Diabetes Prediction Using Various Machine Learning Algorithms

(EEN-366: Course Project)

Anjani Kumar Tiwari (20115015)

Abstract:

In this project we have classified that whether a patient has diabetes or not by making a classifier from its 8 features dataset using several machine learning algorithms and compared them for the best classifier. The dataset is taken from data. world website. We have classified the dataset in three parts training (80%), validation (10%), and testing (10%). then we tuned the hyper-parameters of respective algorithm by observing training and validating accuracies and plotted their graph. Then we found the point where training and validation accuracy is best, by taking that value of hyper-parameter we have modelled the classifier and find the testing accuracy, confusion matrix, precision, recall, and F-1 score. By comparing values of these quantities, we have decided which classifier is best for the given dataset.

3. Main objectives:

- Predicting whether the patient have diabetes or not by observing the features taken from the patient medical data, check-up and applying machine learning algorithms.
- Comparing the results obtained from the different ml algorithms and suggesting the best model out of them.

2. Introduction:

Now a days lot of human being are affected by different type of metabolic disorders. Most common disorder among these is Diabetes. For most of doctors it is not so easy to tell anyone very quickly just by few sampling information. There are other competitors also available in the market, so for proving best result and service we need to detect the disorder in very less time otherwise they can go to somewhere else for medical report. That's why we are training the model with the help of different significant data. For training the model we are using different method like decision tree, ada-boost, random forest, Support vector machine, K-nearest neighbours, logistic regression, Gaussian naïve bays for prediction.

3. Dataset and Pre-processing:

The dataset available on data.world have 768 records and each record have 8 features which have been taken from patient's diagnostic measurements.

Table 1: Snapshot of the Dataset

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1
...	...	...	...	...	...	...	...	...	...

We have only female's data. In the data the column outcome is the label having 1 if the patient is diabetic and 0 otherwise. Missing value in the dataset is handled by using the value of same feature in other record whose label is same with this record.

3.1 Splitting dataset into train-Validation-test sets

we have classified data in three parts training (80%) for training the model, validation (10%) to tune the hyperparameters, testing (10%) for testing the model accuracy for unseen data.

First, we will select the hyperparameter then we will train the model on training dataset then we will evaluate model with validation dataset. After this we will change the hyperparameters according to results on the Validation dataset and then again train the model. Finally, we have selected the best model out of it final classifier then test on test dataset and observe the result.

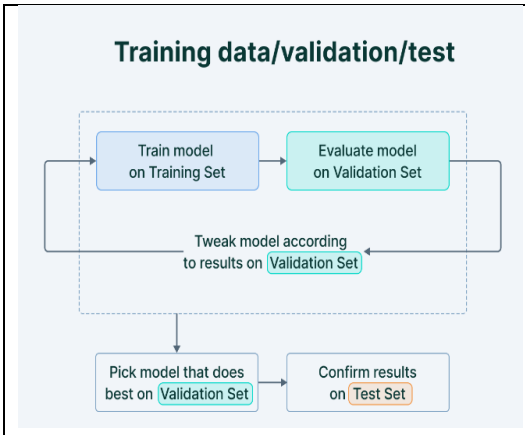


Fig-1: Validation Process

+

4. Data visualization:

We have used Principal Component Analysis for converting the data in 3 features space. PCA reduces the data in lower number of features by maximizing the variance of data. So here PCA reduces the data in 3 dimensions.

The eigenvalues in decreasing order are

```
[13456.57298   932.76013   390.57783]
```

And the percentage of eigenvalues are

```
[88.855  6.159  2.579]
```

In figure the yellow points the patient's not having diabetes and in violet the patient's having diabetes.

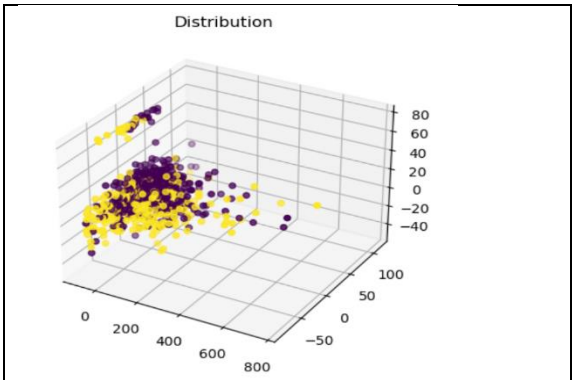
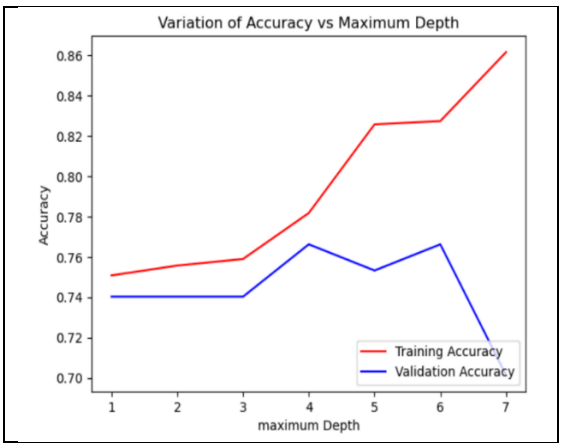


Fig-2: Data Visualization

5. Decision Trees:

Data is classified with Decision Tree. It has 8 features, so it is not going to much complex, but it is taking too much time. For selecting root node at every subtree, we have used Gini index and Gini gain for calculating randomness in the data.

The training accuracy of decision tree classifier depends on the maximum depth of the tree but, with increase in the depth the complexity increases, and validation accuracy decreases, below is the graph between the accuracy vs maximum depth of the tree.



We can see that a good Training and validation accuracy occurs at maximum depth of 4.

So, we have trained the classifier using maximum depth = 4 as hyperparameter and observed the results.

Accuracy of the model for training is 78.17%, on testing is 76.62%

Confusion-Matrix on testing

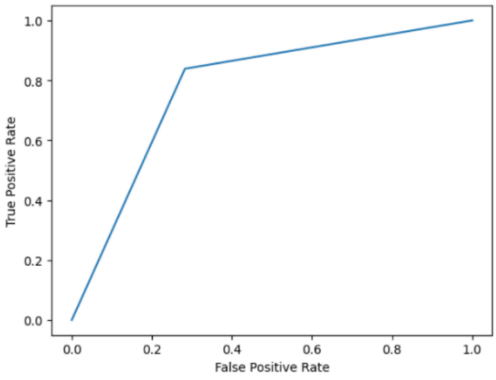
33	5
13	26

Precision score = 0.83

Recall score = 0.667
F-1 score = 0.74

Area-under ROC = 0.78

ROC-curve



6. AdaBoost:

The adaboost method is boosting ensembled model used for improvement in generalization. Its important slogan is learning from mistakes. In this method for making training and testing sets we divide the dataset, and in next classifier training set we take the sample with more probability which is miss-classified by the previous classifier. This method is robust in reducing the bias between different models. The Flowchart of applying the adaboost algorithm to dataset is given below

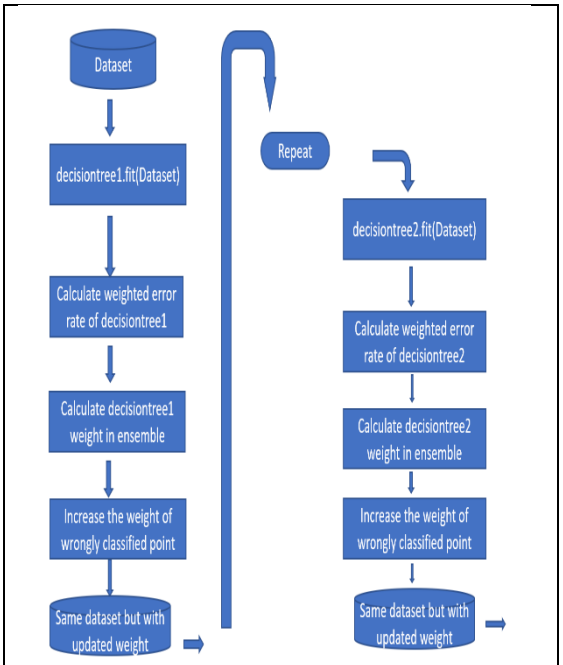
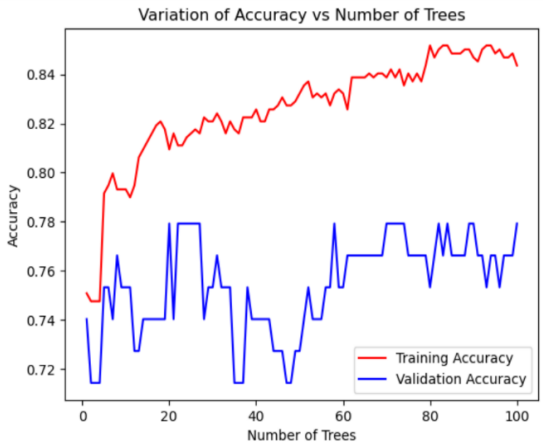


Fig-3: Flowchart of Adaboost

The graph of number of weak learners (trees) in taken in adaboost vs accuracy of classifier is given below.



We can observe that the training and validation accuracy is best achieved when the number of weak learners(trees) is around 90. So, we have taken 90 weak learners in a classifier and trained and test the classifier and observed the result.

The Training Accuracy is 84.69% and testing accuracy is 76.62%

Confusion-Matrix on testing

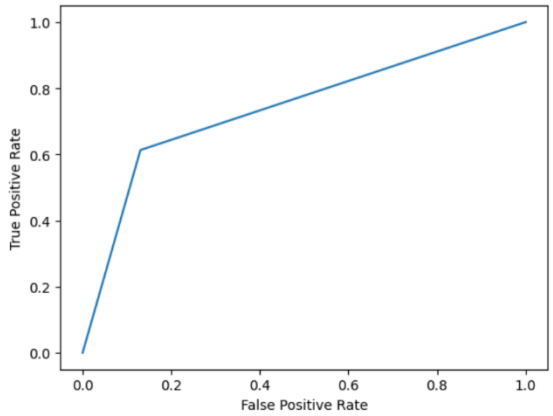
40	12
6	19

Precision score = .6129

Recall score = 0.76
F-1 score = 0.6785

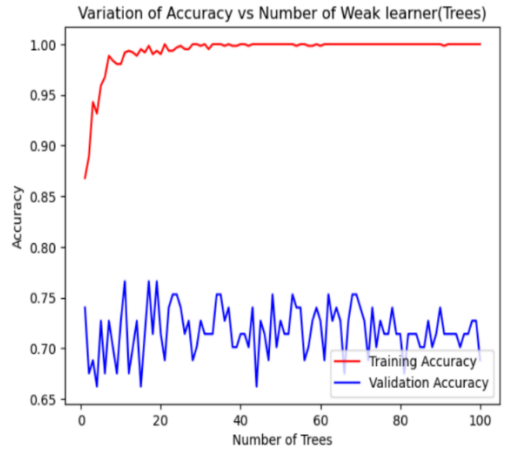
Area-under ROC = 0.7412

ROC-curve



7.Random Forests:

In classification using Random Forest we have used bagging to select the samples for training the different trees. The graph of training and validation accuracy vs number of trees plotted.



The maximum validation accuracy occurs at number of trees = 68. So, we have trained and tested final the classifier taking number of trees = 68 and observed their result.

Accuracy of the model for training is nearly 100%, on testing is 79.22%

Confusion-Matrix on testing

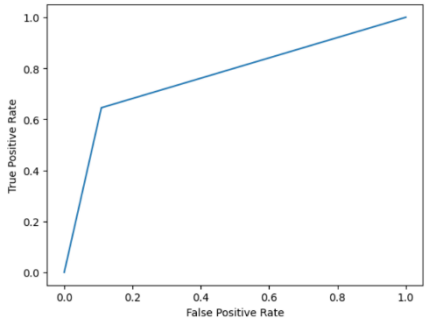
41	11
5	20

Precision score = .6451

Recall score = 0.8
F-1 score = 0.7142

Area-under ROC = 0.7682

ROC-curve

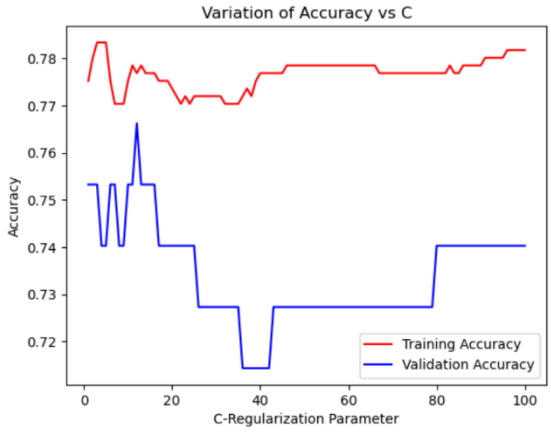


8.Support Vector Machine (SVM):

In this method we have used the support vector classifier on the training data. The SVM classify the data based on the maximum margin width between the two labels. It also has soft-Margin classifier for handling outliers. Outliers are the data points that are misclassified by model because of having feature values like of opposite class. Soft-Margin reduce complexity of the model and improve the generalization. SVM used here is polynomial kernel of degree 2. We have varied the penalty term (C), and calculate the training and validation accuracy, C is trade off factor which decides how much soft margin on outliers can be allowed.

Higher value of C means we are allowing less outlier and it will become a hard-margin SVM.

The below graph is training and validation accuracy vs C value from 1 to 100.



The maximum validation accuracy occurs at C=16 so we have taken the testing results at C=16

Accuracy of the model for training is 77.68%, on testing is 76.623%

Confusion-Matrix on testing

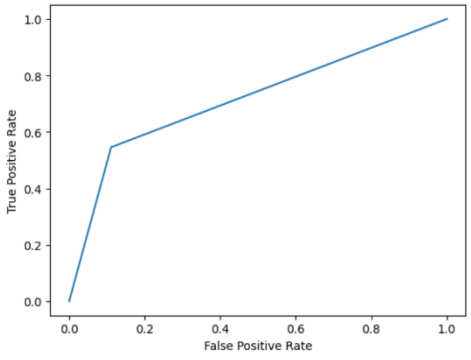
42	14
4	17

Precision score = 0.5485

Recall score = 0.8095
F-1 score = 0.6538

Area-under ROC = 0.7307

ROC-curve



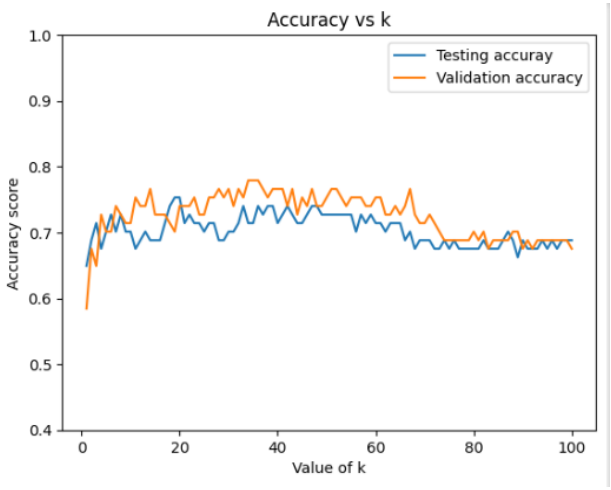
9.K -Nearest Neighbours:

It is one of most important classification algorithms that belong to supervised learning. It is used in machine learning. It is widely used for solving real-life problems.

The steps involved in this method are...

- First find the distance between them by Euclidean.
- Then sort the distance in ascending order
- Then select k-values from sorted distance.
- Assign the class to the sample based on the most frequent class in above K values.

We have hyper tune the classifier and calculate Accuracies of classifier and plot the graph w.r.t. to K value



A good validation accuracy occurs at K = 34. So, we have Model our classifier for this K. and calculate testing results,

We have found that

Testing accuracy =71.4%

Training accuracy =75.4%

Confusion matrix

42	4
18	13

Precision score=0.767

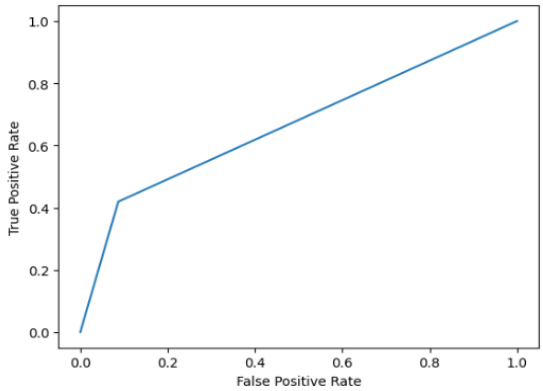
Recall score=0.419

F1 score=0.54

AUC=0.66

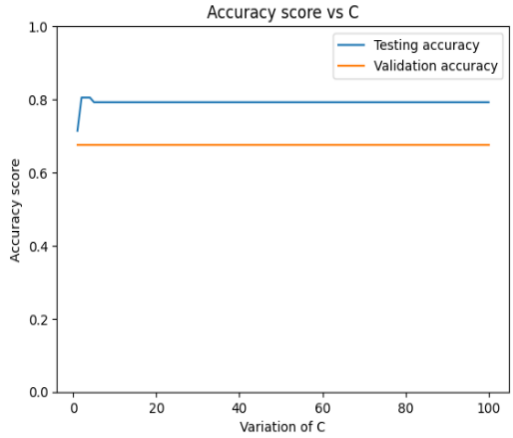
AUC=0.77

ROC curve



10.Logistic Regression:

In this method we can use different penalty, different solver and C. If we vary the value of C (inverse of regularisation strength) then overfitting can be avoided. We can vary the penalty and solver.



Testing accuracy = 80.51%

Training accuracy=78.99%

Confusion Matrix

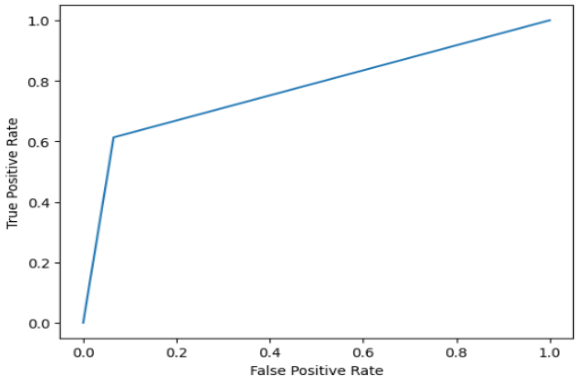
43	3
12	19

Precision score=0.86

Recall score=0.6129

F1-score=0.717

ROC Curve



11. Gaussian Naïve Bays:

It is based on Bays theorem of probability. It follows Gaussian normal distribution. It support continuous data. In this method we take some assumption that occurrence of certain feature is independent of the occurrence of other features.

Training accuracy =75.32%

Testing accuracy=78.99

Confusion Matrix

40	6
13	18

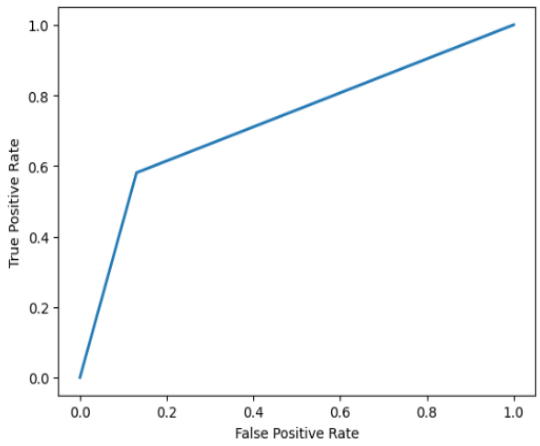
Recall score=0.58

F1-score=0.6545

AUC=0.725

Precision score=0.75

ROC Curve



11. Result and Conclusion:

Based on the results obtained in the following table, we can conclude that logistic regression gives a more accurate result. On the other hand, if we consider the precision then also logistic regression performs better, with the help of this we can reduce the chance of unnecessary treatment. If we want to identify all the patients who actually have diabetes then we need to consider the recall, if recall is maximum then there will be very less chance of leaving diabetes patients. Then, we need to consider the SVM model that has the highest recall. F1-score helps us to find the performance of the model which considers both precision and recall simultaneously then, the decision tree will be better.

Considering AUC, it will help us to correctly predict a randomly selected positive case as compared to a negative case. Hence, from the given data we can say that decision tree will be better for predicting randomly selected data.

Model	Accuracy	Precision	Recall	F1-score	Area under ROC
Decision Tree	0.77	0.83	0.67	0.74	0.78
Adaboost	0.76	0.61	0.76	0.68	0.74
Random Forest	0.79	0.64	0.80	0.71	0.77
SVM	0.77	0.55	0.81	0.65	0.73
KNN	0.79	0.77	0.42	0.54	0.66
Logistic Regression	0.80	0.86	0.61	0.72	0.77
Gaussian Naïve Bays:	0.75	0.75	0.58	0.65	0.72

12.References:

1.The data is taken from here https://data.world/data-society/pima-indians-diabetes-database

2. https://towardsdatascience.com/top-machine-learning-algorithms-for-classification-2197870ff501

3. https://www.diva-portal.org/smash/get/diva2:1448074/FULLTEXT01.pdf

4. https://en.wikipedia.org/wiki/Logistic_regression