

Introduction to Hierarchical Clustering

Lieven Clement

Contents

1	Introduction	1
1.1	Objective	1
1.2	Example 1	1
1.3	Example 2.	2
2	Hierarchical Cluster Analysis: Agnes	2
2.1	General Algorithm	2
2.2	Intercluster Dissimilarities	2
2.3	Cluster Tree	4
3	Toy example	4
3.1	Single linkage	5
3.2	Complete linkage	6
3.3	Average linkage	7

1 Introduction

1.1 Objective

Objective: grouping of observations into **clusters**, so that

- similar observations appear in the same cluster
- dissimilar observations appear in distinct clusters

→ need for a measure for **similarity** and **dissimilarity**?

1.2 Example 1

Single cell transcriptomics: $n \times p$ Matrix for which - every column contains the expression levels of one of p genes for n cells - every row contains the expression levels of p genes for one cell ($\{sample\}$)

- *Research question: look for groups of cells that have similar gene expression patterns*

- Or, look for groups of genes that have similar expression levels across the different cells. This can help us in understanding the regulation and functionality of the genes.

→ both **observations** (rows) and **variables** (columns) can be clustered

1.3 Example 2.

Abundance studies: the abundances of n plant species are counted on p plots (habitats)

- look for groups that contain species that live in the same habitats, or, look for groups of habitats that have similar species communities

→ both **observations** (rows) and **variables** (columns) can be clustered

2 Hierarchical Cluster Analysis: Agnes

2.1 General Algorithm

- In step 0 each observations is considered as a cluster (i.e. n clusters).
- Every next step consists of:
 1. merge the two clusters with the smallest intercluster dissimilarity
 2. recalculate the intercluster dissimilarities

In step 0 the intercluster dissimilarity coincides with the dissimilarity between the corresponding observations
 | ↔ intercluster dissimilarity?

2.2 Intercluster Dissimilarities

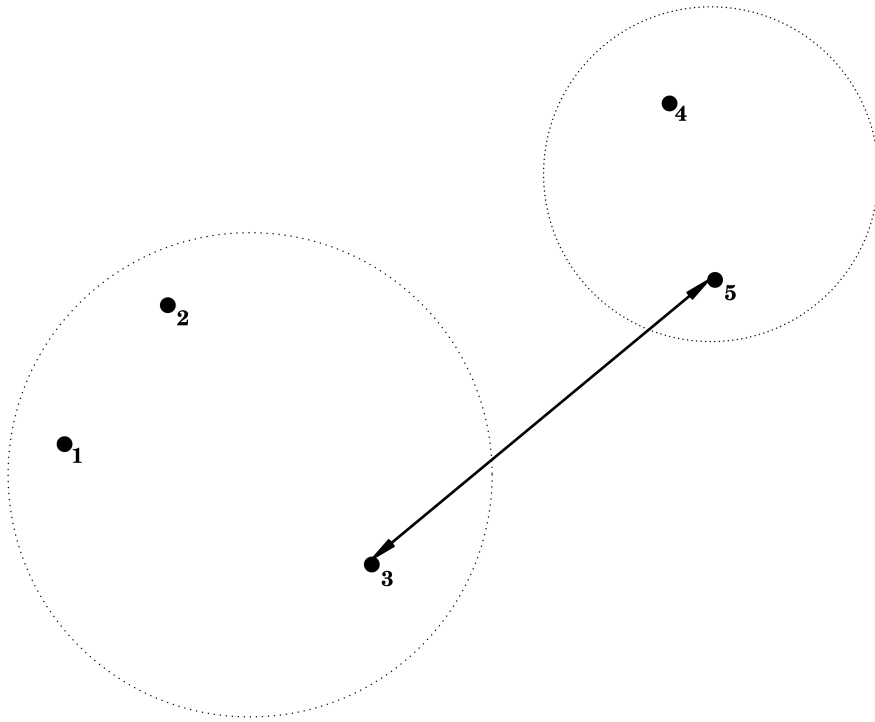
- Represent clusters (e.g. C_1 and C_2) as sets of points \mathbf{x}_i which belong to that cluster
- $d(C_1, C_2)$: intercluster dissimilarity between

We consider three intercluster dissimilarities.

2.2.1 Single Linkage = Nearest Neighbour

$$d(C_1, C_2) = \min_{\mathbf{x}_1 \in C_1; \mathbf{x}_2 \in C_2} d(\mathbf{x}_1, \mathbf{x}_2),$$

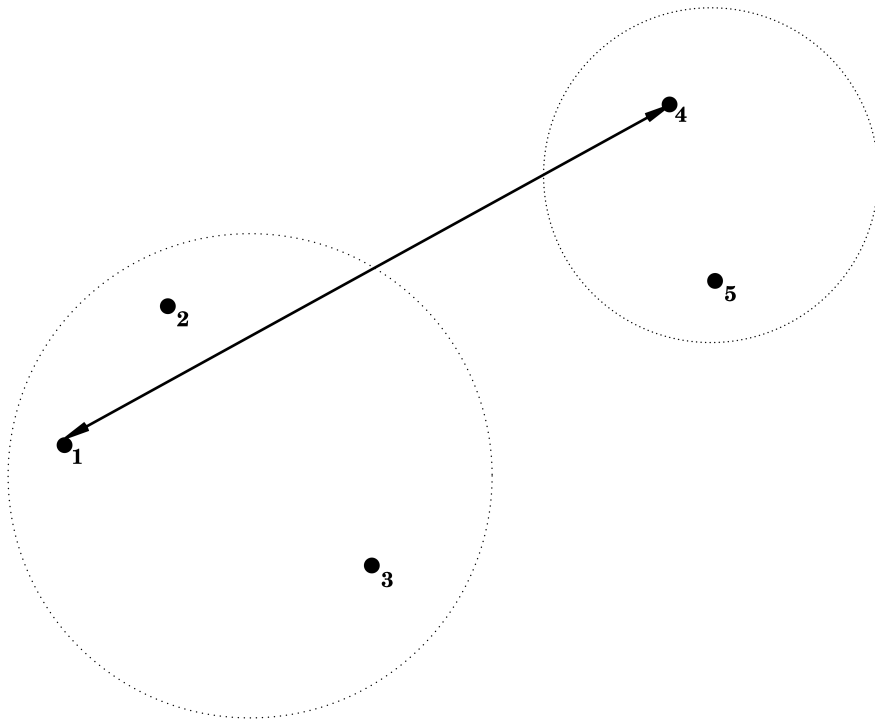
i.e. the dissimilarity between C_1 and C_2 is determined by the smallest dissimilarity between a point of C_1 and a point of C_2 .



2.2.2 Complete Linkage = Furthest Neighbour

$$d(C_1, C_2) = \max_{\mathbf{x}_1 \in C_1; \mathbf{x}_2 \in C_2} d(\mathbf{x}_1, \mathbf{x}_2),$$

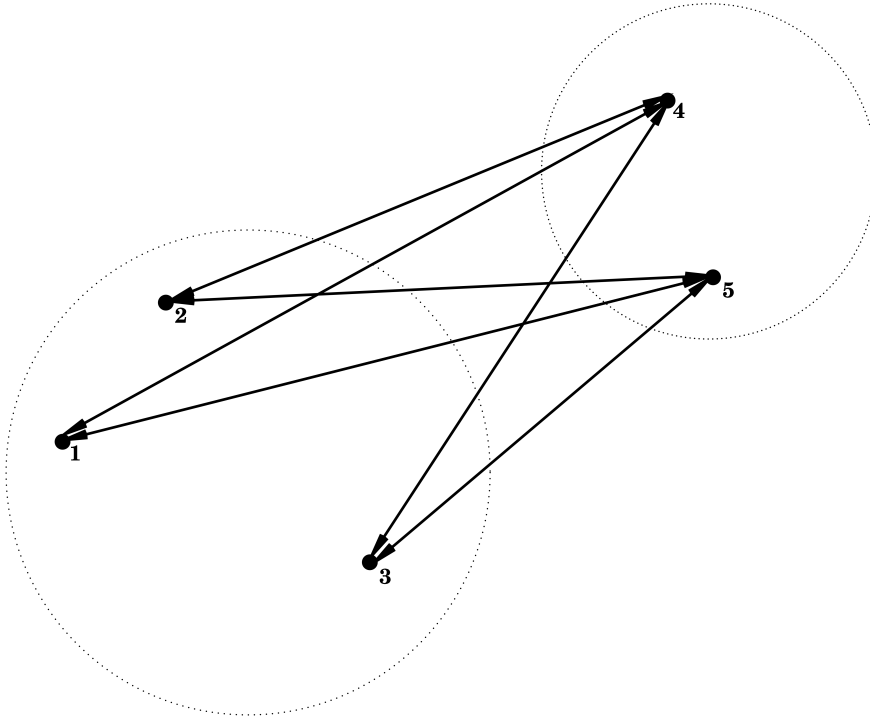
i.e. the dissimilarity between C_1 and C_2 is determined by the largest dissimilarity between a point of C_1 and a point of C_2 .



2.2.3 Average Linkage = Group Average

$$d(C_1, C_2) = \frac{1}{|C_1||C_2|} \sum_{\mathbf{x}_1 \in C_1; \mathbf{x}_2 \in C_2} d(\mathbf{x}_1, \mathbf{x}_2),$$

i.e. the dissimilarity between C_1 and C_2 is determined by the average dissimilarity between all points of C_1 and all points of C_2 .



2.3 Cluster Tree

Hierarchical nature of the algorithm:

- *Nested sequence of clusters \rightarrow visualisation via a tree*
- *height of branches indicate the intercluster dissimilarity at which clusters are merged.*
- *Can used as instrument for deciding the number of clusters in the data*

3 Toy example

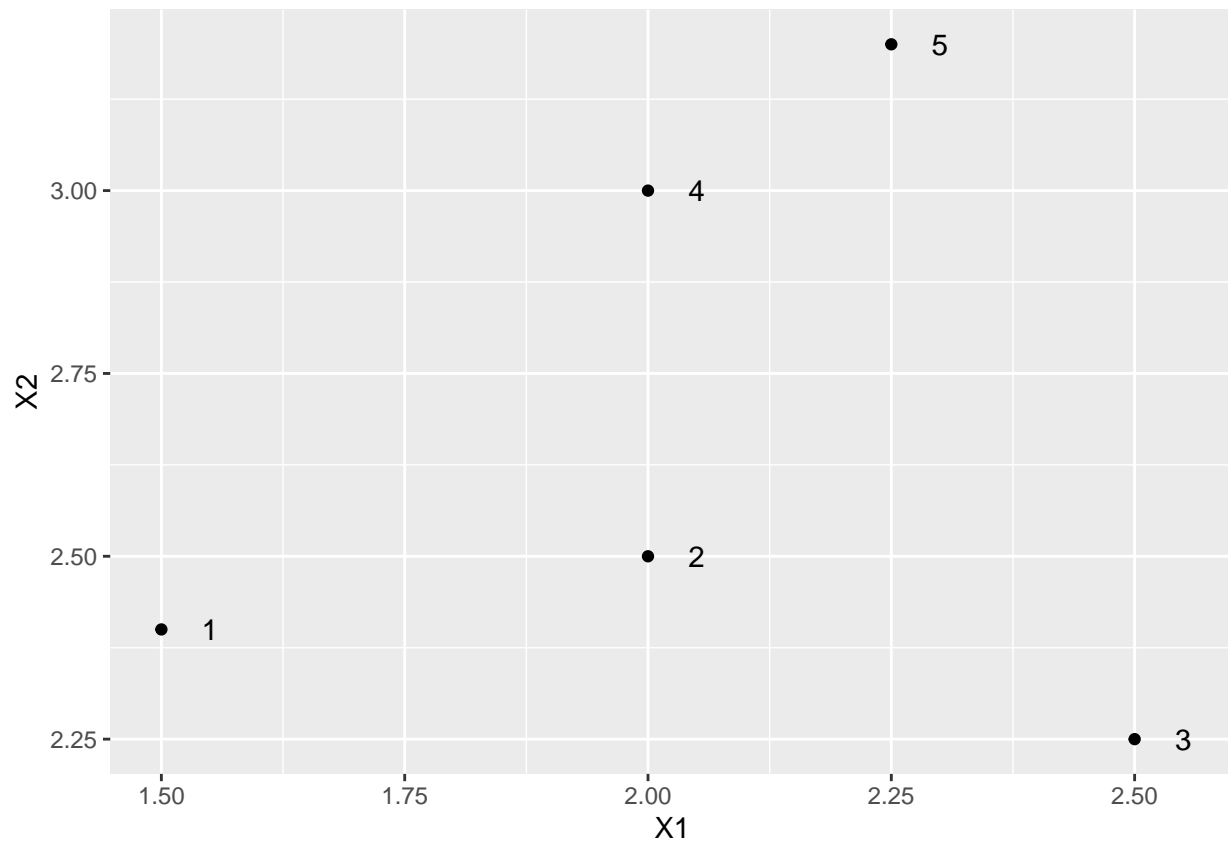
$X1$	$X2$	label
1.50	2.40	1
2.00	2.50	2
2.50	2.25	3
2.00	3.00	4
2.25	3.20	5

```

library(cluster)
library(tidyverse)

toy %>%
  ggplot(aes(X1, X2, label = label)) +
  geom_point() +
  geom_text(nudge_x = .05)

```



```
toy[,1:2] %>% dist
```

```

##          1          2          3          4
## 2 0.5099020
## 3 1.0111874 0.5590170
## 4 0.7810250 0.5000000 0.9013878
## 5 1.0965856 0.7433034 0.9823441 0.3201562

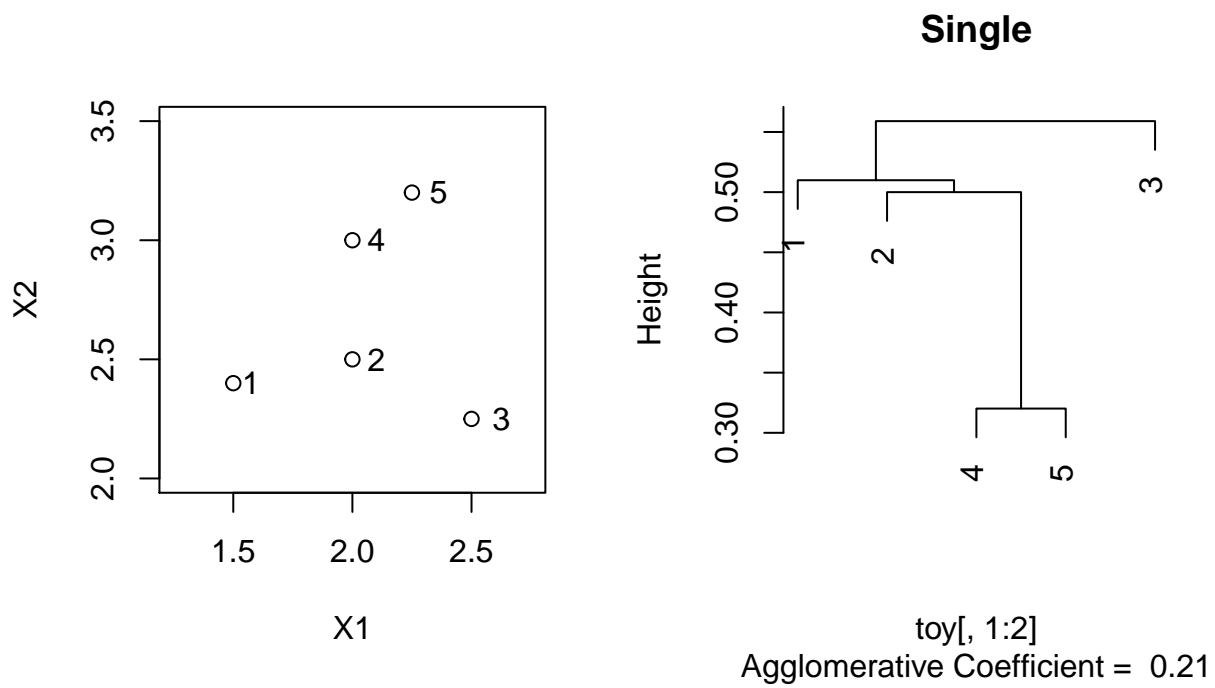
```

3.1 Single linkage

```

toySingle <- agnes(toy[,1:2], method = "single")
par(mfrow=c(1,2),pty="s")
plot(X2 ~ X1, toy, xlim = c(1.25,2.75),ylim = c(2,3.5))
text(toy$X1*1.05,toy$X2,label=toy$label)
plot(toySingle, which.plot = 2, main = "Single")

```

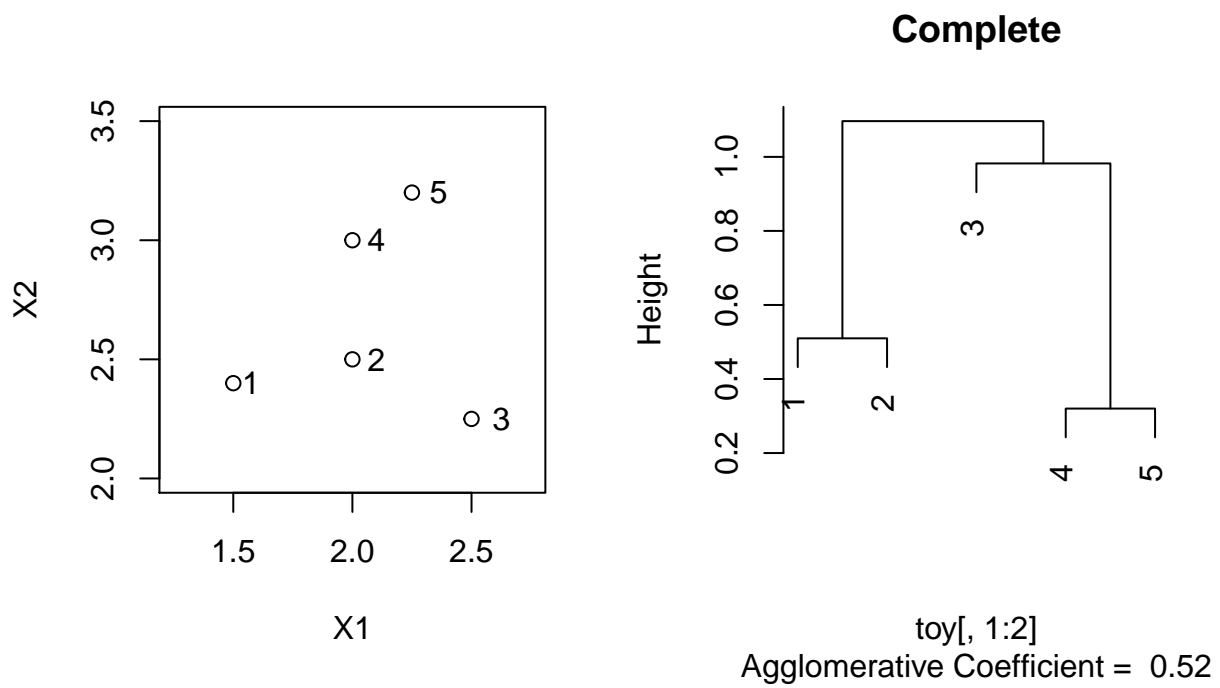


```
toy[,1:2] %>% dist
```

```
##           1           2           3           4
## 2 0.5099020
## 3 1.0111874 0.5590170
## 4 0.7810250 0.5000000 0.9013878
## 5 1.0965856 0.7433034 0.9823441 0.3201562
```

3.2 Complete linkage

```
toyComplete <- agnes(toy[,1:2], method = "complete")
par(mfrow=c(1,2),pty="s")
plot(X2 ~ X1, toy, xlim = c(1.25,2.75),ylim = c(2,3.5))
text(toy$X1*1.05,toy$X2,label=toy$label)
plot(toyComplete, which.plot = 2, main = "Complete")
```

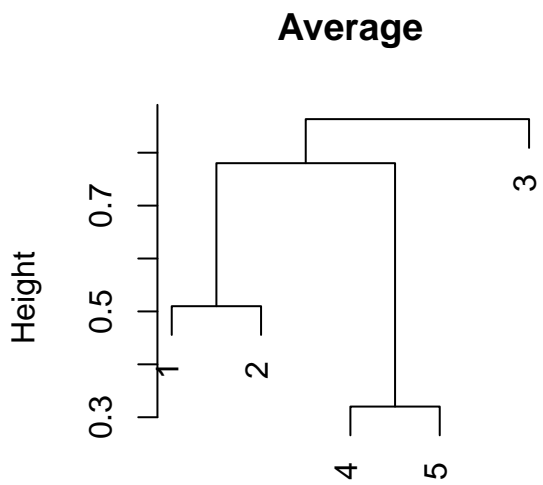
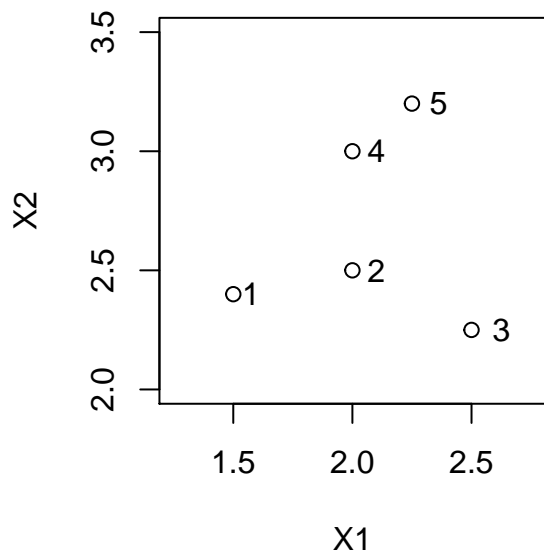


```
toy[,1:2] %>% dist
```

```
##           1           2           3           4
## 2 0.5099020
## 3 1.0111874 0.5590170
## 4 0.7810250 0.5000000 0.9013878
## 5 1.0965856 0.7433034 0.9823441 0.3201562
```

3.3 Average linkage

```
toyAvg <- agnes(toy[,1:2], method = "average")
par(mfrow=c(1,2),pty="s")
plot(X2 ~ X1, toy, xlim = c(1.25,2.75),ylim = c(2,3.5))
text(toy$X1*1.05,toy$X2,label=toy$label)
plot(toyAvg, which.plot = 2, main = "Average")
```



toy[, 1:2]
 Agglomerative Coefficient = 0.42

```
toy[,1:2] %>% dist
```

```
##           1           2           3           4
## 2 0.5099020
## 3 1.0111874 0.5590170
## 4 0.7810250 0.5000000 0.9013878
## 5 1.0965856 0.7433034 0.9823441 0.3201562
```