Design and Application of a Machine Learning System for a Practical Problem

Prashant Raj, School of Computer Science and Electronic Engineering University of Essex

May 4, 2022

Registration Number:	2112123
Name:	Prashant Raj
Subject:	Machine Learning(CE802)

1 Introduction

When it comes to the decision making in the field of healthcare, Machine learning plays an important role.It helps us to make the informed decision which can be perfectly allied to the different problems in this sector.our aim to establish whether the addition of decision support software to assist in the interpretation of diabetic they have or not using classification problems.The classification is done based on the most influential from the data set.Random forest,Decision tree and K-nearest Neighbours model will be use to predict accuracy in this case study .

2 Methodology

The diabetic data collected from the University Of Essex repository where given data are very imbalanced with majority of false cases exhibit variance in the histogram values.The given data is in some missing value but there are in not duplicate value.The data set contains 15 features and 1500 rows to monitor the diabetic they have or not.

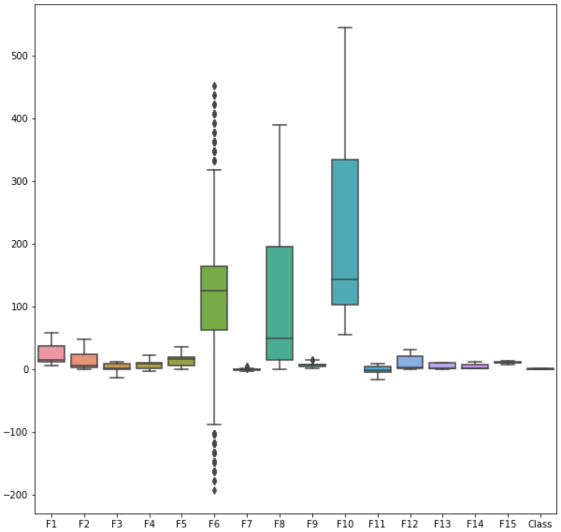


Figure 1: Box plot of diabetic data

3 Classification

3.1 Exploratory data analysis (EDA)

Exploratory data analysis is the initial analysis of the data which completed before pre-processing. It is also helping to discover co-relation between the data and perform the impact the natural of the data set being deployed for the model which is help to improve model performance and summarise the critical feature of the data set. Initially to build our classifier models all the required libraries must be imported (NumPy, pandas,sklearn). After importing the packages, load the give data.csv file using python command into the data frame. Once csv has loaded the file needs to be checked if there are any nan or missing values and need to fill them using mean. The data types of the variables are examined to see whether any category features exist as there are no categorical variables in the data, it is examined for missing values. There are 750 missing values in the column "F15.", For the sake of efficiency, the column had updated bu median in the dataset. Similarly, the same steps had performed on the test dataset as well. Fig 2 shows the information regarding class and count. Need to separate the data attributes and labels into train and test data sets now that we have retrieved the data attributes and labels. We will use the 'train test split' function from scikit-learn for this, which takes the attributes and labels as inputs and outputs the train and test sets.

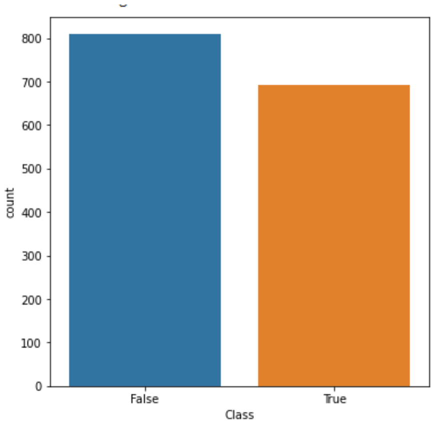


Figure 2: Bar chart of diabetic data

Feature Engineering is the process of transforming raw data into useful feature that help us understand our model better.

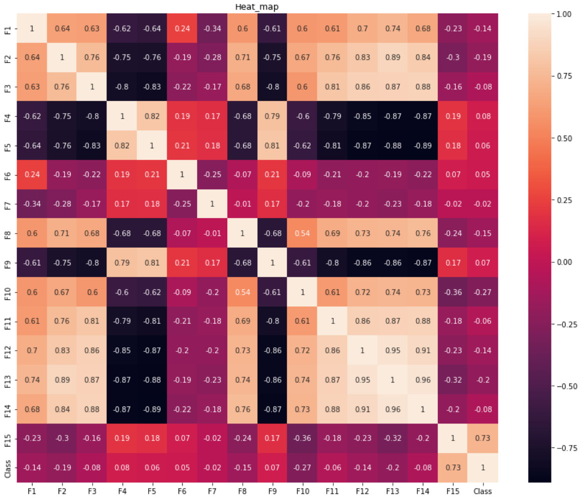
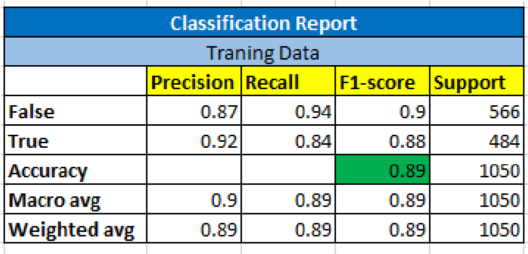


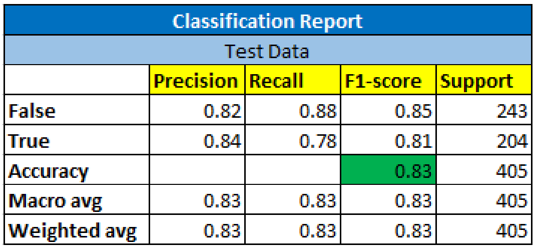
Figure 3: heat map of diabetic data

3.2 Decision Tree Classifier

In this section we concentrate on the classification on the decision tree result which obtained by the trained decision tree model using sklearn library's.Our work identified the get best result after the data pruning with changing fold cross-validation. We performed gridserchview operation to find the best parameters that give optimum accuracy.After the prediction of training model we achieved highest accuracy 89 percent on train data and 83 percent on test data. classification report is the another way to evaluate the classification model performance.it display the precision ,recall,F1 and support score for the model.below are the classification report.



Classification Report				
Traning Data				
	Precision	Recall	F1-score	Support
False	0.87	0.94	0.9	566
True	0.92	0.84	0.88	484
Accuracy			0.89	1050
Macro avg	0.9	0.89	0.89	1050
Weighted avg	0.89	0.89	0.89	1050



Classification Report				
Test Data				
	Precision	Recall	F1-score	Support
False	0.82	0.88	0.85	243
True	0.84	0.78	0.81	204
Accuracy			0.83	405
Macro avg	0.83	0.83	0.83	405
Weighted avg	0.83	0.83	0.83	405

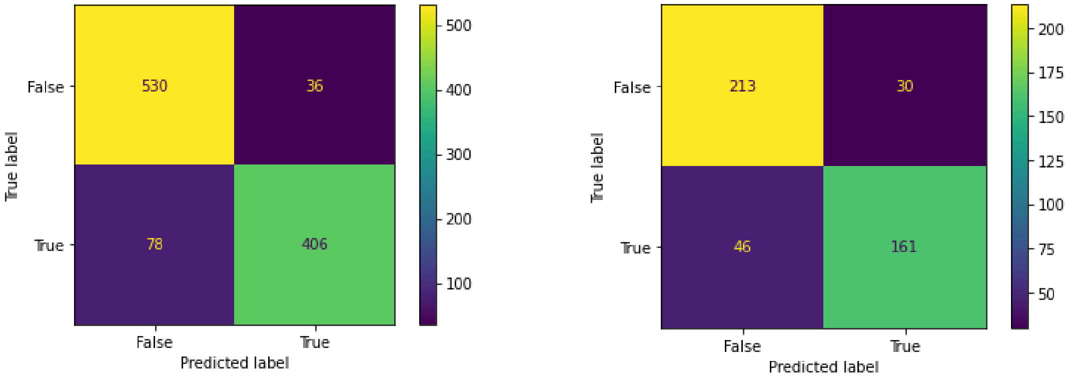
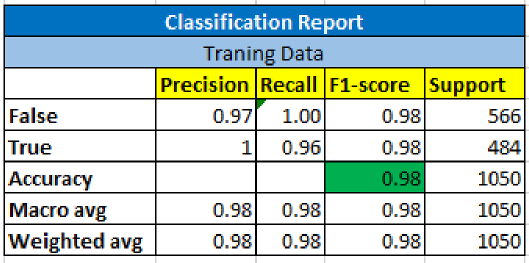


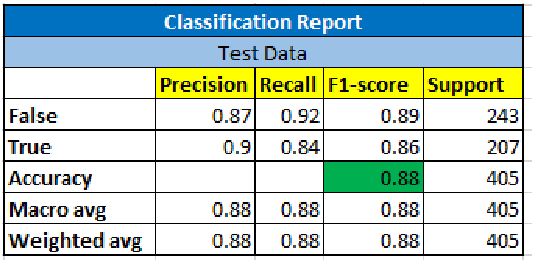
Figure 4: Confusion Tree

3.3 Random Forest Classifier

In this section,we use same train test spilt data and will fit in the random forest classification model.Here our aim to get better accuracy from the other model.so first time after fitting random forest model we got 100 score after that data pruning and finding the best parameters using grid search view we get 98 percent accuracy on training data and 88 percent on test data.Below is shown in the below classification report.



Classification Report				
Traning Data				
	Precision	Recall	F1-score	Support
False	0.97	1.00	0.98	566
True	1	0.96	0.98	484
Accuracy			0.98	1050
Macro avg	0.98	0.98	0.98	1050
Weighted avg	0.98	0.98	0.98	1050



Classification Report				
Test Data				
	Precision	Recall	F1-score	Support
False	0.87	0.92	0.89	243
True	0.9	0.84	0.86	207
Accuracy			0.88	405
Macro avg	0.88	0.88	0.88	405
Weighted avg	0.88	0.88	0.88	405

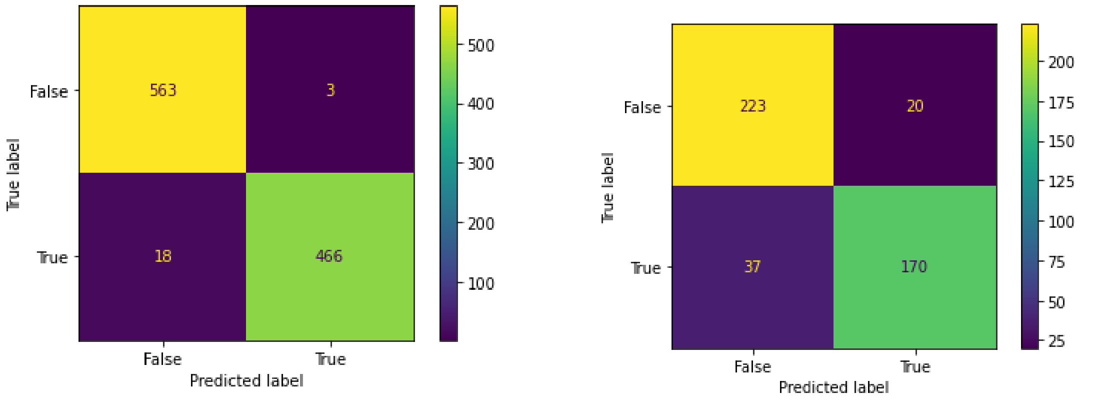
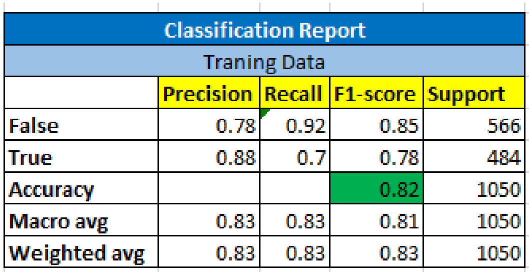


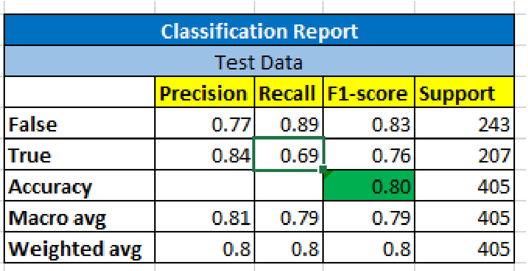
Figure 5: Confusion Tree

3.4 K-Nearest Neighbors Classifier

In K-Nearest Neighbors Classifier where first need to be scale training data and test data after that we achieved 83 percent score but our aim to identified higher score to get best result form this model.However after using grid search view we find best parameters and fit the model with best parameter then we identify 82 percent on training data and 80 percent on test data which is highest.



Classification Report				
Traning Data				
	Precision	Recall	F1-score	Support
False	0.78	0.92	0.85	566
True	0.88	0.7	0.78	484
Accuracy			0.82	1050
Macro avg	0.83	0.83	0.81	1050
Weighted avg	0.83	0.83	0.83	1050



Classification Report				
Test Data				
	Precision	Recall	F1-score	Support
False	0.77	0.89	0.83	243
True	0.84	0.69	0.76	207
Accuracy			0.80	405
Macro avg	0.81	0.79	0.79	405
Weighted avg	0.8	0.8	0.8	405

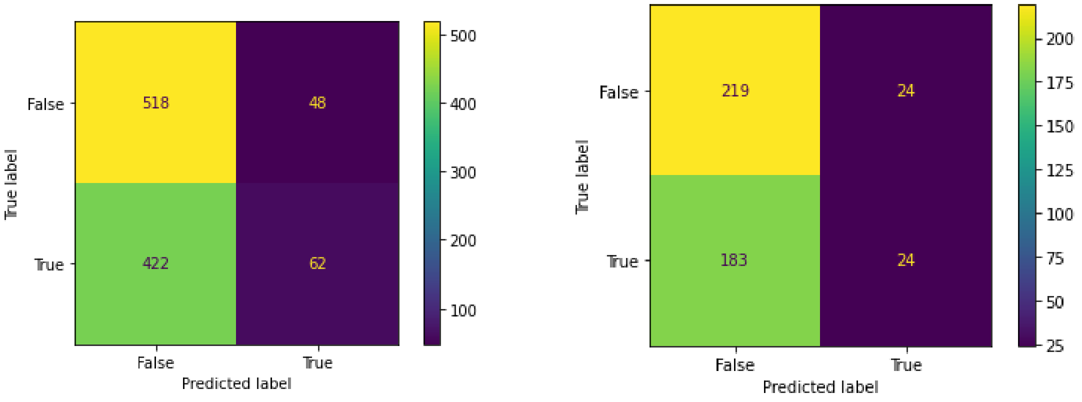
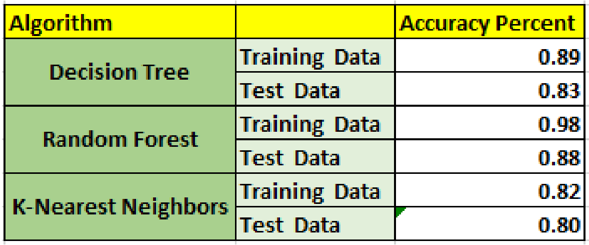


Figure 6: Confusion Tree

4 Classification Result

From the classification results,we can conclude that the Random Forest outperforms all the other classification methods we have used in the model where accuracy is 98 percent with train data and 83 percent on test data. .The next better performing model after the random forest is the decision tree with 89 percent accuracy with train data and 83 percent with test data. In our classification we received the lowest accuracy in K-nearest Neighbour with 82 percent accuracy on the training data and 80 percent with test data. We can clearly see that Random Forest perform best accuracy as comapre to others.



Algorithm		Accuracy Percent
Decision Tree	Training Data	0.89
	Test Data	0.83
Random Forest	Training Data	0.98
	Test Data	0.88
K-Nearest Neighbors	Training Data	0.82
	Test Data	0.80

Figure 7: Classification Result Report

5 Regression

5.1 Data Pre-processing

In this section ,we use perform same opertaion as per classification .However we examined whether any category features exist as there are two categorical variables in the data, it is examined missing values. There are no missing values in the input data. The categorical features are encoded using one hot encoding to convert them into binary form. The input and output variable are separated in order to predict the score similarly, Need to separate the data attributes and labels into trainand test data sets now that we have retrieved the data attributes and labels. We will use the 'train test split' function from scikit-learn for this, which takes the attributes and labels as inputs and outputs the train and test sets.

5.2 Linear Regression

After splitting the data, the model is fitted over the training data set using Linear Regression from sklearn library .We must import the Linear Regression class, instantiate it, and pass our training data to the fit() method. Linear regression model finds the optimal value for the intercept and slope, resulting in the best-fitting line for the data using gridsearch .after getting best paramter algorithm has been trained the predictions are made on the test data and obtained the below details.

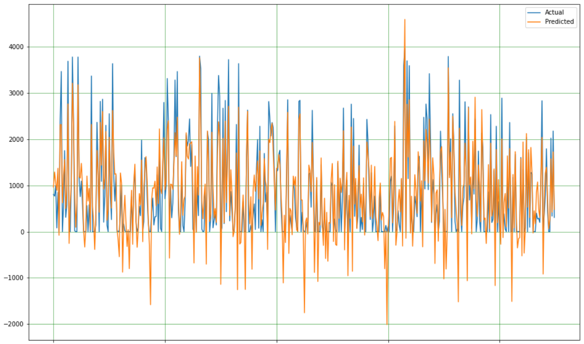
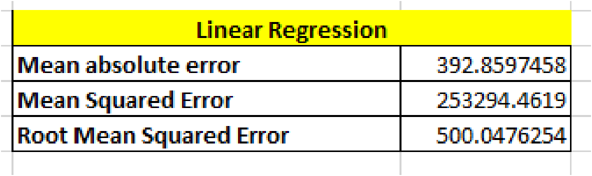


Figure 8: Linear Regression

5.3 Random Forest Regression

The Random Forest regression class is imported and is set to the regression variable to train the model. .fit() is used to fit the model on training data. In this scenario random forest is implemented using grid search cross validation which searches for the combination of best parameters to be implemented



Linear Regression	
Mean absolute error	392.8597458
Mean Squared Error	253294.4619
Root Mean Squared Error	500.0476254

Figure 9: Linear Regression

on the model for best accuracy. The results of grid search are passed into the random forest regression which is in turn used to fit the training data. The same regression model is used to make predictions on test data and unseen data. Mean squared error metric is used to evaluate the model and the value of mean squared error of the random forest regression is below.

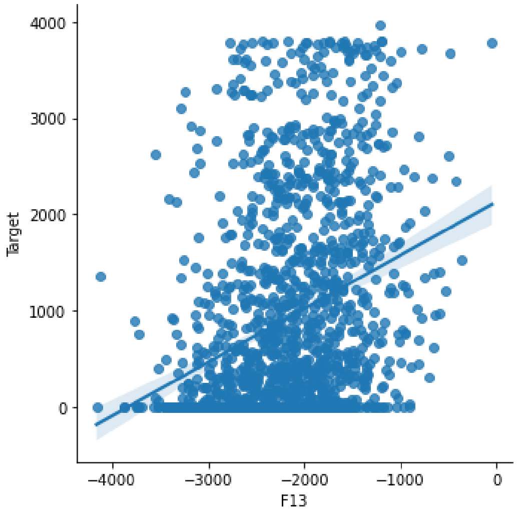
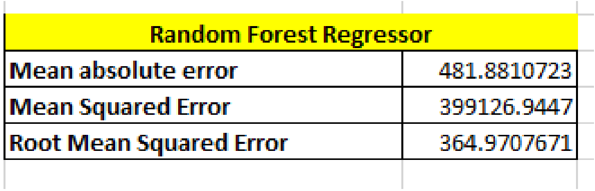


Figure 10: Random Forest

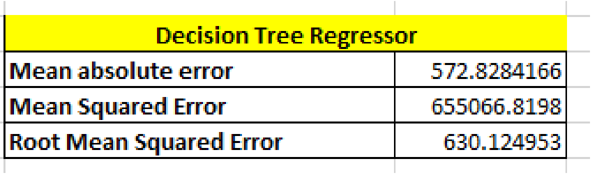


Random Forest Regressor	
Mean absolute error	481.8810723
Mean Squared Error	399126.9447
Root Mean Squared Error	364.9707671

Figure 11: Random Forest

5.4 Decision Tree Regression

In this model,The best parameters according to the grid search Passing these parameters to the decision tree, the model is fit on the training data. The model is further used to make predictions on the test data and unseen data. Mean squared error metric is used to evaluate the model and the value of mean squared error of the random forest regression is below.



Decision Tree Regressor	
Mean absolute error	572.8284166
Mean Squared Error	655066.8198
Root Mean Squared Error	630.124953

Figure 12: Decision Tree

6 Regression Result

On comparing the Mean Squared Errors of the three Regression techniques, Linear Regression model has less Mean Squared Error when compared to the other two. Therefore, it is suitable to make predictions on diabetes patients. The predictions are copied to the csv file.

References

[1] Sharanya, S., and Venkataraman, R. (2020). An intelligent context-based multi-layered Bayesian inferential, predictive analytic framework for classifying machine states. Journal of Ambient Intelligence and Humanized Computing, 12, 1–9.

[2] Cömert, Z., Şengür, A., Budak, Ü., and Kocamaz, A. F. (2019). Prediction of intrapartum fetal hypoxia considering feature selection algorithms and machine learning models. Health Information Science and Systems, 7(1), 1– 9.

[3] Akhtar, F., Li, J., Azeem, M., Chen, S., Pan, H., Wang, Q., and Yang, J. J. (2019). Effective large for gestational age prediction using machine learning techniques with monitoring biochemical indicators. The Journal of Supercomputing, 76, 1– 9.

[4] Magenes, G., and Signorini, M. G. (2021). Cardiotocography for fetal monitoring: Technical and methodological aspects. In In innovative technologies and signal processing in perinatal medicine (pp. 73– 97). Springer.