**Hotel Reservations Project**

Abstract: This project provides an overview of Online Hotel Reservations, and we have a dataset that has columns belonging to anyone's booked room we want to know through your data, whether will he come or cancel this booking. From the data we have, we will build a model and train it on this data and make it can predict from person’s data whether will he come or cancel this booking. We work on this project to build model and develop by ensemble methods to create best model for this, and we can depend on it on work, let’s see how we do that.

**Introduction**

The online hotel reservation channels have dramatically changed booking possibilities and customers’ behavior. A significant number of hotel reservations are called off due to cancellations or no-shows. The typical reasons for cancellations include changes of plans, scheduling conflicts, etc. This is often made easier by the option to do so free of charge or preferably at a low cost, which is beneficial to hotel guests, but it is a less desirable and possibly revenue-diminishing factor for hotels to deal with. Can you predict if the customer is going to honor the reservation or cancel it? I am very excited to build a model that can solve this problem. [1]

**We will consider person’s data as input like: (Columns Description)**

[ no\_of\_adults, no\_of\_children, no\_of\_weekend\_nights, no\_of\_week\_nights, type\_of\_meal\_plan, required\_car\_parking\_space, room\_type\_reserved, lead\_time, arrival\_year, arrival\_month, arrival\_date, market\_segment\_type, repeated\_guest, no\_of\_previous\_cancellations, no\_of\_previous\_bookings\_not\_canceled, avg\_price\_per\_room, no\_of\_special\_requests]

**We will build three model for this project which are:**

* Logistic Regression
* Decision Tree Classification
* SVC

**Our model will train on data and predict this column:**

- booking\_status

**Dataset**

I got my dataset from [here](https://www.kaggle.com/datasets/ahsan81/hotel-reservations-classification-dataset?datasetId=2783627&sortBy=dateRun&tab=profile). I chose it because it is very simple and has very important columns and I love to start on it. There is example like my data [here](https://www.kaggle.com/code/aminizahra/hotel-booking-analysis/data).

**Columns Description:**

01- **Booking\_ID**: unique identifier of each booking

02- **no\_of\_adults**: Number of adults

03- **no\_of\_children**: Number of Children

04- **no\_of\_weekend\_nights**: Number of weekend nights (Saturday or Sunday) the guest stayed or booked to stay at the hotel

05- **no\_of\_week\_nights**: Number of weeknights (Monday to Friday) the guest stayed or booked to stay at the hotel

06- **type\_of\_meal\_plan**: Type of meal plan booked by the customer:

07- **required\_car\_parking\_space**: Does the customer require a car parking space? (0 - No, 1- Yes)

08- **room\_type\_reserved**: Type of room reserved by the customer. The values are ciphered (encoded) by INN Hotels.

08- **lead\_time**: Number of days between the date of booking and the arrival date

09- **arrival\_year**: Year of arrival date

10- **arrival\_month**: Month of arrival date

11- **arrival\_date**: Date of the month

12- **market\_segment\_type**: Market segment designation.

13- **repeated\_guest**: Is the customer a repeated guest? (0 - No, 1- Yes)

14- **no\_of\_previous\_cancellations**: Number of previous bookings that were canceled by the customer prior to the current booking

15- **no\_of\_previous\_bookings\_not\_canceled**: Number of previous bookings not canceled by the customer prior to the current booking

16- **avg\_price\_per\_room**: Average price per day of the reservation; prices of the rooms are dynamic. (In euros)

17- **no\_of\_special\_requests**: Total number of special requests made by the customer (e.g. high floor, view from the room, etc.)

18- **booking\_status**: Flag indicating if the booking was canceled or not.

**Then, after presenting the data, I will convert the data into two parts, the first is numeric and the second is categorical**.

Do data normalization, For numerical data, it was analyzed, and visualized it, and from this, all the data inside it was converted into data linked to each other, and it was re-given a number that helps make the model deal with it better.

As for the categorical data, it was converted from categorical data to numerical data so that the model can deal with it and identify it.

**The all size of our data is 36275 Rows And 19 Columns. And we split it to subset:**

* **Training-set: 25392 rows and 19 Columns**
* **Testing-set: 10883 Rows and 19 columns**

**Methods**

In this step I will explain ML algorithms that are used and Methods to Develop it and get best result. We used Logistic Regression, Decision Tree, SVC

**First Model:**

Logistic Regression is a statistical method for analyzing a dataset in which there are one or more independent variables that determine an outcome. The outcome is measured with a dichotomous variable (in which there are only two possible outcomes). It is used to predict a binary outcome (1 / 0, Yes / No, True / False) based on one or several predictor variables. The model estimates probabilities of the dependent variable given the values of the independent variables and maps them to a binary output using a logistic function. Logistic regression is widely used in fields such as biology, economics, and psychology. [2]

**Second Model:**

A Decision Tree Classifier is a type of algorithm used in machine learning for classification problems. It uses a tree-like model of decisions and their possible consequences. Each internal node in the tree represents a test on an attribute, each branch represents the outcome of the test, and each leaf node represents a class label. The topmost node in the tree is the root node, and it tests the most significant attribute to distinguish the target classes. The nodes lower in the tree test attributes that refine the target classes. The class label of the new data is determined by traversing the tree from the root to a leaf node, making a decision at each node based on the value of the attribute being tested. Decision Trees are simple to understand, interpret, and visualize and can handle both categorical and numerical data. [3]

**Third Model:**

Support Vector Classifier (SVC) is a type of algorithm used in machine learning for classification problems. It works by finding a hyperplane that best separates the data into different classes. The hyperplane is selected in such a way that it has the maximum margin, which means it has the largest distance between the closest data points of any two classes. These closest data points are called support vectors and define the position of the hyperplane. The SVC algorithm tries to find the hyperplane that separates the classes while maximizing the margin. In the case of non-linearly separable data, SVC can be used with a kernel trick, which maps the input data into a higher-dimensional space, where it becomes linearly separable. The SVC algorithm is effective for large datasets and for datasets with multiple features, but it can be time-consuming for large datasets. [4]

**And we chose from step 2 two way to develop our model:**

1. Develop an ensemble learning algorithm of the chosen classifiers based on the soft voting rule.
2. Develop an ensemble learning algorithm on one of the chosen classifiers based on bagging method.

**Soft Voting Classifiers** are a type of ensemble classifier that combines the predictions of several individual classifiers and predicts the class that has the highest average predicted probability. The predictions are combined using a soft voting rule, which means that the class with the highest average predicted probability is the predicted class. In contrast, with a hard voting rule, the predicted class would be the class that has the greatest number of votes from the individual classifiers. [5]

**Bagging Classifiers** are a type of ensemble classifier that uses the bagging method. In bagging, multiple models are trained on different random subsets of the training data, with replacement. The final prediction is made by combining the predictions of these models. This can reduce the variance of the predictions and make the model more robust to overfitting. [6]

**Experiments And Results**

In this step, we will discuss all the results that we obtained while working on this project, let’s see this. First, the data was analyzed, and it was discovered that 67.2% of the reservations in the data that were not canceled and were completed, and there are 32.8% of the data that were canceled as shown in the figure.

Chart, pie chart

Description automatically generated

After that, the data was divided, and the three models were trained on the training data, and each model was made to expect the output as shown below.

A picture containing table

Description automatically generated

**After making the model expect, we evaluate the model through:**

**accuracy\_score:** a performance metric used to evaluate the accuracy of a classifier.

**f1\_score:** It is the harmonic mean of precision and recall, two other commonly used performance metrics in machine learning.

**Precision Score**: the proportion of true positive predictions made by the classifier out of all the positive predictions made by the classifier.

**Recall Score**: the proportion of positive instances that are correctly classified as positive by the classifier.

**Classification** **Report**: a summary of the performance of a classifier.

**Confusion Matrix**: It is a summary of the number of correct and incorrect predictions made by the classifier for each class.

**ROC Curve**: (Receiver Operating Characteristic curve) is a graphical representation of the performance of a binary classifier system as its discrimination threshold is varied.

**Result of Train & Test three model on data**

There are all results of evaluate three model and show below:Table

Description automatically generatedChart, line chart

Description automatically generated

Table

Description automatically generatedGraphical user interface, text, email

Description automatically generatedFrom above we see that all model work will put maybe work best if we use any way from ensembling methods so we will develop my models by two ways. let’s do it using bagging with Three model.

**Result of Train & Test bagged three model on data.**

After building a model based on the packaging method, each user model was trained and tested. This was the result.

Table

Description automatically generated

Table

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Chart, line chart

Description automatically generated

As we noticed from the above, the decision tree was developed through bagging and made it expect better than the previous one.

**Result of Train & Test Voting Classifier for three model on data**

Then we tried to develop our model with the soft voting rule for Classification so started to build model with three model and Voting Classifier then we get after train model on data and make it predict.

A picture containing table

Description automatically generated

A picture containing table

Description automatically generatedText, letter

Description automatically generated

A picture containing graphical user interface

Description automatically generated

This model works well but here is a bit of an overfit.

**Conclusion**

We will compare analysis with three models we use and three model with begging we noticed as shown below.

Table

Description automatically generated

**we abled to finish all require like:**

First, we can choose dataset from Kaggle. Then Cleaning data and remove duplicated, then analyzing and visualization my data, then prepare data to model and split it to training set and test set, chose three classifiers, there are Logistic Regression, Decision Tree Classifier and SVC and show accuracy for them, Then choose Decision Tree Classifier to develop it with Bagging and show accuracy for it. Then applied on three classifiers Soft Voting Method to develop them and shown accuracy for them, from all Result above, we did comparative analysis for all model we used it, The Result from above we noticed **Bagged Decision Trees** are best model for our data.

**References and Contributions**

[1] <https://www.kaggle.com/datasets/ahsan81/hotel-reservations-classification-dataset>.

[2]<https://en.wikipedia.org/wiki/Logistic_regression#:~:text=Logistic%20regression%20is%20a%20statistical%20model%20that%20in,logistic%20model%20%28a%20form%20of%20binary%20regression%20%29>..

[3] <https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm#:~:text=Decision%20Tree%20is%20a%20Supervised%20learning%20technique%20that,rules%20and%20each%20leaf%20node%20represents%20the%20outcome>..

[4] <https://monkeylearn.com/blog/introduction-to-support-vector-machines-svm/#:~:text=To%20sum%20up%3A%201%20A%20support%20vector%20machine,better%20to%20just%20stick%20to%20a%20linear%20kernel>..

[5] <https://www.geeksforgeeks.org/ml-voting-classifier-using-sklearn/>

[6] <https://www.geeksforgeeks.org/ml-voting-classifier-using-sklearn/>