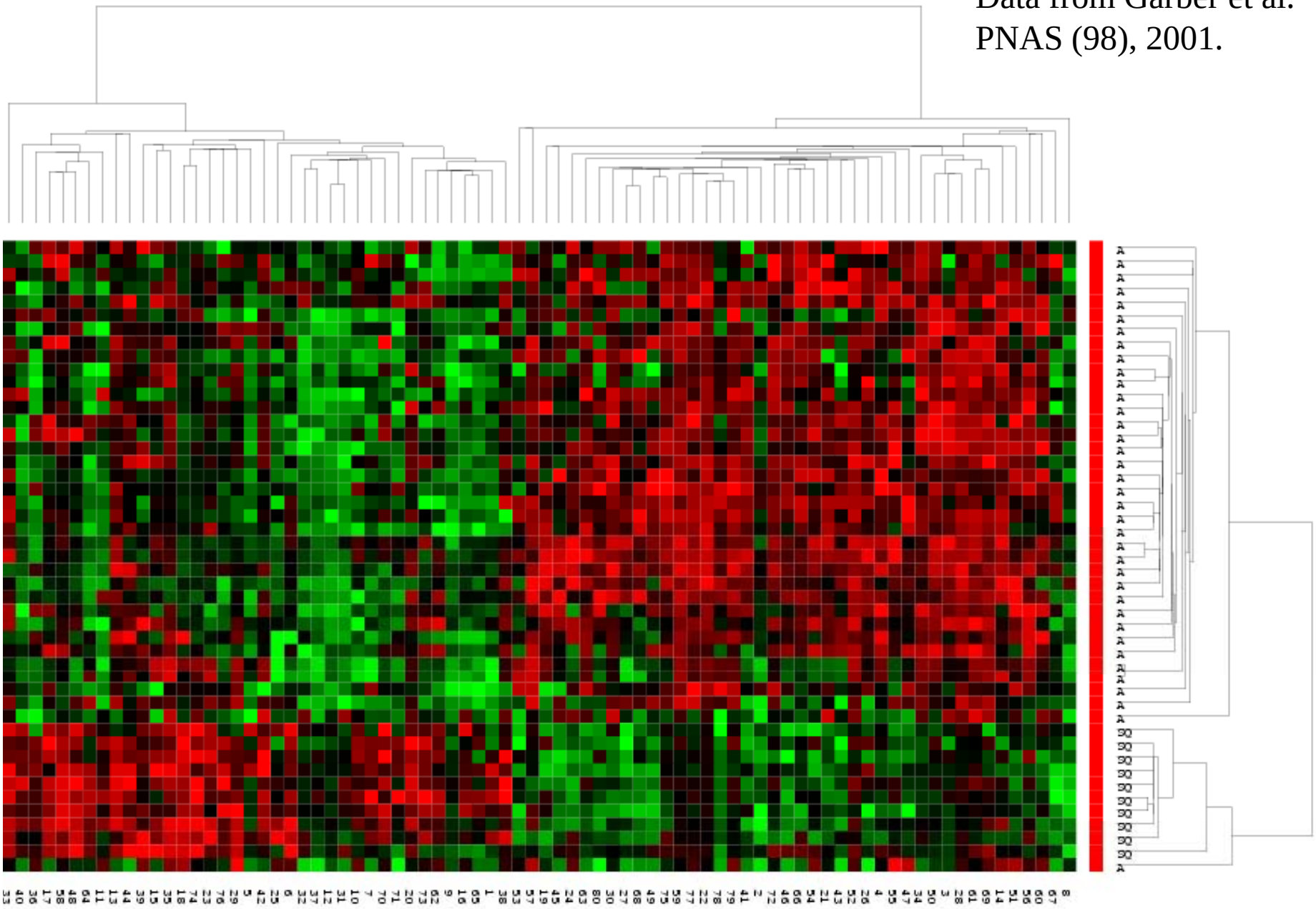
Clustering Gene Expression Data:

Acknowledgement: Elizabeth Garrett-Mayer;
Shirley Liu; Robert Tibshirani; Guenther Walther;
Trevor Hastie

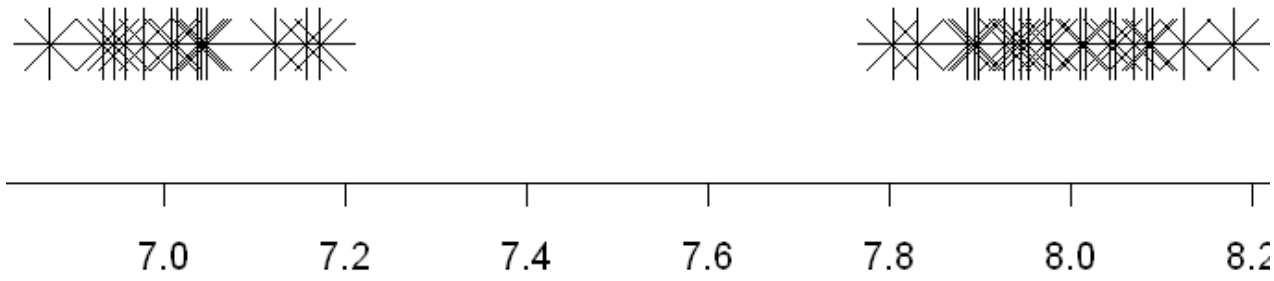
Data from Garber et al.
PNAS (98), 2001.



Clustering

- "Clustering" is an exploratory tool for looking at associations within gene expression data
- Hierarchical clustering dendrograms allow us to visualize gene expression data.
- These methods allow us to hypothesize about relationships between genes and classes.
- We should use these methods for visualization, hypothesis generation, selection of genes for further consideration
- You can cluster without having a probability model. In fact most methods are model-less. There is no measure of "strength of evidence" or "strength of clustering structure" provided.
- It is difficult to judge if one clustering result is better than another, or how similar two clustering results are
- Hierarchical clustering specifically: we are provided with a picture from which we can make many/any conclusions.

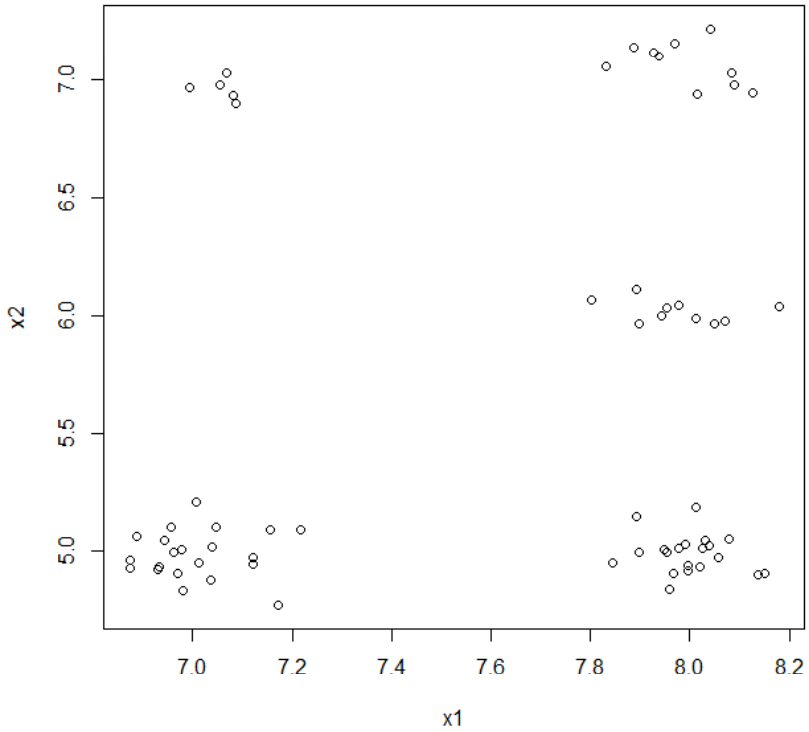
- Cluster analysis arranges objects into groups based on their values in multiple characteristics.
  - In an extreme example, suppose we have observation on only one gene , gene X, in a collection of samples.



It seems that the samples form two clusters: one with relatively low expression level (aroud 7) and another with relatively high expression level (around 8) for gene X.

Mathematically how do we formulate what our eyes have done intuitively?

What if we have two genes?

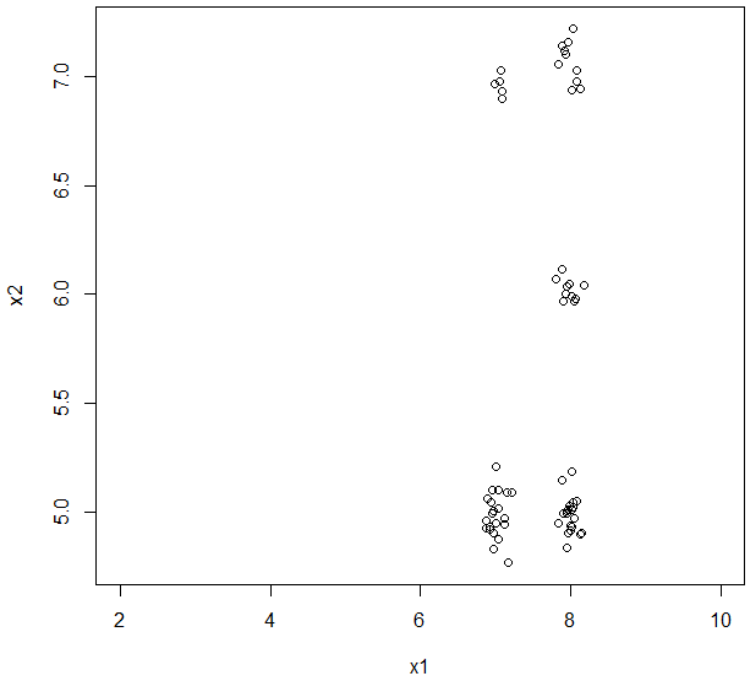


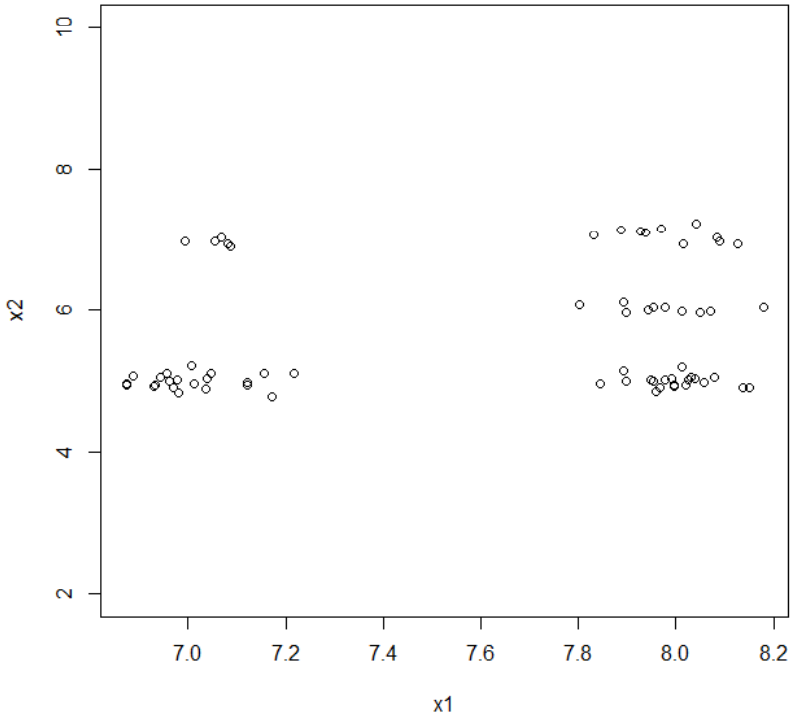
It seems now we see 5 clusters of samples:

Groups of samples	GeneX1	GeneX2
1	Low	Low
2	Low	High
3	High	Low
4	High	Medium
5	High	High

Again, how did we group the samples? What does it mean mathematically that some points are close to each other?

pictures can be misleading





How we stretched the picture → how each dimension is weighted in determining distance.

Distance and Similarity

- Every clustering method is based **solely** on the measure of distance or similarity.
- E.g. correlation: measures linear association between two samples or genes.
  - What if data are not properly transformed?
  - What if there are outliers?
  - What if there are saturation effects?
- Even with large number of samples, bad measure of distance or similarity will not be helped.

Distance versus similarity

Distance

- Non-negativity
  - $d(x,y)>=0$
- Symmetry
  - $d(x,y)=d(y,x)$
- Identification mark
  - $d(x,x)=0$
- Definiteness
  - $d(x,y)=0$ iff $x=y$
- Triangle inequality
  - $D(x,y)+d(y,z)>=d(x,z)$

Similarity

- Non-negativity
  - $s(x,y)>=0$
- Symmetry
  - $s(x,y)=s(y,x)$
- Higher when x,y are more similar

- Cluster analysis arranges samples and genes into groups based on their expression levels.
- This arrangement is determined purely by the measured distance or similarity between samples and genes.
- Popular Distances:
  - Euclidian distance $\sum_i (x_i - y_i)^2$ $d = \sqrt{\sum_{i=1}^{k} (X_i - Y_i)^2}$
  - Manhattan distance $\sum_i abs(x_i - y_i)$
  - 1- correlation coefficient

$$ss_{xx} = \sum (x_i - \bar{x})^2$$

$$ss_{yy} = \sum (y_i - \bar{y})^2$$

$$ss_{xy} = \sum (x_i - \bar{x})(y_i - \bar{y})$$

$$r = \frac{ss_{xy}}{\sqrt{ss_{xx} ss_{yy}}}$$

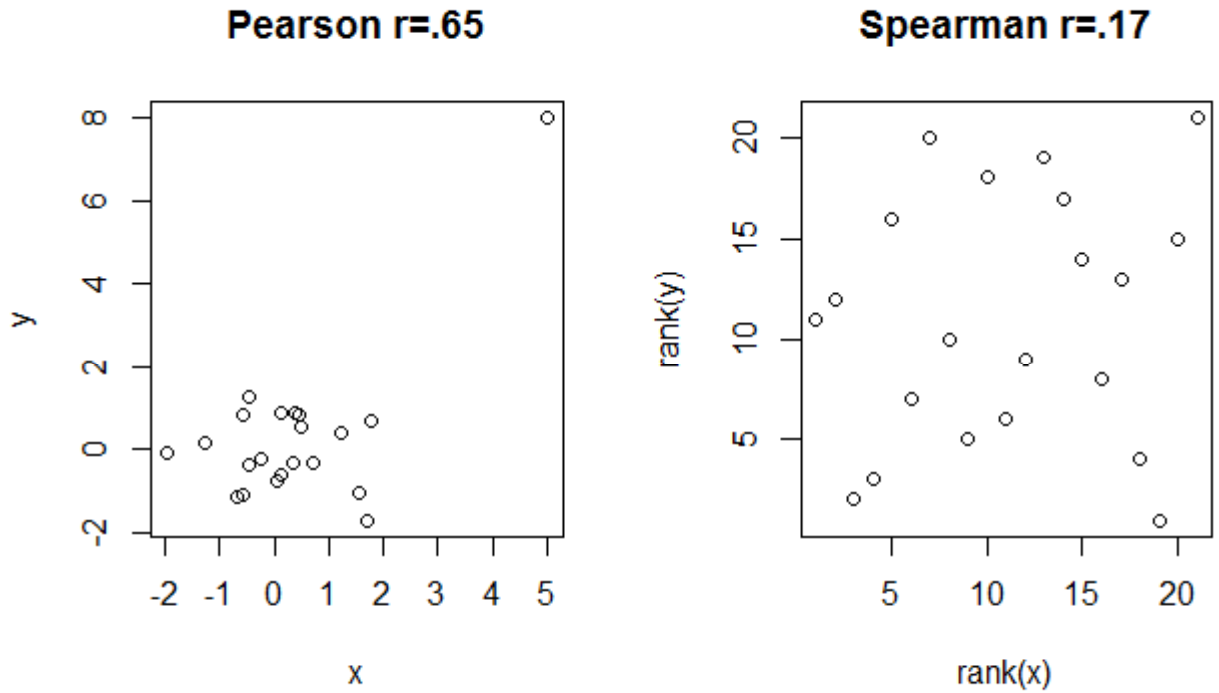
Pearson Correlation vs. Euclidean distance

- Exercise:
  - When the data are standardized (with mean 0 and sd 1), there is a simple linear relationship between the Pearson correlation coefficient *r* and the squared Euclidean distance

Correlation based distance

- Pearson correlation distance

  1-r(x,y)
- Spearman correlation distance
  - 1-r (rank(x), rank(y))
  - More robust, less sensitive



Sometimes we consider abs(r)

- Strong positive correlation often suggest “closeness” or “similarity”
- In some cases, strong negative correlation is also interesting, especially between genes: both inhibition and induction are common

Mahalanobis distance

- If a pair of vectors, **x** and **y**, are generated from multivariate distribution with mean and variance-covariance matrix $\sum$

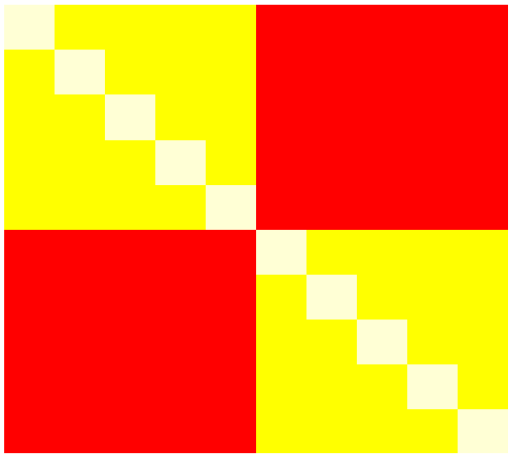
$$(x-y)'\Sigma^{-1}(x-y)$$

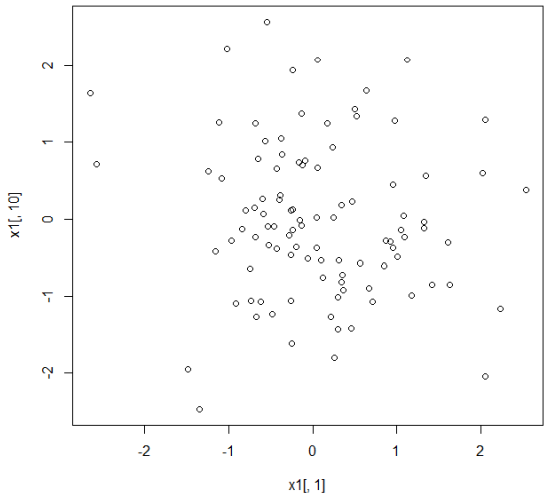
- When each element has sd 1, this reduces to squared Euclidean distance
- When the elements (dimension) are independent, $\sum$ is a diagonal matrix, and we are simply standardizing. This is the same as first standardizing each dimension, and then compute Euclidean distance
- When the elements are dependent, this accounts for the similarity between the elements.

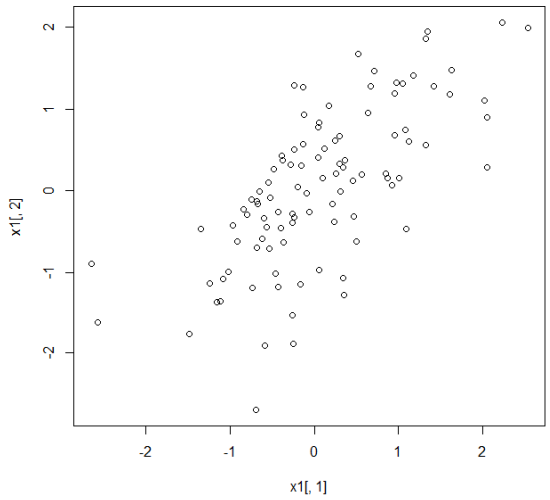
Mahalanobis distance between genes

- For a pair of genes measured over N experiments, **x** and **y** are vector of length N. $\sum$ is N x N, representing the variance-covariance of the samples.
- Typically after normalization, the samples have the same distribution. Thus the mean and sd of the sample is the same. So if the samples are independent, the mahalanobis distance works the same as Euclidean distance
- What if there is correlation between experiments?
  - nested design
  - technical replicates and biological replicates

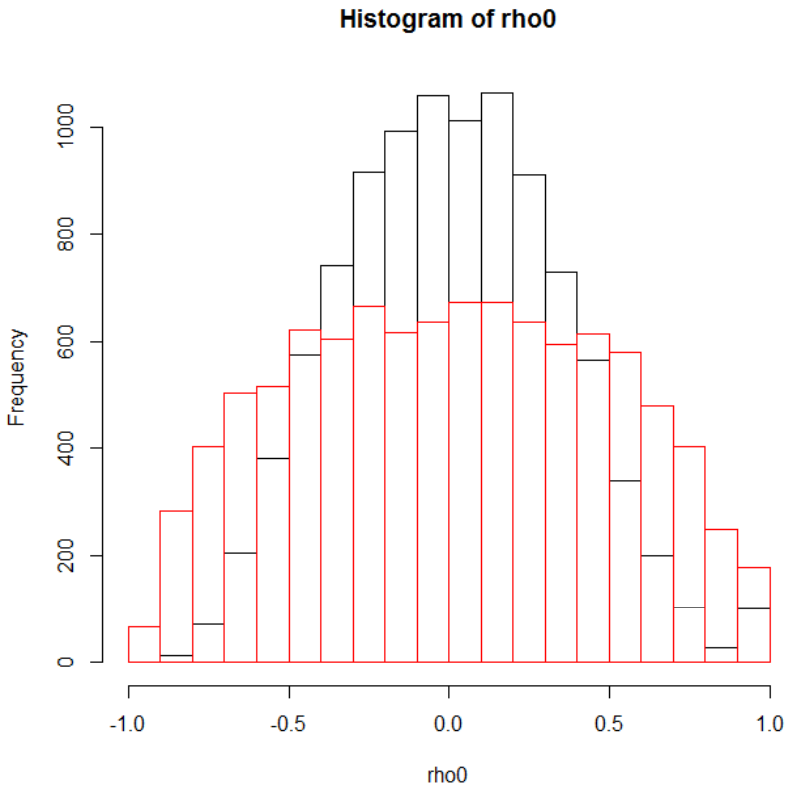
- A: Consider two biological conditions with 5 independent biological samples
- B: Consider two biological experiments with 5 technical replicates each
  - The technical replicates are correlated
- Consider 100 genes that are unrelated







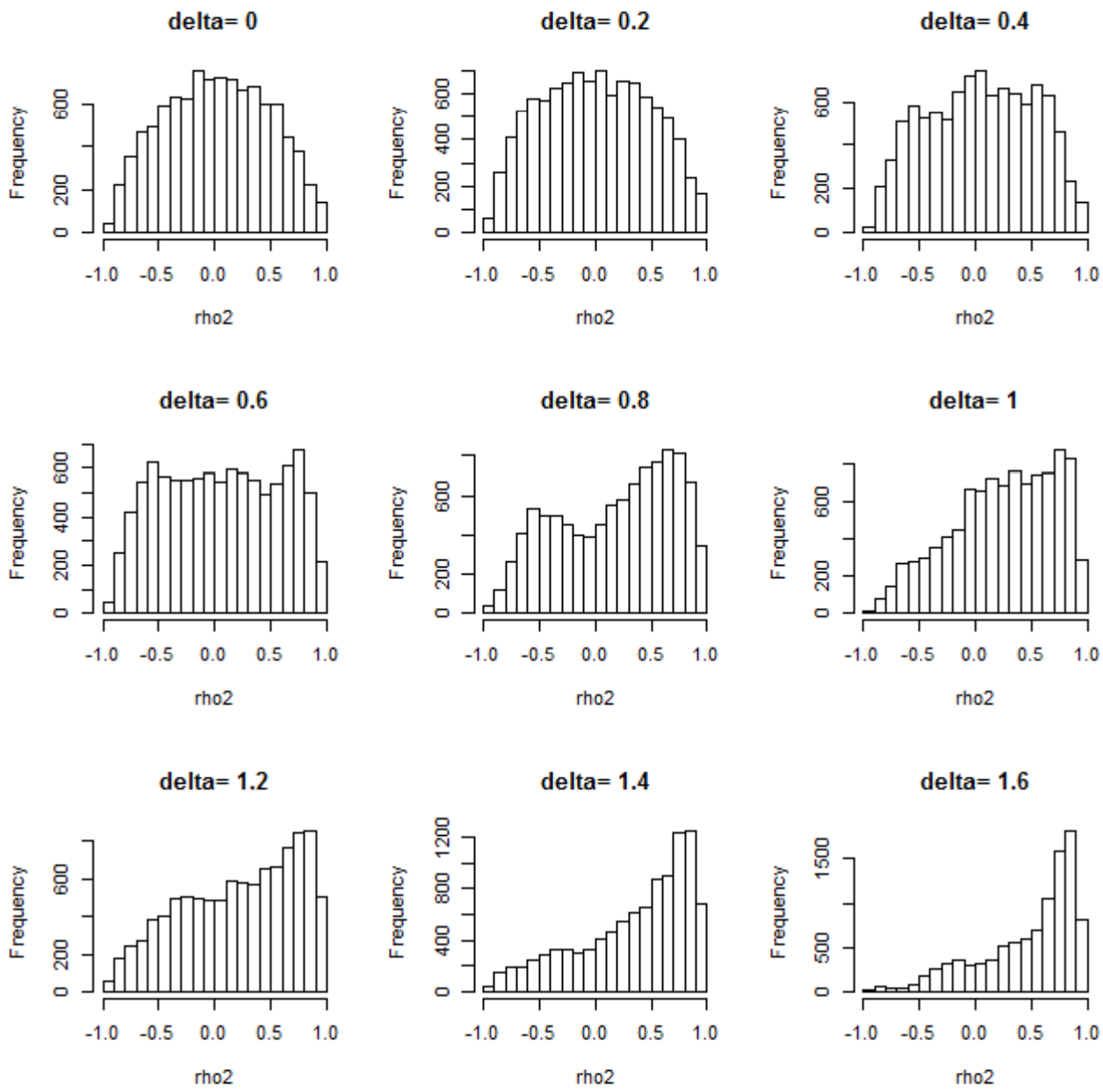
```
Sigma[1:5,1:5]=.7
Sigma[6:10,6:10]=.7
diag(Sigma)=1

image2=function(x) ## to visualize as the matrix shows
 image(t(x)[ncol(x):1,],axes=F)
image2(Sigma)

set.seed(123)
x0=mvrnorm(100,mu=rep(0,10),diag(10))
x1=mvrnorm(100,mu=rep(0,10),Sigma)
rho0=cor(t(x0))
rho1=cor(t(x1))
 hist(rho0,xlim=c(-1,1))
 hist(rho1,add=T,border=2)
```



- It is much more likely to see strong correlation between genes when the experiments are not independent

- This is even worse when there is some differential expression
- Consider two biological experiments with 5 technical replicates each
- The technical replicates are correlated
- If there is differential expression between the two experiments:
  Average expression (0,0,0,0,0,δ,δ,δ,δ,δ)

```
set.seed(123)
par(mfrow=c(3,3))
sapply(0:8*.2,function(delta){ # for different levels of DE
  x2=mvrnorm(100,mu=c(rep(0,5),rep(delta,5)),Sigma)
  rho2=cor(t(x2))
  hist(rho2,main=paste("delta=",delta))
})
```


delta= 0
delta= 0.2
delta= 0.4
delta= 0.6
delta= 0.8
delta= 1
delta= 1.2
delta= 1.4
delta= 1.6
Frequency
rho2
-1.0
-0.5
0.0
0.5
1.0
0
200
400
600
800
1200
500
1500

Mahalanobis distance between samples

- For a pair of samples including G genes, **x** and **y** are vector of length G. ∑ is G x G, representing the variance-covariance of the genes.
- The genes may have different variances. If the genes are independent

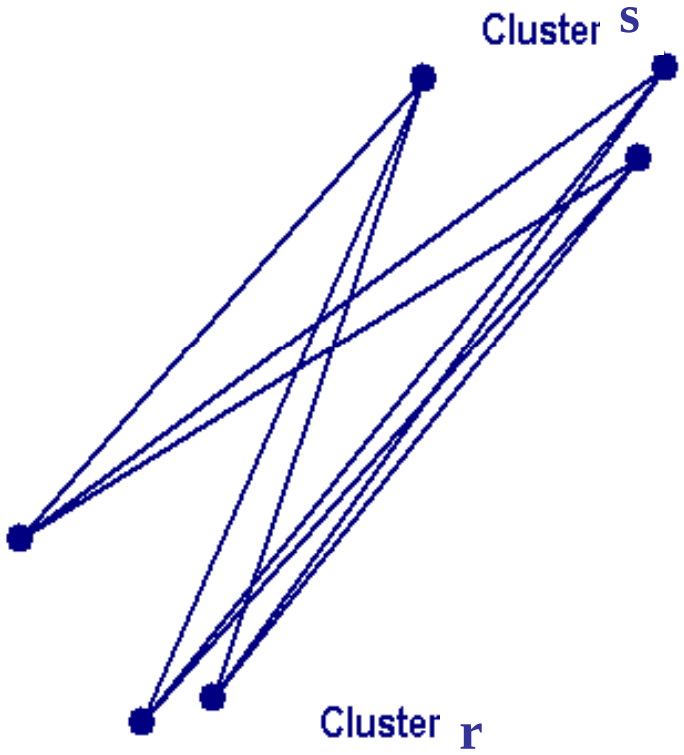
$$(x-y)'\Sigma^{-1}(x-y)$$

  is the same as standardizing the genes first, then compute Euclidean distance
- But we know the genes are not independent
- And many genes may not be of interest at all – we'll get back to this again

Suppose we settled the issue of distance between subjects. How do we cluster?

- Maximize inter-cluster distance
- Minimize intra-cluster distance
- Linkage:
  - Average linkage: average distance between any pair of elements in different clusters
  - Complete linkage: longest distance between two elements in different clusters
  - Single linkage: shortest distance between two elements in different clusters

Average linkage

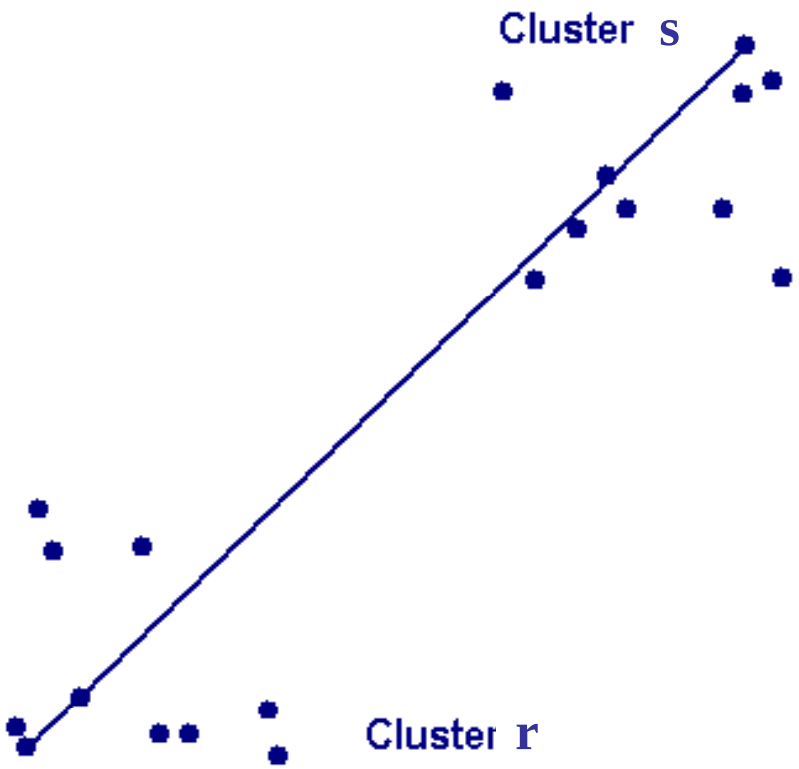


$$D(r,B) = T_{ab} / (N_a * N_b)$$

Where ***Trs*** is the sum of all pairwise distances between cluster ***A*** and cluster ***B.***

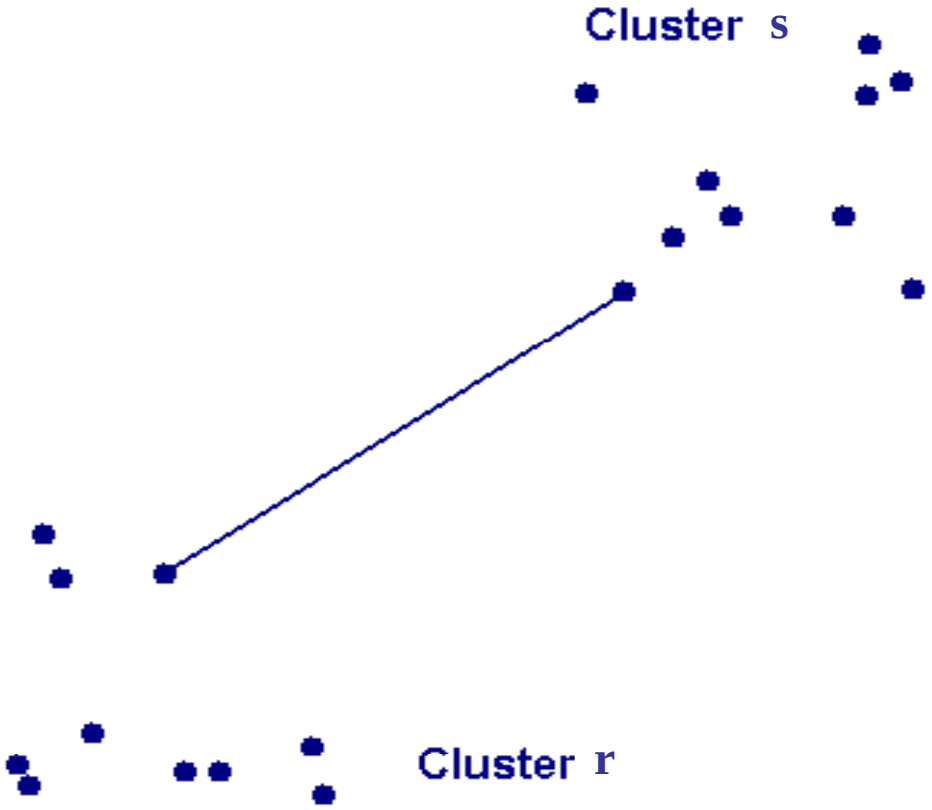
Na and ***Nb*** are the sizes of the clusters ***A*** and ***B*** respectively

Complete Linkage



- ***D(r,s)*** = Max { *d(i,j)* : Where object *i* is in cluster ***r*** and object *j* is cluster ***s*** }

Single linkage



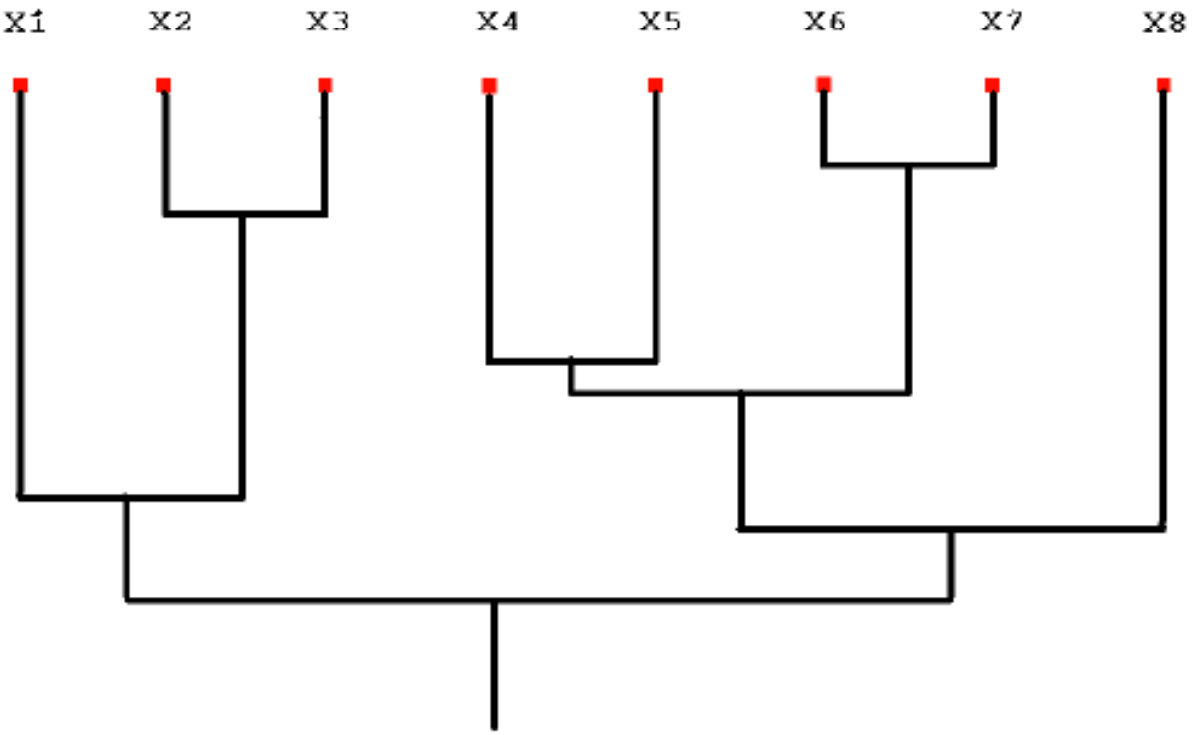
D(r,s) = Min { *d(i,j)* : Where object *i* is in cluster ***r*** and object *j* is cluster ***s*** }

- Single linkage can lead to “string-like” clusters in high dimensions
- Complete linkage tends to find more compact “spherical” clusters
- Average linkage produces something intermediate

Hierarchical Clustering

- Divisive Hierarchical Clustering
  - Top down
  - Start from one cluster and split into smaller clusters
  - Maximize inter cluster distance in each split
- Agglomerative Hierarchical Clustering
  - Bottom up
  - Start from singletons and merge into larger clusters
  - Join closest clusters at each stage
  - More common than Divisive approach

Example: hierarchical clustering



Partitional clustering

- K-means

  Minimize $J = \sum_{j=1}^{K} \sum_{n \in S_j} |x_n - \mu_j|^2,$

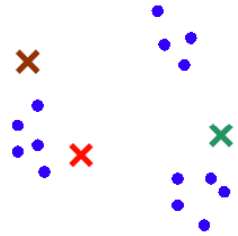
  - μ_j is the centroid of each cluster.
  - other distances can be used as well.

  – Choose cluster size k first
  – Initial centroids can be random
  – Update clustering by assigning elements to closest centroid
  – Re-define centroids by current clustering
  – Iterate till clusters stablize

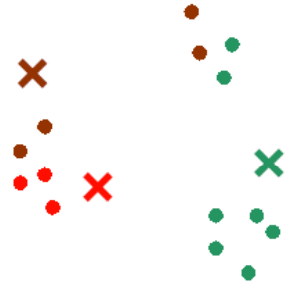
More detailed Kmeans algorithm

1. Choose K centroids at random
2. Make initial partition of objects into k clusters by assigning objects to closest centroid
3. Calculate the centroid (mean) of each of the k clusters.
4. a. For object i, calculate its distance to each of the centroids.
   b. Allocate object i to cluster with closest centroid.
   c. If object was reallocated, recalculate centroids based on new clusters.
4. Repeat 3 for object i = 1,….N.
5. Repeat 3 and 4 until no reallocations occur.
6. Assess cluster structure for fit and stability

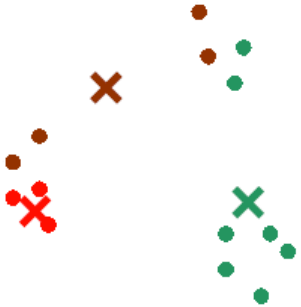
Kmeans



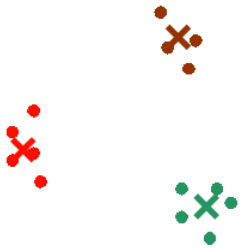
Random centroids



Each point is assigned to the closest centroid

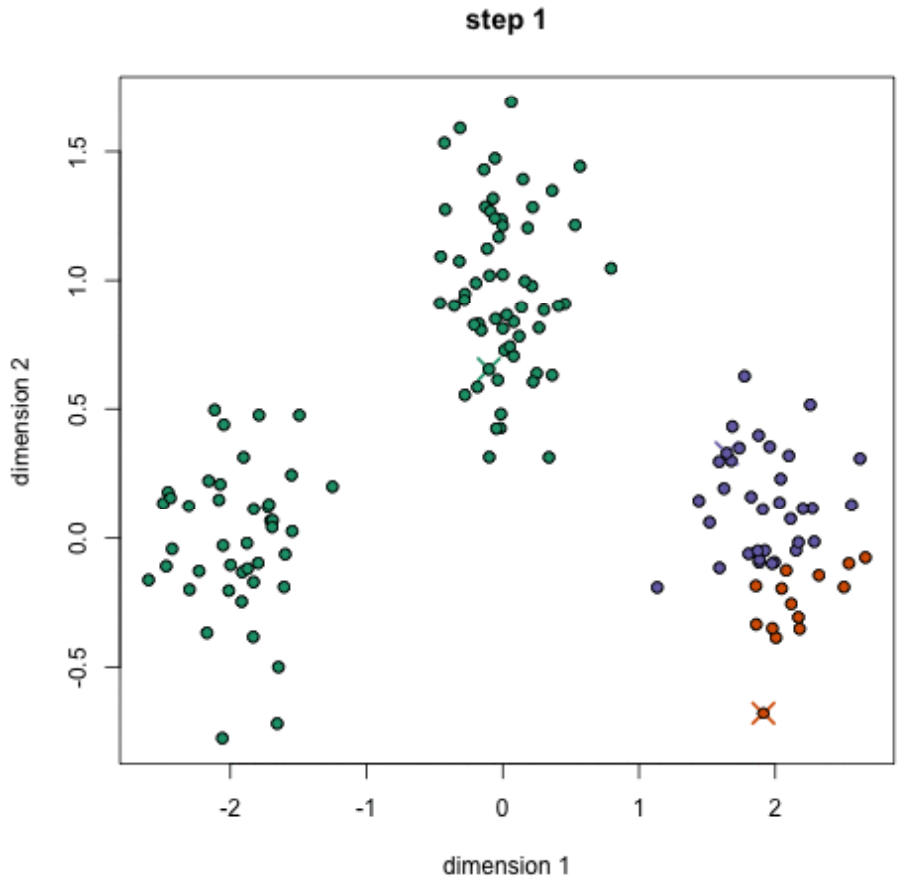


Update centroids location based on current clustering



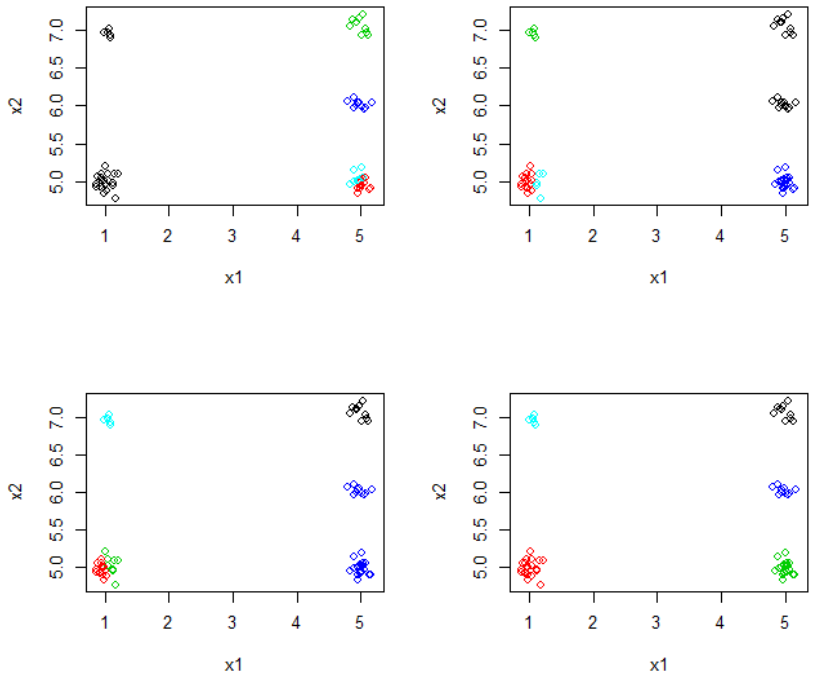
Update clustering again with new centroids

How Kmeans work in a movie



http://simplystatistics.org/2014/02/18/k-means-clustering-in-a-gif/kmeans/

Kmeans can stuck in local optimum

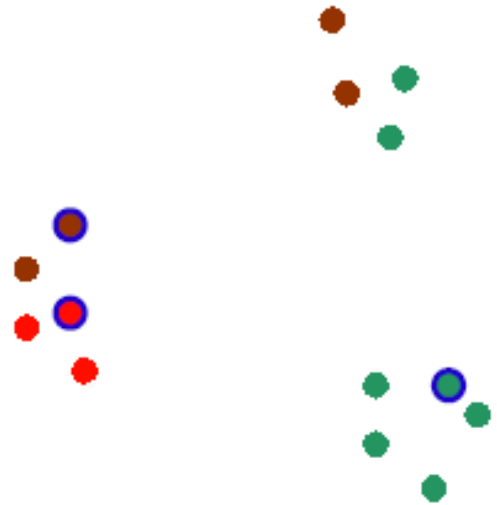



```
set.seed(123)
x1=c(rnorm(13,1,.1),rnorm(20,5,.1),rnorm(13,1,.1),rnorm(20,5,.1))
x2=c(rnorm(13,5,.1),rnorm(10,6,.1),rnorm(15,7,.1),rnorm(66-13-10-
      15,5,.1))
dat1=cbind(x1,x2)
#############################################################
par(mfrow=c(2,2));set.seed(111)
cluster1=kmeans(dat1,5)
cluster1$tot.withinss
plot(dat1,col=cluster1$cluster)
cluster1=kmeans(dat1,5);cluster1$tot.withinss
plot(dat1,col=cluster1$cluster)
cluster1=kmeans(dat1,5);cluster1$tot.withinss
plot(dat1,col=cluster1$cluster)
cluster1=kmeans(dat1,5);cluster1$tot.withinss
plot(dat1,col=cluster1$cluster)
```

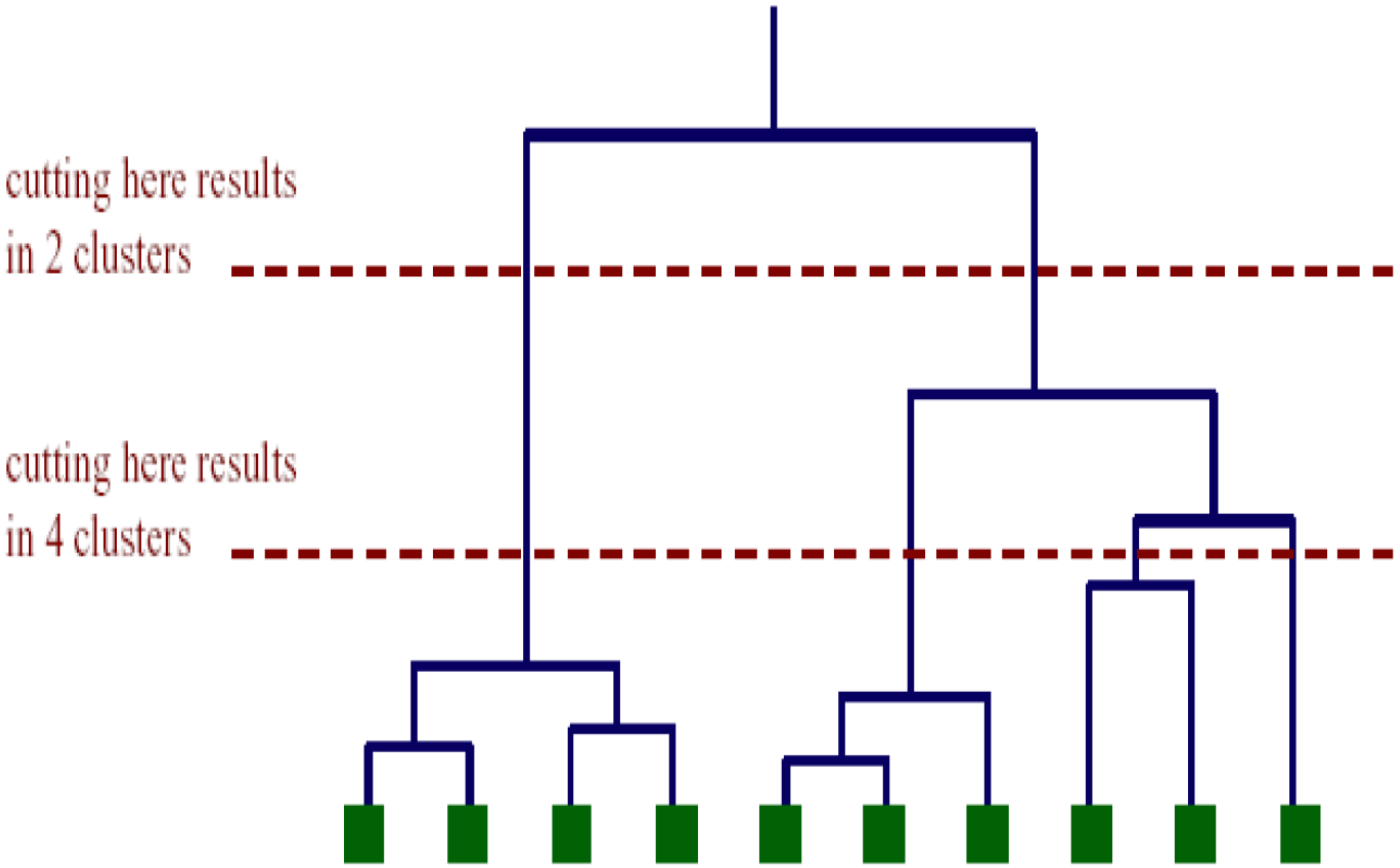
This appears trivial but in higher dimension our eyes cannot check the result easily.

Partitioning around medoids

- Use one representative point rather than average as centroid for each cluster
- Often more robust than k-means



From partitions from hierarchical clustering



How to determine the number of clusters

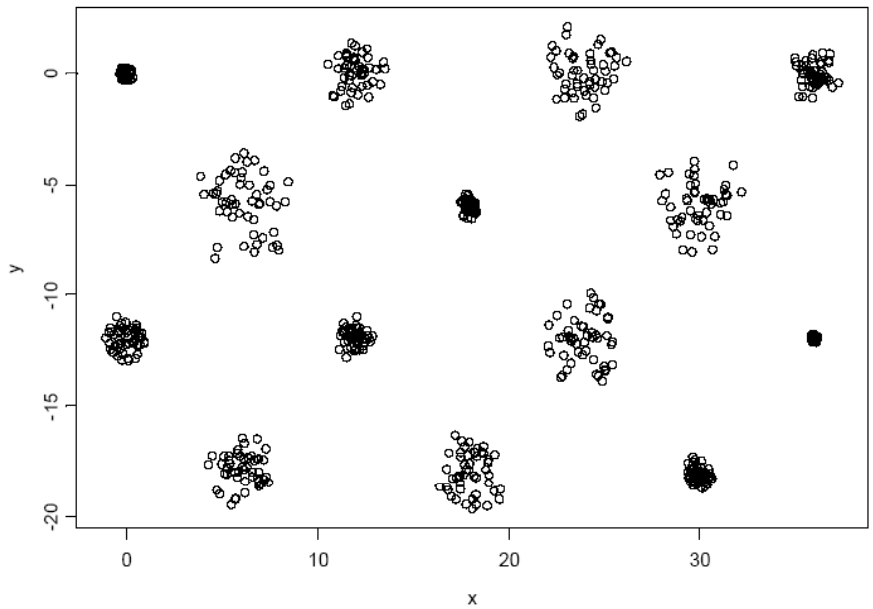
- W(k): total sum of squares within clusters
- B(k): sum of squares between clusters
- Calinski & Harabasz

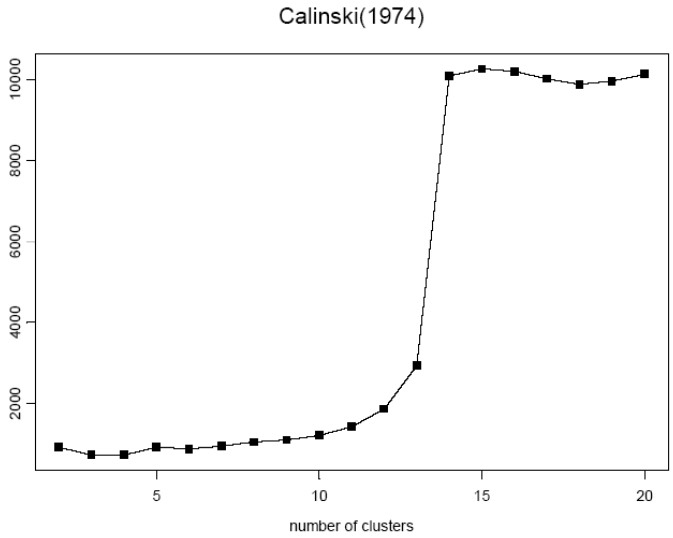
$$\max CH(k) = \frac{B(k)/(k-1)}{W(k)/(n-k)}$$

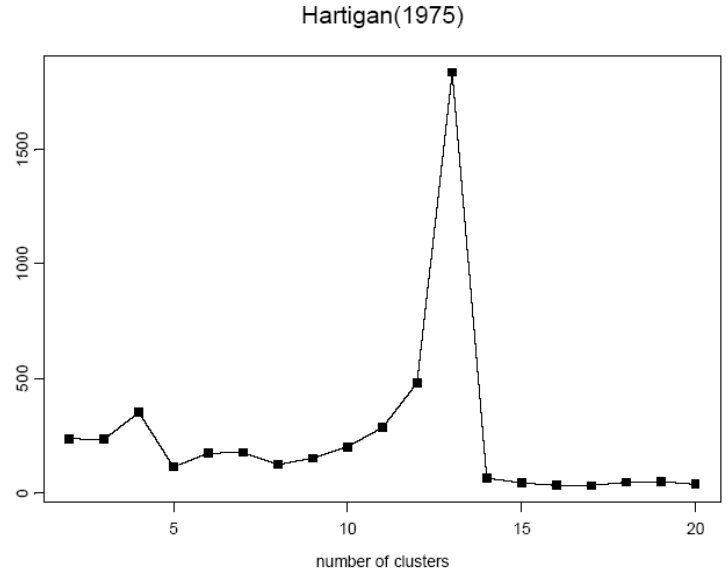
- Hartigan:

  Stop when H(k)<10

$$H(K) = \left(\frac{W(k)}{W(k+1)} - 1\right)(n-k-1)$$


y
x
0
-5
-10
-15
-20
0
10
20
30


Calinski(1974)
number of clusters
2000
4000
6000
8000
10000
5
10
15
20

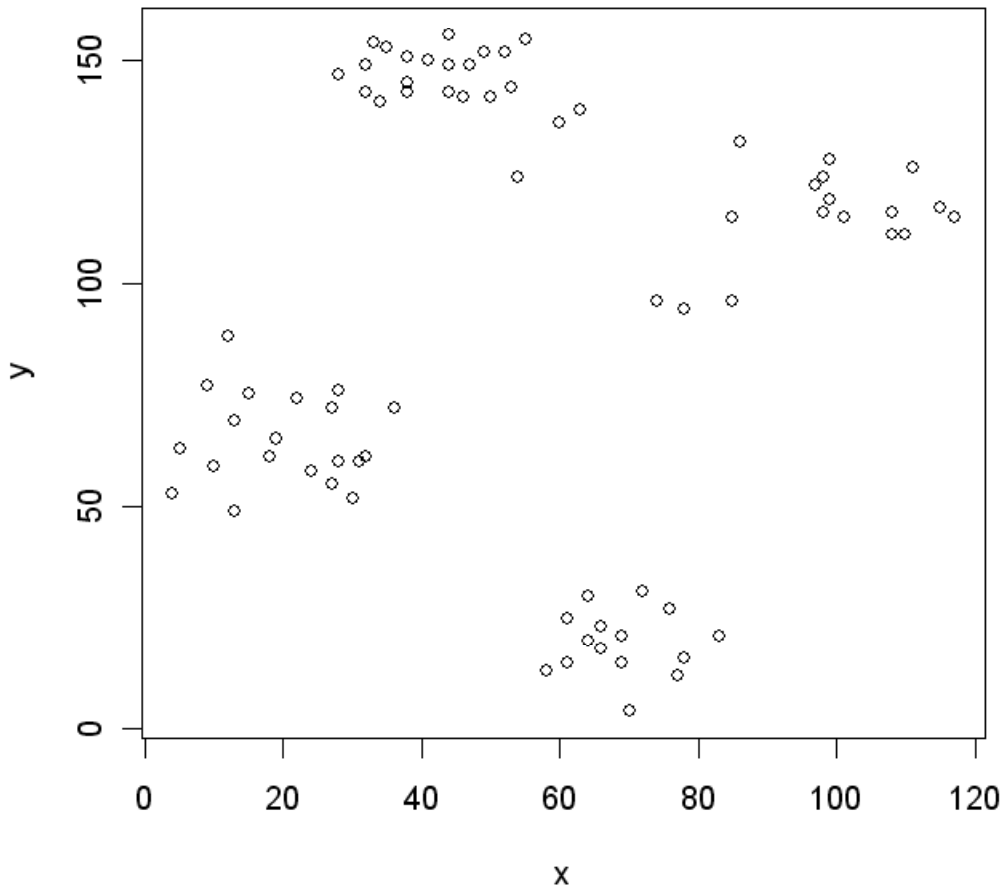

Hartigan(1975)
number of clusters
0
500
1000
1500
5
10
15
20

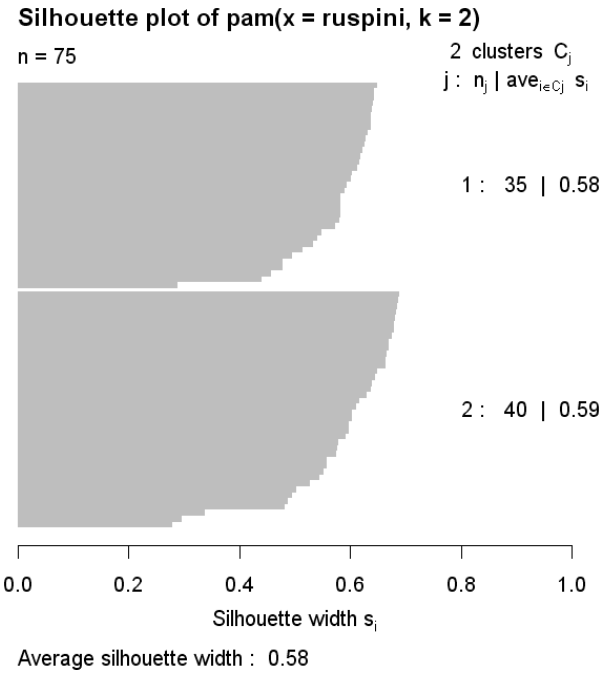
Silhouettes

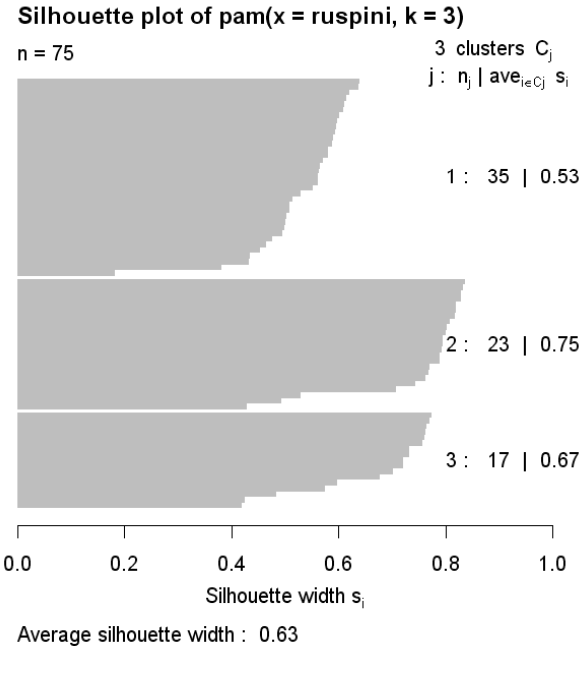
- Silhouette of gene *i* is defined as:

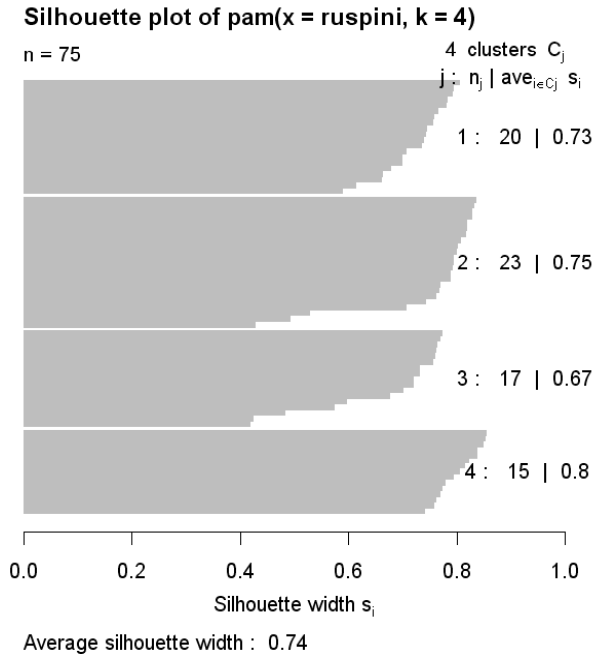
$$s(i) = \frac{b_i - a_i}{\max(a_i, b_i)}$$

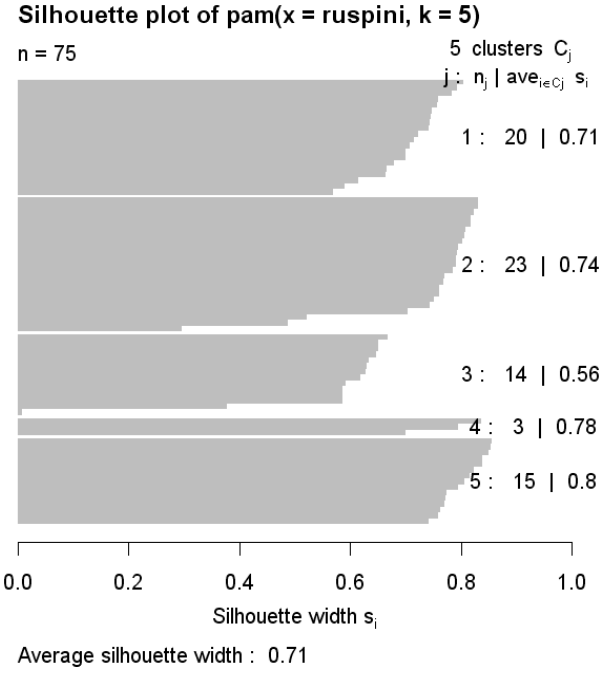
- a_i = average distance of gene *i* to other genes in same cluster
- b_i = average distance of gene *i* to genes in its nearest neighbor cluster


y
x
0
50
100
150
0
20
40
60
80
100
120

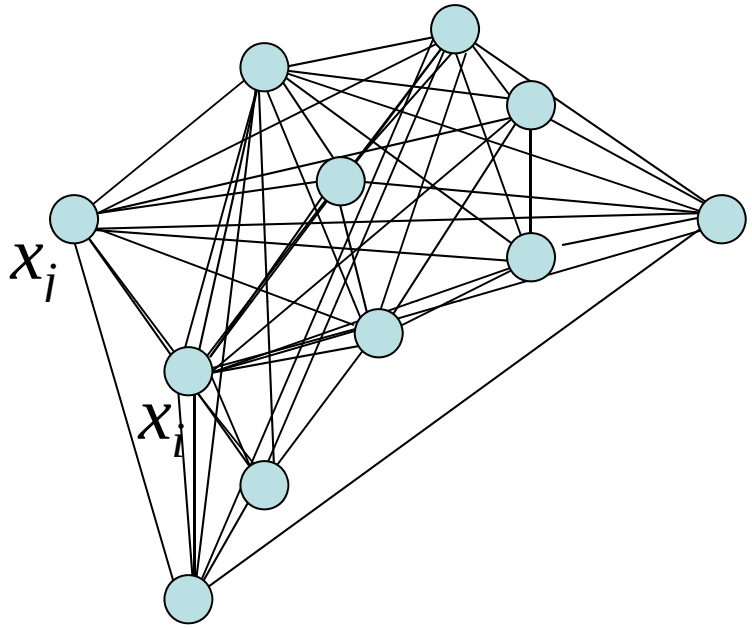

Silhouette plot of pam(x = ruspini, k = 2)
n = 75
2 clusters C_j
j : n_j | $ave_{i \in Cj}$ s_i
1 : 35 | 0.58
2 : 40 | 0.59
0.0
0.2
0.4
0.6
0.8
1.0
Silhouette width s_i
Average silhouette width : 0.58


Silhouette plot of pam(x = ruspini, k = 3)
n = 75
3 clusters C_j
j : n_j | $ave_{i \in Cj}$ s_i
1 : 35 | 0.53
2 : 23 | 0.75
3 : 17 | 0.67
0.0
0.2
0.4
0.6
0.8
1.0
Silhouette width s_i
Average silhouette width : 0.63


Silhouette plot of pam(x = ruspini, k = 4)
n = 75
4 clusters C_j
j : n_j | $ave_{i \in Cj}$ s_i
1 : 20 | 0.73
2 : 23 | 0.75
3 : 17 | 0.67
4 : 15 | 0.8
0.0
0.2
0.4
0.6
0.8
1.0
Silhouette width s_i
Average silhouette width : 0.74


Silhouette plot of pam(x = ruspini, k = 5)
n = 75
5 clusters C_j
j : n_j | $ave_{i \in Cj}$ s_i
1 : 20 | 0.71
2 : 23 | 0.74
3 : 14 | 0.56
4 : 3 | 0.78
5 : 15 | 0.8
0.0
0.2
0.4
0.6
0.8
1.0
Silhouette width s_i
Average silhouette width : 0.71

Within-Cluster Sum of Squares



$$\left\| x_i - x_j \right\|^2$$

Within-Cluster Sum of Squares

$$D_r = \sum_{i \in C_r} \sum_{j \in C_r} \left\| x_i - x_j \right\|^2$$

$$= 2n_r \sum_{i \in C_r} \left\| x_i - \bar{x} \right\|^2$$

$$W_k = \sum_{r=1}^{k} \frac{1}{2n_r} D_r$$

Measure of compactness of clusters

Gap statistic

If K is the “true” number of clusters:

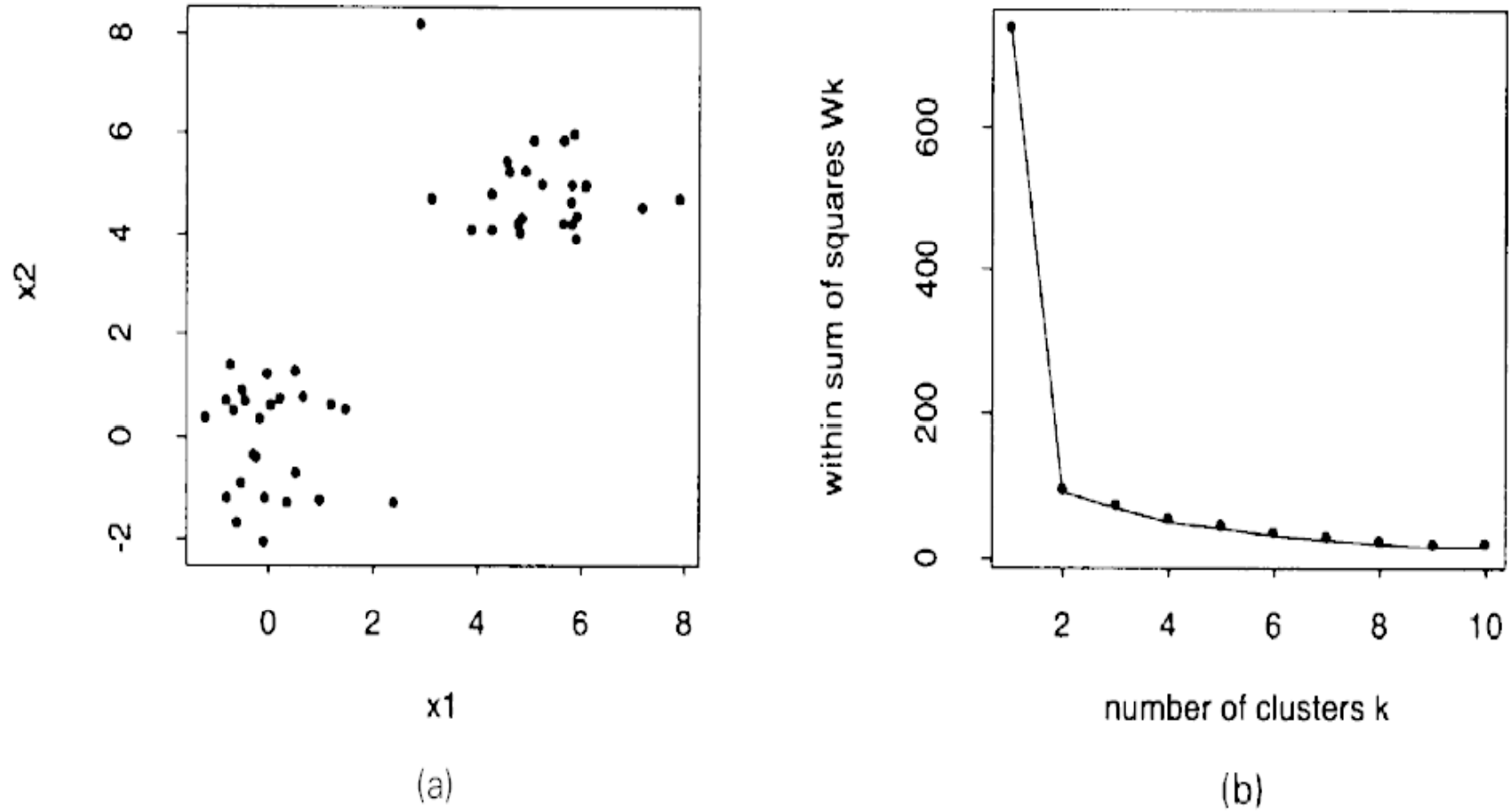
The within cluster sum of squares W(k) decrease faster when k<K, and slower when k>K

Pick K with largest drop in w(k)

Some helpful R code can be found here

http://www.stat.rutgers.edu/~rebecka/RCode/

Using W_k to determine # clusters

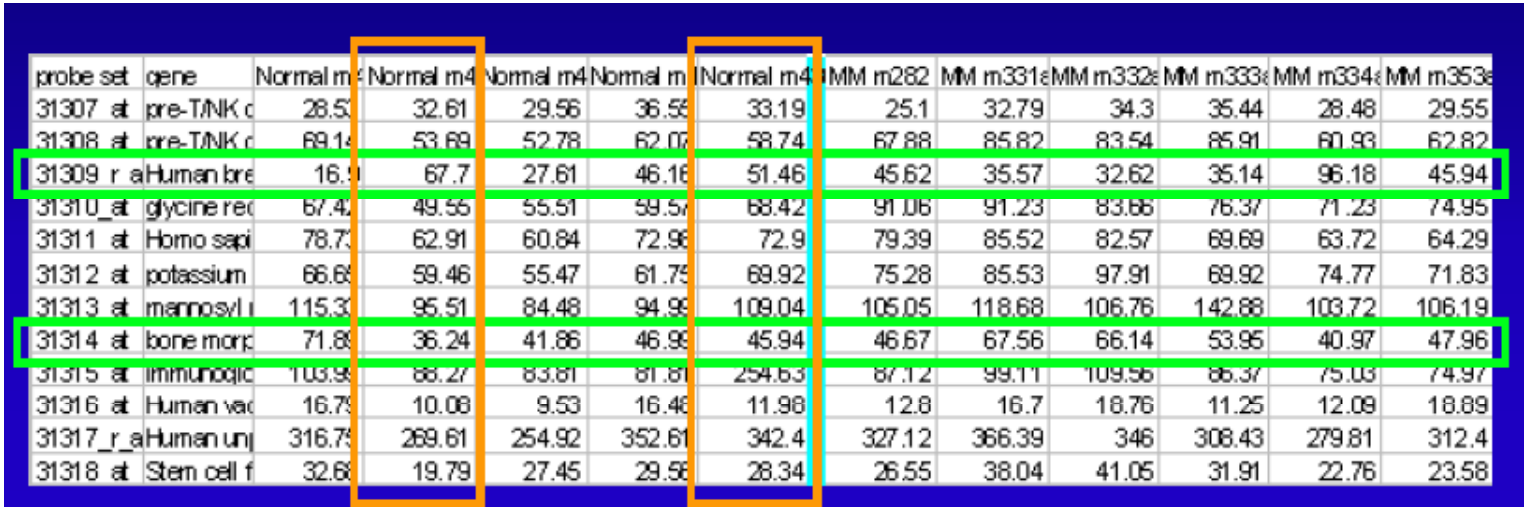


Idea of L-Curve Method: use the *k* corresponding to the "elbow"

(the most significant increase in goodness-of-fit)

Clustering gene expression

- Cluster Genes with similar expression profile over different samples or conditions
- Cluster samples with similar expression profile over all the genes



probe set	gene	Normal m	Normal m4	Normal m4	Normal m	Normal m4	MM m282	MM m331	MM m332	MM m333	MM m334	MM m353
31307_at	pre-T/NK c	28.5	32.61	29.56	36.5	33.19	25.1	32.79	34.3	35.44	28.48	29.55
31308_at	pre-T/NK c	69.1	53.69	52.78	62.0	58.74	67.88	85.82	83.54	85.91	80.93	62.82
31309_r_a	Human bre	16.9	67.7	27.61	46.1	51.46	45.62	35.57	32.62	35.14	96.18	45.94
31310_at	glycine rec	67.4	49.55	55.51	59.5	68.42	91.06	91.23	83.66	76.37	71.23	74.95
31311_at	Homo sapi	78.7	62.91	60.84	72.9	72.9	79.39	85.52	82.57	69.69	63.72	64.29
31312_at	potassium	66.6	59.46	55.47	61.7	69.92	75.28	85.53	97.91	69.92	74.77	71.83
31313_at	mannosyl	115.3	95.51	84.48	94.9	109.04	105.05	118.68	106.76	142.88	103.72	106.19
31314_at	bone morp	71.8	36.24	41.86	46.9	45.94	46.67	67.56	66.14	53.95	40.97	47.96
31315_at	immunoglo	103.9	88.27	83.81	81.8	254.63	87.12	99.11	109.56	86.37	75.03	74.97
31316_at	Human vac	16.7	10.08	9.53	16.4	11.98	12.8	16.7	18.76	11.25	12.09	18.89
31317_r_a	Human unp	316.7	269.61	254.92	352.6	342.4	327.12	366.39	346	308.43	279.81	312.4
31318_at	Stem cell f	32.6	19.79	27.45	29.5	28.34	26.55	38.04	41.05	31.91	22.76	23.58

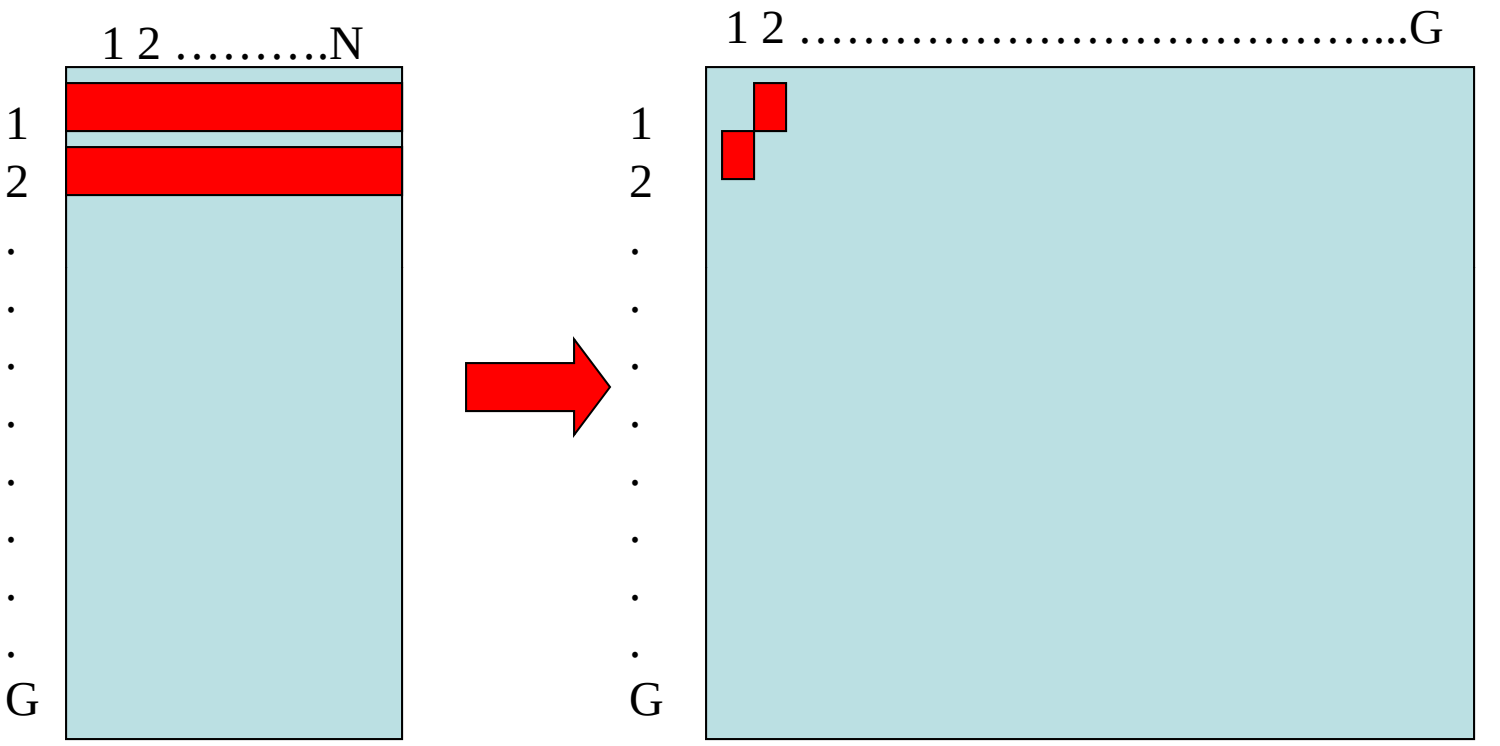
More specifically….

- Cluster analysis arranges samples and genes into groups based on their expression levels.
- This arrangement is determined purely by the measured distance or similarity between samples and genes.
- Arrangements are sensitive to choice of distance
  - Outliers
  - Distance mis-specification
- In hierarchical clustering, the VISUALIZTION of the arrangement (the dendrogram) is not unique!
  - Just because two samples are situated next to each other does not mean that they are similar.
  - Closer scrutiny is needed to interpret dendrogram than is usually given

Distance and Similarity

- Every clustering method is based **solely** on the measure of distance or similarity.
- E.g. correlation: measures linear association between two samples or genes.
  - What if data are not properly transformed?
  - What if there are outliers?
  - What if there are saturation effects?
- Even with large number of samples, bad measure of distance or similarity will not be helped.

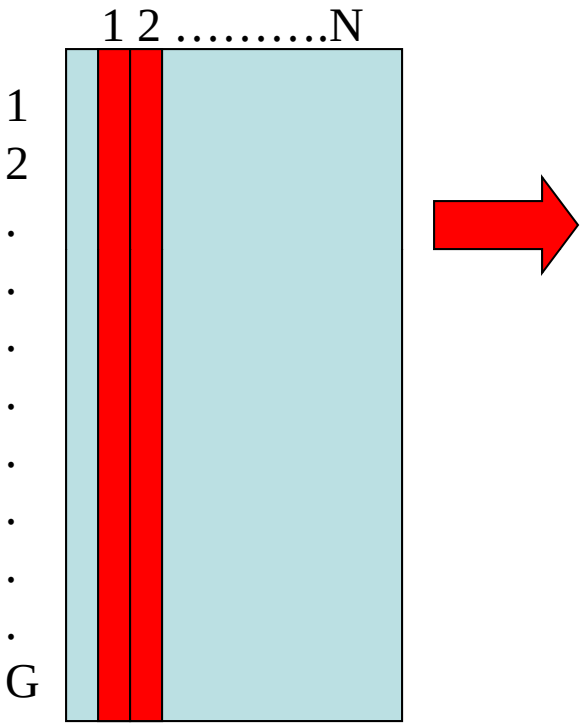
The similarity/distance matrices



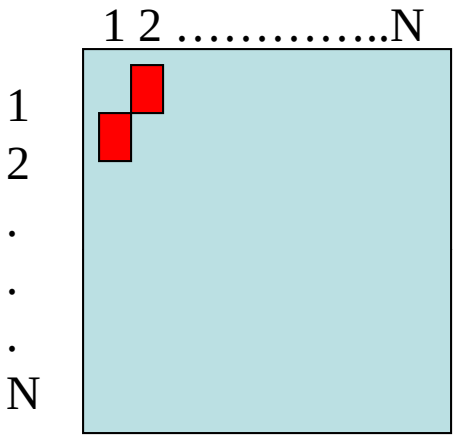
DATA MATRIX

GENE SIMILARITY MATRIX

The similarity/distance matrices



DATA MATRIX



SAMPLE SIMILARITY MATRIX

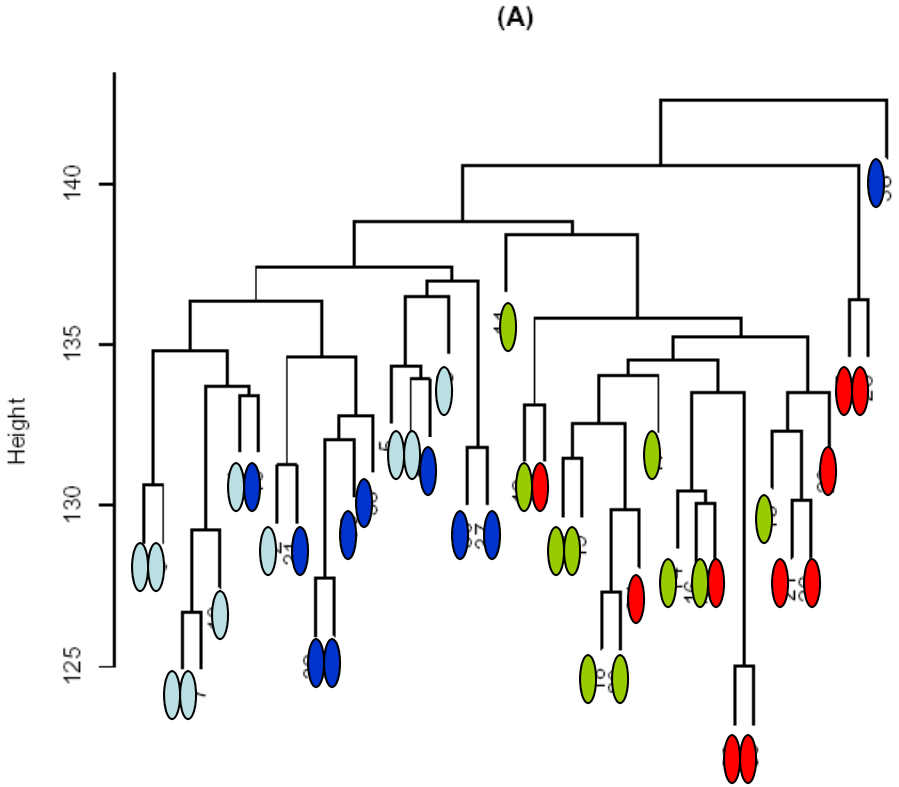
How to make a hierarchical clustering

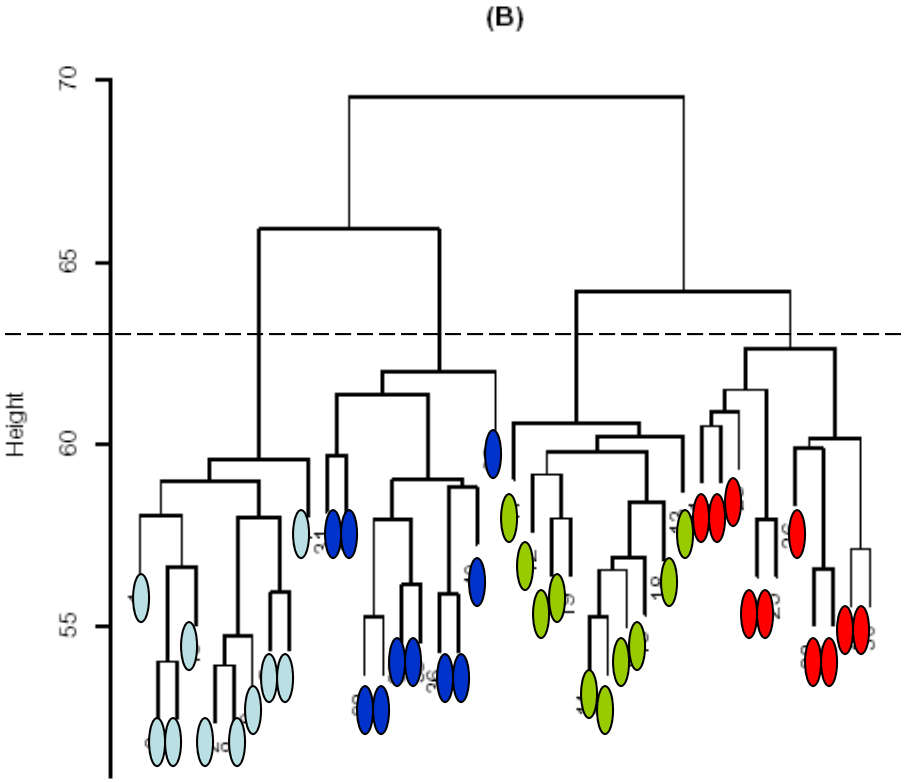
1. Choose samples and genes to include in cluster analysis
2. Choose similarity/distance metric
3. Choose clustering direction (top-down or bottom-up)
4. Choose linkage method (if bottom-up)
5. Calculate dendrogram
6. Choose height/number of clusters for interpretation
7. Assess cluster fit and stability
8. Interpret resulting cluster structure

1. Choose samples and genes to include

- Important step!
- Do you want housekeeping genes included?
- What to do about replicates from the same individual/tumor?
- Genes that contribute noise will affect your results.
- Including all genes: dendrogram can’t all be seen at the same time.
- Perhaps screen the genes?

Simulated Data with 4 clusters: 1-10, 11-20, 21-30, 31-40





A: 450 relevant genes plus
450 “noise” genes.

B: 450 relevant genes.

2. Choose similarity/distance matrix

- Think hard about this step!
- Remember: garbage in ➔ garbage out
- The metric that you pick should be a valid measure of the distance/similarity of genes.
- Examples:
  - Applying correlation to highly skewed data will provide misleading results.
  - Applying Euclidean distance to data measured on categorical scale will be invalid.
- Not just “wrong”, but which makes most sense

Some correlations to choose from

- Pearson Correlation:

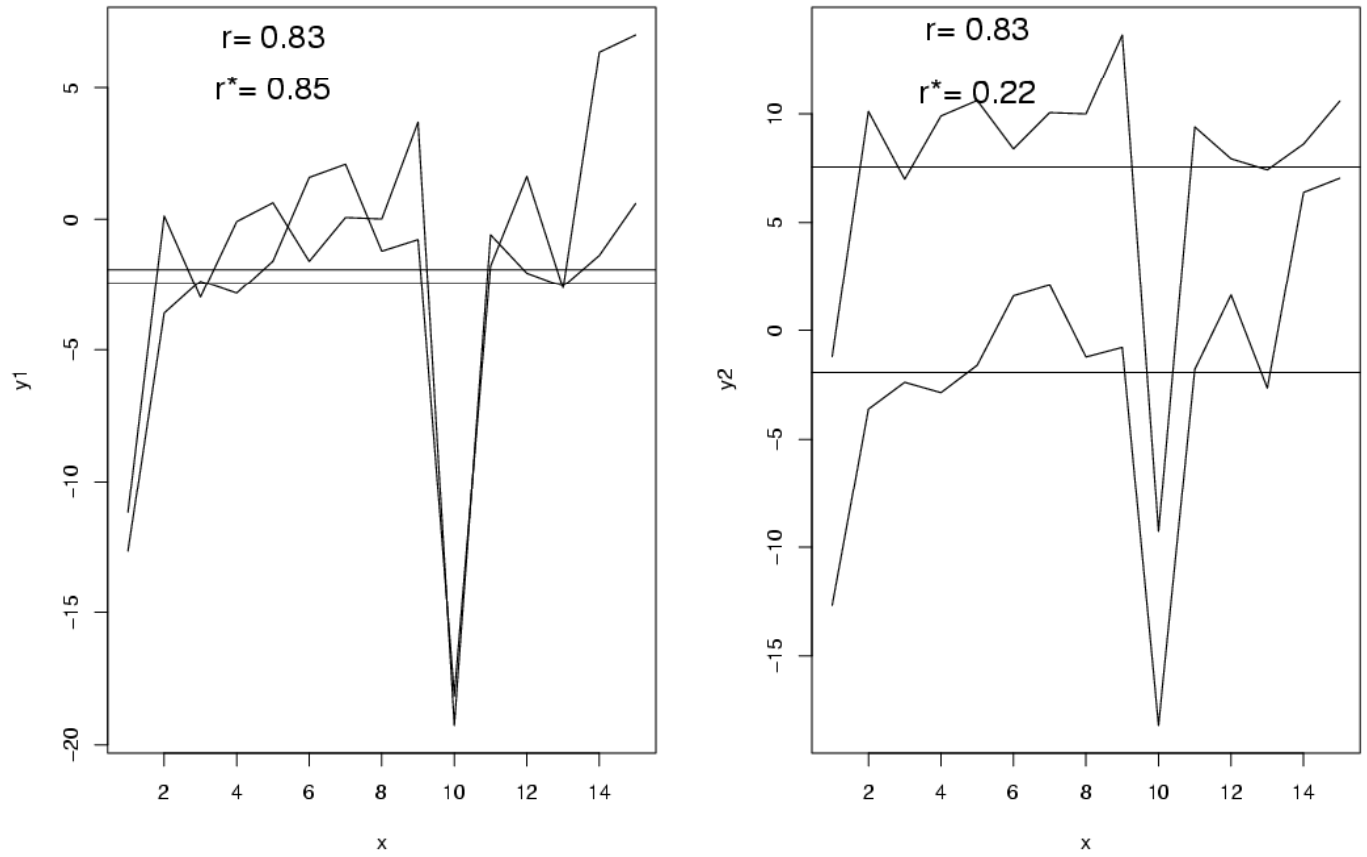
$$s(x_1, x_2) = \frac{\sum_{k=1}^{K} (x_{1k} - \bar{x}_1)(x_{2k} - \bar{x}_2)}{\sqrt{\sum_{k=1}^{K} (x_{1k} - \bar{x}_1)^2 \sum_{k=1}^{K} (x_{2k} - \bar{x}_2)^2}}$$

- Uncentered Correlation:

$$s(x_1, x_2) = \frac{\sum_{k=1}^{K} x_{1k} x_{2k}}{\sqrt{\sum_{k=1}^{K} x_{1k}^2 \sum_{k=1}^{K} x_{2k}^2}}$$

- Absolute Value of Correlation:

$$s(x_1, x_2) = \left| \frac{\sum_{k=1}^{K} (x_{1k} - \bar{x}_1)(x_{2k} - \bar{x}_2)}{\sqrt{\sum_{k=1}^{K} (x_{1k} - \bar{x}_1)^2 \sum_{k=1}^{K} (x_{2k} - \bar{x}_2)^2}} \right|$$



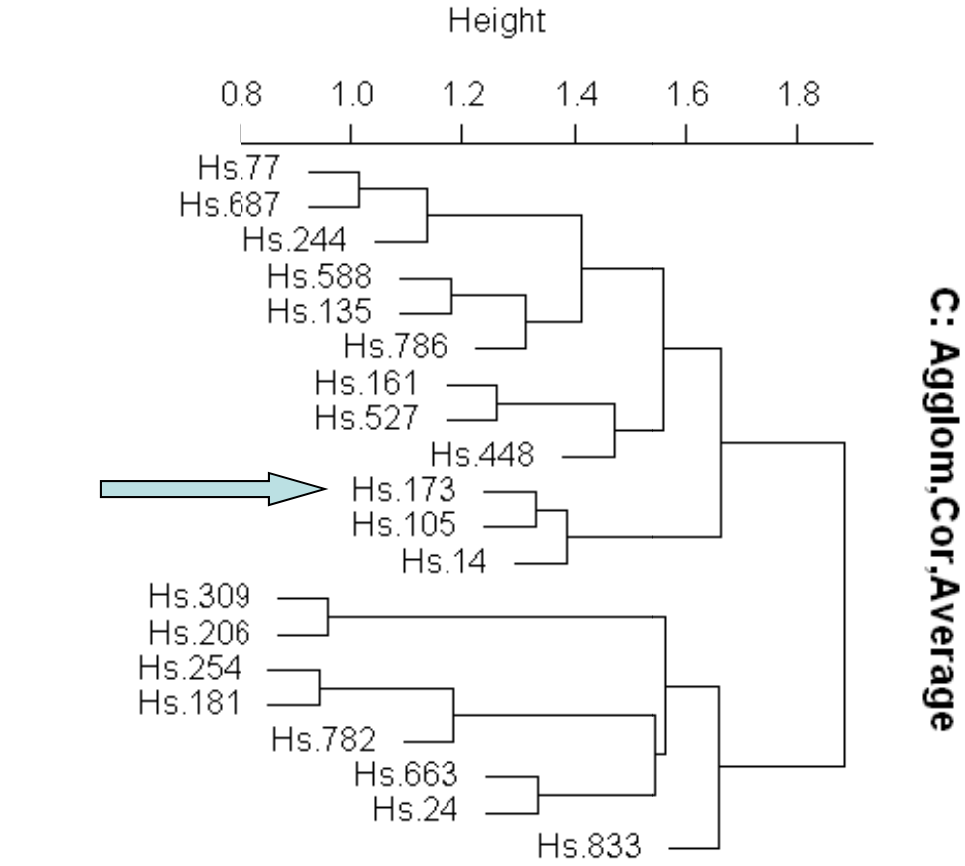
The difference is that, if you have two vectors X and Y with identical shape, but which are offset relative to each other by a fixed value, they will have a standard Pearson correlation (centered correlation) of 1 but will not have an uncentered correlation of 1.

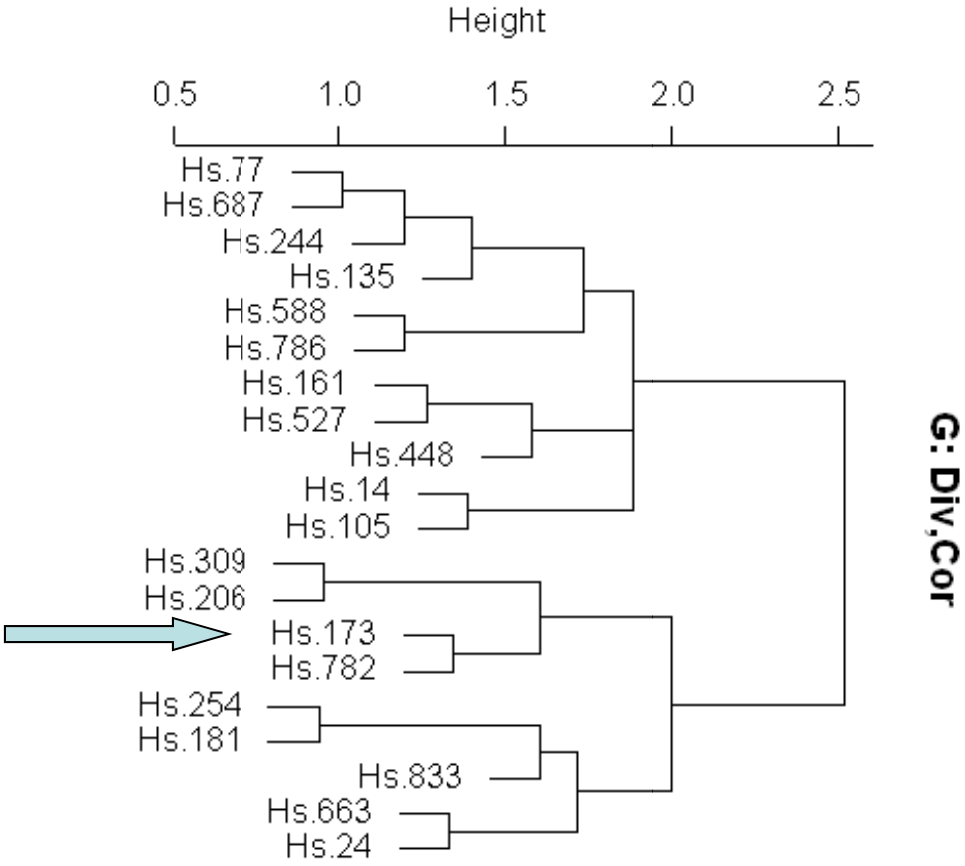
3. Choose clustering direction (top-down or bottom-up)

- Agglomerative clustering (bottom-up)
  - Starts with as each gene in its own cluster
  - Joins the two most similar clusters
  - Then, joins next two most similar clusters
  - Continues until all genes are in one cluster
- Divisive clustering (top-down)
  - Starts with all genes in one cluster
  - Choose split so that genes in the two clusters are most similar (maximize “distance” between clusters)
  - Find next split in same manner
  - Continue until all genes are in single gene clusters

Which to use?

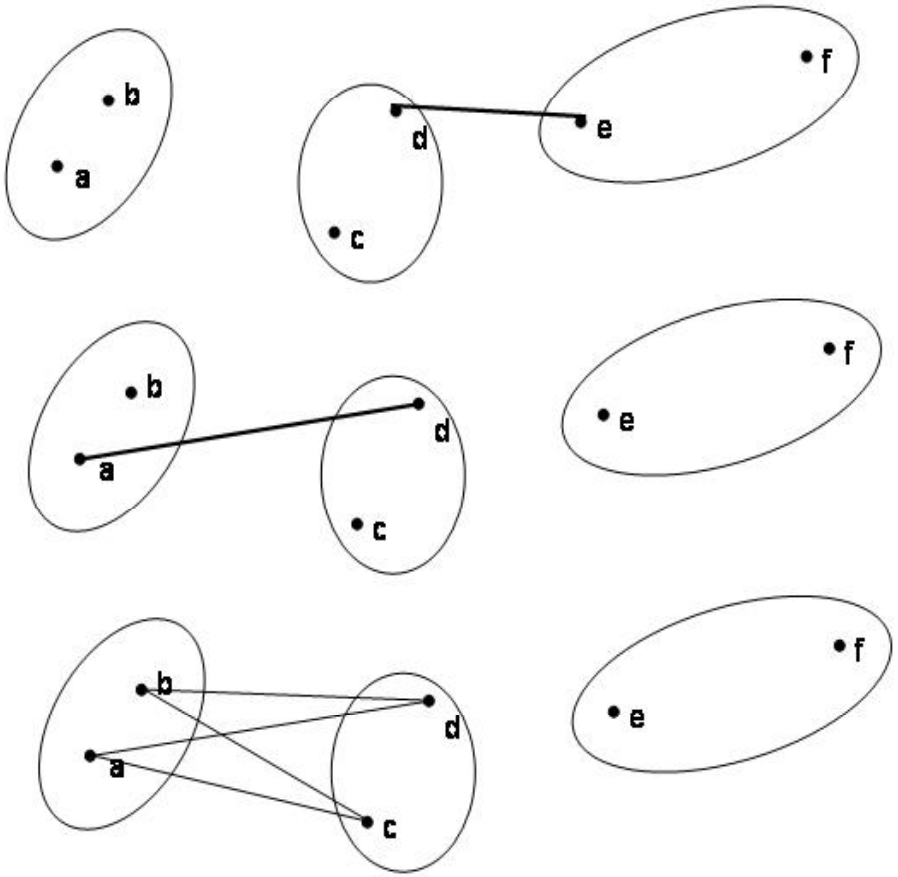
- Both are **only** 'step-wise' optimal: at each step the optimal split or merge is performed
- This does not imply that the final cluster structure is optimal!
- Agglomerative/Bottom-Up
  - Computationally simpler, and more available.
  - More "precision" at bottom of tree
  - When looking for small clusters and/or many clusters, use agglomerative
- Divisive/Top-Down
  - More "precision" at top of tree.
  - When looking for large and/or few clusters, use divisive
- **In gene expression applications, divisive makes more sense**.
- Results ARE sensitive to choice!


Height
0.8
1.0
1.2
1.4
1.6
1.8
Hs.77
Hs.687
Hs.244
Hs.588
Hs.135
Hs.786
Hs.161
Hs.527
Hs.448
Hs.173
Hs.105
Hs.14
Hs.309
Hs.206
Hs.254
Hs.181
Hs.782
Hs.663
Hs.24
Hs.833
C: Agglom,Cor,Average


Height
0.5
1.0
1.5
2.0
2.5
Hs.77
Hs.687
Hs.244
Hs.135
Hs.588
Hs.786
Hs.161
Hs.527
Hs.448
Hs.14
Hs.105
Hs.309
Hs.206
Hs.173
Hs.782
Hs.254
Hs.181
Hs.833
Hs.663
Hs.24
G: Div,Cor

4. Choose linkage method (if bottom-up)

- Single Linkage: join clusters whose distance between closest genes is smallest (elliptical)
- Complete Linkage: join clusters whose distance between furthest genes is smallest (spherical)
- Average Linkage: join clusters whose average distance is the smallest.



5. Calculate dendrogram
6. Choose height/number of clusters for interpretation

- In gene expression, we don't see "rule-based" approach to choosing cutoff very often.
- Tend to look for what makes a good story.
- There are more rigorous methods. (more later)
- "Homogeneity" and "Separation" of clusters can be considered. (Chen et al. Statistica Sinica, 2002)
- Other methods for assessing cluster fit can help determine a reasonable way to "cut" your tree.

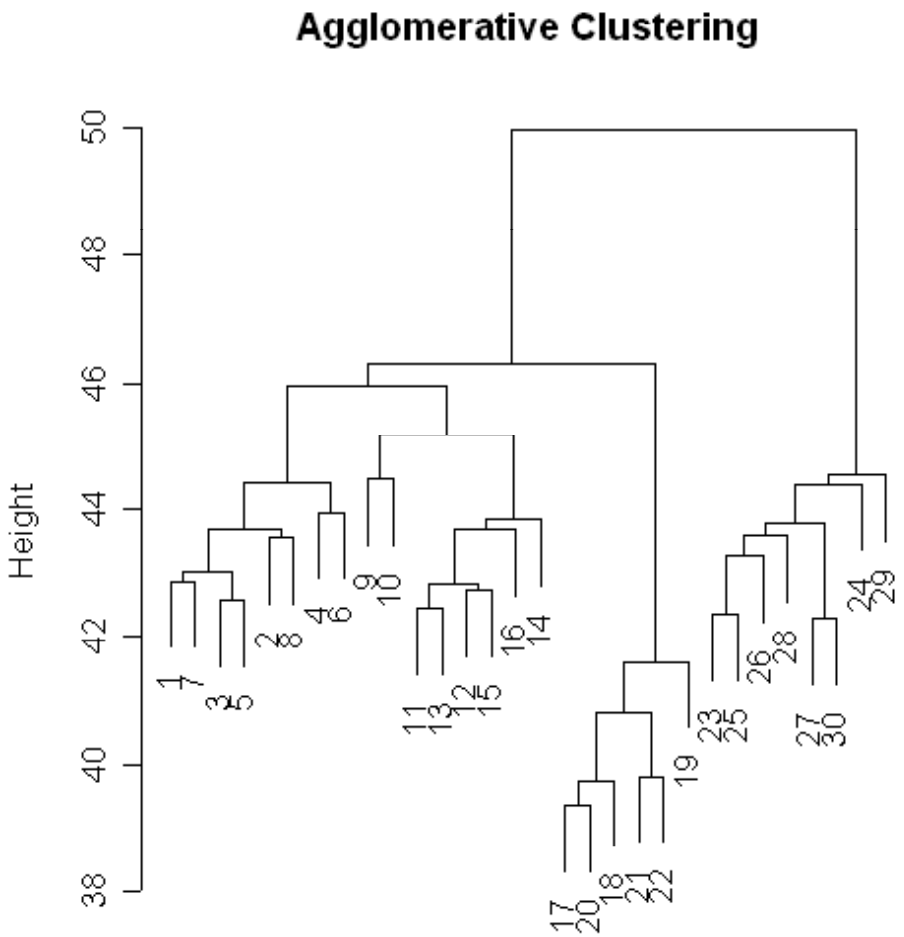
7. Assess cluster fit and stability

- Most often ignored.
- Cluster structure is treated as reliable and precise
- Usually the structure is rather unstable, at least at the bottom.
- Can be VERY sensitive to noise and to outliers
- Homogeneity and Separation
- Cluster Silhouettes and Silhouette coefficient: how similar genes within a cluster are to genes in other clusters (composite separation and homogeneity) (more later with K-medoids) (Rousseeuw Journal of Computation and Applied Mathematics, 1987)
- One approach is to try different approaches and see how tree differs.
  - Use average instead of complete linkage
  - Use divisive instead of agglomerative
  - Use Euclidean distance instead of correlation

Assess cluster fit and stability (FYI)

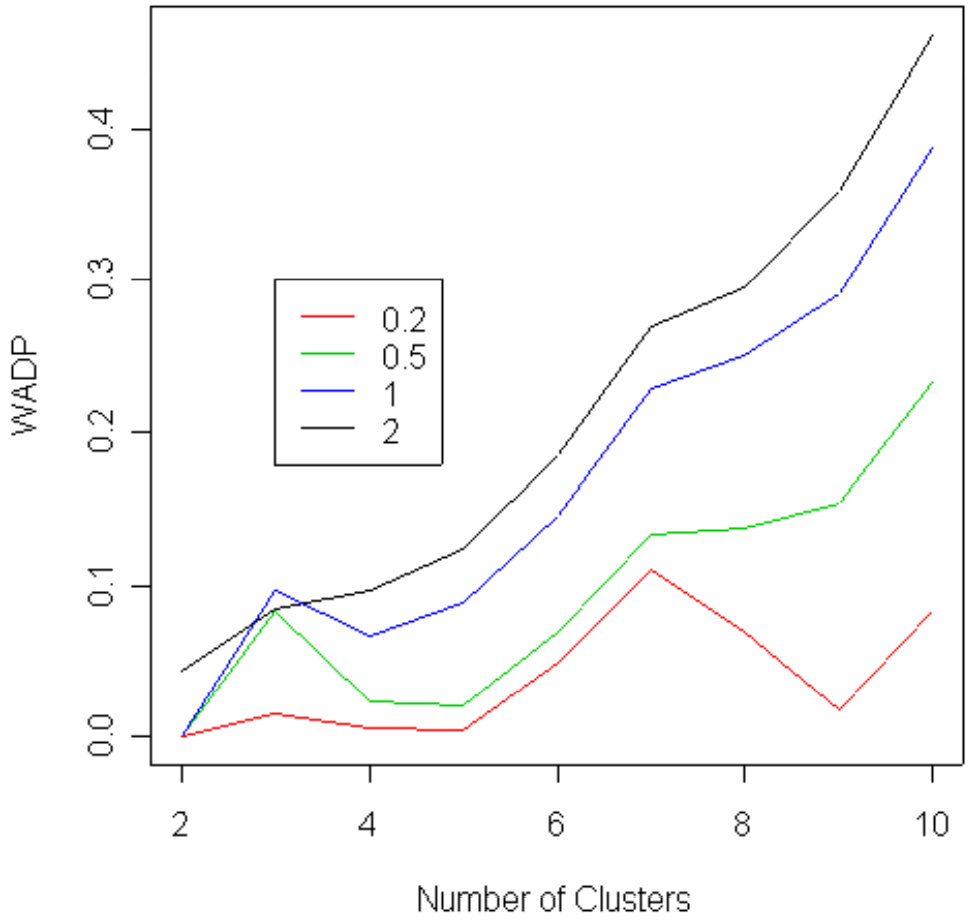
- WADP: Weighted Average Discrepant Pairs
  - Bittner et al. Nature, 2000
  - Fit cluster analysis using a dataset
  - Add random noise to the original dataset
  - Fit cluster analysis to the noise-added dataset
  - Repeat many times.
  - Compare the clusters across the noise-added datasets.
- Consensus Trees
  - Zhang and Zhao Functional and Integrative Genomics, 2000.
  - Use parametric bootstrap approach to sample new data using original dataset
  - Proceed similarly to WADP.
  - Look for nodes that are in a “majority” of the bootstrapped trees.
- More not mentioned…..

WADP:
Weighted Average Discrepancy Pairs



- Add perturbations to original data
- Calculate the number of paired samples that cluster together in the original cluster that didn't in the perturbed
- Repeat for every cutoff (i.e. for each k)
- Do iteratively
- Estimate for each k the proportion of discrepant pairs.

WADP



- Different levels of noise have been added
- By Bittner's recommendation, 1.0 is appropriate for our dataset
- But, not well-justified.
- External information would help determine level of noise for perturbation
- We look for largest k before WADP gets big.

Some Take-Home Points

- Clustering can be a useful exploratory tool
- Cluster results are very sensitive to noise in the data
- It is crucial to assess cluster structure to see how stable your result is
- Different clustering approaches can give quite different results
- For hierarchical clustering, interpretation is almost always subjective
- There is model based clustering too– but we will discuss it separately

Extra references

- **Cluster analysis and display of genome-wide expression patterns** Eisen et al, 1998, PNAS 95(25): 14863
- How does gene expression clustering work? D'haeseleer (2005) Nature Biotechnology 23, 1499 – 1501
- **A systematic comparison and evaluation of biclustering methods for gene expression data** Prelic et al (2006) Bioinformatics *22 (9): 1122-1129.*