DATA*6300 (01) W24

ANALYSIS of BIG DATA

TAKE – HOME PROJECT REPORT

Submitted by

Namrata Roy (1260835)

Master of Data Science

Department of Mathematics and Statistics

University of Guelph

Data Pre-Processing: The first step was to load the 'diabetic_data.csv' dataset. Then, a statistical and features data type visualization was performed using the info() and the describe().T methods accordingly. After that, histograms were used to examine the numerical features. Next, the '?' values were replaced with 'NaN' for handling missing values. The columns with missing values and their counts were then measured. Then missing values were addressed by using the most frequent diagnosis.

In the next step, the dataset was extracted as the feature 'X' and the target 'y.' A notable mention is a few features like '*weight,' 'encounter_id', 'patient_nbr'* etc. were dropped due to higher missing values or insignificant indicating to ensure the data quality.

Two types of pipelines were generated: a numeric pipeline using StandardScaler() and a categorical pipeline using OneHotEncoder(). Then a pipeline named 'preprocessor' was generated as a bundle of those pipelines.

In the last stage of pre-processing, the target and the feature were split into training and validation sets using the train_test_split() method from model_selection in scikit learn. Here, 'X_train' and 'y_train' were the feature matrix and target variable for the training set, which were used as the machine learning model to train the data. Furthermore, to evaluate the performance of your trained model 'X_val' and 'y_val' were the feature matrix and target variable for the validation set. To ensure the data was split in the same way while running the code multiple times and get a consistent result the 'random_state' parameter value was fixed to 42. Moreover, the 'test_size' value was 0.2, which means 20% of the data was used for validation and 80% for training the model.

Model Selection: In order to predict the outcome, eight machine learning algorithms were utilized. Out of these, five were single machine learning algorithms, and three were ensemble learning algorithms. The five single machine learning algorithms include Logistic Regression (LR), Decision Tree (DT), K-Nearest Neighbour (KNN), Naive Bayes (NB), and Multi-Layer Perceptron (MLP) from Artificial Neural networks (ANN). The three ensemble learning algorithms used were Random Forest (RF), Gradient Boosting (GB), and AdaBoost (AB).

Comparison and Analysis: Eight pipelines were created using the 'preprocessor' pipeline and the classifiers for each machine learning algorithm. The feature matrix and target variable for the training set were fitted with all eight pipelines and the prediction outcome was generated for each of the models. Afterward, the testing accuracy and F1 score were measured for each predicted outcome and compared in different graphs and heatmaps using the matplotlib.pyplot and the seaborn libraries. Table: 1 shows the comparison of the testing accuracy and F1 score of the final prediction. Fig: 1, Fig:2 and Fig: 3 show the testing accuracy and F1 score comparison of all the ML algorithms' predictions in a bar chart, line graph, and heatmap respectively. All the tables and figures show that Gradient Boosting has the highest accuracy of 59%. Then Random Forest and AdaBoost with 58%, Logistic Regression with 57%, K-Nearest Neighbour with 51%, and Decision Tree and Artificial Neural networks with 50%. Naive Bayes has the lowest accuracy of 14%.

Table: 3.1- The comparison of the testing accuracy and F1 score of the final prediction

	Logistic Regression	Decision Tree	K-Nearest Neighbour	Naive Bayes	Artificial Neural networks	Random Forest	Gradient Boosting	AdaBoost
Testing Accuracy	57%	50%	51%	14%	50%	58%	59%	58%
F1 score	0.52	0.49	0.50	0.08	0.50	0.54	0.53	0.54

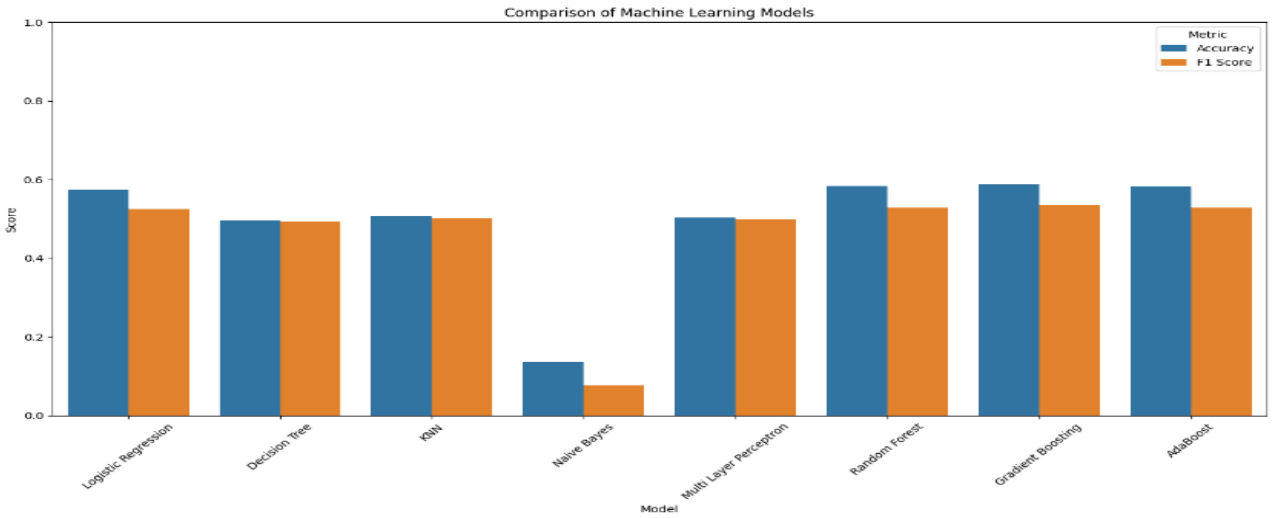

Fig: 1 - The comparison of the testing accuracy and F1 score of the final prediction in bar chart

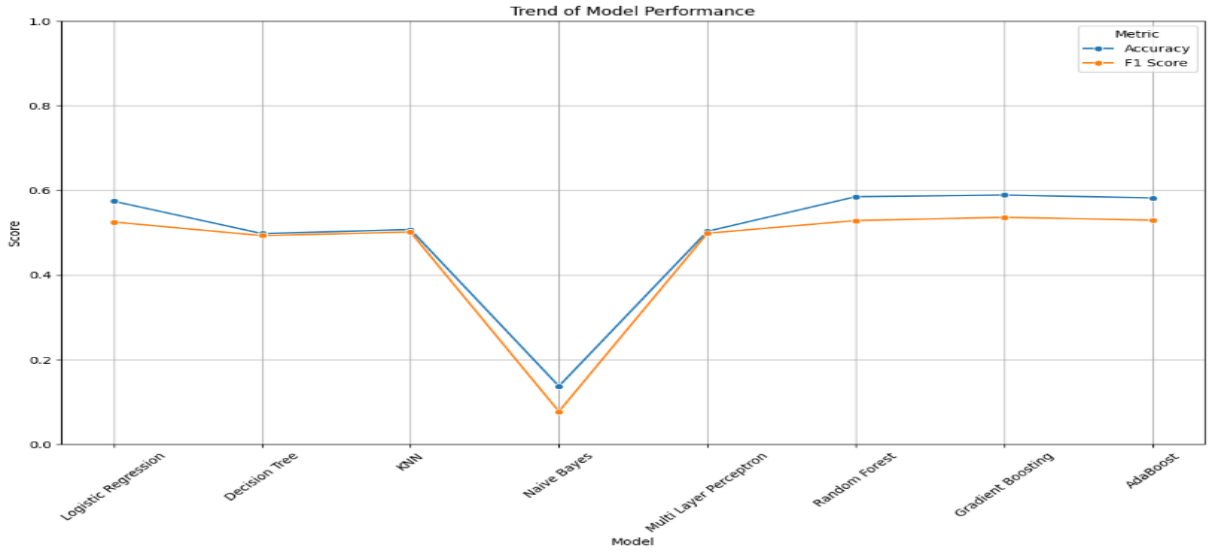

Fig: 2 - The comparison of the testing accuracy and F1 score of the final prediction in a line graph

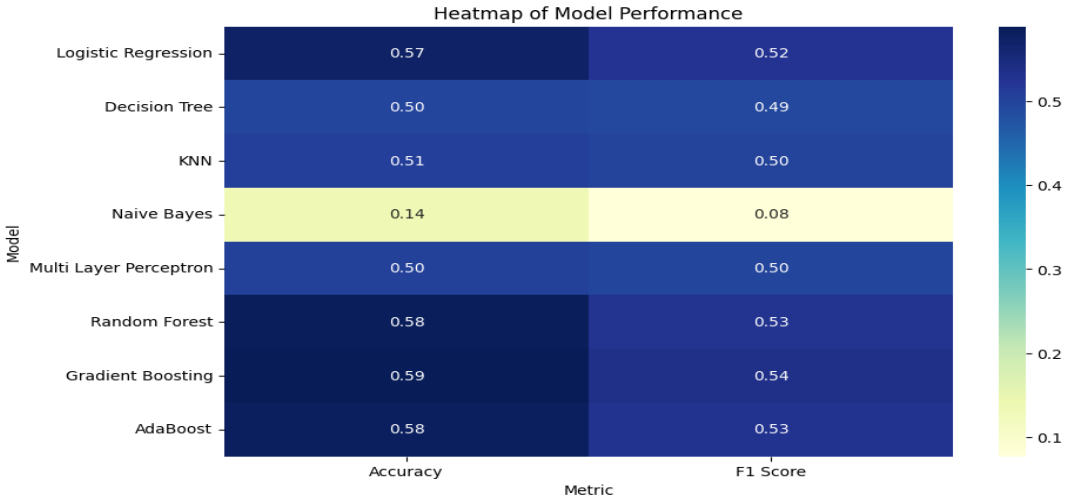

Fig: 3 - The comparison of the testing accuracy and F1 score of the final prediction in a heatmap