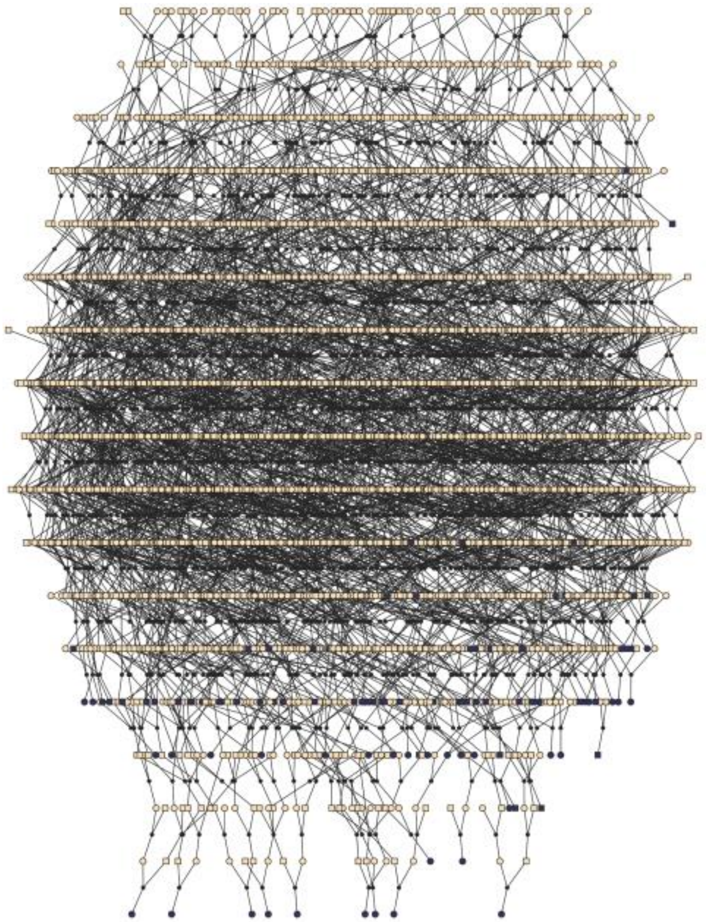
Other ways of clustering biological sequences

So far

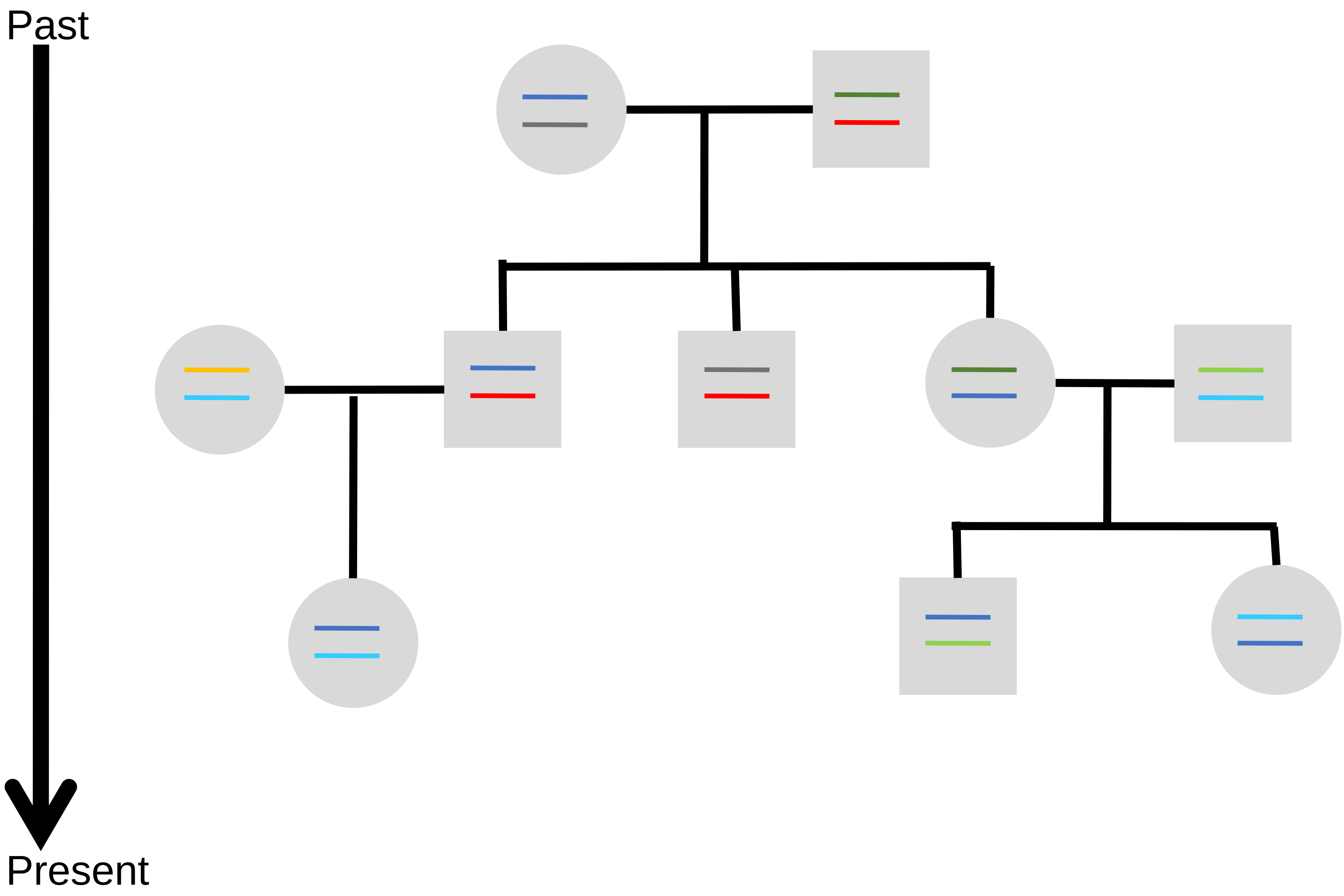
- Trees as a way to represent the evolutionary relationship between sequences
- We need methods for exploring the space of possible trees
    - Branch movements!
- Different ways of constructing a tree
    - Distance based methods
        - UPGMA
        - Neighbour-joining
        - ….
    - Character based methods
        - Maximum parsimony
        - Maximum likelihood
        - Bayesian

EVOLUTION: IS ALL ABOUT GENEALOGIES

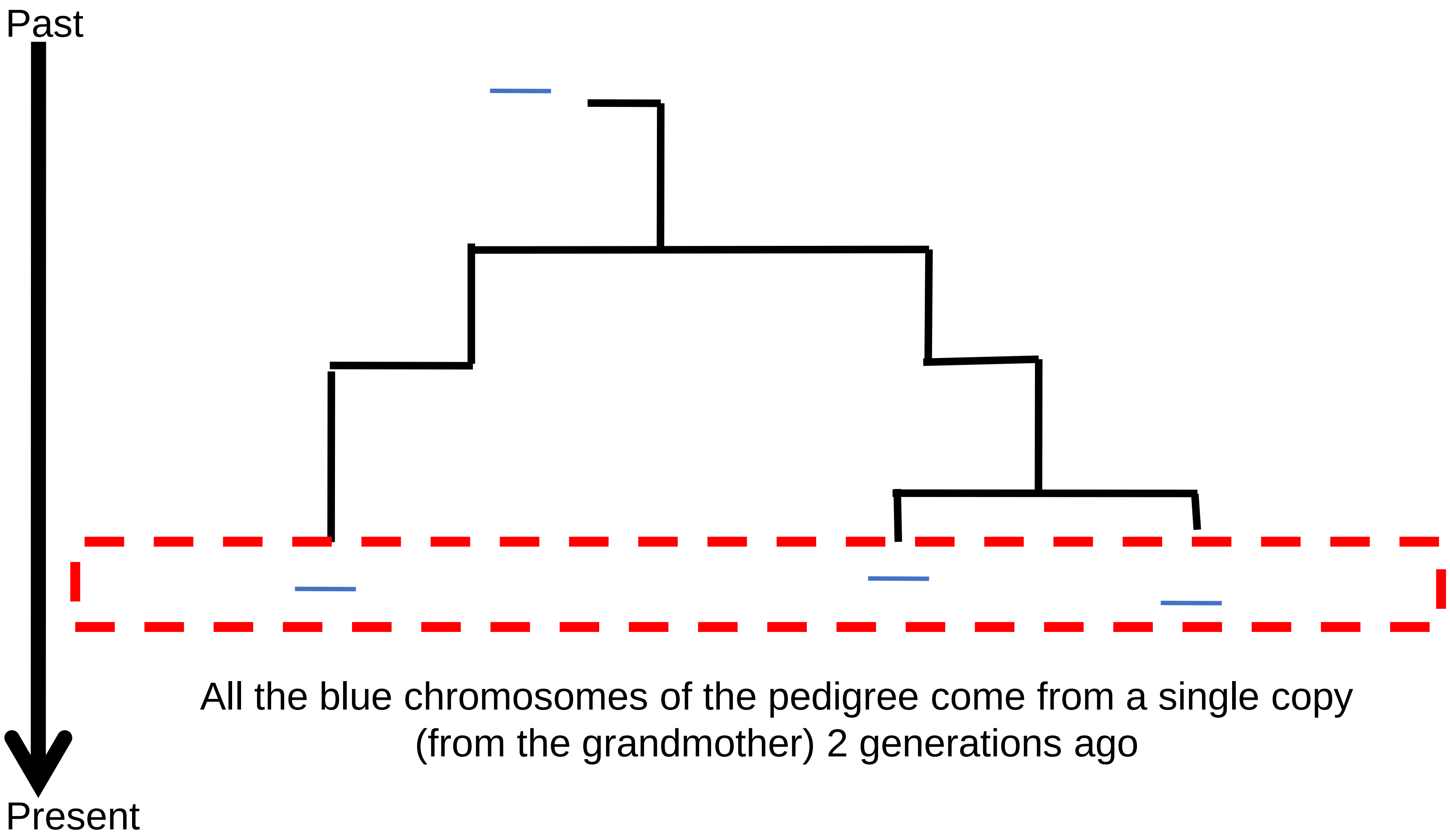


A population is a complex genealogy (but a genealogy anyway!)

EVOLUTION: IS ALL ABOUT GENE GENEALOGIES



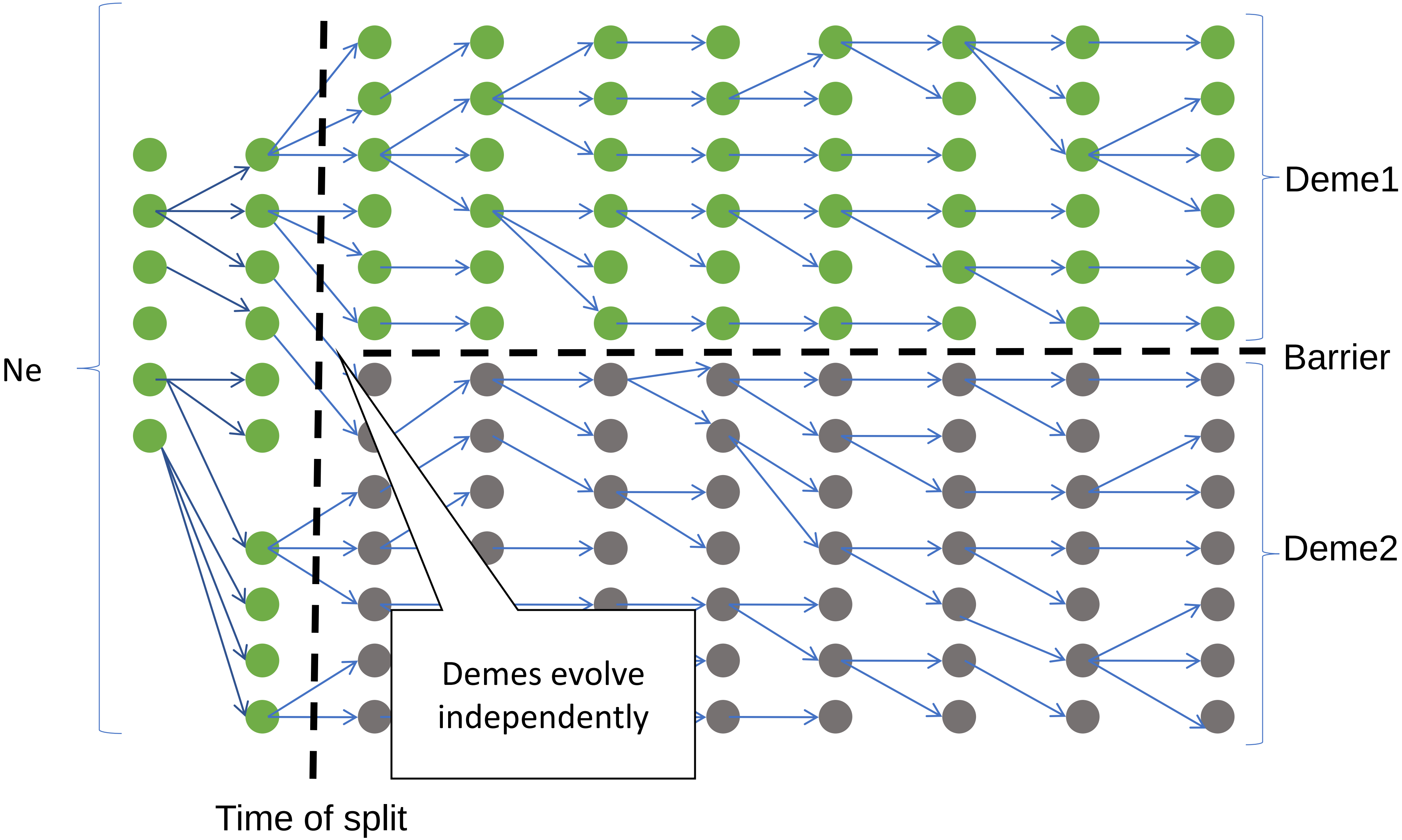
EVOLUTION: IS ALL ABOUT GENE GENEALOGIES



All the blue chromosomes of the pedigree come from a single copy (from the grandmother) 2 generations ago

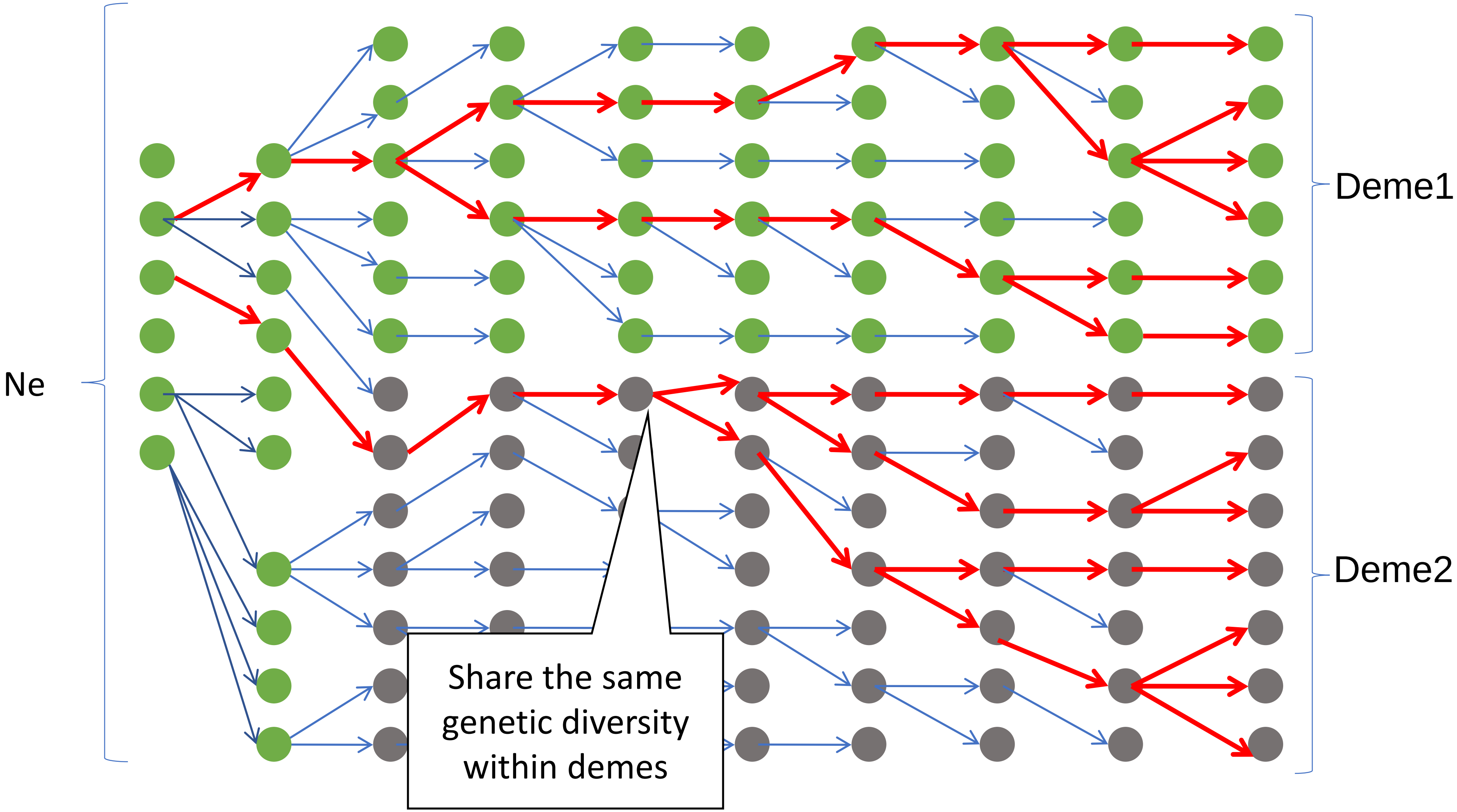
GOING BEYOND WRIGHT-FISHER

POPULATION SUBSTRUCTURE AND COALESCENCE



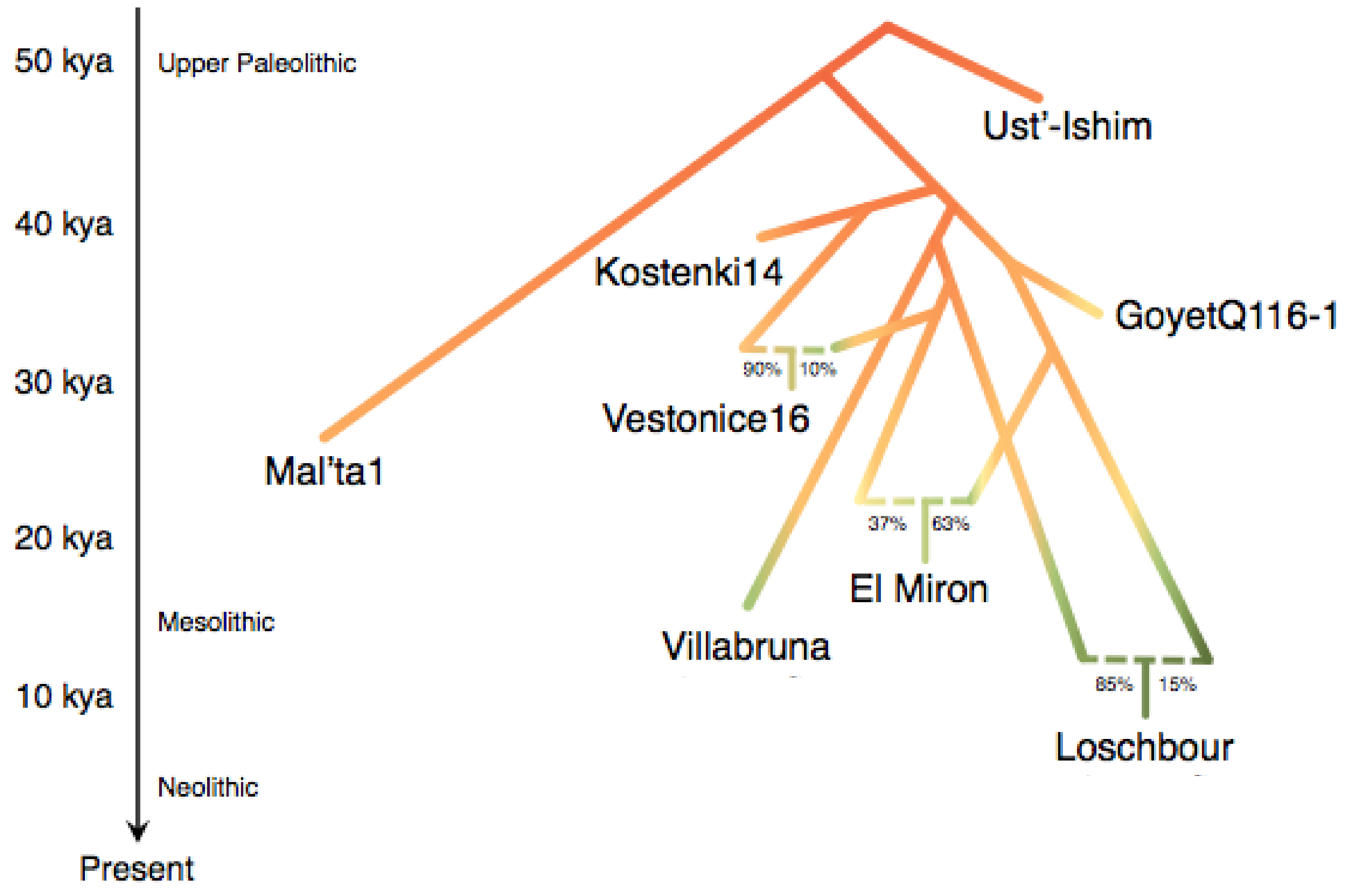
GOING BEYOND WRIGHT-FISHER

POPULATION SUBSTRUCTURE AND COALESCENCE



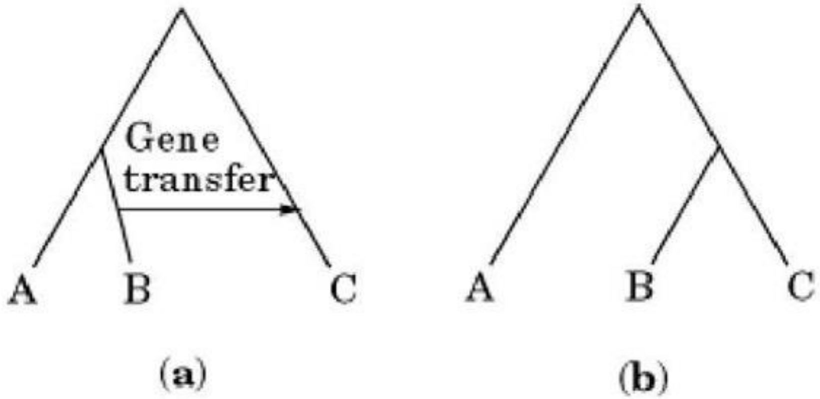
Share the same genetic diversity within demes

But do we have to think in trees?

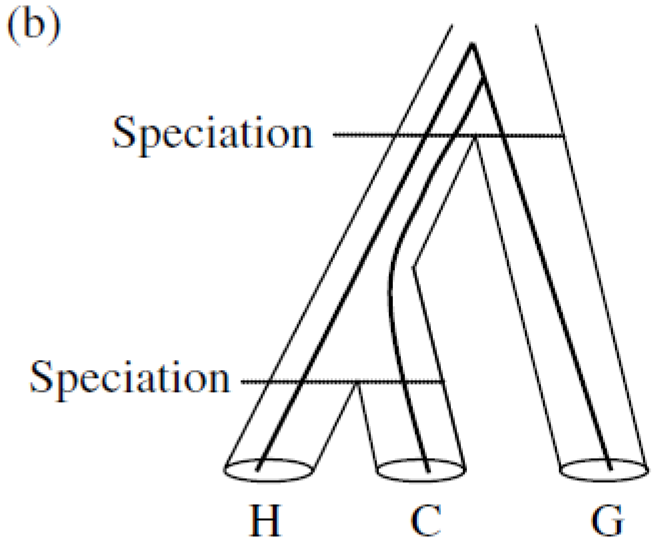


But do we have to think in trees?

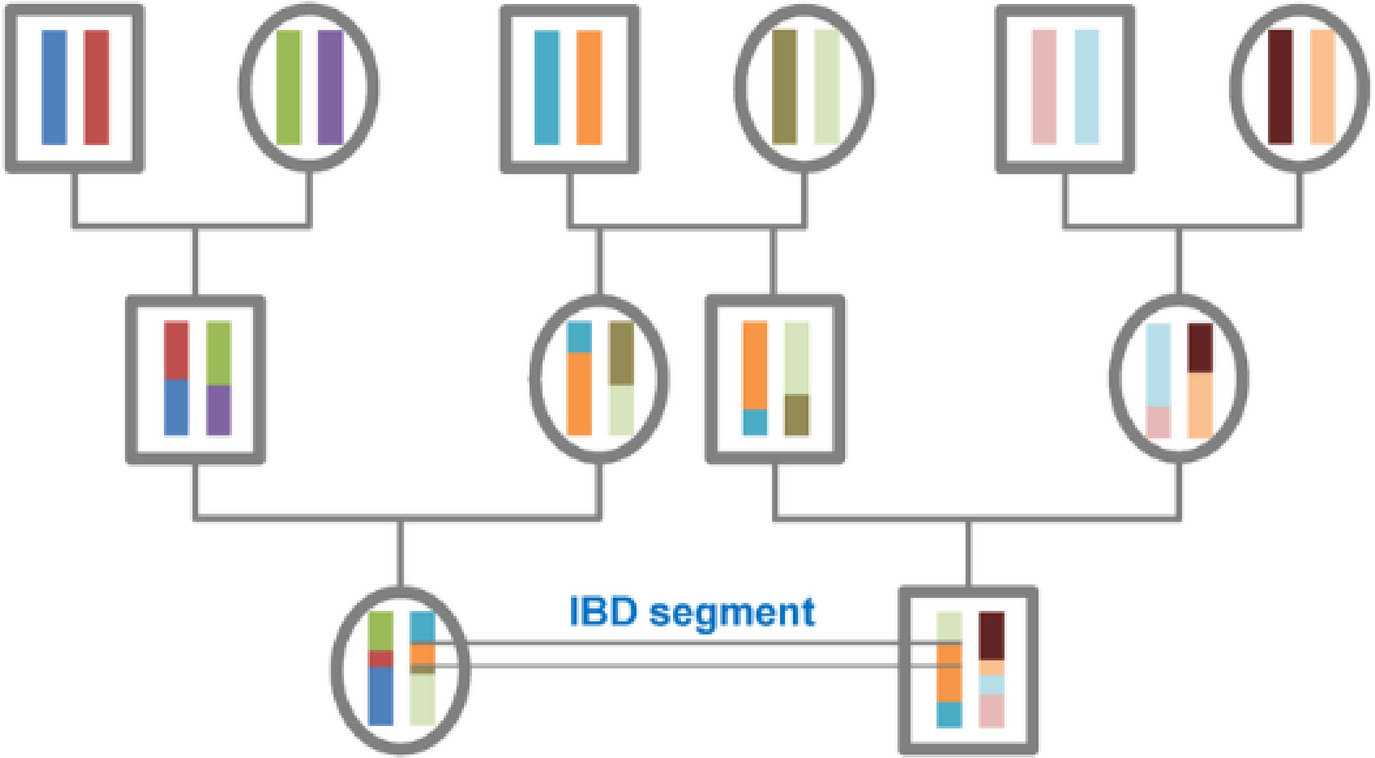
Horizontal gene transfer



Recent speciation

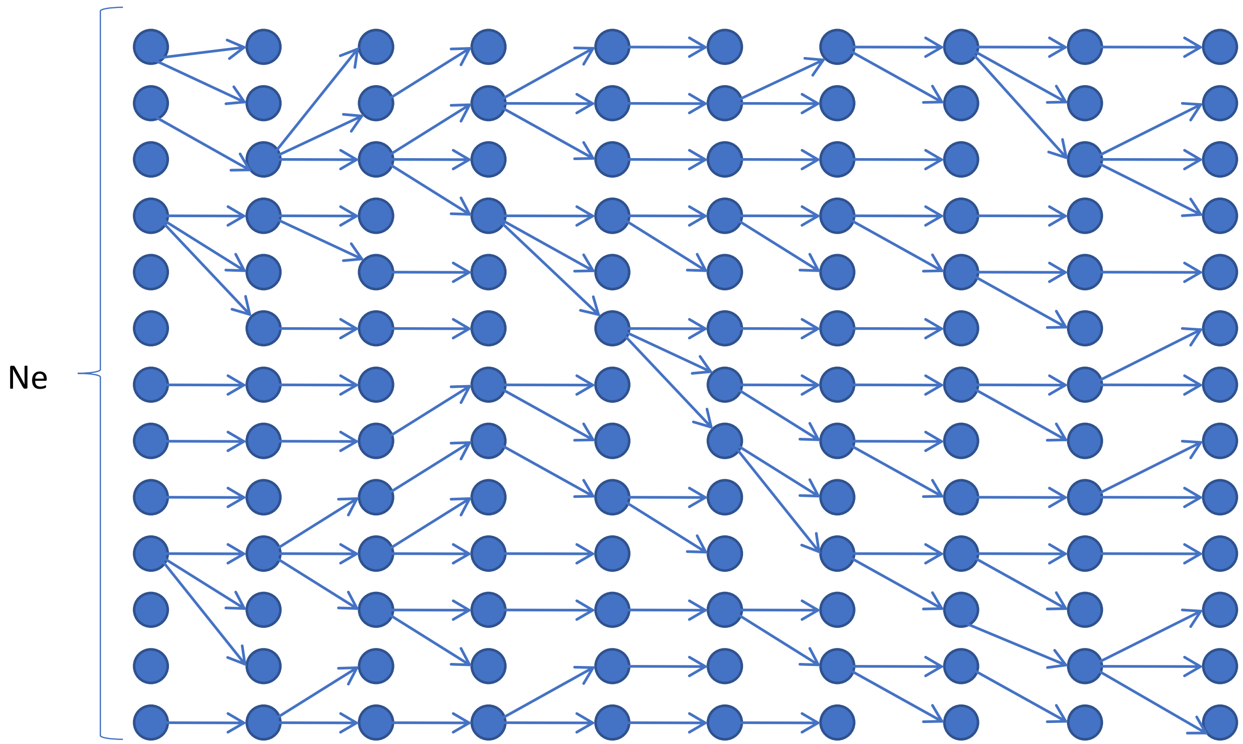


But do we have to think in trees?



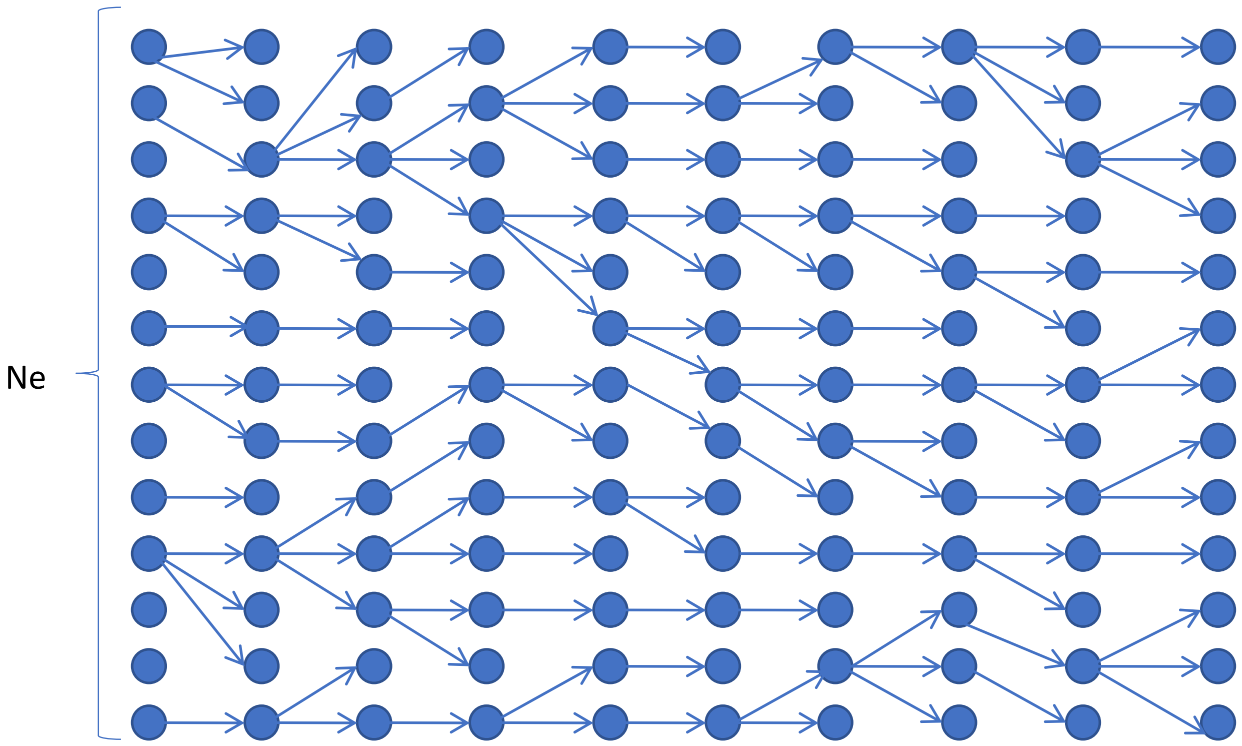
But do we have to think in trees?

Looking the gene genealogy of one locus in a pop



But do we have to think in trees?

Looking the gene genealogy of another (unlinked) locus in the same pop



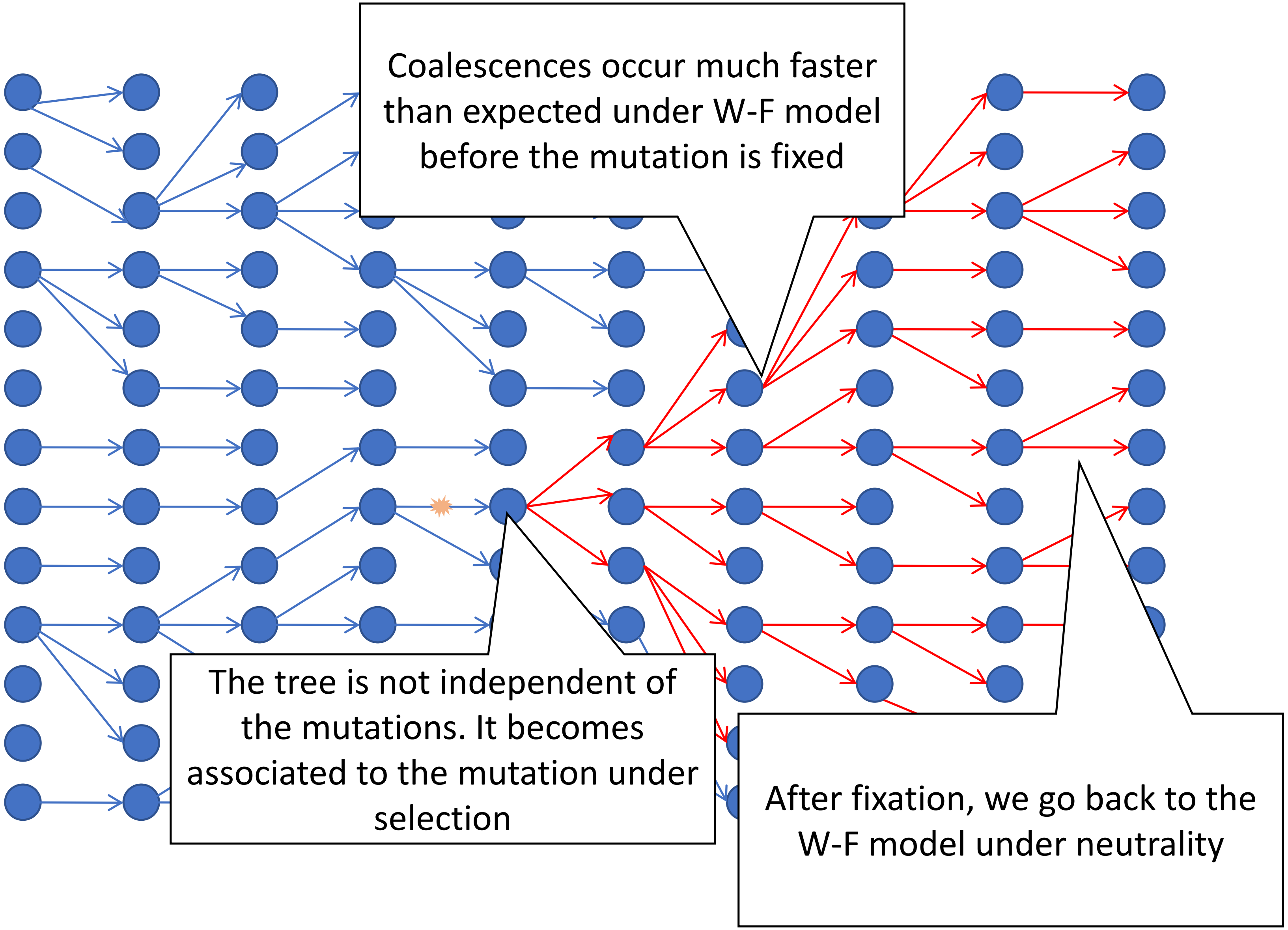
GOING BEYOND WRIGHT-FISHER

SELECTION: WORKS BY COMPARISON!

John Lydgate (1440): *Odyous of olde been comparisonis, And of comparisonis engendyrd is haterede. (Comparisons are odious)*

GOING BEYOND WRIGHT-FISHER

SELECTION AND COALESCENCE

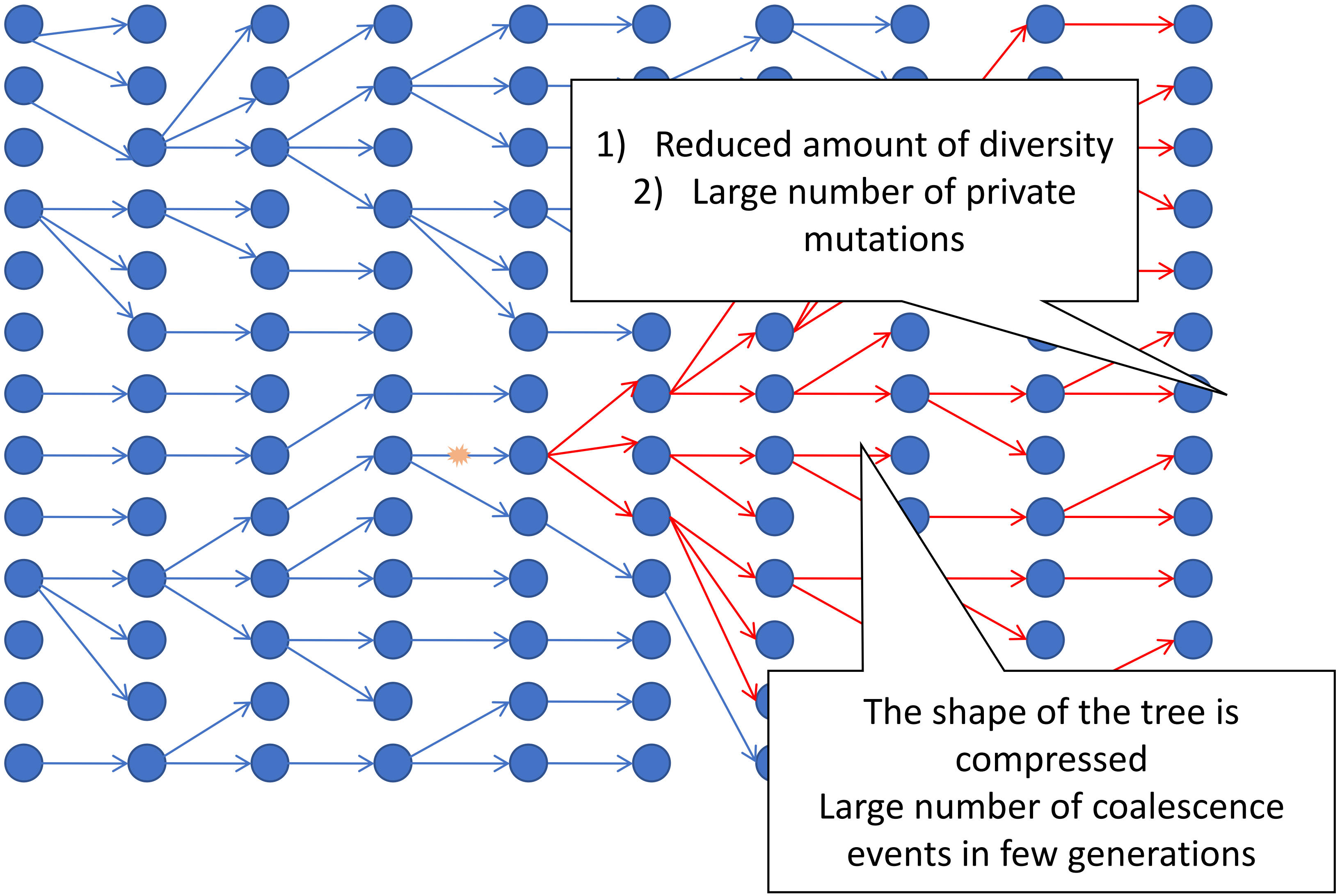


Coalescences occur much faster than expected under W-F model before the mutation is fixed

The tree is not independent of the mutations. It becomes associated to the mutation under selection

After fixation, we go back to the W-F model under neutrality

Coalescence under strong selection



1) Reduced amount of diversity
2) Large number of private mutations

The shape of the tree is compressed
Large number of coalescence events in few generations

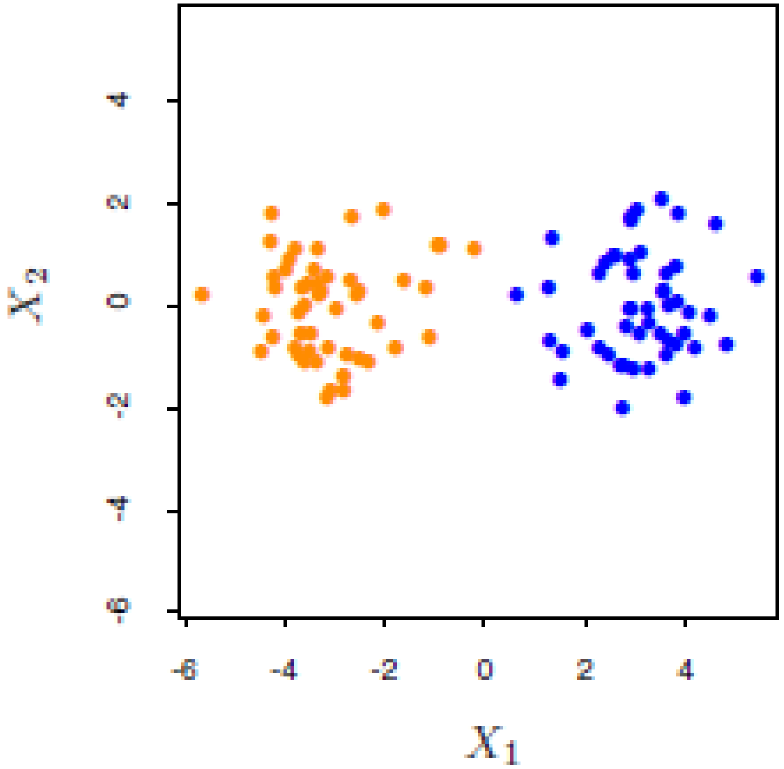
Which is the purpose of a tree? (What do we want to represent when we construct a tree from a sequence?)

Is there any other way of reducing the dimensionality of the data to highlight the inner structure represented by the sequences? And of clustering?

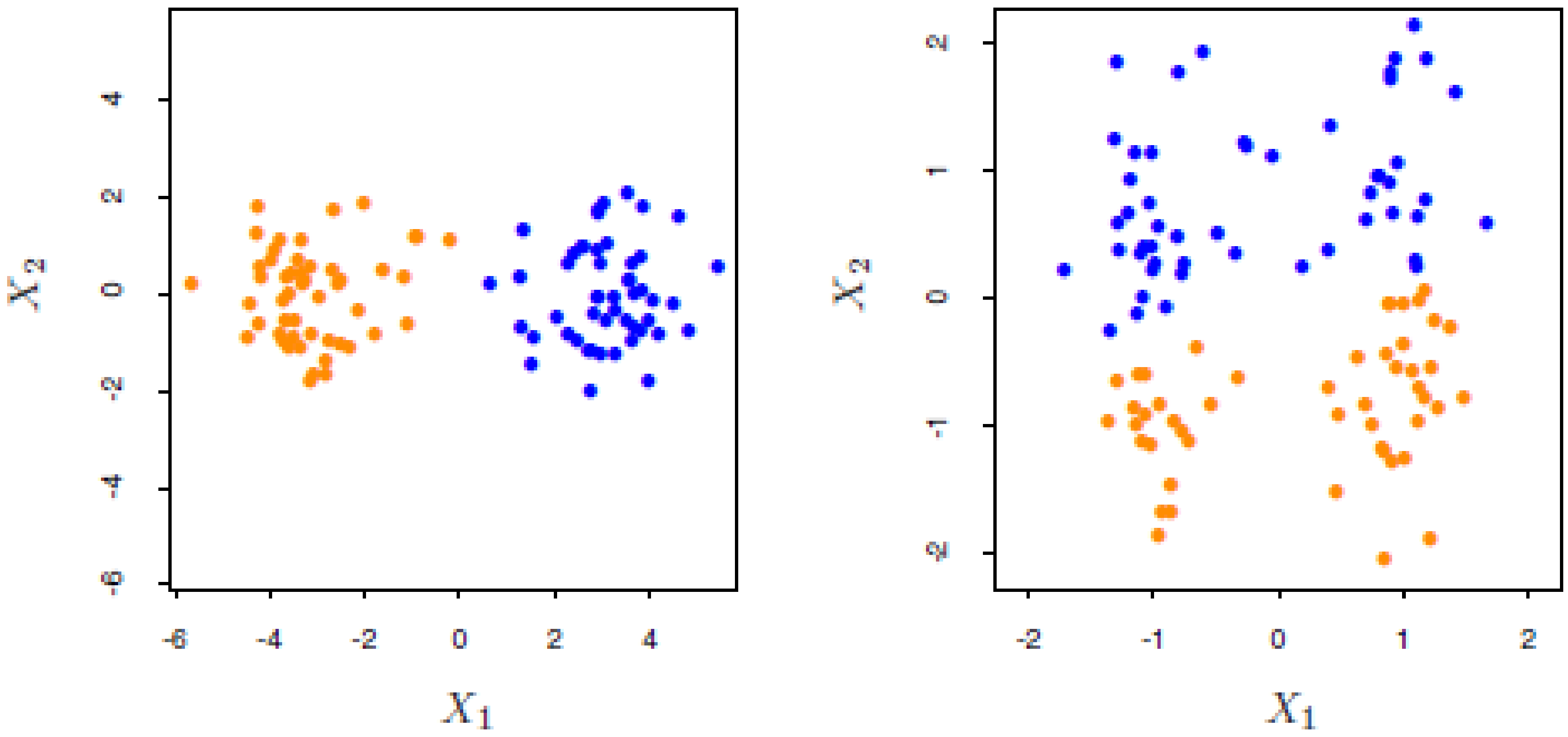
OTU	1	2	3	4	…	n
a	A	T	A	T	A	C
b	A	T	C	T	A	C
c	G	T	C	G	A	C
d	T	T	C	G	T	C

OTU	V1	V2	V3
a			
b			
c			
d			

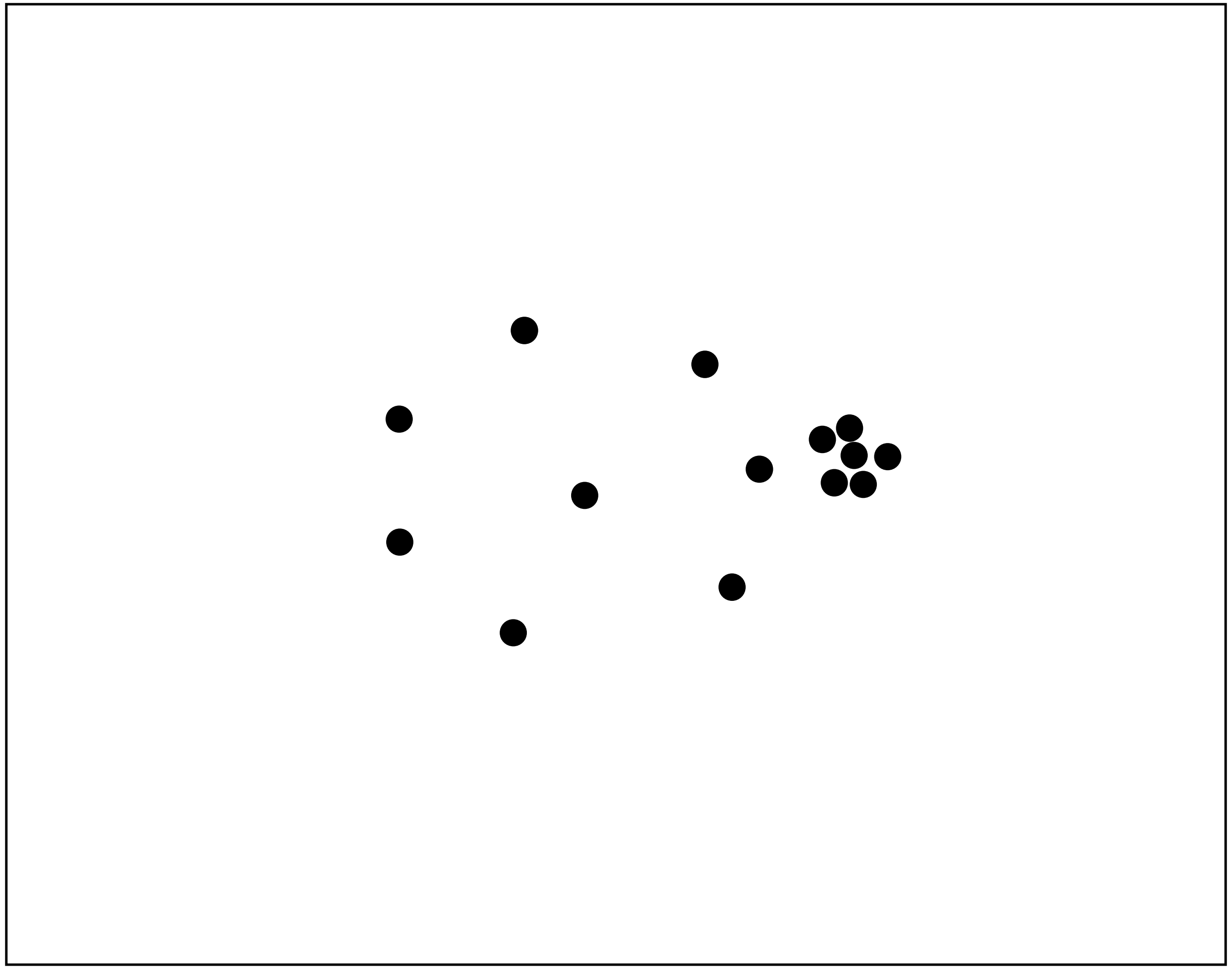
The hard way of getting a meaningful clustering…



The hard way of getting a meaningful clustering…



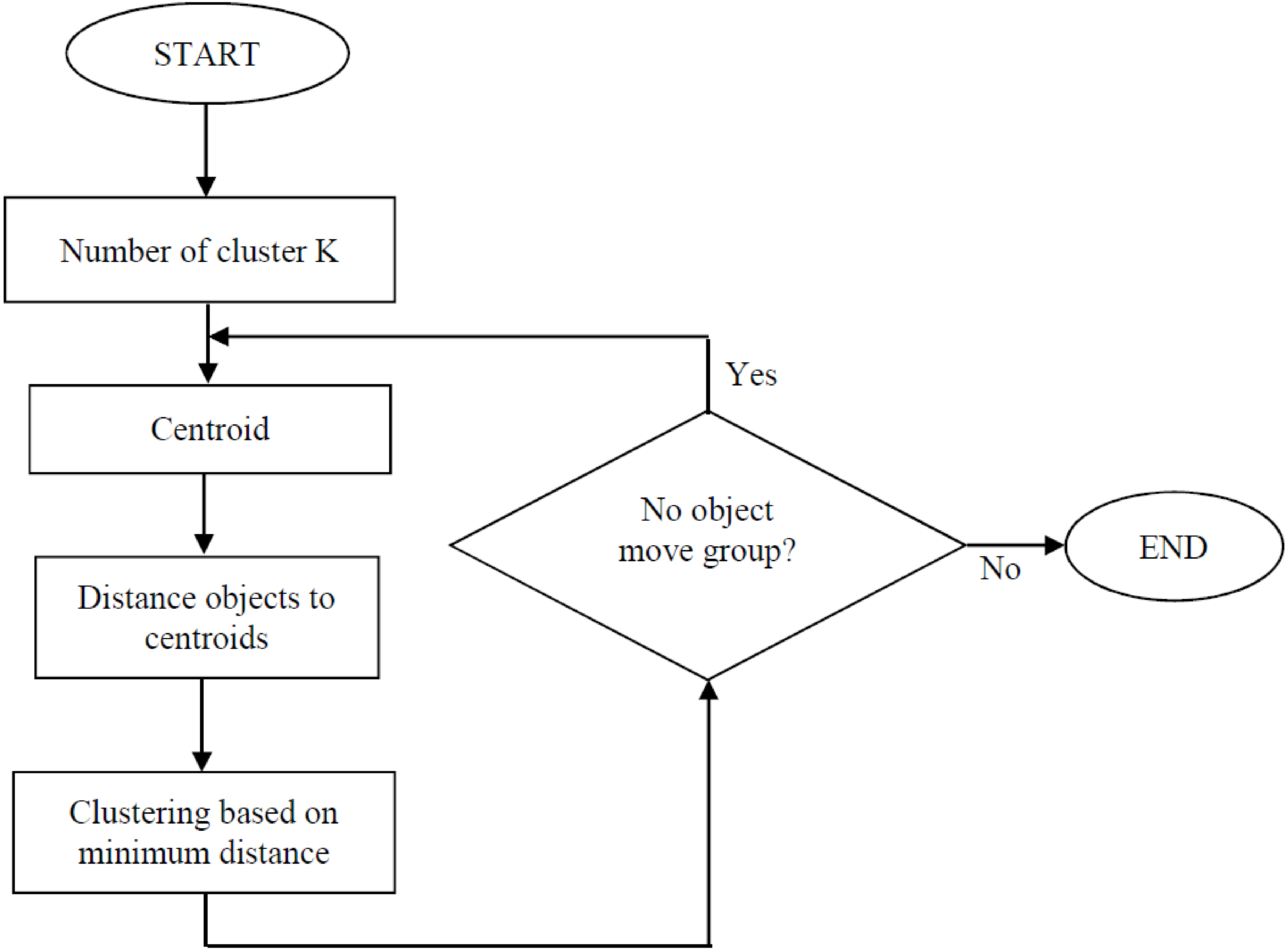
And here?



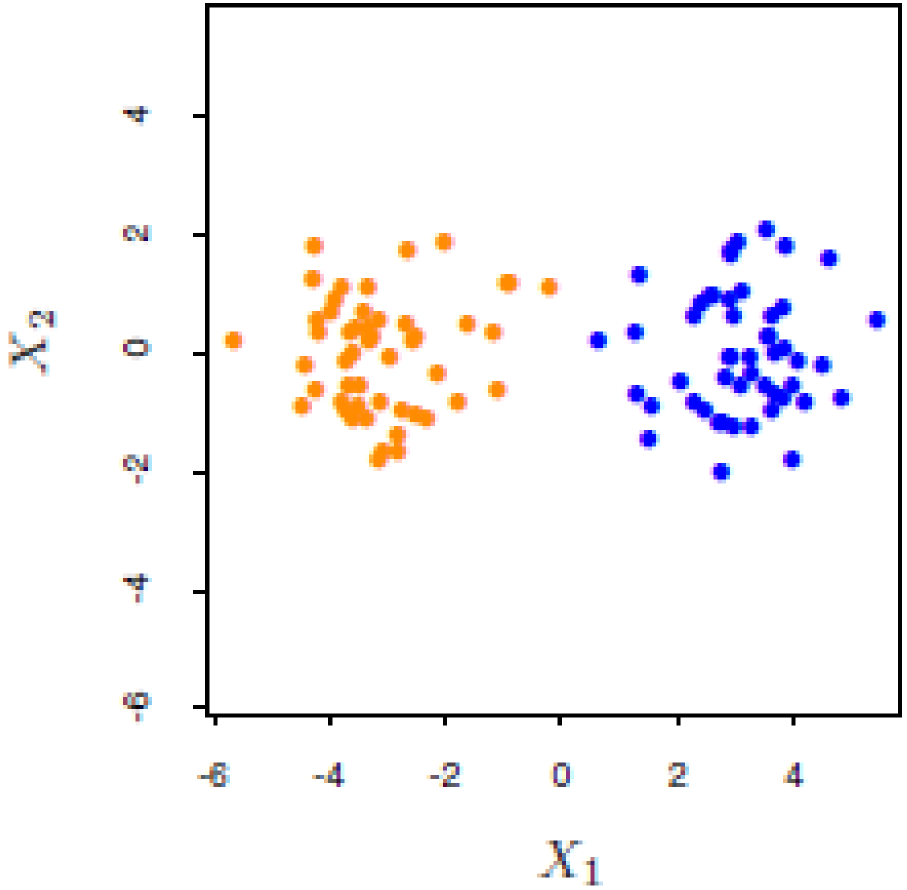
Which criteria should we use to cluster points?

- Partitional vs hierarchical
  - Partitional: define groups by iteratively optimizing an objective function. Need of number of prototype points to define what a cluster is. They are also called prototype-based clustering algorithms.
    - Generative: explain how the data is generated from a probabilistic point of view
    - Set of rules
  - Hierarchical: Iteratively clusters points creating a (binary) dendogram. Clusters are automatically defined by setting a threshold in the deepness of the tree.
    - Bottom-up
    - Top-down

K-means


START
Number of cluster K
Centroid
Distance objects to centroids
Clustering based on minimum distance
No object move group?
Yes
No
END

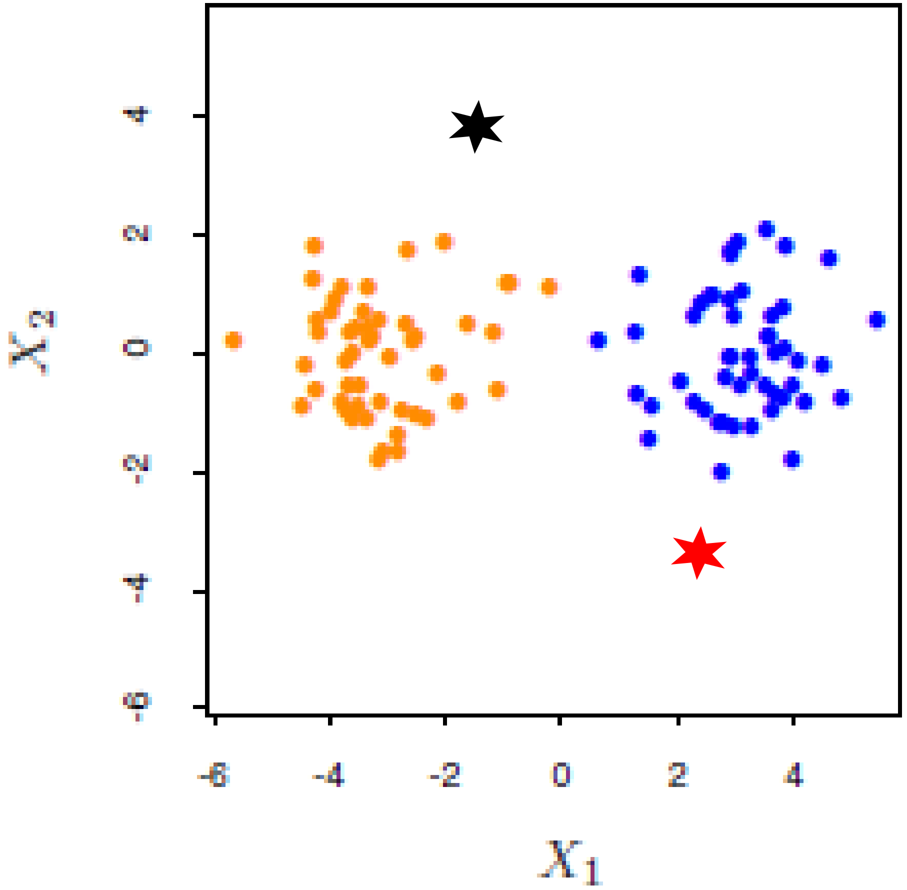
The K-means approach



$$d(x_i, x_{i'}) = \sum_{j=1}^{p}(x_{ij} - x_{i'j})^2 = ||x_i - x_{i'}||^2$$

The K-means approach

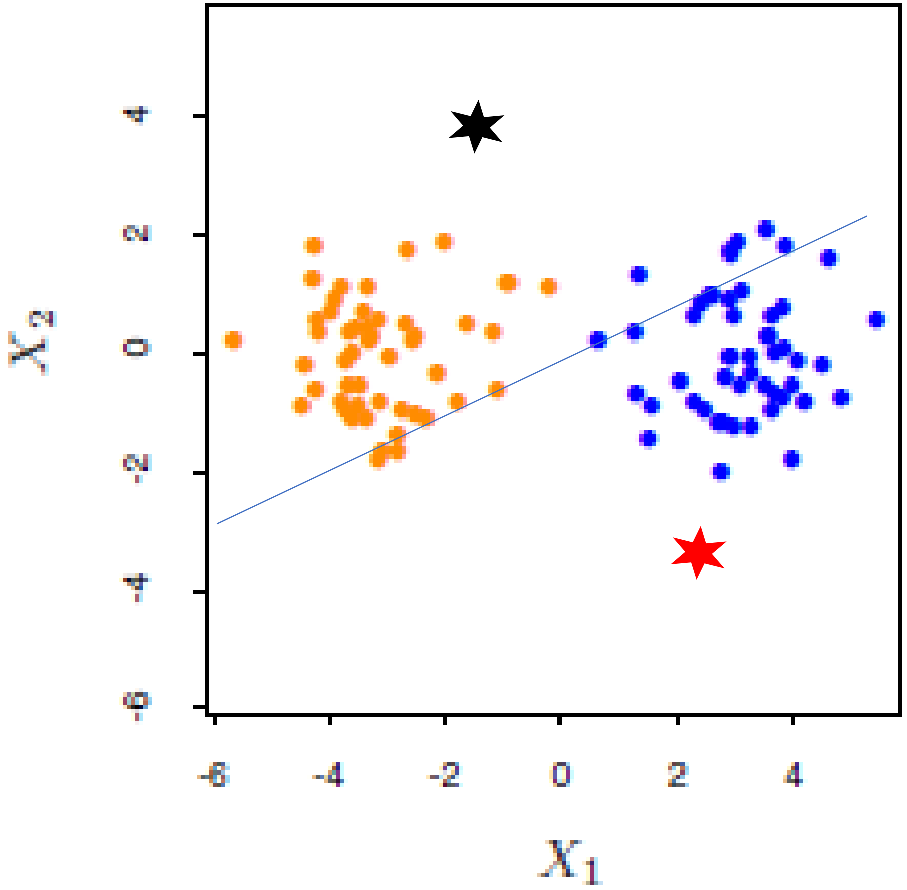
Set the coordinates at random



$$d(x_i, x_{i'}) = \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 = ||x_i - x_{i'}||^2$$

The K-means approach

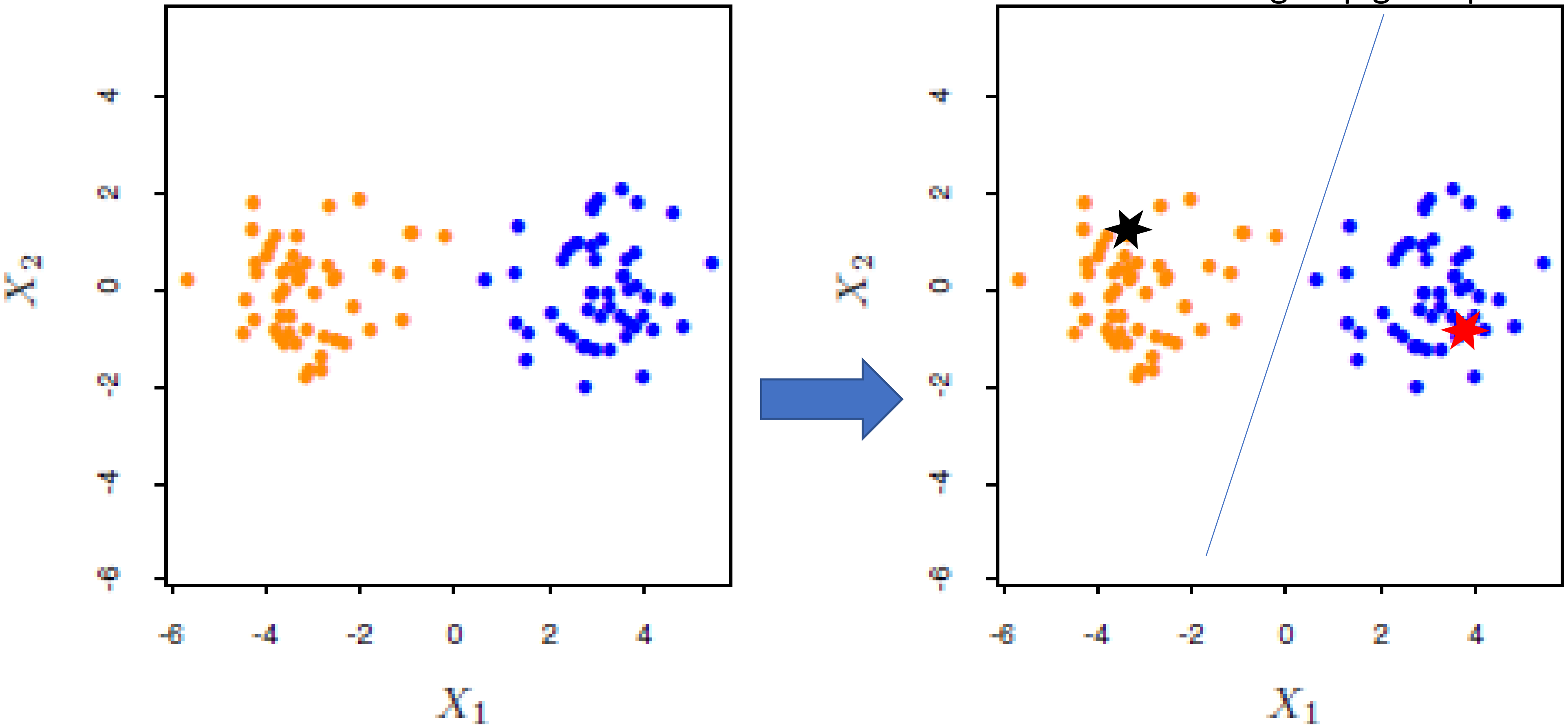
Define close neighbours



$$d(x_i, x_{i'}) = \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 = ||x_i - x_{i'}||^2$$

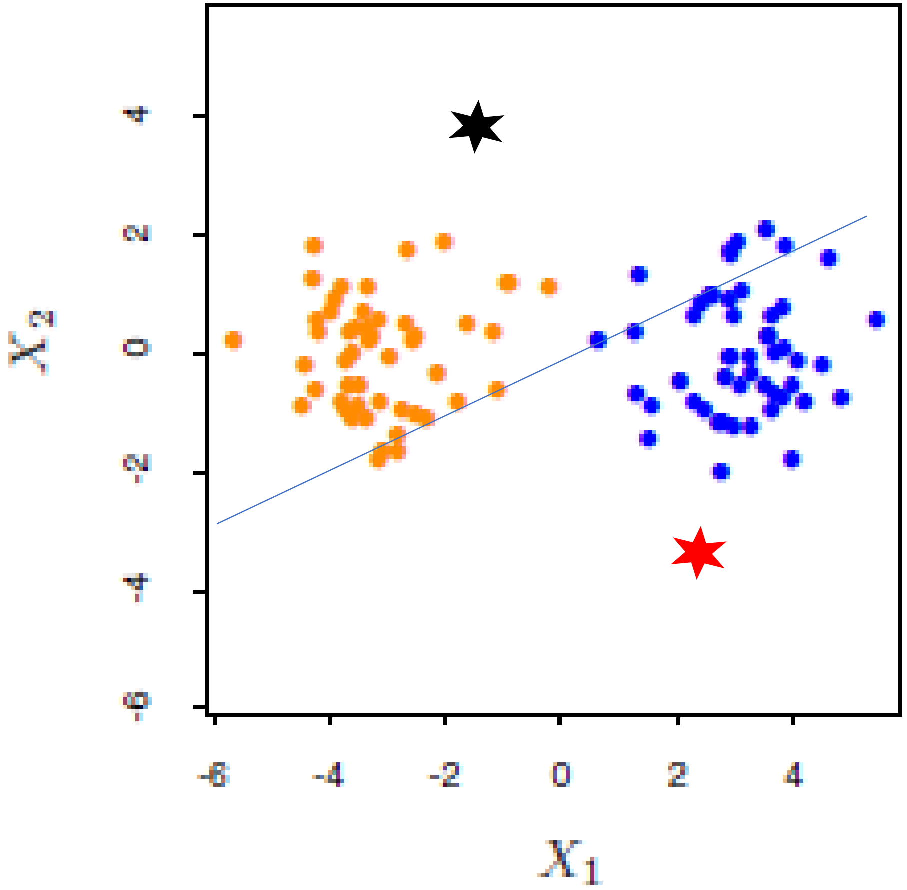
The K-means approach

Compute the new coordinates as the mean of the points
That fall in each group given previous coordinates

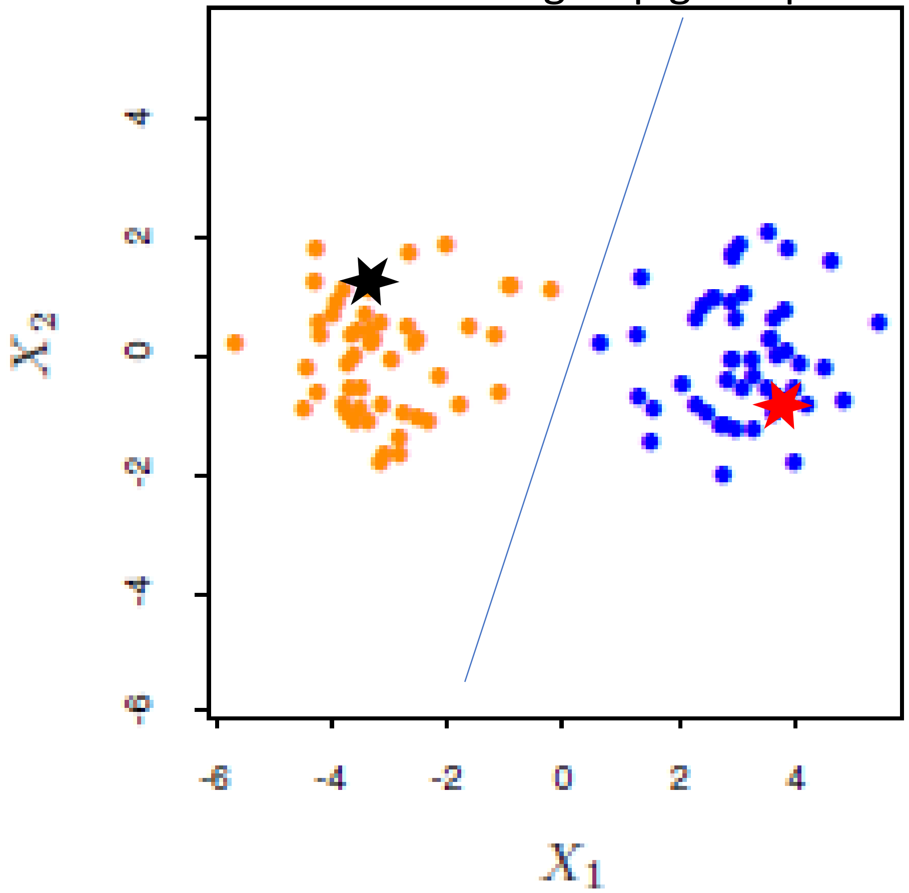


The K-means approach

Set the coordinates at random

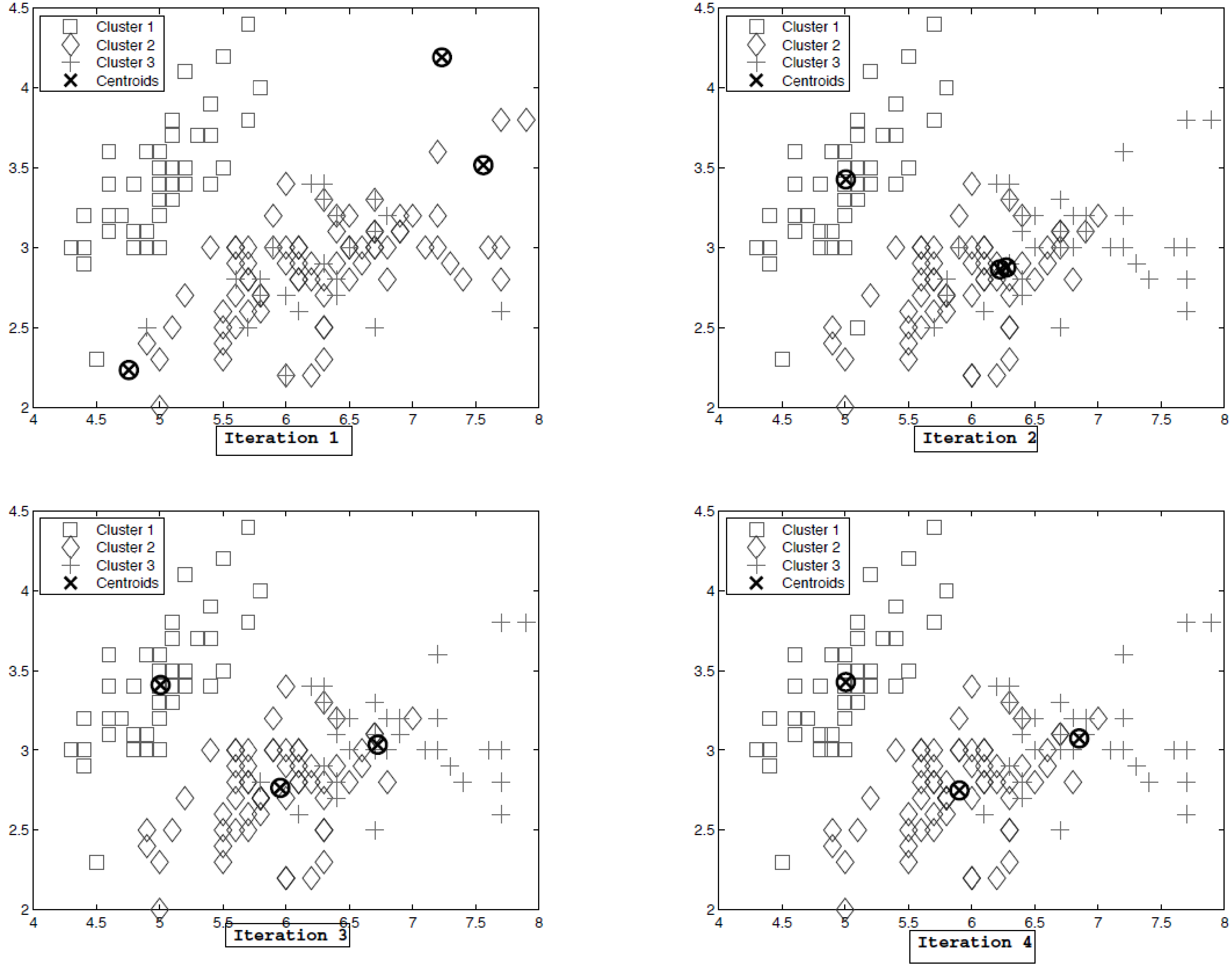


Compute the new coordinates as the mean of the points
That fall in each group given previous coordinates



$$d(x_i, x_{i'}) = \sum_{j=1}^{p}(x_{ij} - x_{i'j})^2 = ||x_i - x_{i'}||^2$$

K-means.



Cluster 1
Cluster 2
Cluster 3
Centroids
4.5
4
3.5
3
2.5
2
4
4.5
5
5.5
6
6.5
7
7.5
8
Iteration 1
Iteration 2
Iteration 3
Iteration 4

K-means. Pseudocode

The K-means approach

Objective function to minimize: Sum of Squared Errors (SSE)

For each cluster

For point that has been assigned to that cluster

$$
\begin{aligned}
SSE(C) &= \sum_{k=1}^{K} \sum_{x_i \in C_k} \| x_i - c_k \|^2 \\
c_k &= \frac{\sum\limits_{x_i \in C_k} x_i}{|C_k|}
\end{aligned}
$$

Centroid of cluster k

$$\|x_i - c_k\|^2 = \sum_{j=0}^{n} \left(x_i^j - c_k^j\right)^2$$

L2 norm is the Euclidean distance

The K-means approach

$$SSE(C) = \sum_{k=1}^{K} \sum_{x_i \in C_k} (c_k - x_i)^2$$

$$\begin{aligned}
\frac{\partial}{\partial c_j} SSE &= \frac{\partial}{\partial c_j} \sum_{k=1}^{K} \sum_{x_i \in C_k} (c_k - x_i)^2 \\
&= \sum_{k=1}^{K} \sum_{x_i \in C_j} \frac{\partial}{\partial c_j} (c_j - x_i)^2 \\
&= \sum_{x_i \in C_j} 2 * (c_j - x_i) = 0
\end{aligned}$$

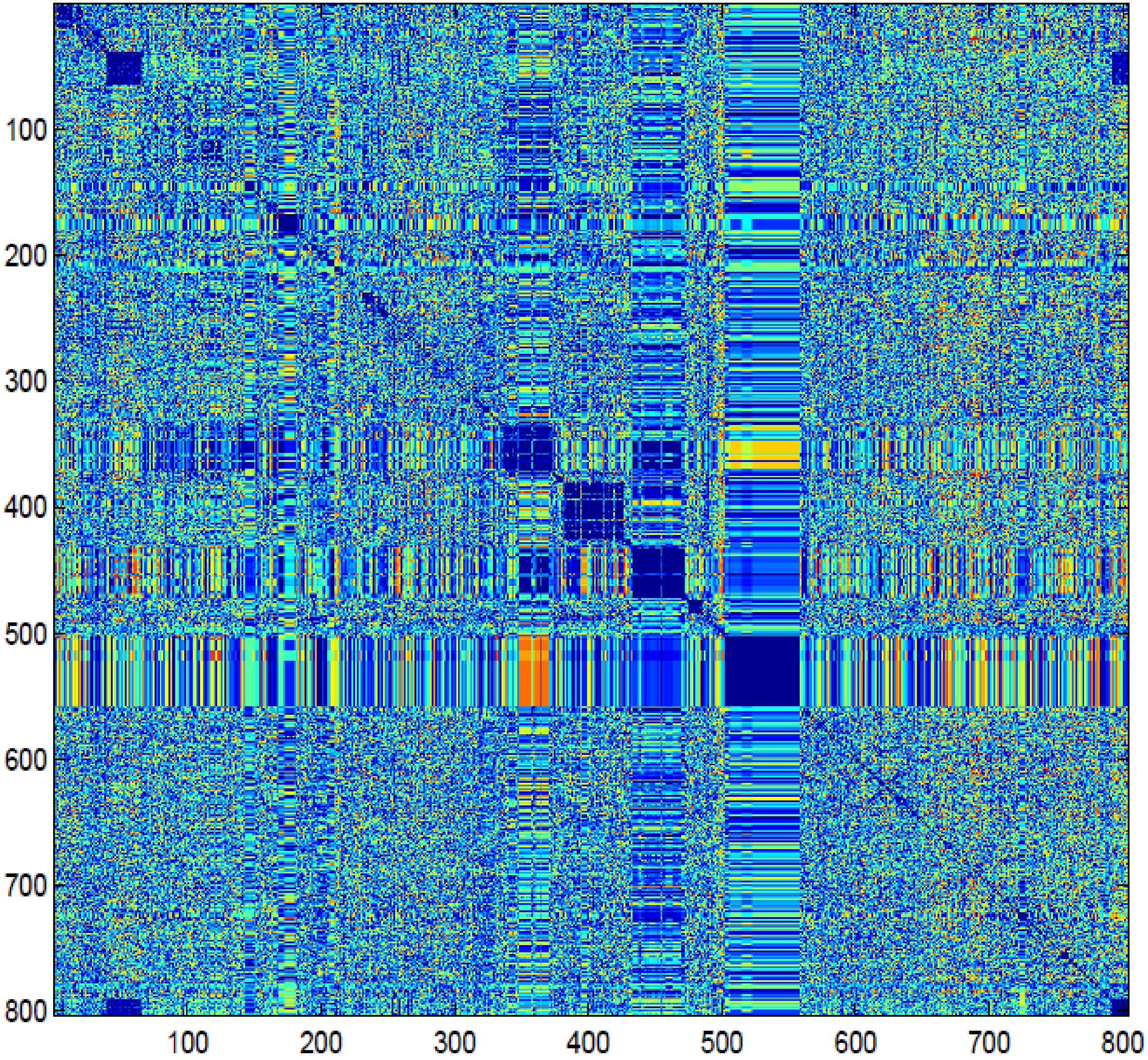
$$\sum_{x_i \in C_j} 2 * (c_j - x_i) = 0 \Rightarrow |C_j| \cdot c_j = \sum_{x_i \in C_j} x_i \Rightarrow c_j = \frac{\sum_{x_i \in C_j} x_i}{|C_j|}$$

This means that the centroid of the cluster allows minimizing the distance of assignation of each element that belongs to a cluster to its cluster

K-means. Pseudocode

- Clustering protein sequences

MATRIX OF SIMILARITY BETWEEN SEQUENCES

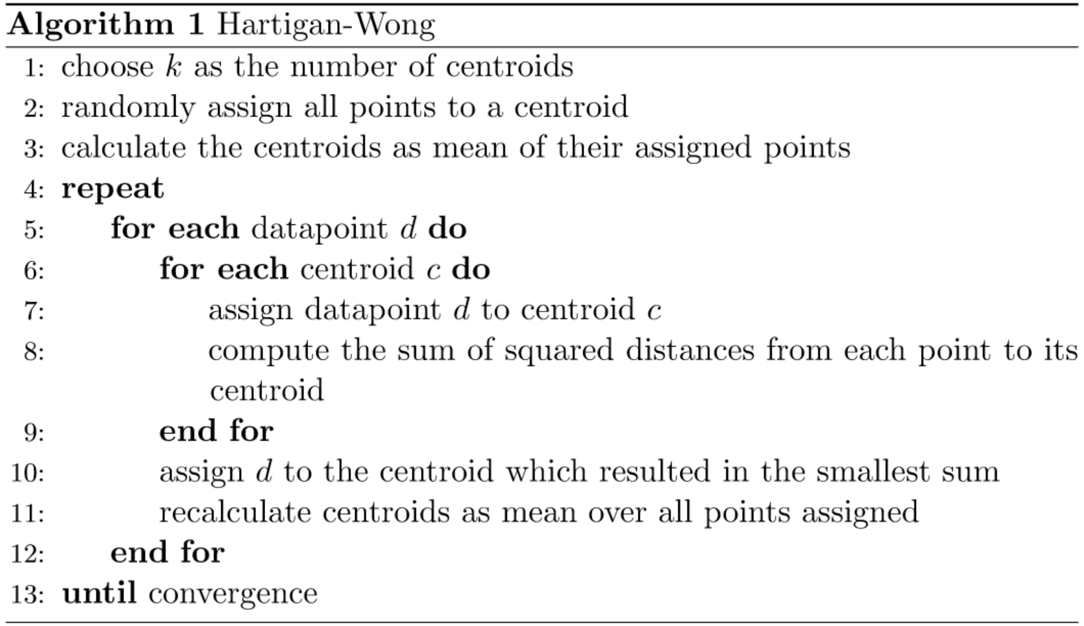


K-means.

- K-means is a greedy algorithm.
- Clusters depend not only on the nature of the data, but also on which **hyperparameters** you use.

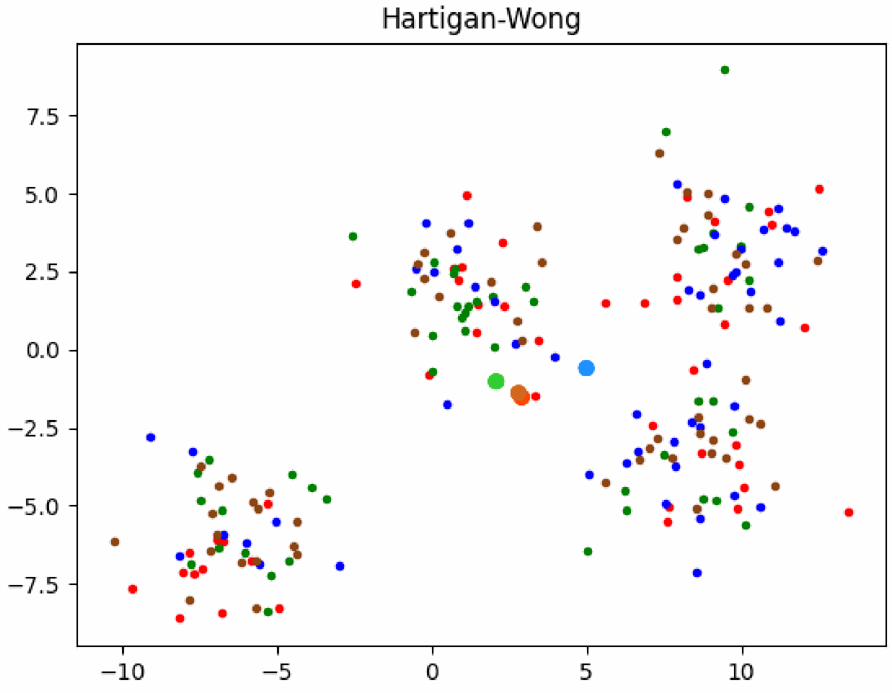
K-means.

- Inizialization points
  - **Hartigan and Wong**: points which are well separated and have a large number of points within their surrounding multi-dimensional sphere can be good candidates for initial points.



Algorithm 1 Hartigan-Wong

```
choose $k$ as the number of centroids
randomly assign all points to a centroid
calculate the centroids as mean of their assigned points
**repeat**
    **for each** datapoint $d$ **do**
        **for each** centroid $c$ **do**
            assign datapoint $d$ to centroid $c$
            compute the sum of squared distances from each point to its
            centroid
        **end for**
        assign $d$ to the centroid which resulted in the smallest sum
        recalculate centroids as mean over all points assigned
    **end for**
**until** convergence
```



K-means.

- Inizialization points
  - **Milligan**: Apply first agglomerative hierarchical clustering
  - **K-Means++**: First centroid is selected at random. The next centroid selected is the one which is farthest from the currently selected centroid. This selection is decided based on a weighted probability score. The selection is continued until we have K centroids and then K-means clustering is done using these centroids.
  - ***Put yours here***

The k-means approach

When to stop

- As the number of clusters K is changed, the cluster memberships can change in arbitrary ways (things that before were not in the same cluster at k-1, can be in the same cluster at k!)
- Which is the optimal number of clusters????

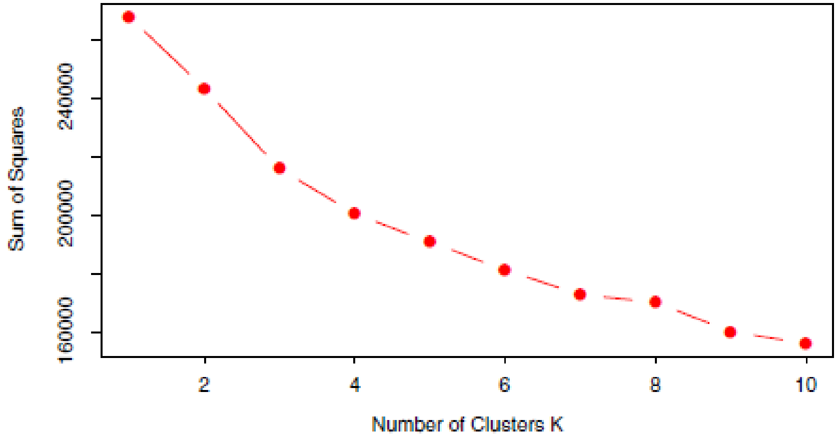


FIGURE 14.8. *Total within-cluster sum of squares for K-means clustering applied to the human tumor microarray data.*

The k-means approach

- **Calinski–Harabasz Index**:

$$CH(K) = \frac{\frac{B(K)}{(K-1)}}{\frac{W(K)}{N-K}}$$

Distance of points Between clusters

Distance of points within clusters

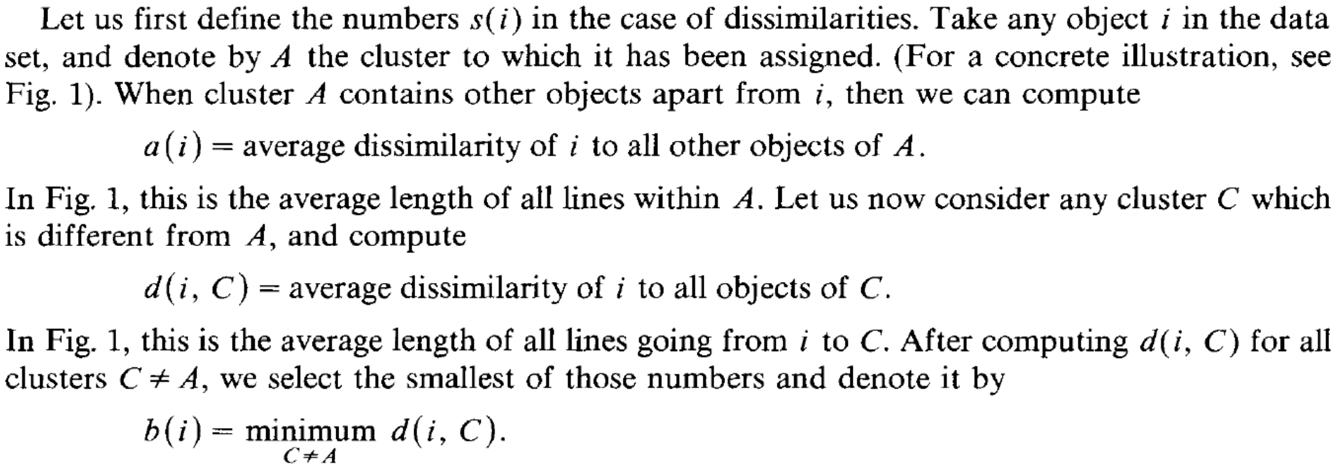
Do we want to maximize or
minimize this function?

Number of points

Number of clusters

The k-means approach

- A silhouette approach



Let us first define the numbers $s(i)$ in the case of dissimilarities. Take any object i in the data set, and denote by A the cluster to which it has been assigned. (For a concrete illustration, see Fig. 1). When cluster A contains other objects apart from i, then we can compute

$$a(i) = \text{average dissimilarity of } i \text{ to all other objects of } A.$$

In Fig. 1, this is the average length of all lines within A. Let us now consider any cluster C which is different from A, and compute

$$d(i, C) = \text{average dissimilarity of } i \text{ to all objects of } C.$$

In Fig. 1, this is the average length of all lines going from i to C. After computing $d(i, C)$ for all clusters $C \neq A$, we select the smallest of those numbers and denote it by

$$b(i) = \underset{C \neq A}{\text{minimum}}\ d(i, C).$$

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i),\ b(i)\}}\ .$$

$$-1 \leqslant s(i) \leqslant 1$$

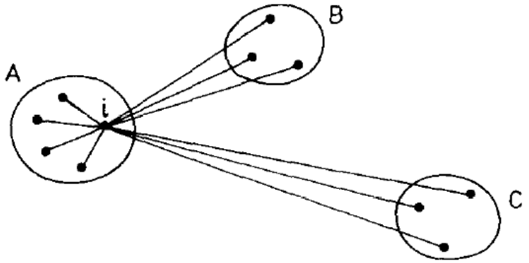


Fig. 1. An illustration of the elements involved in the computation of $s(i)$, where the object i belongs to cluster A.

The k-means approach

- Gap Statistic: Generate B different datasets. Compute within cluster sum of squares (W_b^*).

$$Gap(K) = \frac{1}{B} \times \sum_b log(W_b^*(K)) - log(W(K))$$

Pick the number of clusters K so:

$$Gap(K) \geq Gap(K+1) - s_{k+1}$$

where S_{k+1} represents the estimate of standard deviation of $log(W_b^*(K+1))$.

- Many others!

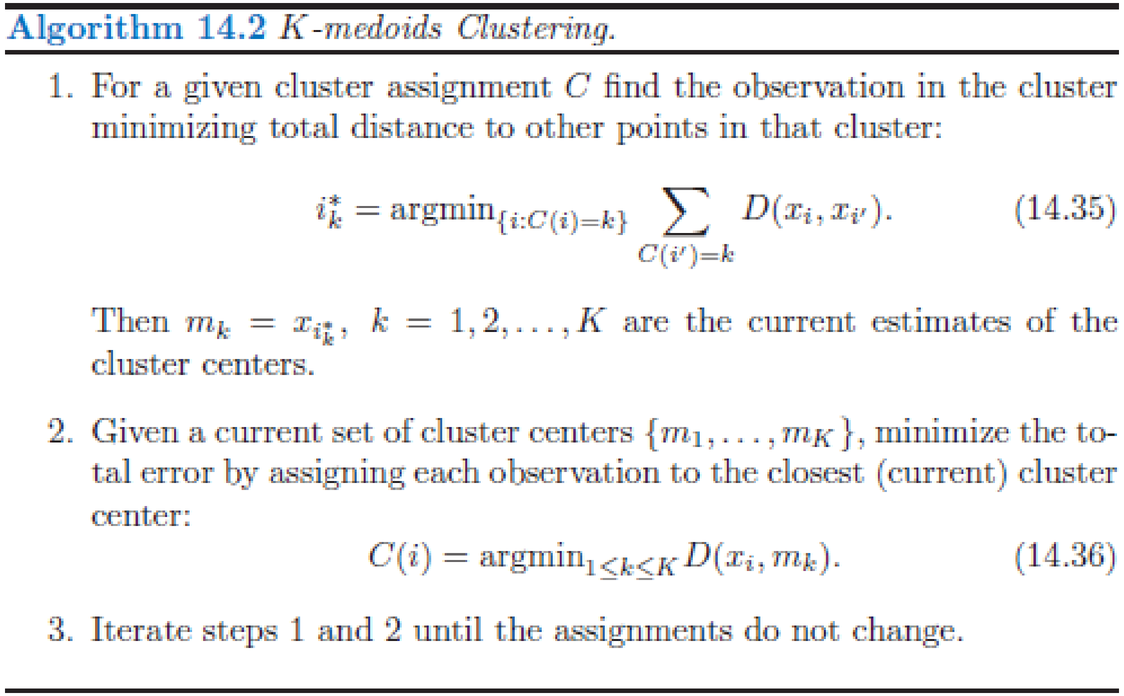
Variations of K means: K-medoids

- K-means algorithm is appropriate when the dissimilarity measure is taken to be squared Euclidean distance D(xi, xi0).
    - All of the variables MUST be of the quantitative type.
    - Squared Euclidean distance places the highest influence on the largest distances.
    - Lack robustness against outliers that produce very large distances.

K-medoids

- The algorithm can be generalized for use with arbitrarily defined dissimilarities D(xi, xi0) by replacing this step by an explicit optimization
- Centers for each cluster are restricted to be one of the observations assigned to the cluster

K-medoids. Pseudocode



Algorithm 14.2 *K-medoids Clustering.*

1. For a given cluster assignment C find the observation in the cluster minimizing total distance to other points in that cluster:

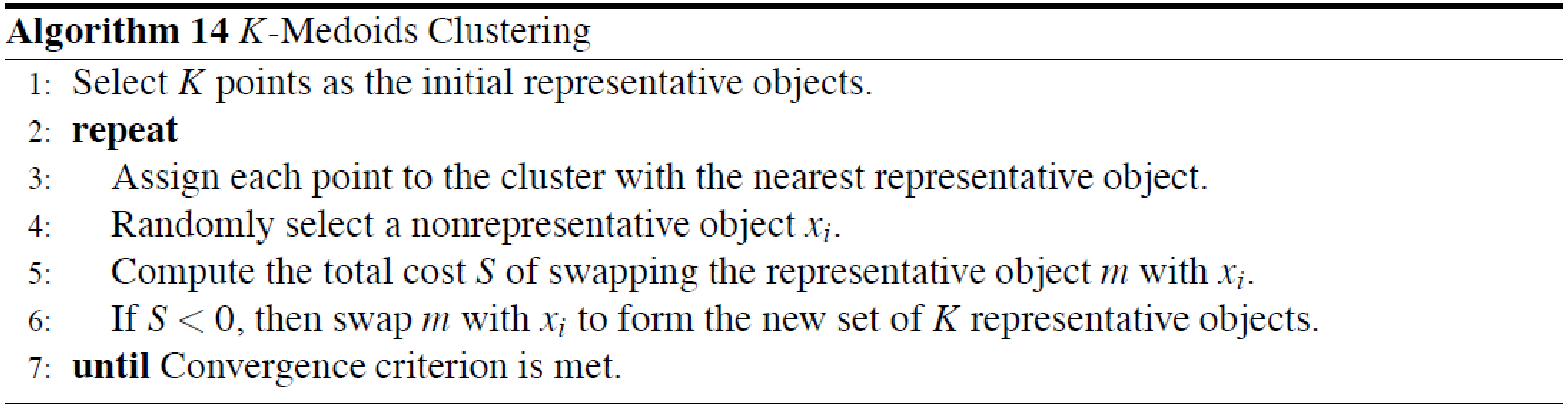
$$i_k^* = \text{argmin}_{\{i:C(i)=k\}} \sum_{C(i')=k} D(x_i, x_{i'}). \qquad (14.35)$$

   Then $m_k = x_{i_k^*}$, $k = 1, 2, \ldots, K$ are the current estimates of the cluster centers.

2. Given a current set of cluster centers $\{m_1, \ldots, m_K\}$, minimize the total error by assigning each observation to the closest (current) cluster center:

$$C(i) = \text{argmin}_{1 \leq k \leq K} D(x_i, m_k). \qquad (14.36)$$

3. Iterate steps 1 and 2 until the assignments do not change.



Algorithm 14 *K*-Medoids Clustering

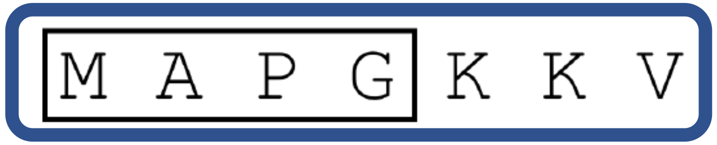
```
Select $K$ points as the initial representative objects.
repeat
    Assign each point to the cluster with the nearest representative object.
    Randomly select a nonrepresentative object $x_i$.
    Compute the total cost $S$ of swapping the representative object $m$ with $x_i$.
    If $S < 0$, then swap $m$ with $x_i$ to form the new set of $K$ representative objects.
until Convergence criterion is met.
```

Other applications based on k-means approach

- Identify the function of a protein based on its sequence.
    - Similarity of protein domains should be associated to function.
    - What is similarity?
- How to do it?

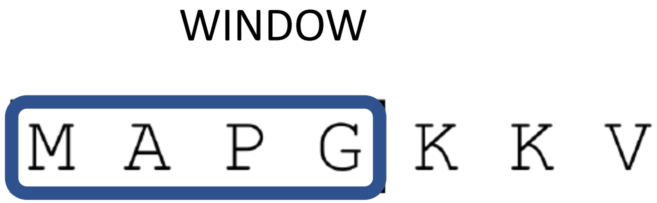
Other applications based on k-means approach

PROTEIN SEQUENCE



M A P G K K V

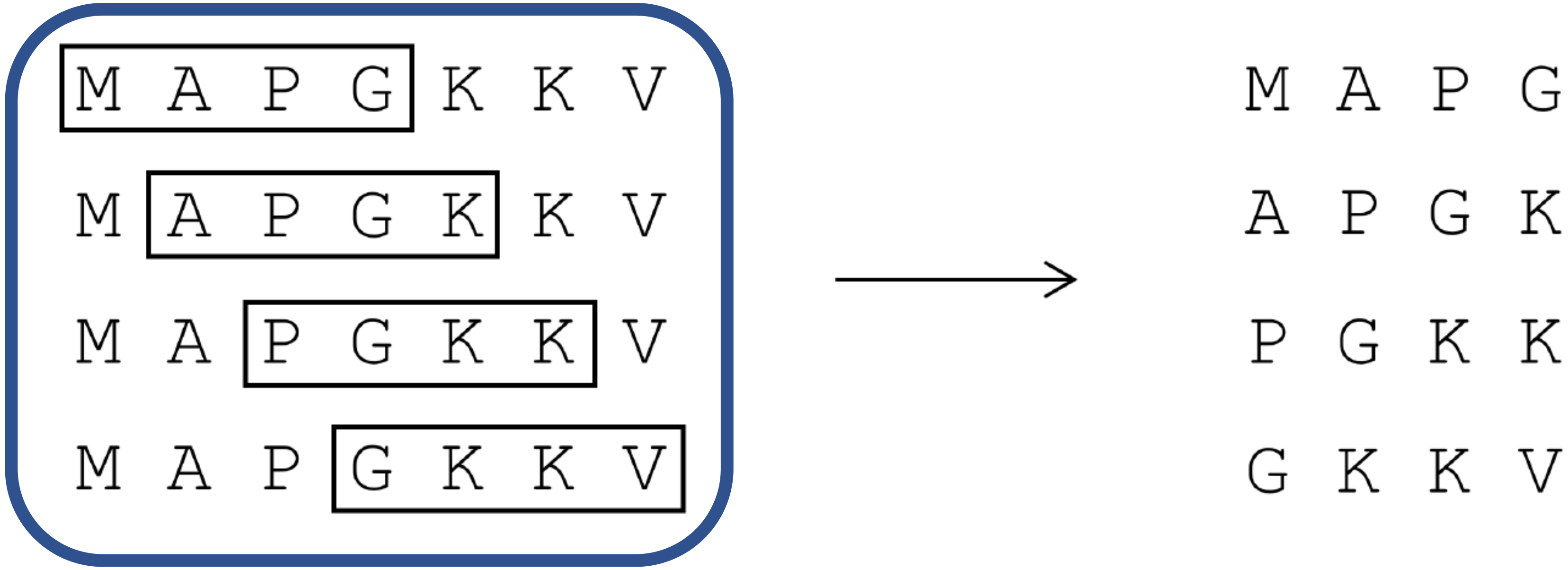
Other applications based on k-means approach



WINDOW

M A P G K K V

Other applications based on k-means approach



MOVE THE WINDOW

M A P G K K V

M A P G K K V

M A P G K K V

M A P G K K V

M A P G

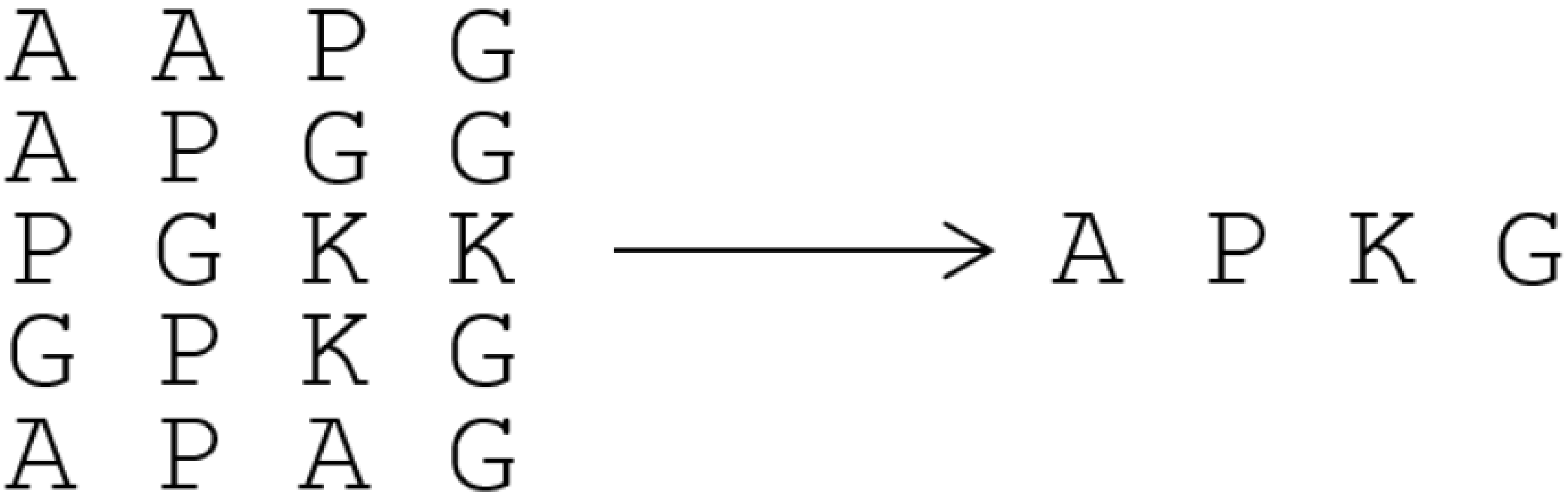
A P G K

P G K K

G K K V

Other applications based on k-means approach

APPLY K-MEANS TO IDENTIFY CENTROID WINDOWS

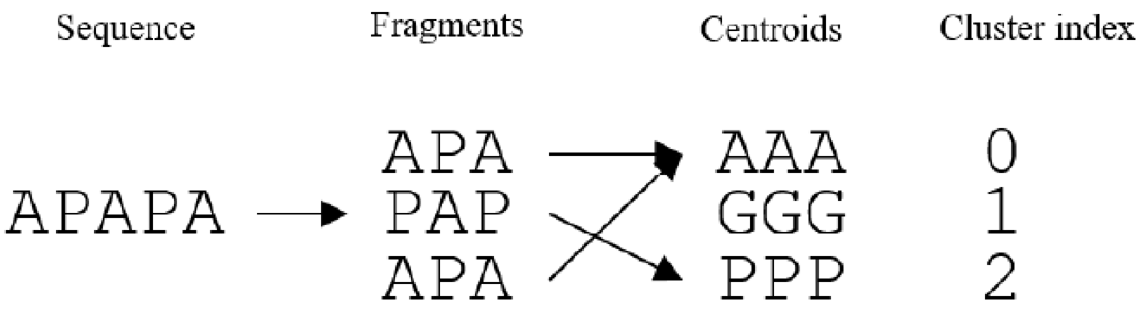


A A P G
A P G G
P G K K
G P K G
A P A G

A P K G

Other applications based on k-means approach

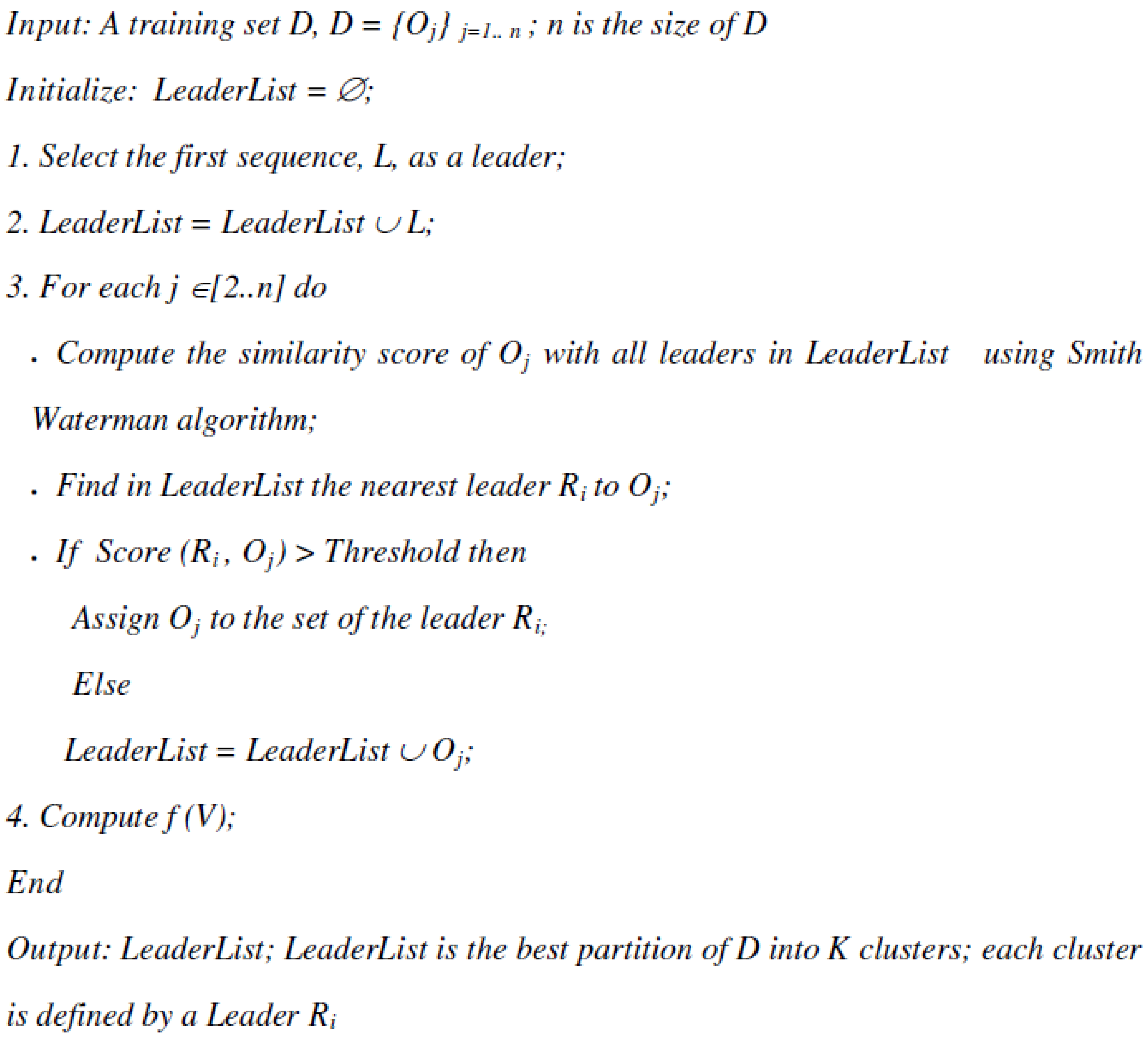
ENCODE THE PROTEIN SEQUENCE INTO A SEQUENCE OF CLUSTER INDEXES



Final vector 2 0 1

This vector can then be used to perform a clustering, train a model on functional prediction, etc

LEADER algorithm



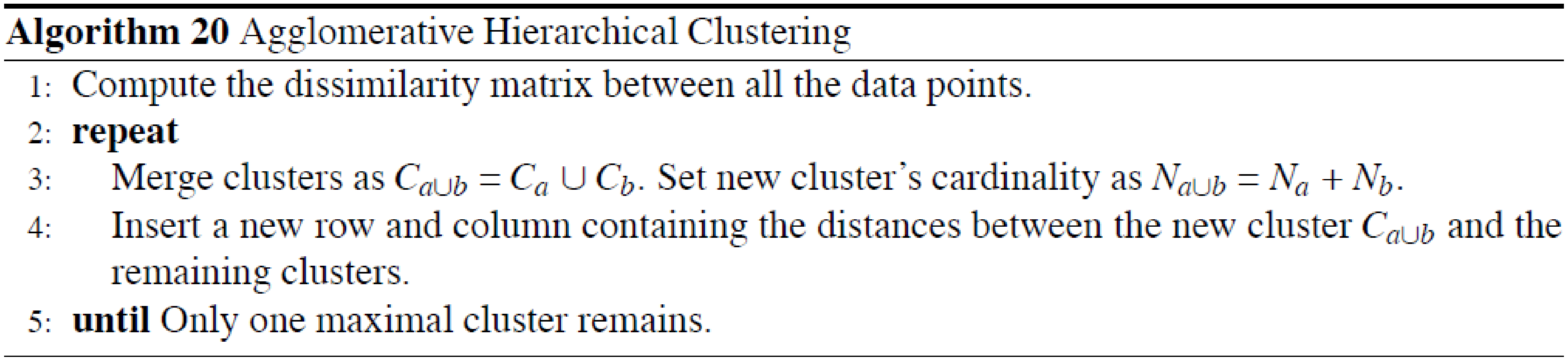
```
Input: A training set D, $D = \{O_j\}_{j=1..n}$ ; n is the size of D
Initialize:  LeaderList = $\varnothing$;
1. Select the first sequence, L, as a leader;
2. LeaderList = LeaderList $\cup$ L;
3. For each $j \in [2..n]$ do
   . Compute the similarity score of $O_j$ with all leaders in LeaderList   using Smith
   Waterman algorithm;
   . Find in LeaderList the nearest leader $R_i$ to $O_j$;
   . If  Score $(R_i , O_j)$ > Threshold then
       Assign $O_j$ to the set of the leader $R_{i;}$
       Else
      LeaderList = LeaderList $\cup$ $O_j$;
4. Compute f (V);
End
Output: LeaderList; LeaderList is the best partition of D into K clusters; each cluster
is defined by a Leader $R_i$
```

Hierarchical clustering

- Agglomerative (bottom-up) and divisive (top-down).
    - With both paradigms there are N − 1 levels in the hierarchy.
    - All agglomerative and some divisive methods (when viewed bottom-up) possess a monotonicity property. That is, the dissimilarity between merged clusters is monotone increasing with the level of the merger. Thus the binary tree can be plotted so that the height of each node is proportional to the value of the intergroup dissimilarity between its two daughters. The terminal nodes representing individual observations are all plotted at zero height.

Hierarchical clustering

- Agglomerative (bottom-up).



Algorithm 20 Agglomerative Hierarchical Clustering

```
Compute the dissimilarity matrix between all the data points.
repeat
    Merge clusters as $C_{a \cup b} = C_a \cup C_b$. Set new cluster's cardinality as $N_{a \cup b} = N_a + N_b$.
    Insert a new row and column containing the distances between the new cluster $C_{a \cup b}$ and the remaining clusters.
until Only one maximal cluster remains.
```

Hierarchical clustering

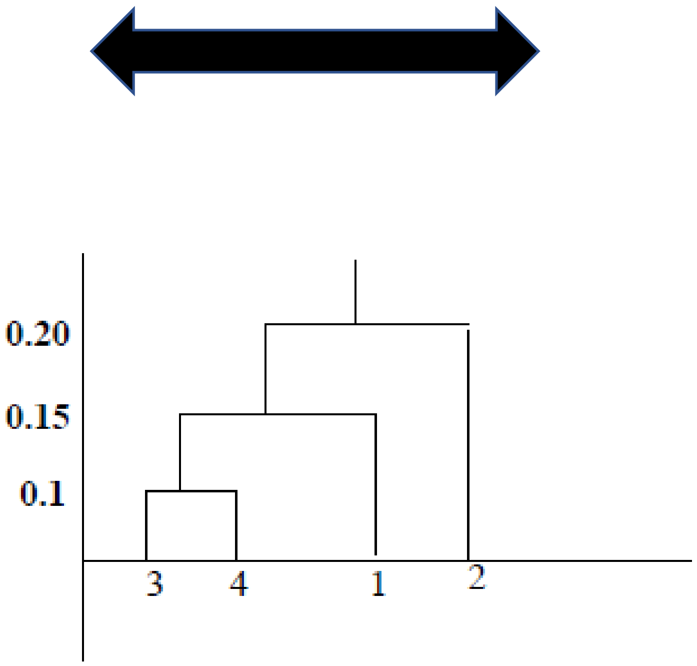
- Agglomerative (bottom-up).
  - Single link clustering. The similarity of two clusters is the similarity between their most similar (nearest neighbor) members. Local similarity-based clustering (allows modelling non-elliptical clusters). Sensitive to OUTLIERS.
  - Complete link clustering. Measures the similarity of two clusters as the similarity of their most dissimilar members. Generally obtains compact shaped clusters. Sensitive to OUTLIERS.

Hierarchical clustering

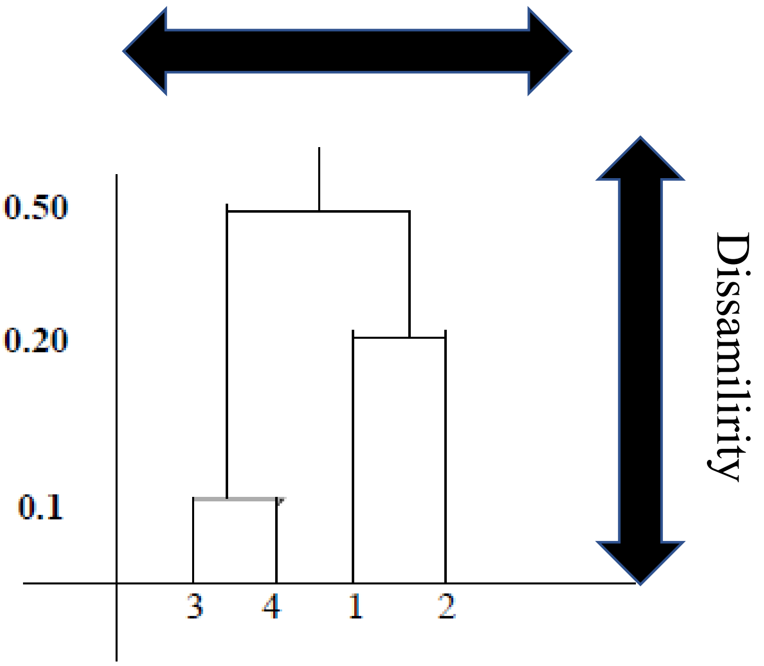
Data objects

$$\begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0.0 & 0.20 & 0.15 & 0.30 \\ 2 & 0.20 & 0.0 & 0.40 & 0.50 \\ 3 & 0.15 & 0.40 & 0.0 & 0.10 \\ 4 & 0.30 & 0.50 & 0.10 & 0.0 \end{array}$$

(a) Dissimilarity Matrix



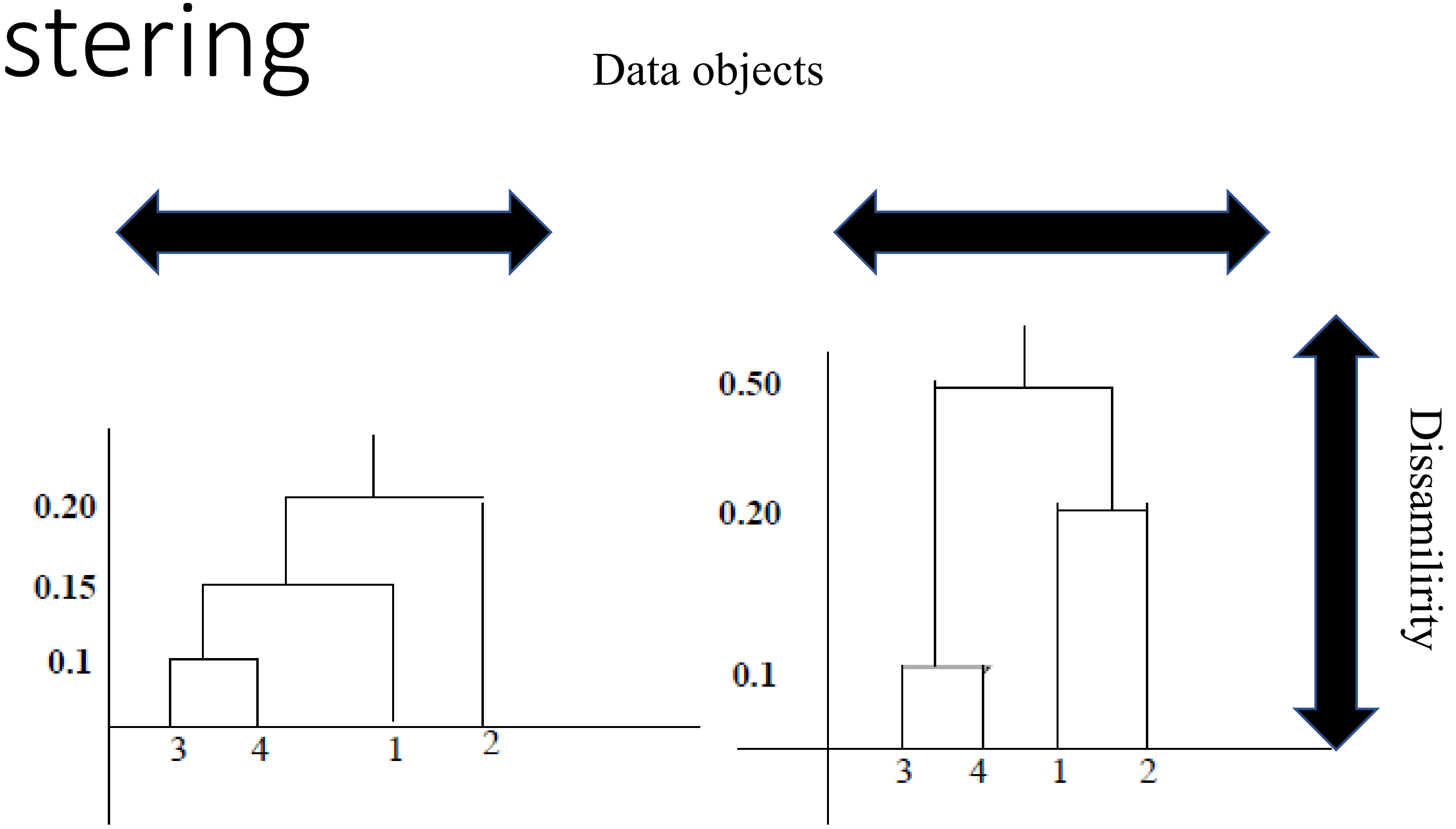
(b) Single Link



(c) Complete Link

Hierarchical clustering

$$\begin{aligned}
d_{min}((3,4),1) &= min(d(3,1),d(4,1)) = 0.15 \\
d_{min}((3,4,1),2) &= min(d(3,2),d(4,2),d(1,2)) = 0.20 \\
d_{max}((3,4),1) &= max(d(3,1),d(4,1)) = 0.30 \\
d_{max}((3,4),2) &= max(d(3,2),d(4,2)) = 0.50 \\
d_{max}((3,4),(1,2)) &= max(d(3,1),d(3,2),d(4,1),d(4,2)) = 0.50
\end{aligned}$$



(b) Single Link

(c) Complete Link

Hierarchical clustering

- Group Averaged and Centroid Agglomerative Clustering
  - Similarity is computed as the average distance between the elements of the clusters

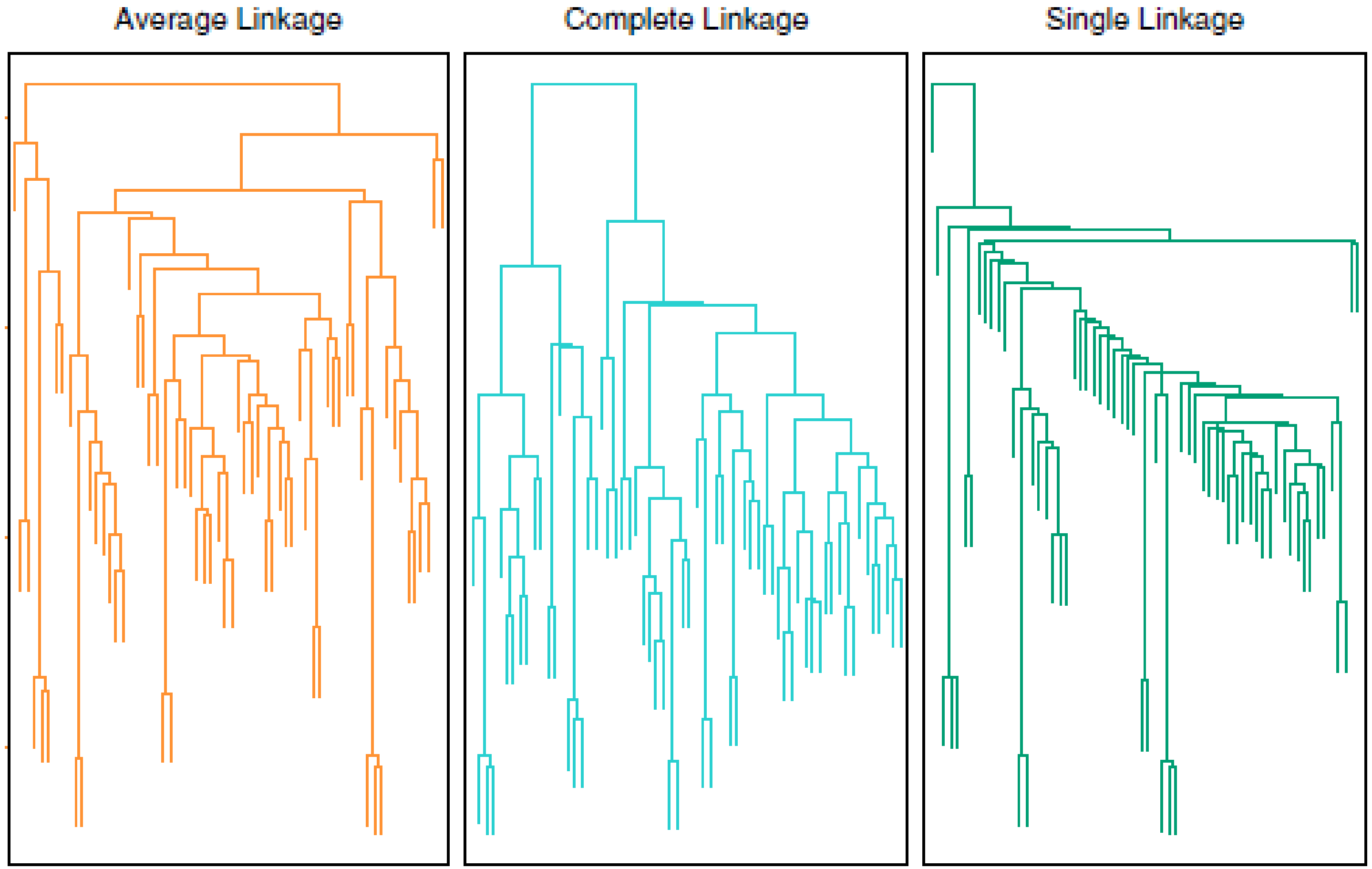
Similarity between cluster a and b

$$S_{GAAC}(C_a, C_b) = \frac{1}{(N_a + N_b)(N_a + N_b - 1)} \sum_{i \in C_a \cup C_b} \sum_{j \in C_a \cup C_b, i \neq j} d(i, j)$$

Cluster a

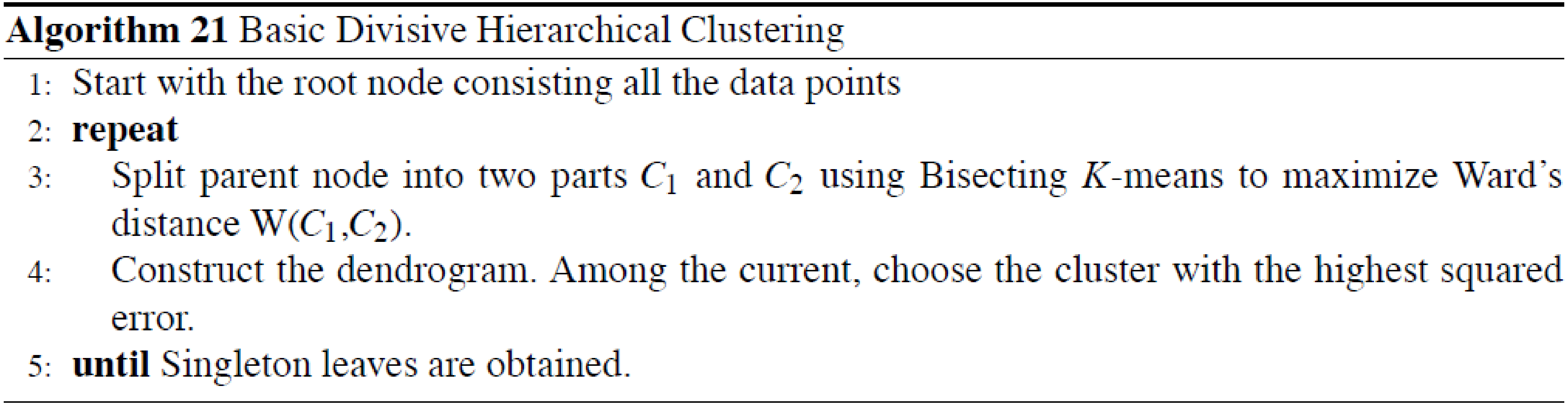
Cluster b

Agglomerative clustering



Divisive clustering

- begin with the entire data set as a single cluster, and recursively divide one of the existing clusters into two daughter clusters at each iteration in a top-down fashion.



Algorithm 21 Basic Divisive Hierarchical Clustering

```
Start with the root node consisting all the data points
repeat
    Split parent node into two parts $C_1$ and $C_2$ using Bisecting $K$-means to maximize Ward's distance W($C_1$,$C_2$).
    Construct the dendrogram. Among the current, choose the cluster with the highest squared error.
until Singleton leaves are obtained.
```

Divisive clustering

Centroid of cluster a for variable v

- *Splitting criterion*: The Ward's *K*-means square error:

$$
\begin{aligned}
W(C_{a\cup b}, c_{a\cup b}) - W(C, c) &= \frac{N_a N_b}{N_a + N_b} \sum_{v=1}^{M} (c_{av} - c_{bv})^2 \\
&= \frac{N_a N_b}{N_a + N_b} d(c_a, c_b)
\end{aligned}
$$

The greater reduction obtained in the difference in the SSE criterion should reflect the goodness of the split. Gini index (which we will see in next sessions) can be used for handling the nominal data.

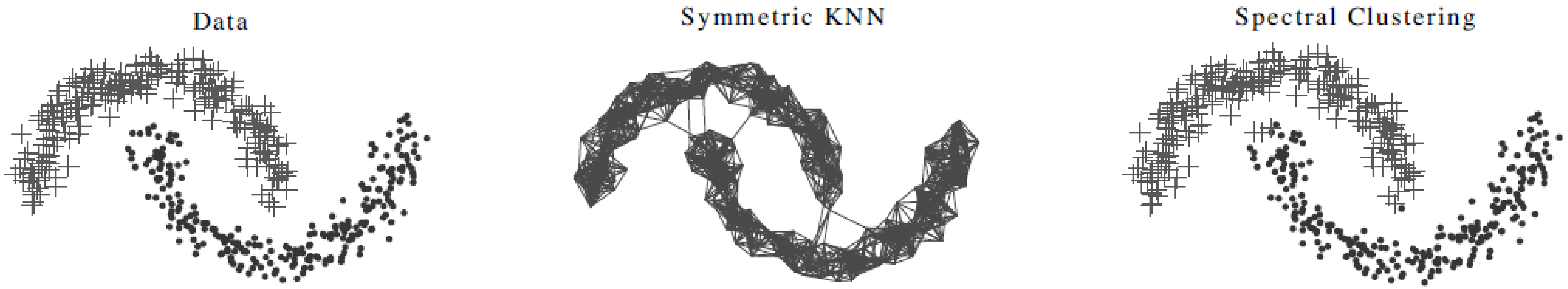
- *Splitting method*: The splitting method used to obtain the binary split of the parent node is also critical since it can reduce the time taken for evaluating the Ward's criterion. The Bisecting *K*-means approach can be used (with *K* = 2) to obtain good splits.
- *Handling noise*: Noise points can produce meaningless clusters. A threshold can be used to determine termination.

Hierarchical clustering (Other methods)

- CURE (Clustering Using REpresentatives): Each cluster is represented by a set of representative points. The distance between two clusters is calculated by looking at the minimum distance between the representative points chosen.
- CHAMELEON is a clustering algorithm which uses graph partitioning methods on the K-nearest neighbor graph of the data. These initial partitions are then used as the seed clusters for the agglomerative hierarchical clustering process.
- Self-Organizing Maps (SOM). Come from the field of Artificial Neural Networks. Data points are assigned to their closest centroids using a non-linear function to define the similarity of points.

Other clustering methods

- Spectral decomposition: Create a graph similarity of points, eigentransformation of the weights of the similarity, detection of clusters



Other clustering methods

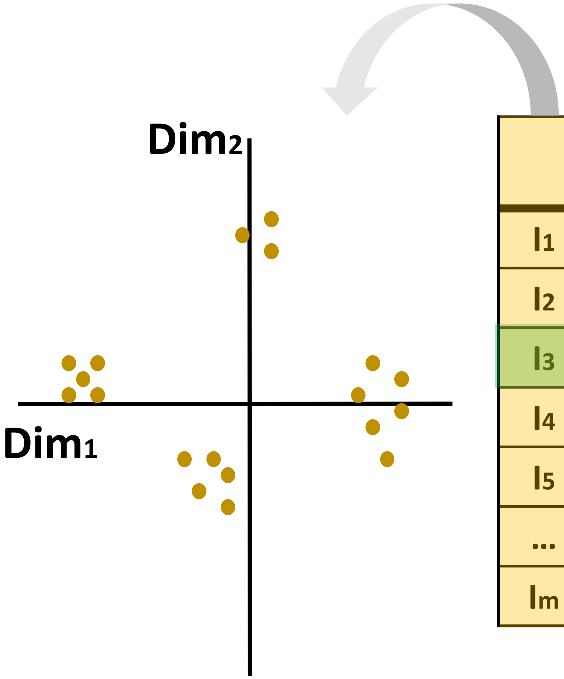
- MANY MANY MANY OTHERS…

Other ways of reducing the dimensionality

- How can we interpret a clustering from an machine learning information point of view?

TIME FOR A BREAK

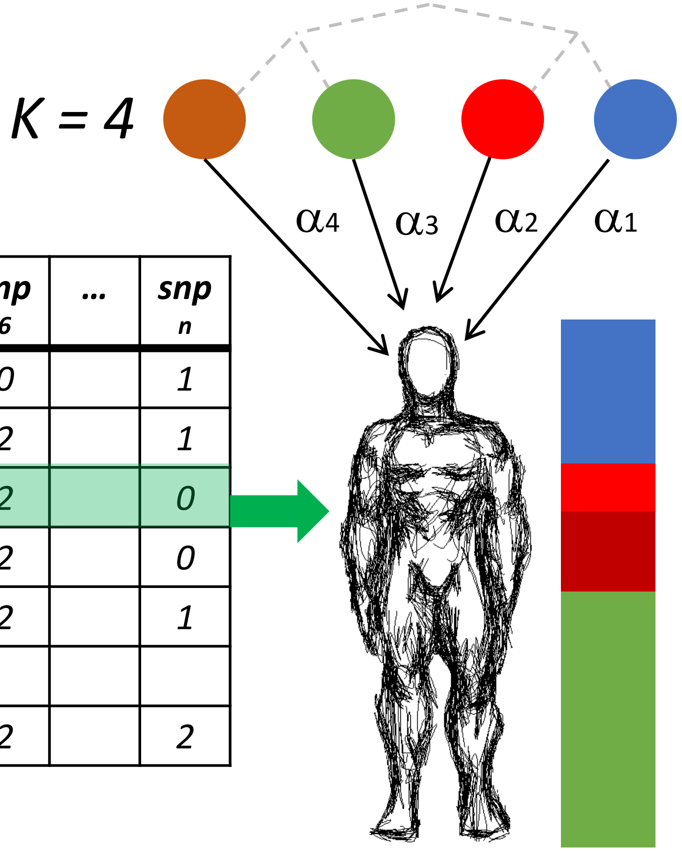
Basics of dimensionality reduction with sequences



Unconstrained dimensions

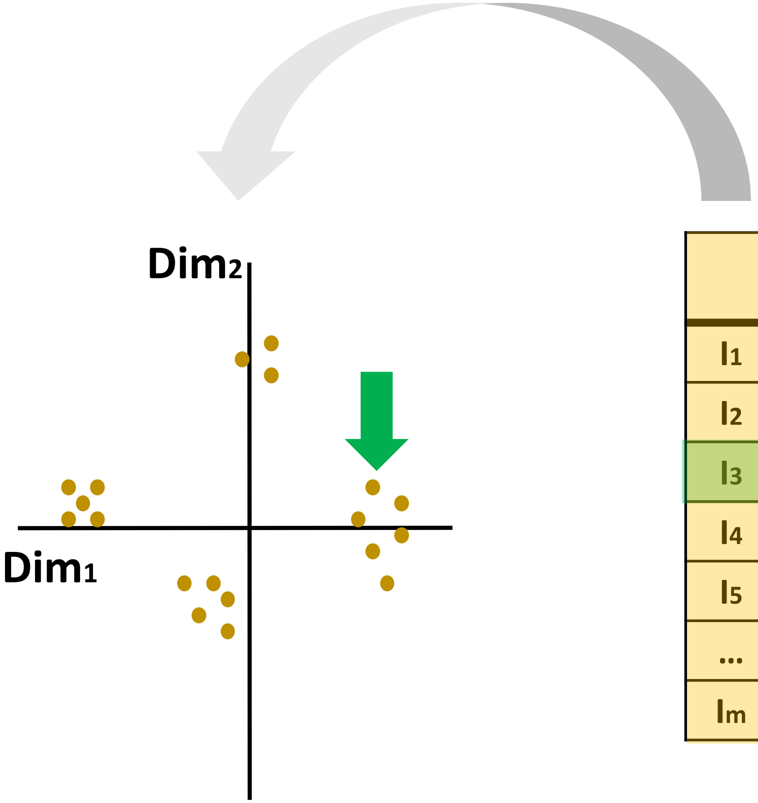
	snp 1	snp 2	snp 3	snp 4	snp 5	snp 6	…	snp n
I_1	0	1	2	1	0	0		1
I_2	0	0	2	0	0	2		1
I_3	1	1	2	0	0	2		0
I_4	0	1	2	0	1	2		0
I_5	1	1	2	0	1	2		1
…								
I_m	2	0	1	0	1	2		2

AA = 0
AB = 1
BB = 2



Constrained dimensions to a certain number (for example, 1)

Basics of dimensionality reduction with sequences

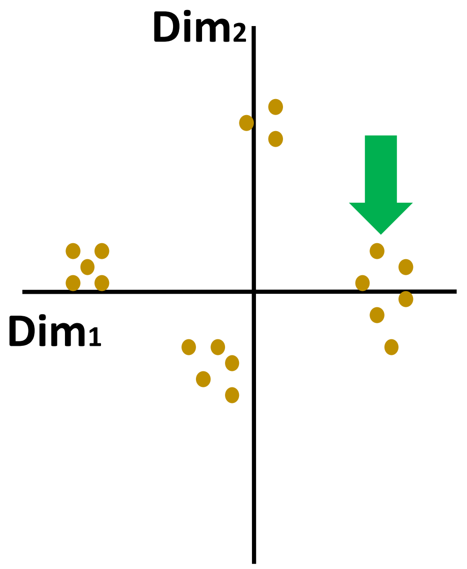


	1	*2*	*3*	*4*	*5*	*6*	…	*n*
I_1	*0*	*1*	*1*	*1*	*0*	*0*		*1*
I_2	*0*	*0*	*1*	*0*	*0*	*1*		*1*
I_3	*1*	*1*	*1*	*0*	*0*	*1*		*0*
I_4	*0*	*1*	*1*	*0*	*1*	*1*		*0*
I_5	*1*	*1*	*1*	*0*	*1*	*0*		*1*
…								
I_m	*0*	*0*	*1*	*0*	*1*	*0*		*0*

A = 0
D = 1

- Matrix of *m* OTUs and *n* polymorphic nucleotides
- The algorithm does not understand a nucleotide. It must be transformed into a number.
- After running the algorithm, we keep the variables/dimensions explaining most of the variation present in the original matrix.
- We try to identify patterns/relationships in this new space.

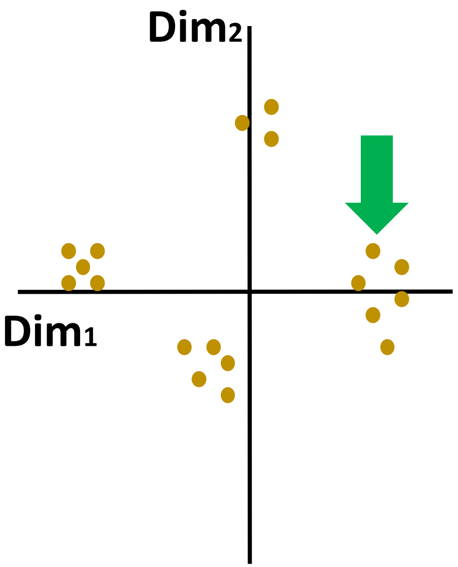
Basics of dimensionality reduction with sequences



	1	*2*	*3*	*4*	*5*	*6*	…	*n*
I_1	*0*	*1*	*1*	*1*	*0*	*0*		*1*
I_2	*0*	*0*	*1*	*0*	*0*	*1*		*1*
I_3	*1*	*1*	*1*	*0*	*0*	*1*		*0*
I_4	*0*	*1*	*1*	*0*	*1*	*1*		*0*
I_5	*1*	*1*	*1*	*0*	*1*	*0*		*1*
…								
I_m	*0*	*0*	*1*	*0*	*1*	*0*		*0*
	m1 sd1	m2 sd2	m3 sd3	m4 sd4	m5 sd5	m6 sd6		m7 sd7

- All variables must be in the same “units”, otherwise algorithms will give more weight to variables with greater variances

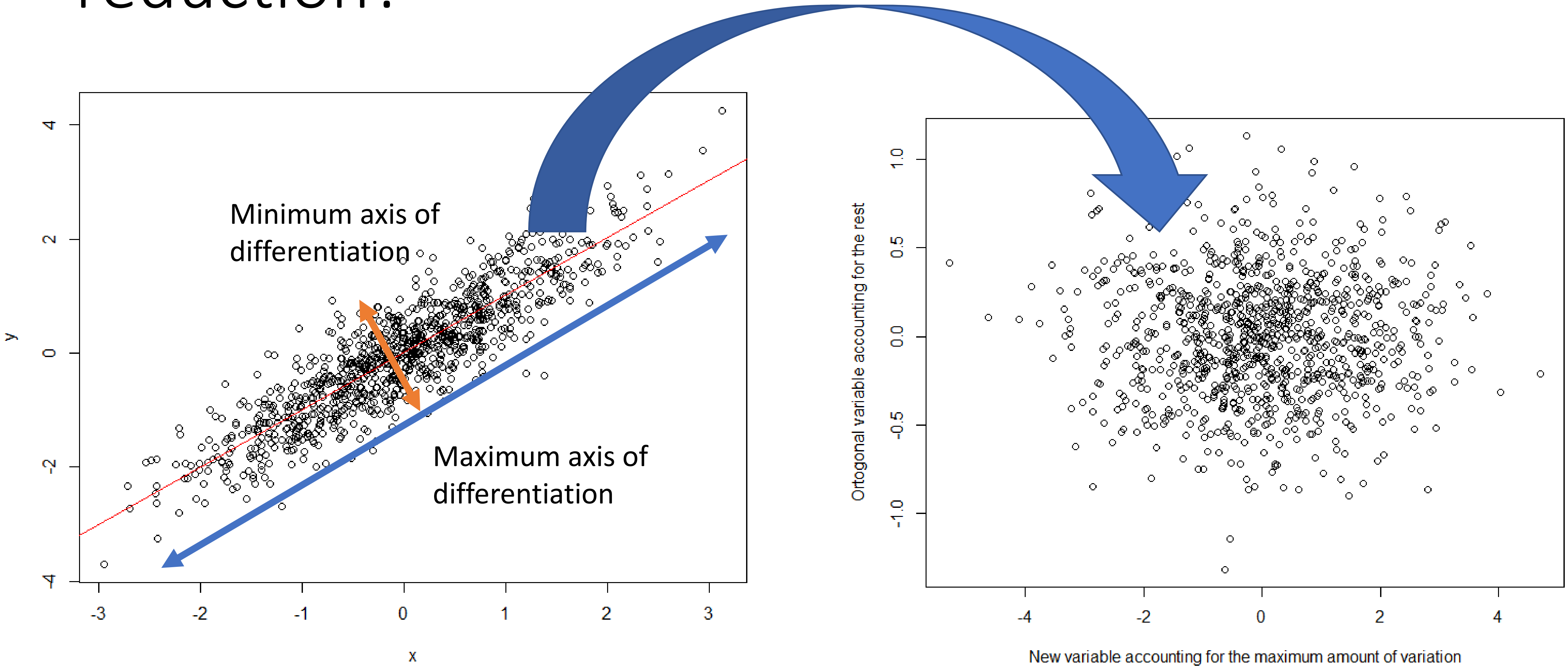
Basics of dimensionality reduction with sequences



	1	2	3	4	5	6	…	n
I_1	*(0-m1)/sd1*	*(1-m2)/sd2*	*(1-m3)/sd*					
I_2								
I_3								
I_4								
I_5								
…								
I_m								

- All variables must be in the same "units"

Which is the principle of a dimensionality reduction?



Minimum axis of differentiation

Maximum axis of differentiation

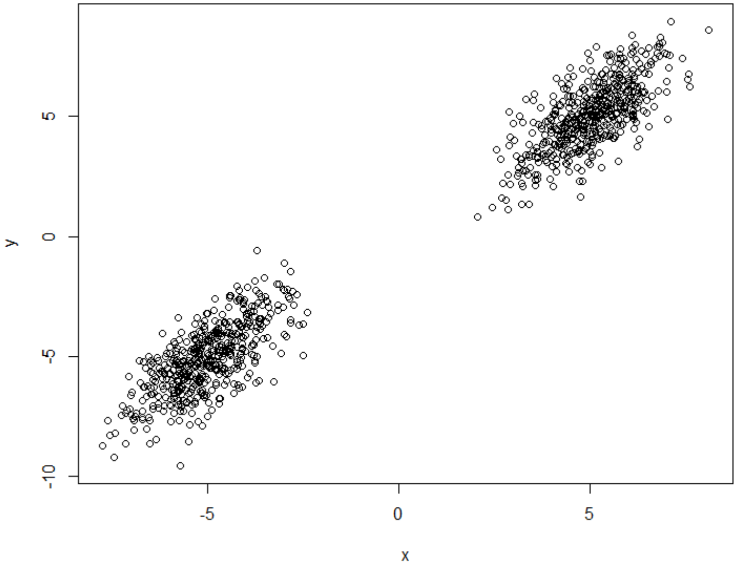
Some methods for dimensionality reduction

- Continuous variables
  - Principal Component Analysis (PCA)
  - Coordinate analysis (PCoA)/classical multidimensional scaling (MDS)
  - …
- Categorical variables
  - Categorical PCA
  - Non-metric MDS
  - …
- Variables must add to a value (i.e. percentage)
  - Correspondence analysis
  - Logratio analysis/Weighted PCA
  - …
- Non-linear PCA / Autoencoders
- tSNE
- …

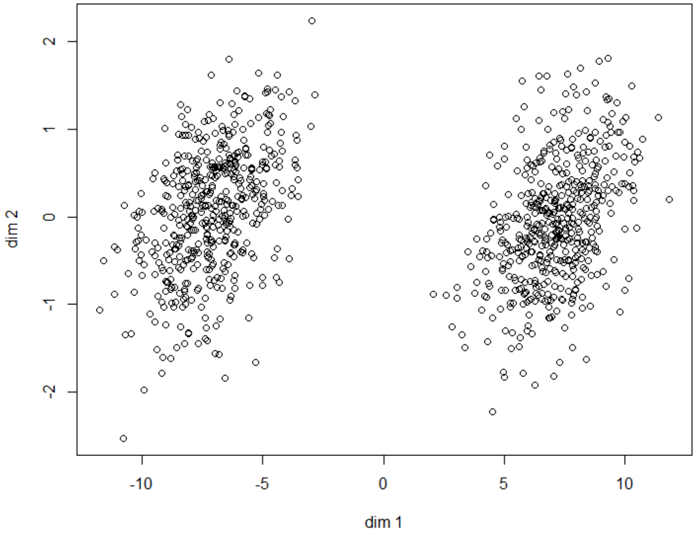
A coordinate analysis approach (classical multidimensional scaling)

$$D_{i,j} = \|x\|_2 = w_{dim1}(dim1_i - dim1_j)^2 + w_{dim2}(dim2_i - dim2_j)^2$$

$$w_{dim1} > w_{dim2}$$



	I1	I2	…	In
I1	0			
I2	$D_{2,1}$	0		
…	$D_{…,1}$	$D_{…,2}$	0	
In	$D_{n,1}$	$D_{n,2}$	$D_{n,…}$	0



$$d_{i,j} = \|x\|_2 = (x_i - x_j)^2 + (y_i - y_j)^2$$

$$d_{ij}(\mathbf{X}) = \left[\sum_{a=1}^{m}(x_{ia} - x_{ja})^2\right]^{1/2}. \qquad d_{rs}^2 = (\mathbf{x}_r - \mathbf{x}_s)^T(\mathbf{x}_r - \mathbf{x}_s).$$

Many other distances can be used!

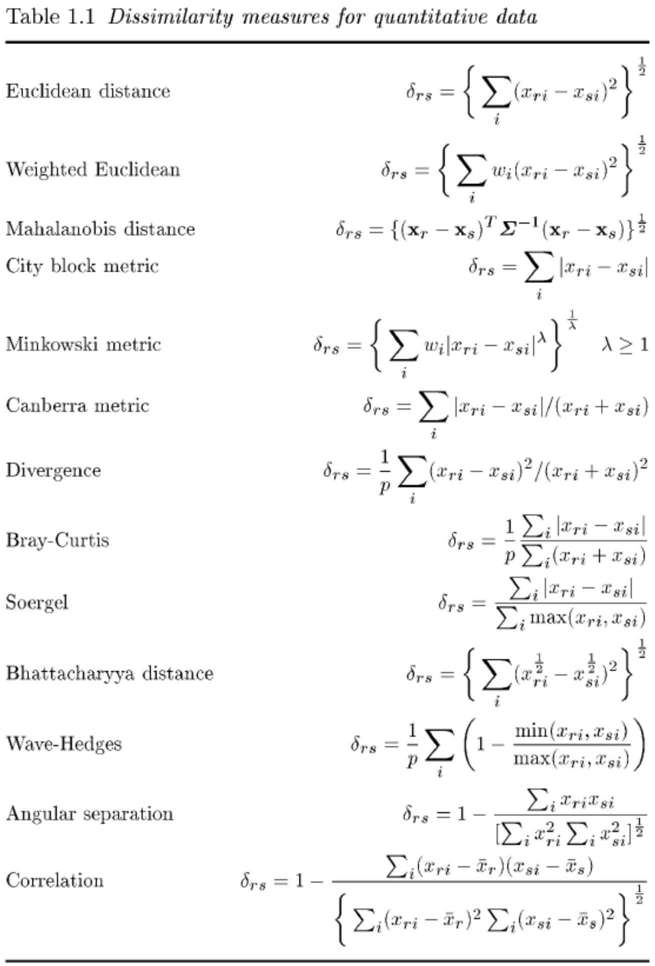
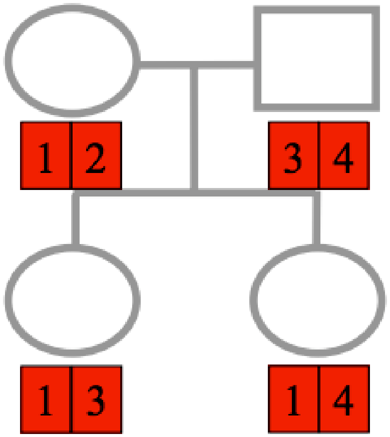


Table 1.1 *Dissimilarity measures for quantitative data*

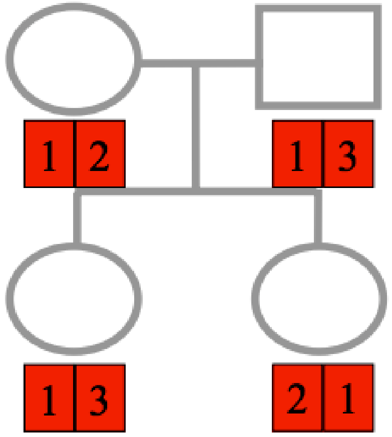
Euclidean distance	$\delta_{rs} = \left\{ \sum_i (x_{ri} - x_{si})^2 \right\}^{\frac{1}{2}}$
Weighted Euclidean	$\delta_{rs} = \left\{ \sum_i w_i (x_{ri} - x_{si})^2 \right\}^{\frac{1}{2}}$
Mahalanobis distance	$\delta_{rs} = \{(\mathbf{x}_r - \mathbf{x}_s)^T \boldsymbol{\Sigma}^{\mathbf{-1}} (\mathbf{x}_r - \mathbf{x}_s)\}^{\frac{1}{2}}$
City block metric	$\delta_{rs} = \sum_i \lvert x_{ri} - x_{si} \rvert$
Minkowski metric	$\delta_{rs} = \left\{ \sum_i w_i \lvert x_{ri} - x_{si} \rvert^{\lambda} \right\}^{\frac{1}{\lambda}} \quad \lambda \geq 1$
Canberra metric	$\delta_{rs} = \sum_i \lvert x_{ri} - x_{si} \rvert / (x_{ri} + x_{si})$
Divergence	$\delta_{rs} = \frac{1}{p} \sum_i (x_{ri} - x_{si})^2 / (x_{ri} + x_{si})^2$
Bray-Curtis	$\delta_{rs} = \frac{1}{p} \frac{\sum_i \lvert x_{ri} - x_{si} \rvert}{\sum_i (x_{ri} + x_{si})}$
Soergel	$\delta_{rs} = \frac{\sum_i \lvert x_{ri} - x_{si} \rvert}{\sum_i \max(x_{ri}, x_{si})}$
Bhattacharyya distance	$\delta_{rs} = \left\{ \sum_i (x_{ri}^{\frac{1}{2}} - x_{si}^{\frac{1}{2}})^2 \right\}^{\frac{1}{2}}$
Wave-Hedges	$\delta_{rs} = \frac{1}{p} \sum_i \left(1 - \frac{\min(x_{ri}, x_{si})}{\max(x_{ri}, x_{si})} \right)$
Angular separation	$\delta_{rs} = 1 - \frac{\sum_i x_{ri} x_{si}}{[\sum_i x_{ri}^2 \sum_i x_{si}^2]^{\frac{1}{2}}}$
Correlation	$\delta_{rs} = 1 - \frac{\sum_i (x_{ri} - \bar{x}_r)(x_{si} - \bar{x}_s)}{\left\{ \sum_i (x_{ri} - \bar{x}_r)^2 \sum_i (x_{si} - \bar{x}_s)^2 \right\}^{\frac{1}{2}}}$

Genetic distances

Between individuals



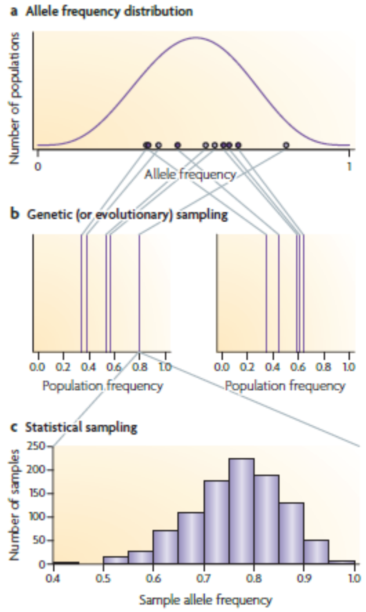
Identical by Descent
and
Identical by State



Identical by state only

Between populations

- Fst family
- Nei's standard genetic distance
- Nei's minimum genetic distance
- Informativeness of ancestry
- …



A principal coordinate analysis (PCoA) approach
Some algebra (sigh!)

$$\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 3 & 5 \\ 4 & 7 \end{bmatrix}$$

A is 3 X 2 matrix

$$\mathbf{A}' = \begin{bmatrix} 1 & 3 & 4 \\ 2 & 5 & 7 \end{bmatrix}.$$

Transpose A

$$(\mathbf{A}')' = \mathbf{A}.$$

$$\mathbf{A} + \mathbf{B} = (a_{ij} + b_{ij}) = (c_{ij}) = \mathbf{C}.$$

Add A and B

$$\mathbf{AB} = [\textstyle\sum_k a_{ik} \cdot b_{kj}] = [c_{ij}].$$

Matrix multiplication

$$< \mathrm{x}, \mathrm{y} >= \mathrm{x}'\mathrm{y} = [\begin{array}{ccc} 1 & 3 & 4 \end{array}] \begin{bmatrix} 2 \\ 5 \\ 7 \end{bmatrix} = 45.$$

Vector multiplication

$$\|\mathrm{x}\| = \sqrt{\mathrm{x}'\mathrm{x}} = (x_1^2 + \ldots + x_n^2)^{1/2}.$$

Norm of a vector

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Identity matrix

$$\mathbf{IA} = \mathbf{AI} = \mathbf{A}$$

$$\mathbf{A}'\mathbf{A} = \mathbf{I}.$$

Orthonormal matrix

$$k\mathbf{A} = (k \cdot a_{ij})$$

Multiplication by a scalar

Norm of a vector

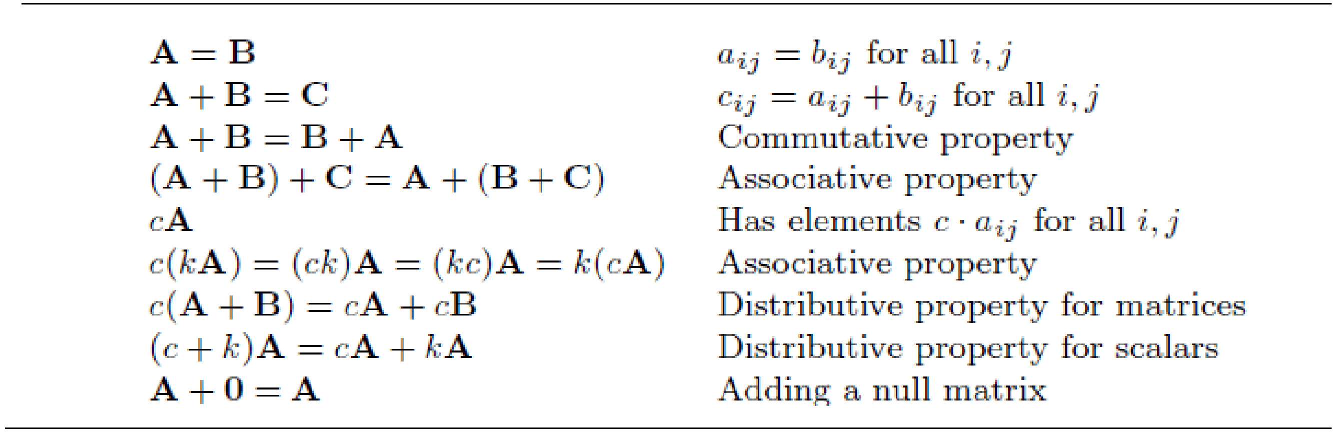
$$\begin{aligned} \|\mathrm{x}\| &\geq 0 \text{ for } \mathrm{x} \neq \mathbf{0} \text{ and} \\ \|\mathrm{x}\| &= 0 \text{ precisely when } \mathrm{x} = \mathbf{0} \text{ (nonnegativity),} \\ \|k\mathrm{x}\| &= |k|\|\mathrm{x}\|, \text{ for any scalar } k, \\ \|\mathrm{x} + \mathrm{y}\| &\leq \|\mathrm{x}\| + \|\mathrm{y}\| \text{ (triangle inequality).} \end{aligned}$$

$$\mathbf{u} = (1/\|\mathbf{x}\|)\mathbf{x} \qquad \|\mathbf{u}\| = 1.$$

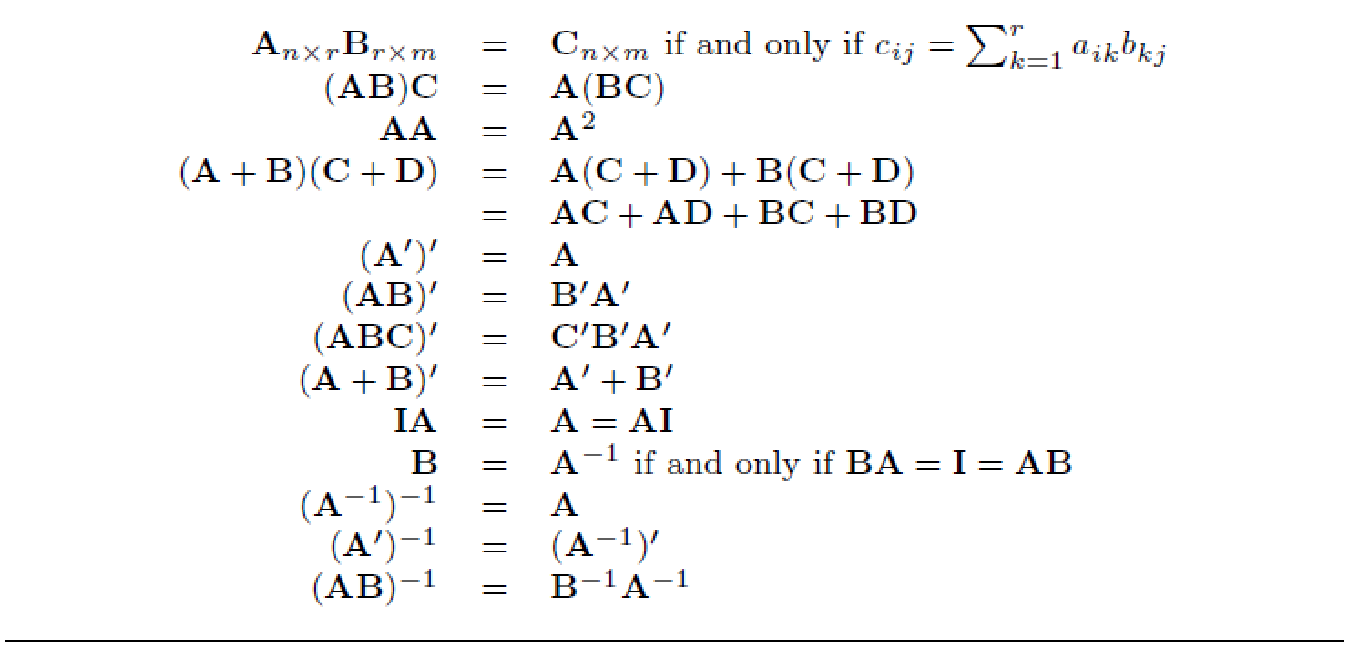
$$\text{tr } \mathbf{A} = \sum_{i=1}^{n} a_{ii},$$

Trace

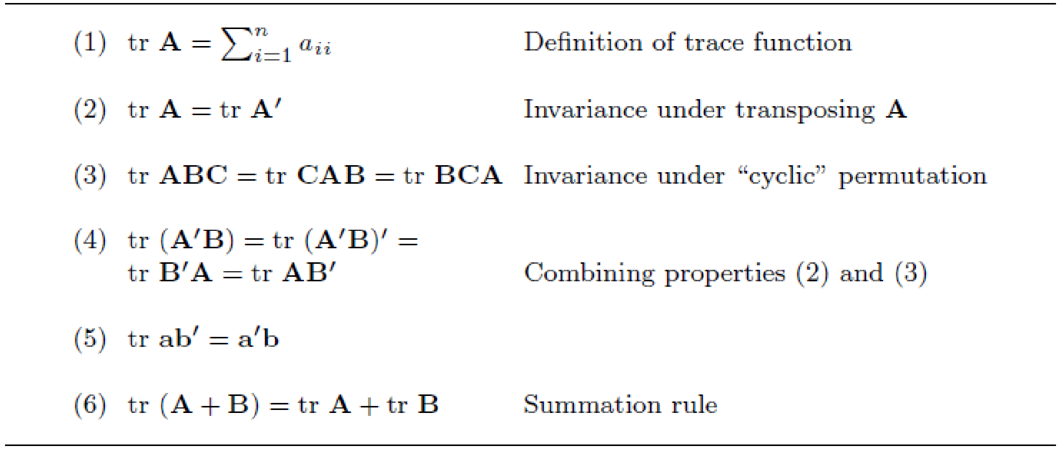
Some algebra (sigh! Sigh!)



$\mathbf{A} = \mathbf{B}$	$a_{ij} = b_{ij}$ for all i, j
$\mathbf{A} + \mathbf{B} = \mathbf{C}$	$c_{ij} = a_{ij} + b_{ij}$ for all i, j
$\mathbf{A} + \mathbf{B} = \mathbf{B} + \mathbf{A}$	Commutative property
$(\mathbf{A} + \mathbf{B}) + \mathbf{C} = \mathbf{A} + (\mathbf{B} + \mathbf{C})$	Associative property
$c\mathbf{A}$	Has elements $c \cdot a_{ij}$ for all i, j
$c(k\mathbf{A}) = (ck)\mathbf{A} = (kc)\mathbf{A} = k(c\mathbf{A})$	Associative property
$c(\mathbf{A} + \mathbf{B}) = c\mathbf{A} + c\mathbf{B}$	Distributive property for matrices
$(c + k)\mathbf{A} = c\mathbf{A} + k\mathbf{A}$	Distributive property for scalars
$\mathbf{A} + \mathbf{0} = \mathbf{A}$	Adding a null matrix

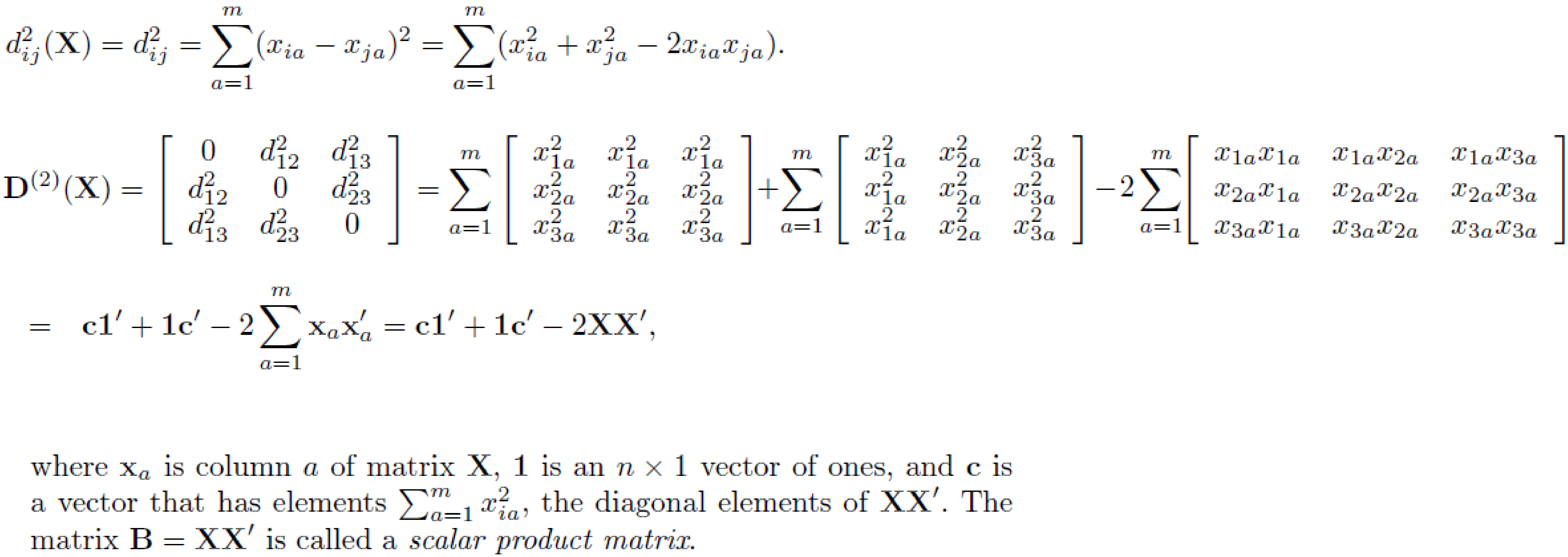


$$\begin{aligned}
\mathbf{A}_{n\times r}\mathbf{B}_{r\times m} &= \mathbf{C}_{n\times m} \text{ if and only if } c_{ij} = \textstyle\sum_{k=1}^{r} a_{ik}b_{kj} \\
(\mathbf{AB})\mathbf{C} &= \mathbf{A}(\mathbf{BC}) \\
\mathbf{AA} &= \mathbf{A}^2 \\
(\mathbf{A} + \mathbf{B})(\mathbf{C} + \mathbf{D}) &= \mathbf{A}(\mathbf{C} + \mathbf{D}) + \mathbf{B}(\mathbf{C} + \mathbf{D}) \\
&= \mathbf{AC} + \mathbf{AD} + \mathbf{BC} + \mathbf{BD} \\
(\mathbf{A}')' &= \mathbf{A} \\
(\mathbf{AB})' &= \mathbf{B}'\mathbf{A}' \\
(\mathbf{ABC})' &= \mathbf{C}'\mathbf{B}'\mathbf{A}' \\
(\mathbf{A} + \mathbf{B})' &= \mathbf{A}' + \mathbf{B}' \\
\mathbf{IA} &= \mathbf{A} = \mathbf{AI} \\
\mathbf{B} &= \mathbf{A}^{-1} \text{ if and only if } \mathbf{BA} = \mathbf{I} = \mathbf{AB} \\
(\mathbf{A}^{-1})^{-1} &= \mathbf{A} \\
(\mathbf{A}')^{-1} &= (\mathbf{A}^{-1})' \\
(\mathbf{AB})^{-1} &= \mathbf{B}^{-1}\mathbf{A}^{-1}
\end{aligned}$$



(1)	tr $\mathbf{A} = \sum_{i=1}^{n} a_{ii}$	Definition of trace function
(2)	tr $\mathbf{A}$ = tr $\mathbf{A}'$	Invariance under transposing $\mathbf{A}$
(3)	tr $\mathbf{ABC}$ = tr $\mathbf{CAB}$ = tr $\mathbf{BCA}$	Invariance under "cyclic" permutation
(4)	tr $(\mathbf{A}'\mathbf{B})$ = tr $(\mathbf{A}'\mathbf{B})'$ = tr $\mathbf{B}'\mathbf{A}$ = tr $\mathbf{AB}'$	Combining properties (2) and (3)
(5)	tr $\mathbf{ab}' = \mathbf{a}'\mathbf{b}$	
(6)	tr $(\mathbf{A} + \mathbf{B})$ = tr $\mathbf{A}$ + tr $\mathbf{B}$	Summation rule

Algebra derivation of PCoA (multidimensional classical scaling)



$$d_{ij}^2(\mathbf{X}) = d_{ij}^2 = \sum_{a=1}^{m}(x_{ia} - x_{ja})^2 = \sum_{a=1}^{m}(x_{ia}^2 + x_{ja}^2 - 2x_{ia}x_{ja}).$$

$$\mathbf{D}^{(2)}(\mathbf{X}) = \begin{bmatrix} 0 & d_{12}^2 & d_{13}^2 \\ d_{12}^2 & 0 & d_{23}^2 \\ d_{13}^2 & d_{23}^2 & 0 \end{bmatrix} = \sum_{a=1}^{m} \begin{bmatrix} x_{1a}^2 & x_{1a}^2 & x_{1a}^2 \\ x_{2a}^2 & x_{2a}^2 & x_{2a}^2 \\ x_{3a}^2 & x_{3a}^2 & x_{3a}^2 \end{bmatrix} + \sum_{a=1}^{m} \begin{bmatrix} x_{1a}^2 & x_{2a}^2 & x_{3a}^2 \\ x_{1a}^2 & x_{2a}^2 & x_{3a}^2 \\ x_{1a}^2 & x_{2a}^2 & x_{3a}^2 \end{bmatrix} - 2\sum_{a=1}^{m} \begin{bmatrix} x_{1a}x_{1a} & x_{1a}x_{2a} & x_{1a}x_{3a} \\ x_{2a}x_{1a} & x_{2a}x_{2a} & x_{2a}x_{3a} \\ x_{3a}x_{1a} & x_{3a}x_{2a} & x_{3a}x_{3a} \end{bmatrix}$$

$$= \quad \mathbf{c1}' + \mathbf{1c}' - 2\sum_{a=1}^{m} \mathbf{x}_a\mathbf{x}_a' = \mathbf{c1}' + \mathbf{1c}' - 2\mathbf{XX}',$$

where $\mathbf{x}_a$ is column a of matrix $\mathbf{X}$, $\mathbf{1}$ is an $n \times 1$ vector of ones, and $\mathbf{c}$ is a vector that has elements $\sum_{a=1}^{m} x_{ia}^2$, the diagonal elements of $\mathbf{XX}'$. The matrix $\mathbf{B} = \mathbf{XX}'$ is called a *scalar product matrix.*

Algebra derivation of PCoA (classical multidimensional scaling)

$$\mathbf{X} = \begin{bmatrix} 1 & 2 \\ 3 & 1 \\ 2 & 0 \end{bmatrix} = \begin{array}{c|c|c|} & \mathbf{x}_1 & \mathbf{x}_2 \\ \hline p_1 & 1 & 2 \\ \hline p_2 & 3 & 1 \\ \hline p_3 & 2 & 0 \\ \hline \end{array}$$

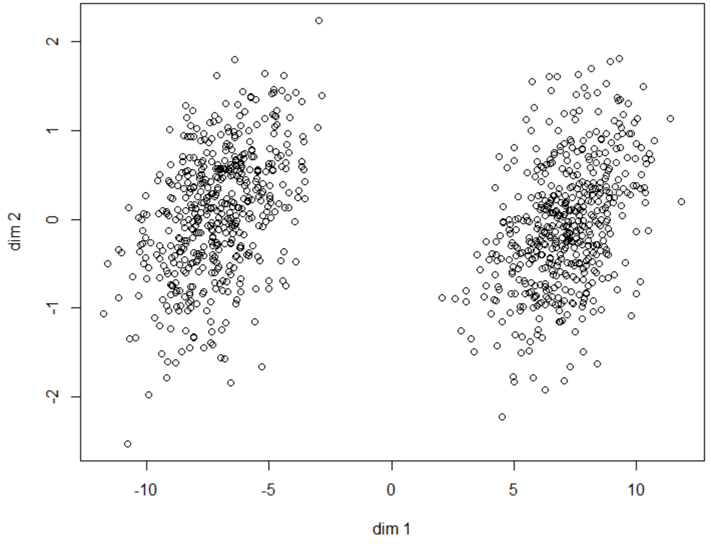
$$\mathbf{XX}' = \begin{bmatrix} 1 & 2 \\ 3 & 1 \\ 2 & 0 \end{bmatrix} \begin{bmatrix} 1 & 3 & 2 \\ 2 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 5 & 5 & 2 \\ 5 & 10 & 6 \\ 2 & 6 & 4 \end{bmatrix} = \mathbf{B}.$$

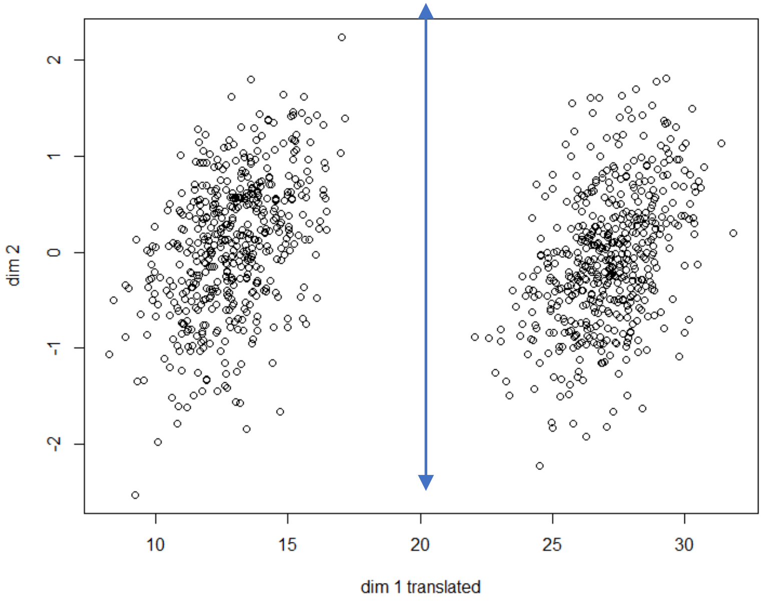
$$\mathbf{c}' = (5, 10, 4)$$

$$\begin{aligned} \mathbf{D}^{(2)}(\mathbf{X}) &= \begin{bmatrix} 0 & d_{12}^2 & d_{13}^2 \\ d_{12}^2 & 0 & d_{23}^2 \\ d_{13}^2 & d_{23}^2 & 0 \end{bmatrix} = \begin{bmatrix} 5 & 5 & 5 \\ 10 & 10 & 10 \\ 4 & 4 & 4 \end{bmatrix} \\ &+ \begin{bmatrix} 5 & 10 & 4 \\ 5 & 10 & 4 \\ 5 & 10 & 4 \end{bmatrix} - 2 \begin{bmatrix} 5 & 5 & 2 \\ 5 & 10 & 6 \\ 2 & 6 & 4 \end{bmatrix} = \begin{bmatrix} 0 & 5 & 5 \\ 5 & 0 & 2 \\ 5 & 2 & 0 \end{bmatrix}. \end{aligned}$$

A coordinate analysis approach Step by Step

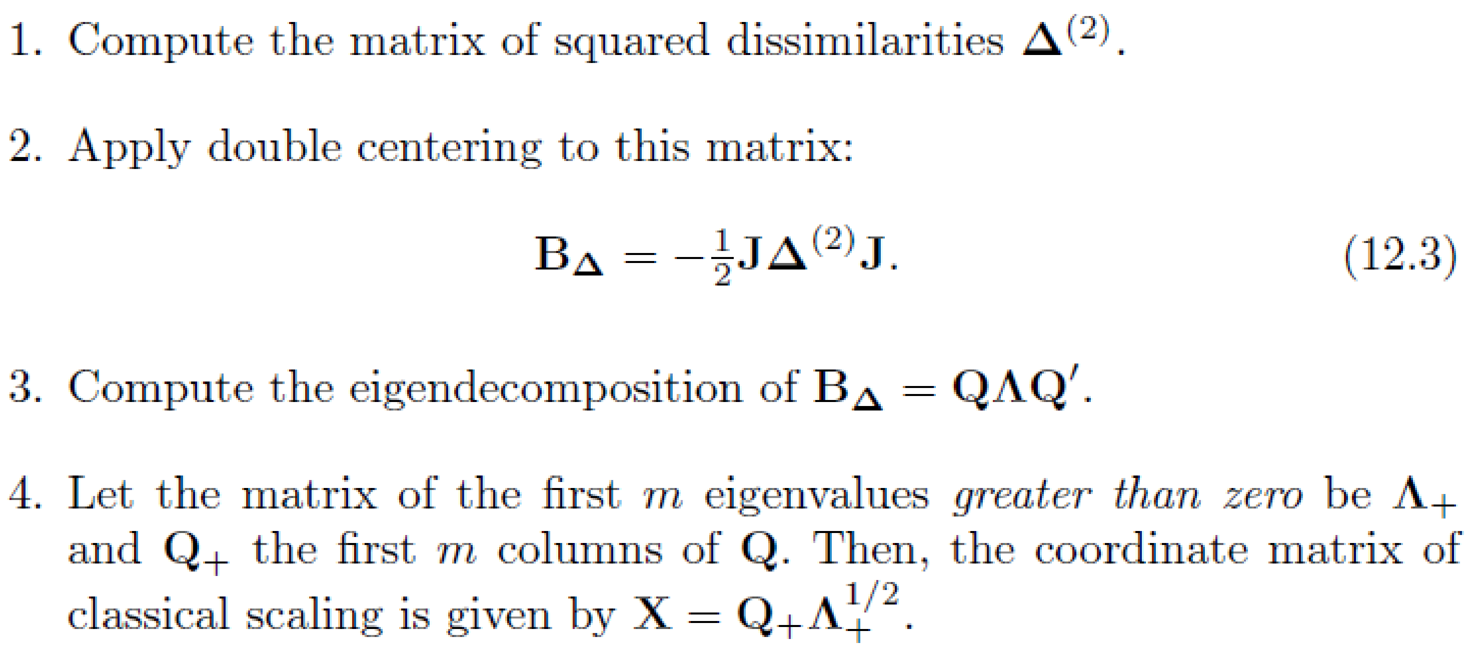
	I1	I2	…	In
I1	0			
I2	$D_{2,1}$	0		
…	$D_{\dots,1}$	$D_{\dots,2}$	0	
In	$D_{n,1}$	$D_{n,2}$	$D_{n,\dots}$	0





The same distance matrix can produce infinite solutions…
We will center the solution by
setting that the sum of dim1 = 0 and sum dim2 = 0.

The algorithm



1. Compute the matrix of squared dissimilarities $\mathbf{\Delta}^{(2)}$.

2. Apply double centering to this matrix:

$$\mathbf{B}_{\mathbf{\Delta}} = -\tfrac{1}{2}\mathbf{J}\mathbf{\Delta}^{(2)}\mathbf{J}. \tag{12.3}$$

3. Compute the eigendecomposition of $\mathbf{B}_{\mathbf{\Delta}} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}'$.

4. Let the matrix of the first m eigenvalues *greater than zero* be $\mathbf{\Lambda}_+$ and $\mathbf{Q}_+$ the first m columns of $\mathbf{Q}$. Then, the coordinate matrix of classical scaling is given by $\mathbf{X} = \mathbf{Q}_+\mathbf{\Lambda}_+^{1/2}$.

Interpretation of results

$$\mathbf{B}_\Delta = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}' \qquad \mathbf{X} = \mathbf{Q}_+\mathbf{\Lambda}_+^{1/2}$$

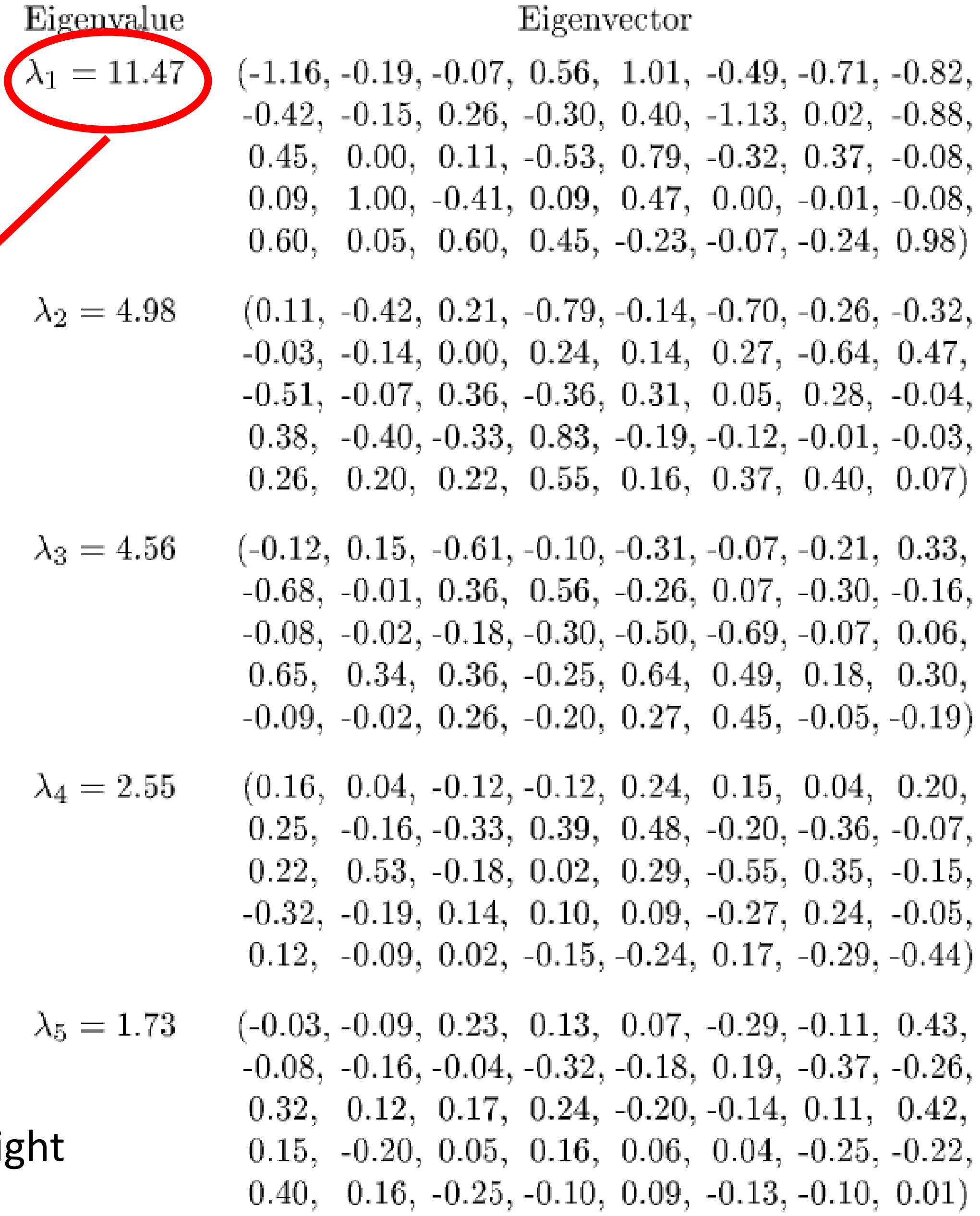
Eigenvalues (n-1)

Eigenvector (n-1)

Positive

$$d_{rs}^2 = \sum_{i=1}^{n-1} \boxed{\lambda_i} (x_{ri} - x_{si})^2,$$

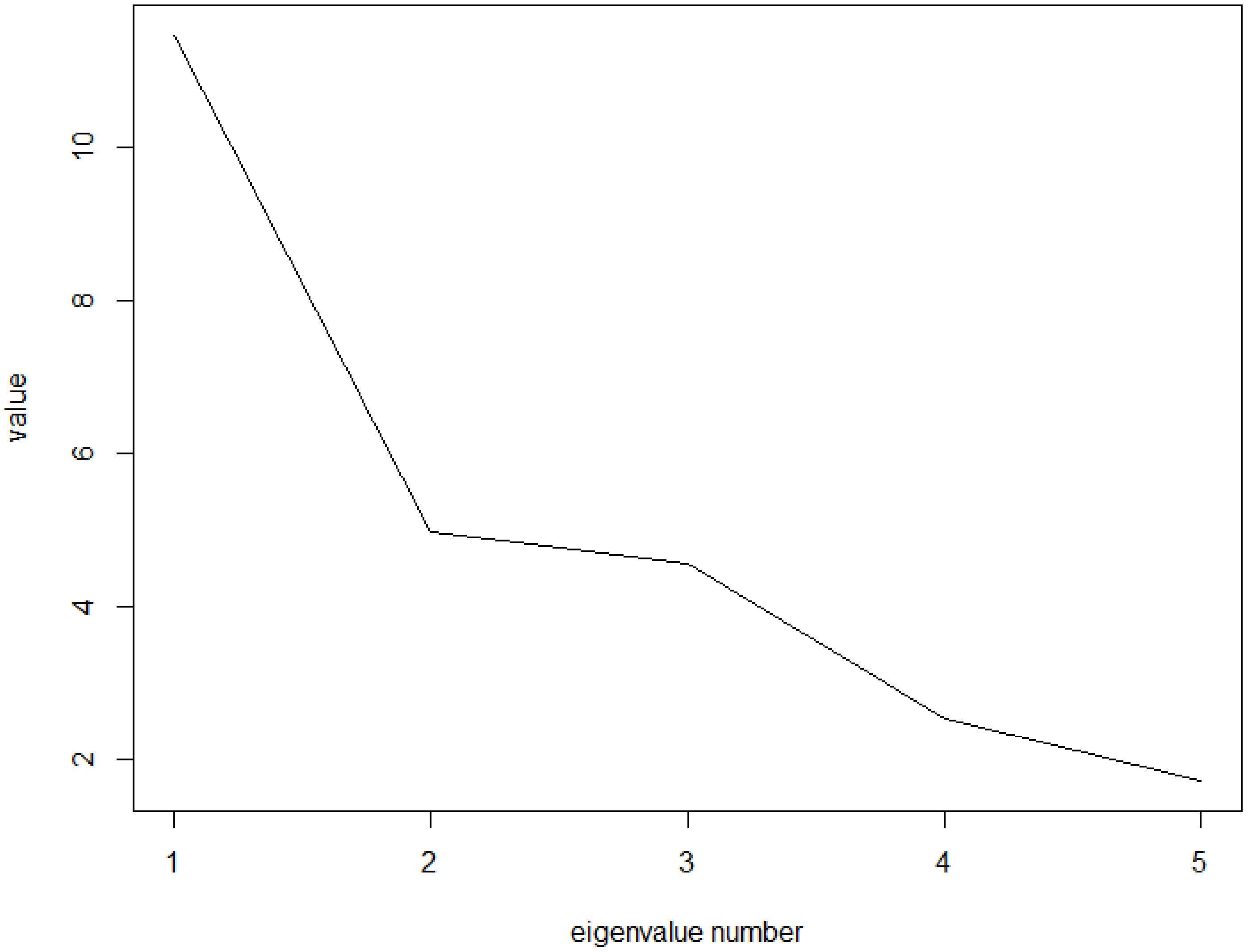
More weight



Eigenvalue	Eigenvector
$\lambda_1 = 11.47$	(-1.16, -0.19, -0.07, 0.56, 1.01, -0.49, -0.71, -0.82, -0.42, -0.15, 0.26, -0.30, 0.40, -1.13, 0.02, -0.88, 0.45, 0.00, 0.11, -0.53, 0.79, -0.32, 0.37, -0.08, 0.09, 1.00, -0.41, 0.09, 0.47, 0.00, -0.01, -0.08, 0.60, 0.05, 0.60, 0.45, -0.23, -0.07, -0.24, 0.98)
$\lambda_2 = 4.98$	(0.11, -0.42, 0.21, -0.79, -0.14, -0.70, -0.26, -0.32, -0.03, -0.14, 0.00, 0.24, 0.14, 0.27, -0.64, 0.47, -0.51, -0.07, 0.36, -0.36, 0.31, 0.05, 0.28, -0.04, 0.38, -0.40, -0.33, 0.83, -0.19, -0.12, -0.01, -0.03, 0.26, 0.20, 0.22, 0.55, 0.16, 0.37, 0.40, 0.07)
$\lambda_3 = 4.56$	(-0.12, 0.15, -0.61, -0.10, -0.31, -0.07, -0.21, 0.33, -0.68, -0.01, 0.36, 0.56, -0.26, 0.07, -0.30, -0.16, -0.08, -0.02, -0.18, -0.30, -0.50, -0.69, -0.07, 0.06, 0.65, 0.34, 0.36, -0.25, 0.64, 0.49, 0.18, 0.30, -0.09, -0.02, 0.26, -0.20, 0.27, 0.45, -0.05, -0.19)
$\lambda_4 = 2.55$	(0.16, 0.04, -0.12, -0.12, 0.24, 0.15, 0.04, 0.20, 0.25, -0.16, -0.33, 0.39, 0.48, -0.20, -0.36, -0.07, 0.22, 0.53, -0.18, 0.02, 0.29, -0.55, 0.35, -0.15, -0.32, -0.19, 0.14, 0.10, 0.09, -0.27, 0.24, -0.05, 0.12, -0.09, 0.02, -0.15, -0.24, 0.17, -0.29, -0.44)
$\lambda_5 = 1.73$	(-0.03, -0.09, 0.23, 0.13, 0.07, -0.29, -0.11, 0.43, -0.08, -0.16, -0.04, -0.32, -0.18, 0.19, -0.37, -0.26, 0.32, 0.12, 0.17, 0.24, -0.20, -0.14, 0.11, 0.42, 0.15, -0.20, 0.05, 0.16, 0.06, 0.04, -0.25, -0.22, 0.40, 0.16, -0.25, -0.10, 0.09, -0.13, -0.10, 0.01)

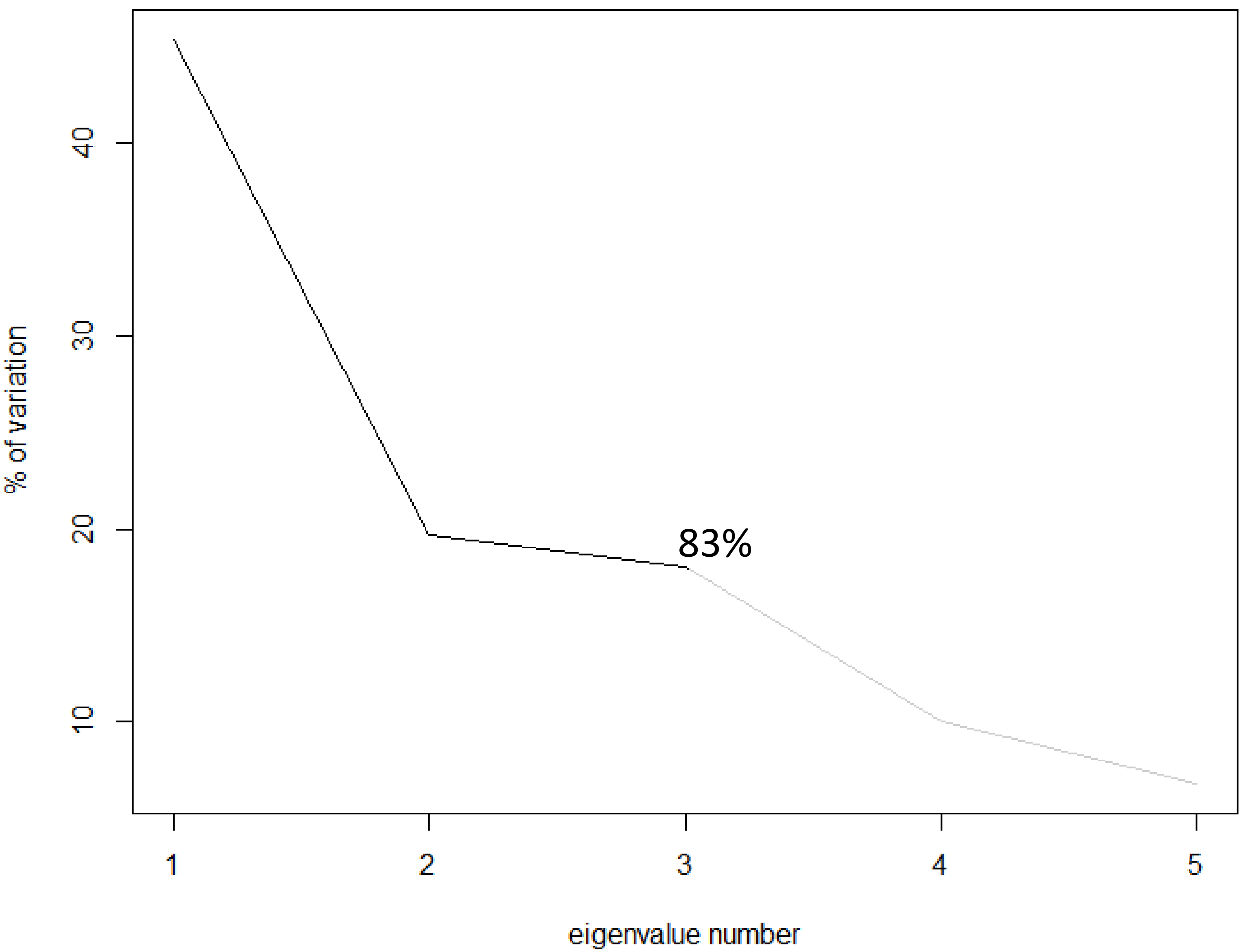
Less weight

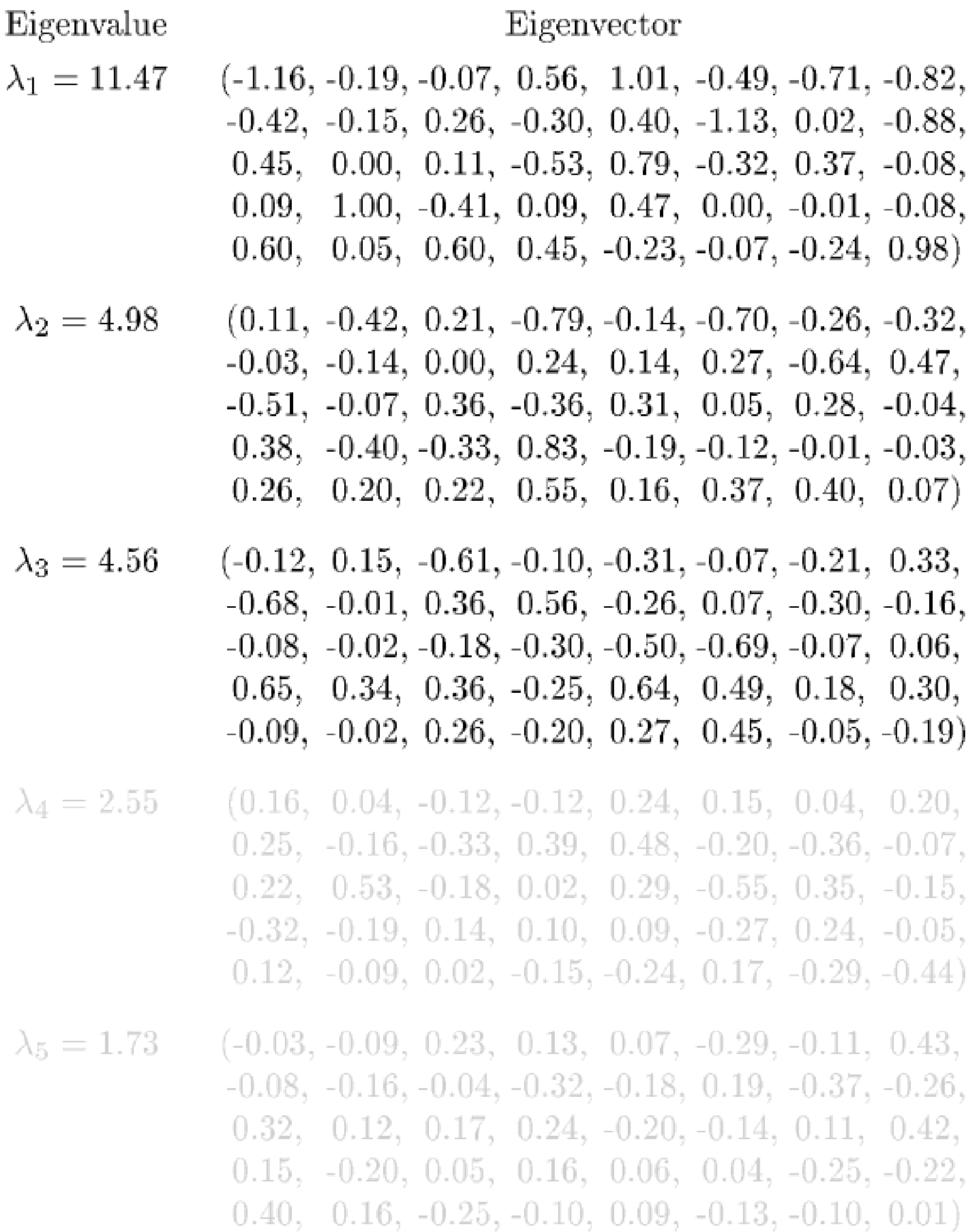
Interpretation of results



Eigenvalue	Eigenvector
$\lambda_1 = 11.47$	(-1.16, -0.19, -0.07, 0.56, 1.01, -0.49, -0.71, -0.82, -0.42, -0.15, 0.26, -0.30, 0.40, -1.13, 0.02, -0.88, 0.45, 0.00, 0.11, -0.53, 0.79, -0.32, 0.37, -0.08, 0.09, 1.00, -0.41, 0.09, 0.47, 0.00, -0.01, -0.08, 0.60, 0.05, 0.60, 0.45, -0.23, -0.07, -0.24, 0.98)
$\lambda_2 = 4.98$	(0.11, -0.42, 0.21, -0.79, -0.14, -0.70, -0.26, -0.32, -0.03, -0.14, 0.00, 0.24, 0.14, 0.27, -0.64, 0.47, -0.51, -0.07, 0.36, -0.36, 0.31, 0.05, 0.28, -0.04, 0.38, -0.40, -0.33, 0.83, -0.19, -0.12, -0.01, -0.03, 0.26, 0.20, 0.22, 0.55, 0.16, 0.37, 0.40, 0.07)
$\lambda_3 = 4.56$	(-0.12, 0.15, -0.61, -0.10, -0.31, -0.07, -0.21, 0.33, -0.68, -0.01, 0.36, 0.56, -0.26, 0.07, -0.30, -0.16, -0.08, -0.02, -0.18, -0.30, -0.50, -0.69, -0.07, 0.06, 0.65, 0.34, 0.36, -0.25, 0.64, 0.49, 0.18, 0.30, -0.09, -0.02, 0.26, -0.20, 0.27, 0.45, -0.05, -0.19)
$\lambda_4 = 2.55$	(0.16, 0.04, -0.12, -0.12, 0.24, 0.15, 0.04, 0.20, 0.25, -0.16, -0.33, 0.39, 0.48, -0.20, -0.36, -0.07, 0.22, 0.53, -0.18, 0.02, 0.29, -0.55, 0.35, -0.15, -0.32, -0.19, 0.14, 0.10, 0.09, -0.27, 0.24, -0.05, 0.12, -0.09, 0.02, -0.15, -0.24, 0.17, -0.29, -0.44)
$\lambda_5 = 1.73$	(-0.03, -0.09, 0.23, 0.13, 0.07, -0.29, -0.11, 0.43, -0.08, -0.16, -0.04, -0.32, -0.18, 0.19, -0.37, -0.26, 0.32, 0.12, 0.17, 0.24, -0.20, -0.14, 0.11, 0.42, 0.15, -0.20, 0.05, 0.16, 0.06, 0.04, -0.25, -0.22, 0.40, 0.16, -0.25, -0.10, 0.09, -0.13, -0.10, 0.01)

Interpretation of results



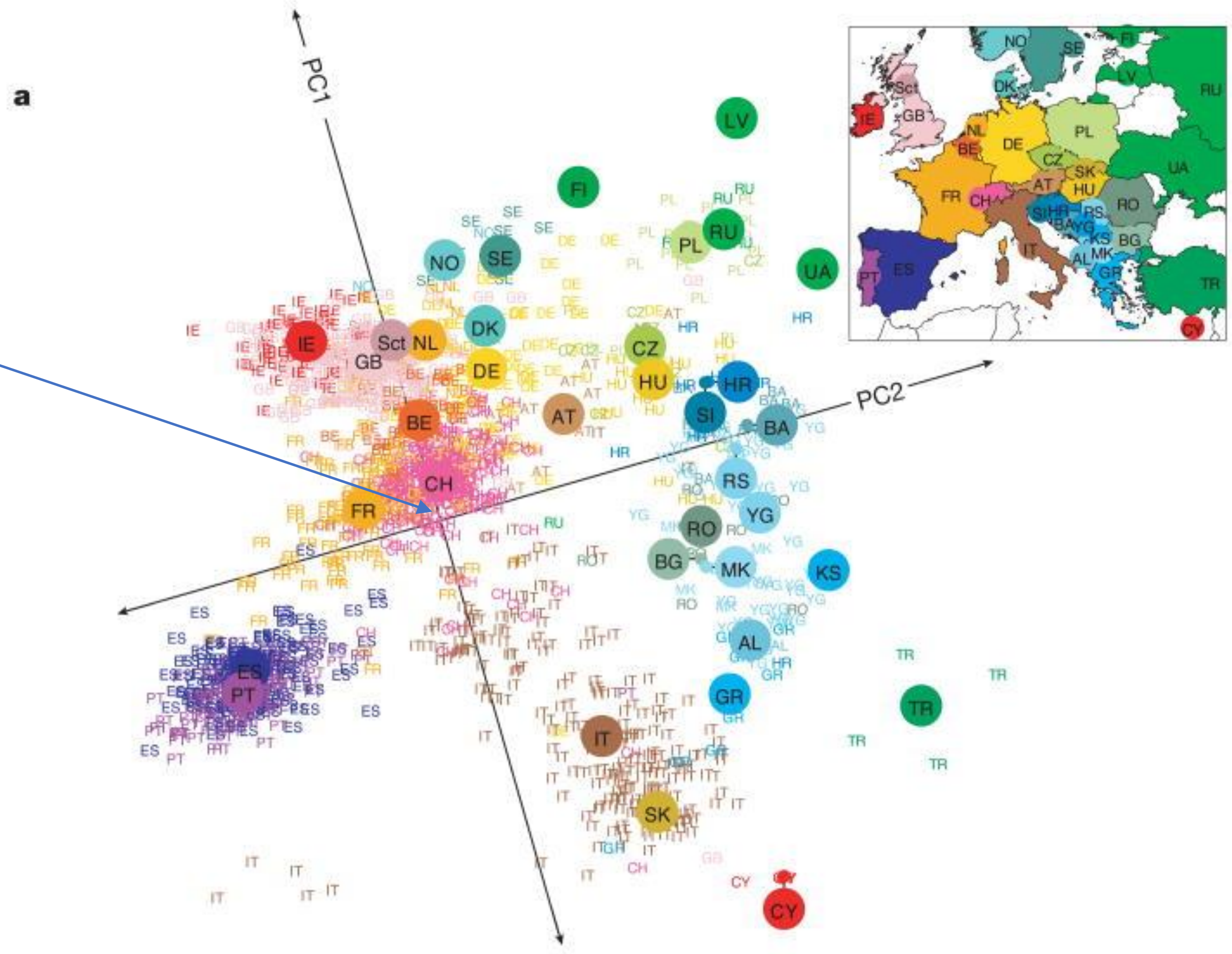


Eigenvalue	Eigenvector
$\lambda_1 = 11.47$	(-1.16, -0.19, -0.07, 0.56, 1.01, -0.49, -0.71, -0.82, -0.42, -0.15, 0.26, -0.30, 0.40, -1.13, 0.02, -0.88, 0.45, 0.00, 0.11, -0.53, 0.79, -0.32, 0.37, -0.08, 0.09, 1.00, -0.41, 0.09, 0.47, 0.00, -0.01, -0.08, 0.60, 0.05, 0.60, 0.45, -0.23, -0.07, -0.24, 0.98)
$\lambda_2 = 4.98$	(0.11, -0.42, 0.21, -0.79, -0.14, -0.70, -0.26, -0.32, -0.03, -0.14, 0.00, 0.24, 0.14, 0.27, -0.64, 0.47, -0.51, -0.07, 0.36, -0.36, 0.31, 0.05, 0.28, -0.04, 0.38, -0.40, -0.33, 0.83, -0.19, -0.12, -0.01, -0.03, 0.26, 0.20, 0.22, 0.55, 0.16, 0.37, 0.40, 0.07)
$\lambda_3 = 4.56$	(-0.12, 0.15, -0.61, -0.10, -0.31, -0.07, -0.21, 0.33, -0.68, -0.01, 0.36, 0.56, -0.26, 0.07, -0.30, -0.16, -0.08, -0.02, -0.18, -0.30, -0.50, -0.69, -0.07, 0.06, 0.65, 0.34, 0.36, -0.25, 0.64, 0.49, 0.18, 0.30, -0.09, -0.02, 0.26, -0.20, 0.27, 0.45, -0.05, -0.19)
$\lambda_4 = 2.55$	(0.16, 0.04, -0.12, -0.12, 0.24, 0.15, 0.04, 0.20, 0.25, -0.16, -0.33, 0.39, 0.48, -0.20, -0.36, -0.07, 0.22, 0.53, -0.18, 0.02, 0.29, -0.55, 0.35, -0.15, -0.32, -0.19, 0.14, 0.10, 0.09, -0.27, 0.24, -0.05, 0.12, -0.09, 0.02, -0.15, -0.24, 0.17, -0.29, -0.44)
$\lambda_5 = 1.73$	(-0.03, -0.09, 0.23, 0.13, 0.07, -0.29, -0.11, 0.43, -0.08, -0.16, -0.04, -0.32, -0.18, 0.19, -0.37, -0.26, 0.32, 0.12, 0.17, 0.24, -0.20, -0.14, 0.11, 0.42, 0.15, -0.20, 0.05, 0.16, 0.06, 0.04, -0.25, -0.22, 0.40, 0.16, -0.25, -0.10, 0.09, -0.13, -0.10, 0.01)

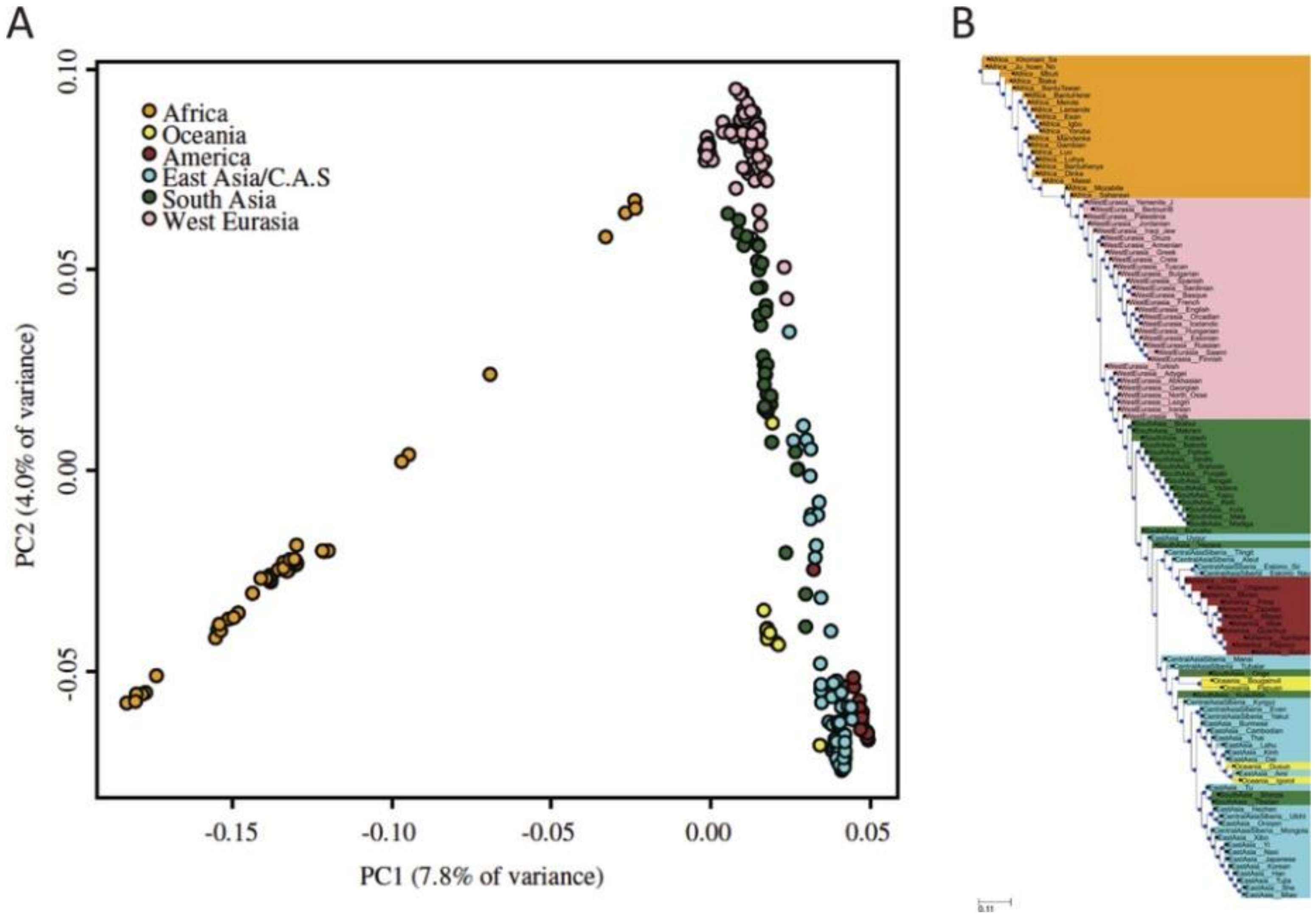
Example: European genetic substructure

Novembre et al 2007 Nature

Orthogonal new variables



Example: worldwide populations



Another way of computing the Principal Coordinate analysis. Error between the real distance matrix and the distance matrix computed with the new coordinates

$$e_{ij}^2 = (d_{ij} - \delta_{ij})^2$$

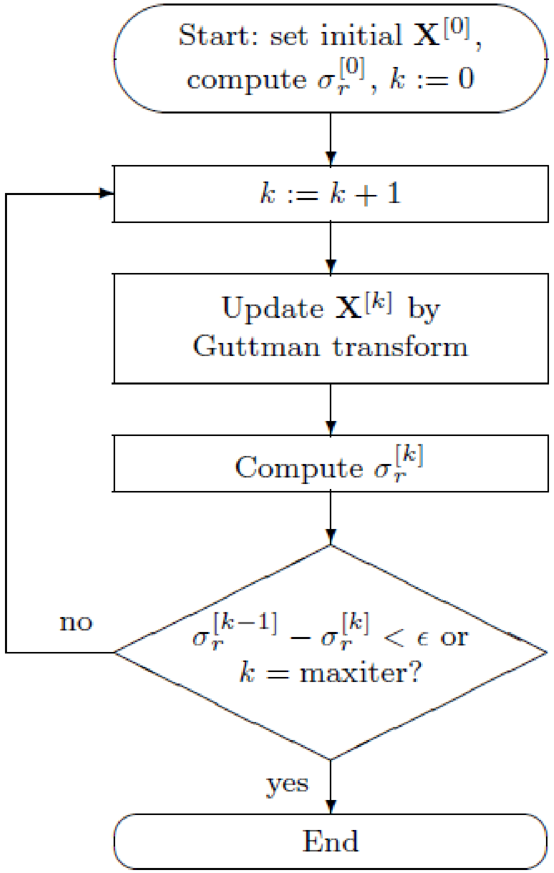
$$\sigma_r(\mathbf{X}) = \sum_{i=1}^{n} \sum_{j=i+1}^{n} (d_{ij} - \delta_{ij})^2 \quad = \quad \sigma_r(\mathbf{X}) = \sum_{i<j} (d_{ij} - \delta_{ij})^2 \quad \text{for all available } \delta_{ij}$$

Raw stress $=$

$$\sigma_r(\mathbf{X}) = \sum_{i<j} w_{ij} (d_{ij}(\mathbf{X}) - \delta_{ij})^2 \quad \begin{cases} w_{ij} = 0 \text{ if } \delta_{ij} \text{ is missing} \\ w_{ij} \text{ with value 1 if } \delta_{ij} \end{cases}$$

$$\begin{aligned} \sigma_r(\mathbf{X}) &= \sum_{i<j} w_{ij} \left(\delta_{ij} - d_{ij}(\mathbf{X})\right)^2 \\ &= \sum_{i<j} w_{ij} \delta_{ij}^2 + \sum_{i<j} w_{ij} d_{ij}^2(\mathbf{X}) - 2 \sum_{i<j} w_{ij} \delta_{ij} d_{ij}(\mathbf{X}) \\ &= \eta_\delta^2 + \eta^2(\mathbf{X}) - 2\rho(\mathbf{X}), \end{aligned}$$

SMACOF algorithm


Start: set initial $\mathbf{X}^{[0]}$, compute $\sigma_r^{[0]}$, $k := 0$
$k := k + 1$
Update $\mathbf{X}^{[k]}$ by Guttman transform
Compute $\sigma_r^{[k]}$
$\sigma_r^{[k-1]} - \sigma_r^{[k]} < \epsilon$ or $k =$ maxiter?
no
yes
End

A working example

$$\boldsymbol{\Delta} = \begin{bmatrix} 0 & 5 & 3 & 4 \\ 5 & 0 & 2 & 2 \\ 3 & 2 & 0 & 1 \\ 4 & 2 & 1 & 0 \end{bmatrix}$$

and the matrix of coordinates $\mathbf{Z}$ and their distances be

$$\mathbf{Z} = \begin{bmatrix} -.266 & -.539 \\ .451 & .252 \\ .016 & -.238 \\ -.200 & .524 \end{bmatrix} \quad \text{and} \quad \mathbf{D}(\mathbf{Z}) = \begin{bmatrix} .000 & 1.068 & .412 & 1.065 \\ 1.068 & .000 & .655 & .706 \\ .412 & .655 & .000 & .792 \\ 1.065 & .706 & .792 & .000 \end{bmatrix}$$

$$\sigma_r(\mathbf{X}^{[0]}) = 34.29899413.$$

$$\begin{aligned} \mathbf{X}^u &= n^{-1}\mathbf{B}(\mathbf{Z})\mathbf{Z} \\ &= \frac{1}{4}\begin{bmatrix} 15.712 & -4.683 & -7.273 & -3.756 \\ -4.683 & 10.570 & -3.052 & -2.835 \\ -7.273 & -3.052 & 11.588 & -1.263 \\ -3.756 & -2.835 & -1.263 & 7.853 \end{bmatrix} \begin{bmatrix} -.266 & -.539 \\ .451 & .252 \\ .016 & -.238 \\ -.200 & .524 \end{bmatrix} \end{aligned}$$

$\mathbf{B}(\mathbf{Z})$ has elements

$$\begin{aligned} b_{ij} &= \begin{cases} -\dfrac{w_{ij}\delta_{ij}}{d_{ij}(\mathbf{Z})} & \text{for } i \neq j \text{ and } d_{ij}(\mathbf{Z}) \neq 0 \\ 0 & \text{for } i \neq j \text{ and } d_{ij}(\mathbf{Z}) = 0 \end{cases} \\ b_{ii} &= -\sum_{j=1, j\neq i}^{n} b_{ij}. \end{aligned}$$

$$\begin{aligned} b_{12} &= -w_{12}\delta_{12}/d_{12}(\mathbf{Z}) = -5/1.068 = -4.682 \\ b_{13} &= -w_{13}\delta_{13}/d_{13}(\mathbf{Z}) = -3/0.412 = -7.273 \\ b_{14} &= -w_{14}\delta_{14}/d_{14}(\mathbf{Z}) = -4/1.065 = -3.756 \\ b_{11} &= -(b_{12} + b_{13} + b_{14}) = -(-4.682 - 7.273 - 3.756) = 15.712. \end{aligned}$$

$$\sigma_r(\mathbf{X}^{[1]}) = 0.58367883$$

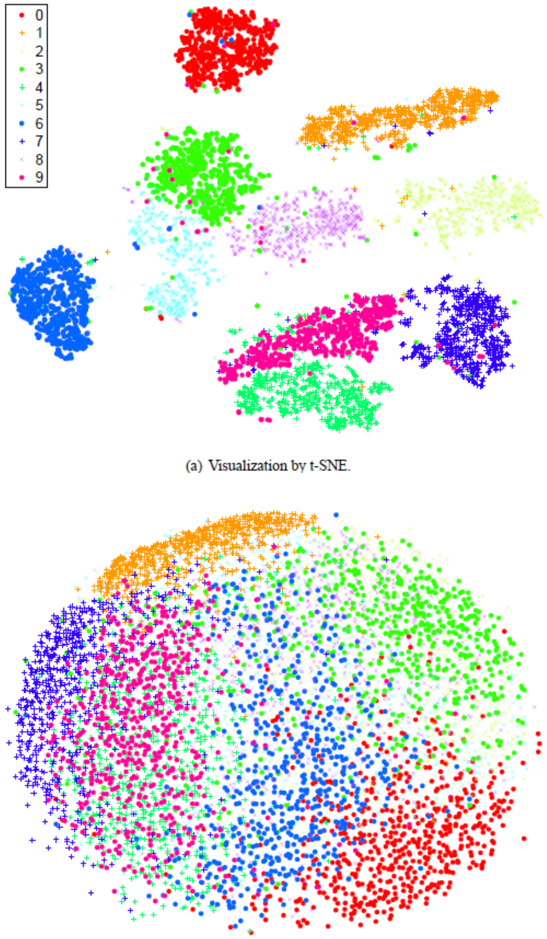
More complex distance based transformations. tSNE (Stochastic Neighbor Embedding)

The conditional probability that xi would take j as neighbor

$$p_{j|i} = \frac{\exp\left(-\|x_i - x_j\|^2 / 2\sigma_i^2\right)}{\sum_{k\neq i} \exp\left(-\|x_i - x_k\|^2 / 2\sigma_i^2\right)},$$

Similarity of map point yi to yj

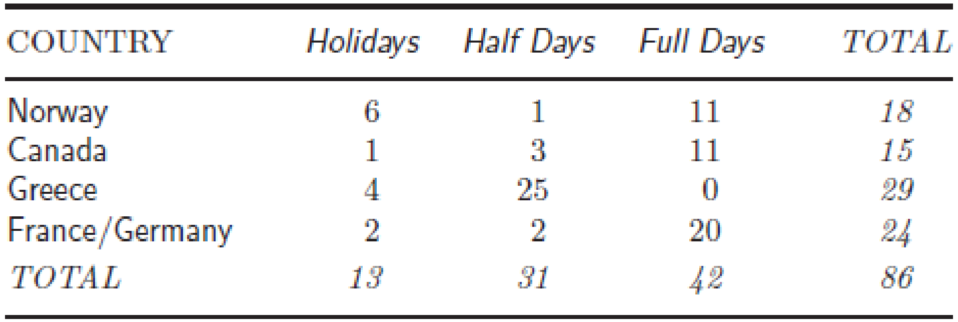
$$q_{j|i} = \frac{\exp\left(-\|y_i - y_j\|^2\right)}{\sum_{k\neq i} \exp\left(-\|y_i - y_k\|^2\right)}.$$



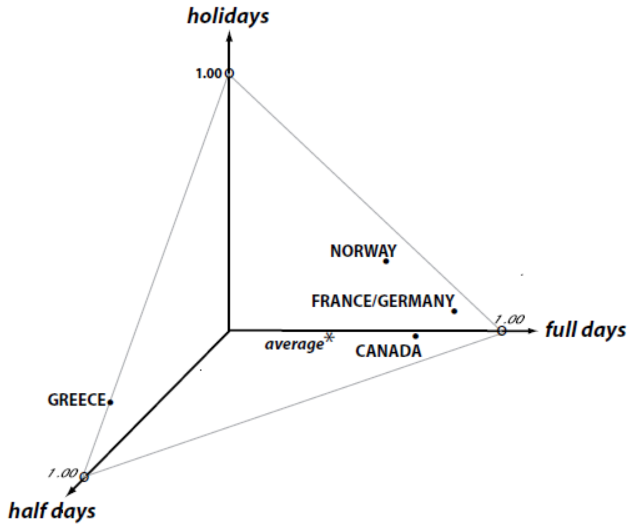
(a) Visualization by t-SNE.

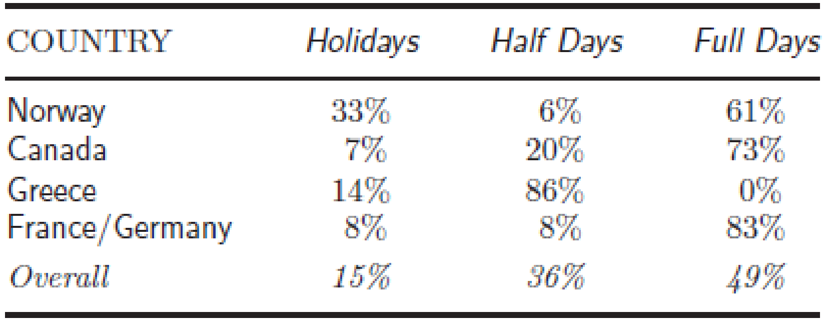
(b) Visualization by Sammon mapping.

Reduction of dimensionality when the data add to a constant. Correspondence analysis

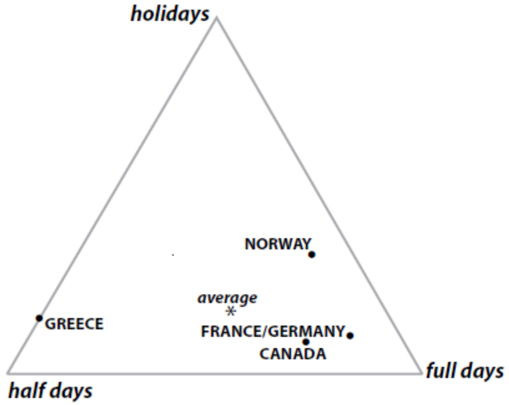


COUNTRY	*Holidays*	*Half Days*	*Full Days*	*TOTAL*
Norway	6	1	11	*18*
Canada	1	3	11	*15*
Greece	4	25	0	*29*
France/Germany	2	2	20	*24*
TOTAL	*13*	*31*	*42*	*86*



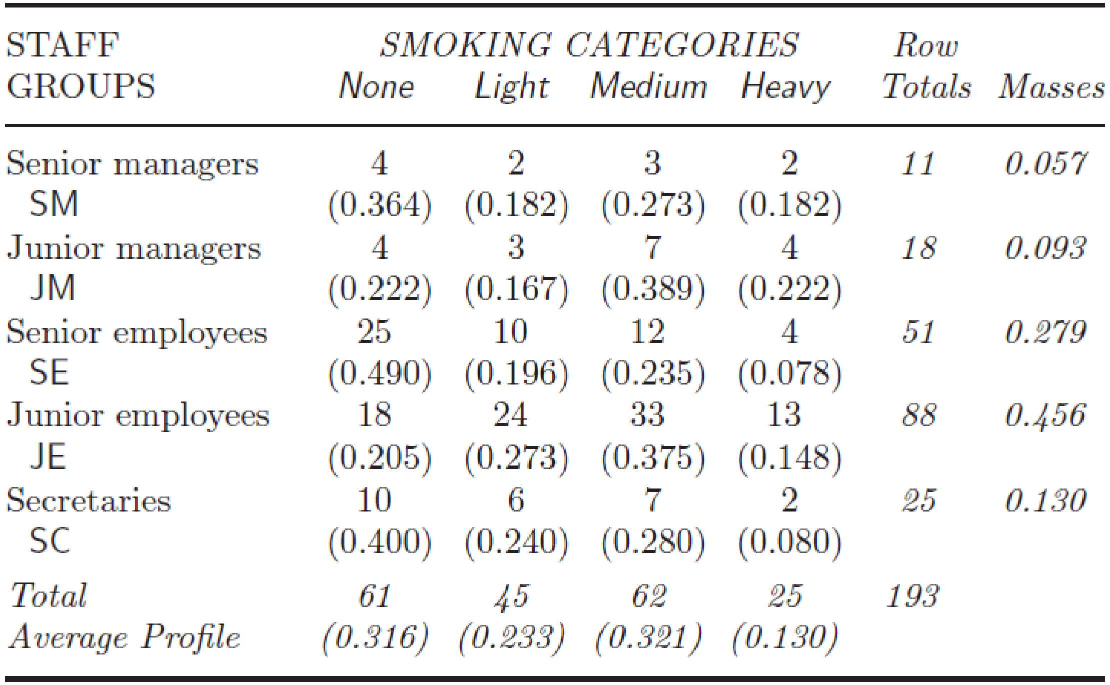


COUNTRY	*Holidays*	*Half Days*	*Full Days*
Norway	33%	6%	61%
Canada	7%	20%	73%
Greece	14%	86%	0%
France/Germany	8%	8%	83%
Overall	*15%*	*36%*	*49%*

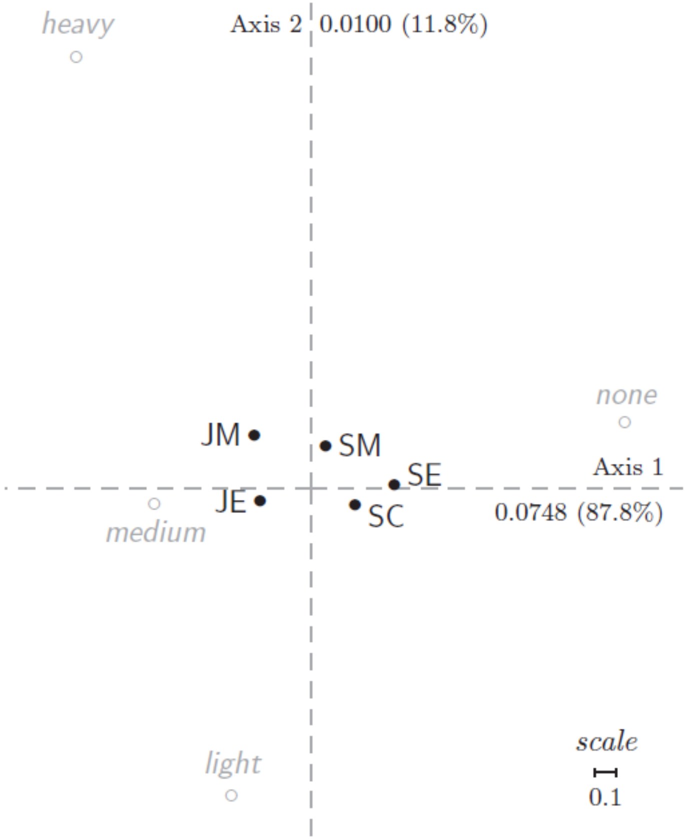


Reduction of dimensionality when the data add to a constant. Correspondence analysis

Based on Chi-Square statistic

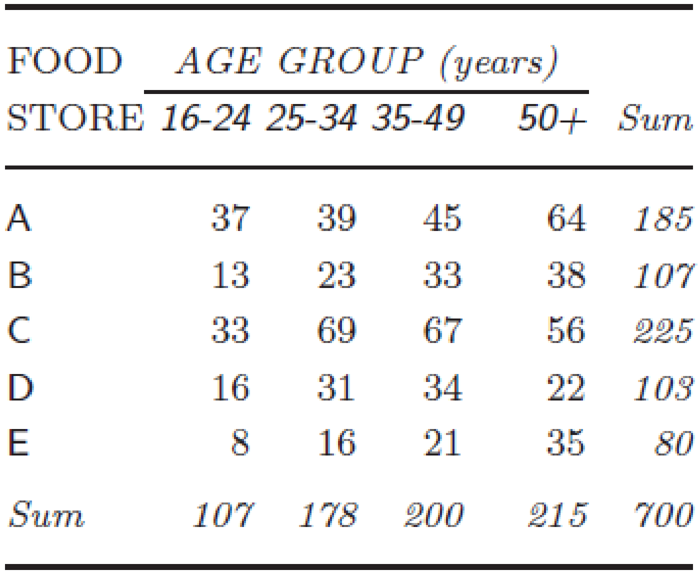


STAFF GROUPS	*SMOKING CATEGORIES* None	Light	Medium	Heavy	*Row Totals*	*Masses*
Senior managers SM	4 (0.364)	2 (0.182)	3 (0.273)	2 (0.182)	*11*	*0.057*
Junior managers JM	4 (0.222)	3 (0.167)	7 (0.389)	4 (0.222)	*18*	*0.093*
Senior employees SE	25 (0.490)	10 (0.196)	12 (0.235)	4 (0.078)	*51*	*0.279*
Junior employees JE	18 (0.205)	24 (0.273)	33 (0.375)	13 (0.148)	*88*	*0.456*
Secretaries SC	10 (0.400)	6 (0.240)	7 (0.280)	2 (0.080)	*25*	*0.130*
Total *Average Profile*	*61* *(0.316)*	*45* *(0.233)*	*62* *(0.321)*	*25* *(0.130)*	*193*	

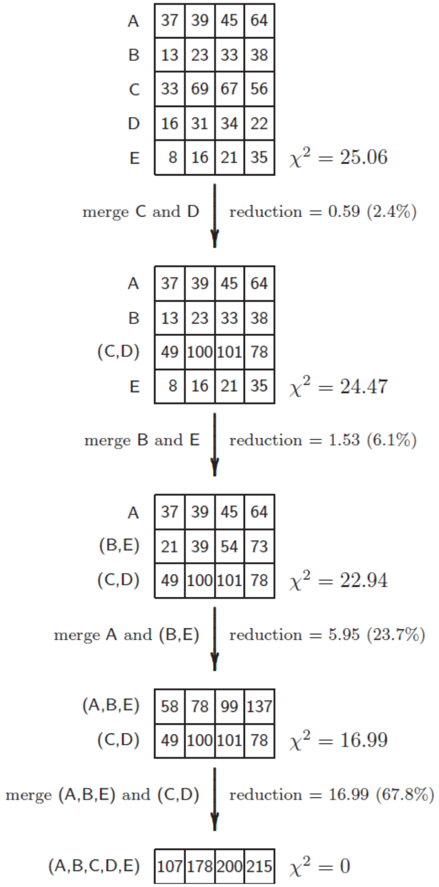


Reduction of dimensionality when the data add to a constant. Correspondence analysis

Based on Chi-Square statistic



FOOD STORE	*AGE GROUP (years)* 16-24	25-34	35-49	50+	*Sum*
A	37	39	45	64	*185*
B	13	23	33	38	*107*
C	33	69	67	56	*225*
D	16	31	34	22	*103*
E	8	16	21	35	*80*
Sum	*107*	*178*	*200*	*215*	*700*



A	37	39	45	64
B	13	23	33	38
C	33	69	67	56
D	16	31	34	22
E	8	16	21	35

$\chi^2 = 25.06$

merge C and D — reduction = 0.59 (2.4%)

A	37	39	45	64
B	13	23	33	38
(C,D)	49	100	101	78
E	8	16	21	35

$\chi^2 = 24.47$

merge B and E — reduction = 1.53 (6.1%)

A	37	39	45	64
(B,E)	21	39	54	73
(C,D)	49	100	101	78

$\chi^2 = 22.94$

merge A and (B,E) — reduction = 5.95 (23.7%)

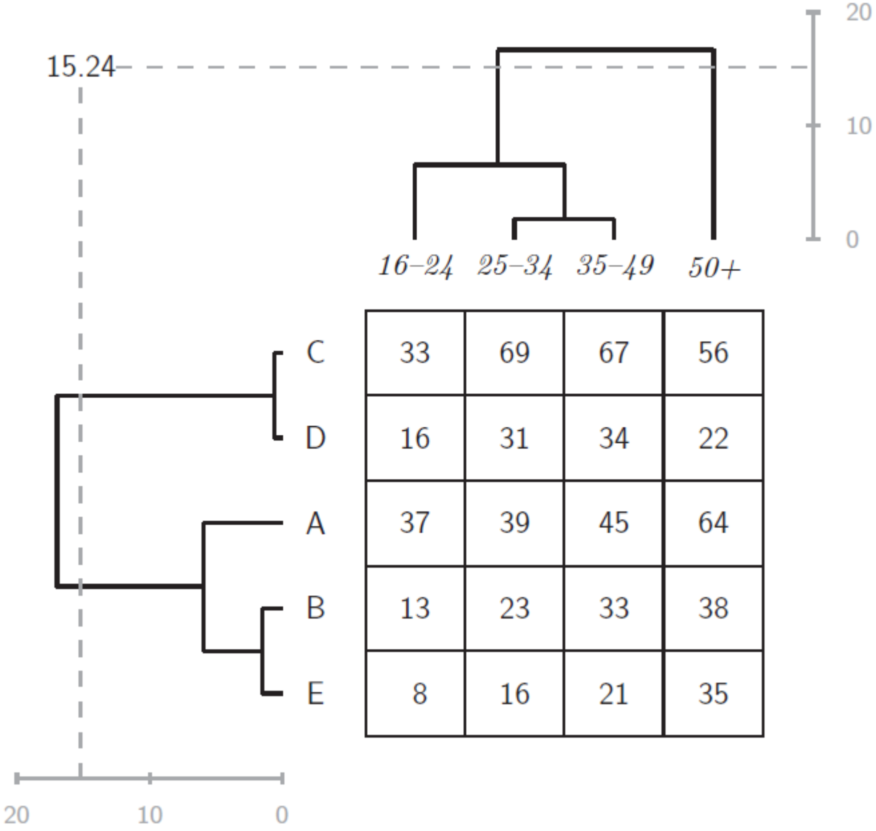
(A,B,E)	58	78	99	137
(C,D)	49	100	101	78

$\chi^2 = 16.99$

merge (A,B,E) and (C,D) — reduction = 16.99 (67.8%)

(A,B,C,D,E)	107	178	200	215

$\chi^2 = 0$



	16–24	25–34	35–49	50+
C	33	69	67	56
D	16	31	34	22
A	37	39	45	64
B	13	23	33	38
E	8	16	21	35

Which things are important in clustering sequences?