

[**E-retail factors for customer activation and retention: A case study from Indian e-commerce customers**](https://www.researchgate.net/publication/346412647_E-retail_factors_for_customer_activation_and_retention_An_empirical_study_from_Indian_e-commerce_customers)

Submitted by:

Rajat Jibhkate

**ACKNOWLEDGMENT**

During completion of this project, I refer various sources like GitHub, Data Trained institute’s reference materials.

**INTRODUCTION**

Business Problem Framing

Customer retention is a top priority for many companies. Acquiring new customers can be several times more expensive than retaining existing ones.

Customer satisfaction is important parameter that guaranteed the success of online store; It has been posited as a key stimulant of purchase, repurchase intentions and customer loyalty.

Conceptual Background of the Domain Problem

Gaining an understanding of the customer’s churn is a powerful component of designing a customer retention strategy.

Review of Literature

A comprehensive review of the literature, theories and models have been caried out to propose the models for customers activation and customer retention. Five major factors that contributed to success of e-commerce store have been identified as: service quality, system quality, information quality, trust and net benefits. The research furthermore investigated the factors that influence the online customers repeat purchase intention. The combination of both utilitarian value and hedonistic values are needed to affect the repeat purchase intention (loyalty) positively. The data is collected from the Indian online shoppers. Results indicate the e-retail success factors, which are very much critical for customer satisfaction.

Motivation for the Problem Undertaken

The goal of this project is to perform a deep analysis of data in terms of customer retention, building a mechanism for identifying customers at risk of churn and supporting the prevention of churn.

**Analytical Problem Framing**

Mathematical/ Analytical Modelling of the Problem

During project building, we run statistical analysis of all available attributes, analyse existing data structure.

Tasks that have been performed from data point of view –

1. Analysis of available data types
2. Visual data analysis
3. Correlation analysis
4. Outlier detection
5. Analysis and definition of the ‘’target’’ variable.

Based on the results and insight obtained regarding these steps, we have a better understanding of what variables we will be able to generate at the data preparation stage and what the system architecture will look like.

Data Sources and their formats

The original dataset is in excel format. There are more than 47 columns present in dataset like Gender of respondent, How old are you? etc. The dataset stored in df variable. df.head(), df.tail() and df.shape helps to provide information on first five, last five rows and dimension of dataset. Dataset is combination of object and integer type of data. There are no null values present in dataset. For object type of data, LabelEncoder is used to convert it into integer type of data. df.describe() gives stastical details like count, mean, standard deviation, min, max, 25%,50%, 75% etc

Data Pre-processing

At this stage, the main task is to prepare data for machine learning modelling. It is important to properly aggregate data, create all available variables.

It is also very important to define the target variables.

In data processing stage, we checked for dimension of data (df.shape), Type of data (df.info()), Null values (df.isnull().sum()) present in dataset. If null values present in dataset, then fill it with data with help of mean/median or mode methods. For string/object type of data, it is important to convert it into integer datatype. So for these purpose LabelEncoder() used. For any unnecessary columns, we can dropped with help of df.drop().

Data Inputs- Logic- Output Relationship

Input parameters (features) and Output (labels/target values) are two important parameters of any dataset. Based on features, target values changed. So, it is important to analysed features parameter to predict correct target values.

In this project, ‘Since How Long You are Shopping Online ?’ is taken as output parameter. Based on input parameters like How many times you have made an online purchase in the past 1 year? Longer time to get logged in (promotion, sales period) etc will define customer loyalty and number of years customer stayed with that particular online retail shop.

Hardware and Software Requirements and Tools Used

Libraries used while building model are –

1. pandas and numpy – pandas is mainly used for data analysis. Pandas allows importing data from various file formats such as comma-separated values etc. Pandas allows various data manipulation operations such as data cleaning, data wrangling, selecting etc.

numpy provides a multidimensional array object. It can be used for various math operations.

1. Matplotlib.pyplot and seaborn – These are visualization techniques. It helps to plot various graph based on datatypes like scatter plot, Bar graph, distplot etc which are used in this model.
2. Warnings – used to avoid any unnecessary popup while running model.
3. LabelEncoder – It helps to convert string/object dataset into integer dataset.
4. Zscore – Helps to remove skewness present in dataset.
5. Classification\_report, accuracy\_report,

confusion\_matrix – classification\_report is used to measure the quality of prediction from classification algorithm, accuracy\_report gives Number of correct predictions to Total number of predictions, confusion\_matrix is a tabular summary of the number of correct and incorrect predictions made by a classifier.

1. LogisticRegression/KNeighborsClassifier/RandomForestClassifier/DecisionTreeClassifier – It helps for model instantiating and training.
2. Cross\_val\_score – It gives cross validation score.
3. GridSearchCV – It helps to give correct accuracy score for model after adjusting any overperformance of model.
4. Joblib – It helps to save the model.

**Model/s Development and Evaluation**

Identification of possible problem-solving approaches (methods)

At this stage, It is important to create a proper machine leaning model in accordance with best practices. It involves steps –

1. Data pre-processing – Clean and transform data into an appropriate format
2. Conduct features selection in order to choose the most relevant set of variables.
3. Selecting appropriate metrics to measures the performance of the model.
4. Train several models
5. Validate stability of the model
6. Analyse result of model

Testing of Identified Approaches (Algorithms)

List of algorithms used in models are –

1. LogisticRegression
2. KNeighborsClassifier
3. RandomForestClassifier
4. DecisionTree Classifier

Run and Evaluate selected models

1.LogisticRegression – It is classification Model. It is used when output parameter is discreet.

from sklearn.linear\_model import LogisticRegression

LR = LogisticRegression()

LR.fit(x\_train,y\_train)

predlr = LR.predict(x\_test)

print(accuracy\_score(y\_test,predlr))

print(confusion\_matrix(y\_test,predlr))

print(classification\_report(y\_test,predlr))

Output –

0.6470588235294118

[[ 4 3 2 0 1]

[ 0 3 0 0 1]

[ 0 0 11 3 6]

[ 1 0 1 3 0]

[ 0 3 2 1 23]]

precision recall f1-score support

1 0.80 0.40 0.53 10

2 0.33 0.75 0.46 4

3 0.69 0.55 0.61 20

4 0.43 0.60 0.50 5

5 0.74 0.79 0.77 29

accuracy 0.65 68

macro avg 0.60 0.62 0.57 68

weighted avg 0.69 0.65 0.65 68

2.KNeighborsClassifier – It is also classification model. It looks for the 5 nearest neighbors.

from sklearn.neighbors import KNeighborsClassifier #Importing requried lib.

KNN = KNeighborsClassifier()

KNN.fit(x\_train,y\_train)

predknn = KNN.predict(x\_test)

print(accuracy\_score(y\_test,predknn))

print(confusion\_matrix(y\_test,predknn))

print(classification\_report(y\_test,predknn))

Output –

0.5882352941176471

[[ 7 0 3 0 0]

[ 0 3 0 1 0]

[ 0 0 8 6 6]

[ 1 0 0 3 1]

[ 2 3 3 2 19]]

precision recall f1-score support

1 0.70 0.70 0.70 10

2 0.50 0.75 0.60 4

3 0.57 0.40 0.47 20

4 0.25 0.60 0.35 5

5 0.73 0.66 0.69 29

accuracy 0.59 68

macro avg 0.55 0.62 0.56 68

weighted avg 0.63 0.59 0.60 68

3.RandomForestClassifier – It is ensemble learning method for classification constructing a multitude of decision trees at training.

from sklearn.ensemble import RandomForestClassifier

RAN = RandomForestClassifier()

RAN.fit(x\_train,y\_train)

predran = RAN.predict(x\_test)

print(accuracy\_score(y\_test,predran))

print(confusion\_matrix(y\_test,predran))

print(classification\_report(y\_test,predran))

Output –

0.8235294117647058

[[ 6 3 0 0 1]

[ 0 3 0 0 1]

[ 0 0 18 1 1]

[ 0 0 0 5 0]

[ 0 3 1 1 24]]

precision recall f1-score support

1 1.00 0.60 0.75 10

2 0.33 0.75 0.46 4

3 0.95 0.90 0.92 20

4 0.71 1.00 0.83 5

5 0.89 0.83 0.86 29

accuracy 0.82 68

macro avg 0.78 0.82 0.77 68

weighted avg 0.88 0.82 0.84 68

4.DecisionTreeClassifier – It is tree structured classifier, where internal nodes represent the features, branches represent the decision rules and each leaf node represents the outcome.

from sklearn.tree import DecisionTreeClassifier

dt = DecisionTreeClassifier()

dt.fit(x\_train,y\_train)

preddt = dt.predict(x\_test)

print(accuracy\_score(y\_test,preddt))

print(confusion\_matrix(y\_test,preddt))

print(classification\_report(y\_test,preddt))

Outcome –

0.8235294117647058

[[ 4 3 3 0 0]

[ 1 3 0 0 0]

[ 0 0 19 1 0]

[ 0 0 0 5 0]

[ 1 3 0 0 25]]

precision recall f1-score support

1 0.67 0.40 0.50 10

2 0.33 0.75 0.46 4

3 0.86 0.95 0.90 20

4 0.83 1.00 0.91 5

5 1.00 0.86 0.93 29

accuracy 0.82 68

macro avg 0.74 0.79 0.74 68

weighted avg 0.86 0.82 0.83 68

Key Metrics for success in solving problem under consideration

1. CrossValidationScore –

from sklearn.model\_selection import cross\_val\_score

1. scr = cross\_val\_score(LR,x,y,cv=3)

print('Cross validation score for Logistic Regression :',scr.mean()

Output –

Cross validation score for Logistic Regression : 0.6915938410320432

1. scr = cross\_val\_score(dt,x,y,cv=3)

print('Cross validation score for Decision Tree Classifer:',scr.mean())

Output –

Cross validation score for Decision Tree Classifer: 0.8404078235538911

1. scr = cross\_val\_score(KNN,x,y,cv=3)

print('Cross validation score for KNN :',scr.mean())

Output –

Cross validation score for KNN : 0.6359966708281315

D . scr = cross\_val\_score(RAN,x,y,cv=3)

print('Cross validation score for Random Forest Classifer :',scr.mean())

Output –

Cross validation score for Random Forest Classifer : 0.8554723262588432

2.HyperParameterTuning –

from sklearn.model\_selection import GridSearchCV

parameters = {'n\_estimators':[200,700],

'max\_features':['auto','sqrt','log2'],

'max\_depth':[4,5,6,7,8],

'criterion':['gini','entropy']}

GCV = GridSearchCV(RandomForestClassifier(),parameters,cv=2)

GCV.fit(x\_train,y\_train)

Output –



Out[291]:

GridSearchCV(cv=2, estimator=RandomForestClassifier(),

param\_grid={'criterion': ['gini', 'entropy'],

'max\_depth': [4, 5, 6, 7, 8],

'max\_features': ['auto', 'sqrt', 'log2'],

'n\_estimators': [200, 700]})

GCV.best\_params\_

Output –

{'criterion': 'entropy',

'max\_depth': 7,

'max\_features': 'sqrt',

'n\_estimators': 200}

mod = RandomForestClassifier(criterion='entropy',max\_features='sqrt',max\_depth=7, n\_estimators=200)

mod.fit(x\_train,y\_train)

pred = mod.predict(x\_test)

print(accuracy\_score(y\_test,pred)\*100)

Output – 73.52941176470588

Interpretation of the Results

This shows the significance of variables from a trained ML model.

The most important variables that determines whether a customer decides to stay or not.

This type of result helps to find important factors that influence decision to leave or not.

**CONCLUSION**

Key Findings and Conclusions of the Study

Customer retention is the ability of a company to retain its customer during a given time period.

Having a high customer retention means that a low percentage of existing customers leaves your company. This is a very important key performance indicators of company.

Retaining current customer is easier and cheaper than acquiring new ones.

Long-time customers spend more and easier to up-sell.

Five major factors that contributed to success of e-commerce store have been identified as: service quality, system quality, information quality, trust and net benefits.