**Project Report**

**On**

**Telecom Customer Defection Prediction**



Submitted in partial fulfilment for the award of

Post Graduate Diploma in Big Data Analytics (PG-DBDA)

From Know-IT (Pune)

**Guided by:**

**Mr. Tushar Kute**

**Mrs Trupti Joshi**

## Submitted By:

Anand Gaurav (220943025005)

Shrinidhi Siddhanti (220943025041)

Vaibhav Bakhal (220943025053)

**CERTIFICATE**

**TO WHOMSOEVER IT MAY CONCERN**

**This is to certify that**

Anand Gaurav (220943025005)

Shrinidhi Siddhanti (220943025041)

Vaibhav Bakhal (220943025053)

**Have successfully completed their project on**

# Telecom Customer Defection Prediction

**Under the guidance of Mr. Tushar Kute, Mrs Trupti Joshi**

**ACKNOWLEDGEMENT**

This project **“Telecom Customer Defection Prediction”** was a great learning experience for us and we are submitting this work to CDAC Know-IT (Pune).

We all are very glad to mention the name of **Mr. Tushar Kute** and **Mrs Trupti Joshi** for their valuable guidance to work on this project. Their guidance and support helped us to overcome various obstacles and intricacies during the course of project work.

We are highly grateful to **Mr. Vaibhav Inamdar** Manager (Know-IT), C-DAC, for his guidance and support whenever necessary while doing this course Post Graduate Diploma in Big Data

Analytics (PG-DBDA) through C-DAC ACTS, Pune.

Our most heartfelt thanks go to **Mrs. Bakul Joshi** (Course Coordinator-DBDA) who gave all the required support and kind coordination to provide all the necessities like required hardware, internet facility and extra Lab hours to complete the project and throughout the course up to the last day here in C-DAC Know-IT, Pune.

**From:**

Anand Gaurav (220943025005)

Shrinidhi Siddhanti (220943025041)

Vaibhav Bakhal (220943025053)

**TABLE OF CONTENTS**

**ABSTRACT**

**1. INTRODUCTION**

#### 2. SYSTEM REQUIREMENTS

2.1 Software Requirements

2.2 Hardware Requirements

**3. FUNCTIONAL REQUIREMENTS**

#### 4. SYSTEM ARCHITECTURE

1. **METHODOLOGY**

#### 6. DATA VISUALIZATION AND REPRESENTATION

#### 7. RESULT AND FINDING

#### 8. CONCLUSION AND FUTURE SCOPE

**References**

## Abstract

Customer defection is a major problem and one of the most important concerns for large companies. Due to the direct effect on the revenues of the companies, especially in the telecom field, companies are seeking to develop means to predict potential customer to defect. Therefore, finding factors that increase customer defection is important to take necessary actions to reduce this defection. The Objective is to develop a defection prediction model which assists companies to predict customers who are most likely subject to defect. We are processing ETL operation on top of that data using Apache Spark and dumping required data into Mongo Db for data visualization through Tableau and build Machine Learning for estimating the customer defection rate. Prediction of customer defection help in planning process and strategic decision making in large companies.

1. **INTRODUCTION**

* Defection is the process of customers switching from one firm to another in given time. Retaining the existing customers is more profitable than fetching the new customers. The Companies concentrate to the extant customers to avert defection. A defection prediction model is needed to predict the defectors.
* The fast growth of marketplace in every business is giving rise to increased subscriber base.

Accordingly, companies have recognized the significance of retaining the customers who is on hand. It has become necessary for service-providers to reduce the defection rate of customers since the inattention might negatively influence profitability of the company.

* Defection prediction contributes to identify those users who are likely to switch a company over another. The dataset used for customer defection is of telecom industry. It is collected from www.ibm.com that contains customer and service information for telecom industry.

#### Datasets and features:

Data used in the project is structured in nature. It was collected from IBM and Kaggle to develop a defection prediction model which assists companies to predict customers who are most likely subject to defect. Logistic Regression, Decision Tree, Random Forest, GBT Classifier and Gaussian NB Classifier models were used to predict customer defection rate.

**2. SYSTEM REQUIREMENTS**

#### Hardware Requirements:

* Platform – Windows 10
* RAM – 8 GB of RAM
* Peripheral Devices – Mouse, Keyboard, Monitor
* A network connection for data recovering over network.

#### Software Requirements:

* Python 3
* Apache Spark
* MongoDB
* Tableau

####  OS – Window

**3. FUNCTIONAL REQUIREMENTS**

#### (1) Python 3:

* Python is a general purpose and high level programming language.
* It is use for developing desktop GUI applications, websites and web applications.
* Python allows to focus on core functionality of the application by taking care of common programming tasks.
* Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68,
* Small Talk, and UNIX shell and other scripting languages.

#### (2) Apache Spark:

* Apache Spark is an open-source cluster computing system that provides high-level API in Java, Scala, Python and R.
* Apache Spark is one of the fastest-growing big data projects in the history of the Apache Software Foundation. With its memory-oriented architecture, flexible processing libraries, and ease-of-use, Spark has emerged as a leading distributed computing framework for real-time analytics.
* Spark is used for many types of data processing – it comes packaged with support for machine learning, interactive queries (SQL), statistical queries with R, graph processing, ETL, and streaming.
* For loading and storing data, Spark integrates with a number of storage MongoDB, and more.

#### (3) MongoDB:

* **MongoDB**, the most popular NoSQL database, is an open-source document-oriented database.
* MongoDB allows a highly flexible and scalable document structure.
* MongoDB has built in solution for partitioning and sharing your database.
* MongoDB provides a variety of storage engines, allowing you to choose one most suited to your application.
* A real-life scenario for this kind of data manipulation is storing and querying real-time, intraday market data in MongoDB.

#### (4) Tableau:

* Data visualization is the graphical representation of information and data.
* It helps create interactive elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data.
* Tableau is widely used for Business Intelligence but is not limited to it.
* It helps create interactive graphs and charts in the form of dashboards and worksheets to gain business insights.
* All of this is made possible with gestures as simple as drag and drop.

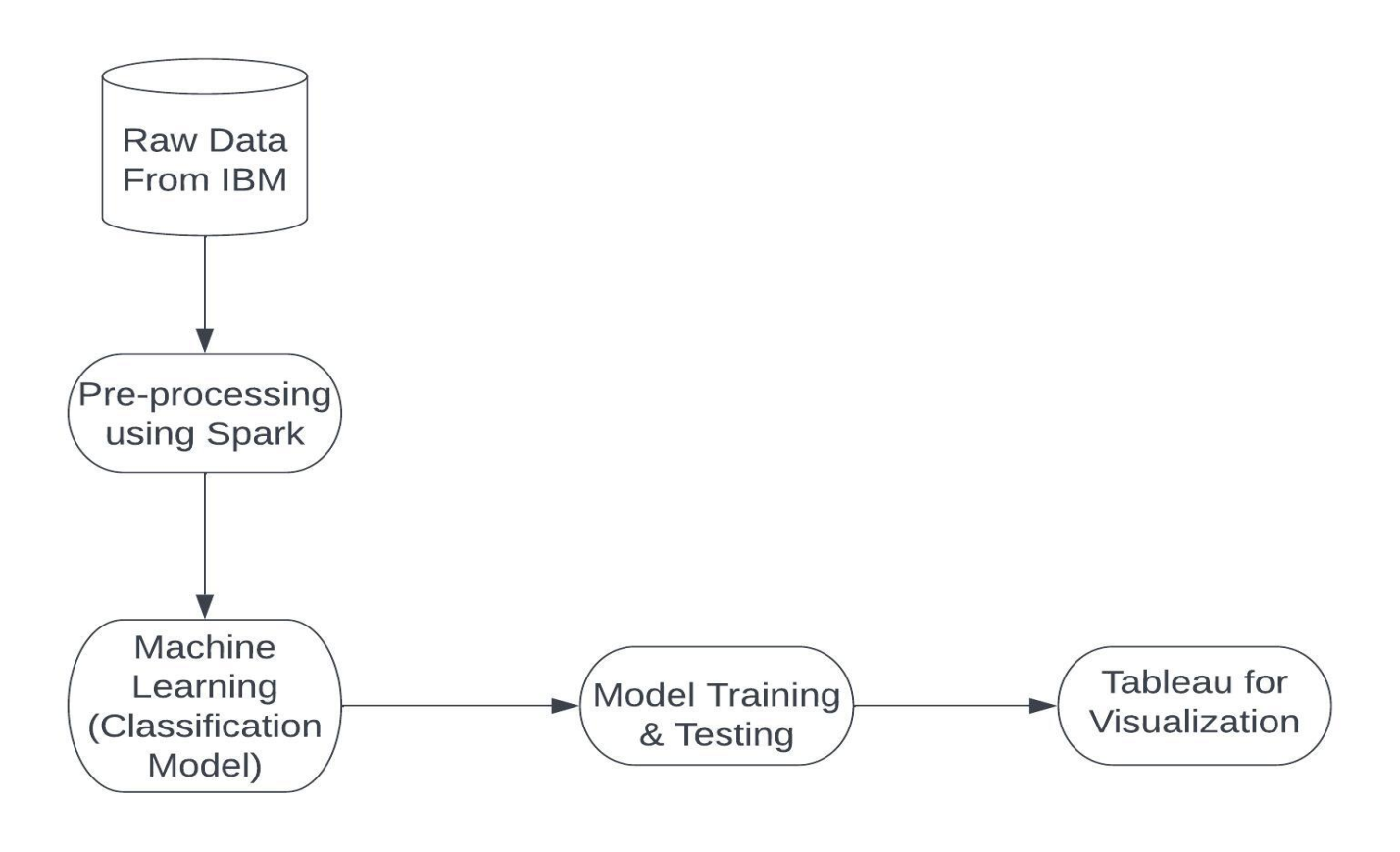
#### Data Cleaning Process:



**Fig: Data Cleaning Process**

Data cleansing or data cleaning is the process of detecting and correcting (or removing) corrupt or inaccurate records from a record set, table, or database and refers to identifying incomplete, incorrect, inaccurate or irrelevant parts of the data and then replacing, modifying, or deleting the dirty or coarse data. Data cleansing may be performed interactively with data wrangling tools, or as batch processing through scripting. After cleansing, a data set should be consistent with other similar data sets in the system. The inconsistencies detected or removed may have been originally caused by user entry errors, by corruption in transmission or storage, or by different data dictionary definitions of similar entities in different stores. Data cleaning differs from data validation in that validation almost invariably means data is rejected from the system at entry and is performed at the time of entry, rather than on batches of data.

### 4. SYSTEM ARCHITECTURE



**Fig: System Architecture of Telecom Customer Defection Prediction**

**5. METHODOLOGY**

In this project we have applied various different types of Classification.

* Logistic Regression
* Decision Tree Classifier
* Random Forest Classifier
* GBT Classifier
* Gaussian NB Classifier

During the implementation we analyze the accuracy of all the algorithms.

**Machine Learning Algorithms**

**Logistic Regression**

Logistic regression is used for solving the classification problems. In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).

**Decision Tree Classifier**

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome. In a Decision tree, there are two nodes, which are the Decision Node and Leaf Node. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.

**Random Forest Classifier**

It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model. As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset."

Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

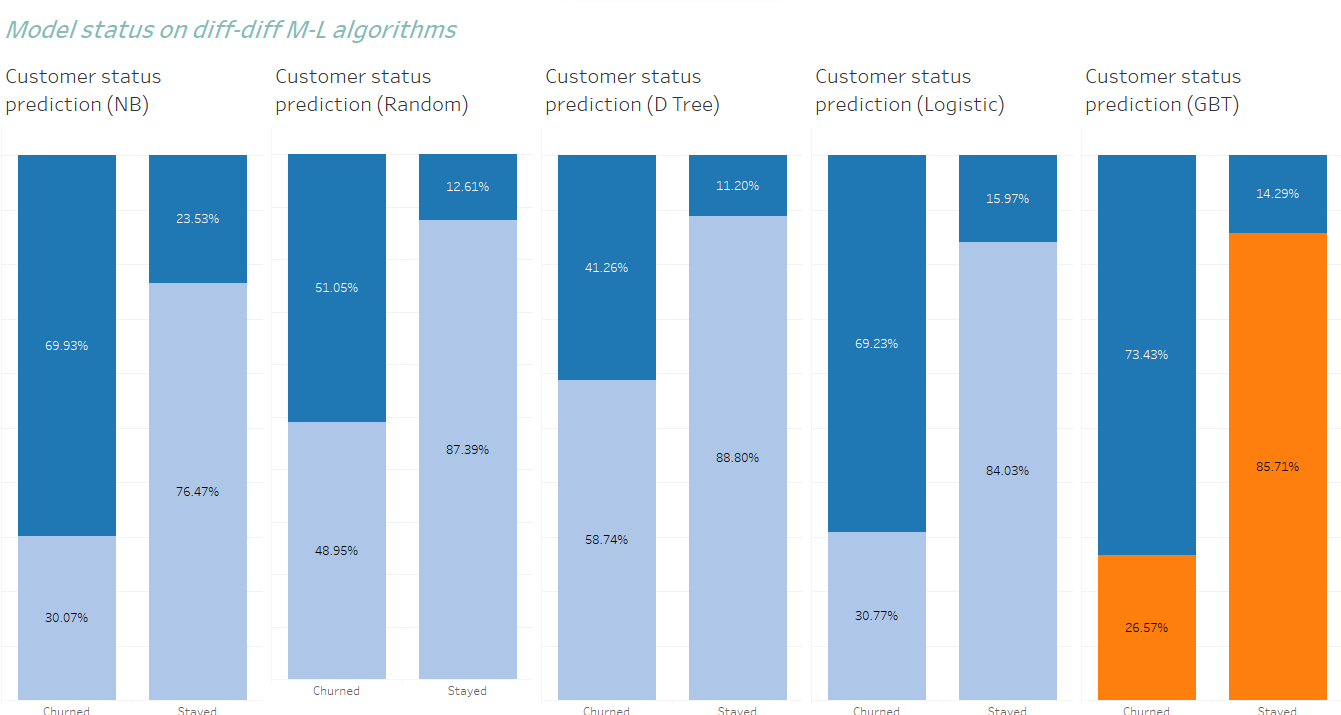
**GBT Classifier**

GBT stands for Gradient Boosted Trees, which is a type of ensemble machine learning algorithm that combines multiple decision tree models to make predictions. In GBT, each decision tree model is trained on the residuals (or errors) of the previous model, with the aim of reducing the overall prediction error.

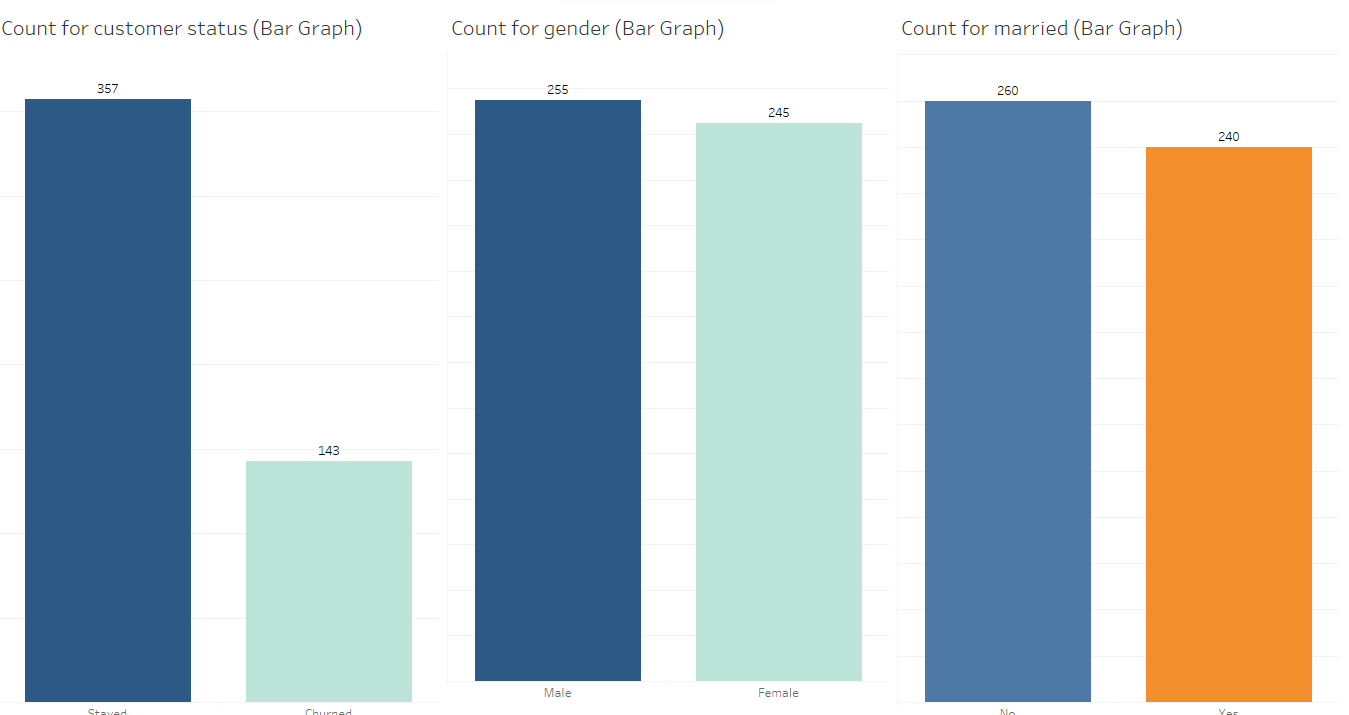
**Gaussian NB Classifier**

A Gaussian NB Classifier is a type if Naïve Bayes Classifier, which is probabilistic algorithm used for classification task. The classifier assumes that the features are normally distributed and calculates the probability of a sample belonging to a particular class based on the mean and variance of each feature in that class.

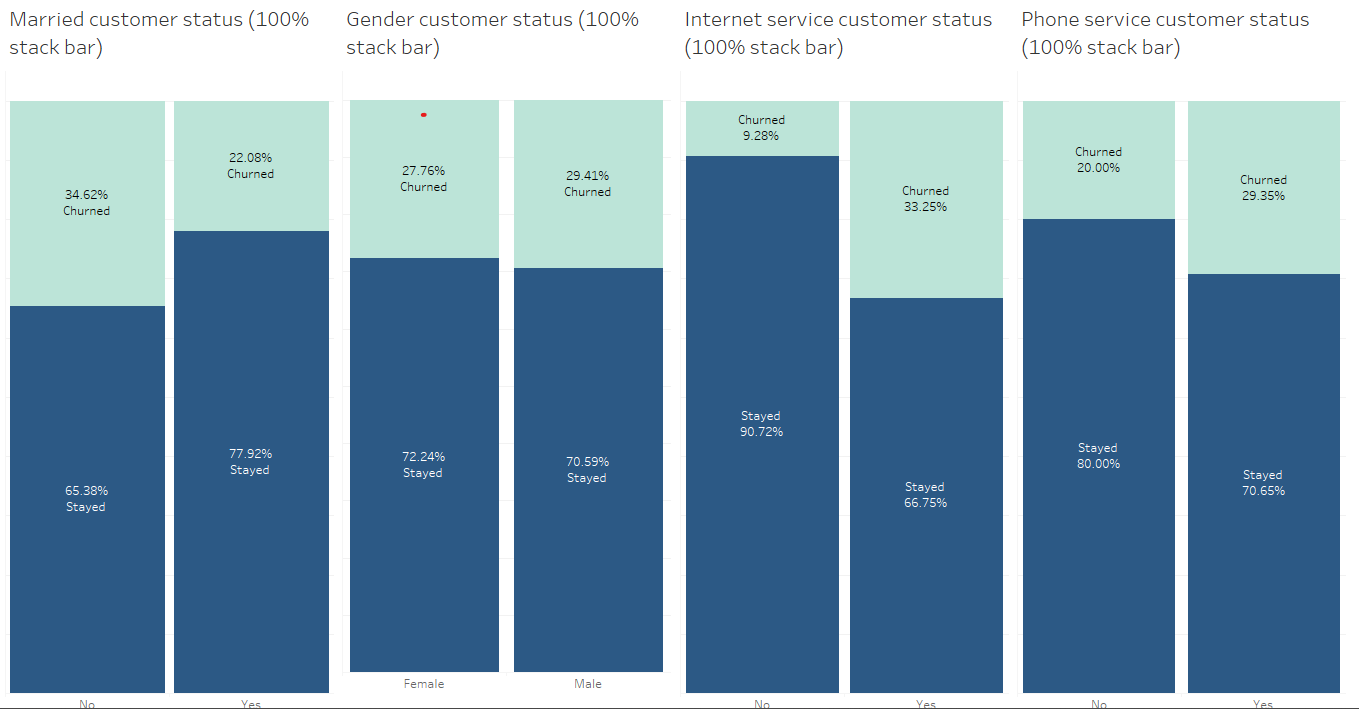
### 6. DATA VISUALIZATION AND REPRESENTATION



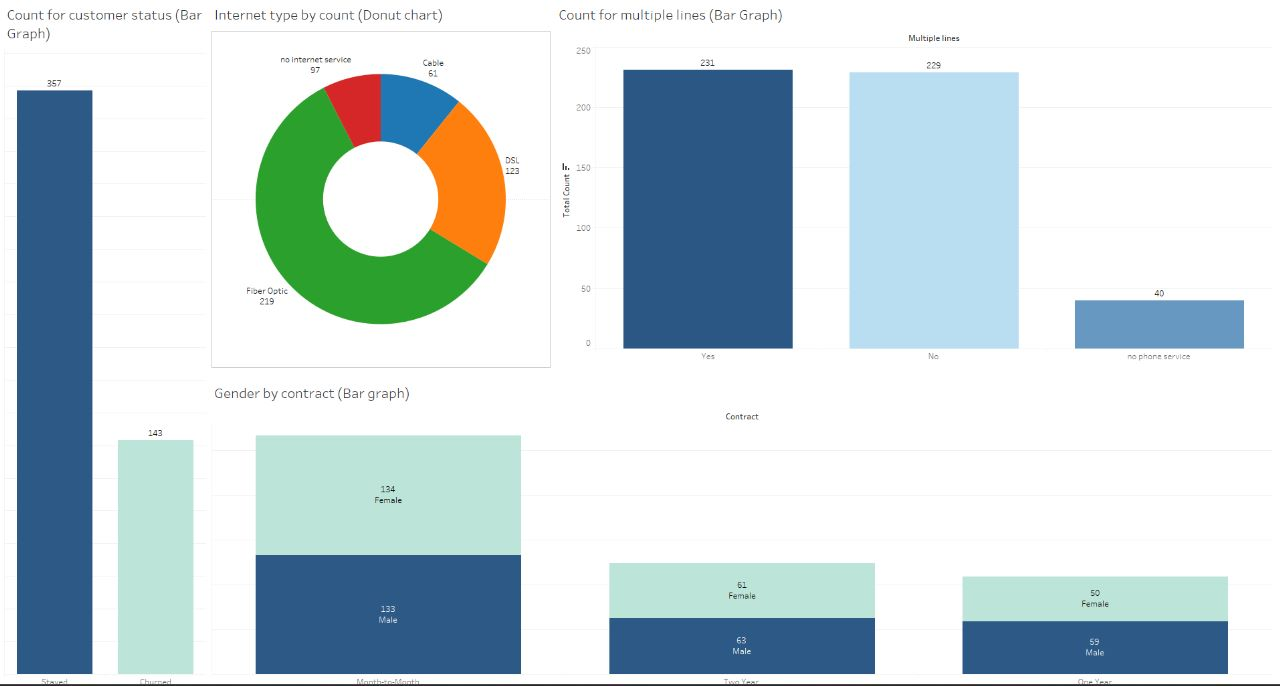
**Fig.: Customer Defection Prediction Dashboard**



**Fig.: Customer Category (Partner, Gender, Senior Citizen, Dependents) Defection Percentage**

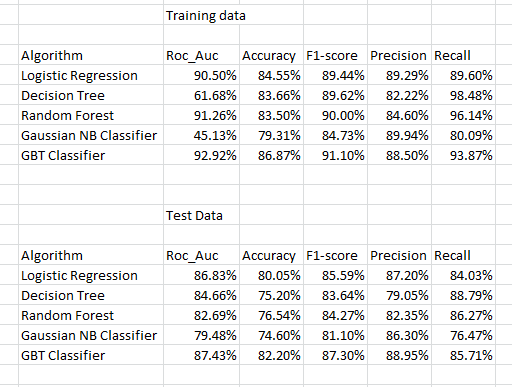


**Fig.: Customer Account Information Defection Analysis**



**Fig.: Gender by Contract Defection Analysis**

# Result and Finding



# 8. CONCLUSION AND FUTURE SCOPE

Telecom customer defection is a central issue for telecom companies, since it decreases profits Furthermore, preventing customer defection is crucial as the global telecom industry is becoming more dissolving and companies are increasingly struggling to retain customers. Currently, most companies invest heavily in marketing to attract new customers. However, keeping existing customers is cheaper than acquiring new customers. Thus, it is becoming more critical and a significant concern for telecommunication companies to prevent customer defection.

We used various classification models to predict telecom defection using customer defection data of telecom industry.

Moreover, the results of this project will give us the ability to predict customer behavior and loss accurately and to optimize their strategies to improve customer retention rates.

Meanwhile, the findings will help companies reduce costs and optimize their budgets. Furthermore, for telecom companies, it will be possible to improve customer targeting through the results of this project and to increase the profits of telecom companies.

# References

* <https://raw.githubusercontent.com/IBM/telco-customer-churn-on-icp4d/master/data/Telco-Customer-Churn.csv>
* <https://towardsdatascience.com/using-principal-component-analysis-pca-for-machine-learning-b6e803f5bf1e>
* <https://imbalanced-learn.org/stable/references/generated/imblearn.over_sampling.RandomOverSampler.html>
* <https://imbalanced-learn.org/stable/over_sampling.html>
* <https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/>
* Breiman, L., “Random forests,” Machine learning, Vol. 45, No. 1, 2001, pp. 5–32.
* <https://www.investopedia.com/terms/c/churnrate.asp>