**CHURN PREDICTION USING CLASSIFICATION TECHNIQUES**

**GITHUB LINK: (**[**https://github.com/Ramu0515/churn-prediction.git**](https://github.com/Ramu0515/churn-prediction.git)**)**

**TEAM:**

1. Ramanjaneyulu Reddy BayapuReddy Lakshmi Reddy

2. Sandeep Bellary

3. Mohammad Turab Ali

**Overview of project, tasks involved and approach, steps implemented, and final products and results:**

The objective of this project is “Churn Prediction using logistic regression” i.e evaluating a telecom customer’s satisfaction and identifying whether he/she is willing to switch to another network. This analysis is very important for a telecom company to ‘know its customers’. If we find a customer is not satisfied with the network, we can give him better offers for stopping him to switch network. This analysis will also help us to understand the customer and suggest him/her best plans to make sure he/she is happy with network.

Tasks involved:

We wrote the code for logistic regression from scratch in which we used gradient descent to find the best possible values for the weights. Significant amount of time was put in to tackling different errors like pickling error, dimension mismatch error…etc that were arising during this process . But we were finally able to solve all the issues and produce a working model for the logistic regression.

We used in built libraries to apply naïve bayes classification algorithm on our dataset. Not many significant difficulties were faced during this process as the libraries handled most of the logical work involved.

The important observations that caught our eye were that naïve bayes performed better than logistic regression which was soothing as we know that naïve bayes performs better than logistic when the data is not very very huge.

The other important observation that was made was that the spark program that we have written for logistic was actually taking(2 sec for each gradient descent ) more time to execute than a normal python program. We also observed that even though naïve bayes performed than logistic but the time taken by it was nearly two times more than logistic.

Initially we have read the data in csv file to python using Pandas library and preprocessed it. During preprocessing we have deleted null value records and invalid records and converted text field columns to numeric values, so that we can use it for our classification.

The we have generated several plots and analyzed the data, the plots helped us in identifying the distribution of data , finding outliers, finding error values etc. below are some of the plots we have generated out of the actual data.

IMAGES

Feature selection:

For selecting suitable features for prediction, we have initially gone through all the features and listed out some of the important features that may affect the predicted value

Then we have calculated the correlation co-efficients for all the features and selected three columns which are common in both the lists and used them for prediction.

The below are the three features which we have selected for prediction

* rev\_Mean : mean of monthly revenue obtained from the customer
* mou\_Mean : mean of minutes of usage
* totmrc\_Mean : mean of total mrc

**Context and motivation for project, stating what is interesting/important/useful/fun about it:**

The main source of revenue for a telecom company is the revenue from its customers who are buying plans from the company. The plans include internet plans, call plan, international call packages, roaming packages etc. So, we can conclude that the revenue for company is the revenue obtained from every one of their customers, losing a customer means losing a part of the company’s revenue. So, it is very important for the company to keep track of customer’s satisfaction towards network. The context of our project is to calculate the satisfaction index of customers using services from a telecom company. This is very important because losing a customer will result in a loss of revenue the company is obtaining from that customer permanently(in most of the cases). So, it is better to implement preventive measures to help a customer find plans which are useful for him, so that the customer will be happy with the service and will stay in network and will not switch.

**Data set(s) used for project and how they were obtained, size of data, and features used:** <https://www.kaggle.com/abhinav89/telecom-customer/download>

We have found the above dataset in Kaggle. The dataset contains over one lakh records with nearly 100 attributes. Each record indicate a customer. The details include, the customer ID, the total revenue obtained from the customer, number of minutes he/she used, his current plan, number of people using the network in his family, last recharge cost, data usage units etc. Few of these attributes are used to predict whether or not a customer will churn.

**What algorithms and techniques were implemented, and frameworks (e.g., Hadoop, Spark) used?**

We have implemented Logistic Regression using python’s numpy library and Naïve Bayes algorithm using spark mLlib library.

Frameworks: Apache Spark.

**Illustrative results and examples and Performance evaluation, preferably quantitative:**

**Logistic Regression:**

Average time taken for each iteration:2 sec

Accuracy:50.46983

Further enhancements : Feature selection could be made more precise, not considering the complexity of the features one hot encoding could be used.

|  |  |  |
| --- | --- | --- |
| Logistic regression | Churn=0 | Churn=1 |
| Actual=0 | 11168 | 9848 |
| Actual=1 | 8758 | 9836 |

Naïve Bayes:

Average time taken for each iteration:3.5 seconds

Accuracy:54.3666

Further enhancements : Feature selection could be made more precise, not considering the complexity of the features one hot encoding could be used.

|  |  |  |
| --- | --- | --- |
| NaiveBayes | Churn=0 | Churn=1 |
| Actual=0 | 8855 | 13741 |
| Actual=1 | 4782 | 7558 |

**What aspects of the definitely will do, likely will do, and would ideally like to do items did you accomplish?**

WE have completely achieved all the aspects of the

a)Definitely will do

We performed data preprocessing on the input dataset, meaning we removed the empty and invalid records from the data. We converted any text value fields in the data to numeric in our calculation of the satisfactory index. First, we split the input data into train data and test data. 80 percent of the input data as train data is used to train the machine learning algorithm and remaining 20percent of the test data is used to test the accuracy of the algorithm, to make sure it is providing accurate results or not.

b)Likely will do

we designed a system which uses the rating given by a customer to predict if he is going to stay on this network or not and depending upon this rating we designed business techniques that can make the customer stay on our network. We used spark for parallelizing of our implementation of logistic regression .We performed classification using other classifiers using predefined libraries and then compare the performance of logistic regression with other classification algorithms (Naïve Bayes classifier)

and we tried our best to achieve the aspects of

c)Would ideally like to do.

We tried to implement the whole algorithm (Naïve Bayes classifier) without using predefined libraries so that we can get a clear picture of the internals of this algorithm.

**Work division among team members:**

Each of the team member has collectively worked on the whole project and tried collectively to optimize the performance and achieve the results.

A) Ramanjaneyulu Reddy BayapuReddy Lakshmi Reddy (student id: 801101005):

Analyze data set and splitting of data for training and testing. Implement naïve bayes algorithm on the dataset using Spark MLlib library.

B) Sandeep Bellary (student id: 801101004):

Preprocessing Build the model for logistic regression Illustrate the results.

C) Mohammad Turab Ali (student id: 80

1096281):

Implement naïve bayes algorithm on the data set and compare the performance of logistic regression and naïve bayes algorithms.

**Any important references that you used:**

We have used Apache Spark documentation on Logistic Regression and Naive Bayes Algorithms as the main reference points while implementing our project.

1)<https://spark.apache.org/docs/1.6.2/mllib-guide.html>

2)