

The Krishnaswamy Laboratory
Yale Genetics and Yale SEAS present

Machine Learning for Single Cell Analysis

Online - May 20-29, 2020

When poll is active, respond at **PollEv.com/yaleml**

Text **YALEML** to **22333** once to join

What is your favorite model organism?

Machine Learning for Single Cell Analysis

Course introduction

Search Krishnaswamy Lab Help

KL Krishnaswamy L... Daniel Burkhardt

Y2 Saved items

Channel browser

People

Apps

Files

Show less

Channels

2020-workshop-byod...

2020-workshop-codin...

2020-workshop-group...

2020-workshop-main

2020-workshop-math-...

2020-workshop-misc-...

2020-workshop-tas

general

magic

meld

phate

random

scprep

workshop

#2020-workshop-main ★

20 | 1 | Add a topic

#2020-workshop-main

You created this channel on May 15th. This is the very beginning of the #2020-workshop-main channel.

Add description Add an app Add people

Friday, May 15th

Daniel Burkhardt 1:45 PM joined #2020-workshop-main along with 19 others.

Today

Pinned by you

Daniel Burkhardt 11:53 AM

Hi everyone! Welcome to the main channel for the 2020 Machine Learning for Single Cell Analysis Workshop! Please join the following channels:

1. #2020-workshop-coding-help
2. #2020-workshop-math-help
3. #2020-workshop-byod-help
4. #2020-workshop-misc-help

Message #2020-workshop-main

Aa @ 😊 🗑

<https://krishnaswamylab.org/get-help>



UNIVERSITY OF
COPENHAGEN

Washington
University in St. Louis

UNIVERSITY OF
CAMBRIDGE

W
UNIVERSITY of
WASHINGTON



FACULTY OF
MEDICINE

novo nordisk



IRELL AND
MANELLA
GRADUATE SCHOOL OF
BIOLOGICAL SCIENCES

JOHNS HOPKINS
UNIVERSITY

McMaster
University



מִצְמָן וַיִּצְמָן לְמִדְעָת
WEIZMANN INSTITUTE OF SCIENCE



COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK



Northeastern
University



WHITEHEAD
INSTITUTE



Yale

Georgia
Tech



UNIVERSITY
of VIRGINIA

UCLA

McGill



St. Jude Children's
Research Hospital
Finding cures. Saving children.



Fundación Progreso y Salud
CONSEJERÍA DE SALUD



Institut Pasteur



University of
Zurich UZH



Cold
Spring
Harbor
Laboratory

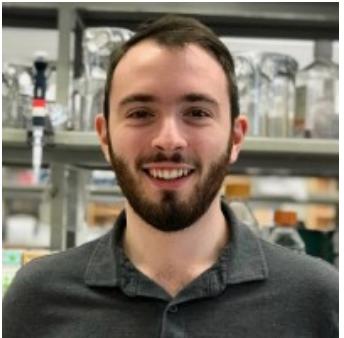


HARVARD
UNIVERSITY

IGIB
INSTITUTE OF GENOMICS & INTEGRATIVE BIOLOGY
Genomics Knowledge Partner



Smita Krishnaswamy, PhD



Daniel Burkhardt



Scott Gigante



Nur-Taz Rahman, PhD



Mamie Wang



Francesc Lopez, PhD



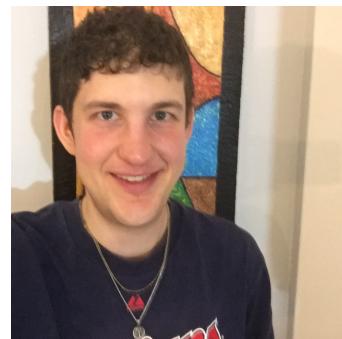
Andrew Benz



Egbert Castro



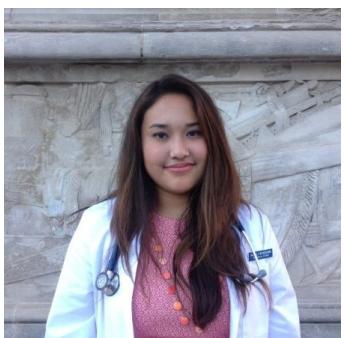
Wes Lewis



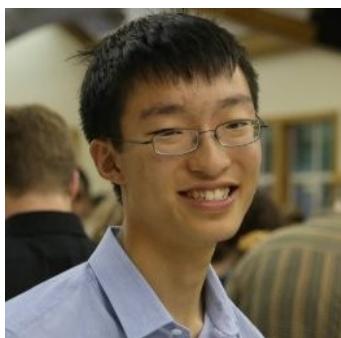
Matthew Amodio



Edel Aron



Danielle Miyagishima



Alexander Tong



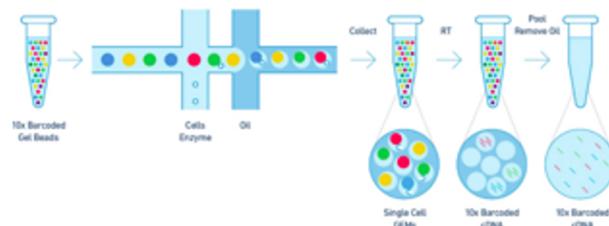
Evan Cudone



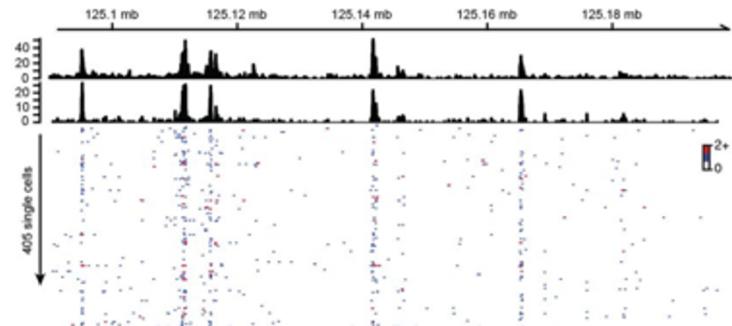
Aarthi Venkat

...and Sameet Mehta, PhD!

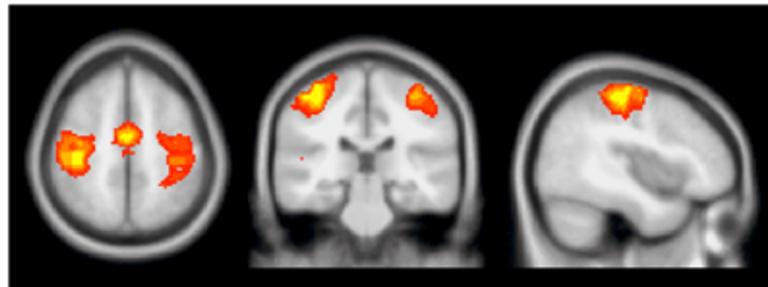
Big biomedical data



ScRNA-seq



ScATAC-seq



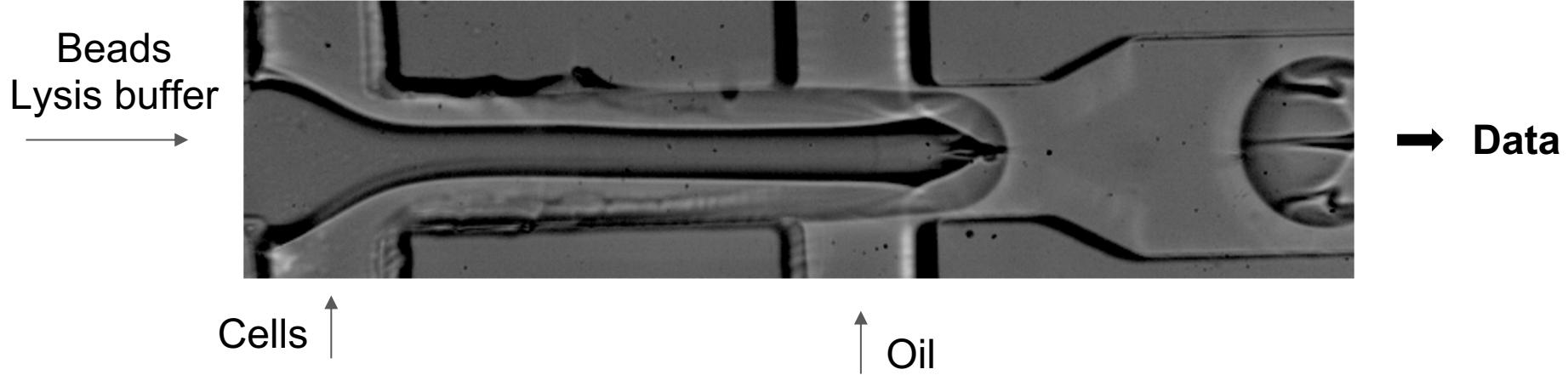
FMRI



Patient Data

Big = Any dataset with many many observations

The single cell revolution



The single cell revolution

Interesting Biological Experiments



Computation



High impact paper

LETTER

<https://doi.org/10.1038/nature08606>

RNA velocity of single cells

Görel La Manno^{1,3}, Rulan Sudjic^{1,6}, Amit Zelzer^{1,7}, Esterle Braun^{1,3}, Hannah Hochegger^{1,3}, Viktor Pejković^{1,8}, Katja Lischefski^{1,9}, Maria F. Kastell¹, Peter Lönnerberg^{2,3}, Alessandro Pusztai¹, Jean Fan¹, Lars E. Birn^{1,2}, Zehan Liu¹, Daniel C. Westover¹, Michael A. Hodge¹, Daniel J. Sisodia¹⁰, Michael S. Henschen¹¹, Gonzalo Caočić¹¹, Bruno P. Svartman¹², Igor Mamedov¹³, Sten Linnarsson^{1,2,4} & Peret V. Kharchenko^{1,14}

RNA abundance is a powerful indicator of the state of individual cells. Single-cell RNA sequencing can reveal RNA abundance with high quantitative accuracy, without the need for labeling or microscopy. However, a static snapshot of a point in time poses a challenge for the analysis of time-resolved phenomena such as gene expression dynamics. The concept of RNA velocity—the time derivative of the gene expression state—can be used to predict the future state of a cell by monitoring the rate of change of RNA abundance. RNA velocity is a high-dimensional vector that predicts the future state of a cell based on its current RNA profile. By applying RNA velocity to the neural crest lineage, demonstrated its use on multiple published datasets, and used it to predict the fate of the neural crest, the kinetics of the developing mouse hippocampus, and examine the kinetics of differentiation in human pluripotent stem cells. RNA velocity can greatly aid the analysis of developmental lineage and cellular heterogeneity, particularly in humans.

During development, transcription occurs on a timescale of hours to days, which is comparable to the typical half-life of mRNA. The RNA velocity model is able to capture this timescale and can be exploited to estimate the rates of gene splicing and degradation, and to predict the future state of a cell based on the observed rate of change of the entire transcriptome during dynamic processes.

All common single-cell RNA-seq protocols rely on oligo-dT primers to select polyA+ RNA. We found that RNA velocity can be calculated using single-cell RNA-seq datasets based on the SMART-seq², STRT³, and Dropbead⁴ protocols. We analyzed 12,353 RNA-seq samples (5–25% of reads contained unspliced intrinsic sequences (Fig. 1a), in 1–20% RNA seq). Most such reads originated from secondary structures of pre-mRNAs, which are predominantly found in Genomics Chromatin libraries, we also found abundant discordant RNA-seq reads, which commonly occur in native polyA⁺ RNA. After filtering for these artifacts, we performed RNA velocity calculations by priming on the first-stranded cDNA. The substantial fraction of unspliced RNA in the dataset, however, prevents us from extrapolating RNA velocity to the future. To do so, we had to extrapolate the measured mRNA abundance into the future, we examined a time course of mRNA abundance in the mouse hippocampus during development¹². Unspliced mRNA levels at each time point were consistently higher than spliced mRNA levels (Extended Data Fig. 1a–c), and many circadian associated genes showed the expected excess of unspliced mRNA relative to the splice¹³ during spermatogenesis, and a similar pattern was observed in the mouse hippocampus. Using the proposed differential equations for each gene allowed us to extrapolate the measured mRNA abundance into the future, we then compared the expected direction of progression of the circadian cycle (Fig. 1b). The direction of progression of the circadian cycle was consistent with the direction of progression of the hippocampus during development¹². In single-cell measurements, we analyzed recently published single-cell mRNA abundance data from the mouse hippocampus during development¹³ (Fig. 2). During development, a substantial proportion of olfactory ensheathing cells, which are neuroectoderm cells of the olfactory modula, show simple first-order kinetics, as expected if unspliced RNA expressed in these cells.

To quantify the time-dependent relationship between the abundance of precursor and mature mRNA we assumed a simple model

¹Department of Molecular Neurobiology, National Institute of Mental Health, Bethesda, Maryland, USA. ²Department of Cell Biology, University of Massachusetts Medical School, Worcester, Massachusetts, USA. ³Department of Cell Biology, University of Massachusetts Medical School, Worcester, Massachusetts, USA. ⁴Department of Biochemistry and Molecular Biology, University of Massachusetts Medical School, Worcester, Massachusetts, USA. ⁵Department of Cell Biology, University of Massachusetts Medical School, Worcester, Massachusetts, USA. ⁶Department of Cell Biology, University of Massachusetts Medical School, Worcester, Massachusetts, USA. ⁷Department of Cell Biology, University of Massachusetts Medical School, Worcester, Massachusetts, USA. ⁸Department of Cell Biology, University of Massachusetts Medical School, Worcester, Massachusetts, USA. ⁹Department of Cell Biology, University of Massachusetts Medical School, Worcester, Massachusetts, USA. ¹⁰Department of Cell Biology, University of Massachusetts Medical School, Worcester, Massachusetts, USA. ¹¹Department of Cell Biology, University of Massachusetts Medical School, Worcester, Massachusetts, USA. ¹²Department of Cell Biology, University of Massachusetts Medical School, Worcester, Massachusetts, USA. ¹³Department of Cell Biology, University of Massachusetts Medical School, Worcester, Massachusetts, USA. ¹⁴Department of Cell Biology, University of Massachusetts Medical School, Worcester, Massachusetts, USA.

474 | NATURE | VOL 540 | 23 AUGUST 2016

The single cell revolution

Interesting Biological Experiments



Computation



High impact paper

LETTER

<https://doi.org/10.1038/nature08416>

RNA velocity of single cells

Gioele La Manno^{1,3}, Rulan Sudhakar⁴, Amit Zelzer¹, Esterle Braun^{1,3}, Hannah Hochegger^{1,3}, Viktor Pejchal^{1,3}, Katja Lischefski^{1,2}, Maria E. Kastell¹, Peter Lönnerberg^{2,3}, Alessandro Pusztai¹, Jean Fan¹, Lars E. Börre^{1,3}, Zehan Liu¹, Daniel C. Westover¹, Michael A. Hodge¹, Michael S. Hengenberger¹, Gonzalo Caočić¹, Barbara A. Knoblich², Igor Malenky⁵, Sten Linnemann^{2,3} & Peter V. Kharchenko^{1,3*}

RNA abundance is a powerful indicator of the state of individual cells. Single-cell RNA sequencing can reveal RNA abundance with high quantitative accuracy, without the need for a reference transcriptome. However, a static snapshot of a transcriptome in time poses a challenge for the analysis of time-resolved phenomena such as gene expression dynamics. The concept of RNA velocity—the time derivative of the gene expression state—can be used to predict the future state of a cell. RNA velocity is calculated as the difference between the total RNA abundance in a cell and the RNA abundance in common single-cell RNA sequencing protocols. RNA velocity is a high-dimensional vector that predicts the future state of a cell based on its current transcriptomic profile. We analyzed the neural crest lineage, demonstrated its use on multiple published datasets, and applied it to the analysis of the early development of the developing mouse hippocampus, and examine the kinetics of gene expression dynamics in the hippocampus. Our results show that RNA velocity can greatly aid the analysis of developmental lineages and cellular differentiation, particularly in humans.

During transcription, RNA production occurs on a timescale of hours to days, which is comparable to the typical half-life of mRNA. The RNA velocity of a gene can therefore be used to predict the direction of change of the entire transcriptome during dynamic processes.

All common single-cell RNA-seq protocols rely on oligo-dT primers to select poly(A)+ RNA. This selection bias has been shown to affect RNA-seq data quality^{1–3}. We examined the effect of this selection bias on RNA-seq data using single-cell RNA-seq datasets based on the SMART-seq², STRT⁴, and Dropbead⁵ protocols. We found that approximately 15–25% of reads contained unspliced intrinsic sequences (Fig. 1a), in contrast to the expected 1–2% of unspliced RNA in a standard 1–20% RNA-seq. Most such reads originated from secondary priming by reverse transcriptase, as shown by sequencing of Genomics Chromatin libraries, we also found abundant discordant sequencing reads, commonly occurring intrinsic polyA⁺ sequences that were not aligned to the poly(A) tail. These discordant reads were amplified by priming on the first-stranded cDNA. The substantial fraction of unspliced RNA in the RNA-seq data suggests that RNA amplification in single-cell RNA-seq may result in a significant bias in the quantification of RNA abundance. To validate our findings, we attempted to extrapolate the expected RNA abundance in the future, we examined a time course of RNA abundance in the mouse hippocampus during development⁶. Unspliced mRNA levels at each time point were consistently higher than spliced mRNA levels (Fig. 1b). In addition, many circadian-associated genes showed the expected excess of unspliced mRNA relative to the splice during spermatogenesis, and a similar pattern was observed in the mouse hippocampus (Fig. 1c). The proposed differential equations for each gene allowed us to extrapolate the expected direction of progression of the circadian cycle (Fig. 1d). The results of the RNA velocity analysis were consistent with the expected direction of progression of the circadian cycle (Fig. 1e).

To validate the RNA velocity analysis, we analyzed the RNA abundance in single-cell RNA-seq data from the mouse hippocampus in single-cell measurements, we analyzed recently published single-cell mRNA abundance data from the mouse hippocampus using STRT⁴ (Fig. 2). During development, a substantial proportion of olfactory ensheathing cells, which are neuroectoderm cells of the olfactory lamina, show a distinct RNA velocity profile compared to other cells in which the direction of differentiation can be validated by lineage tracing. Phase portraits of many genes showed the expected deviation

*Correspondence: kharchenko@mit.edu. Author contributions: G.L.M., R.S., A.Z., E.B., H.H., V.P., K.L., M.A.H., M.S.H., D.C.W., M.A.K., G.C., B.A.K., I.M., S.L., P.V.K. and P.Z. contributed equally to this work.

¹Department of Molecular Neuroscience, Novartis Research Foundation, San Diego, California, USA. ²Department of Cell Biology, Harvard Medical School, Boston, Massachusetts, USA. ³Department of Systems Biology, Harvard Medical School, Boston, Massachusetts, USA. ⁴Department of Resources and Robotics, Karlsruhe Institute of Technology, Karlsruhe, Germany. ⁵Department of Physics and Pharmacology, Max-Planck-Institut für Biophysik, Frankfurt am Main, Germany. ⁶Department of Cell Biology, Harvard Medical School, Boston, Massachusetts, USA. ⁷Department of Cell Biology, Harvard Medical School, Boston, Massachusetts, USA. ⁸Department of Cell Biology, Harvard Medical School, Boston, Massachusetts, USA. ⁹Department of Cell Biology, Harvard Medical School, Boston, Massachusetts, USA. ¹⁰Department of Cell Biology, Harvard Medical School, Boston, Massachusetts, USA. ¹¹Department of Cell Biology, Harvard Medical School, Boston, Massachusetts, USA. ¹²Department of Cell Biology, Harvard Medical School, Boston, Massachusetts, USA. ¹³Department of Cell Biology, Harvard Medical School, Boston, Massachusetts, USA. ¹⁴Department of Cell Biology, Harvard Medical School, Boston, Massachusetts, USA.

474 | NATURE | VOL 460 | 23 AUGUST 2009

- Machine learning
- Linear algebra
- Probability theory
- Statistical analysis
- Algorithm design

It's all Greek to me...

Definition 1. The t -step potential distance is defined as $\mathfrak{V}^t(x, y) \triangleq \|U_x^t - U_y^t\|_2$, $x, y \in \mathcal{X}$.

The following proposition shows a relation between the two metrics by expressing the potential distance in embedded diffusion map coordinates¹ for fixed-bandwidth Gaussian-based diffusion (i.e., generated by P_ε from Eq. 2):

Proposition 1. Given a diffusion process defined by a fixed-bandwidth Gaussian kernel, the potential distance from Def 1 can be written as $\mathfrak{V}^t(x, y) = \left(\sum_{z \in \mathcal{X}} \log^2 \left(\frac{1 + \langle \Phi^{t/2}(x), \Phi^{t/2}(z) \rangle}{1 + \langle \Phi^{t/2}(y), \Phi^{t/2}(z) \rangle} \right) \right)^{1/2}$

Proof. According to the spectral theorem, the entries of P_ε^t can be written as

$$[P_\varepsilon^t]_{(x,y)} = \psi_0(y) + \sum_{i=1}^{n-1} \lambda_i^t \phi_i(x) \psi_i(y)$$

since powers of the operator P_ε only affect the eigenvalues, which are taken to the same power, and since the trivial eigenvalue λ_0 is one and the corresponding right eigenvector ϕ_0 only consists of ones. Furthermore, it can be verified that the left and right eigenvectors of P_ε are related by $\psi_i(y) = \phi_i(y) \psi_0(y)$, thus, combined with Eqs. 4 and 6, we get

$$p_{\varepsilon,x}^t(y) = \psi_0(y) \left(1 + \sum_{i=1}^{n-1} \lambda_i^t \phi_i(x) \phi_i(y) \right) = \psi_0(y) (1 + \langle \Phi_\varepsilon^{t/2}(x), \Phi_\varepsilon^{t/2}(y) \rangle) .$$

By applying the logarithm to both ends of this equation we express the entries of the potential representation $U_{\varepsilon,x}^t$ as

$$U_{\varepsilon,x}^t(y) = -\log(1 + \langle \Phi_\varepsilon^{t/2}(x), \Phi_\varepsilon^{t/2}(y) \rangle) - \log(\psi_0(y)) ,$$

and thus for any $j = 1, \dots, N$,

$$\begin{aligned} (U_{\varepsilon,x}^t(x_j) - U_{\varepsilon,y}^t(x_j))^2 &= [\log(1 + \langle \Phi_\varepsilon^{t/2}(x), \Phi_\varepsilon^{t/2}(x_j) \rangle)]^2 \\ &\quad - [\log(1 + \langle \Phi_\varepsilon^{t/2}(y), \Phi_\varepsilon^{t/2}(x_j) \rangle)]^2 \\ &= \log^2 \left(\frac{1 + \langle \Phi_\varepsilon^{t/2}(x), \Phi_\varepsilon^{t/2}(x_j) \rangle}{1 + \langle \Phi_\varepsilon^{t/2}(y), \Phi_\varepsilon^{t/2}(x_j) \rangle} \right) , \end{aligned}$$

which yields the result in the proposition. \square

What reading single cell methods can feel like



What is machine learning?

What is machine learning?

Machine learning is the process of identifying patterns in data.

Two kinds of machine learning

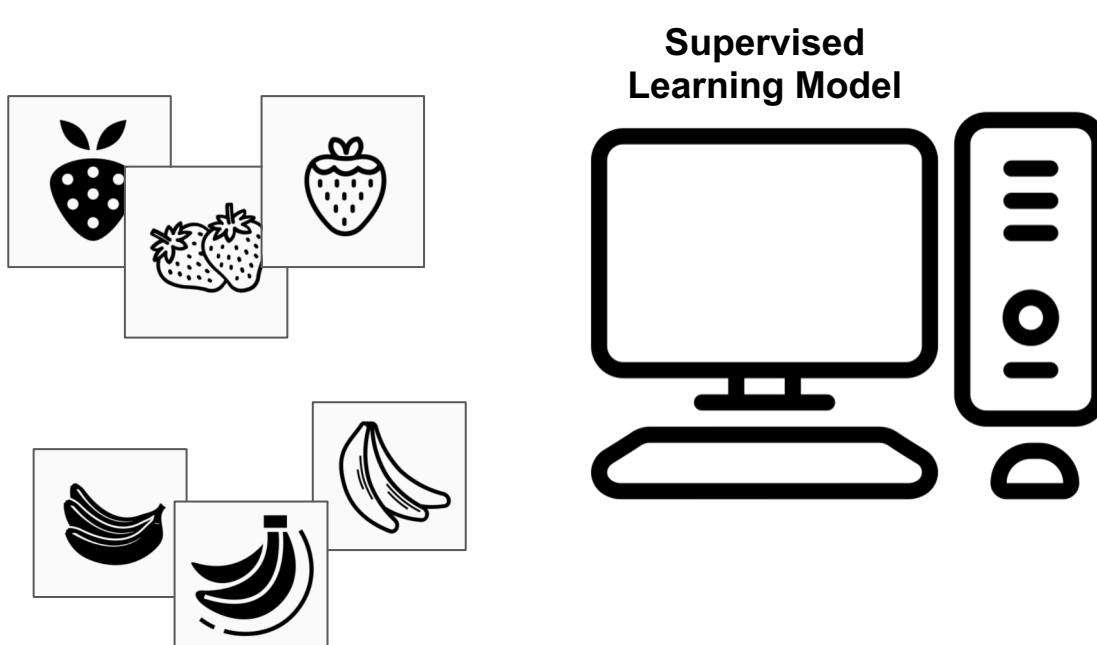
Supervised learning

- Have a bunch of labelled data, want to label new data

Two kinds of machine learning

Supervised learning

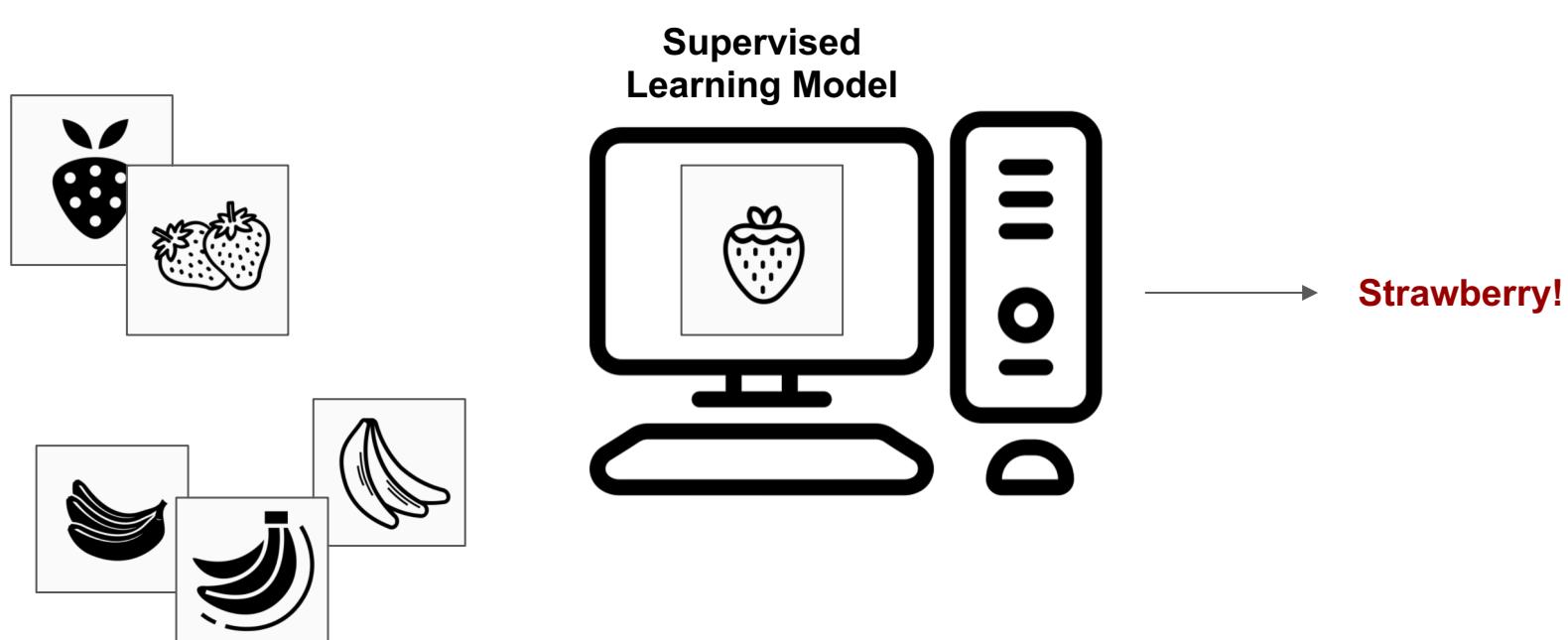
- Have a bunch of labelled data, want to label new data



Two kinds of machine learning

Supervised learning

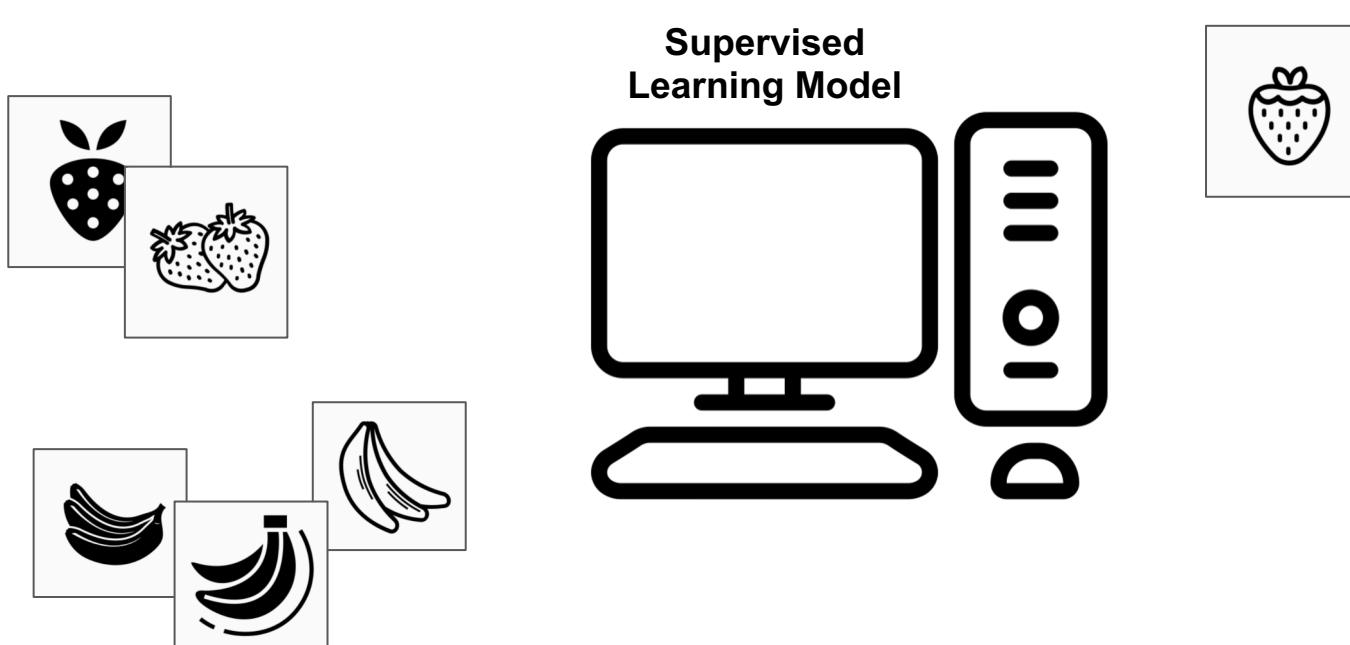
- Have a bunch of labelled data, want to label new data



Two kinds of machine learning

Supervised learning

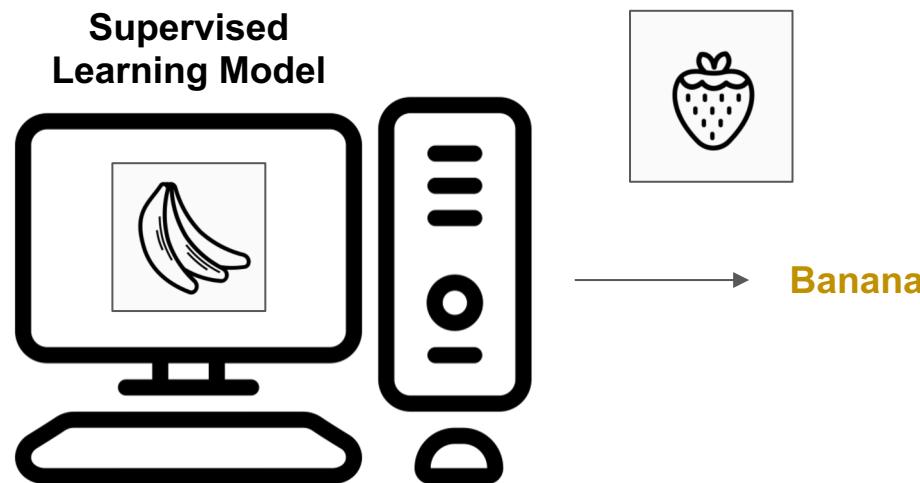
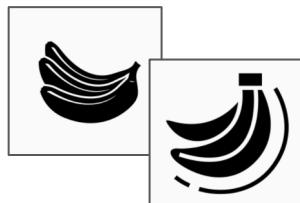
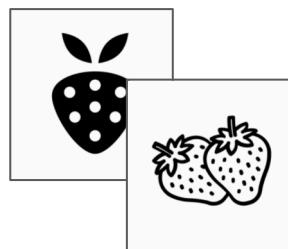
- Have a bunch of labelled data, want to label new data



Two kinds of machine learning

Supervised learning

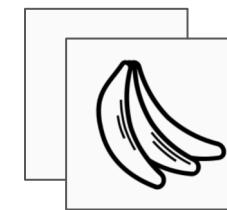
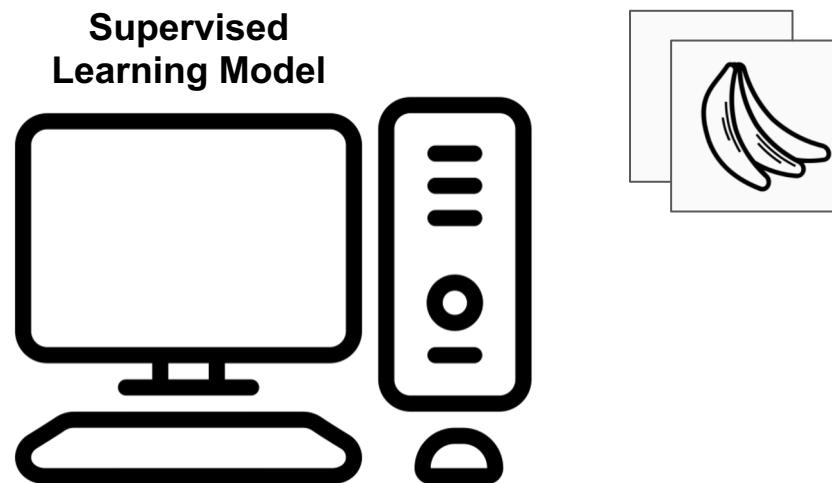
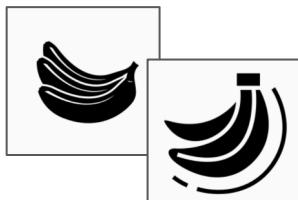
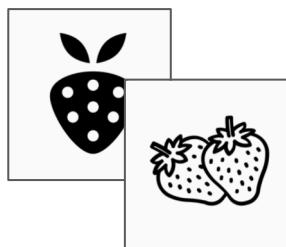
- Have a bunch of labelled data, want to label new data



Two kinds of machine learning

Supervised learning

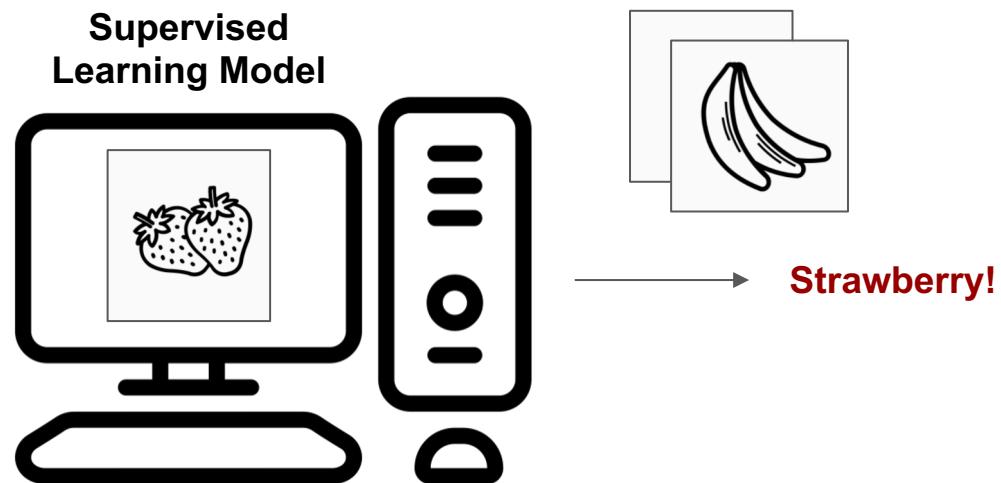
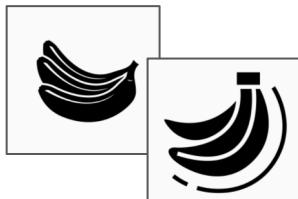
- Have a bunch of labelled data, want to label new data



Two kinds of machine learning

Supervised learning

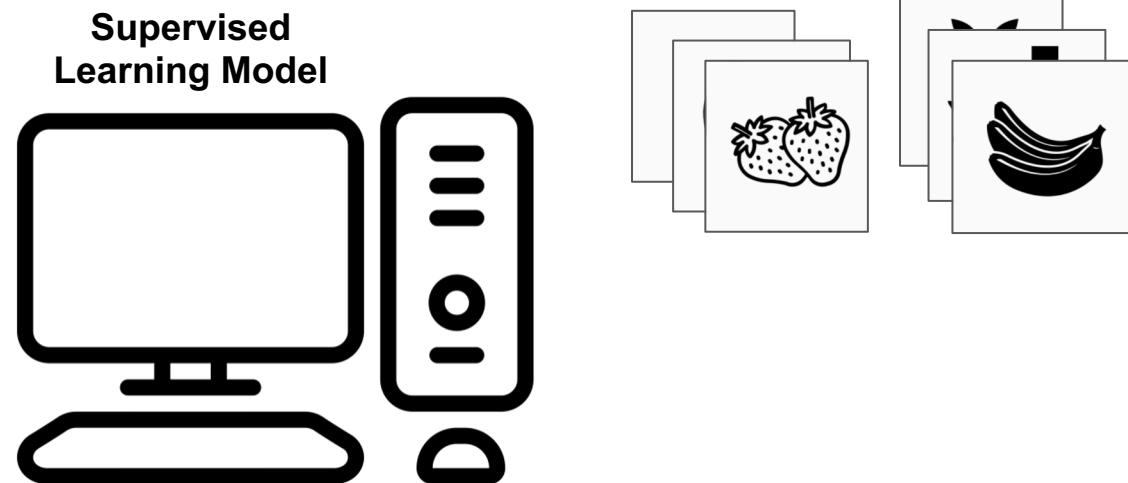
- Have a bunch of labelled data, want to label new data



Two kinds of machine learning

Supervised learning

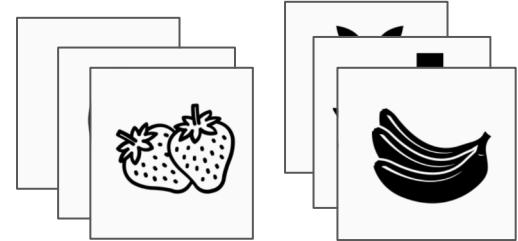
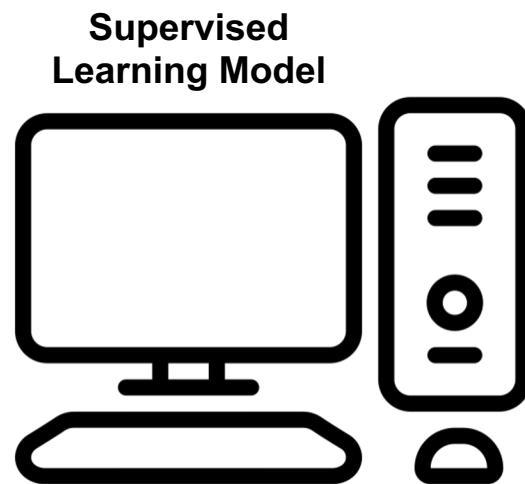
- Have a bunch of labelled data, want to label new data



Two kinds of machine learning

Supervised learning

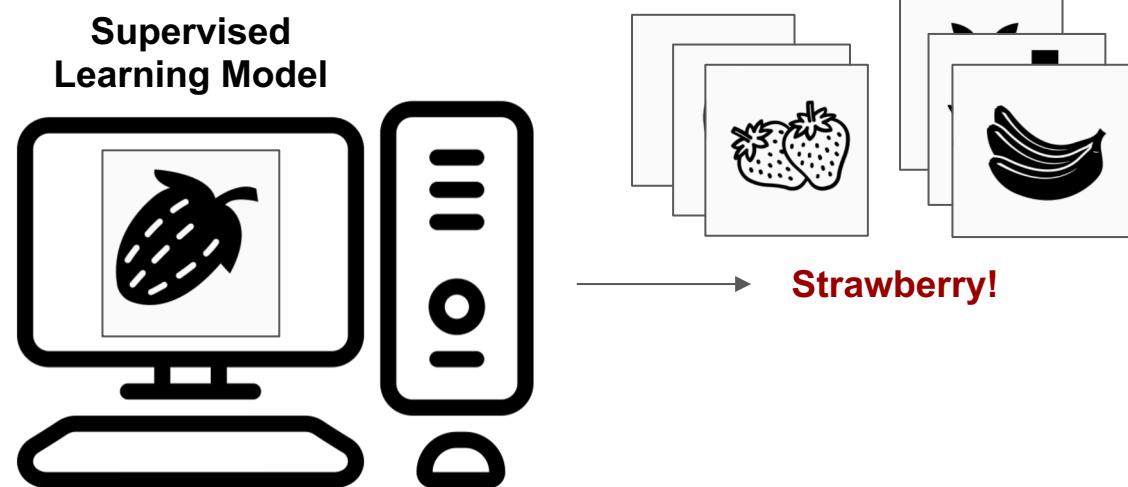
- Have a bunch of labelled data, want to label new data



Two kinds of machine learning

Supervised learning

- Have a bunch of labelled data, want to label new data



Two kinds of machine learning

Supervised learning

- Have a bunch of labelled data, want to label new data

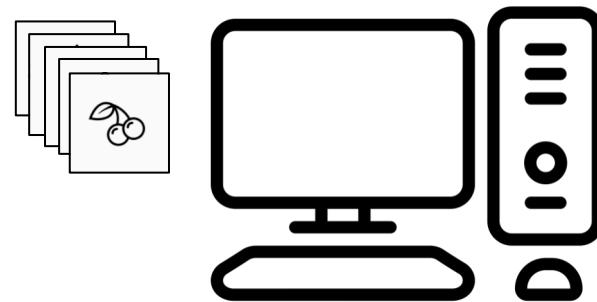
Supervised Learning Model



Unsupervised learning

- Have a bunch of unlabeled data, want to organize it

Unsupervised Learning Model



Two kinds of machine learning

Supervised learning

- Have a bunch of labelled data, want to label new data

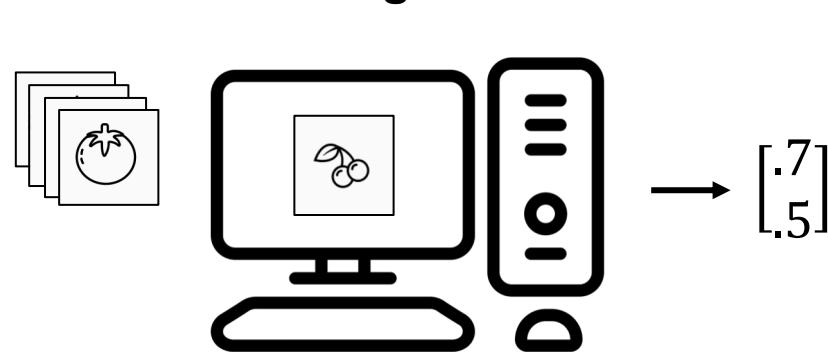
Supervised Learning Model



Unsupervised learning

- Have a bunch of unlabeled data, want to organize it

Unsupervised Learning Model



Two kinds of machine learning

Supervised learning

- Have a bunch of labelled data, want to label new data

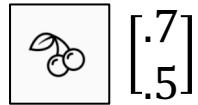
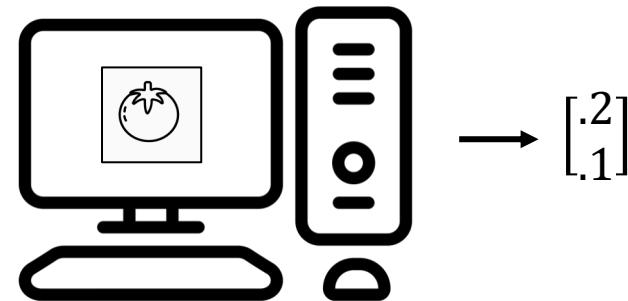
Supervised Learning Model



Unsupervised learning

- Have a bunch of unlabeled data, want to organize it

Unsupervised Learning Model



Two kinds of machine learning

Supervised learning

- Have a bunch of labelled data, want to label new data

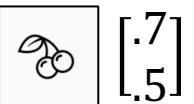
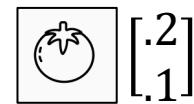
Supervised Learning Model



Unsupervised learning

- Have a bunch of unlabeled data, want to organize it

Unsupervised Learning Model

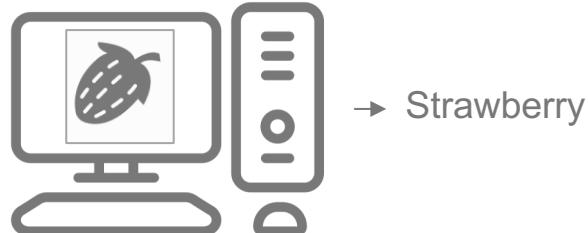


Two kinds of machine learning

Supervised learning

- Have a bunch of labelled data, want to label new data

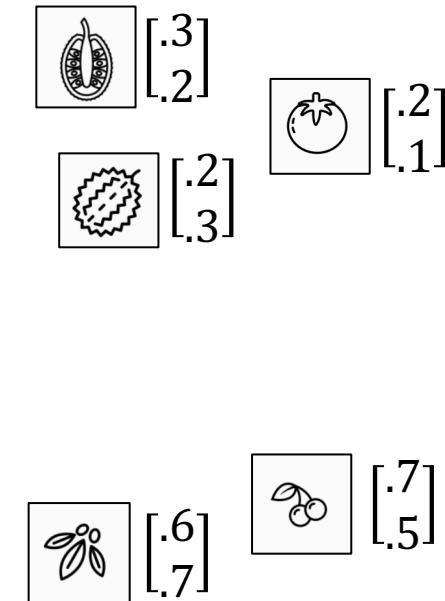
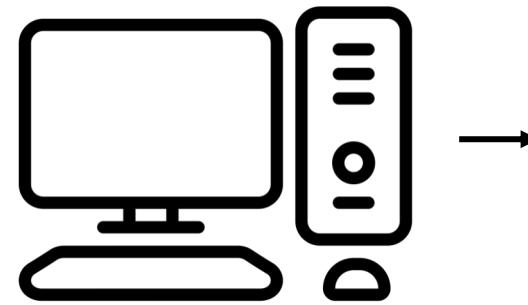
Supervised Learning Model



Unsupervised learning

- Have a bunch of unlabeled data, want to organize it

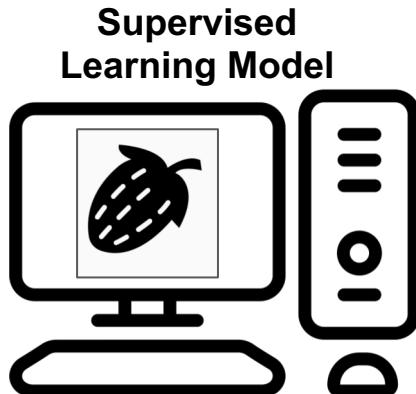
Unsupervised Learning Model



Two kinds of machine learning

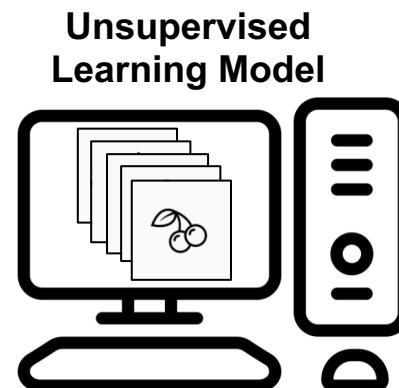
Supervised learning

- Have a bunch of labelled data, want to label new data
- Learn a function $f(X) \rightarrow Y$ where all values of Y are known for some samples of X



Unsupervised learning

- Have a bunch of unlabeled data, want to organize it
- Learn an embedding $f(X) \rightarrow Y, X \in \mathbb{R}^n, Y \in \mathbb{R}^m, n \gg m$
- Lower dimensional, easier to interpret (e.g. as clusters)



Is linear regression an example of supervised or unsupervised machine learning?

Supervised
machine
learning

Unsupervised
machine
learning

Is clustering an example of supervised or unsupervised machine learning?



Supervised
machine learning **A**

Unsupervised
machine learning **B**

Course Schedule

The screenshot shows a web browser window with the title "Workshop — Krishnaswamy Lab" and the URL "https://www.krishnaswamylab.org/workshop". The page content is organized into sections for each day of the workshop.

Course Schedule

Day 1 – Wednesday, May 20th

Lecture	View on Google Drive	Introduction to scRNA-seq and Preprocessing
Exercise	Run in Google Colab	1.0. Preprocessing Embryoid Body Data (Beginner)
	Run in Google Colab	1.0. Preprocessing Embryoid Body Data (Advanced)
	Run in Google Colab	1.1. Loading and pre-processing your own data (optional)

Day 2 – Thursday, May 21st

Lecture	View on Google Drive	Manifold Learning and Dimensionality Reduction
Exercise	Run in Google Colab	2.0. Plotting UCI Wine Data
	Run in Google Colab	2.1. Learning Graphs from Data
	Run in Google Colab	2.2. Visualizing UCI Wine Data
	Run in Google Colab	2.3. PCA on Retinal Bipolar Data
	Run in Google Colab	2.4. Visualizing Retinal Bipolar Data
	Run in Google Colab	2.5. Visualizing Embryoid Body Data (Advanced)

Day 3 – Friday, May 22nd

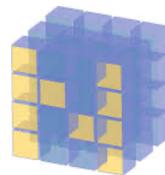
Lecture	View on Google Drive	Clustering and Data Denoising
Exercise	Run in Google Colab	3.0 Clustering Toy Data (Beginner)
	Run in Google Colab	3.0 Clustering Toy Data (Advanced)
	Run in Google Colab	3.1 Clustering & Denoising Embryoid Body Data (Advanced)
	Run in Google Colab	3.2 Batch correction in PBMCs

Day 4 – Wednesday, May 27th

<https://www.krishnaswamylab.org/workshop>

Why Python?

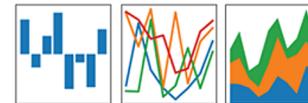




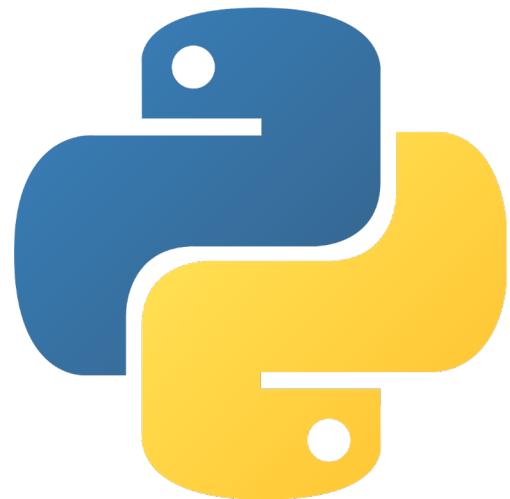
NumPy

pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



Why Python?



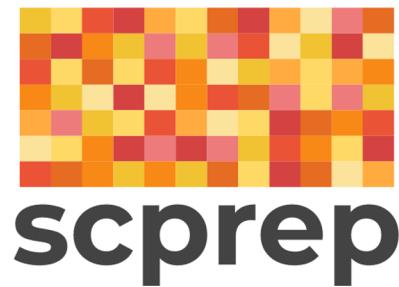
Tensorflow



Pytorch



scanpy



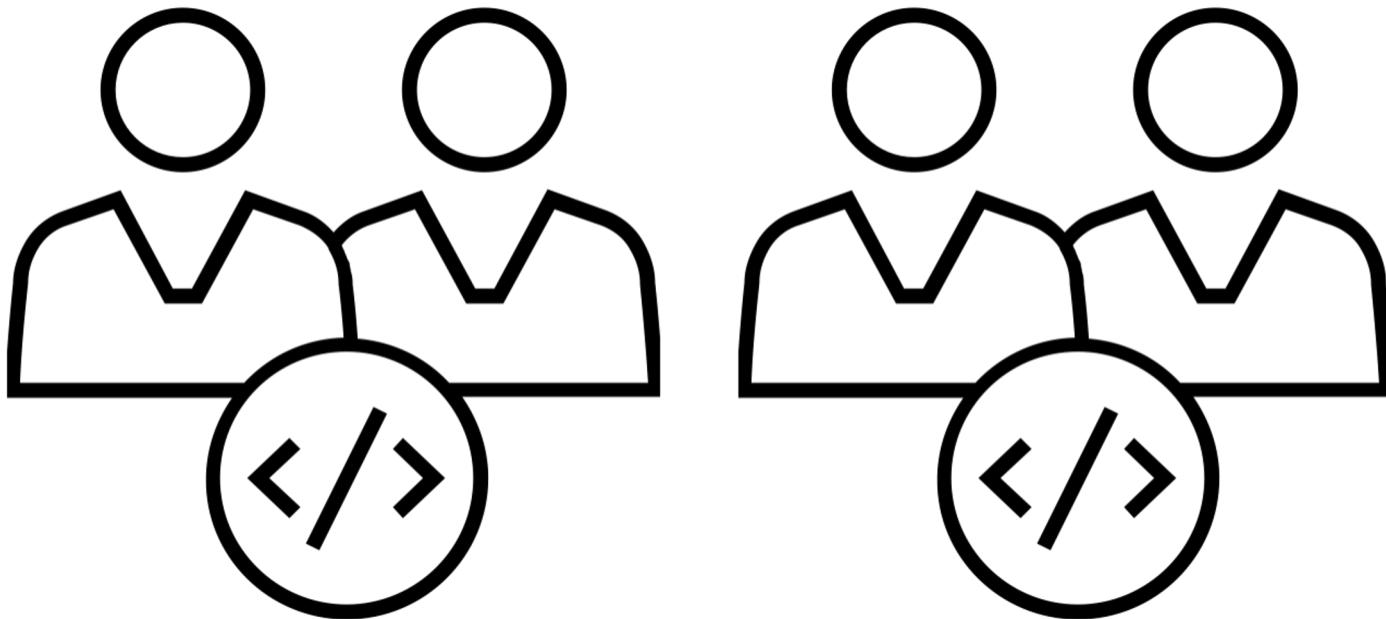
scprep

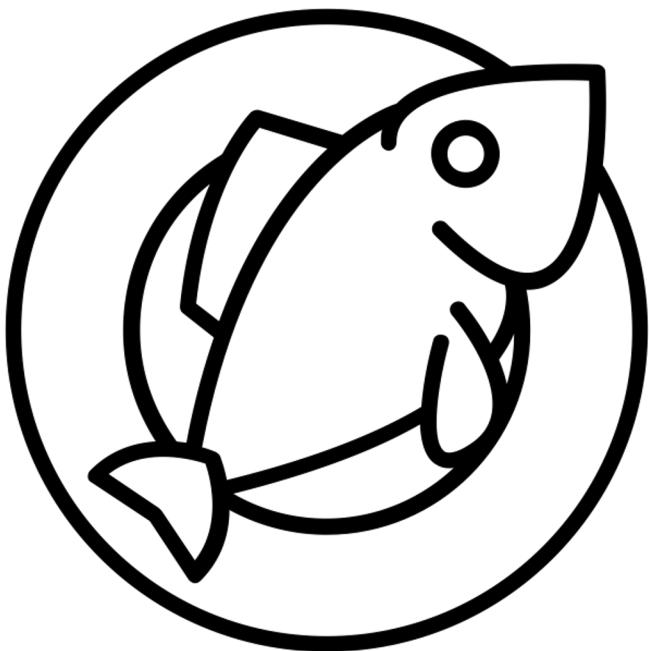
The screenshot shows the Google Colab interface. At the top, there's a browser-like header with tabs, a search bar, and various icons. Below it is the Colab navigation bar with links for File, Edit, View, Insert, Runtime, Tools, Help, and user profile. A sidebar on the left contains a 'Table of contents' section with links to Getting started, Data science, Machine learning, More Resources, and Machine Learning Examples. There's also a '+ SECTION' button. The main content area features a large yellow 'CO' logo and the title 'What is Colab?'. It explains that Colab allows writing and executing Python in a browser with zero configuration, free access to GPUs, and easy sharing. It encourages users to watch the 'Introduction to Colab' video. Below this, a section titled 'Getting started' is expanded, explaining that the document is an interactive Colab notebook. It shows a code cell with the Python script:

```
[ ] seconds_in_a_day = 24 * 60 * 60  
seconds_in_a_day
```

 and the output:  72000. A note says variables defined in one cell can be used in others.

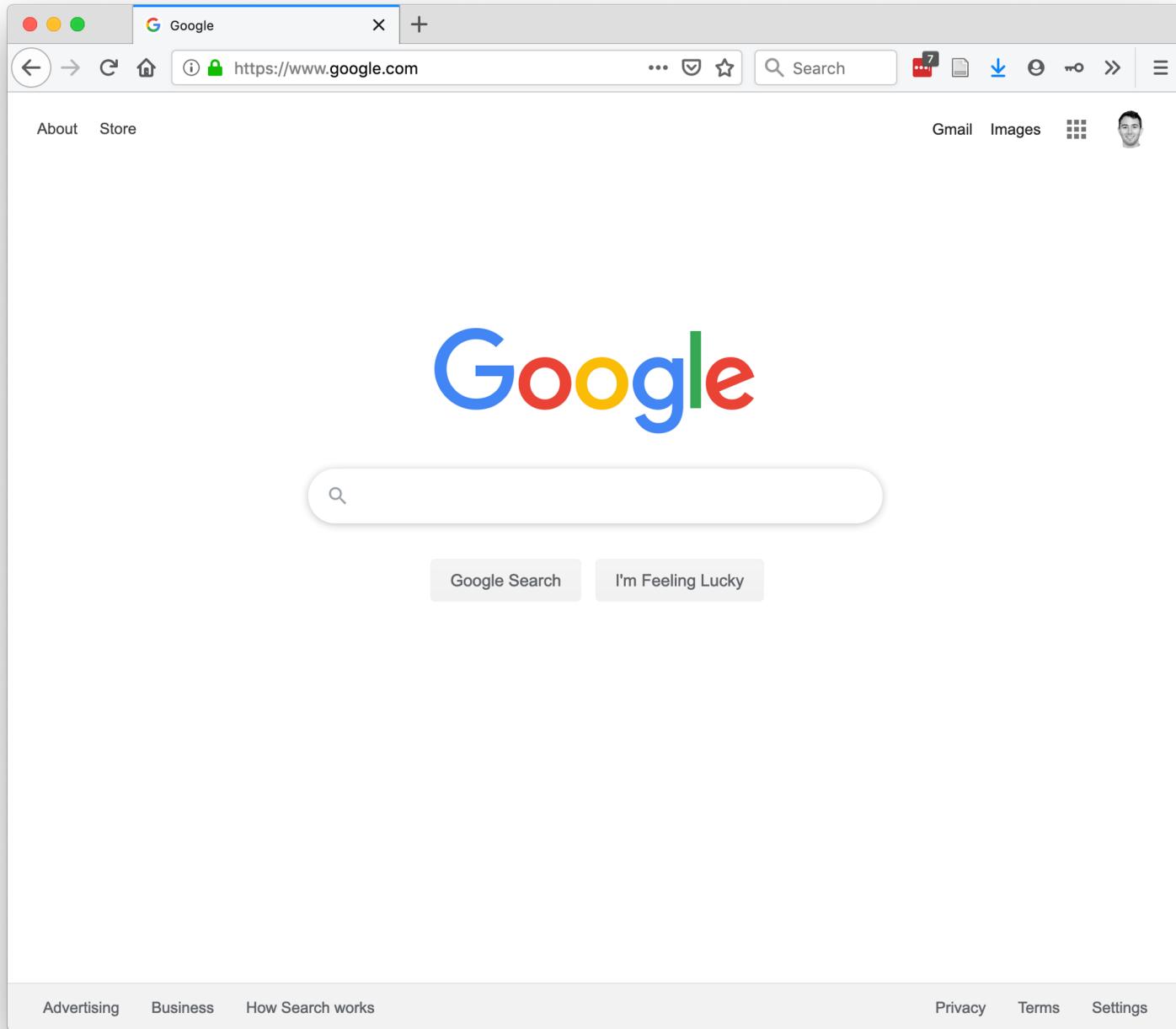
Team programming





vs.





Reference — scprep 1.0.1 documentation

scprep.io.load_10X(*data_dir*, *sparse=True*, *gene_labels='symbol'*, *allow_duplicates=None*) [source]

Basic IO for 10X data produced from the 10X Cellranger pipeline.

A default run of the *cellranger count* command will generate gene-barcode matrices for secondary analysis. For both “raw” and “filtered” output, directories are created containing three files: ‘matrix.mtx’, ‘barcodes.tsv’, ‘genes.tsv’. Running *scprep.io.load_10X(data_dir)* will return a Pandas DataFrame with genes as columns and cells as rows.

Parameters:

- ***data_dir* (string)** – path to input data directory expects ‘matrix.mtx’, ‘genes.tsv’, ‘barcodes.tsv’ to be present and will raise an error otherwise
- ***sparse* (boolean)** – If True, a sparse Pandas DataFrame is returned.
- ***gene_labels* (string, {‘id’, ‘symbol’, ‘both’} optional, default: ‘symbol’)** – Whether the columns of the dataframe should contain gene ids or gene symbols. If ‘both’, returns symbols followed by ids in parentheses.
- ***allow_duplicates* (bool, optional (default: None))** – Whether or not to allow duplicate gene names. If None, duplicates are allowed for dense input but not for sparse input.

Returns:

Return type:

scprep.io.load_10X_HDF5(*filename*, *genome=None*, *sparse=True*, *gene_labels='symbol'*, *allow_duplicates=None*, *backend=None*) [source]

Basic IO for HDF5 10X data produced from the 10X Cellranger pipeline.

Installation Examples Reference Data Input/Output HDF5 Download Filtering Normalization Transformation Measurements Statistics Plotting Dimensionality Reduction Row/Column Selection Utilities External Tools

The POWERFUL PYTHON PLAYBOOK for intermediate+ Python. Download free here

Sponsored · Ads served ethically

Read the Docs v: stable ▾

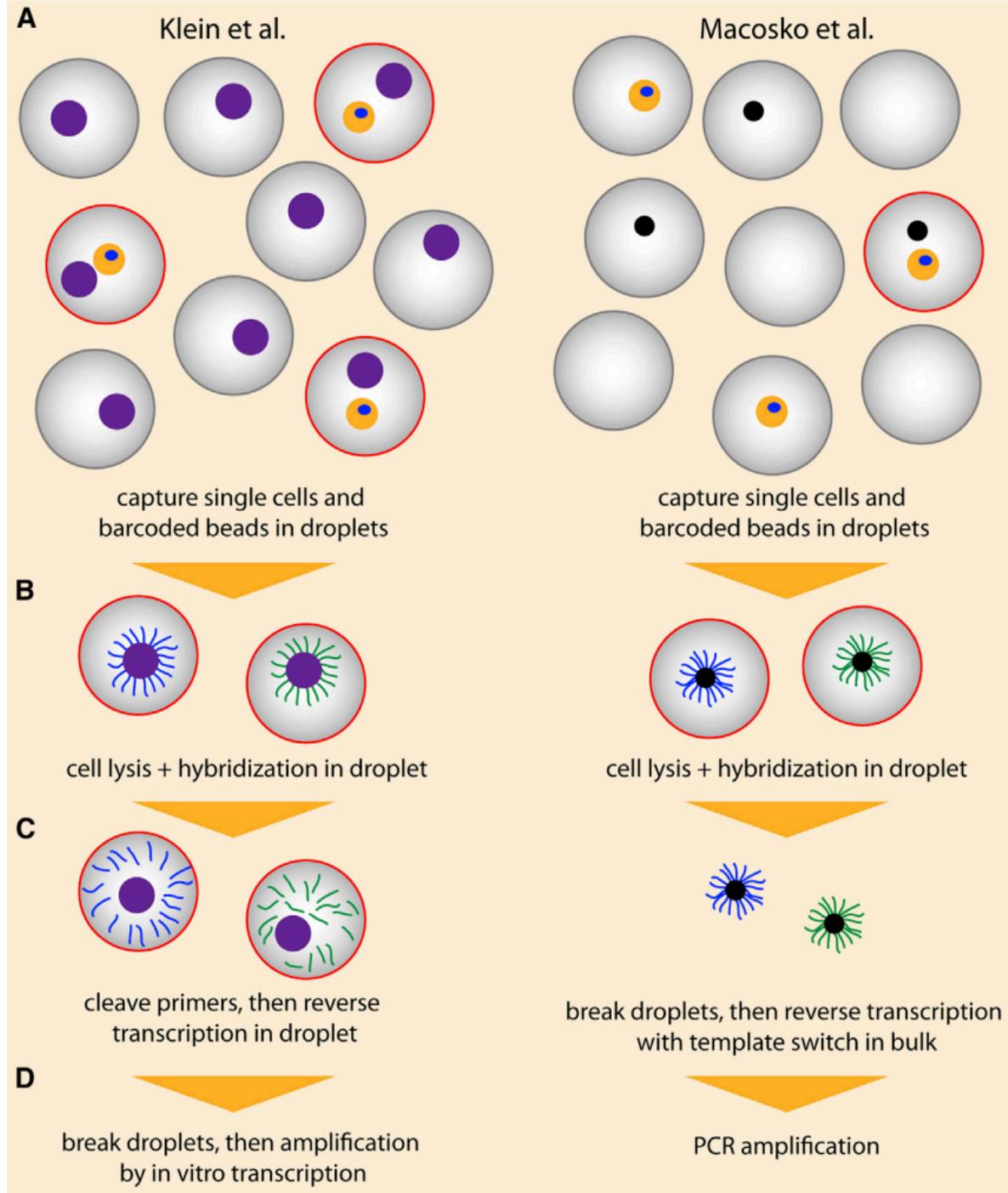
Bring-your-own-data workshop



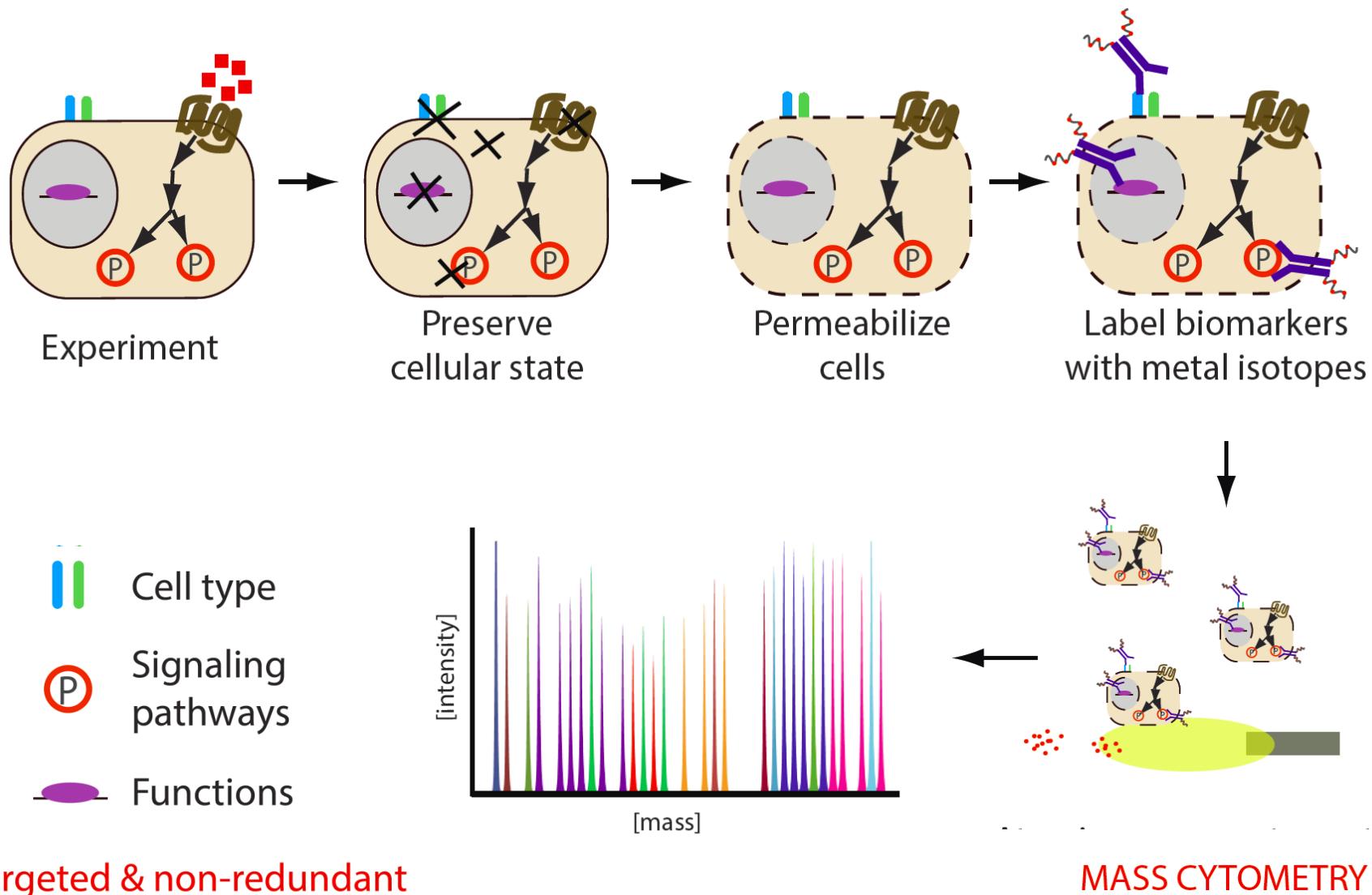
#2020-workshop-byod-help
<https://krishnaswamylab.org/get-help>

Challenges and Opportunities in Single Cell Data

Droplet-based Technologies



Single-Cell Proteomics: Mass Cytometry



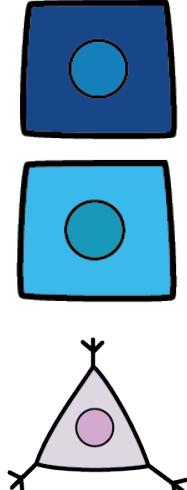
Single Cell Data

- Each cell is a vector of measurements
 - e.g. Cell A = [40 0 20 18 5 0 ...]
- The whole data is a matrix with many observations (cells) and features (proteins, genes)

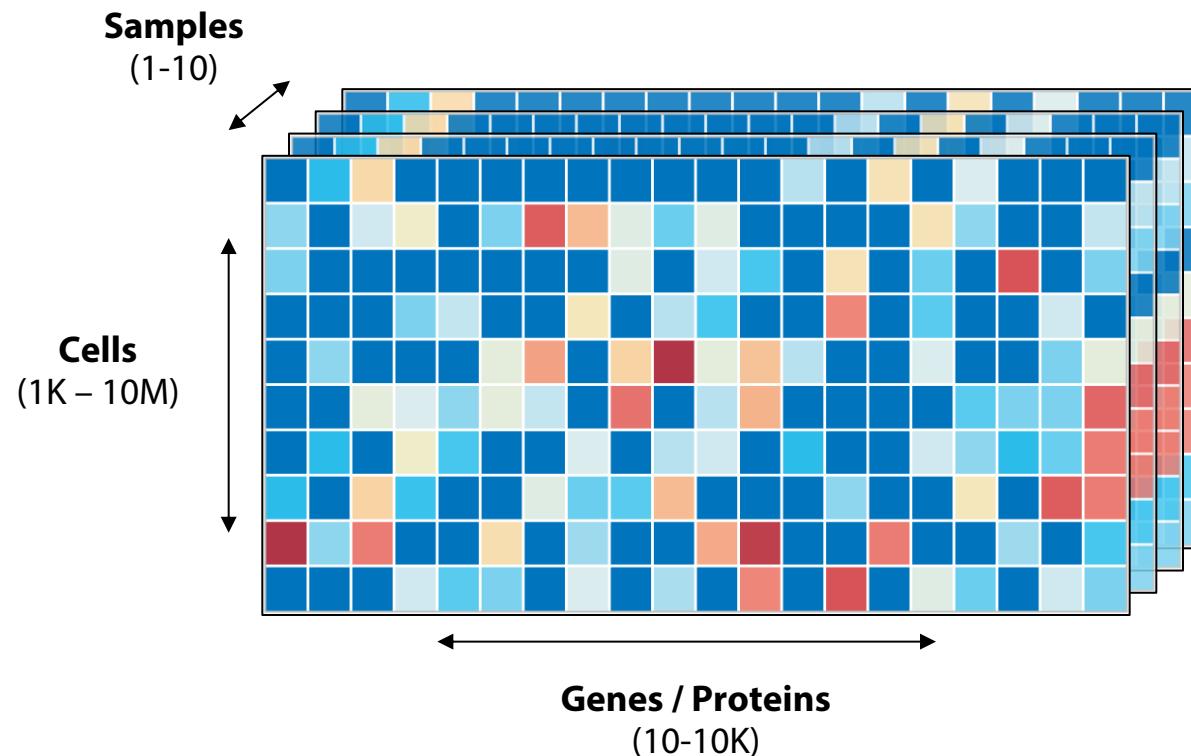
Features
(e.g. genes)

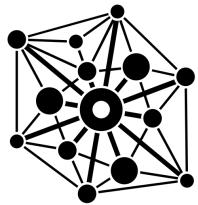
	X	Y	Z
A	10	20	70
B	20	40	140
C	20	0	80

Observations
(e.g. cells)

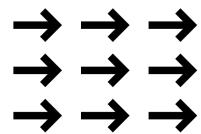


Single Cell Data





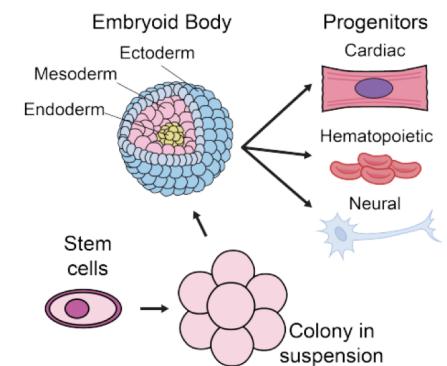
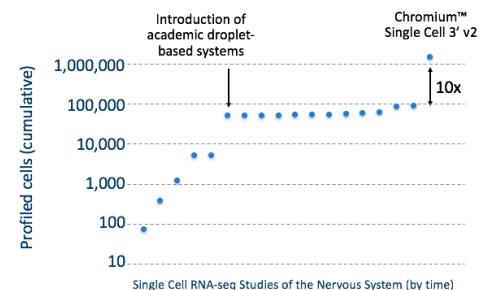
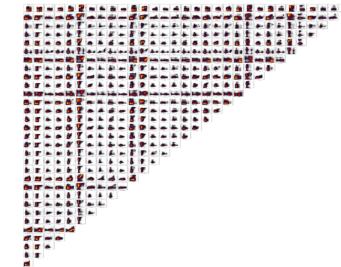
High Dimensional



High Throughput



Heterogeneous



Many dimensions = many measurements



Diagnoses



labs



drug response assays



ECG



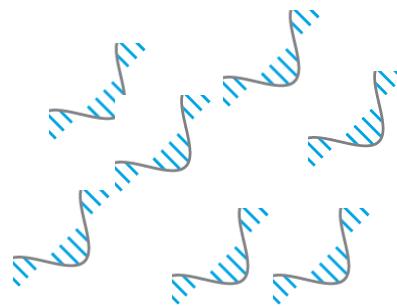
Gene 1



Gene 2



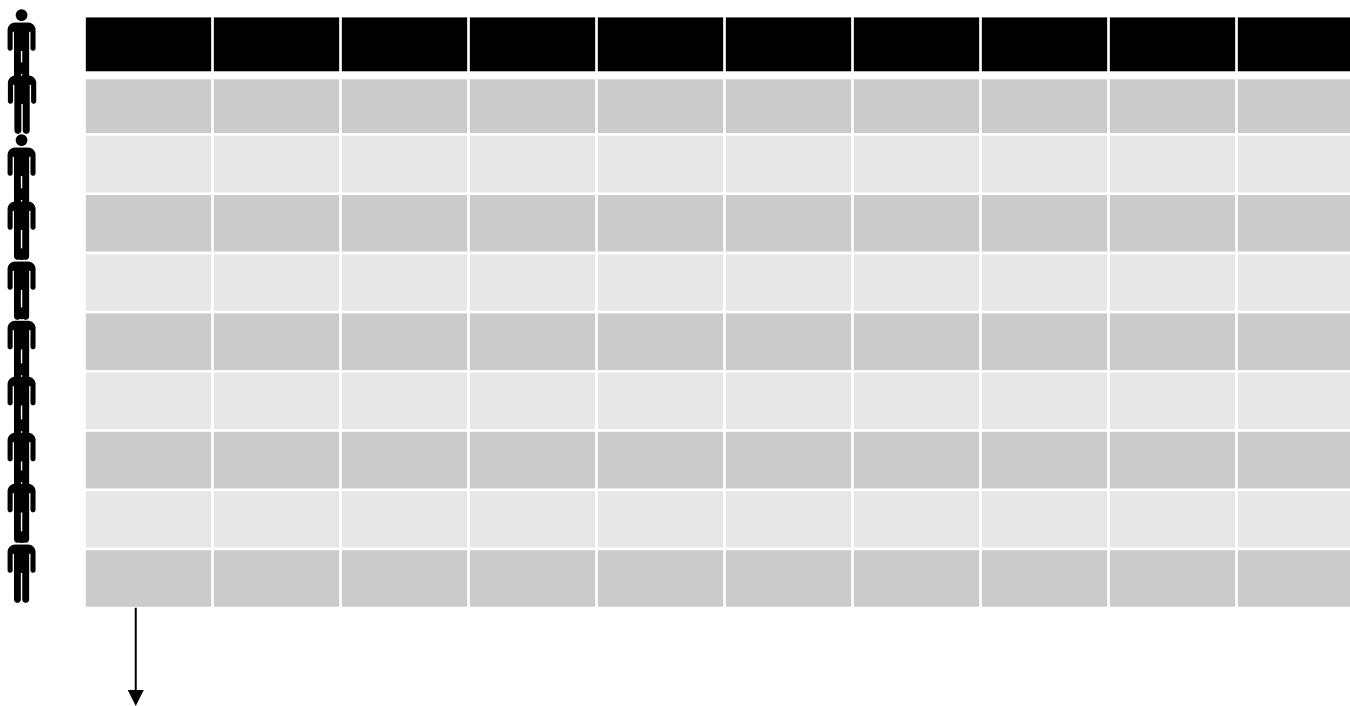
Gene 3



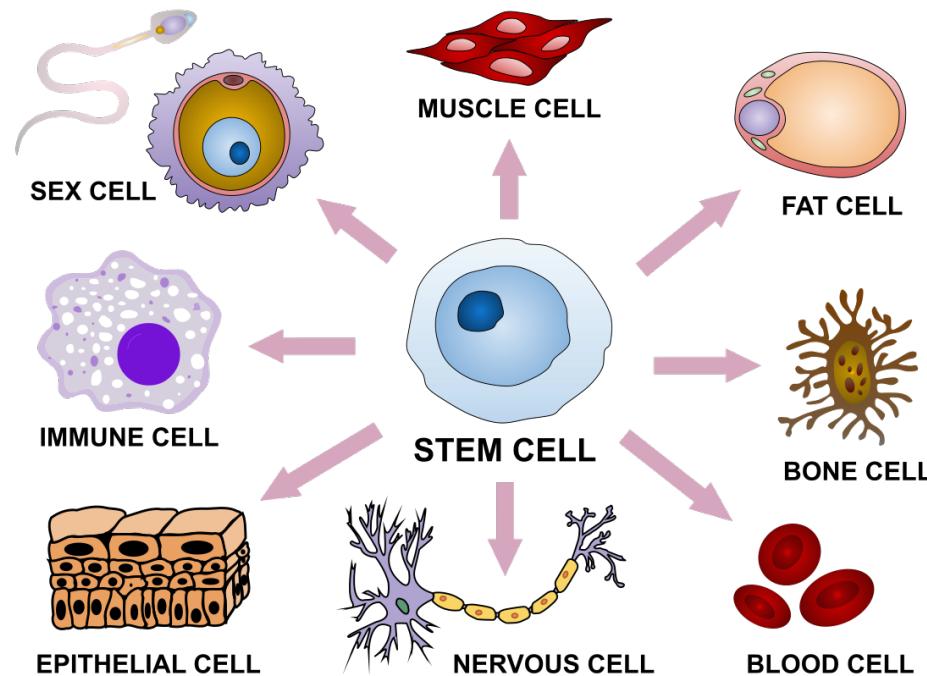
Proteins



High Throughput = Many observations

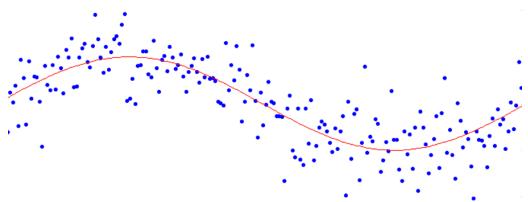


Heterogeneous Observations

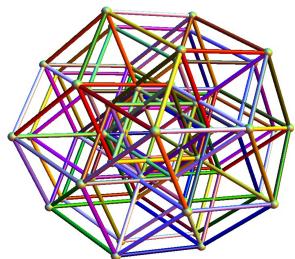
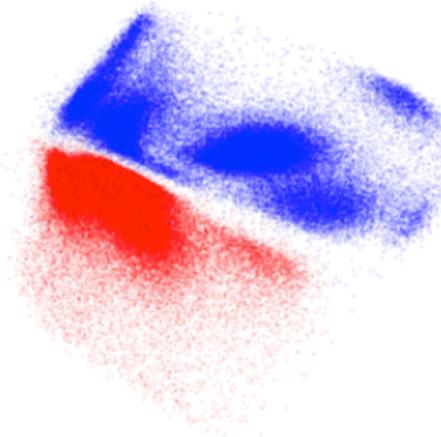


Challenges

Noise



Batch Effects

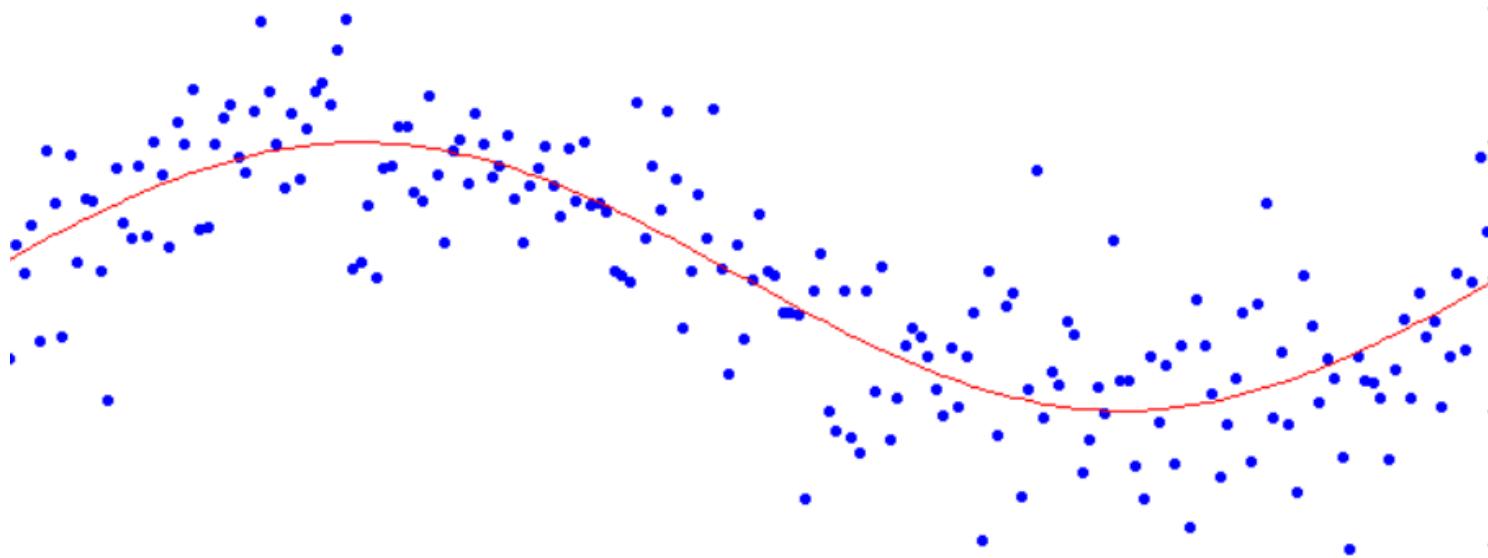


Dimensionality

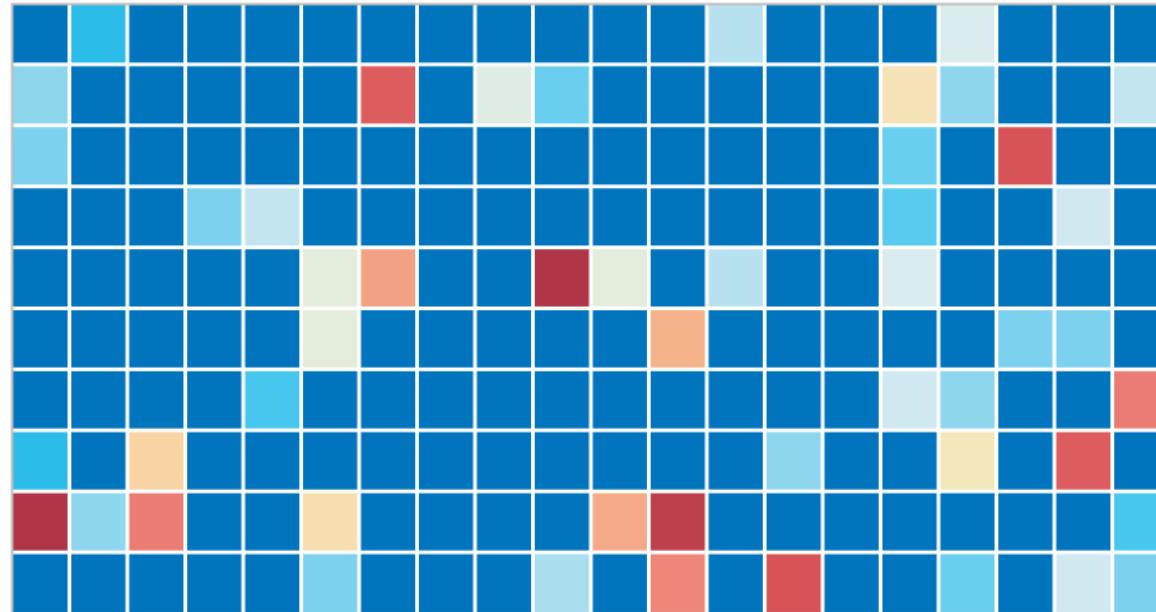


Scale

Noise



Dropout

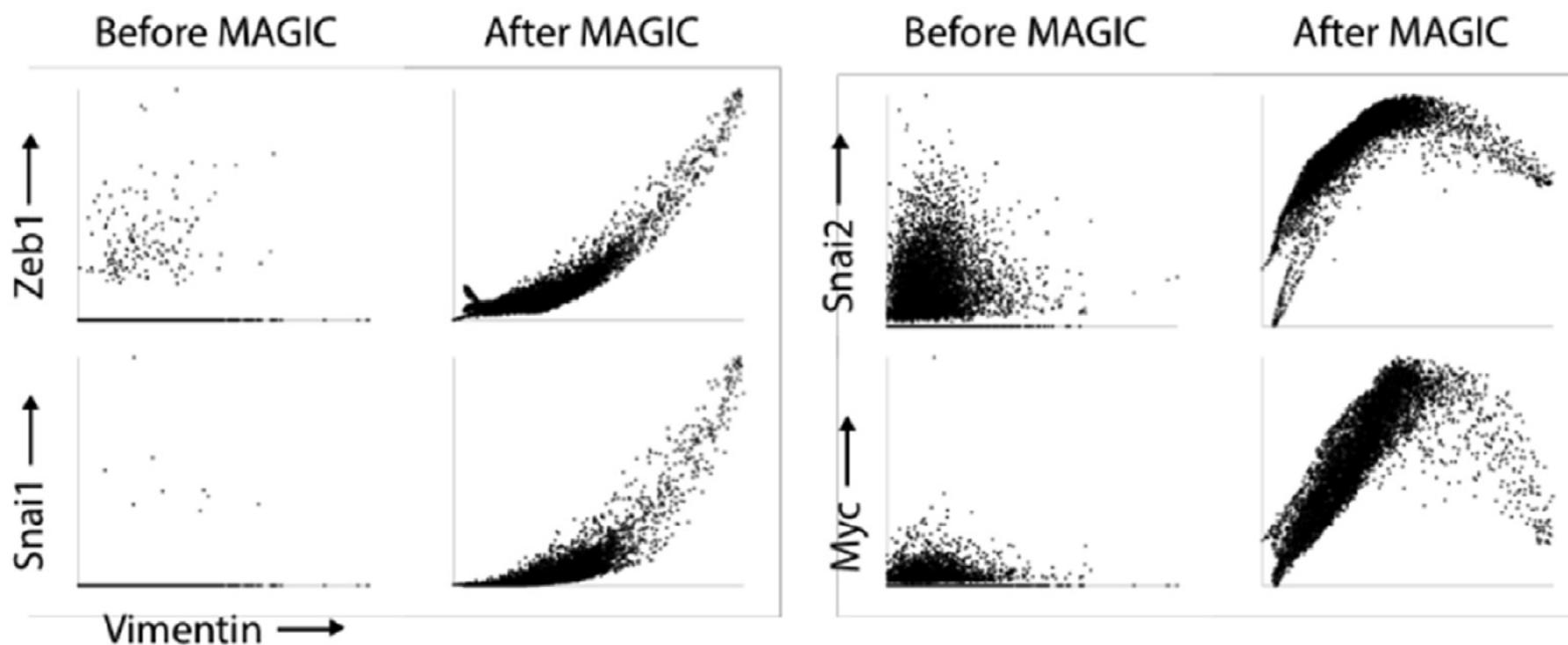


Dropout vs Missing Data

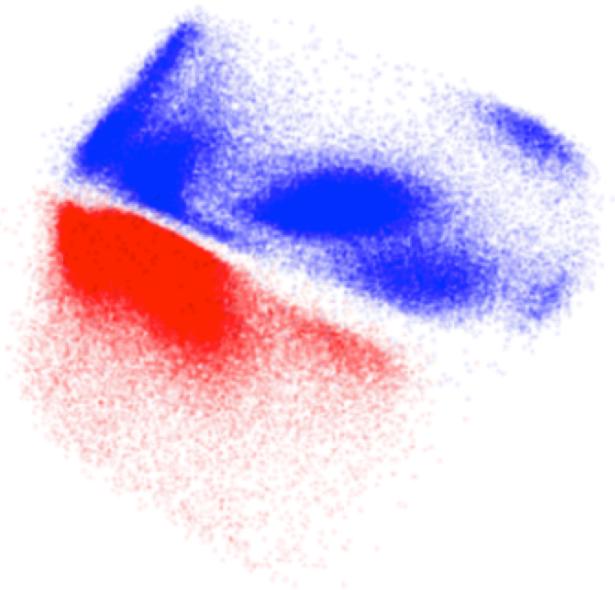
Missing values

PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
1	0	3	male	22	1	0	A/5 21171	7.25		S
2	1	1	female	38	1	0	PC 17599	71.2033	C85	C
3	1	3	female	26	0	0	STON/O2. 3101282	7.925		S
4	1	1	female	35	1	0	113803	53.1	C123	S
5	0	3	male	35	0	0	373450	8.05		S
6	0	3	male		0	0	330877	8.4583		Q

Denoising Data

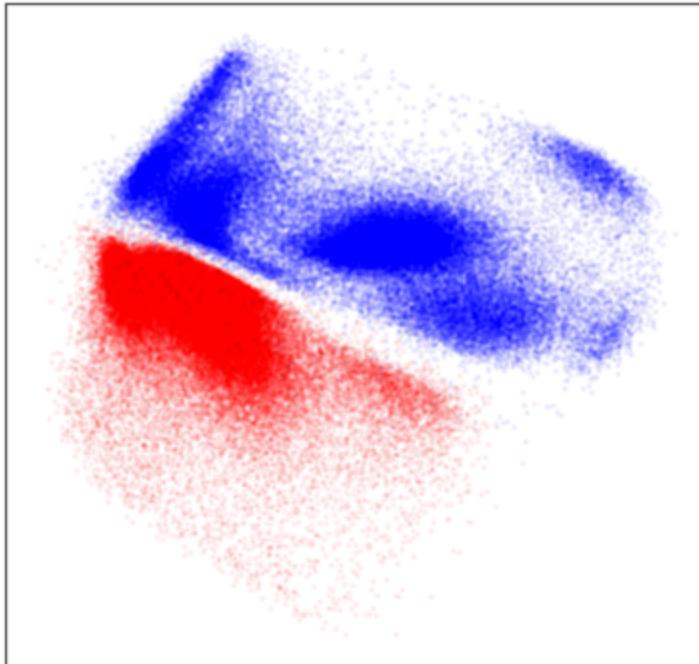


Batch Effects

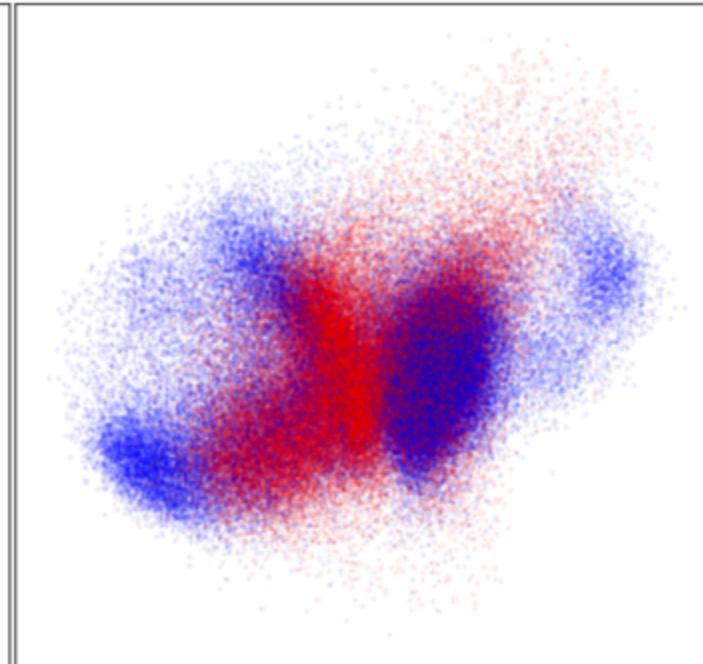


- Systematic differences between samples due to machine calibration, ambient environmental effects
- Variation that is uninteresting to examine and confounds biological variation
- Renders samples uncomparable

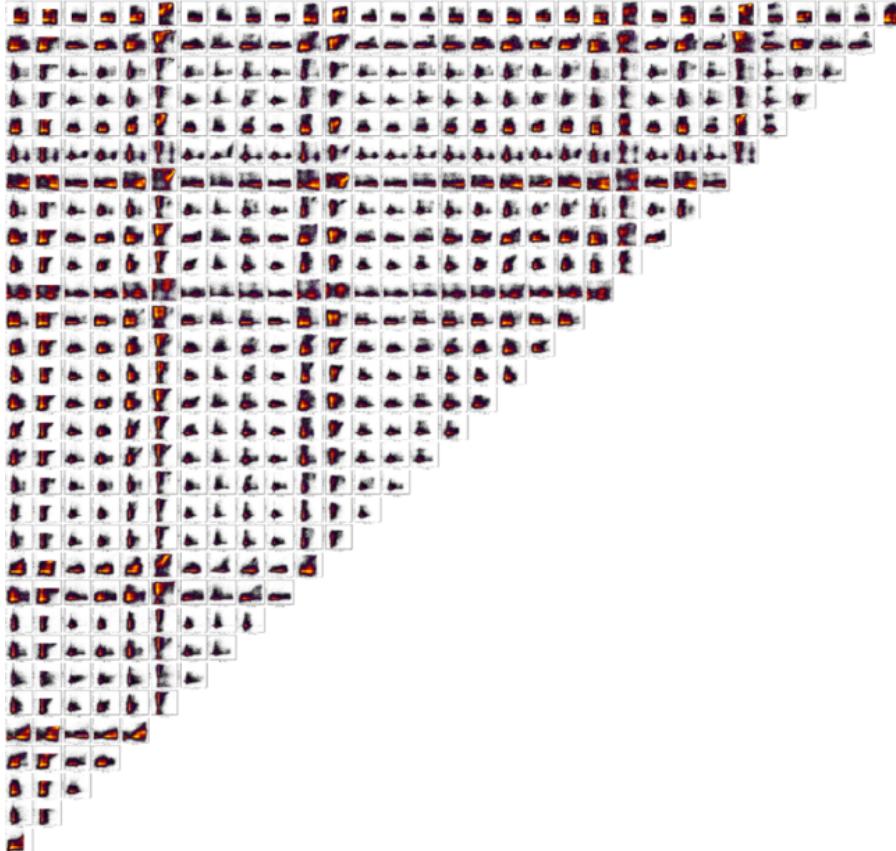
Before MMD



After MMD

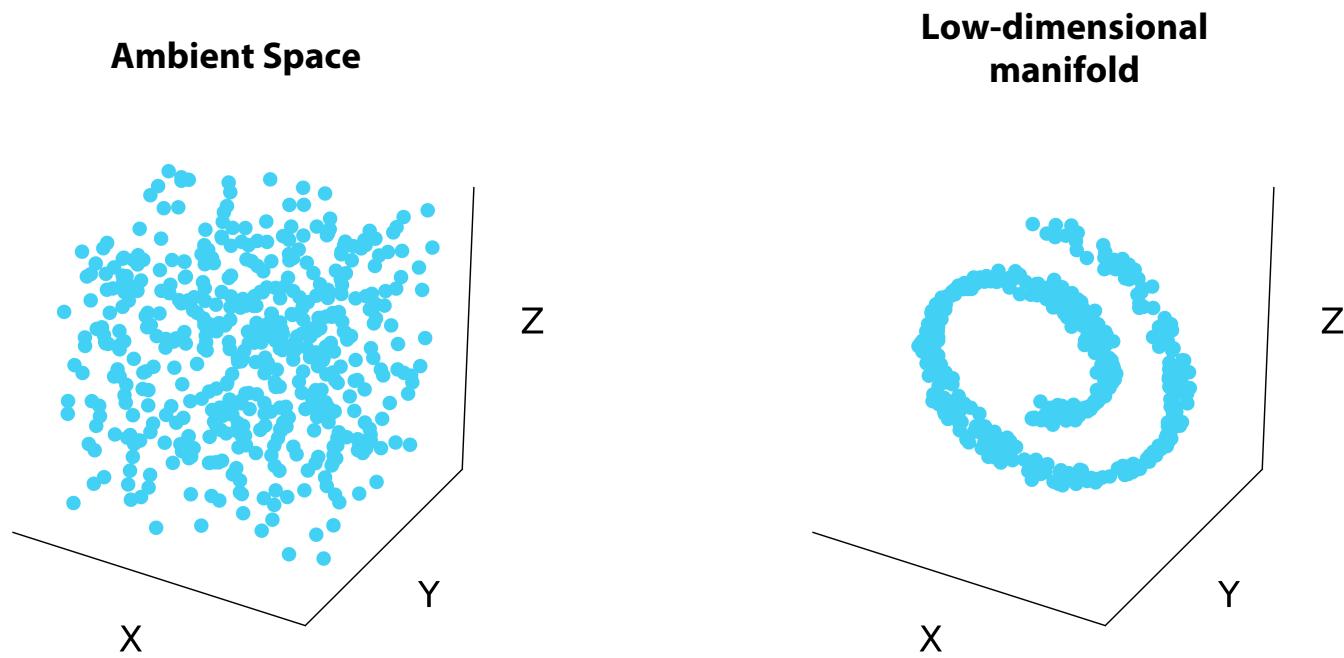


High Dimensionality

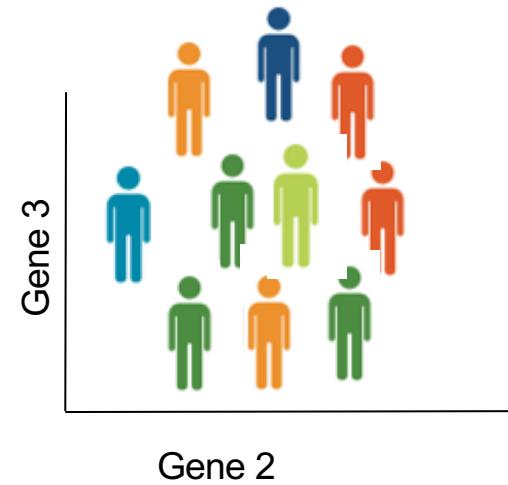
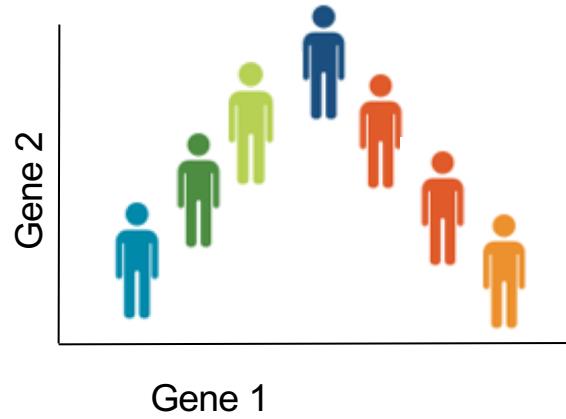


- Difficult to understand the overall shape of the data
- Number of subgroups, separations, progression

Latent structure in high dimensional data

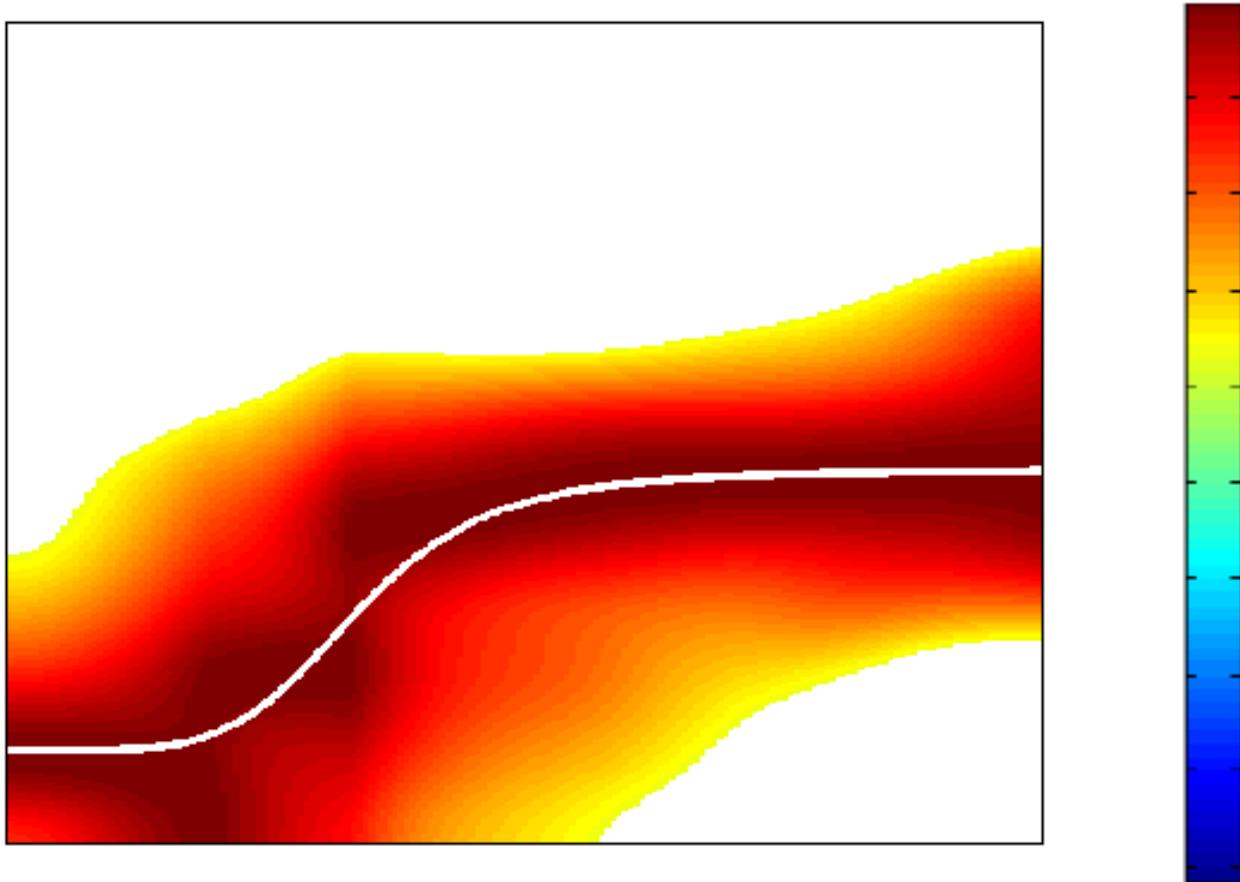
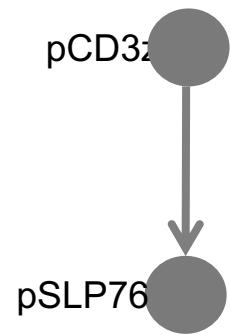


Gene-gene Relationships

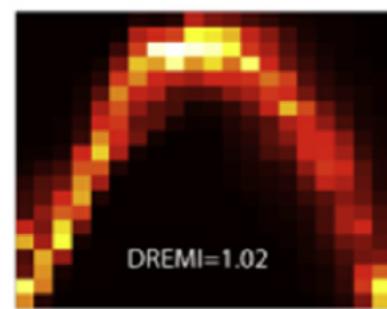
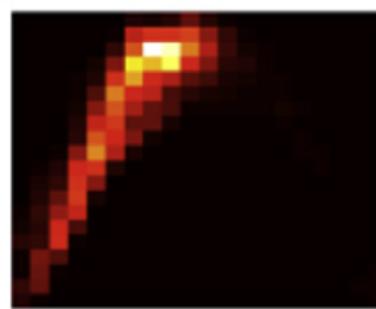
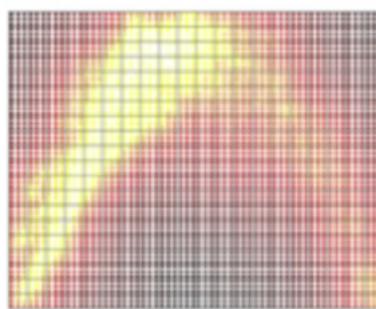
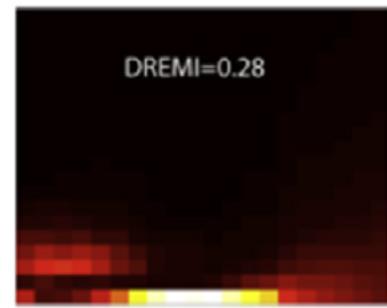
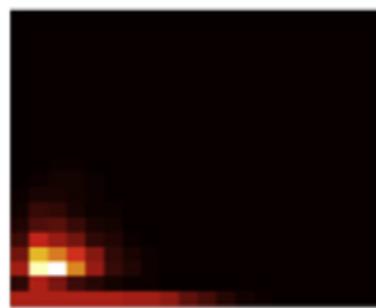
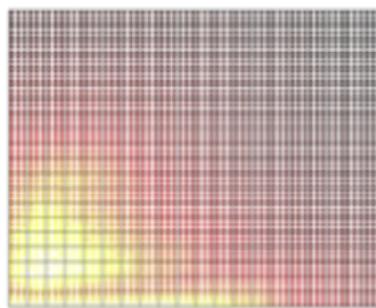
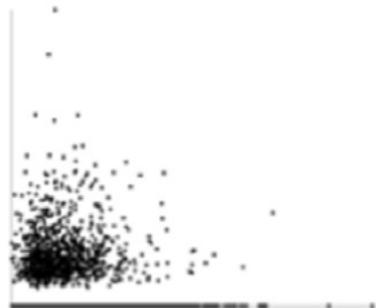


Relationship between features?

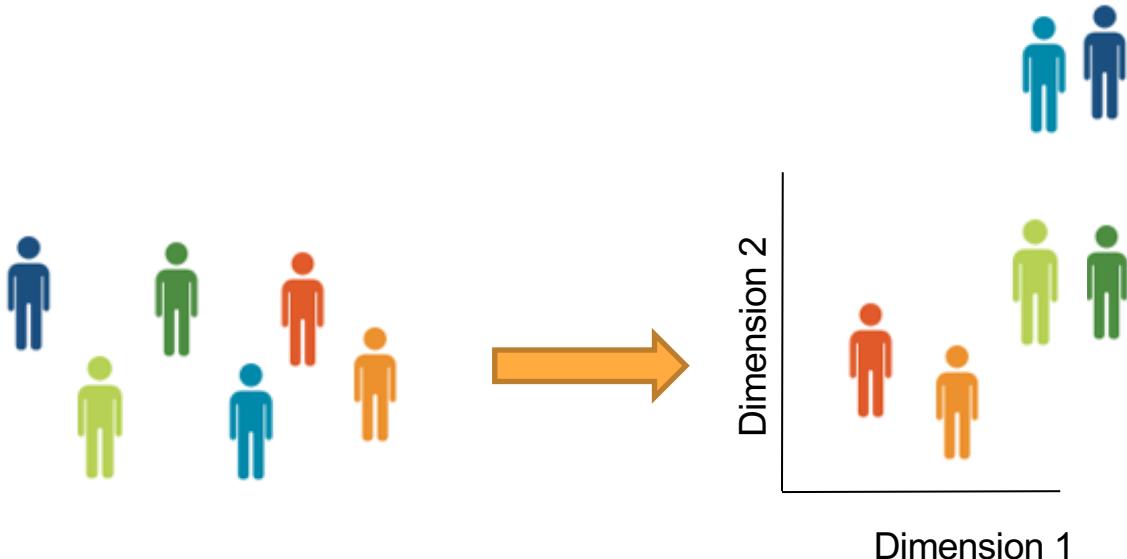
Non-linear Relationships



Mutual Information



Embedding reveals structure



Use high dimensional features and high throughput to understand shape of data

Cluster structure



Excellent
Responders

Group 1



Fair
Responde
rs

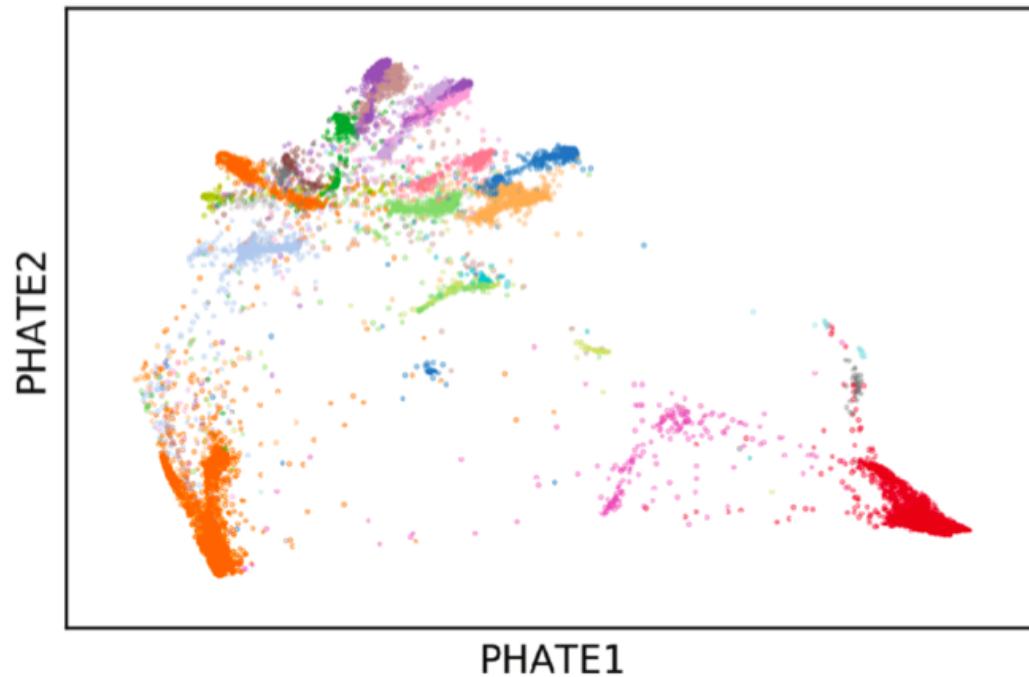
Group 2



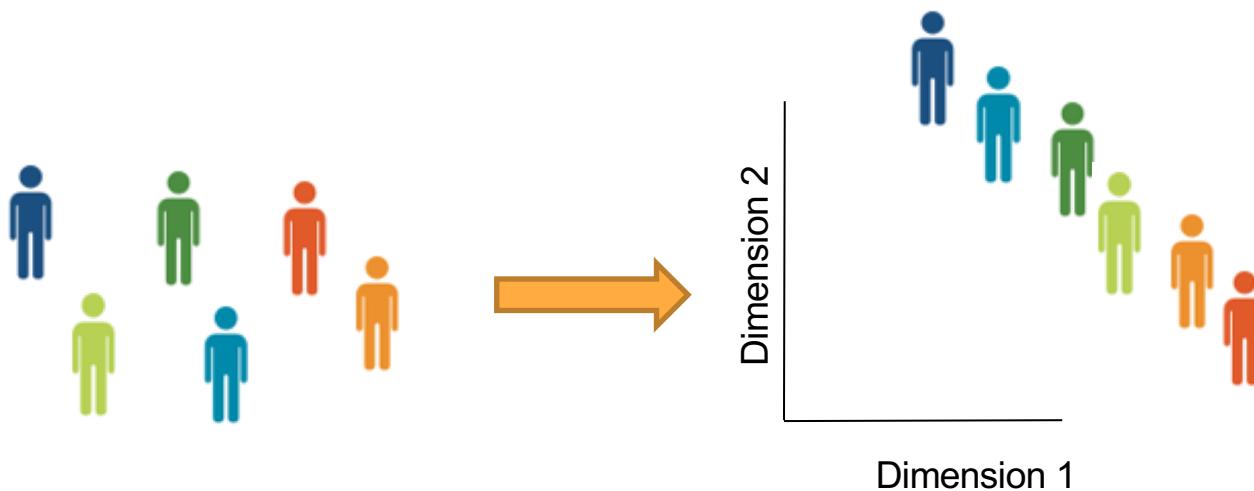
Poor
responde
rs

Group 3

Retinal Bipolar Cells



Progression continuum

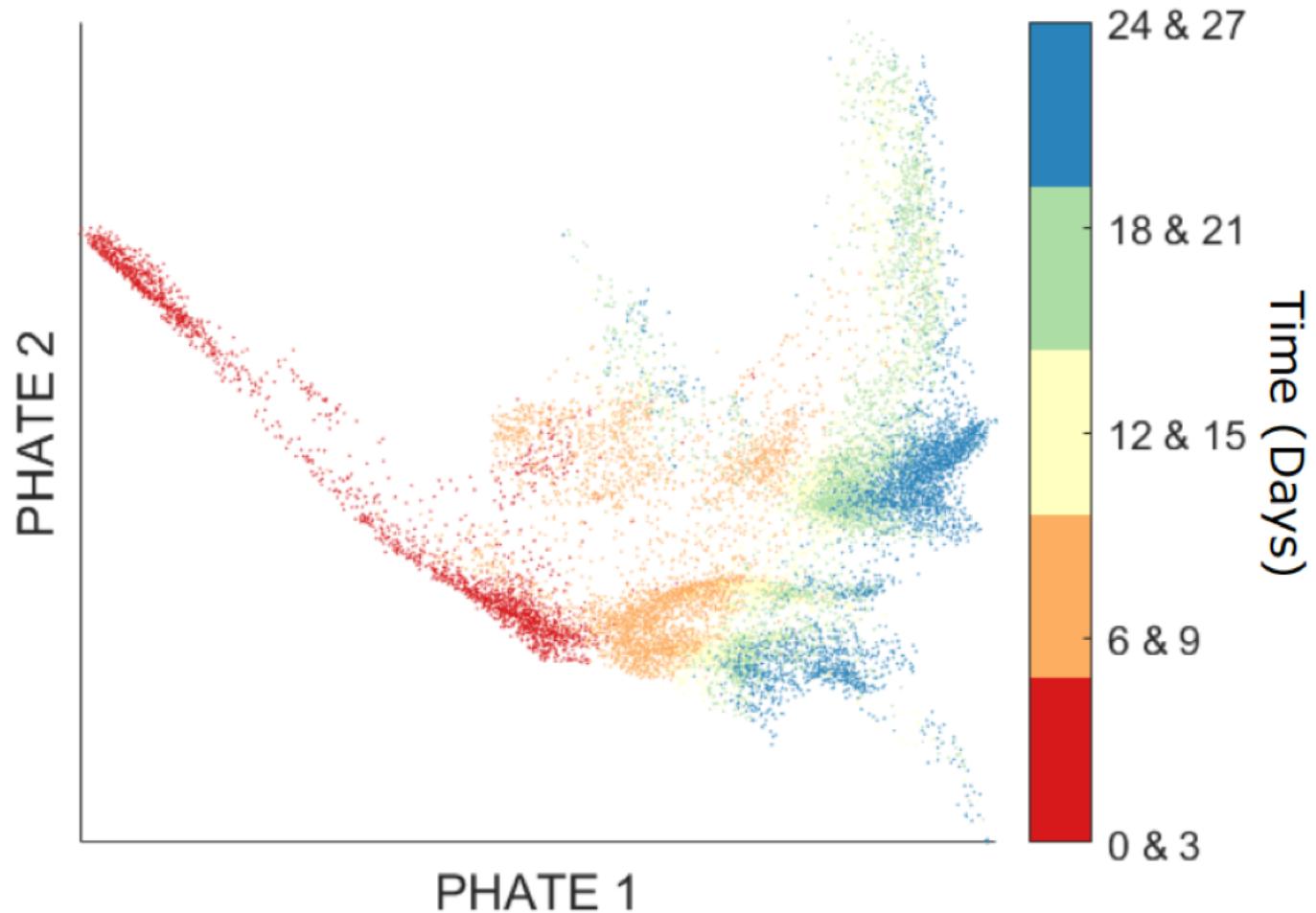


Use high dimensional features and high throughput to understand shape of data

Pseudotime

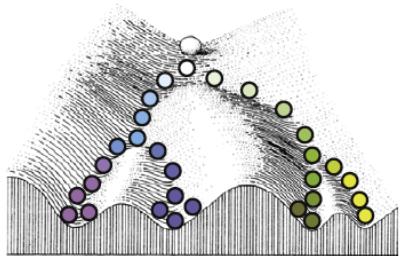


Progressions

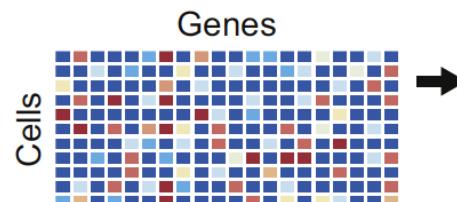


Manifold Learning

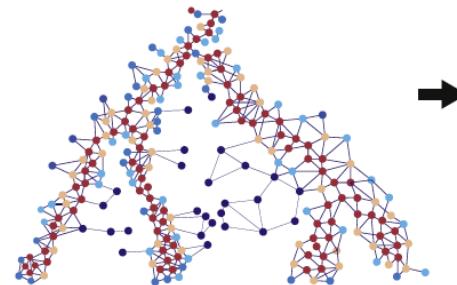
Cells are sampled from an underlying manifold



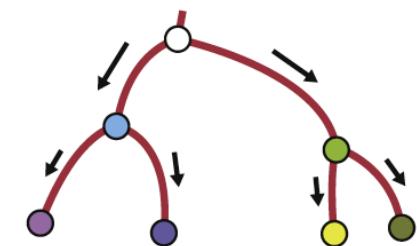
Each cell is represented by a vector of gene expression



Neighborhood structure of the observations is identified

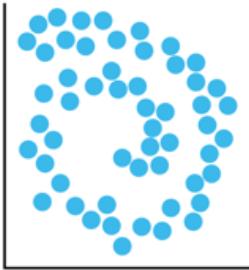


The latent manifold is learned from the data

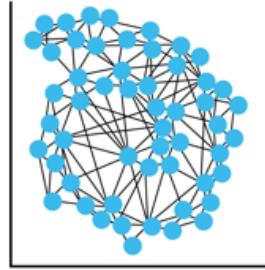


Understanding the shape of data

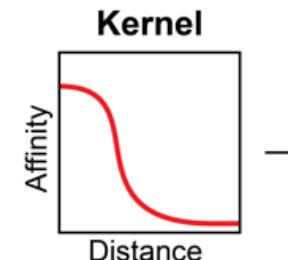
Data in two dimensions



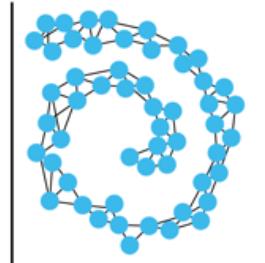
Distances between all points are calculated



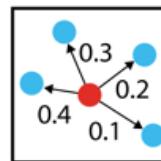
A kernel function calculates affinities from distance



Only local relationships are preserved

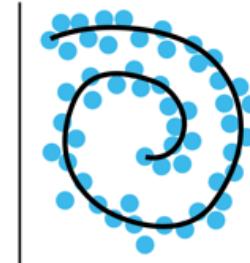


Diffusion shares information between nodes



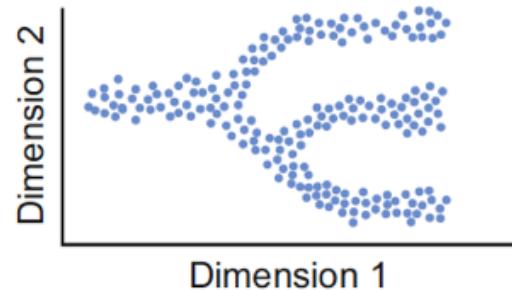
Diffusion distance
≈
Random walk dist.

Underlying manifold is calculated

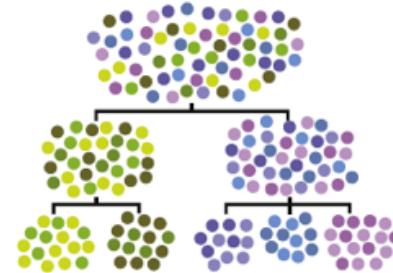


Analysis Tasks

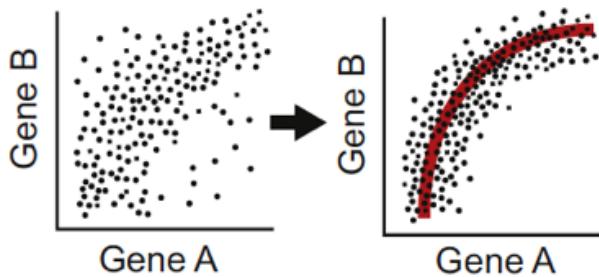
Vizualization



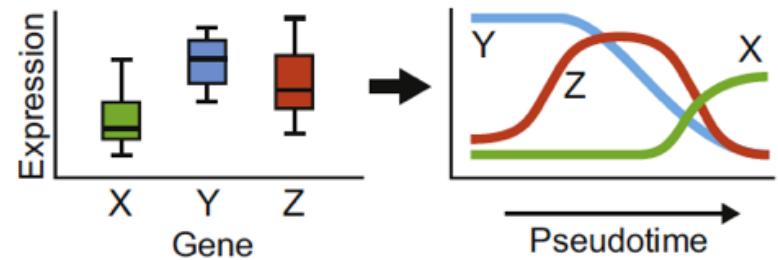
Clustering



Denoising



Pseudotime analysis



Preprocessing single-cell data

Current best practices in single-cell RNA-seq analysis: a tutorial

Malte D Luecken¹  & Fabian J Theis^{1,2,*} 

Abstract

Single-cell RNA-seq has enabled gene expression to be studied at an unprecedented resolution. The promise of this technology is attracting a growing user base for single-cell analysis methods. As more analysis tools are becoming available, it is becoming increasingly difficult to navigate this landscape and produce an up-to-date workflow to analyse one's data. Here, we detail the steps of a typical single-cell RNA-seq analysis, including pre-processing (quality control, normalization, data correction, feature selection, and dimensionality reduction) and cell- and gene-level downstream analysis. We formulate current best-practice recommendations for these steps based on independent comparison studies. We have integrated these best-practice recommendations into a workflow, which we apply to a public dataset to further illustrate how these steps work in practice. Our documented case study can be found at <https://www.github.com/theislab/single-cell-tutorial>. This review will serve as a workflow tutorial for new entrants into the field, and help established users update their analysis pipelines.

Keywords analysis pipeline development; computational biology; data analysis tutorial; single-cell RNA-seq

DOI 10.15252/msb.20188746 | Received 16 November 2018 | Revised 15 March 2019 | Accepted 3 April 2019

Mol Syst Biol. (2019) 15: e8746

Introduction

In recent years, single-cell RNA sequencing (scRNA-seq) has significantly advanced our knowledge of biological systems. We have been able to both study the cellular heterogeneity of zebrafish, frogs

outline current best practices to lay a foundation for future analysis standardization.

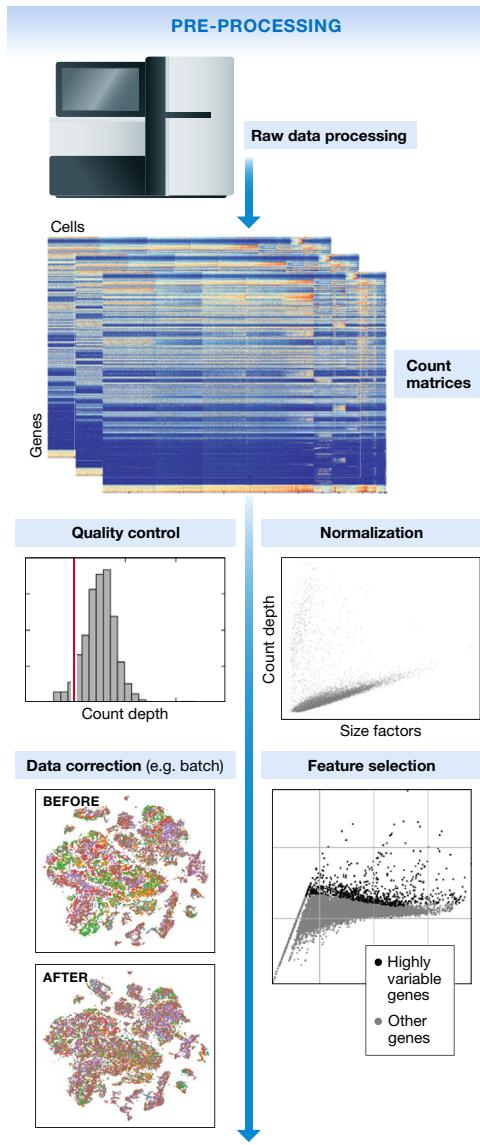
The challenges to standardization include the growing number of analysis methods (385 tools as of 7 March 2019) and exploding dataset sizes (Angerer *et al.*, 2017; Zappia *et al.*, 2018). We are continuously finding new ways to use the data at our disposal. For example, it has recently become possible to predict cell fates in differentiation (La Manno *et al.*, 2018). While the continuous improvement of analysis tools is beneficial for generating new scientific insight, it complicates standardization.

Further challenges for standardization lie in technical aspects. Analysis tools for scRNA-seq data are written in a variety of programming languages—most prominently R and Python (Zappia *et al.*, 2018). Although cross-environment support is growing (preprint: Scholz *et al.*, 2018), the choice of programming language is often also a choice between analysis tools. Popular platforms such as Seurat (Butler *et al.*, 2018), Scater (McCarthy *et al.*, 2017), or Scanpy (Wolf *et al.*, 2018) provide integrated environments to develop pipelines and contain large analysis toolboxes. However, out of necessity these platforms limit themselves to tools developed in their respective programming languages. By extension, language restrictions also hold true for currently available scRNA-seq analysis tutorials, many of which revolve around the above platforms (R and bioconductor tools: <https://github.com/drissi/bioc2016singlecell> and <https://hemberg-lab.github.io/scRNA.seq.course/>; Lun *et al.*, 2016b; Seurat: https://satijalab.org/seurat/get_started.html; Scanpy: <https://scanpy.readthedocs.io/en/stable/tutorials.html>).

Considering the above-mentioned challenges, instead of targeting a standardized analysis pipeline, we outline current best practices and common tools independent of programming language. We guide the reader through the various steps of a scRNA-seq analysis pipeline (Fig 1), present current best practices, and discuss analysis

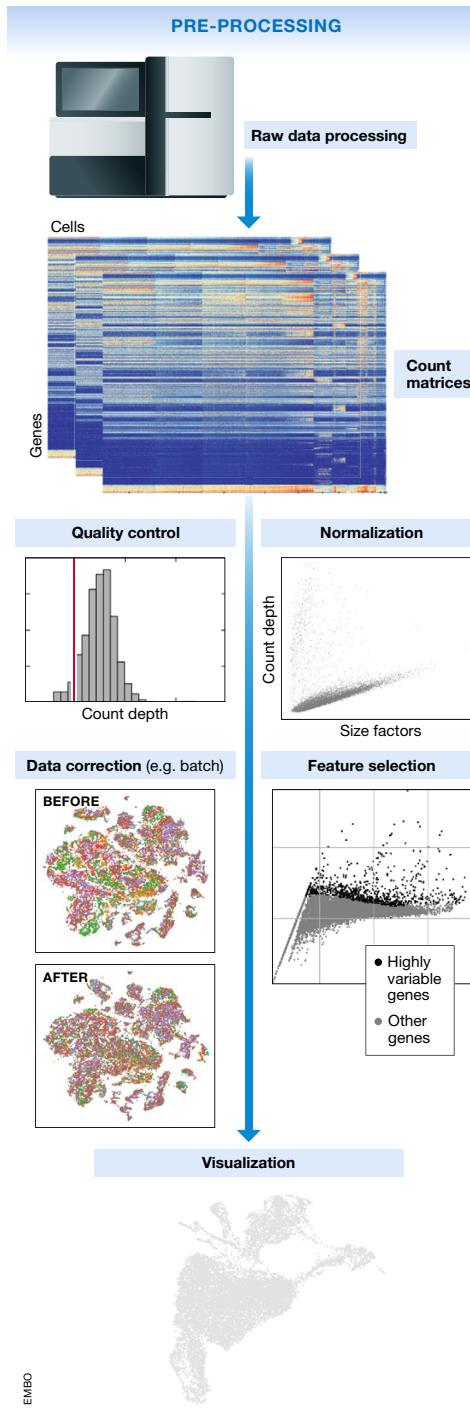
Standard Single-Cell RNA-seq Workflow

1. Sequencing and read mapping
2. Quality control and filtering
3. Normalization
4. Data Correction



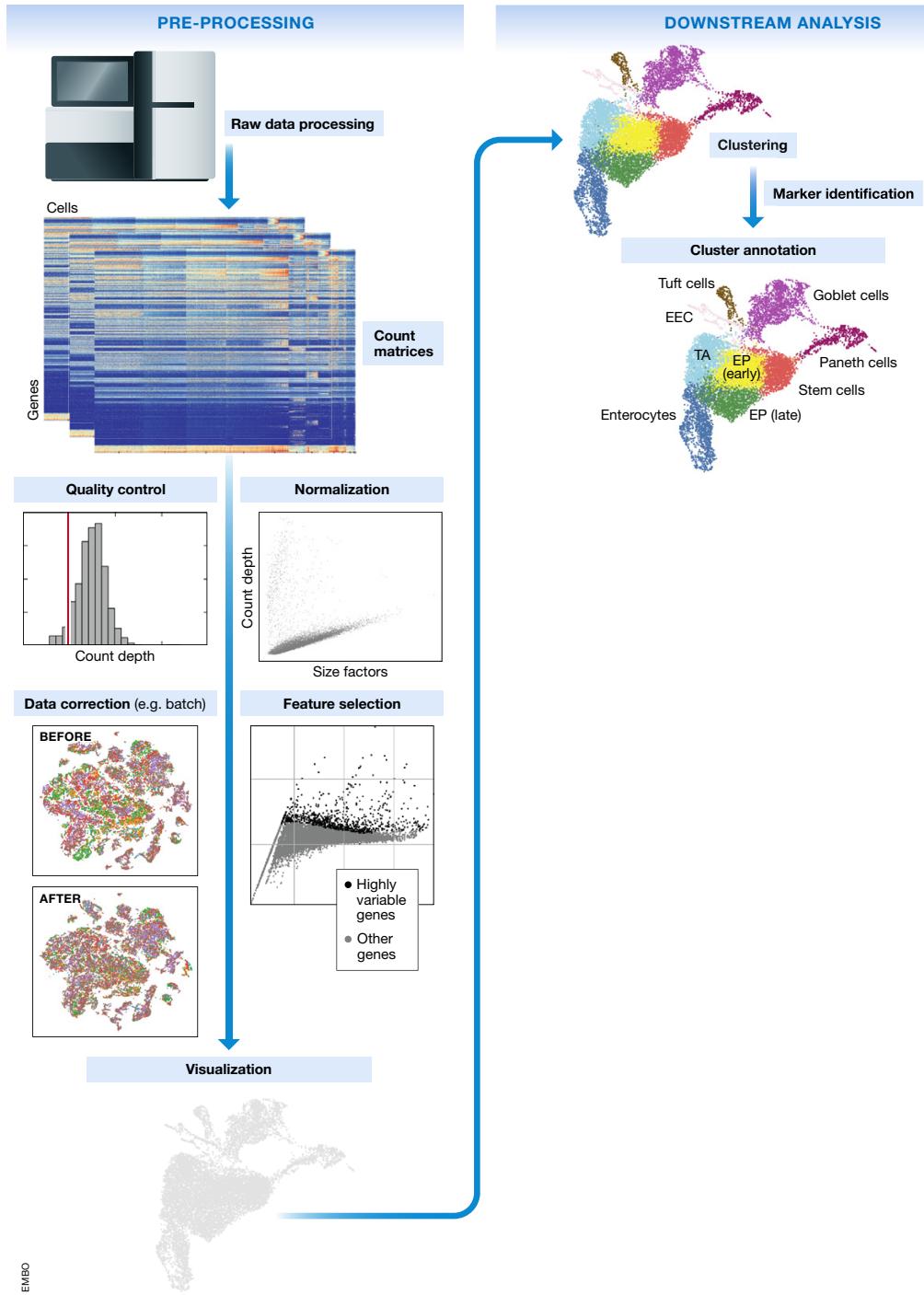
Standard Single-Cell RNA-seq Workflow

1. Sequencing and read mapping
2. Quality control and filtering
3. Normalization
4. Data Correction
5. Dimensionality reduction and visualization



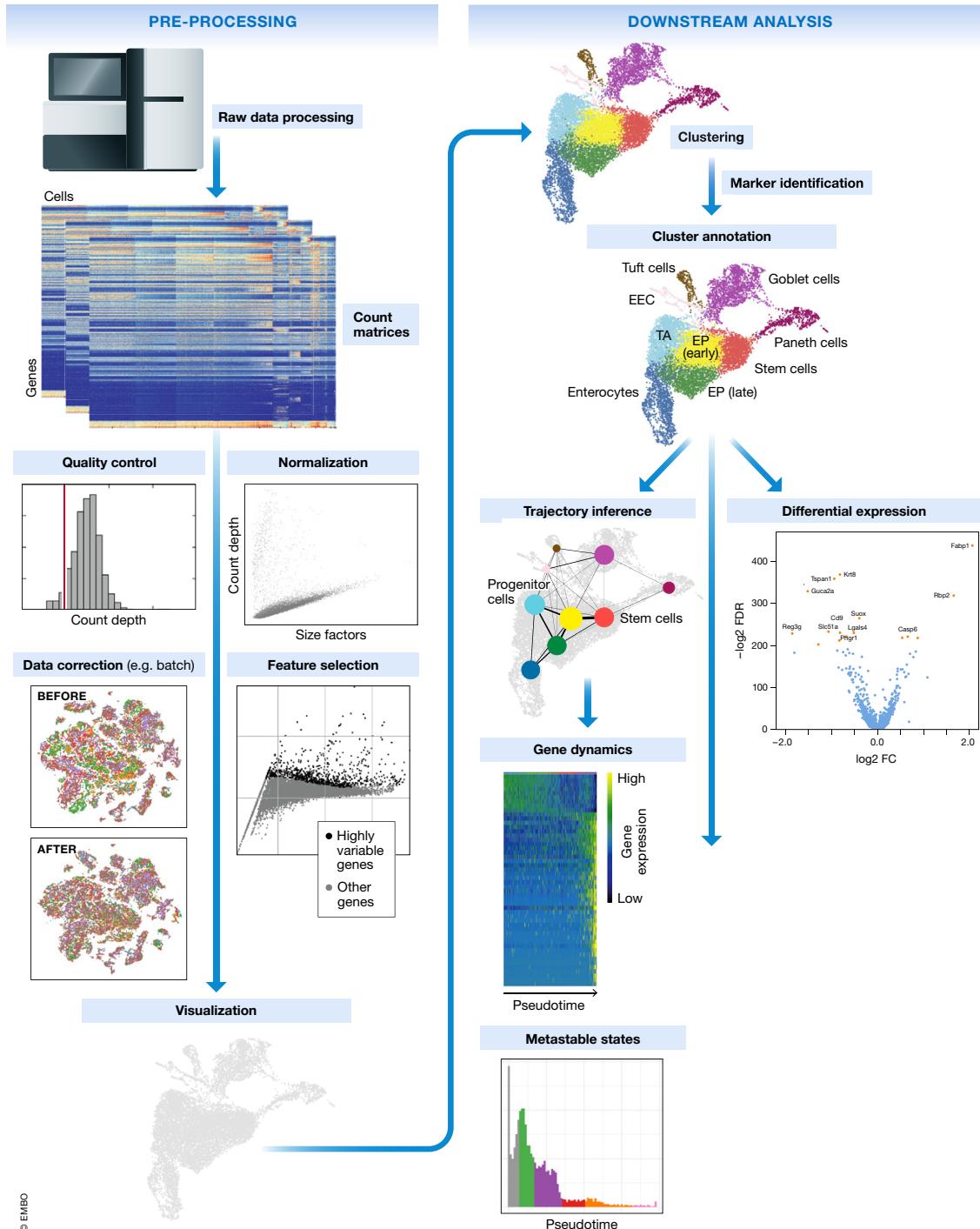
Standard Single-Cell RNA-seq Workflow

1. Sequencing and read mapping
2. Quality control and filtering
3. Normalization
4. Data Correction
5. Dimensionality reduction and visualization
6. Downstream analysis
 1. Clustering
 2. Trajectory inference



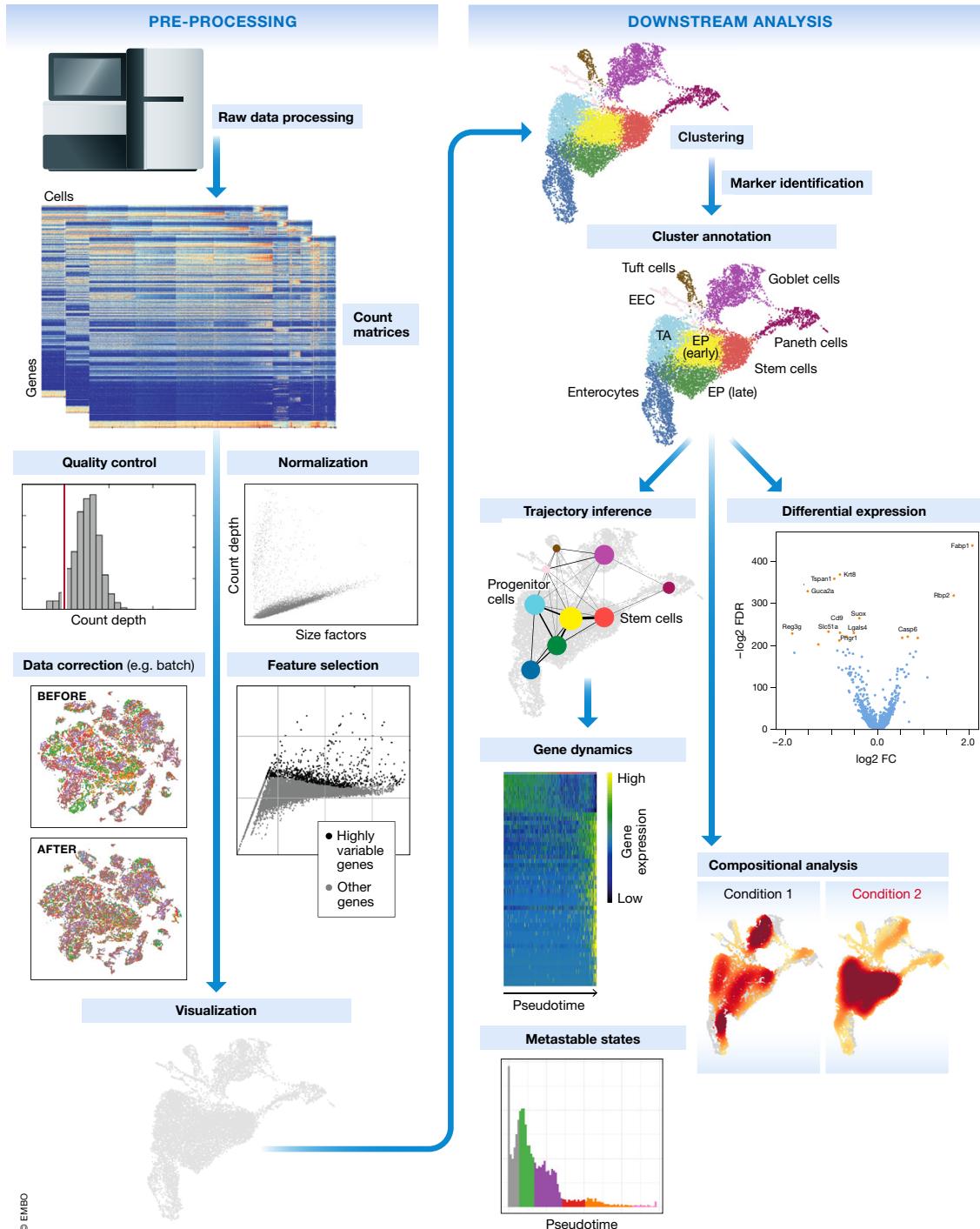
Standard Single-Cell RNA-seq Workflow

1. Sequencing and read mapping
2. Quality control and filtering
3. Normalization
4. Data Correction
5. Dimensionality reduction and visualization
6. Downstream analysis
 1. Clustering
 2. Trajectory inference
 3. Differential expression

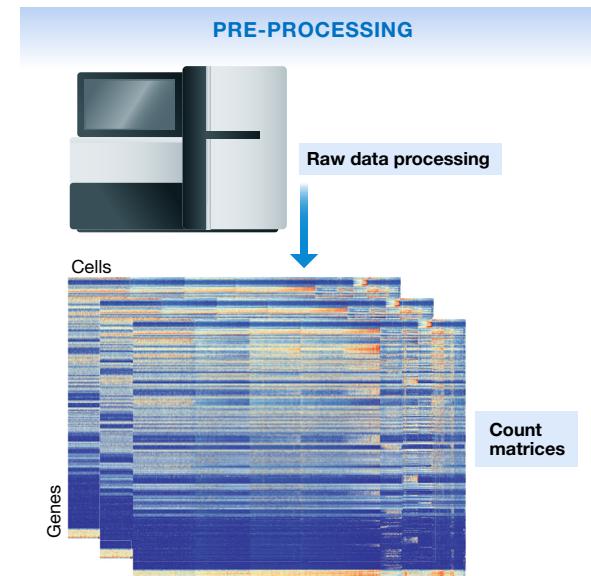
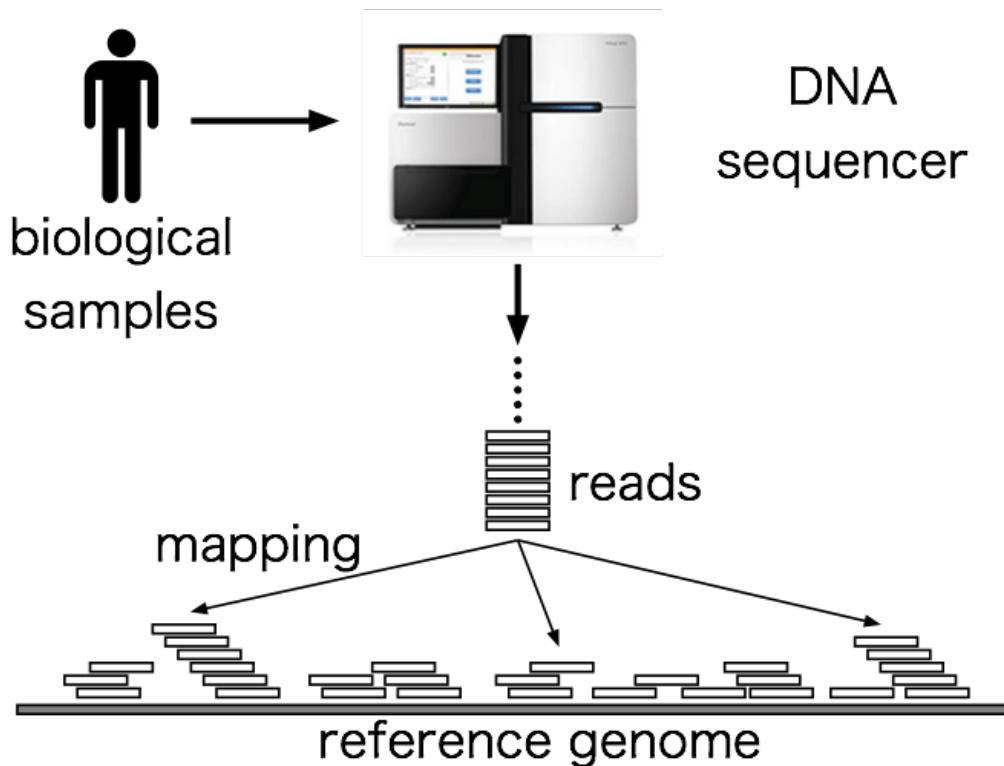


Standard Single-Cell RNA-seq Workflow

1. Sequencing and read mapping
2. Quality control and filtering
3. Normalization
4. Data Correction
5. Dimensionality reduction and visualization
6. Downstream analysis
 1. Clustering
 2. Trajectory inference
 3. Differential expression
7. Comparison of multiple conditions



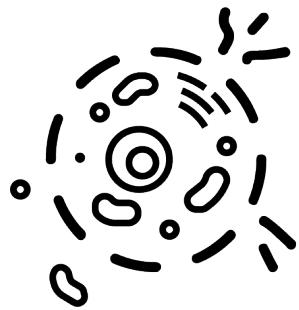
Step 1 - Sequencing & read mapping



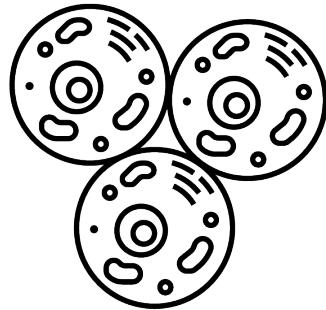
Building Counts Matrix → QC & Filtering → Normalization → Visualization → Analysis

Step 2 – Quality control and filtering

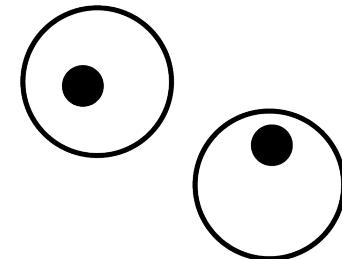
Dying cells



Multiplets



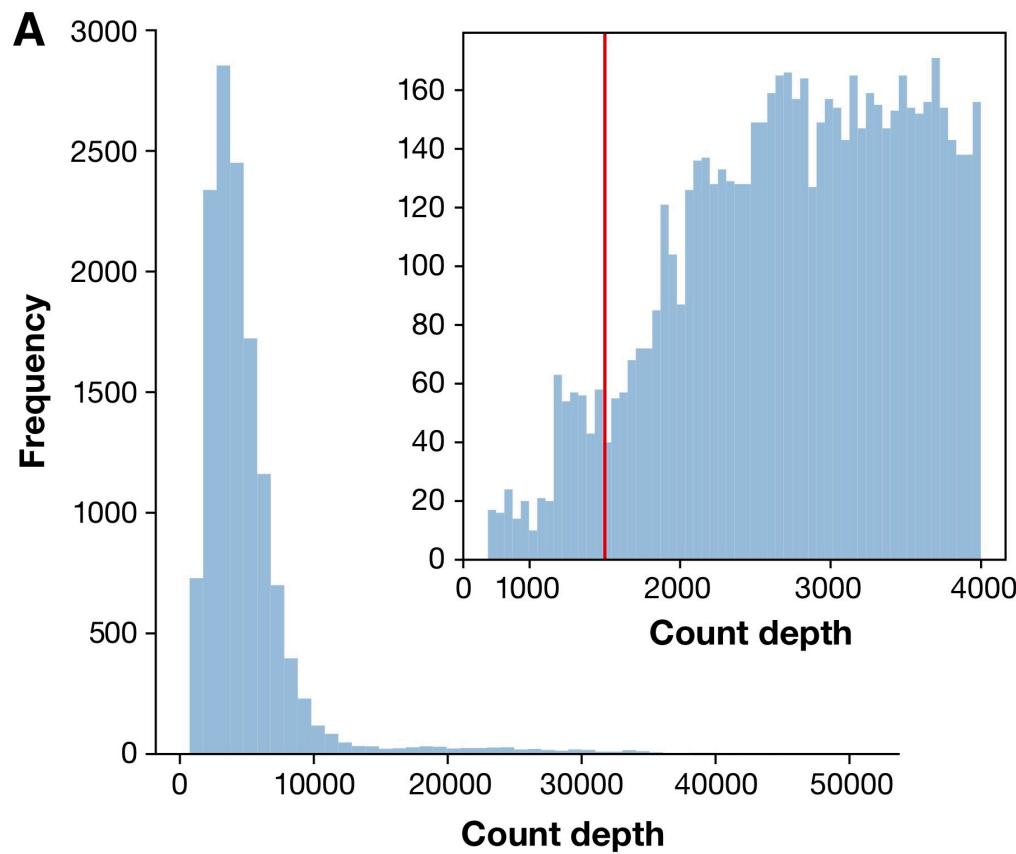
Empty Droplets



What could we look at to discriminate between dying cells, multiplets, or empty droplets and healthy single cells?

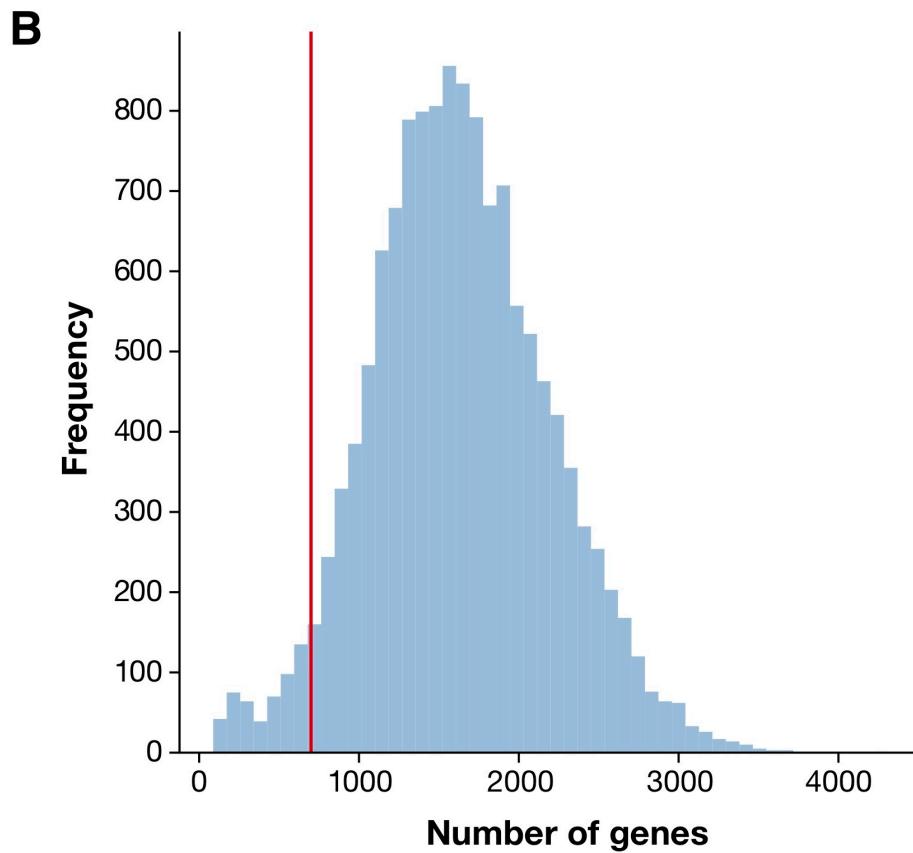
Top

Step 2 – Quality control and filtering



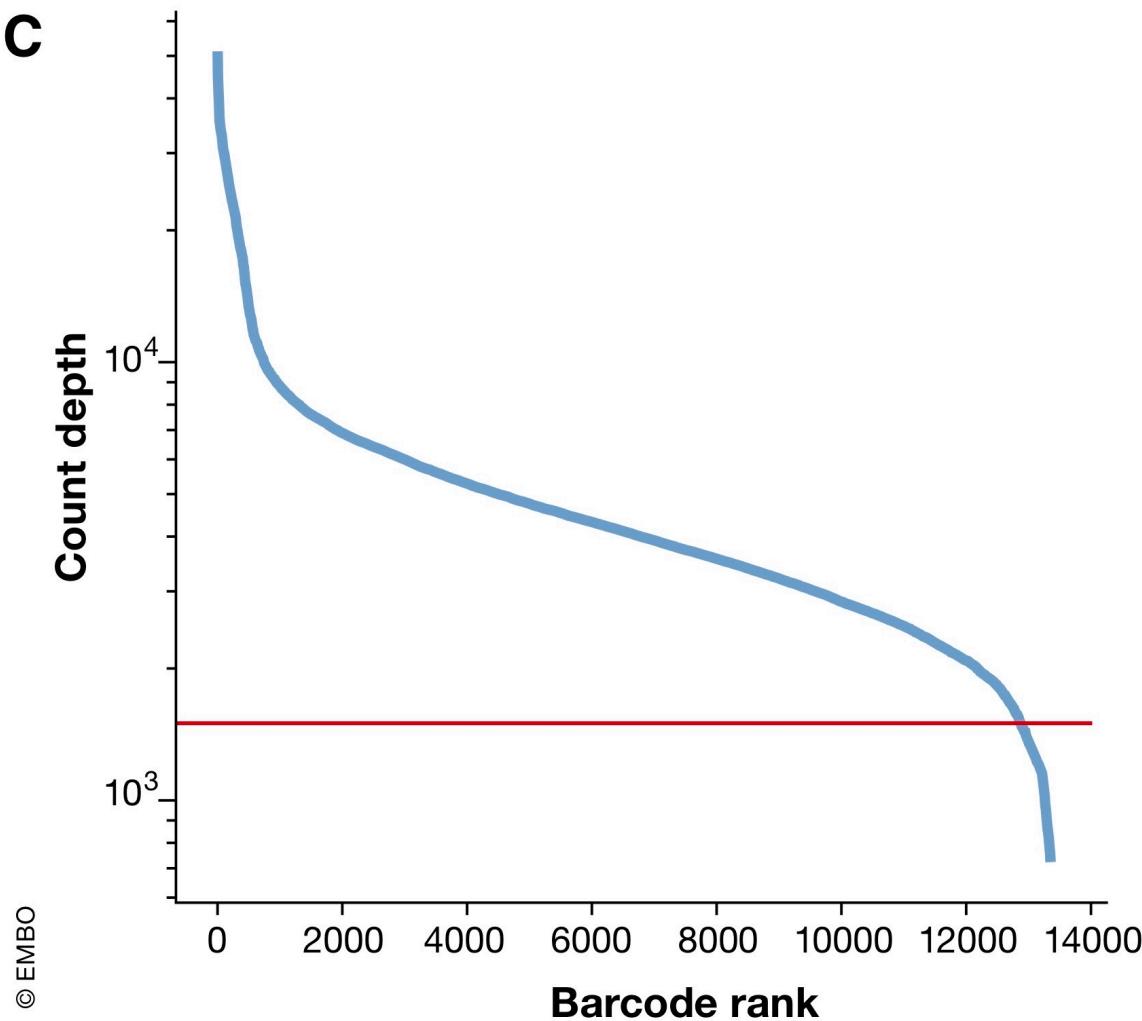
Building Counts Matrix → QC & Filtering → Normalization → Visualization → Analysis

Step 2 – Quality control and filtering



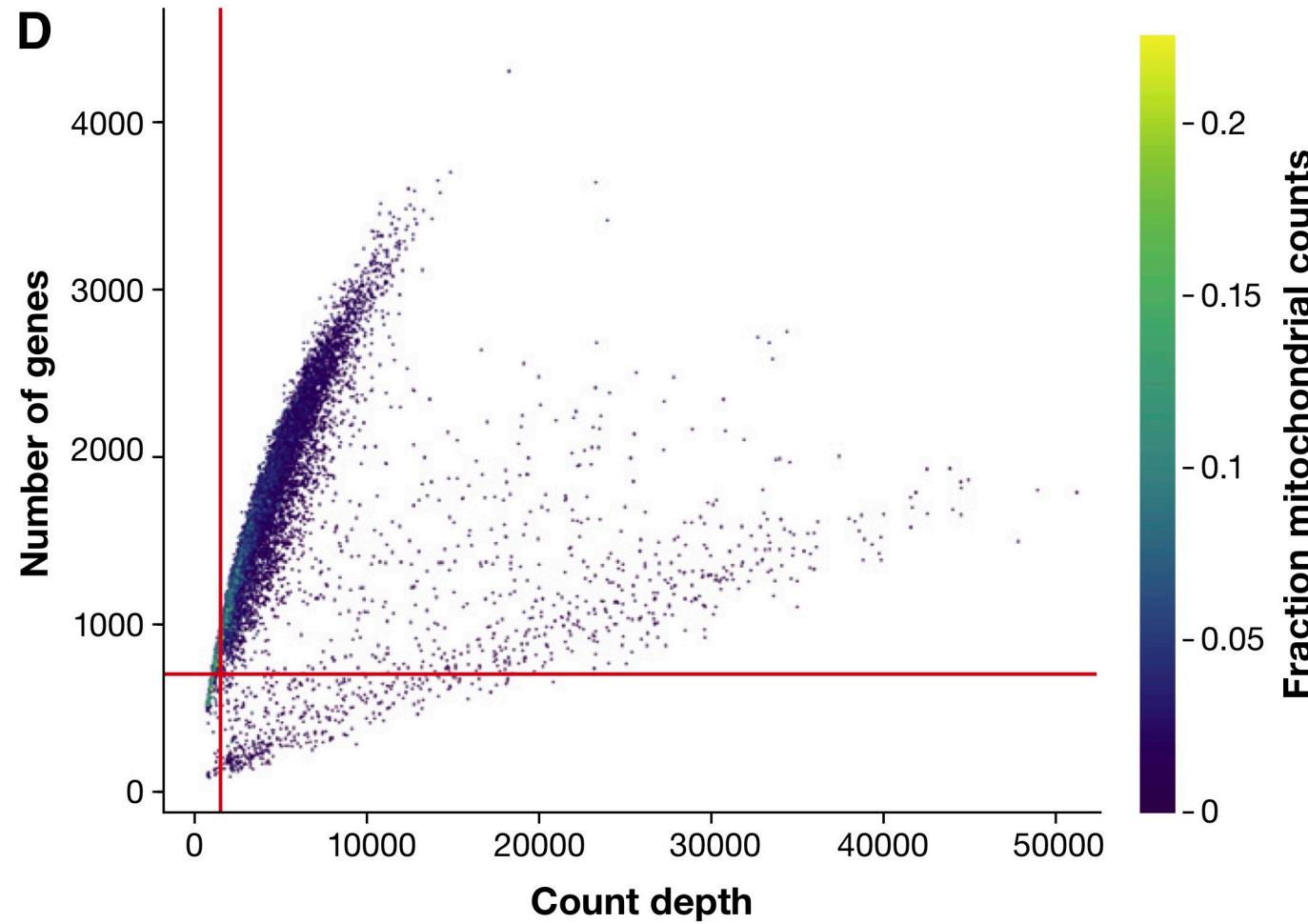
Building Counts Matrix → QC & Filtering → Normalization → Visualization → Analysis

Step 2 – Quality control and filtering



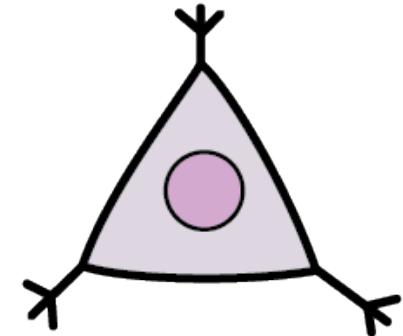
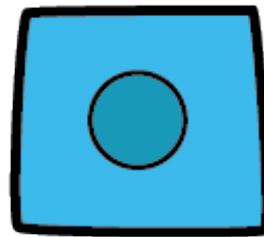
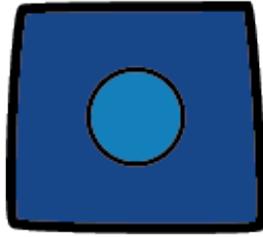
Building Counts Matrix → QC & Filtering → Normalization → Visualization → Analysis

Step 2 – Quality control and filtering



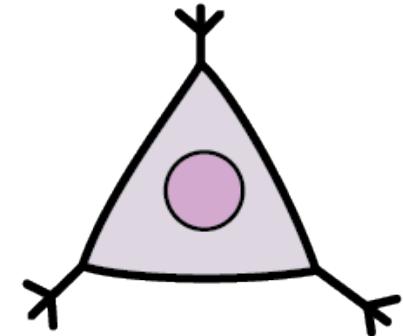
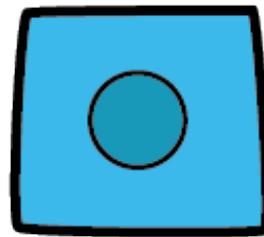
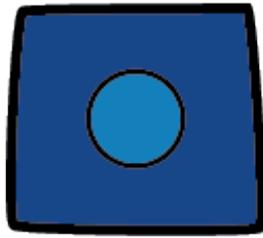
Building Counts Matrix → QC & Filtering → Normalization → Visualization → Analysis

Step 3 - Normalization



If we only have gene expression, how can we determine which cells are similar?

Step 3 - Normalization

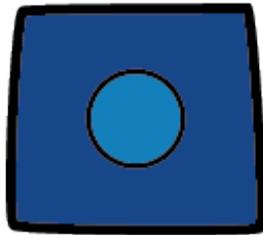


10% Capture Efficiency

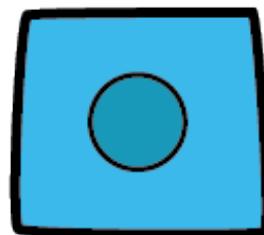
Gene	Cell A
X	10
Y	20
Z	70

Building Counts Matrix → QC & Filtering → Normalization → Visualization → Analysis

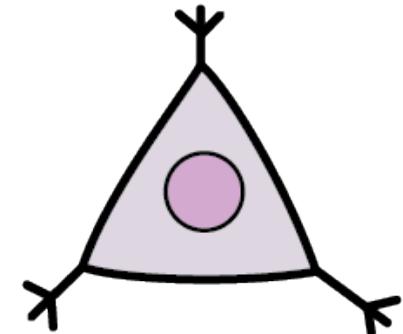
Step 3 - Normalization



10% Capture Efficiency



20% CE

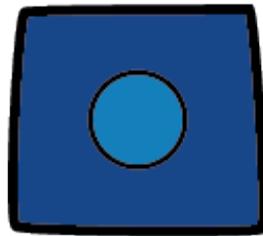


Gene	Cell A
X	10
Y	20
Z	70

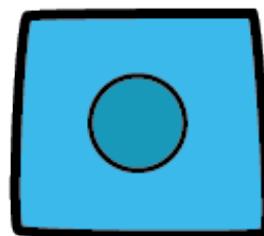
Gene	Cell B
X	20
Y	40
Z	140

Building Counts Matrix → QC & Filtering → Normalization → Visualization → Analysis

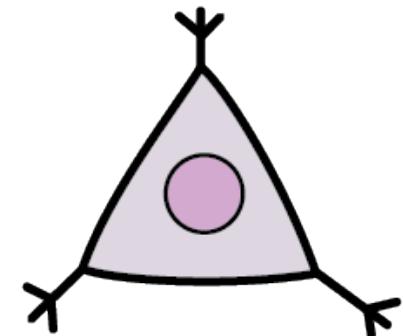
Step 3 - Normalization



10% Capture Efficiency



20% CE



20% CE

Gene	Cell A
X	10
Y	20
Z	70

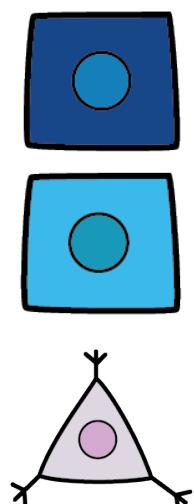
Gene	Cell B
X	20
Y	40
Z	140

Gene	Cell C
X	20
Y	0
Z	80

Building Counts Matrix → QC & Filtering → Normalization → Visualization → Analysis

Step 3 - Normalization

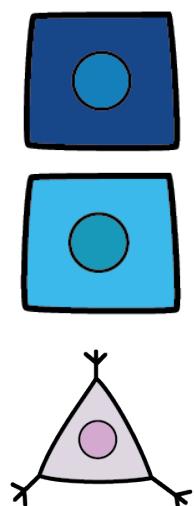
Raw counts



	X	Y	Z
A	10	20	70
B	20	40	140
C	20	0	80

Step 3 - Normalization

Raw counts

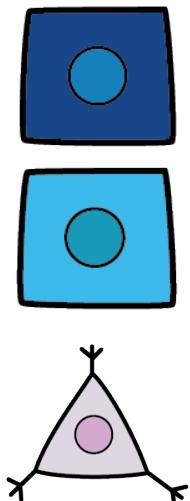


	X	Y	Z
A	10	20	70
B	20	40	140
C	20	0	80

$$\text{dist}(A,B) = \sqrt{(x_A - x_B)^2 + (y_a - y_b)^2 + (z_a - z_b)^2}$$

Step 3 - Normalization

Raw counts



	X	Y	Z
A	10	20	70
B	20	40	140
C	20	0	80

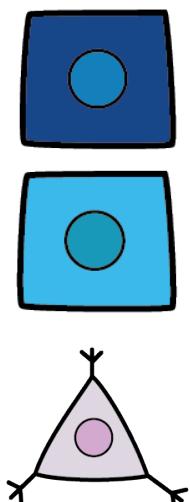
Pairwise distances

$$\text{dist}(A,B) = 71.4$$

$$\text{dist}(A,B) = \sqrt{(x_A - x_B)^2 + (y_a - y_b)^2 + (z_a - z_b)^2}$$

Step 3 - Normalization

Raw counts



	X	Y	Z
A	10	20	70
B	20	40	140
C	20	0	80

Pairwise distances

$$\text{dist}(A,B) = 71.4$$

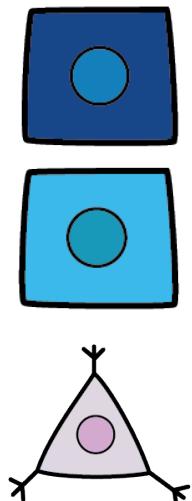
$$\text{dist}(A,C) = 24.5$$

$$\text{dist}(B,C) = 67.1$$

$$\text{dist}(A,B) = \sqrt{(x_A - x_B)^2 + (y_a - y_b)^2 + (z_a - z_b)^2}$$

Step 3 - Normalization

Raw counts

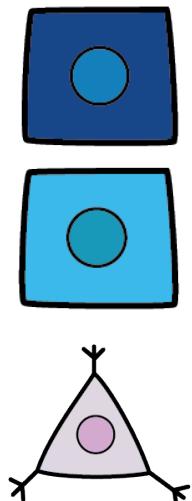


	X	Y	Z	Library Size	Pairwise distances
A	10	20	70	100	$\text{dist}(A,B) = 71.4$
B	20	40	140	200	$\text{dist}(A,C) = 24.5$
C	20	0	80	100	$\text{dist}(B,C) = 67.1$

$$\text{dist}(A,B) = \sqrt{(x_A - x_B)^2 + (y_a - y_b)^2 + (z_a - z_b)^2}$$

Step 3 - Normalization

Normalized counts

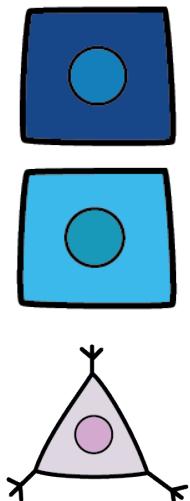


	X	Y	Z	Library Size	Pairwise distances
A	0.1	0.2	0.7	100	$\text{dist}(A,B) = 71.4$
B	0.1	0.2	0.7	200	$\text{dist}(A,C) = 24.5$
C	0.2	0	0.8	100	$\text{dist}(B,C) = 67.1$

$$\text{dist}(A,B) = \sqrt{(x_A - x_B)^2 + (y_a - y_b)^2 + (z_a - z_b)^2}$$

Step 3 - Normalization

Normalized counts

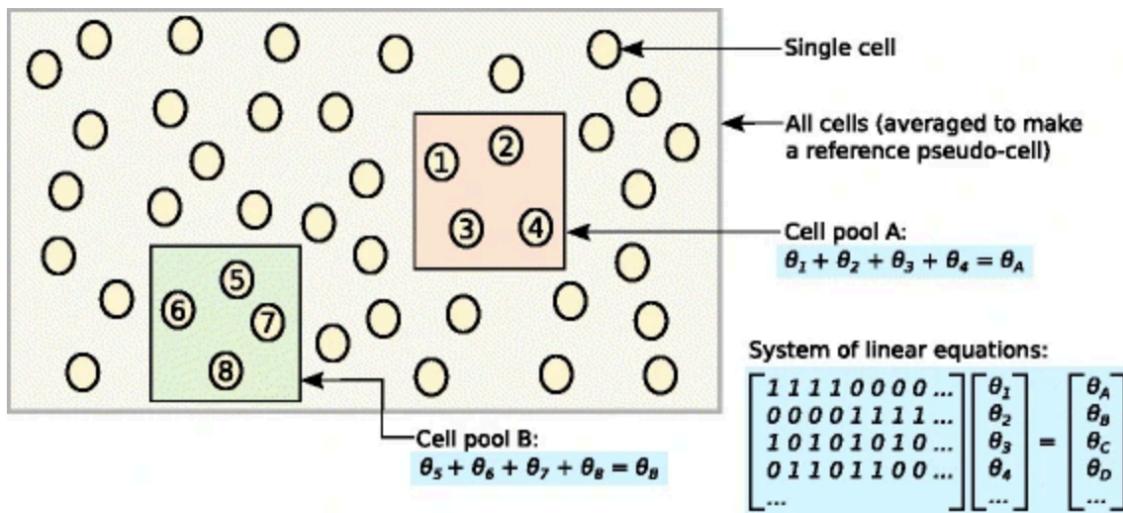


	X	Y	Z	Library Size	Pairwise distances
A	0.1	0.2	0.7	100	$\text{dist}(A,B) = 0$
B	0.1	0.2	0.7	200	$\text{dist}(A,C) = 0.25$
C	0.2	0	0.8	100	$\text{dist}(B,C) = 0.25$

$$\text{dist}(A,B) = \sqrt{(x_A - x_B)^2 + (y_a - y_b)^2 + (z_a - z_b)^2}$$

More complex normalization approaches exist

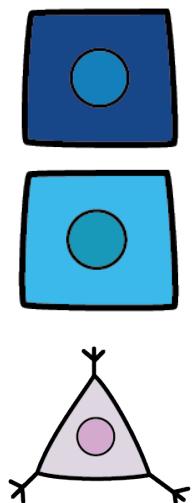
Fig. 3



Schematic of the deconvolution method. All cells in the data set are averaged to make a reference pseudo-cell. Expression values for cells in pool A are summed together and normalized against the reference to yield a pool-based size factor θ_A . This is equal to the sum of the cell-based factors θ_j for cells $j=1-4$ and can be used to formulate a linear equation. (For simplicity, the t_j term is assumed to be unity here.) Repeating this for multiple pools (e.g., pool B) leads to the construction of a linear system that can be solved to estimate θ_j for each cell j

Step 3.5 – Transformation / Scaling

Normalized counts

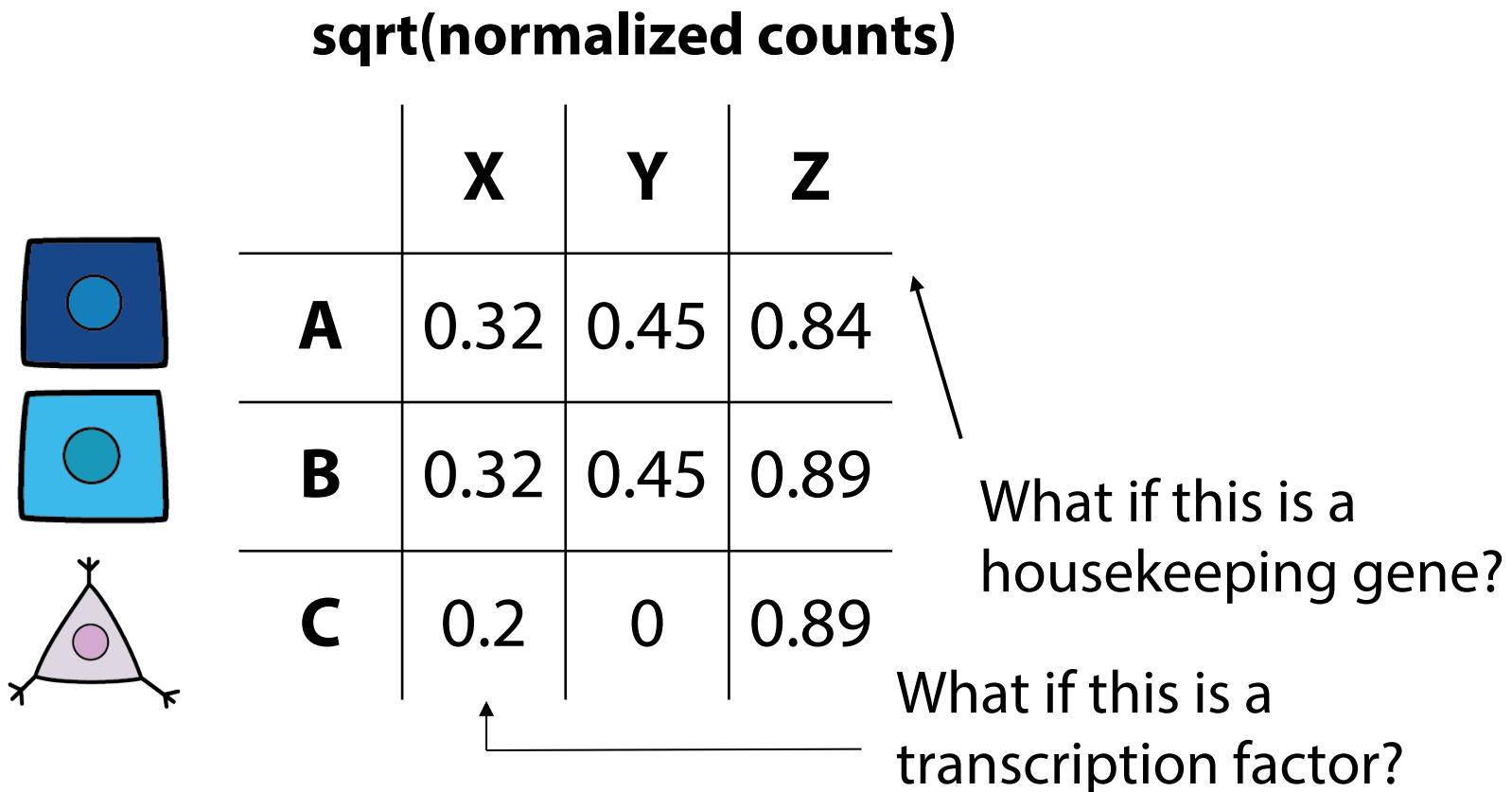


	X	Y	Z
A	0.1	0.2	0.7
B	0.1	0.2	0.7
C	0.2	0	0.8

What if this is a housekeeping gene?

What if this is a transcription factor?

Step 3.5 – Transformation / Scaling



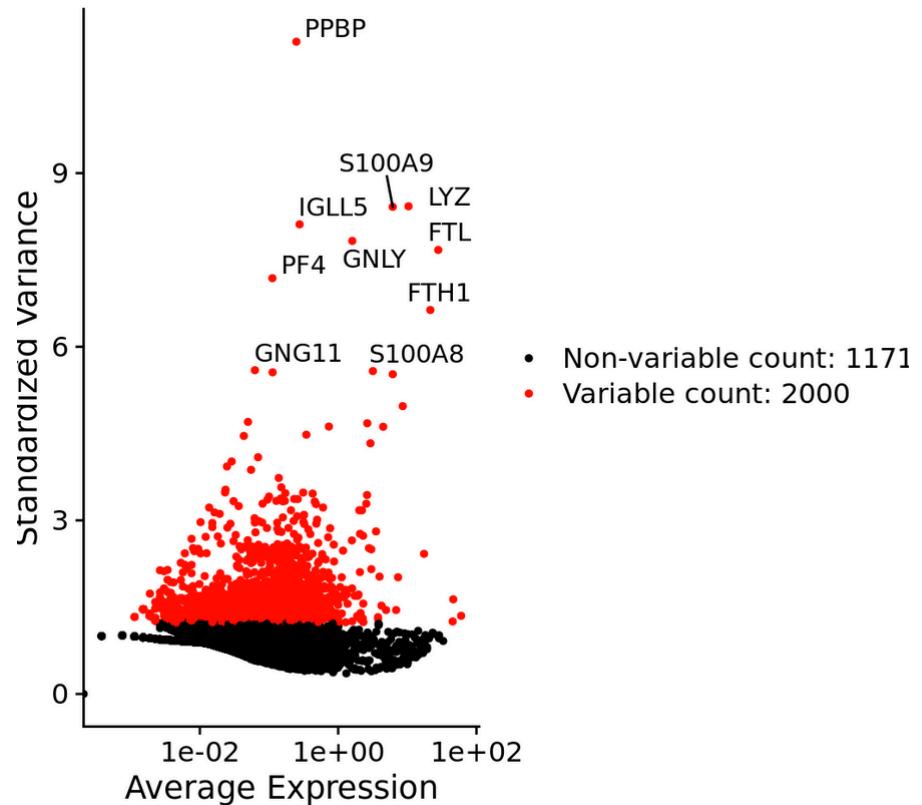
What kind of transformations, other than square-root, could we apply to single cell data?

Top

Step 5 – Dimensionality reduction and visualization

Selecting highly variable genes (HVGs):

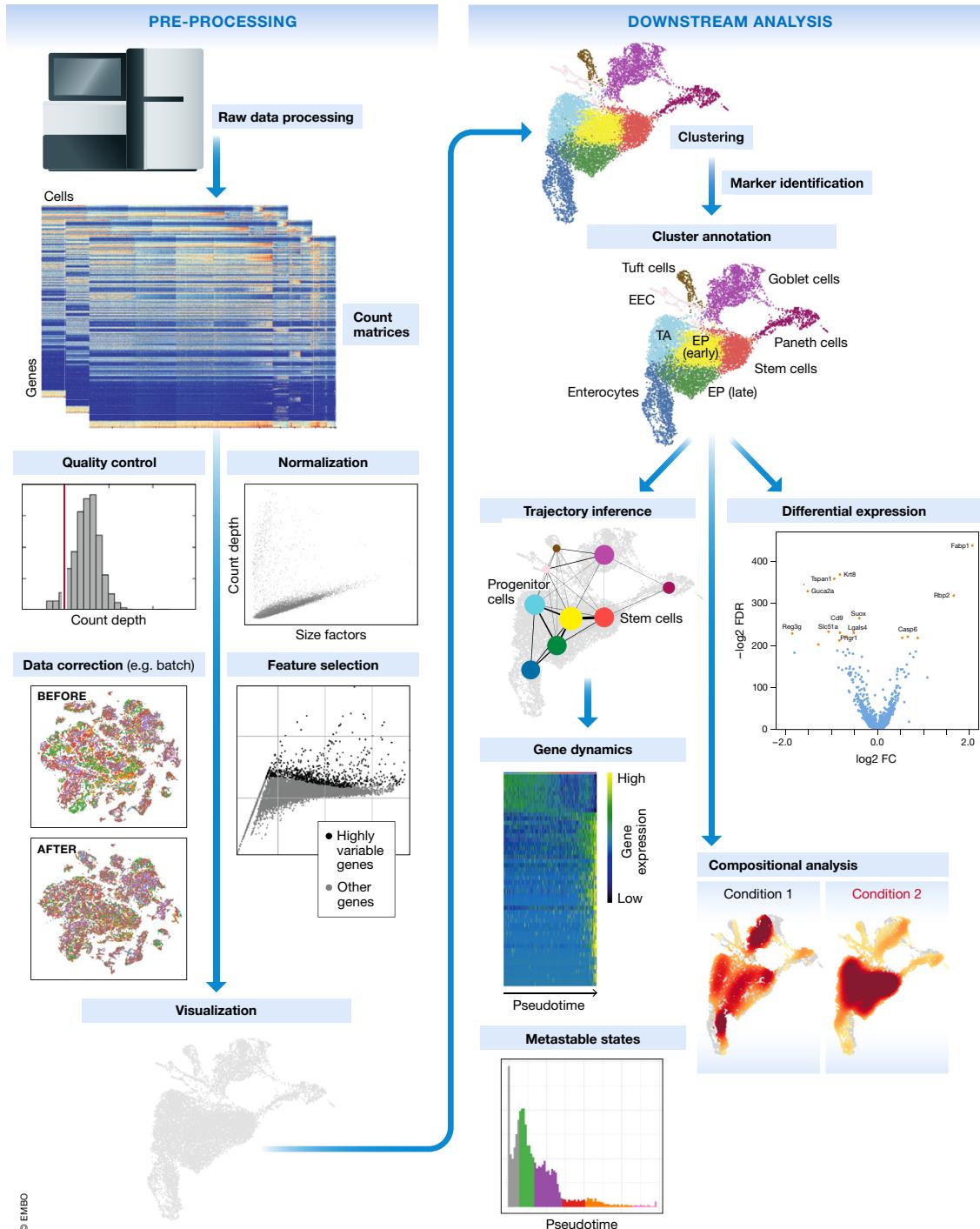
- Calculate log10 mean expression and variance
- Fit a loess curve
- Standardize variance to mean 0 std 1
- Take the top 2000 HVGs



Building Counts Matrix → QC & Filtering → Normalization → Visualization → Analysis

Standard Single-Cell RNA-seq Workflow

1. Sequencing and read mapping
2. Quality control and filtering
3. Normalization
4. Data Correction
5. Dimensionality reduction and visualization
6. Downstream analysis
 1. Clustering
 2. Trajectory inference
 3. Differential expression
7. Comparison of multiple conditions



What questions do you have about today's material?

Top



Exercise!

Load, preprocess, and visualize a scRNAseq dataset generated from a time course of embryoid bodies

