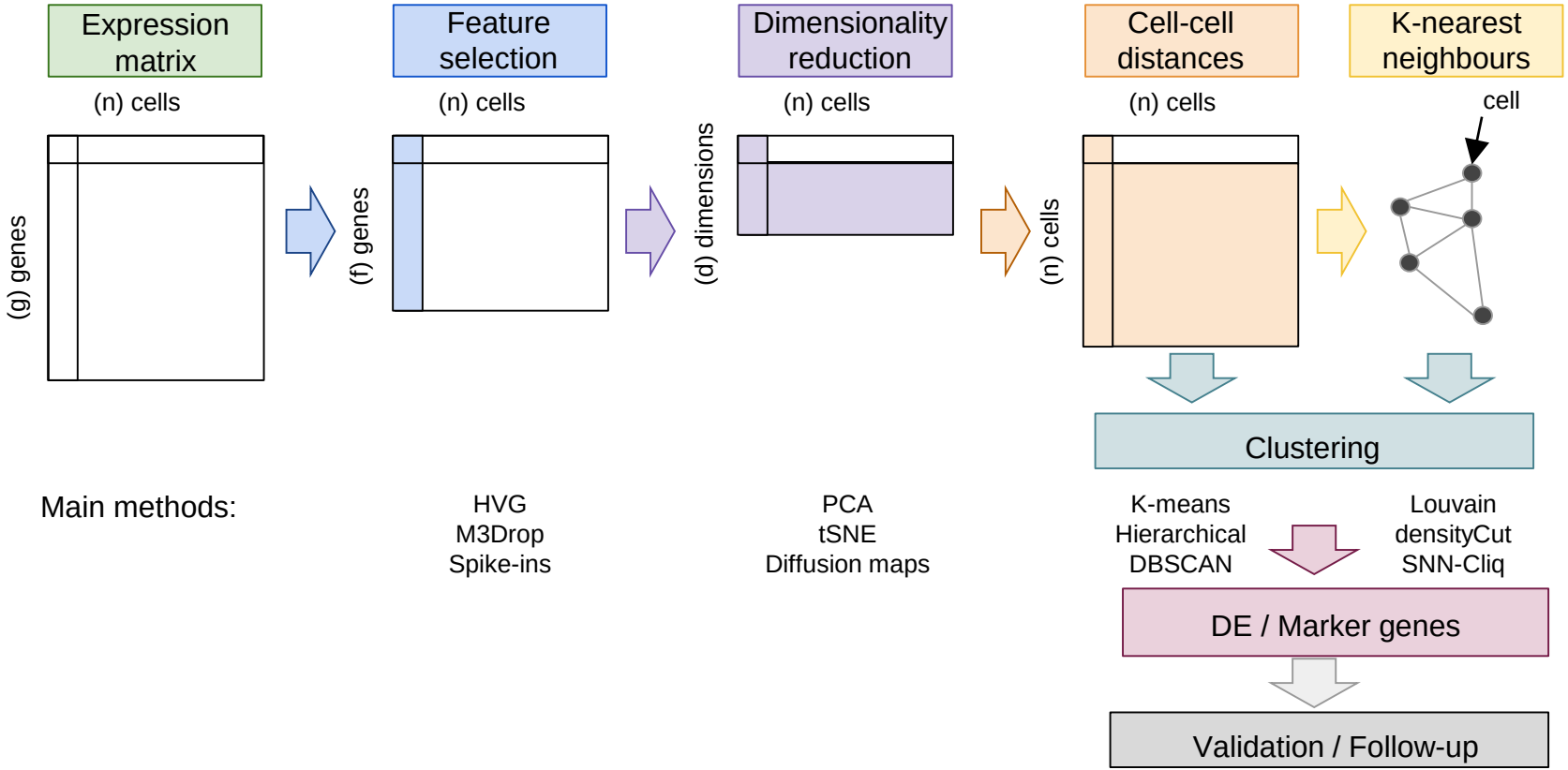
Identifying cell populations and Feature selection

Physalia : Lecture 4

Outline


Expression matrix
(n) cells
(g) genes
Feature selection
(n) cells
(f) genes
Dimensionality reduction
(n) cells
(d) dimensions
Cell-cell distances
(n) cells
(n) cells
K-nearest neighbours
cell
Clustering
Main methods:
HVG
M3Drop
Spike-ins
PCA
tSNE
Diffusion maps
K-means
Hierarchical
DBSCAN
Louvain
densityCut
SNN-Cliq
DE / Marker genes
Validation / Follow-up

Curse of Dimensionality

- High dimensional data poses specific statistical challenges
- As the number of dimensions increase:
  - Data becomes more and more sparse
  - Distances between points become more uniform
  - k-NN contain more “hubs”

E.g. Consider a classification based on a series of markers that are either “on” or “off”

# Markers	1	2	3	4
# Groups	2	4	8	16
Dist Btw Groups	1x1 1x0	1x2, 2x1 1x0	1x3, 3x2, 3x1 1x0	1x4, 4x3, 6x2, 4x1 1x0

Curse of Dimensionality

- High dimensional data poses specific statistical challenges
- As the number of dimensions increase:
  - Data becomes more and more sparse
  - Distances between points become more uniform
  - k-NN contain more “hubs”

E.g. Consider a classification based on a series of markers that are either “on” or “off”

# Markers	1	2	3	4
# Groups	2	4	8	16
Dist Btw Groups	1x1 1x0	1x2, 2x1 1x0	1x3, 3x2, 3x1 1x0	1x4, 4x3, 6x2, 4x1 1x0

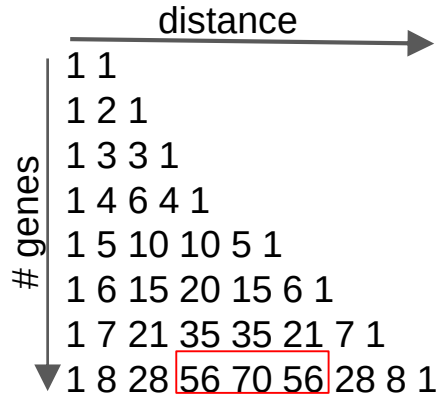
Note: that distances are become more and more concentrated in the middle

Curse of Dimensionality

- High dimensional data poses specific statistical challenges
- As the number of dimensions increase:
  - Data becomes more and more sparse
  - Distances between points become more uniform
  - k-NN contain more “hubs”

E.g. Consider a classification based on a series of markers that are either “on” or “off”

# Markers	1	2	3	4
# Groups	2	4	8	16
Dist Btw Groups	1x1 1x0	1x2, 2x1 1x0	1x3, 3x2, 3x1 1x0	1x4, 4x3, 6x2, 4x1 1x0



>70%

Curse of Dimensionality

- High dimensional data poses specific statistical challenges
- As the number of dimensions increase:
  - Data becomes more and more sparse
  - Distances between points become more uniform
  - k-NN contain more “hubs”

E.g. Consider a classification based on a series of markers that are either “on” or “off”

# Markers	1	2	3	4	1,000
# Groups	2	4	8	16	1e301
Dist Btw Groups	1x1 1x0	1x2, 2x1 1x0	1x3, 3x2, 3x1 1x0	1x4, 4x3, 6x2, 4x1 1x0	. . . . .

With only 1,000 genes assayed we have more possible combinations of “on” and “off” than there are molecules in the universe (1e100).

scRNASeq assays ~10,000 genes

Questions?

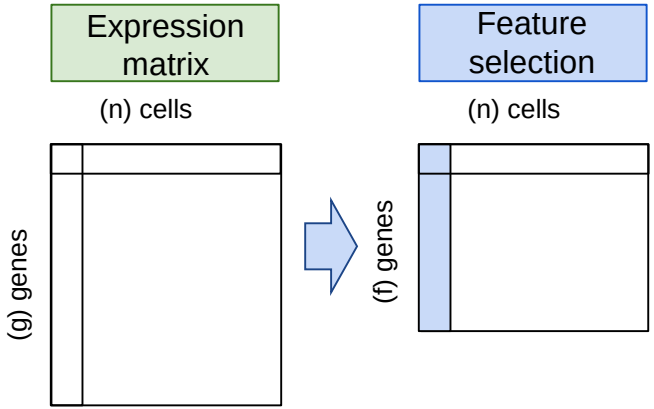
Feature selection

Reducing Dimensionality: Feature Selection

One way to reduce the dimensionality of our data is to identify and remove “dimensions” (i.e. genes) which don’t contain any biological signal.

Single-cell RNASeq assays thousands of genes:

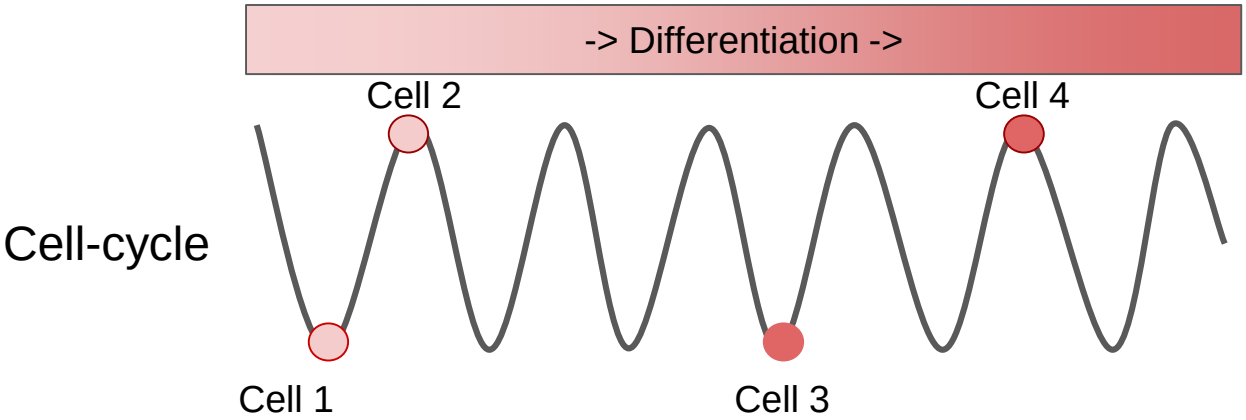
- All contain technical noise
- All contain batch effects
- Only a subset respond to biology



Supervised Feature Selection

(1) If “true” biological structure is known, then genes with expression agreeing with that structure can be selected (e.g. DE).

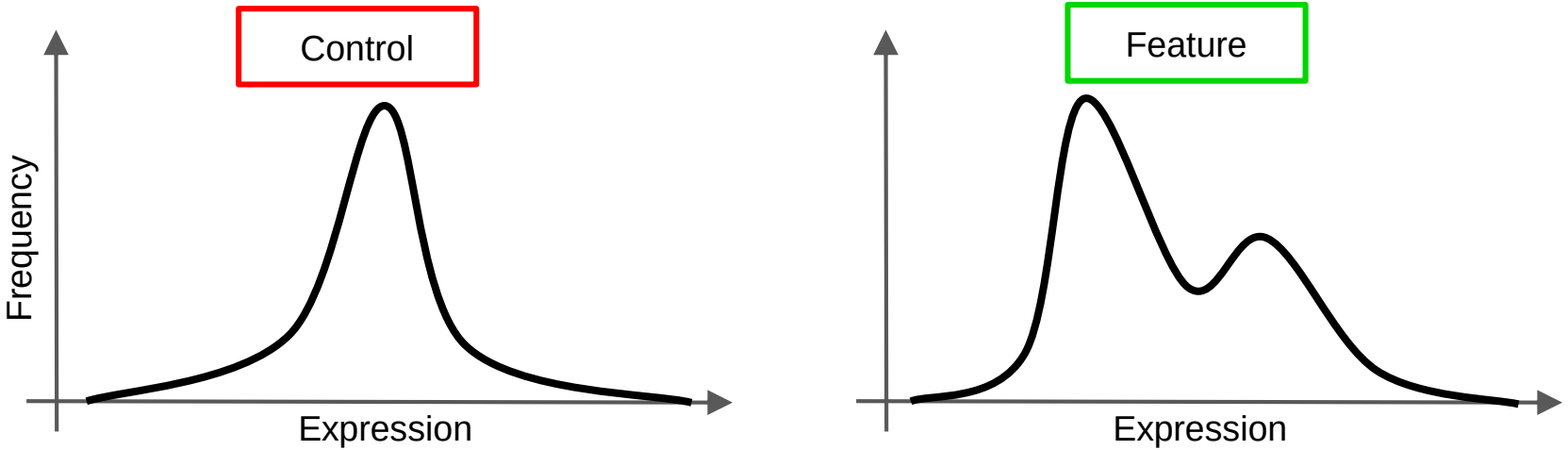
(2) If “true” function of genes are known, we can eliminate those that are not relevant to our biological question. E.g. cell-cycle



Unsupervised Feature Selection

One way to reduce the dimensionality of our data is to identify and remove “dimensions” (i.e. genes) which don’t contain any biological signal.

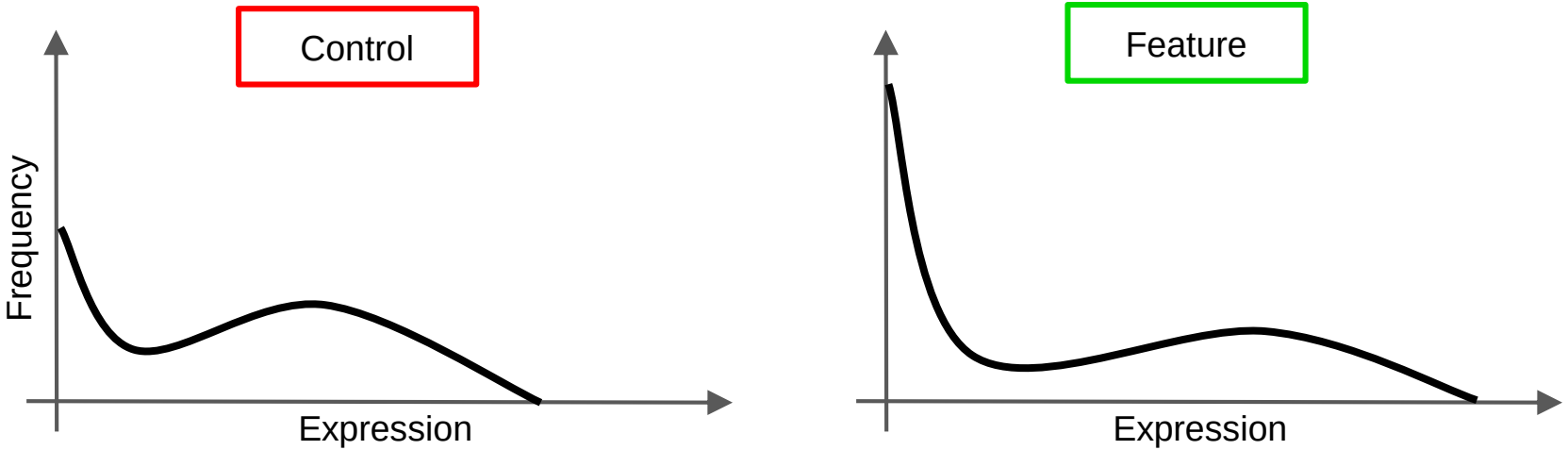
(1) If “control genes” are known, features with patterns of expression different from the controls can be selected:



Unsupervised Feature Selection

Bimodality can be detected using variance:mean.

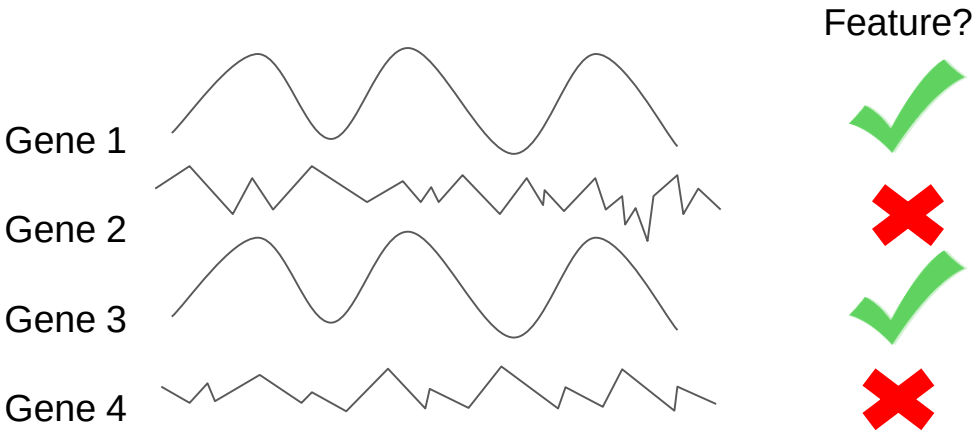
scRNASeq bimodality is generally caused by an excess of zeros so can be detected by #zeros:mean.



Unsupervised Feature Selection

One way to reduce the dimensionality of our data is to identify and remove “dimensions” (i.e. genes) which don’t contain any biological signal.

(2) If the biological signal affects many genes similarly, we can select features that are strongly correlated with each other.



Dimensionality Reduction

Dimensionality Reduction: Metafeatures

Feature selection is limited to the features which are present in the data.

Underlying biological features can be a result of a combination of individual genes, e.g.

- Cell-cycle stage
- Tissue identity
- Differentiation state
- Morphology

Dimensionality Reduction: Metafeatures

Feature selection is limited to the features which are present in the data.

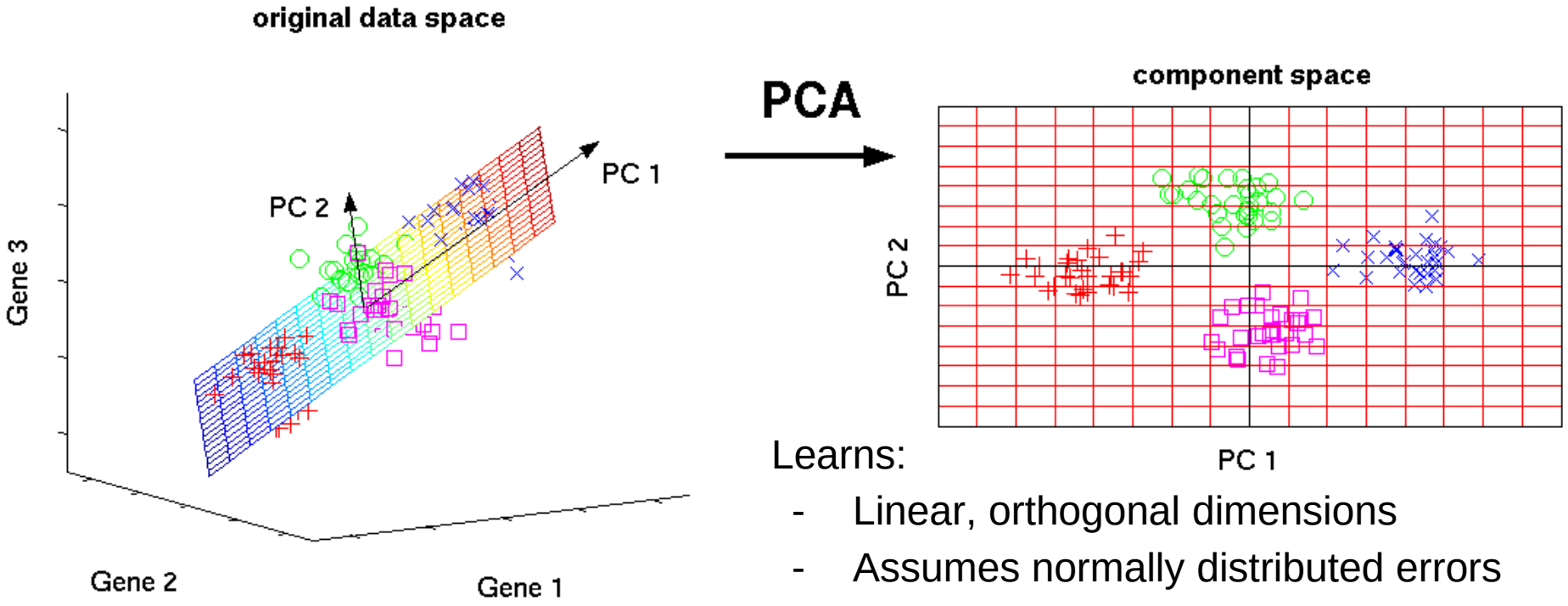
Underlying biological features can be a result of a combination of individual genes, e.g.

- Cell-cycle stage
- Tissue identity
- Differentiation state
- Morphology

Usually we don’t know:

- the identity of the underlying meta-features
- genes responsible for any particular feature

Principal Component Analysis



Learns:

- Linear, orthogonal dimensions
- Assumes normally distributed errors
- ZINB-WaVE, assumes zero-inflated negative binomial.

t-SNE

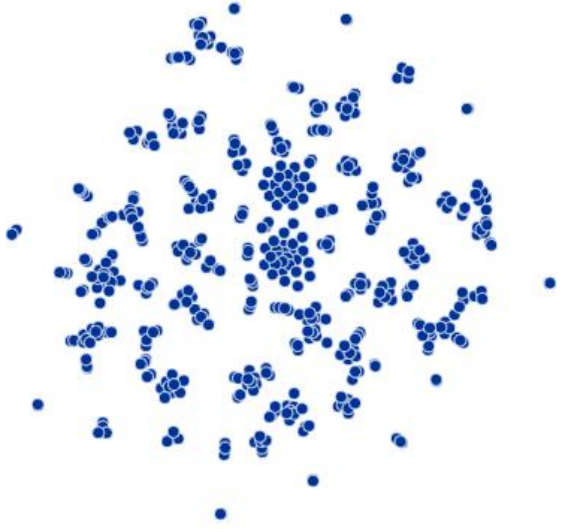
t-SNE is NOT a dimensionality reduction method

t-SNE was designed for visualizing high dimensional data in 2D plots.

It is stochastic and sensitive to parameters.

The algorithm specifically tries to find and highlight local clustering in the data.

perplexity = 2



Does this data contain clusters?

t-SNE

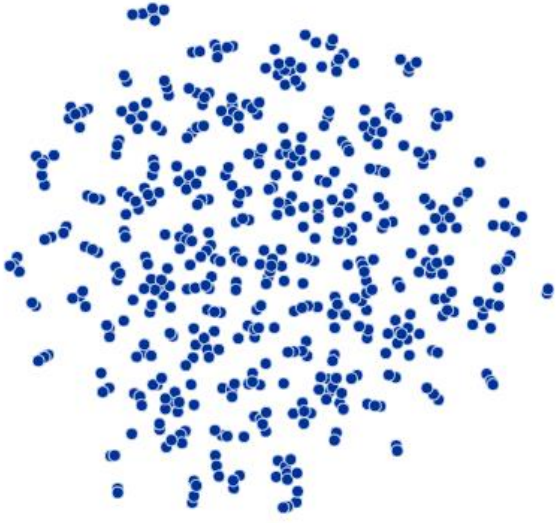
t-SNE is NOT a dimensionality reduction method

t-SNE was designed for visualizing high dimensional data in 2D plots.

It is stochastic and sensitive to parameters.

The algorithm specifically tries to find and highlight local clustering in the data.

perplexity = 5



Does this data contain clusters?

t-SNE

t-SNE is NOT a dimensionality reduction method

t-SNE was designed for visualizing high dimensional data in 2D plots.

It is stochastic and sensitive to parameters.

The algorithm specifically tries to find and highlight local clustering in the data.

perplexity = 50

Does this data contain clusters?

t-SNE: What does it do?

t-Stochastic Neighbour Embedding

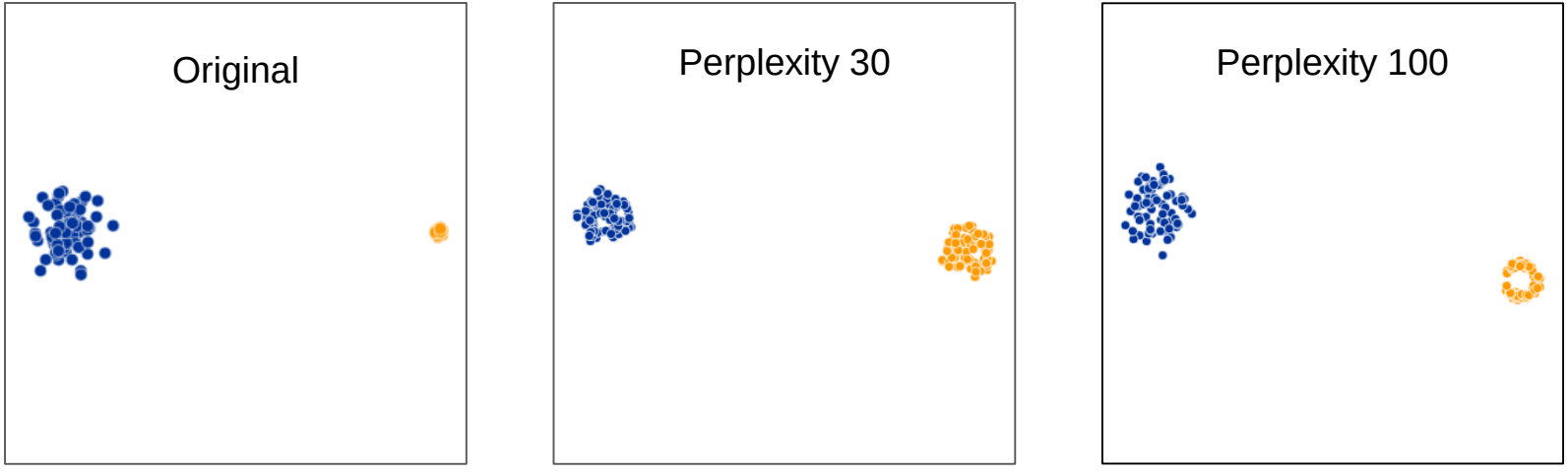
t-SNE creates a 2D “embedding” of a set of points.

- Non-linear, stochastic method
- Local adaptation/tuning within a “neighbourhood”

“Perplexity” parameter controls the “local-ness” of the algorithm.

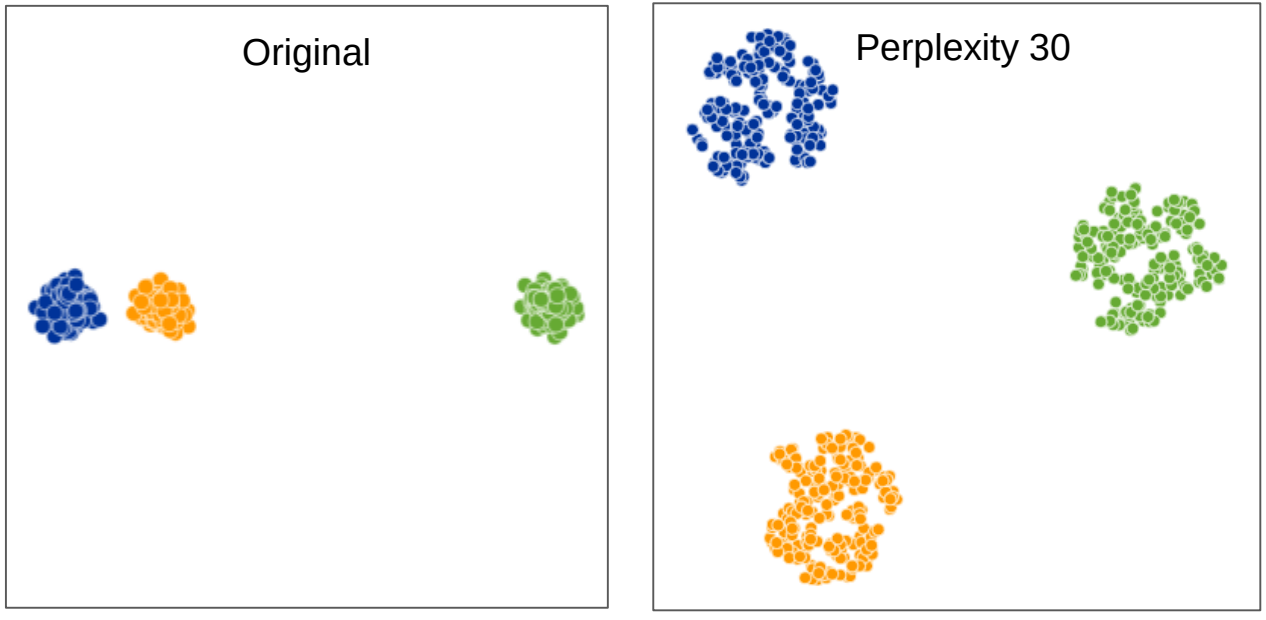
Local tuning means global information is generally lost after t-SNE

t-SNE: Challenges (Cluster size)



Local adaptation of distance will even out cluster densities.

t-SNE: Challenges (Distance btw clusters)

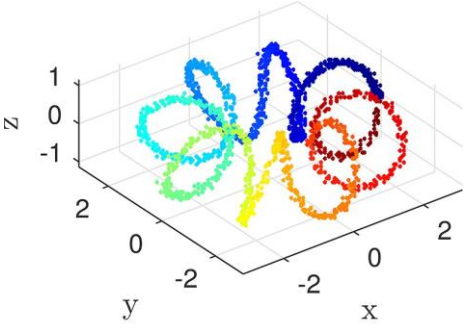


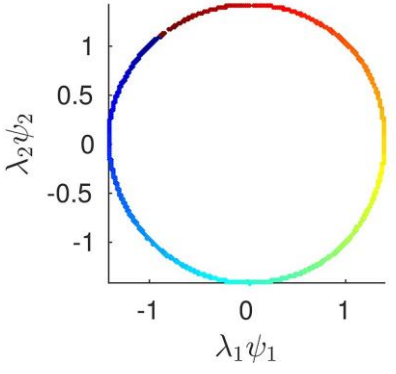
Local adaptation of distance will evenly spread clusters.

Diffusion Maps

Learns the underlying non-linear ‘manifold’ of the data

- Models a random walk on the data using diffusion
  - Probability of A->B = f(d(A,B)) [f is usually gaussian]
- This is turned into Markov chain
  - Running this Markov chain for different amounts of time captures local & global structure in the data
- Eigendecomposition of the Markov chain can be used to find a lower dimensional representation of the data.

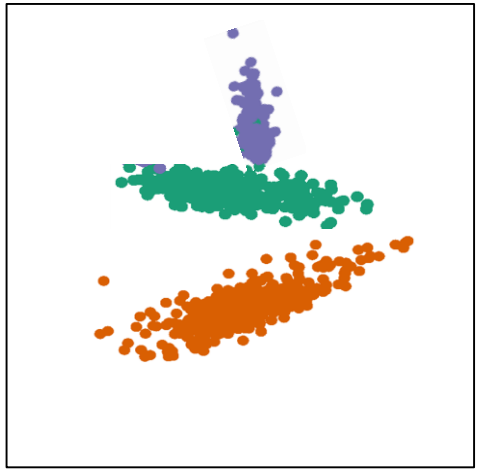


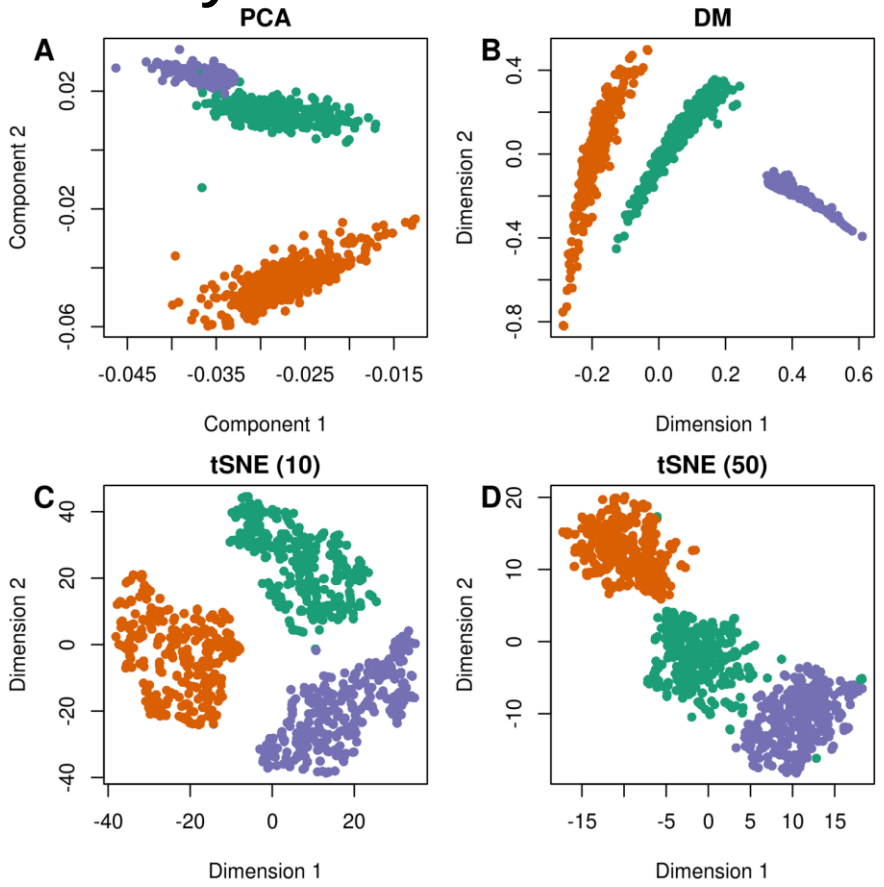


Coifman, 2006, Applied and Computational Harmonic Analysis, doi:10.1016/j.acha.2006.04.006

Comparison of Dimensionality Reduction

~Reality





Clustering

Identifying populations: clustering

A common application of scRNA-seq is the de novo discovery/annotation of cell-types

Computationally this is a hard “unsupervised clustering” problem:

- Number of possible clusterings is enormous
  - There are 1e29 possible ways to place 100 cells into two equally sized clusters
- Often we do not know the number/size of clusters *a priori*

In addition,

- scRNA-seq data is noisy and high dimensional.

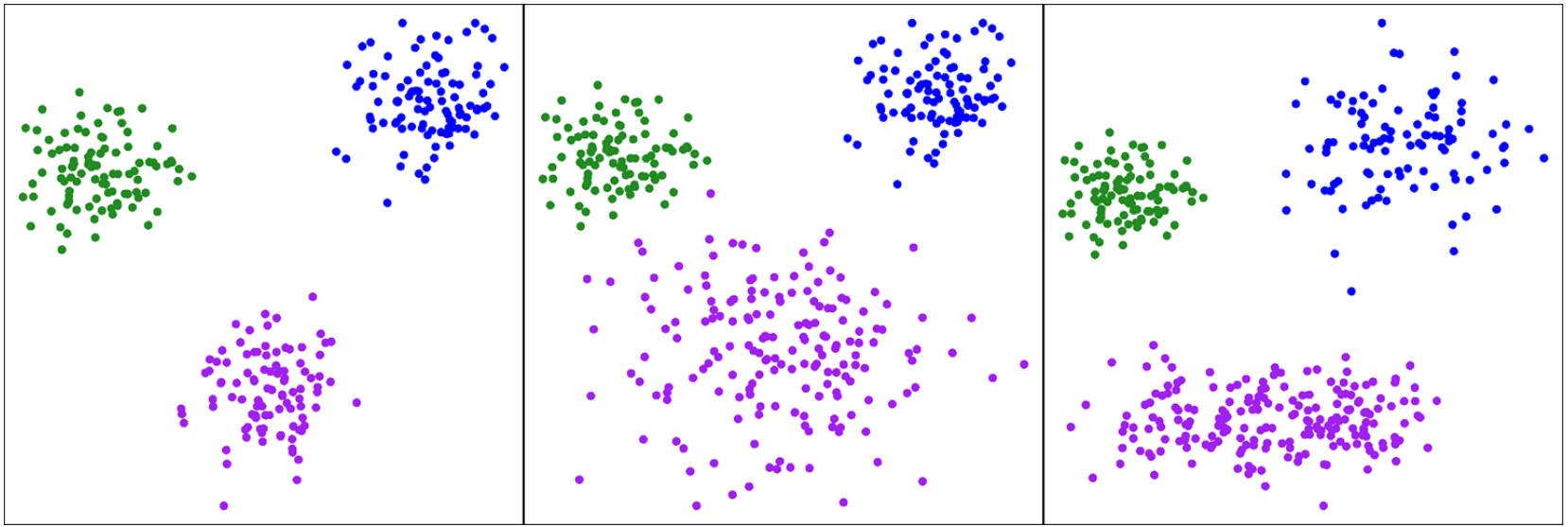
Identifying populations: clustering

How do we define a cluster?

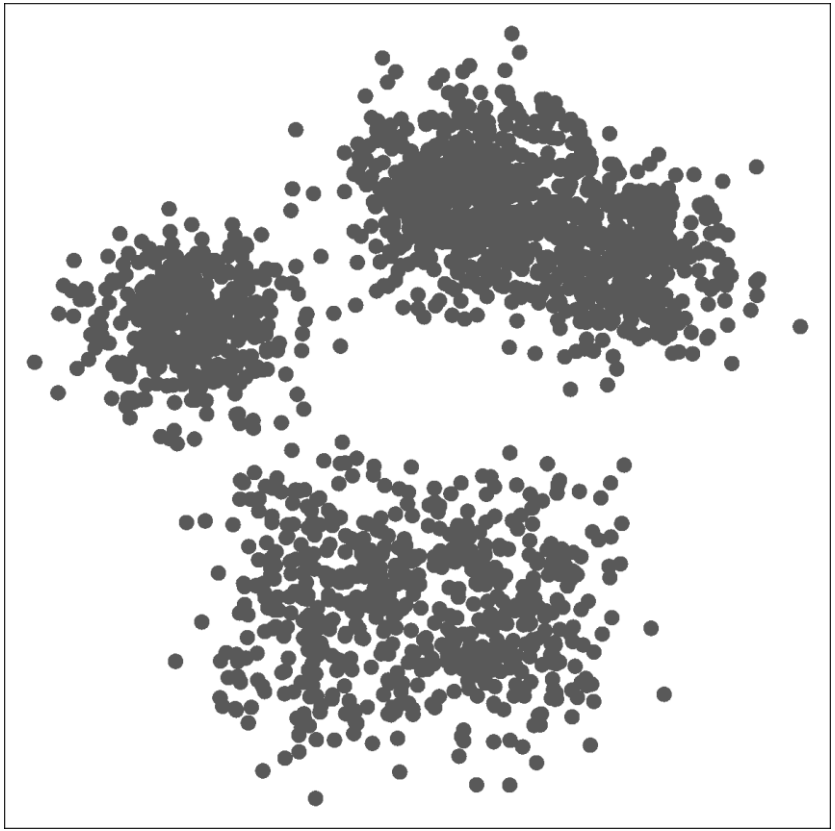
Are they equally sized round groups of points?

Do they have varying size/density?

Can they be shapes other than round?



How many clusters are there?

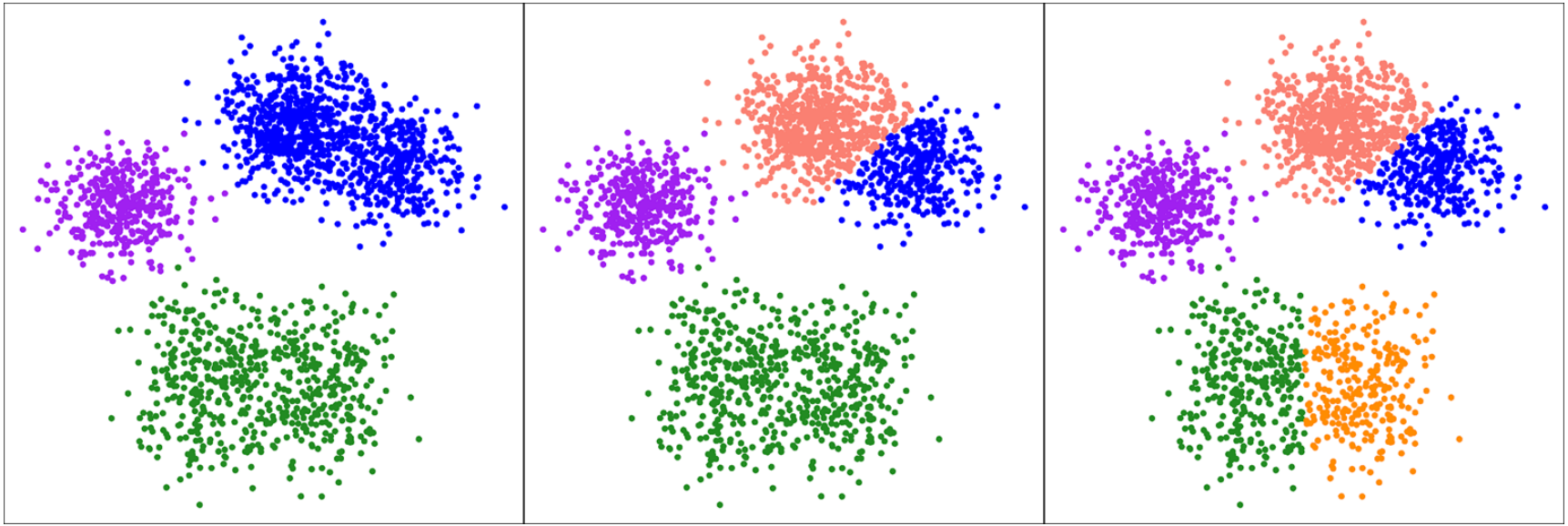


How many clusters are there?

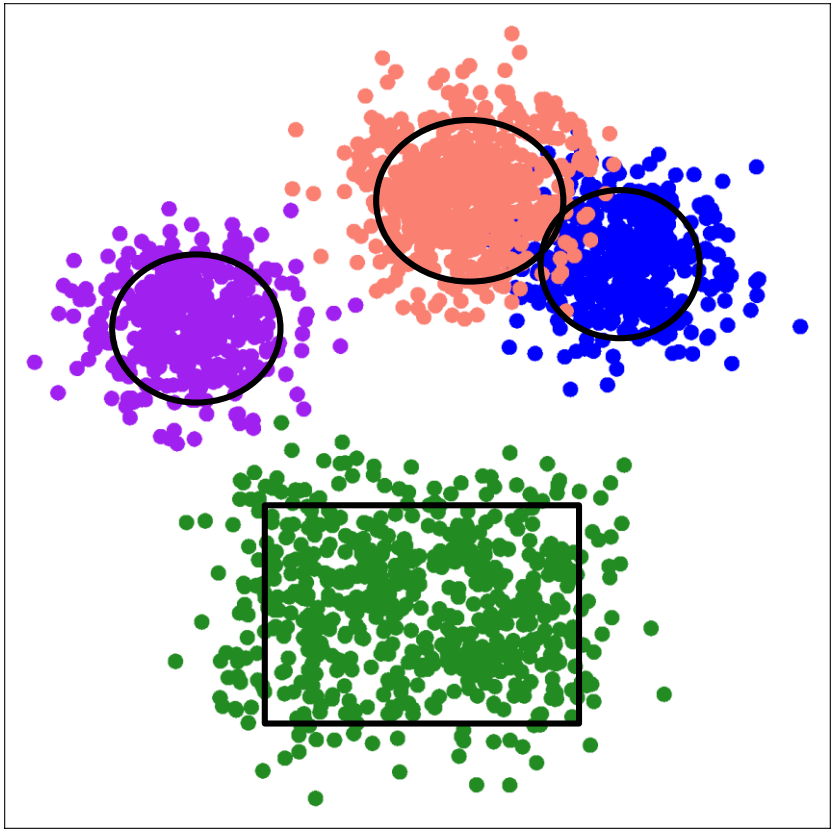
k=3?

k=4?

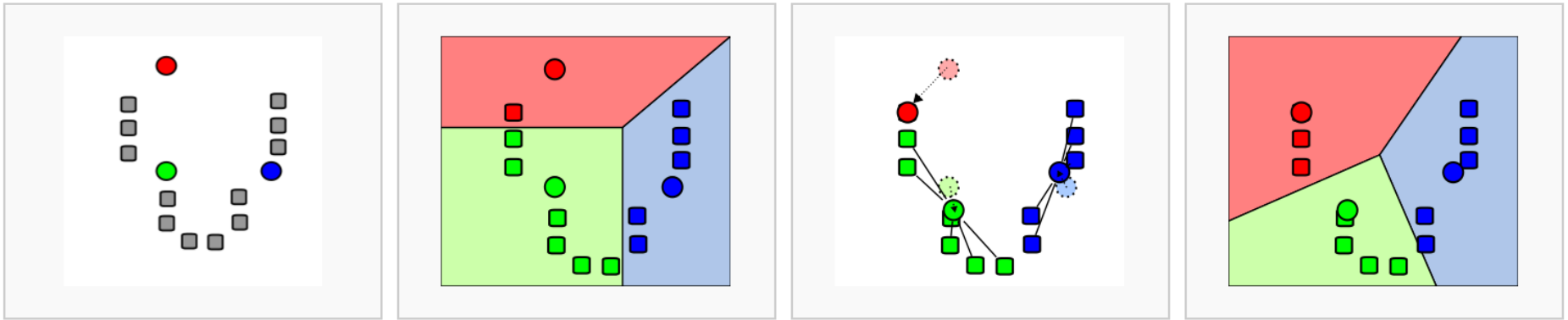
k=5?



How many clusters were there?



K-Means



What does a cluster look like?

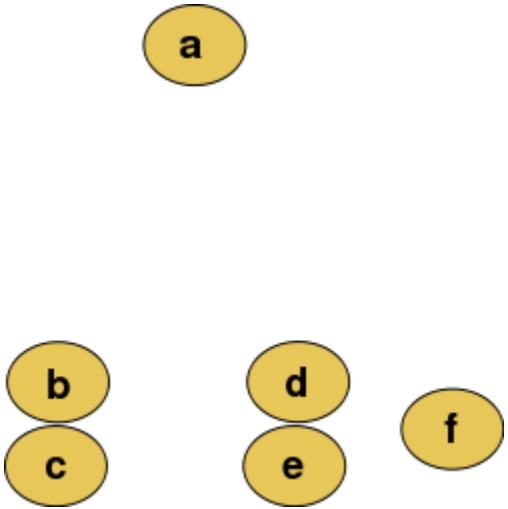
- k-means assumes “round” clusters containing roughly equal numbers of cells

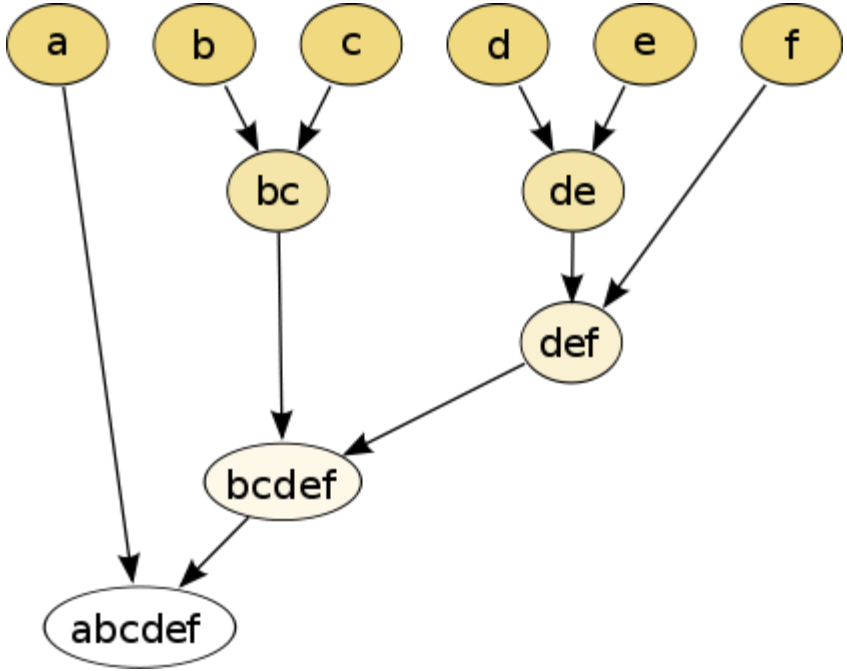
How many clusters are there?

- Number of clusters (k) is specified with an argument, regardless of the actual structure in the data k clusters will be identified and returned.
- All cells are assigned to exactly one cluster

Scalability: K-means is extremely fast and scalable O(n)
Stochasticity: starting positions are random, algorithm is deterministic

Hierarchical





Hierarchical

What does a cluster look like?

- Different “linkage” methods identify different types of clusters:

Round equal sized:

“complete” : the distance between clusters is the maximum distance between all pairs of points

“ward” : the distance between clusters is the sum of squared distances to the cluster centre

“average” : the distance between clusters is the mean distance between their points

“centroid” : the distance between clusters is the distance between their centers

Any shape:

“single” : the distance between clusters is the smallest distance between all pairs of points

How many clusters are there?

- The resulting clustering tree can be cut at different heights (h) to generate any number (k) clusters
- Height may be algorithmically chosen using a variety of methods, or a dynamic cutting procedure can be used.
- All cells are assigned to exactly one cluster

Scalability: Hierarchical clustering is relatively slow $O(n^2)$

Stochasticity: all hierarchical algorithms are completely deterministic

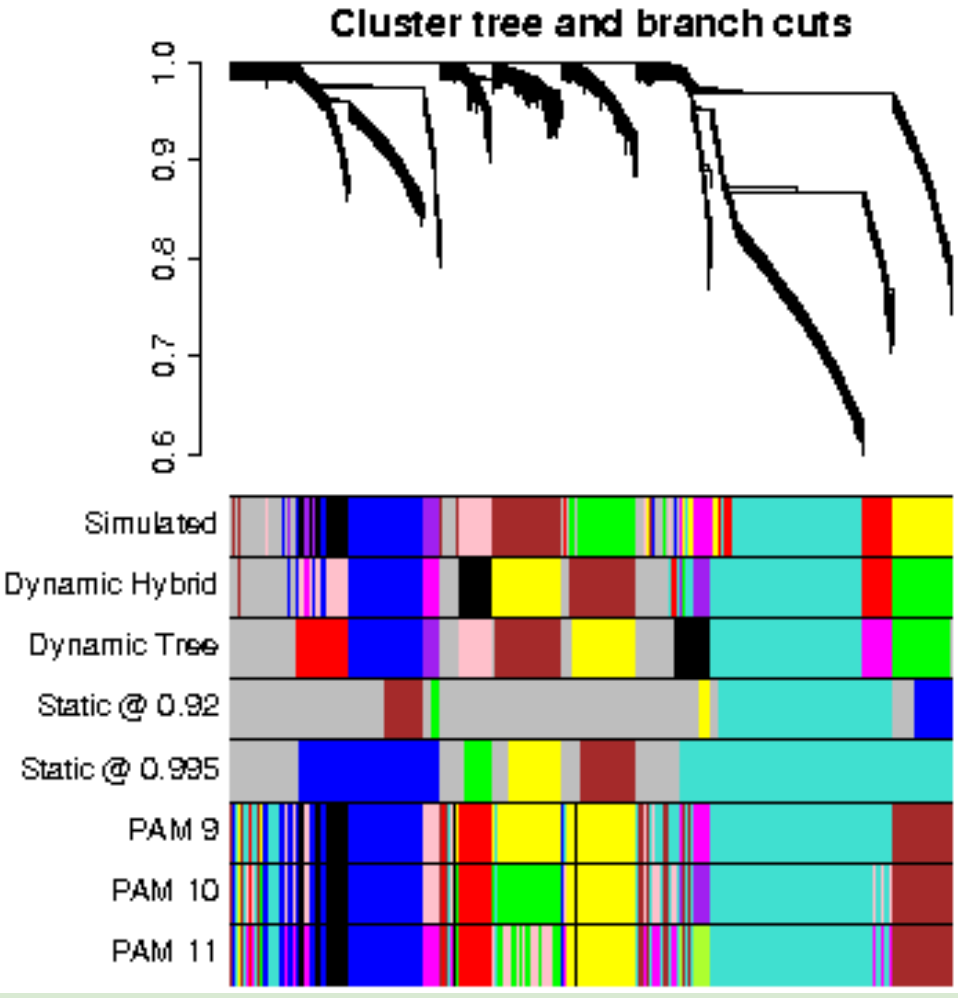
Cutting Algorithms

Various algorithms available

Dynamic Tree Cut is most popular
- designed for gene-trees

Single-cell studies often just pick a long branch by eye.

Single-cell clustering algorithms use other cutting rules.



Density (DBSCAN)

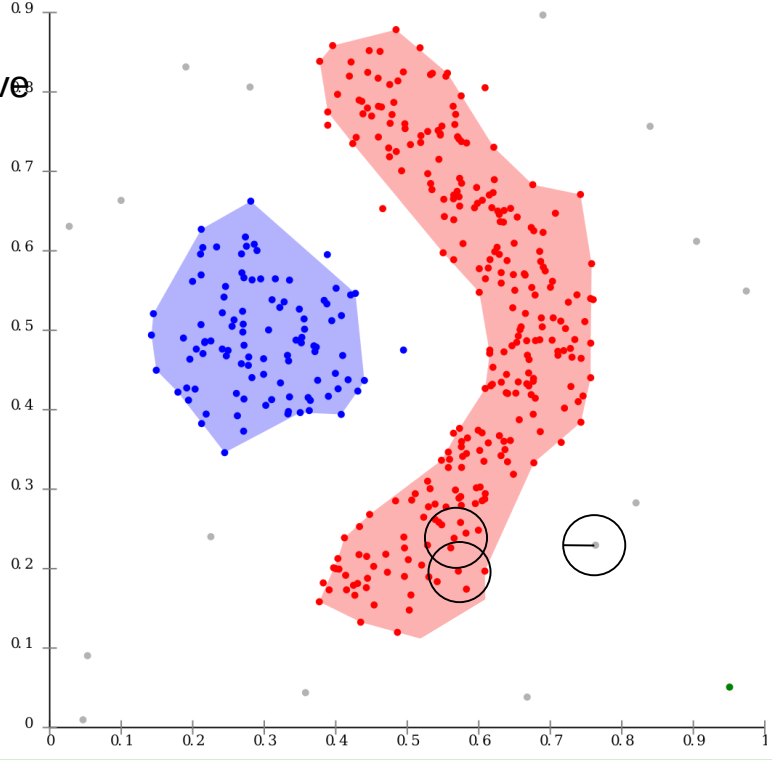
What does a cluster look like?

- Clusters are contiguous sets of points with density above a specified threshold containing at least a minimum number of points (minPts)

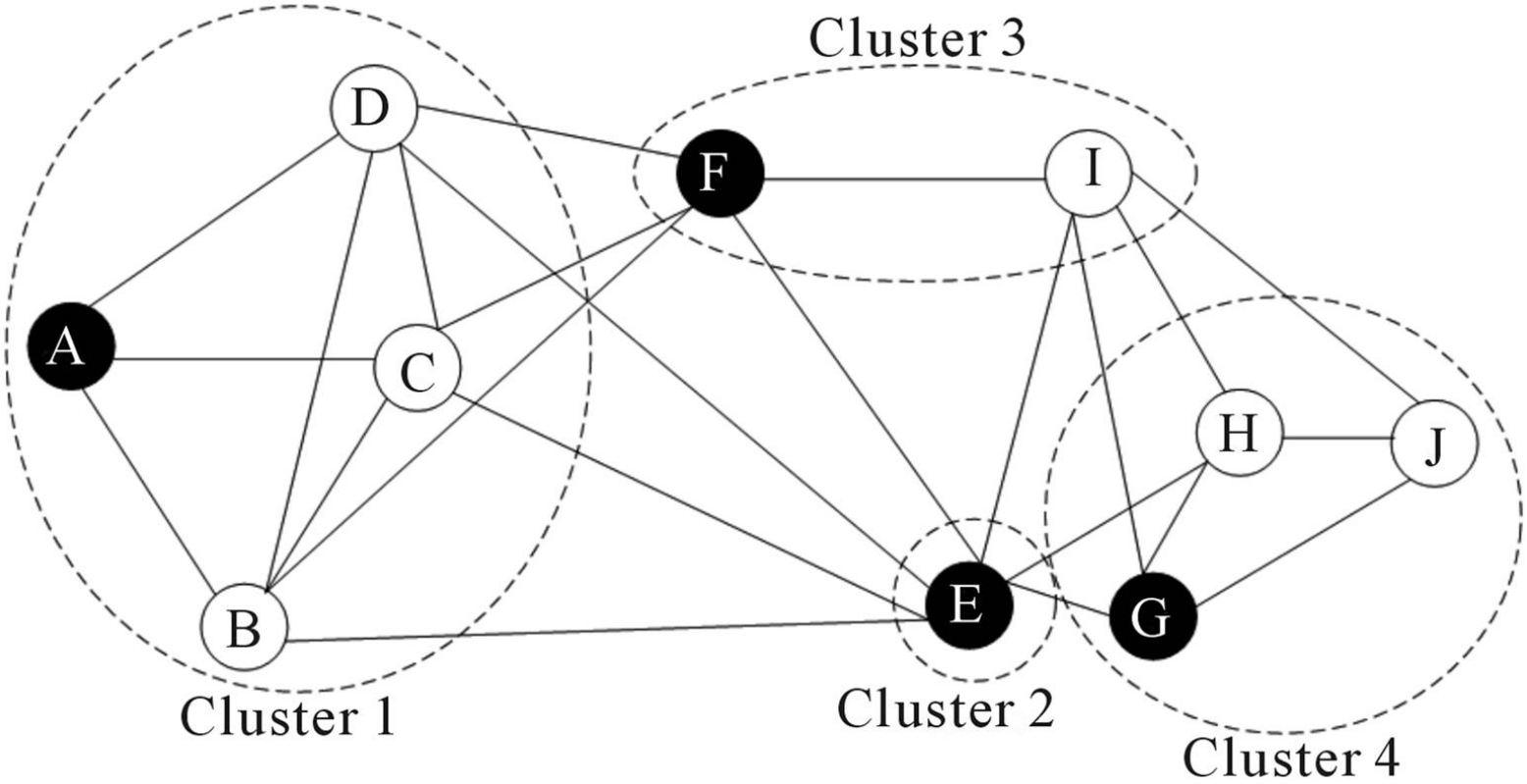
How many clusters are there?

- The number of clusters is determined by the algorithm based on the density parameter (***ε***)
- Cells may be assigned to one or no cluster

Scalability: density clustering is fast and scalable O(n log n)
Stochasticity: outcome is deterministic

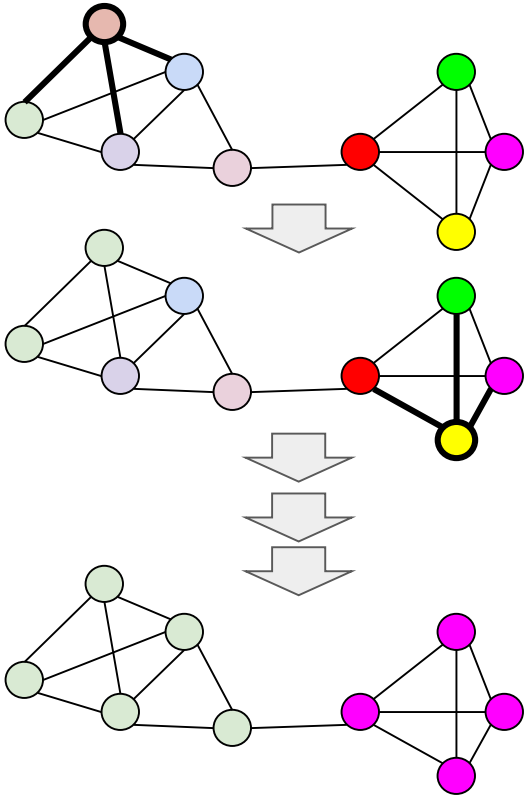


Graph-based



Graph-based - Louvain Modularity

$$Q = \frac{1}{2m} \sum_{ij} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j),$$



What does a cluster look like?

- Clusters are contiguous sets of points with higher density of edges between them than “expected”.

How many clusters are there?

- Number of clusters (k) is identified by the algorithm during optimization
- All cells are assigned to exactly one cluster

Scalability: Louvain is fast and scalable O(n log n)
Stochasticity: order of cells is stochastic

Graph-based - Louvain Modularity

$$Q = \frac{1}{2m} \sum_{ij} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j),$$

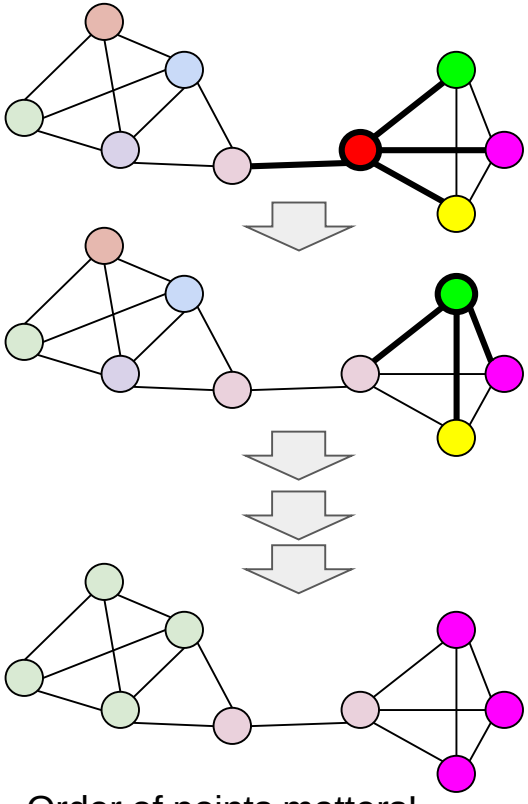
What does a cluster look like?

- Clusters are contiguous sets of points with higher density of edges between them than “expected”.

How many clusters are there?

- Number of clusters (k) is identified by the algorithm during optimization
- All cells are assigned to exactly one cluster

Scalability: Louvain is fast and scalable O(n log n)
Stochasticity: order of cells is stochastic

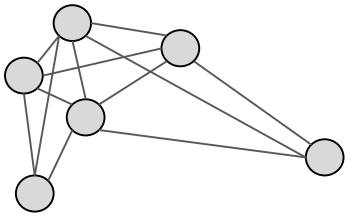


Order of points matters!

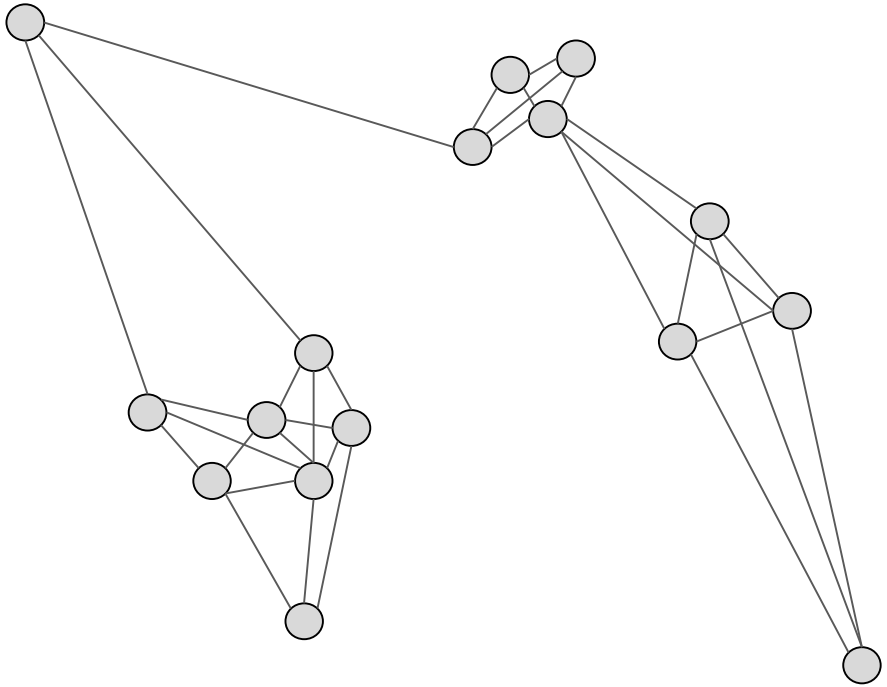
Turning scRNASeq data into a network: kNN

Connect each cell to its k-nearest neighbouring cells.

Optimal k will depend on actual cluster size and number of cells.



k = 3

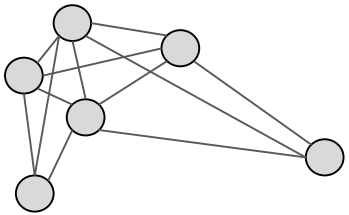


Turning scRNASeq data into a network: kNN

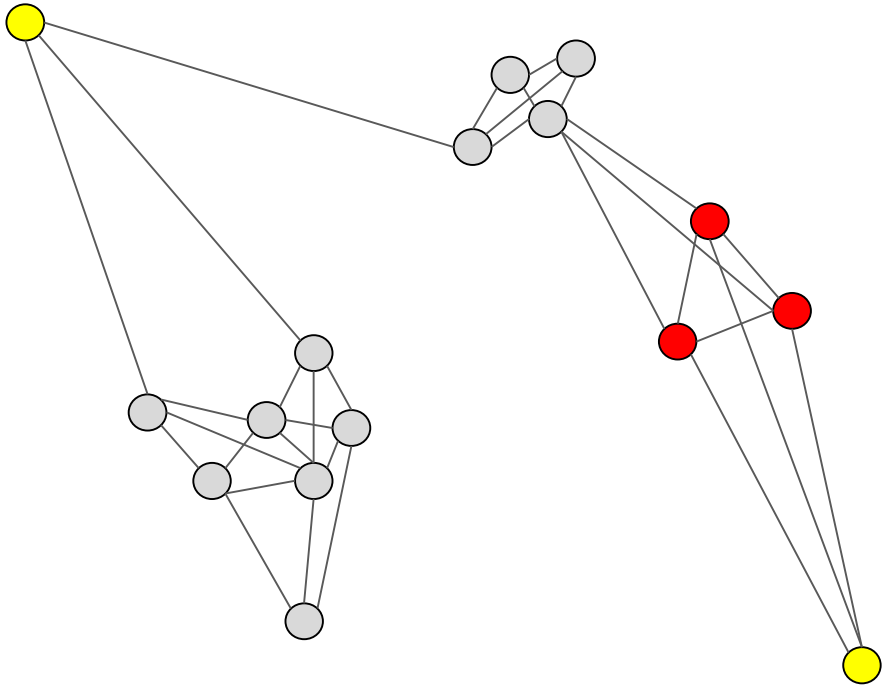
Connect each cell to its k-nearest neighbouring cells.

Optimal k will depend on actual cluster size and number of cells.

Clusters with < k+1 cells tend to merge with larger clusters, as do outliers.



k = 3

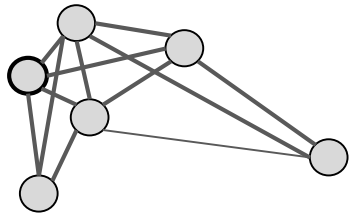


Turning scRNASeq data into a network: sNN

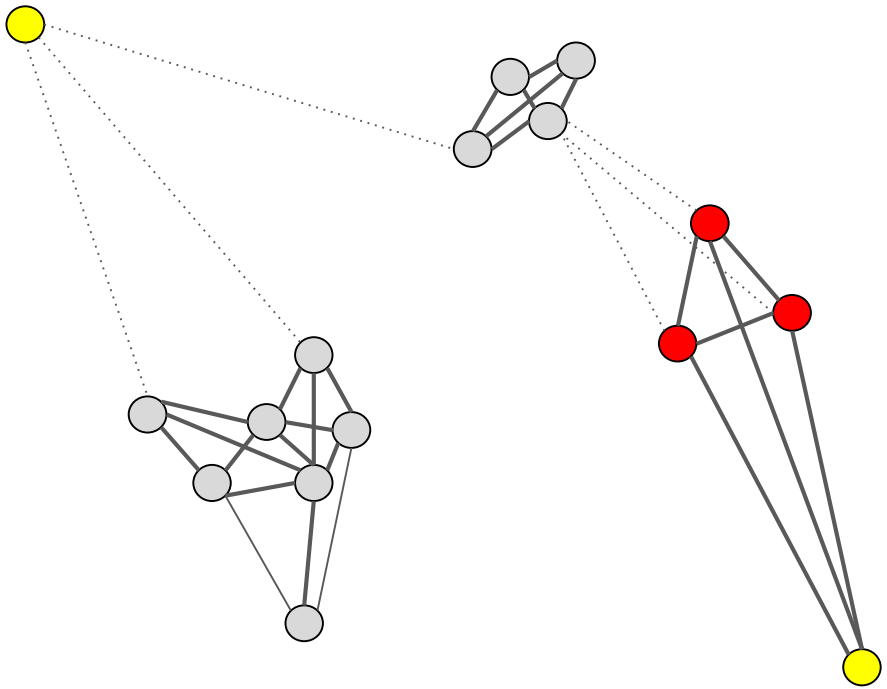
1) Connect each cell to its k-nearest neighbouring cells.

2) Weight edges between cells by the number of neighbours they have in common.

Better resolution of small clusters & outliers



k = 3

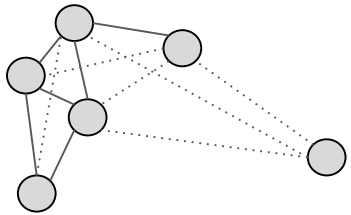


Turning scRNASeq data into a network: mNN

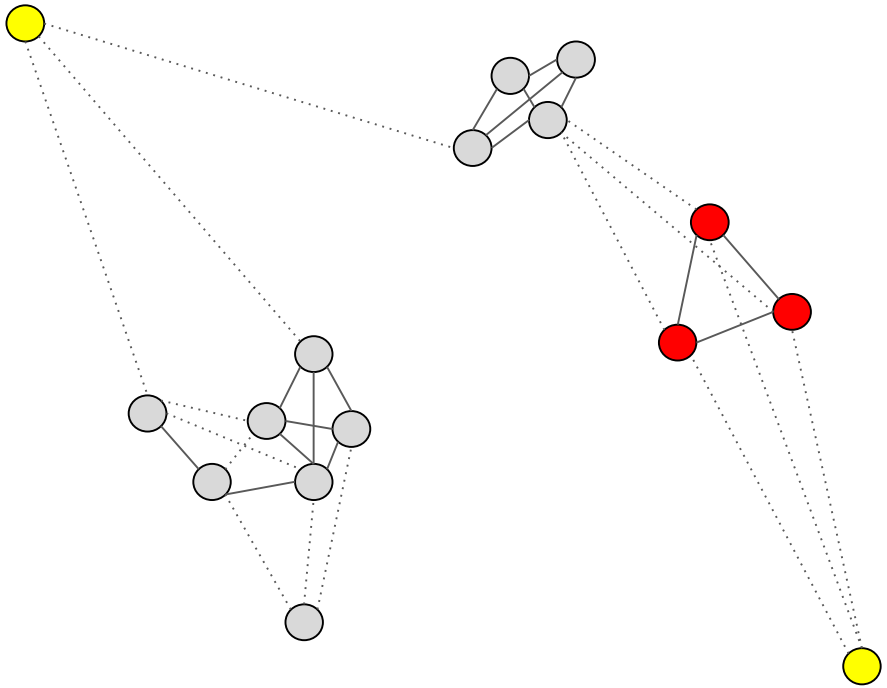
1) Connect each cell to its k-nearest neighbouring cells.

2) Remove edges which are not reciprocal i.e. A is a kNN of B but B is not a kNN of A

Better resolution of small clusters & outliers.
Produces a very sparse graph.



k = 3



Choosing K

Choosing K

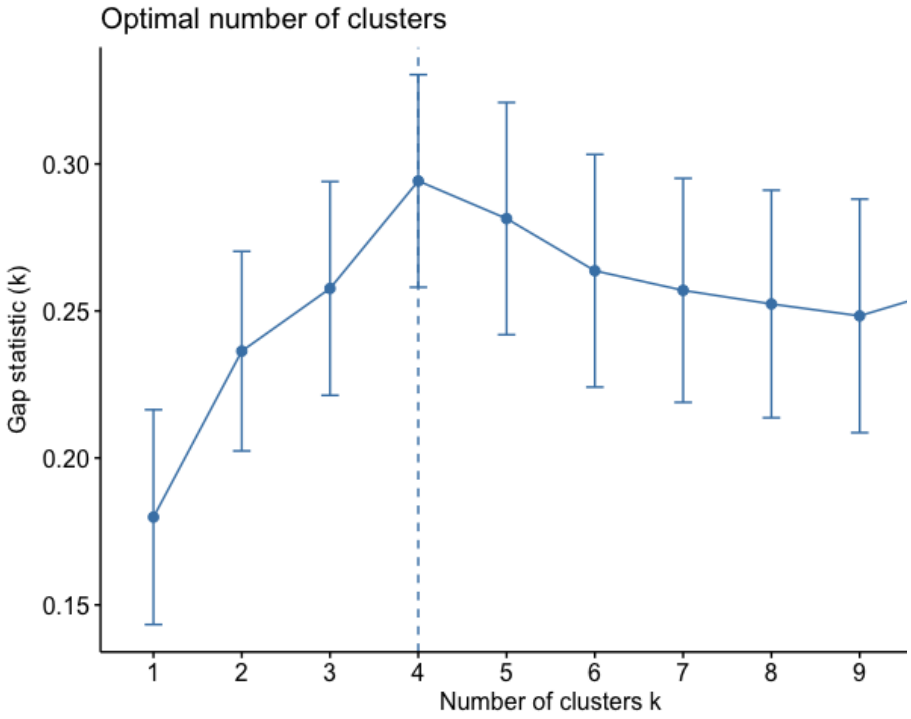
Most clustering methods have some way to tune the number of clusters directly or indirectly with parameters.

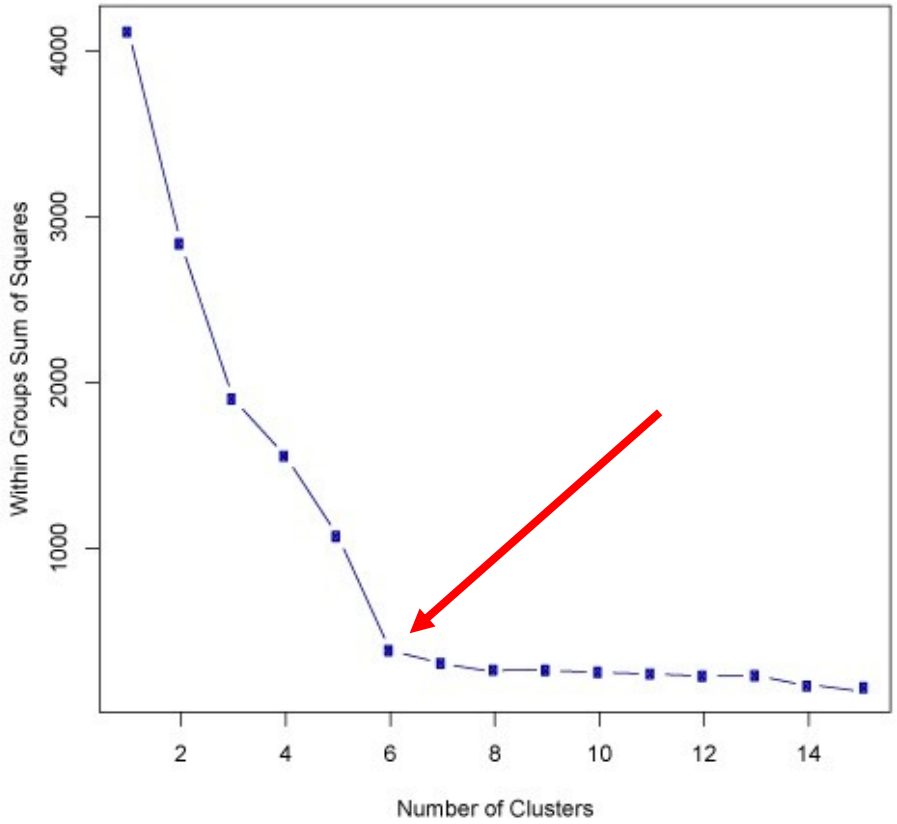
Depending on your dataset multiple different levels of K may be relevant/interesting.

Many tools will estimate a “good” number of clusters algorithmically.

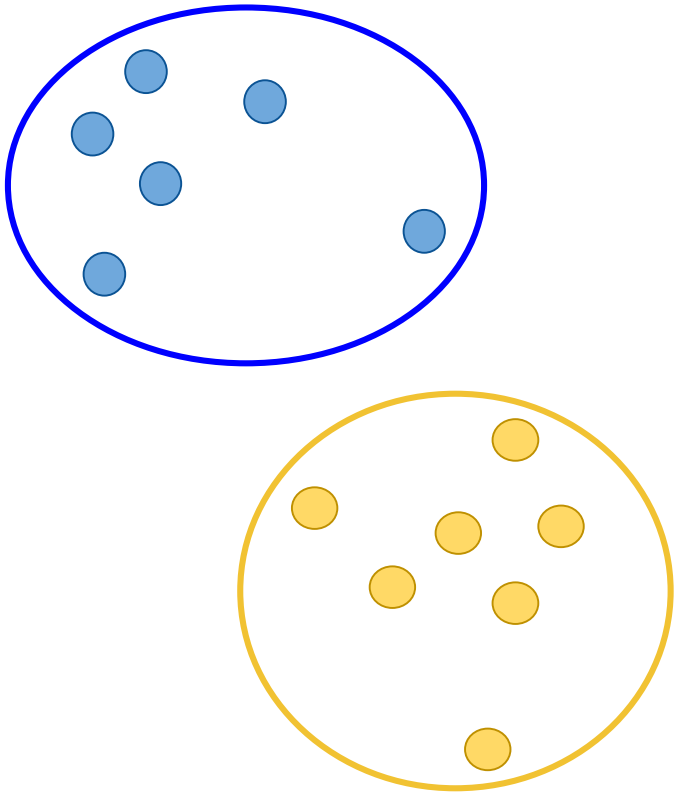
- Often this requires running the method at different granularities and considering some “quality criterion”.
- Many quality measures are included in the “clusterSim” R package.

Choosing K - Example



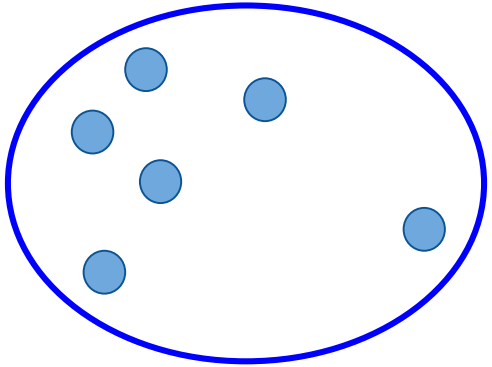


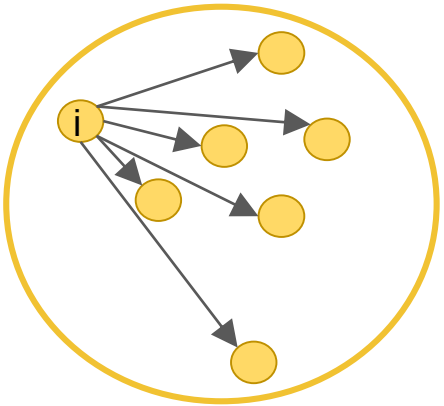
Internal Measures : Silhouette index



For each point i, let:
C_i be the cluster of which i is a member
d(i,j) be the distance from point i to point j

Internal Measures : Silhouette index

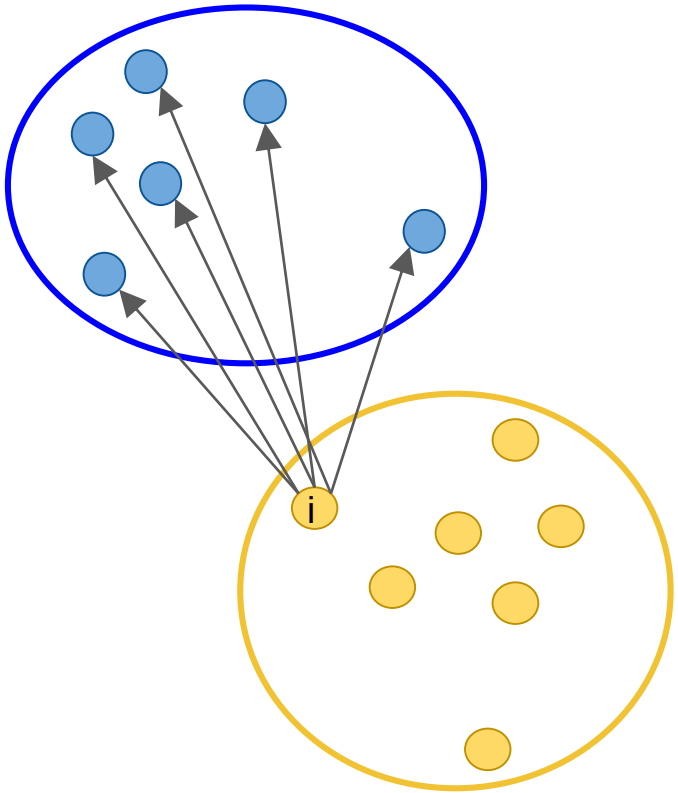




For each point i, let:
C_i be the cluster of which i is a member
d(i,j) be the distance from point i to point j

$a(i) = \text{mean}[\, d(i,j),\ j \in C_i]$

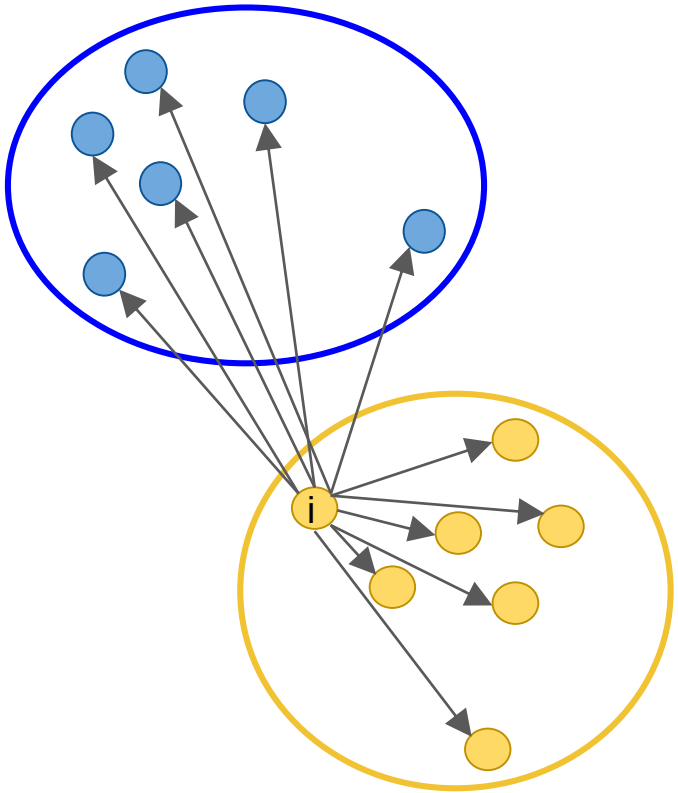
Internal Measures : Silhouette index



For each point i, let:
C_i be the cluster of which i is a member
d(i,j) be the distance from point i to point j

$a(i) = \text{mean}[\, d(i,j),\ j \in C_i]$
$b(i) = \min\_C_k \{\, \text{mean}[\, d(i,j),\ j \in C_k\,]\, \}$

Internal Measures : Silhouette index



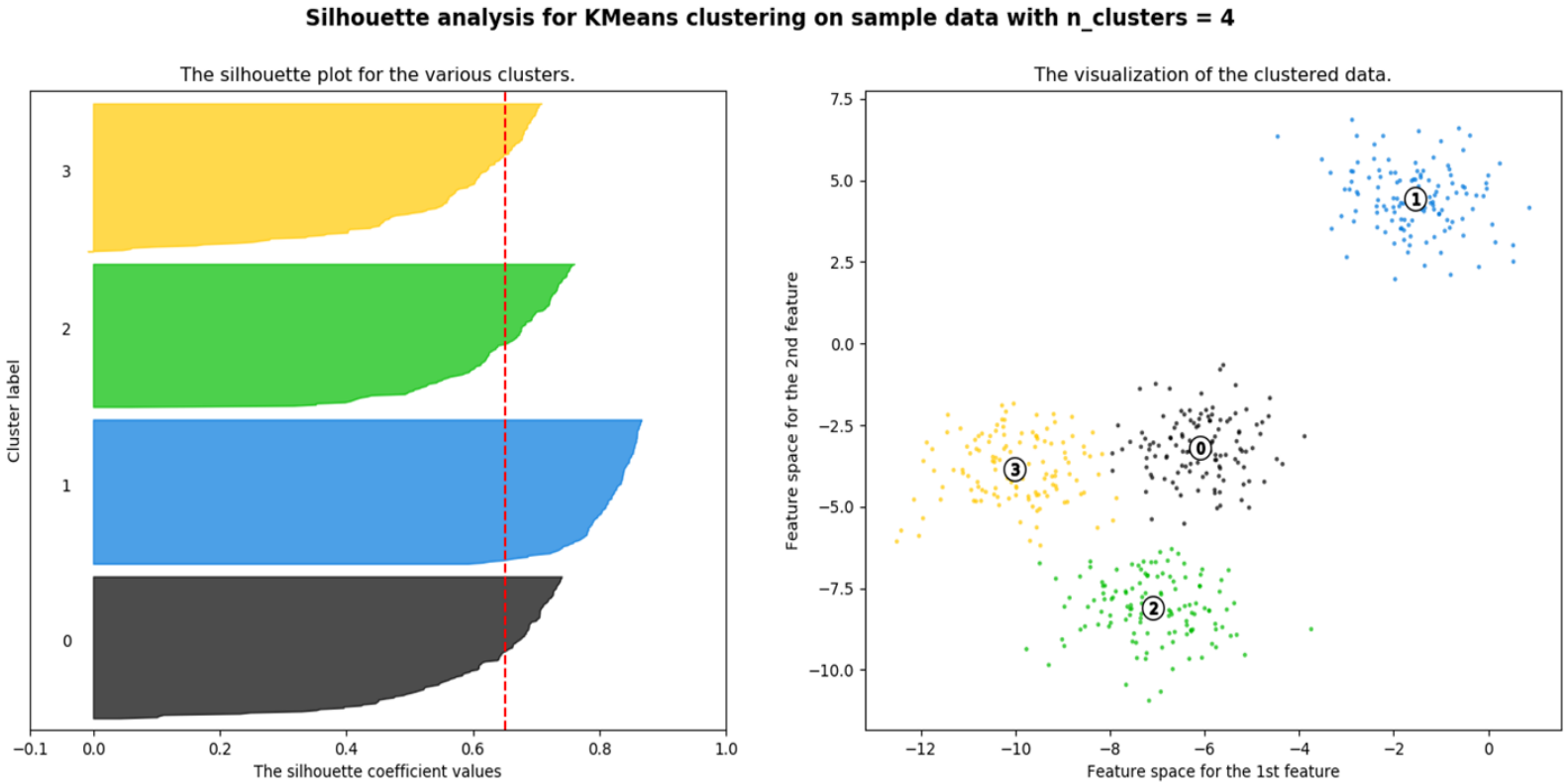
For each point i, let:
C_i be the cluster of which i is a member
d(i,j) be the distance from point i to point j

$a(i) = \text{mean}[\, d(i,j),\ j \in C_i]$
$b(i) = \min\_C_k \{\ \text{mean}[\, d(i,j),\ j \in C_k\ ]\ \}$

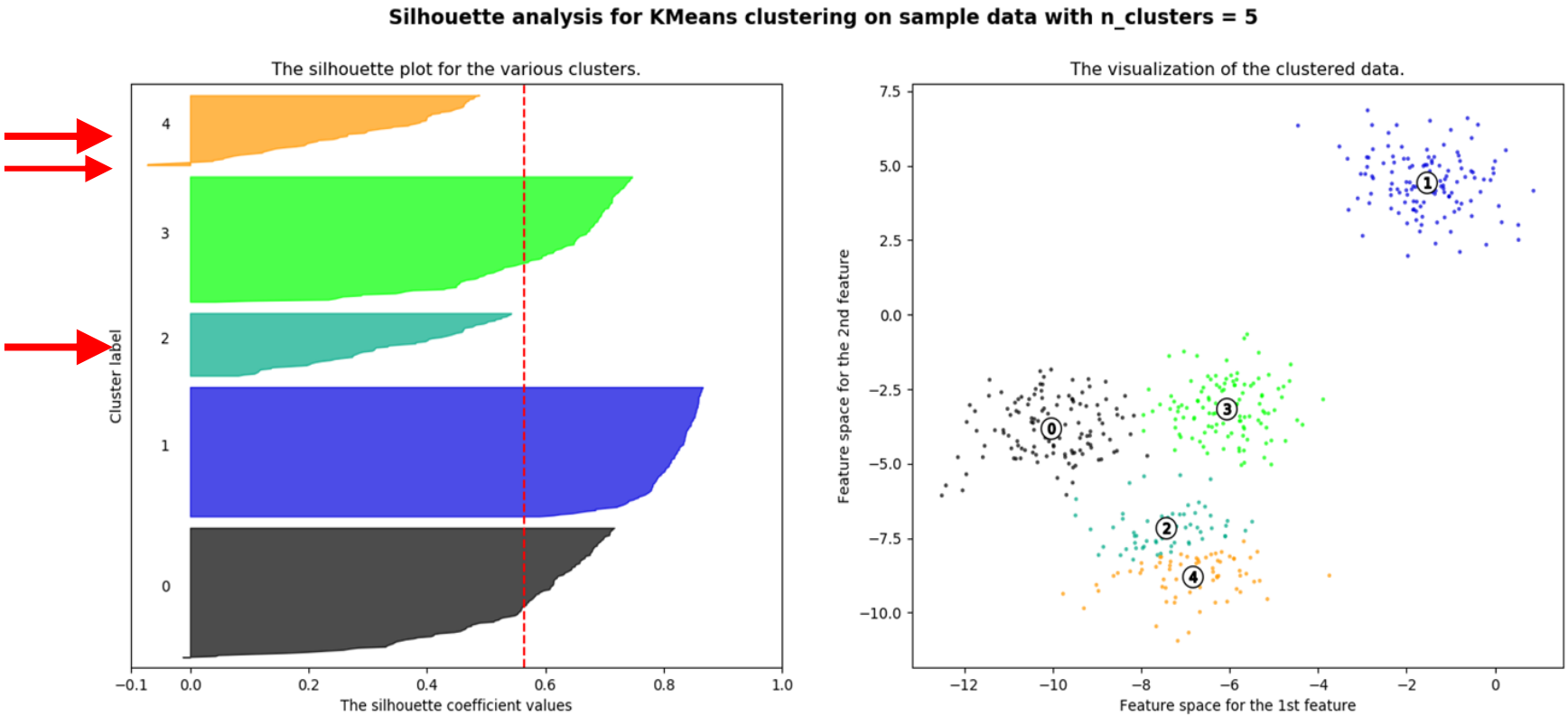
$\text{Sil} = [\ b(i) - a(i)\ ] / \max\{\ a(i),\ b(i)\ \}$

Produces both a cell-specific quality score and nearest neighbouring cluster.

Internal Measures : Silhouette index



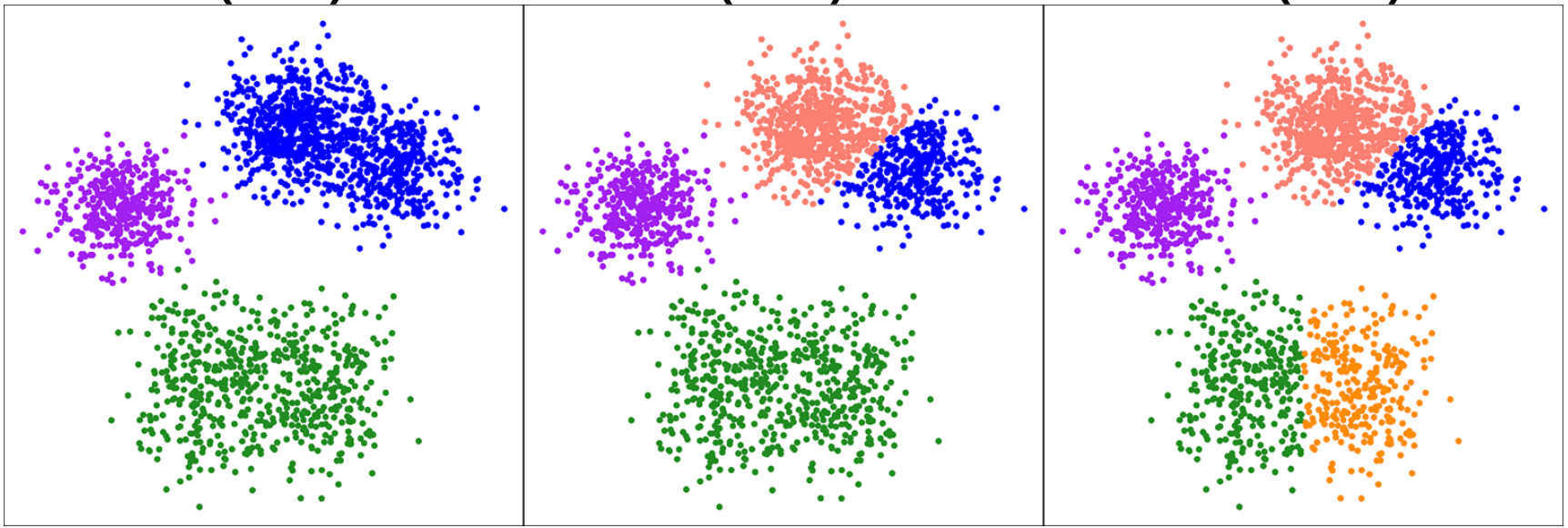
Internal Measures : Silhouette index



High Quality isn’t necessarily most accurate

Using the k with the highest silhouette index wouldn’t have found the most accurate clustering for our previous example

Sil(k = 3) **>** **Sil(k = 4)** **>** **Sil(k = 5)**



Quality of clustering: Stability

More stable the clustering the more reproducible the result is thought to be

- Not necessarily correct - technical errors or biased algorithms can produce very stable though incorrect clusterings.

Many potential perturbations:

- Change clustering parameters
- Simulate technical noise (BEARscc)
- Bootstrapping/downsampling of data
- Inherent stochasticity of the algorithm

The calculate the similarity between clusters under different perturbations.

Stability : Comparing Clusterings

To estimate the “stability” of a clustering we must have a measure to compare clustering results.

- These measures can also be used to compare a clustering to a “gold standard”

Three main classes of methods are used:

1) Counting pairs of points that agree/disagree and summarizing this into a single statistic
   e.g. Rand Index = $(N_{11}+N_{00}) / (n(n-1)/2)$
   Adjusted Rand Index (ARI) = Rand Index rescaled to be [0,1]

1) Matching clusters then calculating similarity of “matched” clusters
   e.g. van Dongen = $2n - sum_A(max_B n_{AB}) - sum_B(max_A n_{AB})$

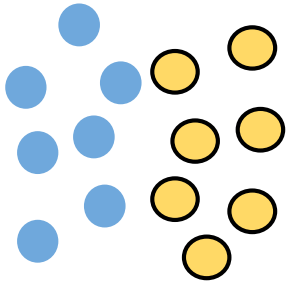
1) Information-theoretic
   e.g. Variation of Information = $H(A) + H(B) - 2I(A,B)$

Meila, 2007, Comparing clusterings -- an information based distance. Journal of Multivariate Analysis. 98(5):873-895

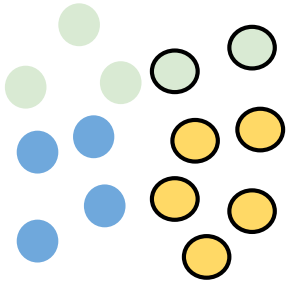
Comparing Clusterings - Cluster-specific Stability

- Used in SC3
- Cluster-specific score across a large number of clusterings

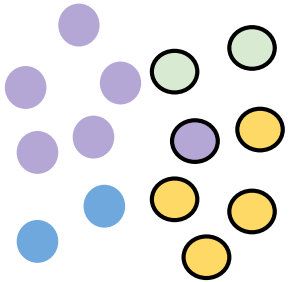
K = 2



K = 3



K = 4



Overlapping fraction

Stability of k2(yellow) =

Green: 2/5
Yellow: 5/5

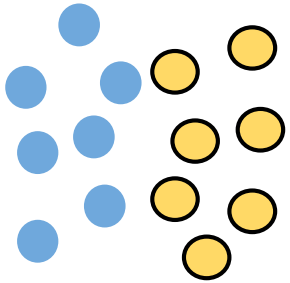
+

Green: 2/2
Purple: 1/6
Yellow: 4/4

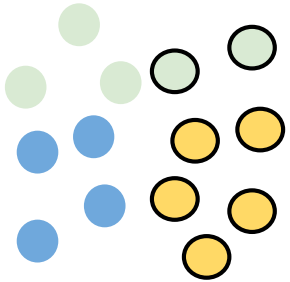
Comparing Clusterings - Cluster-specific Stability

- Used in SC3
- Cluster-specific score across a large number of clusterings

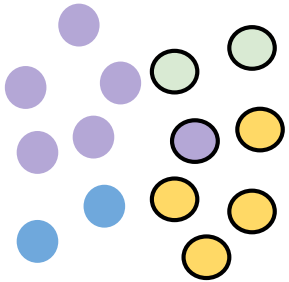
K = 2



K = 3



K = 4



Stability of k2(yellow) =

Number of overlapping clusters

Green: 2/5
Yellow: 5/5
= 7/5 / 2

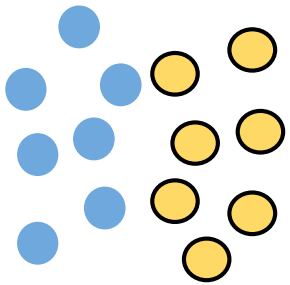
+

Green: 2/2
Purple: 1/6
Yellow: 4/4
= 13/6 / 3

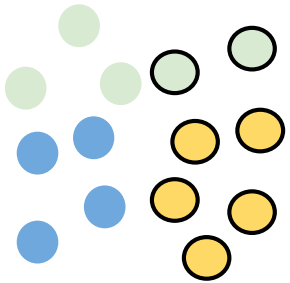
Comparing Clusterings - Cluster-specific Stability

- Used in SC3
- Cluster-specific score across a large number of clusterings

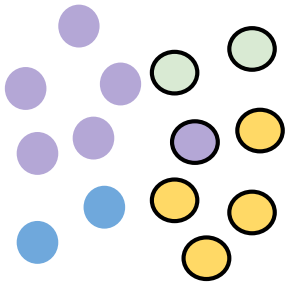
K = 2



K = 3



K = 4



Stability of k2(yellow) =

Green: 2/5
Yellow: 5/5
= 7/5 / 2 / 2
= 0.35

+

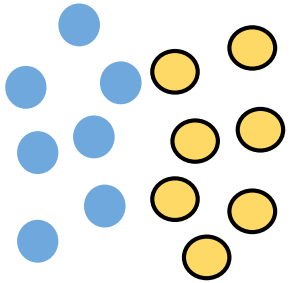
Green: 2/2
Purple: 1/6
Yellow: 4/4
= 13/6 / 3 / 3
= 0.24

Penalty for splitting the cluster

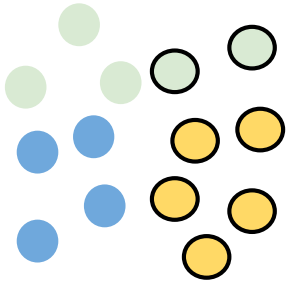
Comparing Clusterings - Cluster-specific Stability

- Used in SC3
- Cluster-specific score across a large number of clusterings

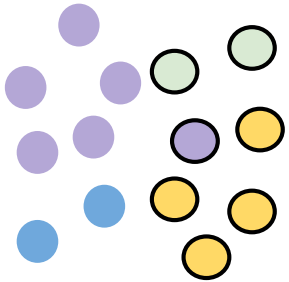
K = 2



K = 3



K = 4



Stability of k2(yellow) =

Green: 2/5
Yellow: 5/5
= 7/5 / 2 / 2
= 0.35

+

Green: 2/2
Purple: 1/6
Yellow: 4/4
= 13/6 / 3 / 3
= 0.24

= 0.59 / 2 = 0.295

Number of clusterings

Differential Expression and Markers

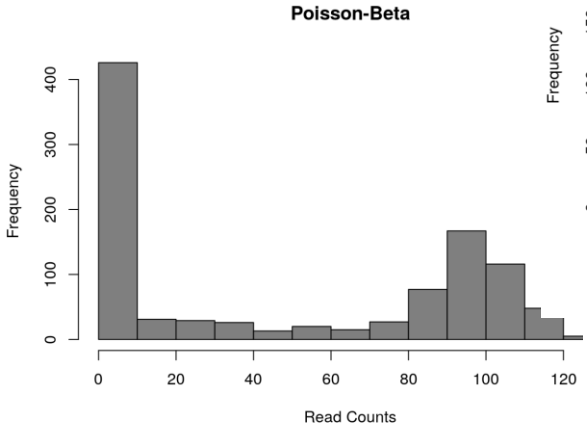
DE/Marker Genes

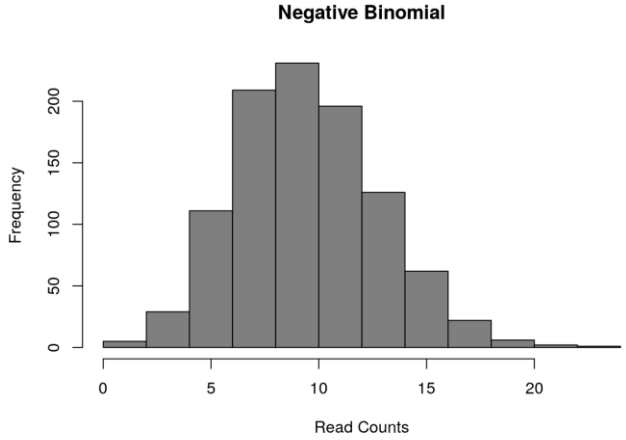
Cluster1	Cluster2	Cluster3	Cluster4	Gene type
High	Low	Low	Low	Positive Marker
High	High	Low	High	Negative Marker
High	Medium	High	Low	Differentially Expressed
High	Low	Low	High	Differentially Expressed
Medium	Medium	Medium	Medium	

Parametric

Three main parametric models:

- Negative binomial (edgeR, DESeq, Monocle)
- Zero-inflated negative binomial (MAST, SCDE)
- Poisson-Beta (BPSC)





Negative Binomial

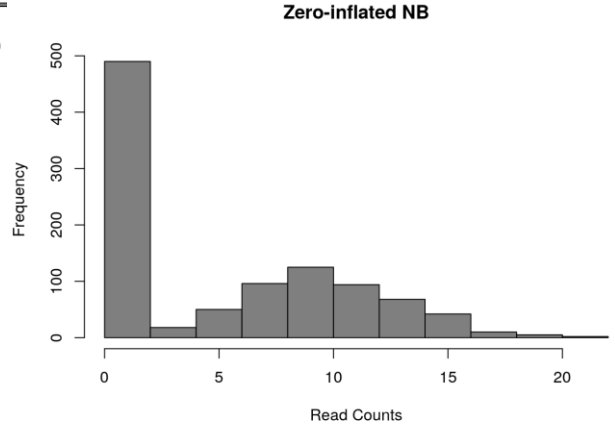
Mean: μ

Variance: $\sigma^2 = \mu + \mu^2/r$

ZINB

Mean: $\mu*(1-d)$

Variance: $\sigma^2 = \mu*(1-d)*(1+d*\mu + \mu/r)$

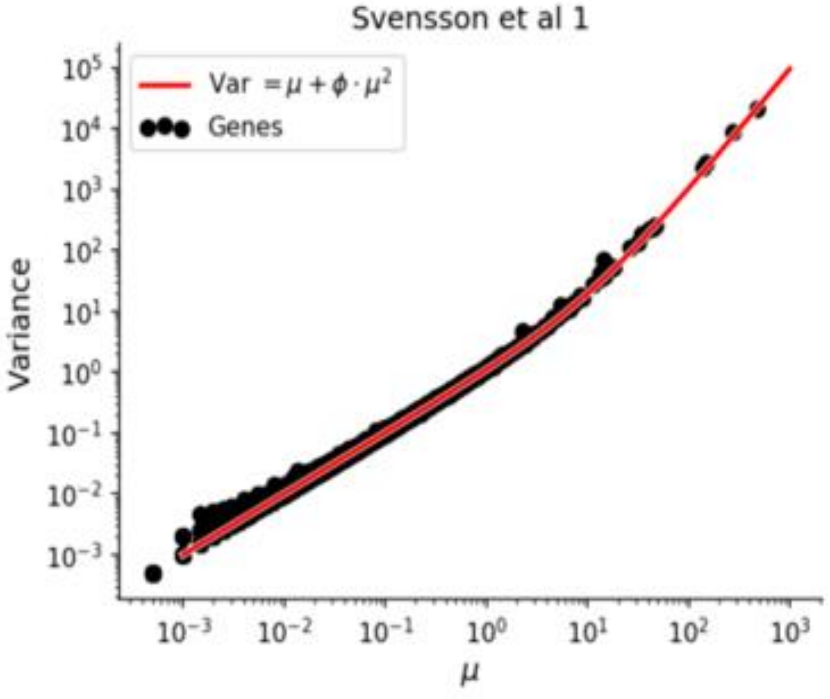


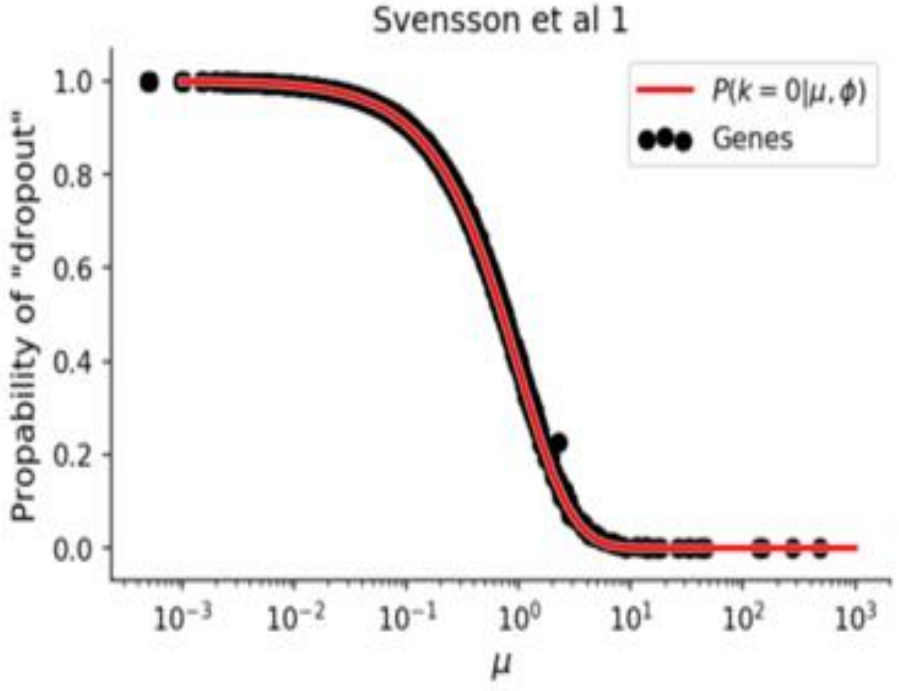
Poisson-Beta

Mean: g*a/(a+b)

Variance: $\sigma^2 = g^2*a*b/((a+b+1)*(a+b)^2)$

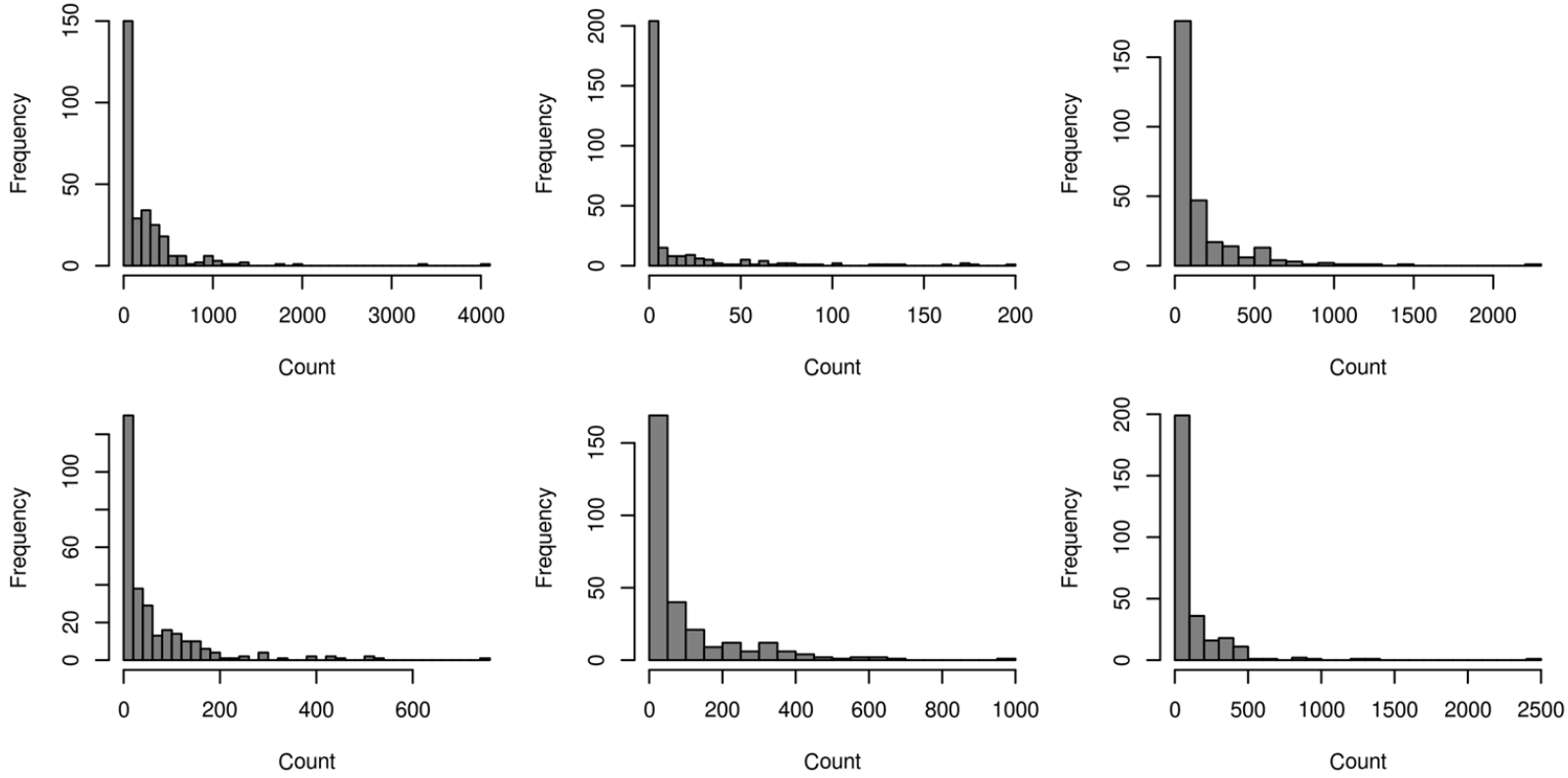
Droplet-UMI data follows a negative binomial





Svensson, 2017, http://www.nxn.se/valent/2017/11/16/droplet-scrna-seq-is-not-zero-inflated

Full Transcript data is zero-inflated



GLM-based

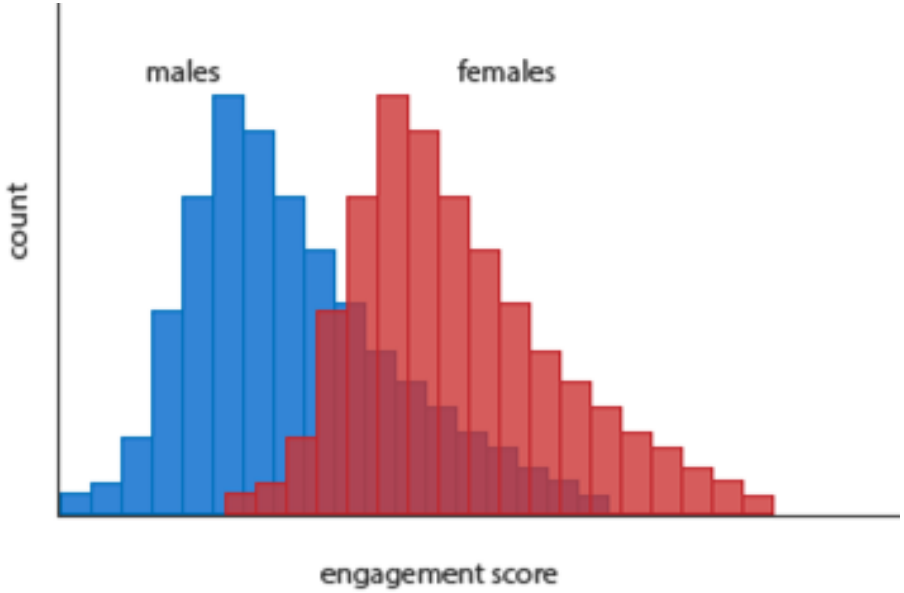
General Linear Models are flexible regression models:

- Error distributions other than normal (e.g. Poisson, Negative Binomial)
- Combine maximum likelihood and least squares to estimate model parameters

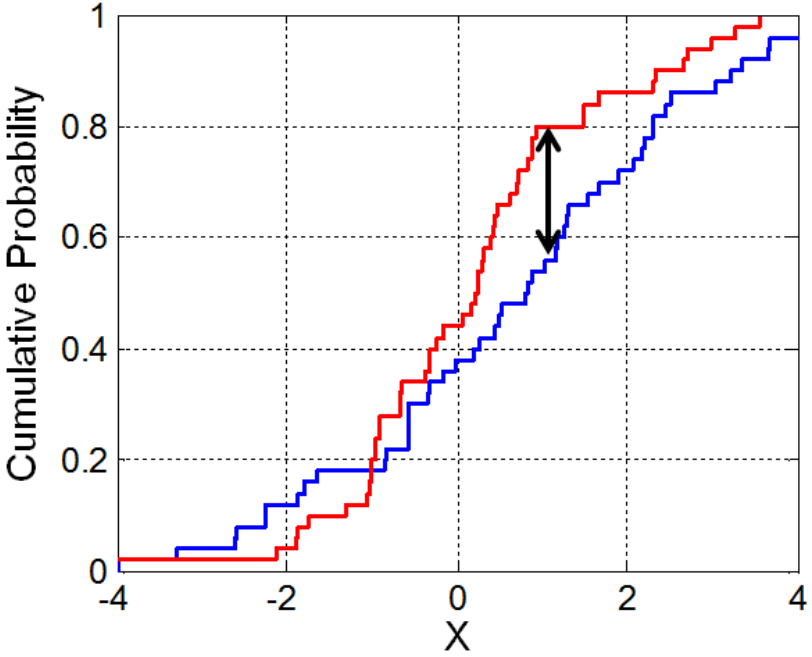
Response variables (i.e. expression level) can be modelled as any linear combination of predictors

- Predictors can be:
  - discrete, e.g. batch, cell-type, condition
  - continuous, e.g. pseudotime, library size, number of detected genes
- May include interaction terms :
  - e.g. interaction between biological condition and cell-type
- A “link” function describes how the linear combination of predictors affects the response variable
  - Linear : e.g. increase predictor by 2 = increase response by 2
  - Log-linear : e.g. increase predictor by 2 = double response
  - Log-odds (logit) : e.g. increase predictor by 2 = double the odds of the response

Non Parametric



Wilcox/Mann-Whitney U



K-S test

Analysis of complex experiments

How would you test the following questions? (Hint: there is more than one good answer)

1) Effect of stimulation on gene expression in three batches of T-cells, each batch contains 50% stimulated and 50% unstimulated.

2) Effect of a CRISPR knockout in a single lane of 10X from a mixed population of with/without gRNA.

3) Cell-type specific effects of diabetes in the pancreas, three replicates were performed for diabetic and non-diabetic pancreas samples.

Summary

High dimensionality of scRNASeq data makes clustering cells difficult

- Feature selection both reduces dimensionality and improve signal:noise
- Dimensionality reduction highlights specific aspect of the data

tSNE should only be used for visualization

- Densities, sizes, and relationships between clusters are systematically distorted

Many clustering algorithms exist which can be applied to scRNASeq

- Different algorithms make different assumptions about the structure of the data
- Graph-based methods can be used on nearest-neighbour networks between cells
- All have parameters which affect the number/size of clusters identified

Quality metrics and evaluating stability can help identify the appropriate number of clusters

- Depending on the data multiple levels of clustering might exist

Identifying DE / Marker genes helps interpretation / validation of clusters

Validation / Follow-up

Reproducibility

- Find similar/same clusters in other datasets (cross-dataset comparison)
- Reproduce marker co-expression using other technique (RT-qPCR, flow cytometery, FISH)
- Cross-species comparisons
- Reproduce in different conditions (healthy/diseased)

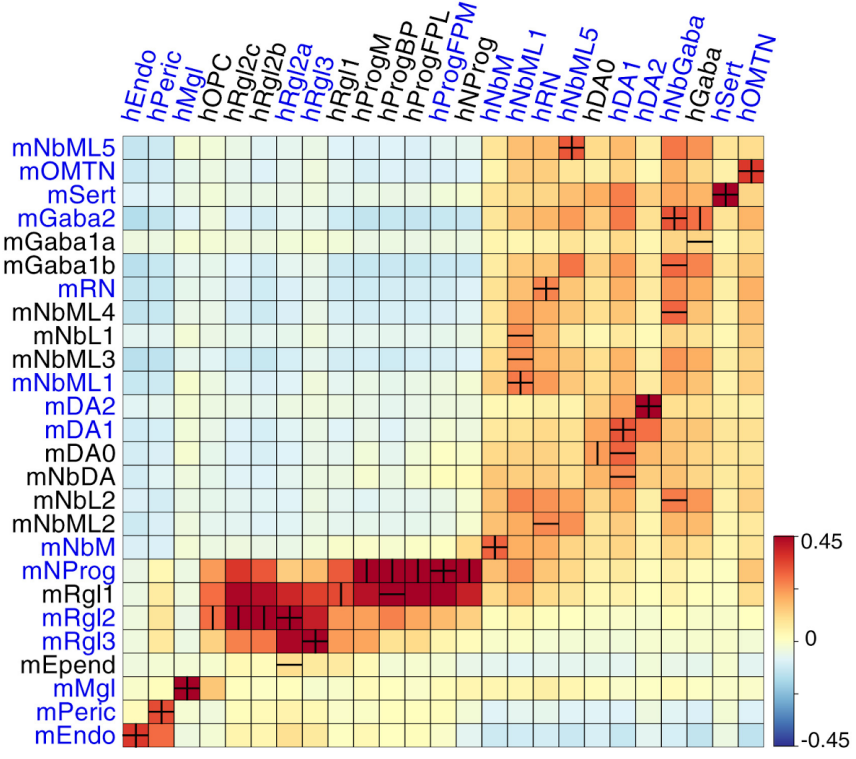
Functional assays

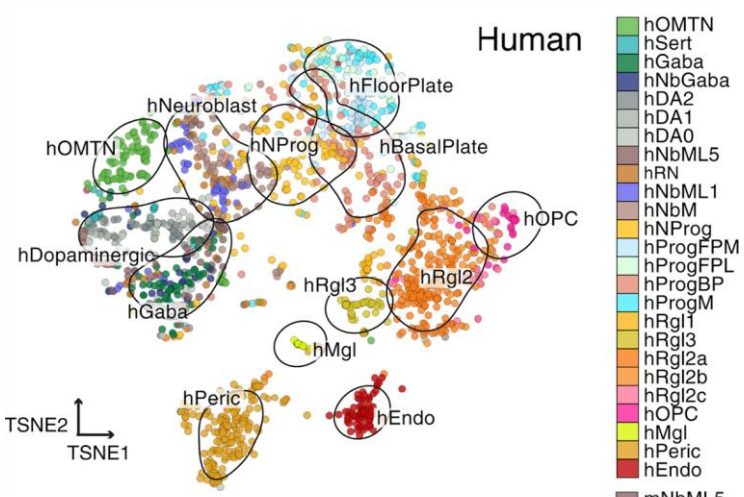
- Rely on surface markers for isolation of specific cell populations
- Knockout studies (depends on detecting regulatory TFs)

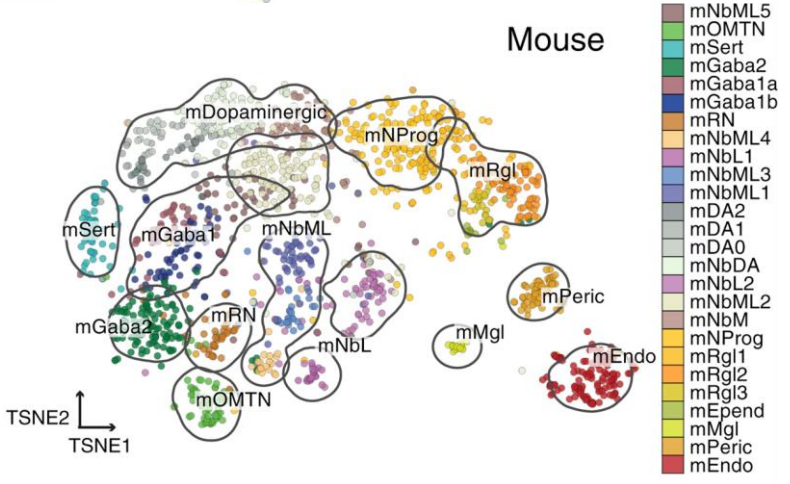
Further Characterization

- Morphology, spatial location

Cross-species comparison

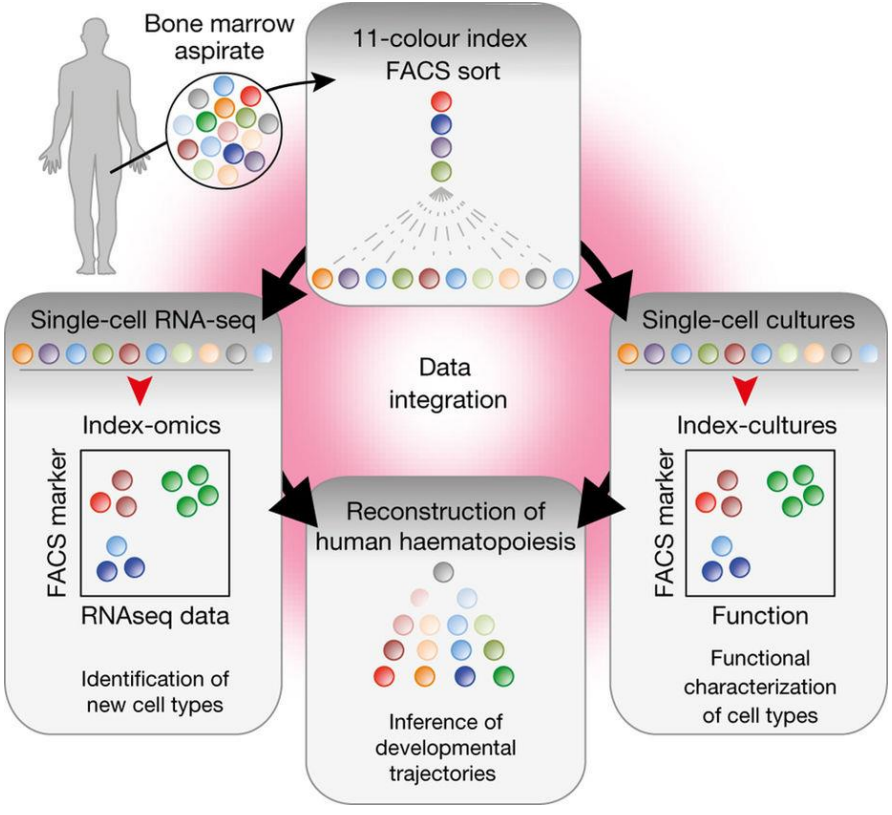


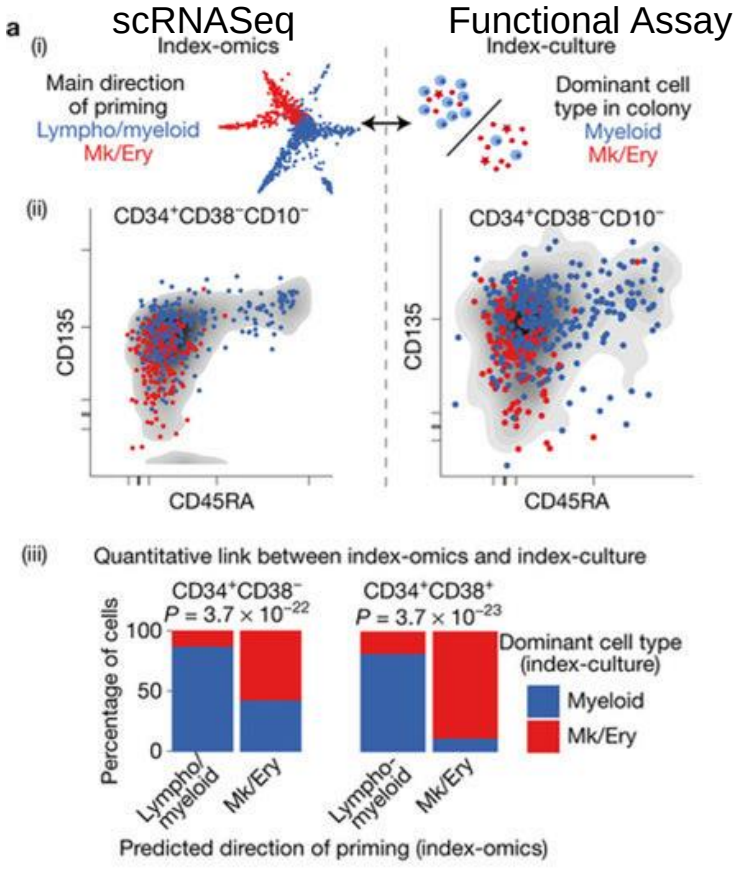




Manno, 2016, Cell, doi: 10.1016/j.cell.2016.09.027

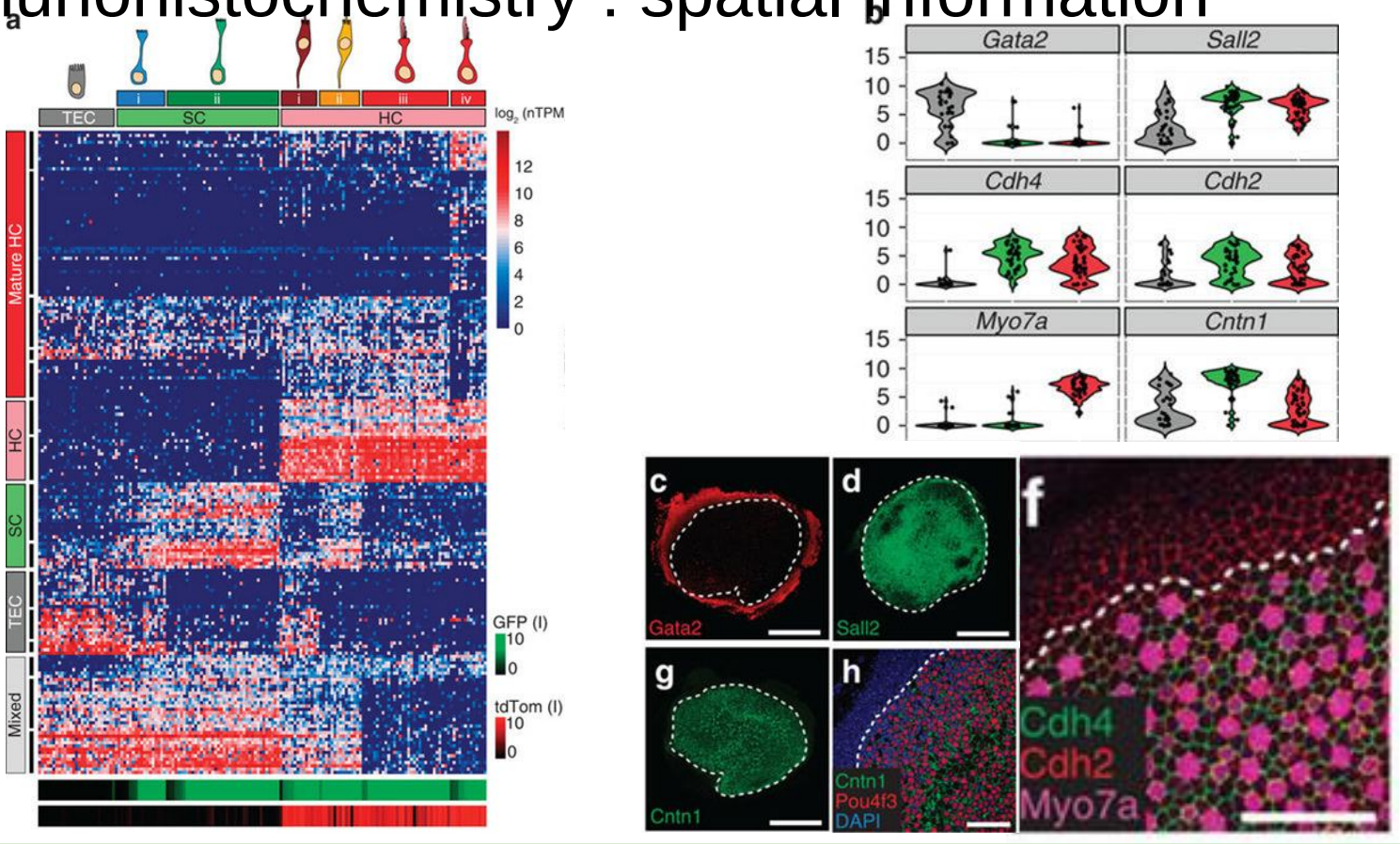
Parallel functional assay & scRNASeq





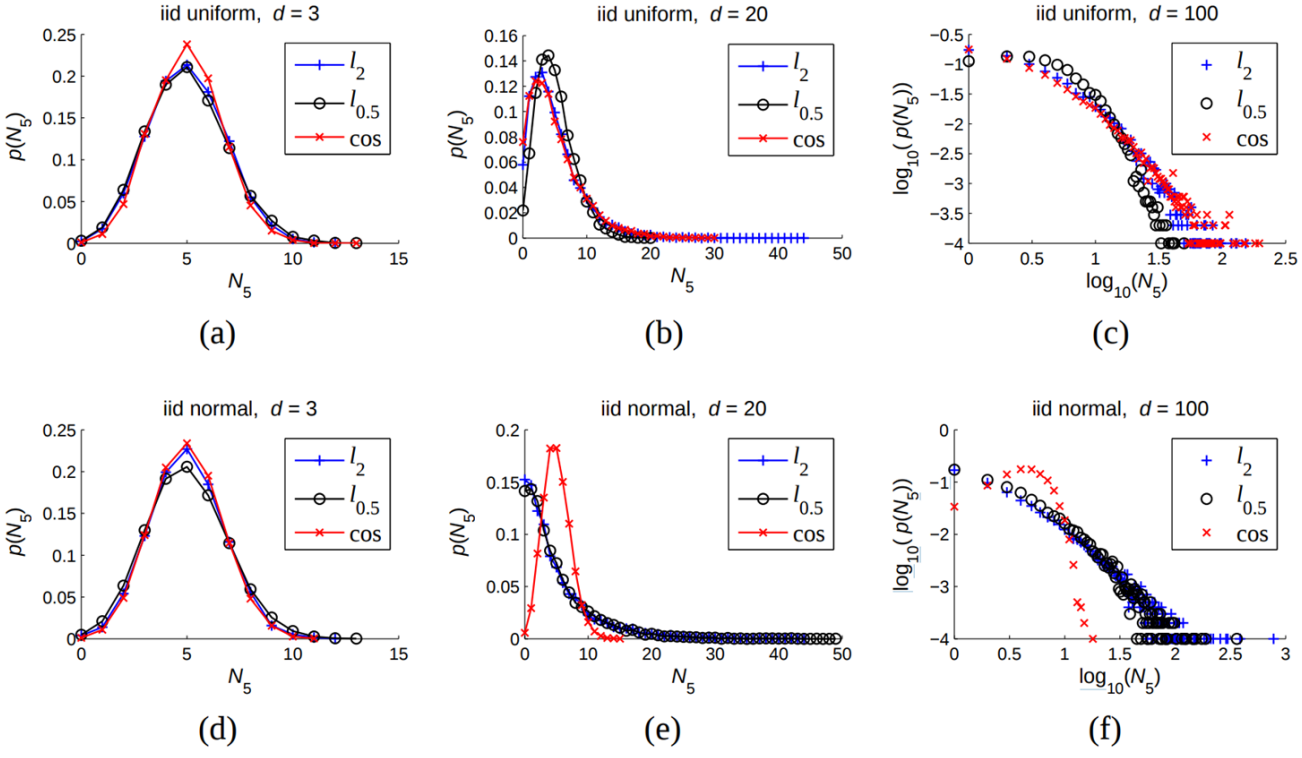
Velten, 2017, Nature Cell Biology, doi:10.1038/ncb3493

Immunohistochemistry : spatial information

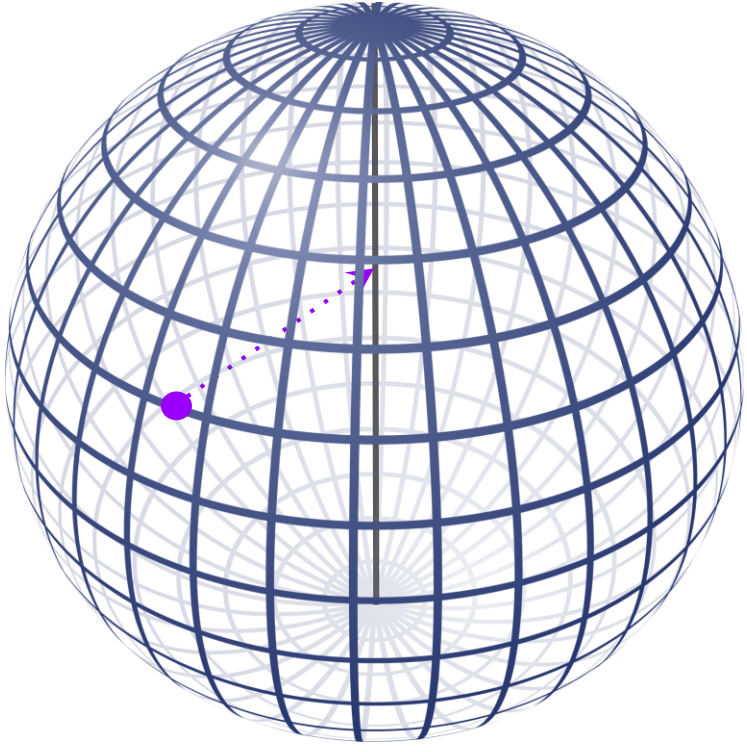


Burns, 2015, Nature Communications, doi: 10.1038/ncomms9557

k-NN graphs in High Dimension

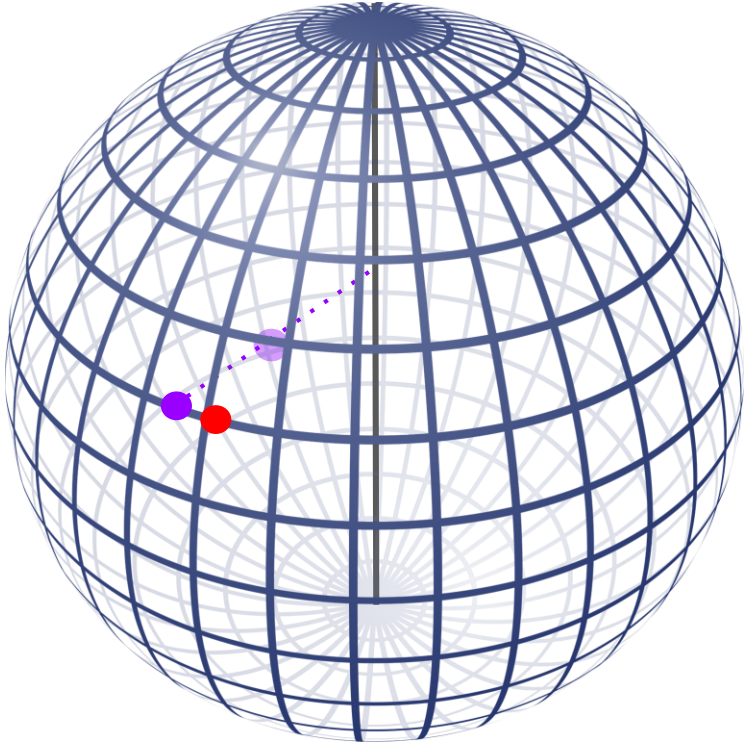


k-NN graphs in High Dimension



Radovanovic, 2010, Journal of Machine Learning Research

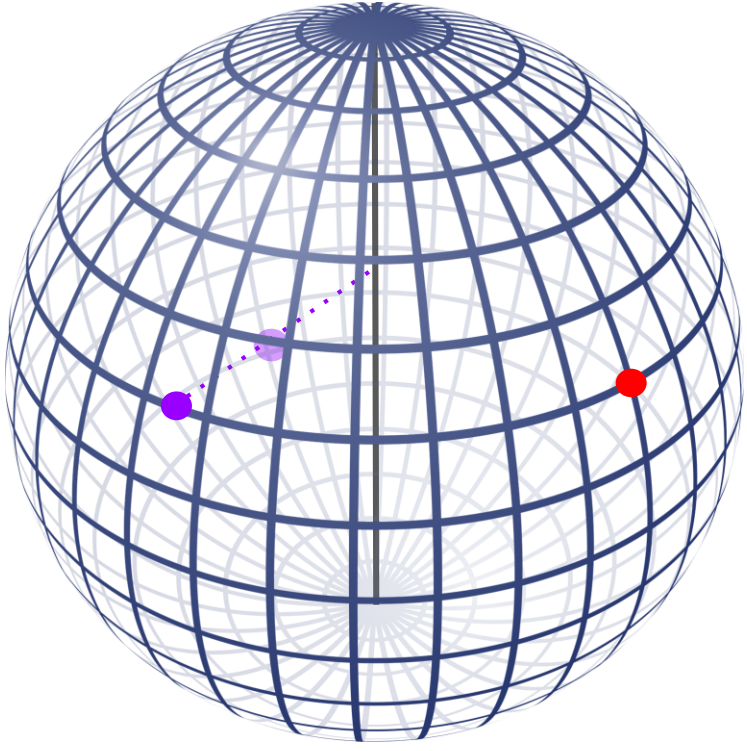
k-NN graphs in High Dimension



Purple -> center
d(Purple, red) increases

Radovanovic, 2010, Journal of Machine Learning Research

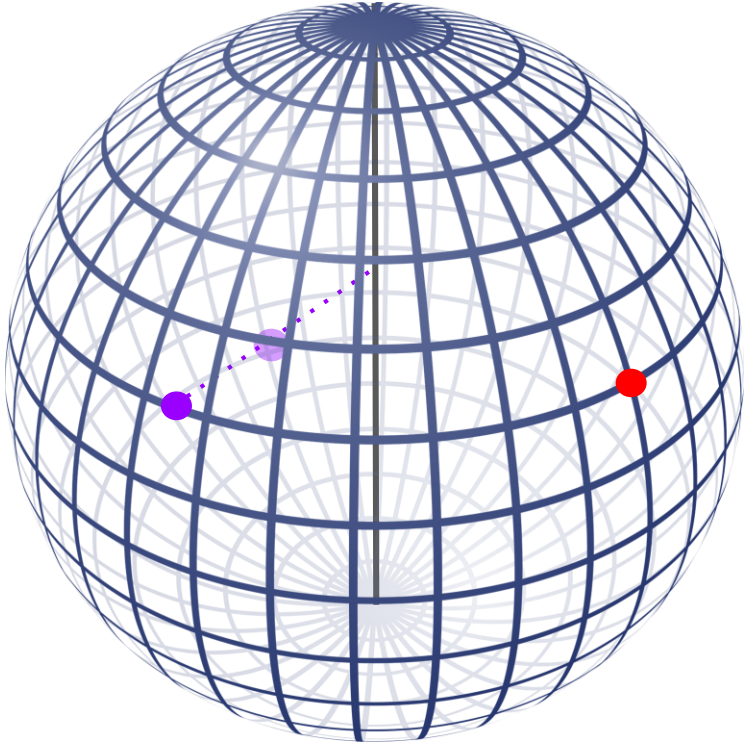
k-NN graphs in High Dimension



Purple -> center
d(Purple, red) decreases

Radovanovic, 2010, Journal of Machine Learning Research

k-NN graphs in High Dimension



As dimensions increase:

- data gets more sparse

- points closer to the centre become relatively closer to a larger number of points

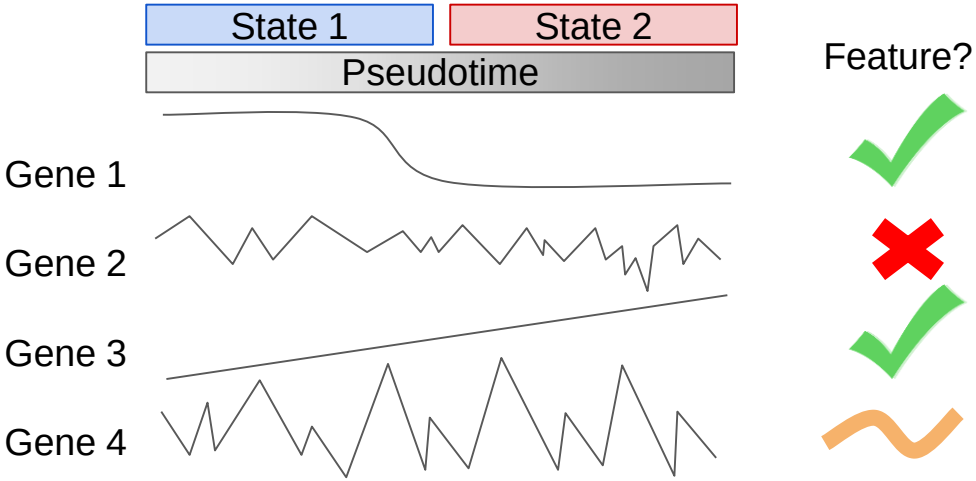
- k-NN networks become more skewed

Radovanovic, 2010, Journal of Machine Learning Research

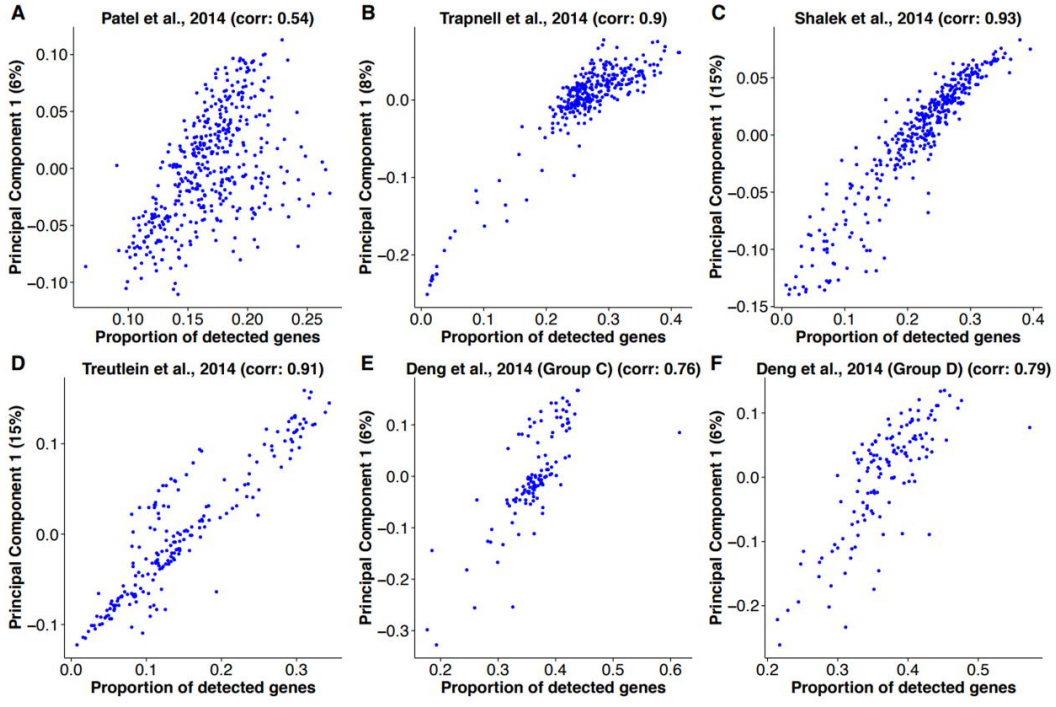
Supervised Feature Selection

One way to reduce the dimensionality of our data is to identify and remove “dimensions” (i.e. genes) which don’t contain any biological signal.

(1) If “true” biological structure is known, then genes with expression agreeing with that structure can be selected (e.g. DE).



PCA: Sensitive to zeros



Apparent if using raw counts.

Log-transformed normalized data can be generally be used in regular PCA.

Factor analysis, such as ZINB-WaVE, can be used to perform similar decomposition accounting for technical covariates.

Hicks, 2015, biorxiv