Marketing Email Campaign
&
Subscription Retention Rate Analysis

CMPE 256 - Project Report

SAN JOSÉ STATE UNIVERSITY

By,

TEAM 6

Poojitha Amin (011811306)
Charu Ramnani (012551422)
Preksha Pansheriya (012539293)
Prajwal Venkatesh (012557792)
Yogeswara Sarat Bankapalli (012431692)
Praneeth Varma Alluri (012406277)

Professor: Dr. Shih Yu Chang

Table of Contents

List of Figures

Project 1 - Marketing Email Campaign

1. Introduction

Marketing email campaign focuses on sending emails for promotional events and selling the items to the customer. Email is a good way of marketing upcoming new products to the consumer. Many e-commerce websites like Amazon and the retailing companies like Target tries to send emails to their registered customers/buyers about the upcoming products or the sale available or something that is back on stock.

Fig. 1. Email Marketing Components

In this project we focus on the data stating how many emails were sent to the existing users in accordance to how all took the pain to open the mail and going further, also clicked the link inside the email. Let us understand it with an example, when you are sent a promotional email from Amazon website, how often do you open the mail and click the link to redirect yourself to directed website. If the subject is of your interest, if the offer on the product is attractive enough, or it is the one product you were looking for, then you would be intrigued to open the mail and see what offer it has inside. Sometimes, we don't have that interest and don't find it aligned to our likes and requirements. In such cases we simply delete the mail. Understanding this problem, we would like to first understand with the existing data, who all open the mail and clicked the link and who don't like the same. Then we would like to share the recommendations for the user according to their likes and past history actions using machine learning techniques and recommendation systems.

1.1. Motivation

The motivation to do this project is to improve the company sales by marketing the products and offers through email. Email is fast, paper free service to do promotions. Also does not require door to door selling schemes or throwing advertisement papers on individual houses. With internet, it is easy to know, what the user is searching to buy and accordingly can be sent with personalized emails recommending their preferences and providing attractive deals.

Furthermore, the huge amount of search done by people every day on internet for online shopping gives us tremendous amount of data to study and accordingly build a code to train the machine to understand people's likes and choices and create unique email subjects and headings. Let us understand what algorithms we have applied to the found data set and our readings.

1.2. Objective

We have predefined objectives to work upon given by our profesor as following:

- What percentage of users opened the email and what percentage clicked on the link within the email?
- The VP of marketing thinks that it is stupid to send emails to a random subset and in a random way. Based on all the information you have about the emails that were sent, can you build a model to optimize in future email campaigns to maximize the probability of users clicking on the link inside the email?
- By how much do you think your model would improve click through rate (defined as # of users who click on the link / total users who received the email). How would you test that?
- Did you find any interesting pattern on how the email campaign performed for different segments of users? Explain.

Now understanding the goals:

- **Welcoming new subscribers** and telling them about your business and values so you start to build a relationship with them.
- **Boosting engagement** with your content and your business, whether that's promoting a webinar or trying to make an initial sale.
- **Nurturing existing subscribers** by providing something they'll value.
- **Re-engaging subscribers** who haven't been particularly active.

- **Segmenting your subscribers** so you can send more targeted email marketing campaigns.

1.3. Literature/Market Review

With around **269 billion emails sent each day**, there's a lot of competition for your audience's attention. Those email numbers keep going up, too. That's why it's essential to learn how to do email marketing right, so you can reach your target audience and keep their interest continued. With this project,we demonstrate the steps to run a successful email marketing campaign in order to get more attention, engagement, leads and sales.

In the literature, multiple approaches have been proposed to build effective models for email marketing. As part of a study done by Bawm, Z. L., & Nath, R. P. (2014), they introduced a conceptual model for an effective email marketing system clustering and segmenting subscribers based on their activity throughout a marketing campaign. The model consists of two main components: Collecting subscriber activity data and Clustering and Segmenting subscribers. According to Bhat, S. Y., & Abulaish, M. (2013), existence of overlapping communities of users in social networks can be used for viral marketing. Their paper aims to present the importance of identifying overlapping communities for the task of viral marketing in social networks and also provides some experimental results on an email network to back the claims.

In another study done by Li, Y., Murali, P., Shao, N., & Sheopuri, A. (2015), they discuss a key element in building predictive models is the ability to introduce features that capture historical user behaviors in an effective manner so as to differentiate between those consumers who are most likely to convert without nurturing, those who are likely to convert with nurturing, and those who are unlikely to convert irrespective of the marketing campaign and channel to which they may be subjected. The evaluation shows that the random oversampling approach has the best performance giving the largest area under the curve (AUC) and an up to 160% improvement in the lift index. However, none of these techniques work towards building a recommendation system based approach to build an email marketing and improve its related metrics, such as the click through rate rate. We aim to build an email marketing system, which combines the benefits of data mining machine learning techniques and recommendation systems.

2. System Design and Implementation

2.1. Algorithms Considered

1. **Logistic Regression (Predictive Learning Model) :** It is a statistical method for analysing a data set in which there are one or more independent variables that determine an outcome. The outcome is measured with a dichotomous variable (in which there are only two possible outcomes).

2. **Decision Trees:** Decision tree builds classification or regression models in the form of a tree structure. It breaks down a data set into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes.

3. **Random Forest:** Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees.

4. **Nearest Neighbor:** The k-nearest-neighbors algorithm is a classification algorithm, and it is supervised. To label a new point, it looks at the labelled points closest to that new point (those are its nearest neighbors), and has those neighbors vote, so whichever label the most of the neighbors have is the label for the new point.

5. **XGBoost Model:** XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible and portable. XGBoost provides a parallel tree boosting (also known as GBDT, GBM) that solve many data science problems in a fast and accurate way.

2.2. Technologies and Tools

- **Scikit-learn** - Image preprocessing, dimensionality reduction and classification.
- **Numpy** - Standard python library used with Scikit-Learn and OpenCV.
- **Pandas** - Easy to use data structure libraries and data analysis tools.
- **Matplotlib and Seaborn** - Used for data visualization choices for different plot styles
- **Pickle** - Store trained models which are later used to classify a given image.
- **Python 3.6** - Programming languages.

2.4. System Design and Data Flow

2.4.1. Machine Learning Module System Design

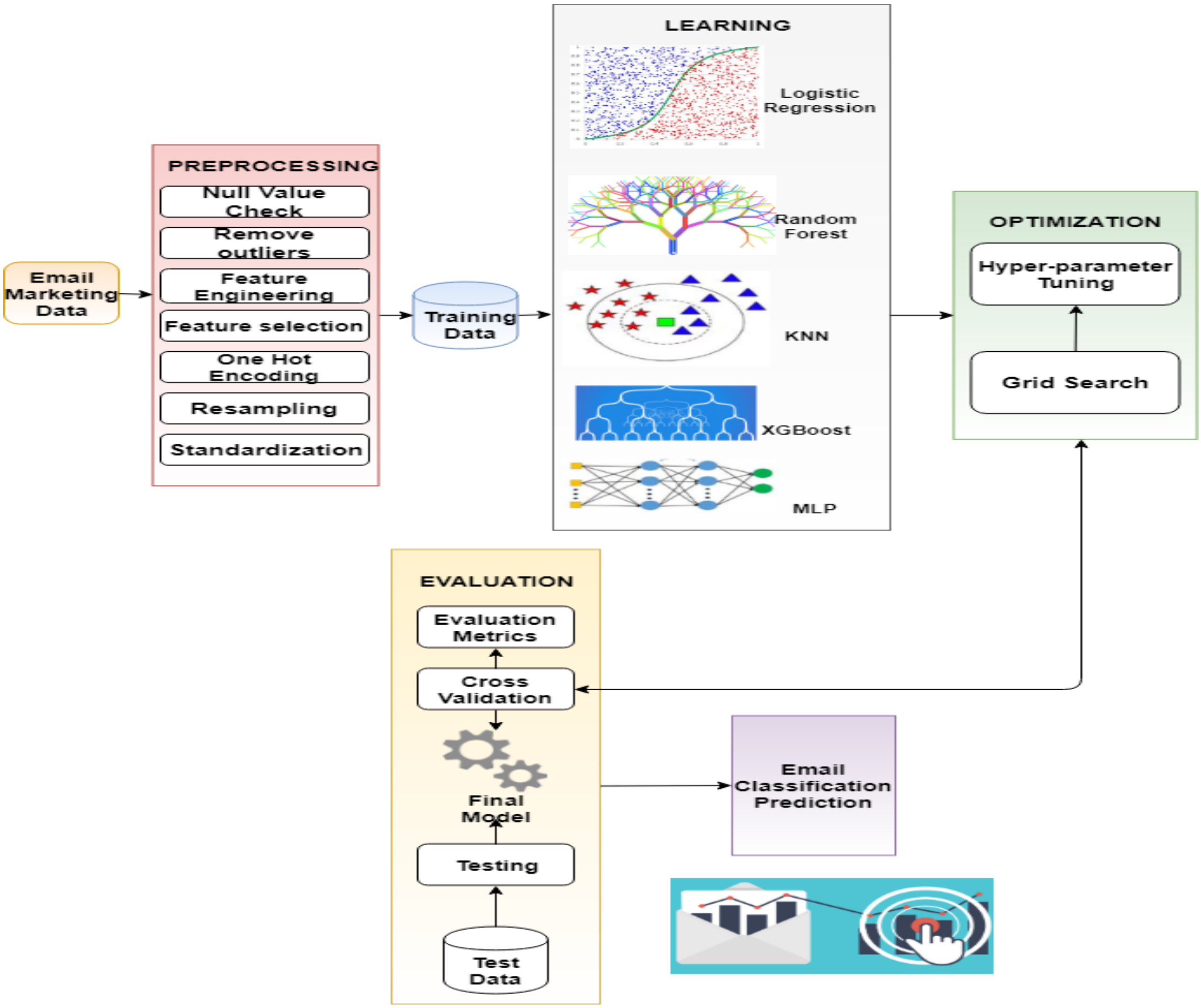


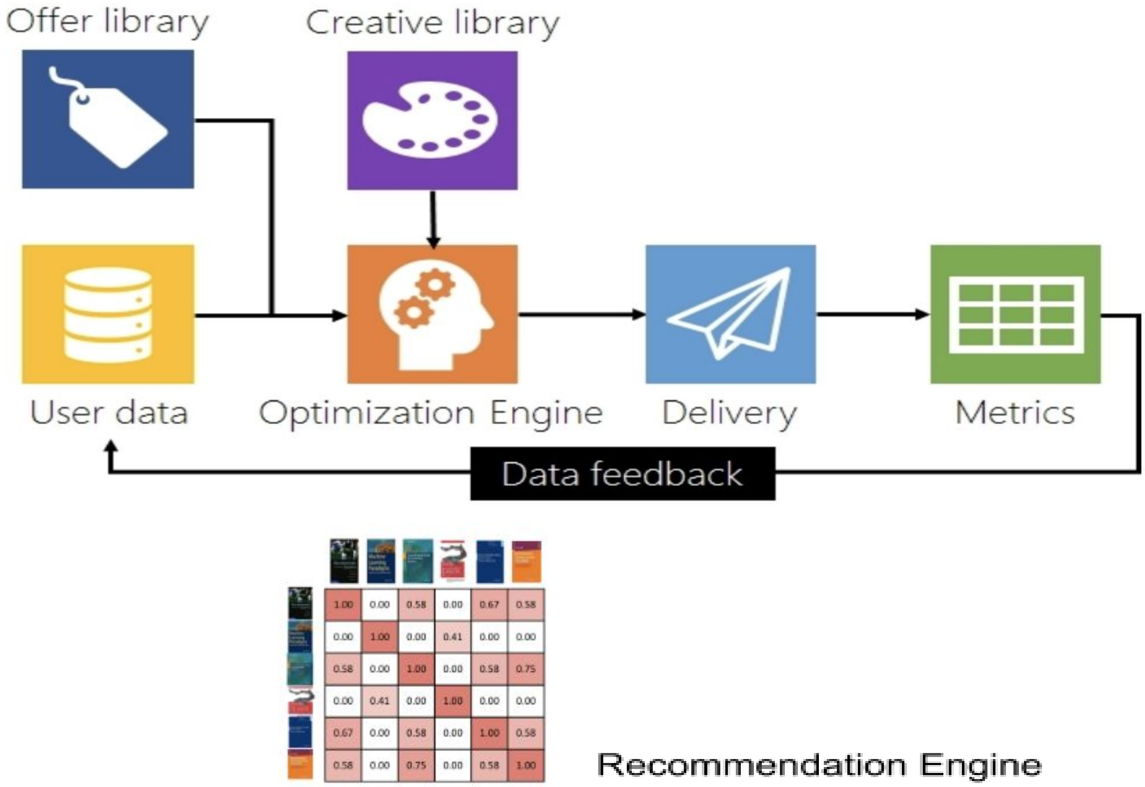
Fig. 2. Machine learning module data flow

2.4.2. Recommendation System Design

The goal of the recommendation system design is to incorporate recommendation element in the email campaign system. The Figure 2, demonstrates the architecture. The optimization engine takes the data from the following:

1) **Creative library** - includes the guidelines on the email content and subject, send time scheduled etc.

2) **Offer Library** - includes the offer related information for which the marketing has to be done.
3) **User Data** - includes the user specific information, such as the past ratings, his part purchases, his likes and dislikes etc.



1.00	0.00	0.58	0.00	0.67	0.58
0.00	1.00	0.00	0.41	0.00	0.00
0.58	0.00	1.00	0.00	0.58	0.75
0.00	0.41	0.00	1.00	0.00	0.00
0.67	0.00	0.58	0.00	1.00	0.58
0.58	0.00	0.75	0.00	0.58	1.00

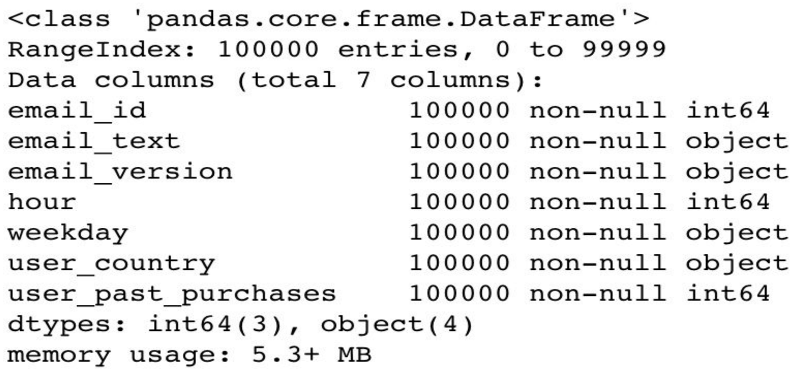
Fig. 3. Recommendation System Architecture

The recommendation engine component provides the additional support to the whole marketing email process by giving recommendations on email subjects and contents. We will be elaborating on the recommendation engine implementation in the next subsections. Once the email has been delivered, the metrics collected are sent back to the email campaign system. It gathers information on how many users opened the email and how many users clicked on the link inside the email and how many users ignored the email. Based on the feedback, new strategies will be used or existing strategies can be enhanced in order to further improve the click-through rates. In this framework, the scope of the current project includes designing and building the recommendation engine component.

2.5. Data Preprocessing Steps

- **Null value and duplicate check**

  Cleaning missing data is a process of dealing with incomplete data instances. The original dataset provided as part of the project proposal did not have any missing values, and hence, did not have to be cleaned. The Figure 3, we can see that there are 100000 records and none of the columns have missing values.



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100000 entries, 0 to 99999
Data columns (total 7 columns):
email_id                 100000 non-null int64
email_text               100000 non-null object
email_version            100000 non-null object
hour                     100000 non-null int64
weekday                  100000 non-null object
user_country             100000 non-null object
user_past_purchases      100000 non-null int64
dtypes: int64(3), object(4)
memory usage: 5.3+ MB
```

Fig. 4. Null value counts

- **Remove outliers**

  To avoid skewing and misleading of the training process, which results in poor results and not so accurate models, because of the outliers. The below Figure 3, shows points between 15 and 20, these are the outliers which are not near the quartiles.

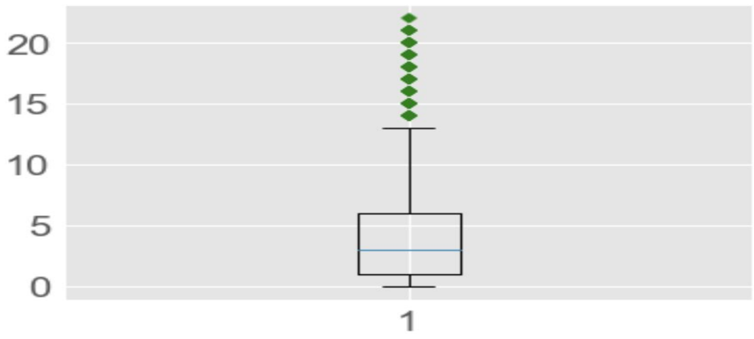
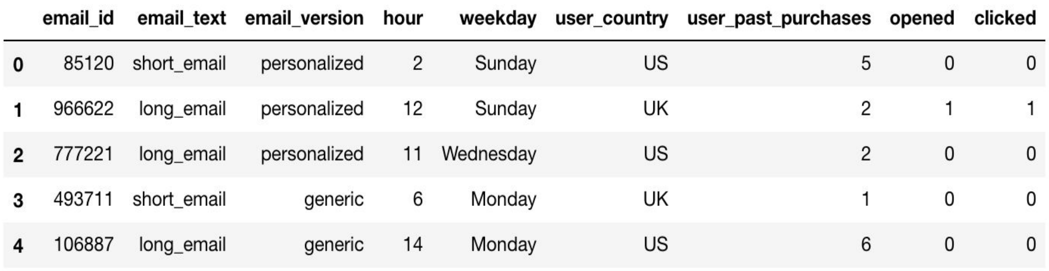


Fig. 5. Outlier detection using boxplot

- **Feature engineering by combining tables**

  We initially had three tables - email_table, email_opened_table and link_clicked_table. In order to proceed with our analysis, we had to combine the three tables. This was done by joining the three tables and the addition of two columns - opened, clicked.

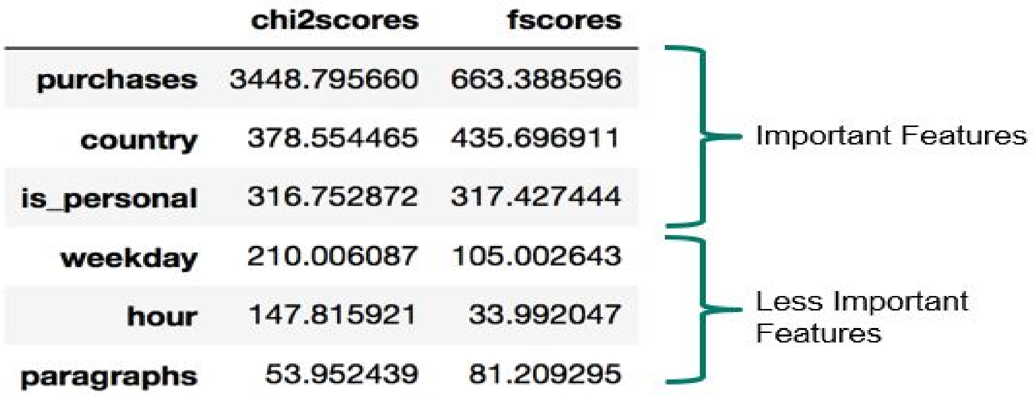


	email_id	email_text	email_version	hour	weekday	user_country	user_past_purchases	opened	clicked
0	85120	short_email	personalized	2	Sunday	US	5	0	0
1	966622	long_email	personalized	12	Sunday	UK	2	1	1
2	777221	long_email	personalized	11	Wednesday	US	2	0	0
3	493711	short_email	generic	6	Monday	UK	1	0	0
4	106887	long_email	generic	14	Monday	US	6	0	0

Fig. 6. Merged Features

- **Feature selection**

  The Chi Square statistic is commonly used for testing relationships between categorical variables. The null hypothesis of the Chi-Square test is that no relationship exists on the categorical variables in the population; they are independent. Chi square test for testing goodness of fit is used to decide whether there is any difference between the observed (experimental) value and the expected (theoretical) value.



	chi2scores	fscores
purchases	3448.795660	663.388596
country	378.554465	435.696911
is_personal	316.752872	317.427444
weekday	210.006087	105.002643
hour	147.815921	33.992047
paragraphs	53.952439	81.209295

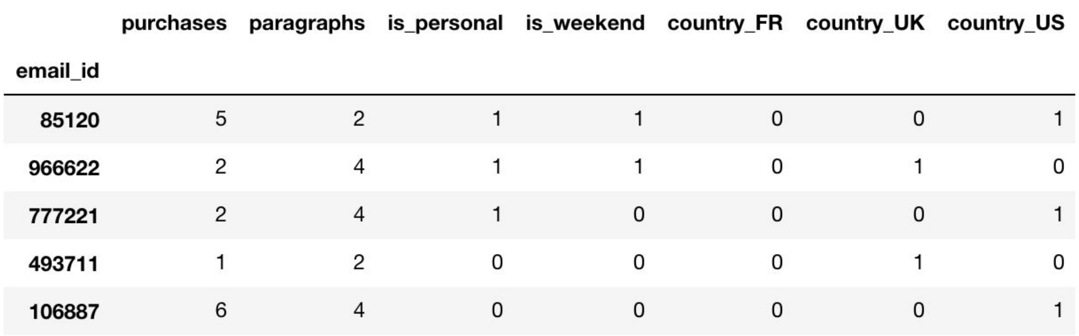
Important Features

Less Important Features

Fig. 7. Feature selection scores

- **One hot encoding**

  We converted the categorical variables to numeric representations. This is done to remove any kind of ordinal relationship in categorical data and allow the application of all kinds of machine learning algorithm on our dataset. This is implemented using the following methods:
  - LabelEncoder
  - OneHotEncoder



	purchases	paragraphs	is_personal	is_weekend	country_FR	country_UK	country_US
email_id							
85120	5	2	1	1	0	0	1
966622	2	4	1	1	0	1	0
777221	2	4	1	0	0	0	1
493711	1	2	0	0	0	1	0
106887	6	4	0	0	0	0	1

Fig. 8. One hot encoding for country column

- **Resampling to handle imbalanced dataset**

  We have an imbalanced dataset at hand with the count of emails with link being clicked is much less than the count of emails that are actually sent as seen in Figure 7. A classifier is generally biased, with it being more sensitive in detecting majority class and less sensitive in detecting minority class. To overcome this we explored the following techniques of sampling.
  - Oversampling
  - Undersampling
  - SMOTE

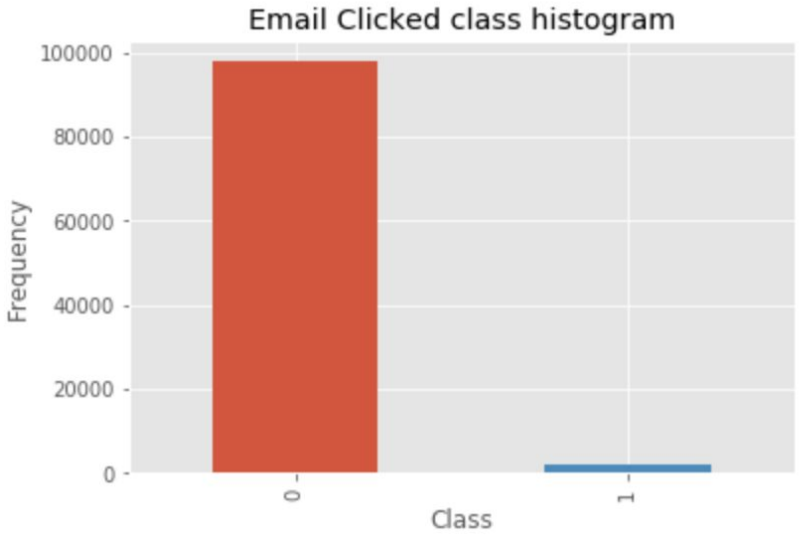


Fig. 9. Class distribution

- **Standardization to scale the values**

  To handle the parameters of varying units and scales, we standardized the dataset. This is done to enable fair comparison between them. Standardization will transform it to have zero mean and unit variance.

$x_{new} = \frac{x - \mu}{\sigma}$

where,

- μ is the mean of the vector
- σ is the standard deviation of the vector

3. Experiments

3.1. Dataset

In our project, we have considered 3 datasets as given below:
Dataset 1: email_table (100000 records), email_opened_table (10345 records), link_clicked_table (2119 records).
Features: email_id, email_text, email_version, hour, weekday, user_country, past_purchases.
Dataset 2: SpamClickBait Dataset (3,089,781 records)
Features: headline_text: Text of the headline in English with rare utf8 chars (<1k)
Dataset 3: Enron dataset, web ads dataset, Google adword

3.2. Exploratory Data Analysis (Graphs and Plots)

3.2.1. Email Text Visualization

The graph shown below represents the count plot of short and long emails.It is observed that users prefer to open short emails compared to longer ones as it is easy to read and less time consuming.Hence,click through rate for short emails is comparatively more.

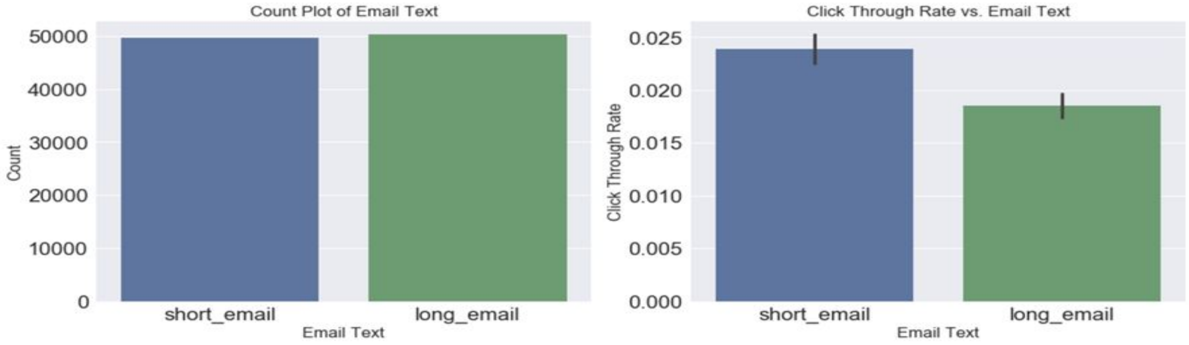


Fig. 10. Email text vs. Click through rate

3.2.2. Email Version Visualization

In the given dataset, two versions of emails are available. They are personalized and generic respectively. Usually people are interested to check the emails when it has some personal information regarding their areas of interest. Generic emails are mostly ignored. We can notice from the graph that the click through rate for personalized email is double compared to generic ones.

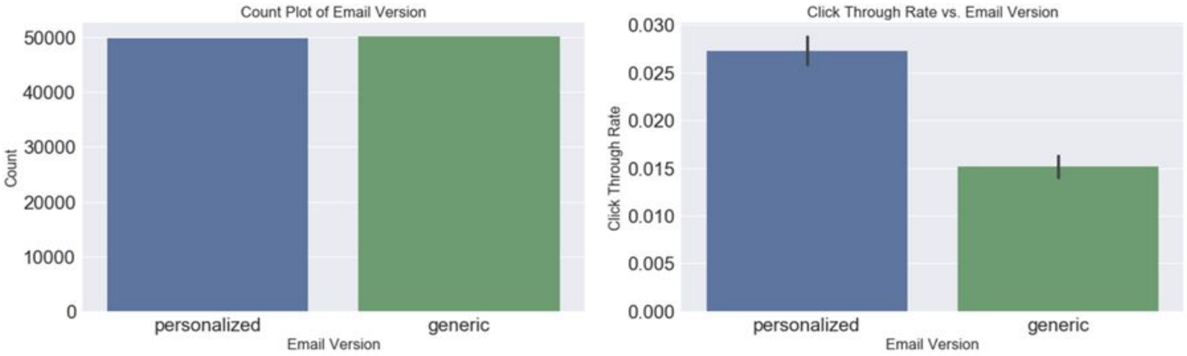


Fig. 11. Email version visualization vs. Click through rate

3.2.3. Sent Hour Visualization

The graph below depicts the number of emails sent in 24 hours of day. It is clear from the count plot that more emails are sent during the daytime. Users prefer to open online shopping related emails around midnight, hence click through rate is higher at night.

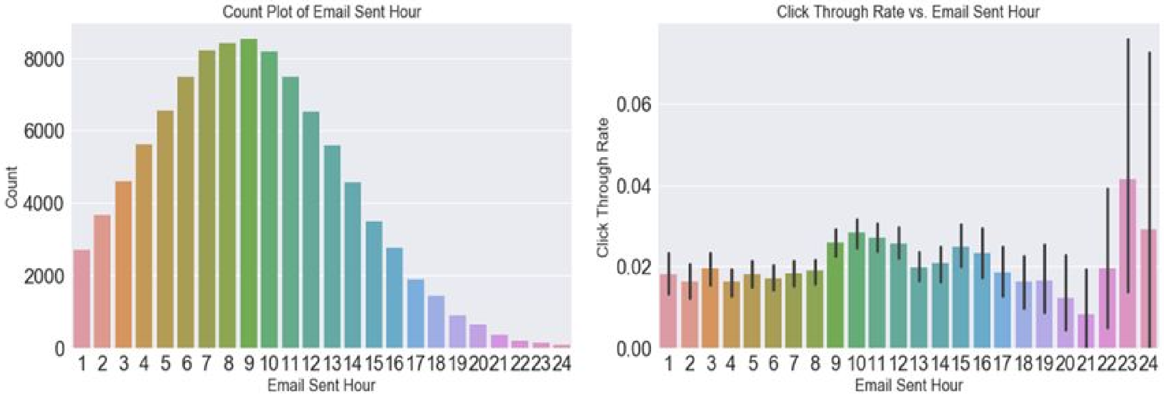


Fig. 12. Email sent hour vs. Click through rate

3.2.4. Weekday Visualization

E-Commerce websites tend to send same number of emails throughout the week to promote new products launched in the market and make the customers aware about different offers and sale going online. It is noted that customers prefer to visit and buy online products during mid of the week compared to other days.

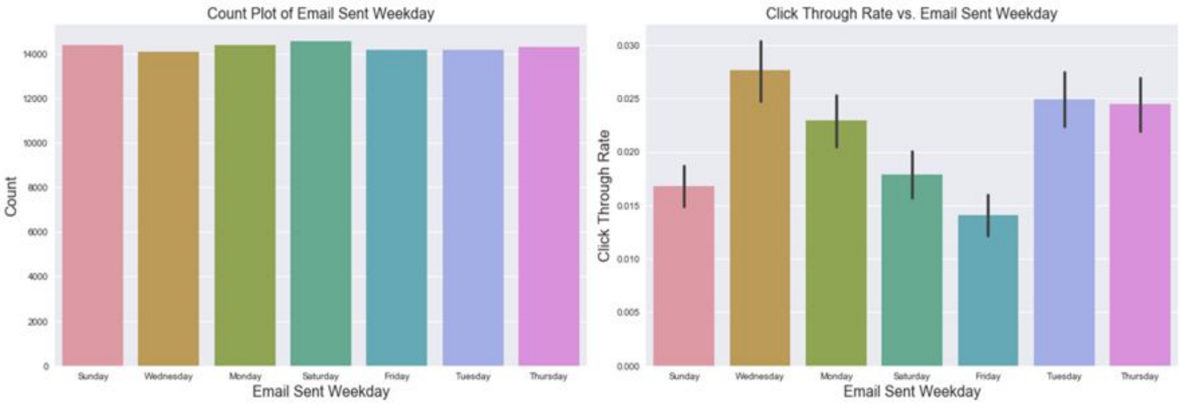


Fig. 13. Email sent weekday vs. Click through rate

3.2.5. User Country Visualization

90% population in United States favor online shopping. It is depicted from the graph that the click through rate of online promotional offers mails is towering in countries like UK and US.

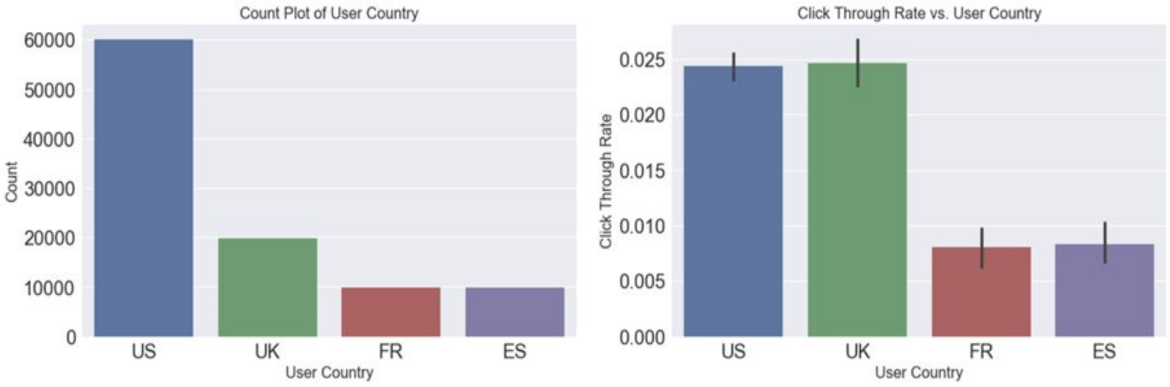


Fig. 14. User country vs. Click through rate

3.2.6. Past Purchases Visualization

The visualization plotted depicts the emails sent to new vs. old users. Click through rate shows that the existing users check the advancing offers to the maximum while the new users do not refer to these emails as much.

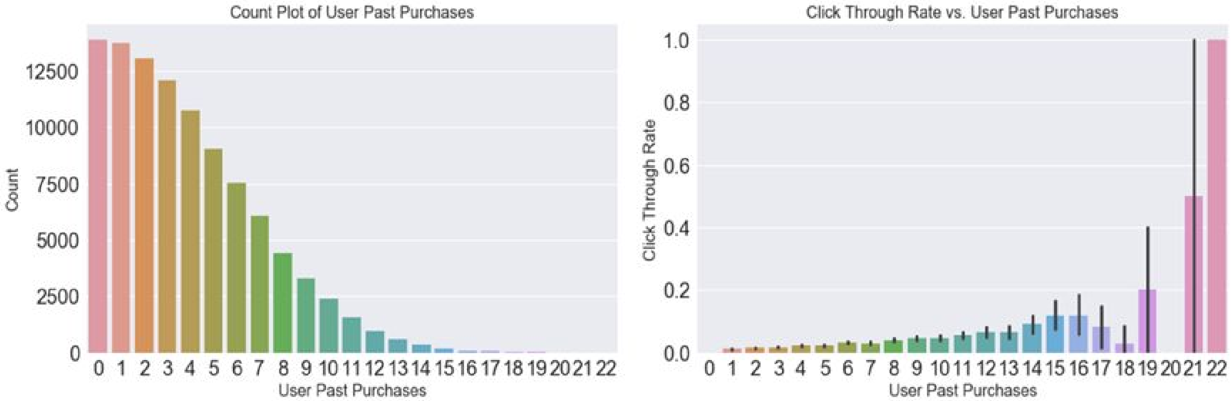


Fig. 15. Past purchases vs. Click through rate

3.3. Methodology

3.3.1. XGBoost Model

XGBoost stands for eXtreme Gradient Boosting. It is a highly flexible and versatile tool that can work through most regression, classification and ranking problems as well as user-built objective functions. The implementation of XGBoost offers several advanced features for model tuning, computing environments and algorithm enhancement. It is capable of performing the three main forms of gradient boosting (Gradient Boosting (GB), Stochastic GB and Regularized GB) and it is robust enough to support fine tuning and addition of regularization parameters.

The two reasons to use XGBoost are also the two goals of the project:

1. **Execution Speed.**
2. **Model Performance**

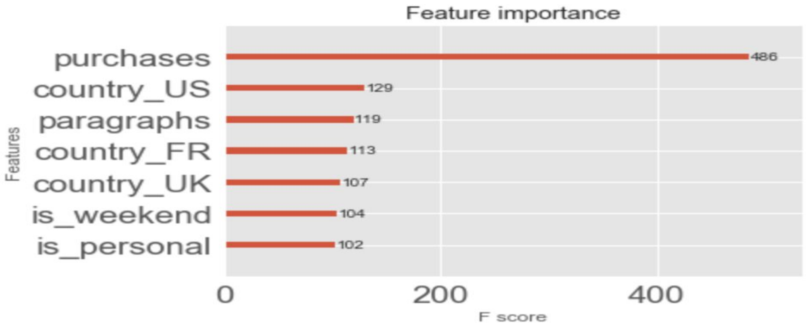


Fig. 16. XGBoost feature importance plot

It is used in our project for tuning hyper parameters in the dataset. The above figure shows the bar chart of the feature importance on our predictive modeling problem. Also the threshold value has been adjusted to improve the accuracy of the model.

3.3.2. Random Forest Model

It is supervised learning algorithm.It builds multiple decision trees and merges them together to get more accurate and stable prediction.Each decision tree in the forest considers a random subset of features when forming questions and only has access to a random set of the training data points.While making prediction,this algorithm takes an average of all individual decision trees estimates.

The advantage of this model is that it can be used for both classification and regression problems.In our project,we will be using this model to solve classification problem.The following tasks are performed using the given model:-

- Basic exploratory analysis
- Grid search for tuning the model
- Hyperparameter optimization

Thus,using Random Forest Model we have considered random subset of features for splitting the root node and making predictions accordingly.

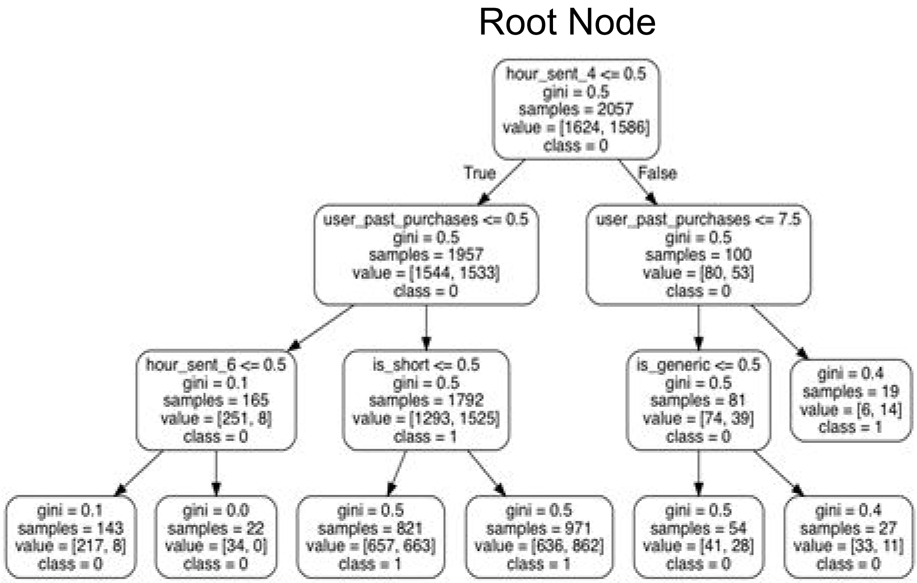

Root Node
hour_sent_4 <= 0.5 gini = 0.5 samples = 2057 value = [1624, 1586] class = 0
True
False
user_past_purchases <= 0.5 gini = 0.5 samples = 1957 value = [1544, 1533] class = 0
user_past_purchases <= 7.5 gini = 0.5 samples = 100 value = [80, 53] class = 0
hour_sent_6 <= 0.5 gini = 0.1 samples = 165 value = [251, 8] class = 0
is_short <= 0.5 gini = 0.5 samples = 1792 value = [1293, 1525] class = 1
is_generic <= 0.5 gini = 0.5 samples = 81 value = [74, 39] class = 0
gini = 0.4 samples = 19 value = [6, 14] class = 1
gini = 0.1 samples = 143 value = [217, 8] class = 0
gini = 0.0 samples = 22 value = [34, 0] class = 0
gini = 0.5 samples = 821 value = [657, 663] class = 1
gini = 0.5 samples = 971 value = [636, 862] class = 1
gini = 0.5 samples = 54 value = [41, 28] class = 0
gini = 0.4 samples = 27 value = [33, 11] class = 0

Fig. 17. Sample tree representation of random forest model

3.4. Model Evaluation

We evaluated our model using the following metrics.

3.4.1. Accuracy

Accuracy is measured as,

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

which can also be written as,

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

However, as we have an imbalanced dataset in hand. Accuracy is not a good metric to measure the performance of the model.

3.4.2. ROC curve

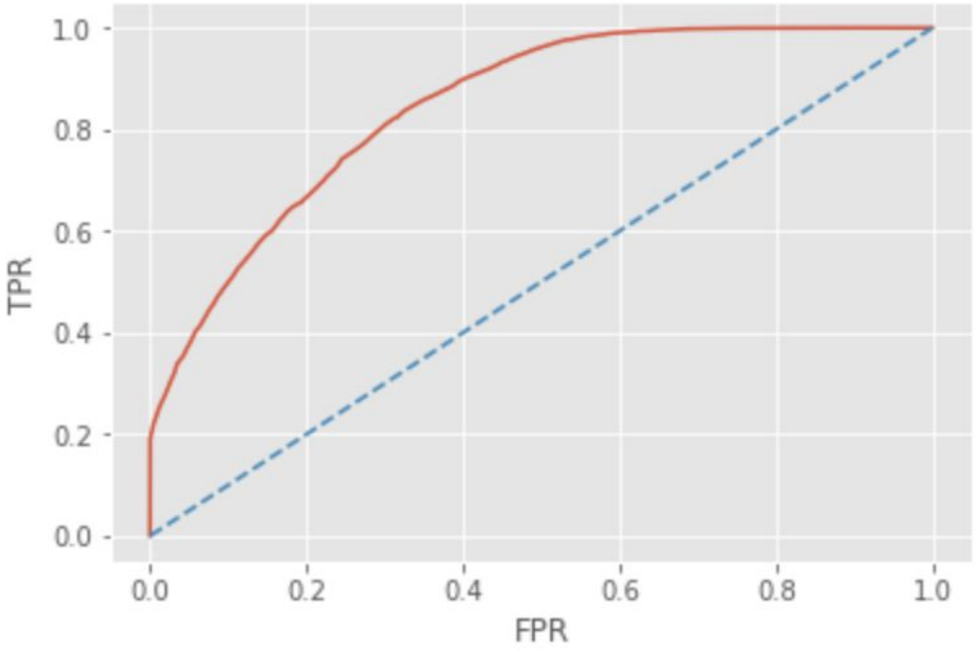


Fig. 18. ROC model evaluation

3.4.3. Confusion matrix

Our result from the xgboost model in the confusion matrix form is as below-

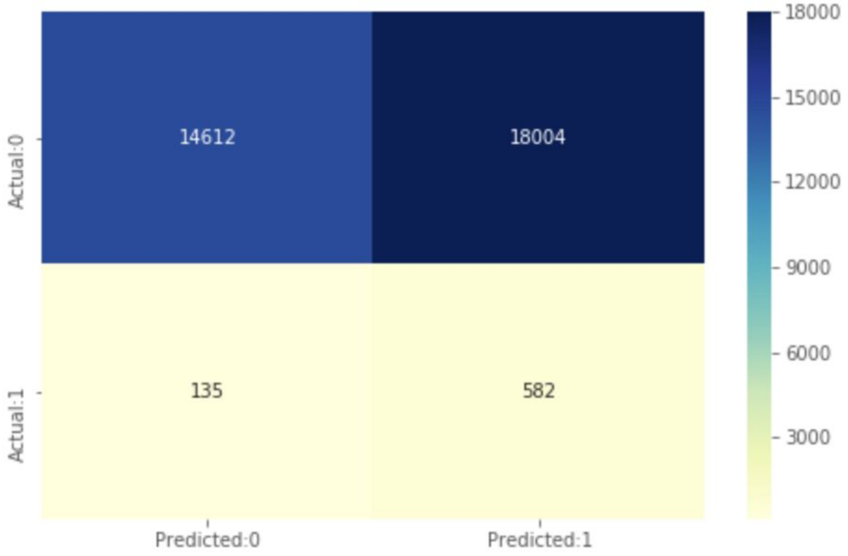


Fig. 19.XGBoost confusion matrix

3.4.4. Sensitivity (Recall) and Specificity

One of the best metrics to maximize in an imbalanced dataset scenario is Recall, also known as sensitivity of the model.

Sensitivity $= \frac{TP}{TP+FN}$

$$\Rightarrow \frac{Positive\ Email\ click\ \ correctly\ identified}{Positive\ email\ click\ \ correctly\ identified * Positive\ email\ click\ incorrectly\ labeled\ as\ negative\ email\ click}$$

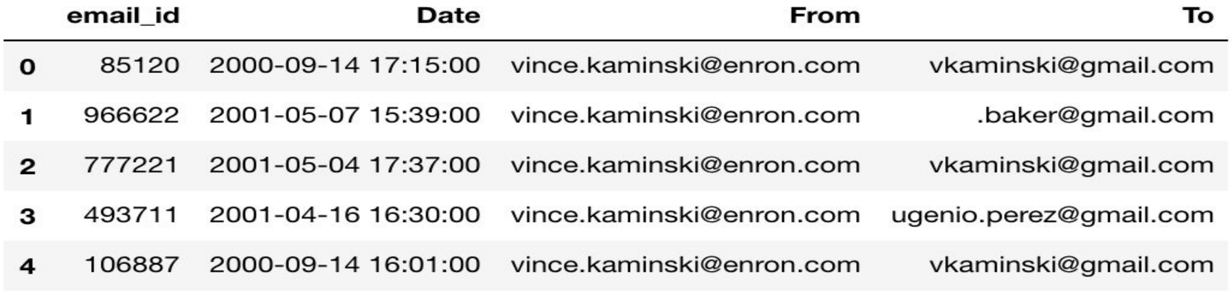
$$\Rightarrow \frac{582}{582 + 135} = 81\%$$

3.4.5. K-folds Cross Validation

We performed 10-fold cross validation to completely test our model. We trained the model with a subset of 9 folds and tested the model with the remaining subset of data. This allowed us in measuring the actual accuracy of the model, covering all possible combinations of datasets. We achieved mean recall value close to what is listed above.

3.4. Email Marketing Network Analysis

After we merged the Dataset 1 and Dataset 3, the relevant columns for email marketing network analysis are as shown below. ‘From’ column represents the email marketing agents email address, while ‘To’ column represents the customer’s email addresses who received the email.



	email_id	Date	From	To
0	85120	2000-09-14 17:15:00	vince.kaminski@enron.com	vkaminski@gmail.com
1	966622	2001-05-07 15:39:00	vince.kaminski@enron.com	.baker@gmail.com
2	777221	2001-05-04 17:37:00	vince.kaminski@enron.com	vkaminski@gmail.com
3	493711	2001-04-16 16:30:00	vince.kaminski@enron.com	ugenio.perez@gmail.com
4	106887	2000-09-14 16:01:00	vince.kaminski@enron.com	vkaminski@gmail.com

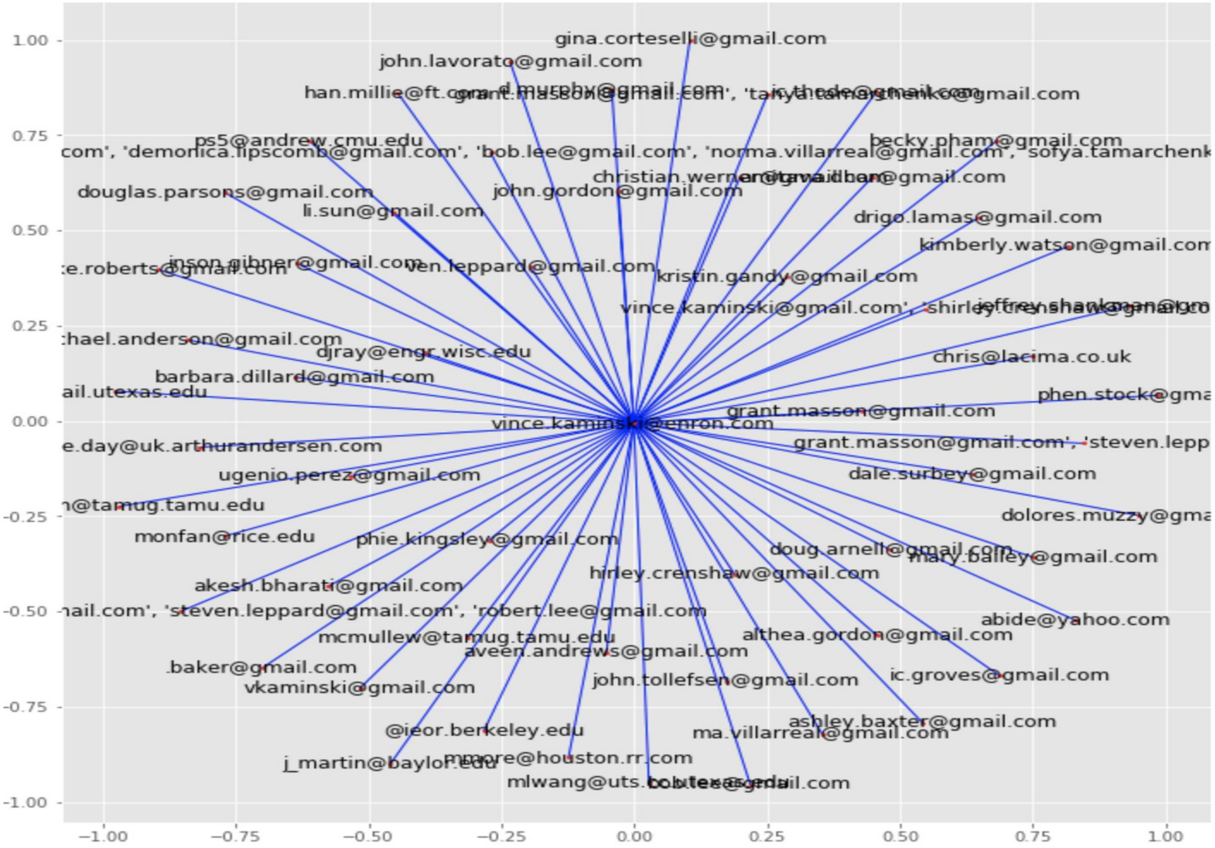


Fig.20. Marketing Email Network Plot

Network Plot:

From our network plot in Figure 16, we arrived at the following conclusions:

- Shows dominance of one node.
- One sales executive, Mr. Vince Kaminski, is connected to most customers,

providing highest click through rate in the promotion emails sent by him.

- Can be useful in visually understanding the
number of neighbours and check if a connection exists between an executive and a customer.
- Analyse distribution of message spread time if a referral program is introduced.

Degree Centrality:

Apart from the network plot, we also plotted the **Degree Centrality** of the email network and the plot is as seen the figure below:

- The degree centrality for a node v is the fraction of nodes it is connected to.
- Customer with Email ID klay@gmail.com is most connected to our executives.
- Vince Kaminski is the most actively connected company executive.

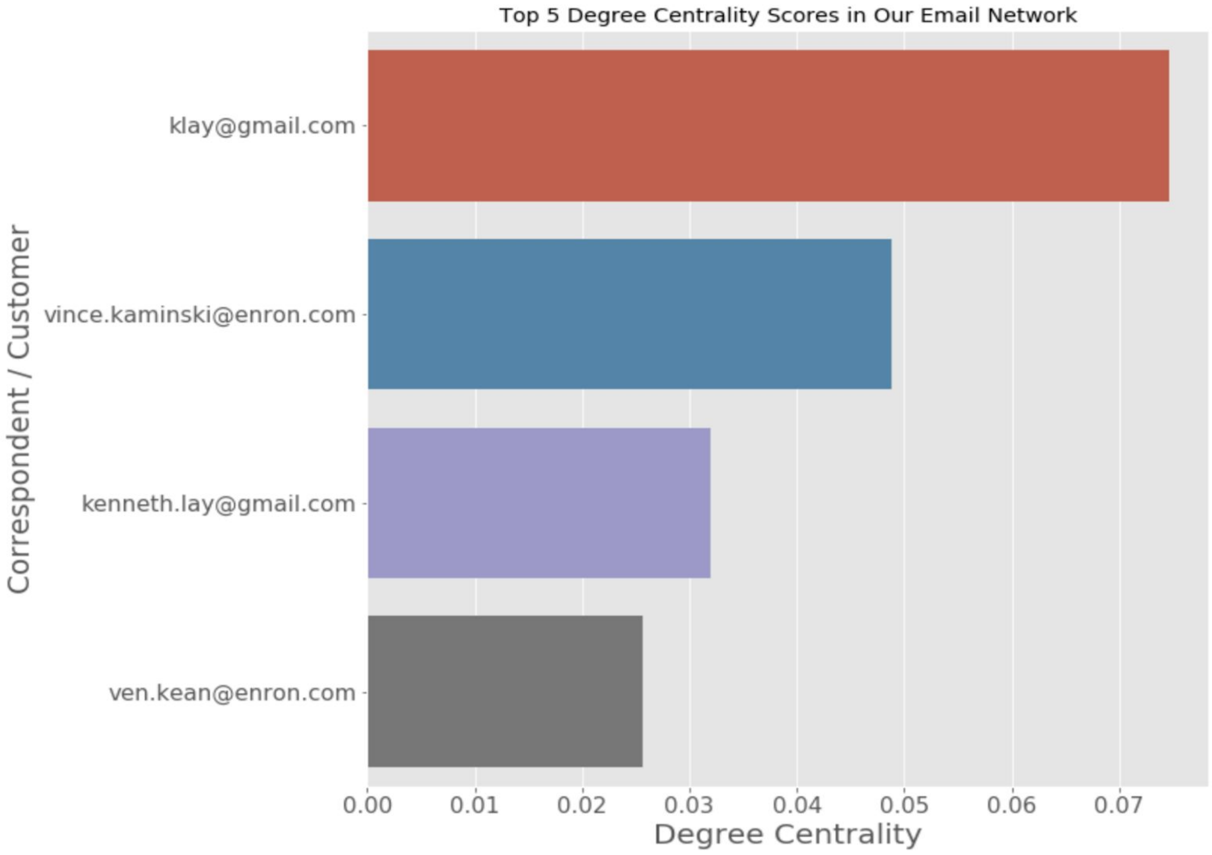


Fig.21. Top 5 degree centralities for email network

3.5. Email Marketing Recommendation System Implementation

3.5.1. Recommendation System 1 - TF-TDF Content Based recommendation system

We implemented a Content Based recommender system using the concept of TF-IDF. This mechanism can be used to suggest the most similar email subjects.

- Steps:
  - Define a **TF-IDF** Vectorizer Object.
  - Removed stop words from email subjects.
  - Compute the cosine similarity matrix.
  - Get pairwise similarity scores of all previous email subjects with the new campaign email to be broadcast.
  - Return the top 5 most similar email subjects sent in the past.

With this mechanism, before starting the new campaign, the business can look at their history of emails sent, get the most similar emails from the past, based on the email subject and see how the users have responded to the old emails. This way, the business can target customers who have responded positively to the previous campaign (most similar to the new campaign).

➔ *get_recommendations('Entertainment farewells')*

Rank	Email Subject	Similarity Score
1	Your Entertainment	0.7560
2	Weekend Entertainment!	0.7560
3	Entertainment For Children	0.5560
4	2010: entertainment lineup	0.4471
5	Bowl XLIV Entertainment	0.4449

3.5.2. Recommendation System 2 - TensorRec Based recommendation system

Using Tensorflow for building the recommendation system for email marketing. TensorRec is a Python recommendation system that allows you to quickly develop recommendation algorithms and customize them using TensorFlow. A TensorRec system consumes three pieces of data: user_features, item_features, and interactions. It uses this data to learn to make and rank recommendations.

1. Build a recommendation engine capable of learning from explicit positive and negative feedback.
2. Allow for arbitrary TensorFlow graphs to be used as representation functions and loss functions.
3. Provide reasonable defaults for representation functions and loss functions.
4. Pack as many Machine Learning buzzwords into a Medium post as possible.

TensorRec scores recommendations by consuming user and item features (ids, tags, or other metadata) and building two low-dimensional vectors, a “user representation” and an “item representation”. The dot product of these two vectors is the score for the relationship between that user and that item — the highest scores are predicted to be the best recommendations.

In our tensorflow environment, we install tensorrec framework version 2.0, on which we first generate random data for user and item features. Now as told in above para, we make predictions by taking dot products.

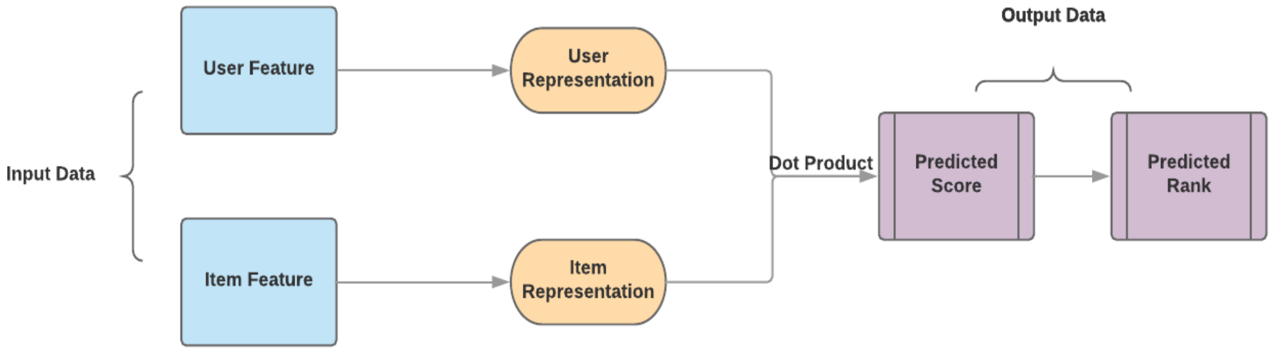
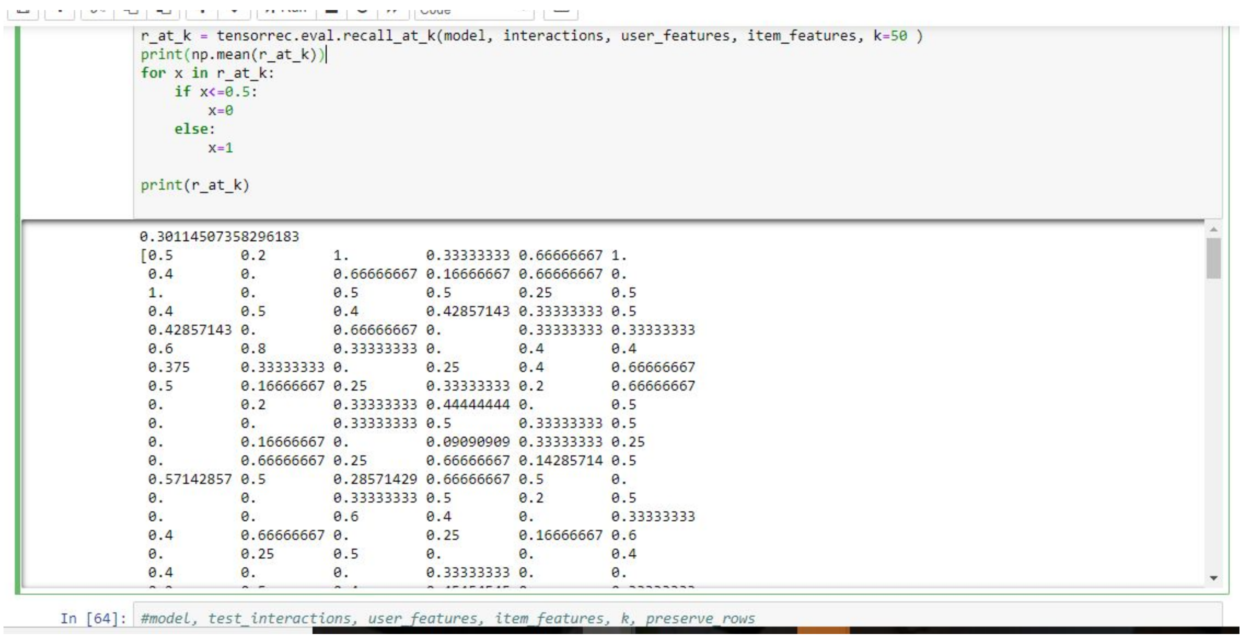


Fig. 22. TensorRec recommendation System

Outcome for running the code for 1000 users and 150 items with 10 epochs. Taking 50 sample set and analysing who all will click the link and not in the range of 0 and 1. According to this, predicting the score to recommend new items to the users.



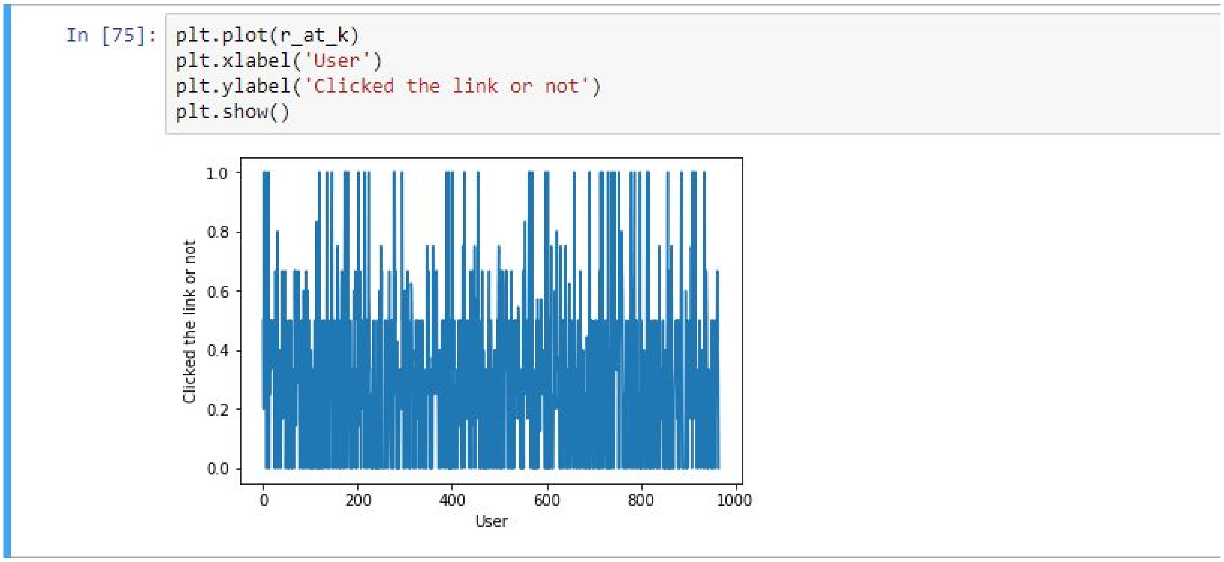
```
r_at_k = tensorrec.eval.recall_at_k(model, interactions, user_features, item_features, k=50 )
print(np.mean(r_at_k))
for x in r_at_k:
    if x<=0.5:
        x=0
    else:
        x=1

print(r_at_k)
```

```
0.30114507358296183
[0.5        0.2        1.         0.33333333 0.66666667 1.
 0.4        0.         0.66666667 0.16666667 0.66666667 0.
 1.         0.         0.5        0.5        0.25       0.5
 0.4        0.5        0.4        0.42857143 0.33333333 0.5
 0.42857143 0.         0.66666667 0.         0.33333333 0.33333333
 0.6        0.8        0.33333333 0.         0.4        0.4
 0.375      0.33333333 0.         0.25       0.4        0.66666667
 0.5        0.16666667 0.25       0.33333333 0.2        0.66666667
 0.         0.2        0.33333333 0.44444444 0.         0.5
 0.         0.         0.33333333 0.5        0.33333333 0.5
 0.         0.16666667 0.         0.09090909 0.33333333 0.25
 0.         0.66666667 0.25       0.66666667 0.14285714 0.5
 0.57142857 0.5        0.28571429 0.66666667 0.5        0.
 0.         0.         0.33333333 0.5        0.2        0.5
 0.         0.         0.6        0.4        0.         0.33333333
 0.4        0.66666667 0.         0.25       0.16666667 0.6
 0.         0.25       0.5        0.         0.         0.4
 0.4        0.         0.         0.33333333 0.         0.
```

```
In [64]: #model, test_interactions, user_features, item_features, k, preserve_rows
```

Fig. 23. TensorRec Sample recommendation for link clicks



```
In [75]: plt.plot(r_at_k)
         plt.xlabel('User')
         plt.ylabel('Clicked the link or not')
         plt.show()
```

Fig. 24. Plot to show user clicks according to their likes

3.5. Analysis of results

3.5.1. Campaign Performance Results

1) What percentage of users opened the email and what percentage clicked on the link within the email?

- ❖ Percent opened = 10.345
- ❖ Percent clicked = 2.119

2) Build a model to optimize in future email campaigns to maximize the probability of users clicking on the link inside the email.

- ❖ Gradient Boosting Tree model to predict whether user will click the link or not.

3) By how much do you think your model would improve click through rate?

- ❖ Emails sent in new way = 9904.000000
- ❖ Emails sent in old way = 33333.000000
- ❖ Saving percentage (%) = 70.287703

4) Did you find any interesting pattern on how the email campaign performed for different segments of users? Explain.

- ❖ All the interesting patterns observed have been documented under the exploratory data analysis section.

3.5.2. Machine Learning Model Comparison

Model	Recall Score
XG Boost	81%
Random Forest	63%
Logistic Regression	60%
Support Vector Machine	64%

4. Discussions and Conclusions

Difficulties faced:

Challenge 1. Email Campaign Dataset availability to suit a machine learning, network analysis and recommendation system based implementation.

Solution:

The dataset provided with the assignment just had basic information and did not have any kind of sender or recipient information. It also lacked email specific information such as email subject. The email marketing dataset is generally confidential and is not available as an open dataset.

In order to work around this, we simulated the complete email marketing dataset by combining three datasets.

- Dataset 1 : email_table (100000 records), email_opened_table (10345 records), link_clicked_table(2119 records).
- Dataset 2 : SpamClickBait Dataset (Used the email subject from this dataset)
  We used this information in building our TF-IDF based content based recommendation system.
- Dataset 3 : Enron dataset - sender and recipient email details. (Used sender and recipient email addresses from this dataset).
  We used this information in performing network analysis by linking the email sender and recipients.

With this complete dataset we were able to build the following:

- Machine learning model to predict the output class
- Network analysis on the sender recipient information
- TF-IDF content based recommendation system

Challenge 2. Working on TensorRec to get the recommendations for the users.

Solution:

- TensorRec is new to us and understanding, installing and implementing it was a challenge. We went through the online tutorials to understand this.
- Transforming the user item information to suit the requirement was also a challenge. TensorRec scores the user-item feature pair into 2 dimension vector.
- The dot product of this relation is the recommendation score. Higher the score, better the recommendation.
- Did multiple trial and error runs to choose the right model parameters with respect to number of layers, epochs etc.

Challenge 3. Understanding the pattern in user's response to the emails.

Solution:

We performed a detailed exploratory analysis to understand user behavior patterns with respect to various features in the dataset using histograms, boxplots, correlation graphs. We analyzed the click through rate against the type of email, time of the day, day of the week, user's country etc, to clearly understand the factors that influence the user's behavior towards marketing email. We also eliminated unwanted features using chi square test and F test.

Challenge 4. Identification of right classification models and parameters..

Solution:

Used techniques like Grid Search, ROC curve, classification report generation, confusion matrix to choose the right model and parameters. Our measure of accuracy was a combination of good sensitivity and specificity. ROC curve is another way of visualizing the performance of a binary classifier. The area under the resulting ROC curve (AUC) gives a complete picture of the performance of a trained classifier on a given dataset.

For each threshold the True Positive rate is plotted against the False Positive rate. Based on our experiments XGBoost turned out to be the best model with highest Recall value for positive click through scenario.

The end goal was to find a model that was fast, explainable and accurate. We also validated the model using k-fold cross validation. The imbalanced dataset was handled using resampling techniques like SMOTE, Random Oversampling and Random Undersampling.

Things that did not work:

1. We wanted to get the data from the search engine and create our own dataset by exploiting the cache of each IP address which has login on the google page, but due to privacy settings and google restrictions, we were not able to get the real data. So we created the dummy random user-item data to perform the recommendation system.
2. Adding the email subject attribute in the existing tables provided by the professor was not easy and finally we decided to drop the idea and simulate the column using another dataset which has the email subject so that we could perform the content based filtering.
3. We also tried using 'Surprise' framework in scikit tool for recommendation, but after running the code on the dummy data for almost 1M users, we found that the output data was not understandable and the error in sample and out sample was more than 50%. For this reason we turned to TensorRec tool.

Conclusion

We started with some questions and datasets provided by the professor to initiate the marketing email campaign report. We first collaborated all the data together, used python code to solve the questions. It helped us work on machine learning algorithm, data cleaning process. After this, we wanted to explore more on this project so searched for more data sets as the data set provided initially was not enough to perform network analysis and recommendation systems.

After finding the relevant data set, we needed more data preparation. We then started with the network analysis through which we could analyse the user information with the sender and recipient information. Second, we applied the recommendation system to make recommendations to the users for better sales. For this we used two methods, one was content based recommendation system for email subjects and their similarities, and two TensorRec tool to predict rank between user and item and accordingly predicting new item for the user.

In the end, we would conclude that we got the opportunity to learn and successfully apply the knowledge given to us by our **Professor Dr. Shih Yu Chang** for this course Large Scale Analytics: **Data exploration, machine learning algorithm, recommendation systems and Network analysis**.

5. Task Distribution

Poojitha Amin	**Charu Ramnani**	**Preksha Pansheriya**
1. Data Preprocessing 2. Exploratory Data Analysis 3. Machine learning models implementation 4. Network Analysis 5. Content Based Recommendation System 6. Report 7. Involved in brainstorming for Project 2 and provided inputs.	1. Recommendation System coding 2. TensorRec installation and running the code 3. Tested ‘Surprise’ scikit framework for recommendation system 4. Report 5. Involved in brainstorming for Project 2 and provided inputs.	1. DataSet Finding and parsing 2. Exploratory Data Analysis 3. Solving the click rate percentage 4. Report 5. Involved in brainstorming for Project 2 and provided inputs.

References

Bawm, Z. L., & Nath, R. P. (2014). A Conceptual Model for effective email marketing. *2014 17th International Conference on Computer and Information Technology (ICCIT)*. doi:10.1109/iccitechn.2014.7073103

Bhat, S. Y., & Abulaish, M. (2013). Overlapping Social Network Communities and Viral Marketing. *2013 International Symposium on Computational and Business Intelligence*. doi:10.1109/iscbi.2013.56

Li, Y., Murali, P., Shao, N., & Sheopuri, A. (2015). Applying Data Mining Techniques to Direct Marketing: Challenges and Solutions. *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*. doi:10.1109/icdmw.2015.30

Project 2 - Subscription Retention Rate Analysis

1. Introduction

The advent of price and product comparison sites now makes it even more important to retain customers and identify those that might be at risk of leaving. The use of data mining methods has been widely advocated for predicting customer churn. The high increase in the number of companies competing in mature and competitive markets makes customer subscription retention an important factor for any company to survive. It is no surprise then that so many companies have subscription business models (or try very hard to come up with one!). Thus, many methodologies such as data mining and statistics have been proposed to analyze and study customer retention.

Subscriptions are a great business model. There are so many advantages for businesses in having subscribers compared to single purchase users: revenue by customer is much higher, it is possible to cross-sell to the subscribers, future revenue is easily predictable, there is a significant cost (time/effort/etc.) for the customer in canceling the subscription, etc.

Let's first define what is customer retention rate. The subscription retention rate is the percentage of an organization's existing customers that are kept or retained during a measured period. The subscription retention rate measures how a company is doing generating loyalty among its customer base; however, like any single metric, it is only one piece of a more complex view of an organization's success in retaining customers.

Customer Retention Rate (CRR) can be defined as, the percentage of customers retained over a given period of time and calculated using the following formula,

$$\mathbf{CRR = ((E\text{-}N)/S) \ X \ 100}$$

where,

E is the number of customers you had at the END of the period (E),

N is the New customers you gained during the period (N),

S is the number by what you started with.

1.1. Goal and Objectives

The goal of this project is to model subscription retention rate.

Following are the predefined objectives to work upon given by our professor,

Company XYZ started a subscription model in January 2015. You get hired as a first data scientist at the end of August and, as a first task, you are asked to help executives understand how the subscription model is doing. As a result of this, you decide to pull data from all the users who subscribed in January and see, for each month, how many of them unsubscribed. In particular, your boss is interested in:

➔ A model that predicts monthly retention rate for the different subscription price points.

➔ Based on your model, for each price point, what percentage of users is still subscribed after at least 12 months?

➔ How do user country and source affect subscription retention rate?

➔ How would you use these findings to improve the company revenue?

1.2. Literature/Market Review

Some services or products require customers to use or subscribe to them in the relatively long term such as insurance services, sports membership, gyms, and satellite/cable TV. In these long-term contract industries, it is very important for marketers to keep or maintain their customers in the long run. The previous research has suggested many factors that influence customer retention. One factor is the impact of previous price or sales promotions on retention [1][2]. Along with this stream of research, we examine the impact of previous price promotion on customer retention in this paper.

Hee-Su Kim, and Choong-Han Yoon [3] focused on the loyalty of the customers of a mobile service provider in terms of churning as well as recommending the service to others. The factors such as the “level of satisfaction with alternative-specific service attributes including call quality, tariff level, handsets, brand image, as well as income, and subscription duration" were affected for the customer loyalty of retaining and "the factors such as call quality, handset type, and brand image affect customer loyalty as measured by the intention/non-intention to recommend the service provider to other people".

Yu, W. Sobey et al., [4] emphasized the weaknesses in existing churn prediction methodologies in terms of determining the reasons behind the customer churn.

Shin-Yuan Hung, David C. Yen and Hsiu-Yu Wang [5] used Customer demography (age, tenure, gender), bill and payment analysis (monthly fee, billing amount, count of overdue payment), call detail records analysis (within network call duration, call type) and customer care/service analysis as features in order to build the models. In their paper they have mentioned specifically to avoid special festival seasons when taking the records as they have experienced some abnormalities in call pattern in special seasons (e.g. Chinese New Year). They have used the decision trees and neural networks in building their models [5].

2. Implementation

2.1. Data set description

The data is collected by recording the information about the users who had subscribed in January 2015 and their subscription status till August 2015. The dataset is a supervised learning dataset containing 50000 data points with seven attributes. The dataset is clean as there are no outliers and missing values in the dataset. The subscription_monthly_cost, source and is_active attributes are categorical variables with three categories ($29, $49 and $99), three categories (ads, seo, friend referral) and two categories (0 and 1), respectively where, 0 indicates that the user unsubscribed and 1 indicates user is an active subscriber. The underlying problem in the dataset is the regression problem.

2.2. Libraries and tools used

In any Machine Learning model, we use independent variables to predict a dependent variable. When the inputs are given, the libraries will do the required task. And will return the required outputs. There are three essential libraries that are used quite often. They are numpy, matplotlib and pandas.

Python 3.6

Python is a very powerful tool, which is also open sourced and flexible, adding more to its popularity. It is known to have massive libraries for manipulation of data and is extremely easy to learn and use for all data analysts.

Numpy

Traditionally, we start our list with the libraries for scientific applications, and NumPy is one of the principal packages in this area. It includes an incredibly versatile structure for working with arrays, which are the primary data format that scikit-learn uses for input data.

Pandas

Pandas is a Python library that provides high-level data structures and a vast variety of tools for analysis. The great feature of this package is the ability to translate rather complex operations with data into one or two commands. Pandas contains many built-in methods for grouping, filtering, and combining data, as well as the time-series functionality.

Scikit-Learn

This Python module based on NumPy and SciPy is one of the best libraries for working with data. It provides algorithms for many standard machine learning and data mining tasks such as clustering, regression, classification, dimensionality reduction, and model selection. The cross validation has been modified, providing an ability to use more than one metric.

Matplotlib

Matplotlib is a low-level library for creating two-dimensional diagrams and graphs. It is the fundamental package for data visualization in Python. This module allows for the creation of everything from simple scatter plots to 3-dimensional contour plots.

2.3. Data Preprocessing

The real-world data always inconsistent and contain missing values and many errors. The raw data can be transformed into an understandable format using a technique called data preprocessing. Data preprocessing involves data cleaning, data integration, data transformation, data reduction and data discretization. Data cleaning deals with missing values, outliers, and other noise. In this project, the dataset is very clean and have no missing values and no outliers.

2.3.1. Get the dataset

The first step to do before starting the project is to import the dataset and set a working directory. Anytime after importing the dataset, the working directory folder must be specified and this folder must contain the dataset. Any folder can be set as the working directory as long as this folder contains the dataset. The subscription dataset can be loaded in the environment using pandas.

2.3.2. Import Essential Libraries

A library is a tool that can be used to perform a specific job or a task. There are three most important libraries. They are matplotlib, pandas, and numpy. There libraries are very useful in the code for the ease of implementation.

2.3.3. Feature Engineering (Retention rates calculation):

The retention rate from January to July is calculated for different price points and used this data for training the model. Billing cycles and is_active features are used for this calculation.

The retention rate can be calculated as follows:

Subscription Retention Rate = (E/S) *100

- E: Number of customers at the end of a period
- S: Total number of customers at the start of that period

The subscription retention rates for each price point for the first seven months are shown in Table. 1

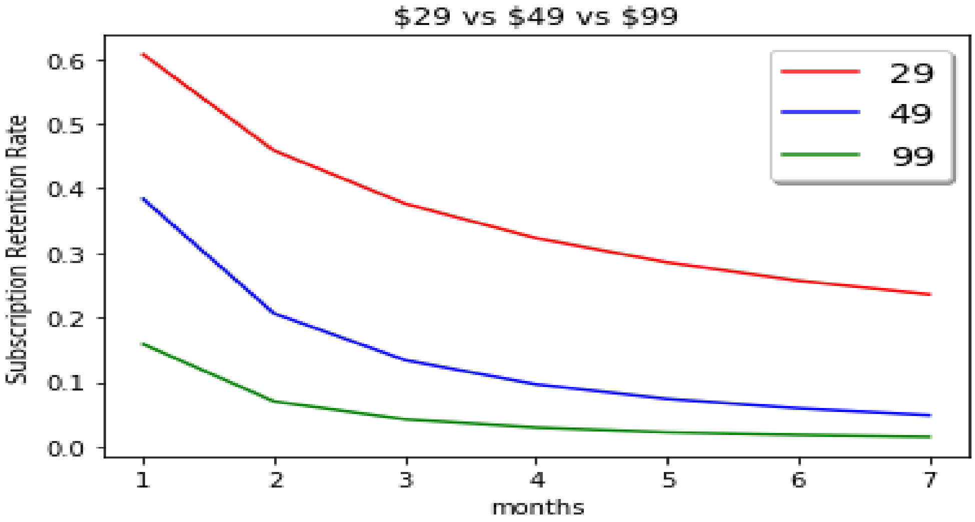


Figure. 1: Comparison of monthly Retention Rates for different price points

Table.1 : Retention Rates of different price points for each month

Months	**Subscription Retention Rate for $29**	**Subscription Retention Rate for $49**	**Subscription Retention Rate for $99**
1 month	60.75%	38.36%	15.87%
2 months	45.84%	20.62%	7.00%
3 months	37.62%	13.40%	4.24%

4 months	32.30%	9.64%	2.96%
5 months	28.54%	7.40%	2.21%
6 months	25.71%	5.94%	1.79%
7 months	23.59%	4.88%	1.50%

2.3.4. Feature Selection

By observing the dataset, the user id and subscription date do not affect the retention rate in any way. For the calculation of retention rate for each price point, subscription_monthly_cost, billing cycle and is_active are used. The remaining features are used to know how they can affect the retention rate.

3. Training Models

3.1. Linear Regression Model

Simple linear regression is an approach for predicting a **response** using a **single feature**. It is assumed that the two variables are linearly related. Hence, we try to find a linear function that predicts the response value(y) as accurately as possible as a function of the feature or independent variable(x). In a SLR model, build a model is built based on data — the slope and Y-intercept derive from the data; furthermore, we don't need the relationship between *X* and *Y* to be exactly linear. SLR models also include the errors in the data (also known as residuals).

It is important to note that in a linear regression, we are trying to predict a continuous variable. In a regression model, we are trying to minimize these errors by finding the "line of best fit" — the regression line from the errors would be minimal. We are trying to minimize the length of the black lines (or more accurately, the distance of the blue dots) from the red line — as close to zero as possible. It is related to (or equivalent to) minimizing the mean squared error (MSE) or the sum of squares of error (SSE), also called the "residual sum of squares." (RSS)

3.1.1. Linear Regression Fit for different price points

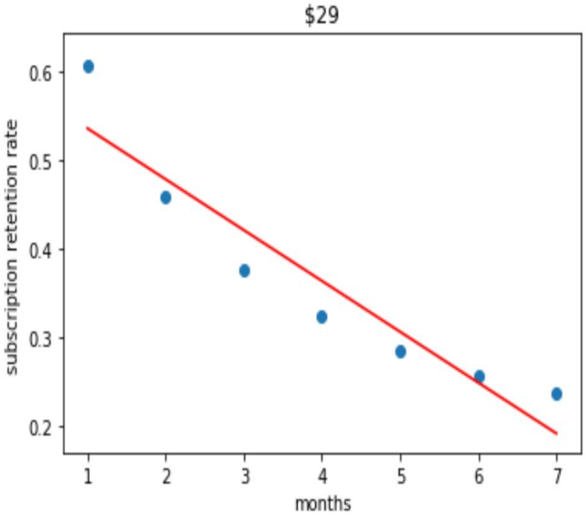


Figure. 2 (a): Linear Regression fit for subscription retention rates for $29 price point

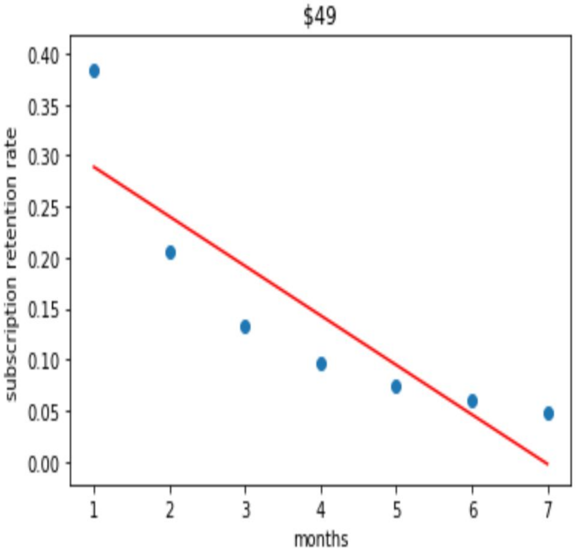


Figure. 2 (b): Linear Regression fit for subscription retention rates for $49 price point

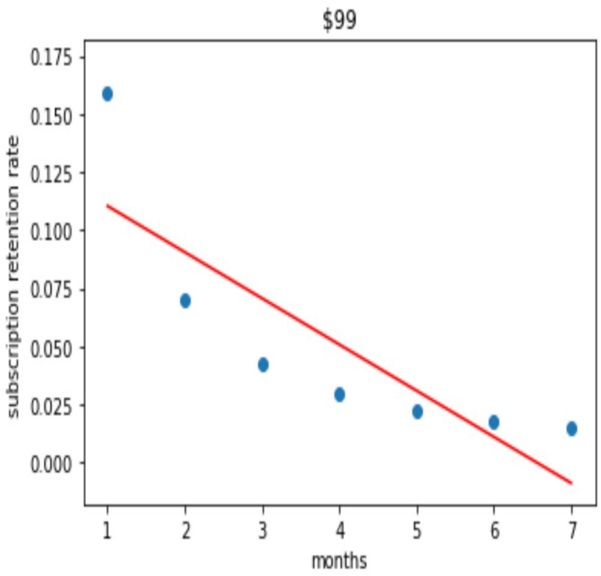


Figure. 2 (c): Linear Regression fit for subscription retention rates for $99 price point

3.2. Exponential Regression Model

This model finds an exponential equation that can best fit the provided data. The resulting equation will look like $y=a.b^x$, where $a\neq0$. The accuracy of this model can be obtained by the relative predictive power (R^2), where $0 \leq R^2 \leq 1$. The closer the value of R^2 to 1, the better the accuracy.

3.2.1. Exponential Regression Fit for different price points

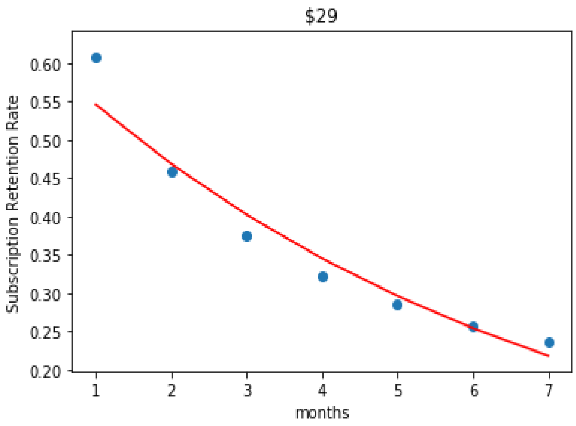


Figure. 3 (a): Exponential Regression fit for $29 price point

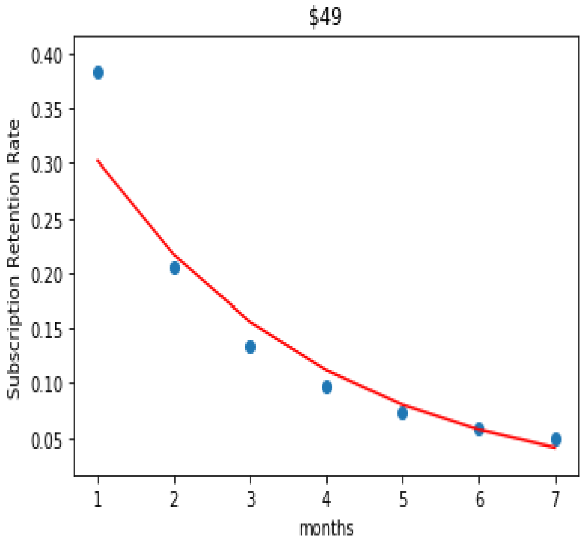


Figure. 3 (b): Exponential Regression fit for $49 price point

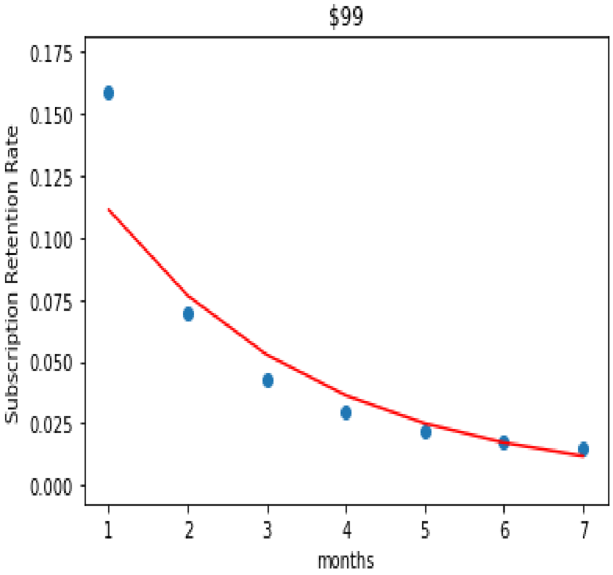


Figure. 3 (c): Exponential Regression fit for $99 price point

4. Model Selection

4.1. Root Mean Square Error (RMSE) values

RMSE is the difference between the observed values and the predicted values. To build a good a good model, the RMSE values for the training set and the test set must be similar. Overfitting occurs when the RMSE for the test set is much higher than the training set. Mathematical representation of RMSE is shown below.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)^2}$$

Table.2: RMSE values for Linear Regression and Exponential Regression

Monthly Cost	**Linear Regression**	**Exponential Regression**
$29	0.04084	0.02809
$49	0.05195	0.03292
$99	0.02589	0.01877

The RMSE values for Linear Regression is greater than the RMSE values for the Exponential Regression. Therefore, Exponential Regression performs well on this dataset than the Linear Regression.

Table.3: Predictions based on Exponential Regression

Months	$29	$49	$99
After 8 months	18.76%	2.97%	0.81%
After 9 months	16.10%	2.13%	0.56%
After 10 months	13.83%	1.53%	0.38%
After 11 months	11.87%	1.10%	0.26%
After 12 months	10.19%	0.79%	0.18%

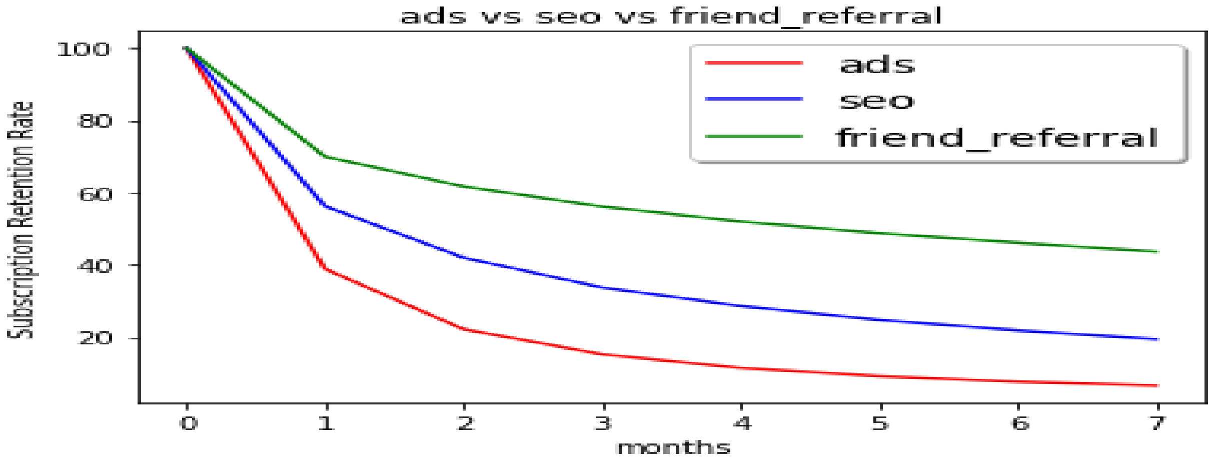


Figure 4: Comparison of Retention Rates of different sources

According to the obtained predictions, the retention rates through ads are less compared to the retentions rates through friend referral and seo.

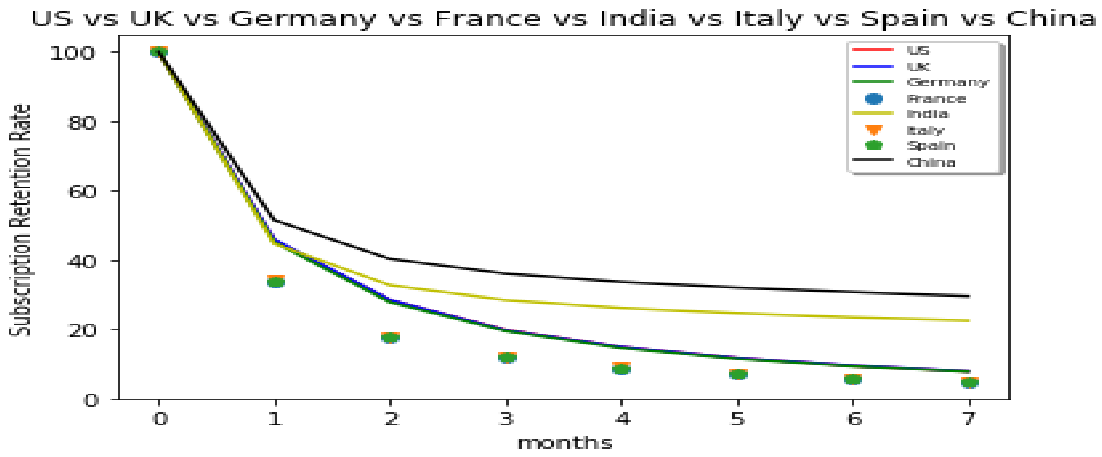


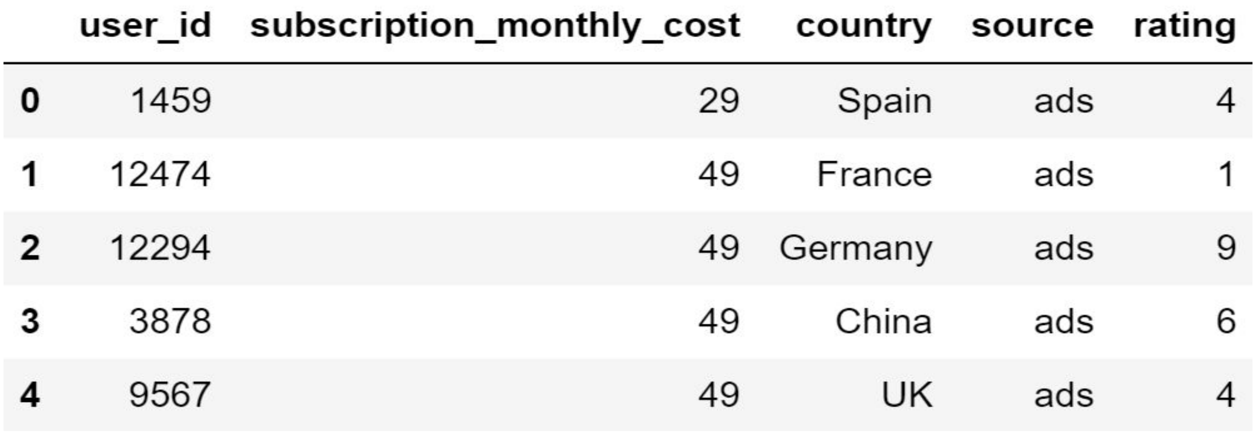
Figure 5: Comparison of retention rates of different countries

5. Business Recommendations

- Company should invest less on ads as it has less retention rate.
- Company should give special offers to users in India China to encourage more friend referrals which in turn will increase the retention rate

5.1 Recommendation Dataset

The first five rows of the recommendation dataset is shown in figure 6.



	user_id	subscription_monthly_cost	country	source	rating
0	1459	29	Spain	ads	4
1	12474	49	France	ads	1
2	12294	49	Germany	ads	9
3	3878	49	China	ads	6
4	9567	49	UK	ads	4

Figure 6: Recommendation Dataset

5.2 User-Based Collaborative Filtering

Used user based collaborative filtering recommendation system. Country and source are considered as the main features. The recommendation system is designed in such a way that if the average of user ratings of similar users (same country and same source) for a product is greater than 5, then that product will be recommended to our particular user.

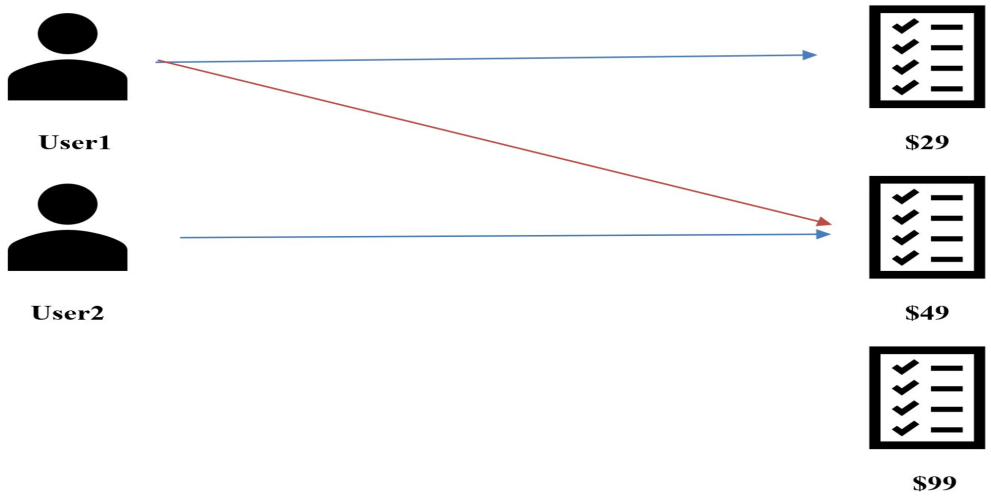


Figure 7: Illustration of user-based recommendation system

6. Challenges

Challenge 1: According to the problem statement, we have to predict the subscription retention rates for each month. However, the dataset does not contain the retention rate label.

Solution:

The output variable subscription retention rate was not present in the dataset. Since it is a supervised learning problem, it is impossible to create ML model without the output variable. After clearly understanding what retention rate is, we came to the conclusion that we had enough features in the dataset using which we could calculate retention rates of different monthly costs. We then used the features Billing_Cycles and Is_Active to calculate retention rates with formula

SRR = (E/S)*10

E: Number of customers at the end of a period

S: Total number of customers at the start of that period

Challenge 2: The dataset is not suitable to create a recommendation system.

Solution:

The dataset was not suitable for the recommendation as it didn't have enough user/item features in it. It didn't have a feature which can be used as a deciding factor or metric for recommendation of different products to a user. To solve this problem, we decided to find other datasets which have similar features, as joining such dataset to the available one could result in a dataset with a good amount of features useful for the recommendation. But, we couldn't find any similar datasets. So, finally, we decided to manually add new features to the existing dataset. We added a new feature (RATING). The feature values were randomly generated and have a range of 1-10.

Challenge 3: Choice of right model for performing regression.

Solution:

Choice of right model for performing regression is key for getting the most accurate predictions possible. After getting the scatter plots of the 3 datasets, we had an intuition that linear regression models would perform poorly on this dataset. So, we had to research what are the different non-linear models available which closely fits the datasets. We used RMSE as a metric to evaluate the accuracy of the predictions and compared the performance of linear and non-linear models on the datasets. We came to the conclusion that exponential regression is the best choice available for the datasets, as the plot of its mathematical formula is similar to the relations between out input and output in the dataset.

7. Conclusion

After carefully getting the insights of the provided dataset, we used Python programming language for data cleaning and other necessary data preprocessing steps. Then the questions that were mentioned in the problem statement were solved which helped us creating a robust recommendation system that can improve the company revenue. User-based collaborative filtering recommendation system is used to make the recommendations to the user.

We are deeply indebted to and want to thank our **Professor, Dr. Shih Yu Chang** for his ideas, guidance and valuable feedback on this project.

8. Task Distribution

Yogeswara Sarat	**Praneeth**	**Prajwal**
1. Data pre-processing. 2. Calculation of retention rates. 3. Exploratory Data Analysis. 4. Involved in brainstorming for Project 1 and provided inputs.	1. Feature Engineering. 2. Training Machine Learning models. 3. Recommendation System Implementation.. 4. Involved in brainstorming for Project 1 and provided inputs.	1. Data pre-processing 2. Presentation 3. Report 4. Involved in brainstorming for Project 1 and provided inputs.

REFERENCES

[1] R.C. Blattberg, E.C. Malthouse, S.A. Neslin Customer lifetime value: empirical generalizations and some conceptual questions. J. Interact. Mark., 23 (2) (2009), pp. 157-168

[2] Y. Polo,F.J. Sese, P.C. Verhoef The effect of pricing and advertising on customer retention in a liberalizing market J. Interact. Mark., 25(4) (2011), pp. 201-214

[3] Hee-Su Kim and Choong-Han Yoon, “Determinants of subscriber churn and customer churn and customer loyalty in the Korean mobile telephony market, “Telecommunications Policy, vol. 28, no. 9-10, pp. 751-765, October-November 2004

[4] Wei Yu, Dawn N. Jutla, and Shyamala C. Sivakumar, “A Churn-Strategy Alignment Model for Managers in Mobile Telecom,” in Communication Networks and Services Research Conference , 2005. Proceedings of the 3rd Annual, May 2005, pp. 48-53

[5] Shin-Yuan Hung, David C. Yen, and Hsiu-Yu Wang, “Applying data mining to telecom churn management,” in Expert Systems with Applications., October 2006, vol. 31, pp. 515-524