Unsupervised learning: Clustering

Contents

Introduction

In this practical, we will learn how to perform clustering.

We will use the packages `igraph`, `ggdendro`, and `dendextend`. For this, you will probably need to `install.packages()` before running the `library()` functions.

```r
library(igraph)
library(ggdendro)
library(dendextend)
library(ISLR)
library(tidyverse)
```

---

1. **Load the dataset `data/clusterdata.csv`. Create a scatter plot of this dataset, mapping `x1` to the x position and `x2` to the y position. Use `coord_fixed()` to ensure that the x and y axes are the same size w.r.t. their values.**

K-means clustering

The `kmeans()` function implements k-means clustering.

---

2. **Create two cluster objects with the `kmeans()` function using the same data, one with 3 clusters and one with 5 clusters.**

3. **Create two scatterplots where you map the cluster assignment of the cluster objects to the colour of the points.**

Hierarchical clustering

The `hclust()` function implements hierarchical clustering.

4. **Compute hierarchical cluster objects with the `hclust()` function using the same data, one with complete-linkage and one with average-linkage. (Hint: use `dist()` function to produce dissimilarity structure)**

5. **Use the `ggdendrogram()` function from `library(ggdendro)` to plot two dendrograms for the clustering objects.**

6. **Now we want to compare dendrograms. First start by transforming the results as dendrograms and create a list to hold the two dendrograms using `dendlist()` function. And then visualise the comparison of two dendrograms with `tanglegram()` function.**

7. **Does complete-linkage hierarchical clustering with a cutoff at 3 clusters yield the same result as 3-means clustering? Hint: use the `cutree()` function to cut off the hierarchical clustering object at 3 clusters.**

Programming assignment: manual K-means clustering

The euclidian distance between two vectors $\mathbf{x}$ and $\mathbf{y}$ of length n is $D = ||\mathbf{x} - \mathbf{y}||_2 = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$. These two vectors represent points in n-dimensional space and the euclidian distance is the straight-line distance between these points.

8. **Write a function `l2_dist(x, y)` that takes in two vectors and outputs the euclidian distance between the two vectors.**

9. **Program a k-means clustering algorithm and apply it to this data. Use Algorithm 10.1 from the ISLR book. Visualise your result.**