CUSTOMER CHURN PREDICTION

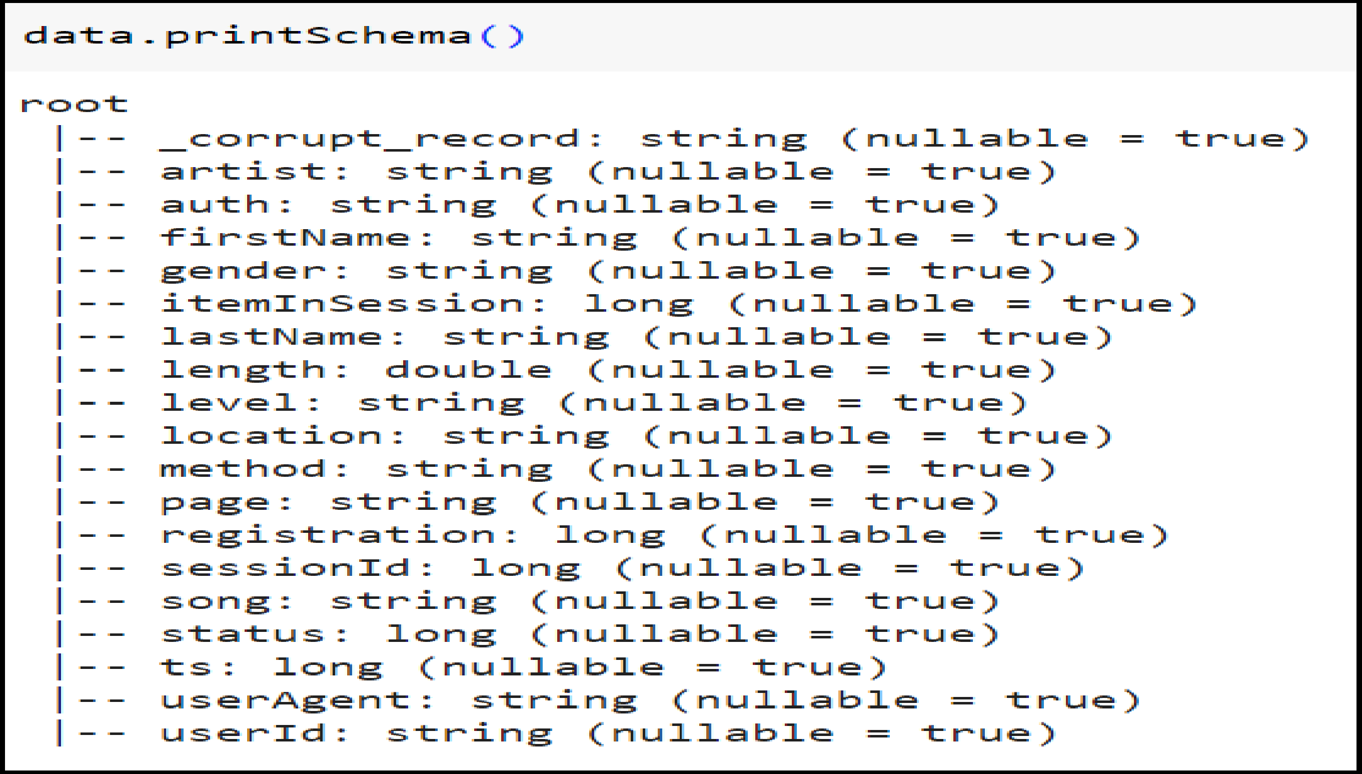
Prachi Mehta (202318008)
Dhruvi Mehta (202318003)
Simran Dalvi (202318042)

PROBLEM STATEMENT

- Acquiring new customer is more costly than retaining current customer
- Goal: In this project, we aim to leverage PySpark on the Sparkify music streaming dataset to construct a predictive model for anticipating customer churn. By analyzing user interactions including page visits, likes, dislikes, and cancellations, the model endeavors to forecast churn behavior accurately. Through this predictive framework, we seek to provide insights into user attrition patterns, enabling proactive measures to retain customers and enhance the overall user experience within the Sparkify platform.

DATA

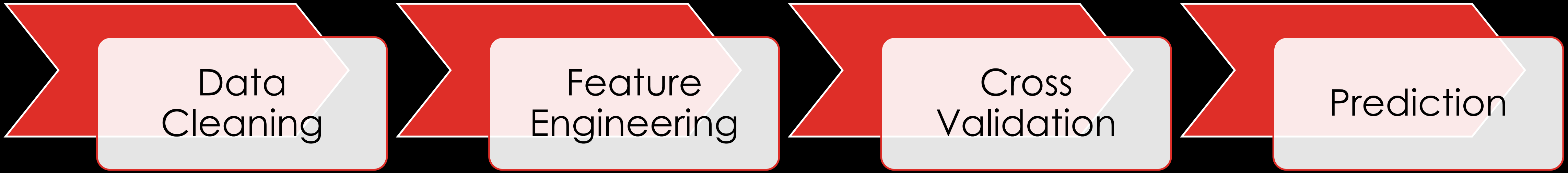
- Sparkify is an imaginary digital music service similar to Spotify.
- The dataset contain 12GB of user interactions with this service.



```
data.printSchema()
```

```
root
 |-- _corrupt_record: string (nullable = true)
 |-- artist: string (nullable = true)
 |-- auth: string (nullable = true)
 |-- firstName: string (nullable = true)
 |-- gender: string (nullable = true)
 |-- itemInSession: long (nullable = true)
 |-- lastName: string (nullable = true)
 |-- length: double (nullable = true)
 |-- level: string (nullable = true)
 |-- location: string (nullable = true)
 |-- method: string (nullable = true)
 |-- page: string (nullable = true)
 |-- registration: long (nullable = true)
 |-- sessionId: long (nullable = true)
 |-- song: string (nullable = true)
 |-- status: long (nullable = true)
 |-- ts: long (nullable = true)
 |-- userAgent: string (nullable = true)
 |-- userId: string (nullable = true)
```

METHODOLOGY



Data Cleaning

Remove null values from various columns

Feature Engineering

Extract features from user log data

Cross Validation

User grid search cross validation for hyper-parameter tuning

Prediction

Test the model using 20% of the dataset

DATA PREPROCESSING

Data selection

Columns that were not significant to the modelling process will be dropped

- Firstname
- Lastname
- Id_copy

userID was retained as it was used for feature engineering step.

Unit conversion

Registration and TS were given in milliseconds.
These fields were converted to seconds by dividing the values by 1000.

Create churn label

Dataset only contains user log data used page column to identify churners:

- Visiting cancellation confirmation page indicated a churned user
- Creating a label column where 1 indicates a churned user and 0 indicated otherwise

FEATURE ENGINEERING

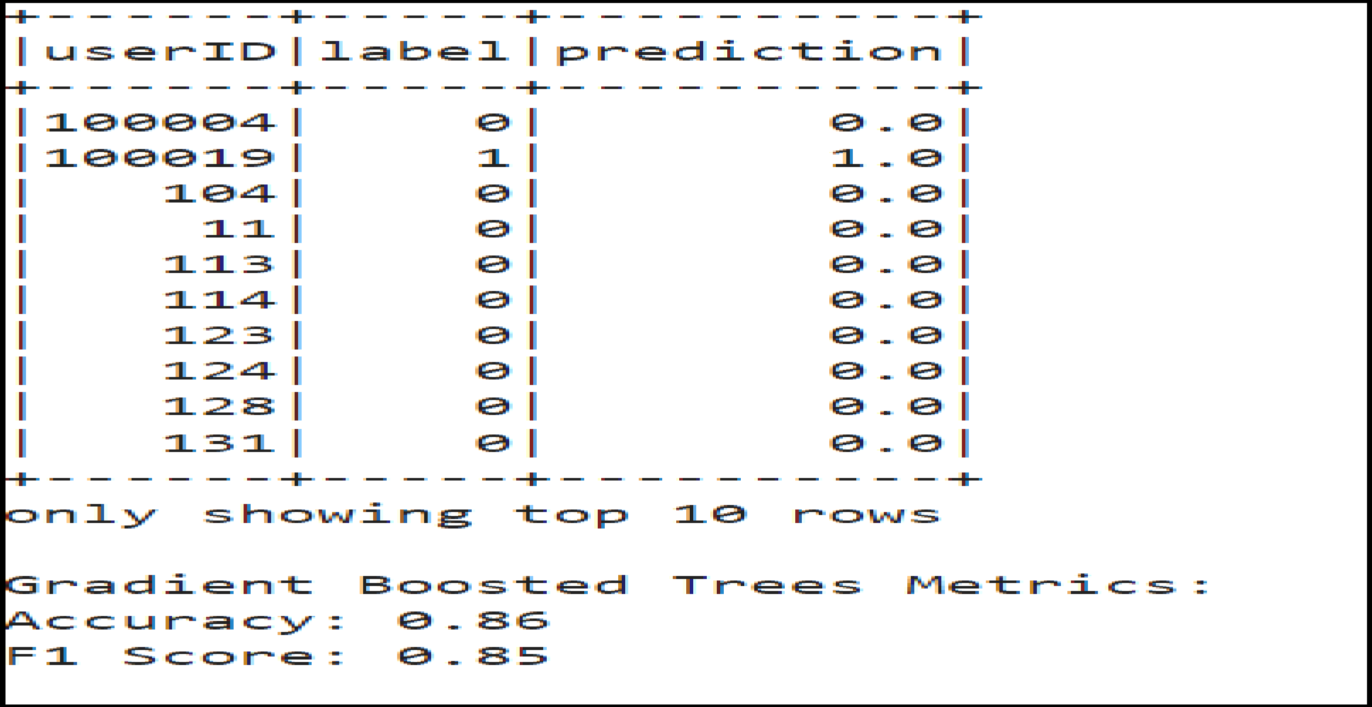
- Meaningful data has to be created from the user log data that could be used by the prediction models.
- The following features were used
  - Time since registration
  - Number of friends referred
  - Total songs listened to
  - Total songs liked
  - Total songs disliked
  - Number of songs in user playlist
  - Average songs played
  - Number of artists listened to
  - Number of user sessions logged
- More features were used initially but discarded after observing less than 1% feature importance during training of models.

MODELLING

- Dataset will be split into 80-20 train test split
- Grid search cross validation with three folds was used to built the following models
  - Gradient boosting trees
  - Random forests
  - Logistic regression
  - Support vector machine
  - Hybrid model

Gradient boosting trees

- Gradient boosting trees is a method in machine learning where multiple decision trees are combined to improve predictions, focusing on correcting errors from previous trees for better accuracy.

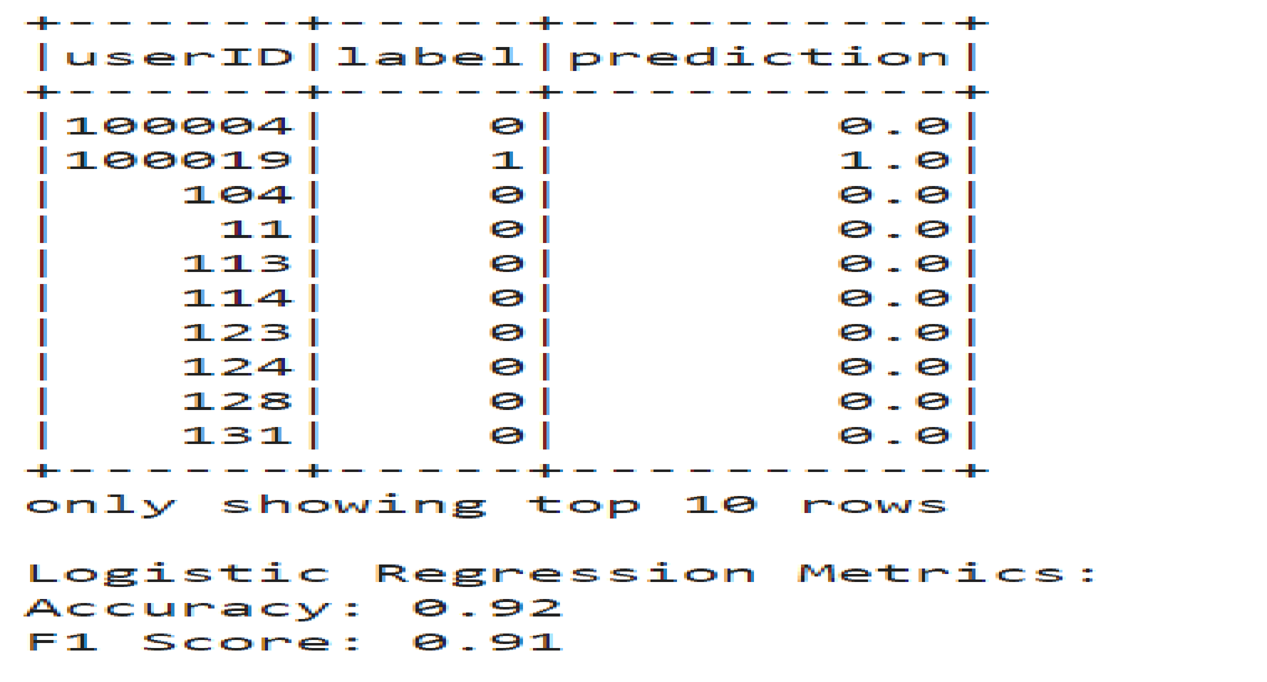


```
+------+-----+----------+
|userID|label|prediction|
+------+-----+----------+
|100004|    0|       0.0|
|100019|    1|       1.0|
|   104|    0|       0.0|
|    11|    0|       0.0|
|   113|    0|       0.0|
|   114|    0|       0.0|
|   123|    0|       0.0|
|   124|    0|       0.0|
|   128|    0|       0.0|
|   131|    0|       0.0|
+------+-----+----------+
only showing top 10 rows

Gradient Boosted Trees Metrics:
Accuracy: 0.86
F1 Score: 0.85
```

LOGISTIC REGRESSION

- Logistic regression is a statistical method used for binary classification, predicting outcomes like whether a user will churn from a subscription service based on given features.

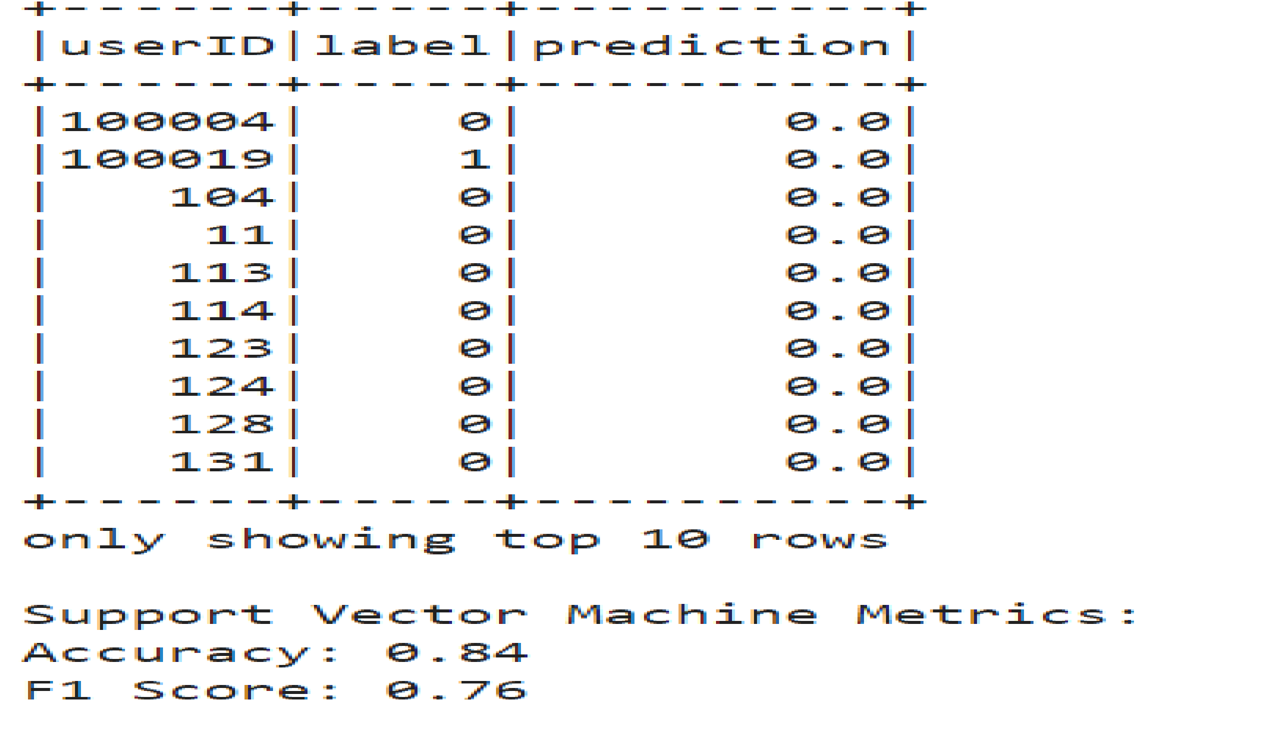


```
+------+-----+----------+
|userID|label|prediction|
+------+-----+----------+
|100004|    0|       0.0|
|100019|    1|       1.0|
|   104|    0|       0.0|
|    11|    0|       0.0|
|   113|    0|       0.0|
|   114|    0|       0.0|
|   123|    0|       0.0|
|   124|    0|       0.0|
|   128|    0|       0.0|
|   131|    0|       0.0|
+------+-----+----------+
only showing top 10 rows

Logistic Regression Metrics:
Accuracy: 0.92
F1 Score: 0.91
```

SUPPORT VECTOR MACHINE (SVM)

- Support Vector Machine (SVM) is a machine learning algorithm for classification tasks, finding the best separation line (or hyperplane) between different classes in the data.

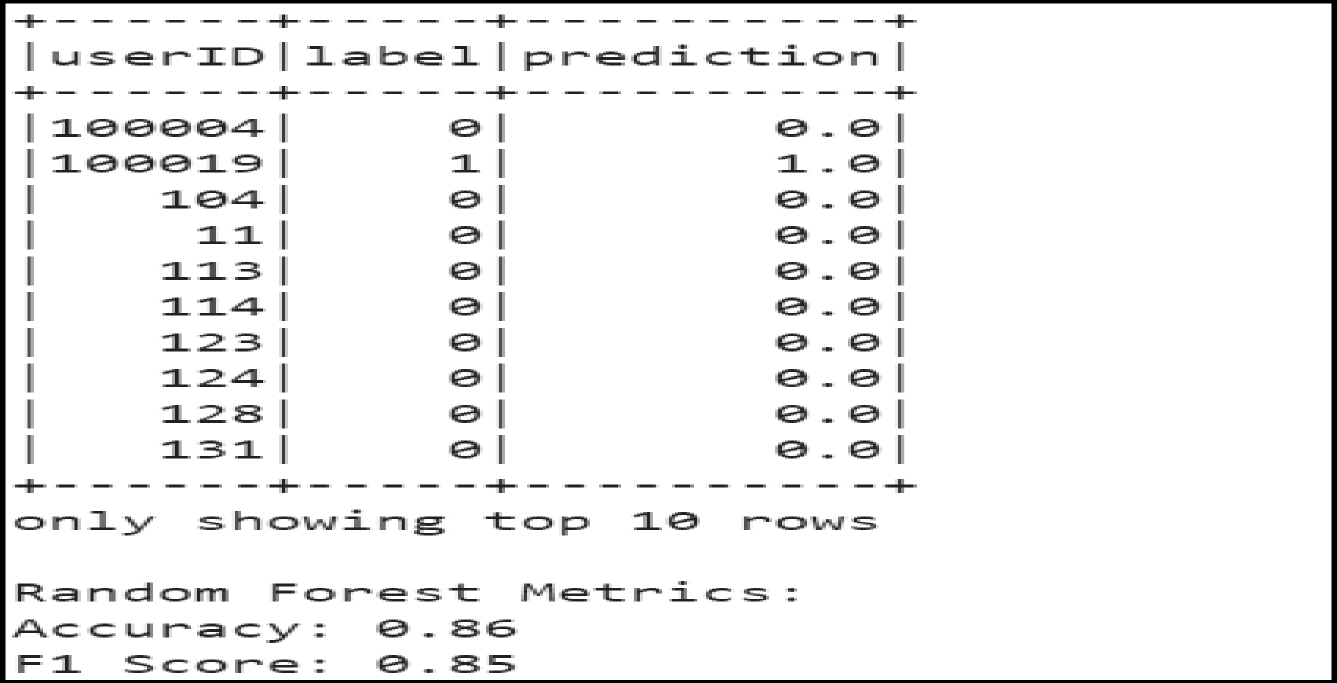


```
+------+-----+----------+
|userID|label|prediction|
+------+-----+----------+
|100004|    0|       0.0|
|100019|    1|       0.0|
|   104|    0|       0.0|
|    11|    0|       0.0|
|   113|    0|       0.0|
|   114|    0|       0.0|
|   123|    0|       0.0|
|   124|    0|       0.0|
|   128|    0|       0.0|
|   131|    0|       0.0|
+------+-----+----------+
only showing top 10 rows

Support Vector Machine Metrics:
Accuracy: 0.84
F1 Score: 0.76
```

RANDOM FOREST CLASSIFIER

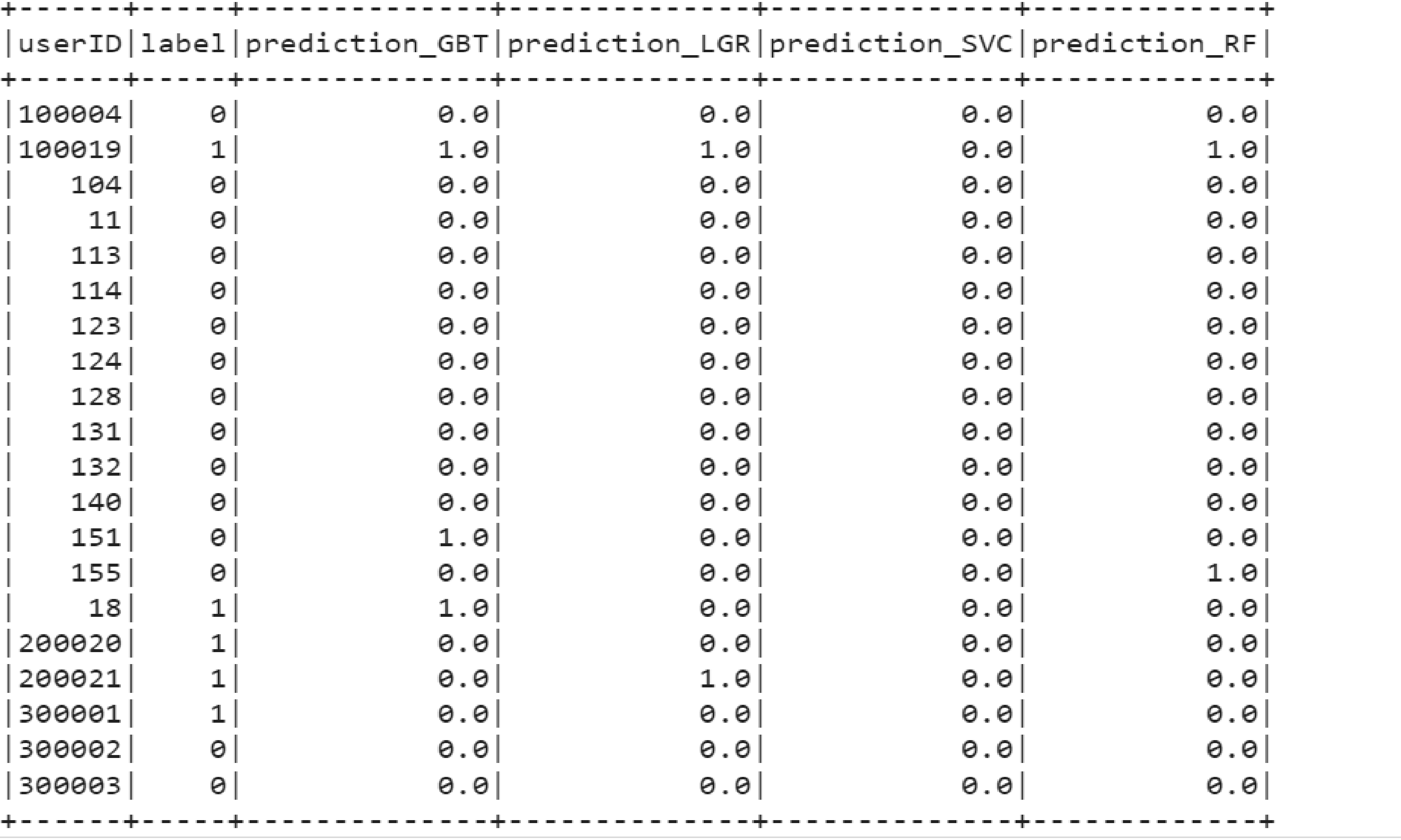
- Random Forest Classifier is Machine learning Model that combines multiple decision trees to improve classification accuracy by voting on the most common prediction.



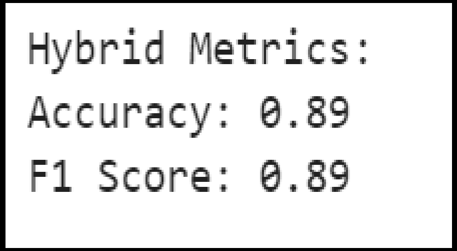
```
+------+-----+----------+
|userID|label|prediction|
+------+-----+----------+
|100004|    0|       0.0|
|100019|    1|       1.0|
|   104|    0|       0.0|
|    11|    0|       0.0|
|   113|    0|       0.0|
|   114|    0|       0.0|
|   123|    0|       0.0|
|   124|    0|       0.0|
|   128|    0|       0.0|
|   131|    0|       0.0|
+------+-----+----------+
only showing top 10 rows

Random Forest Metrics:
Accuracy: 0.86
F1 Score: 0.85
```

HYBRID MODEL



```
+------+-----+--------------+--------------+--------------+-------------+
|userID|label|prediction_GBT|prediction_LGR|prediction_SVC|prediction_RF|
+------+-----+--------------+--------------+--------------+-------------+
|100004|    0|           0.0|           0.0|           0.0|          0.0|
|100019|    1|           1.0|           1.0|           0.0|          1.0|
|   104|    0|           0.0|           0.0|           0.0|          0.0|
|    11|    0|           0.0|           0.0|           0.0|          0.0|
|   113|    0|           0.0|           0.0|           0.0|          0.0|
|   114|    0|           0.0|           0.0|           0.0|          0.0|
|   123|    0|           0.0|           0.0|           0.0|          0.0|
|   124|    0|           0.0|           0.0|           0.0|          0.0|
|   128|    0|           0.0|           0.0|           0.0|          0.0|
|   131|    0|           0.0|           0.0|           0.0|          0.0|
|   132|    0|           0.0|           0.0|           0.0|          0.0|
|   140|    0|           0.0|           0.0|           0.0|          0.0|
|   151|    0|           1.0|           0.0|           0.0|          0.0|
|   155|    0|           0.0|           0.0|           0.0|          1.0|
|    18|    1|           1.0|           0.0|           0.0|          0.0|
|200020|    1|           0.0|           0.0|           0.0|          0.0|
|200021|    1|           0.0|           1.0|           0.0|          0.0|
|300001|    1|           0.0|           0.0|           0.0|          0.0|
|300002|    0|           0.0|           0.0|           0.0|          0.0|
|300003|    0|           0.0|           0.0|           0.0|          0.0|
+------+-----+--------------+--------------+--------------+-------------+
```



```
Hybrid Metrics:
Accuracy: 0.89
F1 Score: 0.89
```

CONCLUSISON

- In conclusion, our project successfully utilized various user interaction features such as time since registration, number of friends referred, total songs listened to, liked, and disliked, along with playlist size, average songs played, number of artists listened to, and user session logs to predict customer churn.
- Through thorough experimentation, logistic regression emerged as the optimal model for churn prediction in our context.
- This model demonstrates promising accuracy and interpretability, providing valuable insights for proactive churn management strategies within the Sparkify music streaming platform.

REFERENCES

- (PDF) Computational Efficiency Analysis of Customer Churn Prediction Using Spark and Caret Random Forest Classifier (researchgate.net)
- Customer churn prediction system: a machine learning approach | Computing (springer.com)

THANK YOU