Employee Absenteeism

Vinayak Chaturvedi

Date: 13-May-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 Set

The details of the dataset are as follows:

Dataset Characteristics: Timeseries Multivariant
Number of Attributes: 21 (Independent Variable = 20 + Target Variable = 1)
Missing Values : Yes

Attribute 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 behavioural disorders
VI Diseases of the nervous system
VII Diseases of the eye and adnexa
VIII Diseases of the ear and mastoid process
IX Diseases of the circulatory system
X Diseases of the respiratory system
XI Diseases of the digestive system
XII Diseases of the skin and subcutaneous tissue
XIII Diseases of the musculoskeletal system and connective tissue

XIV Diseases of the genitourinary system
XV Pregnancy, childbirth and the puerperium
XVI Certain conditions originating in the perinatal period
XVII Congenital malformations, deformations and chromosomal abnormalities XVIII Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified
XIX Injury, poisoning and certain other consequences of external causes
XX External causes of morbidity and mortality
XXI Factors influencing health status and contact with health services.
And 7 categories without (CID) patient follow-up (22), medical consultation (23), blood donation (24), laboratory examination (25), unjustified absence (26), physiotherapy (27), dental consultation (28).

3. Month of absence
4. Day of the week (Monday (2), Tuesday (3), Wednesday (4), Thursday (5), Friday (6))
5. Seasons (summer (1), autumn (2), winter (3), spring (4))
6. Transportation expense
7. Distance from Residence to Work (kilo meters)
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)

Given below is a sample of the data set that we are using:

	ID	Reason_for_absence	Month_of_absence	Day_of_the_week	Seasons	Transportation_expense	Distance_from_Residence_to_Work	Service_time	Age
0	11	26	7	3	1	289.0	36.0	13.0	33.0
1	36	NaN	7	3	1	118.0	13.0	18.0	50.0
2	3	23	7	4	1	179.0	51.0	18.0	38.0
3	7	7	7	5	1	279.0	5.0	14.0	39.0
4	11	23	7	5	1	289.0	36.0	13.0	33.0

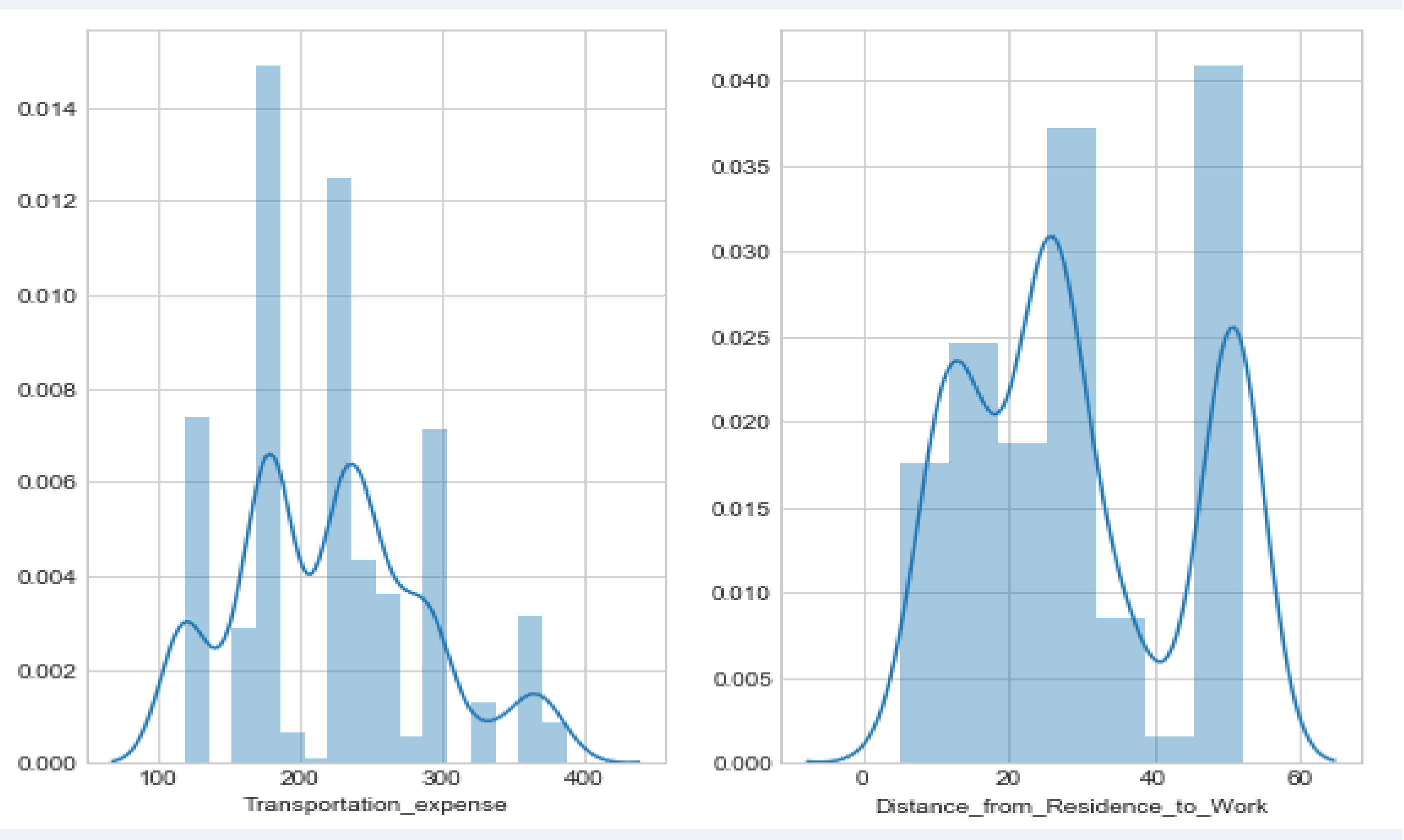
rk_load_Average/day	...	Disciplinary_failure	Education	Son	Social_drinker	Social_smoker	Pet	Weight	Height	Body_mass_index	Absenteeism_time_in_hours
239554.0	...	0.0	1.0	2.0	1.0	0.0	1.0	90.0	172.0	30.0	4.0
239554.0	...	1.0	1.0	1.0	1.0	0.0	0.0	98.0	178.0	31.0	0.0
239554.0	...	0.0	1.0	0.0	1.0	0.0	0.0	89.0	170.0	31.0	2.0
239554.0	...	0.0	1.0	2.0	1.0	1.0	0.0	68.0	168.0	24.0	4.0
239554.0	...	0.0	1.0	2.0	1.0	0.0	1.0	90.0	172.0	30.0	2.0

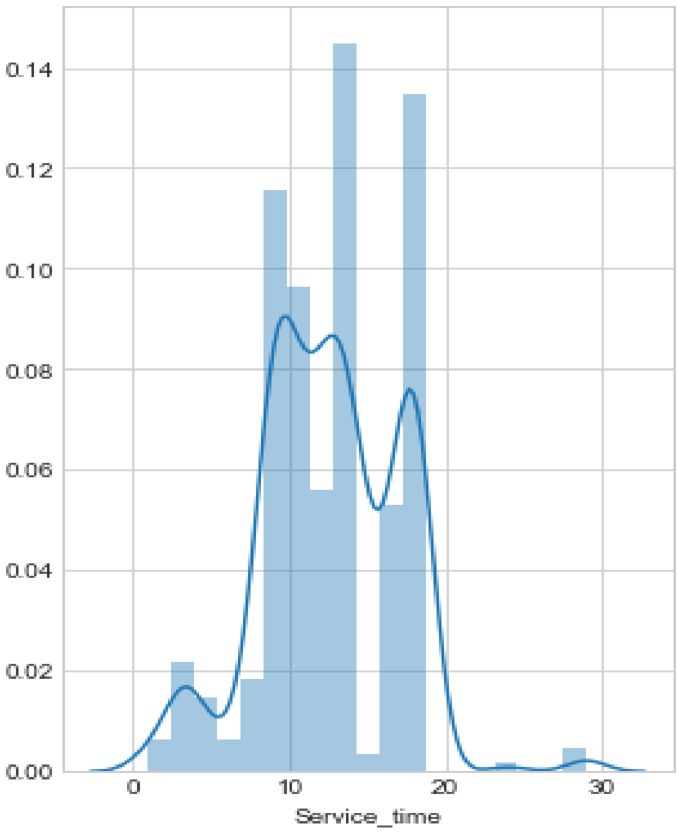
Chapter 2
Methodology

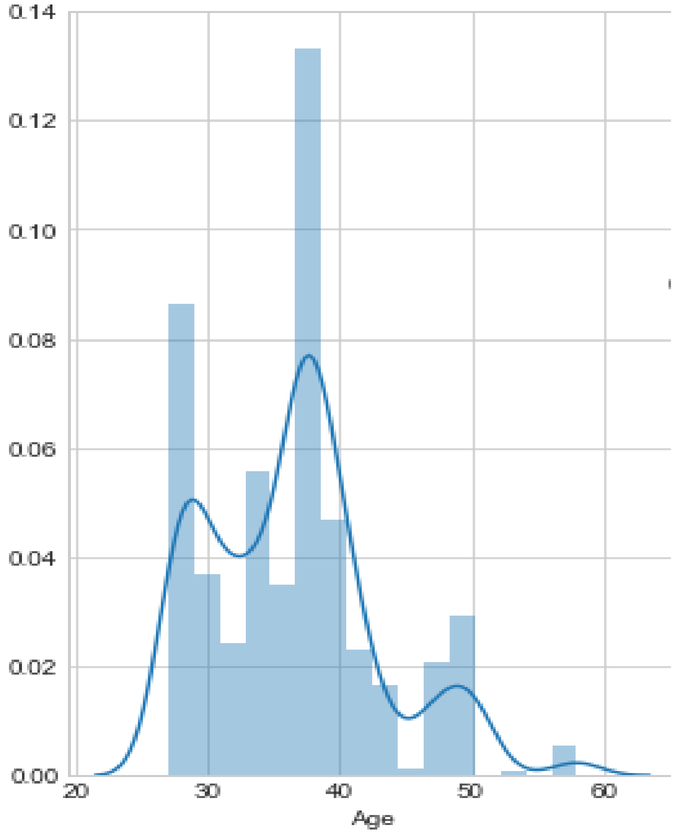
2.1 Pre Processing

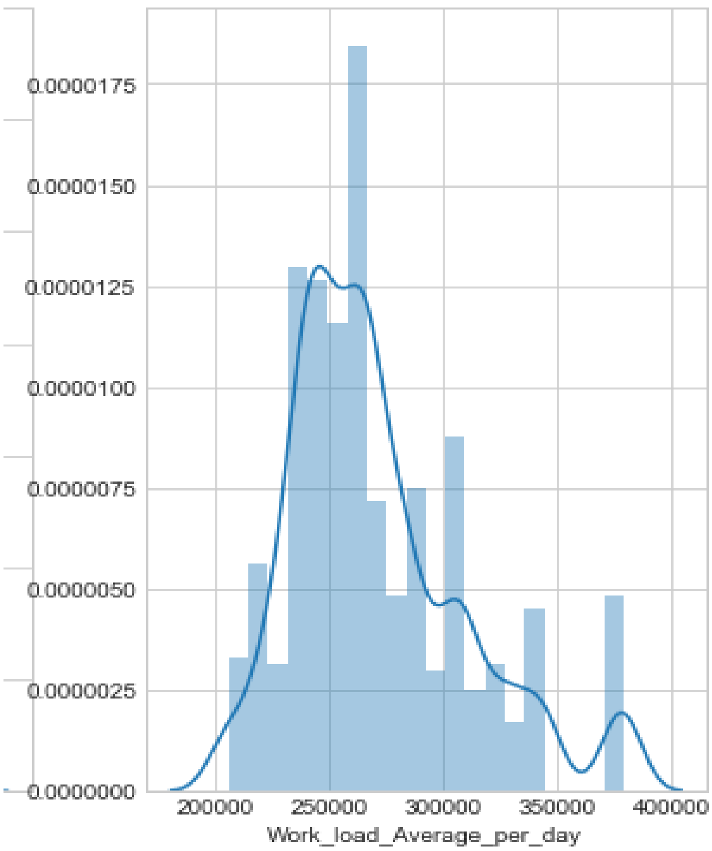
Any predictive modelling requires that we look at the data before we start modelling. However, in data mining terms looking at data refers to so much more than just looking. Looking at data refers to exploring the data, cleaning the data as well as visualizing the data through graphs and plots. This is often called as Exploratory Data Analysis. To start this process, we will first try and look at all the distributions of the variables. Most analysis like regression, require the data to be normally distributed. We can visualize our data and check its distribution of Continuous variables:

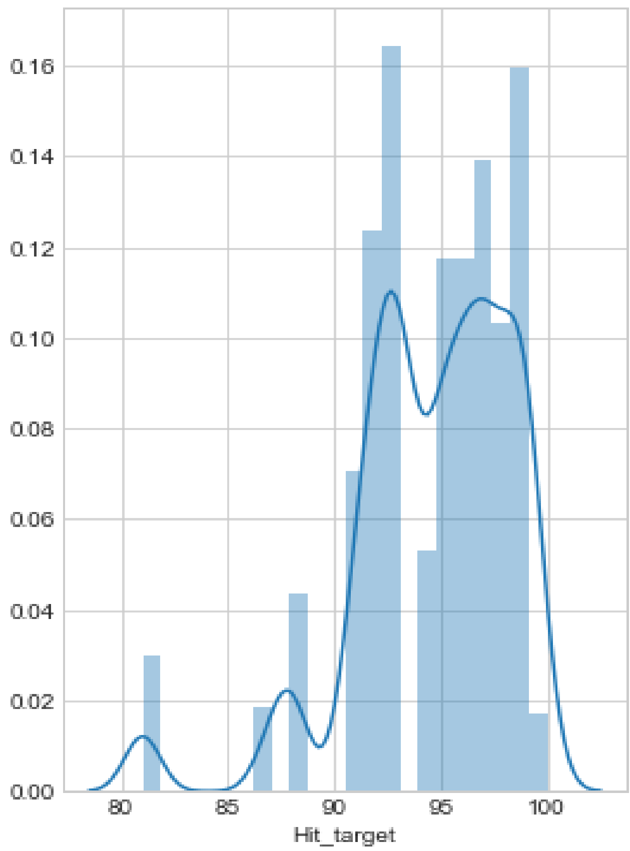
2.1.1 Data Distribution:

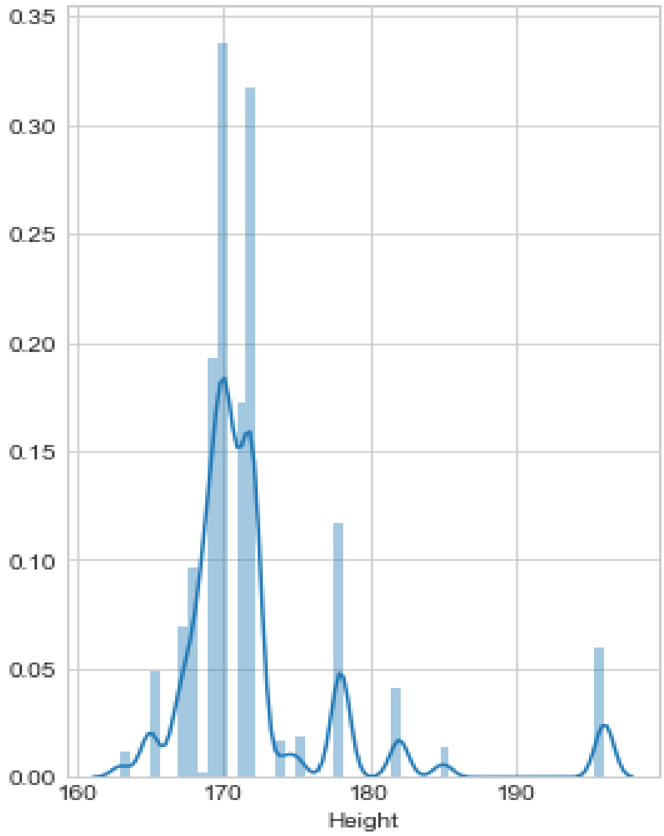



0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
0
10
20
30
Service_time

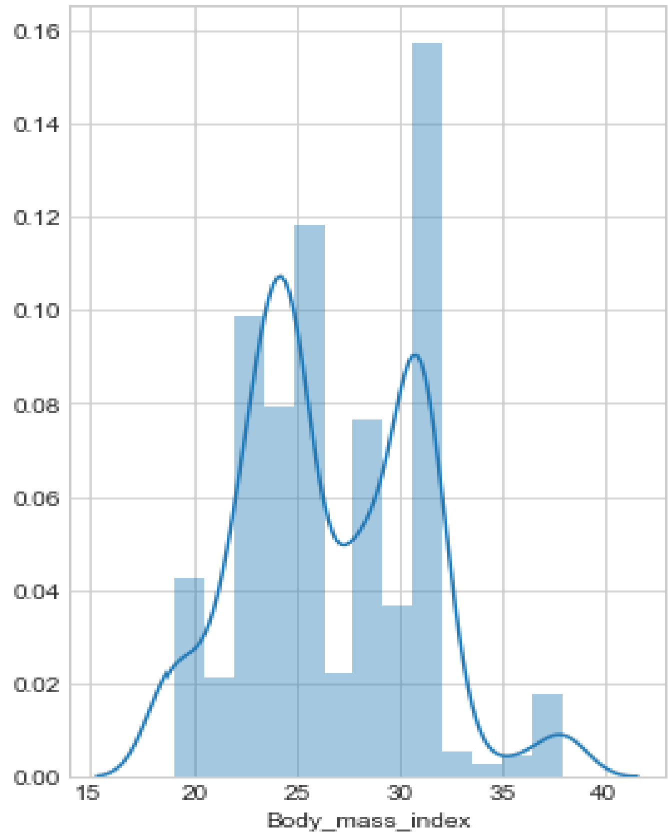

0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
20
30
40
50
60
Age


0.0000175
0.0000150
0.0000125
0.0000100
0.0000075
0.0000050
0.0000025
0.0000000
200000
250000
300000
350000
400000
Work_load_Average_per_day

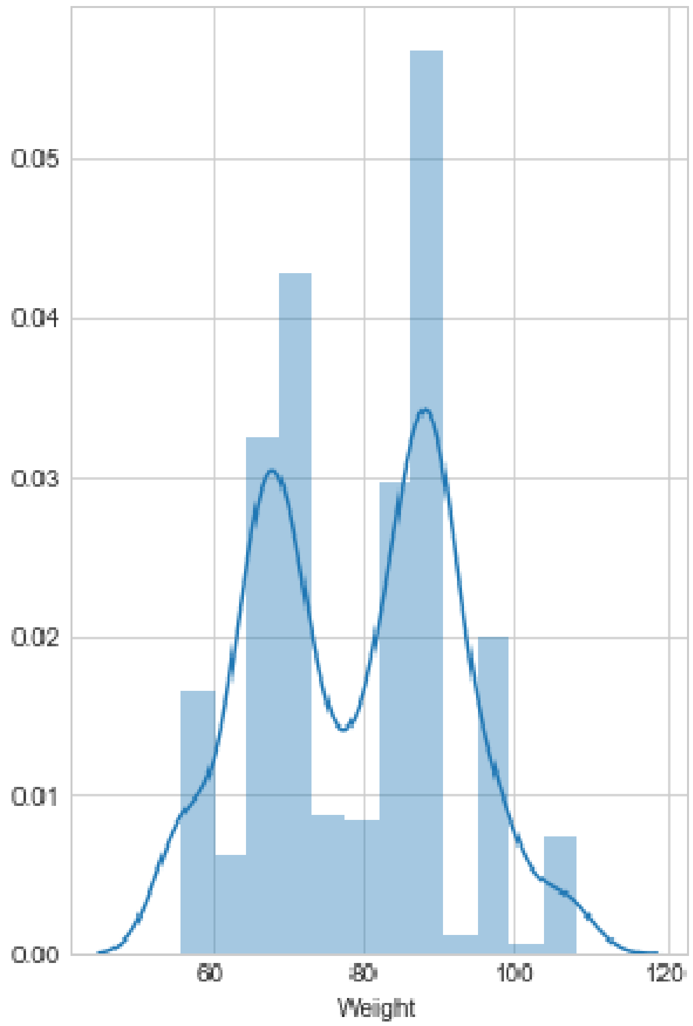

0.16
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
80
85
90
95
100
Hit_target



0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
160
170
180
190
Height



0.16
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
15
20
25
30
35
40
Body_mass_index



0.05
0.04
0.03
0.02
0.01
0.00
60
80
100
120
Weight

As we can see in the above graphs some variables are normally distributed and some are not.

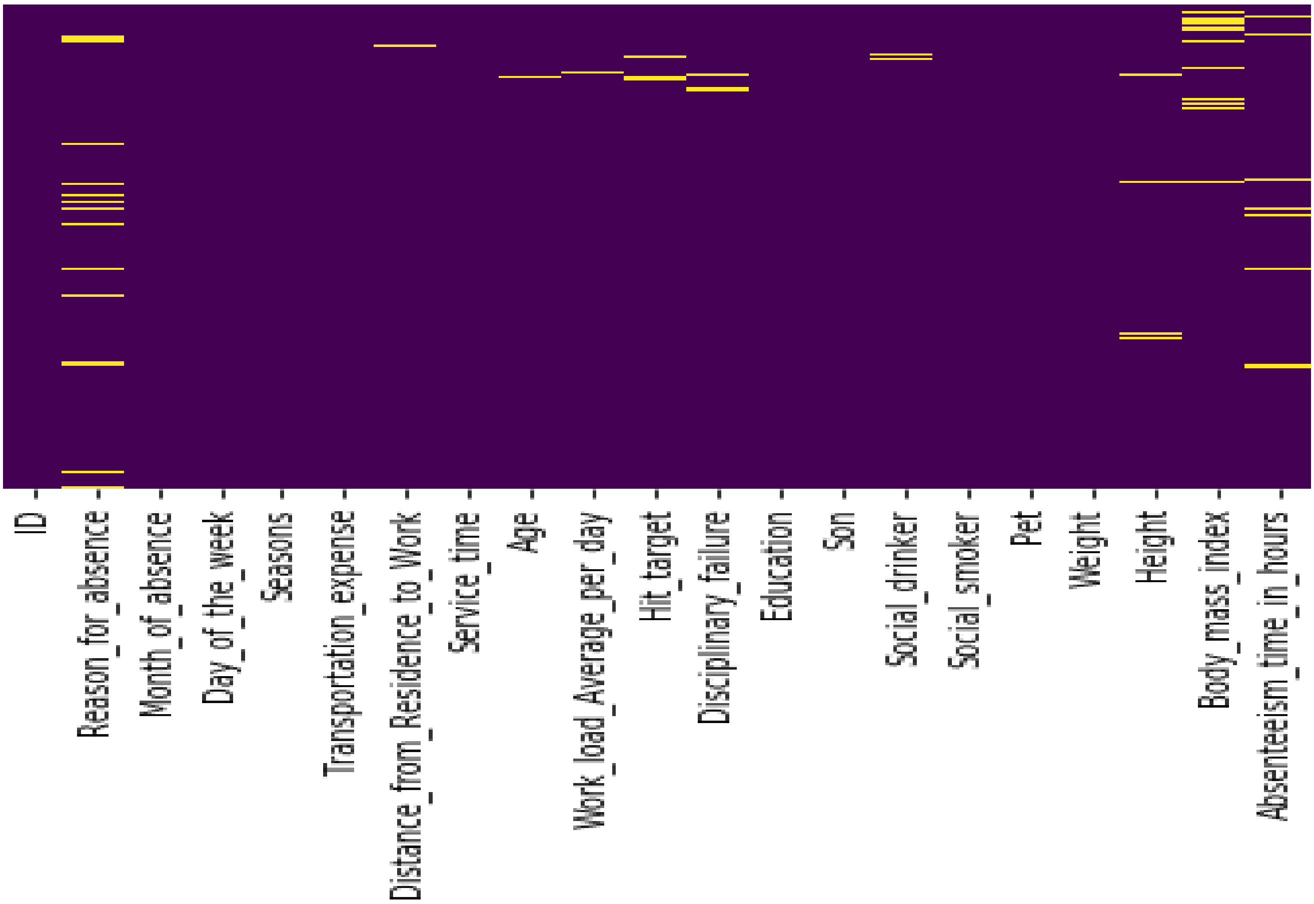
2.1.2 Data Statistics

	Transportation_expense	Distance_from_Residence_to_Work	Service_time	Age	Work_load_Average_per_day	Hit_target
count	740.000000	740.000000	740.000000	740.000000	740.000000	740.000000
mean	221.355742	29.659219	12.554054	36.448222	271176.375697	94.587840
std	66.899163	14.849632	4.384873	6.477678	38821.256046	3.779312
min	118.000000	5.000000	1.000000	27.000000	205917.000000	81.000000
25%	179.000000	16.000000	9.000000	31.000000	244387.000000	93.000000
50%	225.000000	26.000000	13.000000	37.000000	264249.000000	95.000000
75%	260.000000	50.000000	16.000000	40.000000	284853.000000	97.000000
max	388.000000	52.000000	29.000000	58.000000	378884.000000	100.000000

	Weight	Height	Body_mass_index
count	740.000000	740.000000	740.000000
mean	79.035135	172.126008	26.683273
std	12.883211	6.033813	4.277049
min	56.000000	163.000000	19.000000
25%	69.000000	169.000000	24.000000
50%	83.000000	170.000000	25.000000
75%	89.000000	172.000000	31.000000
max	108.000000	196.000000	38.000000

2.1.3 Missing Value Analysis:

In the dataset there are several observations present which contains missing values. In the below pic yellow lines represent the missing values in that respective column



So to deal to missing value we can fill the missing values using various techniques:

1. Mean
2. Median
3. KNN Imputation

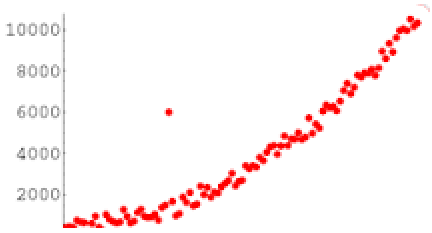
To find the best fit technique, we can replace any 1 known observation and then we will apply all 3 techniques and then we will check which is giving us the best result:

Eg:
#Actual #dataset[1, 6] = 289
#Mean = 220.9426
#Median = 225
#KNN = 289

So here we can see that KNN is given us the exact value so we will use KNN imputation to fill the missing values.

2.1.4 Outlier Analysis:

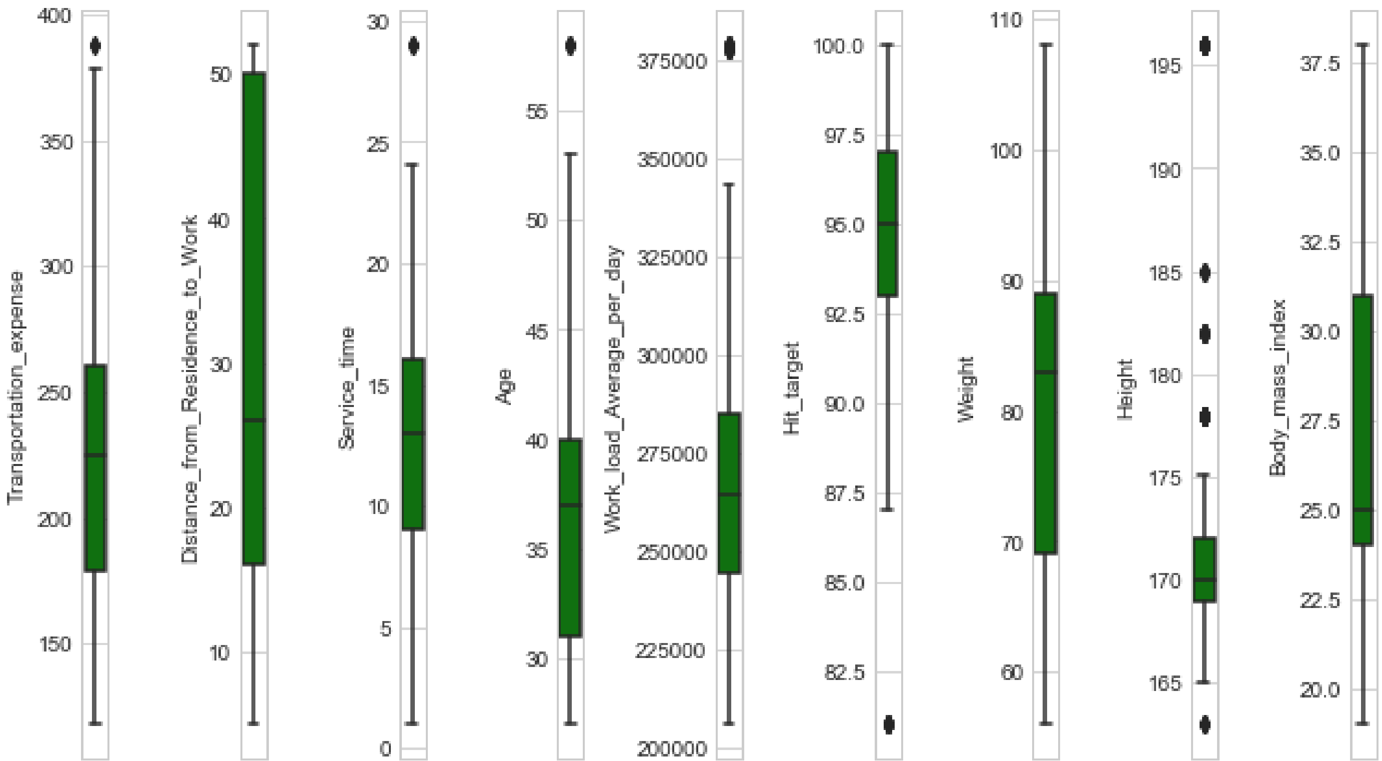
One of the important steps in data pre-processing is outlier analysis. In statistics, an Outlier is an observation point that is distant from other observations. An outlier may be due to variability in the measurement or it may indicate experimental error; the latter are sometimes excluded from the data set. Outliers are only present in continuous variable not in categorical variable.



An **outlier** can cause serious problems in statistical analyses.
So to prevent these problems we need to take care of outliers that means we need to either remove that observation or replace the outlier using missing values and then refill them using missing values analysis.

First we need to identify outliers:

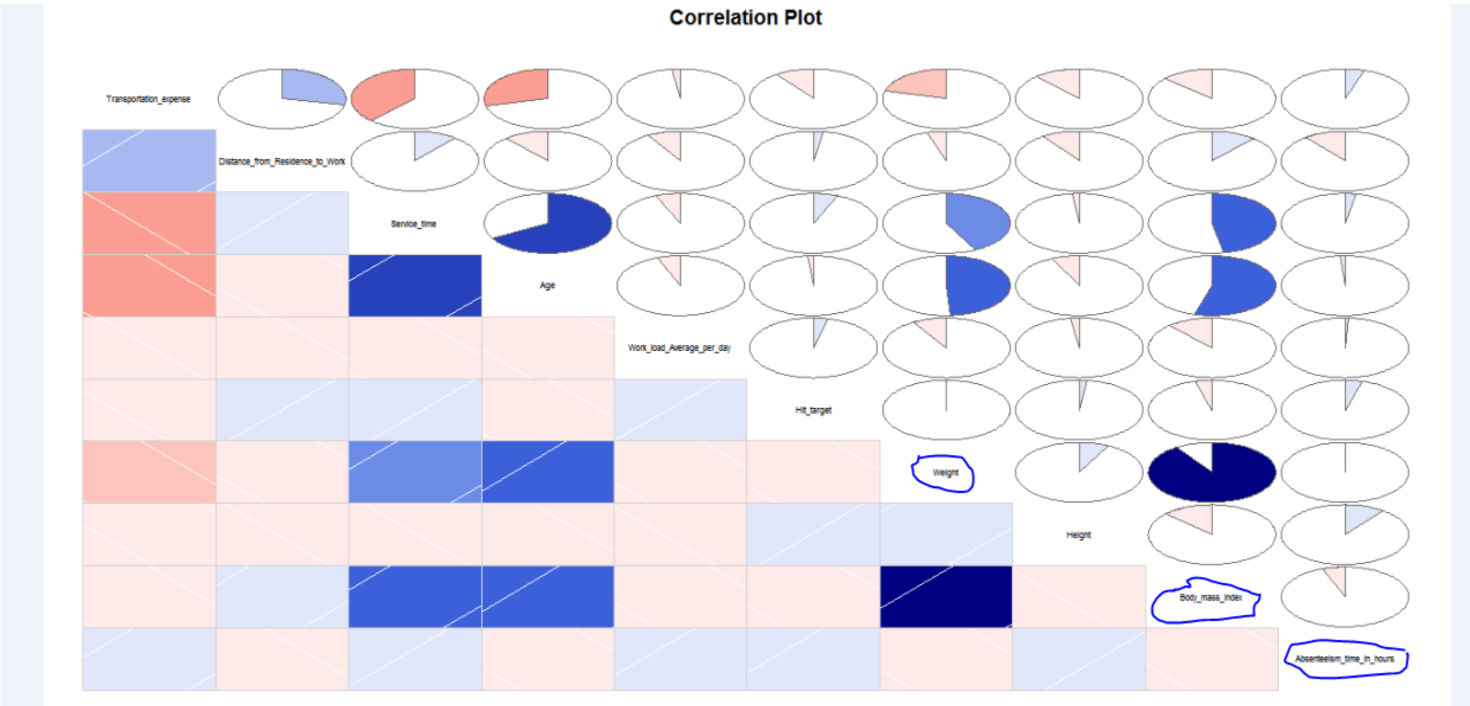
We can identify using **Boxplot.** In the Boxplot the points that are present above the header line or below the tailor line are the outliers



2.1.5 Feature Selection:

Before performing any type of modelling we need to make sure that all the variables which we will use in the model must have unique characteristics there should be no correlation between independent variables so here I used correlation plot to identify the correlation between continuous variable: (Python)

(In R)



So, as we can see in the above plot that weight and body mass index are very highly correlated so we can remove any one of them and before removing them we need to identify which one has the less impact on the target variable, in this case weight has less impact on the target variable so here I am removing weight.

Correlation between Categorical variable: To identify correlation between categorical variable we can use Chi-Square test.

P-Value relation between categorical variable (only those who have p-value<0.05)
"Social_drinker VS Social_smoker" : 0.00406
"Education VS Social_smoker": 2.2e-16
"Education VS Social_drinker":2.2e-16
"Disciplinary_failure VS Social_smoker"= 0.003241
"Month_of_absence VS Social_smoker"= 0.02312
"Month_of_absence VS Social_drinker"= 0.007571

Here we can see that social smoker is dependent on almost all other independent variable
So we can remove it

2.1.6 Feature Scaling:

Our final goal is to identify the reason behind absenteeism and how to prevent absenteeism, So to identify the solution for it feature scaling is not required actually, we just need to analyse the data patterns.

2.2 Analizations and Solutions

Que1: What changes company should bring to reduce the number of absenteeism?
Ans1: To identify the solution for this question we need to identify which independent variable contributing more in increasing the absenteeism hours.
To identify it model is not required, we just need to analyse out data.

There are 2 types of independent variables available in our dataset:

1. Continuous Variable
2. Categorical variable

1. **Continuous Variable:** To identify the impact of continuous variable on target variable we need to group the values in target variable by taking mean of independent variable.

bsenteeism_time_in_hou	ansportation_expen	stance_from_Residence_to_Wo	Service_time	Age	Vork_load_Average_per_da	Hit_target	Height	Body_mass_index	Count
0	242.56106	26.694444	12.444444	39.444444	268425.8	93.738086	170.25613	28.522477	36
1	209.32812	30.099125	13.221786	37.244508	274025.29	94.083871	170.21626	27.158747	89
2	198.10191	28.038217	12.278424	35.665319	263727.13	95.433121	170.03574	25.947118	157
3	199.35381	32.598214	12.92619	35.870256	264526.69	95.500924	169.90192	27.035411	112
4	232.66667	33.818182	12.5	36.212121	255642.21	94.224286	169.72633	26.386628	66
5	227.88889	25.666667	11	35.333333	267045.61	93.666667	170.66667	24.888889	9
6	303	40	11	32.333333	264428.26	94.333333	170.33333	27.666667	3
7	361	52	3	28	205917	92	172	27	1
8	244.30016	29.641791	11.970149	35.798916	267235.94	95.076544	170.41411	27.031146	201
12	330	16	4	28	205917	92	171.67074	25	1
16	224.52632	26.789474	11.894737	34.368421	284364.3	94.842105	169.83189	25.105263	19
18	246	25	16	40.904034	291510.69	94	170	23	1
21	118	13	18	50	253465	93	171.88735	30.963162	1
24	222.875	27.8125	14.375	37.0625	289177.37	94.509372	170.76515	26.75	16
25	157	27	6	29	265017	88	171	22	1
32	217.1998	22.8	13	37	289482.2	95.4	170.82142	26.2	5
40	261.14286	32.714286	11.571429	36.857143	283785.14	94.211551	170.28102	24.571429	7
48	155	12	14	34	237656	99	170.60633	25	1
56	189	30	10.5	36.5	291141.25	96.5	170	25.5	2
64	199	20.666667	10.666667	36.666667	254659.67	94.666667	172.66667	24.093918	3
80	214	14	14.666667	38.333333	266570.26	95	170.45802	27	3

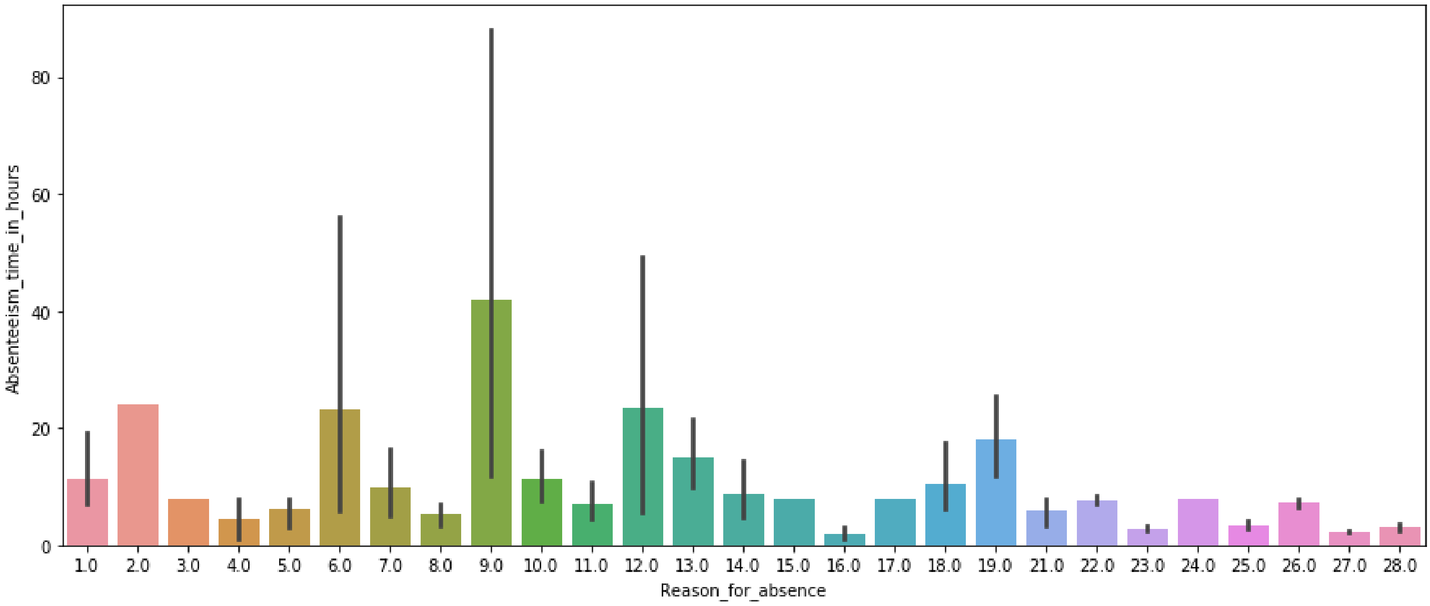
a. In above table we can see that mostly those employees who distance of residence from office is above 26 have more absenteeism hours.

Solution: Company should provide better residence options to their employees near the office.

b. Company should decrease the Working load average per day because higher working load leads to high absenteeism hours

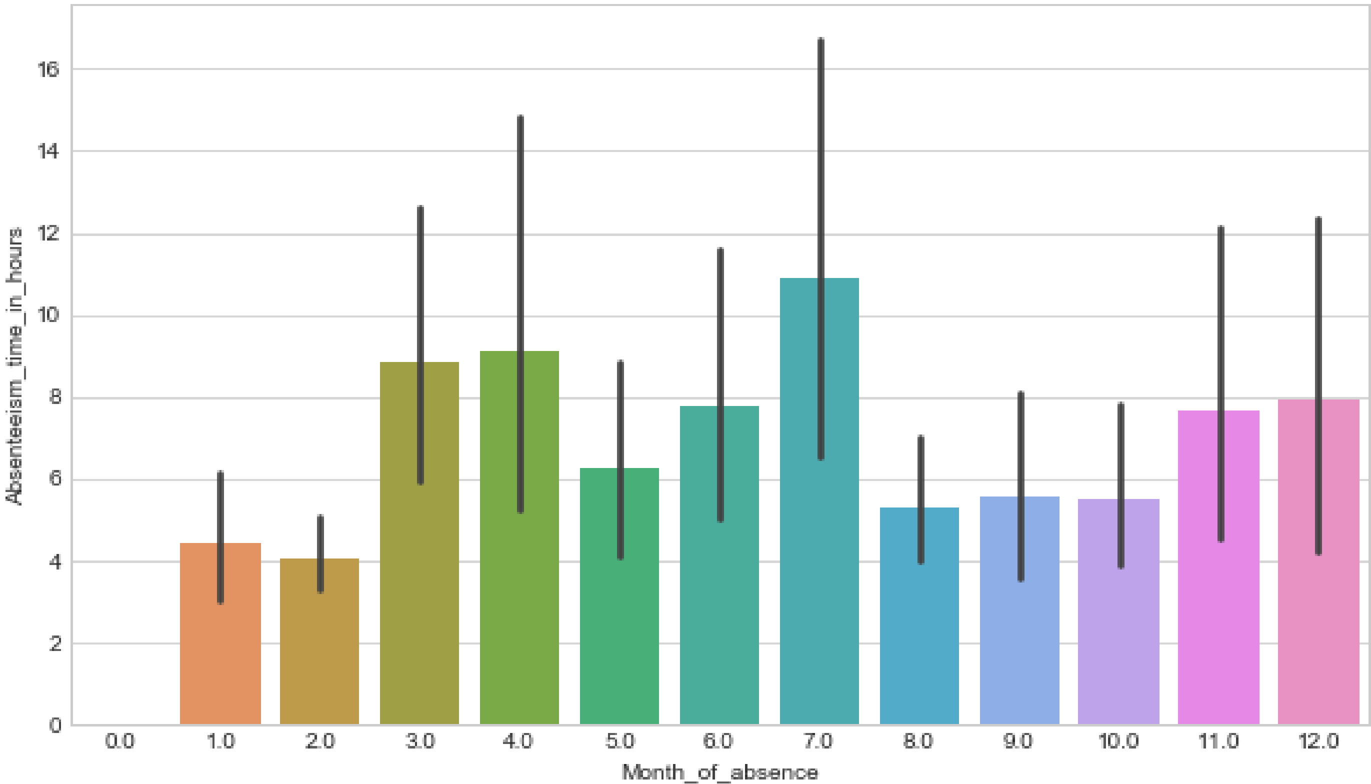
2. **Categorical Variables:** We can identify the impact of categorical variable on target using graphs.

a. Reason for absence:



As we can see in the above graph because of "(9) Diseases of the circulatory system" reason absenteeism hours are very high so company should provide some good Doctor consultancy to their employees to prevent the employees from this disease.

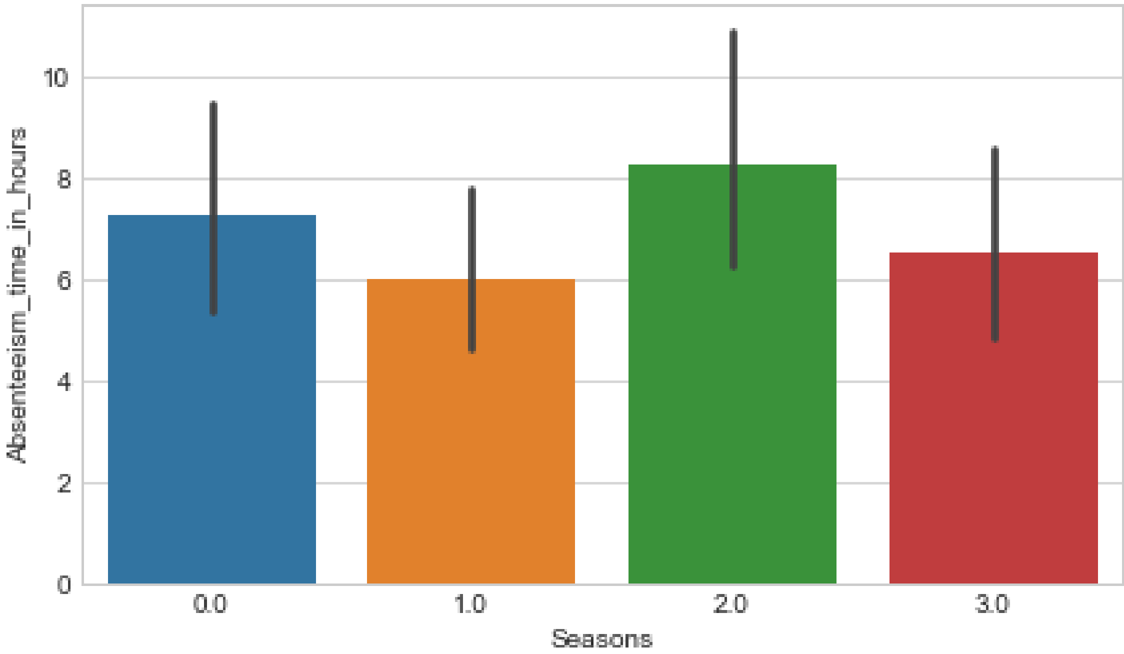
b. Month of absence:



In the month of March, April, July, November and December absenteeism hours are very high.

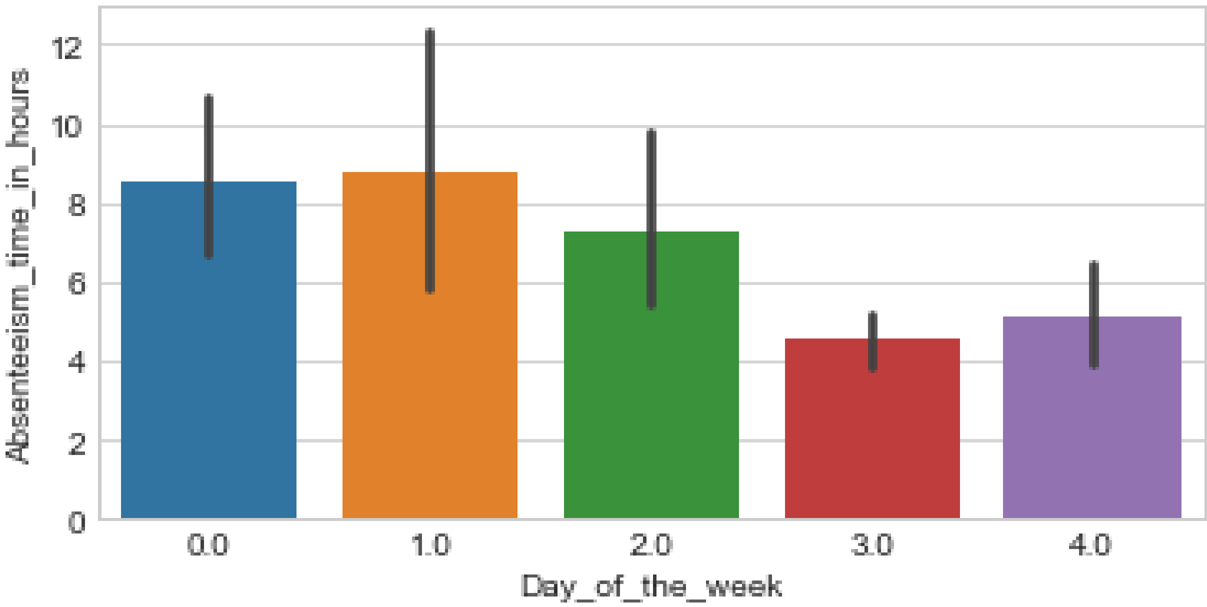
Solution: Company should organise some functions like: Annual function, Competitions etc.In the month of March, April, July, November and December to decrease the absenteeism time

c. Season: (summer (0), autumn (1), winter (2), spring (3))



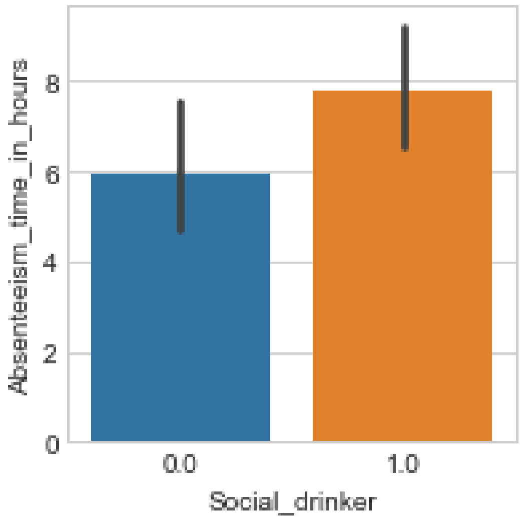
As we can see in the above graph in summer and winter absenteeism time is high.

d. Day of the Week: Monday (0), Tuesday (1), Wednesday (2), Thursday (3), Friday (4)



As we can see in the above graph on Monday and Tuesday absenteeism hours are very high, since it is starting of week, company should organise some motivational and fun activities on these days to keep employees more delighted and motivated.

e. Social Drinker: (0: No, 1: Yes)



As we can see those employees who drinks have high absenteeism hours, so company should organize some campaign to make their employees stop drinking.

Que2: How much losses every month can we project in 2011 if same trend of absenteeism continues?

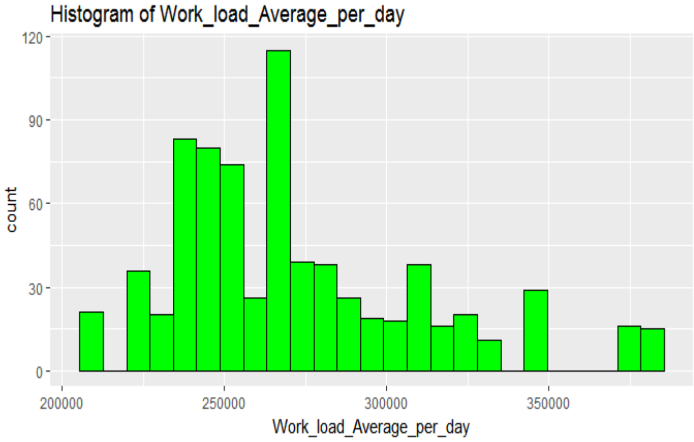
Ans2: To find out the solution for this we need to find the mean of absenteeism hours' group by months.

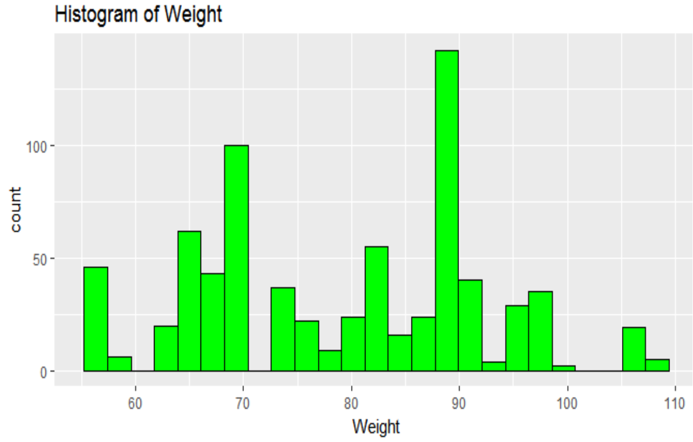
Month_of_absence	Absenteeism_time_in_hours
1	4.44
2	4.0833333
3	8.8505747
4	9.0943396
5	6.28125
6	7.7962963
7	10.895522
8	5.2962963
9	5.5660377
10	5.5352113
11	7.6507937
12	7.9183673

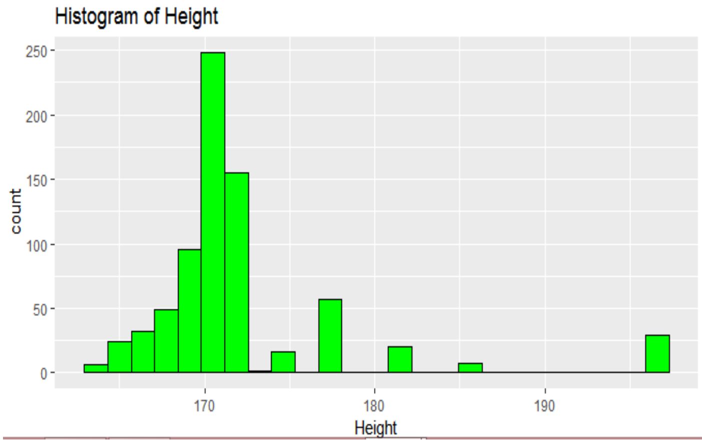
Chapter 3
Conclusion

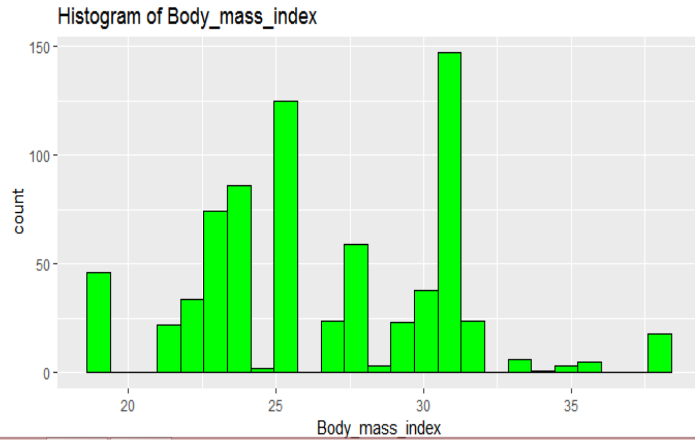
According to the data analysis company must focus on prevent their employees from diseases and also do some fun activities and functions in March, April, July, November and December.

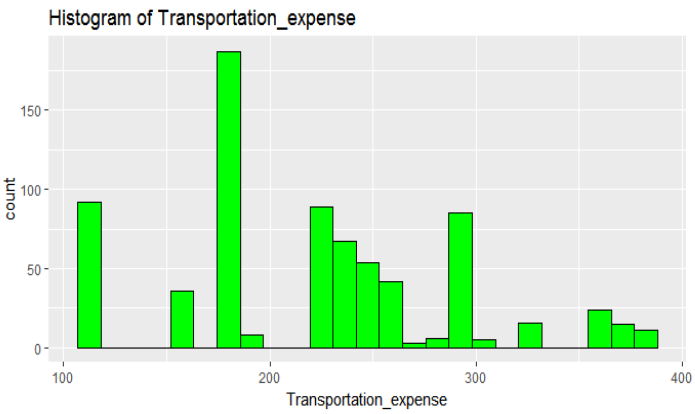
Appendix A - Extra Figures

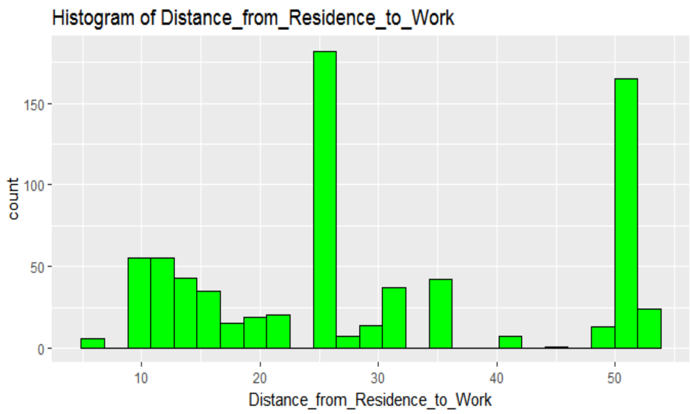


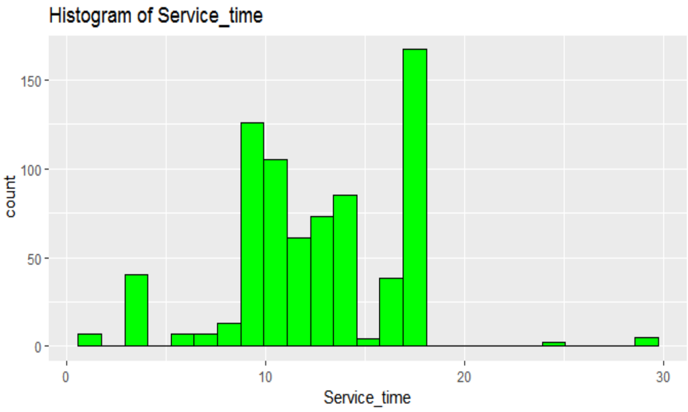


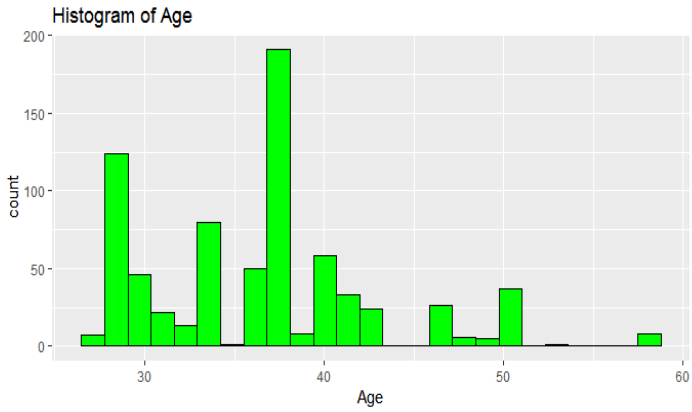












Appendix B - Python Code:

```
# -*- coding: utf-8 -*-
"""
Created on Sat May  4 23:27:13 2019

@author: vinayak
"""

#Load libraries

import os
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from fancyimpute import KNN
from scipy.stats import chi2_contingency

#Set working directory
os.chdir("C:/Users/vinayak\Desktop/EmployeeAbsenteesm")

#Load data
df = pd.read_excel("Absenteeism_at_work_Project.xls")
#Remove space and characters from column names
df.columns = df.columns.str.strip().str.replace('/',' per ').str.replace(' ','_')

##################################### Exploratory Data Analysis
##################################################
#Information about datatype of columns
df.info()
df['Reason_for_absence'] = df['Reason_for_absence'].replace(0, np.nan)

#Univariate Analysis and Variable Consolidation --> Transform into proper data type
cnumber_factor = [0,1,2,3,4,11,12,13,14,15,16]
for i in cnumber_factor:
    df.iloc[:,i] = df.iloc[:,i].astype('object')

df.info()

#Categorical variable
cnames_factor = df.select_dtypes(include=['object']).columns
#Numeric variable
```

```
cnumber_numeric = [5,6,7,8,9,10,17,18,19,20]
cnames_numeric = df.select_dtypes(exclude=['object']).columns
#Remove target variable from cnames_numeric
cnames_numeric = cnames_numeric.drop('Absenteeism_time_in_hours')

##################################Missing value
analysis###############################################
sns.heatmap(df.isnull(),cbar=False,yticklabels=False,cmap = 'viridis')

missing_val = pd.DataFrame(df.isnull().sum())

#Reset index
missing_val = missing_val.reset_index()

#Rename variable
missing_val = missing_val.rename(columns = {'index': 'Variables', 0: 'Missing_count'})

#descending order
missing_val = missing_val.sort_values('Missing_count', ascending =
False).reset_index(drop = True)

#save output results
missing_val.to_csv("Missing_count.csv")

missing_val['Missing_count'].sum()
#There are 178 missing values are present in the dataset so we need to perform missing
value analysis.

#KNN imputation
#Assigning levels to the categories
lis = []
for i in range(0, df.shape[1]):
    if(df.iloc[:,i].dtypes == 'object'):
        df.iloc[:,i] = pd.Categorical(df.iloc[:,i])
        df.iloc[:,i] = df.iloc[:,i].cat.codes
        df.iloc[:,i] = df.iloc[:,i].astype('object')
        lis.append(df.columns[i])

#replace -1 with NA to impute
for i in range(0, df.shape[1]):
    df.iloc[:,i] = df.iloc[:,i].replace(-1, np.nan)

#Apply KNN imputation algorithm
df = pd.DataFrame(KNN(k = 3) .fit_transform(df), columns = df.columns)
```

```
#Convert into proper datatypes
for i in lis:
    df.loc[:,i] = df.loc[:,i].round()
    df.loc[:,i] = df.loc[:,i].astype('object')

#################################### Analyze Data Insights
############################################

df[cnames_numeric].describe().to_csv("C:/Users/vinayak/Desktop/EmployeeAbsenteesm/ab.
csv")
df[cnames_numeric].describe()

#Analyze Distribution
number_of_columns=9
number_of_rows = len(cnames_numeric)-1/number_of_columns
plt.figure(figsize=(5*number_of_columns,8*number_of_rows))
for i in range(0,len(cnames_numeric)):
    plt.subplot(number_of_rows + 1,number_of_columns,i+1)
    sns.distplot(df[cnames_numeric[i]],kde=True)


####################################Outlier
Analysis##################################################
plt.figure(figsize=(number_of_columns,5*number_of_rows))
for i in range(0,len(cnames_numeric)):
    plt.subplot(number_of_rows + 1,number_of_columns,i+1)
    sns.set_style('whitegrid')
    sns.boxplot(df[cnames_numeric[i]],color='green',orient='v')
    plt.tight_layout()


#Detect and replace with NA
#Extract quartiles

for i in cnames_numeric:
    q75, q25 = np.percentile(df.loc[:,i], [75 ,25])
    #Calculate IQR
    iqr = q75 - q25
    #Calculate inner and outer fence
    minimum = q25 - (iqr*1.5)
    maximum = q75 + (iqr*1.5)
    #Replace with NA
    df.loc[df.loc[:,i] < minimum,i] = np.nan
```

```
    df.loc[df.loc[:,i] > maximum,i] = np.nan

missing_val = pd.DataFrame(df.isnull().sum())

#Apply KNN imputation algorithm
df = pd.DataFrame(KNN(k = 3) .fit_transform(df), columns = df.columns)

#Convert into proper datatypes
for i in lis:
    df.loc[:,i] = df.loc[:,i].round()
    df.loc[:,i] = df.loc[:,i].astype('object')

df.loc[:,'Absenteeism_time_in_hours'] = df.loc[:,'Absenteeism_time_in_hours'].round()
#####################################Feature
Selection##################################################
#Remove the variable that are not useful for the analysis

df_corr = df.loc[:,cnames_numeric]
#Set the width and hieght of the plot
f, ax = plt.subplots(figsize=(10, 10))
#Generate correlation matrix
corr = df_corr.corr()
#Plot using seaborn library
sns.heatmap(corr, mask=np.zeros_like(corr, dtype=np.bool), cmap = 'viridis',
            square=True, ax=ax, annot=True)

#As we can see in the plot weight and body mass index are very highly +ve correlated so we
can remove 1 of them
#And weight is less related to AbsenteeismHour as compared to body mass index so I am
removing weight


for i in range(0,len(cnames_factor)):
    for j in range(i+1,len(cnames_factor)):
        print(cnames_factor[i], " VS ", cnames_factor[j])
        chi2, p, dof, ex = chi2_contingency(pd.crosstab(df[cnames_factor[i]],
df[cnames_factor[j]]))
        print(p)

#In the Ch-square test we can see that social smoker is dependent on almost all other
independent variable
# So we can remove it
df = df.drop(["Social_smoker","Weight"], axis=1)
cnames_numeric=cnames_numeric.drop('Weight')
```

```
cnames_factor = cnames_factor.drop('Social_smoker')
#################################### Feature Scaling
###################################################
#Normality check - Done in Analyze Data Insights :: Data is not normally distributed
# Apply normalization

#plt.hist(df['Transportation_expense'], bins='auto')

#Nomalisation
#for i in cnames_numeric:
  #  print(i)
  #  df[i] = (df[i] - min(df[i]))/(max(df[i]) - min(df[i]))

#No need to apply feature scaling as in our problem we need to identify the reason for
absenteeism
#nothing to predict here (Human Readable -- Actual Values)

#################################### Result
###################################################

#Que1: What changes company should bring to reduce the number of absenteeism?
count = pd.DataFrame(df['Absenteeism_time_in_hours'].value_counts()).sort_index()
numeric_impact = df.groupby('Absenteeism_time_in_hours')[cnames_numeric].mean()
count = count.reset_index()
numeric_impact = numeric_impact.reset_index()
count = count.rename(columns = {'index': 'Absenteeism_time_in_hours',
'Absenteeism_time_in_hours': 'Count'})

result1 = pd.merge(numeric_impact, count, on='Absenteeism_time_in_hours')

plt.figure(figsize=(15,6))
sns.barplot(data=df, x="Reason_for_absence", y="Absenteeism_time_in_hours")

plt.figure(figsize=(10,6))
sns.barplot(data=df, x="Month_of_absence", y="Absenteeism_time_in_hours")

plt.figure(figsize=(7,4))
sns.barplot(data=df, x="Seasons", y="Absenteeism_time_in_hours")

plt.figure(figsize=(3,3))
sns.barplot(data=df, x="Social_drinker", y="Absenteeism_time_in_hours")

plt.figure(figsize=(6,3))
sns.barplot(data=df, x="Day_of_the_week", y="Absenteeism_time_in_hours")
```

#Que2: How much losses every month can we project in 2011 if same trend of absenteeism continues?
result2 = df.groupby('Month_of_absence')['Absenteeism_time_in_hours'].mean()
result2 = result2.reset_index()

Appendix C - R Code:

```
#Clear the environment
rm(list=ls())

#set the working directory
setwd(dir = "C:/Users/vinayak/Desktop/EmployeeAbsenteesm")

#Load Libraries
x = c("ggplot2", "corrgram", "DMwR", "caret", "randomForest", "unbalanced", "C50",
"dummies", "e1071", "Information",
      "MASS", "rpart", "gbm", "ROSE", 'sampling', 'DataCombine',
'inTrees',"usdm","randomForest","e1071","plyr", "dplyr")

install.packages(x)
lapply(x, require, character.only = TRUE)
rm(x)

#Load the dataset
dataset = readxl::read_excel("Absenteeism_at_work_Project.xls")
dataset = as.data.frame(dataset)
removeCharacter <- function(x) {colnames(x) <- gsub("/", " per ", colnames(x));x}
spaceless <- function(x) {colnames(x) <- gsub(" ", "_", colnames(x));x}
dataset <- removeCharacter(dataset)
dataset <- spaceless(dataset)
#################################### Exploratory Data Analysis
####################################################

#Check the structure of the dataset
str(dataset)

dataset$Reason_for_absence[dataset$Reason_for_absence %in% 0] = NA

#Univariate Analysis and Variable Consolidation --> Transform into proper data type
factor_col_no = c(1,2,3,4,5,12,13,14,15,16,17)
dataset[,factor_col_no] <- lapply(dataset[,factor_col_no] , factor)
```

```
#Check the structure of the dataset
str(dataset)

######################################Missing value
analysis##################################################
sum(is.na(dataset))

#There are 178 missing values in the dataset so we need to perform missing value analysis.
missing_val = data.frame(apply(dataset,2,function(x){sum(is.na(x))}))
missing_val$Columns = row.names(missing_val)
names(missing_val)[1] =  "Missing_percentage"
missing_val$Missing_percentage = (missing_val$Missing_percentage/nrow(dataset)) * 100
missing_val = missing_val[order(-missing_val$Missing_percentage),]
row.names(missing_val) = NULL
missing_val = missing_val[,c(2,1)]
write.csv(missing_val, "Miising_perc.csv", row.names = F)

ggplot(data = missing_val[1:3,], aes(x=reorder(Columns, -Missing_percentage),y =
Missing_percentage))+
  geom_bar(stat = "identity",fill = "grey")+xlab("Parameter")+
  ggtitle("Missing data percentage (EmployeeAbsenteeism)") + theme_bw()

#To replace the missing values there are 3 ways
#1. KNN, 2. Mean, 3. Median
#dataset[1, 6] = 289
#dataset[1, 6] = NA
#Mean Method
#dataset$`Transportation expense`[is.na(dataset$`Transportation expense`)] =
mean(dataset$`Transportation expense`, na.rm = T)

#Median Method
#dataset$`Transportation expense`[is.na(dataset$`Transportation expense`)] =
median(dataset$`Transportation expense`, na.rm = T)

# kNN Imputation
dataset = knnImputation(dataset, k = 3)
sum(is.na(dataset))

#Actual #dataset[1, 6] = 289
#Mean = 220.9426
#Median = 225
#KNN = 289
```

```
##################################### Analyze Data Insights (Distribution)
###########################################
summary(dataset)

numeric_index = sapply(dataset,is.numeric) #selecting only numeric
numeric_data = dataset[,numeric_index]

factor_index = sapply(dataset,is.factor)  #selecting only factor
factor_data = dataset[,factor_index]

cnames_numeric = colnames(numeric_data)
cnames_factor = colnames(factor_data)

#Remove target variable from cnames_numeric
cnames_numeric <- cnames_numeric[!cnames_numeric %in%
"Absenteeism_time_in_hours"]

#Analyze Distribution
for(i in 1:length(cnames_numeric)) {
  assign(paste0("gn",i), ggplot(data = dataset, aes_string(x = cnames_numeric[i])) +
geom_histogram(bins = 25, fill="green", col="black")+ ggtitle(paste("Histogram
of",cnames_numeric[i])))
}

gridExtra::grid.arrange(gn1,gn2,gn3,gn4,ncol=2)
gridExtra::grid.arrange(gn5,gn7,gn8,gn9,ncol=2)

##################################### Outlier Analysis
#################################################
for (i in 1:length(cnames_numeric)) {
  assign(paste0("gn",i), ggplot(aes_string(y = (cnames_numeric[i]), x =
"Absenteeism_time_in_hours"), data = subset(dataset))+
        stat_boxplot(geom = "errorbar", width = 0.5) +
        geom_boxplot(outlier.colour="red", fill = "grey" ,outlier.shape=18,
                outlier.size=1, notch=FALSE) +
        theme(legend.position="bottom")+
        labs(y=cnames_numeric[i],x="Absenteeism_time_in_hours")+
        ggtitle(paste("Box plot of AbsenteeismTime for",cnames_numeric[i])))
}
#
## Plotting plots together
gridExtra::grid.arrange(gn1,gn2,gn3,ncol=3)
gridExtra::grid.arrange(gn4,gn5,gn7,ncol=3)
gridExtra::grid.arrange(gn8,gn9,ncol=3)
```

```
for(i in cnames_numeric) {
  val = dataset[,i][dataset[,i] %in% boxplot.stats(dataset[,i])$out]
  #print(length(val))
  dataset[,i][dataset[,i] %in% val] = NA
}
sum(is.na(dataset))

#Impute NA using KNN impute
dataset = knnImputation(dataset, k = 3)
dataset['Absenteeism_time_in_hours'] <-  round(dataset['Absenteeism_time_in_hours'], 0)

#################################### Feature Selection
##################################################
## Correlation Plot
corrgram(dataset[,numeric_index], order = F,
         upper.panel=panel.pie, text.panel=panel.txt, main = "Correlation Plot")

#As we can see in the plot weight and body mass index are very highly +ve correlated so we
can remove 1 of them
#And weight is less related to AbsenteeismHour as compared to body mass index so I am
removing weight
dataset=dataset[, !(colnames(dataset) %in% c("Weight"))]

#Chi-square test for correlation between categorical variable
for (i in 1:length(cnames_factor)) {
  for(j in i+1:length(cnames_factor)) {
    if(j<=length(cnames_factor)) {
      print(paste(names(factor_data)[i], " VS ", names(factor_data)[j]))
      print(chisq.test(table(factor_data[,i],factor_data[,j])))
    }
  }
}

# P-Value relation between categorical variable (only those who have p-value<0.05)
# "Social_drinker  VS  Social_smoker" : 0.00406
# "Education  VS  Social_smoker": 2.2e-16
# "Education  VS  Social_drinker":2.2e-16
# "Disciplinary_failure  VS  Social_smoker"= 0.003241
# "Month_of_absence  VS  Social_smoker"= 0.02312
# Here we can see that social smoker is dependent on almost all other independent variable
# So we can remove it
dataset=dataset[, !(colnames(dataset) %in% c("Social_smoker"))]
```

```
##################################### Feature Scaling
##################################################
#Normality check
# qqnorm(dataset$Transportation_expense)
# hist(dataset$Transportation_expense)
#
# numeric_index = sapply(dataset,is.numeric) #selecting only numeric except
AbsenteeismTime
# numeric_data = dataset[,numeric_index]
#
# cnames_numeric = colnames(numeric_data)
# cnames_numeric = cnames_numeric[-11]
# #Apply Normalization
# for(i in cnames_numeric){
#   print(i)
#   dataset[,i] = (dataset[,i] - min(dataset[,i]))/
#     (max(dataset[,i] - min(dataset[,i])))
# }
#No need to apply feature scaling as in our problem we need to identify the reason for
absenteeism
#nothing to predict here (Human Readable -- Actual Values)
##################################### Result
##################################################
#Clean the environment
rmExcept("dataset")

numeric_index = sapply(dataset,is.numeric) #selecting only numeric
numeric_data = dataset[,numeric_index]

factor_index = sapply(dataset,is.factor)  #selecting only factor
factor_data = dataset[,factor_index]

cnames_numeric = colnames(numeric_data)
cnames_factor = colnames(factor_data)

#selecting only numeric except Absenteeism_time_in_hours
cnames_numeric = cnames_numeric[-9]

#Que1: What changes company should bring to reduce the number of absenteeism?
data  =  group_by(dataset,dataset$Absenteeism_time_in_hours)
result1=summarise(data
            , Transportation_expense         =mean(Transportation_expense)
            , Distance_from_Residence_to_Work = mean(Distance_from_Residence_to_Work)
            , Service_time                   = mean(Service_time)
```

```
            , Age                              = mean(Age)
            , Work_load_Average_per_day        = mean(Work_load_Average_per_day)
            , Hit_target                       = mean(Hit_target)
            , Height                           = mean(Height)
            , Body_mass_index                  = mean(Body_mass_index)
            , Count = n())

ggplot(data = dataset, aes(x=Reason_for_absence, y= Absenteeism_time_in_hours)) + geom_bar(stat = 'identity')

ggplot(data = dataset, aes(x=Month_of_absence, y= Absenteeism_time_in_hours)) + geom_bar(stat = 'identity')

ggplot(data = dataset, aes(x=Seasons, y= Absenteeism_time_in_hours)) + geom_bar(stat = 'identity')

ggplot(data = dataset, aes(x=Social_drinker, y= Absenteeism_time_in_hours)) + geom_bar(stat = 'identity')

ggplot(data = dataset, aes(x=Day_of_the_week, y= Absenteeism_time_in_hours)) + geom_bar(stat = 'identity')

#Que2:  How much losses every month can we project in 2011 if same trend of absenteeism continues?
data  =  group_by(dataset,dataset$Month_of_absence)
result2=summarise(data, Absenteeism_time_in_hours=mean(Absenteeism_time_in_hours), Count = n())
```

Reference:

1. https://www.wikipedia.org/
2. https://edwisor.com/home