Customer Churn Rate Prediction

Final Project Report

Group 3

Prayansh Maheshwari

Neel Shirish Anap

+1 (857)-492-4624
+1 (857)-370-7988

maheshwari.pray@northeastern.edu

anap.n@northeastern.edu

Percentage of Effort Contributed by Student 1: 50%

Percentage of Effort Contributed by Student 2: 50%

Signature of Student 1: Prayansh

Signature of Student 2: Neel Anap

Submission Date: 24th June 2022

Contents

1. Problem Setting

The study of data to evaluate the performance of marketing activity is known as marketing analytics. Businesses can understand what drives consumer actions, refine their marketing campaigns, and maximize their return on investment. In this project, we will focus on one marketing metric, Churn Rate Prediction, which describes the number of customers who leave a business over a specific period. Predicting the Churn rate will allow them to target their existing customers and enhance retention rates.

2. Problem Definition

A certain company that hosts a website to offer its services is experiencing high customer attrition across platforms. To overcome this problem, we need to find the patterns/similarities between customers leaving or staying and determine the relationship between attributes, which will lead us to the primary factors that contribute towards customers' leaving. These factors then, will be used to predict the customer churn rate score (0 or 1), with 1 being the most likely to leave. We'd experiment with different classification algorithms and only use the one that provides the best performance measure.

3. Data Sources

This data source was obtained from Kaggle and can be found at https://www.kaggle.com/datasets/undersc0re/predict-the-churn-risk-rate/metadata.

4. Data Description

'Customer Churn' is the name of the dataset that contains the User's demographic information, browsing behavior, and historical purchase data. The dataset has 24 columns and approximately 37k rows.

Column	Description	Type
Age	Represents the age of a customer	Numerical
Gender	Represents the gender of a customer	Categorical
Security_no	Represents a unique security number that is used to identify a person	Numerical
Region_category	Represents the region that a customer belongs	Categorical
Membership_category	Represents the category of the membership that a customer is using	Categorical
Joining_date	Represents the date when a customer became a member	Datetime
Joined_through_referral	Represents whether a customer joined using any referral code or ID	Categorical
Referral_id	Represents a referral ID	String
Preferred_offer_types	Represents the type of offer that a customer prefers	Categorical
Medium_of_operation	Represents the medium of operation that a customer uses for transactions	Categorical
Internet_option	Represents the type of internet service a customer uses	Categorical
Last_visit_time	Represents the last time a customer visited the website	Datetime

Days_since_last_login	Represents the no. of days since a customer last logged into the website	Numerical
Avg_time_spent	Represents the average time spent by a customer on the website	Numerical
Avg_transaction_value	Represents the average transaction value of a customer	Numerical
Avg_frequency_login_days	Represents the no. of times a customer has logged in to the website	Numerical
Points_in_wallet	Represents the points awarded to a customer on each transaction	Numerical
Used_special_discount	Represents whether a customer uses special discounts offered	Categorical
Offer_application_preference	Represents whether a customer prefers offers	Categorical
Past_complaint	Represents whether a customer has raised any complaints	Categorical
Complaint_status	Represents whether the complaints raised by a customer was resolved	Categorical
Feedback	Represents the feedback provided by a customer	Categorical
Churn_risk_score	Represents the churn risk score that 0 or 1	Numerical

5. Data Pre-processing

Three columns in the dataset have null values: region_category, preferred_offer_types, and points_in_wallet. Null values were present in 14.67 %, 0.78 %, and 9.30% respectively. However, some unidentified values are present in other columns, such as Medium_of_operation and joined_through_referral, which have '?' in some of their cells, Avg_frequency_login_days, which has 'Error,' and Gender, which has 'Unknown' values. Other columns, such as Days_since_last_login, Avg_time_spent, and Points_in_wallet, have negative values, which aren't allowed. To continue processing, we'll replace all of these unidentified/wrong entries with null values.

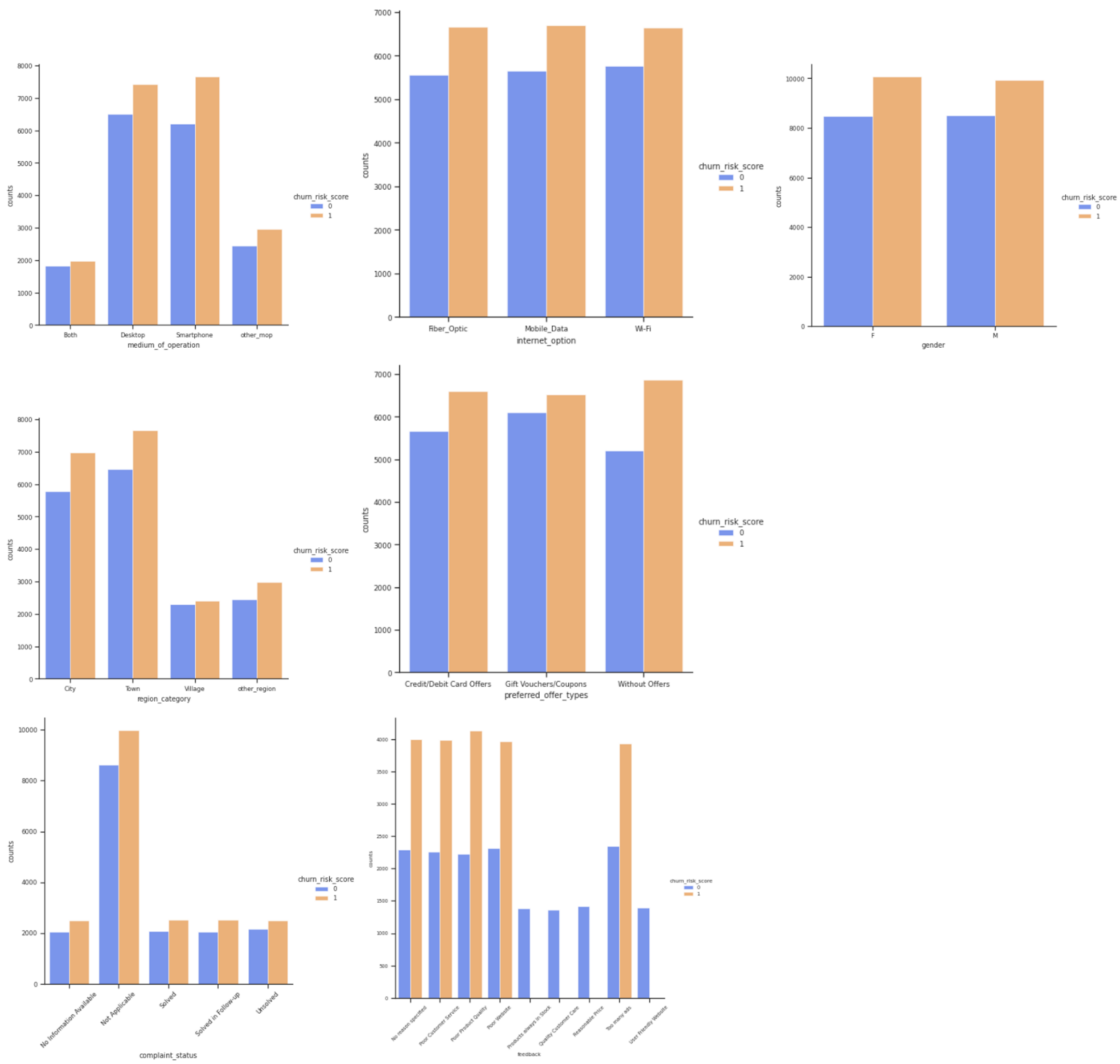
	Percent Null
joined_through_referral	14.700476
region_category	14.673443
medium_of_operation	14.578828
avg_frequency_login_days	11.367323
points_in_wallet	9.675065
days_since_last_login	5.403871
avg_time_spent	4.646951
preferred_offer_types	0.778547
gender	0.159494
age	0.000000
feedback	0.000000
complaint_status	0.000000
past_complaint	0.000000
offer_application_preference	0.000000
used_special_discount	0.000000
last_visit_time	0.000000
avg_transaction_value	0.000000
internet_option	0.000000
referral_id	0.000000
joining_date	0.000000
membership_category	0.000000
security_no	0.000000
churn_risk_score	0.000000

There are 9 variables with null values, and we used the mode method to fill the nulls in categorical variables and the mean method to fill the nulls in numerical variables, with the exception of ‘days_since_last_login’, which was highly skewed, so we used the median method to fill the null values there.

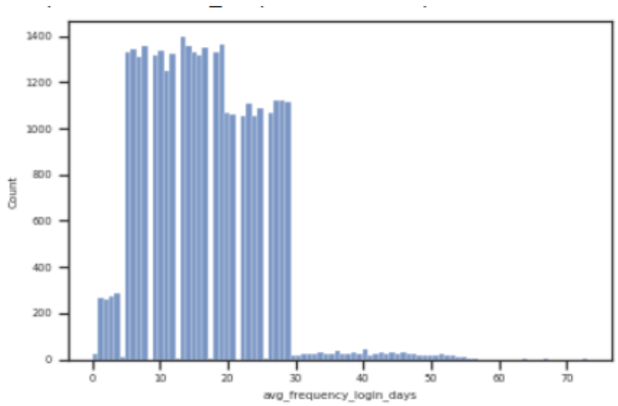
Using domain knowledge, we dropped the ‘referral_id’ column since it has no connection to the response variable. We further checked for duplicates in the ‘security_no’ column and found it to be unique throughout, so we decided to drop it as it would only represent the index. After this, the Ordinal categorical variables like ‘membership_category’ were replaced with hierarchical numeric values. We replaced ‘Yes’/’No’ values with 1/0 in variables like ‘used_special_discount’, ‘offer_application_preference’ and ‘past_complaint’. We created dummy variables using one hot encoding for ‘preferred_offer_types’, ‘medium_of_opertation’ and ‘complaint_status’

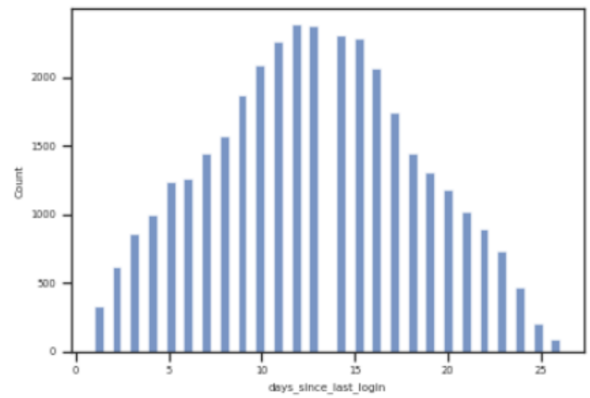
6. Data Exploration and Visualization

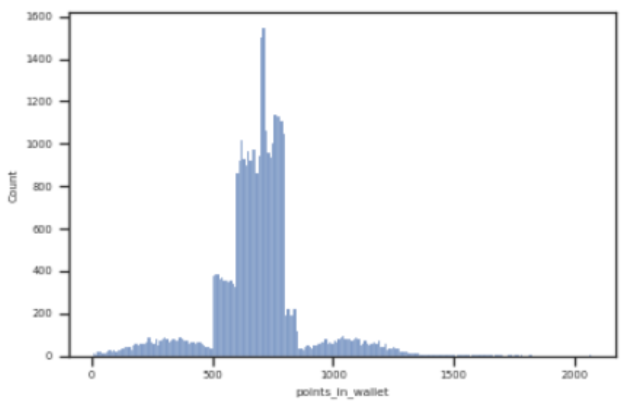
The distribution of each categorical variable is plotted against the target variable Churn risk score. Smartphones and desktops appear to have the most values in the medium of operation variable. Internet option, preferred offer types, and gender data are evenly distributed across the categories. Most of the data in region category is either in city or town. The 'Not Applicable' category has the most data in complaint status, while the other categories have an equal number of values. The feedback variable is especially interesting. We can see that we have a 0 churn_risk_score if the feedback is positive. It's a good finding, even if it doesn't prove causation. Overall, churn_risk_score=1 appears to have more data than its counterpart.

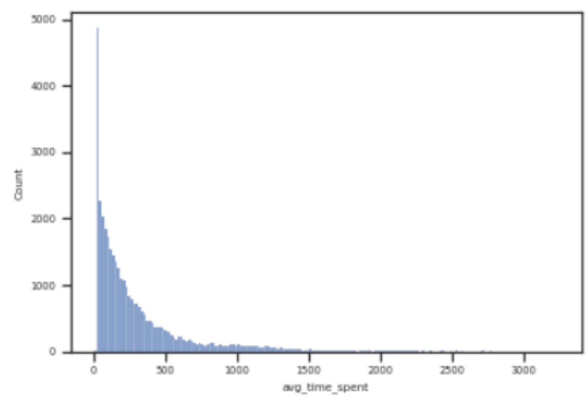


The data for points_in_wallet follows a normal distribution, so we fill nulls in this variable with mean of that variable. The average_time_spent and avg_frequency_login_days variables are highly right skewed, so we fill nulls with median. The days_since_last_login seems to follow a normal distribution.

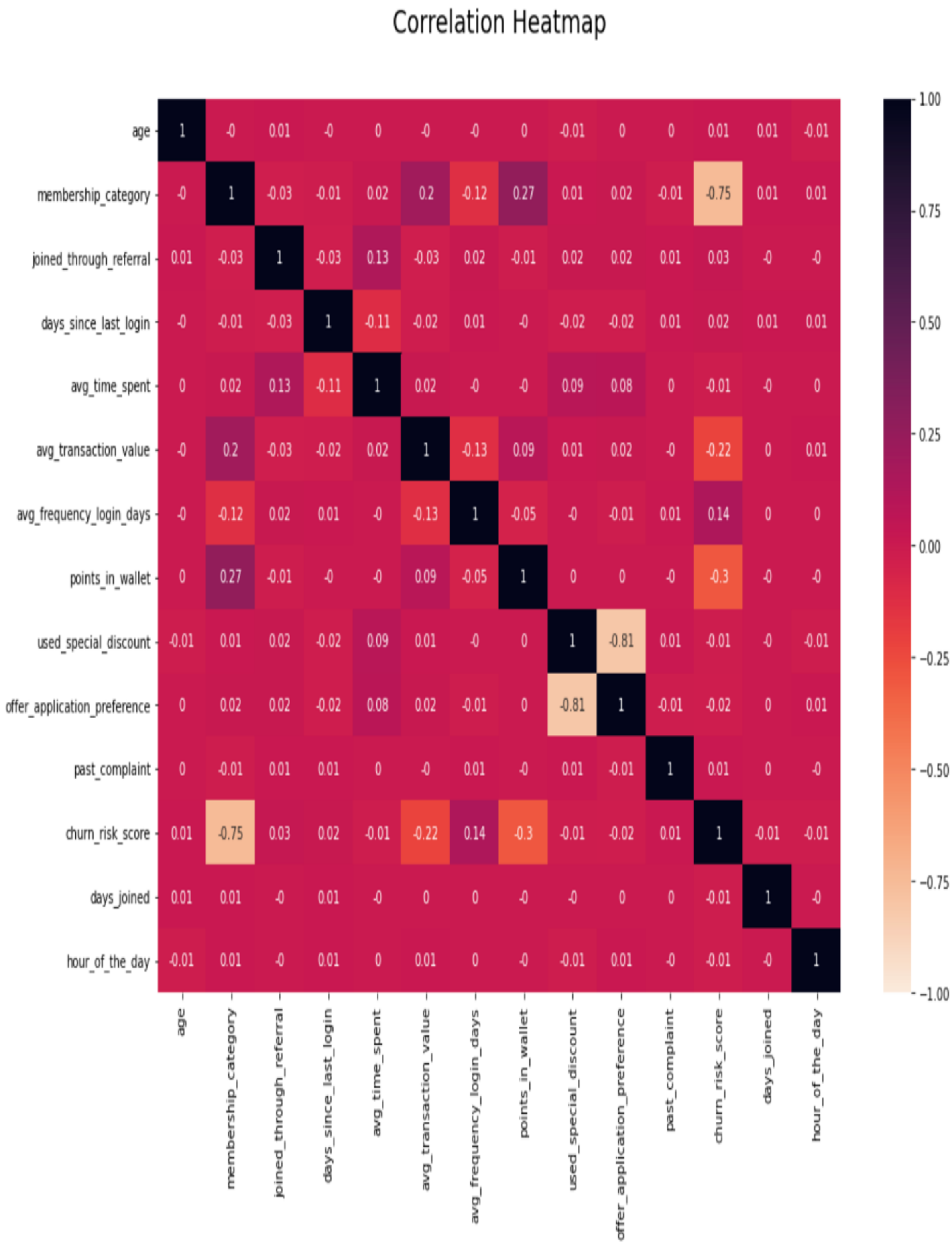








Creating a correlation heatmap for all the variables to see the correlation between them. It is observed that used_special_discount and offer_application_preference has high negative correlation and our target variable churn_risk_score and membership category have high negative correlation, which makes sense as the membership category increases it is less likely for members to churn out. Apart from that, there is no significant relationship between churn_risk_score and any other variables.


Correlation Heatmap
age
membership_category
joined_through_referral
days_since_last_login
avg_time_spent
avg_transaction_value
avg_frequency_login_days
points_in_wallet
used_special_discount
offer_application_preference
past_complaint
churn_risk_score
days_joined
hour_of_the_day
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00

7. Dimension Reduction and Variable Selection

After performing PCA on the standardized data, we can observe that first 35 components capture 100% of the variance of the original dataset. Since almost all variables are not correlated with the response variable that is churn_risk_score, we expected the number of principal components to be close to the number of dimensions in the original data. The cumulative gain of captured variance also increases by almost 3-4% for each principal component.

1	0.059957
2	0.107615
3	0.154141
4	0.196338
5	0.233598
6	0.270295
7	0.305551
8	0.340428
9	0.375239
10	0.409783
11	0.443782
12	0.473448
13	0.501926
14	0.529985
15	0.558021
16	0.586053
17	0.613992
18	0.640941
19	0.667507
20	0.694006
21	0.720420
22	0.744701
23	0.768931
24	0.793082
25	0.816829
26	0.840317
27	0.863515
28	0.886422
29	0.908925
30	0.931213
31	0.951645
32	0.969127
33	0.985490
34	0.996157
35	1.000000
36	1.000000
37	1.000000
38	1.000000
39	1.000000
40	1.000000
41	1.000000
42	1.000000
43	1.000000

8. Model Exploration and Selection

Firstly, we partitioned our data into 75% training and 25% test set randomly. 27,744 records are used for training dataset and remaining 9,248 records as test dataset. Churn_risk_score is our target variable, and all variables after data exploration phase are found significant are used as predictors. After that we scaled the dataset both the training and test dataset using Standard Scalar method to eliminate the error generated due to difference in scales of different variables. As the target variable Churn_risk_score is a binary outcome; we will be implementing classification models to predict it.

We will be implementing multiple models and then will evaluate the performance of each model to select the best fit. The following models will be implemented:

1. Logistic regression
2. KNN
3. Naïve Bayes
4. Decision Tree

8.1 Logistic Regression

Logistic regression is a machine learning classification technique. The dependent variable is modeled using a logistic function. The dependent variable is dichotomous, which means that only two classes are possible. We computed the confusion matrix after fitting the model to training dataset and predicting on test dataset.

```
[[3445  795]
 [ 569 4439]]
accuracy score: 0.85
```

The accuracy of the Logistic regression classifier is 85% for our case.

8.2 KNN

The k-nearest neighbors algorithm, also known as KNN or k-NN, is a non-parametric, supervised learning classifier that makes classifications or predictions about the grouping of individual data points based on proximity. The confusion matrix was generated after fitting the model to training dataset and predicting on test dataset.

```
[[2667 1573]
 [ 423 4585]]
accuracy score: 0.78
```

The accuracy of the KNN classifier is 78% for our case.

8.3 Naïve Bayes

The Nave Bayes Classifier is a simple and effective classification algorithm that helps in building fast machine learning models capable of making quick predictions. It's a probabilistic classifier, which means it makes predictions based on an object's probability. Naïve Bayes assumption is that the predictors are independent which in our case is true. We computed the confusion matrix after fitting the model to training dataset and predicting on test dataset.

```
[[1378 2862]
 [   0 5008]]
accuracy score: 0.69
```

The accuracy of the Naïve Bayes classifier is 69% for our case.

8.4 Decision Tree

Decision Trees can have both numerical and categorical predictors, or both, and they work with categorical response variables. Decision trees have the advantage of having no underlying assumptions, except that they require a large amount of training data, which we have. The confusion matrix was generated after fitting the model to training dataset and predicting on test dataset.

```
[[3836  467]
 [ 404 4541]]
accuracy score: 0.91
```

The accuracy of the Decision Tree classifier is 91% for our case.

8.5 Neural Networks

Neural networks combine predictor information in a very flexible way, allowing them to capture complex relationships between these variables and between them and the outcome variable. We have set the input dimension to 44 as we have 44 predictors, and the same number of hidden layers were used in this model. The output layer has one node and uses sigmoid activation function to get 1 or 0 as the output.

```
[[3754  292]
 [ 486 4716]]
accuracy score: 0.92
```

The accuracy of the Neural Networks is 92% of our case.

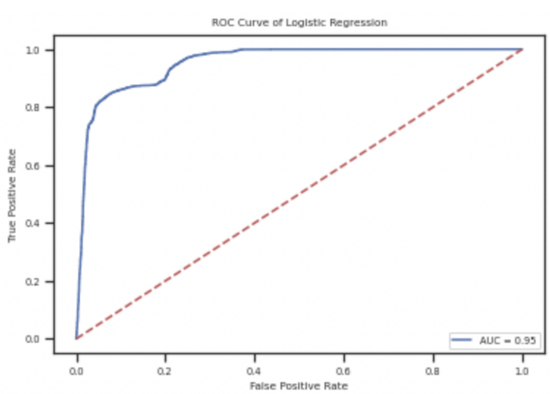
9. Performance Evaluation and Interpretation

In this section, we evaluate and compare the performance of our implemented models in order to find the model that best fits our needs. We have discussed metrics such as AUROC, Sensitivity, Error, Accuracy, Specificity, and F1 score.

9.1 Logistic regression

Logistic Regression is used with the cutoff value of 0.5.

On plotting ROC Curve:



AUC value of ROC curve is 0.95

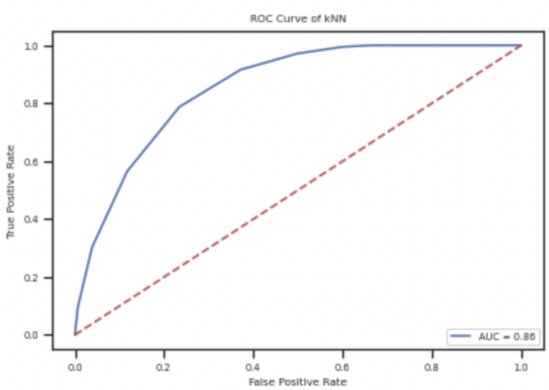
The other metrics are as follows:

```
accuracy score: 0.85
Error: 0.15
Precision: 0.85
Recall: 0.89
F1 Score: 0.87
```

9.2 KNN

The KNN classifier was fine-tuned to find the k value that maximized accuracy, which turned out to be k=9. The KNN classifier was computed using the default Minkowski method.

On Plotting the ROC Curve:



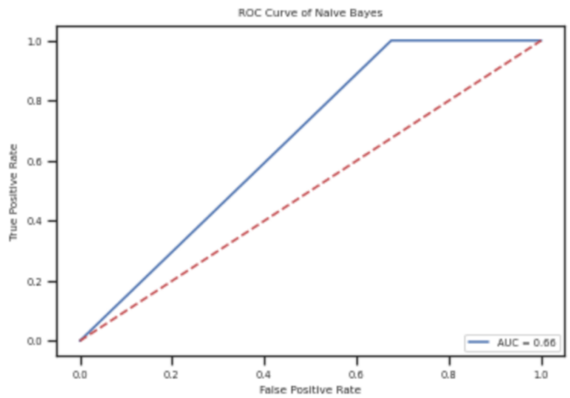
AUC value of ROC curve is 0.86

The other metrics are as follows:

```
accuracy score: 0.78
Error: 0.22
Precision: 0.74
Recall: 0.92
F1 Score: 0.82
```

9.3 Naïve Bayes

On plotting the ROC curve, we get the following:



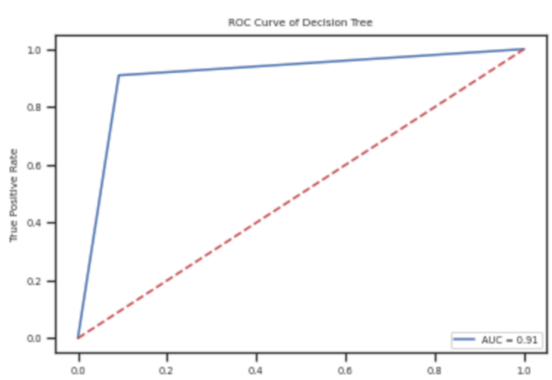
AUC value of ROC curve is 0.66

The other metrics are as follows:

```
accuracy score: 0.69
Error: 0.31
Precision: 1
Recall: 0.64
F1 Score: 0.78
```

9.4 Decision Tree

On plotting the ROC curve, we get the following:



AUC value of ROC curve is 0.91

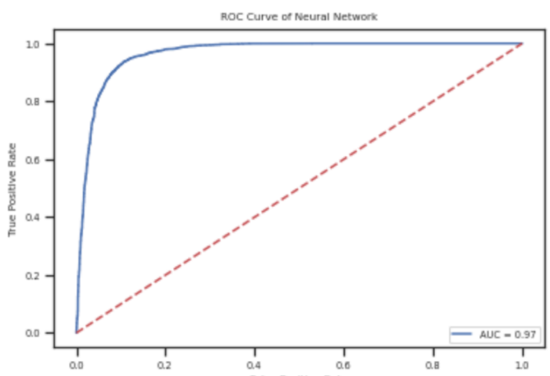
The other metrics are as follows:

```
accuracy score: 0.91
Error: 0.09
Precision: 0.91
Recall: 0.92
F1 Score: 0.91
```

9.5 Neural Networks

A sequential model with three layers is implemented for neural networks. The nodes in the hidden layer are set to 44. There is only one node in the output layer, and the sigmoid function is used.

On Plotting the ROC Curve:



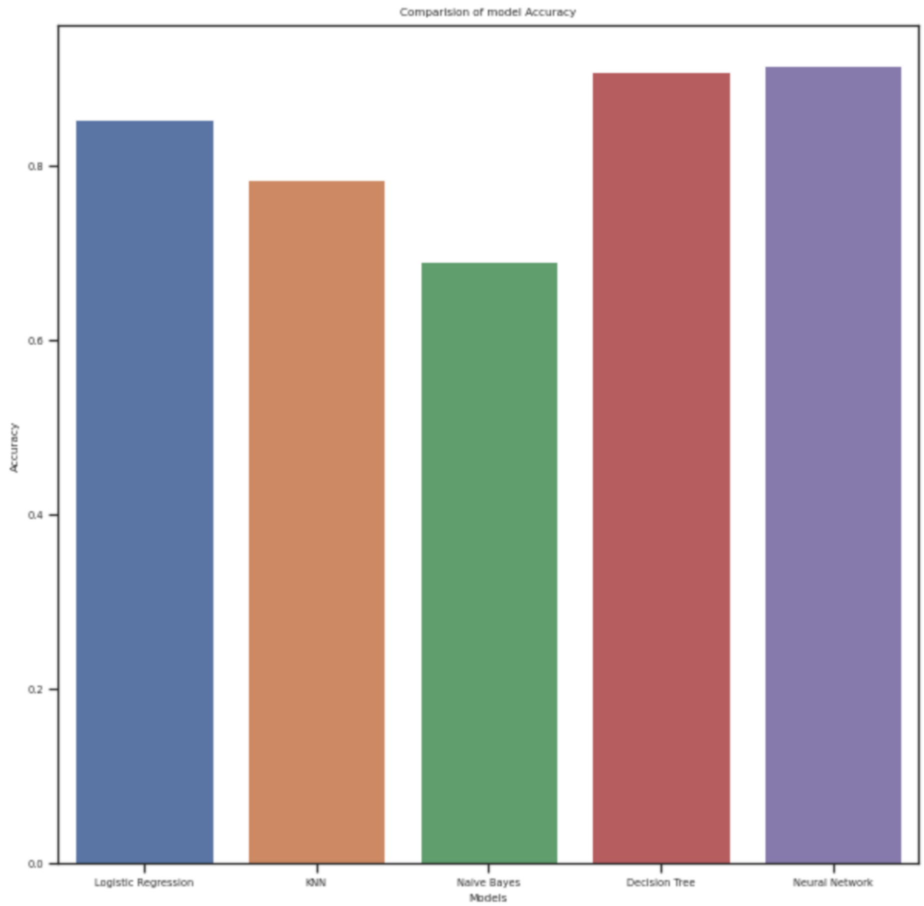
AUC value of ROC curve is 0.97

The other metrics are as follows:

```
accuracy score: 0.92
Error: 0.08
Precision: 0.94
Recall: 0.91
F1 Score: 0.92
```

10. Results and Impacts

Now, comparing Accuracy for all the models:



We require higher accuracy and sensitivity at the expense of low specificity because our model aims to understand whether a customer intends to stay a customer or churn. Even if some customers who are not likely to make a purchase are identified as potential customers, showing them recommendations may result in increased revenue.

Decision Trees and Neural Networks have the best accuracy, according to the metric comparison shown above. The AUROC value of Neural Networks is 0.97, indicating that it is more capable of distinguishing both classes than Decision Trees. Hence, for our use case Neural Networks will perform the best with 92% accuracy and 90% sensitivity.

The main objective of this project was to accurately predict the churn risk score of customers of a particular e-commerce website, and a neural network is the best model for this task. Customer Churn is a common problem that is prevalent in every industry. E-commerce companies and websites can utilize this model to come up with ways to reduce customer churn. Since the logistic regression model gave an accuracy of 85%, companies can make use of this to understand the dependency of the target variable on the predictors, thereby narrowing down the solution since it is not possible to view this dependency using neural networks.

References:

1. https://www.kaggle.com/datasets/undersc0re/predict-the-churn-risk-rate/metadata.
2. https://scikit-learn.org/stable/supervised_learning.html
3. https://towardsdatascience.com/churn-prediction-using-neural-networks-and-ml-models-c817aadb7057