scRNAseq clustering tools

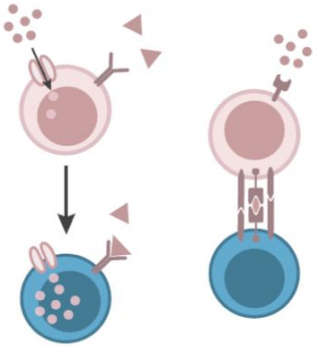
Åsa Björklund
asa.bjorklund@scilifelab.se

NBIS NATIONAL BIOINFORMATICS INFRASTRUCTURE SWEDEN

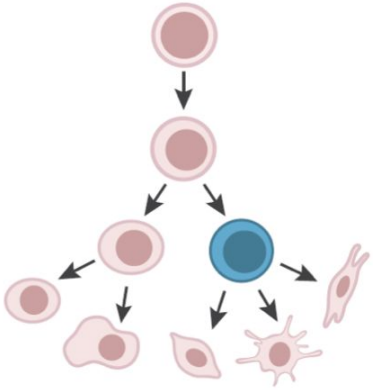
SciLifeLab

Cell identity

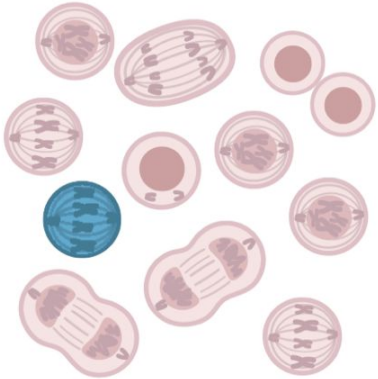
Environmental stimuli



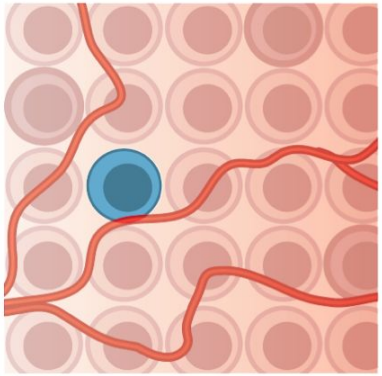
Cell development



Cell cycle



Spatial context



Wagner et al. Nat. Biotech 2016

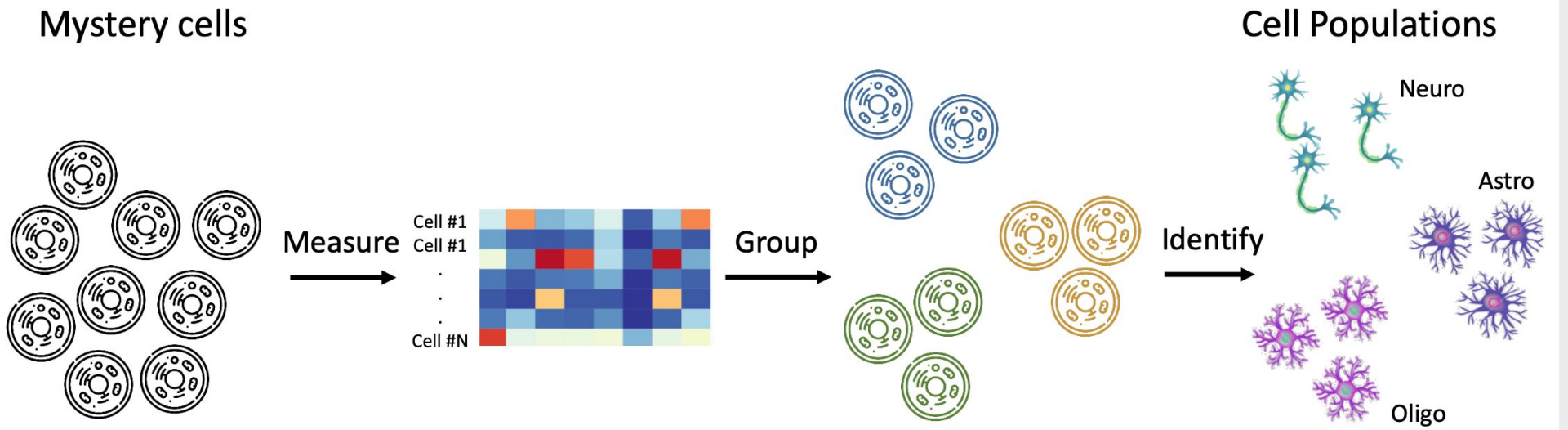
What is a cell type?

- A cell that performs a specific function?
- A cell that performs a specific function at a specific location/tissue?
- Not clear where to draw the line between cell types and **subpopulations** within a cell type.
- Also important to distinguish between **cell type** and **cell state**.
  - A cell state may be infected/non infected
  - Metabolically active/inactive
  - Cell cycle stages
  - Apoptotic

Outline

- Basic clustering theory
- Graph theory introduction
- Distance metrics
- scRNAseq clustering with graphs
- Examples of different tools

How can we identify populations?



Considerations for clustering

- Hypotheses:
  - What is a cell type? What cell types are in my tissue?
  - What is the number of clusters *k*?
- Choices:
  - Gene set selection
  - Similarity measure / Space to calculate similarity
  - Algorithm and hyper parameters of that algorithm.
- Different choice leads to different results. Validate, interpret and repeat steps.

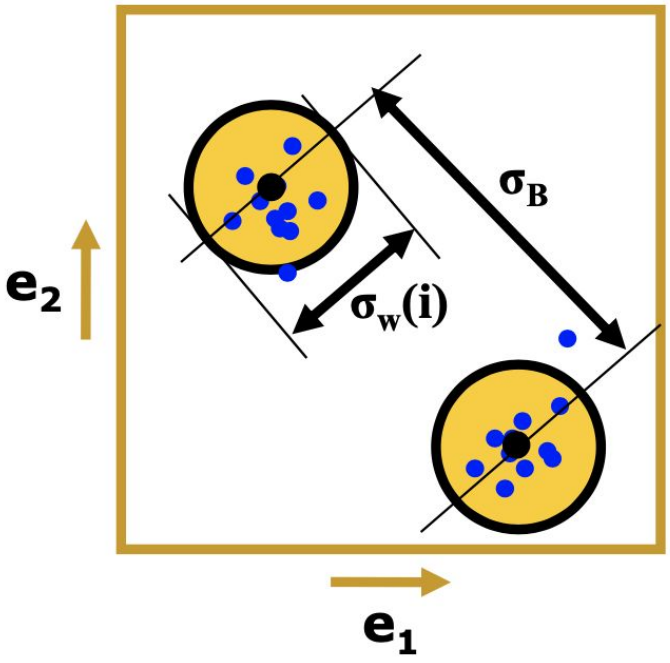
What is clustering?

- “The process of organizing objects into groups whose members are similar in some way”
- Typical methods are:
  - Hierarchical clustering
  - K-means clustering
  - Density based clustering
  - Graph based clustering

The main idea

- Structure when:
  1) Samples within cluster resemble each other *(within variance, σW(i))*
  2) Clusters deviate from each other

  (between variance, σB)



Group samples such that:

$$\min\left(\frac{\sum_{\forall clusters}\sigma_W(i)}{\sigma_B}\right) \rightarrow \sigma_W\text{: small \& } \sigma_B\text{ : large}$$

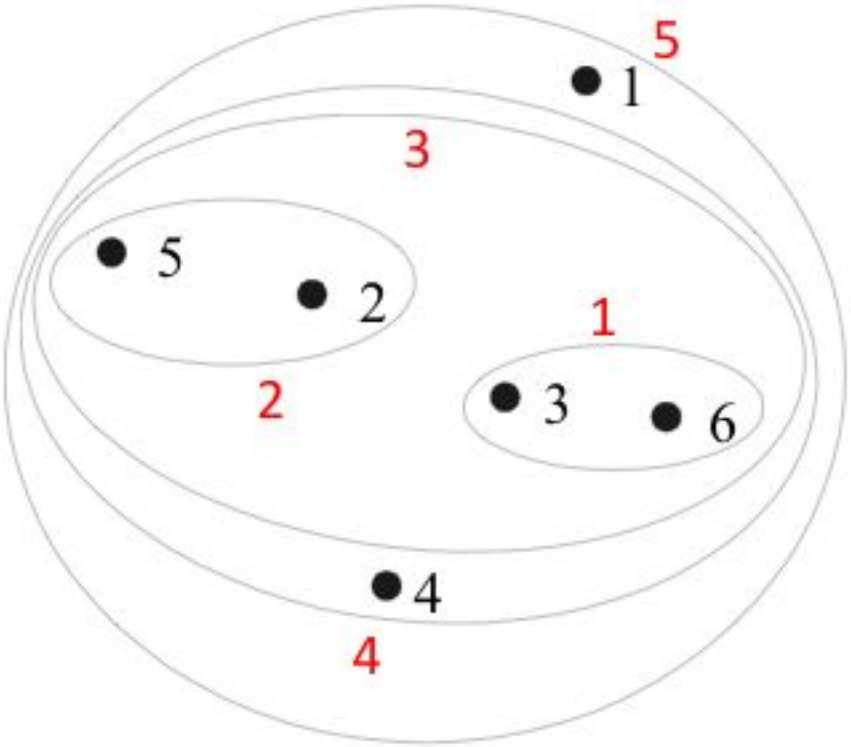
NBIS NATIONAL BIOINFORMATICS INFRASTRUCTURE SWEDEN

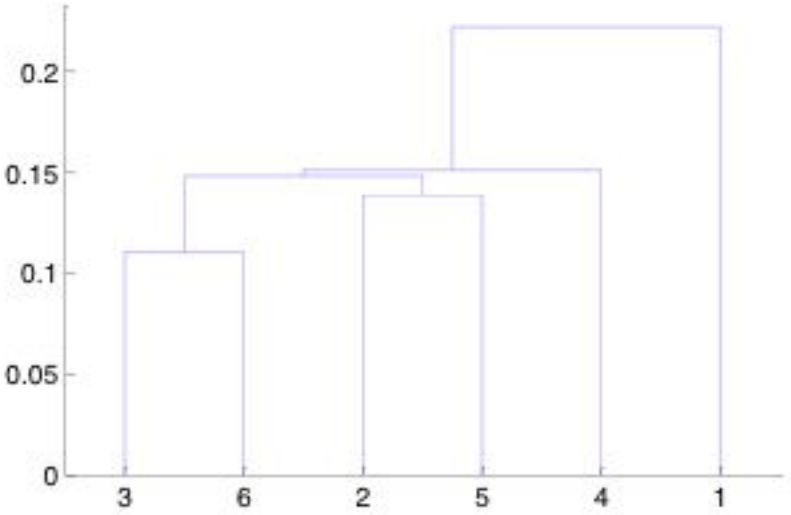
SciLifeLab

Hierarchical clustering

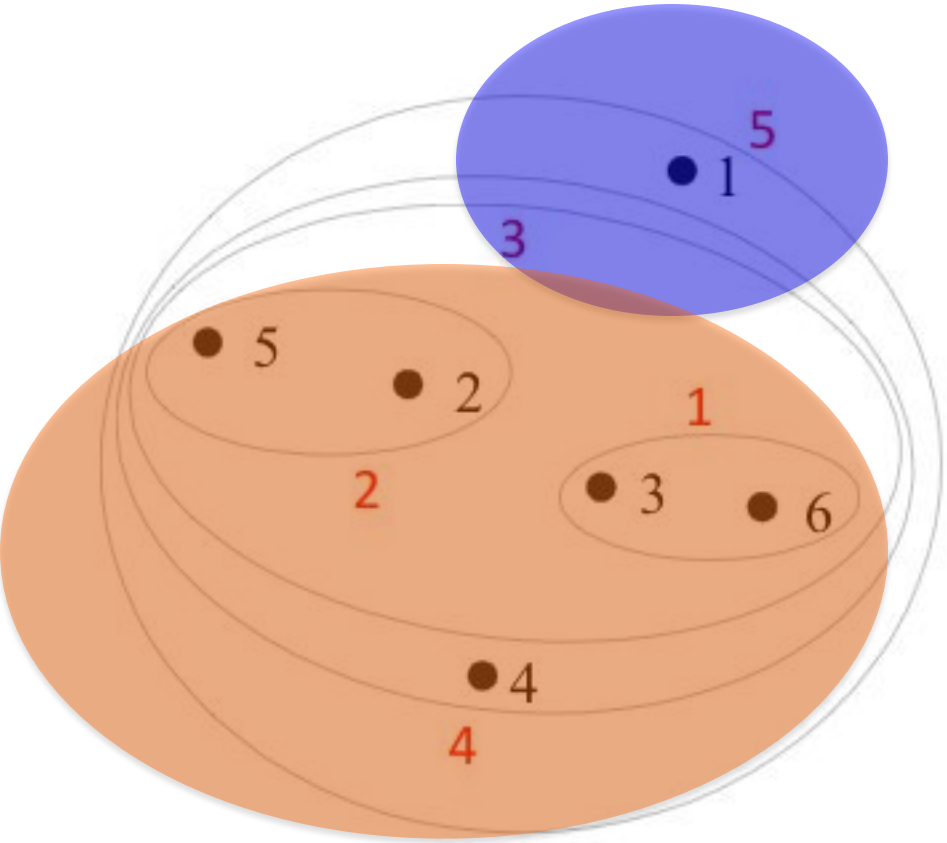
- Builds on **distances** between data points
- **Agglomerative** – starts with all data points as individual clusters and joins the most similar ones in a bottom-up approach
- **Divisive** – starts with all data points in one large cluster and splits it into 2 at each step. A top-down approach
- Final product is a **dendrogram** representing the decisions at each merge/division of clusters

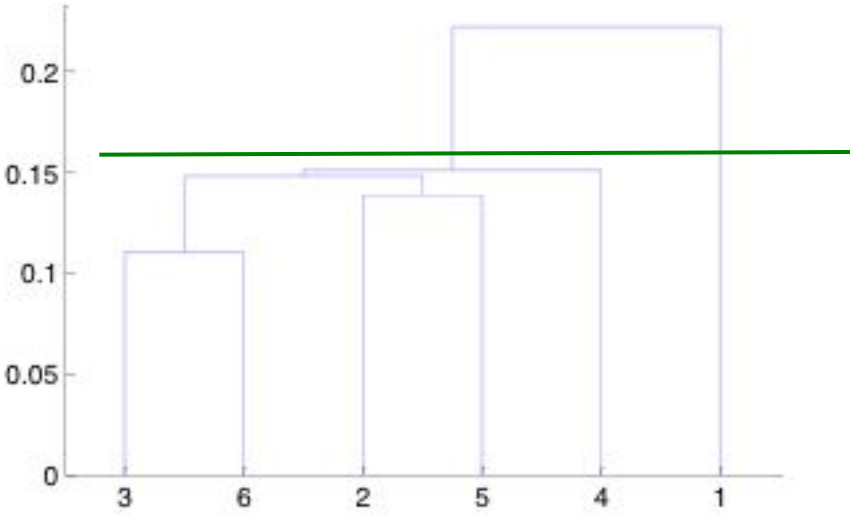
Hierarchical clustering





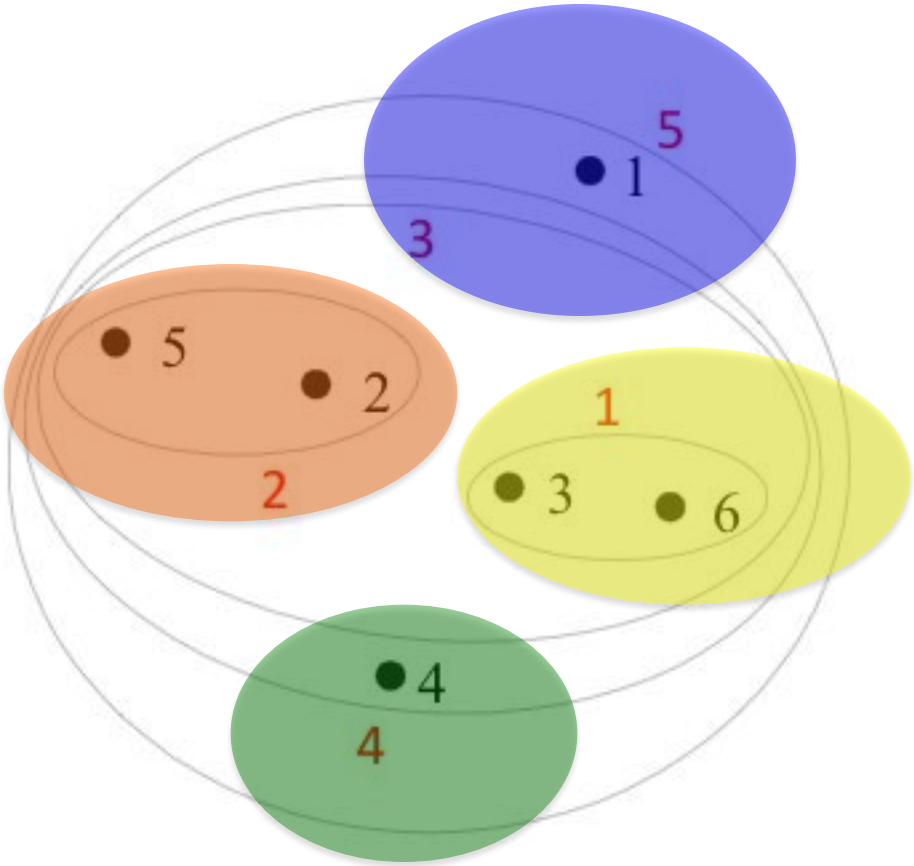
Hierarchical clustering

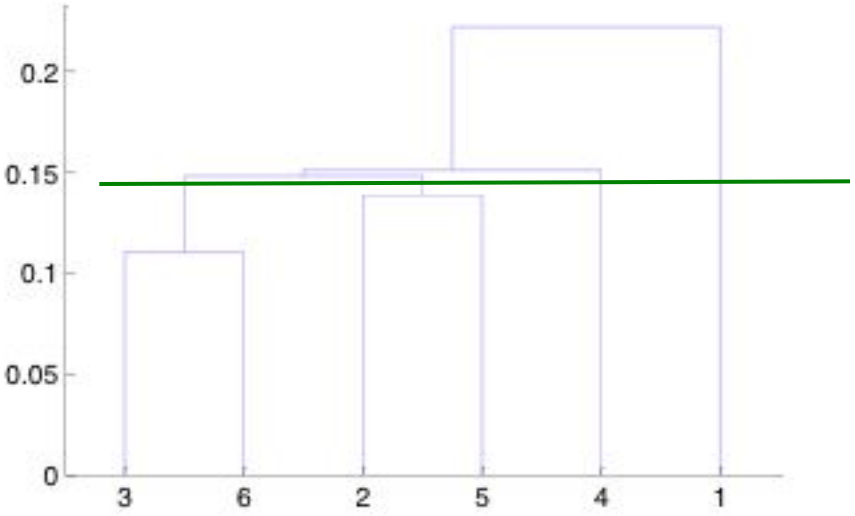




Clusters are obtained by cutting the tree at a desired level

Hierarchical clustering

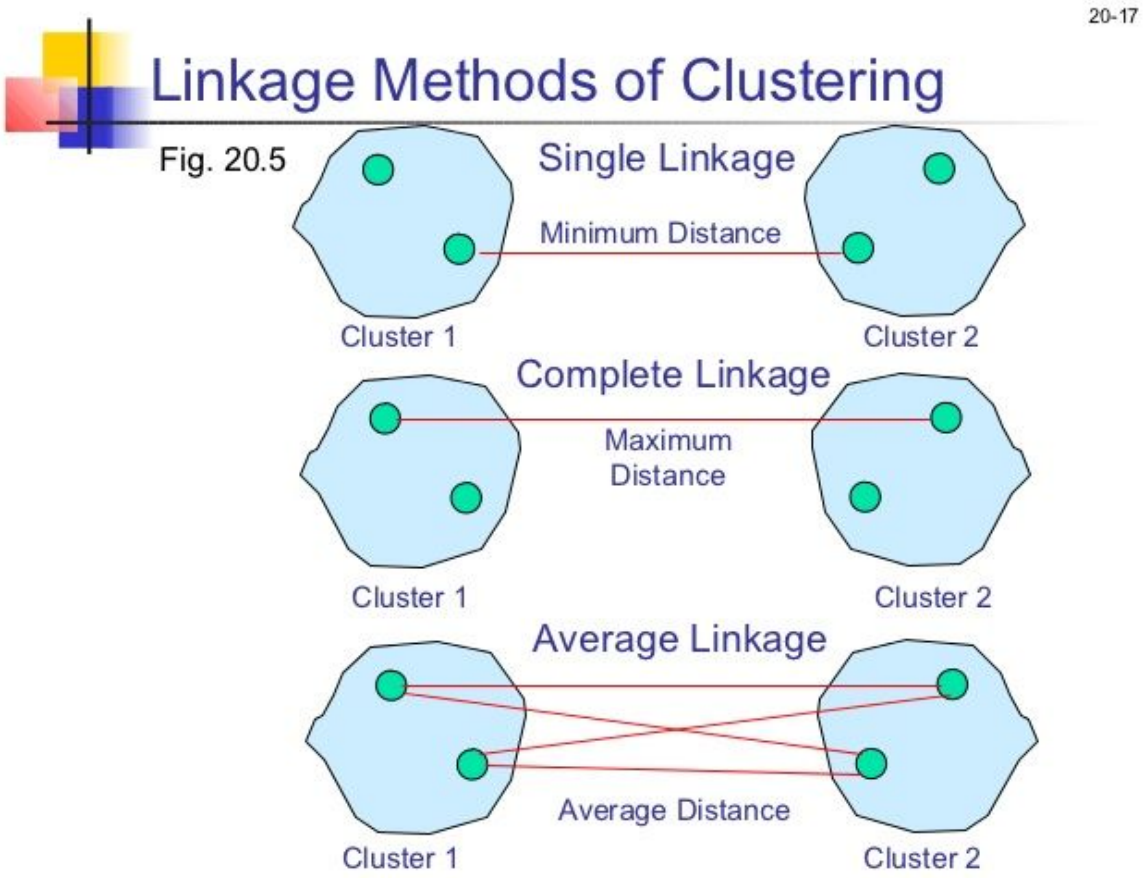




Clusters are obtained by cutting the tree at a desired level

Linkage criteria

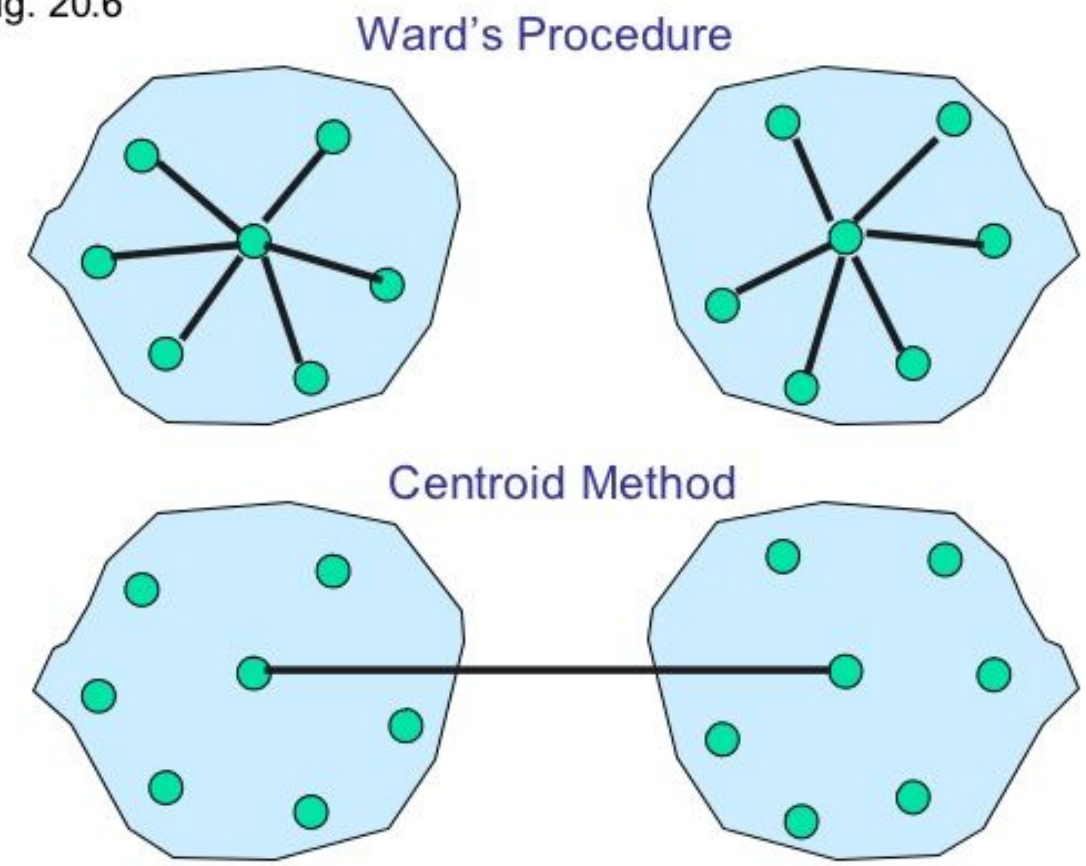
- Calculation of similarities between 2 clusters (or a cluster and a data point)



http://www.slideshare.net/uzairjavedsiddiqui/malhotra20

Other Agglomerative Clustering Methods

Fig. 20.6

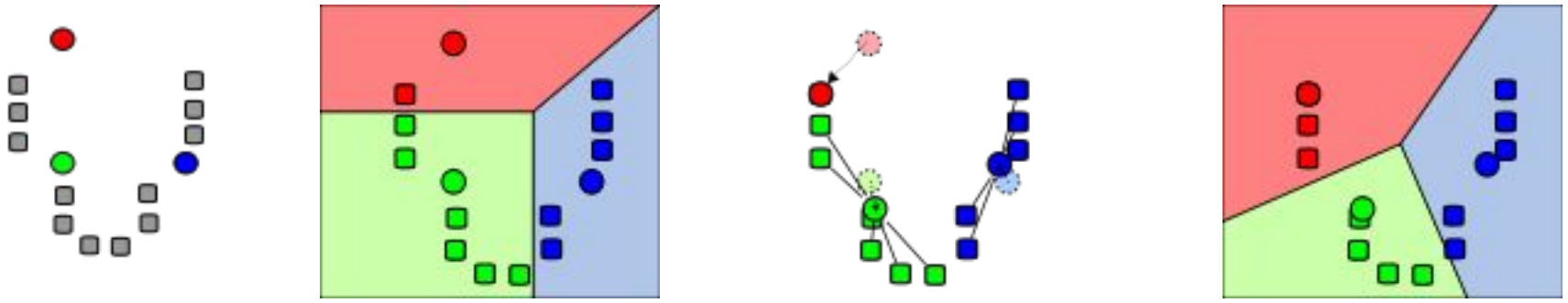


- Ward (minimum variance method). Similarity of two clusters is based on the increase in squared error when two clusters are merged.

K-means clustering

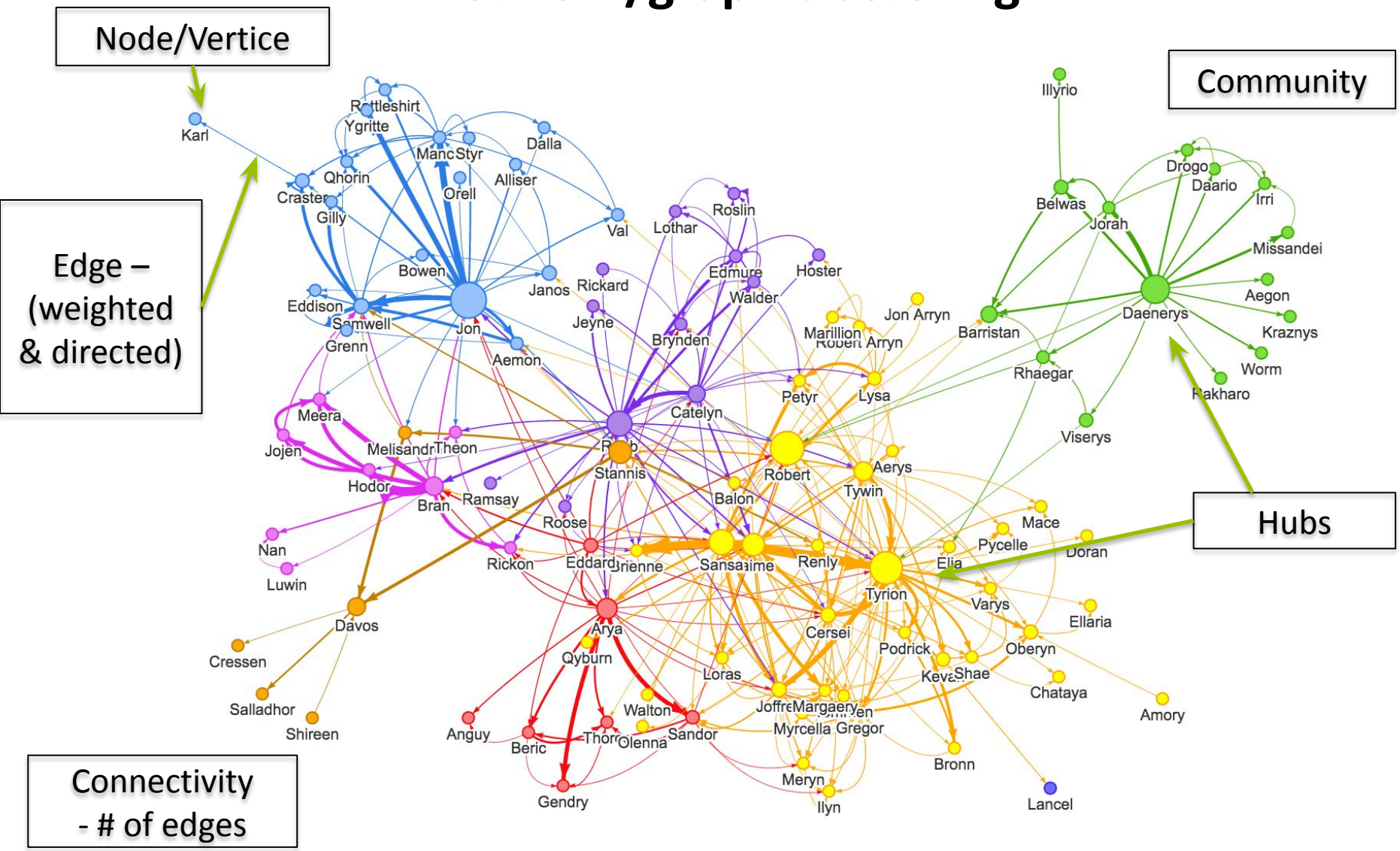
1. Starts with random selection of cluster centers (centroids)
2. Then assigns each data points to the nearest cluster
3. Recalculates the centroids for the new cluster definitions
4. Repeats steps 2-3 until no more changes occur.

Can use same distance measures as in hclust.



https://en.wikipedia.org/wiki/K-means_clustering

Network/graph clustering



(http://www.lyonwj.com/2016/06/26/graph-of-thrones-neo4j-social-network-analysis/)

NBIS NATIONAL BIOINFORMATICS INFRASTRUCTURE SWEDEN

SciLifeLab

Types of graphs

- The ***k*-Nearest Neighbor** (***k*NN**) graph is a graph in which two vertices *p* and *q* are connected by an edge, if the distance between *p* and *q* is among the *k*-th smallest distances from *p* to other objects from *P*.
- The **Shared Nearest Neighbor** (**SNN**) graph has weights that defines proximity, or similarity between two edges in terms of the number of neighbors (i.e., directly connected vertices) they have in common.

SNN graph

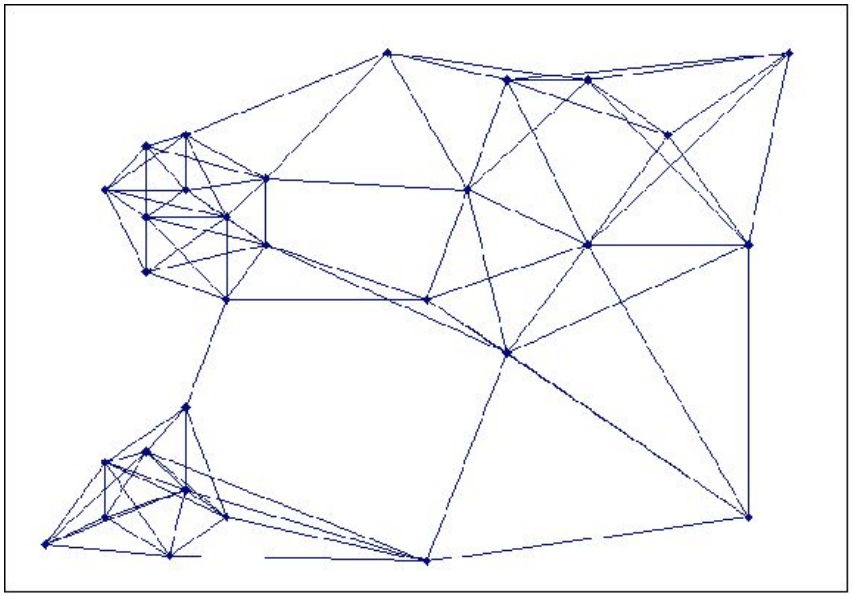


Figure 2. Near Neighbor Graph

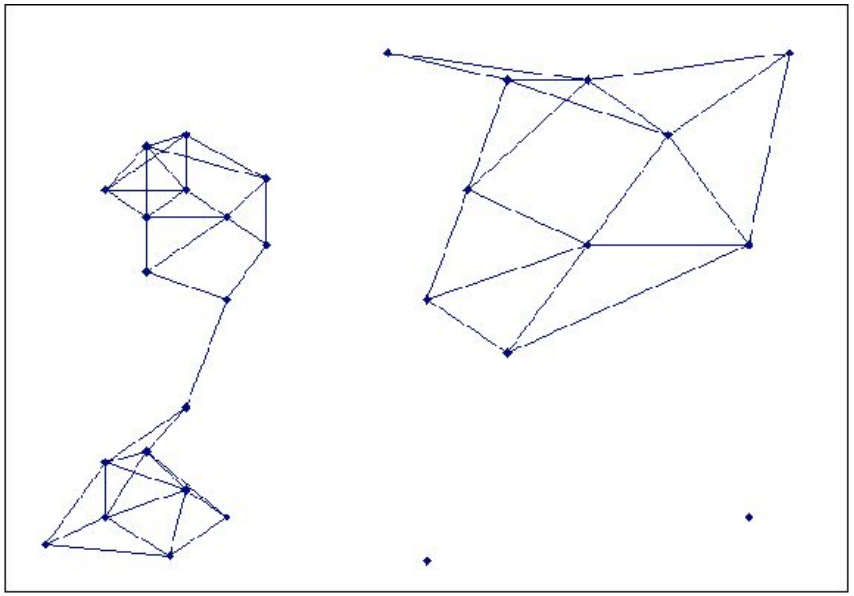


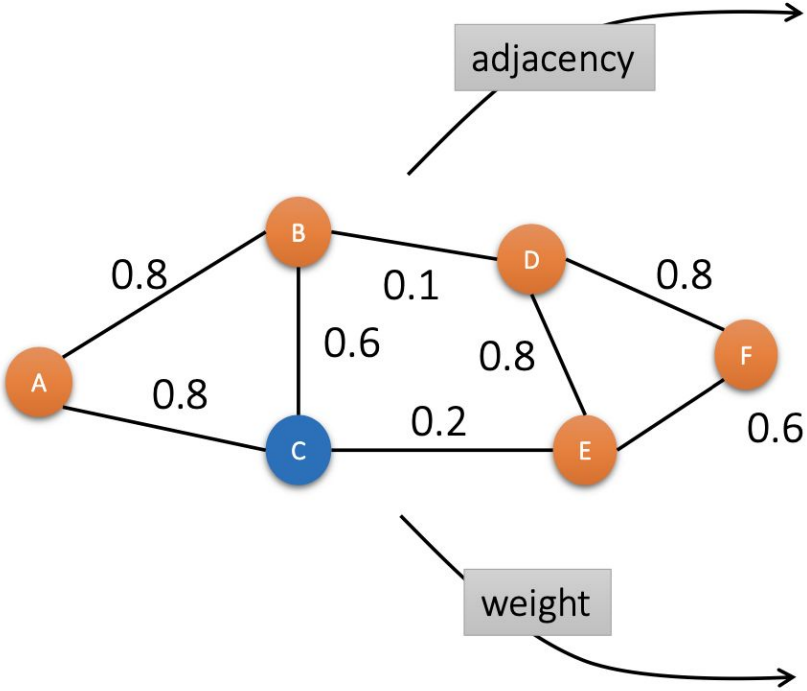
Figure 3. Unweighted Shared Near Neighbor Graph

(Ertöz et al. Semantic scholar, 2002)

SNN graph

- A common measure of shared neighbors is the Jaccard index:
  - Shared neighbors / Total neighbors for both. $J(A,B) = \frac{|A \cap B|}{|A \cup B|}$
- Other measures includes rank, number of shared neighbors, overlap coefficient
- Common to do pruning – remove all edges between nodes with e.g. Jaccard similarity < cutoff.

Graphs, adjacency and weight matrices

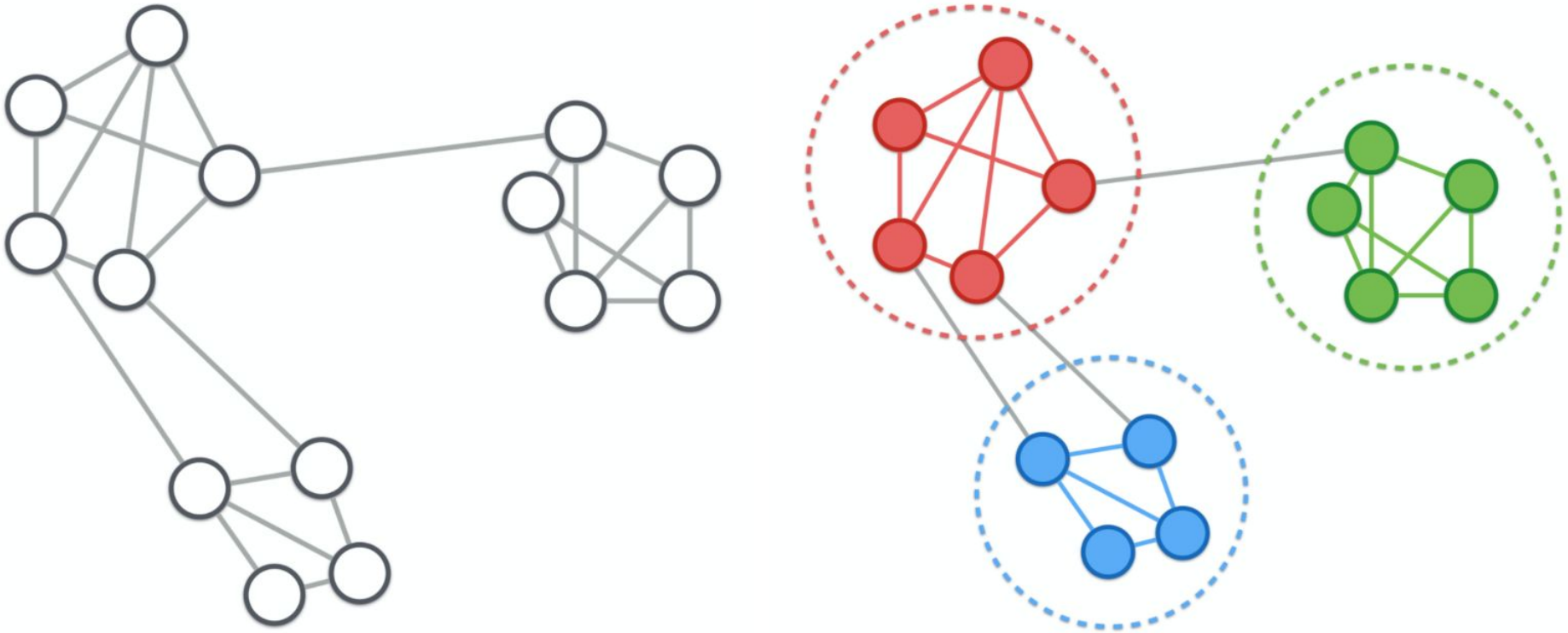


$$A = \begin{array}{c} \\ A \\ B \\ C \\ D \\ E \\ F \end{array} \begin{array}{c} \begin{array}{cccccc} A & B & C & D & E & F \end{array} \\ \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 &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

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

Community detection

Communities, or clusters, are usually groups of vertices having higher probability of being connected to each other than to members of other groups.

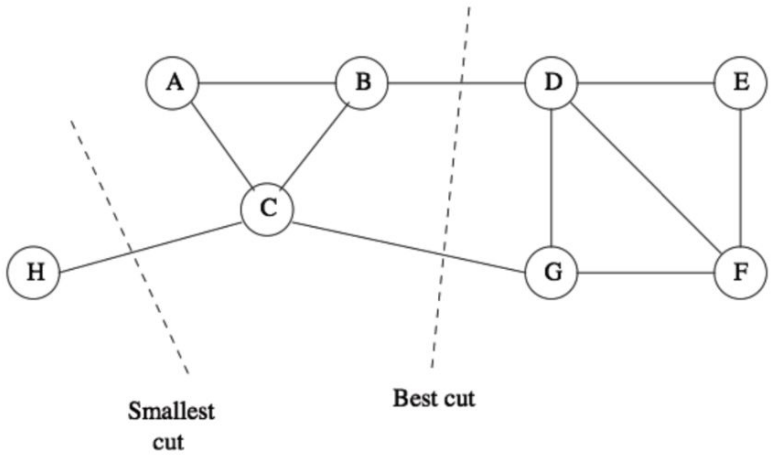


Community detection

- Main objective is to find a group (community) of vertices with more edges **inside** the group than edges linking vertices of the group with the rest of the graph.

Graph cuts

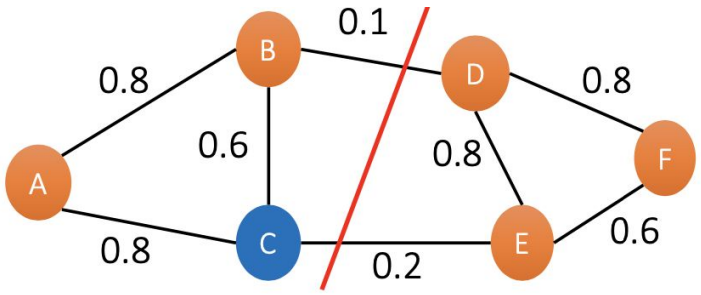
- Graph cut partitions a graph into subgraphs
- Cut cost is the sum of weights of the edges.
- Clustering by graph cuts: find the smallest cut that bi-partitions the graph
- The smallest cut is not always the best cut – may give many small disjoint cluster



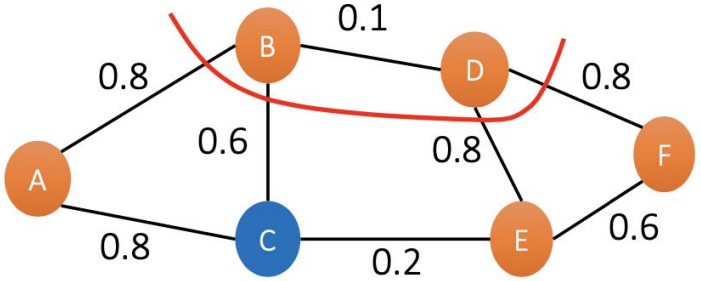
Normalized cut

- Normalized cut computes the cut cost as a fraction of the total edge connections to all the nodes in the graph.

- cut(S,T) = 0.1 + 0.2 = 0.3
- vol(S)= 0.3 + 0.6 + 0.8 + 0.8 = 2.5
- vol(T)= 0.3 + 0.8 + 0.8 + 0.6 = 2.5
- Ncut(S,T) = 0.3/2.5 + 0.3/2.5 = 0.24



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

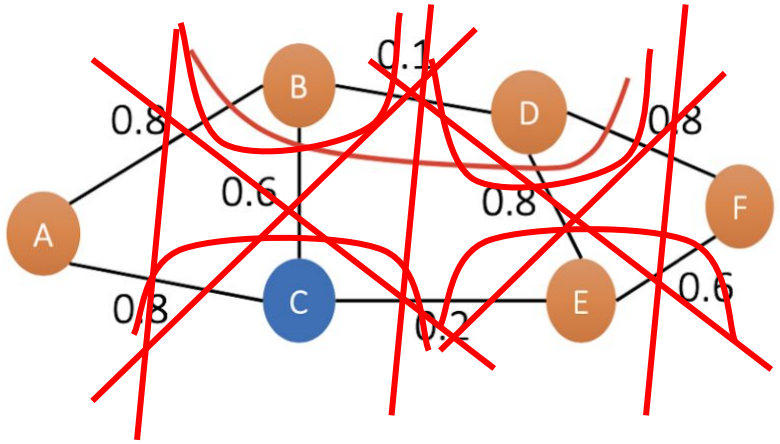


NBIS NATIONAL BIOINFORMATICS INFRASTRUCTURE SWEDEN

SciLifeLab

Normalized cut

- Searching for the best normalized cut is NP-hard
- We need a heuristic method to solve the problem:
  - Spectral clustering
  - Markov clustering
  - Louvain
  - Leiden
  - …



For single cell data

- Can start with distances based on correlation, euklidean distances in PCA space etc. Same as for hclust/k-means.
- Build a KNN graph with cells as vertices.
  - Find **k** nearest neighbors to each cell.
  - The size of **k** will strongly influence the network structure.
- Can create weighted network based on shared neighbors (SNN).
- Find clusters with community detection method.
- Graphs can also be used for trajectory analysis etc.

Community detection

- Louvain / Leiden most often used.
- Cannot define number of clusters – is often tuned by resolution parameter.
- Graph building parameters like K (number of neighbors) and pruning cutoffs will influence clusters quite a bit.
- Singleton cells will in most cases be assigned to closest cluster.

How to work with networks

- Igraph package – implemented for both R, python and Ruby
- Has most commonly used layout optimization methods and community detection methods implemented.

- Simple R example at:

https://jef.works/blog/2017/09/13/graph-based-community-detection-for-clustering-analysis/

- Tutorial to igraph at:

http://kateto.net/networks-r-igraph

- Example how to build your own graph with scRNAseq data: https://github.com/NBISweden/workshop-scRNAseq/blob/feb2023/oldlabs/igraph.md

Distance between cells

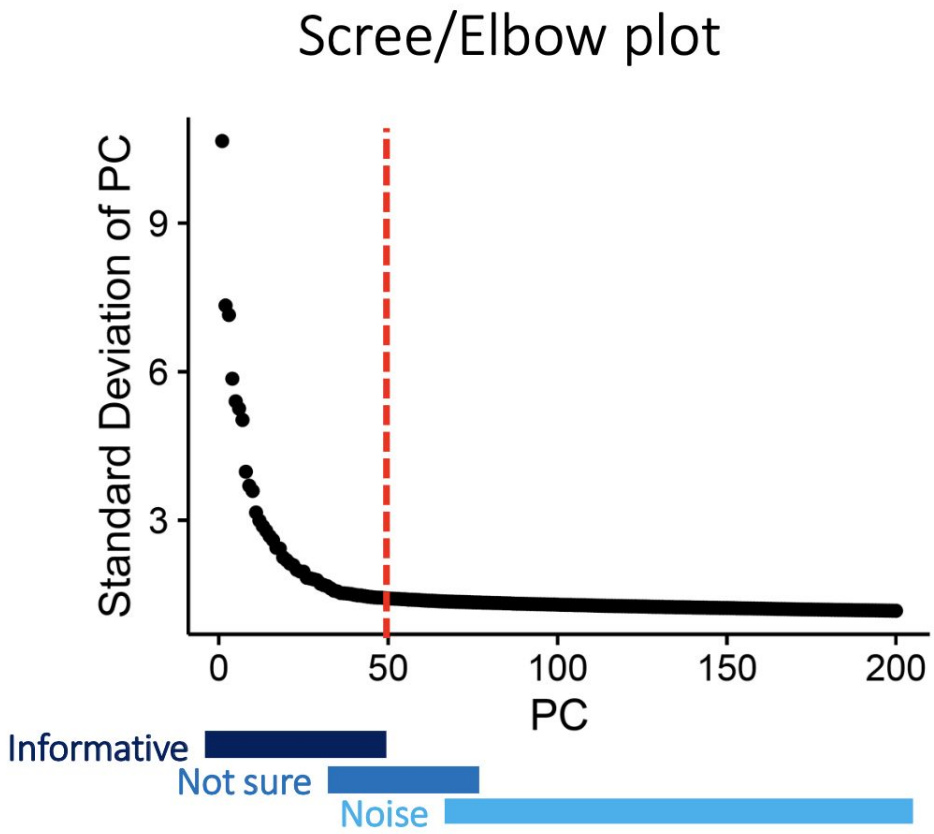
- All clustering methods need to define distances between cells. Things to consider are:
  - What gene set should be included?
    - Commonly used: Highly variable genes
  - What space to calculate distance in?
    - Commonly done in PCA space, or an integrated dimred space.
    - Can also be full space, tSNE, UMAP etc.
  - How many dimensions to include?
  - What distance measure?

Different distance measures

- Most commonly used in scRNA-seq:
  - Euclidean distance
  - Inverted pairwise correlations (1-correlation)
- Other common methods are:
  - Cosine distance
  - Manhattan distance
  - Mahalanobis distance
  - Maximum distance

Selection of 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
- Depending on the heterogeneity of your data more/less PCs should be selected.



NBIS NATIONAL BIOINFORMATICS INFRASTRUCTURE SWEDEN

SciLifeLab

Seurat clustering

- FindNeighbors:
  - First construct a KNN (k-nearest neighbor) graph – default is based on the euclidean distance in PCA space
  - Then SNN graph the edge weights between any two cells based on the shared overlap in their local neighborhoods (Jaccard distance) and pruning of distant edges.
- Important parameters:
  - reduction: default is “pca”
  - dims: number of PCs - default 1:10
  - k.param: Number of neighbors in KNN graph - default 20
  - prune.snn: Cutoff for pruning - default 1/15

Seurat clustering

- FindClusters: To cluster the cells, modularity optimization:
  - Louvain
  - Louvain with multilevel refinement
  - Leiden
  - SLM
- Important parameters:
  - resolution: default 0.8, larger values give more communities, smaller gives less.
  - Algorithm

Scran clustering

- buildKNNGraph: Constructs the KNN graph
- buildSNNGraph: Constructs KNN and then SNN graph. Adds weighted edges to cells that shares neighbors.
  - Allows for different similarity measures, default is “rank” but also includes “jaccard” or “number”
- Important parameters:
  - use.dimred: dimensionality reduction to use
  - k: number of neighbors
  - type: weighting method

Scran clustering

- Community detection is done with the igraph package.
  - cluster_louvain
  - cluster_leiden
  - cluster_infomap
  - Many more…

Scanpy clustering

- sc.pp.neighbors – creates KNN graph
  - Has many different options for distance calculation, default is euclidean.
  - No SNN graph construction
  - Method is by default “umap” but can also have weight by gaussian kernel.
- Clustering:
  - sc.tl.leiden
  - sc.tl.louvain
  - Can specify resolution like in Seurat.

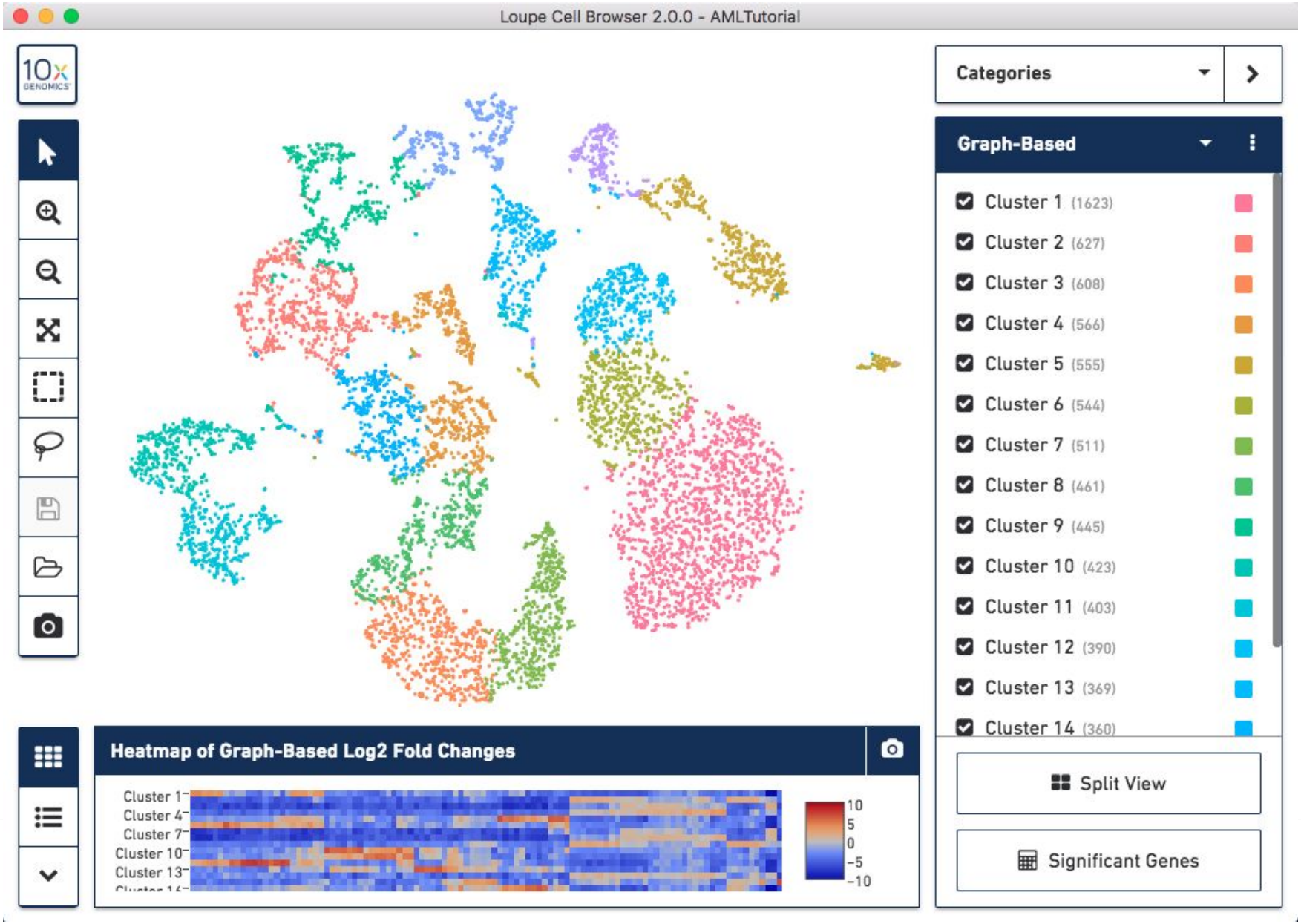
Some comments on clustering space

- Clustering on 2D umap/tSNE – looks nice in plot, but you always loose information when reducing data to 2D.
- Clustering with more dimensions (PCs, ICs, etc.) – chance to detect more different celltypes.
- Rule of thumb:
  - Heterogeneous dataset with many celltypes – use more variable genes and more dimensinos
  - Homogeneous dataset with few/single celltypes – use fewer genes/dimensions.

Why does my clustering not agree with the umap?

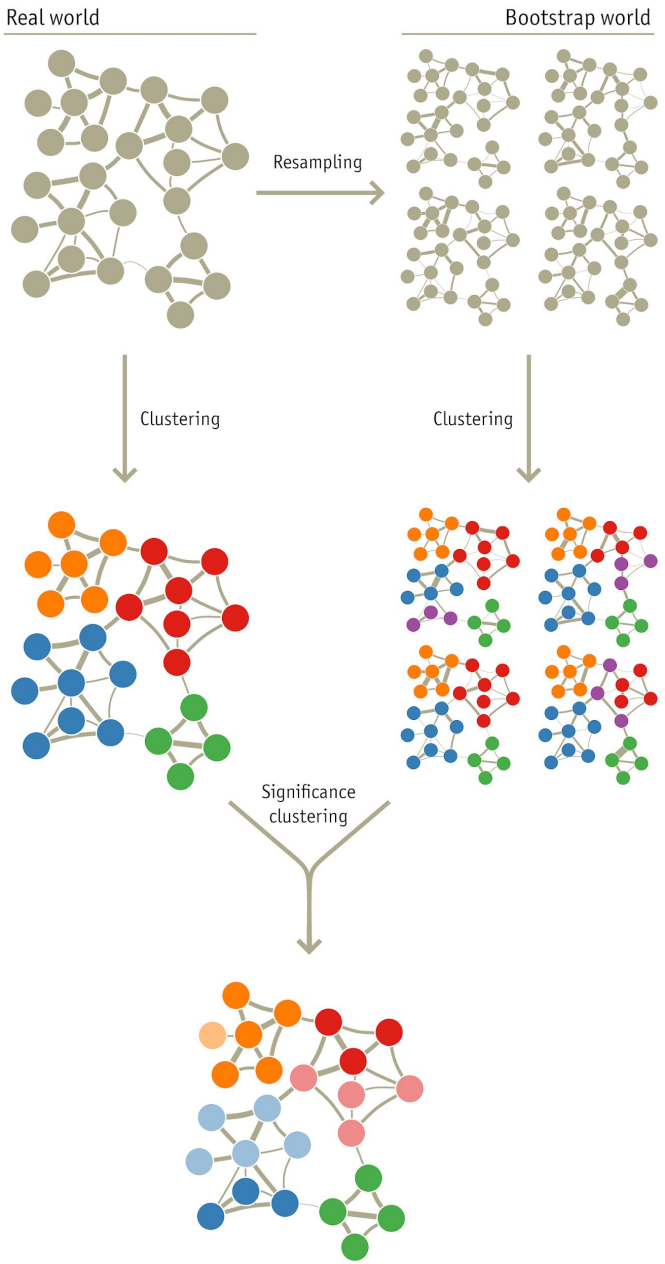
- Both are done in same space (nPC, var.genes etc.) but different algorithms to optimize
- UMAP needs to make compromises to compress the data to 2D
- Cells that are different between umap and clustering are often problematic cells – doublets or low quality cells.
- UMAP on graph is often more similar to clustering (default in scanpy).

Loupe – Cell Browser, from 10X Genomics



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.
- Most scRNAseq packages do not include any bootstrapping. Scran has function **bootstrapCluster.**



(Rosvall et al. *Plos One* 2010)

Which clustering method is best?

- Depends on the input data
- Consistency between several methods gives confidence that the clustering is robust
- The clustering method that is most consistent – best bootstrap values is not always best
- In a simple case where you have clearly distinct celltypes, simple hierarchical clustering based on euclidean or correlation distances will work fine.

Comparison of clustering methods

F1000Research

F1000Research 2018, 7:1141 Last updated: 11 SEP 2018

Check for updates

RESEARCH ARTICLE

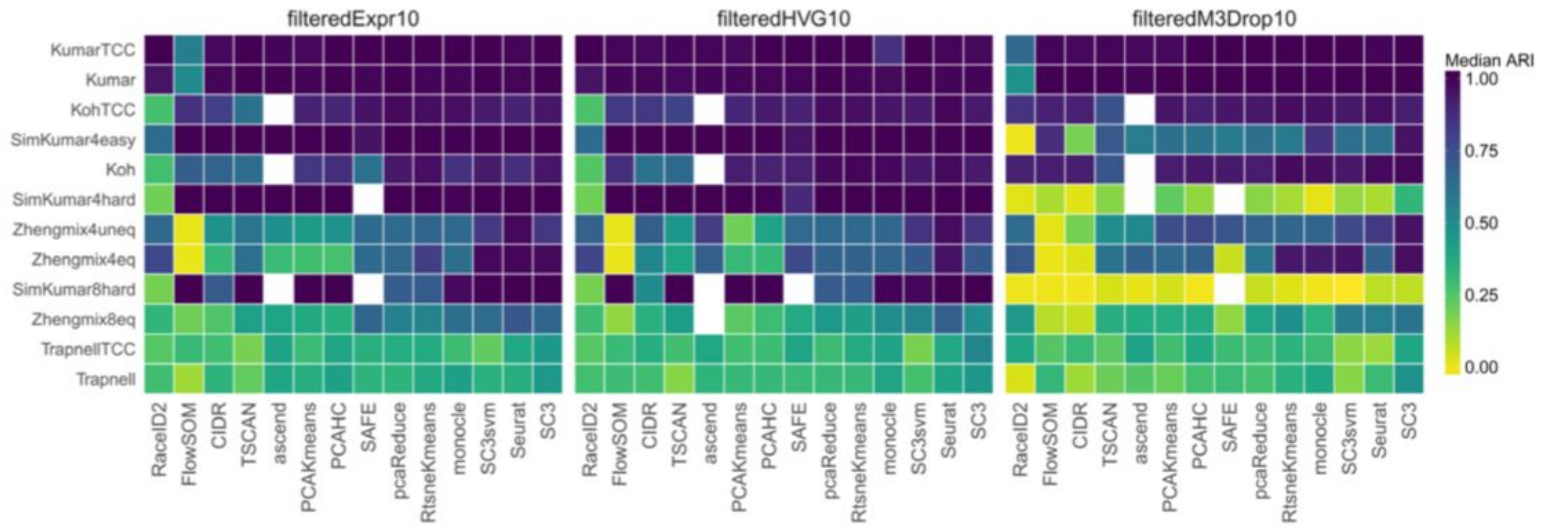
REVISED A systematic performance evaluation of clustering methods for single-cell RNA-seq data [version 2; referees: 2 approved]

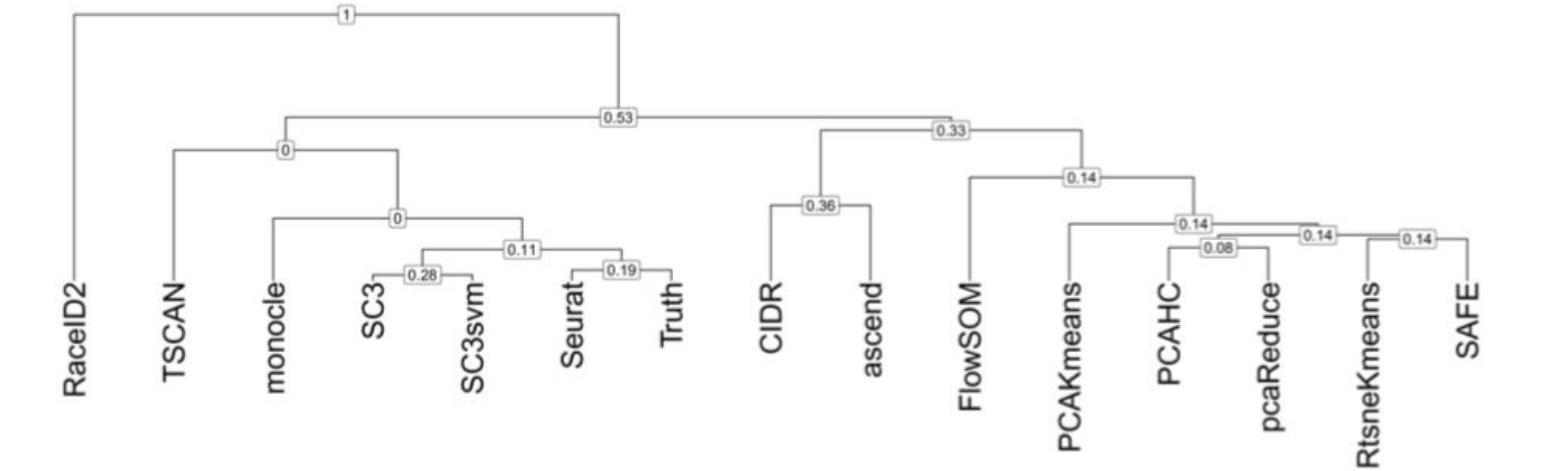
Angelo Duò[1,2], Mark D. Robinson[1,2], Charlotte Soneson[1,2]

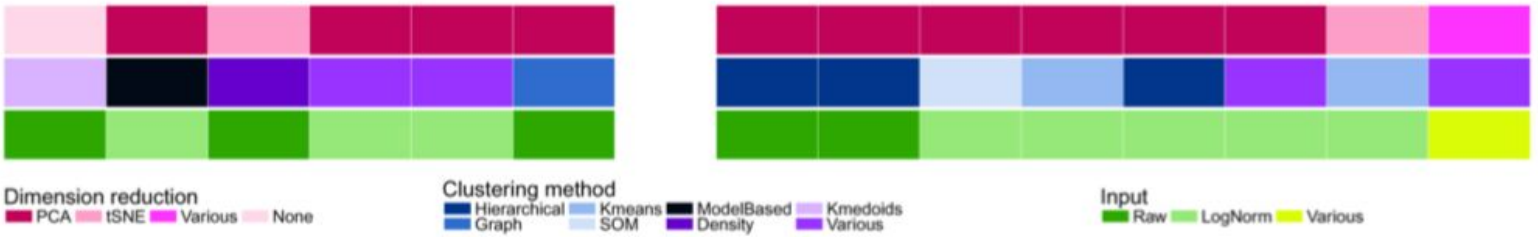
[1]Institute of Molecular Life Sciences, University of Zurich, Zurich, 8057, Switzerland
[2]SIB Swiss Institute of Bioinformatics, Zurich, 8057, Switzerland

NBIS NATIONAL BIOINFORMATICS INFRASTRUCTURE SWEDEN

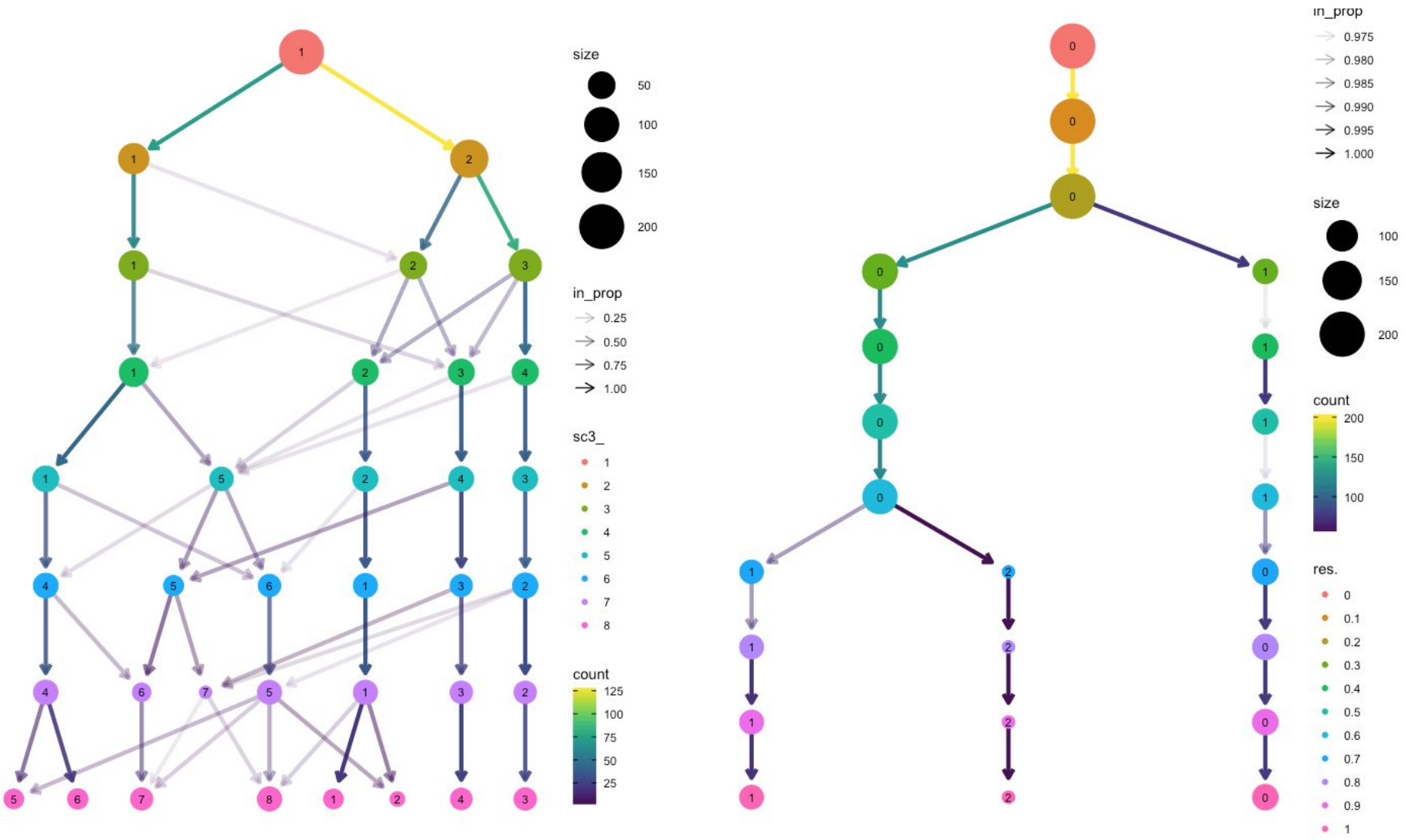
SciLifeLab


filteredExpr10
filteredHVG10
filteredM3Drop10
KumarTCC
Kumar
KohTCC
SimKumar4easy
Koh
SimKumar4hard
Zhengmix4uneq
Zhengmix4eq
SimKumar8hard
Zhengmix8eq
TrapnellTCC
Trapnell
RaceID2
FlowSOM
CIDR
TSCAN
ascend
PCAKmeans
PCAHC
SAFE
pcaReduce
RtsneKmeans
monocle
SC3svm
Seurat
SC3
Median ARI
1.00
0.75
0.50
0.25
0.00


1
0.53
0.33
0
0.14
0
0.36
0.14
0.11
0.14
0.14
0.28
0.19
0.08
RaceID2
TSCAN
monocle
SC3
SC3svm
Seurat
Truth
CIDR
ascend
FlowSOM
PCAKmeans
PCAHC
pcaReduce
RtsneKmeans
SAFE


Dimension reduction
PCA
tSNE
Various
None
Clustering method
Hierarchical
Kmeans
ModelBased
Kmedoids
Graph
SOM
Density
Various
Input
Raw
LogNorm
Various

Clustree – R package

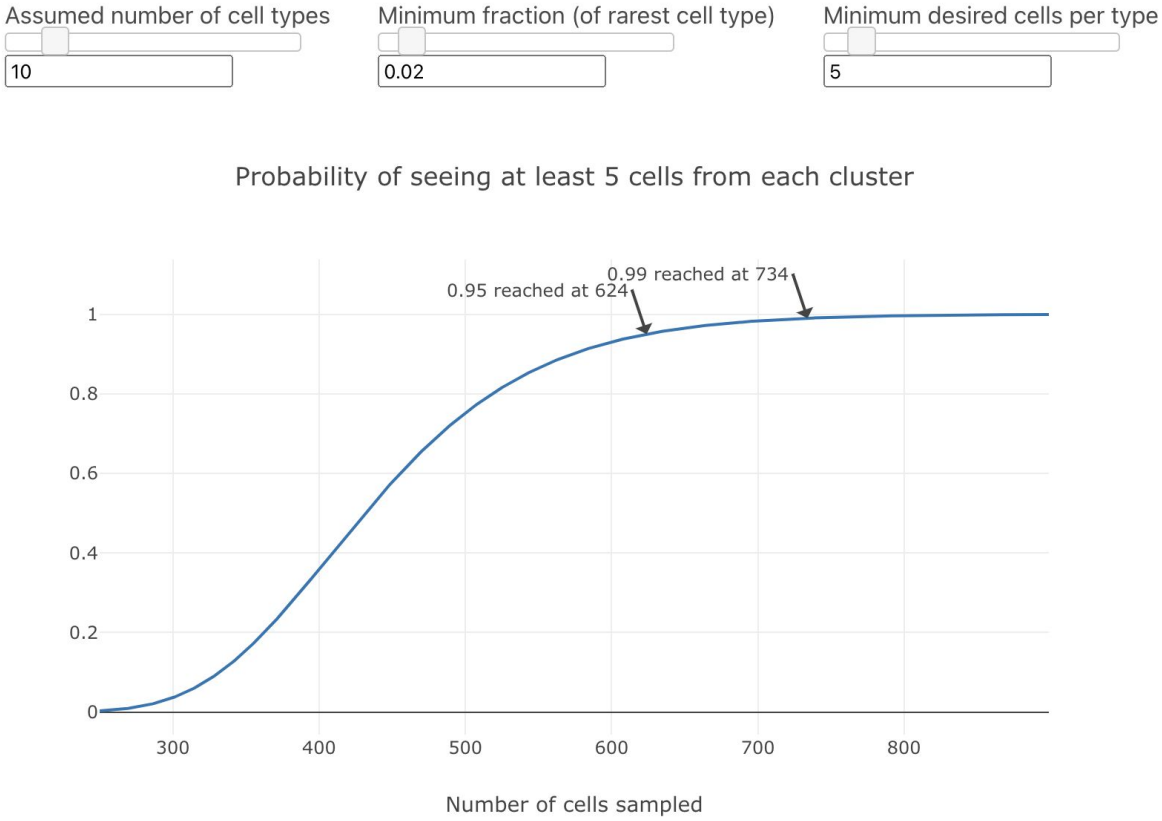


https://cran.r-project.org/web/packages/clustree/vignettes/clustree.html

How many clusters do you really have?

- It is hard to know when to stop clustering – you can always split the cells more times.
- Can use:
  - Do you get any/many significant DE genes from the next split?
  - Some tools have automated predictions for number of clusters – may not always be biologically relevant
- Always check back to QC-data – is what you are splitting mainly related to batches, qc-measures (especially detected genes)

How many cells are needed to find clusters?

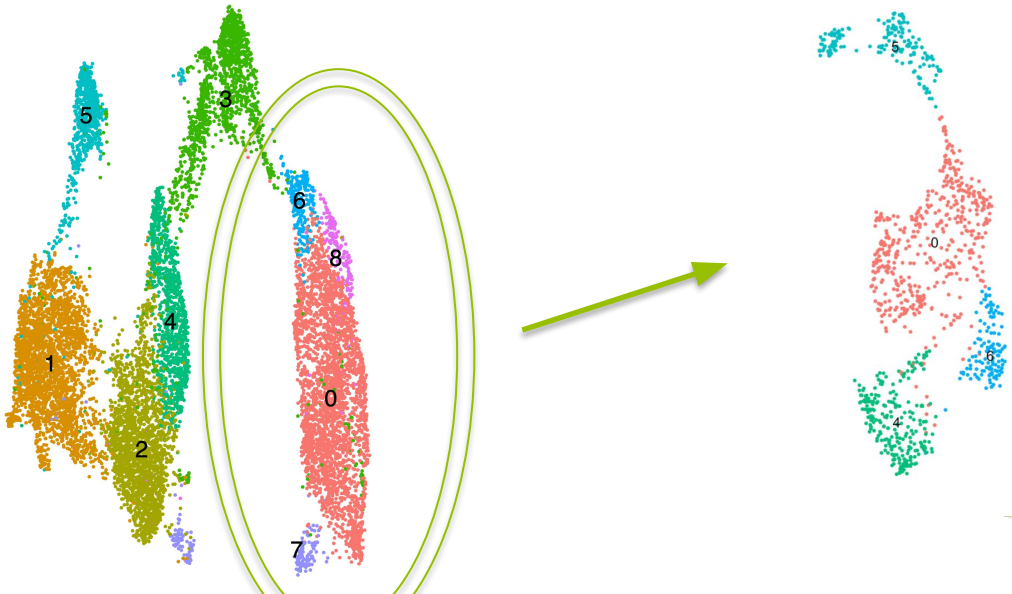


5 cells is probably low for detecting a distinct cluster and to do DEG detection – however 5 cells in 10 samples each can be enough to detect them.
Clearly distinct celltypes -> 5 cells can be enough for clustering.

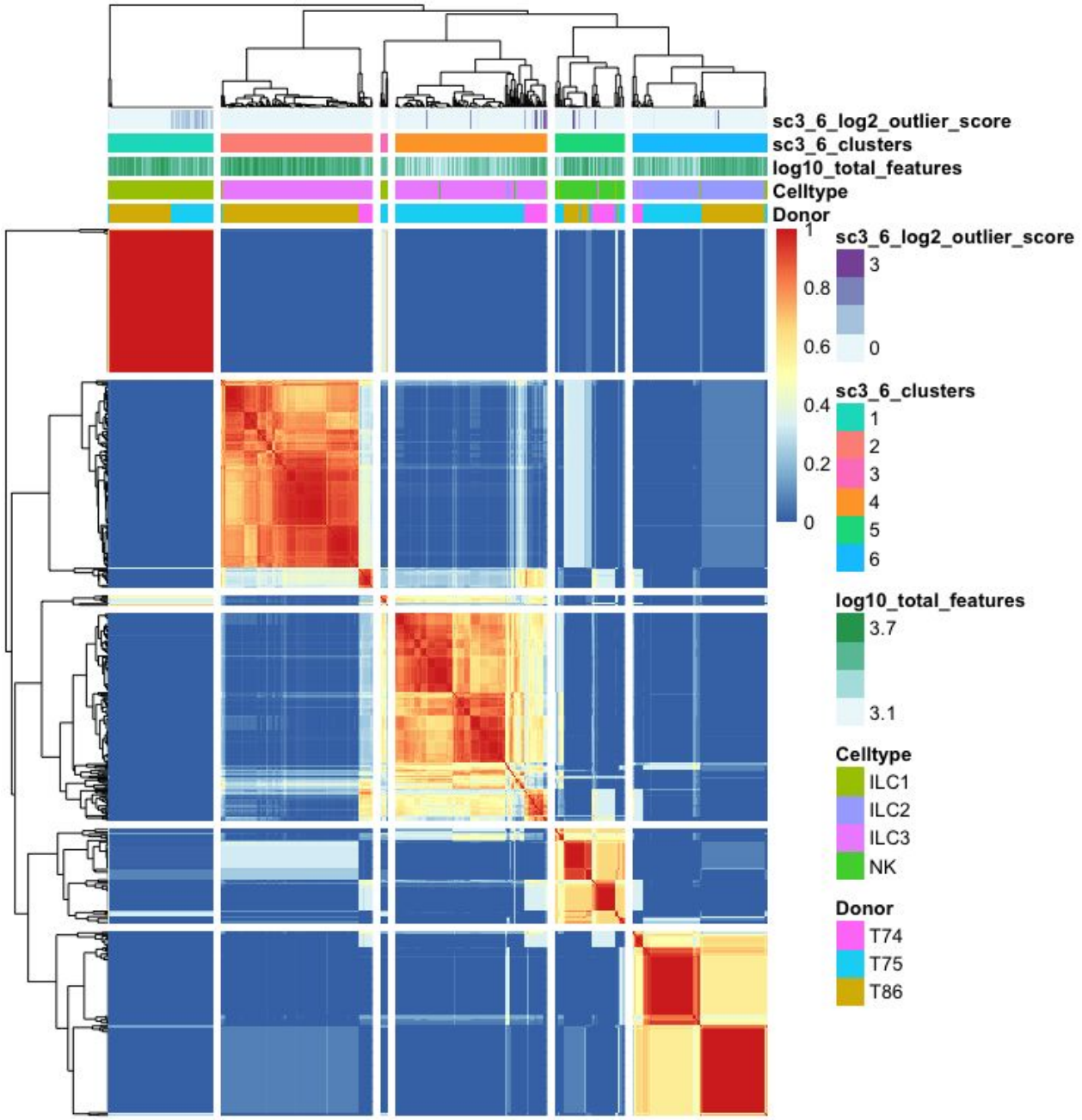
https://satijalab.org/howmanycells

Subclustering

- Most of the variation in a heterogeneous data set will be between broad celltypes.
- By selecting one celltype and rerunning HVG-selection and PCA – most of the variation will be differences between subtypes.



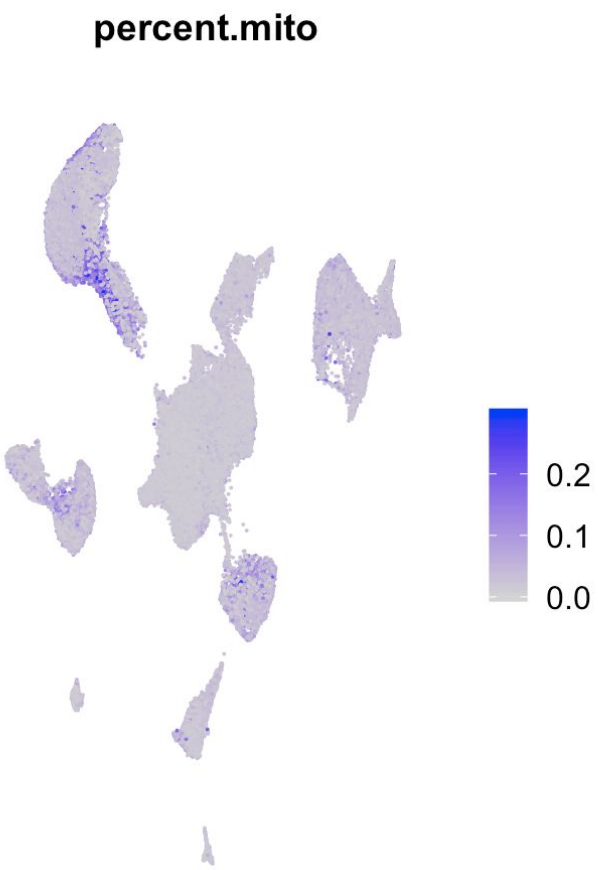
Check QC data

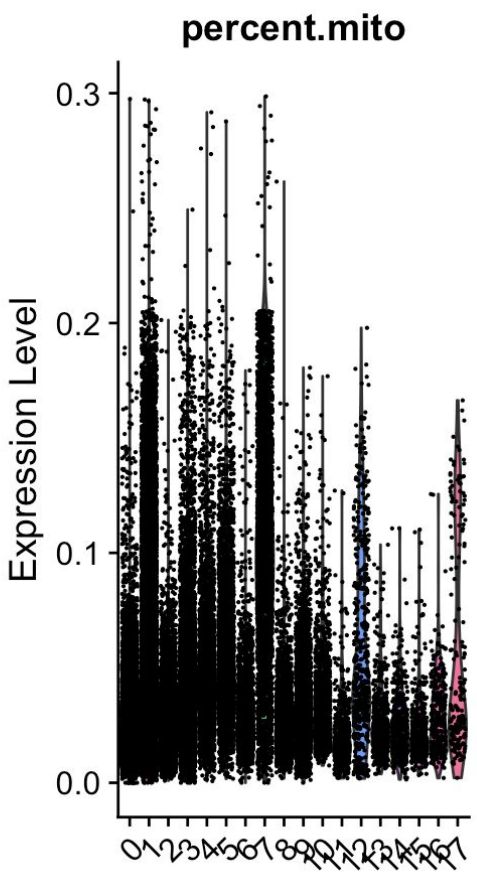


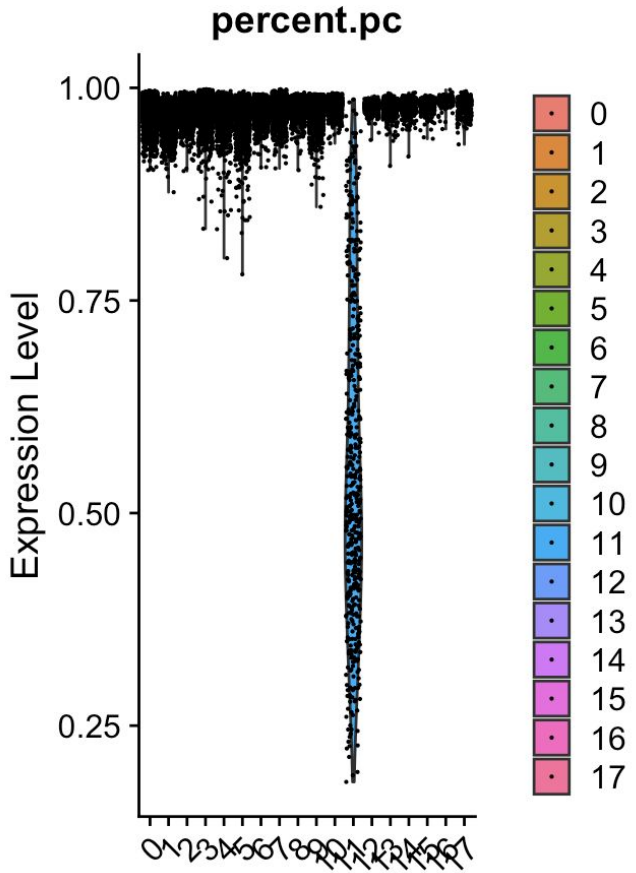
NBIS
NATIONAL BIOINFORMATICS INFRASTRUCTURE SWEDEN

SciLifeLab

Check QC data



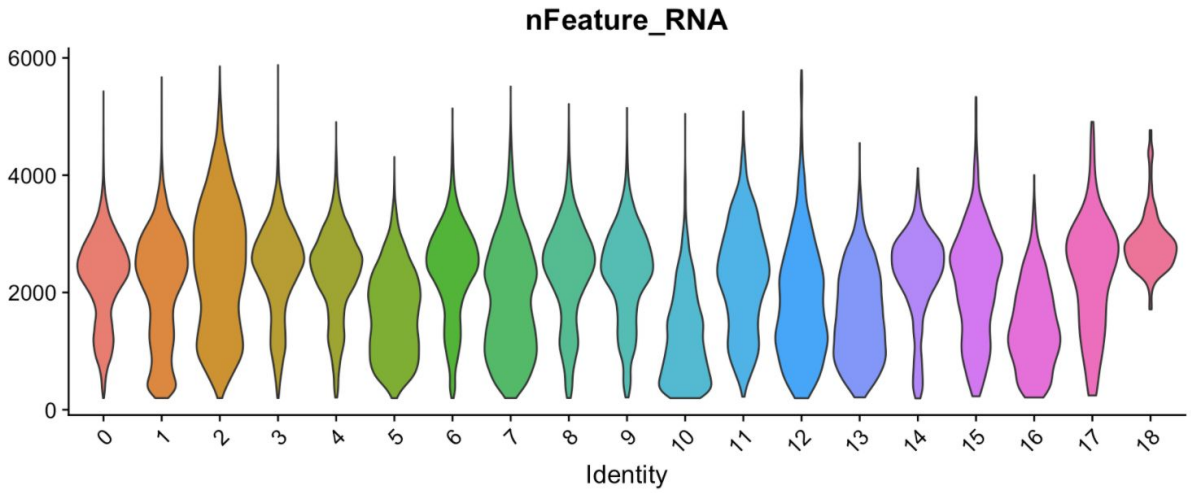


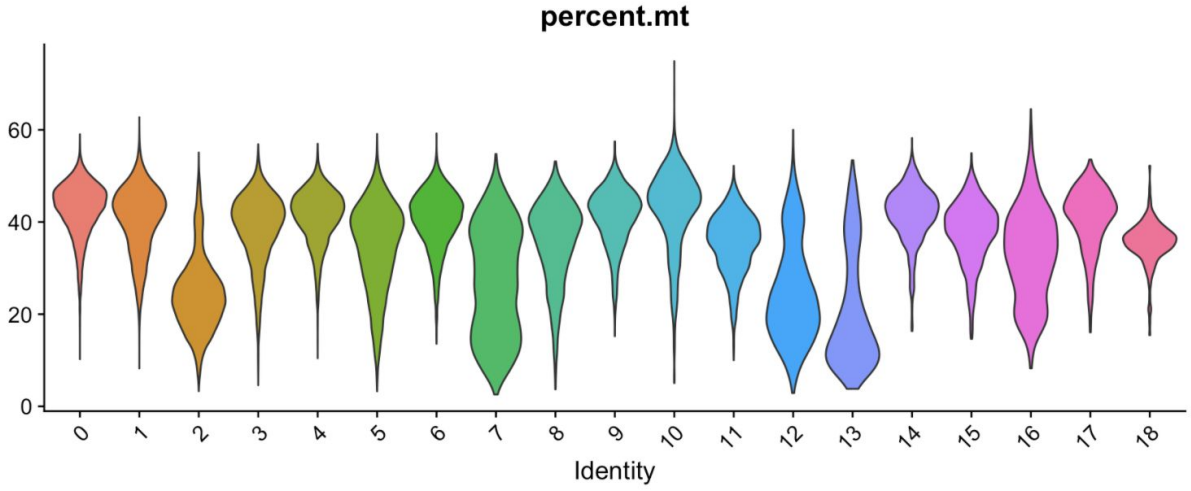


NBIS NATIONAL BIOINFORMATICS INFRASTRUCTURE SWEDEN

SciLifeLab

Check QC data

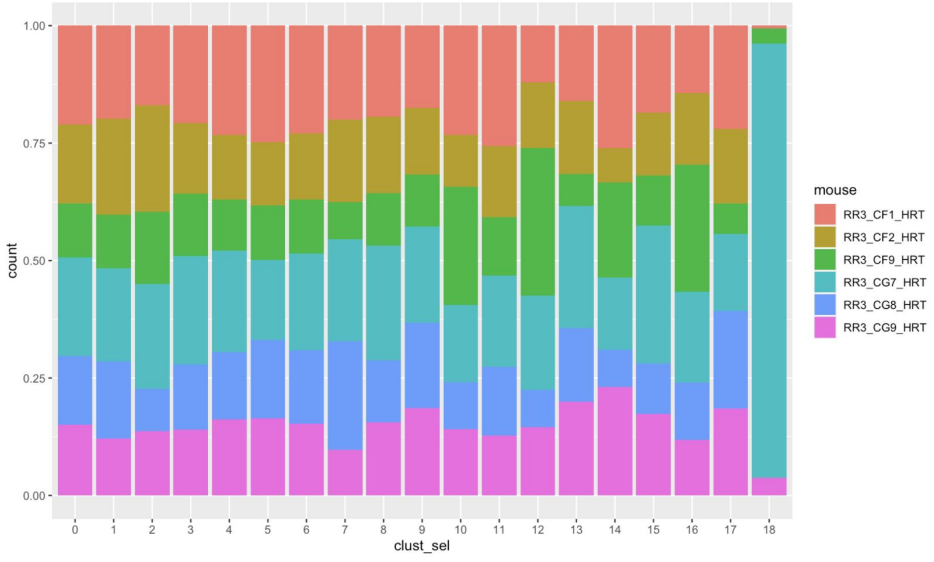


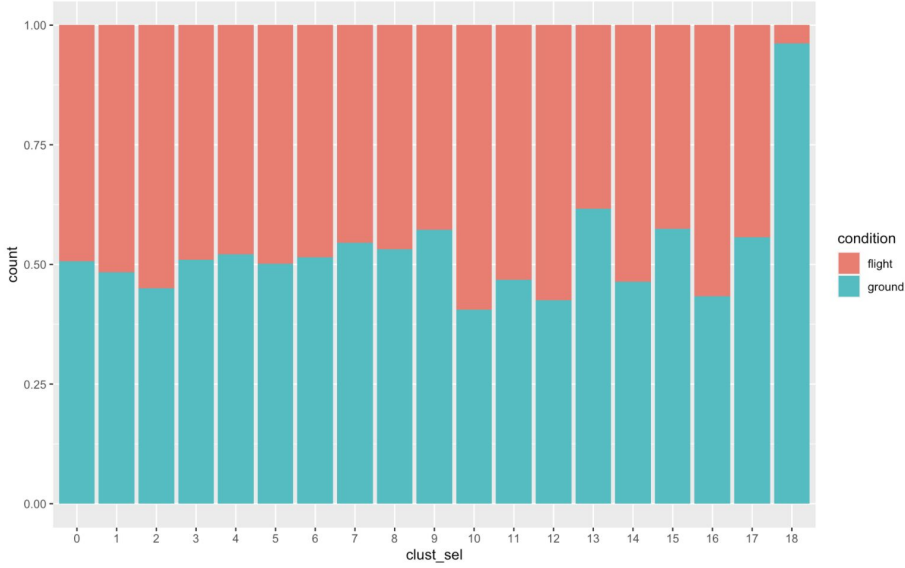


NBIS
NATIONAL BIOINFORMATICS INFRASTRUCTURE SWEDEN

SciLifeLab

Check batches / conditions





NBIS
NATIONAL BIOINFORMATICS INFRASTRUCTURE SWEDEN

SciLifeLab

Considerations for clustering

- Hypotheses:
  - What is a cell type? What cell types are in my tissue?
  - What is the number of clusters *k*?
- Choices:
  - Gene set selection
  - Similarity measure / Space to calculate similarity
  - Algorithm and hyper parameters of that algorithm.
- Different choice leads to different results. Validate, interpret and repeat steps.

Conclusions

- Clearly distinct celltypes will give similar results regardless of method
- Subclustering within celltypes may require careful selection of variable genes, dim reduction etc.
- Consistent results from different methods and agreement with UMAP layout is always best!
- Use your biological knowledge to evaluate the results – but try to be unbiased!