Predicting Absenteeism

Pranjul Gupta

24 February 2018

Contents

Chapter 1

Introduction

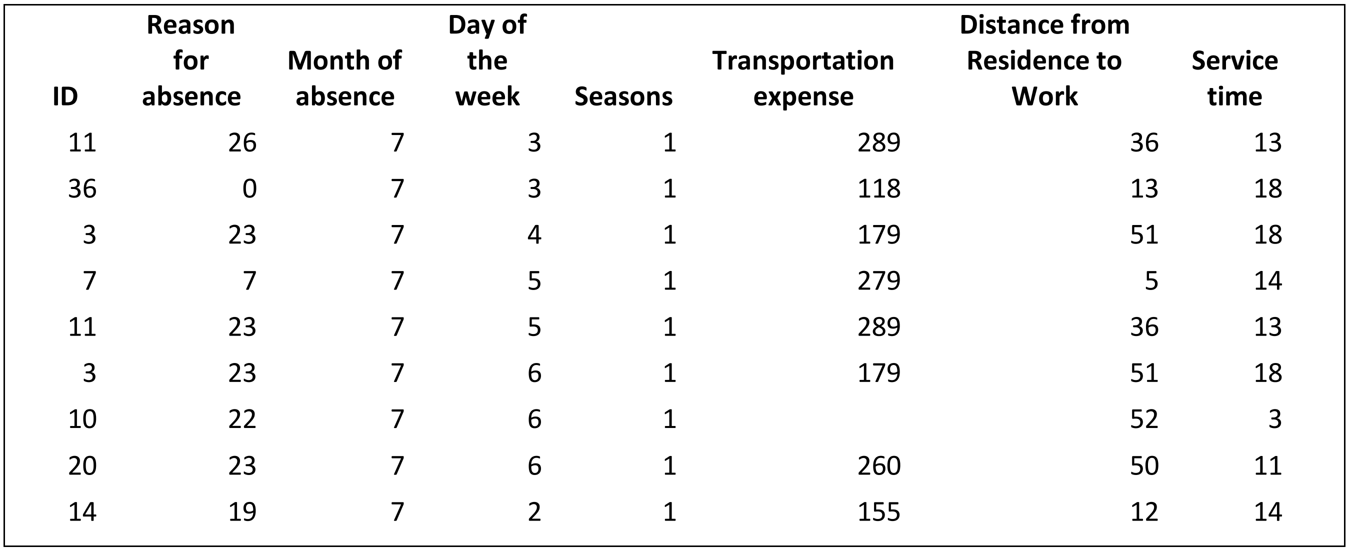
1.1 Problem Statement

Absenteeism is a problem of parcel industry, where each employee's accountability is so important for good productivity of company. Each employee is equally important for the company to client customer satisfaction. In this even for hourly absent in work also effects the delivery service time, and ultimately it is getting adverb effect on profitability, accountability and business operations. This data set actually showing employee absent in hours in a particular month, season and day. We need to predict future absent of hours in month, and provide solutions we should take to make presentences of employees.

1.2 Data

Our task is to build a model to predict future absent hours of employees, as our set is looking like a normal data which have different numerical variables, but if we observe the data it is a time series data which have seasonality. Here are sample of the various predictor variables.

Table 1.1: Absenteeism Sample Data (Columns: 1-8)



ID	Reason for absence	Month of absence	Day of the week	Seasons	Transportation expense	Distance from Residence to Work	Service time
11	26	7	3	1	289	36	13
36	0	7	3	1	118	13	18
3	23	7	4	1	179	51	18
7	7	7	5	1	279	5	14
11	23	7	5	1	289	36	13
3	23	7	6	1	179	51	18
10	22	7	6	1		52	3
20	23	7	6	1	260	50	11
14	19	7	2	1	155	12	14

Table 1.2: Absenteeism Sample Data (Columns: 9-16)

Age	Work load Average/day	Hit target	Disciplinary failure	Education	Son	Social drinker	Social smoker
33	239,554	97	0	1	2	1	0
50	239,554	97	1	1	1	1	0
38	239,554	97	0	1	0	1	0
39	239,554	97	0	1	2	1	1
33	239,554	97	0	1	2	1	0
38	239,554	97	0	1	0	1	0
28	239,554	97	0	1	1	1	0
36	239,554	97	0	1	4	1	0
34	239,554	97	0	1	2	1	0

Table 1.3: Absenteeism Sample Data (Columns: 17-21)

Pet	Weight	Height	Body mass index	Absenteeism time in hours
1	90	172	30	4
0	98	178	31	0
0	89	170	31	2
0	68	168	24	4
1	90	172	30	2
0	89	170	31	4
4	80	172	27	8
0	65	168	23	4
0	95	196	25	40

Table 1.4: Absenteeism data set predictor Variables

Variables List 1	Variable List 2
ID	Work load
Average/day	Disciplinary failure
Hit target	Education
Reason for absence	Son
Month of absence	Distance from Residence to Work
Day of the week	Service time
Transportation expense	Weight
Seasons	Height
Body mass index	Absenteeism time in hours

Social smoker	Social drinker

Methodology

2.1 Pre Processing

Preprocessing is very important part of any predictive analysis, we need to select right data for the model. If you choose data which is biased and supporting some special cases, so it will generate biased and inaccurate results. We can't choose right data just looking on it, so we used analysis methods and graphs to see the right data for model. We use box plot, histogram, scatter plot and various graph for this, we are using the term **Exploratory Data Analysis** for all these process.

In figure 2.1, we plotted the box plots for various variables, and we can easily identify the some instances are really out of range, and these data can make erroneous prediction model, and along with this data is also tested on correlation diagram.

2.1.1 Outlier Analysis

We can clearly observe the predictive variables are skewed, near to mean, cyclic. The observations we get from the graph that body mass index is dependent on height and weight,

Various methods for outliers, one of the other steps of **pre-processing** apart from checking for normality, is the presence of outliers. In this case we use a classic approach of removing outliers. We visualize the outliers using *boxplots*.

In preprocessing, we filter the data on the basis missing values, outliers [Turkey's method[1]] and skewness[2] tendency of data. Here we have the whole population, and we applied the analysis on each variable, some variables like ID are always unique, month and season have cyclic data, so we have different tendencies, we need to recognize them for best model.

This data having features of time series data, so we also checked for **seasonality**, **trend** and **cyclic** tendencies. So we observed that this data is actually cyclic flow with the respect to absent hours and month.

[1] In Turkey's Method, outliers have been defined as the data points which are ±1.5*SD*, and should be removed from the data. It was given by J. W. Turkey in his famous 1977 book *Exploratory Data Analysis*[2]

Skewness is asymmetry in a statistical distribution, in which the curve appears distorted or skewed either to the left or to the right.

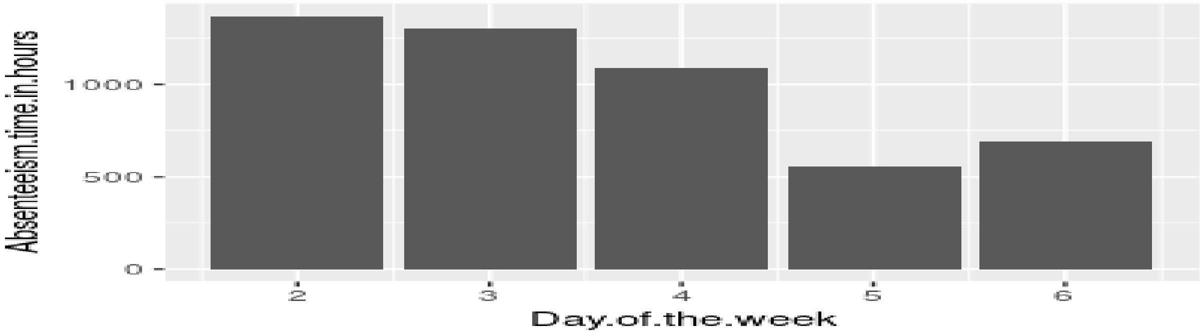


Figure 2.1 - **Plot of hours according to days of week**

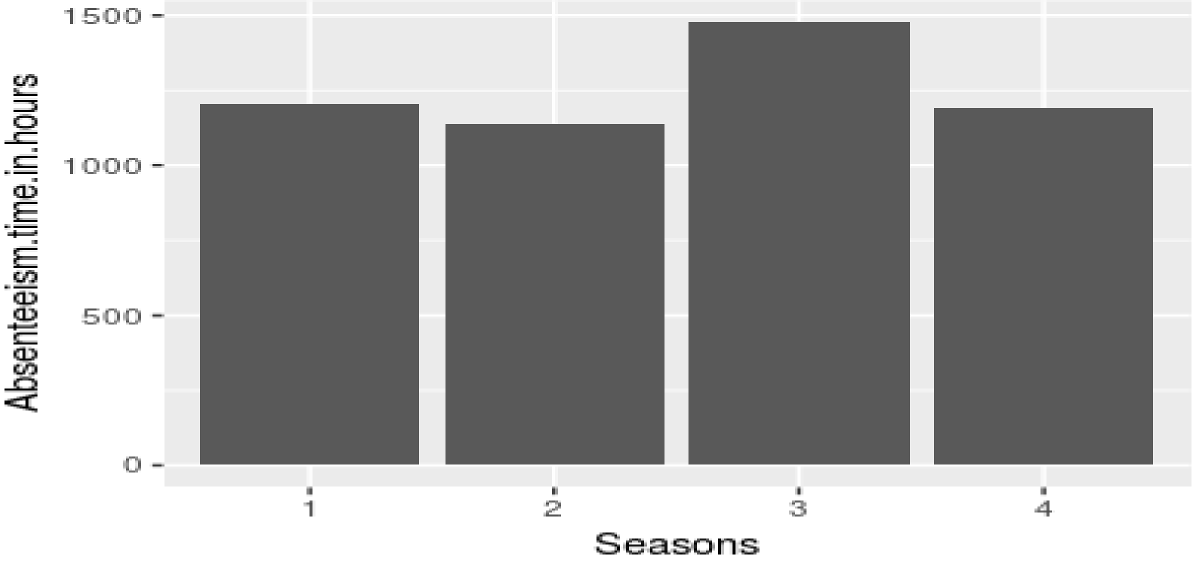


Figure 2.2 - **Plot of Hours according to the Seasons of week**

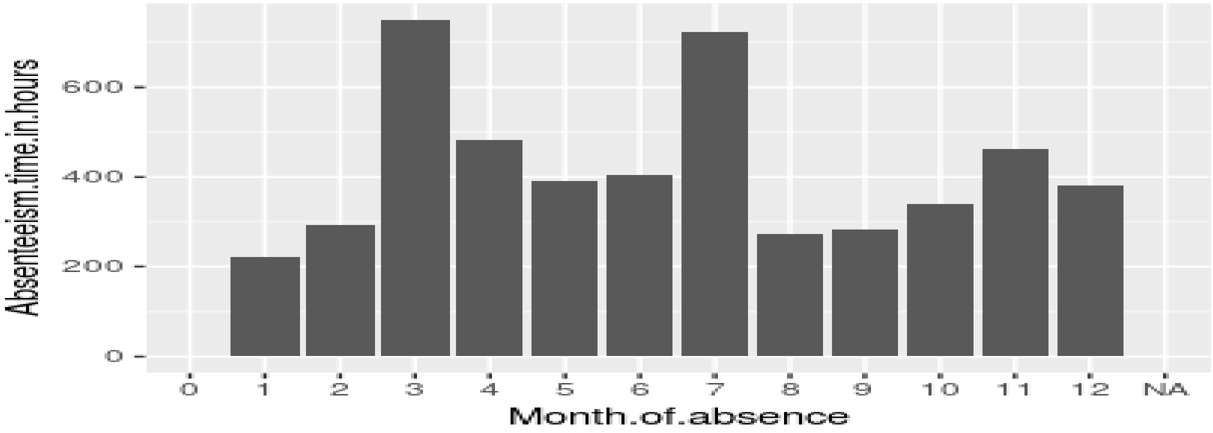


Figure 2.3- **Plot of hours according to the Month**

We plotted bar graphs above, as we observed that from above graph fig 2.2, in season plot clearly in season 3,4 having more number of absent hours. In month plots 3, 4 and 5 having more number of hours comparing to any other month, so we could say these data sets are following seasonality

For getting better stimulation, plotting the hour's data graph with the respect to year cycle. And we will have prescribed and detailed aspect of data.

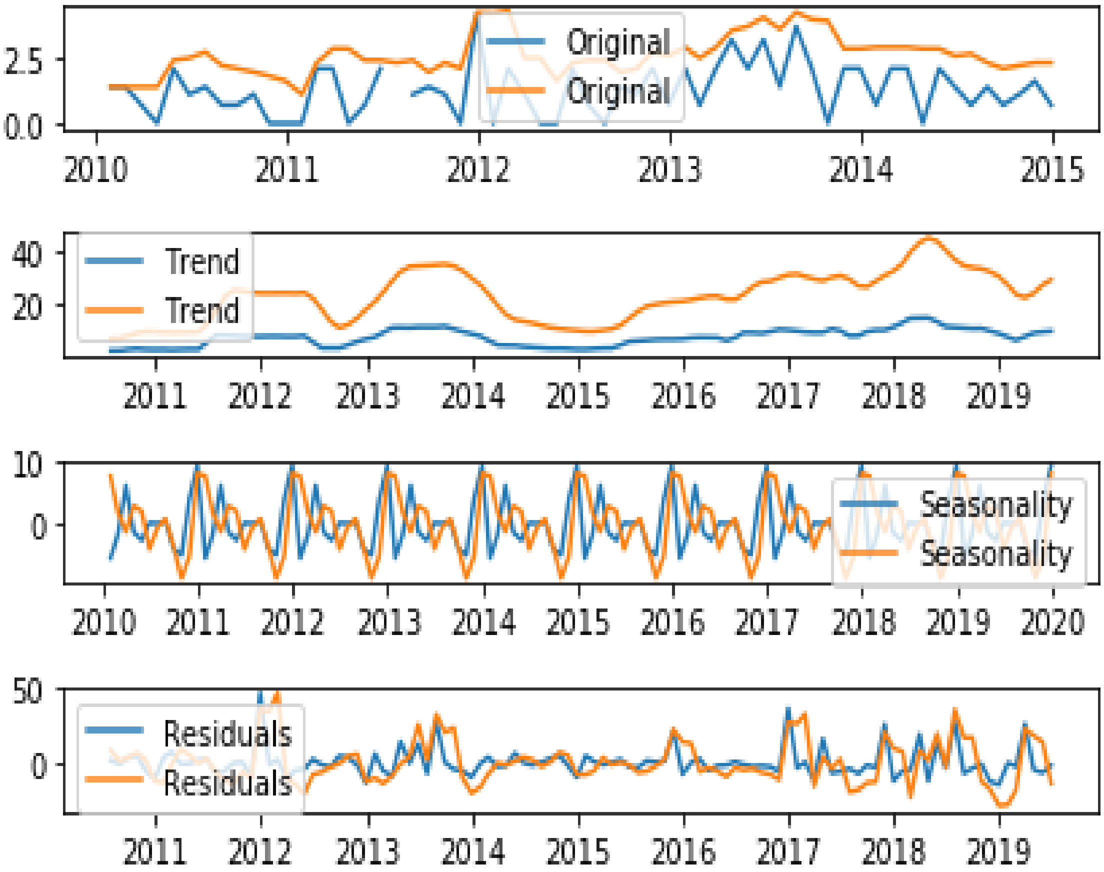


Figure 2.4- Figure are showing Original, trend, seasonality and Residuals of data

In above graph data is swinging in period, and that is confirming seasonality is, but we can see there is no trend in actual data. With all these observation we can tell, there might be an occasion, or something important, that can be a reason of more absenteeism.

Then we plotted the box plots to check the outliers of data, first drawn as whole data, then we plotted data with classification month, season and days.

Here below box plots of variables.

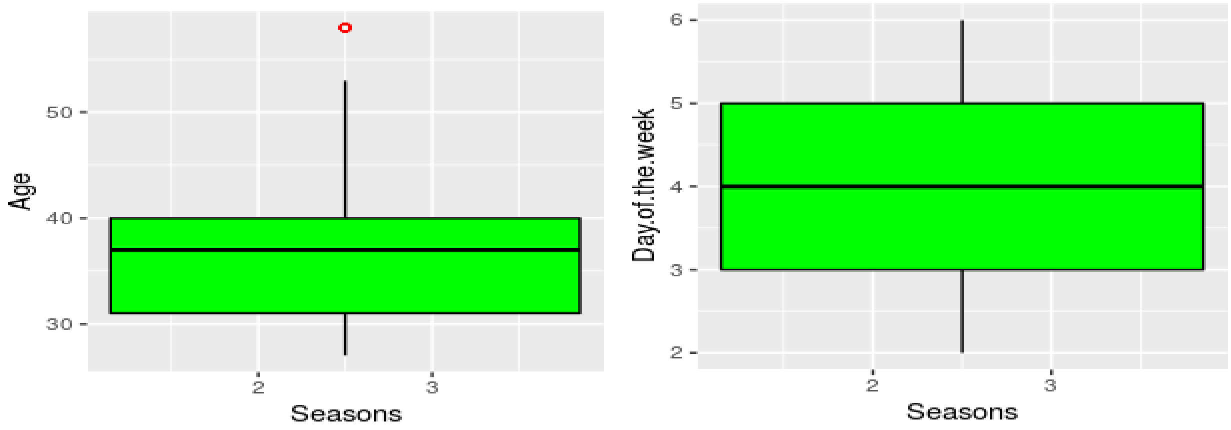


Figure 2.5 Box plot of Age and Day of week

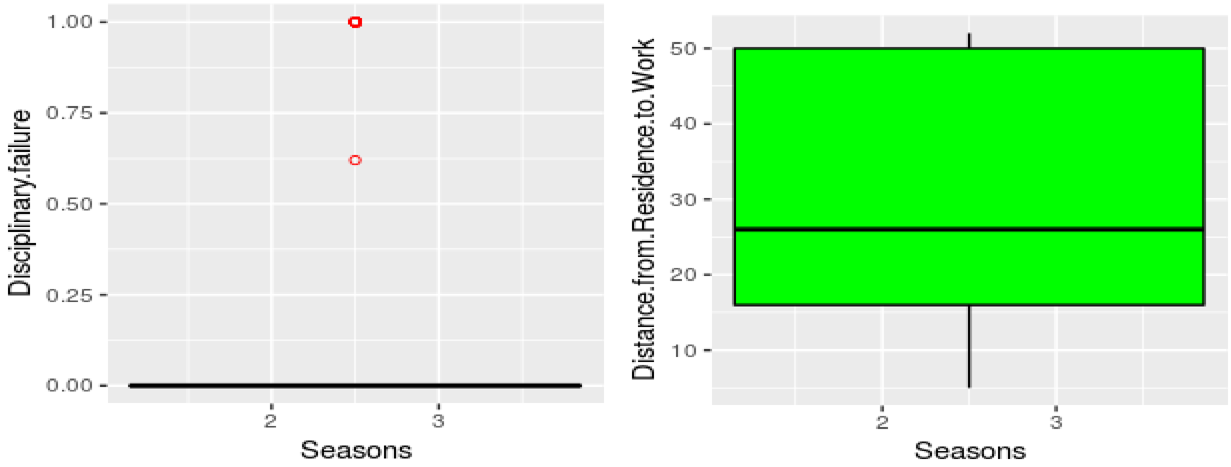


Figure 2.6 Box plot of Disciplinary Failure and Distance from Work

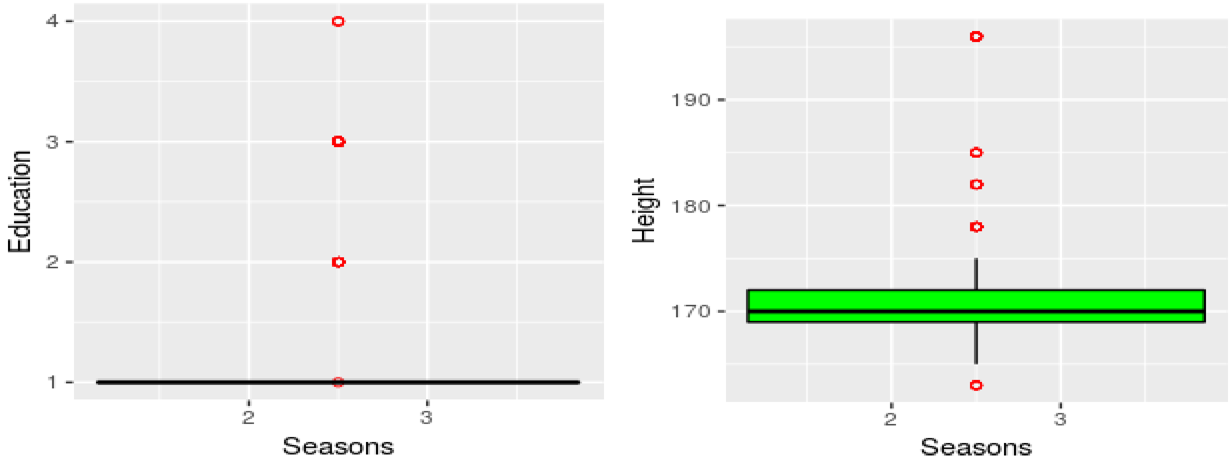


Figure 2.7 Box plot of Education and Height

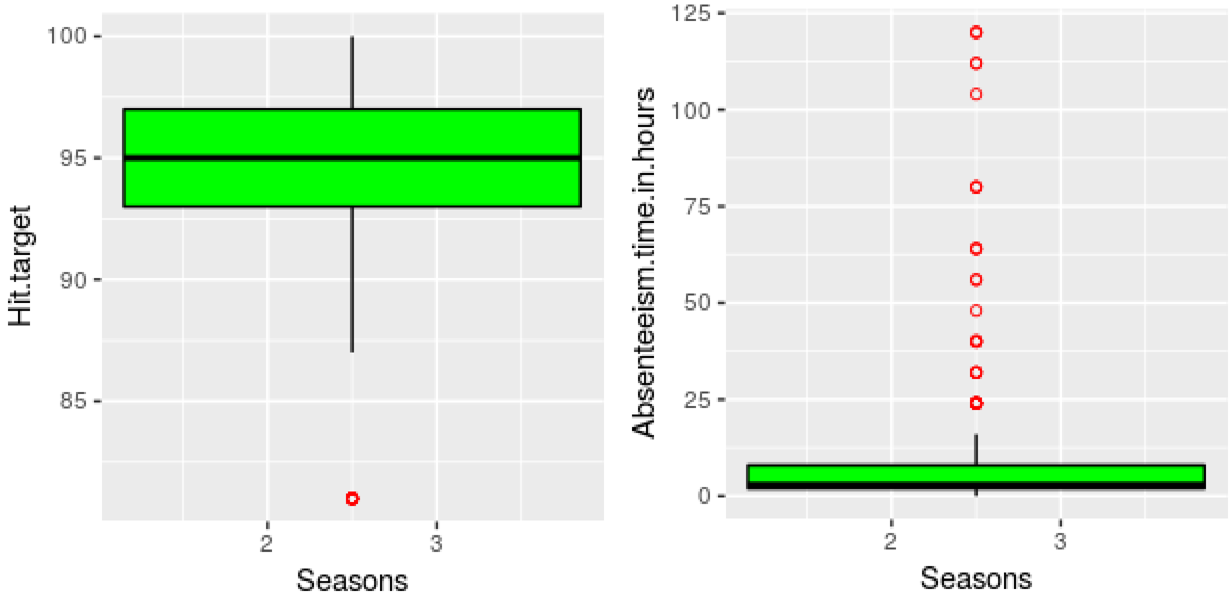


Figure 2.8 Box plot of Hit Target and absent hours

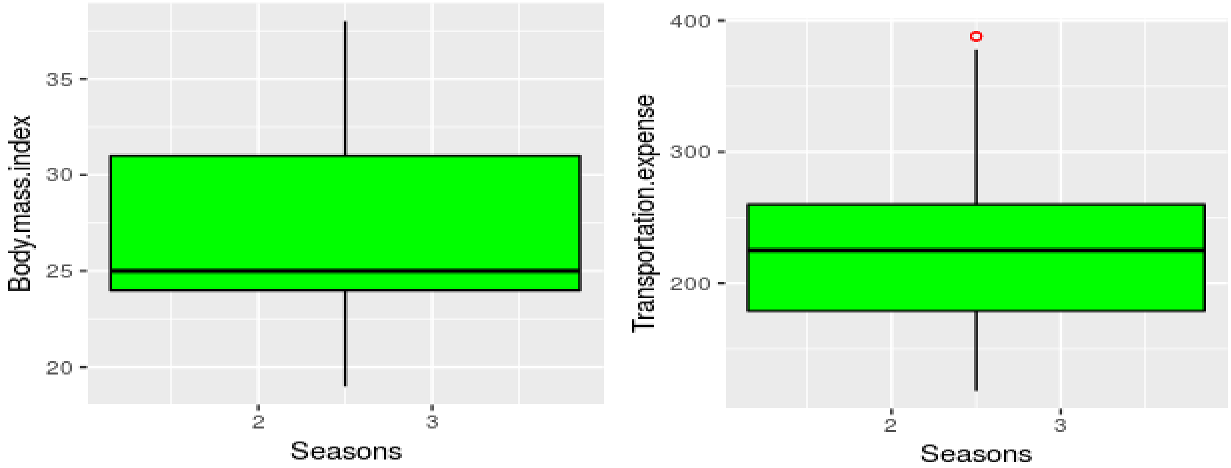


Figure 2.9 Box plot of BMI and Transportation Expense

Here all the boxplots of numeric variable, these out of range data can make model bias, so for further data analysis plotting box plot against various time periods.

Here are plots of predictors against various time periods.

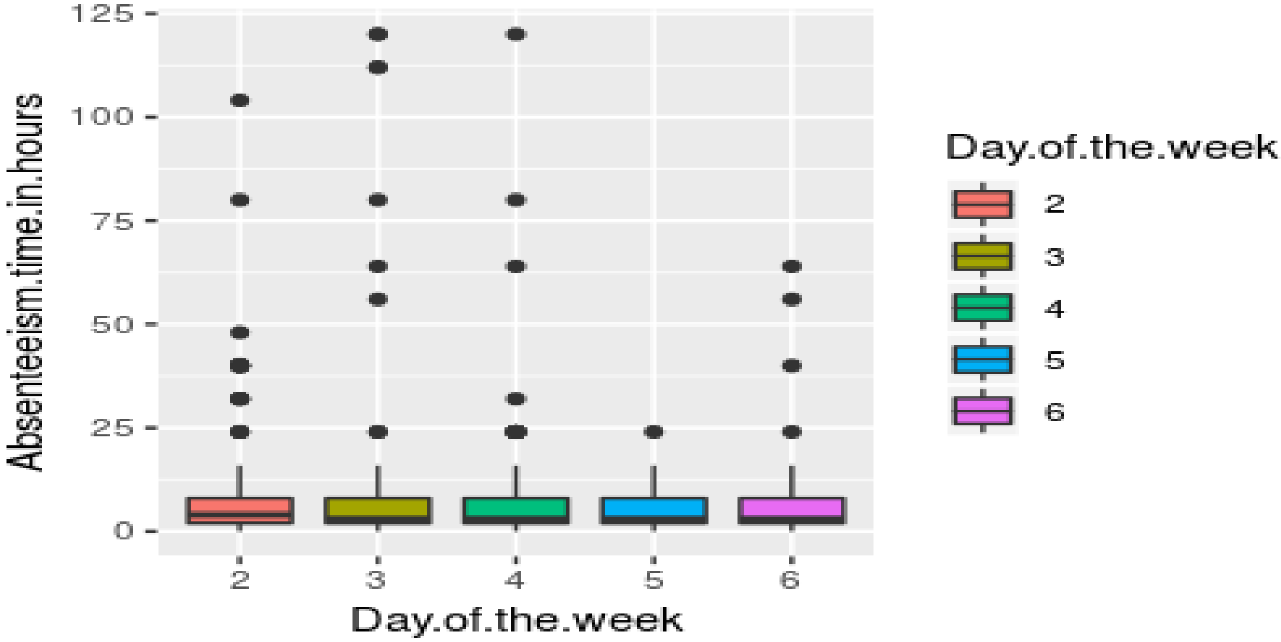


Figure 2.10 Box plots against Hours against day of week

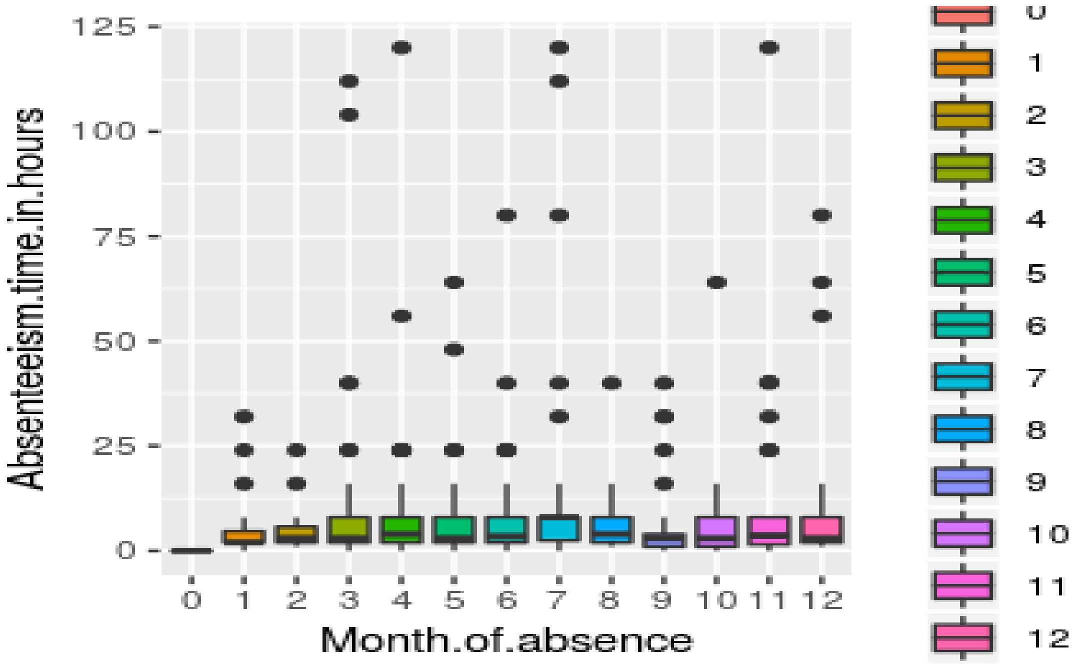


Figure 2.11 Box plots against Hours against Months

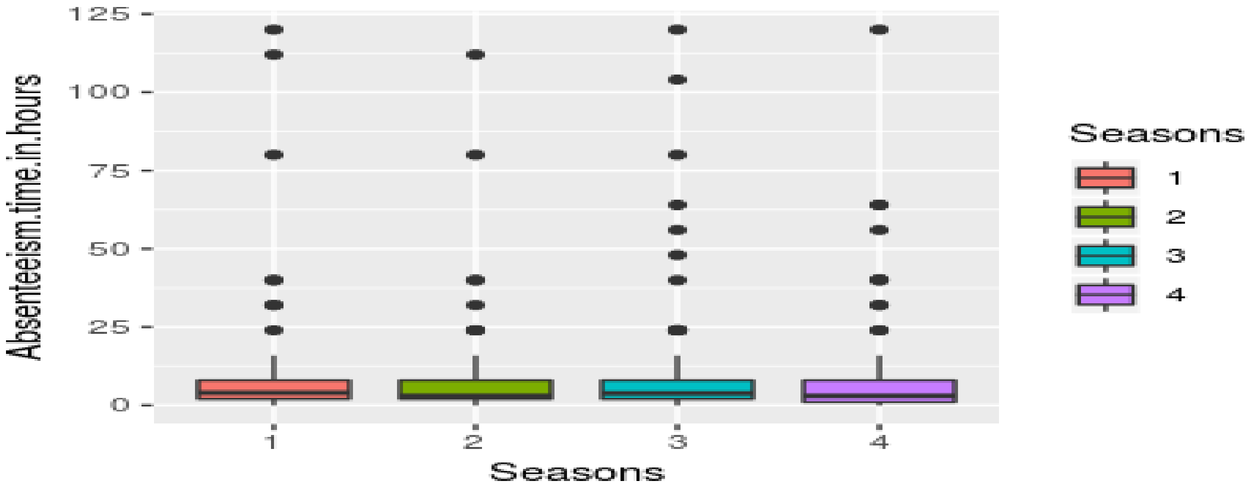


Figure 2.12 Box plots against Hours against Seasons

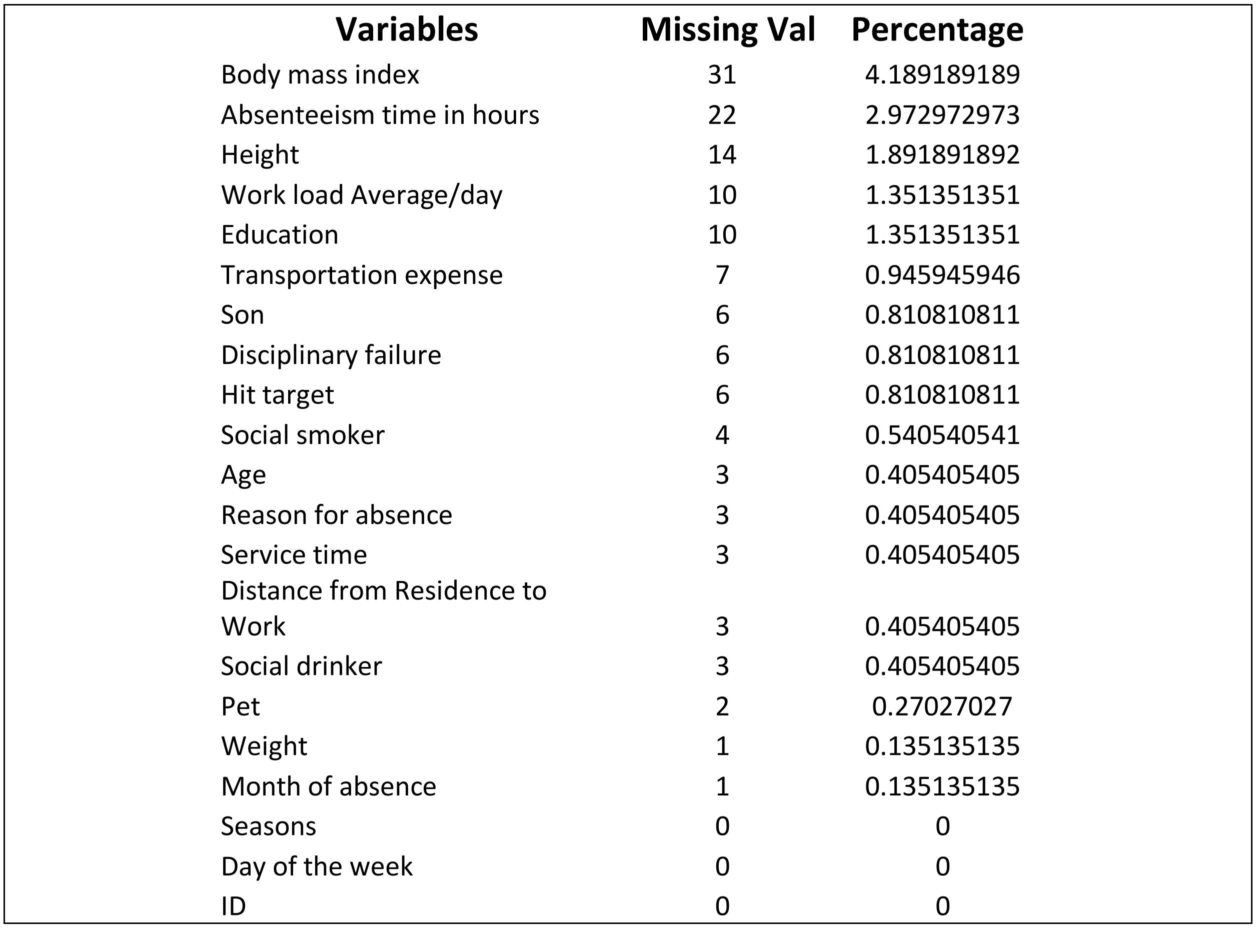
We can have different inferences from these plots, each plot depict facts of different levels. The season 3 having more outliers on upside, and in month 6,7,8 having less no. of out of range values, but it is clearly noted that we don't have any outliers on down side of box plots.

But for the same we go for day's analysis, we figured out, there is not much difference in hours with respect to days of week.

Missing Values

In this data set we have confronted with a problem of missing values, some variables have missing values, and mentioned with the help of table below.

Table 2.1: Missing Values (Columns: 9-16)



Variables	Missing Val	Percentage
Body mass index	31	4.189189189
Absenteeism time in hours	22	2.972972973
Height	14	1.891891892
Work load Average/day	10	1.351351351
Education	10	1.351351351
Transportation expense	7	0.945945946
Son	6	0.810810811
Disciplinary failure	6	0.810810811
Hit target	6	0.810810811
Social smoker	4	0.540540541
Age	3	0.405405405
Reason for absence	3	0.405405405
Service time	3	0.405405405
Distance from Residence to Work	3	0.405405405
Social drinker	3	0.405405405
Pet	2	0.27027027
Weight	1	0.135135135
Month of absence	1	0.135135135
Seasons	0	0
Day of the week	0	0
ID	0	0

In above table have the sum of missing values of various variables, and in another column having the percentage of missing values. Most of them less than 3%. And maximum percentage of missing 4% in **"Body Mass Index[3]"**

For this, we applied body mass index formula to calculate BMI from height and weight.

[3] In Body Mass Index, **Body Mass Index** is a simple calculation using a person's height and weight. The **formula** is **BMI** = kg/m^2 where kg is a person's weight in kilograms and m^2 is their height in meters squared.

2.1.3 Feature Selection

Before performing different models 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 regression. Feature selection become more important in this data set, because it have features of time series as well as regression. So we analyzed all the data again with various plots, we are plotting the correlation plot below.

We have 20 numeric variables but 5 of them are with levels.

- Disciplinary failure
- Education
- Son
- Social drinker
- Social smoker
- Pet

And Id is a unique value for each customer.

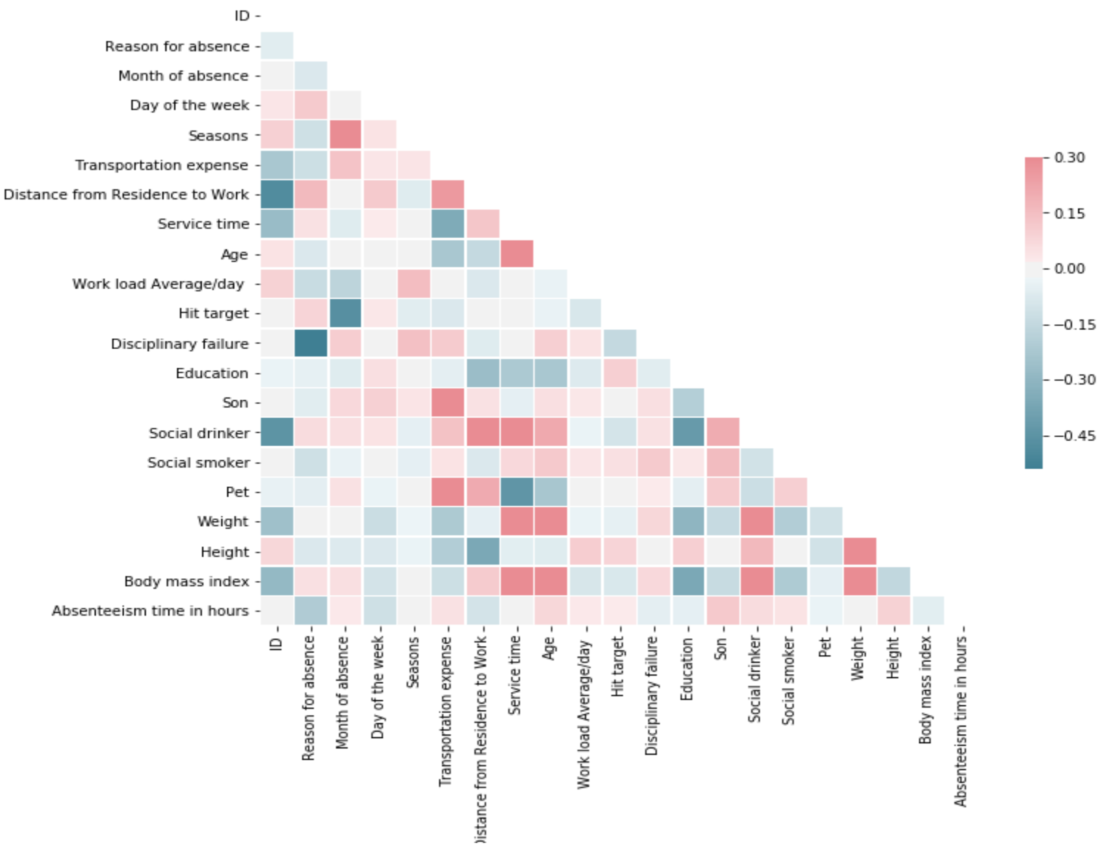
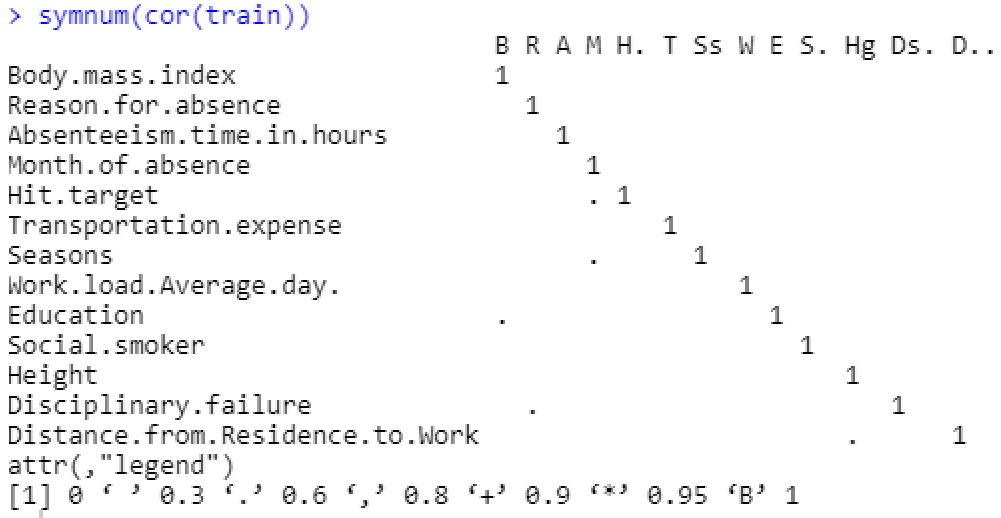


Figure 2.13 Correlation plot



```
> symnum(cor(train))
                                B R A M H. T Ss W E S. Hg Ds. D..
Body.mass.index                 1
Reason.for.absence                1
Absenteeism.time.in.hours           1
Month.of.absence                      1
Hit.target                            . 1
Transportation.expense                     1
Seasons                               .      1
Work.load.Average.day.                          1
Education                       .                 1
Social.smoker                                       1
Height                                                 1
Disciplinary.failure              .                       1
Distance.from.Residence.to.Work                        .      1
attr(,"legend")
[1] 0 ‘ ’ 0.3 ‘.’ 0.6 ‘,’ 0.8 ‘+’ 0.9 ‘*’ 0.95 ‘B’ 1
```

Figure 2.14 Summary of Correlation stats

Above is correlation plot of selected variable, and most them selected on the basis independence from each other. And we observed that some variables have more collinear to others.

For example BMI is correlated to height, weight and age, with the age BMI is increasing and this is also affecting the absent hours.

For further to see the effect of various variables on hours. We plotting more bar graphs for this.

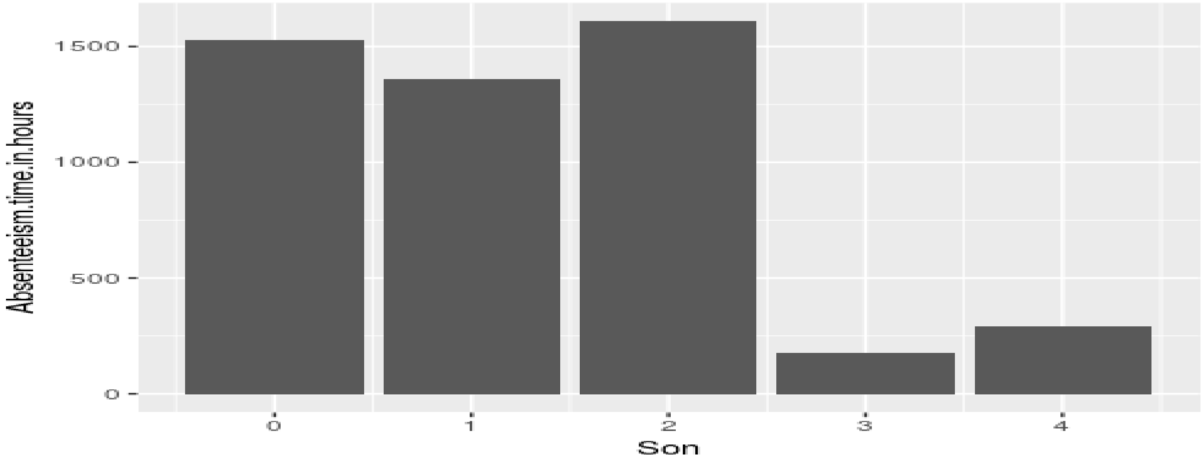


Figure 2.15 Bar graphs of hours against Son

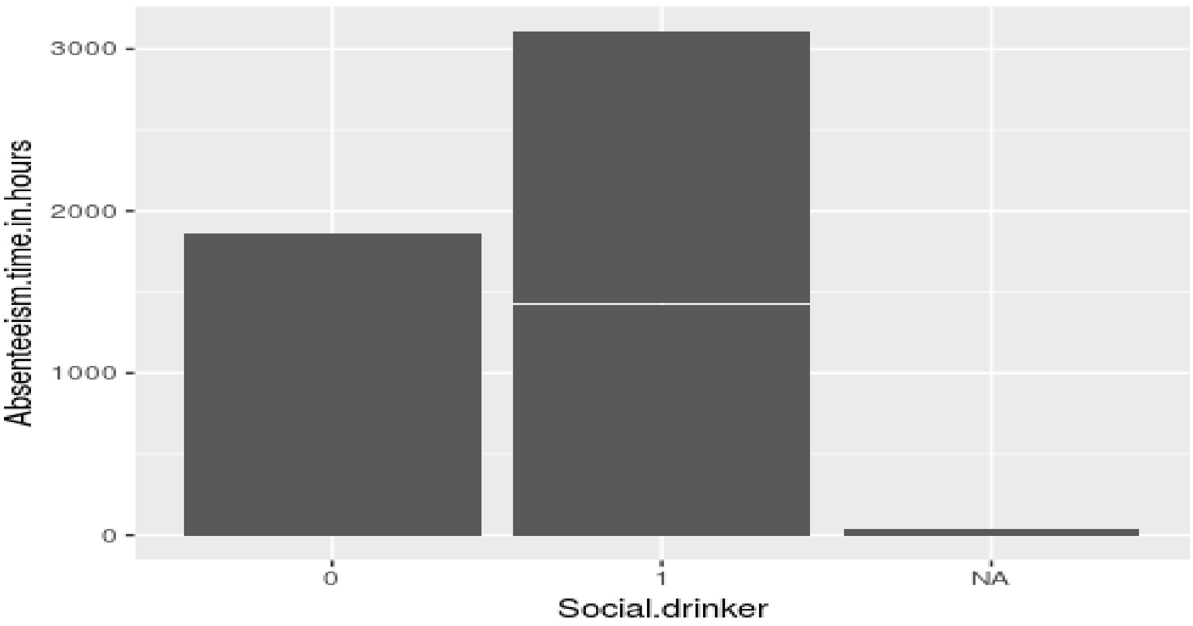


Figure 2.16 Bar graphs of hours against Social drinker

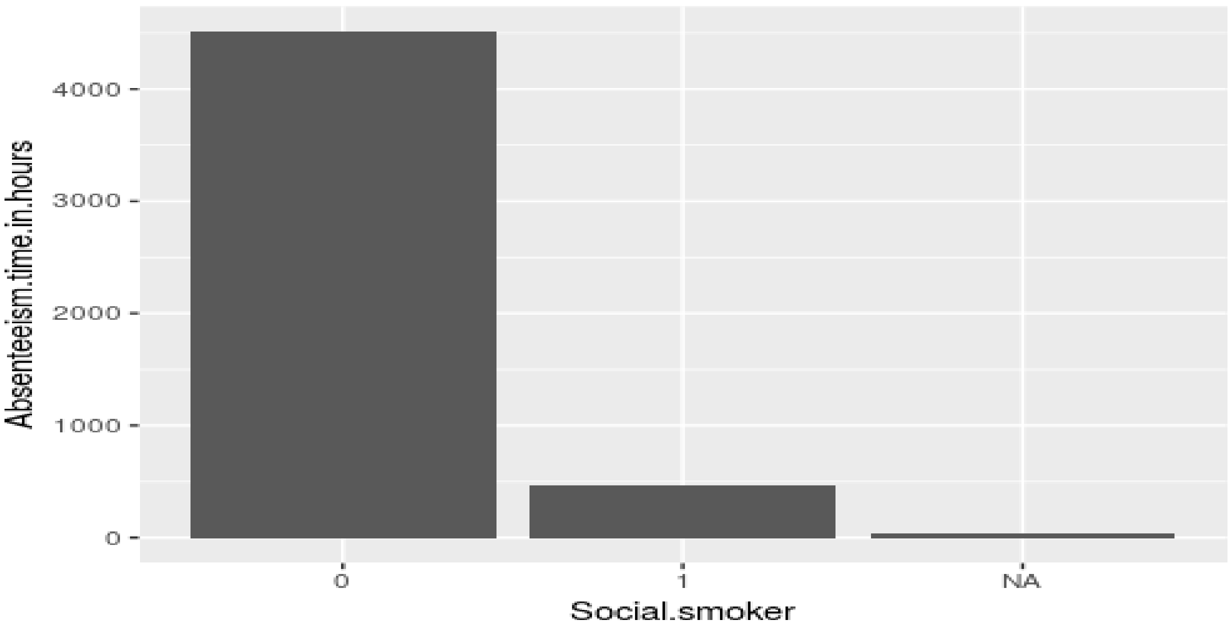


Figure 2.17 Bar graphs of hours against Social drinker

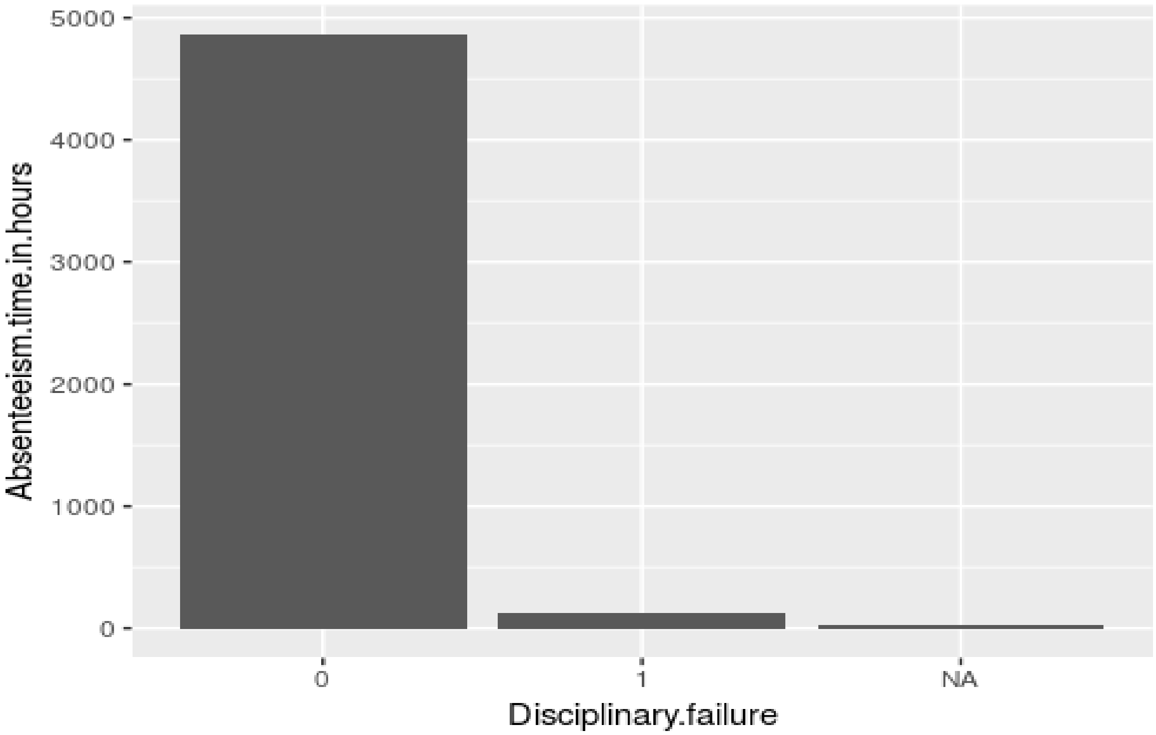


Figure 2.18 Bar graphs of hours against Disciplinary failure

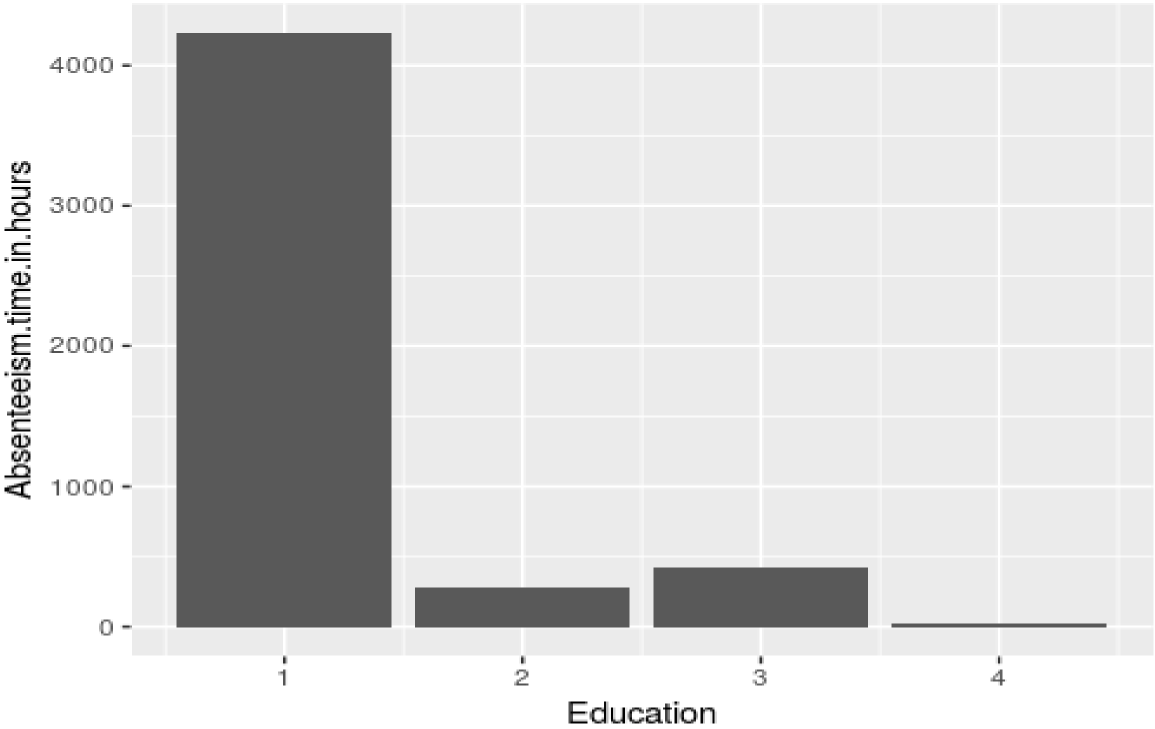


Figure 2.19 Bar graphs of hours against Education

Facts from figures

- We figured out from the figures that, social drinker and hours is regressively related, we can see in the sums having significant hike comparing to nondrinker.
- In next point the education is also affecting, the hours, the more educated level have less no. of absenteeism hours.
- In figure of Discipline, we have clear cut off that a non-discipline employee is getting more absents comparing discipline employee.
- In other figure is not depicting that much about the trend and cause of absent hours.

There are various methods we applied for feature selection.

2.2 Modeling

2.2.1 Model Selection

In our early stages of analysis during pre-processing we have come to understand that hours have trend as well as effect of other predictors so need to choose our model that both aspects in modelling. As we researched online that parcel industry have seasonality, for example have more delivery load in different season. And it also depend employees that dedicated and punctual.

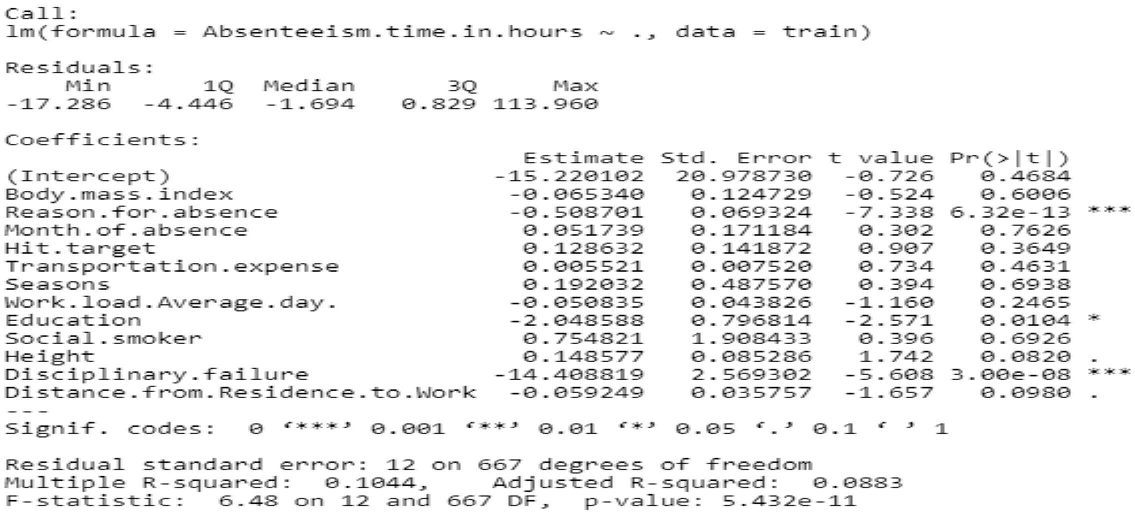
If the dependent variable, in our case *absent hours,* is Continuous value, so in predictive analysis that we can perform is **Regression**, and if the dependent variable is ordinal or levels the normal method is to do a **Classification** analysis, or classification after binning. But this needs to use linear regression, for specific and precise results we used time series model

There for we started to build a model with time series, then applied other methods also.

2.2.2 Linear Regression

In above figure, the summary stating all the variables of model. In this on the basis of previous analysis, we used all the variables for model which is filtered from correlation test and graph analysis.

Linear regression model summary is stated below.



```
Call:
lm(formula = Absenteeism.time.in.hours ~ ., data = train)

Residuals:
    Min      1Q  Median      3Q     Max 
-17.286  -4.446  -1.694   0.829 113.960 

Coefficients:
                                Estimate Std. Error t value Pr(>|t|)    
(Intercept)                   -15.220102  20.978730  -0.726   0.4684    
Body.mass.index                -0.065340   0.124729  -0.524   0.6006    
Reason.for.absence             -0.508701   0.069324  -7.338 6.32e-13 ***
Month.of.absence                0.051739   0.171184   0.302   0.7626    
Hit.target                      0.128632   0.141872   0.907   0.3649    
Transportation.expense          0.005521   0.007520   0.734   0.4631    
Seasons                         0.192032   0.487570   0.394   0.6938    
Work.load.Average.day.         -0.050835   0.043826  -1.160   0.2465    
Education                      -2.048588   0.796814  -2.571   0.0104 *  
Social.smoker                   0.754821   1.908433   0.396   0.6926    
Height                          0.148577   0.085286   1.742   0.0820 .  
Disciplinary.failure          -14.408819   2.569302  -5.608 3.00e-08 ***
Distance.from.Residence.to.Work  -0.059249   0.035757  -1.657   0.0980 .  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 12 on 667 degrees of freedom
Multiple R-squared:  0.1044,	Adjusted R-squared:  0.0883 
F-statistic:  6.48 on 12 and 667 DF,  p-value: 5.432e-11
```

Figure 2.20 Linear regression stats and coefficients

2.2.3 Time series model

Univariate analysis is the simplest form of analysing data. “Uni” means “one”, so in other words your data has only one variable. It doesn't deal with causes or relationships (unlike regression) and its major purpose is to describe; it takes data, summarizes that data and finds patterns in the data.

We used this method to see the other perspective of data set, before using this we have change the data to make data sets to put in the model, we changed data into two variables dates and data, and dates are in range of years. And after the analysis we got good results in respect of prediction. We also tried different order of model to get the best result. Actually ARIMA model is having 3 inputs for creating model. A seasonal ARIMA model is classified as an **ARIMA (p,d,q)x(P,D,Q)** model, where P=number of seasonal autoregressive (SAR) terms, D=number of seasonal differences, Q=number of seasonal moving average (SMA)

2.2.4 Regression

We could not use simple technique to predict the values of upcoming absent hours, we need a model which need to fit to the real time scenario, we have used that model but the errors are high, and can’t use this model. In this we used ARIMA model which change, it’s fitting according to seasonality. Because as we clearly explored that this data is having seasonality.

Chapter 3

Conclusion

3.1 Model Evaluation

Now that we have number of 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. Mean absolute error percentage
2. R squared

In our case of Absenteeism, The absent hours matter always having significant variable for parcel industry, if we evaluate rightly evaluate, we can work on employees, and can hire the people which have healthy life and education.

,

Predictive performance can be measured by comparing Predictions of the models with real values of the target variables, and calculating absolute error percentage of model.

3.1.1 Accuracy

Accuracy is defined as how much predicted value is deviated from the real values.

Below is the code for this

```
#testing data
absent <- predict(linearMod, test)  # predict absentism
actuals_preds <- data.frame(cbind(actuals=test$Absenteeism.time.in.hours,
predicteds=absent))  # make actuals_predicteds dataframe.
correlation_accuracy <- cor(actuals_preds)  # 82.7%
head(actuals_preds)
```

Figure 3.1 (<u>code in appendix)</u>

3.1.2 MAPE

MAPE is defined as square of the different between the average and actual value divided with actual value. So we can calculate with this the actual percentage error in our model

```
#percentage error
min_max_accuracy <- mean(apply(actuals_preds, 1, min) / apply(actuals_preds, 1,
max)) #42.7
mape <- sum(abs((actuals_preds$predicteds - actuals_preds$actuals))) # for it has
zero values
```

Figure 3.2 (code in appendix)

We calculated all the stats, but as we observed all the details, accuracy is approx. 42% but the values are too deviated on linear regression. So then we used ARIMA model for this.

3.2 Model Selection

We can see all three models perform comparatively on average and therefore we can select either of the models without any loss of information.

Solutions

1- We created model, company should think about life style and education of employees, as we observe from data the social drinker have much more probability to have absent hours. The education is also a sector company need to attention. These attributes should kept in mind in hiring of employees.

2- In the month of 2011 we have data to project, its seasonal data it will continue with some differences. As in **figure 2.3** we can take reverence from that figure to see what will be the predicted absent hours in 2011, but it can reduce, after taking the improvement steps.

.

Appendix A - Extra Figures

We potting some extra figures for better perspective of data

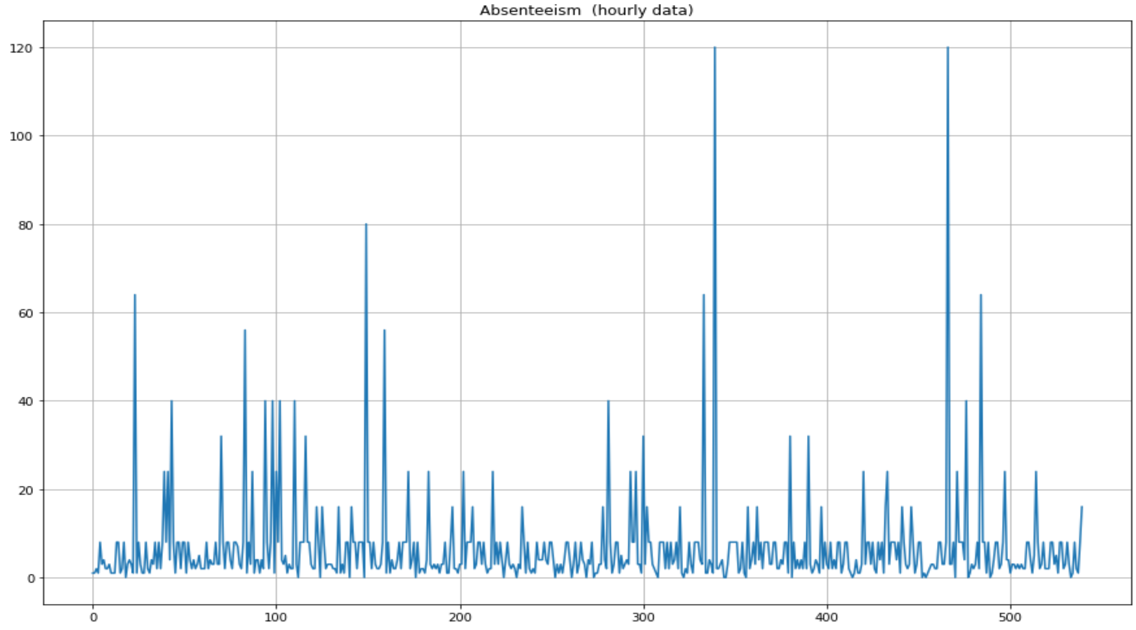


Figure 4.1 Plot of Absent hours

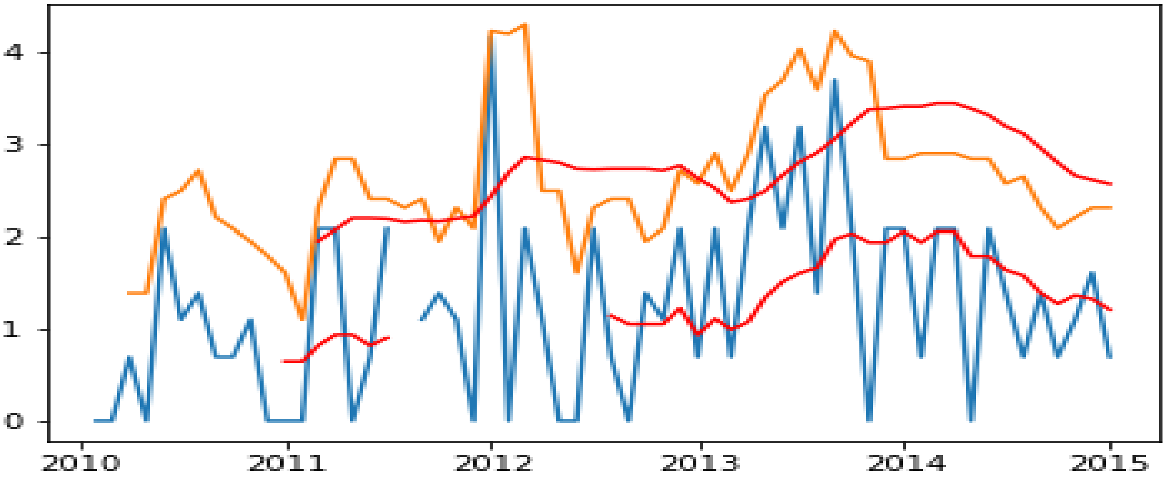


Figure 4.2 Plot after changing the data into Time series

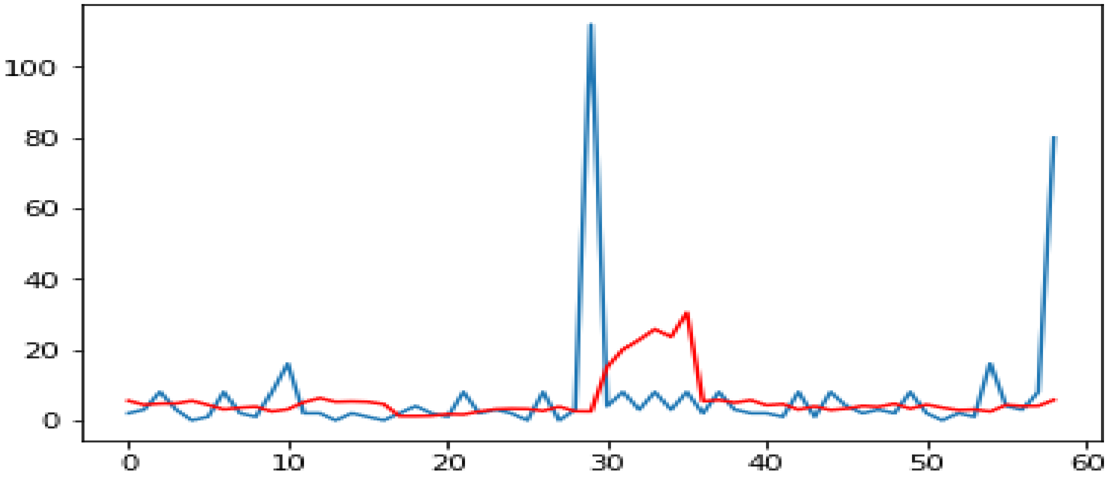


Figure 4.3 Plot of Actual vs Predicted data

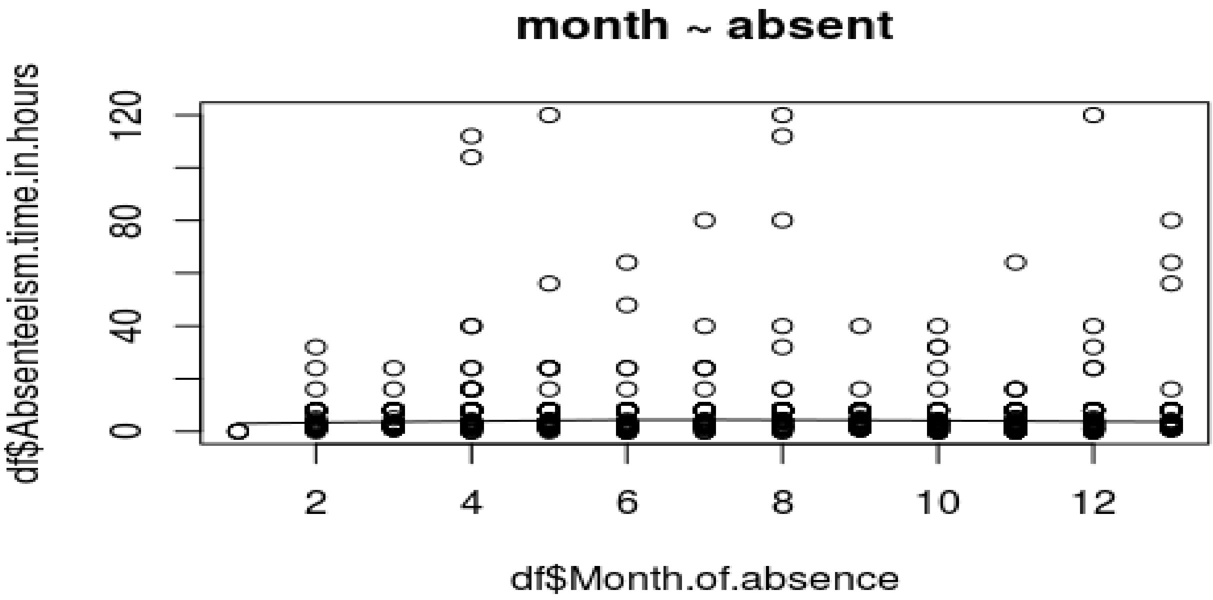


Figure 4.3 Plot of Actual vs Predicted data

Appendix B - R Code

```
#clearing RAM
rm(list=ls())
setwd("/cloud/project/Project-ed2") #seeting working directory
getwd() #getting working directory
install.packages(c("dplyr","DMwR","ggplot2"))
library("dplyr") #used for manipulationg select arrange
library("DMwR")  #used for msining with R
library("ggplot2") #used for plotting the graphs
require(gdata) #to read xl data
#reading file df
df = read.xls("Absent.xls", sheet = 1, header = TRUE)
#missing values
miss=data.frame(apply(df,2,function(x){sum(is.na(x))}))
miss$variables=row.names(miss)
colnames(miss)[1]="values"
miss=miss[,c(2,1)]
miss$percentage=((miss$value/dim(df)[1]))*100
row.names(miss)=NULL
#histogram of missing values in data set
hist(miss$values)
#replacing missing values with data
df=knnImputation(df,k=3)
df$Body.mass.index=ceiling(df$Weight/((df$Height/100)**2)) #replaced with actual formula
df$Month.of.absence=as.integer(df$Month.of.absence)
cname=colnames(df[,c(-1)])
```

```
#Box plot analysis

for (i in  1:length(cname))

{nam=paste0("box_",cname[i])

  assign(nam,ggplot(data=df, aes_string(x="Seasons",y=cname[i]), group="Seasons")+

  geom_boxplot(fill="green", color="black",outlier.color = "red",outlier.shape = 1))

}

 #box plot seasons

ggplot(df,aes(x=Month.of.absence,y=Absenteeism.time.in.hours,fill=Seasons))+geom_boxplot()

#box plot vs month

ggplot(df,aes(x=Seasons,y=Absenteeism.time.in.hours,fill=Seasons))+geom_boxplot()


#plotting sum of absent hours with respect to Season and month

bar_season=ggplot(df, aes(x=Seasons, y=Absenteeism.time.in.hours)) + geom_bar(stat="identity")

bar_month=ggplot(df, aes(x=Month.of.absence, y=Absenteeism.time.in.hours)) +
geom_bar(stat="identity")

bar_day=ggplot(df, aes(x=Day.of.the.week, y=Absenteeism.time.in.hours))+geom_bar(stat="identity")


#plotting with bin variables

bar_disciplinary=ggplot(df, aes(x=Disciplinary.failure, y=Absenteeism.time.in.hours)) +
geom_bar(stat="identity")

bar_education=ggplot(df, aes(x=Education, y=Absenteeism.time.in.hours)) + geom_bar(stat="identity")

bar_son=ggplot(df, aes(x=Son, y=Absenteeism.time.in.hours))+geom_bar(stat="identity")

bar_drinker=ggplot(df, aes(x=Social.smoker, y=Absenteeism.time.in.hours))+geom_bar(stat="identity")

bar_smoker=ggplot(df, aes(x=Social.drinker, y=Absenteeism.time.in.hours))+geom_bar(stat="identity")

#corrgram plot

corrgram(df[,c(-1)],order=TRUE,lower.panel= panel.ellipse,

        upper.panel = panel.shade, text.panel = panel.txt, main="correlation plot"  )
```

```
#correlation plot
numeric=sapply(df, is.numeric)
numeric=df[,numeric]
cor(numeric)
#scatter plot
scatter.smooth(x=df$Month.of.absence, y=df$Absenteeism.time.in.hours, main="month ~ absent")  # scatterplot
#applying linear regression
cor(cars$speed, cars$dist)
df_reducted=subset(df, select=c("Body.mass.index","Reason.for.absence","Absenteeism.time.in.hours","Month.of.absence","Hit.target","Transportation.expense","Seasons","Work.load.Average.day.","Education","Social.smoker","Height","Disciplinary.failure","Distance.from.Residence.to.Work"))
train=df_reducted[1:680,]
test=df_reducted[681:740,]
linearMod <- lm(Absenteeism.time.in.hours ~ ., data=train)  # build linear regression model on full data
print(linearMod)
summary(linearMod)
#testing data
absent <- predict(linearMod, test)  # predict absentism
actuals_preds <- data.frame(cbind(actuals=test$Absenteeism.time.in.hours, predicteds=absent))  # make actuals_predicteds dataframe.
correlation_accuracy <- cor(actuals_preds)  # 82.7%
head(actuals_preds)


#percentage error
min_max_accuracy <- mean(apply(actuals_preds, 1, min) / apply(actuals_preds, 1, max)) #42.7
mape <- sum(abs((actuals_preds$predicteds - actuals_preds$actuals))) # for it has zero values
symnum(cor(train))
```

Python Code

```
#importing all the libraries
import numpy as np
import pandas as pd
import math
from scipy.stats import chi2_contingency
import os
import seaborn as sns #for plotting
import matplotlib.pyplot as plt
%matplotlib inline
#setting working directory
os.getcwd()
df=pd.read_excel("Absent.xls")
#checking data sample
df.shape
df.head()
#missing data analysis
miss = pd.DataFrame(df.isnull().sum())
miss
miss=miss.reset_index() #old index added as column
miss=miss.rename(columns={"index":"variables",0:"miss_values"})
miss['percentage']=(miss['miss_values'] / len(df))*100
miss=miss.sort_values("miss_values",ascending=False).reset_index(drop=True)

#saving missing value as csv data
miss.to_csv("missing.csv",index=False)
df=df.fillna(df.median())
```

```
#plotting correlation

corr=df.corr()

summary(corr)

#plotting diagram

f, ax= plt.subplots(figsize=(11,9))

mask = np.zeros_like(corr, dtype=np.bool)

mask[np.triu_indices_from(mask)] = True

cmap = sns.diverging_palette(220, 10, as_cmap=True)

sns.heatmap(corr, mask=mask, cmap=cmap, vmax=.3, center=0,

            square=True, linewidths=.5, cbar_kws={"shrink": .5})



#loading some essential libraries

from dateutil.relativedelta import relativedelta # working with dates with style

from scipy.optimize import minimize          # for function minimization

import statsmodels.formula.api as smf         # statistics and econometrics

import statsmodels.tsa.api as smt

import statsmodels.api as sm

import scipy.stats as scs


#plotting absent in hours

plt.figure(figsize=(15, 10))

plt.plot(train["Absenteeism time in hours"])

plt.title('Absenteeism  (hourly data)')

plt.grid(True)

plt.show()

df=df.sort_values("Month of absence",ascending=True).reset_index(drop=True)
```

```
#manipulating the data frame to time series
df1=df.loc[df['Month of absence'] == 1]
df2=df.loc[df['Month of absence'] == 2]
df3=df.loc[df['Month of absence'] == 3]
df4=df.loc[df['Month of absence'] == 4]
df5=df.loc[df['Month of absence'] == 5]
df6=df.loc[df['Month of absence'] == 6]
df7=df.loc[df['Month of absence'] == 7]
df8=df.loc[df['Month of absence'] == 8]
df9=df.loc[df['Month of absence'] == 9]
df10=df.loc[df['Month of absence'] == 10]
df11=df.loc[df['Month of absence'] == 11]
df12=df.loc[df['Month of absence'] == 12]
#creating a new time series Data frame for this
len(df12)
new=df[0:1]
for i in range(0,49):
  j=i
  new=pd.concat([new,df1[j:j+1]])
  new=pd.concat([new,df2[j:j+1]])
  new=pd.concat([new,df3[j:j+1]])
  new=pd.concat([new,df4[j:j+1]])
  new=pd.concat([new,df5[j:j+1]])
  new=pd.concat([new,df6[j:j+1]])
  new=pd.concat([new,df7[j:j+1]])
  new=pd.concat([new,df8[j:j+1]])
  new=pd.concat([new,df9[j:j+1]])
  new=pd.concat([new,df10[j:j+1]])
  new=pd.concat([new,df11[j:j+1]])
  new=pd.concat([new,df12[j:j+1]])
```

```
#changing data frame to correct format
mod.to_csv('modi.csv', index=False)
mod=pd.read_csv("modi.csv")
mod= mod.drop(mod[mod.index==0].index)
mod=mod.drop('index', axis=1)
mod['Month of absence'] = mod['Month of absence'].astype(int)
#dividing the data in parts
train=mod[0:540]
test=mod[541:588]
#Date time library
from datetime import datetime
date_rng = pd.date_range(start='1/1/2010', end='1/1/2059', freq='M')
ts_uni = pd.DataFrame(date_rng, columns=['date'])
ts_uni["hours"]=mod['Absenteeism time in hours']
ts_uni.to_csv("timeseries.csv")
ts_uni['datetime'] = pd.to_datetime(ts_uni['date'])
ts_uni = ts_uni.set_index('datetime')
ts_uni.drop(['date'], axis=1, inplace=True)
ts_uni['rolling_sum'] = ts_uni.rolling(3).sum()
plt.plot(ts_uni[0:120])
#getting load to remove seasonality
ts_log=np.log(ts_uni[0:60])
moving_avg = ts_log.rolling(12).mean()
plt.plot(ts_log)
plt.plot(moving_avg, color='red')
#loading stats model to decompose the trend and season
from statsmodels.tsa.seasonal import seasonal_decompose
decomposition = seasonal_decompose(ts_uni[0:120])
trend = decomposition.trend
seasonal = decomposition.seasonal
```

```
#plotting subplots
plt.subplot(411)
plt.plot(ts_log, label='Original')
plt.legend(loc='best')
plt.subplot(412)
plt.plot(trend, label='Trend')
plt.legend(loc='best')
plt.subplot(413)
plt.plot(seasonal,label='Seasonality')
plt.legend(loc='best')
plt.subplot(414)
plt.plot(residual, label='Residuals')
plt.legend(loc='best')
plt.tight_layout()
#preparing the data for ARIMA model
X = ts_uni['hours'].values
size = int(len(X) * 0.9)
train, test = X[0:size], X[size:len(X)]
history = [x for x in train]
predictions = list()
for t in range(len(test)):
	model = ARIMA(history, order=(5,1,0))
	model_fit = model.fit(disp=0)
	output = model_fit.forecast()
	yhat = output[0]
	predictions.append(yhat)
	obs = test[t]
	history.append(obs)
```

```
#importing sklearn for testing errors

from sklearn.metrics import mean_squared_error

error = mean_squared_error(test, predictions)

print('Test MSE: %.3f' % error)

# plotting test vs prediction

pyplot.plot(test)

pyplot.plot(predictions, color='red')

pyplot.show()
```

References

Bibliography:

1- Hands on Machine Learning With Python Paperback – John Anderson (Author)

2- Fundamentals of Mathematical Statistics Paperback – SC Gupta (author)

Websites:

1- https://www.r-statistics.com

2- https://www.analyticsvidhya.com/blog/2015/12/complete-tutorial-time-series-modeling/