

Biostatistics-Lecture 3

Ruibin Xi

Peking University

School of Mathematical Sciences

Supervised versus unsupervised

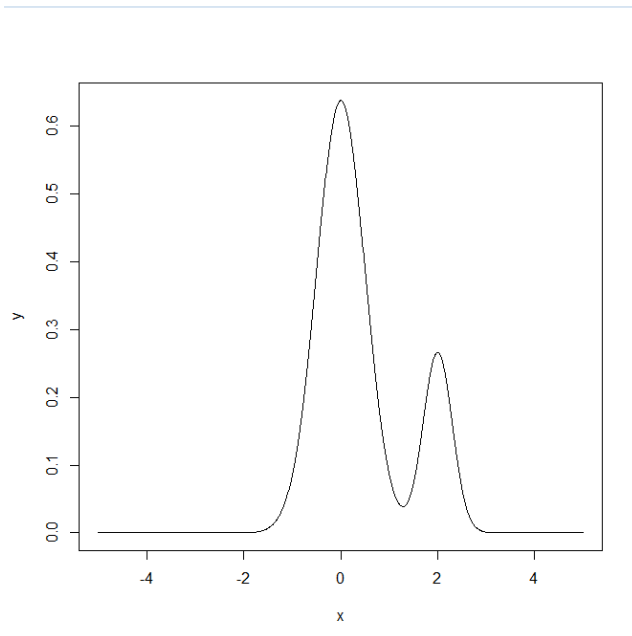
- Supervised
 - You have an outcome Y
 - and some covariates X You typically want to solve something like $\operatorname{argminf} E[(Y-f(X))^2]$
- Unsupervised
 - You have a bunch of observations X
 - and you want to understand the relationships between them. You are usually trying to understand patterns in X or group the variables in X in some way

Techniques for unsupervised analysis

- Kernel density estimation
- Clustering
- Principal components analysis/svd
- MDS/Isomap/diffusion map

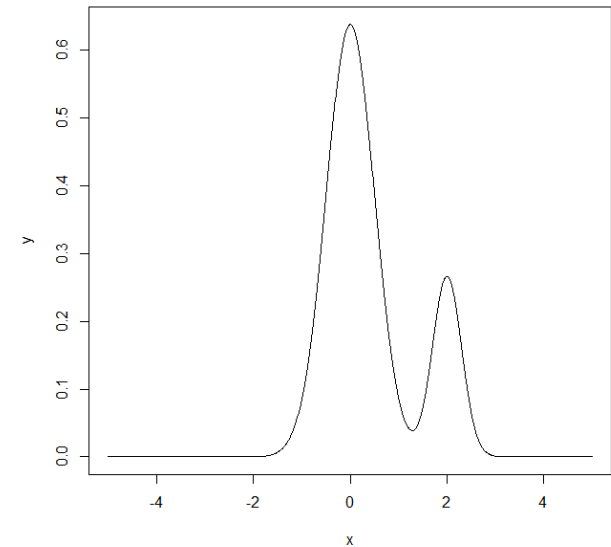
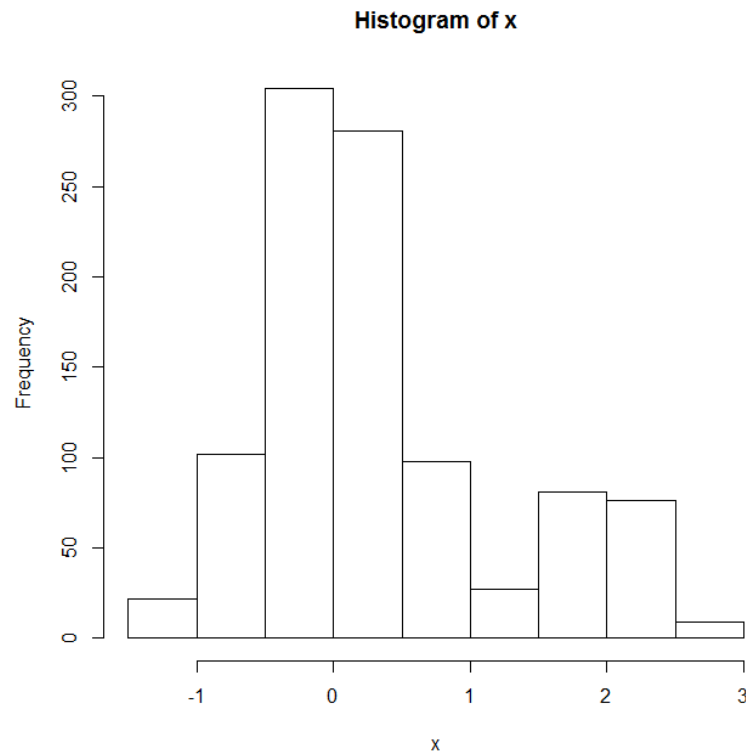
Estimating a univariate density

- Suppose that $X_1, \dots, X_n \sim F$ with the density $f(\cdot)$
- How to given an estimate \hat{f} of the density?



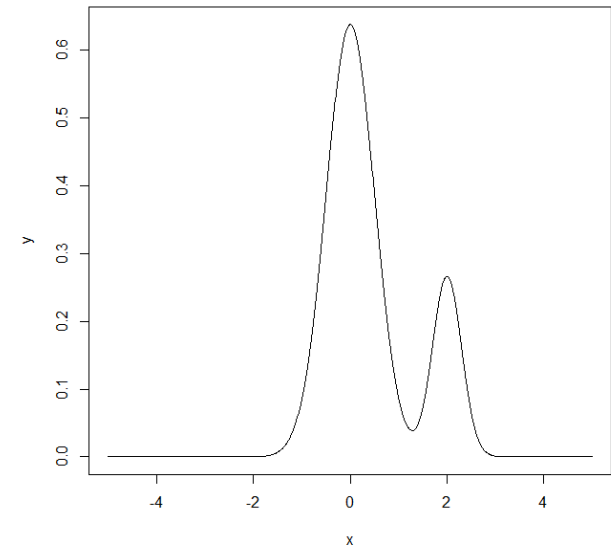
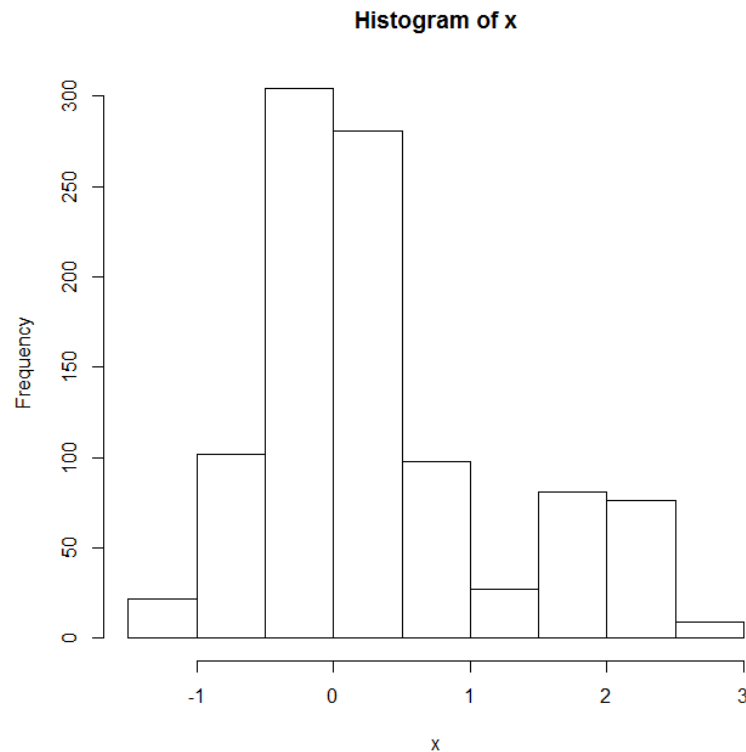
Estimating a univariate density

- You have seen the histogram



Estimating a univariate density

- You have seen the histogram



How to give a smooth estimate of the density function?

Binning

- Suppose that $X_1, \dots, X_n \sim F$ with the density $f(\cdot)$
- Bin the data; In math this is

$$I_j = (x_0 + j \times h, x_0 + (j + 1) \times h], j = -1, 0, 1, \dots$$

- Calculate the counts in bins

$$C_j = \sum_{i=1}^n I(x_i \in I_j)$$

- Parameters are x_0, h

Binning

- We may use the following to estimate $f(\cdot)$

$$\hat{f}(x) = \frac{1}{2hn} \# \{i; X_i \in (x - h, x + h]\}$$

- This can be viewed as an approximation of the density

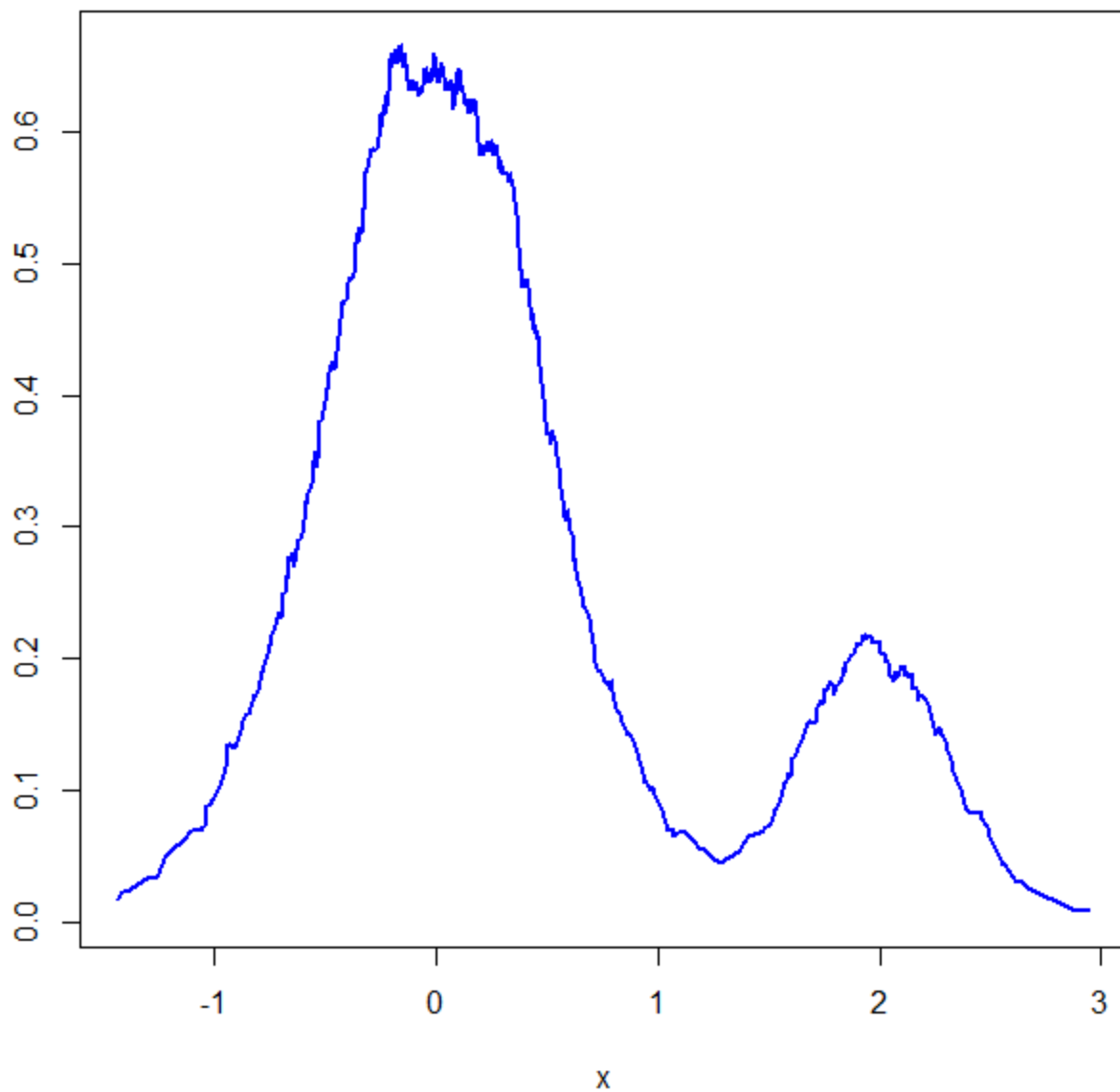
$$f(x) = \lim_{h \rightarrow 0} \frac{1}{2h} \mathbb{P}[x - h < X \leq x + h]$$

- This just replaces the theoretical expectation to with their empirical counterparts

- We I

- This dens

- This with



the

n to

The kernel density estimator

- The formula $\hat{f}(x) = \frac{1}{2hn} \#\{i; X_i \in (x - h, x + h]\}$ may be written as

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^n w\left(\frac{x - X_i}{h}\right)$$

$$w(x) = \begin{cases} 1/2 & \text{if } |x| < 1 \\ 0 & \text{otherwise} \end{cases}$$

- In general, you can write a kernel smoother as:

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x - X_i}{h}\right)$$

where $\int K(x)dx = 1$ (this guarantees that $\int \hat{f}(x)dx = 1$)

The kernel density estimator

- The formula $\hat{f}(x) = \frac{1}{2hn} \#\{i; X_i \in (x - h, x + h]\}$ may be written as

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^n w\left(\frac{x - X_i}{h}\right)$$

$$w(x) = \begin{cases} 1/2 & \text{if } |x| < 1 \\ 0 & \text{otherwise} \end{cases}$$

- In general, you can write a kernel smoother as:

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x - X_i}{h}\right)$$

where $\int K(x)dx = 1$ (this guarantees that $\int \hat{f}(x)dx = 1$)

h is the bandwidth

Kernel and bandwidth

- The bandwidth can be chosen in a large number of ways
- Typically it is automatically chosen (e.g. in statistical software)
- Popular kernels add more weight to nearby points

- Gaussian
$$K_{\lambda}(x_0, x_i) = D\left(\frac{|x_0 - x_i|}{\lambda}\right) \quad D(t) = (2\pi)^{-1/2} e^{-t^2/2}$$

- Tukey Biweight


$$K_{\lambda}(x_0, x_i) = D\left(\frac{|x_0 - x_i|}{\lambda}\right) \quad D(t) = (1 - t^2)^2 \text{ if } t \leq 1$$

- See more in [Density Estimation for Statistics and Data Analysis](#) By Silverman

Bias variance tradeoff

- We often want to minimize

$$\begin{aligned}MSE(x) &= \mathbb{E} \left[\left(\hat{f}(x) - f(x) \right)^2 \right] \\&= \left(\mathbb{E}[\hat{f}(x)] - f(x) \right)^2 + \text{Var}(\hat{f}(x))\end{aligned}$$

bias 

- The bias of \hat{f} increases and the variance of \hat{f} as h increases. This is the bias-variance tradeoff .
- The best h is $O(h^{-1/5})$. Also see [here](#).

Clustering

- Clustering organizes things that are close into groups
 - How do we define close?
 - How do we group things?
 - How do we visualize the grouping?
 - How do we interpret the grouping?

Hierarchical clustering

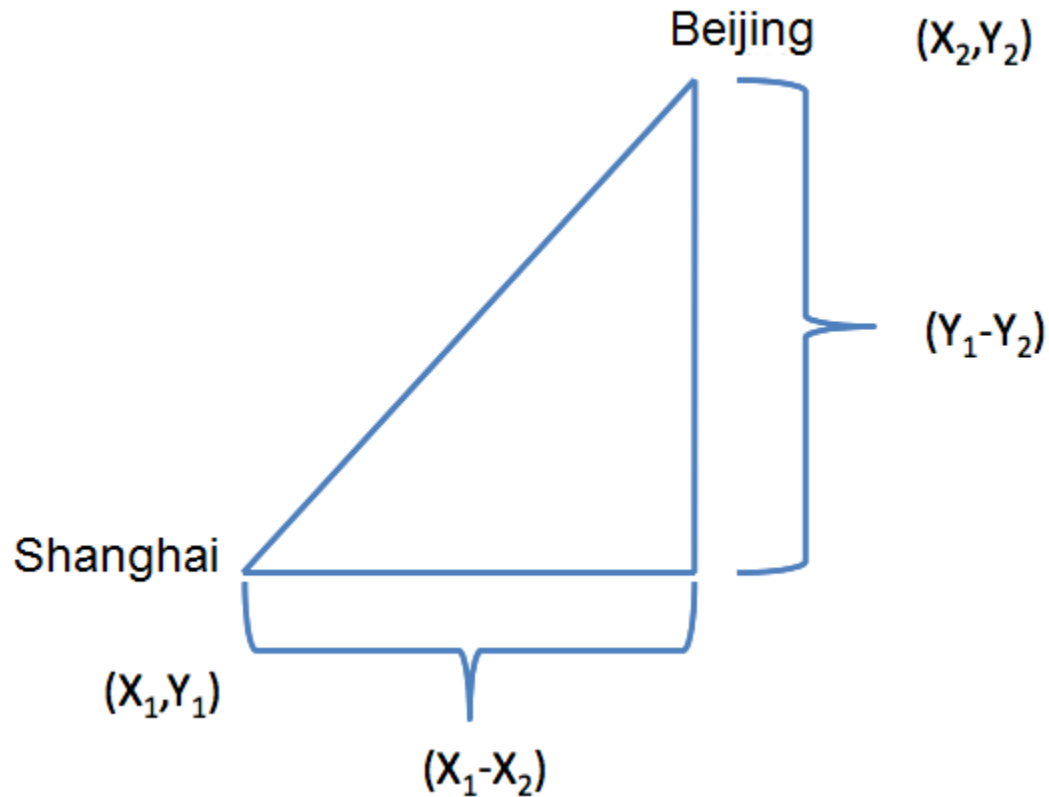
- An agglomerative approach
 - Find closest two things
 - Put them together
 - Find next closest
- Requires
 - A defined distance
 - A merging approach
- Produces
 - A tree showing how close things are to each other

How do we define close?

- Very important
 - Often depends on the problem you are considering
- Distance or similarity
 - Euclidean distance
 - Correlation similarity

Example distance-Euclidean

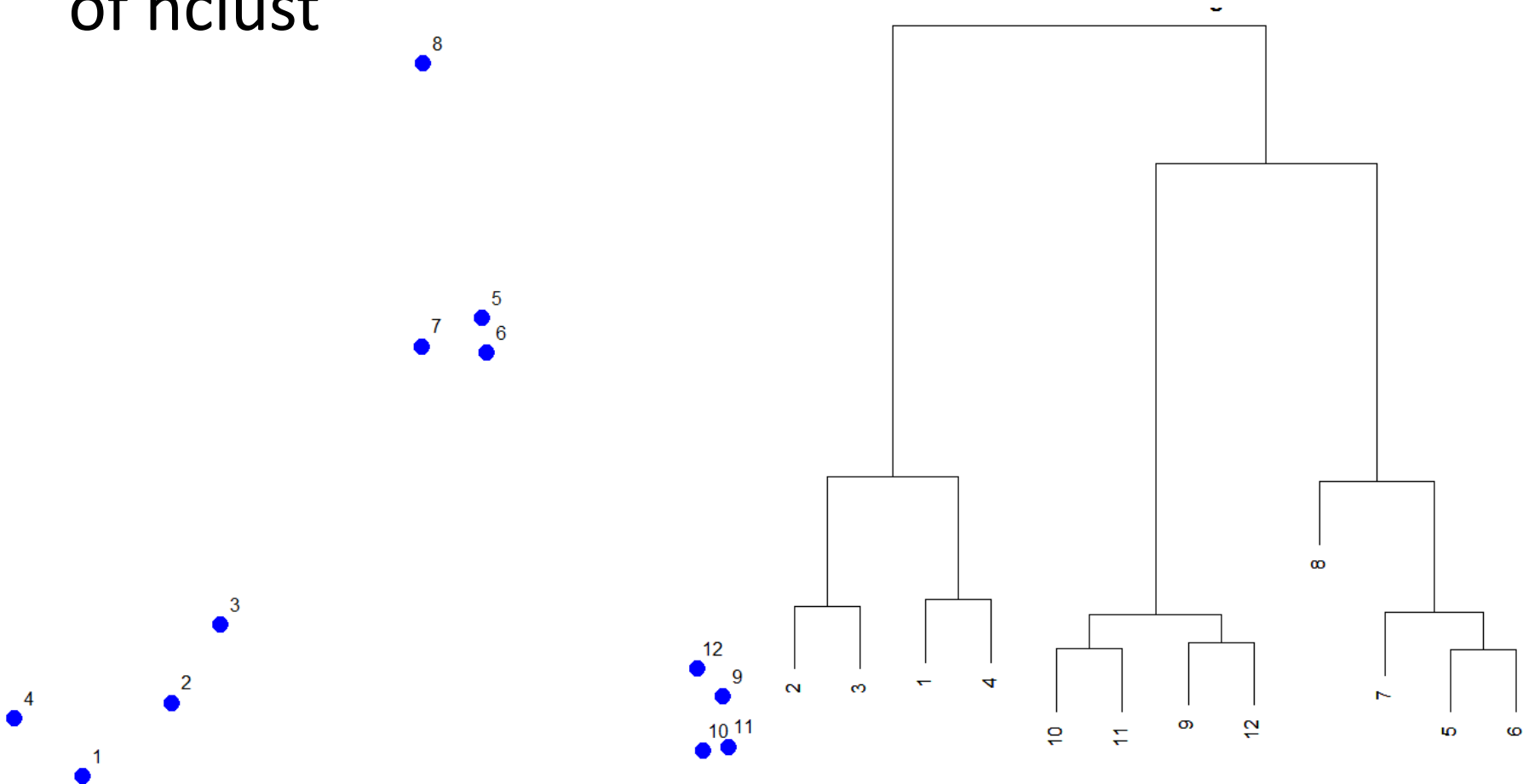
$$\sqrt{(X_1 - X_2)^2 + (Y_1 - Y_2)^2}$$



$$\sqrt{(A_1 - A_2)^2 + (B_1 - B_2)^2 + \dots + (Z_1 - Z_2)^2}$$

Hierarchical clustering

- See code and [here](#) for a faster implementation of hclust



Merging choices

Single

$$d_{SL}(G, H) = \min_{i \in G, i' \in H} d_{ii'}$$

Complete

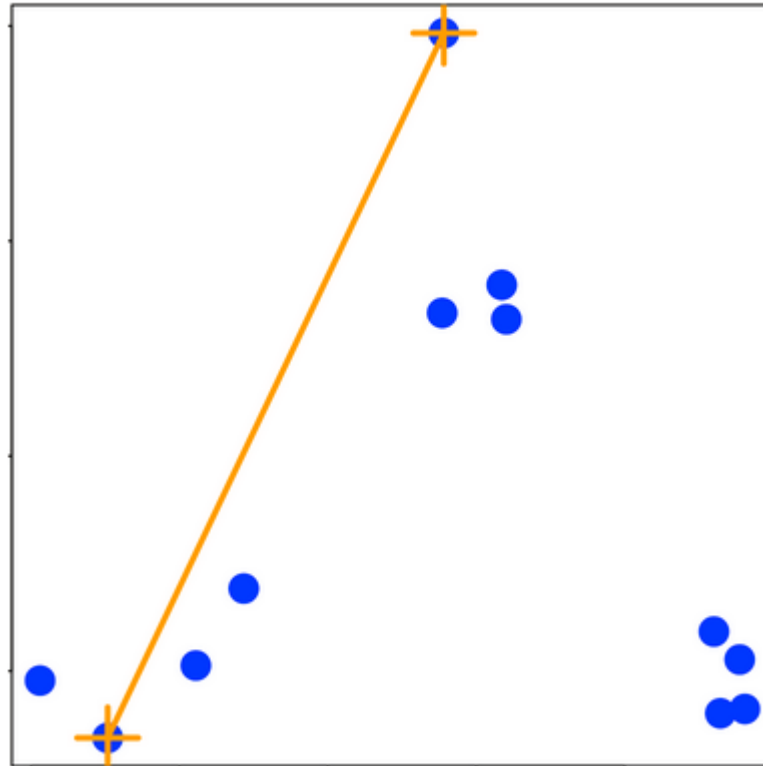
$$d_{SL}(G, H) = \max_{i \in G, i' \in H} d_{ii'}$$

Average

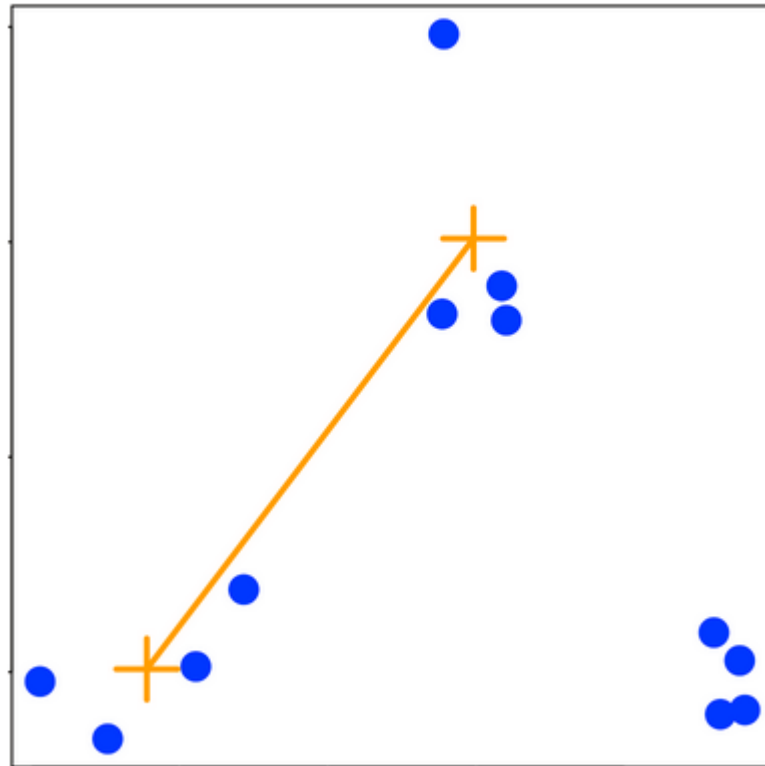
$$d_{GA}(G, H) = \frac{1}{N_G N_H} \sum_{i \in G} \sum_{i' \in H} d_{ii'}$$

Merging points - complete

- Maximum distance between points of two sets



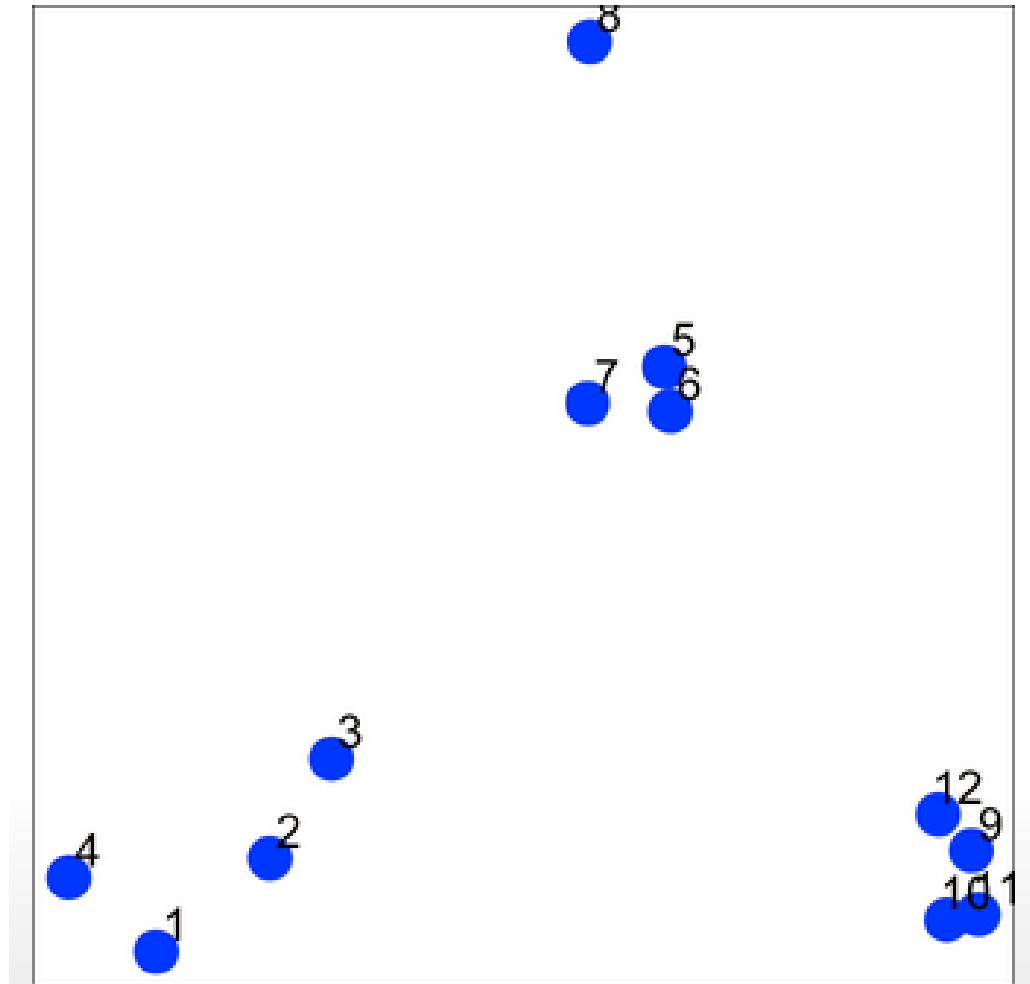
Merging points - Average



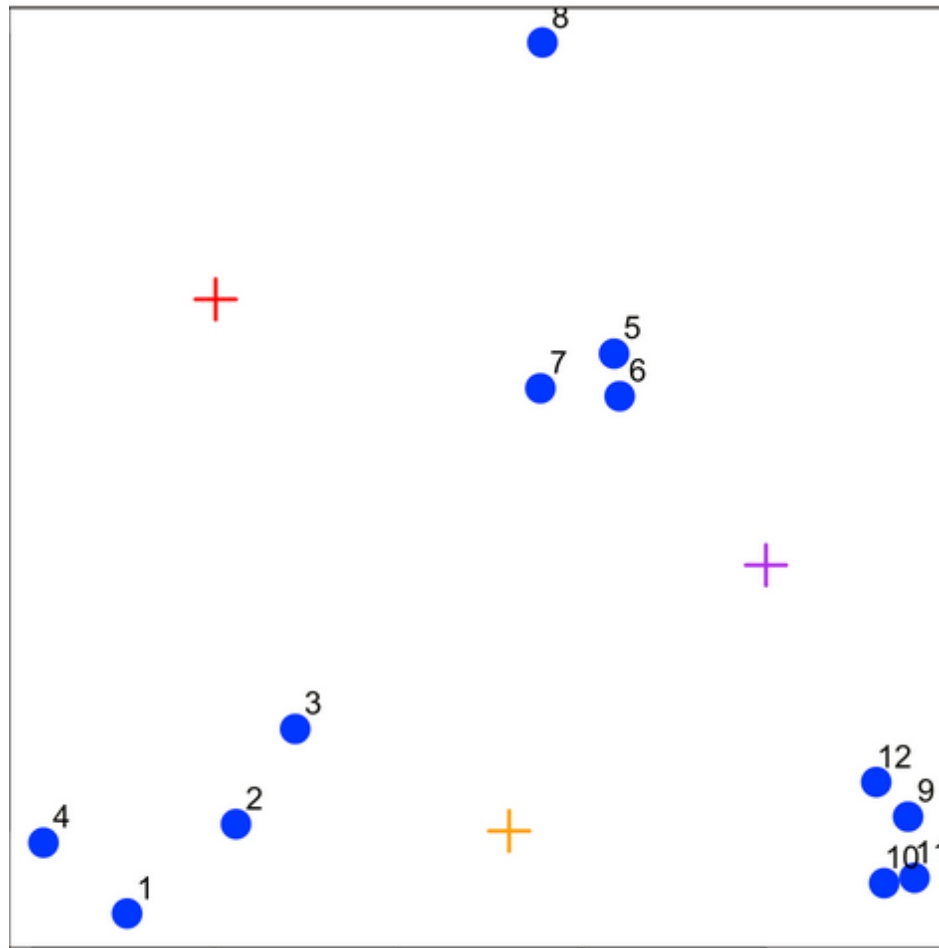
K-means clustering

- A partitioning approach
 - Fix a number of clusters
 - Get "centroids" of each cluster
 - Assign things to closest centroid
 - Recalculate centroids
- Requires
 - A defined distance metric
 - A number of clusters
 - An initial guess as to cluster centroids
- Produces
 - Final estimate of cluster centroids
 - An assignment of each point to clusters

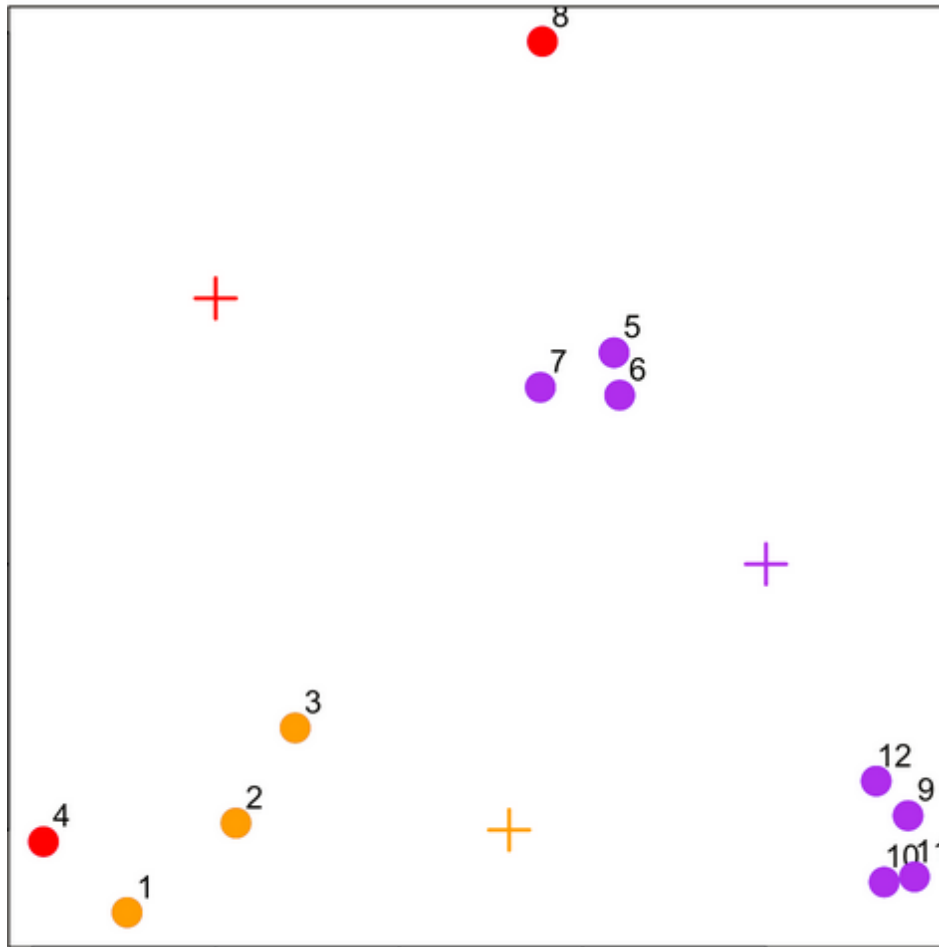
K-means-example



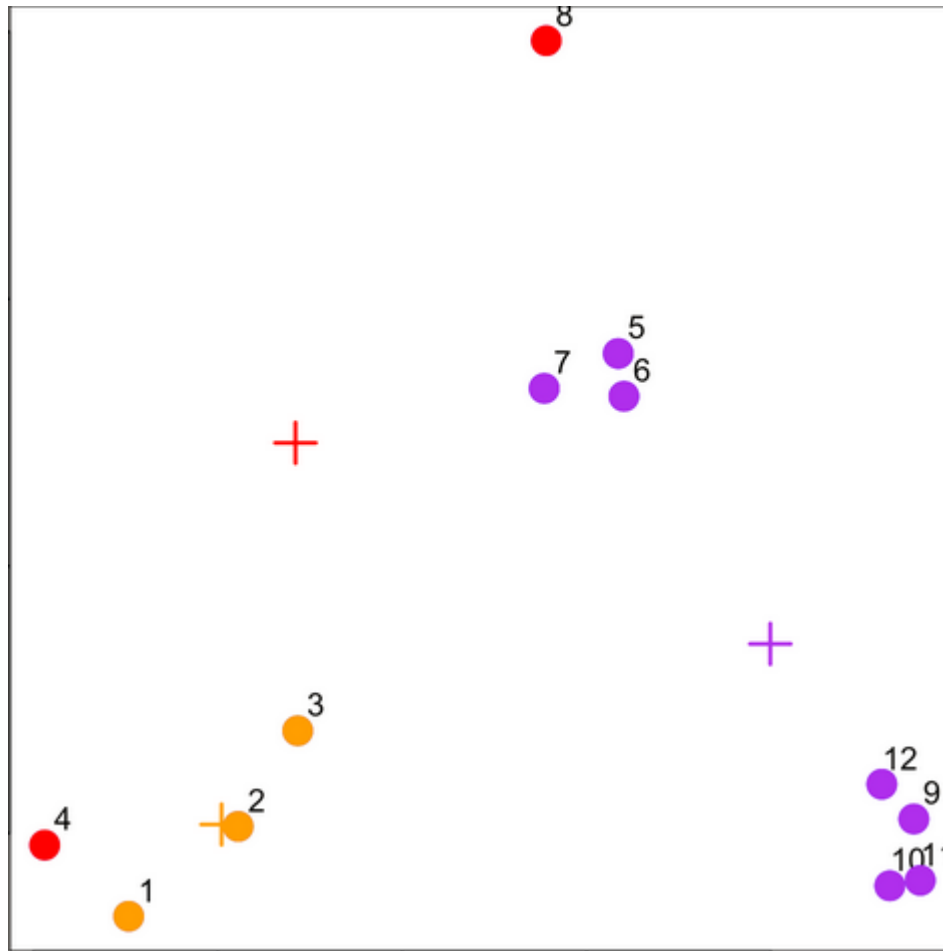
K-means clustering - starting centroids



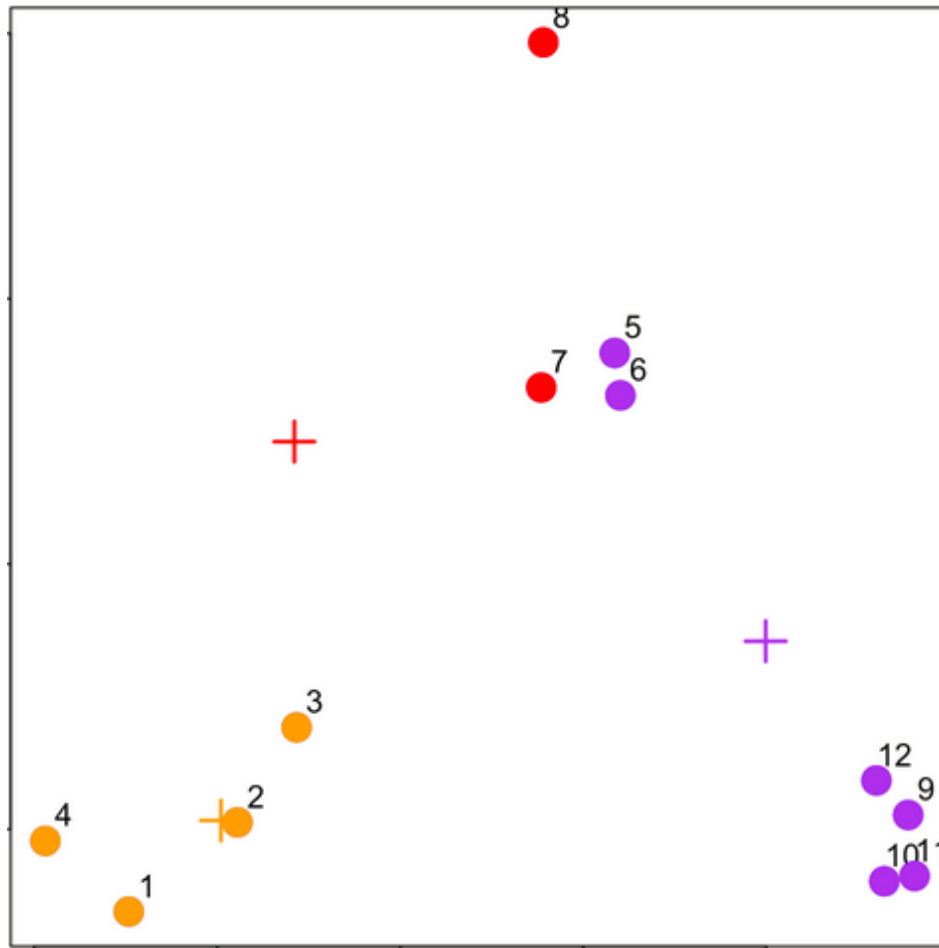
K-means clustering - assign to closest centroid



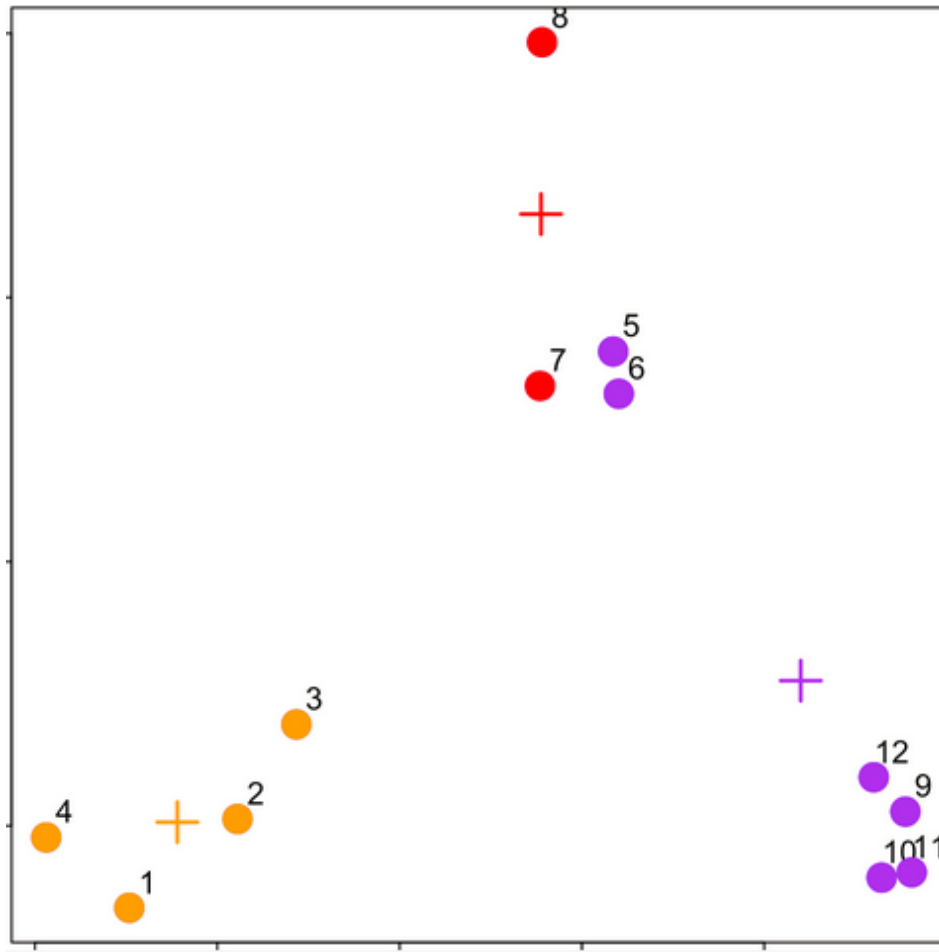
K-means clustering - recalculate centroids



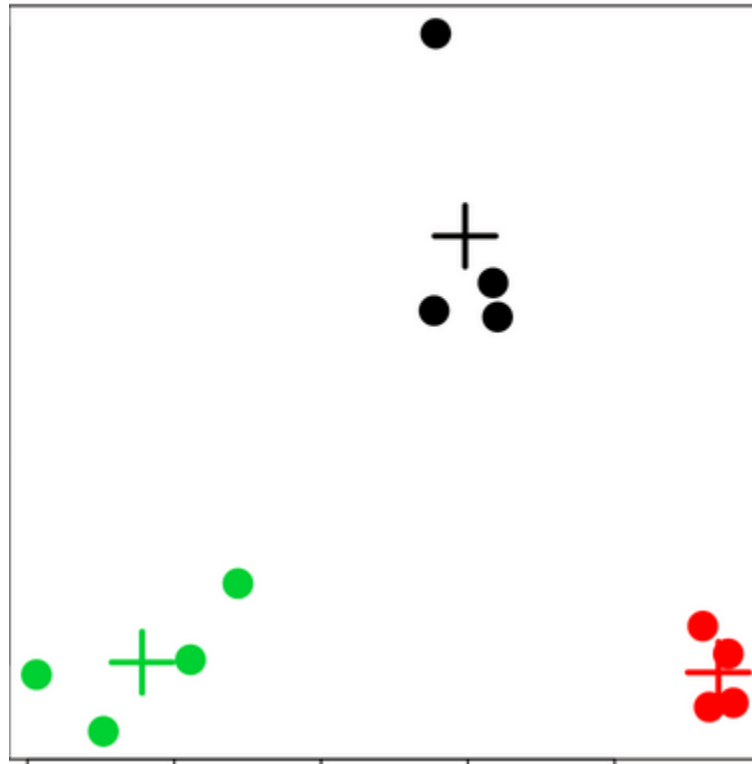
K-means clustering - reassign values



K-means clustering - update centroids



K-means



Important parameters: x , *centers*, *iter.max*, *nstart*

K-means algorithm

Given initial clusters $m_1^{(1)}, \dots, m_k^{(1)}$ we iterate between:

Assign each point to a cluster

$$S_i^{(t)} = \left\{ x_p : \|x_p - m_i^{(t)}\|^2 \leq \|x_p - m_j^{(t)}\|^2, \forall j \right\}$$

Update means

$$m_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{x_j \in S_i^{(t)}} x_j$$

Stop when the m_i have converged to local modes.

Model based clustering

Assume the data are drawn from a distribution:

$$f(x|\pi, \mu, \Sigma) = \sum_{g=1}^G \pi_g \phi(x|\mu_g, \Sigma_g)$$

where π_g is the probability a point belongs to group g and $\phi(x|\mu_g, \Sigma_g)$ is the multivariate Gaussian density.

- You can do this with other densities but you usually have to "roll your own"
- Gaussian densities are surprisingly flexible in many cases

Estimating parameters

[EM-algorithm](#): first proposed by Dempster, A.P.; Laird, N.M.; Rubin, D.B. in their seminal [work](#)

$$\pi_{ik}^{(s)} = \frac{\pi_k^{s-1} \phi(x_i; \mu_k^{s-1}, \Sigma_k^{s-1})}{\sum_{k'=1}^K \pi_{k'}^{s-1} \phi(x_i; \mu_{k'}^{s-1}, \Sigma_{k'}^{(s-1)})}$$

$$\pi_k^{(s)} = \frac{1}{n} \sum_{i=1}^n \pi_{ik}^{(s)}$$

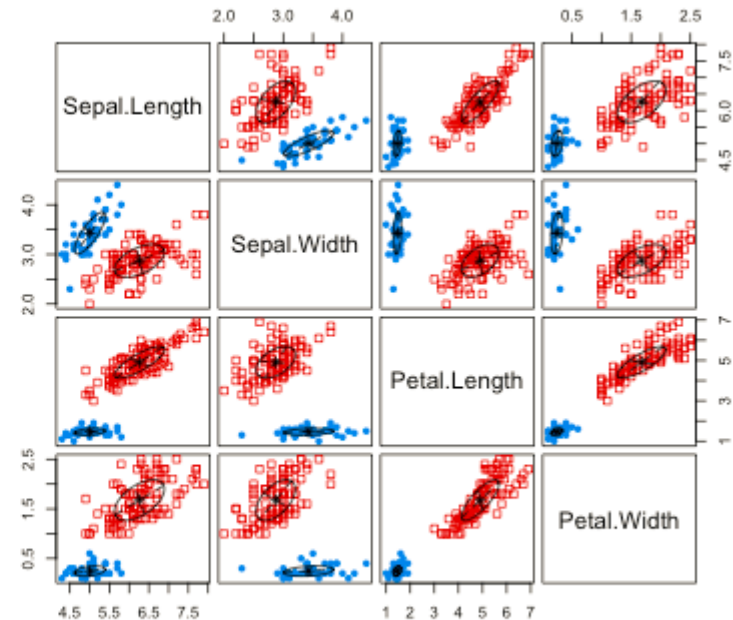
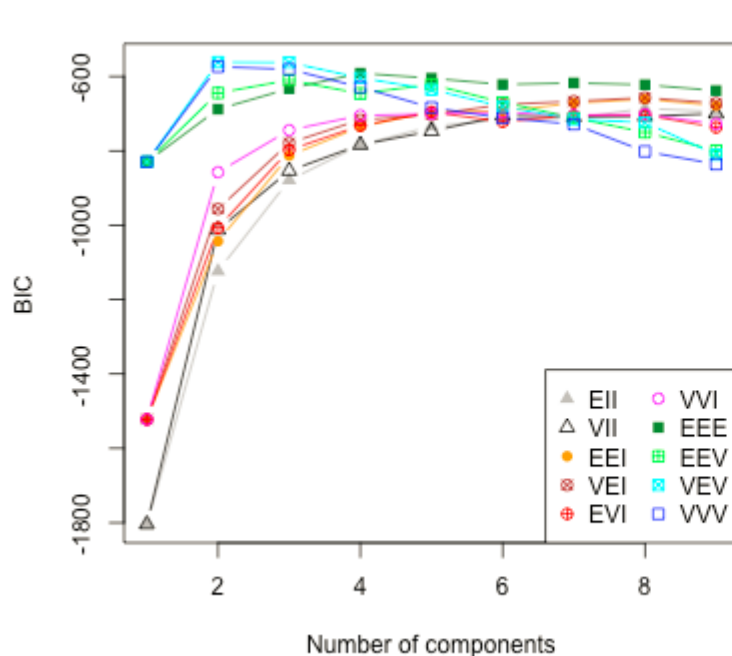
$$\mu_k^{(s)} = \frac{\sum_{i=1}^n \pi_{ik}^{(s)} x_i}{\sum_{i=1}^n \pi_{ik}^{(s)}}$$

$$\Sigma_k^{(s)} = \frac{\sum_{i=1}^n \pi_{ik}^{(s)} (x_i - \mu_k^{(s)}) (x_i - \mu_k^{(s)})^T}{\sum_{i=1}^n \pi_{ik}^{(s)}}$$

Selecting the model with Bayesian Information Criterion

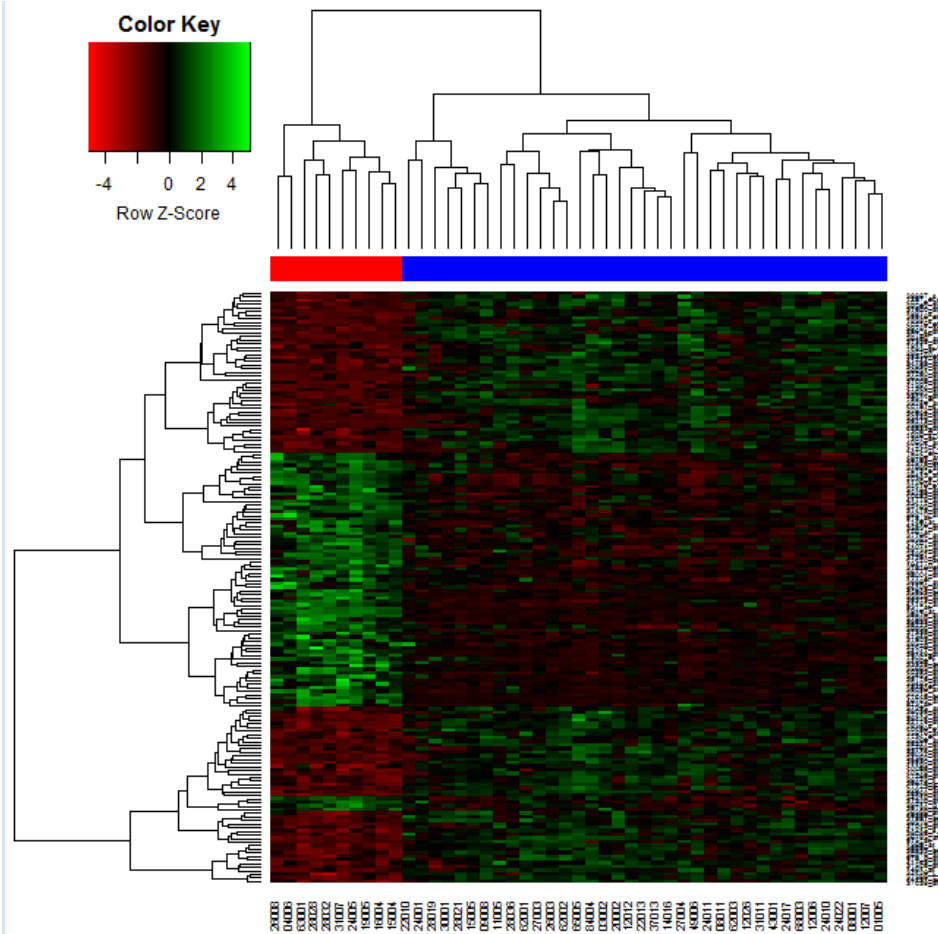
- The Bayesian Information Criterion (BIC)

$$BIC = -2\log\text{lik} + (\log N)d$$



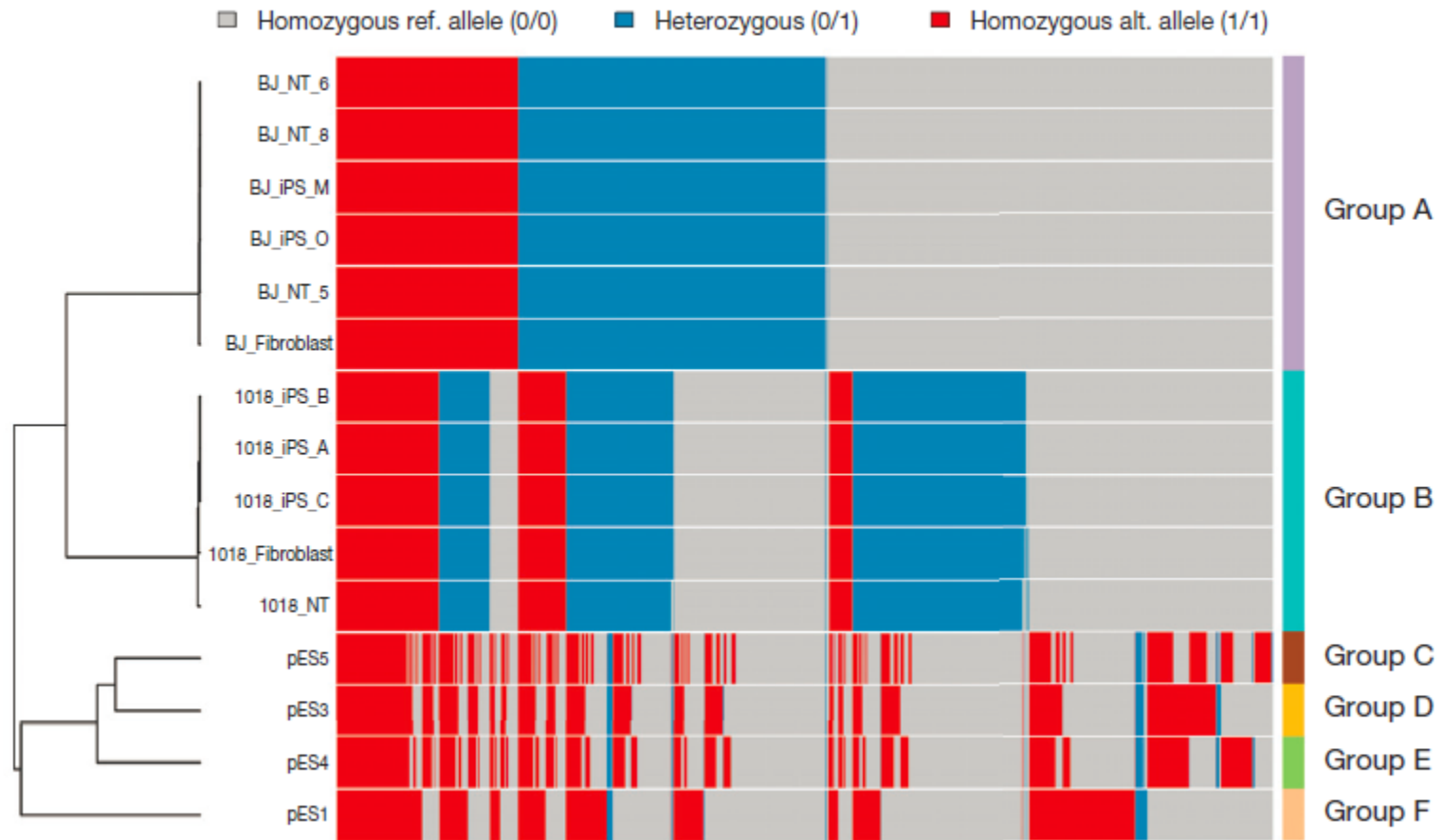
Note the BIC in mclust are -BIC defined here

Heatmap in real applications



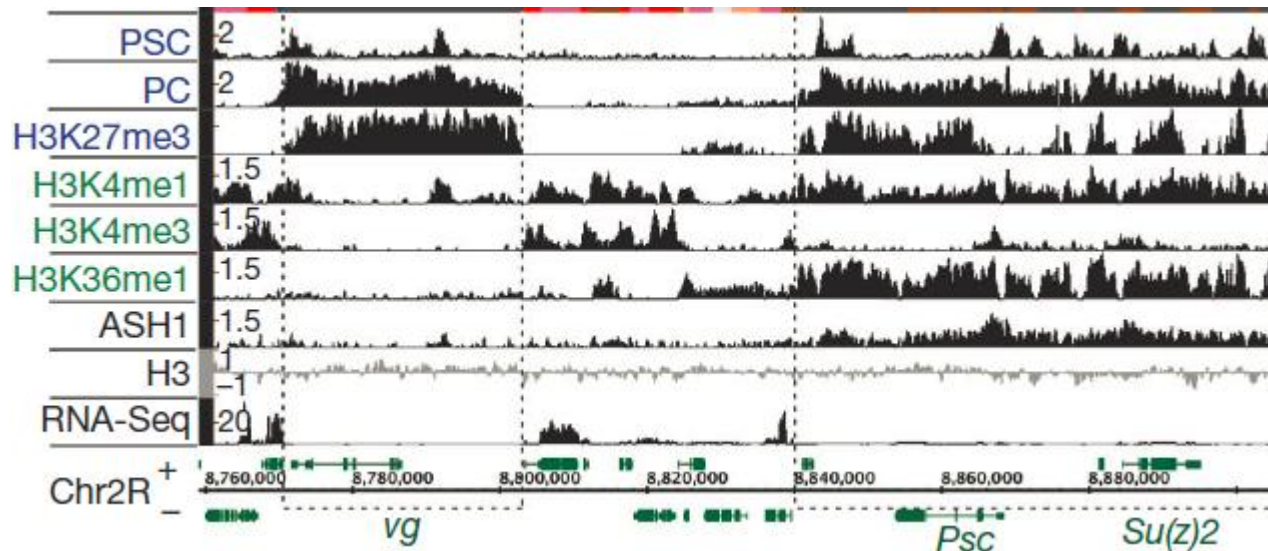
[ALL data](#) See code

Heatmap in real applications



Picture Taken from [De Los Angeles et al. Nature2015](#)

Clustering with spatial information



How to perform clustering for this data?

Picture taken from [Kharchenko et al. Nature 2011](#)

Clustering with spatial information

- Hidden Markov model
 - See blackboard
- Rpackage: RHmm