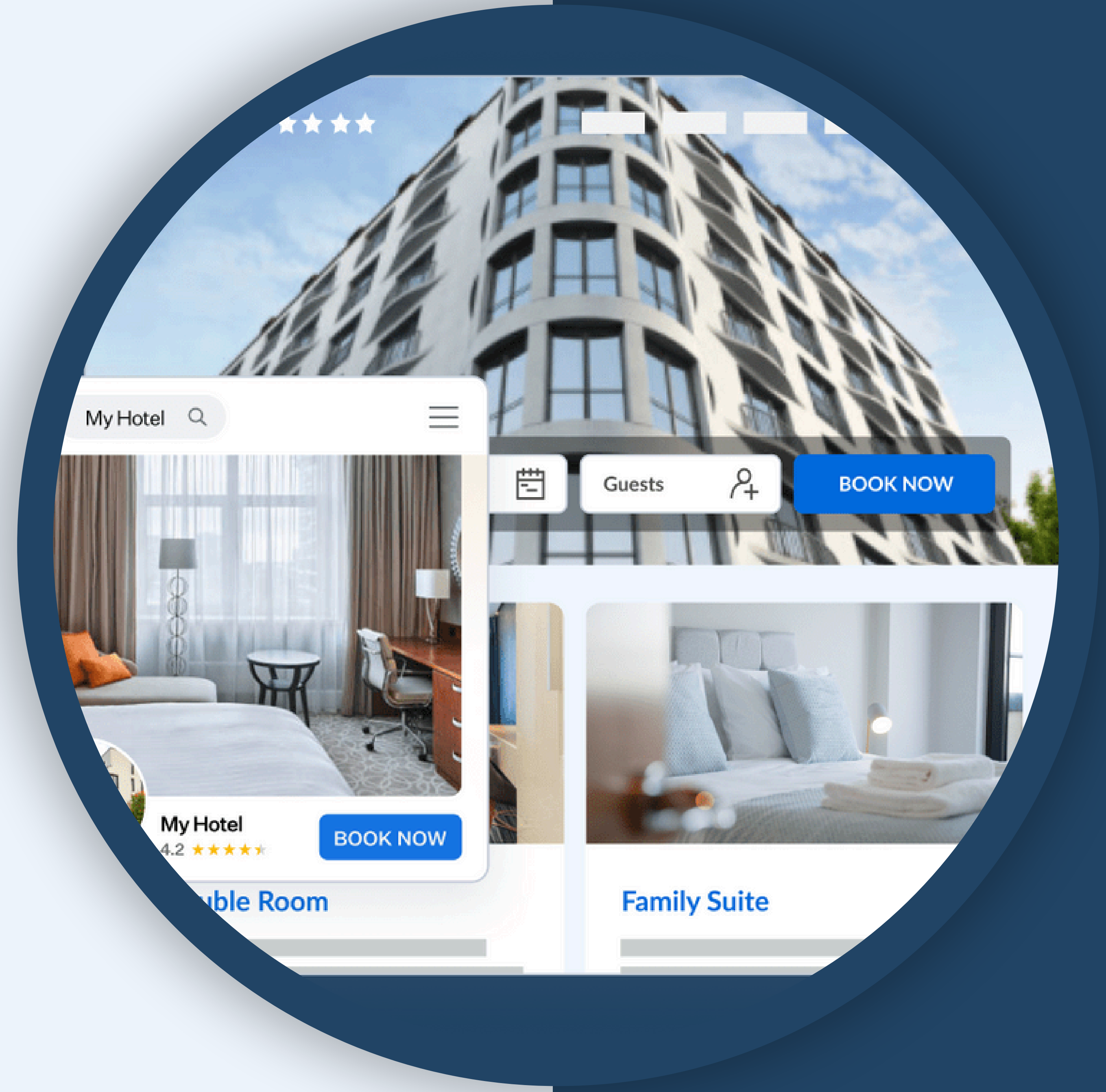
EDA

HOTEL RESERVATIONS

By: Asmaa Ibrahim



Content

- Dataset Overview
- Data Preprocessing
- Data Visualization
- Train the model

Dataset Overview

Dataset contains 36,285 records and 17 columns related to hotel booking details, such as:

GUEST DETAILS

Number of adults, children, meal plan, special requests.

BOOKING INFORMATION

Lead time, room type, market segment, booking status.

STAY DETAILS

Weekend and weekday nights, car parking, repeated guests.

FINANCIAL ASPECT

Average price.

Data Preprocessing

01 **CHECK FOR DUPLICATES DATA**

02 ***Check for Missing Values***

03 Convert date columns to datetime format

04 Extract new features

05 Encode Categorical Features

06 Handle Outliers using IQR Method

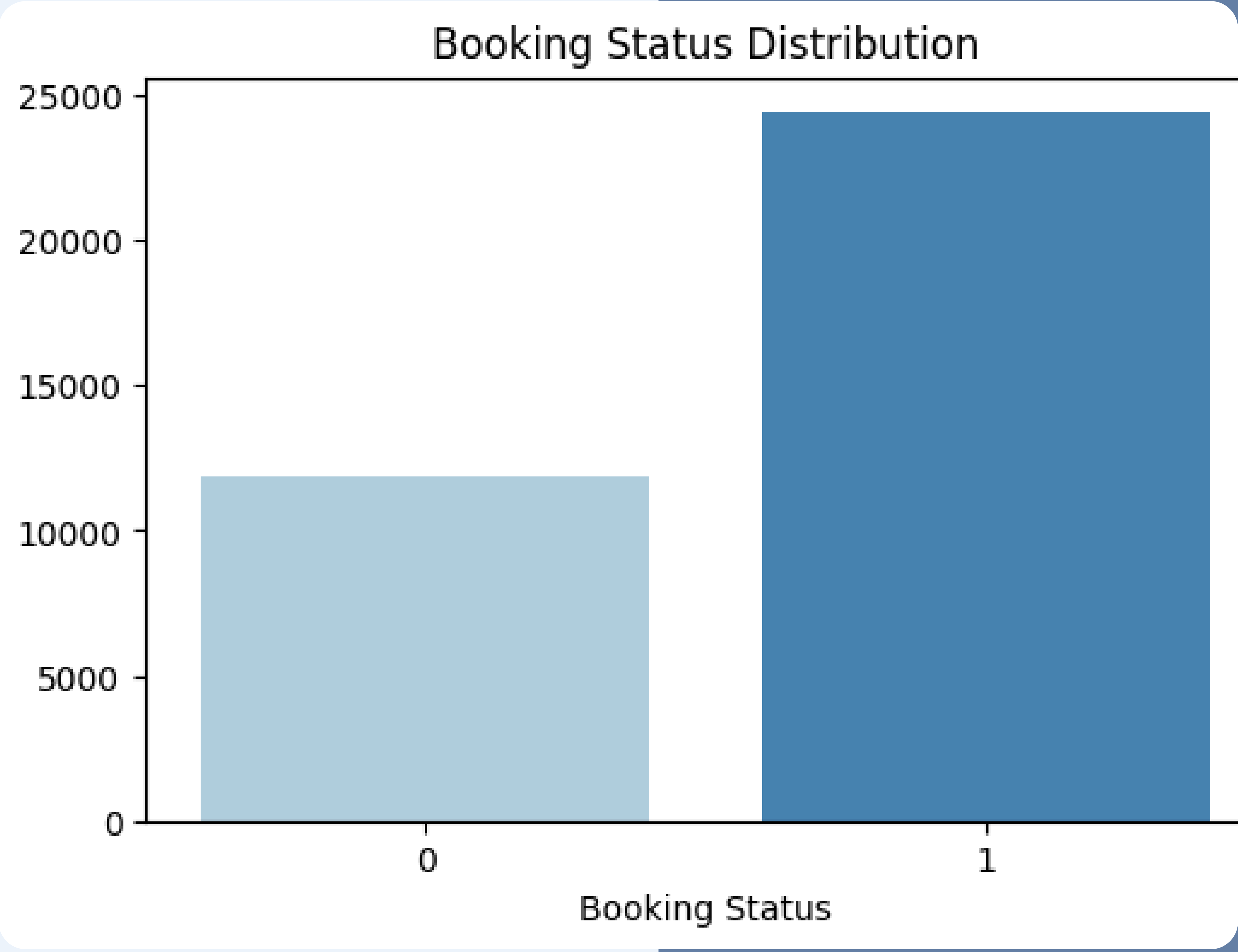
07 Drop Unnecessary Columns

Data Visualization

BOOKING STATUS

1

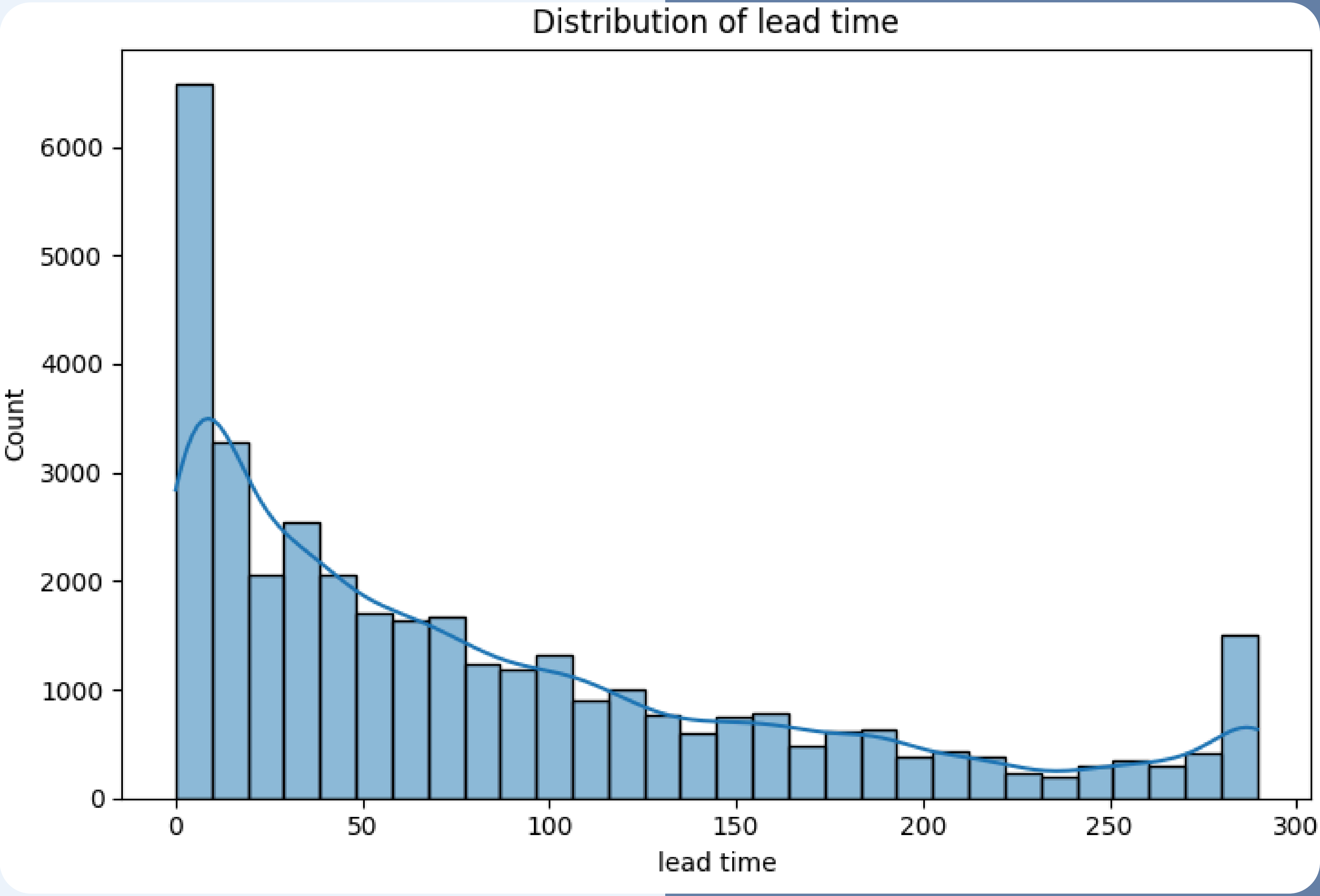
- There are more non-canceled bookings than canceled ones.
- Understanding what factors influence cancellations can help improve hotel retention.



LEAD TIME DISTRIBUTION

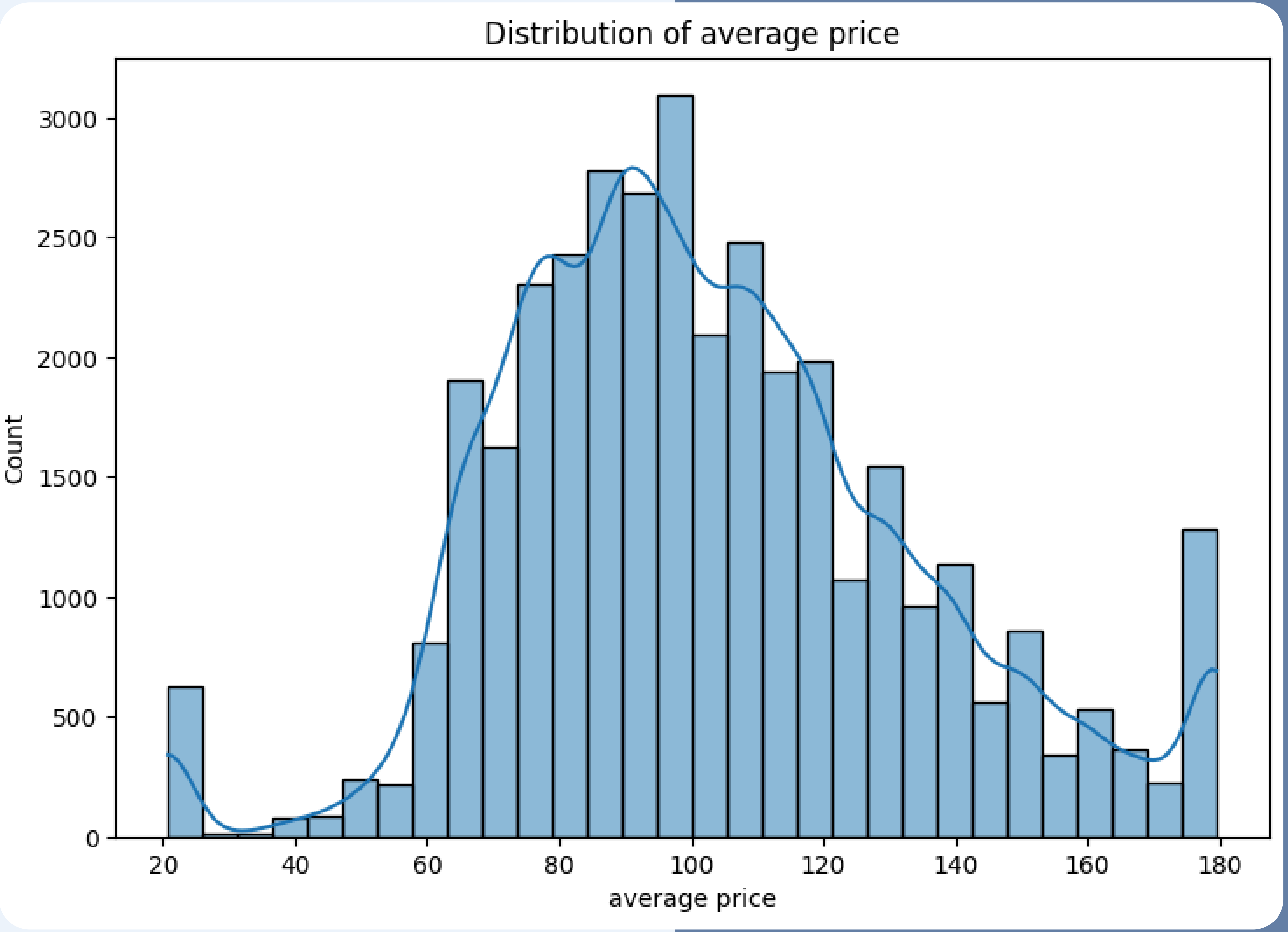
2

Most bookings happen with a short lead time, but there are still a few long lead-time reservations.



AVERAGE PRICE PER BOOKING 3

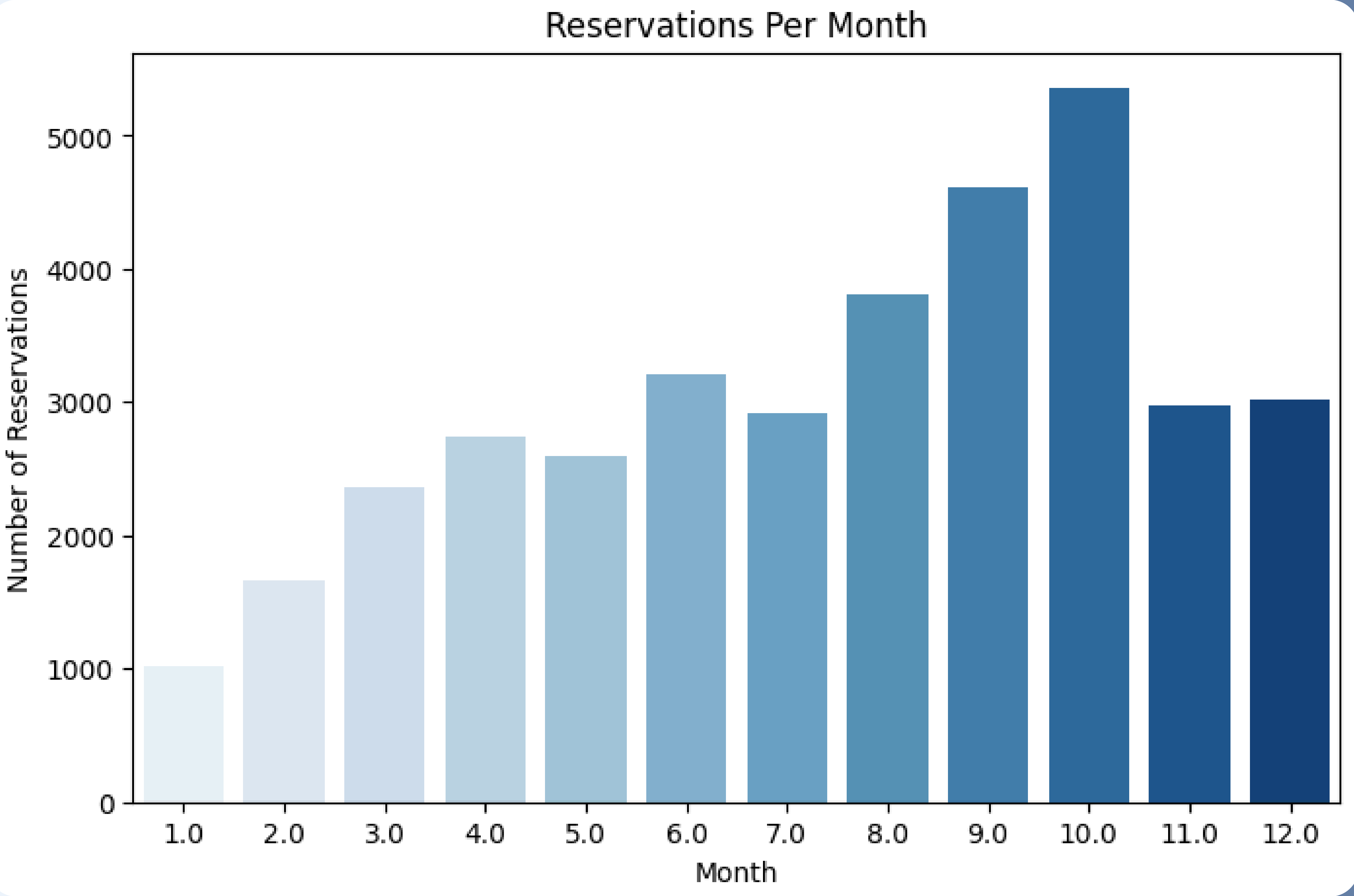
The average price per booking is around $103, with some variation.



SEASONAL BOOKING TRENDS

4

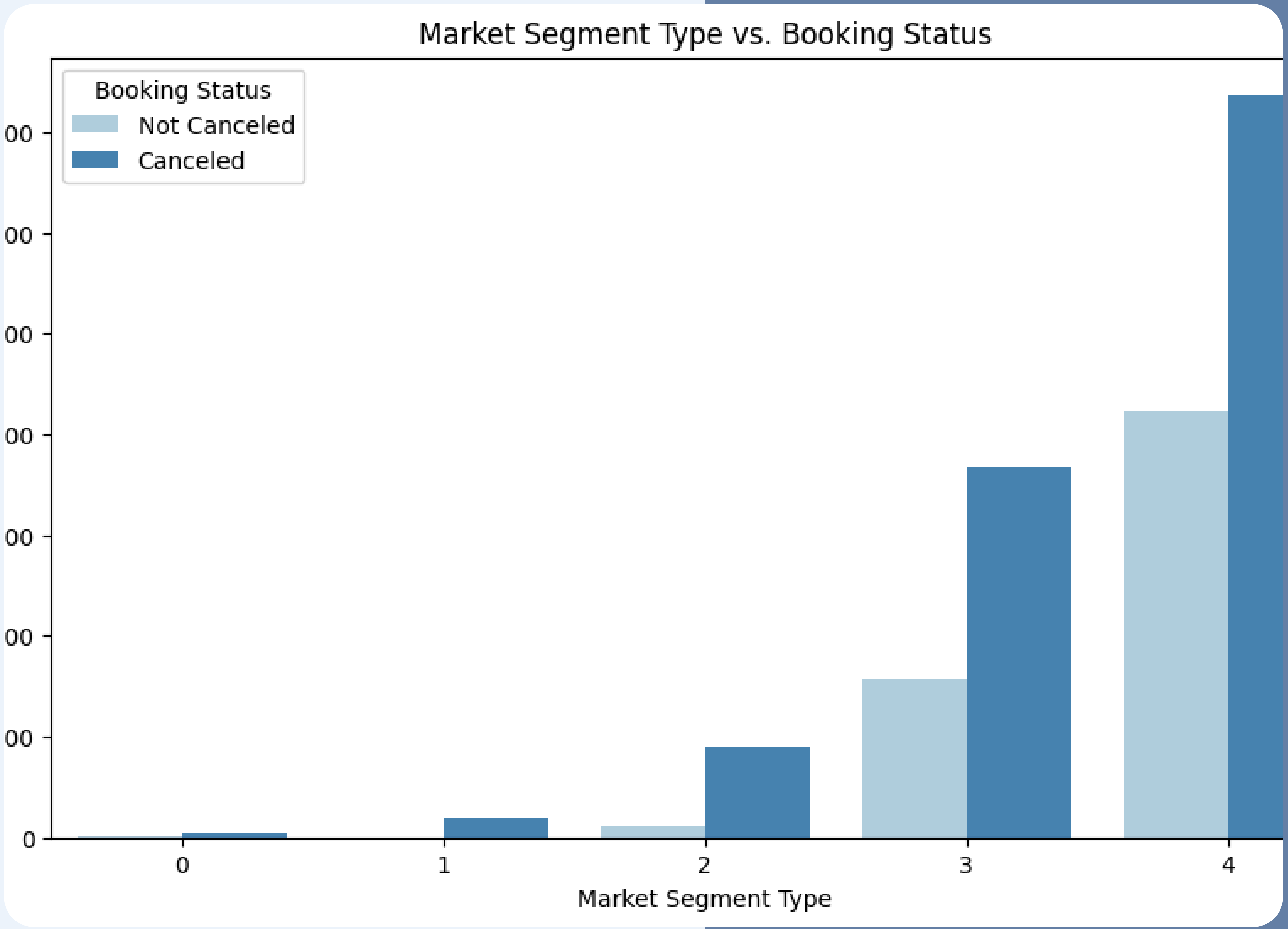
October is the peak booking month

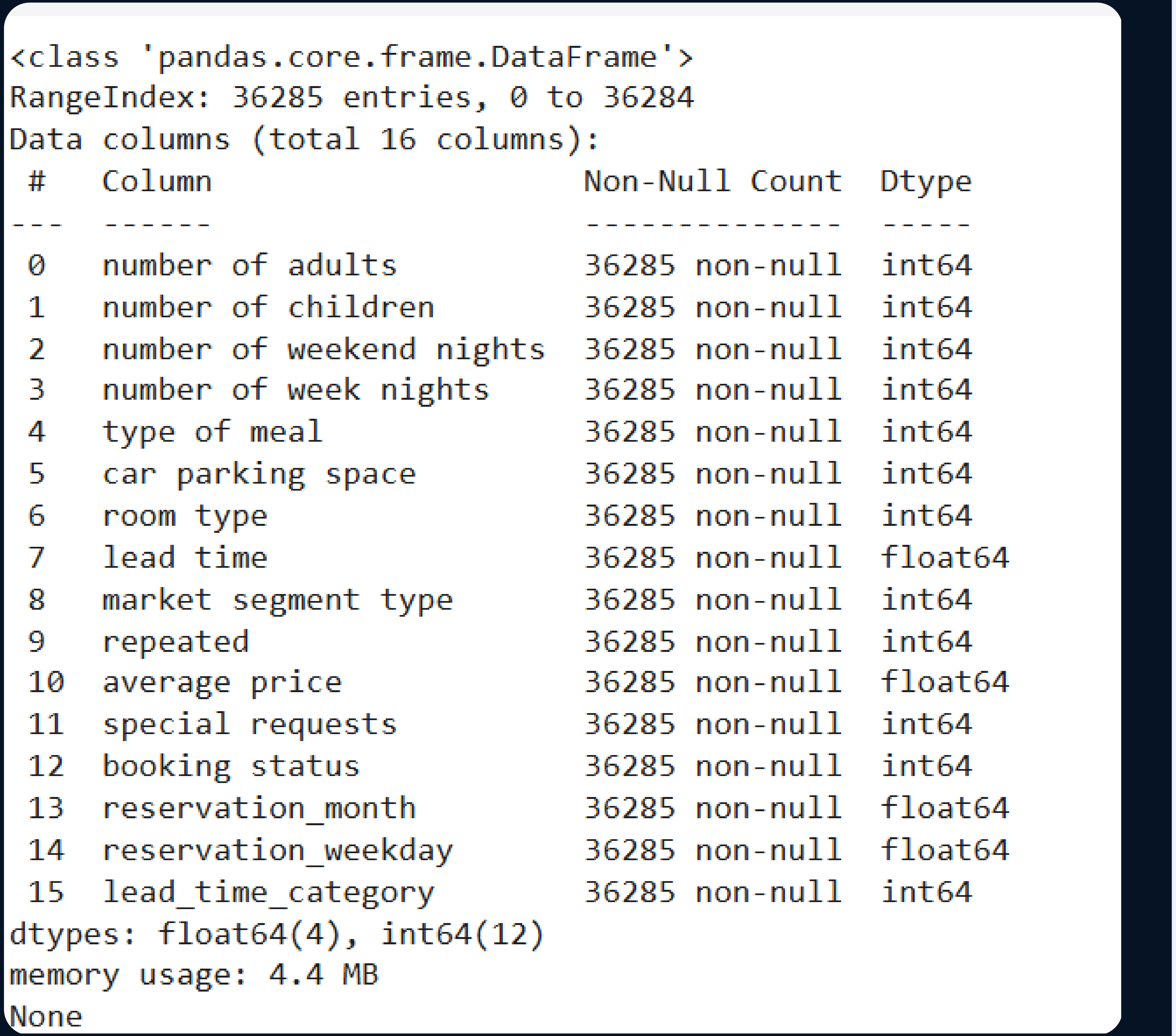


MARKET SEGMENT VS. BOOKING STATUS 5

- Online bookings are the highest among all market segments, but they also have a high cancellation rate.

- Online bookings have a high risk of cancellation, so strategies such as prepayment incentives or flexible pricing could be useful.





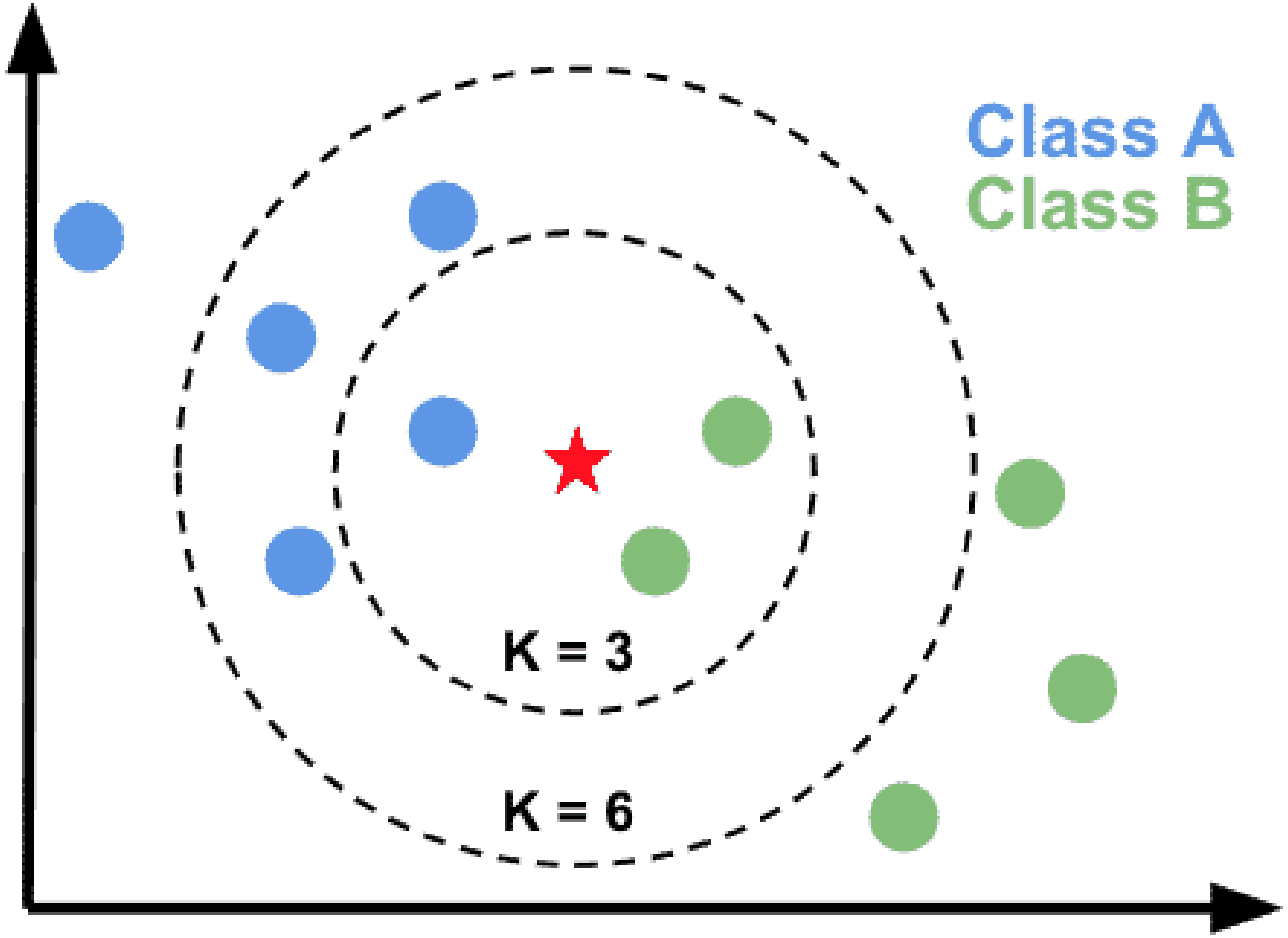
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 36285 entries, 0 to 36284
Data columns (total 16 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   number of adults         36285 non-null  int64  
 1   number of children       36285 non-null  int64  
 2   number of weekend nights 36285 non-null  int64  
 3   number of week nights    36285 non-null  int64  
 4   type of meal             36285 non-null  int64  
 5   car parking space        36285 non-null  int64  
 6   room type                36285 non-null  int64  
 7   lead time                36285 non-null  float64
 8   market segment type      36285 non-null  int64  
 9   repeated                 36285 non-null  int64  
 10  average price            36285 non-null  float64
 11  special requests         36285 non-null  int64  
 12  booking status           36285 non-null  int64  
 13  reservation_month        36285 non-null  float64
 14  reservation_weekday      36285 non-null  float64
 15  lead_time_category       36285 non-null  int64  
dtypes: float64(4), int64(12)
memory usage: 4.4 MB
None
```

NOW
Data is Ready

Train The Model using KNN

KNN (K-NEAREST NEIGHBORS)

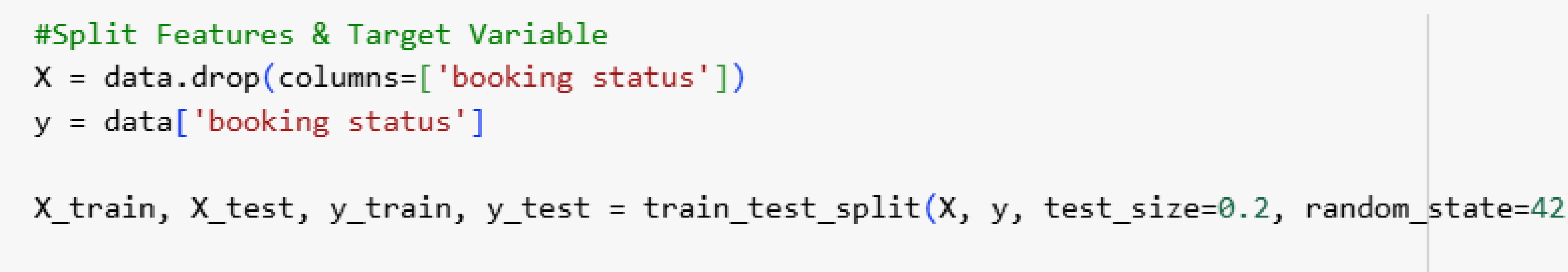
It classifies data points based on their 'nearest neighbors' in feature space.-Nearest Neighbors) is a classification algorithm.



Step 1

Feature Selection & Data Splitting

- FEATURES (X) ARE SELECTED BY REMOVING THE TARGET COLUMN.
- THE TARGET VARIABLE (Y) REPRESENTS THE CLASS WE WANT TO PREDICT.
- DATA IS SPLIT INTO TRAINING (80%) AND TESTING (20%) USING TRAIN_TEST_SPLIT().



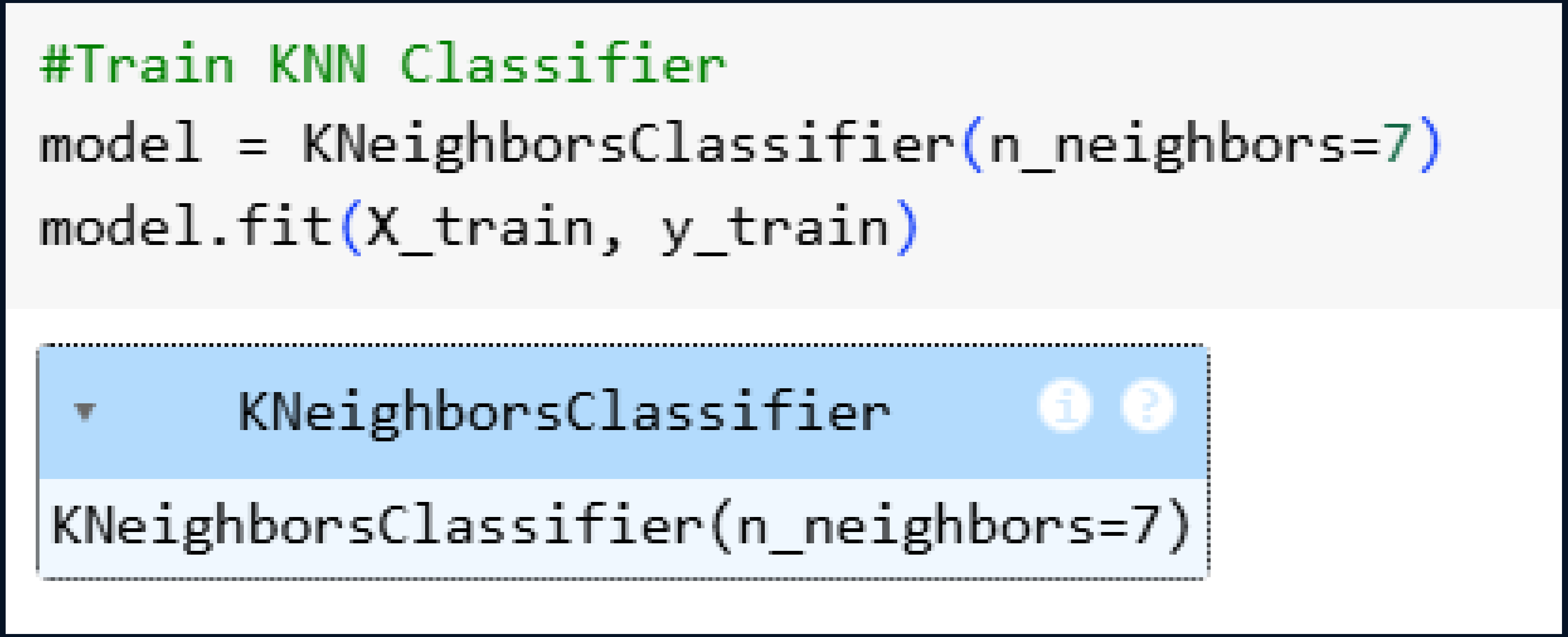
```
#Split Features & Target Variable
X = data.drop(columns=['booking status'])
y = data['booking status']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42
```

Step 2

Training the KNN Model

- **A KNN CLASSIFIER IS INITIALIZED WITH K=7.**
- **THE FIT() FUNCTION LEARNS PATTERNS FROM THE TRAINING SET.**



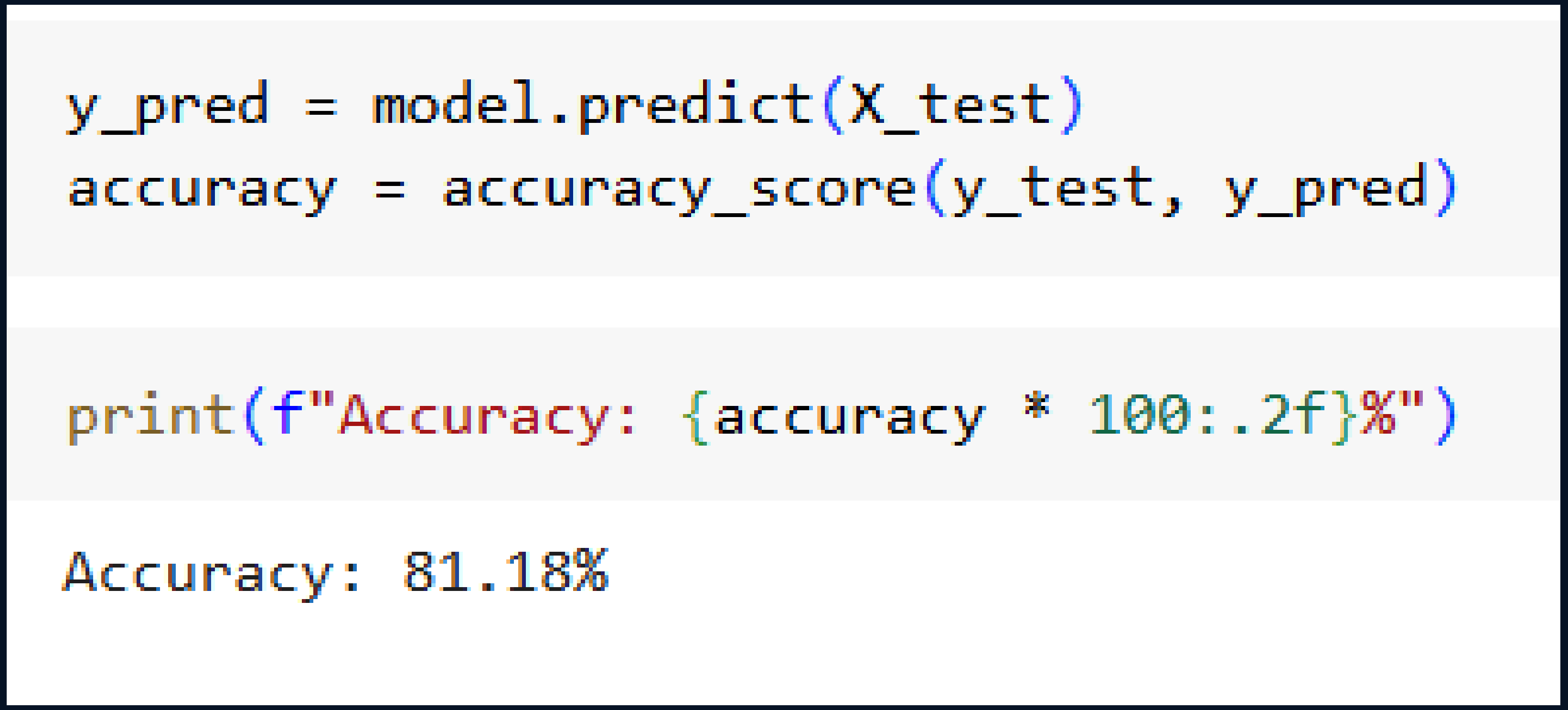
```python
#Train KNN Classifier
model = KNeighborsClassifier(n_neighbors=7)
model.fit(X_train, y_train)
```

```
KNeighborsClassifier
KNeighborsClassifier(n_neighbors=7)
```

Step 3

Evaluating Accuracy

- **ACCURACY IS MEASURED USING ACCURACY_SCORE().**
- **THE MODEL ACHIEVED 81.18% ACCURACY, INDICATING STRONG PERFORMANCE.**
- **TUNING THE "K" VALUE CAN FURTHER IMPROVE PERFORMANCE.**



```
y_pred = model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
```

```
print(f"Accuracy: {accuracy * 100:.2f}%")
```

```
Accuracy: 81.18%
```

Conclusion

Machine learning can significantly improve hotel management and customer retention strategies.

This provides a data-driven approach to predicting cancellations, helping businesses optimize operations.

THANK YOU

ASMAA IBRAHIM