Predictive Analytics for Better Marketing Performance

Prerna Mishra (pm3152)

Overview

Digital advertising is promotional material delivered to a target audience through digital platforms, including social media, email, search engines, mobile apps, affiliate programs, and websites. One of the main benefits of digital advertising is that an advertiser can track in real-time the success of the ad-campaigns. The digital marketing spending market has the potential to grow by USD 128.83 billion during 2021-2025, and the market's growth momentum will accelerate at a CAGR of 6.53%. New digital marketing strategies are continuously emerging based on the unprecedented access to vast amounts of information about products, firms, and consumer behavior.

Traditional marketing has always been about the 4Ps: Product, Price, Place, and Promotion. Here we see how the digital revolution has transformed everything and augmented them with the 5th P of Participation (by consumers). To increase the participation by consumers, the goal of digital advertising becomes to inorganically advertise where consumers are and to customize ads to the target audience's preferences, to achieve higher conversions (ratio of Products bought through ads / number of ads shown).

This project is about coming up with a predictive model to identify the segments / clusters of users that demonstrate higher conversion rates for specific ads. This in-turn could help the companies target specific segments of users to improve their conversion rates.

Introduction

With the consumer data available with companies, and the dependency on digital advertising, there are several impactful questions that we want to re-consider with this project. In this project, we work on a dataset that has information about consumers and viewers of several ads, and associated details - age, gender, clicks, impressions, conversions. Through this project, we try to identify the segments of users who are more likely to end up buying the product once they are shown the ad.

This same solution could then be extended to work with a more sophisticated dataset containing user-preferences, interests, order-history, etc. to classify users as prospective consumers with more confidence. This could help companies identify better their target prospective customers

and help them reduce their advertising costs by targeting their advertisements towards these prospective customers.

Typically, the conversion rate of enquiries to purchases is about 25%. The problem statement could be thought of in two ways. Firstly, we could treat it as a regression problem, and figure out the estimated number of purchases using values of other variables. Secondly, we could treat it as a classification problem, where we could try to figure out, based on the values of other variables, whether the chances for conversion are above a certain threshold (say the typical of 25%) and find the customer conversion to buy a certain product based on their profile and ads shown.

In this project, we have focused on the second way to look at the problem statement. We have analyzed the problem using several models especially focusing on XGBoost Classifier and have captured the parameters that work best for us.

Related Work

There are several relevant research papers that work on similar lines as this project. For instance, the paper – "*An Empirical Analysis of Search Engine Advertising: Sponsored Search and Cross-Selling in Electronic Markets"* by Anindya Ghose and Sha Yang (Stern School of Business, NYU) uses Bayesian models to analyze search engine advertisements and related costs per ad-slot. The paper has limitations mainly related to the quality of the dataset, which is an underlying problem in this analysis as well. "*Classification and prediction based data mining algorithms to predict email marketing campaigns*" is on similar lines, but mainly related to email marketing campaigns. Specifically with respect to the optimizations and parameters with respect to XGBoost, we can see a similar set of classification attempted in "An Xgboost based system for financial fraud detection", by Shimin LEI1,a, Ke XU2,b, YiZhe HUANG3,c, Xinye SHA4, and the paper "Predicting the Risk of Chronic Kidney Disease" by Hippisley-Cox, Julia and Coupland, Carol.

There is some research on using Bayesian Learning in Retailing, for example in this paper - "The Role of Big Data and Predictive Analytics in Retailing", by Eric T. Bradlow Praveen Kopalle Sudhir Voleti. As a parallel to this project, where the learning has been applied explicitly for advertisements, the paper draws conclusions for a larger dataset for the entire retail funnel - search / discovery, etc.

Data Understanding

Dataset

The dataset that we have used has been picked up from Kaggle. The data in this dataset has been collected from Social Media Marketing Ad campaigns - Facebook Advertising Campaigns. It uses social media consumer profile data which uses Google Adwords to track conversions. This dataset had features which contained consumer profiles along with ad campaigns on social media.

This specific dataset was chosen for this analysis because this dataset had a good number of relevant features, such as - age, gender, clicks, impressions, conversions, and spent - that seemed useful for the analysis. Additionally, unlike other datasets, this dataset does not contain any user-sensitive information such as their earnings / preferences / demographics.

This dataset helps us understand how different segments of consumers interact with different categories of advertisements. With the help of different consumer features like gender and age-group, we try to segment them based on their interactions with ads - like impressions, clicks, and spent. Using these features, we try to maximize the conversion ratio from a random ad-viewer to a prospective customer. This in turn will help us better understand customer behavior (based on past interactions, and known interests of different customer-segments), potentially optimize advertisement costs from the company's perspective, and help companies qualify and prioritize leads (identify ideal audience that is closer to conversion), and retain customers.

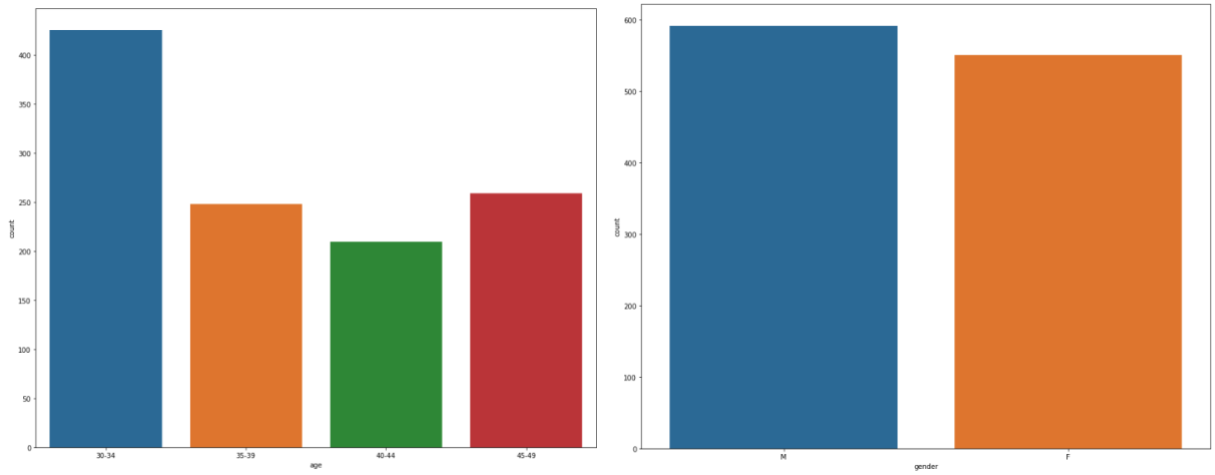
The features in the dataset are:

1. ad_id: unique ID for each ad.
2. xyz*campaign*id: an ID associated with each ad campaign of XYZ company.
3. fb*campaign*id: an ID associated with how Facebook tracks each campaign.
4. age: age of the person to whom the ad is shown.
5. gender: gender of the person to whom the ad is shown.
6. interest: a code specifying the category to which the person's interest belongs (interests are as mentioned in the person's Facebook public profile).
7. Impressions: the number of times the ad was shown.
8. Clicks: number of clicks on for that ad.
9. Spent: Amount paid by company xyz to Facebook, to show that ad.
10. Total conversion: Total number of people who enquired about the product after seeing the ad.
11. Approved conversion: Total number of people who bought the product after seeing the ad.

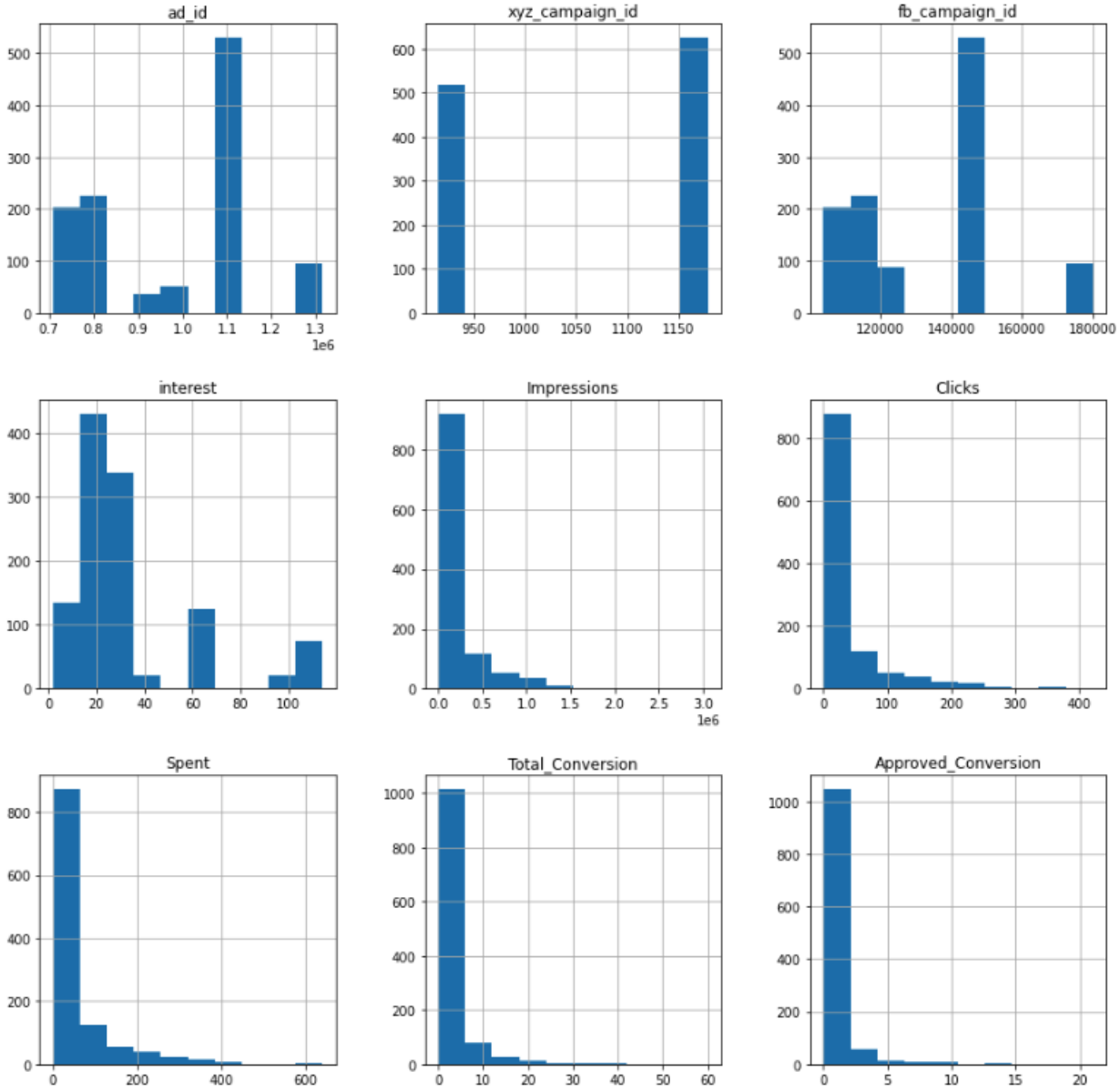
Data Preparation

This dataset has 1143 rows, which contain unique ad_ids for each ad and 11 different features containing non-null values. This dataset has been pre-processed for each ad, fb_campaign_id, company's campaign id, which contains information about how social media like Facebook tracks each ad along with the users' profile information.

We have numerical data as well as categorical data. The categorical data includes gender (M, F), and age (age groups: specified as 5 year brackets - 30-34, 35-39, 40-44, 45-49).



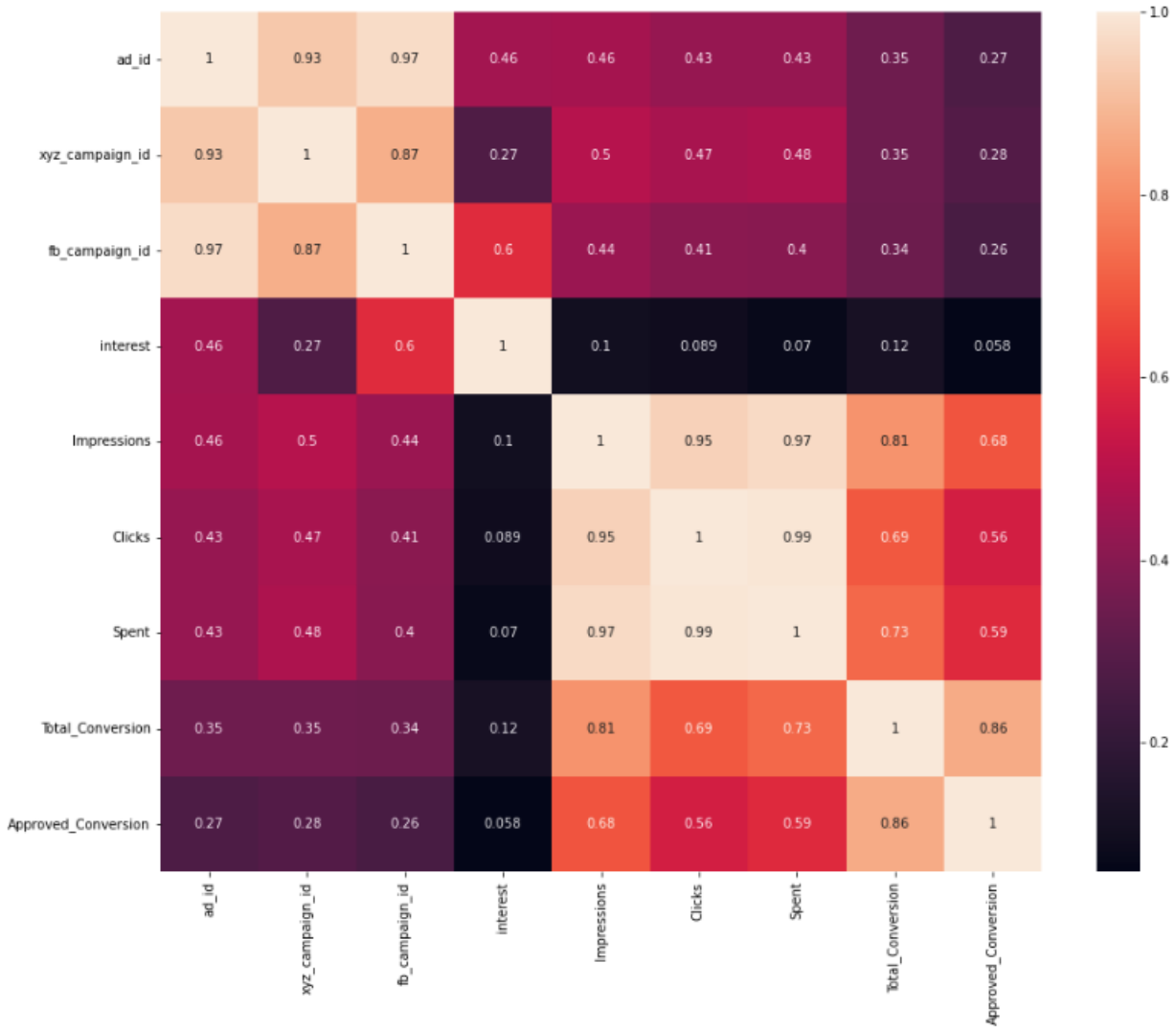
Distribution of other variables can be visualized as -



Since the dataset is clearly pre-processed and does not contain any missing values, removal/handling of NaN or skewed values was not required. Categorical variables have been encoded with LabelEncoder, which assigns numerical values against different values of categorical variables. Since the values in some of the columns had a broad range of values, to scale the distance we have used StandardScaler.

Correlation analysis of the features in the dataset was done using the heatmap and the features ad_id and xyz_campaign_id were very highly correlated with each other. Other features like

interest, impressions, clicks, spent are comparatively less correlated. Also, the feature interest variable seems to be the least correlated.

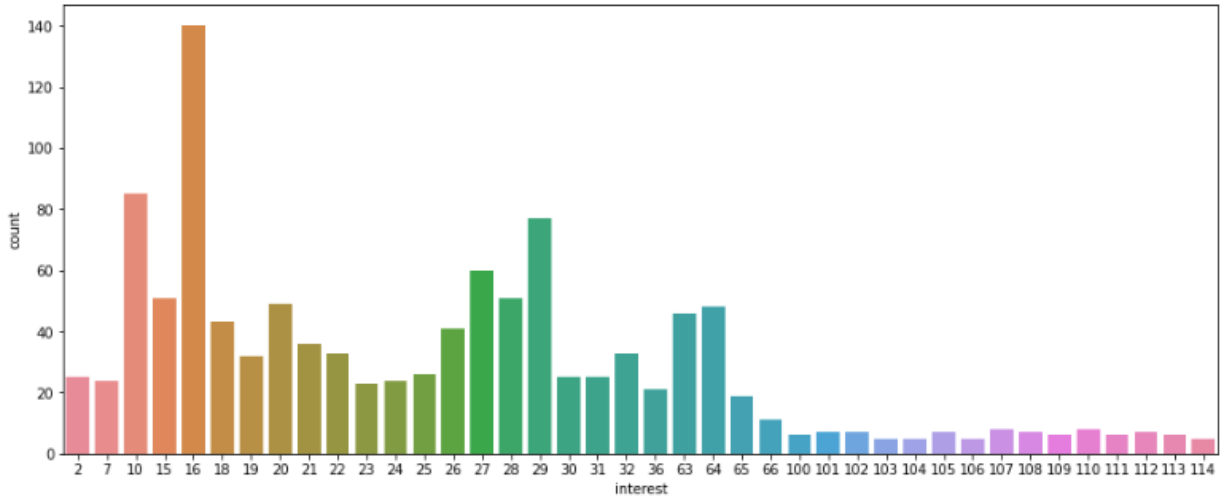


Hence, looking at the correlation matrix, for higher correlation coefficients, we can reduce the dimension of the dataset using Principal Component Analysis. Since we have 11 different features, looking at the coefficients we can reduce the dimensions to 5 components.

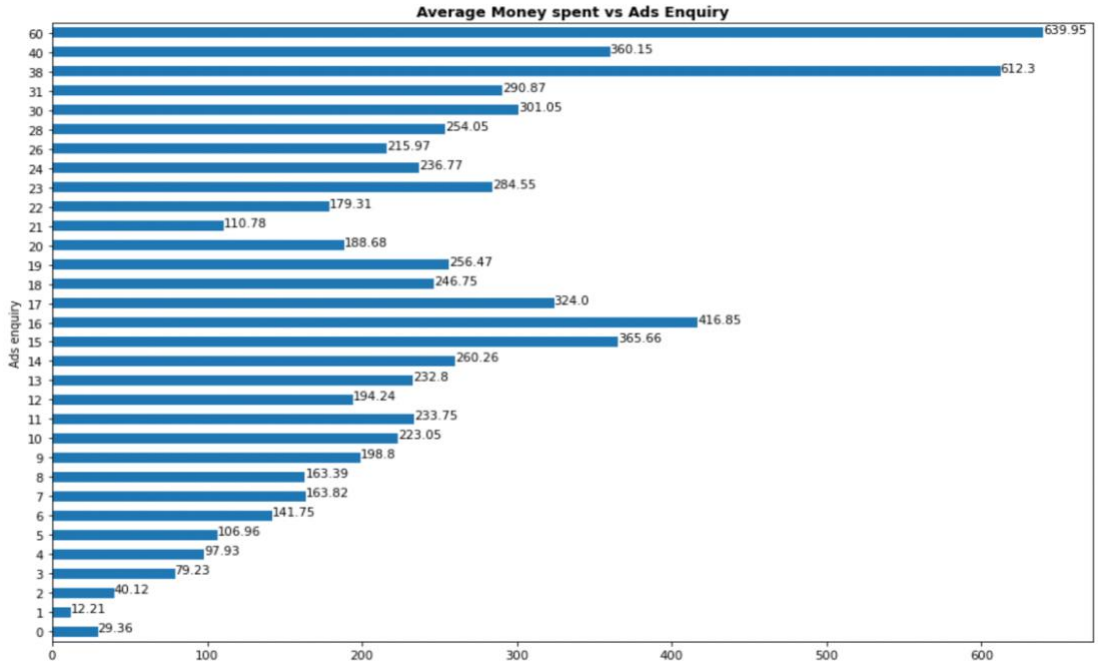
Data Visualization

We have further done Exploratory Data Analysis on our dataset and derived insights for our predictive modeling using scatter plots from Matplotlib and Seaborn libraries.

Mean spending on ads by Interest topic



Average Money spent vs Ads Enquiry

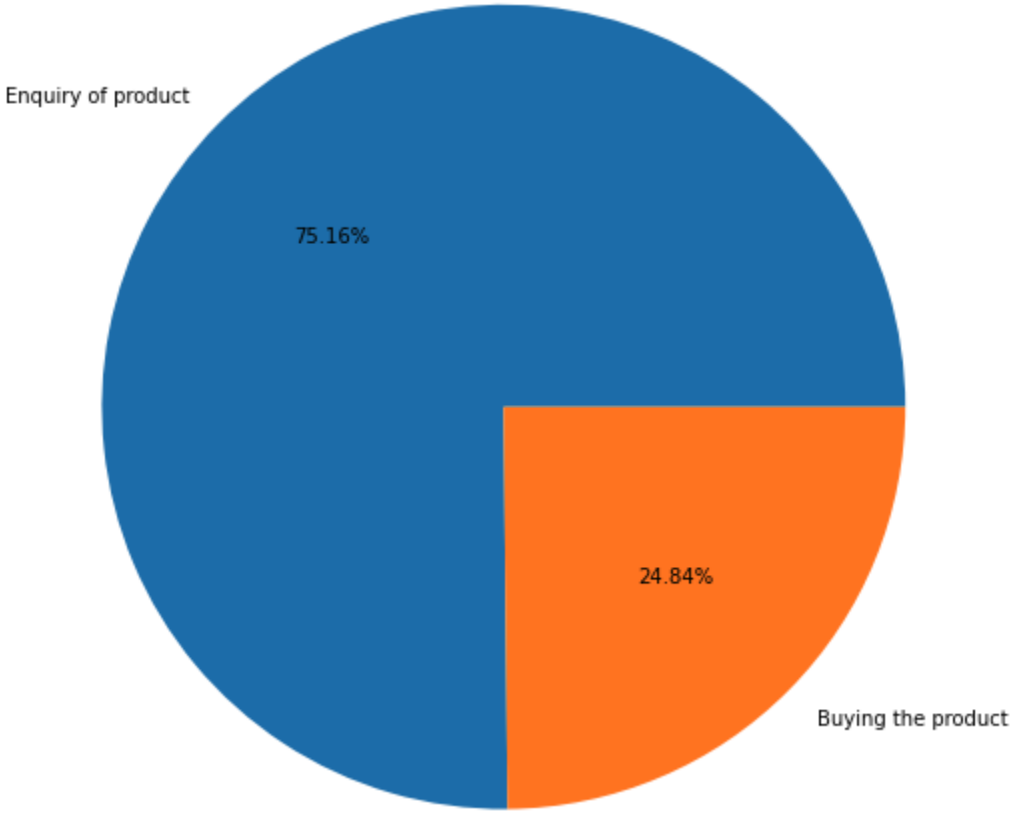


Insights from above graph: When the company spent an average of:

1. If a company spends $10-200 on Ads, they get 120 Ads enquiries.
2. If a company spends $200-400 on Ads, they get 319 Ads enquiries.
3. If a company spends more than $400 on Ads, they get 114 Ads enquiries.

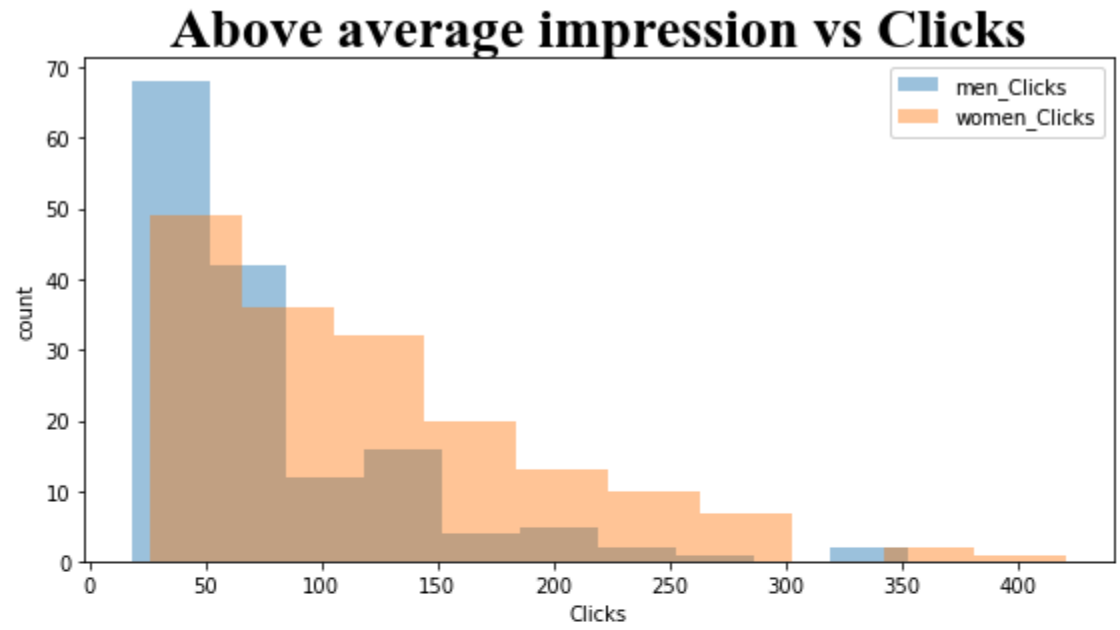
From above insights, we can conclude that spending between $200-400 on Ads seems reasonable for any company because it yields the max Ads enquiry.

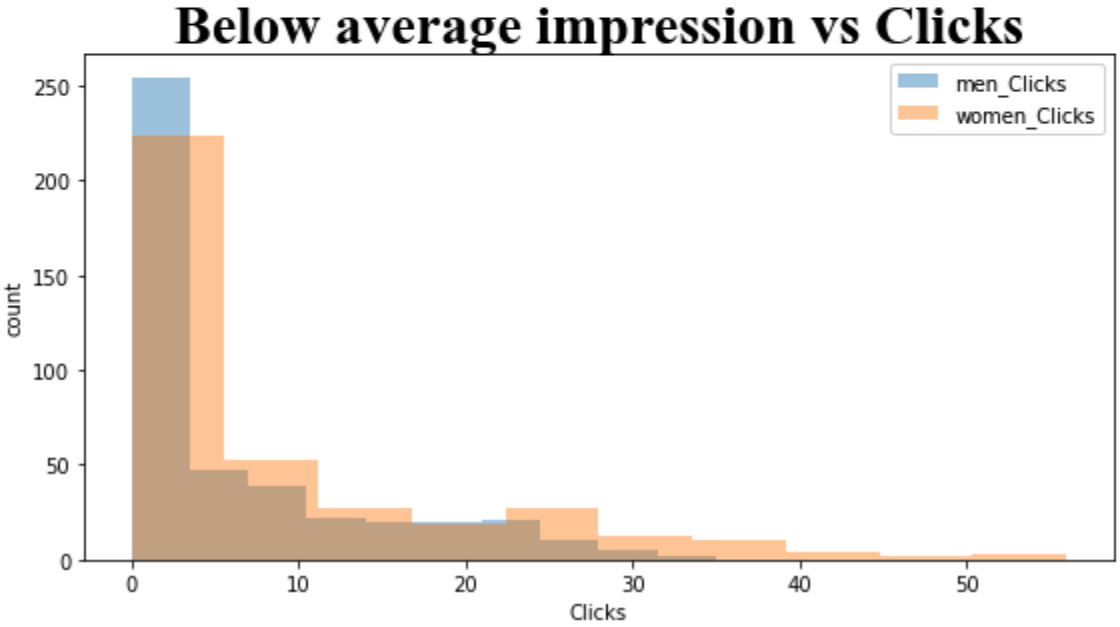
Clicks vs Conversions (Enquiry vs Bought)



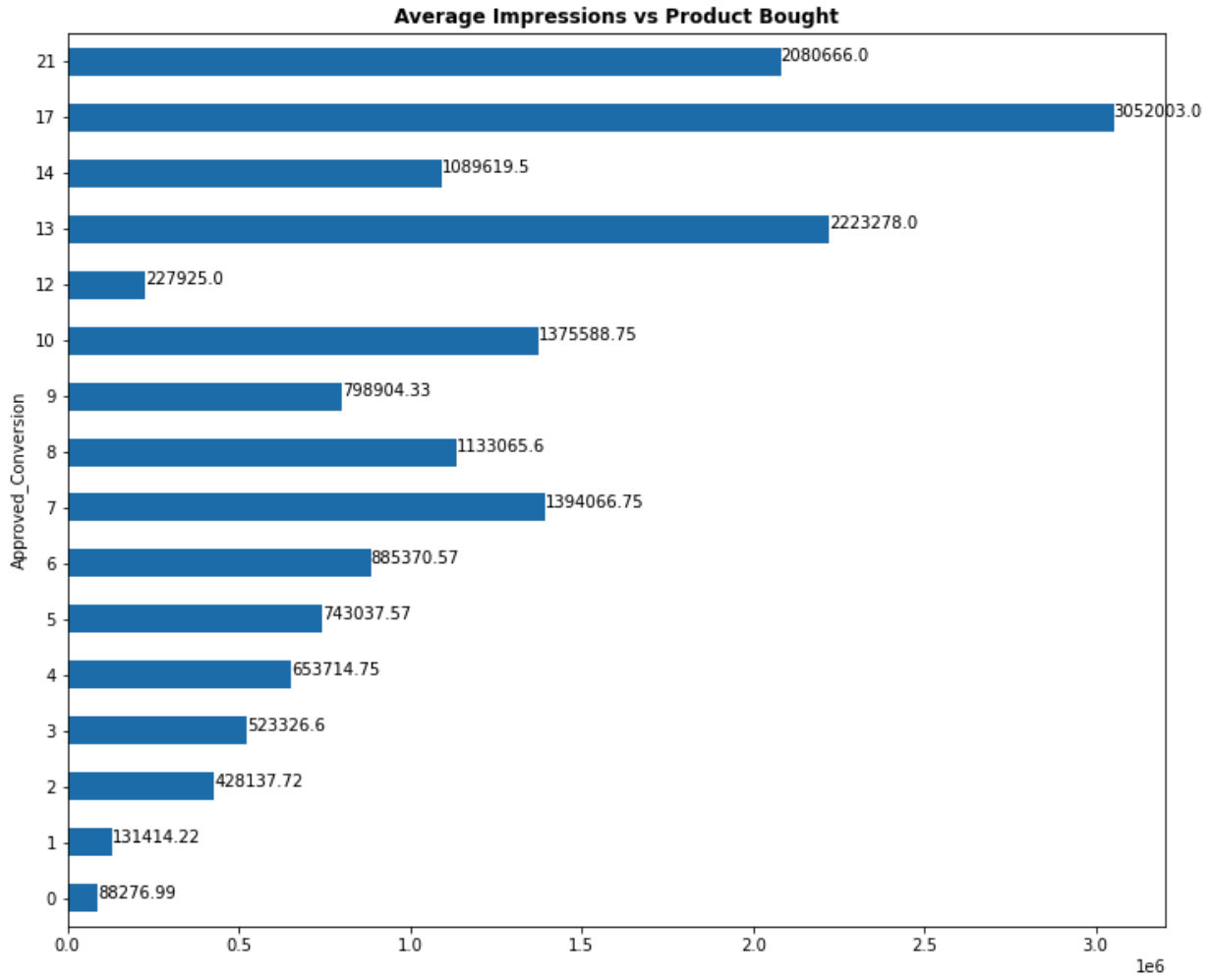
From above, we can conclude that about 25% of the people who enquired about the product end up buying the product.

The distribution plot of above average Impression for men and women with respect to Clicks gives the following histograms:





Average Impressions vs Product Bought



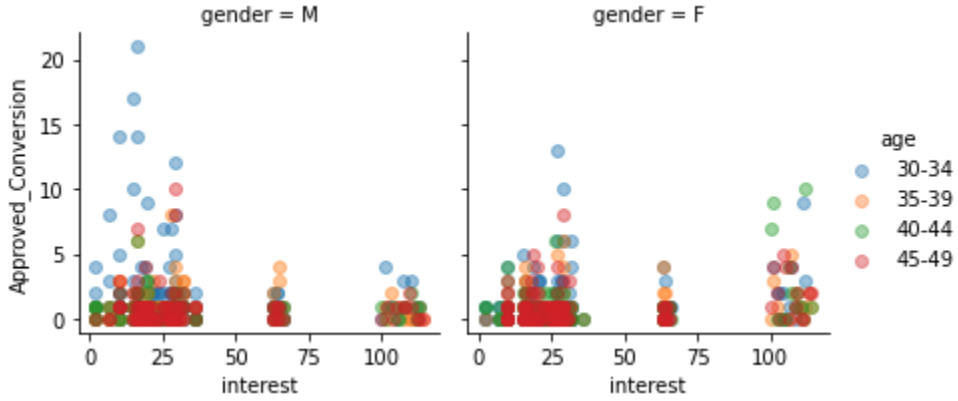
From the above graph, it is clear that:

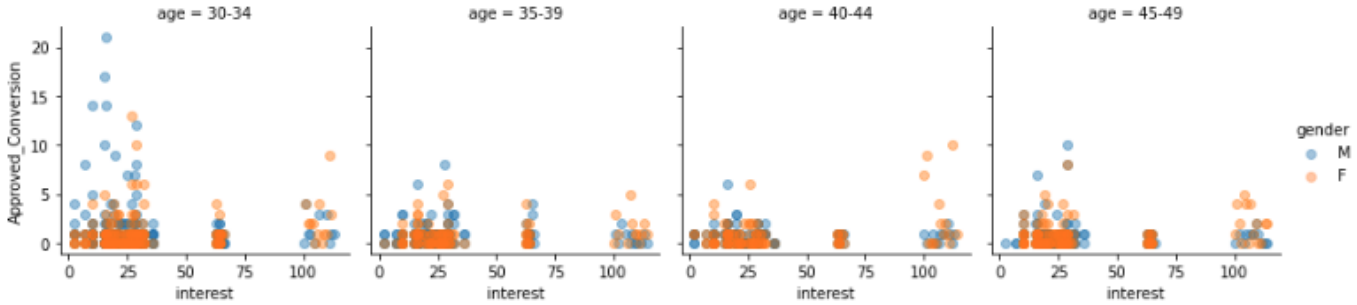
1. When the ads were shown more than 1,300,000 times, a total of 68 products were bought by the people.
2. A total of 64 products were sold when the Impressions were less than 1,300,000.

Hence, it looks like more impressions lead to higher conversions.

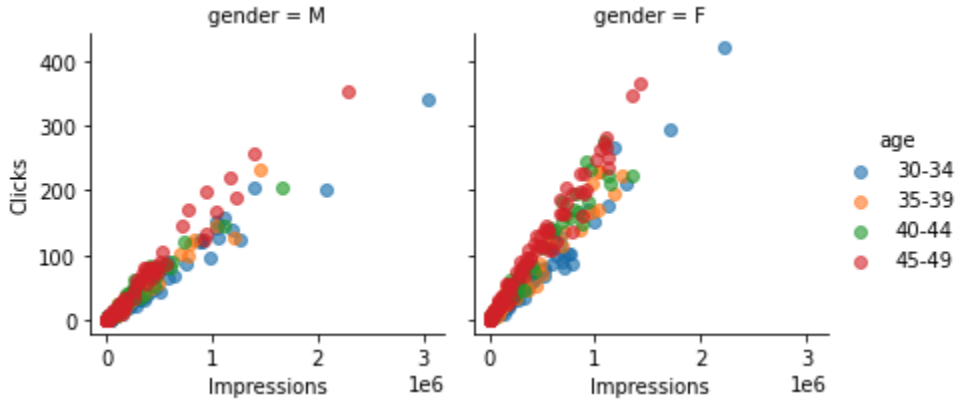
With respect to the categorical variables, the variations in the scatter plot for Approved Conversion, Interest, and Spent can be visualized as:

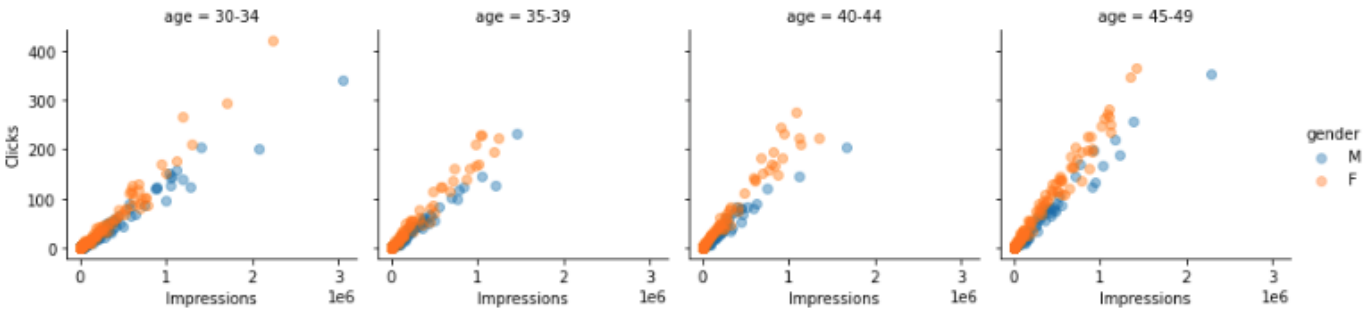
Approved Conversion vs Interest



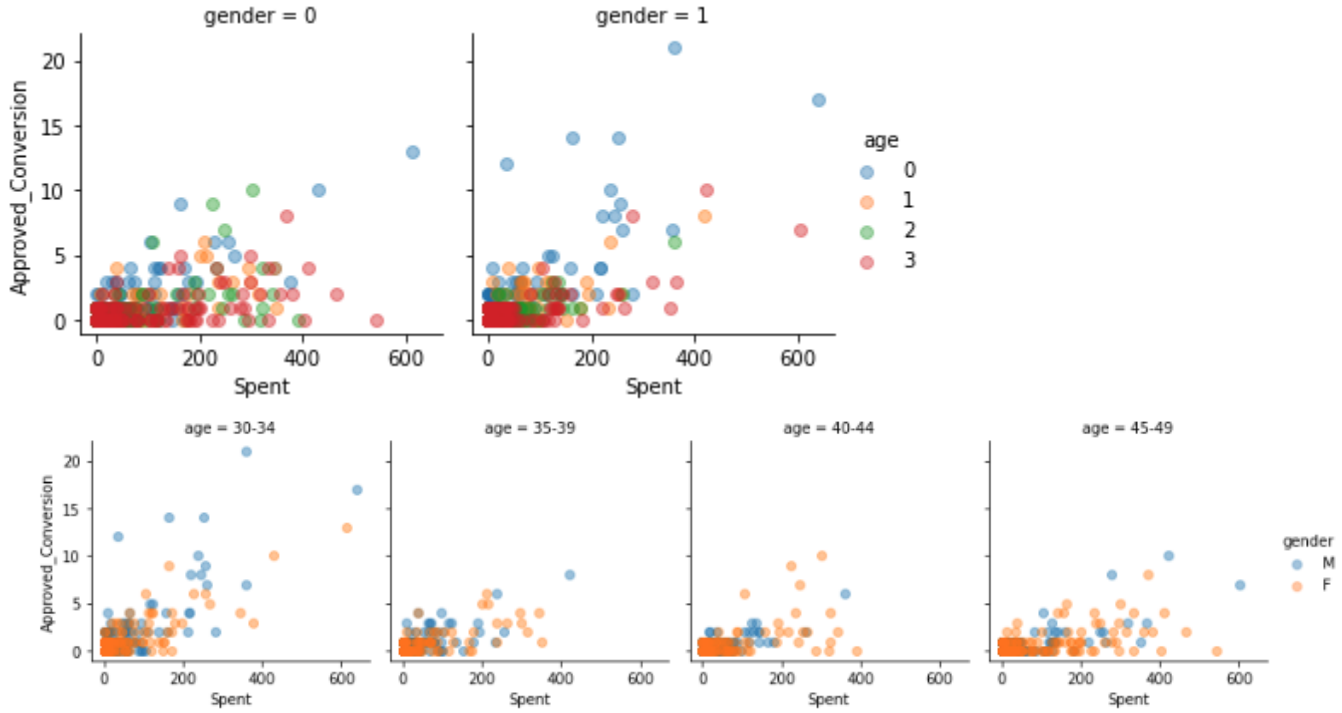


Clicks vs Impressions





Approved Conversion vs Spent



Data Modeling

Supervised Learning

In this project, we worked with labeled data. The actual conversions (and max_conversions) worked as labels for our data. This type of learning is called supervised learning. This type of learning helps organizations solve for a variety of real-world problems at scale.

Classification - Based on the target variable and selected features

We have used test_train_split() to split the input data into training data and test data. The training data is 80% of the overall dataset.

Logistic Regression

The first classification model that we used on this dataset is the Logistic Regression model, which is a model to understand the relationship between the dependent variable and one or more independent variables, and in turn help in predicting the class given values of other variables.

Decision Tree

To test the dataset with a tree-based model, we start with the Decision Tree model. This method aims to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. Decision Trees learners can often lead to very complex trees, resulting in overfitting. In this analysis, the decision tree based classification model proved to be more accurate than Logistic Regression. For our decision tree classifier, we used Gini impurity, and a max_depth of 3.

Random Forest

To test the dataset with an ensemble decision-tree model, we started with the Random Forest Model, which combines the output of multiple decision trees to reach a single result. The Random Forest model utilizes both bagging and feature randomness to create an uncorrelated forest of decision trees. Feature randomness generates a random subset of features, which ensures low correlation among decision trees. This is a key difference between decision trees and random forests. While decision trees consider all the possible feature splits, random forests only select a subset of those features. This way, Random Forest has lesser likelihood of over-fitting as compared to Decision Tree model as the averaging of uncorrelated trees in Random Forest lowers the overall variance and prediction error.

XGBoost

Further, to improve the performance of the decision-tree-based models, we use an incremental training model in the form of gradient-descent based boosting through the ensemble method - XGBoost. In this model, each tree boosts attributes that led to misclassification of the previous tree. This method is a regularized boosting method and prevents overfitting. Additionally, the performance of XGBoost is better than other tree ensemble methods (such as Gradient Boosting Machines) is generally better because of the Tree Pruning.

In our case, we have tried different values of max_depth, from 5 to 13, for tree-pruning. Additionally, we have tried suggested values between 5 and 11 for min_child_weight parameter, which is the minimum sum of instance weight needed in a child. To reduce overfitting, we have chosen a subsample, which would randomly sample half of the training data prior to growing trees, between (0.6 and 1). Similarly, we have chosen colsample_bytree, which is the subsample ratio of columns for each level, between 0.6 and 1.

We have chosen a gamma between 0.05 and 1. Gamma is the minimum loss reduction required to make a further partition on a leaf node of the tree. The tree is more conservative if the values

of gamma are high. After experimenting with learning rate, eta, ranging from values between 0.01 and 0.1, it is visible that the lower learning rate with higher number of iterations performs well.

Effectively, in this analysis, XGBoost, with a low learning-rate and high number of iterations, proved to be the most accurate learning model.

The best results that we got for XGBoost had the following parameters -

modelXGB = XGBClassifier(n_estimators=200,subsample= 0.8,colsample_bytree= 0.6,eta=0.01,gamma=0.01,max_depth=3,min_child_weight=0.1,random_state=42,eval_metric='mlogloss')

KMeans

Further we classify our dataset into groups based on impressions, clicks, and spent. Market segmentation is a strategy in marketing that aims at analyzing market data and extracting subsets of consumers that share common characteristics (needs, interests, behaviors, priorities…) Here in our project we have clustered the dataset based on consumer features like Male and Female, Age-group brackets with respect to the common characteristics like impressions, clicks and spent based on which consumers to target and how to target them. Identifying clusters of similar customers can help marketers develop a marketing strategy that addresses the needs of specific clusters.

In K-means clustering, to evaluate the model's performance, clustering analysis doesn't have a solid evaluation metric that we can use to evaluate the outcome of different clustering algorithms. Here we have calculated two metrics that may give us some intuition about *k*:

- Silhouette analysis
- Elbow method

We have done Silhouette analysis to determine the degree of separation between clusters. For each sample:

- Computing the average distance from all data points in the same cluster (ai).
- Computing the average distance from all data points in the closest cluster (bi).
- Computing the coefficient:

$$\frac{b^i - a^i}{max(a^i, b^i)}$$

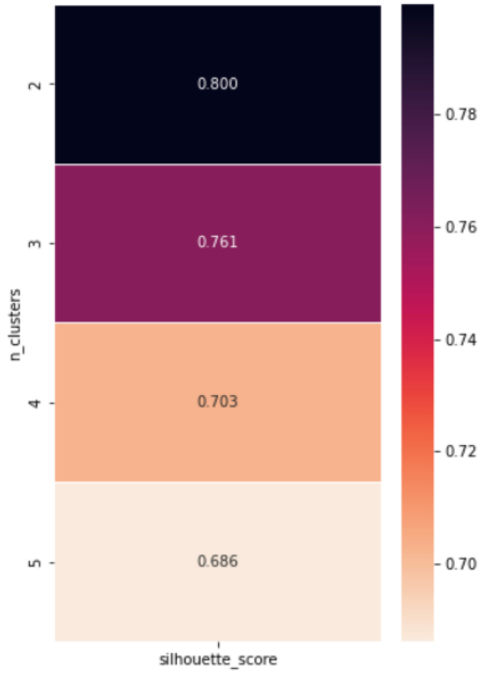
The coefficient can take values in the interval [-1, 1].

- If it is 0 –> the sample is very close to the neighboring clusters.
- It it is 1 –> the sample is far away from the neighboring clusters.
- It it is -1 –> the sample is assigned to the wrong clusters.

Therefore, we want the coefficients to be as big as possible and close to 1 to have a good cluster.

The Elbow method gives us an idea on what a good *k* number of clusters would be based on the sum of squared distance (SSE) between data points and their assigned clusters' centroids. We pick *k* at the spot where SSE starts to flatten out and form an elbow. Silhouette score and Elbow method.

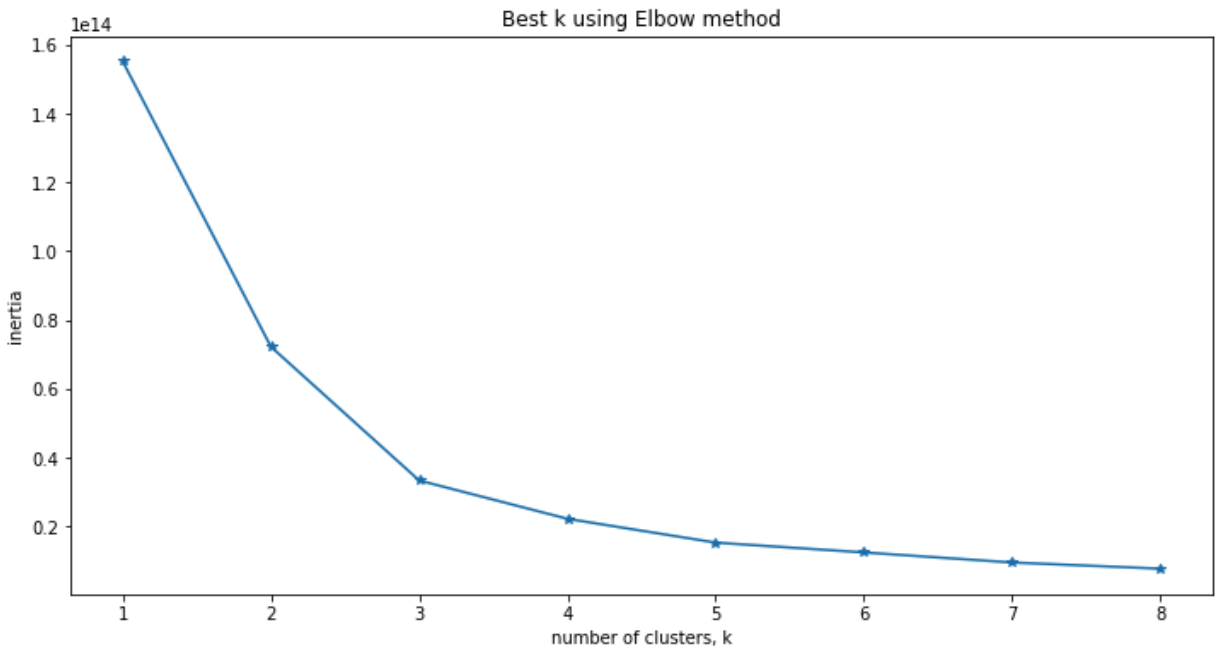
We check the Inertia Sum of squared distances of samples to their closest cluster center, weighted by the sample weights if provided.



Lesser the inertia better the result.

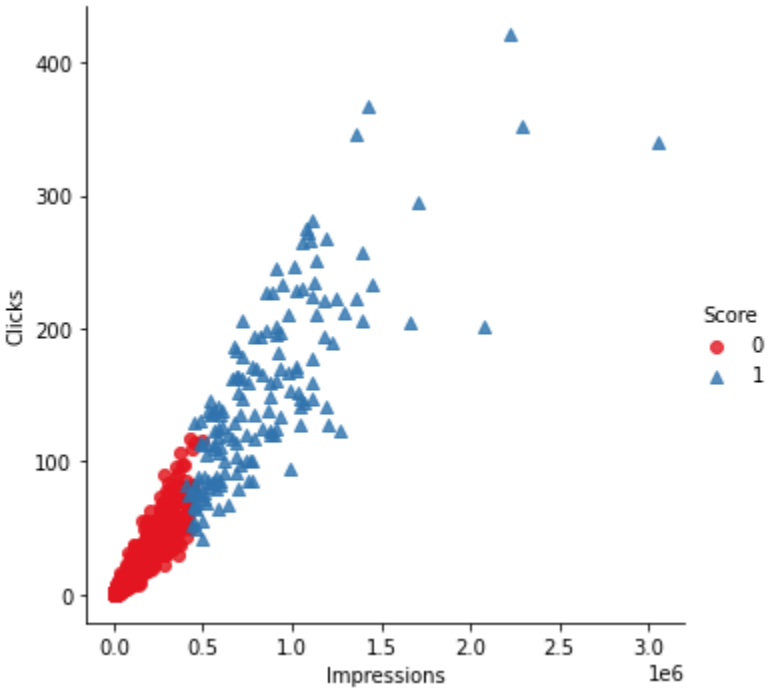
But, we can see the drop in inertia from k=1 to k=2 and k=2 to k=3 is much greater.

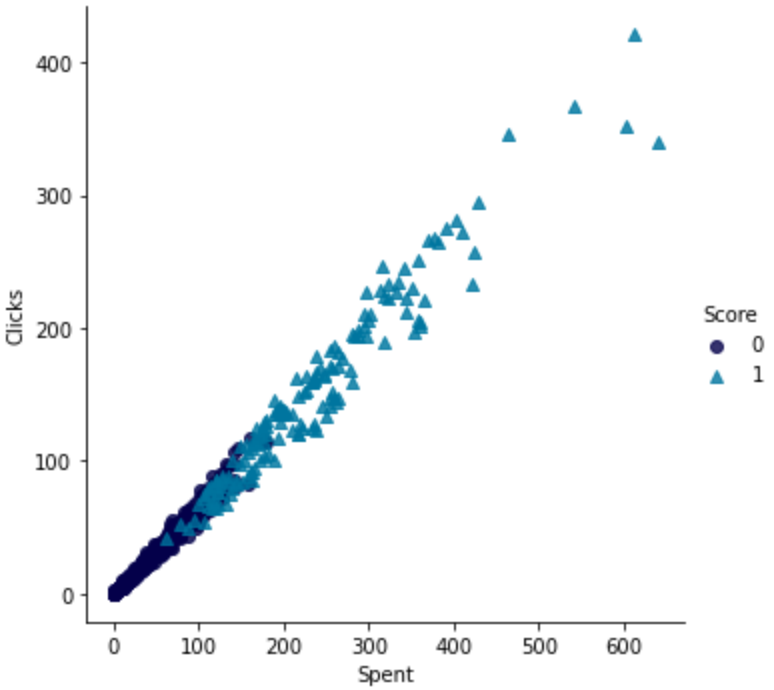
Though, inertia decreases with increase in k, but the rate is very low.

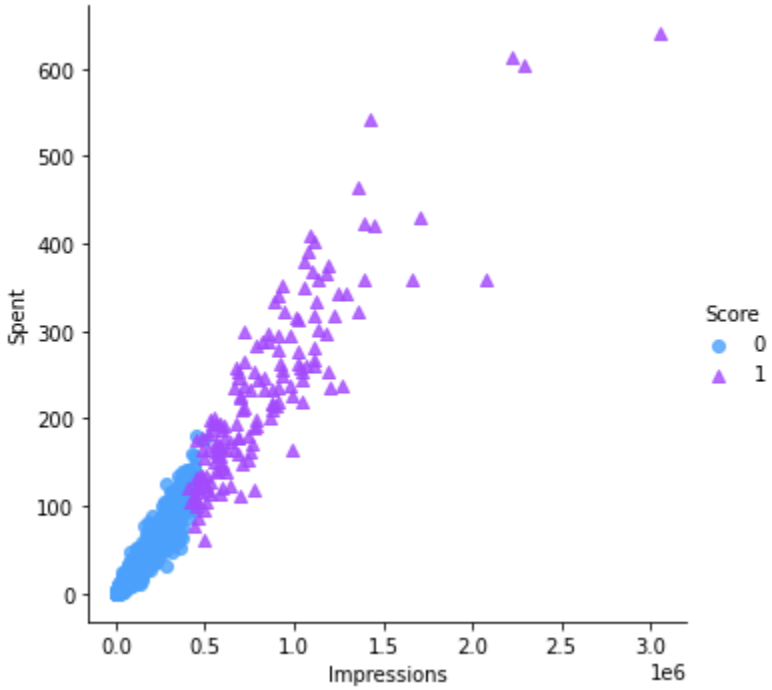


From the above elbow graph, we can see that for k=2, inertia drops with a high rate, so, we can choose k=2 or k=3 (to be on the safe side).

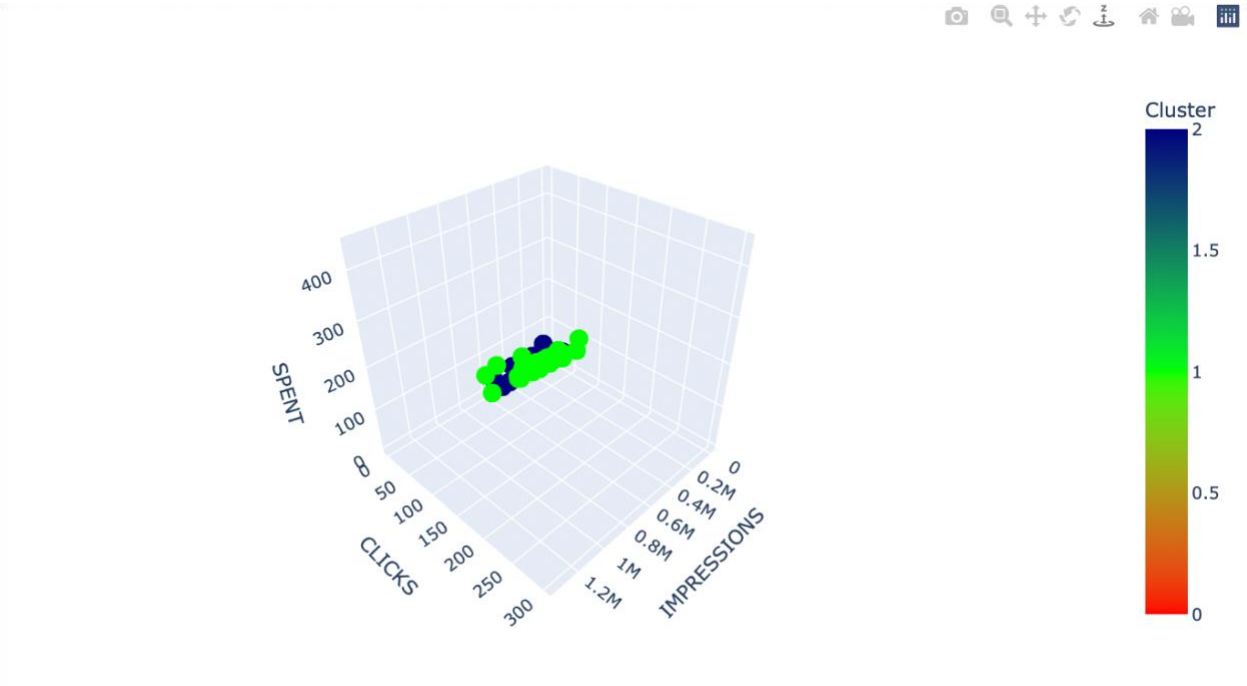
Our visualization of the result:




Clicks
Spent
Score
0
1
0
100
200
300
400
0
100
200
300
400
500
600


Spent
Impressions
1e6
Score
0
1
0
100
200
300
400
500
600
0.0
0.5
1.0
1.5
2.0
2.5
3.0

3D visualization of the clusters



Results and Evaluation:

Accuracy:

We have evaluated our models based on different error calculation metrics like Recall, Precision, F1score and accuracy scores. Out of all the classifiers used XGBoost Classifier performed the best with a an accuracy score of 0.825 followed by Decision Tree Classifier with a score of 0.821, which can be seen as:

	model	recall1	recall0	precision1	precision0	precision_sample	real_c_count	f1score	accuracyscore
4	XGBClassifier	0.882759	0.726190	0.847682	0.782051	151	145	0.825328	0.825
1	DecisionTreeClassifier	0.889655	0.702381	0.837662	0.786667	154	145	0.820961	0.821
0	LogisticRegression	0.910345	0.630952	0.809816	0.803030	163	145	0.807860	0.808
3	RandomForestClassifier	0.834483	0.702381	0.828767	0.710843	146	145	0.786026	0.786
2	LGBMClassifier	0.834483	0.666667	0.812081	0.700000	149	145	0.772926	0.773

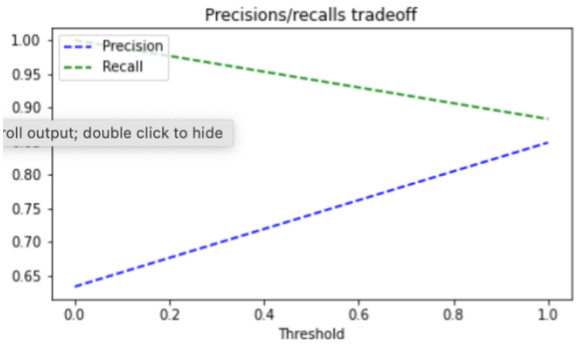
Confusion matrix:

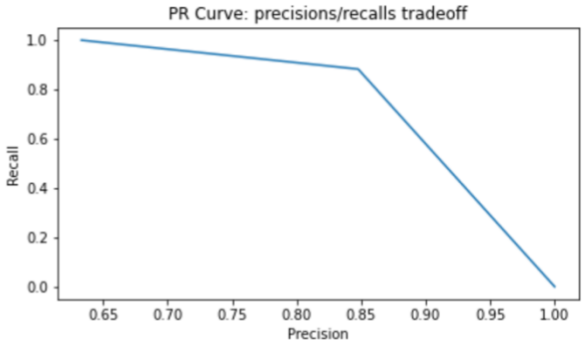
Computing confusion matrix to evaluate the accuracy of a classification.

```
Confusion matrix

 [[ 61  23]
 [ 17 128]]

True Positives(TP) =  61

True Negatives(TN) =  128

False Positives(FP) =  23

False Negatives(FN) =  17
```

Precision and Recall

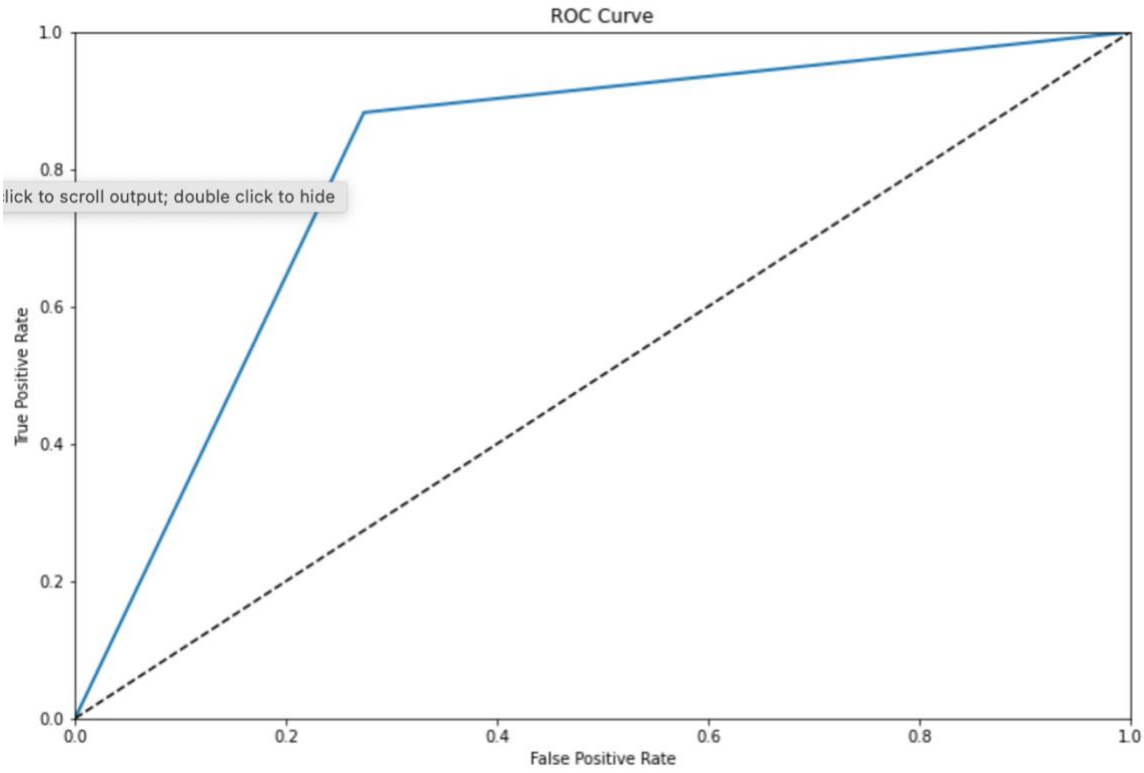
Precision is the fraction of relevant instances among the retrieved instances, while recall is the fraction of relevant instances that have been retrieved over the total amount of relevant instances. Precision and Recall are used as a measurement of the relevance. In our work we got:





ROC curve (Receiver Operating Characteristics)

ROC curve is used for visual comparison of classification models which shows the trade-off between the true positive rate and the false positive rate. The area under the ROC curve is a measure of the accuracy of the model. When a model is closer to the diagonal, it is less accurate and the model with perfect accuracy will have an area of 1.0.



Conclusion

This Data Analysis and Processing helped better understand which features to target based on their variations on age-group, gender, and money spent on various campaigns. Predictive Models like Xgboost, Lightgbm, Decision Tree , Logistic Regression and Random Forest Classifier were used and XGBoost performed relatively better. Additionally, clustering Algorithm K-means with 2 clusters based on high Silhouette score helped analyze and cluster Clicks, Impressions and Money spent. Better grouping the customers and targeting them based on optimization of these features can help strategize digital marketing campaigns for better revenue generation, through more informed targeting.

Effectively, using predictive analytics, marketers can gain a better knowlegde of which campaigns are working and what sorts of targeted advertising will lead to an increase in sales in future. This project could also be extended to planning around planning for advertisement costs, and approaches for targeting for prospective customers.

Future Work

Future work in this project would involve working with extended datasets, and introducing more complex concepts, such as Empirical Evaluation, Greedy Strategy, and “Reward Distribution” to the learning process. This would be on the lines of “Multi-armed Bandit Algorithms and Empirical

Evaluation" by Joannès Vermore, Mehryar Mohri. Additional work could be done to use Exploration and Exploitation for Maximum user conversion and target advertisement recommendation like the ideas from this paper. Additionally, merging the dataset based on customer profile and campaign datasets based on CPC(Cost per click) model for maximum revenue generation for ad bidding models could be another direction of future-work on this project.

Appendix

Running the Code

1. The most convenient way to run the code is through Jupyter Notebook.
2. Download the .ipynb file, and then proceed by running sequentially.
3. Additional libraries have been used to create plots, and other visualizations.

   These libraries include - seaborn, matplotlib, numpy, pandas, sklearn, itertools,

   Models from - sklearn, pandas profiling, libraries for xgboost - optuna, libomp, lightgbm

4. OS: MacOS Big Sur. (Hence, some of the files have been downloaded using Homebrew).