Data Mining Techniques

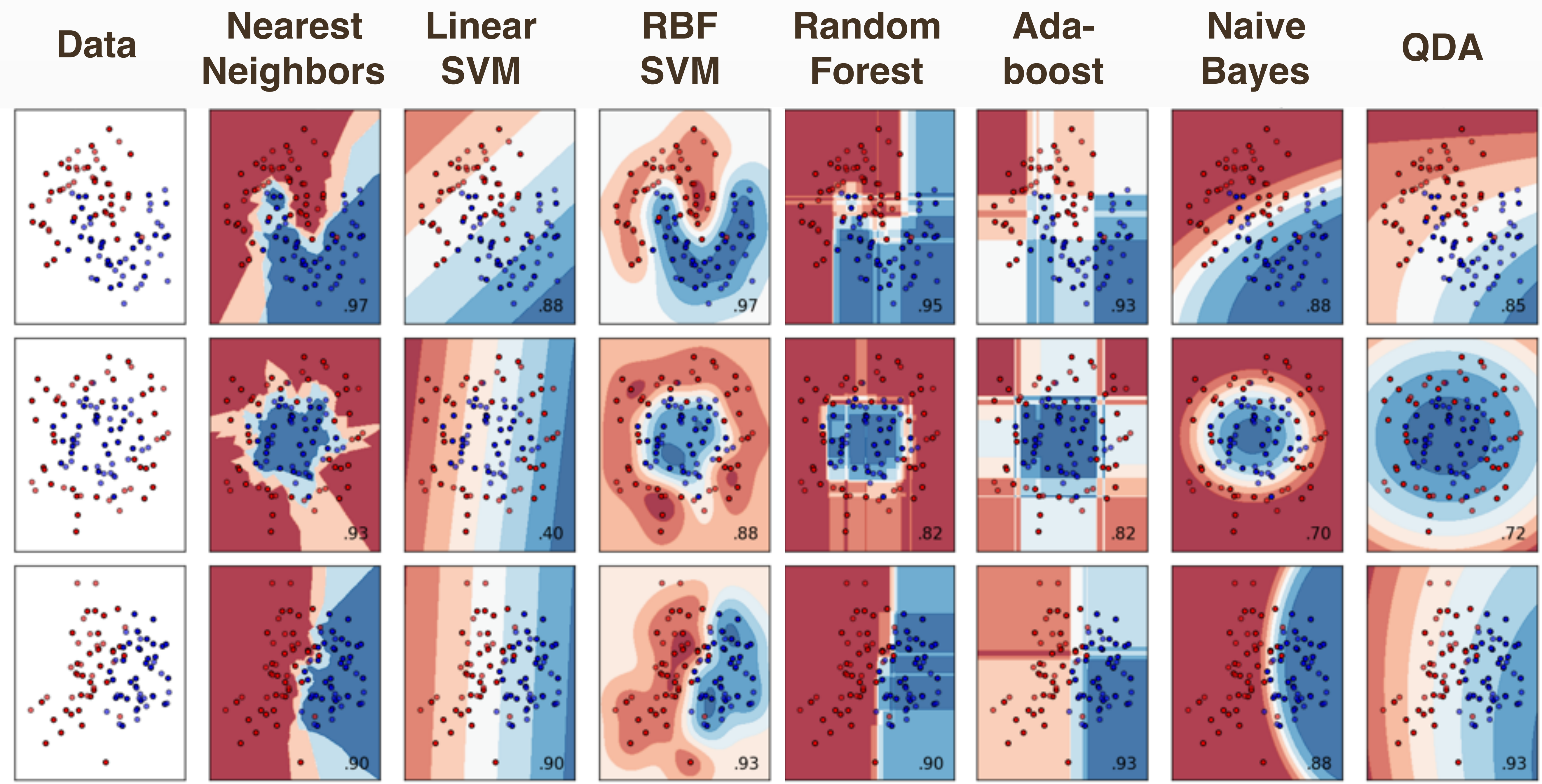
CS 6220 - Section 3 - Fall 2016

Lecture 8

Jan-Willem van de Meent
(*credit*: Yijun Zhao, Carla Brodley, Eamonn Keogh)

Classification Wrap-up

Classifier Comparison



Confusion Matrix

	Truth	
Prediction	`email`	`spam`
`email`	57.3%	4.0%
`spam`	5.3%	33.4%

Confusion Matrix

Prediction	Truth: email	Truth: spam
email	True Pos 57.3%	False Pos 4.0%
spam	False Neg 5.3%	True Neg 33.4%

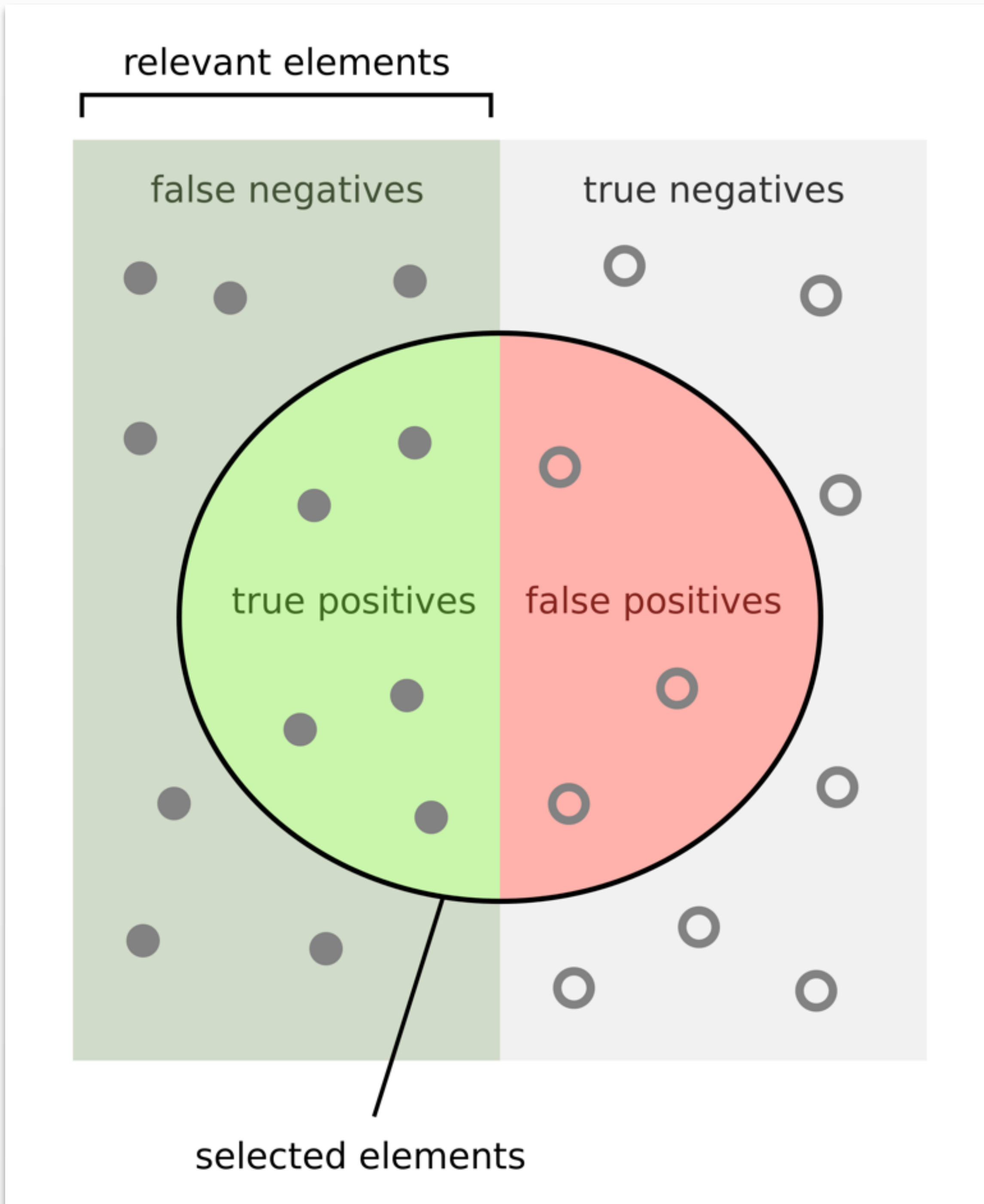
True Positive (TP): Hit (show e-mail)
True Negative (TN): Correct rejection
False Positive (FP): False alarm, type I error
False Negative (FN): Miss, type II error

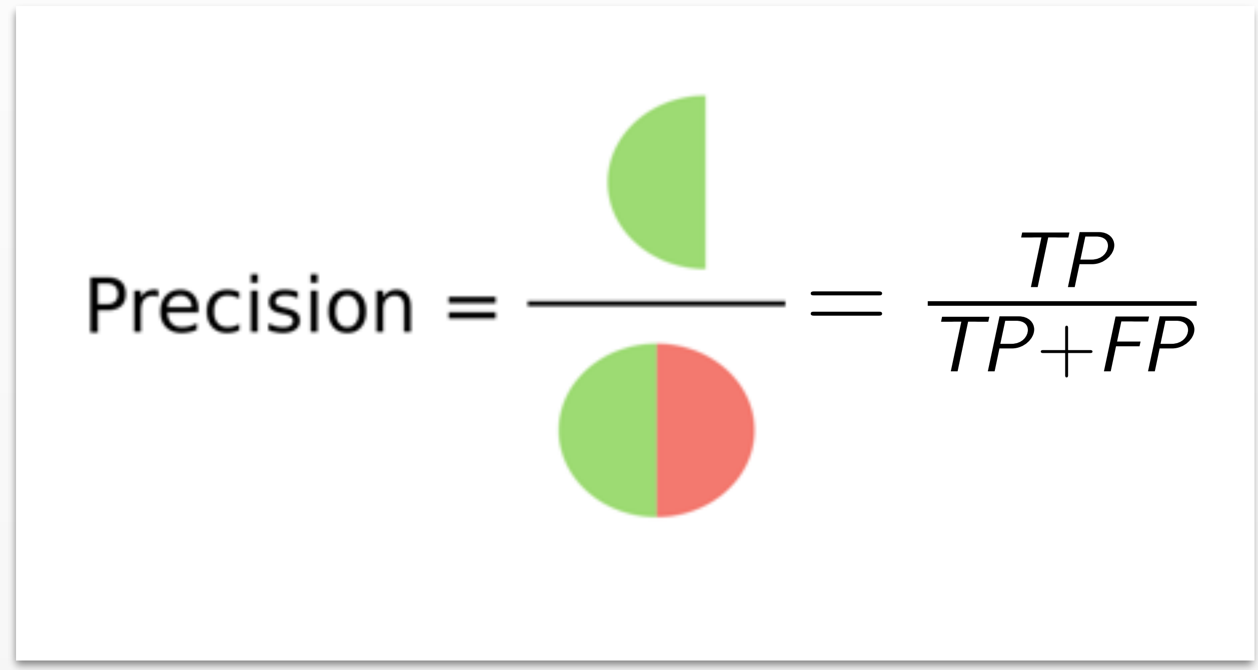
Decision Theory

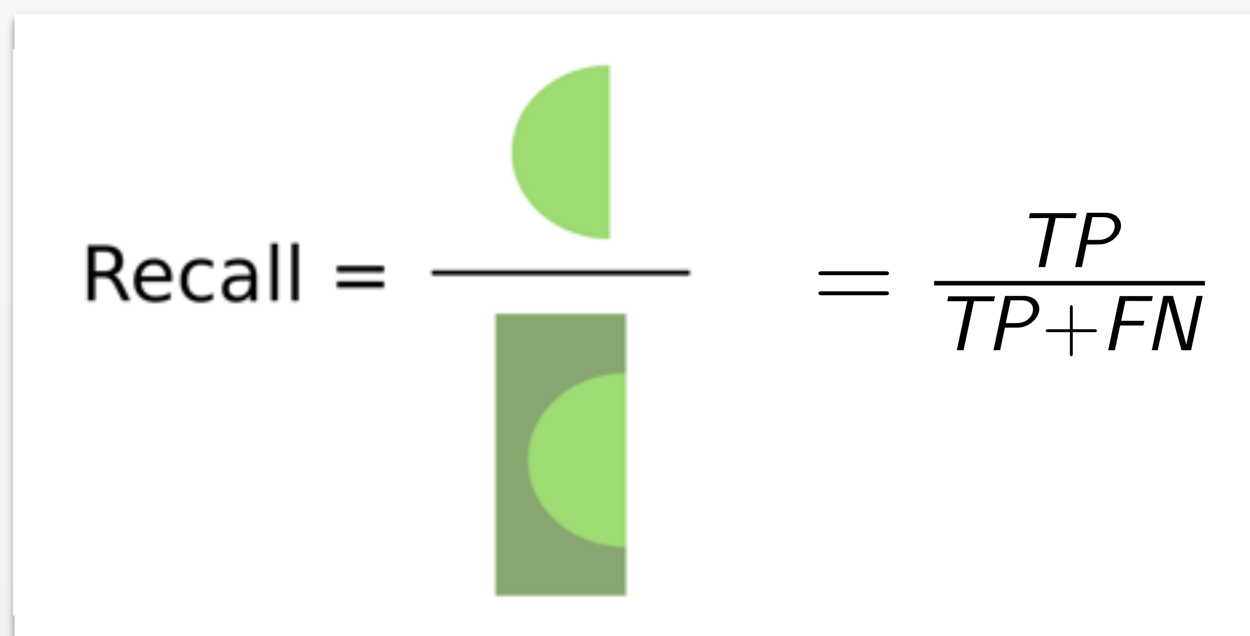
	Predicted	
True	email	spam
email	λ_{11}	λ_{12}
spam	λ_{21}	λ_{22}

$$
\begin{aligned}
R(\alpha_2|\mathbf{x}) &> R(\alpha_1|\mathbf{x}) \\
\lambda_{21}p(Y=1|\mathbf{x}) + \lambda_{22}p(Y=2|\mathbf{x}) &> \lambda_{11}p(Y=1|\mathbf{x}) + \lambda_{12}p(Y=2|\mathbf{x}) \\
(\lambda_{21}-\lambda_{11})p(Y=1|\mathbf{x}) &> (\lambda_{12}-\lambda_{22})p(Y=2|\mathbf{x}) \\
\frac{p(Y=1|\mathbf{x})}{p(Y=2|\mathbf{x})} &> \frac{\lambda_{12}-\lambda_{22}}{\lambda_{21}-\lambda_{11}}
\end{aligned}
$$

Precision and Recall







Precision and Recall

Precision or Positive Predictive Value (PPV)

$$PPV = \frac{TP}{TP+FP}$$

Recall or Sensitivity, True Positive Rate (TPR)

$$TPR = \frac{TP}{TP+FN}$$

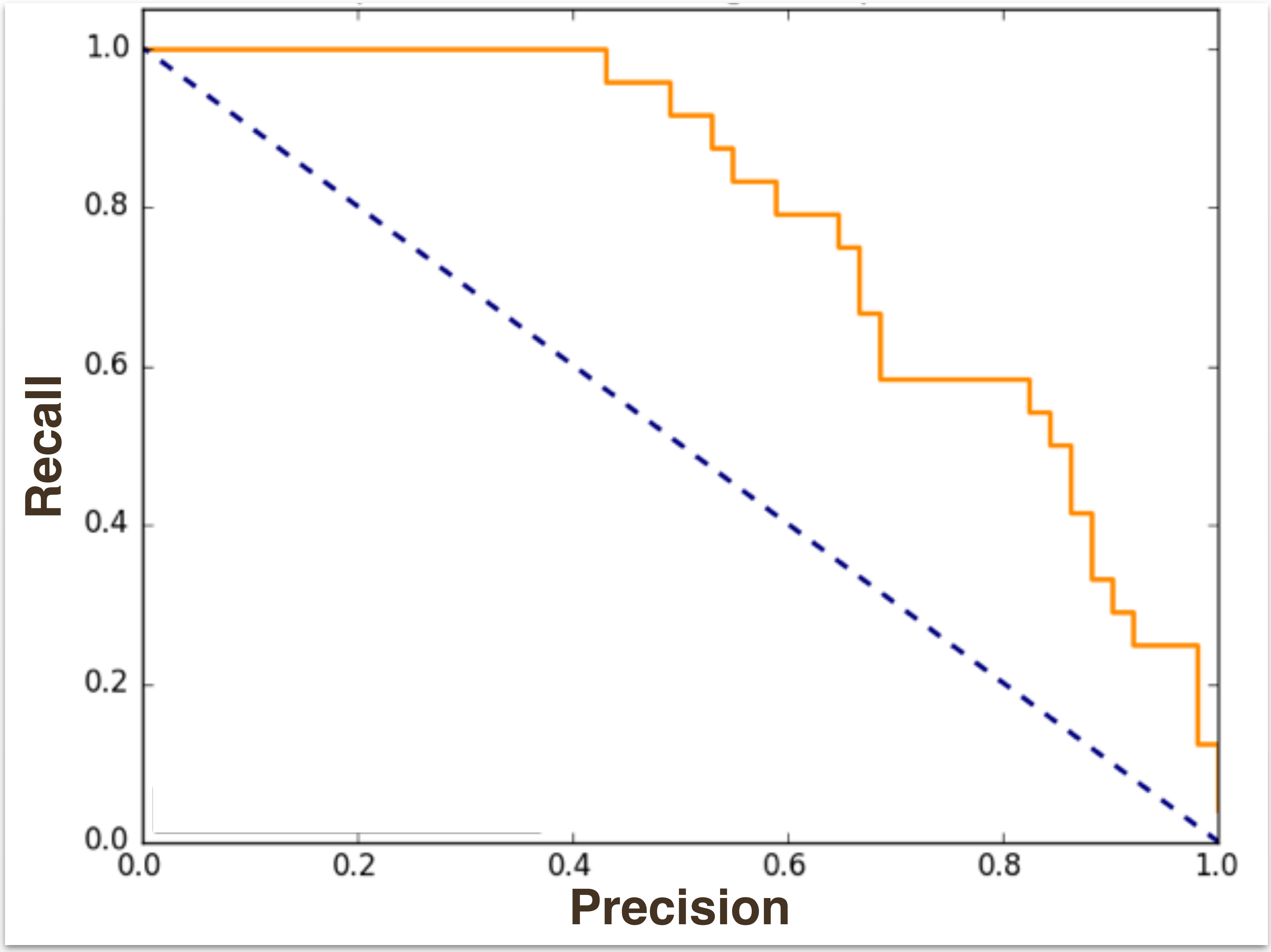
F1 score: harmonic mean of Precisin and Recall

$$F1 = \frac{2TP}{(2TP+FP+FN)}$$

Specificity (SPC) or True Negative Rate (TNR)

$$SPC = \frac{TN}{(FP+TN)}$$

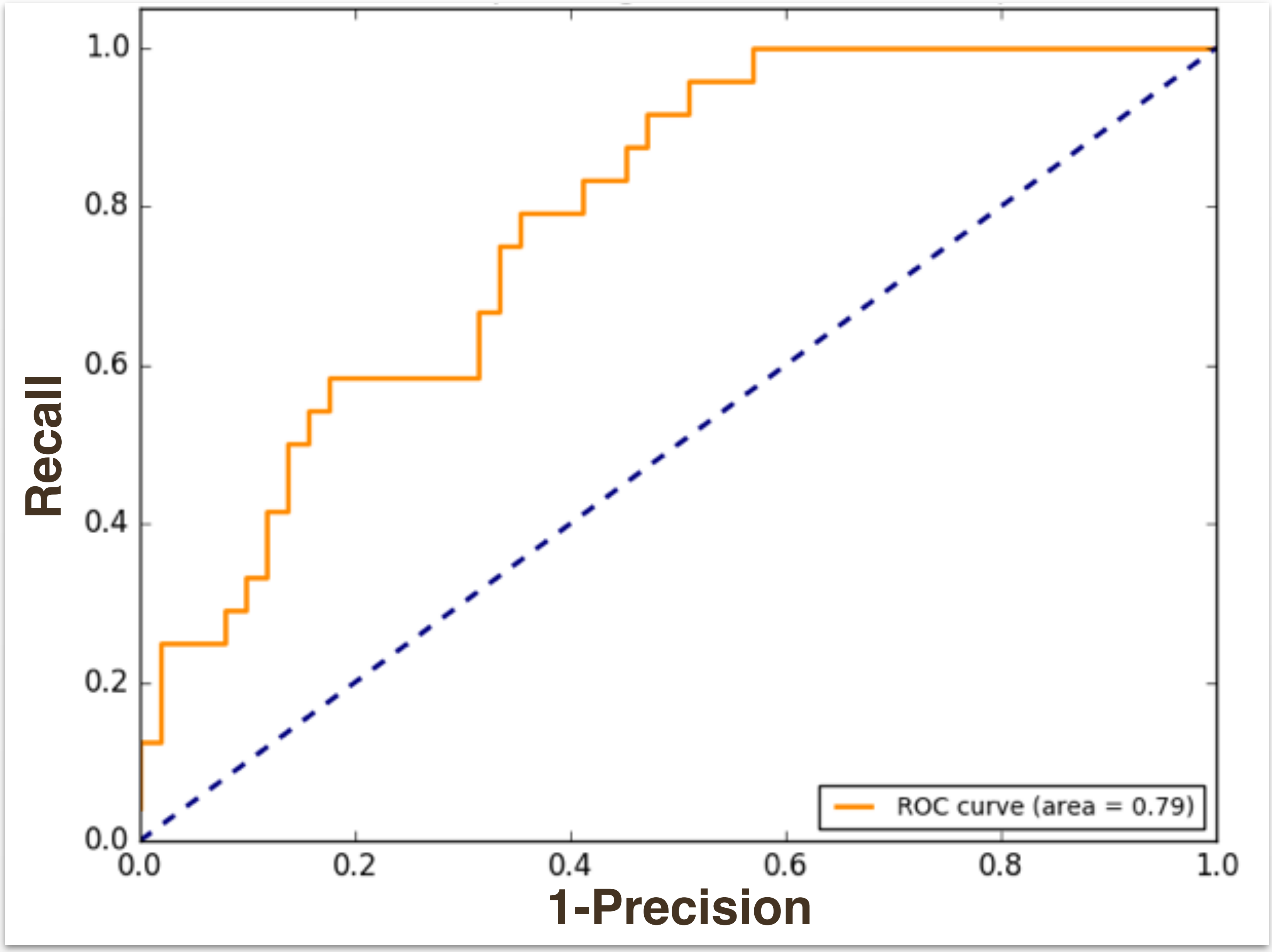
Precision-Recall Curve



Vary detection threshold

$$\frac{p(Y=1|\mathbf{x})}{p(Y=2|\mathbf{x})} > \frac{\lambda_{12}-\lambda_{22}}{\lambda_{21}-\lambda_{11}}$$

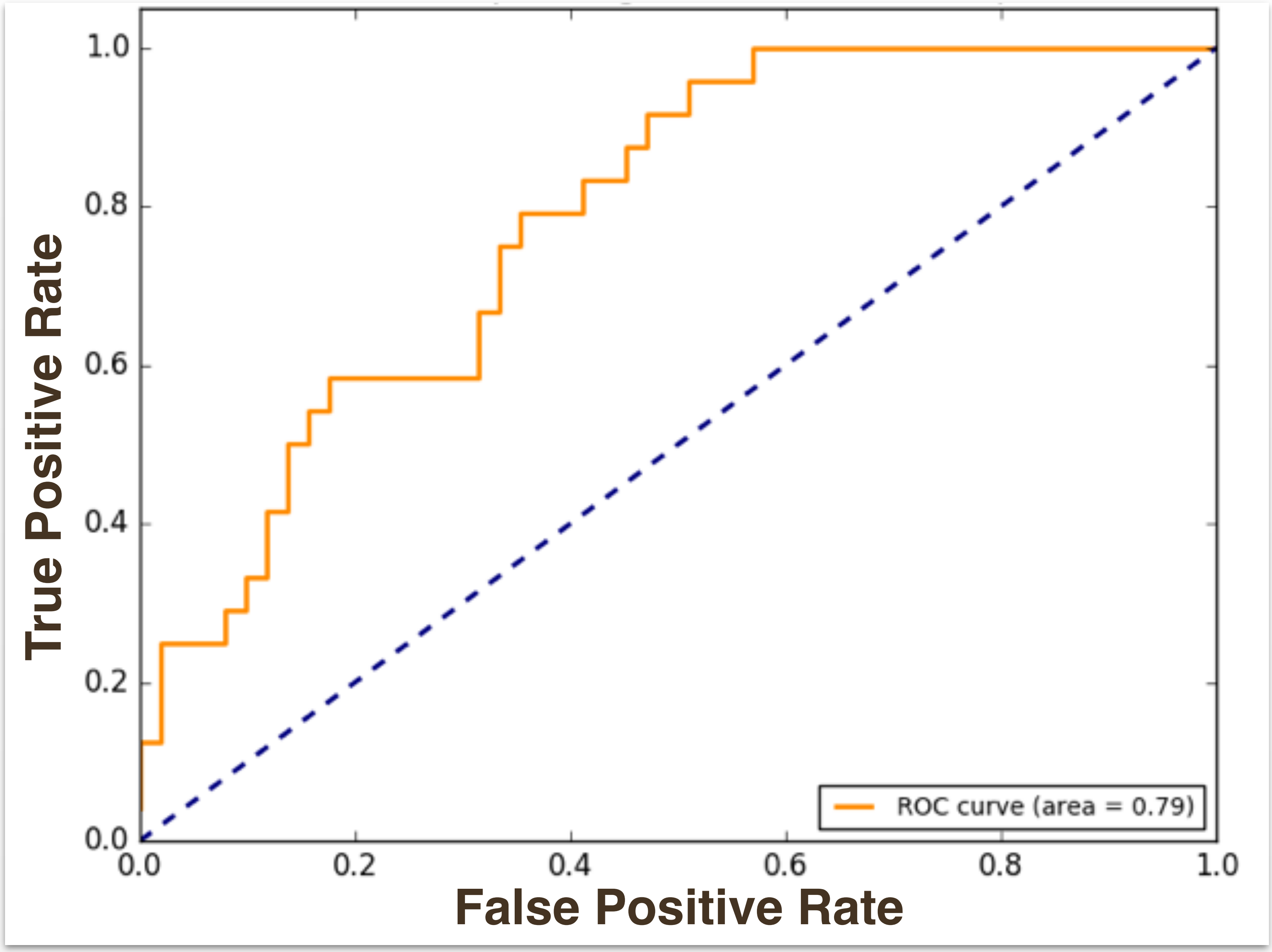
ROC Curve



Vary detection threshold

$$\frac{p(Y=1|\mathbf{x})}{p(Y=2|\mathbf{x})} > \frac{\lambda_{12}-\lambda_{22}}{\lambda_{21}-\lambda_{11}}$$

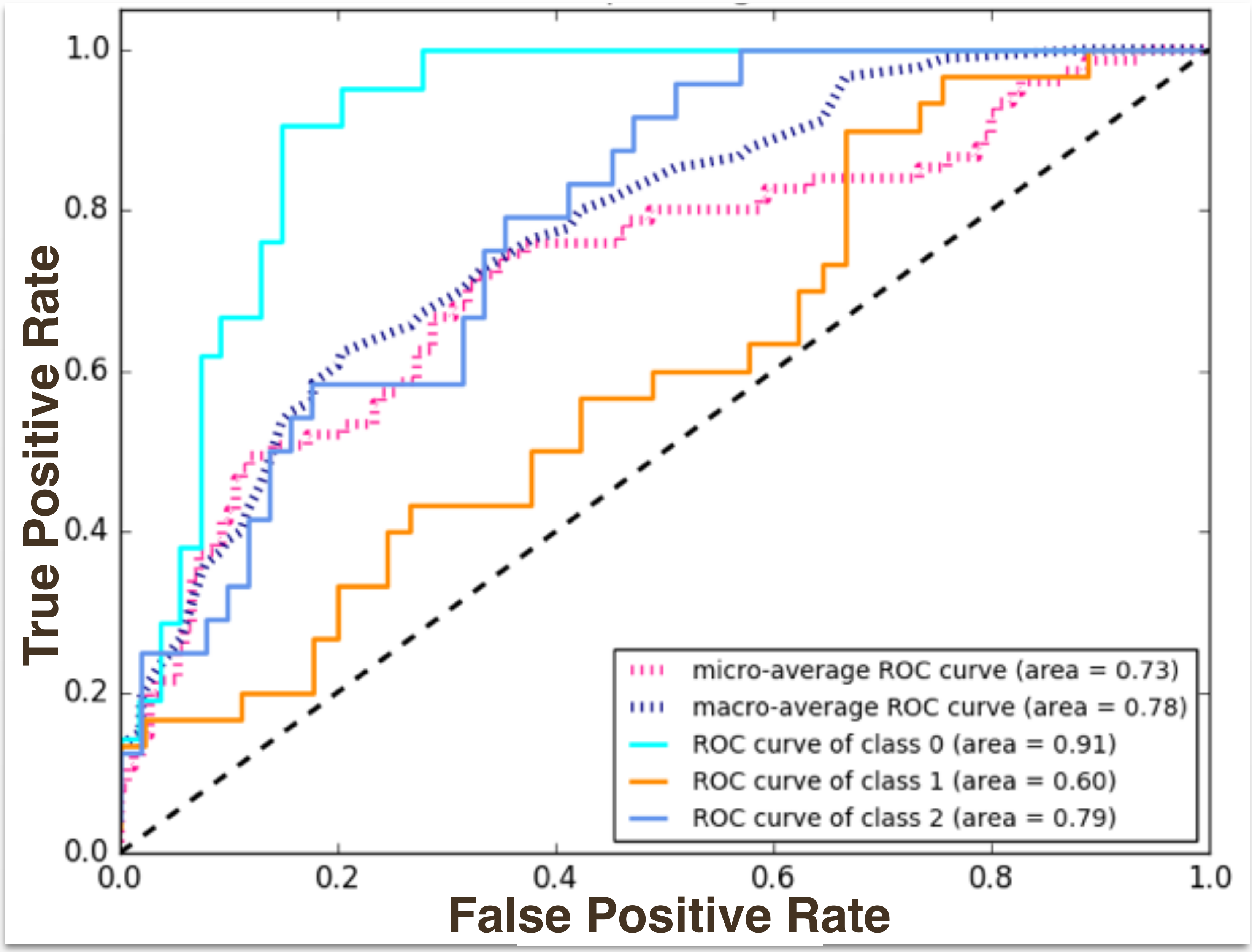
ROC Curve



Vary detection threshold

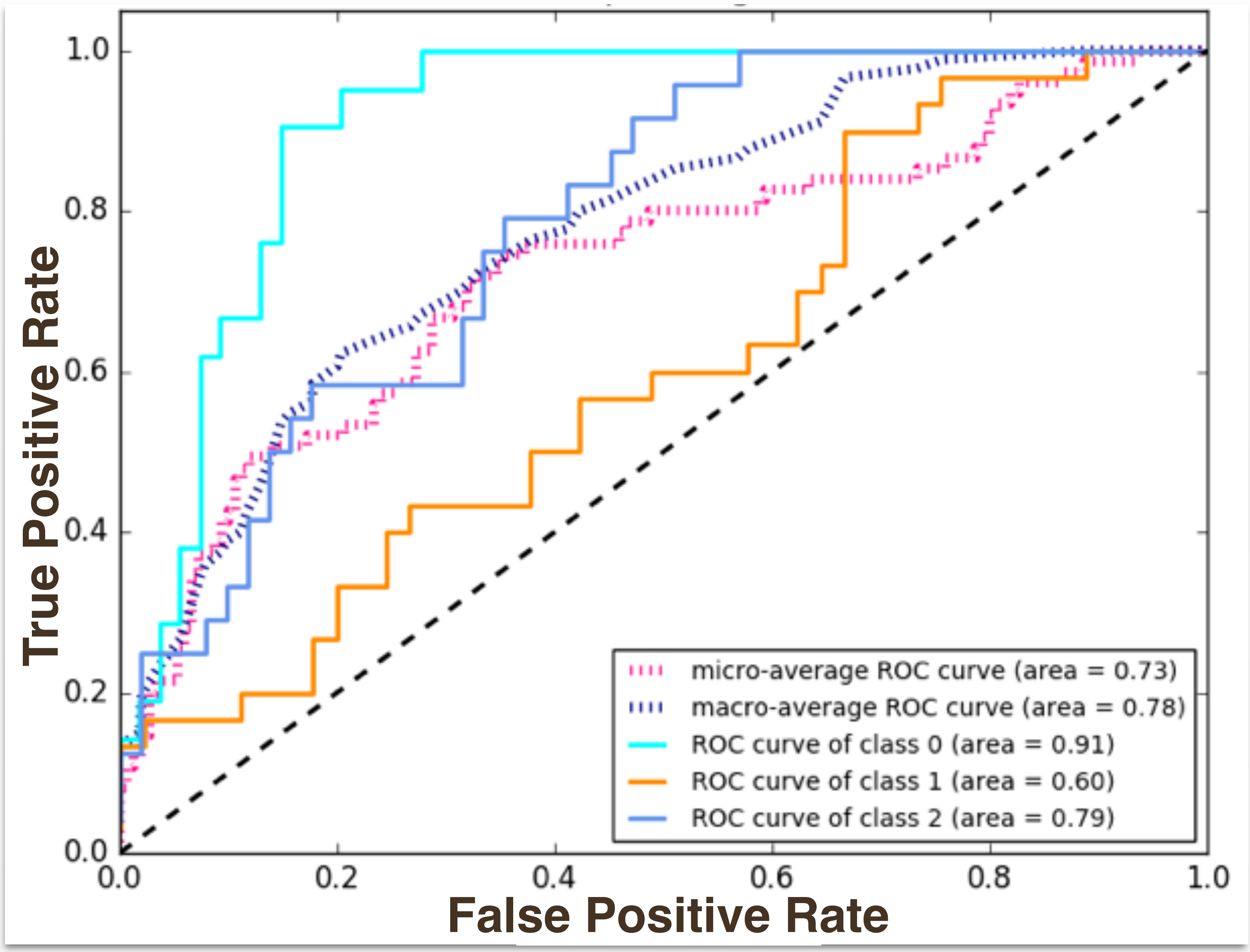
$$\frac{p(Y=1|\mathbf{x})}{p(Y=2|\mathbf{x})} > \frac{\lambda_{12}-\lambda_{22}}{\lambda_{21}-\lambda_{11}}$$

ROC Curve



True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
micro-average ROC curve (area = 0.73)
macro-average ROC curve (area = 0.78)
ROC curve of class 0 (area = 0.91)
ROC curve of class 1 (area = 0.60)
ROC curve of class 2 (area = 0.79)

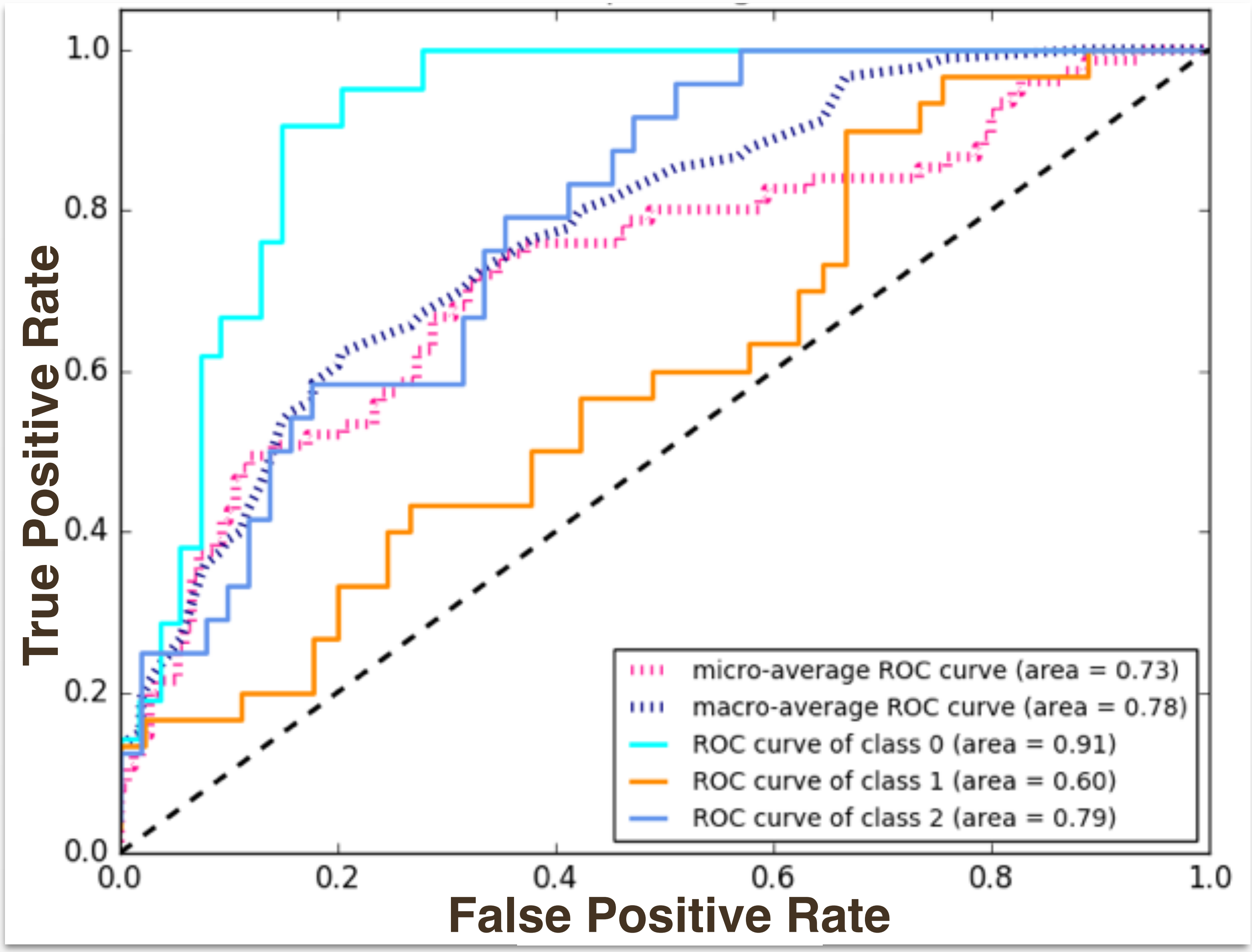
ROC Curve



Macro-average
(True Positive Rate)

$$\frac{1}{2}\left(\frac{TP_1}{TP_1+FP_1}+\frac{TP_2}{TP_2+FP_2}\right)$$

ROC Curve



Micro-average
(True Positive Rate)

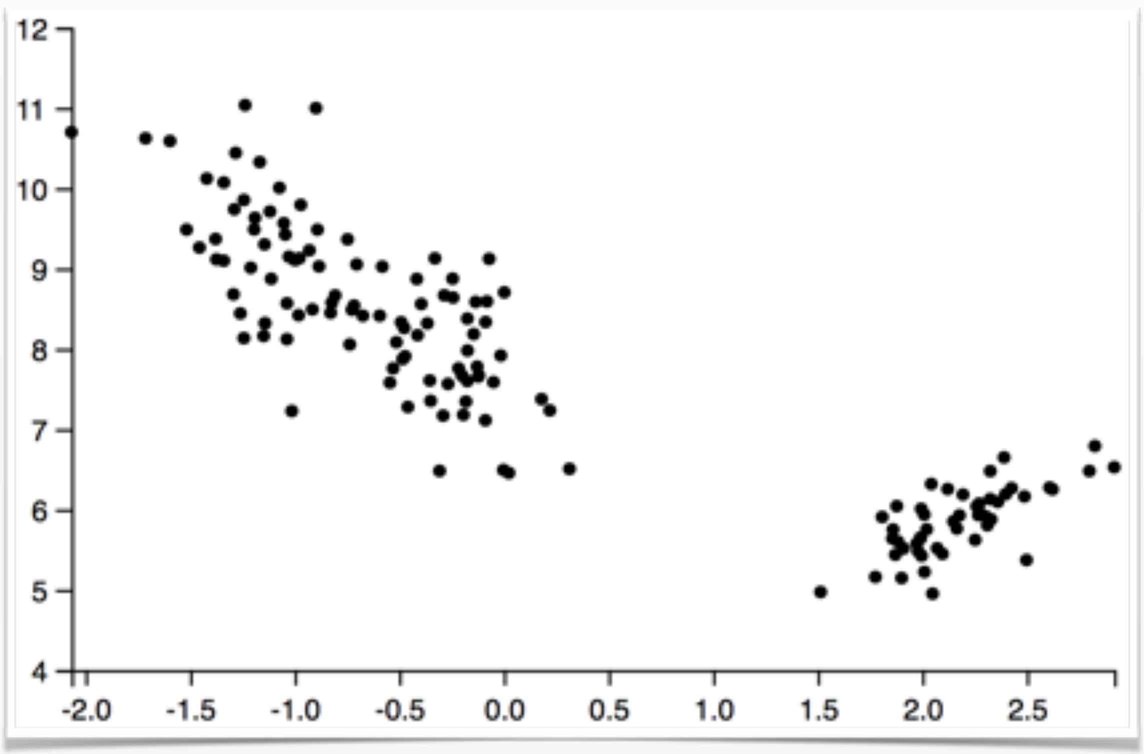
$$\frac{TP_1 + TP_2}{TP_1 + TP_2 + FP_1 + FP_2}$$

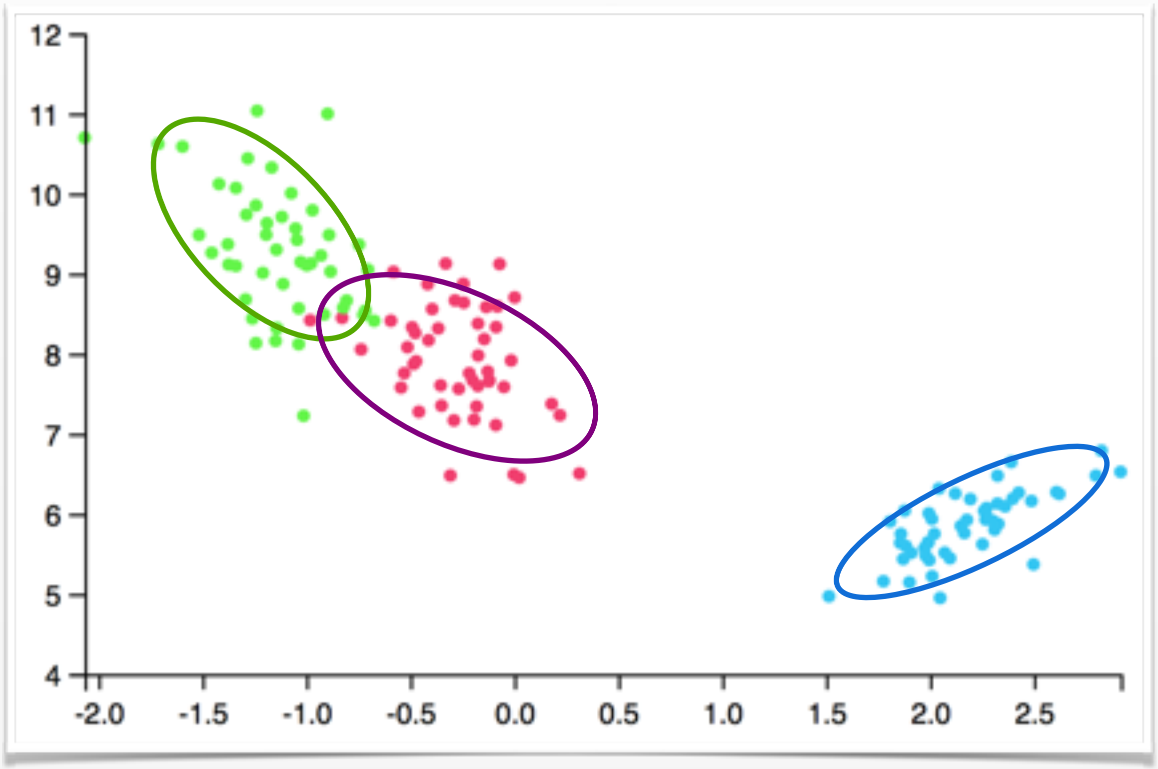
Clustering

(a.k.a. unsupervised classification)

with slides from
Eamonn Keogh
(UC Riverside)

Clustering

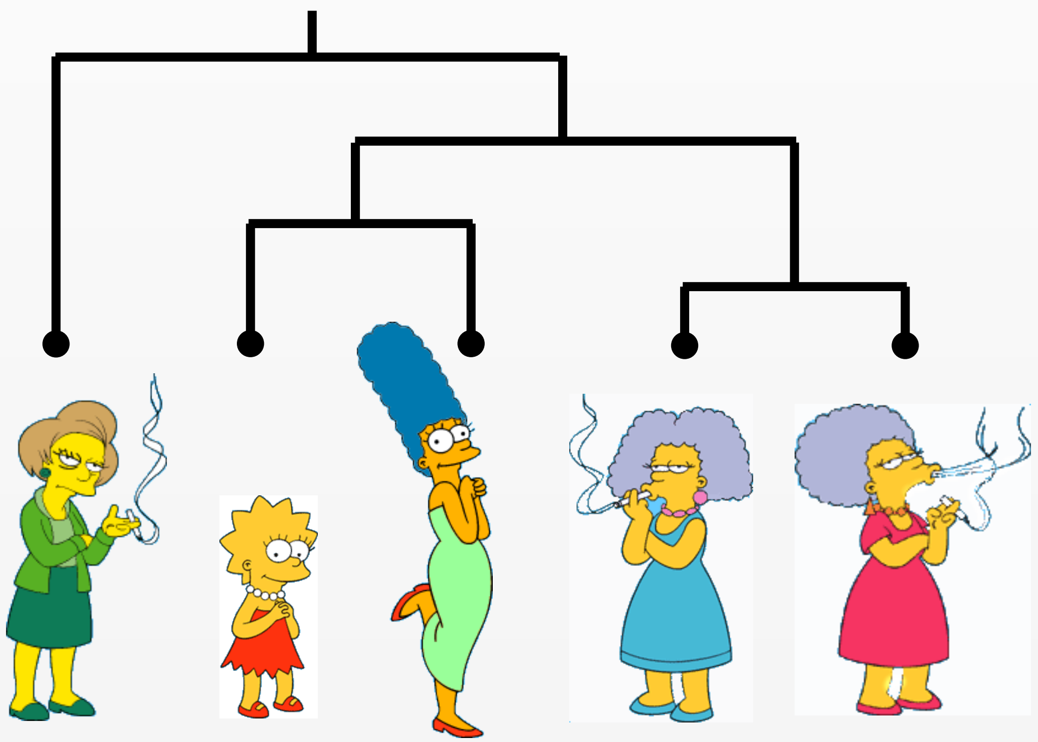




- Unsupervised learning (no labels for training)
- Group data into similar classes that
  - Maximize *inter-cluster* similarity
  - Minimize *intra-cluster* similarity

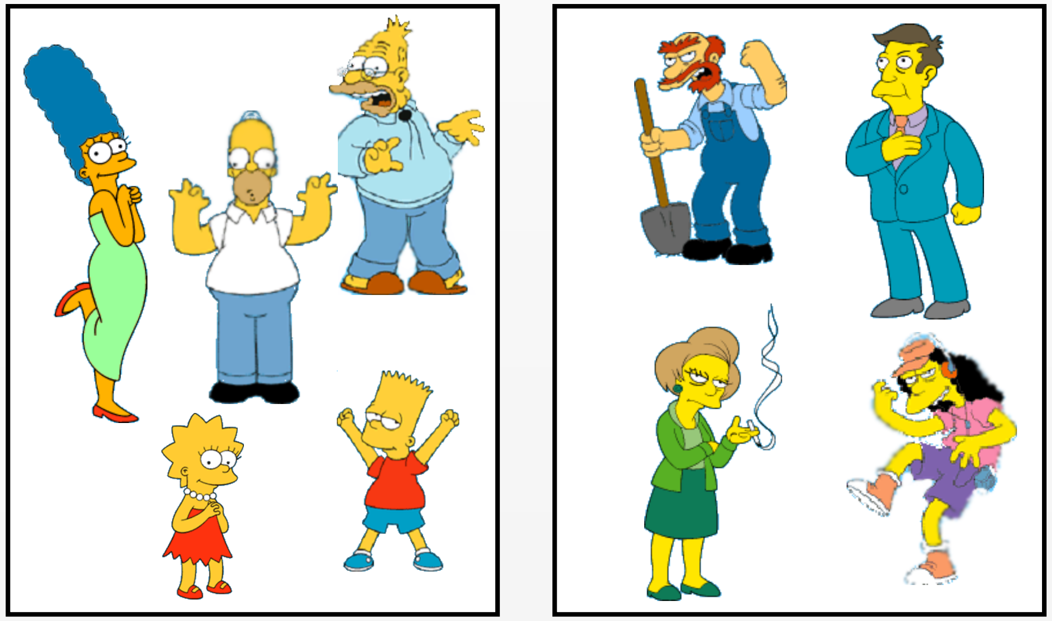
Two Types of Clustering

Hierarchical



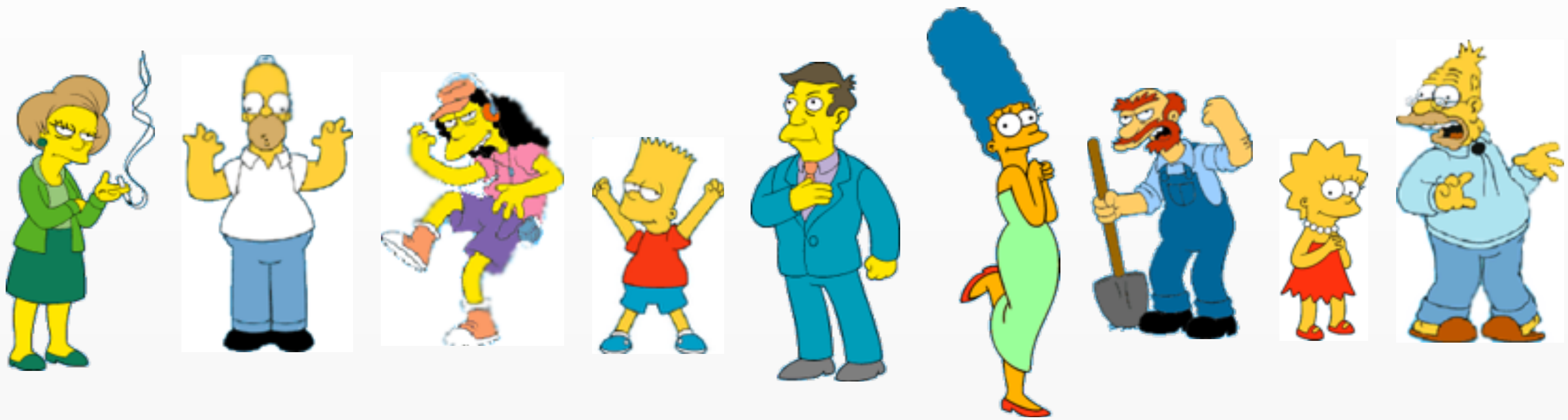
Create a hierarchical decomposition using *"some criterion"*

Partitional

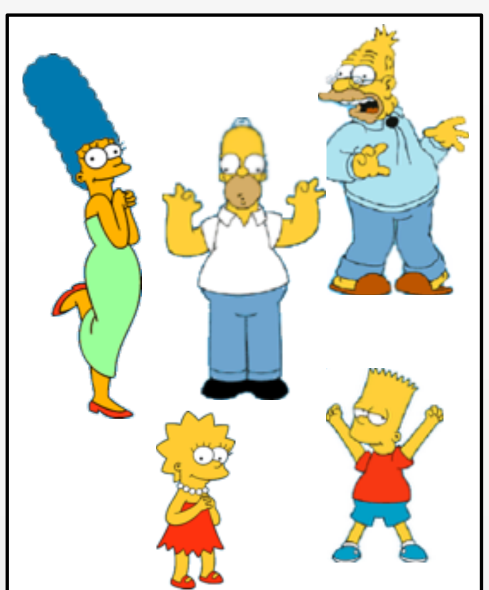


Construct partitions and evaluate them using *"some criterion"*

What is a natural grouping?

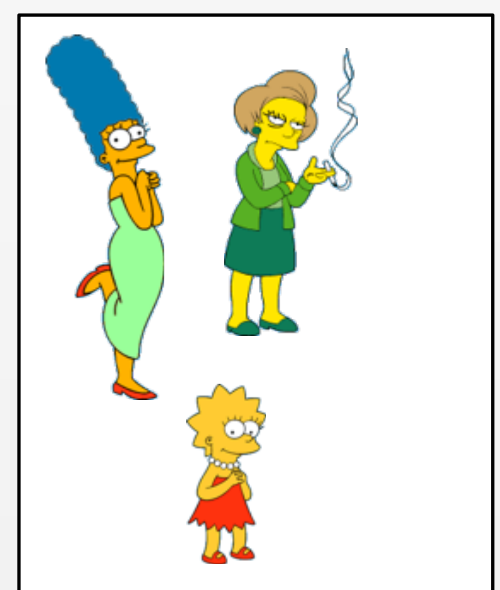


Choice of clustering criterion can be task-dependent

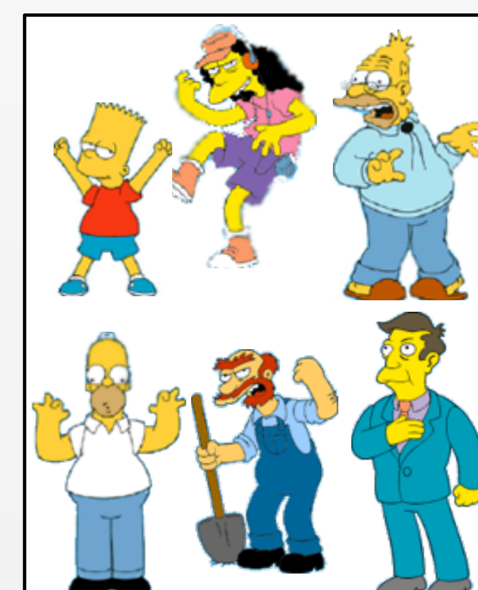


Simpson's Family

School Employees

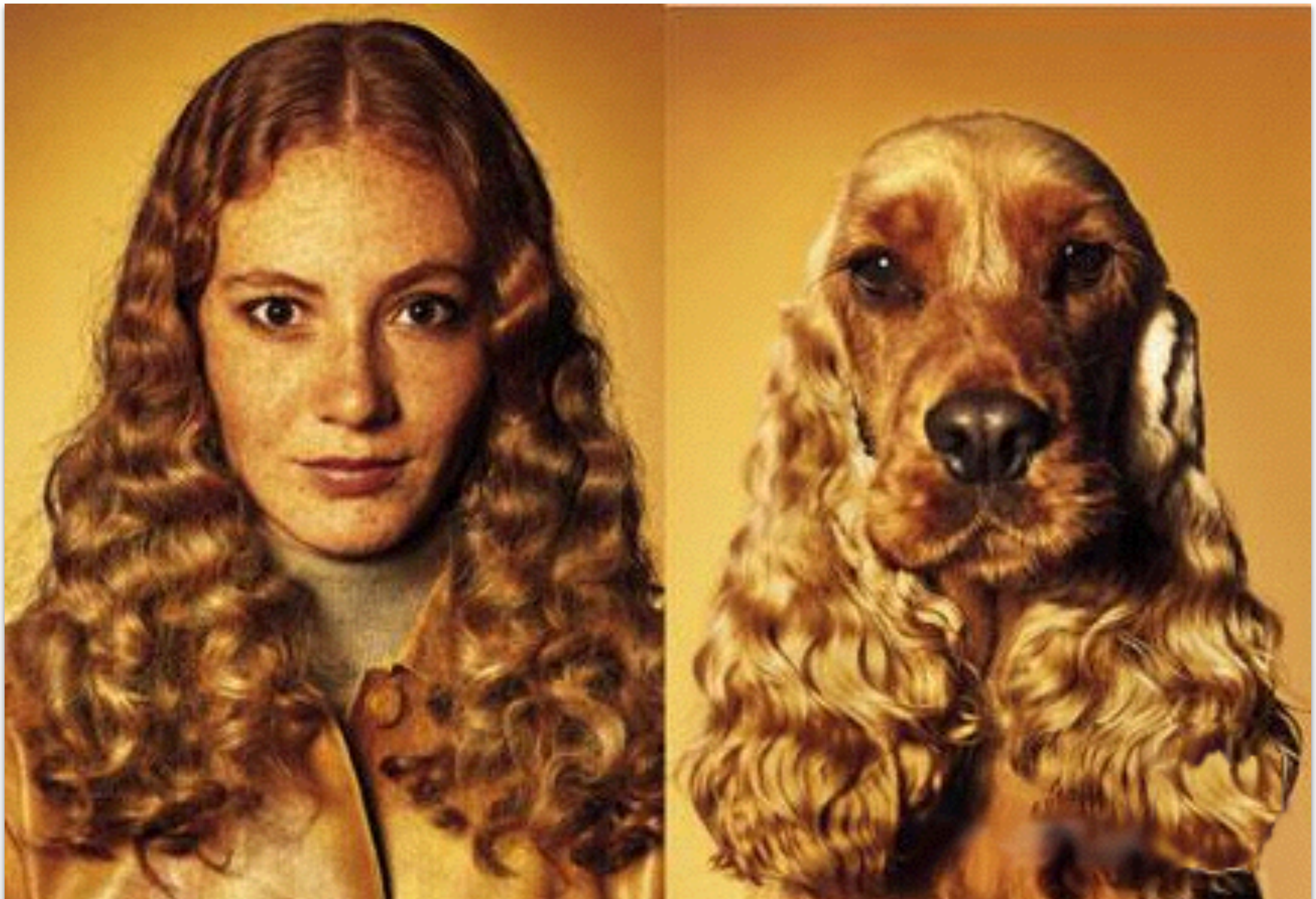


Females



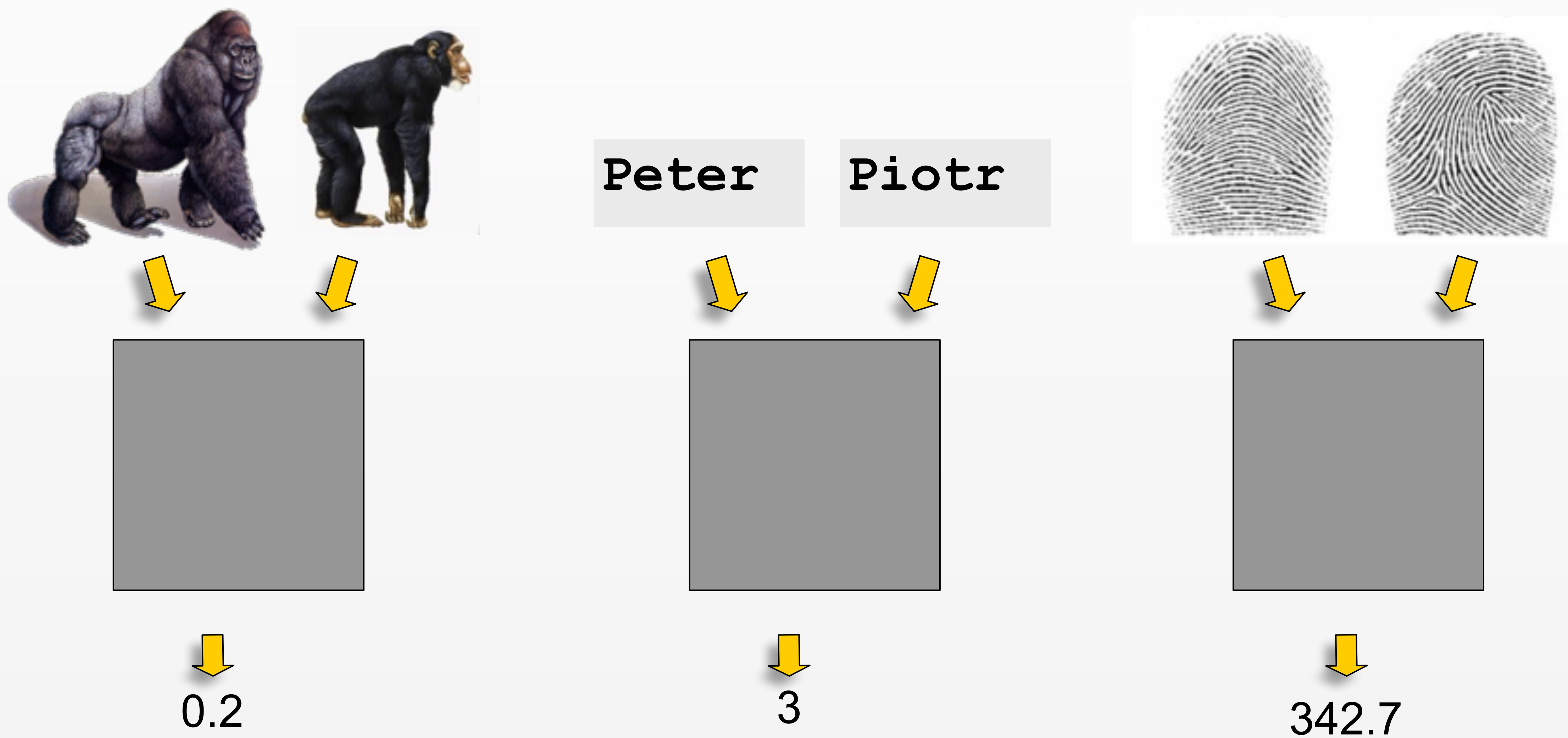
Males

What is Similarity?



Can be hard to define, but we know it when we see it.

Defining Distance Measures



***Need*:** Some function $D(\boldsymbol{x}_1, \boldsymbol{x}_2)$ that represents degree of dissimilarity

Example: Distance Measures

- Euclidean Distance $$\sqrt{(\sum_{i=1}^{k}(x_i - y_i)^2)}$$
- Mahattan Distance $$\sum_{i=1}^{k}|x_i - y_i|$$
- Minkowski Distance $$\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{\frac{1}{q}}$$

Example: Kernels

Polynomial	$k(\boldsymbol{x}, \boldsymbol{x}') = (\langle \boldsymbol{x}, \boldsymbol{x}' \rangle + c)^m$
Radial Basis Function (RBF)	$k(\boldsymbol{x}, \boldsymbol{x}') = \exp^{-\frac{1}{2}\gamma^{-2}\|\|\boldsymbol{x}-\boldsymbol{x}'\|\|^2}$
Squared Exponential (SE)	$k(\boldsymbol{x}, \boldsymbol{x}') = \exp^{-\frac{1}{2}\boldsymbol{x}^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{x}'}$
Automatic Relevance Determination (ARD)	$k(\boldsymbol{x}, \boldsymbol{x}') = \exp^{-\frac{1}{2}\sum_{i=1}^{d}\frac{(x_i-x_i')^2}{\sigma_i^2}}$

Inner Product vs Distance Measure

Inner Product

- $\langle A, B \rangle = \langle B, A \rangle$ — *Symmetry*
- $\langle \alpha A, B \rangle = \alpha \langle A, B \rangle$ — *Linearity*
- $\langle A, A \rangle = 0$, $\langle A, A \rangle = 0$ iff $A = 0$ — *Postive-definiteness*

Distance Measure

- $D(A, B) = D(B, A)$ — *Symmetry*
- $D(A, A) = 0$ — *Constancy of Self-Similarity*
- $D(A, B) = 0$ iff $A = B$ — *Positivity (Separation)*
- $D(A, B) \leq D(A, C) + D(B, C)$ — *Triangular Inequality*

An inner product $\langle A, B \rangle$ induces
a distance measure $D(A, B) = \langle A-B, A-B \rangle^{1/2}$

Inner Product vs Distance Measure

Inner Product

- $\langle A, B \rangle = \langle B, A \rangle$ — *Symmetry*
- $\langle \alpha A, B \rangle = \alpha \langle A, B \rangle$ — *Linearity*
- $\langle A, A \rangle = 0$, $\langle A, A \rangle = 0$ iff $A = 0$ — *Postive-definiteness*

Distance Measure

- $D(A, B) = D(B, A)$ — *Symmetry*
- $D(A, A) = 0$ — *Constancy of Self-Similarity*
- $D(A, B) = 0$ iff $A = B$ — *Positivity (Separation)*
- $D(A, B) \leq D(A, C) + D(B, C)$ — *Triangular Inequality*

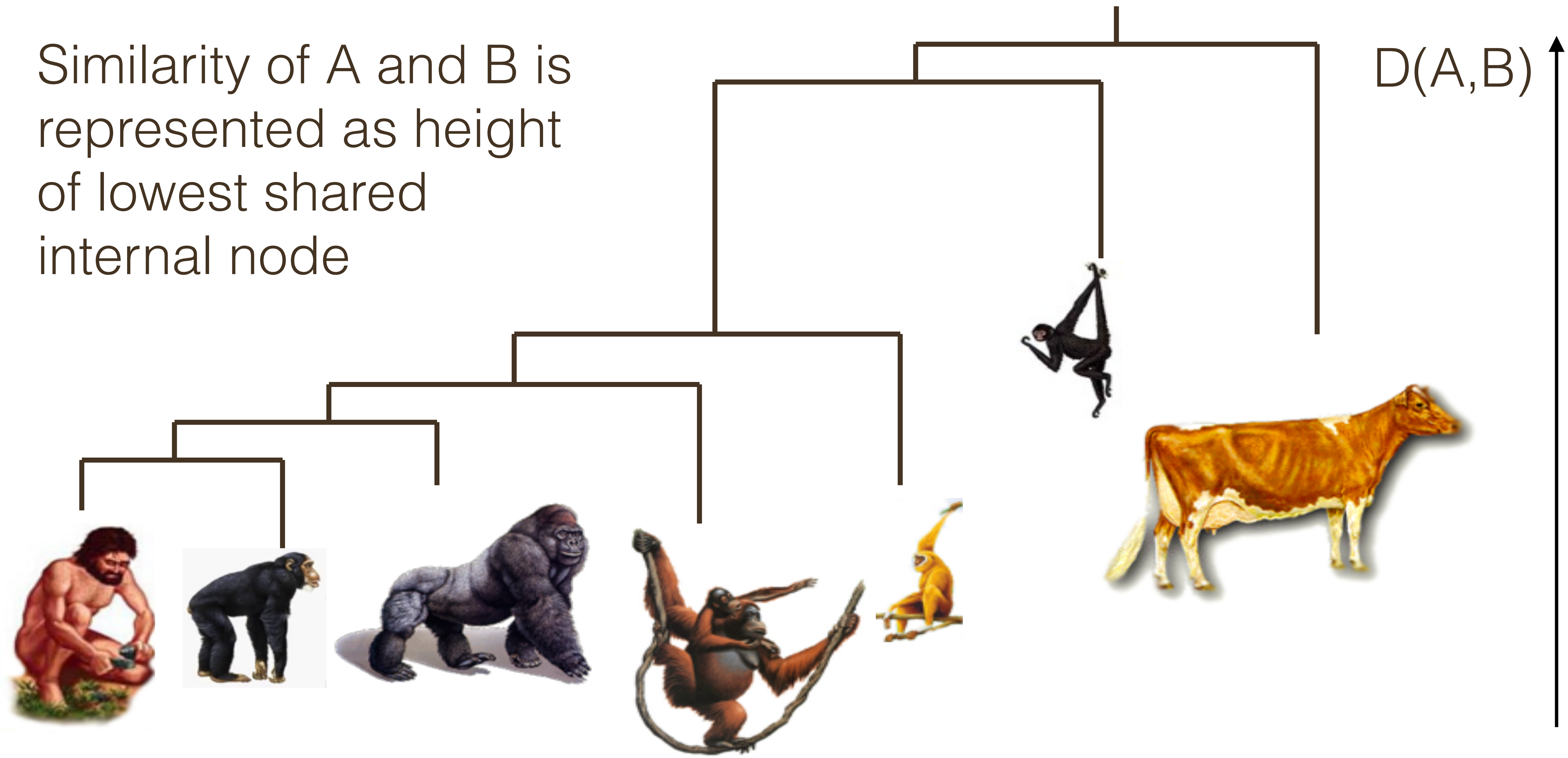
Is the reverse also true?
Why?

Hierarchical Clustering

Dendrogram

(*a.k.a. a similarity tree*)

Similarity of A and B is represented as height of lowest shared internal node

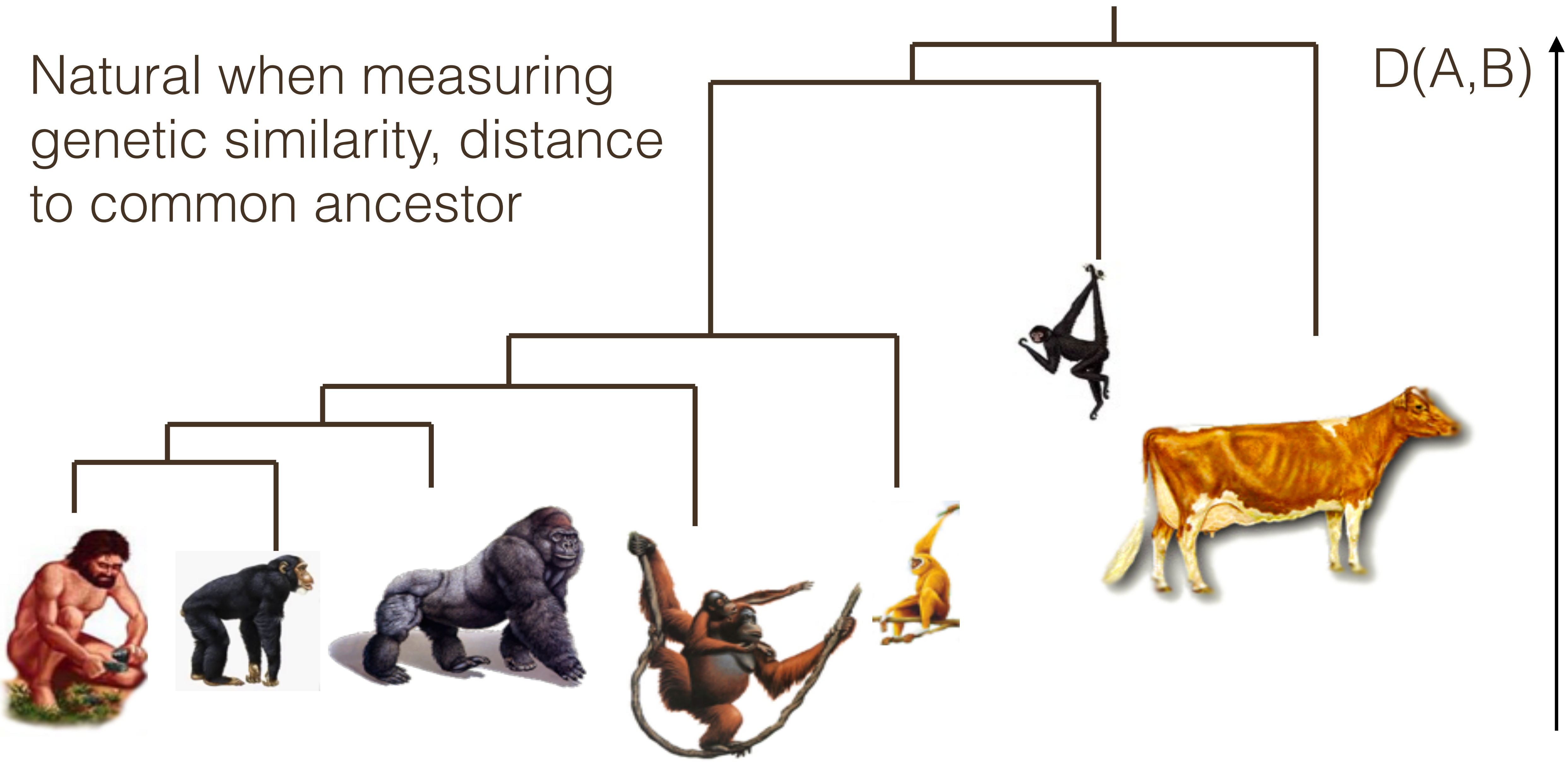


(Bovine: 0.69395, (Spider Monkey: 0.390, (Gibbon:0.36079,(Orang: 0.33636, (Gorilla: 0.17147, (Chimp: 0.19268, Human: 0.11927): 0.08386): 0.06124): 0.15057): 0.54939);

Dendrogram

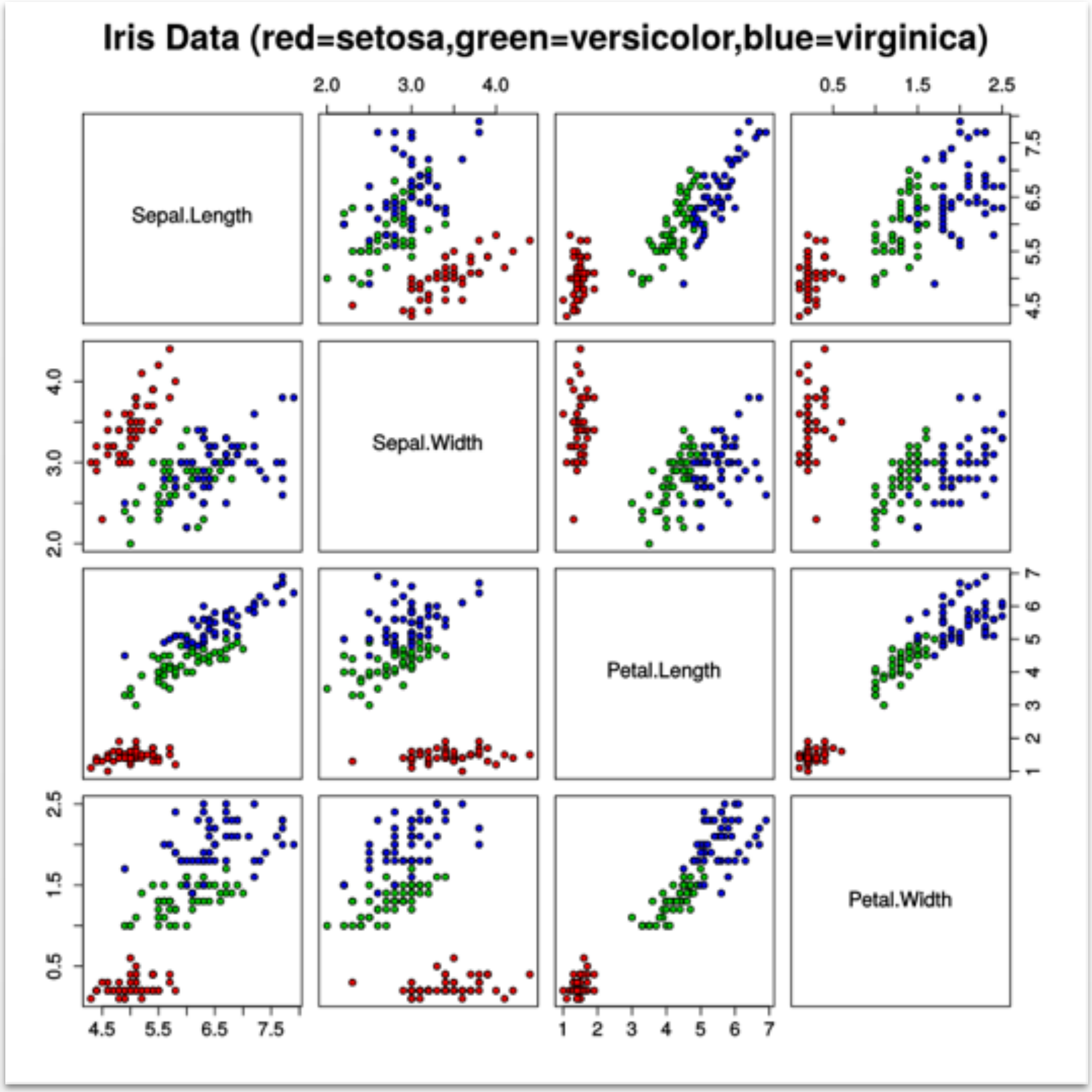
(*a.k.a. a similarity tree*)

Natural when measuring genetic similarity, distance to common ancestor



(Bovine: 0.69395, (Spider Monkey: 0.390, (Gibbon:0.36079,(Orang: 0.33636, (Gorilla: 0.17147, (Chimp: 0.19268, Human: 0.11927): 0.08386): 0.06124): 0.15057): 0.54939);

Example: Iris data



Iris Setosa

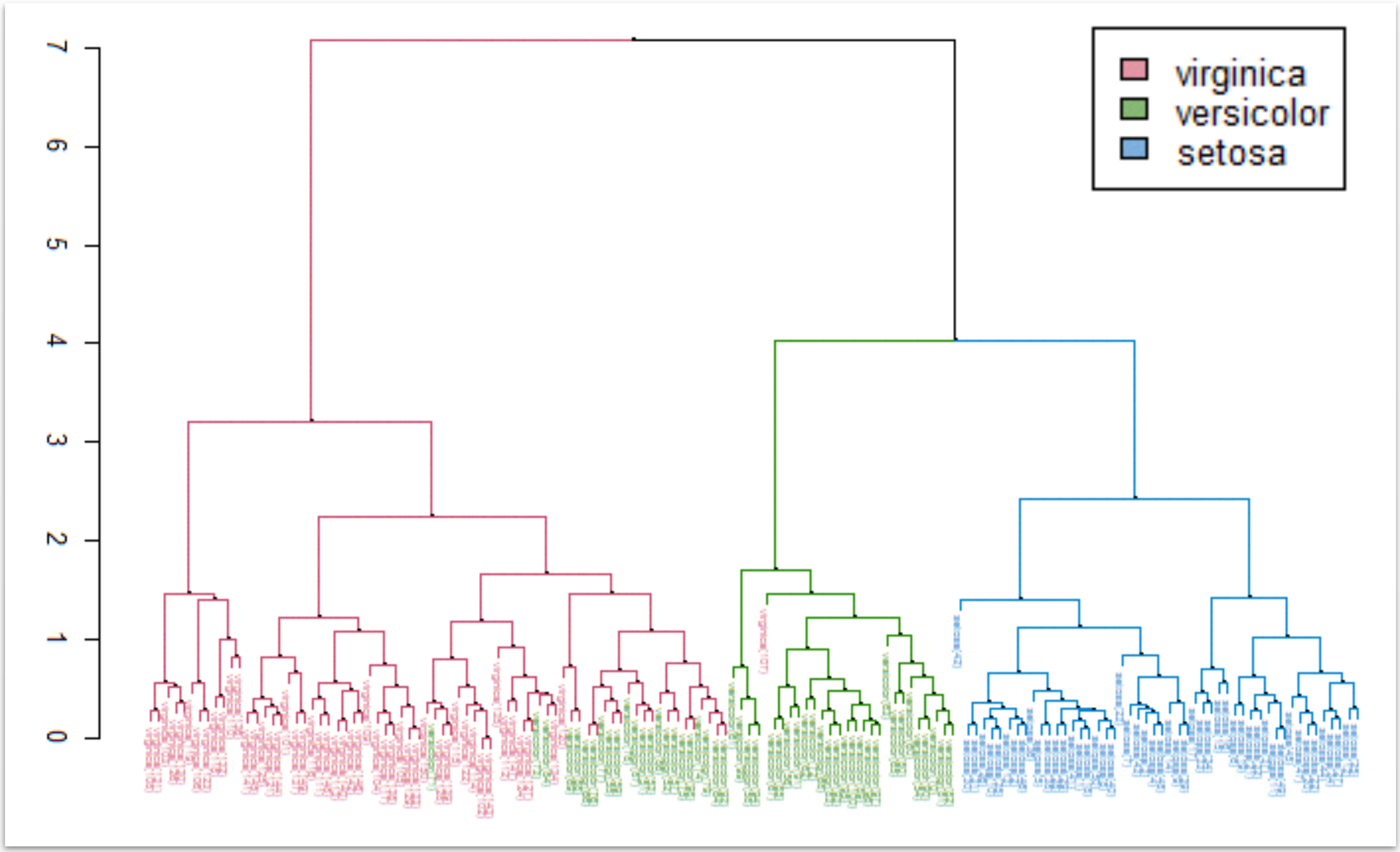
Iris versicolor

Iris virginica

https://en.wikipedia.org/wiki/Iris_flower_data_set

Hierarchical Clustering

(*Euclidian Distance*)



https://en.wikipedia.org/wiki/Iris_flower_data_set

Edit Distance

Distance Patty and Selma

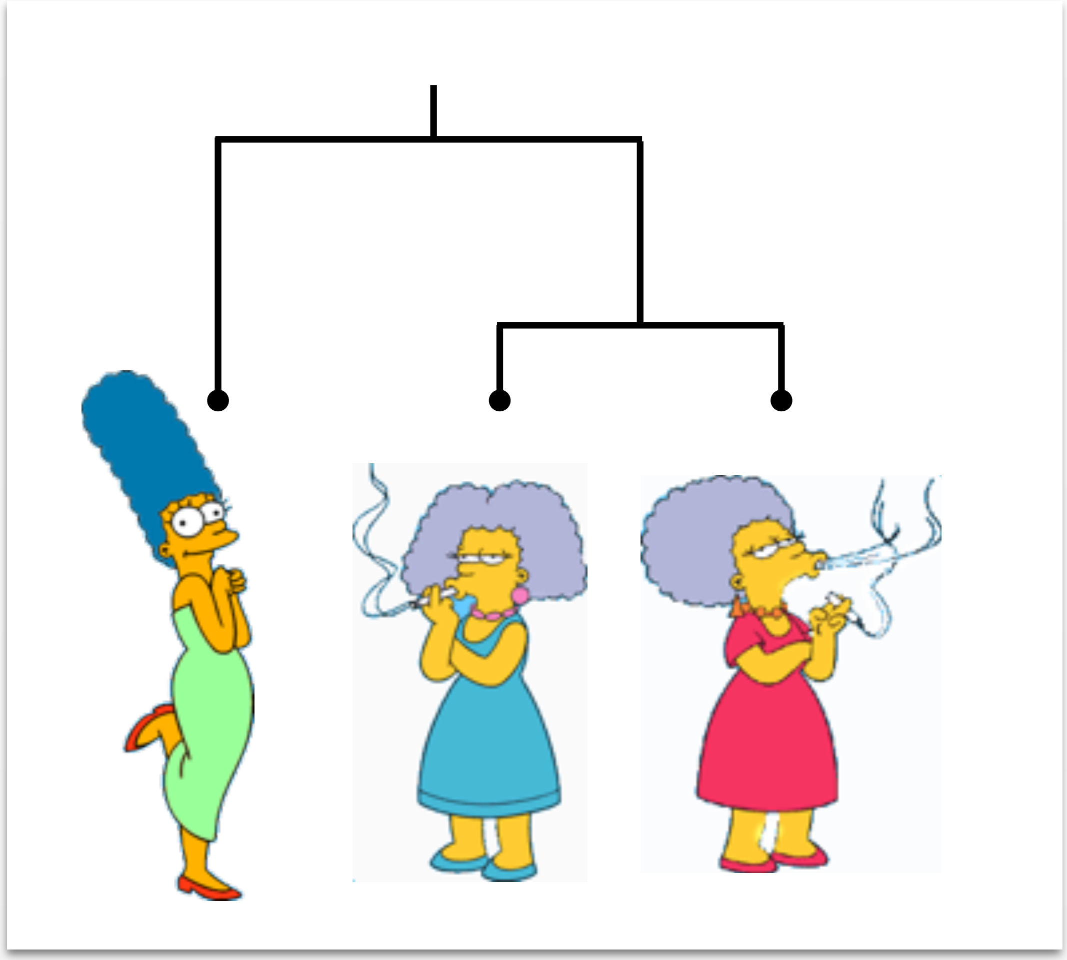
```
Change dress color,    1 point
Change earring shape, 1 point
Change hair part,      1 point
```

D(Patty, Selma) = **3**

Distance Marge and Selma

```
Change dress color,    1 point
Add earrings,          1 point
Decrease height,       1 point
Take up smoking,       1 point
Lose weight,           1 point
```

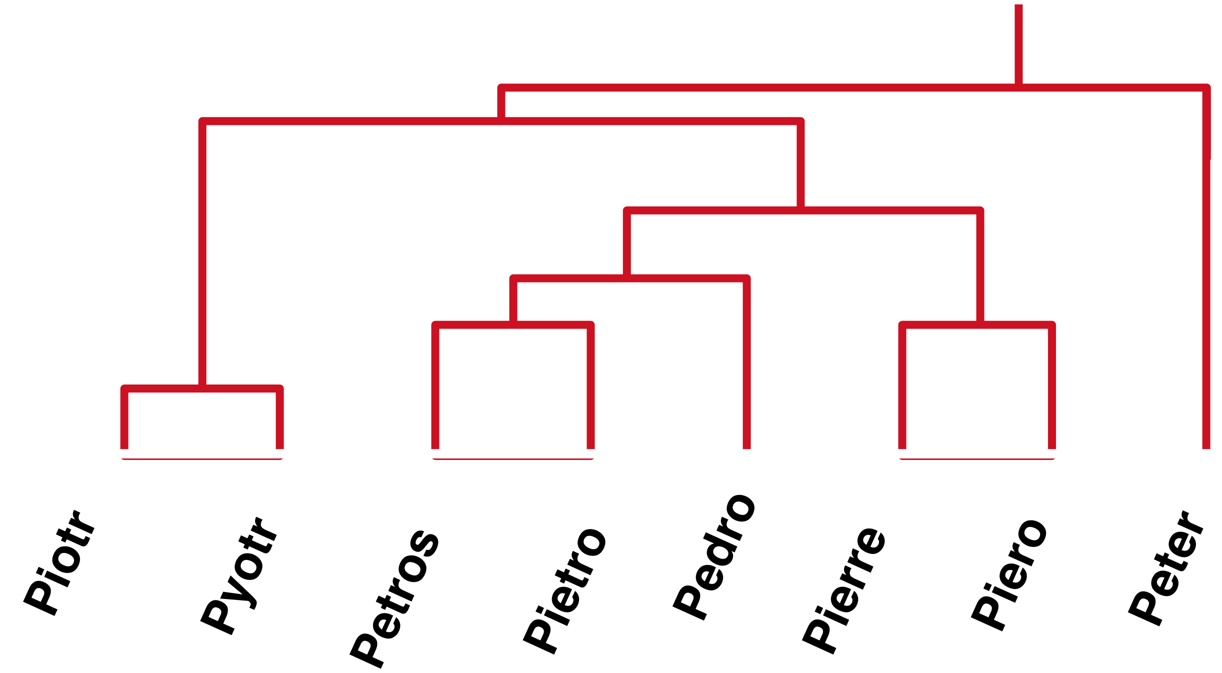
D(Marge,Selma) = **5**



Can be defined for any set of discrete features

Edit Distance for Strings

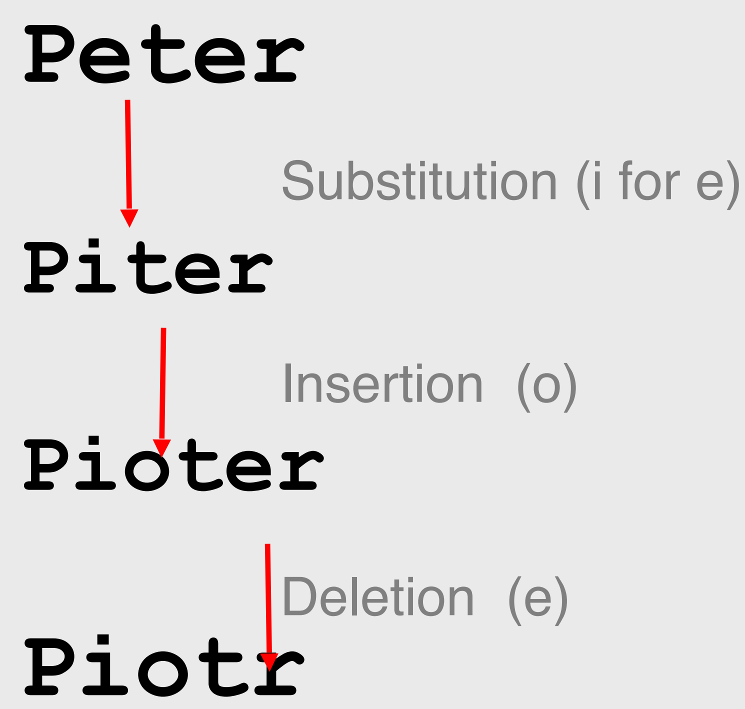
- Transform string *Q* into string *C*, using only ***Substitution***, ***Insertion*** and ***Deletion***.
- Assume that each of these operators has a **cost** associated with it.
- The similarity between two strings can be defined as the cost of the ***cheapest*** transformation from *Q* to *C*.



Similarity “Peter” and “Piotr”?

Substitution	1 Unit
Insertion	1 Unit
Deletion	1 Unit

D(**Peter**,**Piotr**) is 3



Hierarchical Clustering

(*Edit Distance*)

Pedro (Portuguese)

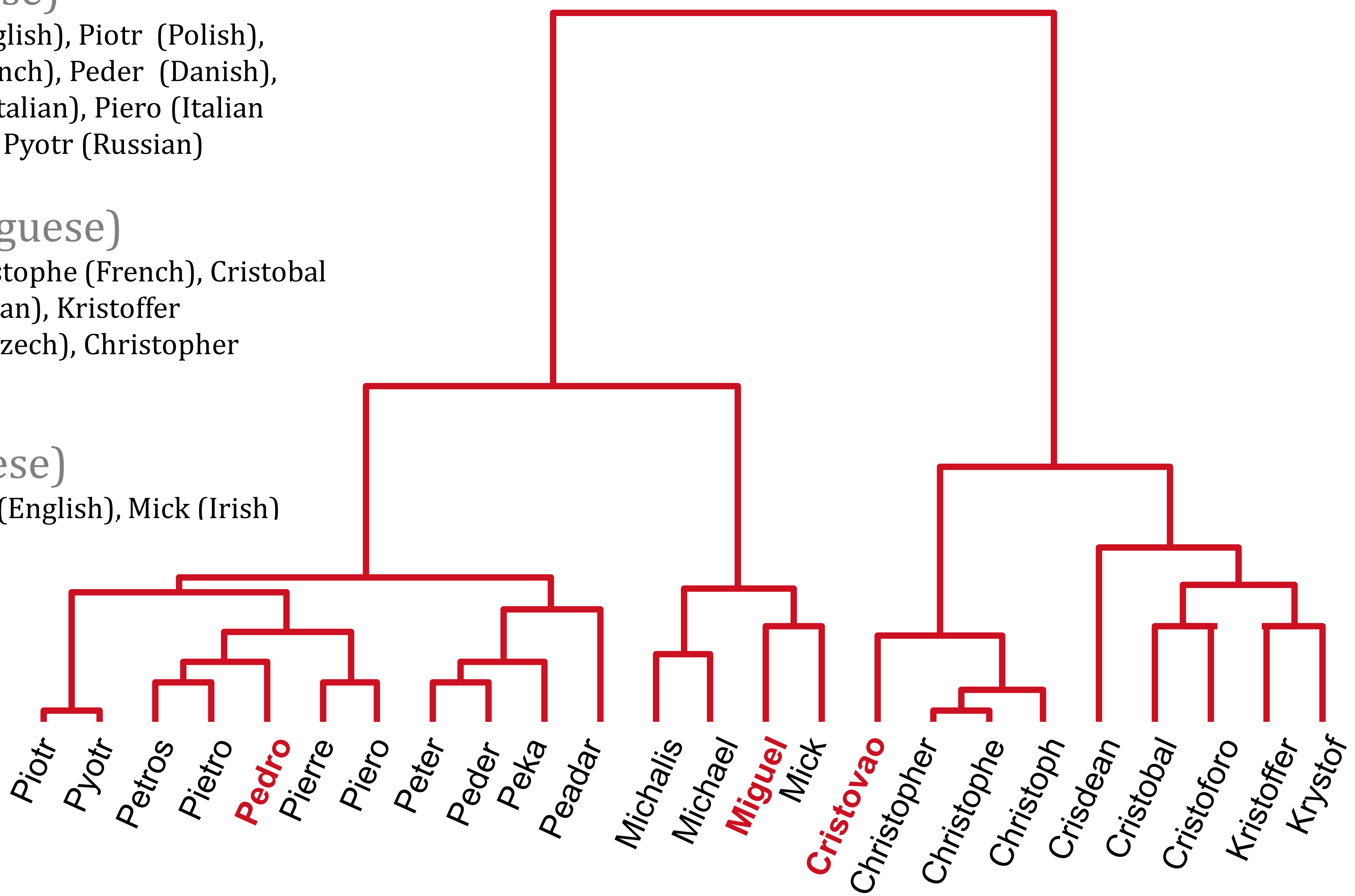
Petros (Greek), Peter (English), Piotr (Polish), Peadar (Irish), Pierre (French), Peder (Danish), Peka (Hawaiian), Pietro (Italian), Piero (Italian Alternative), Petr (Czech), Pyotr (Russian)

Cristovao (Portuguese)

Christoph (German), Christophe (French), Cristobal (Spanish), Cristoforo (Italian), Kristoffer (Scandinavian), Krystof (Czech), Christopher (English)

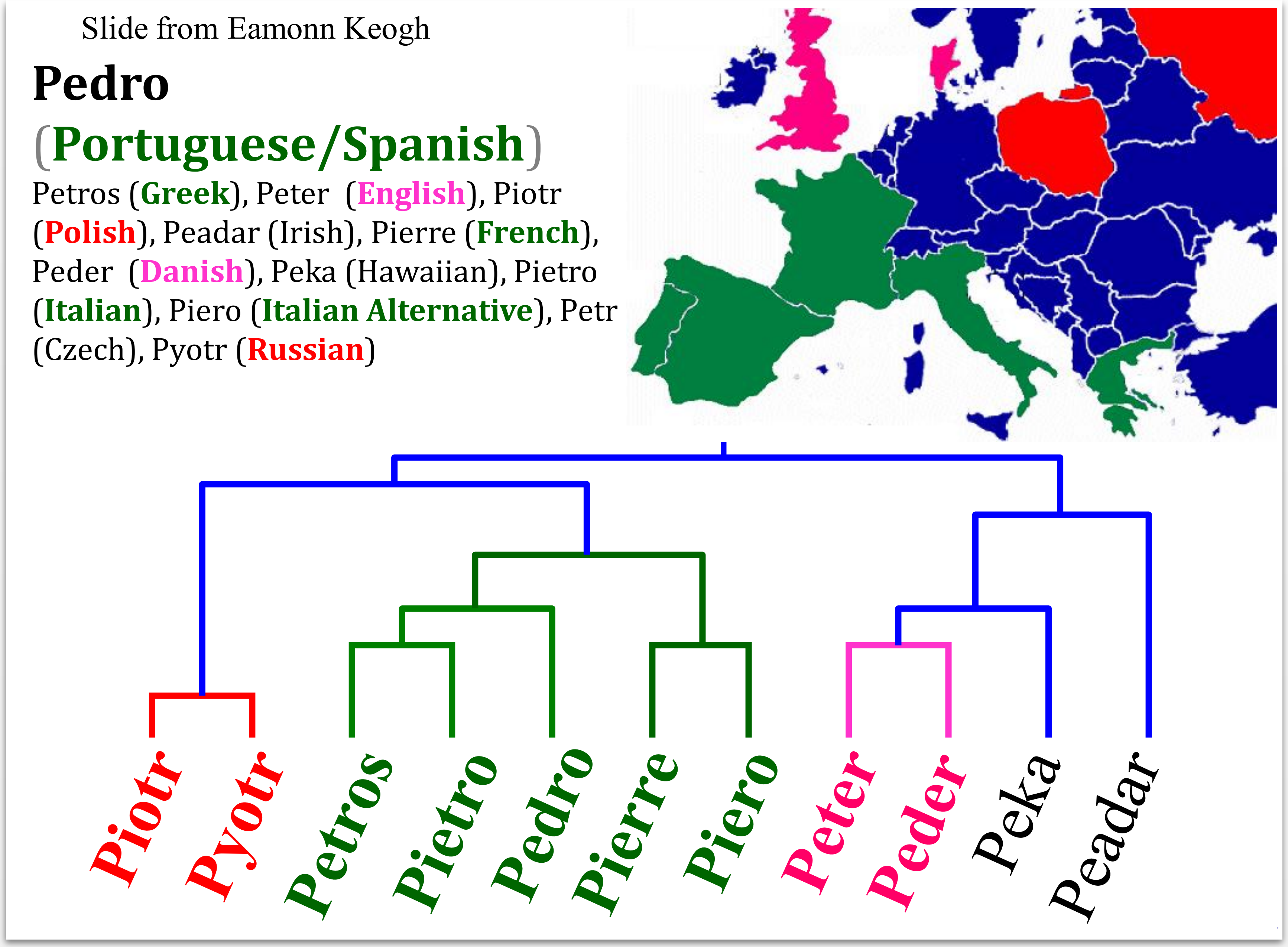
Miguel (Portuguese)

Michalis (Greek), Michael (English), Mick (Irish)



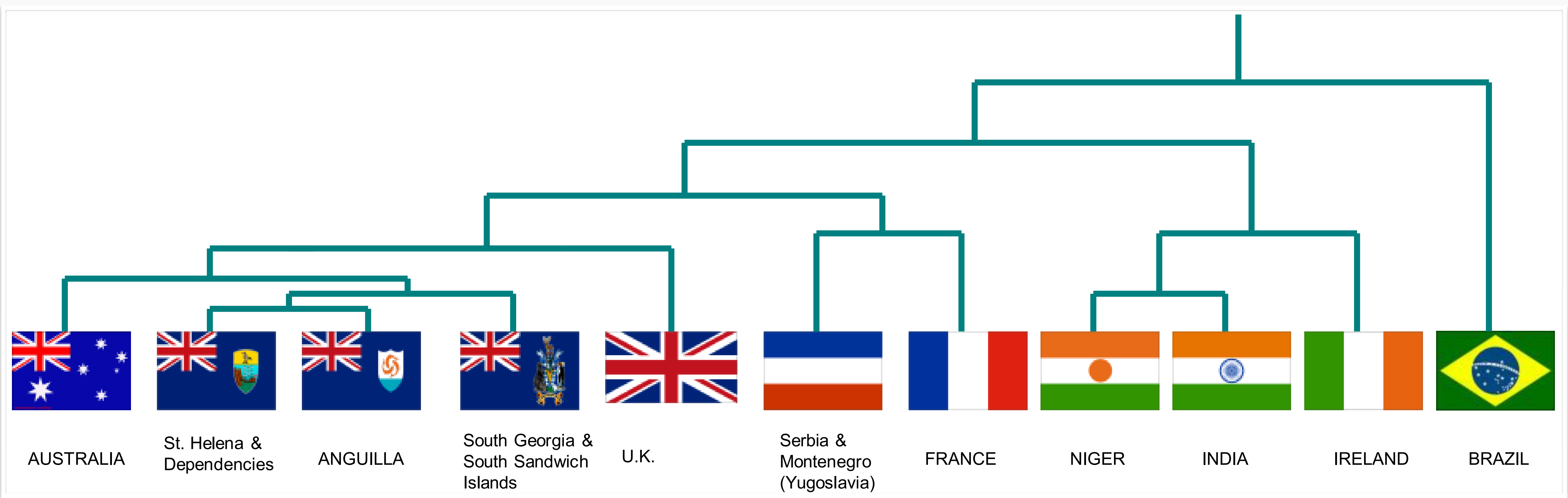
Meaningful Patterns

Edit distance yields clustering according to geography



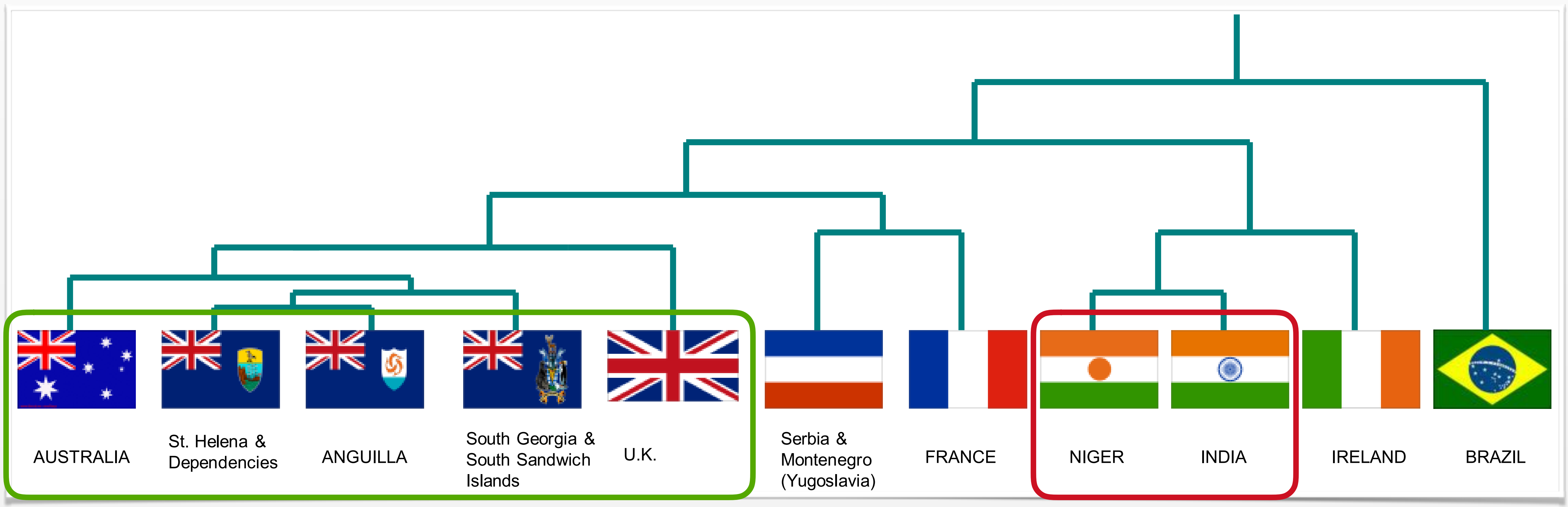
Spurious Patterns

In general clusterings will only be
as meaningful as your distance metric



Spurious Patterns

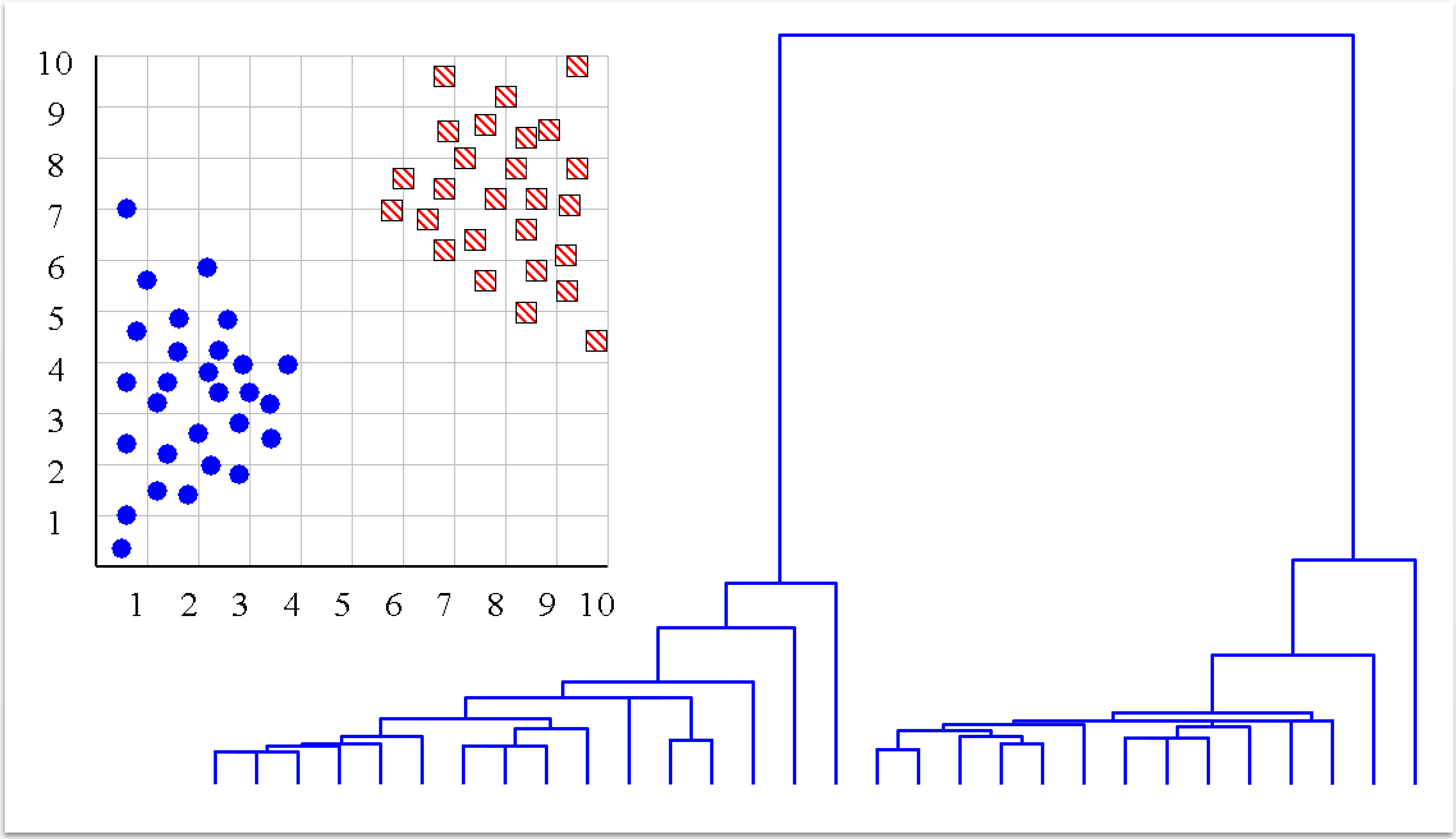
In general clusterings will only be
as meaningful as your distance metric



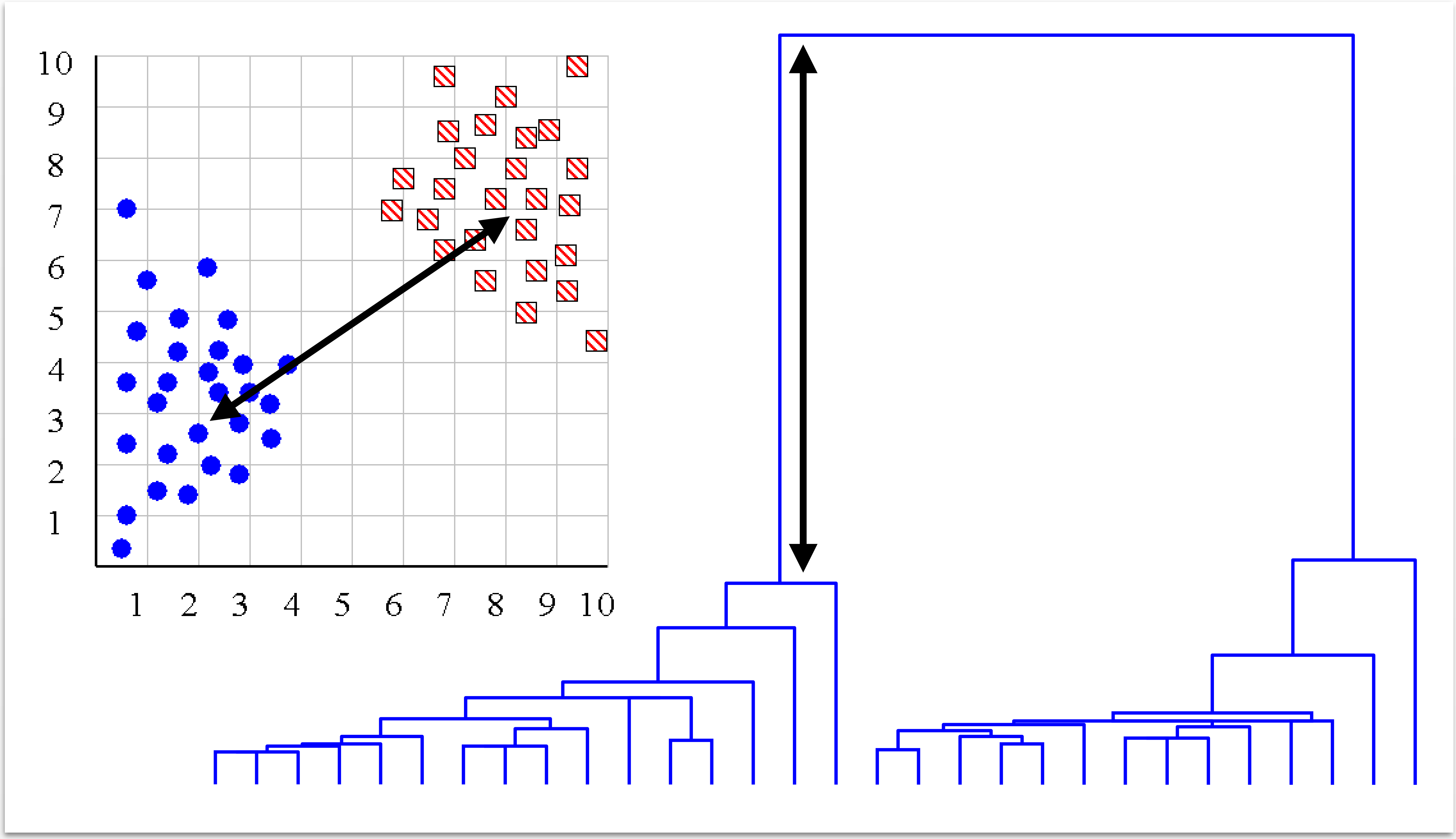
Former UK colonies

No relation

“Correct” Number of Clusters



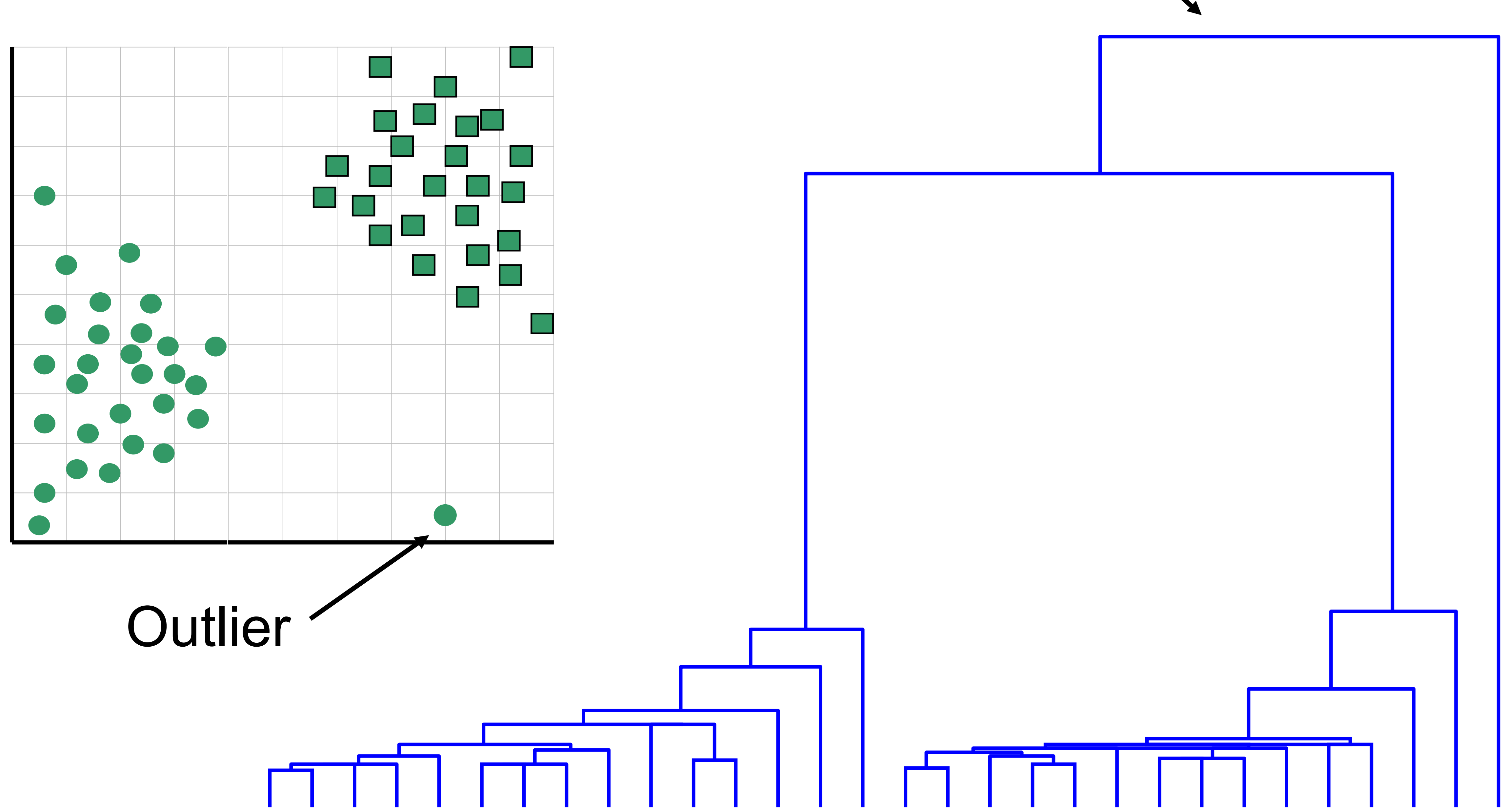
“Correct” Number of Clusters



Determine number of clusters by looking at distance

Detecting Outliers

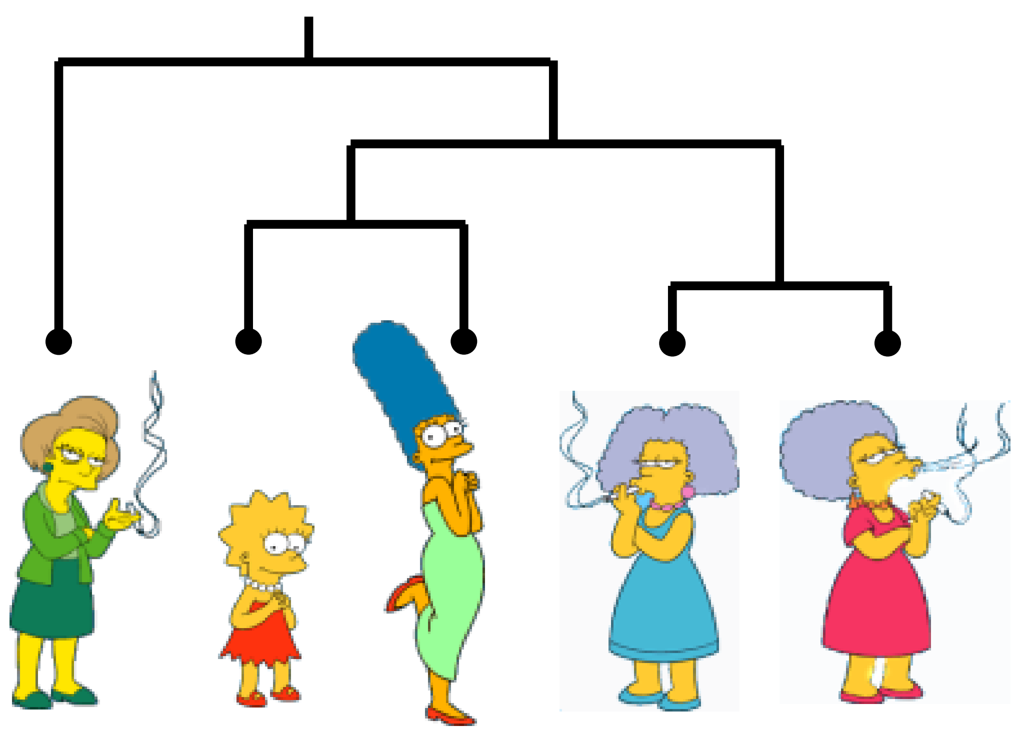
The single isolated branch is suggestive of a data point that is very different to all others



Bottom-up vs Top-down

The number of dendrograms with n leafs $= (2n - 3)!/[(2^{(n-2)})\,(n - 2)!]$

Number of Leafs	Number of Possible Dendrograms
2	1
3	3
4	15
5	105
...	…
10	34,459,425



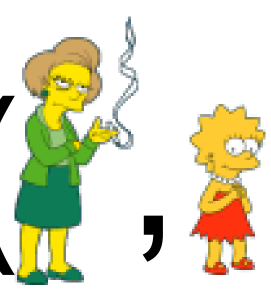
Since we cannot test all possible trees we will have to heuristic search of all possible trees. We could do this..

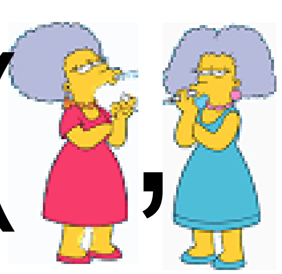
Bottom-Up (agglomerative): Starting with each item in its own cluster, find the best pair to merge into a new cluster. Repeat until all clusters are fused together.

Top-Down (divisive): Starting with all the data in a single cluster, consider every possible way to divide the cluster into two. Choose the best division and recursively operate on both sides.

Distance Matrix

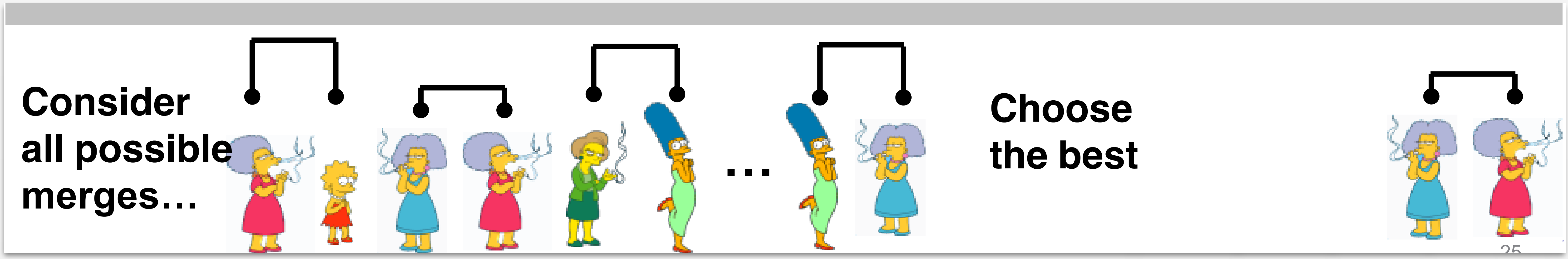
We begin with a distance matrix which contains the distances between every pair of objects in our database.

D() = 8

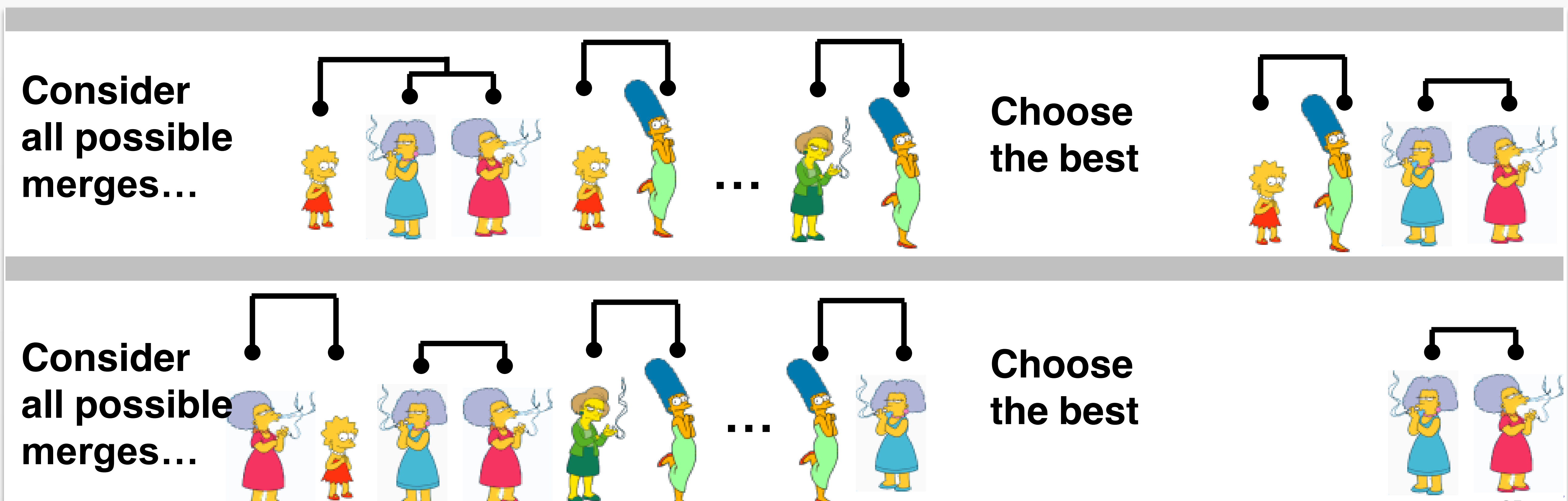
D() = 1

0	8	8	7	7
	0	2	4	4
		0	3	3
			0	1
				0

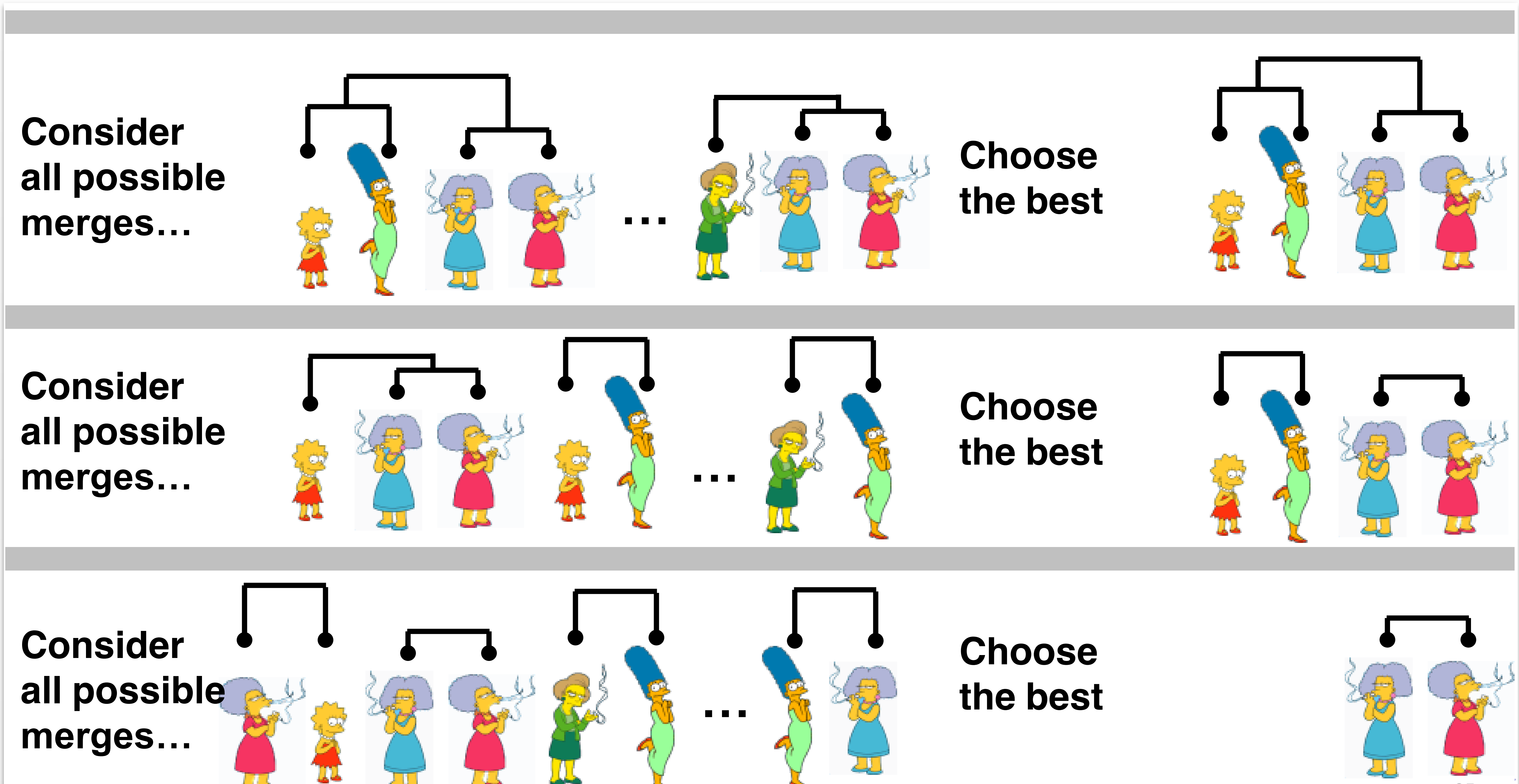
Bottom-up (Agglomerative Clustering)



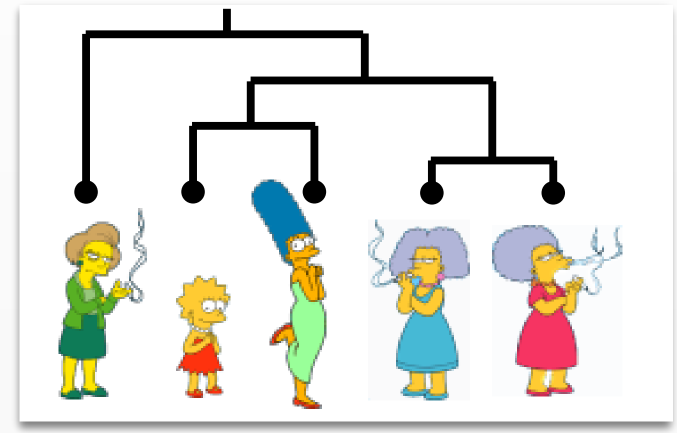
Bottom-up (Agglomerative Clustering)

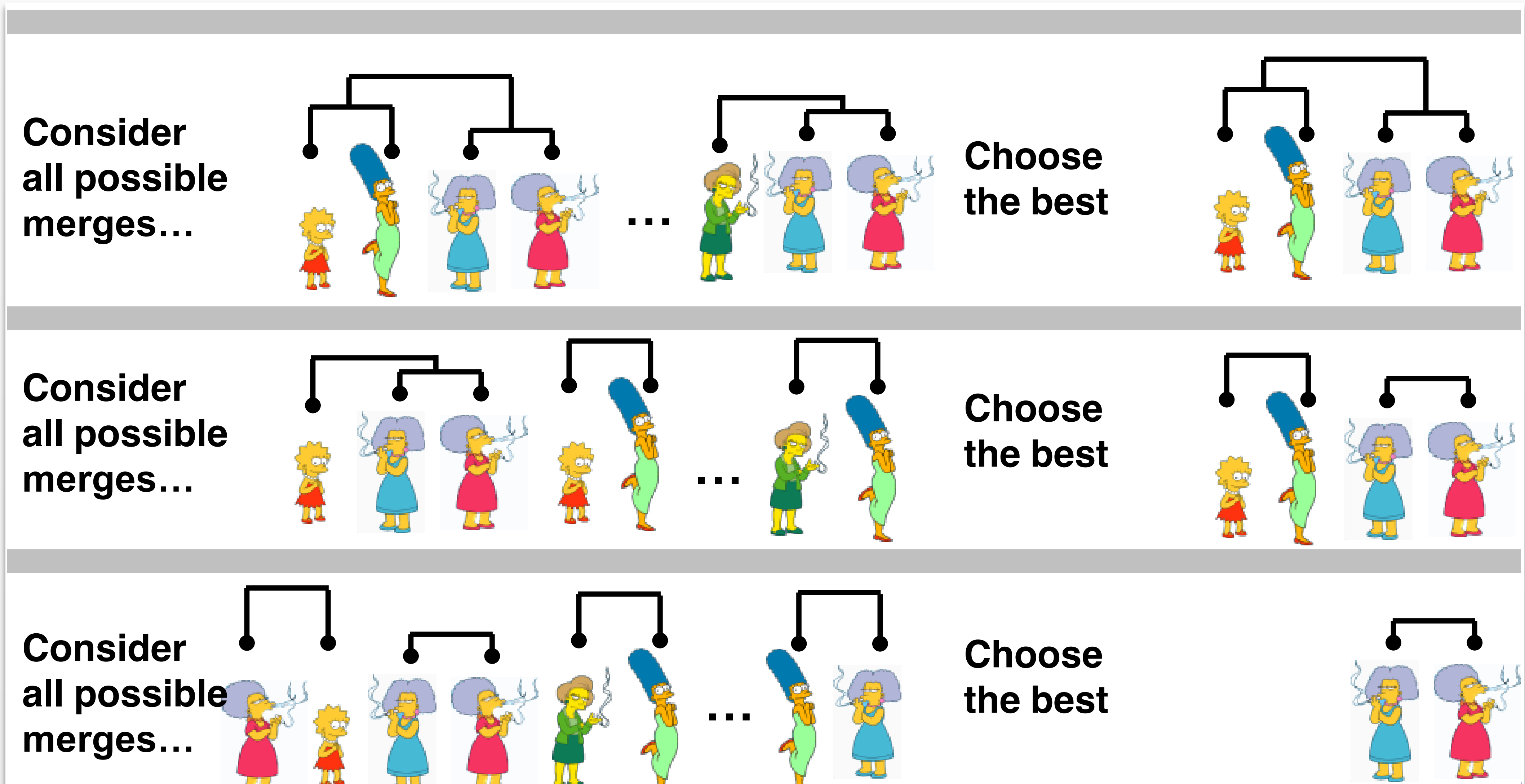


Bottom-up (Agglomerative Clustering)



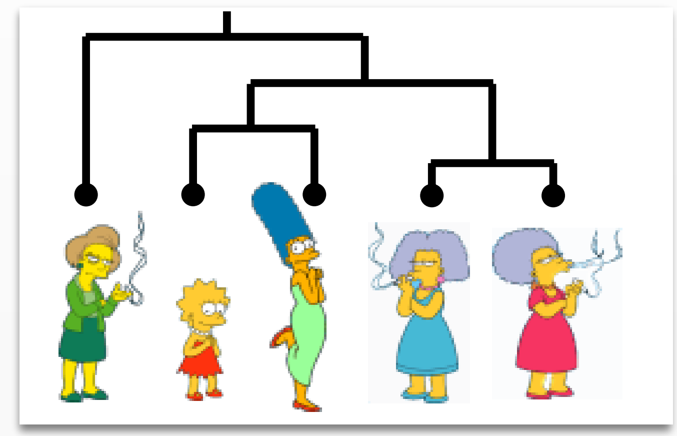
Bottom-up (Agglomerative Clustering)

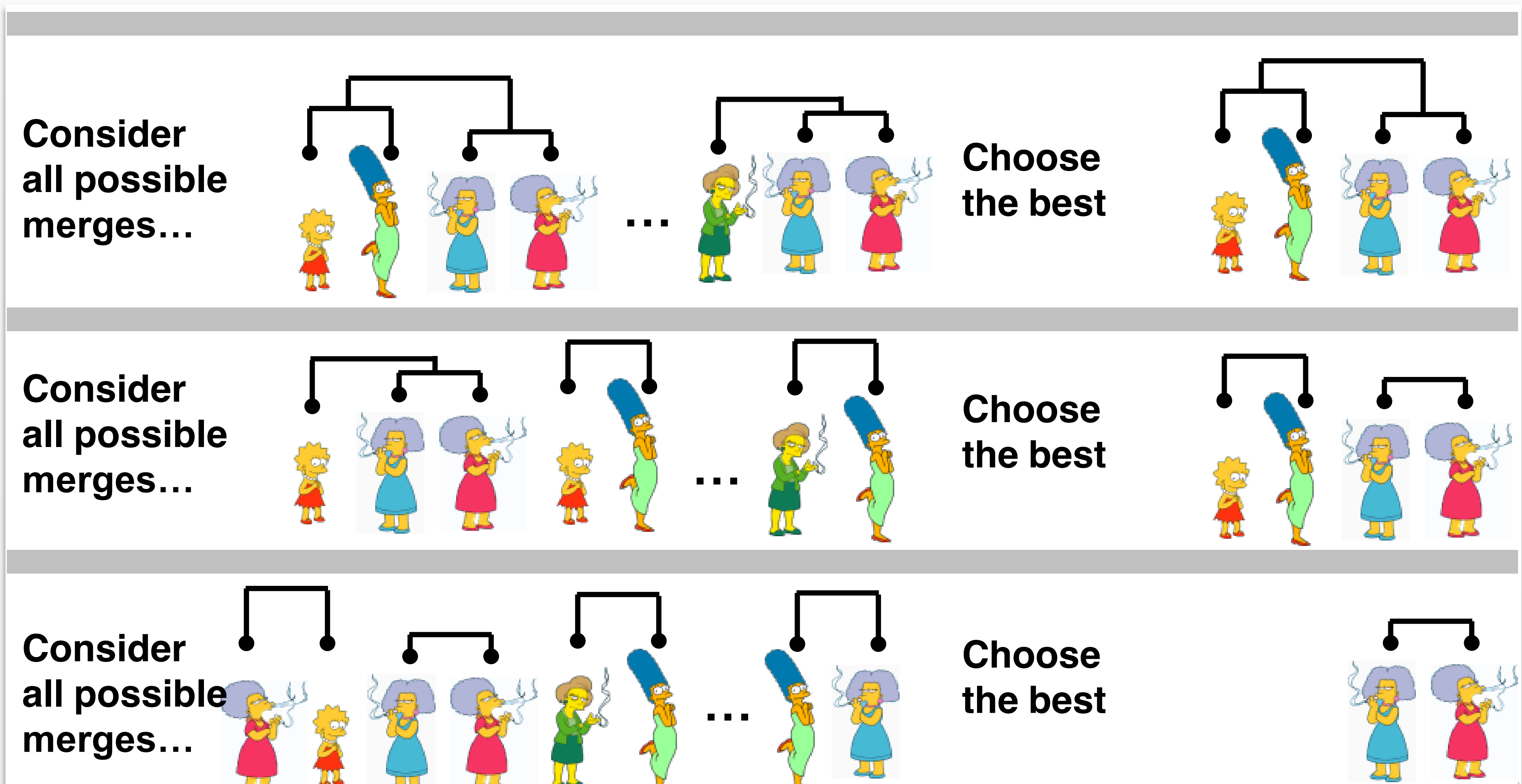




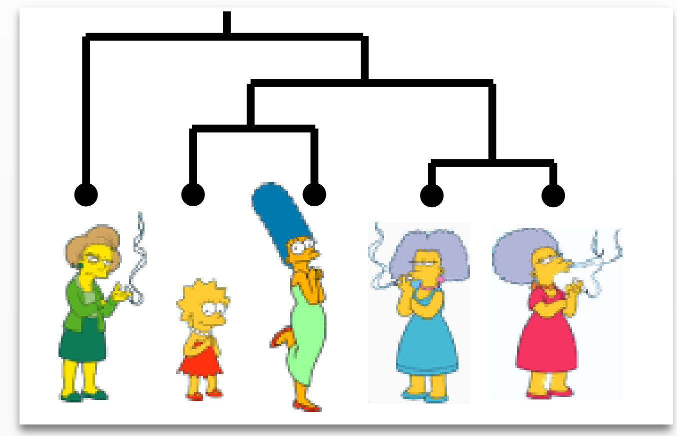
Bottom-up (Agglomerative Clustering)

Can you now implement this?

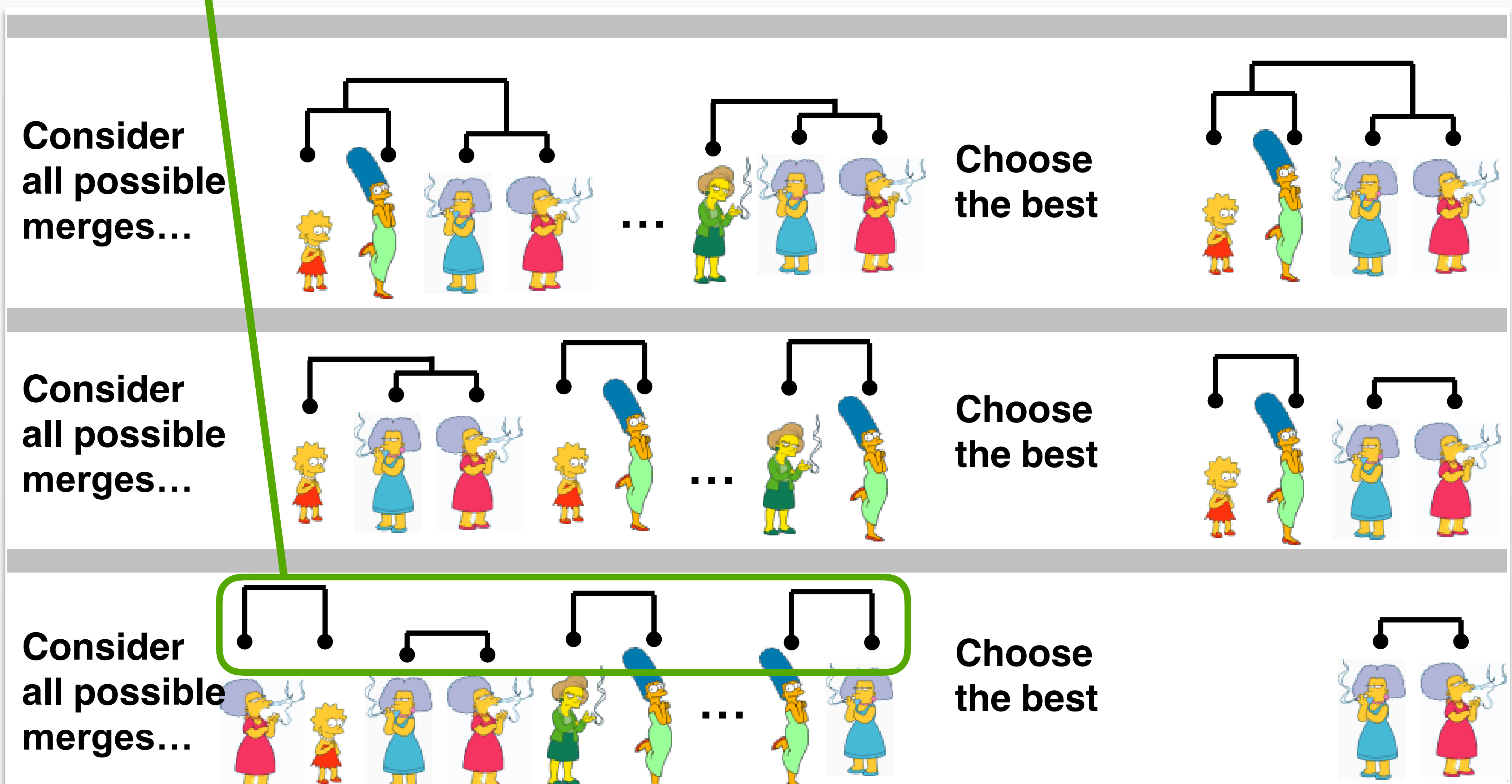




Bottom-up (Agglomerative Clustering)

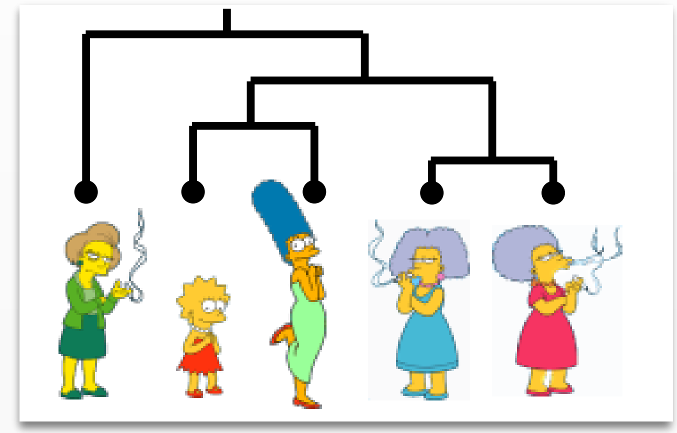


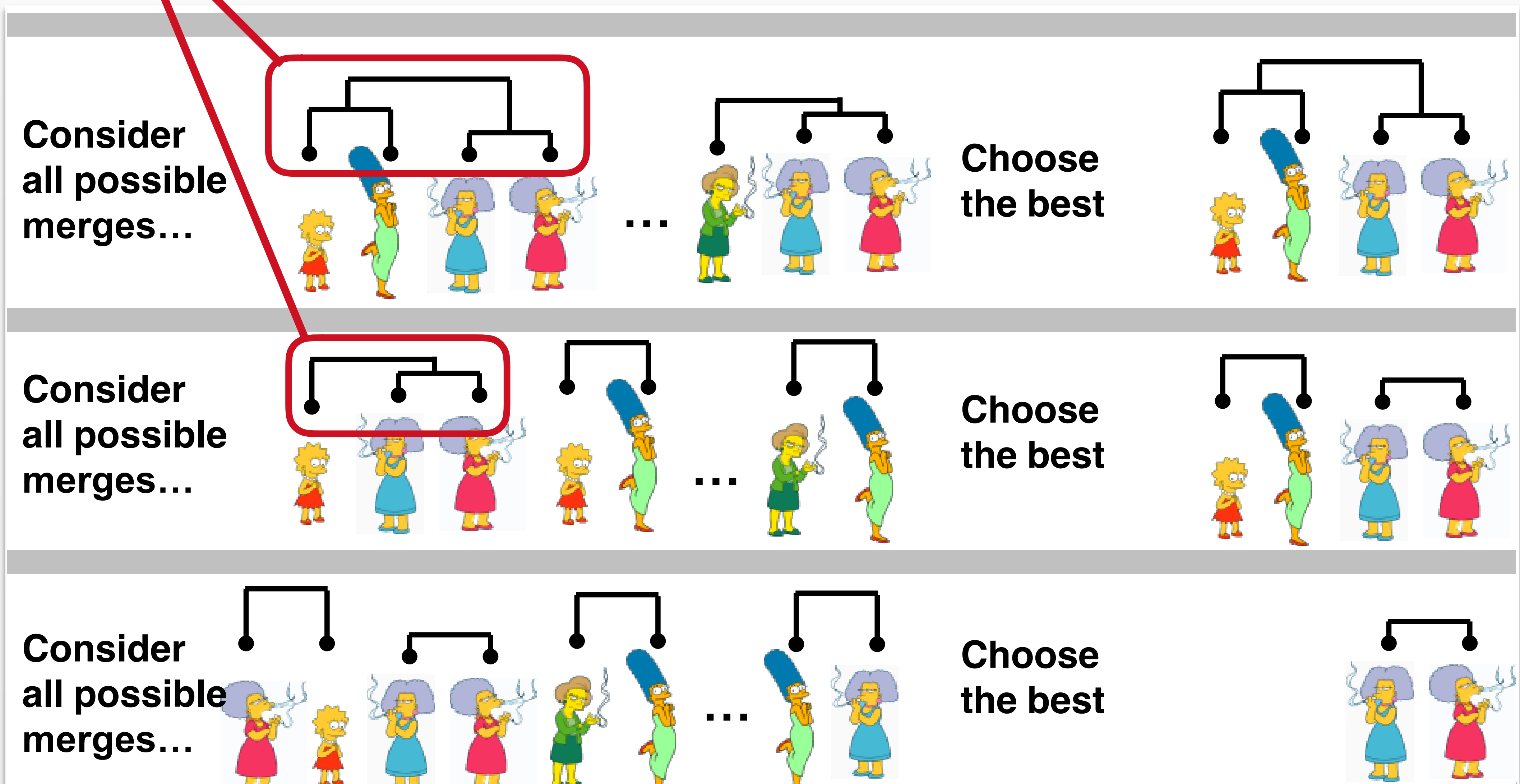
Distances between examples
(can calculate using metric)



Bottom-up (Agglomerative Clustering)

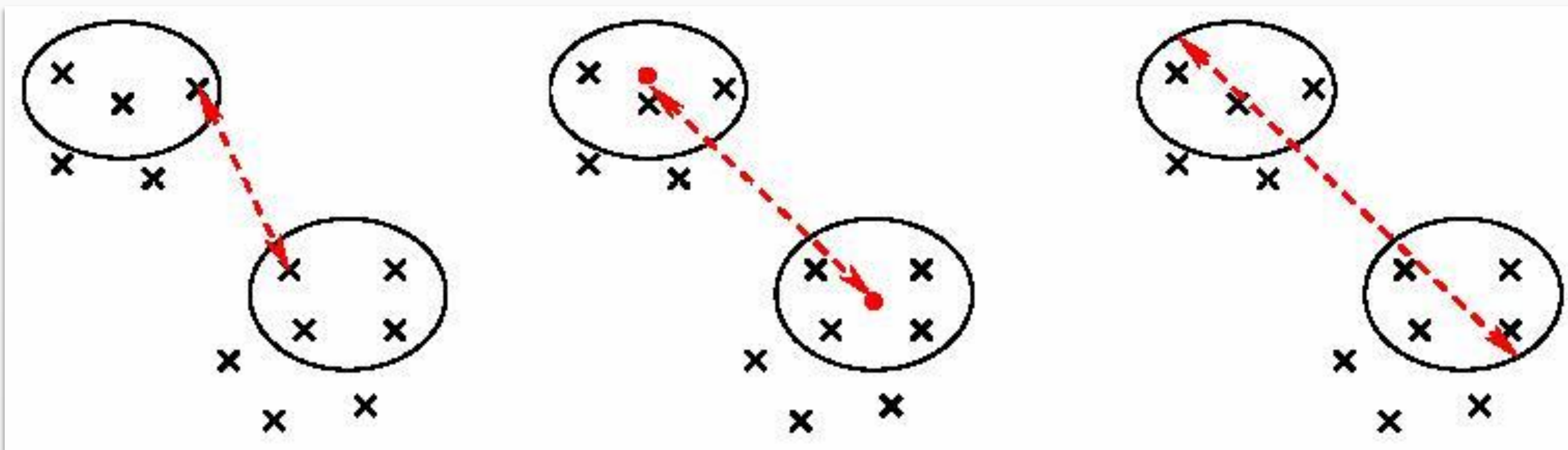
How do we calculate the distance to a cluster?





Distance Between Clusters

Single Linkage | Average Linkage | Complete Linkage



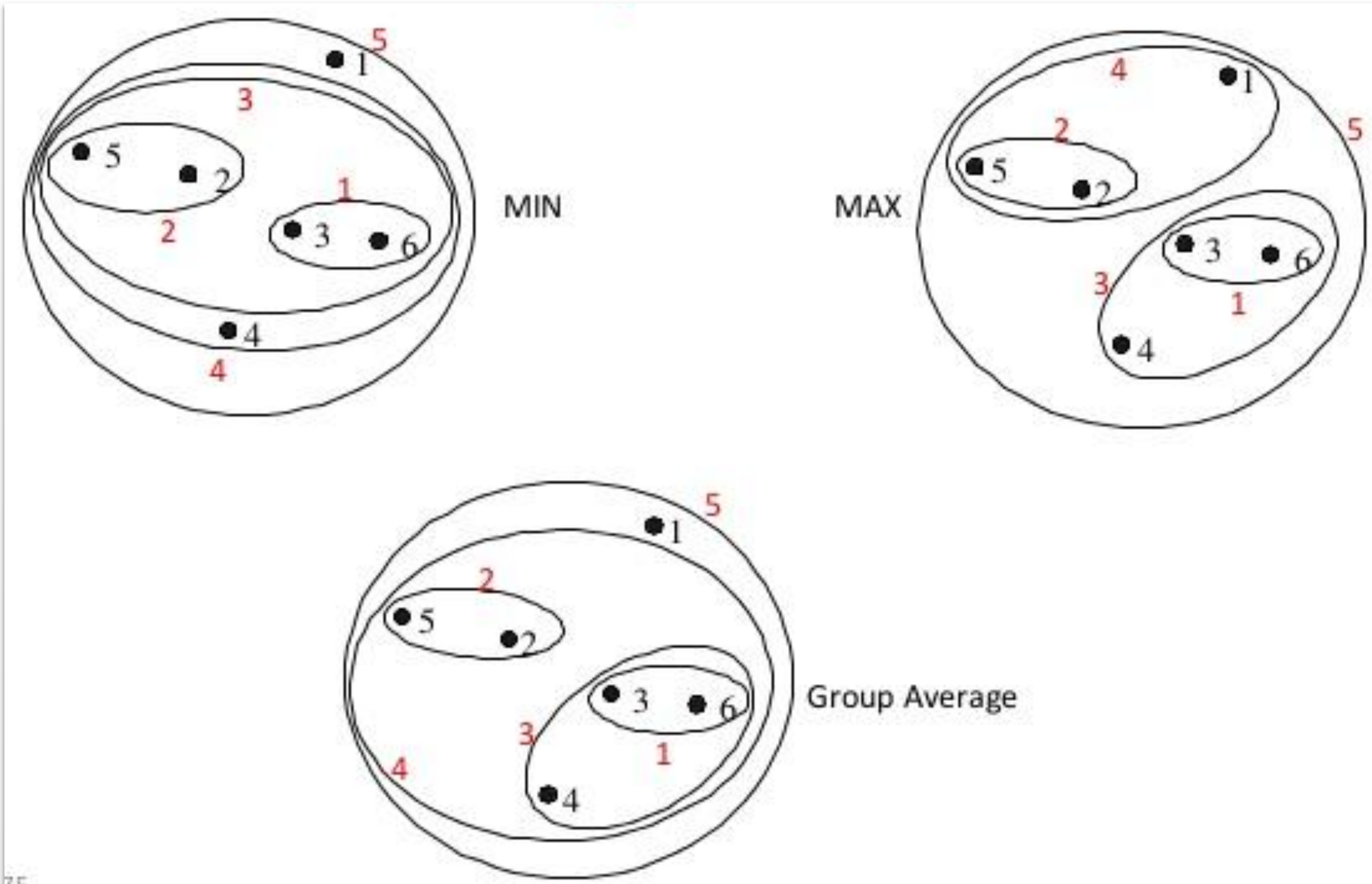
(nearest neighbor) | (mean distance) | (furthest neighbor)

Example

	P1	P2	P3	P4	P5	P6
P1	0	0.24	0.22	0.37	0.34	0.23
P2	0.24	0	0.15	0.2	0.14	0.25
P3	0.22	0.15	0	0.15	0.28	0.11
P4	0.37	0.2	0.15	0	0.29	0.22
P5	0.34	0.14	0.28	0.29	0	0.39
P6	0.23	0.25	0.11	0.22	0.39	0

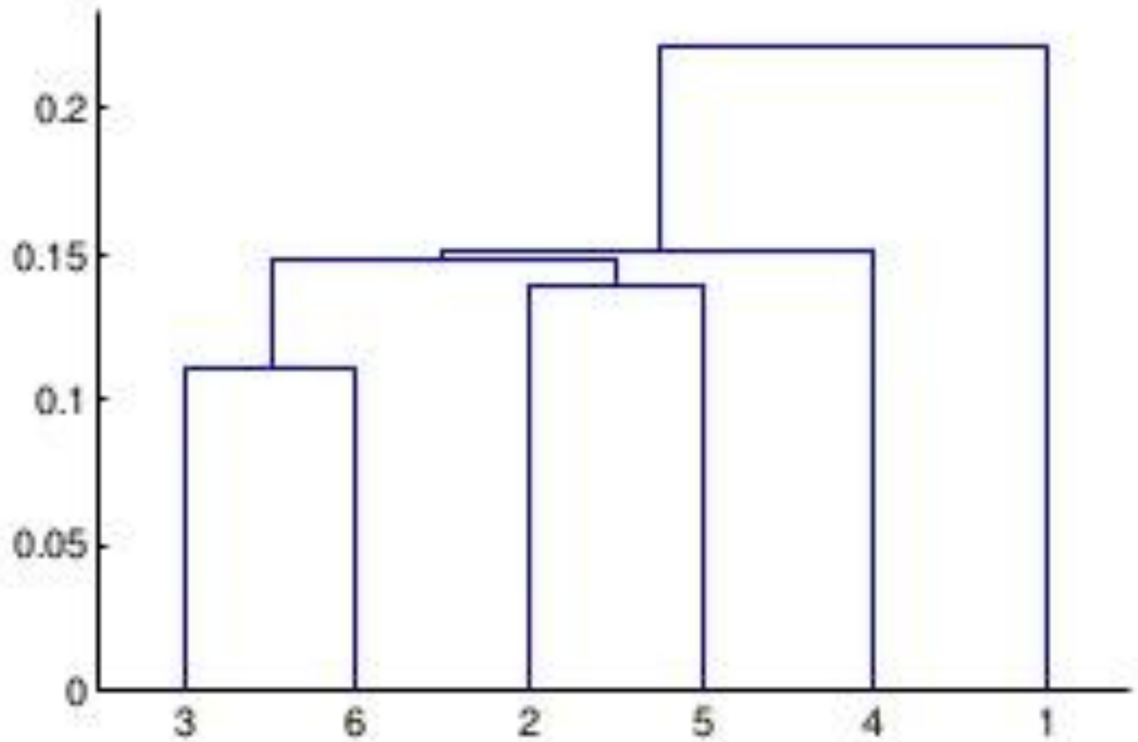
Euclidean distance

Example

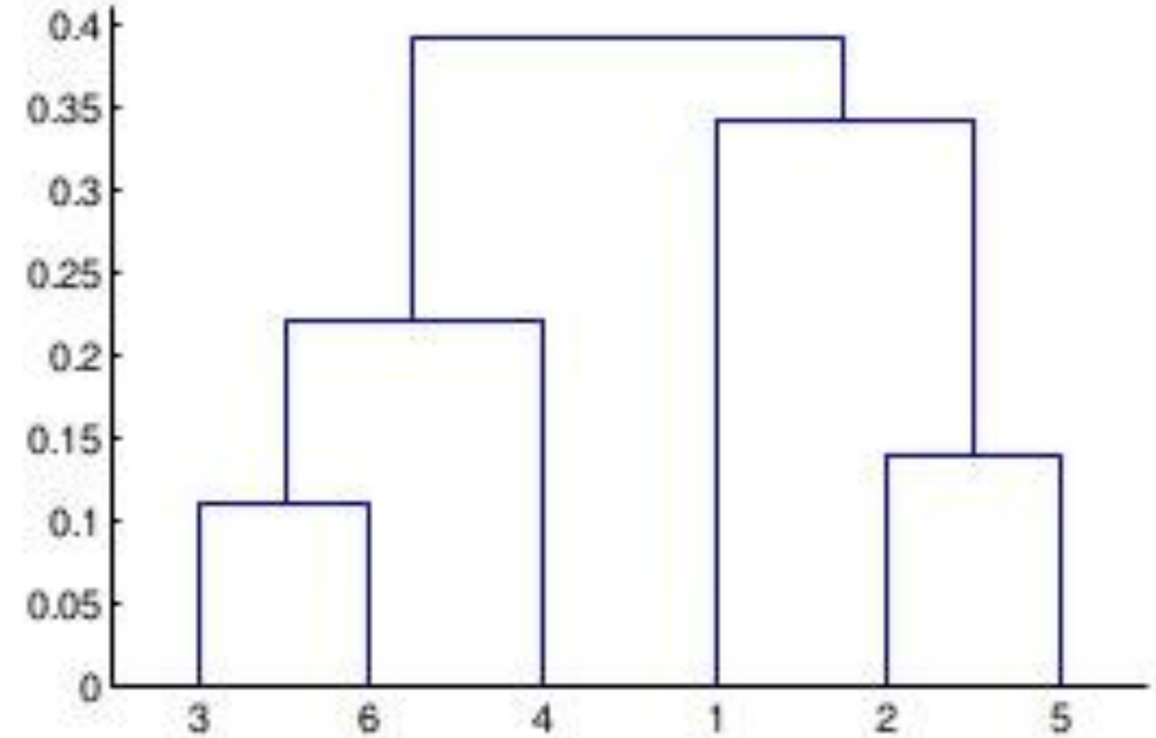


5
1
3
5
2
1
3
6
2
4
4
MIN
4
1
2
5
5
2
MAX
3
6
3
1
4
5
1
2
5
2
3
6
Group Average
3
4
1
4
75

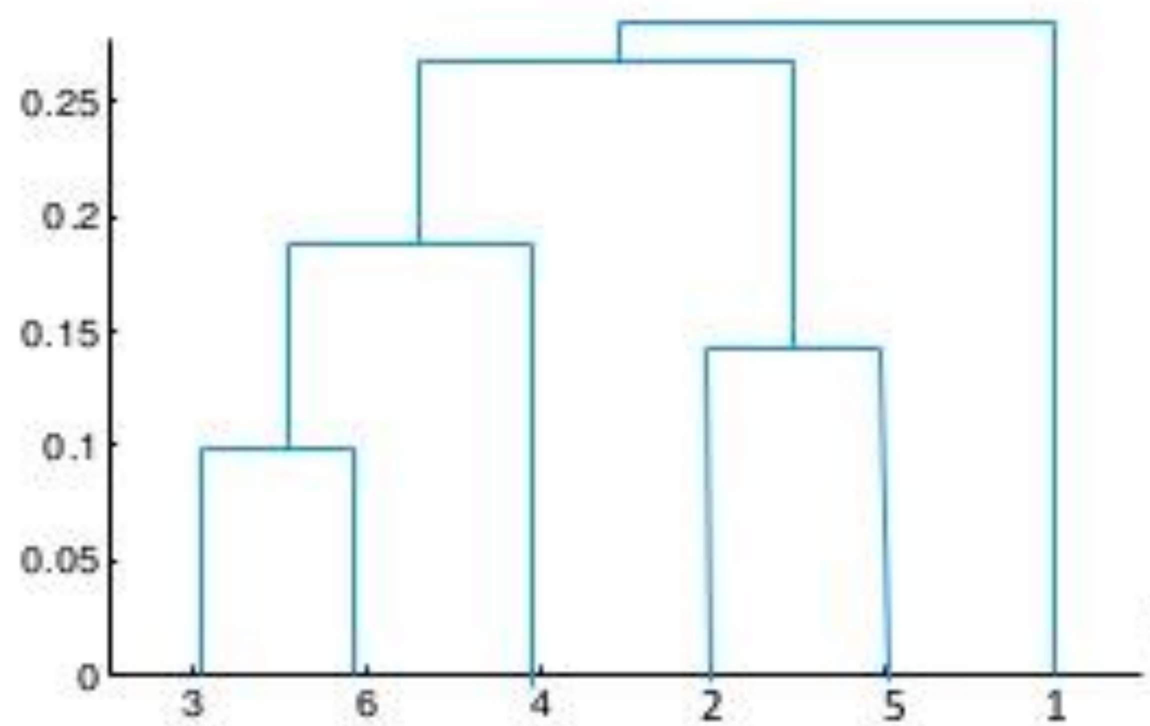
Example



MIN



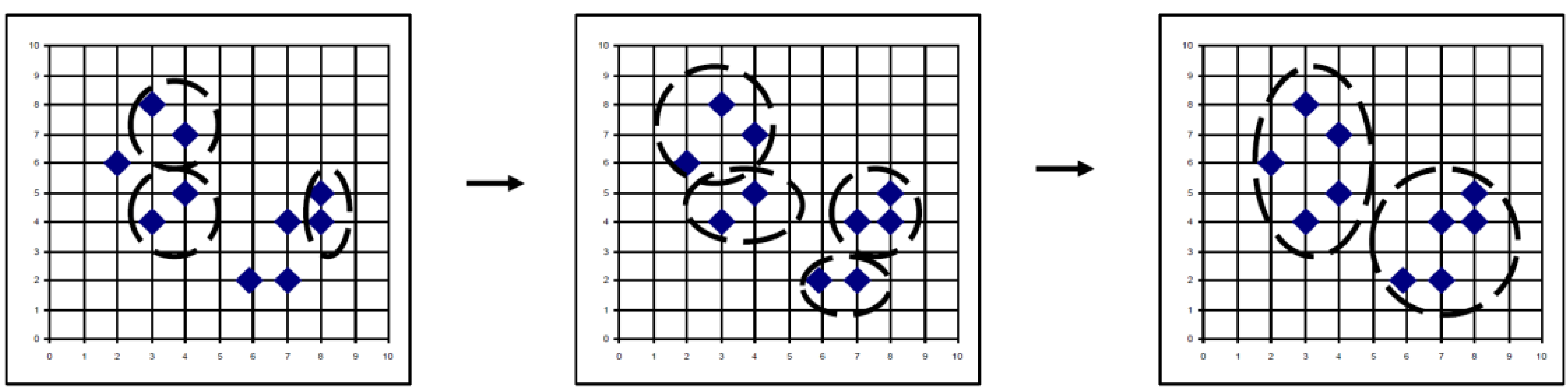
MAX



Group Average

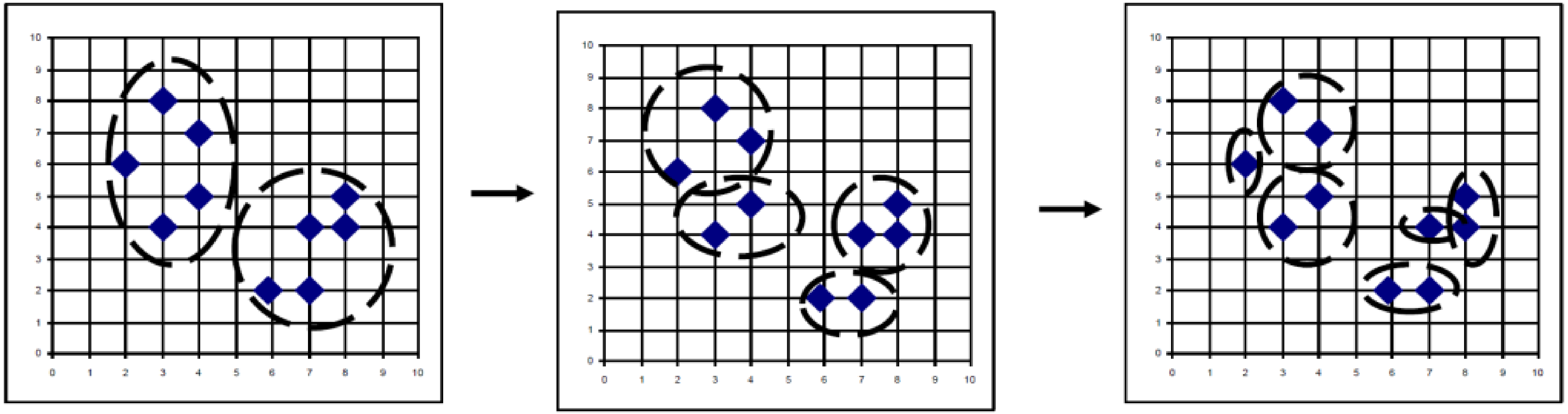
AGNES (Agglomerative Nesting)

- Introduced in Kaufmann and Rousseeuw (1990)
- Implemented in statistical packages, e.g., Splus
- Use the **single-link** method and the dissimilarity matrix
- Merge nodes that have the least dissimilarity
- Go on in a non-descending fashion
- Eventually all nodes belong to the same cluster



DIANA (Divisive Analysis)

- Introduced in Kaufmann and Rousseeuw (1990)
- Implemented in statistical analysis packages, e.g., Splus
- Inverse order of AGNES
- Eventually each node forms a cluster on its own



Hierarchical Clustering Summary

+ No need to specify number of clusters
+ Hierarchical structure maps nicely onto human intuition in some domains
- *Scaling:* Time complexity at least $O(n^2)$ in number of examples
- *Heuristic search method*: Local optima are a problem
- *Interpretation* of results is (very) subjective

Next Lecture: Partitional Clustering

Agglomerative Clustering	MiniBatch KMeans	Affinity Propagation	Spectral Clustering	DBSCAN