Project Report
Employee Absenteeism

Sourav

19 December, 2018

Contents

1. Introduction

2. Methodology

3. Modelling

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

There are 21 variables in our data in which 20 are independent variables and 1 (Absenteeism time in hours) is dependent variable. Since our target variable is a continuous variable, this is a regression problem.

Variables Information:

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 behavioral 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)

1.2.1 <u>Data set:</u>

Data is described upon parameters such as the Reason for Absence, various things involved, health issue or work load would be the reason. The table represents a sample of various fields available in the data.

Table 1.1 Absenteeism at Work (Column 1-7)

ID	Reason for absence	Month of absence	Day of the week	Seasons	Transportation expense	Distance from Residence to Work	Service time	Age	Work load Average/day	Hit target	Disciplinary failure
11	26	7	3	1	289	36	13	33	239,554	97	0
36	0	7	3	1	118	13	18	50	239,554	97	1
3	23	7	4	1	179	51	18	38	239,554	97	0
7	7	7	5	1	279	5	14	39	239,554	97	0
11	23	7	5	1	289	36	13	33	239,554	97	0

Table 1.2 Absenteeism at Work (Column 8-14)

Education	Son	Social drinker	Social smoker	Pet	Weight	Height	Body mass index	Absenteeism time in hours
1	2	1	0	1	90	172	30	4
1	1	1	0	0	98	178	31	0

1	0	1	0	0	89	170	31	2
1	2	1	1	0	68	168	24	4

As we can see in the table below we have the following 21 variables, using which we have to correctly predict the Employee Absenteeism time in hour for our target variable.
Summary of data is given below to know variables types and dimension of data.

Fig 1.1 Summary of data

```
'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 NaN 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 NaN 8 4 40 8 ...
```

1.3 <u>Exploratory Data Analysis</u>

Exploratory Data Analysis (EDA) is an approach to analyzing data sets to summarize their main characteristics. In the given data set there are 21 variables and data types of all variables are either float64 or int64. There are 740 observations and 21 columns in our data set. Missing value is also present in our data.

<u>List of variables and their types:</u>

```
ID                                 int64
Reason for absence                object
Month of absence                  object
Day of the week                   object
Seasons                           object
Transportation expense           float64
Distance from Residence to Work  float64
Service time                      object
```

```
Age                                  float64
Work load Average/day                float64
Hit target                            object
Disciplinary failure                  object
Education                             object
Son                                   object
Social drinker                        object
Social smoker                         object
Pet                                   object
Weight                               float64
Height                               float64
Body mass index                      float64
Absenteeism time in hours            float64
dtype: object
```

From EDA we have concluded that there are 10 continuous variable and 11 categorical variable in nature.

2. Methodology

2.1 Data Preprocessing

Data in real world is dirty it of no use until unless we apply data preprocessing on it. In other words, Pre- processing refers to the transformations applied to your data before feeding it to the algorithm. It's a data mining technique which that involves transforming raw data into an understandable format or we can say that it prepares raw data to further processing. There are so many things that we do in data preprocessing like data cleaning, data integration, data transformation, or data reduction.

2.1.1 Missing Value Analysis

Missing Values Analysis is use to fill NULL values in data with some imputation techniques
But here in our Employee Absenteeism Data, we have null Values. By the way our data contain missing value. We will impute those values using KNN.

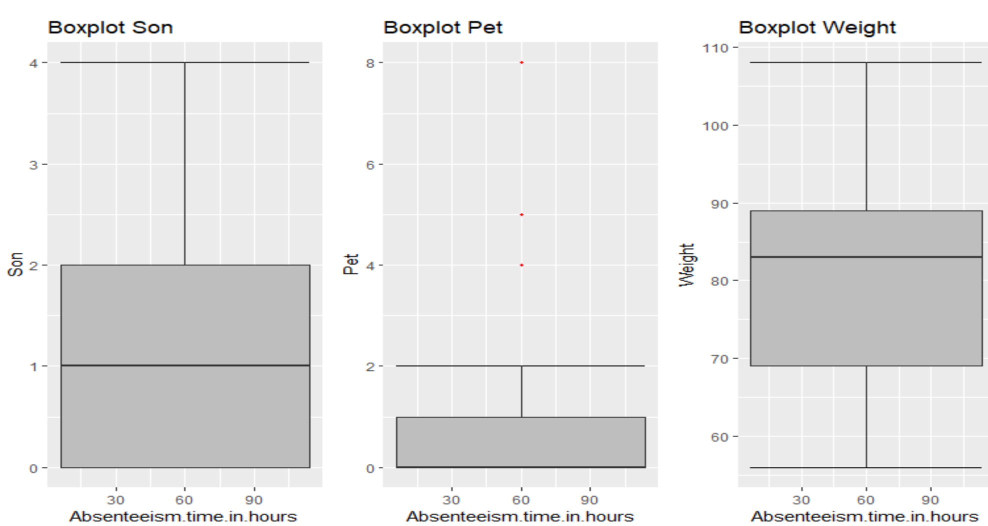
Fig 2.1 Number of missing Values

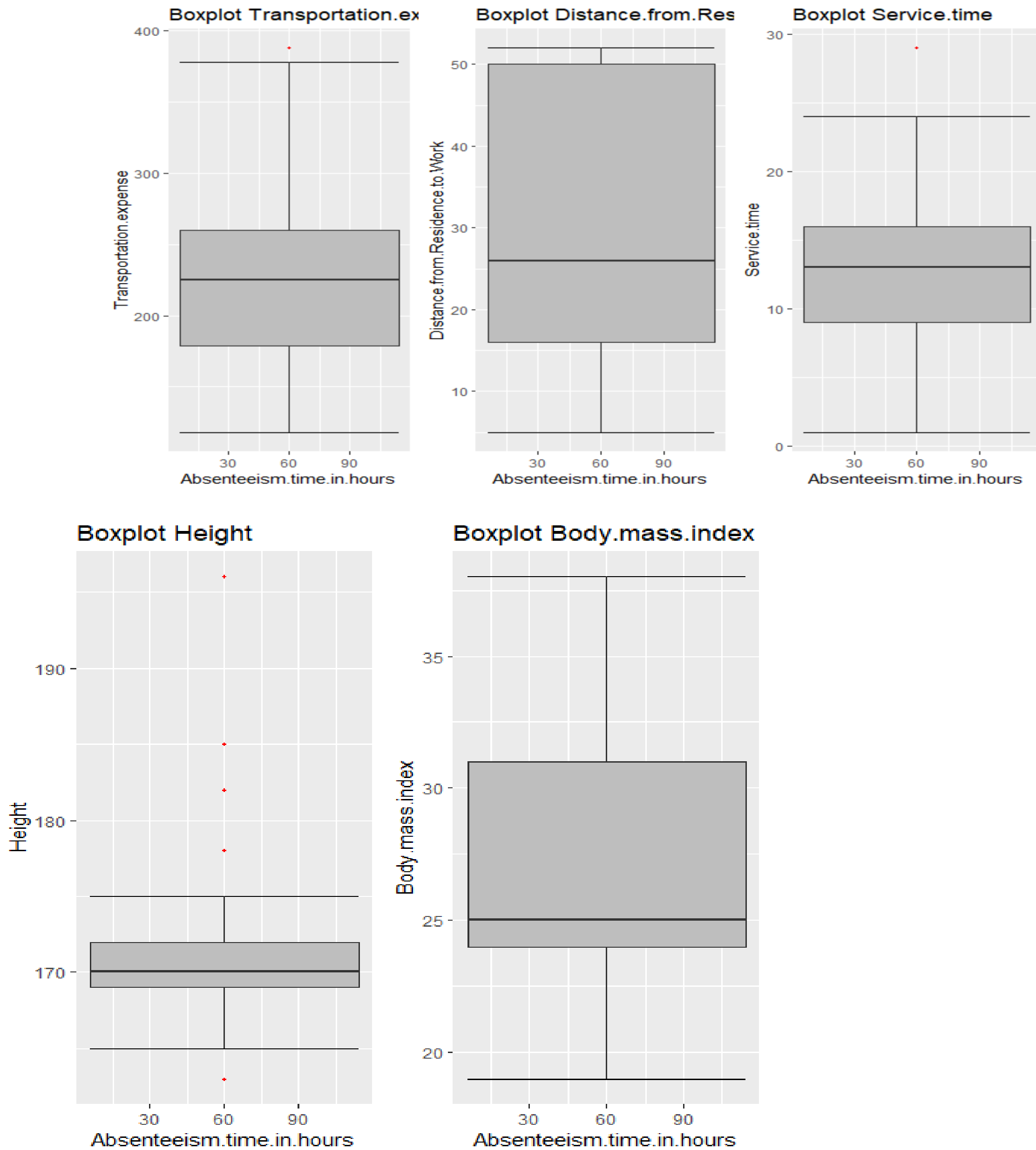
```
                             ID               Reason.for.absence                 Month.of.absence
                              0                                3                                1
                Day.of.the.week                          Seasons           Transportation.expense
                              0                                0                                7
Distance.from.Residence.to.Work                     Service.time                              Age
                              3                                3                                3
          Work.load.Average.day.                       Hit.target              Disciplinary.failure
                             10                                6                                6
                      Education                              Son                   Social.drinker
                             10                                6                                3
                   Social.smoker                              Pet                           Weight
                              4                                2                                1
                         Height                  Body.mass.index        Absenteeism.time.in.hours
                             14                               31                               22
```

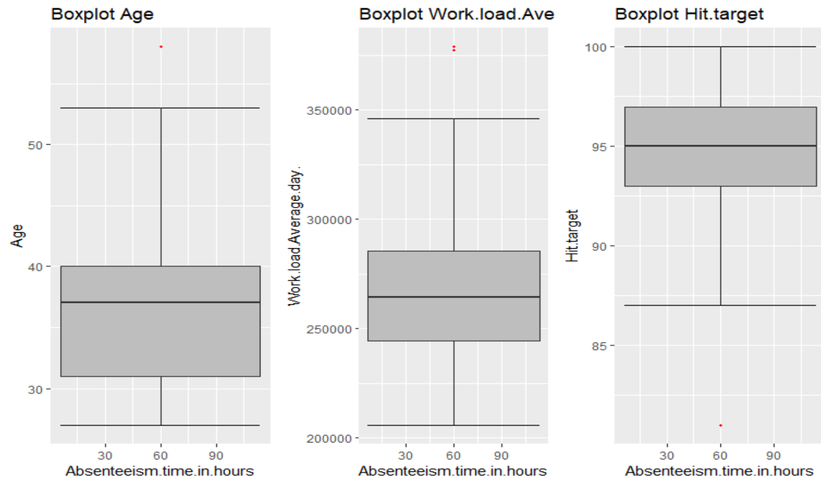
2.1.2 Outlier Analysis

The shown boxplot Fig: 2.3 refers outliers on the predictors variables, we can see various outliers associated with the features. Even though, the data has considerable amount of outliers, the approach is to retain every outlier and grab respective behavior of all employees. As shown there are significant amount of outliers present in the target variable, which indicates a trend on Employee ' behavior, there can be pattern , we need to treat those outliers.

Fig 2.3 Outlier Values




Boxplot Transportation.ex
Transportation.expense
Absenteeism.time.in.hours
Boxplot Distance.from.Res
Distance.from.Residence.to.Work
Absenteeism.time.in.hours
Boxplot Service.time
Service.time
Absenteeism.time.in.hours
Boxplot Height
Height
Absenteeism.time.in.hours
Boxplot Body.mass.index
Body.mass.index
Absenteeism.time.in.hours



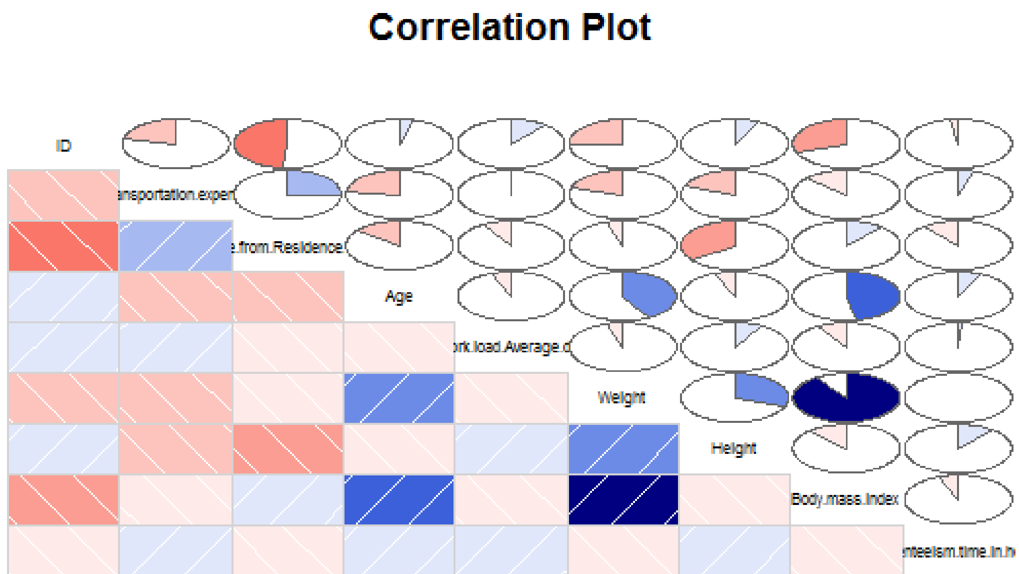
From the boxplot almost all the variables **except "Distance from residence to work", "Weight" and "Body mass index"** consists of outliers. We have converted the outliers (data beyond minimum and maximum values) as NA i.e. missing values and fill them by **KNN** imputation method.

In figure we have plotted the boxplots of the 11 predictor variables with respect to **Absenteeism time in hour**. 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.

2.1.3 Feature Engineering

Feature Engineering is described as the knowledge extraction process, where important features are selected using domain knowledge to make a machine learning algorithm work. There can be features that aren't relevant for the analysis, we can remove such variables using numerous ways. However, we
Considered taking correlation on the variables and make a heat map Fig: 2.5 to check relationships among the features and then dropping redundant variables.

Fig 2.5 Correlation plot of variables



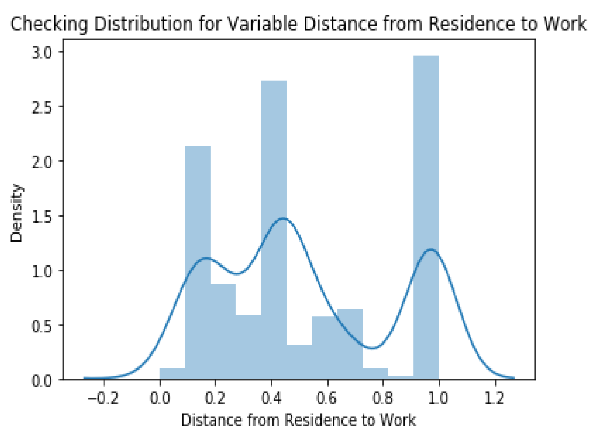
From these graph we can see that there are some variables which have collinearity problems or they are highly correlated.

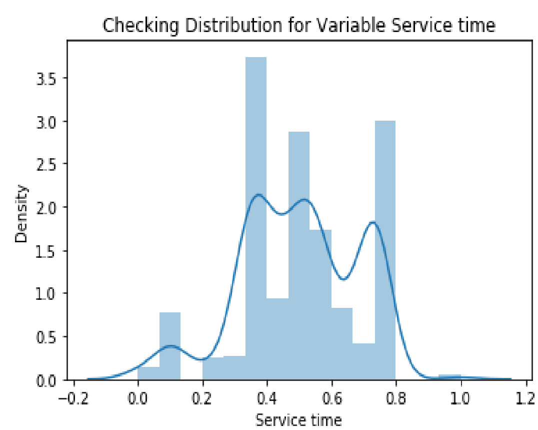
1. The weight predictor is highly correlated to body mass index
2. On applying the chi square test, the p values of the following variables are found to be greater than 0.05, ID, Education, Social.smoker, Pet.

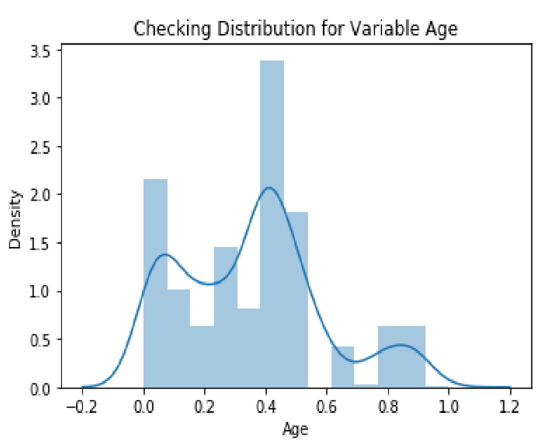
One of the assumptions of logistic regression is that logistic regression requires there to be little or no multicollinearity among the independent variables. This means that the independent variables should not be too highly correlated with each other. Due to this assumption, one the predictors from each set was removed when logistic learner was trained.

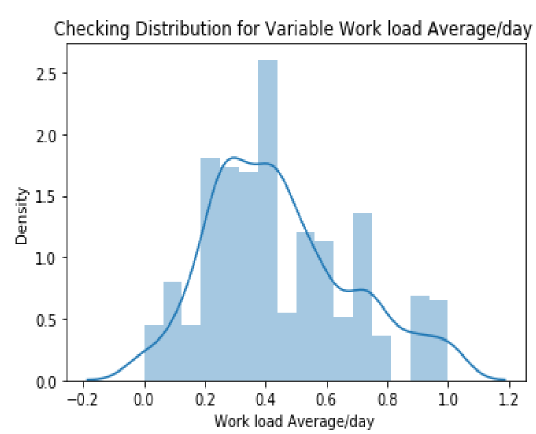
2.1.4 Feature Scaling

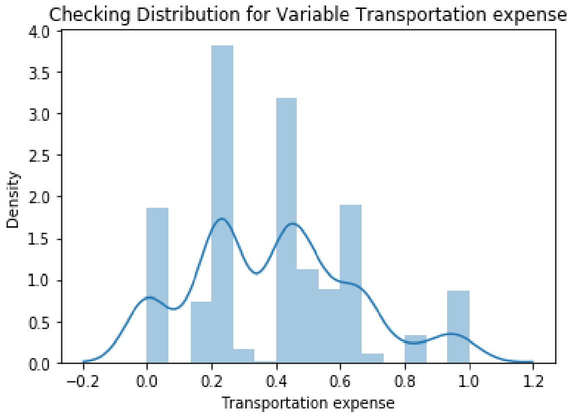
Checking distribution curve for all continuous variable

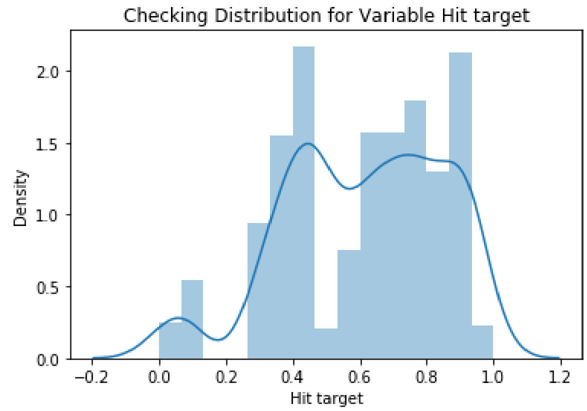


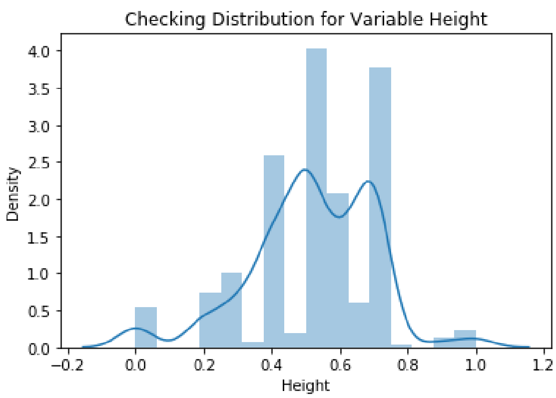


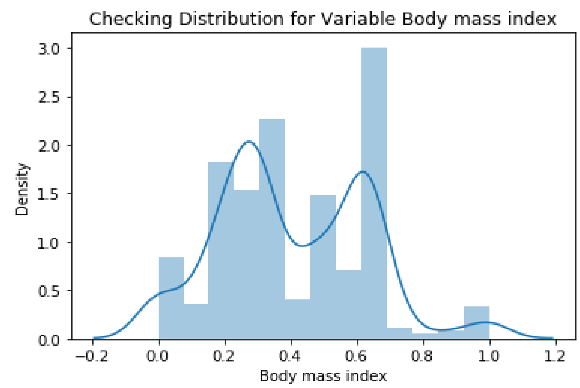












Feature scaling is a method used to standardize the range of independent variables or features of data. In data processing, it is also known as data normalization and is generally performed during the data preprocessing step. Since the range of values of raw data varies widely, in some machine learning algorithms, objective functions will not work properly without normalization. For example, the majority of classifiers calculate the distance between two points by the Euclidean distance. If one of the features has a broad range of values, the distance will be governed by this particular feature. Therefore, the range of all features should be normalized so that each feature contributes approximately proportionately to the final distance. Since our data is not uniformly distributed we will use **Normalization** as Feature Scaling Method.

2.1.5 Sampling

The dataset is been divided into 2 parts train data to train a machine learning model and test data to test the model accuracy
The data is divided into 8:2 ratio that means 80% of data is training data and rest 20% data is test data.

3. Modelling

Absenteeism at work is a regression problem. Here according to the problem statement, we are supposed to predict the loss incurred by the company if the same pattern of absenteeism continues. Hence we are selection the following two models,

1. Decision tree
2. Random forest model

Both training models Decision tree and random forest were implemented in R and python. After building an initial model, performance tuning was done using hyper parameter tuning for optimized parameters.

3.1 Decision Tree

Train data was divided into train dataset and validation set.

- Logistic regression models were trained on train dataset.
- Validation set and AIC score was used to select the best models out of all trained models.
- Final test and prediction was performed on test data which was provided separately.

R implementation:

##model development (decision tree regression model)

#reg_model = rpart(Absenteeism.time.in.hours ~. , data = train, method = "anova")

#predicting reg_model for test cases

#predictions = predict(reg_model , test[,-14])

#RMSE OR MSEis the technique which can be used to evaluate the performance of a regression model

#here, i will use root mean square error technique to evaluate the performance of the model, moreover the data is a time series data

```
#library("DMwR")

#RMSE = regr.eval(test[,14], predictions, stats = 'rmse')

#RMSE #14.92
```

##accuracy = 85.02%

Thus in Decision tree regression model the error is 14.92 which tells that our model is 85.08% accurate____

Python implementation:

```
from sklearn.tree import DecisionTreeRegressor
#Decision tree for regression
fit_DT = DecisionTreeRegressor(max_depth=2).fit(train.iloc[:,0:9], train.iloc[:,9])

#checking for any missing valuses that has leeked in
np.where(data.values >= np.finfo(np.float64).max)

test = test.fillna(train.mean())

#Decision tree for regression
fit_DT = DecisionTreeRegressor(max_depth=2).fit(train.iloc[:,0:15], train.iloc[:,15])

#Apply model on test data
predictions_DT = fit_DT.predict(test.iloc[:,0:15])

def rmse(predictions, targets):
   return np.sqrt(((predictions - targets) ** 2).mean())

rmse(test.iloc[:,15], predictions_DT)


#rmse using DT = 0.16972039562573937
```

3.2 Random Forest

After decision tree, random forest was trained. It was implemented in both R and python. In both implementations random forest was first trained with default setting and the hyper parameters tuning was used to find the best parameters.

R Implementation:

library("randomForest")

RF_model = randomForest(Absenteeism.time.in.hours~. , train, importance = TRUE, ntree=100)

#Extract the rules generated as a result of random Forest model

library("inTrees")

rules_list = RF2List(RF_model)

#Extract rules from rules_list

rules = extractRules(rules_list, train[,-14])

rules[1:2,]

#Convert the rules in readable format

read_rules = presentRules(rules,colnames(train))

read_rules[1:2,]

#Determining the rule metric

rule_metric = getRuleMetric(rules, train[,-14], train$Absenteeism.time.in.hours)

rule_metric[1:2,]

#Prediction of the target variable data using the random Forest model

RF_prediction = predict(RF_model,test[,-14])

RMSE_RF = regr.eval(test[,13], RF_prediction, stats = 'rmse')

#RMSE = 7.88

#Accuracy = 92.12%

#Thus the error rate in Random Forest Model is 7.88% and the accuracy of the model is 100-7.88 = 92.12%.

Python implementation:

#Divide data into train and test

X = data.values[:, 0:15]

Y = data.values[:,15]

X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size = 0.2)

from sklearn.ensemble import RandomForestRegressor

 # Building model on top of training dataset

fit_RF = RandomForestRegressor(n_estimators = 500).fit(X_train,y_train)

 # Calculating RMSE for test data to check accuracy

RF_predictions_test = fit_RF.predict(X_test)

rmse_for_test =np.sqrt(mean_squared_error(y_test,RF_predictions_test))

```
rmse_for_test = 0.18017389276028373
```

3.3 Linear regression:

###________________ LINEAR REGRESSION _______________________________

#library("usdm")

#LR_data_select = subset(data_sorted ,select = -c(Reason.for.absence,Day.of.the.week))

#colnames(LR_data_select)

#vif(LR_data_select[,-12])

#vifcor(LR_data_select[,-12], th=0.9)

####Execute the linear regression model over the data

#lr_model = lm(Absenteeism.time.in.hours~. , data = train)

#summary(lr_model)

#colnames(test)

###___Multiple R-squared: 0.2674, Adjusted R-squared: 0.1953, which means our target variable can explain 26.74% of variance which is not acceptable.

#Predict the data

#LR_predict_data = predict(lr_model, test[,1:13])

#Calculate MAPE

#MAPE(test[,14], LR_predict_data)

#library(Matrix)

#rmse(test[,14],LR_predict_data)

##linear regression model works best for continuous variables, but here in LR_data_select we have categorical variables.

##________ Till here we have implemented Decision Tree, Random Forest and Linear Regression. Among all of these Random Forest is having highest accuracy.

Python implementation:

Importing libraries for Linear Regression

from sklearn.linear_model import LinearRegression

Building model on top of training dataset

fit_LR = LinearRegression().fit(X_train , y_train)

Calculating RMSE for training data to check for over fitting

LR_pred_train = fit_LR.predict(X_train)

rmse_for_train = np.sqrt(mean_squared_error(y_train,LR_pred_train))

Calculating RMSE for test data to check accuracy

LR_pred_test = fit_LR.predict(X_test)

rmse_for_test =np.sqrt(mean_squared_error(y_test,LR_pred_test))

print("Root Mean Squared Error For Training data = "+str(rmse_for_train))

print("Root Mean Squared Error For Test data = "+str(rmse_for_test))

```
Root Mean Squared Error For Training data = 0.17189637356023962
Root Mean Squared Error For Test data = 0.20422938819536107
```

4. Conclusion

4.1 Model Evaluation

As we can see, we have applied all the possible preprocessing analysis to our dataset to make it suitable

For calculation.

We have also removed the missing values and outliers.

Now since our data is a regression model, we have applied suitable models

Such as decision tree and random forest.

The error metric results of both the models are as follows,

Using R,

Rmse value applying decision tree, **0.1492**

This means that our predictions vary from the actual value by about 0.1492

Rmse value using random forest, **0.0788**

This means that our predictions vary from the actual value by about 0.0.0788

Rmse value using linear regression, **0.2694**

This means that our predictions vary from the actual value by about 0.2694

Using python,

Rmse value applying decision tree, **0.16972**
This means that our predictions vary from the actual value by about 0.16972
Rmse value using random forest**, 0.18017**

This means that our predictions vary from the actual value by about 0.18017

Rmse value using linear regression, **0.20422**

This means that our predictions vary from the actual value by about 0.20422

Hence comparing R and python, since the error rate of R is comparatively better, we consider th e code of R

AND on comparing the values of decision tree and random forest, since the error rate of random forest is comparatively better, we consider the value of random forest.

Hence, finally, we are accepting the random forest model of R, which has an RMSE value of 0.18017, which is negligible.

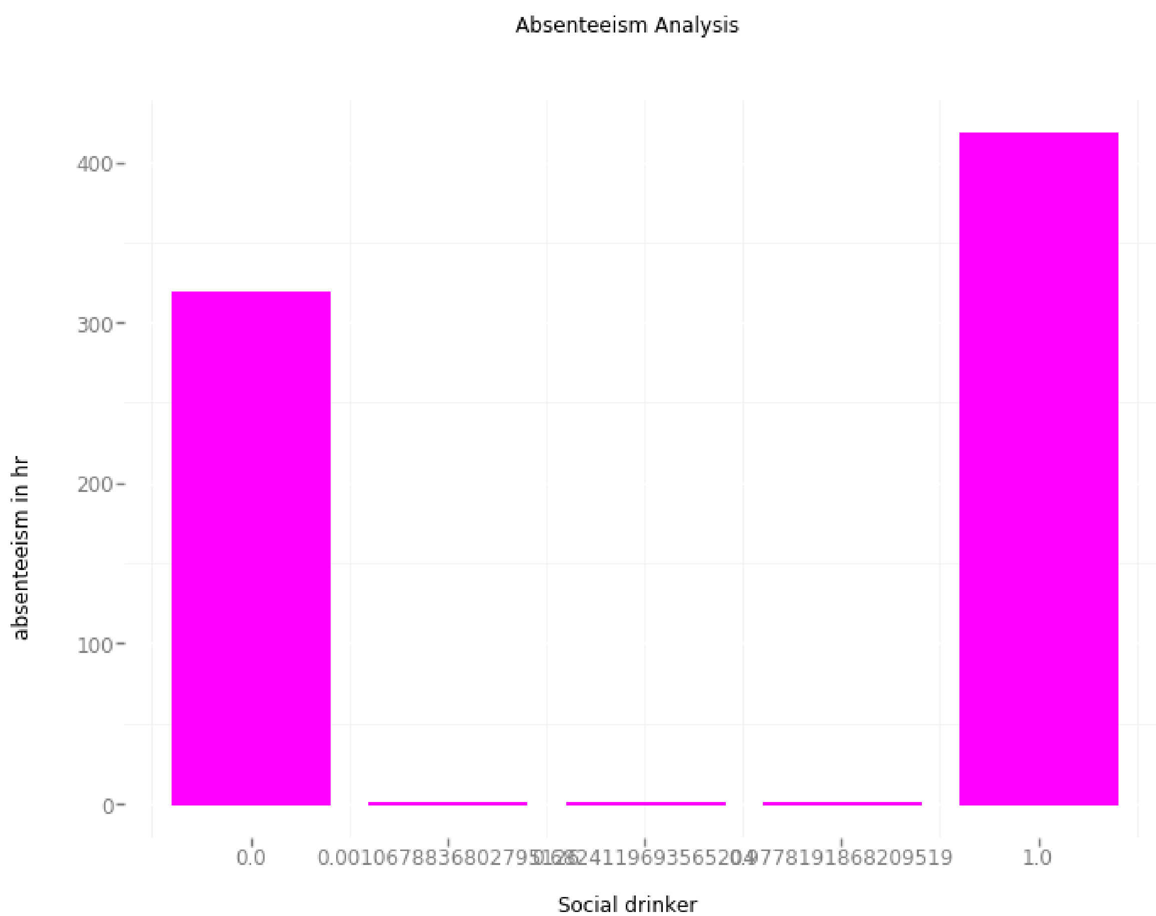
5. Visualizations:

ANALYSING ABSENTEEISM TREND ON TEST DATA

Below are some suggestions and visualizations that a company should take forward to reduce its rate of absenteeism based on the data provided.

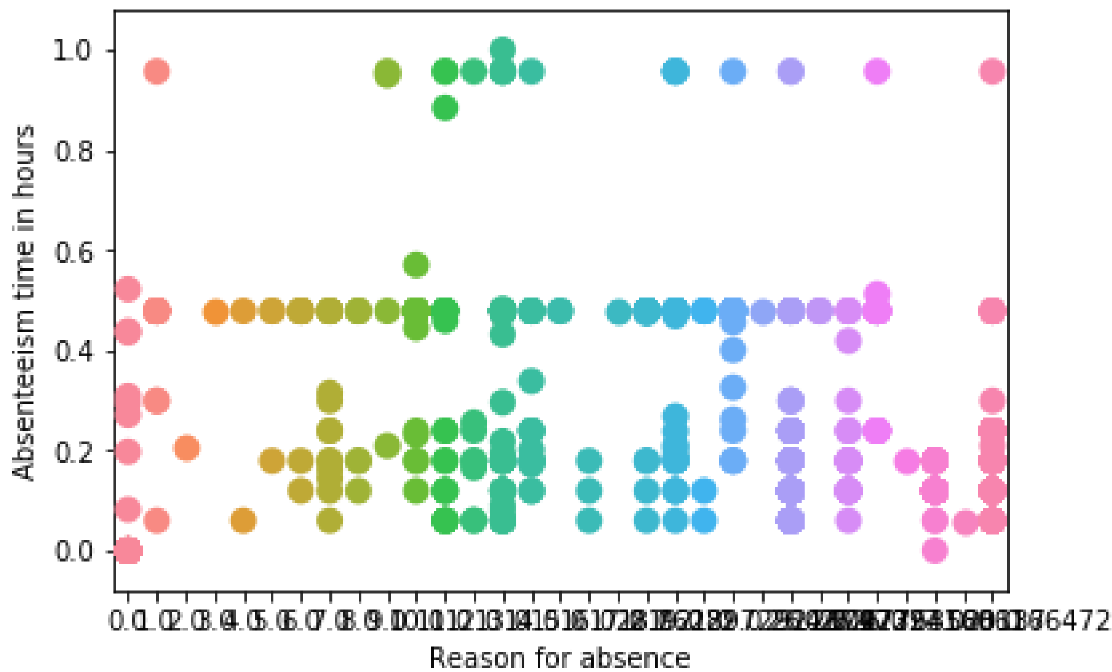
The Changes which company should bring to reduce the number of absenteeism –

1. Employees who are social drinker have more absentee hour than who are not social drinker.



As a drinker is more prone to bad health condition so that causes a lot of absenteeism. So a firm should conduct health campaigns to educated employee about the harmful effects of smoking.

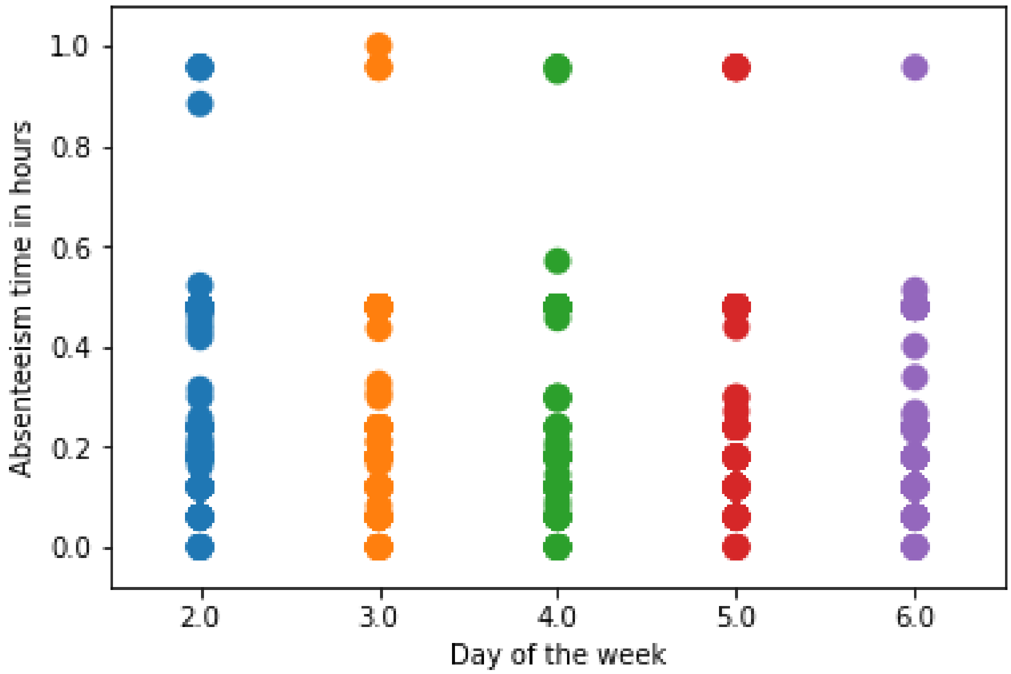
2. Most often Reason for absence are medical consultation and dental consultation, company should take care of it. The maximum people taking the absent hours are from category 23 followed by 28 and 27. These category are not attested by doctors. 23: Medical Consultation. 28: Dental consultation 27: Physiotherapy .



Other than the above statements

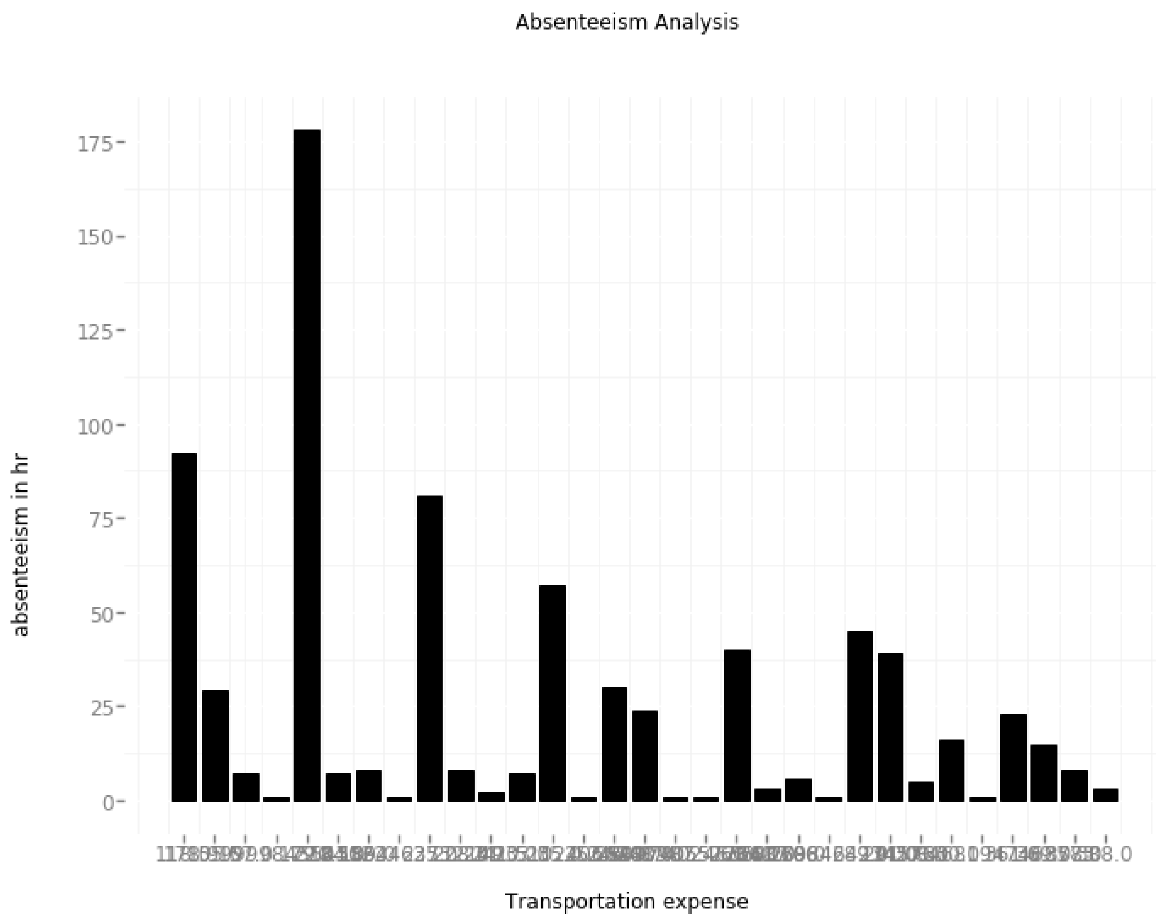
- A company should introduce 100% attendance incentive bonus.
- Should introduce new policy for salary deduction for un-informed or un-approved leaves
- A sandwich leave policy would be a good option to reduce absenteeism on day near to weekends.

3. Most of the employees are absent on Monday.

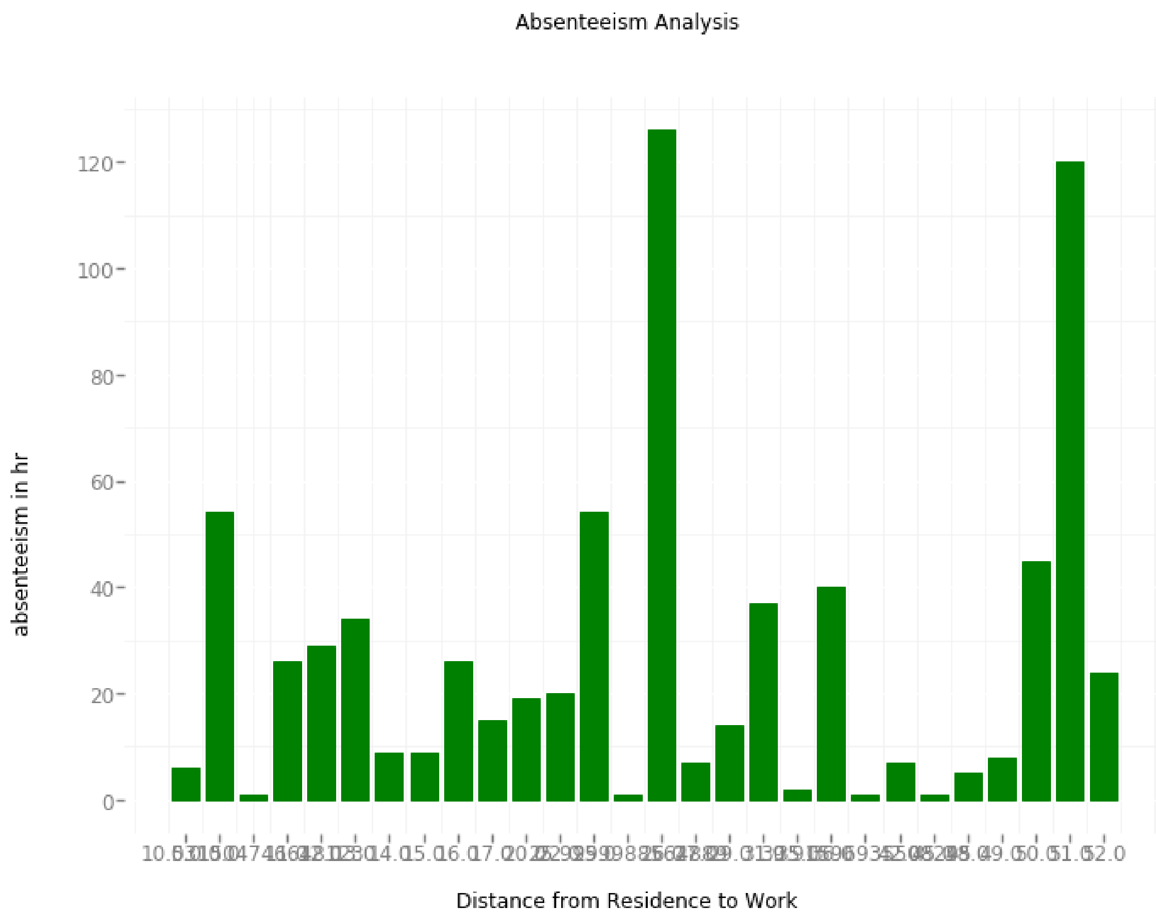


A company should motivate their employees for not being lazy.

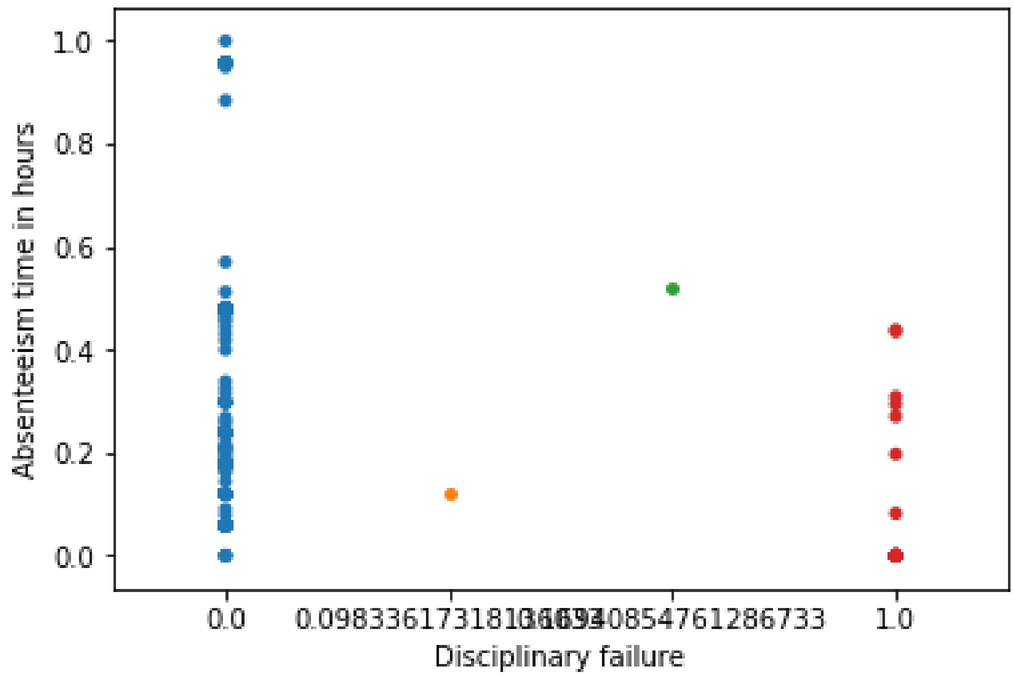
4. Company should provide transportation expense to employees who are communicating from a considerable distance.



5. A company should provide cab facility to ease the transportation towards employees.



6. A company should provide proper teaching and training to their employees after interval of time as the discipline is the key to growth.


1.0
0.8
0.6
0.4
0.2
0.0
Absenteeism time in hours
0.0
0.09833617318136659
0.16340854761286733
1.0
Disciplinary failure