SOCIAL MEDIA TOURISM

BUSINESS REPORT - CAPSTONE PROJECT

SONA

PGPDSBA.O.MAY24.A

CONTENTS

LIST OF TABLES

LIST OF FIGURES

Problem Statement

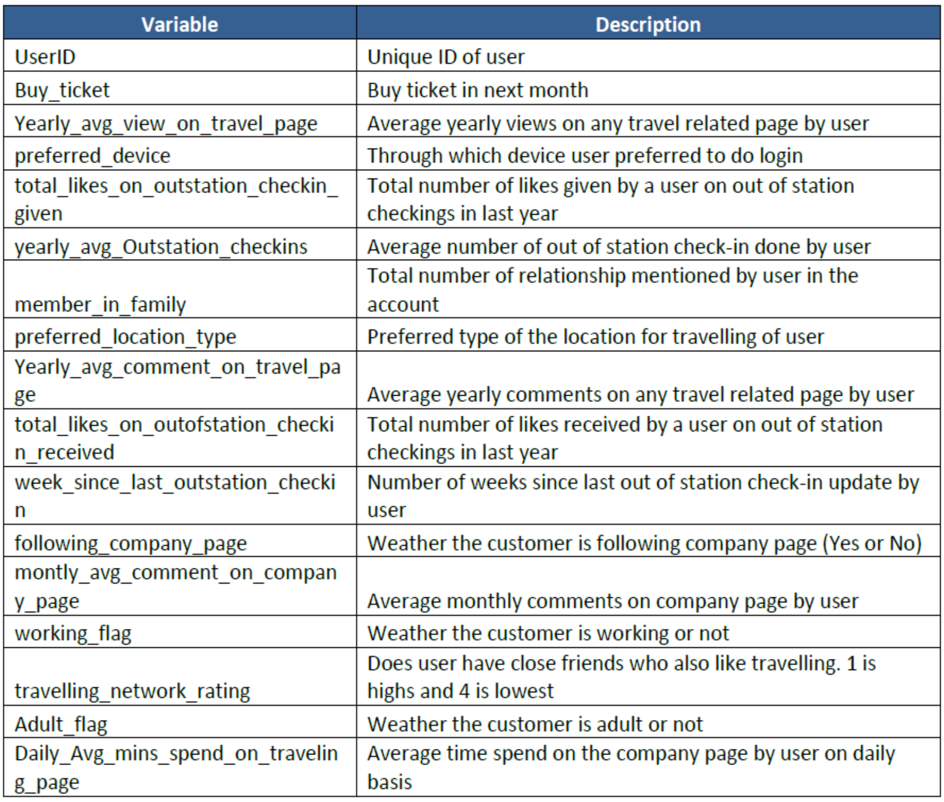
Business Objective

An aviation company that provides domestic as well as international trips to the customers now wants to apply a targeted approach instead of reaching out to each of the customers. This time they want to do it digitally instead of tele calling. Hence they have collaborated with a social networking platform, so they can learn the digital and social behaviour of the customers and provide the digital advertisement on the user page of the targeted customers who have a high propensity to take up the product.

Propensity of buying tickets is different for different login devices. Hence, you have to create 2 models separately for Laptop and Mobile. [Anything which is not a laptop can be considered as mobile phone usage.]

The advertisements on the digital platform are a bit expensive; hence, you need to be very accurate while creating the models.

Data Description



Variable	Description
UserID	Unique ID of user
Buy_ticket	Buy ticket in next month
Yearly_avg_view_on_travel_page	Average yearly views on any travel related page by user
preferred_device	Through which device user preferred to do login
total_likes_on_outstation_checkin_given	Total number of likes given by a user on out of station checkings in last year
yearly_avg_Outstation_checkins	Average number of out of station check-in done by user
member_in_family	Total number of relationship mentioned by user in the account
preferred_location_type	Preferred type of the location for travelling of user
Yearly_avg_comment_on_travel_page	Average yearly comments on any travel related page by user
total_likes_on_outofstation_checkin_received	Total number of likes received by a user on out of station checkings in last year
week_since_last_outstation_checkin	Number of weeks since last out of station check-in update by user
following_company_page	Weather the customer is following company page (Yes or No)
montly_avg_comment_on_company_page	Average monthly comments on company page by user
working_flag	Weather the customer is working or not
travelling_network_rating	Does user have close friends who also like travelling. 1 is highs and 4 is lowest
Adult_flag	Weather the customer is adult or not
Daily_Avg_mins_spend_on_traveling_page	Average time spend on the company page by user on daily basis

1. Introduction - What did you wish to achieve while doing the project ?

a. Defining the problem statement

Digital marketing involves a strategic effort to support or achieve a business objective through the use of one or more social media platforms. Unlike regular social media activities, marketing campaigns are more focused, targeted, and measurable. In this project, we are working with an aviation company aiming to leverage digital marketing to design a targeted campaign. They have partnered with a social networking platform to analyze customer behavior. Notably, purchasing behavior varies based on the device used - Laptop or Mobile - so separate predictive models are developed for each device type. Given the high cost of digital advertising, it is crucial for these models to be highly accurate. The objective of this project is to predict whether a customer will purchase a tourism package as a result of the social media campaign. Social media allows businesses to reach a wider audience, build brand awareness, drive website traffic, and ultimately boost revenue. In today's tech-driven world, effective use of social media is essential for business success.

b. Need of the study/project

With the rise of social media and growing digital footprints, digital marketing offers more precise targeting compared to traditional methods like tele-calling or physical ads.
This project aims to help the aviation company move towards a data-driven approach to reach potential customers more effectively.

By partnering with a social networking platform, the company can:

- Analyze customers' digital and social behaviour
- Identify users with a higher likelihood of purchasing flight tickets
- Serve targeted ads only to those most likely to convert

Given the high cost of digital ads, accuracy is critical. Key factors influencing customer interest include:

- **Login device** (Laptop vs. Mobile)
- **Family size**
- **Preferred travel locations**
- **History of website visits and interactions**

Understanding these behaviours will help the company reduce manual outreach, improve targeting, enhance customer satisfaction, and drive revenue through personalized offers and travel packages.

c. Understanding business/social opportunity

• Business Opportunity
This project allows the company to focus on the **right audience**, increasing the likelihood of converting leads into customers. This targeted approach helps boost revenue, reduce tele-calling costs, and improve ROI on marketing spend. With better customer targeting, the company can enhance retention and control marketing expenses more efficiently.

• Social Opportunity
As the business grows, it creates **more job opportunities**, contributing to social development. Customers also benefit from timely service, relevant offers, and a better overall experience—while avoiding unnecessary calls, saving their time and improving satisfaction.

2. EDA - Univariate / Bivariate / Multivariate analysis to understand relationship between variables. - Both visual and non-visual understanding of the data.

a. Understanding how data was collected in terms of time, frequency, and methodology

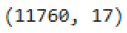
The data, collected by a third-party social networking platform, captures the travel-related behaviour of 11,760 unique customers. It includes:

- Likes, comments, reviews on travel content
- Outstation check-ins and related interactions
- Personal details (family size, work status, adult status, time spent on travel pages)
- A target label indicating if a customer booked a ticket

Data was recorded at different intervals:

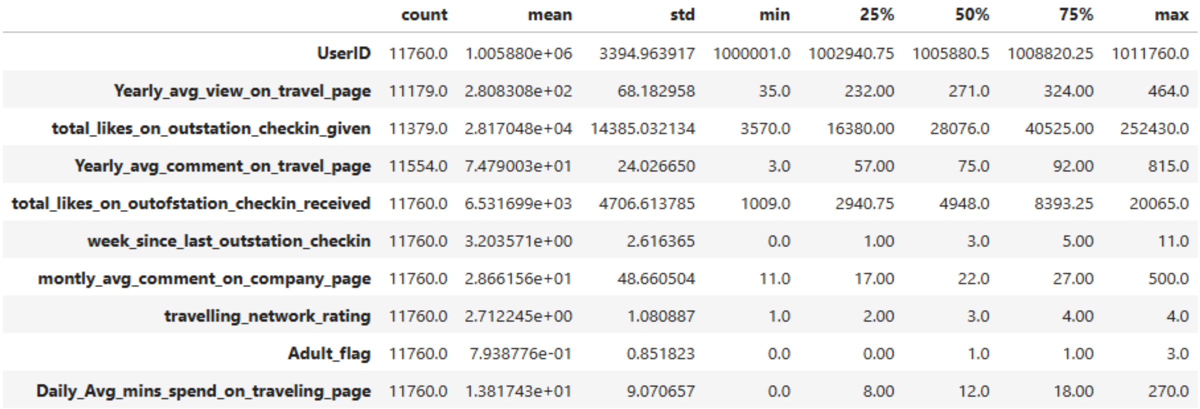
- Yearly averages for views, check-ins, and comments
- Monthly average for company page comments
- Some metrics are collected daily or weekly, then averaged for consistency

b. Visual inspection of data (rows, columns, descriptive details)



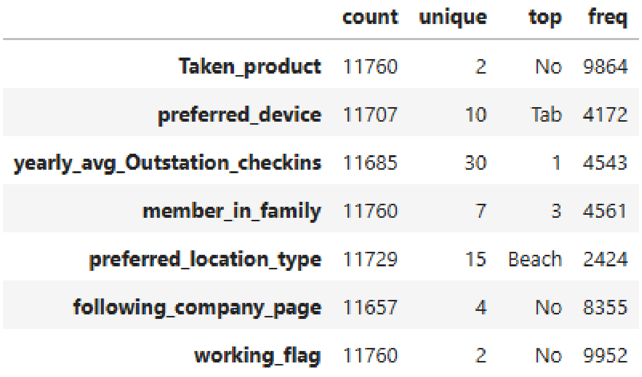
```
(11760, 17)
```

Table 1: Shape of dataset



	count	mean	std	min	25%	50%	75%	max
UserID	11760.0	1.005880e+06	3394.963917	1000001.0	1002940.75	1005880.5	1008820.25	1011760.0
Yearly_avg_view_on_travel_page	11179.0	2.808308e+02	68.182958	35.0	232.00	271.0	324.00	464.0
total_likes_on_outstation_checkin_given	11379.0	2.817048e+04	14385.032134	3570.0	16380.00	28076.0	40525.00	252430.0
Yearly_avg_comment_on_travel_page	11554.0	7.479003e+01	24.026650	3.0	57.00	75.0	92.00	815.0
total_likes_on_outofstation_checkin_received	11760.0	6.531699e+03	4706.613785	1009.0	2940.75	4948.0	8393.25	20065.0
week_since_last_outstation_checkin	11760.0	3.203571e+00	2.616365	0.0	1.00	3.0	5.00	11.0
montly_avg_comment_on_company_page	11760.0	2.866156e+01	48.660504	11.0	17.00	22.0	27.00	500.0
travelling_network_rating	11760.0	2.712245e+00	1.080887	1.0	2.00	3.0	4.00	4.0
Adult_flag	11760.0	7.938776e-01	0.851823	0.0	0.00	1.0	1.00	3.0
Daily_Avg_mins_spend_on_traveling_page	11760.0	1.381743e+01	9.070657	0.0	8.00	12.0	18.00	270.0

Table 2: Continuous numerical variable



	count	unique	top	freq
Taken_product	11760	2	No	9864
preferred_device	11707	10	Tab	4172
yearly_avg_Outstation_checkins	11685	30	1	4543
member_in_family	11760	7	3	4561
preferred_location_type	11729	15	Beach	2424
following_company_page	11657	4	No	8355
working_flag	11760	2	No	9952

Table 3: Categorical variable

- Majority of customers haven't purchased the product and rarely engage with travel content—averaging 2.62 weeks since last check-in and a low travel rating of 2.71.
- **Most users prefer "Tab"** over laptops and don't follow the company page.
- Average family size is 3 with 1 annual trip.
- Users **spend around 13.82 minutes** daily on the travel page.

Data Wrangling

- Reclassified Preferred_device into two categories: **Laptop** and **Mobile** (for all others).
- Replaced '*' in Yearly_avg_Outstation_checkins with mode and converted to float.

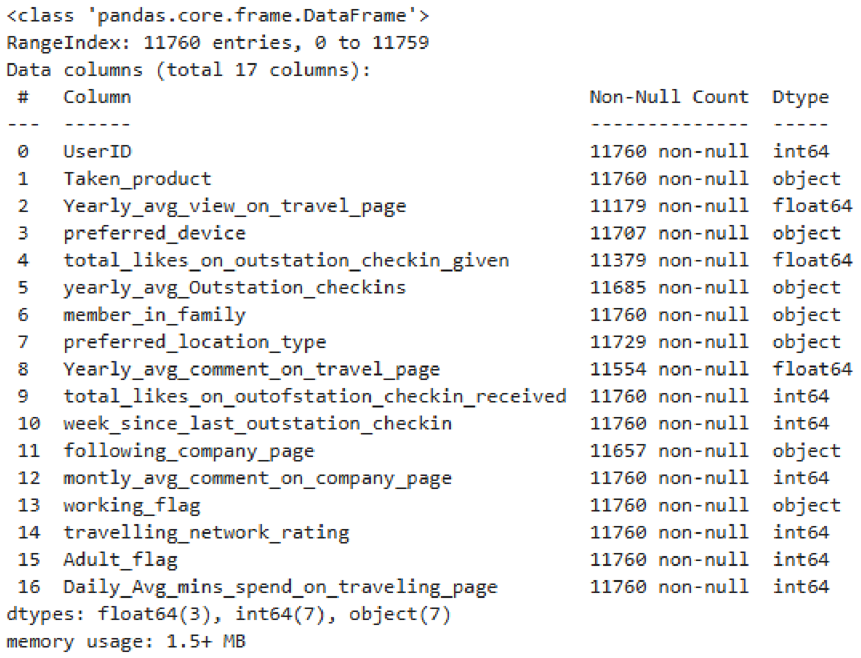
- Converted 'Three' to 3 in Member_in_family. Merged 'Tour Travel' and 'Tour and Travel' into a single category.
- Standardized Adult_flag to binary: 0 for non-adult, any other value as adult.

c. Understanding of attributes (variable info, renaming if required)

```
0
```

Table 4: Duplicate value

There are **no duplicate values** in the dataset.



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11760 entries, 0 to 11759
Data columns (total 17 columns):
 #   Column                                 Non-Null Count  Dtype  
---  ------                                 --------------  -----  
 0   UserID                                 11760 non-null  int64  
 1   Taken_product                          11760 non-null  object 
 2   Yearly_avg_view_on_travel_page         11179 non-null  float64
 3   preferred_device                       11707 non-null  object 
 4   total_likes_on_outstation_checkin_given  11379 non-null  float64
 5   yearly_avg_Outstation_checkins         11685 non-null  object 
 6   member_in_family                       11760 non-null  object 
 7   preferred_location_type                11729 non-null  object 
 8   Yearly_avg_comment_on_travel_page      11554 non-null  float64
 9   total_likes_on_outofstation_checkin_received  11760 non-null  int64  
 10  week_since_last_outstation_checkin     11760 non-null  int64  
 11  following_company_page                 11657 non-null  object 
 12  montly_avg_comment_on_company_page     11760 non-null  int64  
 13  working_flag                           11760 non-null  object 
 14  travelling_network_rating              11760 non-null  int64  
 15  Adult_flag                             11760 non-null  int64  
 16  Daily_Avg_mins_spend_on_traveling_page  11760 non-null  int64  
dtypes: float64(3), int64(7), object(7)
memory usage: 1.5+ MB
```

Table 5: Data type

- **Null values** are present in several columns and need to be handled.
- The dataset has **17 variables**: 3 float, 7 int, and 7 object types, with data for **11,760 individuals**.
- As per the problem statement, any device other than a laptop is categorized as mobile usage. Therefore, all other device types have been recoded to 'Mobile' under the ‘preferred_device’ variable.

Univariate Analysis

Univariate analysis is the most basic form of data analysis, focusing on a single variable. It doesn't explore relationships or causes like regression does—its main goal is to **summarize and identify patterns** within that one variable.

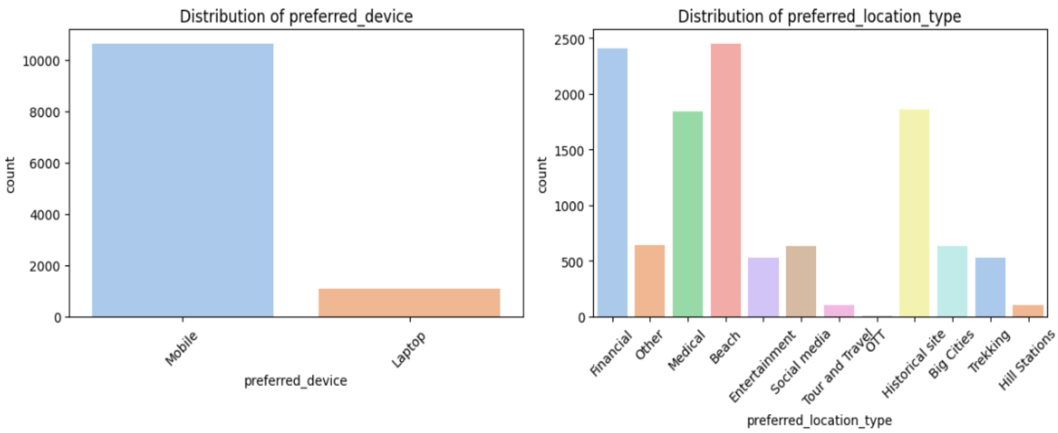
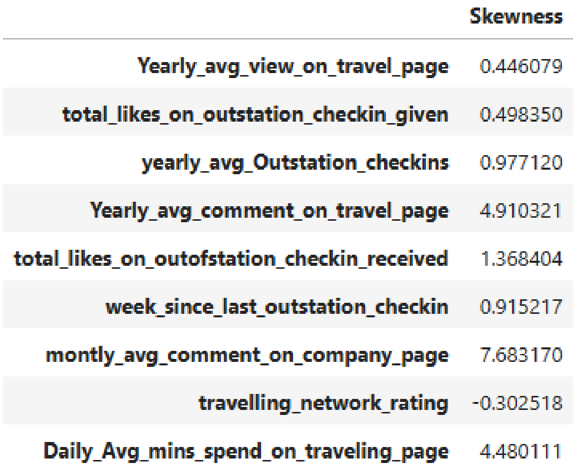


Fig 1. Distribution of Device & Location Type

- Mobile devices are overwhelmingly preferred, with a significantly higher count than laptops, highlighting the dominance of mobile usage among users.
- Among travel preferences, 'Beach' and 'Financial' locations are the most favored, indicating a mix of leisure and practical travel interests.
- Categories like 'Hill Stations', 'Tour and Travel', and 'Trekking' are least preferred, suggesting they may not be appealing or accessible to most users.
- Preferences are diverse, but a noticeable segment still opts for 'Medical' and 'Historical sites', possibly driven by necessity or cultural interest.



	Skewness
Yearly_avg_view_on_travel_page	0.446079
total_likes_on_outstation_checkin_given	0.498350
yearly_avg_Outstation_checkins	0.977120
Yearly_avg_comment_on_travel_page	4.910321
total_likes_on_outofstation_checkin_received	1.368404
week_since_last_outstation_checkin	0.915217
montly_avg_comment_on_company_page	7.683170
travelling_network_rating	-0.302518
Daily_Avg_mins_spend_on_traveling_page	4.480111

Fig 2. Skewness Analysis of Continuous Variables

Continuous Variables - Skewness

Skewness indicates how much a variable's distribution deviates from symmetry. A value of 0 means the data is symmetrical, values between ±0.5 to ±1 indicate mild skewness, and values greater than ±1 represent high skewness.

A positive skewness means the data is right-skewed, while a negative skewness indicates left-skewed data.

In this dataset:

- Right-skewed variables include: Yearly_avg_view_on_travel_page, Total_likes_on_outstation_checkin_given, Yearly_avg_Outstation_checkins, Yearly_avg_comment_on_travel_page, Total_likes_on_outofstation_checkin_received, Week_since_last_outstation_checkin, Montly_avg_comment_on_company_page, and Daily_Avg_mins_spend_on_traveling_page.
- Left-skewed variable: travelling_network_rating.

Categorical Variables

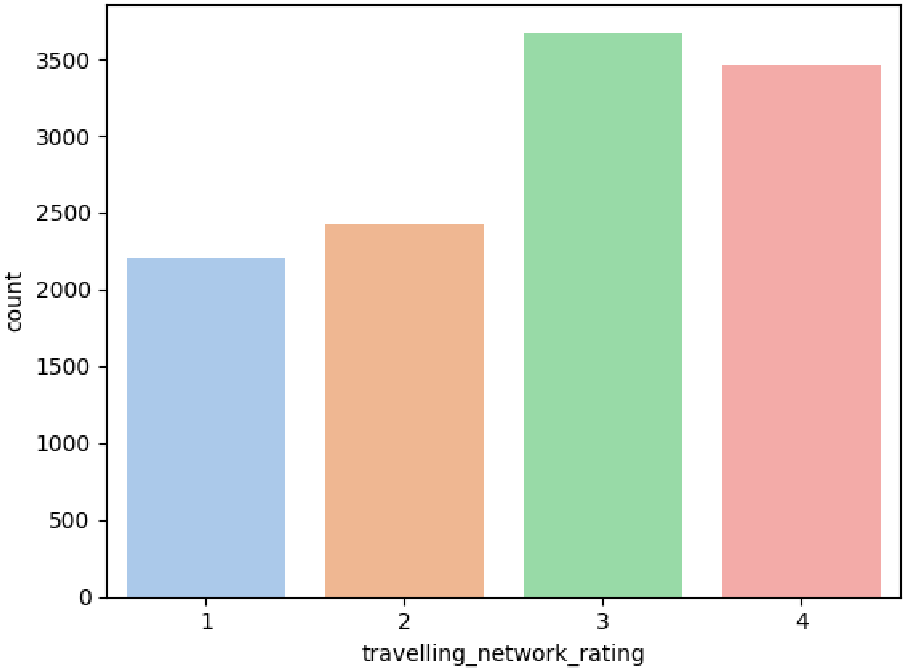


Fig 3. Travelling network

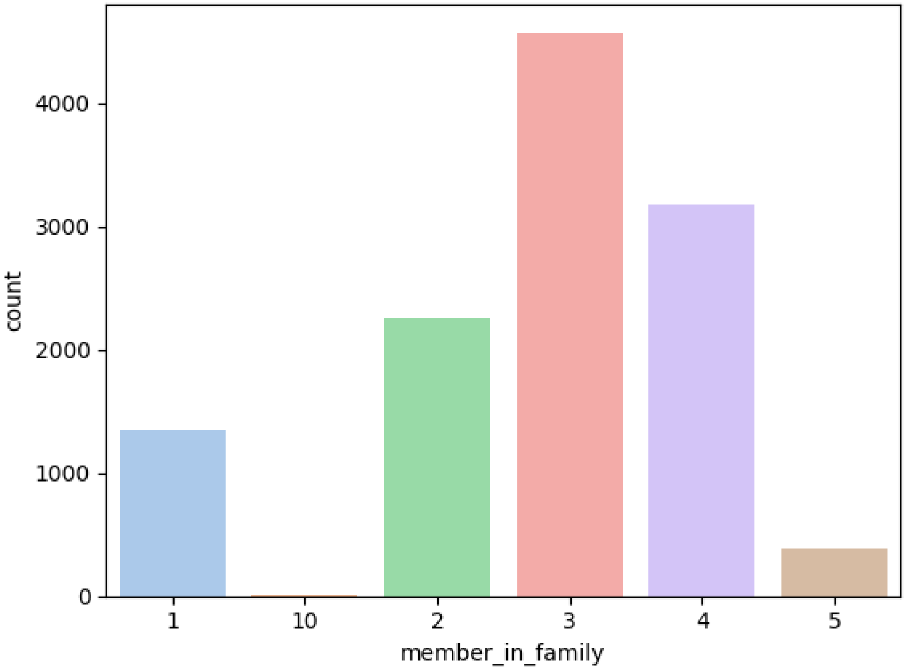


Fig 4. Member in family

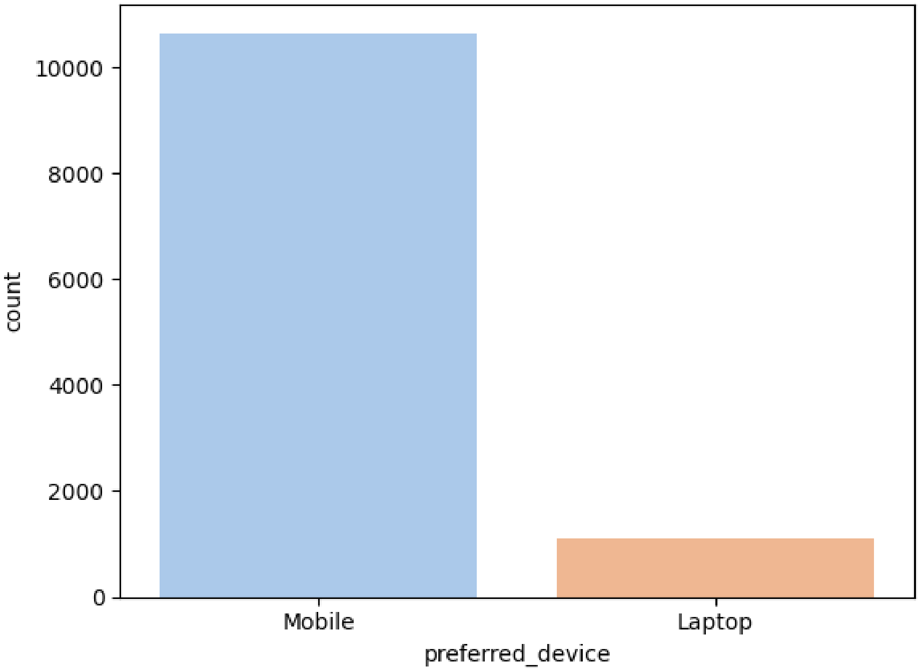


Fig 5. Preferred device

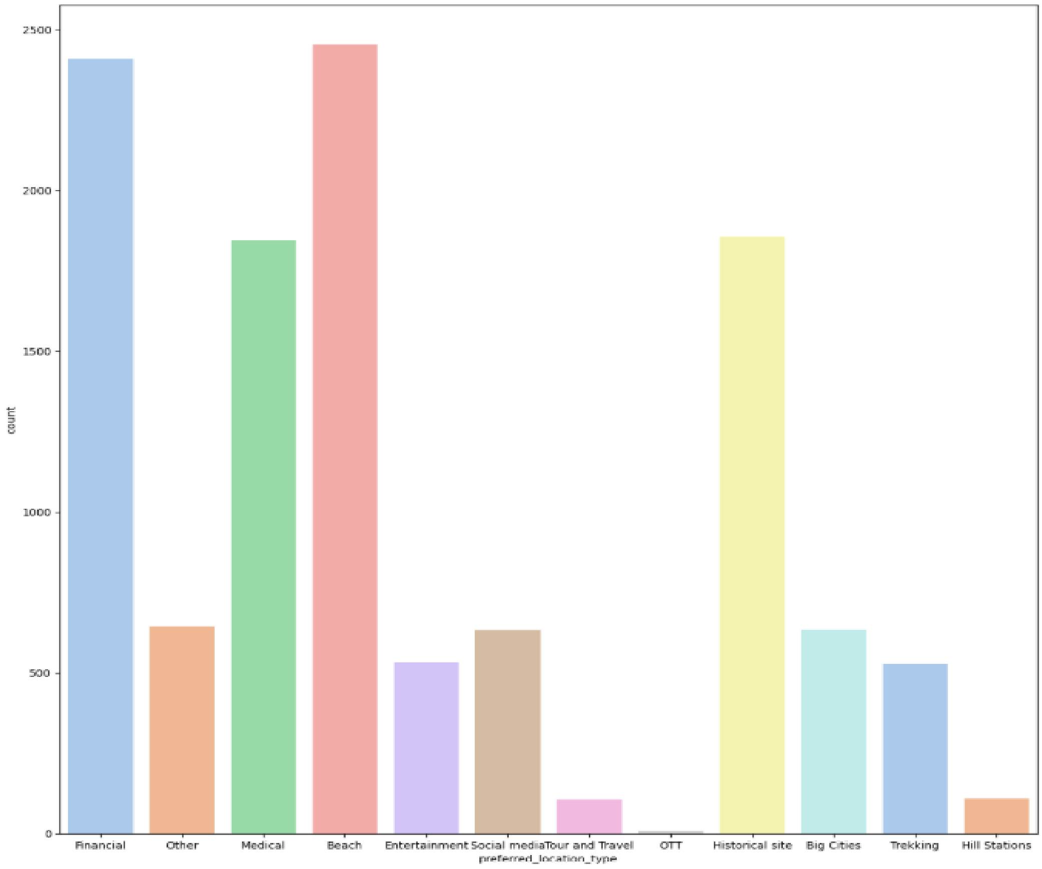


Fig 6. Categorical variable

Count plots are used to analyze categorical variables by showing the frequency of each category.

- The plots show that most users have a travelling_network_rating of 3, suggesting their friends aren't keen on travelling.
- **Family size of 3** is most common among users.

The **majority prefer using Mobile** as their device and favor beach and financial as location types.

Bivariate analysis and Multivariate Analysis

Bivariate analysis examines the relationship between two variables to identify patterns or associations. Pairplots are used for numerical variables, while boxplots are used when comparing categorical and numerical variables.

Pair plot

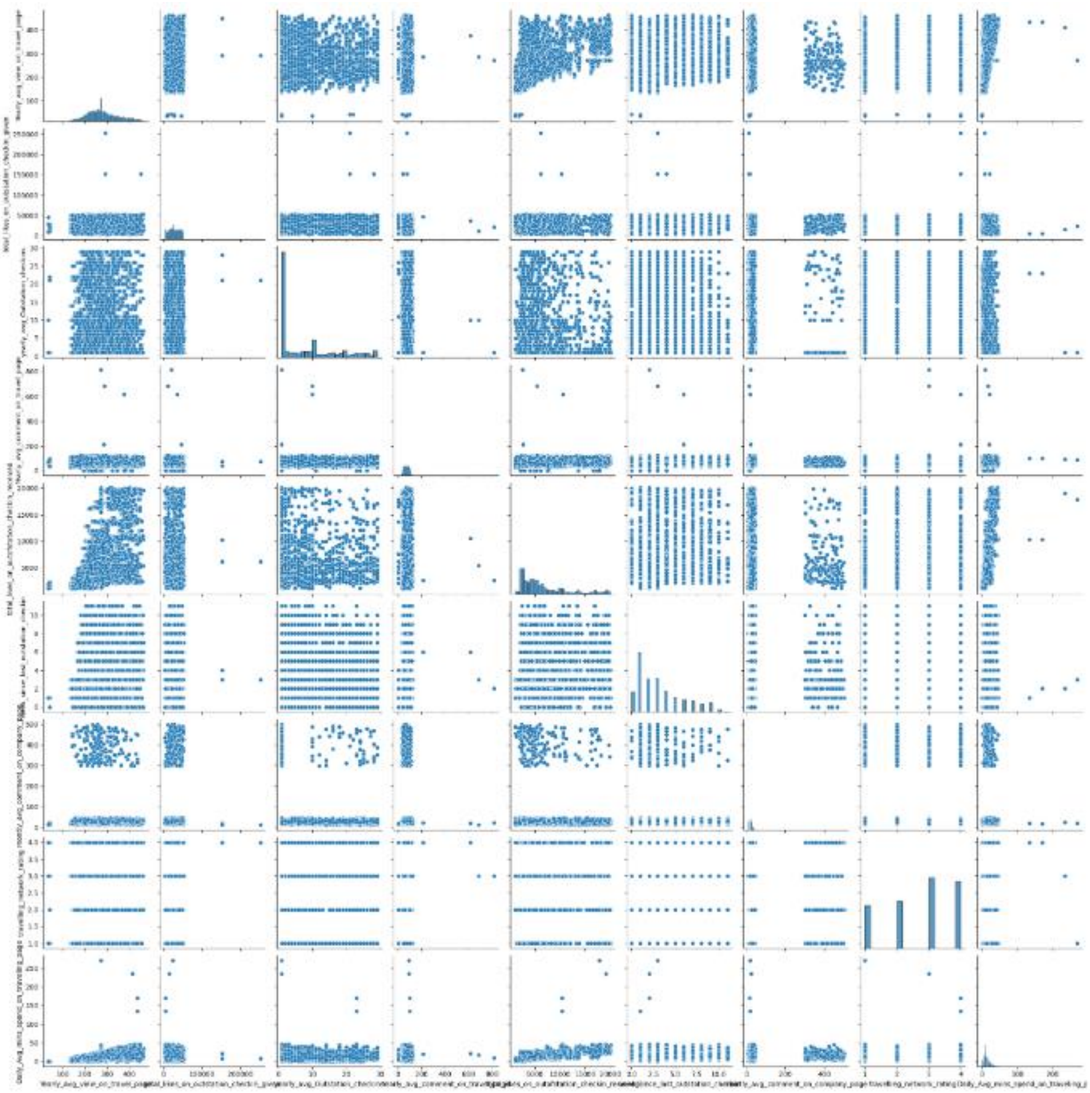


Fig 7. Pair plot

The pair plot shows very little correlation between features, with noticeable skewness in most of them.

Heat map

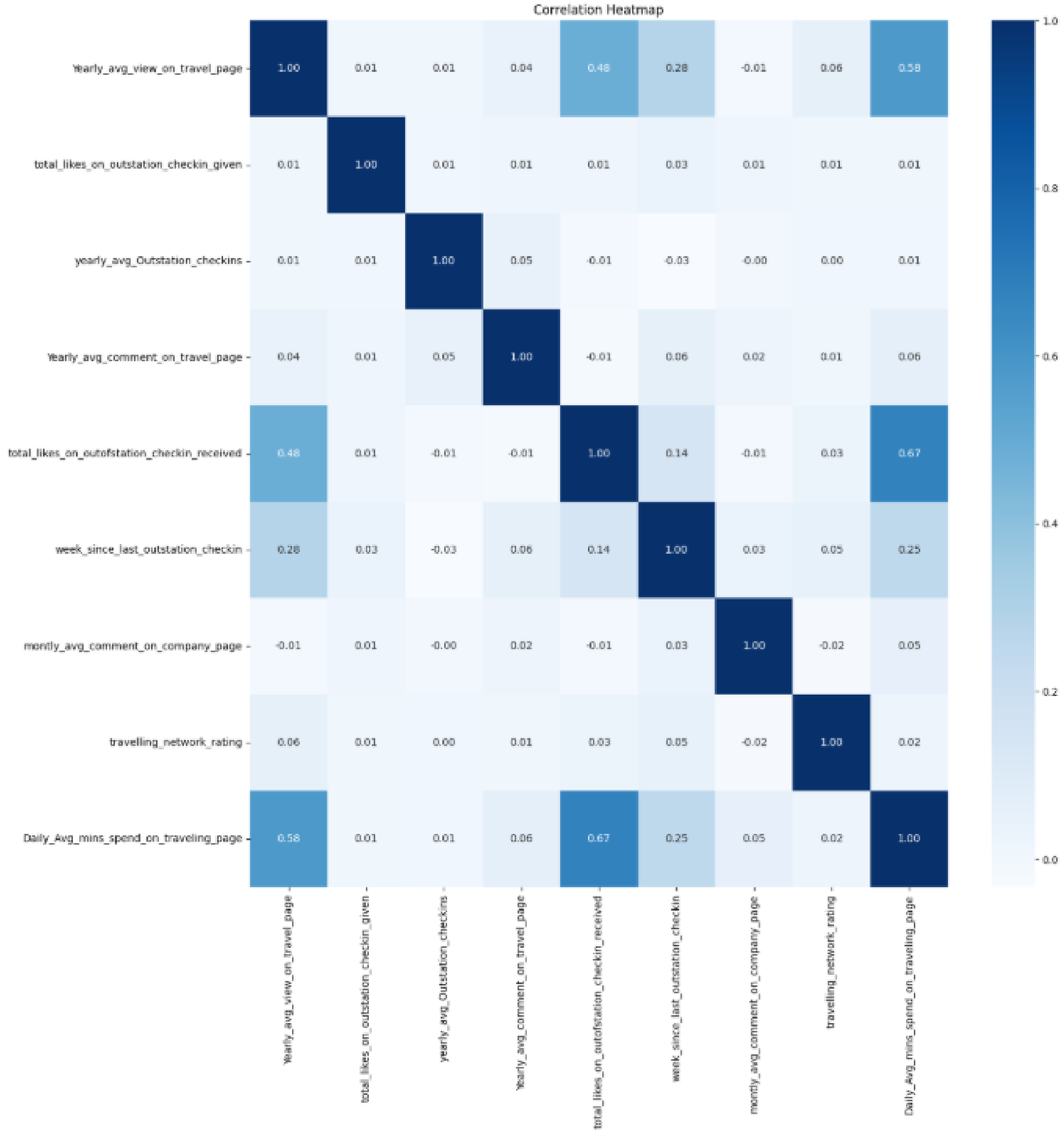
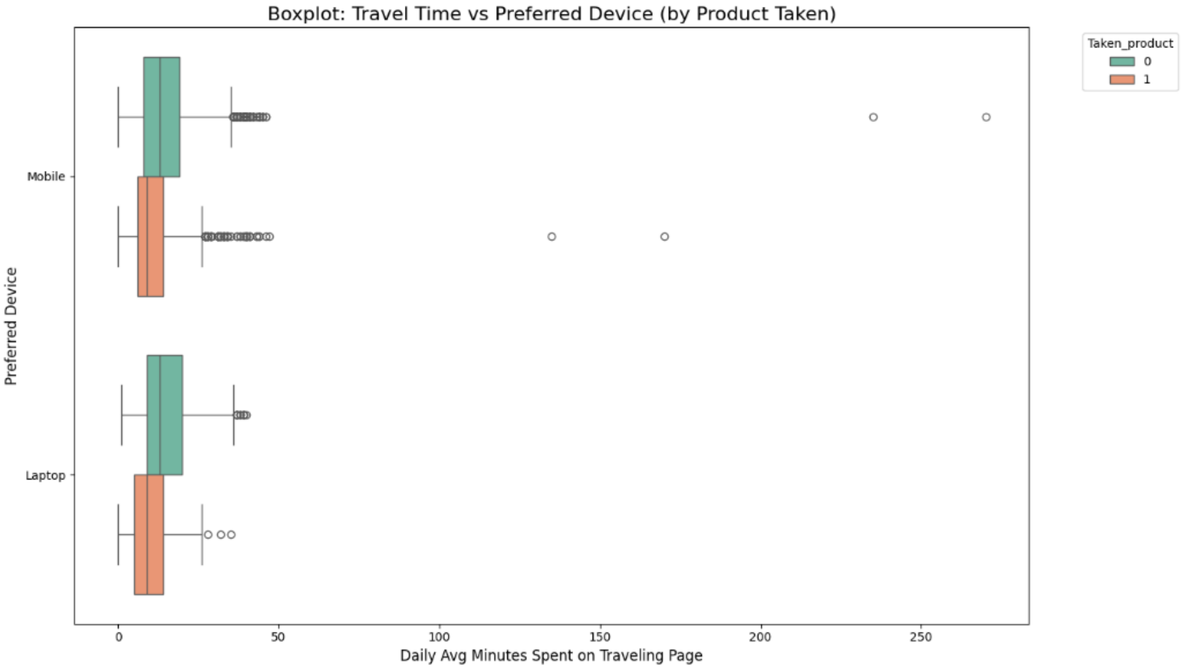
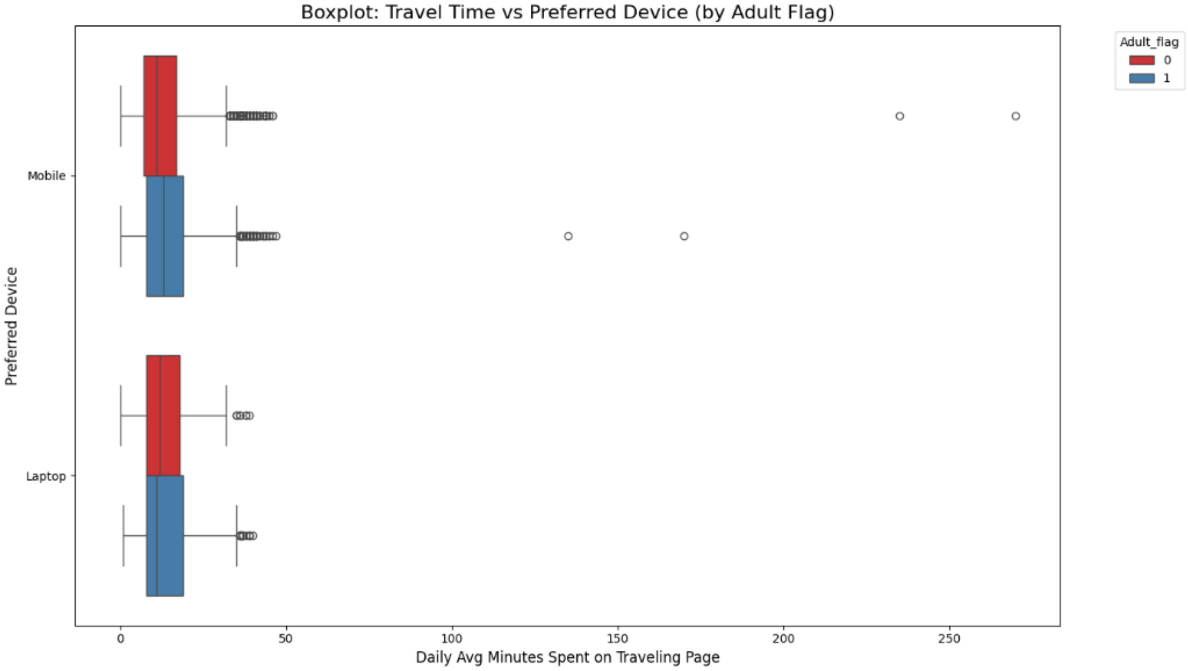


Fig 8. Heat map

Above is the heatmap displaying all numerical variables. The heatmap, used to check associations between numeric variables, confirms that none have a correlation above 0.8, indicating no strong relationships.

Categorical variables - Multivariate Analysis





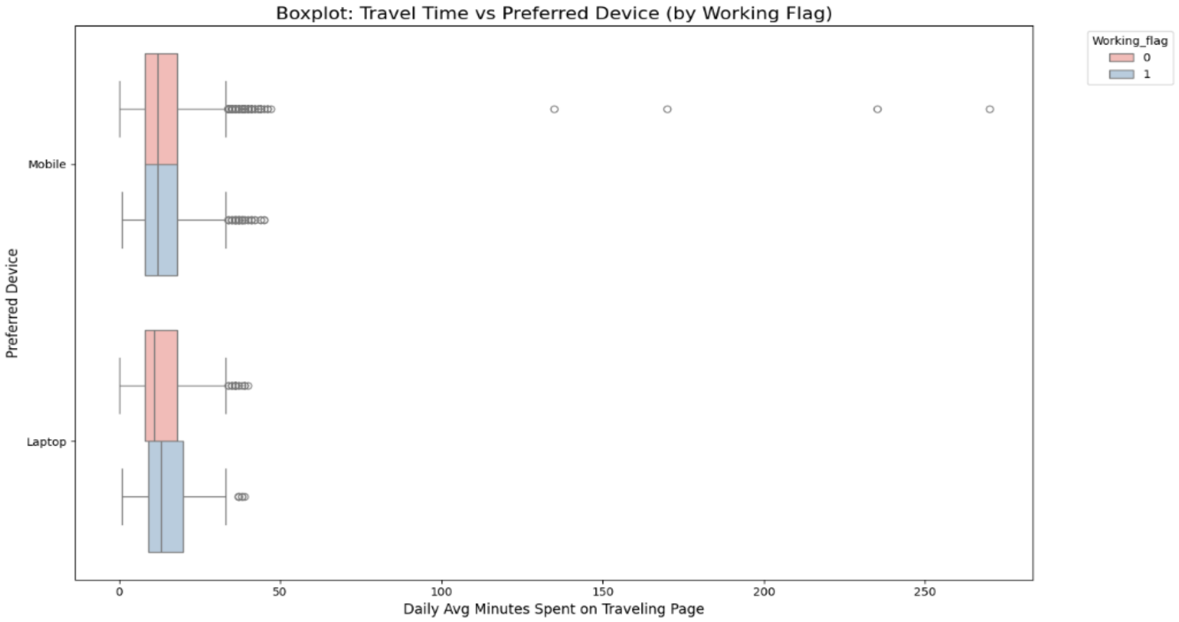


Fig 9. Multivariate analysis

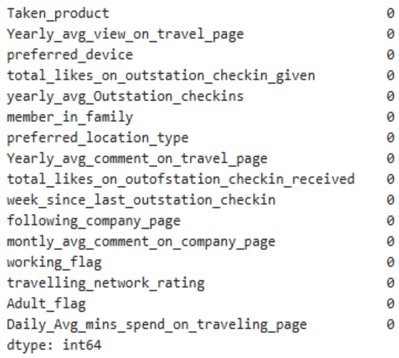
Mobile and laptop users show a **similar median** in terms of product adoption, indicating no major difference in usage preference. Among adult users, **both devices are commonly used**. Interestingly, Mobile users are almost evenly divided between those who have taken the product and those who haven't.

3. Data Cleaning and Pre-processing - Approach used for identifying and treating missing values and outlier treatment (and why) - Need for variable transformation (if any) - Variables removed or added and why (if any)

a. Removal of unwanted variables (if applicable)

Currently, there's no need to drop any variables except the **UserID** column, as it doesn't contribute any meaningful information related to the target variable.

b. Missing Value treatment (if applicable)



```
Taken_product                                 0
Yearly_avg_view_on_travel_page                0
preferred_device                              0
total_likes_on_outstation_checkin_given       0
yearly_avg_Outstation_checkins                0
member_in_family                              0
preferred_location_type                       0
Yearly_avg_comment_on_travel_page             0
total_likes_on_outofstation_checkin_received  0
week_since_last_outstation_checkin            0
following_company_page                        0
montly_avg_comment_on_company_page            0
working_flag                                  0
travelling_network_rating                     0
Adult_flag                                    0
Daily_Avg_mins_spend_on_traveling_page        0
dtype: int64
```

Table 6: Missing value

The variables **Yearly_avg_view_on_travel_page**, **preferred_device**, **total_likes_on_outstation_checkin_given**, **yearly_avg_Outstation_checkins**, **preferred_location_type**, **Yearly_avg_comment_on_travel_page**, and **following_company_page** contained missing values. Object-type variables were imputed using the **mode**, while numerical ones were filled using the **median**, as both are robust against outliers.

c. Outlier treatment (if required)

Several continuous variables contain **outliers** that need to be handled, as algorithms like **Logistic Regression** are sensitive to them. Outliers are capped using the **IQR method**, where values beyond 1.5×IQR from Q1 or Q3 are limited to the respective bounds.

The following variables show clear outliers based on boxplots: **Yearly_avg_view_on_travel_page**, **Total_likes_on_outstation_checkin_given**, **Yearly_avg_comment_on_travel_page**, **Total_likes_on_outofstation_checkin_received**, **Montly_avg_comment_on_company_page**, and **Daily_Avg_mins_spend_on_traveling_page**.

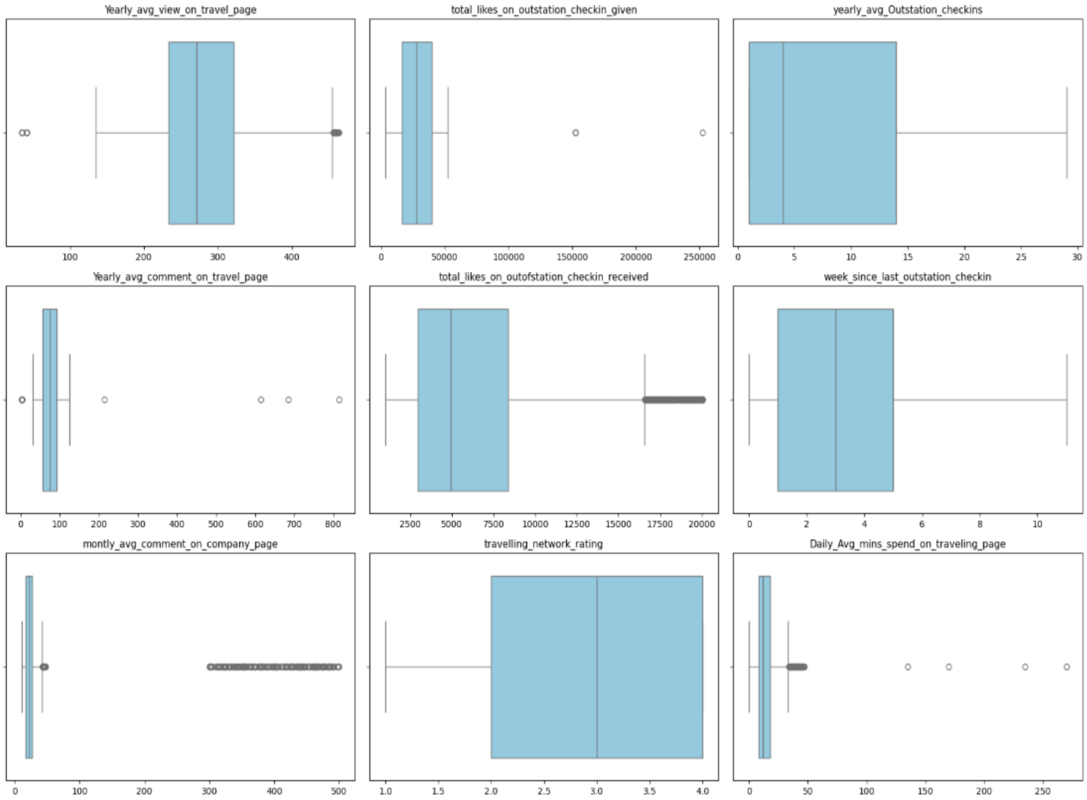


Fig 10. Outliers on boxplot

Outliers in the dataset are handled using the **IQR (Interquartile Range) method**. This technique identifies outliers by calculating the range between the first quartile (Q1) and the third quartile (Q3), where:

- **Q1** marks the 25th percentile,
- **Q3** marks the 75th percentile,
- **IQR = Q3 - Q1**.

Any value falling **below Q1 – 1.5×IQR** or **above Q3 + 1.5×IQR** is treated as an outlier. These limits define the normal range.

Boxplots visually represent this concept, showing the minimum and maximum bounds for typical values. Data points outside these bounds are flagged as outliers and removed from further analysis.

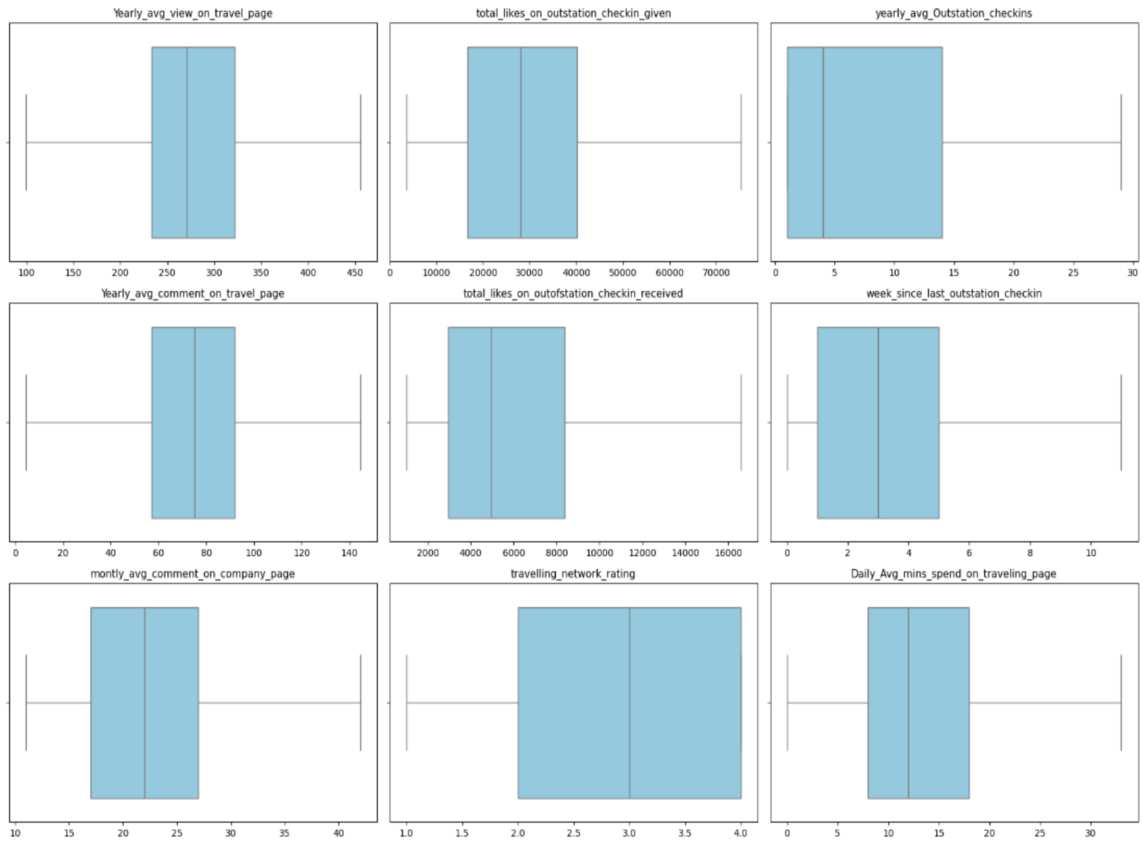


Fig 11. Outliers on boxplot 1

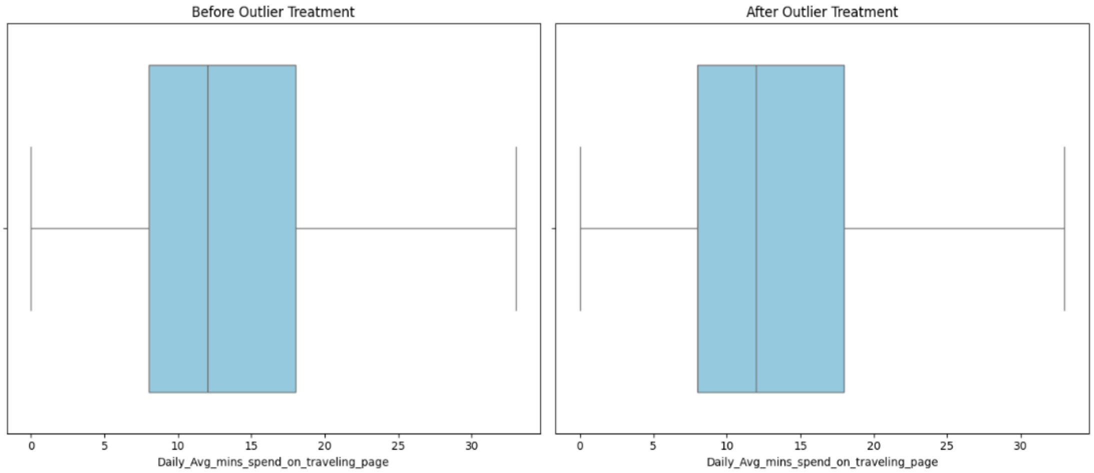


Fig 12. Outlier treatment before and after

The boxplot on the left (Before Outlier Treatment) shows that the data distribution has a slightly wider range, and a few points may have extended slightly beyond the whiskers, indicating the presence of mild outliers.

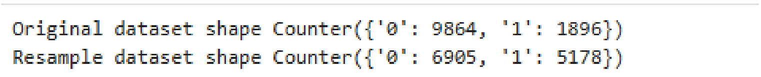
In the boxplot on the right (After Outlier Treatment), the whiskers are more tightened, showing that extreme values have been capped using the IQR method. The overall distribution appears more compact and clean, with potential outliers effectively treated.

d. Variable transformation (if applicable)

Several **data transformations** were performed. The **member_in_family** column had mixed formats (e.g., *3* and *three*), which were standardized to numeric. **Binary variables** were encoded as 0 (No) and 1 (Yes). The **Adult_flag** was simplified to indicate adulthood, and redundant values in **preferred_location_type** were reduced.

Due to varying feature scales, **StandardScaler** was used to normalize **numerical variables**, ensuring consistent data for modeling.

Data Unbalanced



```
Original dataset shape Counter({'0': 9864, '1': 1896})
Resample dataset shape Counter({'0': 6905, '1': 5178})
```

Table 7: Dataset shape counter

The dataset is **imbalanced**, with class ‘1’ having only **1,896 instances** compared to **9,864 instances** of class ‘0’.

To address this, **SMOTE (Synthetic Minority Over-sampling Technique)** was applied, which significantly balanced the data by increasing class ‘1’ to **5,178 instances**, while class ‘0’ has **6,905 instances.**

Hierarchical Clustering

HC was performed on the scaled data using ward linkage.

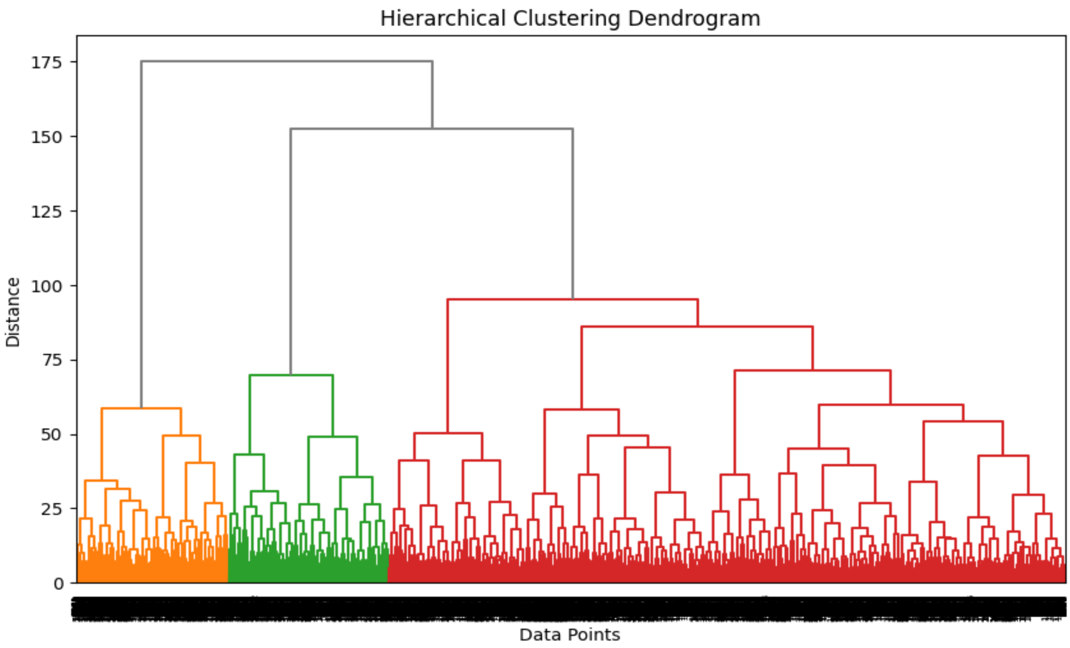


Fig 13. HC dendogram

The dendrogram above shows that the data is divided into **three distinct clusters**. Additionally, using the **countplot** and value_counts() function, we determined the **number of data points** in each cluster.

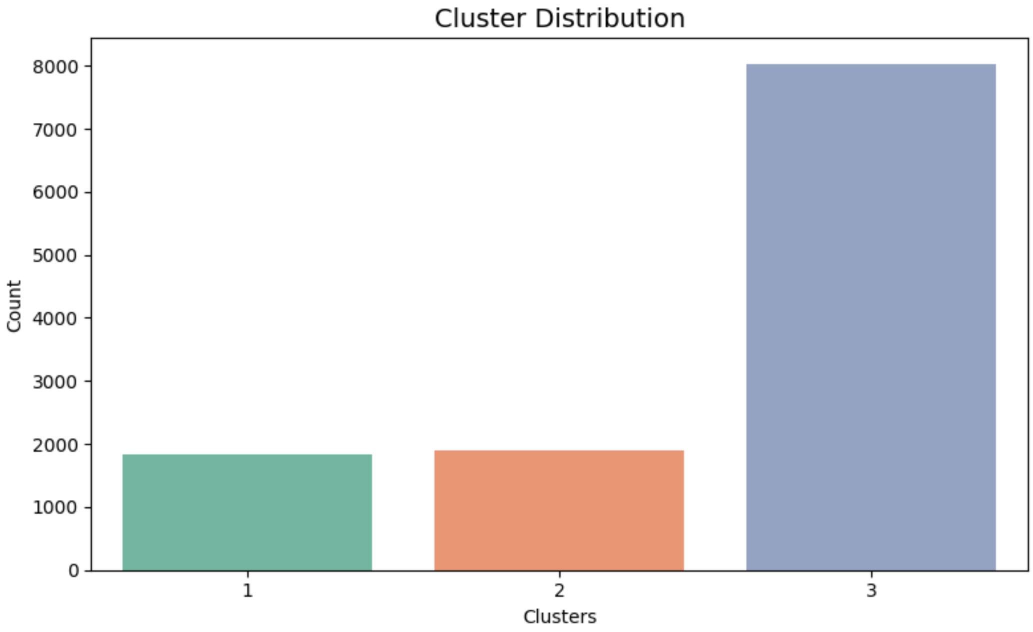
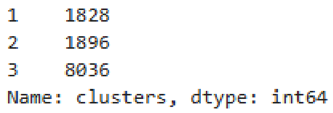


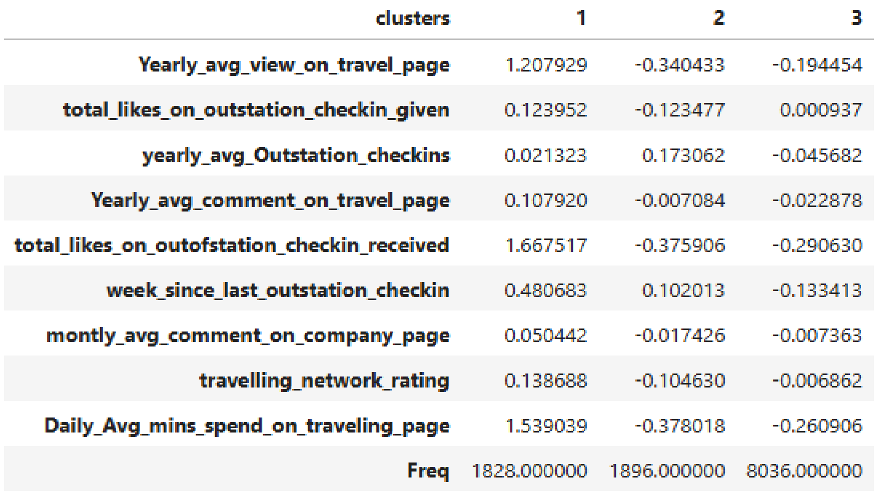
Fig 14. Cluster distribution



```
1    1828
2    1896
3    8036
Name: clusters, dtype: int64
```

Table 8: Cluster value count

The above details indicate that **Cluster 3 contains the highest number** of data points, followed by Cluster 2, and then Cluster 1.



clusters	1	2	3
Yearly_avg_view_on_travel_page	1.207929	-0.340433	-0.194454
total_likes_on_outstation_checkin_given	0.123952	-0.123477	0.000937
yearly_avg_Outstation_checkins	0.021323	0.173062	-0.045682
Yearly_avg_comment_on_travel_page	0.107920	-0.007084	-0.022878
total_likes_on_outofstation_checkin_received	1.667517	-0.375906	-0.290630
week_since_last_outstation_checkin	0.480683	0.102013	-0.133413
montly_avg_comment_on_company_page	0.050442	-0.017426	-0.007363
travelling_network_rating	0.138688	-0.104630	-0.006862
Daily_Avg_mins_spend_on_traveling_page	1.539039	-0.378018	-0.260906
Freq	1828.000000	1896.000000	8036.000000

Table 9: Cluster value

The mean values of each numerical feature for the different clusters are shown above.

Cluster 3 has the most data points, with a positive mean only for total_likes_on_outstation_checkin_given, indicating users mostly liked outstation check-ins last year. While variables like monthly_avg_comment_on_company_page, travelling_network_rating, yearly_avg_Outstation_checkins, and yearly_avg_comment_on_travel_page have slightly negative means close to zero, features such as yearly_avg_view_on_travel_page, total_likes_on_outofstation_checkin_received, week_since_last_outstation_checkin, and daily_avg_mins_spend_on_traveling_page show notably negative means and need improvement.

K-Means Clustering

KMeans and WSS methods were used to determine the optimal number of clusters, which were then assigned to the data points (refer to the Python Notebook for details). The resulting cluster distribution is shown below using countplot and value_counts().

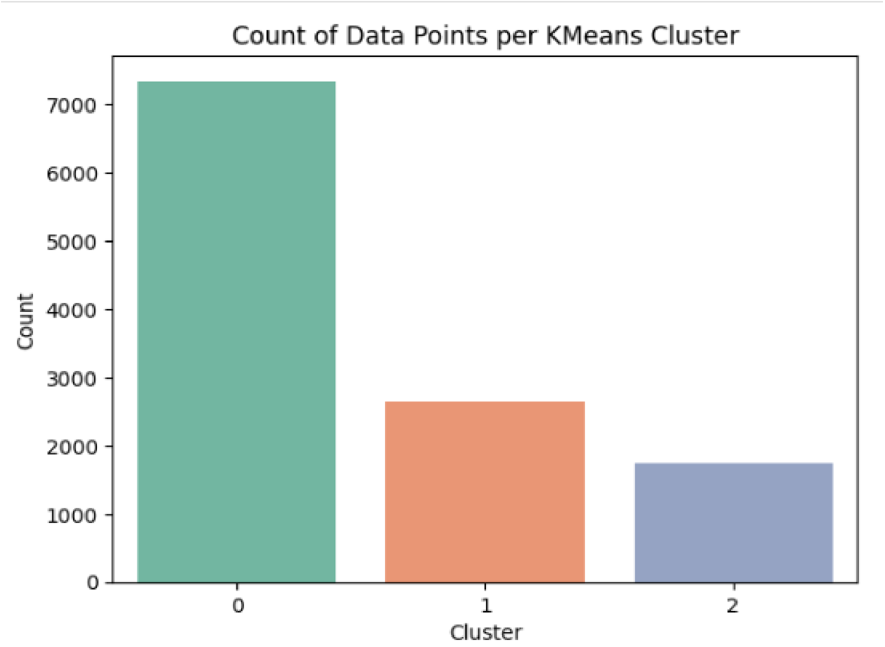
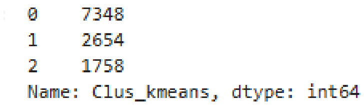
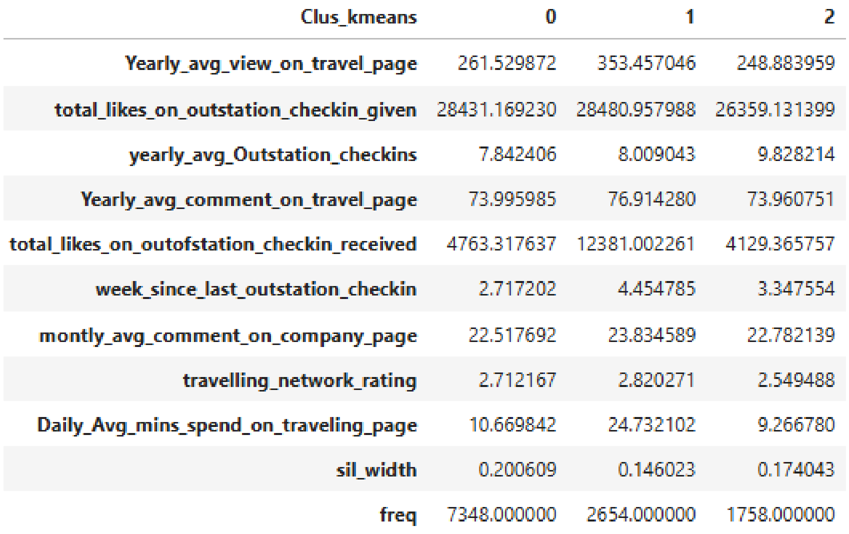


Fig 15. K-means cluster distribution



```
0    7348
1    2654
2    1758
Name: Clus_kmeans, dtype: int64
```

Table 10: K-means cluster value count



Clus_kmeans	0	1	2
Yearly_avg_view_on_travel_page	261.529872	353.457046	248.883959
total_likes_on_outstation_checkin_given	28431.169230	28480.957988	26359.131399
yearly_avg_Outstation_checkins	7.842406	8.009043	9.828214
Yearly_avg_comment_on_travel_page	73.995985	76.914280	73.960751
total_likes_on_outofstation_checkin_received	4763.317637	12381.002261	4129.365757
week_since_last_outstation_checkin	2.717202	4.454785	3.347554
montly_avg_comment_on_company_page	22.517692	23.834589	22.782139
travelling_network_rating	2.712167	2.820271	2.549488
Daily_Avg_mins_spend_on_traveling_page	10.669842	24.732102	9.266780
sil_width	0.200609	0.146023	0.174043
freq	7348.000000	2654.000000	1758.000000

Table 11: K-means cluster value

- **Cluster 0** (7348 users) - **Low engagement** (majority group but low activity); needs awareness-focused campaigns.
- **Cluster 1** (2654 users) **-** Highly active, **ideal for premium offers and retargeting.**
- **Cluster 2** (1758 users) - Moderate activity, respond well to **discounts or promotions**.

4. Model building - Clear on why was a particular model(s) chosen. - Effort to improve model performance.

Model Building and interpretation

Dividing the dataset into 'Predictor' and 'Target' variables

Both data set for '**Laptop' and 'Mobile' devices are split into Predictor and Target variables** (X and Y) respectively before further split into train and test data.

Normalize the 'predictor' variables, as **some models are sensitive to differences in variable scales**. **Use StandardScaler from Sklearn** to standardize the numerical data to a common scale.

Laptop Dataset	Mobile Dataset
Size of X_train for laptop: (775, 15)	Size of X_train for mobile: (7456, 15)
Size of X_test for laptop: (333, 15)	Size of X_test for mobile: (3196, 15)
Size of y_train for laptop: (775,)	Size of y_train for mobile: (7456,)
Size of y_test for laptop: (333,)	Size of y_test for mobile: (3196,)

Table 12. Size of dataset

Choice of Models

We approached this binary classification problem (Target: Yes = 1, No = 0) by splitting the data into two sets: Mobile and Laptop users. Separate models were built for each. To handle class imbalance, we used SMOTE.
We applied multiple classification models: CART, Random Forest, Logistic Regression, LDA, and KNN—all known for strong performance in such tasks.

- **CART** performs well on large datasets.
- **Random Forest**, an ensemble method, enhances accuracy.
- **Logistic Regression** is fast, simple, and effective when assumptions are met.
- **KNN** is easy to implement, needing just K-value and distance function.

Choice of model Evaluation Metrics

We focus on **Precision** because missing a potential client is more costly than targeting someone unlikely to buy. Losing potential clients impacts the business, making Precision the key metric.

Recall and F1-score were also considered. Recall measures how well the model identifies actual positives—higher recall means more positive cases correctly detected.

Hyperparameter Tuning

Tune hyperparameters to prune trees and prevent overfitting, and to reduce cost function complexity in non-tree models.

XGBoost Classifier

- **Number of Estimators**: Number of trees in the ensemble. More trees generally lead to a **more robust and consistent model**.
- **Learning Rate**: Controls the **step size** toward the optimal point. Lower values improve accuracy but **increase training time and resource use**.
- **Max Depth**: Limits how deep each tree can grow. Higher depth can lead to **overfitting** if not controlled.
- **Min Child Weight**: Minimum sum of instance weight in a node required for a split. Lower values allow more splits; higher values **prevent overgrowth**.
- **Subsample**: Fraction of training data used in each boosting round, helping to **reduce overfitting**.

Logistic Regression

- Regularization: Controls the **strength of penalty**-higher values reduce the penalty's impact.

- Penalty: Specifies the **type of regularization** used to guide the model toward the optimal fit.
- Solver Function: Determines the **optimization algorithm** used to minimize the loss function and find the best fit.

Decision Tree

- **Max Depth:** Limits how deep the tree can grow. By default, it grows until no further splits are possible, which can lead to overfitting.
- **Max Features:** Number of features to consider at each split. A common starting point is half the total features.
- **Min Samples Leaf:** Minimum samples needed for a node to become a leaf. Higher values reduce overfitting by limiting leaf creation.
- **Min Samples Split:** Minimum samples required to split an internal node. Increasing this helps prune the tree and prevent overfitting.

Random Forest

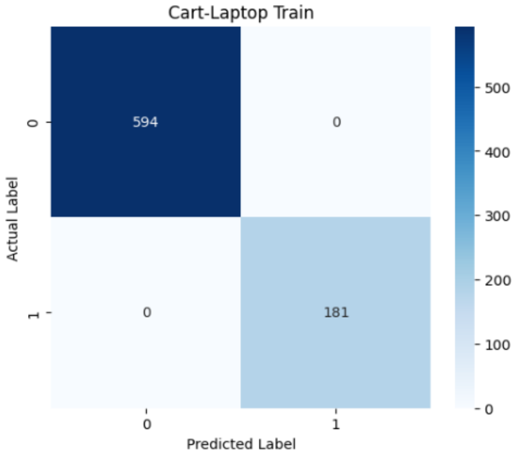
- **max_depth:** Maximum depth of each tree. Unlimited depth can cause overfitting; setting a limit helps control it.
- **max_features**: Number of features to consider at each split. A good starting point is half of the available features.
- **min_samples_leaf:** Minimum samples needed to form a leaf node. Higher values help reduce overfitting by avoiding overly specific leaves.
- **min_samples_split:** Minimum samples required to split a node. Increasing this value helps prune the tree and prevent overfitting.
- **n_estimators:** Number of trees in the ensemble. More trees generally lead to a more stable and robust model.

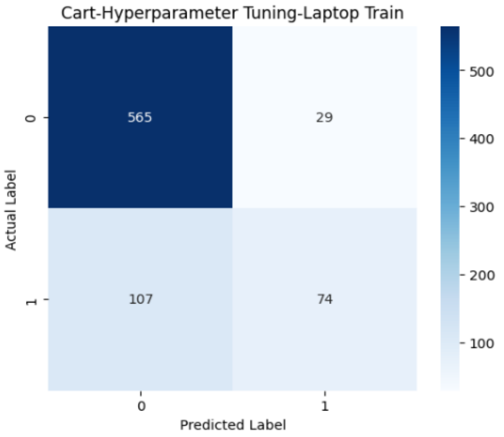
5. Model validation - How was the model validated ? Just accuracy, or anything else too ?

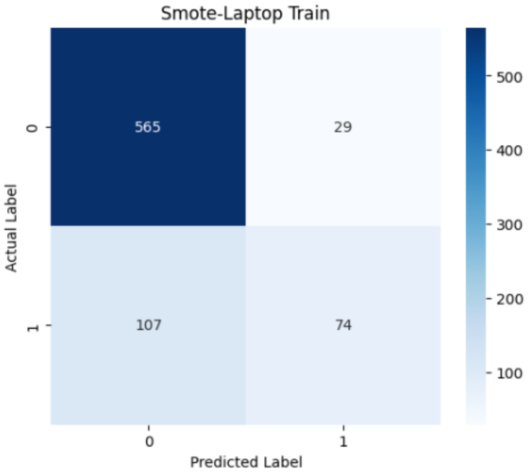
Models for Laptop

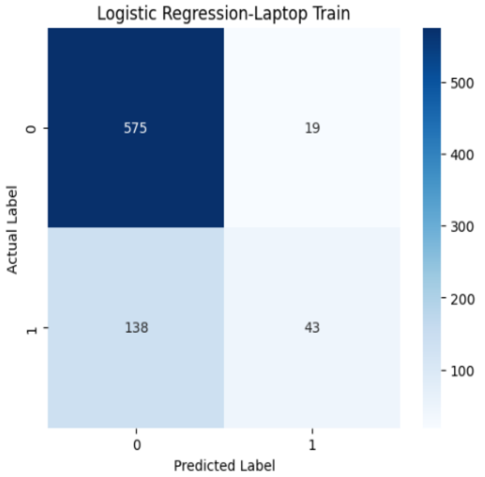
Performance metrics for models - Confusion Matrix

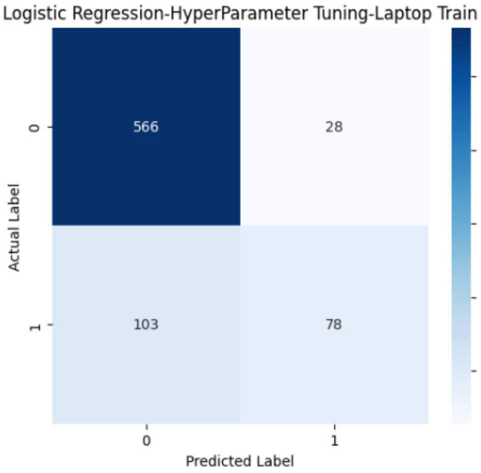
Laptop Train Data

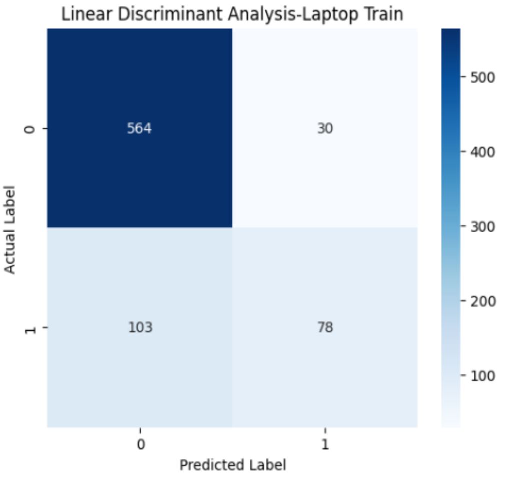


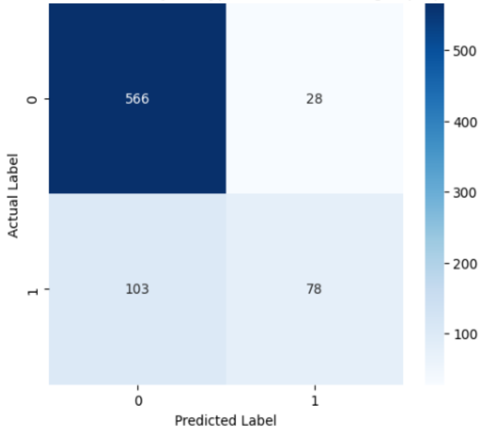




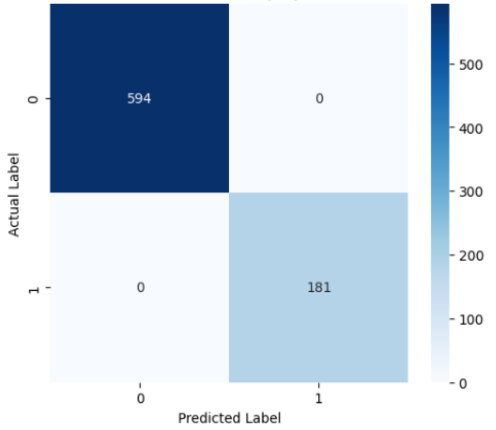




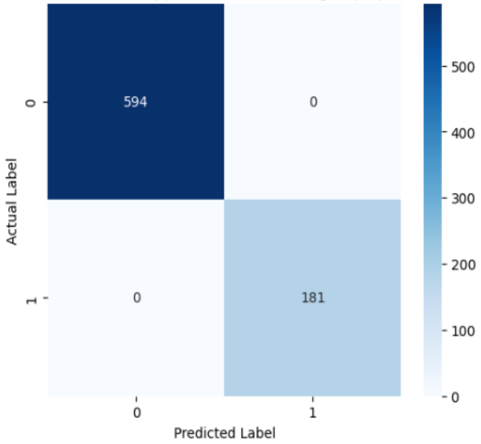




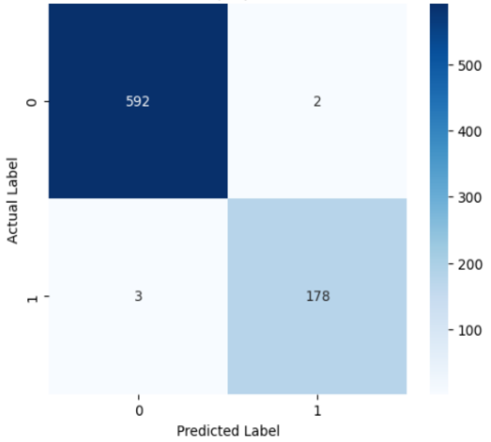
Linear Discriminant Analysis-HyperParameter Tuning-Laptop Train



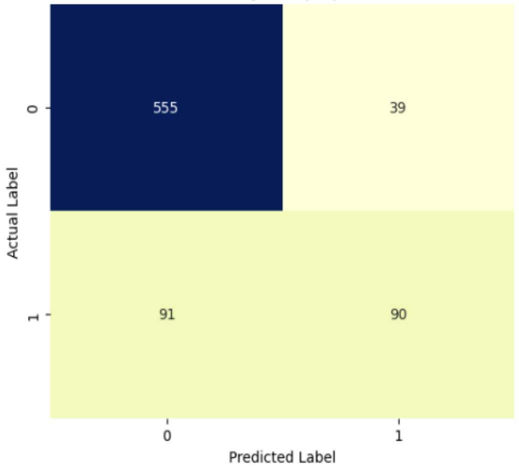
Random Forest - Laptop Train



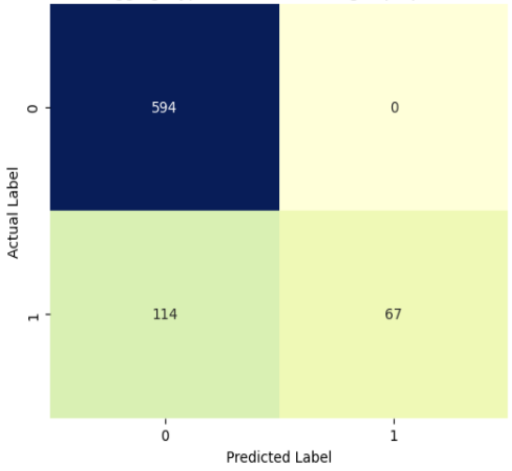
Random Forest-HyperParameter Tuning-Laptop Train



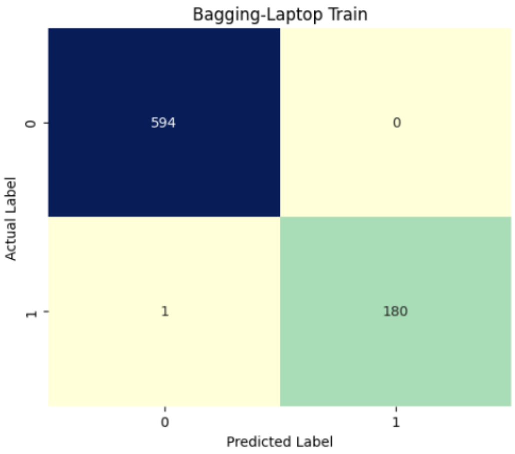
KNN-Laptop Train

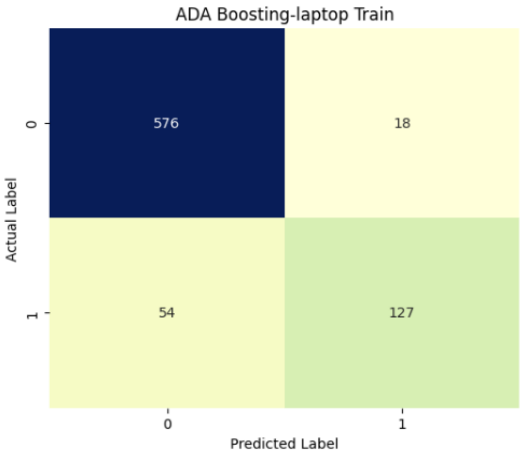


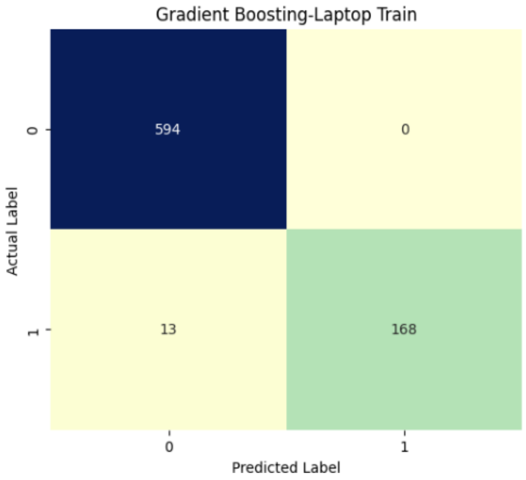
Naive Bayes-Laptop Train

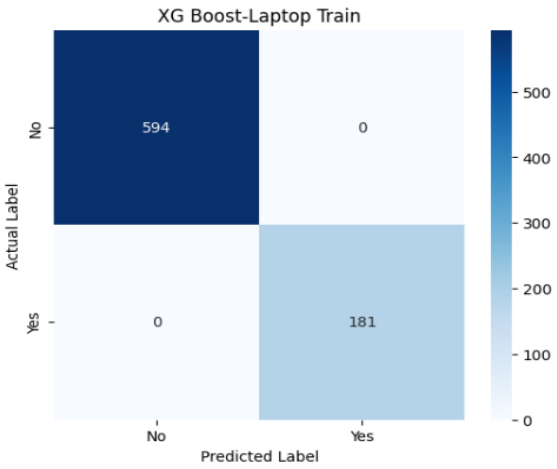


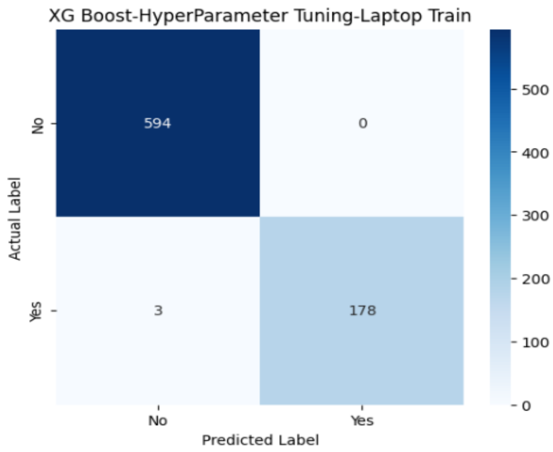
Bagging-HyperParameter Tuning-Laptop Train

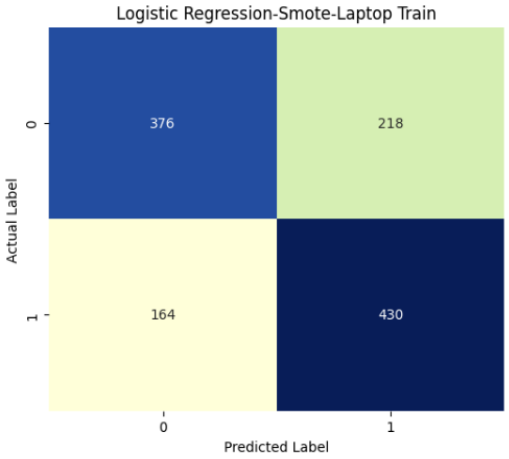

Bagging-Laptop Train


ADA Boosting-laptop Train


Gradient Boosting-Laptop Train


XG Boost-Laptop Train


XG Boost-HyperParameter Tuning-Laptop Train


Logistic Regression-Smote-Laptop Train

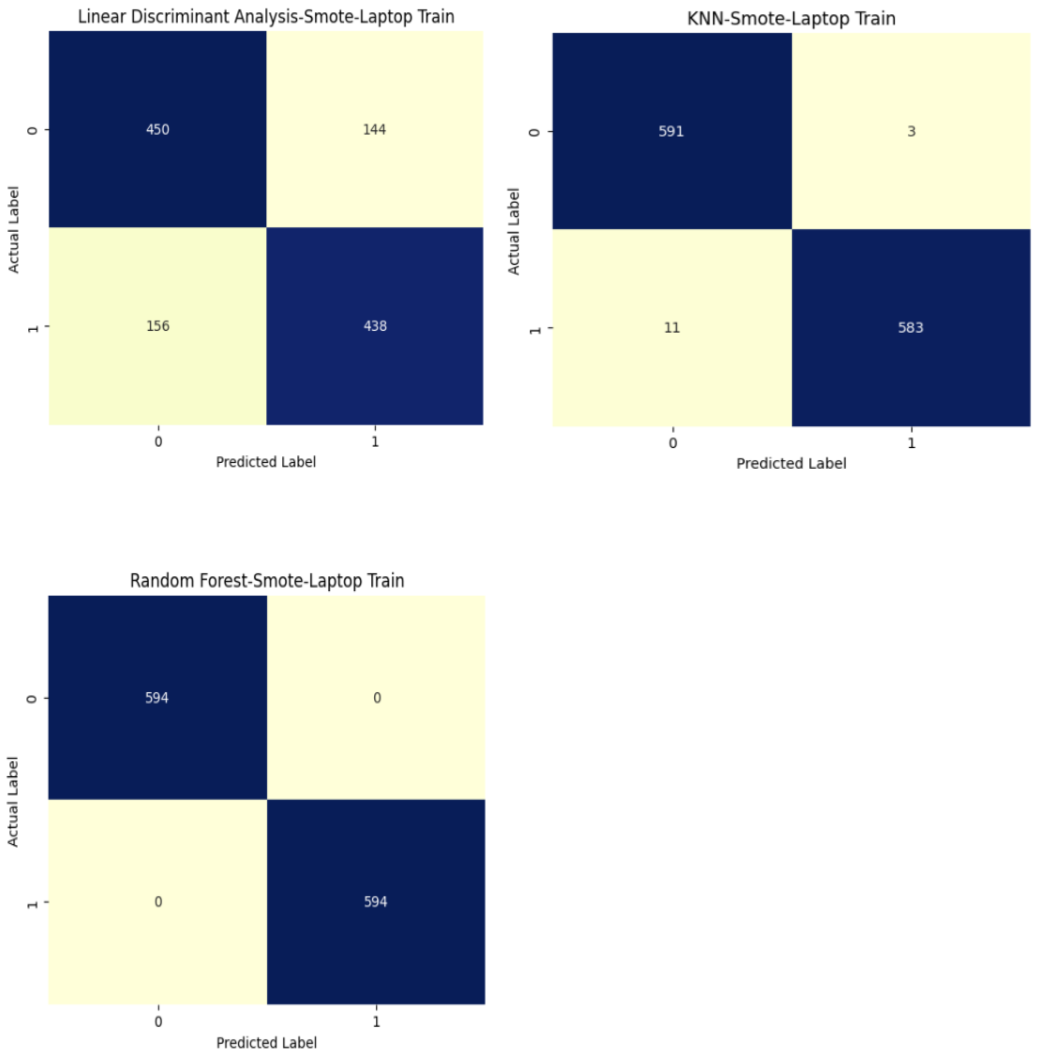
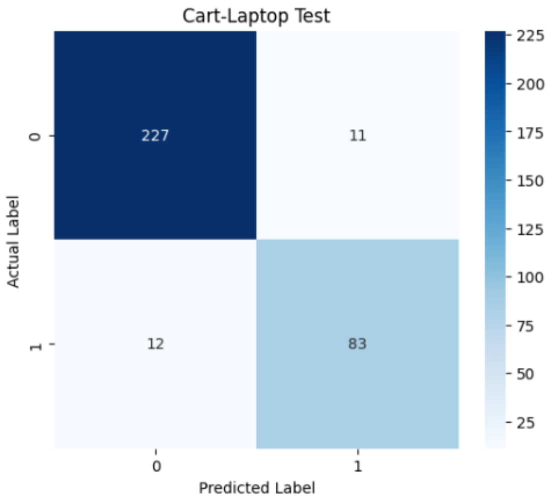


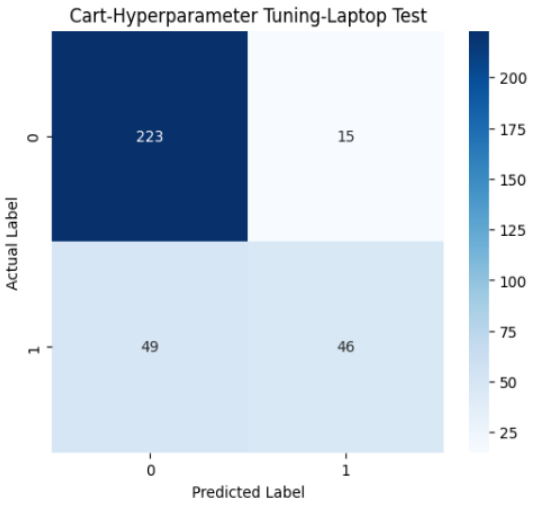
Fig 16. Laptop Train Data

Cart, Random Forest, and XGBoost have the lowest number of false positives (Type II errors) for the Positive Class (1).

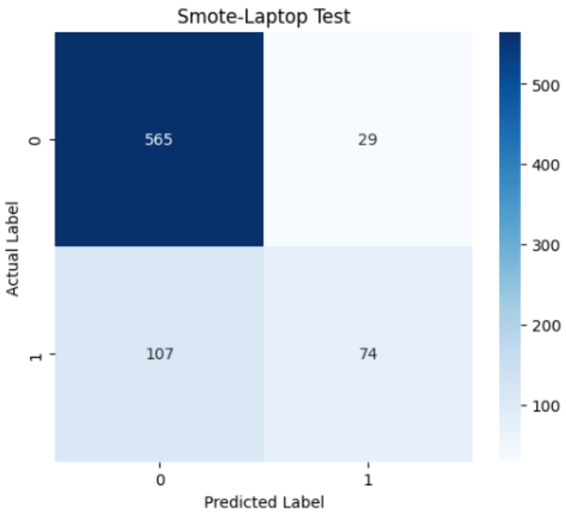
Laptop Test Data



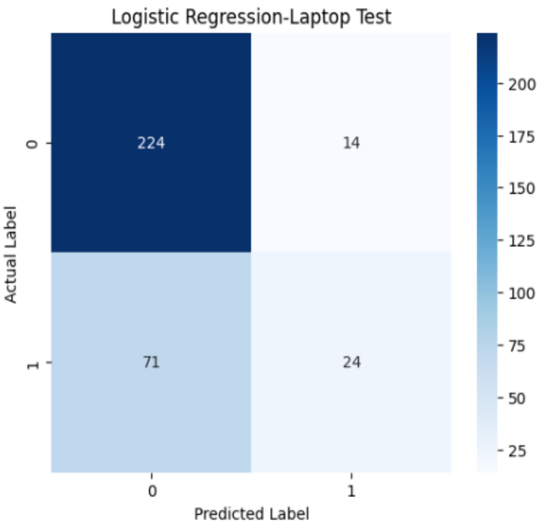
Cart-Laptop Test



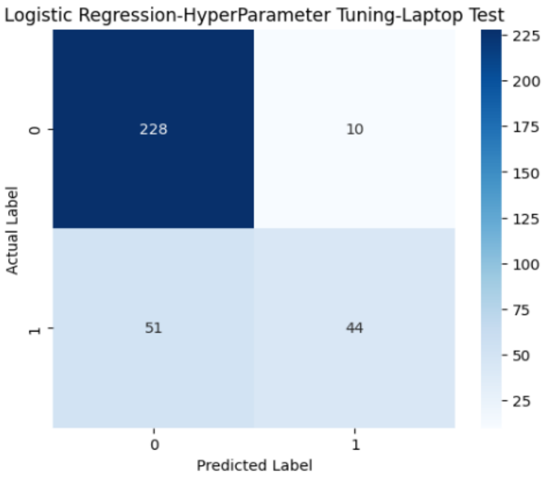
Cart-Hyperparameter Tuning-Laptop Test



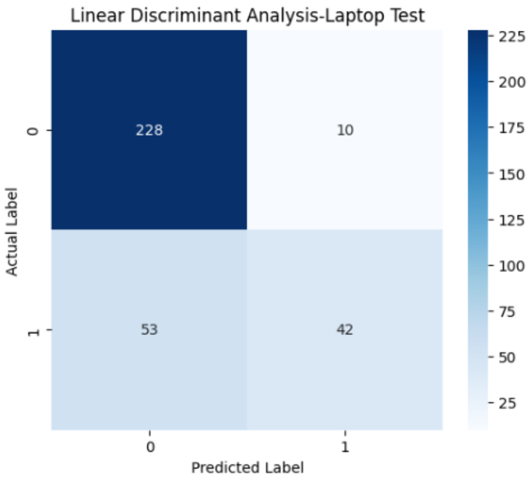
Smote-Laptop Test



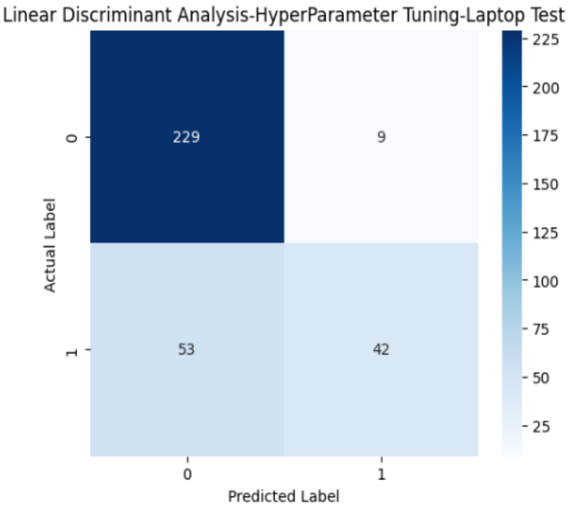
Logistic Regression-Laptop Test



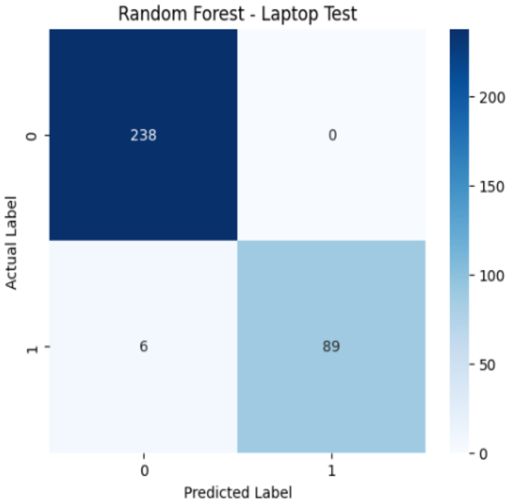
Logistic Regression-HyperParameter Tuning-Laptop Test



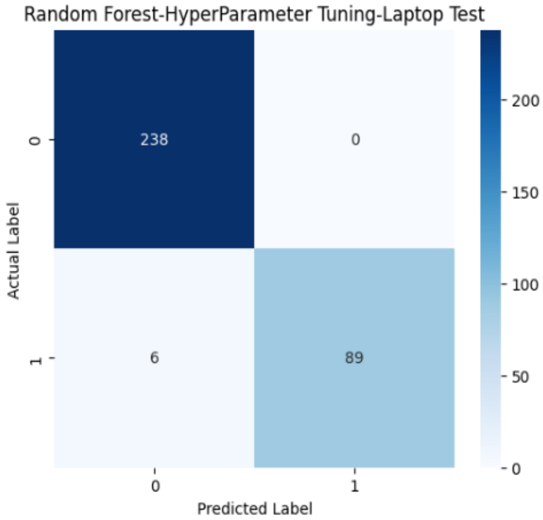
Linear Discriminant Analysis-Laptop Test



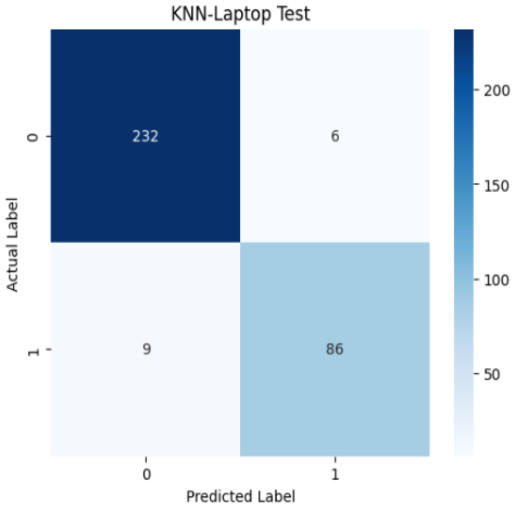
Linear Discriminant Analysis-HyperParameter Tuning-Laptop Test



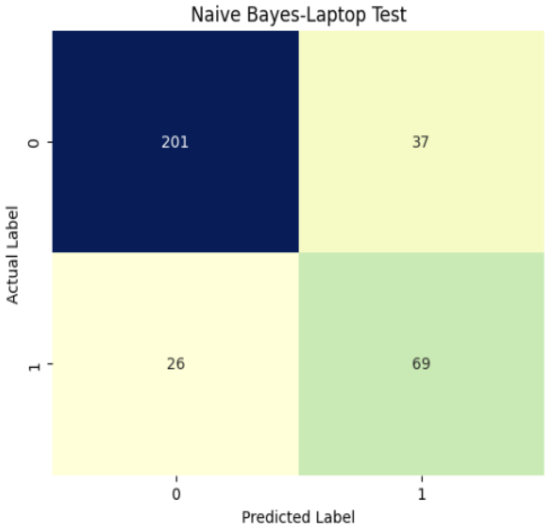
Random Forest - Laptop Test



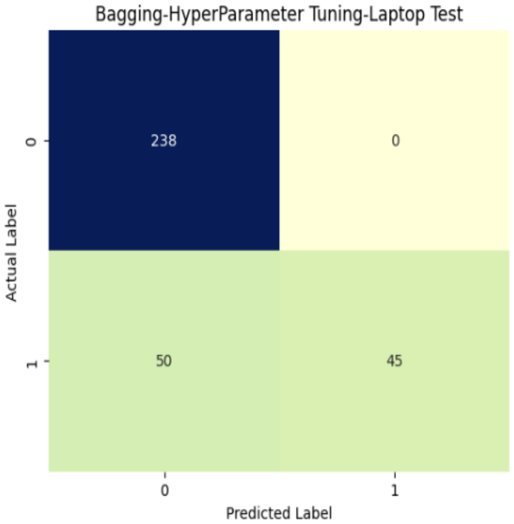
Random Forest-HyperParameter Tuning-Laptop Test



KNN-Laptop Test

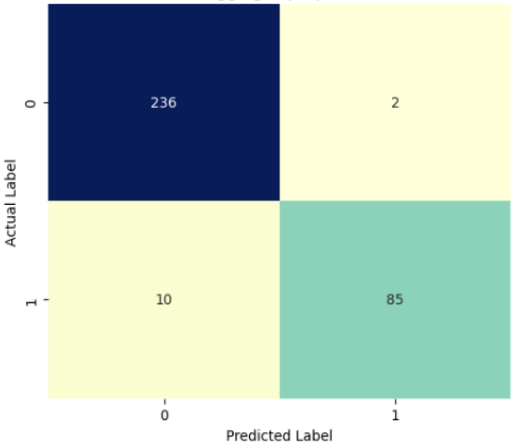


Naive Bayes-Laptop Test

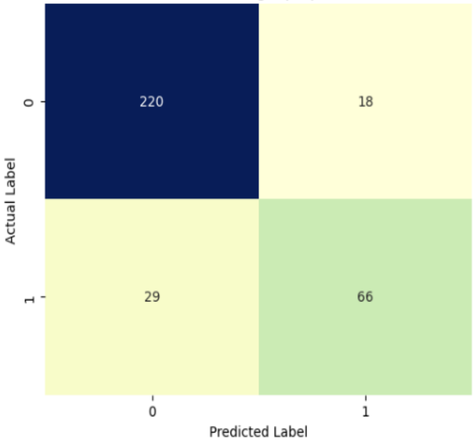


Bagging-HyperParameter Tuning-Laptop Test

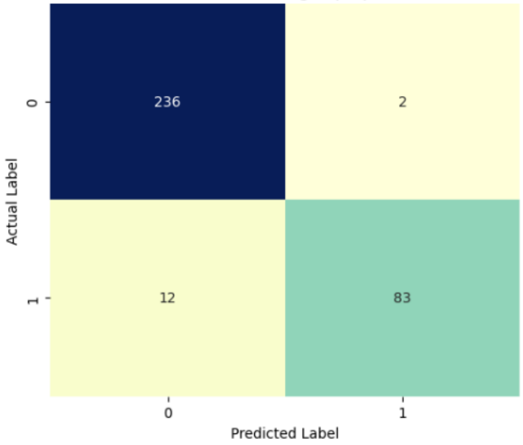
Bagging-Laptop Test



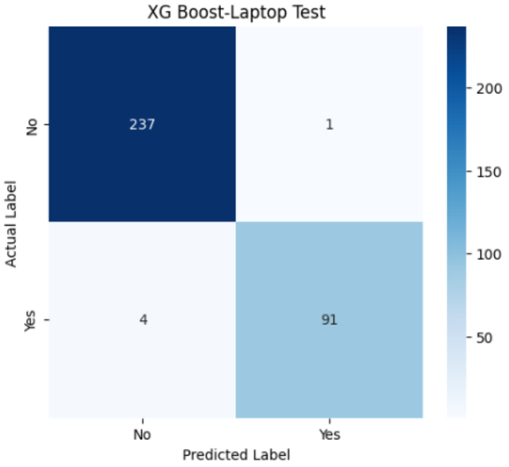
ADA Boosting-laptop Test



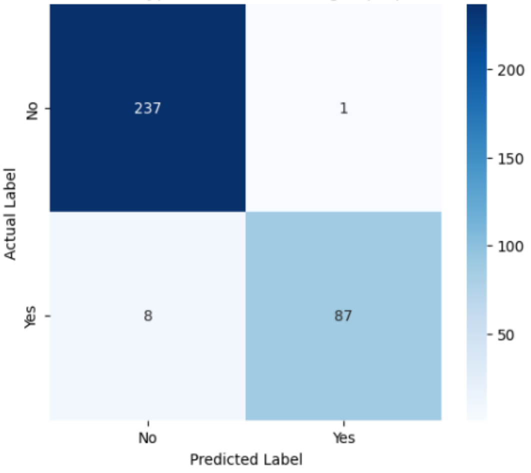
Gradient Boosting-Laptop Test



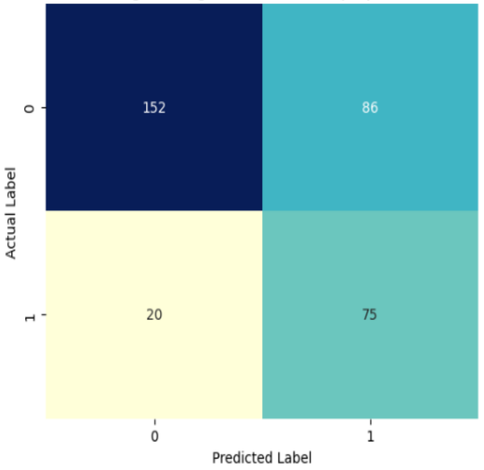
XG Boost-Laptop Test



XG Boost-HyperParameter Tuning-Laptop Test



Logistic Regression-Smote-Laptop Test



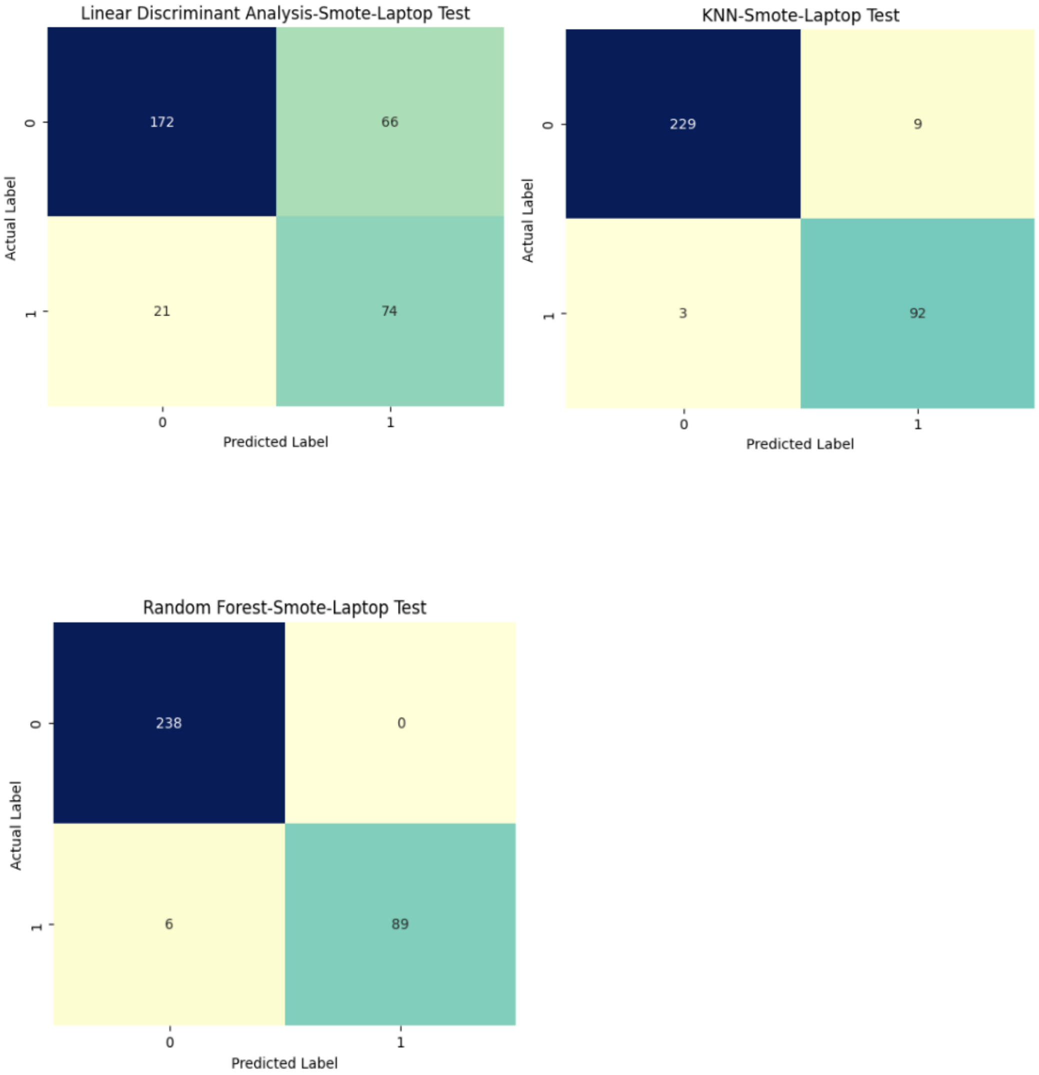


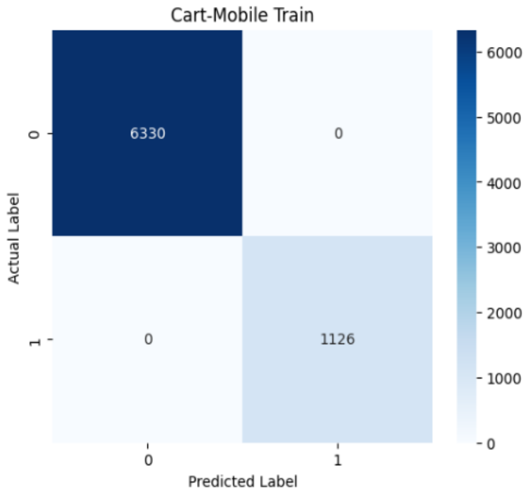
Fig 17. Laptop Test Data

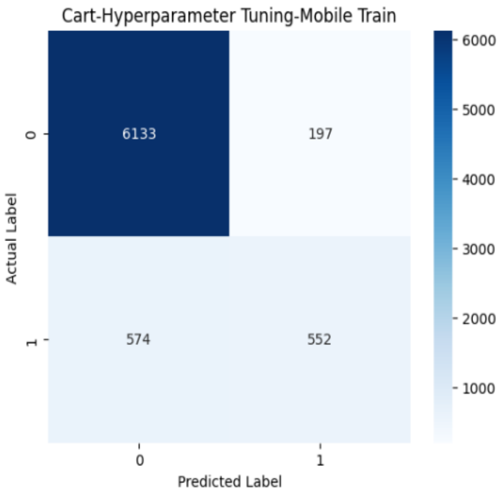
Random Forest and XGBoost show the lowest number of false positives (Type II errors) in the Positive Class (1).

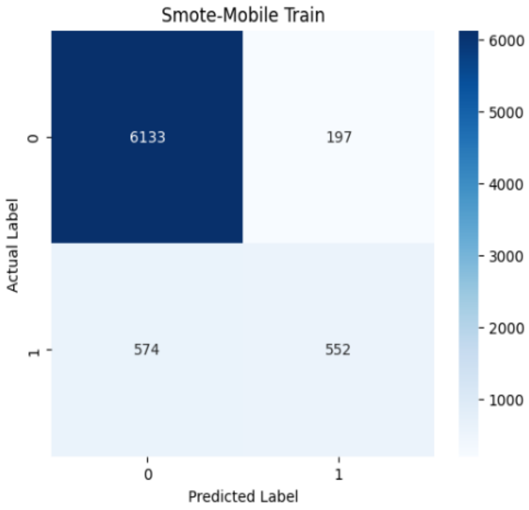
Models for Mobile

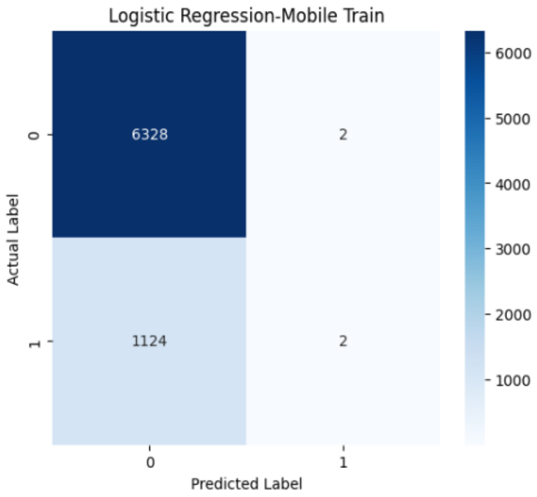
Performance metrics for models - Confusion Matrix

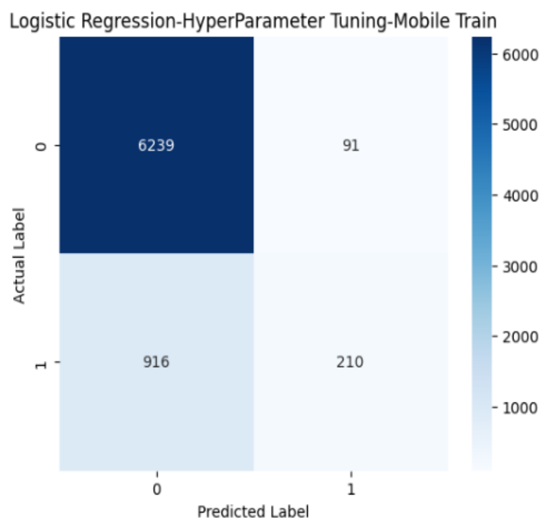
Mobile Train Data

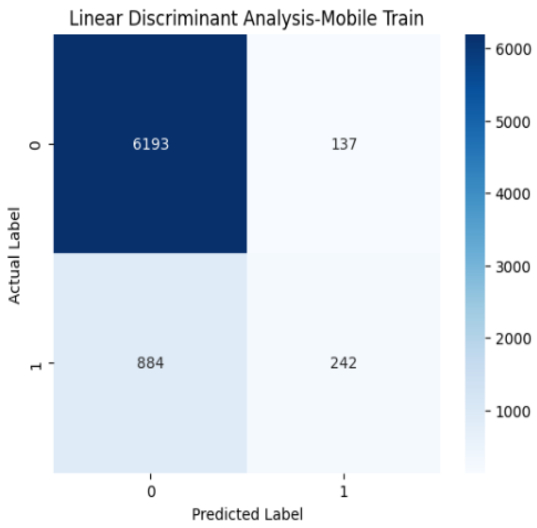


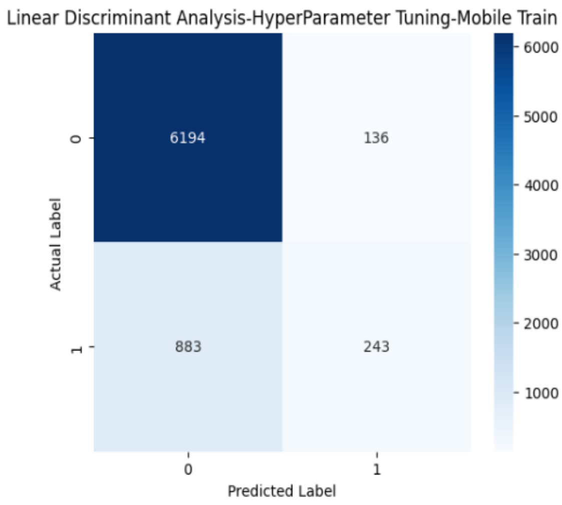


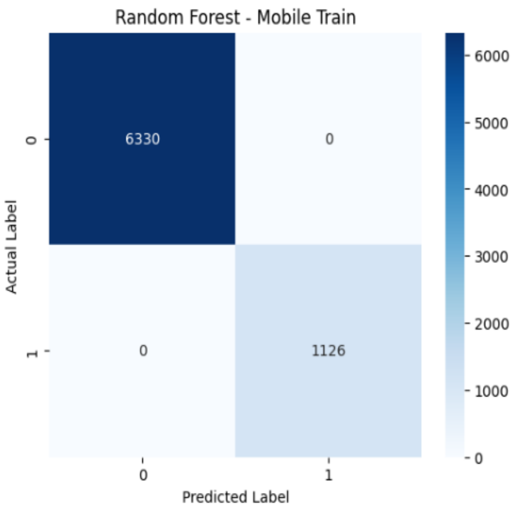


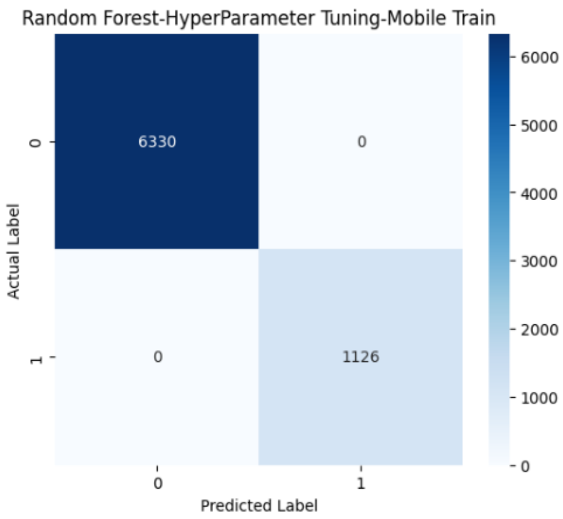


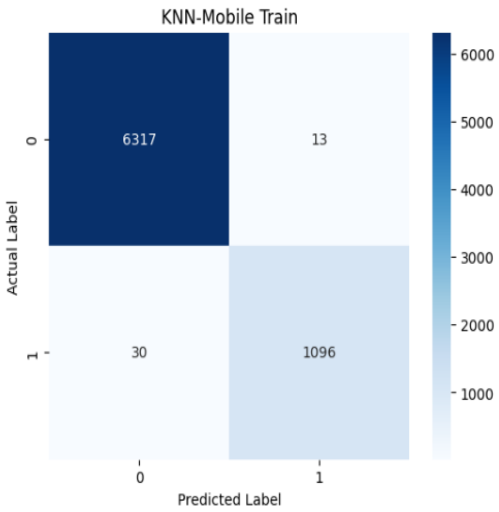

Logistic Regression-HyperParameter Tuning-Mobile Train

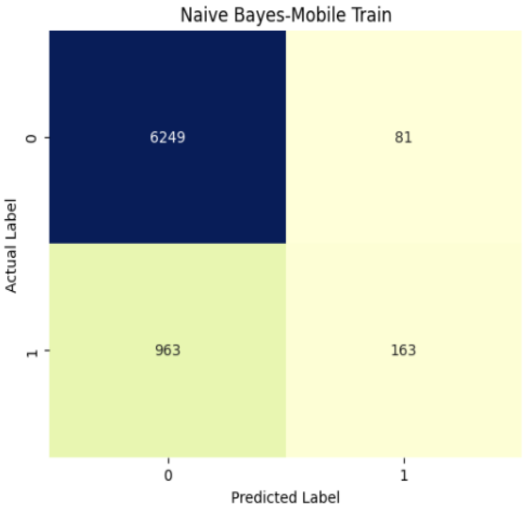

Linear Discriminant Analysis-Mobile Train

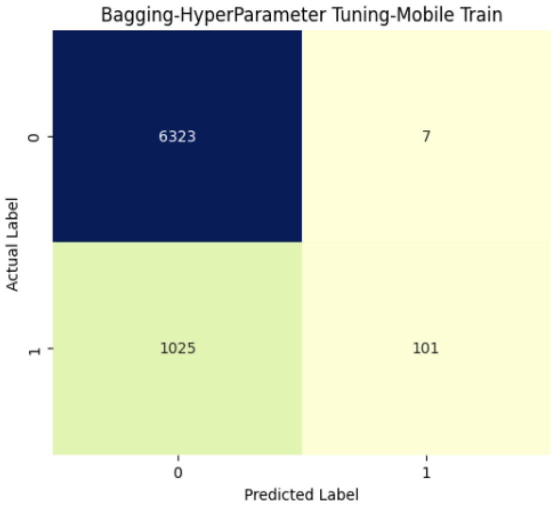

Linear Discriminant Analysis-HyperParameter Tuning-Mobile Train

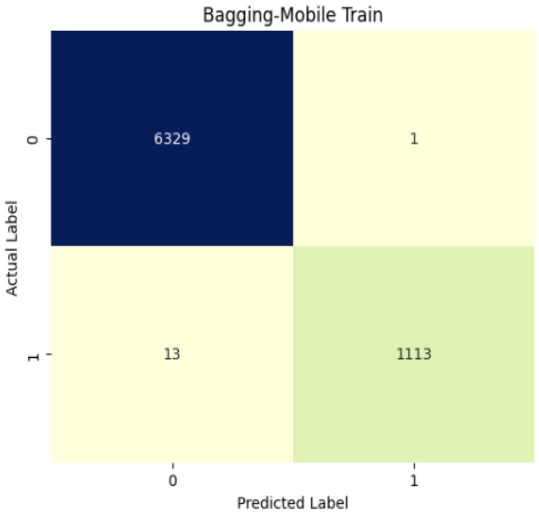

Random Forest - Mobile Train

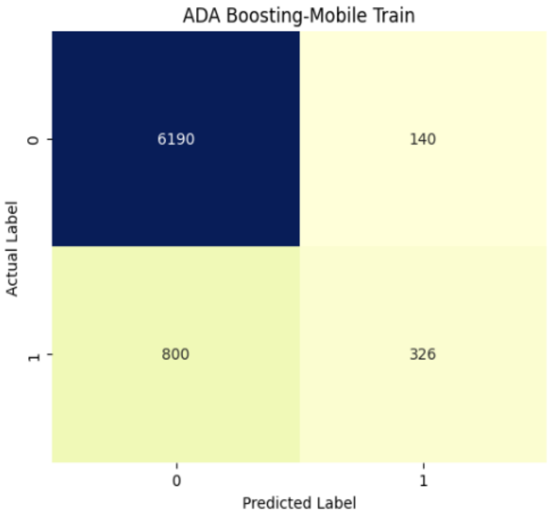

Random Forest-HyperParameter Tuning-Mobile Train

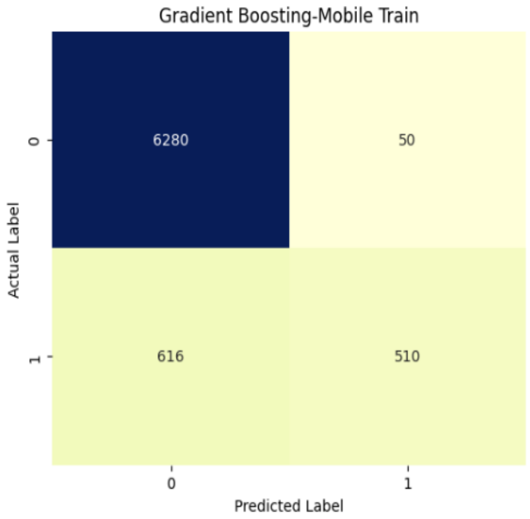

KNN-Mobile Train

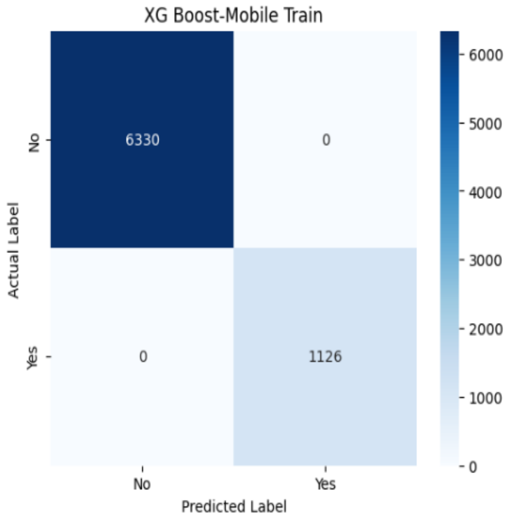

Naive Bayes-Mobile Train
Actual Label
0
1
6249
81
963
163
0
1
Predicted Label

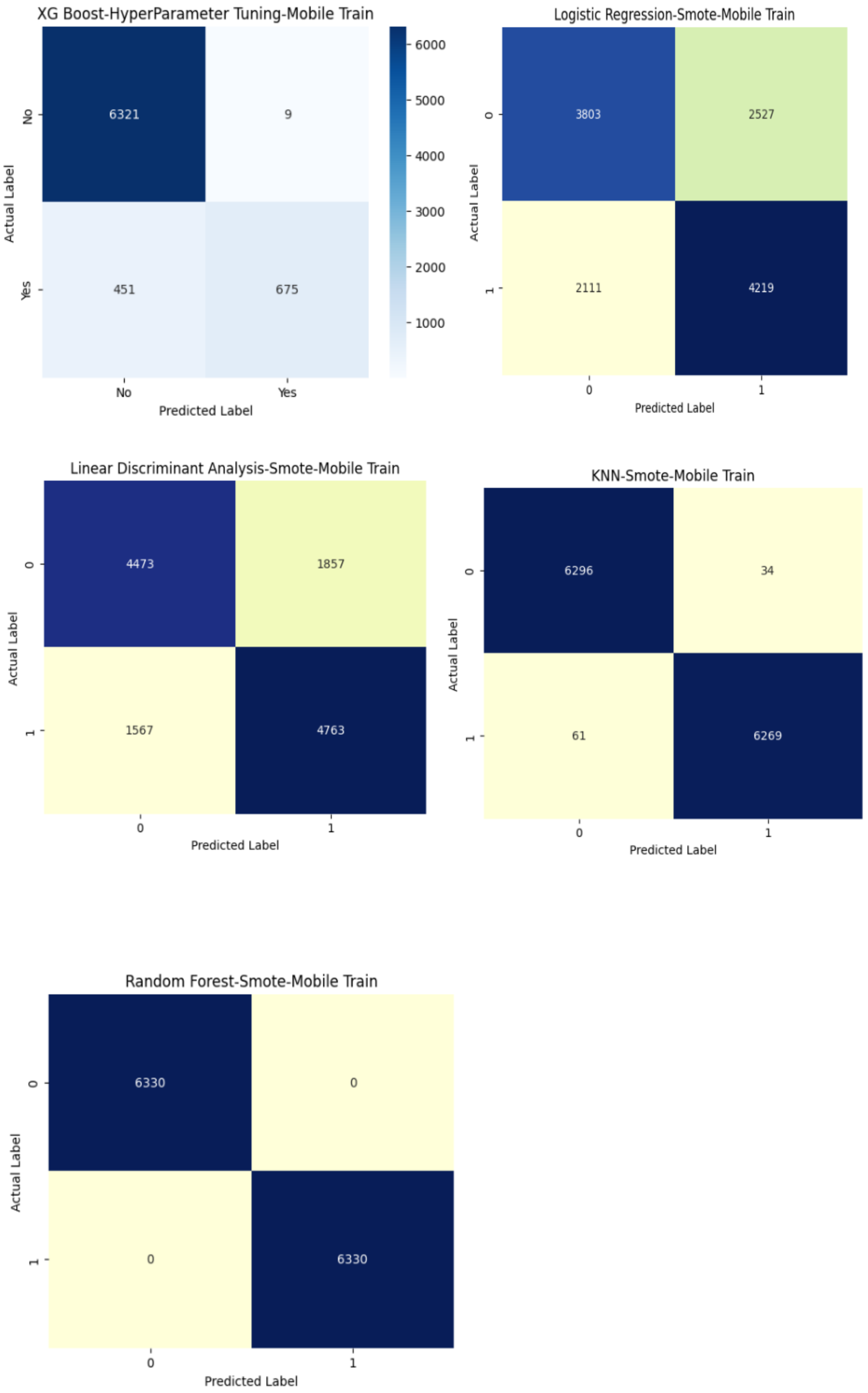

Bagging-HyperParameter Tuning-Mobile Train
Actual Label
0
1
6323
7
1025
101
0
1
Predicted Label


Bagging-Mobile Train
Actual Label
0
1
6329
1
13
1113
0
1
Predicted Label


ADA Boosting-Mobile Train
Actual Label
0
1
6190
140
800
326
0
1
Predicted Label


Gradient Boosting-Mobile Train
Actual Label
0
1
6280
50
616
510
0
1
Predicted Label


XG Boost-Mobile Train
Actual Label
No
Yes
6330
0
0
1126
No
Yes
Predicted Label
6000
5000
4000
3000
2000
1000
0



XG Boost-HyperParameter Tuning-Mobile Train

Logistic Regression-Smote-Mobile Train

Linear Discriminant Analysis-Smote-Mobile Train

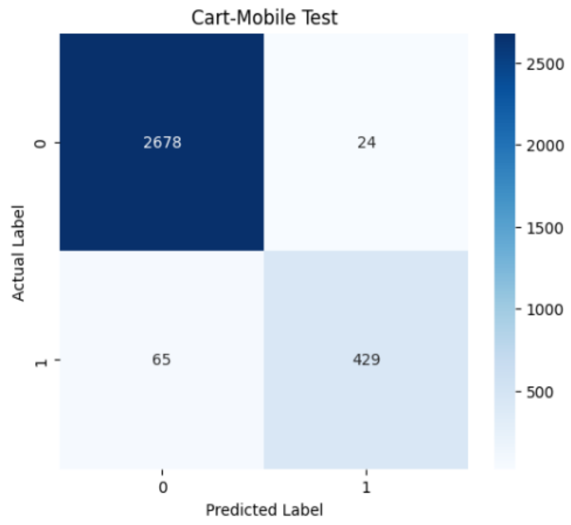
KNN-Smote-Mobile Train

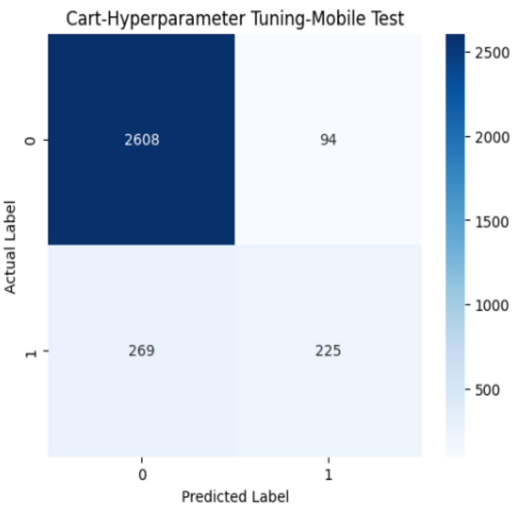
Random Forest-Smote-Mobile Train

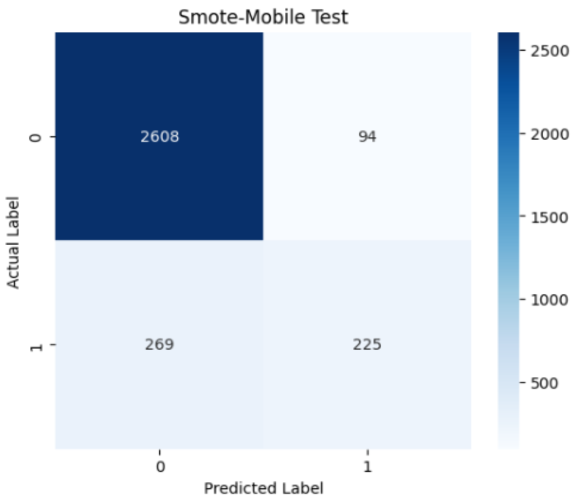
Fig 18. Mobile Train Data

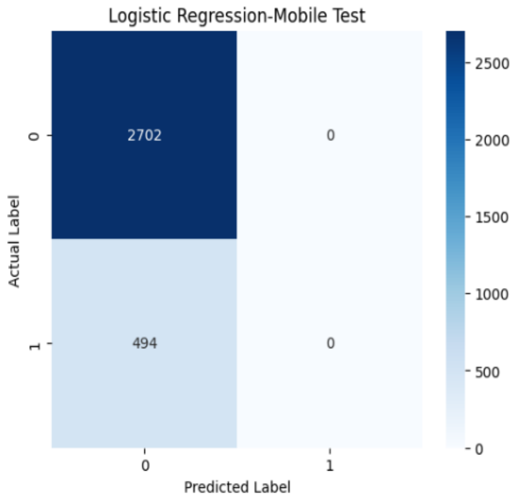
Random Forest and XGBoost have the fewest false positives (Type II errors) in the Positive Class (1).

Mobile Test Data

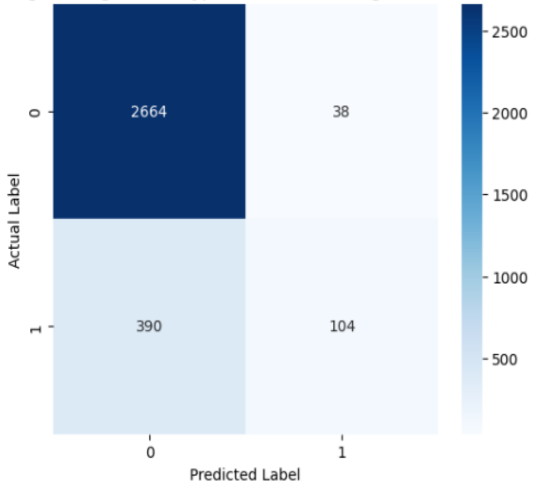




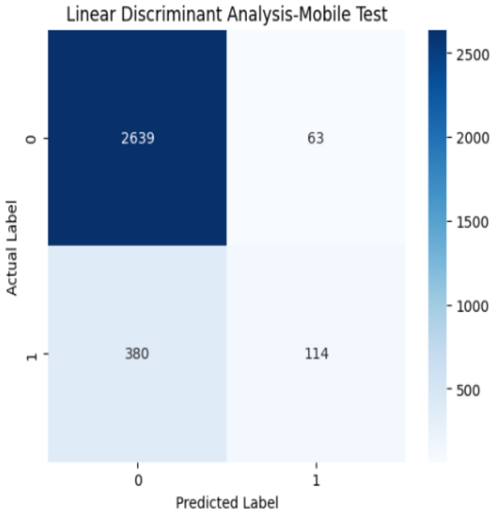




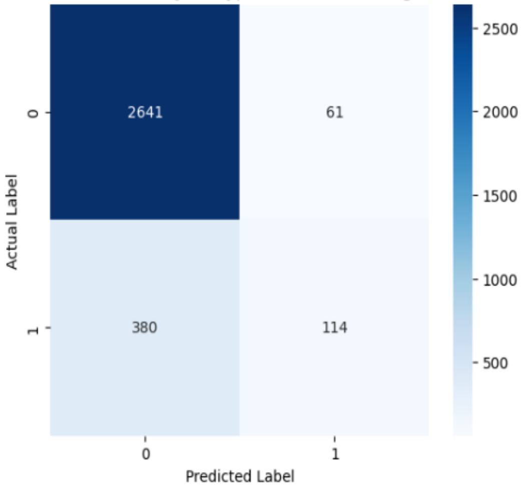
Logistic Regression-HyperParameter Tuning-Mobile Test



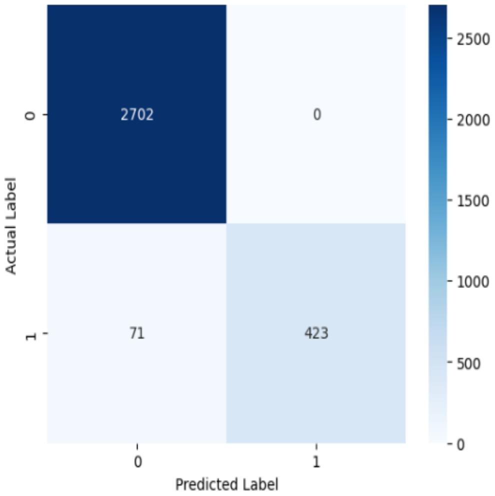
Linear Discriminant Analysis-Mobile Test



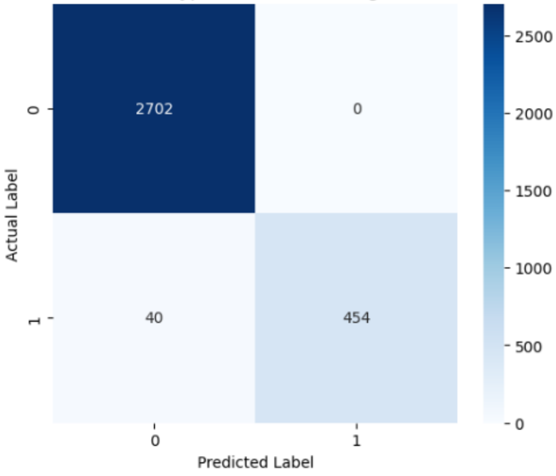
Linear Discriminant Analysis-HyperParameter Tuning-Mobile Test



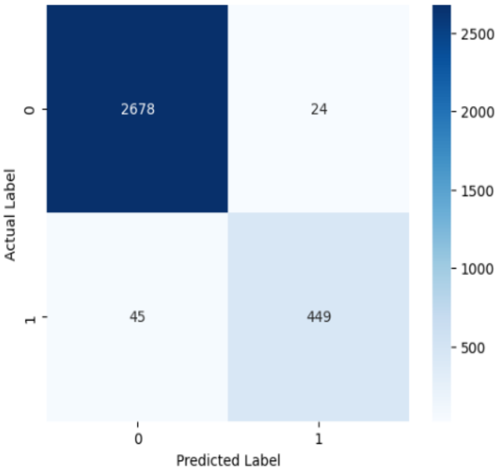
Random Forest - Mobile Test

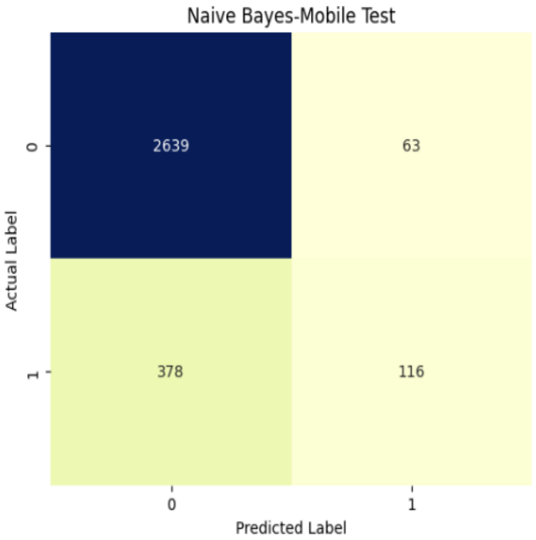


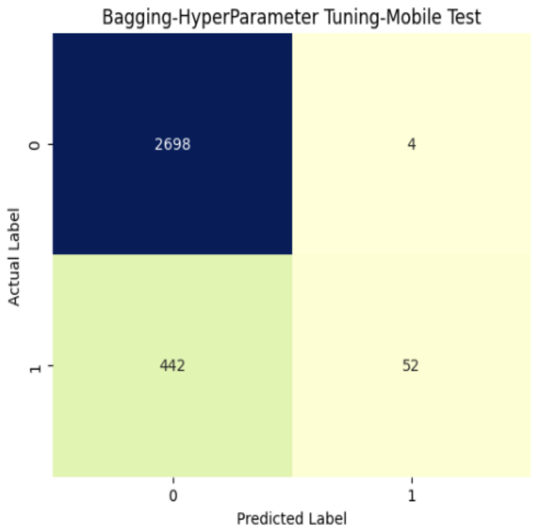
Random Forest-HyperParameter Tuning-Mobile Test

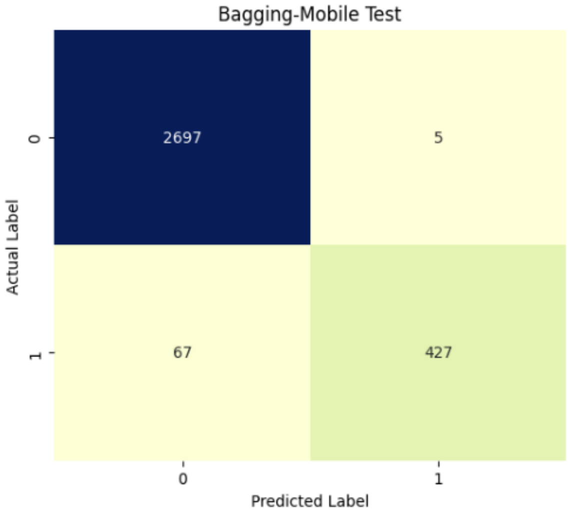


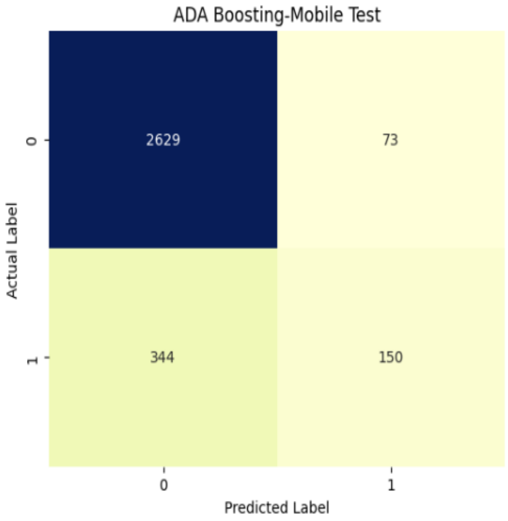
KNN-Mobile Test

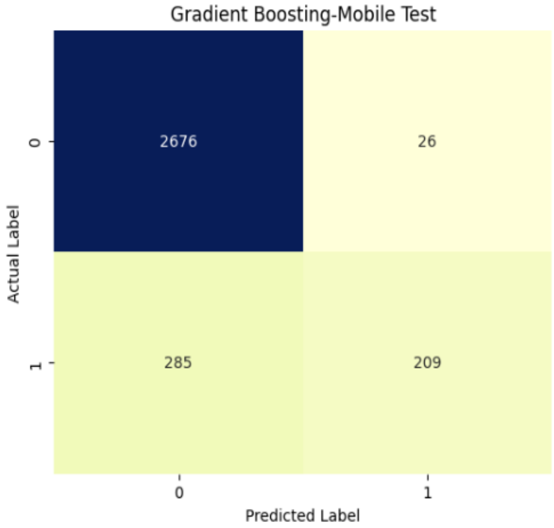


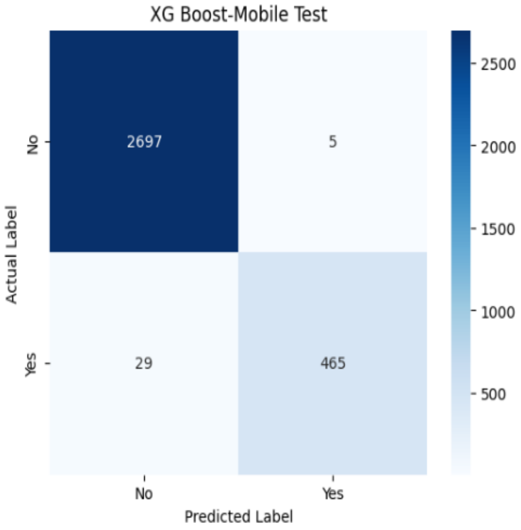

Naive Bayes-Mobile Test
Actual Label
0
1
2639
63
378
116
0
1
Predicted Label


Bagging-HyperParameter Tuning-Mobile Test
Actual Label
0
1
2698
4
442
52
0
1
Predicted Label


Bagging-Mobile Test
Actual Label
0
1
2697
5
67
427
0
1
Predicted Label


ADA Boosting-Mobile Test
Actual Label
0
1
2629
73
344
150
0
1
Predicted Label


Gradient Boosting-Mobile Test
Actual Label
0
1
2676
26
285
209
0
1
Predicted Label


XG Boost-Mobile Test
Actual Label
No
Yes
2697
5
29
465
No
Yes
Predicted Label
2500
2000
1500
1000
500

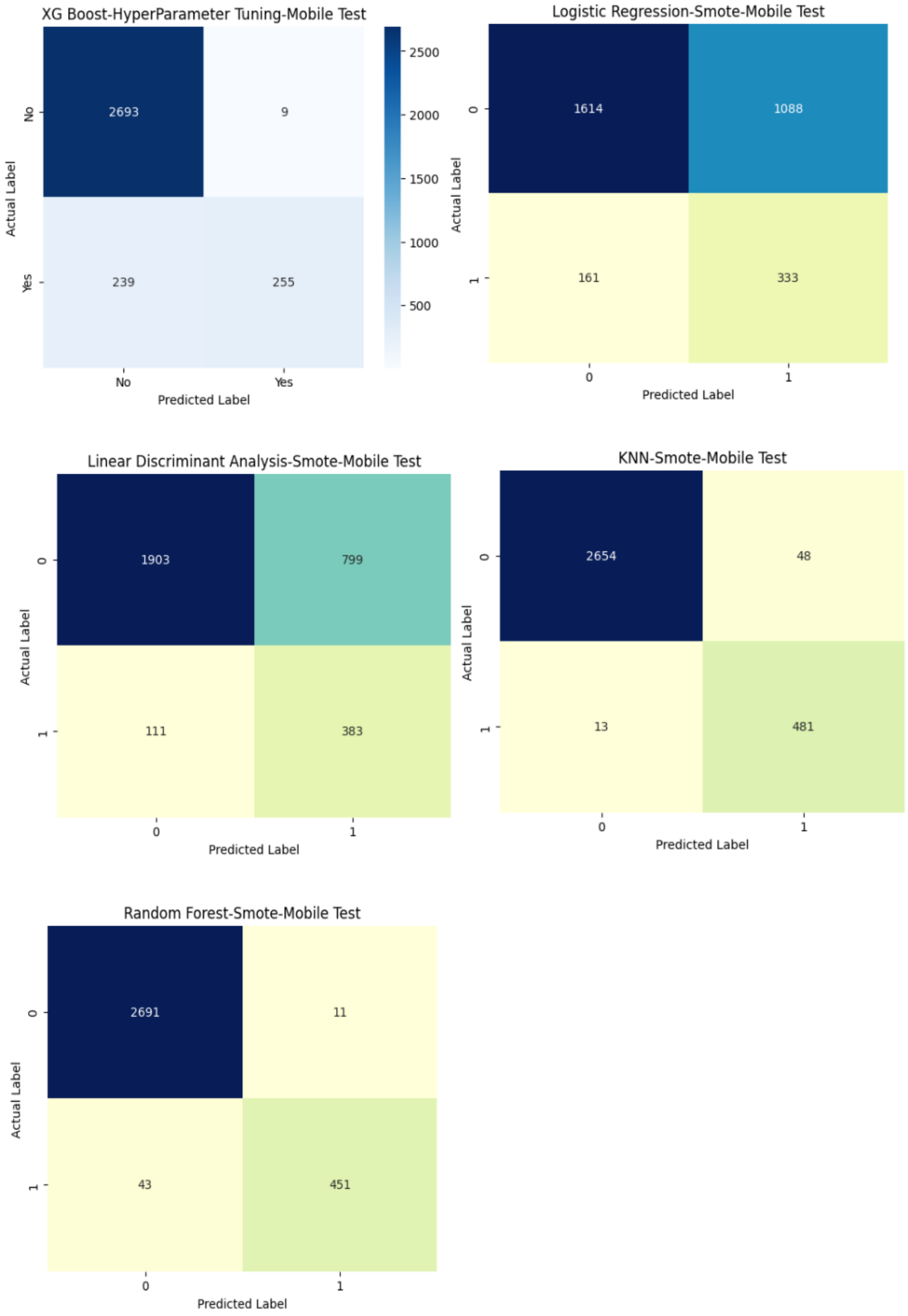


Fig 19. Mobile Test Data

Random Forest and XGBoost exhibit the lowest number of false positives (Type II errors) in the Positive Class (1).

Classification Report
Laptop - Training Data

Models	Accuracy	Precision	Recall	F1_Score
Cart	1.00	1.00	1.00	1.00
Cart using pruning	0.82	0.72	0.41	0.52
Smote	0.82	0.72	0.41	0.52
Logistic Regression	0.80	0.69	0.24	0.35
Linear Discriminant Analysis	0.83	0.72	0.43	0.54
Random Forest	1.00	1.00	1.00	1.00
KNN	0.99	0.99	0.98	0.99
Smote – Logistic Regression	0.68	0.66	0.72	0.69
Smote - LDA	0.75	0.75	0.74	0.74
Smote – KNN	0.99	0.99	0.98	0.99
Smote - RF	1.00	1.00	1.00	1.00
Logistic Regression-GridSearch CV	0.83	0.74	0.43	0.54
LDA- GridSearch CV	0.83	0.74	0.43	0.54
Random Forest-GridSearch CV	1.00	1.00	1.00	1.00
Naive Bayes	0.83	0.70	0.50	0.58
Bagging	1.00	1.00	0.99	1.00
Bagging-GridSearch CV	0.85	1.00	0.37	0.54
ADA Boosting	0.91	0.88	0.70	0.78
Gradient Boosting	0.98	1.00	0.93	0.96
XG Boost	1.00	1.00	1.00	1.00
XG Boost Hyperparameter Tune	1.00	1.00	0.98	0.99

Table 13. Laptop Train data report

Classification Report
Laptop - Testing Data

Models	Accuracy	Precision	Recall	F1_Score
Cart	0.93	0.88	0.87	0.88
Cart using pruning	0.81	0.75	0.48	0.59
Smote	0.81	0.75	0.48	0.59
Logistic Regression	0.74	0.63	0.25	0.36
Linear Discriminant Analysis	0.81	0.81	0.44	0.57

Random Forest	0.98	1.00	0.94	0.97
KNN	0.95	0.93	0.91	0.92
Smote – Logistic Regression	0.68	0.47	0.79	0.59
Smote - LDA	0.74	0.53	0.78	0.63
Smote – KNN	0.96	0.91	0.97	0.94
Smote - RF	0.98	1.00	0.94	0.97
Logistic Regression-GridSearch CV	0.82	0.81	0.46	0.59
LDA- GridSearch CV	0.81	0.82	0.44	0.58
Random Forest-GridSearch CV	0.98	1.00	0.94	0.97
Naive Bayes	0.81	0.65	0.73	0.69
Bagging	0.96	0.98	0.89	0.93
Bagging-GridSearch CV	0.85	1.00	0.47	0.64
ADA Boosting	0.86	0.79	0.69	0.74
Gradient Boosting	0.96	0.98	0.87	0.92
XG Boost	0.98	0.99	0.96	0.97
XG Boost Hyperparameter Tune	0.97	0.99	0.92	0.95

Table 14. Laptop Test data report

Classification Report
Mobile - Training Data

Models	**Accuracy**	**Precision**	**Recall**	**F1_Score**
Cart	1.00	1.00	1.00	1.00
Cart using pruning	0.90	0.74	0.49	0.59
Smote	0.90	0.74	0.49	0.59
Logistic Regression	0.85	0.50	0.00	0.00
Linear Discriminant Analysis	0.86	0.64	0.21	0.32
Random Forest	1.00	1.00	1.00	1.00
KNN	0.99	0.99	0.97	0.98
Smote – Logistic Regression	0.63	0.63	0.67	0.65
Smote - LDA	0.73	0.72	0.75	0.74
Smote – KNN	0.99	0.99	0.99	0.99
Smote - RF	1.00	1.00	1.00	1.00
Logistic Regression-GridSearch CV	0.86	0.70	0.19	0.29
LDA- GridSearch CV	0.86	0.64	0.22	0.32
Random Forest-GridSearch CV	1.00	1.00	1.00	1.00
Naive Bayes	0.86	0.67	0.14	0.24

Bagging	1.00	1.00	0.99	0.99
Bagging-GridSearch CV	0.86	0.94	0.09	0.16
ADA Boosting	0.87	0.70	0.29	0.41
Gradient Boosting	0.91	0.91	0.45	0.60
XG Boost	1.00	1.00	1.00	1.00
XG Boost Hyperparameter Tune	0.94	0.99	0.60	0.75

Table 15. Mobile Train data report

Classification Report
Mobile - Testing Data

Models	Accuracy	Precision	Recall	F1_Score
Cart	0.97	0.95	0.87	0.91
Cart using pruning	0.89	0.71	0.46	0.55
Smote	0.89	0.71	0.46	0.55
Logistic Regression	0.85	0.00	0.00	0.00
Linear Discriminant Analysis	0.86	0.64	0.23	0.34
Random Forest	0.98	1.00	0.86	0.92
KNN	0.98	0.95	0.91	0.93
Smote – Logistic Regression	0.61	0.23	0.67	0.35
Smote - LDA	0.72	0.32	0.78	0.46
Smote – KNN	0.98	0.91	0.97	0.94
Smote - RF	0.98	0.98	0.91	0.94
Logistic Regression-GridSearch CV	0.87	0.73	0.21	0.33
LDA- GridSearch CV	0.86	0.65	0.23	0.34
Random Forest-GridSearch CV	0.99	1.00	0.92	0.96
Naive Bayes	0.86	0.65	0.23	0.34
Bagging	0.98	0.99	0.86	0.92
Bagging-GridSearch CV	0.86	0.93	0.11	0.19
ADA Boosting	0.87	0.67	0.30	0.42
Gradient Boosting	0.90	0.89	0.42	0.57
XG Boost	0.99	0.99	0.94	0.96
XG Boost Hyperparameter Tune	0.92	0.97	0.52	0.67

Table 16. Mobile Test data report

- Linear Discriminant Analysis (LDA) had slightly better results but still suffered from low recall, missing many positive cases.
- Logistic Regression was the weakest performer across both datasets, showing low precision and recall, especially on test data making it unreliable for classification.
- **CART, Random Forest, Bagging, and XGBoost** achieved **100% training accuracy**, which signals **overfitting** and lack of generalization.
- After applying **SMOTE** to balance the classes and **GridSearchCV** for tuning, most models showed noticeable improvement on the test data.
- KNN delivered strong test results (precision and recall > 90%) after SMOTE, though it was slightly less consistent compared to ensemble models.
- **Boosting models** (Gradient Boosting and XGBoost) consistently performed well, with XGBoost showing top metrics across both train and test sets.
- Among all, **XGBoost with hyperparameter tuning** showed **the most stable and accurate performance** - high precision, high recall, and the lowest false positives making it the **best choice** for this classification task.

6. Final interpretation / recommendation - Very clear and crisp on what recommendations do you want to give to the management / client.

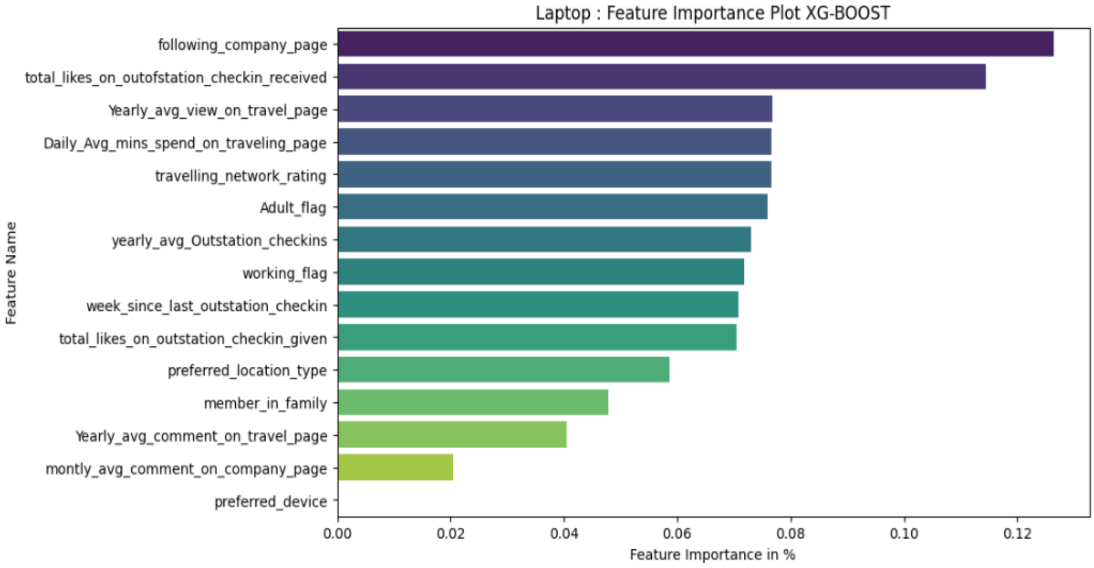


Fig 20. Laptop feature importance plot XG-Boost

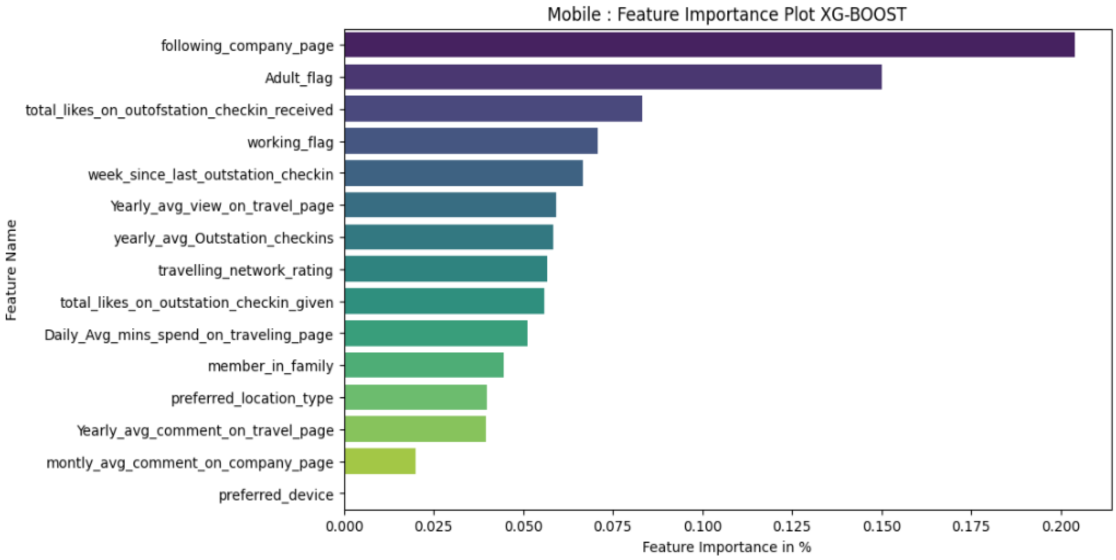


Fig 21. Mobile feature importance plot XG-Boost

- Based on the feature importance chart above, personalized tour plans can be recommended to users by leveraging the variables identified as most influential.
- The variable **following_company_page** ranks highest in importance, indicating that users who follow the company page are highly interested in travel and are more likely to purchase tour packages. This insight can help in crafting targeted offers for these users.
- The variable **total_likes_on_outstation_checkin_given** also reflects a strong interest in travel. It suggests that these users are active on social media and show a clear inclination toward tourism, making them good candidates for customized travel packages.
- Likewise, the **total_likes_on_outofstation_checkin_received** variable reinforces the users' enthusiasm for travel, further supporting their potential interest in tour offerings.
- Metrics such as **Yearly_avg_view_on_travel_page** and **Yearly_avg_comment_on_travel_page** provide deeper insights into user preferences. These can guide the development of specialized tour packages aligned with user interests.
- Lastly, the **week_since_last_outstation_checkin** variable is valuable in estimating when a user might be planning their next trip, helping in the timely delivery of relevant travel recommendations.

Business Recommendation

- **To optimize ad spending, the company should prioritize mobile devices** for digital campaigns, as most users browse via smartphones.
- **Customized travel packages** can be created for fun destinations like beaches, with **group discounts** for bookings of 3 or more people.
- **The dataset analysis has given the company a clear view of user behavior on social media.** By focusing on key features, the organization can boost revenue effectively.
- **Business or finance travelers can be offered complimentary services** bundled with their stay when they choose a company package.
- **Discounts for medical travelers** can help build trust and long-term loyalty.
- **Interactive games on the company's page** can keep users engaged, offering rewards like discounts or free stays for winners.
- **Encouraging users to comment or review** can earn them **redeemable points** for future discounts.
- **Loyal customers can receive referral coupons** for family and friends, expanding the customer base and increasing revenue.
- **Users with high engagement (likes/comments) should be prioritized**, as they are more likely to purchase.
- **Those incorrectly predicted as converted can be re-targeted**, if budget allows.
- **Laptop users who prefer hill stations and have high travel ratings show strong buying intent** and should be targeted accordingly.
- **Mobile users who are adults, working, and entertainment-focused** also show high conversion potential.
- **Travel combos can be introduced for frequent travelers** to encourage repeat bookings.
- **Testimonials from loyal customers** can strengthen brand image and attract new buyers.
- **Destination-specific ads** can be sent based on user interests.
- **Discounts can be offered to users who didn't convert earlier** to spark renewed interest.