Single-cell RNA-seq analysis

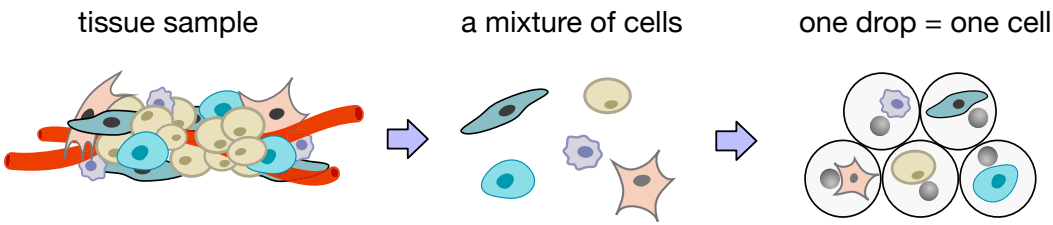
Yongjin Park
University of British Columbia

27 March, 2022

Today's lecture

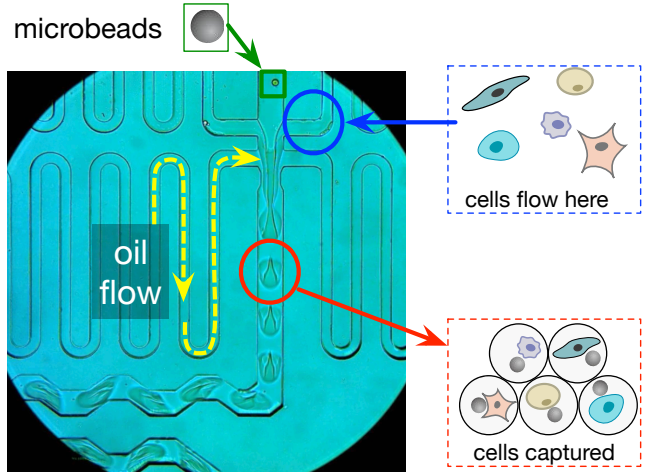
- Single-cell sequencing technology
- Basic Data Q/C
- Doublet detection in single-cell data
- Data normalization across many batches
- Latent topic modelling
- Other interesting topics in scRNA-seq analysis

Droplet-based single-cell sequencing technology



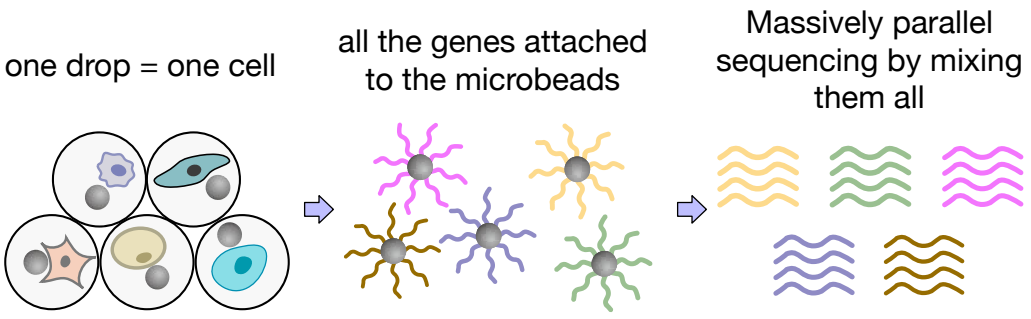
Macosko *et al.*, *Cell* (2015)

Drop-seq idea 1: Capture one cell with a microbead in a droplet



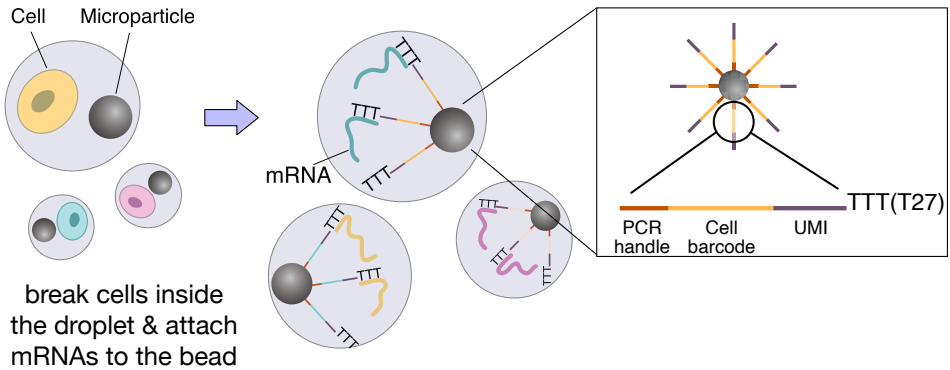
Macosko *et al.*, *Cell* (2015)

Drop-seq idea 2: Massively-parallel sequencing followed by cell-specific barcoding



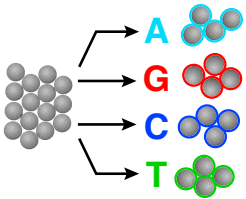
Macosko *et al.*, *Cell* (2015)

Drop-seq idea 3: How do we keep track of mRNA short reads' membership to a certain droplet?



Macosko *et al.*, *Cell* (2015)

How do we construct millions of unique barcodes? Use DNA as a hashing function!

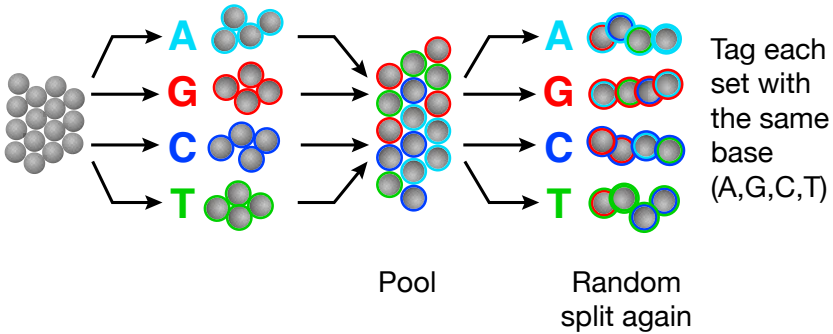


Randomly split the beads into 4 sets

Tag each set with the same base (A,G,C,T)

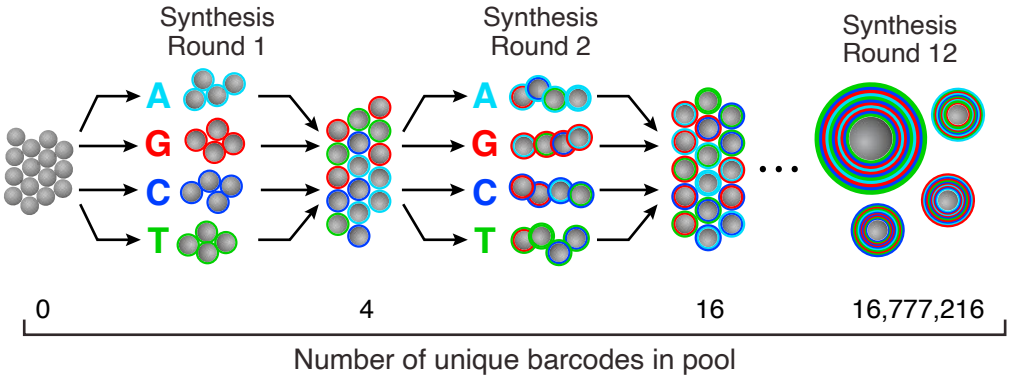
Macosko *et al.*, *Cell* (2015)

How do we construct millions of unique barcodes? Use DNA as a hashing function!



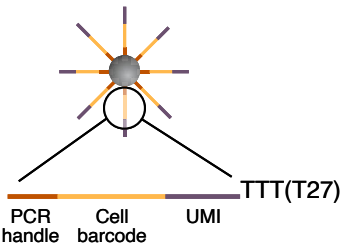
Macosko *et al.*, *Cell* (2015)

How do we construct millions of unique barcodes? Use DNA as a hashing function!

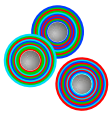


Macosko *et al.*, *Cell* (2015)

The lengths of barcode sequences determine data dimensionality



4^{12} 4^{8}



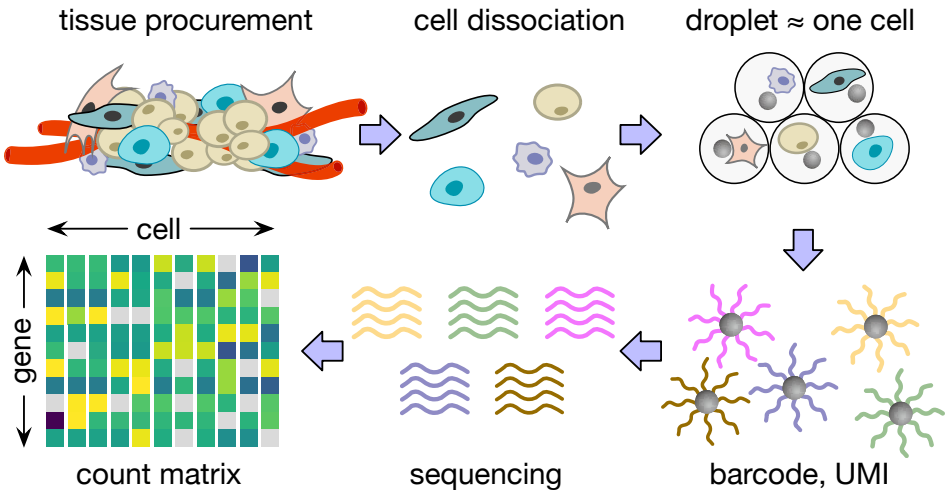
+ × 8 rounds of synthesis

- Millions of the same **cell barcode** per bead
- 4^8 different **molecular barcodes** (UMIs) per bead

Technically, we can build up to a 65,536 × 16,777,216, gene × cell expression matrix in one single-cell RNA-seq experiment.

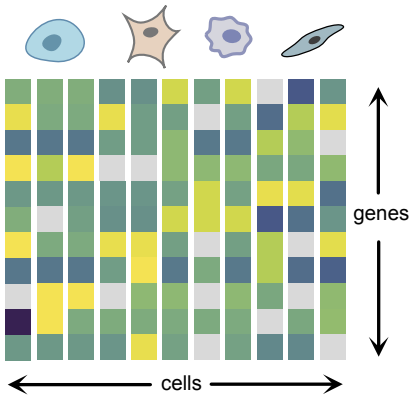
Macosko *et al.*, *Cell* (2015)

Droplet-based single-cell sequencing pipeline

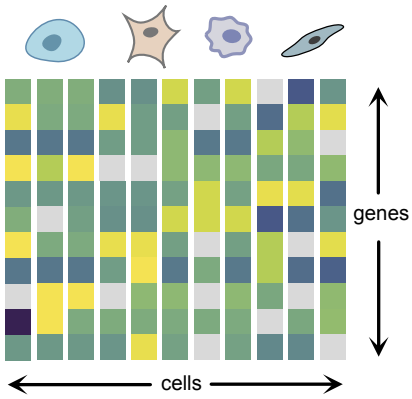


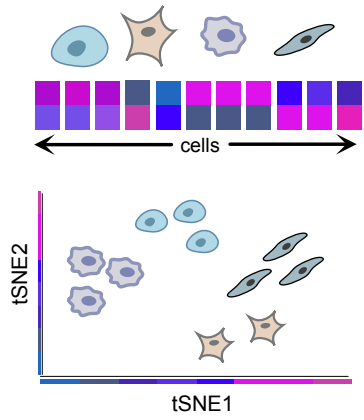
Macosko *et al.*, *Cell* (2015)

People love to see this high-dimensional data matrix in 2D/3D space



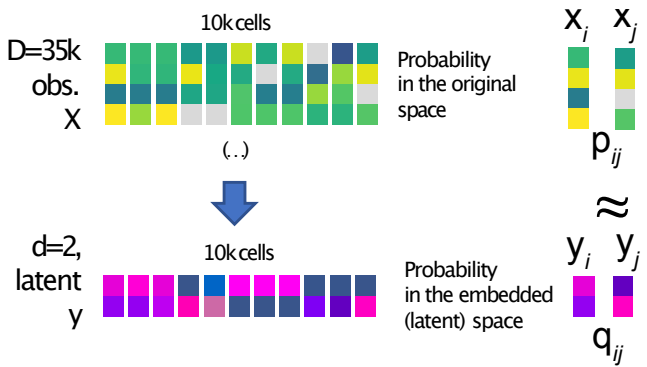
People love to see this high-dimensional data matrix in 2D/3D space





`tSNE`: t-distributed Stochastic Neighbourhood Embedding (Van der Maaten & Hinton, 2008).

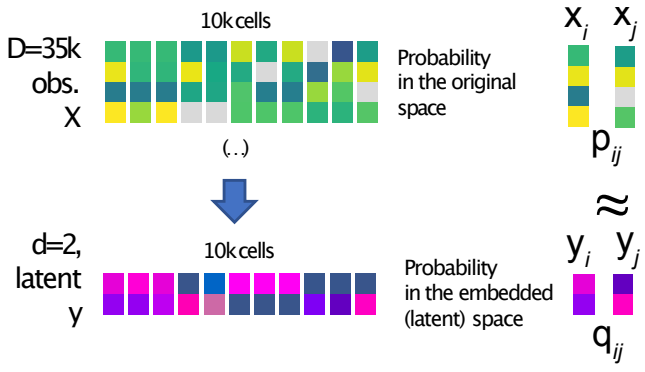
SNE: What is "stochastic neighbourhood embedding?"



- p_{ij}: probability between cells i and j in the original high-dimensional space

$$p_{ij} \propto \exp(\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma^2)$$

SNE: What is "stochastic neighbourhood embedding?"



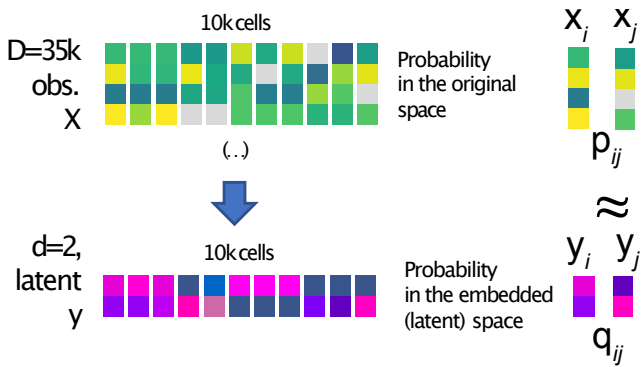
- p_{ij}: probability between cells i and j in the original high-dimensional space

$$p_{ij} \propto \exp(\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma^2)$$

- q_{ij}: probability between cells i and j in the embedded low-dimensional space

$$q_{ij} \propto \exp(\|\mathbf{y}_i - \mathbf{y}_j\|^2 / 2\sigma^2)$$

SNE: What is "stochastic neighbourhood embedding?"



Goal: make pairwise probabilities between cells in the observed and latent space as close as possible.

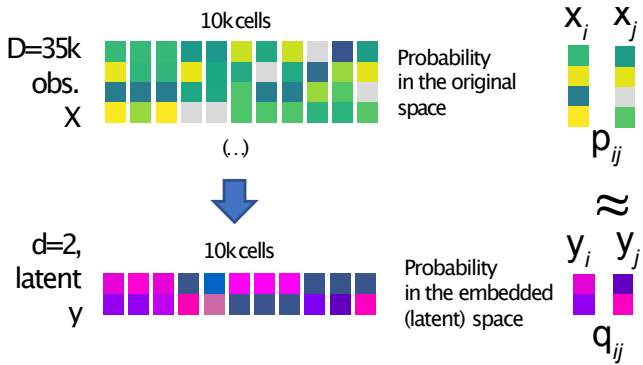
- p_{ij}: probability between cells i and j in the original high-dimensional space

$$p_{ij} \propto \exp(\|\mathbf{x}_i - \mathbf{x}_j\|^2/2\sigma^2)$$

- q_{ij}: probability between cells i and j in the embedded low-dimensional space

$$q_{ij} \propto \exp(\|\mathbf{y}_i - \mathbf{y}_j\|^2/2\sigma^2)$$

SNE: What is "stochastic neighbourhood embedding?"



Goal: make pairwise probabilities between cells in the observed and latent space as close as possible.

- p_{ij}: probability between cells i and j in the original high-dimensional space

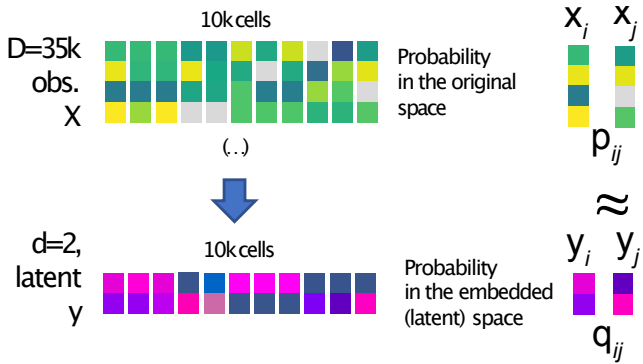
$$p_{ij} \propto \exp(\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma^2)$$

- q_{ij}: probability between cells i and j in the embedded low-dimensional space

$$q_{ij} \propto \exp(\|\mathbf{y}_i - \mathbf{y}_j\|^2 / 2\sigma^2)$$

$$\min D_{\mathrm{KL}}\left(p_{ij} \| q_{ij}\right) = \sum_{ij} p_{ij} \frac{p_{ij}}{q_{ij}}$$

tSNE: What is t-distributed "stochastic neighbourhood embedding?"



- p_{ij}: probability between cells i and j in the original high-dimensional space

$$p_{ij} \propto \exp(\|\mathbf{x}_i - \mathbf{x}_j\|^2/2\sigma^2)$$

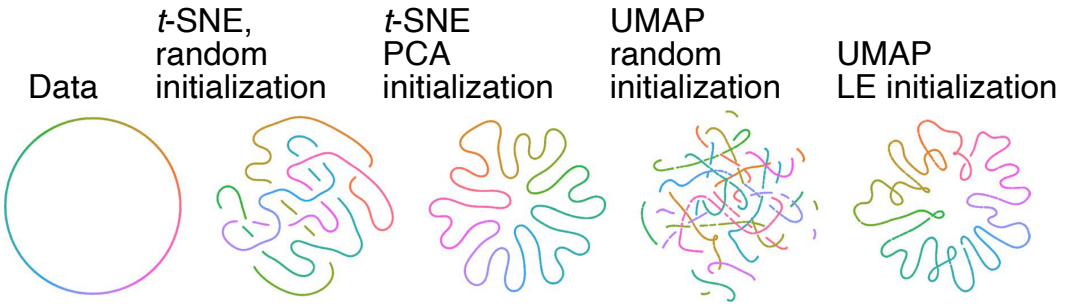
- q_{ij}: probability between cells i and j in the embedded low-dimensional space

$$q_{ij} \propto \left(1 + \|\mathbf{y}_i - \mathbf{y}_j\|^2\right)^{-1}$$

Goal: make pairwise probabilities between cells in the observed and latent space as close as possible.

$$\min D_{\mathsf{KL}}\left(p_{ij} \| q_{ij}\right) = \sum_{ij} p_{ij} \frac{p_{ij}}{q_{ij}}$$

Warning: Don't make over-interpretation on embedding results



Kobak and Berens, *Nature Biotech* (2021)

Check out these papers:

Initialization is critical for preserving global data structure in both t-SNE and UMAP

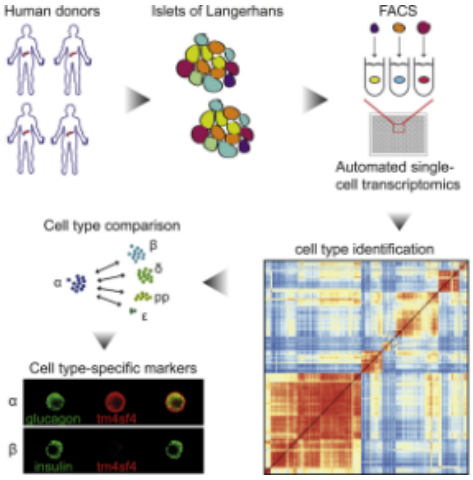
The art of using t-SNE for single-cell transcriptomics

Dimensionality reduction for visualizing single-cell data using UMAP

Today's lecture

- Single-cell sequencing technology
- Basic Data Q/C
- Doublet detection in single-cell data
- Data normalization across many batches
- Latent topic modelling
- Other interesting topics in scRNA-seq analysis

Example: single-cell RNA-seq data of human pancreatic cells

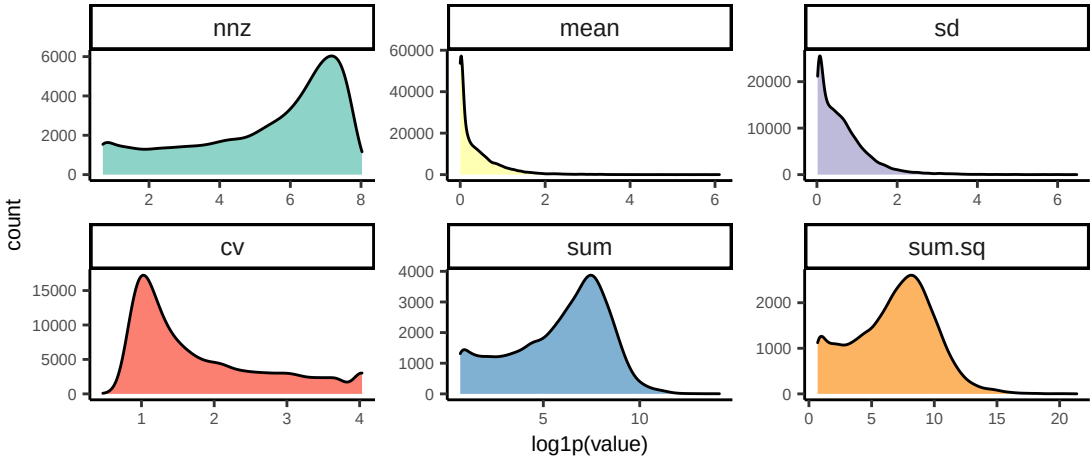


Muraro *et al. Cell Systems* (2016)

We will use scRNA-seq data (GEO accession: GSE85241) as a working example.

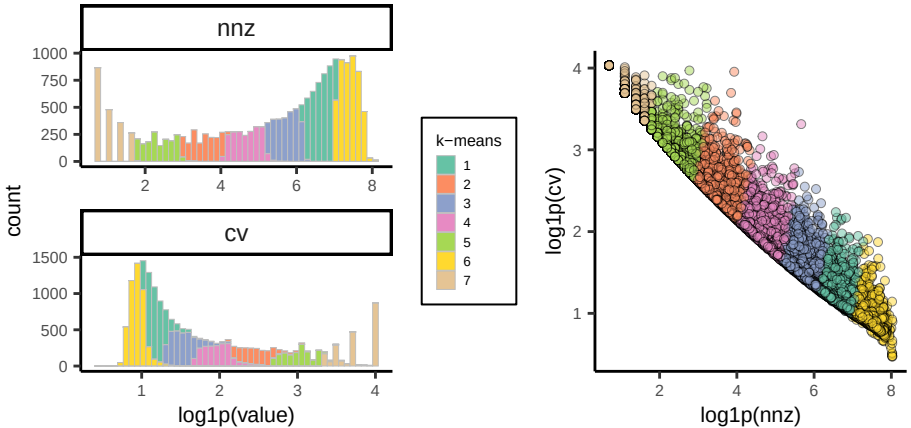
- genes/features/rows: 19,140
- cells/columns: 3,072
- non-zero elements: 12,442,034
- $\sim$ 21 % non-zero

Gene-level statistics across cells



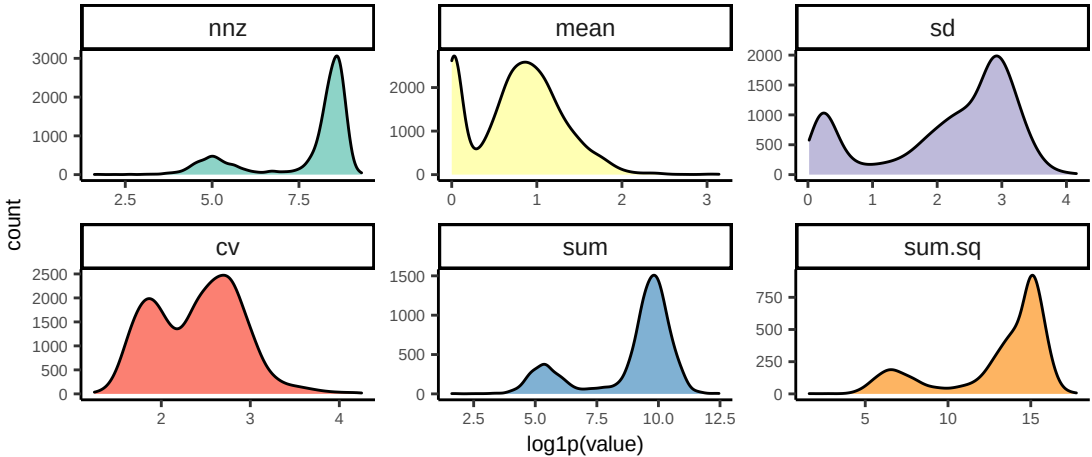
`nnz`: number of non-zero elements, `sd`: standard deviation, `cv`: coefficient of variation (`sd`/`mean`), `sum.sq`: sum of squares.

Can we drop any genes?



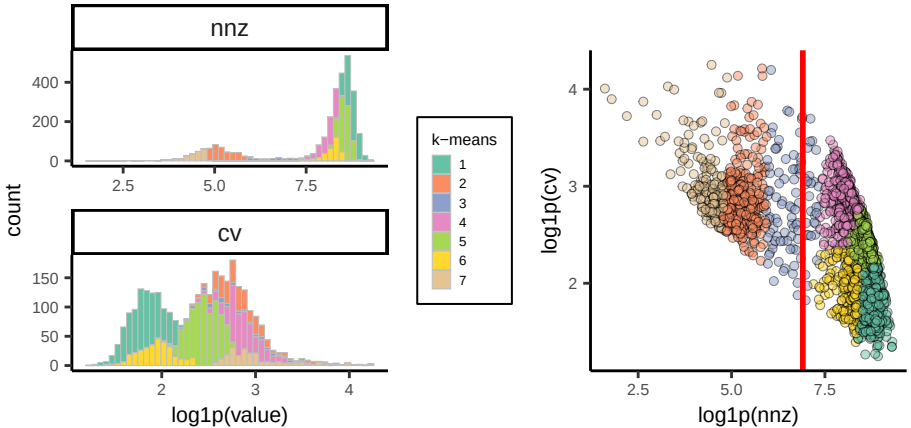
We don't like unstable genes with low average expressions and high CV.

Cell-level statistics across genes within each cell



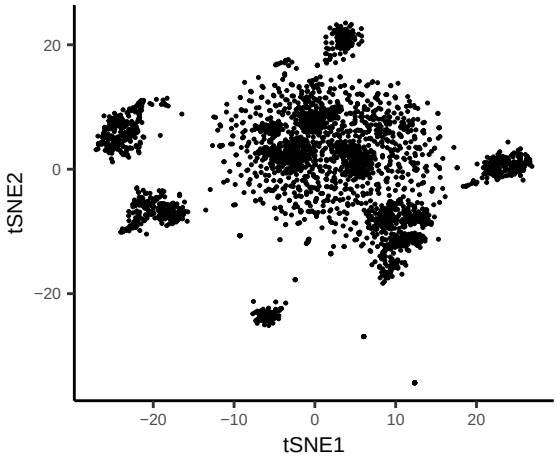
`nnz`: number of non-zero elements, `sd`: standard deviation, `cv`: coefficient of variation (`sd`/`mean`), `sum.sq`: sum of squares.

Filter out cells that are not informative...?

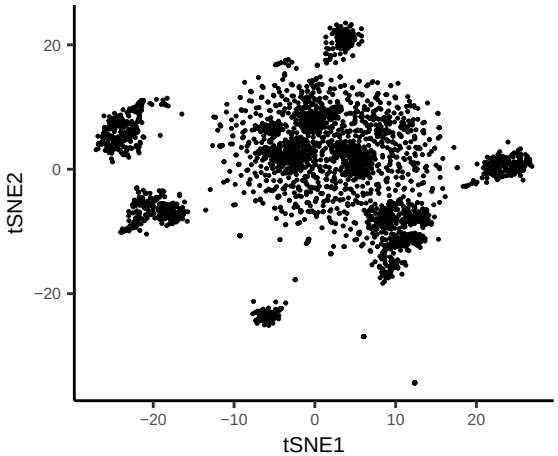


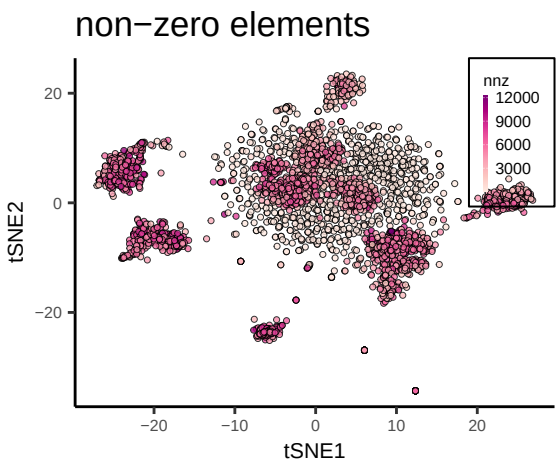
We may remove cells with too few non-zero elements (e.g., NNZ < 1000).

Exploratory Data Analysis with tSNE

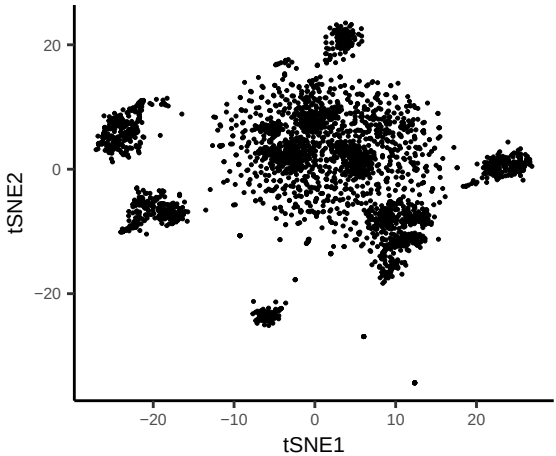


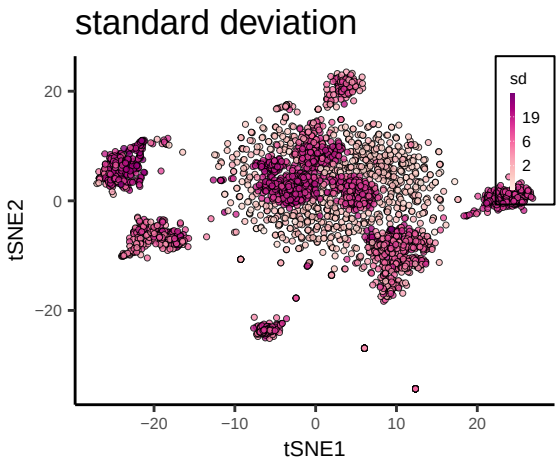
Exploratory Data Analysis with tSNE



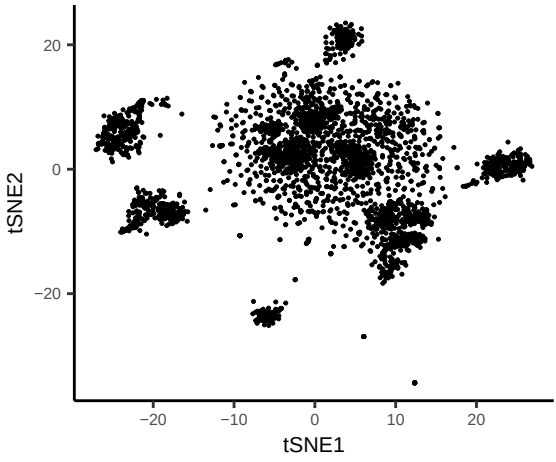


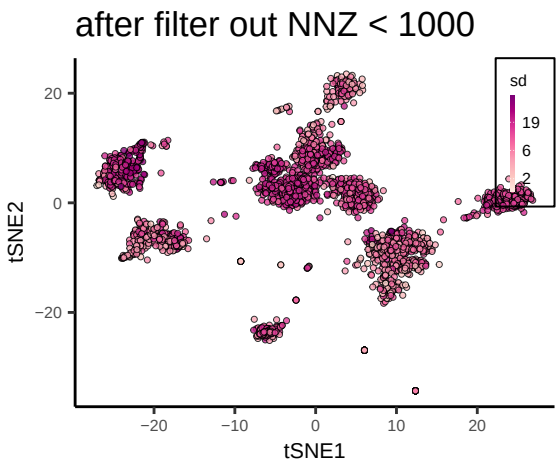
Exploratory Data Analysis with tSNE





Exploratory Data Analysis with tSNE

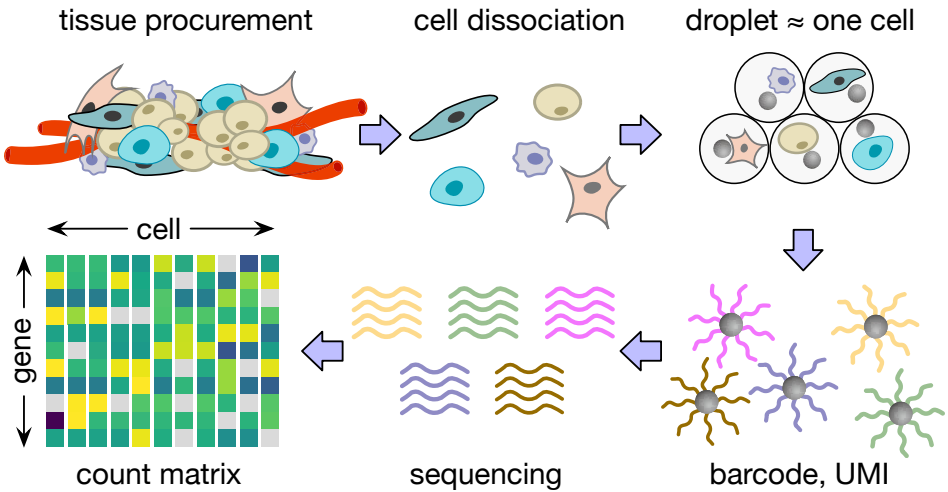




Today's lecture

- Single-cell sequencing technology
- Basic Data Q/C
- Doublet detection in single-cell data
- Data normalization across many batches
- Latent topic modelling
- Other interesting topics in scRNA-seq analysis

scRNA-seq pipeline: What if we capture more than one cell in a droplet?



Macosko *et al.*, *Cell* (2015)

What is a doublet in single-cell data?

Biological/technical definition:

- One or more cells captured (usually at most two cells by chance)
- Thus, multiple cells accidental share the same cell barcode sequence
- Not so clear in general... since we missed the chance to assign different tags to different cells encapsulated in the same droplet.

Statistical definition:

- If we could find marker genes of multiple cell types are simultaneously expressed...
- An unvetted approach: Find ambiguous/intermediate coordinates in PCA/tSNE/UMAP (after removing ambient cells).

Can we create artificial doublets?

A straightforward definition (used in `DoubletFinder`):

For each cell i:

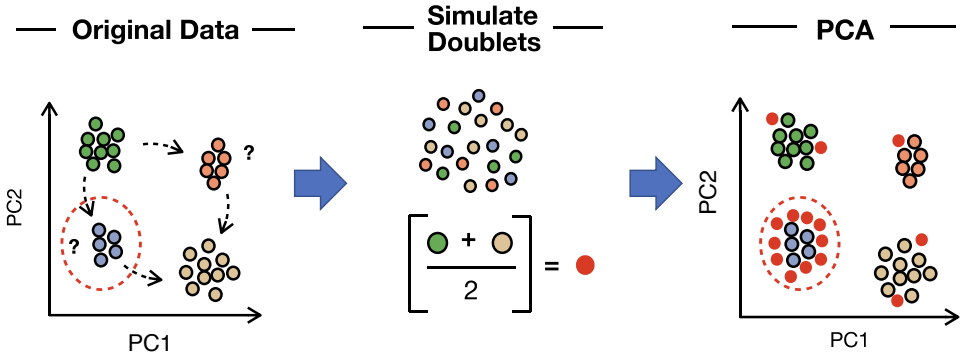
- Take some other j by random selection
- Create an artificial doublet

$$\tilde{\mathbf{x}} \leftarrow \frac{1}{2}(\mathbf{x}_i + \mathbf{x}_j)$$

Some thought questions:

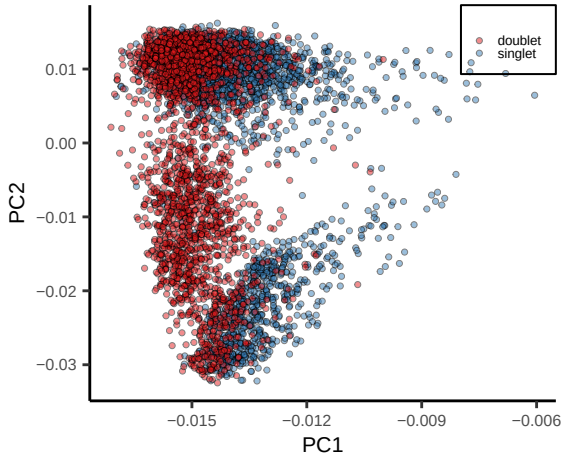
- Doublets within the same cell type?
- Doublets between the different cell types?

k-Nearest Neighbour classification for doublet detection

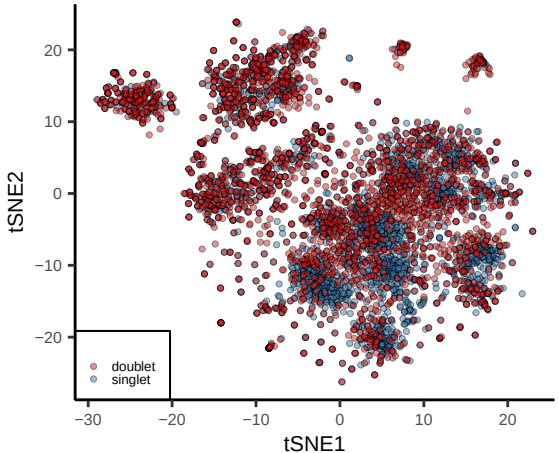


McGinnis et al. Cell Systems (2019)

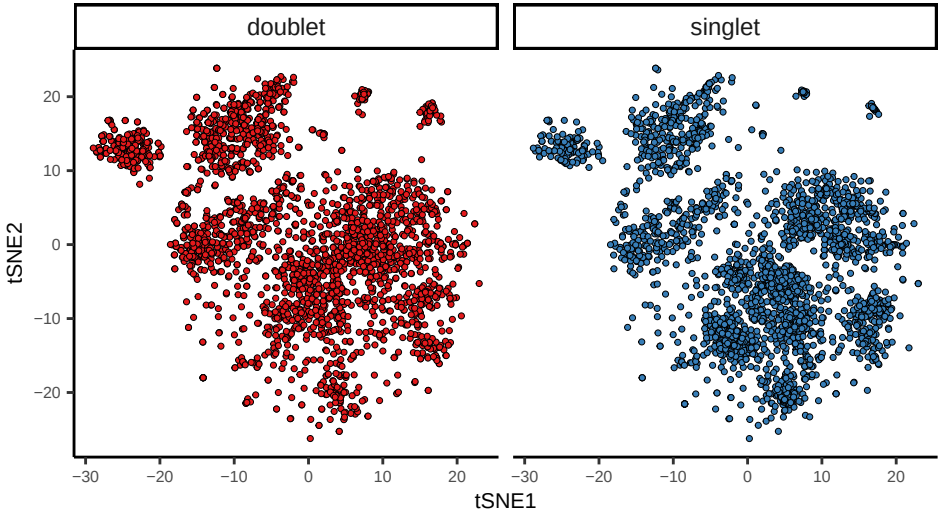
Can you tell the difference by a quick visual inspection?



Can you tell the difference by a quick visual inspection?



Can you tell the difference by a quick visual inspection?



Can we design a classifier to distinguish singlets vs. doublets?

k-Nearest Neighbour classification for doublet detection

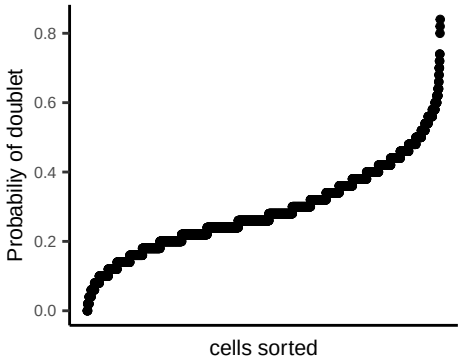
- Step 1. Create artificial doublets, $\tilde{\mathbf{x}}$
- Step 2. Mix them with the original cells and perform PCA
- Step 3. Find nearest neighbours of the original cells (using #PC=50)
- Step 4. Count the number of doublets in the neighbourhood

k-Nearest Neighbour classification for doublet detection

- **Q: How many of my neighbours are indeed a doublet?**

$$\hat{P}_i = \frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} I\{j \text{ is a doublet}\}$$

Key assumption: There is a principal component that can set apart hidden doublets from the most of singlets.

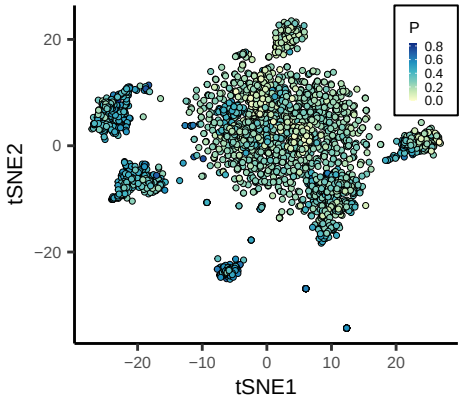


k-Nearest Neighbour classification for doublet detection

- **Q: How many of my neighbours are indeed a doublet?**

$$\hat{P}_i = \frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} I\{j \text{ is a doublet}\}$$

Key assumption: There is a principal component that can set apart hidden doublets from the most of singlets.

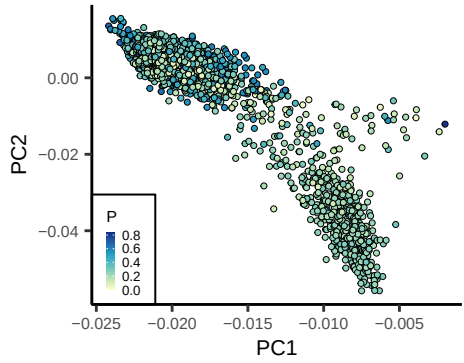


k-Nearest Neighbour classification for doublet detection

- **Q: How many of my neighbours are indeed a doublet?**

$$\hat{P}_i = \frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} I\{j \text{ is a doublet}\}$$

Key assumption: There is a principal component that can set apart hidden doublets from the most of singlets.

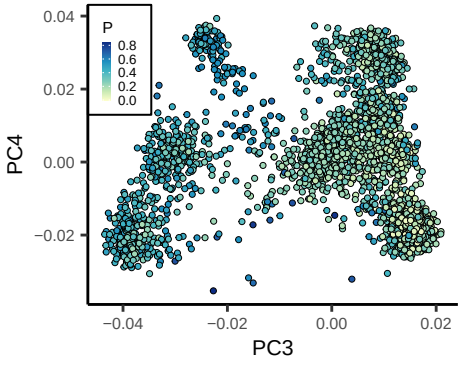


k-Nearest Neighbour classification for doublet detection

- **Q: How many of my neighbours are indeed a doublet?**

$$\hat{P}_i = \frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} I\{j \text{ is a doublet}\}$$

Key assumption: There is a principal component that can set apart hidden doublets from the most of singlets.

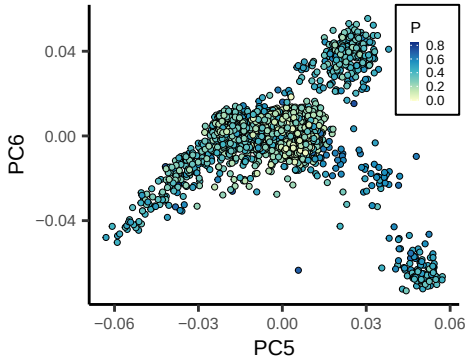


k-Nearest Neighbour classification for doublet detection

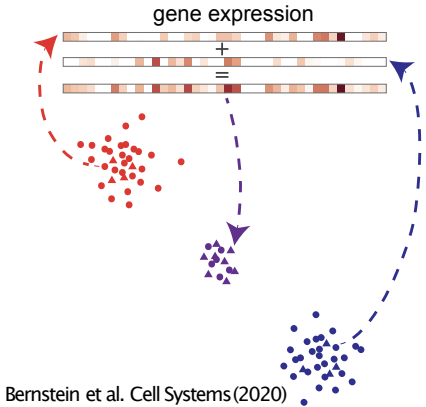
- **Q: How many of my neighbours are indeed a doublet?**

$$\hat{P}_i = \frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} I\{j \text{ is a doublet}\}$$

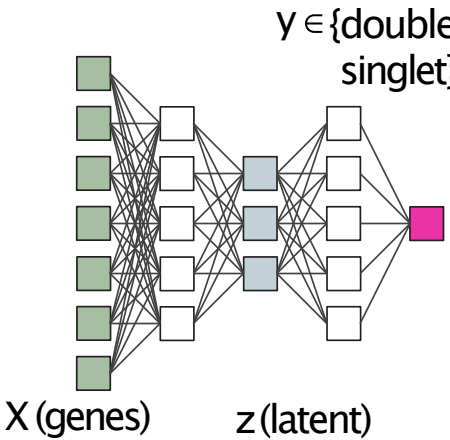
Key assumption: There is a principal component that can set apart hidden doublets from the most of singlets.



Artificial Neural Network-based classification for doublet detection



Bernstein et al. Cell Systems (2020)



Training a parametric classifier to discern doublets vs. singlets

$$f : \mathbf{x}_i \rightarrow y_i,\ y \in \{0, 1\}$$

```
build.classifier <-
    nn_module(
        classname = "classfier",
        initialize = function(d, k = 5) {
            self$fc <- nn_sequential(
                nn_batch_norm1d(d),
                nn_linear(d, k),
                nn_batch_norm1d(k),
                nn_relu(),
                nn_linear(k, 2 * k),
                nn_batch_norm1d(2 * k),
                nn_relu(),
                nn_linear(2 * k, 1),
                nn_sigmoid())
        },
        forward = function(x, min_=.01, max_=.99) {
            torch_clamp(self$fc(x), min_, max_)
        })
```

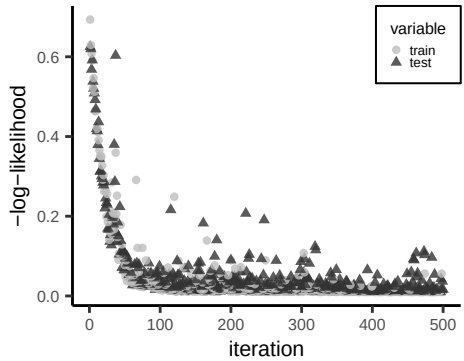
$$\prod_{i=1}^{n} f(\mathbf{x}_i)^{y_i}(1 - f(\mathbf{x}_i))^{1-y_i}$$

Training a parametric classifier to discern doublets vs. singlets

$$f : \mathbf{x}_i \rightarrow y_i,\ y \in \{0, 1\}$$

```
build.classifier <-
    nn_module(
        classname = "classfier",
        initialize = function(d, k = 5) {
            self$fc <- nn_sequential(
                nn_batch_norm1d(d),
                nn_linear(d, k),
                nn_batch_norm1d(k),
                nn_relu(),
                nn_linear(k, 2 * k),
                nn_batch_norm1d(2 * k),
                nn_relu(),
                nn_linear(2 * k, 1),
                nn_sigmoid())
        },
        forward = function(x, min_=.01, max_=.99) {
            torch_clamp(self$fc(x), min_, max_)
        })
```

$$\prod_{i=1}^{n} f(\mathbf{x}_i)^{y_i}(1 - f(\mathbf{x}_i))^{1-y_i}$$

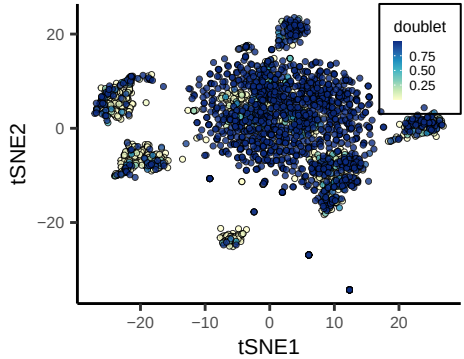


Training a parametric classifier to discern doublets vs. singlets

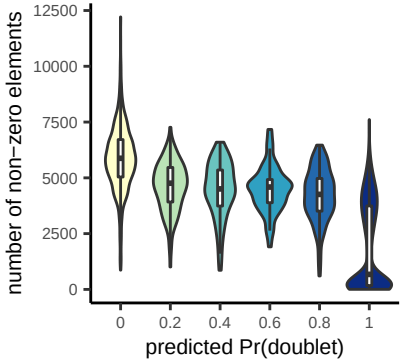
$$f : \mathbf{x}_i \rightarrow y_i,\ y \in \{0, 1\}$$

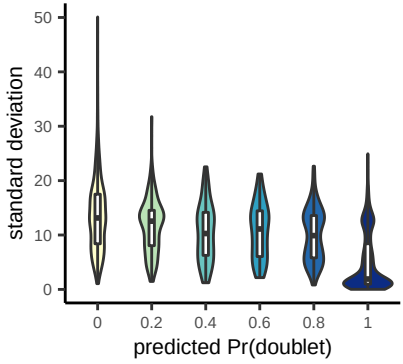
```
build.classifier <-
    nn_module(
        classname = "classfier",
        initialize = function(d, k = 5) {
            self$fc <- nn_sequential(
                nn_batch_norm1d(d),
                nn_linear(d, k),
                nn_batch_norm1d(k),
                nn_relu(),
                nn_linear(k, 2 * k),
                nn_batch_norm1d(2 * k),
                nn_relu(),
                nn_linear(2 * k, 1),
                nn_sigmoid())
        },
        forward = function(x, min_=.01, max_=.99) {
            torch_clamp(self$fc(x), min_, max_)
        })
```

$$\prod_{i=1}^{n} f(\mathbf{x}_i)^{y_i}(1 - f(\mathbf{x}_i))^{1-y_i}$$



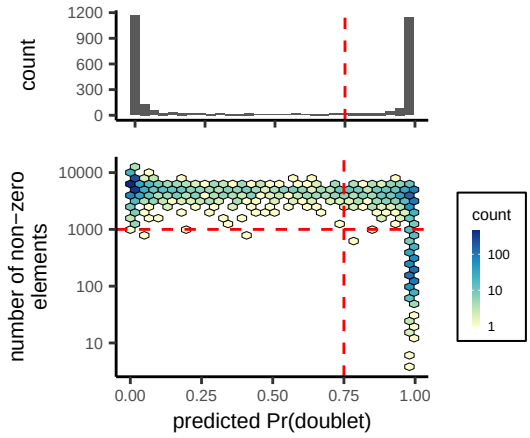
The predicted doublets generally correspond to cells with few non-zero elements





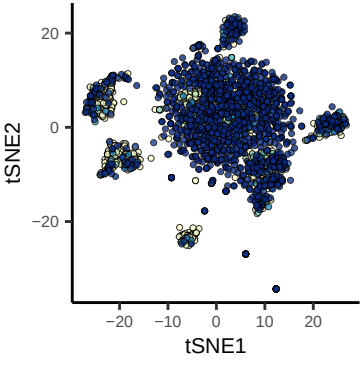
Low expression within a cell may stem from unwanted burst-out cells or ambient RNA molecules.

The predicted doublets generally correspond to cells with few non-zero elements

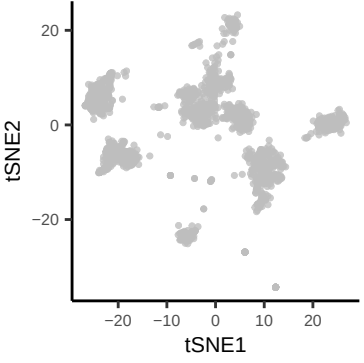


After removing potential doublets

- All the cells

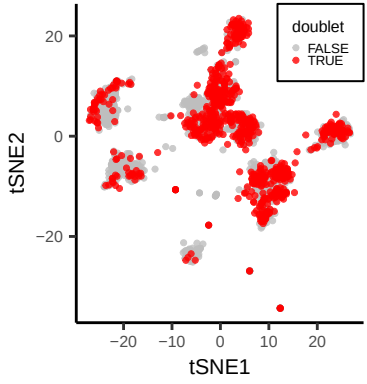


- P(doublet) $< .75$, #non-zeros > 1k

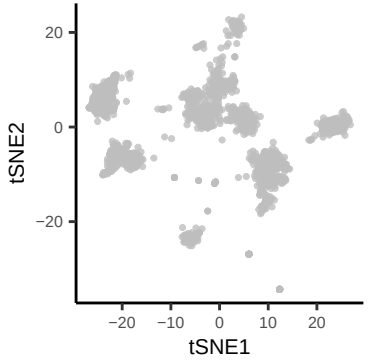


Maybe it is more than just low expression cells

- #non-zeros > 1k



- P(doublet) $< .75$, #non-zeros > 1k



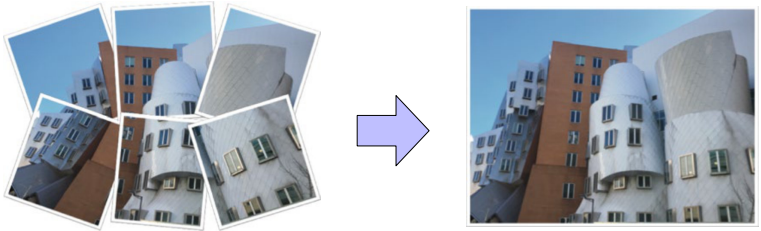
Discussion on doublet Q/C

- It's a unique routine in single-cell sequencing data analysis
- Do we need it in practice? How frequently doublets emerge?
- Perhaps a majority of them simply stem from "dying" cells or broken cells... If so, we can just filter out low-expressed cells.

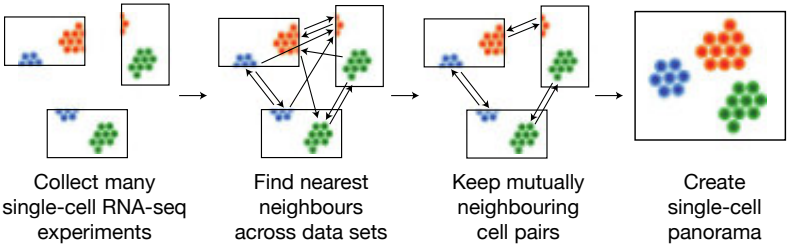
Today's lecture

- Single-cell sequencing technology
- Basic Data Q/C
- Doublet detection in single-cell data
- Data normalization across many batches
- Latent topic modelling
- Other interesting topics in scRNA-seq analysis

Batch normalization for joint analysis across multiple single-cell RNA-seq data

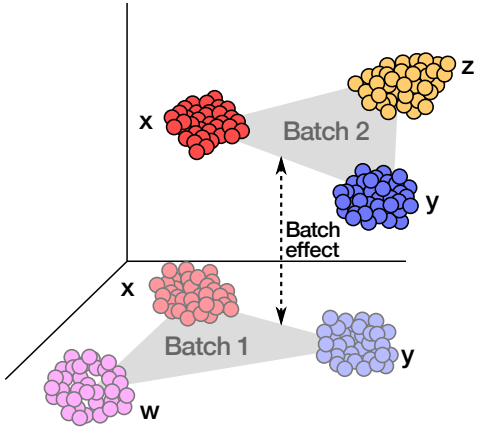


How do we integrate multiple samples/batches?



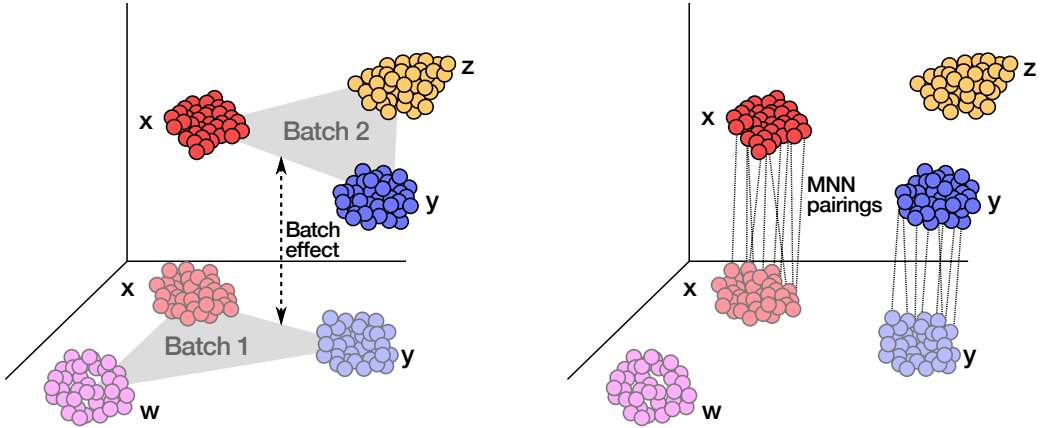
Scanorama: mutual nearest neighbourhood-based data integration

Batch normalization aims to minimize the difference between nearest cells across different batches



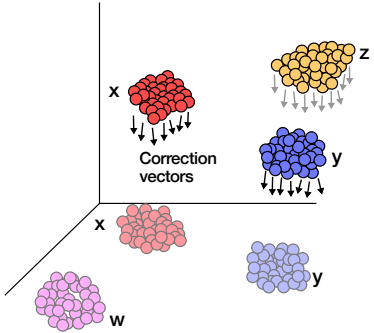
Haghverdi, .., and Marioni, *Nature Biotechnology* (2018)

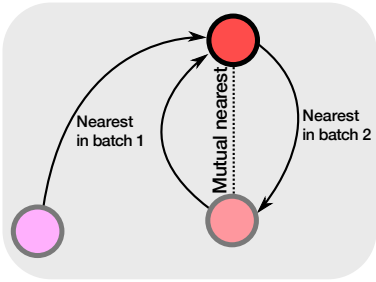
Batch normalization aims to minimize the difference between nearest cells across different batches



Haghverdi, .., and Marioni, *Nature Biotechnology* (2018)

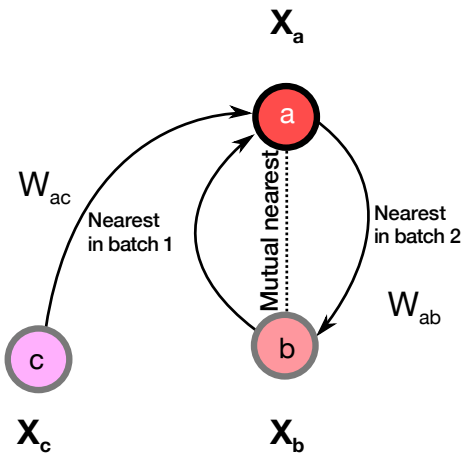
Batch normalization aims to minimize the difference between nearest cells across different batches





Haghverdi, .., and Marioni, *Nature Biotechnology* (2018)

Batch normalization aims to minimize the difference between nearest cells across different batches



What is the gap Δ between the batches?

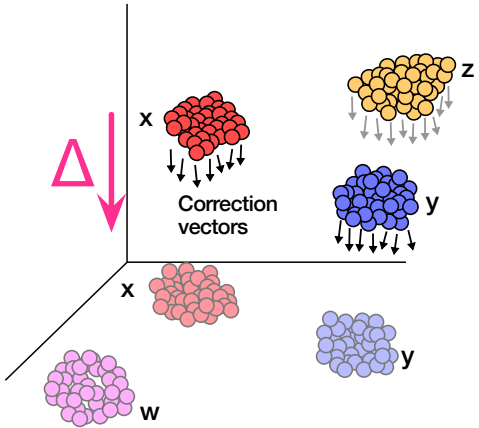
$$\min_{\Delta} \sum_{a,b} W_{ab} \|\mathbf{x}_a - \mathbf{x}_b - \Delta\|_2$$

Assume that the similarity between cells

$$0 \approx W_{ac} < W_{ab}$$

Haghverdi, .., and Marioni, *Nature Biotechnology* (2018)

Batch normalization aims to minimize the difference between nearest cells across different batches



What is the gap Δ between the batches?

$$\min_{\Delta} \sum_{a,b} W_{ab} \|\mathbf{x}_a - \mathbf{x}_b - \Delta\|_2$$

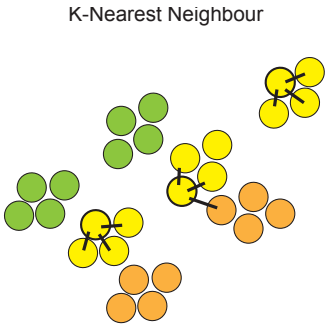
Fixed point (local) optimal solution:

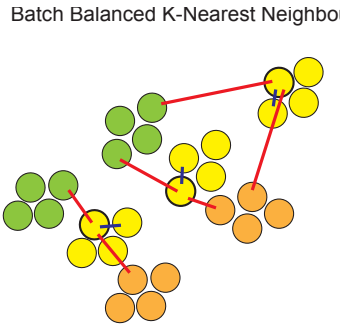
$$\Delta \leftarrow \frac{\sum_{a,b} W_{ab}(\mathbf{x}_a - \mathbf{x}_a)}{\sum_b W_{ab}}$$

Haghverdi, .., and Marioni, *Nature Biotechnology* (2018)

A batch-balancing k-nearest neighbour graph

BBKNN method strikes balance between over- and under-normalization



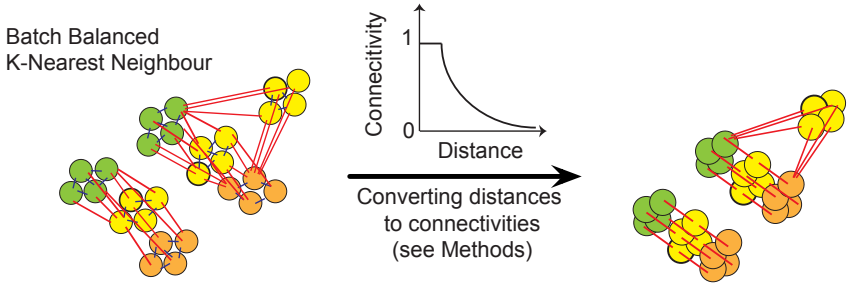


What kind of differences in due to inter-batch, technical discrepancy, not inter-cell-type divergence?

Polanski, .., Teichmann *Bioinformatics* (2019)

A batch-balancing k-nearest neighbour graph

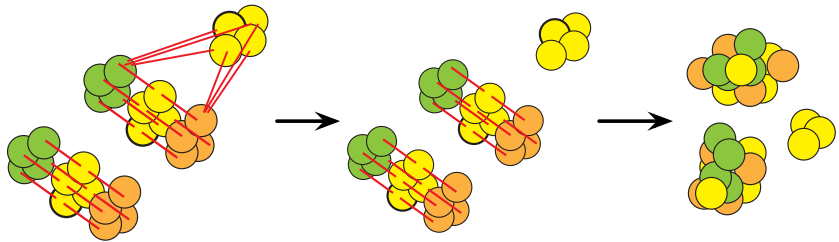
BBKNN method strikes balance between over- and under-normalization



Polanski, .., Teichmann *Bioinformatics* (2019)

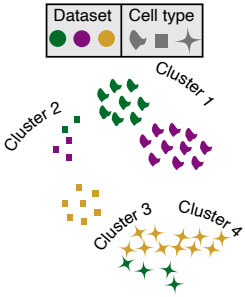
A batch-balancing k-nearest neighbour graph

BBKNN method strikes balance between over- and under-normalization



Polanski, .., Teichmann *Bioinformatics* (2019)

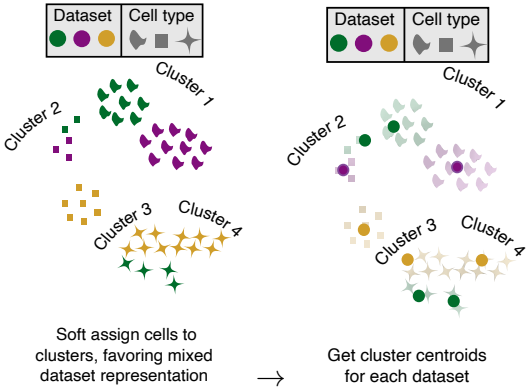
Harmony: clustering-based data normalization



Soft assign cells to clusters, favoring mixed dataset representation

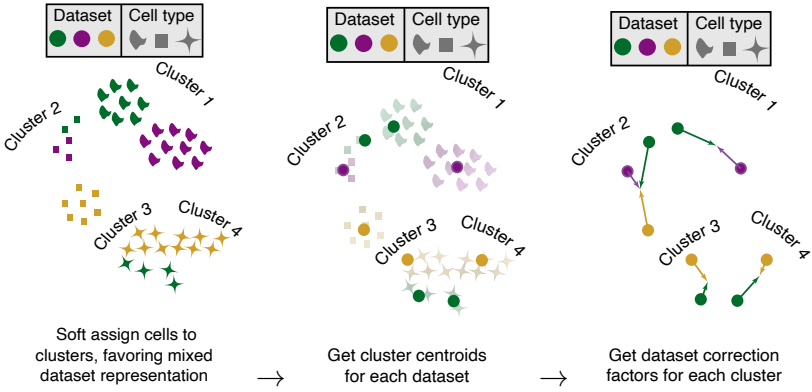
Korsunsky, .., Loh, Raychaudhuri *Nature Methods* (2019)

Harmony: clustering-based data normalization



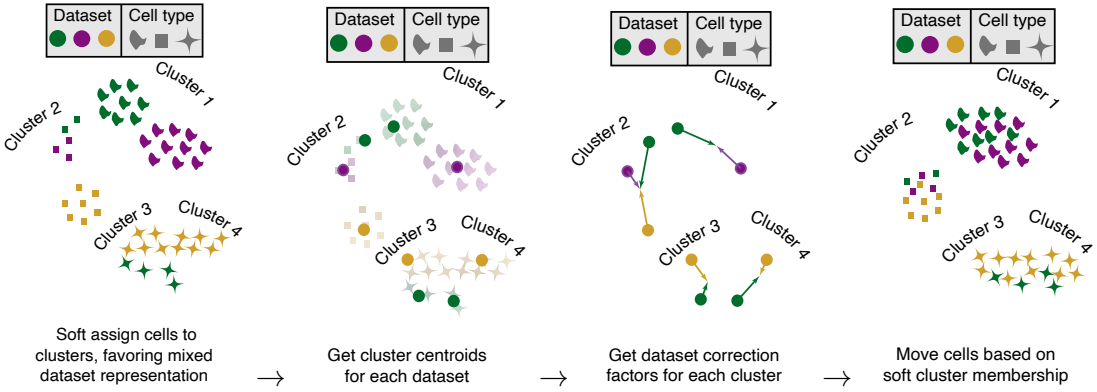
Korsunsky, .., Loh, Raychaudhuri *Nature Methods* (2019)

Harmony: clustering-based data normalization



Korsunsky, .., Loh, Raychaudhuri *Nature Methods* (2019)

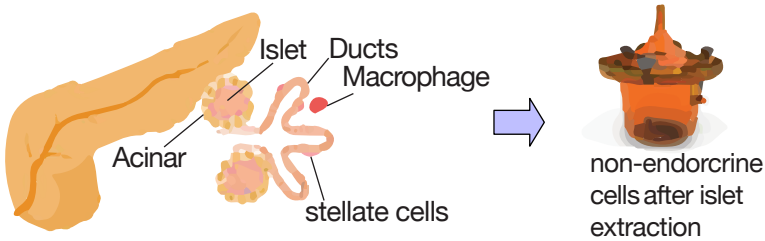
Harmony: clustering-based data normalization



Korsunsky, .., Loh, Raychaudhuri *Nature Methods* (2019)

Example: another scRNA-seq data on human pancreatic islet cells

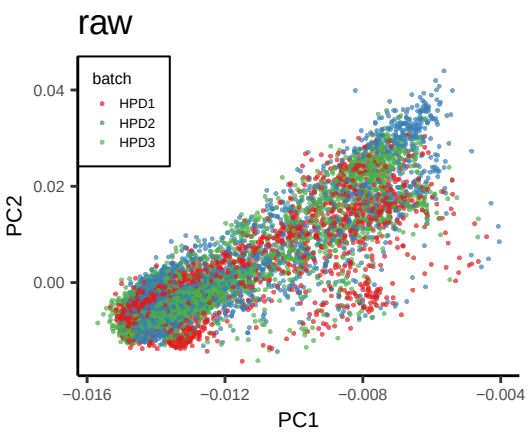
Single-cell RNA-seq data from three donors (three batches)

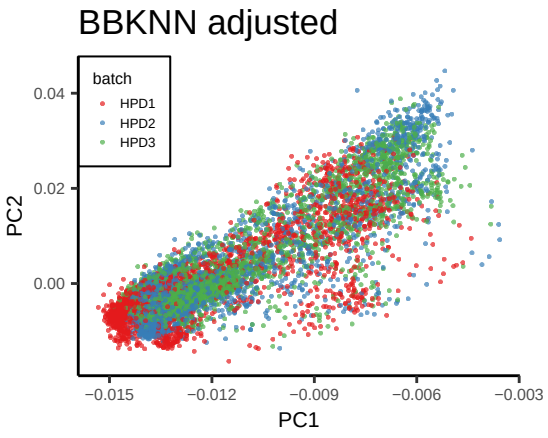


Goal: Remove potential batch effects across different donors. (1) Construct BBKNN graphs between cells; (2) compute average discrepancy Δ between batches in the PC space; (3) adjust them.

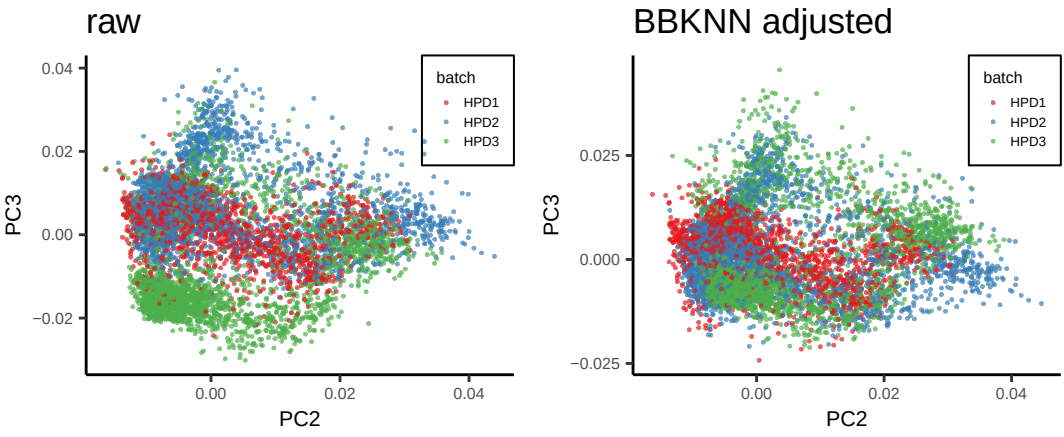
Qadir, *et al.*, *PNAS* (2020)

BBKNN-guided normalization (Δ) adjusts the inter-batch discrepancy

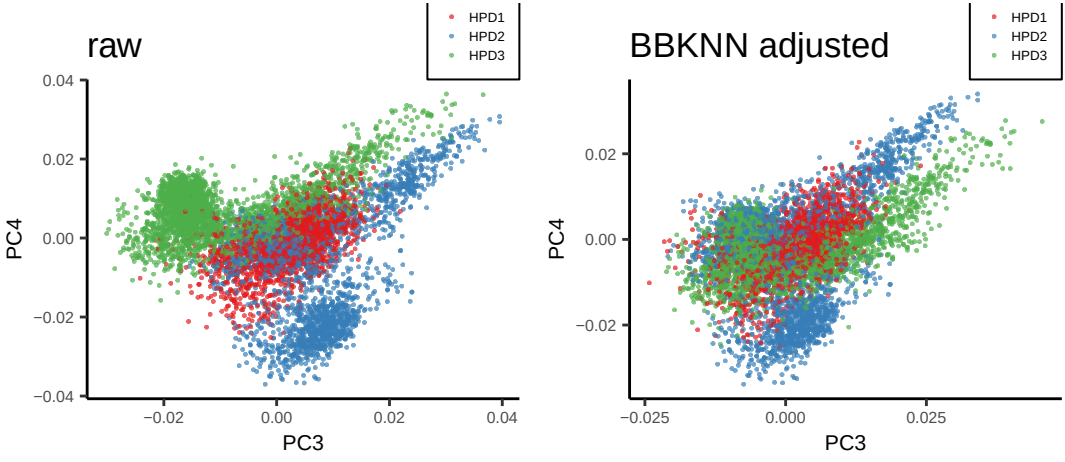




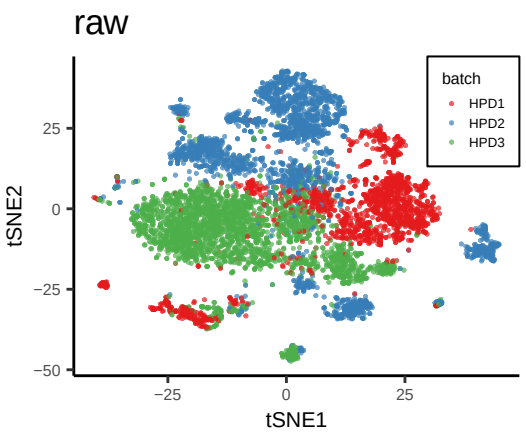
BBKNN-guided normalization (Δ) adjusts the inter-batch discrepancy

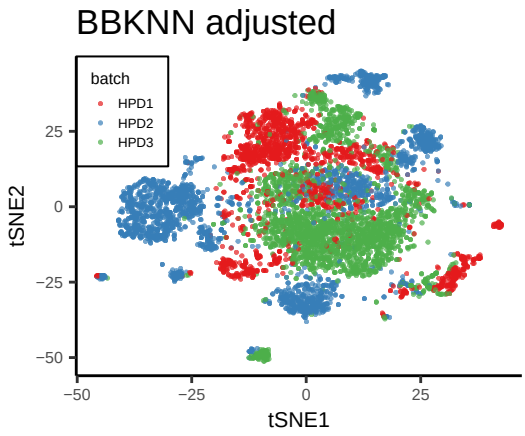


BBKNN-guided normalization (Δ) adjusts the inter-batch discrepancy



BBKNN-guided normalization (Δ) adjusts the inter-batch discrepancy





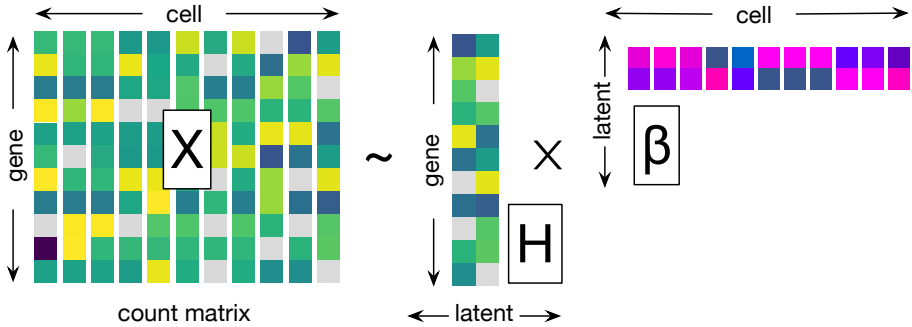
Discussions

- What can we do with BBKNN graphs?
- Why do we need batch normalization?
- Is it possible to over-correct the differences?
- Is it also possible to under-correct the differences?

Today's lecture

- Single-cell sequencing technology
- Basic Data Q/C
- Doublet detection in single-cell data
- Data normalization across many batches
- Latent topic modelling
- Other interesting topics in scRNA-seq analysis

What is a generative model of single-cell data?



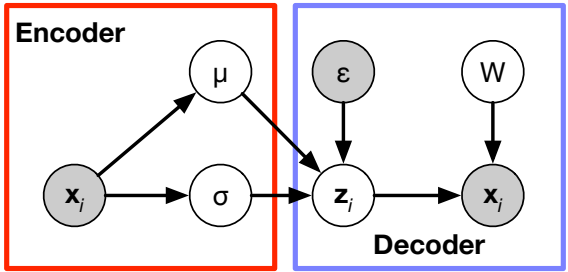
$$\mathbb{E}[X] \approx f\left(H\beta\right)$$

Can we assign tens of thousands of cells to some hidden probability space (H)?

Why do we need unsupervised learning for single-cell RNA-seq data?

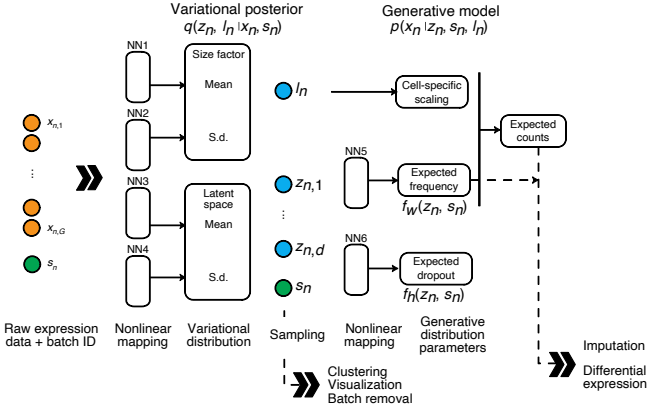
- Probabilistic interpretation of latent states
- Incomplete single-cell data, lots of drop-out measurements
- We can design generative model parameters as interpretable as possible!

Variation autoencoder (VAE): a Bayesian inference framework for easy/scalable inference of latent variable model



- Define relationships between variables (auto generative process)
- Usually, the decoder side captures our scientific hypothesis
- We can use an "auto-diff" algorithm (e.g., Facebook `torch` or Google `tensorflow`) to calculate gradients for the model parameters to optimize.

Deep generative modeling for single-cell transcriptomics

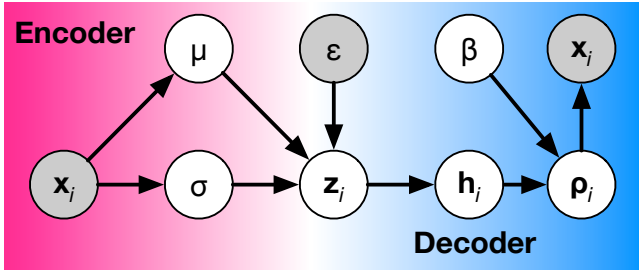


Generative model: zero-inflated negative binomial distribution

Lopez, .., Jordan, Yosef, *Nature Methods* (2018)

Multinomial topic modelling for (incomplete) single-cell expression data

Can we simply model scRNA-seq counts by multinomial distribution?



- X_{ig}: gene expression of a gene g in a single cell i
- H_{ik}: latent topic proportion of a cell i to a topic k
- β_{kg}: topic k-specific gene probability

Multinomial topic modelling for (incomplete) single-cell expression data

Can we simply model scRNA-seq counts by multinomial distribution?

Likelihood model:

$$\mathcal{L} = \prod_{i=1}^{n} \prod_{g=1}^{\text{genes}} \left(\sum_{k} H_{ik} \beta_{kj} \right)^{X_{ij}}$$

- X_{ig}: gene expression of a gene g in a single cell i
- H_{ik}: latent topic proportion of a cell i to a topic k
- β_{kg}: topic k-specific gene probability

Multinomial topic modelling for (incomplete) single-cell expression data

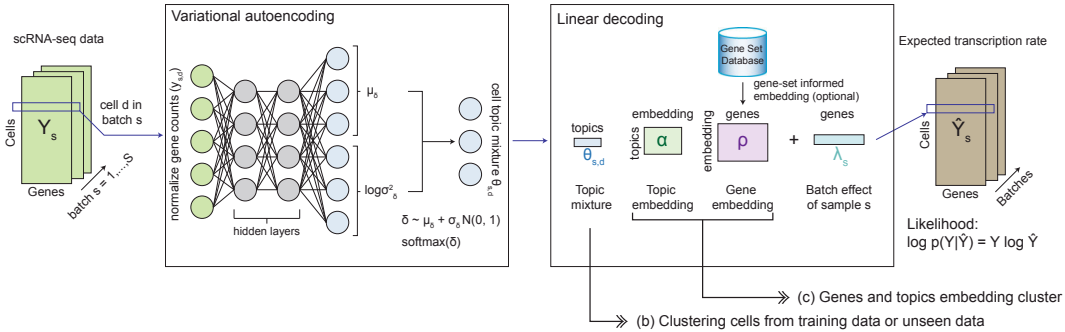
Can we simply model scRNA-seq counts by multinomial distribution?

Likelihood model:

$$\mathcal{L} = \prod_{i=1}^{n} \prod_{g=1}^{\text{genes}} \left(\underbrace{\sum_{k} H_{ik}\beta_{kj}}_{\text{a gene } g\text{'s probability in a cell } \iota \equiv \rho_{ig}} \right)^{X_{ij}}$$

- X_{ig}: gene expression of a gene g in a single cell i
- H_{ik}: latent topic proportion of a cell i to a topic k
- β_{kg}: topic k-specific gene probability

Single-cell Embedded Topic Model



Zhao, Cai, .., Li, *Nature Comm.* (2021)

Topic Modelling: Compare between document vs. single-cell

We think of a cell as a document, which is $\approx$ a bag of words, or $\approx$ a bag short mRNA reads.

variables	in document topic model	in single cell ETM
D	Total number of documents (corpus)	Total number of cells
d	Document index	Cell index
N_d	Number of words in a document d	Number of read counts in a cell d
j	Word index, $j \in [N_d]$	Read index
K	Total number of topics	Total number of cell type topics
k	Topic index, $k \in [K]$	Cell topic index
V	Size of vocabulary	Total number of genes
v	Vocabulary index $v \in [V]$	Gene index
W_{dj}^v	Indicator for a word to vocabulary $\in \{0, 1\}$	Indicator for a read to a gene $\in \{0, 1\}$
X_{dv}	Vocabulary v occurrence in a document d	Gene expression of a gene v in a cell $d \in [0, N_d]$

$W_{dj}^v = 1$ if and only if a word j in a document d takes v-th word in the vocabulary; otherwise, $W_{dj}^v = 0$.

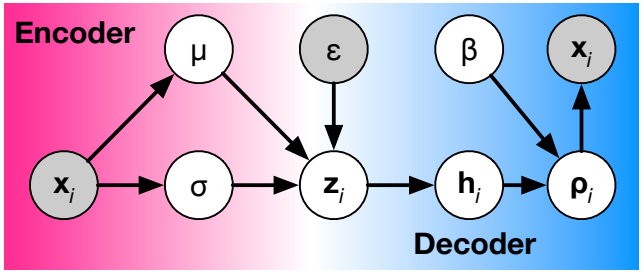
Single-cell Embedded topic model's latent states and model parameters

variables	in document topic model	in single cell ETM
Z_{dj}^k	Indicator for assigning a word to a topic k	Indicator for assigning a read to a topic k
H_{dk}	Hidden state k of a document d	Hidden state k of a cell d
β_{kv}	topic k-specific vocabulary v frequency	topic k-specific, a gene v's exression

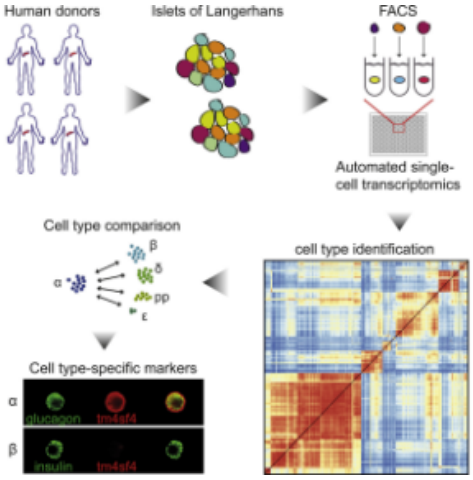
- In Latent Dirichlet Allocation: $\sum_{k=1}^{K} H_{dk} = 1$ and $H_{dk} > 0$, and $\mathbf{h}_d \sim \text{Dirichlet}(\alpha/K, \ldots, \alpha/K)$ *a priori*. Approximately, we have $\hat{H}_{dk} = \sum_j^{N_d} Z_{dj}^k / N_d$.
- In Embedded Topic model, H_{dk} with the simplex constraints; $H_{dk} = \exp(\delta_{dk}) / \sum_{k'} \exp(\delta_{dk'})$ where $\delta_{dk} \sim \mathcal{N}(0, 1)$ *a priori*.
- Additional constraints: $\beta_{kv} > 0$ and $\sum_v \beta_{kv} = 1$, meaning that only a handful of vocabulary v contribute to a topic k.

Multinomial topic modelling for (incomplete) single-cell expression data

Can we simply model scRNA-seq counts by multinomial distribution?



Example: single-cell RNA-seq data of human pancreatic cells



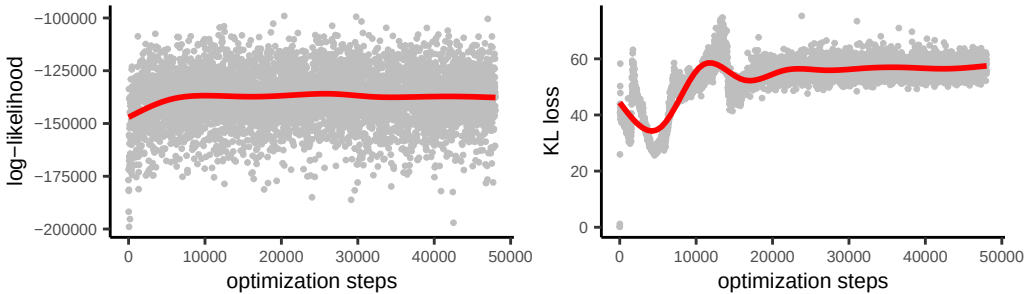
Muraro *et al. Cell Systems* (2016)

We will use scRNA-seq data (GEO accession: GSE85241) as a working example.

- genes/features/rows: 19,140
- cells/columns: 3,072
- non-zero elements: 12,442,034
- $\sim$ 21 % non-zero

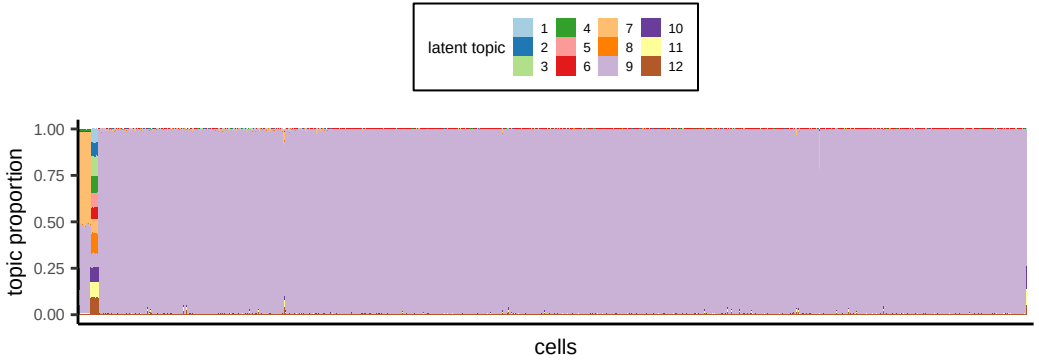
Variational inference $\approx$ maximum likelihood regularized by a KL-divergence term

$$\underbrace{\mathbb{E}[\log p(\mathbf{x}|\mathbf{h}(\mathbf{z}))]}_{\text{expected data likelihood}} - \underbrace{\mathbb{E}[\log q(\mathbf{z})/p(\mathbf{z})]}_{\text{KL loss}}$$



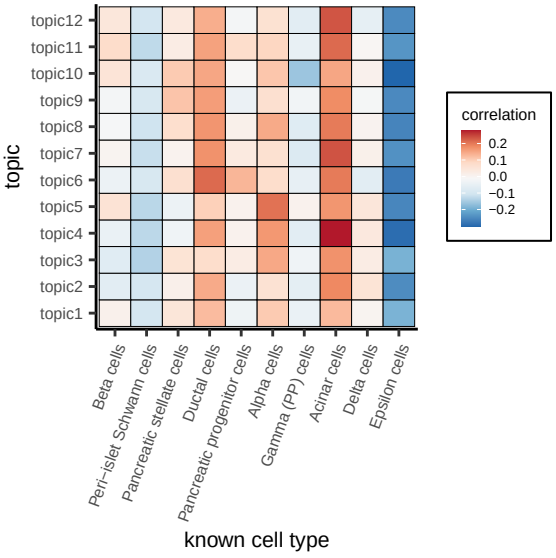
We may need to train longer than usual... (don't be fooled by log-likelihood)

ETM learning just started ... (hidden states $\mathbf{h}$)



There is no obvious pattern... yet

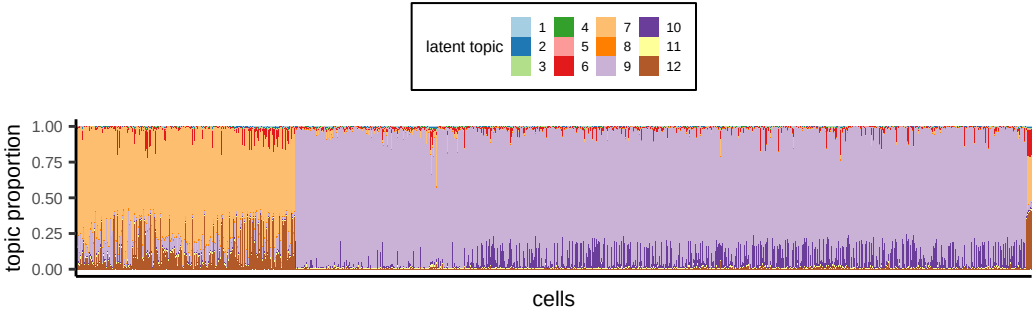
ETM learning just started ... (weight parameters β)



- We can correlate each topic-specific gene $\times 1$ weight vector, β_k, with known cell type-specific marker genes
- No obvious concepts emerged yet, not so specific correlation patterns, yet...

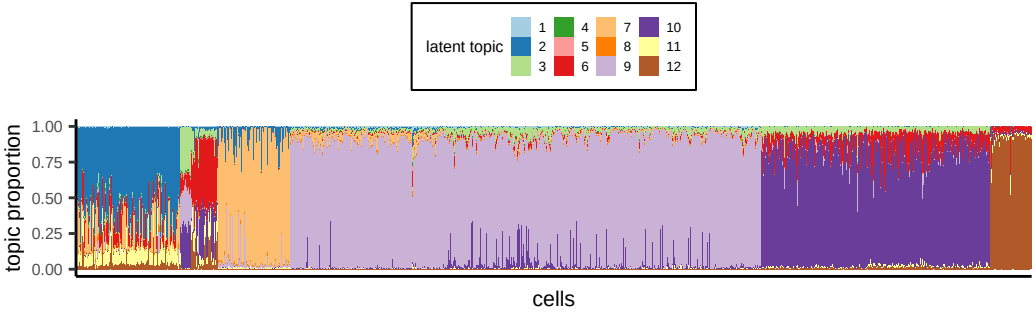
If we keep on training ETM (hidden states) ..

epoch = 300



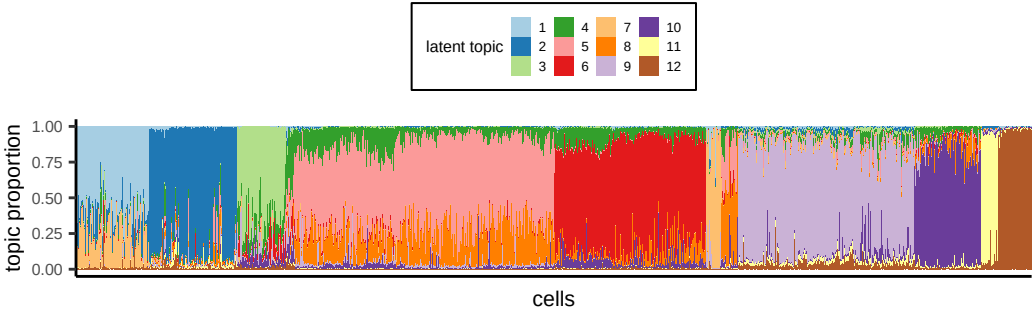
If we keep on training ETM (hidden states) ..

epoch = 600



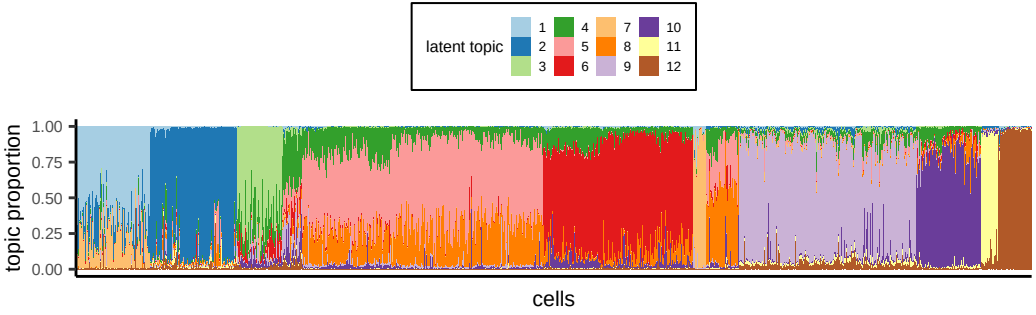
If we keep on training ETM (hidden states) ..

epoch = 900

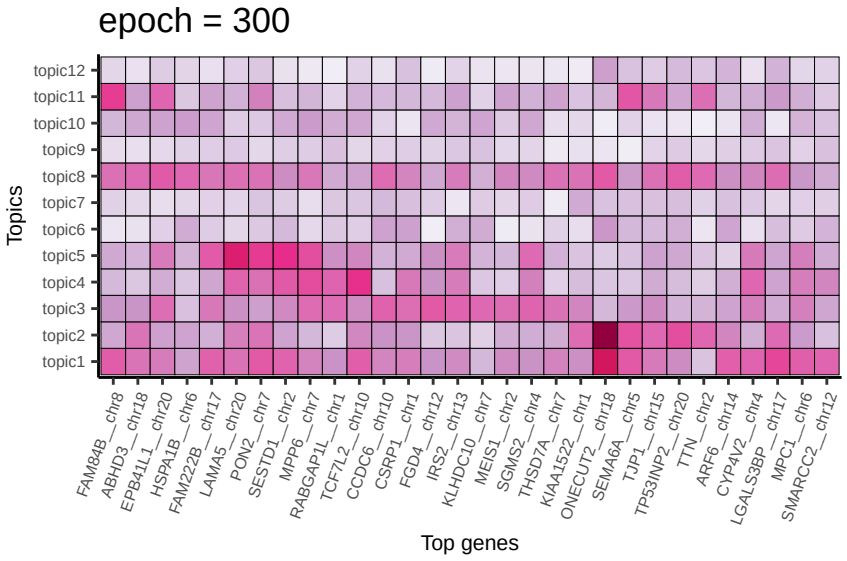


If we keep on training ETM (hidden states) ..

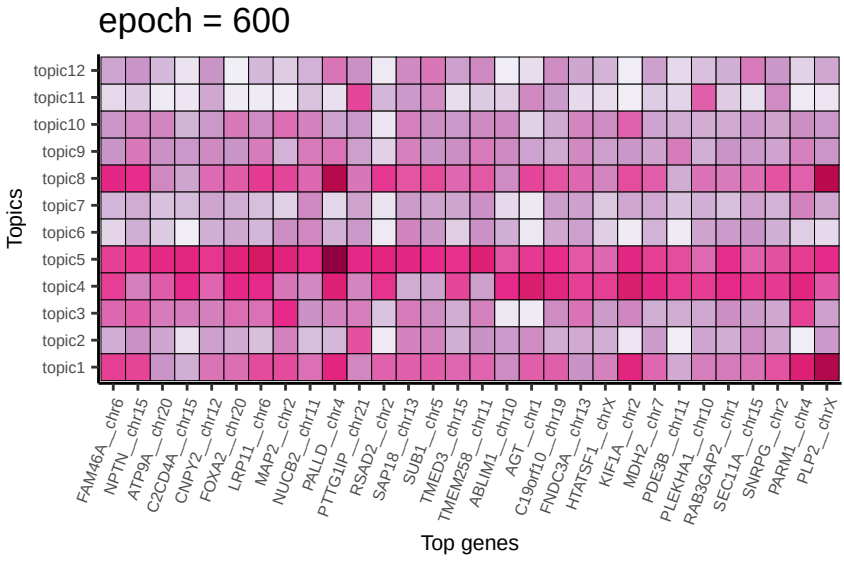
epoch = 1200



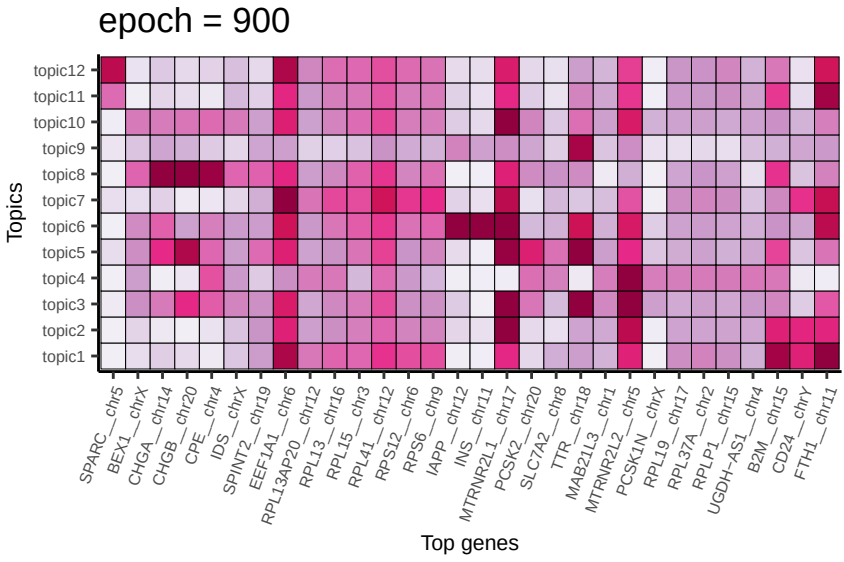
If we keep on training ETM (weight parameters) ...



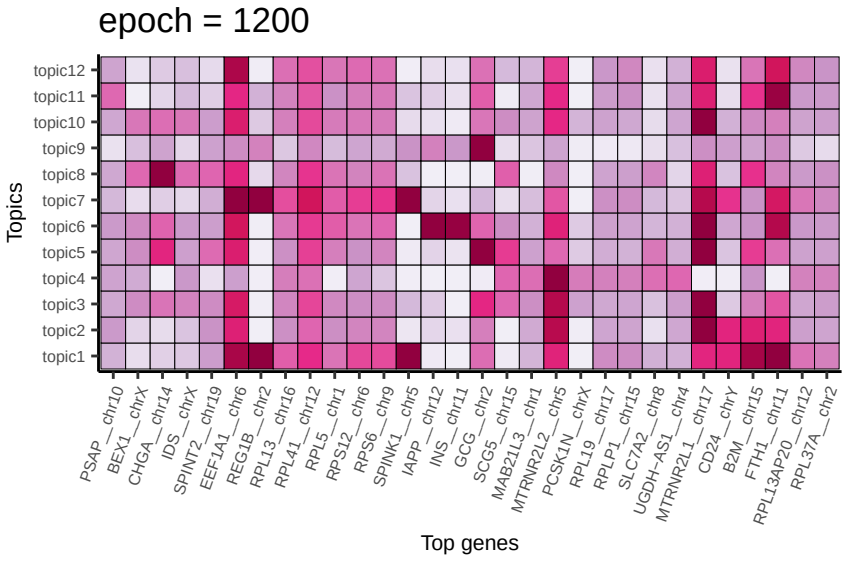
If we keep on training ETM (weight parameters) ...



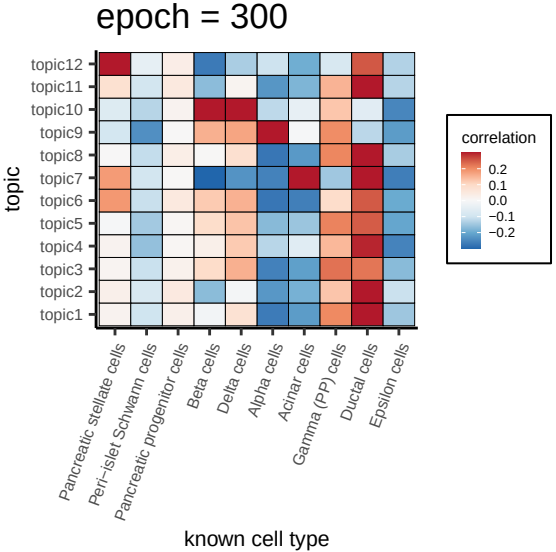
If we keep on training ETM (weight parameters) ...



If we keep on training ETM (weight parameters) ...

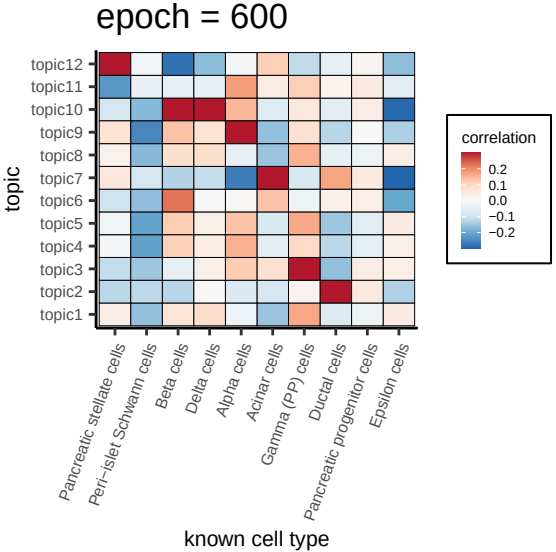


If we keep on training ETM (weight parameters) ...



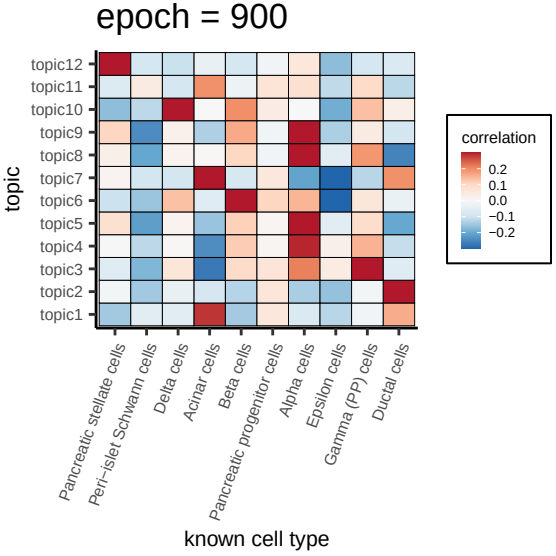
- We can correlate each topic-specific gene $\times 1$ weight vector, β_k, with known cell type-specific marker genes
- We retrieved marker gene information of known cell types from PangaloDB

If we keep on training ETM (weight parameters) ...



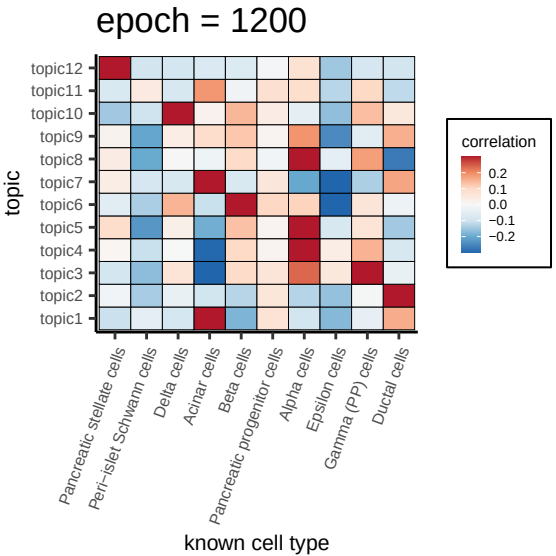
- We can correlate each topic-specific gene $\times 1$ weight vector, β_k, with known cell type-specific marker genes
- We retrieved marker gene information of known cell types from PangaloDB

If we keep on training ETM (weight parameters) ...



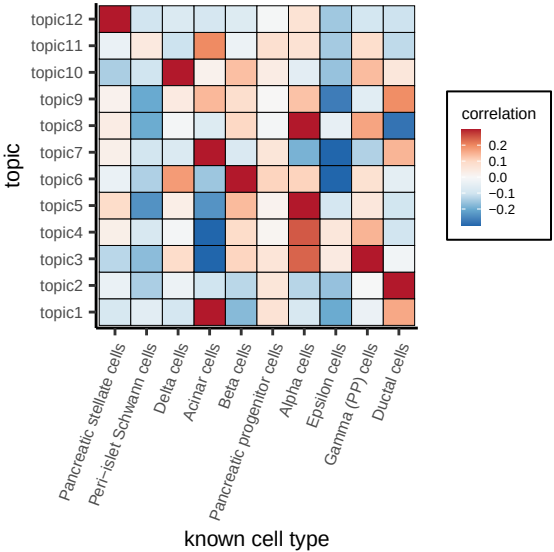
- We can correlate each topic-specific gene $\times 1$ weight vector, β_k, with known cell type-specific marker genes
- We retrieved marker gene information of known cell types from PangaloDB

If we keep on training ETM (weight parameters) ...



- We can correlate each topic-specific gene $\times 1$ weight vector, β_k, with known cell type-specific marker genes
- We retrieved marker gene information of known cell types from PangaloDB

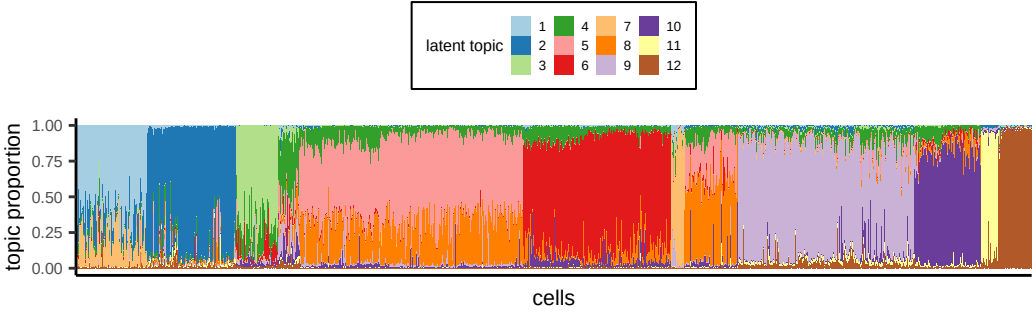
After enough training steps...



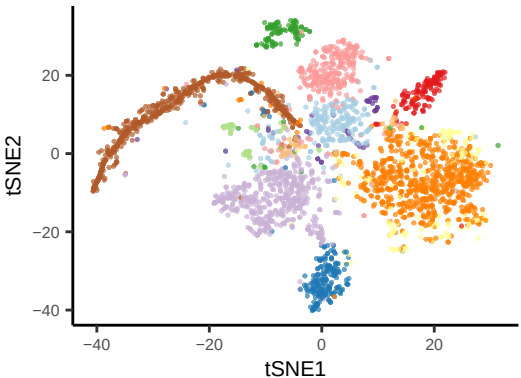
- We have recapitulated most known Pancreatic cell types in our single-cell analysis

Single-cell ETM effectively learns cellular admixture model

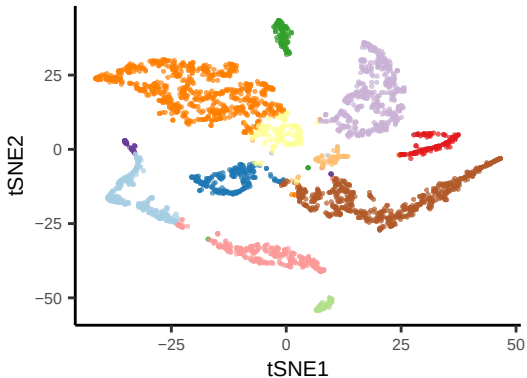
epoch = 2010



using top 50 PCs

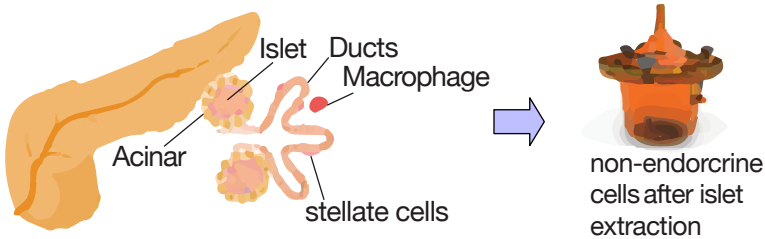


tSNE on the latent topic space



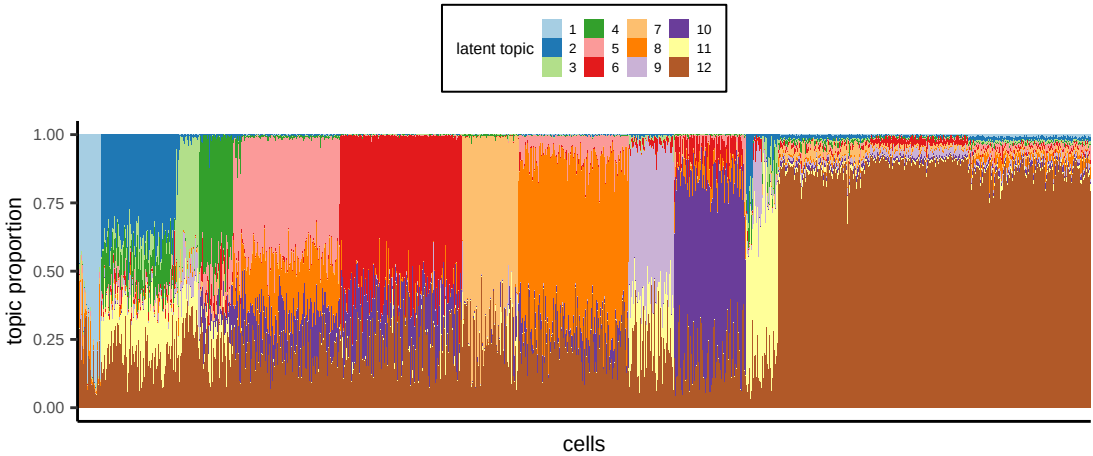
A latent topic model robustly capture cell states, avoiding batch effects

Single-cell RNA-seq data from three donors (three batches)

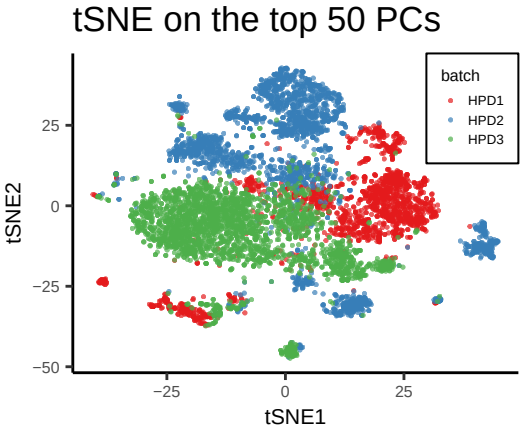


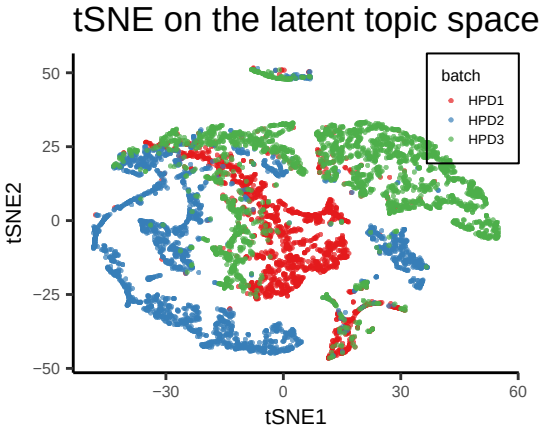
Goal: Topic space for 6,873 cells shared across multiple donors (batches).

Qadir, *et al.*, *PNAS* (2020)


latent topic
1
2
3
4
5
6
7
8
9
10
11
12
1.00
0.75
0.50
0.25
0.00
topic proportion
cells

Multiple batches are well mixed!





Discussions on latent topic modelling

- Most cells predominantly belong to one topic (one colour). Why?
- If we model cells as a mixture of cell topics, we can capture doublets or triplets
- The underlying generative model assumes no sequencing depth! This can help avoid batch-specific differences in practice.
- VAE offers a flexible framework with which our scientific hypothesis can be formulated in a probabilistic language
- Potentially, this pure unsupervised learning framework can be combined with supervised, semi-supervised learning models.

Today's lecture

- Single-cell sequencing technology
- Basic Data Q/C
- Doublet detection in single-cell data
- Data normalization across many batches
- Latent topic modelling
- Other interesting topics in scRNA-seq analysis

Other topics that we don't have time to discuss now

1. Differential expression analysis
2. RNA velocity and pseudo-time analysis
3. Multiomics data integration
4. Spatial transcriptomics
5. Joint analysis with bulk sequencing data

Summary

- Single-cell RNA-seq technology
- Doublet finding and other Q/C in scRNA-seq anlaysis
- Data normalization across multiple batches
- Model-based latent representation identification