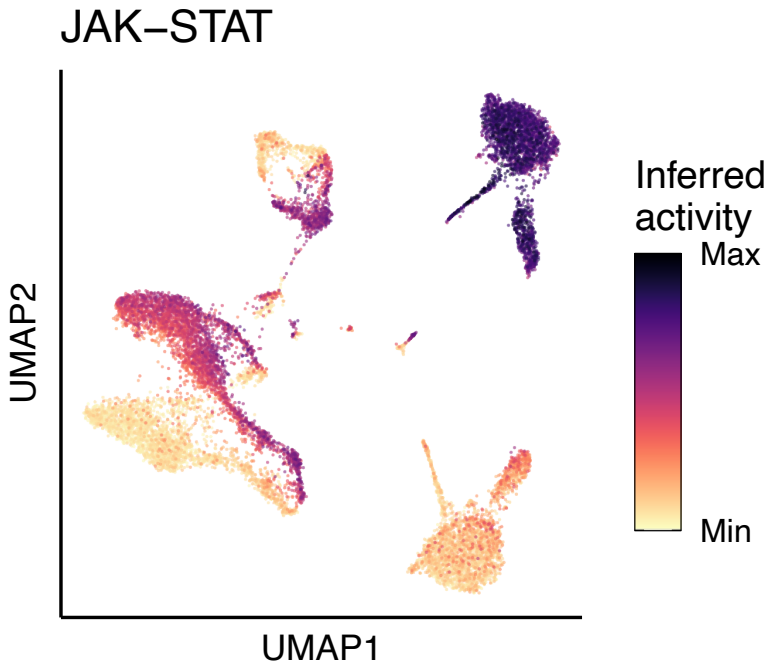
Goal

Predict the relative activity of signalling pathways and/or transcription factors across cells

Approach 1: De novo gene regulatory network construction
Eg. SCENIC



Approach 2: Use pre-built models/databases with TF/signalling-target relationships, score query data based on signatures
Eg. decoupleR
(14 signalling pathways - PROGENy; transcription factors – DoRothEA)



Higher inferred JAK-STAT signalling by PROGENy in IFNb-treated PBMCs

Downstream Analysis

Cell-cell communication inference

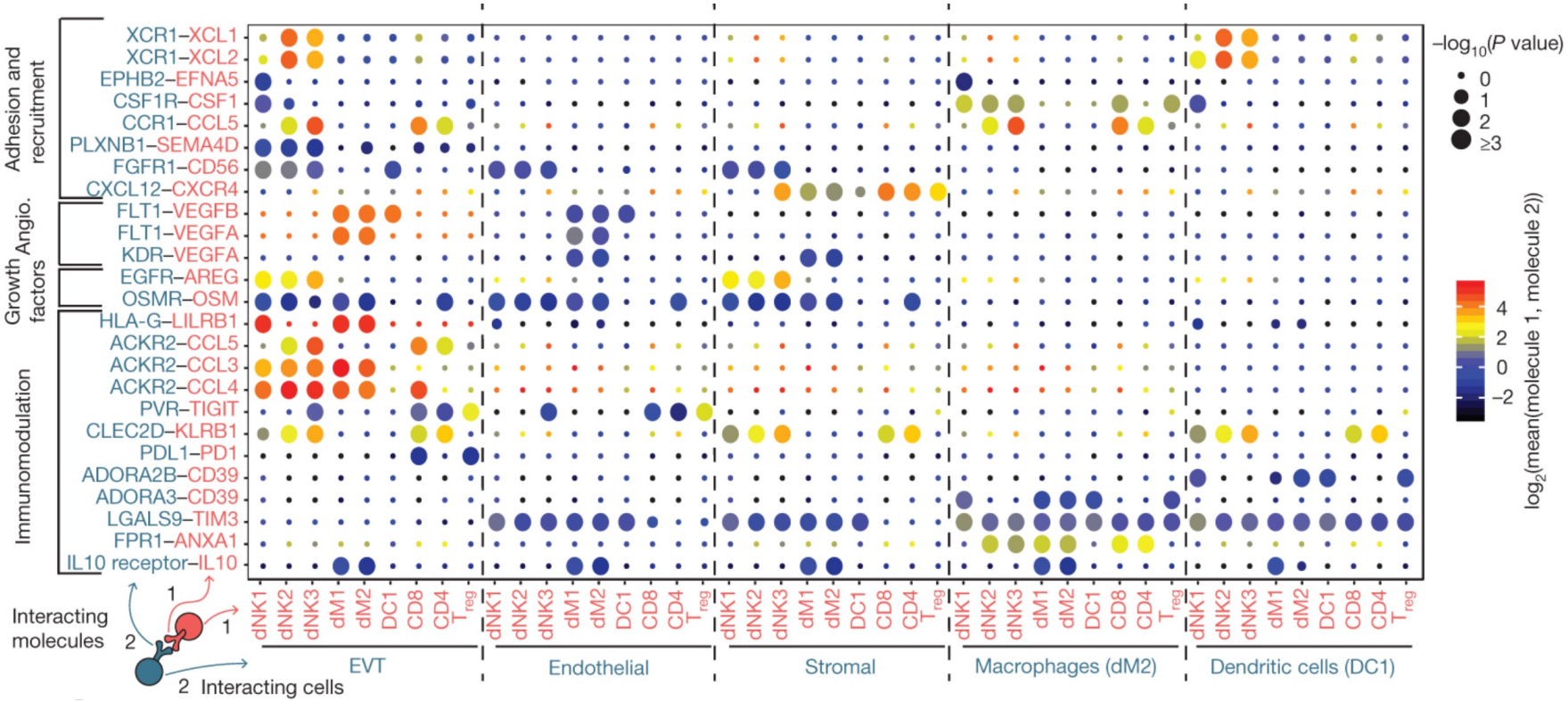
Cell-cell communication inference

Goal

Predict cell communication from the expression patterns of ligands and their cognate receptors

Approach 1: Identify ligand-receptor expression patterns specific to cell type pairs

Eg. CellPhoneDB, CellChat



Vento-Tormo *et al., Nature*, 2018

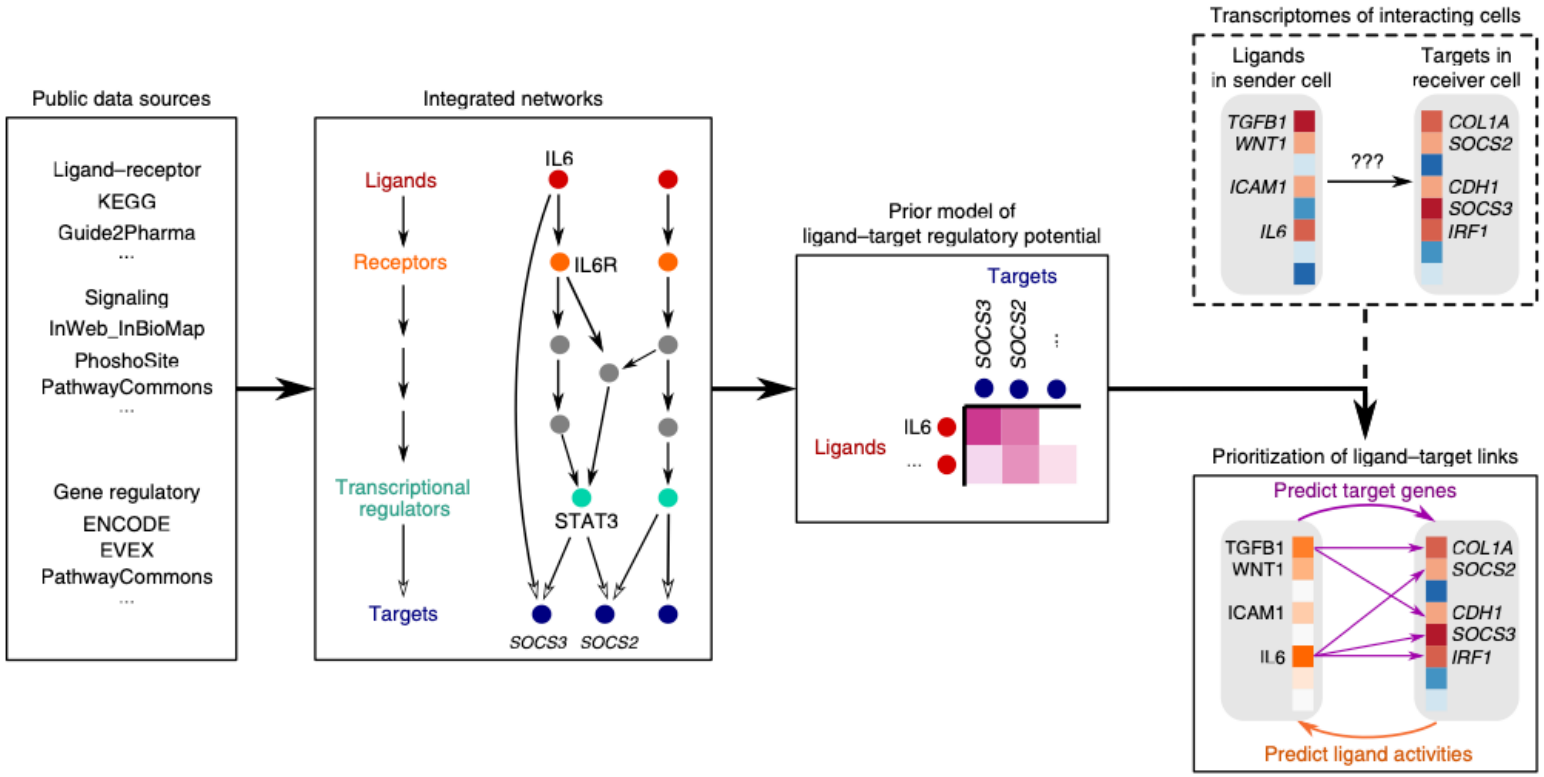
Cell-cell communication inference

Goal

Predict cell communication from the expression patterns of ligands and their cognate receptors

Approach 2: Extend on Approach 1, incorporating intracellular signalling to predict which ligand-receptor pairs are likely driving some query gene expression pattern (eg. a differential expression signature)

Eg. NicheNet



Browaeys, Saelens, & Saeys, *Nat Methods*, 2019

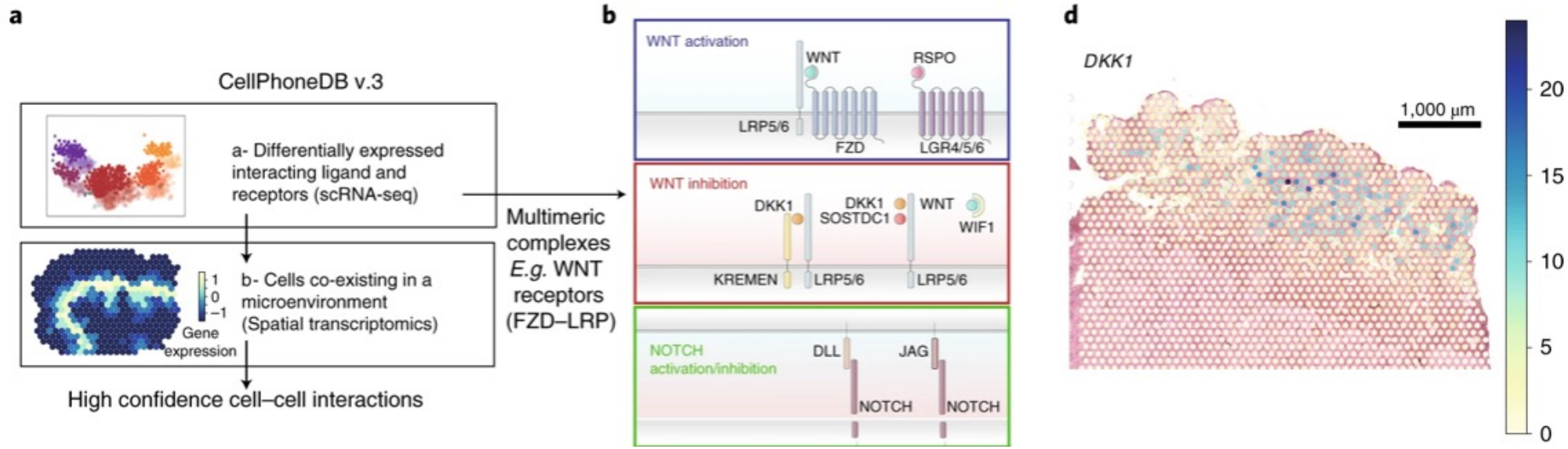
Cell-cell communication inference

Goal

Predict cell communication from the expression patterns of ligands and their cognate receptors

Approach 3: Incorporate spatial data to restrict analysis to adjacent populations

Eg. SpaTalk (Shao et al., Nat Comm, 2022), CellPhoneDB v3 (Garcia-Alonso et al., Nat Genet, 2021)



With spatial methods becoming increasingly available (and utilized), there will likely be many developments on this in the next couple years

General Advice

- Be skeptical of all results and double check the details of your code and analysis—just because you didn't get an error message doesn't mean that it's correct
- In many cases, I would argue that it's better to be a biologist first. Biological insight is critical to know if your results make any sense (and to come up with next questions!)
- Do you really want to look for trajectories or can we just cluster the cells and do differential expression?
- Please don't over-cluster your data. Cluster count is not proportional to how exciting the data is
- Don't seek help from a colleague at the first sight of an error. Spend time trying to figure out why you're getting it. It's a pain, but it's an incredible learning opportunity
- The Python ecosystem is growing rapidly. If you're in it for the long run, it wouldn't hurt to dedicate some time to learning it

Questions?