SHRI VAISHNAV VIDYAPEETH VISHWAVIDAYALAYA, INDORE

SHRI VAISHNAV INSTITUTE OF INFORMATION TECHNOLOGY, INDORE

JULY-AUGUST 2020

TRAINING PROJECT REPORT

DATA SCIENCE USING PYTHON

"ABSENTEEISM AT WORK"

SUBMITTED BY:

CHHAYA GUPTA
HARSH AGRAWAL
SALONI PATIL
VANSHIKA SAWLE

COMPUTER SCIENCE AND ENGINEERING

REPORT OF TWO WEEK INDUSTRIAL TRAINING AT

WebTek Labs Pvt. Ltd., Kolkata

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR

THE AWARD

OF THE DEGREE OF BACHELOR OF TECHNOLOGY

IN COMPUTER SCIENCE & ENGINEERING

JULY-AUGUST 2020

SUBMITTED BY:

CHHAYA GUPTA
HARSH AGRAWAL
SALONI PATIL
VANSHIKA SAWLE

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
SHRI VAISHNAV INSTITUTE OF INFORMATION TECHNOLOGY, INDORE
SHRI VAISHNAV VIDYAPEETH VISHWAVIDALAYA, INDORE

CANDIDATE'S DECLARATION

We hereby declare that we have undertaken industrial training at **'WEBTEK PVT. LTD., KOLKATA'** during a period from **27th July 2020** to **15th August 2020** in partial fulfillment of requirements for the award of degree of Bachelor of Technology in Computer Science and Engineering at **Shri Vaishnav Institute of Information Technology, Indore.**

The work which is being presented in the training report submitted to **Department of Computer Science and Engineering** at **Shri Vaishnav Institute of Information Technology, Indore** is an authentic record of training work.

Chhaya Gupta(1710DMTCECC01342)
Harsh Agrawal (1710DMTCECF01327)
Saloni Patil (1701DMTCSE01308)
Vanshika Sawle (1710DMTCSE01315)

ACKNOWLEDGEMENT

It gives us great pleasure to acknowledge the guidance, assistance and support of Ms. Mousita Dhar in making the Project and this Project report successful, which has been structured under her valued suggestion.

She has helped us to accomplish the challenging task in a very short period of time.

Finally, we express the constant support of our friends, family and professors for inspiring us throughout and encouraging us.

Chhaya Gupta(1710DMTCECC01342)
Harsh Agrawal (1710DMTCECF01327)
Saloni Patil (1701DMTCSE01308)
Vanshika Sawle (1710DMTCSE01315)

CERTIFICATE OF APPROVAL

The project **"Absenteeism at Work"** made by the efforts of **Chhaya gupta, Harsh Agrawal, Saloni Patil** and **Vanshika Sawle** of **Shri Vaihsnav Institute of Information Technology, Indore** is hereby approved as a creditable study for the **Bachelor of Technology** in **Computer Science and Engineering** and presented in a manner of satisfactory to warrant its acceptance as a prerequisite to the degree for which it has been submitted.

It is understood that by this approval the undersigned this project only for the purpose for which it is submitted.

Ms. Mousita Dhar
Project Incharge

CONTENTS

.

ABSTRACT

Absenteeism is the failure of employees to report for work when they are scheduled to work. Employees who are away from work on recognized holidays, vacations, approved leaves of absence, or approved leaves of absence would not be included. Absenteeism is becoming a serious practice in labour oriented industries especially in those large industries where labours are working in mass. It is a matter of prime concern for the supervisors and managers. They have to find the ways to overcome absenteeism. Absenteeism is a serious workplace problem and an expensive occurrence for both employers and employees seemingly unpredictable in nature. A satisfactory level of attendance by employees at work is necessary to allow the achievement from work and thus the world suffers. Absenteeism of employees on work leads to back logs, piling of work and thus work delay.

The Objective of this project is to suggest steps and measures for reducing the rate of Absenteeism. By applying different Machine Learning algorithms on the data set, for the same, and predicting absenteeism with reasons with an accuracy of 92%. This helps the organization to increase employee's commitment towards work and thereby reduce absenteeism.

INTRODUCTION

Data Science

Data science is an inter-disciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from many structural and unstructured data. Data science is related to data mining, machine learning and big data.

Data science is a "concept to unify statistics, data analysis, machine learning, domain knowledge and their related methods" in order to "understand and analyze actual phenomena" with data.[3] It uses techniques and theories drawn from many fields within the context of mathematics, statistics, computer science, domain knowledge and information science. Turing award winner Jim Gray imagined data science as a "fourth paradigm" of science (empirical, theoretical, computational and now data-driven) and asserted that "everything about science is changing because of the impact of information technology" and the data deluge.

Machine Learning

Machine learning (ML) is the study of computer algorithms that improve automatically through experience. It is seen as a subset of artificial intelligence. Machine learning algorithms build a mathematical model based on sample data, known as "training data", in order to make predictions or decisions without being explicitly programmed to do so. Machine learning algorithms are used in a wide variety of applications, such as email filtering and computer vision, where it is difficult or infeasible to develop conventional algorithms to perform the needed tasks.

Machine learning is closely related to computational statistics, which focuses on making predictions using computers. The study of mathematical optimization delivers methods, theory and application domains to the field of machine learning. Data mining is a related field of study, focusing on exploratory data analysis through unsupervised learning. In its application across business problems, machine learning is also referred to as predictive analytics.

Python

Python is a high-level, general-purpose, open source, strictly typed programming language. The language provides constructs intended to enable clear programs on both a small and large scale. Python was created By Guido van Rossum. The Python Software Foundation (PSF) is the organization behind Python.

Python versions:

- First released in 1991.
- Python 2.0 was released on 16 October 2000
- Python 3.0 was released on 3 December 2008

Current Versions:

- Python 3.6.3
- Python 2.7.14

Python features:

- Easy to understand
- Dynamic
- Object oriented
- Multipurpose
- Strongly typed
- Open Sourced

Python is mainly used in many domains:

- Web Development
- Data Analysis
- Machine Learning
- Internet Of Things
- GUI Development
- Image processing
- Data visualization
- Game Development

Integrated Development Environment

Anaconda

Anaconda is a free and open-source distribution of the Python and R programming languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analytics, etc.), that aims to simplify package management and deployment. The distribution includes data-science packages suitable for Windows, Linux, and macOS. It is developed and maintained by Anaconda, Inc., which was founded by Peter Wang and Travis Oliphant in 2012. As an Anaconda, Inc. product, it is also known as Anaconda Distribution or Anaconda Individual Edition, while other products from the company are Anaconda Team Edition and Anaconda Enterprise Edition, which are both not free.

Package versions in Anaconda are managed by the package management system *conda*. This package manager was spun out as a separate open-source package as it ended up being useful on its own and for other things than Python. There is also a small, bootstrap version of Anaconda called Miniconda, which includes only conda, Python, the packages they depend on, and a small number of other packages.

Jupyter Notebook

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.

PROBLEM STATEMENT

As mentioned in Kaggle, the description of problem is:

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?

DATA DESCRIPTION

There are total 21 features in the dataset in which 20 are independent and 1 is dependent (Absenteeism time in hours). Since our target is continuous, but we will convert it into binary variable and apply the classification models.

1. Features Information:
2. Individual identification
3. Reason for absence
    a) Certain infectious and parasitic diseases
    b) Neoplasms
    c) Diseases of the blood and blood-forming organs and certain disorders involving the immune mechanism
    d) Endocrine, nutritional and metabolic diseases
    e) Mental and behavioural disorders
    f) Diseases of the nervous system
    g) Diseases of the eye and adnexa
    h) Diseases of the ear and mastoid process
    i) Diseases of the circulatory system
    j) Diseases of the respiratory system
    k) Diseases of the digestive system
    l) Diseases of the skin and subcutaneous tissue
    m) Diseases of the musculoskeletal system and connective tissue
    n) Diseases of the genitourinary system
    o) Pregnancy, childbirth and the puerperium
    p) Certain conditions originating in the perinatal period
    q) Congenital malformations, deformations and chromosomal abnormalities
    r) Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified
    s) Injury, poisoning and certain other consequences of external causes
    t) External causes of morbidity and mortality
    u) Factors influencing health status and contact with health services.
4. Month of absence
5. Day of week (Monday(2),Tueseday(3),Wednesday(4),Thursday(5),Friday(6))
6. Seasons (Summer(1), autumn(2), winter(3), spring(4))
7. Transportation expense
8. Distance from residence to work (km)
9. Service time
10. Age
11. Work load Average/day
12. Hit target
13. Disciplinary failure
14. Education
15. Son(num of children)
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)**

METHODOLOGY

Packages Used

1. **NumPy**
   NumPy is the fundamental package for scientific computing with Python.
   - It contains among other things:
   - a powerful N-dimensional array object
   - sophisticated (broadcasting) functions
   - tools for integrating C/C++ and Fortran code
   - useful linear algebra, Fourier transform, and random number capabilities

   Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

2. **Pandas**
   Pandas is an open source, BSD-licensed library providing highperformance, easy-to-use data structures and data analysis tools for the Python programming language.
   Pandas library is well suited for data manipulation and analysis using python. In particular, it offers data structures and operations for manipulating numerical tables and time series.

3. **Matplotlib**
   Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and IPython shell, the jupyter notebook, web application servers, and four graphical user interface toolkits.

   Matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, error charts, scatterplots, etc., with just a few lines of code. For simple plotting the pyplot module provides a MATLAB-like interface, particularly when combined with IPython. For the power user, you have full control of line styles, font properties, axes properties, etc, via an object oriented interface or via a set of functions familiar to MATLAB users.

4. **Scikit-learn**
   Scikit-learn provides machine learning libraries for python.Some of the features of Scikit-learn includes: • Simple and efficient tools for data mining and data analysis • Accessible to everybody, and reusable in various contexts • Built on NumPy, SciPy, and matplotlib • Open source, commercially usable - BSD license.

5. **Seaborn**
   Seaborn is a Python visualization library based on matplotlib. It provides a high-level interface for drawing attractive statistical graphics.

Collecting data

Kaggle is a platform for predictive modeling and analytics competitions in which statisticians and data miners compete to produce the best models for predicting and describing the datasets uploaded by companies and users.
On 8 March 2017, Google announced that they were acquiring Kaggle. They will join the

Google Cloud team and continue to be a distinct brand. In January 2018, Booz Allen and Kaggle launched Data Science Bowl, a machine learning competition to analyze cell images and identify nuclei.
We downloaded the dataset from Kaggle from link https://www.kaggle.com/tonypriyanka2913/employeeabsenteeism?select=Absenteeism_at_work_Project.xls and the dataset is named as **Employee Absenteeism.**

Importing libraries

We have imported the following libraries:

1. import numpy as np
2. import pandas as pd
3. import matplotlib.pyplot as plt
4. import seaborn as sns
5. from sklearn.impute import SimpleImputer
6. from sklearn.preprocessing import StandardScaler
7. from sklearn.base import BaseEstimator, TransformerMixin
8. from sklearn.linear_model import LogisticRegression
9. from sklearn.model_selection import train_test_split
10. from sklearn.metrics import classification_report, accuracy_score, confusion_matrix
11. from sklearn import svm, tree
12. from sklearn.ensemble import RandomForestClassifier
13. from sklearn.neighbors import KNeighborsClassifier

Reading the dataset

	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	...	Disciplinary failure	Education	Son	Social drinker	Social smoker	Pet	We
0	11	26.0	7.0	3	1	289.0	36.0	13.0	33.0	239554.0	...	0.0	1.0	2.0	1.0	0.0	1.0	
1	36	0.0	7.0	3	1	118.0	13.0	18.0	50.0	239554.0	...	1.0	1.0	1.0	1.0	0.0	0.0	
2	3	23.0	7.0	4	1	179.0	51.0	18.0	38.0	239554.0	...	0.0	1.0	0.0	1.0	0.0	0.0	
3	7	7.0	7.0	5	1	279.0	5.0	14.0	39.0	239554.0	...	0.0	1.0	2.0	1.0	1.0	0.0	
4	11	23.0	7.0	5	1	289.0	36.0	13.0	33.0	239554.0	...	0.0	1.0	2.0	1.0	0.0	1.0	

5 rows × 21 columns

Data Pre-Processing

Any predictive modelling 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 look at the all the probability distributed. We can visualize that in a glance by looking at the probability distributions or probability density functions of the variable.

Counting Missing Values

```
ID                                0
Reason for absence                3
Month of absence                  1
Day of the week                   0
Seasons                           0
Transportation expense            7
Distance from Residence to Work   3
Service time                      3
Age                               3
Work load Average/day            10
Hit target                        6
Disciplinary failure              6
Education                        10
Son                               6
Social drinker                    3
Social smoker                     4
Pet                               2
Weight                            1
Height                           14
Body mass index                  31
Absenteeism time in hours        22
dtype: int64
```

Checking Information on Dataset

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 734 entries, 0 to 739
Data columns (total 21 columns):
Month of absence                  734 non-null float64
Day of the week                   734 non-null float64
Seasons                           734 non-null float64
Transportation expense            734 non-null float64
Distance from Residence to Work   734 non-null float64
Service time                      734 non-null float64
Age                               734 non-null float64
Work load Average/day             734 non-null float64
Hit target                        734 non-null float64
Disciplinary failure              734 non-null float64
Education                         734 non-null float64
Son                               734 non-null float64
Social drinker                    734 non-null float64
Social smoker                     734 non-null float64
Pet                               734 non-null float64
Body mass index                   734 non-null float64
Absenteeism time in hours         734 non-null float64
R_Known                           734 non-null float64
R_NotSerious                      734 non-null float64
R_Pois_unclass                    734 non-null float64
R_Preg_Birth                      734 non-null float64
dtypes: float64(21)
memory usage: 126.2 KB
```

Filling Missing Values

In this data missing value occur when no data value stored for the variable in an observation. Missing data are a common occurrence and can have significant effect on the conclusions that can be drawn from the data. If a column has more than 30% of data as missing value either we ignore the entire column or we ignore those observations.

In our project we will import Simple Imputer class to fill the missing values with the most frequent values present in that column.

```
ID                                0
Reason for absence                0
Month of absence                  0
Day of the week                   0
Seasons                           0
Transportation expense            0
Distance from Residence to Work   0
Service time                      0
Age                               0
Work load Average/day             0
Hit target                        0
Disciplinary failure              0
Education                         0
Son                               0
Social drinker                    0
Social smoker                     0
Pet                               0
Weight                            0
Height                            0
Body mass index                   0
Absenteeism time in hours         0
dtype: int64
```

MISSING VALUES FILLED

Checking for any non-integer value

There is no non-integer value in our project's dataset hence, we do not need to encode them.

Standardizing the data

The Standard Scaler is used to standardize features by removing the mean and scaling to unit variance.
Centering and scaling happen independently on each feature by computing the relevant statistics on the samples in the training set. Mean and standard deviation are then stored to be used on later data using transform.

Standardization of a dataset is a common requirement for many machine learning estimators: they might behave badly if the individual features do not more or less look like standard normally distributed data (e.g. Gaussian with 0 mean and unit variance).

But the problem with us was that the Standard Scaler is standardizing the dummy variables also because of that our model is giving a bit less accuracy.
That's why we chose to use a Custom Scaler in which we can standardize only those features that we want to, means we can omit the dummy variables of our dataset to increase the accuracy of the model.

We defined a Custom Scaler class in that we have used BaseEstimator and Transformermixin from sklearn package and the traditional Standard Scaler. This class scales only those features which we want to.

EXPLORATORY DATA ANALYSIS

1. Correlation

Visualizing through HeatMap:

Now on finding the correlation of all columns with the target column we will get:

```
#to find the correaltion of absenteeism time in hours
c=data.corr()
cm=c['Absenteeism time in hours']
cm
```

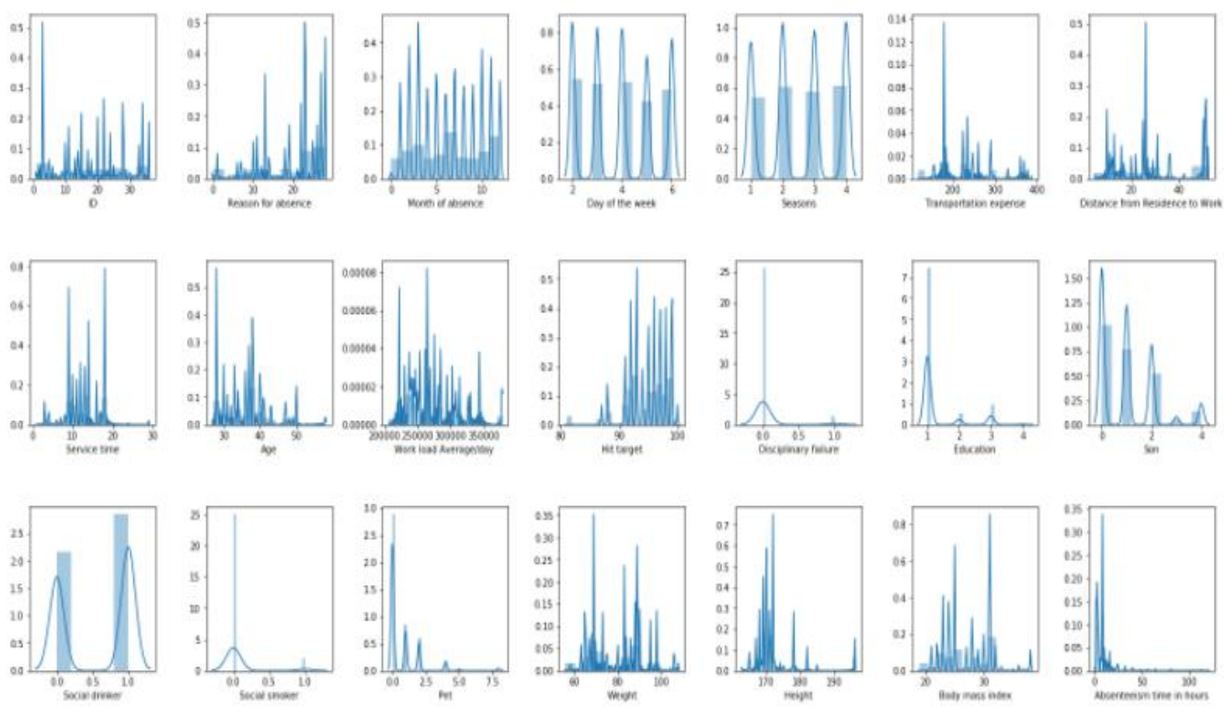
```
ID                               -0.020412
Reason for absence               -0.200085
Month of absence                  0.027415
Day of the week                  -0.114389
Seasons                          -0.000483
Transportation expense            0.045881
Distance from Residence to Work  -0.100133
Service time                      0.013788
Age                               0.076658
Work load Average/day             0.022790
Hit target                        0.022126
Disciplinary failure             -0.048010
Education                        -0.046744
Son                               0.117450
Social drinker                    0.065527
Social smoker                     0.042515
Pet                              -0.028020
Weight                           -0.009661
Height                            0.093374
Body mass index                  -0.056243
Absenteeism time in hours         1.000000
Name: Absenteeism time in hours, dtype: float64
```

2. Distplot for Analysing features

Seaborn distplot lets you show a histogram with a line on it. This can be shown in all kinds of variations. We use seaborn in combination with matplotlib, the Python plotting module.

A distplot plots a univariate distribution of observations. The distplot() function combines the matplotlib hist function with the seaborn kdeplot() and rugplot() functions.

Here since we want to see the distplot of 21 columns we will plot that in a loop.

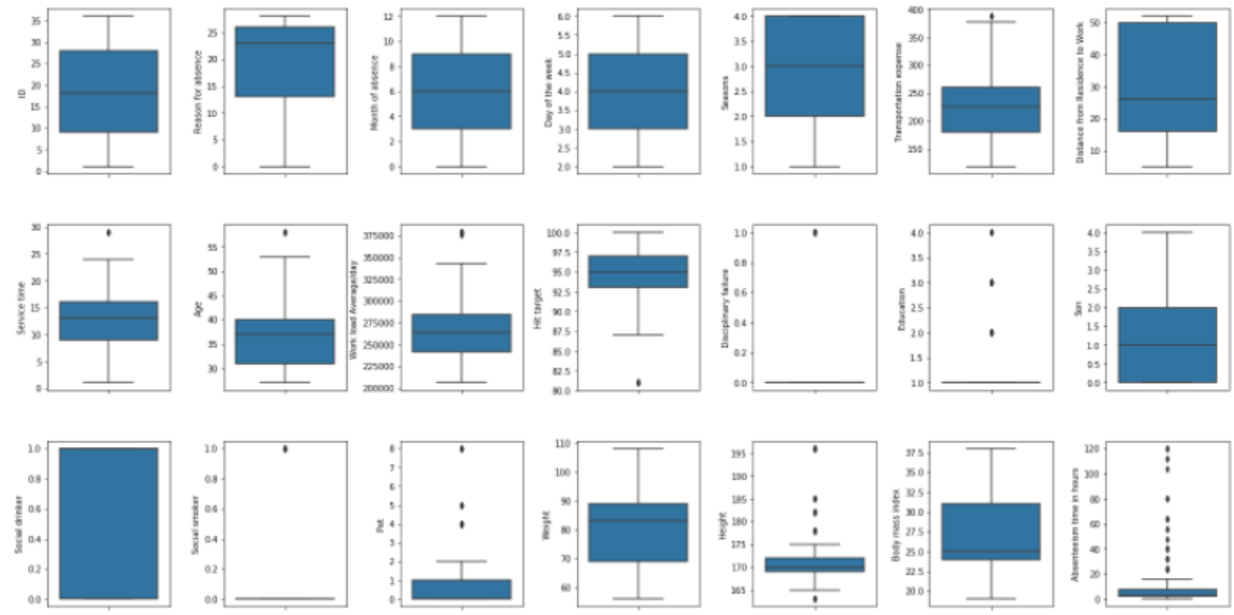


Here 11,14,15 i.e Discipline,Social drinker,Social Smoker are discrete values(0/1).

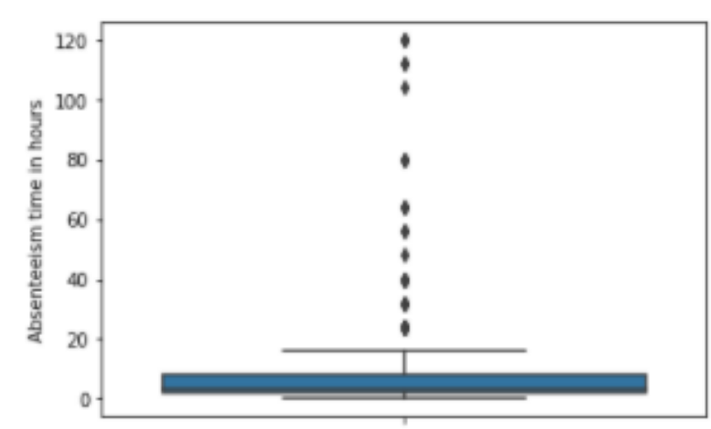
3. Boxplot for Analysing features

A box plot (or box-and-whisker plot) shows the distribution of quantitative data in a way that facilitates comparisons between variables or across levels of a categorical variable. The box shows the quartiles of the dataset while the whiskers extend to show the rest of the distribution, except for points that are determined to be “outliers” using a method that is a function of the inter-quartile range.

Here since we want to see the boxplot of 21 columns we will plot that in a loop.

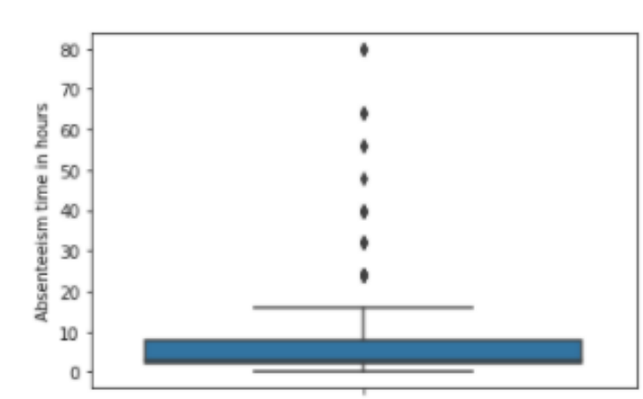


Here if we notice the plot of our target column i.e. Absenteeism time in hours



To remove the outliers we will delete that rows whose values are greater than 100.

And then we will see its boxplot.



6 rows are deleted in handling outliers for target.

Since only 6 rows are deleted it is not that big loss of data.

4. Grouping

Now we will evaluate values of features and will try to group them and then find their relation with absenteeism time in hours.

i. Grouping values according to the season

:

	Seasons	Absenteeism time in hours
0	1.0	6.041667
1	2.0	5.445026
2	3.0	7.093923
3	4.0	5.974227

We can notice here that absenteeism time is largest in **season 3.0.**

ii. Grouping values according to the day of the week

```
data[['Day of the week','Absenteeism time in hours']].groupby(['Day of the week'],as_index=False).mean()
```

	Day of the week	Absenteeism time in hours
0	2.0	8.006250
1	3.0	5.880000
2	4.0	6.522581
3	5.0	4.552000
4	6.0	5.243056

We can notice here that absenteeism time is largest on **day 2.**

iii. Grouping values according to the age
Since in age column there are varieties of values that can't be grouped hence we will form the intervals or say bands for age and will then show its relation with target column.

```
data['AgeBand'] = pd.cut(data['Age'],5) ## Need to sav
```

```
data[['AgeBand','Absenteeism time in hours']].groupby(
```

	AgeBand	Absenteeism time in hours
0	(26.969, 33.2]	6.003802
1	(33.2, 39.4]	6.148014
2	(39.4, 45.6]	7.307018
3	(45.6, 51.8]	4.835616
4	(51.8, 58]	4.285714

We can notice here that absenteeism time is largest in **age band of (39.4,45.6).**

iv. Grouping values according to the month

	Month of absence	Absenteeism time in hours
0	0.0	0.000000
1	1.0	4.440000
2	2.0	4.083333
3	3.0	6.511628
4	4.0	6.961538
5	5.0	6.375000
6	6.0	7.611111
7	7.0	7.815385
8	8.0	5.333333
9	9.0	5.811321
10	10.0	5.314286
11	11.0	6.048387
12	12.0	7.959184

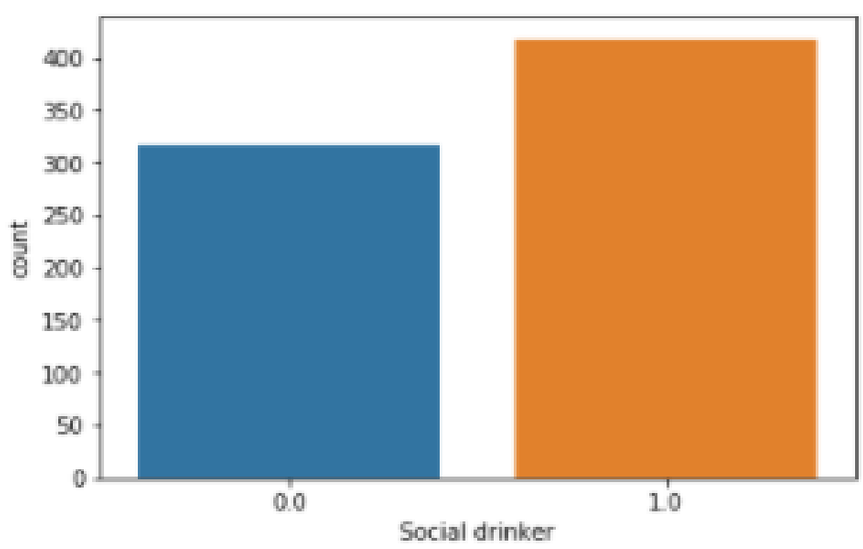
We can notice here that absenteeism time is largest in **month 12.**

v. Grouping values according to the social drinker

	Social drinker	Absenteeism time in hours
0	0.0	4.977848
1	1.0	6.997608

We can notice here that absenteeism time is more for those **who are social drinkers.**

So we should visualize that how many of our employees are social drinkers and we'll do it by using count plot.



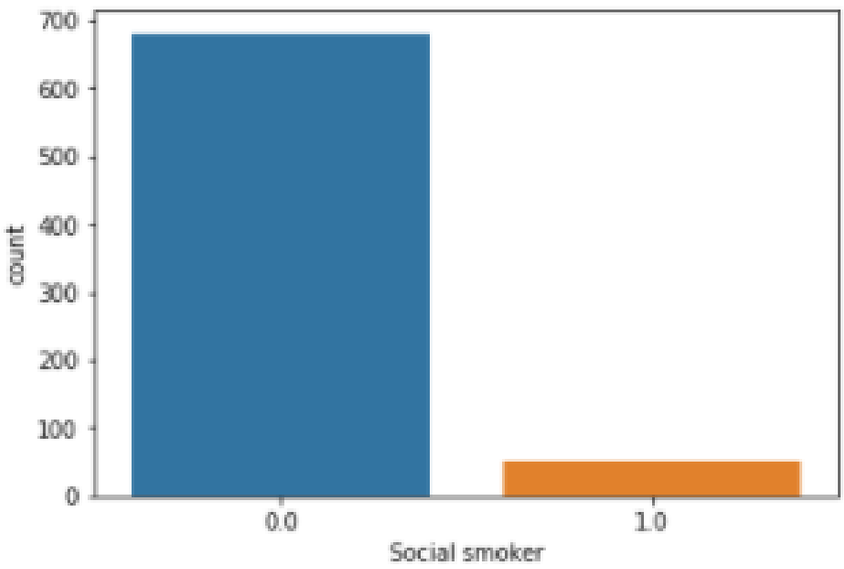
This clearly shows that we have larger number of social drinkers in our dataset.

vi. Grouping values according to the social smoker

	Social smoker	Absenteeism time in hours
0	0.0	6.066079
1	1.0	6.924528

We can notice here that absenteeism time is more(although not a very big difference) for those **who are social smokers .**

So we should visualize that how many of our employees are social drinkers and we wil do it by using count plot.



This clearly shows that most of the people mentioned in our dataset do not smoke.

vii. Grouping values according to the son (number of son)

	Son	Absenteeism time in hours
0	0.0	5.438538
1	1.0	5.243363
2	2.0	7.794702
3	3.0	12.200000
4	4.0	7.707317

We can notice here that absenteeism time is more for those **who have 3 sons.**

viii. Grouping values according to the pet (number of pets)

ix.

	Pet	Absenteeism time in hours
0	0.0	6.424837
1	1.0	6.761194
2	2.0	3.684211
3	4.0	7.312500
4	5.0	4.166667
5	8.0	4.250000

We can notice here that absenteeism time is more for those **who have 4 pets.**

5. Now we will work on Reason for absence column

Here we have 28 unique reasons for a person to be absent. We will divide them in 4 sections as per their reason number.

The 28 reasons are:

1. Certain infectious and parasitic diseases
2. Neoplasms
3. Diseases of the blood and blood-forming organs and certain disorders involving the immune mechanism
4. Endocrine, nutritional and metabolic diseases
5. Mental and behavioural disorders
6. Diseases of the nervous system
7. Diseases of the eye and adnexa
8. Diseases of the ear and mastoid process
9. Diseases of the circulatory system
10. Diseases of the respiratory system
11. Diseases of the digestive system
12. Diseases of the skin and subcutaneous tissue
13. Diseases of the musculoskeletal system and connective tissue
14. Diseases of the genitourinary system
15. Pregnancy, childbirth and the puerperium
16. Certain conditions originating in the perinatal period
17. Congenital malformations, deformations and chromosomal abnormalities
18. Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified
19. Injury, poisoning and certain other consequences of external causes
20. External causes of morbidity and mortality
21. Factors influencing health status and contact with health services.

And 7 categories without (CID)

(22) Patient follow-up,
(23) medical consultation
(24) blood donation
(25) laboratory examination
(26) unjustified absence
(27) physiotherapy
(28) dental consultation

We will group all these reasons in

1- 14: are various diseases
15 -17: pregnancy and given birth related
18-21: poisons or diseases not else were categorise
22 -28: light reason or less serious reasons

We'll add new columns for each of the 4 reasons (dummy variables) and store that in new dataset called new_data and save the changes in a new dataset called **Absenteeism_preprocessed.xls.**

6. Now we'll work on Absenteeism time in hour column

Since the values in our target column have too many unique values while this target should be categorical and that is why we will divide these values and assign 0 or 1 values to it.

We have decided to use median rather than choosing an arbitrary cut off, which might make the data unbalanced.

The reason for this is because we have very few rows of data.

When we will find the median of this column we will get the value 18.

Like this we will convert the absenteeism time in hours column in categorical values.

And now we can apply various models on our dataset.

DATA PREDICTION

We will use 5 different models in our project for the prediction

1. Support Vector Classification (SVC)
2. Random Forest Classifier
3. Logistic Regression
4. K- Neighbours Classifier
5. Gaussian Naive Bayes

1. ***Support Vector Classification (SVC)***

   SVC is one of the standard set of supervised machine learning model employed in classification. Given a two-class training sample the aim of a support vector machine i s to find the best highest-margin separating hyper plane between the two classes. For better generalization hyper plane should not lies closer to the data points belong to the other class. Hyperplane should be selected which is far from the data points from each category. The points that lie nearest to the margin of the classifier are the support vectors.

   The SVM finds the optimal separating hyperplane by maximizing the distance between the two decision boundaries. It is not biased by outliers, as well as not sensitive to over fitting but it is not appropriate for non linear problems and is also not the best choice for large number of features as it is quite complex.

   After training the SVC model on training set of our dataset, we will use the model for predicting the outcome of inputs of test set.

   Accuracy of train set SVC is 0.955706984668
   Accuracy of test set SVC is 0.925170068027

   Then we will compare the predicted outcome with the actual outcome mentioned in the test set using confusion matrix.

n=147	Predicted: NO	Predicted: YES
Actual: NO	136	0
Actual: YES	11	0

Confusion Matrix of SVC

2. *Random Forest Classifier*

In Random Forest Classification, we use the concept of ENSEMBLE learning which means taking multiple machine learning algorithms and put them together to form a final one, hence the final one leveraging different machine learning algorithms. Random forest runs decision tree multiple times.

For making random forest model first it need to pick at random K data points from the Training set and then build the decision tree associated to those K data points. Then choose the number Ntree of tress we want to build and then again repeat the above steps.

For a new data point, each one of the Ntree trees predict the category to which the data points belongs, and assign the new data point to the category that wins the majority vote.

After training the Random Forest Classifier model on training set of our dataset, we will use the model for predicting the outcome of inputs of test set.

Accuracy of train set Random Forest Classifier is 0.979557069847
Accuracy of test set Random Forest Classifier is 0.91156462585

Then we will compare the predicted outcome with the actual outcome mentioned in the test set using confusion matrix.

n=147	Predicted: NO	Predicted: YES
Actual: NO	134	2
Actual: YES	11	0

Confusion Matrix of Random Forest Classifier

3. *Logistic Regression*

Logistic regression is a statistical technique used to predict probability of binary response based on one or more independent variables. It uses logistic or sigmoid function for prediction.

The logistic function or the sigmoid function was developed by statisticians so that they can describe the properties of population growth in ecology, rising quickly and maxing out at the carrying capacity of the environment.

It's an S-shaped curve that can take any real-valued number and map it into a value between 0 and 1, but never exactly at those limits.

Sigmoid Function: $p = 1/ (1+e^{-y})$ Or $y=\ln(p/1-p)$

Hence, we can conclude

$$\ln(p/1-p)=b0 + b1*x + b2*x +\ldots\ldots$$

Where y is the predicted output, b0 is the bias or intercept term and b1 is the coefficient for the input value (x) and ….

After training the Logistic Regression model on training set of our dataset, we will use the model for predicting the outcome of inputs of test set.

Accuracy of train set Logistic Regression is 0.955706984668
Accuracy of test set Logistic Regression is 0.925170068027

Then we will compare the predicted outcome with the actual outcome mentioned in the test set using confusion matrix.

n=147	Predicted: NO	Predicted: YES
Actual: NO	136	0
Actual: YES	11	0

Confusion Matrix of Logistic regression

4. *K- Nearest Neighbours*

The K-Nearest Neighbours or K-NN is a non-parametric method i.e. KNN makes no assumptions about the functional form of the problem being solved can be used. It can be used for both regression and classification.

For building an knn model we first choose any k no. of neighbours , then we determine the k-nearest neighbours of the new data point (using Euclidean distance , Hamming Distance , Manhattan Distance or Minkowski Distance) .Among those k neighbours , it counts the number of data points in each category. Finally, it assigns the new data point to the category where it counted the most neighbours.

K-NN is also called as instance-learning, as raw training instances are used to make predictions. Similarly, it is called as lazy learning because no learning of the model is required and all of the work happens at the time a prediction is requested.

Class probabilities can be calculated as the normalized frequency of samples that belong to each class in the set of K most similar instances for a new data instance. For example, in a binary classification problem (class is 0 or 1):

p (class=0) = count (class=0) / (count(class=0)+count(class=1))

After training the K- NN model on training set of our dataset, we will use the model for predicting the outcome of inputs of test set.

Accuracy of train set K- NN is 0.955706984668
Accuracy of test set K- NN is 0.925170068027

Then we will compare the predicted outcome with the actual outcome mentioned in the test set using confusion matrix.

n=147	Predicted: NO	Predicted: YES
Actual: NO	136	0
Actual: YES	11	0

Confusion Matrix of KNN

5. *Gaussian Naive Bayes*

Gaussian Naive Bayes is a classification technique with a notion which defines all features are independent and unrelated to each other. It defines that status of a specific feature in a class does not affect the status of another feature. Since it is based on conditional probability it is considered as a powerful algorithm employed for classification purpose.

It works well for the data with misbalancing problems and missing values. Naive Bayes is a machine learning classifier which employs the Bayes Theorem. Using Bayes theorem posterior probability P (C|X) can be calculated from P(C), P(X) and P(X|C). Therefore,

$$P(C|X) = (P(X|C)\,P(C))/P(X)$$

Where,

P (C|X) = Target class's posterior probability.
P (X|C) = Predictor class's probability.
P (C) = Class C's probability being true.
P (X) = Predictor's prior probability.

After training the Gaussian NB model on training set of our dataset, we will use the model for predicting the outcome of inputs of test set.

Accuracy of train set Gaussian NB is 0.67461669506
Accuracy of test set Gaussian NB is 0.619047619048

Then we will compare the predicted outcome with the actual outcome mentioned in the test set using confusion matrix.

n=147	Predicted: NO	Predicted: YES
Actual: NO	81	55
Actual: YES	1	10

Confusion Matrix of Gaussian Naive Bayes

RESULT

By evaluating performances of the five classification algorithms that we have applied on the dataset Logistic Regression is giving maximum accuracy score in training and testing both that are **0.955706984668(95 %)** and **0.925170068027 (92%)** respectively.

On analysing the attributes we have found that only 16 attributes are significant which we used in our model and they are as follows:

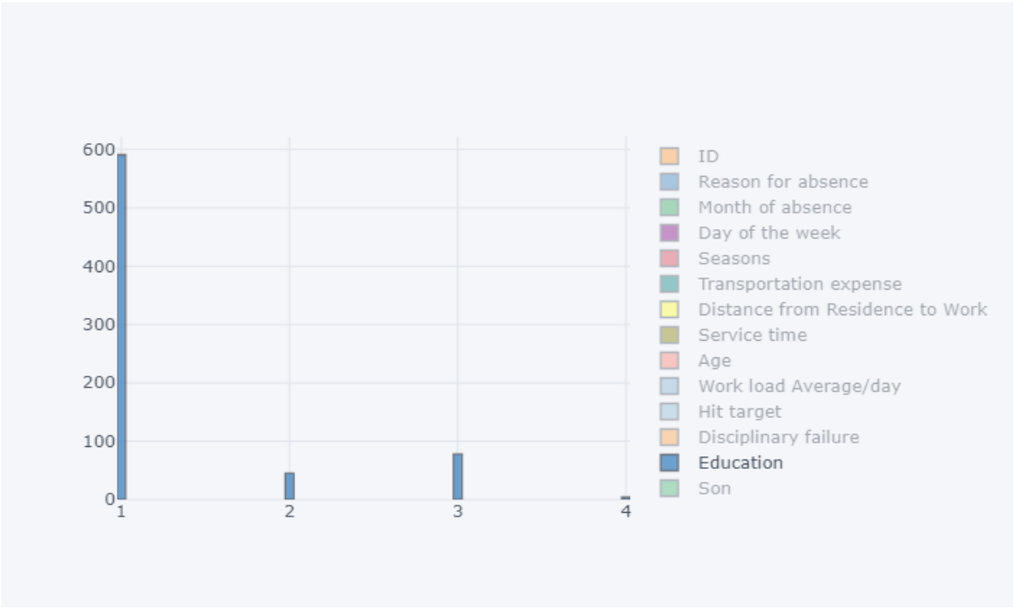
1. Reason_1
2. Reason_2
3. Reason_3
4. Reason_4
5. Month of absence
6. Seasons
7. Transportation expense
8. Distance from Residence to Work
9. Age
10. Work load Average/day
11. Hit target
12. Education
13. Social drinker
14. Social smoker
15. Pet
16. Body mass index

Classification Report of Logistic Regression

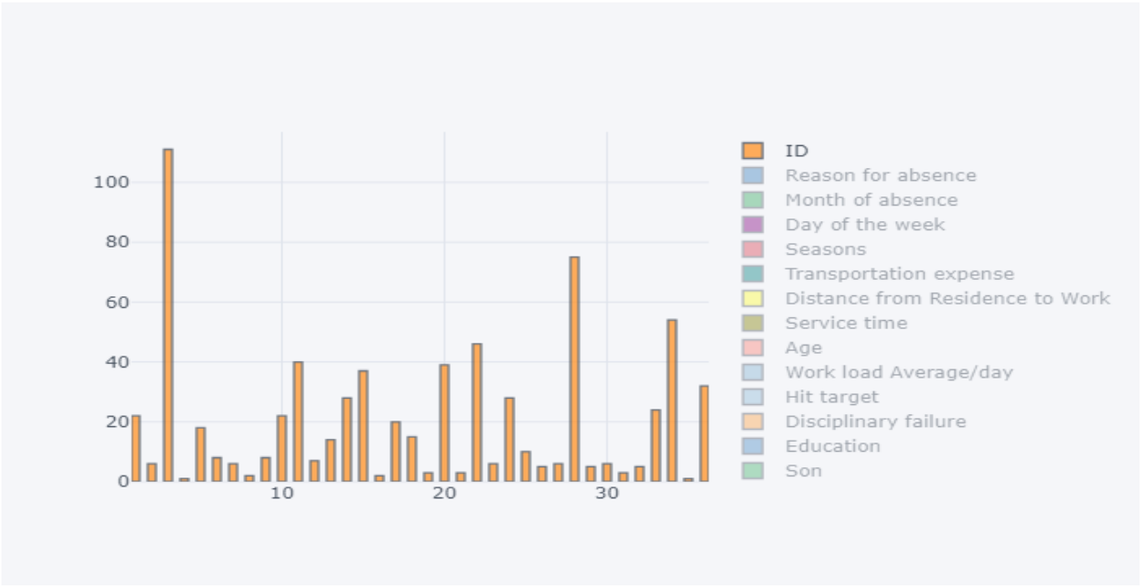
	precision	recall	f1-score	support
0	0.93	1.00	0.96	136
1	0.00	0.00	0.00	11
micro avg	0.93	0.93	0.93	147
macro avg	0.46	0.50	0.48	147
weighted avg	0.86	0.93	0.89	147

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

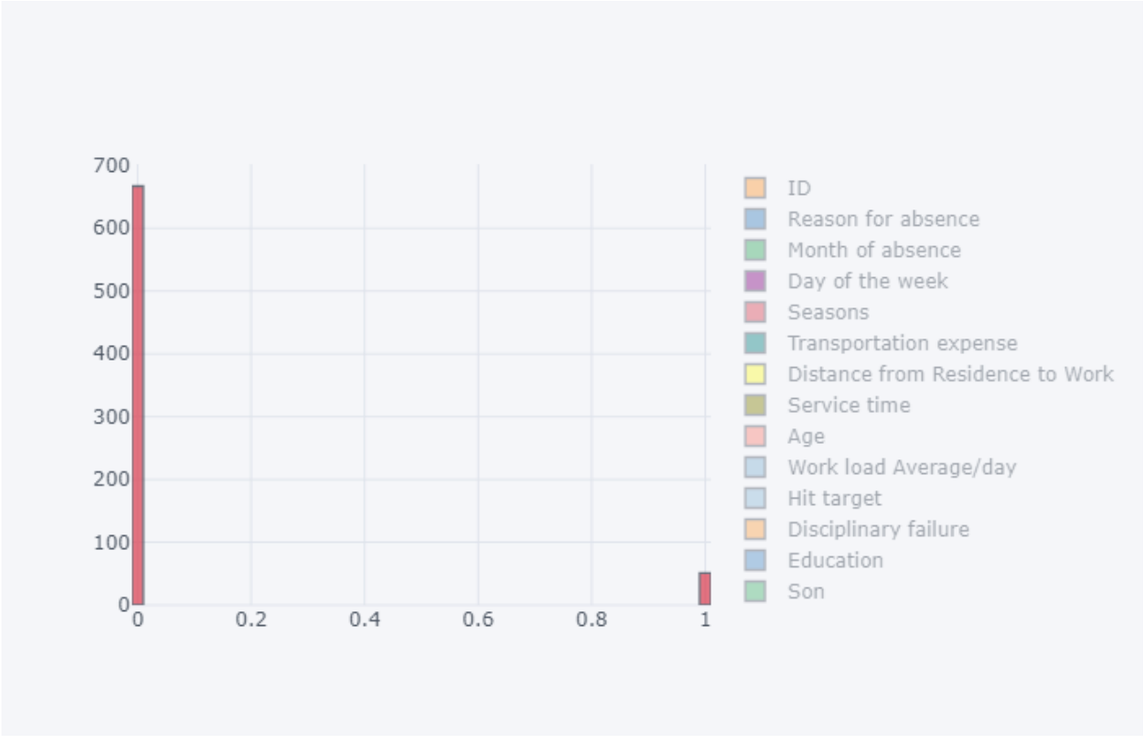
1. It is observed that employee with low education have maximum absentee time.



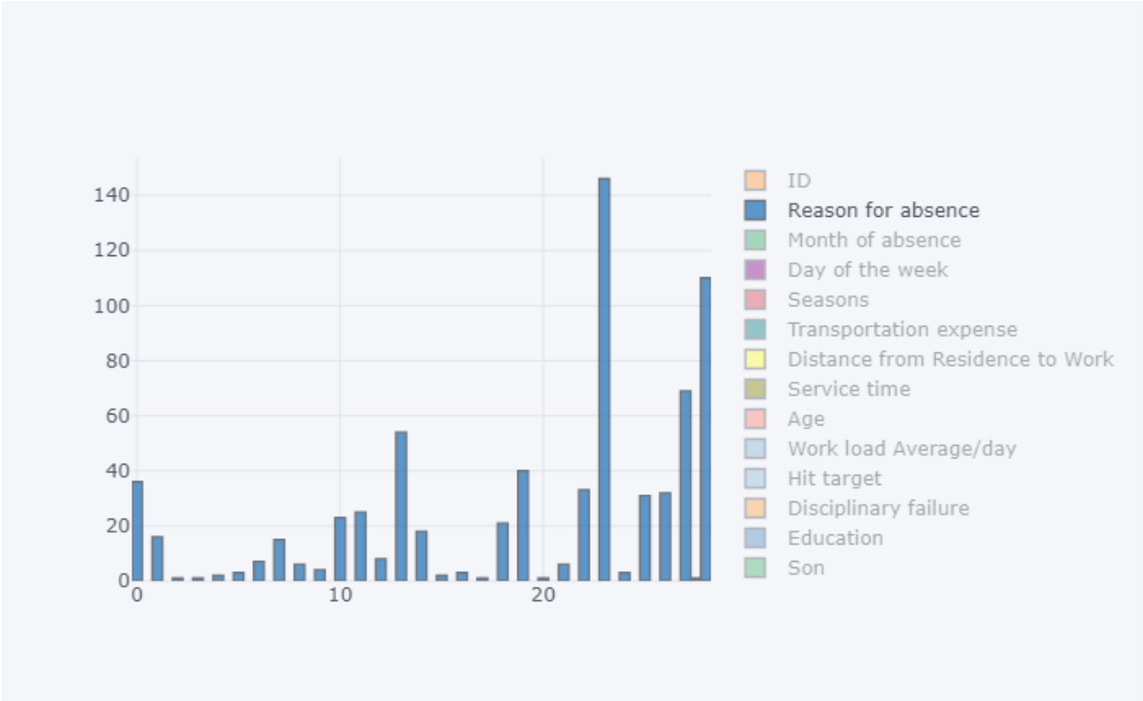
2. Some employee with **ID 3, 28, 34** are often absent from work, company should take action against them.



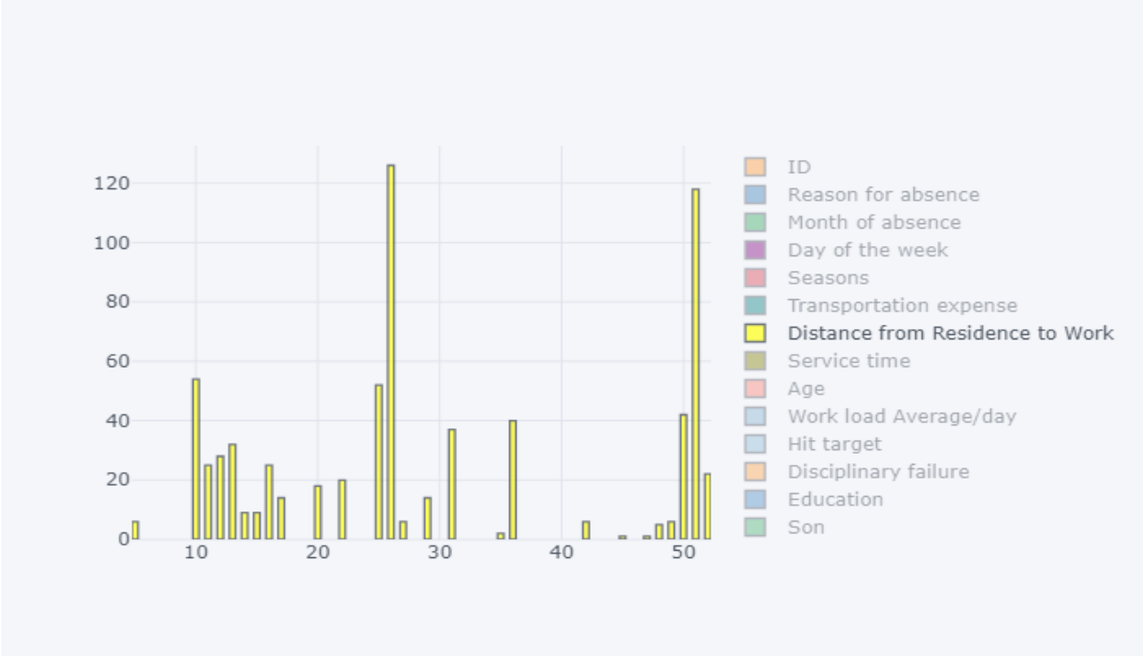
3. Employees who are social smoker have more absentee hour than who are not social smoker.



4. Most often Reason for absence are medical consultation and dental consultation, company should take care of it.



5. Employees who have Distance from Residence to Work high more tends to absent more.



CONCLUSION

Analyzing and predicting absenteeism according to given problem statement is done by us and systematic efforts are made in designing a system which results in the prediction of absenteeism of employee.

During this work, various machine learning classification algorithms are studied and evaluated on various measures. Experiments are performed on a Employee Absenteeism dataset with 740 observations and 21 attributes from which 16 came as significant attributes.

Experimental results determine the adequacy of the designed system with an achieved accuracy of 92% (approx) using the Logistic Regression classification algorithm.

In future the work can be extended and improved for the automation of absenteeism analysis for employees, including some other machine learning algorithms.

REFRENCES

1. https://en.wikipedia.org/wiki/Data_science
2. https://en.wikipedia.org/wiki/Machine_learning
3. https://en.wikipedia.org/wiki/Anaconda_(Python_distribution)
4. https://jupyter.org/
5. https://www.kaggle.com/tonypriyanka2913/employee-absenteeism?select=Absenteeism_at_work_Project.xls
6. https://numpy.org/
7. https://matplotlib.org/
8. https://scikit-learn.org/
9. https://pandas.pydata.org/
10. https://seaborn.pydata.org/