CE-802-7-SP-Machine Learning

Final Report

Submitted by,

Karan Bhatt

(2112102)

Submitted to,

Dr. Vito de feo

TABLE OF CONTENT

Introduction:

Our body produces multiple chemicals inside it. When it stops producing any some or the other disease happens with the body. Same as if our body or pancreas stops producing the enough insulin that human body requires or when the body can not use produced insulin effectively then chronic disease named diabetes occurs, according to WHO [1]. Insulin is a chemical / hormone that take care of blood glucose level in human bodies. High blood sugar or Hyperglycaemia is a one of the side effects of the diabetes and many body organs and system can be affected by that. Since 2000 cases of diabetes are increasing very fast.

Patient analysis to determine if they have diabetes

Step 1: Data importing and loading

The given data set is consisted of 16 independent variables in total, which are F1, F2, F3, F4, F5, F6, F7, F8, F9, F10, F11, F12, F13, F14 and F15 and all of these variables are under one label. The target variable is either True or False, which means if the person is suffering from diabetes or not.

```
In [2]:  1  df_diabetes = pd.read_csv("CE802_P2_Data.csv")
```

```
In [3]:  1  df_diabetes.head()
```

Out[3]:

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15	Class
0	11.7	4.02	-4.34	9.90	29.79	89.58	0.63	23	10.35	158.56	-7.88	0.03	1	1.96	NaN	False
1	11.7	4.20	-3.68	10.98	17.46	179.58	0.05	11	8.30	110.56	-3.10	0.84	1	1.50	NaN	False
2	37.7	25.80	3.60	0.48	12.24	407.58	-0.29	230	4.06	254.56	6.68	21.60	10	7.63	NaN	True
3	7.7	5.40	0.30	9.42	19.86	119.58	0.29	12	7.61	66.56	-1.84	1.05	1	2.27	12.17	True
4	15.7	5.58	-2.58	16.34	17.49	146.58	-0.64	25	9.86	106.56	-4.36	1.68	1	1.28	NaN	False

```
In [4]:  1  df_diabetes.tail()
```

Out[4]:

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15	Class
1495	37.7	33.90	5.80	6.62	10.71	362.58	-1.52	165	5.52	444.56	-1.96	15.30	10	6.93	8.76	False
1496	17.7	29.40	8.00	-0.48	3.54	-102.42	1.17	100	3.76	304.56	6.78	29.25	10	7.53	12.19	True
1497	11.7	2.13	-0.92	12.12	22.65	95.58	-0.57	10	8.47	76.56	-4.76	2.34	1	1.89	NaN	True
1498	11.7	2.94	0.64	11.68	17.49	146.58	1.47	20	8.57	116.56	-5.00	2.67	1	1.48	11.55	False
1499	27.7	30.75	7.76	1.84	8.67	137.58	-2.02	80	4.04	304.56	3.90	20.40	10	6.93	10.41	False

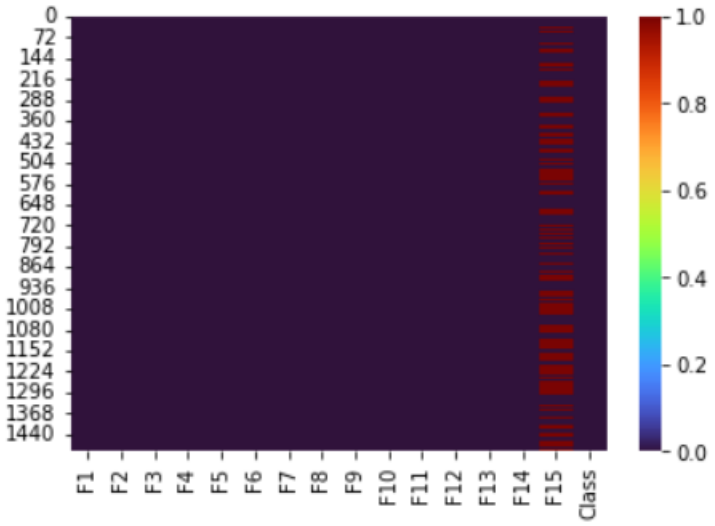
Both of the figures 1 and 2 shows summary of our data.

Step 2: Data Cleaning

Here we can clearly see that for feature 15 there are null values. We must need to fix that to ensure that it doesn't impair the ability/accuracy of the model. So, we will fill the null values with the mean values.

```
In [5]:  1 sns.heatmap(df_diabetes.isnull(),cmap ="turbo")
```

```
Out[5]: <AxesSubplot:>
```



Here Figure 3 describes the null values of our dataset

Step 3: EDA (Exploratory Data Analysis):

Exploratory data analysis allows let us comprehend how data has been dispersed and how we might use it to generate relevant insights [3]. Describes how we obtained some important data insight: - The minimal value for several columns is 5.7, as we can see. As an outcome, based on the distribution, we would have to replace null values throughout the data cleaning process with mean/median values. The number 57.7 was also there in maximum column.

```
In [14]:  1 sns.countplot(x="Class", hue="F1", data=df_diabetes)
Out[14]: <AxesSubplot:xlabel='Class', ylabel='count'>
```

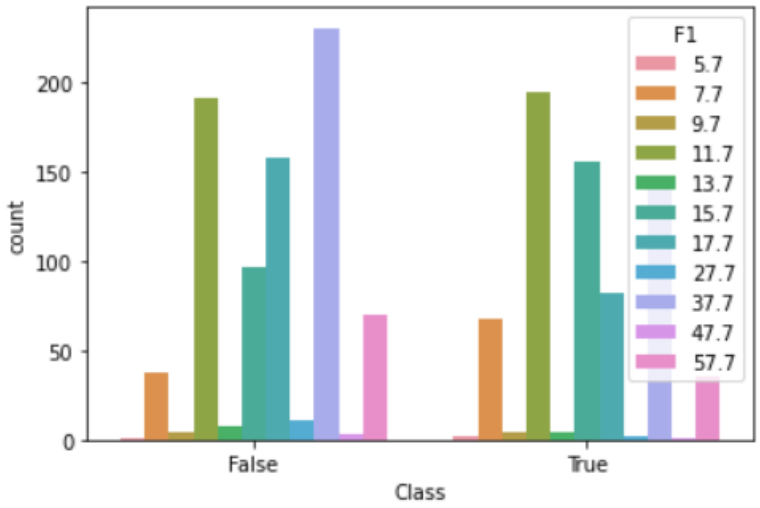


Figure 4 shows target variable F1's distribution

Data visualisation was used to establish if the dataset was balanced or not. Box plots have been used to study distributions and detect outliers, whilst histograms are being used to assess if data has been distributed. Lastly, F2, F4, and F13 have a normally distributed. Outliers are unusual variables in the dataset that might cause statistical analysis to be misinterpreted. at last, how they're addressed is essential. In this scenario, eliminating outliers can lead to loss of data; so, we must cope with it by applying multiple scaling and transformation approaches.

```
In [6]:  1 # distributions plot for feature 'F1'
         2 sns.distplot(df_diabetes.F1)
         3 plt.show()
```

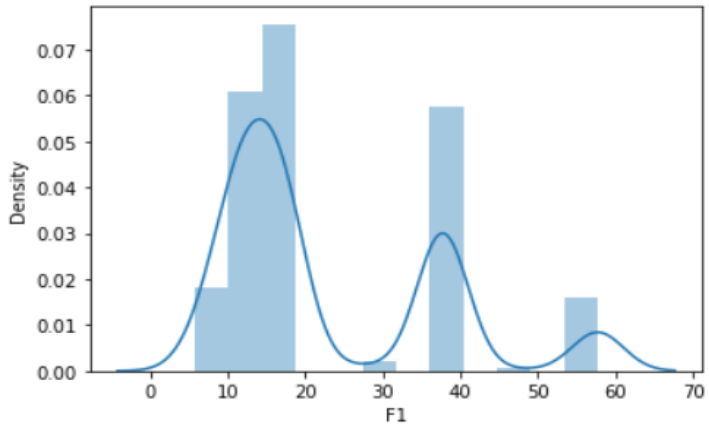


Figure 5 indicates the F1's distribution

Step 4: Engineering of features

Feature engineering, often known as applied machine learning, which is a strategy for increasing the performance of machine learning algorithms. Computation in data analysis and machine learning methods becomes much more easy by focusing on the most essential features and lowering the dimension of the feature set [4].

- *Standard Scaler*

  To cope with outliers & to avoid false predictions, we applied the standard scaler, which standardizes the characteristics to a 0-1 range. This also made sure that negative values have been adjusted to zero and positive values have been adjusted between one and zero.

- *Co-relation Matrix*

  Correlation matrix Assists in establishing the link between two variables. It indicates the strength of the relationship among 2 variables. The coefficient's value can vary between -1 and +1. They are strongly correlated, if they are 1; else, they're not correlated. A heat map is 2-dimensional depiction of information.

Step 5: Data Splitting

Here we will split our data into 2 parts train dataset and test dataset. By dividing the data in 80:20 ratio and separating it into train and test, so that we can examine how well the model would work. This should help us to make the precise model it's parameters and assure accuracy whenever we ultimately include the test data.

Step 6: ML algorithm:

There are plenty of machine learning algorithms with various parameters, but we are going to use and get our final results by using these algorithms: Logistic regression, Decision Tree, K-Neighbour, Random Forest, Support Vector Machine.

Step 7: Performance of the model

Here I have stated accuracy of my model after dropping F15 column. Random Forest has 82.22% accuracy, Support vector machine got 79.4%, Logistic Regression has 79.5%, Support Vector classifier achieved 74.4% accuracy, Decision tree model have got 72.3% accuracy, K-NN got 69% accuracy only. So as random forest got the highest accuracy so it is the best model among all other models that we have created, so we will use it. Furthermore, we could see the importance of the feature, in which it shows that how our features are useful for the prediction.

```
In [54]:
1 plt.figure(figsize=(12,6))
2 feature_imp = pd.Series(rfc.feature_importances_, index = X.columns).sort_values(ascending = False)
3 feature_plot = sns.barplot(x = feature_imp, y = feature_imp.index)
4 feature_plot.set_title("Feature Importance Plot")
5 feature_plot.set_xlabel("Score")
6 feature_plot.set_ylabel("Feature")
```

```
Out[54]: Text(0, 0.5, 'Feature')
```

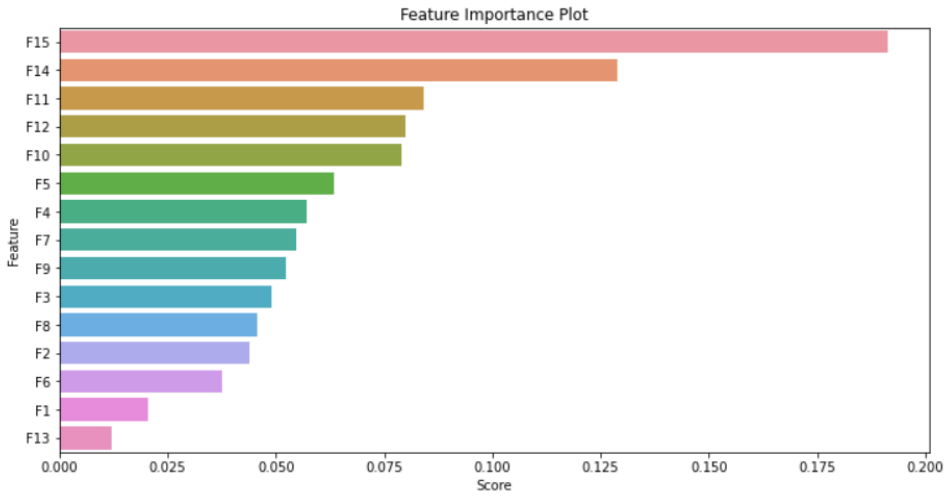


Figure 6 indicates the feature importance of our dataset

Step 8: Evaluation Metrix:

Diabetes people who were already healthy and was misdiagnosed as a diabetic are resulted into false positive. Diabetes patients were mistakenly labelled as healthy, resulting in a false-negative result. Diabetes was appropriately diagnosed as diabetic, resulting in a true positive. The term "healthy" is accurately recognised as a true negative.

The ROC is also shown, which is described as the area under the curve and may be computed by plotting the true positive rate vs the false positive rate at vast

of sets of rules. The F1 Score may be a preferable metric to be used if we need to strike a balance among recall and precision.

```
In [53]:
print('AUC of test: ', roc_auc_score(Y_test, y_prob)) #for test data

fpr, tpr, thresholds = roc_curve(Y_test, y_prob)

plt.plot(fpr, tpr)
plt.plot(fpr, fpr, 'r-')
plt.xlabel('FPR')
plt.ylabel('TPR')
plt.title('ROC_Curve of Test data')
plt.show()
```

```
AUC of test:  0.9136651583710407
```

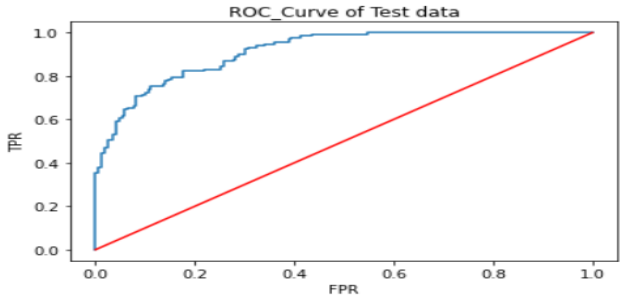


Figure 7: we can see the ROC curve

As discussed above random forest if the best among all other models and we are using is to complete our prediction.

```
In [46]:

conf_matrix =(confusion_matrix(Y_test, predictions_rfc))

# Plotting the confusion matrix graph using Matplotlib

fig, ax = plt.subplots(figsize=(7.5, 7.5))
ax.matshow(conf_matrix, cmap=plt.cm.Blues, alpha=0.3)
for i in range(conf_matrix.shape[0]):
    for j in range(conf_matrix.shape[1]):
        ax.text(x=j, y=i,s=conf_matrix[i, j], va='center', ha='center', size='xx-large')
plt.xlabel('Predictions', fontsize=18)
plt.ylabel('Actuals', fontsize=18)
plt.title('Confusion Matrix', fontsize=18)
plt.show()
```

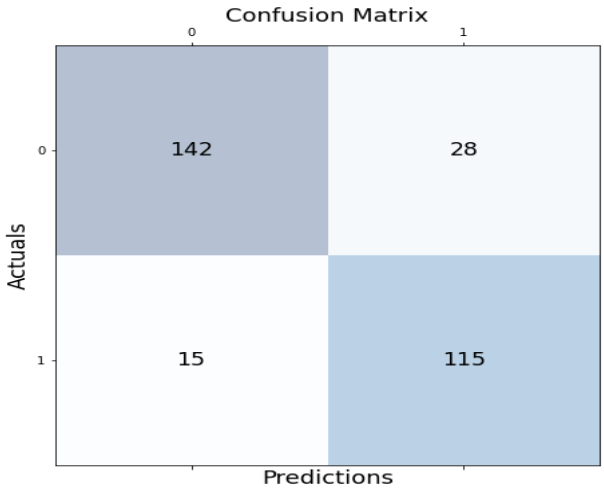


Figure 8 indicates our confusion matrix

ESTIMATE THE DOSE OF A NEW DIABETES MEDICINE THAT A PATIENT SHOULD CONSUME

First, we have imported all the necessary library packages, after that we will start processing our data and check if there are any null values.

```
In [11]:  1 sns.countplot(x="F4", data=df)
```

```
Out[11]: <AxesSubplot:xlabel='F4', ylabel='count'>
```

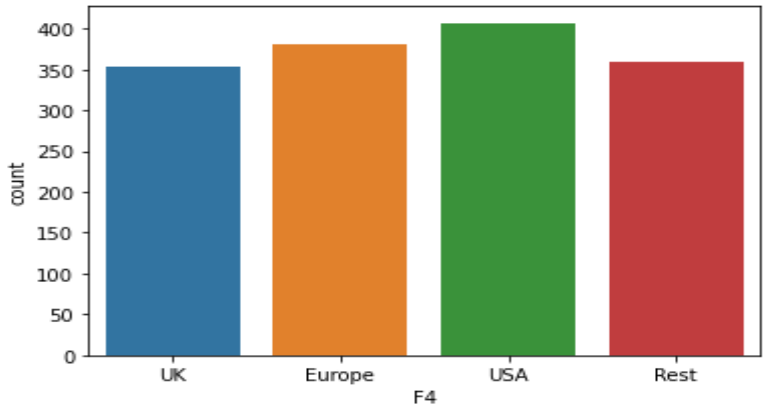


Figure 9 indicates the distribution in different countries

We can clearly see that US has a high diabetic patient than any other countries because their medicine intake is high

Here I have used the pie chart to show the feature F15's distribution.

```
In [7]:  1 print(df['F15'].value_counts())
         2 plt.figure(figsize=(10,5))
         3 df['F15'].value_counts(normalize=True).plot(kind='pie')
         4 plt.ylabel('counts')
         5 plt.xlabel('F15')
```

```
Very low     312
Low          310
High         301
Very high    290
Medium       287
Name: F15, dtype: int64
```

```
Out[7]: Text(0.5, 0, 'F15')
```

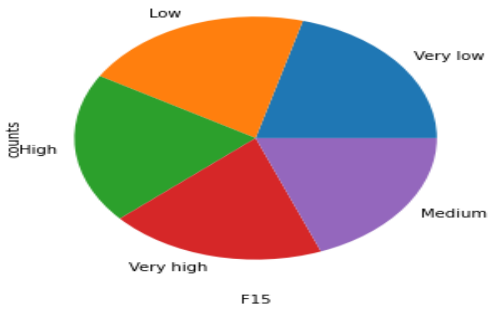


Figure 10 indicates the value count of F15

```
In [13]:
1  corrmat = df.corr()
2
3  f, ax = plt.subplots(figsize =(9, 8))
4  sns.heatmap(corrmat, ax = ax, cmap ="RdYlGn_r", linewidths = 0.1)
```

```
Out[13]: <AxesSubplot:>
```

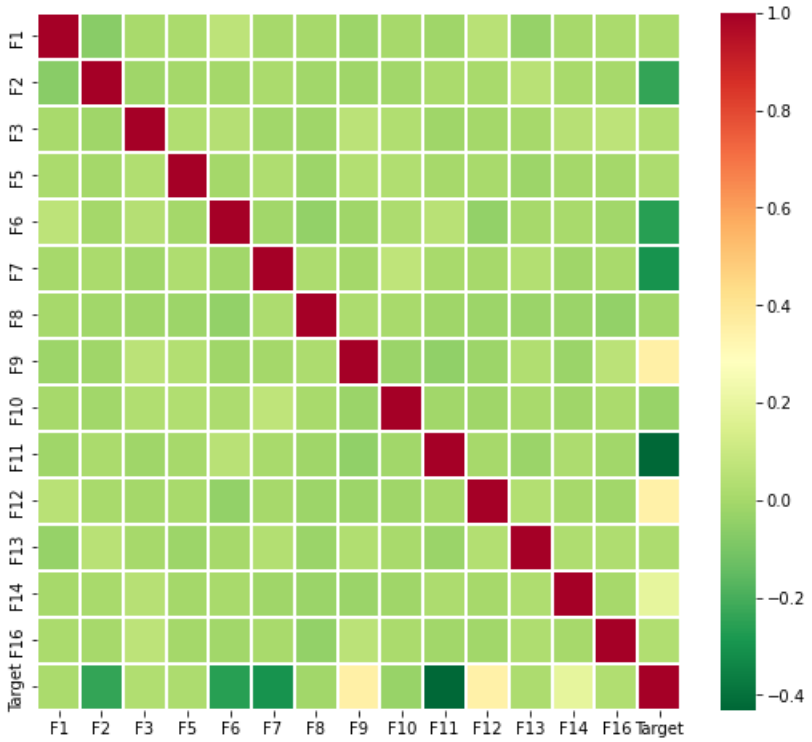


Figure 11 demonstrating the overall correlation.

Here we can see that F9, F12 and F15 are correlated highly, and that can help us to predict.

Data Pre-processing:

Previously said that column F15 is an object, we have converted it into the binary with the help of label encoder. At last, we successfully generated the set of dummy variables based on the country variable.

Data Splitting:

Here we will split our data into 2 parts train dataset and test dataset. By dividing the data in 80:20 ratio and separating it into train and test, and the added them into random state to ensure we can the same output while we execute our code at any time or location.

Machine learning algorithms:

WE are considering following regressors for regression task: Support vector machine, Linear regression, LightGBM, Decision tree regressor.

Model Performance:

After training our model with multiple algorithms, we got result as below: Extra tree regressor got 0.0000000e+00 Mean Squared Error, LightGBM got 1.523499e-03 Mean Square Error. Linear regression got 1.163562e-30 Mean Square Error, Support Vector Classifier got 9.965711e-03 Mean Square Error.

Evaluation Metrics:

The (MSE) mean square error used as a metric. The MSE measure is used to determine how near a fitted line is to data points [5]. The vertical distance amongst each data point and the matching y value on the curve fit should be squared. All other strategies were exceeded by linear regression, making it the ideal option for forecasting hold test results. When the data was not generalised, the additional tree regressor performed badly.

Conclusion:

With diabetes on the rise, we could use machine learning to estimate how much of a new diabetic medicine a patient must consume per day with the help of mean square error which is 1.163562e-30, showing that the model is successful. By doing this, we can be assured that our patients are safe and also the right amount of medications are taken. In the future, the developed methodology using machine learning classification algorithms might be utilised to predict or detect various illnesses. In the future, we may propose for the use of characteristics which can be detected instead of those who have been evaluated through principal component analysis [6].

References:

[1] T. J. e. a. Harris, "Impact of the new American Diabetes Association and World Health Organisation diagnostic criteria for diabetes on subjects from three ethnic groups living in the UK.," *NUTRITION METABOLISM AND CARDIOVASCULAR DISEASES ,* vol. 10, no. 6, pp. 305-310, 2000.

[2] W. e. a. Ling, "Global trend of diabetes mortality attributed to vascular complications," *Cardiovascular Diabetology,* vol. 19, no. 1, pp. 1-12, 2020.

[3] M. L. C. D. a. M. C. Vigni, "Exploratory data analysis," *ata handling in science and technology,* vol. 28, pp. 55-126, 2013.

[4] A. a. A. C. Zheng, "eature engineering for machine learning: principles and techniques for data scientists.," *" O'Reilly Media, Inc.",* 2018.

[5] D. M. Allen, "Mean square error of prediction as a criterion for selecting variables.," *Technometrics ,* pp. 469-475., 1971.

[6] J. e. a. Yang, ""Two-dimensional PCA: a new approach to appearance-based face representation and recognition.," *IEEE transactions on pattern analysis and machine intelligence,* vol. 26, no. 1, pp. 131-137, 2004.