**Customer Churn Prediction using Machine Learning**

Ayesha Ali (657067448)

Ridhima Mehta (653499720)

Shubham Chaudhary(658885025)

**1. Problem Setting**

We are trying to predict when a customer might churn for a music company. This prediction is essential since it helps companies know what necessary changes, they need to make to ensure their customers stay. Since every individual is different and may also have different reasons in churning, it is essential for companies to be able to predict when and why an individual might churn. By accurately predicting when an individual churns, it will allow businesses to improve in certain areas and increase their revenue. However, it is most relevant in membership-based businesses where there is a certain fee required over a specific amount of time.

**2. Data Description**

We have used the dataset from the Udacity website that is present in JSON format for data analysis and to gain an understanding of the customers. It has total 2 million columns out of which we are using 100,000 rows for our analysis. The dataset is divided into 3 main categories: User information – userID, FirstName, Last Name, Gender, Location, User agent, Registration. Log Specific Information – Timestamp, Page, Authentication Level, Session ID, itemInSession and Artist Details: Artist Name, Genre and Song

**3. Techniques**

We have used the following methodology in the project to predict the churn rate of the customers:

* **Exploratory Data Analysis:**

1. In this we have checked the dataset for any missing values and dropped those missing values from the dataset for each column we didn’t find any null values.
2. There are total 19 attributes in the dataset which we are using for our analysis
3. Since we don’t have a defined churn variable, we have created the variable for the users that have cancelled their subscriptions.
4. We have a lot of repeating user IDs in our dataset which we have aggregated so that it is easy to find the churn rate of the customers.
5. We are then performing analysis to see the churn rate by gender, age, location, artists categories etc. which we will help us in the analysis of selection of features
6. Customer Churn Rate by Gender, Artists, Location and Categories.

Churn rate by Gender

Text

Description automatically generated with low confidence

Top 5 artists possible cause for churn rate of customers:

Text

Description automatically generated

Table

Description automatically generated

Churn rate based on customer’s behavior

A picture containing chart

Description automatically generated

A picture containing diagram

Description automatically generated

Churn rate by time- depending on the day, month, and hour.

Chart, bar chart

Description automatically generated

Chart, bar chart

Description automatically generated

Chart, bar chart

Description automatically generated

* **Feature Selection:**

1. From the different attributes we are going to select the attributes where we see a considerable difference in the proportion of churners and non-churners
2. For this we create functions where we compare the customers Ids churned and non-churned based on their time of registration, songs played, gender and artists.
3. From the total 19 attributes we select the 10 attributes which we use for our analysis.
4. These 10 attributes are :
5. timestamp(ts) – time since registration of the customer
6. song – songs played by churned and non-churned customers
7. thumps up – it is selected by customers when they like a song
8. thumps down – when customers dislike a song they choose it
9. length – duration of songs heard by the customers
10. Gender – Avg churn rate of male gender is higher than female
11. Artists – Customers dislike the songs with no artist name
12. sessionID – helps in understanding the type of songs played
13. Add\_friend – When customers like any song they can recommend to friends
14. Add\_to\_playlist – When customers like any song they add it in their playlist

Non-churned customers listened to more songs compared to churned customers

Table

Description automatically generated

Non-churned customers had many thumps up compared to churned customers

Table

Description automatically generated

Non-churned customers had many thumps down compared to churned customers

Table

Description automatically generated

Customer Lifetime is the no of days the customer has been since they registered – we see that the churned customers stayed the shortest compared to non-churned customers

A screenshot of a computer

Description automatically generated with low confidence

Non-churned customers on average added more friends than churned customers

Table

Description automatically generated

Non-churned customers listened to more songs than churned customers

Table

Description automatically generated

For the 10 features we are then combining them into a data frame which we use for our model building using outer join

Table

Description automatically generated

* **Model development:**

1. For predicting the churn rate of the customers, we have developed two supervised learning models – Gradient Boosting and Logistic Regression
2. Before developing our model, we split the dataset into training, testing and validation set.
3. Since the churn label is unbalanced as we have higher customer that are non-churners compared to churners we have developed two base models for our reference.
4. We use vector assembler scale the features using normalization to avoid bias in our model
5. We then compare the accuracy of the models and the F-score of the models to determine our best model based on the validation dataset.
6. After determining our best model we then fine tune the parameters to get a higher accuracy score on the test dataset and use it derive the conclusions of our analysis.
7. Gradient Boosting model:
   1. We have used cross validation in our model to avoid the case of overfitting
   2. We are using the GBTClassifier function with max iterations as 5 and seed = 42
   3. MulticlassClassificationEvaluator : This is used for binary evaluation of the class
   4. Accuracy comes around 60% with F-score of 0.57 without hyperparameter tuning
   5. For fine tuning our model we are using the grid seach function: paramgridbuilder

# initialize classifier

GradBoostTree = GBTClassifier(maxIter=5,seed=42)

# set evaluator

f1\_evaluator = MulticlassClassificationEvaluator(metricName='f1')

# build paramGrid

paramGrid = ParamGridBuilder().build()

crossval\_GradBoostTree = CrossValidator(estimator=GradBoostTree,

                          estimatorParamMaps=paramGrid,

                          evaluator=f1\_evaluator,

                          numFolds=5)

cvModel\_GradBoostTree = crossval\_GradBoostTree.fit(train)

cvModel\_GradBoostTree.avgMetrics

results\_GradBoostTree = cvModel\_GradBoostTree.transform(validation)

evaluator = MulticlassClassificationEvaluator(predictionCol="prediction")

Logistic Regression Model:

* We have use the multi classification evaluator as we are predicting a binary class of variable
* We have used cross validation to avoid the case of overfitting in our model
* We are using the function LogisticRegression in spark to initiate the model
* For fine tuning our machine learning model we are using the grid search algorithm ParamGridBuilder() to get a higher accuracy score.
* We find the accuracy of the model to be 70% on the validation dataset and F-score as 0.60
* We have used the following code in our model:

lr = LogisticRegression(maxIter=10)

f1\_evaluator = MulticlassClassificationEvaluator(metricName='f1')

paramGrid = ParamGridBuilder() \

    .build()

crossval\_lr = CrossValidator(estimator=lr,

                          evaluator=f1\_evaluator,

                          estimatorParamMaps=paramGrid,

                          numFolds=3)

cvModel\_lr = crossval\_lr.fit(train)

cvModel\_lr.avgMetrics

results\_lr = cvModel\_lr.transform(validation)

evaluator = MulticlassClassificationEvaluator(predictionCol="prediction")

Based on the above two models accuracy we find that Logistic Regression is our best model because of high accuracy and F-score on the validation data set.

Text

Description automatically generated Text

Description automatically generated with low confidence

We select the best model to be Logistic Regression and perform hyperparameter tuning to improve the accuracy on the test dataset and get the high accuracy of the overall model.

After performing hyperparameter tuning we find out the accuracy to be:

Text

Description automatically generated

We then perform the feature importance of the 10 different features and find their feature ranking to derive our conclusion.

**4. Results**

From the above prediction on the test dataset, we find out that Logistic Regression to be our best model with accuracy of 78%. After performing the feature importance, we find out that the top 3 features FriendsAdd, Customer Lifetime and Thumps down page are crucial in determining the churn rate of the customers. We also find out that most of the customers that have churned are due to poor quality of songs, songs recommended to them by their friends and customers that have used the app for shorter period that churned due to the quality of the service offered by the company.

**5. Role of team members in the project**

|  |  |  |
| --- | --- | --- |
| **Role** | **Description** | **Team Member** |
| Exploratory Data Analysis | Performing EDA on the data dropped the missing values and  created churn variable  Developed graphs to analyze churn rate based on different attributes of the dataset | Ayesha Ali |
| Feature Selection | Selecting relevant features based on the group analysis, time analysis and customer analysis.  Selected the 10 features among the total 19 features that are used for building the model. | Ridhima Mehta |
| Model Development | Developed the models Gradient boosting and Logistic Regression  and selected the best model.  Performed Hyperparameter tuning on the test dataset to select the best model. | Shubham Chaudhary |
| Result | Model’s prediction and feature importance have been used in determining the reasons for customer churn rate | All the members |