Impact of EDA and Feature Engineering on Machine Learning Algorithms in Predicting Booking Cancellations
School of CSEE, University of Essex

Vasudev Subash Nair
snairvasudev@gmail.com

Abstract—The hotel industry is ever expanding and highly competitive industry.Data driven decisions can only be made by analysing all data available in short amounts of time.Exploratory data analysis is a critical step that brings insights into the various attributes of the data set which helps in uncovering patterns, relationships and anomalies in the data.Exploratory data analysis can help identify which factors affect customer's decisions. Transforming and selecting the most important features is an important step to optimise the models performance. Classification models such as logistic regression, decision trees, and random forests, are effective in predicting customer behavior. In this instance we shall be using five different classifiers to evaluate the model.

Index Terms—Hotel Booking, EDA, feature extraction, classification model.

1 Introduction

A Drop in occupancy rates and aggressive pricing are results of the pandemic's severe impact on the hotel sector. Hotels are dealing with a number of difficulties, such as workforce shortages and safety regulations to follow. This study investigates how feature engineering and exploratory data analysis (EDA) can boost the predictive power of machine learning systems for hotel booking cancellations. The study emphasises how crucial feature engineering and data cleansing are to building powerful machine learning models. The study explains the following steps in the research process for feature engineering and highlights the outcomes of the EDA process, including class imbalance and correlations between various features. I wish you the best of success.

2 Literature review/Background

When creating machine learning models, data preparation and feature engineering have become essential tasks. Davide Chicco, Luca Oneto, and Erica Tavazzi's article "Eleven Quick Tips for Data Cleaning and Feature Engineering" [1] offers a thorough overview of useful advice for carrying out these crucial processes. To prevent overfitting, the authors stress the significance of addressing missing values, treating outliers, handling categorical variables, and performing feature selection. The paper also discusses several feature engineering techniques, including imputation, scaling, normalisation, and encoding. "Determining the Intervening Effects of Exploratory Data Analysis and Feature Engineering in Telecoms Customer Churn Modelling" by Halibas et al. [2] is another work that examines the importance of these techniques in telecoms customer churn modelling. The study emphasises how crucial data preparation methods are for increasing the precision of machine learning models. The authors show that feature engineering and exploratory data analysis significantly alter the connection between the independent and dependent variables in churn models. Overall, these studies emphasise the significance of feature engineering and data cleaning in machine learning and offer insightful information about the methods that can be employed to prepare data for predictive modelling.

- *Vasudev Subash Nair was a post graduate student with the Department of Computer Science and Electrical Engineering, University of Essex, Colchester, UK, CO4 3SQ.*
  E-mail: snairvasudev@gmail.com

Manuscript received April 27, 2023

3 Methodology

Research Question: What is the impact of exploratory data analysis (EDA) and feature engineering on the accuracy of machine learning algorithms in predicting whether a booking will be cancelled based on available data?

3.1 Data Collection

The data used was obtained from the dept of CSEE of the University of Essex. The data contains 'Booking_ID', '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' and 'booking_status'. The dataset contained 36,275 instances

```
#analysing the data
data_df.describe().T
```

	count	mean	std	min	25%	50%	75%	max
no_of_adults	36275.0	1.844962	0.518715	0.0	2.0	2.00	2.0	4.0
no_of_children	36275.0	0.105279	0.402648	0.0	0.0	0.00	0.0	10.0
no_of_weekend_nights	36275.0	0.810724	0.870644	0.0	0.0	1.00	2.0	7.0
no_of_week_nights	36275.0	2.204300	1.410905	0.0	1.0	2.00	3.0	17.0
required_car_parking_space	36275.0	0.030986	0.173281	0.0	0.0	0.00	0.0	1.0
lead_time	36275.0	85.232557	85.930817	0.0	17.0	57.00	126.0	443.0
arrival_year	36275.0	2017.820427	0.383836	2017.0	2018.0	2018.00	2018.0	2018.0
arrival_month	36275.0	7.423653	3.069894	1.0	5.0	8.00	10.0	12.0
arrival_date	36275.0	15.596995	8.740447	1.0	8.0	16.00	23.0	31.0
repeated_guest	36275.0	0.025637	0.158053	0.0	0.0	0.00	0.0	1.0
no_of_previous_cancellations	36275.0	0.023349	0.368331	0.0	0.0	0.00	0.0	13.0
no_of_previous_bookings_not_canceled	36275.0	0.153411	1.754171	0.0	0.0	0.00	0.0	58.0
avg_price_per_room	36275.0	103.423539	35.089424	0.0	80.3	99.45	120.0	540.0
no_of_special_requests	36275.0	0.619655	0.786236	0.0	0.0	0.00	1.0	5.0

Fig. 1. Analysing data using describe().

and 19 attributes with the target label containing values of 'Canceled' and 'Not_Canceled'.

3.2 Exploratory Data Analysis

The data was loaded onto a pandas dataframe. Valuable insights were found by plotting different types of graphs and other methods for the different attributes present in the data. It was found that there weren't any 'NULL' values present in any of the columns. By using the 'pd.describe()' function of the pandas library, a general idea regarding the distribution of numerical data was understood. The data consists of instances for the years 2017 and 2018 with most of the bookings being in the year 2018. The number of repeated guests were few and no more than 1. The most a person has repeatedly cancelled their bookings was 13 and most not cancelled was 58, but since 75% of cancellation can be seen to be 0, it suggests that the number of cancellations were lesser than 25% of the total bookings. The average price per room being a minimum of 0 suggests that there were rooms that were booked for free, but since 25% of the average price per room is 80.3 we can conclude that the number of free bookings are very few. Also since the distribution of the avg_price_per_room is almost more than 4.5 times its 75% data, we can assume that it has more outlier data for that field. The maximum number of special requests made was 5 and the minimum being 0.

By checking the data types of the data under each attribute, it was noticed that four of the attribute columns contained the datatype 'Object' which indicated that these were categorical values.

Since the primary aim is to classify the data according to 'booking_status', the number of 'Canceled' and 'Not_Canceled' data were plotted against each other. It was observed that there was a class imbalance between the two with only 36.7% as 'Canceled'.

The column 'lead_time' contains data regarding the number of days since booking was made until the date of booking. The frequency of the data in 'lead_time' was plotted according to 'Canceled' and 'Not_Canceled' booking status. The plot showed that as the lead time increases, the chances of teh booking being cancelled also increases. The mean of the values of this column was taken according to the 'booking_status' and then plotted. Putting these two

```
#checking the datatypes of the data present in each column
data_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 36275 entries, 0 to 36274
Data columns (total 19 columns):
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   Booking_ID                            36275 non-null  object
 1   no_of_adults                          36275 non-null  int64
 2   no_of_children                        36275 non-null  int64
 3   no_of_weekend_nights                  36275 non-null  int64
 4   no_of_week_nights                     36275 non-null  int64
 5   type_of_meal_plan                     36275 non-null  object
 6   required_car_parking_space            36275 non-null  int64
 7   room_type_reserved                    36275 non-null  object
 8   lead_time                             36275 non-null  int64
 9   arrival_year                          36275 non-null  int64
 10  arrival_month                         36275 non-null  int64
 11  arrival_date                          36275 non-null  int64
 12  market_segment_type                   36275 non-null  object
 13  repeated_guest                        36275 non-null  int64
 14  no_of_previous_cancellations          36275 non-null  int64
 15  no_of_previous_bookings_not_canceled  36275 non-null  int64
 16  avg_price_per_room                    36275 non-null  float64
 17  no_of_special_requests                36275 non-null  int64
 18  booking_status                        36275 non-null  object
dtypes: float64(1), int64(13), object(5)
memory usage: 5.3+ MB
```

Fig. 2. Checking datatype of columns

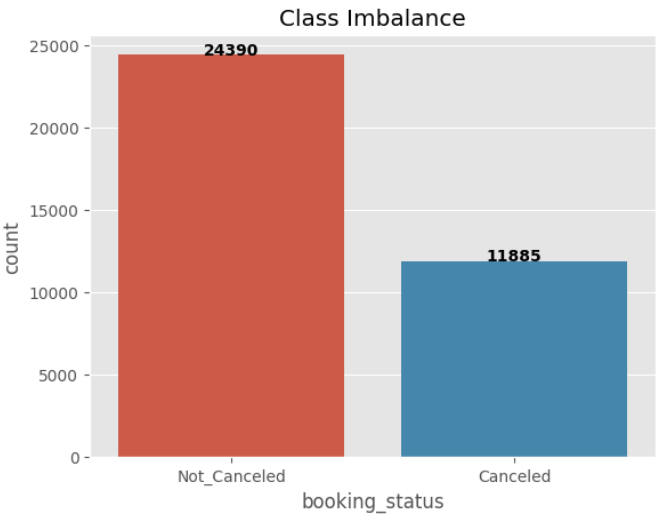

Fig. 3. Booking status count

plots together, it was found that the most bookings were cancelled when the lead time went beyond 140 days.

A bar plot distribution of the bookings were plotted according to month and year. The number of bookings were observed to be lower during the last 5 months of the year 2017 and drastically improved starting from the month of February in 2018. The highest number of bookings were during the month of October in 2018 before it fell again in November and December during the same year.

Plotting the type of room reserved for each booking by grouping it with the mean of the average room price, it was observed that room type 6 was the most expensive compared to all the other rooms whereas room type 3 was the least expensive. Combining this with the plot regarding frequency of the type of rooms booked according to booking type, we can see that most people booked room type 1 which

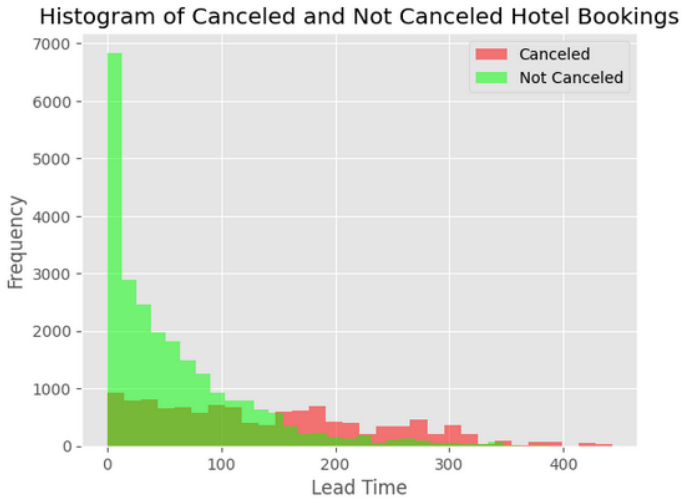

Fig. 4. Lead time vs booking type

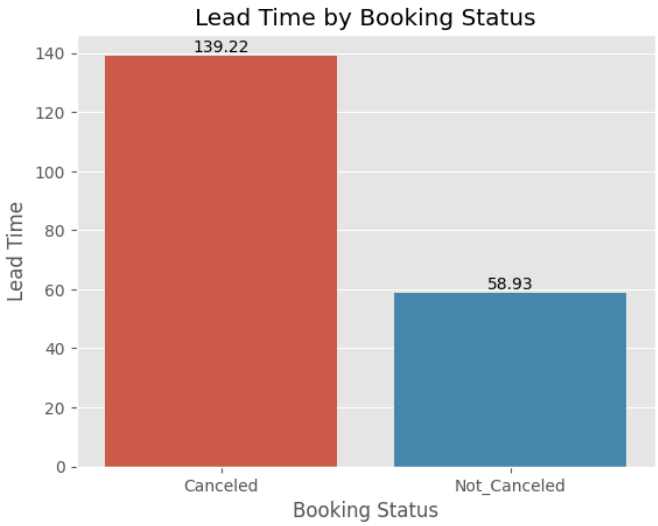

Fig. 5. avg lead time by booking status

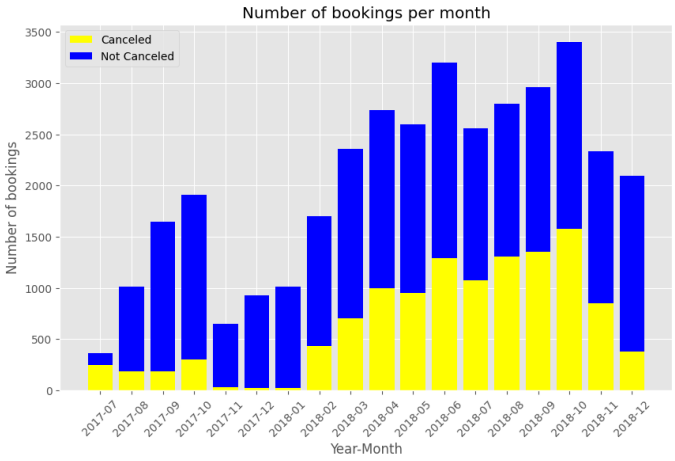

Fig. 6. bookings by month

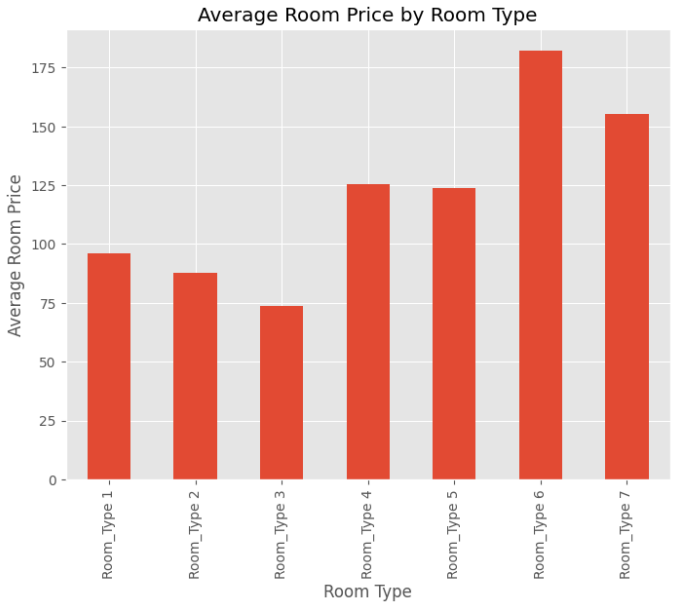

Fig. 7. Room type by average room price

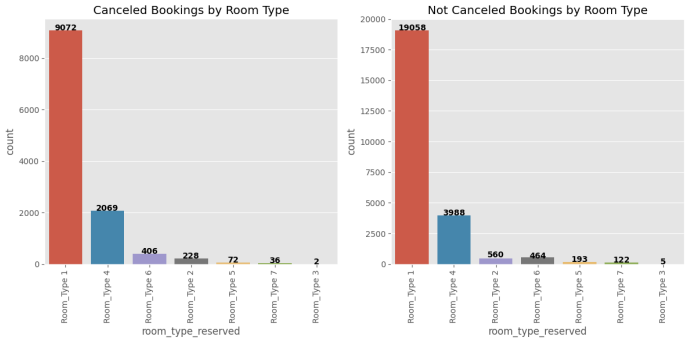

Fig. 8. room type reserved count

had the third lowest price and only a total of 7 bookings were made for room type 3. Almost half of the bookings made for room type 6 were cancelled.

After plotting the type of meal plan for all the bookings against both cancelled and not cancelled booking status, it is understood that 52% of the total bookings that were not cancelled had chosen type of meal plan 1. Almost 43% of the total number of people who chose type of meal 2 had cancelled their bookings.

Plots on the different types of market segments from which the bookings came from were also recorded in the data. Out of this most of the cancellations were from online bookings which is only 23% of the total data. Aviation had the least number of bookings out of which 29% of the total bookings from the aviation market were cancelled.

The pairplot generated showed various outlier data for columns like average room price, number of children, market segment type etc.

The correlation heatmap suggests that the attributes that influence each other the most are repeated guests,number of previous cancellations and number of previous bookings not cancelled. There is also a good correlation between average price per room with number of adults and number

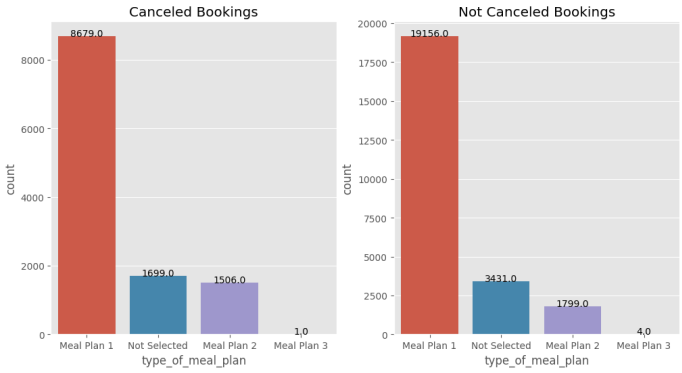

Fig. 9. type of meal plan count

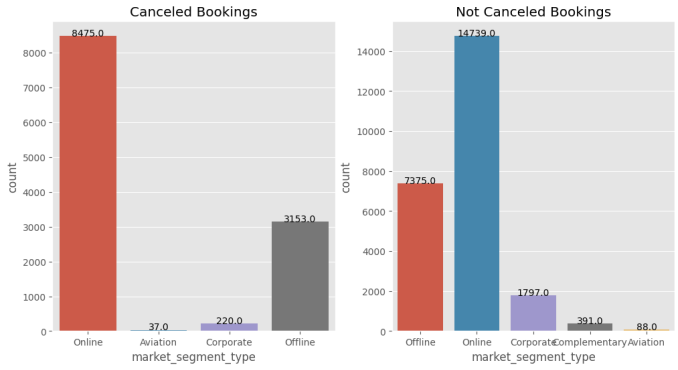

Fig. 10. market segment type count

of children. There is a negative correlation between arrival month and arrival year. Some of the attributes that have slightly good correlation with each other are as follows,

1) Lead time with number of week nights
2) Arrival year and arrival month with lead time
3) Repeated guest with required car parking space

3.3 Data Cleaning and Preprocessing

The attribute 'Booking_ID' was dropped as it had too many unique values and would not improve the accuracy of the models used for classification.

There were rows discovered which had no_of_adults = 0, which indicated that there was false data. To solve this, the no_of_children in these rows were analysed wherever the no_of_adults were equal to 0 and the values were changed according to the count of the no_of children with the values assigned as no_of_adults = 2.

The outlier data present in the column avg_price_per_room was identified using the Interquartile Range method. The equation for the method is as follows,

$$IQR = Q_3 – Q_1 \text{ [3]}$$

Where, Q3 - The data point present quarter way through the list Q1 - The data point present three quarters of the way through the list. The values for these were obtained using the quantile() function of numpy. The rows in which the outliers were found to be present were dropped.

The data was divided into 'train' and 'target', where 'target' contained the 'booking_type' attribute.

```
model:  K Nearest Neighbor
Accuracy:  0.8195488721804511
Classification report:                precision    recall  f1-score   support

           0       0.84      0.90      0.87      2286
           1       0.77      0.66      0.71      1172

    accuracy                           0.82      3458
   macro avg       0.81      0.78      0.79      3458
weighted avg       0.82      0.82      0.82      3458

ROC_curve:  0.7816050510751003
model:  Random Forest
Accuracy:  0.9100636205899364
Classification report:                precision    recall  f1-score   support

           0       0.91      0.95      0.93      2286
           1       0.90      0.82      0.86      1172

    accuracy                           0.91      3458
   macro avg       0.91      0.89      0.90      3458
weighted avg       0.91      0.91      0.91      3458

ROC_curve:  0.8889422631897976
```

Fig. 11. Results of KNearest Neighbor and Random Forest

Categorical encoding was applied to all attributes that contained data of the data type 'object. The 'target' data was converted into binary values where 'Canceled' = 1 and 'Not_Canceled' = 0. The other categorical data in the 'train' dataframe was encoded using one hot encoding where each categorical data is converted into an attribute containing binary values indicating whether it is present for a particular row or not.

The data was split into 'X_train', 'y_train', 'X_test' and 'y_test' using 'train_test_split()' function of the 'sklearn.model_selection'.

3.4 Training Model

The training models chosen for this particular problem was Random Forest, K Nearest Neighbor, Logistic Regression, Decision Tree and XGBoost classifier. The model was fitted with the 'X_train' and 'Y_train' which are the training data and then validated using X_test and Y_test which is the validation data.

4 Results

The highest accuracy was found to be for Random Forest with an accuracy of 91%, followed by Decision Tree and XGBoost. The lowest accuracy was for Logistic Regression. Logistic regression showed a higher difference in f1_score and recall between teh two classes, which was 21% and 33%. The use of EDA and feature engineering has significantly increased the accuracy of the model. This is a classic example of supervised learning where in training and test sets are used to create the classification model. Confusion matrix was plotted for each classifier model.

5 Conclusion

The goal of this study was to increase the accuracy of the classification model using EDA and feature engineering. The models were evaluated using 'Accuracy', 'fi_score' and 'recall'.

```
model:  Logistic Regression
Accuracy:  0.7776171197223829
Classification report:                precision    recall  f1-score   support

           0       0.80      0.89      0.84      2286
           1       0.72      0.56      0.63      1172

    accuracy                           0.78      3458
   macro avg       0.76      0.72      0.74      3458
weighted avg       0.77      0.78      0.77      3458

ROC_curve:  0.723695054329813
model:  Decision Tree
Accuracy:  0.871602082128398
Classification report:                precision    recall  f1-score   support

           0       0.91      0.90      0.90      2286
           1       0.81      0.82      0.81      1172

    accuracy                           0.87      3458
   macro avg       0.86      0.86      0.86      3458
weighted avg       0.87      0.87      0.87      3458

ROC_curve:  0.8579810629473364
```

Fig. 12. Results of Logistic Regression and Decision Tree

```
model:  XGBoost
Accuracy:  0.85106998264893
Classification report:                precision    recall  f1-score   support

           0       0.86      0.93      0.89      2286
           1       0.83      0.70      0.76      1172

    accuracy                           0.85      3458
   macro avg       0.85      0.81      0.83      3458
weighted avg       0.85      0.85      0.85      3458

ROC_curve:  0.8141775580100269
```

Fig. 13. Results of XGBoost

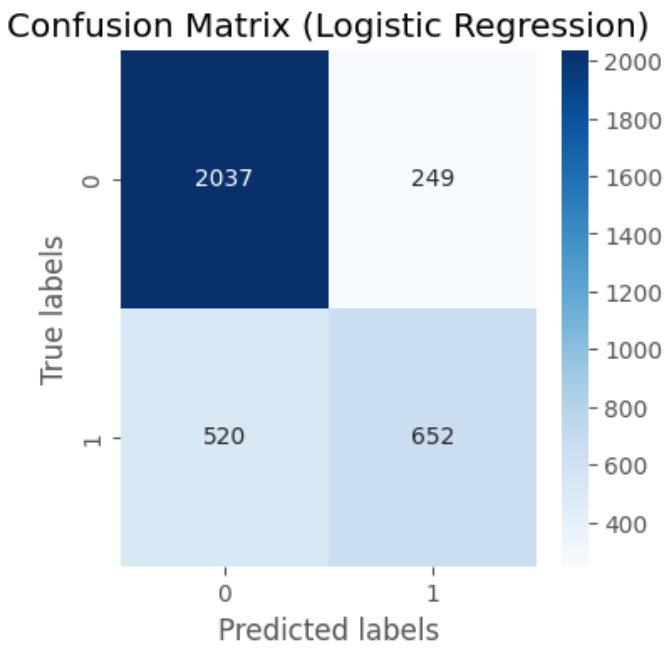

Fig. 14. Confusion matrix for Logistic Regression

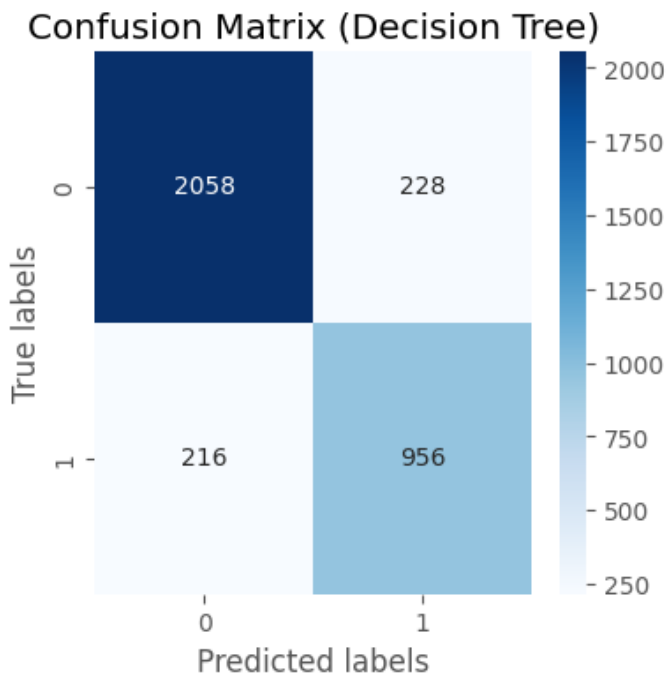

Fig. 15. Confusion matrix for Decision Tree

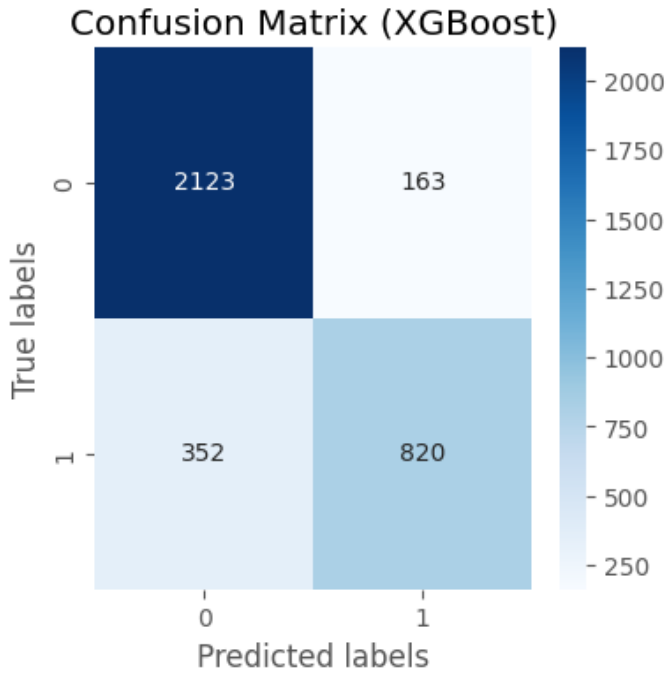

Fig. 16. Confusion matrix for XGBoost

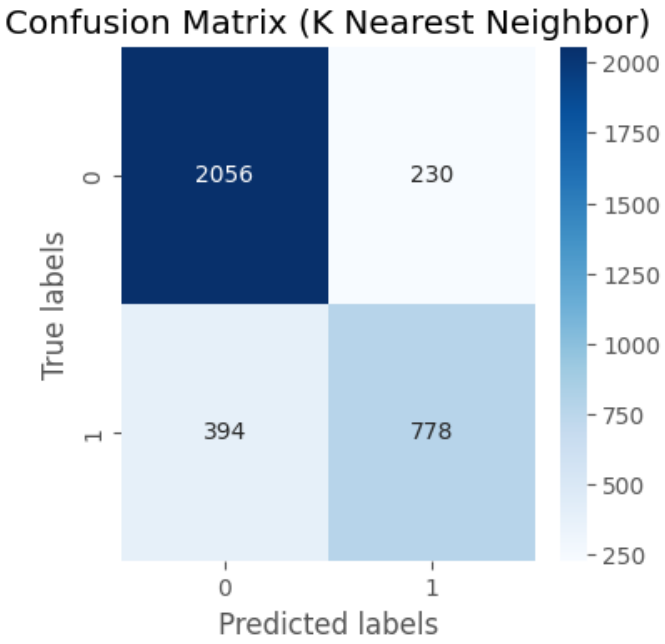

Fig. 17. Confusion matrix for KNearest Neighbor

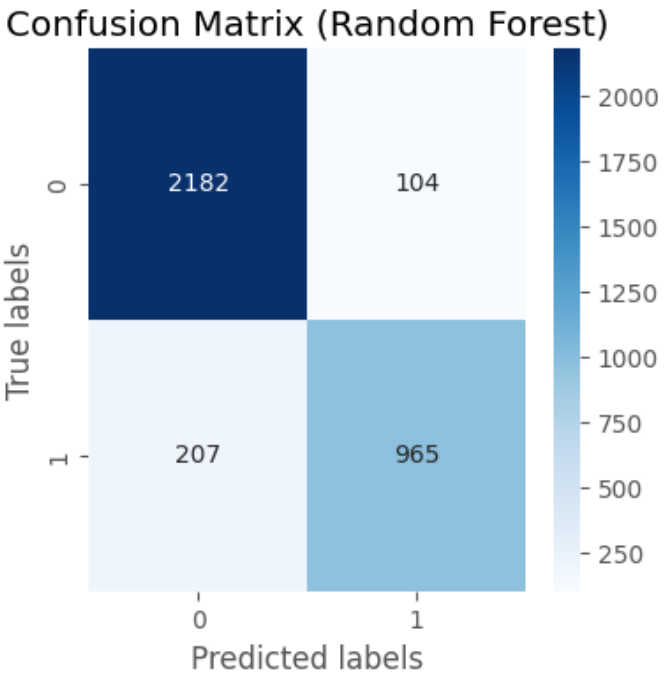

Fig. 18. Confusion matrix for Random Forest

REFERENCES

[1] D. Chicco, L. Oneto, and E. Tavazzi, "Eleven quick tips for data cleaning and feature engineering," *Journal of Machine Learning Research*, vol. 19, no. 1, pp. 1–6, 2018.

[2] A. Halibas, A. Cherian, I. Govinda Pillai, J. H. Reazol, E. G. Delvo, and L. Reazol, "Determining the intervening effects of exploratory data analysis and feature engineering in telecoms customer churn modelling," pp. 1–7, 01 2019.

[3] C. Taylor, "What is the interquartile range rule?." https://www.thoughtco.com/what-is-the-interquartile-range-rule-3126244, accessed 2023-04-27.

[2] [1]