Predicting Absenteeism Time in Hours.

Nishkarsh Bansal

Contents

# Introduction

* 1. Problem Statement. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3
  2. Data . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3

# Methodology

* 1. Pre Processing. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .8
     1. Missing Value Analysis. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9
     2. Outlier Analysis. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10
     3. Feature Selection . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13
     4. Feature Scaling. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 16
  2. Model Development. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .17  
     2.2.1 Model Selection. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .18  
     2.2.2 Decesion Tree. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 18  
     2.2.3 Random Forest. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .19  
     2.2.4 Multiple Linear Regression. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20

# Conclusion

3.1 Model Evaluation. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. . . . . . . . . . . .21  
 3.1.1 Mean Absolute Error (MAE). . . . . . . . . . . . . . . . . . . . . . . . . . . .. . . . . 22  
 3.1.2 Mean Squared Error (MSE). . . . . . . . . . . . . .. . . . . . . . . .. . . . . . . . . . . 22  
3.2 Model Selection. . . . . . . . . . . . . . . . . .. . . . . .. . . . . . . . . .. . . . . . . . .. . . . . . .22  
3.3 Solution. . . . . . . . . . .. . . . . . . . . . .. . . .. . . . . . . . . . . . . .. . . . . .. . . . . . . . . . 23  
 3.3.1 Problem 1. . . . . . . . . . . . .. . . . . . . . . . . . .. . .. . . . .. . . . .. . . . . . . . . . . 23  
 3.3.2 Problem 2 . . . . . .. . . . . . . . . . . . . . . .. . . . . . . .. . . . . . .. . . .. . . . . … . . . 24

Appendix A- Complete R code

Appendix B-Complete Python Code

Chapter 1

Introduction

1.1 Problem Statement

XYZ is a courier company .As we appreciate that human capital plays an important role in collection , transporatation and delievery. The company is passing through genuine issue of Absenteeism. So company wants to know the impact of Employee absenteeism have . So they want the answer   
What changes should company bring to reduce the number of absenteeism , also they want to know how much loss company project every month in 2011 if same trand continues.

1.2 Data

Our task is to built the regression model to predict the absenteeism time in hours which will depend on multiple given factors . Data set consisit of 740 observations and 21 variables and ‘Absenteeism time in hours ‘is our is our target variable.Below is the list of our given variables.  
  
  
Table 1.1 Absenteeism time sample data (columns 1-10)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Id | Reason for absence | Month of absence | Day of the week | Seasons | Transportation expense | Distance from Residence to Work | Service time | Age | Work load Average/day |
| 11 | 26 | 7 | 3 | 1 | 289 | 36 | 13 | 33 | 239,554 |
| 36 | 0 | 7 | 3 | 1 | 118 | 13 | 18 | 50 | 239,554 |
| 3 | 23 | 7 | 4 | 1 | 179 | 51 | 18 | 38 | 239,554 |
| 7 | 7 | 7 | 5 | 1 | 279 | 5 | 15 | 39 | 239,554 |

Table 1.2 Absenteeism time sample data (columns 11-21)

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Hit target | Disciplinary failure | Education | Son | Social drinker | Social smoker | Pet | Weight | Height | Body mass index | Absenteeism time in hours |
| 97 | 0 | 1 | 1 | 1 | 1 | 1 | 90 | 172 | 30 | 4 |
| 97 | 1 | 1 | 2 | 1 | 1 | 0 | 89 | 178 | 31 | 0 |
| 97 | 0 | 1 | 0 | 1 | 1 | 0 | 68 | 170 | 31 | 2 |
| 97 | 1 | 1 | 2 | 1 | 0 | 0 | 80 | 168 | 24 | 4 |

As you can see we have 21 variables out of which we have Absenteeism time as our target variable.Here is the list of our predictor variables.  
  
  
Table 1.3 List of predictor variables

|  |  |
| --- | --- |
| S.No | Variables |
| 1 | ID |
| 2 | Reason for absence |
| 3 | Month of absence |
| 4 | Day of the week |
| 5 | Seasons |
| 6 | Transportation Expense |
| 7 | Distance from Residence to Work |
| 8 | Service Time |
| 9 | Age |
| 10 | Work load Average/Day |
| 11 | Hit target |
| 12 | Disciplinary Failure |
| 13 | Education |
| 14 | Son |
| 15 | Social drinker |
| 16 | Social smoker |
| 17 | Pet |
| 18 | Weight |
| 19 | Height |
| 20 | Body mass Index |

Below we are attaching the codes for various variables.  
In Reason for absence   
Absences attached by the International Code of Disease(ICD) stratified into

I Certain stratified and parasitic diseases.  
II Neoplasms  
III Diseases of the blood.  
IV Endocrine , nutritional and metabolic diseases.  
V Mental and behavioral diseases .  
VI Disease of the nervous system.  
VII Disease of the eye and adnexa.  
VIII Disease of the ear and mastoid process.  
IX Disease of the circulatory system.  
X Disease of the respiratory system.  
XI Disease of the digestive system.  
XII Disease of the skin and subcontaneous tissue.  
XIII Disease of the muscoskeletal and connective tissue.  
XIV Disease of the genitournary system.  
XV Pregnancy  
XVI Certain condition originating in the perinatal period  
XVII Congestional malfornation.  
XIX Injury , poisoning and certain causes of external causes.  
XX External causes of morbidity and mortality.  
XXI Factors influencing health status and contact with health services.  
  
AND seven categories without (CID) patients follow- up are

XXII Medical Consultation.  
XXIII Blood Donation.  
XXIV Laboratory examination.  
XXV Unjustified absence.  
XXVI Pysiotheraphy.  
XXVII Dental consultation.

# 

# 

2)Month of Absence

3)Day of the week (Monday-2,Tuesday-3,Wednesday-4,Thrusday-5,Friday-6.

4)Seasons (Summer-1, Autumn-2,winter-3, Spring-4)

5)Transportation Expense.

6)Distance from Residence to work

7)Service time

8)Age

9)Work Load Average/Day

10)Hit target

11)Disciplinary Failure(yes-1 ,no-0)

12)Education(high school-1, graduate-2,postgraduate-3,masterand doctor-4)

13)Son (Number of children)

14)Social Drinker (yes-1,no-0)

15)Social Smoker(yes-1,no-0)

16)Pet (number of pets)

17)Weight

18)Height

19)Body mass index

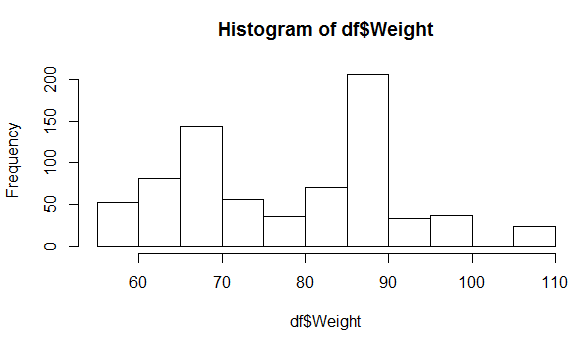
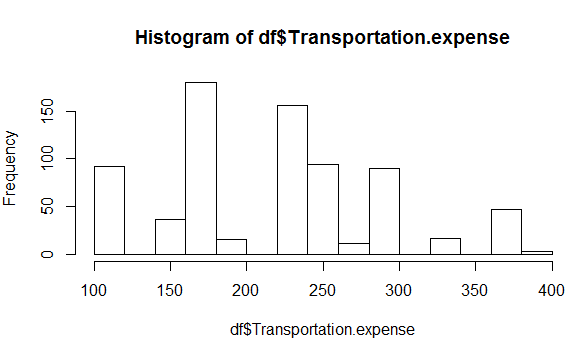
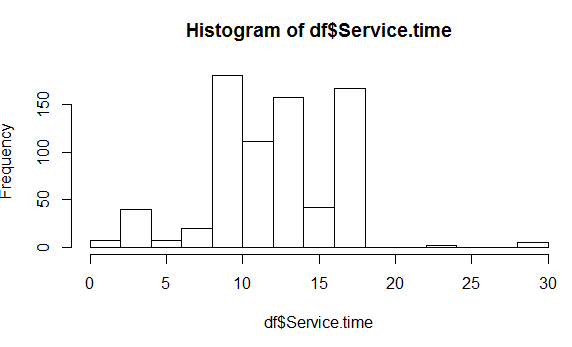
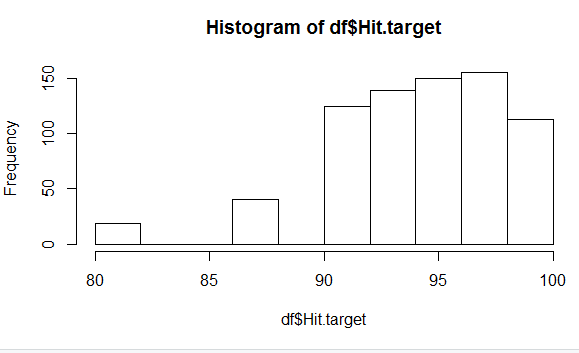
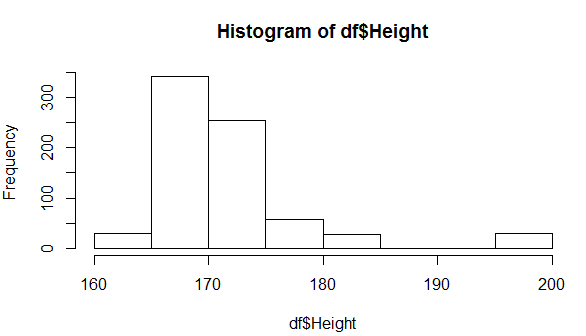
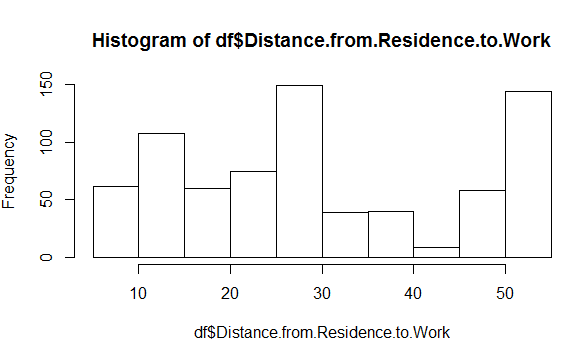
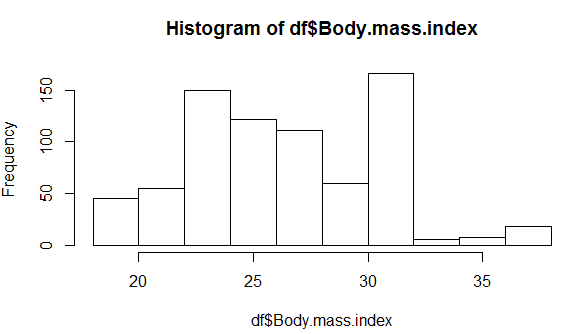
20)Absenteeism time in hours.

# 

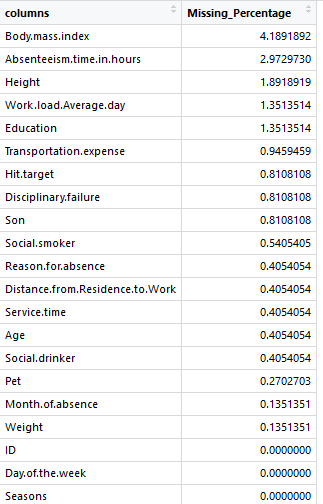
Chapter 2

Methodology

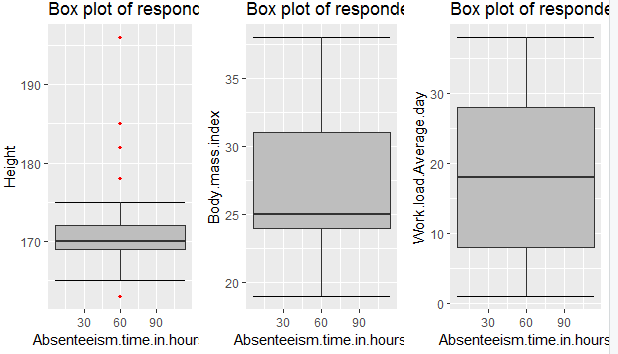
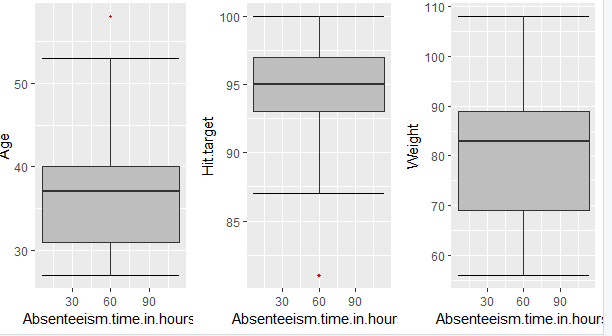
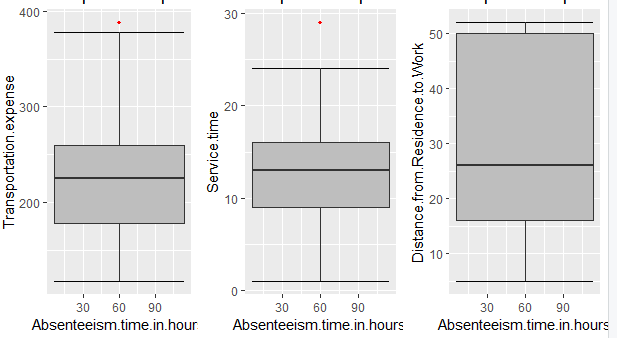
2.1 Pre-Processing

Any predictive model ling requires that we look at the data before processing.However, in data minning it is not just to look data it is more than that.Looking the data refers to exploring the data, cleaning the data and exploring the data cleaning the data and visualising the data with graphs and plots.This is called **Exploratory** **Data Analysis**.In exporatory data analysis we need to find missing values in a data set then impute the missing values after that use outlier analysis to remove outliers from the data set .Then come the feature selection and after that feature scaling .Below are the histogram plots.  
  
 

2.1.1 Missing Value Analysis

Missing Value occurs due to human error or the value was not applicable.So if the missing value percentage is more than 30% then we neglect that variable. In data set absenteeism time in hours missing values are present , here is the table of total missing values present in the dataset 

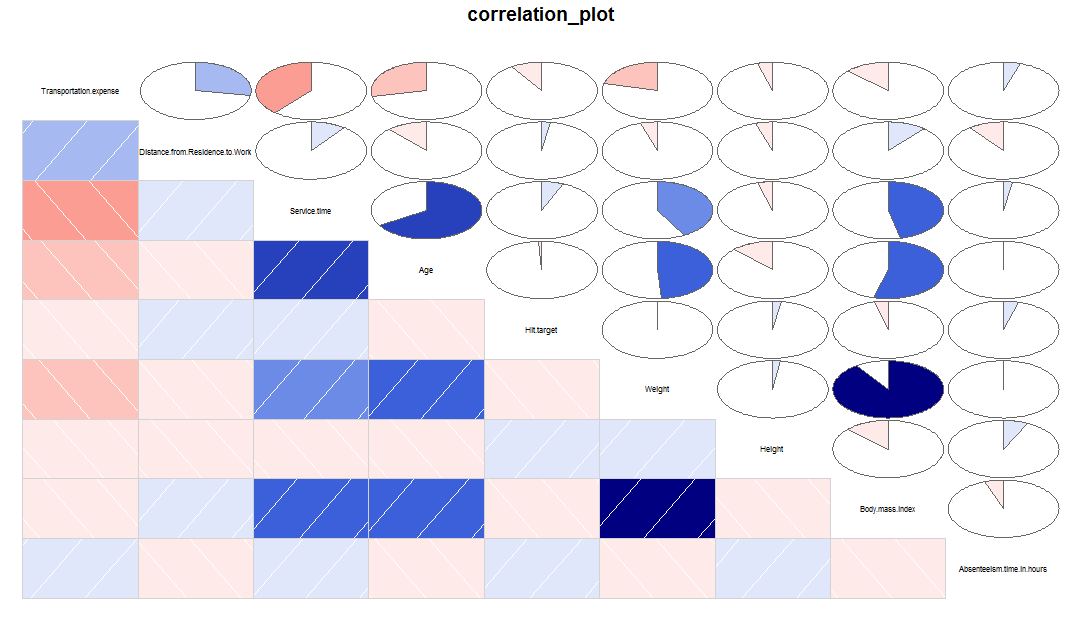
2.1.2 Outlier Analysis

We can clearly observed from the plots that every variable is skewed . The skewnesss in the distribution is explained by the presence of outliers and extreme values in the dataset.We can clearly see the presence of outliers and its effect in the skewness. Below we have plot the boxplot for various variables present in the data set

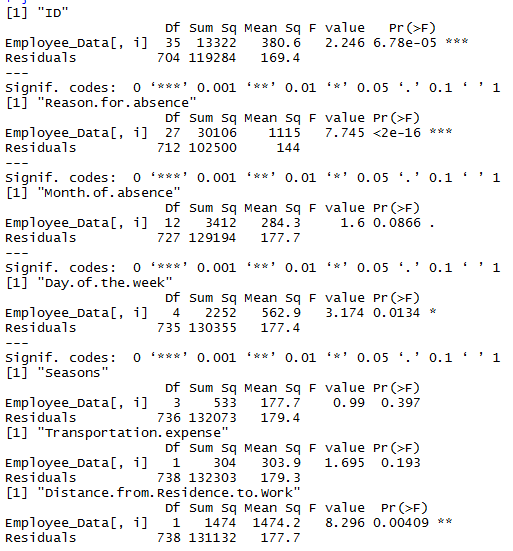
|  |  |
| --- | --- |
| **Variables** | **Number of Outliers** |
| Transportation expense | 3 |
| Distance from Residence to work | 0 |
| Service Time | 5 |
| Age | 8 |
| Hit target | 19 |
| Weight | 0 |
| Height | 119 |
| Body mass index | 0 |
| Work Load Average/Day | 0 |

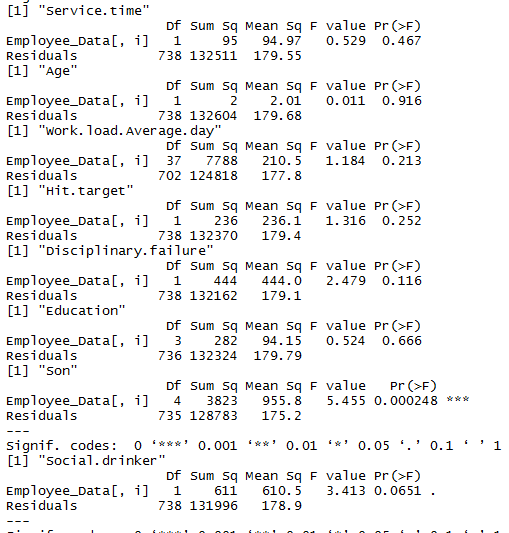
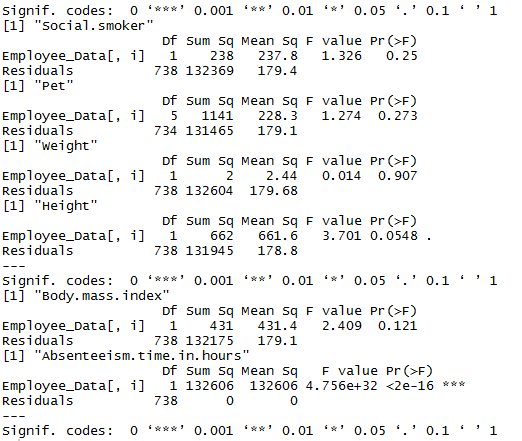
Therefore we have analysed these number of outliers in the given variables.Now we need to remove these outliers and fill these outliers with KNN .

2.1.3 Feature Selection.

In feature selection we need to eliminate those variables which do not play any imporatant role in the target variable means they are independent on target variable.  
Firstly on numerical variables we need to perform Correlation analysis , variables which are highly correlated need to eliminate any one of them as they play same role in the prediction of target variable.Below is the correlation plot.  
  


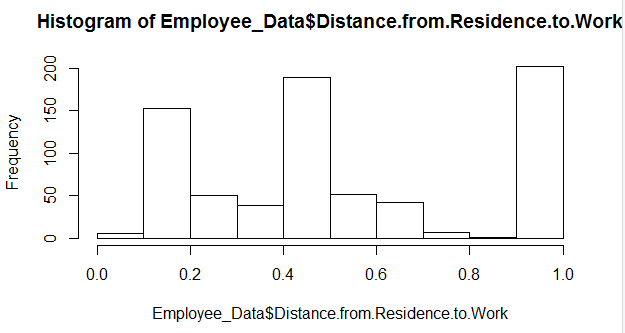
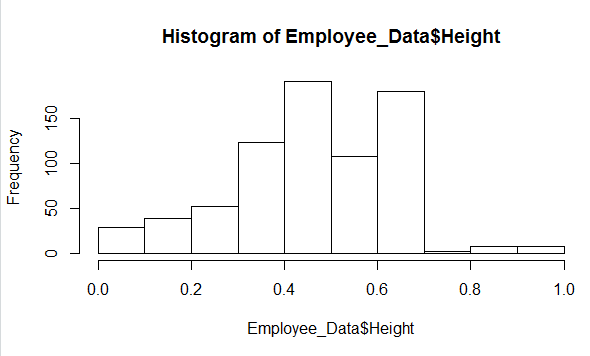
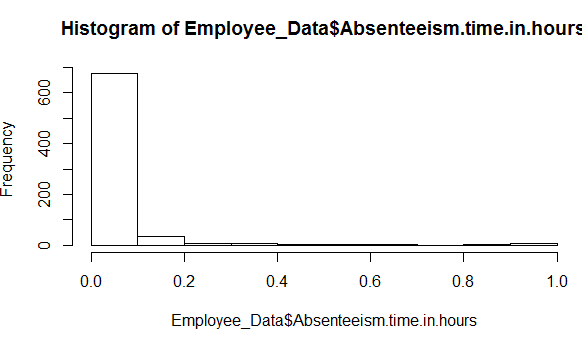
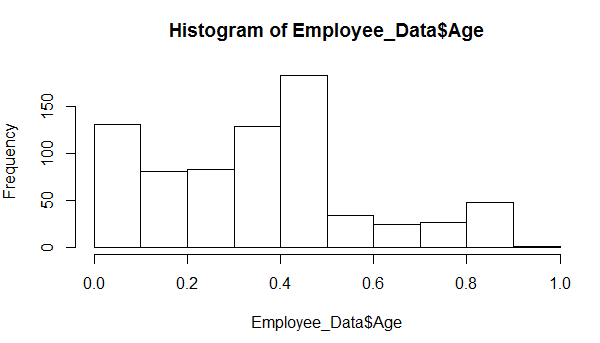
From the correlation plot we can clearly see that weight and Body mass index are highly correlated so we need to eliminate any one of the variables as both have the same impact in the prediction of target variable.So we eliminate Body mass index from the main data set.Also with ANOVA test we can find the actual dependence of each variable with target variable.  
Following are the results of ANOVA test:



  
  
We can analyse from the importance of variable from the p-value result of the ANOVA.Smaller the value of p higher the importance of variable.The threshold value of p is 0.05 i.e. if p value of variable is less than 0.05 , we select the variable for model development.By this analysis we drop the variables like weight,education, service time,Work load Average/day,Seasons,Social smoker,pet,Hit target,Transportation expense,Body mass index,disciplinary failure,Month of absence,Social drinker as all this pvalue is greater than 0.05.

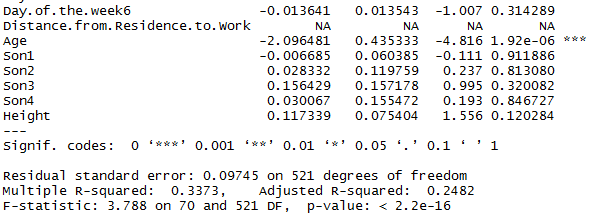
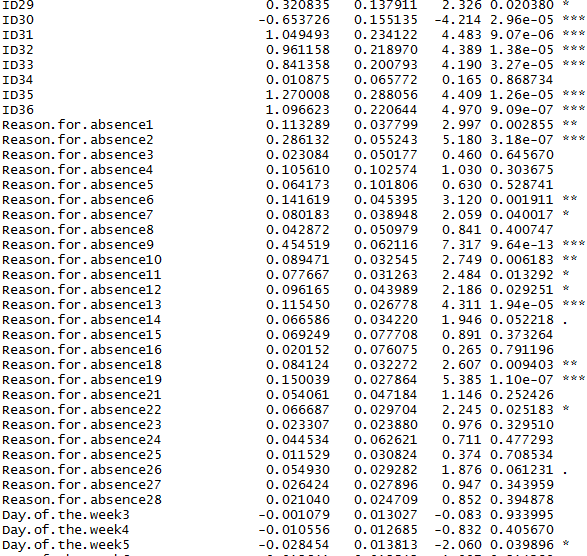
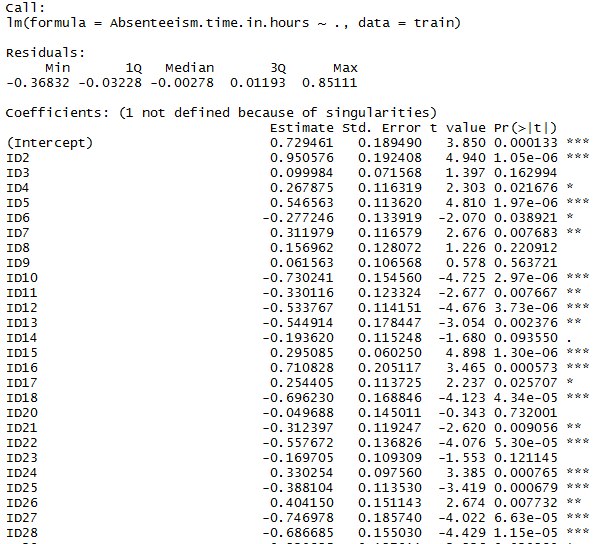
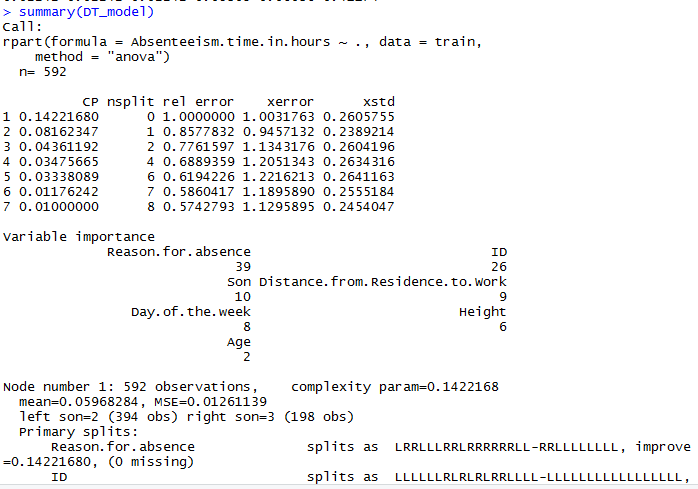
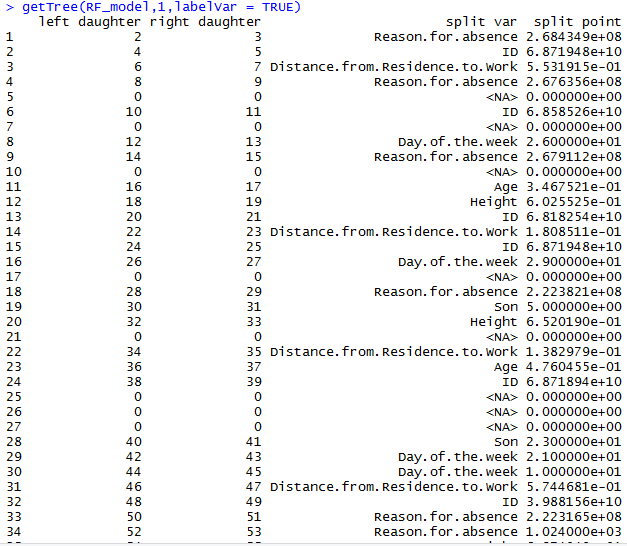
2.1.4 Feature Scaling.

In feature scaling we scale our features so that it does not have difference in between the variables.This can be done such that by two process Normalization and Standardization  
Normalization is done when data is skewed and Standardization is done when data is normally distributed.  
Normalization = x(i)-min(x(i))/max(x(i))-min(x(i))  
Standarization = x(i)-mean(x(i)/SD((x(i))

2.2 Model Development.

2.2.1 Model Selection :After preprocessing of data we must proceed with model development.For employee absenteeism project, we want to find what changes should company bring to reduce absenteeism problem and also what is the loss in year 2011 per month, if same trend follow.So we need to find the importance of each variable with respect to target variable to suggest the changes for the company and also predict the result of year 2011 loss due to absenteeism.For this we can choose following regression model :

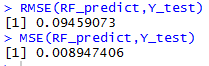
I) Multiple Linear Regression.  
 II)Decesion Tree Regression.  
 III) Random Forest Regression.  
   
  
  
2.2.2 Multiple Linear Regression :  
  
As we see that R-squared value is 0.2482 or 24.82% or 25% which is not very impressive  
  
2.2.3 Decision Tree  
 As with implementation of decision tree for regression, we get the importance of each variable for predicting target variables  
   
  
2.2.4 Random Forest  
 Glimpse of first tree of random forest tree  
  
  
  
  
CHAPTER 3  
  
CONCLUSION  
  
3.1 Model Evaluation  
3.1.1 Accuracy : After model development it is important to check accuracy.For time series data it is important to check accuracy we use Mean Square Error(MSE) and Root Mean Square Error(RMSE).

Linear Regression accuracy.PNG

1. Multiple Linear Regression.

Accuracy Decision Tree.PNG

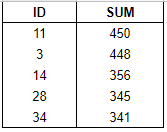
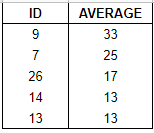
1. Decision Tree

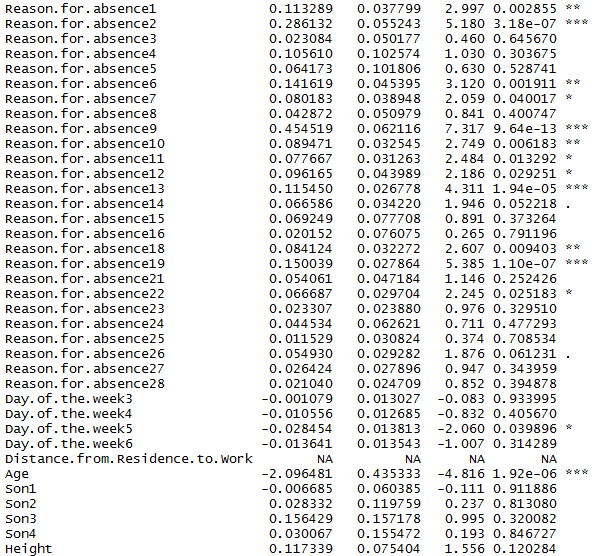


(c)Random Forest

3.2 Model Selection :

As it can be clearly seen from the MSE or RMSE result , Random Forest Regression model is performing best for Employee Absenteeism dataset.So we select Random Forest model for the prediction of the model.  
  
  
  
3.3 Solution  
3.3.1 Problem 1  
What changes company should bring to reduce the number of absenteeism?

1.Company should take disciplinary actions against employees with ID3, ID11 and ID9(As per figure)  
 

2. As per results of multiple linear regression , some of the reasons for high absenteeism were due to health condition like  


So from above test main reasons were:

* Disease of the nervous system
* Disease of the eye and adnexa.
* Disease of the circulatory.
* Disease of digestive system
* Disease of the respiratory system
* Disease of the skin and subcontaneous tissue.
* Disease of the muscoskeletol system and connective tissue.
* Disease of the genitournary system.
* Injury , poisoning and certain other consequences of external causes.

So company should take preventive measures to prevent these diseases and also check the claim made by the employee to reduce the absenteeism issue.

3.3.2 Problem 2

How much loss company project in 2011 if the same trend of absenteeism continues?

For calculating the loss company incured in 2011 we need to consider some parameters as no information about loss is given .So for doing that we are considering some parameters as nothing is mentioned about loss in the problem statement.So,I have assumed following things for calculating the loss:  
i)Company incurred the average loss of 500 rupees per hour for employee absenteeism.  
ii)If the employee is completed the target, then no loss is incurred i.e. if Hit Target =100, then loss=0.  
iii)The loss depends on the Disciplinary failure by factor of 1000.  
iv)The loss also depends on the age and education of employeesc i.e if employee with higher education and more Age(experience) will be more imporatant to the company and will have more impact on the company earning.

**Appendix A  
Complete R Code**  
  
rm(list=ls())

getwd()

x = c("ggplot2", "corrgram", "DMwR", "caret", "randomForest", "unbalanced", "C50", "dummies", "e1071", "Information",

"MASS", "rpart", "gbm", "ROSE", 'sampling', 'DataCombine', 'inTrees')

install.packages(x)

lapply(x, require, character.only = TRUE)

install.packages("rmarkdown")

Employee\_Data = read.csv("AAW.csv", header = T, na.strings = c(" ", "", "NA"))

str(Employee\_Data)

class(Employee\_Data)

catnames=c("ID","Reason.for.absence","Month.of.absence","Day.of.the.week","Seasons","Disciplinary.failure","Education","Son","Social.drinker","Social.smoker","Pet")

str(Employee\_Data)

#Exploratory Data Analysis

Employee\_Data$ID=as.factor(Employee\_Data$ID)

Employee\_Data$Reason.for.absence=as.factor(Employee\_Data$Reason.for.absence)

Employee\_Data$Month.of.absence=as.factor(Employee\_Data$Month.of.absence)

Employee\_Data$Day.of.the.week=as.factor(Employee\_Data$Day.of.the.week)

Employee\_Data$Seasons=as.factor(Employee\_Data$Seasons)

Employee\_Data$Disciplinary.failure=as.factor(Employee\_Data$Disciplinary.failure)

Employee\_Data$Education=as.factor(Employee\_Data$Education)

Employee\_Data$Son=as.factor(Employee\_Data$Son)

Employee\_Data$Social.drinker=as.factor(Employee\_Data$Social.drinker)

Employee\_Data$Social.smoker=as.factor(Employee\_Data$Social.smoker)

Employee\_Data$Pet=as.factor(Employee\_Data$Pet)

for (i in 1:nrow(Employee\_Data)) {

if(Employee\_Data$Absenteeism.time.in.hours[i]!=0 || is.na(Employee\_Data$Absenteeism.time.in.hours[i])){

if(Employee\_Data$Reason.for.absence[i]==0 || is.na(Employee\_Data$Reason.for.absence[i])){

Employee\_Data$Reason.for.absence[i]=NA

}

if(Employee\_Data$Month.of.absence[i]==0 || is.na(Employee\_Data$Month.of.absence[i])){

Employee\_Data$Month.of.absence[i]=NA

}

}

}

#Missing Value Analysis

#Impute values related to Id

Dependent\_ID=c("ID","Transportation.expense","Service.time","Age","Height","Distance.from.Residence.to.Work","Education","Son","Weight","Social.smoker","Social.drinker","Pet","Body.mass.index")

Dependent\_ID\_Data=Employee\_Data[,Dependent\_ID]

Dependent\_ID\_Data=aggregate(.~ID,data = Dependent\_ID\_Data,FUN=function(e) c(x=mean(e)))

for (i in Dependent\_ID) {

for (j in (1:nrow(Employee\_Data))) {

ID=Employee\_Data[j,"ID"]

if(is.na(Employee\_Data[j,i])){

Employee\_Data[j,i]=Dependent\_ID\_Data[ID,i]

}

}

}

sum(is.na(Employee\_Data))

missing\_index=sapply(Employee\_Data,is.na)

missing\_val=data.frame(apply(Employee\_Data,2,function(x){sum(is.na(x))}))

nrow(Employee\_Data)

#Impute values for other variables

Employee\_Data=knnImputation(Employee\_Data,k=7)

sum(is.na(Employee\_Data))

missingValueCheck(Employee\_Data)

#Outlier Analysis

class(Employee\_Data)

class(Employee\_Data$ID)

num\_index=sapply(Employee\_Data, is.numeric)

numeric\_data=Employee\_Data[,num\_index]

num\_cnames=colnames(numeric\_data)

for (i in num\_cnames) {

hist(Employee\_Data[,i],xlab = i,main = "\_",col = (c("lightblue","darkgreen"))

}

num\_cnames=num\_cnames[num\_cnames!="Absenteeism.time.in.hours"]

num\_cnames

for (i in 1:length(num\_cnames))

{

assign(paste0("gn",i), ggplot(aes\_string(y = (num\_cnames[i]), x = "Absenteeism.time.in.hours"), data = subset(Employee\_Data))+

stat\_boxplot(geom = "errorbar", width = 0.5) +

geom\_boxplot(outlier.colour="red", fill = "grey" ,outlier.shape=18,

outlier.size=1, notch=FALSE) +

theme(legend.position="bottom")+

labs(y=num\_cnames[i],x="Absenteeism.time.in.hours")+

ggtitle(paste("Box plot of responded for",num\_cnames[i])))

}

# ## Plotting plots together

gridExtra::grid.arrange(gn1,gn3,gn2,ncol=3)

gridExtra::grid.arrange(gn4,gn5,ncol=2)

gridExtra::grid.arrange(gn6,gn7,ncol=2)

df=Employee\_Data

Employee\_Data=df

for(i in num\_cnames){

print(i)

val = Employee\_Data[,i][Employee\_Data[,i] %in% boxplot.stats(Employee\_Data[,i])$out]

print(length(val))

}

#Replace all outliers with NA and impute

for(i in num\_cnames){

val = Employee\_Data[,i][Employee\_Data[,i] %in% boxplot.stats(Employee\_Data[,i])$out]

print(length(val))

Employee\_Data[,i][Employee\_Data[,i] %in% val] = NA

}

sum(is.na(Employee\_Data))

Employee\_Data = knnImputation(Employee\_Data, k = 7)

sum(is.na(Employee\_Data))

#Feature Selection

#Correlation Plot

corrgram(Employee\_Data,upper.panel = panel.pie,text.panel = panel.txt,main="correlation\_plot")

#As weight and body mass index are highly correlated so one variable need to be remove

#ANOVA test

cnames=colnames(Employee\_Data)

for (i in cnames) {

print(i)

print(summary(aov(Employee\_Data$Absenteeism.time.in.hours~Employee\_Data[,i],Employee\_Data)

}

Employee\_Data=subset(Employee\_Data,select=-c(Weight,Education ,Service.time ,Body.mass.index ,Seasons ,Transportation.expense ,Pet ,Disciplinary.failure ,Month.of.absence ,Hit.target ,Social.drinker ,Work.load.Average.day ,Social.smoker))

Employee\_Data

#Feature Scaling

num\_index=sapply(Employee\_Data, is.numeric)

numeric\_data=Employee\_Data[,num\_index]

num\_cnames=colnames(numeric\_data)

num\_cnames

for (i in num\_cnames) {

Employee\_Data[,i]=(Employee\_Data[,i]-min(Employee\_Data[,i]))/(max(Employee\_Data[,i])-min(Employee\_Data[,i]))

}

hist(Employee\_Data$Distance.from.Residence.to.Work)

hist(Employee\_Data$Age)

hist(Employee\_Data$Height)

hist(Employee\_Data$Absenteeism.time.in.hours)

#Model Development

set.seed(123)

X\_index=sample(1:nrow(Employee\_Data),0.8\*nrow(Employee\_Data))

X\_train=Employee\_Data[X\_index,-8]

X\_test=Employee\_Data[-X\_index,-8]

Y\_train=Employee\_Data[X\_index,8]

Y\_test=Employee\_Data[-X\_index,8]

train=Employee\_Data[X\_index,]

test=Employee\_Data[-X\_index,]

#Calculate RMSE

RMSE= function(y, yhat){

sqrt(mean((y - yhat)^2))

}

#Calculate MSE

MSE = function(y, yhat){

(mean((y - yhat)^2))

}

###########################Multiple Linear Regression###################

lm\_model = lm(Absenteeism.time.in.hours ~., data = train)

summary(lm\_model)

#Predict for new test cases

cat\_index=sapply(Employee\_Data, is.factor)

cat\_data=Employee\_Data[,cat\_index]

cat\_cnames=colnames(cat\_data)

cat\_cnames

for (i in cat\_cnames) {

lm\_model$xlevels[[i]]=union(lm\_model$xlevels[[i]],levels(X\_test[[i]]))

}

lm\_predict=predict(lm\_model,newdata = X\_test)

lm\_predict

RMSE(lm\_predict,Y\_test)

MSE(lm\_predict,Y\_test)

######################Decesion Tree Regression#####################

DT\_model=rpart(Absenteeism.time.in.hours ~.,data=train,method = "anova")

#Predict for new test cases

DT\_predict=predict(DT\_model,X\_test)

summary(DT\_model)

RMSE(DT\_predict,Y\_test)

MSE(DT\_predict,Y\_test)

#####################Random Forest###############

RF\_model=randomForest(x=X\_train,y=Y\_train,ntree = 100)

#Predict for new test cases

RF\_predict=predict(RF\_model,X\_test)

getTree(RF\_model,1,labelVar = TRUE)

summary(RF\_model)

RMSE(RF\_predict,Y\_test)

MSE(RF\_predict,Y\_test)

##########Problems#############

#Suggesting the changes

lm\_model\_pr=lm(Absenteeism.time.in.hours ~.,data=Employee\_Data)

summary(lm\_model\_pr)

#Calculating Losses

p2\_data=Employee\_Data[,-8]

#Predict for new test cases

p2\_predict=predict(RF\_model,p2\_data)

#convert predicted values to actual value

p2\_predict=(p2\_predict\*120)

#Add predicted values to the data set

p2\_dataset=merge(df[,-21],p2\_predict,by="row.names",all.x=TRUE)

#Calculate the total loss

loss=0

loss=0

for(i in 1:nrow(p2\_dataset)){

if (p2\_dataset$Hit.target[i]!=100)

if(p2\_dataset$Age[i]>=25 && p2\_dataset$Age<=32){

loss=loss+as.numeric(p2\_dataset$Disciplinary.failure[i])\*1000 +(as.numeric(p2\_dataset$Education[i])+1)\*500 + p2\_dataset$y[i]\*1000

}else if(p2\_dataset$Age[i]>=33 && p2\_dataset$Age[i]<=40){

loss=loss+as.numeric(p2\_dataset$Disciplinary.failure[i]\*1000) +(as.numeric(p2\_dataset$Education[i])+2)\*500 +p2\_dataset$y[i]\*1000

}else if(p2\_dataset$Age[i]>=41 && p2\_dataset$Age[i]<=49){

loss=loss+as.numeric(p2\_dataset$Disciplinary.failure[i]\*1000) +(as.numeric(p2\_dataset$Education[i])+3)\*500 +p2\_dataset$y[i]\*1000

}else if(p2\_dataset$Age[i]>=50 && p2\_dataset$Age[i]<=60){

loss=loss+as.numeric(p2\_dataset$Disciplinary.failure[i]\*1000) +(as.numeric(p2\_dataset$Education[i])+4)\*500 +p2\_dataset$y[i]\*1000

}

}

loss=loss/12

loss

**Appendix B  
Complete Python Code**import os

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from scipy.stats import chi2\_contingency

import seaborn as sns

employee\_data=pd.read\_csv("AAW.csv")

#Get column names

def get\_cnames(data):

all\_cnames=[]

num\_cnames=[]

cat\_cnames=[]

for i in data.columns:

all\_cnames.append(str(i))

if(data[i].dtype=="object"):

cat\_cnames.append(str(i))

else:

num\_cnames.append(str(i))

cnames=[all\_cnames,num\_cnames,cat\_cnames]

return(cnames)

#get cnames cnames[0]-all

#cnames[1]-numeric cnames

#cnames[2]-categorical cnames

cnames=get\_cnames(employee\_data)  
#get rows and columns

rows=employee\_data.shape[0]

columns=employee\_data.shape[1]  
  
#Exploratory Data Analysis

for i in cnames[0]:

print(str(i)+"\_"+str(type(employee\_data[i][2])))

#Changing Data as per requirement

for i in range(0,rows):

if employee\_data["Absenteeism time in hours"][i]!=0:

if employee\_data["Reason for absence"][i]==0:

employee\_data["Reason for absence"][i]=np.nan

if employee\_data["Month of absence"][i]==0:

employee\_data["Month of absence"][i]=np.nan

cat\_names=["ID","Reason for absence","Month of absence","Day of the week","Seasons","Education","Son","Social drinker","Social smoker","Pet"]

for i in cat\_names:

print(i)

employee\_data.loc[:,i]=employee\_data.loc[:,i].astype(str)

for i in cnames[0]:

print(str(i)+"\_"+str(type(employee\_data[i][1])))

cat\_names=["ID","Reason for absence","Month of absence","Day of the week","Seasons","Education","Son","Social drinker","Social smoker","Pet"]

for i in cat\_names:

employee\_data.loc[:,i]=employee\_data.loc[:,i].replace("nan",np.nan)

#Converting factor values to various labels

for i in range(0,len(employee\_data.columns)):

if(employee\_data.iloc[:,i].dtypes=='object'):

employee\_data.iloc[:,i]=pd.Categorical(employee\_data.iloc[:,i])

employee\_data.iloc[:,i]=employee\_data.iloc[:,i].cat.codes

employee\_data.iloc[:,i]=employee\_data.iloc[:,i].astype('object')

#Convert-1 values back to NAN

for i in cnames[0]:

for j in range(0,rows):

if employee\_data.loc[j,i]==-1:

employee\_data.loc[j,i]=np.nan

#Missing Value Analysis

def missing\_value\_check(data):

print(data.isna().sum())

missing\_value\_check(employee\_data)

#Impute values related to ID

dependent\_ID=["ID","Transportation expense","Service time","Age","Height","Distance from Residence to Work","Education","Son","Weight","Social drinker","Social smoker","Body mass index","Pet"]

dependent\_ID\_data=employee\_data[dependent\_ID].copy()

dependent\_ID\_data=dependent\_ID\_data.groupby("ID").max()

dependent\_ID.remove('ID')

for i in dependent\_ID:

for j in range(0,rows):

RI=employee\_data["ID"][j]

if np.isnan(employee\_data.loc[j,i]):

employee\_data[i][j]=dependent\_ID\_data.loc[RI,i]

employee\_data['Month of absence']=employee\_data['Month of absence'].fillna(employee\_data['Month of absence'].median())

employee\_data['Hit target']=employee\_data['Hit target'].fillna(employee\_data['Hit target'].median())

employee\_data['Disciplinary failure']=employee\_data['Disciplinary failure'].fillna(employee\_data['Disciplinary failure'].median())

employee\_data['Absenteeism time in hours']=employee\_data['Absenteeism time in hours'].fillna(employee\_data['Absenteeism time in hours'].median())

#Outlier analysis

onames = ["Transportation expense","Distance from Residence to Work","Age","Weight","Height","Body mass index","Service time"]

for i in onames:

q75,q25=np.percentile(employee\_data.loc[:,i],[75,25])

iqr=q75-q25

min=q25 - (1.5\*iqr)

max=q75 + (1.5\*iqr)

employee\_data.loc[employee\_data[i]<min,i]=np.nan

employee\_data.loc[employee\_data[i]>max,i]=np.nan

employee\_data['Reason for absence']=employee\_data['Reason for absence'].fillna(employee\_data['Reason for absence'].median())

employee\_data['Transportation expense']=employee\_data['Transportation expense'].fillna(employee\_data['Transportation expense'].median())

employee\_data['Service time']=employee\_data['Service time'].fillna(employee\_data['Service time'].median())

employee\_data['Age']=employee\_data['Age'].fillna(employee\_data['Age'].median())

employee\_data['Height']=employee\_data['Height'].fillna(employee\_data['Height'].median())

#Correlation Analysis

cor=["Transportation expense","Distance from Residence to Work","Service time","Age","Hit target","Weight","Height","Body mass index","Absenteeism time in hours"]

employee\_data\_corr=employee\_data.loc[:,cor]

f,ax=plt.subplots(figsize=(7,5))

corr=employee\_data\_corr.corr()

import seaborn as sns

sns.heatmap(corr,mask=np.zeros\_like(corr,dtype=np.bool),cmap=sns.diverging\_palette(220,50,as\_cmap=True),square=True,ax=ax)

#Anova Test

from sklearn.feature\_selection import SelectionKBest

from sklearn.feature\_selection import f\_regression

X=employee\_data.iloc[:,:-1].values

Y=employee\_data.iloc[:,20].values

#Create an SelectionKBest object to select features with two best ANOVA F values

selector=SelectKBest(f\_regression,k=7)

#Dimensionality Reduction:

employee\_data=employee\_data.drop(['Weight','Education','Service time','Body mass index','Seasons','Transportation expense','Pet','Disciplinary failure','Month of absence','Hit target','Social drinker','Work load Average/day ','Social smoker'],axis=1)

df=employee\_data.copy()

employee\_data=df.copy()

#Normality check

% matplotlib inline

plt.hist(employee\_data['Age'],lines='auto')

#Feature Scaling

c\_final=['Age','Height','Distance from Residence to Work','Absenteeism time in hours']

for i in c\_final:

print(i)

employee\_data[i]=(employee\_data[i]-np.min(employee\_data[i]))/(np.max(employee\_data[i])-np.min(employee\_data[i]))

#Error metrix

from sklearn.metrics import mean\_squared\_error

from math import sqrt

def RMSE(y,yhat):

print(sqrt(mean\_squared\_error(y,yhat)))

def MSE(y,yhat):

print(mean\_squared\_error(y,yhat))

X=employee\_data.iloc[:,:-1].values

Y=employee\_data.iloc[:,7].values

#Splitting the data in train and testing

from sklearn.model\_selection import train\_test\_split

X\_train,X\_test,Y\_train,Y\_test=train\_test\_split(X,Y,test\_size=0.2)

#Multiple Linear Regression:

from sklearn.linear\_model import LinearRegression

lm\_model=LinearRegression()

lm\_model.fit(X\_train,Y\_train)

lm\_predict=lm\_model.predict(X\_test)

RMSE(Y\_test,lm\_predict)

MSE(Y\_test,lm\_predict)

#Decision Tree Regressor

from sklearn.tree import DecisionTreeRegressor

DT\_model=DecisionTreeRegressor()

DT\_model.fit(X\_train,Y\_train)

DT\_predict=DT\_model.predict(X\_test)

RMSE(Y\_test,DT\_predict)

MSE(Y\_test,DT\_predict)

#Random Forest Regressor

from sklearn.ensemble import RandomForestRegressor

RF\_model=RandomForestRegressor()

RF\_model.fit(X\_train,Y\_train)

RF\_predict=RF\_model.predict(X\_test)

RMSE(Y\_test,RF\_predict)

MSE(Y\_test,RF\_predict)

#Problem

#Suggesting the changes

RF\_Regressor\_p1=RandomForestRegressor().fit(X,Y)

RF\_Regressor\_p1.feature\_importances\_

#Losses  
p2\_data=X

#Predicting for new test cases

p2\_predict=RF\_model.predict(p2\_data)

p2\_predict=pd.DataFrame(p2\_predict)  
  
p2\_frames=[employee\_data,p2\_predict]

p2\_dataset=pd.concat(p2\_frames,axis=1)

#Calculate the total Loss

Loss=0

for i in range (0,p2\_dataset.shape[0]):

if(p2\_dataset[‘Hit target’][i]!=100):

if(p2\_dataset[‘Age’][i]>=25 and p2\_dataset[‘Age’][i]<=32):

loss+=(p2\_dataset[‘Disciplinary failure’][i] +1)\*1000+  
 (p2\_dataset[‘Education’][i]+1+1)\*500 +  
 (p2\_dataset[‘predictions’][i]\*500

elif(p2\_dataset[‘Age’][i]>=33 and p2\_dataset[‘Age’][i]<=40):

loss+=(p2\_dataset[‘Disciplinary failure’][i] +1)\*1000+  
 (p2\_dataset[‘Education’][i]+1+2)\*500 +  
 (p2\_dataset[‘predictions’][i]\*500

elif(p2\_dataset[‘Age’][i]>=40 and p2\_dataset[‘Age’][i]<=49):

loss+=(p2\_dataset[‘Disciplinary failure’][i] +1)\*1000+  
 (p2\_dataset[‘Education’][i]+1+3)\*500 +  
 (p2\_dataset[‘predictions’][i]\*500

elif(p2\_dataset[‘Age’][i]>=50 and p2\_dataset[‘Age’][i]<=60):

loss+=(p2\_dataset[‘Disciplinary failure’][i] +1)\*1000+  
 (p2\_dataset[‘Education’][i]+1+4)\*500 +  
 (p2\_dataset[‘predictions’][i]\*500

#Loss per month

Loss=loss/12