EMPLOYEE ABSENTEEISM

Raghav Kotwal

21 APRIL 2019

Contents

Chapter 1

Introduction

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

We need to build a regression model on the given data to predict employee Absenteeism and provide the courier company ways by which they can bring down the absenteeism rate.

```
'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 ...
```

Table 1.1 Structure of the Data

	ID	Reason.for.absence	Month.of.absence	Day.of.the.week	Seasons	Transportation.expense
1	11	26	7	3	1	289
2	36	0	7	3	1	118
3	3	23	7	4	1	179
4	7	7	7	5	1	279
5	11	23	7	5	1	289
6	3	23	7	6	1	179

	Distance.from.Residence.to.Work	Service.time	Age	Work.load.Average.day.	Hit.target	Disciplinary.failure
1	36	13	33	239554	97	0
2	13	18	50	239554	97	1
3	51	18	38	239554	97	0
4	5	14	39	239554	97	0
5	36	13	33	239554	97	0
6	51	18	38	239554	97	0

	Education	Son	Social.drinker	Social.smoker	Pet	Weight	Height	Body.mass.index	Absenteeism.time.in.hours
1	1	2	1	0	1	90	172	30	4
2	1	1	1	0	0	98	178	31	0
3	1	0	1	0	0	89	170	31	2
4	1	2	1	1	0	68	168	24	4
5	1	2	1	0	1	90	172	30	2
6	1	0	1	0	0	89	170	31	NA

Table 1.2 First five rows of the Data

The table below contains the variables which we will be using to predict the target variable.

S.No.	Predictor
1	ID
2	Reason for Absence
3	Month of Absence
4	Transportation Expense
5	Distance from Residence to Work
6	Service Time
7	Age
8	Work Load
9	Average Day
10	Hit Target
11	Disciplinary Failure
12	Son
13	Pet
14	Height
15	Body Mass Index
16	Absenteeism in Hours

Chapter 2

Methodology

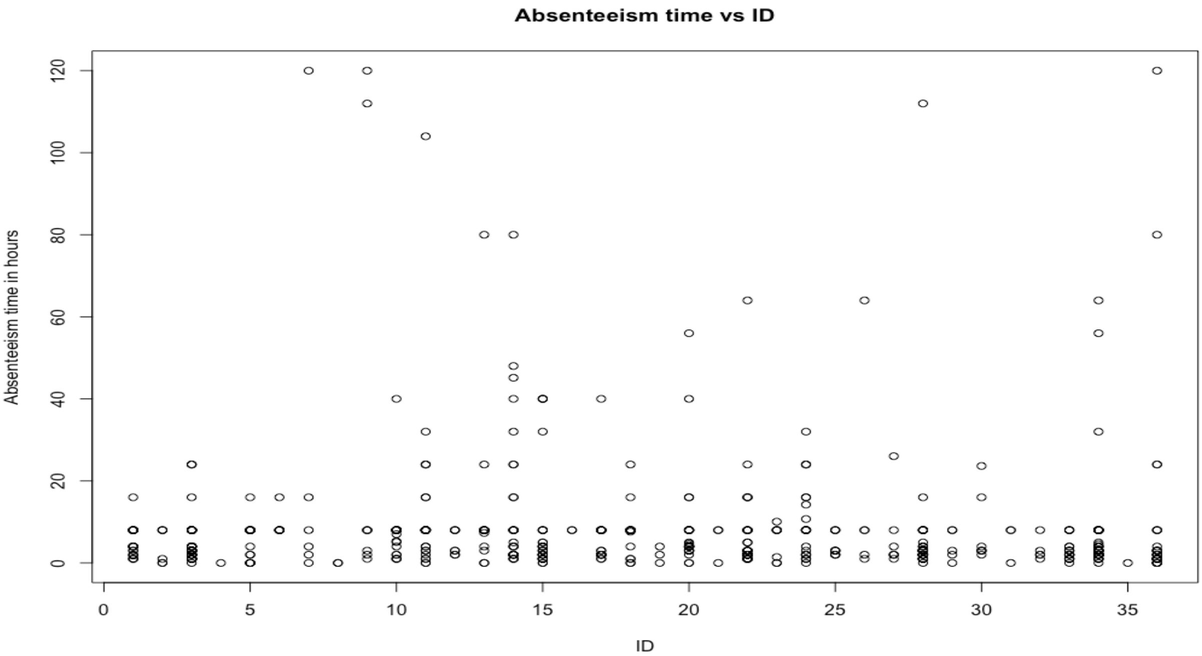
2.1 Pre-Processing

Any predictive modeling requires that we look at the data before we start modeling. However, in data mining terms looking at data refers to so much more than just looking. Looking at data refers to exploring the data, cleaning the data as well as visualizing the data through graphs and plots. This is often called as Exploratory Data Analysis. To start this process, we will do missing value analysis followed by visualization of data for distribution of all the variables.

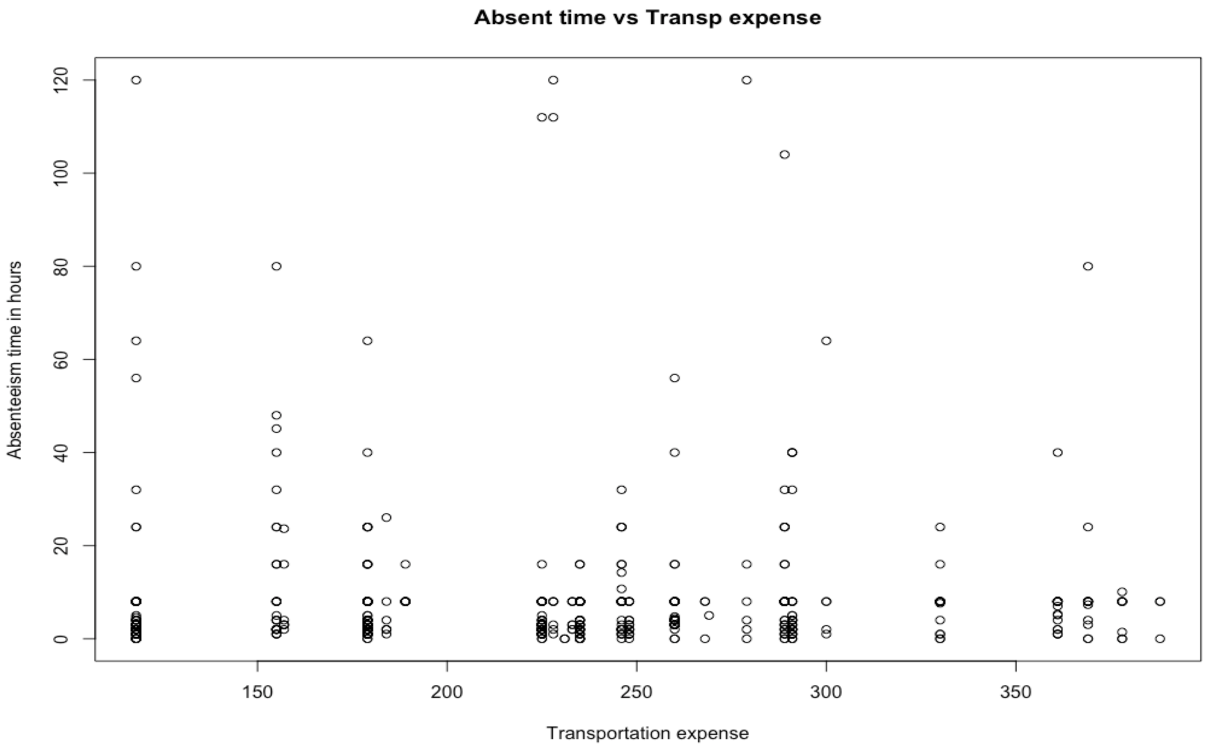
2.1.1 Missing Value Analysis

Missing value play a vital role in demonstrating how a model would perform. If a variable has more than 30% of its values missing, then those values can be ignored, or the column itself is ignored. In our case, none of the columns have a high percentage of missing values. The maximum missing percentage is 4.18% i.e., Body Mass Index column. After evaluating missing values using mean, median and KNN methods we have found KNN to be most accurate and hence imputed values using KNN method.

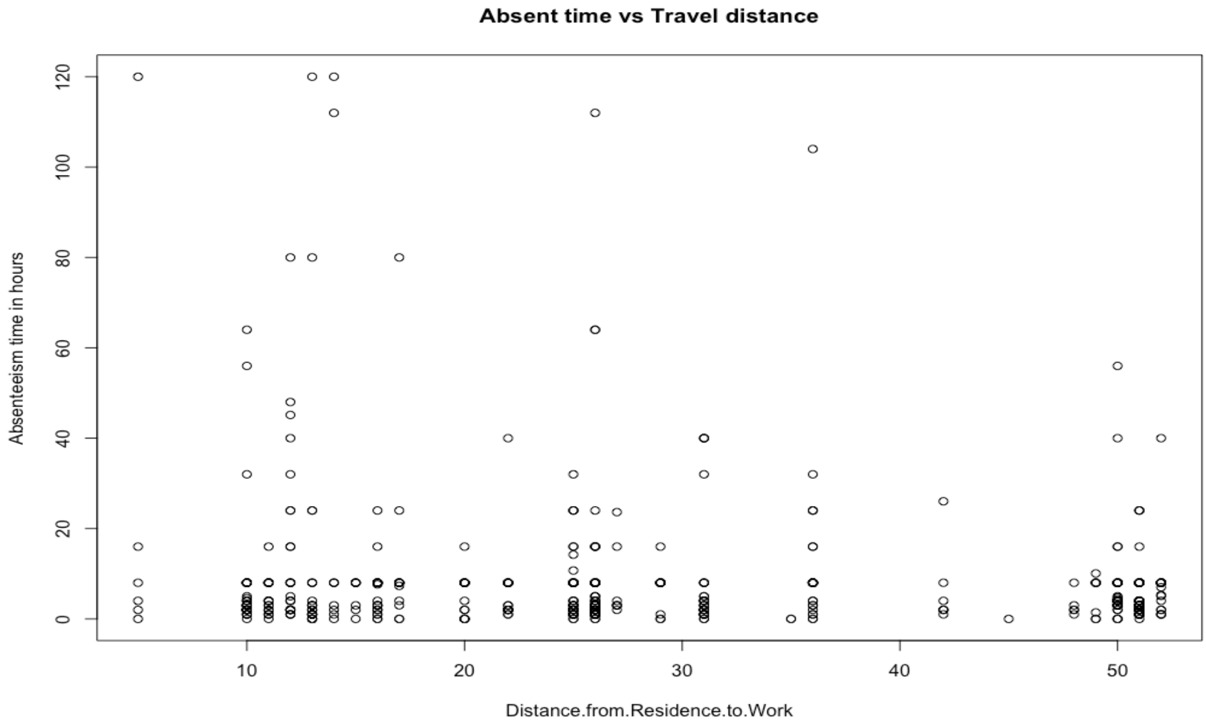
2.1.2 Visualizing Continuous Variables vs Target variable



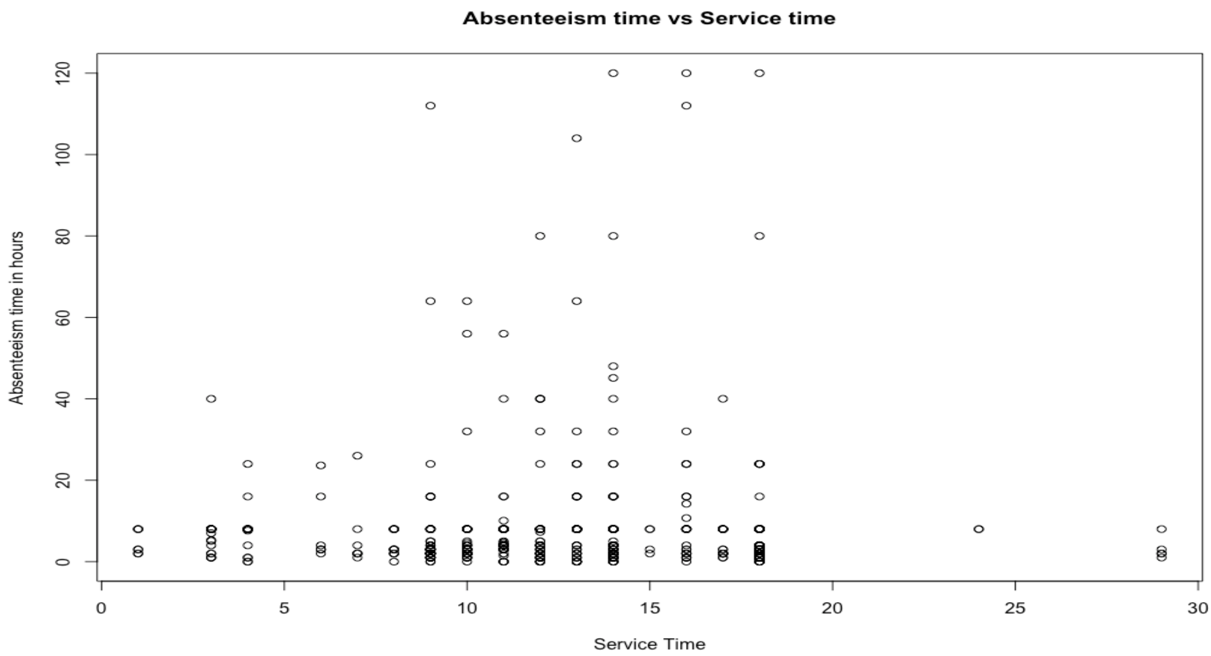
By visualizing the variable, ID it can be seen that some ID's are seen to be occurring more than others and hence the plot is scattered. Also, the variable ID's is having average Absenteeism hour in range 10-15.



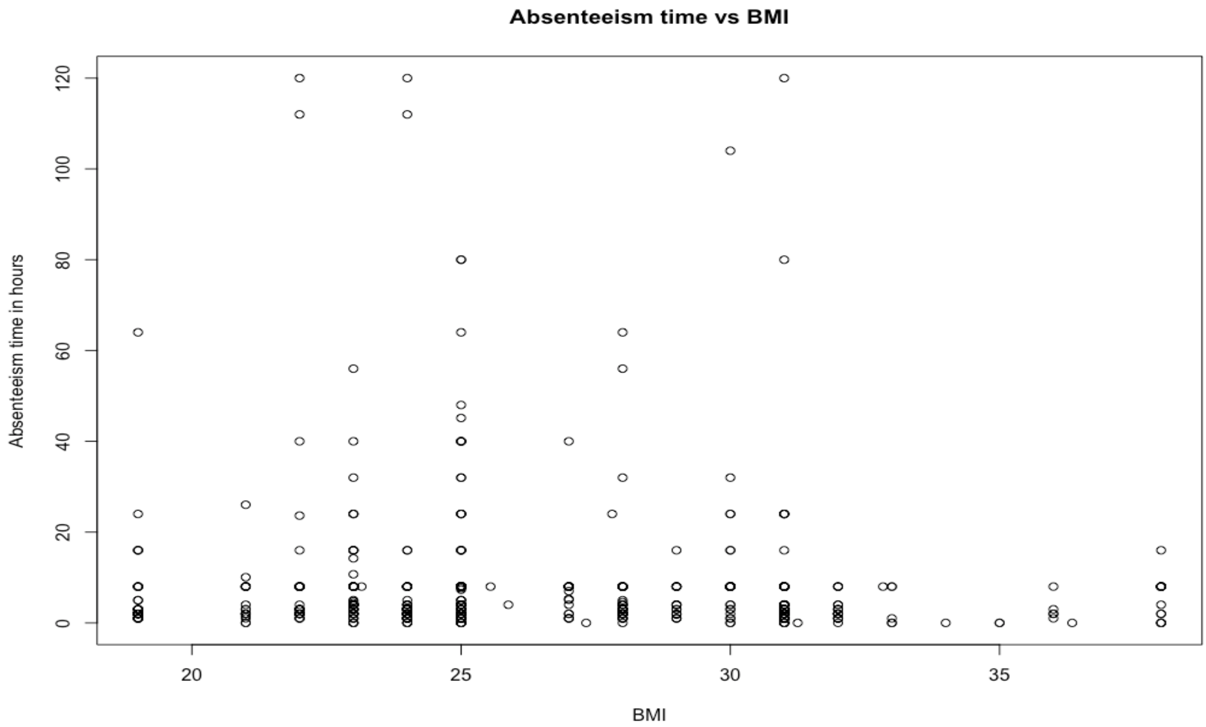
The plot above shows that most of the transportation expense is also cause target variable to be around 10 hours average.



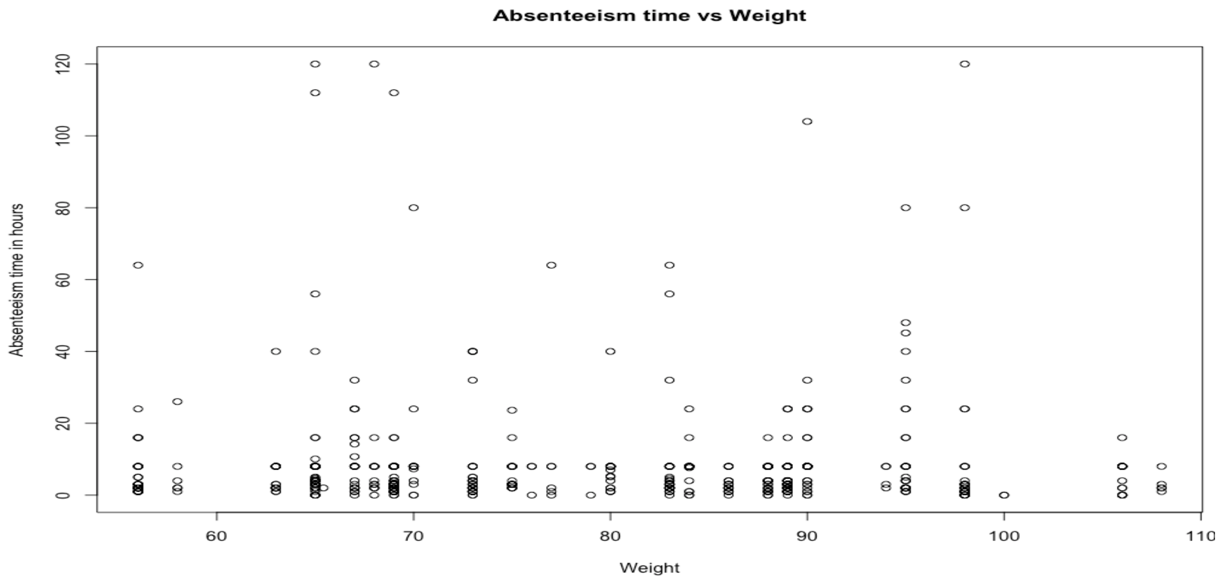
The variable distance from work is also scattered still showing maximum Absenteeism around 30 and 50 kms.



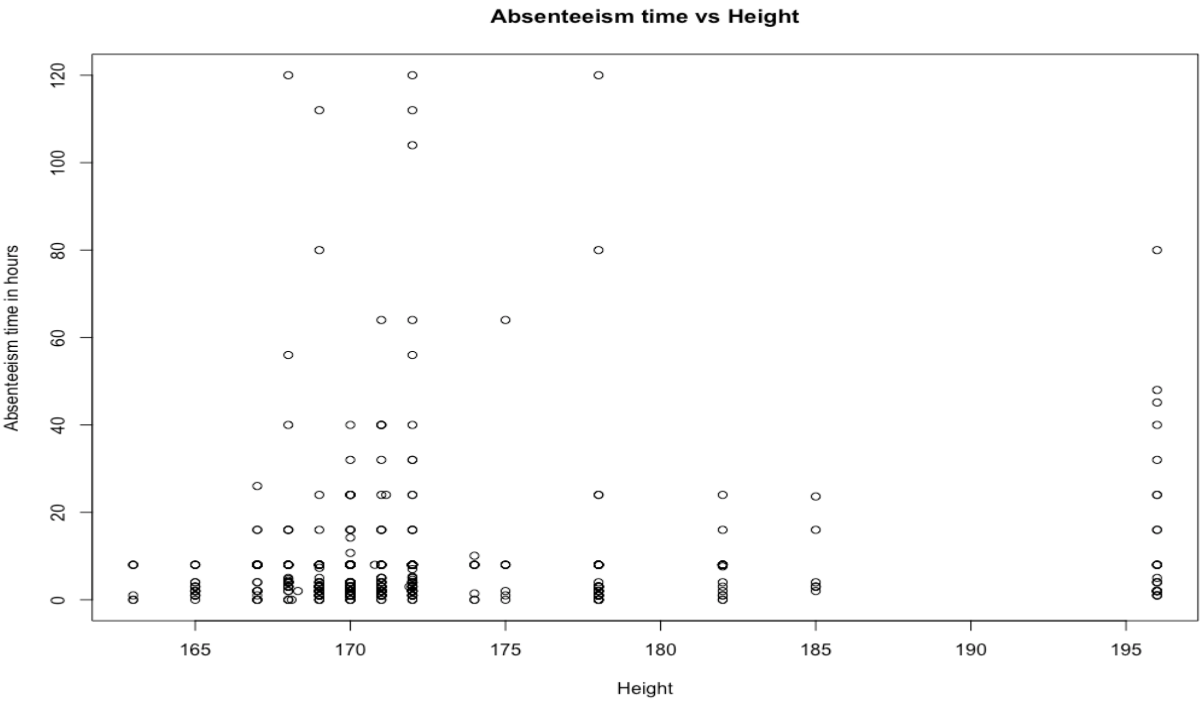
This variable shows that almost all the absenteeism is around employees working from 7- 17 hours.



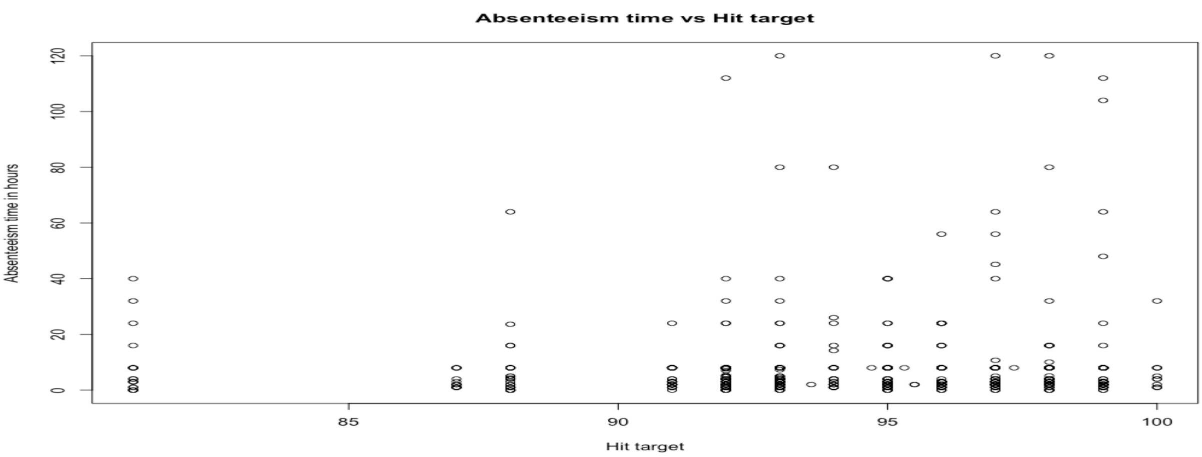
For variable BMI higher Absenteeism hours are seen around the BMI rate 23 above till 31.



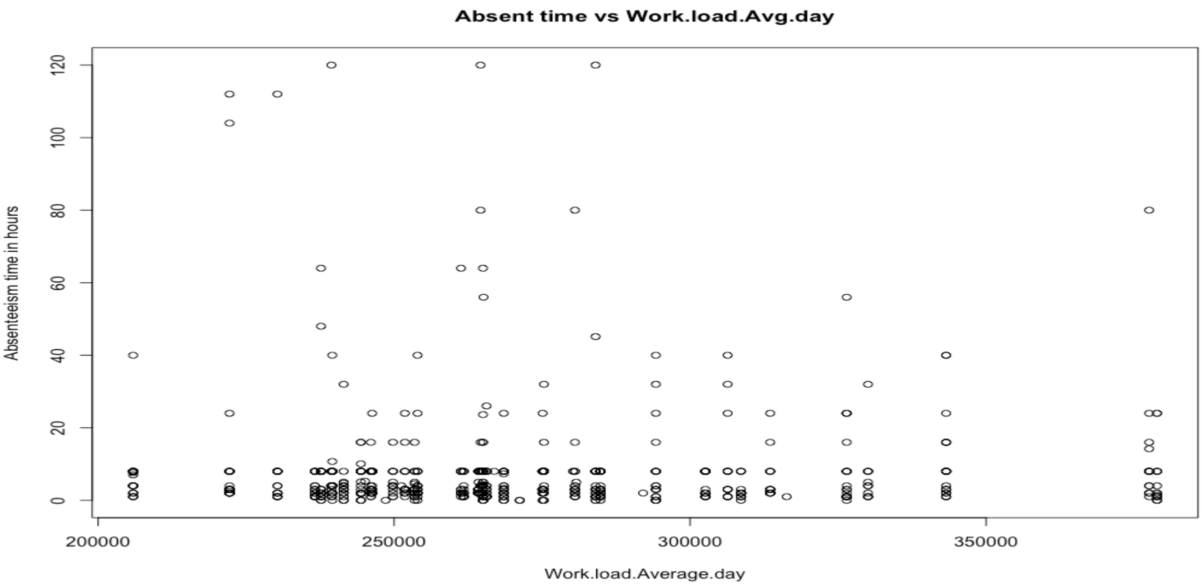
The above plot for Weight vs Absenteeism is high for certain weight categories.



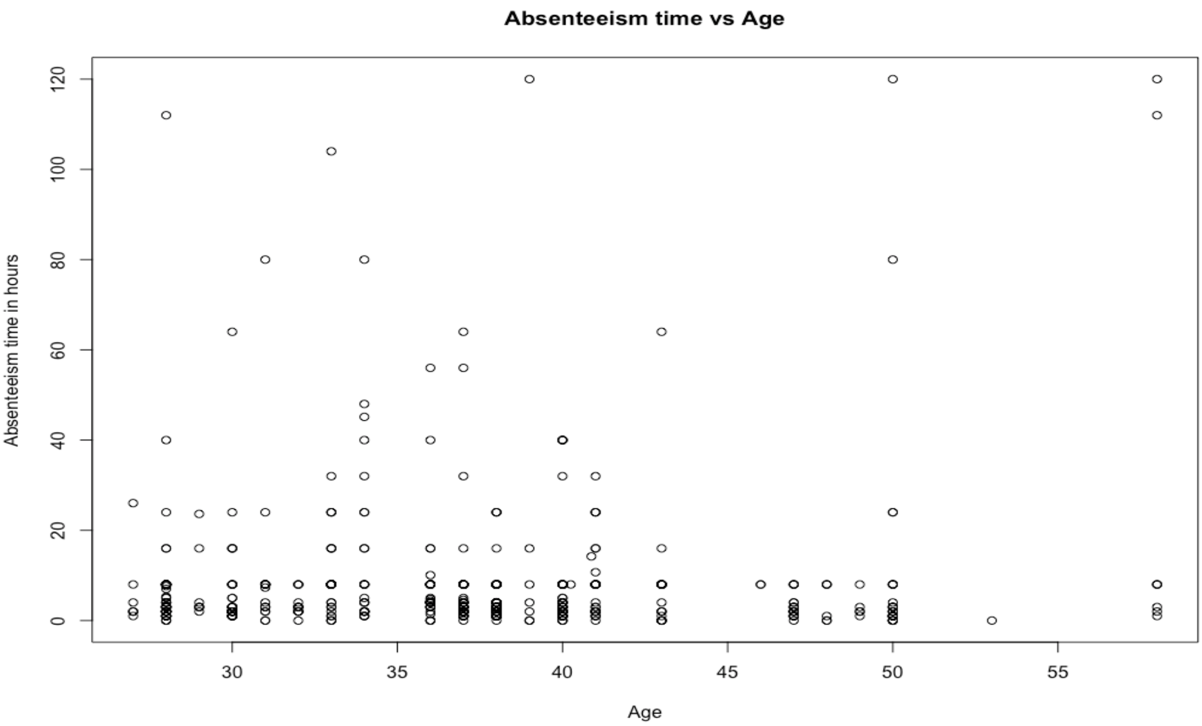
The employees are mostly of 170 unit’s high. Absenteeism hours is mostly concentrated 168-172 units of height. Some outlier can be see for more than 195 units also.



Hit target variable is mainly concentrated above 90 and we can see that maximum absenteeism in this variable.

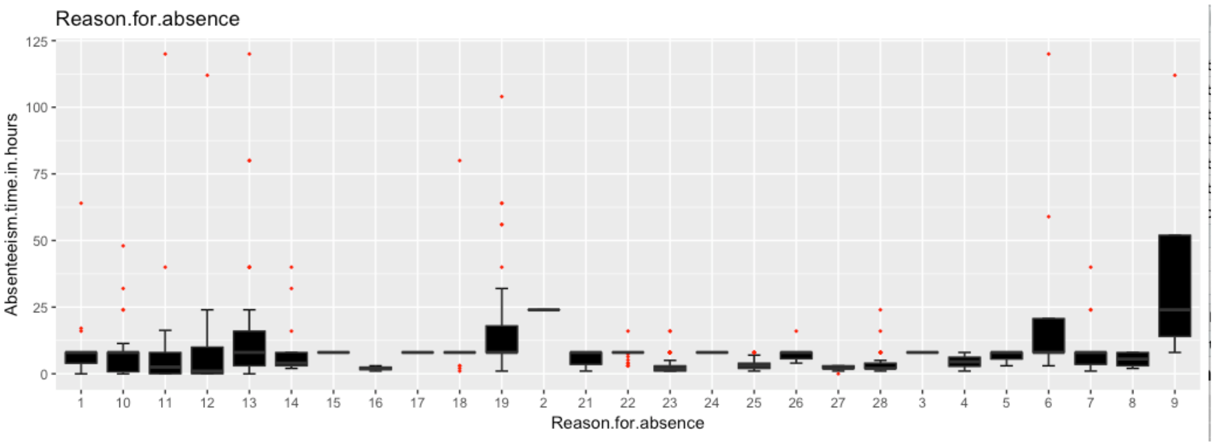


From the given plot, it is clearly visible that maximum absenteeism is around 230000- 280000.

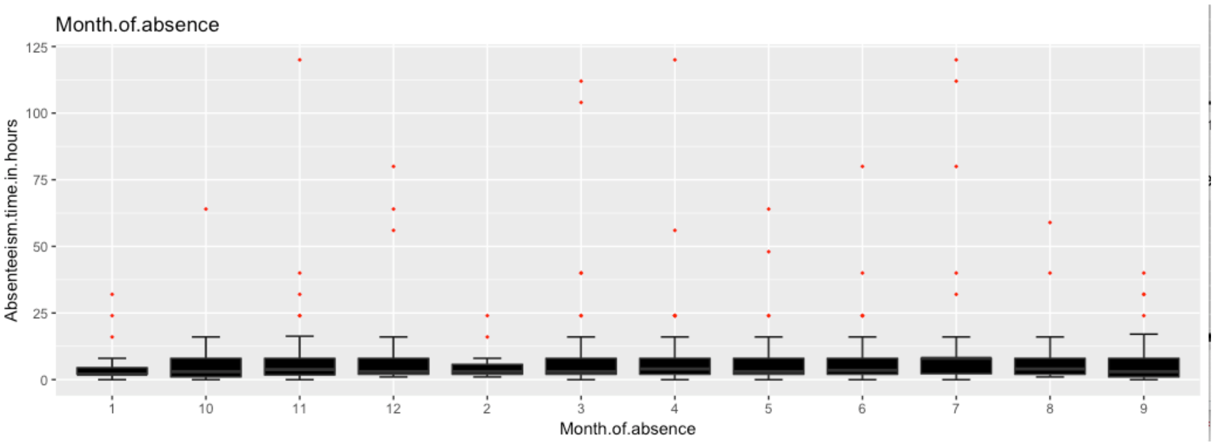


On comparing age against absenteeism hours, we see that absenteeism hours are concentrated in range with peak around 33-34 years of age.

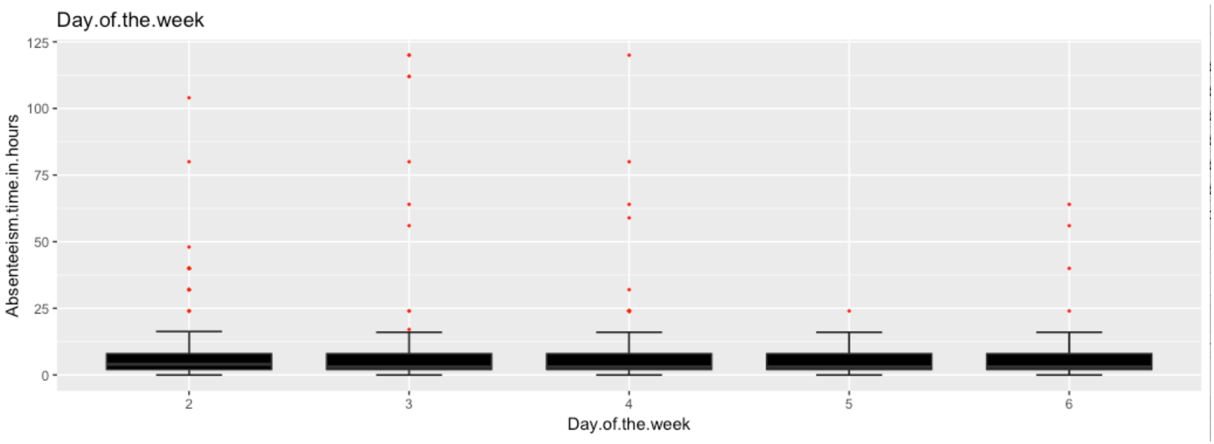
2.1.3 Visualizing Categorical Variables vs Target Variable



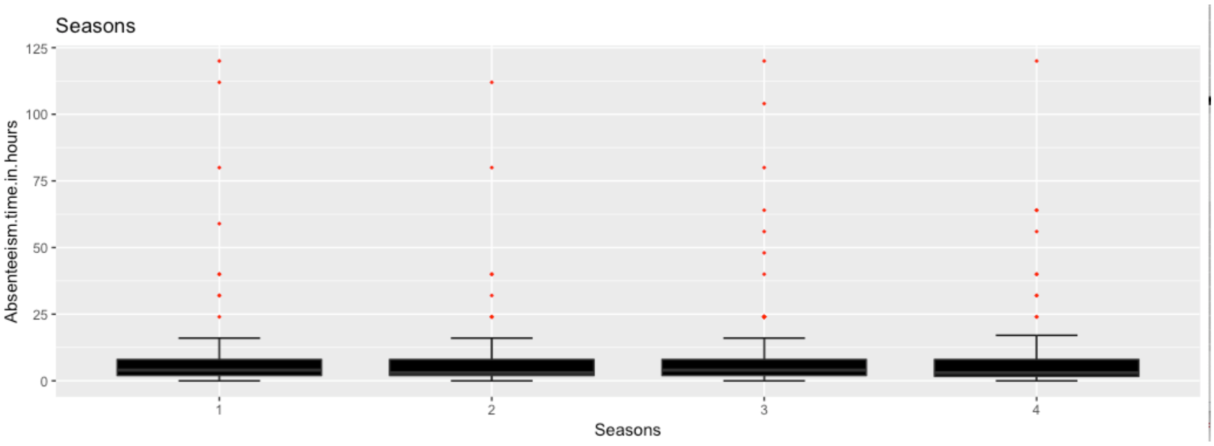
For the variable Reason for Absence it can see that maximum median is found for reason no. 9 (Disease of Circulatory system). We see that there is wide range of median absenteeism hours for this feature set.



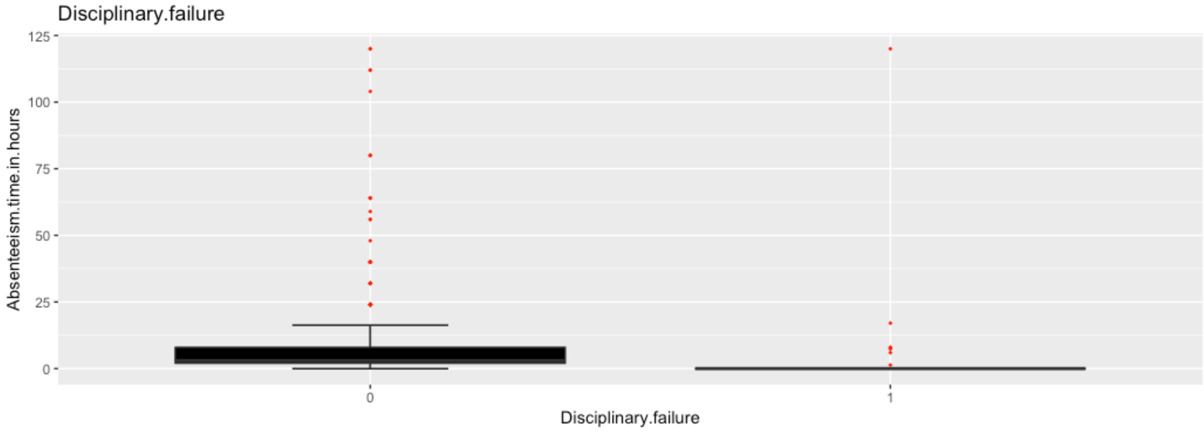
It is clearly visible that Month of absence has higher absenteeism in some months.



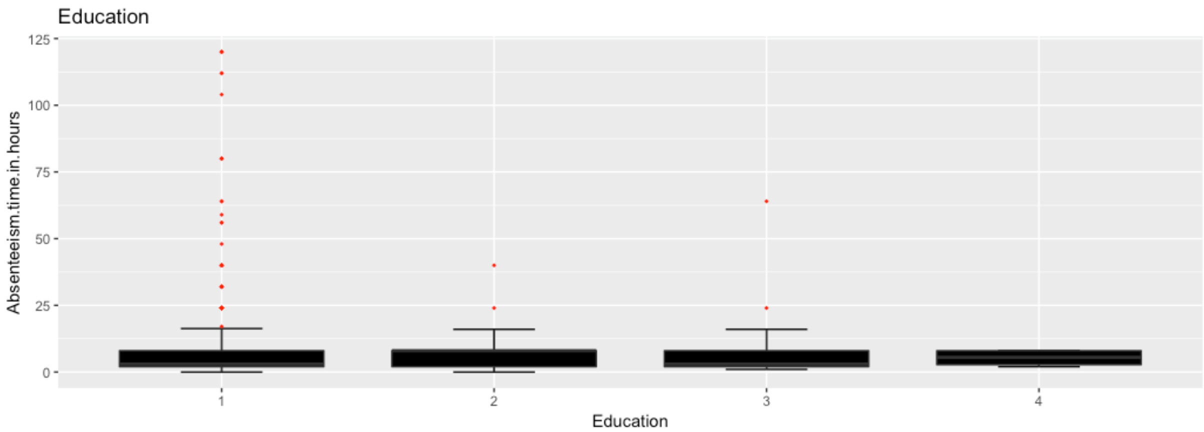
The range and absenteeism values are almost uniform for all the days of the week.



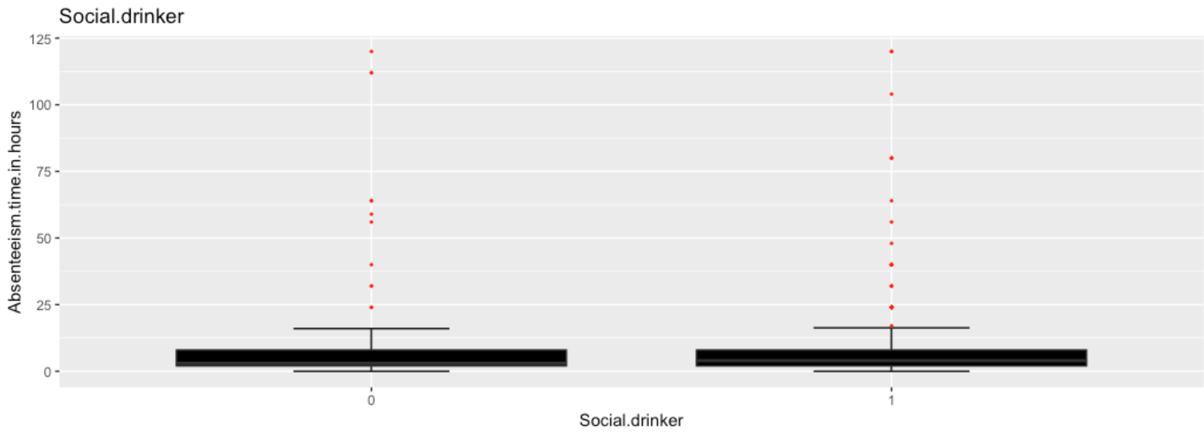
The range and absenteeism values are almost uniform for all the seasons.



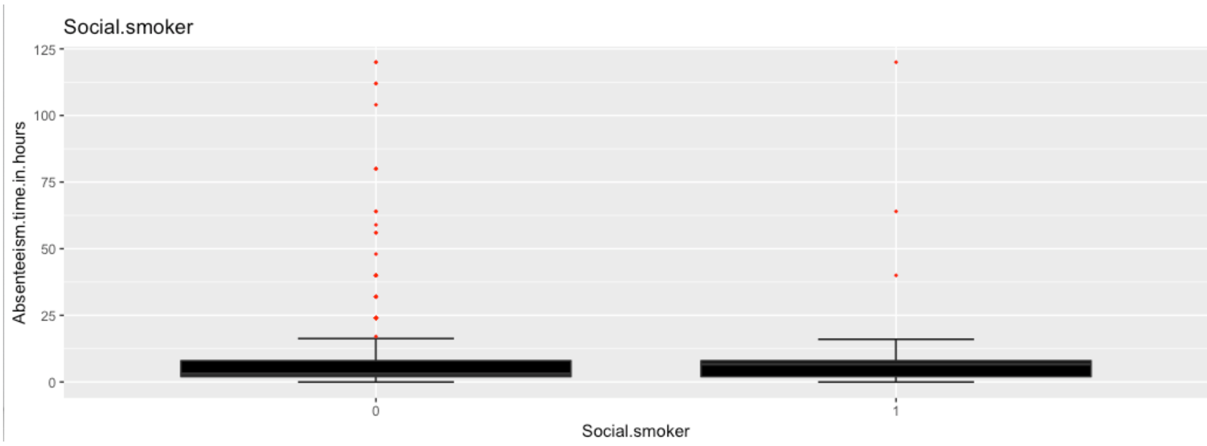
we see that absenteeism hours are found slightly higher in employees with no disciplinary failure.



The range and median of absenteeism hours grouped by the education level is mostly uniform. Also, high school educated employees show more number of absenteeism hours.



Social drinker data shows uniformity and also the absenteeism is also equal number of absenteeism hours.



There is mostly no social smoker in our data set. When we observe absenteeism hours grouped by social smoker, the median value and the range are almost the same.

2.1.4 Outlier Analysis

By visualizing the data, it can be seen that seen that some continuous variables are containing outliers thus we need to replace outlier values with NA and then apply KNN imputation to replace these values.

Before applying Outlier Analysis

```
[1] "ID"
numeric(0)
[1] "Transportation.expense"
[1] 388 388 388
[1] "Distance.from.Residence.to.Work"
numeric(0)
[1] "Service.time"
[1] 29 29 29 29 29
[1] "Age"
[1] 58 58 58 58 58 58 58 58
[1] "Work.load.Average.day."
 [1] 378884 378884 378884 378884 378884 378884 378884 378884 378884 378884 378884 378884 378884 378884
[15] 378884 377550 377550 377550 377550 377550 377550 377550 377550 377550 377550 377550 377550 377550
[29] 377550 377550 377550
[1] "Hit.target"
 [1] 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81
[1] "Son"
factor(0)
Levels: 0 1 2 3 4
[1] "Pet"
factor(0)
Levels: 0 1 2 4 5 8
[1] "Height"
  [1] 178 196 182 185 163 163 163 196 178 178 196 196 178 196 196 178 178 182 182 185 196 196 196 196 178
 [26] 196 163 196 196 196 182 178 178 196 178 178 178 196 178 182 196 182 182 185 185 178 178 178 178 178
 [51] 178 178 185 178 196 182 178 185 196 196 178 185 178 178 178 178 182 182 178 178 182 178 182 178 182
 [76] 163 178 182 196 178 178 182 178 178 196 196 182 178 178 196 178 178 178 196 178 178 196 178 182 178
[101] 182 182 178 178 178 196 178 182 178 196 196 178 178 163 178 178 178 178 178
[1] "Weight"
numeric(0)
[1] "Body.mass.index"
numeric(0)
[1] "Absenteeism.time.in.hours"
 [1]  40.00000  40.00000  32.00000  17.06891  32.00000  40.00000  24.00000  64.00000  56.00000  40.00000
[11]  40.00000  24.00000  24.00000  24.00000  56.00000  24.00000  24.00000  24.00000  24.00000  80.00000
[21]  32.00000  58.95413  24.00000  32.00000  40.00000  64.00000 120.00000  32.00000  24.00000 120.00000
[31]  40.00000  24.00000 112.00000  24.00000  80.00000  24.00000 112.00000  24.00000 104.00000  24.00000
[41]  64.00000  48.00000  24.00000 120.00000  80.00000
```

After applying Outlier Analysis

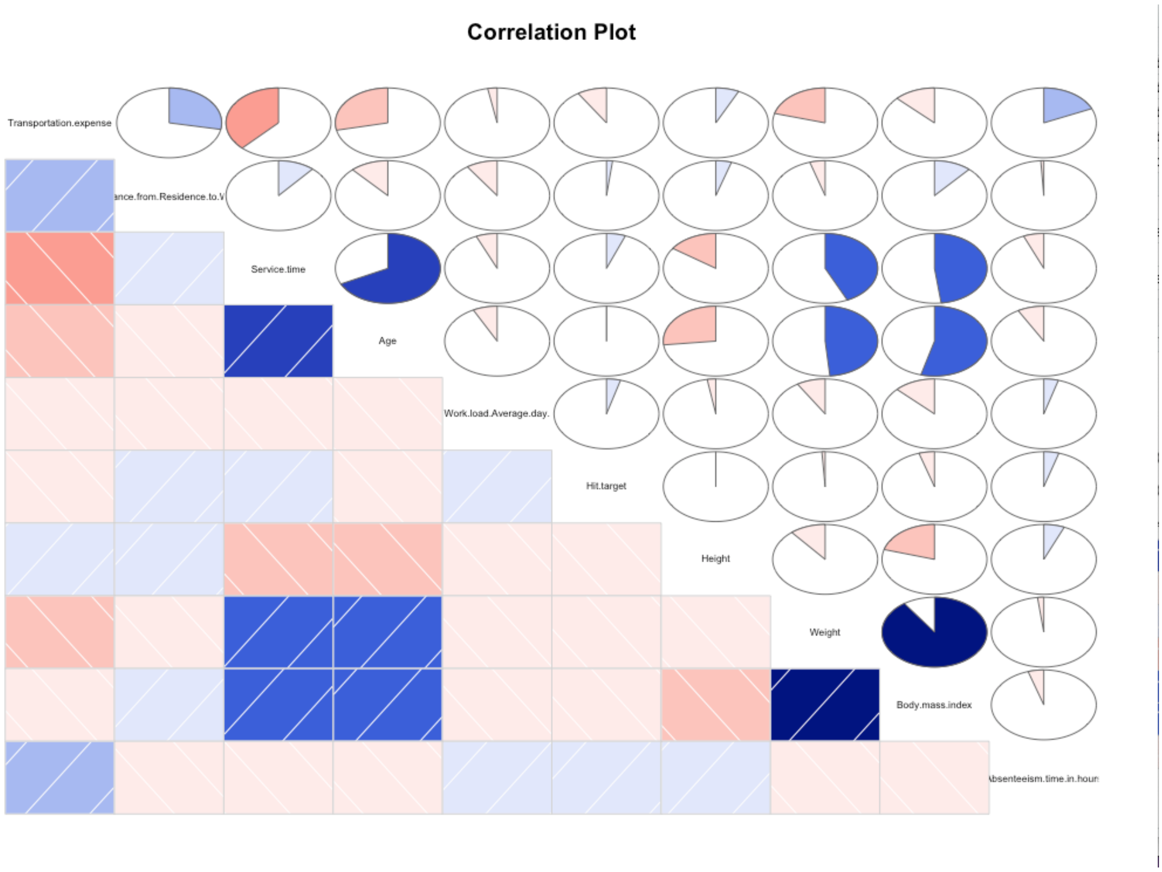
```
[1] "ID"
numeric(0)
[1] "Transportation.expense"
numeric(0)
[1] "Distance.from.Residence.to.Work"
numeric(0)
[1] "Service.time"
numeric(0)
[1] "Age"
numeric(0)
[1] "Work.load.Average.day."
numeric(0)
[1] "Hit.target"
numeric(0)
[1] "Son"
factor(0)
Levels: 0 1 2 3 4
[1] "Pet"
factor(0)
Levels: 0 1 2 4 5 8
[1] "Height"
numeric(0)
[1] "Weight"
numeric(0)
[1] "Body.mass.index"
numeric(0)
[1] "Absenteeism.time.in.hours"
numeric(0)
```

2.1.5 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. We will be doing this with the use of correlation analysis and Annova test.

A correlation analysis gives us the idea about the multicollinearity between different independent variables. If the correlation between two variables is high, it means they are actually saying the same thing. So, for better analysis we need to remove such variables.

The figure 4.1 shows the correlation between the numerical variables in the data.



As it is clearly seen in the correlation plot that there is high correlation between the variable's 'Weight' and 'Body Mass Index'. So, to make the data a better fit for analysis we should remove 'Weight'. Also, after applying annova test on categorical variables we can eliminate "Day of the week", "Seasons", "Education", "Social smoker", "Social drinker".

2.2 Modelling

2.2.1 Model Selection

In our early stages of analysis during preprocessing we have come to understand that our data is not skewed and is in normal continuous form and since our target variable is also continuous, so it will be best to predict the target variable using Regression models.

We are starting the modelling with Linear regression and then Decision Tree.

2.2.1.1 Linear Regression

```
modelLR = lm(Absenteeism.time.in.hours~.,data = df[train,])
summary(modelLR)

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

Residuals:
    Min      1Q  Median      3Q     Max 
-6.1432 -1.3665 -0.1247  0.8582 12.8180 

Coefficients:
                                  Estimate Std. Error t value Pr(>|t|)
```

```
(Intercept)                        1.938e+00  1.579e+01   0.123  0.90238
ID                                -5.504e-02  2.105e-02  -2.615  0.00920 **
Reason.for.absence10               8.617e-02  8.057e-01   0.107  0.91487
Reason.for.absence11              -1.311e+00  7.929e-01  -1.654  0.09881 .
Reason.for.absence12              -7.615e-01  1.032e+00  -0.738  0.46093
Reason.for.absence13              -1.018e+00  7.765e-01  -1.311  0.19058
Reason.for.absence14              -1.943e+00  9.763e-01  -1.991  0.04707 *
Reason.for.absence15               5.952e-01  2.022e+00   0.294  0.76863
Reason.for.absence16              -5.397e+00  1.712e+00  -3.153  0.00171 **
Reason.for.absence17              -7.867e-01  2.794e+00  -0.282  0.77835
Reason.for.absence18              -2.036e-01  9.521e-01  -0.214  0.83077
Reason.for.absence19               8.650e-02  8.293e-01   0.104  0.91697
Reason.for.absence2               -4.974e+00  2.795e+00  -1.780  0.07572 .
Reason.for.absence21              -1.785e+00  1.372e+00  -1.301  0.19380
Reason.for.absence22               6.082e-01  8.758e-01   0.694  0.48769
Reason.for.absence23              -3.770e+00  7.070e-01  -5.333 1.46e-07 ***
Reason.for.absence24               6.970e-01  1.741e+00   0.400  0.68898
Reason.for.absence25              -3.765e+00  8.557e-01  -4.399 1.32e-05 ***
Reason.for.absence26               1.351e-02  8.540e-01   0.016  0.98738
Reason.for.absence27              -4.554e+00  8.180e-01  -5.567 4.21e-08 ***
Reason.for.absence28              -4.042e+00  7.314e-01  -5.526 5.25e-08 ***
Reason.for.absence3                2.719e+00  2.810e+00   0.968  0.33357
Reason.for.absence4               -2.486e+00  2.072e+00  -1.200  0.23086
Reason.for.absence5                2.184e-01  1.696e+00   0.129  0.89758
Reason.for.absence6               -6.721e-01  1.287e+00  -0.522  0.60166
Reason.for.absence7               -1.850e+00  1.011e+00  -1.829  0.06795 .
Reason.for.absence8               -1.247e+00  1.377e+00  -0.905  0.36577
Reason.for.absence9                1.845e+00  1.705e+00   1.082  0.27977
Month.of.absence10                 4.186e-01  7.733e-01   0.541  0.58859
Month.of.absence11                 4.366e-01  6.706e-01   0.651  0.51528
Month.of.absence12                 4.544e-01  7.394e-01   0.615  0.53907
Month.of.absence2                  4.662e-01  6.372e-01   0.732  0.46477
Month.of.absence3                  1.737e+00  6.458e-01   2.689  0.00739 **
Month.of.absence4                  6.566e-01  6.876e-01   0.955  0.34011
Month.of.absence5                  2.509e-01  7.452e-01   0.337  0.73649
Month.of.absence6                  2.701e-01  7.199e-01   0.375  0.70767
Month.of.absence7                  8.124e-01  7.196e-01   1.129  0.25949
Month.of.absence8                  4.802e-01  8.159e-01   0.588  0.55649
Month.of.absence9                  1.789e-01  7.886e-01   0.227  0.82061
Transportation.expense             6.659e-03  3.729e-03   1.786  0.07476 .
Distance.from.Residence.to.Work -1.926e-02  1.578e-02  -1.220  0.22298
Service.time                      -6.532e-02  8.840e-02  -0.739  0.46031
Age                                1.945e-02  5.766e-02   0.337  0.73597
Work.load.Average.day.             2.383e-06  4.623e-06   0.515  0.60647
Hit.target                        -6.255e-02  5.599e-02  -1.117  0.26449
Disciplinary.failure1             -5.788e+00  6.060e-01  -9.551  < 2e-16 ***
Son1                              -3.793e-01  4.937e-01  -0.768  0.44269
Son2                               6.880e-01  5.428e-01   1.267  0.20558
Son3                              -1.624e+00  1.153e+00  -1.408  0.15970
Son4                               1.016e+00  6.649e-01   1.528  0.12711
Pet1                              -1.531e+00  5.516e-01  -2.776  0.00571 **
```

```
Pet2                              -2.106e-02  6.205e-01  -0.034  0.97294
Pet4                              -1.104e+00  1.224e+00  -0.901  0.36782
Pet5                              -1.466e-01  1.656e+00  -0.089  0.92951
Pet8                              -2.527e+00  1.593e+00  -1.586  0.11332
Height                             5.437e-02  8.431e-02   0.645  0.51927
Body.mass.index                    3.654e-02  4.312e-02   0.847  0.39714
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.646 on 508 degrees of freedom
Multiple R-squared:  0.4594,  Adjusted R-squared:  0.3998
F-statistic: 7.709 on 56 and 508 DF,  p-value: < 2.2e-16
```

```
par(mar = c(2,2,2,2))
par(mfrow = c(3,1))
plot(modelLR)
predictLR = predict(modelLR,df[test,])
plot(df$Absenteeism.time.in.hours[test])
lines(predictLR,col='blue')
```

From the summary of the data it is clearly seen that there are four variables which have higher importance i.e. Reason for absence, Month of Absence, disciplinary failure and Pet. Thus, we will remodel the Linear Regression using these four variables.

```
modelLR =
lm(Absenteeism.time.in.hours~Reason.for.absence+Month.of.absence+Dis
ciplinary.failure+Pet,data = df[train,])
summary(modelLR)
par(mar = c(2,2,2,2))
par(mfrow = c(3,1))
plot(modelLR)
predictLR = predict(modelLR,df[test,])
```

```
Call:
lm(formula = Absenteeism.time.in.hours ~ Reason.for.absence +
    Month.of.absence + Disciplinary.failure + Pet, data = df[train,
    ])
```

```
Residuals:
    Min      1Q  Median      3Q     Max 
-6.1103 -1.4496 -0.1189  0.9345 12.9249 

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)    
(Intercept)            6.589113   0.825470   7.982 9.23e-15 ***
Reason.for.absence10   0.014175   0.810657   0.017  0.98606    
Reason.for.absence11  -1.369550   0.800815  -1.710  0.08782 .  
Reason.for.absence12  -1.321472   1.034230  -1.278  0.20191    
Reason.for.absence13  -1.156118   0.783550  -1.475  0.14069    
Reason.for.absence14  -2.051440   0.973509  -2.107  0.03557 *  
Reason.for.absence15   0.906688   2.032508   0.446  0.65572    
Reason.for.absence16  -5.152222   1.726225  -2.985  0.00297 ** 
Reason.for.absence17  -0.266406   2.810433  -0.095  0.92452    
Reason.for.absence18  -0.212855   0.964498  -0.221  0.82542    
Reason.for.absence19   0.069913   0.836625   0.084  0.93343    
Reason.for.absence2   -3.689146   2.810665  -1.313  0.18991    
Reason.for.absence21  -1.878808   1.386169  -1.355  0.17588    
Reason.for.absence22   0.744656   0.873870   0.852  0.39453    
Reason.for.absence23  -4.241444   0.707267  -5.997 3.76e-09 ***
Reason.for.absence24   1.349307   1.720680   0.784  0.43330    
Reason.for.absence25  -3.944662   0.866458  -4.553 6.60e-06 ***
Reason.for.absence26   0.194742   0.851839   0.229  0.81926    
Reason.for.absence27  -4.942729   0.791357  -6.246 8.76e-10 ***
Reason.for.absence28  -4.210393   0.727683  -5.786 1.24e-08 ***
Reason.for.absence3    2.082272   2.841893   0.733  0.46407    
Reason.for.absence4   -2.477334   2.046659  -1.210  0.22666    
Reason.for.absence5   -0.769424   1.702807  -0.452  0.65156    
Reason.for.absence6   -0.525086   1.294602  -0.406  0.68521    
Reason.for.absence7   -1.899257   1.021108  -1.860  0.06345 .  
Reason.for.absence8   -1.514714   1.391427  -1.089  0.27683    
Reason.for.absence9    1.163486   1.718281   0.677  0.49863    
Month.of.absence10     0.727466   0.642965   1.131  0.25840    
Month.of.absence11     0.608864   0.635187   0.959  0.33823    
Month.of.absence12     0.431670   0.673483   0.641  0.52184    
Month.of.absence2      0.633871   0.596060   1.063  0.28808    
Month.of.absence3      1.677293   0.590072   2.843  0.00465 ** 
Month.of.absence4      0.686616   0.644034   1.066  0.28686    
```

```
Month.of.absence5      0.316097   0.646574   0.489  0.62513
Month.of.absence6      0.153919   0.660423   0.233  0.81581
Month.of.absence7      1.070920   0.628475   1.704  0.08898 .
Month.of.absence8      0.862565   0.675628   1.277  0.20228
Month.of.absence9      0.552985   0.673478   0.821  0.41197
Disciplinary.failure1 -5.655580   0.603759  -9.367  < 2e-16 ***
Pet1                  -0.671385   0.326666  -2.055  0.04035 *
Pet2                  -0.382694   0.373544  -1.024  0.30608
Pet4                   0.009269   0.622415   0.015  0.98812
Pet5                   1.029897   1.432889   0.719  0.47261
Pet8                  -2.310676   1.086549  -2.127  0.03392 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.705 on 521 degrees of freedom
Multiple R-squared:  0.4205,     Adjusted R-squared:  0.3727
F-statistic: 8.792 on 43 and 521 DF,  p-value: < 2.2e-16
```

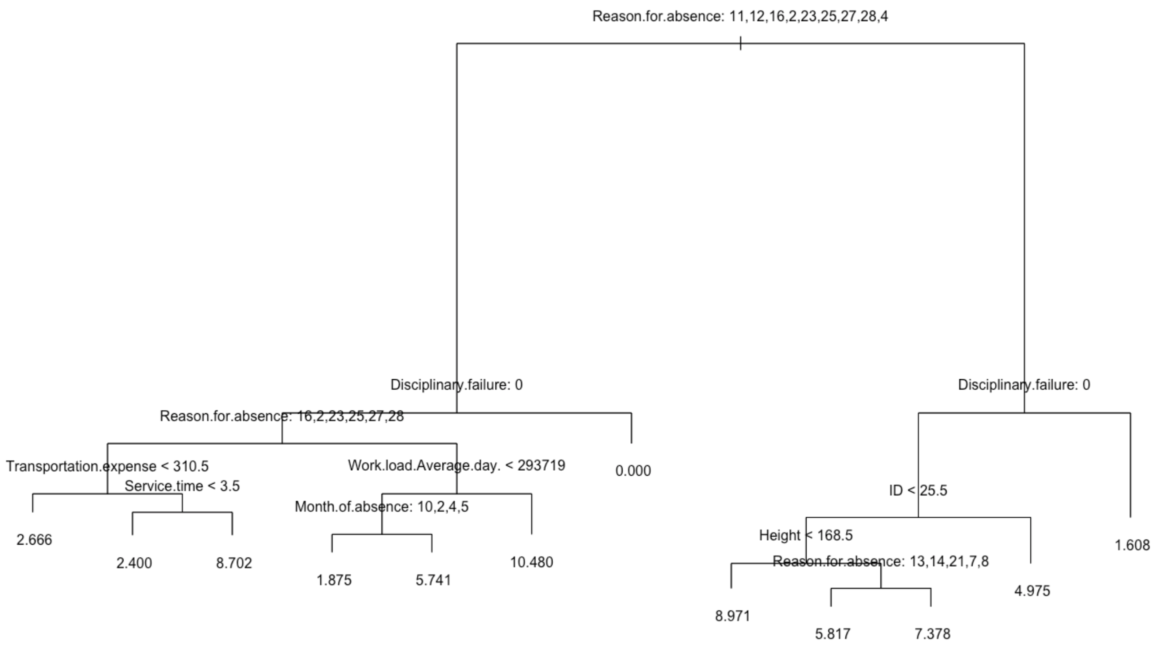
After modelling with the variables which impact the target variables the most we see significant change in the model. Making the model more robust.

2.2.1.2 Decision Tree

```
modelDT = tree(Absenteeism.time.in.hours~.,df,subset = train)
summary(modelDT)
plot(modelDT)
text(modelDT,pretty = 0)
predictDT = predict(modelDT,newdata = df[test,])
```

```
Regression tree:
tree(formula = Absenteeism.time.in.hours ~ ., data = df, subset =
train)
Variables actually used in tree construction:
[1]    "Reason.for.absence"                    "Disciplinary.failure"
"Transportation.expense" "Service.time"
```

```
[5]  "Work.load.Average.day."  "Month.of.absence"              "ID"
"Height"
Number of terminal nodes:  12
Residual mean deviance:  6.059 = 3350 / 553
Distribution of residuals:
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-6.3780 -1.6080 -0.6660  0.0000  0.6224 13.3300
```

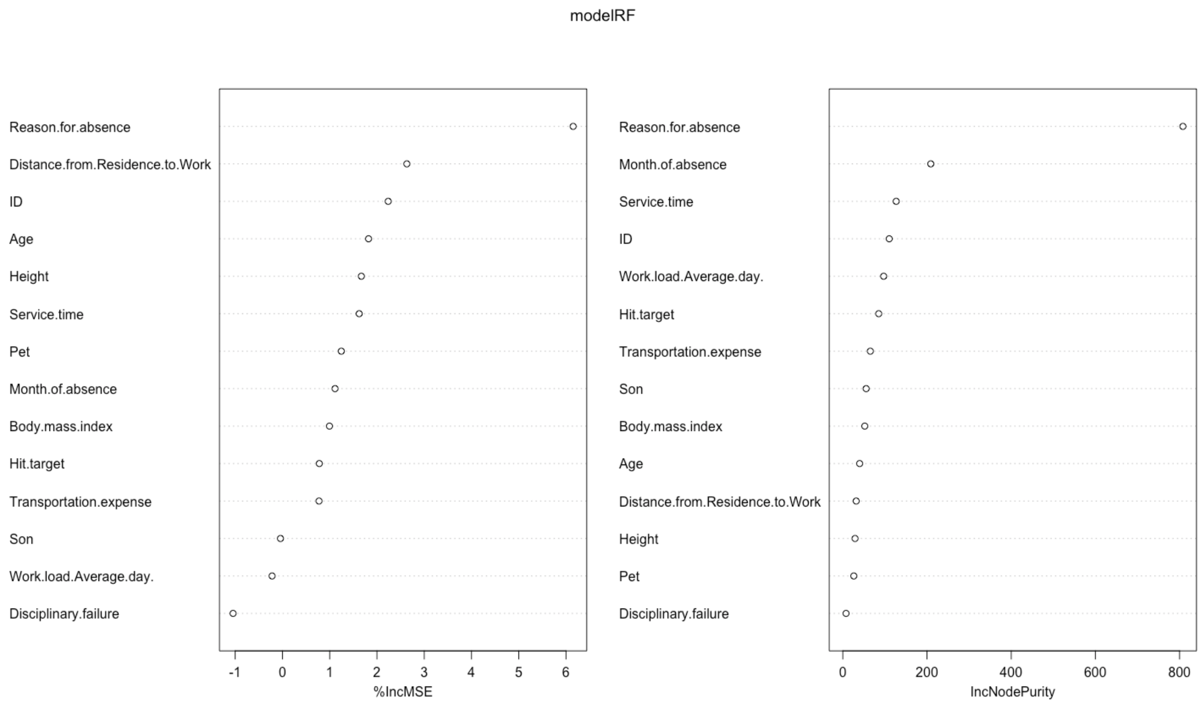


2.2.1.3 Random Forest

```
modelRF = randomForest(Absenteeism.time.in.hours~.,data = df,subset
= test,mtry = 12,ntree=12,importance = TRUE)
varImpPlot(modelRF)
importance(modelRF)
predictRF = predict(modelRF,newdata = df[test,])
```

```
                                %IncMSE  IncNodePurity
ID                            2.23765717    110.218487
Reason.for.absence            6.15057698    807.835873
Month.of.absence              1.11301756    208.917724
Transportation.expense        0.77381296     65.339205
Distance.from.Residence.to.Work  2.63215254     31.783604
Service.time                  1.62401044    126.778921
Age                           1.82191096     39.974298
Work.load.Average.day.       -0.21866419     96.976921
```

```
Hit.target                         0.78020104     85.241522
Disciplinary.failure              -1.04446594      7.661065
Son                               -0.04137263     55.414339
Pet                                1.24590620     26.043207
Height                             1.66951682     29.025098
Body.mass.index                    0.99471084     52.073015
```



Chapter 3

Conclusion

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. Since our models are regression type so we will measure using the following methods.

3.1.1 RMSE (Root Mean Squared Error)

```
sqrt(mean((predictLR-df$Absenteeism.time.in.hours[test])^2))
```

Linear Regression – 2.94

Decision Tree – 3.06

Random Forest – 1.42

3.2 Model Conclusion

From the result of the three models used above it is very clear that random Forest has the best prediction. Thus, by using Random forest for prediction we can see that Reason for Absence has the most importance in predicting the model and also, we have seen that out of the Reason of Absence medical reasons are occurring the most (from data visualization).

What changes company should bring to reduce the number of absenteeism?

On summarising the Count, Sum of Absenteeism hours and Mean of absenteeism hours Reason wise, we see that medical consultation and dental consultation are the most common cause of Absenteeism. Hence the company should work to fix this issue.

How much losses every month can we project in 2011 if same trend of absenteeism continues?

Work Loss = (Work.load.Average.day/Service.time)*Absenteeism.time.in.hours

Month	WorkLoss
1	4824824
10	7798330
11	7425217
12	7955108
2	7967229
3	10968466
4	6358008
5	6863317
6	10151872
7	11640825
8	7064595
9	4380616

Appendix (R-code)

```
#Clear the environment

rm(list = ls())


#Set working Directory

setwd("/Users/raghavkotwal/Documents/Data Science/Employee Absenteeism")

getwd()


#Load the librarires

libraries = c("dummies","caret","rpart.plot","plyr","dplyr", "ggplot2","rpart","dplyr","DMwR","randomForest","usdm","corrgram","DataCombine","xlsx","tree")

lapply(X = libraries,require, character.only = TRUE)

rm(libraries)


#Read the Data

df = read.xlsx(file = "Absenteeism_at_work.xls", header = T, sheetIndex = 1)



######Explore Data#######


#Check Dimensions of Data

dim(df)


#Check Structure of Variables

str(df)


#view the top 5 rows of the data
```

```
head(df)


#Transform required data types into categorical data
df$Reason.for.absence[df$Reason.for.absence %in% 0] = NA
df$Reason.for.absence = as.factor(as.character(df$Reason.for.absence))


df$Month.of.absence[df$Month.of.absence %in% 0] = NA
df$Month.of.absence = as.factor(as.character(df$Month.of.absence))


df$Day.of.the.week = as.factor(as.character(df$Day.of.the.week))
df$Seasons = as.factor(as.character(df$Seasons))
df$Disciplinary.failure = as.factor(as.character(df$Disciplinary.failure))
df$Education = as.factor(as.character(df$Education))
df$Son = as.factor(as.character(df$Son))
df$Social.drinker = as.factor(as.character(df$Social.drinker))
df$Social.smoker = as.factor(as.character(df$Social.smoker))
df$Pet = as.factor(as.character(df$Pet))




##########MISSING VALUE ANALYSIS#############


sapply(df,function(x){sum(is.na(x))})
missing_values = data.frame(sapply(df,function(x){sum(is.na(x))}))


#Calculate missing percentage and arrage in order
missing_values$Var = row.names(missing_values)
```

```
row.names(missing_values) = NULL

names(missing_values)[1] = "Percentage"

missing_values$Percentage = ((missing_values$Percentage/nrow(df)) *100)

missing_values = missing_values[,c(2,1)]

missing_values = missing_values[order(-missing_values$Percentage),]


#Create missing value and impute using mean, median and knn

#Value = 31

#Mean = 26.68

#Median = 25

#KNN = 31

#df1[["Body.mass.index"]][6]

#df1[["Body.mass.index"]][6] = NA

#df1[["Body.mass.index"]][6] = mean(df$Body.mass.index, na.rm = T)

#df1[["Body.mass.index"]][6] = median(df$Body.mass.index, na.rm = T)


df = knnImputation(data = df, k = 5)


#Check if any missing values

sum(is.na(df))


#Check Structure of Variables

str(df)

df1 = df
```

```
####### Data visualisation #####

library(ggplot2)

library(corrplot)

#Definig variable types

numerical_set = c("ID","Transportation.expense","Distance.from.Residence.to.Work","Service.time","Age","Work.load.Average.day.","Hit.target","Son","Pet","Height","Weight","Body.mass.index","Absenteeism.time.in.hours")

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

## plotting numerical data set vs the target variable

plot(df$ID,df$Absenteeism.time.in.hours,x = "ID",ylab = "Absenteeism time in hours",main = "Absenteeism time vs ID",col="Black")

plot(df$Transportation.expense,df$Absenteeism.time.in.hours,xlab = "Transportation expense",ylab = "Absenteeism time in hours",main = "Absent time vs Transp expense",col="Black")

plot(df$Distance.from.Residence.to.Work,df$Absenteeism.time.in.hours,xlab = "Distance.from.Residence.to.Work",ylab = "Absenteeism time in hours",main = "Absent time vs Travel distance",col="Black")

plot(df$Service.time,df$Absenteeism.time.in.hours,xlab = "Service Time",ylab = "Absenteeism time in hours",main = "Absenteeism time vs Service time",col="Black")

plot(df$Age,df$Absenteeism.time.in.hours,xlab = "Age",ylab = "Absenteeism time in hours",main = "Absenteeism time vs Age",col="Black")

plot(df$Work.load.Average.day.,df$Absenteeism.time.in.hours,xlab = "Work.load.Average.day",ylab = "Absenteeism time in hours",main = "Absent time vs Work.load.Avg.day",col="Black")

plot(df$Hit.target,df$Absenteeism.time.in.hours,xlab = "Hit target",ylab = "Absenteeism time in hours",main = "Absenteeism time vs Hit target",col="Black")
```

```
plot(df$Son,df$Absenteeism.time.in.hours,xlab = "Son",ylab = "Absenteeism time in hours",main = "Absenteeism time vs Son",col="Black")

plot(df$Pet,df$Absenteeism.time.in.hours,xlab = "Pet",ylab = "Absenteeism time in hours",main = "Absenteeism time vs Pet",col="Black")

plot(df$Height,df$Absenteeism.time.in.hours,xlab = "Height",ylab = "Absenteeism time in hours",main = "Absenteeism time vs Height",col="Black")

plot(df$Weight,df$Absenteeism.time.in.hours,xlab = "Weight",ylab = "Absenteeism time in hours",main = "Absenteeism time vs Weight",col="Black")

plot(df$Body.mass.index,df$Absenteeism.time.in.hours,xlab = "BMI",ylab = "Absenteeism time in hours",main = "Absenteeism time vs BMI",col="Black")




##plotting categorical data set vs target variable

dev.off()

for(i in 1:length(categorical_set)){


assign(paste0("gg",i),ggplot(aes_string(y=df$Absenteeism.time.in.hours,x=df[,categorical_set[i]]),data = subset(df))

         + stat_boxplot(geom = "errorbar",width = 0.3) +

           geom_boxplot(outlier.colour = "red",fill = "black",outlier.shape = 18,outlier.size = 1) +

           labs(y = "Absenteeism.time.in.hours",x=names(df[categorical_set[i]])) +

           ggtitle(names(df[categorical_set[i]])))



}

gridExtra::grid.arrange(gg1,gg2,nrow = 2,ncol=1)

gridExtra::grid.arrange(gg3,gg4,nrow = 2,ncol = 1)

gridExtra::grid.arrange(gg5,gg6,nrow = 2,ncol = 1)

gridExtra::grid.arrange(gg7,gg8,nrow = 2,ncol = 1)
```

```
####Outlier Analysis#####

## Replace outliers in numerical dataset with NAs using boxplot method
for(i in numerical_set){
  outlier_value = boxplot.stats(df[,i])$out
  print(names(df[i]))
  print(outlier_value)
  df[which(df[,i] %in% outlier_value),i] = NA
}

#Compute the NA values using KNN imputation
df = knnImputation(df, k = 5)

#Check if any missing values
sum(is.na(df))



######Feature Selection######

## Correlation Plot
corrgram(df[,numerical_set], order = F,
         upper.panel=panel.pie, text.panel=panel.txt, main = "Correlation Plot")

## ANOVA test for Categprical variable
summary(aov(formula = Absenteeism.time.in.hours~ID,data = df))
```

```
summary(aov(formula = Absenteeism.time.in.hours~Reason.for.absence,data = df))

summary(aov(formula = Absenteeism.time.in.hours~Month.of.absence,data = df))

summary(aov(formula = Absenteeism.time.in.hours~Day.of.the.week,data = df))

summary(aov(formula = Absenteeism.time.in.hours~Seasons,data = df))

summary(aov(formula = Absenteeism.time.in.hours~Disciplinary.failure,data = df))

summary(aov(formula = Absenteeism.time.in.hours~Education,data = df))

summary(aov(formula = Absenteeism.time.in.hours~Social.drinker,data = df))

summary(aov(formula = Absenteeism.time.in.hours~Social.smoker,data = df))

summary(aov(formula = Absenteeism.time.in.hours~Son,data = df))

summary(aov(formula = Absenteeism.time.in.hours~Pet,data = df))



## Dimension Reduction

df = subset(df, select = -(which(names(df) %in% c("Weight","Day.of.the.week","Seasons","Education","Social.smoker","Social.drinker"))))



## Using createdataPartition for sampling we create 75% train 25% test data set using variable reason for Absence

train = createDataPartition(df$Reason.for.absence,times = 1,p = 0.75,list = F)

test = -(train)



#####Model Development#####

## Linear regression

modelLR = lm(Absenteeism.time.in.hours~.,data = df[train,])

summary(modelLR)

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

par(mfrow = c(3,1))

plot(modelLR)
```

```
predictLR = predict(modelLR,df[test,])

sqrt(mean((predictLR-df$Absenteeism.time.in.hours[test])^2))

##RMSE : 2.94




## Decision trees

modelDT = tree(Absenteeism.time.in.hours~.,df,subset = train)

summary(modelDT)

plot(modelDT)

text(modelDT,pretty = 0)

predictDT = predict(modelDT,newdata = df[test,])

sqrt(mean((predictDT-df$Absenteeism.time.in.hours[test])^2))

##RMSE 3.06



## Random Forest

modelRF = randomForest(Absenteeism.time.in.hours~.,data = df,subset = test,mtry =
12,ntree=12,importance = TRUE)

varImpPlot(modelRF)

importance(modelRF)

predictRF = predict(modelRF,newdata = df[test,])

sqrt(mean((predictRF-df$Absenteeism.time.in.hours[test])^2))

##RMSE 1.44



#### Conclusion
```

```r
## Sum and mean of the absenteeism hours reason wise

reason_sum_hrs = aggregate(df$Absenteeism.time.in.hours,by = list(Category = df$Reason.for.absence),FUN = sum)

names(reason_sum_hrs)=c("Reason no.","Sum of absent hours")

reason_mean_hrs = aggregate(df$Absenteeism.time.in.hours,by = list(Category = df$Reason.for.absence),FUN = mean)

names(reason_mean_hrs)=c("Reason no.","Mean of absent hours")

table(df$Reason.for.absence)



## Monthly loss for the Company



loss_data = df[,c("Month.of.absence","Work.load.Average.day.","Service.time","Absenteeism.time.in.hours")]

str(loss_data)

loss_data$WorkLoss = round((loss_data$Work.load.Average.day./loss_data$Service.time)*loss_data$Absenteeism.time.in.hours)

View(loss_data)

monthly_loss = aggregate(loss_data$WorkLoss,by = list(Category = loss_data$Month.of.absence),FUN = sum)

names(monthly_loss) = c("Month","WorkLoss")
```