MAT394 report
E-Commerce Customer Churn Prediction

Team:-
Deepanshu Purohit(1710110101)
Chaitanya Reddy(1710110092)

Abstract

Machine learning is an integral part of many businesses today. One such sector where machine learning plays an important role is E-commerce. Many of these E-commerce firms work on a subscription-based revenue system. This revenue system, although profitable, comes with its own set of challenges. One such challenge is customer churn where customers cancel their subscription thus opting out of the firm's service. To avoid customer churn, firms hire analysts who build machine learning models using historical data of customer churn so that they can predict which current customers are more likely to churn in the future. This allows the company to act before the customer unsubscribes. In this project, we have built two such ML models using data from an E-commerce company. The two ML models are based on logistic regression and random forest techniques. We find that the accuracy of the random forest model is better than the accuracy of a logistic regression model. Apart from that, we have also tried to explain the mathematics behind the two techniques.

Introduction

Simply put, customer churn occurs when customers or subscribers stop doing business with a company or service. Customer churn is a costly affair for these businesses as it essentially means they are losing out on customers. So, to avoid losing customers, the company needs to be able to predict which of their customers is more likely to churn. This is where machine learning comes into play. Using machine learning techniques the business creates a model using historical data and uses this model on current customers to generate the likelihood of a churn. This model also can give the company an idea as to which factor is influencing customer churn in their business. Once the company predicts the likely churners it can offer these customers some form of incentive to stay subscribed to the company. This machine learning model can not only increase the company's growth but also help them in saving costs by letting them know in which part of the business model they need to focus their money and efforts.

In this report, we have used two machine learning techniques to predict customer churn for an E-Commerce company whose data we have obtained from Kaggle(the link is attached in the reference section). The two techniques we have used are logistic regression and random forest.
We have created two models using the same dataset one with logistic regression and the other using random forest. And then we have compared the accuracy of the two models. We find that the accuracy of random forest is 5-6% better than the logistic regression. An overview of the logistic regression and the random forest is given below. The mathematical part of these techniques is explained in the methodology section
Logistic Regression is a Machine Learning algorithm that is used for classification problems, it is a predictive analysis algorithm and based on the concept of probability. Logistic regression is a commonly used technique when the outcome is a discrete variable with only two values.
Random forest is a supervised learning algorithm. The "forest" it builds is an ensemble of decision trees, usually trained with the "bagging" method. The general idea of the bagging method is that a combination of learning models increases the overall result.

Data

The data used in the two models have been obtained from Kaggle, the link to which is attached in the appendix. The data contains 5630 observations and 19 variables which are simply the factors affecting customer churn.
To train the model and test its accuracy we have divided the historical data into train and test datasets. The train data set contains random 80% values of the original dataset. We will be using this training dataset to create a model and then test its accuracy on the test dataset which contains the remaining 20% values of the original dataset.
The table below shows the factors that have been used and their type.

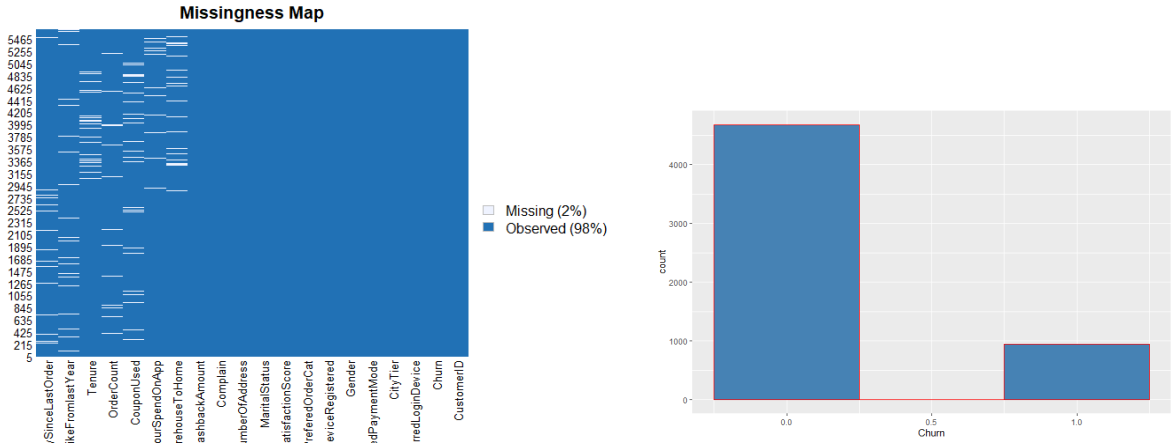
Variable	Description	Data type
CustomerID	Unique customer ID	
Churn	Churn Flag	categorical
Tenure	Tenure of customer in organization	integer
PreferredLoginDevice	Preferred login device of customer	categorical
CityTier	City tier	categorical
WarehouseToHome	Distance in between warehouse to home of customer	integer
PreferredPaymentMode	Preferred payment method of customer	categorical
Gender	Gender of customer	categorical
HourSpendOnApp	Number of hours spend on mobile application or website	integer
NumberOfDeviceRegistered	Total number of deceives is registered on particular customer	integer
PreferedOrderCat	Preferred order category of customer in last month	categorical
SatisfactionScore	Satisfactory score of customer on service	categorical
MaritalStatus	Marital status of customer	categorical
NumberOfAddress	Total number of added added on particular customer	integer
Complain	Any complaint has been raised in last month	integer
OrderAmountHikeFromlastYear	Percentage increases in order from last year	integer
CouponUsed	Total number of coupon has been used in last month	integer
OrderCount	Total number of orders has been places in last month	integer
DaySinceLastOrder	Day Since last order by customer	integer
CashbackAmount	Average cashback in last month	integer

The main dependent variable for the logistic regression is the “churn” dummy which takes the value 1 if the customer churned and 0 if the customer did not churn. The independent variables are the variables mentioned in the above table.

Methodology

➔ Descriptive statistics

About 2% of the historical data contains N/A values which can be seen from the graph below the X-axis the graph contains variable names and on the Y-axis it contains the data points. The white lines in the graph represent the missing values in the data.



The second graph shows the frequency of churners in the data.The right bar represents the customer who churned.

➔ Data manipulation

The first step in this is to change the type of variables. Now the variables we got when we imported the dataset were mostly character type or numeric type, so we needed to change the type of variables to factor or numeric. The variables that were listed as categorical in the data table are given the “factor” data type while the variables that were integer are given “numeric” data. The changed data types have been attached in the appendix.
The second step is to deal with N/A’s. Now we cannot delete all the observations with N/A as we will be losing almost 30% of our data this way. We can replace the N/A values with the mean or the median of the variable.In this project we have used the median.

➔ **Logistic regression**

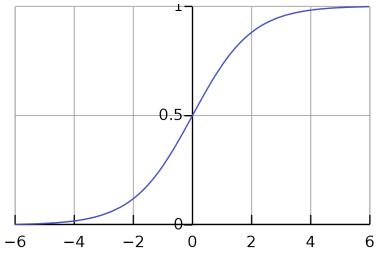
Before writing the regression equation we need to decide how to approach factor variables as they cannot be used directly, for that, R is converting each level of the factor variable into a dummy variable.

Logistic regression equation:

$$Churn_{0,1} = \beta_0+\beta_1 x_1+\beta_2 x_2+.....+\beta_{19} x_{19}$$

where x_1 to x_n are the different factors that have been mentioned in the variables table in the data section. We can write the above equation as Churn=β^TX where β^T=(β_0,β_1,..... β_{19}) and X is the vector for the features.
Now, one of the problems with this hypothesis equation is that the RHS can take any value, but the LHS can take only two values. So, we need to translate this function into a logistic regression function which we will be doing using the logistic/sigmoid function.

$$\theta(t) = e^t/(1 + e^t)$$



We transform the hypothesis function into :

$$h(X)=\theta(\beta^T X)$$

This h(X) takes values between 0 and 1.
Labels for Churn(y) : y=1 or y=0
Assigning probabilities to the labels: Pr(y=1|X) =h(X),Pr(y=0|X)=1-h(X)
For the samples labeled “1” the main aim is to get h(X) → 1 as it would imply the Pr(y=1|X)→1 and for the samples labeled “0” the main aim is to get 1-h(X) → 1 as it would imply Pr(y=0|X) → 1.
We need to estimate β^T as X contains the features whose value is already given to us so,β^T is the only unknown parameter.
The goal of the learning algorithm is to estimate β =(β_0,β_1,.....β_{19}) which is the only unknown parameter in the h(X) function.
Combining the two probabilities for h(X) we can generate a probability function as given below.

$$Pr(y|x)=(h(X))^y.(1-h(X))^{(1-y)}$$

This probability function encapsulates both the probability function given above. We need to maximize this probability for each value in the training data set in order to get the best values of β .To do that we will use the maximum likelihood estimation.

$$\begin{aligned} L(\beta) &= P(\vec{y}|X;\beta) \\ &= \prod_{i=1}^{n} P\left(y^{(i)}|x^{(i)};\beta\right) \\ &= \prod_{i=1}^{n} h_\beta\left(x^{(i)}\right)^{y^{(i)}} \times \left(1-h_\beta\left(x^{(i)}\right)\right)^{1-y^{(i)}} \end{aligned}$$

Lastly, we find the maximum likelihood estimator β -hat of β =(β_0,β_1,.....β_{19}) which maximizes this likelihood function.For this last task, we need computational abilities that is why we use statistical software like R to perform such tasks.

$$\ell(\beta) = \sum_{i=1}^{n} [y_i log(p_i) + (1-y_i) log(1-p_i)]$$

This is the final equation that the statistical software minimizes by trying the different values of beta to get the maximum likelihood estimator β -hat.

The R code for performing this regression is quite simple. We used the “glm” function for the logistic regression to find the coefficients for all the variables. This analysis is

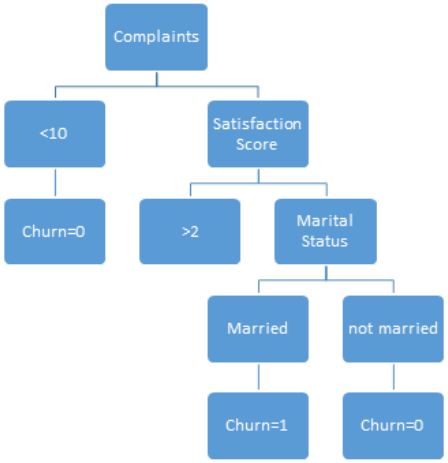
performed on the training dataset, the coefficients we get are then used to predict the outcome variable in the test dataset.

```
#logistic regression
logistic<- glm(Churn ~.,data=train,family=binomial)
summary(logistic)
```

➔ **Random Forest**

In order to understand random forest we need to understand classification trees first. Decision trees classify the examples by sorting them down the tree from the root to some leaf/terminal node, with the leaf/terminal node providing the classification of the example.Each node in the tree acts as a test case for some attribute, and each edge descending from the node corresponds to the possible answers to the test case. This process is recursive in nature and is repeated for every subtree rooted at the new node.

We use recursive binary splitting to grow a classification tree.



Since we cannot assign a RSS value to each class in case of classification decision tree ,the value returned by the decision tree is the most frequently occurring value in that region.For example for complaints <10 the most frequently occurring class in the region is Churn=0.Classification error rate is calculated using the Gini index as follows:

$$G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk}),$$

Here pmk represents the proportion of training observations in the mth region that are from the kth class.Gini index takes a measure of total variance across the K classes. When building a classification tree, the Gini index is typically used to evaluate the quality of a particular split.
Another important attribute for decision trees is information gain.When we use a node in a decision tree to partition the training instances into smaller subsets the entropy changes. Information gain is a measure of this change in entropy gain. Suppose S is a set of instances, A is an attribute, Sv is the subset of S with A = v, and Values (A) is the set of all possible values of A, then

$$Gain(S, A) = Entropy(S) - \sum_{v \epsilon Values(A)} \frac{|S_v|}{|S|}.Entropy(S_v)$$

The information gain is based on the decrease in entropy after a dataset is split on an attribute. Constructing a decision tree is all about finding the attribute that returns the highest information gain (i.e., the most homogeneous branches).

In a random forest model we build a number forest of decision trees on bootstrapped training samples. But when building these decision trees, each time a split in a tree is considered, a random sample of m predictors is chosen as split candidates from the full set of p predictors. The split is allowed to use only one of those m predictors. A fresh sample of m predictors is taken at each split, and typically we choose m ≈ √p—that is, the number of predictors considered at each split is approximately equal to the square root of the total number of predictors

In this model we have used the “randomForest” function in R to predict the values of the dependent variable(churn) using different attributes(factors) that were given to us.After finding the predicted values we compared these predictors to the actual values in test dataset to find the accuracy of the model.

```
##random forest

rf_random<-randomForest(Churn~.,data=train,importance=TRUE)

print(rf_random)
```

```
##
## Call:
##  randomForest(formula = Churn ~ ., data = train, importance = TRUE)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 4
##
##         OOB estimate of  error rate: 2.98%
## Confusion matrix:
##      0   1 class.error
## 0 3720  37 0.009848283
## 1   97 650 0.129852744
```

➔ **Accuracy**

We calculated the accuracy of the two models by subtracting each of the actual values from its predicted value ,taking the absolute value of it and then taking the average of this over the whole test dataset.

Results and Conclusion

The results of the logistic regression have been attached in the appendix.We find that the factors such as preferred payment mode ,satisfaction preferred login device have high value coefficients and also are significant. One important thing to notice is that R has created new dummy variables for factor variables for each of the categories for every factor variable. Using this model we try to predict the outcome variable in the test dataset and then based on that prediction calculate the accuracy. We found the accuracy of the random forest model to be 97.06% and the accuracy of the logistic regression model to be 89.7%.

Using the logistic regression model we can also determine which factors affect the churn dummy the most by looking at the coefficient and the significance level of that coefficient. This means the e-commerce firm should focus more of their revenue and effort on these variables in order to become more profitable.

References

https://www.kaggle.com/ankitverma2010/ecommerce-customer-churn-analysis-and-prediction **- Link to the dataset**

Introduction To Statistical Learning - James,Wittten

https://www.hackerearth.com/practice/machine-learning/machine-learning-algorithms/logistic-regression-analysis-r/tutorial/

http://www.sthda.com/english/articles/36-classification-methods-essentials/151-logistic-regression-essentials-in-r/

https://towardsdatascience.com/understanding-random-forest-58381e0602d2

https://www.youtube.com/watch?v=pBrv2t4JQgQ

https://builtin.com/data-science/random-forest-algorithm

https://www.geeksforgeeks.org/decision-tree-introduction-example/#:~:text=Lets%20consider%20the%20dataset%20in,decision%20tree%20using%20gini%20index.&text=In%20the%20dataset%20above%20there,proportion%20for%20both%20the%20classes.

https://www.saedsayad.com/decision_tree.htm#:~:text=The%20information%20gain%20is%20based,%2C%20the%20most%20homogeneous%20branches

Github repository-https://github.com/deepanshu-web/MAT394_Group2

Appendix

Variables type information

```
## tibble [5,630 x 20] (S3: tbl_df/tbl/data.frame)
##  $ CustomerID                : Factor w/ 5630 levels "50001","50002",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ Churn                     : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
##  $ Tenure                    : num [1:5630] 4 9 9 0 0 0 9 9 13 9 ...
##  $ PreferredLoginDevice      : Factor w/ 3 levels "Computer","Mobile Phone",..: 2 3 3 3 3 1 3 3 3 3 ...
##  $ CityTier                  : Factor w/ 3 levels "1","2","3": 3 1 1 3 1 1 3 1 3 1 ...
##  $ WarehouseToHome           : num [1:5630] 6 8 30 15 12 22 11 6 9 31 ...
##  $ PreferredPaymentMode      : Factor w/ 7 levels "Cash on Delivery",..: 5 7 5 5 2 5 1 2 6 5 ...
##  $ Gender                    : Factor w/ 2 levels "Female","Male": 1 2 2 2 2 1 2 2 2 2 ...
##  $ HourSpendOnApp            : num [1:5630] 3 3 2 2 3 3 2 3 3 2 ...
##  $ NumberOfDeviceRegistered  : num [1:5630] 3 4 4 4 3 5 3 3 4 5 ...
##  $ PreferedOrderCat          : Factor w/ 6 levels "Fashion","Grocery",..: 3 4 4 3 4 5 3 4 4 4 ...
##  $ SatisfactionScore         : Factor w/ 5 levels "1","2","3","4",..: 2 3 3 5 5 5 2 2 3 3 ...
##  $ MaritalStatus             : Factor w/ 3 levels "Divorced","Married",..: 3 3 3 3 3 3 1 1 1 3 ...
##  $ NumberOfAddress           : num [1:5630] 9 7 6 8 3 2 4 3 2 2 ...
##  $ Complain                  : Factor w/ 2 levels "0","1": 2 2 2 1 1 2 1 2 2 1 ...
##  $ OrderAmountHikeFromlastYear: num [1:5630] 11 15 14 23 11 22 14 16 14 12 ...
##  $ CouponUsed                : num [1:5630] 1 0 0 0 1 4 0 2 0 1 ...
##  $ OrderCount                : num [1:5630] 1 1 1 1 1 6 1 2 1 1 ...
##  $ DaySinceLastOrder         : num [1:5630] 5 0 3 3 3 7 0 0 2 1 ...
##  $ CashbackAmount            : num [1:5630] 160 121 120 134 130 ...
```

Logistic regression summary

```
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2199  -0.4381  -0.2056  -0.0652   3.6137
##
## Coefficients:
##                                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)                     -0.846729   0.805343  -1.051 0.293080
## Tenure                          -0.202454   0.011334 -17.863  < 2e-16 ***
## PreferredLoginDeviceMobile Phone -0.517118   0.133970  -3.860 0.000113 ***
## PreferredLoginDevicePhone       -0.598734   0.153301  -3.906 9.40e-05 ***
## CityTier2                        1.188944   0.270996   4.387 1.15e-05 ***
## CityTier3                        0.793193   0.139657   5.680 1.35e-08 ***
## WarehouseToHome                  0.040923   0.006148   6.657 2.80e-11 ***
## PreferredPaymentModeCC          -0.917104   0.432238  -2.122 0.033858 *
## PreferredPaymentModeCOD          0.030761   0.405430   0.076 0.939520
## PreferredPaymentModeCredit Card -0.685961   0.376275  -1.823 0.068299 .
## PreferredPaymentModeDebit Card  -0.523669   0.367678  -1.424 0.154371
## PreferredPaymentModeE wallet     0.132647   0.395299   0.336 0.737201
## PreferredPaymentModeUPI         -0.905431   0.421084  -2.150 0.031536 *
## GenderMale                       0.396635   0.113641   3.490 0.000483 ***
## HourSpendOnApp                  -0.071896   0.091150  -0.789 0.430252
## NumberOfDeviceRegistered         0.449040   0.061500   7.301 2.85e-13 ***
## PreferedOrderCatGrocery          0.529896   0.409334   1.295 0.195481
## PreferedOrderCatLaptop & Accessory -1.574708   0.225526  -6.982 2.90e-12 ***
## PreferedOrderCatMobile          -0.082017   0.332468  -0.247 0.805146
## PreferedOrderCatMobile Phone    -0.659492   0.274849  -2.399 0.016419 *
## PreferedOrderCatOthers           2.969402   0.476869   6.227 4.76e-10 ***
## SatisfactionScore2              -0.043949   0.239556  -0.183 0.854437
## SatisfactionScore3               0.755600   0.165700   4.560 5.11e-06 ***
## SatisfactionScore4               0.521500   0.188526   2.766 0.005671 **
## SatisfactionScore5               1.244406   0.178013   6.991 2.74e-12 ***
## MaritalStatusMarried            -0.461030   0.165980  -2.778 0.005476 **
## MaritalStatusSingle              0.586438   0.165293   3.548 0.000388 ***
## NumberOfAddress                  0.248225   0.021602  11.491  < 2e-16 ***
## Complain1                        1.822495   0.114304  15.944  < 2e-16 ***
## OrderAmountHikeFromlastYear     -0.022107   0.015270  -1.448 0.147670
## CouponUsed                       0.045609   0.039196   1.164 0.244588
## OrderCount                       0.143257   0.028853   4.965 6.87e-07 ***
## DaySinceLastOrder               -0.113427   0.020882  -5.432 5.58e-08 ***
## CashbackAmount                  -0.016321   0.003255  -5.014 5.34e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3805.6  on 4221  degrees of freedom
## Residual deviance: 2307.4  on 4188  degrees of freedom
## AIC: 2375.4
##
```