Employee Absenteeism

VINEET UNNITHAN

Contents

Chapter 1

Introduction

1.1 Problem Statement

Human capital plays a very vital role in the business of a courier company. It has the key roles in collection, transportation and delivery. As the XYZ company passes through the genuine issue of "Absenteeism". It wants the Machine Learning approach to exploit the available dataset to find out the root causes of the problem. It mainly wants to address the below two questions –

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?

The study aims to ft a machine learning method to predict the "Absenteeism hours" based on the various attributes of the provided dataset.

1.2 Data

Our aim is to be build a regression model which can be used to predict the Absenteeism hours for the company based on the various interpersonal as well as economic attributes. The details of the dataset can be seen below –

The attributes present in the dataset can be seen below –

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

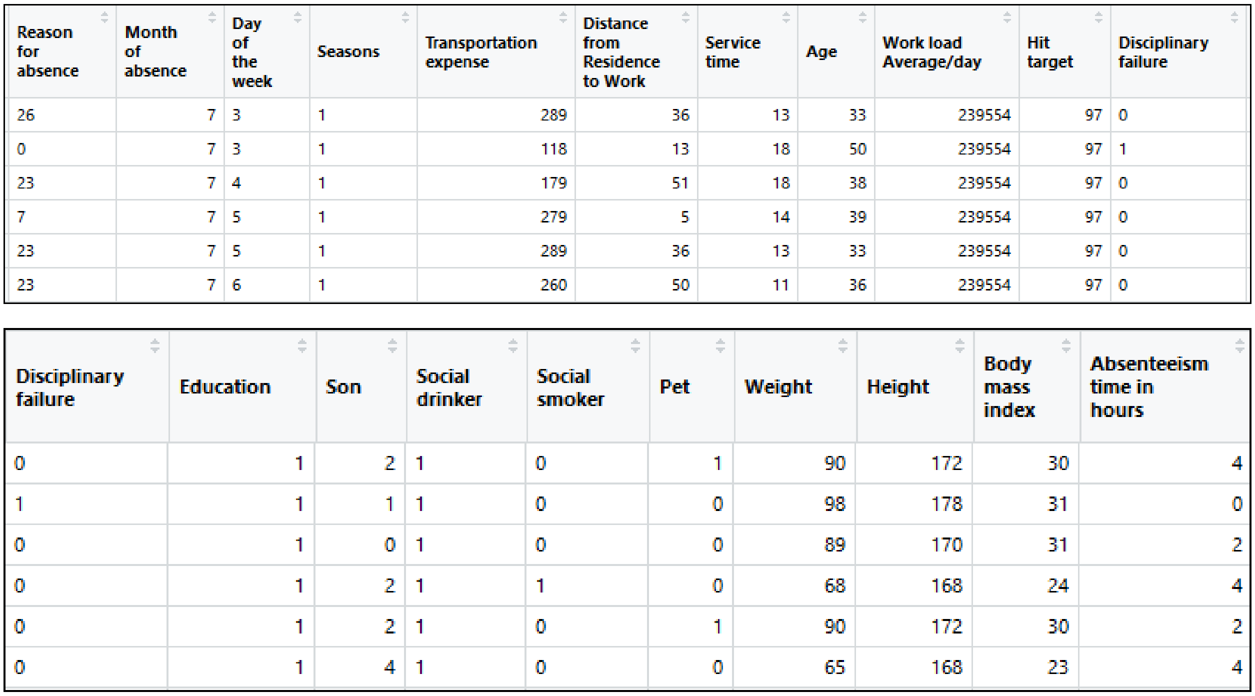
Absences attested by the International Code of Diseases (ICD) stratified into 21 categories (I to XXI) as follows:

I Certain infectious and parasitic diseases
II Neoplasms
III Diseases of the blood and blood-forming organs and certain disorders involving the immune mechanism
IV Endocrine, nutritional and metabolic diseases
V Mental and behavioural disorders
VI Diseases of the nervous system
VII Diseases of the eye and adnexa
VIII Diseases of the ear and mastoid process
IX Diseases of the circulatory system
X Diseases of the respiratory system
XI Diseases of the digestive system
XII Diseases of the skin and subcutaneous tissue
XIII Diseases of the musculoskeletal system and connective tissue
XIV Diseases of the genitourinary system
XV Pregnancy, childbirth and the puerperium
XVI Certain conditions originating in the perinatal period
XVII Congenital malformations, deformations and chromosomal abnormalities
XVIII Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified
XIX Injury, poisoning and certain other consequences of external causes
XX External causes of morbidity and mortality
XXI Factors influencing health status and contact with health services.

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

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

Table 1: Attribute of the dataset

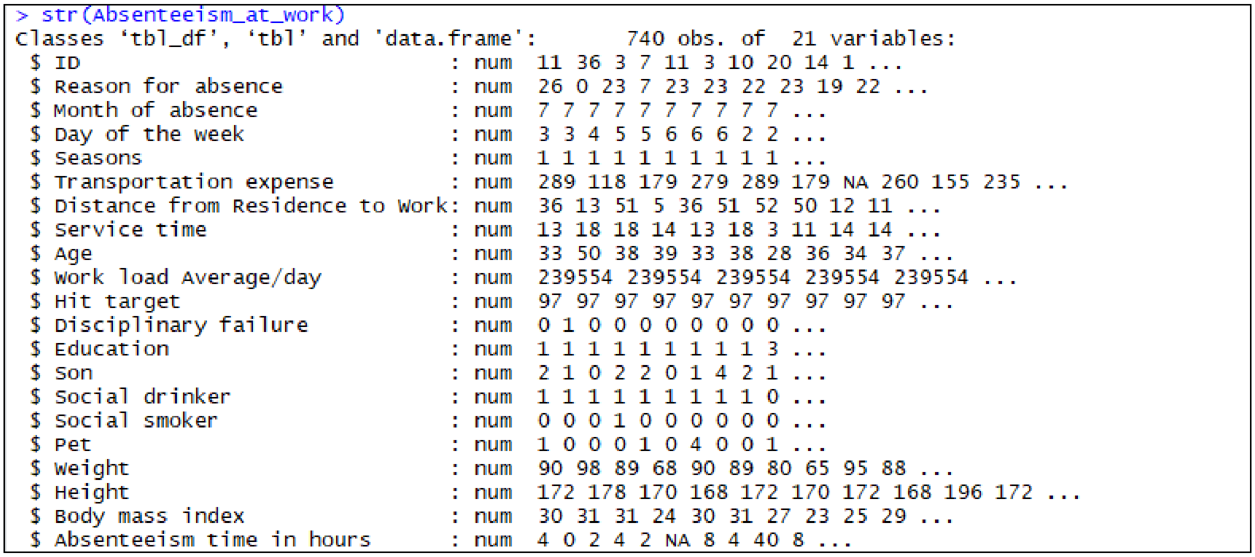


Reason for absence	Month of absence	Day of the week	Seasons	Transportation expense	Distance from Residence to Work	Service time	Age	Work load Average/day	Hit target	Disciplinary failure
26	7	3	1	289	36	13	33	239554	97	0
0	7	3	1	118	13	18	50	239554	97	1
23	7	4	1	179	51	18	38	239554	97	0
7	7	5	1	279	5	14	39	239554	97	0
23	7	5	1	289	36	13	33	239554	97	0
23	7	6	1	260	50	11	36	239554	97	0

Disciplinary failure	Education	Son	Social drinker	Social smoker	Pet	Weight	Height	Body mass index	Absenteeism time in hours
0	1	2	1	0	1	90	172	30	4
1	1	1	1	0	0	98	178	31	0
0	1	0	1	0	0	89	170	31	2
0	1	2	1	1	0	68	168	24	4
0	1	2	1	0	1	90	172	30	2
0	1	4	1	0	0	65	168	23	4

Figure 1: First six rows of the dataset

The shared dataset is first imported in the R environment and then the preprocessing and analysis has been carried out. Below is the structure of the default dataset imported into the R environment –



```
> str(Absenteeism_at_work)
Classes ‘tbl_df’, ‘tbl’ and 'data.frame':	740 obs. of  21 variables:
 $ ID                             : num  11 36 3 7 11 3 10 20 14 1 ...
 $ Reason for absence             : num  26 0 23 7 23 23 22 23 19 22 ...
 $ Month of absence               : num  7 7 7 7 7 7 7 7 7 7 ...
 $ Day of the week                : num  3 3 4 5 5 6 6 6 2 2 ...
 $ Seasons                        : num  1 1 1 1 1 1 1 1 1 1 ...
 $ Transportation expense         : num  289 118 179 279 289 179 NA 260 155 235 ...
 $ Distance from Residence to Work: num  36 13 51 5 36 51 52 50 12 11 ...
 $ Service time                   : num  13 18 18 14 13 18 3 11 14 14 ...
 $ Age                            : num  33 50 38 39 33 38 28 36 34 37 ...
 $ Work load Average/day          : num  239554 239554 239554 239554 239554 ...
 $ Hit target                     : num  97 97 97 97 97 97 97 97 97 97 ...
 $ Disciplinary failure           : num  0 1 0 0 0 0 0 0 0 0 ...
 $ Education                      : num  1 1 1 1 1 1 1 1 1 3 ...
 $ Son                            : num  2 1 0 2 2 0 1 4 2 1 ...
 $ Social drinker                 : num  1 1 1 1 1 1 1 1 1 0 ...
 $ Social smoker                  : num  0 0 0 1 0 0 0 0 0 0 ...
 $ Pet                            : num  1 0 0 0 1 0 4 0 0 1 ...
 $ Weight                         : num  90 98 89 68 90 89 80 65 95 88 ...
 $ Height                         : num  172 178 170 168 172 170 172 168 196 172 ...
 $ Body mass index                : num  30 31 31 24 30 31 27 23 25 29 ...
 $ Absenteeism time in hours      : num  4 0 2 4 2 NA 8 4 40 8 ...
```

Figure 2: Structure of the default dataset imported into the R environment

Chapter 2

Methodology

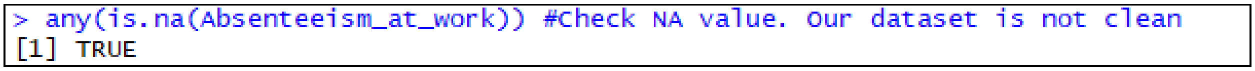
2.1 Data Preprocessing

For the data analytics and the Machine Learning, we should always start with the cleaning of the data first. As the given data to us is known and we have seen the structure and found that all the attributes are by default taken as the integer or float. But as per the description of the attributes we can see that not all the attributes are numeric, rather many of them are categorical. Hence, we need to perform the required transformation.

Apart from the required transformation, we also need to perform the "NA" check (missing value check). As we don't know how clean the data is, hence we need to check the missing value. This is also particularly needed, as not all the ML algorithms are robust with the missing values. Hence, many of the algorithms will not converge in case of the missing values, or even can cause the misleading results. Next section is dedicated for the analysis of the missing value and the distribution of the data.

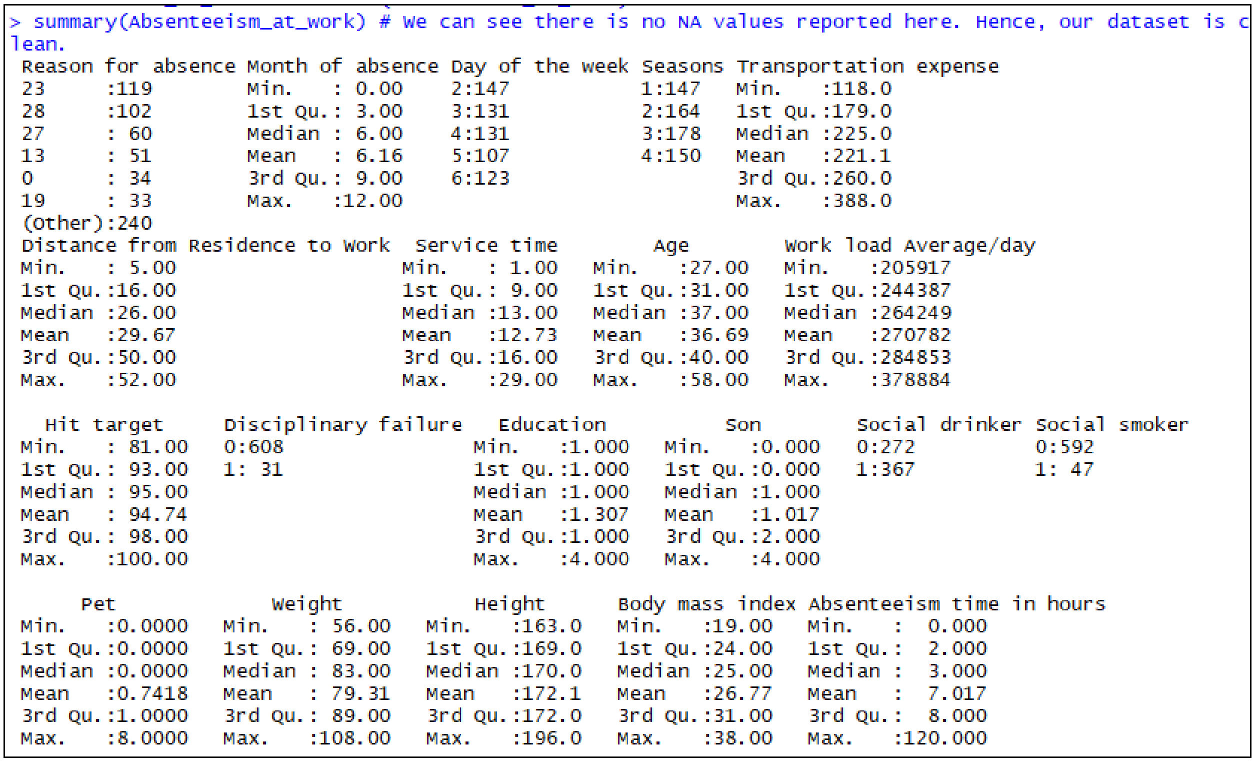
2.1.1 Data Distribution and Missing Value analysis

First thing which is needed to start the analysis is to check the missing values. In the given dataset, there are small amount of missing data. The detail can be seen below –



```
> any(is.na(Absenteeism_at_work)) #Check NA value. Our dataset is not clean
[1] TRUE
```

Hence, to treat the missing value, it decided to remove the instances which have missing values. We can also substitute it using the mean values, but it doesn't make sense as many of the attributes are categorical. Also, only 31 instances are needed to be removed in order to make the dataset "NA" value free. Below is the code used for the same. After the treatment of the NA values, the descriptive statistics has been performed using the summary() function. It also returns the presence of missing values in each instance.



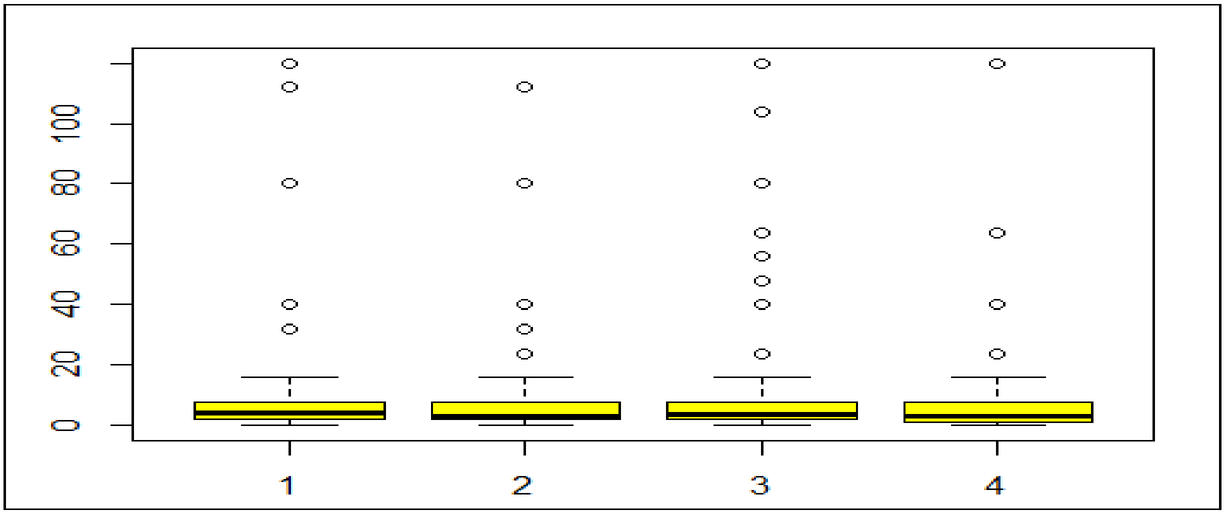
```
> summary(Absenteeism_at_work) # We can see there is no NA values reported here. Hence, our dataset is c
lean.
 Reason for absence Month of absence Day of the week Seasons Transportation expense
 23     :119        Min.   : 0.00    2:147           1:147   Min.   :118.0         
 28     :102        1st Qu.: 3.00    3:131           2:164   1st Qu.:179.0         
 27     : 60        Median : 6.00    4:131           3:178   Median :225.0         
 13     : 51        Mean   : 6.16    5:107           4:150   Mean   :221.1         
 0      : 34        3rd Qu.: 9.00    6:123                   3rd Qu.:260.0         
 19     : 33        Max.   :12.00                            Max.   :388.0         
 (Other):240                                                                       
 Distance from Residence to Work  Service time        Age        Work load Average/day 
 Min.   : 5.00                   Min.   : 1.00   Min.   :27.00   Min.   :205917        
 1st Qu.:16.00                   1st Qu.: 9.00   1st Qu.:31.00   1st Qu.:244387        
 Median :26.00                   Median :13.00   Median :37.00   Median :264249        
 Mean   :29.67                   Mean   :12.73   Mean   :36.69   Mean   :270782        
 3rd Qu.:50.00                   3rd Qu.:16.00   3rd Qu.:40.00   3rd Qu.:284853        
 Max.   :52.00                   Max.   :29.00   Max.   :58.00   Max.   :378884        
                                                                                       
   Hit target     Disciplinary failure   Education          Son        Social drinker Social smoker
 Min.   : 81.00   0:608                Min.   :1.000   Min.   :0.000   0:272          0:592        
 1st Qu.: 93.00   1: 31                1st Qu.:1.000   1st Qu.:0.000   1:367          1: 47        
 Median : 95.00                        Median :1.000   Median :1.000                               
 Mean   : 94.74                        Mean   :1.307   Mean   :1.017                               
 3rd Qu.: 98.00                        3rd Qu.:1.000   3rd Qu.:2.000                               
 Max.   :100.00                        Max.   :4.000   Max.   :4.000                               
                                                                                                   
      Pet             Weight           Height      Body mass index Absenteeism time in hours
 Min.   :0.0000   Min.   : 56.00   Min.   :163.0   Min.   :19.00   Min.   :  0.000          
 1st Qu.:0.0000   1st Qu.: 69.00   1st Qu.:169.0   1st Qu.:24.00   1st Qu.:  2.000          
 Median :0.0000   Median : 83.00   Median :170.0   Median :25.00   Median :  3.000          
 Mean   :0.7418   Mean   : 79.31   Mean   :172.1   Mean   :26.77   Mean   :  7.017          
 3rd Qu.:1.0000   3rd Qu.: 89.00   3rd Qu.:172.0   3rd Qu.:31.00   3rd Qu.:  8.000          
 Max.   :8.0000   Max.   :108.00   Max.   :196.0   Max.   :38.00   Max.   :120.000          
```

Figure 3: Descriptive statistics

After treatment of the data, it’s necessary to dig down the details of the dataset. For this purpose, first step is to check the distribution of all the attributes. Below are the boxplots and histograms used for the graphical analysis of the dataset.

BOXPLOTS

The below boxplot is used to see the distribution of the Absenteeism hours in the various seasons as per the given dataset –



The plot shows clearly, that the average of the first season is more in compare to that of other seasons. In summer more, people are more absent in compare to other seasons and there can be many reasons for this.

The density distribution of the data can be seen in the below histograms.

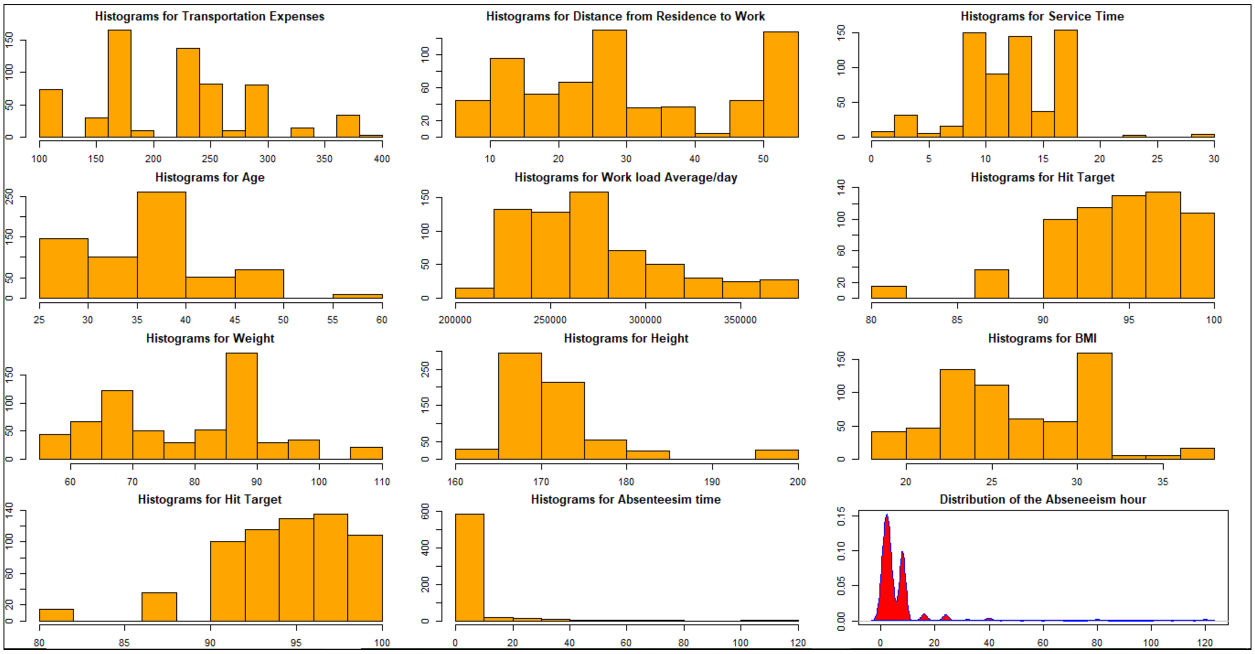


Figure 4: Histograms and density plot for all the attributes

We can see that most of the attributes are not so properly normally distributed. Also, for this study we are not primarily using the simple linear regression, where normality of the distribution is very important. The reason is, our dataset is having both numerical as well as the categorical attributes.

As we are planning to build the decision tree model, we have decided to use all the attributes of the dataset.

2.2 Modelling

For modelling, we have decided to use the C&RT method (Classification and Regression Tree Method). These methods are very well suited for the supervised learning task which has both categorical and the numerical attributes.

Looking at the benefits of such model the study aims to exploits the given dataset using the C&RT method. The method is capable of deciding which factor is most responsible for the outcome and classifies that based on the tree taxonomy.

Before digging the dataset with the novel ML methods, we need to dig it little more using the graphical tools to see the relationship between the different attributes. Below are the some of the attribute relations, which shows how all the attributes are interlinked, which is also a motivation behind using all the attributes for this analysis.

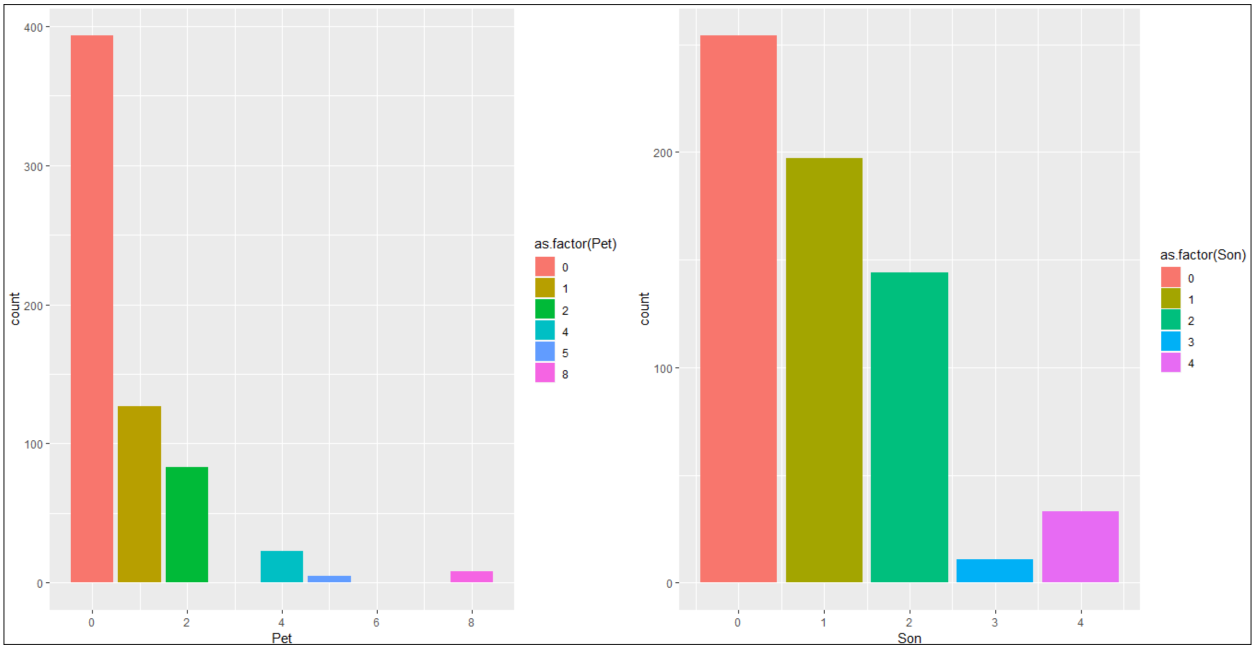


Figure 5: The distribution of the Absent hours between the people who have pets and who have son

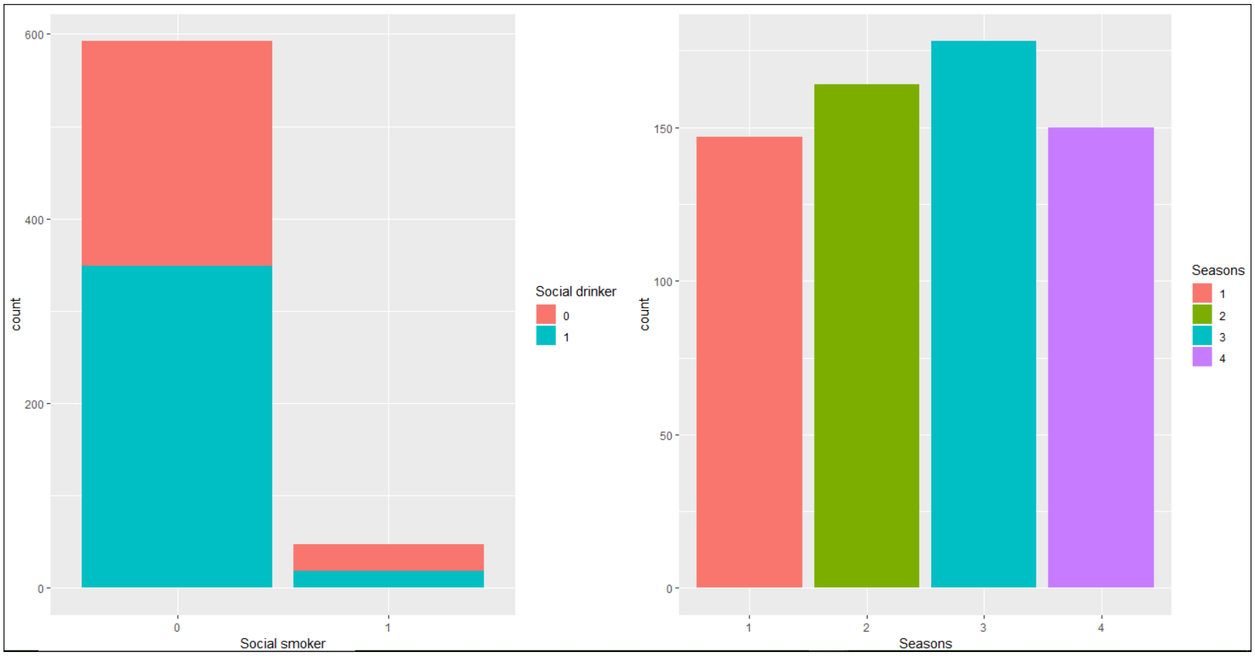


Figure 6: Distribution of the Absent hours between the social drinkers and social smokers and the seasons of the year

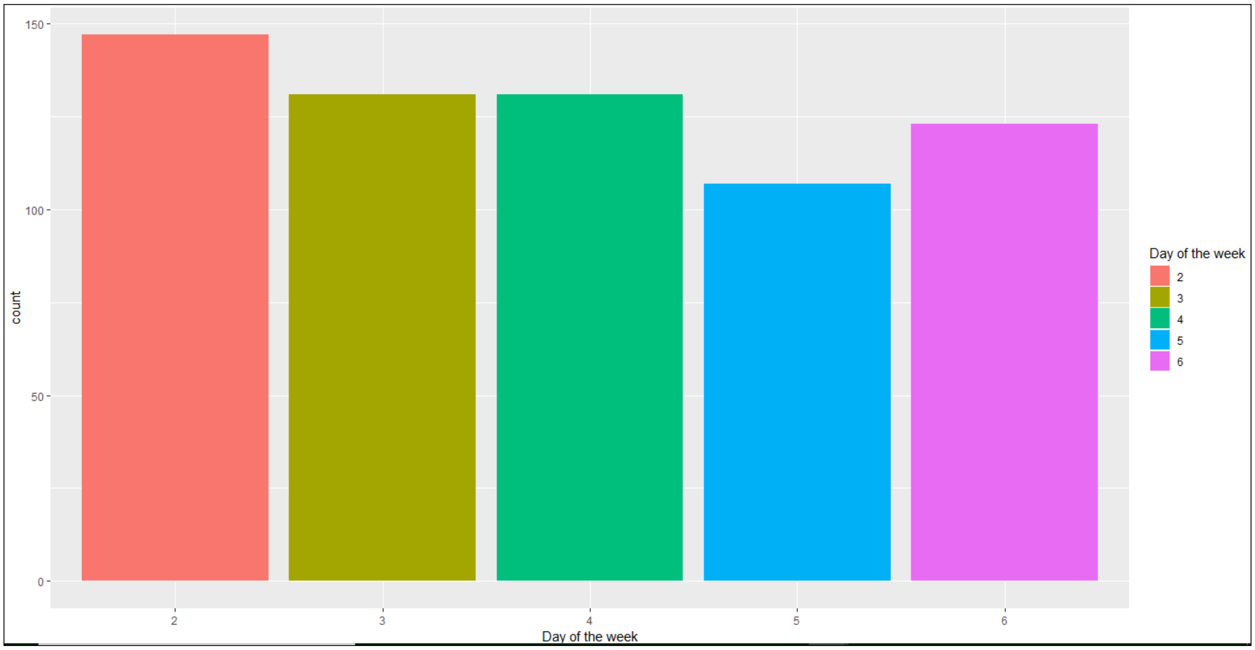
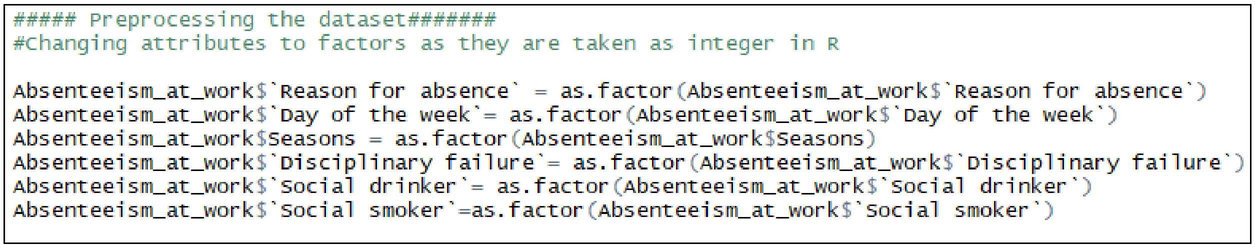


Figure 7: Distribution of the absent hours during the week days

Since, our R environment has read the dataset by default as the numeric, we need to change the categorical attributes, into the factors. Below are the codes which is used to change the six attributes to the factors.



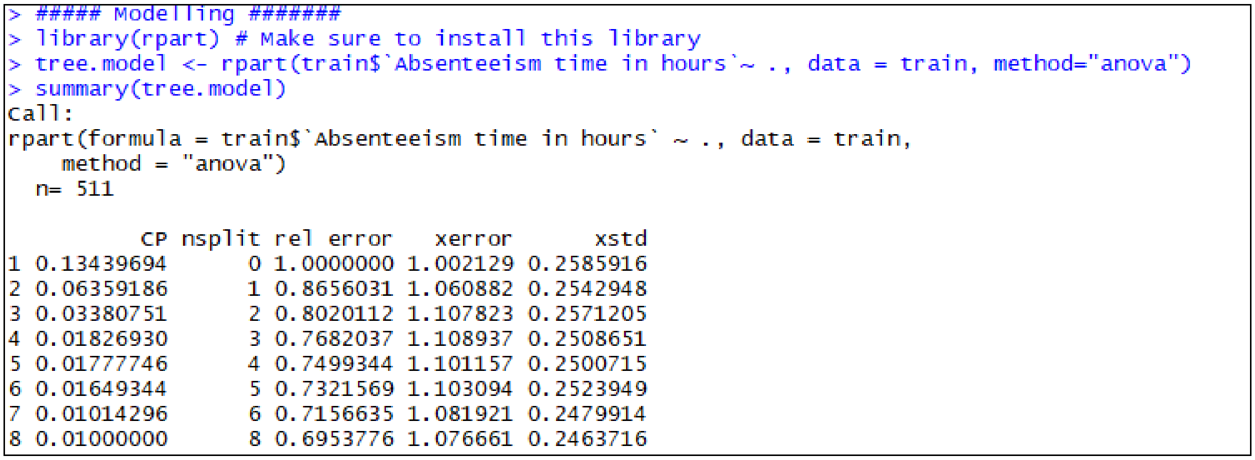
```
##### Preprocessing the dataset#######
#Changing attributes to factors as they are taken as integer in R

Absenteeism_at_work$`Reason for absence` = as.factor(Absenteeism_at_work$`Reason for absence`)
Absenteeism_at_work$`Day of the week`= as.factor(Absenteeism_at_work$`Day of the week`)
Absenteeism_at_work$Seasons = as.factor(Absenteeism_at_work$Seasons)
Absenteeism_at_work$`Disciplinary failure`= as.factor(Absenteeism_at_work$`Disciplinary failure`)
Absenteeism_at_work$`Social drinker`= as.factor(Absenteeism_at_work$`Social drinker`)
Absenteeism_at_work$`Social smoker`=as.factor(Absenteeism_at_work$`Social smoker`)
```

Figure 8: Changing the categorical attributes to factors

2.2.1 Decision Regression Tree

The decision regression tree has been used for the modelling purpose We used the 80% of the data as the training data and the rest 20% of the data as the test data The RMS (Root mean square value) has been used to check the quality of the fitted model.

Below is the fitted tree and the R codes used to obtain the model –



```
> ##### Modelling #######
> library(rpart) # Make sure to install this library
> tree.model <- rpart(train$`Absenteeism time in hours`~ ., data = train, method="anova")
> summary(tree.model)
Call:
rpart(formula = train$`Absenteeism time in hours` ~ ., data = train,
    method = "anova")
  n= 511

          CP nsplit rel error   xerror      xstd
1 0.13439694      0 1.0000000 1.002129 0.2585916
2 0.06359186      1 0.8656031 1.060882 0.2542948
3 0.03380751      2 0.8020112 1.107823 0.2571205
4 0.01826930      3 0.7682037 1.108937 0.2508651
5 0.01777746      4 0.7499344 1.101157 0.2500715
6 0.01649344      5 0.7321569 1.103094 0.2523949
7 0.01014296      6 0.7156635 1.081921 0.2479914
8 0.01000000      8 0.6953776 1.076661 0.2463716
```

Figure 9: Fitted C&RT model

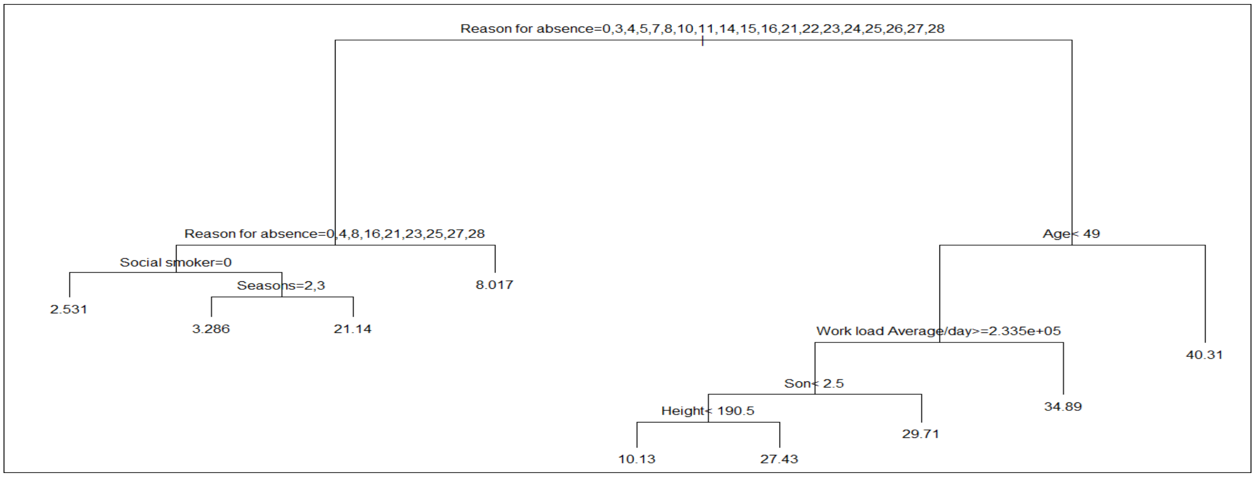
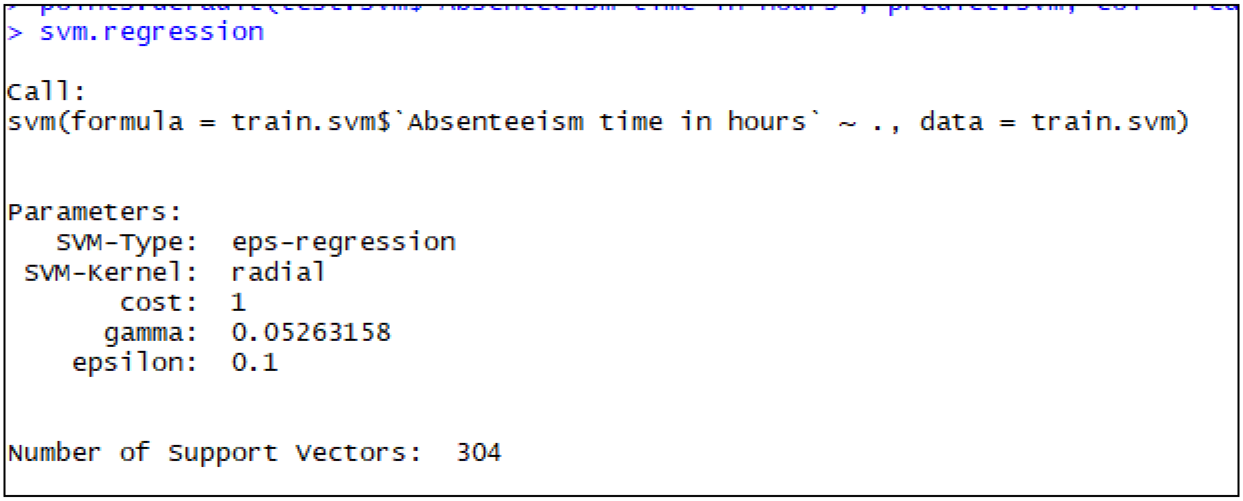

Reason for absence=0,3,4,5,7,8,10,11,14,15,16,21,22,23,24,25,26,27,28
Reason for absence=0,4,8,16,21,23,25,27,28
Social smoker=0
Seasons=2,3
2.531
3.286
21.14
8.017
Age< 49
Work load Average/day>=2.335e+05
Son< 2.5
Height< 190.5
10.13
27.43
29.71
34.89
40.31

Figure 10: The fitted decision tree. We can see the split reasons and the thresholds.

2.2.3 Support Vector Machine Regression

For this study we also tried the SVM regression and the results for the model can be seen below-



```
> svm.regression

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


Parameters:
   SVM-Type:  eps-regression
 SVM-Kernel:  radial
       cost:  1
      gamma:  0.05263158
    epsilon:  0.1


Number of Support Vectors:  304
```

Figure 11: SVM Regression fitted

Chapter 3

Conclusion

3.1 Results and Discussion

Model selection is the method of selecting the best performing model. As we used the C&RT model with all the default parameters and all the attributes. The model selection will primarily will be decided on the basis of –

1- Computational Efficiency
2- Interpretability
3- Predictive Performance

As our given dataset was very small, hence, computational efficiency was not an issue associated with this study. Also, the decision tree methods are highly optimized methods, which does not require lots of computational resources. Our model was fitted in just a matter of milliseconds. Hence, this parameter is of least importance in this study.

Interpretability is a highly required features for this study. We can see that the decision tree obtained above is highly interpretable. Also, the layout is very easy to understand. And hence, it makes it easy to understand.

Decision Tree

The predictive performance of the model can be seen with the RMS value and the residual plots. Below we can see the RMS value and the residual plots of the model.

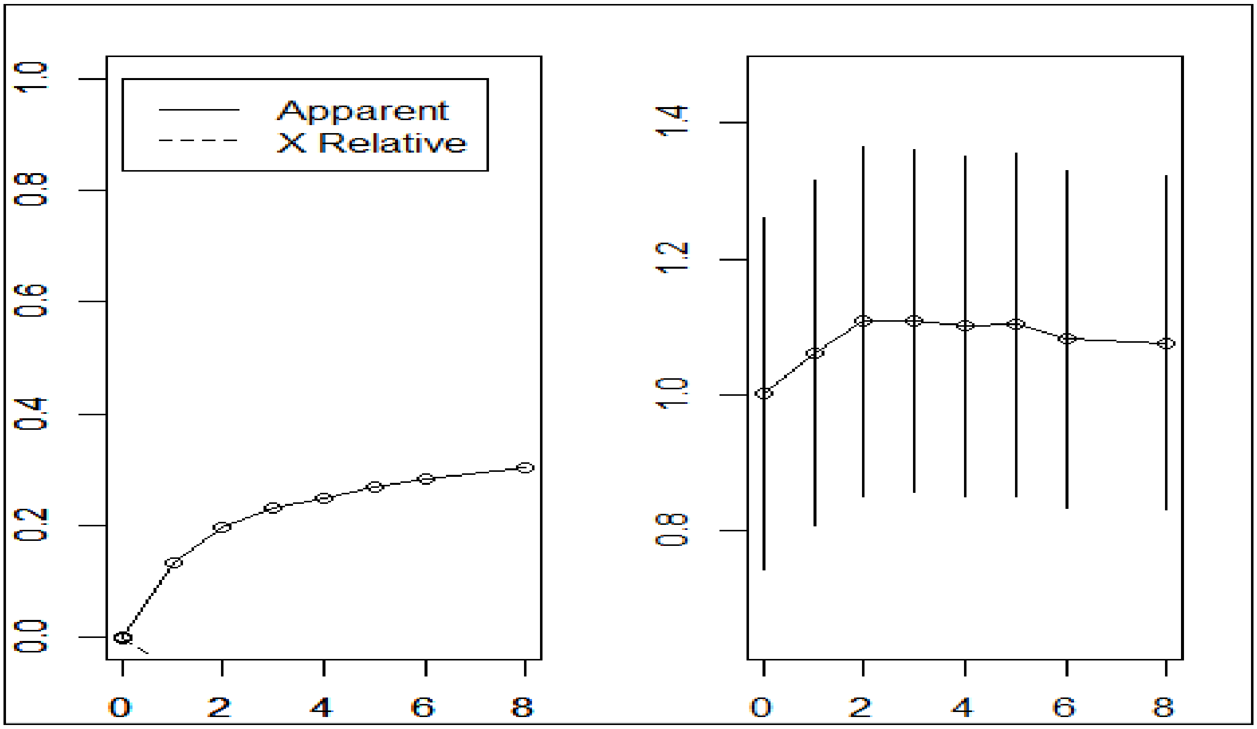
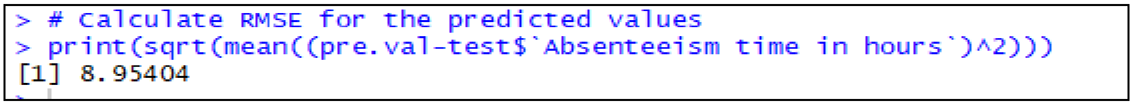


Figure 12: Plot shows the R-square and the residual distribution of the fitted model

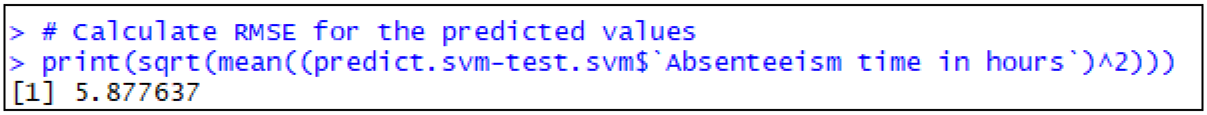
And the obtained RMS value for the prediction is-



```
> # Calculate RMSE for the predicted values
> print(sqrt(mean((pre.val-test$`Absenteeism time in hours`)^2)))
[1] 8.95404
```

SVM Regression

The RMS value for the SVM regression is 5.88



```
> # Calculate RMSE for the predicted values
> print(sqrt(mean((predict.svm-test.svm$`Absenteeism time in hours`)^2)))
[1] 5.877637
```

Hence, the RMSE value of the predicted model is nearly 9, which is pretty good for the simple model of this size. R-Square value is nearly 40% and this can further be improved by taking the interaction effect of the attributes. While on the other hand the RMSE value for the SVM regression is 5.88, which is the improvement over the decision tree.

Hence, SVM Regression is best for this study and the given data set.

Appendix A – R Codes

```
##### Reading data in R environment####


library(readxl) # Make sure to install this library

library(httr)


link = "https://s3-ap-southeast-1.amazonaws.com/edwisor-india-bucket/projects/data/DataN0101/Absenteeism_at_work_Project.xls"


GET(link, write_disk(dataset1 <- tempfile(fileext = ".xls")))

Absenteeism_at_work <- read_excel(dataset1)


# Checking the structure of the dataset


str(Absenteeism_at_work)

View(Absenteeism_at_work)


#### Cleaning the dataset

# Removing the first column, as it won't participate in modelling

Absenteeism_at_work = Absenteeism_at_work[-1]

data4svm = Absenteeism_at_work

data4svm = na.omit(data4svm)

##### Preprocessing the dataset#######

#Changing attributes to factors as they are taken as integer in R


Absenteeism_at_work$`Reason for absence` = as.factor(Absenteeism_at_work$`Reason for absence`)

Absenteeism_at_work$`Day of the week`= as.factor(Absenteeism_at_work$`Day of the week`)

Absenteeism_at_work$Seasons = as.factor(Absenteeism_at_work$Seasons)

Absenteeism_at_work$`Disciplinary failure`= as.factor(Absenteeism_at_work$`Disciplinary failure`)
```

```
Absenteeism_at_work$`Social drinker`= as.factor(Absenteeism_at_work$`Social drinker`)

Absenteeism_at_work$`Social smoker`=as.factor(Absenteeism_at_work$`Social smoker`)


# Again verifying the structure of the dataset

str(Absenteeism_at_work)


####### Cleanig the dataset. Looking for the NA values #########

# We will use the summary function to see the descriptive statistics and it will also check the NA values


any(is.na(Absenteeism_at_work)) #Check NA value. Our dataset is not clean

# Removing NA values from the dataset

Absenteeism_at_work = na.omit(Absenteeism_at_work)


####### Descriptive statistics###############


summary(Absenteeism_at_work) # We can see there is no NA values reported here. Hence, our dataset is clean.




###### Graphical visualization of the dataset  ######

# Scatter plot to check the correlation

plot.new()

pairs(Absenteeism_at_work)


# Find out which season has highest number of absetism

boxplot(Absenteeism_at_work$`Absenteeism time in hours`~Absenteeism_at_work$Seasons, col = "yellow")
```

```
# Digging more with visualization
library(ggplot2)  # Make sure to install this library
library(grid)  # Make sure to install this library
library(gridExtra)  # Make sure to install this library
plot.new()

p1t1 <- ggplot(Absenteeism_at_work, aes(x = Pet, fill = as.factor(Pet))) + geom_bar()
s1n1 <- ggplot(Absenteeism_at_work, aes(x = Son, fill = as.factor(Son))) + geom_bar()

S1S1 <- ggplot(Absenteeism_at_work, aes(x = `Social smoker`, fill =`Social drinker`)) + geom_bar()

Day1.wk1 <- ggplot(Absenteeism_at_work, aes(x = `Day of the week`, fill =  `Day of the week`)) + geom_bar()
Sn1s1 <- ggplot(Absenteeism_at_work, aes(x =   Seasons,fill = Seasons)) + geom_bar()

grid.arrange(p1t1,s1n1, nrow = 1)
grid.arrange(S1S1,Sn1s1, nrow = 1)
grid.arrange(Day1.wk1, nrow = 1)

# Histograms of numeric attributes of signifiance. Means parameters whcih are measured no the obvious one like Months
graphics.off()
par("mar")
par(mar = c(2,2,2,2))
plot.new()
par(mfrow=c(4,3))
hist(Absenteeism_at_work$`Transportation expense`, main="Histograms for Transportation Expenses", col="orange")
hist(Absenteeism_at_work$`Distance from Residence to Work`, main="Histograms for Distance from Residence to Work", col="orange")
hist(Absenteeism_at_work$`Service time`, main="Histograms for Service Time", col="orange")

hist(Absenteeism_at_work$Age, main="Histograms for Age", col="orange")

hist(Absenteeism_at_work$`Work load Average/day`, main="Histograms for Work load Average/day", col="orange")

hist(Absenteeism_at_work$`Hit target`, main="Histograms for Hit Target", col="orange")

hist(Absenteeism_at_work$Weight, main="Histograms for Weight", col="orange")

hist(Absenteeism_at_work$Height, main="Histograms for Height", col="orange")

hist(Absenteeism_at_work$`Body mass index`, main="Histograms for BMI", col="orange")

hist(Absenteeism_at_work$`Hit target`, main="Histograms for Hit Target", col="orange")

hist(Absenteeism_at_work$`Absenteeism time in hours`, main="Histograms for Absenteesim time", col="orange")


# We are also interested in knowing the density distribution of the Absenttes hours

plot(density(Absenteeism_at_work$`Absenteeism time in hours`), main="Distribution of the Abseneeism hour")

polygon(density(Absenteeism_at_work$`Absenteeism time in hours`), col="red", border="blue")



####### Planning for the modelling. ########

# As our data has both categorical and numerical data, we cannot do simple Linear regression for this.

# Hence, we choose to do deision tree regression Analysis.


# Making training data and testing data

set.seed(101)

index = sample(seq(nrow(Absenteeism_at_work)), size = 0.8*nrow(Absenteeism_at_work))


train = Absenteeism_at_work[index,]

test = Absenteeism_at_work[-index,]
```

```
##### Modelling #######

library(rpart) # Make sure to install this library


tree.model <- rpart(train$`Absenteeism time in hours`~ ., data = train, method="anova")


summary(tree.model)

tree.model

# Visualize the fitted tree

graphics.off()

par("mar")

par(mar = c(2,2,2,2))

plot.new()

plot(tree.model)

text(tree.model, pretty=0)


par(mfrow=c(1,2))

rsq.rpart(tree.model) # So maximum R square is 40% with 7 splits


###### Predictiona ######

pre.val = predict(tree.model, newdata = test, method = "anova")

plot(pre.val, test$`Absenteeism time in hours`)

abline(0,1)

# Calculate RMSE for the predicted values

print(sqrt(mean((pre.val-test$`Absenteeism time in hours`)^2)))



#### Model 2 svm regression####

#As svm works on numeric data, we have to use all the numeric value instead of categorical
```

```
# Preparing test and train data for the svm model

set.seed(101)
index = sample(seq(nrow(data4svm)), size = 0.8*nrow(data4svm))

train.svm = data4svm[index,]
test.svm = data4svm[-index,]



library(e1071)

svm.regression <- svm(train.svm$`Absenteeism time in hours` ~ . , data=train.svm)
svm.regression

#Prediction

predict.svm = predict(svm.regression, newdata = test.svm, method = "anova")
predict.svm
# Calculate RMSE for the predicted values
print(sqrt(mean((predict.svm-test.svm$`Absenteeism time in hours`)^2)))

####Final Comments#####
#The model can be further improved with higher level complex interaction between the numerical variables
#For the simplicity that has not be been included in this study
```

Appendix B – Python Code

```
# Make sure to install all these libraries
import pandas as pd
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
from sklearn import metrics
Absent_work = pd.read_excel("https://s3-ap-southeast-1.amazonaws.com/edwisor-india-bucket/projects/data/DataN0101/Absenteeism_at_work_Project.xls")
Absent_work
Absent_work.describe() # We can see that all the attributes are treated as integers. Which is not desired
# Further checking the data type of the attributes
Absent_work.info(verbose = True)
# Cleaning the dataset
Absent_work.columns
Absent_work.drop(['ID'], axis = 1,inplace = True)
Absent_work.columns
#### Preprocessing the dataset#######
#Changing attributes to factors as they are taken as integer in python
# Checking NA values. And removing the missing value rows
Absent_work.isnull().sum()
# Changing the attribtues to categorical
Absent_work['Reason for absence']  = Absent_work['Reason for absence'].astype('category')
Absent_work['Seasons']  = Absent_work['Seasons'].astype('category')
Absent_work['Day of the week']  = Absent_work['Day of the week'].astype('category')
Absent_work['Disciplinary failure']  = Absent_work['Disciplinary failure'].astype('category')
Absent_work['Social drinker']  = Absent_work['Social drinker'].astype('category')
Absent_work['Social smoker']  = Absent_work['Social smoker'].astype('category')
```

```
Absent_work.dtypes

get_ipython().magic('matplotlib inline')

Absent_work.hist(column=None, by=None, grid=True, xlabelsize=None, xrot=None, ylabelsize=None, yrot=None, ax=None, sharex=False, sharey=False, figsize=(20,20), layout=(5,3), bins=10)

get_ipython().magic('matplotlib inline')

boxplot = Absent_work.boxplot(column=['Month of absence',

       'Transportation expense', 'Distance from Residence to Work',

       'Service time', 'Age', 'Work load Average/day ', 'Hit target', 'Education', 'Son', 'Pet', 'Weight', 'Height', 'Body mass index',

       'Absenteeism time in hours'], return_type='axes', figsize = (30,30))

Boxplot

##### Training the model#####

# Making the train and test split

from sklearn.model_selection import train_test_split

train, test = train_test_split(Absent_work, test_size=0.2)

print("Length of train set: "+str(len(train)))

print("Length of test set: " +str(len(test)))

# Training set creation

train_x = train.drop(['Absenteeism time in hours'], axis=1)

train_y = train['Absenteeism time in hours']


# Test set creation


test_x = test.drop(['Absenteeism time in hours'], axis=1)

test_y = test['Absenteeism time in hours']

print("column name of train dataset:  ", train_x.columns, "\n \n column name of the test dataset: " ,test_x.columns)

from sklearn.tree import DecisionTreeRegressor

d_tree = DecisionTreeRegressor(random_state = 0)
```

```
model = d_tree.fit(train_x, train_y)
#Prediction
predict_value = model.predict(test_x)
predict_value
# Calculate RMSE for the predicted values
print((((predict_value - test_y)**2).mean())**(1/2))
```