Cluster analysis of weather in Amsterdam, 1891-1939

Introduction

The aim of this document is to explore how similar months between 1891 and 1939 were in terms of weather in Amsterdam, the Netherlands. The main method is cluster analysis.

Preparation

```
library(readxl)
library(ggplot2)
library(factoextra)

## Welcome! Want to learn more? See two factoextra-related books at
https://goo.gl/ve3WBa

library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse
1.3.1 --

## v tibble  3.1.6     v purrr   0.3.4
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------
tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(cluster)

## Warning: package 'cluster' was built under R version 4.1.3

setwd("~/LUMC postdoc/DS portfolio/weather")
```

STEP 1: preparing the data

For cluster analysis, you need to make sure that the database contains only the variables that you want to cluster your observations on. This means that you need to set the index (year/month) and leave only the weather variables as columns in your dataset.

```
## Warning: Setting row names on a tibble is deprecated.
```

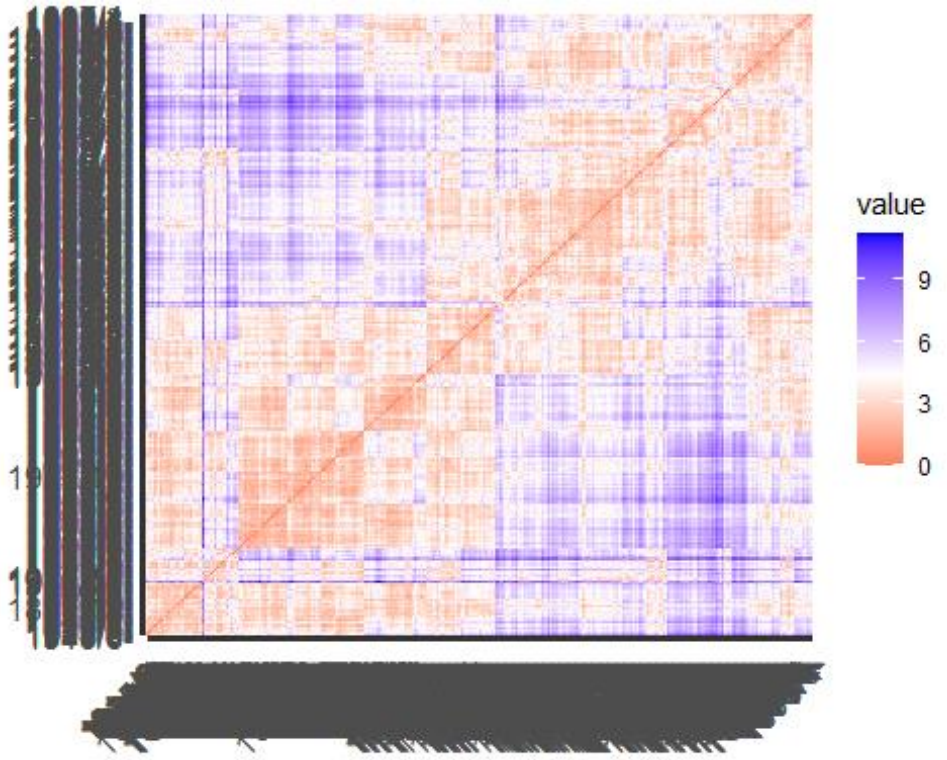
STEP 2: calculating and visualizing the distance matrix

This step calculates the basis for the cluster analysis: the multivariate distance between months. You can select between multiple distance measures, the default is Euclidean which we also use here. Since the resulting matrix is organized based on similarity, you can already have an idea if certain months are more similar to each other, for instance winter months can be very similar to March.

```
# calculate distance matrix

distance <- get_dist(data, method = "euclidean")
```

Since the distance matrix is large, we might want to visualize it.

```
distance_plot <- fviz_dist(distance, lab_size = 12, show_labels = TRUE,
gradient = list(low = "red", mid = "white", high = "blue"))
distance_plot
```



STEP 3: executing cluster analysis and visualizing the results

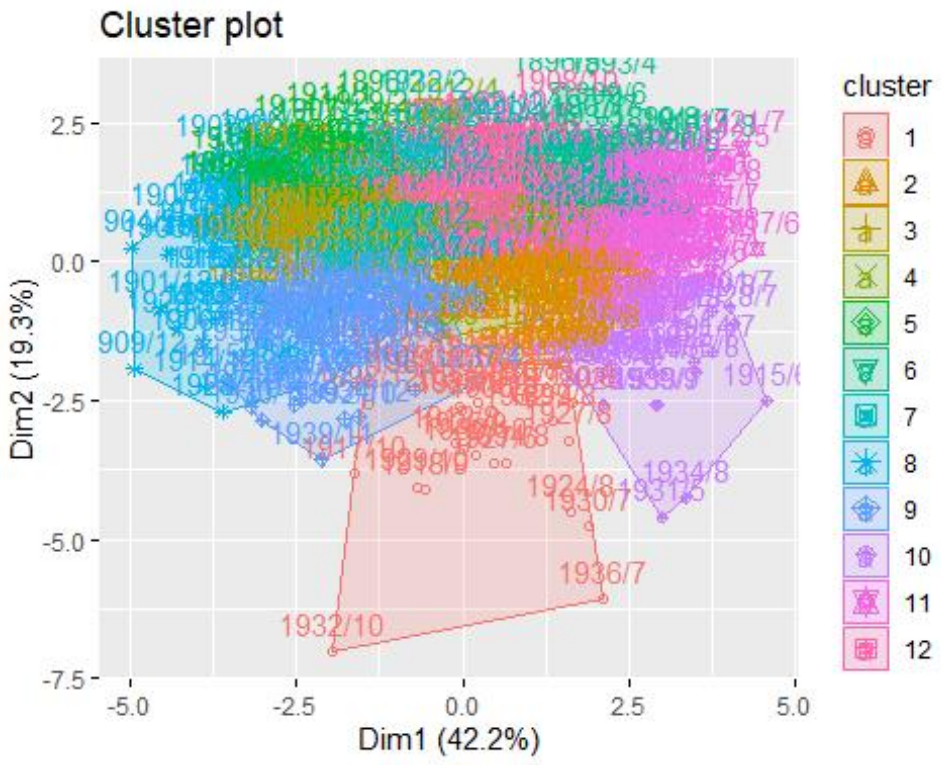
We perform k-means cluster analysis where we set the number of resulting clusters 12 (k=12). The reason is that we want to see how the actual 12 months of the year can be classified into 12 groups which contains months which are the most similar to each other. Then, we can see if, for instance, all Januaries are assigned to the same cluster or not and if not, which months are in the same cluster as January?

Another approach is to set the number of k to 4, because we are supposed to find four seasons. Alternatively, you might want to find the optimal number of clusters (more on this issue: https://www.r-bloggers.com/2017/02/finding-optimal-number-of-clusters/).

```
cluster_12 <- kmeans(data, centers = 12, nstart = 25, iter.max = 40,
algorithm = "Forgy")
```

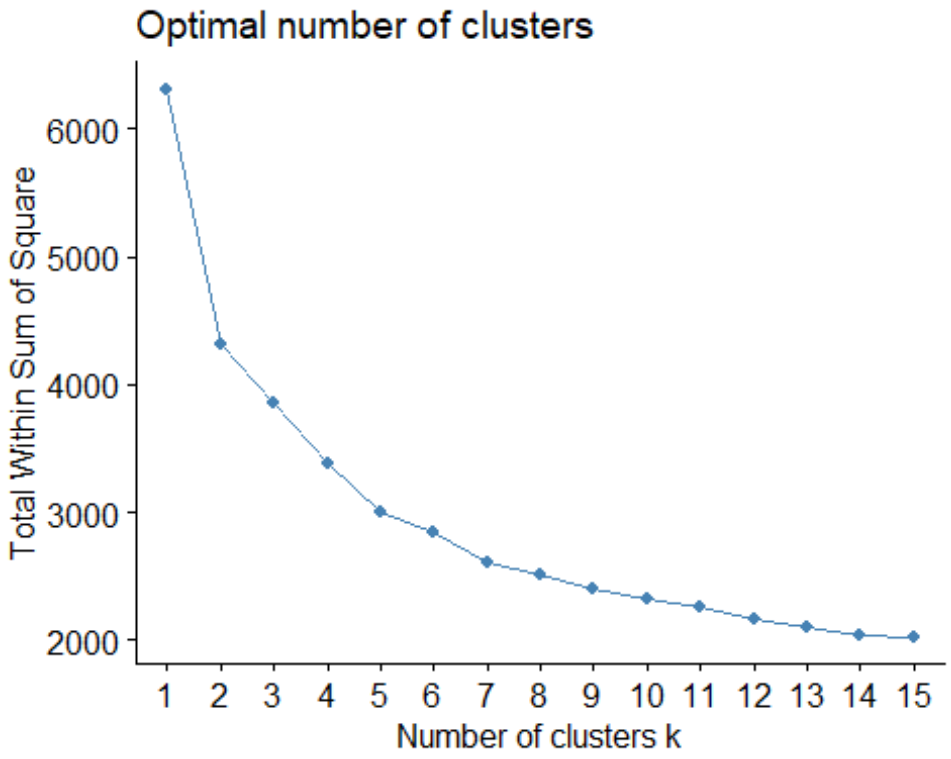
Visualizing clusters along the first two principal components. We can see that the clusters are overlapping, which means that they are not so distinct.

```
fviz_cluster(cluster_12, data = data, axes = c(1, 2))
```



One option to try to find the optimal number of clusters is with the fviz_nbclust(). The figure below cannot be interpreted in a straightforward manner: it seems that there is no clear cut. Maybe the optimal number of clusters is 9?

```
fviz_nbclust(data, kmeans, k.max=15, method = "wss")
```



STEP 4: Inspecting the results in depth

First, we explore how many observations (N=575) we have per cluster.

```
cluster_12$size
```

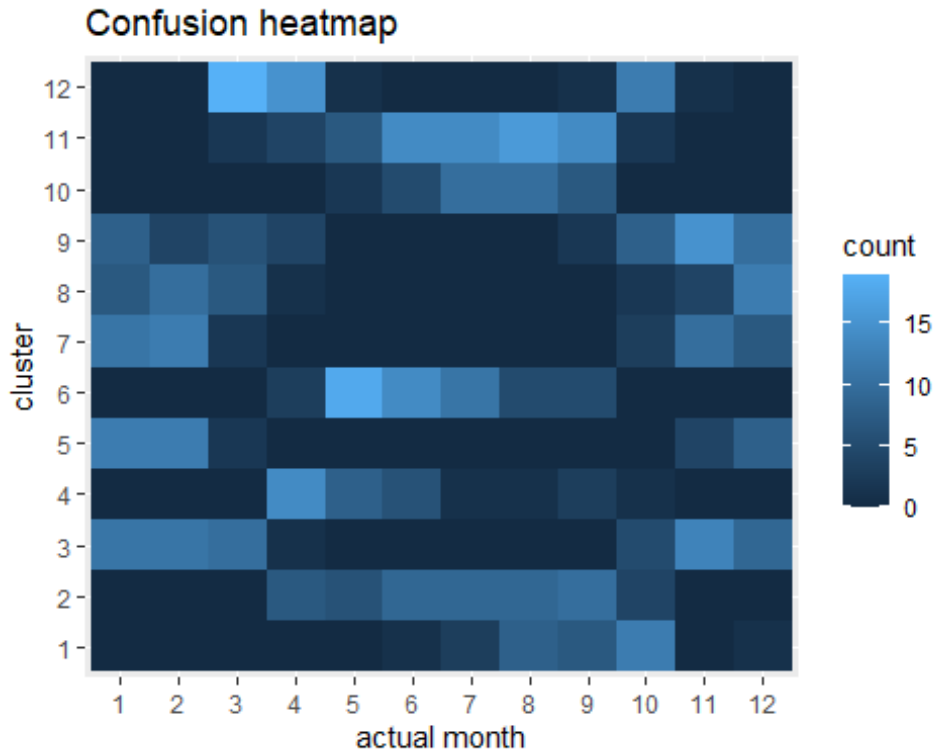
```
##  [1] 32 54 60 34 38 56 45 43 57 34 73 49
```

Then, we create a "confusion plot" which shows the true month on the vertical axis and the assigned cluster on the horizontal axis. This will basically give us a heatmap.

```
# we need to add some extra columns to our data

# add assigned cluster
data$cluster <- cluster_12$cluster

# add actual month
data$month <- month

# confusion matrix

confusion <- as.data.frame(table(data$month, data$cluster))
colnames(confusion) <- c("month", "cluster", "count")
confusion$cluster <- as.numeric(as.character(confusion$cluster))
confusion$month <- as.numeric(as.character(confusion$month))
```

Visualizing how many times each month was assigned to the 12 clusters.

```
heatmap <- ggplot(confusion, aes(as.factor(month), as.factor(cluster), fill=
count)) +
  geom_tile() +
  labs(x = "actual month", y = "cluster", title = "Confusion heatmap")
heatmap
```



The heatmap can be interpreted from the point of view of the x axis or the y axis. The following observations are striking. Cluster 7 must be the “winter cluster” because the majority consists of November, December, January and February. Cluster 3 is similar to Cluster 7. Both Cluster 1 and 11 contain mostly summer months. Cluster 12 is dominated by October which suggests that October must have special features.

Finally, we visualize the mean of each variable by cluster to see how different the created clusters really are.

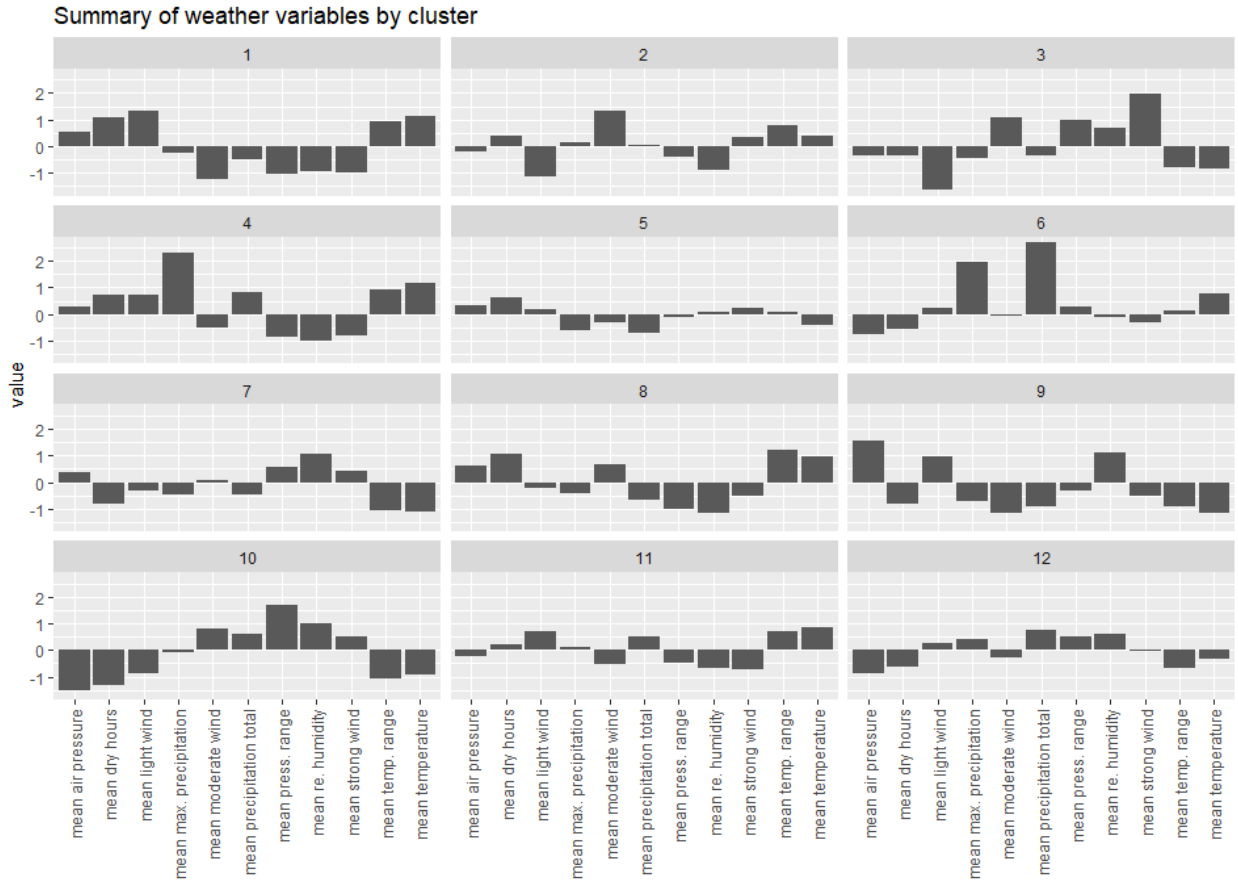
```
cluster_means <- data %>%
  group_by(cluster) %>%
  dplyr::summarise(`mean temperature` = mean(z_mean_temp),
                   `mean temp. range` = mean(z_temp_range),
                   `mean air pressure` = mean(z_mean_press),
                   `mean press. range` = mean(z_press_range),
                   `mean dry hours` =  mean(z_dry),
                   `mean light wind` =  mean(z_lightwind),
                   `mean moderate wind` =  mean(z_modwind),
                   `mean strong wind` =  mean(z_strongwind),
                   `mean precipitation total` =  mean(z_prec_total),
                   `mean max. precipitation` =  mean(z_prec_max),
                   `mean re. humidity` =  mean(z_rel_hum)
                   ) %>%
  ungroup() %>%
  pivot_longer(cols = c("mean temperature","mean temp. range",
                        "mean air pressure", "mean press. range", "mean dry
hours",
                        "mean light wind", "mean moderate wind", "mean strong
wind",
                        "mean precipitation total", "mean max.
precipitation",  "mean re. humidity"),
               names_to = "variable",
               values_to = "value")
```

Now, visualizing the mean of each weather variable by cluster using bar charts.

This plot shows how the clusters differ by variable.

This plot also help us interpret the heatmap. For instance, we have seen that cluster 1 and 8 are both contain summer month. However, the bar chart sheds light on the main difference between the two clusters: the two clusters differ in terms of precipitation and windstrength.

```
barchart <- ggplot(data=cluster_means, aes(x=variable, y=value)) +
  geom_bar(stat="identity")+
  labs(title = "Summary of weather variables by cluster", x="")+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))+
  facet_wrap(~cluster, ncol = 3)
barchart
```


Summary of weather variables by cluster
1
2
3
4
5
6
7
8
9
10
11
12
value
2
1
0
-1
mean air pressure
mean dry hours
mean light wind
mean max. precipitation
mean moderate wind
mean precipitation total
mean press. range
mean re. humidity
mean strong wind
mean temp. range
mean temperature