Employee Absenteeism

Radhika Haresh Luvani

17/12/2018

Contents

Chapter 1

Introduction

1.1 Problem Statement

The high competitiveness in the market, professional development combined with the development of organizations and the pressure to reach increasingly audacious goals, create increasingly overburdened employees and end up acquiring some disturbance in the state of health related to the type of work activity, including depression considered the evil of the 21st century. Taking employees to absenteeism. Absenteeism is defined as absence to work as expected, represents for the company the loss of productivity and quality of work.[1]
Employee absenteeism at work is a genuine issue and need to figure out what are the reason and circumstances for the absenteeism. As we appreciate that human capital plays an important role in collection, transportation and delivery.
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 Sabsenteeism continues?

1.2 Data

Our task is to build absenteeism models which will predict the next year or coming year forecasting depending on multiple predictors. We have the same data set. Given below is a sample of the data set that we are using to predict the trend of absenteeism.

Table 1.2.1: Sample Data (Columns: 1-8)

ID	Reason.for.absence	Month.of.absence	Body.mass.index	Absenteeism.time.in.hours	Year	Day.of.the.week	Seasons
11	26	7	30	4	2007	3	1
36	0	7	31	0	2007	3	1
3	23	7	31	2	2007	4	1
7	7	7	24	4	2007	5	1
11	23	7	30	2	2007	5	1
3	23	7	31	NA	2007	6	1
10	22	7	27	8	2007	6	1
20	23	7	23	4	2007	6	1
14	19	7	25	40	2007	2	1
1	22	7	29	8	2007	2	1

Table 1.2.2: Sample Data (Columns: 9-14)

Transportation.expense	Distance.from.Residence.to.Work	Service.time	Age	Work.load.Average.day	Hit.target
289	36	13	33	239,554	97
118	13	18	50	239,554	97
179	51	18	38	239,554	97
279	5	14	39	239,554	97
289	36	13	33	239,554	97
179	51	18	38	239,554	97
NA	52	3	28	239,554	97
260	50	11	36	239,554	97
155	12	14	34	239,554	97
235	11	14	37	239,554	97

Table 1.2.2: Sample Data (Columns: 15-22)

Disciplinary.failure	Education	Son	Social.drinker	Social.smoker	Pet	Weight	Height
0	1	2	1	0	1	90	172
1	1	1	1	0	0	98	178
0	1	0	1	0	0	89	170
0	1	2	1	1	0	68	168
0	1	2	1	0	1	90	172
0	1	0	1	0	0	89	170
0	1	1	1	0	4	80	172
0	1	4	1	0	0	65	168
0	1	2	1	0	0	95	196
0	3	1	0	0	1	88	172

As you can see in the table below we have the following 22 variables, using which we have to correctly predict the customer behavior:

Table 1.2.3: Predictor Variables

S.No.	Predictor
1	ID
2	Reason for absence
3	Month of absence
4	Year
5	Day of the week
6	Seasons
7	Transportation expense
8	Distance from Residence to Work
9	Service time
10	Age
11	Work load Average day
12	Hit target
13	Disciplinary failure
14	Education
15	Son
16	Social drinker
17	Social smoker
18	Pet
19	Weight
20	Height
21	Body mass index
22	Absenteeism time in hours

Chapter 2

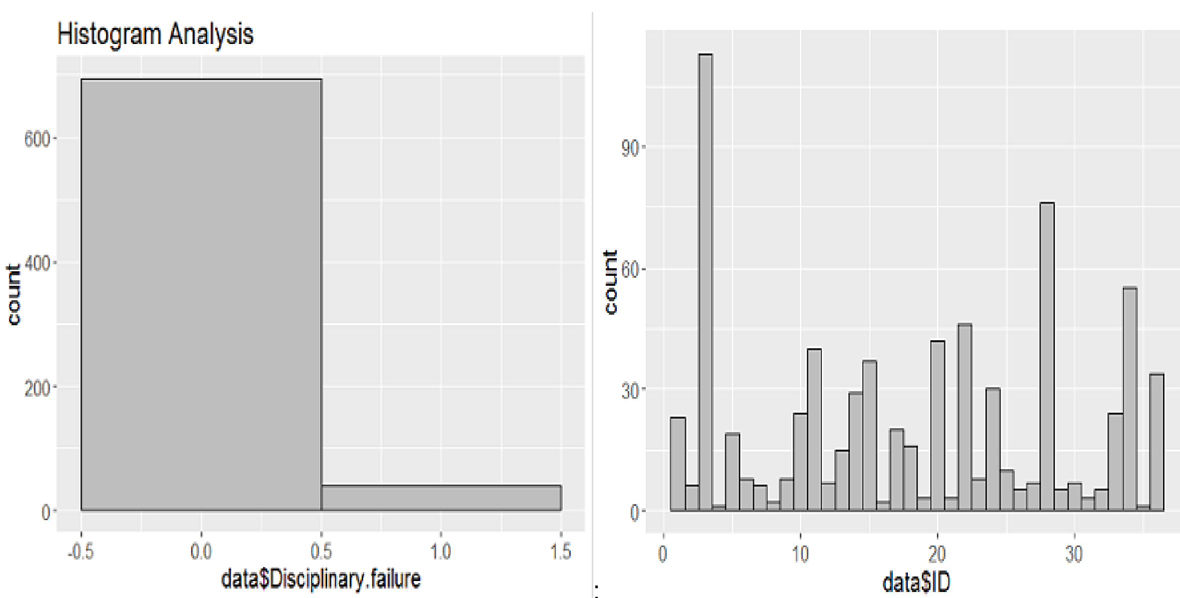
Methodology

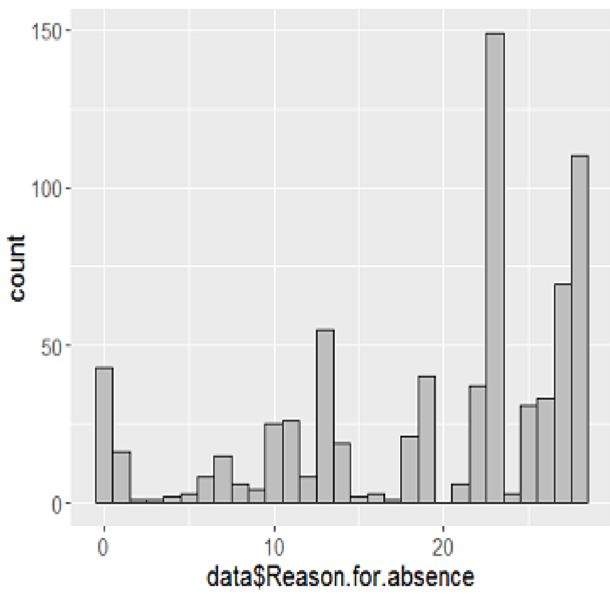
2.1 Pre Processing

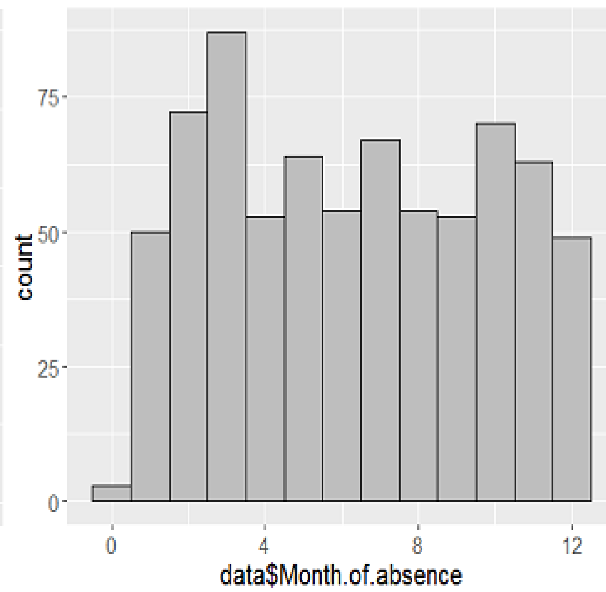
Any predictive modeling requires that we look at the data before we start modeling. However, in data mining terms looking *at* data refers to so much more than just looking. Looking at data refers to exploring the data, cleaning the data as well as visualizing the data through graphs and plots. This is often called as Exploratory Data Analysis. To start this process we will first try and look at all the probability distributions of the variables. Most analysis like regression, require the data to be normally distributed. We can visualize that in a glance by looking at the probability distributions or probability count functions of the variable.

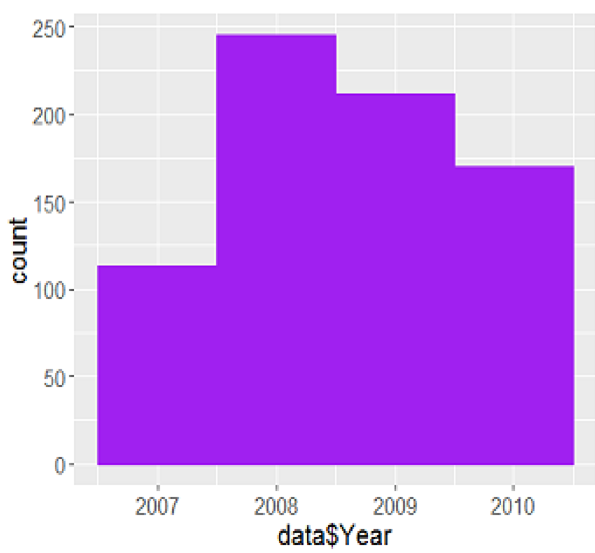
In the fig (2.1) it’s showing an analysis of the individual predictors and its count with the help of Histogram Analysis.

Fir 2.1 Predictors histogram analysis

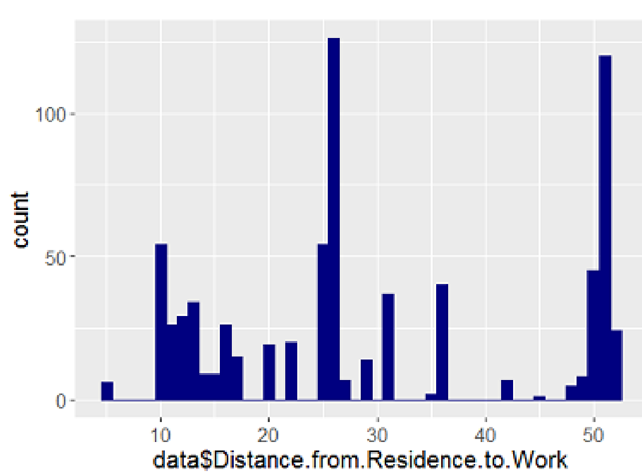


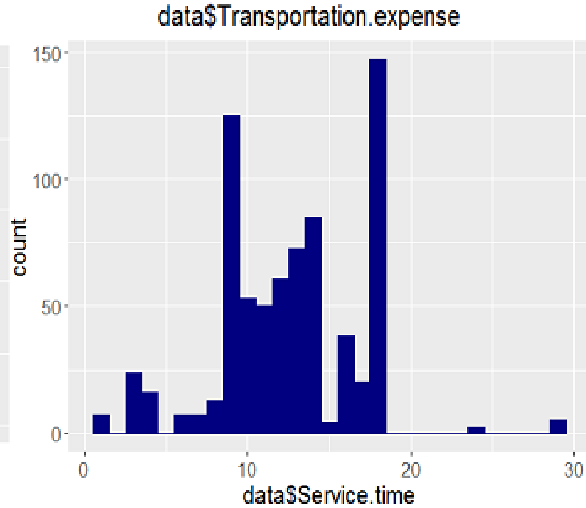

150
100
50
0
count
0
10
20
data$Reason.for.absence

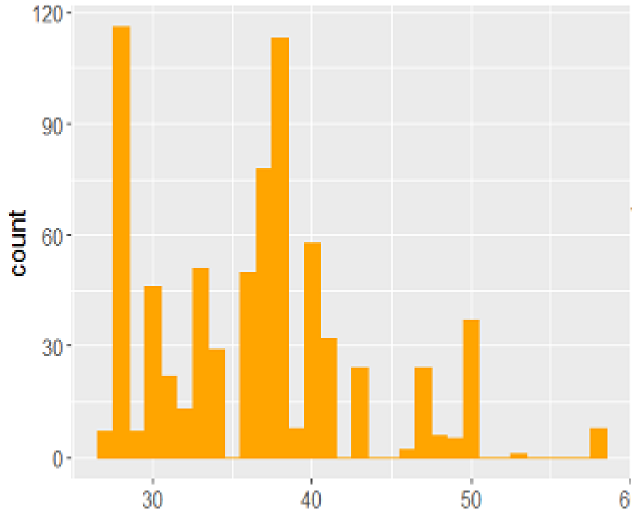

75
50
25
0
count
0
4
8
12
data$Month.of.absence

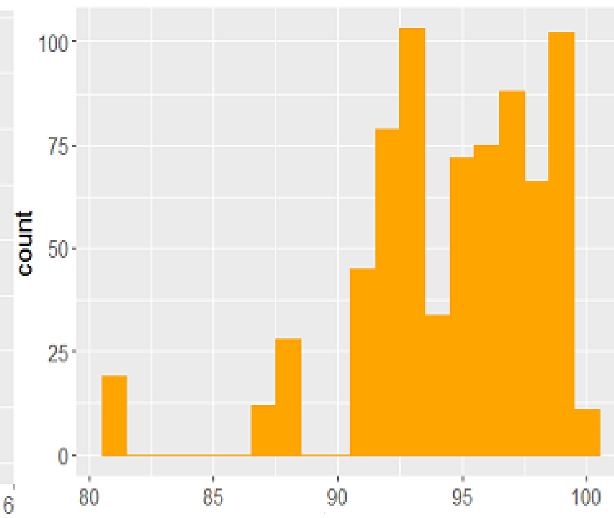

250
200
150
100
50
0
count
2007
2008
2009
2010
data$Year

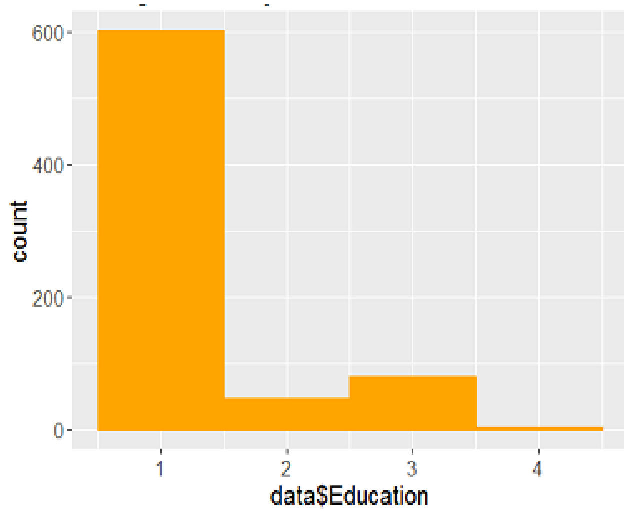

150
100
50
0
count
200
300
data$Transportation.expense

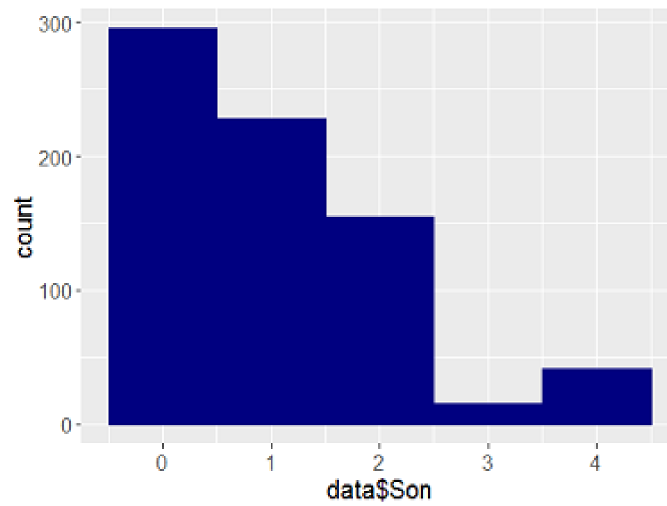

100
50
0
count
10
20
30
40
50
data$Distance.from.Residence.to.Work

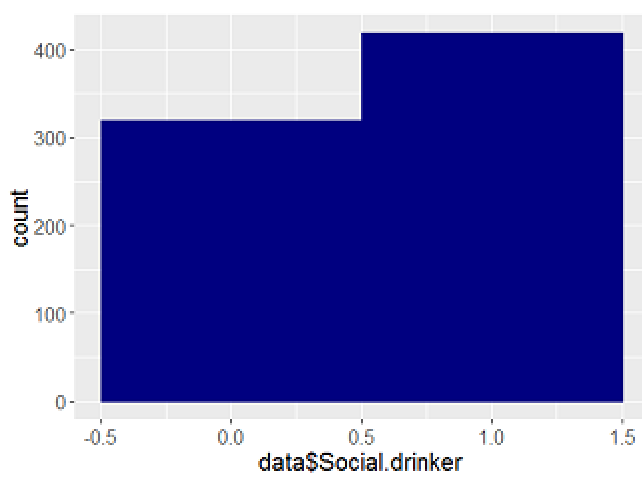

150
100
50
0
count
0
10
20
30
data$Service.time

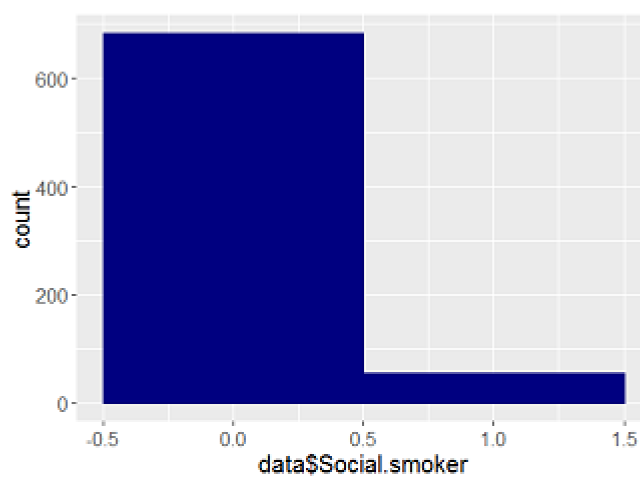

count
120
90
60
30
0
30
40
50
6

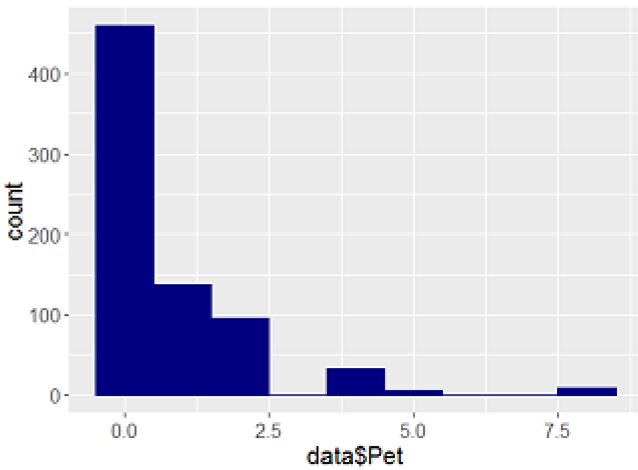

count
100
75
50
25
0
80
85
90
95
100

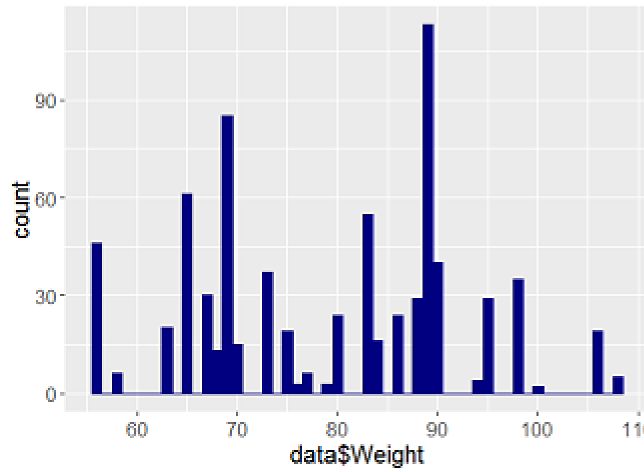

count
600
400
200
0
1
2
3
4
data$Education

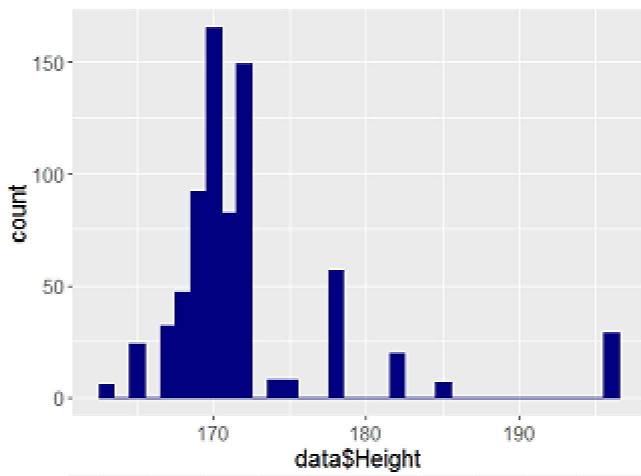

count
300
200
100
0
0
1
2
3
4
data$Son

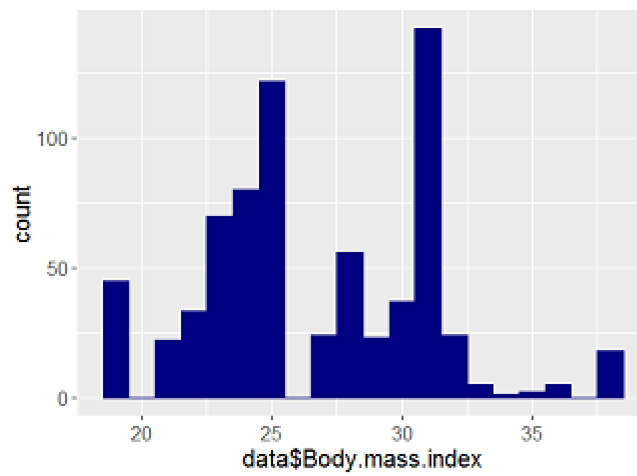

count
400
300
200
100
0
-0.5
0.0
0.5
1.0
1.5
data$Social.drinker

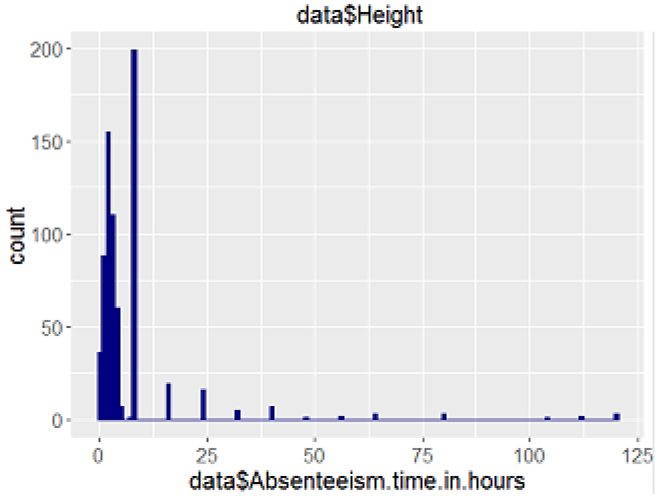

count
600
400
200
0
-0.5
0.0
0.5
1.0
1.5
data$Social.smoker


count
data$Pet


count
data$Weight

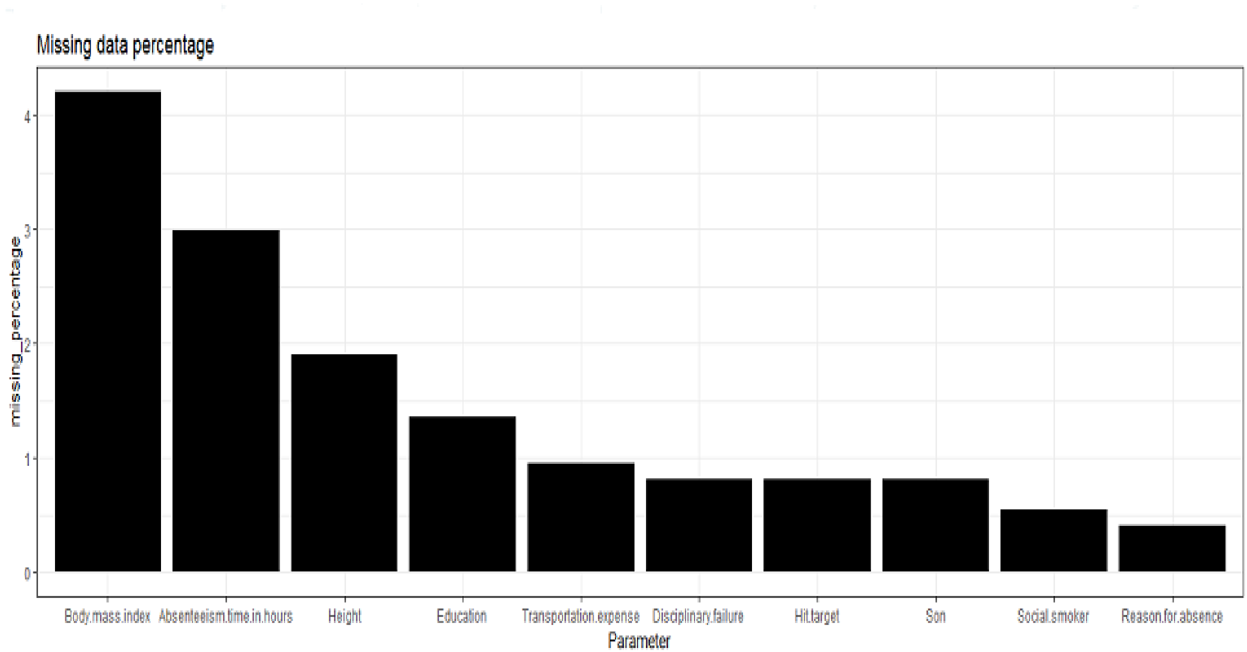

count
data$Height


count
data$Body.mass.index


count
data$Absenteeism.time.in.hours

2.1.1 Missing Value Analysis

A missing value can signify a number of different things in your data. Data mining methods vary in the way they treat missing values. Typically, they ignore the missing values, or exclude any records containing missing values, or replace missing values with the mean, or infer missing values from existing values.

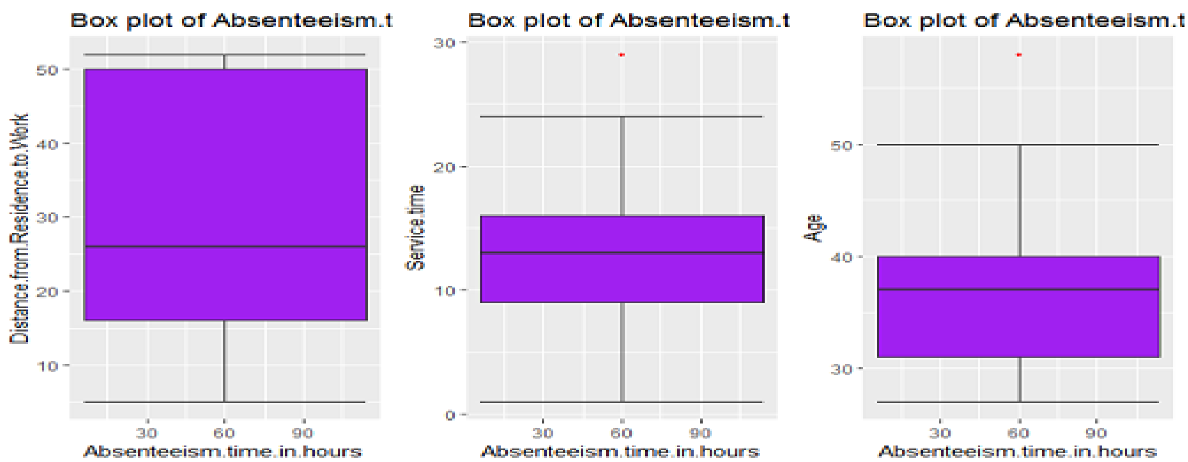


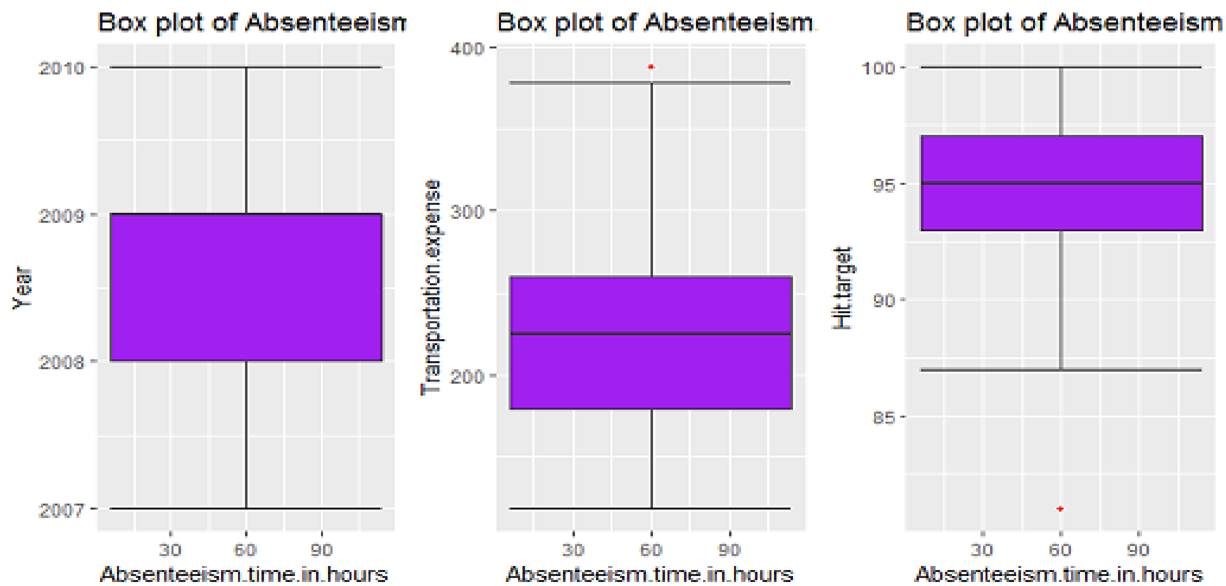
2.1.2 Outlier Analysis

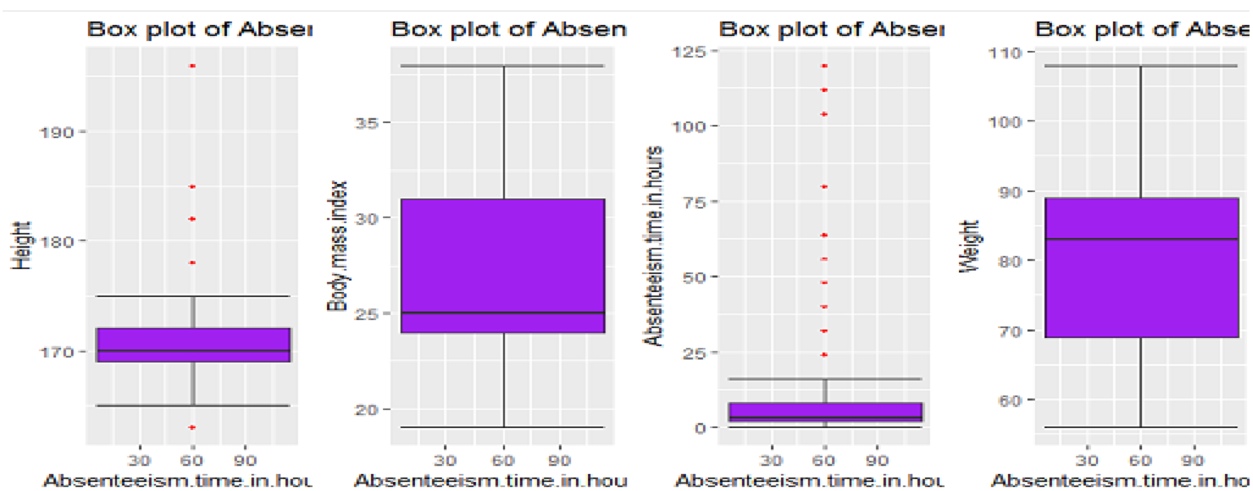
An outlier is an observation point that is distant from other observations. An outlier may be due to variability in the measurement or it may indicate experimental error; the latter are sometimes excluded from the data set.

Observations inconsistent with rest of the dataset Global Outlier. Fig (2.1.2.1) will show the effect of outliers in each predictor, with the help of boxplot. In figure 2.1.2.1 we have plotted the boxplots of the 16 predictor variables with respect to absenteeism. A lot of useful inferences can be made from these plots. First as you can see, we have a lot of outliers and extreme values in each of the data set.

Figure 2.1.2.1: Outliers in each predictor


Box plot of Absenteeism.t
Distance.from.Residence.to.Work
Absenteeism.time.in.hours
Box plot of Absenteeism.t
Service.time
Absenteeism.time.in.hours
Box plot of Absenteeism.t
Age
Absenteeism.time.in.hours


Box plot of Absenteeisn
Year
Absenteeism.time.in.hours
Box plot of Absenteeism
Transportation.expense
Absenteeism.time.in.hours
Box plot of Absenteeism
Hit.target
Absenteeism.time.in.hours


Box plot of Absei
Height
Absenteeism.time.in.hou
Box plot of Absen
Body.mass.index
Absenteeism.time.in.hou
Box plot of Absei
Absenteeism.time.in.hours
Absenteeism.time.in.hou
Box plot of Abse
Weight
Absenteeism.time.in.ho

2.1.3 Feature Selection

Before performing any type of modeling we need to assess the importance of each predictor variable in our analysis. There is a possibility that many variables in our analysis are not important at all to the problem of class prediction. There are several methods of doing that. Below we have used correlation plot.

Data Preparation and Feature Selection In this study used publically available dataset. On other hand, the feature selection is an important step in knowledge discovery process, to identify those relevant variables or attributes from the large number of attributes in a dataset which are too relevant and reduce the computational cost [2]. To make sure that only relevant features are included into decision table that also reduces the computational Cost and address to P1 (Which features are more indicative for employee absenteeism.)

The selection of most appropriate attributes from the dataset.

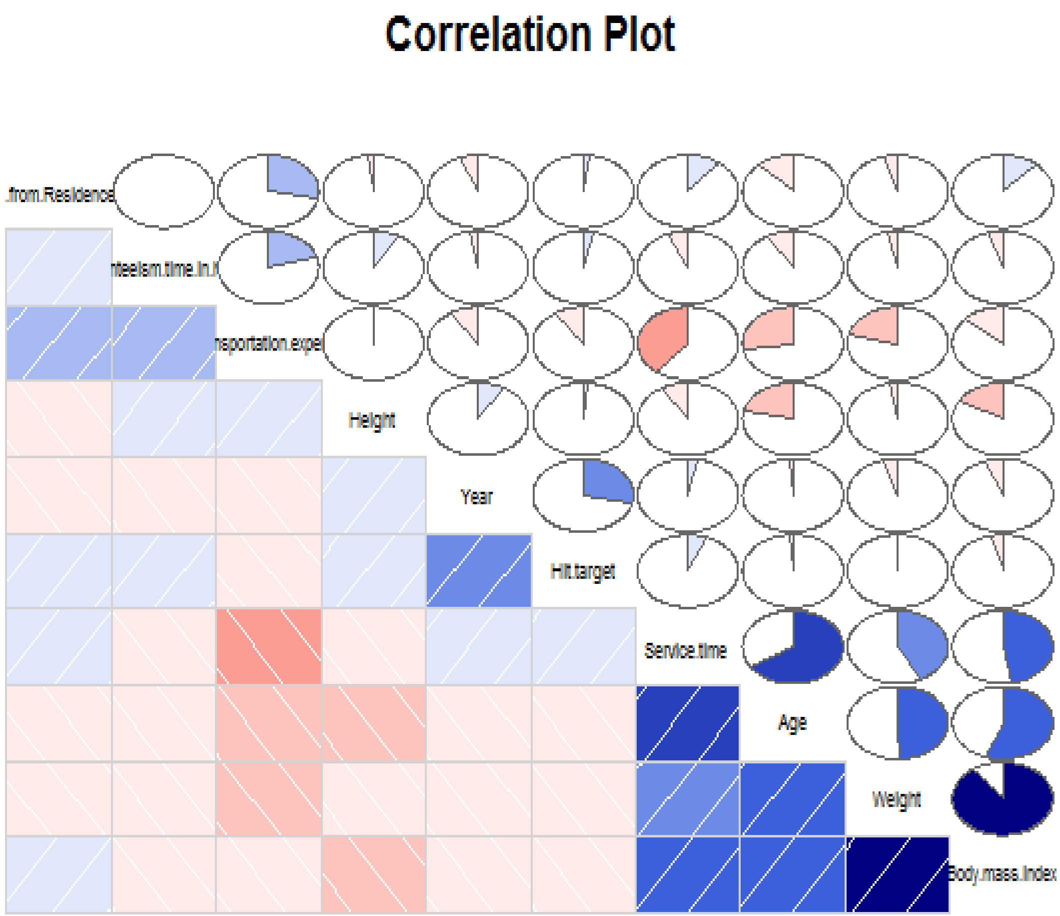


Fig 2.1.3.1 Feature Selection

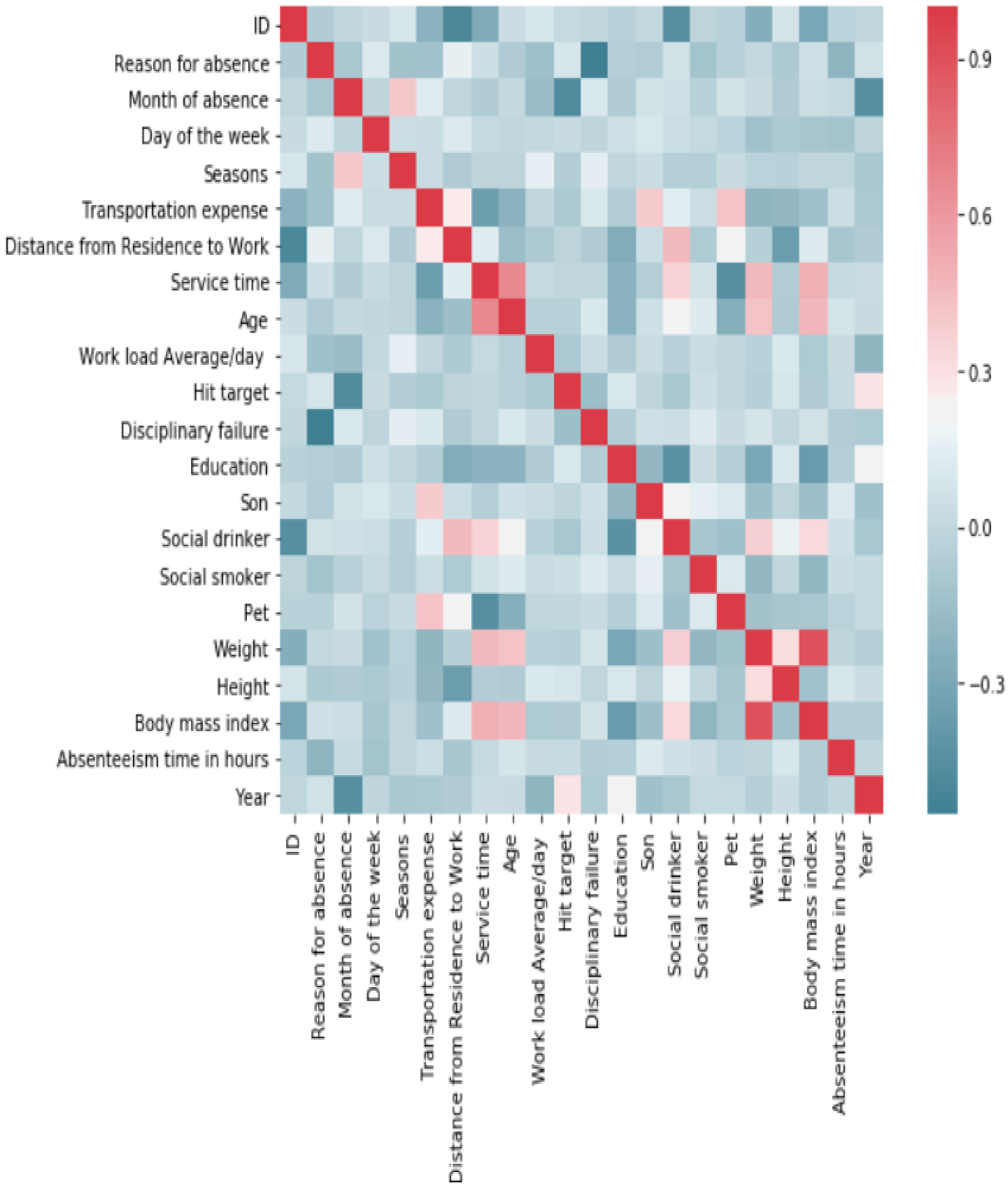
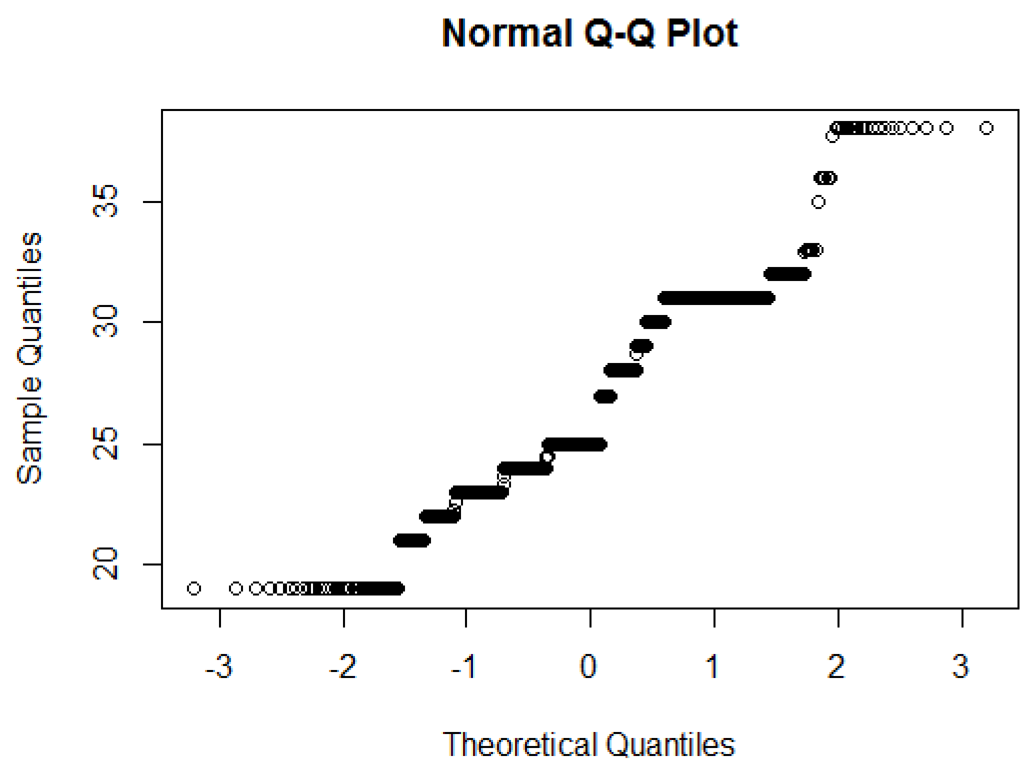
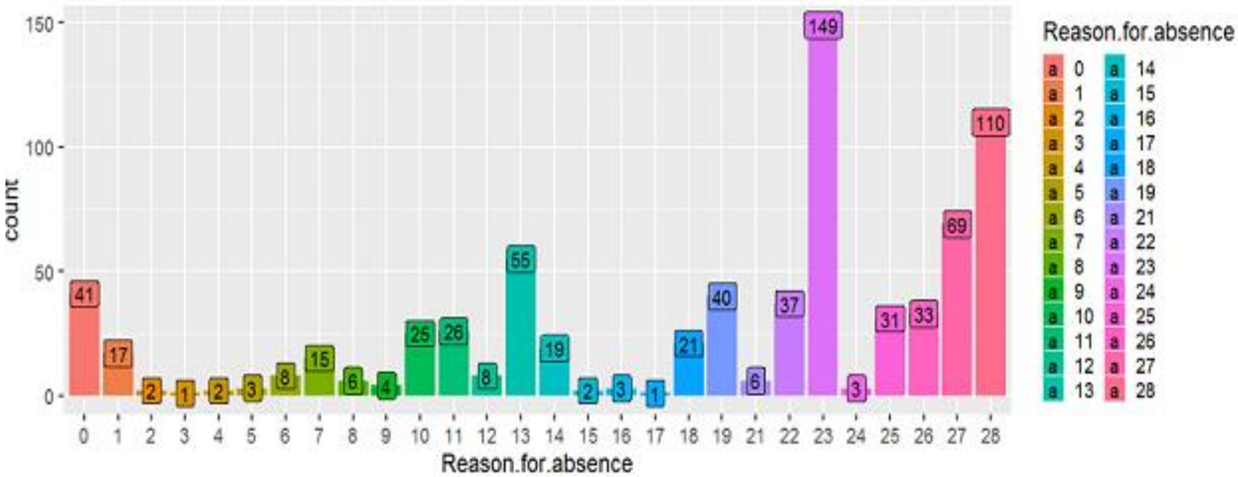


Fig 2.1.3.2 Feature Selection

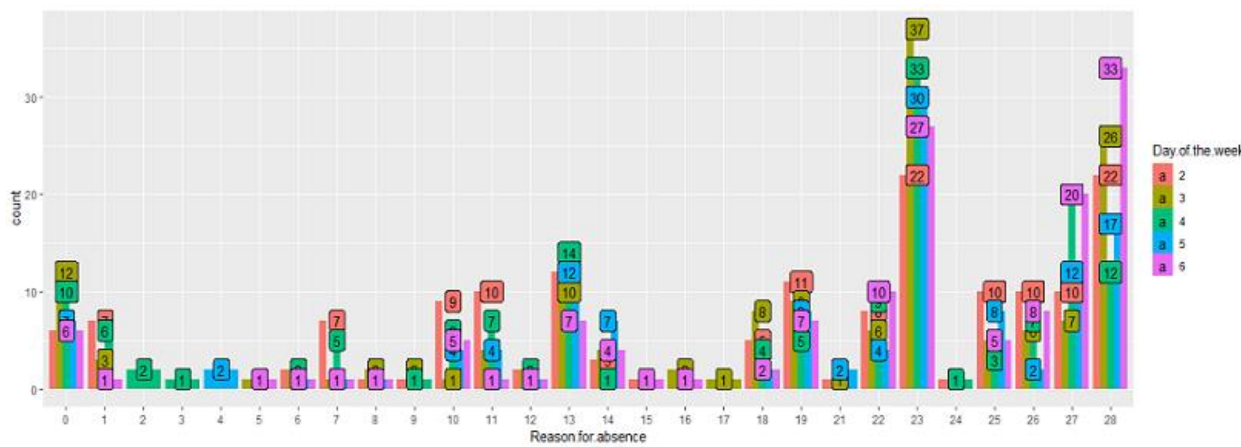

Histogram of data$Absenteeism.time.in.hours
Frequency
0
50
100
150
200
250
0
5
10
15
data$Absenteeism.time.in.hours


Normal Q-Q Plot
Sample Quantiles
20
25
30
35
-3
-2
-1
0
1
2
3
Theoretical Quantiles

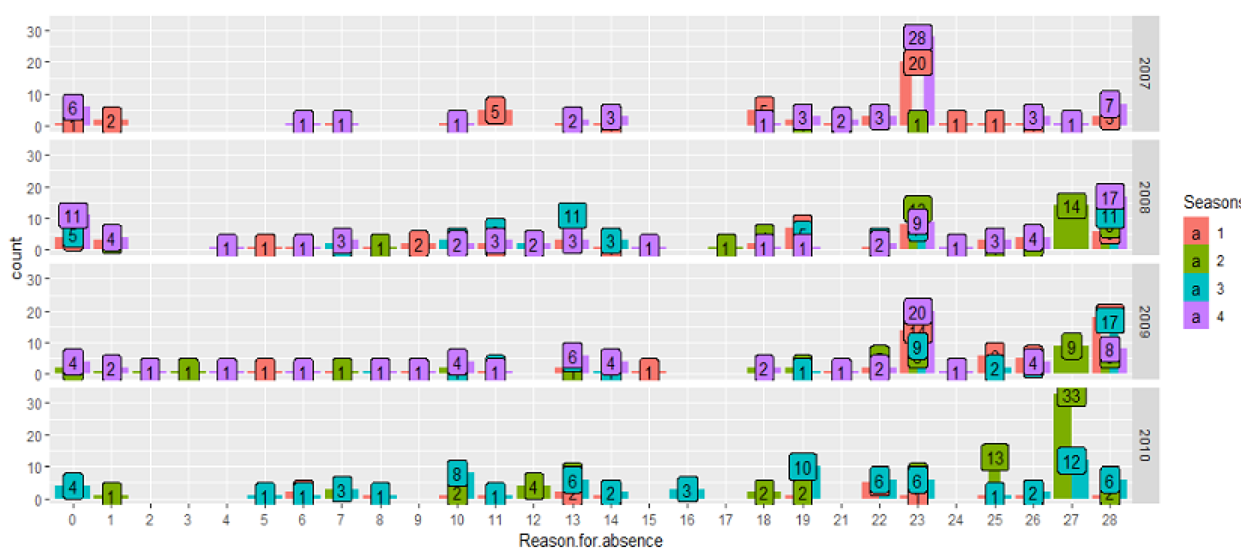
Count of Reason of absence based on Reason of absence



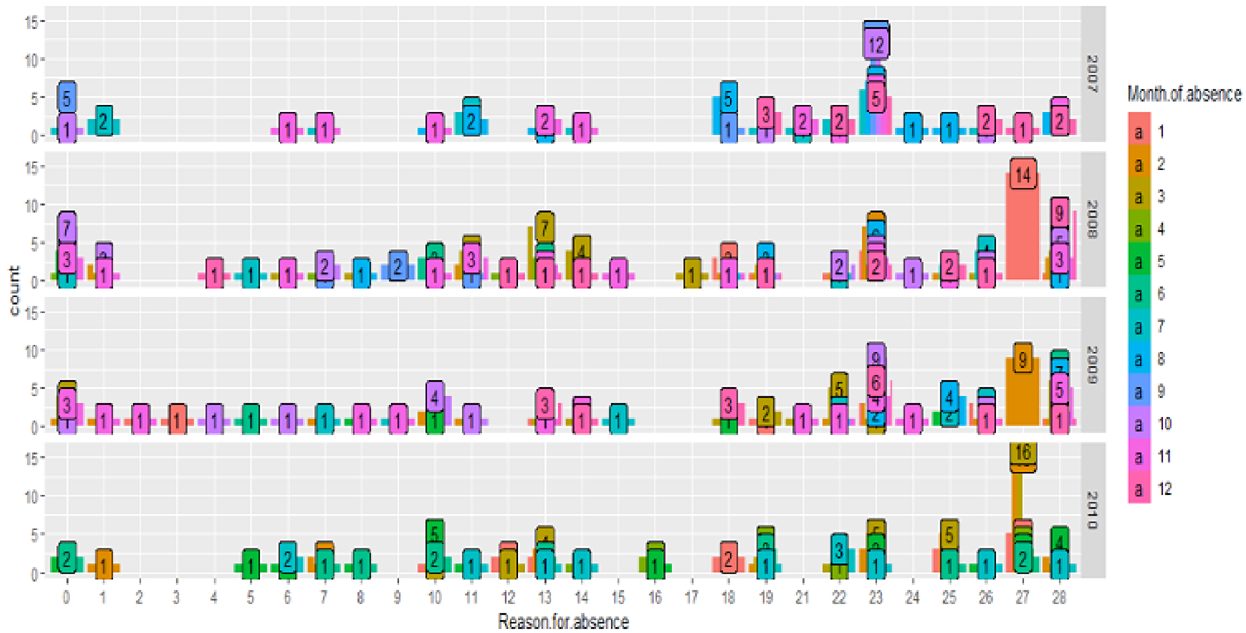
Count of Reason on basis of day of week



Count of Reason on basis of season



Count of Reason on basis of season



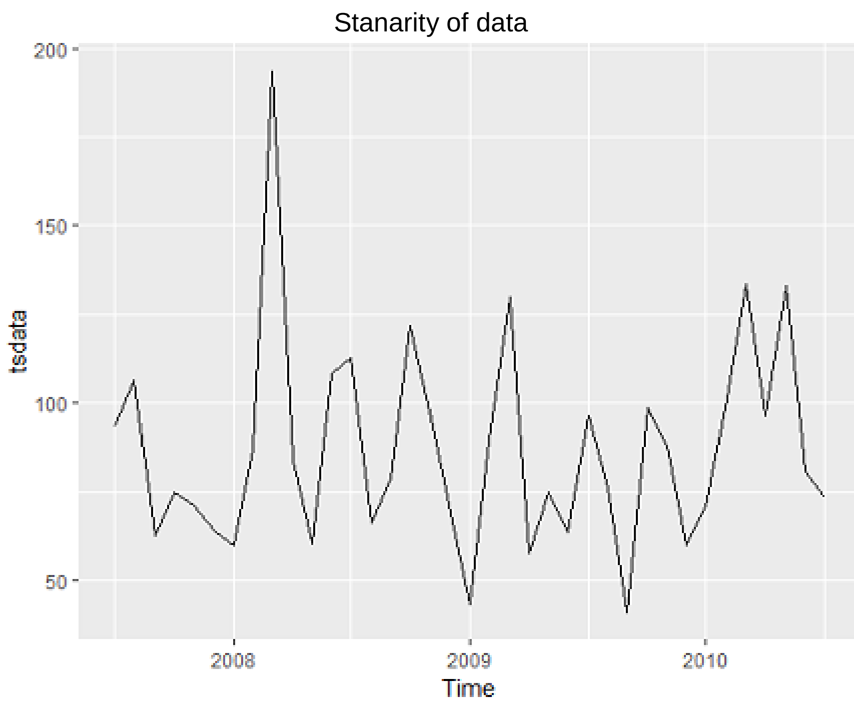
2.2 Modeling

2.2.1 Model Selection

In our early stages of analysis during pre-processing we have come to understand that data behaves the same way. Generate the models for the given data.
The dependent variable can fall in forecasting category:

- **Linear Regression with Trend/TSLM**
- **ARIMA**

Before applying model over data first we need to check the data and its trend and Stationarity of the data.

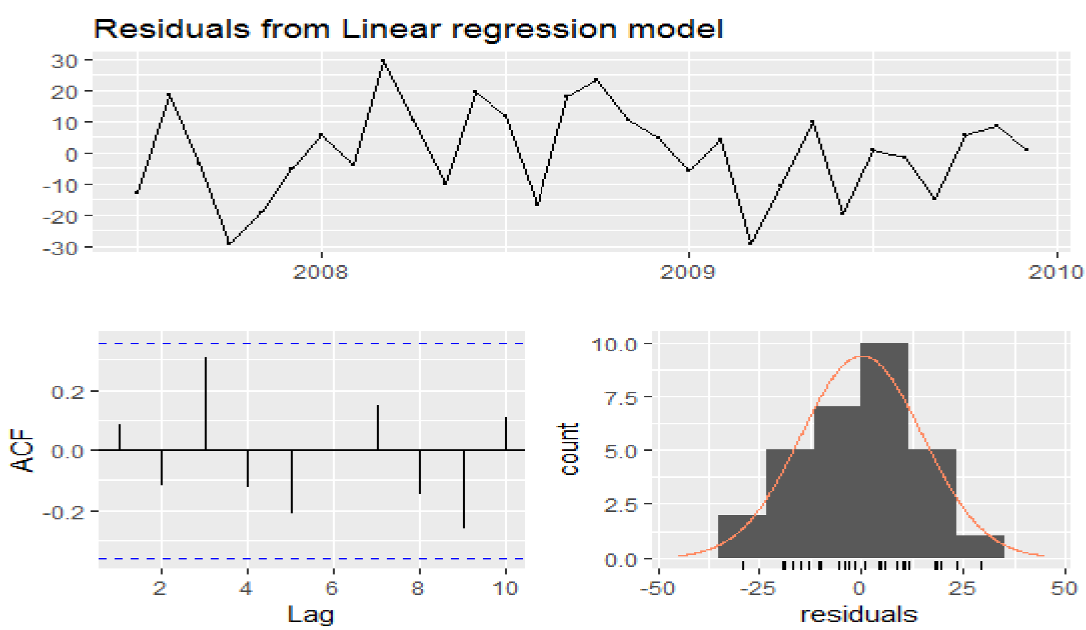



```
        Augmented Dickey-Fuller Test

data:  tsdata
Dickey-Fuller = -5.5232, Lag order = 1, p-value = 0.01
alternative hypothesis: stationary
```

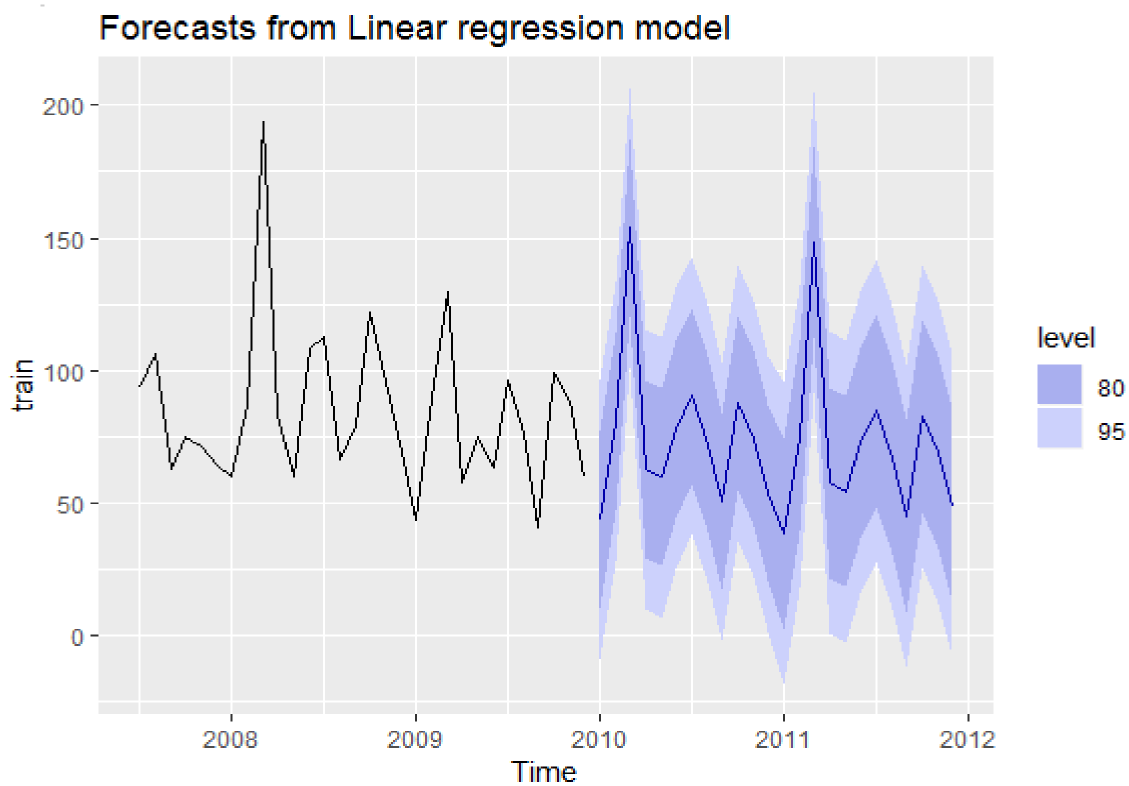
➢ 2.2.2 Linear Regression with Trend/TSLM

R-code

```
Call:
tslm(formula = train ~ season + trend)

Residuals:
     Min       1Q   Median       3Q      Max 
-29.2412 -10.2480   0.9058  10.2480  29.2412 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  57.3787    14.8215   3.871  0.00123 ** 
season2      37.1808    19.4957   1.907  0.07355 .  
season3     110.9877    19.5093   5.689 2.66e-05 ***
season4      20.4516    19.5318   1.047  0.30972    
season5      17.7437    19.5633   0.907  0.37709    
season6      36.4714    19.6037   1.860  0.08021 .  
season7      49.1641    17.7930   2.763  0.01330 *  
season8      31.7341    17.7979   1.783  0.09245 .  
season9      10.1606    17.8127   0.570  0.57586    
season10     48.1545    17.8374   2.700  0.01519 *  
season11     35.1067    17.8719   1.964  0.06605 .  
season12     14.8755    17.9161   0.830  0.41789    
trend        -0.4394     0.4194  -1.048  0.30945    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 19.49 on 17 degrees of freedom
Multiple R-squared:  0.7538,	Adjusted R-squared:   0.58 
F-statistic: 4.337 on 12 and 17 DF,  p-value: 0.003145

> AIC(fit)
[1] 274.2946
```



	Point Forecast	Lo 80	Hi 80	Lo 95	Hi 95
Jan 2010	43.75770	10.373979	77.14141	-9.065594	96.58098
Feb 2010	80.49906	47.115347	113.88278	27.675774	133.32235
Mar 2010	153.86666	120.482944	187.25038	101.043371	206.68995
Apr 2010	62.89116	29.507448	96.27488	10.067876	115.71445
May 2010	59.74382	26.360101	93.12753	6.920528	112.56711
Jun 2010	78.03219	44.648470	111.41590	25.208897	130.85548
Jul 2010	90.28547	57.411463	123.15949	38.268695	142.30225
Aug 2010	72.41607	39.542060	105.29008	20.399292	124.43285
Sep 2010	50.40321	17.529199	83.27722	-1.613569	102.41999
Oct 2010	87.95772	55.083704	120.83173	35.940936	139.97450
Nov 2010	74.47048	41.596466	107.34449	22.453698	126.48726
Dec 2010	53.79991	20.925901	86.67392	1.783133	105.81669
Jan 2011	38.48505	2.504646	74.46545	-18.446995	95.41709
Feb 2011	75.22642	39.246014	111.20682	18.294374	132.15846
Mar 2011	148.59401	112.613612	184.57441	91.661971	205.52606
Apr 2011	57.61852	21.638116	93.59892	0.686475	114.55056
May 2011	54.47117	18.490768	90.45157	-2.460873	111.40321
Jun 2011	72.75954	36.779137	108.73994	15.827496	129.69158
Jul 2011	85.01283	48.876327	121.14933	27.833790	142.19186
Aug 2011	67.14342	31.006925	103.27992	9.964387	124.32246
Sep 2011	45.13056	8.994063	81.26706	-12.048474	102.30960
Oct 2011	82.68507	46.548569	118.82157	25.506031	139.86411
Nov 2011	69.19783	33.061331	105.33433	12.018793	126.37687
Dec 2011	48.52726	12.390765	84.66376	-8.651773	105.70630



	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1	Theil's U
Training set	-4.742698e-16	14.67246	12.08983	-2.780279	14.64067	0.5148152	0.08405837	NA
Test set	1.720102e+01	34.52717	27.73557	16.325432	27.08958	1.1810499	-0.01752254	1.100461

Python code-

```
#Calculating the Root mean square error
lin_mse = mean_squared_error(pred, y_cv)
lin_rmse = np.sqrt(lin_mse)
print('Linear Regression RMSE: %.4f' % lin_rmse)
```

```
Linear Regression RMSE: 68.6820
```

```
def mean_absolute_percentage_error(y_true, y_pred):
    y_true, y_pred = np.array(y_true), np.array(y_pred)
    return np.mean(np.abs((y_true - y_pred) / y_true)) * 100

mean_absolute_percentage_error(y_cv, pred)
```

```
45.46511338098518
```

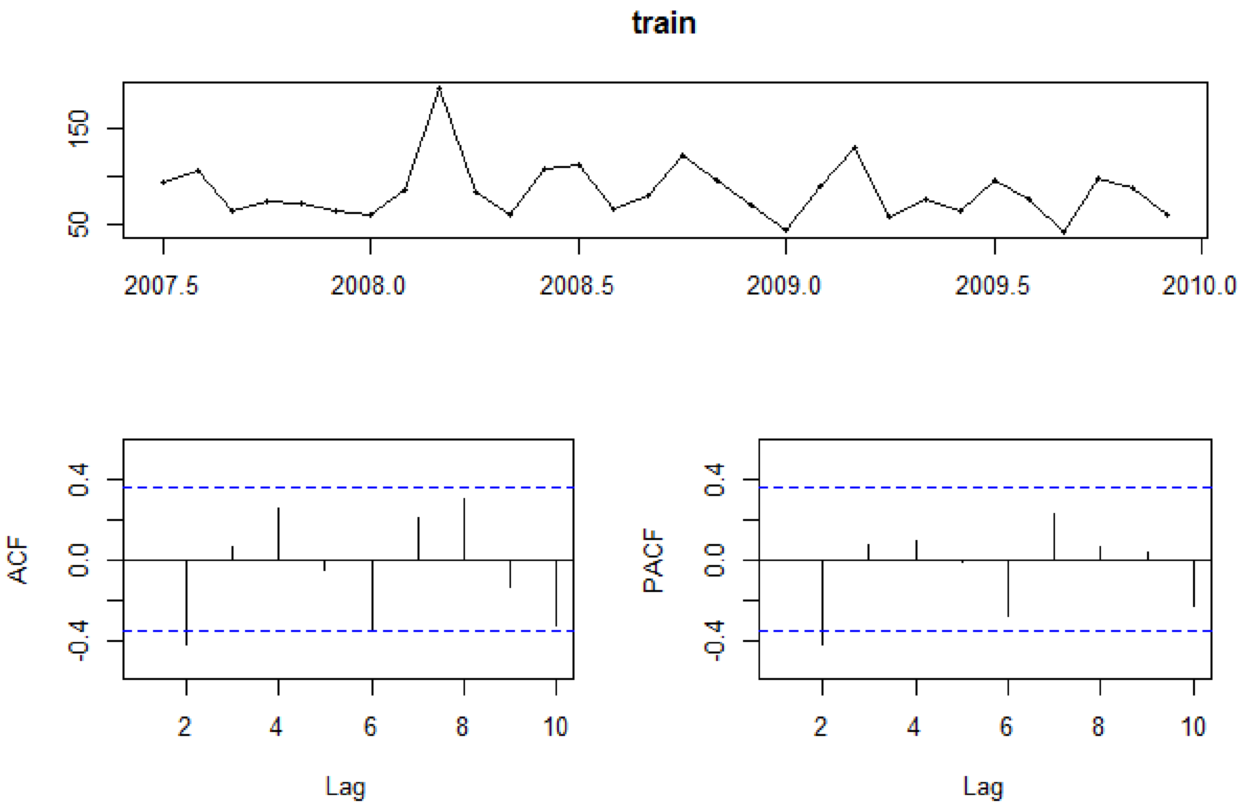
```
#AIC Score
regr = OLS(y_train, add_constant(x_train)).fit()
```

```
regr.aic
```

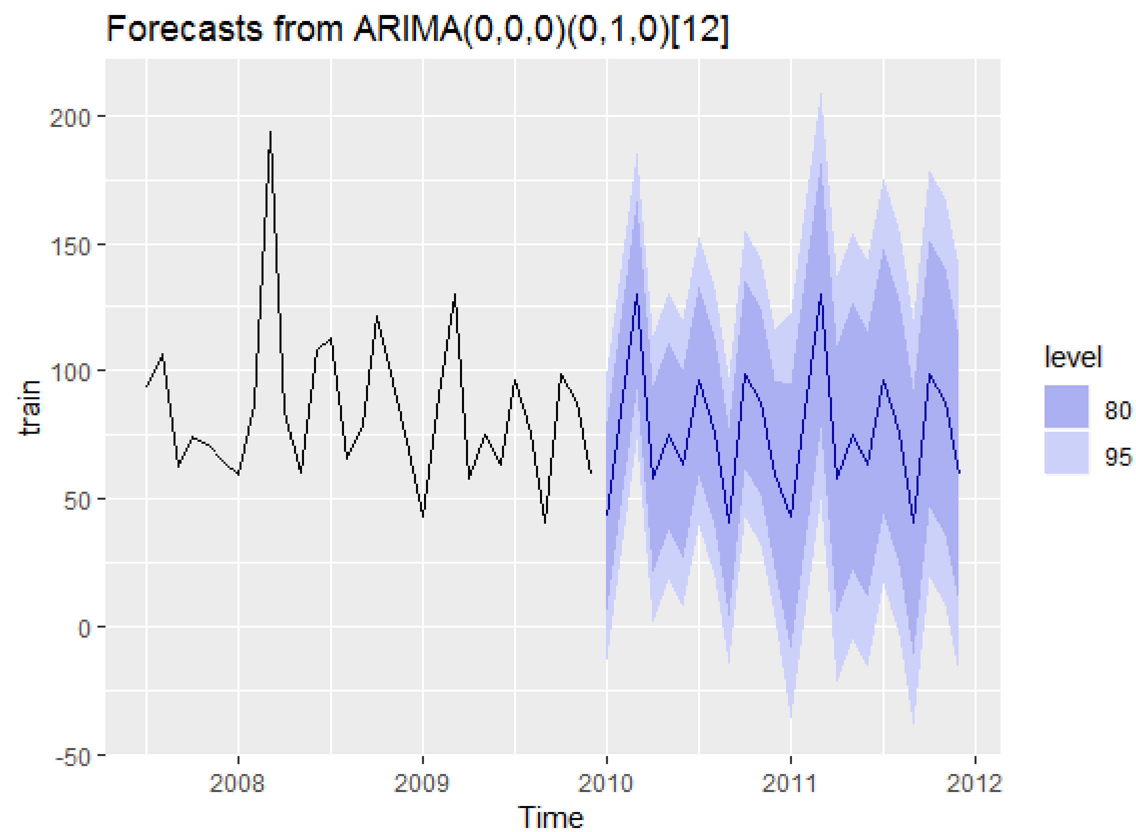
```
261.27525066824626
```

➢ 2.2.2 ARIMA

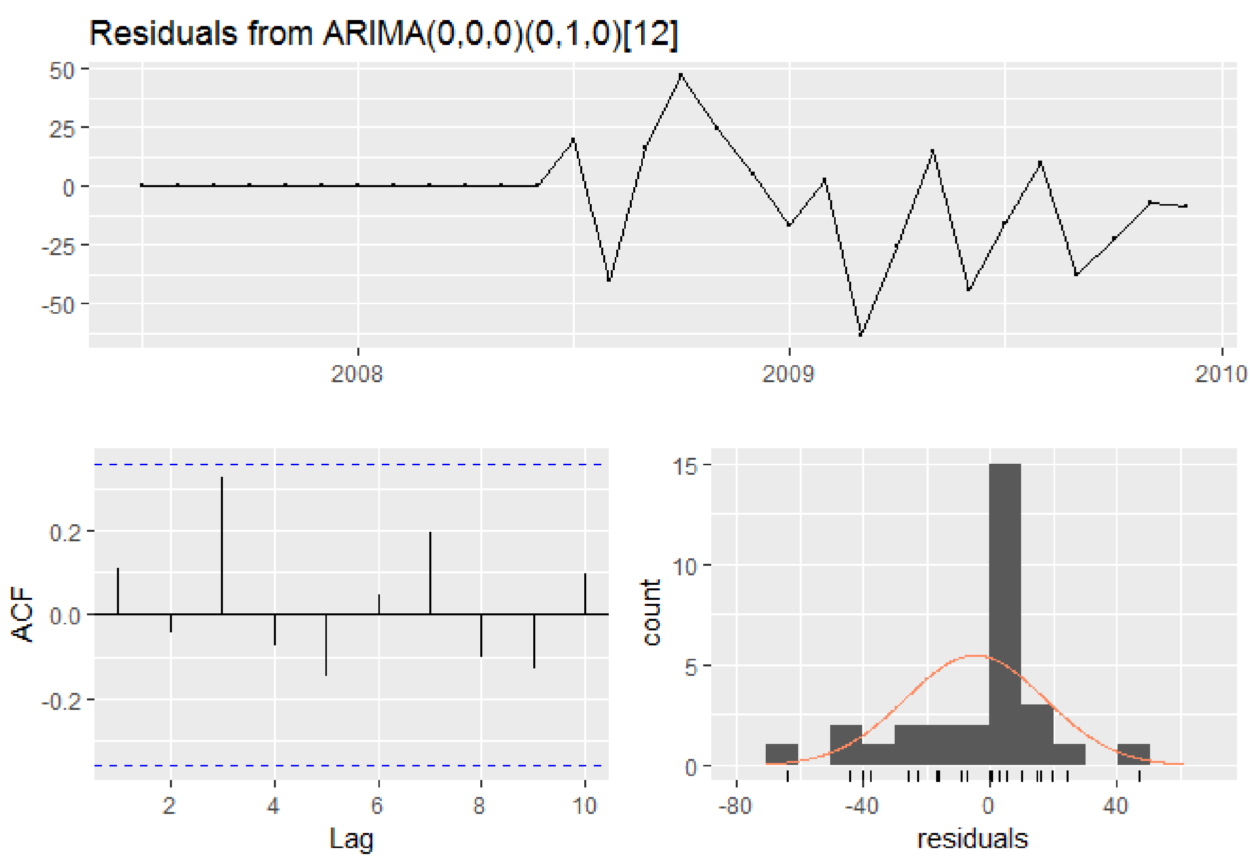
R-code



	Point Forecast	Lo 80	Hi 80	Lo 95	Hi 95
Jan 2010	43.33333	6.693291	79.97338	-12.702776	99.36944
Feb 2010	89.81607	53.176027	126.45611	33.779960	145.85218
Mar 2010	129.89815	93.258108	166.53819	73.862041	185.93426
Apr 2010	57.82768	21.187636	94.46772	1.791569	113.86379
May 2010	75.00000	38.359957	111.64004	18.963890	131.03611
Jun 2010	63.77700	27.136960	100.41705	7.740893	119.81311
Jul 2010	96.52123	59.881188	133.16127	40.485121	152.55734
Aug 2010	76.00000	39.359957	112.64004	19.963890	132.03611
Sep 2010	41.00000	4.359957	77.64004	-15.036110	97.03611
Oct 2010	99.00000	62.359957	135.64004	42.963890	155.03611
Nov 2010	88.15614	51.516098	124.79618	32.120031	144.19225
Dec 2010	59.92100	23.280956	96.56104	3.884889	115.95711
Jan 2011	43.33333	-8.483512	95.15018	-35.913693	122.58036
Feb 2011	89.81607	37.999225	141.63292	10.569044	169.06310
Mar 2011	129.89815	78.081306	181.71500	50.651125	209.14518
Apr 2011	57.82768	6.010833	109.64452	-21.419348	137.07470
May 2011	75.00000	23.183155	126.81685	-4.247026	154.24703
Jun 2011	63.77700	11.960158	115.59385	-15.470023	143.02403
Jul 2011	96.52123	44.704385	148.33808	17.274204	175.76826
Aug 2011	76.00000	24.183155	127.81685	-3.247026	155.24703
Sep 2011	41.00000	-10.816845	92.81685	-38.247026	120.24703
Oct 2011	99.00000	47.183155	150.81685	19.752974	178.24703
Nov 2011	88.15614	36.339296	139.97299	8.909115	167.40317
Dec 2011	59.92100	8.104153	111.73784	-19.326028	139.16803



	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1	Theil's U
Training set	-4.820532	22.14601	14.12582	-9.110168	18.78134	0.601513	0.1077019	NA
Test set	19.044246	30.62291	25.55911	18.079553	26.91695	1.088371	0.1286739	0.9229464



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. We can compare the models using any of the following criteria:

1. Predictive Performance
2. Interpretability
3. Computational Efficiency

In our case of Wine Data, the latter two, Interpretability and Computation *Efficiency*, do not hold much significance. Therefore we will use *Predictive performance* as the criteria to compare and evaluate models. Predictive performance can be measured by comparing Predictions of the models with real values of the forecasting, and calculating some average error measure.

3.1.1 Mean Absolute Error (MAE)

MAE is one of the error measures used to calculate the predictive performance of the model. We will apply this measure to our models that we have generated in the previous section.

```
> accuracy(forecast_tslm, test)
                        ME     RMSE      MAE       MPE     MAPE      MASE        ACF1 Theil's U
Training set -4.742698e-16 14.67246 12.08983 -2.780279 14.64067 0.5148152  0.08405837        NA
Test set      1.720102e+01 34.52717 27.73557 16.325432 27.08958 1.1810499 -0.01752254  1.100461
> accuracy(arimafore, test)
                    ME     RMSE      MAE       MPE     MAPE     MASE      ACF1 Theil's U
Training set -4.820532 22.14601 14.12582 -9.110168 18.78134 0.601513 0.1077019        NA
Test set     19.044246 30.62291 25.55911 18.079553 26.91695 1.088371 0.1286739 0.9229464
```

```
def mean_absolute_percentage_error(y_true, y_pred):
    y_true, y_pred = np.array(y_true), np.array(y_pred)
    return np.mean(np.abs((y_true - y_pred) / y_true)) * 100

mean_absolute_percentage_error(y_cv, pred)
```

```
275.0736790849102
```

3.1.2 Mean Squared Error (RMSE)

MSE can be obtained as follows

```
> accuracy(arimafore, test)
                    ME     RMSE      MAE       MPE     MAPE     MASE      ACF1 Theil's U
Training set -4.820532 22.14601 14.12582 -9.110168 18.78134 0.601513 0.1077019        NA
Test set     19.044246 30.62291 25.55911 18.079553 26.91695 1.088371 0.1286739 0.9229464
>
```

```
> accuracy(forecast_tslm, test)
                        ME     RMSE      MAE       MPE     MAPE      MASE        ACF1 Theil's U
Training set -4.742698e-16 14.67246 12.08983 -2.780279 14.64067 0.5148152  0.08405837        NA
Test set      1.720102e+01 34.52717 27.73557 16.325432 27.08958 1.1810499 -0.01752254  1.100461
```

```
lin_mse = mean_squared_error(pred, y_cv)
lin_rmse = np.sqrt(lin_mse)
print('Linear Regression without trend RMSE: %.4f' % lin_rmse)
```

```
Linear Regression without trend RMSE: 28.4931
```

Changes should company made is –

- It should hold a free Health checkup twice in year so it will help to improve the absenteeism.
- Company should take initiative at discipline because major there is major failure of disciplines should start awards and prizes for the top discipline employee and it will attract to employee to maintain discipline, or provide them an bonus.
- For the bad discipline people over a period of time should give warning and rights to fire them if it continues the same after the couple of warnings.
- Trend is often remains absent on starting of week to overcome the cost and absenteeism on week start office hours should start late.
- To avoid Absenteeism Company should set a threshold number of hours they can remain absent and if it exceed there should certain amount of deduction from their monthly ventures.

3.3 Model Selection

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

However in python Linear Regression Model works better then R models RMSE is 28% which is less than others.

Appendix A – R code

```
> summary(data)
       ID      Reason.for.absence Month.of.absence Day.of.the.week Seasons Transportation.expense
 3      :113   23     :149        3      : 87      2:161           1:169   Min.   :118.0
 28     : 76   28     :110        2      : 72      3:153           2:191   1st Qu.:179.0
 34     : 55   27     : 69        10     : 71      4:155           3:182   Median :225.0
 22     : 46   13     : 55        7      : 67      5:125           4:195   Mean   :221.5
 20     : 42   0      : 41        5      : 64      6:143                   3rd Qu.:260.0
 11     : 40   19     : 40        11     : 63                              Max.   :388.0
 (Other):365   (Other):273        (Other):313
 Distance.from.Residence.to.Work  Service.time        Age        Work.load.Average.day   Hit.target
 Min.   : 5.00                   Min.   : 1.00   Min.   :27.0   222,196: 36           Min.   : 81.00
 1st Qu.:16.00                   1st Qu.: 9.00   1st Qu.:31.0   264,249: 33           1st Qu.: 93.00
 Median :26.00                   Median :13.00   Median :37.0   237,656: 32           Median : 95.00
 Mean   :29.62                   Mean   :12.55   Mean   :36.4   343,253: 29           Mean   : 94.59
 3rd Qu.:50.00                   3rd Qu.:16.00   3rd Qu.:40.0   265,017: 28           3rd Qu.: 97.00
 Max.   :52.00                   Max.   :29.00   Max.   :58.0   284,853: 25           Max.   :100.00
                                                                (Other):554
 Disciplinary.failure Education Son     Social.drinker Social.smoker Pet       Weight        Height
 0:698                1:608     0:300   0:320          0:683         0:461   Min.   : 56   Min.   :163.0
 1: 39                2: 46     1:227   1:417          1: 54         1:136   1st Qu.: 69   1st Qu.:169.0
                      3: 79     2:154                                2: 95   Median : 83   Median :170.0
                      4:  4     3: 15                                4: 32   Mean   : 79   Mean   :172.1
                                4: 41                                5:  6   3rd Qu.: 89   3rd Qu.:172.0
                                                                     8:  7   Max.   :108   Max.   :196.0

 Body.mass.index Absenteeism.time.in.hours      Year
 Min.   :19.00   Min.   :  0.000           Min.   :2007
 1st Qu.:24.00   1st Qu.:  2.000           1st Qu.:2008
 Median :25.00   Median :  3.000           Median :2009
 Mean   :26.65   Mean   :  7.025           Mean   :2009
 3rd Qu.:31.00   3rd Qu.:  8.000           3rd Qu.:2009
 Max.   :38.00   Max.   :120.000           Max.   :2010
> str(data)
'data.frame':	737 obs. of  22 variables:
 $ ID                             : Factor w/ 34 levels "1","2","3","5",..: 10 34 3 6 10 3 9 19 13 1 ...
 $ Reason.for.absence             : Factor w/ 28 levels "0","1","2","3",..: 26 1 23 8 23 23 22 23 20 22 ...
 $ Month.of.absence               : Factor w/ 12 levels "1","2","3","4",..: 7 7 7 7 7 7 7 7 7 7 ...
 $ Day.of.the.week                : Factor w/ 5 levels "2","3","4","5",..: 2 2 3 4 4 5 5 5 1 1 ...
 $ Seasons                        : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 1 1 1 1 ...
 $ Transportation.expense         : num  289 118 179 279 289 ...
 $ 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          : Factor w/ 39 levels "","205,917","222,196",..: 8 8 8 8 8 8 8 8 8 8 ...
 $ Hit.target                     : num  97 97 97 97 97 97 97 97 97 97 ...
 $ Disciplinary.failure           : Factor w/ 2 levels "0","1": 1 2 1 1 1 1 1 1 1 1 ...
 $ Education                      : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 1 1 1 3 ...
 $ Son                            : Factor w/ 5 levels "0","1","2","3",..: 3 2 1 3 3 1 2 5 3 2 ...
 $ Social.drinker                 : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 1 ...
 $ Social.smoker                  : Factor w/ 2 levels "0","1": 1 1 1 2 1 1 1 1 1 1 ...
 $ Pet                            : Factor w/ 6 levels "0","1","2","4",..: 2 1 1 1 2 1 4 1 1 2 ...
 $ Weight                         : num  90 98 89 68 90 89 80 65 95 88 ...
 $ Height                         : num  172 170 170 168 172 ...
 $ Body.mass.index                : num  30 31 31 24 30 31 27 23 25 29 ...
 $ Absenteeism.time.in.hours      : num  4 0 2 4 2 ...
 $ Year                           : num  2007 2007 2007 2007 2007 ...
```

Count of reasons according to in week, season , reasons of absence

```
> #Count for Reason of absence
> g11 = ggplot(data, aes(x = Reason.for.absence, fill = Reason.for.absence)) + geom_bar(stat = 'count')  +
+   geom_label(stat='count',aes(label=..count..), size=5) + labs(ggtitle("Count for Reason of Absence")) +
+   theme_grey(base_size = 18)
> g11
> #Reasons for absence according to day or week, seasons and month of absence
> g12 = ggplot(data, aes(x = Reason.for.absence, fill = Day.of.the.week)) + geom_bar(stat = 'count', position = 'dodge')
+   geom_label(stat = 'count', aes(label = ..count..)) + labs(ggtitle("Reason of absence based day of week"))+
+   theme_grey()
> g12
>
> g13 = ggplot(data, aes(x = Reason.for.absence, fill = Seasons)) + geom_bar(stat = 'count', position = 'dodge')+
+   geom_label(stat = 'count', aes(label = ..count..)) + labs(ggtitle("Reason of absence based on seasons of all the yea
s"))+
+   facet_grid(Year~.)
> theme_grey()
```

Histogram –

```
> ggplot(data , aes(x = data$Social.drinker))+
+   geom_histogram(binwidth = 1 , fill = "navyblue" ,  colour = "navyblue")+
+   ggtitle("Histogram Analysis") +  theme(text=element_text(size=15))
Warning message:
Removed 3 rows containing non-finite values (stat_bin).
>
> ggplot(data , aes(x = data$Social.smoker))+
+   geom_histogram(binwidth = 1 , fill = "navyblue" ,  colour = "navyblue")+
+   ggtitle("Histogram Analysis") +  theme(text=element_text(size=15))
Warning message:
Removed 4 rows containing non-finite values (stat_bin).
>
> ggplot(data , aes(x = data$Pet))+
+   geom_histogram(binwidth = 1 , fill = "navyblue" ,  colour = "navyblue")+
+   ggtitle("Histogram Analysis") +  theme(text=element_text(size=15))
Warning message:
Removed 2 rows containing non-finite values (stat_bin).
>
> ggplot(data , aes(x = data$Weight))+
+   geom_histogram(binwidth = 1 , fill = "navyblue" ,  colour = "navyblue")+
+   ggtitle("Histogram Analysis") +  theme(text=element_text(size=15))
Warning message:
Removed 1 rows containing non-finite values (stat_bin).
>
> ggplot(data , aes(x = data$Height))+
+   geom_histogram(binwidth = 1 , fill = "navyblue" ,  colour = "navyblue")+
+   ggtitle("Histogram Analysis") +  theme(text=element_text(size=15))
Warning message:
Removed 14 rows containing non-finite values (stat_bin).
>
> ggplot(data , aes(x = data$Body.mass.index))+
+   geom_histogram(binwidth = 1 , fill = "navyblue" ,  colour = "navyblue")+
+   ggtitle("Histogram Analysis") +  theme(text=element_text(size=15))
Warning message:
Removed 31 rows containing non-finite values (stat_bin).
>
> ggplot(data , aes(x = data$Absenteeism.time.in.hours))+
+   geom_histogram(binwidth = 1 , fill = "navyblue" ,  colour = "navyblue")+
+   ggtitle("Histogram Analysis") +  theme(text=element_text(size=15))
```

Appendix B – Python code

Boxplot-

```
plt.figure(figsize=(10,10))
plt.figure(figsize=(10,10))
plt.suptitle("boxplot",fontsize=15)
for i in num_var :
    sns.boxplot(
    y = df[i],
    data=df)
    plt.show()
```

Correlation plot-

```
#Set the width and hieght of the plot
f, ax = plt.subplots(figsize=(10, 7))

#Generate correlation matrix
corr = df_corr.corr()

#Plot using seaborn library
sns.heatmap(corr, mask=np.zeros_like(corr, dtype=np.bool), cmap=sns.diverging_palette(220, 10, as_cmap=True),
            square=True, ax=ax)
```

Median Imputation

```
df['reason_for_absence'] = df['reason_for_absence'].fillna(df['reason_for_absence'].median())
df['month_of_absence'] = df['month_of_absence'].fillna(df['month_of_absence'].median())
df['transportation_expense'] = df['transportation_expense'].fillna(df['transportation_expense'].median())
df['distance_from_residence_to_work'] = df['distance_from_residence_to_work'].fillna(df['distance_from_residence_to_work'].median
df['service_time'] = df['service_time'].fillna(df['service_time'].median())
df['age'] = df['age'].fillna(df['age'].median())
df['work_load_average/day'] = df['work_load_average/day'].fillna(df['work_load_average/day'].median())
df['hit_target'] = df['hit_target'].fillna(df['hit_target'].median())
df['disciplinary_failure'] = df['disciplinary_failure'].fillna(df['disciplinary_failure'].median())
df['education'] = df['education'].fillna(df['education'].median())
df['son'] = df['son'].fillna(df['son'].median())
df['social_drinker'] = df['social_drinker'].fillna(df['social_drinker'].median())
df['social_smoker'] = df['social_smoker'].fillna(df['social_smoker'].median())
df['pet'] = df['pet'].fillna(df['pet'].median())
df['weight'] = df['weight'].fillna(df['weight'].median())
df['height'] = df['height'].fillna(df['height'].median())
df['body_mass_index'] = df['body_mass_index'].fillna(df['body_mass_index'].median())
df['absenteeism_time_in_hours'] = df['absenteeism_time_in_hours'].fillna(df['absenteeism_time_in_hours'].median())
```

References

ARTIFICIAL NEURAL NETWORK AND THEIR APPLICATION IN THE PREDICTION OF ABSENTEEISM AT WORK Ricardo Pinto Ferreira., Andréa Martiniano., Domingos Napolitano.,Edquel Bueno Prado Farias and Renato José Sassi