Absenteeism at work prediction-CSE780

Paria, Fakhrzad - Student ID: 400353290

Professor: Dr. Pratheepa Jeganathan

04-Dec-2021

Contents

1 Introduction

1.1 literature review on absenteeism at work

This project is about an issue at companies when employees are frequently absent at work. We found that many studies carried out absenteeism as an important concern that companies need to struggle with. An statistical data shows that the rate of absenteeism increased steadily during $1997 - 2007$ in Canada [4], this article reviewed causes, effect and cures for this issue. These papers [3], and [5], have literature review to show the importance of analyzing reasons of absenteeism as a measurable problem that has cost for any organization and negatively affect in financial part. Addressed to psychological part of this problem [6] analyze individuals for finding three archetypes of attendance cultures,health focused, individual decision and presentistic. These cures that have presented by mentioned studies, most help to decrease absenteeism but deep understanding of employees and their behavior algorithm definitely helps to predict their absenteeism and apply more compatible cure before it happens. ALso these cures should be applied for employees based on their clusters and the same cure doesn't always works for all.

1.2 literature review on machine learning in absenteeism

We saw many machine learning algorithm has applied for this purpose, the algorithms That have been used by Richardo is neural network for prediction of absenteeism [2], in their model the mean error was 0.95. There is a post in kaggle by Kozak used sklearn library in python to predict the absenteeism and their MSE was 2629, shows that their model didn't work well.

In this project we are going to focus on this subject as an measurable problem and use machine learning algorithm for answering to these questions:

1- Can we predict the hours of absence for each employee?

2- Can we classify these absences as short-term or long-term?

To answer these questions we consider performing two separate tasks that will be explained with more detail in following:

1- Hours of being absent prediction that is a supervised problem by multiple linear regression and regression tree models

2- Absence classification to short-term and long-term by logistic regression and clas-

sification tree

When we use logistic regression in this problem, we want to specify the probability of an employee be in the range of employees with most absenteeism hours. Also, with fitting Regression tree model we aims to fit a model to predict absenteeism hours.

2 Data

The dataset that we will use for this project to train and test models is available on UCI from 2018 [1]. The data is related to a company in Brazil that collected their employees information for three years. The dataset has 740 observations and 20 features and 1 label. We can see the more detail about this dataset in tabel1.

Table 1: Dataset summary- numeric features

	Distance	Service_Time	Age	HitTarget	Hours(Label)	Expense
Min	5	1	27	81	0	118
1st Qu	16	9	31	93	2	179
Median	26	13	37	95	3	225
Mean	29	12	36	94	6.9	221.3
3rd Qu	50	16	40	97	8	260
Max	52	29	58	100	120	388

	Children	Pet	Weight	Height	Body_mass_index
Min	0	0	56	163	19
1st Qu	0	0	69	169	24
Median	1	0	83	170	25
Mean	1.02	0.74	79	172	26.68
3rd Qu	2	1	89	172	31
Max	4	8	108	196	38

Table 2: Dataset summary- factor features

	Reason	Month	Day	Season	Disciplinary	Education	Drinker	Smoker
Range	1-28	0-12	2-6	1-4	0-1	1-4	0-1	0-1

The reason of absence was shown in 28 categories, 21 of which are based on international code of diseases (ICD) and 6 remained reasons are in miscellaneous categories. You can see definition of these reasons in Table3:

Table 3: Reasons of absence

Reason code	Description
1	Certain infectious and parasitic diseases
2	Neoplasms
3	Diseases of the blood
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
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
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
22	Patient follow-up
23	Medical consultation
24	Blood donation
25	laboratory examination
26	Unjustified absence
27	Physiotherapy
28	Dental consultation

The meaning of numbers for these factor feature are:

Days of week: Monday (2), Tuesday (3), Wednesday (4), Thursday (5), Friday (6)

Seasons:summer (1), autumn (2), winter (3), spring (4)

Education:high school (1), graduate (2), postgraduate (3), master and doctor (4)

2.1 Pre-processing

First of all since the name of features is long also have some sign we change the column names. We can see the distribution of response column in figure1. We add a new binary target variable as absence class. For this class we need to have a cut off point, so we use median of hour column for this mean, the median here is 3 and we can see that 45.81% of observations were absent more than 3 hours so the data is balance to fit the classification

models.

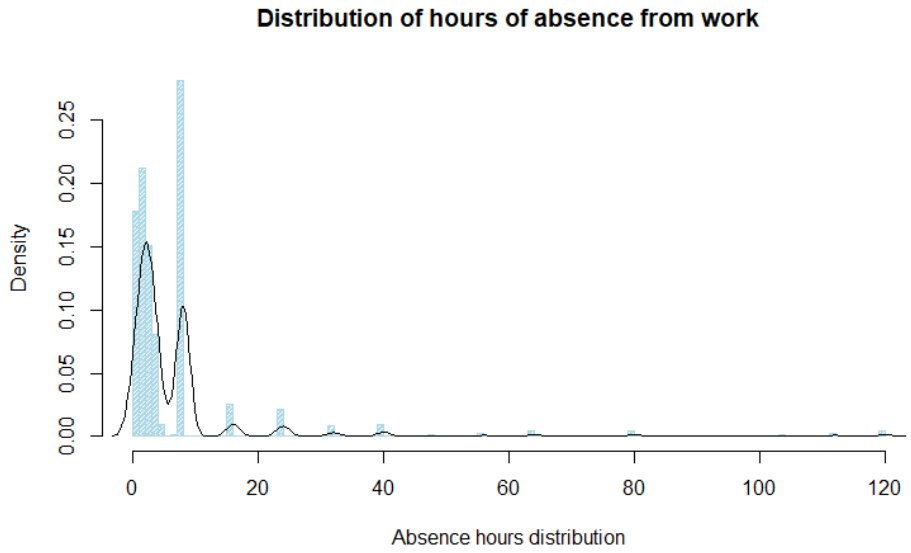

Figure 1: Response distribution

In next step we search for missing value in our dataset and none of observations does not have any empty or 'NA' value.

There are 3 observations that the month value is zero, since this value is not valid in our definition we need to adjust it. we can remove or just replace it with the most repeated value in months that is 3 with frequency 87. We choose to replace it because the number of observation is limit.

In reason column we see 22 observations have value of 0 that in the reason definition range it is not valid, so we do the same imputation that we did for month and replace reason 23 instead of 0.

We changed the type of features that are factors but it shows as numbers. Also since there are different scales for numeric features we normalize the numeric columns.

Another feature preprocessing that has applied in this dataset is creating dummy variables for features that are factor. so the number of features have been increased to 64.

In order to finding outlier firstly we look at summary table of dataset to see in which features, min or max is far of mean or median. We can see in Table1 that expense and pet needs to be checked, so we see their boxplots in following figures.

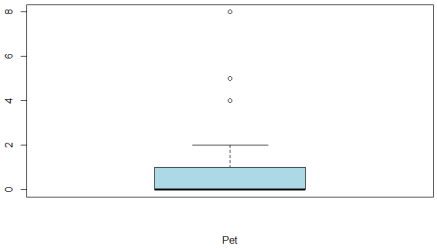

Figure 2: number of pet

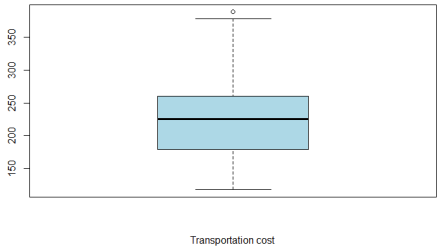

Figure 3: Transportation cost

Based on boxplots we impute the observations that are out of range with the mean of that features.

2.2 Data Exploration

We calculate the correlation between features that the result can be seen in figure4 and figure5. There are collinearity between weight and body_mass around 0.9. Also the correlation between weight and height, service time and body_mass, service time and age is huge so in feature selection part we will remove these features.

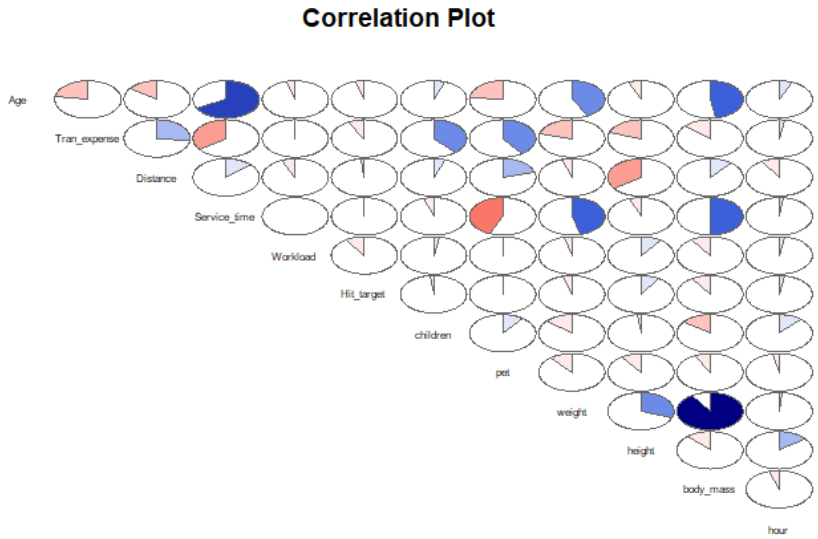

Figure 4: Correlation Plot

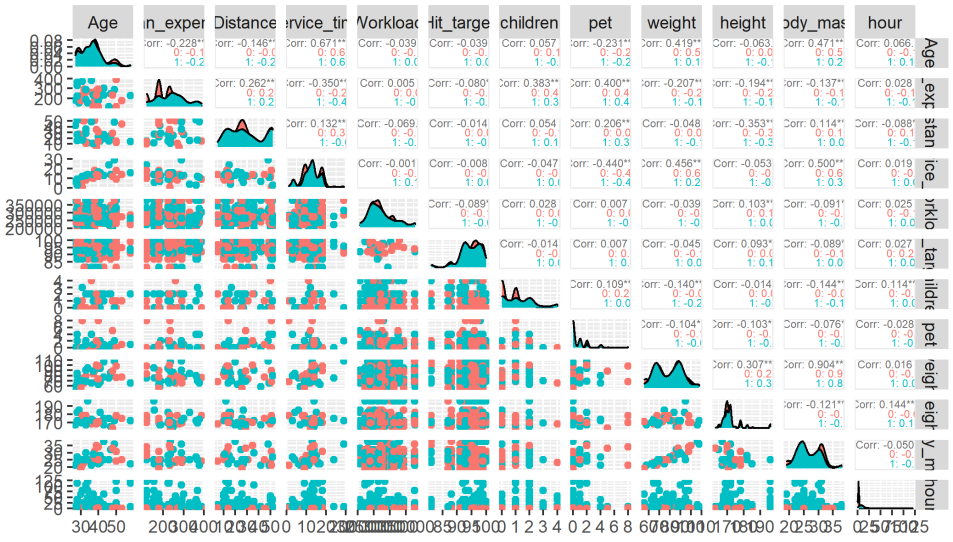

Figure 5: Correlation plot and Absence label

In figure6 and figure7 we check the distribution of reasons in this dataset, we can see that reason 23 has the most repeat and reason 13 has has the most observations for long-term absence that is related to musculoskeletal disorders so need more focus in this area.

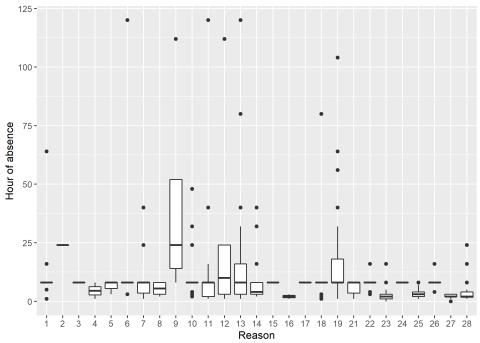

Figure 6: hour of absence

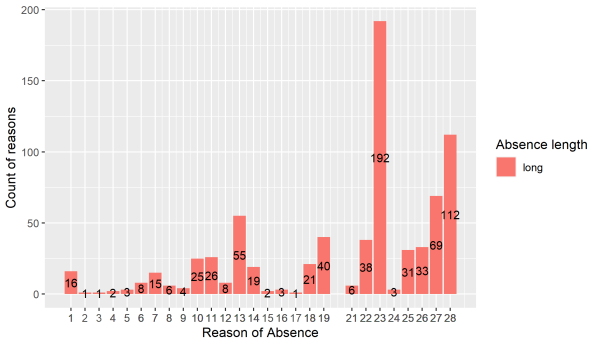

Figure 7: Reason count in dataset

In figure8 we can see that employees with 1 or 2 children tend to have more hour of absence on Mondays. In figure9 there are some pick on lines of Mondays and Tuesday, also in months of March, July and November there are more absence hours.

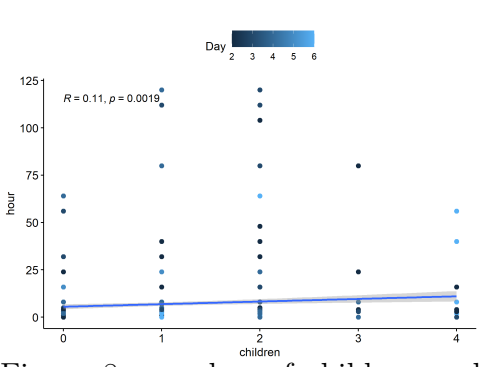

Figure 8: number of children and hour of absence

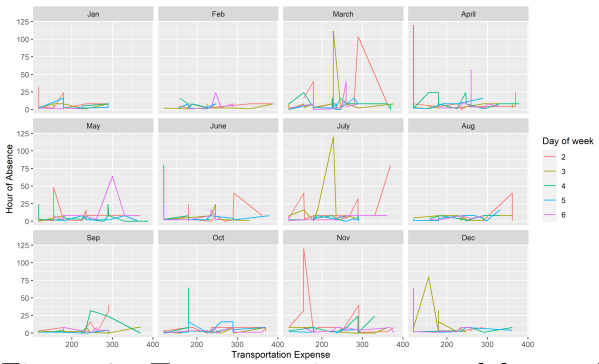

Figure 9: Transportation cost and hour of absence

With ploting the distributions of features we can see which value of features have more absence hours in this dataset. For example in figure8, employees with age between 35 - 40 had more absence, beside this employees who touch the target of their work more than 90% have more absence.

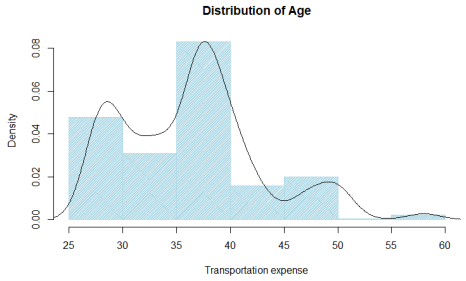

Figure 10

Figure 11

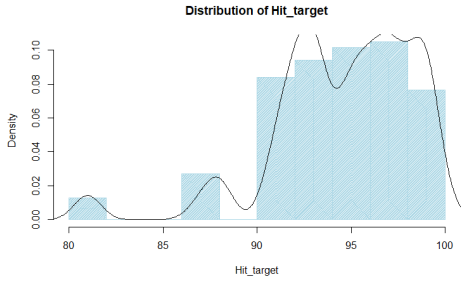

Figure 12

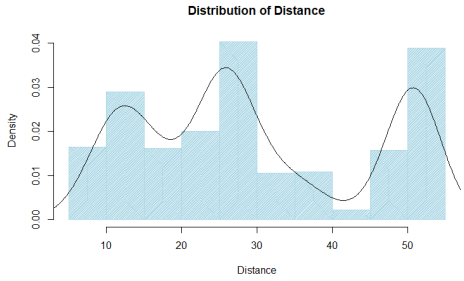

Figure 13

3 Machine learning Methods

In this part we will build machine learning models for final dataset. We will use 80% of data for training the models and 20% for test. Also, in following we applied K-Fold cross validation to estimate the test error.

3.1 Feature selection

There are many ways to filter the features that can perform and fit better models, in part 3 we used some statistical tools such as correlation for removing features with high colinearity, and in this part we will use best subset selection and PCA.

3.1.1 Best Subset Selection

In result of best subset feature selection in figure14, we can see that model with 10 feature has the best adjusted R^2, with these features: reason,age,day, disciplinary_failure, education, children, body_mass drinker and smoker. So we will fit both with all features and just these selected ones and compare the results.

3.1.2 PCA

PCA is an unsupervised algorithm to extract new features. In figure15 we can see that 10 new PCA explain more than 80% of data. We will rebuild the models with new principal components to compare the results.

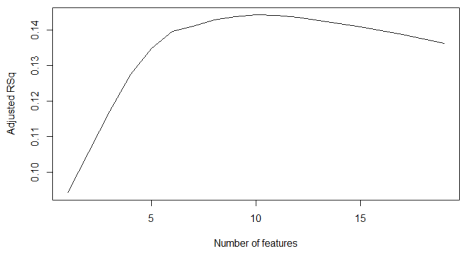

Figure 14: Best subset selection

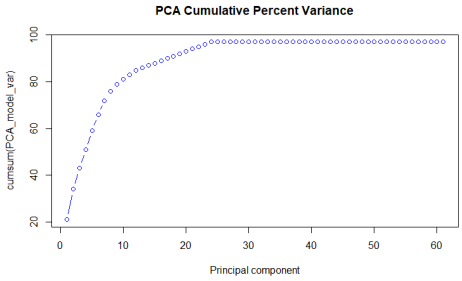

Figure 15: PCA

3.2 Classification

When we have a supervised problem with a factor label, can perform classification algorithms in our dataset. In our dataset we add new binary label as long and short absence hour. The aim of this part is putting the employees in these two classes based on features. Here we select two algorithm, logistic regression and classification tree. the reason that we select these algorithms are the interpretability of these models are good and both work well in having not much observations. Models will be compared by accuracy of predicting the correct labels in test dataset.

3.2.1 Logistic Regression

This parametric algorithm can perform in a binary response dataset and gives probability of being in a class by maximum likelihood. When we build the model with all features, accuracy of test data is 84.45%. When we use of just 10 features instead of 19 features the accuracy would be 83.78%. The cut-off point in this model is selected 0.4 that has the most accuracy.

3.2.2 Classification Tree

When we fit the model with classification tree we see that the accuracy is 83.1% in test data and when we rebuild the tree for just selected features the accuracy will be improved to 85.13%. It shows tree models will work better when the number of features is limited and with the less correlation. In this part we use cross validation to find the best cut point of tree. and we can see that in figure17 that the best cut point is 8, so we prune the tree with 8 terminal nodes and we see that the accuracy will not change.

Figure 16: classification Tree model

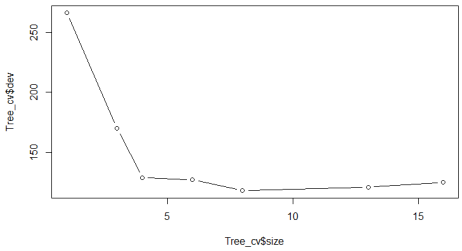

Figure 17: classification Tree CV

3.3 Regression

In regression part we are going to predict the hour of absence. Here we will use two algorithms, multiple linear regression and regression tree. The comparison parameters in this part would be root mean square error(RMSE) and MAE.

3.3.1 Multiple Linear Regression

In first step we split the data to train and set with respect to response which is a continuous valuable that shows the hour of absence. Then we build the regression model with all features that $adjustedR^2 = .30$, $RMSE = 17.29$ and $MAE = 6.7$. In next step rebuild it just with selected features, so in result $adjustedR^2 = .34$, $RMSE = 11.11$ and

$MAE = 5.22$. It is obvious the prediction model works better in limited features that has most effect in the predicting labels.

3.3.2 Regression Tree

When we build regression tree with selected features we see that $RMSE = 7.48$ and $MAE = 4.6$. When we prune the tree to 8 terminal nodes, we can see that the model parameters will be improved,$RMSE = 7.3$ and $MAE = 3.9$

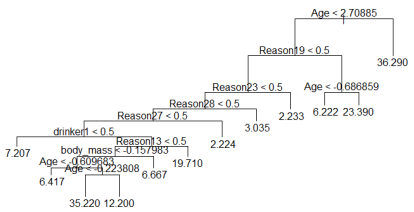

Figure 18: Regression Tree model

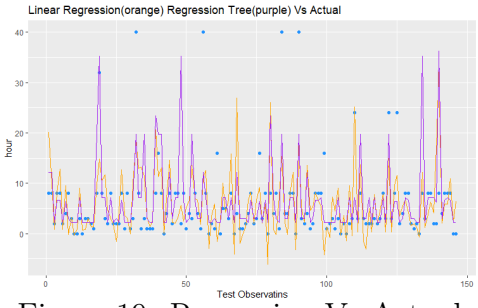

Figure 19: Regression Vs Actual

4 Method Evaluation

In classification part our models have been fitted and worked well, the accuracy that have been calculated by confusion matrix, for classification tree is the most between others. Since cv in tree model can specify the important features and cut the tree in the best way. In table5 we can see the result of regression the hour of absence and based on calculated errors the multiple linear regression fits better.

Table 4: Classification models comparison- Test Accuracy

Model	with all predictors	with best subset selection
Logistic Regression	84.45%	82.43%
Tree	83.10%	85.13%
Random Forest		81.08%
LDA		83.78%

Table 5: Regression models comparison

Model	RMSE	MAE
Multiple linear Regression	7.84	5.01
Tree	10.08	4.99
Random Forest	9.35	5.10

5 Discussion and limitations

After fitting the models we can see that the prediction of hours doesn't work well and it needs more focus on feature selection or even feature extraction by PCA, or using sparsity models. But in classification part the accuracy is fair enough to see the model works good. We can see than in feature selection below features have been selected as important ones

Table 6: Significant features

feature
Reason
Day
Disciplinary_failure
Age
smoker
drinker
children
Education
weight

6 References

[1] Absenteeism at work. [Online].Available:https://www.kaggle.com/racholsan/customer-data, 2018. URL.

[2] Artificial neural network and their application in the prediction of absenteeism at work. *Journal of International Business Research and Marketing*, 2018.

[3] Reuben Mokwena Badubi. A critical risk analysis of absenteeism in the work place. *Journal of International Business Research and Marketing*, 2(6):32–36, 2017.

[4] Kelley A. G. Mitchell K. M. Ruggieri M. P. Kocakulah, M. C. Absenteeism problems and costs: Causes, effects and cures. international business economics. *International Business Economics Research Journal (IBER)*, 15(3):89–96, 2016.

[5] K.J. Mullen and S Rennane. Worker absenteeism and employment outcomes: A literature review. *National Bureau of Economic Research*, 2017.

[6] Süß S. Ruhle, S.A. Presenteeism and absenteeism at work—an analysis of archetypes of sickness attendance cultures. *J Bus Psychol*, 35:241–255, 2020.

Supplementry- Prediction of Absenteeism Project-CSE780

Paria Fakhrzad Student ID 400353290

10/24/2021

Contents

Load The libraries

```
library(dplyr)
library(Hmisc)
library(magrittr)
library(readr)
library(ggplot2)
library(ISLR2)
library(class)
library(ggpubr)
library(corrplot)
library(GGally)
library(PreProcess)
library(caTools)
library(caret)
library(GGally)       #ggpairs()
library(PreProcess)
library(tree)         #tree/CART
library(MASS)
library(mclust)       #Gaussian Mixtures
library(car)
library(boot)         #CV
library(e1071)
library(leaps)
library(glmnet)
library(pls)
library(gridExtra)
library(mgcv)         #GAM
library(randomForest) #Random Forest
library(corrgram)     #corrgram
library(ROCR)
library(pROC)
library(ROCit)
library(plotROC)
```

1- Uploading the dataset

The main dataset has been downloaded from UCI repository in this link https://archive.ics.uci.edu/ml/datasets/Absenteeism+at+work# that is the data of Absenteeism at work for years between 2007-2010. * Read CSV file

```r
#Reading CSV files
Absence_df<- readr::read_csv("Absenteeism_at_work  UCI.csv")

#Summary of Dataset
selected <- c("Distance_from_Residence_to_Work","Service_time", "Age",
              "Work_load_Average/day_","Hit_target","Weight","Height",
              "Absenteeism_time_in_hours","Transportation_expense")

xtable::xtable(summary(Absence_df[,selected]))
xtable::xtable(summary(Absence_df[,c("Son","Pet","Body_mass_index")]))


# number of rows and columns
dim(Absence_df)


#Structure of dataset
str(Absence_df)
```

2- Pre-Processing

- We change the name of columns to be more simple for visualization, also for using in model fitting

```r
#Change the Colnames
colnames(Absence_df) <- c('ID', 'Reason', 'Month', 'Day',
'Seasons','Tran_expense', 'Distance', 'Service_time', 'Age','Workload'
, 'Hit_target', 'Disciplinary_failure', 'Education', 'children', 'drinker'
, 'smoker', 'pet', 'weight', 'height', 'body_mass', 'hour')
```

- We can add new label for classification purpose:

```r
median(Absence_df$hour) # median will be separate point that is 3
Absence_df <- mutate(Absence_df, Absence=ifelse(Absence_df$hour>3,1,0))

# Percent of Absence "True" Label
sum(Absence_df$Absence)/nrow(Absence_df)*100
```

46% of observations have absence hour more that median(3)

- With using below function we found that there is no NA dataset.

```r
#Columns that are totally empty
table(sapply(Absence_df,function(x)all(is.na(x))))

#Columns with NA
table(lapply(Absence_df,function(x){length(which(is.na(x)))}))

#Another way for checking missing values
as.data.frame(colSums(is.na(Absence_df)))
```

- Finding the outlines
- 3 sample rows have month number 0 that they needs to be adjusted

```r
#Adjusting the non-valid month values
table(Absence_df$Month)
Absence_df$Month[Absence_df$Month %in% 0] =3

#Change null reasons with the mode
table(Absence_df$Reason)
Absence_df$Reason[Absence_df$Reason %in% 0] =23
```

- checking the outliers by boxplot

```r
boxplot(Absence_df$Tran_expense,col = "lightBlue",xlab="Transportation cost")
```

```r
boxplot(Absence_df$pet, col="lightBlue",xlab="Pet")
```

- Imputation of outliers

```r
# Transportation expense
Absence_df$Tran_expense[Absence_df$Tran_expense>370] <- 360

# Pet
Absence_df$pet[Absence_df$pet>4] <- 4
```

3- Data Exploration

In this part We are going to check the multicollinearity between numeric features:

```
#Correlation
rcorr(as.matrix(Absence_df))

#separate numeric columns
numeric = sapply(Absence_df, is.numeric)
Absence_numeric= Absence_df[,numeric]

plot2 <-ggpairs(Absence_df[,c("Age","Tran_expense","Distance","Service_time",
         "Workload","Hit_target","children","pet","weight","height","body_mass","hour")],
         ggplot2::aes(colour=as.factor(Absence_df$Absence)),
         upper = list(continuous = wrap("cor",size = 2, alignPercent = 1)))
plot2

plot1<- corrgram(Absence_numeric[,c("Age","Tran_expense","Distance","Service_time",
               "Workload","Hit_target","children","pet","weight","height",
               "body_mass","hour")], order = F,
               upper.panel=panel.pie, lower.panel = panel.number(),
         text.panel=panel.txt, main = "Correlation Plot")

ggsave("figure2.png",plot2,width = 7, height = 4)
```

We can see between body_mass and height and weight there are correlation so we will remove this.

- In this part we used Chi-square test to see the correlation between factor columns

```
chisq.test(Absence_df$Reason,Absence_df$Month)
chisq.test(Absence_df$Reason,Absence_df$Disciplinary_failure)
chisq.test(Absence_df$Reason,Absence_df$Education)
chisq.test(Absence_df$Reason,Absence_df$drinker)
chisq.test(Absence_df$Reason,Absence_df$smoker)
```

all of P-values are less than 0.05.

- Distribution of factor features

```
#Reason Distribution
reason<-as.data.frame(xtabs(~Reason,Absence_df))
plot(reason)
```

```
#Reason of absence box_plot
plot3 <- ggplot(Absence_df,aes_string(x=as.factor(Absence_df$Reason),
               y=Absence_df$hour))+geom_boxplot()+xlab('Reason')+
  ylab('Hour of absence')

#change the factor for having clear plot
Absence_df$Absence<-ifelse(Absence_df$Absence==0,"short", "long")

plot4 <- ggplot(Absence_df,aes(x=Reason,fill=factor(Absence)))+
  geom_bar(stat="count")+
  stat_count(geom = "text", colour = "black", size = 3.5,
aes(label = ..count..),position=position_stack(vjust=0.5))+
  labs( x = "Reason of Absence", y = "Count of reasons", fill = "Absence length")+
  scale_x_continuous(labels=Absence_df$Reason, breaks=Absence_df$Reason)

ggsave("figure3.png",plot3,width = 7, height = 5)
ggsave("figure4.png",plot4,width = 7, height = 4)

#Showing Plot beside eachother
#gridExtra::grid.arrange(graph1, graph2,ncol=2)
```

- Distribution of numerical data by histogram

```
#Distribution of hour for checking normality
plot5<-ggplot(Absence_df, aes(x=hour))+geom_bar()+
  ggtitle("Distribution of absence hour ")+theme_classic()

plot5
```

```
plot6<-hist(Absence_df$hour,breaks=100 ,density=100,prob=TRUE, col="lightblue",
     xlab="Absence hours distribution",
     main="Distribution of hours of absence from work")
lines(density(Absence_df$hour))
```

```
plot6
```

In This plot , we can see that IDs that touched the target tends to be absence more.

```
plot7<-ggplot(Absence_df, aes(x=ID, y= hour ))+
         geom_bar(stat='identity')+facet_grid(.~Hit_target)
plot7
```

- Ploting histogram of continouse variables

```
#Transportation expense
hist(Absence_df$Tran_expense,prob = TRUE,breaks=10 ,density=100,col="lightblue",xlab = '
lines(density(Absence_df$Tran_expense))

#Age
hist(Absence_df$Age,prob = TRUE,breaks=10 ,density=100,col="lightblue",xlab = 'Transport
lines(density(Absence_df$Age))

#Hit target
hist(Absence_df$Hit_target,prob = TRUE,breaks=10 ,density=100,col="lightblue",xlab = 'Hi
lines(density(Absence_df$Hit_target))

#Distance
hist(Absence_df$Distance,prob = TRUE,breaks=10 ,density=100,col="lightblue",xlab = 'Dist
lines(density(Absence_df$Distance))
```

```
# children and absence hours
plot9 <-ggscatter(Absence_df,x ='children',y ='hour', color  = 'Day',
          add = "reg.line", conf.int = TRUE,
          cor.coef = TRUE, cor.method = "pearson")
plot9

ggsave("figure5.png",plot9, width=7, height=5)
```

```
# Transportation expense and hours
plot10 <-ggplot(Absence_df,
                aes(x =Tran_expense, y=hour, colour= factor(Day)))+
  geom_line()+
  facet_wrap(.~Month,
             labeller = as_labeller( c('1'="Jan",'2'="Feb",
                                       '3'="March","4"="April",
                                       "5"="May","6"="June","7"="July",
                                       "8"="Aug","9"="Sep","10"="Oct",
                                       "11"="Nov","12"="Dec")))+
  labs( x = "Transportation Expense",
        y = "Hour of Absence",
        color = "Day of week")+
  scale_color_manual(labels=c("Monday","Tuesday","Wednesday","Thursday","Friday"),
                     values=c("red","blue","darkorange","darkGreen","purple"))

plot10
ggsave("figure6.png",plot10, width=10, height=6)
```

4- Classification Models

- Feature Scaling

```
#Change target variable to factor
Absence_df$Absence<- ifelse(Absence_df$Absence=="short",0, 1)
```

- Normalize dataset

```
#remove ID
Absence_df<-Absence_df[,-1]
Absence_Scaled_df <-Absence_df
Absence_Scaled_df[c(5,6,7,8,9,10,13,16,17,18,19)] <-
  scale(Absence_Scaled_df[c(5,6,7,8,9,10,13,16,17,18,19)])
```

- Here we make dummy features from reason column

```
#transform factor variables
Absence_Scaled_df$Reason <- as.factor(as.character(Absence_Scaled_df$Reason))
Absence_Scaled_df$Month <- as.factor(as.character(Absence_Scaled_df$Month))
Absence_Scaled_df$Day <- as.factor(as.character(Absence_Scaled_df$Day))
Absence_Scaled_df$Seasons <- as.factor(as.character(Absence_Scaled_df$Seasons))
Absence_Scaled_df$Disciplinary_failure <-
  as.factor(as.character(Absence_Scaled_df$Disciplinary_failure))
Absence_Scaled_df$Education <- as.factor(as.character(Absence_Scaled_df$Education))
Absence_Scaled_df$drinker <- as.factor(as.character(Absence_Scaled_df$drinker))
Absence_Scaled_df$smoker <- as.factor(as.character(Absence_Scaled_df$smoker))

#Dummy features
Absence_df_Dummy <- as.data.frame(model.matrix(~.,Absence_Scaled_df))
Absence_df_Dummy <-Absence_df_Dummy[,-1]
```

- Split Data

```
set.seed(1)

#Split of Scaled data
 ts_split <- createDataPartition(Absence_df_Dummy$Absence, p = 0.8, list = FALSE)
 train_c <- Absence_df_Dummy[ts_split,]
 test_c <- Absence_df_Dummy[-ts_split,]

 #Split of main data
```

```
ts_split <- createDataPartition(Absence_df$Absence, p = 0.8, list = FALSE)
 train_c2 <- Absence_df[ts_split,]
 test_c2 <- Absence_df[-ts_split,]

 #check the split accuracy
table(train_c$Absence)/sum(table(train_c$Absence))
table(test_c$Absence)/sum(table(test_c$Absence))
table(train_c2$Absence)/sum(table(train_c2$Absence))
table(test_c2$Absence)/sum(table(test_c2$Absence))
```

- Here we just consider 10 features that we selected from part6 that is related to Feature selection

```
Absence_selected_df <-Absence_Scaled_df[,c(1,3,8,11,12,13,14,15,19,20,21)]
#transform factor variables
Absence_selected_df$Reason <- as.factor(as.character(Absence_selected_df$Reason))
Absence_selected_df$Disciplinary_failure <-
  as.factor(as.character(Absence_selected_df$Disciplinary_failure))
Absence_selected_df$Education <- as.factor(as.character(Absence_selected_df$Education))
Absence_selected_df$smoker <- as.factor(as.character(Absence_selected_df$smoker))

#Dummy features
Absence_selected_Dummy <- as.data.frame(model.matrix(~.,Absence_selected_df))
Absence_selected_Dummy <-Absence_selected_Dummy[,-1]

#Split data
set.seed(1)
ts_split_selected <- createDataPartition(Absence_selected_Dummy$Absence,
                                         p = 0.8, list = FALSE)
 train_selected_c <- Absence_selected_Dummy[ts_split_selected,]
 test_selected_c <- Absence_selected_Dummy[-ts_split_selected,]
 train_selected_c <-train_selected_c[,-40]
 test_selected_c <-test_selected_c[,-40]
#check the split accuracy
table(train_selected_c$Absence)/sum(table(train_selected_c$Absence))
table(test_selected_c$Absence)/sum(table(test_selected_c$Absence))
```

4.1- Logistic Regression

```
set.seed(1)
#Logistic regression
```

```
LR_model<- glm(Absence~ .-hour-Reason3,
               data =  train_c,
               family = binomial("logit"))

LR_predict <-predict(LR_model, newdata = test_c, type = "response")

predicted_data <- data.frame(probability2=LR_predict, probability=LR_predict,
                             Absence=test_c$Absence,Absence2=test_selected_c$Absence)

predicted_data$probability=ifelse(predicted_data$probability>.4,1,0)

#Accuracy
mean(predicted_data$probability==predicted_data$Absence)

#Interpret the output
summary(LR_model)
```

Accuracy for test dataset is 84.45%.

rebuild the model with selected features

```
#Logistic regression with selected features
LR_model2<- glm(Absence~ .-Reason3-Reason2,
               data =  train_selected_c,
               family = binomial("logit"))
LR_predict2<-predict(LR_model2, newdata = test_selected_c, type = "response")

predicted_data$probability2 <- LR_predict2
predicted_data$probability2=ifelse(predicted_data$probability2>.60,1,0)

#Accuracy
mean(predicted_data$probability2==predicted_data$Absence2)

#Interpret the output
summary(LR_model2)
```

- Cross Validation
  - K-fold Logistic Regression

```
set.seed(0)
# fit with whole dataset
LR_model_cv <- glm(Absence~ .-hour-Reason3,
               data =  Absence_df_Dummy,
```

```
                family =binomial("logit"))
# MSE per K
set.seed(0)
cv.glm(Absence_df_Dummy, LR_model_cv, K=5)$delta[1]
LR_CV_table <- data.frame(matrix(ncol = 2, nrow= 0))
colnames(LR_CV_table) <- c('k', 'MSE')
for(i in seq(from = 3, to = 20, by = 1)){
 LR_CV_table[i,'k'] <- i
 LR_CV_table[i,'MSE'] <- cv.glm(Absence_df_Dummy, LR_model_cv, K=i)$delta[1]
}
LR_CV_table
plotcv<- ggplot(LR_CV_table[3:20,],aes(x=k, y=MSE))+geom_line()
ggsave("plotcv.png",plotcv, width=5, height=3)
plotcv
```

We can see that the first drop is in k=7 and deviation here is 142

4.2- Classification Tree

```
train_c$Absence<-as.factor(train_c$Absence)
test_c$Absence<-as.factor(test_c$Absence)

# Fit
Tree_model <- tree(Absence~.-hour,data=train_c)

#Predict
Tree_predict <- predict(Tree_model, test_c, type = "class")

#Interpret the output
plot(Tree_model)
text(Tree_model, pretty = 1)
```

```
#Accuracy
mean(Tree_predict==test_c$Absence)
```

Accuracy of train data is 84%. and in test data is 83.1%.Residual mean deviance is 0.73.

- rebuild using of just selected features

```
train_selected_c$Absence<-as.factor(train_selected_c$Absence)
test_selected_c$Absence<-as.factor(test_selected_c$Absence)

# Fit
Tree_model2 <- tree(Absence~.,data=train_selected_c)

#Predict
Tree_predict2 <- predict(Tree_model2, test_selected_c, type = "class")

#Interpret the outptut
plot(Tree_model2)
text(Tree_model2, pretty = 1)
```

```
#Accuracy
mean(Tree_predict2==test_selected_c$Absence)
```

- Cross Validation here we use CV to find the optimal nodes for this tree

```
#Pruning
Tree_cv <- cv.tree(Tree_model,FUN = prune.misclass)
Tree_cv
plot(Tree_cv$size, Tree_cv$dev, type = "b")
```

tree with 8 terminal nodes have minimum cross-validation errors that is 118 and is less than Logistic regression CV So we cut the tree with 9 nodes and we can see the result:

```
prune.Tree <- prune.misclass(Tree_model2, best = 8)

plot(prune.Tree)
text(prune.Tree, pretty = 0)
```

```
Tree_predict3 <- predict(prune.Tree, test_selected_c, type = "class")

#Accuracy
mean(Tree_predict3==test_selected_c$Absence)
```

we can see that with less number of nodes that is more interpretable, we have the same test accuracy rate that is 85.13%.

4.3- Random forest

```
bag_model <- randomForest(Absence~.,data=train_selected_c,
                          mtry=13,importance=TRUE,type="class")
bag_model
predict_bag <- predict(bag_model,test_selected_c,type="class")

#Accuracy
mean(predict_bag==test_selected_c$Absence)
```

Train accuracy is 81% and test accuracy is 74% and both are less than decision tree.

4.4- LDA

```
#fit
lda_model <- lda(Absence ~.-Reason3,data=train_selected_c)

#predict
lda_predict <- predict(lda_model,test_selected_c)

#Accuracy
mean(lda_predict$class==test_selected_c$Absence)
```

The accuracy of LDA is 83.7% that is less than classification tree

5- Regression Models

- Split Data

```
#Split of Scaled data
 ts_split_R <- createDataPartition(Absence_df_Dummy$hour, p = 0.8, list = FALSE)
 train_R <- Absence_df_Dummy[ts_split_R,-63]
 test_R <- Absence_df_Dummy[-ts_split_R,-63]


ts_split_R_selected <- createDataPartition(Absence_selected_Dummy$hour,
                                           p = 0.8, list = FALSE)
 train_selected_R <- Absence_selected_Dummy[ts_split_R_selected,-41]
 test_selected_R <- Absence_selected_Dummy[-ts_split_R_selected,-41]
```

5.1- Multiple Linear Regression

```
#fit
lm_model <- lm(hour ~.-Reason3-Reason2-Reason17, data = train_R)

#predict
lm_predict <- predict(lm_model, newdata = test_R)

#Linear Regression MSE
#mean((lm_predict - test_R$hour)^2)
library("forecast")
print(postResample(pred = lm_predict, obs = test_R$hour))
forecast::accuracy(test_R$hour,lm_predict)

#Linear Regression Rsq
summary(lm_model)$r.sq
#summary(lm_model)
#vif(lm_model)
```

$R^2 = .30$ and, $RMSE = 17.73$, $MAE = 6.75$

rebuild linear regression based on selected features

```
#fit
lm_model2 <- lm(hour ~.-Reason2-Reason3, data = train_selected_R)

#predict
lm_predict2 <- predict(lm_model2, newdata = test_selected_R)

#Linear Regression MSE
#mean((lm_predict - test_R$hour)^2)
print(postResample(pred = lm_predict2, obs = test_selected_R$hour))

#Linear Regression Rsq
summary(lm_model2)$r.sq
summary(lm_model2)
#vif(lm_model2)
```

$R^2 = .33$ and, $RMSE = 15.24$, $MAE = 6.79$

```
#Residuals Vs fitted values
plot(predict(lm_model), residuals(lm_model))
```

5.2- Regression Tree

```
#Fit the tree model
Tree_model_r <- tree(hour ~., data = train_selected_R)

#Predict
Tree_predict_r <-predict(Tree_model_r,newdata = test_selected_R)

#Interpret the outptut
plot(Tree_model_r)
text(Tree_model_r, pretty = 0)
```

```
summary(Tree_model_r)

#MSE
print(postResample(pred = Tree_predict_r, obs = test_selected_R$hour))

#prune tree
prune.Tree2 <- prune.tree(Tree_model_r, best = 8)

plot(prune.Tree2)
text(prune.Tree2, pretty = 0)
```

```
Tree_predict_r <- predict(prune.Tree2, test_selected_R)
#MSE
print(postResample(pred = Tree_predict_r, obs = test_selected_R$hour))
```

number of terminal nodes is 16 and Residual mean deviance is 103.1 here. MSE of regression tree is 253 that is more than linear regression and in plot we can see the predicted amount has less accuracy for test dataset. Also the tree is complex for interpret.

```
test_selected_R$hour[test_selected_R$hour>50] <- 40
data<- tibble(hour=test_selected_R$hour,
              predicted_hour1 = lm_predict2,
              predicted_hour2 = Tree_predict_r )
data$rank<-1:nrow(data)

Regression<-ggplot(data) +
  geom_point(aes(x = rank, y = hour), color = "dodgerblue") +
  geom_line(aes(x = rank, y = predicted_hour1), color = "orange")+
  geom_line(aes(x = rank, y = predicted_hour2), color = "purple")+
  xlab("Test Observatins")+
```

```
  ggtitle("Linear Regression(orange) Regression Tree(purple) Vs Actual ")

Regression
```

5.3- Random forest

```
# fit the model
bag_model_r <- randomForest(hour~.,data=train_selected_R,
                            mtry=13,importance=TRUE)
bag_model_r
predict_bag_r <- predict(bag_model_r,test_selected_R)

#calculating RMSE, Rsw and MAE
print(postResample(pred = predict_bag_r, obs = test_selected_R$hour))
```

```
plot(predict_bag_r,test_selected_R$hour)
abline(0,1)
```

6- Feature Selection

6.1- PCA

```
#build PCA model
PCA_model <- prcomp(Absence_df_Dummy[,-c(62,63)], scale=FALSE)

#Calculate the variance per PCA
PCA_model_var <- round(((PCA_model$sdev^2)/sum(PCA_model$sdev^2))*100,0)

#plot PCA
plot10 <- (plot(cumsum(PCA_model_var), main="PCA Cumulative Percent Variance",
                xlab="Principal component",type="b", col="blue"))
```

6.3- Best subset selection

```
best_FS <- regsubsets(hour ~.-Absence, Absence_df, nvmax = 20)
summary(best_FS)
```

```
#Plot
plot(summary(best_FS)$rss, xlab = "Number of features", ylab = "RSS", type = "l")
```

```
plot(summary(best_FS)$adjr2, xlab = "Number of features",
     ylab = "Adjusted RSq", type = "l")
```

```
which.max(summary(best_FS)$adjr2)

#see the selected features
summary(best_FS)$which[which.max(summary(best_FS)$adjr2), ]
```

we can see tha reason,Age,Day,childern, Disciplinary_failure,Education,children

6.3- Forward subset selection

```
Fwd_FS<- regsubsets(hour ~.-Absence,
                    Absence_df,
                    nvmax = 20,
                    method = "forward")

summary(Fwd_FS)

plot(summary(Fwd_FS)$rss, xlab = "Number of features",
ylab = "RSS", type = "l")
```

```
plot(summary(Fwd_FS)$adjr2, xlab = "Number of features",
ylab = "Adjusted RSq", type = "l")
```

```
#Summary
which.max(summary(Fwd_FS)$adjr2)
coef(Fwd_FS, 15)
coef(best_FS, 15)
```

Canty and Ripley (2021) Breiman et al. (2018) Ripley (2021)

7- Model with PCA

7.1- Regression

```
Model_PCA <- pcr(hour~. ,data=train_selected_R, validation="CV")
Predict_Model_PCA <- predict(Model_PCA, test_selected_R[,-40],ncomp=15)
print(postResample(pred = Predict_Model_PCA, obs = test_selected_R$hour))
```


Breiman, Leo, Adele Cutler, Andy Liaw, and Matthew Wiener. 2018. *randomForest: Breiman and Cutler's Random Forests for Classification and Regression*. https://www.stat.berkeley.edu/~breiman/RandomForests/.

Canty, Angelo, and Brian Ripley. 2021. *Boot: Bootstrap Functions (Originally by Angelo Canty for s)*. https://CRAN.R-project.org/package=boot.

Ripley, Brian. 2021. *Tree: Classification and Regression Trees*. https://CRAN.R-project.org/package=tree.