**Final Project Report**

**INFO6105 Data Science Methods and Tools**

**A close up of a logo

Description automatically generated**

**Topic: Predicting Hotel Booking Cancellations**

**Team Members:**

Akash Sinha ()

Anuradha Choudhary ()

Sriya Vaidyanathan (001376835)

Vividha Singh (001085454)

**Contents:**

1. Introduction

2. Background of the Project

3. Methodology

4. Dataset

5. Results and Analysis

6. Conclusion

7. Future Scope

**INTRODUCTION**

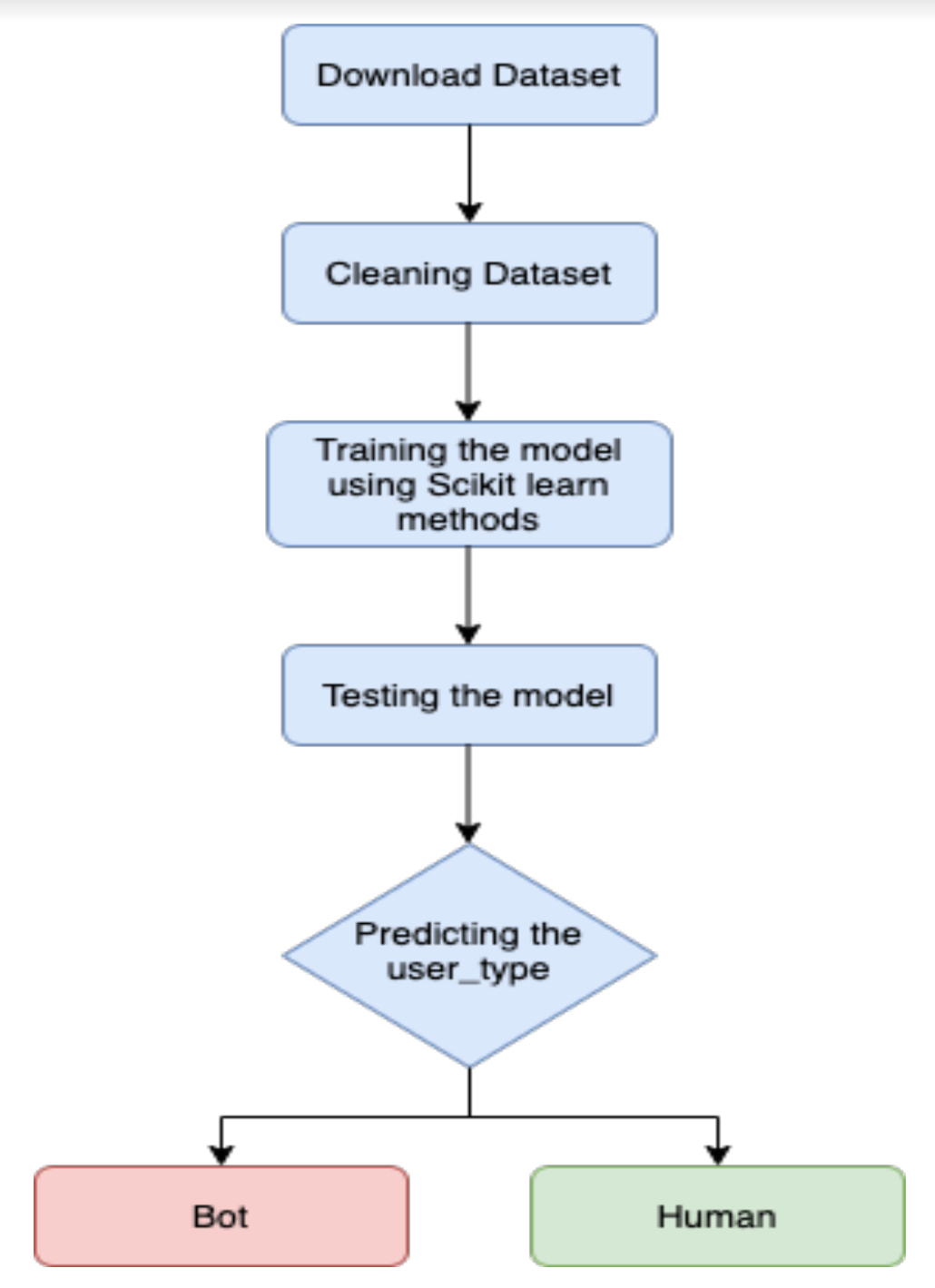
The data describes datasets with hotel booking demand data. One of the hotels (H1) is a resort hotel and the other is a city hotel (H2). Both datasets share the same structure, with 31 variables describing the 40,060 observations of H1 and 79,330 observations of H2. Each observation represents a hotel booking. Both datasets comprehend bookings due to arrive between the 1st of July of 2015 and the 31st of August 2017, including bookings that effectively arrived and bookings that were canceled. Since this is real data, all data elements pertaining to the identification of the hotel or customer were deleted. Due to the scarcity of real business data for scientific and educational purposes, these datasets can have an important role for research and education in revenue management, machine learning, data mining, as well as in other fields.

In the hotel industry it is quite common for customers to change their booking׳s attributes, like the number of persons, staying duration, or room type preferences, either at the time of their check-in or during their stay. It is also common for hotels not to know the correct nationality of the customer until the moment of check-in. Therefore, even though the capture of data was considered a timespan prior to the arrival date, it is understandable that the distribution of some variables differ between non canceled and canceled bookings. Consequently, the use of these datasets may require this difference in distribution to be taken into account.

**BACKGROUND**

The purpose of using these algorithms is to identify features that best help to predict whether a customer will cancel their hotel booking. This is the dependent variable, where (1 = cancel, 0 = follow through with booking).

**METHODOLOGY**



In this project we have used the following libraries for analysis:

* Numpy
* Pandas
* Scikit-learn
* Matplotlib
* Seaborn

1. **NUMPY**

NumPy is the fundamental package for scientific computing with Python. NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases. In our project we have implemented NumPy for computing mean darkness and mean RGB value of each image, data cleaning (if the ratio is infinity we set it to 99999 and if it’s NaN set to 0).

1. **PANDAS**

Pandas is an open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language. It is built on the NumPy package and its key data structure is called the DataFrame. DataFrames allow you to store and manipulate tabular data in rows of observations and columns of variables. In our project we have implemented pandas to import a csv file as a dataframe, concatenate multiple data frames into a single dataframe, and for setting image features.

1. **SCIKIT-LEARN**

Scikit-learn library is used to implement various algorithms like support vector machines, random forests, Gaussian Naive Bayes, Scalers, and k-neighbours. It also supports Python numerical and scientific libraries like NumPy and SciPy. In our project, we have used SVC, Scalers and k-neighbors for model training

1. **MATPLOTLIB**

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits. In our project, we use MatPlotLib to display a Bar Chart to visualize the Weather details by Month versus Days.

1. **SEABORN**

The **seaborn** package was developed based on the **Matplotlib** library. It is used to create more attractive and informative statistical graphics. While **seaborn** is a different package, it can also be used to develop the attractiveness of **matplotlib** graphics. While **matplotlib** is great, we always want to do **better**

We have used the following machine learning algorithms such as:

**LOGISTIC REGRESSION:**

Logistic regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary).  Like all regression analysis, logistic regression is a predictive analysis.  Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.

**GAUSSIAN NAÏVE BAYES:**

Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.

**KNN ALGORITHM:**

KNN can be used for both classification and regression predictive problems. However, it is more widely used in classification problems in the industry. To evaluate any technique we generally look at 3 important aspects:

1. Ease to interpret output

2. Calculation time

3. Predictive Power

**SUPPORT VECTOR CLASSIFICATION:**

Support Vector Machine (SVM) is a supervised machine learning algorithm capable of performing classification, regression and even outlier detection. The linear SVM classifier works by drawing a straight line between two classes.

**DECISION TREE:**

Decision Tree is one of the most powerful and popular algorithm. Decision-tree algorithm falls under the category of supervised learning algorithms. It works for both continuous as well as categorical output variables.

**RANDOM FOREST:**

Random forest, like its name implies, consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model’s prediction. It uses bagging and features randomness when building each individual tree to try to create an uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree. The model uses two key concepts that gives it the name random:

1. Random sampling of training data points when building trees
2. Random subsets of features considered when splitting nodes

The result of these models can be further analysed to determine the model with highest

accuracy. However, the dataset being big enough may require high computing power.

The problem can be resolved by dimensionality reduction using Principal Component

Analysis (PCA). The 25+ columns that are present in the dataset can be further reduced

by selecting the uncorrelated variables which are also known as principal components.

These components comprise the necessary information required to train the model.The

first model, Naive Bayes is a generative model which is very suitable for high-dimensional

datasets. This model makes naive assumptions about the generative model for each label

to find rough approximation for each class and then uses Bayes classification to define a

distribution pattern for each class, here bots and humans. The second model, Logistic

Regression, is a discriminative model that logs the probability of a discrete dependent

variable in the form of linear combination of independent variables. It also penalizes

increasingly large errors at a constant cost. The third model, Support Vector Machine

models data as existing in some p-1 dimensional space, where each data point has p

features. The objective of the SVM is to learn a separating hyperplane that best divides

the classes. In total, six models will be trained: three classification models on full feature

space and three classification models using PCA features.

**DATASET:**

SOURCE OF DATASET:

<https://www.kaggle.com/jessemostipak/hotel-booking-demand>

SPECIFICATIONS OF DATASET

Initially, the dataset consists of 119390 rows and 32 columns and the dataset is of 16.9MB.

Furthermore, the data set contains booking information for a city hotel and a resort hotel, and includes other information such as when the booking was made, length of stay, the number of adults, children, and/or babies, and the number of available parking spaces, among other things.

|  |  |  |
| --- | --- | --- |
| **COLUMN** | **TYPE** | **DESCRIPTION** |
| Hotel | String | It will hold either Resort Hotel or City Hotel |
| Is\_cancelled | Boolean | The value will be 0 or 1 depending on whether ir is cancelled or not |
| ArrivalDateDayOfMonth | Integer | Day of the month of the arrival date |
| ArrivalDateMonth | String | Month of ArrivalDate with 12 categories: “January” to “December” |
| ArrivalDateWeekNumber | Integer | Week number of the arrival date |
| ArrivalDateYear | Integer | Year of arrival date |
| Adults | Integer | Number of Adults |
| Children | Integer | Number of children |
| Babies | Integer | Number of babies |
| Meal | String | Type of meal booked. Categories are presented in standard hospitality meal packages:  **Undefined/SC** – no meal package;  **BB** – Bed & Breakfast;  **HB** – Half board (breakfast and one other meal – usually dinner);  **FB** – Full board (breakfast, lunch and dinner) |
| Market Segment | String | Market segment designation. In categories, the term “TA” means “Travel Agents” and “TO” means “Tour Operators” |
| Distribution\_channel | String | Booking distribution channel. The term  “TA” means “Travel Agents” and  “TO” means “Tour Operators” |
| is\_repeated\_guest | Integer | Value indicating if the booking name was from a repeated guest (1) or not (0)  Meaning:  if a profile was associated with the booking customer. If so, and if the customer profile creation date was prior to the creation date for the booking on the PMS database it was assumed the booking was from a repeated guest |
| previous\_cancellations | Integer | Number of previous bookings that were cancelled by the customer prior to the current booking |
| reserved\_room\_type | String | Code of room type reserved. Code is presented instead of designation for anonymity reasons |
| assigned\_room\_type | String | Code for the type of room assigned to the booking. Sometimes the assigned room type differs from the reserved room type due to hotel operation reasons (e.g. overbooking) or by customer request. Code is presented instead of designation for anonymity reasons |
| booking\_changes | Integer | Number of changes/amendments made to the booking from the moment the booking was entered until the moment of check-in or cancellation |
| deposit\_type | String | Indication on if the customer made a deposit to guarantee the booking. This variable can assume three categories: calculated based on the payments identified for the booking before the booking׳s arrival or cancellation date.  **No Deposit** – no deposit was made;  In case no payments were found the value is “No Deposit”.  .  **Non Refund** – a deposit was made in the value of the total stay cost; If the payment was equal or exceeded the total cost of stay, the value is set as “Non Refund”  **Refundable** – a deposit was made with a value under the total cost of stay. Otherwise the value is set as “Refundable” |
| customer\_type | String | Type of booking, assuming one of four categories:  **Contract** - when the booking has an allotment or other type of contract associated to it;  **Group** – when the booking is associated to a group;  **Transient** – when the booking is not part of a group or contract, and is not associated to other transient booking;  **Transient-party** – when the booking is transient, but is associated to at least other transient booking |
| required\_car\_parking\_spaces | Integer | Number of car parking spaces required by the customer |
| total\_of\_special\_requests | Integer | Number of special requests made by the customer (e.g. twin bed or high floor) |
| reservation\_status | String | Reservation last status, assuming one of three categories:  **Canceled** – booking was canceled by the customer  **Check-Out** – customer has checked in but already departed  **No-Show** – customer did not check-in and did inform the hotel of the reason why |
| reservation\_status\_date | Date | Date at which the last status was set. This variable can be used in conjunction with the ReservationStatus to understand when was the booking canceled or when did the customer checked-out of the hotel |

**DATA PREPROCESSING FLOW**

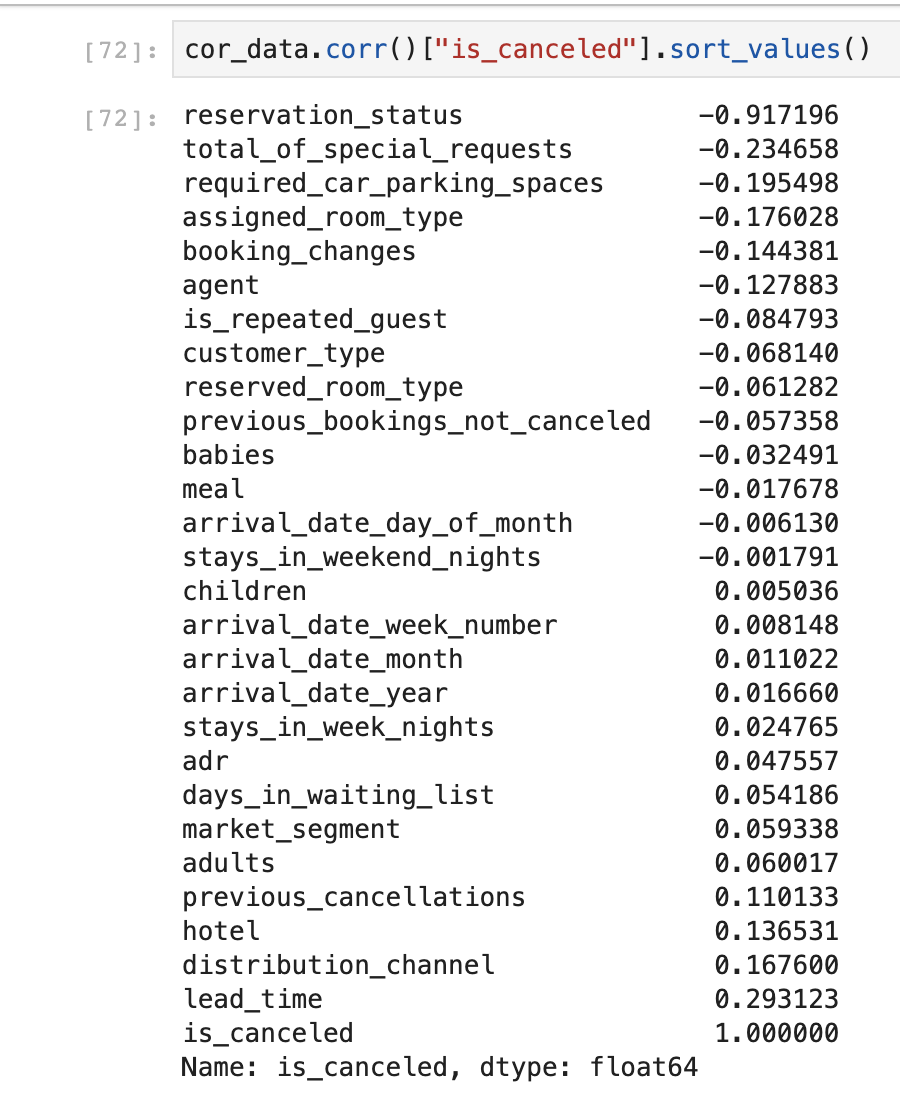
**Data Preprocessing:**

Before the dataset was finalized it had to undergo a lot of preprocessing.

* First, we found the percentage of null values in the 32 columns.   
    
  94.3% of data in the company column were missing values. Therefore the best option was to drop the company column. 13.68% of data in the agent column were missing, but we did not drop the rows because 13.68% of data is a huge amount and those rows have a chance to have crucial information. Since there were just 4 missing rows in the children column, we filled it with 0.
* Second, since hotel and arrival\_Date\_month has unique string values, we map them to integers.

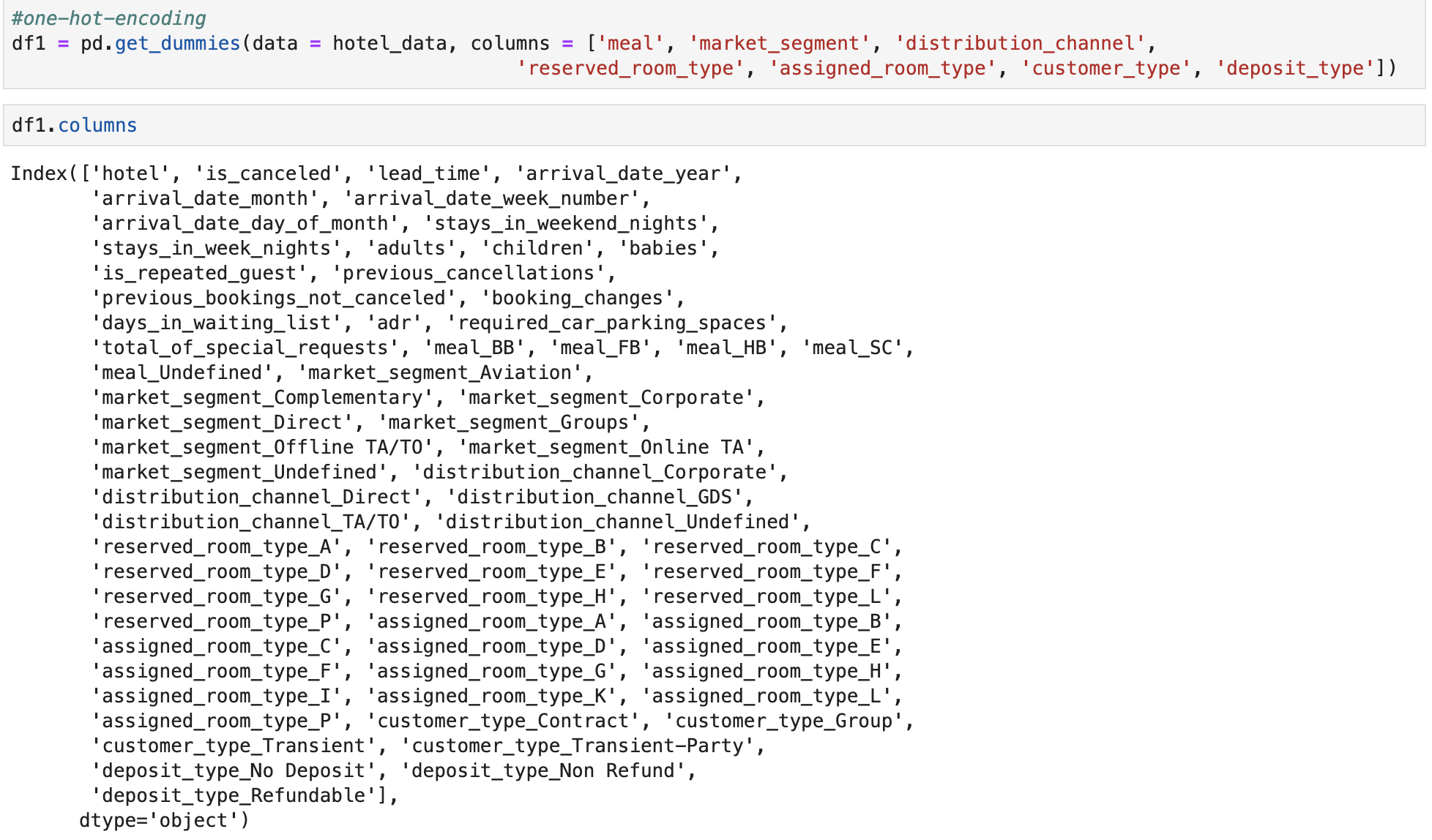


* Third, we correlate the data to understand which is more impactful.



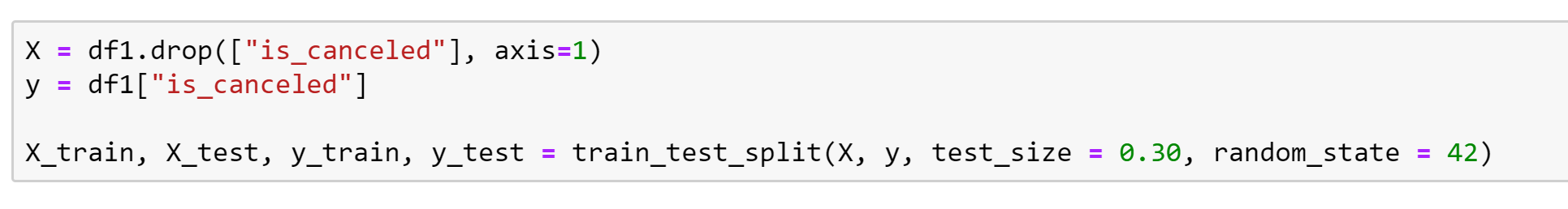
In the sorted list, reservation\_status is an impactful feature. Hence, accuracy rate should be really high. It can be better to drop reservation\_status column to see how other features can predict. I am going to try both. Impacts of three feature that are created:

-> deposit\_given = 0,48131  
-> is\_family = -0,01327  
-> total\_customer = 0,04504  
Also, arrival\_date\_week\_number, stays\_in\_weekend\_nights and arrival\_date\_day\_of\_month will be dropped since their importances are really low while predicting cancellations. Also, the agent column still has missing values. It is important in predicting the cancellation but since the missing values are equal to 13% of the total data it is better to drop that column as it might misguide the predictions.

* In the correlation part, reservation status dominates other features totally. By keeping this in the data set, it is possible to achieve 100% accuracy rate because that feature is a direct way to predict cancellations. For the sake of analysis I will drop reservation\_status and continue analysis without it.
* Finally, some of the columns have a lot of repeated sub parts so for better understanding and clarity of the data set, we will apply *get.dummies* using pandas.  
  

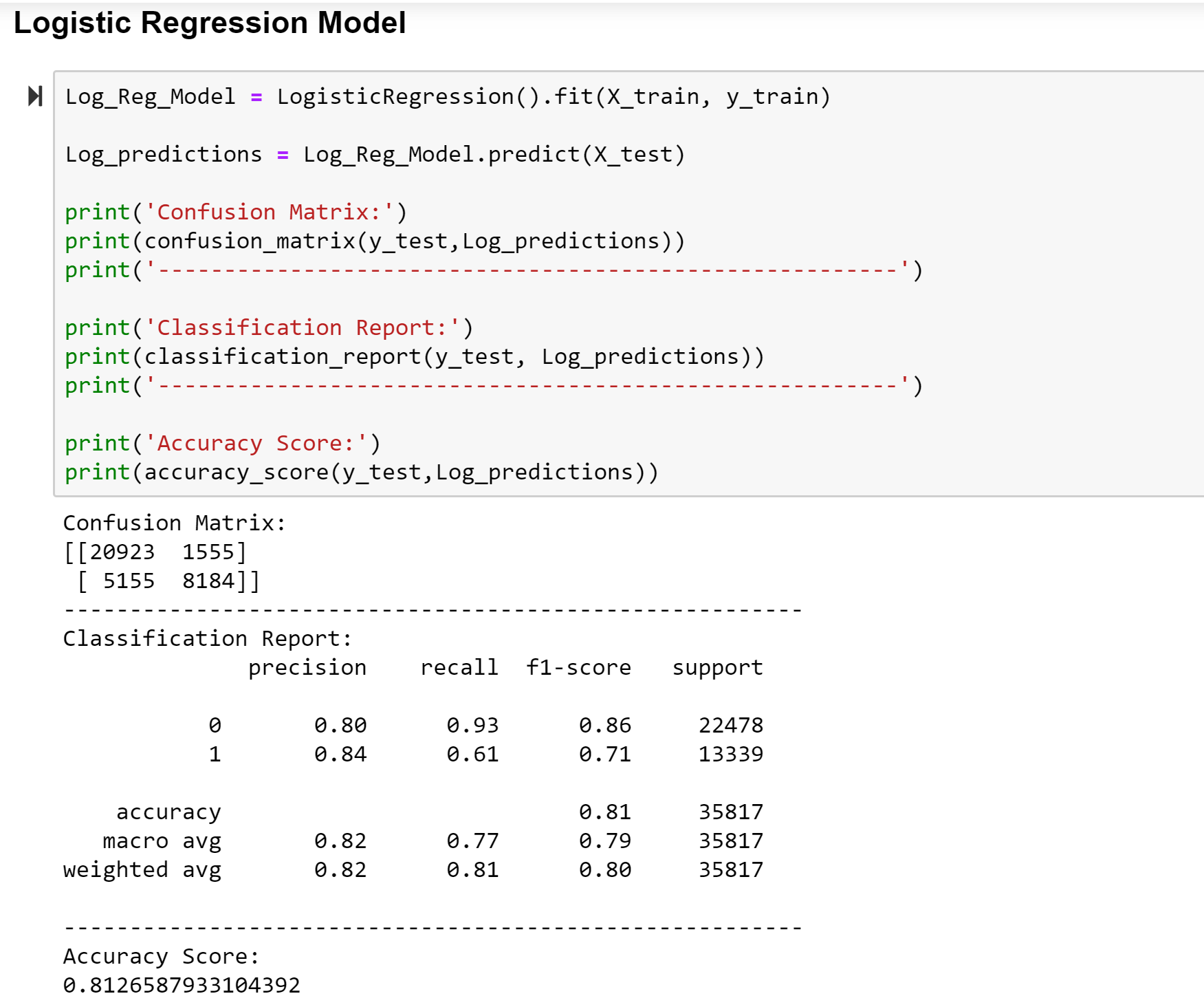
**Results and Analysis**

**Training Model**

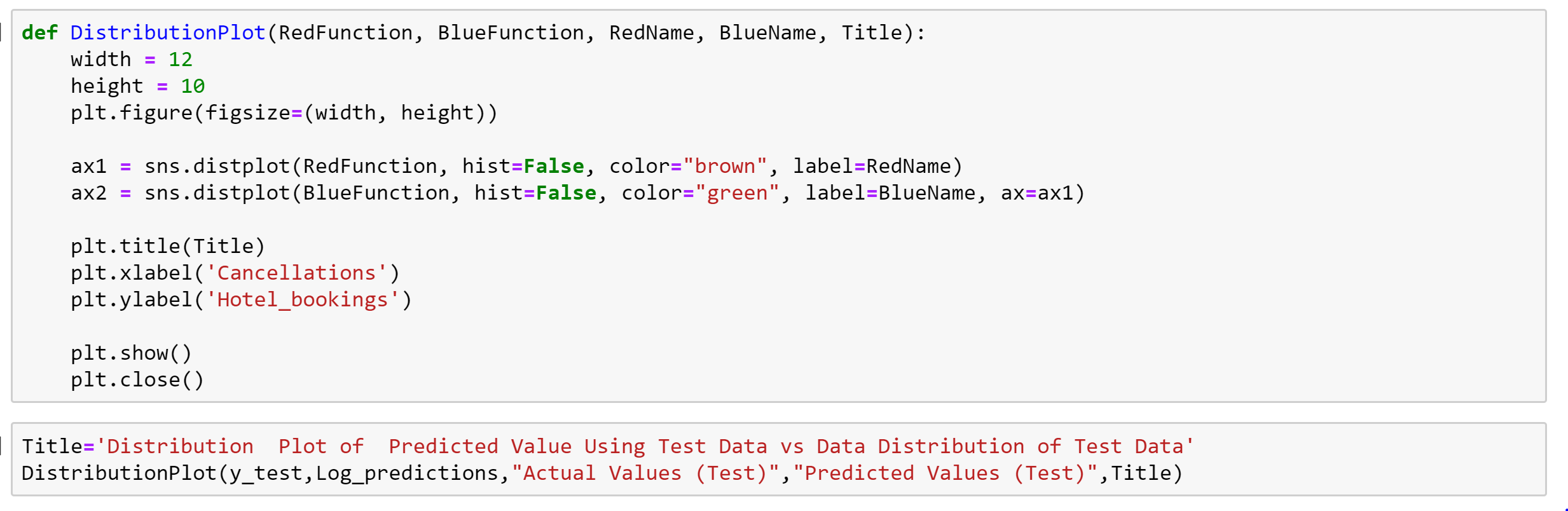


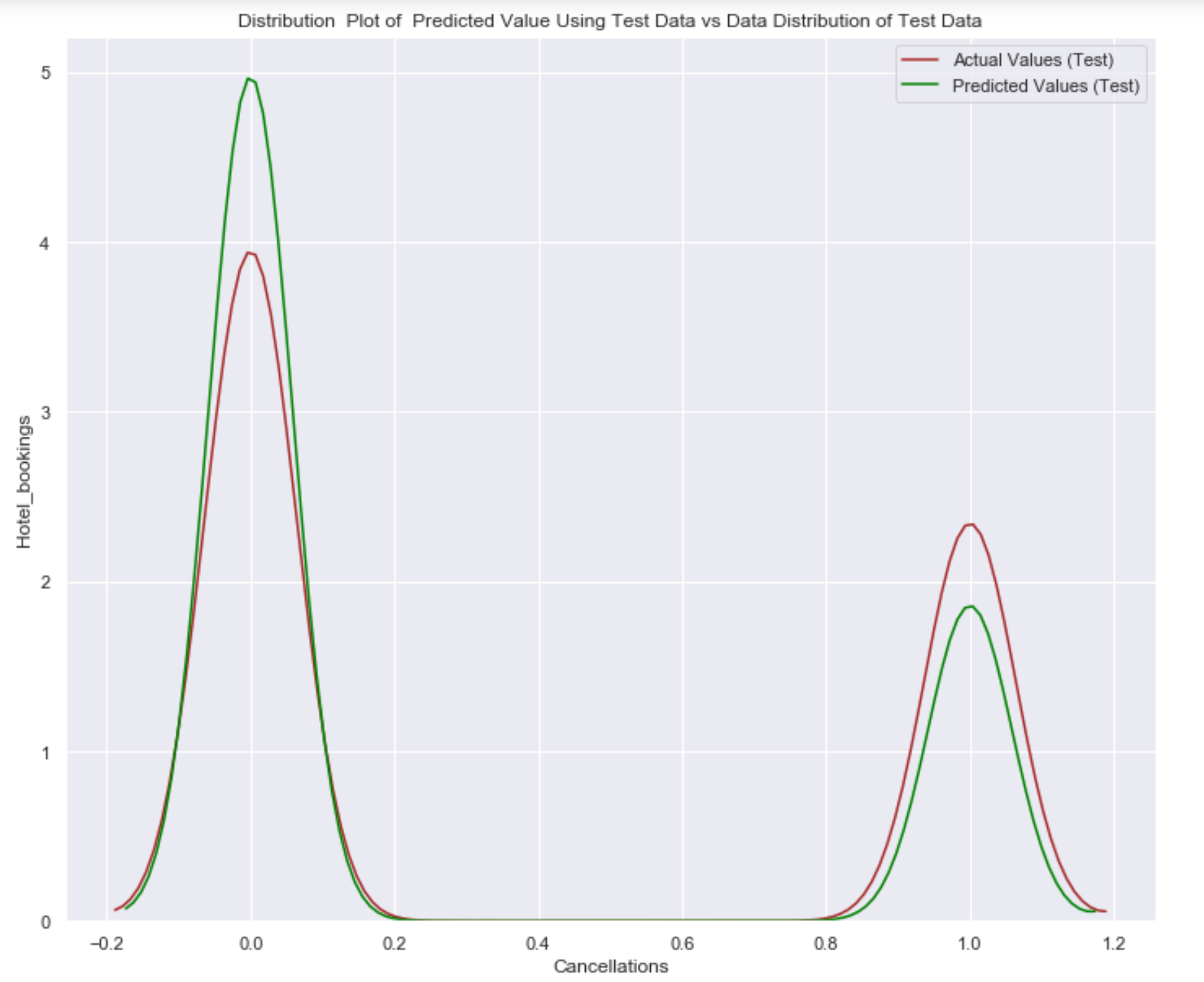
**LOGISTIC REGRESSION MODEL:**

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. In regression analysis, logistic regression (or logit regression) is estimating the parameters of a logistic model (a form of binary regression). Mathematically, a binary logistic model has a dependent variable with two possible values, such as pass/fail which is represented by an indicator variable, where the two values are labeled "0" and "1".

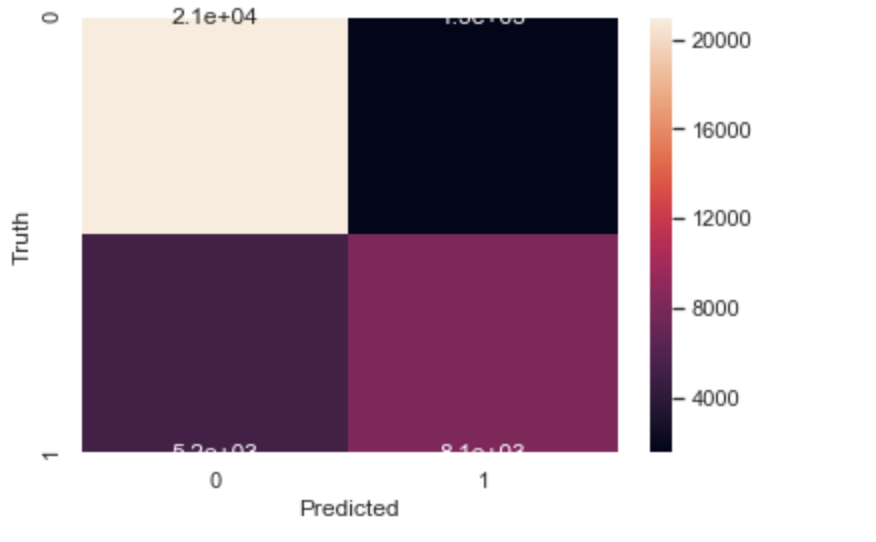


This model results in an accuracy of 81.265 %



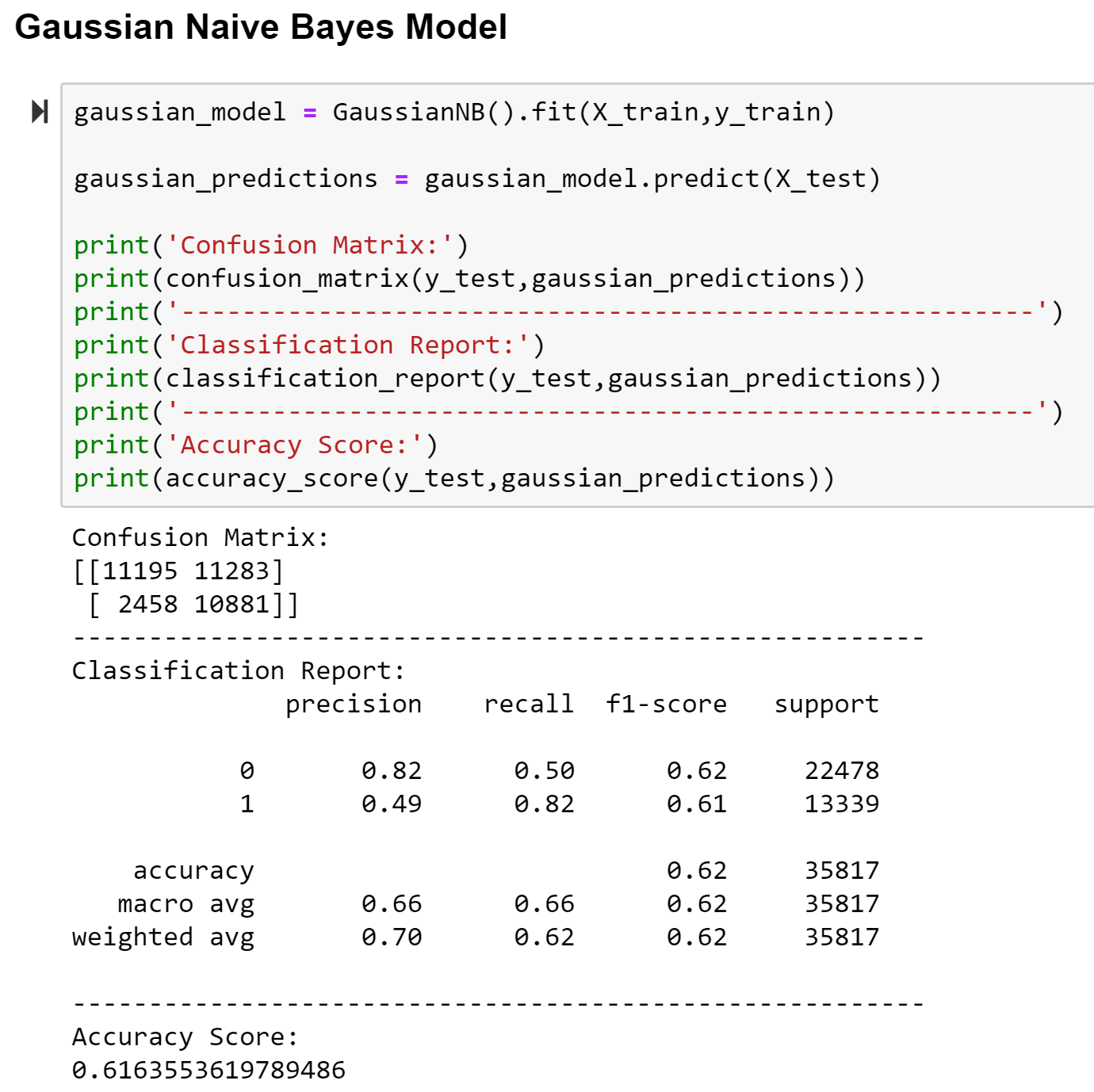


*Confusion Matrix*

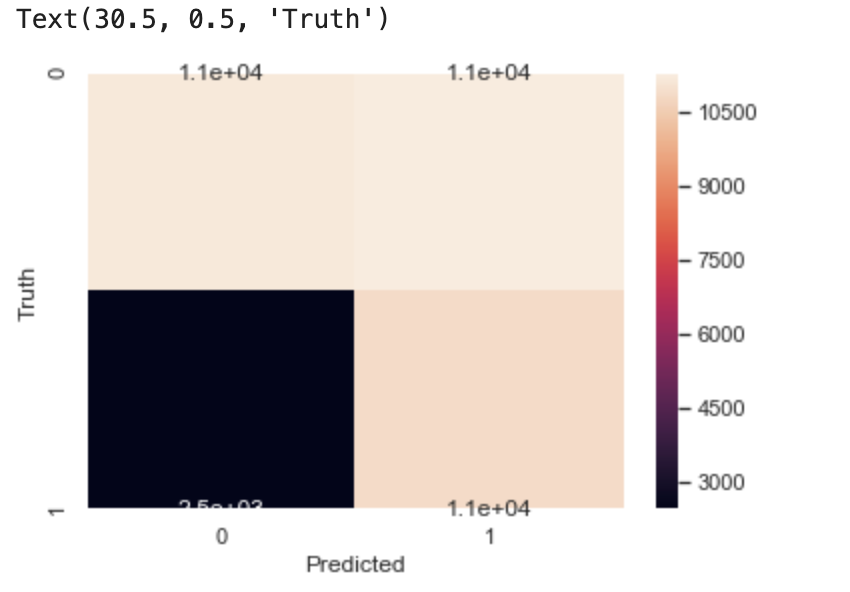


**GAUSSIAN NAÏVE BAYES MODEL:**

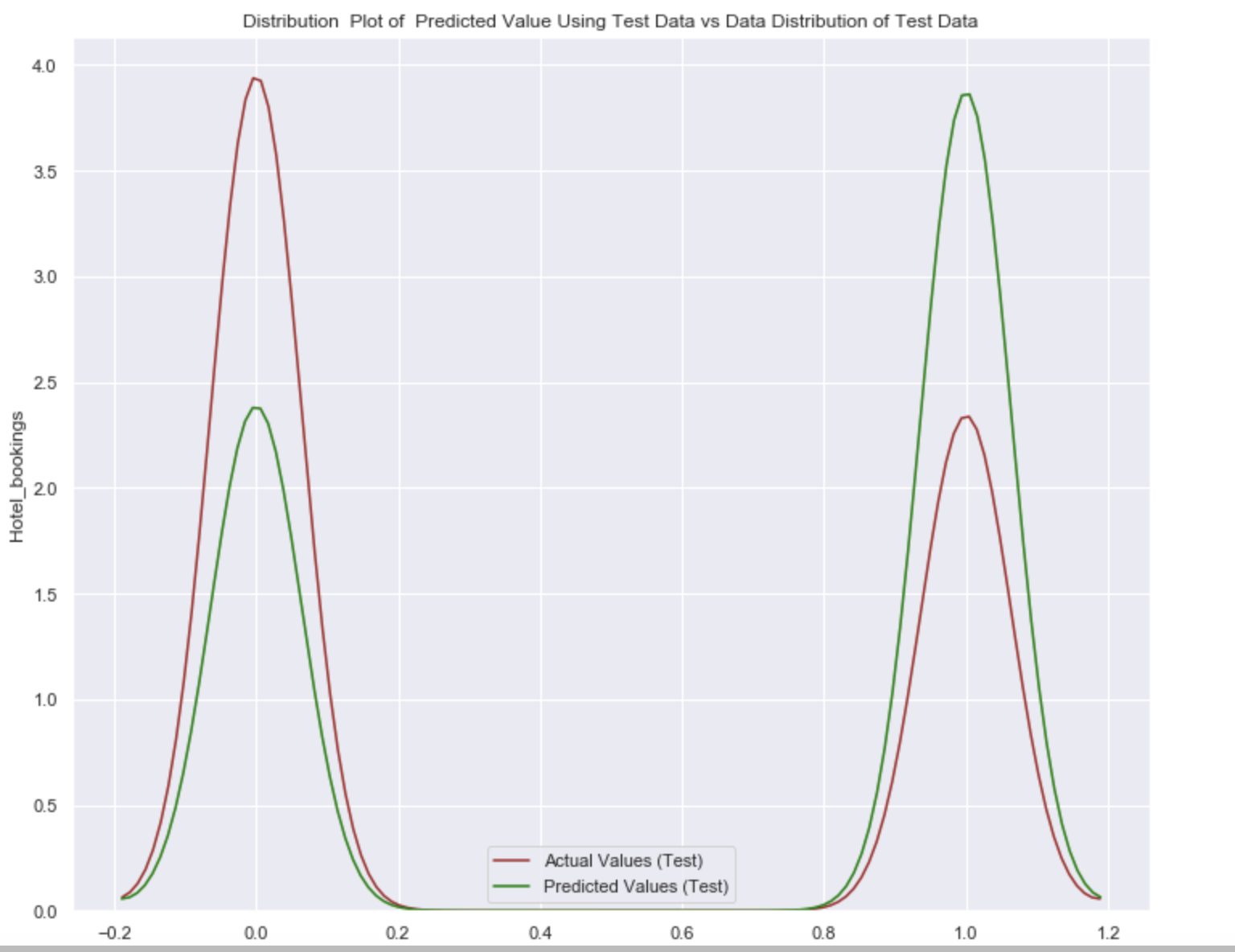
The model has an accuracy rate of 61.63%

****

*Confusion Matrix*

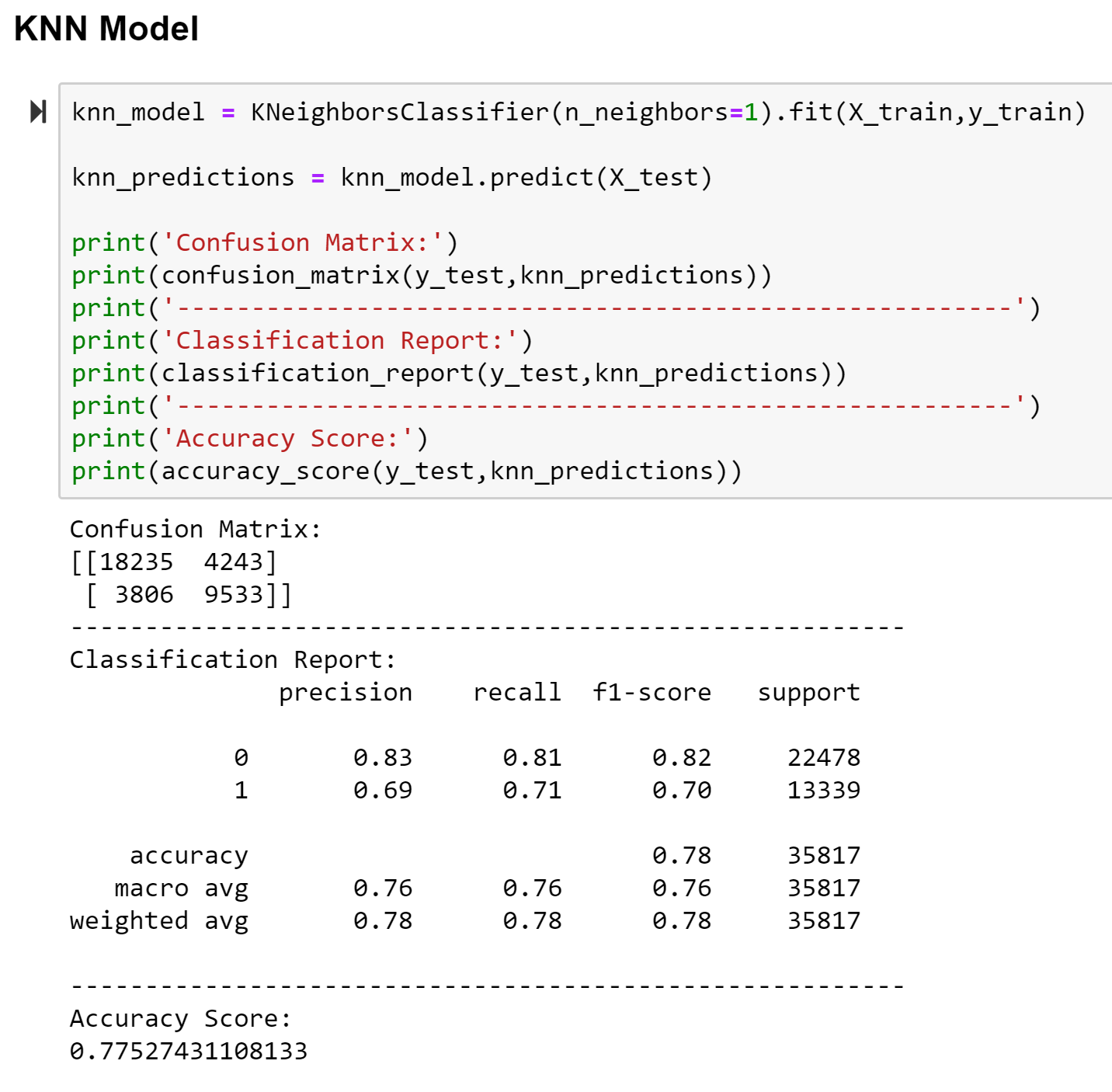
****

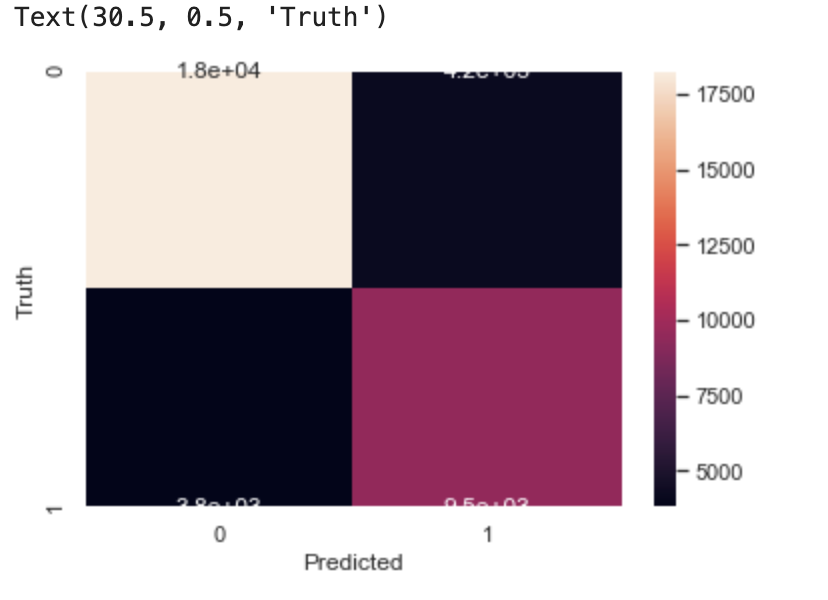
****

****

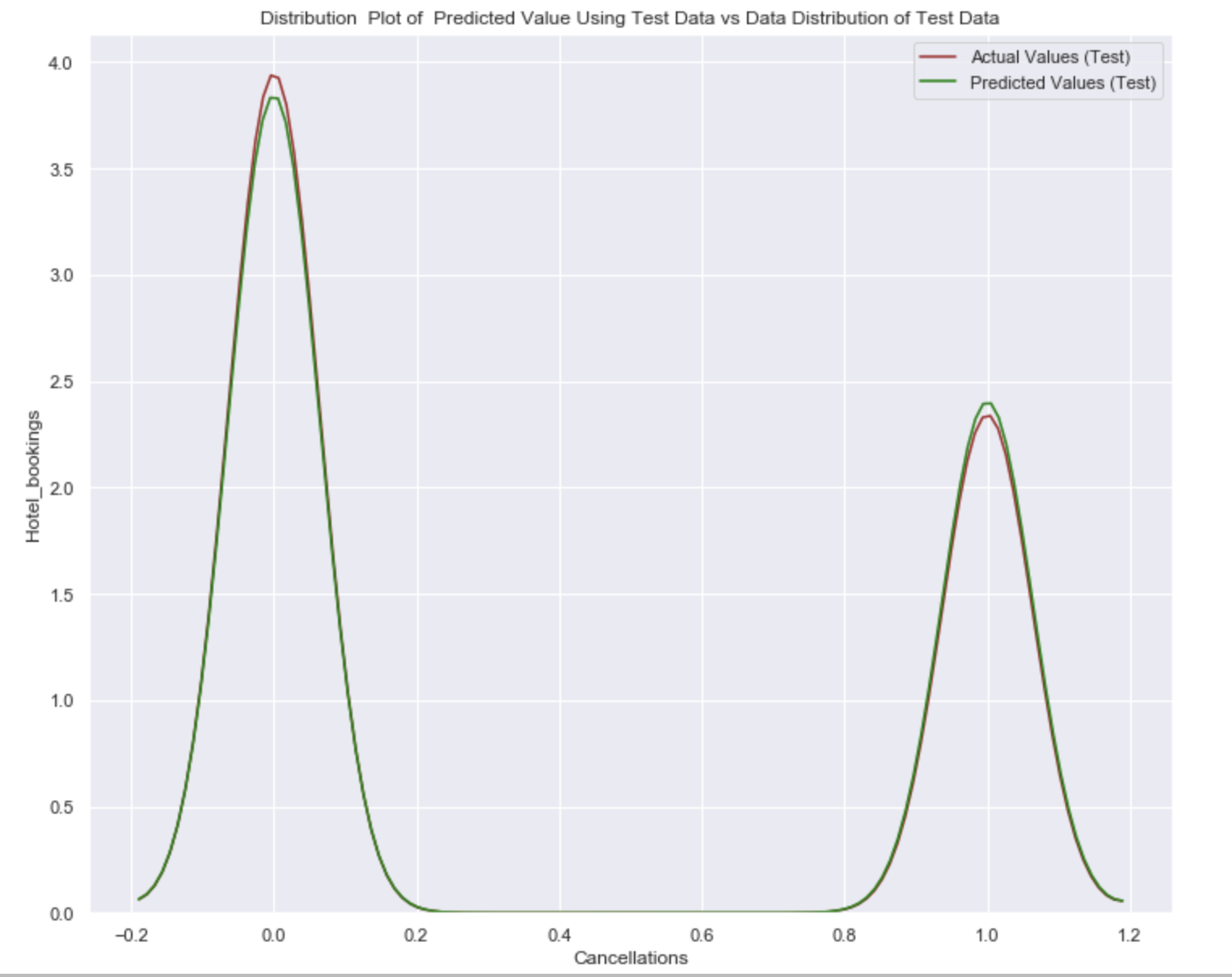
**KNN ALGORITHM MODEL:**

The model has an accuracy rate of 77.527%

****

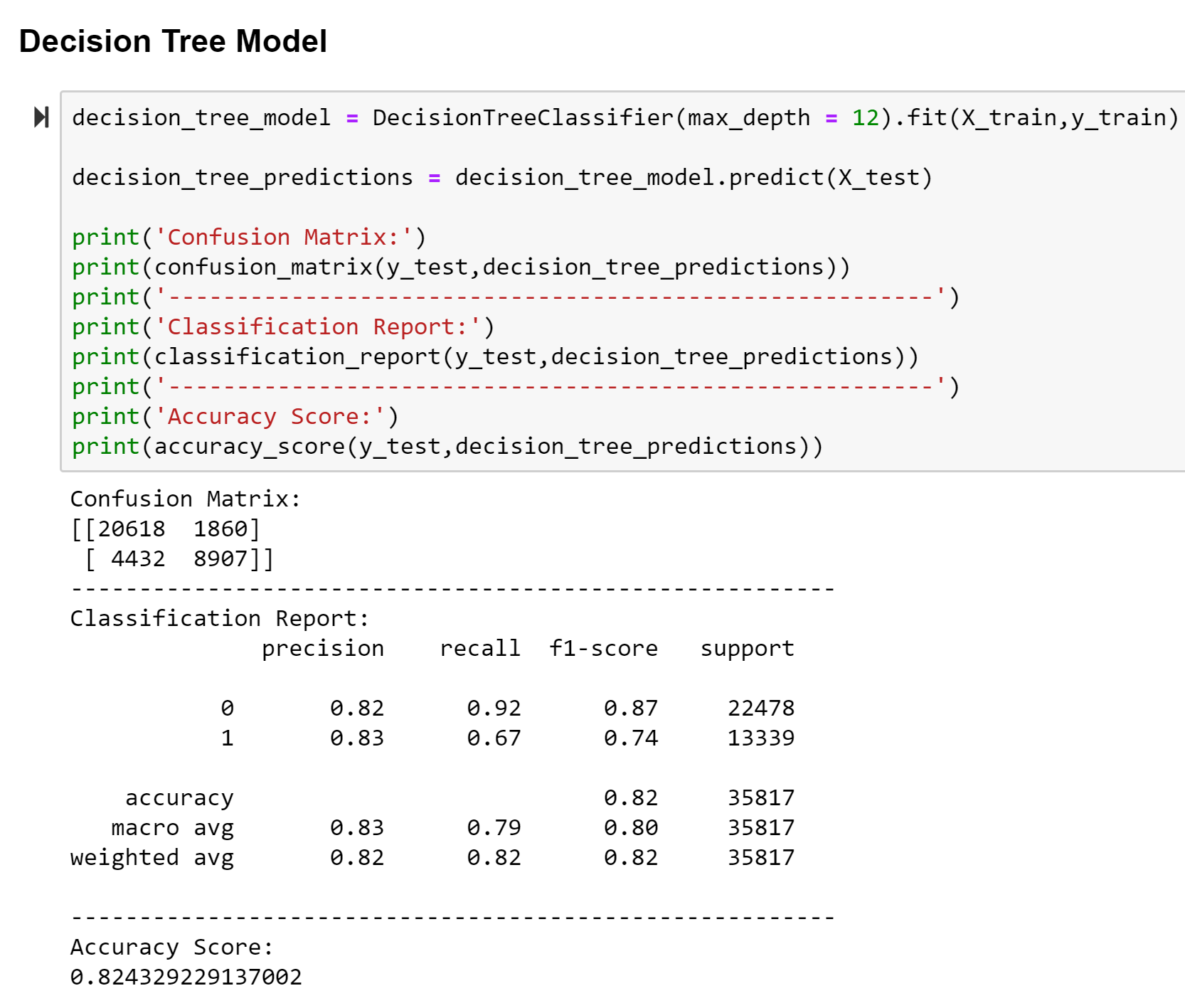
****

****

****

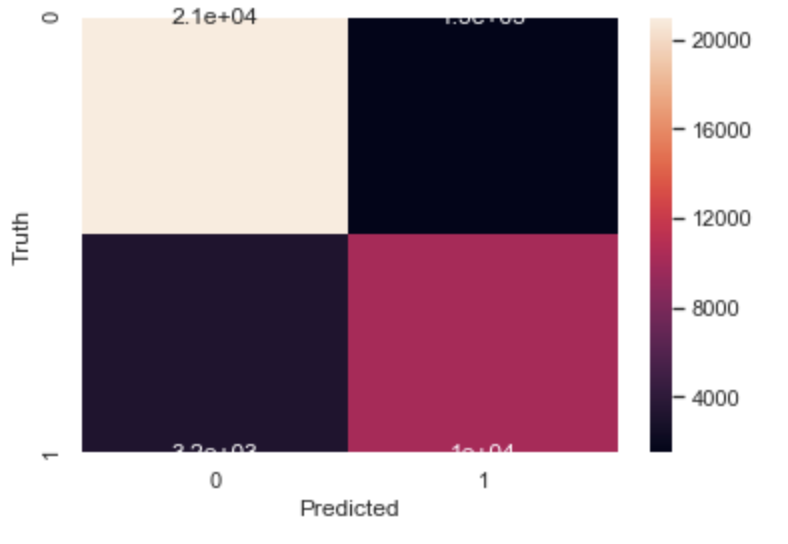
**SUPPORT VECTOR CLASSIFICATION MODEL:**

**DECISION TREE MODEL:**

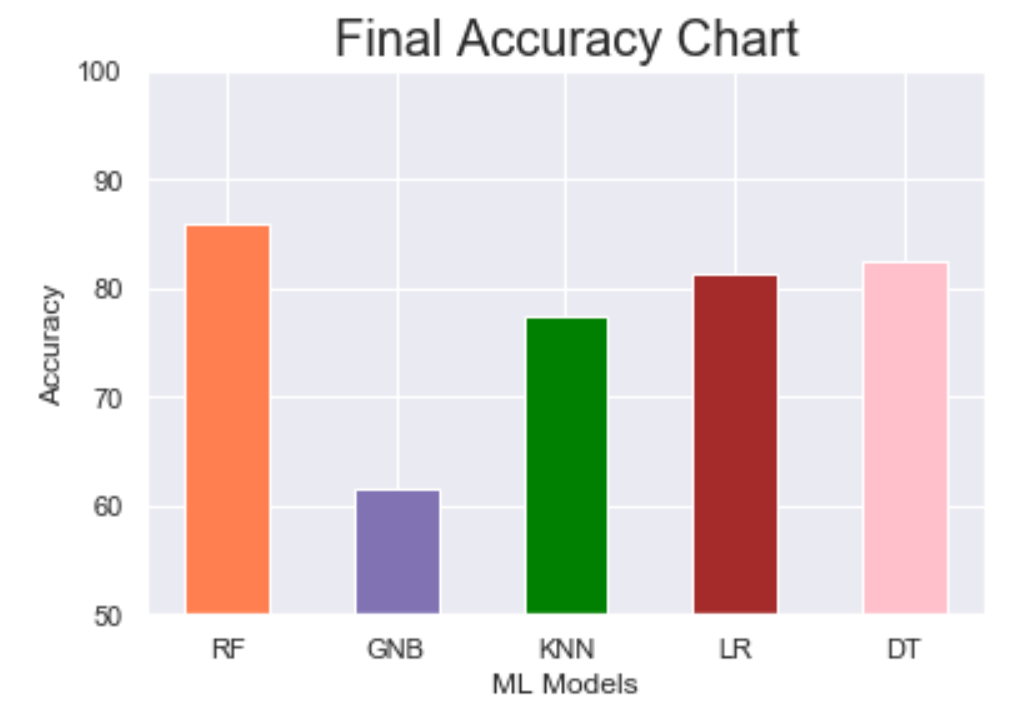
****

**RANDOM FOREST MODEL:**

This model results in an accuracy of 86.852% when we use 60 decision trees. The more decision trees we use, the accuracy gets better but it causes overfitting and increases the execution time as well.



**CONCLUSION:**



From the above chart, it is clear that Random Forest has the highest accuracy as it is the best algorithm to conduct predictive analysis because it tries to create an uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree.

**FUTURE SCOPE:**

**REFERENCES**

* [[1]](https://www.sciencedirect.com/science/article/pii/S2352340918315191#bbib1) N. Antonio, A. Almeida, L. Nunes, Predicting hotel bookings cancellation with a machine learning classification model, in: Proceedings of the 16th IEEE International Conference Machine Learning Application, IEEE, Cancun, Mexicopp. 1049–1054. doi:[10.1109/ICMLA.2017.00-11](https://doi.org/10.1109/ICMLA.2017.00-11), 2017.

[Google Scholar](https://scholar.google.com/scholar?q=N.%20Antonio,%20A.%20Almeida,%20L.%20Nunes,%20Predicting%20hotel%20bookings%20cancellation%20with%20a%20machine%20learning%20classification%20model,%20in:%20Proceedings%20of%20the%2016th%20IEEE%20International%20Conference%20Machine%20Learning%20Application,%20IEEE,%20Cancun,%20Mexicopp.%2010491054.%20doi:10.1109ICMLA.2017.00-11,%202017.)

* <https://towardsdatascience.com/logistic-regression-python-7c451928efee>
* <https://stackabuse.com/k-nearest-neighbors-algorithm-in-python-and-scikit-learn/>
* <https://www.datacamp.com/community/tutorials/naive-bayes-scikit-learn>
* <https://towardsdatascience.com/understanding-random-forest-58381e0602d2>
* <https://towardsdatascience.com/an-implementation-and-explanation-of-the-random-forest-in-python-77bf308a9b76>