**VIETNAM NATIONAL UNIVERSITY HO CHI MINH CITY**

**UNIVERSITY OF INFORMATION TECHNOLOGY**

**FACULTY OF INFORMATION SYSTEM**

**Logo

Description automatically generated**

**FINAL REPORT FOR IS252.N11.TMCL**

*Project****:* FLIGHT DELAY PREDICTION**

***Instructor:*** ***PhD*** Cao Thị Nhạn,

***Master*** Nguyễn Hồ Duy Trí

|  |  |
| --- | --- |
| ***Group member:*** | |
| Bùi Tuấn Kiệt | 20521493 |
| Trần Đình Khôi | 20521482 |
| Nguyễn Thành Lâm | 20521517 |
| Hoàng Đình Hữu | 20521384 |
| Lê Văn Khoa | 20521467 |

**TABLE OF CONTENTS**

**Chapter I: Introduction** 3

1. Topic Introduction 3
2. Data Introduction 3
3. Overview of the data set 4

**Chapter II: Visualization** 5

Graph 1 5

Graph 2 6

Graph 3 6

Graph 4 7

Graph 5 8

Graph 6 8

**Chapter III: Preprocessing** 9

1. Convert datatime 9
2. Drop unnecessary columns 10
3. Prepare for trainning 11
4. “Accounting for destinations” model 11

**Chapter IV: Prediction** 12

1. Bagging Classifier 12
2. Decision Tree 13
3. Random Forest 15
4. AdaBoost Classifier 16
5. Logistics Regression 18
6. Navie Bayes 19
7. K-Nearest Neighbor 21
8. Support Vector Machine 22
9. Nerual Network 24

**Chapter V: Model Evaluation** 25

Graph 7 26

Graph 8 26

**Chapter VI: Conclusion** 27

Table Task 27

Reference 27

**Chapter I: Introduction**

1. **Topic introduction**

Nowadays, air traffic has experienced rapid growth, which makes enormous contributions to economic development. Therefore, it is crucial to limit flight delay because it plays an important role in both profits and loss of the airlines.

Flight delays lead to negative impacts, mainly economical for commuters, airline industries and airport authorities. Furthermore, in the domain of sustainability, it can even cause environmental harm by the rise in fuel consumption and gas emissions. Hence, these factors indicate how necessary and relevant it has become to predict the delays no matter the wide-range of airline meshes.

An accurate estimation of flight delay is critical for airlines because the results can be applied to increase customer satisfaction and incomes of airline agencies. Thus, our team make a decision to reasech into modeling and predicting flight delays throughout extracting important characteristics and most related features.

This report proposes a model for predicting flight delay based on nine distinct Machine Learning algorithms to find out the best prediction.

1. **Data introduction**

This data set is a summary of different air carriers showing their departure delays, arrival delays, scheduled departure, etc.

Data name: The flight delay and cancellation.

Data sources: DOT's Bureau of Transportation Statistics

*Link data*

<https://www.kaggle.com/datasets/usdot/flight-delays?datasetId=810&sortBy=voteCount&select=airlines.csv>

1. **Overview of the dataset**

Each entry of the *flights.csv* file corresponds to a flight and we see that more than 5'800'000 flights have been recorded in 2015. These flights are described according to 31 variables.

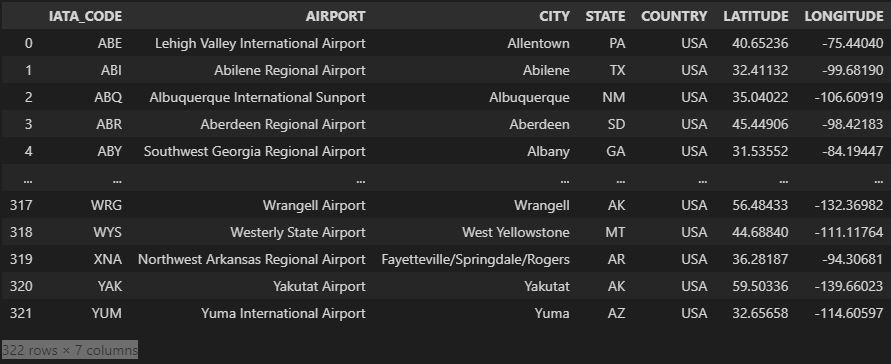
*flights.csv* has 31 features (columns ) and 5819079 rows

Description of the attributes involved in the dataset

* **YEAR, MONTH, DAY, DAY\_OF\_WEEK**: dates of the flight
* **AIRLINE**: An identification number assigned by US DOT to identify a unique airline
* **ORIGIN\_AIRPORT** and **DESTINATION\_AIRPORT**: code attributed by IATA to identify the airports
* **SCHEDULED\_DEPARTURE** and **SCHEDULED\_ARRIVAL** : scheduled times of take-off and landing
* **DEPARTURE\_TIME** and **ARRIVAL\_TIME**: real times at which take-off and landing took place
* **DEPARTURE\_DELAY** and **ARRIVAL\_DELAY**: difference (in minutes) between planned and real times
* **DISTANCE**: distance (in miles)

An additional file of this dataset, the *airports.csv* (322 rows × 7 columns ) file gives a more exhaustive description of the airports and the *airlines.csv* (13 rows x 2 columns) file contains airline name and company abbreviation code.

*airports.csv*

**

A screenshot of a computer screen

Description automatically generated with low confidence*airlines.csv*

**Chapter II: Visualization**

Chart, bar chart

Description automatically generated

**Graph 1**: The graph shows the correlation of delay time between distinct airlines in January

*Comment:* According to this graph, it is obvious that almost airlines companies have a significantly higher on-time flights than delay flights. Furthermore, Southwest Airlines has the highest number of on-time flights while the Virgin America hits the lowest point.

Chart, pie chart

Description automatically generated

**Graph 2**: The pie chart gives the percentage of flights per airline and the mean delay time of each company in January

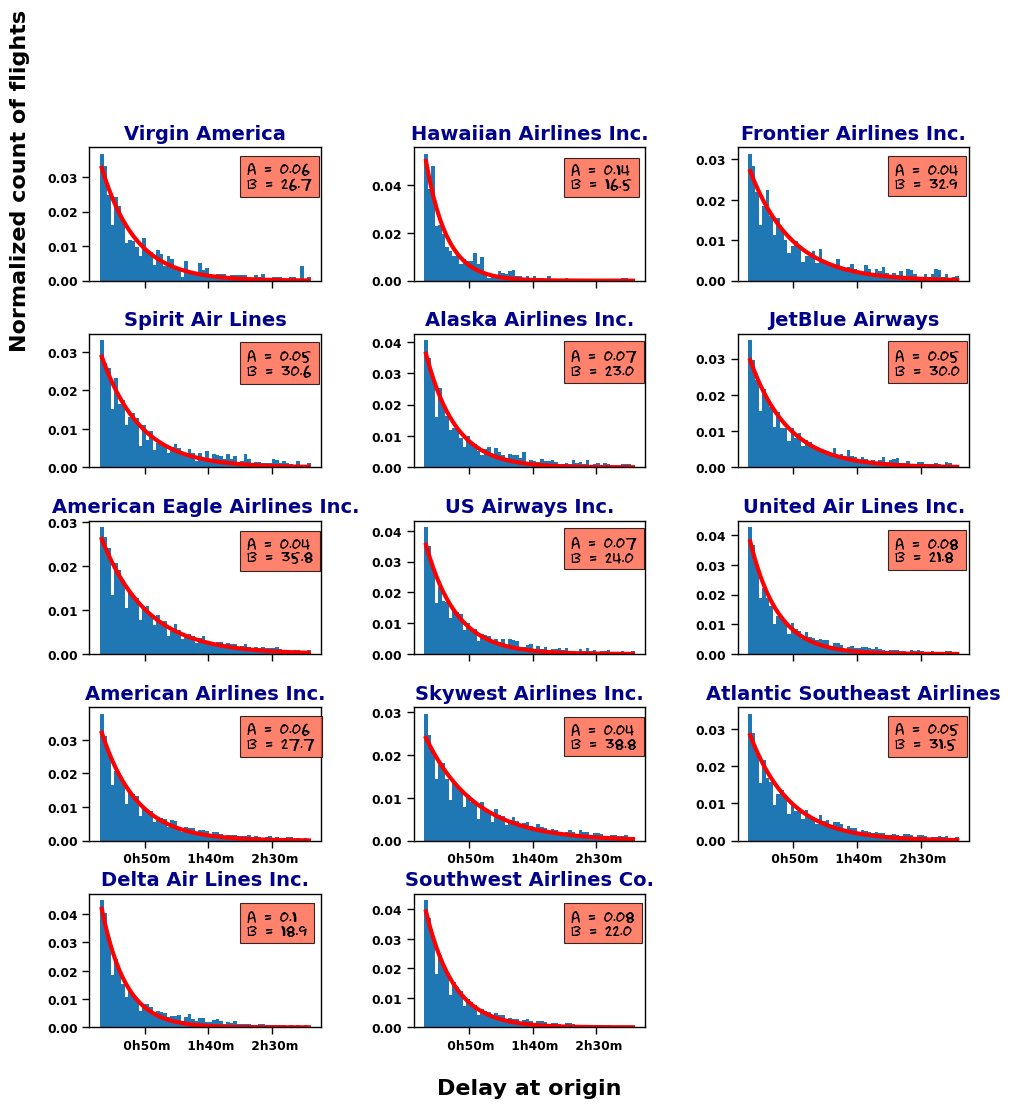
*Comment:* Overall, the Southwest Airlines accounts for approximately one-fifth of the total. Moreover, the average figure for mean delay times oscillates from 6 to 11 minutes while Hawaiian Airlines and Alaska Airlines reported extremely low mean delays.

Chart, pie chart

Description automatically generated

**Graph 3:** The graph shows the depature delay time of each airline in January

*Comment:* The delay time is mostly under 4 hours. However, we see that occasionally, we can face really large delays that can reach a few tens of hours.



**Graph 4:** The graph shows the normalised distribution of delay in January

*Comment:* This figure shows the normalised distribution of delays that we modelised with an exponential distribution . Finally, according to the value of either or , it is possible to establish ranking of the companies: the low values of will correspond to airlines with a large proportion of important delays and, on the contrary, airlines that shine from their punctuality will admit high value. It is described more obviously in the next graph.

Chart

Description automatically generated

**Graph 5:** The graph shows characterizing delays among companies in January

*Comment:* The left panel of this figure gives an overview of the and coefficients of the 14 airlines showing that Hawaiian Airlines and Delta Airlines occupy the first two places. The right panel zooms on 12 other airlines. It can be seen that SouthWest Airlines, which represent ∼ 20% of the total number of flights is well ranked and occupy the third position. According to this ranking, SkyWest Airlines is the worst carrier.

Chart, bar chart

Description automatically generated

**Graph 6:** The graph shows the difference between departure delay and arrival delay in January

*Comment:* On this figure, we can see that delays at arrival are generally lower than at departure. This indicates that airlines adjust their flight speed in order to reduce the delays at arrival.

**Chapter III: Preprocessing**

We just use data in January to speed up the process because the original data is too large, at 5819079 rows × 31 columns

df = df[df['MONTH'] == 1]

Graphical user interface, text

Description automatically generated

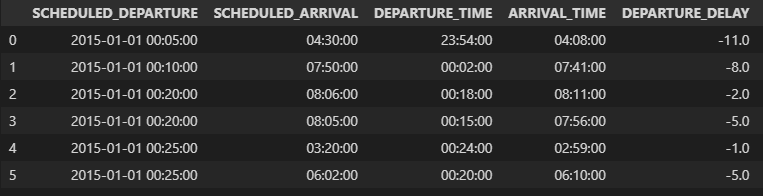
1. **Convert datatime**

Step 1: Convert YEAR, MONTH, DAY into DATE

Text

Description automatically generated

Step 2: Convert time into hour:min:second format



1. **Drop unnecessary columns**

* Features such as 'TAXI\_OUT', 'TAXI\_IN', 'WHEELS\_ON', 'WHEELS\_OFF',… have no contribution to prediction so that drop it.
* Drop 'YEAR', 'MONTH', 'DAY', 'DAY\_OF\_WEEK', 'DATE' because we had converted into hour:min:second format before.

Step 1: Drop features are useless for predictions and have many null values

Graphical user interface, application

Description automatically generated

Step 2: Reorganize data



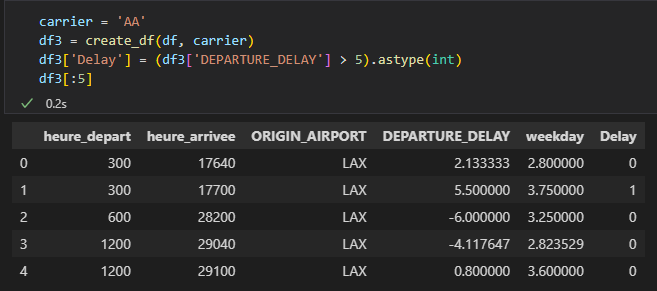
1. **Prepare data for trainning**

We use first 3 weeks of January for data train, the others in January for data test



1. **“Accounting for destinations” model**

* We group flights with different destinations were grouped as soon as they leave at the same time. Now we make a model that accounts for both departure and arrival times.
* We pick up carrier = ‘AA’ to focus on it because almost carriers have similar delay times.
* To prepare for classification, we create decision features (Delay) based on DEPARTURE\_DELAY. If DEPARTURE\_DELAY < 5 mins, the label is 0 (on time). The remaining values are labeled as 1 (delay).



**Chapter IV: Prediciton**

1. **Bagging Classifier**

**1.1 Definition**

A Bagging classifier is an ensemble meta-estimator that fits base classifiers each on random subsets of the original dataset and then aggregate their individual predictions (either by voting or by averaging) to form a final prediction. Such a meta-estimator can typically be used as a way to reduce the variance of a black-box estimator (e.g., a decision tree), by introducing randomization into its construction procedure and then making an ensemble out of it.

**1.2 Advantages**

* The biggest advantage of bagging is that multiple weak learners can work better than a single strong learner.
* It provides stability and increases the machine learning algorithm’s accuracy that is used in statistical classification and regression.
* It helps in reducing variance, i.e. it avoids overfitting.
  1. **Result**

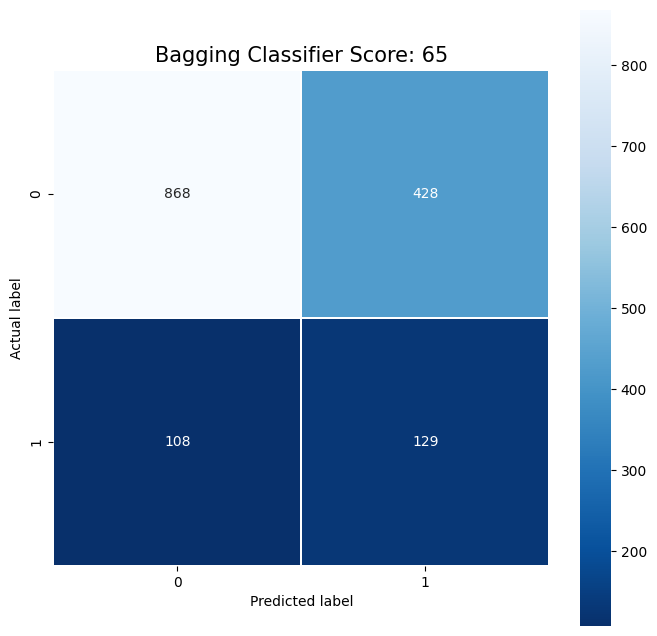
**Execution time:** 0.11828303337097168

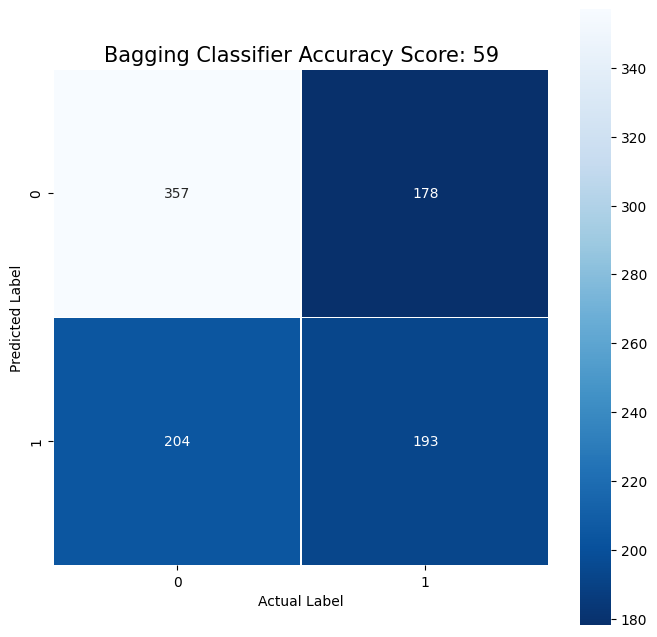
**Test on 30% data train**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.64 | 0.67 | 0.65 | 535 |
| 1 | 0.52 | 0.49 | 0.50 | 397 |
| accuracy |  |  | 0.59 | 932 |
| marco avg | 0.58 | 0.58 | 0.58 | 932 |
| weighted avg | 0.59 | 0.59 | 0.59 | 932 |

**Test on data test**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.89 | 0.67 | 0.76 | 1296 |
| 1 | 0.23 | 0.54 | 0.32 | 237 |
| accuracy |  |  | 0.65 | 1533 |
| marco avg | 0.56 | 0.61 | 0.54 | 1533 |
| weighted avg | 0.79 | 0.65 | 0.70 | 1533 |

**CONFUSION MATRIX**



**On data train** **On data test**

1. **Decision Tree**

**2.1 Definition**

Decision Tree is the most powerful and popular tool for classification and prediction. A Decision tree is a flowchart-like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.

There are two mainly types of this algorithm: ID3 and CART which use different equation to compute the purity of feature.

**2.2 Advantages**

* Simple to understand and to interpret. Trees can be visualized.
* Requires little data preparation.
* The cost of using the tree (i.e., predicting data) is logarithmic in the number of data points used to train the tree.
* Able to handle both numerical and categorical data.
* Able to handle multi-output problems.
* Uses a white box model. If a given situation is observable in a model, the explanation for the condition is easily explained by boolean logic
* Performs well even if its assumptions are somewhat violated by the true model from which the data were generated.
  1. **Result**

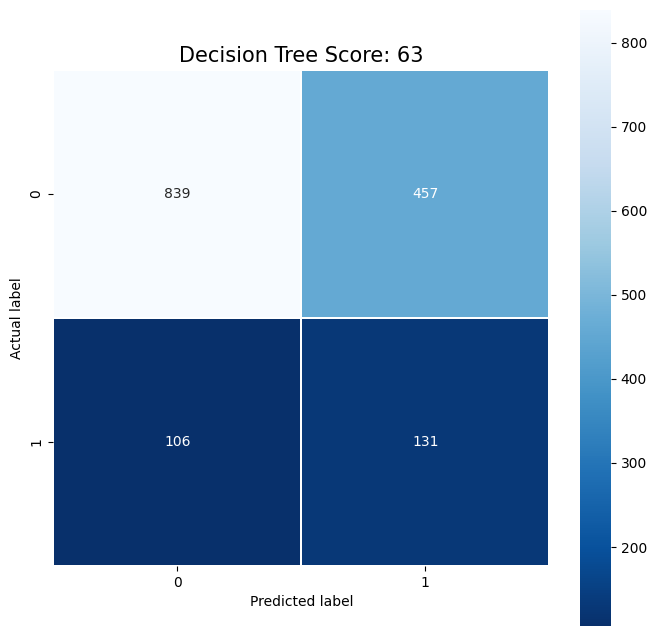
**Execution time:** 0.02712106704711914

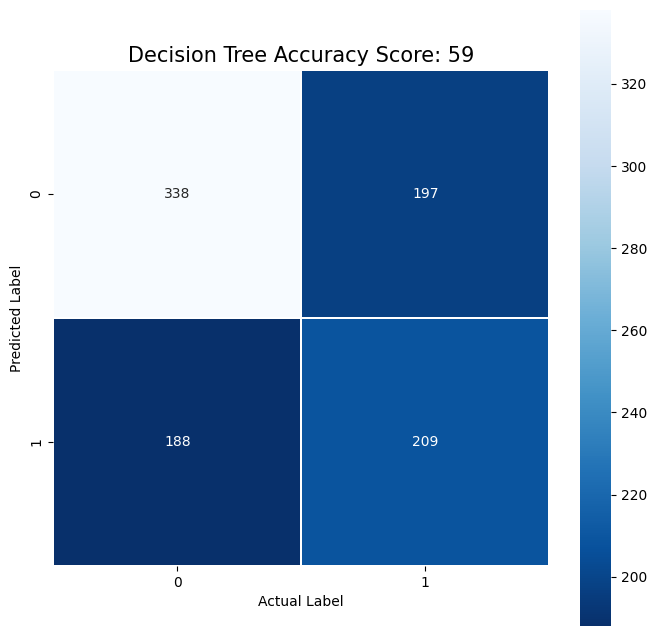
**Test on 30% data train**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.64 | 0.63 | 0.64 | 535 |
| 1 | 0.51 | 0.53 | 0.52 | 397 |
| accuracy |  |  | 0.59 | 932 |
| marco avg | 0.58 | 0.58 | 0.58 | 932 |
| weighted avg | 0.59 | 0.59 | 0.59 | 932 |

**Test on data test**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.89 | 0.65 | 0.75 | 1296 |
| 1 | 0.22 | 0.55 | 0.32 | 237 |
| accuracy |  |  | 0.63 | 1533 |
| marco avg | 0.56 | 0.60 | 0.53 | 1533 |
| weighted avg | 0.79 | 0.63 | 0.68 | 1533 |

**CONFUSION MATRIX**



**On data train** **On data test**

1. **Random Forest**

**3.1 Definition**

Random forest is a supervised learning algorithm and it is wildly used in classification and regression problems. A forest consists of a large number of trees. Similarly, a random forest involves processing many decision trees. Each tree predicts a value for the probability of target variables. We then average the probabilities to produce the final output.

**3.2 Advantages**

* It can perform both regression and classification tasks.
* It is efficient when it comes to large datasets.
* It allows estimating the significance of input variables in classification.
* The random forest algorithm provides a higher level of accuracy in predicting outcomes over the decision tree algorithm.
  1. **Result**

**Execution time:** 0.45551419258117676

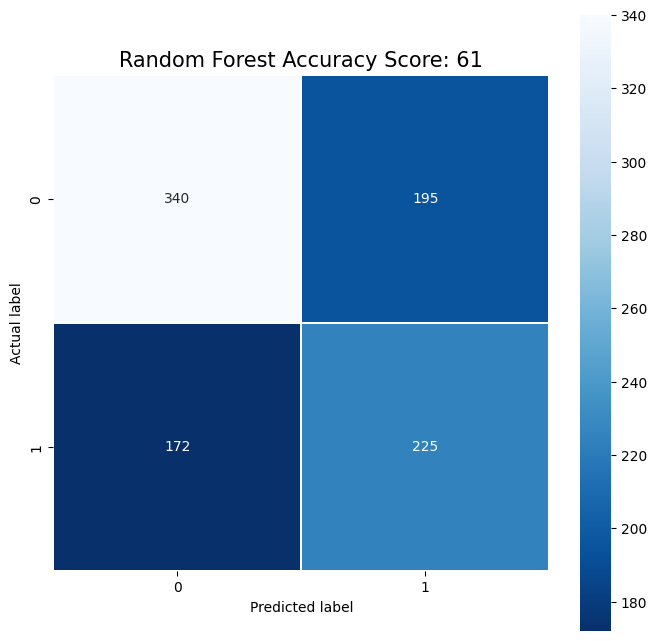
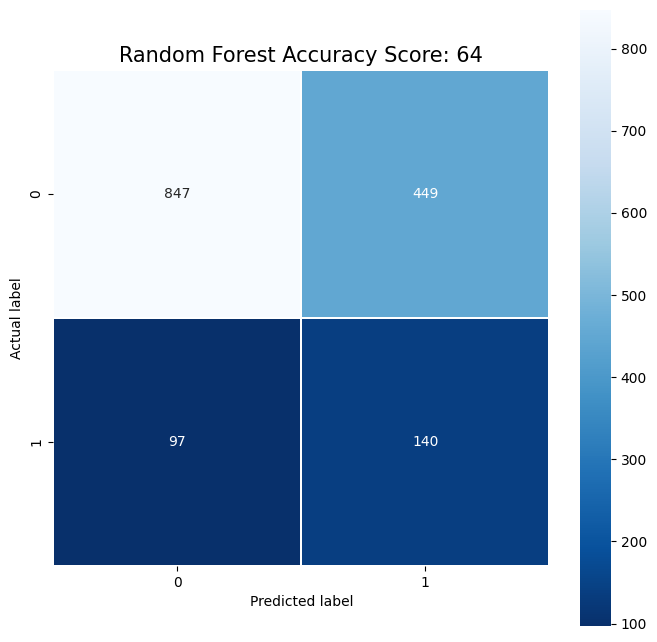
**Test on 30% data train**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.66 | 0.64 | 0.65 | 535 |
| 1 | 0.54 | 0.57 | 0.55 | 397 |
| accuracy |  |  | 0.61 | 932 |
| marco avg | 0.60 | 0.60 | 0.60 | 932 |
| weighted avg | 0.61 | 0.61 | 0.61 | 932 |

**Test on data test**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.90 | 0.65 | 0.76 | 1296 |
| 1 | 0.23 | 0.59 | 0.34 | 237 |
| accuracy |  |  | 0.64 | 1533 |
| marco avg | 0.57 | 0.62 | 0.55 | 1533 |
| weighted avg | 0.80 | 0.64 | 0.69 | 1533 |

**CONFUSION MATRIX**



**On data train** **On data test**

1. **AdaBoost Classifier**

**4.1 Definition**

AdaBoost is an ensemble learning method (also known as “meta-learning”) which was initially created to increase the efficiency of binary classifiers. AdaBoost uses an iterative approach to learn from the mistakes of weak classifiers, and turn them into strong ones. AdaBoost classifier builds a strong classifier by combining multiple poorly performing classifiers.

**4.2 Advantages**

* AdaBoost Classifier is easy to implement.
* It iteratively corrects the mistakes of the weak classifier and improves accuracy by combining weak learners.
* We can use many base classifiers with AdaBoost.
* AdaBoost is not prone to overfitting.
  1. **Result**

**Execution time:** 0.19656872749328613

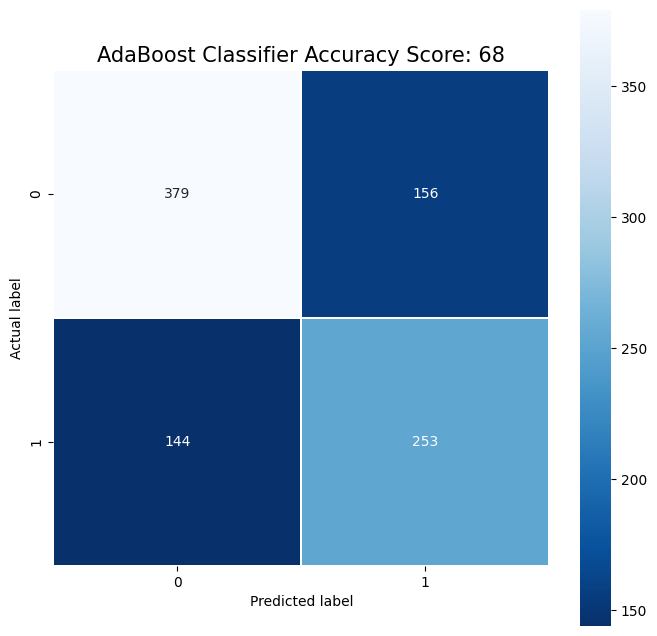
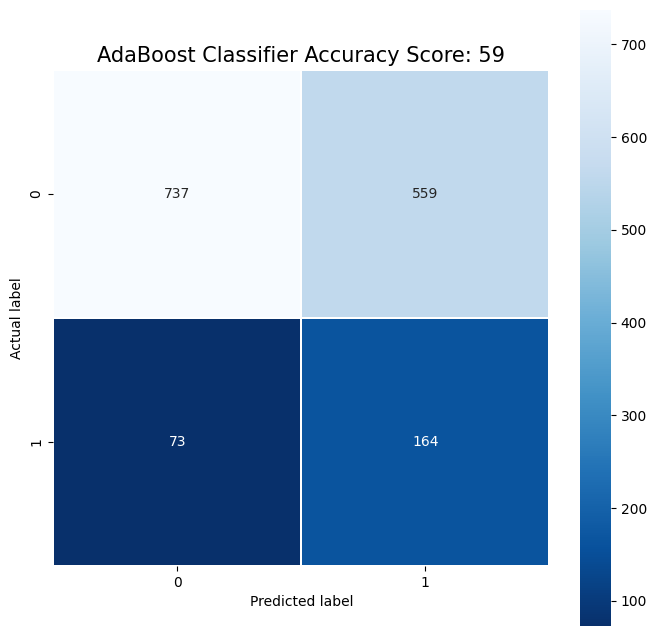
**Test on 30% data train**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.72 | 0.71 | 0.72 | 535 |
| 1 | 0.62 | 0.64 | 0.63 | 397 |
| accuracy |  |  | 0.68 | 932 |
| marco avg | 0.67 | 0.67 | 0.67 | 932 |
| weighted avg | 0.68 | 0.68 | 0.68 | 932 |

**Test on data test**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.91 | 0.57 | 0.70 | 1296 |
| 1 | 0.23 | 0.69 | 0.34 | 237 |
| accuracy |  |  | 0.59 | 1533 |
| marco avg | 0.57 | 0.63 | 0.52 | 1533 |
| weighted avg | 0.80 | 0.59 | 0.64 | 1533 |

**CONFUSION MATRIX**



**On data train** **On data test**

1. **Logistics Regression**

**5.1 Definition**

Logistic regression is a statistical analysis method to predict a binary outcome, such as yes or no, based on prior observations of a data set. A logistic regression model predicts a dependent data variable by analyzing the relationship between one or more existing independent variables.

**5.2 Advantages**

* Is easier to implement, interpret, and very efficient to train.
* It makes no assumptions about distributions of classes in feature space.
* It not only provides a measure of how appropriate a predictor (coefficient size) is, but also its direction of association (positive or negative).
* It is very fast at classifying unknown records.
* Good accuracy for many simple data sets and it performs well when the dataset is linearly separable.
  1. **Result**

**Execution time:** 0.03815150260925293

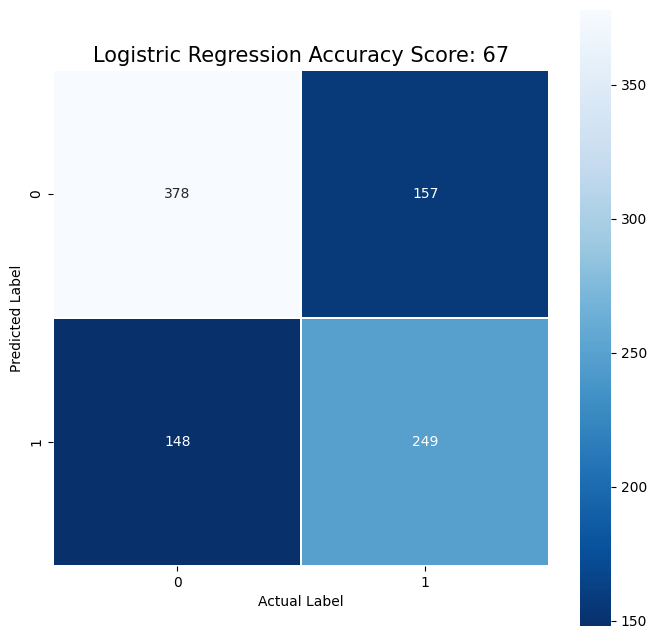
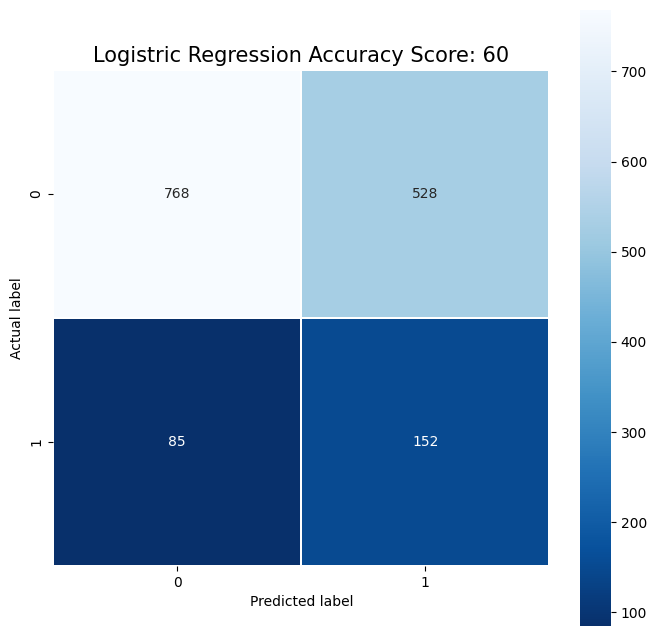
**Test on 30% data train**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.72 | 0.71 | 0.71 | 535 |
| 1 | 0.61 | 0.63 | 0.62 | 397 |
| accuracy |  |  | 0.67 | 932 |
| marco avg | 0.67 | 0.67 | 0.67 | 932 |
| weighted avg | 0.67 | 0.67 | 0.67 | 932 |

**Test on data test**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.90 | 0.59 | 0.72 | 1296 |
| 1 | 0.22 | 0.64 | 0.33 | 237 |
| accuracy |  |  | 0.60 | 1533 |
| marco avg | 0.56 | 0.62 | 0.52 | 1533 |
| weighted avg | 0.80 | 0.60 | 0.66 | 1533 |

**CONFUSION MATRIX**



**On data train** **On data test**

**6. Naïve Bayes**

**6.1 Definition**

The Naïve Bayes Method is a statistical classification that can be used to predict the probability of membership in a class. Naïve Bayes is based on the Bayes theorem that has similar classification capabilities to the Decision Tree and Neural Network. Naïve Bayes proved to have high accuracy and speed when applied to databases with large data.

Bayes theorem formula:

X: Data with unknown class

H: The hypothesis of X data is a class-specific

* 1. **Advantages**
* It is simple and easy to implement.
* It doesn't require as much training data.
* It handles both continuous and discrete data.
* It is highly scalable with the number of predictors and data points.
* It is fast and can be used to make real-time predictions.
  1. **Result**

**Execution time:** 0.009204626083374023

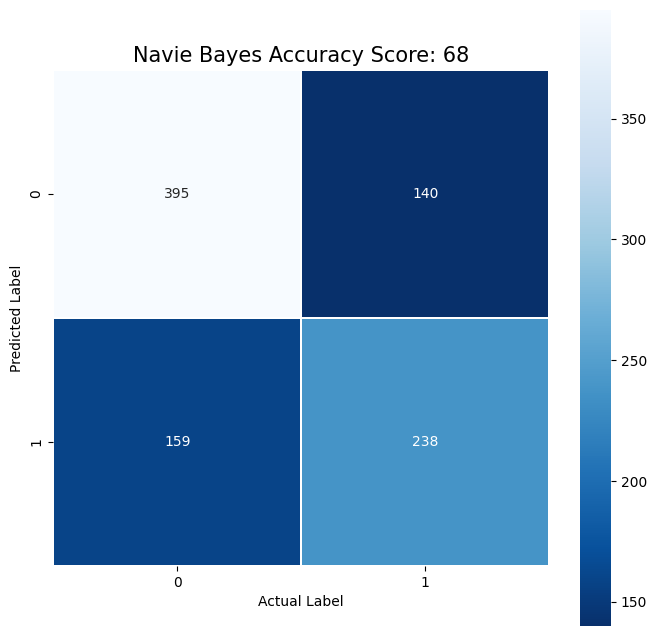
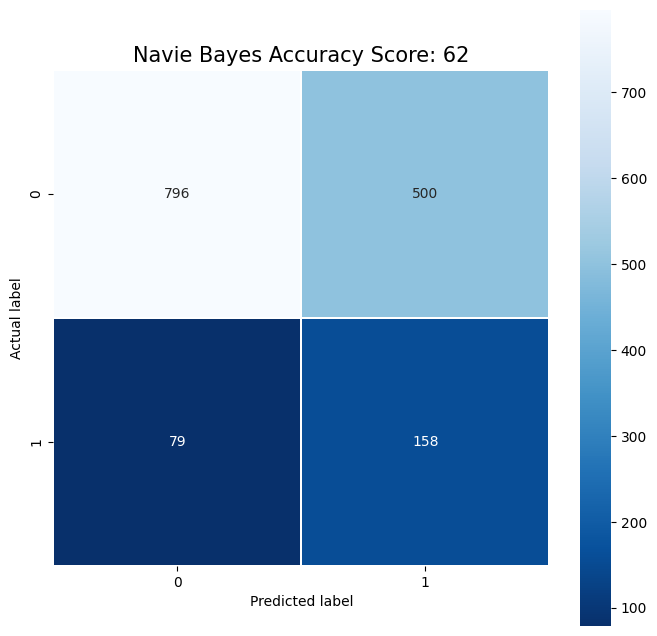
**Test on 30% data train**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.71 | 0.74 | 0.73 | 535 |
| 1 | 0.63 | 0.60 | 0.61 | 397 |
| accuracy |  |  | 0.68 | 932 |
| marco avg | 0.67 | 0.67 | 0.67 | 932 |
| weighted avg | 0.67 | 0.68 | 0.68 | 932 |

**Test on data test**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.91 | 0.61 | 0.73 | 1296 |
| 1 | 0.24 | 0.67 | 0.35 | 237 |
| accuracy |  |  | 0.62 | 1533 |
| marco avg | 0.57 | 0.64 | 0.54 | 1533 |
| weighted avg | 0.81 | 0.62 | 0.67 | 1533 |

**CONFUSION MATRIX**



**On data train** **On data test**

**7. K-Nearest Neighbor**

**7.1 Definition**

The K-Nearest Neighbor Method is a proximity search method between a new case and an old case based on a predetermined amount. Suppose necessary to find a new patient by using the solution from a patient. To find the case of which patient to use, then the proximity of the new patient’s case to the patient’s old patient was calculated. The case of the old patient with the greatest proximity would be considered a new patient problem.

**7.2 Advantages**

* Simple to implement and intuitive to understand.
* Can learn non-linear decision boundaries when used for classfication and regression.
* No Training Time for classification/regression.
* Constantly evolves with new data.
* Choice of distance metric.
  1. **Result**

**Execution time:** 0.004407644271850586

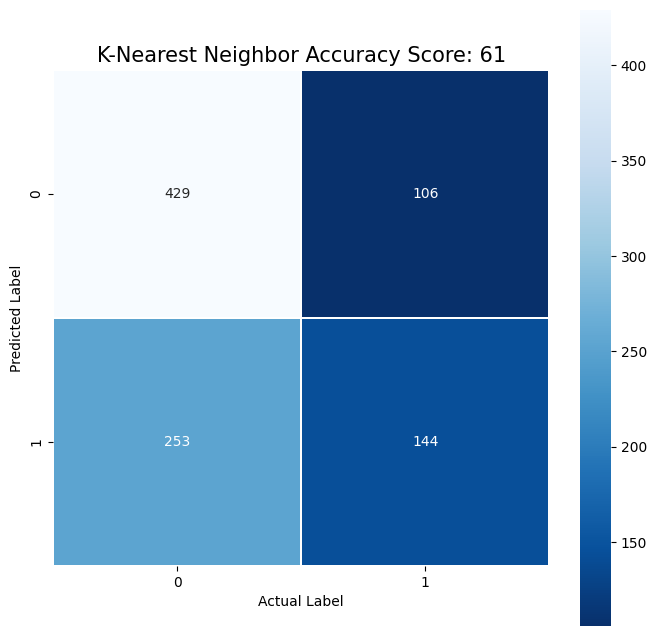
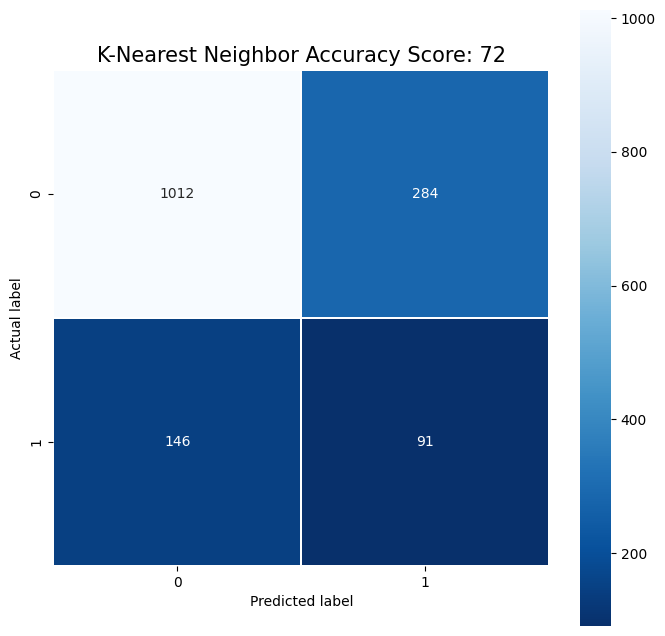
**Test on 30% data train**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.63 | 0.80 | 0.71 | 535 |
| 1 | 0.58 | 0.36 | 0.45 | 397 |
| accuracy |  |  | 0.61 | 932 |
| marco avg | 0.60 | 0.58 | 0.58 | 932 |
| weighted avg | 0.61 | 0.61 | 0.59 | 932 |

**Test on data test**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.87 | 0.78 | 0.82 | 1296 |
| 1 | 0.24 | 0.38 | 0.30 | 237 |
| accuracy |  |  | 0.72 | 1533 |
| marco avg | 0.56 | 0.58 | 0.56 | 1533 |
| weighted avg | 0.78 | 0.72 | 0.74 | 1533 |

**CONFUSION MATRIX**



**On data train** **On data test**

**8. Support Vector Machine**

**8.1 Definition**

A support vector machine (SVM) is machine learning algorithm that analyzes data for classification and regression analysis. SVM is a supervised learning method that looks at data and sorts it into one of two categories. An SVM outputs a map of the sorted data with the margins between the two as far apart as possible. SVMs are used in text categorization, image classification, handwriting recognition and in the sciences.

A support vector machine is also known as a support vector network (SVN).

**8.2 Advantages**

* Support vector machine works comparably well when there is an understandable margin of dissociation between classes.
* It is more productive in high dimensional spaces.
* It is effective in instances where the number of dimensions is larger than the number of specimens.
* Support vector machine is comparably memory systematic.
  1. **Result**

**Execution time:** 0.3919672966003418

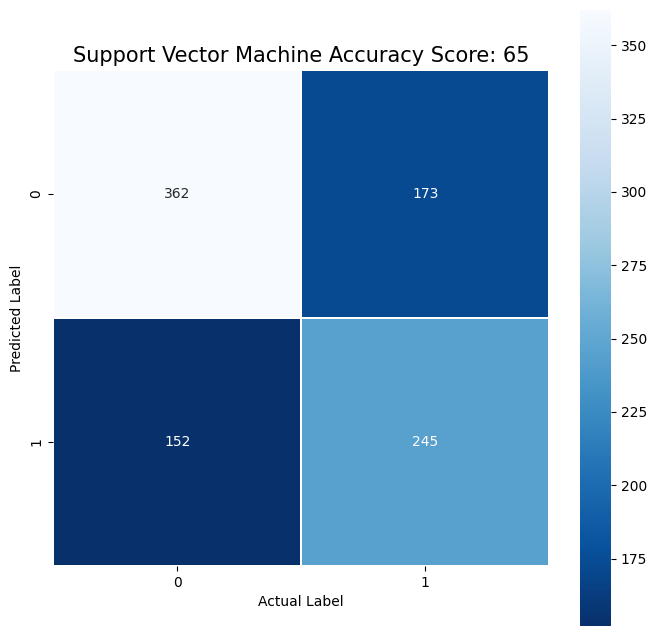
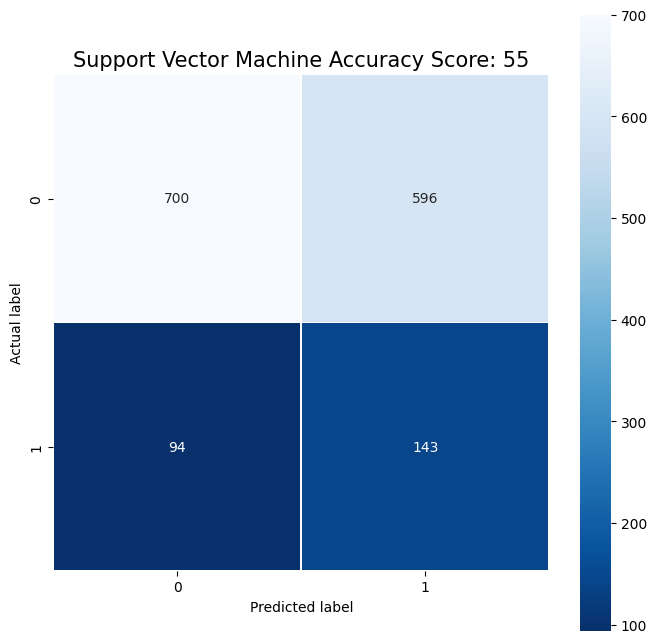
**Test on 30% data train**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.70 | 0.68 | 0.69 | 535 |
| 1 | 0.59 | 0.62 | 0.60 | 397 |
| accuracy |  |  | 0.65 | 932 |
| marco avg | 0.65 | 0.65 | 0.65 | 932 |
| weighted avg | 0.65 | 0.65 | 0.65 | 932 |

**Test on data test**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.88 | 0.54 | 0.67 | 1296 |
| 1 | 0.19 | 0.60 | 0.39 | 237 |
| accuracy |  |  | 0.55 | 1533 |
| marco avg | 0.54 | 0.57 | 0.48 | 1533 |
| weighted avg | 0.78 | 0.55 | 0.61 | 1533 |

**CONFUSION MATRIX**



**On data train** **On data test**

**9. Nerual Network**

**9.1 Definition**

Neural networks, also known as artificial neural networks (ANNs) or simulated neural networks (SNNs), are a subset of machine learning and are at the heart of deep learning algorithms. Their name and structure are inspired by the human brain, mimicking the way that biological neurons signal to one another.

Artificial neural networks (ANNs) are comprised of a node layers, containing an input layer, one or more hidden layers, and an output layer. Each node, or artificial neuron, connects to another and has an associated weight and threshold. If the output of any individual node is above the specified threshold value, that node is activated, sending data to the next layer of the network. Otherwise, no data is passed along to the next layer of the network.

**9.2 Advantages**

* Efficiency
* Continuous learning
* Data retrieval
* Multitasking is one of the common advantages of Neural Networks
* Wide Applications
  1. **Result**

**Execution time:** 4.34818172454834

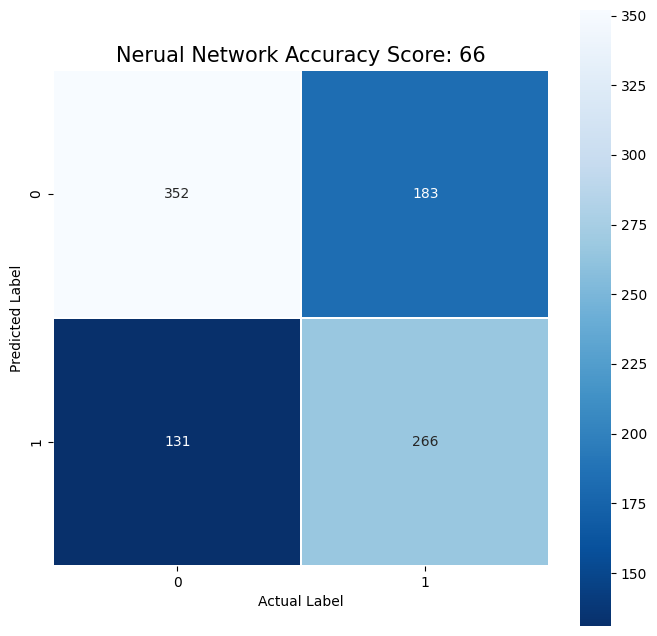
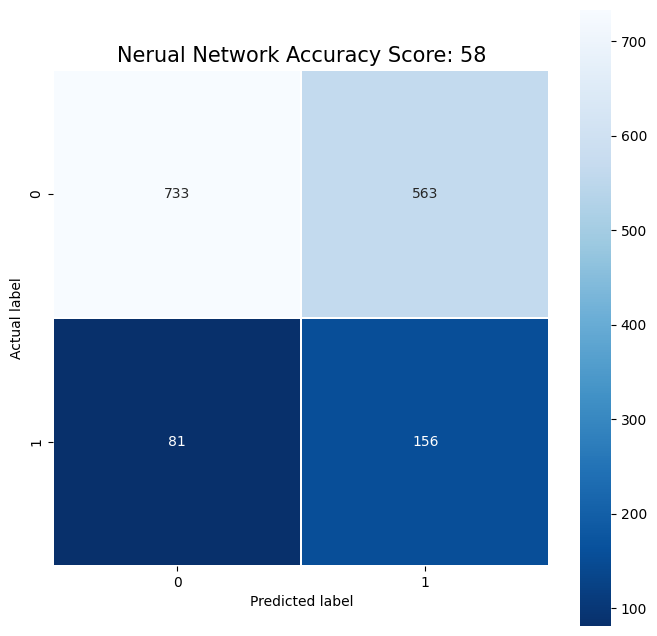
**Test on 30% data train**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.73 | 0.66 | 0.69 | 535 |
| 1 | 0.59 | 0.67 | 0.63 | 397 |
| accuracy |  |  | 0.66 | 932 |
| marco avg | 0.66 | 0.66 | 0.66 | 932 |
| weighted avg | 0.67 | 0.66 | 0.66 | 932 |

**Test on data test**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1 | support |
| 0 | 0.90 | 0.57 | 0.69 | 1296 |
| 1 | 0.22 | 0.66 | 0.33 | 237 |
| accuracy |  |  | 0.58 | 1533 |
| marco avg | 0.56 | 0.61 | 0.51 | 1533 |
| weighted avg | 0.79 | 0.58 | 0.64 | 1533 |

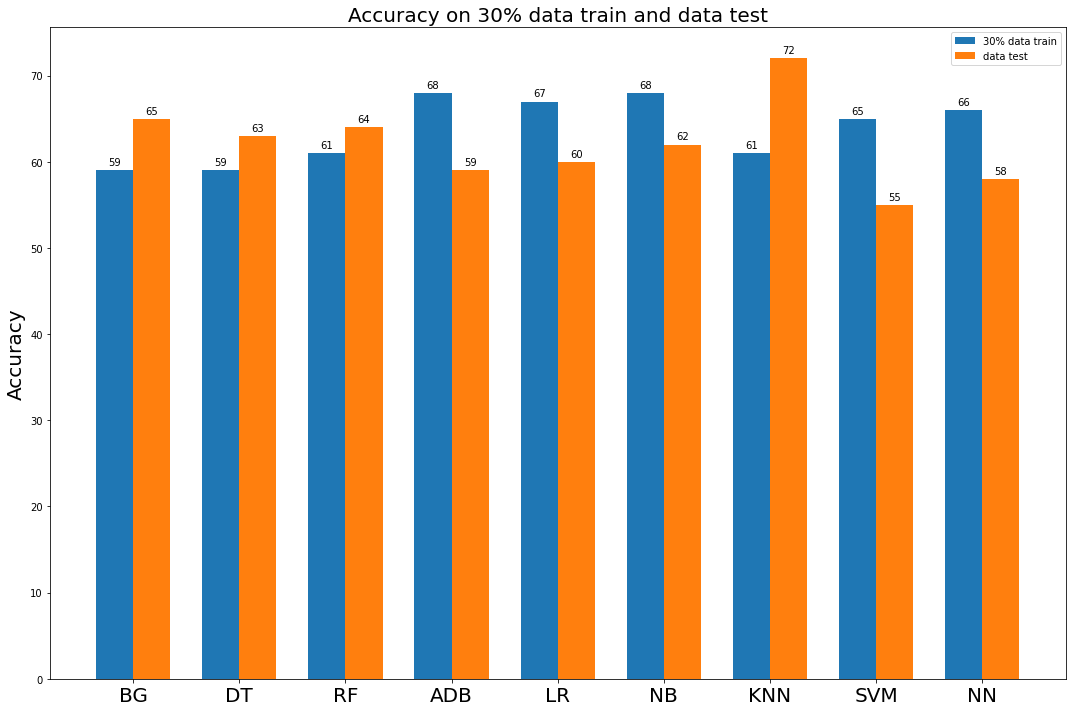
**CONFUSION MATRIX**



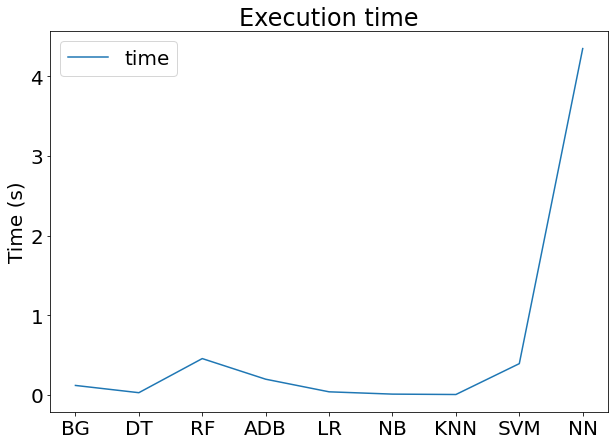
**On data train** **On data test**

**Chapter V: Model Evaluation**

Overall, the results of the algorithms for the model are not very high, in the range of 60-70%. The main reason for this is that this dataset lacks important features that affect flight such as humidity, wind direction, wind speed, wind gust, pressure, temperature, pressure,… Although this data have feautres “WEATHER\_DELAY”, it has a lot of null values. Therefore, if only based on arrival, departure is not enough information to build a model with high accuracy. The solution to improve model peformance is to collect more data that directly affects flight delay as well as reduce null values.



**Graph7:** The accuracy of nine algorithms on 30% data train and data test



**Graph 8:** The execution time of nine algorithms

According to two graph below, it can be seen that K-Nearest Neighbor has the highest accuracy with the quick execution time. After looking at the performance of all my models, I have decided to select K-Nearest Neighbor as best model.

**Chapter VI: Conclusion**

Through this project, we created a machine learning model that can predict the flight departure delays. The best model was K-Nearest Neighbor model. The model was able to catch 72% of the departure delays. Also, it was observed that the delay of the flights was heavily dependent on the departure airports. This clearly implies that if an airport is busy and is a major airport, the chances of flight delays will be more compared to the other airports. By this analysis, it can be made sure that the schedules are better managed, and the functioning of that airport can be improved to avoid such delays. I believe that flights are the fastest way to reach a place and its efficiency matters a lot. Hence, to improve the business for different flights as well as the airports it is essential to minimize these delays. This model will help us evaluate the places we need to work on to make the overall system more efficient.

**Table Task**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Tuấn Kiệt | Thành Lâm | Đình Khôi | Đình Hữu | Văn Khoa |
| Visualization | ✓ | ✓ | ✓ | ✓ | ✓ |
| Preprocessing | ✓ | ✓ | ✓ | ✓ | ✓ |
| Bagging Classifier |  |  |  |  | ✓ |
| Decision Tree |  |  |  |  | ✓ |
| Random Forest |  |  |  | ✓ |  |
| AdaBoost Classifier |  |  |  | ✓ |  |
| Logistics Regression | ✓ |  |  |  |  |
| Navie Bayes |  |  | ✓ |  |  |
| K-Nearest Neighbor |  |  | ✓ |  |  |
| Support Vector Machine |  | ✓ |  |  |  |
| Nerual Network |  | ✓ |  |  |  |
| Report | ✓ |  |  |  |  |

**Reference**

<https://www.kaggle.com/code/abhishek211119/2015-flight-delays-and-cancellation-prediction>