Chair of Decision Sciences and Systems
TUM School of Computation, Information and Technology
Technical University of Munich

Business Analytics & Machine Learning
Homework sheet 8: Clustering – Solution

Prof. Dr. Martin Bichler, Prof. Dr. Jalal Etesami
Julius Durmann, Markus Ewert, Johannes Knörr, Yutong Chao

Exercise H8.1 *Hierarchical clustering*

You are given the following dataset:

i	x_i	y_i
0	0	0
1	1	5
2	4	2
3	-1	4
4	-2	-3
5	3	4
6	1	-5
7	0	1
8	-3	0
9	-1	-1

Table 1 Dataset

a) Complete the missing entries in the distance matrix below based on the L1 norm (Manhattan distance)

$$D_1 = \begin{bmatrix} 0 & \bigcirc & 6 & 5 & 5 & 7 & 6 & 1 & 3 & 2 \\ & 0 & 6 & 3 & 11 & 3 & \bigcirc & 5 & 9 & 8 \\ & & 0 & 7 & 11 & \bigcirc & 10 & 5 & 9 & 8 \\ & & & 0 & 8 & 4 & 11 & 4 & 6 & \bigcirc \\ & & & & 0 & 12 & 5 & 6 & 4 & 3 \\ & & & & & 0 & 11 & \bigcirc & 10 & 9 \\ & & & & & & 0 & 7 & 9 & 6 \\ & & & & & & & 0 & 4 & 3 \\ & & & & & & & & 0 & 3 \\ & & & & & & & & & 0 \end{bmatrix}$$

b) Complete the missing entries in the distance matrix below based on the L2 norm (Euclidean distance)

$$D_2 = \begin{bmatrix} 0 & 5.10 & 4.47 & 4.12 & 3.61 & 5 & 5.10 & 1 & 3 & \bigcirc \\ & 0 & 4.24 & 2.24 & \bigcirc & 2.24 & 10 & 4.12 & 6.40 & 6.32 \\ & & 0 & \bigcirc & 7.81 & 2.24 & 7.62 & 4.12 & 7.28 & 5.83 \\ & & & 0 & 7.07 & 4 & 9.22 & 3.16 & 4.47 & \bigcirc \\ & & & & 0 & 8.60 & 3.61 & 4.47 & 3.16 & 2.24 \\ & & & & & 0 & 9.22 & 4.24 & 7.21 & 6.40 \\ & & & & & & 0 & 6.08 & \bigcirc & 4.47 \\ & & & & & & & 0 & 3.16 & 2.24 \\ & & & & & & & & 0 & 2.24 \\ & & & & & & & & & 0 \end{bmatrix}$$

c) Perform bottom-up (agglomerative) hierarchical clustering using the Manhattan distance d_1 as distance measure. Use *complete-linkage clustering*, i.e. the distance between two sets of observations A, B is defined as $\max_{a \in A, b \in B} d(a, b)$. As tiebreaker rule, merge clusters in the order of their label i.

d) Draw the dendrogram. Which is a reasonable number of clusters in this example? Discuss your reasoning.

Solution

a) The Manhattan distance between two vectors $a, b \in \mathbb{R}^2$ is defined as

$$d_1(a, b) = \|a - b\|_1 = |x_a - x_b| + |y_a - y_b|.$$

For example,

$$\begin{aligned} d_1(0, 1) &= |x_0 - x_1| + |y_0 - y_1| \\ &= |0 - 1| + |0 - 5| \\ &= 6. \end{aligned}$$

With equivalent calculations, we obtain the following distance matrix:

$$D_1 = \begin{bmatrix} 0 & \mathbf{6} & 6 & 5 & 5 & 7 & 6 & 1 & 3 & 2 \\ & 0 & 6 & 3 & 11 & 3 & \mathbf{10} & 5 & 9 & 8 \\ & & 0 & 7 & 11 & \mathbf{3} & 10 & 5 & 9 & 8 \\ & & & 0 & 8 & 4 & 11 & 4 & 6 & \mathbf{5} \\ & & & & 0 & 12 & 5 & 6 & 4 & 3 \\ & & & & & 0 & 11 & \mathbf{6} & 10 & 9 \\ & & & & & & 0 & 7 & 9 & 6 \\ & & & & & & & 0 & 4 & 3 \\ & & & & & & & & 0 & 3 \\ & & & & & & & & & 0 \end{bmatrix}$$

b) The Euclidean distance between two vectors $a, b \in \mathbb{R}^2$ is defined as

$$d_2(a, b) = \|a - b\|_2 = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2}.$$

For example,

$$\begin{aligned} d_2(0,9) &= \sqrt{(x_0 - x_9)^2 + (y_0 - y_9)^2} \\ &= \sqrt{(0-(-1))^2 + (0-(-1))^2} \qquad\qquad = \sqrt{2} \approx 1.41. \end{aligned}$$

With equivalent calculations, we obtain the following distance matrix:

$$D_2 = \begin{bmatrix} 0 & 5.10 & 4.47 & 4.12 & 3.61 & 5 & 5.10 & 1 & 3 & \mathbf{1.41} \\ & 0 & 4.24 & 2.24 & \mathbf{8.54} & 2.24 & 10 & 4.12 & 6.40 & 6.32 \\ & & 0 & \mathbf{5.39} & 7.81 & 2.24 & 7.62 & 4.12 & 7.28 & 5.83 \\ & & & 0 & 7.07 & 4 & 9.22 & 3.16 & 4.47 & \mathbf{5} \\ & & & & 0 & 8.60 & 3.61 & 4.47 & 3.16 & 2.24 \\ & & & & & 0 & 9.22 & 4.24 & 7.21 & 6.40 \\ & & & & & & 0 & 6.08 & \mathbf{6.40} & 4.47 \\ & & & & & & & 0 & 3.16 & 2.24 \\ & & & & & & & & 0 & 2.24 \\ & & & & & & & & & 0 \end{bmatrix}$$

c) For bottom-up or agglomerative hierarchical clustering, we first assign each point to an individual cluster. We then iteratively compute the smallest distance between two clusters and merge them together.

1) From the distance matrix D_1, we obtain the smallest distance between two clusters / points as $d_1(0,7) = 1$. We merge points 0 and 7 to form a single cluster.

   We update the distance matrix (which now reflects distances between *clusters* and not points). We use *complete-linkage clustering*, meaning the distance of the new cluster (0,7) to another point i is $d_1((0,7),i) = \max(d_1(0,i), d_1(7,i))$.

	(0,7)	1	2	3	4	5	6	8	9
(0,7)	0	6	6	5	6	7	7	4	3
1		0	6	3	11	3	10	9	8
2			0	7	11	3	10	9	8
3				0	8	4	11	6	5
4					0	12	5	4	3
5						0	11	10	9
6							0	9	6
8								0	3
9									0

2) From the updated distance matrix, the smallest distance is now $\min(D_1^{(1)}) = 3$. We merge the following clusters:

   - Cluster (0,7) with point 9
   - Point 1 with point 3 (based on the tiebreaker rule, else we could also merge point 1 and point 5)
   - Point 2 with point 5
   - Point 4 or point 8 cannot be merged with point 9, as point 9 was already merged with cluster (0,7)

	(0,7,9)	(1,3)	(2,5)	4	6	8
(0,7,9)	0	8	9	6	7	4
(1,3)		0	7	11	11	9
(2,5)			0	12	11	10
4				0	5	4
6					0	9
8						0

We update the distance matrix using complete-linkage clustering:

3) The smallest distance is now $\min(D_1^{(2)}) = 4$. We merge cluster (0,7,9) with point 8 (based on the tiebreaker rule, else we could also merge point 4 and point 8)

We update the distance matrix using complete-linkage clustering:

	(0,7,8,9)	(1,3)	(2,5)	4	6
(0,7,8,9)	0	9	10	6	9
(1,3)		0	7	11	11
(2,5)			0	12	11
4				0	5
6					0

4) The smallest distance is now $\min(D_1^{(3)}) = 5$. We merge point 4 and point 6.

We update the distance matrix using complete-linkage clustering:

	(0,7,8,9)	(1,3)	(2,5)	(4,6)
(0,7,8,9)	0	9	10	9
(1,3)		0	7	11
(2,5)			0	12
(4,6)				0

5) The smallest distance is now $\min(D_1^{(4)}) = 7$. We merge cluster (1,3) with cluster (2,5).

We update the distance matrix using complete-linkage clustering:

	(0,7,8,9)	(1,2,3,5)	(4,6)
(0,7,8,9)	0	10	9
(1,2,3,5)		0	12
(4,6)			0

6) The smallest distance is now $\min(D_1^{(5)}) = 9$. We merge cluster (0,7,8,9) with cluster (4,6).

We update the distance matrix using complete-linkage clustering:

	(0,4,6,7,8,9)	(1,2,3,5)
(0,4,6,7,8,9)	0	12
(1,2,3,5)		0

7) We merge the remaining two clusters.

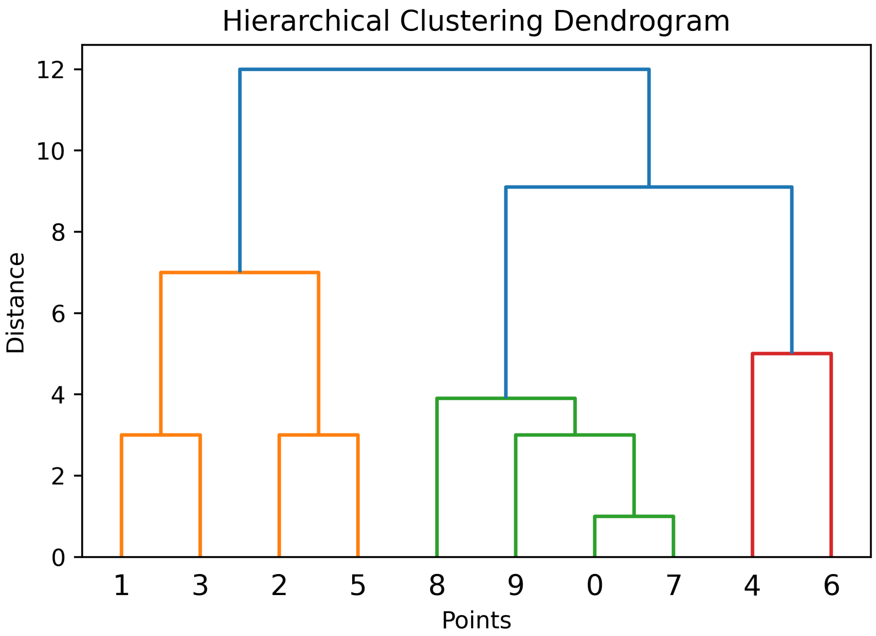


Figure 1 Dendrogram

d) The dendrogram looks as follows:

Generally, the number of clusters depends on the use case and the desired application. A good rule of thumb is to define clusters such that the distance between the clusters is large. Graphically speaking, we draw a horizontal line in the dendrogram such that its longest lines are cut (since the length of the lines represents distance between clusters). In this example, a reasonable cut would be at a distance of 8 (cut the lines marked in blue), leading to three clusters (marked orange, green, and red).

Exercise H8.2 *EM algorithm*

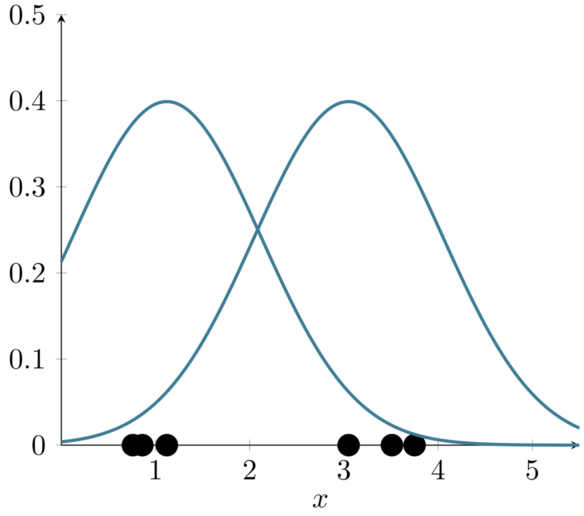
Given $k = 2$, perform EM algorithm with the following instances and initial distribution parameters:

Instance	Value
1	0.76
2	0.86
3	1.12
4	3.05
5	3.51
6	3.75

Table 2 Dataset

Parameter	Value
μ_A	1.12
σ_A	1.00
p_A	50%
μ_B	3.05
σ_B	1.00
p_B	50%

Table 3 Initial Distribution Parameters



Solution

1. Calculate cluster probabilities:

   Assuming all the data fulfill iid condition and are normally distributed, we can count $Pr[A|x]$ and $Pr[B|x]$ for each instance. For this solution, only the calculation of the first instance is shown.

$$\begin{aligned} Pr[A|x] &= \frac{Pr[x|A]Pr[A]}{Pr[x]} \\ Pr[B|x] &= \frac{Pr[x|B]Pr[B]}{Pr[x]} \end{aligned} \tag{1}$$

   We then need to calculate $Pr[x]$, $Pr[x|A]$, and $Pr[x|B]$.
   Since data is assumed to be normally distributed, we use the formula for normal distribution for $Pr[x_i|A]$ and $Pr[x_i|B]$.

$$\begin{aligned} Pr[x_1|A] &= f(x_1; \mu_A, \sigma_A) \\ &= \frac{1}{\sigma_A\sqrt{2\pi}} e^{-\frac{(x_1-\mu_A)^2}{2\sigma_A^2}} \\ &= 0.187 \end{aligned} \tag{2}$$

With the same method, we find $Pr[x_1|B] = 0.0145$. We then use it to find $Pr[x_i]$

$$\begin{aligned} Pr[x_1] &= Pr[x_1|A]Pr[A] + Pr[x_1|B]Pr[B] \\ &= 0.2015 \end{aligned} \tag{3}$$

Using the formula from above, we have $Pr[A|x] = 92.8\%$ and $Pr[B|x] = 7.2\%$. Doing it for each instance, we get the result as shown in Table 4.

Instance	$Pr[A\|x]$	$Pr[B\|x]$
1	92.81%	7.19%
2	91.41%	8.59%
3	86.56%	13.44%
4	13.44%	86.56%
5	6.01%	93.99%
6	3.87%	96.13%

Table 4 Iteration I: Cluster Probabilities

2. Update distribution parameters:

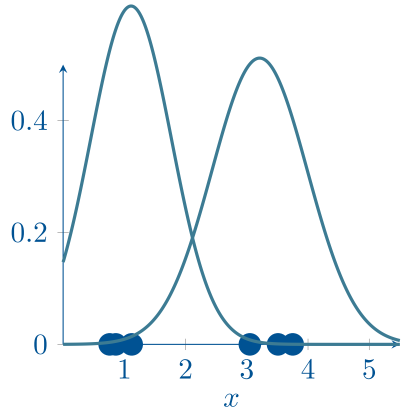
   We update the parameters using the weighted average for group A and B. The calculation shown here is for group A, but can analogously be used for group B.

$$\begin{aligned} \mu_A &= \frac{\sum_{i=1}^{6} Pr[A|x_i]x_i}{\sum_{i=1}^{6} Pr[A|x_i]} = 1.10 \\ \sigma_A^2 &= \frac{\sum_{i=1}^{6} Pr[A|x_i](x_i - \mu_A)^2}{\sum_{i=1}^{6} Pr[A|x_i]} = 0.66 \\ p_A &= \frac{\sum_{i=1}^{6} Pr[A|x_i]}{6} = 49\% \end{aligned} \tag{4}$$

   The result after the update for both groups is as shown in Table 5.

Parameter	Value
μ_A	1.10
σ_A	0.66
p_A	49%
μ_B	3.21
σ_B	0.78
p_B	51%

Table 5 Iteration I: Distribution Parameters



3. Calculate cluster probabilities: See Table 6.

4. Update distribution parameters: See Table 7.

5. Calculate cluster probabilities

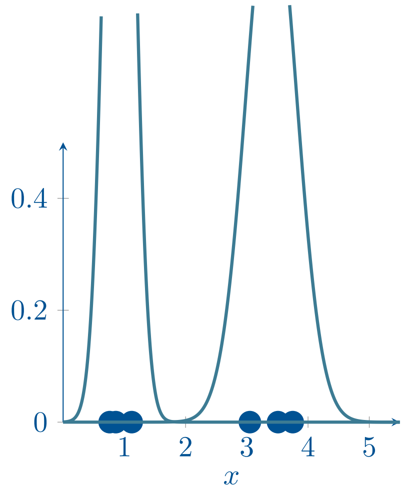
   No change in cluster assignment ⇒ termination

Instance	$Pr[A\|x]$	$Pr[B\|x]$
1	99.25%	0.75%
2	99.97%	1.03%
3	97.55%	2.45%
4	1.49%	98.51%
5	0.16%	99.84%
6	0.05%	99.95%

Table 6 Iteration II: Cluster Probabilities

Parameter	Value
μ_A	0.92
σ_A	0.22
p_A	49%
μ_B	3.40
σ_B	0.41
p_B	51%

Table 7 Iteration II: Distribution Parameters



Exercise H8.3 *Ensemble methods*

a) Name benefits that an ensemble model (ideally) has in comparison to a single model.

b) In terms of the training process, what is a major difference between bagging and boosting?

Solution

a) Ensemble models tend to be more stable than single models. As the final prediction is the summary of a lot of different "expert opinions", a small change in the input data does not necessarily change the final prediction. Moreover, the combination of models might reduce the predictor’s variance. Collectively, this can lead to better prediction performances.

b) The bagging method draws samples and then trains several models at the same time, and can thus be easily parallelized. In contrast, in boosting the kth model depends on the prediction of the k-1st, the k-1st model depends on the prediction of the k-2nd model and so forth (the weights of the data change with each prediction model). In addition, boosting tends to overfit more easily than bagging.