"Absenteeism at work - Exploratory Analysis"

Tools for Decision Making- Project 01

Babak Barghi, Cyra Stamm, Daniel Zöttl

December 04, 2020

Contents

1 Introduction

Coping with the high rate of business environment or gaining competitive advantages and customer satisfaction are mainly based on organizational resources, especially an employee. Employees with low performance cause a vital lose for organizations and the absenteeism consider to be one of the factors that affect performance. Absenteeism is defined as absence to work as expected, represents for the company the loss of productivity and quality of work. In this context, the prediction of absenteeism becomes important in decision making avoiding loss of efficiency and excellence at work. The aim of this project is to discover the factors and causes of employees absence using exploratory data analysis. In this methodology, the work starts with data cleaning and preprocessing the absenteeism database by grouping some type of attributes and studying the correlation between the attributes and the absenteeism to find the significant causes of absenteeism.

2 Software Settings

The analysis is carried out in *R 4.0.2*[1] and the *tidyverse*[2], *kableExtra* [3], *RColorBrewer* [4], *corrplot* [5], *cluster* [6], *factoextra* [7], *dendextend* [8], *gridExtra*[9] and *plyr* [10] packages are used.

3 Data Description

The variables are:

1. Individual identification (ID)
2. Reason for absence (ICD).

Absences attested by the International Code of Diseases (ICD) stratified into 21 categories (I to XXI) as follows:

- 1 Certain infectious and parasitic diseases
- 2 Neoplasms
- 3 Diseases of the blood and blood-forming organs and certain disorders involving the immune mechanism
- 4 Endocrine, nutritional and metabolic diseases
- 5 Mental and behavioural disorders
- 6 Diseases of the nervous system
- 7 Diseases of the eye and adnexa
- 8 Diseases of the ear and mastoid process
- 9 Diseases of the circulatory system

- 10 Diseases of the respiratory system
- 11 Diseases of the digestive system
- 12 Diseases of the skin and subcutaneous tissue
- 13 Diseases of the musculoskeletal system and connective tissue
- 14 Diseases of the genitourinary system
- 15 Pregnancy, childbirth and the puerperium
- 16 Certain conditions originating in the perinatal period
- 17 Congenital malformations, deformations and chromosomal abnormalities
- 18 Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified
- 19 Injury, poisoning and certain other consequences of external causes
- 20 External causes of morbidity and mortality
- 21 Factors influencing health status and contact with health services.

And 7 categories without (CID) patient follow-up (22), medical consultation (23), blood donation (24), laboratory examination (25), unjustified absence (26), physiotherapy (27), dental consultation (28).

3. Month of absence
4. Day of the week (Monday (2), Tuesday (3), Wednesday (4), Thursday (5), Friday (6))
5. Seasons
6. Transportation expense
7. Distance from Residence to Work (kilometres)
8. Service time
9. Age
10. Work load Average/day
11. Hit target
12. Disciplinary failure (yes=1; no=0)
13. Education (high school (1), graduate (2), postgraduate (3), master and doctor (4))
14. Son (number of children)
15. Social drinker (yes=1; no=0)
16. Social smoker (yes=1; no=0)
17. Pet (number of pet)
18. Weight
19. Height
20. Body mass index
21. Absenteeism time in hours (target)

4 Data Cleaning

Data cleaning is crucial as an analysis can only be successful if the data has been imported correctly and there are no errors in the data. In any study, the quality of the data has a direct effect on the results. In order to build a suitable prediction model, the database should be understood and studied carefully to get more

insight about it and perform meaningful modifications. To import the data, first the previously mentioned packages are loaded by using the *library* function then several steps are carried out to check whether the data is correct.

```
library("tidyverse")
library("kableExtra")
library("RColorBrewer")
library("corrplot")
library("cluster")
library("factoextra")
library("dendextend")
library("gridExtra")
library("plyr")
theme_set(theme_minimal(base_size = 12))
```

In the next step the data sets can be imported to *R* using the *read.csv* function. When using the function it has to be considered if the data sets have a header, which symbol is used to separate the data and which decimal operator is used.

```
#importing the data set "Absenteeism_at_work.csv"
myData <- read.csv("Absenteeism_at_work.csv", header=TRUE, sep=";", dec=".")
```

The function *names* can be used to identify if there are any spaces between the variables because if so errors will occur during the analysis. To circumvent these errors the function *make.names* is used. This function removes the spaces and replaces them with points.

```
#data set myData
names(myData)
```

```
##  [1] "ID"                                "Reason.for.absence"
##  [3] "Month.of.absence"                  "Day.of.the.week"
##  [5] "Seasons"                           "Transportation.expense"
##  [7] "Distance.from.Residence.to.Work"   "Service.time"
##  [9] "Age"                               "Work.load.Average.day"
## [11] "Hit.target"                        "Disciplinary.failure"
## [13] "Education"                         "Son"
## [15] "Social.drinker"                    "Social.smoker"
## [17] "Pet"                               "Weight"
## [19] "Height"                            "Body.mass.index"
## [21] "Absenteeism.time.in.hours"
```

```
names(myData) <- make.names(names(myData))
```

In this case there are no errors but when handling a large amount of data it is better to use the *make.names* function to prevent errors. In the next step the type of variables has to be checked. For that the *str* function is used.

```
str(myData,width=80,strict.width="cut")
```

```
## 'data.frame':    740 obs. of  21 variables:
##  $ ID                               : int  11 36 3 7 11 3 10 20 14 1 ...
```

```
##  $ Reason.for.absence             : int  26 0 23 7 23 23 22 23 19 22 ...
##  $ Month.of.absence               : int  7 7 7 7 7 7 7 7 7 7 ...
##  $ Day.of.the.week                : int  3 3 4 5 5 6 6 6 2 2 ...
##  $ Seasons                        : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Transportation.expense         : int  289 118 179 279 289 179 361 260 155 2..
##  $ Distance.from.Residence.to.Work: int  36 13 51 5 36 51 52 50 12 11 ...
##  $ Service.time                   : int  13 18 18 14 13 18 3 11 14 14 ...
##  $ Age                            : int  33 50 38 39 33 38 28 36 34 37 ...
##  $ Work.load.Average.day          : num  240 240 240 240 240 ...
##  $ Hit.target                     : int  97 97 97 97 97 97 97 97 97 97 ...
##  $ Disciplinary.failure           : int  0 1 0 0 0 0 0 0 0 0 ...
##  $ Education                      : int  1 1 1 1 1 1 1 1 1 3 ...
##  $ Son                            : int  2 1 0 2 2 0 1 4 2 1 ...
##  $ Social.drinker                 : int  1 1 1 1 1 1 1 1 1 0 ...
##  $ Social.smoker                  : int  0 0 0 1 0 0 0 0 0 0 ...
##  $ Pet                            : int  1 0 0 0 1 0 4 0 0 1 ...
##  $ Weight                         : int  90 98 89 68 90 89 80 65 95 88 ...
##  $ Height                         : int  172 178 170 168 172 170 172 168 196 1..
##  $ Body.mass.index                : int  30 31 31 24 30 31 27 23 25 29 ...
##  $ Absenteeism.time.in.hours      : int  4 0 2 4 2 2 8 4 40 8 ...
```

We can see that the type of the variables fits to the expected one.

In the next step of the data cleaning the beginning and end of the data set are viewed because there might be blank spaces in the beginning or end which could cause errors when analyzing. To view the beginning and end the *head* and *tail* functions are used.

```
#data set myData
kable(head(myData[,1:6]), align= c('c'),
      caption='Head of Data: first 6 rows', booktabs=TRUE, linesep="") %>%
  kable_styling(latex_options =c('striped', 'scale_down', 'hold_position'))
```

Table 1: Head of Data: first 6 rows

ID	Reason.for.absence	Month.of.absence	Day.of.the.week	Seasons	Transportation.expense
11	26	7	3	1	289
36	0	7	3	1	118
3	23	7	4	1	179
7	7	7	5	1	279
11	23	7	5	1	289
3	23	7	6	1	179

```
  kable(head(myData[,7:13]), align= c('c'),
        caption='Head of Data: rows 7-13', booktabs=TRUE, linesep="") %>%
  kable_styling(latex_options =c('striped', 'scale_down', 'hold_position'))
```

```
  kable(head(myData[,14:21]), align= c('c'),
        caption='Head of Data: rows 14-21', booktabs=TRUE, linesep="") %>%
  kable_styling(latex_options =c('striped', 'scale_down', 'hold_position'))
```

Table 2: Head of Data: rows 7-13

Distance.from.Residence.to.Work	Service.time	Age	Work.load.Average.day	Hit.target	Disciplinary.failure	Education
36	13	33	239.554	97	0	1
13	18	50	239.554	97	1	1
51	18	38	239.554	97	0	1
5	14	39	239.554	97	0	1
36	13	33	239.554	97	0	1
51	18	38	239.554	97	0	1

Table 3: Head of Data: rows 14-21

Son	Social.drinker	Social.smoker	Pet	Weight	Height	Body.mass.index	Absenteeism.time.in.hours
2	1	0	1	90	172	30	4
1	1	0	0	98	178	31	0
0	1	0	0	89	170	31	2
2	1	1	0	68	168	24	4
2	1	0	1	90	172	30	2
0	1	0	0	89	170	31	2

```
kable(tail(myData[,1:6]), align= c('c'),
      caption='Tail of Data: first 6 rows', booktabs=TRUE, linesep="") %>%
  kable_styling(latex_options =c('striped', 'scale_down', 'hold_position'))
```

Table 4: Tail of Data: first 6 rows

	ID	Reason.for.absence	Month.of.absence	Day.of.the.week	Seasons	Transportation.expense
735	13	13	7	2	1	369
736	11	14	7	3	1	289
737	1	11	7	3	1	235
738	4	0	0	3	1	118
739	8	0	0	4	2	231
740	35	0	0	6	3	179

```
kable(tail(myData[,7:13]), align= c('c'),
      caption='Tail of Data: rows 7-13', booktabs=TRUE, linesep="") %>%
  kable_styling(latex_options =c('striped', 'scale_down', 'hold_position'))
```

Table 5: Tail of Data: rows 7-13

	Distance.from.Residence.to.Work	Service.time	Age	Work.load.Average.day	Hit.target	Disciplinary.failure	Education
735	17	12	31	264.604	93	0	1
736	36	13	33	264.604	93	0	1
737	11	14	37	264.604	93	0	3
738	14	13	40	271.219	95	0	1
739	35	14	39	271.219	95	0	1
740	45	14	53	271.219	95	0	1

```
kable(tail(myData[,14:21]), align= c('c'),
      caption='Tail of Data: rows 14-21', booktabs=TRUE, linesep="") %>%
kable_styling(latex_options =c('striped', 'scale_down', 'hold_position'))
```

Table 6: Tail of Data: rows 14-21

	Son	Social.drinker	Social.smoker	Pet	Weight	Height	Body.mass.index	Absenteeism.time.in.hours
735	3	1	0	0	70	169	25	80
736	2	1	0	1	90	172	30	8
737	1	0	0	1	88	172	29	4
738	1	1	0	8	98	170	34	0
739	2	1	0	2	100	170	35	0
740	1	0	0	1	77	175	25	0

In the next step the range of values for the numerical variables is checked with the *summary* function in order to find any outlier in the values, exclude errors such as typing errors or misunderstandings e.g. units. This is carried out for data set.

```
summary(myData)
```

```
##        ID        Reason.for.absence Month.of.absence Day.of.the.week
##  Min.   : 1.00   Min.   : 0.00      Min.   : 0.000   Min.   :2.000
##  1st Qu.: 9.00   1st Qu.:13.00      1st Qu.: 3.000   1st Qu.:3.000
##  Median :18.00   Median :23.00      Median : 6.000   Median :4.000
##  Mean   :18.02   Mean   :19.22      Mean   : 6.324   Mean   :3.915
##  3rd Qu.:28.00   3rd Qu.:26.00      3rd Qu.: 9.000   3rd Qu.:5.000
##  Max.   :36.00   Max.   :28.00      Max.   :12.000   Max.   :6.000
##     Seasons      Transportation.expense Distance.from.Residence.to.Work
##  Min.   :1.000   Min.   :118.0          Min.   : 5.00
##  1st Qu.:2.000   1st Qu.:179.0          1st Qu.:16.00
##  Median :3.000   Median :225.0          Median :26.00
##  Mean   :2.545   Mean   :221.3          Mean   :29.63
##  3rd Qu.:4.000   3rd Qu.:260.0          3rd Qu.:50.00
##  Max.   :4.000   Max.   :388.0          Max.   :52.00
##   Service.time        Age        Work.load.Average.day   Hit.target
##  Min.   : 1.00   Min.   :27.00   Min.   :205.9         Min.   : 81.00
##  1st Qu.: 9.00   1st Qu.:31.00   1st Qu.:244.4         1st Qu.: 93.00
##  Median :13.00   Median :37.00   Median :264.2         Median : 95.00
##  Mean   :12.55   Mean   :36.45   Mean   :271.5         Mean   : 94.59
##  3rd Qu.:16.00   3rd Qu.:40.00   3rd Qu.:294.2         3rd Qu.: 97.00
##  Max.   :29.00   Max.   :58.00   Max.   :378.9         Max.   :100.00
##  Disciplinary.failure   Education          Son        Social.drinker
##  Min.   :0.00000      Min.   :1.000   Min.   :0.000   Min.   :0.0000
##  1st Qu.:0.00000      1st Qu.:1.000   1st Qu.:0.000   1st Qu.:0.0000
##  Median :0.00000      Median :1.000   Median :1.000   Median :1.0000
##  Mean   :0.05405      Mean   :1.292   Mean   :1.019   Mean   :0.5676
##  3rd Qu.:0.00000      3rd Qu.:1.000   3rd Qu.:2.000   3rd Qu.:1.0000
##  Max.   :1.00000      Max.   :4.000   Max.   :4.000   Max.   :1.0000
##  Social.smoker          Pet             Weight          Height
##  Min.   :0.00000   Min.   :0.0000   Min.   : 56.00   Min.   :163.0
##  1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.: 69.00   1st Qu.:169.0
##  Median :0.00000   Median :0.0000   Median : 83.00   Median :170.0
```

```
##  Mean   :0.07297   Mean   :0.7459   Mean   : 79.04   Mean   :172.1
##  3rd Qu.:0.00000   3rd Qu.:1.0000   3rd Qu.: 89.00   3rd Qu.:172.0
##  Max.   :1.00000   Max.   :8.0000   Max.   :108.00   Max.   :196.0
##  Body.mass.index Absenteeism.time.in.hours
##  Min.   :19.00   Min.   :  0.000
##  1st Qu.:24.00   1st Qu.:  2.000
##  Median :25.00   Median :  3.000
##  Mean   :26.68   Mean   :  6.924
##  3rd Qu.:31.00   3rd Qu.:  8.000
##  Max.   :38.00   Max.   :120.000
```

The range for the different variables can be taken from the results after running the *summary* function. Zero in Reason.for.absence for absence is not a valid category. ICD codes do not support it.

```
#Remove observations zero in Reason code.
range(myData$Reason.for.absence)
```

```
## [1]  0 28
```

```
myData <- myData[!(myData$Reason.for.absence == 0),]
```

Also in the Absenteeism.time.in.hours there is value zero which is not logical.

```
# Observations in which reason code is zero and absenteeism time is zero.
time0 <- subset(myData, Absenteeism.time.in.hours <= 0 & Reason.for.absence > 0,
                c(ID, Reason.for.absence, Absenteeism.time.in.hours))
print(time0)
```

```
##     ID Reason.for.absence Absenteeism.time.in.hours
## 135 34                 27                         0
```

It is obvious that for ID 34 and reason 27 is is the only one where Absenteeism time is zero.

```
#Removed zero observation
myData <- myData[!(myData$Absenteeism.time.in.hours==0 & myData$Reason.for.absence > 0) ,]
```

At this point, our dataset have 969 observations and 21 variables, so we would have another close look.

```
dim(myData)
```

```
## [1] 696  21
```

```
glimpse(myData,width=80,strict.width="cut")
```

```
## Rows: 696
## Columns: 21
## $ ID                             <int> 11, 3, 7, 11, 3, 10, 20, 14, 1, 20,...
## $ Reason.for.absence             <int> 26, 23, 7, 23, 23, 22, 23, 19, 22, ...
## $ Month.of.absence               <int> 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7,...
## $ Day.of.the.week                <int> 3, 4, 5, 5, 6, 6, 6, 2, 2, 2, 3, 4,...
```

```
## $ Seasons                          <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ Transportation.expense           <int> 289, 179, 279, 289, 179, 361, 260, ...
## $ Distance.from.Residence.to.Work <int> 36, 51, 5, 36, 51, 52, 50, 12, 11, ...
## $ Service.time                     <int> 13, 18, 14, 13, 18, 3, 11, 14, 14, ...
## $ Age                              <int> 33, 38, 39, 33, 38, 28, 36, 34, 37,...
## $ Work.load.Average.day            <dbl> 239.554, 239.554, 239.554, 239.554,...
## $ Hit.target                       <int> 97, 97, 97, 97, 97, 97, 97, 97, 97,...
## $ Disciplinary.failure             <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Education                        <int> 1, 1, 1, 1, 1, 1, 1, 1, 3, 1, 1, 1,...
## $ Son                              <int> 2, 0, 2, 2, 0, 1, 4, 2, 1, 4, 4, 4,...
## $ Social.drinker                   <int> 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1,...
## $ Social.smoker                    <int> 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Pet                              <int> 1, 0, 0, 1, 0, 4, 0, 0, 1, 0, 0, 0,...
## $ Weight                           <int> 90, 89, 68, 90, 89, 80, 65, 95, 88,...
## $ Height                           <int> 172, 170, 168, 172, 170, 172, 168, ...
## $ Body.mass.index                  <int> 30, 31, 24, 30, 31, 27, 23, 25, 29,...
## $ Absenteeism.time.in.hours        <int> 4, 2, 4, 2, 2, 8, 4, 40, 8, 8, 8, 8...
```

Having a look at the data set, it is obvious that Disciplinary.failure is a column with zero values which can be considered as a noise. So we remove that attribute.

```
#Remove Disciplinary failure attribute.
myData <- myData[,-12]
```

At the last part of data cleaning we would make a missing value analysis to make sure our data set is ready for analysis.

```
#Missing value analysis
sum(is.na(myData))
```

```
## [1] 0
```

There is no missing value in any part of the data set. Thus the data frame is ready for the data analysis.

5 Data Analysis

In order to analyse the given data, an approach is taken to gain an overview of the data volume with the help of classifications. This should make it possible to identify correlations and to further analyse only those groups for which the analysis generates the greatest added value.

This means that it is decided which groups are to be analysed further and which are not. Only then is an attempt made to find out how the respective characteristics of these people are related to the diseases and, subsequently, to their absence.

5.1 Overview

First, the frequencies of the periods of absence are shown in a bar plot.

```
ggplot(myData, aes(Absenteeism.time.in.hours))+ geom_bar() +
  scale_x_continuous(n.breaks = 10) + scale_y_continuous(n.breaks = 10) +
  labs(x = "Absent time in hours", y = "Frequency")
```

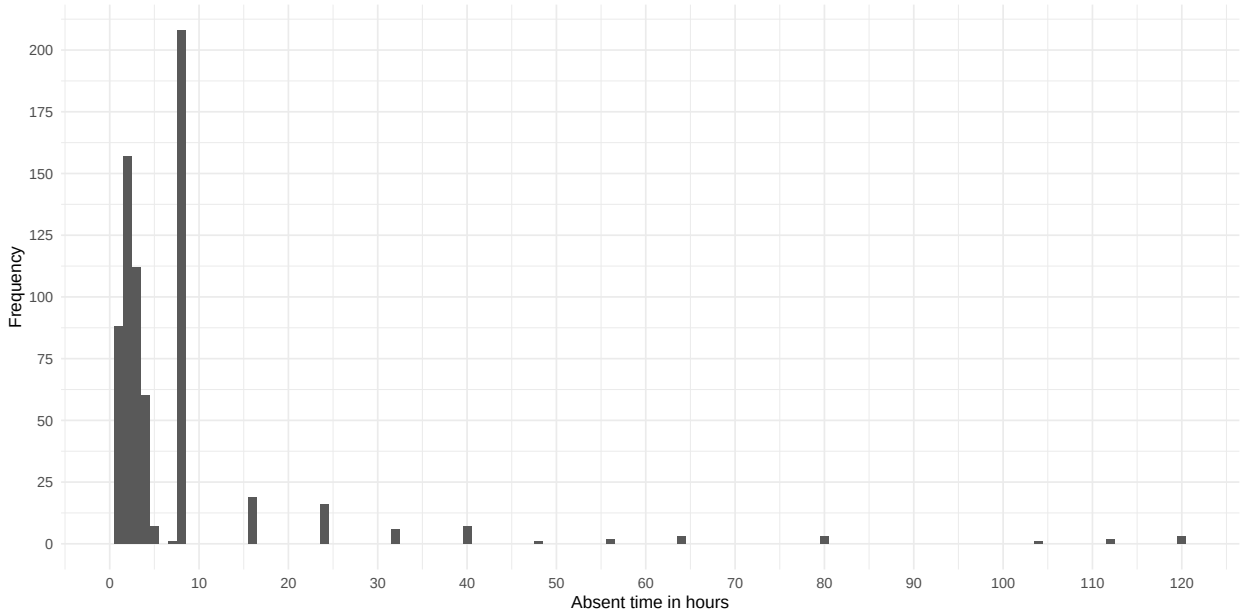

Figure 1: Frequency of absent times

As seen in the graph the most employees are absent in the range from one to seven hours. To better understand the absenteeism, the employees are grouped into different groups according to their total absenteeism time in hours:

Time category	Absent time	Absent time in hours
VS - very short	1-2 days	0-16
S - short	2 days - 1 week	16-40
L - long	1-2 weeks	40-80
VL - very long	more than 3 weeks	more than 80

```
VS <- filter(myData, Absenteeism.time.in.hours <= 16)
Absenteeism.group = 'VS'
VS <- cbind(VS, Absenteeism.group)

S <- filter(myData, Absenteeism.time.in.hours > 16 & Absenteeism.time.in.hours <= 40)
Absenteeism.group = 'S'
S <- cbind(S, Absenteeism.group)

L <- filter(myData, Absenteeism.time.in.hours > 40 & Absenteeism.time.in.hours <= 80)
Absenteeism.group = 'L'
L <- cbind(L, Absenteeism.group)

VL <- filter(myData, Absenteeism.time.in.hours > 80)
Absenteeism.group = 'VL'
VL <- cbind(VL, Absenteeism.group)

myData1 <- rbind(VS, S, L, VL)
```

First, the size of the different groups is compared and the total absent time of the persons in the groups to see the economical impacts of the groups.

```r
#Frequency of diseases
nrow(VS)
nrow(S)
nrow(L)
nrow(VL)

#calculating the total absent time
sum(VS$Absenteeism.time.in.hours)
sum(S$Absenteeism.time.in.hours)
sum(L$Absenteeism.time.in.hours)
sum(VL$Absenteeism.time.in.hours)
```

Time category	Frequency of diseases	Total absent time
VS - very short	652	2988
S - short	29	856
L - long	9	592
VL - very long	6	688

In the **VS** group **652** persons are absent, in the **S** group **29** people, in the **L** group **9** people and in the **VL** group **6** people. It seems that the biggest impact can be made in the VS group, therefore the focus will be on this group.

Next, the most common reasons for the absence of these people are presented.

```r
ggplot(VS, aes(x=(Reason.for.absence))) +
  geom_histogram(binwidth = 0.5) +
  labs(x = "Reason for absence", y = "Frequency") +
  scale_x_continuous(breaks = seq(1, 28, by=1)) +
  scale_y_continuous(n.breaks = 10)
```

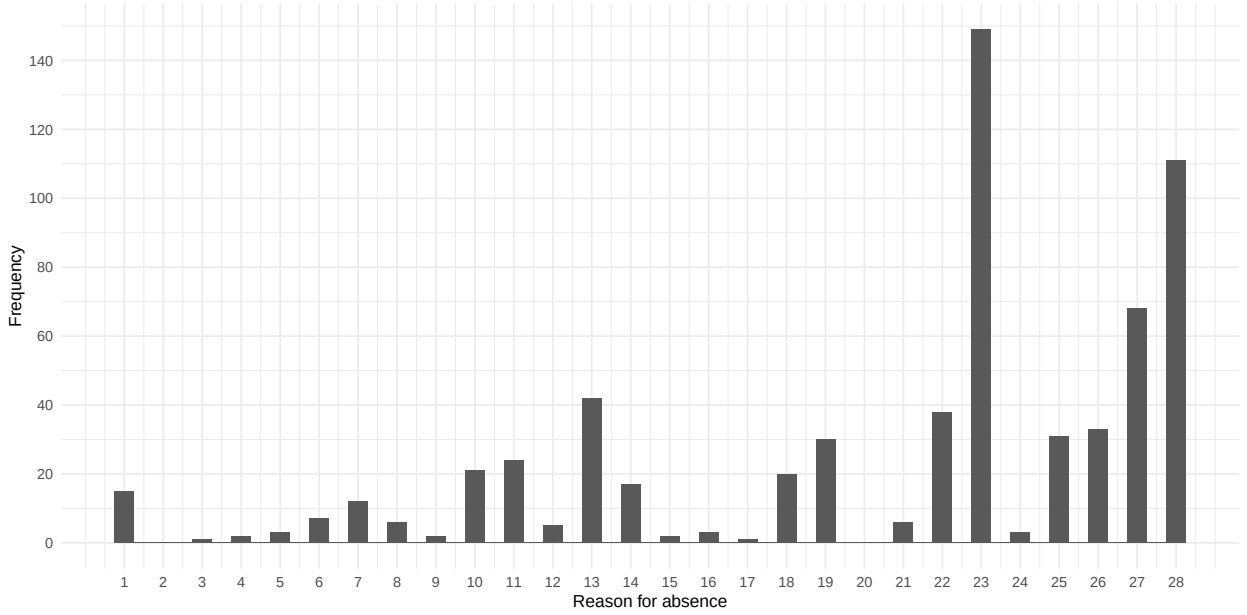

Figure 2: Frequency of several diseases

It is clear that the reasons 13,22,23,27 and 28 are very common and may be the reason for the high number of absences in this group. However, in order to find out how many hours of absence are really caused by this, the hours for each reason of absence have to be summed up.

```
# Sum up all the hours of absence for the several reasons
Reasons=c()
Hours=c()

for (i in 1:28)
{
  Reason <- subset(VS, Reason.for.absence == i)
  HoursReason <- sum(Reason$Absenteeism.time.in.hours)

  Reasons[i]=i
  Hours[i]=HoursReason
}
#make a new data frame
HoursReasons = data.frame(Reasons, Hours)
#sort the Reasons column based on Hours
HoursReasons$Reasons <- factor(HoursReasons$Reasons,
                               levels = HoursReasons$Reasons[order(-HoursReasons$Hours)])

ggplot(HoursReasons, aes(x = Reasons, y = Hours)) +
  geom_col(width = 0.5) +
 labs(x = "Reason for absence", y = "Total hours") +
 geom_hline(yintercept=200, linetype="dashed", color = "red")
```

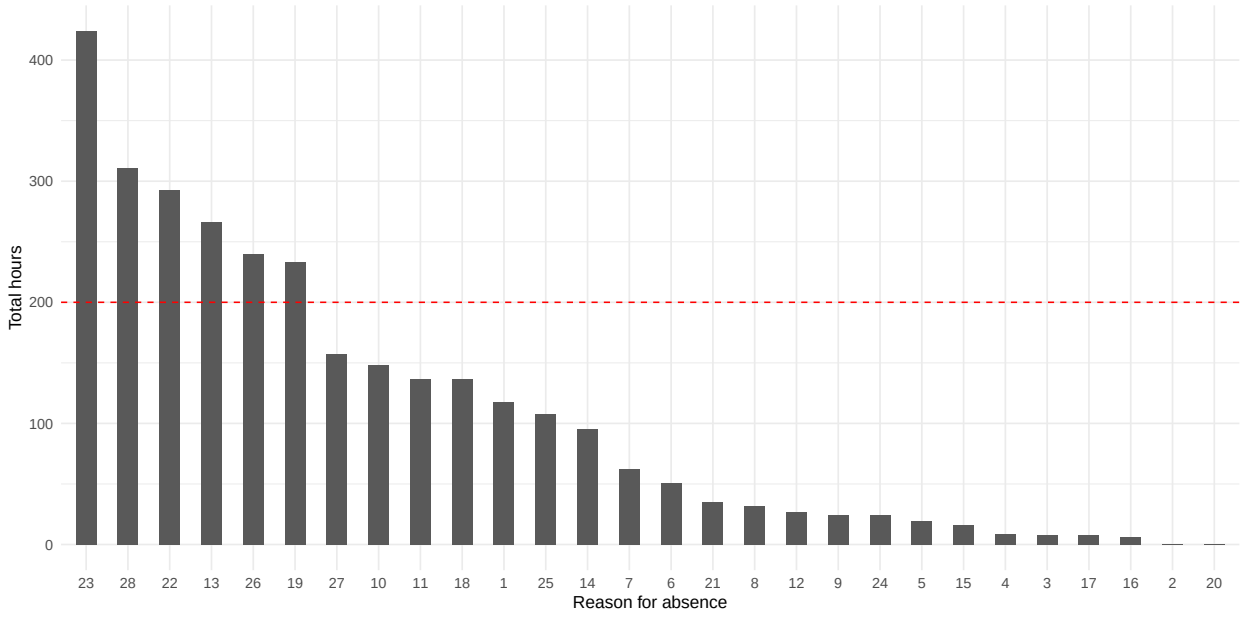

Figure 3: Total hours of absence per reason

A tangible picture now emerges of which reasons for absence cause the most hours of absence and further restrictions can be placed on the group to be analysed.

Of particular interest are the top 6 of those reasons that each cause more than 200 hours of absence. For the

segment to be analysed, therefore, only those persons who were absent for the following reasons are taken into account:

Reason	name
23	Medical consultation
28	Dental consultation
22	Patient follow-up
13	Diseases of the musculoskeletal system and connective tissue
26	Unjustified absence
19	Injury, poisoning and certain other consequences of external causes

```
#Put all the relevant Reasons in a subset

StudyGroup <- HoursReasons %>% top_n(6, Hours)

#prepare the StudyGroup data frame for further analysis
StudyData <- VS %>% filter(Reason.for.absence %in% c("23","28","22","13","26","19"))
str(StudyData)
```

```
## 'data.frame':    403 obs. of  21 variables:
##  $ ID                              : int  11 3 11 3 10 20 1 3 3 33 ...
##  $ Reason.for.absence              : int  26 23 23 23 22 23 22 23 23 23 ...
##  $ Month.of.absence                : int  7 7 7 7 7 7 7 7 7 8 ...
##  $ Day.of.the.week                 : int  3 4 5 6 6 6 2 4 6 4 ...
##  $ Seasons                         : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Transportation.expense          : int  289 179 289 179 361 260 235 179 179 248 ...
##  $ Distance.from.Residence.to.Work: int  36 51 36 51 52 50 11 51 51 25 ...
##  $ Service.time                    : int  13 18 13 18 3 11 14 18 18 14 ...
##  $ Age                             : int  33 38 33 38 28 36 37 38 38 47 ...
##  $ Work.load.Average.day           : num  240 240 240 240 240 ...
##  $ Hit.target                      : int  97 97 97 97 97 97 97 97 97 92 ...
##  $ Education                       : int  1 1 1 1 1 1 3 1 1 1 ...
##  $ Son                             : int  2 0 2 0 1 4 1 0 0 2 ...
##  $ Social.drinker                  : int  1 1 1 1 1 1 0 1 1 0 ...
##  $ Social.smoker                   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Pet                             : int  1 0 1 0 4 0 1 0 0 1 ...
##  $ Weight                          : int  90 89 90 89 80 65 88 89 89 86 ...
##  $ Height                          : int  172 170 172 170 172 168 172 170 170 165 ...
##  $ Body.mass.index                 : int  30 31 30 31 27 23 29 31 31 32 ...
##  $ Absenteeism.time.in.hours       : int  4 2 2 2 8 4 8 4 2 2 ...
##  $ Absenteeism.group               : chr  "VS" "VS" "VS" "VS" ...
```

By the initial analysis it is shown that, the most frequently occurring reason for absence are the category “23”,“28”,“22”,“13”,“26”,“19”. This answer gives us a glimpse of absenteeism. The next step is to prove this claim by checking our distributions, frequencies and outright correlated reasons for absence.

5.2 Correlations

From now on all the analysis will be carried on the **StudyData** data frame. After the initial analysis and having a broad view of the main data, now it is time to find the proper correlation between different variables. Below a correlogram is made which is a graph of correlation matrix. It is a useful tool to highlight the most correlated variables in a data table.

```
StudyData <- StudyData[,-21] #remove the chr variable
DataCor <- cor(StudyData)    #Compute correlation matrix
corrplot(abs(cor(DataCor)), type="upper", tl.col="black",
         tl.cex = .7, col=brewer.pal(n=9, name="Reds"))
```

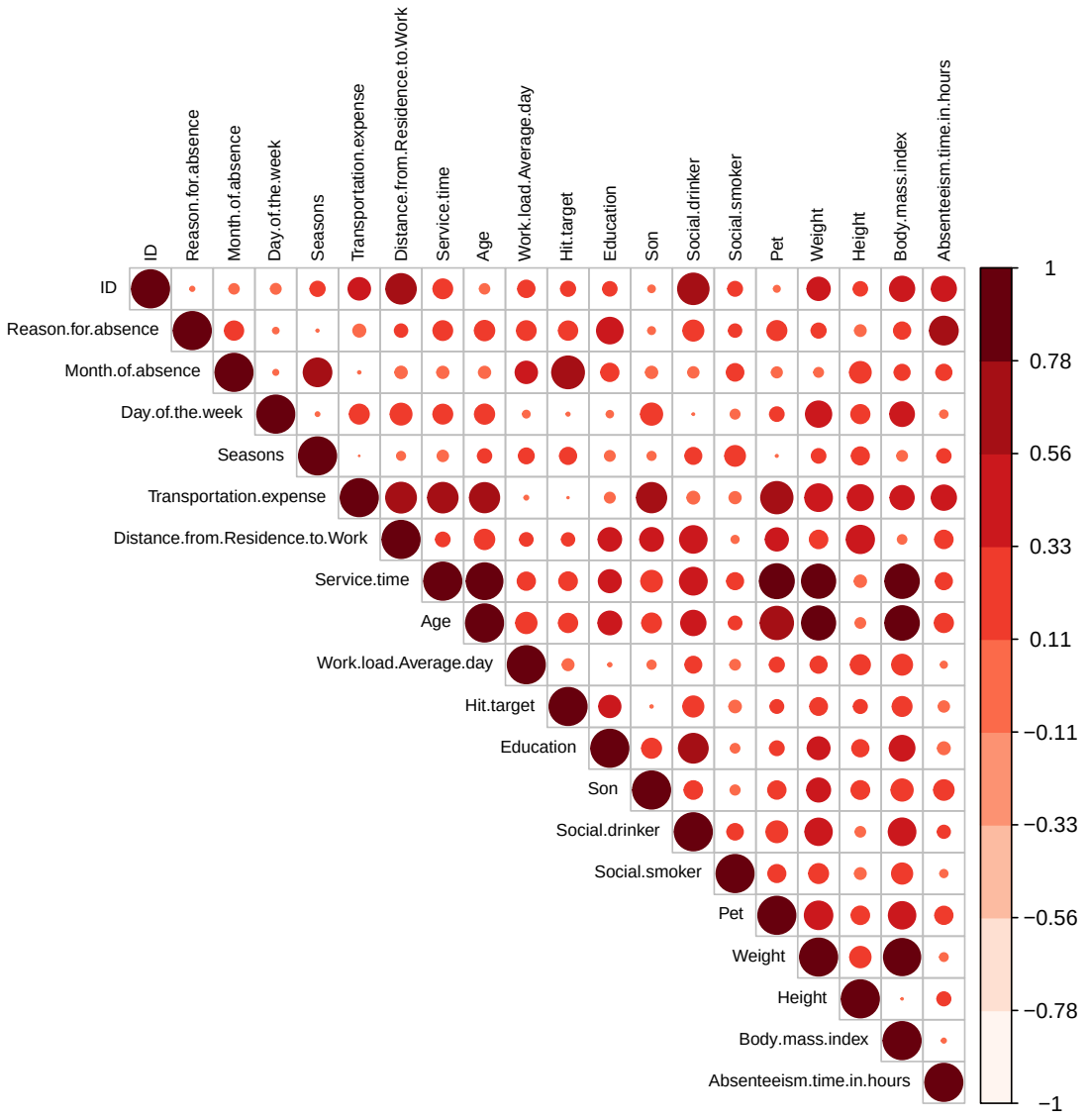

Figure 4: Correlation between variables

Figure 4 illustrates the correlation heatmap with considering the selected data. Darker and bigger circles represent positive correlation and lighter ones show negative correlation with respect to other factors. Height is highly related to the Body.mass.index and also there is correlation between Age and Service.time. Furthermore, no other variable in highly correlated with another factor. By considering these, in the next step a hierarchical and a k-means clustering is carried out to test if the classifications of absenteeism explains the

values of the variables in the data set.

5.3 Clustering and K-means

The idea here is to cluster cases, in other words, to group cases. Therefore the "Complete linkage" clustering method is used that defines the cluster distance between two clusters to be the maximum between their individual components. At every stage of the clustering process, the two nearest clusters are merged into a new cluster. The process is repeated until the whole data set is agglomerated into one single cluster. By creating a physical profile for this group we would have a close look at their absence for provided data set. The function *hclust* is used to identify the method and the *dist* function is used to find the distance. In this report the complete linkage is carried on with Euclidean method for clustering.

```
#Create physical profile
Weight<-StudyData$Weight
Height<-StudyData$Height
PhysicalProfile<-data.frame(Height,Weight)
PhysicalProfile1<-data.frame(Height,Weight)
PhysicalProfile$ID<-StudyData$ID
PhysicalProfile$Reason<-StudyData$Reason.for.absence
PhysicalProfile$BMI<-StudyData$Body.mass.index
PhysicalProfile$Age<-StudyData$Age
PhysicalProfile$Work.load.Average.day <-StudyData$Work.load.Average.day
PhysicalProfile$Distance.from.Residence.to.work<-StudyData$Distance.from.Residence.to.Work
PhysicalProfile$Transportation.expense<-StudyData$Transportation.expense
PhysicalProfile$Service.time<-StudyData$Service.time
PhysicalProfile$Day.of.the.week<-StudyData$Day.of.the.week
PhysicalProfile$Month.of.absence<-StudyData$Month.of.absence

BMI<-table(StudyData$Body.mass.index)
BMI1<-as.data.frame(BMI)
```

After obtaining a physical profile, then we can illustrate the clustering by a dendrogram.

```
PPdistE<-dist(PhysicalProfile1, method="euclidean")
PPhclustEC<-hclust(PPdistE, "complete")
plot(PPhclustEC,main= "Euclidean \nComplete Linkage")
rect.hclust(PPhclustEC,k=7,border="red")
```

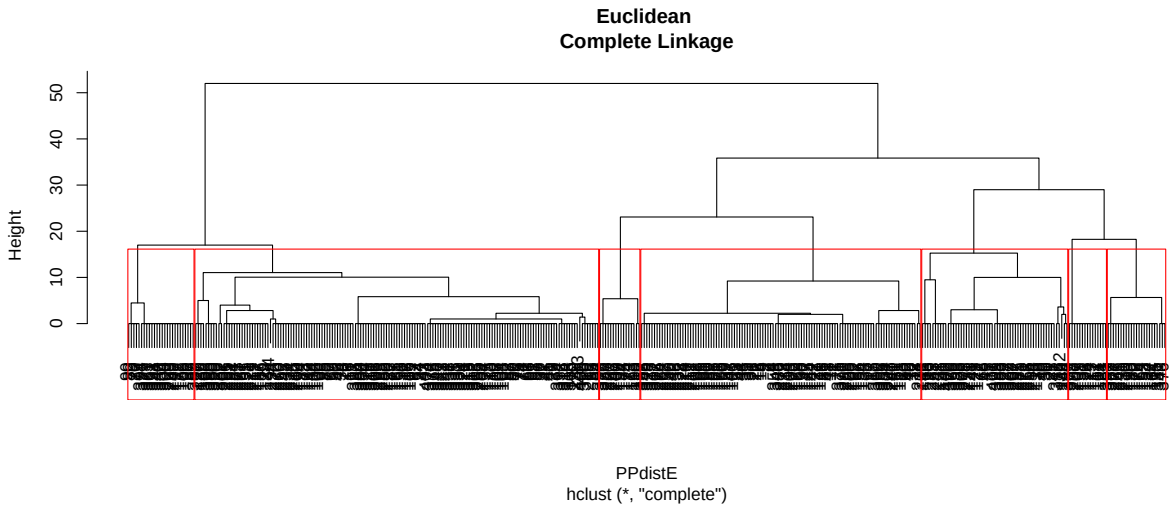

Figure 5: Dendrogram for physical profile

The above dendrogram have seven red boxes to indicate where we are cutting the cluster tree and determine our cluster amount.

```
# Build Hclust and K-means Data
Hclust1 <- cutree(PPhclustEC,k=7)
PhysicalProfile$Groups <- Hclust1
StudyData$Groups <- Hclust1
kmM <- kmeans(dist(PhysicalProfile1,
        method="manhattan"),centers=7,nstart=70)

# k-means vs hierchical cluster comparisons
kmMFrame <- data.frame(StudyData, cluster=factor(kmM$cluster))
P1 <- ggplot(kmMFrame, aes(x=Weight, y=Height, color=cluster)) + geom_point(size = 3) +
  theme(legend.position="none")

ClusterFrame <- data.frame(StudyData, cluster=factor(StudyData$Groups))
P2 <- ggplot(ClusterFrame, aes(x=Weight, y=Height, color=cluster)) + geom_point(size = 3)

grid.arrange(P1,P2, ncol =2, left = "K-Means", right = "Hierarchical Cluster")
```

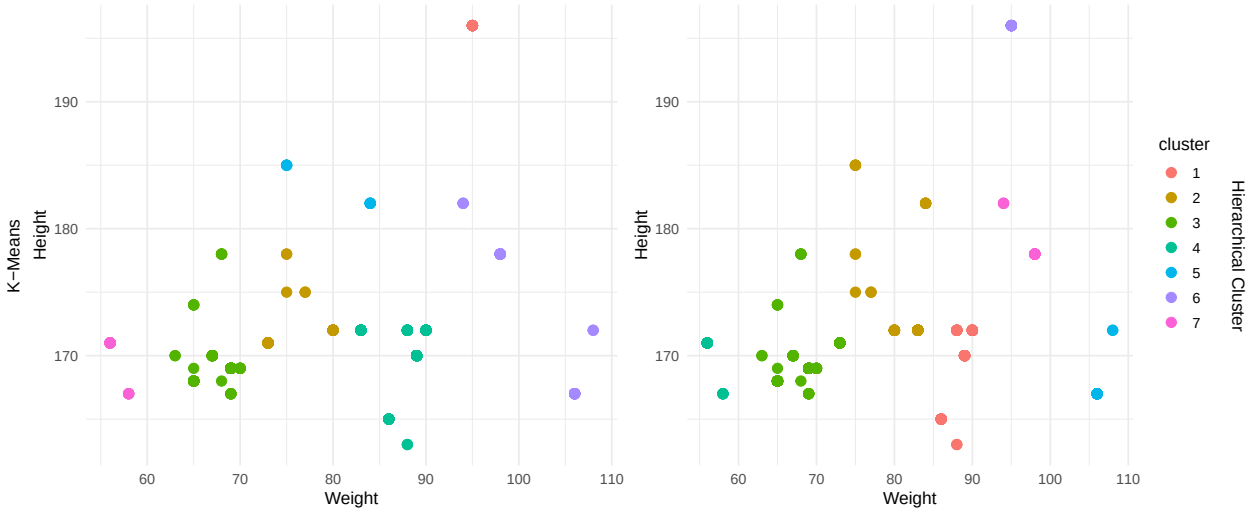

Figure 6: k-means vs hierchical cluster comparisons

Now that we have determined our hierarchical clustering method by Euclidean approach, it is needed to compare our hierarchical model to the K means algorithm. K means clustering works the opposite from hierarchical clustering in that it works from the top down by placing centers in the data first before it can be clustered. Hierarchical is a bottom-up approach shown by the previous dendrogram. Comparing these two techniques, the hierarchical method has proven to cluster better than the K means even after 70 iterations to test the centers "centriods" for accuracy in the code shows (centers=7,nstart=70).

```
#Histogram with density plot for Age
ggplot(PhysicalProfile, aes(x=Age)) +
 geom_histogram(aes(y=..density..), colour="black", fill="darkgrey")+
 geom_density(alpha=.2, fill="red") +
  geom_vline(aes(xintercept=mean(Age)), color="blue",
             linetype="dashed")
```

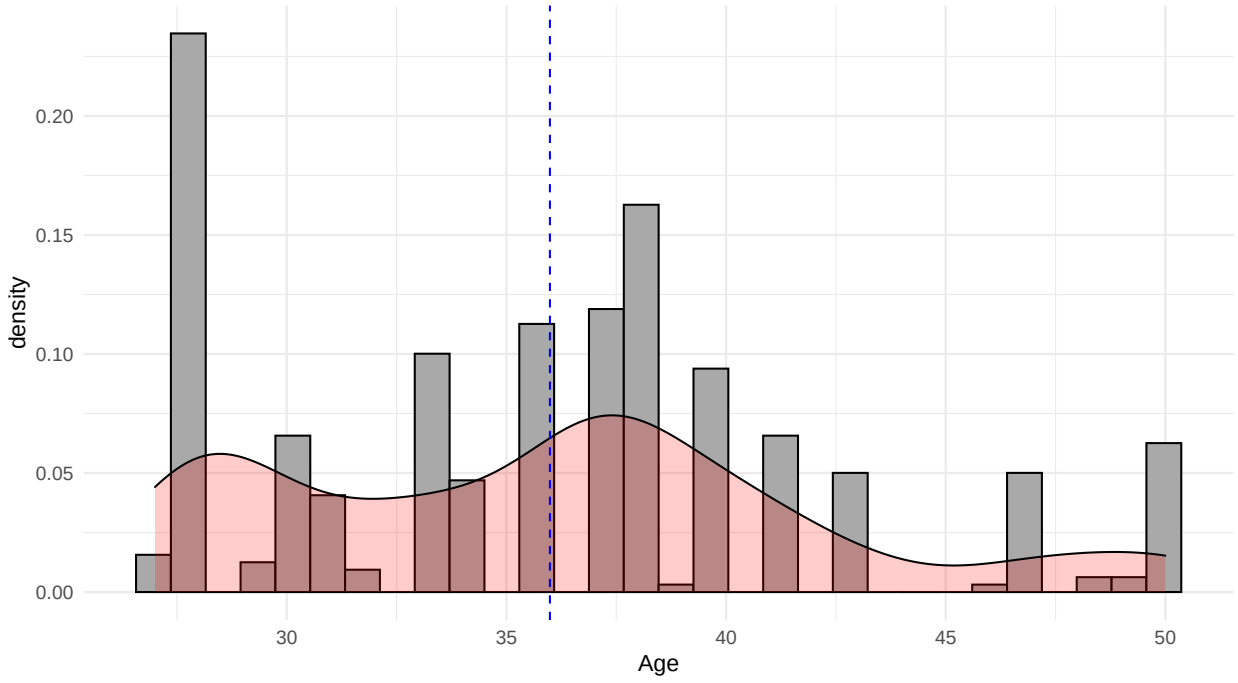

Figure 7: Age density

From figure 7 it is understandable that the Age distribution for the Study Data does not have pattern, however the *mean* is around 36 years.

StudyData people are now clustered into groups. In order to achieve a more general distribution, the persons are divided according to their BMI, i.e. height and weight.

```
#test clustered categories for physical profile answers
stackbar <- with(PhysicalProfile, table(Reason,Groups))
stackbar <- as.data.frame(stackbar)
stackbar$Reason <- revalue(stackbar$Reason,
                  c("13"="Diseases of musculoskeletal and connective tissue",
                    "23"="Medical consultation",
                    "28"="Dental consultation",
                    "22"="Patient follow-up",
                    "26"="Unjustified absence",
                    "19"="Injury, poisoning and other external causes"))

ggplot(stackbar, aes(x=Groups, y=Freq, fill=Reason)) +
    geom_bar(position = "stack", stat = "identity") +
  scale_fill_brewer(palette = "RdYlBu")
```

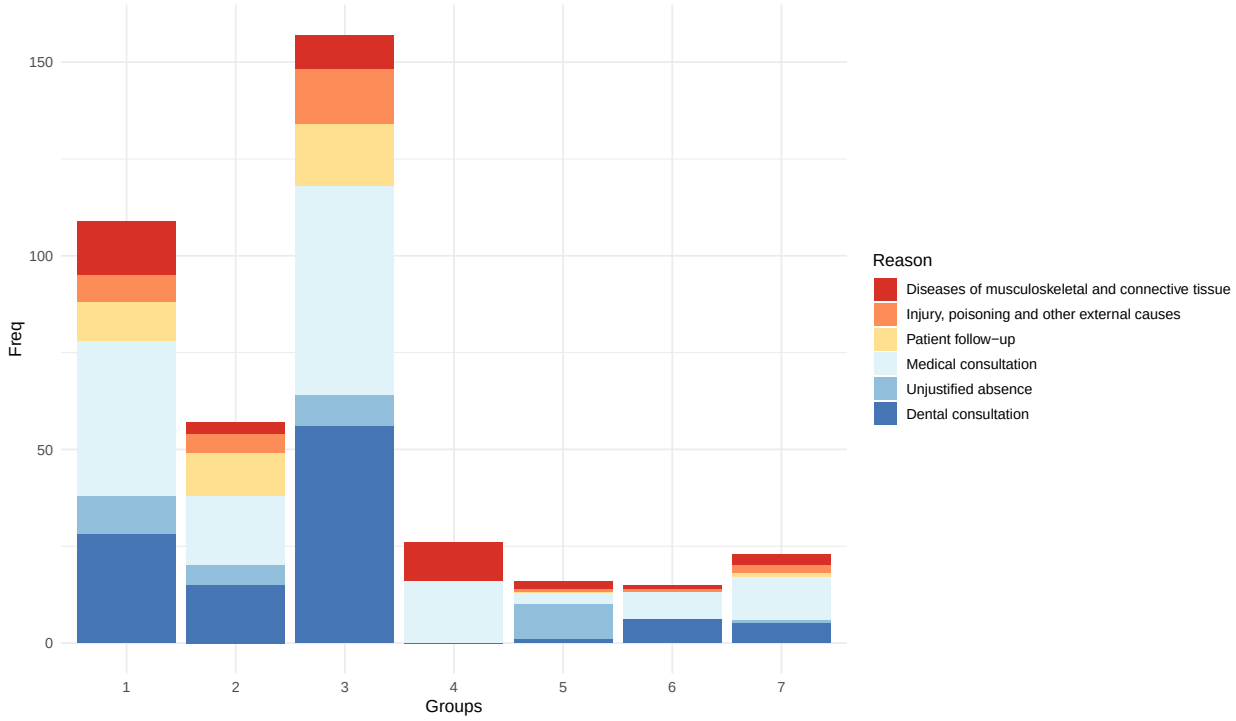

Figure 8: Absense Reasons Per Cluster

The above plot is broken down into the seven clusters. As it is illustrated clusters one, two and three have the majority of employee absenteeism reason frequencies. To find out in which variables groups 1, 2 and 3 differ from the others, the mean values and scatter of the following variables are analysed:

```
P1 <- ggplot(PhysicalProfile, aes(x=as.factor(Groups), y=BMI)) +
      geom_boxplot() +
      stat_summary(fun = "mean", color="red", shape=15) +
      geom_jitter(alpha=0.5, width = 0.15) +
 labs(x = "Cluster Groups") +
  theme_bw()

P2 <- ggplot(PhysicalProfile, aes(x=as.factor(Groups), y=Age)) +
      geom_boxplot() +
      stat_summary(fun = "mean", color="red", shape=15) +
      geom_jitter(alpha=0.5, width = 0.15) +
 labs(x = "Cluster Groups") +
  theme_bw()

P3 <- ggplot(PhysicalProfile, aes(x=as.factor(Groups),
                                  y=Work.load.Average.day)) +
      geom_boxplot() +
      stat_summary(fun = "mean", color="red", shape=15) +
      geom_jitter(alpha=0.5, width = 0.15) +
 labs(x = "Cluster Groups")+
  theme_bw()

P4 <- ggplot(PhysicalProfile, aes(x=as.factor(Groups),
                                  y=Distance.from.Residence.to.work)) +
```

```
      geom_boxplot() +
      stat_summary(fun = "mean", color="red", shape=15) +
      geom_jitter(alpha=0.5, width = 0.15) +
 labs(x = "Cluster Groups")+
  theme_bw()

P5 <- ggplot(PhysicalProfile, aes(x=as.factor(Groups),
                                  y=Transportation.expense)) +
      geom_boxplot() +
      stat_summary(fun = "mean", color="red", shape=15) +
      geom_jitter(alpha=0.5, width = 0.15) +
 labs(x = "Cluster Groups")+
  theme_bw()

P6 <- ggplot(PhysicalProfile, aes(x=as.factor(Groups),
                                  y=Service.time)) +
      geom_boxplot() +
      stat_summary(fun = "mean", color="red", shape=15) +
      geom_jitter(alpha=0.5, width = 0.15) +
 labs(x = "Cluster Groups")+
  theme_bw()

P7 <- ggplot(PhysicalProfile, aes(x=as.factor(Groups),
                                  y=Day.of.the.week)) +
      geom_boxplot() +
      stat_summary(fun = "mean", color="red", shape=15) +
      geom_jitter(alpha=0.5, width = 0.15) +
 labs(x = "Cluster Groups")+
  theme_bw()

P8 <- ggplot(PhysicalProfile, aes(x=as.factor(Groups),
                                  y=Month.of.absence)) +
      geom_boxplot() +
      stat_summary(fun = "mean", color="red", shape=15) +
      geom_jitter(alpha=0.5, width = 0.15) +
 labs(x = "Cluster Groups")+
  theme_bw()



grid.arrange(P1,P2,P3,P4, ncol =2)
```

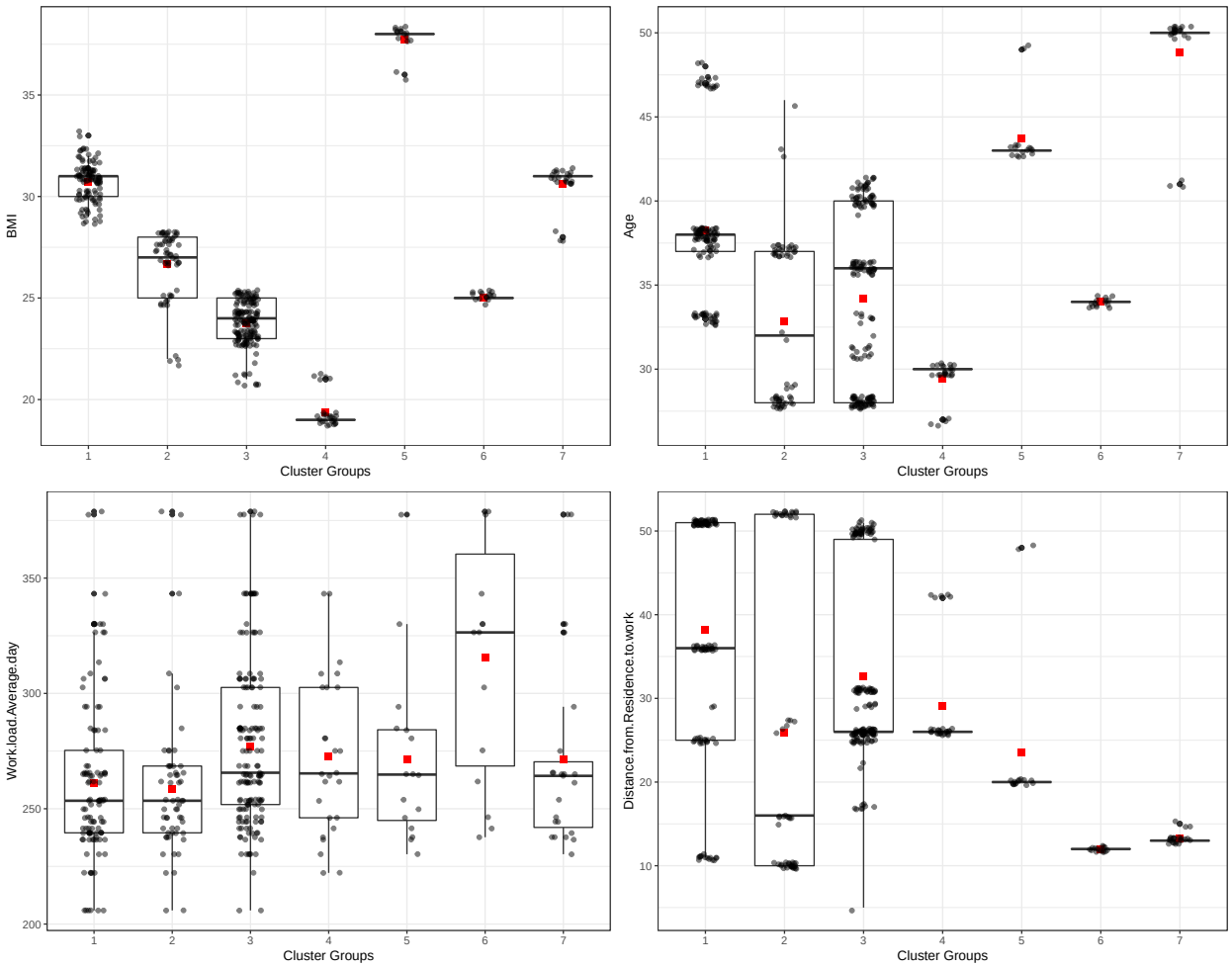

Figure 9: BoxPlots of Variables per Cluster

```
grid.arrange(P5,P6,P7,P8, ncol=2)
```

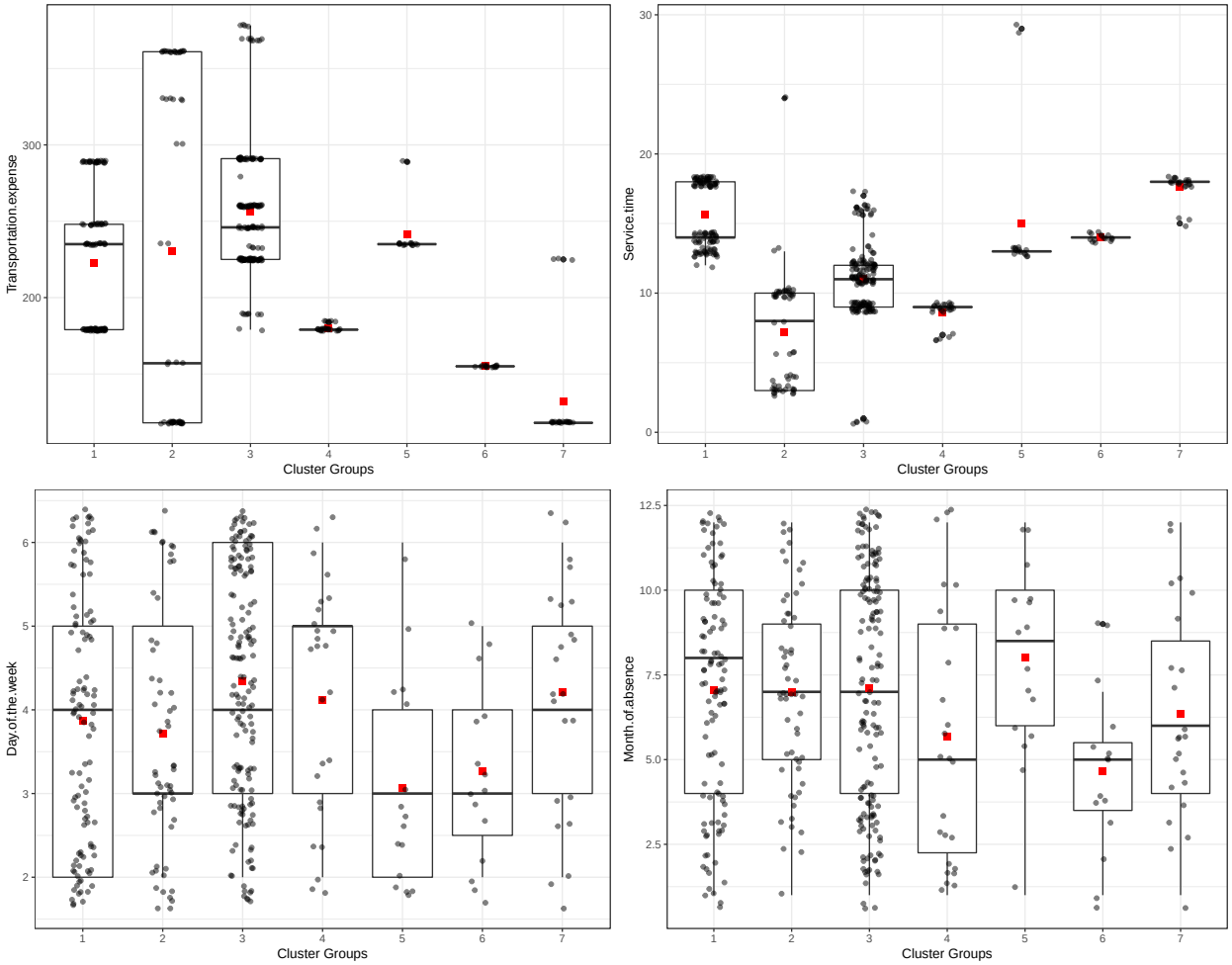

Figure 10: BoxPlots of Variables per Cluster

As can be seen from the above diagrams, groups 1, 2 and 3 differ mainly in the variables BMI, Distance to work and Transportation expense. As the groups have been subdivided according to height and weight, it is clear that the BMI value differs greatly. The distance from work and transportation expenses, however, indicate that those employees who have a long way to work are absent more often.

6 Conclusion

This concludes our exploratory data analysis using graphs and clustering analysis. In this report the analysis approach was hierarchical clustering using the *euclidean* method and the complete linkage. In this clustering, we were able to see more clear what physical profiles and age groups were contributing to employee absenteeism. This method can be taken into account for different grouping and further analysis.

In order to address possible reasons for the frequency of absences of groups 1, 2 and 3, it must first be mentioned that clustering also allocated the most people to these groups and therefore more absences are recorded. Nevertheless, the deviation in distance and transportation expense is noticeable.

The frequent absence of groups 1, 2 and 3 may be due to the fact that the greater distance to the workplace causes a motivation problem and that these employees go to a doctor more quickly in the case of mild symptoms of illness in order to be put on sick leave. If one looks at the reasons for absence in these 3 groups, it is noticeable that **medical consultation** is a main reason for absence.

6.1 Measures

In order to counteract the frequent absences of these employees, various measures can be taken and after a few months it can be checked whether these actually reduce absences:

- Home office

In order to make the employees' everyday work more pleasant and to save them expensive transport costs, the possibilities of operating a home office should be extended.

- Shuttle service

For employees who are unable to work from home, a shuttle service should be set up at the company's expense. For this purpose, employees can be grouped according to their place of residence and can be brought to and from the company by a bus.

7 References

[1] R Core Team (2019). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

[2] Wickham et al., (2019). Welcome to the tidyverse. Journal of Open Source Software, 4(43), 1686, https://doi.org/10.21105/joss.01686

[3] Hao Zhu (2020). kableExtra: Construct Complex Table with 'kable' and Pipe Syntax. R package version 1.2.1. https://CRAN.R-project.org/package=kableExtra

[4] Erich Neuwirth (2014). RColorBrewer: ColorBrewer Palettes. R package version 1.1-2. https://CRAN.R-project.org/package=RColorBrewer

[5] Taiyun Wei and Viliam Simko (2017). R package "corrplot": Visualization of a Correlation Matrix (Version 0.84). Available from https://github.com/taiyun/corrplot

[6] Maechler, M., Rousseeuw, P., Struyf, A., Hubert, M., Hornik, K.(2019). cluster: Cluster Analysis Basics and Extensions. R package version 2.1.0.

[7] Alboukadel Kassambara and Fabian Mundt (2020). factoextra: Extract and Visualize the Results of Multivariate Data Analyses. R package version 1.0.7. https://CRAN.R-project.org/package=factoextra

[8] Tal Galili (2015). dendextend: an R package for visualizing, adjusting, and comparing trees of hierarchical clustering. Bioinformatics. DOI: 10.1093/bioinformatics/btv428

[9] Baptiste Auguie (2017). gridExtra: Miscellaneous Functions for "Grid" Graphics. R package version 2.3. https://CRAN.R-project.org/package=gridExtra

[10] Hadley Wickham (2011). The Split-Apply-Combine Strategy for Data Analysis. Journal of Statistical Software, 40(1), 1-29. URL http://www.jstatsoft.org/v40/i01/.