A Project on Employee Absenteeism

By Sapna Krishnan

15 August 2019

Contents

1

Chapter 1

Introduction

The data analytics lifecycle defines analytics process best practises spanning discovery to project completion. The Data Analytics Life Cycle has 6 major phases. Our project will be directed through each phase to completion.

PHASE – 1 – DISCOVERY

Important activities in this phase include framing the business problem as an analytics challenge that can be addressed in subsequent phases and formulating initial hypothesis to test and begin learning the data.

1.1 Problem Statement

XYZ is a courier company. As we appreciate that human capital plays an important role in collection, transportation and delivery. The company is passing through genuine issue of Absenteeism. The company has shared it dataset and requested to have an answer on the following areas:

1. What changes company should bring to reduce the number of absenteeism?
2. How much losses every month can we project in 2011 if same trend of absenteeism continues?

1.2 Data

Our task is to build a suitable model which will help identify the ‘Trend of Absence’ based on various Absence reasons. This will help us analyse as to how to minimize the total Absence Hours. Given below is a sample of the data set:

	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
0	11	26.0	7.0	3	1	289.0	36.0	13.0	33.0	239554.0
1	36	0.0	7.0	3	1	118.0	13.0	18.0	50.0	239554.0
2	3	23.0	7.0	4	1	179.0	51.0	18.0	38.0	239554.0
3	7	7.0	7.0	5	1	279.0	5.0	14.0	39.0	239554.0
4	11	23.0	7.0	5	1	289.0	36.0	13.0	33.0	239554.0

Table 1.1: Employee Absenteeism Sample Data (Columns: 1-10)

Hit target	Disciplinary failure	Education	Son	Social drinker	Social smoker	Pet	Weight	Height	Body mass index	Absenteeism time in hours
97.0	0.0	1.0	2.0	1.0	0.0	1.0	90.0	172.0	30.0	4.0
97.0	1.0	1.0	1.0	1.0	0.0	0.0	98.0	178.0	31.0	0.0
97.0	0.0	1.0	0.0	1.0	0.0	0.0	89.0	170.0	31.0	2.0
97.0	0.0	1.0	2.0	1.0	1.0	0.0	68.0	168.0	24.0	4.0
97.0	0.0	1.0	2.0	1.0	0.0	1.0	90.0	172.0	30.0	2.0

Table 1.2: Employee Absenteeism Sample Data (Columns: 11-21)

Let us now try to understand the different variables of the data set.

1. Individual identification (ID) – Employee ID which uniquely identifies an employee
2. Reason for absence (ICD) - Absences attested by the International Code of Diseases (ICD) stratified into 21 categories (I to XXI) as follows:
   1. I Certain infectious and parasitic diseases
   2. II Neoplasms
   3. III Diseases of the blood and blood-forming organs and certain disorders involving the immune mechanism
   4. IV Endocrine, nutritional and metabolic diseases
   5. V Mental and behavioural disorders

6. VI Diseases of the nervous system
7. VII Diseases of the eye and adnexa
8. VIII Diseases of the ear and mastoid process
9. IX Diseases of the circulatory system
10. X Diseases of the respiratory system
11. XI Diseases of the digestive system
12. XII Diseases of the skin and subcutaneous tissue
13. XIII Diseases of the musculoskeletal system and connective tissue
14. XIV Diseases of the genitourinary system
15. XV Pregnancy, childbirth and the puerperium
16. XVI Certain conditions originating in the perinatal period
17. XVII Congenital malformations, deformations and chromosomal abnormalities
18. XVIII Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified
19. XIX Injury, poisoning and certain other consequences of external causes
20. XX External causes of morbidity and mortality
21. XXI Factors influencing health status and contact with health services.
22. Patient follow-up (22)
23. Medical consultation (23),
24. Blood donation (24),
25. Laboratory examination (25),
26. Unjustified absence (26),
27. Physiotherapy (27),
28. Dental consultation (28).

3. Month of absence – Different categorical months
4. Day of the week - (Monday (2), Tuesday (3), Wednesday (4), Thursday (5), Friday (6))
5. Seasons - (summer (1), autumn (2), winter (3), spring (4))

6. Transportation expense – Total amount spent by an Employee to reach office
7. Distance from Residence to Work (kilometers)
8. Service time – The total number of years served by employee
9. Age – Employee Age
10. Work load Average/day – Avg workload per day
11. Hit target – Meets target goals or not
12. Disciplinary failure (yes=1; no=0) – Any Disciplinary Issues at work
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 Variable

We need to predict Absenteeism time for future and also analyse the current trend and identify the reason for absence. From the above data, we can confirm that there are 20 Independent variables and 1 Dependent Variable in the data set Employee Absenteeism

Chapter 2

Methodology

PHASE 2 – DATA PREPARATION

The second phase of the Data Analytics Lifecycle involves data preparation, which includes the steps to explore, pre-process and condition data prior to modelling and analysis. In this phase, we must decide how to condition and transform data to get it into a format to facilitate subsequent analysis. Data visualizations help us understand the data including trends, outliers and relationships among data variables. Data preparation tends to be the most labour-intensive step in the analytics lifecycle.

2.1 Exploratory Data Analysis

Being the first stage of data preparation, in this stage, the data is analysed to understand the different variables and their data types. Studying the structure of the data set. Understanding the shape and properties of data.

- The data has a total of 21 variables with 740 observations.
- Renaming variables to make data more feasible

Emp_Id	Abs_Reason	Abs_Month	Abs_Day	Abs_Season
11	26	7	3	1
36	0	7	3	1
3	23	7	4	1
7	7	7	5	1

Table 2.1: Sample Data after Renaming Variables

- Checking for different data types and converting as required

- Identifying categorical and continuous variables from the data-set.
    - There are 11 Categorical Variables and 10 Continuous Variables in our data set.

```
Cont_Var          740 obs. of 10 variables
   Transport_Expense: num 289 118 179 279 289...
   Office_Distance : num 36 13 51 5 36 51 52 ...
   Service_Time : num 13 18 18 14 13 18 3 11 ...
   Emp_Age : num 33 50 38 39 33 38 28 36 34 3...
   Avg_Workload : num 239554 239554 239554 23...
   Hit_Target : num 97 97 97 97 97 97 97 97 9...
   Weight : num 90 98 89 68 90 89 80 65 95 88...
   Height : num 172 178 170 168 172 170 172 1...
   Emp_BMI : num 30 31 31 24 30 31 27 23 25 2...
   Abs_Hrs : num 4 0 2 4 2 ...
```

Table 2.2: Continuous Variables

```
Cat_Var           740 obs. of 11 variables
   Emp_Id : Factor w/ 36 levels "1","10","11"...
   Abs_Reason : Factor w/ 28 levels "1","10",...
   Abs_Month : Factor w/ 12 levels "1","10","...
   Abs_Day : Factor w/ 5 levels "2","3","4","...
   Abs_Season : Factor w/ 4 levels "1","2","3...
   Disciplinary_Failure: Factor w/ 2 levels "...
   Education : Factor w/ 4 levels "1","2","3"...
   Num_of_Kids : Factor w/ 5 levels "0","1","...
   Drinker : Factor w/ 2 levels "0","1": 2 2 ...
   Smoker : Factor w/ 2 levels "0","1": 1 1 1...
   Num_of_Pets : Factor w/ 6 levels "0","1","...
```

Table 2.3: Categorical Variables

2.1.1 Missing Value Analysis

Identifying the number of missing values in each variable and imputing to minimize errors.

Fig 2.1 gives the variation of missing values in the data set.

We use the KNN Imputation Method to eliminate missing values. The results after KNN Imputation is stored in the file “DataFile1_PostKNN.csv” as submitted.

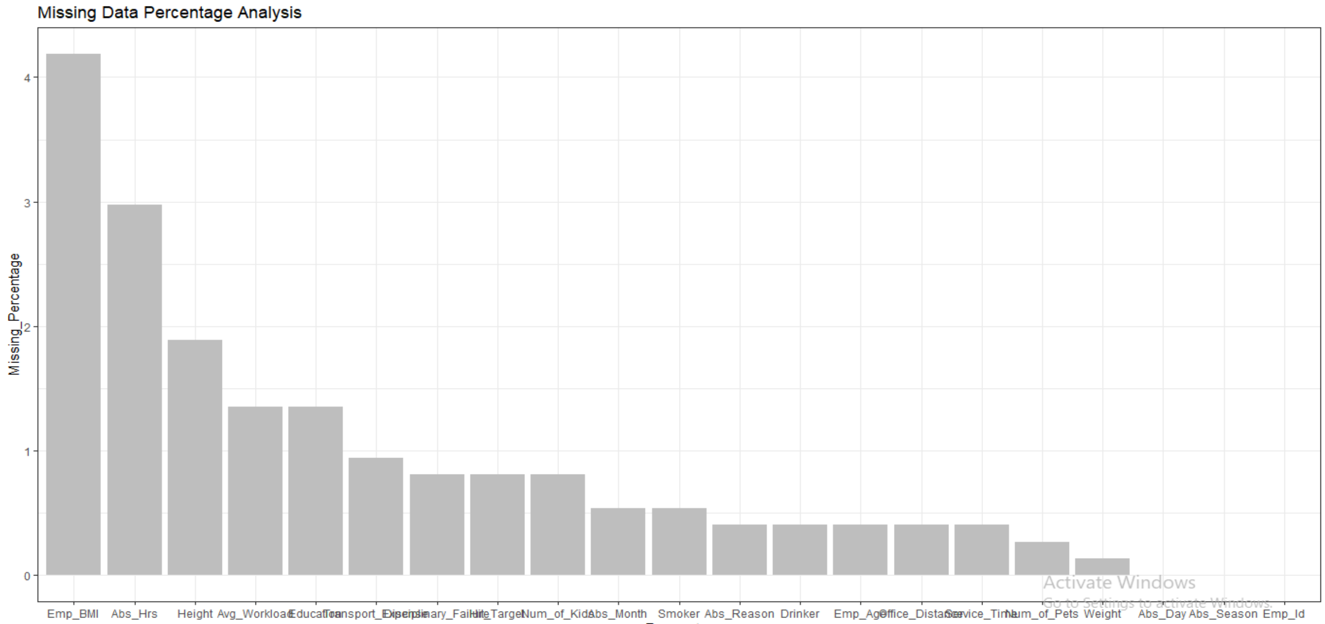

Fig 2.1: Missing Value Analysis

2.1.2 Outlier Analysis

One of the other steps of pre-processing apart from checking for normality is the presence of outliers. We visualize the outliers using boxplots. The below Outliers graph clearly depicts there are a number of outliers in Target variable and it needs to be eliminated.

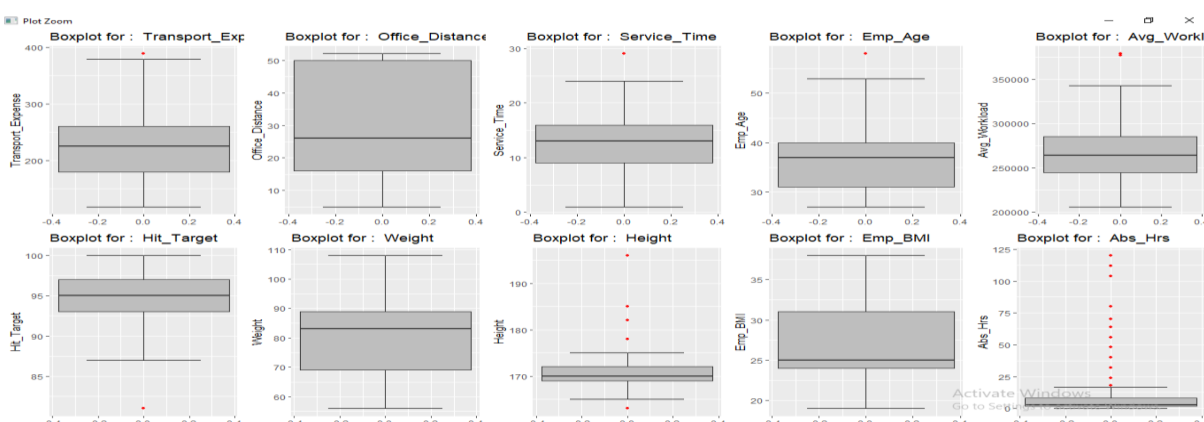

Fig 2.2: Box_Plots for Outlier Detection

2.1.3 Correlation Analysis

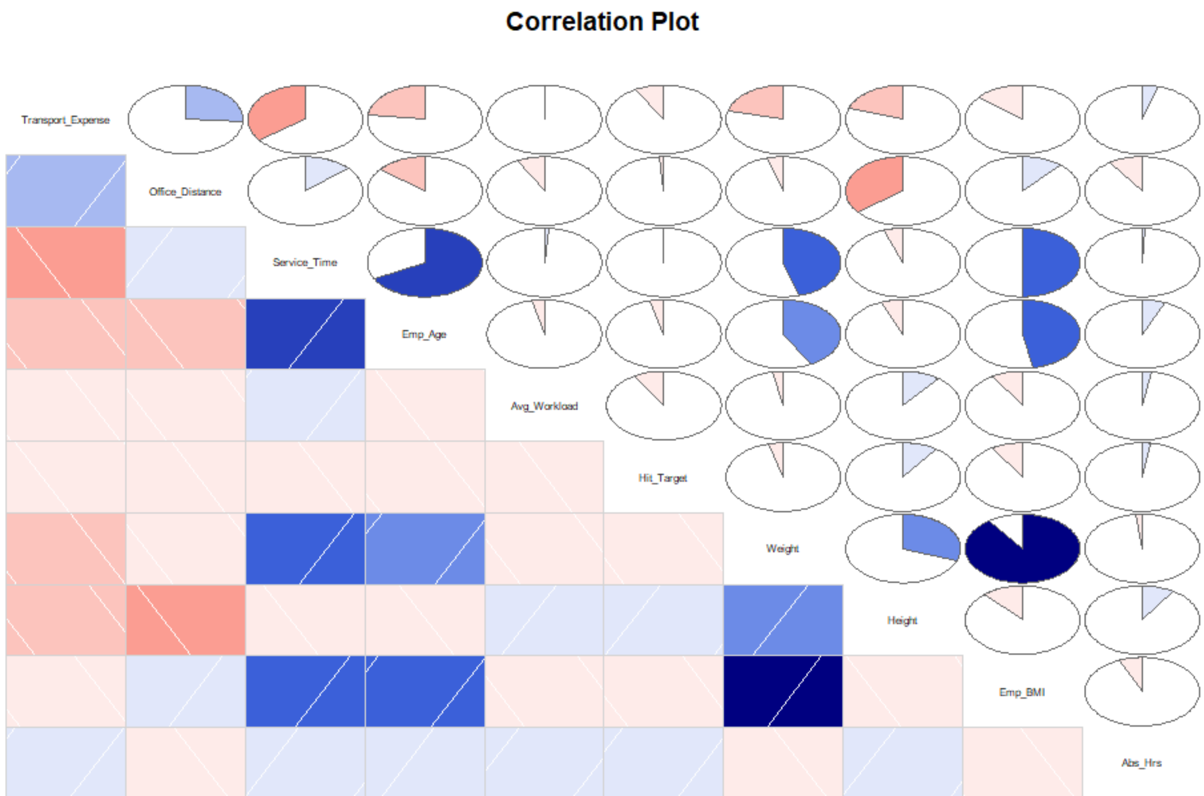

Fig 2.3: Correlation Analysis

This shows that there is multicollinearity in the dataset. BMI and Weight are highly correlated. Service Time and Age are also correlated.

Collinearity can be reduced by eliminating few variables.

PHASE 3: MODEL PLANNING

2.1.4 Distribution of Continuous Variables

This phase includes learning relationships between variables and subsequently selecting key variables and the most suitable models

Continuous Variable vs. Target Variable

Let us now analyse the relation of continuous variables with target variable

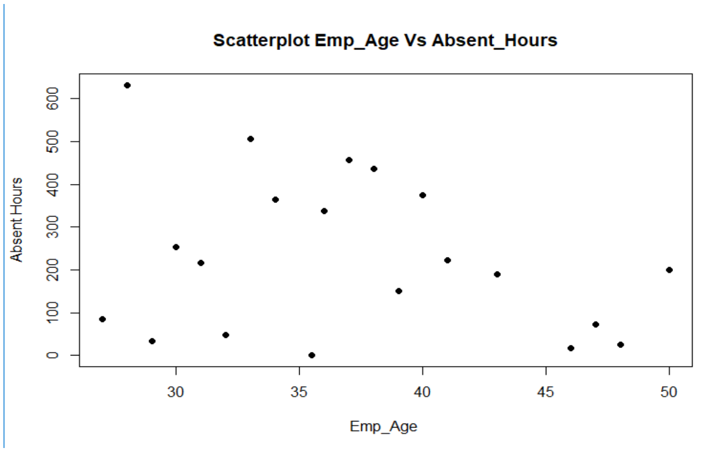

Fig 2.4: Scatter_Plot for Emp_Age vs Target Variable

From Fig 2.4, clearly, people over 40+ years of age tends to take less leaves compare to others

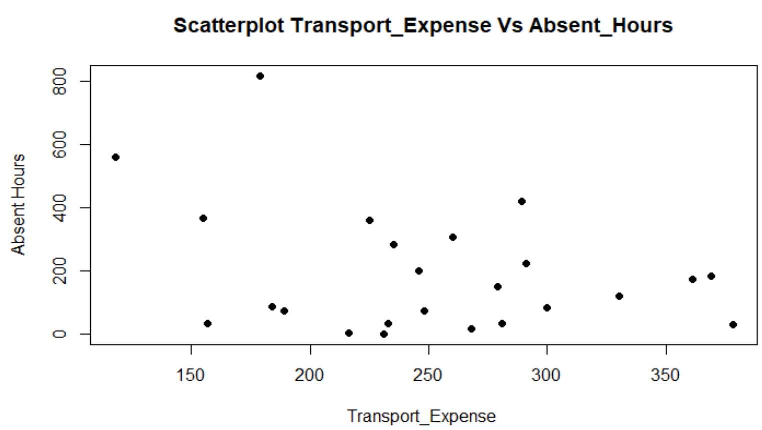

Fig 2.5: Scatter_Plot for Transport Expense vs Target Variable

This clearly shows concentration of leaves more where the Transportation Expense is between 150-300

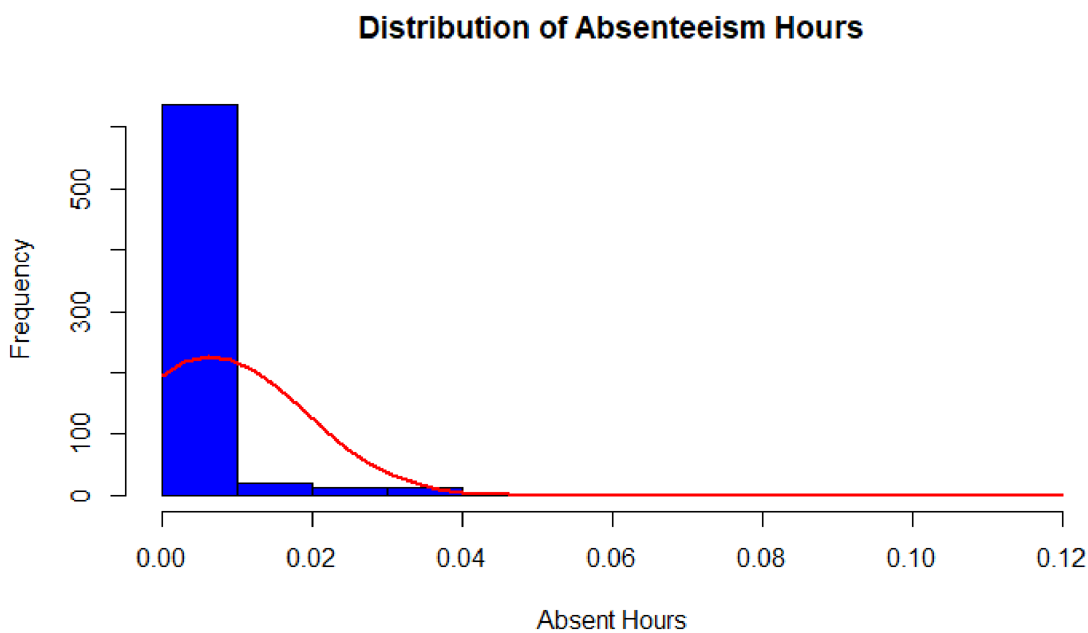

Fig 2.6: Distribution of Target Variable

What we can infer from above analysis of continuous variables:

- Target variable 'Absent_Hours' is not normally distributed, which is not a good thing.We have to look in to this, before feeding the data to model.
- 'Work_Distance','Age','Average_Workload' has good correlation with target feature 'Absent_Hours'. Let's drop others from further analysis.

2.1.5 Distribution of Categorical Variables

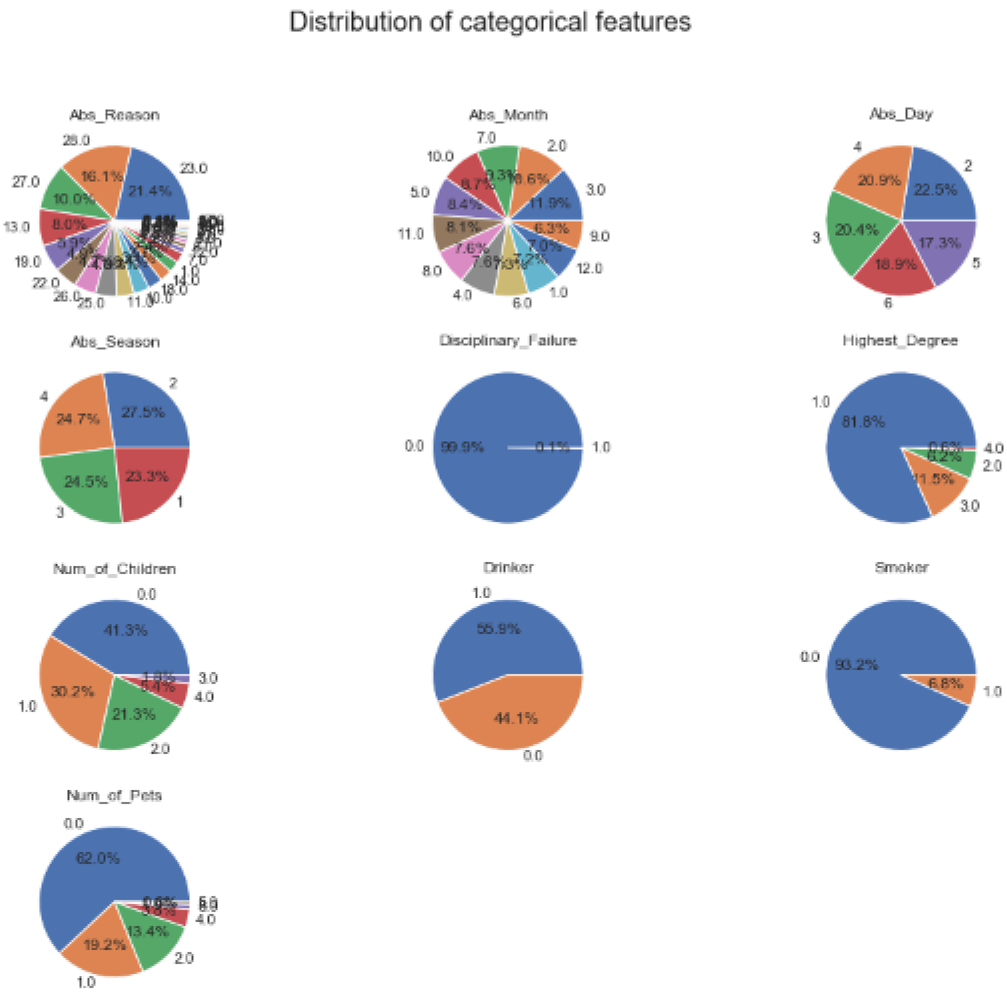

Fig 2.7: Pie Chart Depicting distribution of Categorical Variables

What we can infer from above pi-plot:

- From 'Reason' distribution, we can see that most frequent leaves are taken for the reason 23(Medical Consultation), 28(Dental Consultation), 27(Physiotherapy), 13(Diseases of the musculoskeletal system and connective tissue), 19(Injury, poisoning and certain other consequences of external causes,10(Diseases of the respiratory system)

- From, 'Month' distribution, we can see that frequency of leaves are more or less uniformly distributed over months, with highest no. of leaves taken in March, Feb and July(holiday season)
- From, 'Education' distribution, we can see that frequency of leaves are highest for education = 1(high school)
- From, 'Weekday' distribution, we can see that frequency of leaves are mostly distributed, with most frequent leaves on 'Monday', which makes sense as most people travel/party over weekend and the mood spills over to Monday
- From, 'Son' and 'pet', we can see that people having no kids and no pets (no family responsibilities) tend to take frequent leaves.
- 'Social Drinker' takes little more leaves than non-drinker.

Let us now further analyse the reason and find alternatives to reduce the absence hours.

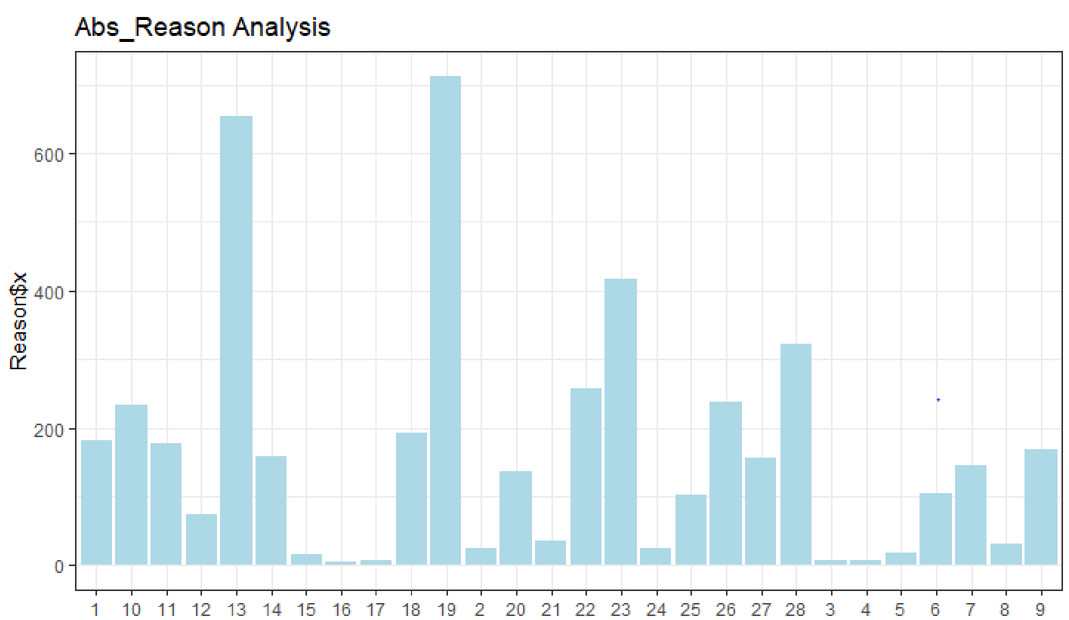

Fig 2.8: Abs_Reason vs Hours

Longest hours of absences for reason medical consultation (23),blood donation (24), physiotherapy (27), dental consultation (28),Diseases of the musculoskeletal system and connective tissue(13),Injury, poisoning and certain other consequences of external causes(19)

Overall, the data can be justified as employees take most absences for medical consultations/dental consultation/physiotherapy.

Genuine Solution:

- These hours can probably be reduced by setting up a medical consultation/dental consultation/physiotherapy sessions at office/facility on a weekly/monthly basis.
- In long term, introducing exercise/yoga sessions in office once/twice a week will help reduce physiotherapy issues

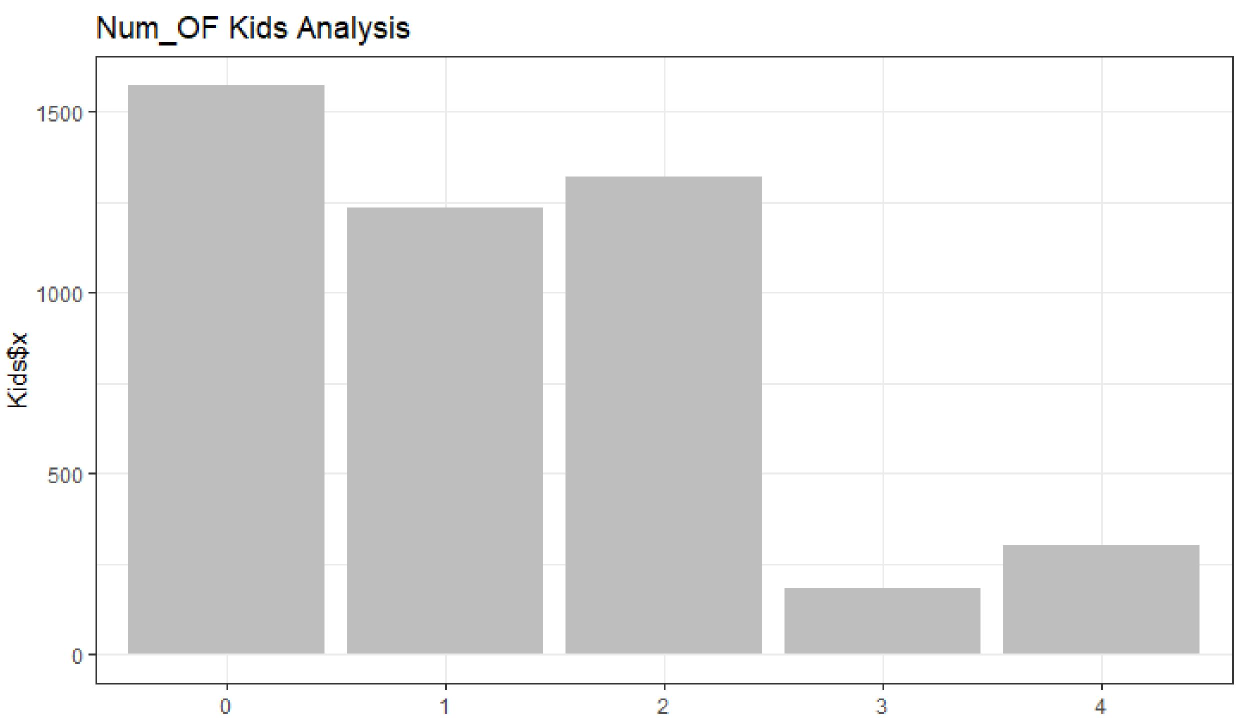

Fig 2.9: Num of kids vs Hours

From above figure, employee with 3-4 kids tend to take less hours of absence

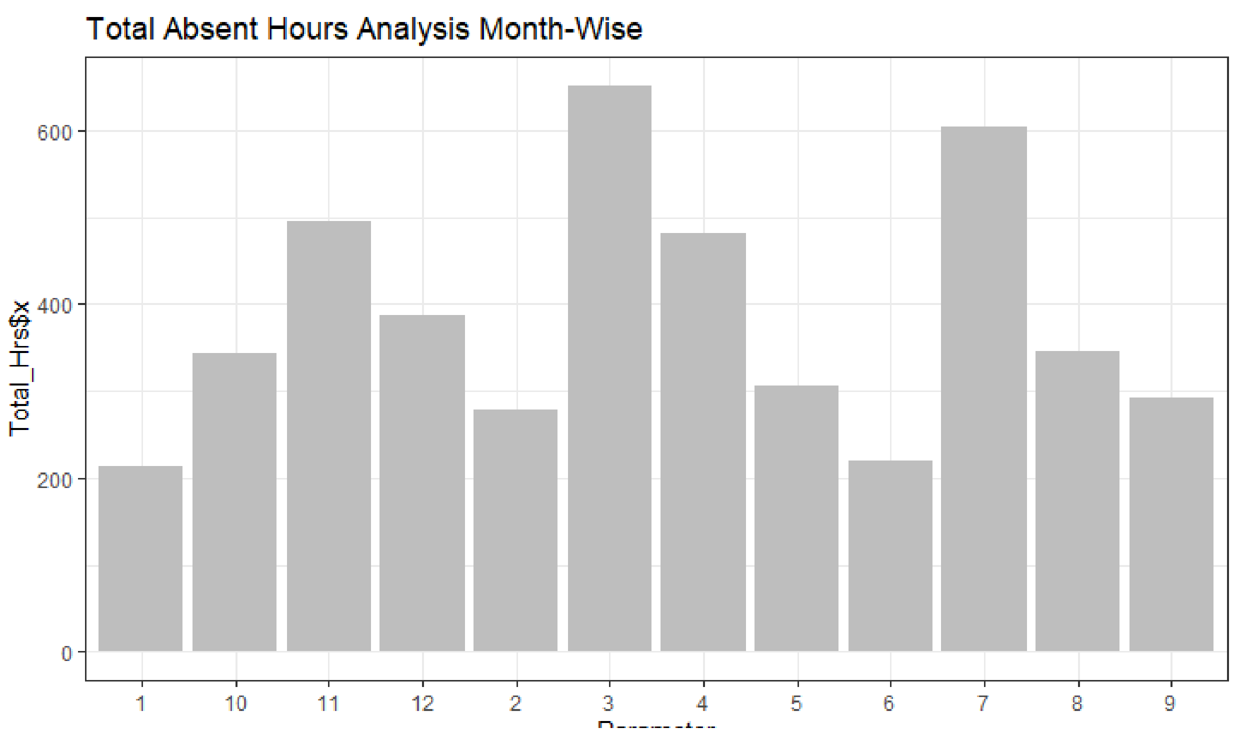

Fig 2.10: Abs_Month vs Hours

Clearly from above figure, March tops the month for most absences. This makes sense as this is peak holiday season. Second one is July, which again is the 'holiday' season

2.2 Feature Selection

Before performing any type of modelling we need to assess the importance of each predictor variable in our analysis. There is a possibility that many variables in our analysis are not important at all to the problem of class prediction. There are several methods of doing that. Below we have used Random Forests to perform features selection.
Considering above analysis, we consider Abs_month, Emp_id, Avg_workload, Office_Distance, Num of kids, smoker, abs_hrs, emp_age, and service_time and eliminate the rest from our data.

As part of training model, we consider 80% of standardized data as training data and rest 20% as test data.

2.3 Modeling

PHASE 4: MODEL BUILDING

Develop datasets for training, testing and production purposes along with building and executing model.

2.3.1 Model Building

In our early stages of analysis during pre-processing we have come to understand how continuous and categorical variables are related to target variable. We use the Random Forest Algorithm for predicting target variable.

Attached below are screenshots of predicted values post usage of Random Forest Algorithm.

Column Re – Abs_Month

X – Predicted hours

	Re	x
1	1	343.3071
2	10	421.0316
3	11	428.3961
4	12	323.7817
5	2	474.1179
6	3	702.1121
7	4	307.4281
8	5	278.1819
9	6	238.6542
10	7	455.2189
11	8	479.6812
12	9	313.8992

Fig 2.11: Values for 2010

Similarly, I have used the Random Forest to predict values for 2011 as well.

	Re	x	monthly_loss_percentage
1	1	236.7241	3.736177
2	10	359.1511	5.668420
3	11	436.1243	6.883276
4	12	368.2510	5.812042
5	2	295.6365	4.665980
6	3	694.1792	10.956112
7	4	466.5980	7.364236
8	5	306.2080	4.832828
9	6	216.6819	3.419853
10	7	630.1858	9.946115
11	8	332.6404	5.250007
12	9	318.3991	5.025238

Fig 2.12: Values for 2011

Chapter 3

Conclusion

PHASE 5 : COMMUNICATE RESULTS

Checking if results are successful or not.

3.1 Model Evaluation

Now that we have a few models for predicting the target variable, we need to decide which one to choose. There are several criteria that exist for evaluating and comparing models. We can compare the models using any of the following criteria:

1. Predictive Performance
2. Interpretability
3. Computational Efficiency

In our case of Wine Data, the latter two, *Interpretability* and *Computation Efficiency*, do not hold much significance. Therefore we will use *Predictive performance* as the criteria to compare and evaluate models.

Predictive performance can be measured by comparing Predictions of the models with real values of the target variables, and calculating some average error measure.

I tried implementing the data on 3 models and chose the best

	Model	RootMeanSquaredError
0	Linear Reg	0.009622
1	Random Forest	0.008849
2	GradientBoost	0.009072

RMSE has the benefit of penalizing large errors more so can be more appropriate and hence I have considered RMSE as metric.

PHASE 6 : OPERATIONALIZE

Delivering final reports, briefings, code and other documents.

COMPLETE R CODE

```
#Clear R environment
rm(list=ls(all=T))

#Set Current Working Directory
setwd("C:/Users/HP/Desktop/Edwisor/Project 1")

#Load Libraries
x = c("ggplot2", "corrgram", "DMwR", "caret", "randomForest", "unbalanced", "C50",
"dummies", "e1071", "Information",
     "MASS", "rpart", "gbm", "ROSE", 'sampling', 'DataCombine', 'inTrees')

#Install Packages
install.packages(x)
lapply(x, require, character.only = TRUE)
rm(x)

#Read Data
Absenteeism_Data = read.csv("Absenteeism_at_work_Project.csv", header = T, na.strings = c("
", "", "NA"))
Data_Copy = Absenteeism_Data
###-------------------------------------------DATA EXPLORATION-------------------------------------------###
#Check Structure of data
str(Absenteeism_Data)

#Renaming Column Variables
names(Absenteeism_Data)[1] = "Emp_Id"
```

```
names(Absenteeism_Data)[2] = "Abs_Reason"
names(Absenteeism_Data)[3] = "Abs_Month"
names(Absenteeism_Data)[4] = "Abs_Day"
names(Absenteeism_Data)[5] = "Abs_Season"
names(Absenteeism_Data)[6] = "Transport_Expense"
names(Absenteeism_Data)[7] = "Office_Distance"
names(Absenteeism_Data)[8] = "Service_Time"
names(Absenteeism_Data)[9] = "Emp_Age"
names(Absenteeism_Data)[10] = "Avg_Workload"
names(Absenteeism_Data)[11] = "Hit_Target"
names(Absenteeism_Data)[12] = "Disciplinary_Failure"
names(Absenteeism_Data)[14] = "Num_of_Kids"
names(Absenteeism_Data)[15] = "Drinker"
names(Absenteeism_Data)[16] = "Smoker"
names(Absenteeism_Data)[17] = "Num_of_Pets"
names(Absenteeism_Data)[20] = "Emp_BMI"
names(Absenteeism_Data)[21] = "Abs_Hrs"

#Converting data types as required
Absenteeism_Data$Emp_Id = as.factor(as.character(Absenteeism_Data$Emp_Id))
Absenteeism_Data$Abs_Reason[Absenteeism_Data$Abs_Reason %in% 0] = 20
Absenteeism_Data$Abs_Reason <- as.factor(as.character(Absenteeism_Data$Abs_Reason))
Absenteeism_Data$Abs_Month[Absenteeism_Data$Abs_Month %in% 0] = NA
Absenteeism_Data$Abs_Month <- as.factor(as.character(Absenteeism_Data$Abs_Month))
Absenteeism_Data$Abs_Day <- as.factor(as.character(Absenteeism_Data$Abs_Day))
Absenteeism_Data$Abs_Season <- as.factor(as.character(Absenteeism_Data$Abs_Season))
Absenteeism_Data$Disciplinary_Failure <-
as.factor(as.character(Absenteeism_Data$Disciplinary_Failure))
Absenteeism_Data$Education <- as.factor(as.character(Absenteeism_Data$Education))
```

```
Absenteeism_Data$Num_of_Kids <- as.factor(as.character(Absenteeism_Data$Num_of_Kids))
Absenteeism_Data$Drinker <- as.factor(as.character(Absenteeism_Data$Drinker))
Absenteeism_Data$Smoker <- as.factor(as.character(Absenteeism_Data$Smoker))
Absenteeism_Data$Num_of_Pets <- as.factor(as.character(Absenteeism_Data$Num_of_Pets))

###-----------------------------------MISSING VALUE ANALYSIS---------------------------------------###
#Creating New DataSet with Missing Values Info
Missing_Data = data.frame(apply(Absenteeism_Data,2,function(x){sum(is.na(x))}))
Missing_Data$Columns = row.names(Missing_Data)

#Creating New Variable in Missing_Val DataSet
names(Missing_Data)[1] = "Missing_Percentage"
Missing_Data$Missing_Percentage =
(Missing_Data$Missing_Percentage/nrow(Absenteeism_Data)) * 100

#Sorting in Descending Order
Missing_Data = Missing_Data[order(-Missing_Data$Missing_Percentage),]
row.names(Missing_Data) = NULL
Missing_Data = Missing_Data[,c(2,1)]
write.csv(Missing_Data, "Missing_Value_Analysis1.csv", row.names = F)

#Plotting a Bar-Graph for Missing Value Analysis
ggplot(data = Missing_Data[1:21,], aes(x=reorder(Columns, -Missing_Percentage),y =
Missing_Percentage))+geom_bar(stat = "identity",fill =
"grey")+xlab("Parameter")+ggtitle("Missing Data Percentage Analysis") + theme_bw()

# KNN Imputation
Absenteeism_Data = knnImputation(Absenteeism_Data, k = 3)
```

```
#Checking for any Missing Value in the data-set
sum(is.na(Absenteeism_Data))

#Data with no missing values
write.csv(Absenteeism_Data, 'DataFile1_PostKNN.csv', row.names = F)

###----------------------------------------OUTLIER ANALYSIS------------------------------------------###
#Identifying Continuous Variables
Cont = sapply(Absenteeism_Data,is.numeric)
Cont_Var = Absenteeism_Data[,Cont]

#Identifying Categorical Variables
Cat = sapply(Absenteeism_Data,is.factor)
Cat_Var = Absenteeism_Data[,Cat]

#Distribution of Continuos Variables using Box-Plots
for(i in 1:ncol(Cont_Var)) {
  assign(paste0("box",i), ggplot(data = Absenteeism_Data, aes_string(y = Cont_Var[,i])) +
        stat_boxplot(geom = "errorbar", width = 0.5) +
        geom_boxplot(outlier.colour = "red", fill = "grey", outlier.size = 1) +
        labs(y = colnames(Cont_Var[i])) +
        ggtitle(paste("Boxplot for : ",colnames(Cont_Var[i]))))
}

#Drawing Box-Plots for each Continuos Variable
gridExtra::grid.arrange(box1,box2,box3,box4,box5,box6,box7,box8,box9,box10,ncol=5)

rm(df)
```

```
#Remove outliers using boxplot method
Copy2 = Absenteeism_Data
Absenteeism_Data = Copy2
for(i in 1:10){
  print(i)
  val = Absenteeism_Data[,i][Absenteeism_Data[,i] %in%
boxplot.stats(Absenteeism_Data[,i])$out]
  print(length(val))
  Absenteeism_Data = Absenteeism_Data[which(!Absenteeism_Data[,i] %in% val),]
}

#Replace all outliers with NA and impute
for(i in 1:10){
val = Absenteeism_Data[,i][Absenteeism_Data[,i] %in%
boxplot.stats(Absenteeism_Data[,i])$out]
print(length(val))
Absenteeism_Data[,i][Absenteeism_Data[,i] %in% val] = NA
}

#KNN Imputation
Absenteeism_Data = knnImputation(Absenteeism_Data, k = 3)
sum(is.na(Absenteeism_Data))

###----------------------------------------Feature Selection----------------------------------------------###
## Correlation Plot
corrgram(Cont_Var, order = F,
        upper.panel=panel.pie, text.panel=panel.txt, main = "Correlation Plot")
```

```
#Exploring Relationship Between Independent Continous Variables and Dependent
Variable('Absenteeism Hours') using scatter plot
Age <- aggregate(Absenteeism_Data$Abs_Hrs, by=list(Age=Absenteeism_Data$Emp_Age),
FUN=sum)
plot(Age$Age, Age$x, main="Scatterplot Emp_Age Vs Absent_Hours",
    xlab="Emp_Age ", ylab="Absent Hours", pch=19)

Expense <- aggregate(Absenteeism_Data$Abs_Hrs,
by=list(Expense=Absenteeism_Data$Transport_Expense), FUN=sum)
plot(Expense$Expense, Expense$x, main="Scatterplot Transport_Expense Vs Absent_Hours",
    xlab="Transport_Expense ", ylab="Absent Hours", pch=19)

ServiceT <- aggregate(Absenteeism_Data$Abs_Hrs,
by=list(SerT=Absenteeism_Data$Service_Time), FUN=sum)
plot(ServiceT$SerT, ServiceT$x, main="Scatterplot Service_Time Vs Absent_Hours",
    xlab="Emp_Age ", ylab="Absent Hours", pch=19)

Distance <- aggregate(Absenteeism_Data$Abs_Hrs,
by=list(SerT=Absenteeism_Data$Office_Distance), FUN=sum)
plot(ServiceT$SerT, ServiceT$x, main="Scatterplot Office_Distance Vs Absent_Hours",
    xlab="Emp_Age ", ylab="Absent Hours", pch=19)

#Checking the Distribution of Dependent Variable('Absenteeism Hours') using Histogram with
Normal Curve
x <- (Absenteeism_Data$Abs_Hrs)/1000
h<-hist(x, breaks=10, col="blue", xlab="Absent Hours",
      main="Distribution of Absenteeism Hours")
xfit<-seq(min(x),max(x),length=40)
yfit<-dnorm(xfit,mean=mean(x),sd=sd(x))
```

```
yfit <- yfit*diff(h$mids[1:2])*length(x)
lines(xfit, yfit, col="red", lwd=2)

#Exploring Distribution of Categorical variables with Dependent Variable('Absenteeism Hours')
for (i in 1:11){
  print(paste("Pie Distribution for", colnames(Cat_Var[i])))
  pie(table(Cat_Var[i]),main = paste("Pie Distribution for", colnames(Cat_Var[i])))
}

#checking the top reasons for absence as per the total numbers of absence
Reason <- aggregate(Absenteeism_Data$Abs_Hrs, by=list(Re=Absenteeism_Data$Abs_Reason),
FUN=sum)
ggplot(data = Reason, aes(x= Reason$Re,y = Reason$x))+geom_bar(stat = "identity",fill =
"lightblue")+xlab("Parameter")+ggtitle("Abs_Reason Analysis") + theme_bw()

#Analyzing absence dependency of no of kids
Kids <- aggregate(Absenteeism_Data$Abs_Hrs,
by=list(Num_kids=Absenteeism_Data$Num_of_Kids), FUN=sum)
ggplot(data = Kids, aes(x= Kids$Num_kids,y = Kids$x))+geom_bar(stat = "identity",fill =
"grey")+xlab("Parameter")+ggtitle("Num_OF Kids Analysis") + theme_bw()
#Analyzing absence dependency of month of year
Total_Hrs <- aggregate(Absenteeism_Data$Abs_Hrs,
by=list(tot=Absenteeism_Data$Abs_Month), FUN=sum)
ggplot(data = Total_Hrs, aes(x= Total_Hrs$tot,y = Total_Hrs$x))+geom_bar(stat = "identity",fill =
"grey")+xlab("Parameter")+ggtitle("Total Absent Hours Analysis Month-Wise") + theme_bw()

#Dimension Reduction for Second part of problem
Absenteeism_Data_1 = subset(Absenteeism_Data,
```

```
                    select = -
c(Abs_Reason,Abs_Day,Abs_Season,Hit_Target,Transport_Expense,Disciplinary_Failure,Educati
on,Smoker,Num_of_Pets,Weight,Height,Emp_BMI))
###----------------------------PART 1 of PRoblem Ends Here-----------------------------------------#####

###___________________________Part 2 Begins________      ____###

###--------------------------------FEATURE SCALING------------------------------------------------###

#Identifying Continuous Variables
Cont1 = saapply(Absenteeism_Data_1,is.numeric)
Cont_Var1 = Absenteeism_Data_1[,Cont1]

#Removing Target Variable
num = names(Cont_Var1)p
num = num[-5]

#Identifying Categorical Variables
Cat1 = sapply(Absenteeism_Data_1,is.factor)
Cat_Var1 = Absenteeism_Data_1[,Cat1]

# #Standardisation
for(i in num){
  print(i)
  Absenteeism_Data_1[,i] = (Absenteeism_Data_1[,i] - mean(Absenteeism_Data_1[,i]))/
    sd(Absenteeism_Data_1[,i])
}
###----------------------------------------MODEL DEVELOPMENT-----------------------------------------###
#Generating Training and Test Data Set
```

```
set.seed(1)
train_index = sample(1:nrow(Absenteeism_Data_1), 0.8 * nrow(Absenteeism_Data_1))
train = Absenteeism_Data_1[train_index,]
test = Absenteeism_Data_1[-train_index,]

###############Random Forest
#Train model using Training Data
RF_model_1 = randomForest(Abs_Hrs ~ ., train, importance = TRUE, ntree = 100)

#Extract rules fromn random forest
#transform rf object to an inTrees' format
treeList = RF2List(RF_model)

#Extract rules
exec = extractRules(treeList, train[,-9])  # R-executable conditions

# #Visualize some rules
exec[1:2,]

# #Make rules more readable:
readableRules = presentRules(exec, colnames(train))
readableRules[1:2,]

# #Get rule metrics
ruleMetric = getRuleMetric(exec, train[,-9], train$Abs_Hrs)  # get rule metrics

# #evaulate few rules
ruleMetric[1:2,]
```

```
#Presdict test data using random forest model
RF_Predictions = predict(RF_model_1, test[,-10])

#Create dataframe for actual and predicted values
rf_pred = data.frame("actual"=test[,-10], "rf_pred"=RF_Predictions)
head(rf_pred)

#Calcuate MAE, RMSE, R-sqaured for testing data
print(postResample(pred = RF_Predictions, obs = test[,9]))

abs_pred2010 = Absenteeism_Data_1[,-c(1,8)]
abs_pred2010$Predicted <- (rf_pred$rf_pred)

Abs_Predict_2010 <- aggregate(abs_pred2010$Predicted ,
by=list(Re=abs_pred2010$Abs_Month), FUN=sum)

###--------------------------------------------Prediction--------------------------------------------###
sum(test$Abs_Hrs)

#Sort Data by Absence Month and View Predicted Data
Abs_Predict <- aggregate(rf_pred$rf_pred , by=list(Re=rf_pred$actual.Abs_Month), FUN=sum)

#For 2011 Data
emp_2011 = Absenteeism_Data_1
emp_2011$Service_Time = Absenteeism_Data_1$Service_Time + 1
emp_2011$Emp_Age = Absenteeism_Data_1$Emp_Age + 1

#Exclude Emp_Id and Abs_hrs
emp_2011= emp_2011[,-c(1,8)]
```

```
# #Standardisation
for(i in num){
  print(i)
  emp_2011[,i] = (emp_2011[,i] - mean(emp_2011[,i]))/
    sd(emp_2011[,i])
}

predict_2011 = randomForest(Abs_Hrs ~ ., emp_2011, ntree = 500)
rf_predictions1 = predict(predict_2011, emp_2011)
rf_pred = data.frame("actual"=emp_2011, "rf_pred"=rf_predictions1)

abs_pred2011 = emp_2011
abs_pred2011$Predicted <- (rf_pred$rf_pred)

Abs_Predict_2011 <- aggregate(abs_pred2011$Predicted ,
by=list(Re=abs_pred2011$Abs_Month), FUN=sum)

tot_Monthly_hours = 22*8*36

Abs_Predict_2011$monthly_loss_percentage = (Abs_Predict_2011$x/tot_Monthly_hours) * 100
```

References

Data Science and Big Data Analytics 2017, Wiley , EMC EDUCATION Services