scRNAseq clustering tools

Åsa Björklund

asa.bjorklund@scilifelab.se

What is a celltype?

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

scRNA-seq analysis overview
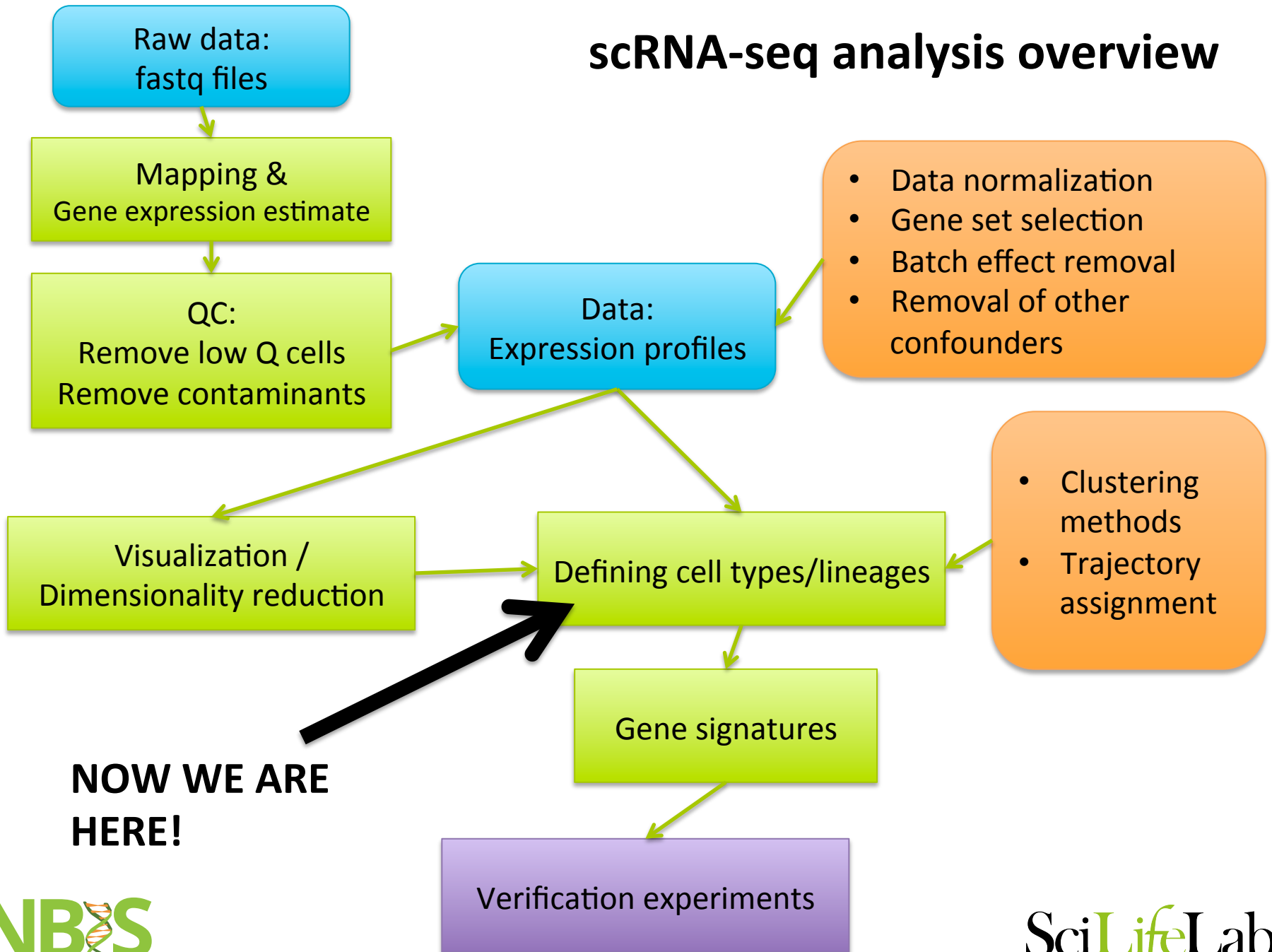
Outline

- Basic clustering theory

- Graph theory introduction

- Examples of different tools for clustering single cell data

- Other types of analyses on scRNAseq data

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
   - Graph based clustering

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
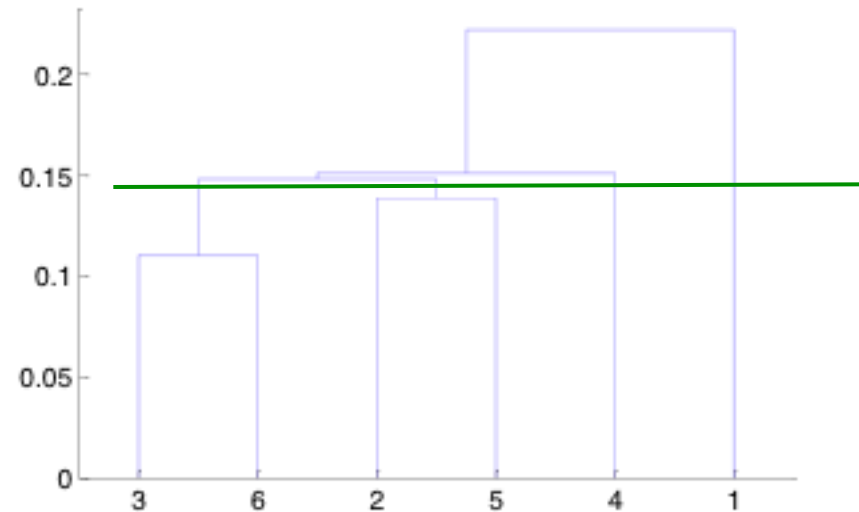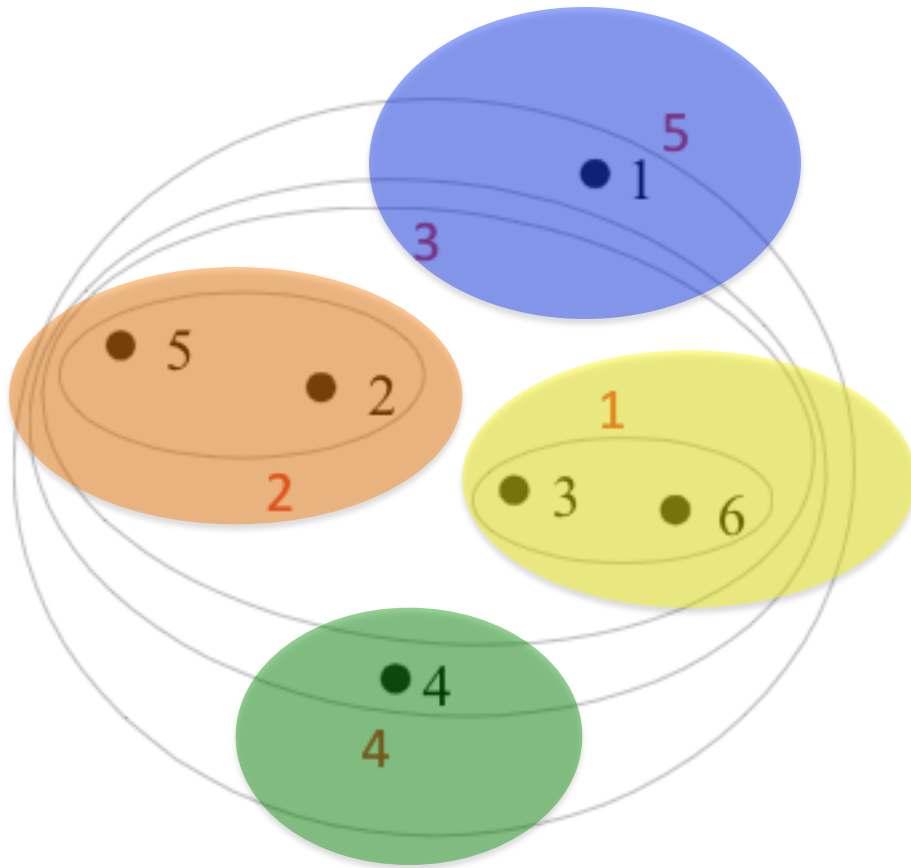
Hierarchical clustering
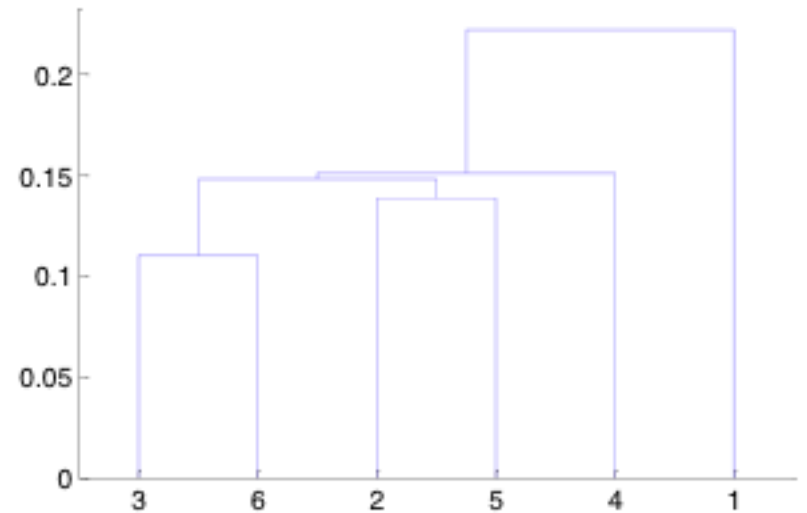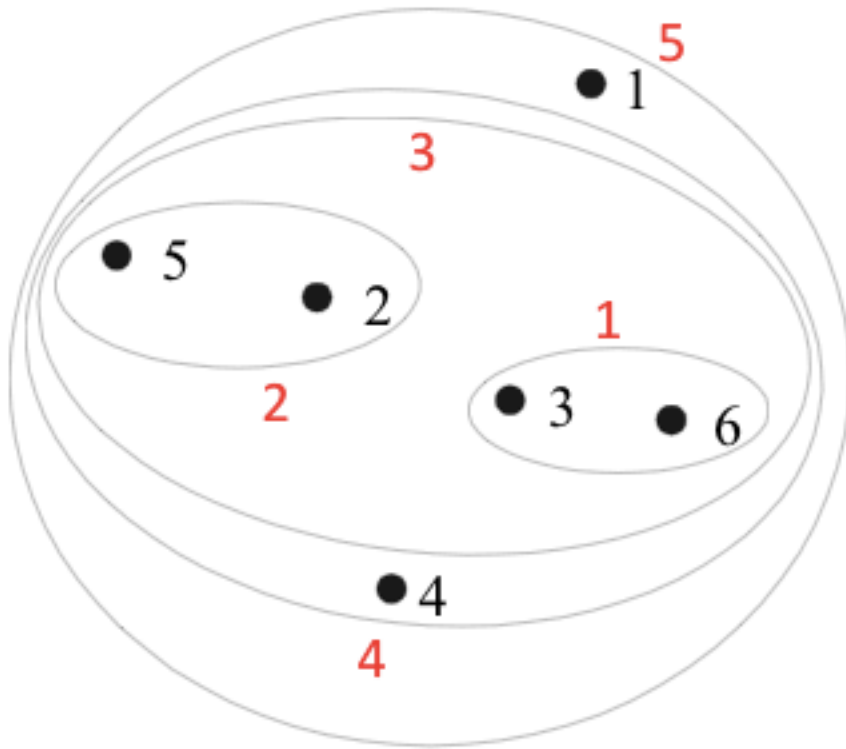
Hierarchical clustering
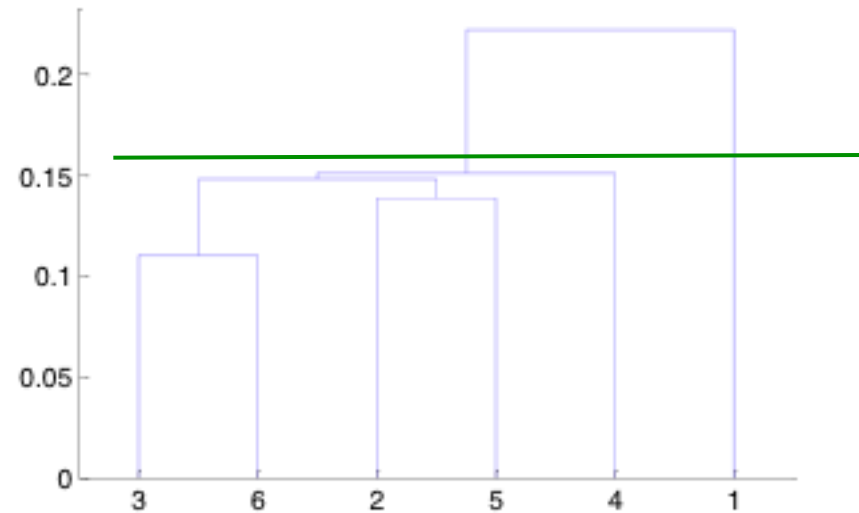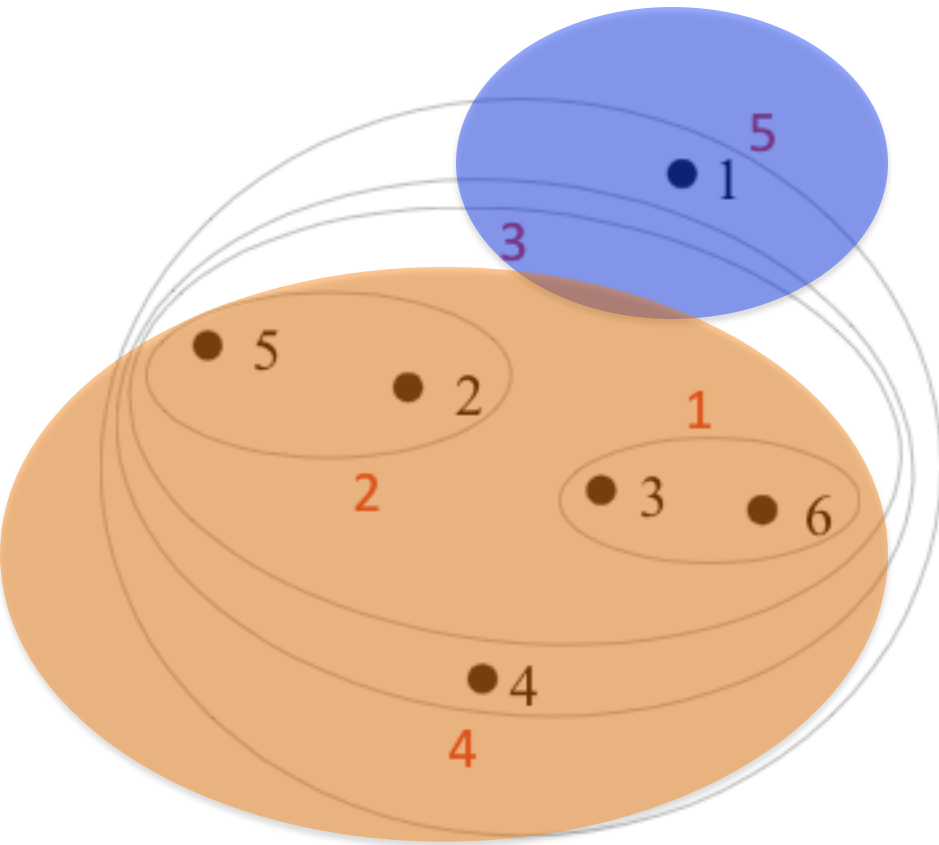
Clusters are obtained by cutting the tree at a desired level

Hierarchical clustering

Clusters are obtained by cutting the tree at a desired level

Different distance measures

- ## Most commonly used in scRNA-seq:
  - Euclidean distance
    - In multidimensional space
    - In PCA/tSNE or other reduced space
  - Inverted pairwise correlations (1-correlation)
- ## Others include:
  - Manhattan distance
  - Mahalanobis distance
  - Maximum distance

Linkage criteria

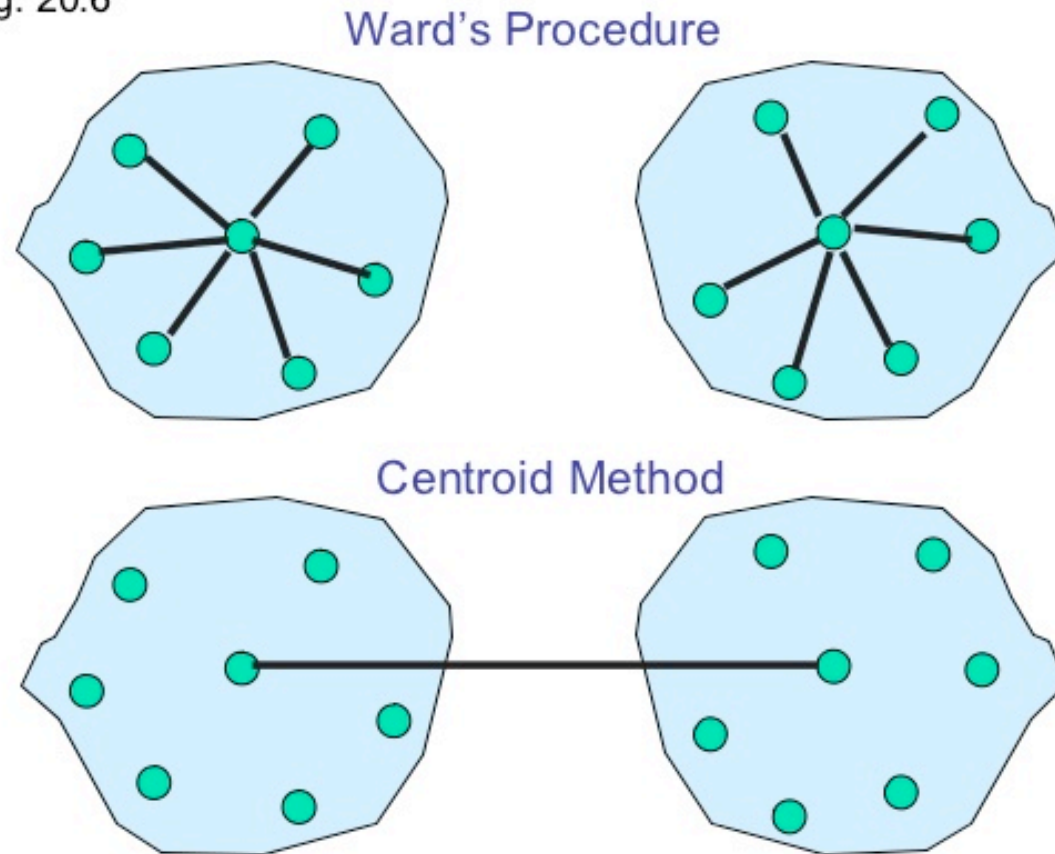- Calculation of similarities between 2 clusters (or a cluster and a data point)
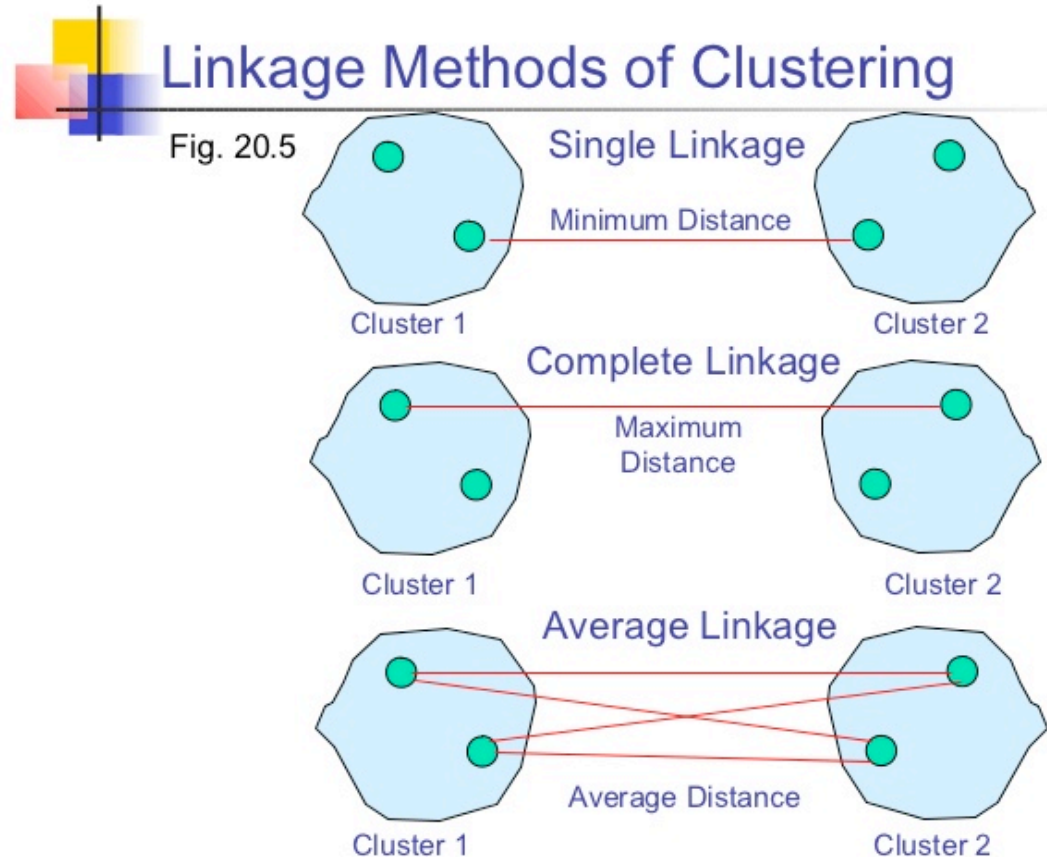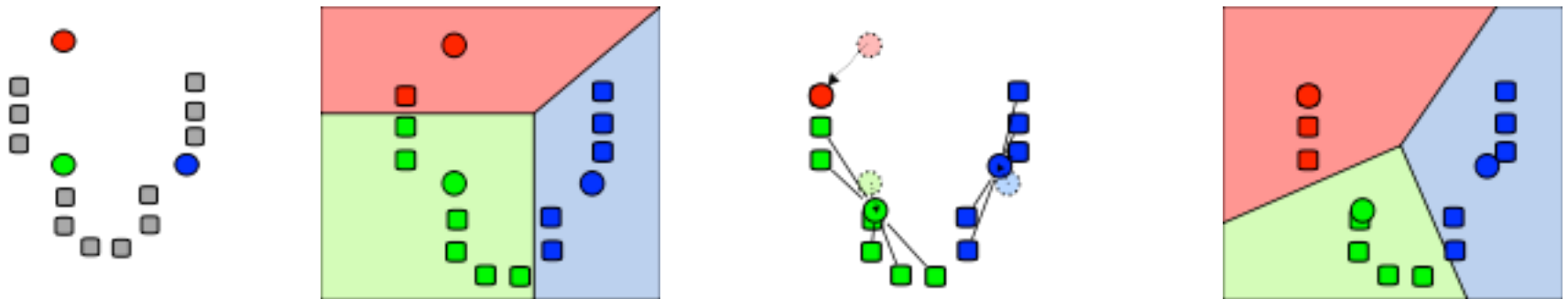
Other Agglomerative Clustering Methods

Fig. 20.6
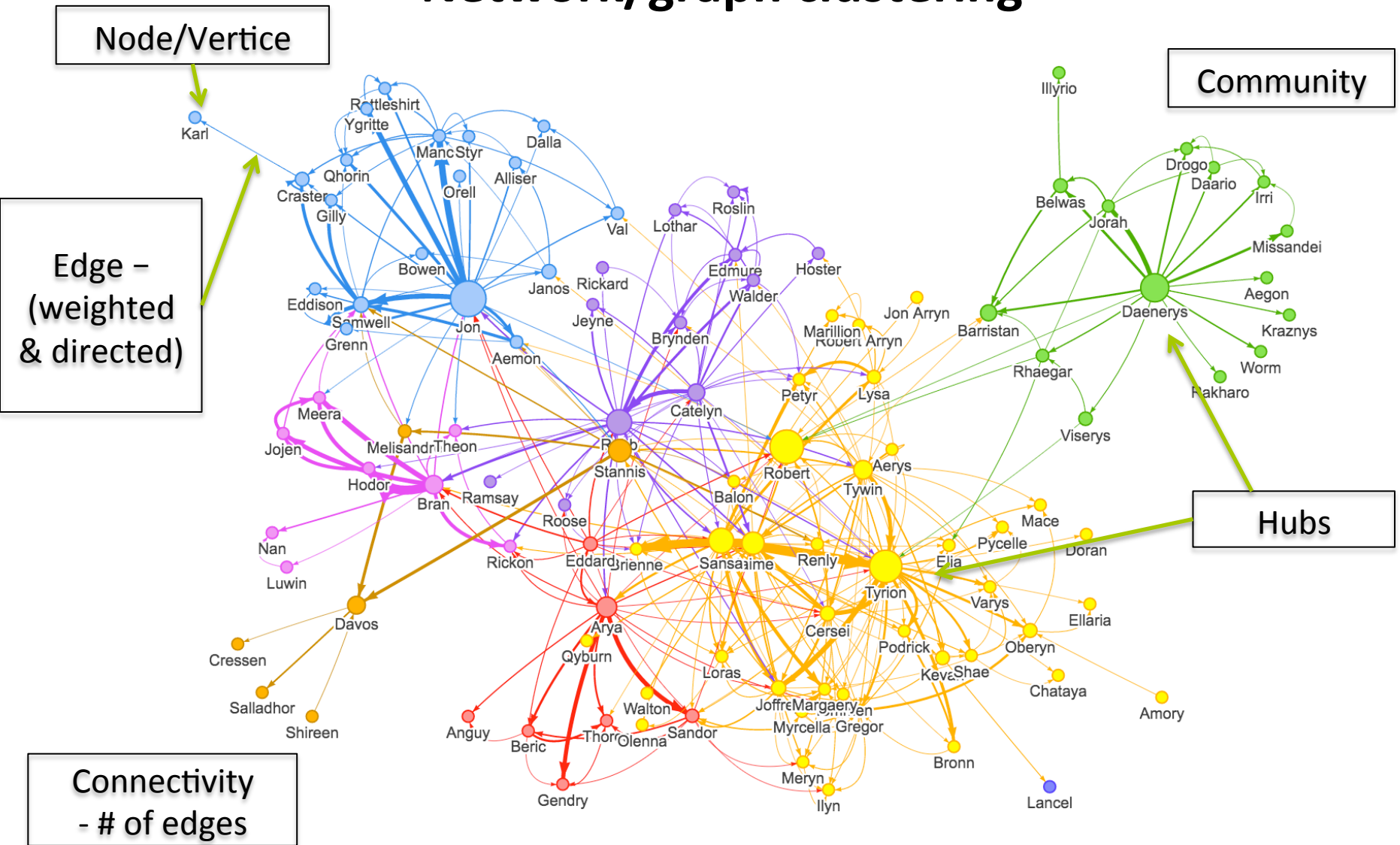
Ward's Procedure

Centroid Method

- 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

Node/Vertice

Community

Edge – (weighted & directed)

Hubs

Connectivity - # of edges

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

Network/graph clustering

Node ID	1	2	3	4	5	6	7	8	9	10
1	0	1	1	0	0	0	0	0	0	1
2	1	0	1	0	0	0	0	0	0	0
3	1	1	0	0	0	0	0	0	0	0
4	0	0	0	0	1	1	0	0	0	1
5	0	0	0	1	0	1	0	0	0	0
6	0	0	0	1	1	0	0	0	0	0
7	0	0	0	0	0	0	0	1	1	1
8	0	0	0	0	0	0	1	0	1	0
9	0	0	0	0	0	0	1	1	0	0
10	1	0	0	1	0	0	1	0	0	0

Fig 2. Network partitions that maximize Q. Maximum Q=0.4896

Adjacency matrix
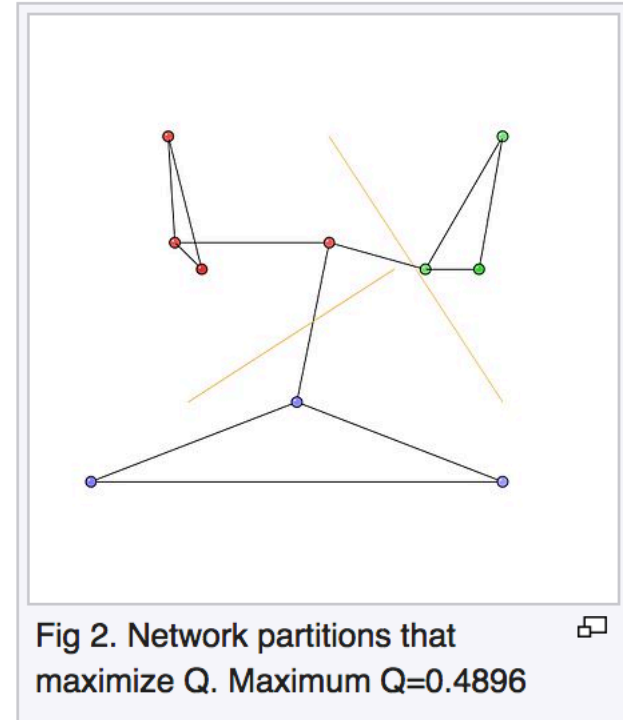
Types of graphs

- 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.
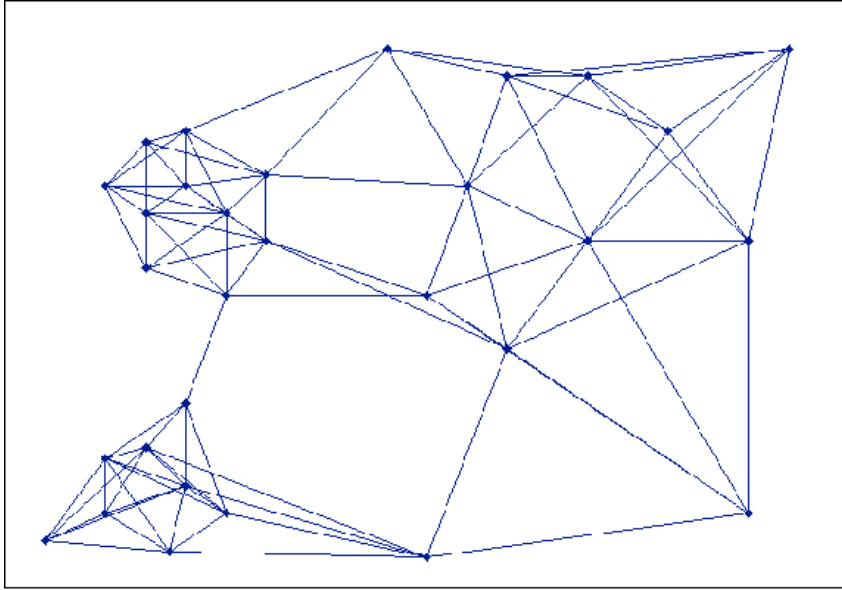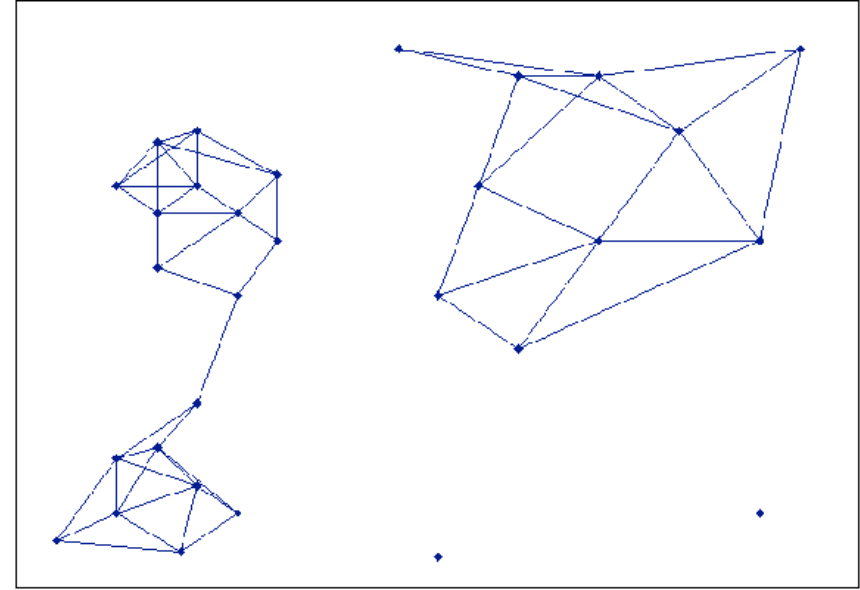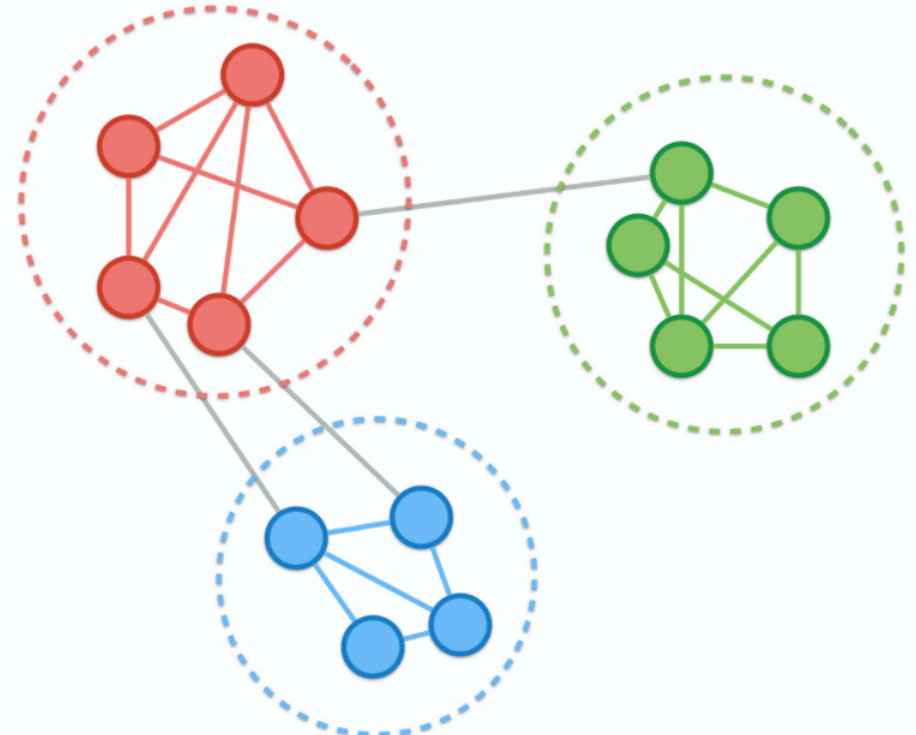
SNN graph

Figure 2. Near Neighbor Graph

Figure 3. Unweighted Shared Near Neighbor Graph

(Ertöz et al. Semantic scholar, 2002)

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.
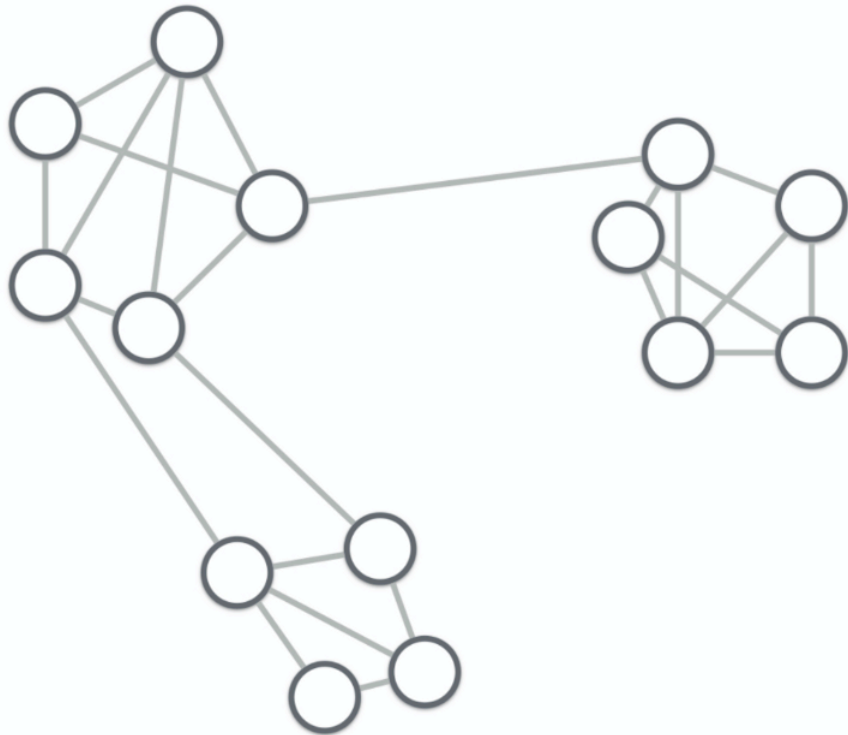
Community detection

- 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.

- Many implemented algorithms to this problem:
  - Different methods of Modularity optimization
  - Infomap
  - Walktrap
  - etc.

- Most methods will automatically define the number of clusters based on some user parameters.

For single cell data

- Can start with distances based on correlation, euklidean distances in PCA space etc. Same as for hclust/k-means.

- Buld 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 reduce network based on shared neighbors.

- Find clusters with community detection method.

- Graphs can also be used for trajectory analysis

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

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

(Rosvall et al. *Plos One* 2010)
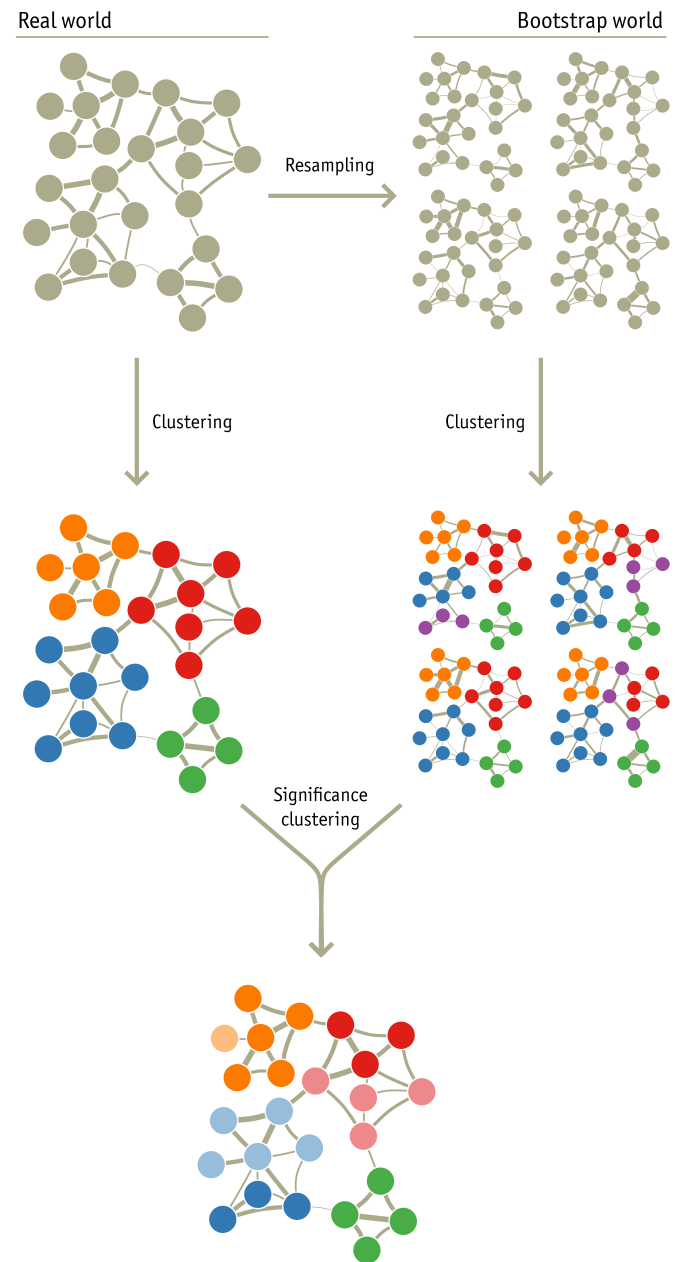
scRNAseq clustering

- Easy case with distinct celltypes:
  - rpkms/counts – Euklidean or correlation distances
  - PCA, tSNE or other dimensionality reduction method
- Examples of programs for clustering (many more out there):
  - WGCNA
  - BackSPIN
  - Pagoda
  - SC3
  - Seurat
  - pcaReduce
  - SNNcliq

SCRAN – Single Cell RNA ANalisys

- Uses SingleCellExperiment class – same as in Scater package

- Includes cyclone method for predicting cell cycle phase.

- Includes Basics deconvolution strategy for size factors.

- Detection of variable genes by deconvolution of technical and biological variance.

http://bioconductor.org/packages/devel/bioc/vignettes/scran/inst/doc/scran.html

Single Cell Consensus Clustering – SC3

(Kiselev et al *Nat. Methods* 2017)
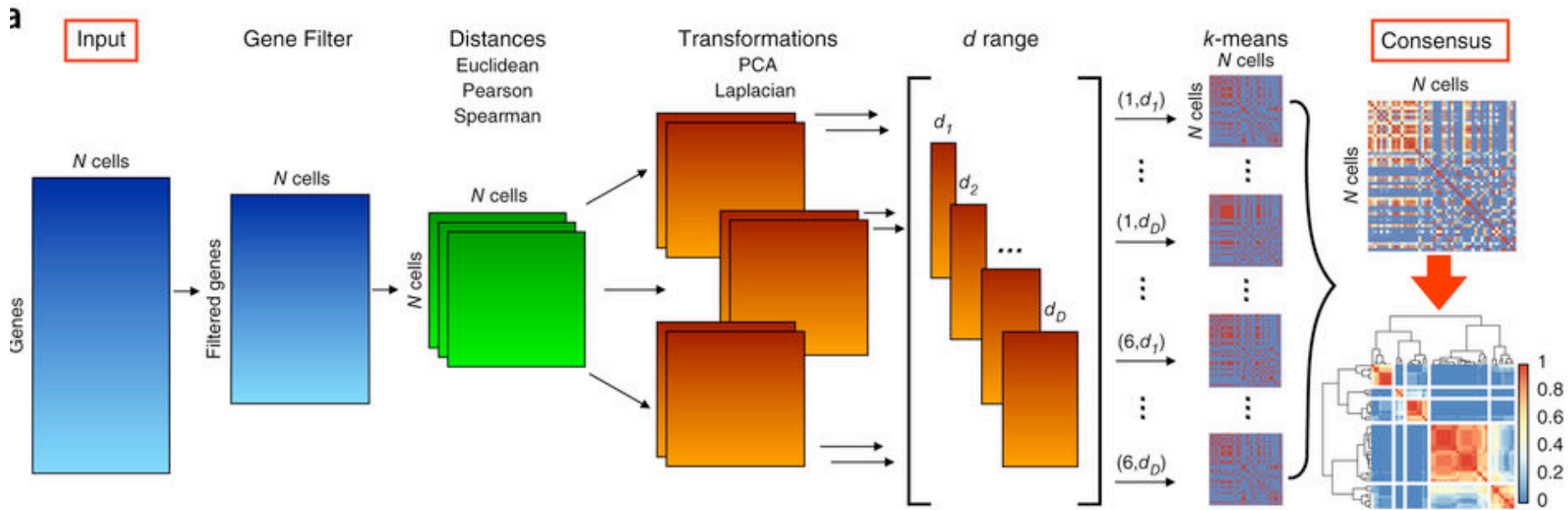
Single Cell Consensus Clustering – SC3

1. Gene filtering – rare and ubiquitous genes

2. Distance matrices (DM) – Euklidean, Spearman, Pearson

3. Transformation of DM with PCA or Laplacian

4. K-means clustering with first d eigenvectors

5. Consensus clustering – distance 1/0 for cells in same/ different clusters -> hierarchical clustering on average distances.

Differential expression with nonparametric Kruskal–Wallis test.

Marker genes with areas under the ROC curve (AUROC) from 100 permutations of cell cluster labels and P-values from Wilcoxon signed-rank test.

(Kiselev et al *Nat. Methods* 2017)

Pagoda – Pathway And Geneset OverDispersion Analysis

Implemented in the SCDE package

(Fan et al. *Nature Methods* 2016)

Pagoda – Pathway And Geneset OverDispersion Analysis

Figure 3
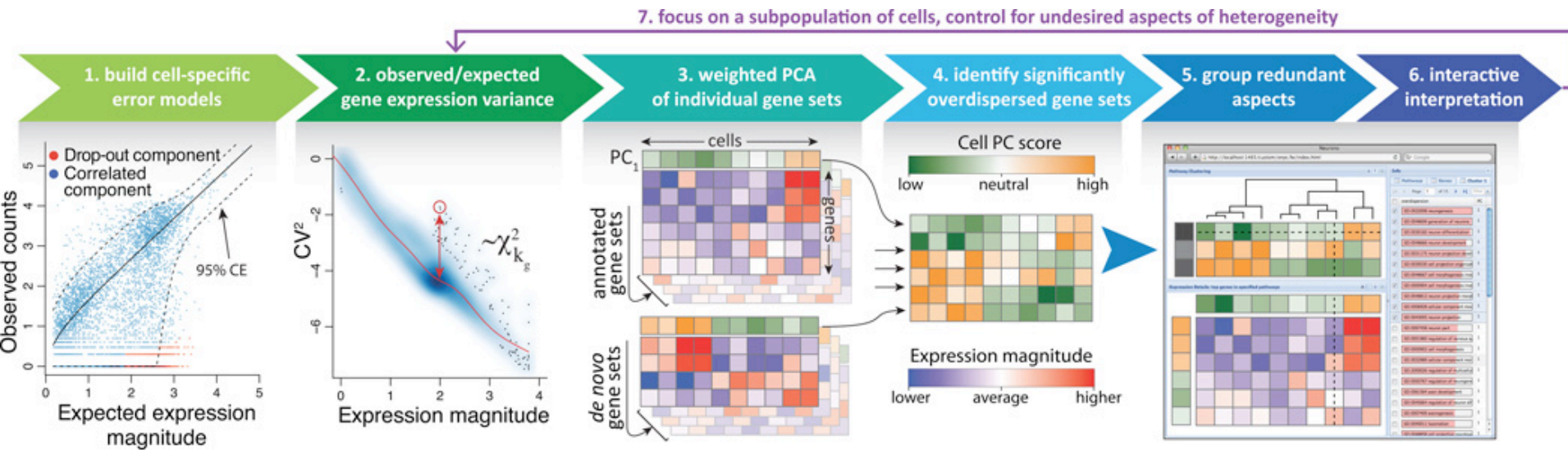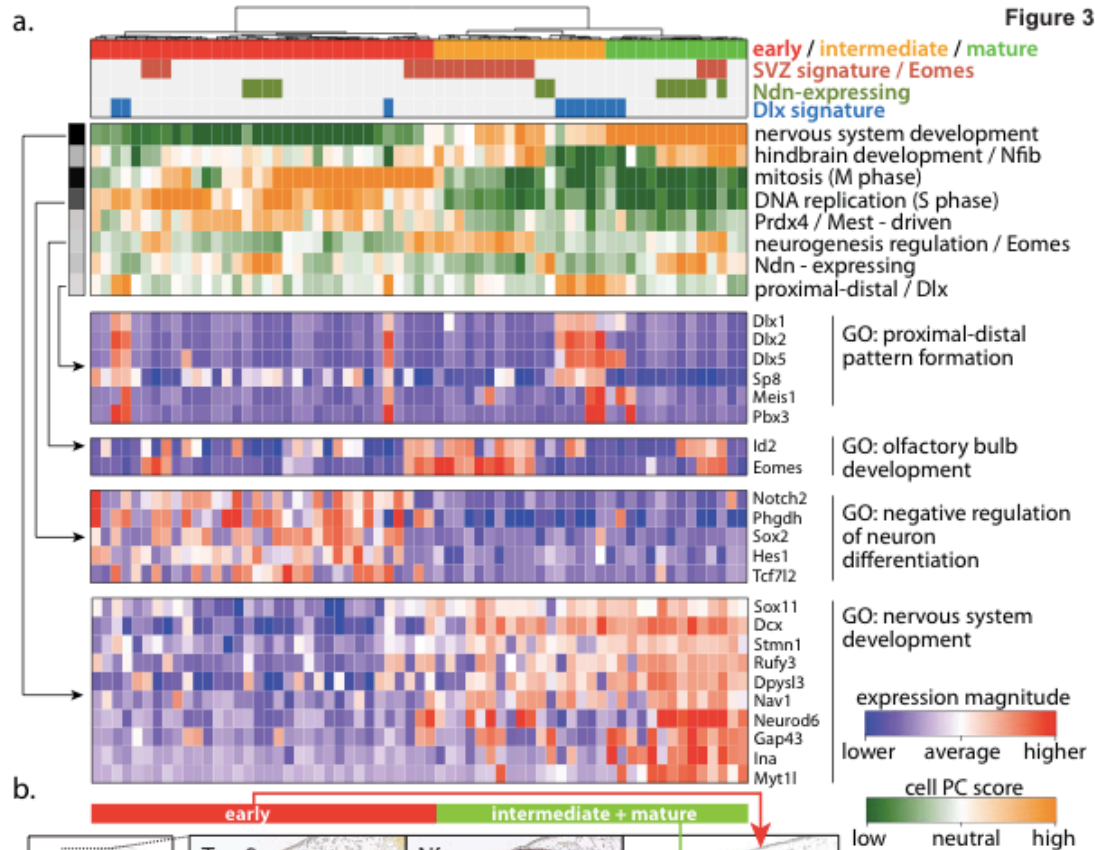
- Helps with biological interpretation of data
- Important to have good and relevant gene sets
- High memory consumption when running Pagoda
- Also has methods for removing batch effect, detected genes, cell cycle etc

(Fan et al. *Nature Methods* 2016)

Pagoda2

- Similar error modelling
- Now include KNN graph clustering
- Can visualize gene sets.
- https://github.com/hms-dbmi/pagoda2

BackSPIN - Biclustering

- Simultaneous clustering genes and cells.
- An iterative, biclustering method based on sorting points into neighborhoods (SPIN) to find shapes in a reduced space
  1. ordering of samples using genes as features,
  2. ordering of genes using samples as features and
  3. zooming in on subsets of the original expression matrix to order objects in a reduced subspace.
- Clusters both genes and cells to identify subpopulations as well as potential markers for each subpopulations.
- Implemented in Python.

(Zeisel et al. *Science* 2015)

Shared nearest neighbor (SNN)-Cliq

- Similarity matrix using Euclidean distance (can use other distances)

- List the k-nearest-neighbors (KNN)

- Edge between cells if at least one shared neighbor

- Weights based on ranking of the neighbors

- Graph partition by finding cliques

- Identify clusters in the SNN graph by iteratively combining significantly overlapping subgraphs

- Implemented in Matlab and Python

(Xu et al *Bioinformatics* 2015)

Seurat

- Developed for drop-seq analysis – compatible with 10X output files. But works also for other types of data.

- Contains function for
  - Data normalization
  - Detection of variable genes
  - Regression of batch effects and other confounders
  - JackStraw to detect significant principal components
  - tSNE and other dimensionality reduction techniques
  - Clustering based on SNN graphs
  - Many different methods for Differential expression

(http://satijalab.org/seurat/)

Seurat - FindClusters

- First construct a KNN (k-nearest neighbor) graph based on the euclidean distance in PCA space.
  - Select which principal components to include
- Refine the edge weights between any two cells based on the shared overlap in their local neighborhoods (Jaccard distance).
- To cluster the cells, modularity optimization techniques to iteratively group cells together.
- **OBS!** Earlier versions of Seurat uses "spectral tSNE" and DBScan density clustering.

Seurat

- Also contains functions for:
  - Spatial reconstruction of single cell data using *in situ* references (Zebrafish embryos)
  - Integrated analysis across platforms
  - Analysis of multimodal datasets (e.g. RNA + protein)

(http://satijalab.org/seurat/)

Loupe – Cell Browser, from 10X Genomics
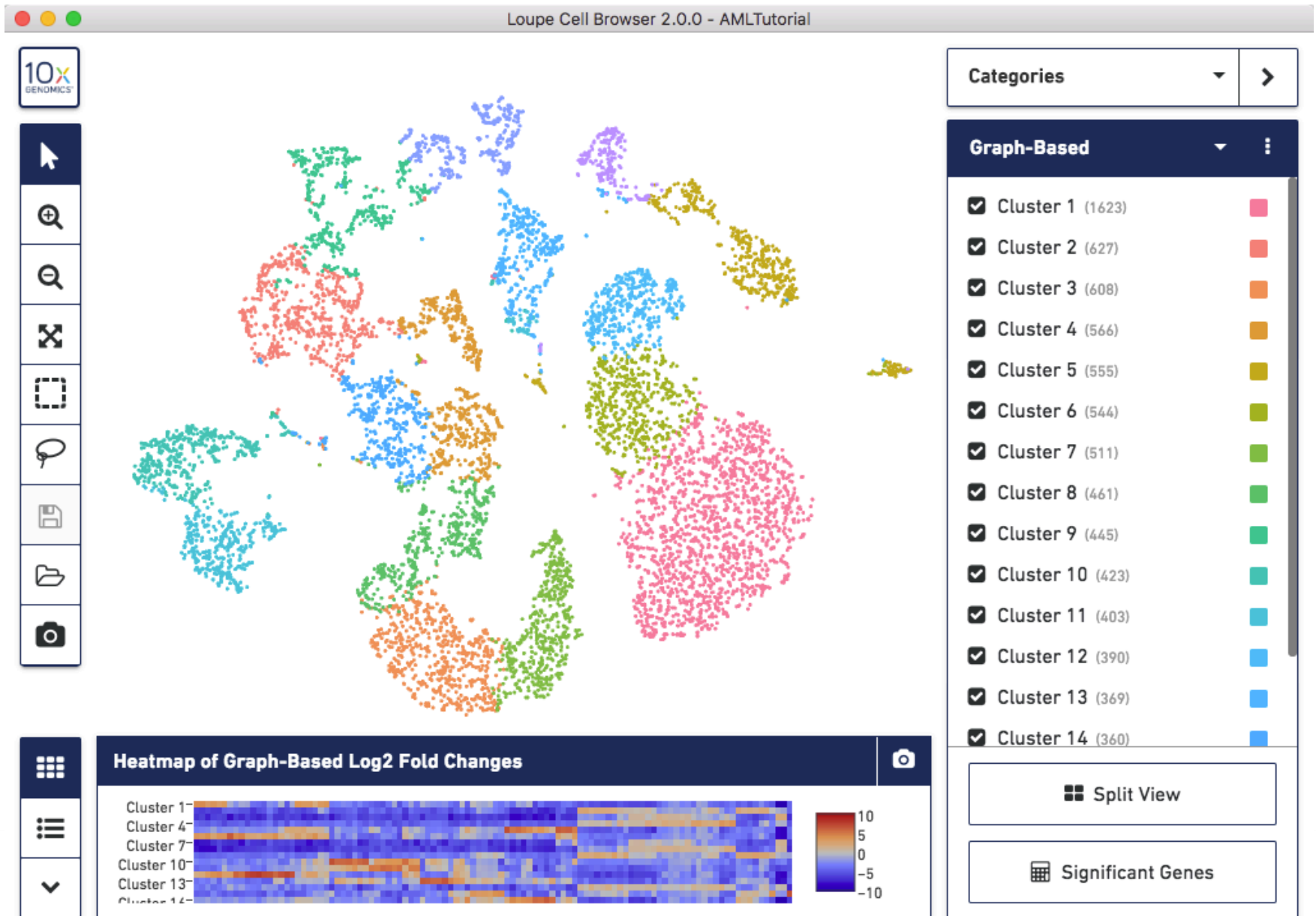
Which clustering method is best?

- 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 euklidean or correlation distances will work fine.

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

Check QC data
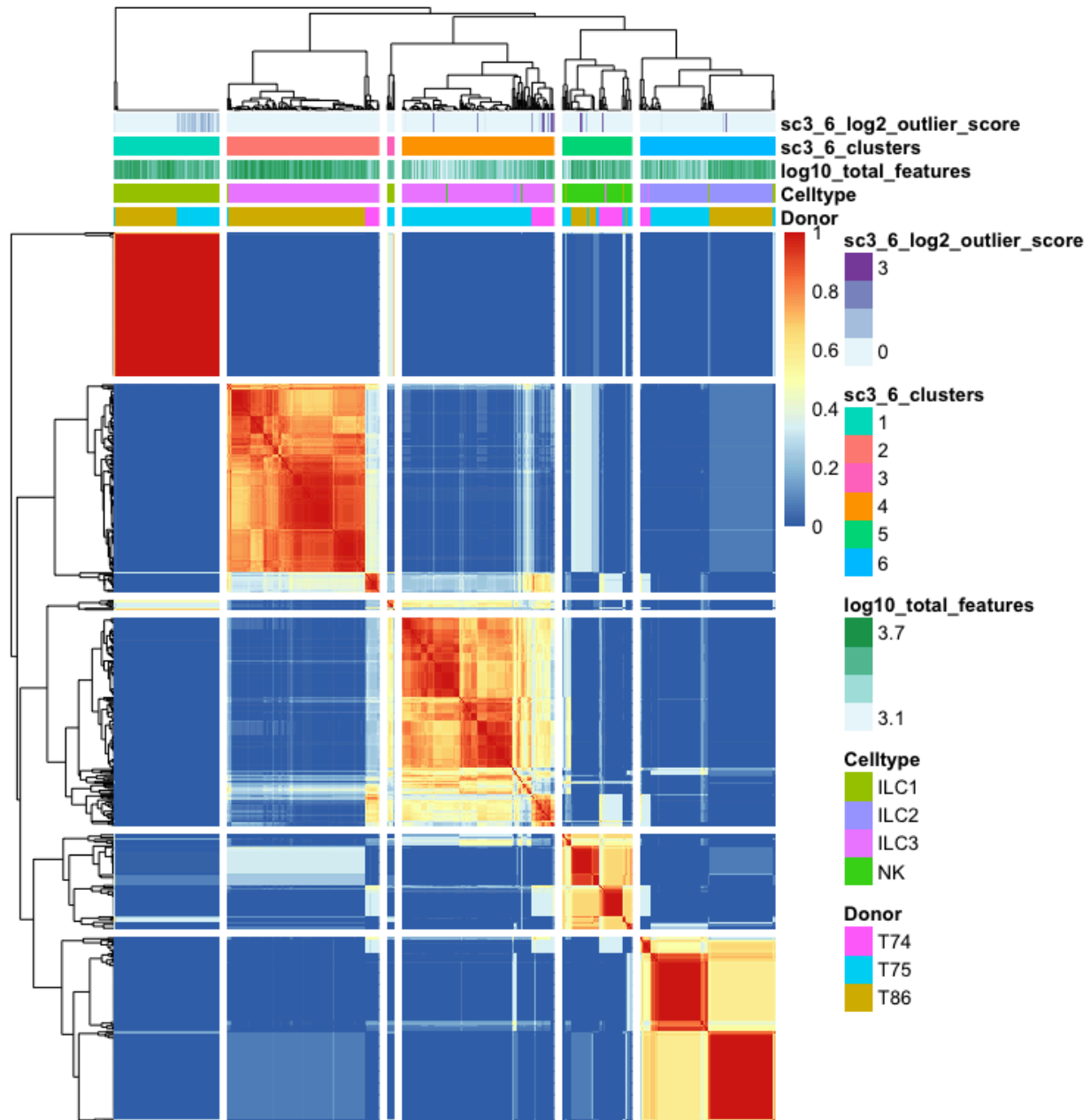
Check QC data
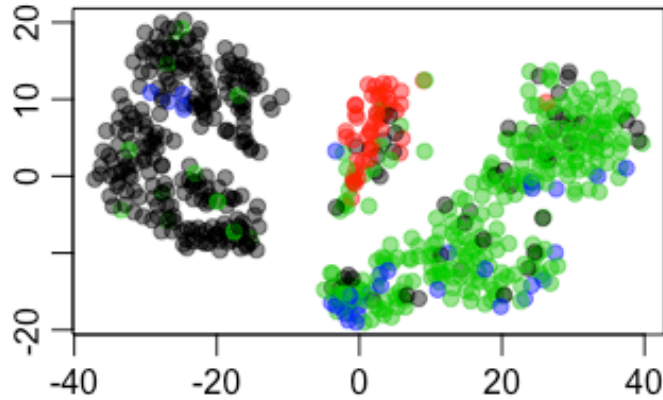
Large scale analysis

(Svensson *et al. Nature Protocols* 2018)

Large scale analysis

(SCANPY – Wolf *et al.* Genome Biology 2018)
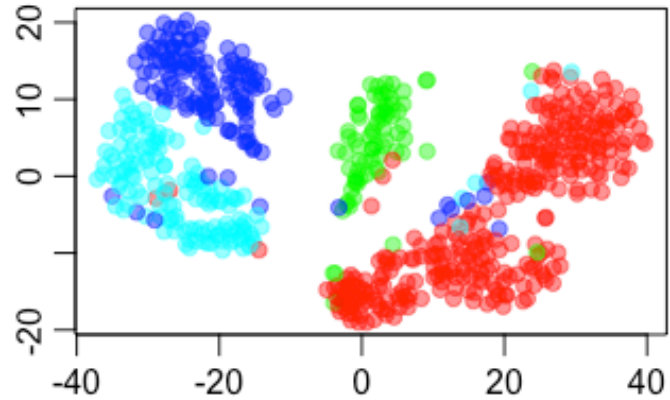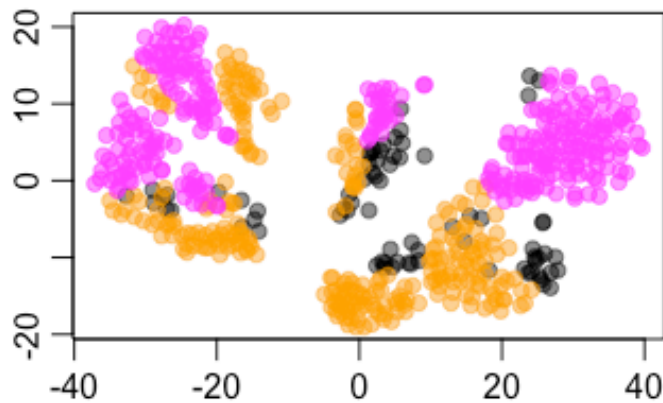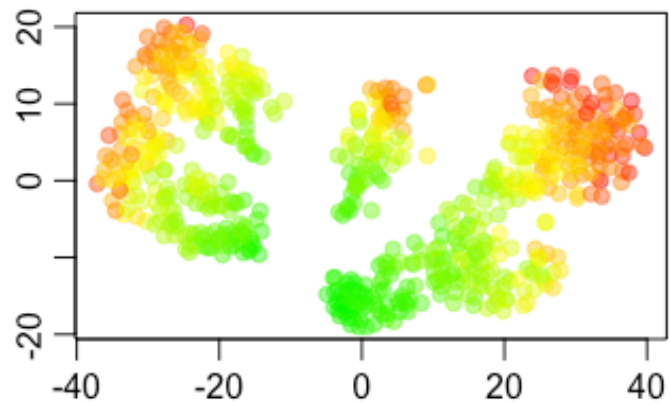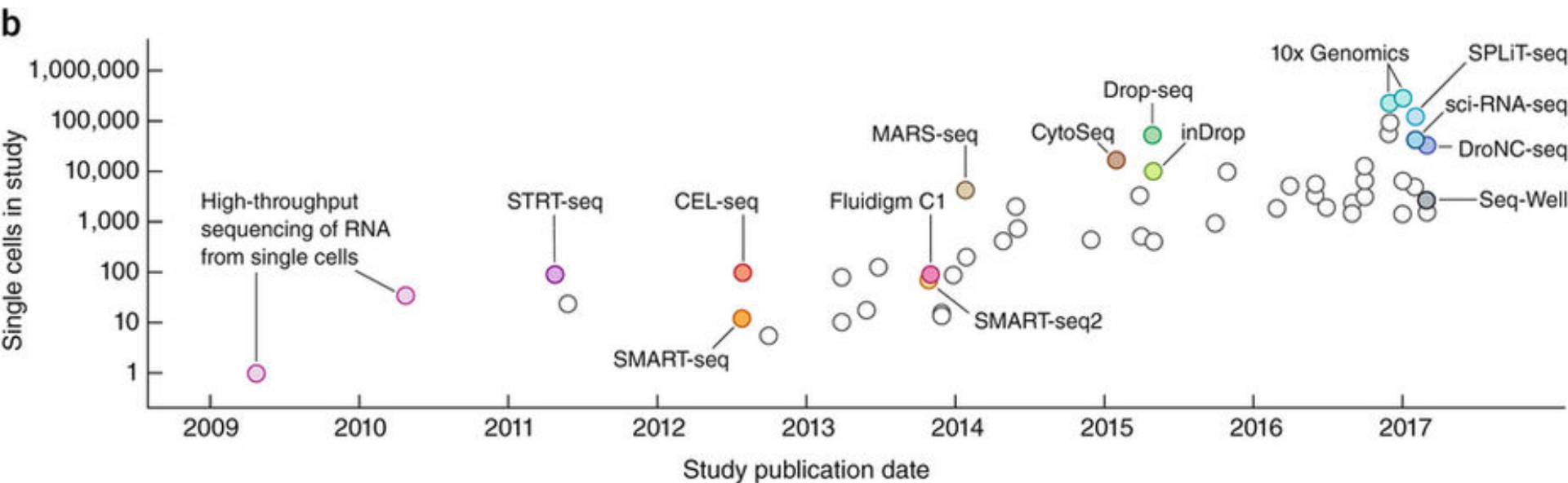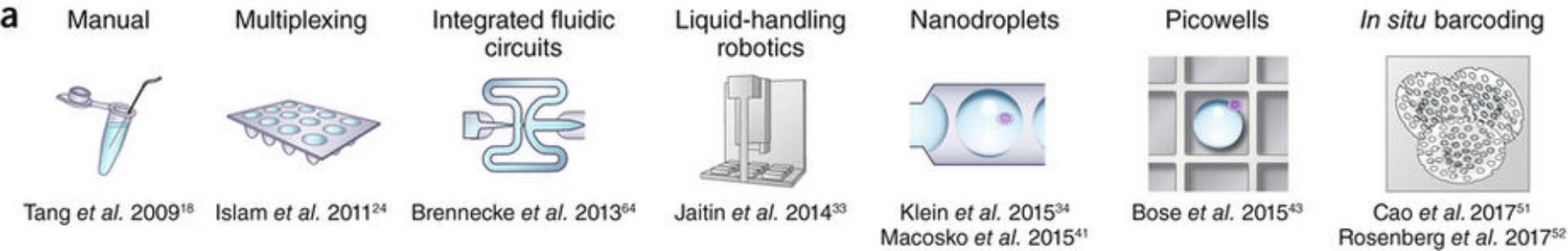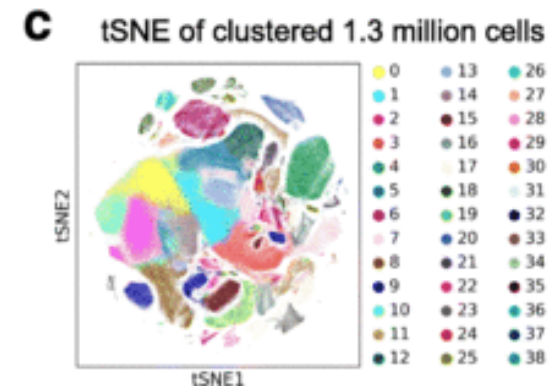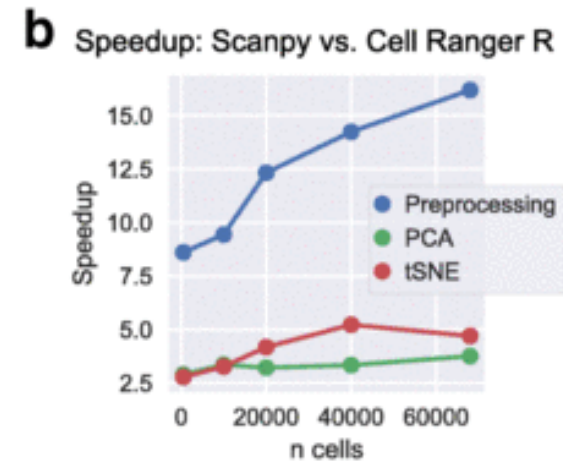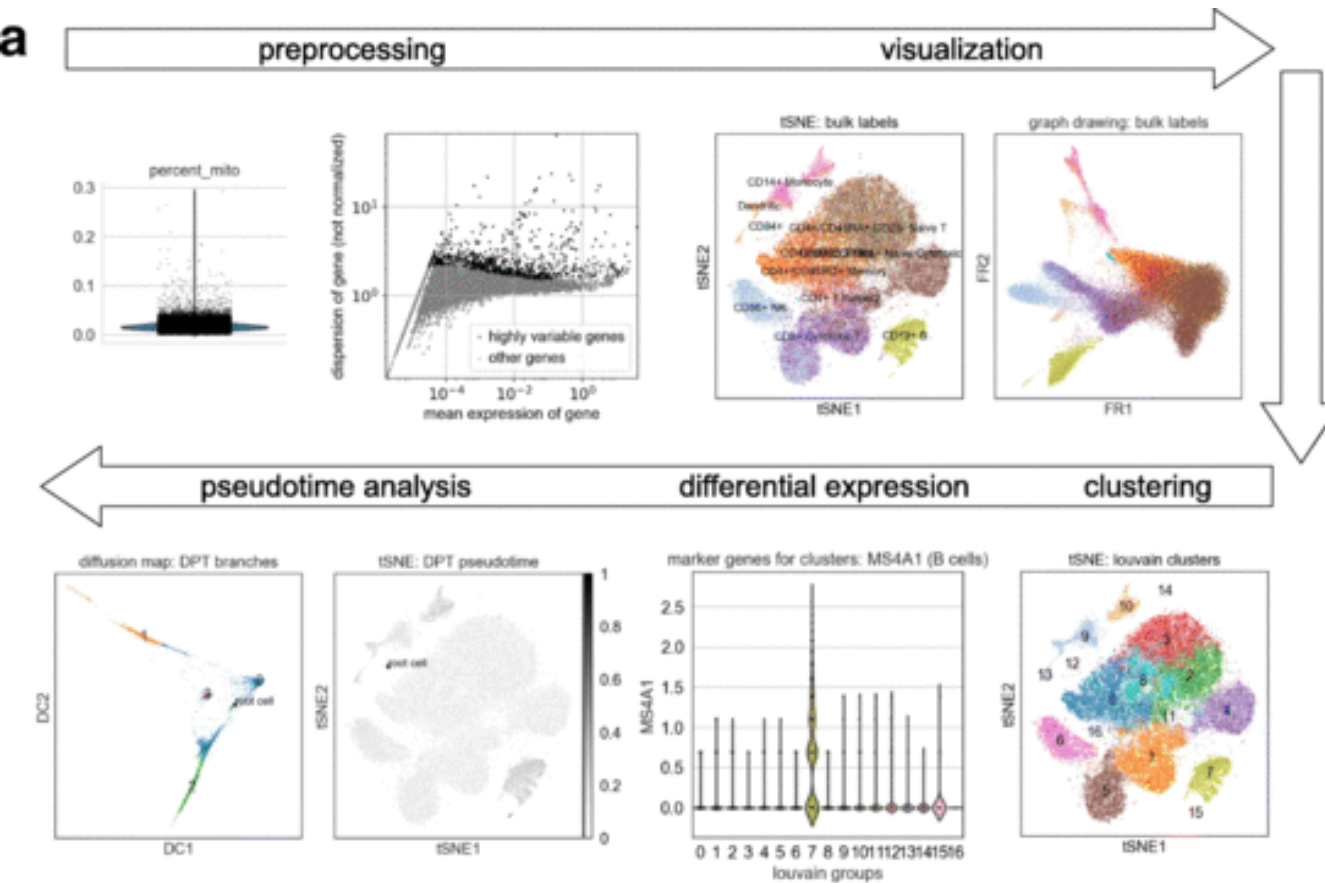
Additional analyses

- Allelic expression
- Variant calling
- Alternative splicing
- Copy-number variation
- Projection between datasets
- CRISPR-editing

Single-Cell RNA-Seq Reveals Dynamic, Random Monoallelic Gene Expression in Mammalian Cells

Single cells

Pooled embryos

(Deng et al. *Science* 2014)
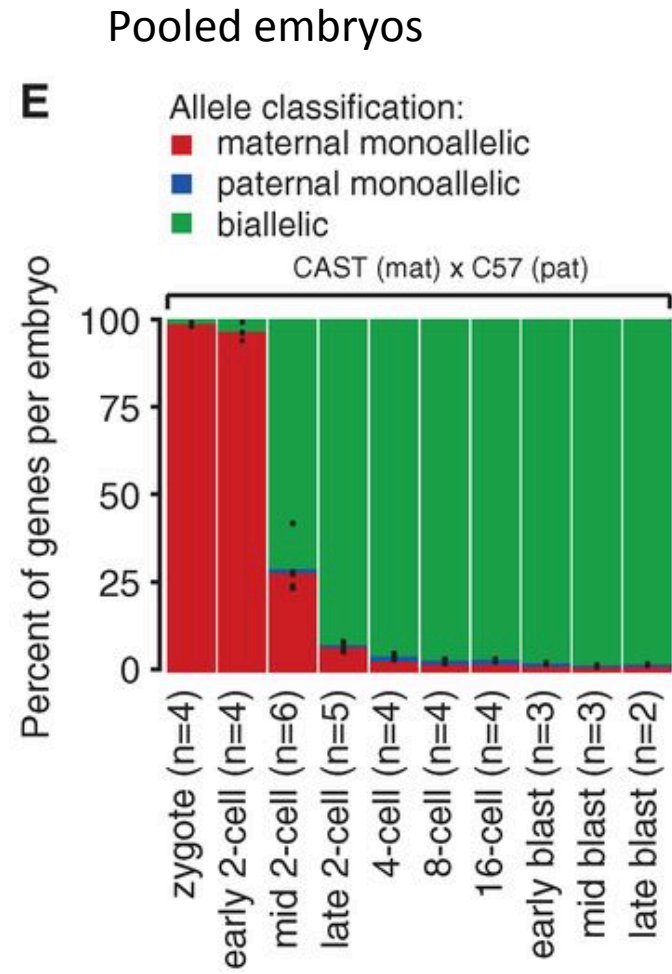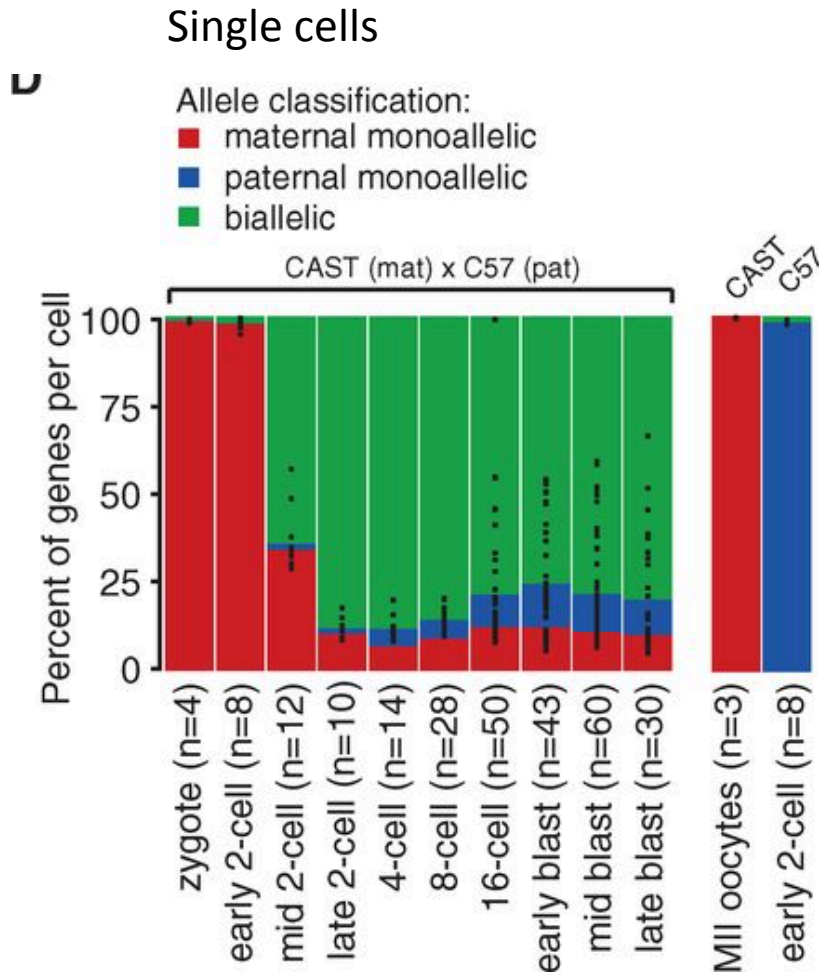
X Chromosome inactivation

(Petropoulos *et al. Cell* 2017)
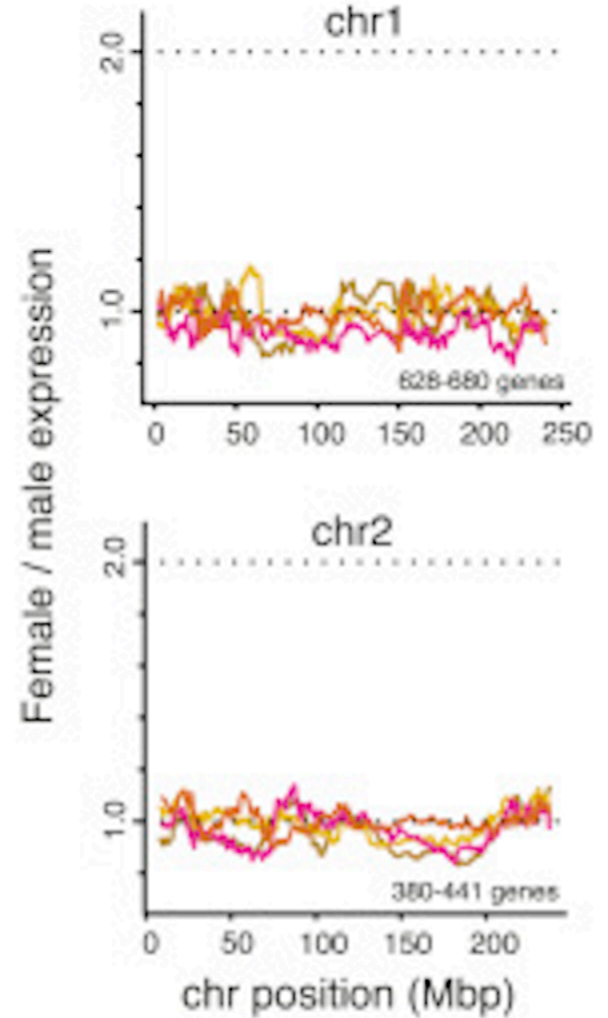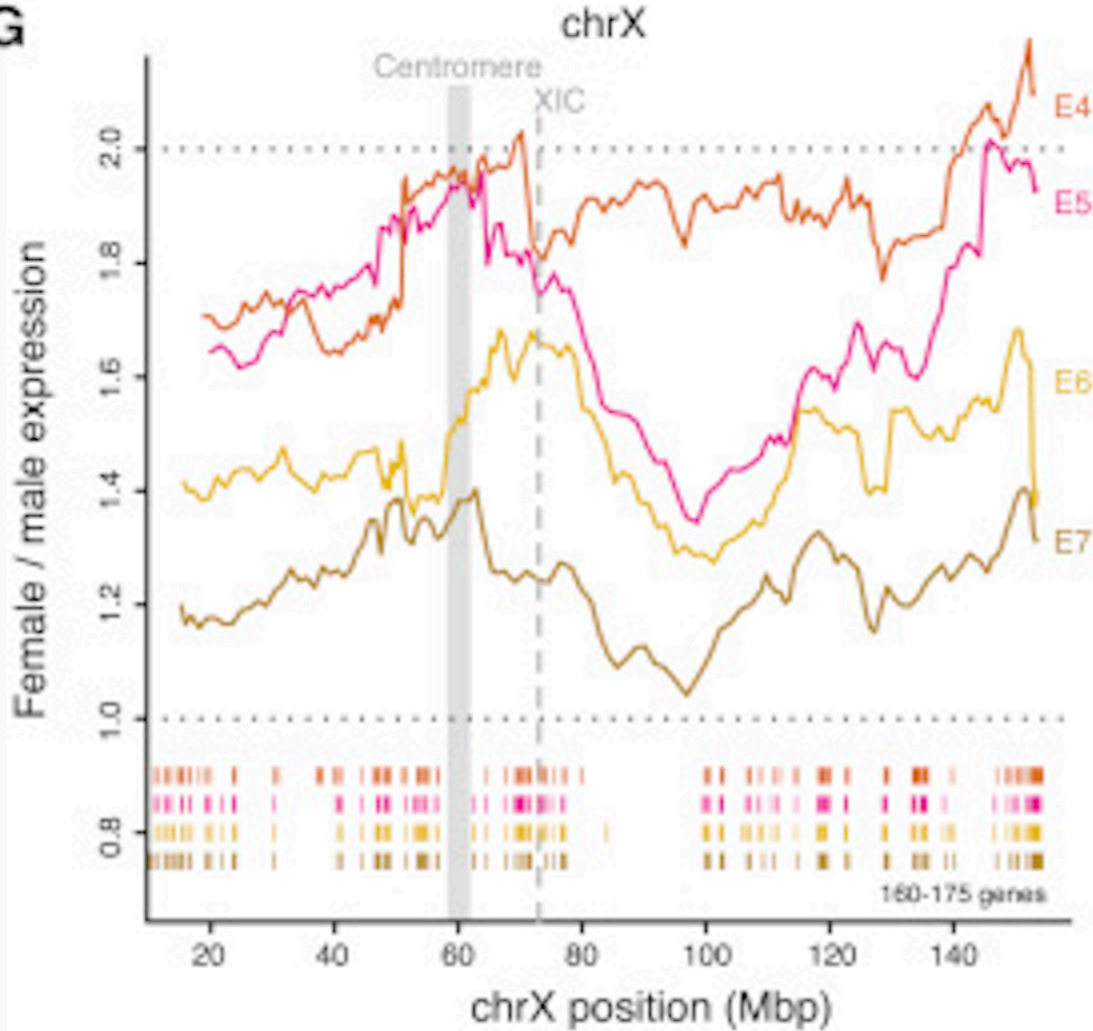
Using Single Nucleotide Variations in Cancer Single-Cell RNA-Seq Data for Subpopulation Identification and Genotype-phenotype Linkage Analysis

(Poiron et al. *BioRxiv* 2016)
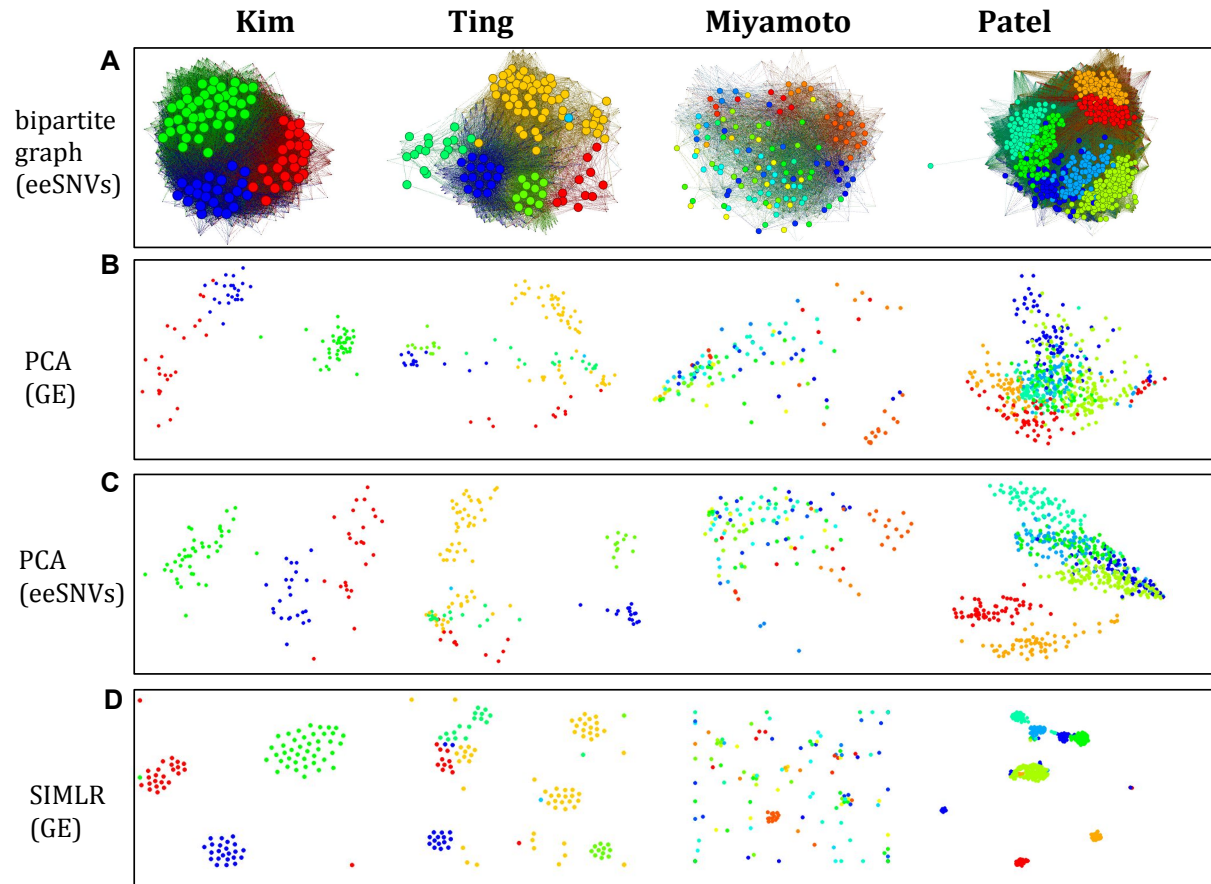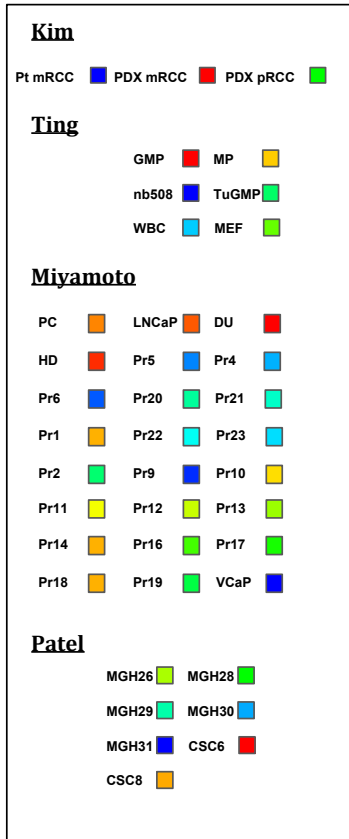
Multiplexed droplet single-cell RNA-sequencing using natural genetic variation

(Kang et al. Nature Biotech 2018)
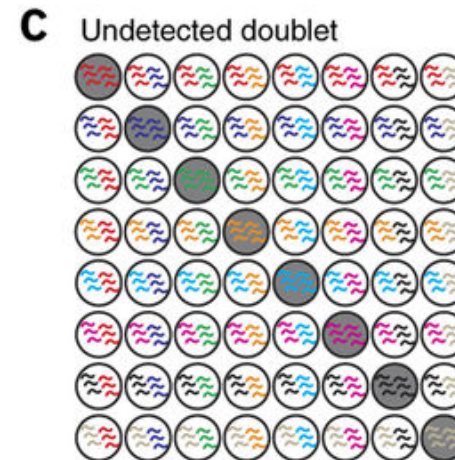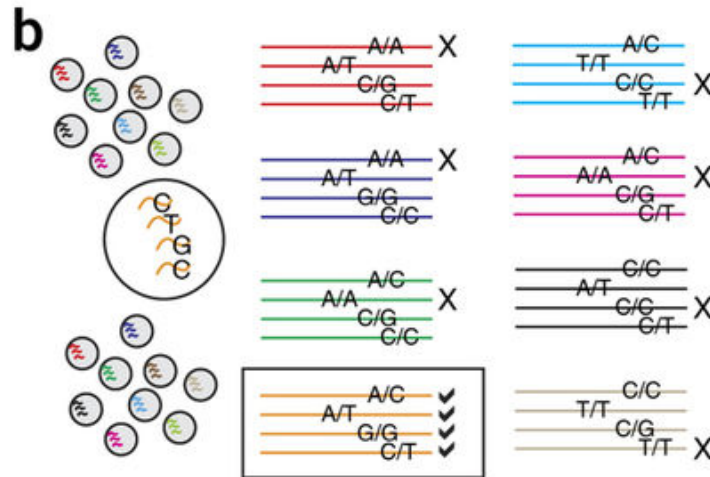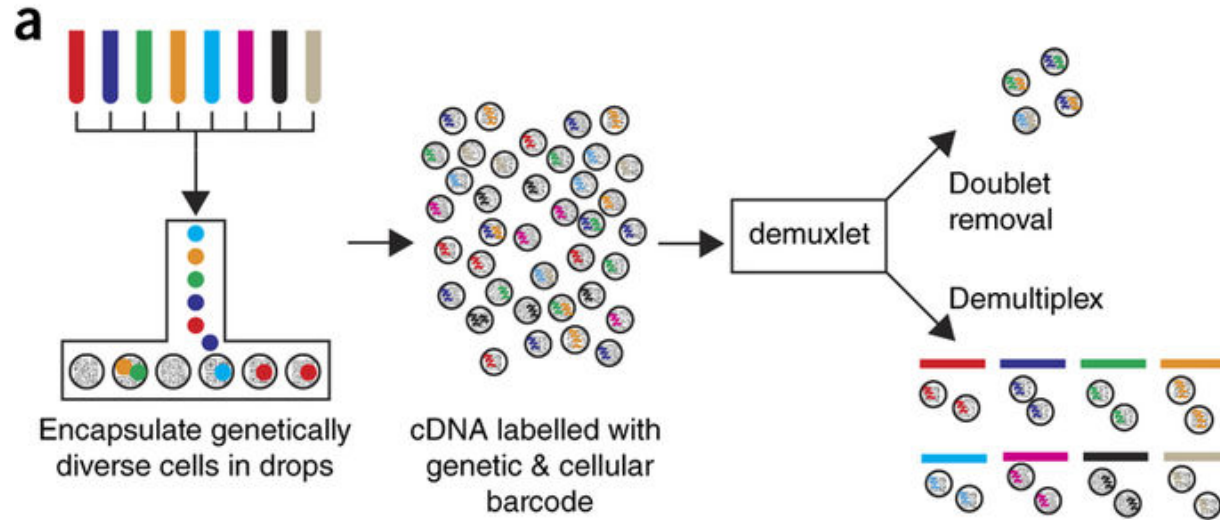
Dissecting the multicellular ecosystem of metastatic melanoma by single-cell RNA-seq

(Tirosh et al. *Science* 2016)
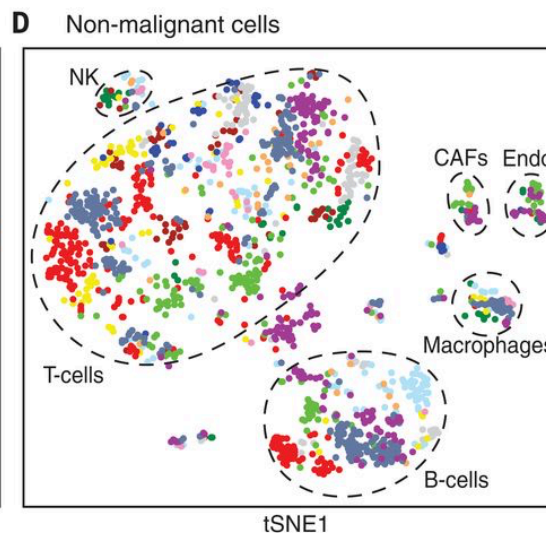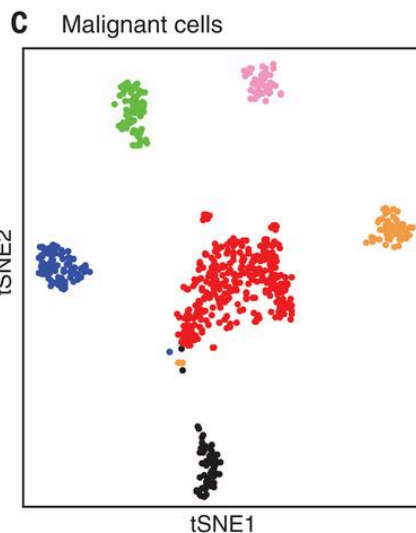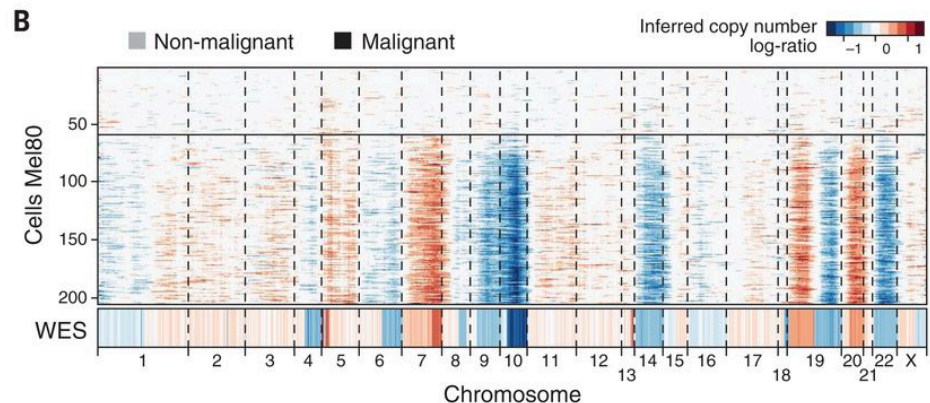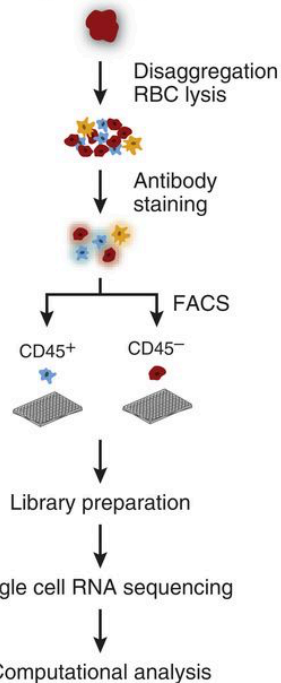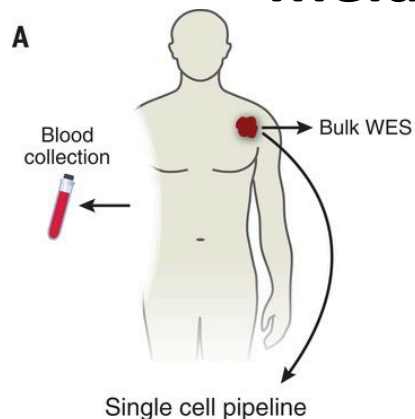
Cell specific alternative splicing

(Shalek et al. *Nature* 2013)

(Zhang et al *Cell* 2016)

scmap – projection between datasets

(Kiselev et al. BioarXiv 2017)
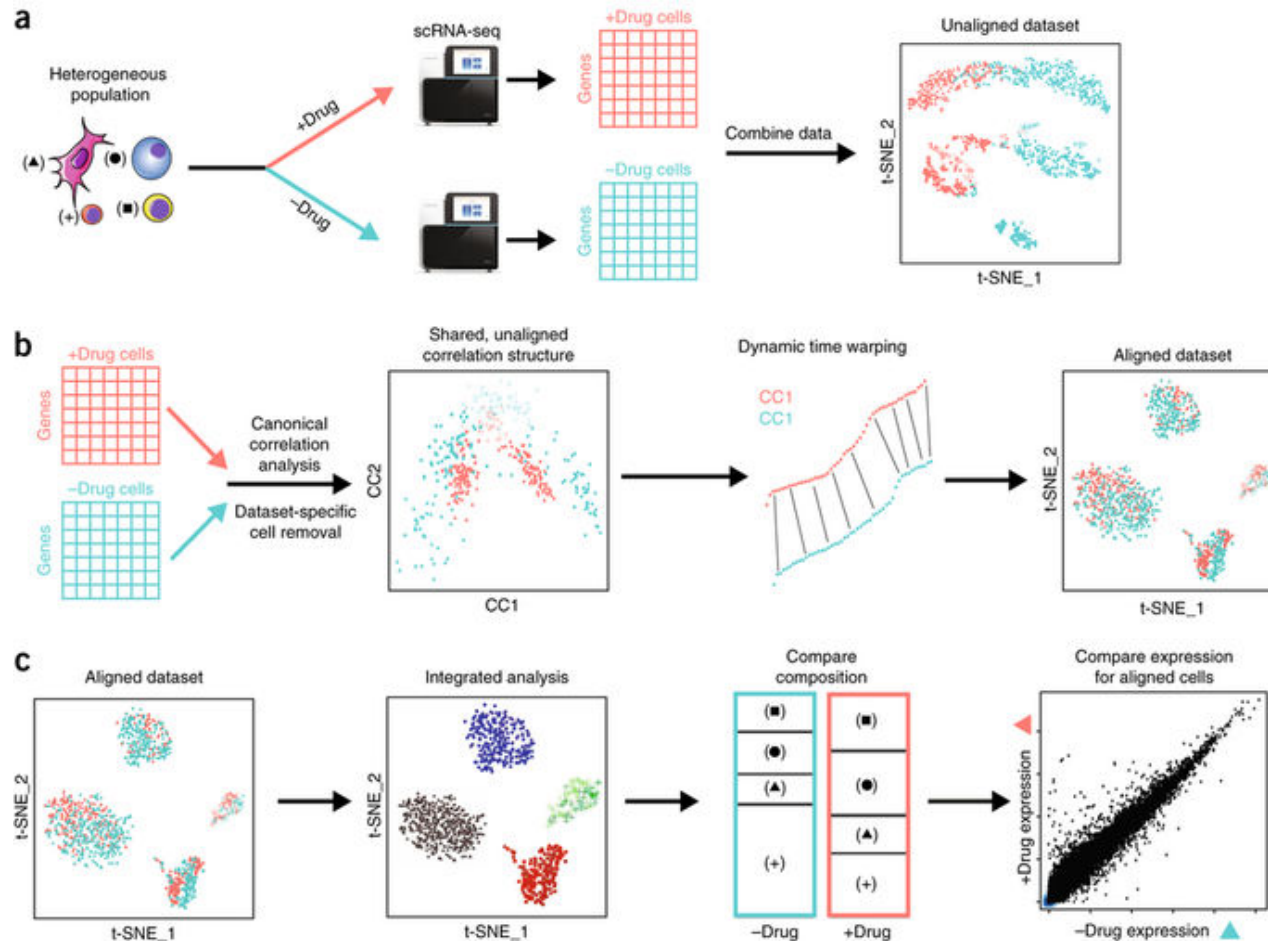
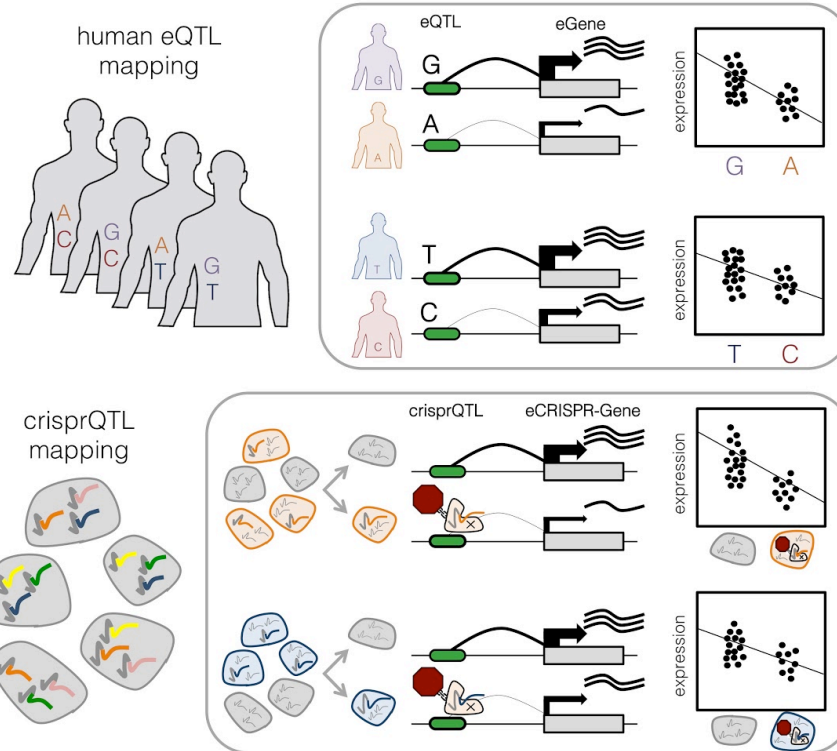# Seurat – canonical correlation analysis (CCA) for dataset integration

(Butler et al. Nature Biotech 2018)

crisprQTL mapping

(Gasperini et. al. BioRxiv 2018)
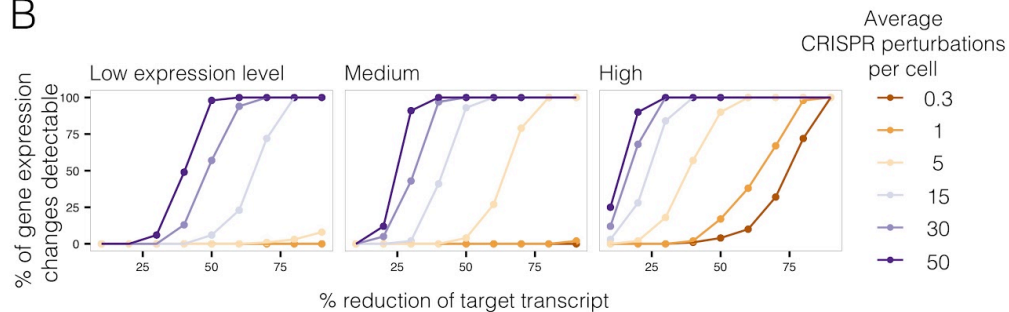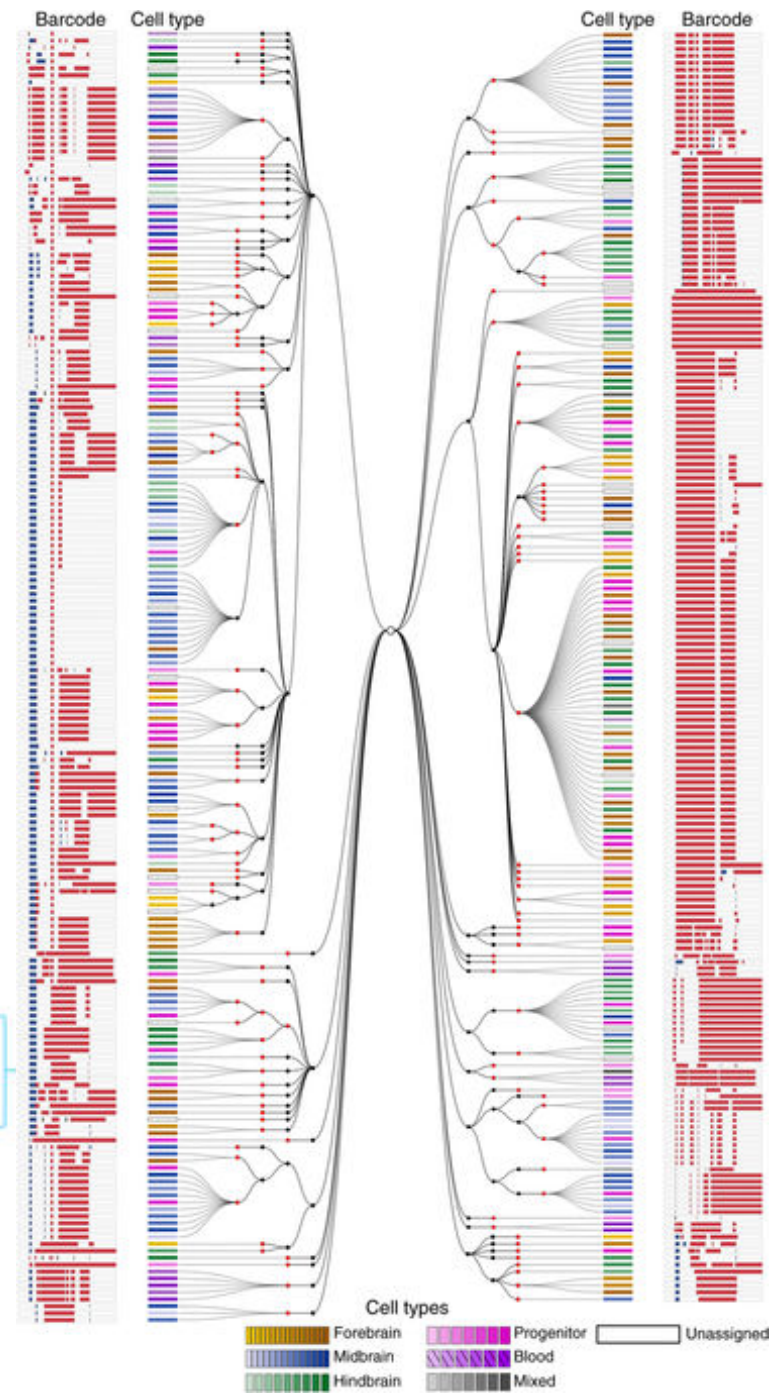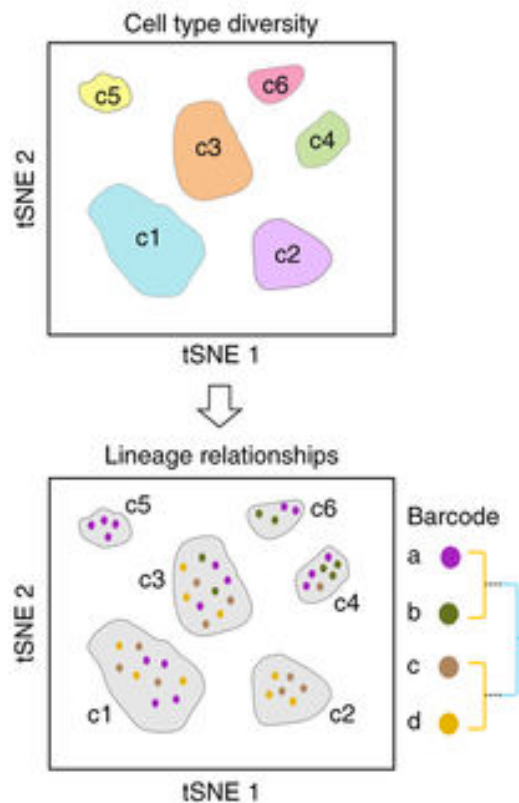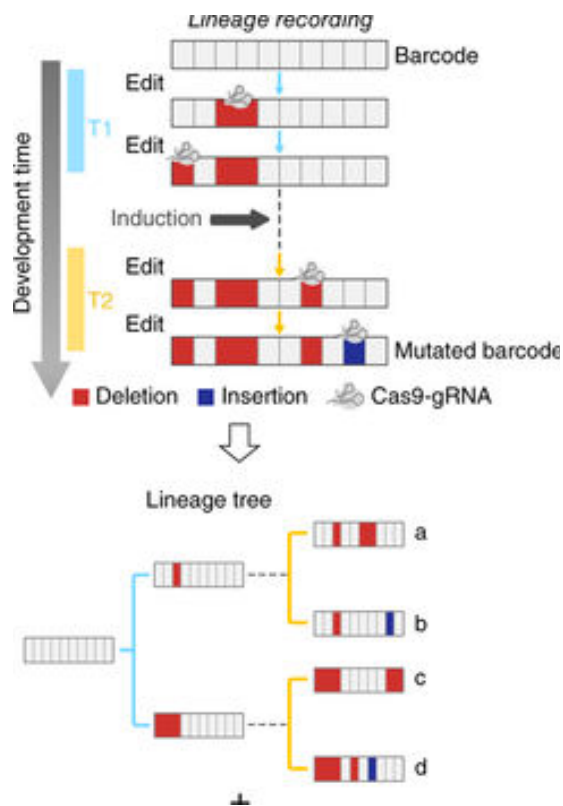
scGESTALT –
lineage tracing and cell profiling with CRISPR-Cas9 editing of barcodes

(Raj et al. Nature Biotech 2018)

Resources

- Good course at: https://hemberg-lab.github.io/scRNA.seq.course/

- Many of the packages have very thorough tutorials on their websites

- Repo with scRNA-seq tools: https://github.com/seandavi/awesome-single-cell

- Single cell assay objects for many datasets: https://hemberg-lab.github.io/scRNA.seq.datasets/

- Conquer datasets - salmon pipeline to many different datasets: http://imlspenticton.uzh.ch:3838/conquer/