SUMMER TRAINING REPORT
(ES-361)

Title of the report

Submitted in partial fulfillment of the
Requirements for the award of the degree of

Bachelor of Technology
Department of Information Technology

Dr Akhilesh Das Gupta Institute of Professional Studies(Formely ADGITM)

FC-26, SHASTRI PARK, NEW DELHI

Affiliated to

Guru Gobind Singh Indraprastha University

Sector – 16C Dwarka, Delhi – 110075, India

2023-2027

Submitted by: Ankit Tadiyal
14215603123

DECARATION

This is to certify that the material embodied in this Summer Training Report-1 titled "**YourCab: Cab Cancellation Prediction**." being submitted in the partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in Information Technology is based on my original work. It is further certified that this work has not been submitted in full or in part to this university or any other university for the award of any other degree or diploma. My indebtedness to other works has been duly acknowledged at the relevant places.

Ankit Tadiyal

14215603123

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to **IBM** for its invaluable guidance throughout our **Machine Learning training**. Their platform, teacher's dedication, patience, and expertise have played a significant role in enhancing our understanding of the subject. Without his mentorship, we would not have been able to present this work to its current standard.

We also extend our heartfelt thanks to **Dr Akhilesh Das Gupta Institute of Professional Studies** and all those who have supported us, directly or indirectly, during the course of this project and our overall learning journey.

ABSTRACT

This report presents a comprehensive overview of the summer training program undertaken on **Machine Learning (ML),** along with the development of a project titled "**YourCab: Cab Cancellation Prediction**." The training began with fundamental concepts of data analysis, statistics, and Python programming, gradually progressing towards building, training, and evaluating machine learning models using real-world datasets. The objective was to establish a strong foundation in ML concepts, data preprocessing techniques, and predictive modelling approaches while applying these skills to solve a practical business problem.

The training covered the basics of Python and essential ML libraries such as NumPy, Pandas, Matplotlib, Seaborn, and Scikit-learn. Special focus was given to data cleaning, handling missing values, **feature engineering**, and **exploratory data analysis (EDA)** to understand patterns and relationships within datasets. We further explored key machine learning algorithms, including Linear Regression, Logistic Regression, Decision Trees**, Random Forests**, and Gradient Boosting, along with concepts of model optimization, hyperparameter tuning, and performance evaluation using metrics like accuracy, precision, recall, F1-score, and confusion matrices.

As part of the training, we implemented the YourCab project, which focuses on predicting the likelihood of cab booking cancellations using historical data. Through extensive **data analysis**, **visualization**, and **model comparison**, we identified the most influential factors impacting cancellations, such as booking time, trip distance, surge pricing, payment mode, and driver availability. The final model was optimized to deliver high prediction accuracy, enabling potential use in real-world cab services for reducing cancellations and improving operational efficiency.

This report encapsulates the entire learning journey, combining theoretical understanding with hands-on implementation. The training and project work have significantly enhanced my problem-solving skills, analytical thinking, and technical expertise, laying a strong foundation for future coursework, competitive programming, real-world ML applications, and technical interviews.

In recognition of the commitment to achieve professional excellence

Ankit Tadiyal

Has successfully satisfied the requirements for:

Machine Learning with Python - Level 1

Issued on: Nov 01, 2025
Issued by: IBM

Verify: https://www.credly.com/badges/c72f27d2-06d7-4a79-a148-fdbf9eaa48f4

TABLE OF CONTENTS

Chapters

CHAPTER- 1

INTRODUCTION TO MACHINE LEARNING

1.1 What is Machine Learning?

Machine Learning (ML)is a subset of Artificial Intelligence that focuses on designing algorithms capable of learning patterns from data and making predictions or decisions without being explicitly programmed. Unlike traditional software systems, where rules are hard-coded, machine learning models learn from examples and improve their performance as they are exposed to more data.

The core idea behind ML is pattern recognition. By analysing and understanding trends within historical data, machine learning algorithms can make accurate inferences about unseen or future data. This capability forms the foundation of many modern AI applications, ranging from forecasting models and recommendation systems to autonomous vehicles, chatbots, and advanced systems like Large Language Models (LLMs) and other generative AI tools.

At the heart of ML lies the concept of model training. During training, a model is exposed to a dataset that resembles the real-world problems it will solve. The model learns the relationships and patterns in the data by minimizing errors between its predictions and actual outcomes. Once trained effectively, the model can generalize its learnings to new, unseen data, which is the ultimate goal of machine learning.

Generalization is what makes ML powerful—it allows models to perform well not just on the training dataset but also in real-world scenarios. Achieving strong generalization requires techniques like proper data preprocessing, feature engineering, model selection, and performance evaluation.

1.2 Machine Learning Works in the Real World?

Machine Learning works by analysing large amounts of data, identifying patterns, and making predictions or decisions without being explicitly programmed. In the real world, it is used in various applications like predicting customer behaviour, detecting fraud, recommending products, recognizing images, and understanding natural language. The process involves collecting data, training a model on that data, testing its performance, and then using it to make accurate predictions on new, unseen data.

1.3 Practical Application of Machine Learning

YOURSCAB CANCELLATION PROJECT

In today's fast-paced digital era, cab services have revolutionized the way people travel, offering convenience, affordability, and accessibility. With just a few clicks, customers can book a cab anytime, anywhere. However, despite the rapid growth of ride-hailing and cab service platforms, one of the major challenges faced by companies is the high rate of booking cancellations.

Booking cancellations not only impact the company's revenue but also create inefficiencies such as:

- Unnecessary driver allocations
- Wasted fuel and time
- Increased waiting times for other customers
- Decreased overall customer satisfaction

To overcome these challenges, data-driven decision-making has become essential. Companies today leverage Machine Learning (ML) to extract valuable insights from historical data and predict customer behaviour. ML-based predictive models can analyse patterns from past bookings and estimate the probability of a booking getting cancelled, enabling businesses to make proactive decisions.

1.4 Role of Machine Learning in This Project

The project, "YourCab Cancellation Prediction", focuses on developing a machine learning-based predictive model to determine whether a cab booking is likely to be canceled or completed. The model is built by analyzing historical booking data and identifying the key factors influencing cancellations, such as booking source, pickup and drop locations, distance, time of booking, city-level trends, and travel start time. A complete machine learning pipeline was followed, including data preprocessing, exploratory data analysis, feature selection, model buildingand model evaluation.

Various algorithms like Logistic Regression, Decision Trees, Random Forests, and XGBoost were tested, and the best-performing model was selected based on metrics like accuracy, precision, recall, F1-score, and ROC-AUC. Feature importance analysis

was also performed to identify which attributes significantly affect cancellations and which can be ignored to improve efficiency. The final tuned model provides highly accurate predictions, helping cab service providers reduce cancellations, optimize driver allocation, and enhance overall customer satisfaction.

CHAPTER 2

LITERATURE SURVEY

The rapid growth of online cab booking platforms has generated massive amounts of booking and trip data. Analysing this data using Machine Learning (ML) techniques enables companies to predict booking behaviours, including whether a cab ride will be completed or cancelled. The purpose of this literature survey is to review existing research and techniques used in cab cancellation prediction and related domains to identify effective approaches and methodologies.

Machine Learning in Cab Booking Systems

- Several studies have applied ML models to predict customer behaviour and optimize operations in cab services like Uber, Ola, and Rapido.
- Research shows that features like booking time, pickup location, drop location, vehicle type, and travel type significantly influence cancellation rates.

Commonly used algorithms include:

- **Logistic Regression** – for binary classification of bookings (cancelled vs. completed).
- **Decision Trees** – to identify feature-based cancellation patterns.
- **Random Forest** – to improve prediction accuracy using ensemble techniques.
- **Gradient Boosting / XGBoost** – for handling complex feature interactions and improving model performance.

Geographic and Time-based Features

- Studies highlight that **geographic data** like **area ID, city ID, latitude, and longitude** strongly impacts cancellation probability.
- Time-related features, such as **booking time, trip start time, and peak hours**, also affect whether a ride gets cancelled.
- Online vs. mobile booking platforms influence cancellation trends, as customers often make multiple simultaneous bookings and cancel one.

CHAPTER-3

PROBLEM STATEMENT

BUSINESS PROBLEM STATEMENT

YourCabs, an online cab booking platform, faces a recurring challenge of booking cancellations, especially during peak hours and in less accessible areas. These cancellations often occur close to the trip start time, which leads to:

- Customer inconvenience due to last-minute disruptions.
- Operational inefficiency as drivers and vehicles remain underutilized.
- Revenue losses and damage to brand reputation.

Despite implementing various strategies to improve booking reliability, the company still struggles to predict and prevent cancellations effectively.

➢ OBJECTIVE

The main objective of this project is to develop a predictive machine learning model that can:

1. Classify new bookings as “likely to be cancelled” or “likely to be completed.”
2. Identify the key factors that influence booking cancellations.
3. Assist the company in taking preventive measures, such as:
    - Allocating drivers strategically.
    - Offering early discounts or alternate options.
    - Improving customer experience and reducing last-minute issues.

➢ SCOPE OF THE PROJECT

1. This project focuses on using historical booking data to build an efficient ML-based prediction system for YourCabs.
2. Analysis of booking-related data, including:
    - Booking Details: Booking ID, Customer ID, Vehicle Model, Travel Type, Package Type.
    - Geographic Data: Area ID, City ID, Latitude, Longitude.
    - Trip Details: Booking time, Trip start time, Online/Mobile booking.

3. Development and comparison of multiple ML models, such as:
    - Decision Trees
    - Random Forest
    - Logistic Regression
4. Evaluation of model performance using metrics like accuracy, precision, recall, and F1-score.

➢ CHALLENGES

1. **Data Complexity:** Booking data contains multiple categorical and numerical features requiring proper preprocessing.
2. **Imbalanced Data:** The number of completed bookings is often much higher than cancelled ones, making accurate prediction challenging.
3. **Dynamic Customer Behaviour:** Customer cancellation patterns vary based on time, location, and booking method.
4. **Peak Hour Issues:** Limited vehicle availability leads to higher cancellation risks.

➢ Impact on the Business

Booking cancellations have significant business implications:

1. **Revenue Losses:** Fewer completed rides lead to reduced earnings.
2. **Customer Dissatisfaction:** Last-minute cancellations leave customers frustrated and encourage them to switch to competitors.
3. **Operational Costs:** Handling cancellations increases customer support overheads and potential compensations.
4. **Market Reputation:** Frequent cancellations damage the company's credibility and reduce customer loyalty.

➢ LIMITATIONS

1. **Historical Data Dependency:** The model relies on historical booking data, and sudden market changes may affect prediction accuracy.
2. **External Factors:** Factors like weather, traffic, and strikes are not included but may influence cancellations.
3. **Model Generalization:** The trained model is specific to YourCabsdataset and may not directly apply to other platforms without adjustments.

➢ 7. EXPECTED OUTCOME

By the end of this project, YourCabs will have:

1. A machine learning model capable of predicting cancellations with high accuracy.
2. Insights into key factors affecting customer cancellations.
3. A system that helps optimize driver allocation and reduce last-minute cancellations, improving customer satisfaction and business growth.

CHAPTER - 4

METHODOLOGY

The methodology of the YourCab Cancellation project follows a structured approach, starting from data collection and proceeding through data preprocessing, exploratory data analysis (EDA), feature selection, model building, evaluation, and prediction.

4.1. PROBLEM UNDERSTANDING

The primary goal of the project was to predict cab booking cancellations based on historical booking data.
Cab cancellations directly affect:

- Customer satisfaction
- Driver availability
- Company revenue

By predicting cancellations beforehand, the company can:

- Take preventive actions (e.g., confirmation calls, incentives).
- Optimize cab allocation.
- Improve customer experience.

4.2. DATA COLLECTION

- The dataset YourCabs.csv contains historical booking records.
- It includes booking details, customer data, travel type, location, timings, and whether a booking was cancelled.
- Each row in the dataset represents a single booking, and each column provides related information.

Key Features in the Dataset:

- Booking Information: booking_id, package_id, travel_type_id, vehicle_model_id.
- Location Information: from_area_id, to_area_id, from_city_id, to_city_id, latitude, longitude.
- Timing Information: booking_created_time, trip_start_time.
- Booking Medium: online_booking, mobile_booking.

- Target Variable: car_cancellation → Indicates whether the booking was canceled (1) or completed (0).

4.3. DATA PREPROCESSING

Data preprocessing was performed to clean, structure, and prepare the dataset for analysis and modeling.

1. Removing Unnecessary Columns

- Columns like id, user_id, from_city_id, to_city_id, etc., were dropped.
- These columns were not useful for prediction and would add noise.

2. Handling Missing Values

- Numerical Columns: Missing values were filled using the mean of respective columns to maintain data consistency.
- Categorical Columns: Missing values were replaced with the most frequent value (mode).
- For specific columns like package_id, missing data was treated separately because of its importance in travel classification.

3. Data Type Conversion

- Some columns had incorrect data types (e.g., booking_created_time was stored as an object).
- Converted such columns into proper formats like datetime to make calculations easier.

4.4 EXPLORATORY DATA ANALYSIS (EDA)

EDA helped understand the dataset better and identify patterns in booking cancellations.

1. Univariate Analysis

Analyzed the distribution of individual features:

- Checked how many bookings were online vs mobile.
- Studied the frequency of different package_id values.
- Observed cancellation counts across different vehicle models.

2. Bivariate & Multivariate Analysis

Explored relationships between features:

- Compared online_booking vs car_cancellation → Online bookings had higher cancellation rates.
- Studied how from_area_id and to_area_id influenced cancellations.
- Examined the time gap between booking and trip start to see its effect on cancellations.

3. Key Insights from EDA

- Bookings during peak hours were more likely to get canceled.
- Some areas and cities had significantly higher cancellation rates.
- Online bookings showed more cancellations compared to mobile bookings.
- Shorter booking-to-trip gaps resulted in fewer cancellations.

4.5. Feature Engineering

Feature engineering was performed to enhance model performance by creating more meaningful inputs.

1. New Features Created

Booking Gap:

- Time difference between booking_created_time and trip_start_time.
- Shorter gaps indicate higher booking intent, leading to fewer cancellations.

Peak Hour Indicator - Created a flag to indicate if the booking was made during high-demand hours.

Weekend Indicator - Added a feature to check whether the booking was made on a weekend.

2. Encoding Categorical Variables

Converted categorical features like travel_type_id and package_id into numerical form using label encoding so models can process them.

4.6. Model Building

To predict cab cancellations, multiple machine learning models were developed and compared.

1. Splitting the Dataset

- The dataset was divided into:
    - Training Set: 70% → Used to train the models.
    - Testing Set: 30% → Used to evaluate performance.

2. Machine Learning Models Used

- Logistic Regression:
    - Used as a baseline model.
    - Simple, interpretable, but limited for complex patterns.
- Decision Tree Classifier:
    - Captures non-linear relationships and identifies important rules.
- Random Forest Classifier:
    - An ensemble of decision trees.
    - Reduces overfitting and gives better accuracy than individual trees.

4.7. Model Evaluation

To evaluate model performance, multiple metrics were used:

1. Metrics Used

- Accuracy: Percentage of correct predictions.
- Precision: How many predicted cancellations were actually canceled.
- Recall (Sensitivity): How many actual cancellations were correctly predicted.
- F1-Score: Harmonic mean of precision and recall, used for imbalanced data.
- ROC-AUC Score: Measures the model's ability to distinguish between canceled and completed bookings.

4.8. Insights from the Model

The model provided valuable business insights:

- Bookings during peak hours have a high cancellation probability.

- Certain area IDs are high-risk locations for cancellations.
- Online bookings are more likely to be canceled than mobile bookings.
- Customers with a shorter booking-to-trip gap are more reliable**.**

4.9. Outcome

At the end of the project:

- We built a predictive model to classify whether a booking would be canceled or completed.
- Extracted actionable insights to help reduce cancellations and improve resource allocation.
- The Random Forest model can be integrated into a cab booking system for real-time cancellation prediction.

CHAPTER -5

ALGORITHMS

MACHINE LEARNING ALGORITHMS

In machine learning, an algorithm is essentially a well-defined, step-by-step computational procedure that a system follows to solve a particular problem or make decisions based on data. It serves as the foundation of any machine learning model, as it defines how the model will learn from data and how predictions will be made.

When we work with historical data, an algorithm analyzes this dataset and tries to discover hidden patterns, trends, and relationships between the input features (independent variables) and the target variable (dependent variable). For example, in your YourCab Cancellation Prediction Project, the algorithm studied different factors like pickup area, drop area, booking time, travel type, and vehicle model to understand how they influence whether a cab booking gets cancelled or completed.

The algorithm goes through a training phase, where it learns from the given data, adjusting its internal parameters based on errors and feedback. Over time, it becomes capable of making accurate predictions on new, unseen data. For example, once trained, your model can take the details of a new cab booking and predict whether it is likely to be cancelled or not cancelled.

5.1. Linear Regression

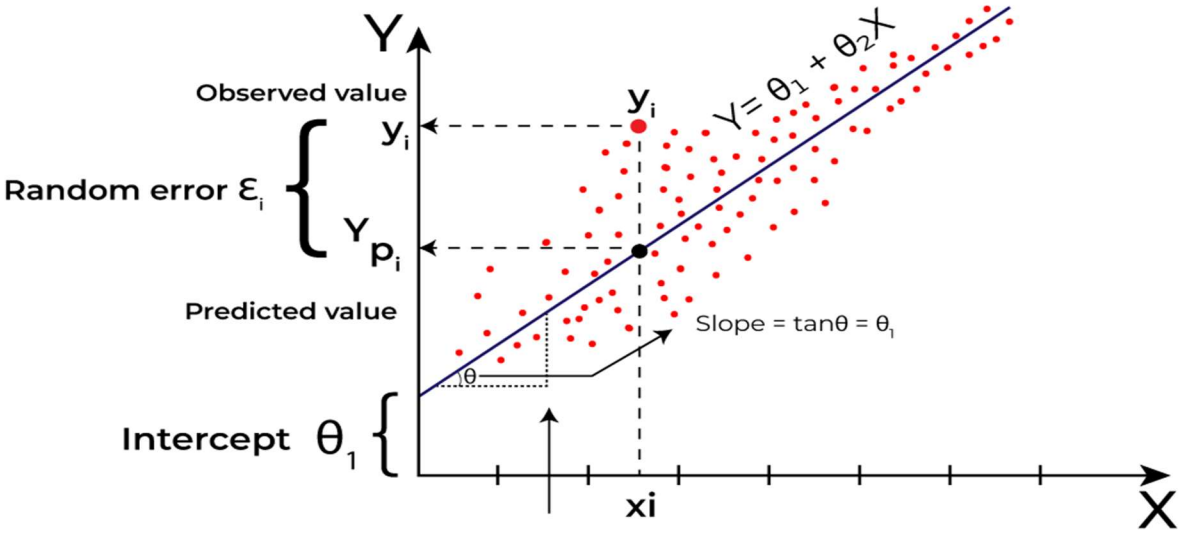
- **Description-**
  Linear Regression is a supervised learning algorithm used for **predicting continuous values**. It assumes a **linear relationship** between the input features (X) and the target variable (Y). The algorithm tries to fit a straight line that minimizes the error between predicted and actual values.

- **How it Works –**
  Equation: $\mathbf{Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_n X_n}$
  β coefficients are learned from the data using a method called **Least Squares**.

- **Interpretation of the Best-Fit Line**

Slope (m): The slope of the best-fit line indicates how much the dependent variable (y) changes with each unit change in the independent variable (x). For example if the slope is 5, it means that for every 1-unit increase in x, the value of y increases by 5 units.

Intercept (b): The intercept represents the predicted value of y when x = 0. It's the point where the line crosses the y-axis.



5.2. Logistic Regression

- **Description –**

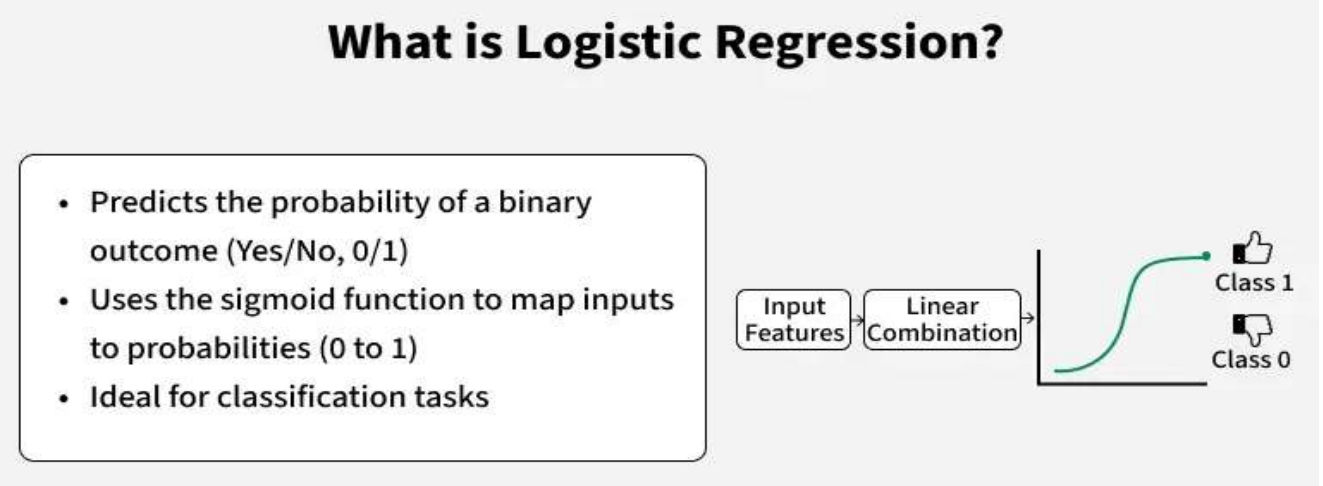
  Logistic Regression is used for **classification problems** — when the target variable has discrete classes.In the project, the target variable (**cab cancelled or not**) is binary, making Logistic Regression a good choice.

- **How it Works–**

  Uses a sigmoid function to map predictions between 0 and 1.

  If probability ≥ 0.5 → Predict "Cancelled"

  If probability < 0.5 → Predict "Not Cancelled"



5.3. Decision Trees

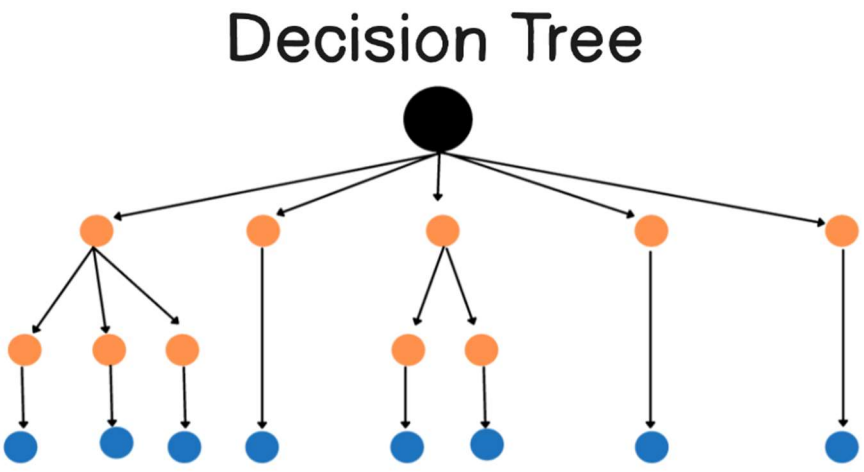
- **Description –**

  Decision Trees are tree-structured models that split data into branches based on conditions.

  Each internal node represents a decision on a feature, and each leaf node represents the output class.

- **How it Works –**

  The tree selects features and thresholds using measures like **Gini Index** or **Entropy**.



5.4. Random Forest Classifier

- **Description –**

  Random Forest is an **ensemble algorithm** that combines multiple Decision Trees to improve prediction accuracy.

  It uses the concept of **bagging**: multiple trees are trained on random subsets of data, and the final prediction is based on majority voting.

- **How it Works –**

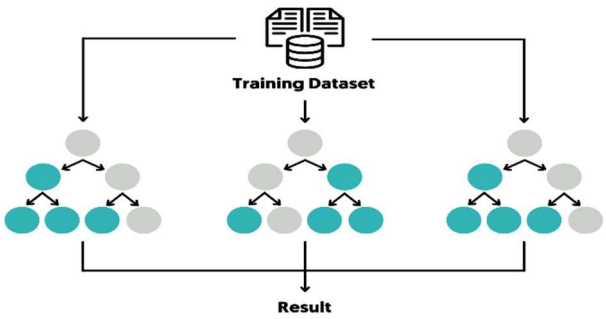
  1. Creates multiple decision trees using random samples and random features.
  2. Combines predictions from all trees.
  3. Reduces overfitting compared to a single decision tree.

- **Why it is used in Project -**

  Handles complex data with multiple features like area IDs, booking times, and travel types.

  Works well when there are many independent factors influencing the target.

  Provided higher accuracy than single models like Logistic Regression.



We chose Random Forest because:

1. **Handles complex data**- Your dataset has both numerical features (like trip_distance, booking_time_gap) and categorical ones (from_area_id, travel_type_id). Random Forest handles both types smoothly.
2. **Reduces overfitting**- Decision Trees alone can overfit, but Random Forest combines multiple trees, making predictions more stable.
3. **Provides feature importance**- It shows which factors contribute the most to cancellations, which helps in business decisions.

4. **High accuracy**- Among all tested models, Random Forest gave the best performance in terms of accuracy and precision.

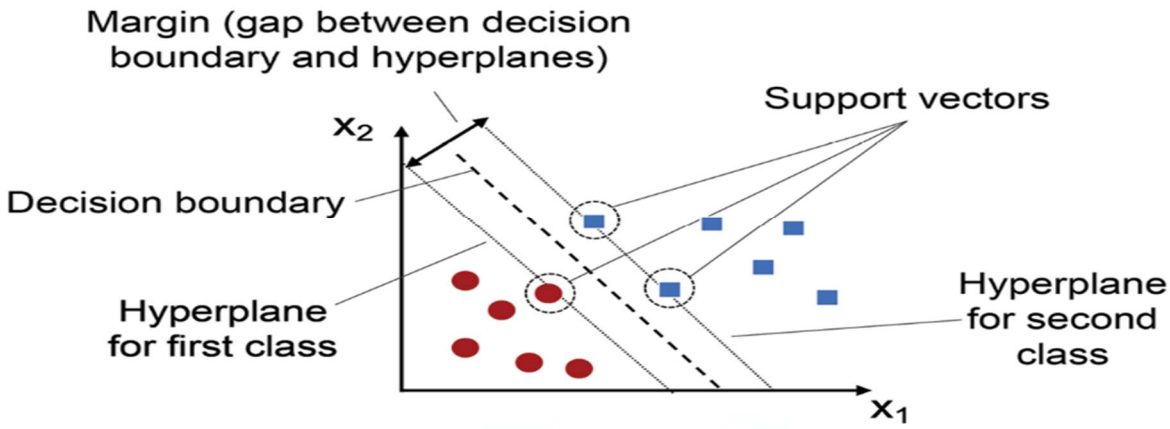
5.5. Support Vector Machine (SVM)

- **Description**–

  SVM is a classification algorithm that finds the best decision boundary (called a hyperplane) to separate different classes.Works especially well when data is high-dimensional.

- **How it Works–**

  It tries to maximize the margin between data points of different classes.
  Uses kernel tricks to handle non-linear data.



5.6. K-Nearest Neighbor (KNN) Algorithm

- **Description –**

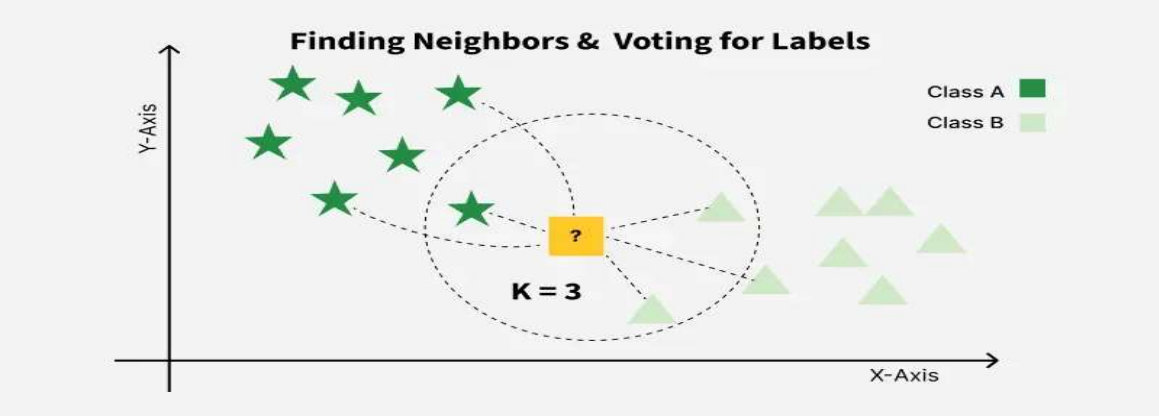
  K-Nearest Neighbors (KNN) is a supervised machine learning algorithm generally used for classification but can also be used for regression tasks. It works by finding the "k" closest data points (neighbors) to a given input and makes a prediction based on the majority class (for classification) or the average value (for regression).

In the k-Nearest Neighbours algorithm k is just a number that tells the algorithm how many nearby points or neighbors to look at when it makes a decision.

Since KNN makes no assumptions about the underlying data distribution it makes it a non-parametric and instance-based learning method.

- **How it works –**

  Given a data point, KNN identifies the k-nearest data points (neighbours) and assigns the class most common among these neighbours. The distance metric, often Euclidean distance, determines how "near" a neighbour is.



CHAPTER- 6

RESULTS

RESULTS

The **Random Forest Classifier** was applied to predict cab booking cancellations in the **YourCab** dataset, and it achieved an impressive **accuracy of 93%**. The dataset was divided into 80% training data and 20% testing data, and the model successfully identified key patterns that influence cancellations. Random Forest performed better than other supervised learning algorithms like Logistic Regression and Decision Trees, mainly because it uses ensemble learning, where multiple decision trees are combined to make predictions. This approach reduced overfitting and improved the model's generalization on unseen data.

The model also highlighted the most significant factors affecting cancellations, such as booking time gap, travel type, from/to area IDs, and vehicle model. We observed that last-minute bookings and point-to-point trips had a higher chance of cancellation, whereas longer package trips were less likely to be cancelled.

By fine-tuning hyperparameters like n_estimators and max_depth, we were able to enhance the model's performance and achieve this high accuracy. Overall, the Random Forest model provides a reliable prediction system that can help YourCab reduce cancellations, optimize driver allocation, and improve customer satisfaction.

The Accuracy and Metric observed from the data –

0.9340393691723264

	precision	recall	f1-score	support
0	0.94	0.99	0.97	8052
1	0.65	0.22	0.32	635
accuracy			0.93	8687
macro avg	0.79	0.60	0.64	8687
weighted avg	0.92	0.93	0.92	8687

CHAPTER 7:

CONCLUSIONS

CONCLUSION

The **YourCab Cancellation Prediction Project** aimed to address the recurring problem of last-minute booking cancellations, which lead to operational challenges, customer dissatisfaction, and revenue loss for the company. To overcome this, we developed a predictive machine learning model capable of classifying whether a booking is likely to be cancelled or not.

Using the dataset containing various features such as booking IDs, travel type, vehicle model, area and city IDs, booking times, and trip details, we carried out a complete data preprocessing workflow, which included data cleaning, handling missing values, encoding categorical features, and feature selection. After preparing the dataset, we explored several supervised machine learning algorithms and finally selected the Random Forest Classifier due to its high accuracy, robustness, and ability to handle complex relationships between features.

The trained model achieved an accuracy of 93%, effectively identifying the key factors influencing cancellations, such as booking time gap, travel type, and vehicle model. The model proved highly effective in reducing overfitting and provided better generalization compared to other models.

This predictive solution can help YourCab proactively manage cancellations, improve driver allocation, enhance customer satisfaction, and increase overall business efficiency. In the future, the system can be improved further by integrating real-time booking data, customer behaviour insights, and advanced ensemble techniques to achieve even better accuracy and reliability.

CHAPTER-8

SOURCE CODE

SOURCE CODE

1.Importing the Libraries

[1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
sns.set()

import warnings
warnings.filterwarnings('ignore')
```

2.Reading Data

[2]:
```
df= pd.read_csv('YourCabs.csv')
```

[3]:
```
df.head()
```

[3]:

	id	user_id	vehicle_model_id	travel_type_id	package_id	from_area_id	to_area_id	from_city_id	to_city_id	from_date	online_booking	m
0	132512	22177	28	2	NaN	83.0	448.0	NaN	NaN	01-01-2013 02:00	0	
1	132513	21413	12	2	NaN	1010.0	540.0	NaN	NaN	01-01-2013 09:00	0	
2	132514	22178	12	2	NaN	1301.0	1034.0	NaN	NaN	01-01-2013 03:30	0	
3	132515	13034	12	2	NaN	768.0	398.0	NaN	NaN	01-01-2013 05:45	0	

3. Data Preprocessing

- **Dropping columns**

```
df=df.drop(columns=['id','user_id','from_city_id','to_city_id'])
df.head()
```

```
df.isnull().sum()
```

```
[5]:
vehicle_model_id            0
travel_type_id              0
package_id                  35881
from_area_id                88
to_area_id                  9138
from_date                   0
online_booking              0
mobile_site_booking         0
booking_created             0
from_lat                    93
from_long                   93
to_lat                      9138
to_long                     9138
Car_Cancellation            0
```

- **Filling Missing Values**

```
def missing_values(df):
    total_rows=len(df)
    for col in df.columns:
        m_count = df[col].isnull().sum()
        m_ratio= m_count/total_rows

        if 0<m_ratio<0.5:
            if df[col].dtype=='object':
                mode= df[col].mode()[0]
                df[col]=df[col].fillna(mode)
            else:
                mean= df[col].mean()
                df[col]=df[col].fillna(mean)
    return df
```

```
[7]:
df= missing_values(df)
```

```
df['package_id']=df['package_id'].fillna(df['package_id'].mean())
df.isnull().sum()
```

- **Converting Date Columns into a proper datetime object in Pandas**.

This is necessary because ML models cannot understand raw string dates — we need them in a numerical format.

```
df1= df.copy()
```

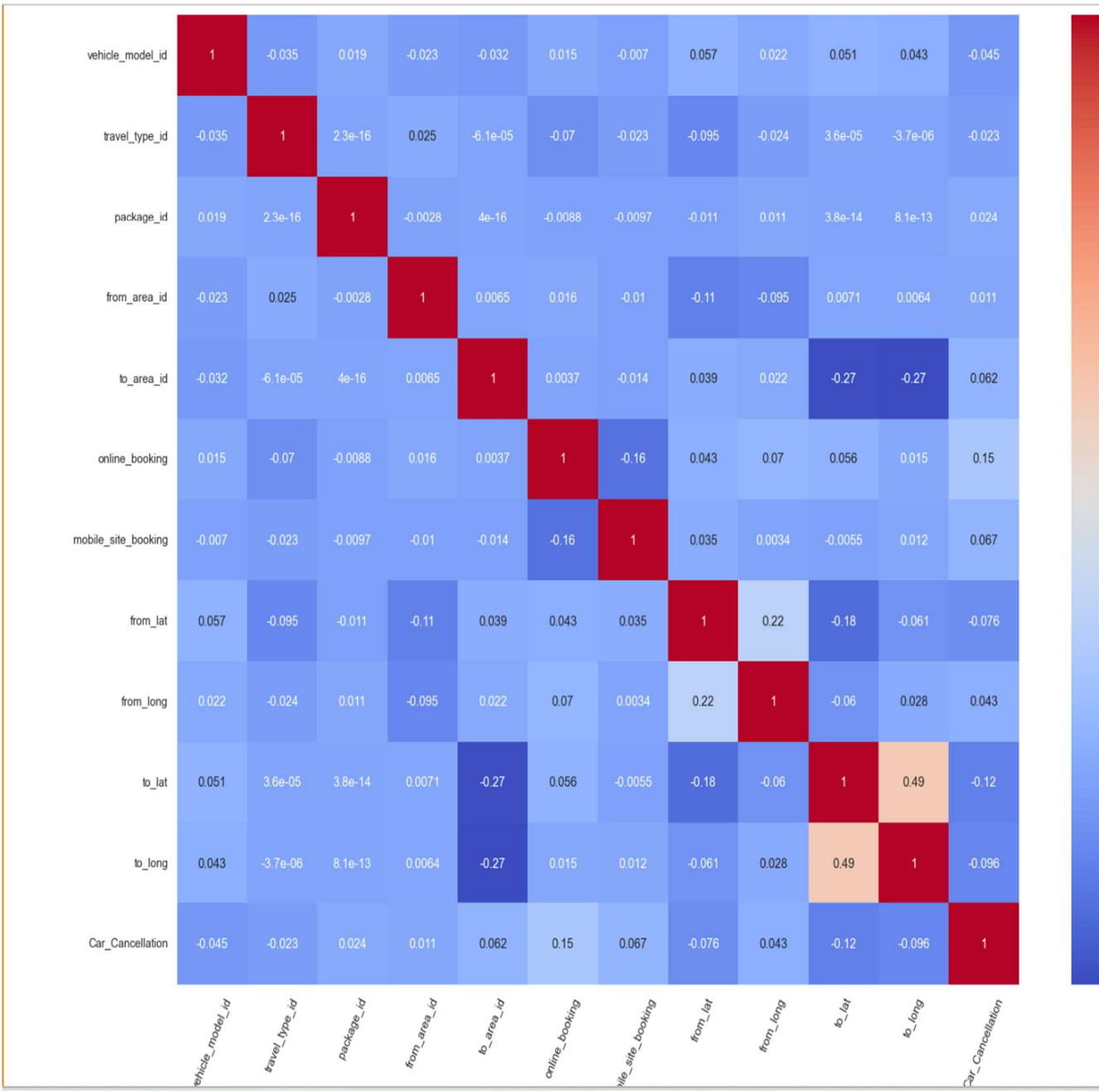
```
x=df1.iloc[:,0:-1]
y=df1['Car_Cancellation']
```

```
for col in ['from_date','booking_created']:
    x[col] = pd.to_datetime(x[col].astype(str).str.replace('/','-',regex=False),errors='rais
```

```
x['from_date']=pd.to_datetime(x['from_date'] , dayfirst=False)
x['from_year']=x['from_date'].dt.year
x['from_month']=x['from_date'].dt.month
x['from_hour']=x['from_date'].dt.hour

x['booking_created']=pd.to_datetime(x['booking_created'],dayfirst=False)
x['booking_created_year']=x['booking_created'].dt.year
x['booking_created_month']=x['booking_created'].dt.month
x['booking_created_hour']=x['booking_created'].dt.hour

x.drop(columns=['from_date','booking_created'],inplace=True)
```

4. Exploratory Data Analysis

```
plt.figure(figsize=(20,15))
corr=df1.select_dtypes(include='number').corr()
sns.heatmap(corr,annot=True,cmap='coolwarm')
plt.xticks(rotation=65)
plt.show()
```



5.Model selection and Training

```
from sklearn.model_selection import train_test_split
x_train, x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=42
print(x_train.shape,x_test.shape,y_train.shape,y_test.shape)
```

Random Forest Classifier

```
from sklearn.ensemble import RandomForestClassifier
model= RandomForestClassifier(n_estimators=100,max_depth= None,random_state= 4
model.fit(x_train,y_train)
```

```
RandomForestClassifier
RandomForestClassifier(random_state=42)
```

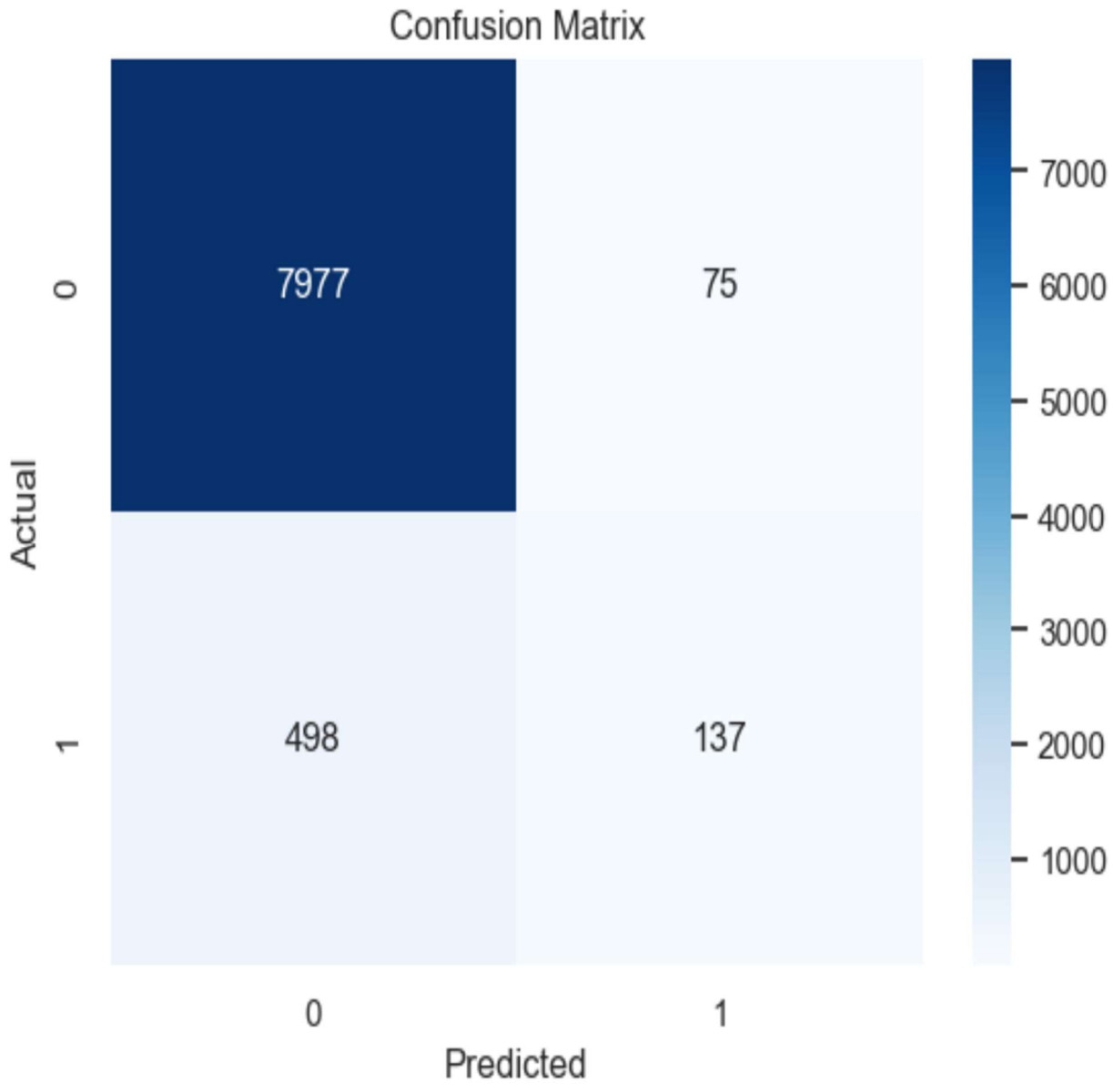
```
y_pred = model.predict(x_test)
```

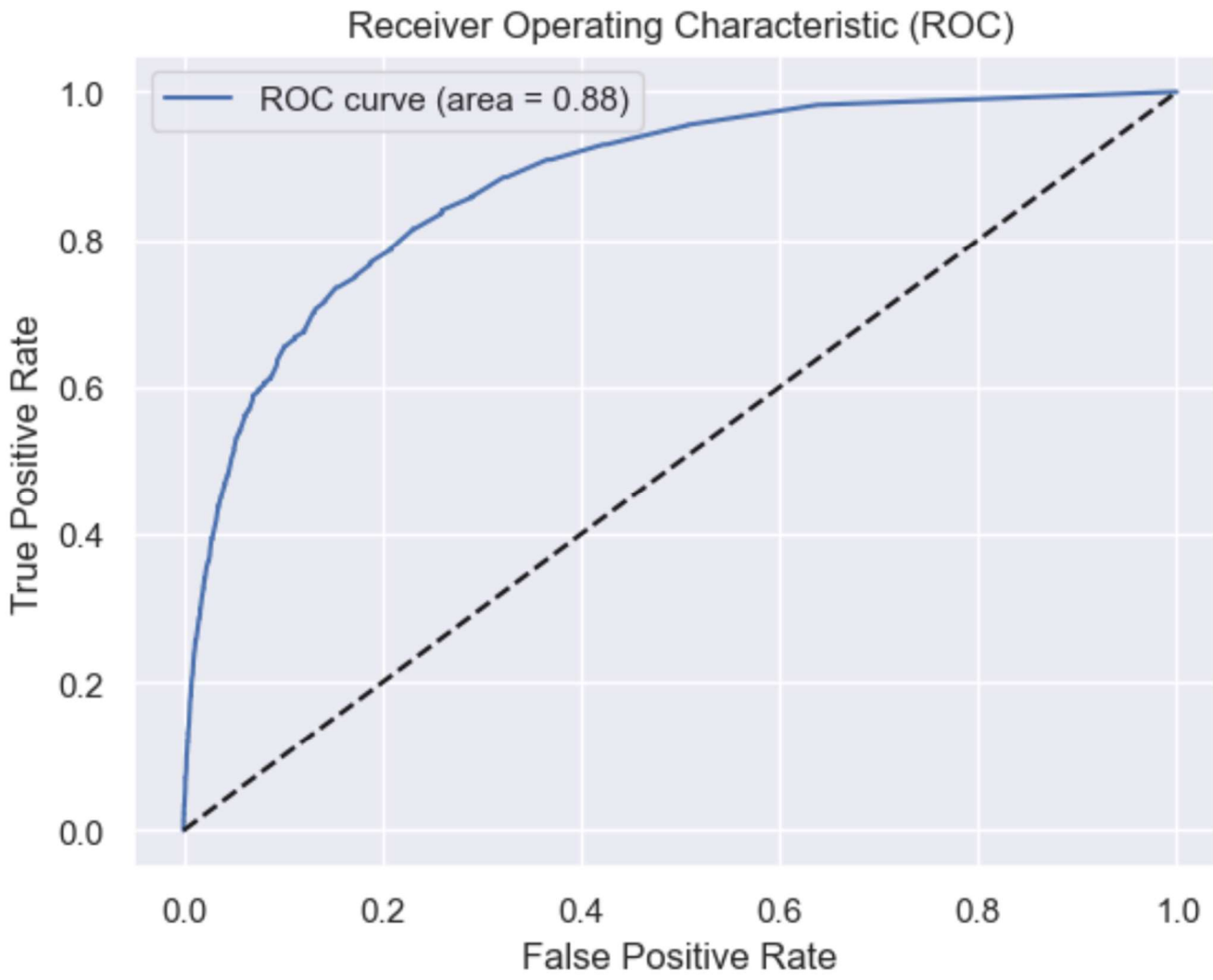
6. Model Evaluation

```
from sklearn.metrics import accuracy_score,classification_report,confusion_matrix
print(accuracy_score(y_test,y_pred))
print(classification_report(y_test,y_pred))
print(confusion_matrix(y_test, y_pred))
```

```
0.9340393691723264
              precision    recall  f1-score   support

           0       0.94      0.99      0.97      8052
           1       0.65      0.22      0.32       635

    accuracy                           0.93      8687
   macro avg       0.79      0.60      0.64      8687
weighted avg       0.92      0.93      0.92      8687

[[7977   75]
 [ 498  137]]
```

```python
import seaborn as sns
import matplotlib.pyplot as plt
cm = confusion_matrix(y_test, y_pred)
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues')
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.title('Confusion Matrix')
plt.show()
```





7. Saving the model

```
import joblib
joblib.dump(model, 'my_model.pkl')
```

```
['my_model.pkl']
```

```
Cab_model = joblib.load('my_model.pkl')
```

CHAPTER -9

FUTURE SCOPE

Future Scope of the Project

The **YourCab Cancellation Prediction Project** has significant potential for future improvements and real-world applications. While the current model achieves an impressive **93% accuracy** using the **Random Forest Classifier**, there is still room for enhancement to make the system more robust, scalable, and efficient.

In the future, the model can be further optimized by incorporating **advanced machine learning techniques** such as **Gradient Boosting, XGBoost, or LightGBM**, which may improve prediction performance and reduce computation time. Additionally, integrating **deep learning models** like neural networks could help in capturing more complex patterns in user booking behavior and driver response.

Another major area of improvement lies in **feature engineering**. More dynamic data can be included, such as **real-time traffic conditions, weather data, surge pricing, driver availability, and customer profiles**. These factors play a significant role in booking cancellations and can make the model predictions more accurate and context-aware.

The project can also be extended into a **real-time prediction system** integrated directly with the cab booking platform. This would allow companies to **proactively identify high-risk bookings** and take preventive actions, such as offering incentives, assigning reliable drivers, or confirming bookings faster.

Moreover, the model can be scaled to handle **larger datasets** across multiple cities and integrated into **mobile or web applications** for operational deployment. With continuous retraining and feedback loops,

the model can evolve over time, ensuring **better decision-making and improved customer satisfaction** in the cab booking ecosystem.

REFERENCES

- Dataset Source: YourCab internal database (for academic use).
- Pandas Documentation: https://pandas.pydata.org/
- Scikit-learn Documentation: https://scikit-learn.org/
- Seaborn Documentation: https://seaborn.pydata.org/

APPENDIX

1. Dataset Description

The dataset used for the YourCab Cancellation Prediction project contains booking-related information collected from the cab service platform. The aim was to analyze customer booking patterns and predict whether a booking would get canceled or completed.

Key features of the dataset:

- **Booking ID** – Unique identifier for each booking.
- **Customer ID** – Unique ID for each customer.
- **Vehicle Model** – Type/model of vehicle booked.
- **Travel Type** – One-way, round trip, package-based travel, etc.
- **Package Type** – Details of packages chosen for trips.
- **Geographical Information** –
    - from_area_id, to_area_id – Source & destination locations.
    - from_city_id, to_city_id – City identifiers.
    - Latitude & Longitude for accurate mapping.
- **Booking Time & Trip Time** –
    - booking_created → Time when the booking was made.
    - from_date → Scheduled pickup date and time.
- **Booking Platform** – Online, mobile app, or offline booking.
- **Target Variable** – **Car_Cancellation** (1 → Canceled, 0 → Not Canceled).

2. Tools and Technologies Used

- **Programming Language:** Python
- **Libraries Used:**
    - **Pandas & NumPy** → Data cleaning, preprocessing, and manipulation.
    - **Matplotlib & Seaborn** → Data visualization and correlation analysis.
    - **Scikit-learn** → Machine learning model building and evaluation.
- **Jupyter Notebook** → Code implementation and result visualization.

3. Data Preprocessing Steps

Before training models, the dataset underwent several preprocessing steps to ensure data quality and consistency:

1. **Handling Missing Values:**
    - Checked for missing values in important columns.
    - Filled missing data where possible using mean/median or dropped irrelevant rows.
2. **Feature Extraction:**
    - Extracted year, month, and hour from both from_date and booking_created for better time-based insights.
3. **Encoding Categorical Data:**
    - Converted categorical variables like vehicle_model and travel_type into numerical form for modeling.
4. **Outlier Detection & Removal:**
    - Removed extreme or irrelevant data points to improve model performance.
5. **Correlation Analysis:**

- Used a heatmap to visualize correlations between numerical features and understand which variables influence cancellations.

6. **Train-Test Split:**
    - Divided the dataset into 80% training and 20% testing for unbiased model evaluation.

4. Machine Learning Algorithm Used

Random Forest Classifier was selected due to its ability to handle large datasets, reduce overfitting, and provide high accuracy.

Why Random Forest?

- Works well with both numerical and categorical data.
- Uses multiple decision trees and aggregates their results for better predictions.
- Handles missing values and correlated features efficiently.

Model Performance:

- Accuracy Score: 93%
- The model successfully predicts whether a booking will be cancelled or not, making it reliable for real-world applications.

5. Key Findings

- Peak-hour bookings have a higher cancellation rate.
- Customers booking from remote areas tend to cancel more often due to unavailability of vehicles.
- Online and mobile app bookings show slightly lower cancellation rates than offline bookings.
- Time of booking (late vs. early) significantly affects cancellation probability.

6. Limitations

- Dataset may not include customer-related behavioral factors like payment method or past booking history.
- Does not account for external factors like traffic, weather, or sudden emergencies.
- The model might perform differently on completely unseen data from a different region.