Diabetes Prediction Using Machine Learning

Members:
Aar Rafi Shahriar Shad(19101469)
Nirjhar Gope(19301140)
Asif Tazwar(21301732)
Saymum Ahmed Nasif(17301086)

Couse Code: CSE422
Lab Section: 11

Submitted to: Atik Tajwar SIR and Sumaiya Aktar Miss
Brac University

Introduction: In this lab report, we will use the well-known Pima Indians Diabetes Database to help us predict if a patient has diabetes or not based on the data we feed into our machine learning model.Here, we will learn how the data analysis phase of a data science life cycle is carried out. In this project, we'll employ 3 machine learning models before selecting the one that performs the best by doing training and testing and calculating the accuracy of the models.

Dataset Description: The National Institute of Diabetes and Digestive and Kidney Diseases is the original source of this dataset. Based on specific diagnostic metrics present in the dataset, the dataset's goal is to diagnostically forecast whether a patient has diabetes or not. These instances were chosen from a bigger database under a number of restrictions. The datasets consist of one target variable, Outcome, and a number of medical predictor variables. The patient's BMI, insulin level, age, number of previous pregnancies, and other factors are predictor variables.

Number	Attributes	Description
1	Pregnancies	Number of times pregnant
2	Insulin	2-Hour insulin serum (μU/ml)
3	BMI	The index of body mass
4	Age	The Age (years)
5	Glucose	Concentration of plasma glucose for 2 hours in an oral glucose tolerance

		check
6	Blood Pressure	Blood Pressure Diastolic (mm Hg)
7	Diabetes PedigreeFunction	Diabetes pedigree function
8	Skin Thickness	Skinfold triceps thickness (mm)
9	Outcome	Range of value: 0 and 1(0 means no 1 means yes)

Pre-processing Techniques :

1.**Import necessary libraries and data**:We are using Pandas and Numpy as our main libraries. Pandas is used for data manipulation and data analysis and Numpy is a fundamental Python package for scientific computing. We are also using Matplotlib and Seaborn for the visualization and Scikit-learn libraries for data preprocessing techniques and algorithms.
2. **Read data**: Here, we are reading the data in the CSV file using pandas.
Checking for missing values
Checking for duplicate values
3. Explore the data
4. Visualize the data
5. Deal with the missing values
6.Show correlation between assets
7.Preprocess the data
8.Split the data for training
9. Train machine learning models
10.Evaluate scores

Models applied

- K-Neighbor Regressor
- Random Forest Regressor
- SVM(support vector machine)

Random Forest:

Random forest is a kind of Ensemble Classifier that focuses on producing better predictions using many learning algorithms that perform better together than they could individually. This classifier consists of a large number of decision trees that operate together. A random forest can have many decision trees within it and it is set using the n_estimator parameter. In our project, we have left this value as default which is 100. A larger n_estimator gives more accurate results but it is slower.

Each of the uncorrelated decision trees produces a prediction of its own and the result is determined by taking the average of these predictions or by majority voting. The random forest does not follow any set of formulas. Features are randomly selected and decision trees are created. Because of this, every decision tree is different from the other. The trees being uncorrelated plays a vital role in achieving high accuracy because it limits errors within the trees. If one tree faces an error, it will not affect the accuracy and performance of the others. Random forest is useful for datasets with a large number of features because even though a lot of trees are being used, it doesn't overfit the model.

Support Vector Classification:

SVM is a supervised machine learning algorithm. In 1963 The SVM was firstly introduced by Vapnik and Chervonenkis.For the binary classification problem is the support vector machine (SVM), which comes in different kernel functions. An SVM model's goal is to cal- culate a hyperplane (or decision boundary) based on a feature set in order to categorize data points.
The goal is to locate the hyperplane that separates the data points of two classes with the greatest margin, which may take many different forms in an N-dimensional space.

The SVM tries to locate an ideal hyperplane ready to isolate the examples of any class. This classifier specifies the hyperplane that isolates the spots to put the most noteworthy number of points of a similar class on a similar side while expanding the interval of each class to such a hyperplane. The support vectors comprise the closest points of the hyperplane. The interval from a class to a hyperplane is the littlest interval among them and the spots in that class
The hyperplane can be utilized for grouping or regression moreover. SVM separates examples in particular groups and can likewise characterize the substances which are not upheld by data. Detachment is finished by a hyperplane that plays out the partition to the nearest training spot of any group.

K-Neighbor Regressor :

The k-nearest neighbors (KNN) algorithm is a simple, easy-to-implement supervised machine learning algorithm that can be used to solve both classification and regression problems

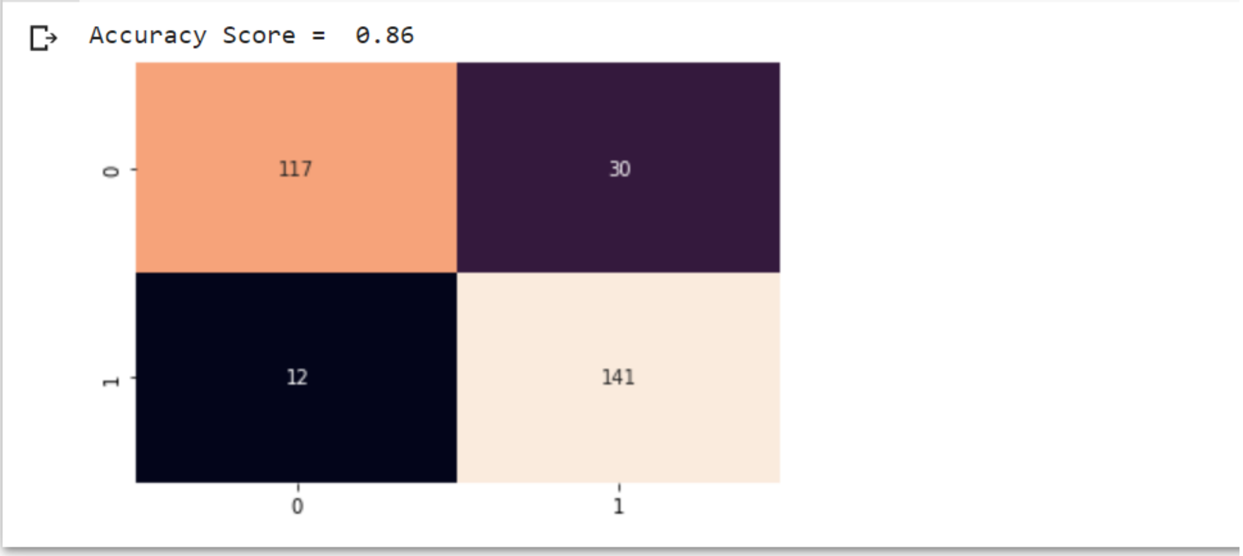
The KNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other.
To select the K that's right for your data, we run the KNN algorithm several times with different values of K and choose the K that reduces the number of errors we encounter while maintaining the algorithm's ability to accurately make predictions when it's given data it hasn't seen before.The algorithm is simple and easy to implement.
There's no need to build a model, tune several parameters, or make additional assumptions.

The KNN algorithm belongs to the group of algorithms that have a slow learning manner. Therefore, the data generalization is delayed until classification. To specify the class of an element that does not belong to the training set, the KNN classifier searches for k elements in the training set that are nearest to this obscure element

KNN is the name given to these k elements. The classes of these k neighbors are verified, and the most common class is assigned to the obscure element's class.

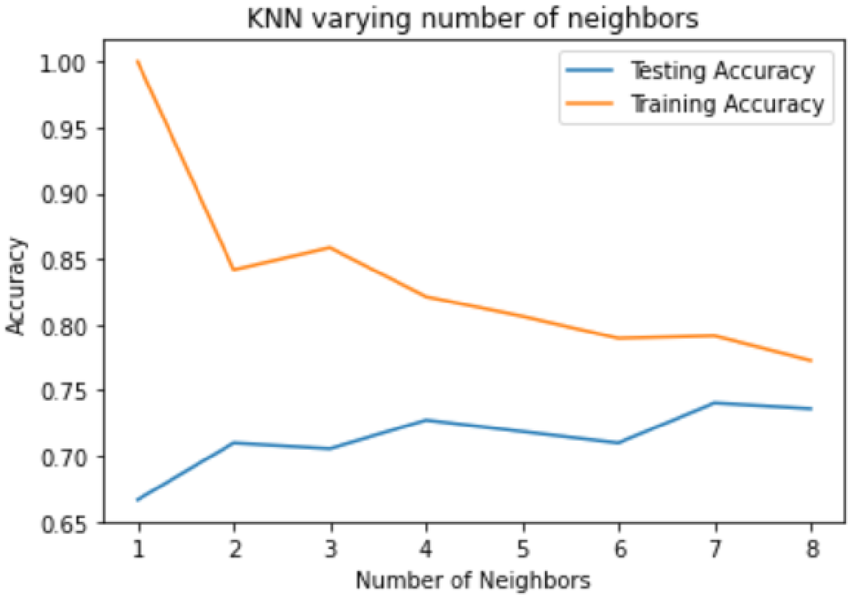
Results:

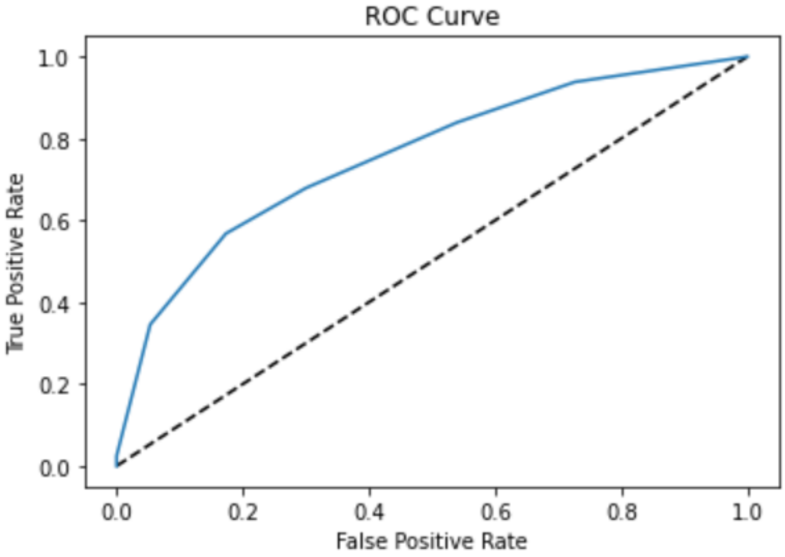
Confusion matrix:A confusion matrix is a table that is used to describe the performance of a classification model on a set of test data for which the true values are known. It makes it easy to see if the model is confusing classes



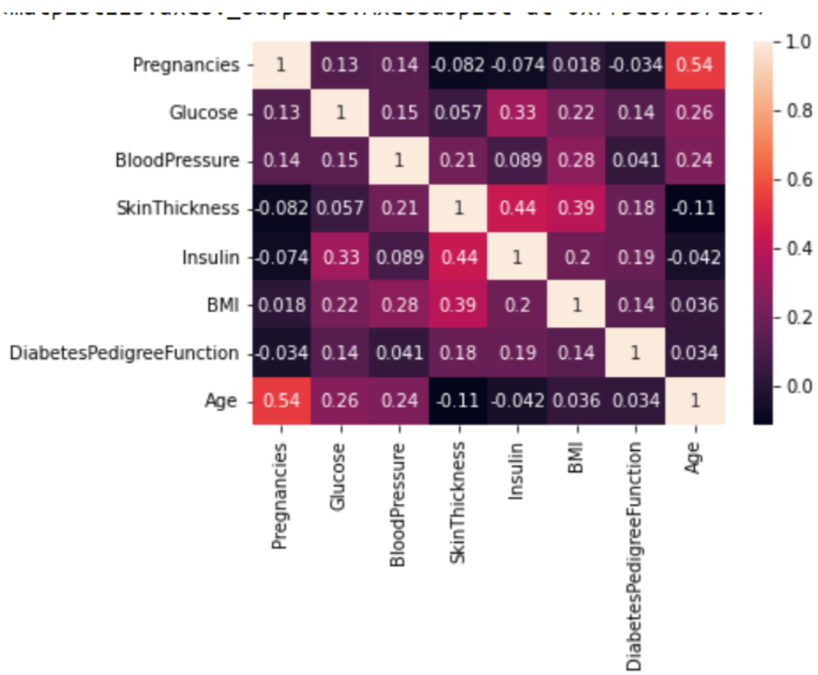
```
Accuracy Score =  0.86
```

	0	1
0	117	30
1	12	141





A useful tool when predicting the probability of a binary outcome is the Receiver Operating Characteristic curve, or ROC curve. It is a plot of the false positive rate (x-axis) versus the true positive rate (y-axis) for a number of different candidate threshold values between 0.0 and 1.0



Heatmaps are used to show relationships between two variables, one plotted on each axis. By observing how cell colors change across each axis, you can observe if there are any patterns in value for one or both variables.

Accuracy score, F1 score, Precision score and Recall score :

We can use some metrics like accuracy score, precision score, recall score, and F1 score to measure the performance of our models. The accuracy score is the ratio of correctly classified data instances and the total number of data instances. Accuracy is not a good metric when it comes to an unbalanced dataset. Precision is the ability of a classifier to label a negative sample as negative. The recall is the ability of the model to predict the positives out of samples that are true positives. F1 takes both precision and recall into account so if the precision score and recall score are both high, the F1 score will be high as well.

We have made a table with all the metrics that we had stored before. We can see the performances of the models at a glance from this table.

Model	Accuracy score	F1 score	Precision score	Recall score
K-Neighbor Regressor	0.75	0.81	0.76	0.87
Random Forest Regressor	0.86	0.85	0.91	0.80
SVM(support vector machine)	0.80	0.63	0.76	0.84

References

- ➔ Parashar, A., Burse, K., & Rawat, K. (2014). A Comparative approach for Pima Indians diabetes diagnosis using lda- support vector machine and feed forward neural network. International Journal of Advanced Research in Computer Science and Software Engineering, 4(11), Pages:378-383.
- ➔ https://www.techopedia.com/definition/30364/support-vector-machine-svm#:~:text=A%20support%20vector%20machine%20(SVM)%20is%20machine%20learning%20algorithm%20that,as%20far%20apart%20as%20possible
- ➔
- ➔ Science & Control Engineering: An Open Access Journal, 2(1),Pages: 602-609.
- ➔ Machine Learning Random Forest Algorithm - Javatpoint
- ➔ Quinlan, J. R. (1986). Induction on decision tree. Mach. Learn.1, Pages:81–106.
- ➔ https://scikit-learn.org/stable/

Dataset link:

https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database