\

Hotel Cancellation Prediction

Karim AbouDaoud
900212779
Department of Mathematics and Actuarial Science
The American University in Cairo
Cairo, Egypt
karimaboudaoud@aucegypt.edu

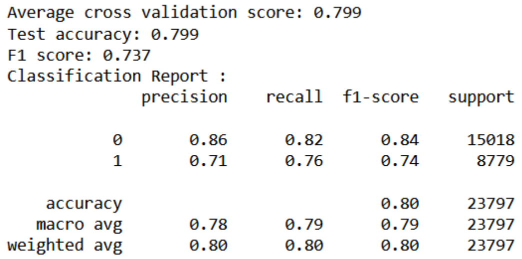
Youssef Nakhla
900201430
Department of Mathematics and Actuarial Science
The American University in Cairo
Cairo, Egypt
youssef_n9212@aucegypt.edu

Experimental Analysis of the Machine Learning Models

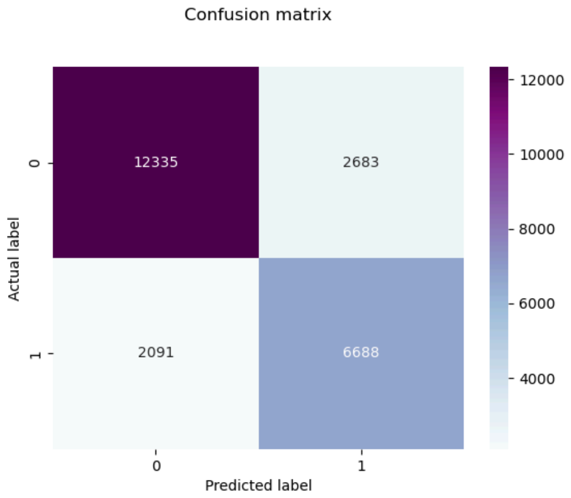
The objective of this project is to forecast hotel booking cancellations. Initially, to determine the suitable machine learning model for implementation, it is crucial to assess the nature of the label. Given that the dataset for constructing the machine learning model comprises a binary label, the task evolves into a binary classification problem. Consequently, the subsequent models have been selected for predicting hotel booking cancellations: Logistic Regression, K-Nearest Neighbours, Naive Bayes Classifier, Multiple Layer perception Learning (Artificial Neural Network), Decision Trees (ID3, CART), and Random Forests. All these models will undergo development, and comparison, and the top 3 performing models will be chosen for further parameter optimization. The development of all models utilized the scikit-learn machine learning package in Python. The dataset was partitioned into training and testing sets before model implementation. The training set comprised 80% of the dataset, while the remaining 20% was allocated for testing purposes.

1. Logistic Regression

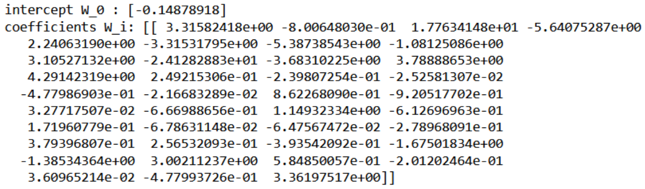
Logistic Regression is generally used to predict a dependent categorical variable. Since our label is a binary variable, logistic regression can be used to predict the likelihood of a hotel booking cancellation. The first step before fitting the model the feature variables were standardized using a standard-scaler, as they are in different scales, to decrease the variability in the dataset, and the features “reservation_status_date” and “arrival_date” were dropped as they were strings unreadable to the models. Moreover, during the learning process, a 5-fold cross-validation was implemented to mitigate the effects of overfitting and to ensure the model’s generalization ability on unseen data. The results of the model show a 79.9% model accuracy where the portion of actual cancellations (TNR, True Negative Rate) that were correctly predicted is the most vital metric in evaluating a model. Furthermore, the TNR of the model is 0.7618179747123818 ≈ 76%, which is relatively high. Upon examining the confusion matrix, it becomes evident that the model was able to correctly predict 82% of the label ‘0’ and 76% of the label ‘1’. The misclassification error of this model is around 100% - 79.9% = 20.1 %.



```
Average cross validation score: 0.799
Test accuracy: 0.799
F1 score: 0.737
Classification Report :
               precision    recall  f1-score   support

           0       0.86      0.82      0.84     15018
           1       0.71      0.76      0.74      8779

    accuracy                           0.80     23797
   macro avg       0.78      0.79      0.79     23797
weighted avg       0.80      0.80      0.80     23797
```



Below are the optimal set of weights, denoted as W, determined by the model which were estimated to maximize the likelihood of the labels for all observations given the feature variables.

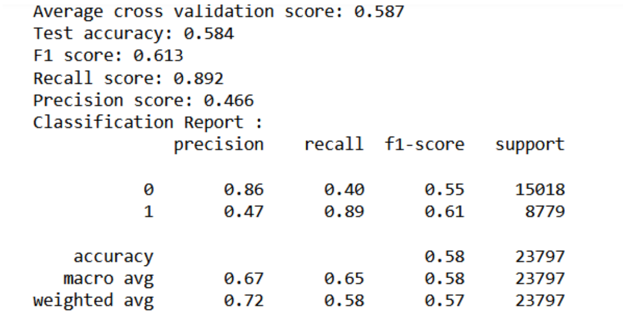


```
intercept W_0 : [-0.14878918]
coefficients W_i: [[ 3.31582418e+00 -8.00648030e-01  1.77634148e+01 -5.64075287e+00
   2.24063190e+00 -3.31531795e+00 -5.38738543e+00 -1.08125086e+00
   3.10527132e+00 -2.41282883e+01 -3.68310225e+00  3.78888653e+00
   4.29142319e+00  2.49215306e-01 -2.39807254e-01 -2.52581307e-02
  -4.77986903e-01 -2.16683289e-02  8.62268090e-01 -9.20517702e-01
   3.27717507e-02 -6.66988656e-01  1.14932334e+00 -6.12696963e-01
   1.71960779e-01 -6.78631148e-02 -6.47567472e-02 -2.78968091e-01
   3.79396807e-01  2.56532093e-01 -3.93542092e-01 -1.67501834e+00
  -1.38534364e+00  3.00211237e+00  5.84850057e-01 -2.01202464e-01
   3.60965214e-02 -4.77993726e-01  3.36197517e+00]]
```

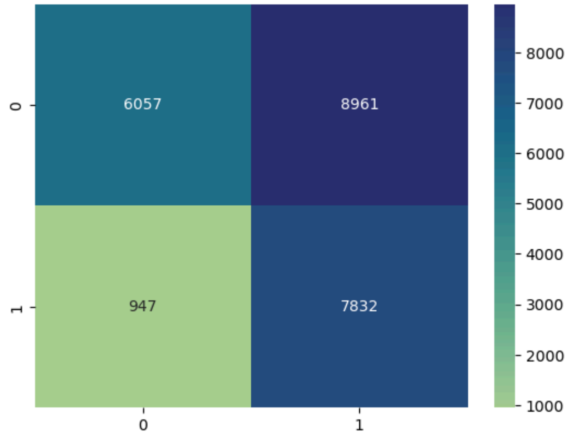
The provided intercept and coefficients offer valuable insights into the factors influencing hotel booking cancellations as determined by the model. The intercept, represented by -0.14878918, indicates the baseline prediction when all other features are zero. Positive coefficients suggest a positive correlation with the likelihood of a booking being canceled, while negative coefficients indicate a negative correlation. For instance, features such as 'previous_cancellations', 'booking_changes', and 'total_stays' exhibit positive coefficients, implying that an increase in these variables is associated with a higher probability of booking cancellations. Conversely, features like 'total_of_special_requests', 'No Deposit' (deposit type), and certain market segment categories like 'Transient' and 'Online TA' demonstrate negative coefficients, suggesting that these factors are associated with lower cancellation probabilities. Understanding these coefficients can assist hotel management in strategizing their booking policies and services to minimize cancellations and enhance overall guest satisfaction.

2. Naive Bayes

Like the logistic regression, Naive Bayes works well with binary labels as well as the fact that it provides a probability score for each class of the label. This is why a Naive Bayes classifier was trained in order to calculate the probability of a cancellation given the input features and predicted the class with the highest probability. Different performance metrics were calculated in order to assess the performance of this classifier.



```
Average cross validation score: 0.587
Test accuracy: 0.584
F1 score: 0.613
Recall score: 0.892
Precision score: 0.466
Classification Report :
                   precision    recall  f1-score   support

              0       0.86      0.40      0.55     15018
              1       0.47      0.89      0.61      8779

       accuracy                           0.58     23797
      macro avg       0.67      0.65      0.58     23797
   weighted avg       0.72      0.58      0.57     23797
```



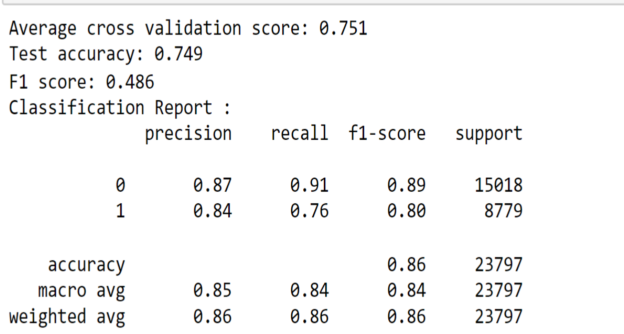
The performance metrics obtained from the classification model suggest a somewhat mixed outcome. The average cross-validation score of 58.7% indicates a moderate level of accuracy across different folds of the data during training. However, when applied to the test data, the model's accuracy slightly drops to 58.4%, suggesting that it may struggle to generalize well to unseen instances. The F1 score, which combines precision and recall, is 61.3%, indicating a decent balance between identifying positive instances (1) and minimizing false positives and false negatives. The recall score of 89.2% suggests that the model is effective at capturing a high proportion of actual positive cases, while the precision score of 46.6% indicates that it may also label a significant number of negative cases as positive. This imbalance between precision and recall is reflected in the classification report, which shows higher precision for class 0 but higher recall for class 1. The TNR was calculated to be 0.7621958121109225≈ 76%. Overall, while the model shows promise in correctly identifying positive instances, there is room for improvement in achieving a better balance between precision and recall, particularly for class 1, and in enhancing generalization to new data.

3. Decision Trees

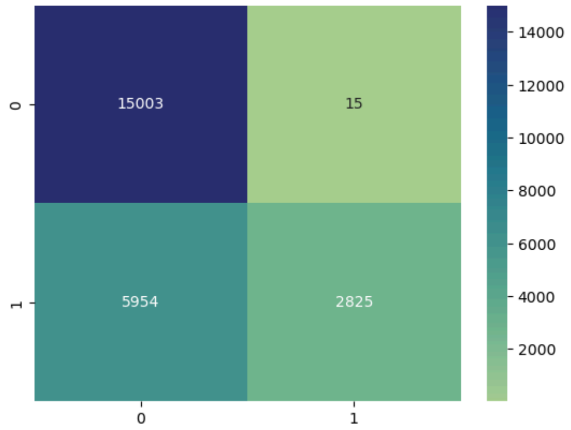
Decision trees stand as a robust tool within the realm of machine learning, applicable across various classification endeavors. In a decision tree, every leaf node signifies a class label or a predicted value, while each internal node represents a decision rooted in a specific feature. The construction of the tree involves iterative partitioning the data into subgroups based on the most advantageous feature, typically determined by maximizing Information Gain. This process allows decision trees to automatically pinpoint the most influential features, thereby bolstering the model's efficacy and overall accuracy. Different performance metrics were calculated in order to assess the performance of this classifier.

ID3:

ID3 is one of the methods used to construct decision trees. ID3 constructs the decision tree in a greedy, top-down manner. It chooses the feature that divides the data into the most distinct classes at each stage. The "purity" of the resultant subsets is determined through the computation of entropy or information gain. ID3 is made to work with attributes that fall into categories. Prior to being given into the algorithm, continuous characteristics must be discretized. Using particular heuristics or binning techniques, this discretization can be accomplished.



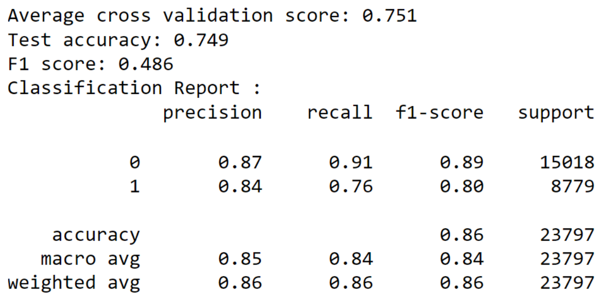
```
Average cross validation score: 0.751
Test accuracy: 0.749
F1 score: 0.486
Classification Report :
              precision    recall  f1-score   support

           0       0.87      0.91      0.89     15018
           1       0.84      0.76      0.80      8779

    accuracy                           0.86     23797
   macro avg       0.85      0.84      0.84     23797
weighted avg       0.86      0.86      0.86     23797
```



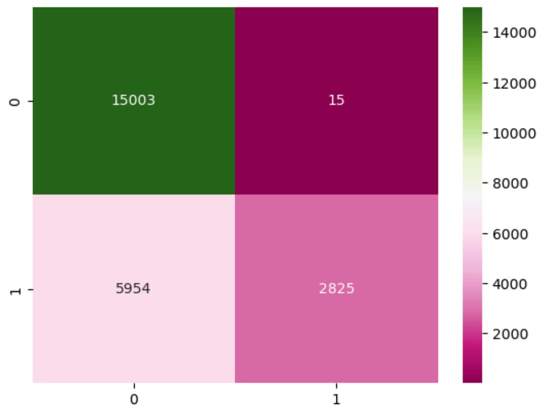
The performance metrics obtained from the classification model suggest a great outcome. The average cross-validation score of 75.1% indicates a high level of accuracy across different folds of the data during training. When applied to the test data, the model's accuracy drops to 74.9%, suggesting that it may struggle to generalize well to unseen instances. The F1 score, which combines precision and recall, is 48.6%, indicating a very bad performance in identifying positive instances (1) and minimizing false positives and false negatives. The recall score of 76% suggests that the model is effective at capturing a high proportion of actual positive cases, while the precision score of 84% indicates that it does not label negative cases as positive. A detailed examination of the classification report reveals that for class 0, the precision and recall values are 87% and 91% respectively, while for class 1, they are 84% and 76%. These metrics collectively suggest a robust model performance, in both correctly identifying instances of class 0, and identifying instances of class 1. The macro average F1 score, considering both classes, is 84%, while the weighted average is 86%. These results indicate that the ID3 classifier effectively contributed to enhancing the model's predictive capabilities.

CART:

CART (Classification and Regression Trees) is another method for constructing decision trees. CART is concentrated on binary splits at each decision tree node, as opposed to ID3, which can accommodate multi-way divides. This indicates that CART takes into account dividing the data into two subsets based on a selected attribute's threshold value at each step. In order to maximize a selected criterion, CART chooses the attribute and matching threshold that divides the data into two subsets the best. The criterion for classification tasks is typically the Gini impurity, which quantifies the probability of incorrectly identifying a randomly selected dataset instance. The criterion for regression tasks may be variance reduction measured in another way, such as mean squared error.



```
Average cross validation score: 0.751
Test accuracy: 0.749
F1 score: 0.486
Classification Report :
              precision    recall  f1-score   support

           0       0.87      0.91      0.89     15018
           1       0.84      0.76      0.80      8779

    accuracy                           0.86     23797
   macro avg       0.85      0.84      0.84     23797
weighted avg       0.86      0.86      0.86     23797
```

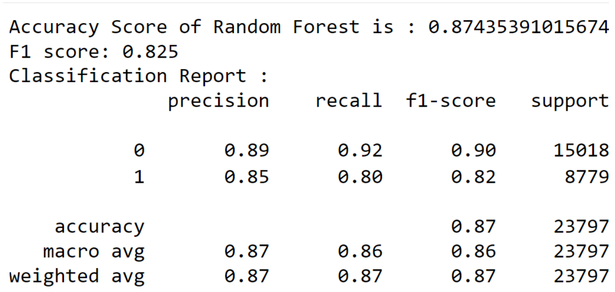


The performance metrics obtained from the classification model suggests a great outcome. The average cross-validation score of 75.1% indicates a high level of accuracy across different folds of the data during training. When applied to the test data, the model's accuracy drops to 74.9%, suggesting that it may struggle to generalize well to unseen instances. The F1 score, which combines precision and recall, is 48.6%, indicating a very bad performance in identifying positive instances (1) and minimizing false positives and false negatives. The recall score of 76% suggests that the model is effective at capturing a high proportion of actual positive cases, while the precision score of 84% indicates that it does not label negative cases as positive. A detailed examination of the classification report reveals that for class 0, the precision and recall values are 87% and 91% respectively, while for class 1, they are 84% and 76%. These metrics collectively suggest a robust

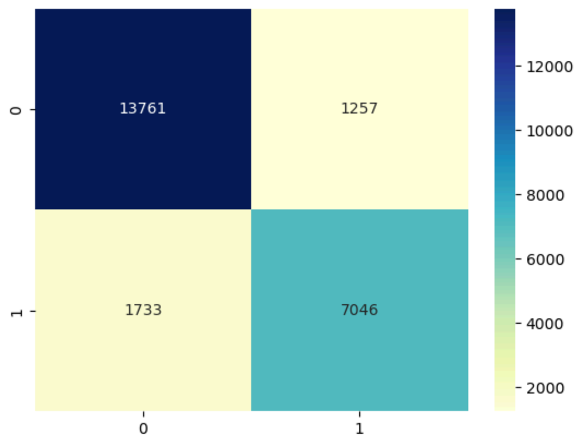
model performance, in both correctly identifying instances of class 0, and identifying instances of class 1. The macro average F1 score, considering both classes, is 84%, while the weighted average is 86%. These results indicate that the CART classifier effectively contributed to enhancing the model's predictive capabilities. It is clear that both ID3 and CART produce the same outputs because they both use decision trees as their base, but the only difference is how each one is calculated.

Random Forest:

Random Forest is an ensemble learning technique. Its ease of use and adaptability make it one of the most potent and popular machine learning algorithms. In Random Forest, several decision trees are built during the training period, and the output class is the mean prediction (regression) or mode of the classes (classification) of each individual tree. Random Forest uses bootstrap sampling, which selects a subset of samples at random using replacement to train each decision tree given a dataset containing N samples and M features. Variety is introduced among the trees by this random sampling. Every decision tree node considers separating a random subset of features. In addition to preventing individual trees from being overly linked, this increases the diversity among the trees. Each decision tree is constructed using the selected subset of samples and features via techniques like CART (Classification and Regression Trees) . The trees are grown deep enough to minimize bias but are typically not pruned. For classification tasks, each tree "votes" for the most popular class among the input sample. The class with the most votes becomes the predicted class label. For regression tasks, the predictions from all trees are averaged to obtain the final prediction.



```
Accuracy Score of Random Forest is : 0.87435391015674
F1 score: 0.825
Classification Report :
              precision    recall  f1-score   support

           0       0.89      0.92      0.90     15018
           1       0.85      0.80      0.82      8779

    accuracy                           0.87     23797
   macro avg       0.87      0.86      0.86     23797
weighted avg       0.87      0.87      0.87     23797
```



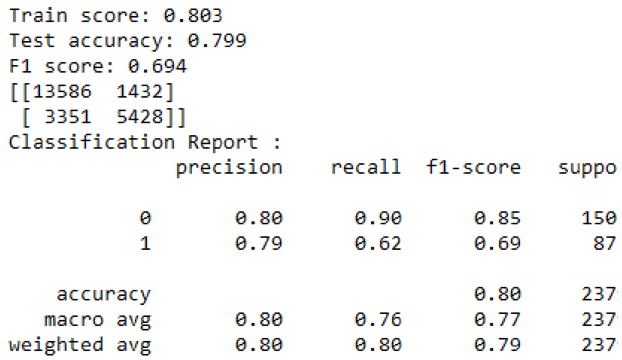
The performance metrics obtained from the classification model suggest a great outcome. The accuracy score of 87.4% indicates a high level of accuracy across different folds of the data during training. The F1 score, which combines precision and recall, is 82.5%, indicating a great performance in identifying positive instances (1) and minimizing false positives and false negatives. The recall score of 80% suggests that the model is effective at capturing a high proportion of actual positive cases, while the precision score of 85% indicates that it does not label negative cases as positive. A detailed examination of the classification report reveals that for class 0, the precision and recall values are 89% and 92% respectively, while for class 1, they are 85% and 80%. These metrics collectively suggest a robust model performance, in both correctly identifying instances of class 0, and identifying instances of class 1. The macro average F1 score, considering both classes, is 86%, while the weighted average is 87%. These results indicate that the Random Forest classifier effectively contributed to enhancing the model's predictive capabilities. Overall, the model has a very high performance in identifying both positive and negative data points correctly.

4. Multiple Layer Perception (Artifical Neural Network)

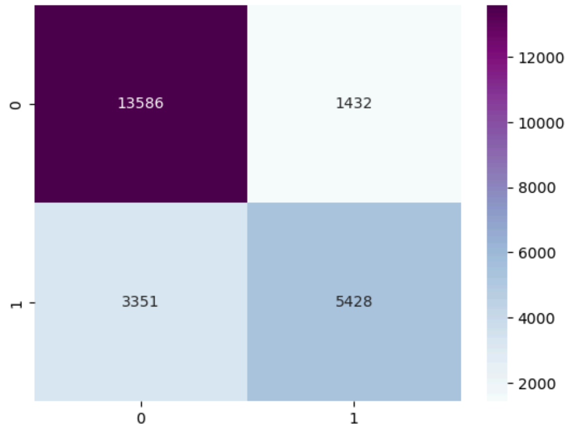
Artificial Neural Networks (ANNs) derive from the concept of perceptions. They intake feature variables with initially random weights, process them through an activation function, and aim to produce an output. Through iterative learning, the weights undergo adjustments until reaching an optimal configuration that minimizes errors. ANNs employ diverse activation functions to ascertain the most effective model among them.

Identity Function Activation

The outcomes were garnered from a multiple-layer perception trained with the identity function as the activation function for the ANN. The model comprises an input layer, a single hidden layer containing 20 neurons, and an output layer. Stochastic gradient descent is utilized to learn and update the weights. The multiple-layer perception model, employing the identity function as its activation function, achieved a training score of 80.3% and a test accuracy of 79.9%. The F1 score, a measure of the model's accuracy in terms of precision and recall, stands at 69.4%. In more detail, the confusion matrix reveals that out of 15,018 instances labeled as class 0, the model correctly identified 13,586, yielding a recall of 90%, while for class 1, out of 8,779 instances, it correctly identified 5,428, resulting in a recall of 62%. Precision for class 0 is 80% and for class 1 is 79%. The macro average F1 score, considering both classes, is 77%, while the weighted average is 79%. These metrics collectively suggest that the model performs reasonably well, particularly in correctly identifying instances of class 0, albeit with a slightly lower performance in correctly identifying instances of class 1.



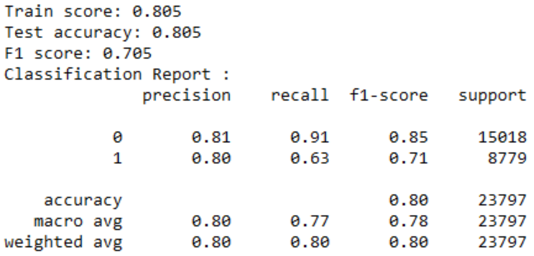
```
Train score: 0.803
Test accuracy: 0.799
F1 score: 0.694
[[13586  1432]
 [ 3351  5428]]
Classification Report :
              precision    recall  f1-score   suppo

           0       0.80      0.90      0.85      150
           1       0.79      0.62      0.69       87

    accuracy                           0.80      237
   macro avg       0.80      0.76      0.77      237
weighted avg       0.80      0.80      0.79      237
```



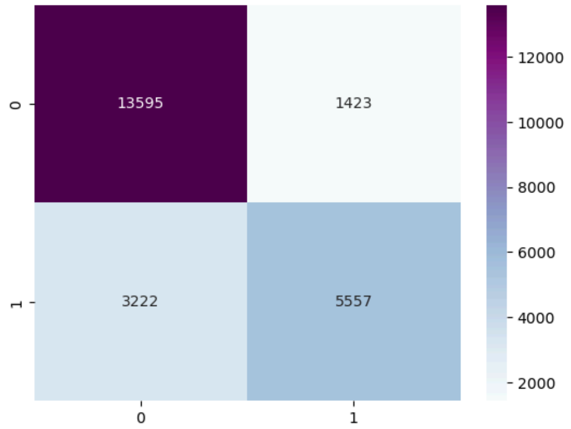
Furthermore, the specificity of the model, calculated as the true negative rate, is approximately 61.83%. This indicates the proportion of correctly identified negative instances (class 0) out of all actual negative instances (sum of false positives and true negatives).

Sigmoid Function Activation

The outcomes were garnered from a multiple-layer perception trained with the sigmoid function as the activation function for the ANN. The model comprises an input layer, a single hidden layer containing 20 neurons, and an output layer. Stochastic gradient descent is utilized to learn and update the weights. With the sigmoid function activation, the multiple-layer perception model achieved notable performance metrics. The training score reached 80.5%, matching the test accuracy at 80.5%. The F1 score stands at a commendable 70.5%, indicating a balanced performance in terms of precision and recall. A detailed examination of the classification report reveals that for class 0, the precision and recall values are 81% and 91% respectively, while for class 1, they are 80% and 63%. These metrics collectively suggest a robust model performance, particularly in correctly identifying instances of class 0, while maintaining a respectable performance in identifying instances of class 1. The macro average F1 score, considering both classes, is 78%, while the weighted average is 80%. These results indicate that the sigmoid function activation effectively contributed to enhancing the model's predictive capabilities.



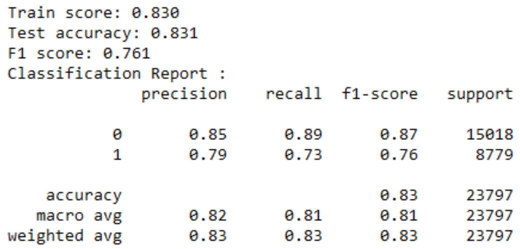
```
Train score: 0.805
Test accuracy: 0.805
F1 score: 0.705
Classification Report :
              precision    recall  f1-score   support

           0       0.81      0.91      0.85     15018
           1       0.80      0.63      0.71      8779

    accuracy                           0.80     23797
   macro avg       0.80      0.77      0.78     23797
weighted avg       0.80      0.80      0.80     23797
```



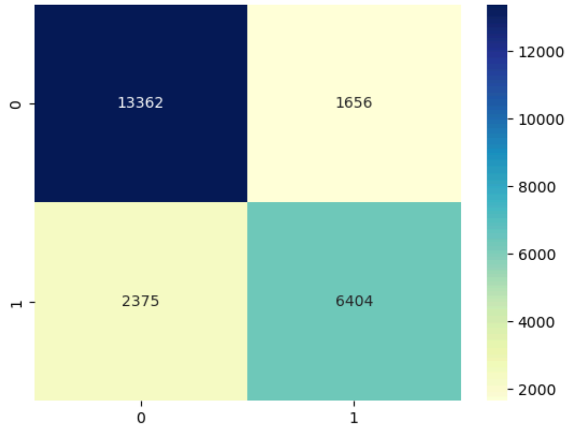
Furthermore, the specificity of the model, calculated using the provided confusion matrix, is approximately 63.30%. This indicates the proportion of correctly identified negative instances (class 0) out of all actual negative instances (sum of false positives and true negatives).

Hyperbolic Tangent Activation

The outcomes were garnered from a multiple-layer perception trained with the hyperbolic tangent function as the activation function for the ANN. The model comprises an input layer, a single hidden layer containing 20 neurons, and an output layer. Stochastic gradient descent is utilized to learn and update the weights. Utilizing the hyperbolic tangent activation function has noticeably enhanced the performance of the multiple-layer perception model. The training score reached an impressive 83%, closely matching the test accuracy at 83.1%. The F1 score significantly improved to 76.1%, indicating a substantial enhancement in the model's precision and recall balance. A detailed analysis of the classification report reveals that for class 0, the precision and recall values are 85% and 89% respectively, while for class 1, they are 79% and 73%. These metrics collectively signify a robust model performance, particularly in correctly identifying instances of class 0, while maintaining a commendable performance in identifying instances of class 1. The macro average F1 score, considering both classes, is 81%, while the weighted average is 83%. These results highlight the effectiveness of the hyperbolic tangent activation function in augmenting the model's predictive capabilities.



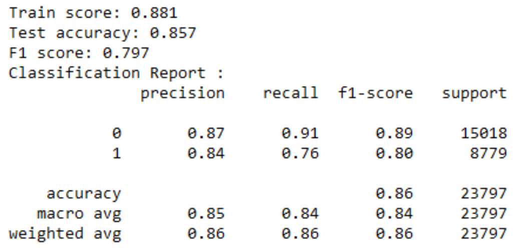
```
Train score: 0.830
Test accuracy: 0.831
F1 score: 0.761
Classification Report :
              precision    recall  f1-score   support

           0       0.85      0.89      0.87     15018
           1       0.79      0.73      0.76      8779

    accuracy                           0.83     23797
   macro avg       0.82      0.81      0.81     23797
weighted avg       0.83      0.83      0.83     23797
```



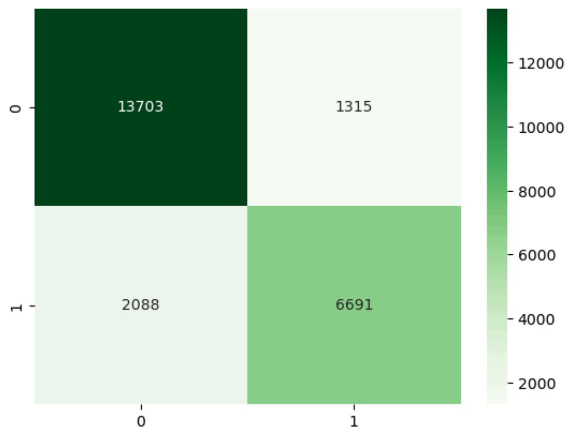
Furthermore, the specificity of the model, calculated using the provided confusion matrix, is approximately 72.95%. This indicates the proportion of correctly identified negative instances (class 0) out of all actual negative instances (sum of false positives and true negatives).

Rectified Linear Unit Function Activation

The adoption of the Rectified Linear Unit function activation has led to a remarkable enhancement in the performance of the multiple-layer perception model. With a training score of 88.1% and a test accuracy of 85.7%, the model showcases substantial proficiency. The F1 score has notably increased to 79.7%, indicating a significant improvement in the precision and recall balance. Upon closer examination of the classification report, it's evident that for class 0, the precision and recall values are 87% and 91% respectively, while for class 1, they stand at 84% and 76%. These metrics collectively underscore a robust model performance, particularly in correctly identifying instances of class 0, while maintaining a commendable performance in identifying instances of class 1. The macro average F1 score, considering both classes, is 84%, while the weighted average stands at 86%. These results underscore the effectiveness of the ReLU activation function in bolstering the model's predictive capabilities.



```
Train score: 0.881
Test accuracy: 0.857
F1 score: 0.797
Classification Report :
              precision    recall  f1-score   support

           0       0.87      0.91      0.89     15018
           1       0.84      0.76      0.80      8779

    accuracy                           0.86     23797
   macro avg       0.85      0.84      0.84     23797
weighted avg       0.86      0.86      0.86     23797
```

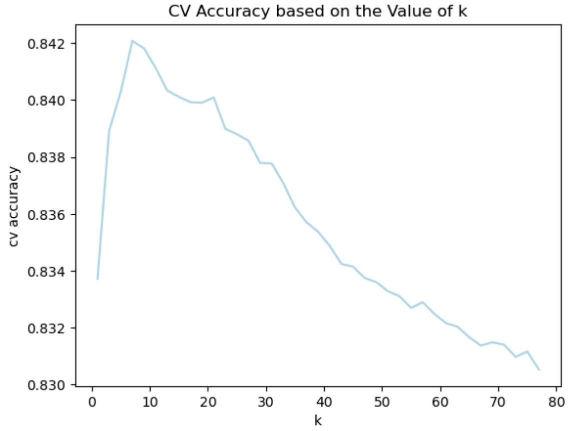


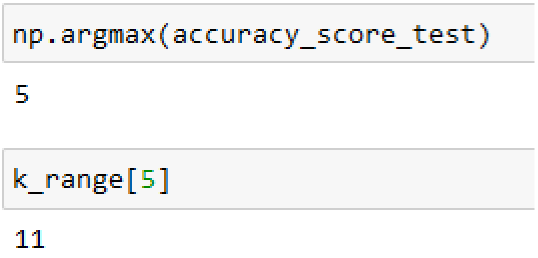
Furthermore, the specificity of the model, computed based on the provided confusion matrix, is approximately 76.22%. This indicates the proportion of correctly identified negative instances (class 0) out of all actual negative instances (sum of false positives and true negatives).

Overall, the artificial neural network has demonstrated considerable performance on the dataset, yet its computational demands and lack of interpretability pose significant challenges. Determining optimal parameters such as the number of hidden layers and neurons is inherently complex, compounded by the difficulty in visualizing and comprehending the multitude of weights generated across network connections. Despite the superior performance observed in the ANN employing the rectified linear unit activation function, practical constraints such as computational complexity and memory requirements necessitate its dismissal. Moreover, ANNs exhibit opacity in their decision-making processes and are sensitive to initialization, further complicating their use. Additionally, ANNs necessitate large datasets to effectively optimize parameters during training, potentially leading to inadequate pattern recognition in smaller datasets.

5. KNN

This machine learning model operates on instance-based supervised learning principles. It calculates the distance between each observation and identifies the k nearest neighbors to classify new instances based on the majority vote of these neighbors. The Euclidean distance function is employed for this purpose. Prior to model fitting, feature variables are standardized using a standard-scaler to address differences in scale that could otherwise bias the Euclidean distance measure, favoring observations with inherently smaller values. Furthermore, to mitigate overfitting and ensure the model's generalization ability, a 5-fold cross-validation is conducted during learning. To determine the optimal k value, the model is trained across a range of k-values from 1 to *n*, excluding even numbers and selecting only odd numbers. Due to computational constraints, a subset containing odd numbers between 1 and 77 is chosen for iteration. For each k value, metrics including accuracy score, classification report, confusion matrix, and AUC are computed, and a ROC curve is plotted. The results are summarized graphically for analysis:





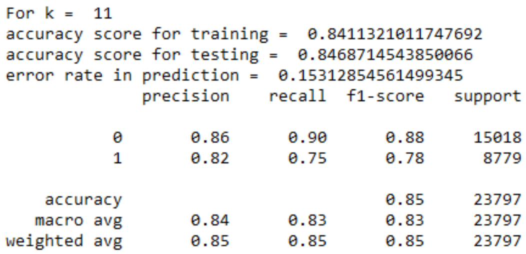
```
np.argmax(accuracy_score_test)
```

```
5
```

```
k_range[5]
```

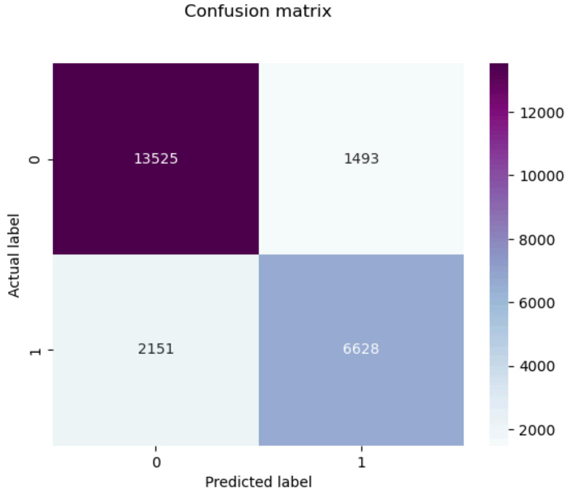
```
11
```

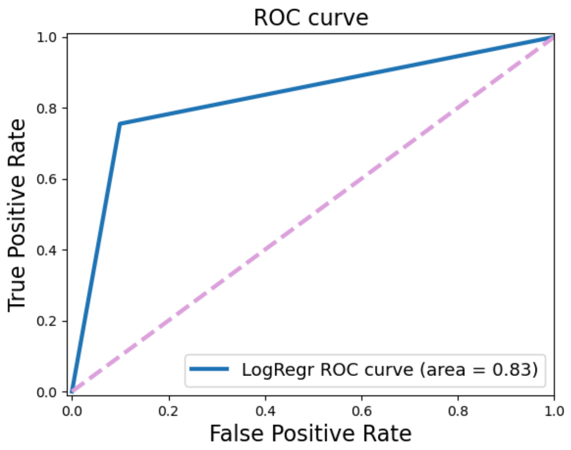
The graph illustrates the 5-fold cross-validation accuracy scores as the value of k increases. Initially, the accuracy is relatively low at k=1, gradually rising until it peaks at k=11, after which it begins to decline with further increases in k. Notably, the k value yielding the highest cross-validation accuracy score is identified as k=11. Subsequently, the model is trained using this optimal k value, resulting in the following outcomes:



```
For k =  11
accuracy score for training =  0.8411321011747692
accuracy score for testing =  0.8468714543850066
error rate in prediction =  0.15312854561499345
              precision    recall  f1-score   support

           0       0.86      0.90      0.88     15018
           1       0.82      0.75      0.78      8779

    accuracy                           0.85     23797
   macro avg       0.84      0.83      0.83     23797
weighted avg       0.85      0.85      0.85     23797
```

For k=11, the machine learning model achieved an accuracy score of approximately 84.11% on the training set and around 84.69% on the testing set. The error rate in prediction is calculated to be about 15.31%. In terms of precision and recall, for class 0 (no hotel cancellations), the precision is 86% and the recall is 90%, resulting in an F1-score of 88%. For class 1 (hotel cancellations), the precision is 82% and the recall is 75%, yielding an F1-score of 78%. The overall accuracy of the model is 85%, with a macro average F1-score of 83% and a weighted average F1-score of 85%. These metrics collectively indicate a reasonably effective performance of the model in classifying hotel cancellations based on the chosen features and the k value.





The ROC curve illustrates the relationship between the True Positive Rate (TPR) and the False Positive Rate (FPR) across different values of k. As the curve is shifted towards the top-left corner, it signifies that the model's performance is favorable, with higher TPR and lower FPR values. The area under the ROC curve (AUC), quantified as 0.83 in this case, further corroborates the effectiveness of the model. A higher AUC value indicates better discrimination ability, suggesting that the model distinguishes between positive and negative instances well. Therefore, with an AUC of 0.83, the model demonstrates good performance in distinguishing between instances of hotel cancellations and non-cancellations.

Indeed, the 11-nearest neighbors (11-NN) model offers a straightforward and easily interpretable approach to classification. It's intuitive, making it simple to comprehend, evaluate, and validate. However, determining the optimal value for the parameter k, which maximizes accuracy without succumbing to overfitting, presents a challenge. This process is subjective and time-consuming, particularly with large datasets. Nonetheless, the 11-NN model aligns well with the characteristics of the dataset and emerges as one of the top-performing models. Its simplicity and effectiveness make it a favorable choice for classification tasks, despite the challenges in parameter selection.

Model Selection

The aim of this project is to accurately predict canceled hotel reservations. Several criteria were taken into account in choosing the most suitable model for the dataset. We extensively experimented with a range of machine learning models to identify the optimal match for our dataset. The table below summarizes the model accuracy and TNR for each model implemented:

Model	Accuracy

Decision Trees Random Forest	87.4%
ANN Recitfied Linear Unit	85.7%
KNN	84.7%
ANN Hyperbolic Tangent	83.1%
ANN Sigmoid Function	80.5%
Logistic Regression	79.9%
ANN Identity Function	79.9%
Decision Trees ID3	74.9%
Decision Trees CART	74.9%
Naive Bayes	58.7%

According to this table and the analysis of each model implementation, the 3 top performing models are KNN, Decision Trees: Random Forest, and ANN: Rectified Linear Unit as Activation function.

The K-Nearest Neighbors (KNN) algorithm, while simple and intuitive, exhibits limitations, particularly with larger datasets. As the dataset size increases, the algorithm's performance tends to decline due to challenges in optimizing parameters for maximum predictive accuracy. Moreover, KNN is memory-intensive, rendering it suitable only for small to medium-sized datasets. The computational cost of classifying a new instance using KNN can be prohibitive, as the algorithm iterates through all training instances. Ultimately, the drawbacks associated with the KNN classifier outweigh its benefits, leading to its exclusion from further consideration.

Ultimately, the Random Forest algorithm emerges as the most suitable and top-performing model for our dataset. Its resilience to overfitting, facilitated by the amalgamation of multiple decision trees, is particularly advantageous. This aligns perfectly with the project's objective of accurately forecasting canceled reservations. Random Forest exhibits high predictive capability and excels in capturing intricate interactions among features. However, it is essential to acknowledge that the model does have drawbacks. Namely, it can be computationally intensive, and its interpretability may pose challenges compared to simpler models. Despite these limitations, the overall strengths of Random Forest make it the preferred choice for our predictive modeling task.

In conclusion, after thorough analysis and experimentation, it is clear that the Random Forest Algorithm is the most suitable model for hotel cancellation prediction.