University of Southern Denmark
IMADA
Ricardo Campello

DM583: Data Mining

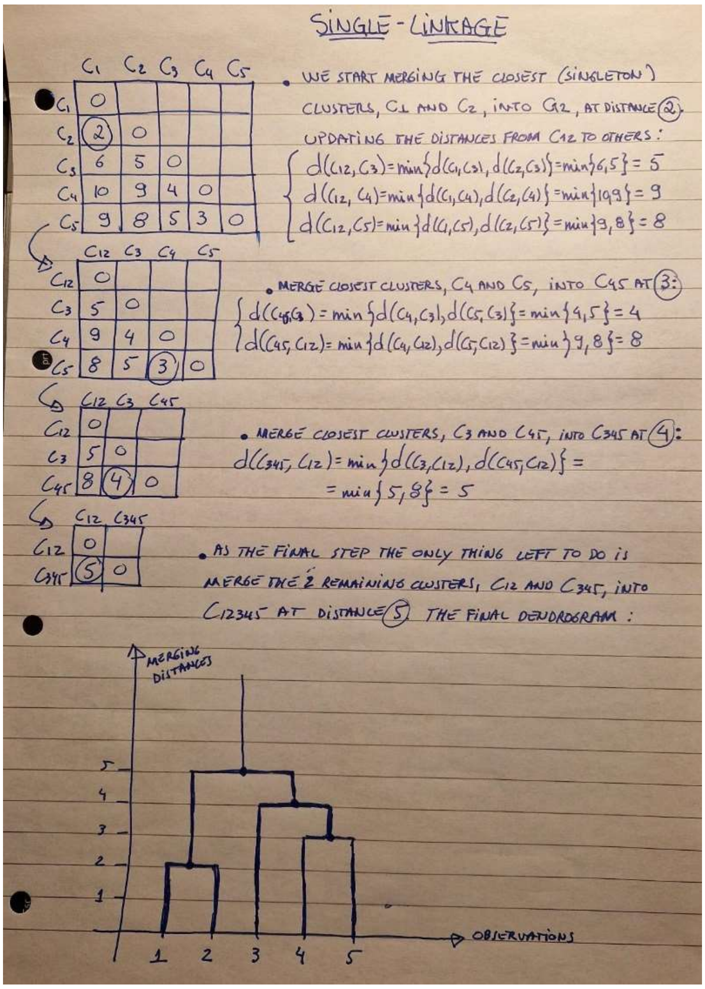
Exercise 4: Hierarchical Agglomerative Clustering (HAC)

Exercise 4-1 Single- and Complete-Linkage

a) Perform *Single-Linkage* step-by-step on the following distance matrix with pairwise distances between 5 data objects ([1], [2], …, [5]), then draw the final dendrogram:

$$
D = \begin{array}{c|ccccc}
 & [1] & [2] & [3] & [4] & [5] \\
\hline
[1] & 0 & 2 & 6 & 10 & 9 \\
[2] & 2 & 0 & 5 & 9 & 8 \\
[3] & 6 & 5 & 0 & 4 & 5 \\
[4] & 10 & 9 & 4 & 0 & 3 \\
[5] & 9 & 8 & 5 & 3 & 0
\end{array}
$$

Proposed Solution:



SINGLE - LINKAGE

$$
\begin{array}{c|ccccc}
 & C_1 & C_2 & C_3 & C_4 & C_5 \\
\hline
C_1 & 0 & & & & \\
C_2 & \textcircled{2} & 0 & & & \\
C_3 & 6 & 5 & 0 & & \\
C_4 & 10 & 9 & 4 & 0 & \\
C_5 & 9 & 8 & 5 & 3 & 0
\end{array}
$$

- WE START MERGING THE CLOSEST (SINGLETON) CLUSTERS, C_1 AND C_2, INTO C_{12}, AT DISTANCE $\textcircled{2}$. UPDATING THE DISTANCES FROM C_{12} TO OTHERS:

$$
\begin{cases}
d(C_{12},C_3)=\min\{d(C_1,C_3),d(C_2,C_3)\}=\min\{6,5\}=5 \\
d(C_{12},C_4)=\min\{d(C_1,C_4),d(C_2,C_4)\}=\min\{10,9\}=9 \\
d(C_{12},C_5)=\min\{d(C_1,C_5),d(C_2,C_5)\}=\min\{9,8\}=8
\end{cases}
$$

$$
\begin{array}{c|cccc}
 & C_{12} & C_3 & C_4 & C_5 \\
\hline
C_{12} & 0 & & & \\
C_3 & 5 & 0 & & \\
C_4 & 9 & 4 & 0 & \\
C_5 & 8 & 5 & \textcircled{3} & 0
\end{array}
$$

- MERGE CLOSEST CLUSTERS, C_4 AND C_5, INTO C_{45} AT $\textcircled{3}$:

$$
\begin{cases}
d(C_{45},C_3)=\min\{d(C_4,C_3),d(C_5,C_3)\}=\min\{4,5\}=4 \\
d(C_{45},C_{12})=\min\{d(C_4,C_{12}),d(C_5,C_{12})\}=\min\{9,8\}=8
\end{cases}
$$

$$
\begin{array}{c|ccc}
 & C_{12} & C_3 & C_{45} \\
\hline
C_{12} & 0 & & \\
C_3 & 5 & 0 & \\
C_{45} & 8 & \textcircled{4} & 0
\end{array}
$$

- MERGE CLOSEST CLUSTERS, C_3 AND C_{45}, INTO C_{345} AT $\textcircled{4}$:

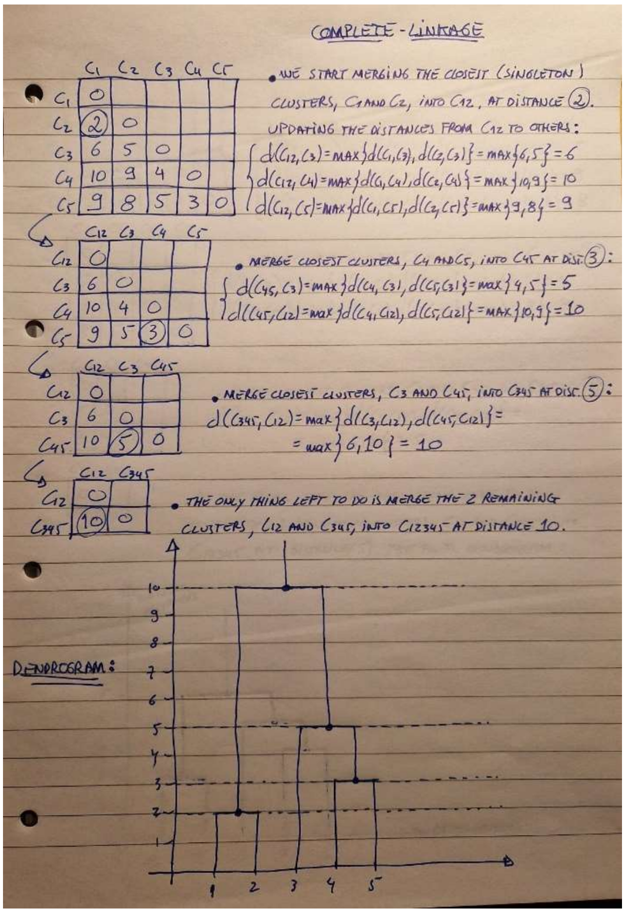
$$
d(C_{345},C_{12})=\min\{d(C_3,C_{12}),d(C_{45},C_{12})\}=
$$
$$
=\min\{5,8\}=5
$$

$$
\begin{array}{c|cc}
 & C_{12} & C_{345} \\
\hline
C_{12} & 0 & \\
C_{345} & \textcircled{5} & 0
\end{array}
$$

- AS THE FINAL STEP THE ONLY THING LEFT TO DO IS MERGE THE 2 REMAINING CLUSTERS, C_{12} AND C_{345}, INTO C_{12345} AT DISTANCE $\textcircled{5}$. THE FINAL DENDROGRAM:

b) Repeat item (a) but now for *Complete-Linkage*.

Proposed Solution:



COMPLETE - LINKAGE

	C_1	C_2	C_3	C_4	C_5
C_1	0				
C_2	2	0			
C_3	6	5	0		
C_4	10	9	4	0	
C_5	9	8	5	3	0

- WE START MERGING THE CLOSEST (SINGLETON) CLUSTERS, C_1 AND C_2, INTO C_{12}, AT DISTANCE 2. UPDATING THE DISTANCES FROM C_{12} TO OTHERS:

$$\begin{cases} d(C_{12},C_3)=\text{MAX}\{d(C_1,C_3),d(C_2,C_3)\}=\text{MAX}\{6,5\}=6 \\ d(C_{12},C_4)=\text{MAX}\{d(C_1,C_4),d(C_2,C_4)\}=\text{MAX}\{10,9\}=10 \\ d(C_{12},C_5)=\text{MAX}\{d(C_1,C_5),d(C_2,C_5)\}=\text{MAX}\{9,8\}=9 \end{cases}$$

	C_{12}	C_3	C_4	C_5
C_{12}	0			
C_3	6	0		
C_4	10	4	0	
C_5	9	5	3	0

- MERGE CLOSEST CLUSTERS, C_4 AND C_5, INTO C_{45} AT DIST. 3:

$$\begin{cases} d(C_{45},C_3)=\max\{d(C_4,C_3),d(C_5,C_3)\}=\max\{4,5\}=5 \\ d(C_{45},C_{12})=\max\{d(C_4,C_{12}),d(C_5,C_{12})\}=\text{MAX}\{10,9\}=10 \end{cases}$$

	C_{12}	C_3	C_{45}
C_{12}	0		
C_3	6	0	
C_{45}	10	5	0

- MERGE CLOSEST CLUSTERS, C_3 AND C_{45}, INTO C_{345} AT DIST. 5:

$$d(C_{345},C_{12})=\max\{d(C_3,C_{12}),d(C_{45},C_{12})\}= \\ =\max\{6,10\}=10$$

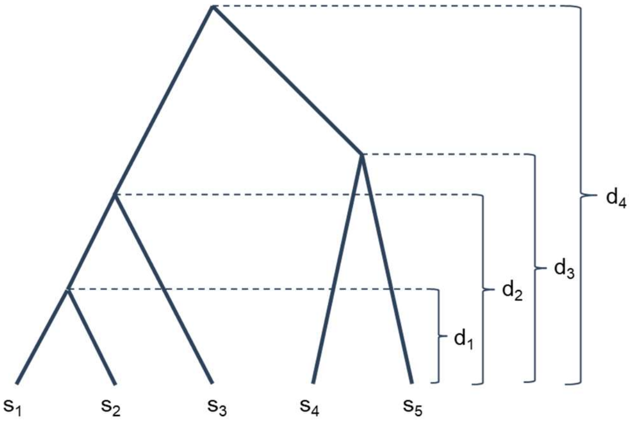
	C_{12}	C_{345}
C_{12}	0	
C_{345}	10	0

- THE ONLY THING LEFT TO DO IS MERGE THE 2 REMAINING CLUSTERS, C_{12} AND C_{345}, INTO C_{12345} AT DISTANCE 10.

DENDROGRAM:

Exercise 4-2 Average-Linkage (UPGMA)

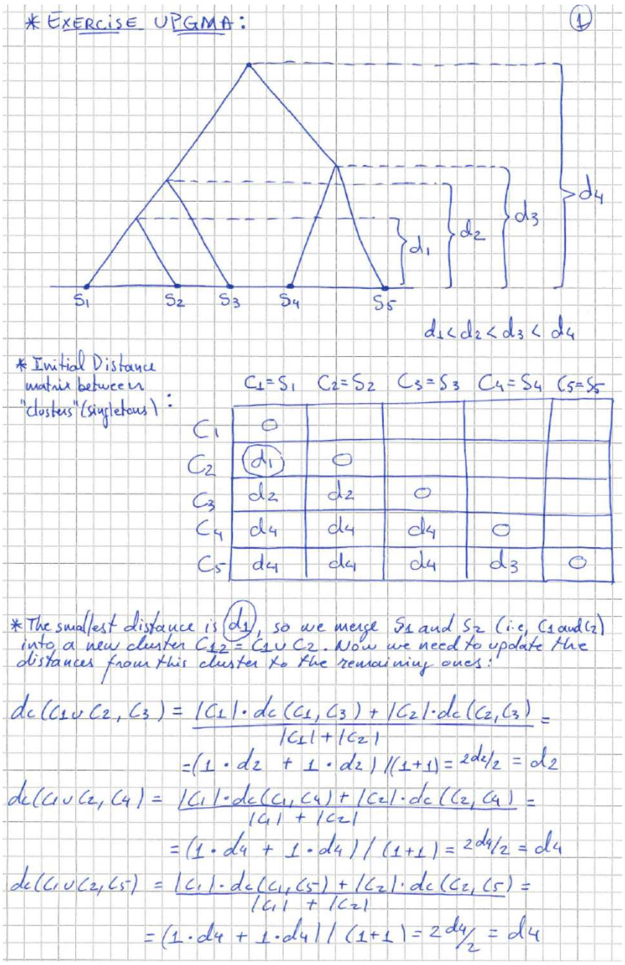
a) In addition to being a general-purpose algorithm for clustering problems, irrespective of any particular domain of application, certain properties of the UPGMA algorithm make this algorithm also suitable for applications in the specific task of constructing *rooted phylogenetic trees* in biology. A rooted phylogenetic tree is essentially an ancestor-descendant tree representing the ancestry relationships between a group of species and their common ancestors, based on some measure of evolutionary distance between them. For instance, consider the following *rooted phylogenetic tree* for a group of five species:



This tree suggests that there is a common former ancestor (the root node at the top) to all 5 existing species S1 through to S5, which has split into two different descendants, one of which is an ancestor for (S1, S2, S3) whereas the other one is an ancestor for S4 and S5. The latter split into S4 and S5 through evolution, whereas the former further branched up by first separating S3 from a common ancestor to S1 and S2, which eventually split into these two. The vertical measurements in the figure indicate the presumed evolutionary distance between groups of species and their more recent common ancestor: d1 is the evolutionary distance between (S1, S2) and their closest common ancestor, d2 is the distance between (S1, S2, S3) and their closest common ancestor, and so forth. For reasons that are beyond the scope of this exercise, if the tree and distances in the figure are an accurate representation of the evolutionary process in question, then UPGMA algorithm can be shown to be guaranteed to exactly recover the correct tree and the corresponding evolutionary distances to common ancestors assuming that pairwise distances between current (existing) species can be measured and are proportional to their evolutionary distances to their closest common ancestor. Show that this is indeed the case in the above example by manually computing the UPGMA algorithm on the following distance matrix between species, which assumes that the distance between any two species is precisely given by their evolutionary distance to their closest common ancestor, i.e. (hint: notice from the figure that d1 < d2 < d3 < d4):

	S_1	S_2	S_3	S_4	S_5
S_1	0				
S_2	d_1	0			
S_3	d_2	d_2	0		
S_4	d_4	d_4	d_4	0	
S_5	d_4	d_4	d_4	d_3	0

Proposed Solution:



* EXERCISE UPGMA: 1

d_4
d_3
d_2
d_1
S_1
S_2
S_3
S_4
S_5

$d_1 < d_2 < d_3 < d_4$

* Initial Distance matrix between "clusters" (singletons):

	$C_1 = S_1$	$C_2 = S_2$	$C_3 = S_3$	$C_4 = S_4$	$C_5 = S_5$
C_1	0				
C_2	d_1	0			
C_3	d_2	d_2	0		
C_4	d_4	d_4	d_4	0	
C_5	d_4	d_4	d_4	d_3	0

* The smallest distance is d_1, so we merge S_1 and S_2 (i.e, C_1 and C_2) into a new cluster $C_{12} = C_1 \cup C_2$. Now we need to update the distances from this cluster to the remaining ones:

$$d_c(C_1 \cup C_2, C_3) = \frac{|C_1| \cdot d_c(C_1, C_3) + |C_2| \cdot d_c(C_2, C_3)}{|C_1| + |C_2|} =$$

$$= (1 \cdot d_2 + 1 \cdot d_2) / (1+1) = 2d_2/2 = d_2$$

$$d_c(C_1 \cup C_2, C_4) = \frac{|C_1| \cdot d_c(C_1, C_4) + |C_2| \cdot d_c(C_2, C_4)}{|C_1| + |C_2|} =$$

$$= (1 \cdot d_4 + 1 \cdot d_4) / (1+1) = 2d_4/2 = d_4$$

$$d_c(C_1 \cup C_2, C_5) = \frac{|C_1| \cdot d_c(C_1, C_5) + |C_2| \cdot d_c(C_2, C_5)}{|C_1| + |C_2|} =$$

$$= (1 \cdot d_4 + 1 \cdot d_4) / (1+1) = 2 d_4/2 = d_4$$

②

*The new distance matrix is therefere:

$$\begin{array}{c|cccc} & C_{12} & C_3 & C_4 & C_5 \\ \hline C_{12} & 0 & & & \\ C_3 & \boxed{d_2} & 0 & & \\ C_4 & d_4 & d_4 & 0 & \\ C_5 & d_4 & d_4 & d_3 & 0 \end{array}$$

The smallest value is $\boxed{d_2}$, so we merge C_{12} and C_3, then update the distances from $C_{123} = C_{12} \cup C_3$ to others.

$$d(C_{12} \cup C_3, C_4) = \frac{|C_{12}| \cdot d_c(C_{12}, C_4) + |C_3| \cdot d_c(C_3, C_4)}{|C_{12}| + |C_3|} = $$

$$= \frac{2 \cdot d_4 + 1 \cdot d_4}{2+1} = \frac{3 d_4}{3} = d_4$$

$$d(C_{12} \cup C_3, C_5) = \frac{|C_{12}| \cdot d_c(C_{12}, C_5) + |C_3| \cdot d_c(C_3, C_5)}{|C_{12}| + |C_3|} = $$

$$= \frac{2 \cdot d_4 + 1 \cdot d_4}{2+1} = \frac{3 \cdot d_4}{3} = d_4$$

*The new distance matrix is therefore:

$$\begin{array}{c|ccc} & C_{123} & C_4 & C_5 \\ \hline C_{123} & 0 & & \\ C_4 & d_4 & 0 & \\ C_5 & d_4 & \boxed{d_3} & 0 \end{array}$$

The smallest distance is $\boxed{d_3}$, so we merge C_4 and C_5 and compute the distances between $C_{45} = C_4 \cup C_5$ and the other cluster remaining:

$$d(C_4 \cup C_5, C_{123}) = \frac{|C_4| \cdot d_c(C_4, C_{123}) + |C_5| \cdot d_c(C_5, C_{123})}{|C_4| + |C_5|} = $$

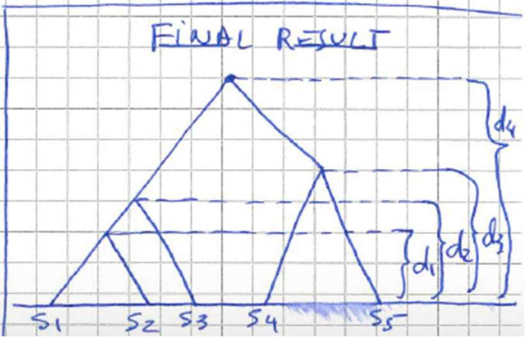
$$= \frac{1 \cdot d_4 + 1 \cdot d_4}{1+1} = \frac{2 d_4}{2} = d_4$$

*The new distance matrix:

$$\begin{array}{c|cc} & C_{123} & C_{45} \\ \hline C_{123} & 0 & \\ C_{45} & \boxed{d_4} & 0 \end{array}$$

*obviously we now can only merge C_{123} and C_4 with distance $\boxed{d_4}$

$\Rightarrow$

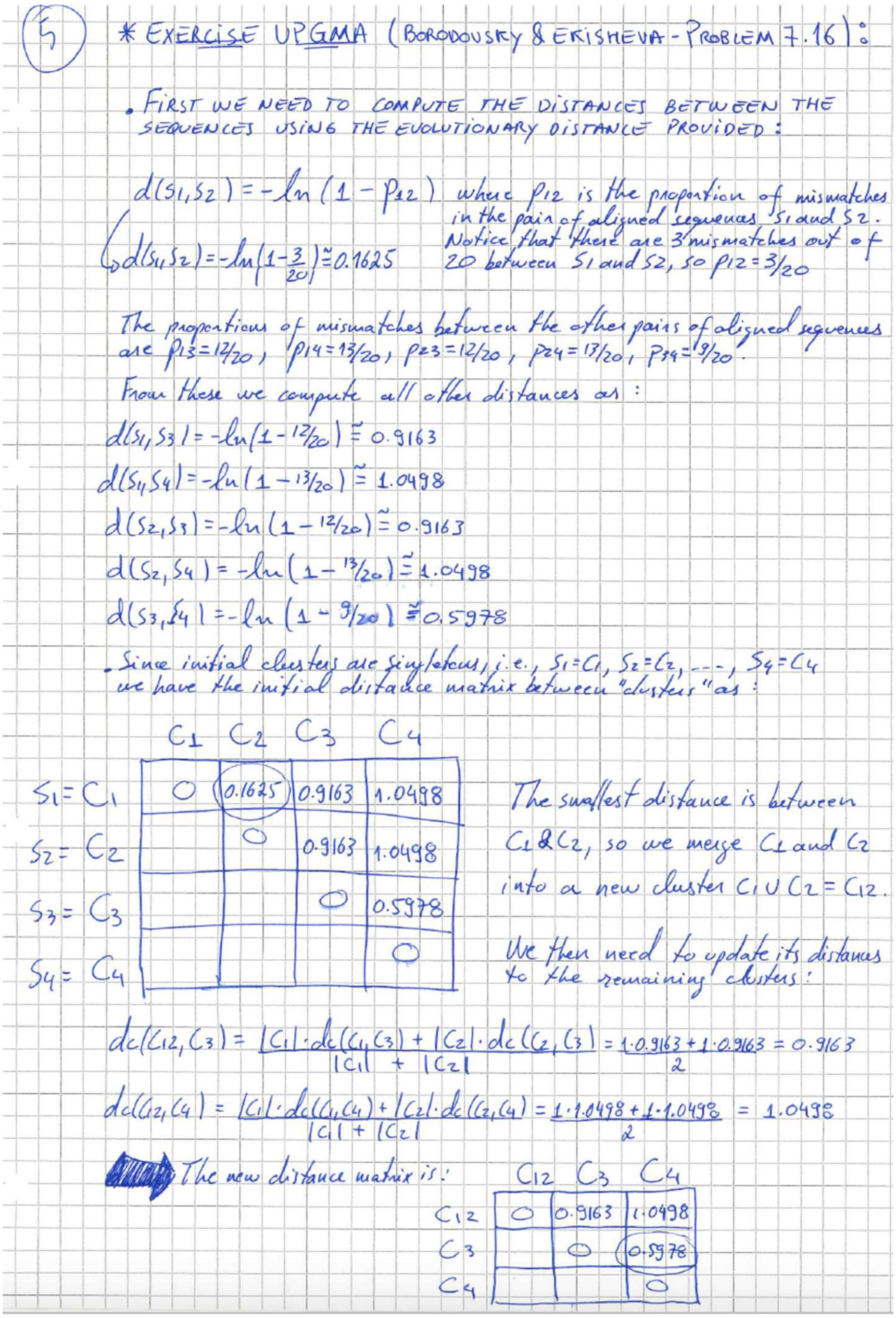


b) The pairwise distance between two species (as a proxy for the evolutionary distance to their common ancestor) can be estimated by measuring dissimilarity between certain molecular/biological sequences (genes or proteins). One possibility is to measure the difference between two aligned segments of amino acid sequences corresponding to a key protein family of interest, such as the following ungapped multiple alignment of the fragments of *Cytochrome C* from four different species, namely, *Rickettsia Conorii*, *Rickettsia Prowazekii*, *Bradyrhizobium Japonicum* and *Agrobacterium Tumefaciens*, as shown below (in top-down order):

NIPELMKTANADNGREIAKK
NIQELMKTANANHGREIAKK
PIEKLLQTASVEKGAAAAKK
PIAKLLASADAAKGEAVFKK

Use the evolutionary distance defined by the formula $dij = -\ln(1 - pij)$, where pij is the fraction of mismatches in the pairwise alignment of sequences i and j, to build a rooted phylogenetic tree for the given sequences by the UPGMA. Note: This problem is originally from [M. Borodovsky & S. Ekisheva "Biological Sequence Analysis", 2006].

Proposed Solution:



5 * EXERCISE UPGMA (BORODOUSKY & EKISHEVA - PROBLEM 7.16):

- FIRST WE NEED TO COMPUTE THE DISTANCES BETWEEN THE SEQUENCES USING THE EVOLUTIONARY DISTANCE PROVIDED:

$d(S_1,S_2) = -\ln(1 - p_{12})$ where p_{12} is the proportion of mismatches in the pair of aligned sequences S_1 and S_2. Notice that there are 3 mismatches out of 20 between S_1 and S_2, so $p_{12} = 3/20$

$\hookrightarrow d(S_1,S_2) = -\ln\left(1 - \frac{3}{20}\right) \cong 0.1625$

The proportions of mismatches between the other pairs of aligned sequences are $p_{13} = 12/20$, $p_{14} = 13/20$, $p_{23} = 12/20$, $p_{24} = 13/20$, $p_{34} = 9/20$.

From these we compute all other distances as:

$d(S_1,S_3) = -\ln(1 - 12/20) \cong 0.9163$

$d(S_1,S_4) = -\ln(1 - 13/20) \cong 1.0498$

$d(S_2,S_3) = -\ln(1 - 12/20) \cong 0.9163$

$d(S_2,S_4) = -\ln(1 - 13/20) \cong 1.0498$

$d(S_3,S_4) = -\ln(1 - 9/20) \cong 0.5978$

- Since initial clusters are singletons, i.e., $S_1 = C_1$, $S_2 = C_2$, ..., $S_4 = C_4$ we have the initial distance matrix between "clusters" as:

	C_1	C_2	C_3	C_4
$S_1 = C_1$	0	0.1625	0.9163	1.0498
$S_2 = C_2$		0	0.9163	1.0498
$S_3 = C_3$			0	0.5978
$S_4 = C_4$				0

The smallest distance is between C_1 & C_2, so we merge C_1 and C_2 into a new cluster $C_1 \cup C_2 = C_{12}$.

We then need to update its distances to the remaining clusters:

$$d_c(C_{12},C_3) = \frac{|C_1| \cdot d_c(C_1,C_3) + |C_2| \cdot d_c(C_2,C_3)}{|C_1| + |C_2|} = \frac{1 \cdot 0.9163 + 1 \cdot 0.9163}{2} = 0.9163$$

$$d_c(C_{12},C_4) = \frac{|C_1| \cdot d_c(C_1,C_4) + |C_2| \cdot d_c(C_2,C_4)}{|C_1| + |C_2|} = \frac{1 \cdot 1.0498 + 1 \cdot 1.0498}{2} = 1.0498$$

⇨ The new distance matrix is:

	C_{12}	C_3	C_4
C_{12}	0	0.9163	1.0498
C_3		0	0.5978
C_4			0

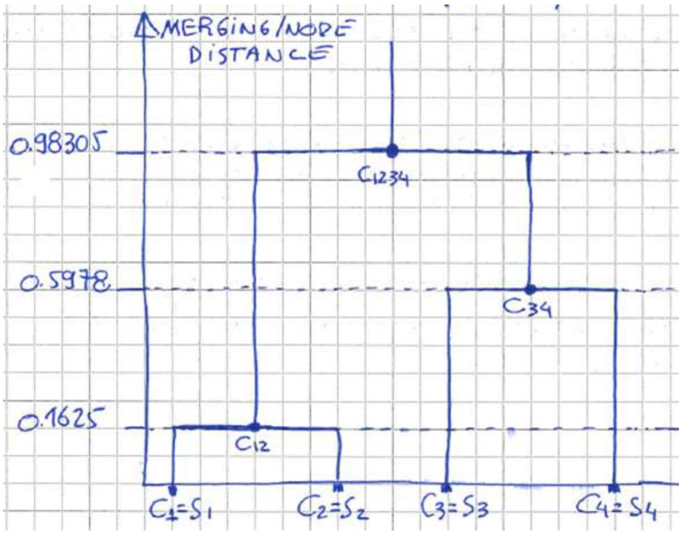
(6) Now, the smallest distance is between C_3 and C_4, so we merge them into a new cluster $C_{34} = C_3 \cup C_4$ and update distances:

$$d_c(C_3, C_{12}) = \frac{|C_3| \cdot d_c(C_3, C_{12}) + |C_4| \cdot d_c(C_4, C_{12})}{|C_3| + |C_4|} =$$

$$= \frac{1 \cdot 0.9163 + 1 \cdot 1.0498}{2} = 0.98305$$

Now, there is nothing else left to do other than merging C_{34} and C_{12} into the final single cluster at distance 0.98305.

The cluster tree is therefore as follows:



c) If pairwise distance estimates fail to satisfy certain critical assumptions as roughly outlined in item (a), then the evolutionary tree distances computed by UPGMA (i.e., the heights of mergers along the dendrogram vertical scale) are not necessarily equal or proportional to them, as they were in item (a). Show that this is the case in the example in item (b).

Proposed Solution:

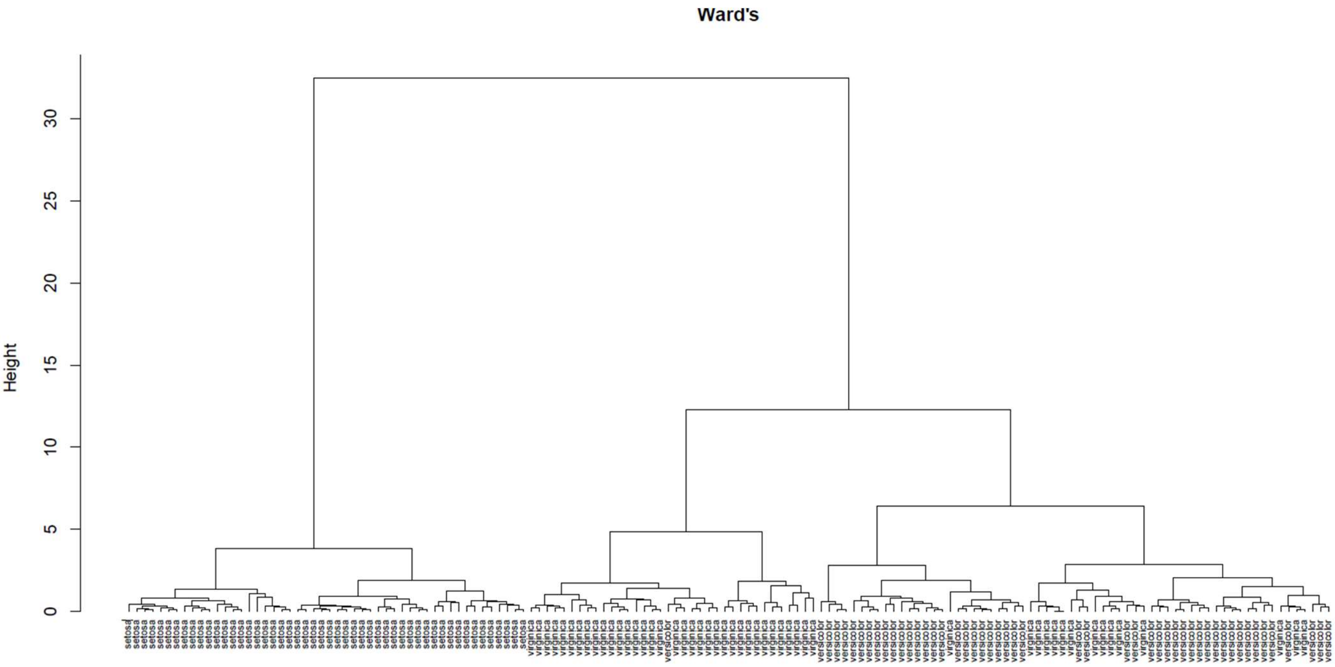
For example, note from the resulting phylogenetic tree (cluster dendrogram) item (b) that the computed evolutionary/tree distances between sequences respect the following equalities: d(S1,S3) = d(S2,S3) = d(S1,S4) = d(S2,S4). The original distances clearly don't respect all of these equalities, but rather d(S1,S3) = d(S2,S3) ≠ d(S1,S4) = d(S2,S4).

Exercise 4-3 HAC in R and Dendrogram Interpretation

a) Compute and plot Ward's dendrogram for the `iris` data set using function `hclust()` from the `stats` package available in base R, with Euclidean distance, which can be readily computed using function `dist()`. Then perform a horizontal cut through the dendrogram to obtain a partition with 2 clusters using function `cutree()`. Note: Use only the numerical variables `iris[1:4]` for the clustering. The 5th variable, `iris$Species`, can only be used to (optionally) display the class labels at the bottom of the dendrogram.

Proposed Solution:

```
# Euclidean distances must be provided so that Ward's has a mathematical interpretation:
DMatrix <- dist(iris[1:4])
# These distances are squared internally in option "ward.D2": this option implements the
# original Ward's (1963) criterion, where the dissimilarities are squared before cluster updating:
Ward_iris <- hclust(DMatrix, method = "ward.D2")
# Cut through dendrogram (partition with two clusters):
partition_2_clusters <- cutree(tree = Ward_iris, k = 2)
# Plot complete dendrogram (with class labels at the bottom, just for evaluation):
plot(Ward_iris, main = "Ward's", xlab = "", sub = "", hang = -1, labels = iris$Species, cex = 0.6)
```

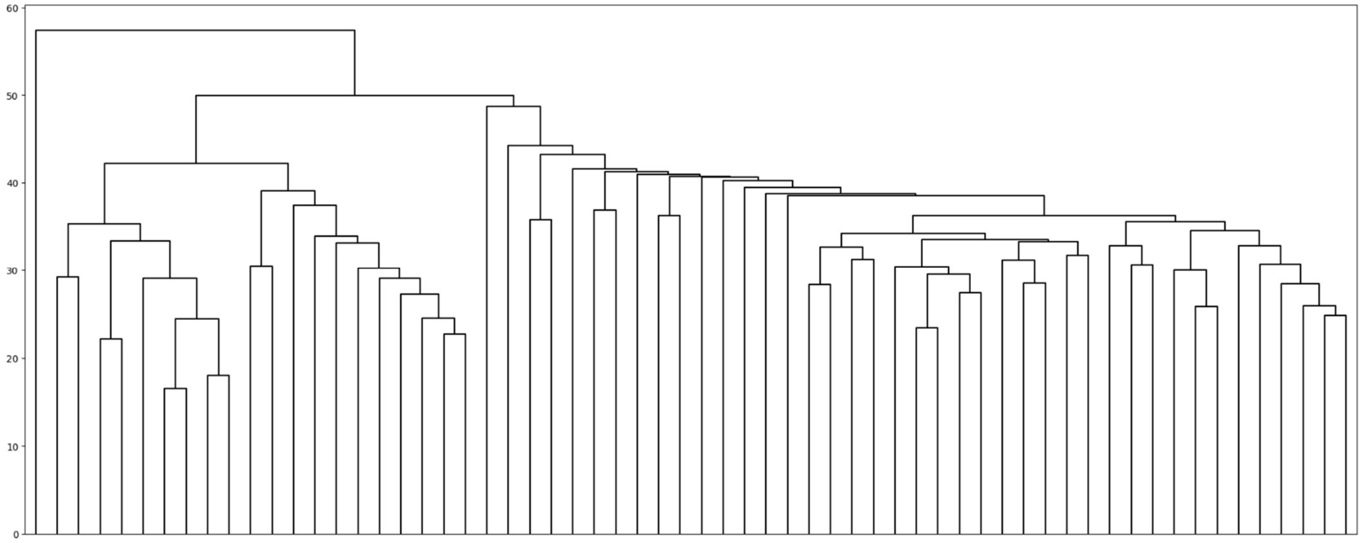


b) Repeat item (a) for Single-Linkage (SL), Complete-Linkage (CL), and Average-Linkage (AL).

Proposed Solution:

Just replace `method = "ward.D2"` with `method = "single"`, `method = "complete"` or `method = "average"` in the code of item (a) to obtain the results for SL, CL and AL, respectively.

c) The following figure displays the dendrogram resulting from the application of a HAC algorithm (AL) to a real-world dataset consisting of gene-expression measurements (from hundreds of genes) from a particular type of tissue sampled from a few tens of different patients, which have been clustered. Interpret the dendrogram.



Proposed Solution:

There is a noticeable (global) outlier, which is the leftmost observation in the dendrogram, as well as two major clusters, namely: (i) a cluster on the right, which is larger in terms of cluster size, i.e., number of observations. This cluster is overall sparser, based on the higher merging distances in the dendrogram scale, and contains no obvious structure in terms of sub-clusters. A close look suggests that this cluster seems to consist of a core, not so sparse cluster along with a sparser cloud of locally outlying observations (or a less dense vicinity of some sort, e.g., a peripheral ring or comet-shaped tail) joining at higher merging distances; (ii) a cluster on the left, which clearly contains two prominent sub-clusters. If a choice must be made, e.g., if a flat partition is supposed to be extracted by cutting through the dendrogram, it is not visually obvious based on the merging distances whether this cluster is better reported/interpreted as a single cluster or should be split into two. Quantitative criteria such as e.g. the Silhouette could be used to assist the choice, but typically the interpretation should be better performed by also taking into account domain knowledge from an expert (in this case, a biologist or bioinformatician).