Bioinformatics Software Lab
Introduction to Analysis of Single Cell Sequencing

Ivan Gesteira Costa, James Nagai, Martin Manolov, Kai Peng, Mehdi Joodaki
Institute for Computational Genomics

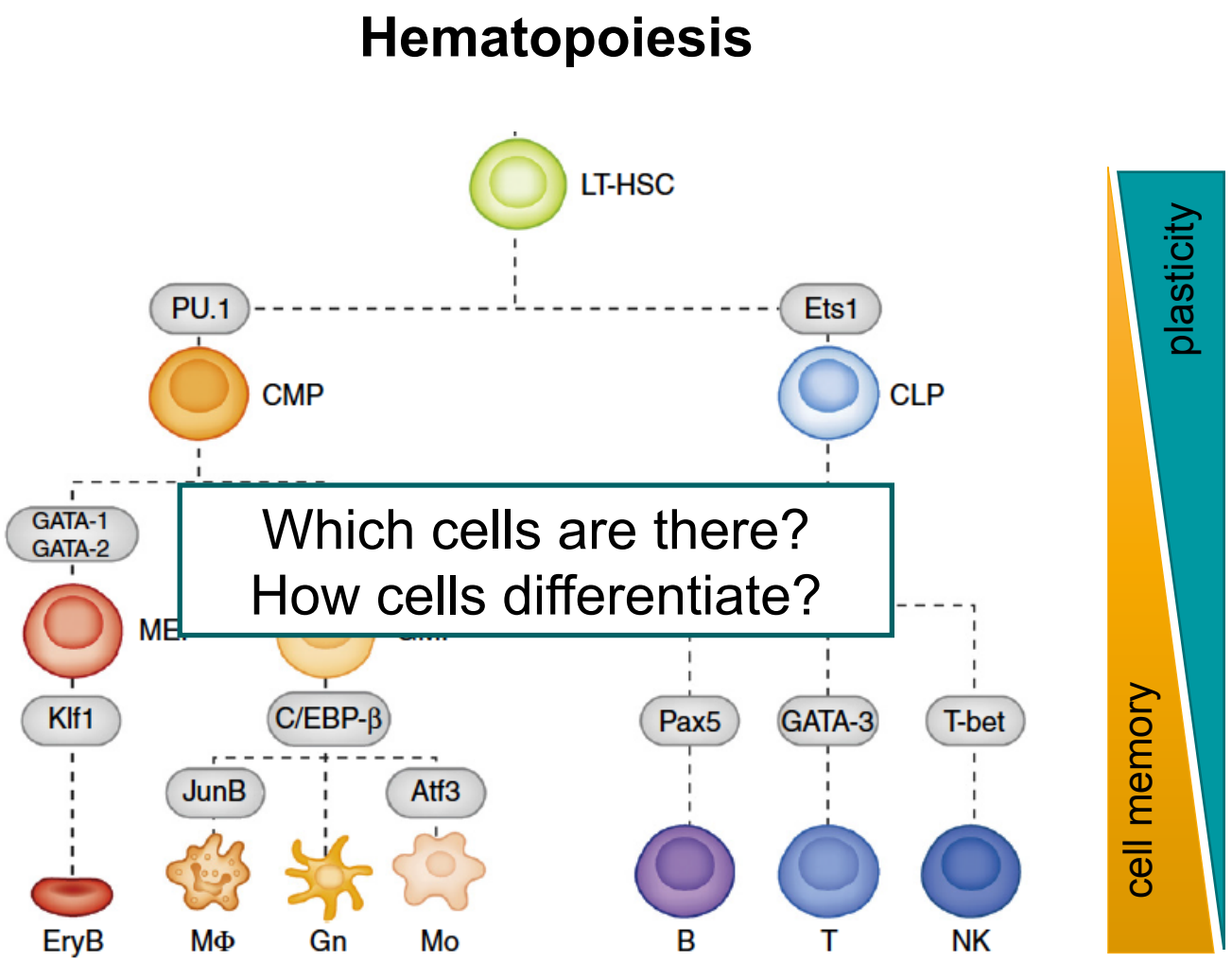
Institute for Computational Genomics
RWTH AACHEN UNIVERSITY

Objectives

1. basics of single cell sequencing
2. basic bioinformatics/computational problems
   - dimension reduction
   - clustering
   - data integration

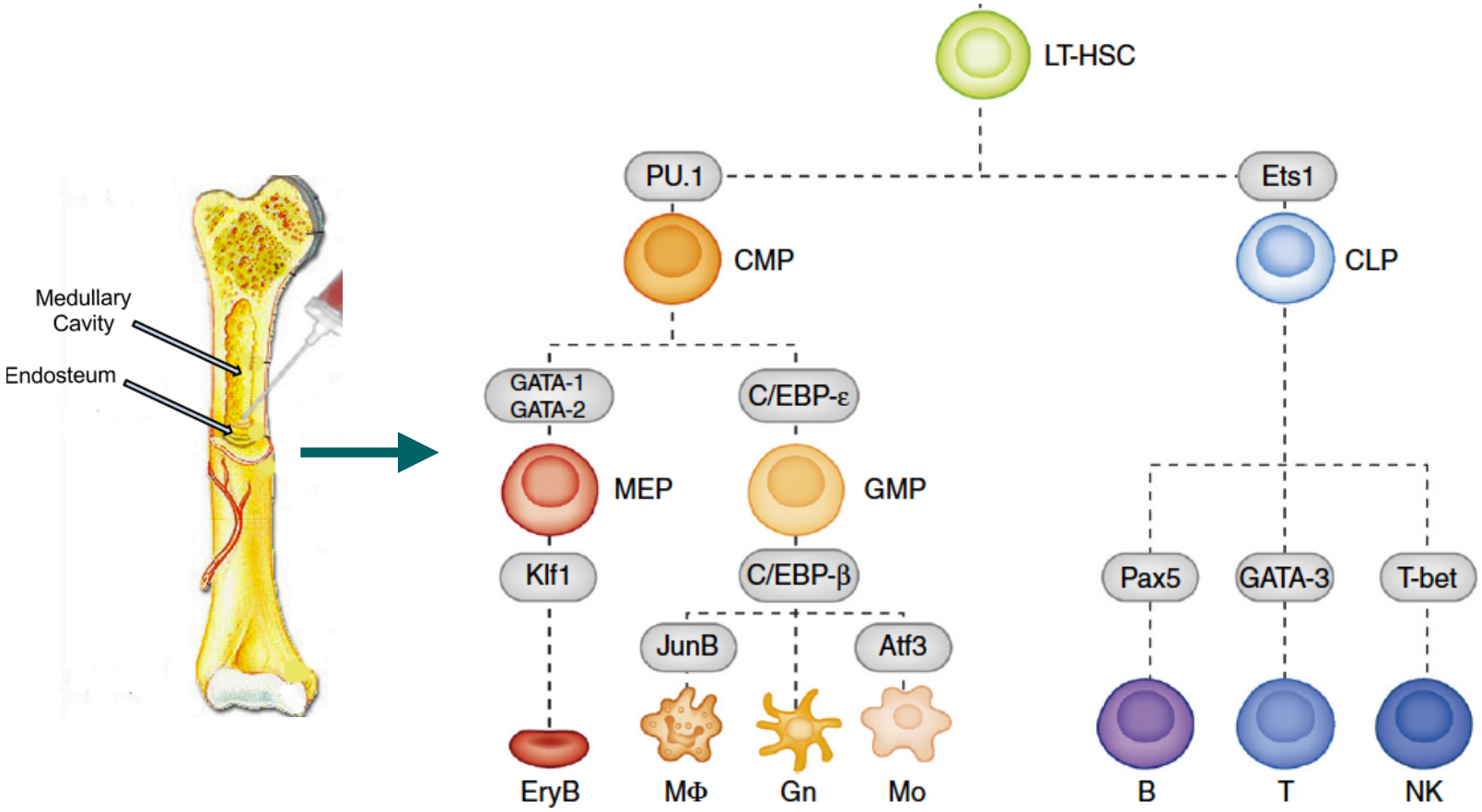
Expression at Single Cell Level

Cell Differentiation



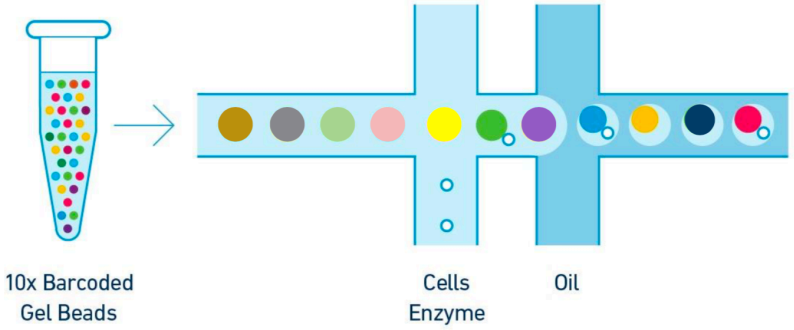
Source: Amit (2016), *Nature Immunoloy.*

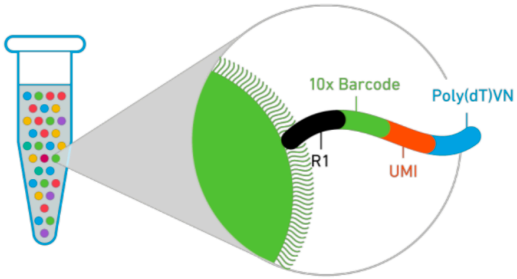
Cell Differentiation

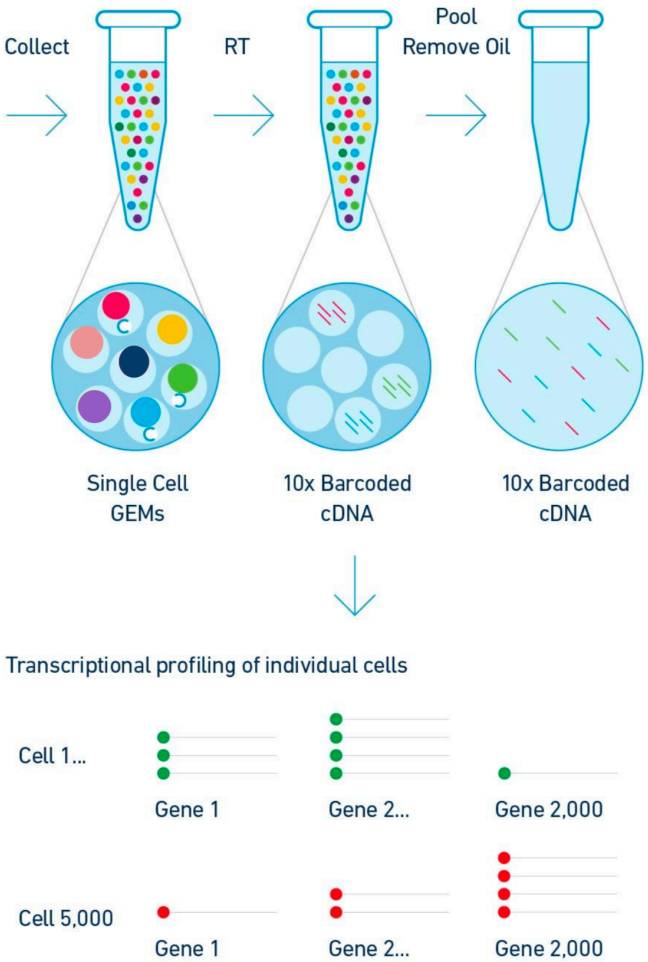


Source: Amit (2016), *Nature Immunoloy.*

Droplet based RNA single cell sequencing



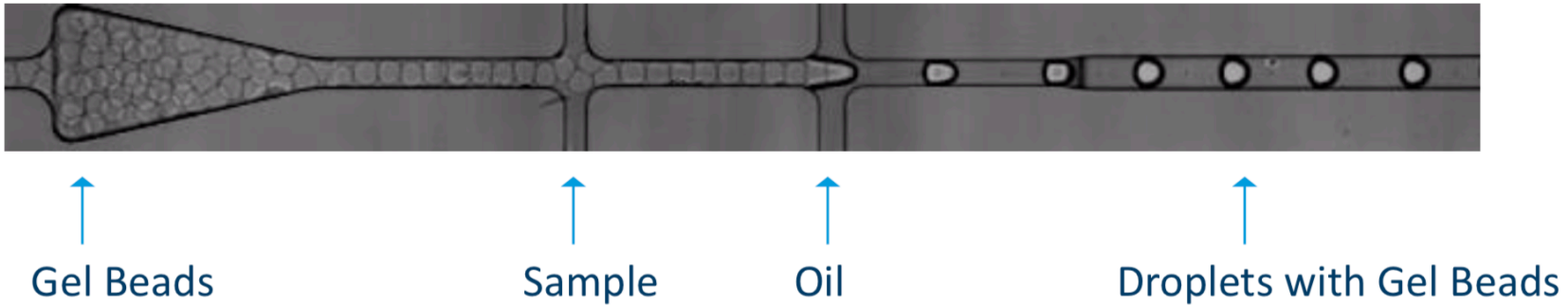


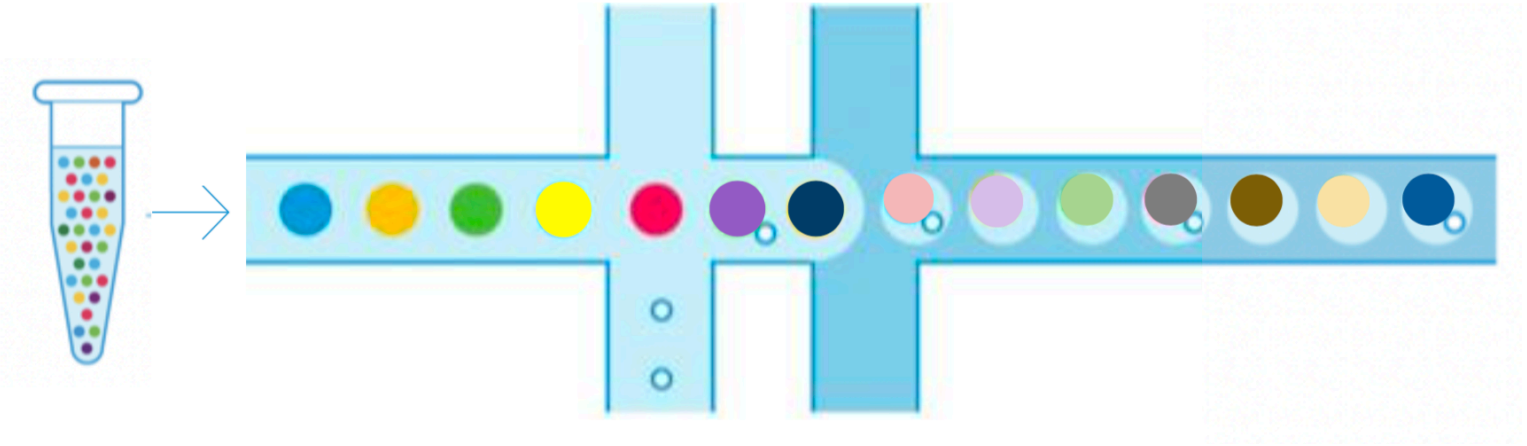


- Input: Single cells in suspension + 10x Gel Beads and Reagents
- Output: Digital gene expression profiles from every partitioned cell

Source: 10x genomics

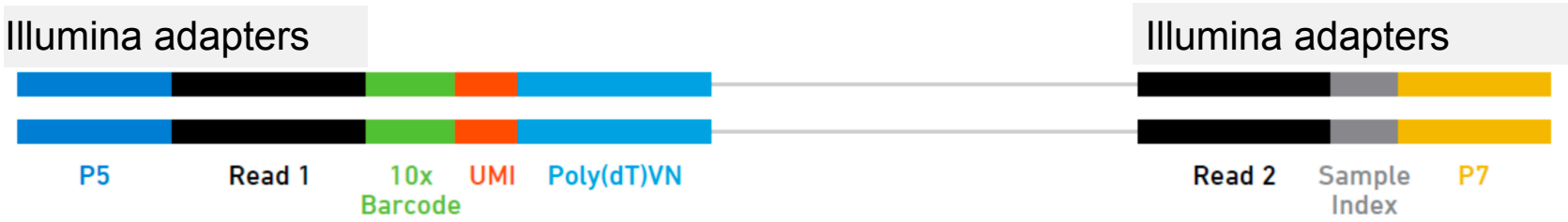
Droplet based RNA single cell sequencing





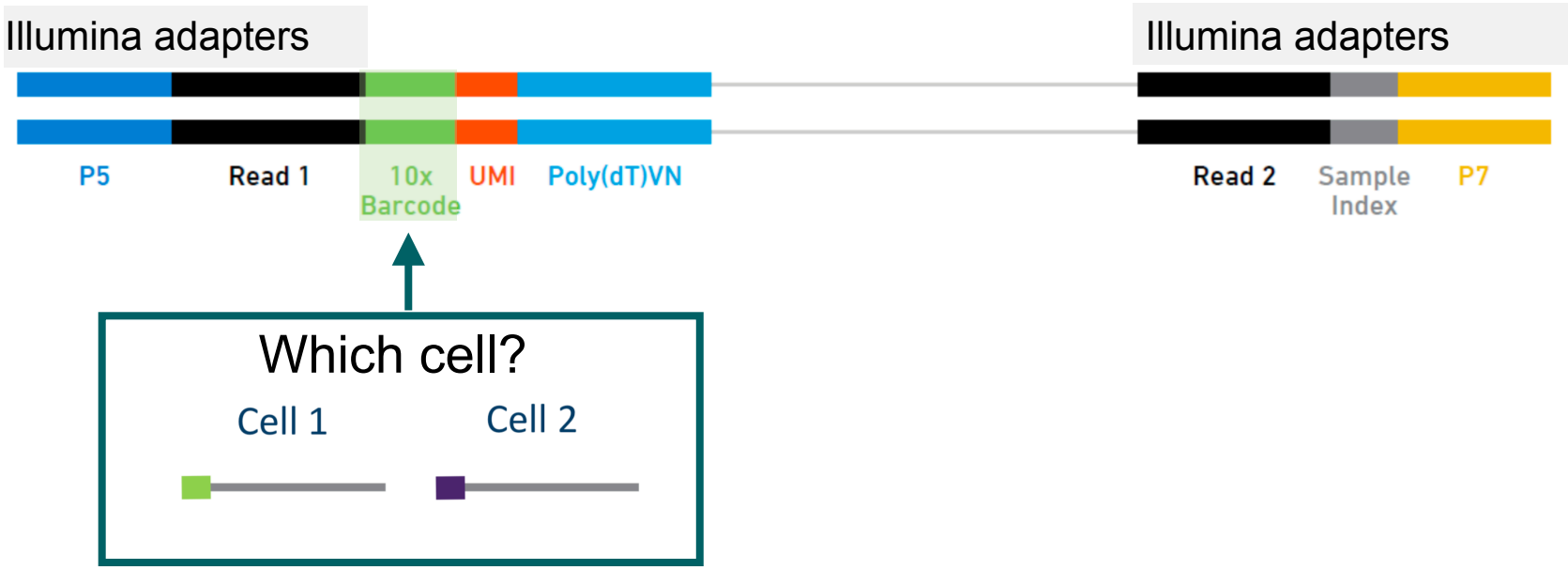
Source: 10x genomics

Basics Bioinformatics - Transcript Counts



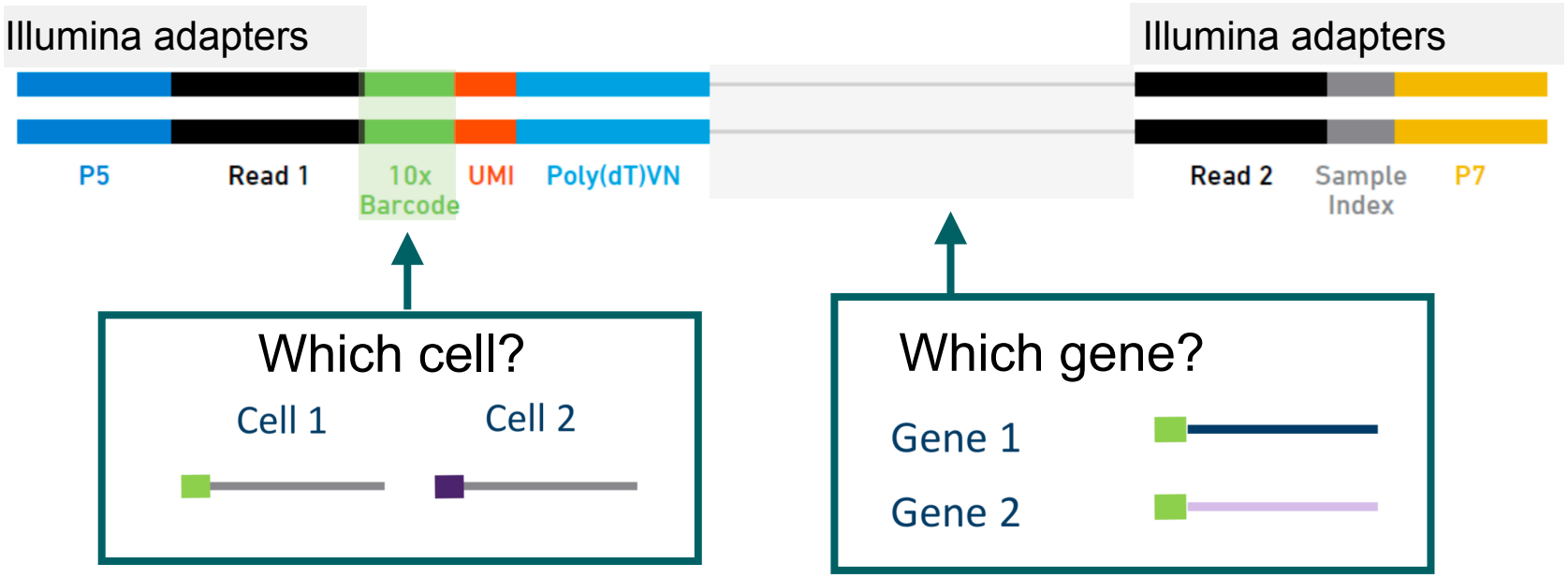
Source: 10x genomics

Basics Bioinformatics - Transcript Counts



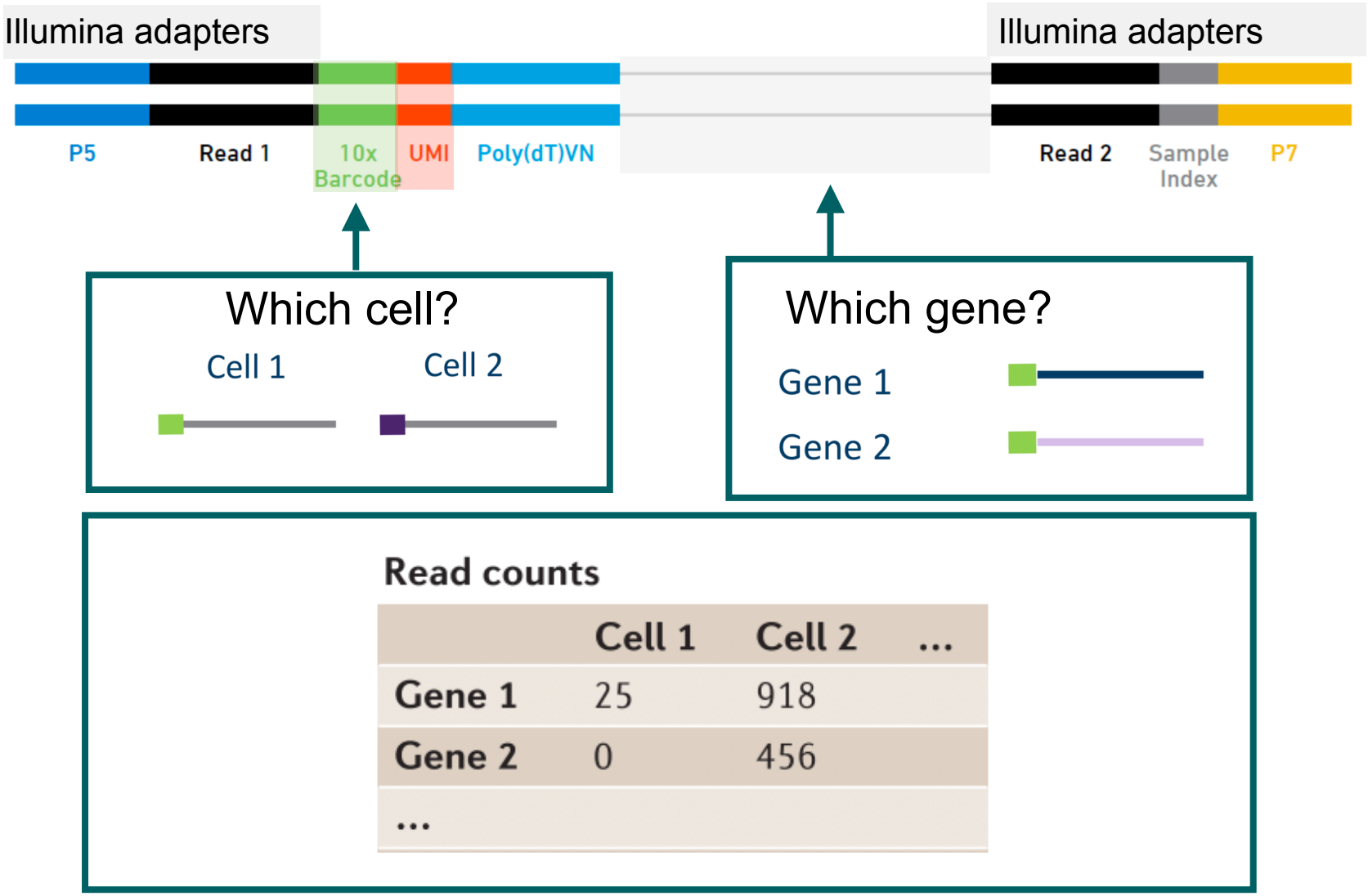
Source: 10x genomics

Basics Bioinformatics - Transcript Counts



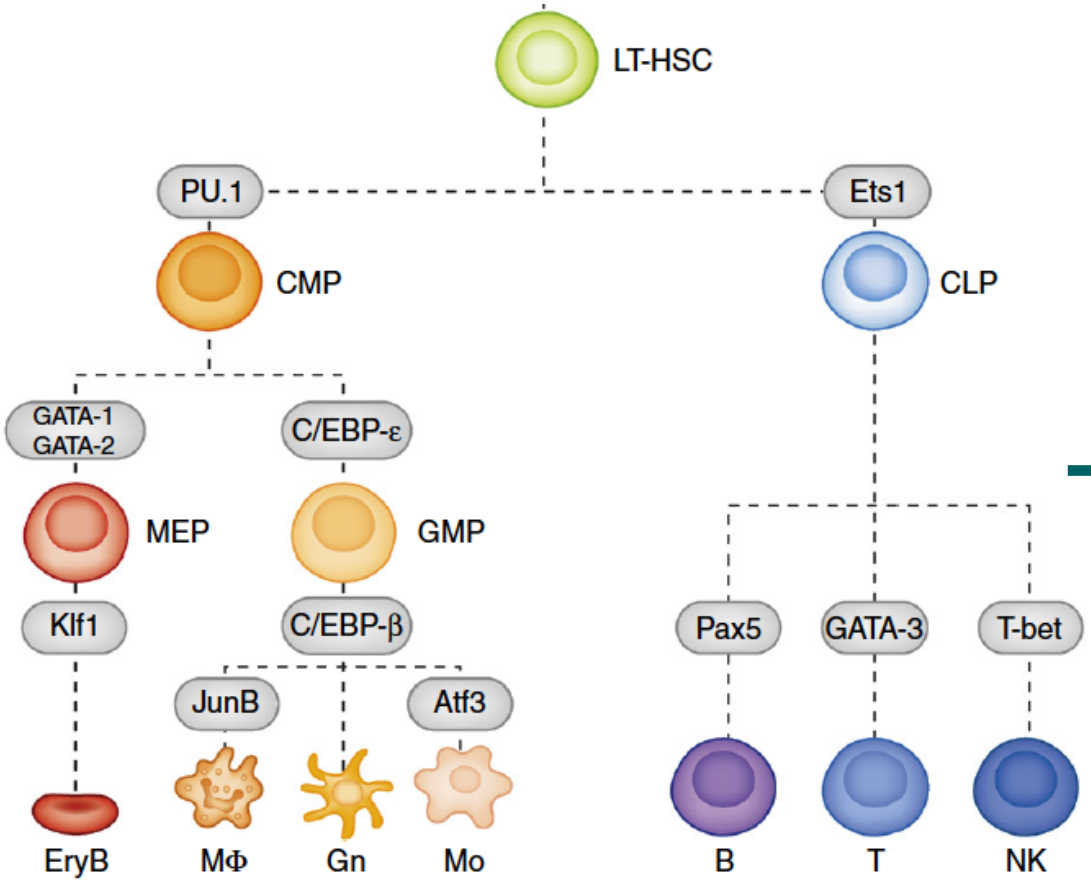
Source: 10x genomics

Basics Bioinformatics - Transcript Counts



Source: 10x genomics

Cell Differentiation & Gene Expression

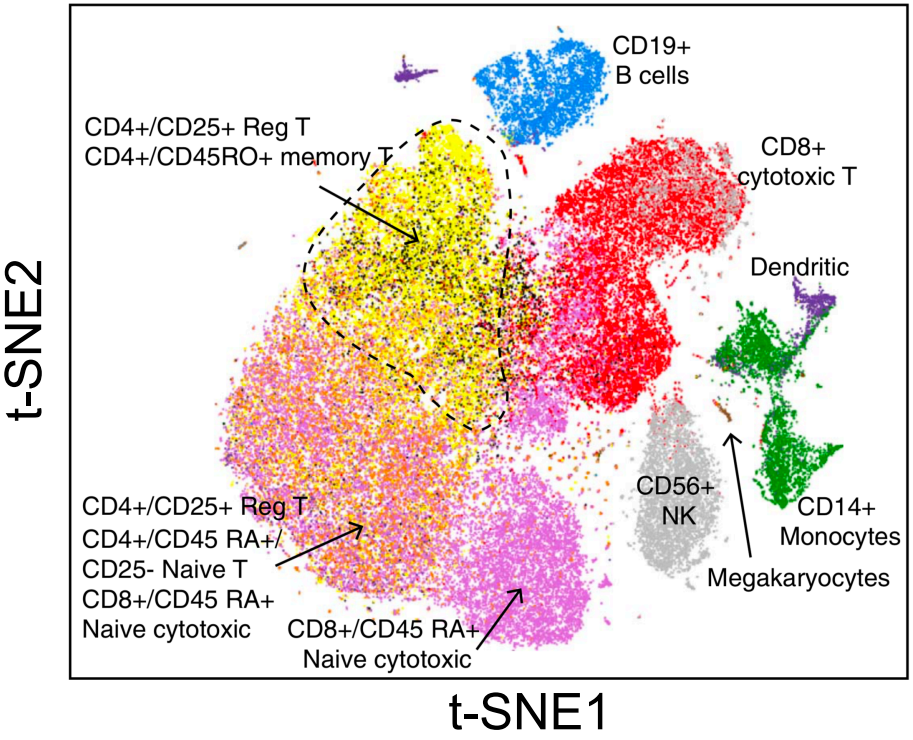


	Cell 1	Cell 2	...
Gene 1	25	918	
Gene 2	0	456	
Gene 3	20	342	
Gene 4	0	214	
...			

Source: Amit (2016), *Nature Immunoloy.*

Gene Expression of Lymphoid Cells

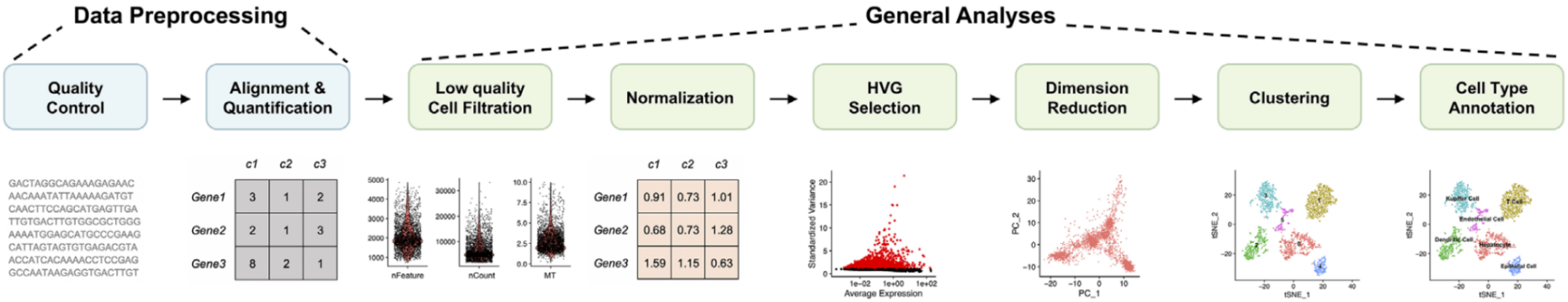
PBMCs from Humans



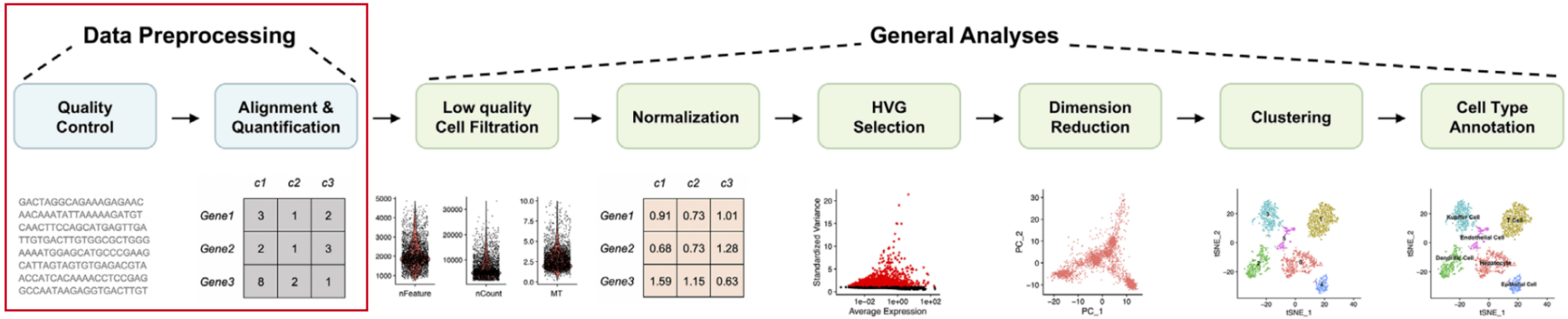
Single cell RNA-seq from 68k cells

Source: Zheng et al. 2017 & Buenrostro et al. 2018

Basics Bioinformatics - single cell RNA-seq

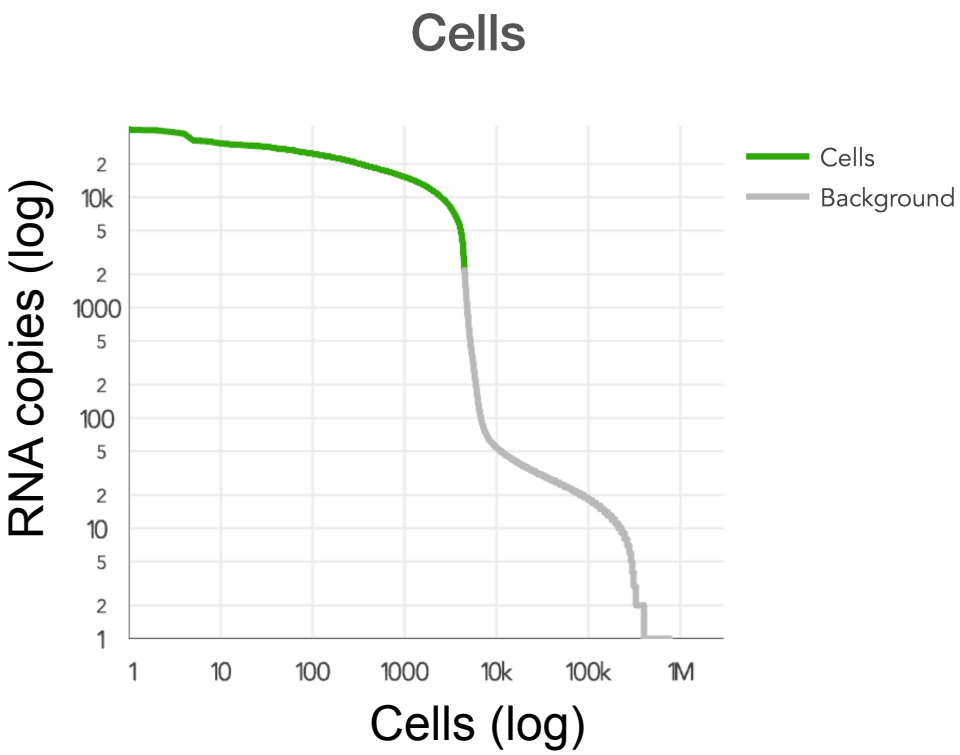


Basics Bioinformatics - single cell RNA-seq



Basics Bioinformatics - Cell Filtering

1. sum UMIs (copy of transcripts) per cell
2. consider cells with total UMI count > 99th of expected recovered cells



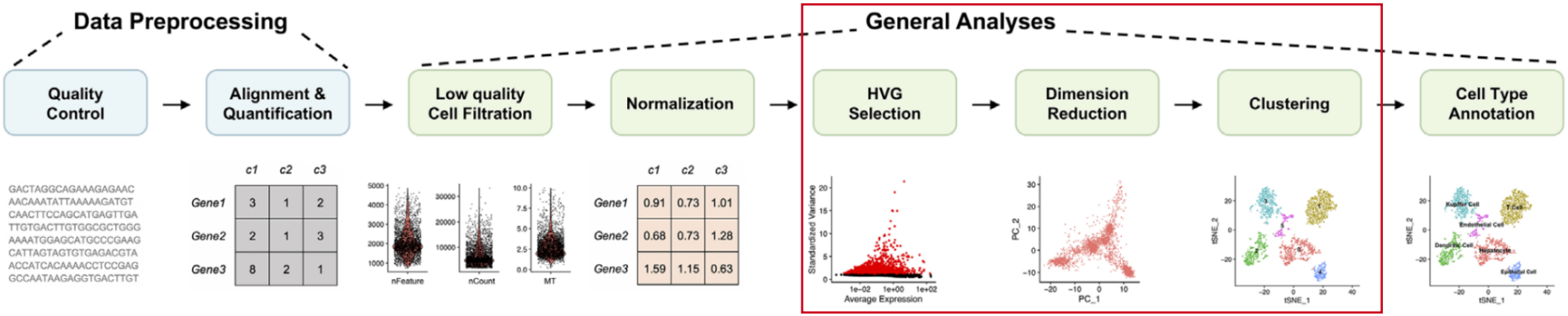
Estimated Number of Cells
4,495

Post-Normalization Mean Reads per Cell
89,289

Median Genes per Cell
2,504

cell ranger - 10x genomics

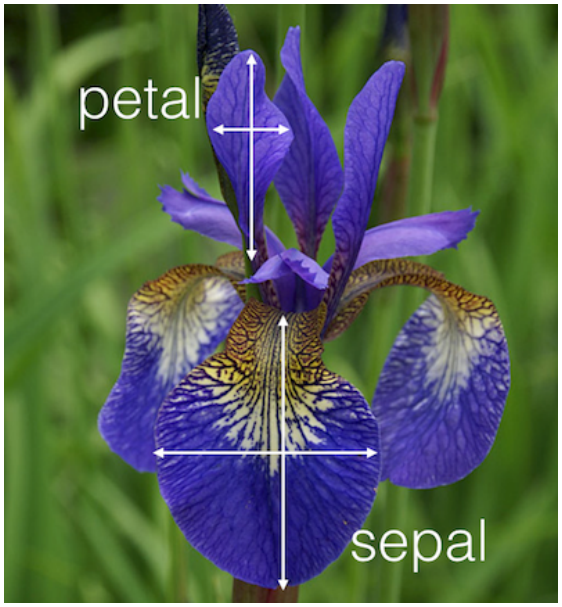
Basics Bioinformatics - single cell RNA-seq



Clustering & Dimension reduction

Clustering

- **Given a data description**
  - i.e. measurement of size of iris flowers
- **Find groups of similar observations**
  - i.e. iris flower sub-types

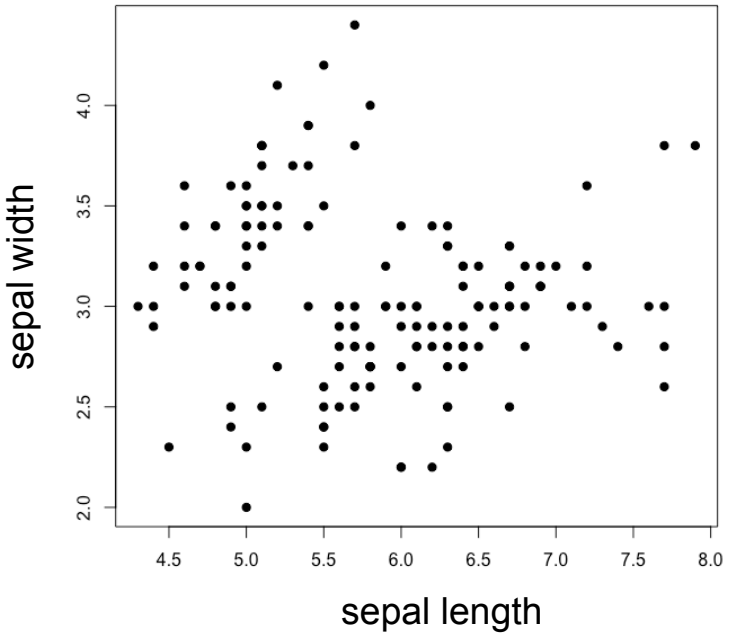


	Sepal Length	Sepal Width	Petal Length	Petal Width
Flower 1	5,1	3,5	1,4	0,2
Flower 2	4,9	3,0	1,4	0,2
Flower 3	4,7	3,2	1,3	0,2
Flower 4	4,6	3,1	1,5	0,2
…	…	…	…	…

Clustering

- **Given a data description**
  - i.e. measurement of size of iris flowers
- **Find groups of similar observations**
  - i.e. iris flower sub-types

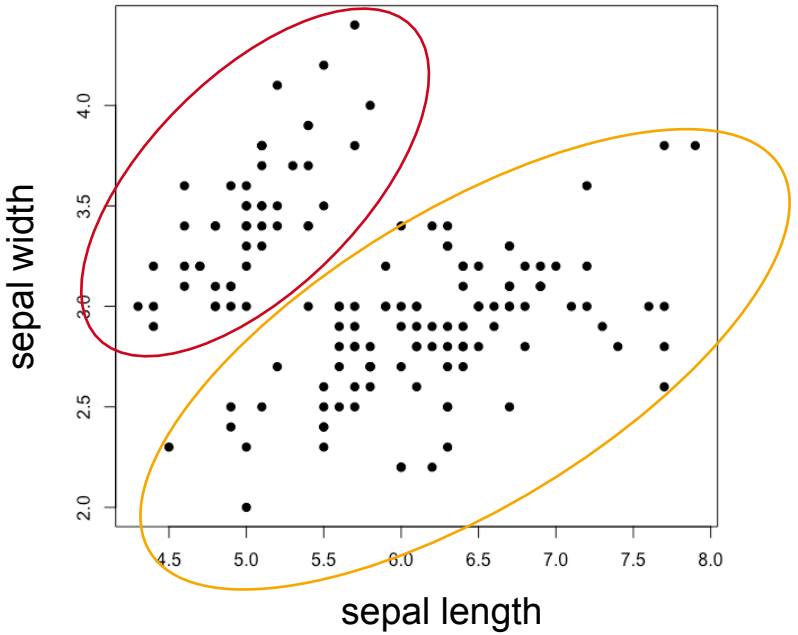
	Sepal Length	Sepal Width	Petal Length	Petal Width
Flower 1	5,1	3,5	1,4	0,2
Flower 2	4,9	3,0	1,4	0,2
Flower 3	4,7	3,2	1,3	0,2
Flower 4	4,6	3,1	1,5	0,2
…	…	…	…	…



Clustering

- **Given a data description**
  - i.e. measurement of size of iris flowers
- **Find groups of similar observations**
  - i.e. iris flower sub-types

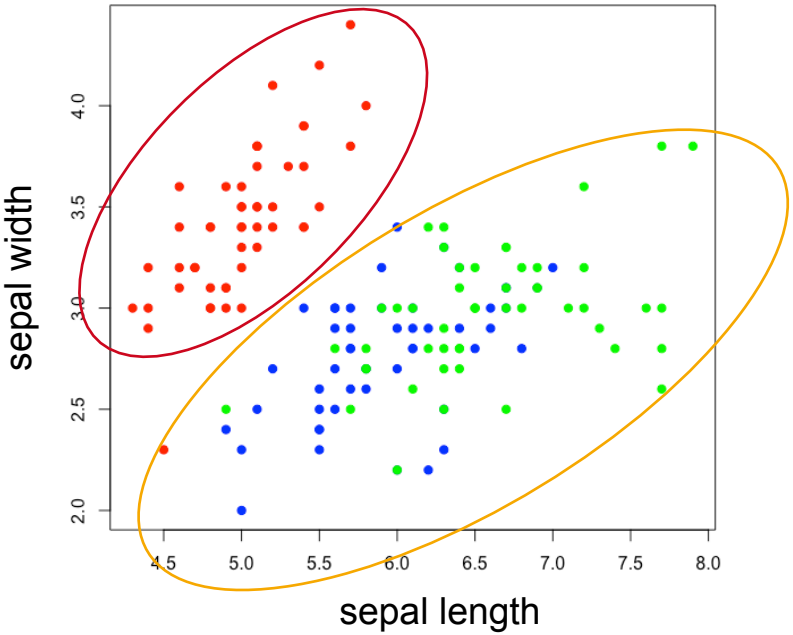
	Sepal Length	Sepal Width	Petal Length	Petal Width
Flower 1	5,1	3,5	1,4	0,2
Flower 2	4,9	3,0	1,4	0,2
Flower 3	4,7	3,2	1,3	0,2
Flower 4	4,6	3,1	1,5	0,2
…	…	…	…	…



Clustering

- **Given a data description**
  - i.e. measurement of size of iris flowers
- **Find groups of similar observations**
  - i.e. iris flower sub-types

Iris Setosa



Iris Virginia

Iris Versicolor

Clustering Formalism

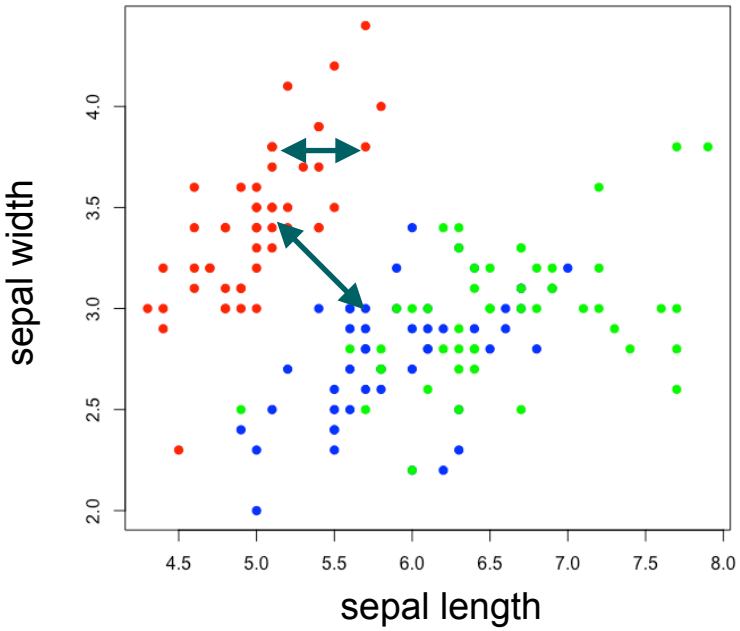
- **For a given data:**
  - Matrix X with N observations and L dimensions
    where x_i is a vector representing observation i

x_{11}	x_{12}	…	x_{1L}
x_{21}	x_{22}	…	x_{2L}
x_{31}	x_{32}	…	x_{3L}
…	…	…	…
x_{N1}	x_{N2}	…	x_{NL}

- **find groups of similar observations**
  - *vector* $Y = (y_1, \ldots, y_N)$
    where $y_i \in \{1, \ldots, K\}$ indicates the cluster of observation i

Distance

- **A important concept in clustering is a distance (similarity) between a pair of objects x_i and x_j**
  - Observations of a same group should be close in space

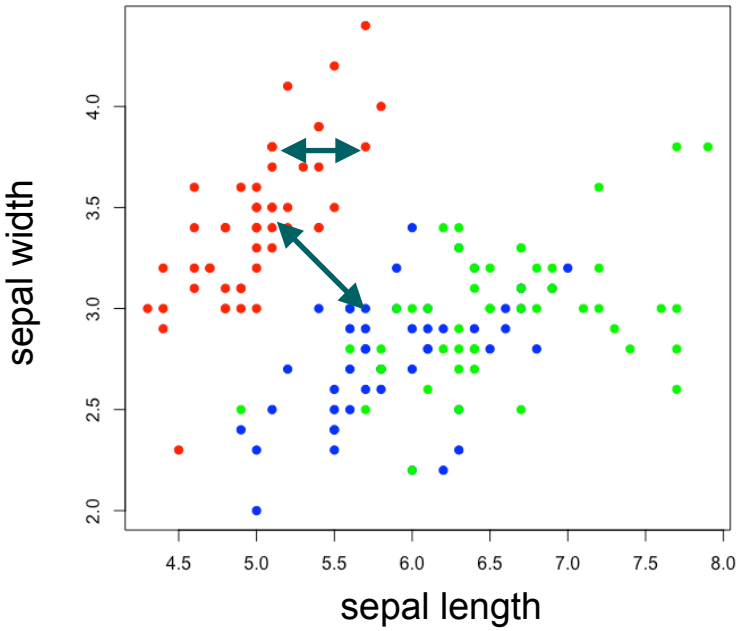


Euclidean distance
(sensitive to scale)

$$d(x_i, x_j) = \sqrt{\sum_{l=1}^{L} (x_{il} - x_{jl})^2}$$

Distance

- **A important concept in clustering is a distance (similarity) between a pair of objects x_i and x_j**
  - Observations of a same group should be close in space



Euclidean distance
(sensitive to scale)

$$d(x_i, x_j) = \sqrt{\sum_{l=1}^{L} (x_{il} - x_{jl})^2}$$

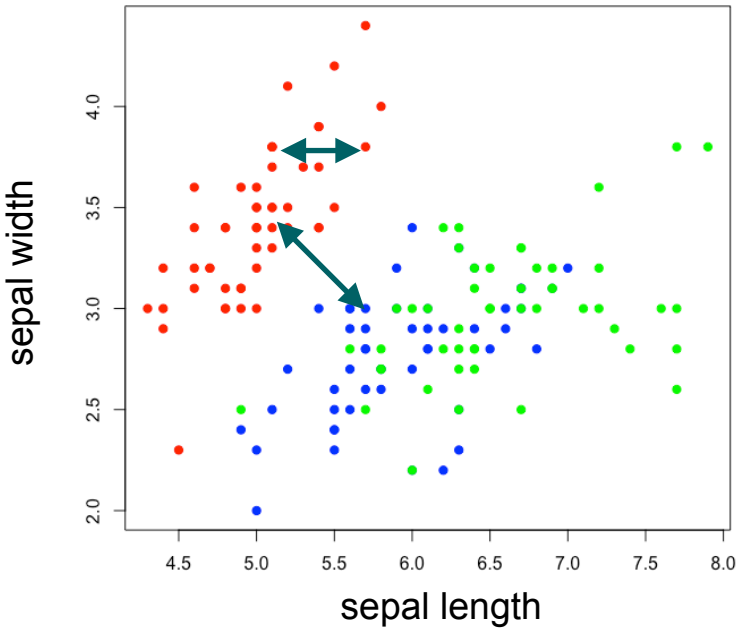
Pearson Correlation

(scale insensitive/ similarity)

$$d(x_i, x_j) = \frac{\sum_{l=1}^{L} (x_{il} - \bar{x}_i)(x_{jl} - \bar{x}_j)}{\sigma_i^2 \sigma_j^2}$$

Distance

- **A important concept in clustering is a distance (similarity) between a pair of objects x_i and x_j**
  - Observations of a same group should be close in space



Euclidean distance
(sensitive to scale)

$$d(x_i, x_j) = \sqrt{\sum_{l=1}^{L} (x_{il} - x_{jl})^2}$$

Pearson Correlation

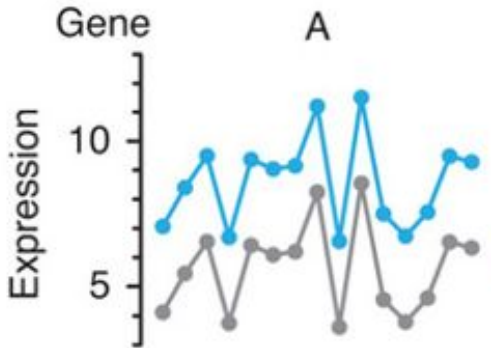
(scale insensitive/ similarity)

$$d(x_i, x_j) = \frac{\sum_{l=1}^{L} (x_{il} - \bar{x}_i)(x_{jl} - \bar{x}_j)}{\sigma_i^2 \sigma_j^2}$$

Distance and Scale

- **In some problems scale can be important!**
  - Similarly in changes are more important / not absolute values.

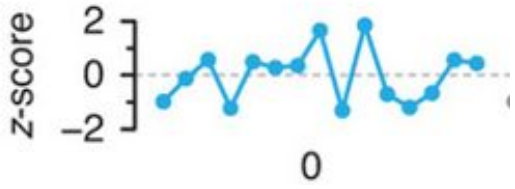
unscaled data



Euclidean - not similar

Correlation - similar

z-score normalised data



$$z = \frac{x_{ij} - \mu_i}{\sigma_i}$$

Euclidean - similar

Correlation - similar

Clustering Methods

- **Hierarchical methods**
  - Mostly bottom up
  - based on distance / simple to interpret
- **Partitional methods (k-means or mixture models)**
  - Mostly top down
  - Use models of groups, centroids
- **Graph based methods**
  - Use graph formalisms to represent data:
    - nodes are objects
    - edges weights represent similarities
    - find well connected graphs

K-means

Iterative algorithm using **centroids** as cluster representations

Requires specification of number of clusters (***K***)

Algorithm:

Start cluster (Y) randomly

Repeat for a number of iterations

- estimate centroid (m_k) for each cluster

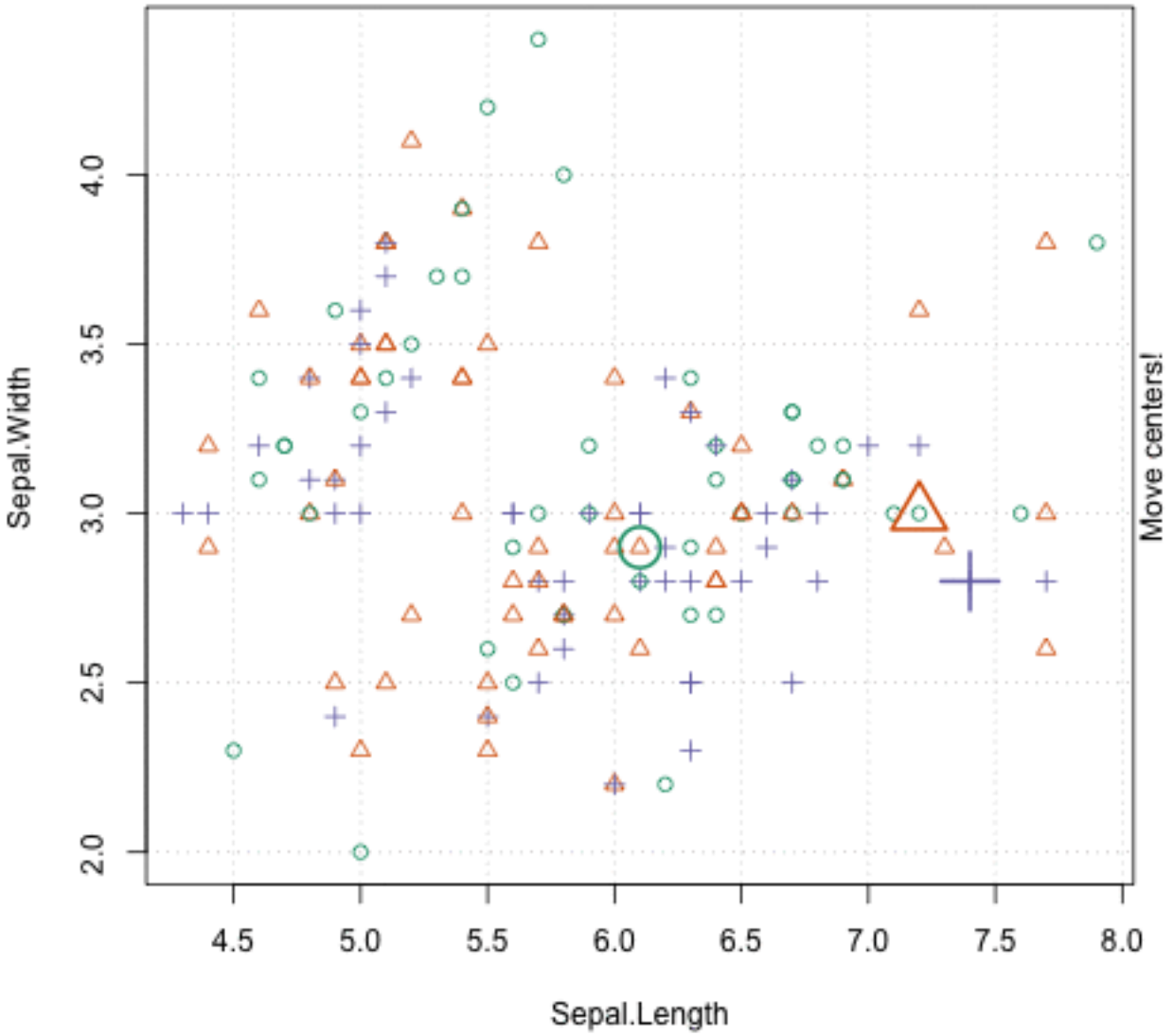
$$m_k = \frac{\sum_{i=1}^{N} 1(y_i = k) x_i}{\sum_{i=1}^{N} 1(y_i = k)}$$

- Assign objects to closest centroid:

$$y_i = \text{argmin}_k \, d(x_i, m_k)$$

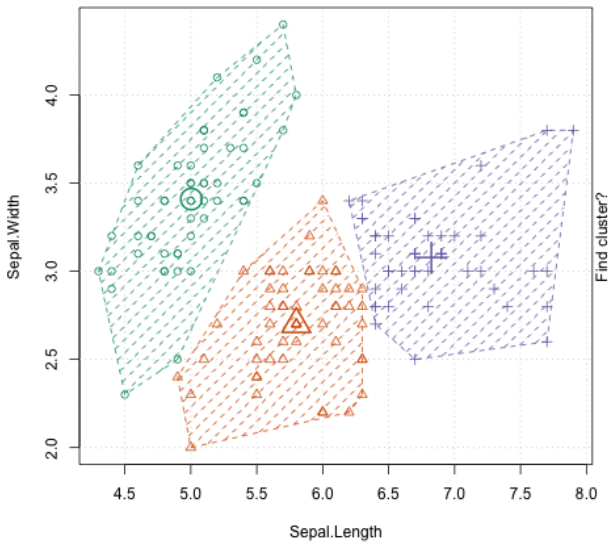
* convergence is only guaranteed for Euclidean distance

K-means on Iris

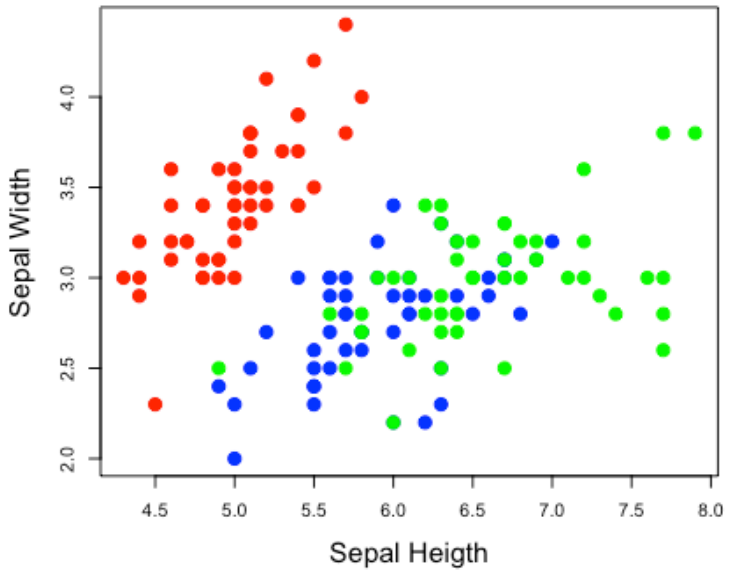


K-means on Iris

K-means solutions

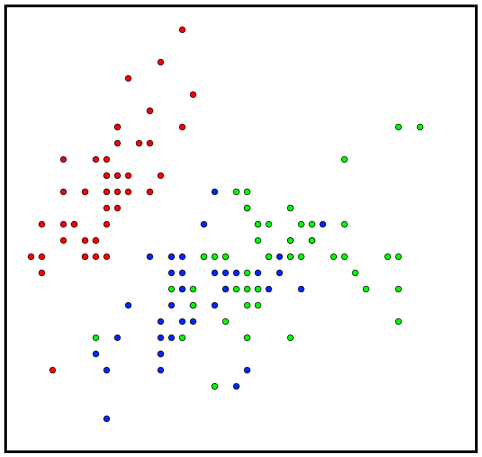


True labels



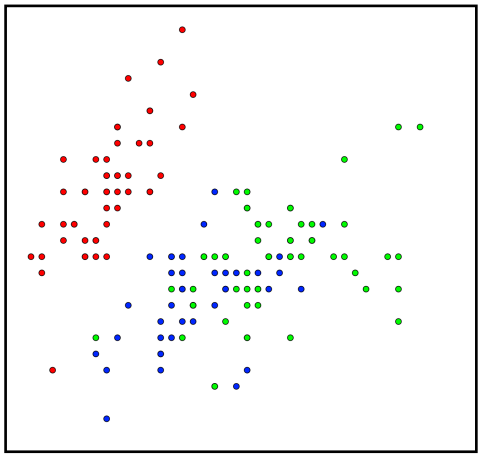
- K-means tends to find spherical clusters
- Sensitive to initialisation

Graph based clustering

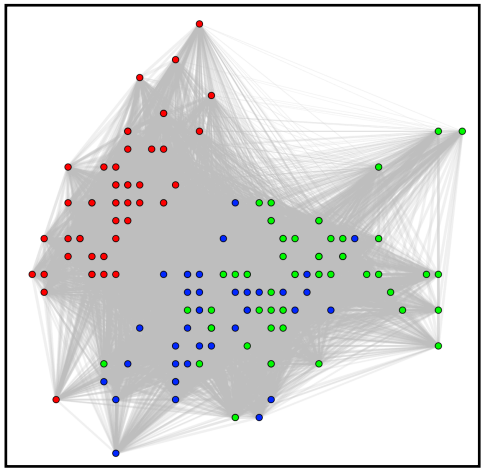


- data points are nodes

Graph based clustering

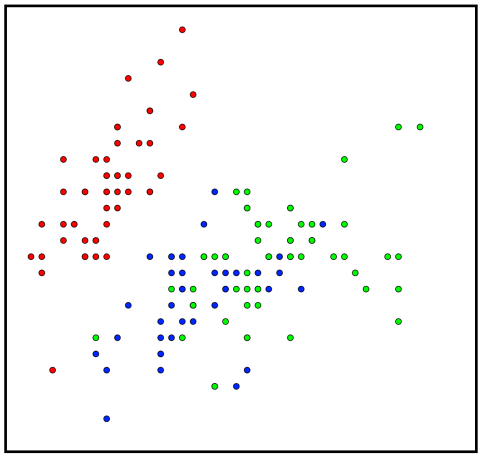


- data points are nodes

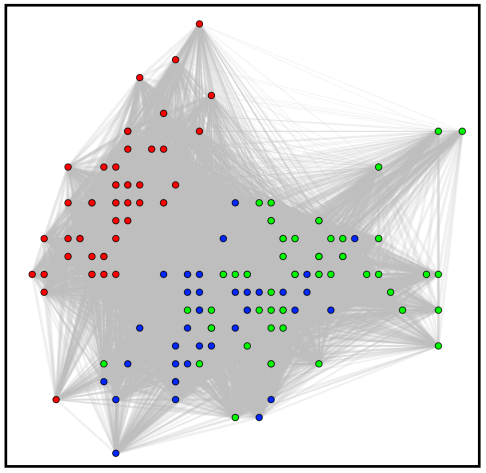


- edges represent similarities

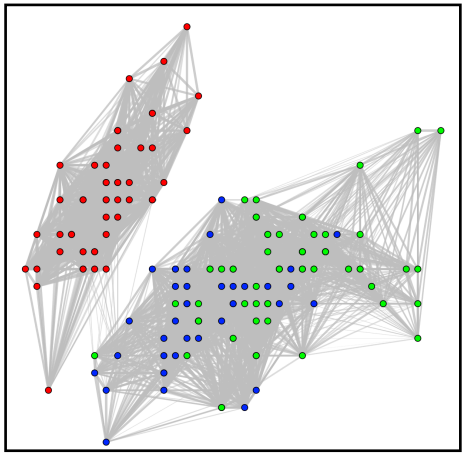
Graph based clustering



- data points are nodes

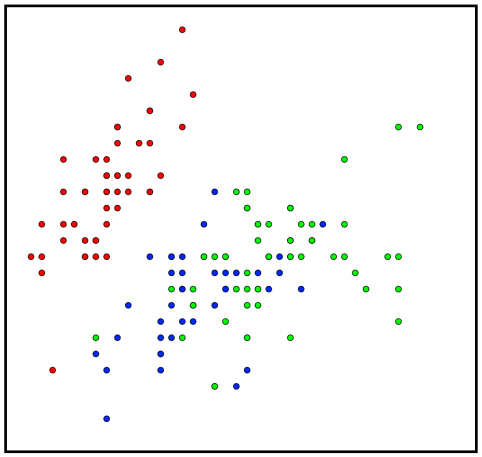


- edges represent similarities

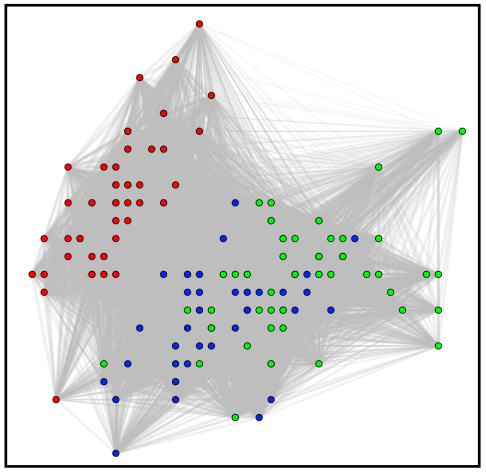


- k-nearest neighbours (KNN) -> sparse graphs

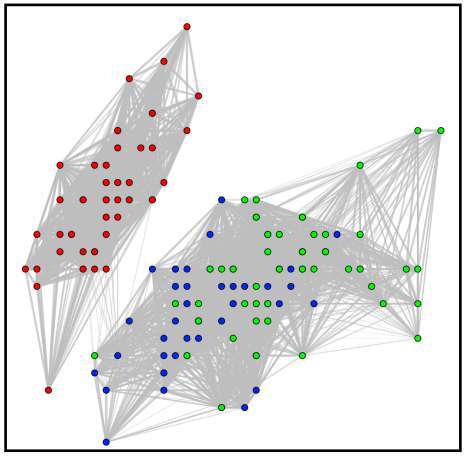
Graph based clustering



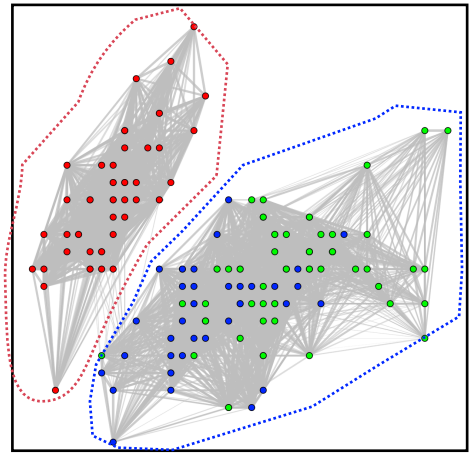
- data points are nodes



- edges represent similarities

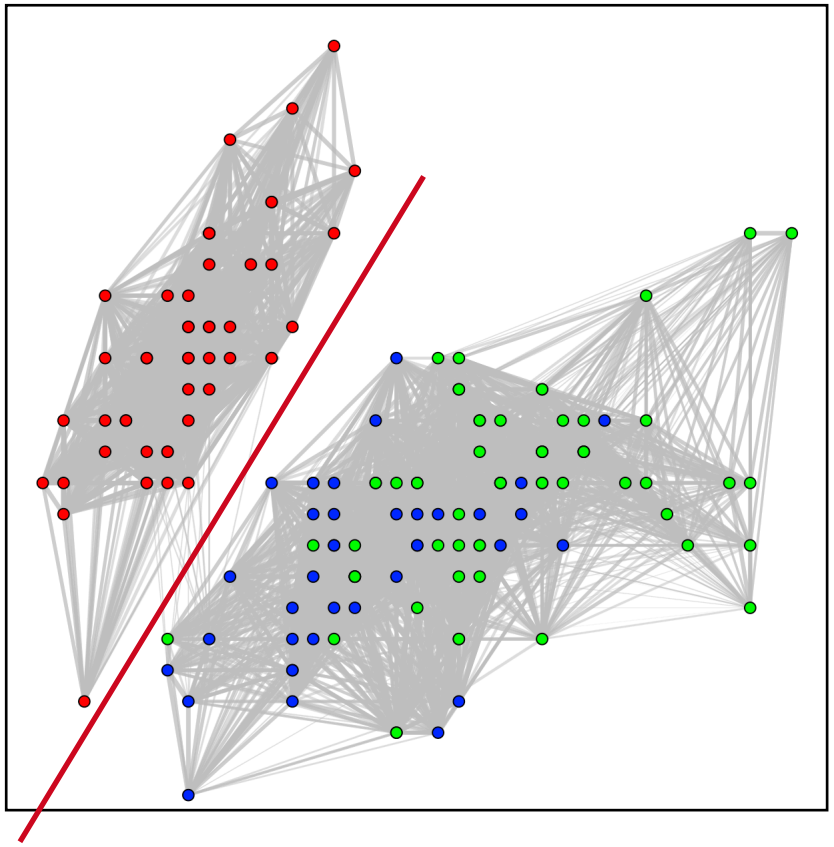


- k-nearest neighbours (KNN) -> sparse graphs



- find well connected sub-graphs

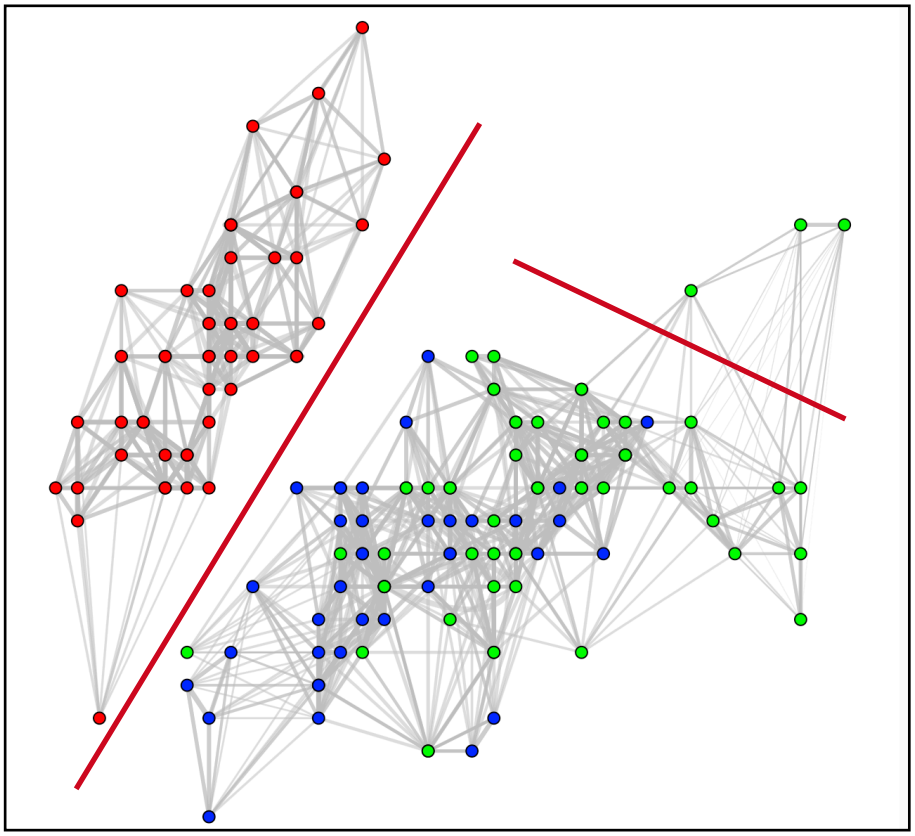
Graph cut



KNN = 50

- Cluster by finding cuts in the graph
- Cut cost **C(*A*,*B*)** = sum of edge weights in cut

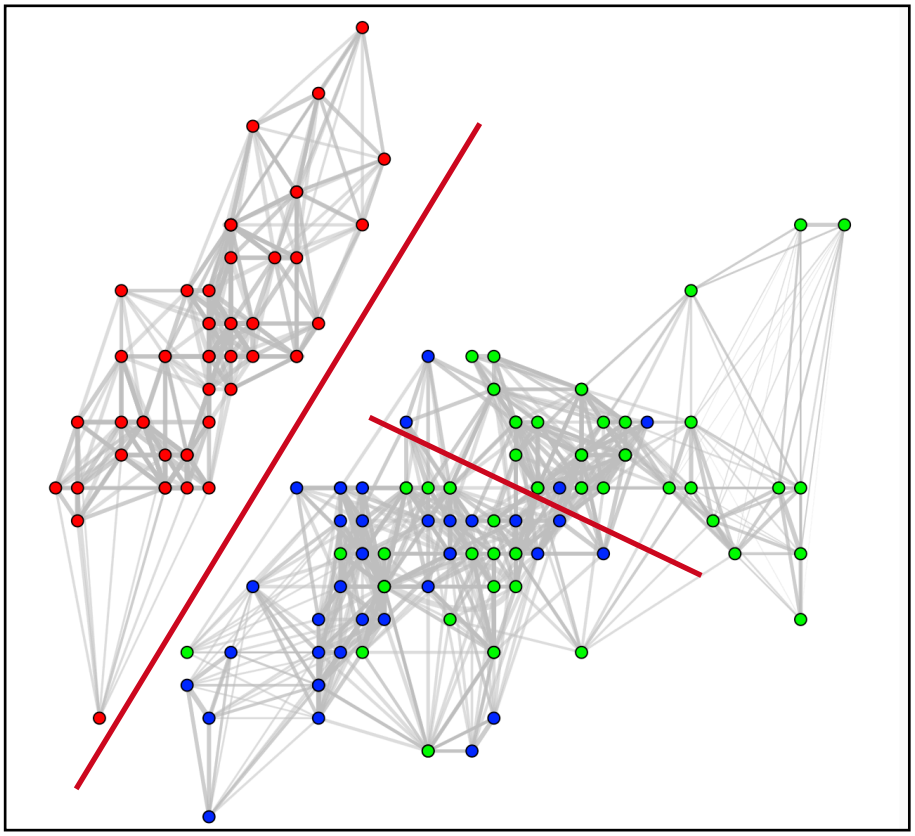
Graph cut



KNN = 10

- Cluster by finding cuts in the graph
- Cut cost **C(*A*,*B*)** = sum of edge weights in cut
  - smallest cuts might not be the best

Normalized graph cut



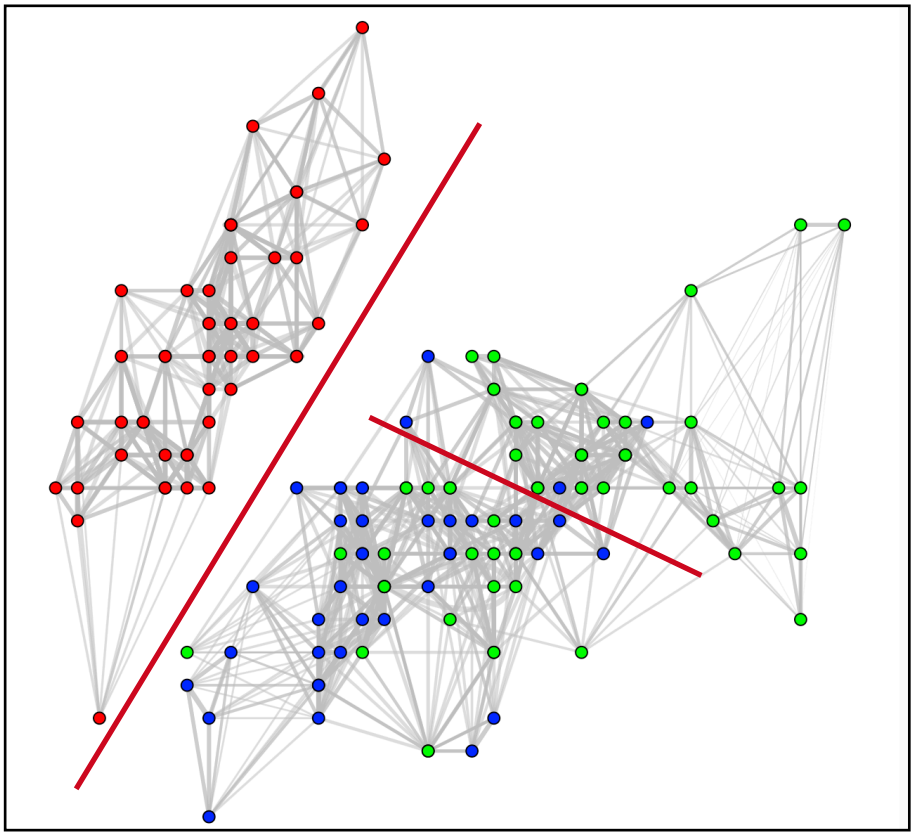
KNN = 10

- Normalized graph cut avoids small graphs

$$normCUT(A,B) = \frac{CUT(A,B)}{VOL(A)} + \frac{CUT(A,B)}{VOL(B)}$$

where $VOL(A)$ is the weight sums of cluster A.

Spectral Clustering



KNN = 10

- Let *A* be an adjacent matrix of the graph:
  - $a_{ij}=1$ if nodes *i* and *j* are connected
- A laplacian matrix is defined as:

  $L = D - A$

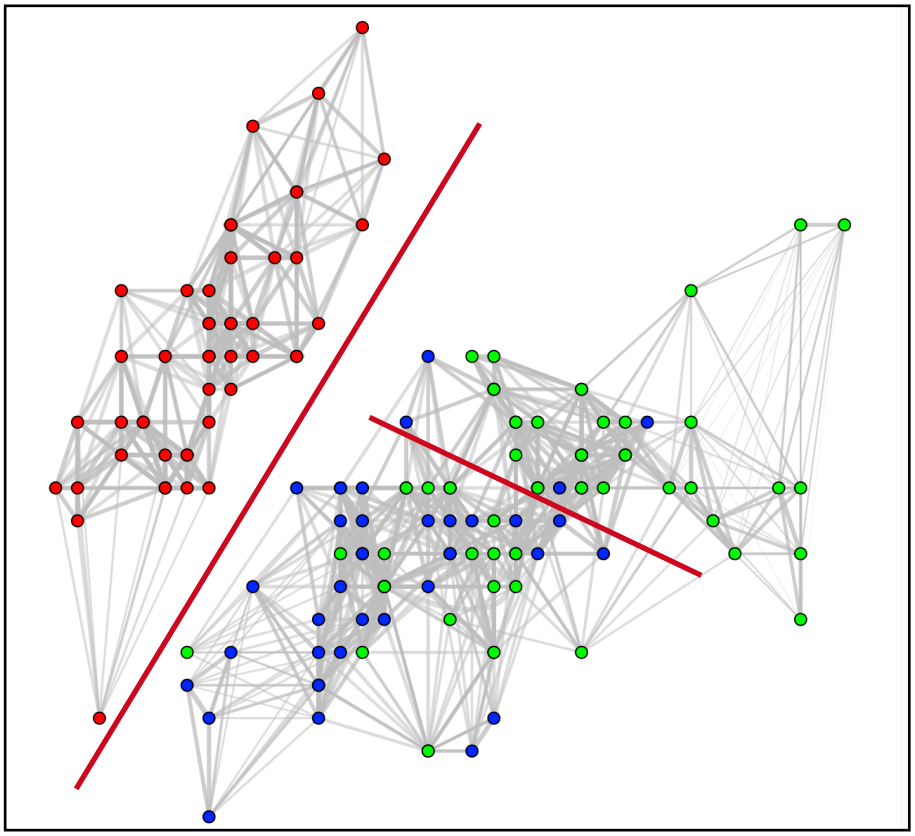
  - where *D* is a diagonal matrix with the number of neighbours of a node
- If we perform a spectral analysis of *L**

  $$L\lambda = u\lambda$$

  - eigenvectors (λ) provides CUTs in the graph
  - eigenvalues (u) provides the cost of the CUT.
  - Perform *k*-means on lowest K eigenvalues

* see for more details: http://www.tml.cs.uni-tuebingen.de/team/luxburg/publications/Luxburg07_tutorial.pdf

Spectral Clustering



KNN = 10

- Let *A* be an adjacent matrix of the graph:
  - $a_{ij}=1$ if nodes *i* and *j* are connected
- A laplacian matrix is defined as:

  $L = D - A$

  - where *D* is a diagonal matrix with the number of neighbours of a node
- If we perform a spectral analysis of *L**

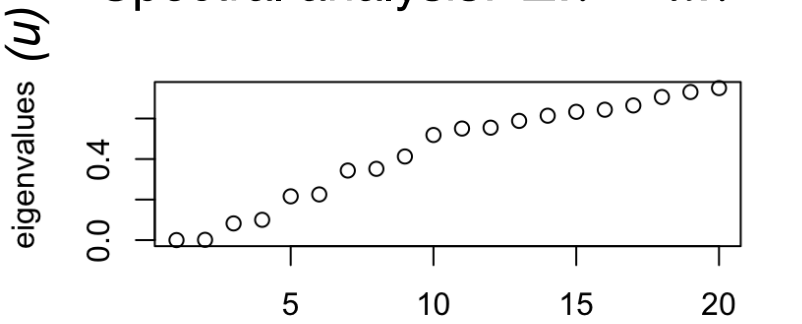
  $$L\lambda = u\lambda$$

  - eigenvectors (λ) provides CUTs in the graph
  - eigenvalues (u) provides the cost of the CUT.
  - Perform *k*-means on lowest K eigenvalues

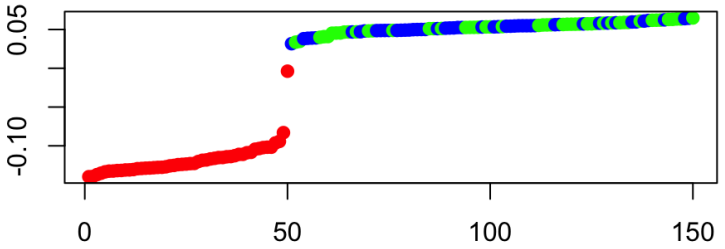
* see for more details: http://www.tml.cs.uni-tuebingen.de/team/luxburg/publications/Luxburg07_tutorial.pdf

Graph cut and spectral analysis of laplacian matrices

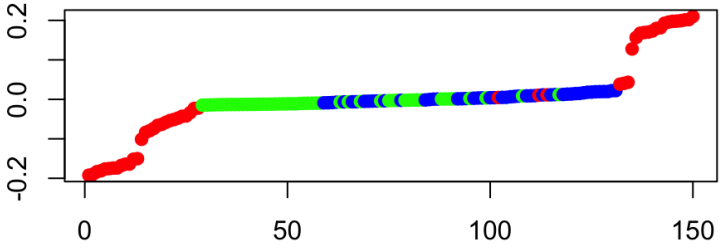
Spectral analysis: $L\lambda = u\lambda$

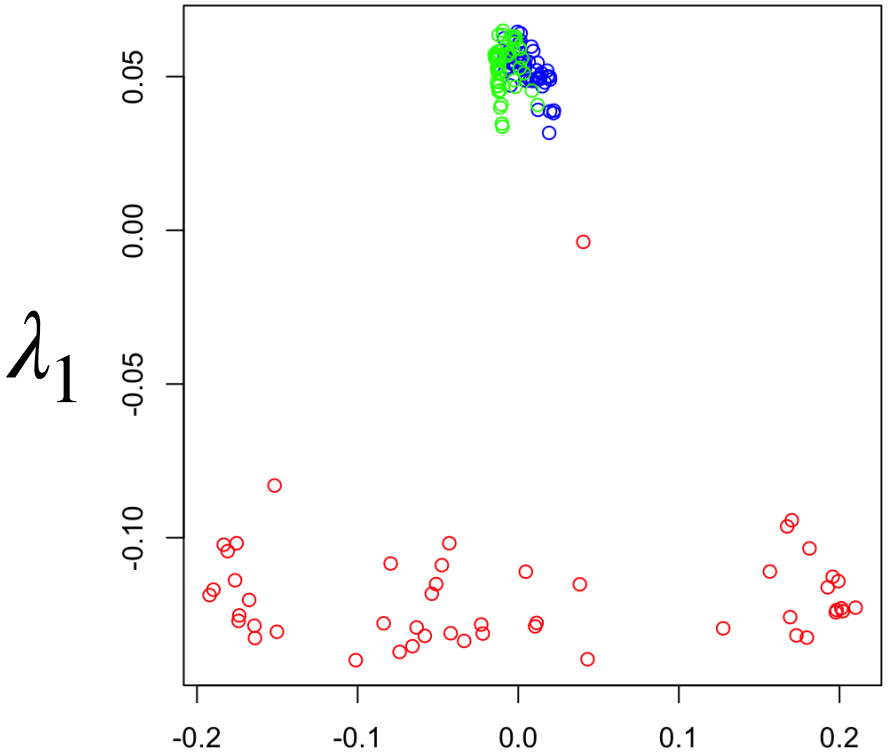


λ_1



λ_2





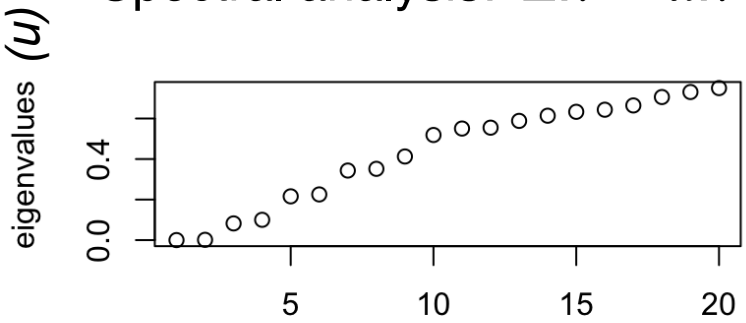
λ_1

λ_2

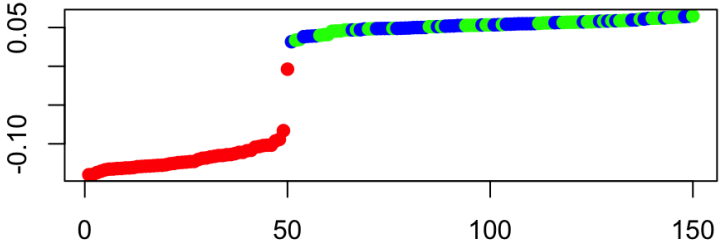
http://www.tml.cs.uni-tuebingen.de/team/luxburg/publications/Luxburg07_tutorial.pdf

Graph cut and spectral analysis of laplacian matrices

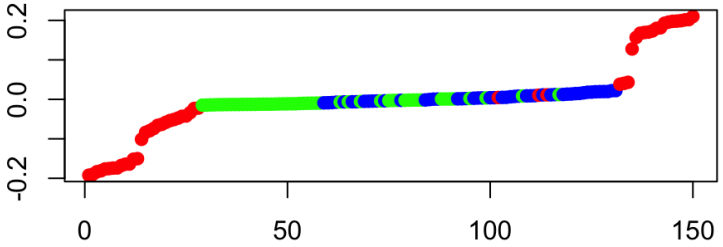
Spectral analysis: $L\lambda = u\lambda$



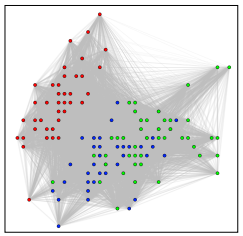
λ_1

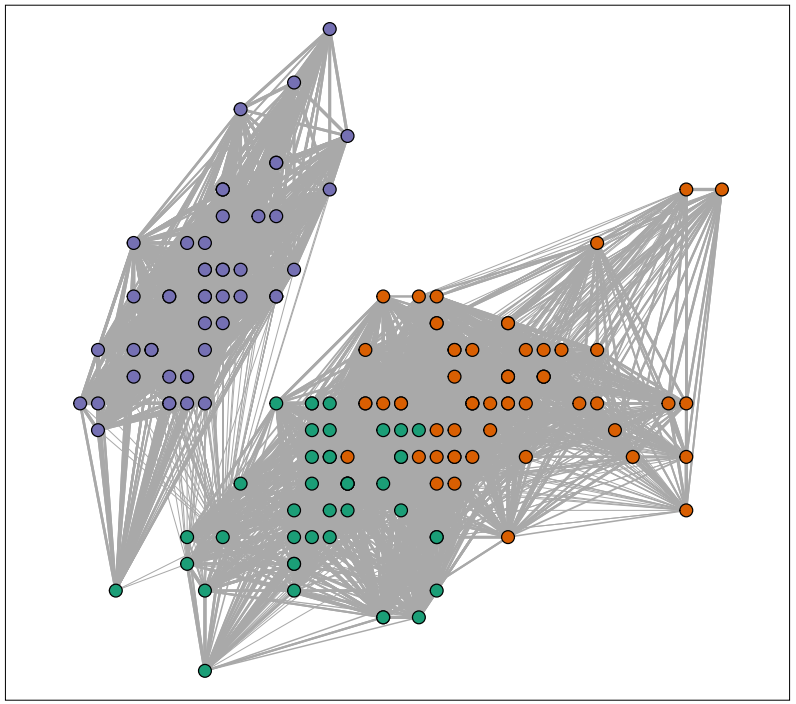


λ_2



original labels

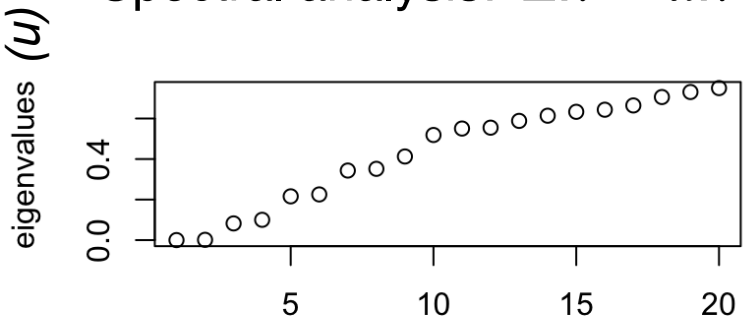




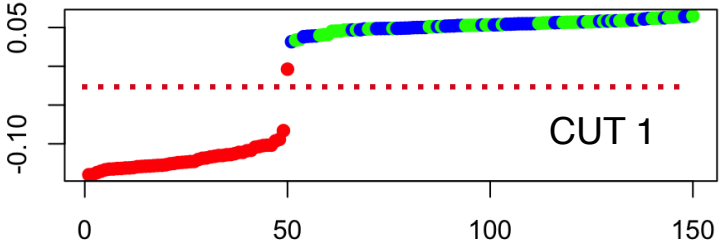
http://www.tml.cs.uni-tuebingen.de/team/luxburg/publications/Luxburg07_tutorial.pdf

Graph cut and spectral analysis of laplacian matrices

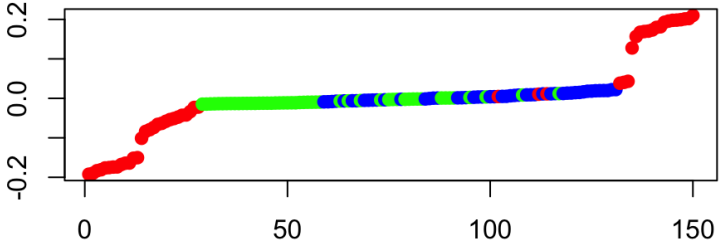
Spectral analysis: $L\lambda = u\lambda$



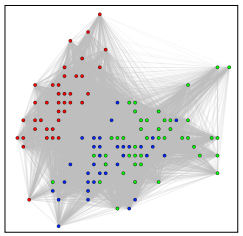
λ_1

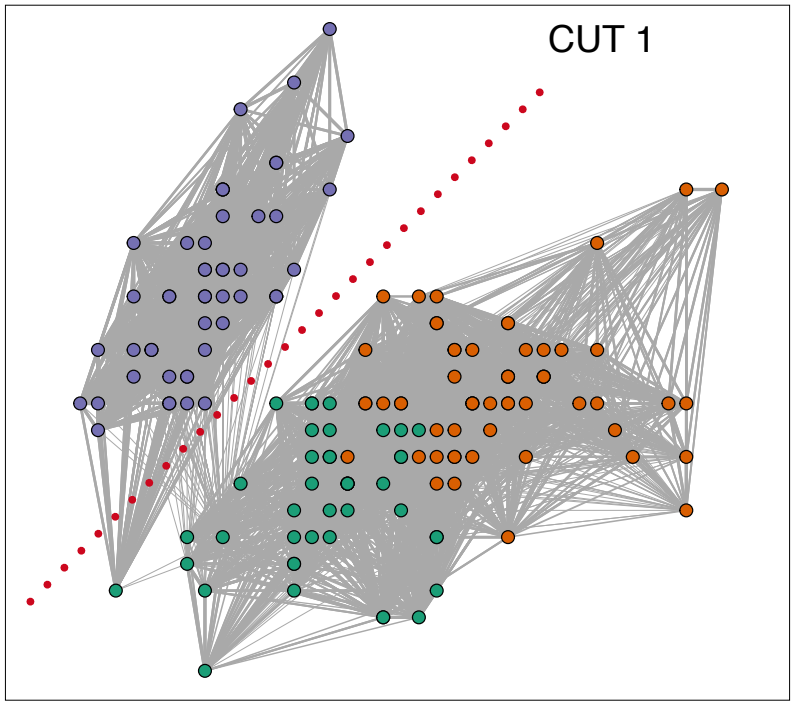


λ_2



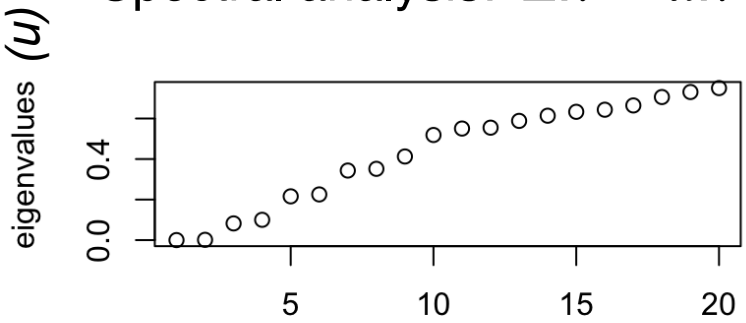
original labels



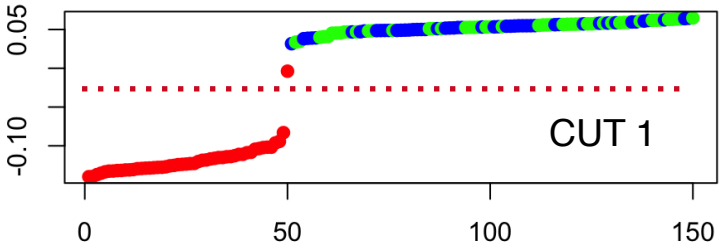


Graph cut and spectral analysis of laplacian matrices

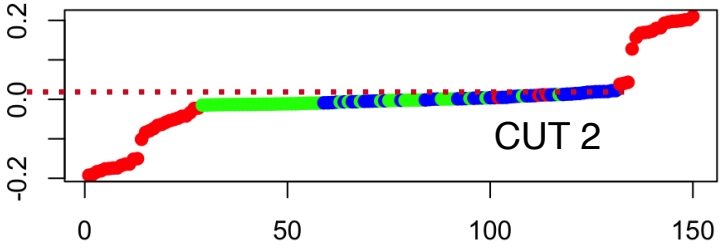
Spectral analysis: $L\lambda = u\lambda$



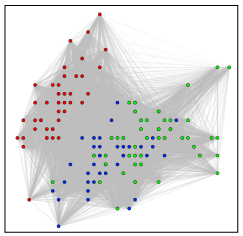
λ_1

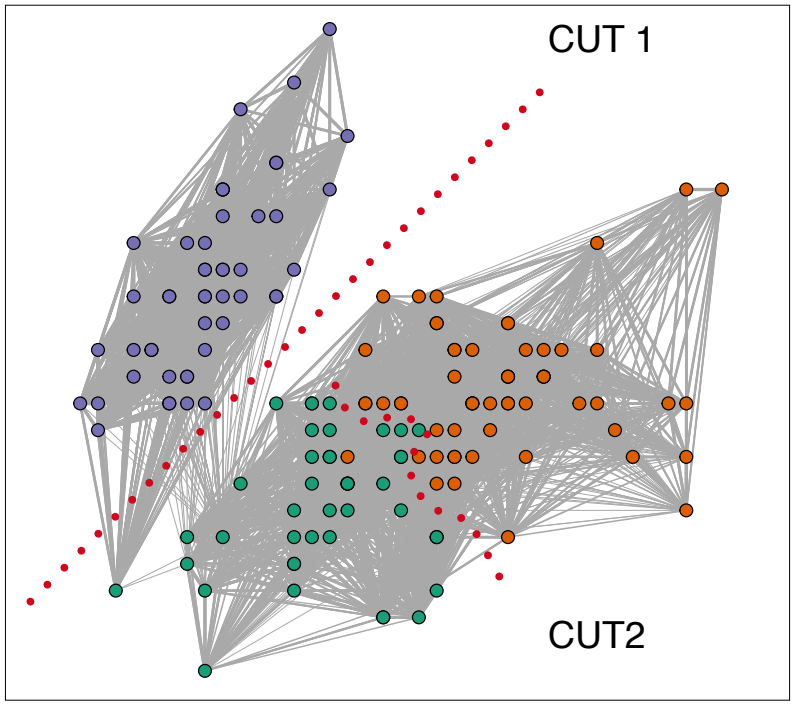


λ_2



original labels

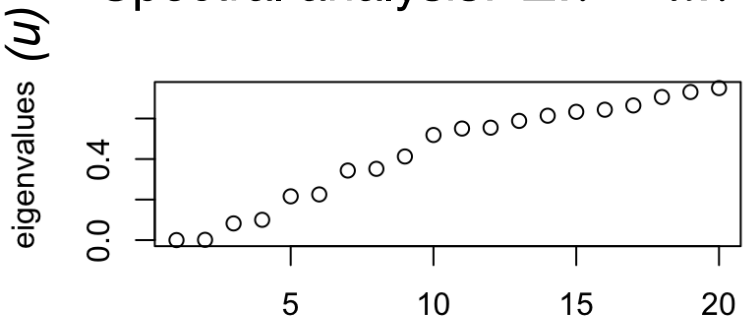




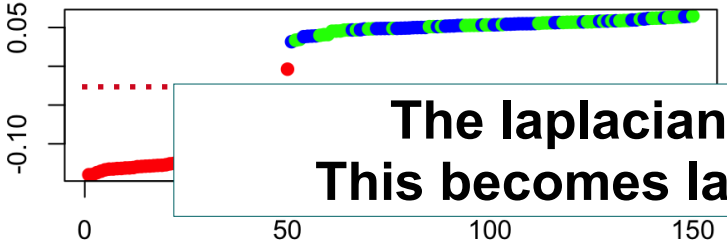
http://www.tml.cs.uni-tuebingen.de/team/luxburg/publications/Luxburg07_tutorial.pdf

Graph cut and spectral analysis of laplacian matrices

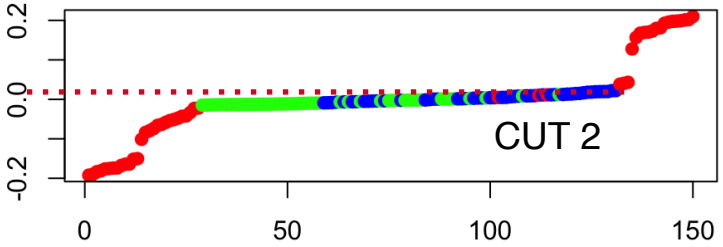
Spectral analysis: $L\lambda = u\lambda$



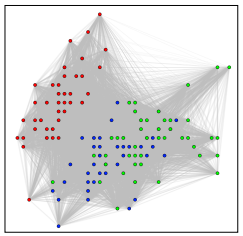
λ_1



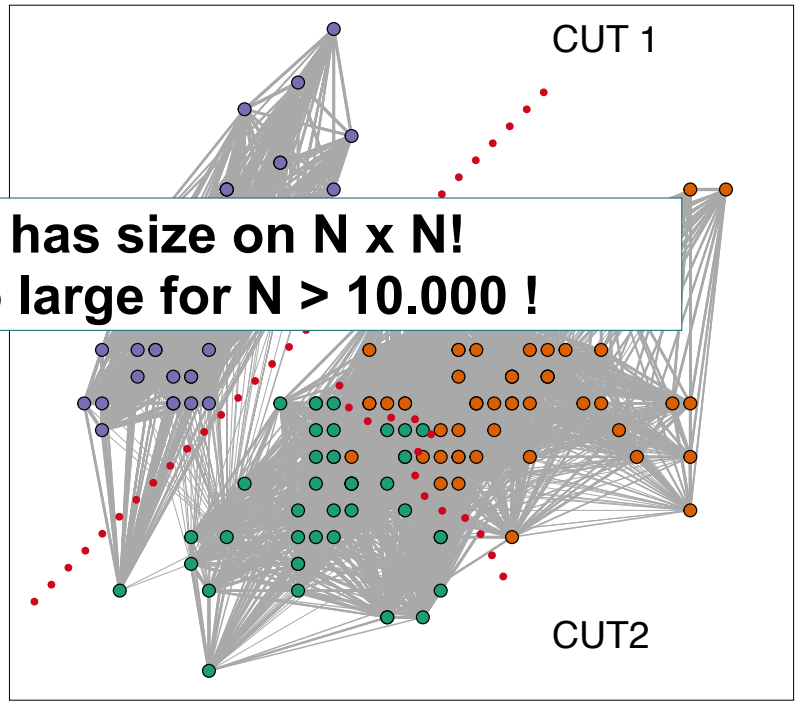
λ_2



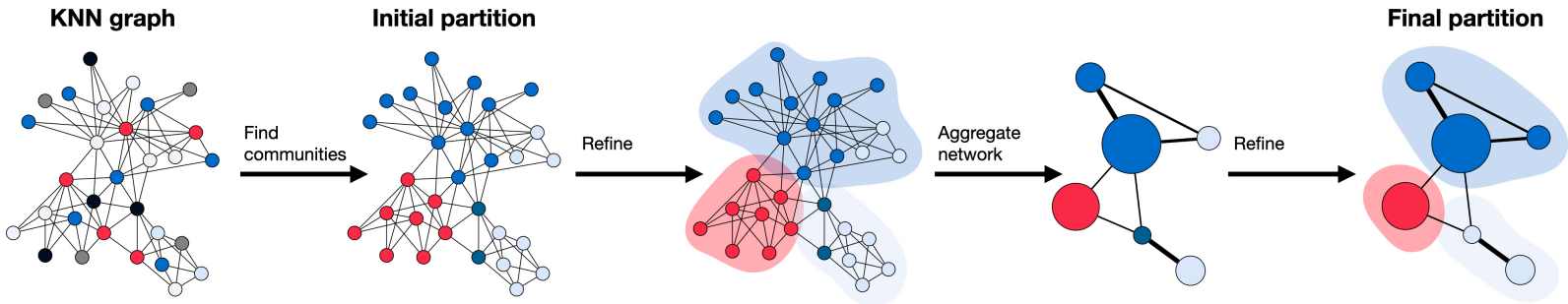
original labels



The laplacian matrix has size on N x N!
This becomes large too large for N > 10.000 !



Single cell Clustering / Louvain & Leiden algorithm



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

Optimize cluster modularity

$$\mathcal{H} = \sum_c [e_c - \gamma(\binom{n_c}{2})],$$

where n_c is the size of cluster and

e_c is the number of expected edges

A) Start with a random partition

B) Cluster objects improving *H*

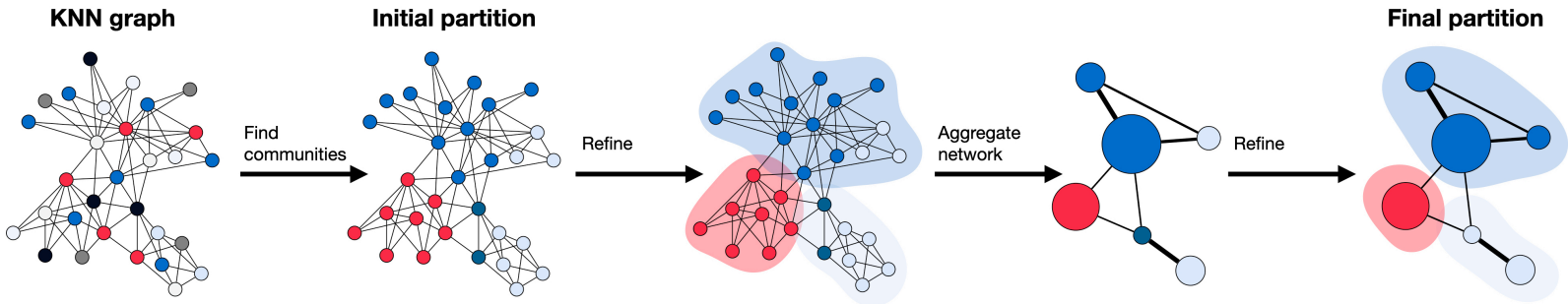
C) Create a meta-graph level:

- one meta-node for each cluster

D) Move objects improving *H*

Van Traag, Scientific Reports, 2019.
Blondel, Journal of Statistical Mechanics, 2008

Single cell Clustering / Louvain & Leiden algorithm



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

Optimize

Meta-nodes and sparse graphs (knn) allows Leiden/Louvain to cope with millions of objects !

$$\mathcal{H} = \sum_c [e_c - \gamma \binom{n_c}{2}],$$

where n_c is the size of cluster and

e_c is the number of expected edges

A) Start with a random partition

B) Cluster objects improving *H*

C) Create a meta-graph level:

- one meta-node for each cluster

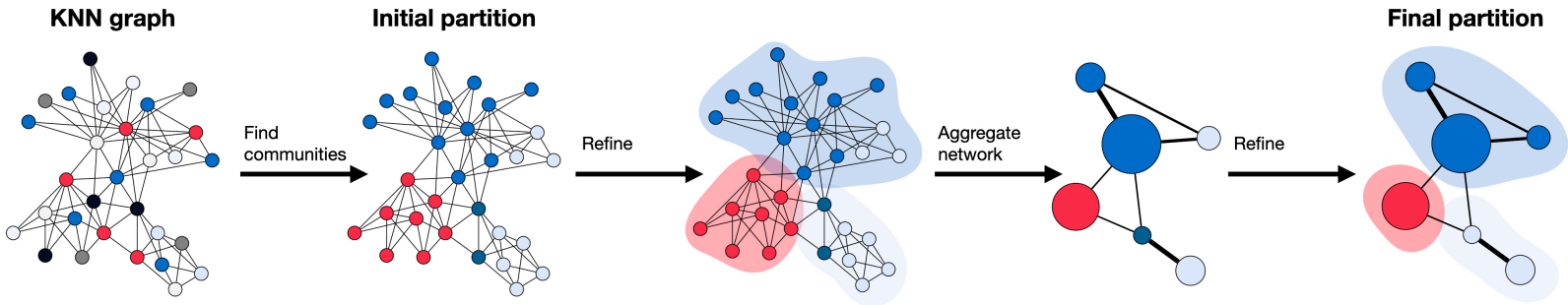
D) Move objects improving *H*

Van Traag, Scientific Reports, 2019.
Blondel, Journal of Statistical Mechanics, 2008

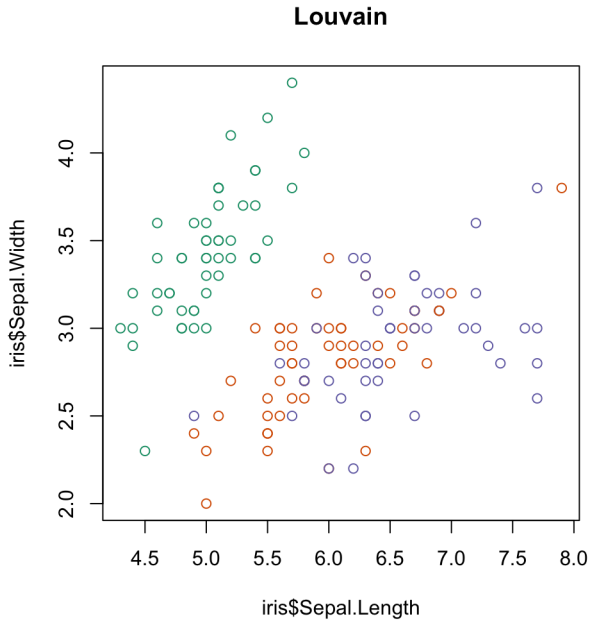
Institute for Computational Genomics

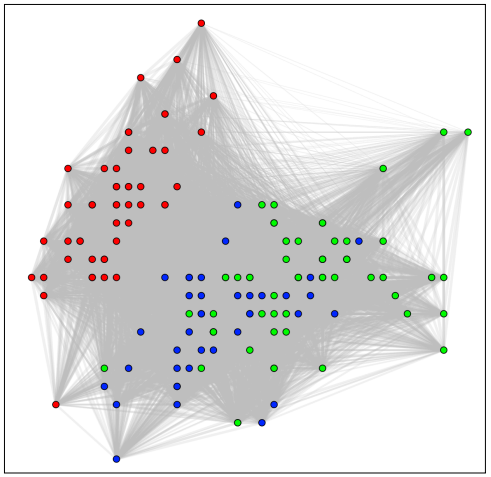
RWTH AACHEN UNIVERSITY

Single cell Clustering / Louvain & Leiden algorithm



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





Van Traag, Scientific Reports, 2019.
Blondel, Journal of Statistical Mechanics, 2008

Resume / Clustering Methods

- K-means, hierarchical clustering, spectral clustering

  - standard algorithms with standard performance on simple clustering problems

- Clustering of single cell algorithms
  - Leiden and louvain clustering
  - Robust and scale well to large data sets on sparse graphs (knn)
- Further issues:
  - Data dimensionality:
    - distances do not work well on high dimension
    - visualisation is easier in low level space
  - Validation:
    - How many clusters is present in the data?
    - Which is the best method?

More details on clustering

- Hastie, Tibshirani and Friedman, The Elements of Statistical Learning, Chapter 14
- Video lecture: https://www.youtube.com/watch?v=Qa6k7RIwltg

Resume / Single cell clustering

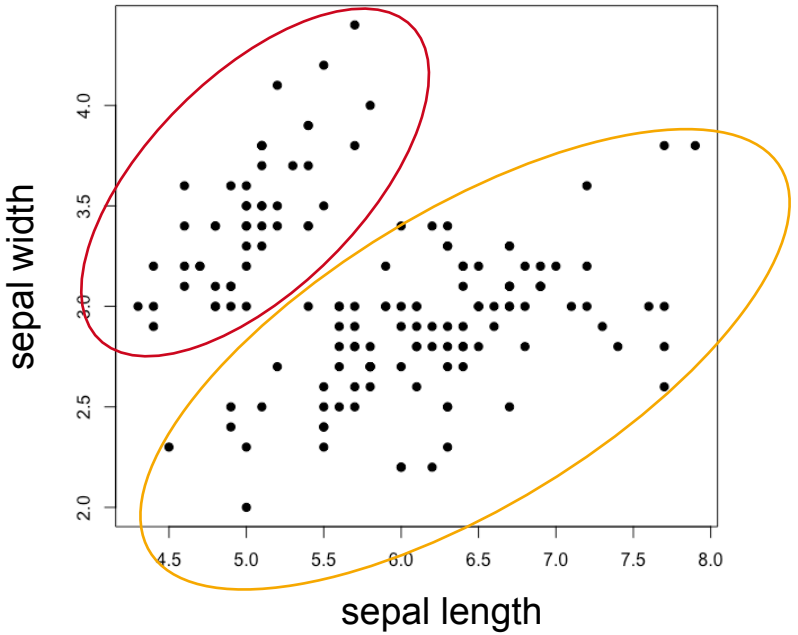
- Finding groups of single cells require complex pipeline:
  - Cell filtering
  - Normalisation
  - Artefact removal
  - **Dimension reduction**
  - **Integration**
  - **Clustering**
  - **Cell annotation / visualisation**

Clustering & Dimension reduction

Clustering

- **Given a data description**
  - i.e. measurement of size of iris flowers
- **Find groups of similar observations**
  - i.e. iris flower sub-types

	Sepal Length	Sepal Width	Petal Length	Petal Width
Flower 1	5,1	3,5	1,4	0,2
Flower 2	4,9	3,0	1,4	0,2
Flower 3	4,7	3,2	1,3	0,2
Flower 4	4,6	3,1	1,5	0,2
…	…	…	…	…

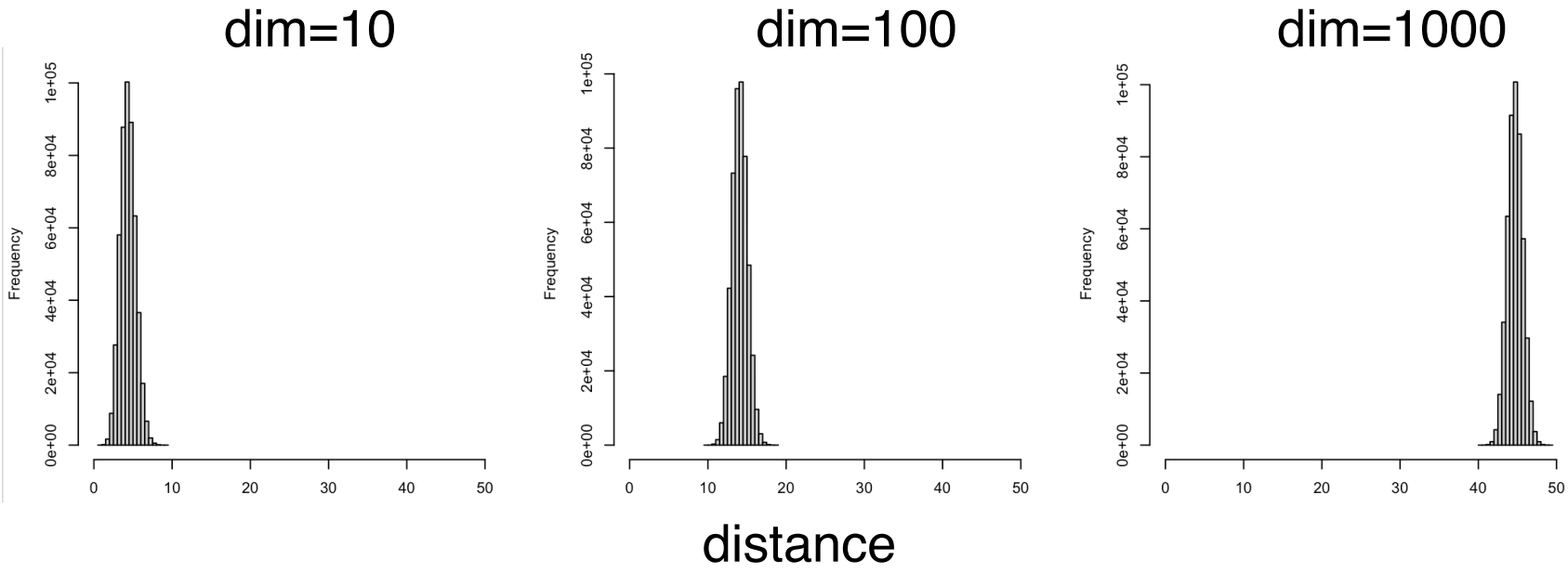


Dimension Reduction

- Distances lose meaning at high dimensional space (curse of dimensionality)

$$\frac{D_{\max} - D_{\min}}{D_{\min}} \to 0.$$

- Example: distance between points sampled from a normal distribution



Dimension Reduction

- Distances lose meaning at high dimensional space (curse of dimensionality)

- Unspecific Filtering (without class labels):
  - Keep variables with highest variance (high variable genes)
    - ***Rationale**:* important features change values across groups

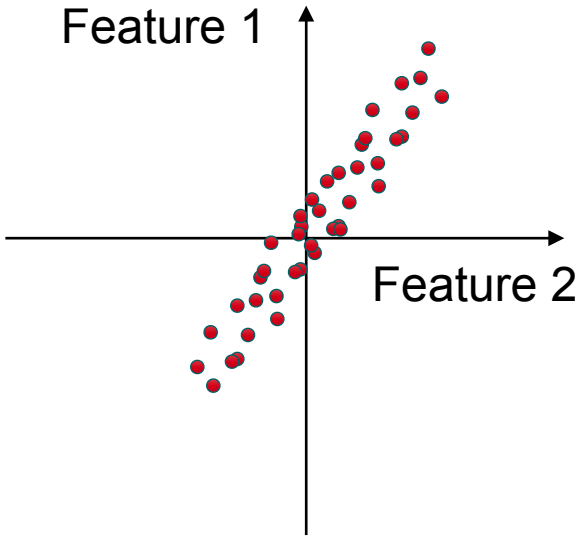
- Dimensionality Reduction by Transformation:
  - linear: principal component analysis (PCA)
  - Non-linear / manifold learning: t-SNE & UMAP (for visualisation)

Principal Component Analysis

- For a data *X*, find linear combination of features (*w)* capturing most of data variance

$$\mathbf{w}_{(1)} = \arg\max_{\|\mathbf{w}\|=1} \{\|\mathbf{X}\mathbf{w}\|^2\}$$

- Can be solved by linear algebra / eigenvector decomposition



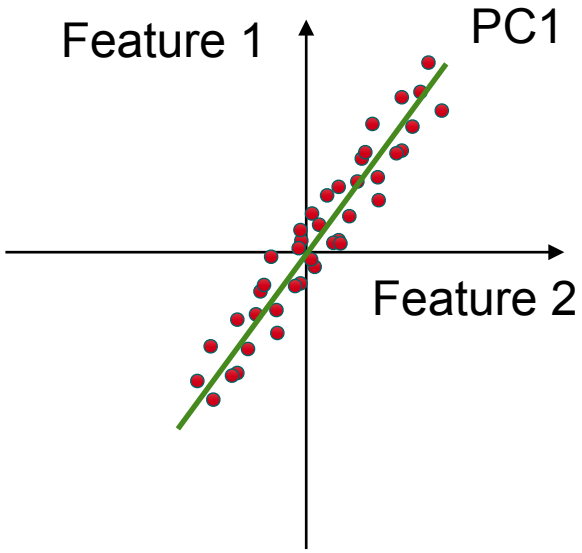
Recommended reading:
Ringner M., *Nature Biotechnology* 26, 303 - 304 (2008)

Principal Component Analysis

- For a data *X*, find linear combination of features (*w)* capturing most of data variance

$$\mathbf{w}_{(1)} = \arg\max_{\|\mathbf{w}\|=1} \{\|\mathbf{X}\mathbf{w}\|^2\}$$

- Can be solved by linear algebra / eigenvector decomposition



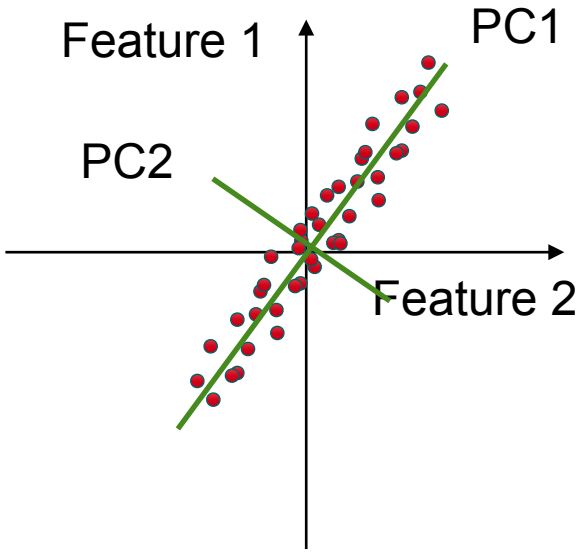
Recommended reading:
Ringner M., *Nature Biotechnology* 26, 303 - 304 (2008)

Principal Component Analysis

- For a data *X*, find linear combination of features (*w)* capturing most of data variance

$$\mathbf{w}_{(1)} = \arg\max_{\|\mathbf{w}\|=1} \{\|\mathbf{X}\mathbf{w}\|^2\}$$

- Can be solved by linear algebra / eigenvector decomposition



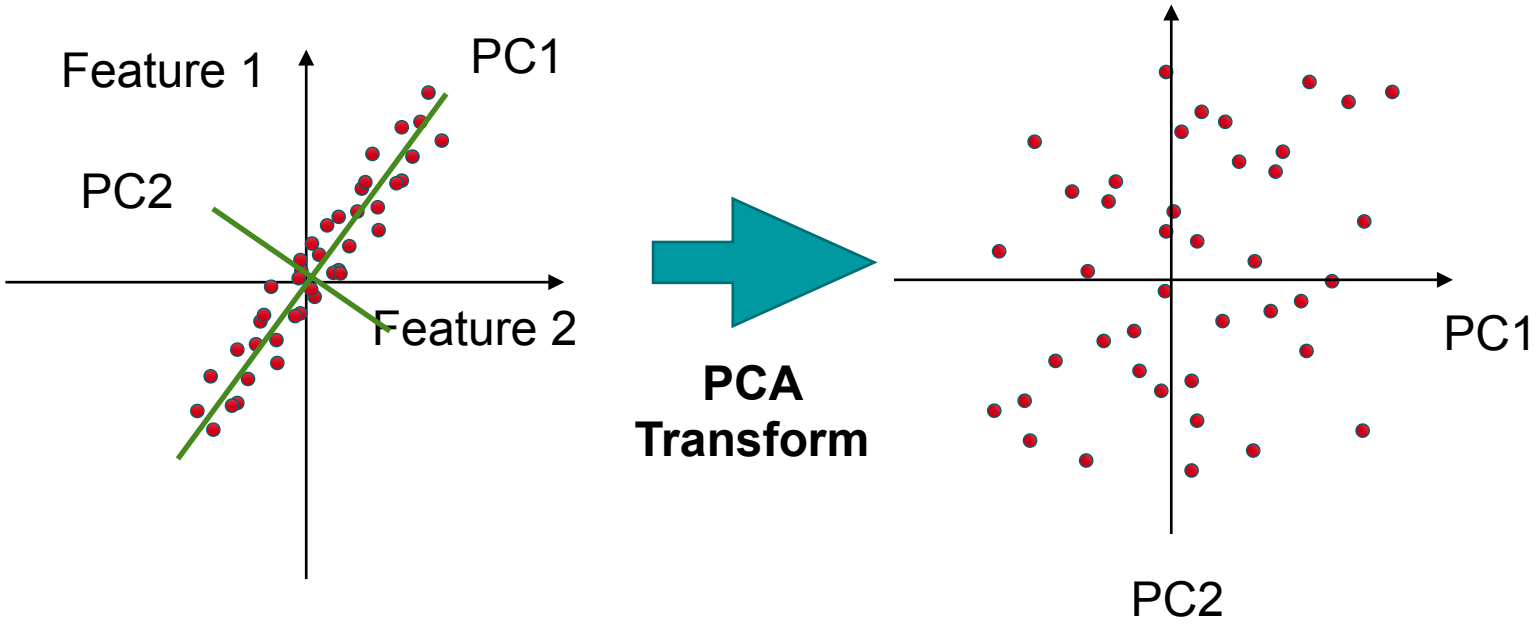
Recommended reading:
Ringner M., *Nature Biotechnology* 26, 303 - 304 (2008)

Principal Component Analysis

- For a data *X*, find linear combination of features (*w)* capturing most of data variance

$$\mathbf{w}_{(1)} = \arg\max_{\|\mathbf{w}\|=1} \{\|\mathbf{X}\mathbf{w}\|^2\}$$

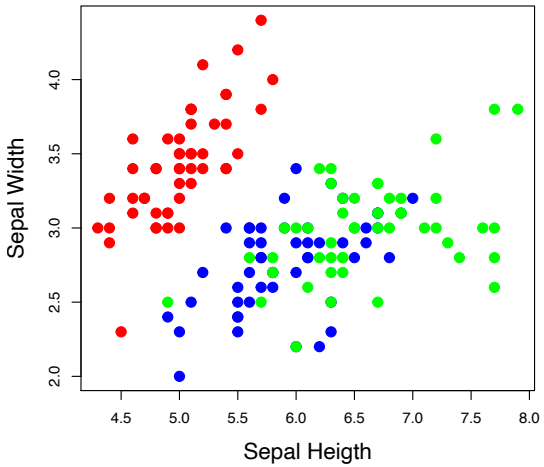
- Can be solved by linear algebra / eigenvector decomposition

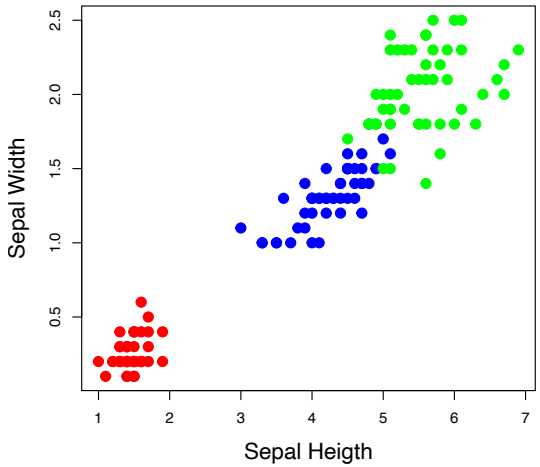


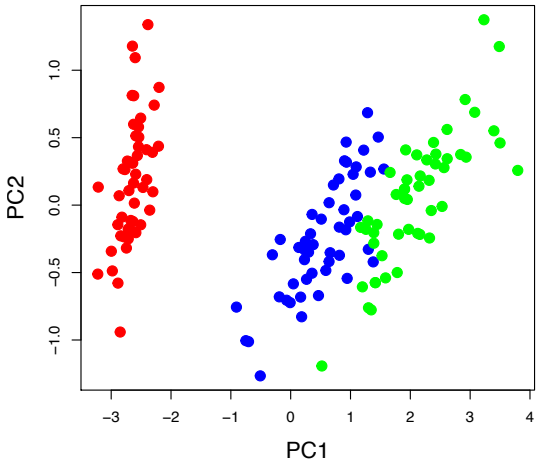
Recommended reading:
Ringner M., *Nature Biotechnology* 26, 303 - 304 (2008)

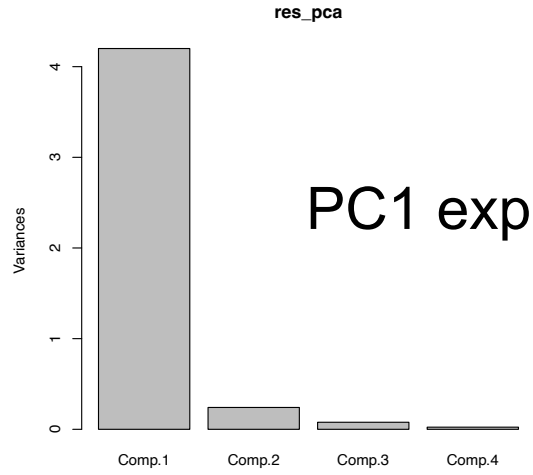
PCA - Iris

- Original iris data had 4 variables







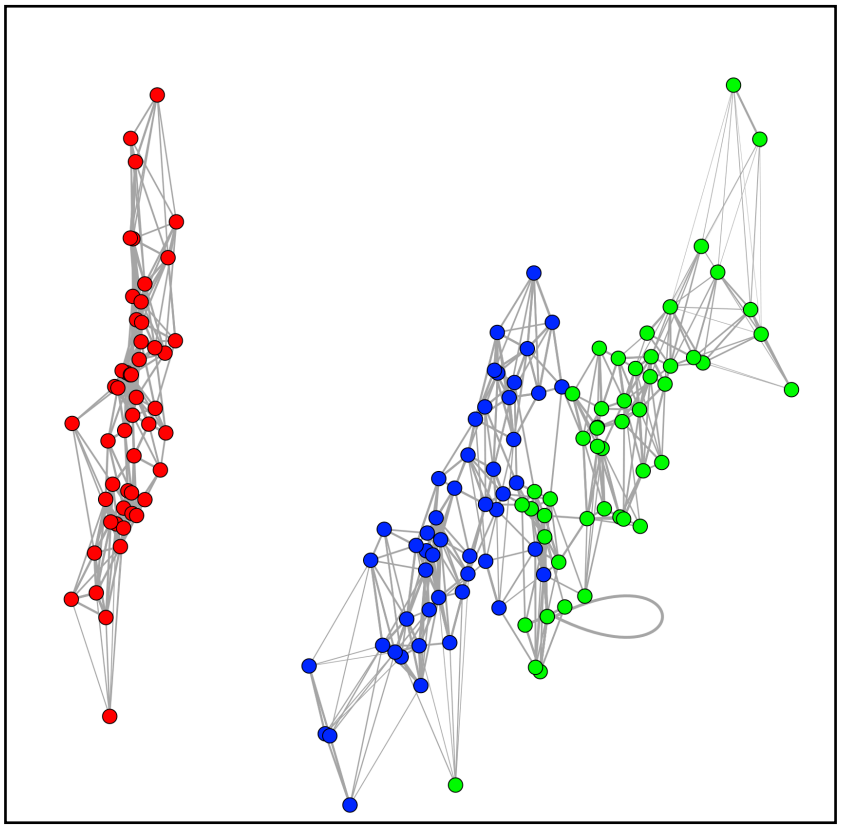


PC1 explains most of variance

Clustering on PCA space

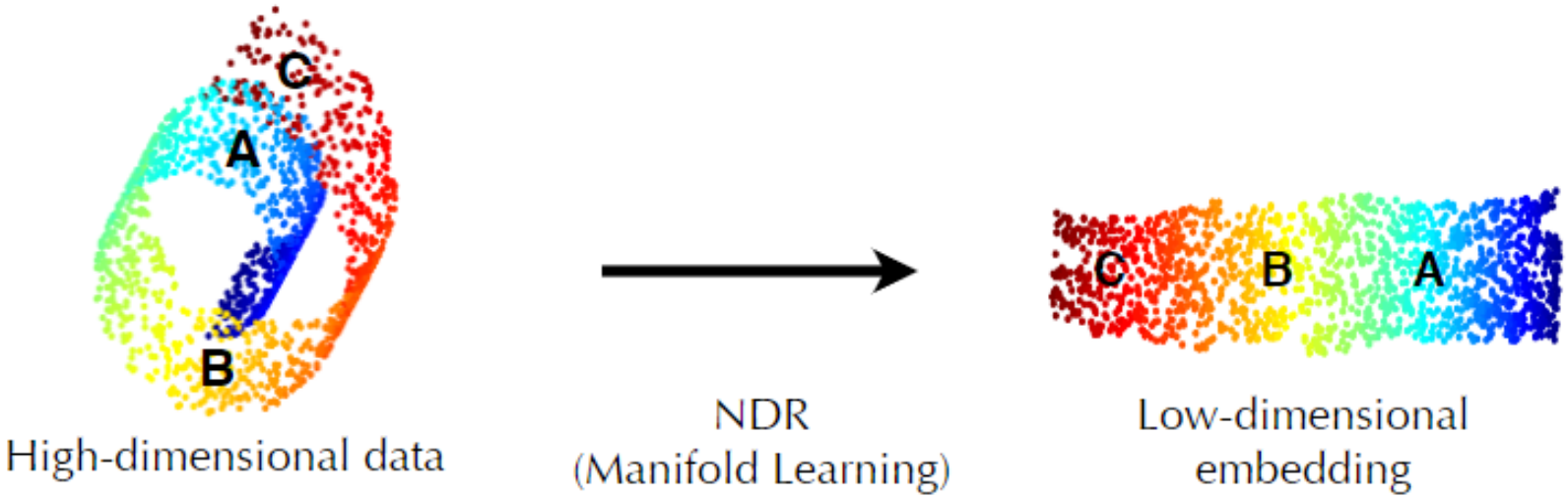
- For single cell data it is usually cluster in PCA space
  - This is crucial for high-dimensional data !

KNN graph of IRIS in PCA space



Non-linear / Manifold methods

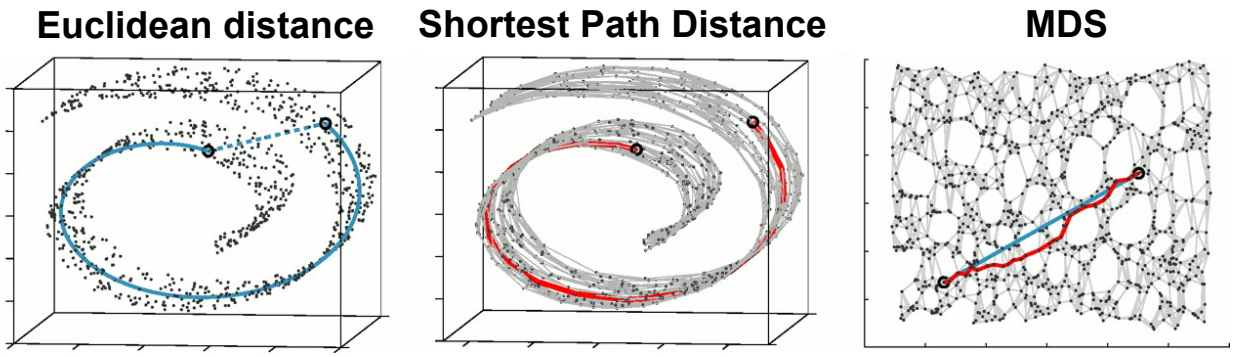
- Data might be distributed at particular regions of a high dimensional space



Adapted from Tenembaum, et al. 2000

Non-linear /Isomap

- Explore topological distance on nearest neighbour graph



Isomap algorithm:
(1) create a *knn* graph
(2) estimate shortest path between nodes (Dijkstra's algorithm)
(3) use multidimensional scaling (MDS) on shortest paths

MDS algorithm:

find vectors $y_1, \ldots, y_n \in Y^N$ such that $\sum_{i,j} (|y_i - y_j| - d_{ij})^2$

where d_{ij} is the similarity between nodes and $N = 2$

Adapted from Tenembaum, et al. 2000

Non-linear methods

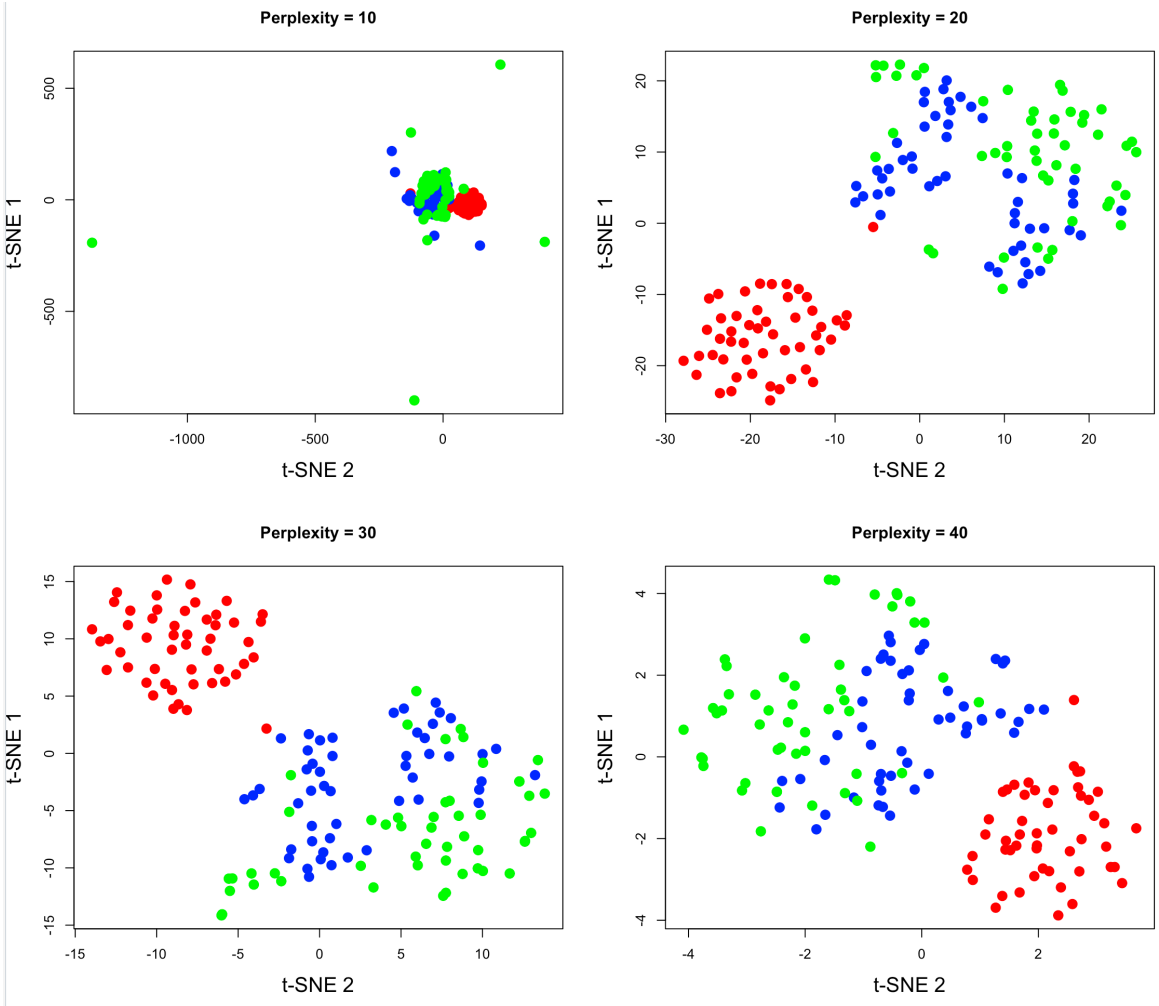
- Current variants of Isomap (t-SNE or UMAP) are often used
- t-SNE - for a given kernel (similarity) ***D*** learn a ***N*** dimensional map ***Y***

$$KL(D \mid Q) = \sum d_{ij} log(\frac{d_{ij}}{q_{ij}}) \qquad \text{where} \qquad q_{ij} = \frac{|y_i - y_j|^2}{\sum_k \sum_l |y_k - y_l|^2}$$

KL - Kullback–Leibler divergence

See for more details: https://www.youtube.com/watch?v=CsUqmug7ZMc

t-distributed stochastic neighbour



- Sensitive to distinct starts and parametrisation
  - Perplexity ~ neighbourhood (*k*) size
- **t-SNE focus on preserving close neighbourhood**

See for more details: https://www.youtube.com/watch?v=9iol3Lk6kyU&t=350s

Non-linear methods

- Variants of Isomap (t-SNE or UMAP) are currently used
- t-SNE - for a given kernel (similarity) ***D*** learn a ***N*** dimensional map ***Y***

$$KL(D\,|\,Q) = \sum d_{ij} log(\frac{d_{ij}}{q_{ij}}) \qquad \text{where} \qquad q_{ij} = \frac{|\,y_i - y_j\,|^2}{\sum_k \sum_l |\,y_k - y_l\,|^2}$$

KL - Kullback–Leibler divergence

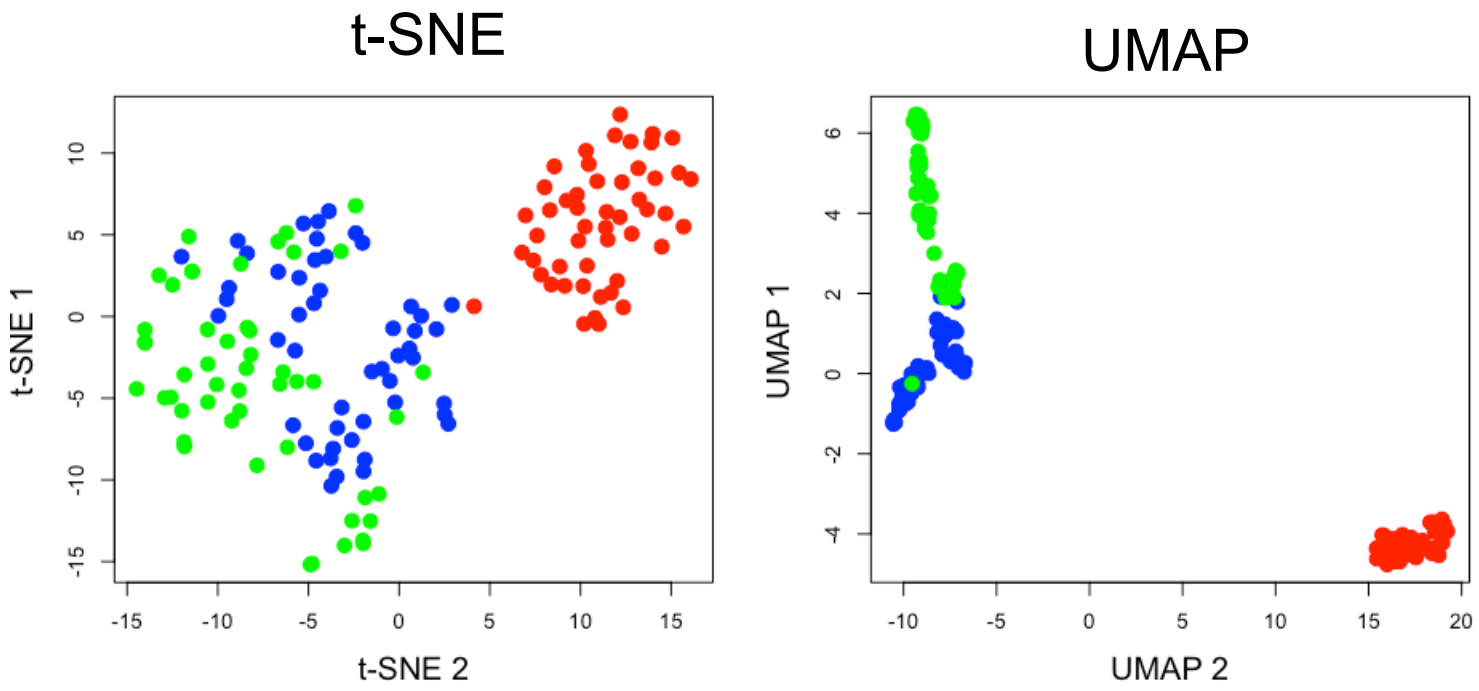
- UMAP - dimension reduction based on Fuzzy Simplicial Sets

$$C((A,\mu),(A,\nu)) = \sum_{a \in A} \mu(a) \log\left(\frac{\mu(a)}{\nu(a)}\right) + (1-\mu(a)) \log\left(\frac{1-\mu(a)}{1-\nu(a)}\right)$$

uses negative samples (non-neighboors)
increasing repulsion between non-neighboors!

See for more details: https://www.youtube.com/watch?v=CsUqmug7ZMc

Manifold learning and IRIS



- Low dimensional visualisation of the data
- Caution: These methods fail capturing global structures (distance between clusters!)

See for more details: https://www.youtube.com/watch?v=9iol3Lk6kyU&t=350s

Resume / Dimension Reduction

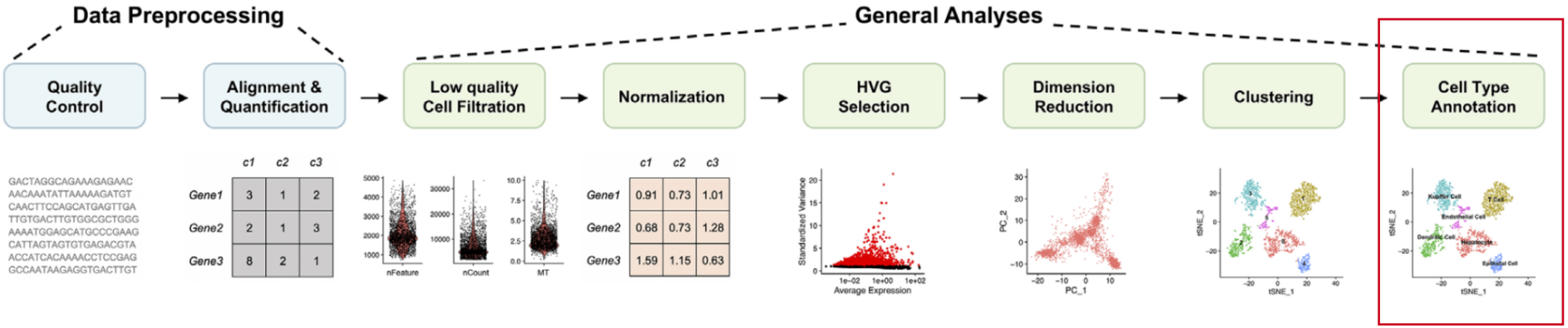
- PCA analysis is a wide spread technique to reduce dimension!
  - Can only capture linear relationships
- Manifold methods
  - Nice low dimensional representation of data
  - Require parametrisation and lose global distance information

Complete course on manifolds/dimension reduction:

https://www.youtube.com/watch?v=evGm6IJKrDI

https://www.youtube.com/watch?v=CsUqmug7ZMc

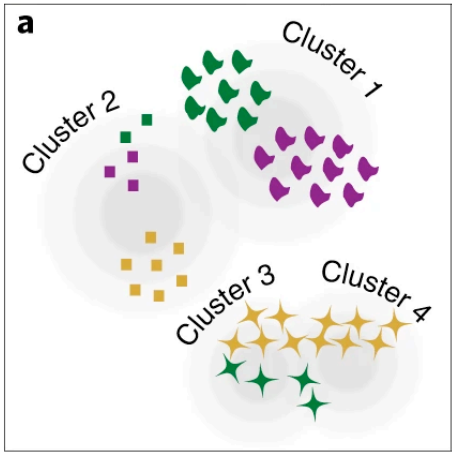
Basics Bioinformatics - single cell RNA-seq



Basic Bioinformatics - Integration

- Usually single cell experiments are performed over distinct conditions
  - comparing disease vs. normal / treated vs. Untreated
- distinct experiments have batch effects, i.e. sequencing depth, cell capture

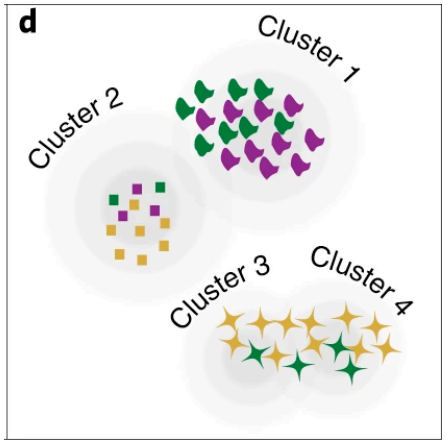
naive integration



- condition specific clusters

- Canonical correlation analysis
- Centroid based correction
- Mutual Information

integrated data

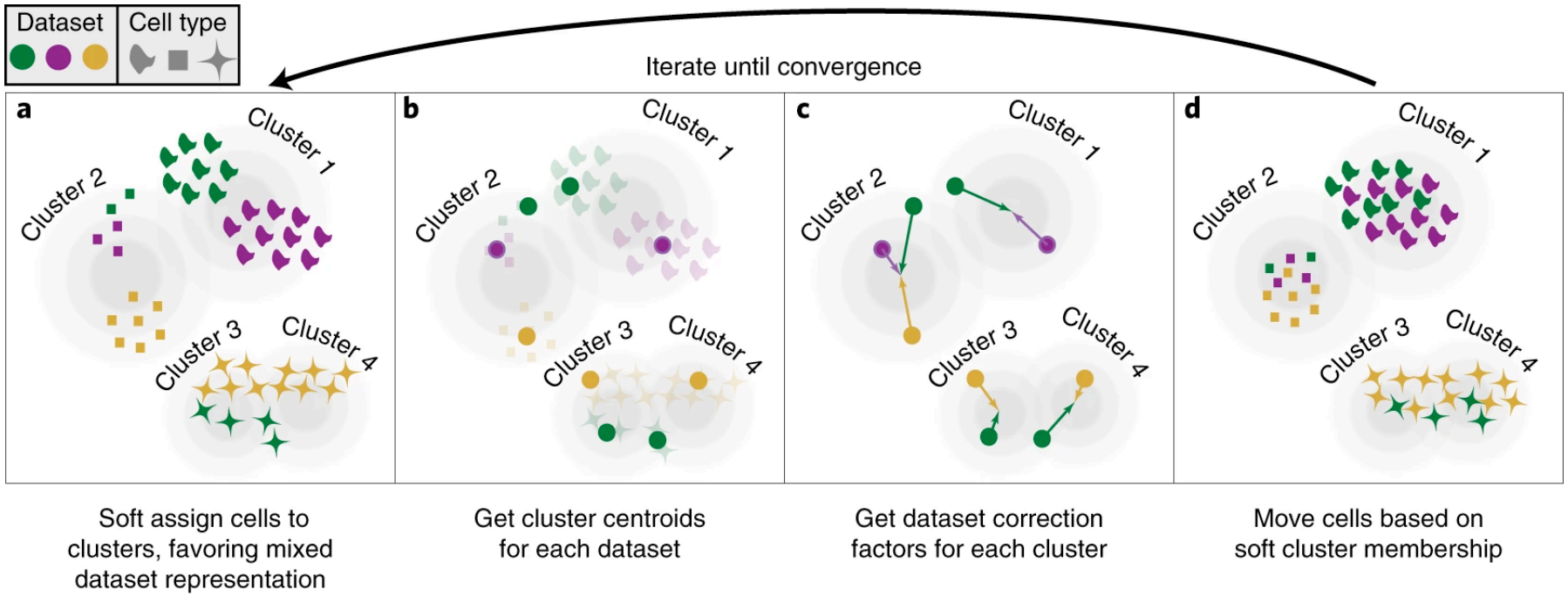


- same/similar cells in same cluster

Adapted from: Korsunsky et al. 2019

Basic Bioinformatics - Integration

- Harmony explores centroids from fuzzy clustering and linear correction models for data integration

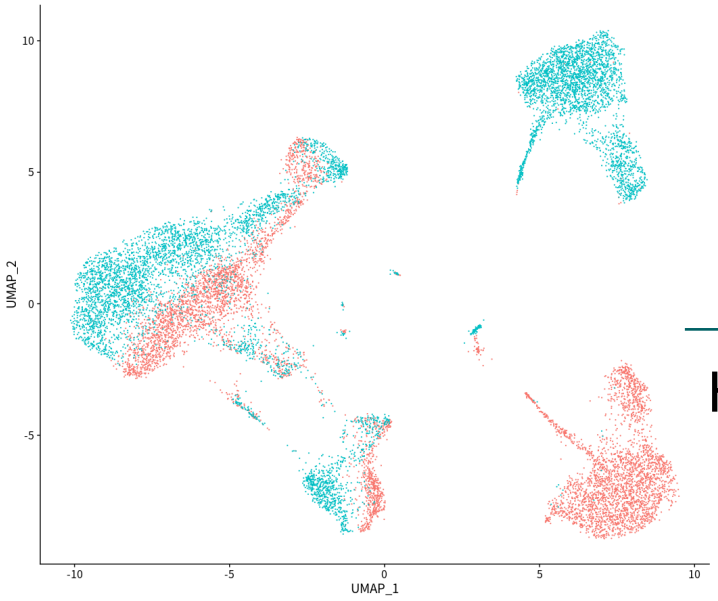


Adapted from: Korsunsky et al. 2019

Basic Bioinformatics - Integration

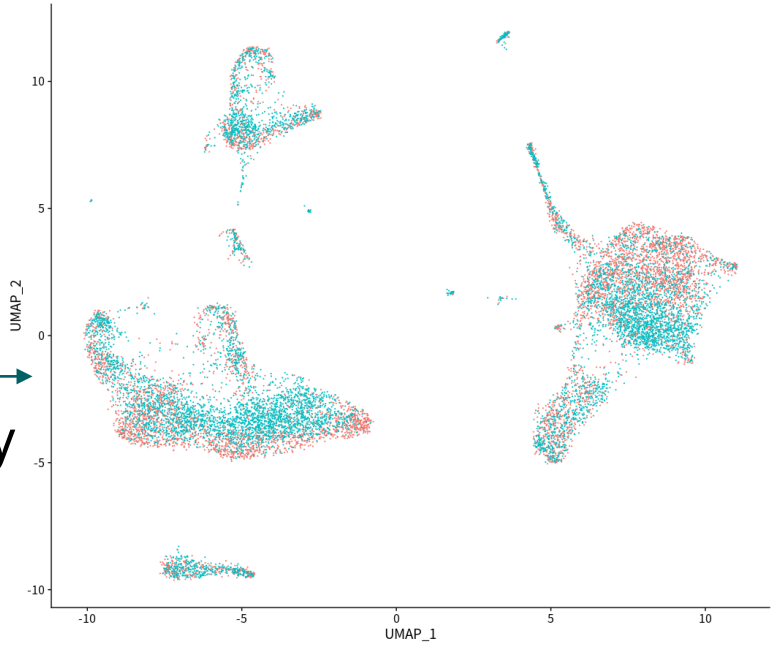
- Real world example: blood cells after stimulation

Raw data



Harmony

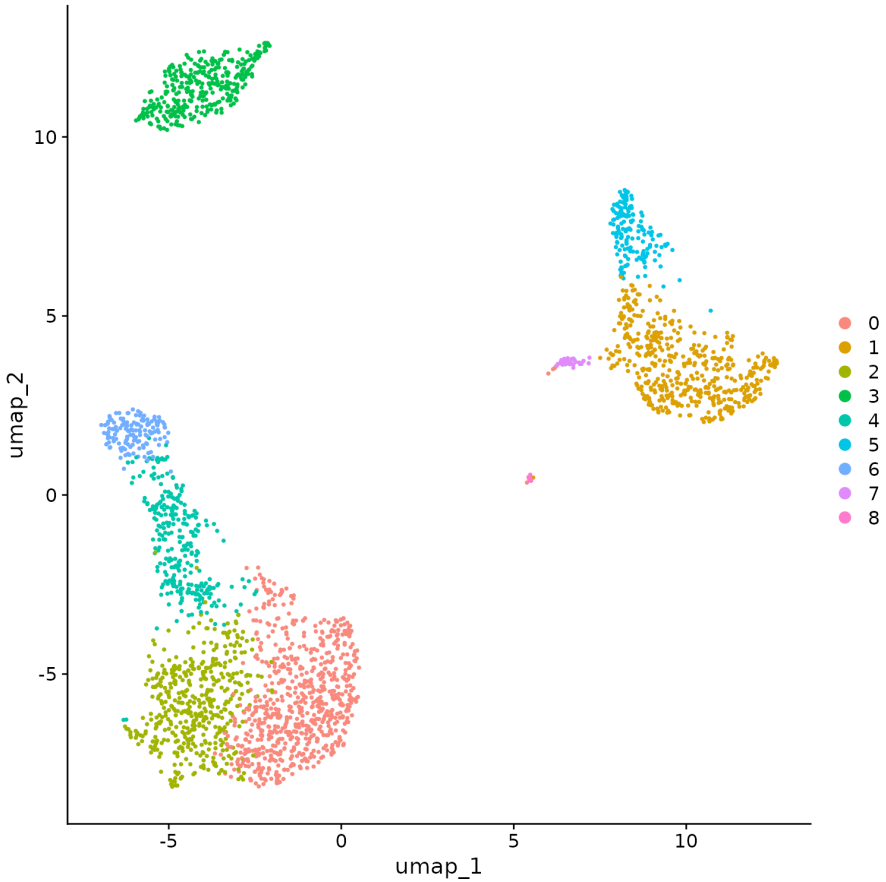
Integrated data



Adapted from: Korsunsky et al. 2019

Basics Bioinformatics - Clustering

Blood cells (PMBC)

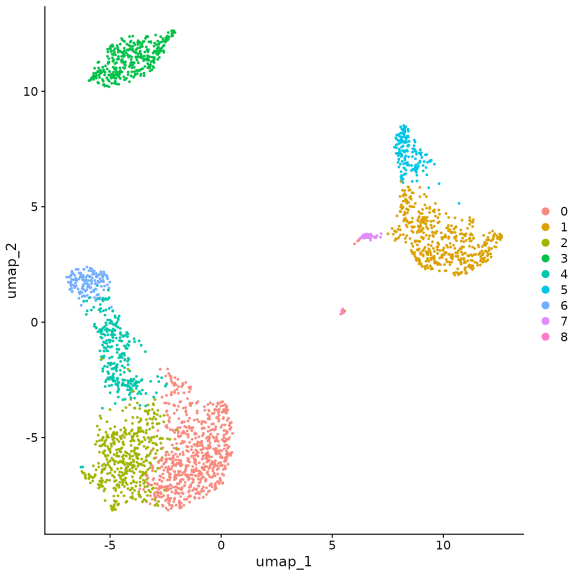


Clustering - identify cells with similar expression patterns

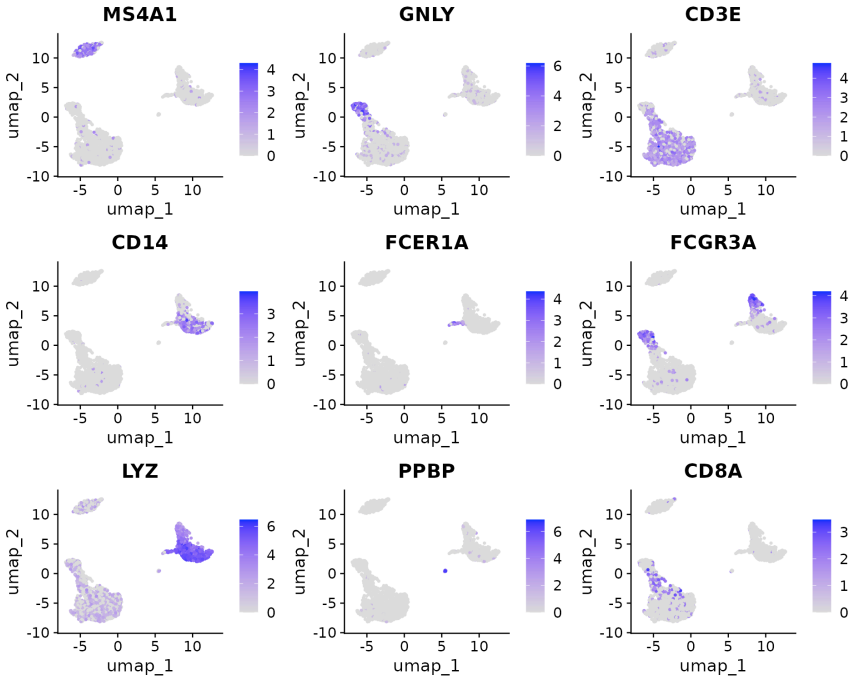
How to identify cell types?

Basics Bioinformatics - Cell Type Annotation

Blood cells (PMBC)



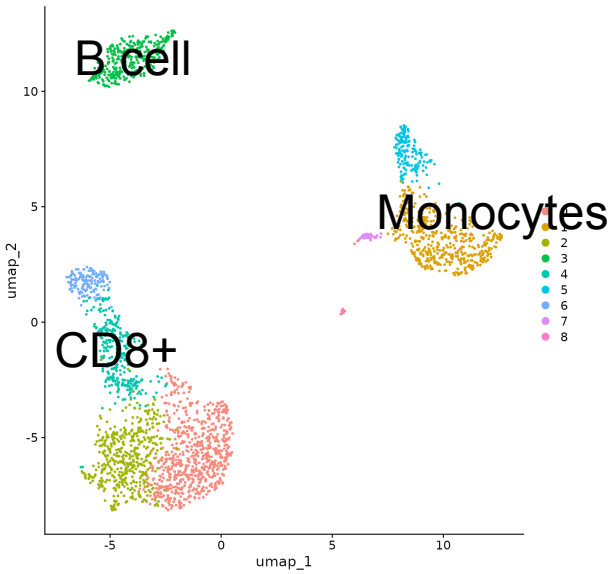
Check expression of known genes:



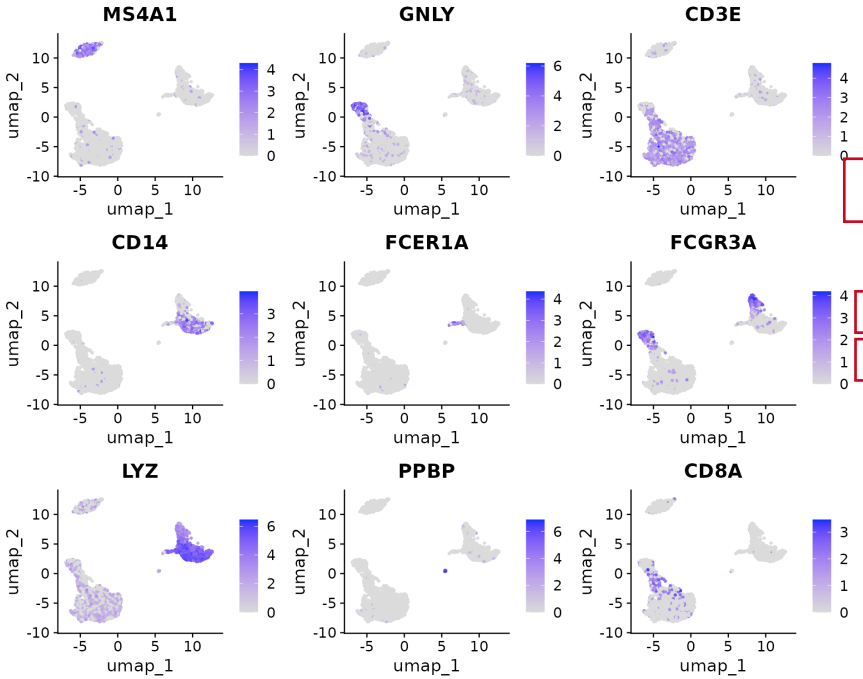
Markers	Cell Type
IL7R, CCR7	Naive CD4+ T
CD14, LYZ	CD14+ Mono
IL7R, S100A4	Memory CD4+
MS4A1	B
CD8A	CD8+ T
FCGR3A, MS4A7	FCGR3A+ Mono
GNLY, NKG7	NK
FCER1A, CST3	DC
PPBP	Platelet

Basics Bioinformatics - Cell Type Annotation

Blood cells (PMBC)



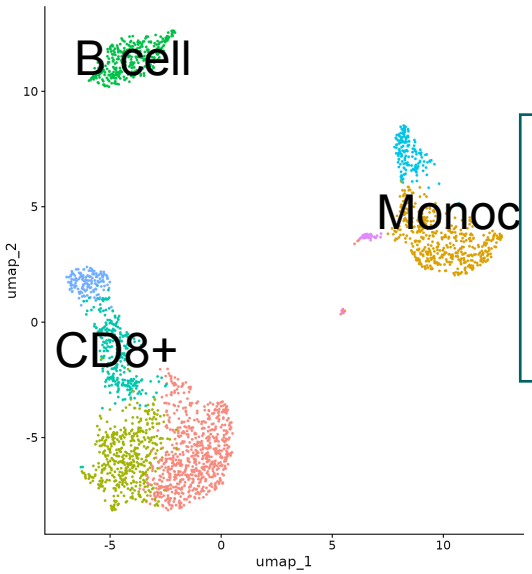
Check expression of known genes:



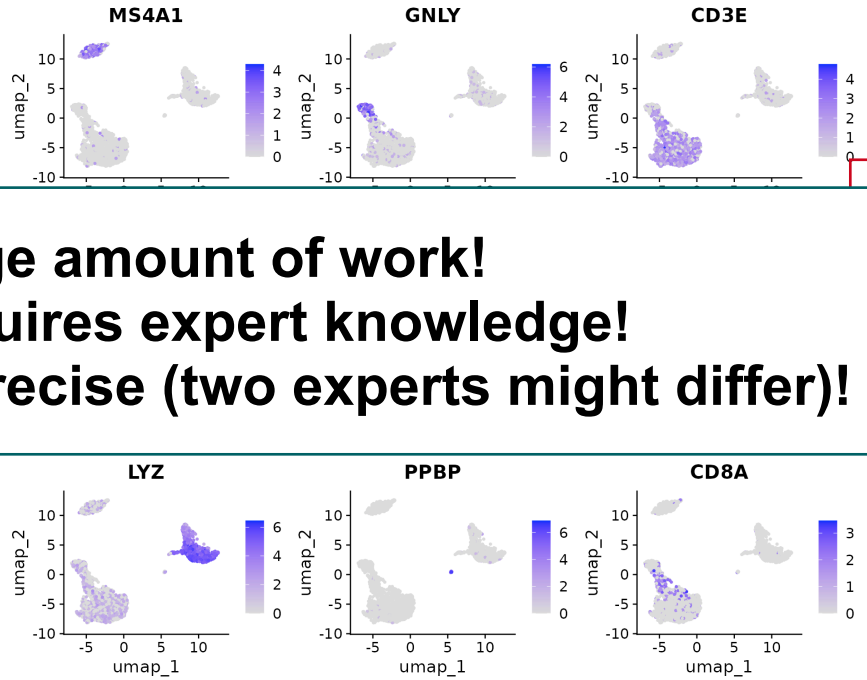
Markers	Cell Type
IL7R, CCR7	Naive CD4+ T
CD14, LYZ	CD14+ Mono
IL7R, S100A4	Memory CD4+
MS4A1	B
CD8A	CD8+ T
FCGR3A, MS4A7	FCGR3A+ Mono
GNLY, NKG7	NK
FCER1A, CST3	DC
PPBP	Platelet

Basics Bioinformatics - Cell Type Annotation

Blood cells (PMBC)



Check expression of known genes:

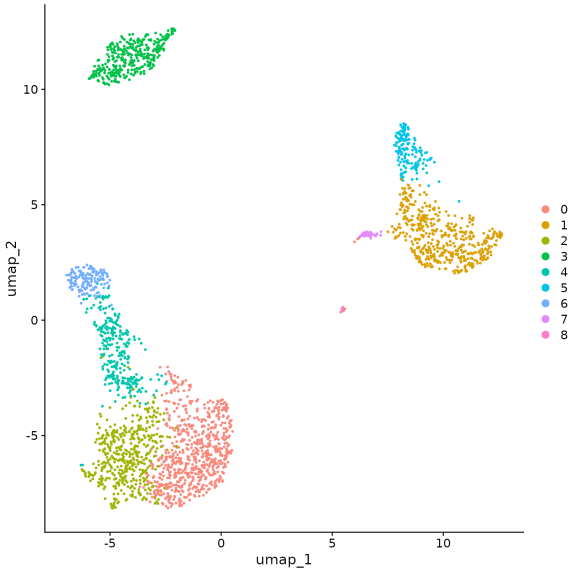


Large amount of work!
Requires expert knowledge!
Imprecise (two experts might differ)!

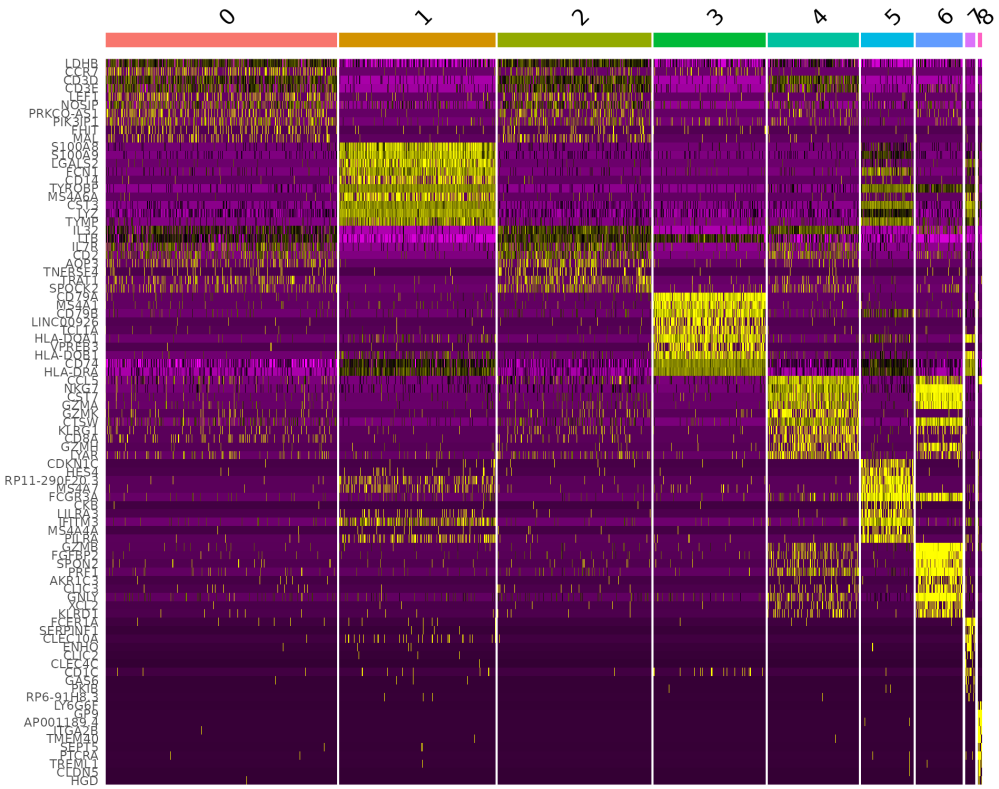
Markers	Cell Type
IL7R, CCR7	Naive CD4+ T
CD14, LYZ	CD14+ Mono
IL7R, S100A4	Memory CD4+
MS4A1	B
CD8A	CD8+ T
FCGR3A, MS4A7	FCGR3A+ Mono
GNLY, NKG7	NK
FCER1A, CST3	DC
PPBP	Platelet

Basics Bioinformatics - Cell Type Annotation

Blood cells (PMBC)

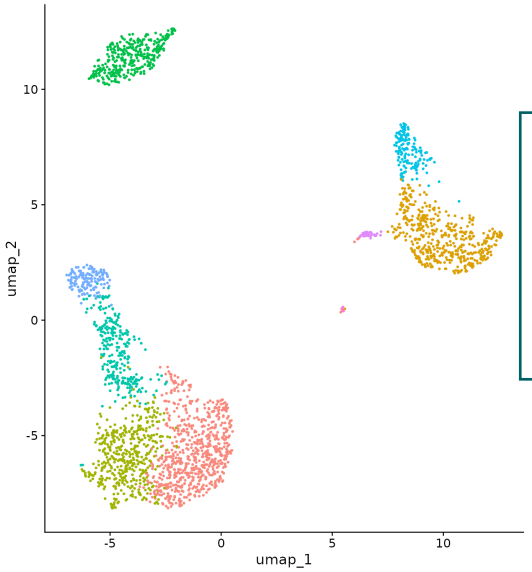


Find features (genes) discriminative of a cluster (via a statistical test)



Basics Bioinformatics - Cell Type Annotation

Blood cells (PMBC)



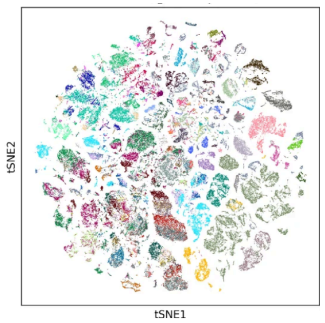
Find features (genes) discriminative of a cluster (via a statistical test)

Requires expert knowledge!

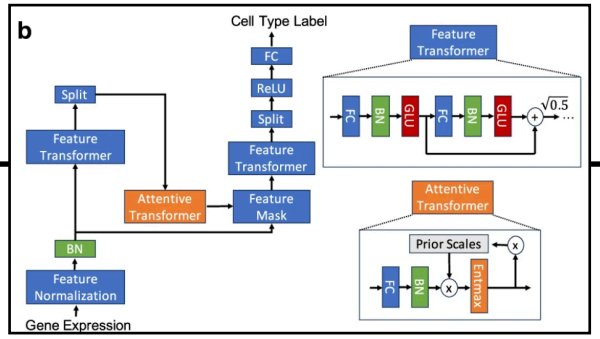
Cell Type Annotation - Cell Type Annotation

ML based annotation

Pan Tissue Cell Atlas

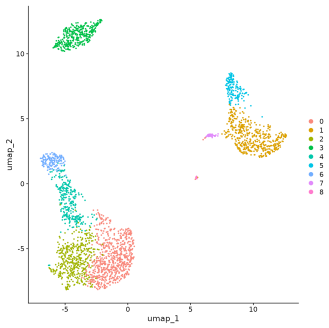


Training of Deep Learning Model

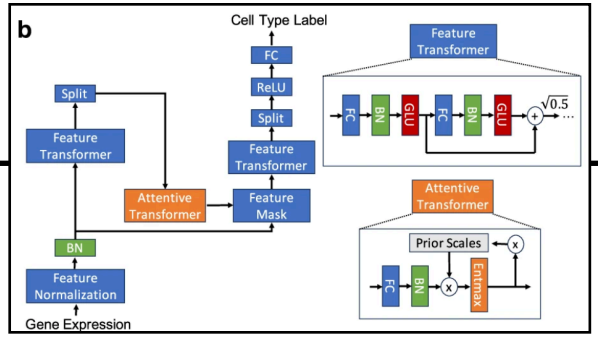


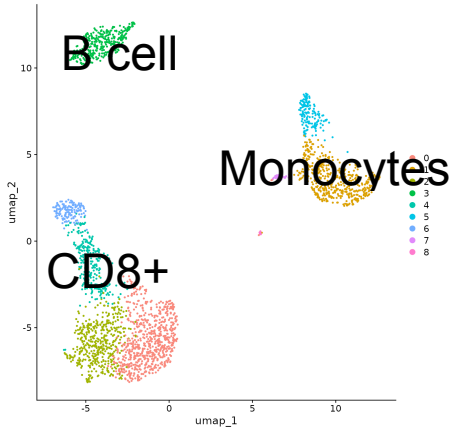
Cell labels

Query Data



Pre-trained Model



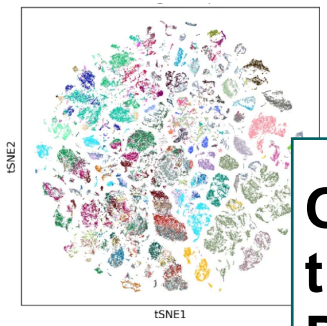


Adapted from Fisher et al. 2023.

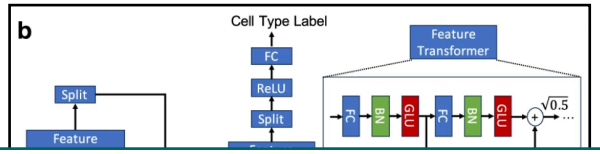
Cell Type Annotation - Cell Type Annotation

ML based annotation

Pan Tissue Cell Atlas



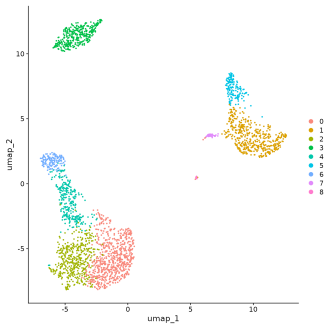
Training of Deep Learning Model



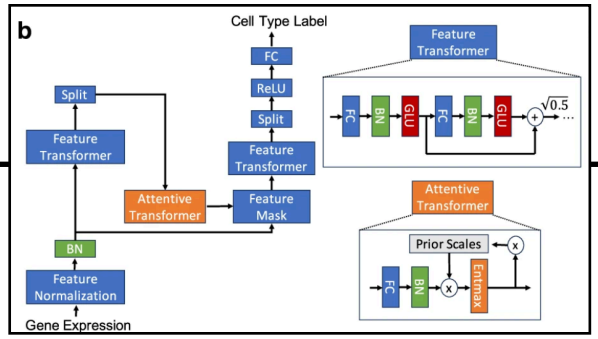
Cell labels

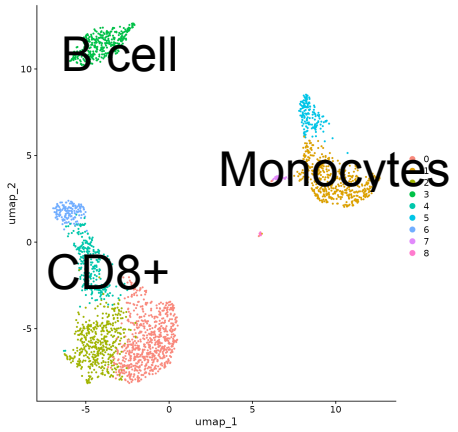
Only works when cell types are in the training data.
Possibly fail for rare cell types

Query Data



Pre-trained Model



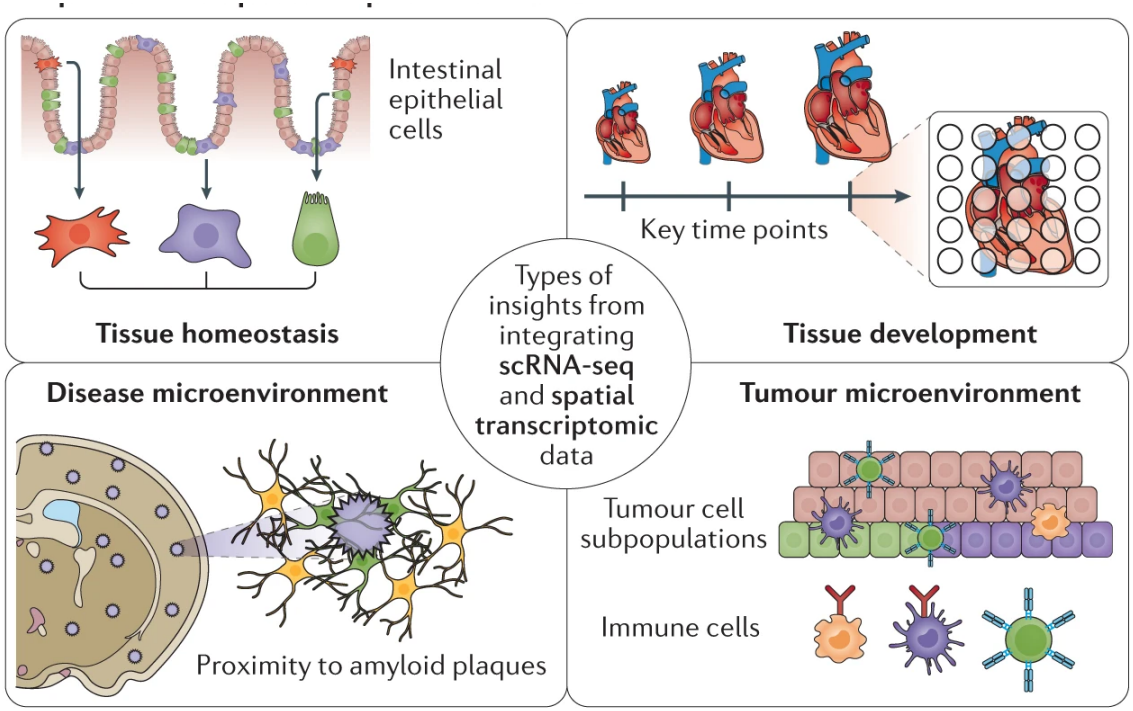


Adapted from Fisher et al. 2023.

Spatially Resolved Transcriptomics

Spatial Transcriptomics

organization of cells is crucial to understand diseases

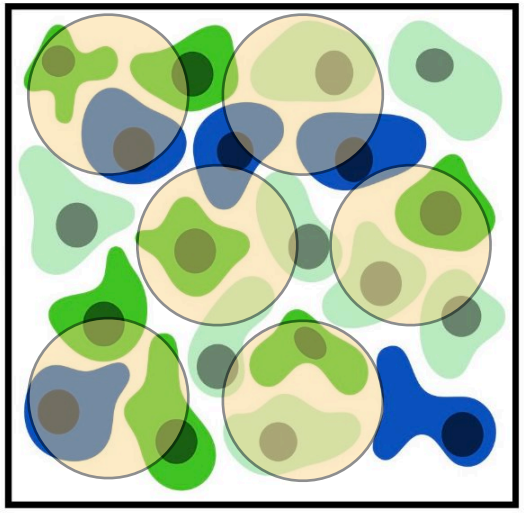


single cell disassociate tissue / spatial information is lost

Adapted from Longo et al. 2021.

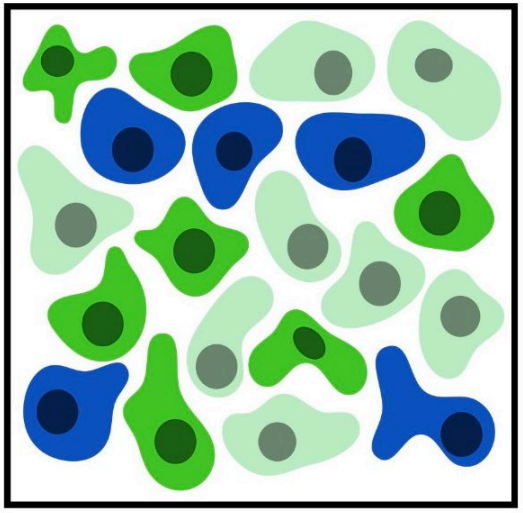
Spatial Transcriptomics - Technologies

Multi-cell resolution



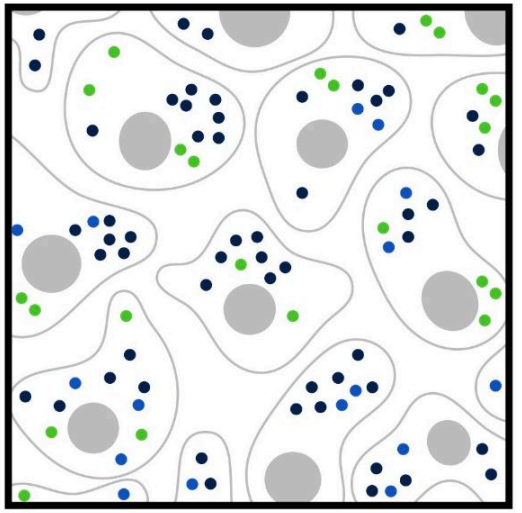
- all genes
- cannot resolve single cells

Single-cell resolution



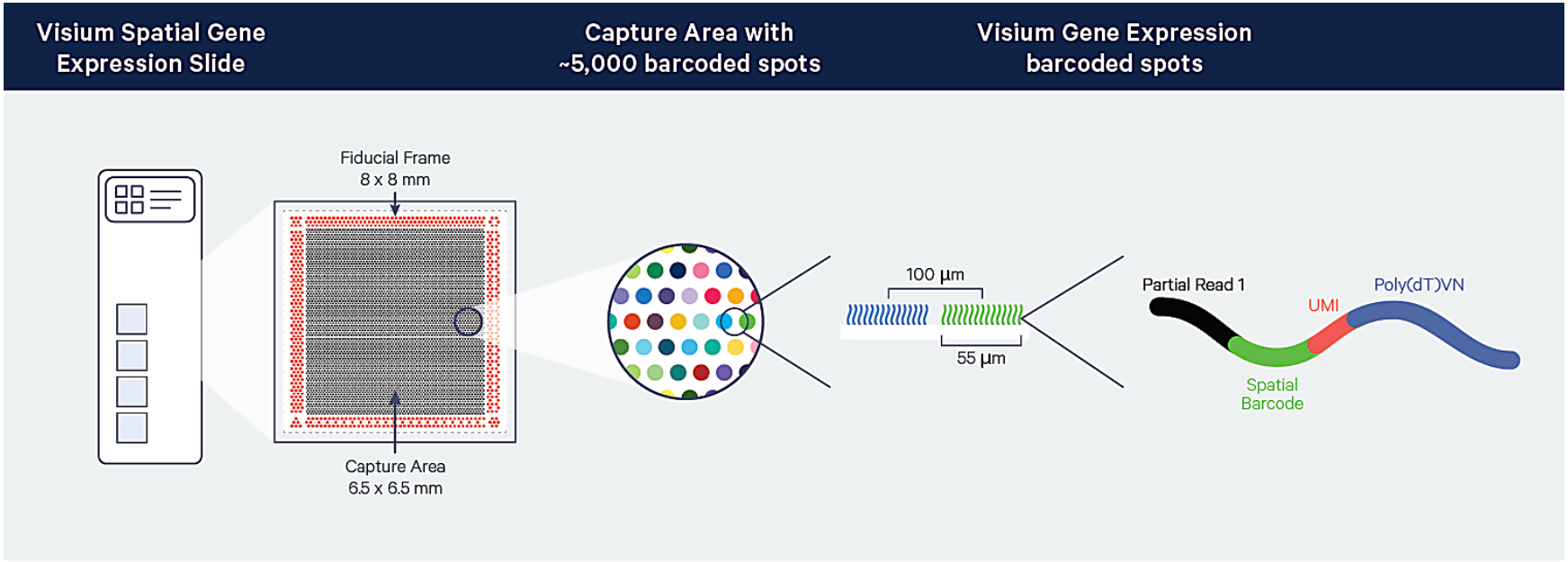
- measure only some genes (100-5000)
- higher costs

Sub-cellular resolution



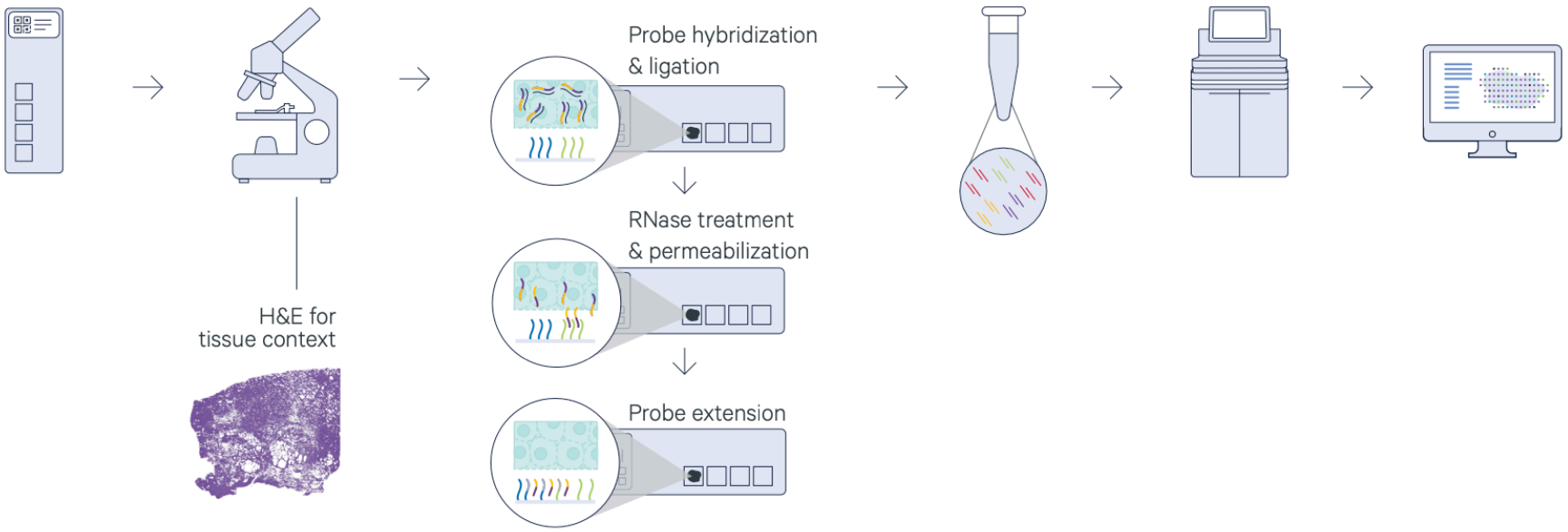
Adapted from https://www.sc-best-practices.org/

Spatial Transcriptomics - Visium 10X



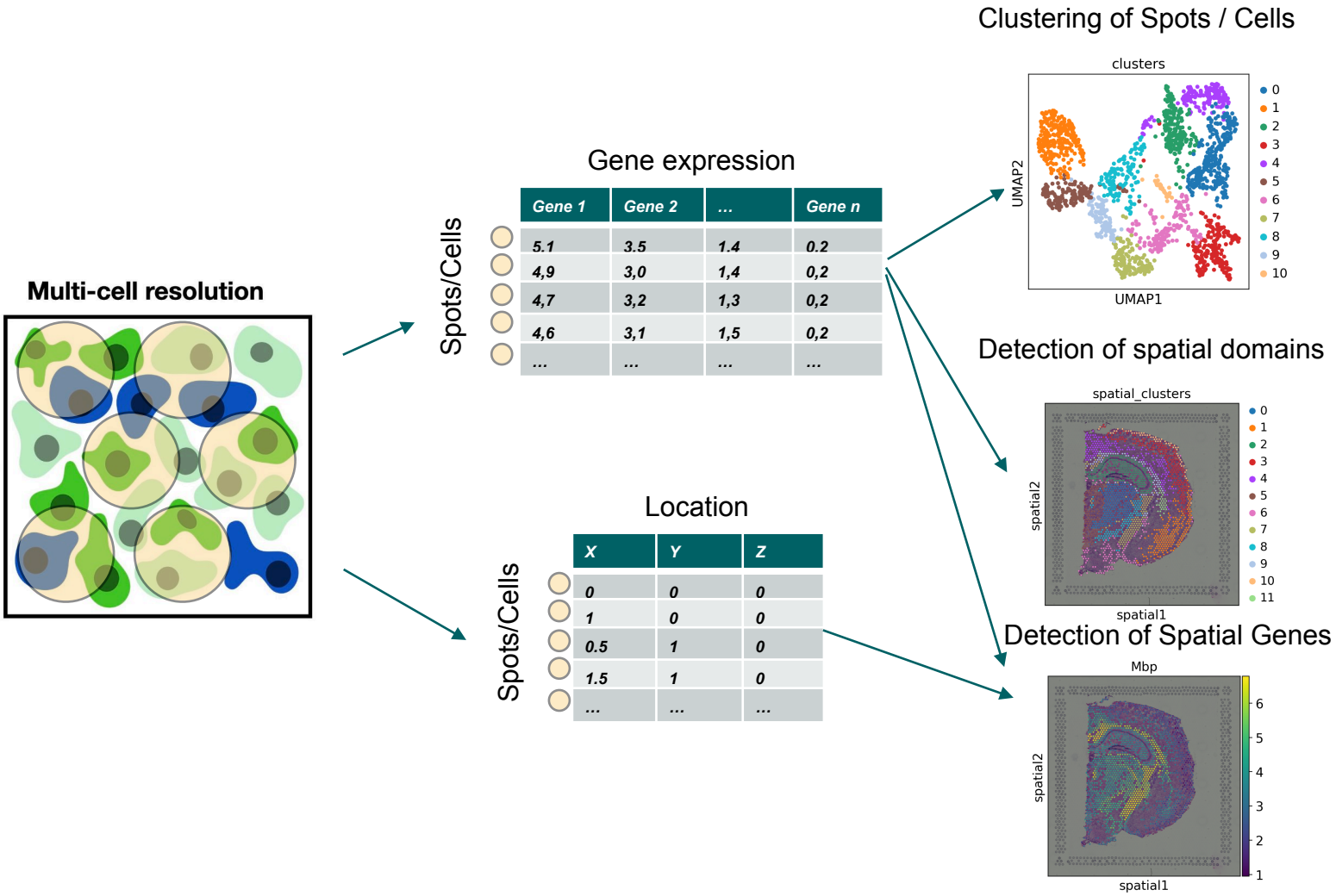
Source: 10x genomics
VIdeo: https://youtu.be/VwNk4d-0RJc

Spatial Transcriptomics - Visium 10X



Source: 10x genomics
VIdeo: https://youtu.be/VwNk4d-0RJc

Spatial Transcriptomics - Technologies

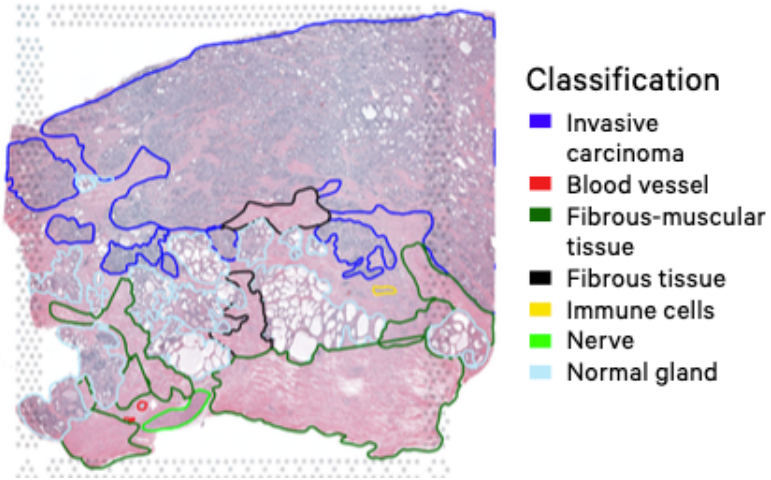


Adapted from https://www.sc-best-practices.org/

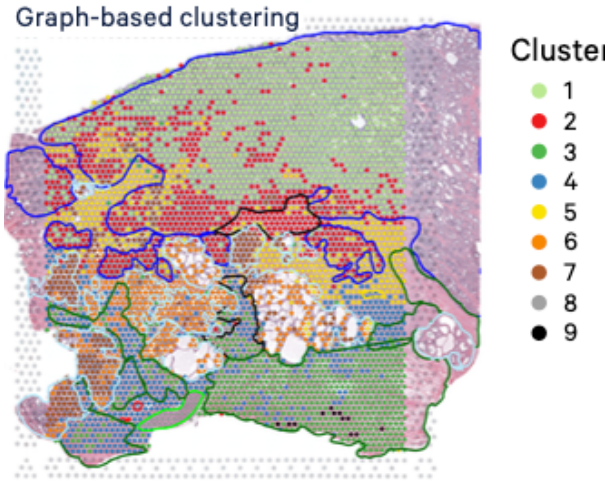
Spatial Transcriptomics - Clinical Relevance

Histology images from tissue sections are used by pathologists for diagnosis of severe diseases

Pathologist Annotation
Rectal Carcinoma



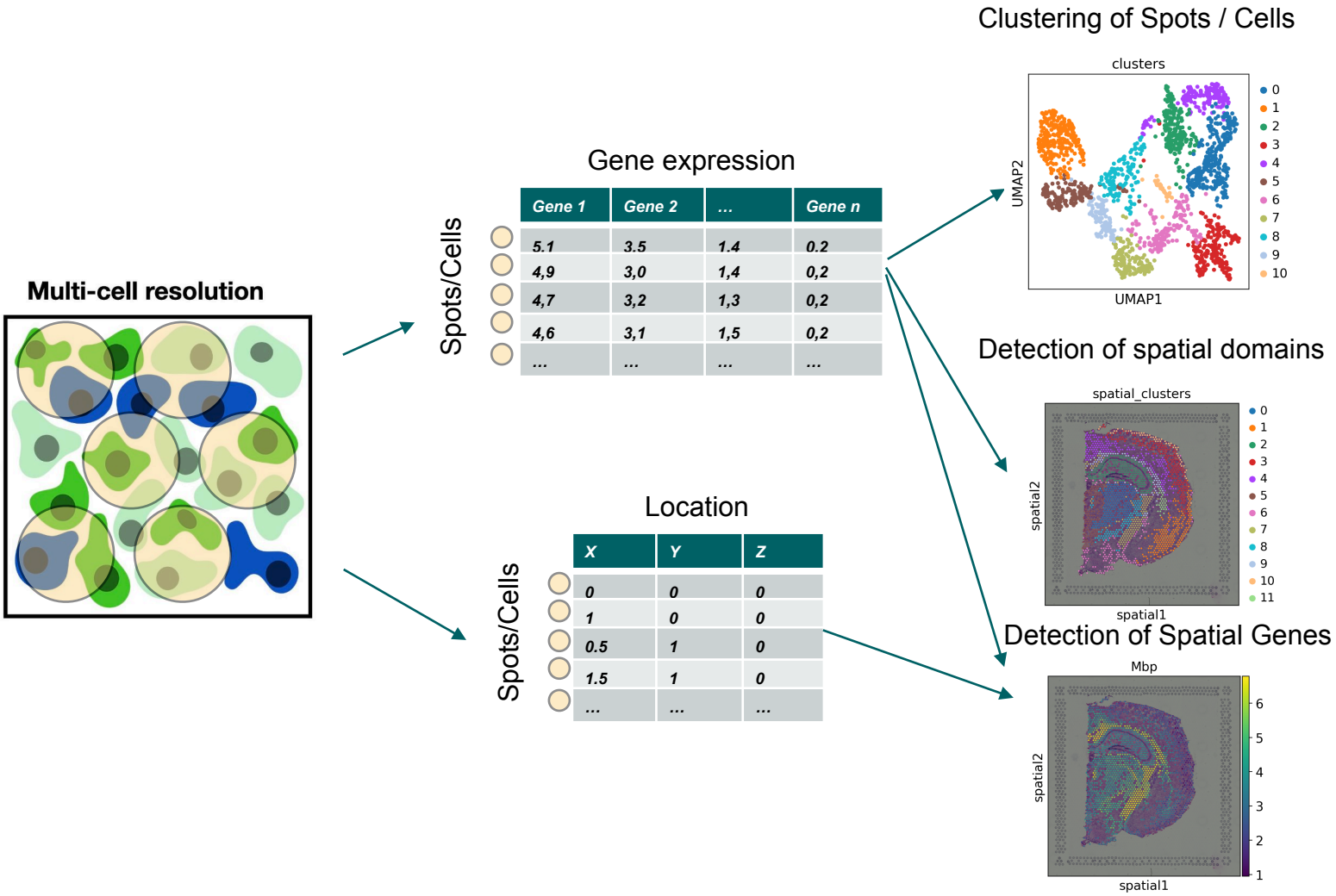
Spatial Domains
Rectal Carcinoma



- pathologists are highly specialized, expensive
- limited/none molecular information from histology
- Diagnosis might differ from pathologists

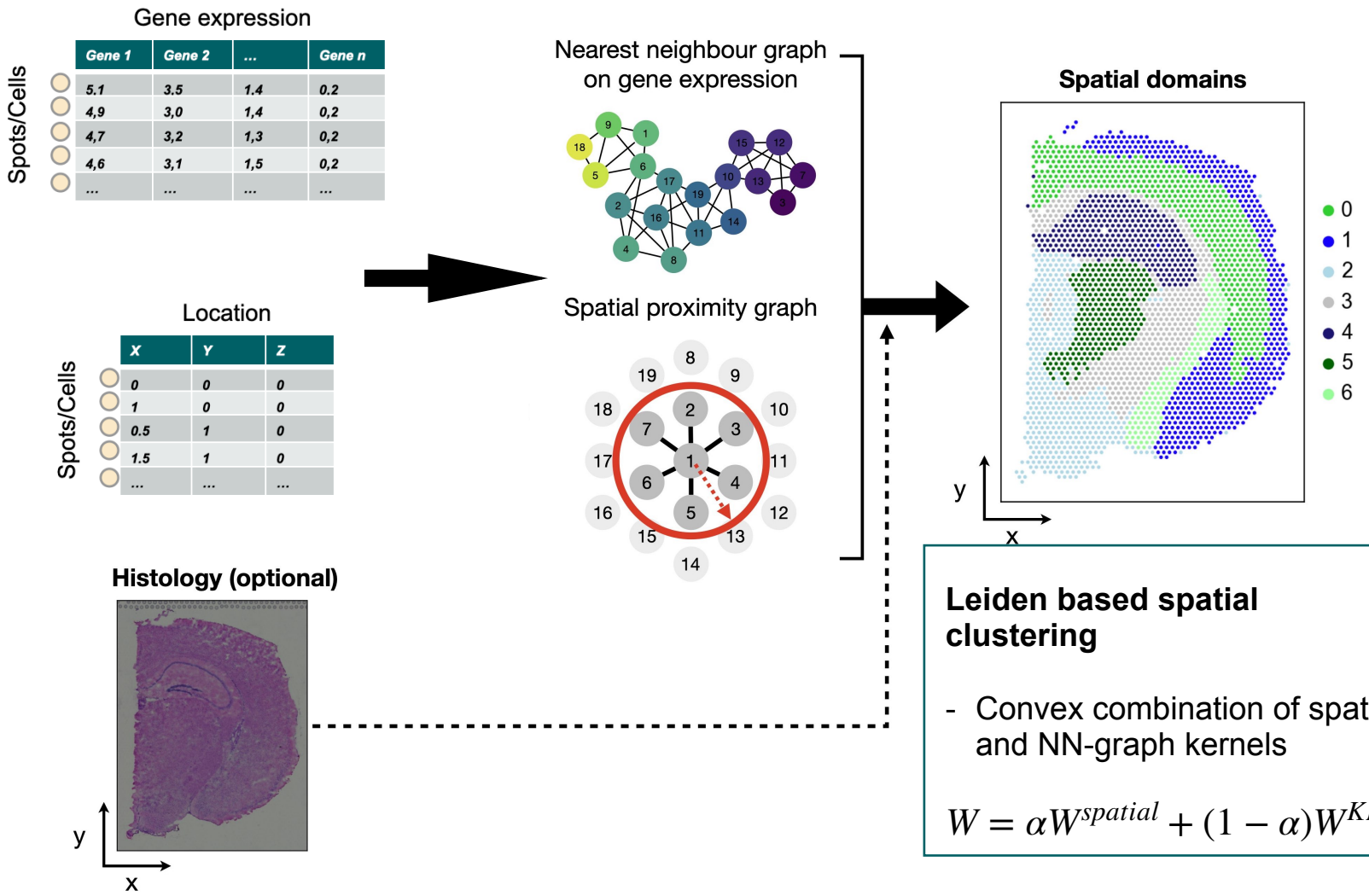
- Richer information than the pathology annotation
  - Find sub-groups of cancers
- Knowning affected genes can help therapy

Spatial Transcriptomics - Technologies



Adapted from https://www.sc-best-practices.org/

Spatial Transcriptomics - Spatial Domains



Adapted from https://www.sc-best-practices.org/

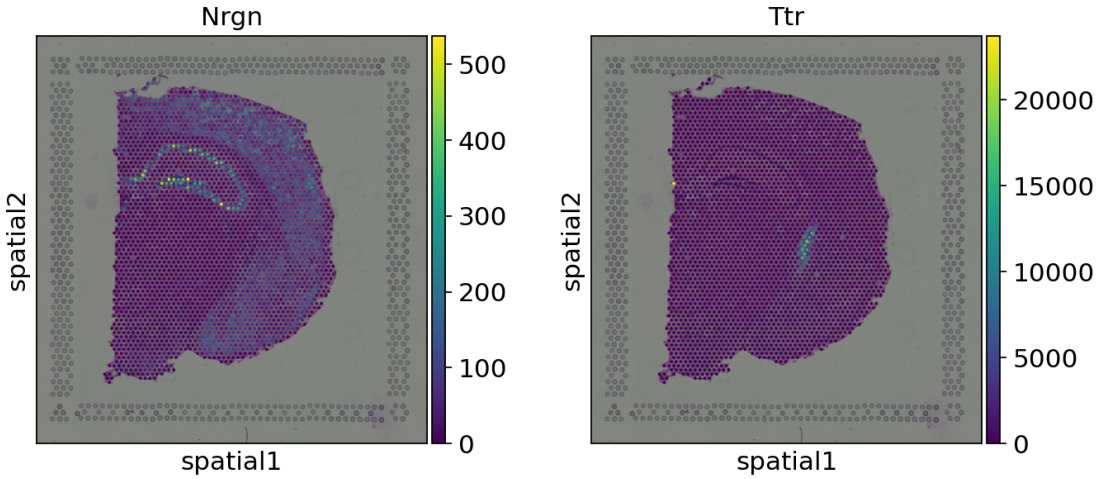
Spatially Variable Genes

Detection of genes with spatially specific expression (independently from the spatial domain analysis).

A common/simple measure is the Moran's I autocorrelation

$$I = \frac{N \cdot \sum_{i=1}^{N} \sum_{j=1}^{N} w_{ij} \cdot x_i \cdot x_j}{W \cdot \sum_{i=1}^{N} (x_i)^2}$$

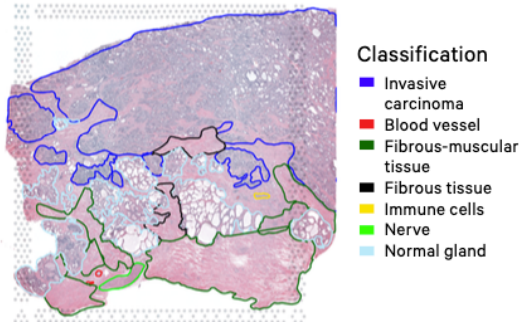
where x is the expression of the gene in spot i, N is the number of cells/spots, and W is an spatial mask, i.e. distance between spots.

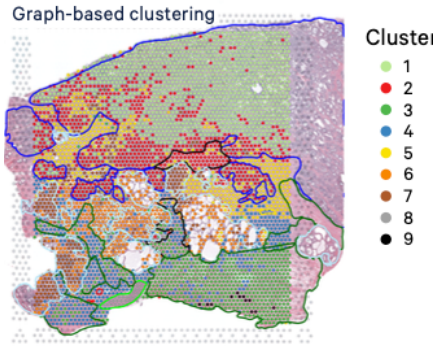


Adapted from https://www.sc-best-practices.org/

Spatial Transcriptomics

- Uncover spatial patterns / crucial for understanding diseases
- Computational approach need to consider addition spatial (or histology) information





Calendar

7.4 – Introduction to Bioinformatics and Single Cell Sequencing Analysis

14.4 – Single Cell Sequencing Analysis (cont.) & Practice

28.4 – Introduction to HPC clusters and GPU / Project Proposal

28.4 - 7.7 – Project development

14.7 – Project Presentation

Communication/discord channel:

https://discord.gg/qtaqUDQP8S

Thank you!