PGPDSE FT Online January 2023

Capstone Project – Interim Report

Project Group Info:

BATCH DETAILS	PGPDSE-FT Online Jan-23
TEAM MEMBERS	• Vaidehi Lehekar • Rohan Chugh • Basava Prabhu • Vaishnavi
DOMAIN OF PROJECT	Web and Social Media Analytics
PROJECT TITLE	**Exploring Facebook User Data: Recommendations**
GROUP NUMBER	GROUP – 4
TEAM LEADER	Vaidehi Lehekar
MENTOR NAME	Mr. Sourab Reddy

Abstract:

The project "Exploring Facebook User Data: Recommendations" aims to analyze and gain insights from a dataset containing Facebook user data, with a specific focus on understanding user preferences and generating personalized recommendations. The dataset includes information about users' gender and the number of likes they have received on their posts.

The project also explores data cleaning procedures to handle missing values, duplicates, and any data quality issues before conducting the analysis. These data cleaning steps ensure the accuracy and reliability of the results.

In conclusion, "Exploring Facebook User Data: Recommendations" aims to provide valuable insights into user behaviour on the platform, highlighting potential gender-based differences in user engagement. The findings can be used to develop personalized recommendation algorithms that cater to users' unique preferences, enhancing user experience and driving increased engagement on the platform.

Objectives:

- Data Exploration: The primary objective is to explore and understand the Facebook user data. This involves examining the structure of the dataset, identifying the variables, and understanding the data types and formats.
- Data Cleaning and Preprocessing: Address missing values, duplicates, and data quality issues in the dataset. Data cleaning ensures the data's integrity and reliability throughout the analysis.
- Descriptive Statistics: Calculate descriptive statistics for the 'likes' variable, including measures of central tendency (mean, median, mode) and measures of variability (standard deviation, range). This will provide a baseline understanding of the user engagement on the platform.
- Gender-Based Analysis: Conduct a gender-based analysis to compare the number of likes received by male and female users. Determine if there are any significant differences in the mean likes between the two gender categories.
- Hypothesis Testing: Formulate and test hypotheses to determine if there is a statistically significant difference in user engagement (likes) between male and female users. Utilize appropriate statistical tests, such as t-tests or Mann-Whitney U tests, to support the conclusions.
- Correlation Analysis: Explore potential correlations between the number of likes and other relevant variables, such as age or content type. Identify any patterns or associations that may exist between likes and these variables.
- Data Visualization: Create visualizations, such as scatter plots and box plots, to visually represent the distribution of likes and gender-based engagement patterns. Visualizations aid in understanding trends and outliers in the data.
- Confidence Intervals: Calculate confidence intervals for the mean number of likes in each gender category to estimate the range within which the true population mean lies. This provides a measure of uncertainty around the sample mean.
- Outlier Detection: Identify and analyze any outliers in the 'likes' variable to understand

their impact on the analysis and recommendations. Decide whether outliers need to be treated or removed for more accurate results.

- Recommendations: Based on the analysis and insights gained, propose strategies for personalized recommendations to improve user engagement on the Facebook platform. Explore potential factors that could influence personalized content recommendations.
- Limitations and Future Scope: Discuss the limitations of the current analysis and recommend future research directions. Address any data limitations and propose ways to enhance the analysis in future iterations.
- Interpretation and Conclusion: Summarize the key findings and their implications for Facebook's user engagement and recommendation algorithms. Provide a clear and actionable conclusion based on the results.

Overall, the detailed objectives of the project aim to explore user behaviour on the Facebook platform, compare engagement between male and female users, identify influential factors, and suggest strategies for personalized content recommendations to enhance the user experience and increase platform engagement.

Industry Review:

The rapid growth of social media platforms has transformed the way people connect, communicate, and consume information. With billions of users worldwide, platforms like Facebook, Twitter, Instagram, and LinkedIn generate vast amounts of data daily. This wealth of data presents significant opportunities for businesses to leverage data analytics and artificial intelligence (AI) technologies to gain valuable insights, enhance user experiences, and drive business growth.

Industry Trends:

- Data-Driven Decision Making: Social media platforms are embracing data-driven decision making, using advanced analytics to understand user behaviour, preferences, and engagement patterns. This allows platforms to tailor content, ads, and recommendations to individual users, resulting in increased user satisfaction and longer engagement.
- Personalization and Recommendations: Data analytics and AI play a pivotal role in delivering personalized content and recommendations to users. By analyzing user interactions, demographics, and interests, social media platforms can suggest relevant content, products, and connections, enhancing user engagement and retention.
- Sentiment Analysis and Social Listening: Social media platforms utilize sentiment analysis to gauge user sentiments towards brands, products, or events. Social listening tools track conversations and comments to understand user feedback, enabling brands to respond promptly to customer concerns and improve brand perception.
- Targeted Advertising: Data analytics enables precise targeting for advertising on

social media platforms. Advertisers can leverage user data to target specific demographics, interests, and behaviour's, leading to higher ad relevance and conversion rates.

- Influencer Marketing: Data analytics plays a crucial role in influencer marketing, identifying influential individuals and measuring their impact on brand promotion. Platforms can analyze engagement metrics to assess the success of influencer campaigns.
- Fraud Detection and User Safety: Social media platforms employ AI algorithms to detect and prevent fraudulent activities, such as fake accounts, spam, and malicious content. These efforts ensure user safety and maintain platform credibility.

Challenges:

- Privacy Concerns: As social media platforms collect vast amounts of user data, ensuring data privacy and complying with data protection regulations pose significant challenges. Platforms must strike a balance between data collection for personalized experiences and user privacy.
- Algorithmic Bias: AI algorithms used for content recommendations and advertising targeting may exhibit bias if not properly calibrated. Platforms must address bias issues to ensure fair and inclusive user experiences.
- Misinformation and Fake News: Social media platforms face the challenge of combating misinformation and fake news, which can spread rapidly on their networks. AI-powered content moderation tools are being developed to identify and flag false information.
- Ethical AI Use: The responsible use of AI in content moderation, recommendations, and user targeting requires careful consideration of ethical implications. Transparent and accountable AI practices are essential to maintain user trust.

Current practices in Industry:

As of the current practices in the industry, data analytics and artificial intelligence (AI) are being extensively employed by social media platforms to enhance user experiences, drive engagement, and optimize business operations. Here are some prominent current practices:

- Personalization: Social media platforms are heavily focused on delivering personalized content and recommendations to users. Advanced algorithms analyze user interactions, browsing history, and preferences to curate customized feeds and suggest relevant content, connections, and advertisements.
- Real-time Analytics: Real-time data analytics is a crucial aspect of the industry. Platforms use streaming data to track user activities, monitor trends, and respond promptly to user interactions. This allows for dynamic content delivery and quick adaptations to user needs.
- Natural Language Processing (NLP): NLP is widely used to understand and analyze user-generated content. Sentiment analysis and language processing techniques help

gauge user sentiments, identify trends, and improve customer support through chatbots and automated responses.

- Image and Video Recognition: Social media platforms employ image and video recognition technologies to automatically tag and categorize media content. AI algorithms can detect objects, faces, and activities within images and videos, making content search and discovery more efficient.
- Influencer Marketing and Social Listening: Data analytics helps identify influential users and track their impact on brand promotion. Social listening tools monitor user conversations, comments, and mentions to understand sentiment and customer feedback.
- Ad Targeting and Performance Analysis: Social media platforms leverage data analytics to target advertisements accurately. Advertisers can define specific audience segments based on demographics, interests, and behaviour's. Performance analysis provides insights into ad effectiveness and return on investment.
- AI-Driven Content Moderation: AI-powered content moderation tools are employed to detect and remove harmful or inappropriate content. These tools assist in maintaining platform safety and compliance with community guidelines.
- User Engagement Analysis: Data analytics is used to measure user engagement metrics, such as likes, shares, comments, and time spent on posts. Platforms analyze this data to understand user preferences and optimize content strategies.
- Fraud Detection and Security: AI algorithms are employed to detect and prevent fraudulent activities, including fake accounts, phishing attempts, and spam. These measures enhance user security and trust in the platform.
- Augmented Reality (AR) and Virtual Reality (VR) Integration: Some social media platforms are exploring AR and VR technologies to provide interactive and immersive experiences for users. These technologies are used in filters, lenses, and 3D content creation.

Conclusion:

The integration of data analytics and AI technologies has revolutionized the social media industry. These tools empower platforms to provide personalized experiences, targeted advertising, and valuable insights. However, the industry must address challenges related to data privacy, algorithmic bias, and misinformation to build a safer and more trustworthy online environment. As technology continues to evolve, the effective use of data analytics and AI will be crucial for social media platforms to remain competitive and meet the ever-changing needs of their users.

Problem Statement:

The primary objective of this analysis is to optimize Facebook's business growth by identifying the most valuable users within the dataset. By leveraging the available data, the aim is to uncover patterns, preferences, and behaviour's that can contribute to increased user engagement, retention, and revenue generation.

Specifically, the analysis aims to achieve the following business objectives:

1. User Segmentation: By exploring the dataset, the goal is to identify distinct user segments based on various demographic, psychographic, and behavioural attributes. This segmentation will enable Facebook to understand the diverse user base and tailor their strategies to cater to the specific needs and preferences of each segment.

2. User Engagement Analysis: The analysis seeks to uncover the factors that contribute to higher user engagement on the platform. By examining user interactions, content consumption patterns, and engagement metrics, Facebook can gain insights into the types of content and features that resonate most with users. This information can then be utilized to optimize the user experience and boost overall engagement levels.

3. Recommendation Optimization: An important aspect of the analysis is to improve the accuracy and effectiveness of Facebook's recommendation algorithms. By examining the dataset, patterns can be identified that indicate the types of content or products that users are more likely to be interested in. This information can be leveraged to enhance personalized recommendations, driving user satisfaction and increasing the likelihood of conversion.

4. User Retention Strategies: The analysis aims to uncover insights related to user churn and retention. By understanding the factors that contribute to user attrition, Facebook can develop targeted retention strategies to mitigate churn. These strategies may include personalized incentives, improved user support, or tailored content recommendations to encourage long-term engagement and loyalty.

By accomplishing these business objectives through exploratory data analysis, Facebook can gain a deeper understanding of its user base and make informed decisions to optimize its business operations. The insights derived from the dataset will serve as a foundation for implementing effective marketing, engagement, and retention strategies, ultimately driving growth and success for the platform.

Dataset and Domain:

Data Dictionary:

99003 entries, 0 to 99002 and data columns (total 15 columns)

The attribute/feature/column names are given below:

Features	Description	Data-Type
userid	Unique User-ID	Int64
age	Age of the user	int64
dob_day	Day of birthdate	int64
dob_year	Year of birthdate	int64
dob_month	Month of birthdate	int64
gender	Gender of the person (Male/Female)	Object
tenure	Tenure of person	float64
friend_count	Friend count	int64
friendships_initiated	Friendships Initiated (Friend requests sent)	int64
likes	Amount of total likes	int64
likes_received	Number of likes received	int64
mobile_likes	Number of mobile likes	int64
mobile_likes_received	Number of mobile likes received	int64
www_likes	Number of www_likes	int64
www_likes_received	Number of www_likes received	int64

1. Variable categorization:

There are 14 numerical columns and 01 categorical column

Numerical Columns:

[‘UserID’, ‘Age’, ‘Dob_day’,’Dob_month’, ’Dob_year’, ‘tenure’, ‘friend_count’, ‘friendships_initiated’, ‘likes’, ’likes_received’, ‘mobile_likes’, ‘mobile_likes_received’, ‘www_likes’, ‘ www_likes_received’]

Categorical Columns:

['Gender']

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 99003 entries, 0 to 99002
Data columns (total 15 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   userid                 99003 non-null  int64
 1   age                    99003 non-null  int64
 2   dob_day                99003 non-null  int64
 3   dob_year               99003 non-null  int64
 4   dob_month              99003 non-null  int64
 5   gender                 98828 non-null  object
 6   tenure                 99001 non-null  float64
 7   friend_count           99003 non-null  int64
 8   friendships_initiated  99003 non-null  int64
 9   likes                  99003 non-null  int64
 10  likes_received         99003 non-null  int64
 11  mobile_likes           99003 non-null  int64
 12  mobile_likes_received  99003 non-null  int64
 13  www_likes              99003 non-null  int64
 14  www_likes_received     99003 non-null  int64
dtypes: float64(1), int64(13), object(1)
memory usage: 11.3+ MB
```

2. Pre-Processing Data Analysis (count of missing/ null values, redundant columns, etc.)

The dataset has 177 missing values.

```
None
userid                      0
age                         0
dob_day                     0
dob_year                    0
dob_month                   0
gender                    175
tenure                      2
friend_count                0
friendships_initiated       0
likes                       0
likes_received              0
mobile_likes                0
mobile_likes_received       0
www_likes                   0
www_likes_received          0
dtype: int64
```

Filling missing values :-
Filled all the missing values using .fillna() function

```
    userid  age  dob_day  dob_year  dob_month  gender  tenure  friend_count  \
0  2094382   14       19      1999         11    male   266.0             0
1  1192601   14        2      1999         11  female     6.0             0
2  2083884   14       16      1999         11    male    13.0             0
3  1203168   14       25      1999         12  female    93.0             0
4  1733186   14        4      1999         12    male    82.0             0

   friendships_initiated  likes  likes_received  mobile_likes  \
0                      0      0               0             0
1                      0      0               0             0
2                      0      0               0             0
3                      0      0               0             0
4                      0      0               0             0

   mobile_likes_received  www_likes  www_likes_received
0                      0          0                   0
1                      0          0                   0
2                      0          0                   0
3                      0          0                   0
4                      0          0                   0
```

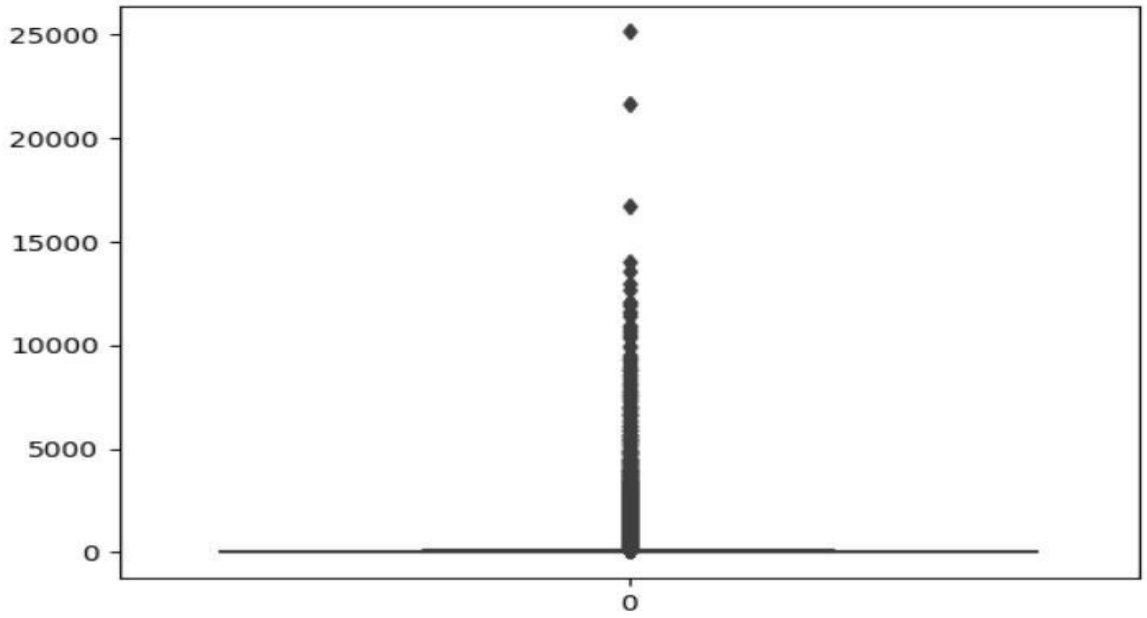
3. VISUALISATION PLOTS

UNIVARIATE ANALYSIS:

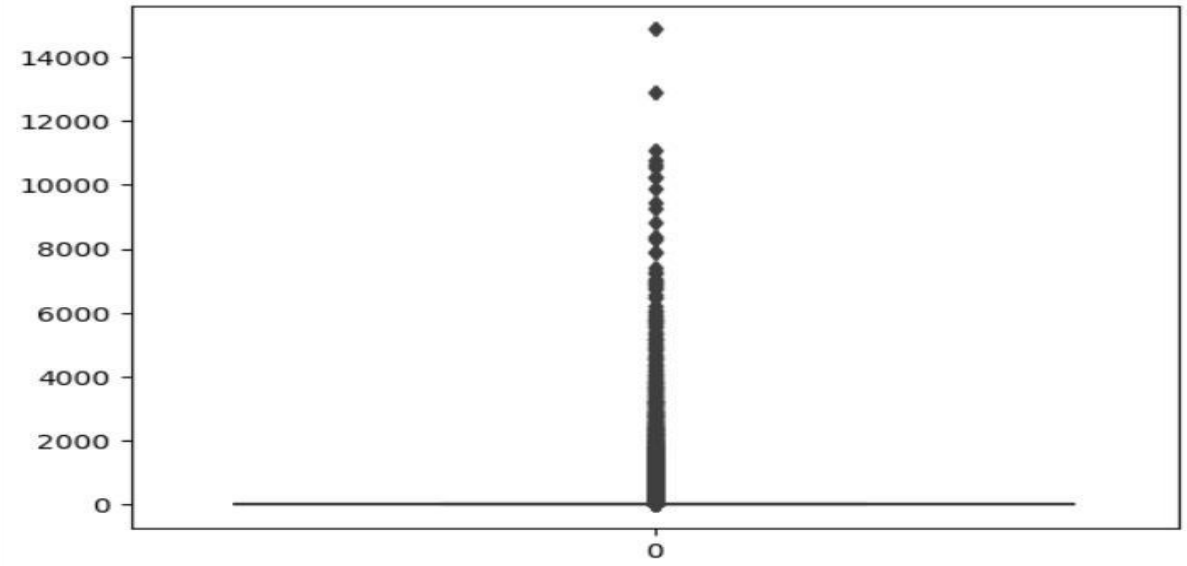
Analysis of Numerical Variables:
From the below plots, each of the numerical variables are analysed individually.

Boxplot :

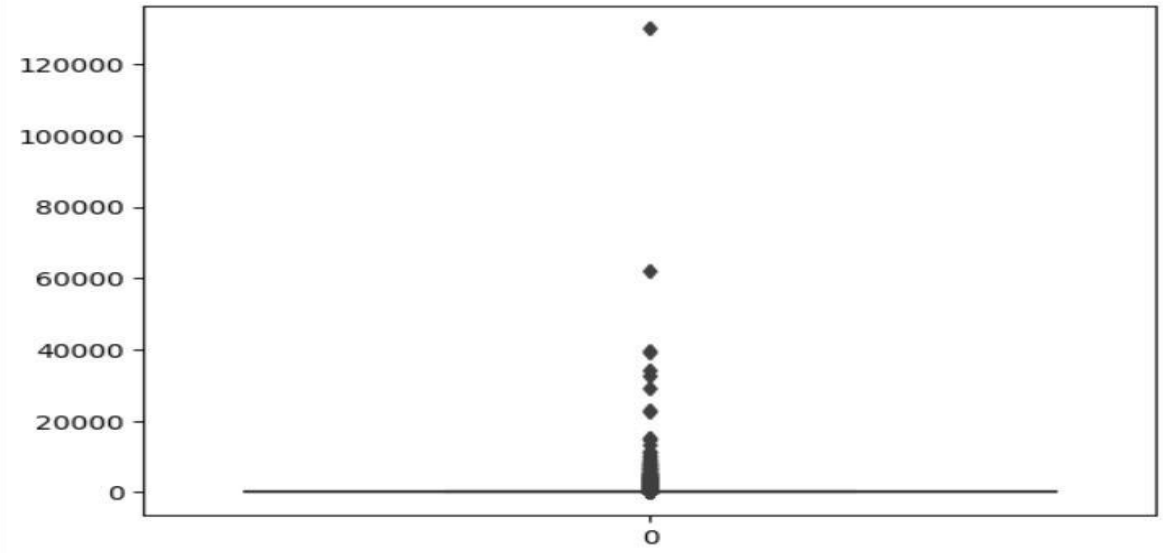
1. mobile_likes



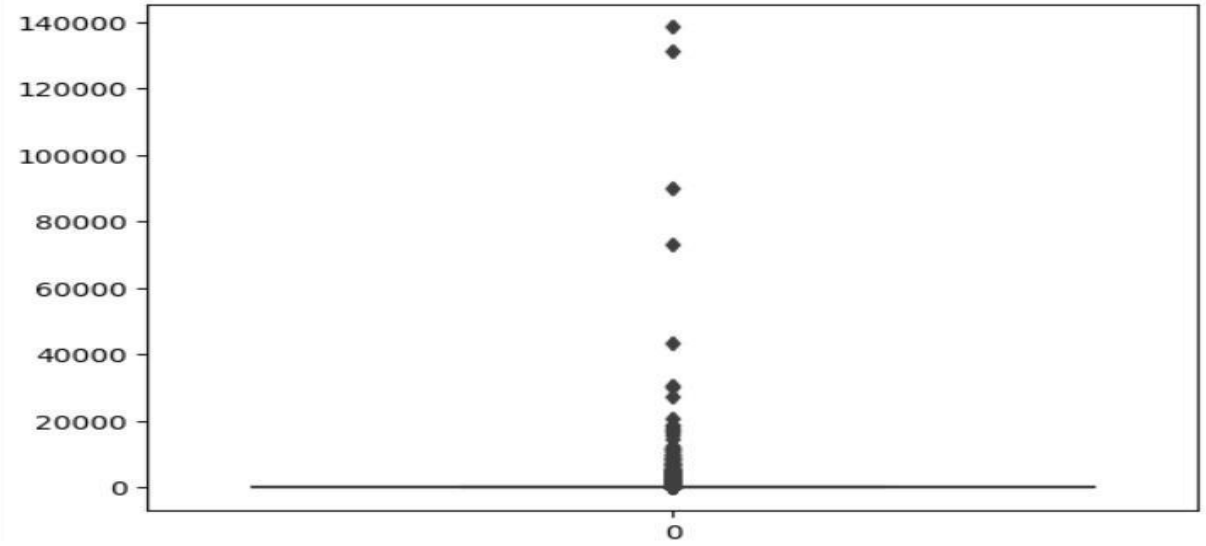
2. www_likes



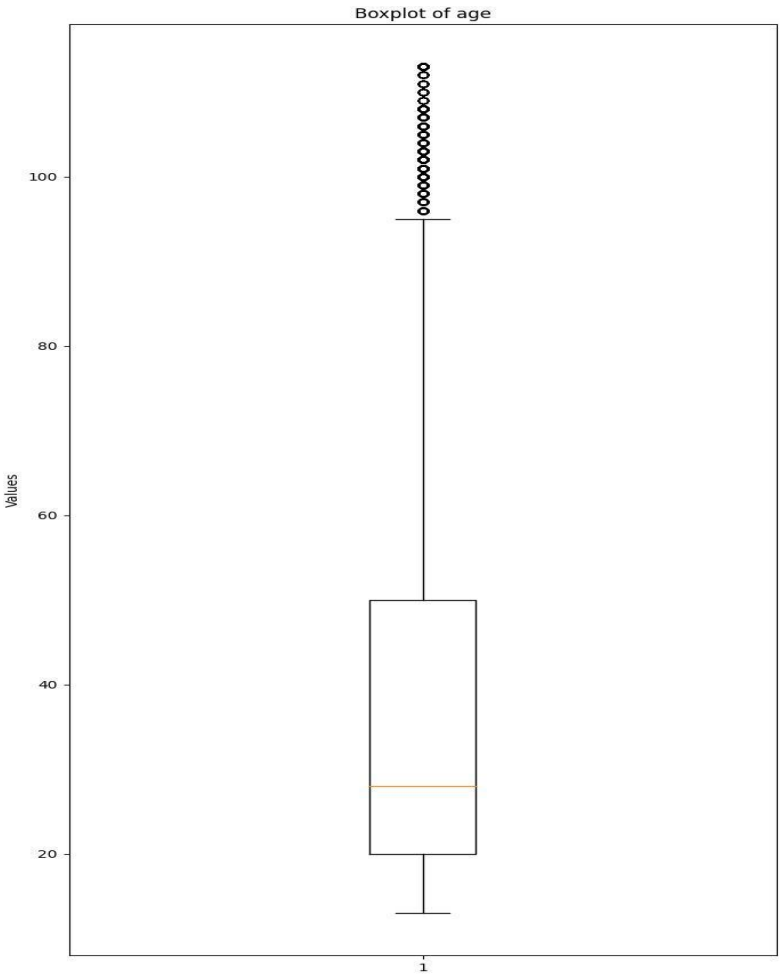
3. www_likes_received



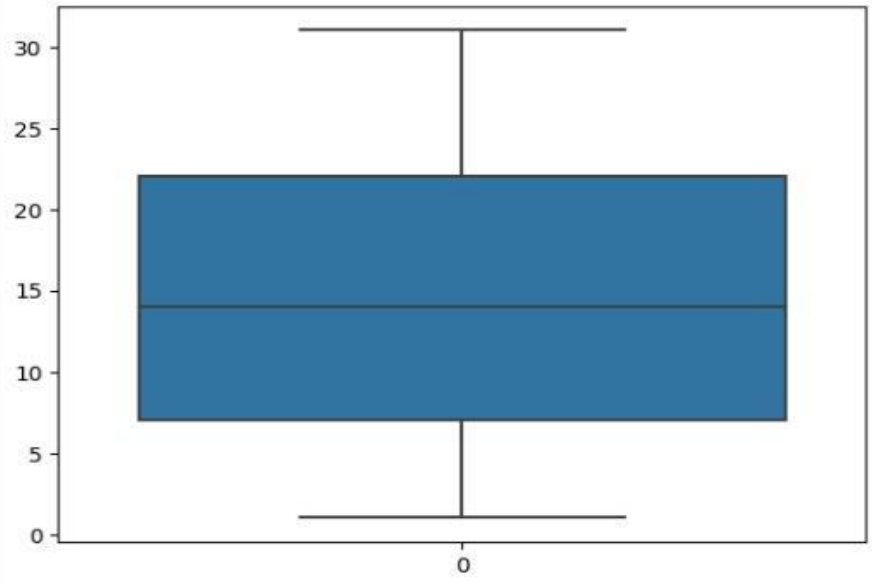
4. mobile_likes_received



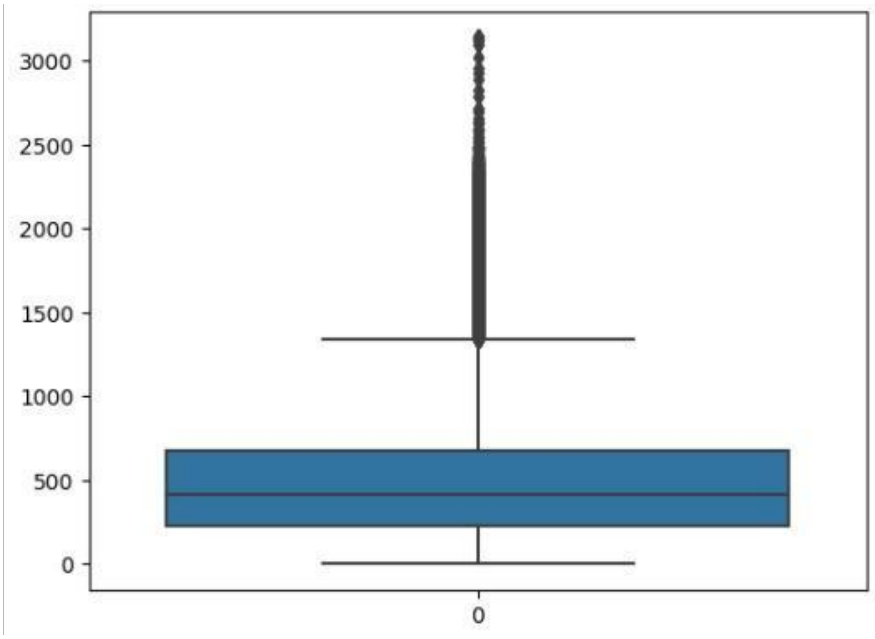
5. age



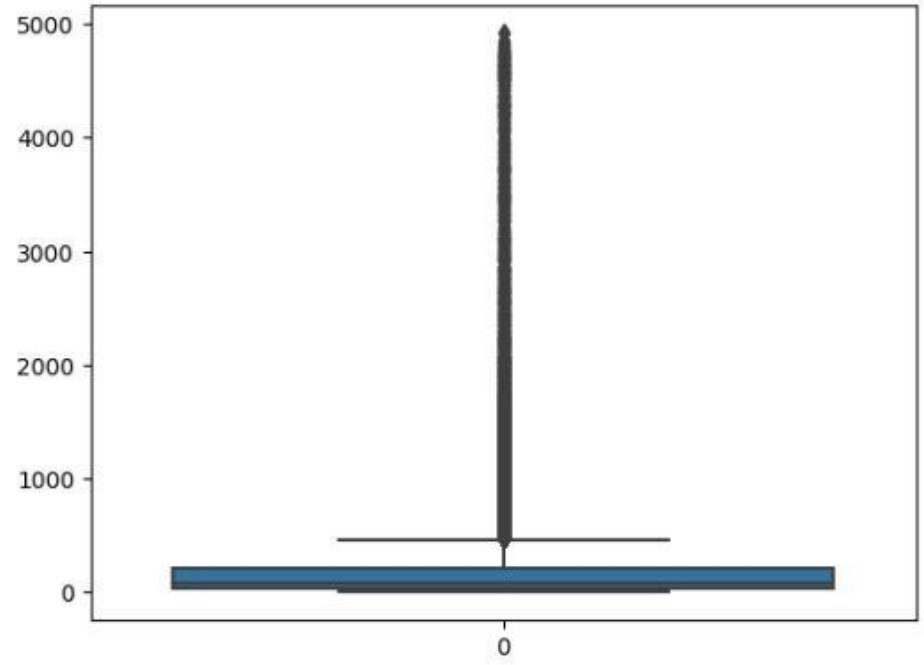
6. dob_day



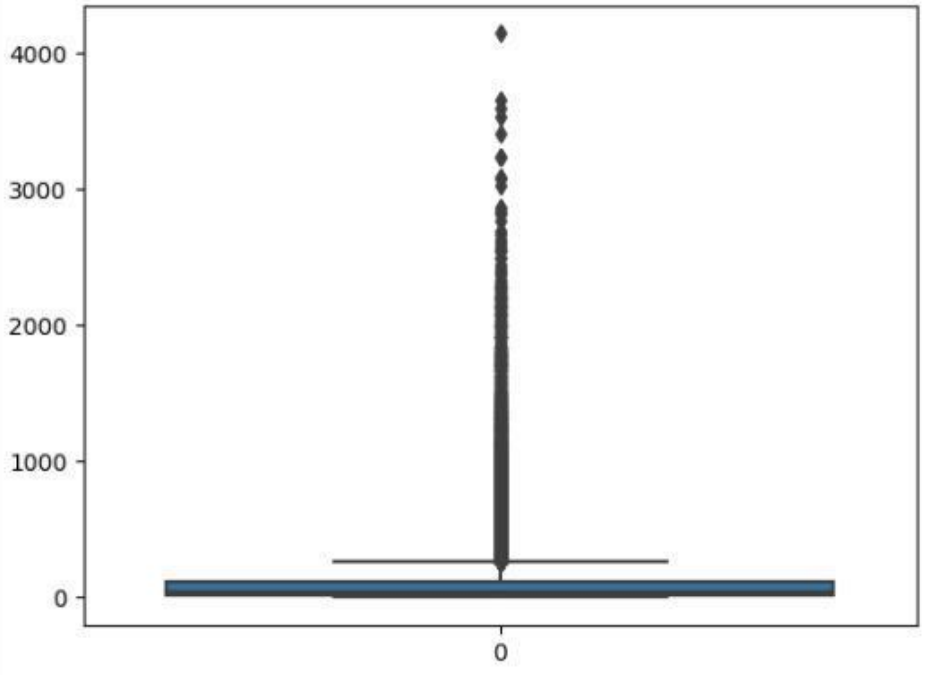
7. tenure



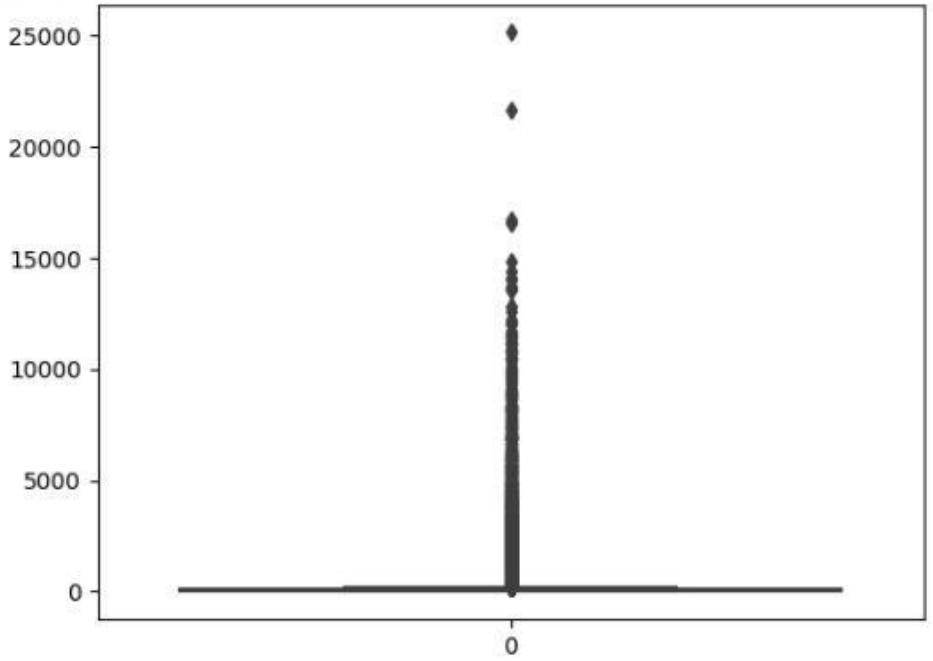
8. friend_count



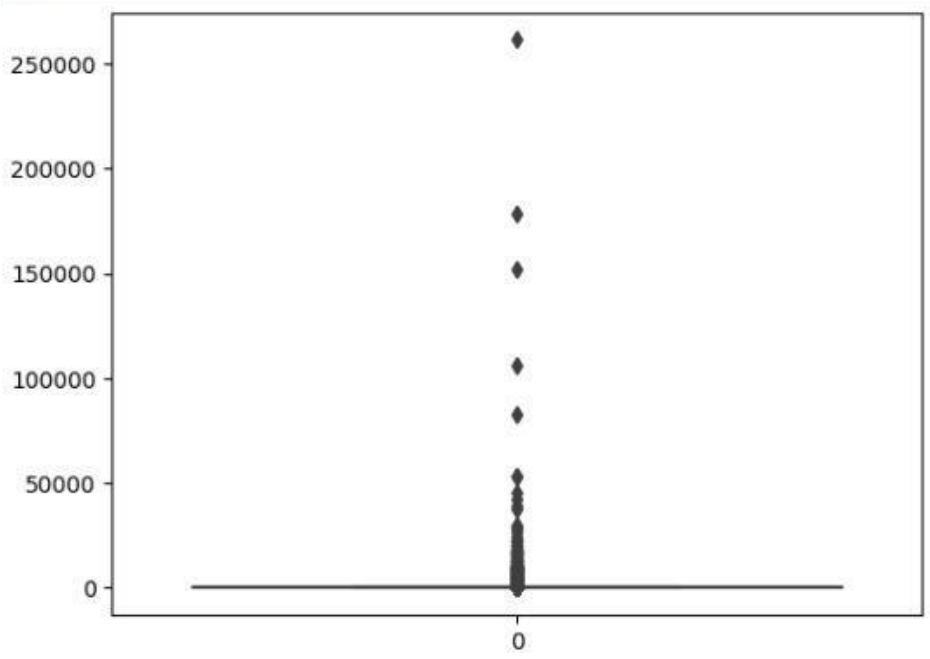
9. friendships_initiated



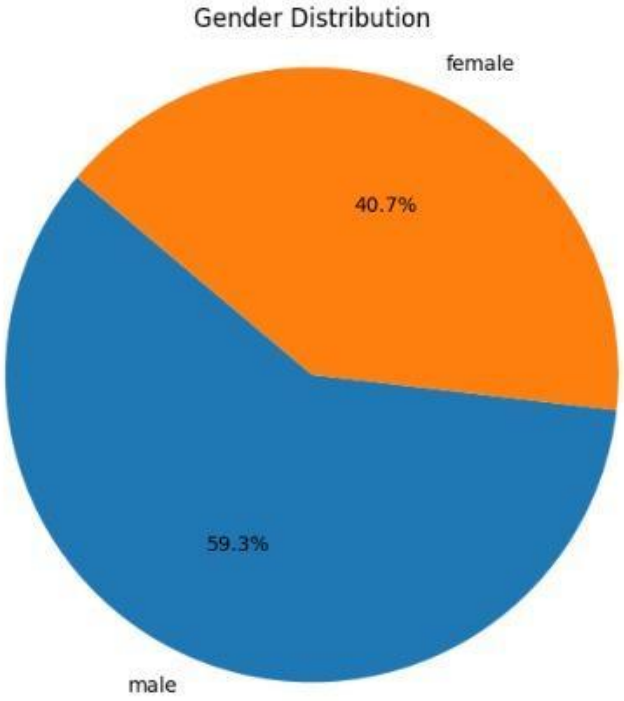
10. likes



11. likes_recieved



For Categorical values :

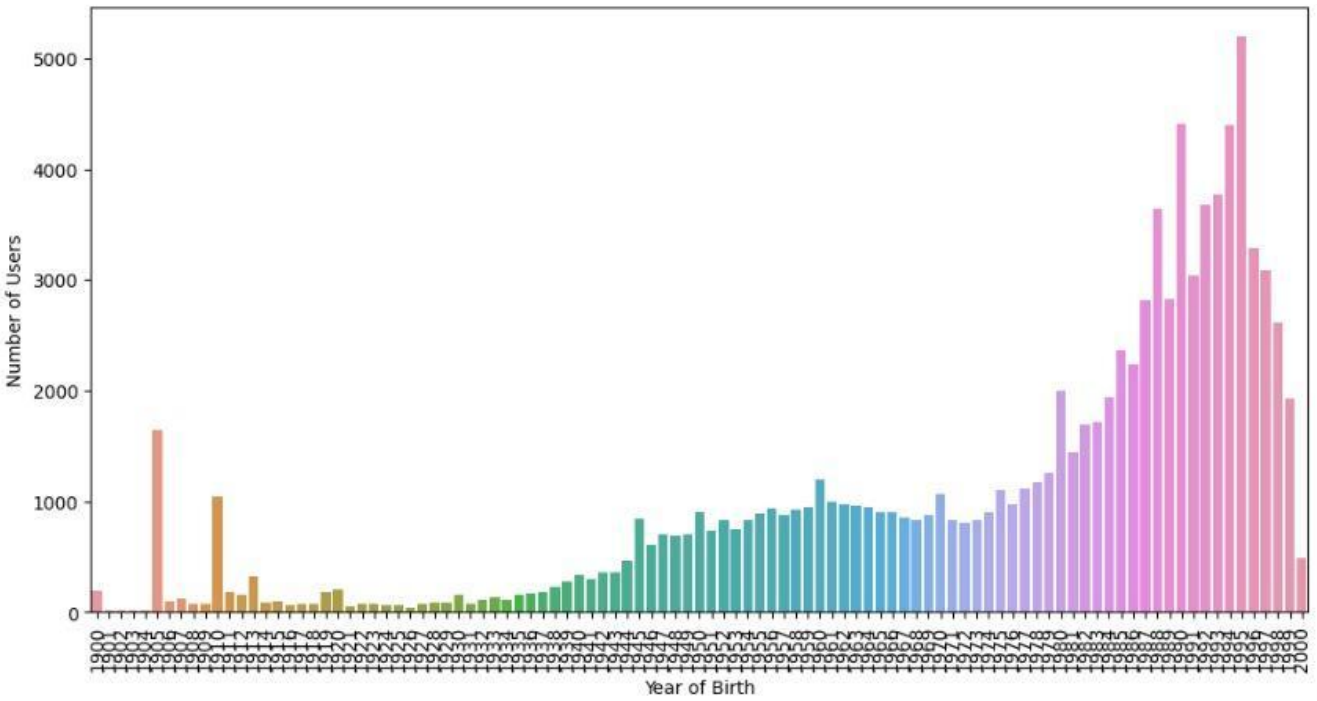


From the above pie chart for categorical variable “Gender” , we can observe that 59.3% of people are male and only 40.7% of people are female.

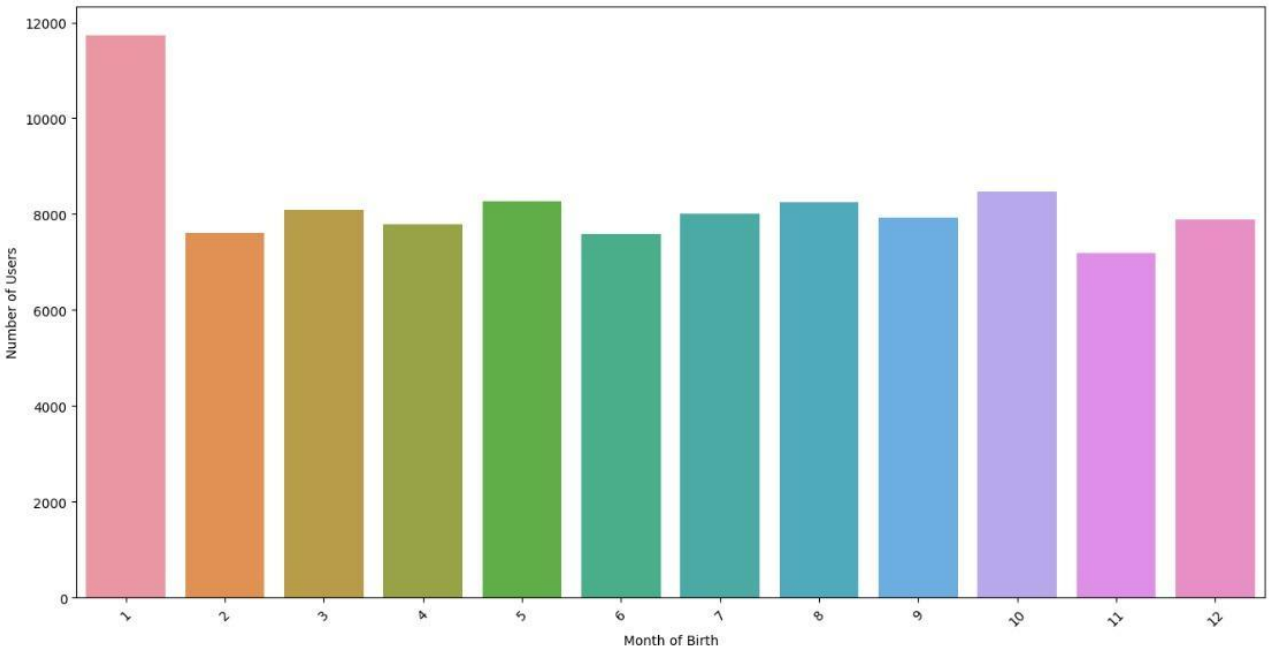
The value counts is as :

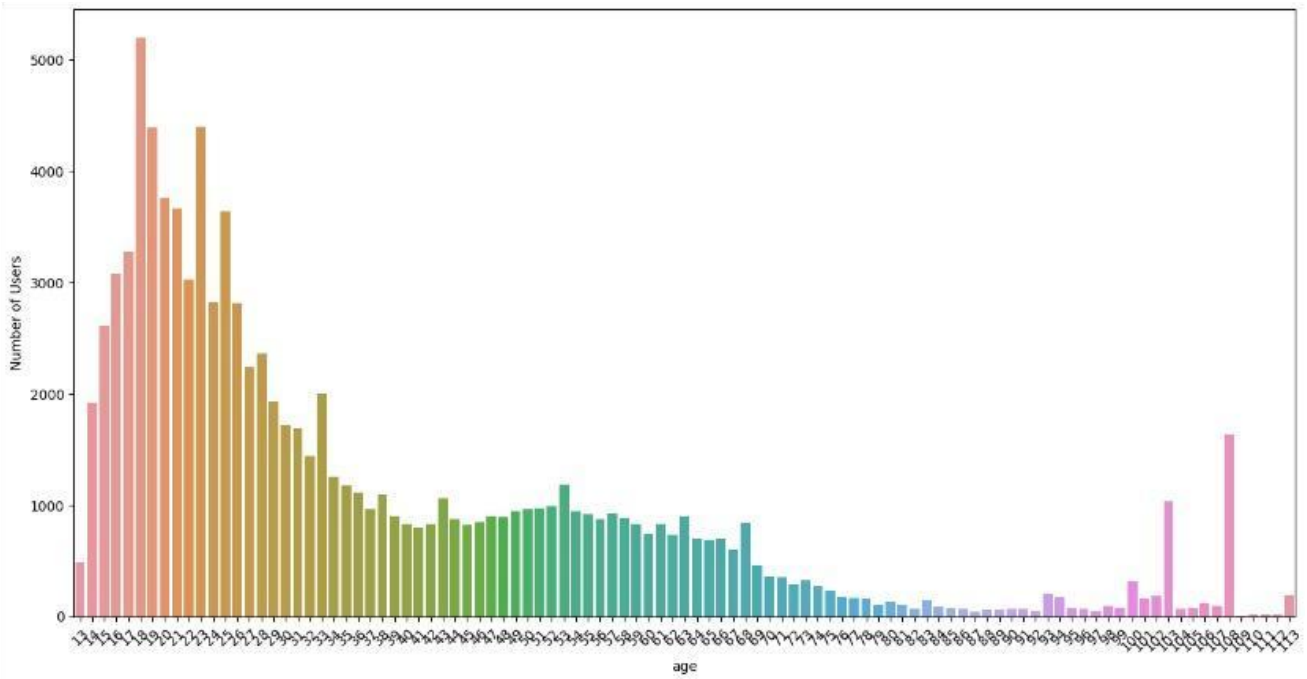
Male – 58574

Female – 40254



People born in 1995 have the highest number of users.
Lowest number of users are from the year of birth 1901 to 1904.

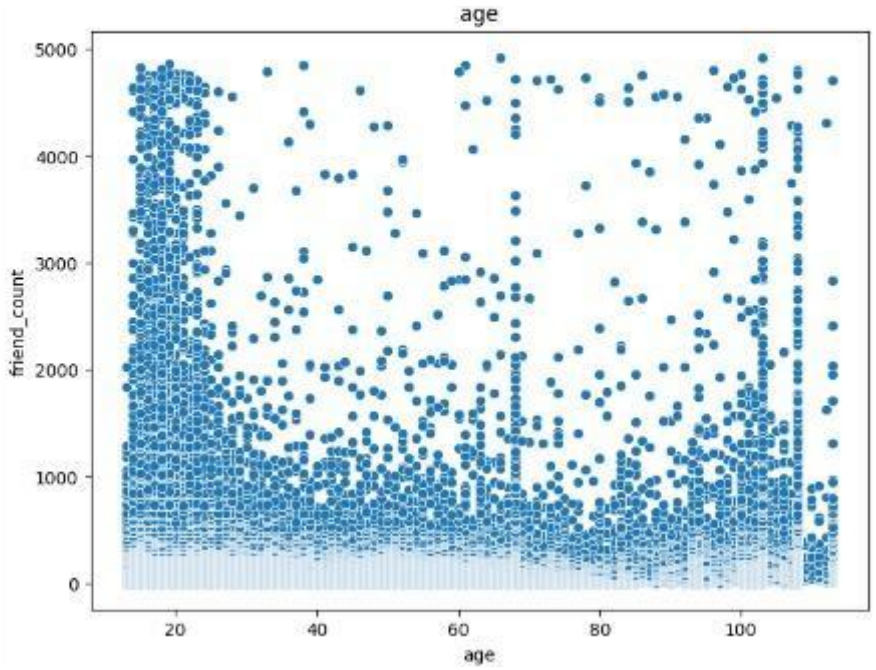




People between the ages 15 to 26 have many more users than people of other ages.

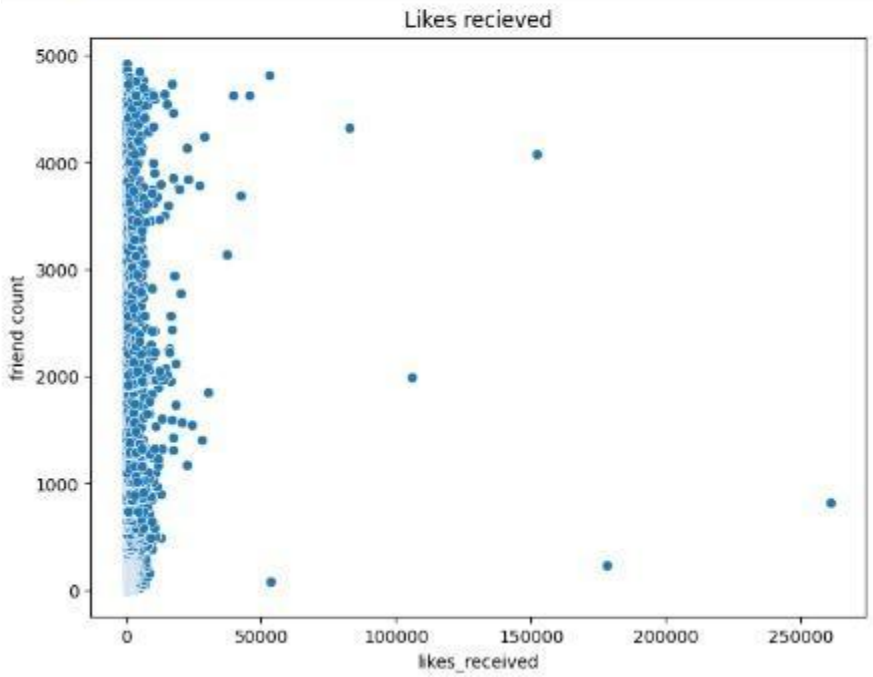
BIVARIATE ANALYSIS

1. age and friend_count

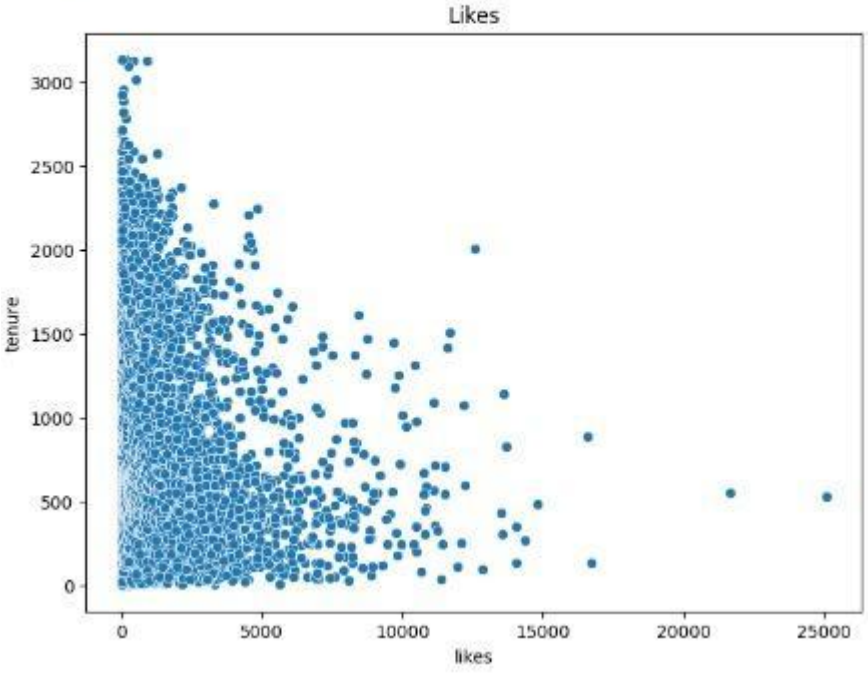


Most number of friend count are from age 15 to 25.

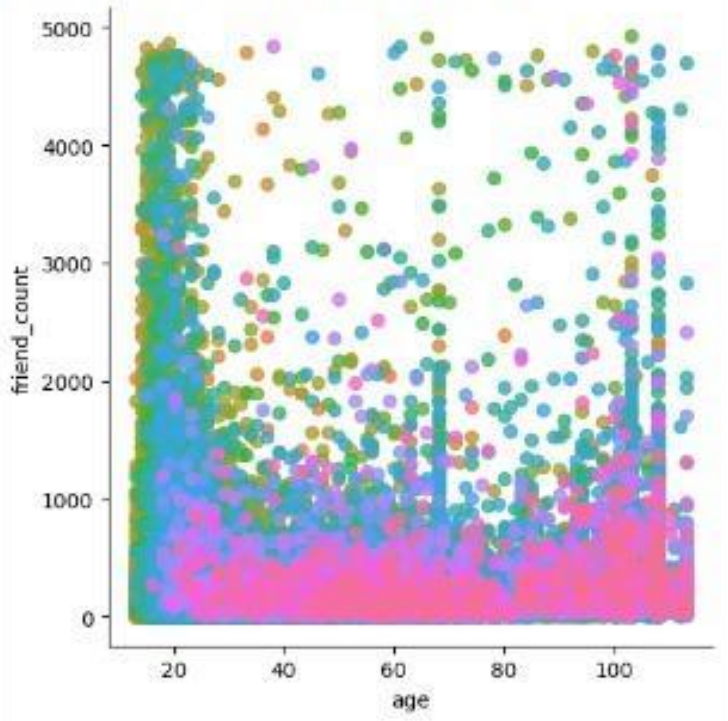
2. likes received and friend count



3. likes and tenure

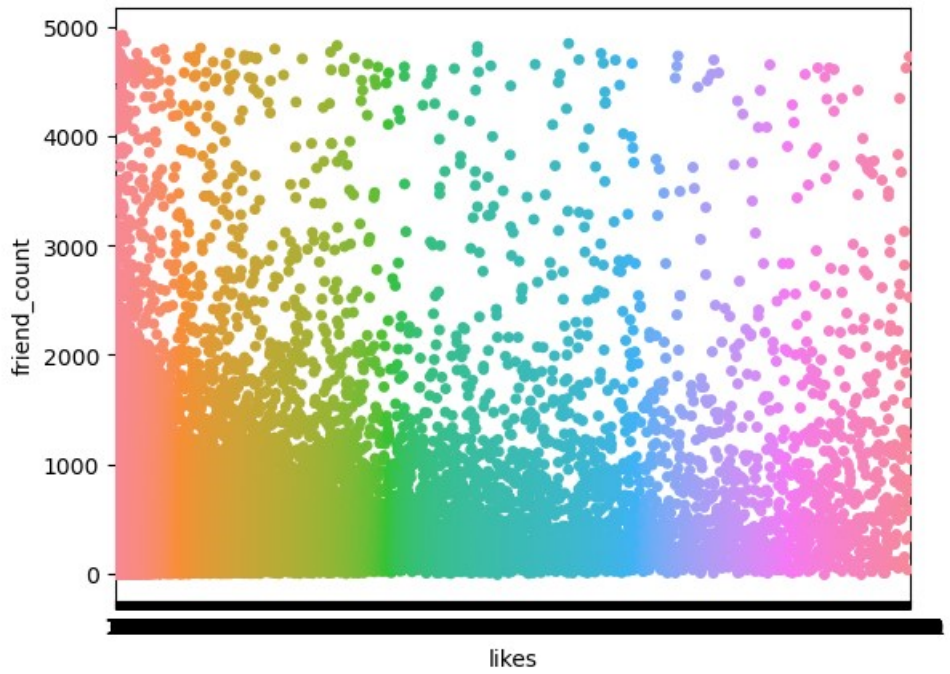


4. Lmplot for friend_count and age:



The above scatter plot represents friend count with age by overlaying tenure in the hue part.

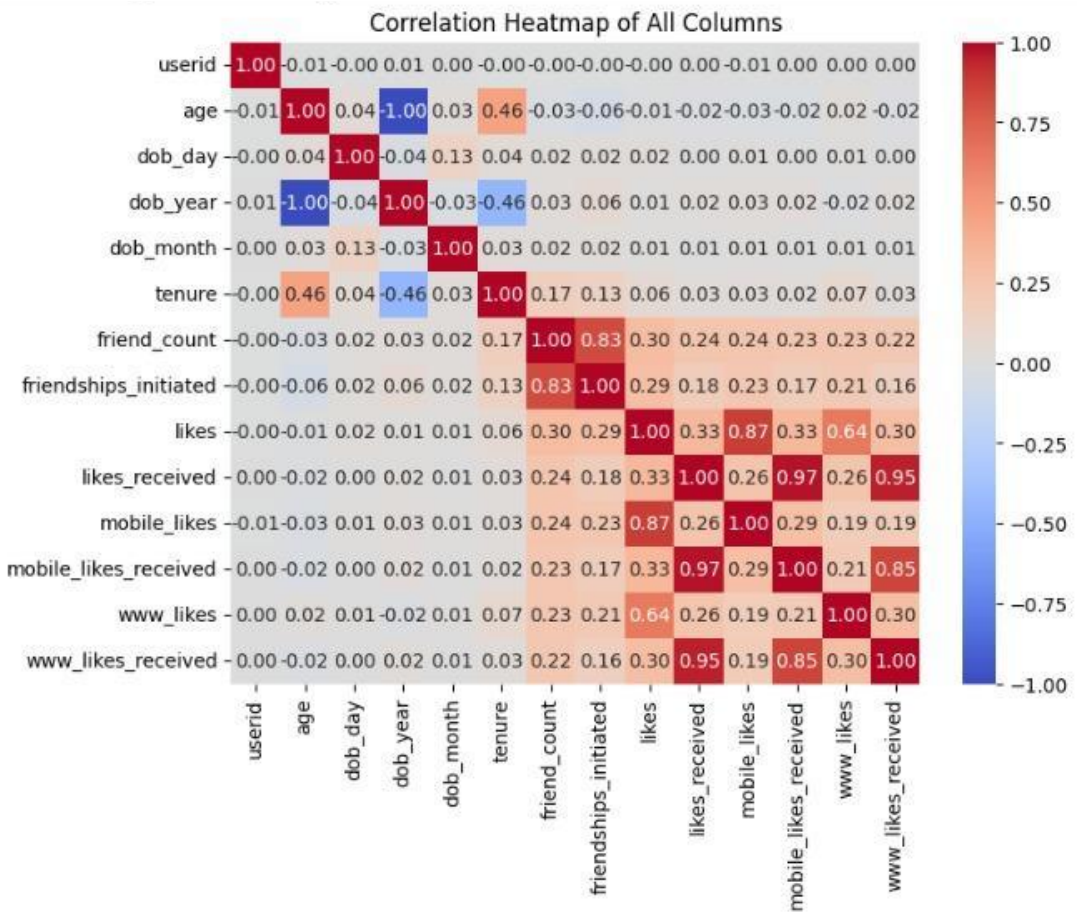
5. Stripplot for friend_count and likes



From the above strip plot we can relate two individual variables friend_count with number of likes people received.

MULTIVARIATE ANALYSIS

Correlation Heatmap :



Moderate Correlation between variables:

mobile_likes_recieved and www_likes_recieved (0.85), friend_count and friend_count_initiated(0.83), mobile_likes and likes (0.87), www_likes and likes(0.64)

Large correlation between Variables:

mobile_likes_recieved and likes_recieved(0.97), www_likes_recieved and likes_recieved (0.95)

No correlation between varaibles:

dob_year and age(-1)

ANALYSIS OF TARGET VARIABLE: ‘GENDER’

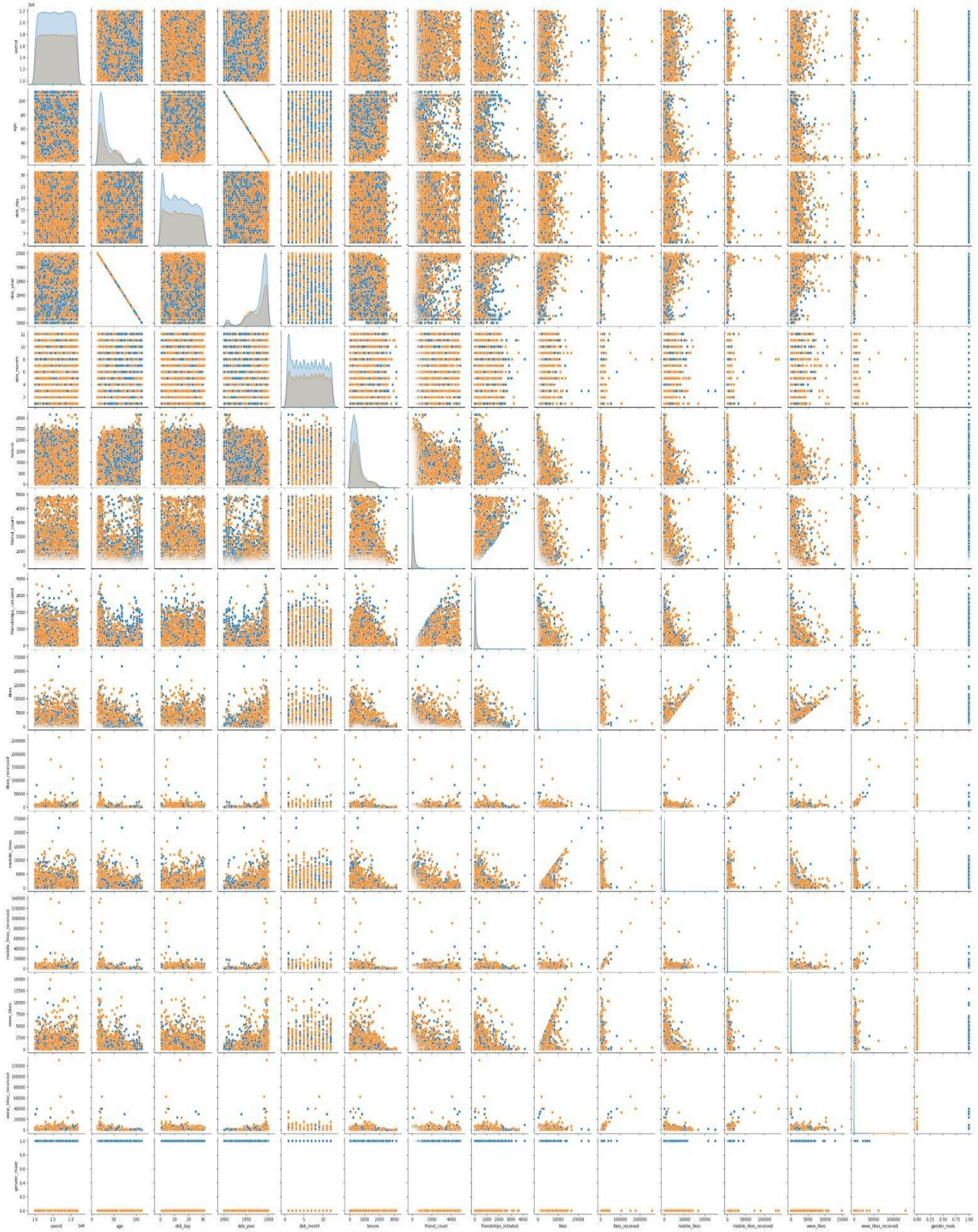
Converting the target variable ‘gender’ to dummy variables to perform categorical analysis

In [304]:
```
encoded_df = pd.get_dummies(df,columns=None)
encoded_df.head()
```

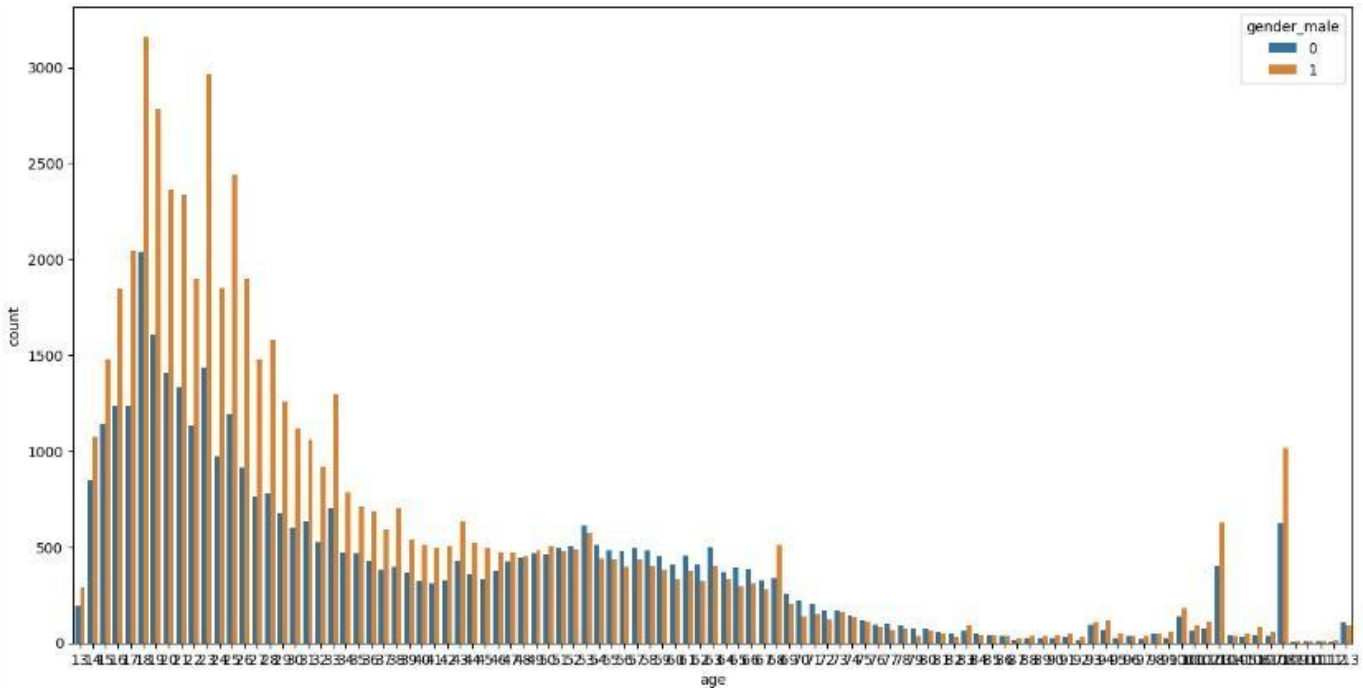
Out[304]:

tenure	friend_count	friendships_initiated	likes	likes_received	mobile_likes	mobile_likes_received	www_likes	www_likes_received	gender_female	gender_male
266.0	0	0	0	0	0	0	0	0	0	1
6.0	0	0	0	0	0	0	0	0	1	0
13.0	0	0	0	0	0	0	0	0	0	1
93.0	0	0	0	0	0	0	0	0	1	0
82.0	0	0	0	0	0	0	0	0	0	1

Pairplot Analysis :

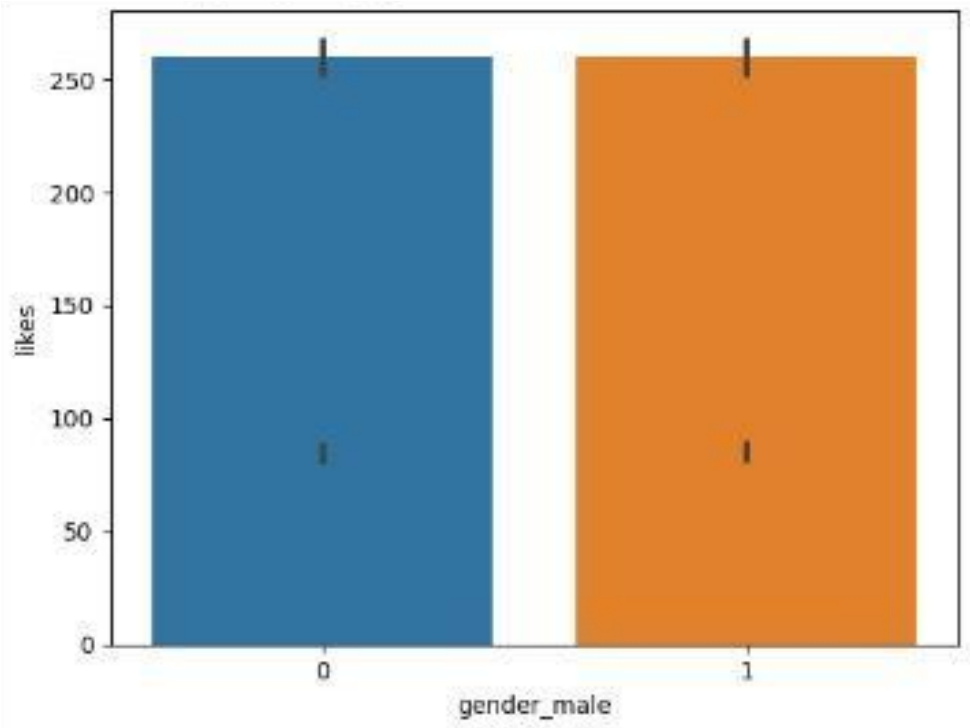


Countplot of Age vs Gender_male and Gender_female:

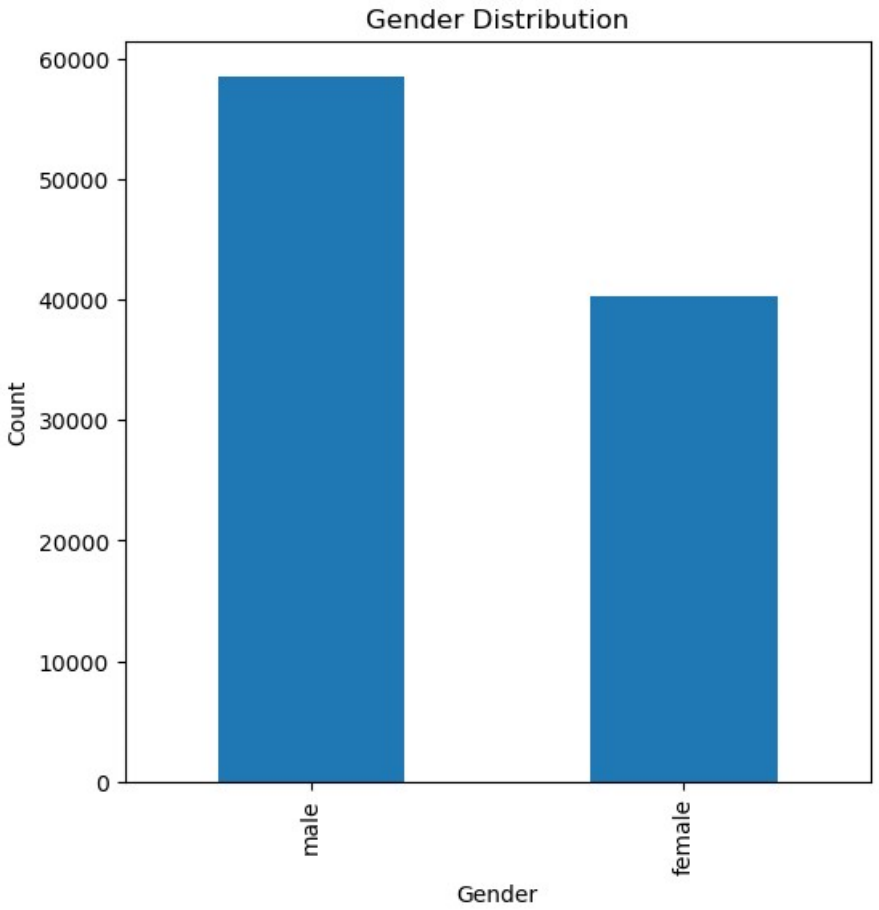


Barplots for analysing the gender distribution:

The above plot shows the count of male with their respective age.



The resulting bar plot has two bars, one for 'male' and one for 'female', each representing the average number of likes for that gender category. The height of each bar corresponds to the average number of likes for that gender.


Gender Distribution
60000
50000
40000
30000
20000
10000
0
Count
male
female
Gender

4. STATISTICAL ANALYSIS AND HYPOTHESIS TESTING

For categorical variable

1. Chi-Square Test for Independence

```
Chi-Square Statistic: 2200.435885316447
P-value: 0.0
Degrees of Freedom: 100
Failed to Reject the null hypothesis
```

2. T-Test

```
likes mean value: 156.10989800461408
likes_received mean value: 142.66334439632493
likes std value: 572.5450013570227
likes_received std value: 1388.9690668718147
p-value 0.0048978657655678595
we reject null hypothesis
```

5. FEATURE TRANSFORMATION AND EXTRACTION

1. Label Encoding for the target variable: ‘gender

	userid	age	dob_day	dob_year	dob_month	gender	tenure	\
0	2094382	14	19	1999	11	1	266.0	
1	1192601	14	2	1999	11	0	6.0	
2	2083884	14	16	1999	11	1	13.0	
3	1203168	14	25	1999	12	0	93.0	
4	1733186	14	4	1999	12	1	82.0	
...	...	...	...	...	...	...	...	
98998	1268299	68	4	1945	4	0	541.0	
98999	1256153	18	12	1995	3	0	21.0	
99000	1195943	15	10	1998	5	0	111.0	
99001	1468023	23	11	1990	4	0	416.0	
99002	1397896	39	15	1974	5	0	397.0	

	friend_count	friendships_initiated	likes	likes_received	\
0	0	0	0	0	
1	0	0	0	0	
2	0	0	0	0	
3	0	0	0	0	
4	0	0	0	0	
...	...	...	...	...	
98998	2118	341	3996	18089	
98999	1968	1720	4401	13412	
99000	2002	1524	11959	12554	
99001	2560	185	4506	6516	
99002	2049	768	9410	12443	

	mobile_likes	mobile_likes_received	www_likes	www_likes_received
0	0	0	0	0
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0
...	...	...	...	...
98998	3505	11887	491	6202
98999	4399	10592	2	2820
99000	11959	11462	0	1092
99001	4506	5760	0	756
99002	9410	9530	0	2913

[98828 rows x 15 columns]

Dropping irrelevant and highly correlated columns (as seen in the correlation heatmap)

```
         userid  age  dob_day  dob_year  dob_month  gender  tenure  \
0       2094382   14       19      1999         11       1   266.0
1       1192601   14        2      1999         11       0     6.0
2       2083884   14       16      1999         11       1    13.0
3       1203168   14       25      1999         12       0    93.0
4       1733186   14        4      1999         12       1    82.0
...         ...  ...      ...       ...        ...     ...     ...
98998   1268299   68        4      1945          4       0   541.0
98999   1256153   18       12      1995          3       0    21.0
99000   1195943   15       10      1998          5       0   111.0
99001   1468023   23       11      1990          4       0   416.0
99002   1397896   39       15      1974          5       0   397.0

       friend_count  friendships_initiated  likes
0                 0                      0      0
1                 0                      0      0
2                 0                      0      0
3                 0                      0      0
4                 0                      0      0
...             ...                    ...    ...
98998          2118                    341   3996
98999          1968                   1720   4401
99000          2002                   1524  11959
99001          2560                    185   4506
99002          2049                    768   9410

[98828 rows x 10 columns]
```

Feature Scaling using Standard Scaler:

	userid	age	dob_day	dob_year	dob_month	tenure	friend_count	friendships_initiated	likes
0	1.445623	-1.030561	0.495575	1.030561	1.335965	-0.593607	-0.506821	-0.569096	-0.27266
1	-1.175754	-1.030561	-1.390415	1.030561	1.335965	-1.165977	-0.506821	-0.569096	-0.27266
2	1.415107	-1.030561	0.162754	1.030561	1.335965	-1.150567	-0.506821	-0.569096	-0.27266
3	-1.145037	-1.030561	1.161219	1.030561	1.619294	-0.974453	-0.506821	-0.569096	-0.27266
4	0.395666	-1.030561	-1.168534	1.030561	1.619294	-0.998669	-0.506821	-0.569096	-0.27266

6. BUILDING OF THE BASE MODEL : Logistic Regression

```
: #BASE MODEL BUILDING

  #LOGISTIC REGRESSION

  from sklearn.linear_model import LogisticRegression

  model_LR = LogisticRegression()

  X_train, X_test, y_train, y_test= train_test_split(X,y, test_size= 0.3, random_state= 42)

  model_LR.fit(X_train, y_train )

  predictions = model_LR.predict(X_test)

  print("Accuracy", accuracy_score(y_test, predictions))
```

```
Accuracy 0.6344227461297177
```

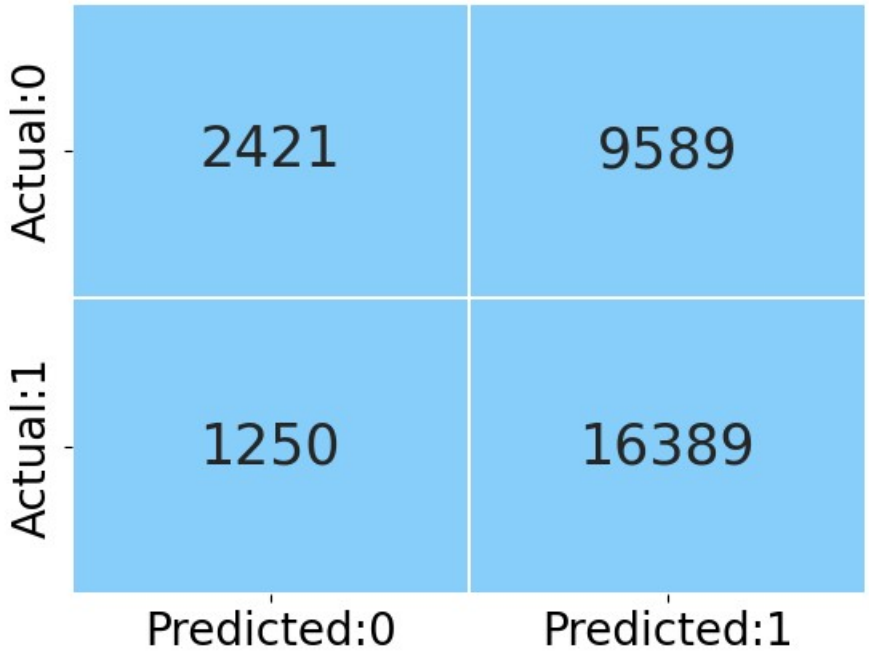
Hyper parameter tuning using Randomized Search CV on Logistic Regression

```
# Create RandomizedSearchCV
random_search = RandomizedSearchCV(model_LR, param_distributions=params_dist_LR, n_iter=10, cv=5, random_state=42)

# Fit the model and perform the hyperparameter search
random_search.fit(X_train, y_train)

# Best hyperparameters found
print("Best hyperparameters found: ", random_search.best_params_)
```

```
Best hyperparameters found:  {'penalty': 'l2', 'C': 0.01}
```

```
Best Hyperparameters for Logistic Regression:  {'penalty': 'l2', 'C': 0.01}
Accuracy:  0.6344227461297177
Classification Report:
              precision    recall  f1-score   support

           0       0.66      0.20      0.31     12010
           1       0.63      0.93      0.75     17639

    accuracy                           0.63     29649
   macro avg       0.65      0.57      0.53     29649
weighted avg       0.64      0.63      0.57     29649
```

Confusion Matrix :



Precision : `0.6308799753637694`
Recall : `0.9291343046657974`
Accuracy : `0.6344227461297177`

Model II: RANDOM FOREST CLASSIFIER

```
#MODEL-2 RANDOM FOREST CLASSIFIER USING RANDOMIZED SEARCH CV

model_RF= RandomForestClassifier()
model_RF.fit(X_train, y_train )

y_pred_RF = model_RF.predict(X_test)
print("Accuracy", accuracy_score(y_test, y_pred_RF))
```

```
Accuracy 0.6843401126513542
```

```
Best Hyperparameters for Gradient Boosting:  {'n_estimators': 150, 'min_samples_split': 12, 'min_samples_leaf': 13, 'max_dept
h': 15, 'bootstrap': True}
Accuracy:  0.6913555263246652
Classification Report:
              precision    recall  f1-score   support

           0       0.66      0.49      0.56     12010
           1       0.71      0.83      0.76     17639

    accuracy                           0.69     29649
   macro avg       0.68      0.66      0.66     29649
weighted avg       0.69      0.69      0.68     29649
```

Model III: GRADIENT BOOSTING

```
#MODEL-3 GRADIENT BOOSTING

model_GB = GradientBoostingClassifier()
model_GB.fit(X_train, y_train )

y_pred_GB = model_GB.predict(X_test)
print("Accuracy", accuracy_score(y_test, y_pred_GB))
```

```
Accuracy 0.6900738642112719
```

```
Best Hyperparameters for Gradient Boosting:  {'n_estimators': 160, 'max_depth': 3, 'learning_rate': 0.2}
Accuracy:  0.6933792033458127
Classification Report:
              precision    recall  f1-score   support

           0       0.66      0.49      0.56     12010
           1       0.71      0.83      0.76     17639

    accuracy                           0.69     29649
   macro avg       0.69      0.66      0.66     29649
weighted avg       0.69      0.69      0.68     29649
```

Model IV: NAIVE BAYES CLASSIFIER

```
#MODEL-4 NAIVE BAYES CLASSIFIER


model_NB = GaussianNB()
model_NB.fit(X_train, y_train )

y_pred_NB = model_NB.predict(X_test)

print("Accuracy", accuracy_score(y_test, y_pred_NB))
```

```
Accuracy 0.6176599548045465
```

```
Best Hyperparameters for Naive Bayes:  {'var_smoothing': 1.232846739442066e-07, 'priors': None}
Accuracy:  0.6205942864852103
Classification Report:
              precision    recall  f1-score   support

           0       0.63      0.16      0.25     12010
           1       0.62      0.94      0.75     17639

    accuracy                           0.62     29649
   macro avg       0.62      0.55      0.50     29649
weighted avg       0.62      0.62      0.55     29649
```

Model V: K NEIGHBORS CLASSIFIER

```
#MODEL-5 K NEIGHBORS CLASSIFIER


model_KNN = KNeighborsClassifier()
model_KNN.fit(X_train, y_train )

y_pred_KNN = model_KNN.predict(X_test)

print("Accuracy", accuracy_score(y_test, y_pred_KNN))
```

```
/opt/anaconda3/lib/python3.9/site-packages/sklearn/neighbors/_classification.py:228: FutureWarning: Unlike other reduction func
tions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, th
is behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will b
e eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.
  mode, _ = stats.mode(_y[neigh_ind, k], axis=1)
```

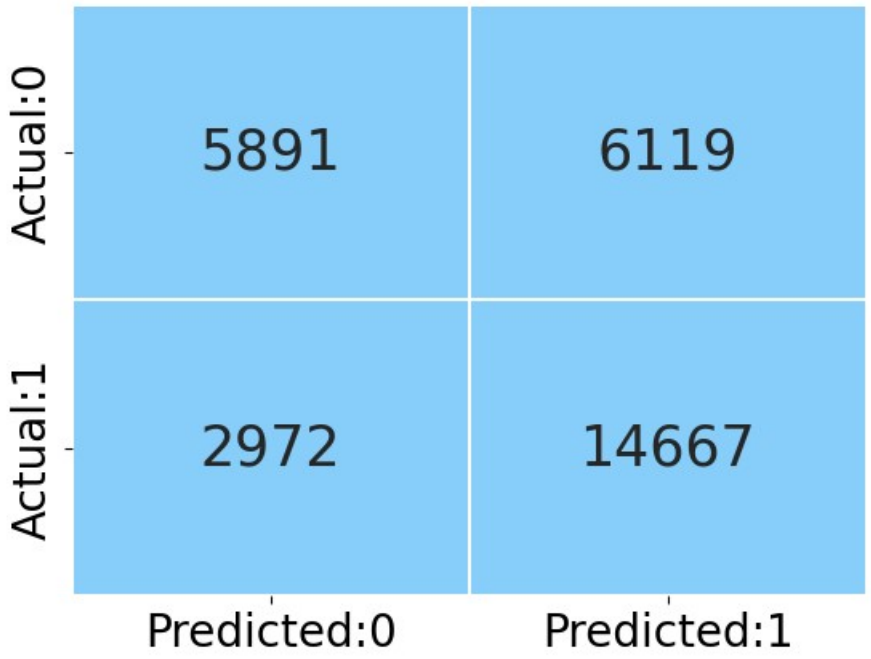
```
Accuracy 0.5825491584876387
```

```
Best Hyperparameters for K Neighbors:  {'weights': 'distance', 'p': 1, 'n_neighbors': 19}
Accuracy:  0.6018078181388917
Classification Report:
              precision    recall  f1-score   support

           0       0.52      0.21      0.30     12010
           1       0.62      0.87      0.72     17639

    accuracy                           0.60     29649
   macro avg       0.57      0.54      0.51     29649
weighted avg       0.58      0.60      0.55     29649
```

7. FINAL MODEL : GRADIENT BOOSTING

Model with the highest accuracy that is 69.33% is chosen

```
#FINAL MODEL CHOSEN:  GRADIENT BOOSTING CLASSIFIER WITH RANDOMIZED SEARCH (ACCURACY=70%)


cm = confusion_matrix(y_test, y_pred_GB)

# label the confusion matrix
# pass the matrix as 'data'
# pass the required column names to the parameter, 'columns'
# pass the required row names to the parameter, 'index'
conf_matrix = pd.DataFrame(data = cm,columns = ['Predicted:0','Predicted:1'], index = ['Actual:0','Actual:1'])


sns.heatmap(conf_matrix, annot = True, fmt = 'd', cmap = ListedColormap(['lightskyblue']), cbar = False,
            linewidths = 0.1, annot_kws = {'size':25})

# set the font size of x-axis ticks using 'fontsize'
plt.xticks(fontsize = 20)

# set the font size of y-axis ticks using 'fontsize'
plt.yticks(fontsize = 20)

# display the plot
plt.show()
```

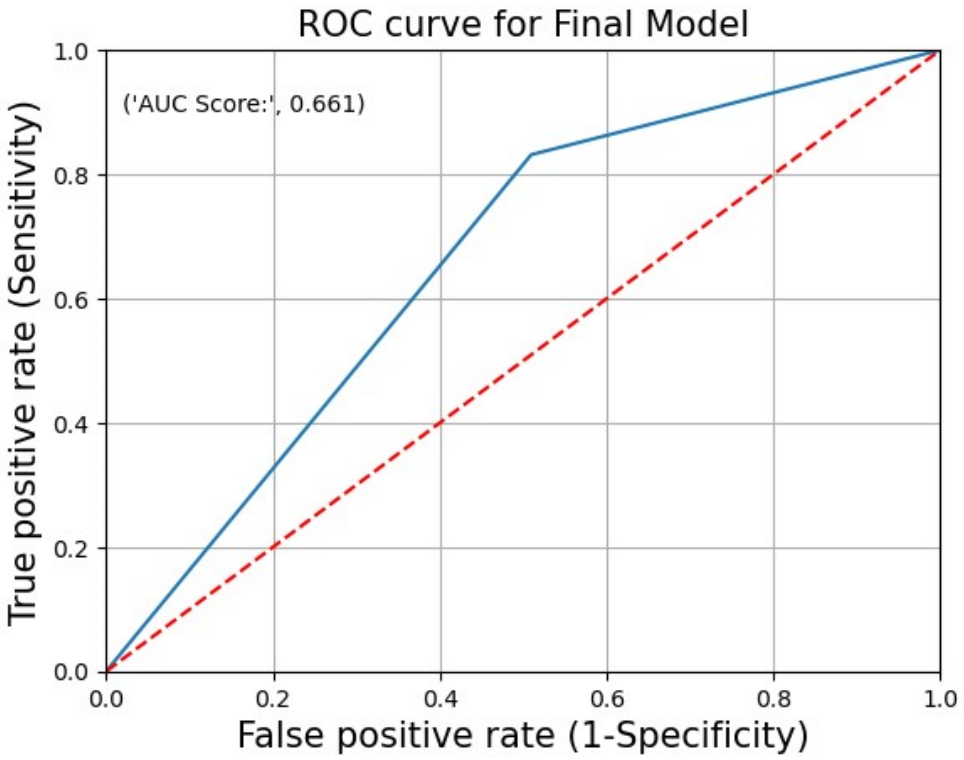



```
# True Negatives are denoted by 'TN'
# Actual '0' values which are classified correctly
TN = cm[0,0]

# True Positives are denoted by 'TP'
# Actual '1' values which are classified correctly
TP = cm[1,1]

# False Positives are denoted by 'FP'
# it is the type 1 error
# Actual '0' values which are classified wrongly as '1'
FP = cm[0,1]

# False Negatives are denoted by 'FN'
# it is the type 2 error
# Actual '1' values which are classified wrongly as '0'
FN = cm[1,0]
```

Precision : 0.7056191667468489
Recall : 0.8315097227733999
Specificity : 0.49050791007493755
F1 Score : 0.7634092387768382
Accuracy : 0.6933792033458127

	precision	recall	f1-score	support
0	0.66	0.49	0.56	12010
1	0.71	0.83	0.76	17639
accuracy			0.69	29649
macro avg	0.69	0.66	0.66	29649
weighted avg	0.69	0.69	0.68	29649

kappa value: 0.3360686927394966



8. Advantages and Limitations of Gradient Boosting Model

Advantages:

1. High Predictive Power: Gradient Boosting often leads to more accurate predictions compared to other algorithms. It can handle complex interactions in data, making it suitable for a wide range of applications.

2. Handles Mixed Data: It can handle different types of data, whether numerical or categorical, without requiring extensive preprocessing.

3. Feature Importance Estimation: The model provides a feature importance analysis, which can help in understanding the relative importance of different features in the prediction.

4. Robustness to Outliers: Gradient Boosting is relatively robust to outliers and can handle them effectively compared to other algorithms like Support Vector Machines.

5. Flexibility and Customization: It allows the use of different loss functions, making it highly customizable for different types of predictive tasks.

Disadvantages:

1. Computational Complexity: Training a Gradient Boosting model can be computationally expensive, especially when dealing with a large number of iterations and complex data.

2. Prone to Overfitting: Without proper parameter tuning, Gradient Boosting models can overfit the training data, leading to poor generalization on unseen data.

3. Sensitivity to Noisy Data and Outliers: While it is relatively robust to outliers compared to some other models, it can still be sensitive to noisy data, which can affect its performance.

4. Black Box Model: Interpretability can be a challenge with Gradient Boosting, especially when dealing with a large number of trees. Understanding the internal workings of the model can be difficult, making it challenging to explain to non-technical stakeholders.

5. Hyperparameter Tuning: The performance of Gradient Boosting models can be sensitive to the choice of hyperparameters. Proper tuning is necessary to achieve the best results, which can be time-consuming and require a good understanding of the algorithms.

9. Closing Reflections

The project "Exploring Facebook User Data: Recommendations" aimed to analyze and gain insights from a dataset containing Facebook user data, with a specific focus on understanding user preferences and generating personalized recommendations. The dataset includes information about users' gender and the number of likes they have received on their posts.

As our target variable was chosen as ‘Gender’ , we decided to go with the Classification algorithm and used the most popular ones like : Logistic regression, Random Forest Classifier, Gradient Boosting, KNN , Naive Bayes. We started by fitting our training set data and predicting test data.

Further , we compared the KPIs of all the models , both using classification report and by calculating metrics manually using confusion matrix. And Validate the model using Random Search Cross-Validation where Random search is an alternative to grid search where random combinations of the hyperparameters are used.

Gradient boosting model showed highest accuracy that is 69.33% and this was selected as the best model to generalise personal recommendation based on gender.