BI MINI PROJECT REPORT

ON

Customer Churn Analysis

By

Saisrijan Gupta	**PC26**
Anjali Nair	**PC33**
Jaya Shivnani	**PC39**
Prathamesh Jadhav	**PE19**

Guided by

Prof. Anita Thengade

MIT-World Peace University (MIT-WPU)

Faculty of Engineering & Technology
School of Computer Engineering & Technology
*** 2020-2021 ***

PAGE INDEX

ABSTRACT

The rapid development of the market in every sector has resulted in service providers having a larger subscriber base. Customer acquisition costs are rising as a result of new rivals, fresh and inventive business strategies, and improved offerings. Service providers have grasped the need of maintaining on-hand consumers in such a quick setup. It is consequently critical for service providers to avoid churn, a phenomenon in which a client seeks to leave a company's service. Customer churn 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 churn. Therefore, finding factors that increase customer churn is important to take necessary actions to reduce this churn. The main contribution of our work is to develop a churn prediction model which assists telecom operators to predict customers who are most likely subject to churn.

Keywords: Customer Churn, Machine Learning Models, Visualization,

1. INTRODUCTION

The globalization and advancements of telecommunication industry, exponentially raises the number of operators in the market that escalates the competition. In this competitive era, it has become mandatory to maximize the profits periodically, for those various strategies have been proposed, namely, acquiring new customers, up-selling the existing customers & increasing the retention period of existing customers. Among all the strategies, retention of existing customers is least expensive as compared to others. In order to adopt the third strategy, companies have to reduce the potential customer churn i.e., customer movement form the one service provider to other. The main reason of churn is the dissatisfaction of consumer service and support system. The key to unlock solutions to this problem is by forecasting the customers which are at risk of churning. One of the main aim of Customer Churn prediction is to help in establishing strategies for customer retention. Along with growing competition in markets for providing services, the risk of customer churn also increases exponentially. Therefore, establishing strategies to keep track of loyal customers (non-churners) has become a necessity. The customer churn models aim to identify early churn signals and try to predict the customers that leave voluntarily.

2. PROBLEM DEFINITION

Customer churn is one of the most important metrics for a growing business to evaluate. Customer churn is the percentage of customers that stopped using your company's product or service during a certain time frame. 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 commonly expressed as the percentage of service subscribers who discontinue their subscriptions within a given period.

3. TOOLS

1. R Studio

The RStudio IDE is a set of integrated tools designed to help you be more productive with R and Python. It includes a console, syntax-highlighting editor that supports direct code execution, and a variety of robust tools for plotting, viewing history, debugging and managing your workspace.

2. Tableau

Tableau is a powerful and fastest-growing data visualization tool used in the Business Intelligence Industry. It helps in simplifying raw data in a very easily understandable format. Tableau helps create data that can be understood by professionals at any level in an organization. It also allows non-technical users to create customized dashboards.

Data analysis is very fast with the Tableau tool and the visualizations created are in the form of dashboards and worksheets.

3. R

R is a programming language for statistical computing and graphics supported by the R Core Team and the R Foundation for Statistical Computing. Created by statisticians Ross Ihaka and Robert Gentleman, R is used among data miners and statisticians for data analysis and developing statistical software.

4. DATASET DESCRIPTION

The data was downloaded from IBM Sample Data Sets.
LINK:

1. https://community.ibm.com/accelerators/catalog/content/Telco-customer-churn

Each row represents a customer, each column contains that customer's attributes:

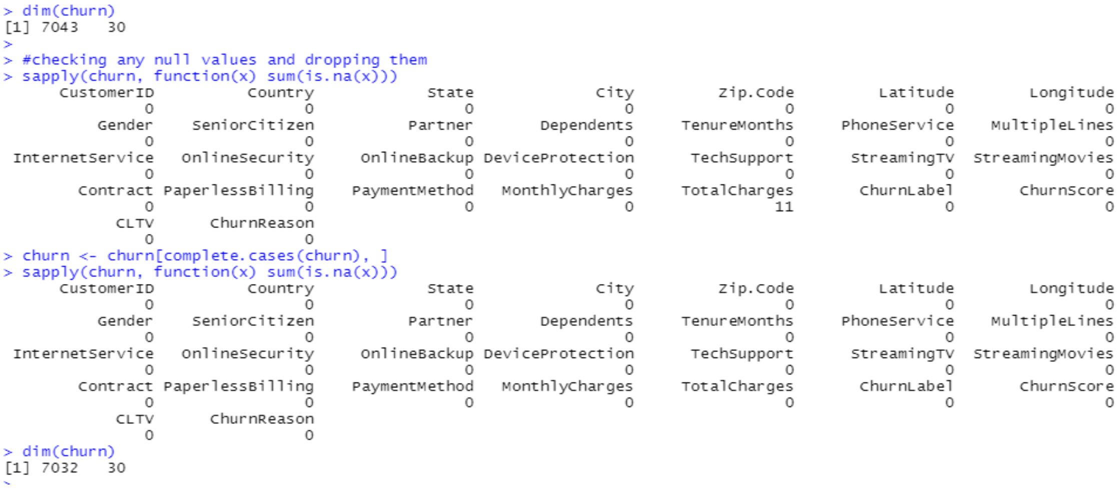
1. CustomerID: A unique ID that identifies each customer.
2. Country: The country of the customer's primary residence.
3. State: The state of the customer's primary residence.
4. City: The city of the customer's primary residence.
5. Zip Code: The zip code of the customer's primary residence.
6. Latitude: The latitude of the customer's primary residence.
7. Longitude: The longitude of the customer's primary residence.
8. Gender: The customer's gender: Male, Female
9. Senior Citizen: Indicates if the customer is 65 or older: Yes, No
10. Partner: Indicate if the customer has a partner: Yes, No
11. Dependents: Indicates if the customer lives with any dependents: Yes, No. Dependents could be children, parents, grandparents, etc.
12. Tenure Months: Indicates the total amount of months that the customer has been with the company by the end of the quarter specified above.
13. Phone Service: Indicates if the customer subscribes to home phone service with the company: Yes, No
14. Multiple Lines: Indicates if the customer subscribes to multiple telephone lines with the company: Yes, No
15. Internet Service: Indicates if the customer subscribes to Internet service with the company: No, DSL, Fiber Optic
16. Online Security: Indicates if the customer subscribes to an additional online security service provided by the company: Yes, No
17. Online Backup: Indicates if the customer subscribes to an additional online backup service provided by the company: Yes, No
18. Device Protection: Indicates if the customer subscribes to an additional device protection plan for their Internet equipment provided by the company: Yes, No
19. Tech Support: Indicates if the customer subscribes to an additional technical support plan from the company with reduced wait times: Yes, No

20. Streaming TV: Indicates if the customer uses their Internet service to stream television programming from a third party provider: Yes, No. The company does not charge an additional fee for this service.
21. Streaming Movies: Indicates if the customer uses their Internet service to stream movies from a third party provider: Yes, No. The company does not charge an additional fee for this service.
22. Contract: Indicates the customer’s current contract type: Month-to-Month, One Year, Two Year.
23. Paperless Billing: Indicates if the customer has chosen paperless billing: Yes, No
24. Payment Method: Indicates how the customer pays their bill: Bank Transfer, Credit Card, Mailed Check, Electronic Check.
25. Monthly Charge: Indicates the customer’s current total monthly charge for all their services from the company.
26. Total Charges: Indicates the customer’s total charges, calculated to the end of the quarter specified above.
27. Churn Label: Yes = the customer left the company this quarter. No = the customer remained with the company. Directly related to Churn Value.
28. Churn Score: A value from 0-100 that is calculated using the predictive tool IBM SPSS Modeler. The model incorporates multiple factors known to cause churn. The higher the score, the more likely the customer will churn.
29. CLTV: Customer Lifetime Value. A predicted CLTV is calculated using corporate formulas and existing data. The higher the value, the more valuable the customer. High value customers should be monitored for churn.
30. Churn Reason: A customer’s specific reason for leaving the company. Directly related to Churn Category.

The raw data contains 7043 rows (customers) and 30 columns (features). The “Churn Label” column is our target.

5. DATA PREPROCESSING

1. **Handling Missing Data:** Missing data is defined as the values or data that is not stored (or not present) for some variable/s in the given dataset. As there were very less missing values, we decided to remove them from our Dataset

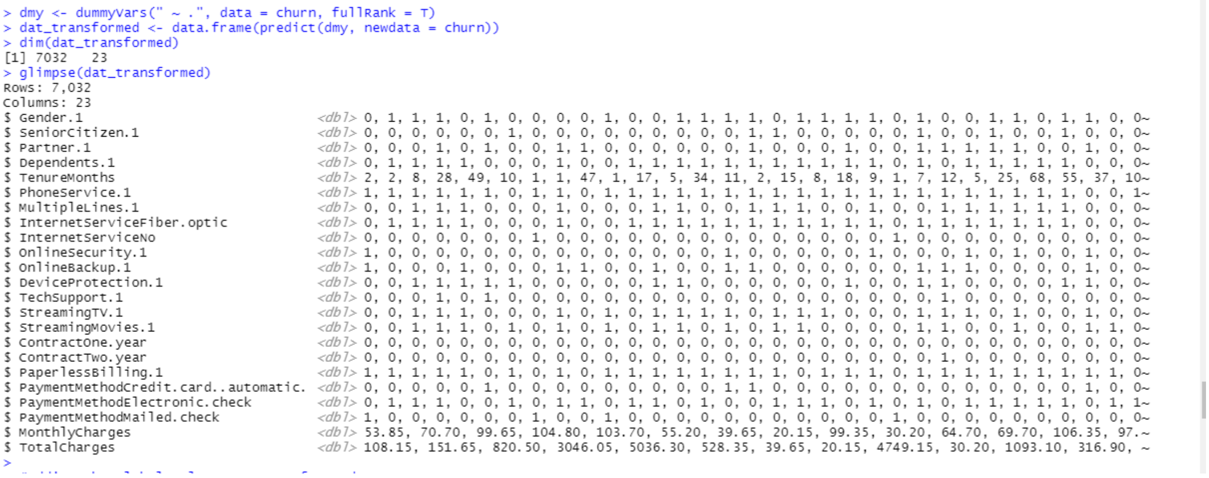


```
> dim(churn)
[1] 7043   30
>
> #checking any null values and dropping them
> sapply(churn, function(x) sum(is.na(x)))
      CustomerID          Country            State             City         Zip.Code         Latitude        Longitude 
               0                0                0                0                0                0                0 
          Gender    SeniorCitizen          Partner       Dependents     TenureMonths     PhoneService    MultipleLines 
               0                0                0                0                0                0                0 
 InternetService   OnlineSecurity     OnlineBackup DeviceProtection      TechSupport      StreamingTV  StreamingMovies 
               0                0                0                0                0                0                0 
        Contract PaperlessBilling    PaymentMethod   MonthlyCharges     TotalCharges       ChurnLabel       ChurnScore 
               0                0                0                0               11                0                0 
            CLTV      ChurnReason 
               0                0 
> churn <- churn[complete.cases(churn), ]
> sapply(churn, function(x) sum(is.na(x)))
      CustomerID          Country            State             City         Zip.Code         Latitude        Longitude 
               0                0                0                0                0                0                0 
          Gender    SeniorCitizen          Partner       Dependents     TenureMonths     PhoneService    MultipleLines 
               0                0                0                0                0                0                0 
 InternetService   OnlineSecurity     OnlineBackup DeviceProtection      TechSupport      StreamingTV  StreamingMovies 
               0                0                0                0                0                0                0 
        Contract PaperlessBilling    PaymentMethod   MonthlyCharges     TotalCharges       ChurnLabel       ChurnScore 
               0                0                0                0                0                0                0 
            CLTV      ChurnReason 
               0                0 
> dim(churn)
[1] 7032   30
```

Fig. 1: Handling Missing Data

2. **Encoding:** Machine learning models require all input and output variables to be numeric. This means that if your data contains categorical data, you must encode it to numbers before you can fit and evaluate a model. Encoding is a required pre-processing step when working with categorical data for machine learning algorithms.

3. **One Hot Encoding:** One-Hot Encoding. In this technique, one-hot (dummy) encoding is applied to the features, creating a binary column for each category level and returning a sparse matrix. In each dummy variable, the label "1" will represent the existence of the level in the variable, while the label "0" will represent its non-existence.



```
> dmy <- dummyVars(" ~ .", data = churn, fullRank = T)
> dat_transformed <- data.frame(predict(dmy, newdata = churn))
> dim(dat_transformed)
[1] 7032   23
> glimpse(dat_transformed)
Rows: 7,032
Columns: 23
$ Gender.1                             <dbl> 0, 1, 1, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 0, 0, 1, 1, 0, 1, 1, 0, 0~
$ SeniorCitizen.1                      <dbl> 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0~
$ Partner.1                            <dbl> 0, 0, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0, 1, 0, 0~
$ Dependents.1                         <dbl> 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 0, 0, 0~
$ TenureMonths                         <dbl> 2, 2, 8, 28, 49, 10, 1, 1, 47, 1, 17, 5, 34, 11, 2, 15, 8, 18, 9, 1, 7, 12, 5, 25, 68, 55, 37, 10~
$ PhoneService.1                       <dbl> 1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 1~
$ MultipleLines.1                      <dbl> 0, 0, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 1, 0, 0, 1, 1, 1, 0, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0, 0, 0~
$ InternetServiceFiber.optic           <dbl> 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0~
$ InternetServiceNo                    <dbl> 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
$ OnlineSecurity.1                     <dbl> 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0~
$ OnlineBackup.1                       <dbl> 1, 0, 0, 0, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0~
$ DeviceProtection.1                   <dbl> 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0~
$ TechSupport.1                        <dbl> 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0~
$ StreamingTV.1                        <dbl> 0, 0, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 0, 1, 0, 0, 1, 0, 0~
$ StreamingMovies.1                    <dbl> 0, 0, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 1, 1, 0~
$ ContractOne.year                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
$ ContractTwo.year                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0~
$ PaperlessBilling.1                   <dbl> 1, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0~
$ PaymentMethodCredit.card..automatic. <dbl> 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0~
$ PaymentMethodElectronic.check        <dbl> 0, 1, 1, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 1, 1, 0, 1, 1~
$ PaymentMethodMailed.check            <dbl> 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
$ MonthlyCharges                       <dbl> 53.85, 70.70, 99.65, 104.80, 103.70, 55.20, 39.65, 20.15, 99.35, 30.20, 64.70, 69.70, 106.35, 97.~
$ TotalCharges                         <dbl> 108.15, 151.65, 820.50, 3046.05, 5036.30, 528.35, 39.65, 20.15, 4749.15, 30.20, 1093.10, 316.90, ~
>
```

Fig. 2: Dataset after One Hot Encoding

6. DATA ANALYTICS

1. **Correlation Matrix:** A correlation matrix is a table showing correlation coefficients between variables. Each cell in the table shows the correlation between two variables. A correlation matrix is used to summarize data, as an input into a more advanced analysis, and as a diagnostic for advanced analyses.

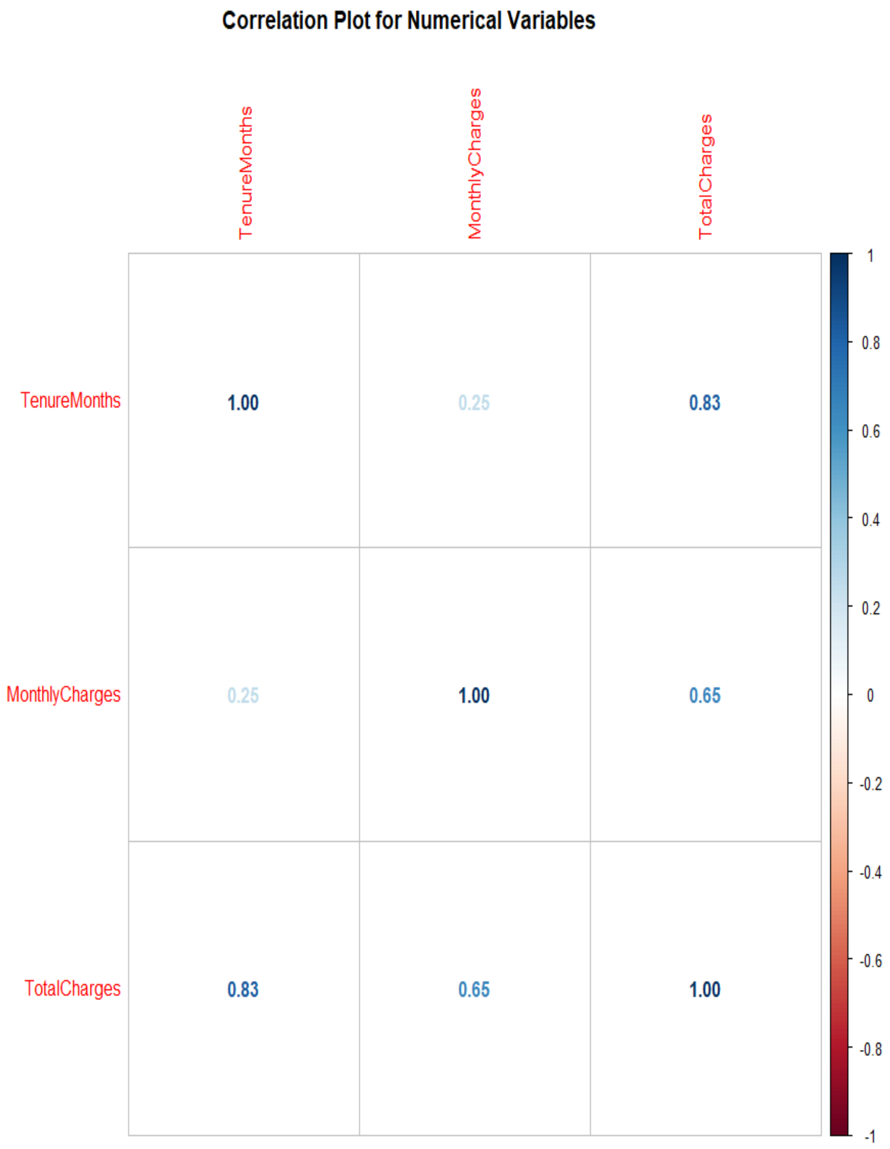
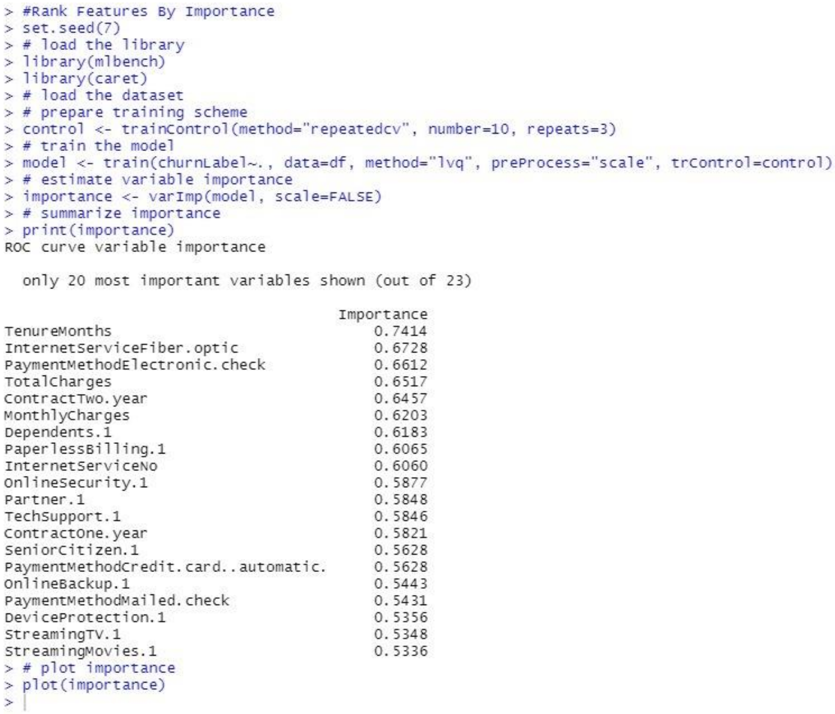


Fig. 3: Correlation Matrix

2. **Rank Feature By Importance:** Constructs an Learning Vector Quantization (LVQ) model. The varImp is then used to estimate the variable importance, which is printed and plotted.



```
> #Rank Features By Importance
> set.seed(7)
> # load the library
> library(mlbench)
> library(caret)
> # load the dataset
> # prepare training scheme
> control <- trainControl(method="repeatedcv", number=10, repeats=3)
> # train the model
> model <- train(churnLabel~., data=df, method="lvq", preProcess="scale", trControl=control)
> # estimate variable importance
> importance <- varImp(model, scale=FALSE)
> # summarize importance
> print(importance)
ROC curve variable importance

  only 20 most important variables shown (out of 23)

                                     Importance
TenureMonths                             0.7414
InternetServiceFiber.optic               0.6728
PaymentMethodElectronic.check            0.6612
TotalCharges                             0.6517
ContractTwo.year                         0.6457
MonthlyCharges                           0.6203
Dependents.1                             0.6183
PaperlessBilling.1                       0.6065
InternetServiceNo                        0.6060
OnlineSecurity.1                         0.5877
Partner.1                                0.5848
TechSupport.1                            0.5846
ContractOne.year                         0.5821
SeniorCitizen.1                          0.5628
PaymentMethodCredit.card..automatic.     0.5628
OnlineBackup.1                           0.5443
PaymentMethodMailed.check                0.5431
DeviceProtection.1                       0.5356
StreamingTV.1                            0.5348
StreamingMovies.1                        0.5336
> # plot importance
> plot(importance)
> |
```

Fig. 4: Feature Importance

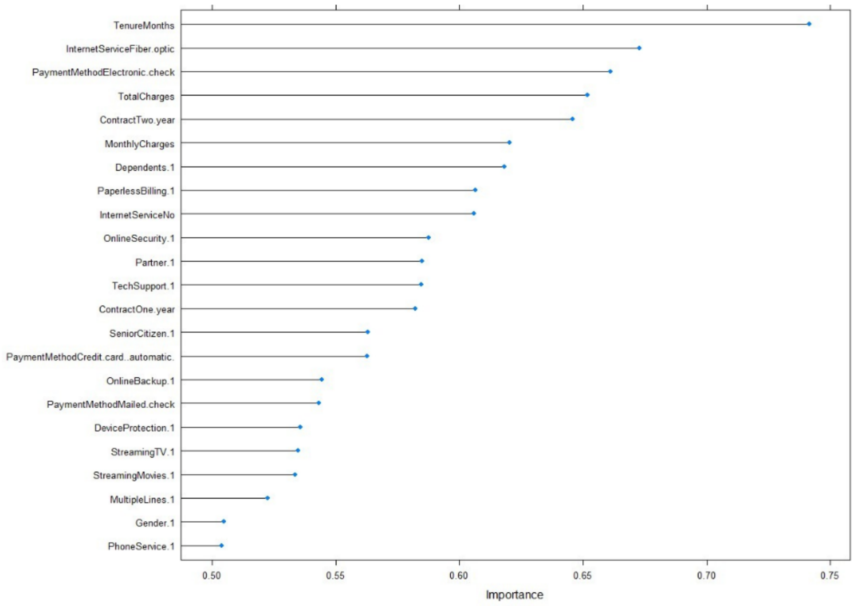
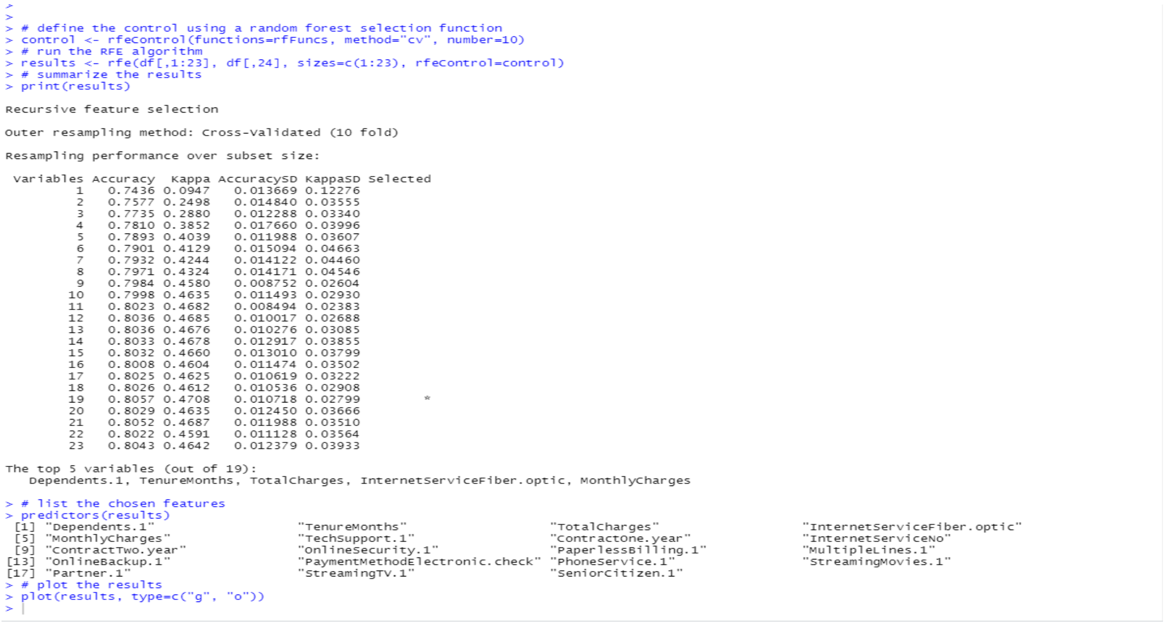


Fig. 5: Plot of Features Importance

3. Feature Selection:

Automatic feature selection methods can be used to build many models with different subsets of a dataset and identify those attributes that are and are not required to build an accurate model. A popular automatic method for feature selection provided by the caret R package is called Recursive Feature Elimination or RFE. A Random Forest algorithm is used on each iteration to evaluate the model. The following shows the output returned from the rfe function. The output indicates that RFE recommends 19 features for the model (see the little asterisk sign under the "Selected" column). Both accuracy and Kappa reach the maximum level when 19 features are retained in the model.



```
>
>
> # define the control using a random forest selection function
> control <- rfeControl(functions=rfFuncs, method="cv", number=10)
> # run the RFE algorithm
> results <- rfe(df[,1:23], df[,24], sizes=c(1:23), rfeControl=control)
> # summarize the results
> print(results)

Recursive feature selection

Outer resampling method: Cross-Validated (10 fold)

Resampling performance over subset size:

 Variables Accuracy  Kappa AccuracySD KappaSD Selected
         1   0.7436 0.0947   0.013669 0.12276
         2   0.7577 0.2498   0.014840 0.03555
         3   0.7735 0.2880   0.012288 0.03340
         4   0.7810 0.3852   0.017660 0.03996
         5   0.7893 0.4039   0.011988 0.03607
         6   0.7901 0.4129   0.015094 0.04663
         7   0.7932 0.4244   0.014122 0.04460
         8   0.7971 0.4324   0.014171 0.04546
         9   0.7984 0.4580   0.008752 0.02604
        10   0.7998 0.4635   0.011493 0.02930
        11   0.8023 0.4682   0.008494 0.02383
        12   0.8036 0.4685   0.010017 0.02688
        13   0.8036 0.4676   0.010276 0.03085
        14   0.8033 0.4678   0.012917 0.03855
        15   0.8032 0.4660   0.013010 0.03799
        16   0.8008 0.4604   0.011474 0.03502
        17   0.8025 0.4625   0.010619 0.03222
        18   0.8026 0.4612   0.010536 0.02908
        19   0.8057 0.4708   0.010718 0.02799        *
        20   0.8029 0.4635   0.012450 0.03666
        21   0.8052 0.4687   0.011988 0.03510
        22   0.8022 0.4591   0.011128 0.03564
        23   0.8043 0.4642   0.012379 0.03933

The top 5 variables (out of 19):
   Dependents.1, TenureMonths, TotalCharges, InternetServiceFiber.optic, MonthlyCharges

> # list the chosen features
> predictors(results)
 [1] "Dependents.1"       "TenureMonths"                    "TotalCharges"          "InternetServiceFiber.optic"
 [5] "MonthlyCharges"     "TechSupport.1"                   "ContractOne.year"      "InternetServiceNo"
 [9] "ContractTwo.year"   "OnlineSecurity.1"                "PaperlessBilling.1"    "MultipleLines.1"
[13] "OnlineBackup.1"     "PaymentMethodElectronic.check"   "PhoneService.1"        "StreamingMovies.1"
[17] "Partner.1"          "StreamingTV.1"                   "SeniorCitizen.1"
> # plot the results
> plot(results, type=c("g", "o"))
> 
```

Fig. 6: Top 19 features

We can also see the same results visually in the following graphs (the blue dot represents the optimal solution — i.e., 19 features).

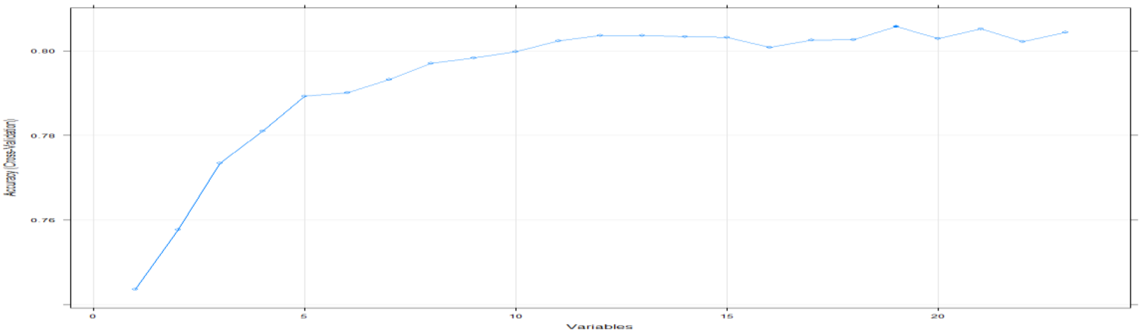
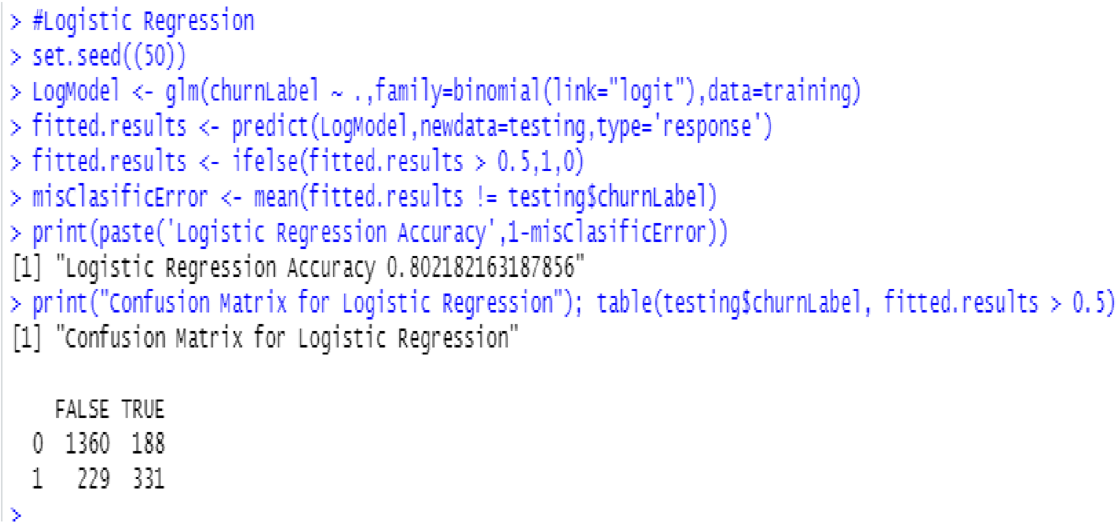


Fig. 7: Number of optimal features i.e. 19 features

7. MACHINE LEARNING ALGORITHM

1. LOGISTIC REGRESSION:

Logistic regression in R Programming is a classification algorithm used to find the probability of event success and event failure. Logistic regression is used when the dependent variable is binary(0/1, True/False, Yes/No) in nature. Logit function is used as a link function in a binomial distribution. It gave an Accuracy of 80%



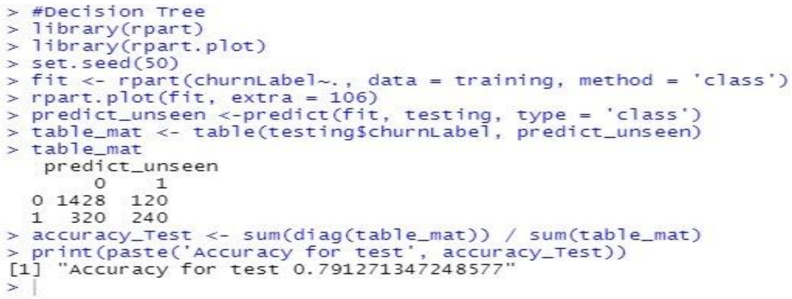
```
> #Logistic Regression
> set.seed((50))
> LogModel <- glm(churnLabel ~ .,family=binomial(link="logit"),data=training)
> fitted.results <- predict(LogModel,newdata=testing,type='response')
> fitted.results <- ifelse(fitted.results > 0.5,1,0)
> misClasificError <- mean(fitted.results != testing$churnLabel)
> print(paste('Logistic Regression Accuracy',1-misClasificError))
[1] "Logistic Regression Accuracy 0.802182163187856"
> print("Confusion Matrix for Logistic Regression"); table(testing$churnLabel, fitted.results > 0.5)
[1] "Confusion Matrix for Logistic Regression"

    FALSE TRUE
  0  1360  188
  1   229  331
>
```

Fig. 8: Accuracy and Confusion Matrix for LR

2. DECISION TREE

Decision tree is a graph to represent choices and their results in form of a tree. The nodes in the graph represent an event or choice and the edges of the graph represent the decision rules or conditions. It gave an accuracy of 79%.



```
> #Decision Tree
> library(rpart)
> library(rpart.plot)
> set.seed(50)
> fit <- rpart(churnLabel~., data = training, method = 'class')
> rpart.plot(fit, extra = 106)
> predict_unseen <-predict(fit, testing, type = 'class')
> table_mat <- table(testing$churnLabel, predict_unseen)
> table_mat
   predict_unseen
       0    1
  0 1428  120
  1  320  240
> accuracy_Test <- sum(diag(table_mat)) / sum(table_mat)
> print(paste('Accuracy for test', accuracy_Test))
[1] "Accuracy for test 0.791271347248577"
> |
```

Fig. 9:Confusion Matrix and Accuracy of Decision Tree

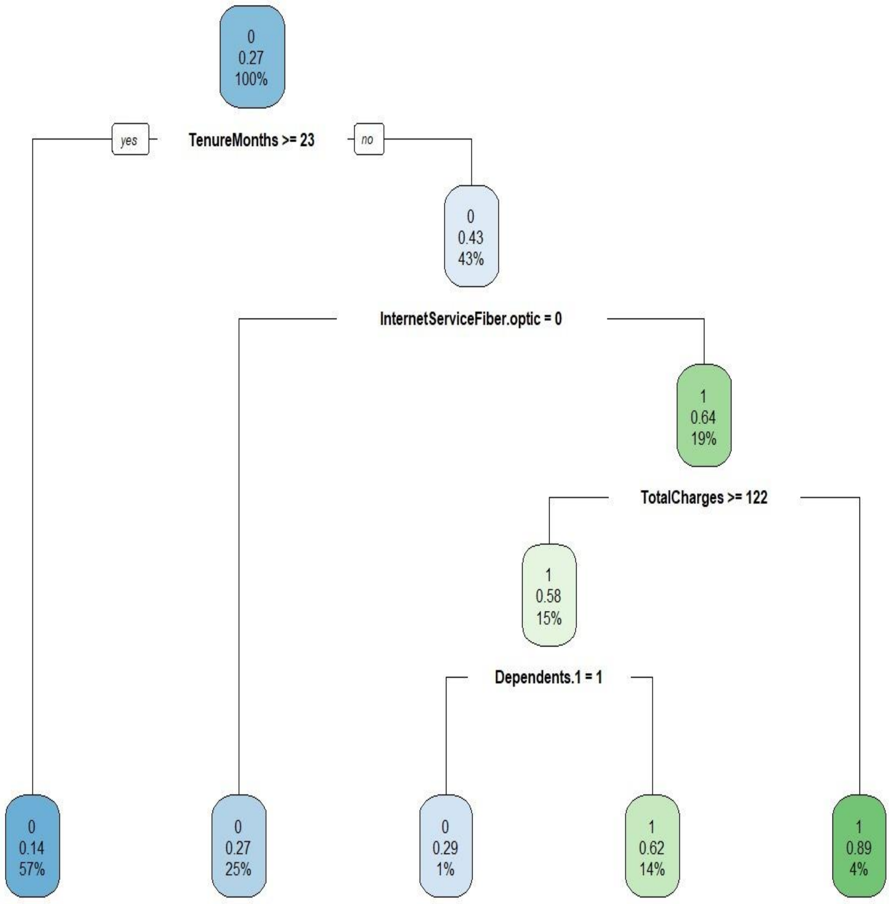
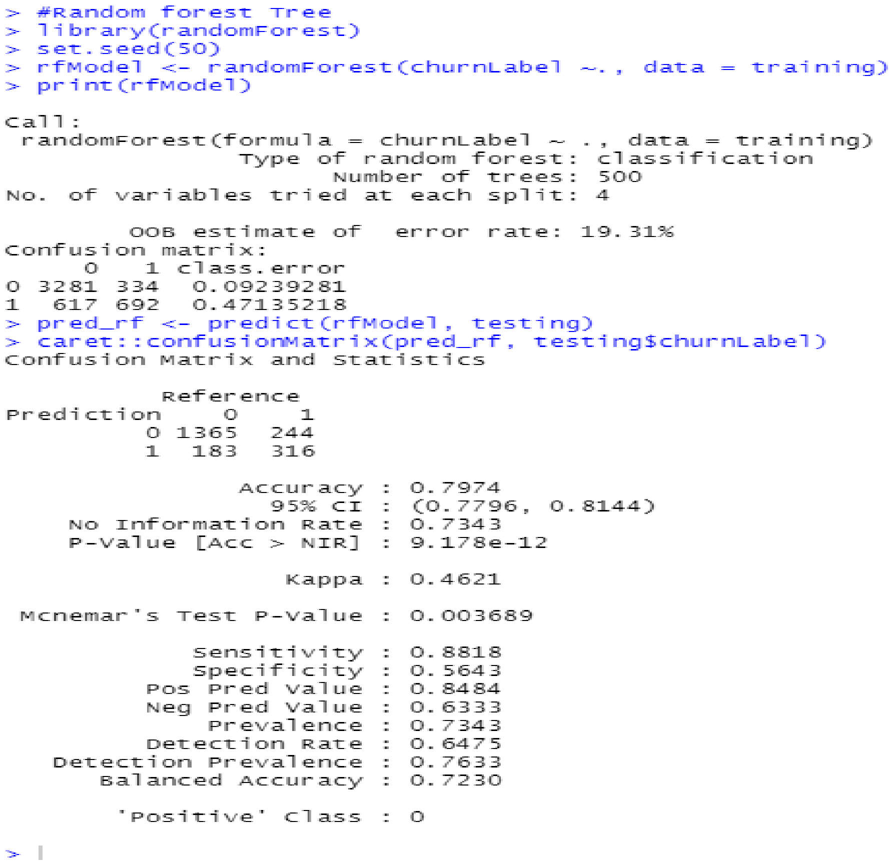


Fig. 10: Decision Tree Plot

3. RANDOM FOREST

Random Forest in R Programming is an ensemble of decision trees. It builds and combines multiple decision trees to get more accurate predictions. It's a non-linear classification algorithm. Each decision tree model is used when employed on its own. It gave an Accuracy of 79%



```
> #Random forest Tree
> library(randomForest)
> set.seed(50)
> rfModel <- randomForest(churnLabel ~., data = training)
> print(rfModel)

Call:
 randomForest(formula = churnLabel ~ ., data = training)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 4

        OOB estimate of  error rate: 19.31%
Confusion matrix:
     0   1 class.error
0 3281 334  0.09239281
1  617 692  0.47135218
> pred_rf <- predict(rfModel, testing)
> caret::confusionMatrix(pred_rf, testing$churnLabel)
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 1365  244
         1  183  316
                                          
               Accuracy : 0.7974          
                 95% CI : (0.7796, 0.8144)
    No Information Rate : 0.7343          
    P-Value [Acc > NIR] : 9.178e-12       
                                          
                  Kappa : 0.4621          
                                          
 Mcnemar's Test P-Value : 0.003689        
                                          
            Sensitivity : 0.8818          
            Specificity : 0.5643          
         Pos Pred Value : 0.8484          
         Neg Pred Value : 0.6333          
             Prevalence : 0.7343          
         Detection Rate : 0.6475          
   Detection Prevalence : 0.7633          
      Balanced Accuracy : 0.7230          
                                          
       'Positive' Class : 0               
                                          
> 
```

Fig. 11: Confusion Matrix and Accuracy of Random Forest

8. BUSINESS INTELLIGENCE PERFORMED

` Business analytics help organizations to reduce risks. By helping them make the right decisions based on available data such as customer preferences, trends, and so on, it can help businesses to curtail short and long-term risk.

Organizations employ Business analytics so they can make data-driven decisions. Business analytics gives businesses an excellent overview and insight on how companies can become more efficient, and these insights will enable such businesses to optimize and automate their processes. It is no surprise that data-driven companies, and also make use of business analytics, usually outperform their contemporaries. The reason for this is that the insights gained via business analytics enable them to; understand why specific results are achieved, explore more effective business processes, and even predict the likelihood of certain results.

There is no denying it that business analytics have come to change the dynamics of businesses and how they operate. Its importance cannot be overestimated, and with more and more companies relying on it for their decision-making process, it is something that businesses should consider incorporating if it hasn't done so already.

To understand the customer churn pattern, we performed various types of analytics like visualization and predictive modelling. We were able to gain useful insights into the data that we had through the various types of visualizations.

9. Visualizations/ Graphs Interpretations

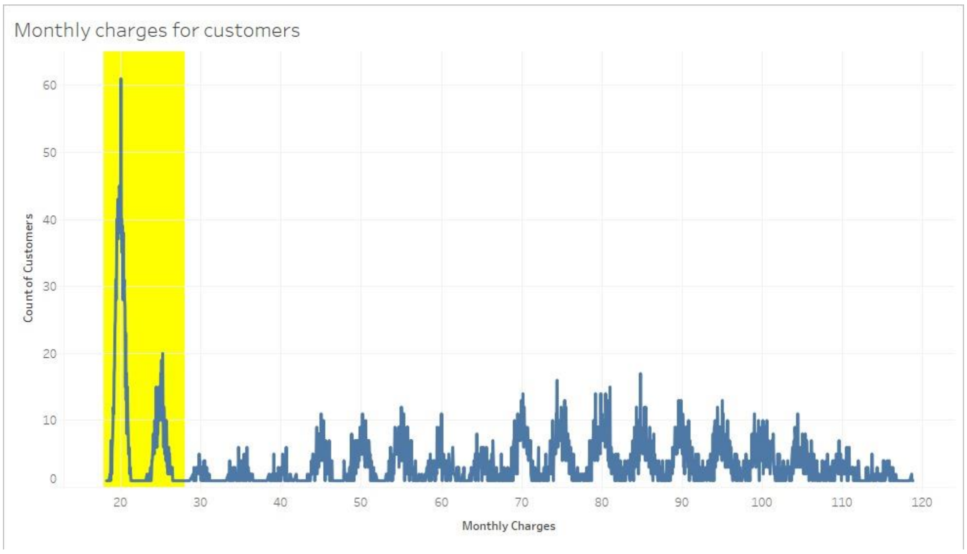


Fig. 12: Customers opted for less monthly charges

In fig:13 the trend indicates that customers are highly enrolled for the low monthly Charges in the range $18-$28.

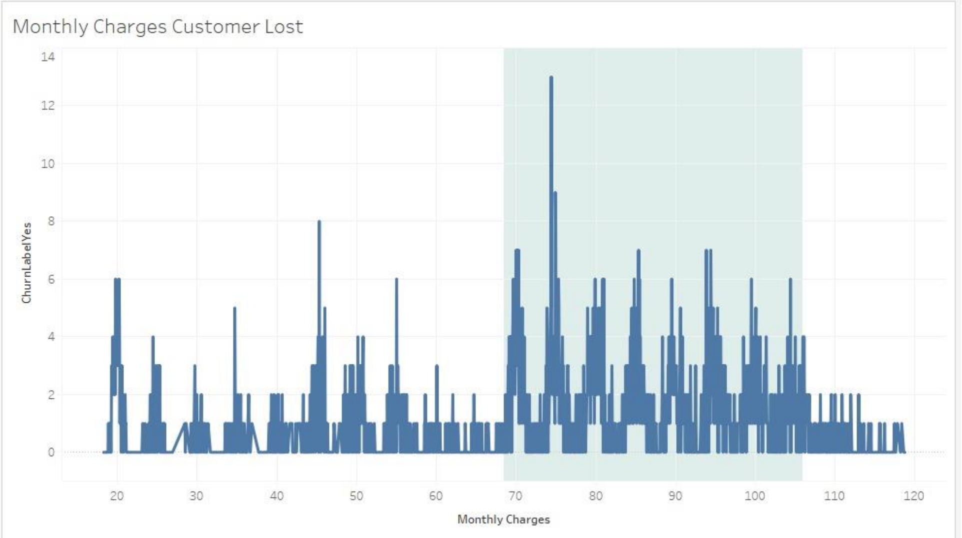


Fig. 13: Lost Customers with high Monthly Charges

In fig:14 We can see a trend that customers that are enrolled for higher monthly Charges between range $68.5-$106 have left the vendor.

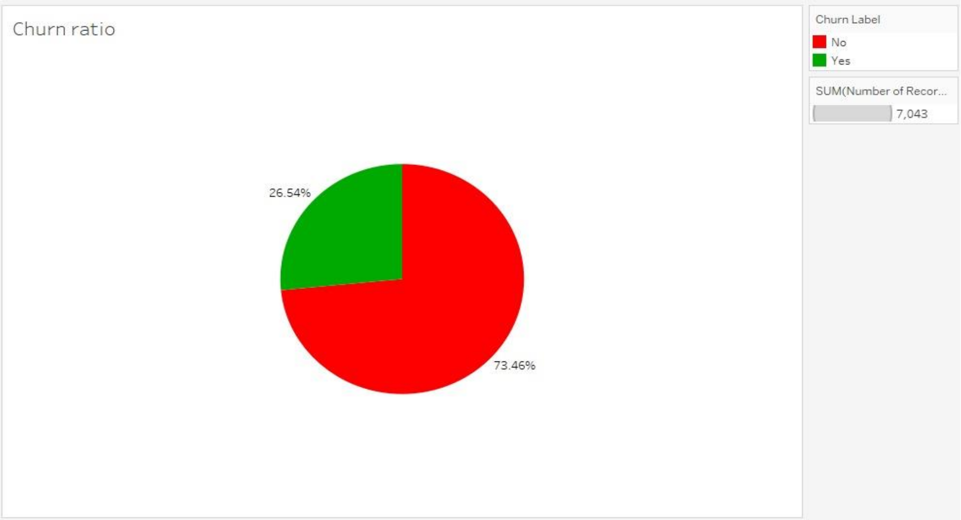
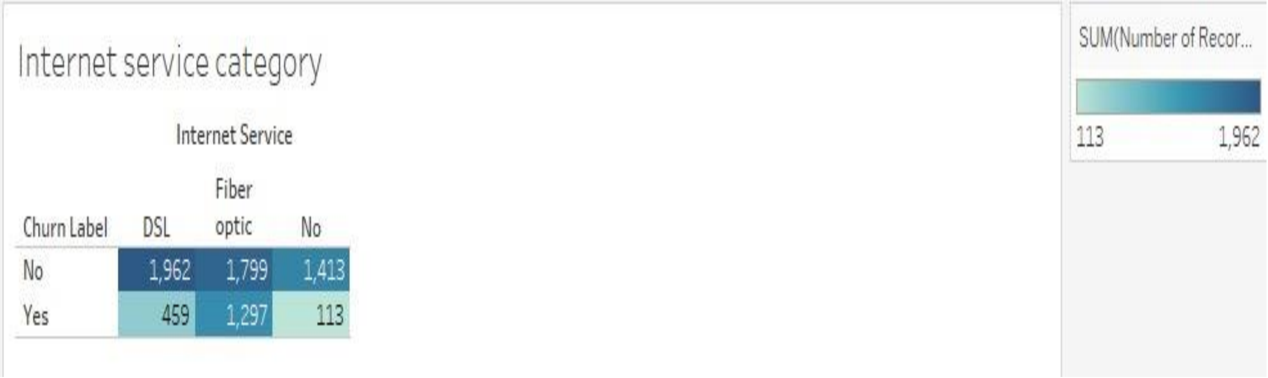


Fig. 14:Churn Ratio

In fig:15 The given chart indicates that 26% customers are churned


Internet service category

SUM(Number of Recor...

113 1,962

	Internet Service		
Churn Label	DSL	Fiber optic	No
No	1,962	1,799	1,413
Yes	459	1,297	113

Fig. 15: Churn By Internet Service

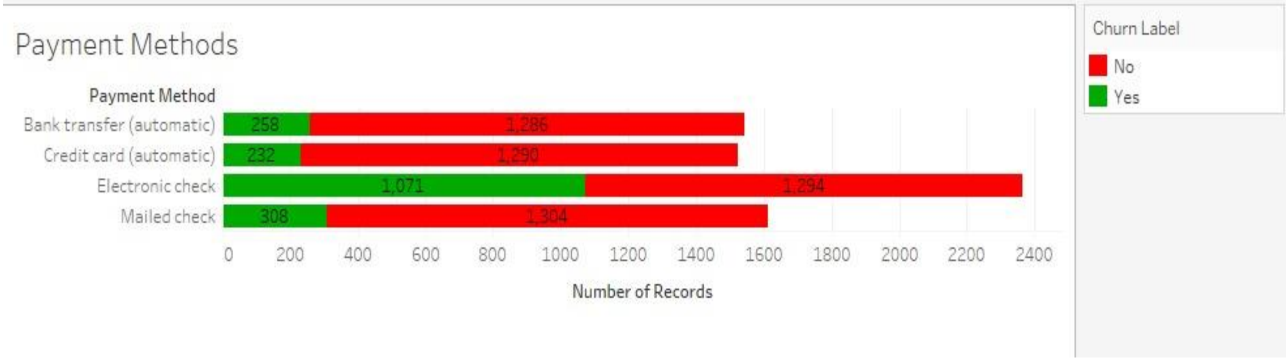


Fig. 16: Churn by Payment Methods

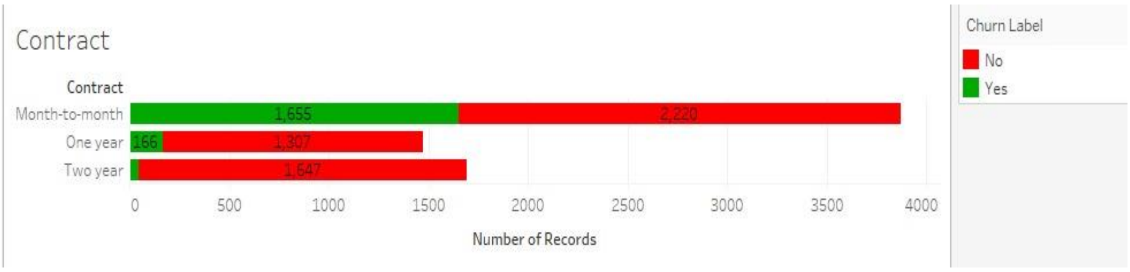


Fig. 17: Churn by Contract Type

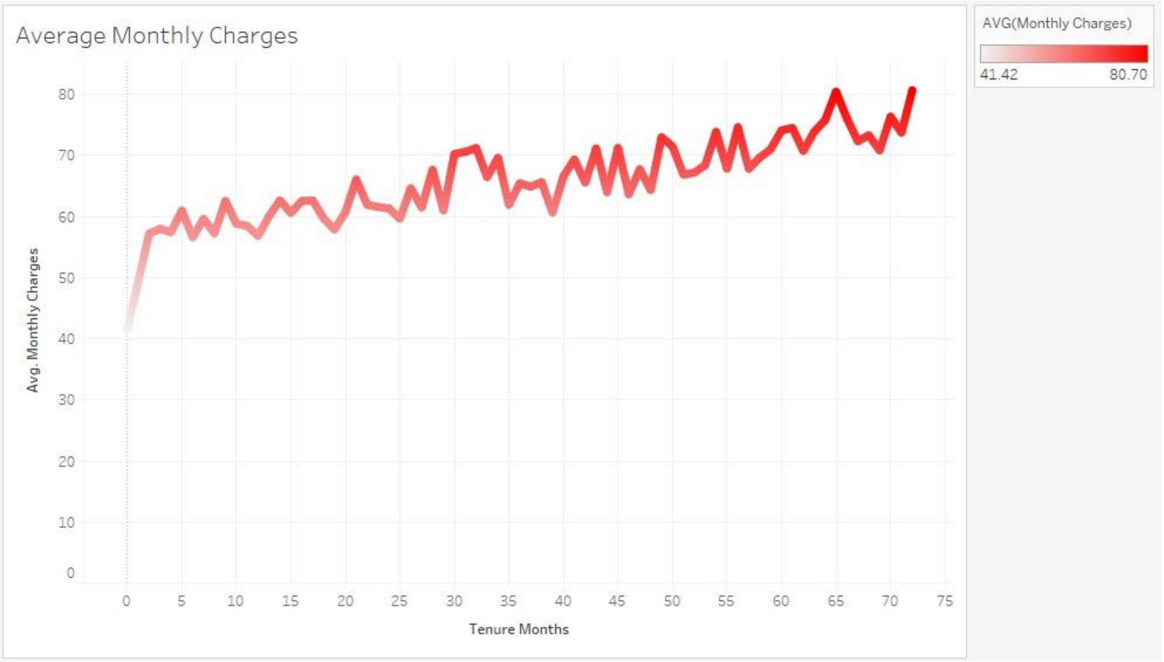
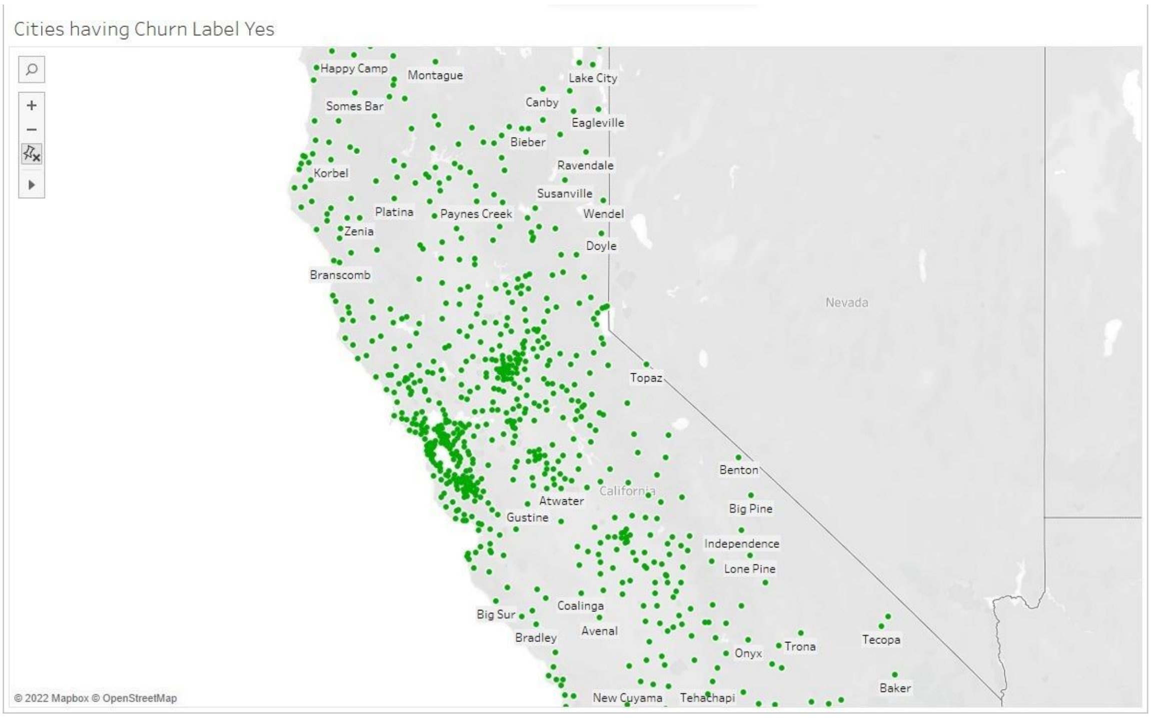


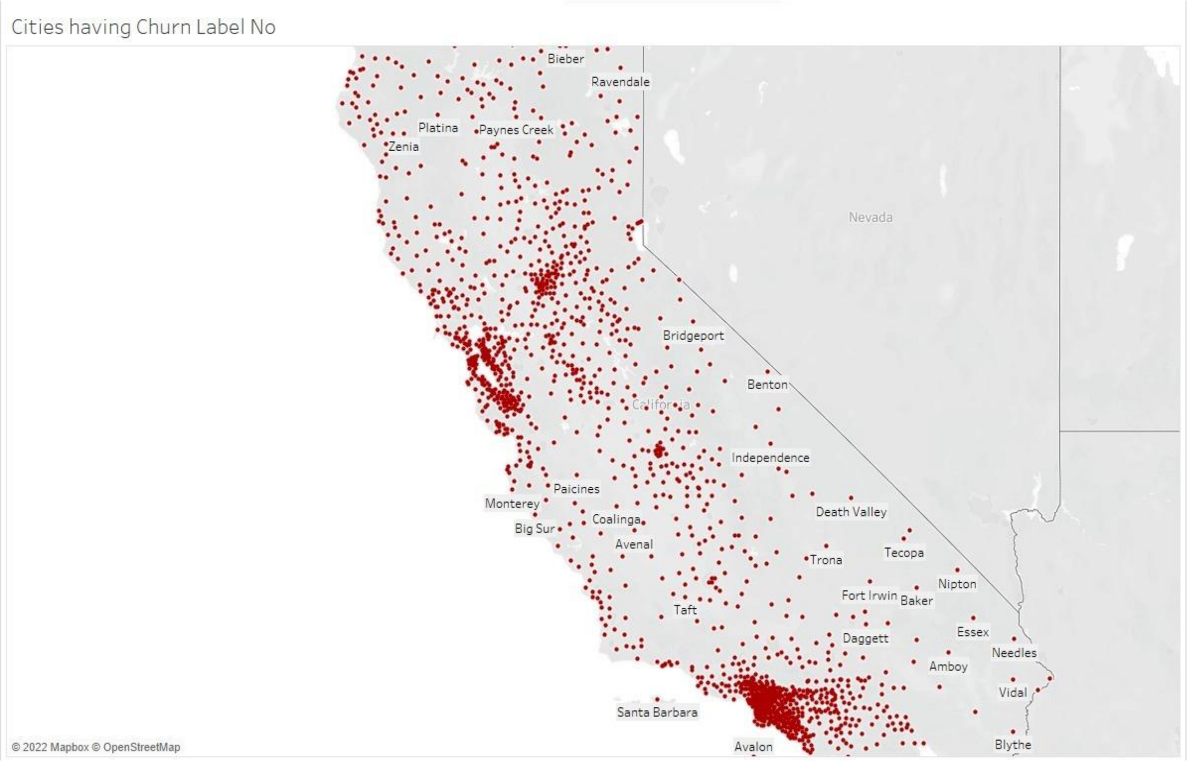
Fig. 18: Average Monthly Charges

Fig 19 depicts average monthly charge across all customers starts out at $41.42, increases sharply in the first three months to $57.21, and then continues to increase gradually to eventually reach the $80 price range.



Cities having Churn Label Yes

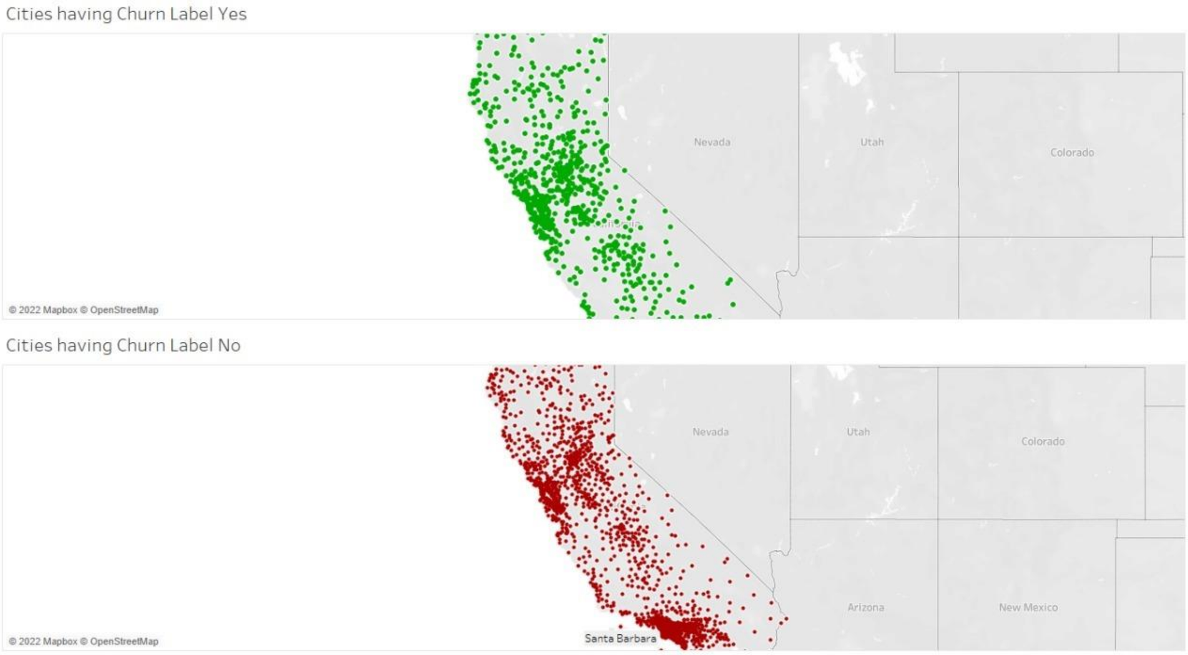
Fig. 19: Cities having Churn Label Yes



Cities having Churn Label No

Fig. 20: Cities having Churn Label No

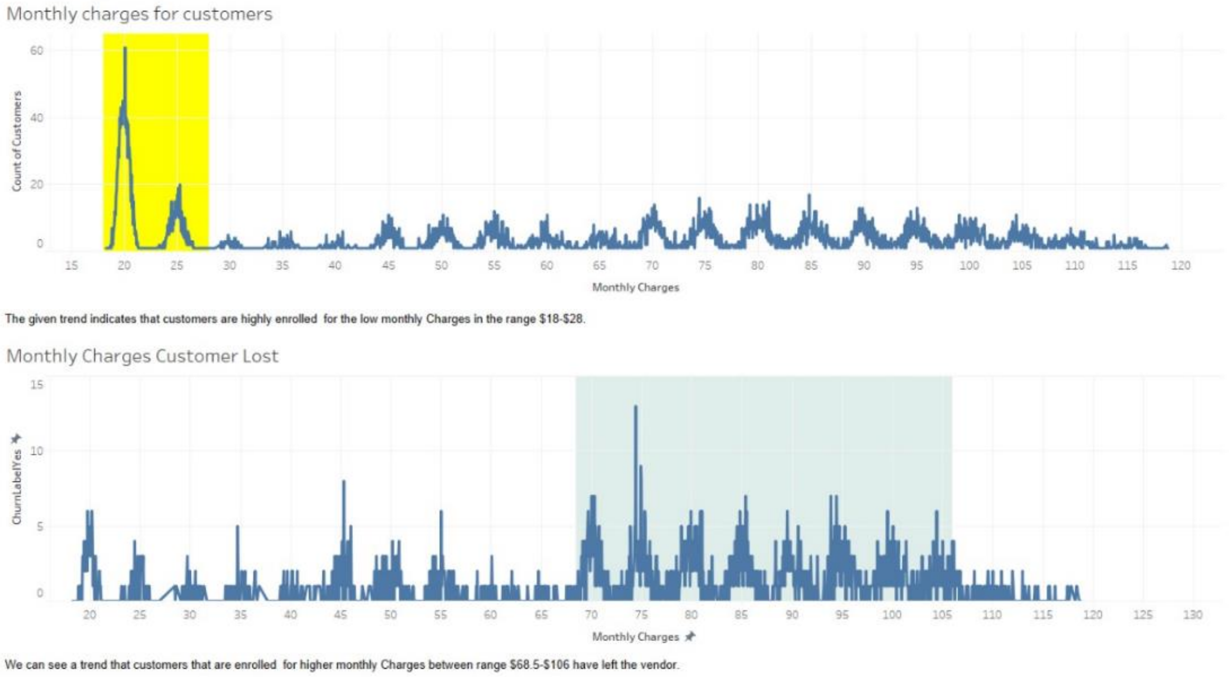
10. DASHBOARD



Cities having Churn Label Yes

Cities having Churn Label No

Fig. 21: Dashboard – I



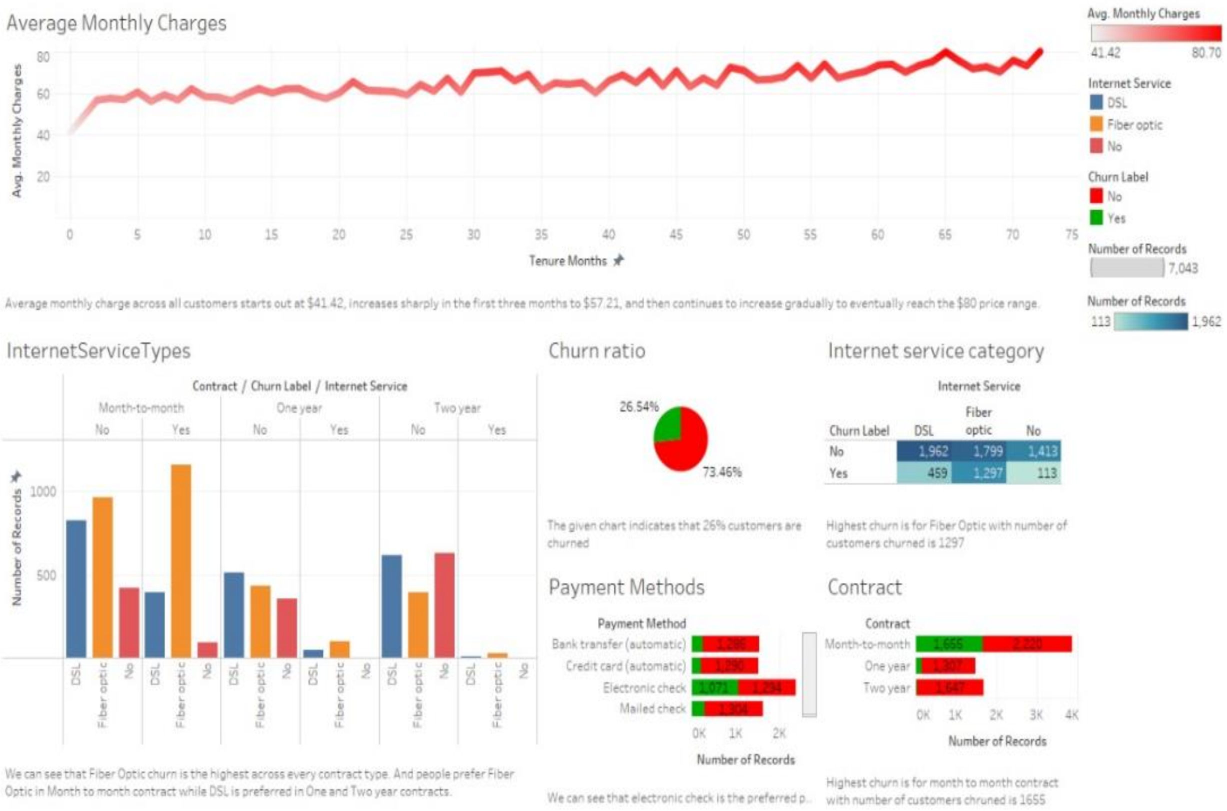
Monthly charges for customers

The given trend indicates that customers are highly enrolled for the low monthly Charges in the range $18-$28.

Monthly Charges Customer Lost

We can see a trend that customers that are enrolled for higher monthly Charges between range $68.5-$106 have left the vendor.

Fig. 22: Dashboard - II



Average Monthly Charges

Average monthly charge across all customers starts out at $41.42, increases sharply in the first three months to $57.21, and then continues to increase gradually to eventually reach the $80 price range.

InternetServiceTypes

We can see that Fiber Optic churn is the highest across every contract type. And people prefer Fiber Optic in Month to month contract while DSL is preferred in One and Two year contracts.

Churn ratio

The given chart indicates that 26% customers are churned

Internet service category

	Internet Service		
Churn Label	DSL	Fiber optic	No
No	1,962	1,799	1,413
Yes	459	1,297	113

Highest churn is for Fiber Optic with number of customers churned is 1297

Payment Methods

We can see that electronic check is the preferred p..

Contract

Highest churn is for month to month contract with number of customers chruned is 1655

Fig. 23: Dashboard - III

11.CONCLUSION

The growth tendency in the twenty-first century has been the most dramatic ever. With the advancement of technology comes a rise of services, and it is difficult for a business to determine which clients are likely to depart. Churn prediction in the telecom industry is an issue that has piqued the interest of many experts in recent years. We used RFE feature selection technique to find out the most relevant features. We used Logistic Regression, Decision Tree and Random Forest Machine Learning algorithms to predict customer churn on IBM Dataset. Logistic Regression gave an accuracy of 80%.