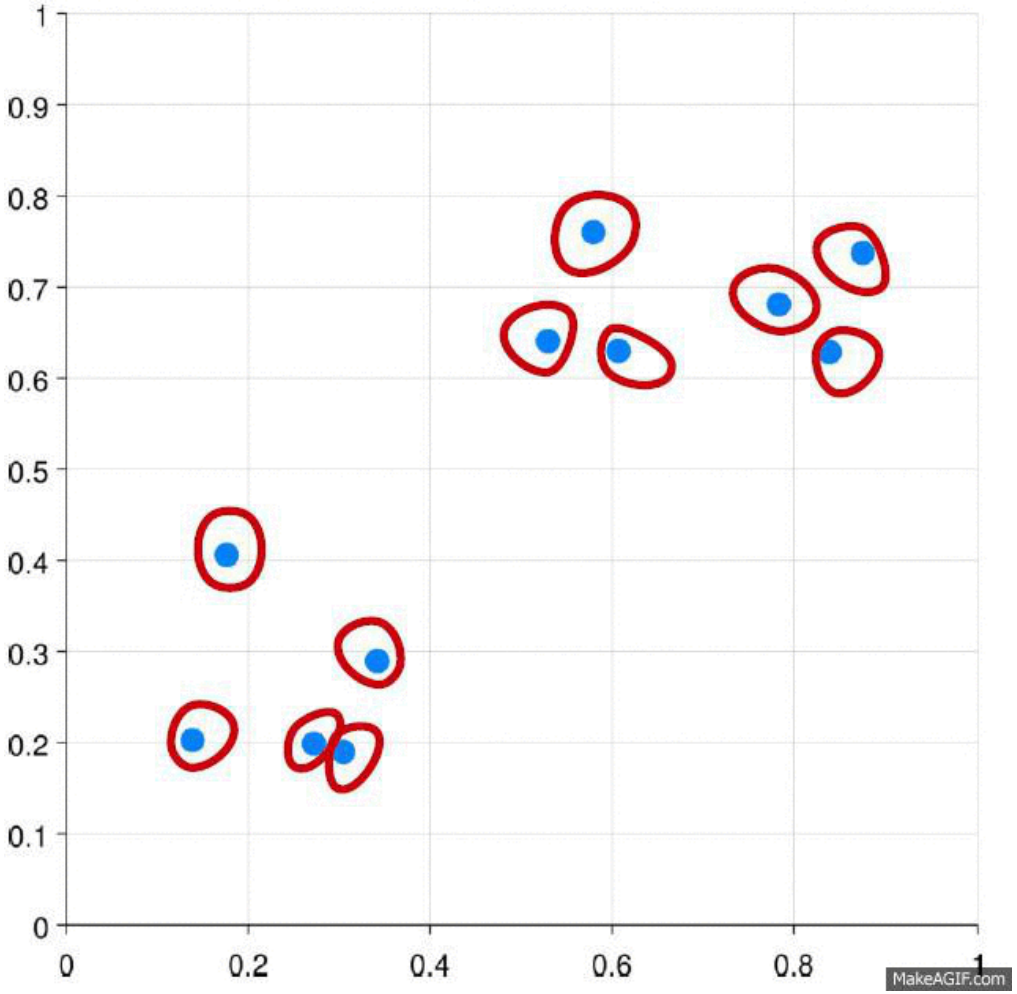
Hierarchical Clustering

Why Hierarchical Clustering?

Types

- There are two main methods of hierarchical clustering algorithm.
- First method is ***agglomerative approach*** , where we start from the bottom where all the objects are and going up (*bottom up approach*) through merging of objects. We begin with each individual objects and merge the two closest objects. The process is iterated until all objects are aggregated into a single group.
- Second method is ***divisive approach*** *(top down approach)* , where we start with assumption that all objects are group into a single group and then we split the group into two recursively until each group consists of a single object. One possible way to perform divisive approach is to first form a minimum spanning tree (e.g using Kruskal algorithm) and then recursively (or iteratively) split the tree by the largest distance.

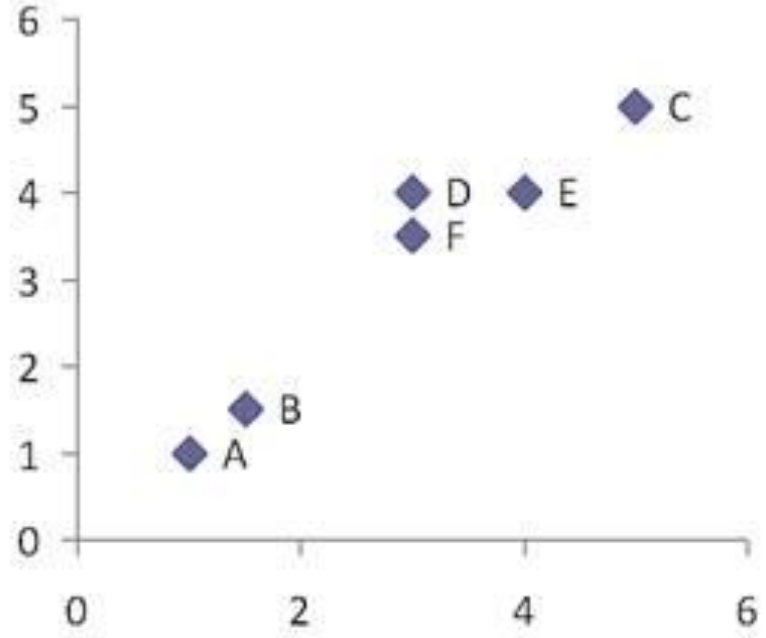
Agglomerative approach



Dataset

- To illustrate hierarchical clustering algorithm, let us use the following simple example. Suppose we have 6 objects (with name A, B, C, D, E and F) and each object have two measured features (X1 and X2). We can plot the features in a scattered plot to get the visualization of proximity between objects.

	X1	X2
A	1	1
B	1.5	1.5
C	5	5
D	3	4
E	4	4
F	3	3.5



Distance Matrix

- The proximity between object can be measured as distance matrix. Suppose we use Euclidean distance , we can compute the distance between objects using the following formula

$$d_{ij} = \left(\sum_k \left(x_{ik} - x_{jk} \right)^2 \right)^{\frac{1}{2}}$$

- For example, distance between object A = (1, 1) and B = (1.5, 1.5) is computed as

$$d_{AB} = \left((1-1.5)^2 + (1-1.5)^2 \right)^{\frac{1}{2}} = \sqrt{\frac{1}{2}} = 0.7071$$

- Another example of distance between object D = (3, 4) and F = (3, 3.5) is calculated as

$$d_{DF} = \left((3-3)^2 + (4-3.5)^2 \right)^{\frac{1}{2}} = 0.5$$

Distance Matrix

- Using the same way as above examples, we can compute all distances between objects and put the distances into a matrix form. Since distance is symmetric (i.e. distance between A and B is equal to distance between B and A), we can focus only on the lower or upper triangular matrix (green or pink part). The diagonal elements of distance matrix are zero represent distance from an object to itself.

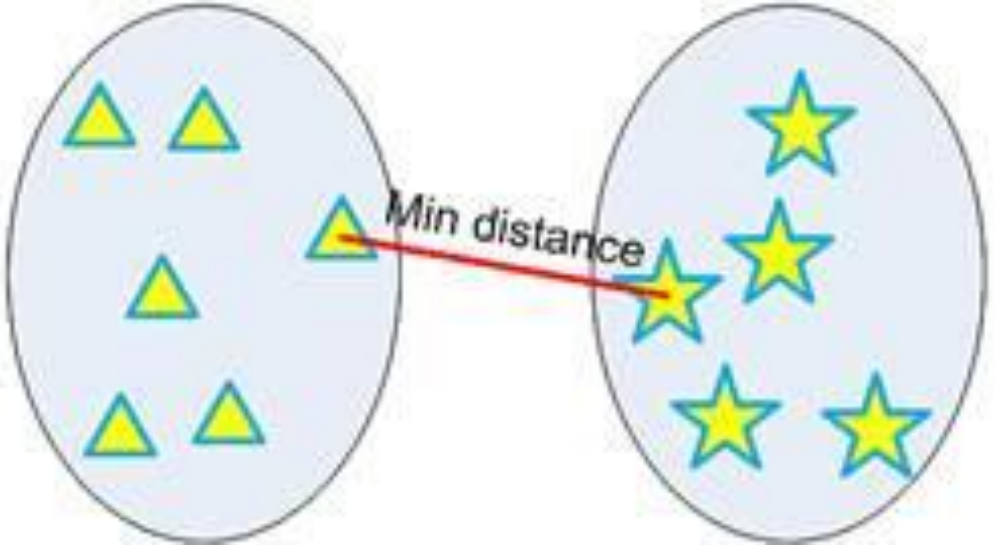
Dist	A	B	C	D	E	F
A	0.00	0.71	5.66	3.61	4.24	3.20
B	0.71	0.00	4.95	2.92	3.54	2.50
C	5.66	4.95	0.00	2.24	1.41	2.50
D	3.61	2.92	2.24	0.00	1.00	0.50
E	4.24	3.54	1.41	1.00	0.00	1.12
F	3.20	2.50	2.50	0.50	1.12	0.00

- Clearly the minimum distance is 0.5 (between object D and F).

0.71	5.66	3.61	4.24	3.20	4.95	2.92	3.54	2.50	2.24	1.41	2.50	1.00	**0.50**	1.12

Numerical Example

- Minimum distance clustering is also called as single linkage hierarchical clustering or nearest neighbor clustering. Distance between two clusters is defined by the minimum distance between objects of the two clusters, as shown below.



Numerical Example

- In each step of the iteration, we find the closest pair clusters. In this case, the closest cluster is between cluster F and D with shortest distance of 0.5. **Thus, we group cluster D and F into cluster (D, F).**

Dist	A	B	C	D	E	F
A	0.00	0.71	5.66	3.61	4.24	3.20
B	0.71	0.00	4.95	2.92	3.54	2.50
C	5.66	4.95	0.00	2.24	1.41	2.50
D	3.61	2.92	2.24	0.00	1.00	0.50
E	4.24	3.54	1.41	1.00	0.00	1.12
F	3.20	2.50	2.50	0.50	1.12	0.00

Numerical Example

- Then we update the distance matrix (see distance matrix below). Distance between ungrouped clusters will not change from the original distance matrix. Now the problem is how to calculate distance between newly grouped clusters (D, F) and other clusters?

Min Distance (Single Linkage)

Dist	A	B	C	D, F	E
A	0.00	0.71	5.66	?	4.24
B	0.71	0.00	4.95	?	3.54
C	5.66	4.95	0.00	?	1.41
D, F	?	?	?	0.00	?
E	4.24	3.54	1.41	?	0.00

Numerical Example

- That is exactly where the linkage rule comes into effect. Using single linkage, we specify minimum distance between original objects of the two clusters.
- Using the input distance matrix, distance between cluster (D, F) and cluster A is computed as

$$d_{(D,F)\mapsto A} = \min\left(d_{DA}, d_{FA}\right) = \min\left(3.61, 3.20\right) = 3.20$$

- Distance between cluster (D, F) and cluster B is

$$d_{(D,F)\mapsto B} = \min\left(d_{DB}, d_{FB}\right) = \min\left(2.92, 2.50\right) = 2.50$$

- Similarly, distance between cluster (D, F) and cluster C is

$$d_{(D,F)\mapsto C} = \min\left(d_{DC}, d_{FC}\right) = \min\left(2.24, 2.50\right) = 2.24$$

- Finally, distance between cluster E and cluster (D, F) is calculated as

$$d_{E\to(D,F)} = \min\left(d_{ED}, d_{EF}\right) = \min\left(1.00, 1.12\right) = 1.00$$

- Then, the updated distance matrix becomes

Numerical Example

- Then, the updated distance matrix becomes

Min Distance (Single Linkage)

Dist	A	B	C	D, F	E
A	0.00	0.71	5.66	3.20	4.24
B	0.71	0.00	4.95	2.50	3.54
C	5.66	4.95	0.00	2.24	1.41
D, F	3.20	2.50	2.24	0.00	1.00
E	4.24	3.54	1.41	1.00	0.00

- Looking at the lower triangular updated distance matrix, we found out that the closest distance between cluster B and cluster A is now 0.71. Thus, we group cluster A and cluster B into a single cluster name (A, B).

Numerical Example

- Now we update the distance matrix. Aside from the first row and first column, all the other elements of the new distance matrix are not changed.

Dist	A,B	C	(D, F)	E
A,B	0	?	?	?
C	?	0	2.24	1.41
(D, F)	?	2.24	0	1.00
E	?	1.41	1.00	0

Numerical Example

- Using the input distance matrix (size 6 by 6), distance between cluster C and cluster (D, F) is computed as

$$d_{C\to(A,B)} = \min\left(d_{CA}, d_{CB}\right) = \min\left(5.66, 4.95\right) = 4.95$$

- Distance between cluster (D, F) and cluster (A, B) is the minimum distance between all objects involves in the two clusters

$$d_{(D,F)\to(A,B)} = \min\left(d_{DA}, d_{DB}, d_{FA}, d_{FB}\right) = \min\left(3.61, 2.92, 3.20, 2.50\right) = 2.50$$

- Similarly, distance between cluster E and (A, B) is

$$d_{E\to(A,B)} = \min\left(d_{EA}, d_{EB}\right) = \min\left(4.24, 3.54\right) = 3.54$$

Numerical Example

- Then the updated distance matrix is

Min Distance (Single Linkage)

Dist	A,B	C	(D, F)	E
A,B	0	4.95	2.50	3.54
C	**4.95**	0	2.24	1.41
(D, F)	**2.50**	2.24	0	1.00
E	**3.54**	1.41	1.00	0

- Observing the lower triangular of the updated distance matrix, we can see that the closest distance between clusters happens between cluster E and (D, F) at distance 1.00. Thus, we cluster them together into cluster ((D, F), E).

Numerical Example

- The updated distance matrix is given below.

Min Distance (Single Linkage)

Dist	(A,B)	C	(D, F), E
(A,B)	0.00	4.95	2.50
C	4.95	0.00	1.41
(D, F), E	**2.50**	**1.41**	0.00

Numerical Example

- Distance between cluster ((D, F), E) and cluster (A, B) is calculated as

$$d_{((D,F),E)\mapsto(A,B)} = \min\left(d_{DA}, d_{DB}, d_{FA}, d_{FB}, d_{EA}, d_{EB}\right) = \min\left(3.61, 2.92, 3.20, 2.50, 4.24, 3.54\right) = 2.50$$

- Distance between cluster ((D, F), E) and cluster C yields the minimum distance of 1.41. This distance is computed as

$$d_{((D,F),E)\mapsto C} = \min\left(d_{DC}, d_{FC}, d_{EC}\right) = \min\left(2.24, 2.50, 1.41\right) = 1.41$$

- After that, we merge cluster ((D, F), E) and cluster C into a new cluster name (((D, F), E), C).

Numerical Example

- The updated distance matrix is shown in the figure below

Min Distance (Single Linkage)

Dist	(A,B)	(D, F), E),C
(A,B)	0.00	2.50
((D, F), E),C	**2.50**	0.00

- The minimum distance of 2.5 is the result of the following computation

$$d_{(((D,F),E),C)\to(A,B)} = \min\left(d_{DA}, d_{DB}, d_{FA}, d_{FB}, d_{EA}, d_{EB}, d_{CA}, d_{CB}\right)$$

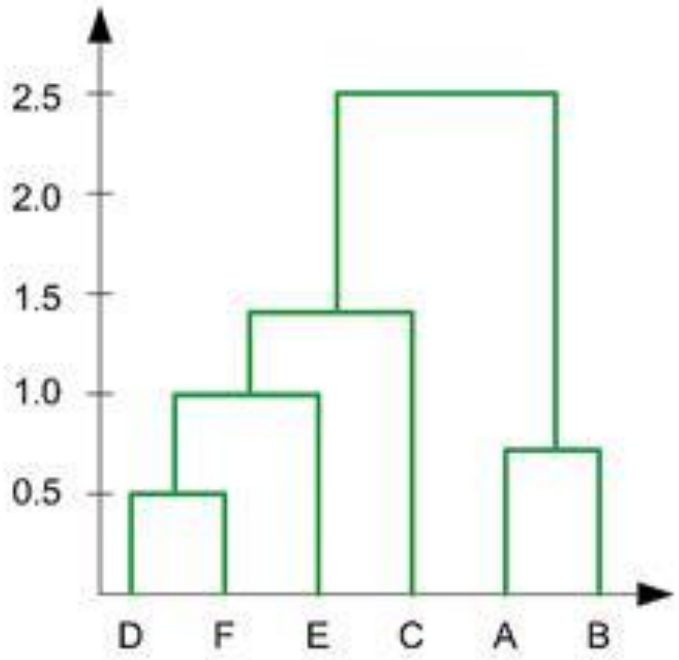
$$d_{(((D,F),E),C)\to(A,B)} = \min\left(3.61, 2.92, 3.20, 2.50, 4.24, 3.54, 5.66, 4.95\right) = 2.50$$

Numerical Example

- Now if we merge the remaining two clusters, we will get only single cluster contain the whole 6 objects. Thus, our computation is finished.
- We summarize the results of computation as follow:
  1. In the beginning we have 6 clusters: A, B, C, D, E and F
  2. We merge cluster D and F into cluster (D, F) at distance **0.50**
  3. We merge cluster A and cluster B into (A, B) at distance **0.71**
  4. We merge cluster E and (D, F) into ((D, F), E) at distance **1.00**
  5. We merge cluster ((D, F), E) and C into (((D, F), E), C) at distance **1.41**
  6. We merge cluster (((D, F), E), C) and (A, B) into ((((D, F), E), C), (A, B)) at distance **2.50**
- The last cluster contain all the objects, thus conclude the computation

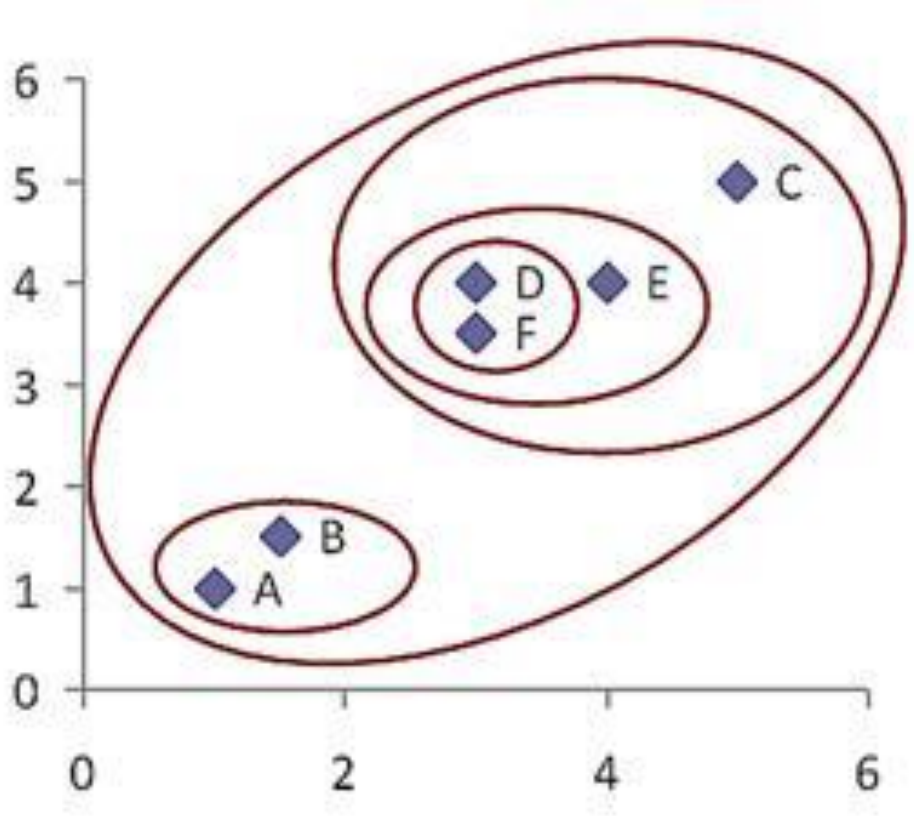
Final Clusters

- Using this information, we can now draw the final results of a dendogram. The dendogram is drawn based on the distances to merge the clusters above.



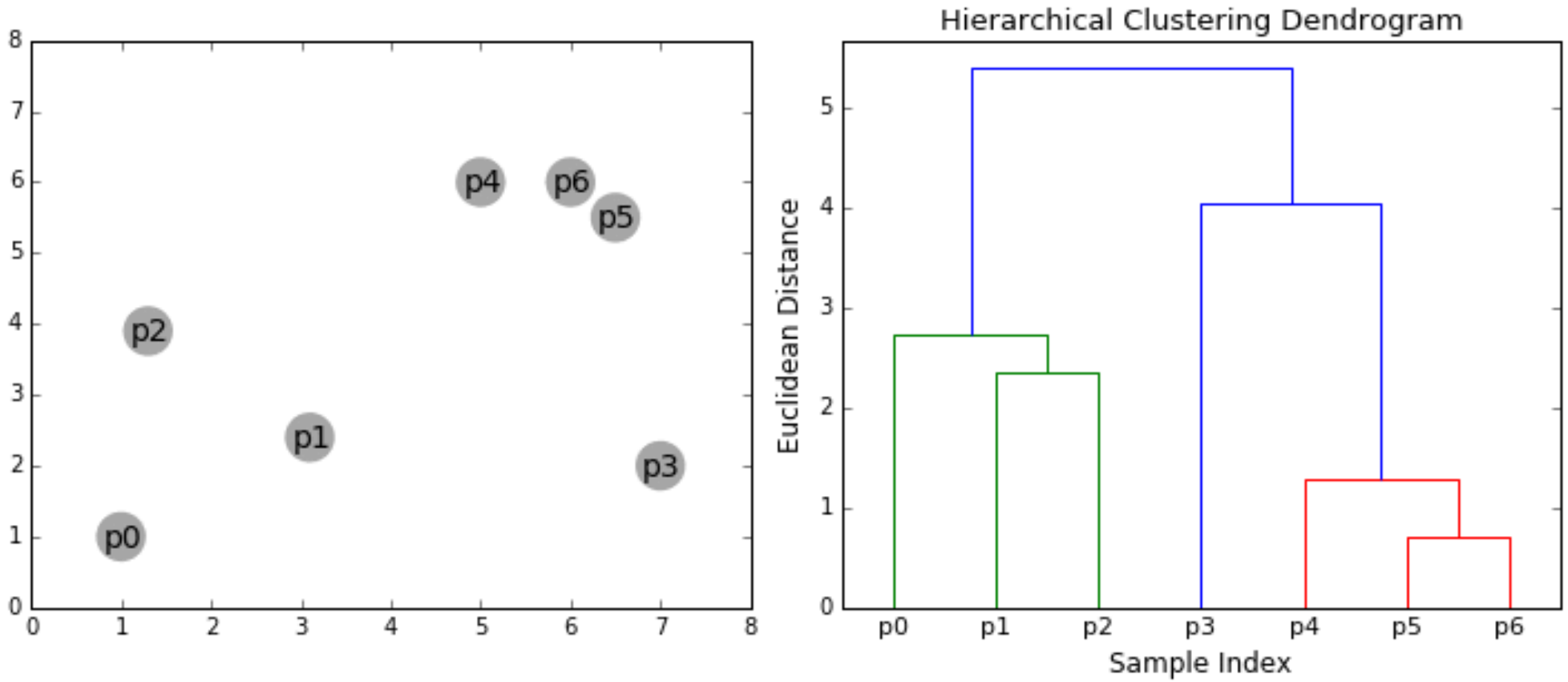
Final Clusters

- The hierarchy is given as (((D, F), E),C), (A,B). We can also plot the clustering hierarchy into XY space



	X1	X2
A	1	1
B	1.5	1.5
C	5	5
D	3	4
E	4	4
F	3	3.5

Animation

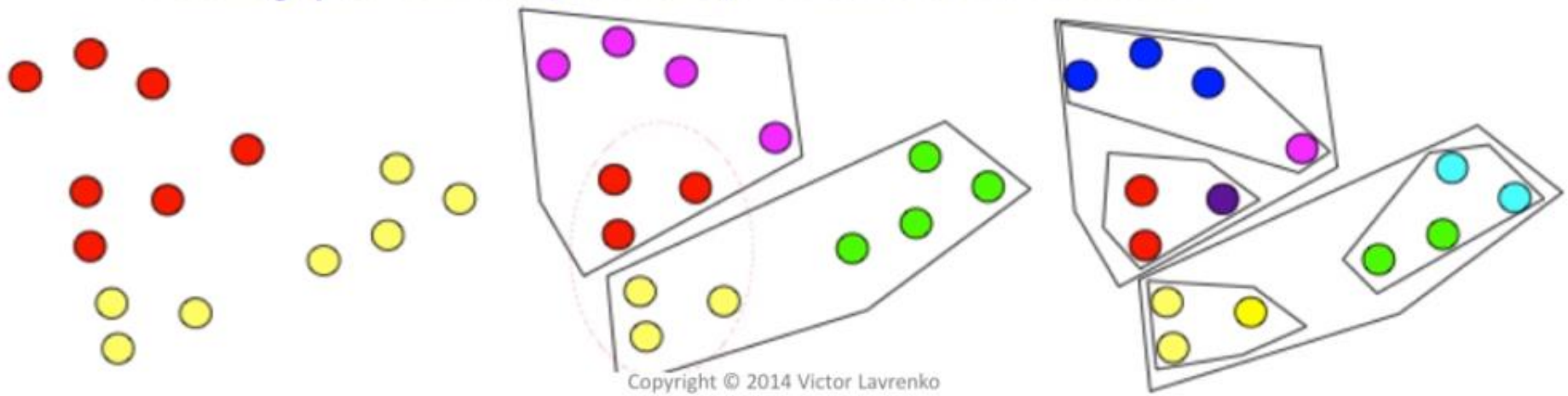

p0
p1
p2
p3
p4
p5
p6
Hierarchical Clustering Dendrogram
Euclidean Distance
Sample Index

Divisive Approach (Top Down)

- We start with assumption that all objects are group into a single group and then we split the group into two recursively until each group consists of a single object.
- One possible way to perform divisive approach is to apply hierarchical k-Means.

Hierarchical K-means

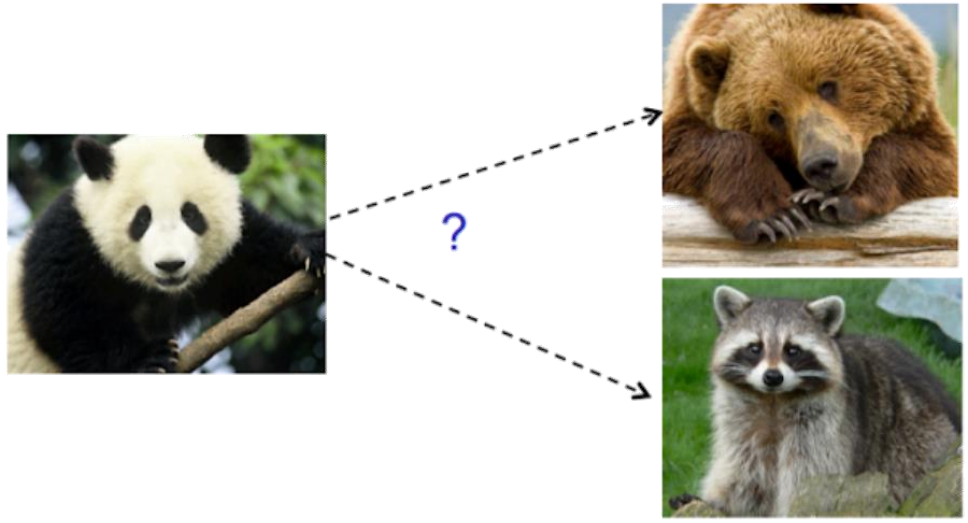
- Top-down approach:
  - run K-means algorithm on the original data $x_1 \ldots x_n$
  - for each of the resulting clusters c_i: $i = 1 \ldots K$
    - recursively run K-means on points in c_i
- Fast: recursive calls operate on a slice: $O(Knd \log_K n)$
- Greedy: can't cross boundaries imposed by top levels
  - nearby points may end up in different clusters



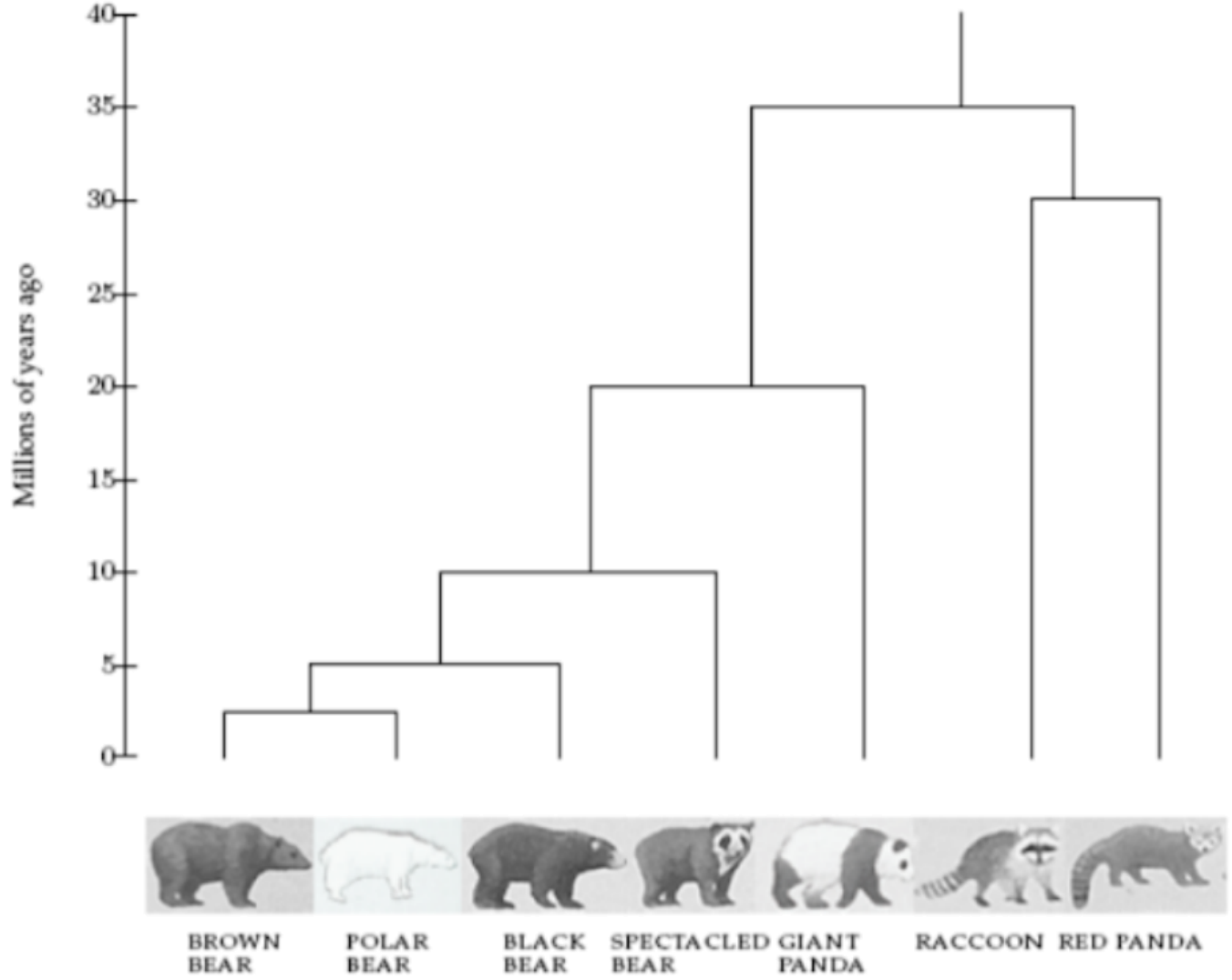
Copyright © 2014 Victor Lavrenko

Charting Evolution through Phylogenetic Trees

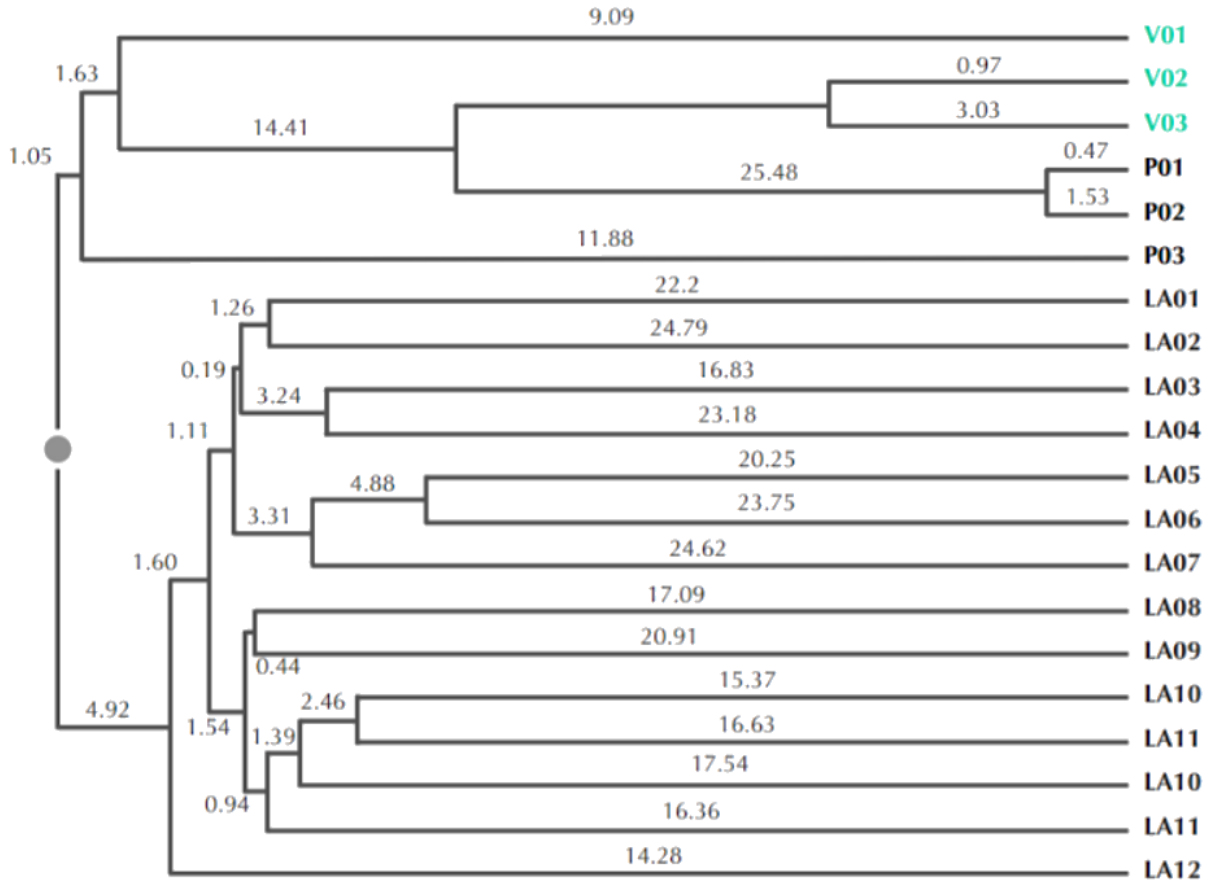
- Nowadays, we can use DNA sequencing and hierarchical clustering to find the phylogenetic tree of animal evolution:
  - Generate the DNA sequences
  - Calculate the edit distance between all sequences.
  - Calculate the DNA similarities based on the edit distances.
  - Construct the phylogenetic tree.



Charting Evolution through Phylogenetic Trees

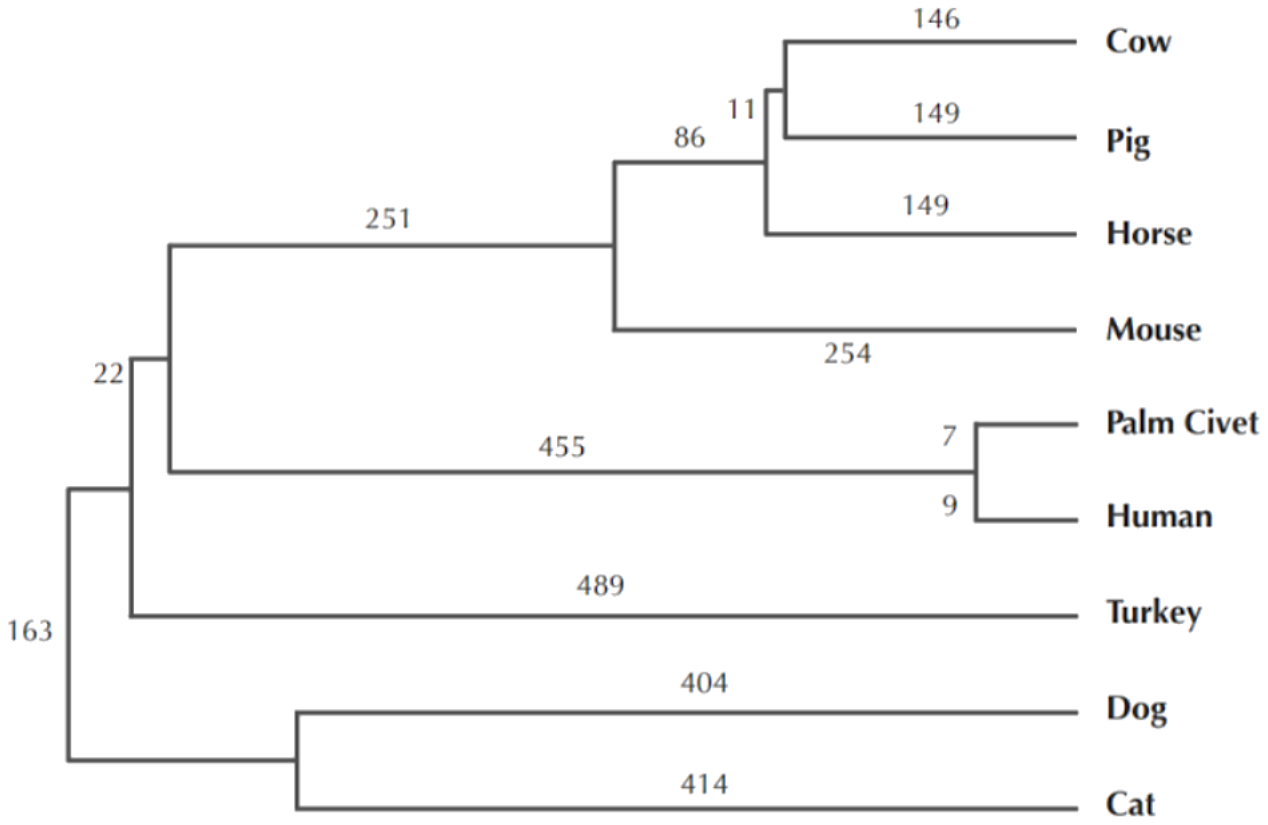


Tracking Viruses through Phylogenetic Trees



Tracking Viruses through Phylogenetic Trees

Study was also done for finding the animal that gave the humans the SARS virus:



*"With the data at hand, we see how the virus used different hosts, moving from **bat to human to civet**, in that order. So the civets actually got SARS **from humans**."*

Happy Pongal

nisi ut aliquip ex ea commodo consequat.
Lorem ipsum dolor sit amet, consectetur adipiscing elit,