**PROJECT REPORT**

**On**

**Telecom Churn Prediction**

***Prepared By:***

***Surbhi Dogra***

**Abstract**

Churn is a one of the biggest problem in the telecom industry. This Project is all about predicting the behavior of customers with churn or no churn ,so, we can retain them and develop focused customer retention programs. The analysis has been done by exploring the overall structure of the dataset and summary statistics of each variable. Also, analyzed the number of observations, their data type. Applied feature engineering on the dataset. As the target value is categorical(discrete) so, this is a classification model and I have used Logistic Regression, Random Forest, Gradient Boosted Classification tree in Apache Spark to solve the problem and evaluated them using AUC. At last, variable importance has been found on the basis of the best model.

**Problem definition and project goals**

**Problem:** The churn rate, also known as the rate of attrition or customer churn, is the rate at which customers stop doing business with an entity. It is most commonly expressed as the percentage of service subscribers who discontinue their subscriptions within a given time period.

Churn is a one of the biggest problem in the telecom industry.

In my dataset, which is available on Kagel “<https://www.kaggle.com/blastchar/telco-customer-churn>”, each row represents a customer, each column contains customer’s attributes described on the column Metadata.The raw data contains 7043 rows (customers) and 21 columns (features).The “Churn” column is our target.

The detailed description of data set is below:

customerID

Customer ID

gender

Whether the customer is a male or a female

SeniorCitizen

Whether the customer is a senior citizen or not (1, 0)

Partner

Whether the customer has a partner or not (Yes, No)

Dependents

Whether the customer has dependents or not (Yes, No)

tenure

Number of months the customer has stayed with the company

PhoneService

Whether the customer has a phone service or not (Yes, No)

MultipleLines

Whether the customer has multiple lines or not (Yes, No, No phone service)

InternetService

Customer’s internet service provider (DSL, Fiber optic, No)

OnlineSecurity

Whether the customer has online security or not (Yes, No, No internet service)

OnlineBackup

Whether the customer has online backup or not (Yes, No, No internet service)

DeviceProtection

Whether the customer has device protection or not (Yes, No, No internet service)

TechSupport

Whether the customer has tech support or not (Yes, No, No internet service)

StreamingTV

Whether the customer has streaming TV or not (Yes, No, No internet service)

StreamingMovies

Whether the customer has streaming movies or not (Yes, No, No internet service)

Contract

The contract term of the customer (Month-to-month, One year, Two year)

PaperlessBilling

Whether the customer has paperless billing or not (Yes, No)

PaymentMethod

The customer’s payment method (Electronic check, Mailed check, Bank transfer (automatic), Credit card (automatic))

MonthlyCharges

The amount charged to the customer monthly

TotalCharges

The total amount charged to the customer

Churn

Whether the customer churned or not (Yes or No)

**Goal:**

The main objective is to predict the behavior of customers that is they are churned or not, so we can retain them and develop focused customer retention programs.

**Related Work**

There are other works similar to my project:

1. <https://www.kaggle.com/liyingiris90/telco-customer-churn-prediction>

In this , the dataset is similar to mine and has been downloaded from IBM Sample Data Sets for customer retention programs.

Methods used are:

* Logistic Regression
* Decision tree
* Random forest

Results:

The Logistic Regression Model (with threshold = 0.5) has Accuracy of 0.79 and the AUC is 0.82 for the test set.

The decision tree model has Accuracy of 0.78 and AUC of 0.78 for the test set. It does not perform as good as the logistic regression model.

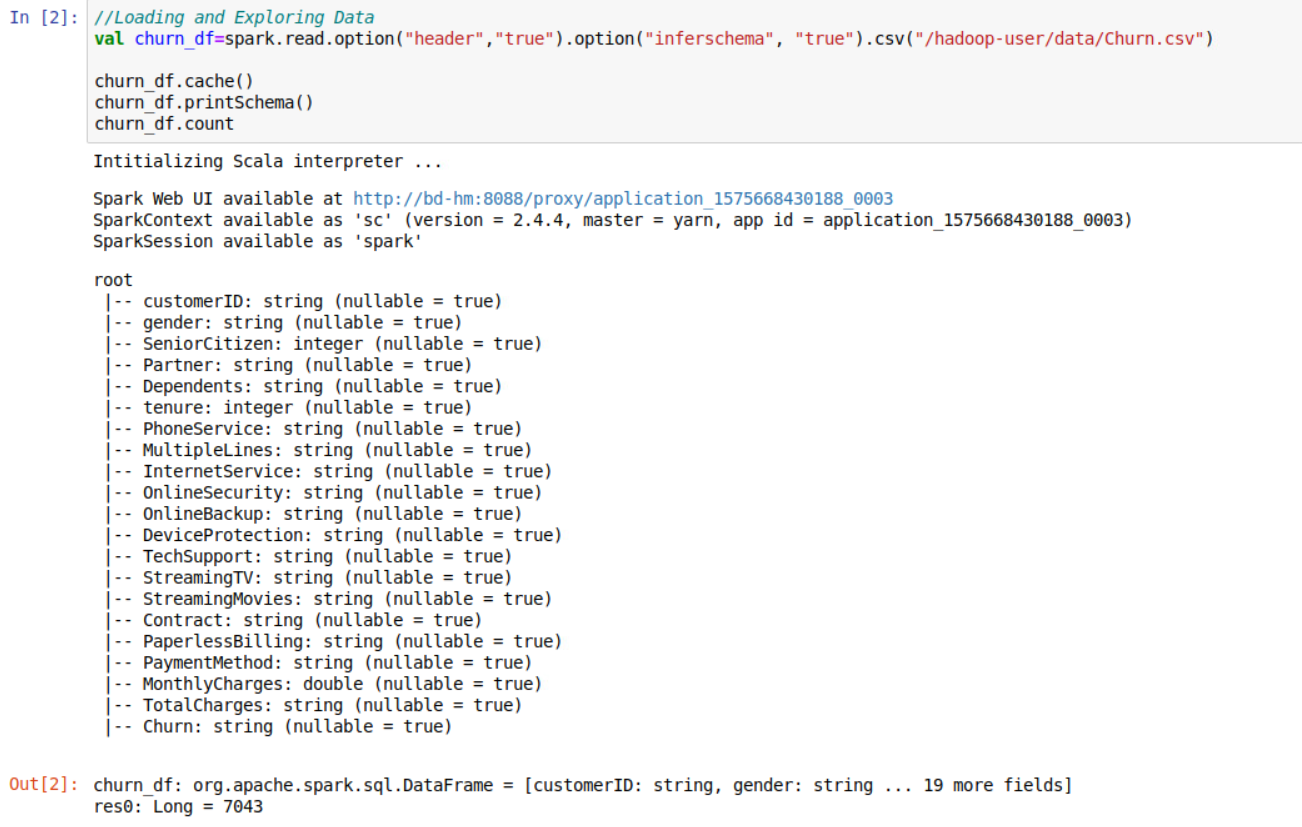
The random forest model has the Accuracy of 0.79 and AUC of 0.82 for the test set.

1. <https://www.kaggle.com/ecosaptarshi/logistic-regression-to-predict-customer-churn>

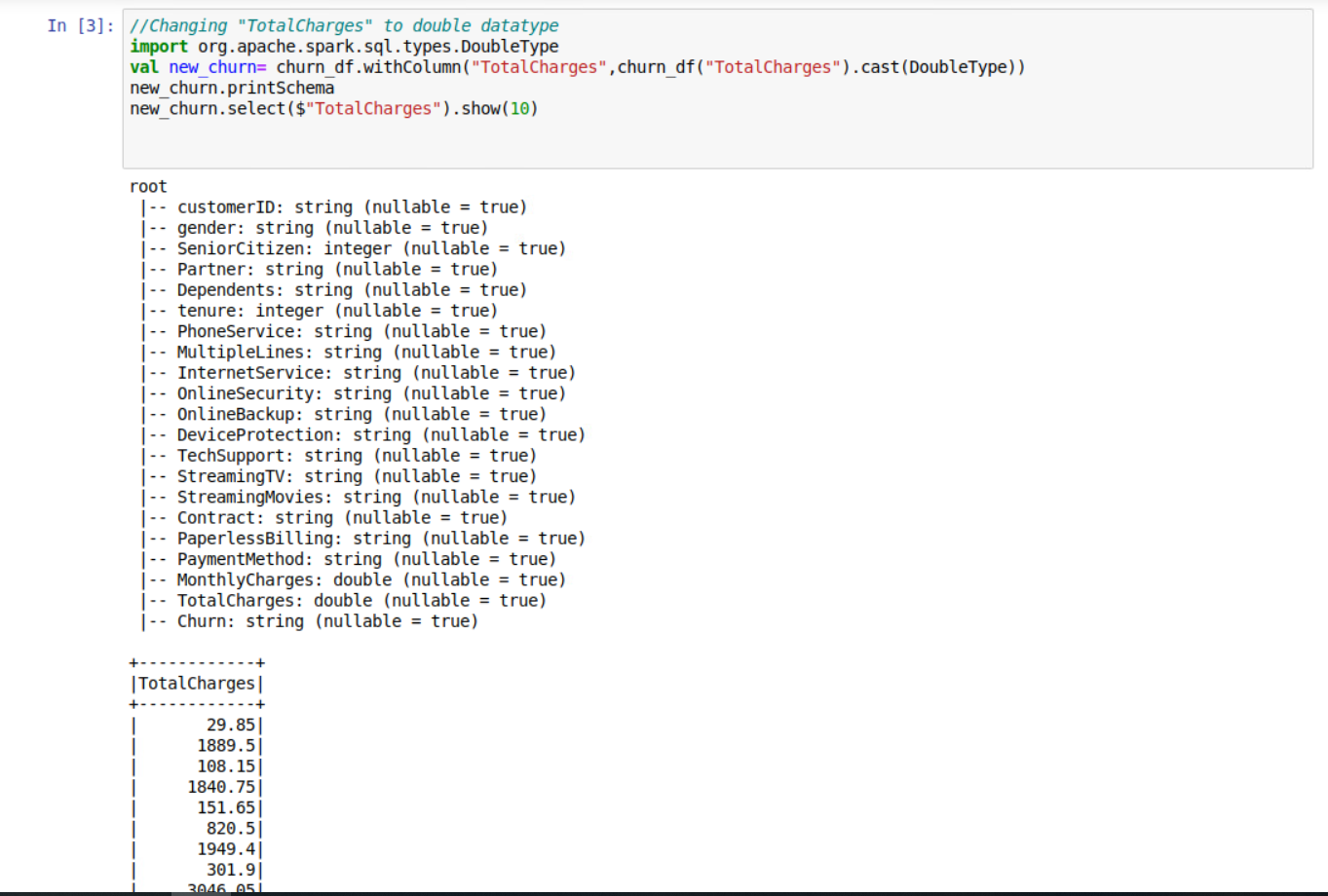
Here , the person has just used the Logistic Regression to predict the target value”Churn”.

The LR model has AUC as 0.803 that is 80.3%.

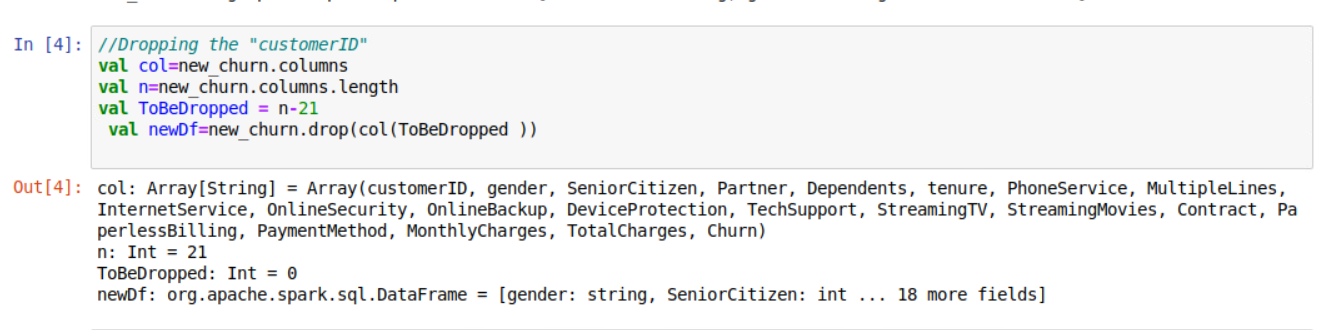
**Data Exploration and preprocessing**

I loaded the dataset first and saw the schema. It can be seen that there are 21 variables and most of them are categorical . 

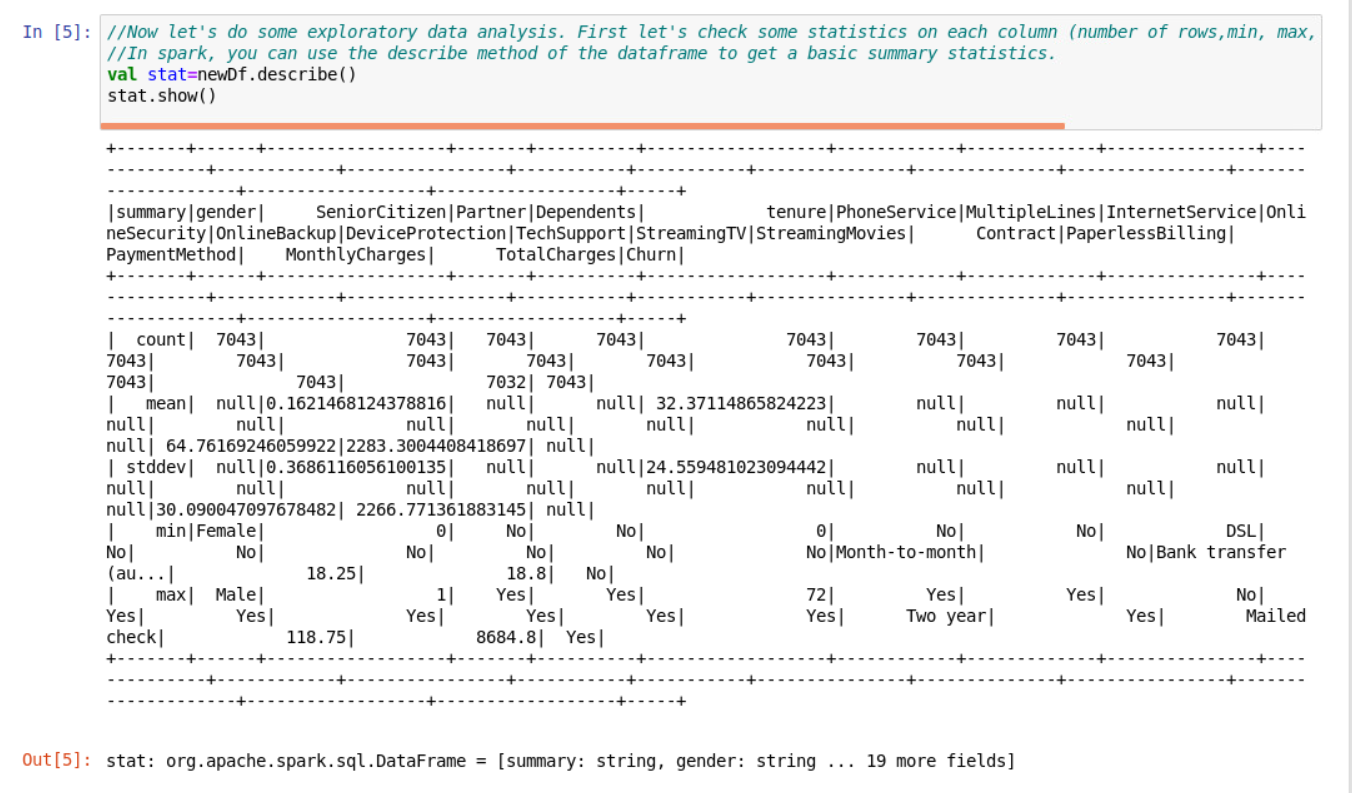
I changed “TotalCharges” to double.



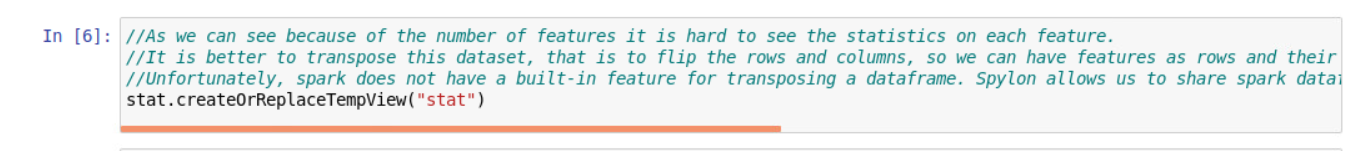
I removed the “CustomerId” variable as it will be unique for every customer and will not help in predicting the target value.

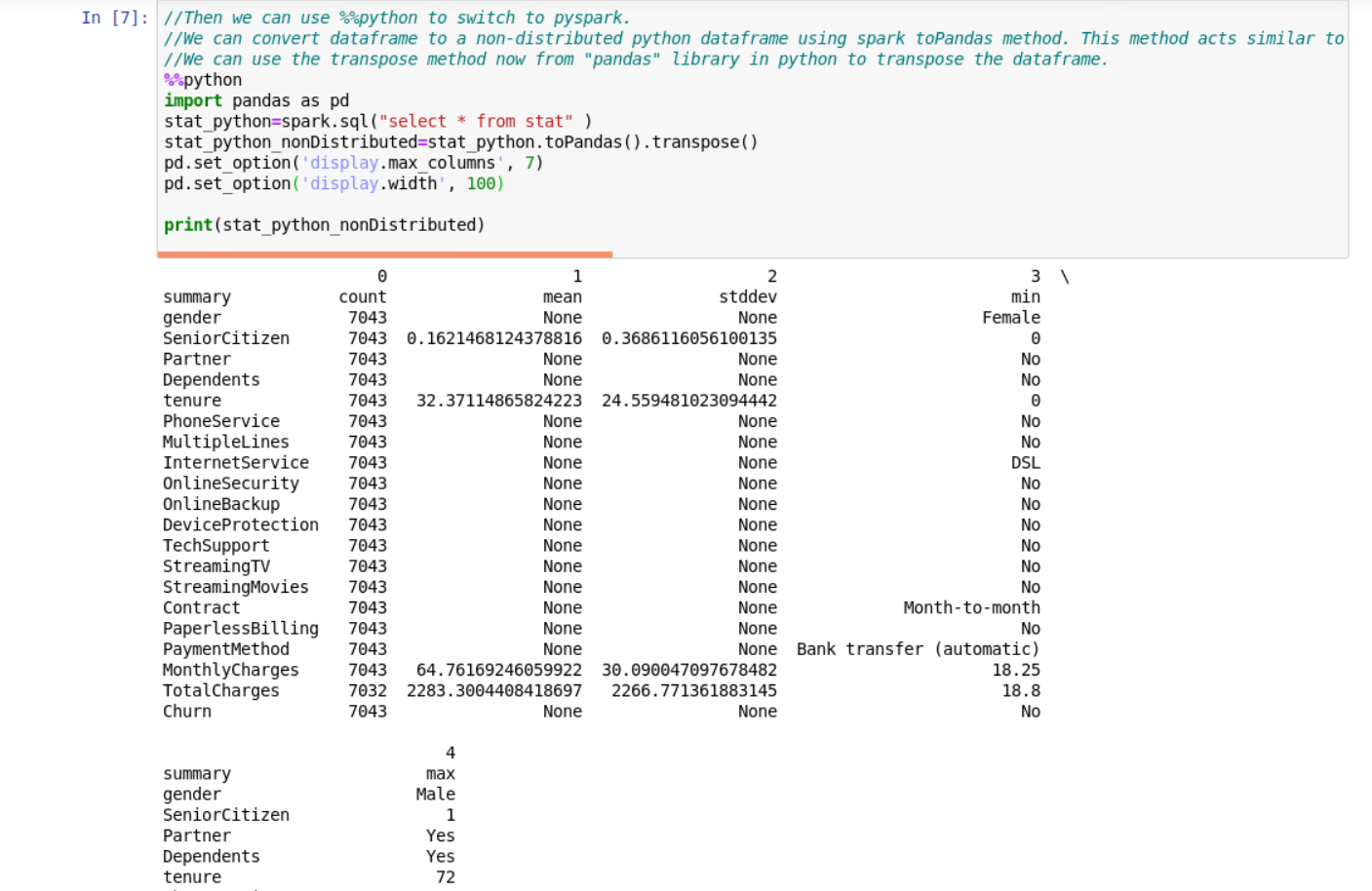


I did some exploratory data analysis. First, checked some statistics on each column(number of rows, min, max, standard deviation etc.) using describe method.



But as it was hard to see summary statistics on each feature so transposed the dataset that is flipped the rows and columns. As spark does not have built-in feature for transforming a dataframe. Spylon allows us to share spark dataframes between python.We just need to create a temporary view from the dataframe.We can use %%python to switch to pyspark.We can convert dataframe to a non-distributed python dataframe using spark toPandas method.I used the transpose method from “pandas” library in python to transpose the dataframe.



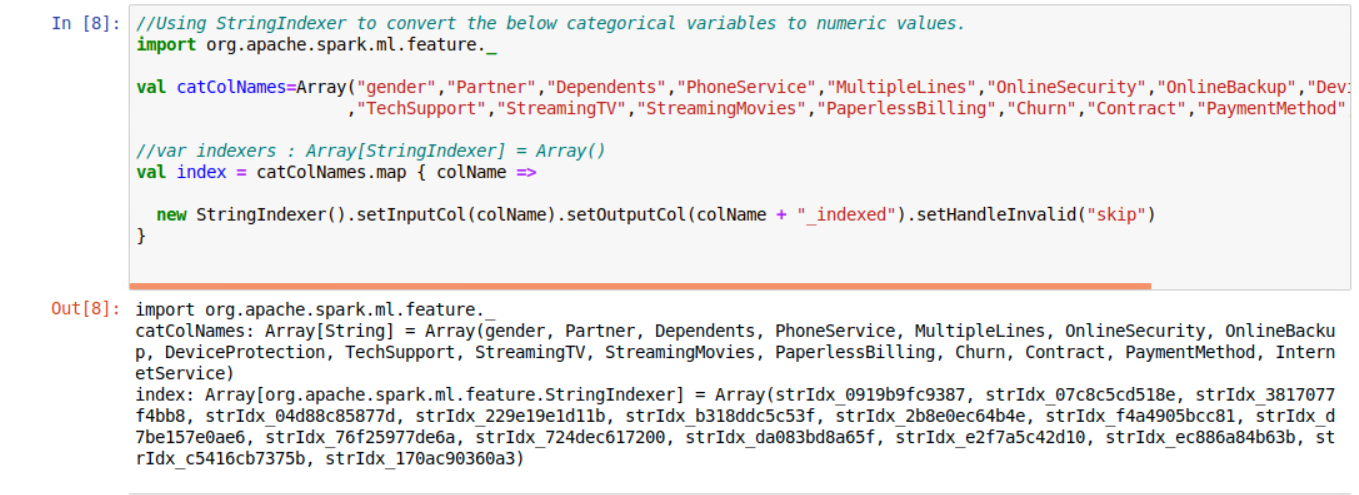


After, using the transpose method, “count” column showed that there were no missing values.

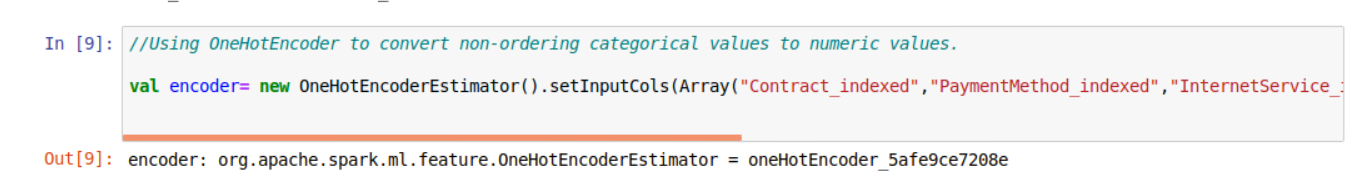
**Feature Engineering**:

As my dataset had categorical values, I needed to convert those in numerical values before sending it to the machine learning model.So, I have used “StringIndexer” for the ordered categorical values(gender, Partner, Dependents, PhoneService, MultipleLines, OnlineSecurity, OnlineBackup, DeviceProtection, TechSupport, StreamingTV, StreamingMovies, PaperlessBilling, Churn, Contract, PaymentMethod, InternetService)

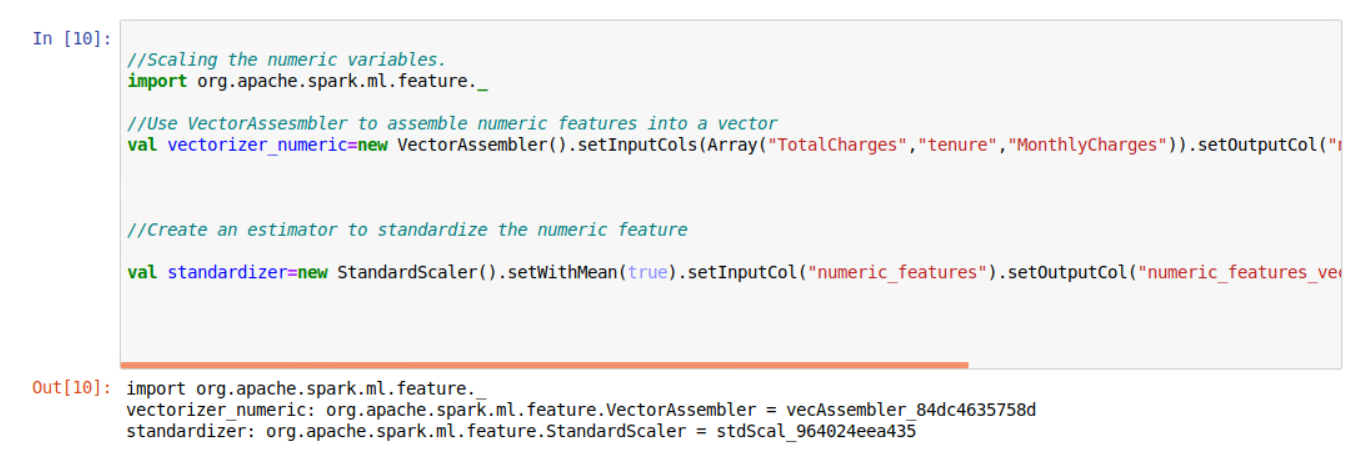
“ SeniorCitizen” already had binary values(Whether the customer is a senior citizen or not (1, 0)), so I have used it directly into the vector assembler later.



Onehot encoding has been used for the non-ordered categorical values. But, we have to use “StringIndexer” before applying One hot encoding.So, I have already indexed them ("Contract\_indexed","PaymentMethod\_indexed","InternetService\_indexed") and then, stored them as the output column("Contract\_coded","PaymentMethod\_coded","InternetService\_coded")

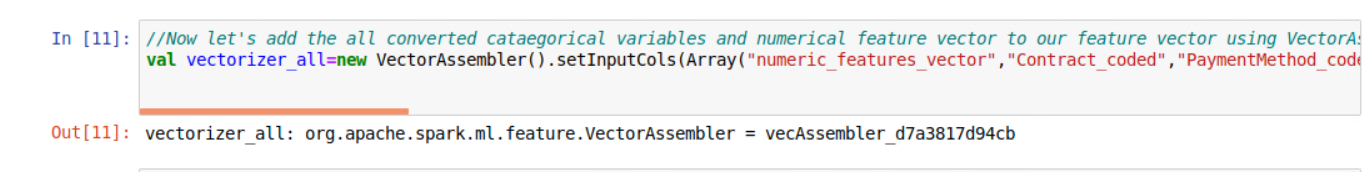


Also, the summary statistics , showed numerical variables(“TotalCharges”, “tenure”,”MonthlyCharges”) with different scale. So, I scaled them using StandardScaler().



Before feeding a dataset to a machine learning algorithm in spark, we need to convert it into (features,label) form where features is a numeric vector of predictors and label is a numeric target variable.

I have assembled all the converted values using “VectorAssembler” and stored it in as “features” .



**Data analysis and experimental Results**

The machine Learning models that I have used are:

*Logistic Regression:*

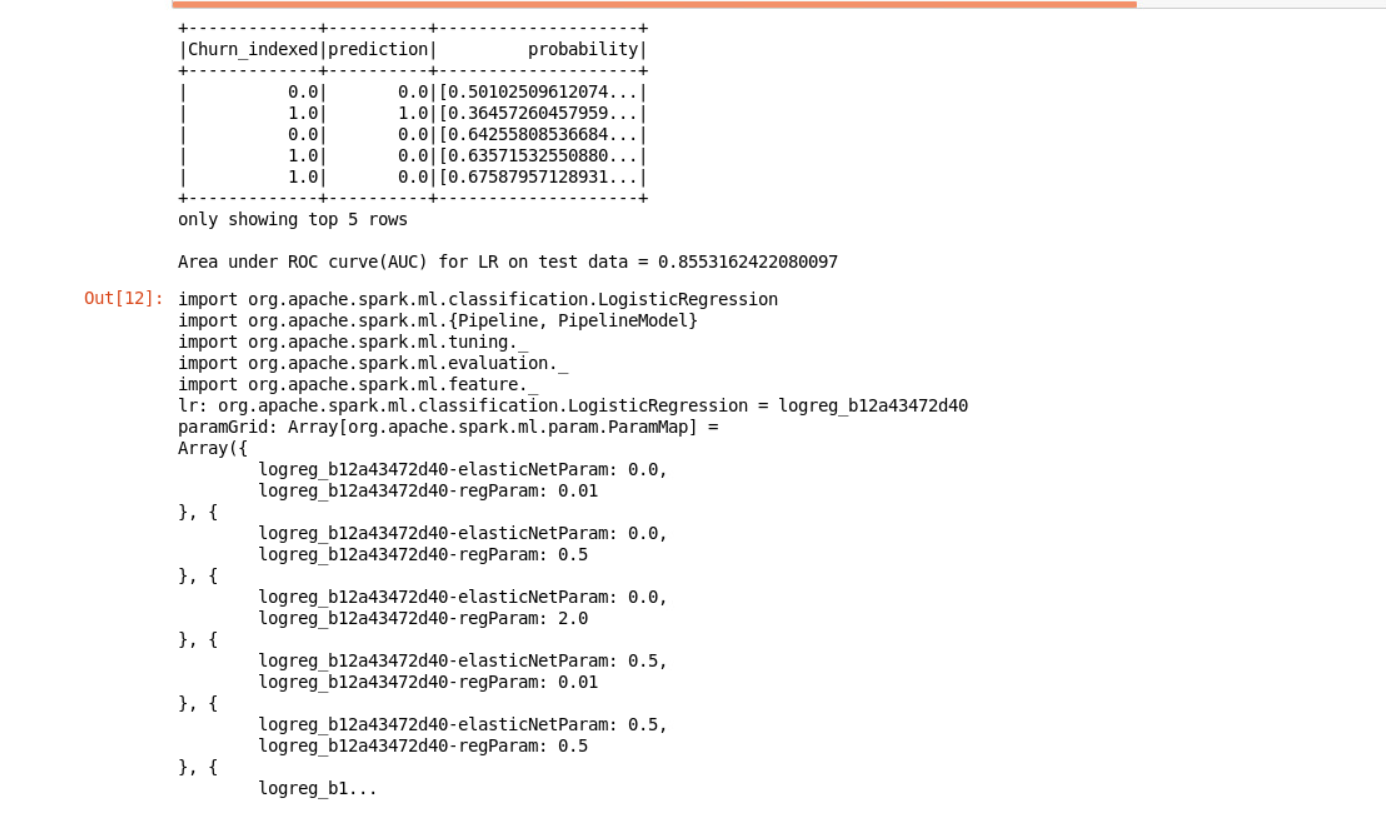
I created a logistic regression model using LogisticRegression class in spark and set its input column to the features vector and its output column to the label indicating whether a Churn is there or not.

I have used binaryClassificationEvaluator with AUC (Area under ROC curve) to evaluate our model. A parameter grid is set up to try different values for hyper-parameters including (regParam and elasticNetParam which are lambda and alpha parameters in elastic\_net regularization) and a 3 fold cross validation is used to tune hyper-parameters.

Finally, I created a pipeline of all the preprocesisng stages as well as the logistic regression and cross validation stages and fit it to the training data. Then, evaluated the model with the test data.

The AUC we get from fitting a logistic regression to the features attribute is 85.5% .





*Random Forest:*

Now, I created a Random Forest model using RandomForestClassifier in spark to predict the label for this dataset. The pipeline is very similar to the pipeline of logistic regression, except that the estimater was set to RandomForestClassifier and the hyper-parameters tuned for Random forest are maxDepth ( The maximum depth of each tree in the random forest) and numTrees( The total number of trees in the RandomForest model).

The AUC we get from fitting random forest to the features attribute is 85.3%.





*Gradient Boosted Classification Tree:*

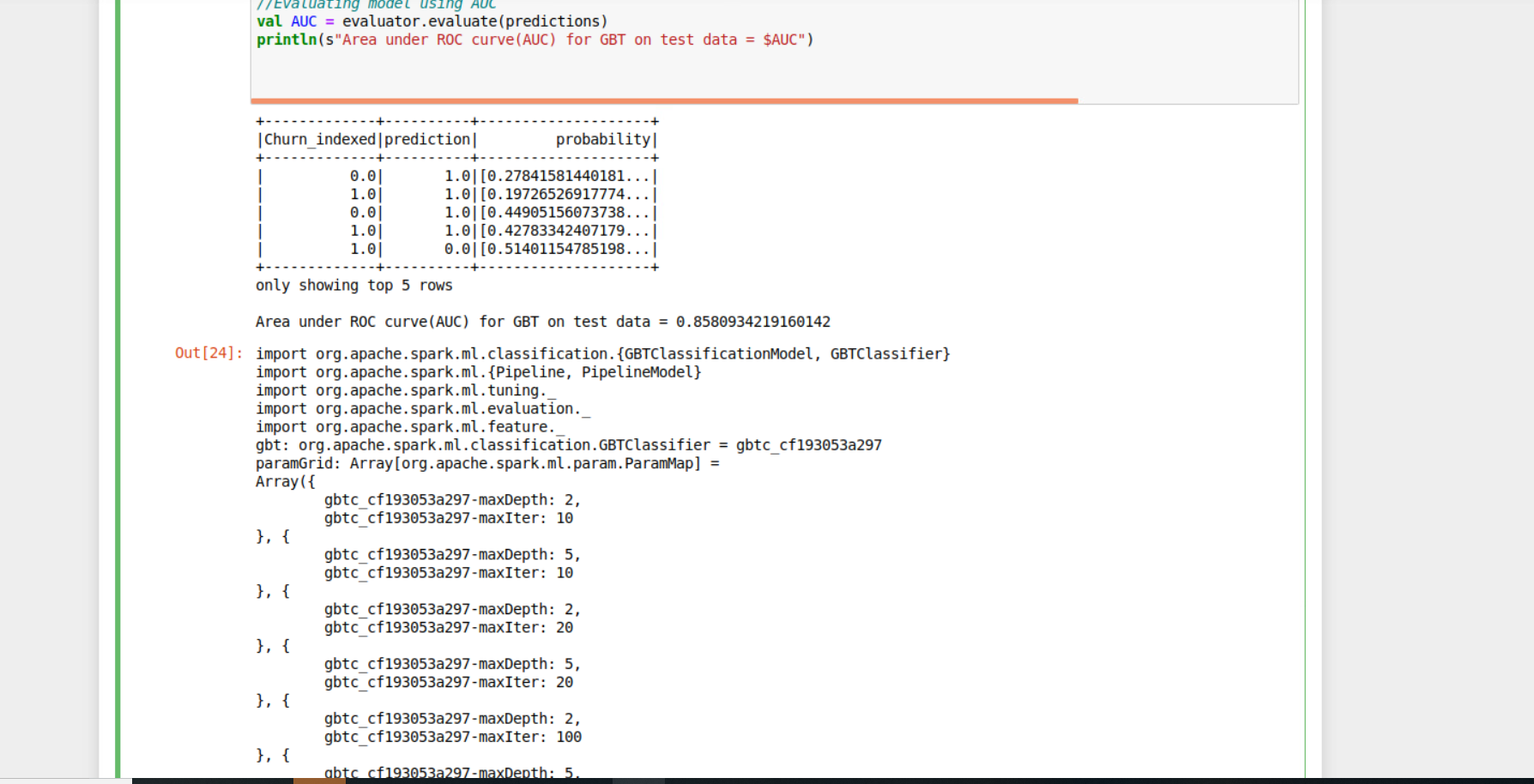
Now, I created Gradient Boosted classification tree to predict label for the dataset. I created a GBT classification model using GBTClassifier .The two hyper-parameter tuned are 1-maxDepth ( the maximum depth of each decision tree, this controls the overfitting due to complexity of each tree) and 2- maxIteration ( The maximum number of trees in the ensemble).

3 fold cross validation is used to tune hyper-parameters.

I created a pipeline of StringIndexer,One hot encoding,scaling , VectorAssembler, and crossValidation stages, fit the pipeline model to training data and use it to transfer and predict the “Churn” for test data.

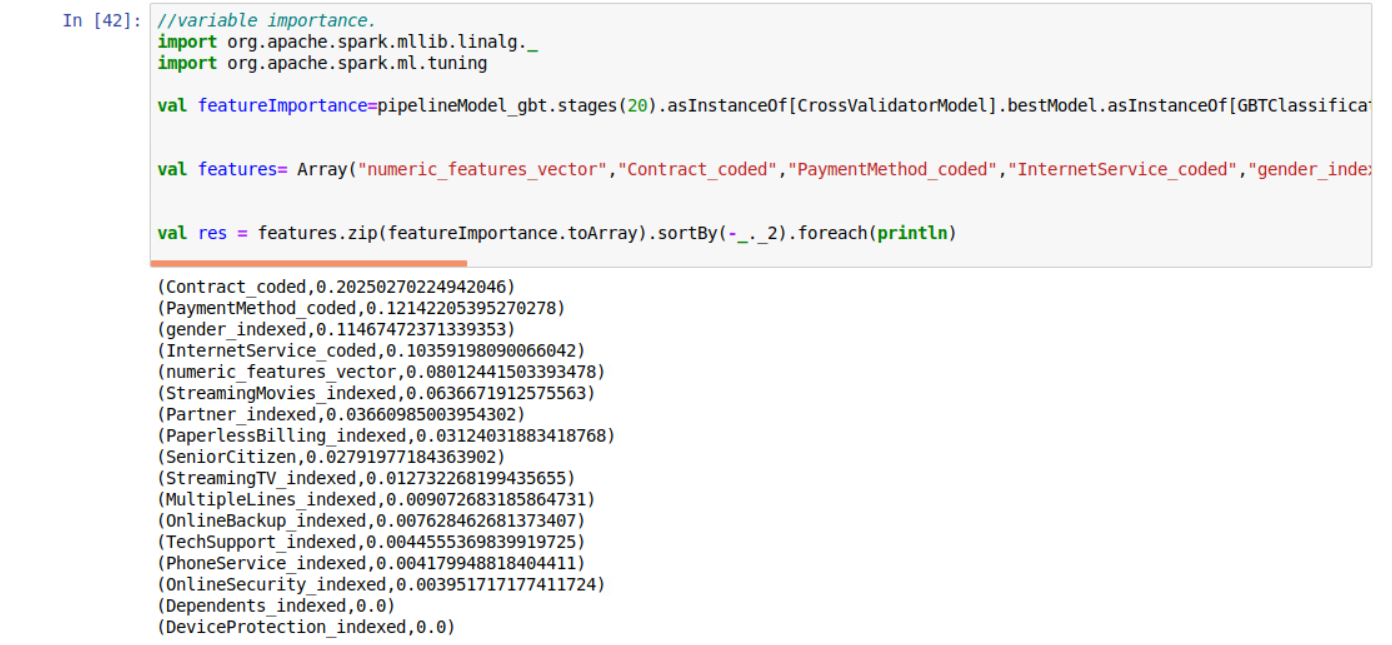
Finally, The AUC we get from fitting gradient boosted classification to the features attribute is 85.8%.





After evaluating the best model, that is the Gradient Boosted classification tree in my project, I have used Variable importance of the best model (the model with the highest AUC). The variable importance is a numeric vector which gives each feature a number between [0,1] indicating the importance of that feature in predicting the outcome.

To get the variable importance we first have to access the best model and in this case it is Gradient Boosted classification tree , pipelineModel\_gbt.stages gives us the array of stages of the pipelineModel. The cross validation stage (cv) is the 21st stage in our pipelineModel and we can access it by index 20, cast it to CrossValidatorModel and get its bestModel. Then we cast the best Model to GBTClassificationModel and use featureImportances method to get the variable importance. This will give us a numeric vector of variable importance. Then we zip this vector to the feature names vector which we used to build our model and sort it in the descending order of its importance.



The result shows that Contract\_coded,PaymentMethod\_coded,gender\_indexed and InternetService\_coded were the most important variables in predicting the target value"Churn".

**Conclusion**

The interesting findings , I got to learn from the project was how to use StringIndexer for the multiple values. How to handle the null values and also finding the variable importance.

**References**

<https://stackoverflow.com/>

<https://www.kaggle.com/>