

# Single Cell RNA-seq Clustering

---

Marcel Reinders

Delft Bioinformatics Lab, TU Delft

Leiden Computational Biology Center, LUMC



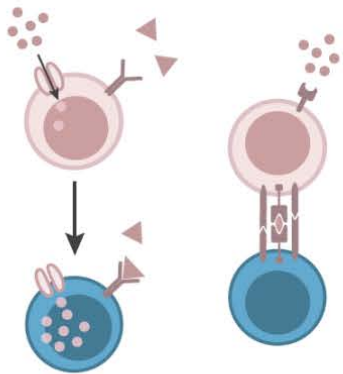
**Leiden**

Computational Biology Center

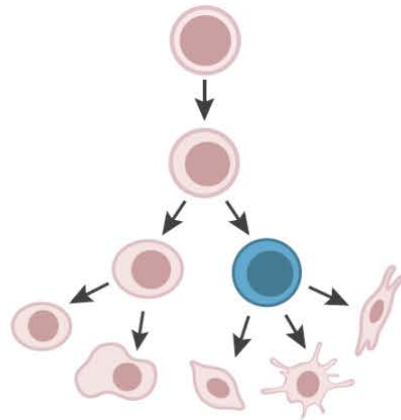


# Cell Identity determined by diverse factors

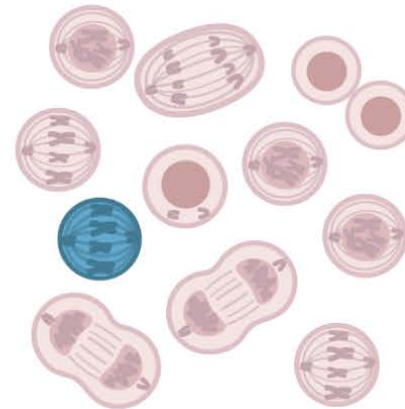
Environmental stimuli



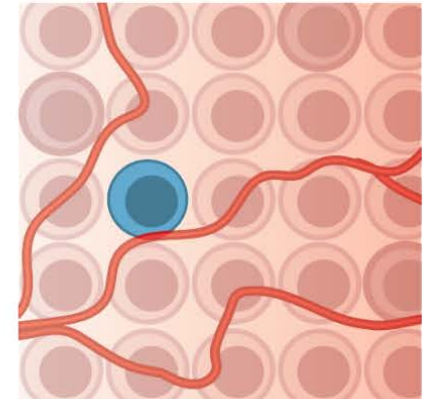
Cell development



Cell cycle

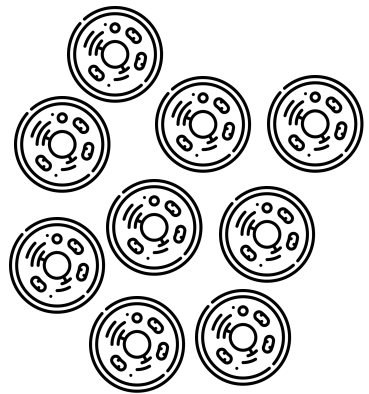


Spatial context



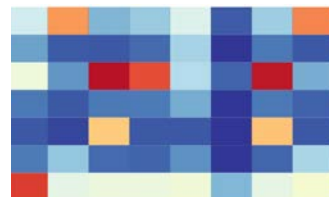
# How can we *automatically* identify cell populations?

Mystery cells

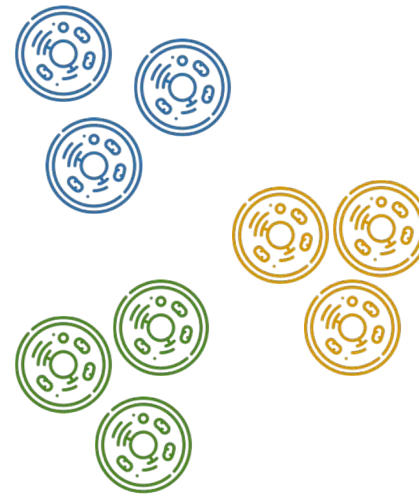


Measure

Cell #1  
Cell #1  
.  
.  
Cell #N

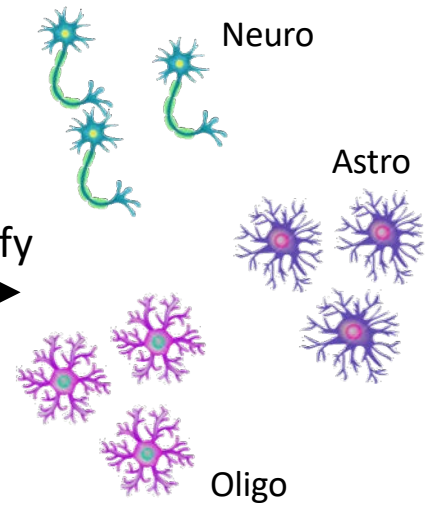


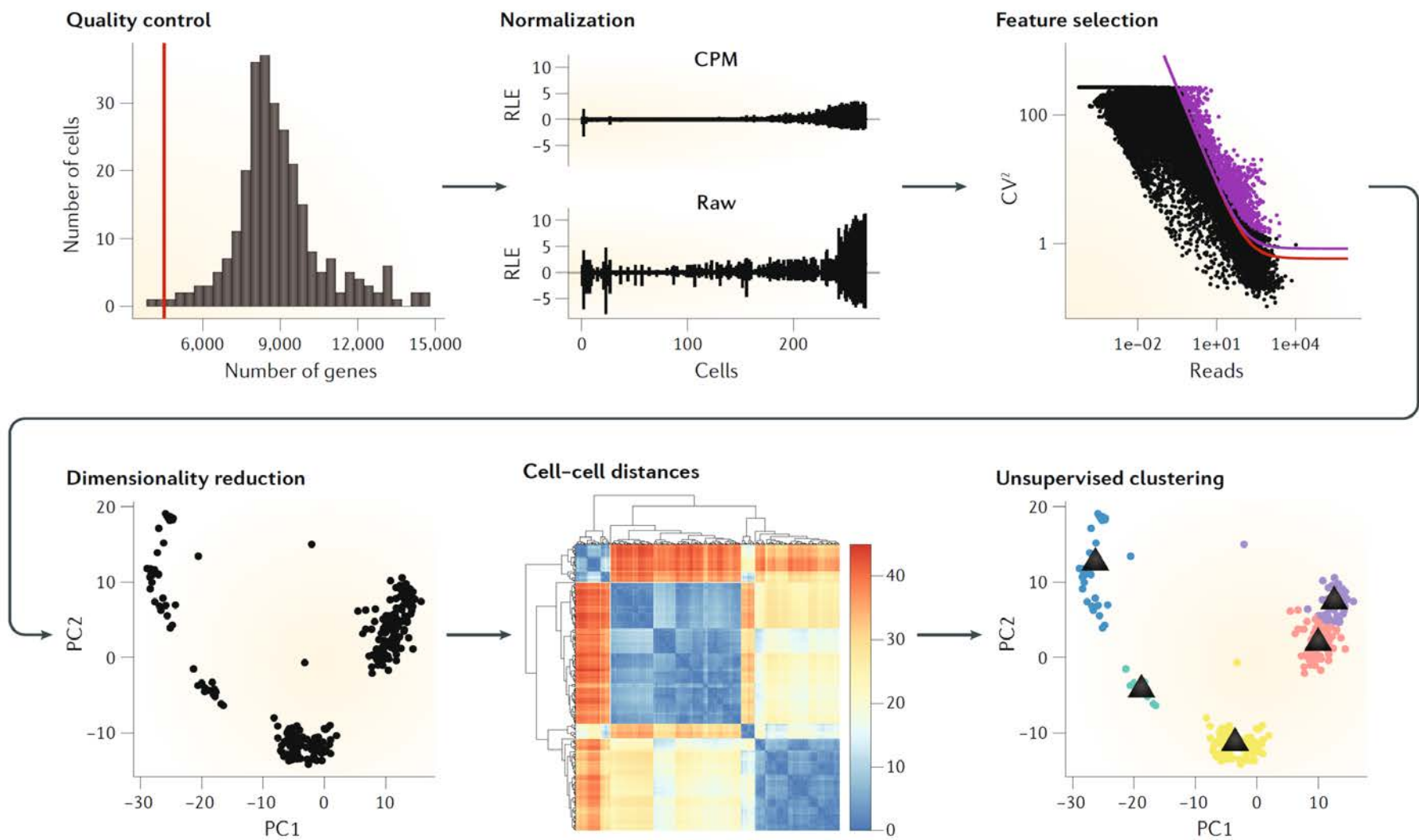
Group

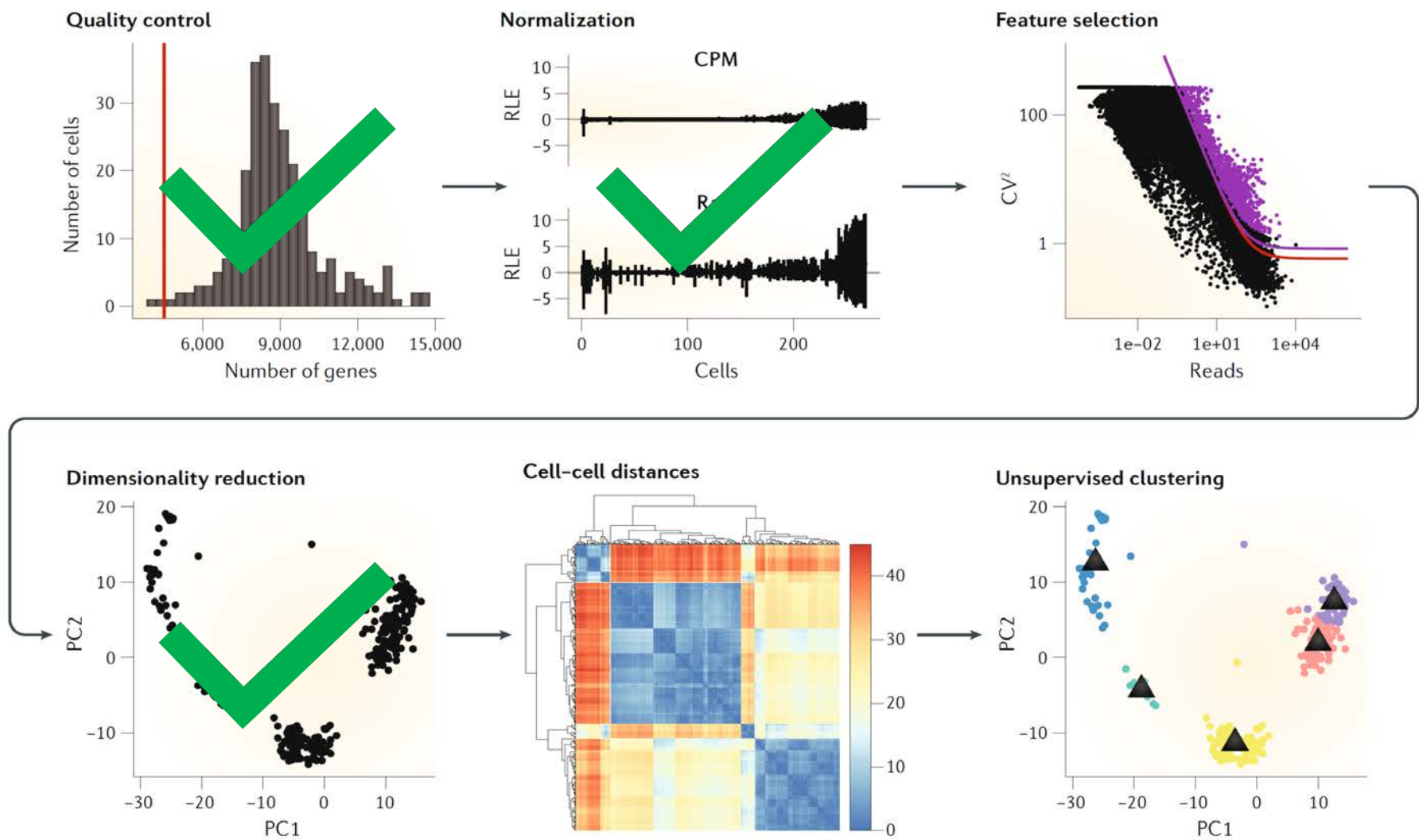


Identify

Cell Populations





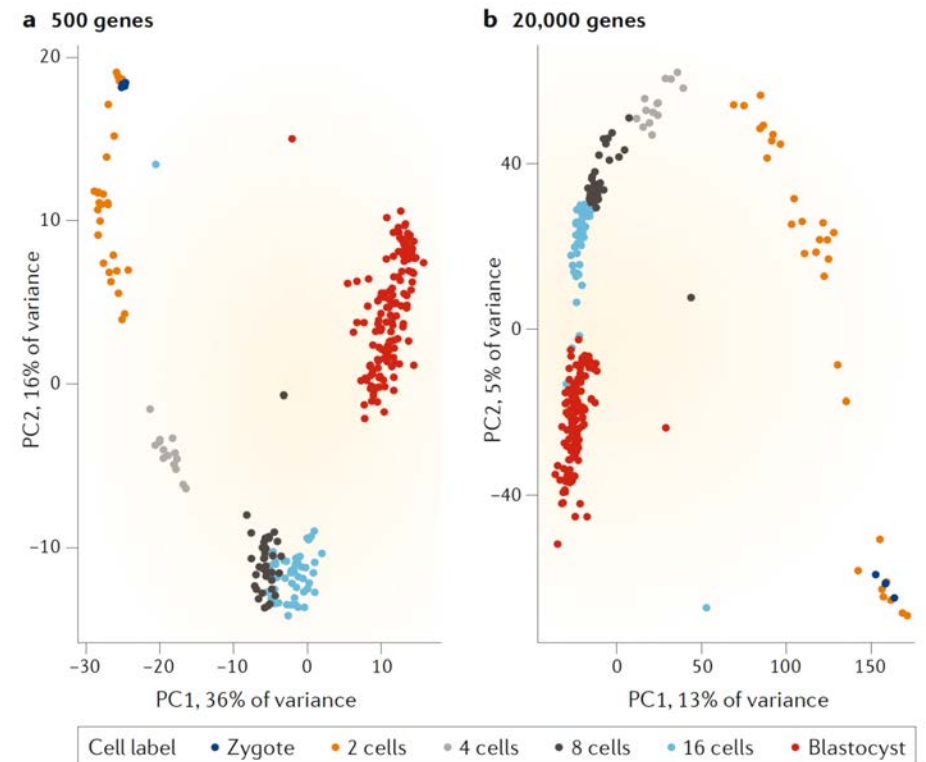


# Outline

- **Feature selection**
- Introduction to clustering
  - Hierarchical clustering
  - $k$ -Means clustering
  - Graph-based clustering
- Validation
- scRNA-seq clustering
  - Single Cell Consensus Clustering (SC3)
  - Seurat

# Feature selection

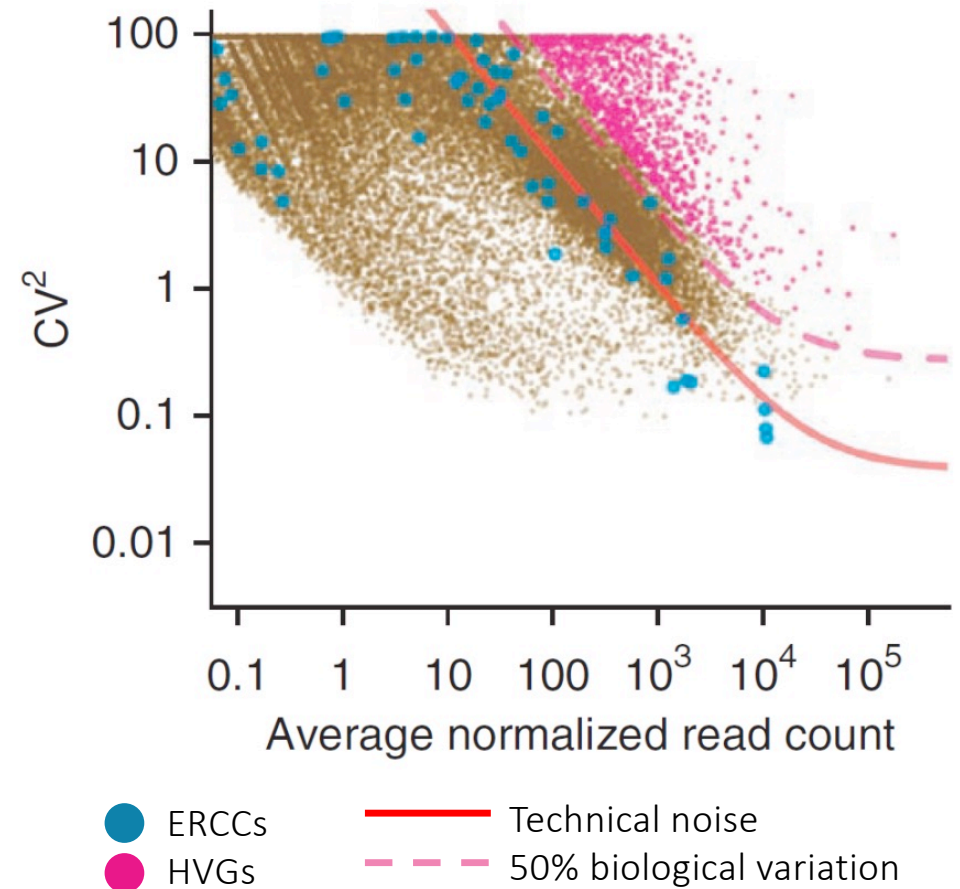
- Curse of dimensionality
  - More features (genes) -> noise dominates distances between samples (cells), effectively all cells get 'same' distance
- Remove genes which only exhibit technical noise
  - Increase the signal:noise ratio
  - Reduce the computational complexity



# Feature selection

## *Highly Variable Genes (HVG)*

- $CV = \frac{var}{mean} = \frac{\sigma}{\mu}$
- Fit a gamma generalized linear model to spike ins (ERCCs)
- No ERCCs?  
Estimate technical noise based on all genes





# Feature selection

## *M3Drop: Dropout-based feature selection*

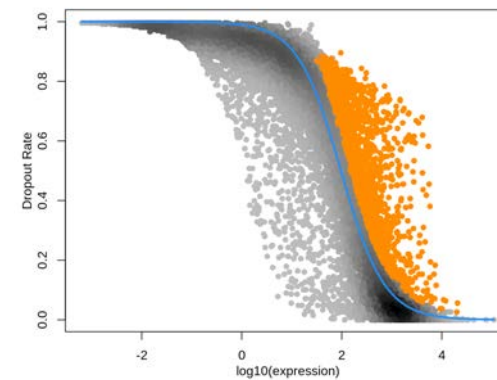
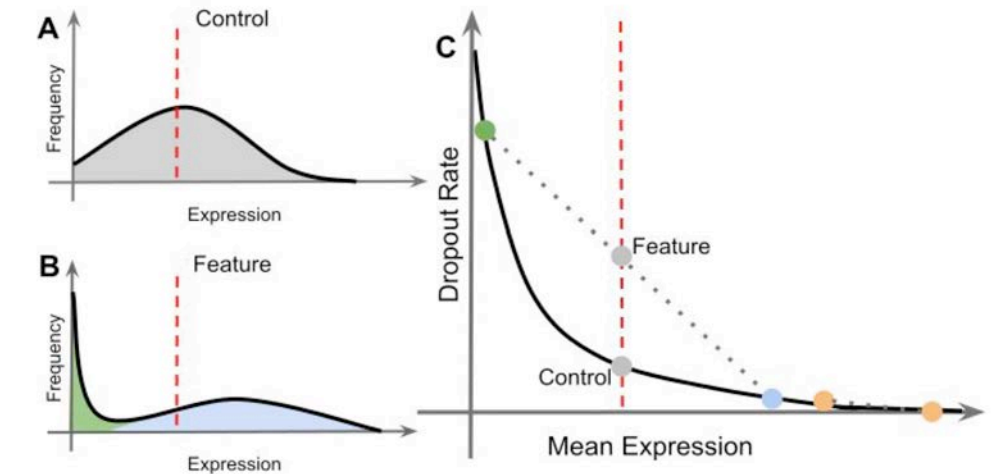
- Reverse transcription is an enzyme reaction thus can be modelled using the Michaelis-Menten equation:

$$P_{dropout} = 1 - \frac{S}{K_M + S}$$

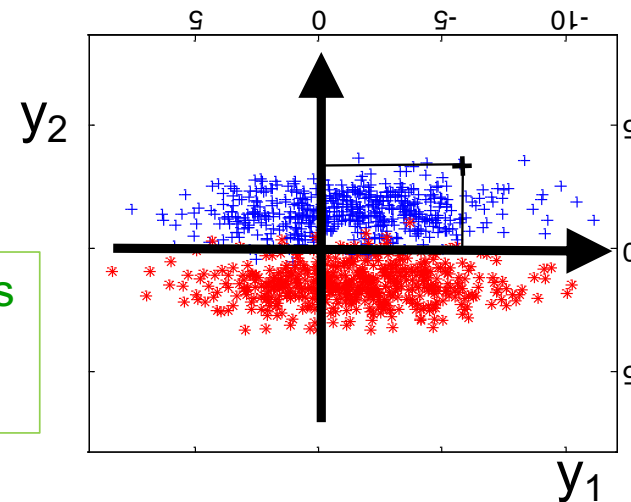
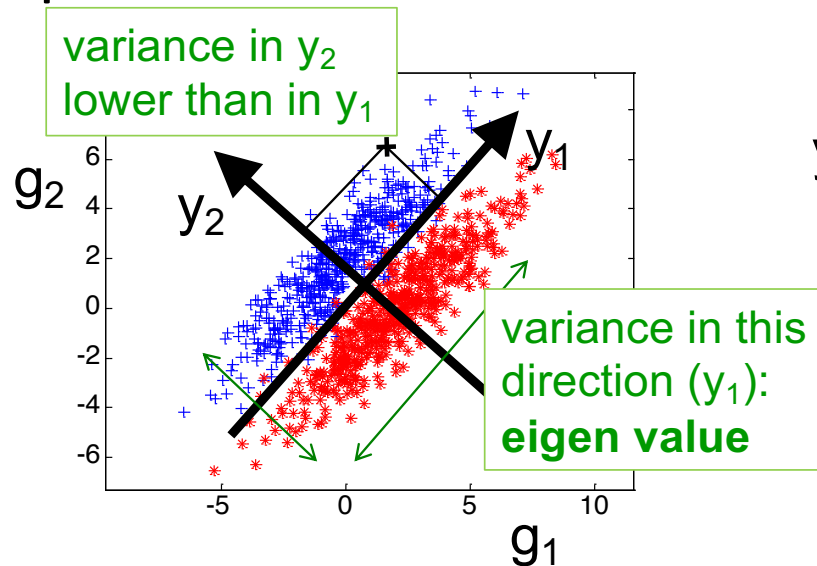
$S$ : average expression

$K_M$ : Michaelis-Menten constant

- Select genes with high dropout AND high expression



# Principle Component Analysis (PCA) rotates the space



- eigen vector
- principal component
- eigen gene

$$y_1 = w_{11}g_1 + w_{12}g_2$$
$$y_2 = w_{21}g_1 + w_{22}g_2$$

loading factor

(importance of gene in contributing to variance)

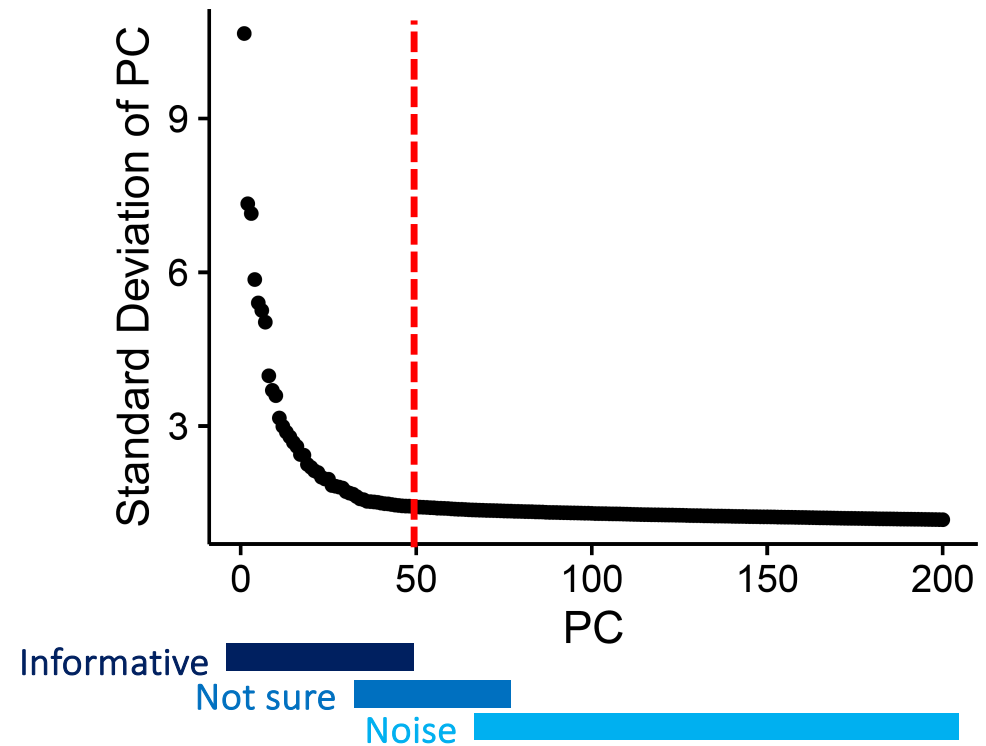
component showing the most variation

# Feature selection

## *Selecting principal components*

- To overcome the extensive technical noise in scRNA-seq data, it is common to cluster cells based on their PCA scores
- Each PC represents a 'metagene' that (linearly) combines information across a correlated gene set

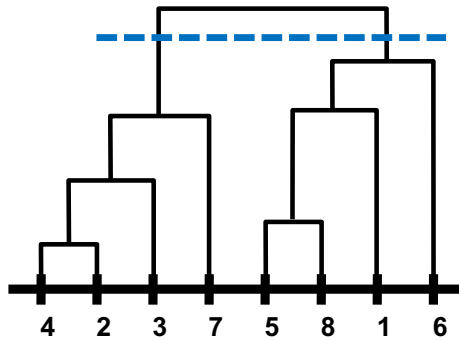
Scree/Elbow plot



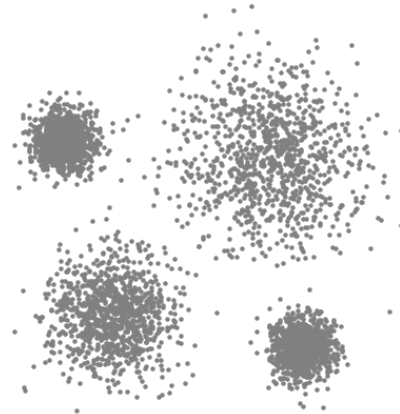
# Outline

- Feature selection
- **Introduction to clustering**
  - Hierarchical clustering
  - $k$ -Means clustering
  - Graph-based clustering
- scRNA-seq clustering
  - Single Cell Consensus Clustering (SC3)
  - Seurat
- Validation

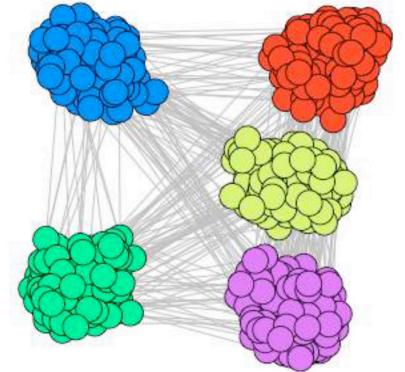
# Many clustering approaches



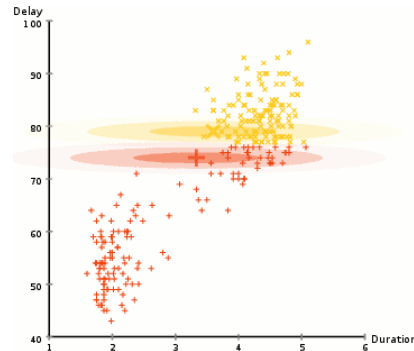
Hierarchical Clustering



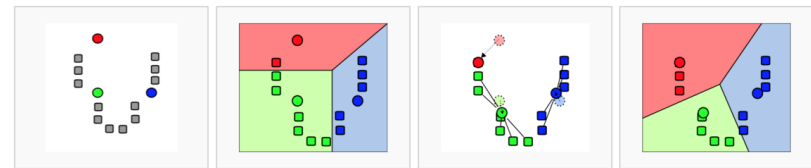
Mean shift clustering



Graph-based clustering

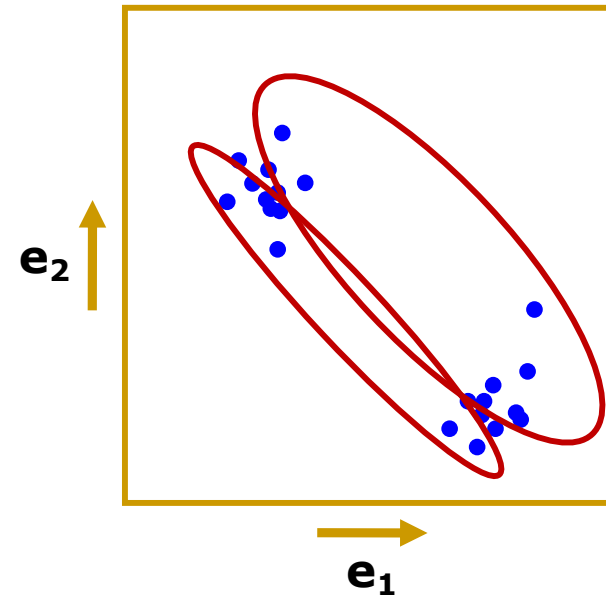
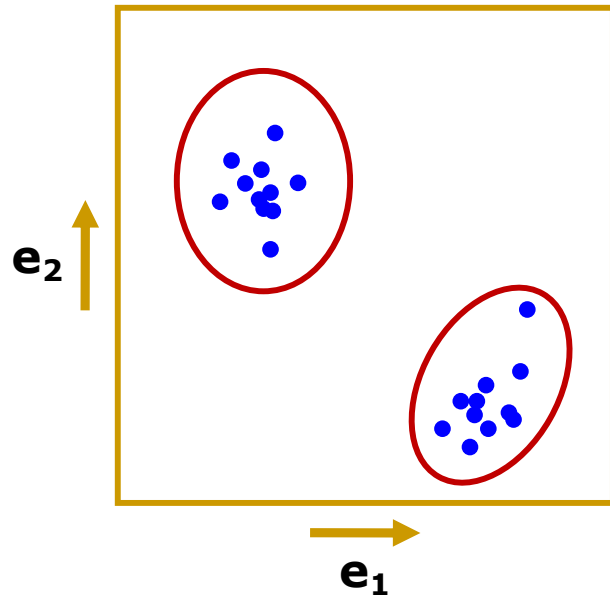


Gaussian mixture modeling



k-means clustering

# Clustering

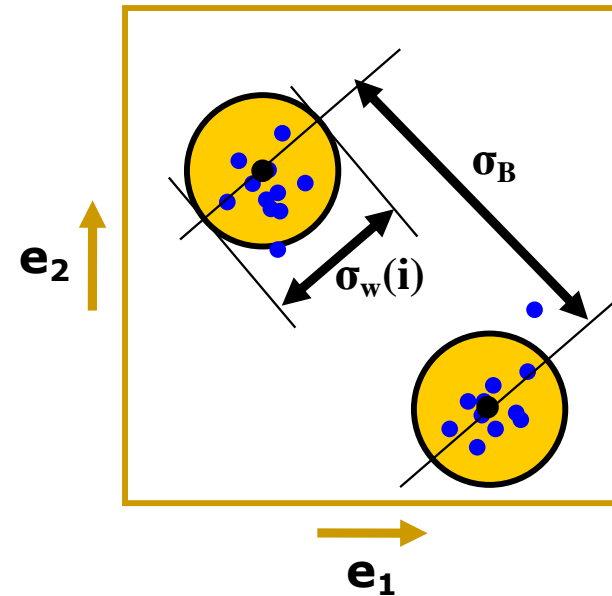


# Clustering

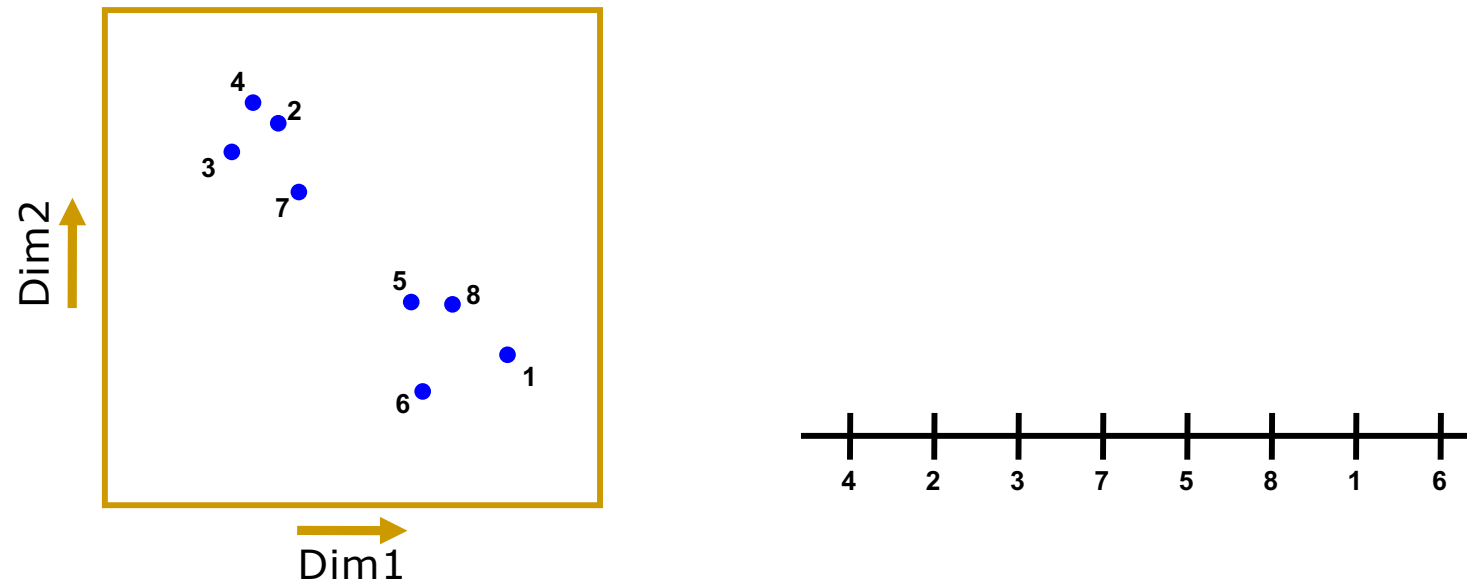
- Structure when:
  - 1) Samples within cluster resemble each other (*within variance*,  $\sigma_w(i)$ )
  - 2) Clusters deviate from each other (*between variance*,  $\sigma_B$ )

- Group samples such that:

$$\min \left( \frac{\sum \sigma_w(i)}{\sigma_B} \right) \rightarrow \begin{array}{l} \sigma_w: \text{small \&} \\ \sigma_B: \text{large} \end{array}$$



# Hierarchical clustering

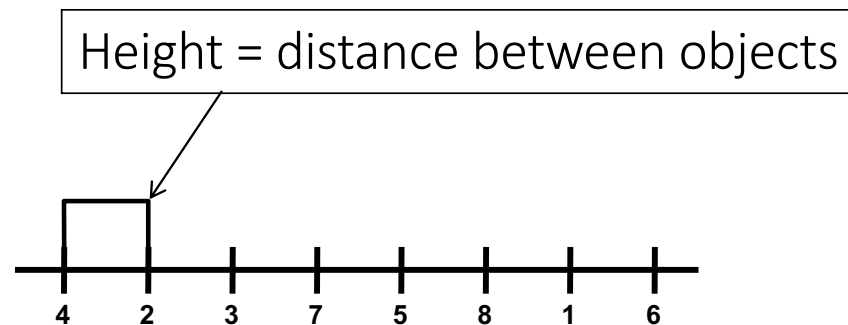
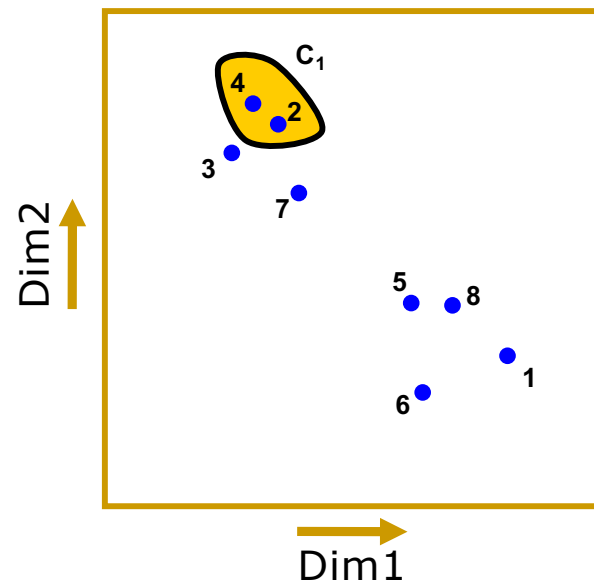


Find most similar objects (genes) and group them



# Hierarchical clustering

dendrogram

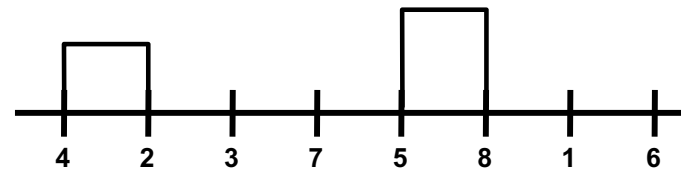
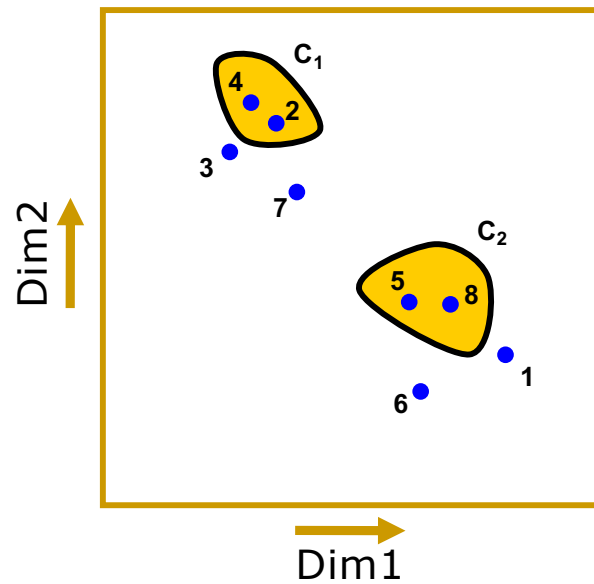


These are: objects 4 and 2

Again, find most similar objects (genes or clusters) and group them

# Hierarchical clustering

dendrogram

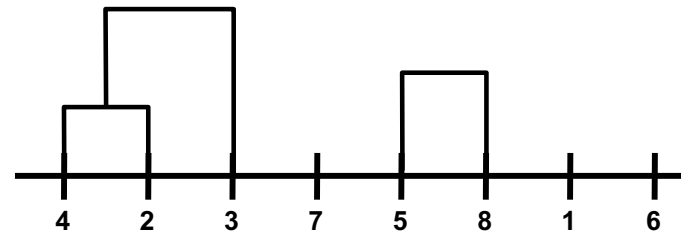
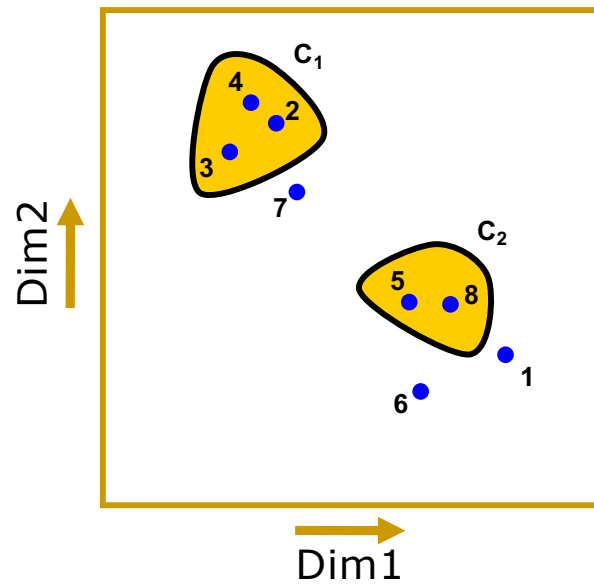


These are: objects 5 and 8

Repeat finding most similar objects (genes or clusters) and grouping them

# Hierarchical clustering

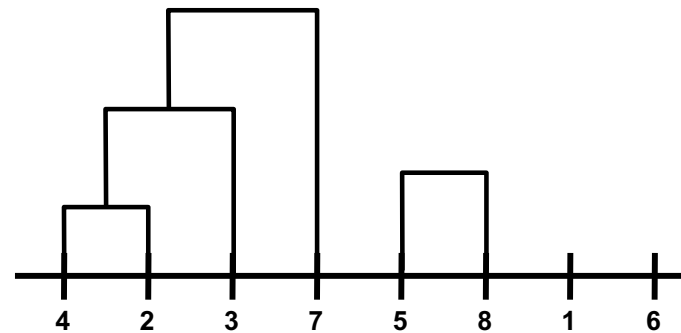
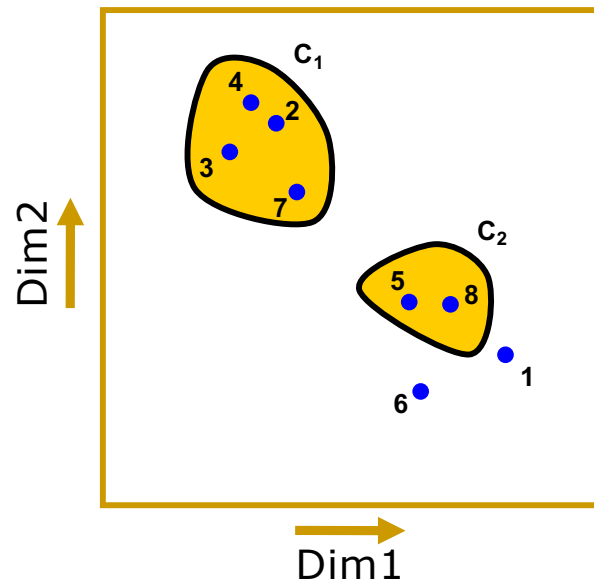
dendrogram



Join object 3 and cluster 1  
Repeat process

# Hierarchical clustering

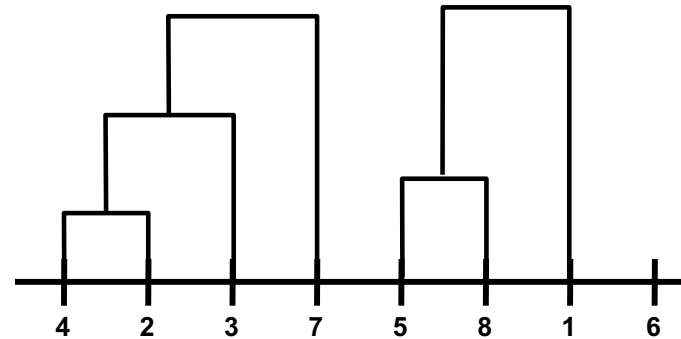
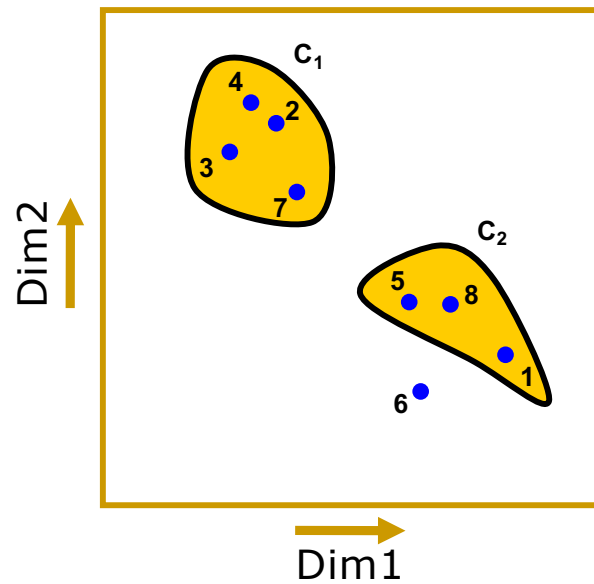
dendrogram



Join [object 7 and cluster 1] -> [cluster 1]  
Repeat process

# Hierarchical clustering

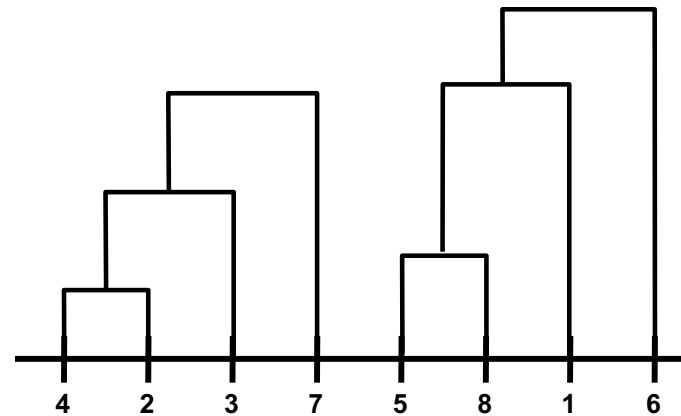
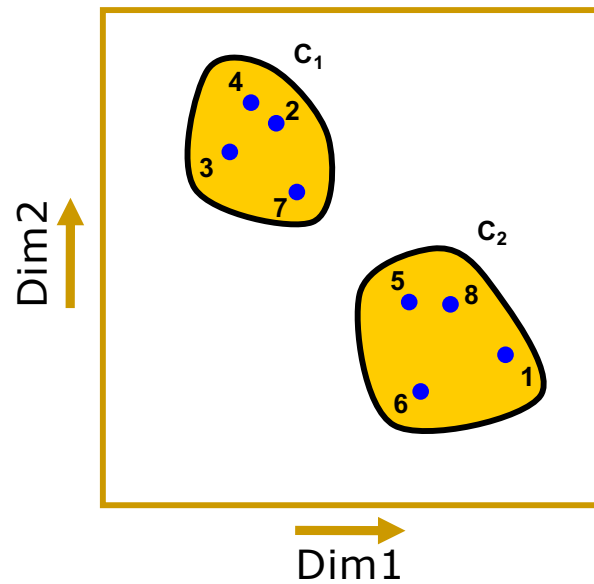
dendrogram



Join [object 1 and cluster 2] -> [cluster 2]  
Repeat process

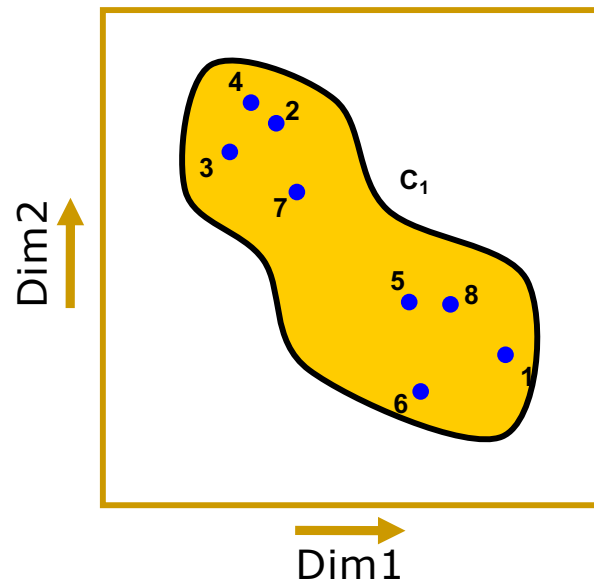
# Hierarchical clustering

dendrogram



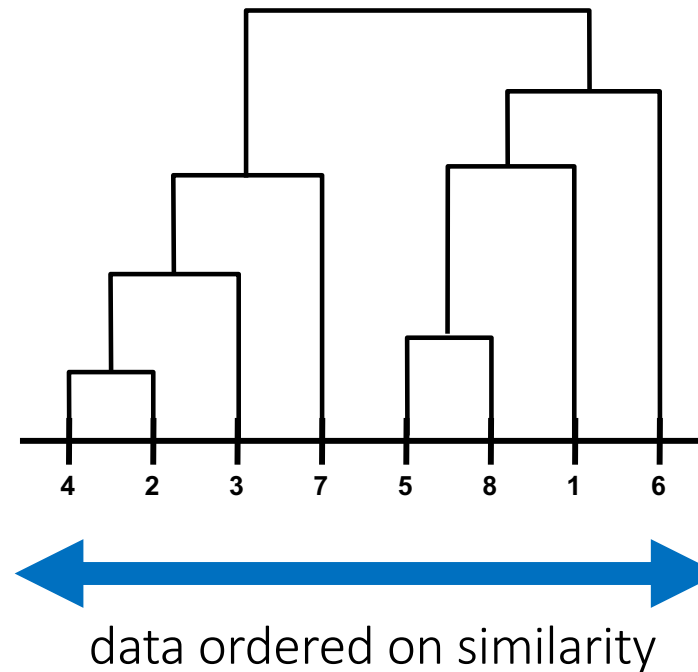
Join [object 6 and cluster 2] -> [cluster 2]  
Repeat process

# Hierarchical clustering

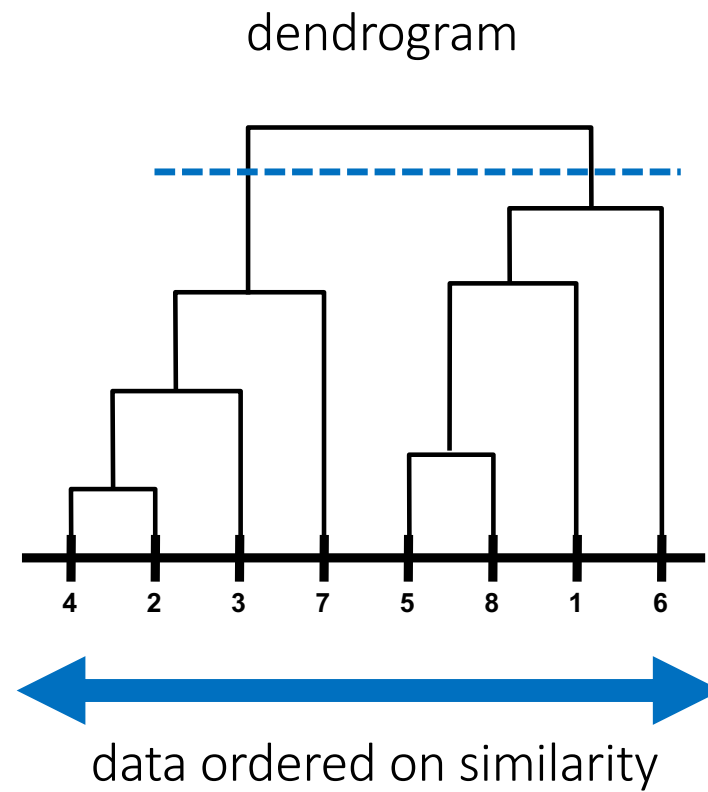
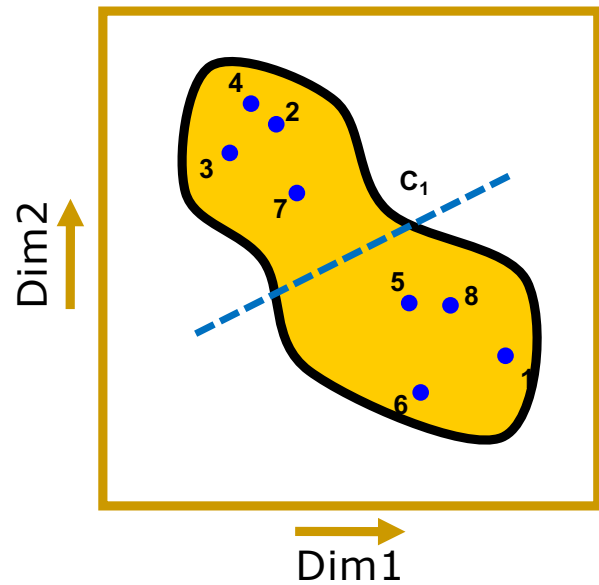


Join [cluster 1 and cluster 2] -> [cluster 1]  
All in one cluster: FINISHED!

dendrogram



# Hierarchical clustering

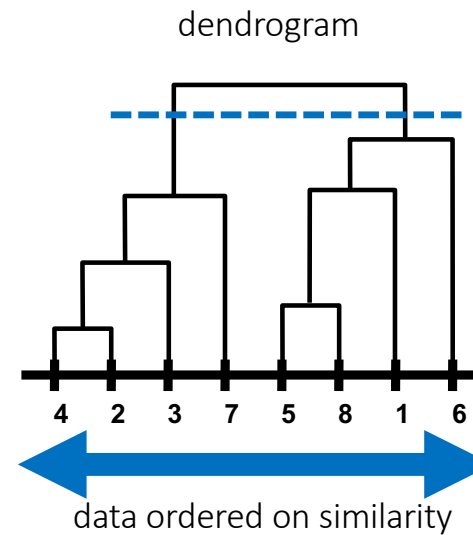
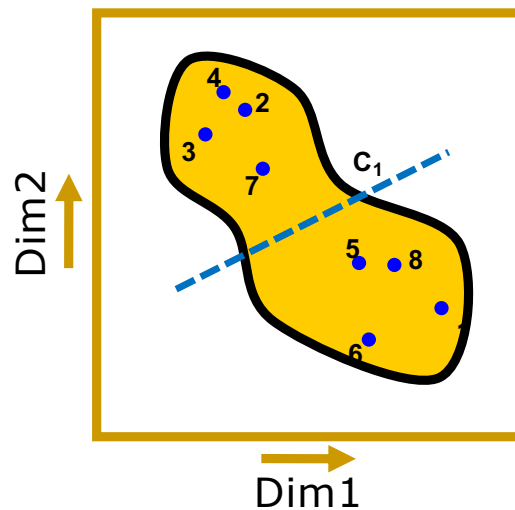




# Hierarchical clustering

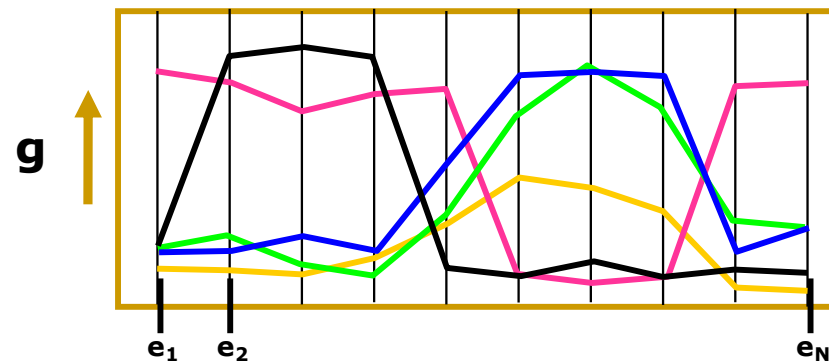
Need to know:

- Similarity between objects
- Similarity between clusters



# Hierarchical clustering

Similarity between objects



## Euclidean distance

$$d(g_i, g_j) = \sqrt{\sum ((x_i - x_j)^2)}$$

$d(\bullet, \text{green}) < d(\bullet, \text{yellow})$   
 $d(\bullet, \text{green}) << d(\bullet, \text{pink})$   
 $d(\bullet, \text{green}) << d(\bullet, \text{black})$

Match exact shape

## Pearson correlation

$$1 - \rho_{ij}$$

$d(\bullet, \text{green}) \approx d(\bullet, \text{yellow})$   
 $d(\bullet, \text{green}) << d(\bullet, \text{pink})$   
 $d(\bullet, \text{green}) << d(\bullet, \text{black})$

Ignore amplitude

## Mixed Pearson correlation

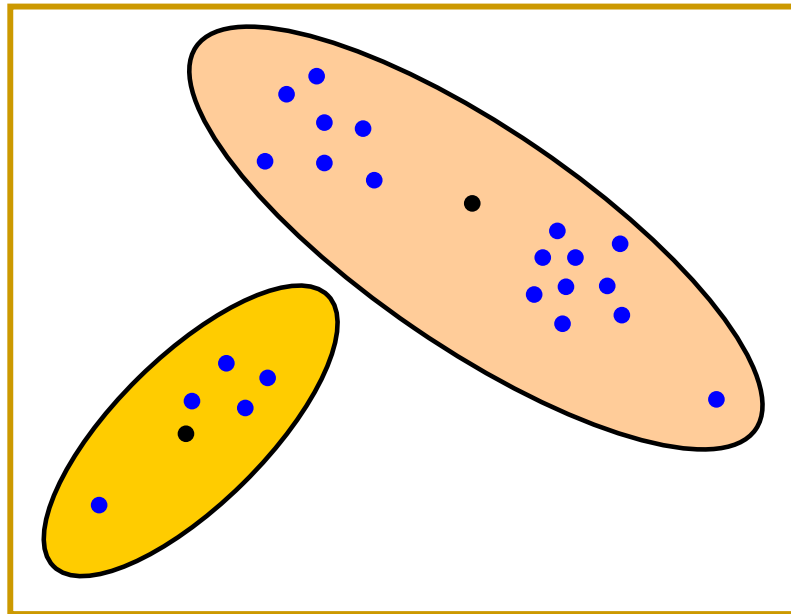
$$1 - |\rho_{ij}|$$

$d(\bullet, \text{green}) \approx d(\bullet, \text{yellow})$   
 $d(\bullet, \text{green}) \approx d(\bullet, \text{pink})$   
 $d(\bullet, \text{green}) << d(\bullet, \text{black})$

Ignore amplitude and sign

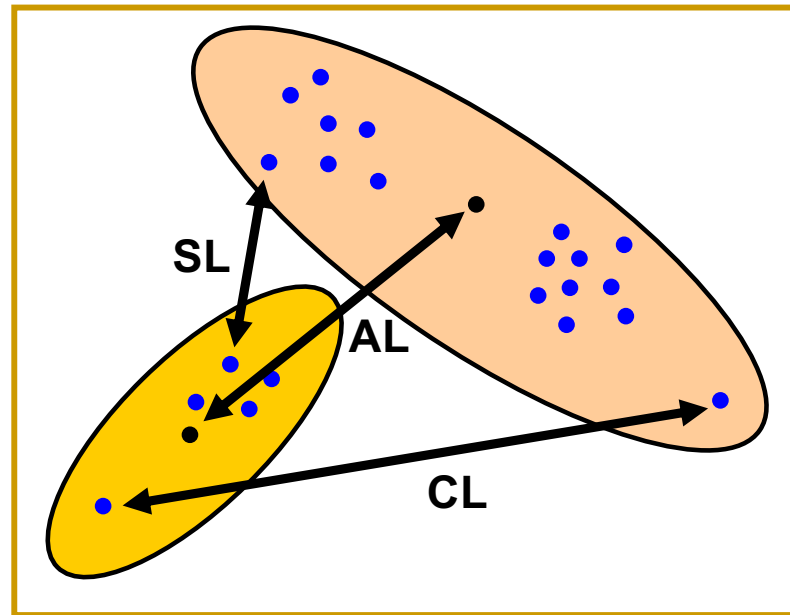
# Hierarchical clustering

*Similarity between clusters*



# Hierarchical clustering

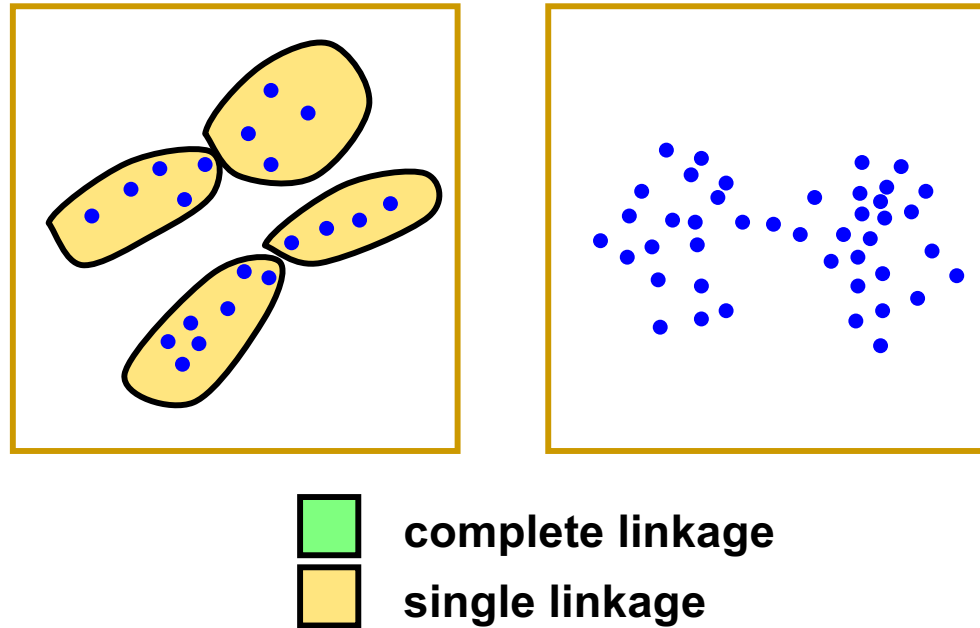
Similarity between clusters



- Single linkage: Closest objects
- Complete linkage: Furthest objects
- Average linkage: Average dissimilarity

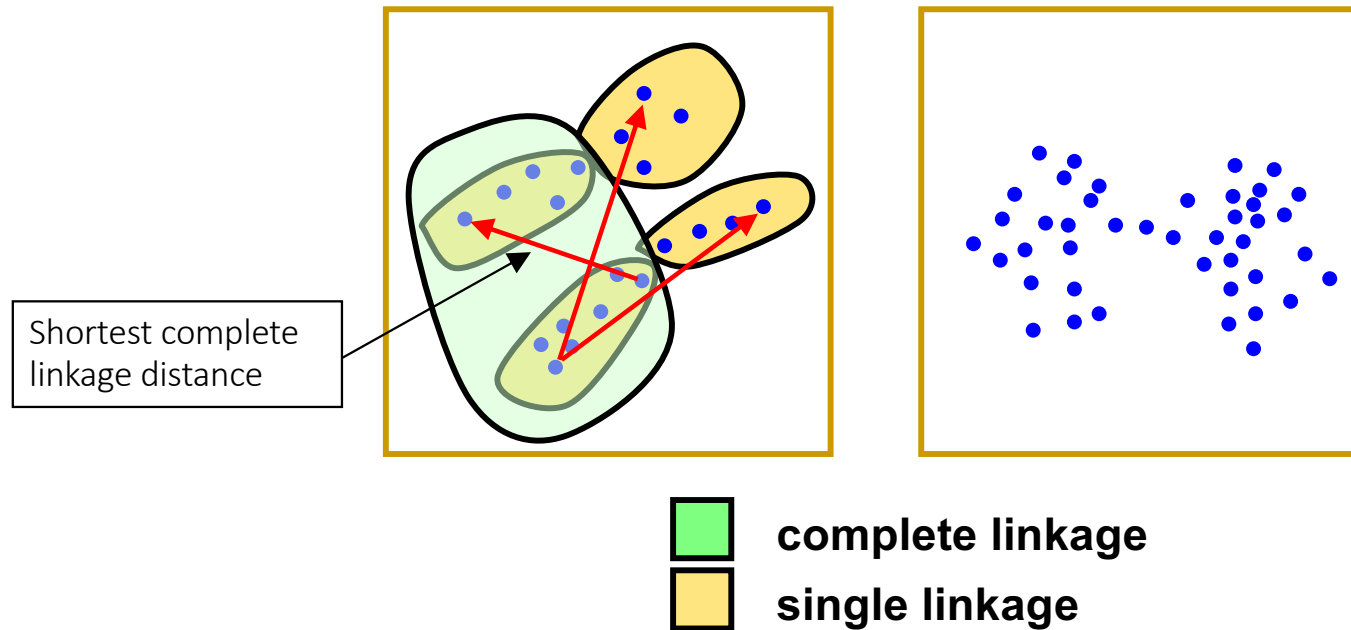
# Hierarchical clustering

Similarity between clusters



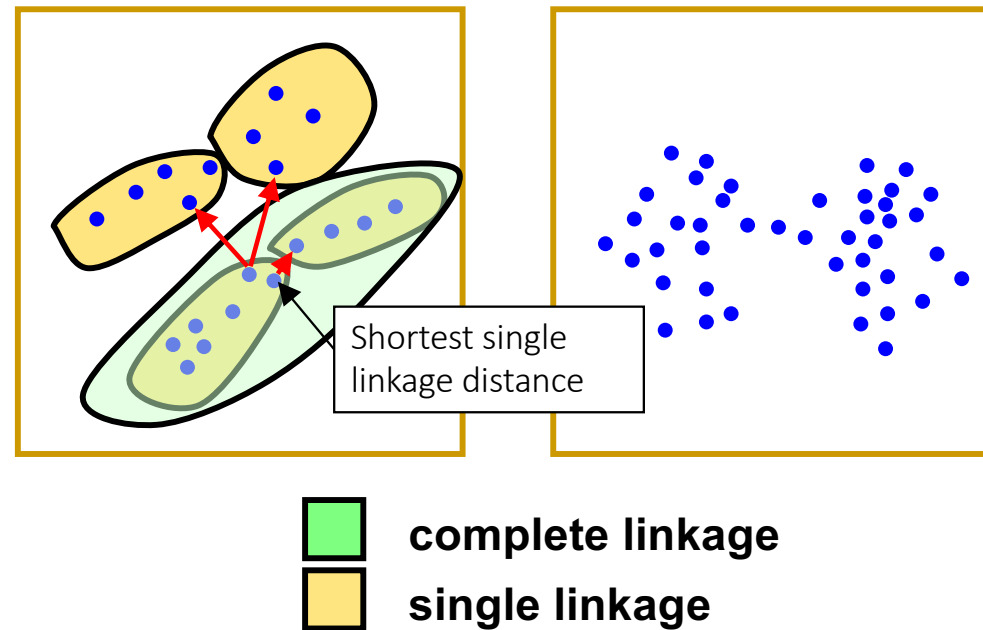
# Hierarchical clustering

Similarity between clusters



# Hierarchical clustering

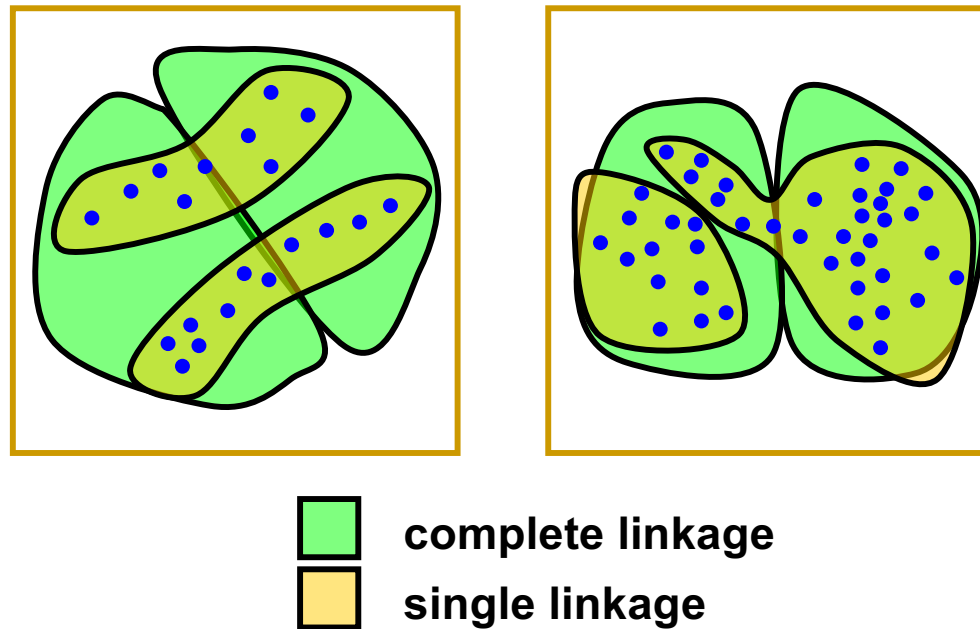
Similarity between clusters



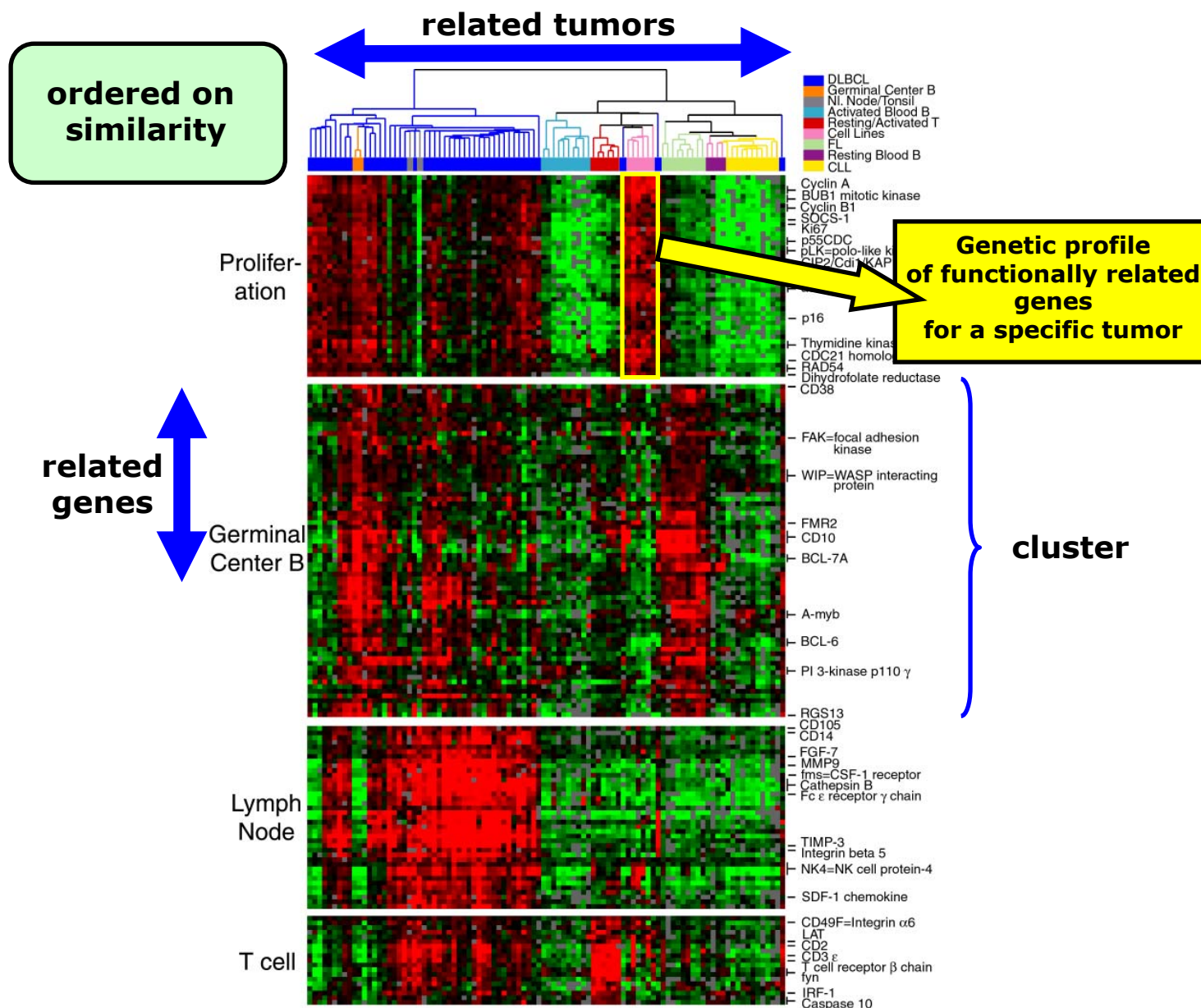
# Hierarchical clustering

Similarity between clusters

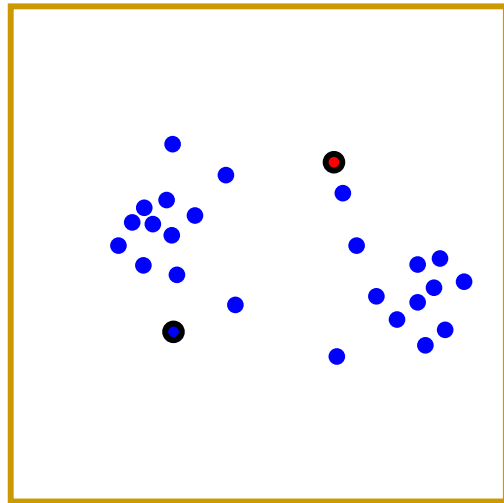
- Single linkage -> long and “loose” clusters
- Complete linkage -> compact clusters





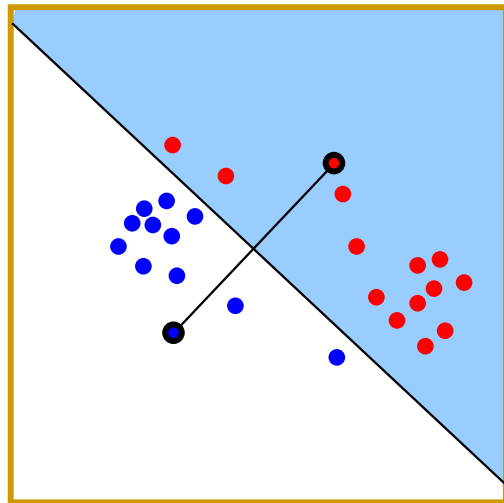


# $k$ -Means clustering



Choose randomly 2 prototypes

# $k$ -Means clustering

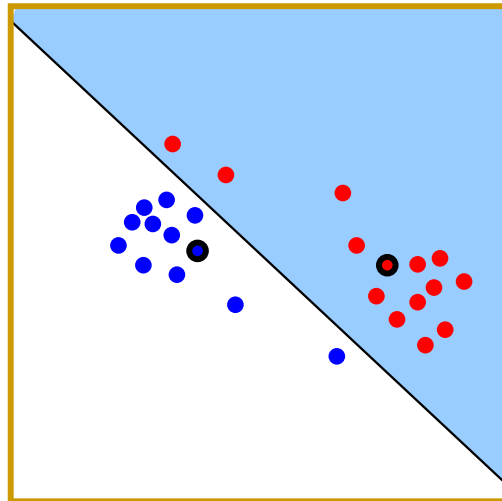


Assign objects to closest prototype

Blue area: cluster 1

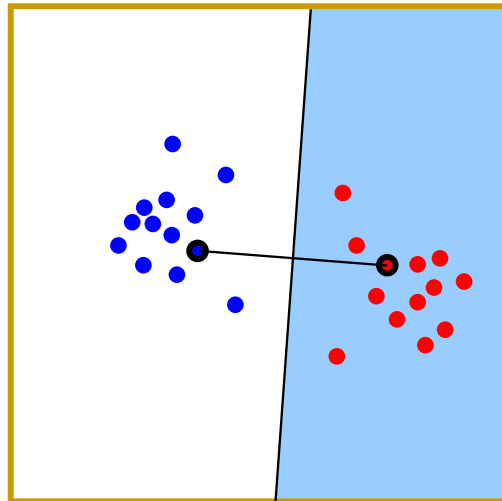
White area: cluster 2

# $k$ -Means clustering



Calculate new cluster prototypes  
By averaging objects

# $k$ -Means clustering

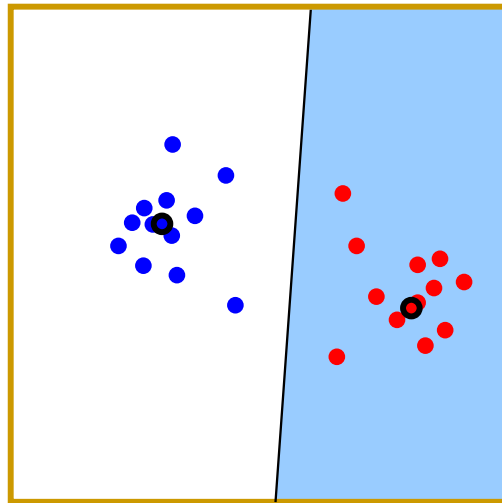


Re-assign objects to closest prototype

Blue area: cluster 1

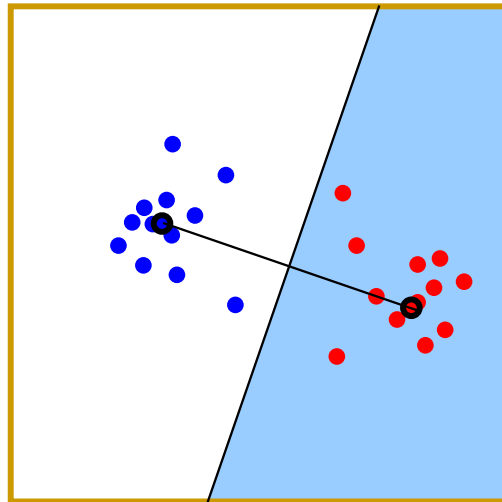
White area: cluster 2

# $k$ -Means clustering



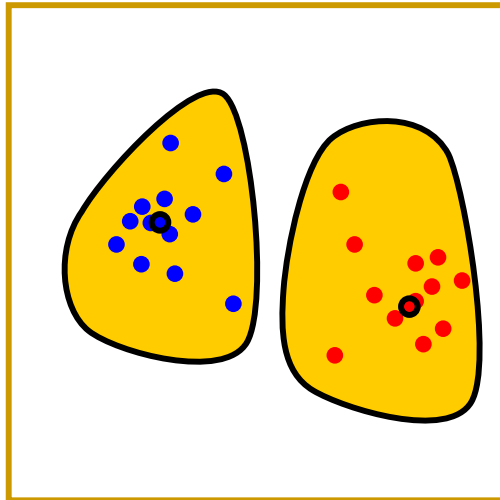
Re-calculate new cluster prototypes

# $k$ -Means clustering



Re-assign objects to closest prototype  
If no objects change cluster then finished

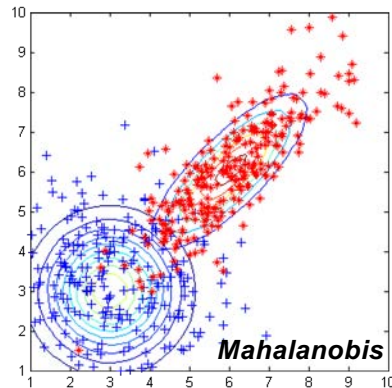
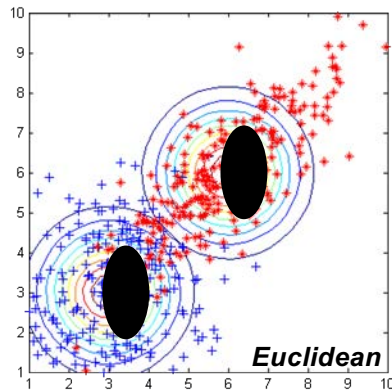
# $k$ -Means clustering



Establish clusters



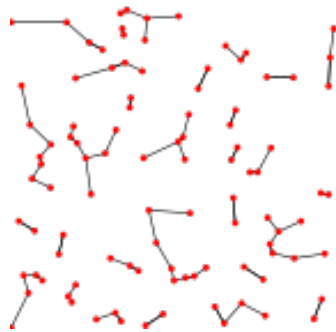
# K-means clustering: Parameters



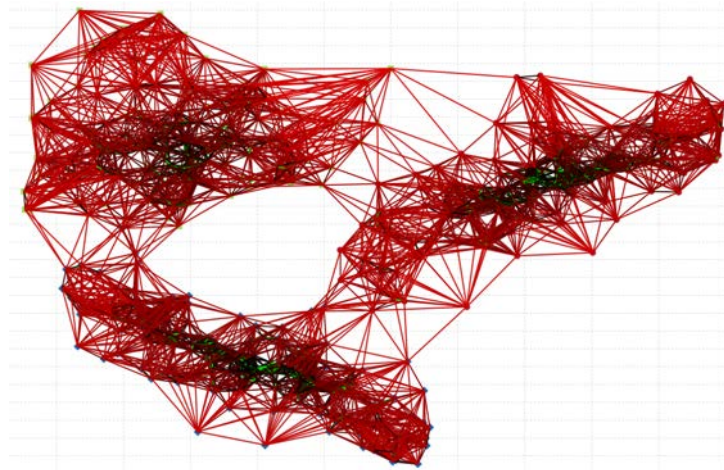
- K-means
  - Fixed number of clusters (need to know a priori)
  - Choice of distance measure
  - Prototype choice
- Distance measure
  - Euclidean: Round clusters
  - Mahalanobis: Elongated clusters
- Prototype choice
  - Point
  - Line etc.
- Number of clusters
  - Validate clustering!

# Graph-based Clustering

- K-NN graph: Connect every node to its  $k$ -nearest nodes
- Find densely connected components

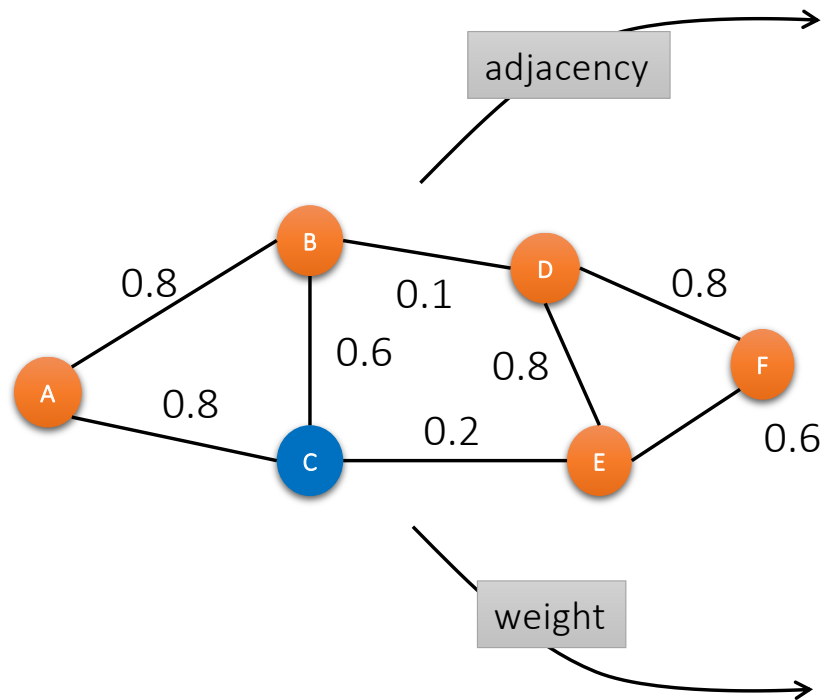


$k=1$



$k=20$

# Graphs, adjacency and weight matrices



$$A = \begin{matrix} & \begin{matrix} A & B & C & D & E & F \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \\ F \end{matrix} & \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix} \end{matrix}$$

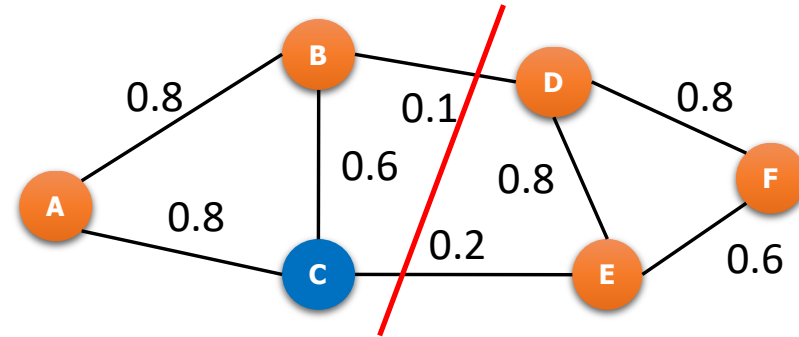
$$W = \begin{matrix} & \begin{matrix} A & B & C & D & E & F \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \\ F \end{matrix} & \begin{pmatrix} 0 & 0.8 & 0.8 & 0 & 0 & 0 \\ 0.8 & 0 & 0.6 & 0.1 & 0 & 0 \\ 0.8 & 0.6 & 0 & 0 & 0.2 & 0 \\ 0 & 0.1 & 0 & 0 & 0.8 & 0.8 \\ 0 & 0 & 0.2 & 0.8 & 0 & 0.6 \\ 0 & 0 & 0 & 0.8 & 0.6 & 0 \end{pmatrix} \end{matrix}$$

# Spectral clustering (1)

- Minimise normalised cut
- Normalised cut between two clusters  $C_1$  and  $C_2$ :

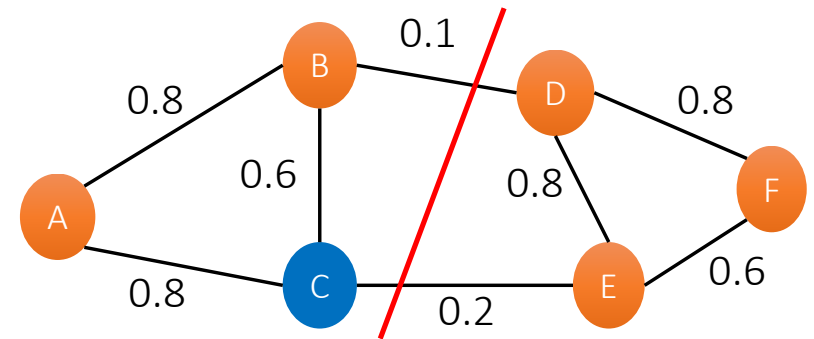
$$NC(C_1, C_2) = \frac{\text{cut}(C_1, C_2)}{\text{assoc}(C_1, V)} + \frac{\text{cut}(C_2, C_1)}{\text{assoc}(C_2, V)} = 2 - \left( \frac{\text{assoc}(C_1, C_1)}{\text{assoc}(C_1, V)} + \frac{\text{assoc}(C_2, C_2)}{\text{assoc}(C_2, V)} \right)$$

- $\text{cut}(C_1, C_2)$  = weight of links between  $C_1$  and  $C_2$
- $\text{cut}(C_2, C_1)$  = same
- $\text{assoc}(C_1, V)$  = total weight of links from nodes in  $C_1$  to entire graph
- $\text{assoc}(C_2, V)$  = total weight of links from nodes in  $C_2$  to entire graph

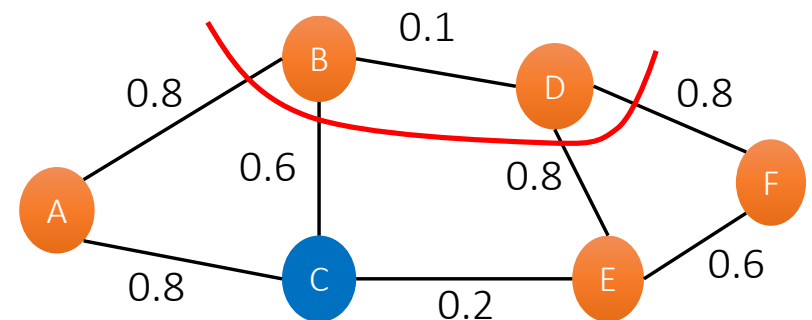


# Normalized cut (example)

- $\text{cut}(S,T) = 0.1 + 0.2 = 0.3$
- $\text{vol}(S) = 0.3 + 0.6 + 0.8 + 0.8 = 2.5$
- $\text{vol}(T) = 0.3 + 0.8 + 0.8 + 0.6 = 2.5$
- $\text{Ncut}(S,T) = 0.3/2.5 + 0.3/2.5 = \mathbf{0.24}$



- $\text{cut}(S,T) = 0.8 + 0.6 + 0.8 + 0.8 = 3.0$
- $\text{vol}(S) = 3.0 + 0.1 = 3.1$
- $\text{vol}(T) = 3.0 + 0.8 + 0.2 + 0.6 = 4.6$
- $\text{Ncut}(S,T) = 3.0/3.1 + 3.0/4.6 = 1.62$



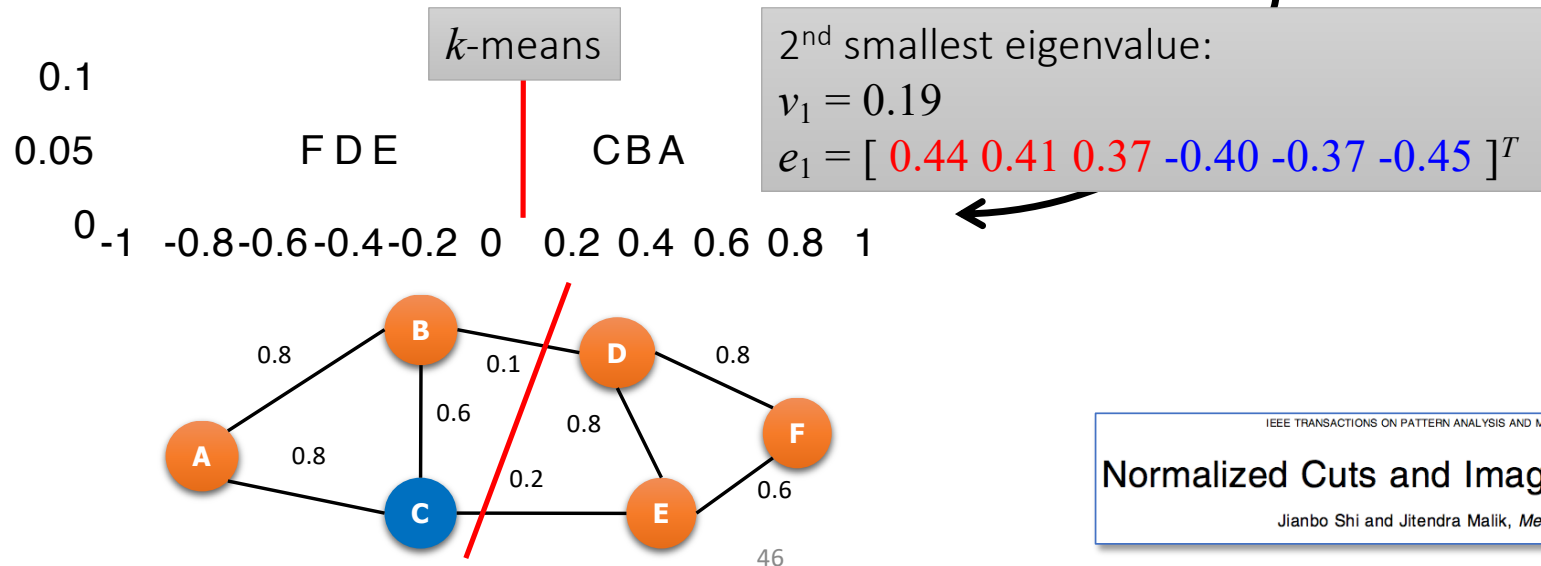
# Spectral clustering (2)

(sum weights on diagonal – W)

$$W = \begin{matrix} & \begin{matrix} A & B & C & D & E & F \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \\ F \end{matrix} & \begin{pmatrix} 0 & 0.8 & 0.8 & 0 & 0 & 0 \\ 0.8 & 0 & 0.6 & 0.1 & 0 & 0 \\ 0.8 & 0.6 & 0 & 0 & 0.2 & 0 \\ 0 & 0.1 & 0 & 0 & 0.8 & 0.8 \\ 0 & 0 & 0.2 & 0.8 & 0 & 0.6 \\ 0 & 0 & 0 & 0.8 & 0.6 & 0 \end{pmatrix} \end{matrix}$$

Laplacian  
 $L = \text{diag}(1^T W) - W$

$$L = \begin{matrix} & \begin{matrix} A & B & C & D & E & F \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \\ F \end{matrix} & \begin{pmatrix} 1.6 & -0.8 & -0.8 & 0 & 0 & 0 \\ -0.8 & 1.5 & -0.6 & -0.1 & 0 & 0 \\ -0.8 & -0.6 & 1.6 & 0 & -0.2 & 0 \\ 0 & -0.1 & 0 & 1.7 & -0.8 & -0.8 \\ 0 & 0 & -0.2 & -0.8 & 1.6 & -0.6 \\ 0 & 0 & 0 & -0.8 & -0.6 & 1.4 \end{pmatrix} \end{matrix}$$



# Markov clustering (1)

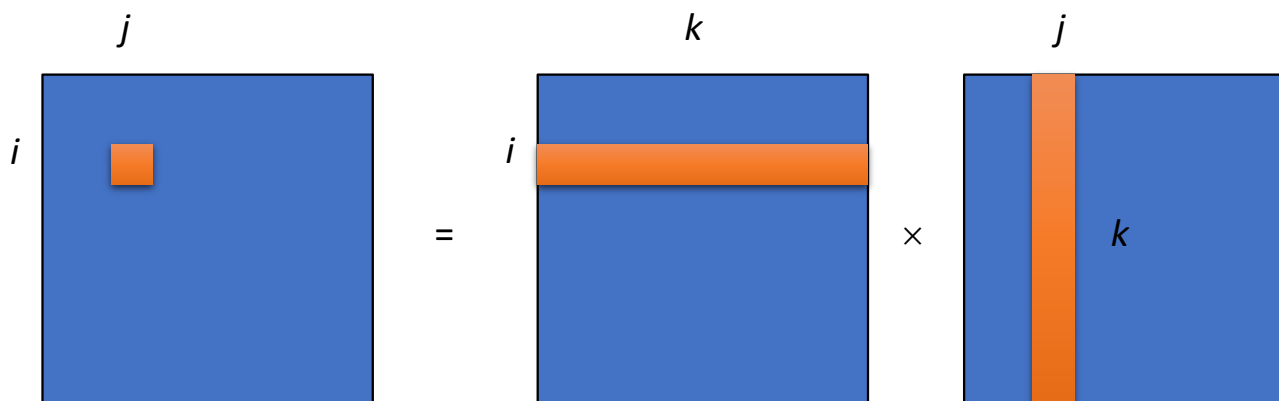
- Markov clustering
  - Random walks are Markov chains
  - Start with Markov matrix:  $M_{ij} = p(j \rightarrow i)$
  - (optionally add loops (diagonal))

$$\begin{array}{c}
 \begin{array}{c} W = A \\ B \\ C \\ D \\ E \\ F \end{array}
 \begin{array}{c} A \quad B \quad C \quad D \quad E \quad F \\ \left( \begin{array}{cccccc} 0 & 0.8 & 0.8 & 0 & 0 & 0 \\ 0.8 & 0 & 0.6 & 0.1 & 0 & 0 \\ 0.8 & 0.6 & 0 & 0 & 0.2 & 0 \\ 0 & 0.1 & 0 & 0 & 0.8 & 0.8 \\ 0 & 0 & 0.2 & 0.8 & 0 & 0.6 \\ 0 & 0 & 0 & 0.8 & 0.6 & 0 \end{array} \right) \end{array}
 \end{array}
 \xrightarrow[\mathbf{M} = \mathbf{W} \text{diag}(\mathbf{1}^T \mathbf{W})^{-1}]{\text{Markov}}
 \begin{array}{c}
 \begin{array}{c} M = A \\ B \\ C \\ D \\ E \\ F \end{array}
 \begin{array}{c} A \quad B \quad C \quad D \quad E \quad F \\ \left( \begin{array}{cccccc} 0 & 0.53 & 0.5 & 0 & 0 & 0 \\ 0.5 & 0 & 0.38 & 0.06 & 0 & 0 \\ 0.5 & 0.4 & 0 & 0 & 0.12 & 0 \\ 0 & 0.07 & 0 & 0 & 0.5 & 0.57 \\ 0 & 0 & 0.12 & 0.47 & 0 & 0.43 \\ 0 & 0 & 0 & 0.47 & 0.38 & 0 \end{array} \right) \end{array}
 \end{array}$$

(sum columns = 1)

## Markov clustering (2)

$M_{ij} = p(j \rightarrow i)$  probability of arriving in  $i$  from  $j$  in 1 step



$(M^2)_{ij} = \sum_k p(j \rightarrow k)p(k \rightarrow i)$  probability of arriving in  $i$  from  $j$  in 2 steps (etc.)



# Markov clustering (3)

- Markov clustering:

- Iterate

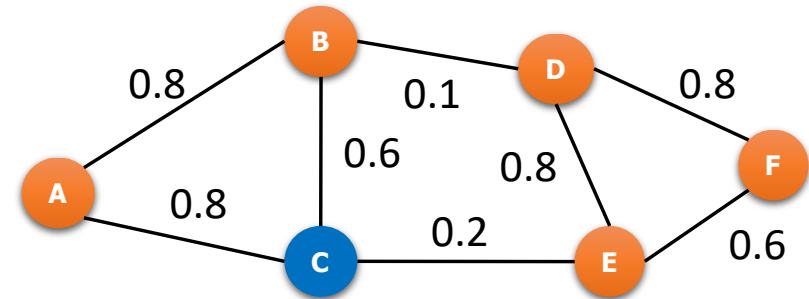
1.  $M = M^\alpha$

2.  $M_{ij} = M_{ij}^\beta / \sum_i M_{ij}^\beta$

- Step 1 : take walk of  $\alpha$  steps

- Step 2 : increase difference between small probabilities ( $<0.5$ ) and large probabilities ( $>0.5$ ) ; converges to maximum value in column

- E.g. for  $\alpha = \beta = 2$



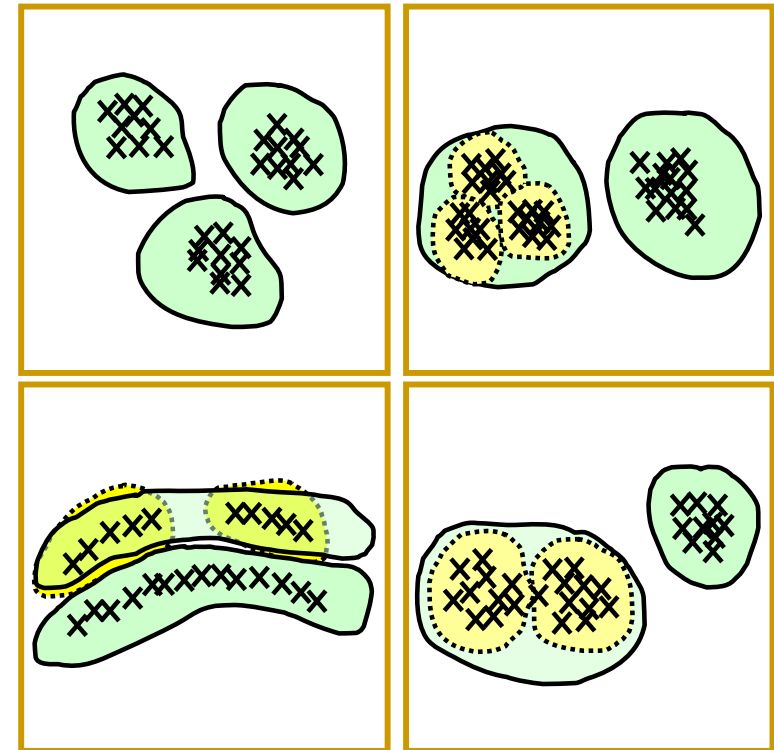
$$M = \begin{matrix} & \begin{matrix} A & B & C & D & E & F \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \\ F \end{matrix} & \begin{pmatrix} \boxed{0.00} & \boxed{0.00} & \boxed{0.00} & 0.00 & 0.00 & 0.00 \\ 0.00 & \boxed{0.00} & \boxed{0.00} & 0.00 & 0.00 & 0.00 \\ 0.00 & \boxed{0.00} & \boxed{0.00} & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & \boxed{0.00} & \boxed{0.00} & \boxed{0.00} \\ 0.00 & 0.00 & 0.00 & 0.00 & \boxed{0.00} & \boxed{0.00} \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & \boxed{0.00} \end{pmatrix} \end{matrix}$$

# Outline

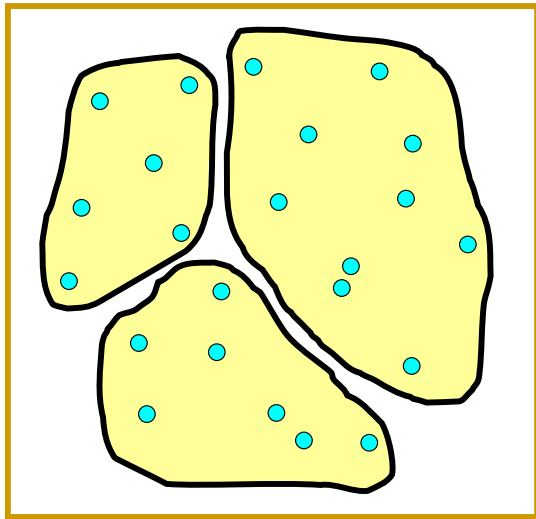
- Feature selection
- Introduction to clustering
  - Hierarchical clustering
  - $k$ -Means clustering
  - Graph-based clustering
- **Validation**
- scRNA-seq clustering
  - Single Cell Consensus Clustering (SC3)
  - Seurat

# Clustering is subjective!

- Principle choices
  - Similarity measure
  - Algorithm
- Different choice leads to different results
  - Subjectivity becomes reality
- Cluster process
  - Validate, interpret (generate hypothesis), repeat steps



# Cluster Validation



- Cluster tendency

Clustering **IMPOSES** structure even though data may not possess it

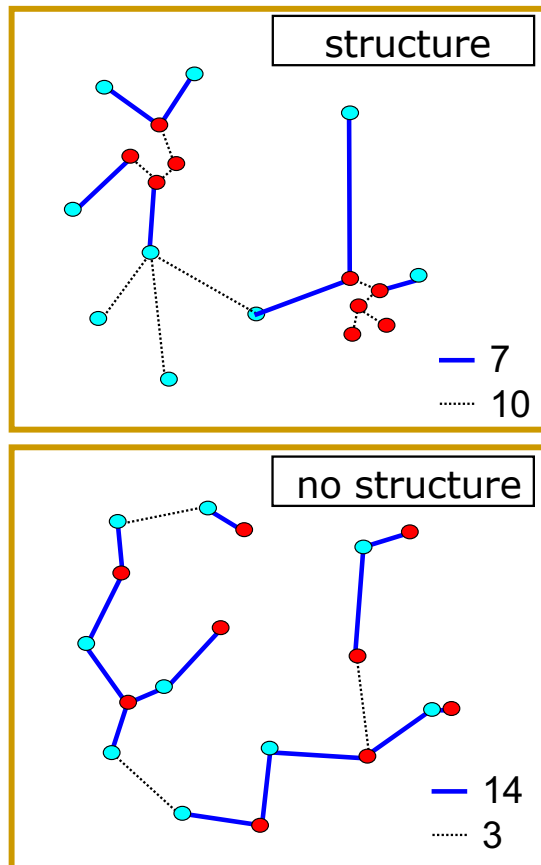
Aim: Test whether data possesses structure

- Cluster validity

Choices impose restrictions on for example shape

Aim: Quantitative evaluation of the clustering results

# Test for spatial randomness



- Test

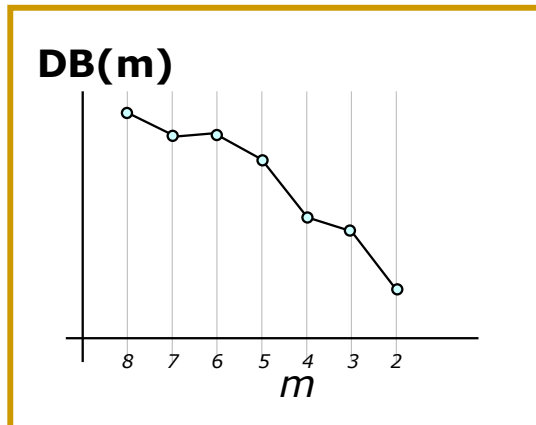
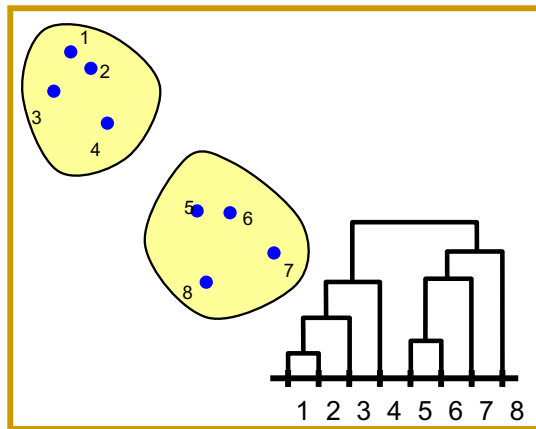
If data (●) clusters frequently with random data (●) then data structureless

- Approach

- Generate random vectors ( $\mathbf{Y}$ ) uniformly over observed region of data ( $\mathbf{X}$ )
- Find MST (single linkage HC) of  $\mathbf{X} \cup \mathbf{Y}$
- Determine number of edges  $q$  that connect vectors of  $\mathbf{X}$  with  $\mathbf{Y}$
- If  $\mathbf{X}$  contains clusters  $q$  should be small!

(multiple random vs random measurements gives likelihood for  $q$ )

# Davis-Bouldin index



- Test

Select specific clustering according to a criteria  
For example: Davis-Bouldin index

- DB index

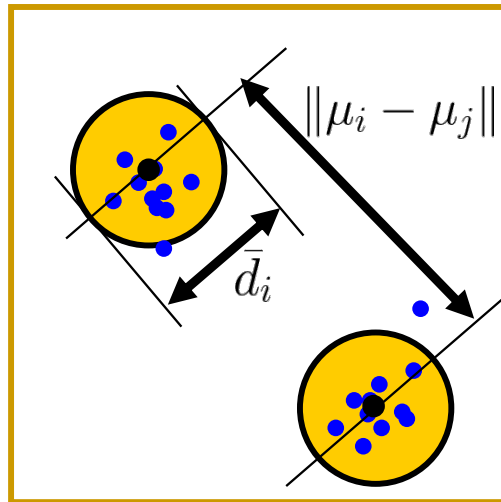
For a specific clustering  $m$ ,  $DB(m)$  : Average similarity of a cluster with its most similar cluster

- Approach

*Goal*: Clusters to have minimal similarity

*Seek*: Clustering that minimize  $DB(m)$  wrt  $m$

# Davis-Bouldin index



- Similarity cluster  $C_i$  and  $C_j$

$$D_{i,j} = \frac{(\bar{d}_i + \bar{d}_j)}{\|\mu_i - \mu_j\|}$$

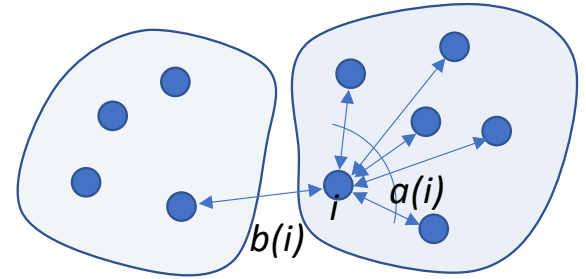
- $\bar{d}_i$ : average distance within cluster  $i$ ,  $\mu_i$ : centroid of cluster  $i$
- Most similar cluster to  $C_i$

$$R_{i,j} = \max_{j \neq i} \{D_{i,j}\}$$

- DB index

$$DB = \frac{1}{k} \sum_{k=1}^k R_{i,j}$$

# Silhouette score



- Measure similarity of object to its own cluster

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$

with  $a(i)$  being average distance to all objects in same cluster and  $b(i)$  being closest object from all other clusters:

$$a(i) = \frac{1}{|C_i|} \sum_{\forall j} d(x_i, x_j) \quad b(i) = \min_{\forall j, j \notin C_i} d(x_i, x_j)$$

$-1 \leq s(i) \leq 1$  ;  $s(i)$  is close to 1, if  $a(i) \ll b(i)$  ; average distance within cluster much smaller than nearest objects

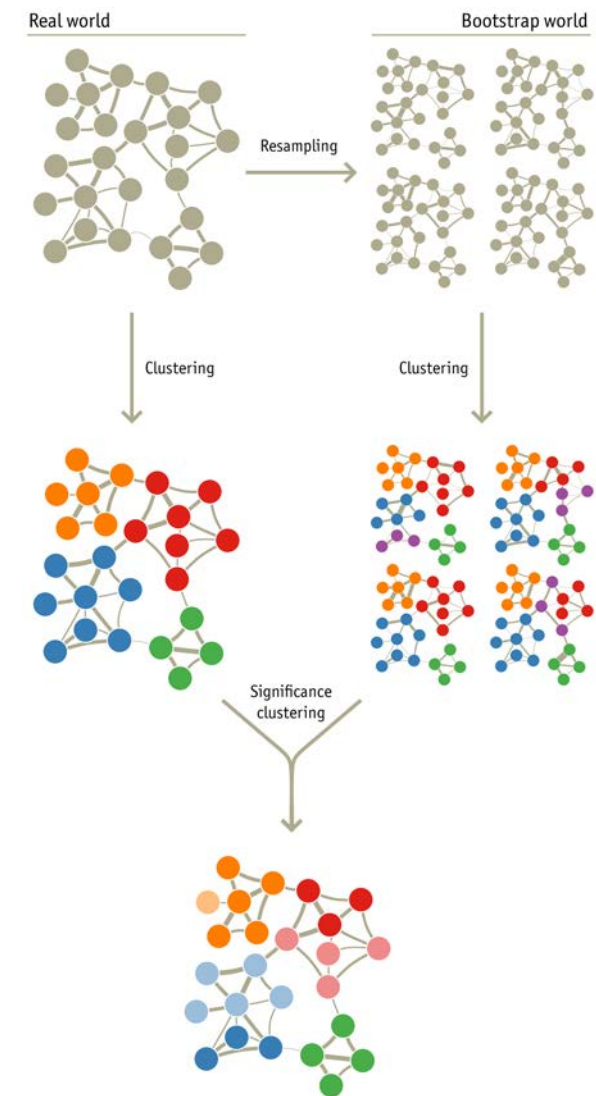
- Silhouette score is average of all these similarities

$$S = \frac{1}{N} \sum S(i)$$



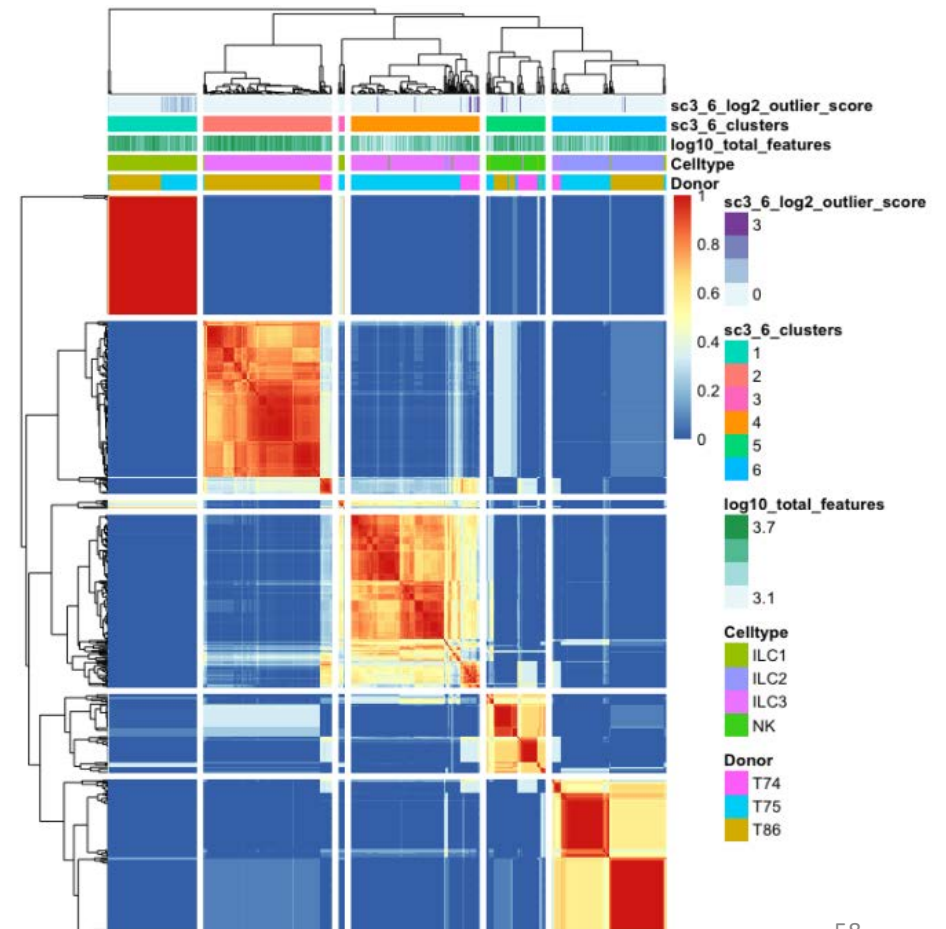
# Bootstrapping

- How confident can you be that the clusters you see are real?
- You can always take a random set of cells from the same cell type and manage to split them into clusters.



# Always check QC data

- Is what your splitting mainly related to batches, qc-measures (especially detected genes)?



# From clusters to cell identities

- Using lists of DE genes and prior knowledge of the biology
- Using lists of DE genes and comparing to other scRNAseq data or sorted cell populations

# Databases with celltype gene signatures

- PanglaoDB (<https://panglaodb.se/>)
  - Human: 295 samples, 72 tissues, 1.1 M cells
  - Mouse: 976 samples, 173 tissues, 4 M cells
  - Franzén et al (<https://doi.org/10.1093/database/baz046>)
- CellMarker (<http://biocc.hrbmu.edu.cn/CellMarker/>)
  - Human: 13,605 cell markers of 467 cell types in 158 tissues
  - Mouse: 9,148 cell makers of 389 cell types in 81 tissues
  - Zhang et al. (<https://doi.org/10.1093/nar/gky900>)

# Outline

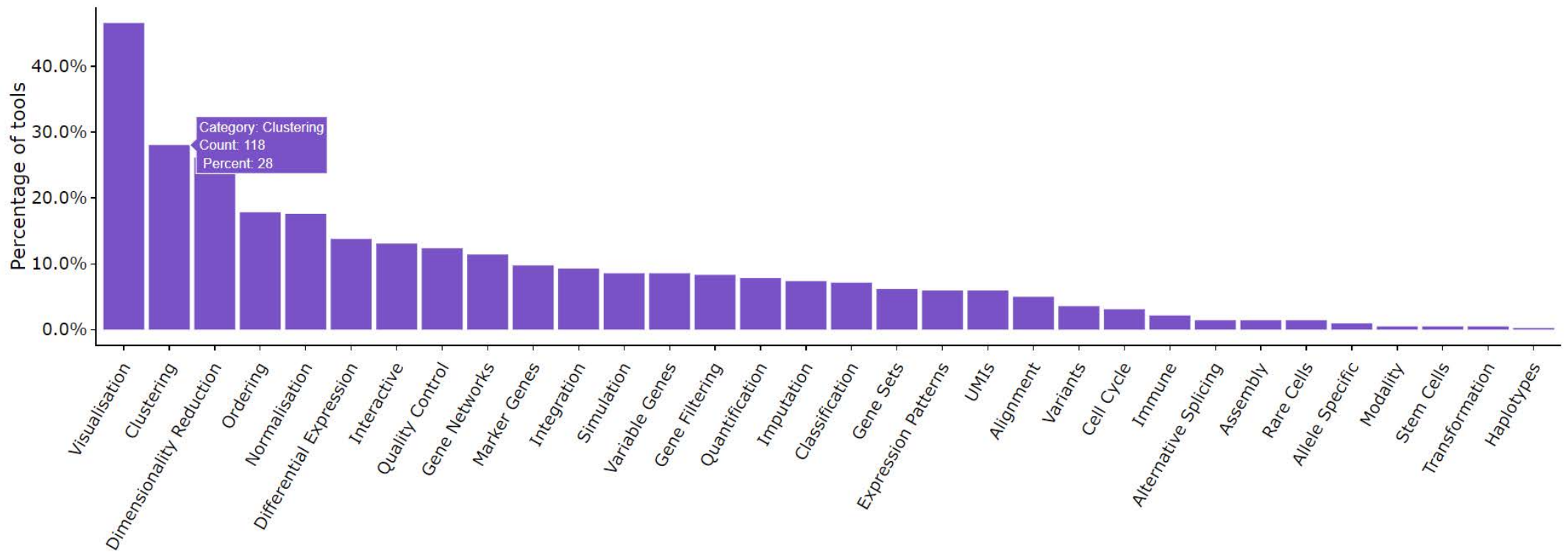
- Feature selection
- Introduction to clustering
  - Hierarchical clustering
  - $k$ -Means clustering
  - Graph-based clustering
- Validation
- **scRNA-seq clustering**
  - Single Cell Consensus Clustering (SC3)
  - Seurat

# scRNA-seq clustering methods

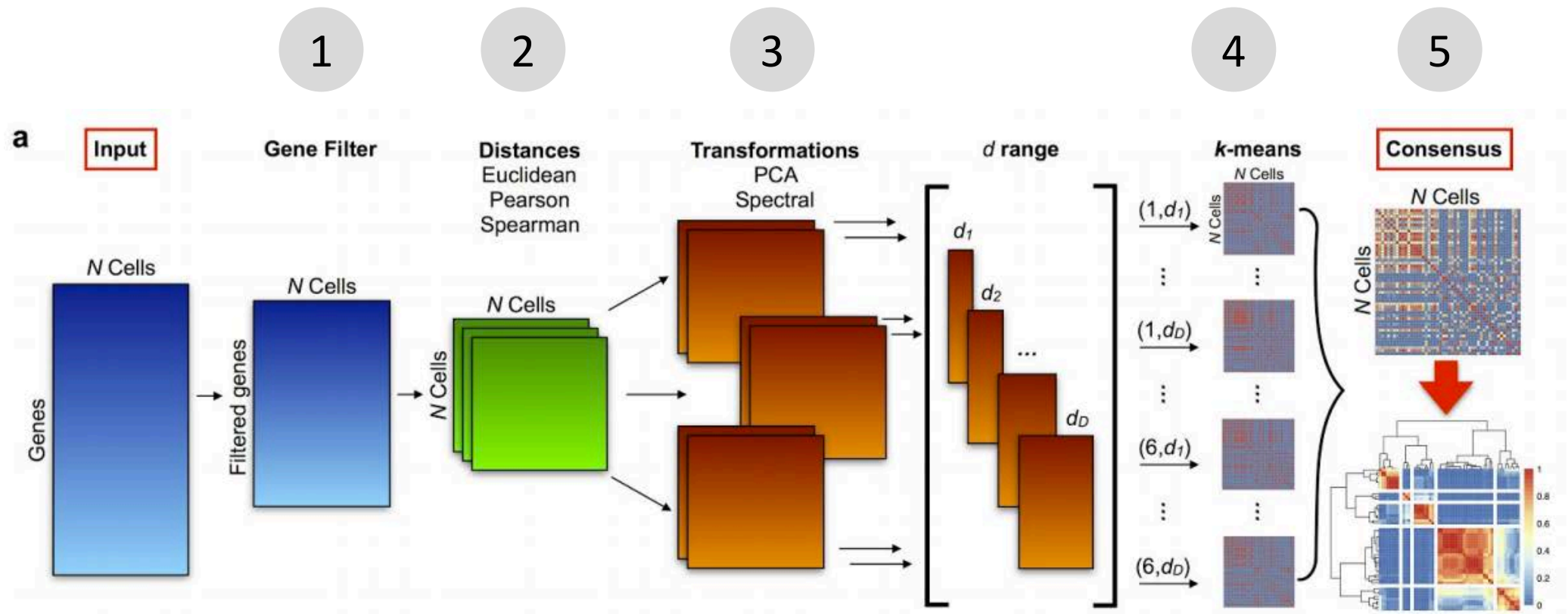
Name	Year	Method type	Strengths	Limitations
scanpy <sup>4</sup>	2018	PCA + graph-based	Very scalable	May not be accurate for small data sets
Seurat (latest) <sup>3</sup>	2016			
PhenoGraph <sup>32</sup>	2015			
SC3 (REF. <sup>22</sup> )	2017	PCA + k-means	High accuracy through consensus, provides estimation of k	High complexity, not scalable
SIMLR <sup>24</sup>	2017	Data-driven dimensionality reduction + k-means	Concurrent training of the distance metric improves sensitivity in noisy data sets	Adjusting the distance metric to make cells fit the clusters may artificially inflate quality measures
CIDR <sup>25</sup>	2017	PCA + hierarchical	Implicitly imputes dropouts when calculating distances	
GiniClust <sup>75</sup>	2016	DBSCAN	Sensitive to rare cell types	Not effective for the detection of large clusters
pcaReduce <sup>27</sup>	2016	PCA + k-means + hierarchical	Provides hierarchy of solutions	Very stochastic, does not provide a stable result
Tasic et al. <sup>28</sup>	2016	PCA + hierarchical	Cross validation used to perform fuzzy clustering	High complexity, no software package available
TSCAN <sup>41</sup>	2016	PCA + Gaussian mixture model	Combines clustering and pseudotime analysis	Assumes clusters follow multivariate normal distribution
mpath <sup>45</sup>	2016	Hierarchical	Combines clustering and pseudotime analysis	Uses empirically defined thresholds and a priori knowledge
BackSPIN <sup>26</sup>	2015	Biclustering (hierarchical)	Multiple rounds of feature selection improve clustering resolution	Tends to over-partition the data
RaceID <sup>23</sup> , RaceID2 (REF. <sup>115</sup> ), RaceID3	2015	k-Means	Detects rare cell types, provides estimation of k	Performs poorly when there are no rare cell types
SINCERA <sup>5</sup>	2015	Hierarchical	Method is intuitively easy to understand	Simple hierarchical clustering is used, may not be appropriate for very noisy data
SNN-Cliq <sup>80</sup>	2015	Graph-based	Provides estimation of k	High complexity, not scalable

Kiselev et al. (<https://doi.org/10.1038/s41576-018-0088-9>)

# scRNA-seq tools



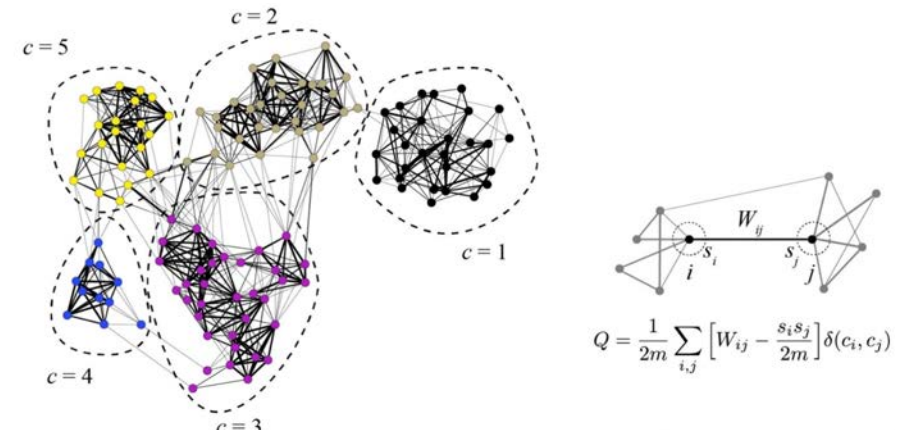
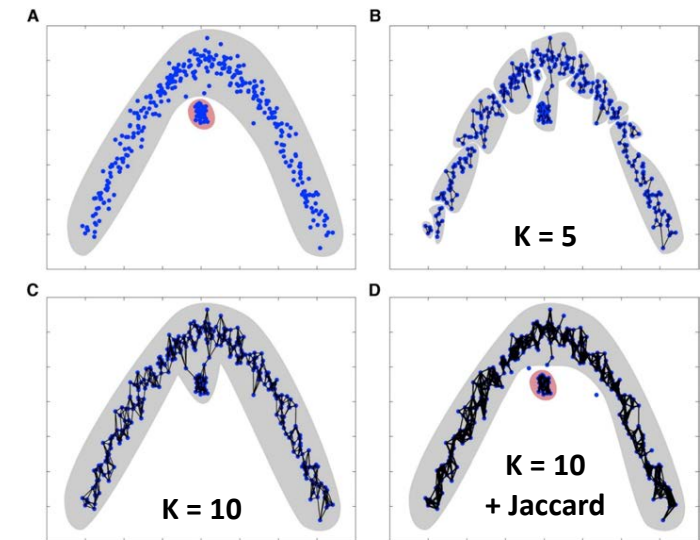
# Single Cell Consensus Clustering – SC3





# Seurat

- 1) Construct KNN (k-nearest neighbor) graph based on the Euclidean distance in PCA space.
- 2) Refine the edge weights between any two cells based on the shared overlap in their local neighborhoods (Jaccard distance).
- 3) Cluster cells by optimizing for modularity (Louvain algorithm)



Xu and Su (<https://doi.org/10.1093/bioinformatics/btv088>)

Levine et al. (<https://doi.org/10.1016/j.cell.2015.05.047>)

# Comparing different clusterings

- Adjusted Rand Index (ARI)

Given a set  $S$  of  $n$  elements, and two groupings or partitions (e.g. clusterings) of these elements  $X = \{X_1, X_2, \dots, X_r\}$  and  $Y = \{Y_1, Y_2, \dots, Y_s\}$

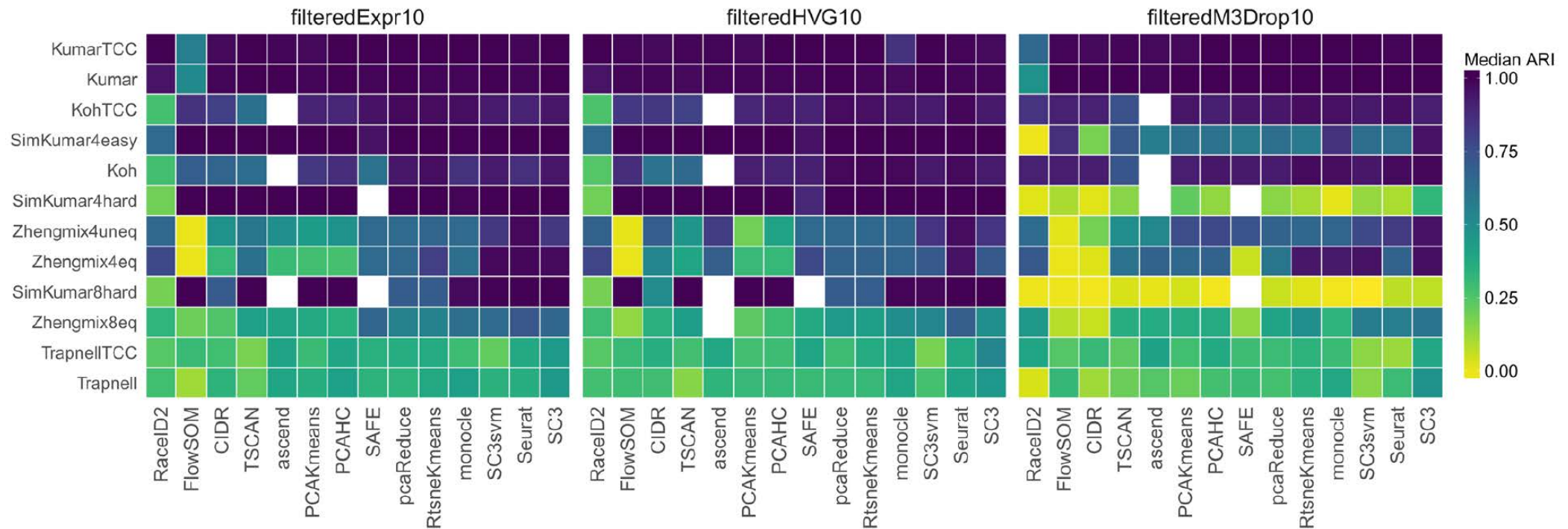
Confusion matrix/contingency table

$X \setminus Y$	$Y_1$	$Y_2$	$\dots$	$Y_s$	Sums
$X_1$	$n_{11}$	$n_{12}$	$\dots$	$n_{1s}$	$a_1$
$X_2$	$n_{21}$	$n_{22}$	$\dots$	$n_{2s}$	$a_2$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
$X_r$	$n_{r1}$	$n_{r2}$	$\dots$	$n_{rs}$	$a_r$
Sums	$b_1$	$b_2$	$\dots$	$b_s$	

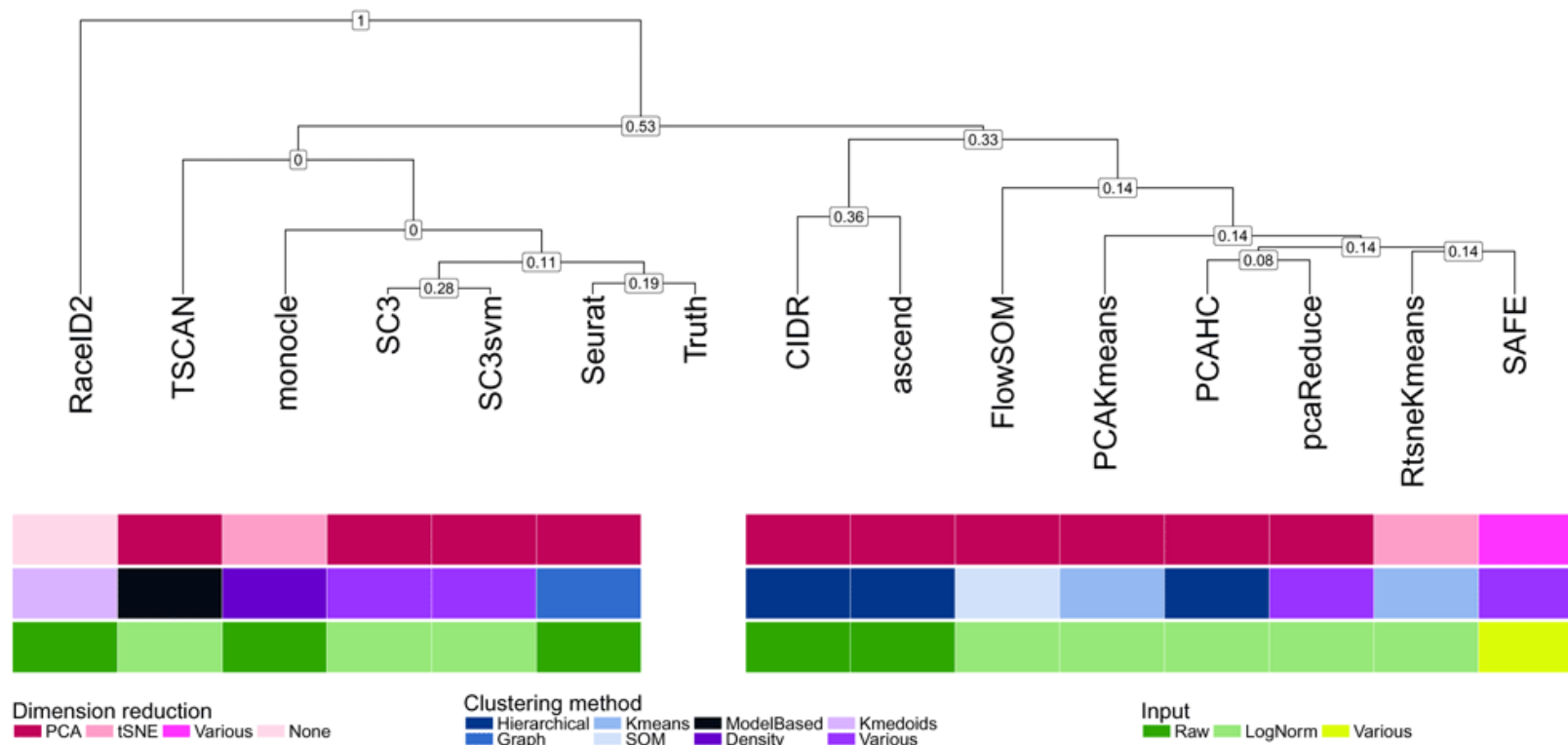
$$n_{ij} = |X_i \cap Y_j|$$

$$\widehat{ARI} = \frac{\overbrace{\sum_{ij} \binom{n_{ij}}{2}}^{\text{Index}} - \overbrace{[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}] / \binom{n}{2}}^{\text{Expected Index}}}{\underbrace{\frac{1}{2} [\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}]}_{\text{Max Index}} - \underbrace{[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}] / \binom{n}{2}}_{\text{Expected Index}}}$$

# Benchmarking scRNA-seq clustering methods



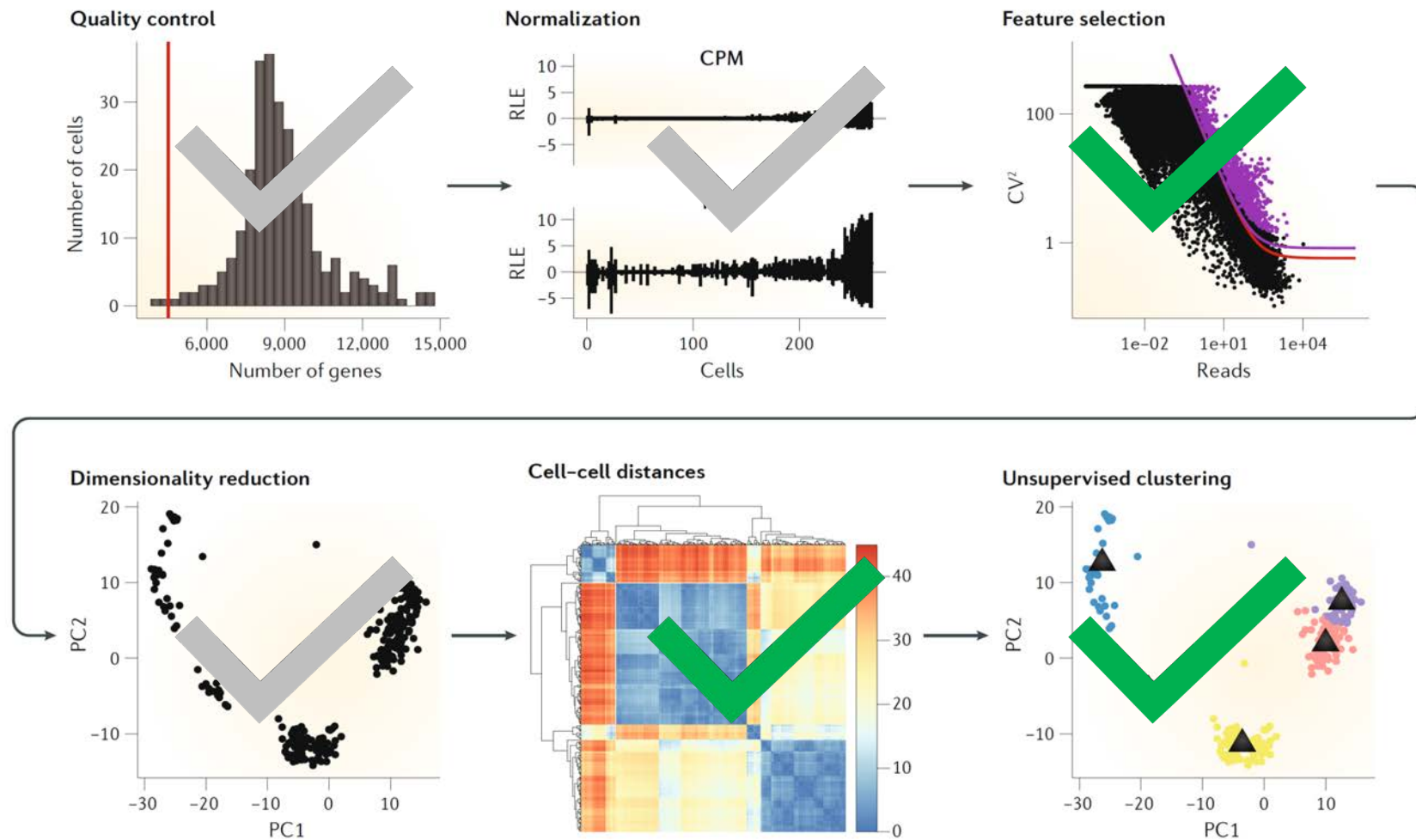
# Benchmarking scRNA-seq clustering methods



# Challenges in clustering

- What is a cell type
  - *Validation using independent tech is important.*
  - *Commonly, spatial mapping (smFISH)*
- **Scalability:** in the last few years the number of cells in scRNA-seq experiments has grown by several orders of magnitude from  $\sim 10^2$  to  $\sim 10^6$ 
  - *Computational efficiency*
  - *Visual exploration, crowding problem*

# Summary



# Clustering practical

- Feature selection (HVG)
- Dimensionality reduction: select principal components
- Hierarchical clustering: distances and linkage methods
- tSNE +  $k$ -Means
- Graph-based clustering

# Resources

- Kiselev et al. "Challenges in unsupervised clustering of single- cell RNA- seq data"  
<https://doi.org/10.1038/s41576-018-0088-9>
- Duò et al. " A systematic performance evaluation of clustering methods for single-cell RNA-seq data"  
<https://doi.org/10.12688/f1000research.15666.2>
- Orchestrating Single-Cell Analysis with Bioconductor  
<https://osca.bioconductor.org/>
- Hemberg single cell course: Analysis of single cell RNA-seq data  
<https://scrnaseq-course.cog.sanger.ac.uk/website/index.html>
- Slides Åsa Björklund (NBIS, SciLifeLab)  
<https://github.com/NBISweden/workshop-scRNAseq/tree/master/slides2019>