USING NAÏVE BAYES TO PREDICT CUSTOMER CHURN IN R (WEEK 4 ASSIGNMENT)

Course: ALY 6020

Name: Yuanying Li

1.Introduction

In the telecom industry, customers are able to choose from multiple service providers and actively switch from one operator to another. In this highly competitive market, the telecommunications industry experiences an average of 15-25% annual churn rate. Given the fact that it costs 5-10 times more to acquire a new customer than to retain an existing one, customer retention has now become even more important than customer acquisition.

For many incumbent operators, retaining high profitable customers is the number one business goal. To reduce customer churn, telecom companies need to predict which customers are at high risk of churn.

In this project, we will analyse customer-level data and build predictive models to identify customers at high risk of churn and identify the main indicators of churn using Navie Bayes Classification.

2. DATA AND PREPROCESSING

The data was downloaded from IBM Sample Data Sets. Each row represents a customer, each column contains that customer's attributes:

2.1 Define the decision variables

Here is the explanation of features' name in dataset.

- customerID: Customer ID
- Gender: Customer gender (female, male)
- 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)

Customers usually do not decide to switch to another competitor instantly, but rather over a period of time (this is especially applicable to high-value customers). In next step, we are focusing which features would influence customer churn.

2.2 Exploratory Data Analysis

Let's explore the data and see whether we can shine some light on the relationships. In doing so, we will prepare the data for use with the Naïve Bayes. We'll begin by importing the CSV data file, as we have done, saving the data to the data frame and change ggplot theme to beauty plots:

Code snippets:

```r
#Load the data and view it
data <- read.csv("telecom_churn.csv")
View(data)

#ggplot theme
theme <- theme(
  axis.text.y = element_blank(), axis.ticks.y = element_blank(),
  legend.position="none"
)

#Check Null values Number
sum(is.na(data))

#Prodide information about the structure of data
str(data)

#Data summary
summary(data)
```

```r
#Load the data and view it
data <- read.csv("telecom_churn.csv")
View(data)

#ggplot theme
theme <- theme(
  axis.text.y = element_blank(), axis.ticks.y = element_blank(),
  legend.position="none"
)
```

Results:

	customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines
1	7590-VHVEG	Female	0	Yes	No	1	No	No phone service
2	5575-GNVDE	Male	0	No	No	34	Yes	No

```
> sum(is.na(data))
[1] 11
```

```
> str(data)
'data.frame':   7043 obs. of  21 variables:
 $ customerID      : Factor w/ 7043 levels "0002-ORFBO","0003-MKNFE",..: 5376 3963 2565
 5536 6512 6552 1003 4771 5605 4535 ...
 $ gender          : Factor w/ 2 levels "Female","Male": 1 2 2 2 1 1 2 1 1 2 ...
 $ SeniorCitizen   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Partner         : Factor w/ 2 levels "No","Yes": 2 1 1 1 1 1 1 1 2 1 ...
 $ Dependents      : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 2 1 1 2 ...
 $ tenure          : int  1 34 2 45 2 8 22 10 28 62 ...
 $ PhoneService    : Factor w/ 2 levels "No","Yes": 1 2 2 1 2 2 2 1 2 2 ...
 $ MultipleLines   : Factor w/ 3 levels "No","No phone service",..: 2 1 1 2 1 3 3 2 3 1
```

```
     customerID      gender     SeniorCitizen    Partner    Dependents
0002-ORFBO:   1   Female:3488   Min.   :0.0000   No :3641   No :4933
0003-MKNFE:   1   Male  :3555   1st Qu.:0.0000   Yes:3402   Yes:2110
0004-TLHLJ:   1                 Median :0.0000
0011-IGKFF:   1                 Mean   :0.1621
0013-EXCHZ:   1                 3rd Qu.:0.0000
0013-MHZWF:   1                 Max.   :1.0000
(Other)   :7037
```

Observations:

1. Sample size has 7043 rows, there are 21 columns with 19 features and Only 11 missing values (next item). the dependent variable is called churn.

2. The variable is an integer variable named id. As this is simply a unique identifier (ID) for each patient in the data, it does not provide useful information, and we will need to exclude it from the model.

2.3 Data preprocessing

2.3.1 Drop unimportant columns and impute missing value

In this study we firstly focus on 18 factors that possibly influence churn, as we mentioned before, ID is not necessary to keep for predicting the final diagnosis, so that we could just drop it. What's more, we could also need to impute 11 missing value.

Code snippets:

```
# Remove columns we didn't see correlation from above
data <- data %>%
  select(
    -customerID, -gender,-PhoneService, -MultipleLines, -Monthl
  )
```

```r
#Check null data in which column
data %>%
  summarise_all(
    funs(sum(is.na(.)))) %>%

gather(ColumnTitle, NAs, customerID:Churn)

data %>%
  select(
    customerID, TotalCharges, TotalCharges
  ) %>%
  filter(
    is.na(TotalCharges)
  )

#Replace null value in titalcharges into 0
data[is.na(data)] <- 0
```

```r
#Check null data in which column
data %>%
  summarise_all(
    funs(sum(is.na(.)))) %>%

gather(ColumnTitle, NAs, customerID:Churn)
```

```
   customerID TotalCharges
1  4472-LVYGI           NA
2  3115-CZMZD           NA
3  5709-LVOEQ           NA
4  4367-NUYAO           NA
5  1371-DWPAZ           NA
6  7644-OMVMY           NA
7  3213-VVOLG           NA
8  2520-SGTTA           NA
9  2923-ARZLG           NA
10 4075-WKNIU           NA
11 2775-SEFEE           NA
```

Results:

```
 gather(ColumnTitle, NAs, customerID:Churn)
      ColumnTitle NAs
       customerID   0
           gender   0
    SeniorCitizen   0
          Partner   0
       Dependents   0
           tenure   0
     PhoneService   0
    MultipleLines   0
  InternetService   0
0  OnlineSecurity   0
1    OnlineBackup   0
2 DeviceProtection   0
3     TechSupport   0
4     StreamingTV   0
5 StreamingMovies   0
6        Contract   0
7 PaperlessBilling   0
8   PaymentMethod   0
9  MonthlyCharges   0
0    TotalCharges  11
1           Churn   0
```

Observation:

1. There are only 11 missing values, all of them for the TotalCharges column. This values are actually a blank space in the csv file and are exclusive for customers with zero tenure. It's possible to concluded that they are missing due to the fact that the customer never paied anything to the company. We will impute this missing values with zero.

2.3.2 Convert data type.

The data type pf SeniorCitizen is int, it is better to convert the data type to factor for modeling.

```
#Convert features which are belongs to int value into factor value column.
a <- sub("0","0", data$SeniorCitizen)
b <- sub("1","1",a)
data$SeniorCitizen <- b
data$SeniorCitizen <- as.factor(data$SeniorCitizen)
```

2.4 Date visualization

2.4.1 Churn analysis

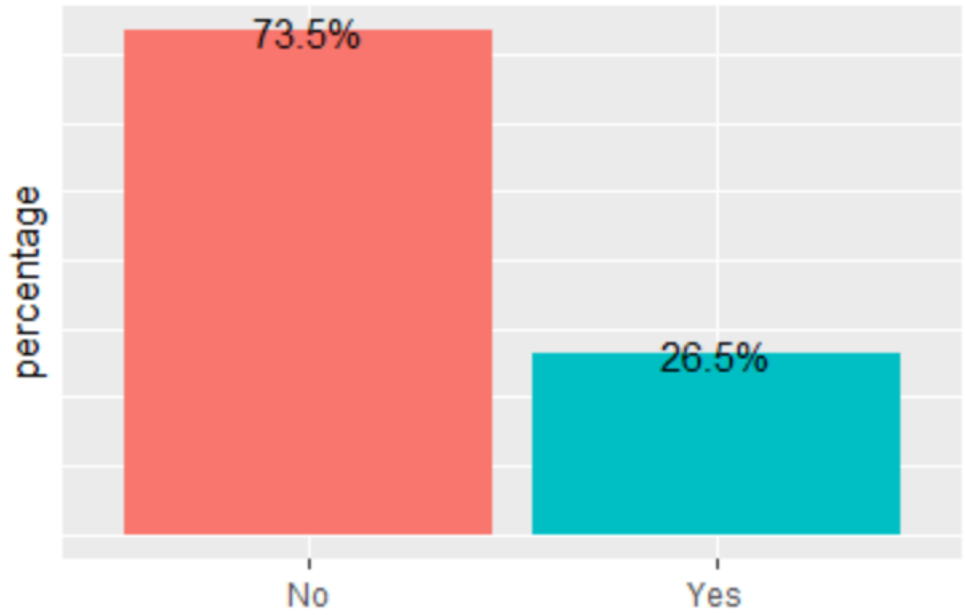
We are trying to predict if the client left the company in the previous month. Therefore we have a binary classification problem with a slightly unbalanced target:

Code snippets:

```
#Check proportion of churn
options(repr.plot.width = 4, repr.plot.height = 3)

data %>%
  group_by(Churn) %>%
  summarize(
    n = n()
  ) %>%
  mutate(
    percentage = round(n / sum(n), 3),
    n = NULL
  ) %>%
  ggplot(aes(x = Churn, y = percentage)) + geom_col(aes(fill = Churn)) +
  theme +
  geom_text(
    aes(x = Churn, y = percentage, label = paste(percentage*100, "%", sep = ""))
  )
```

Results:



Observation:

1. The output indicates that 26.5% customer decided to choose another company means that those customer churn, but majority of people, which is 73.5% retain to stay in this service.

2.4.2 Numerical features distribution

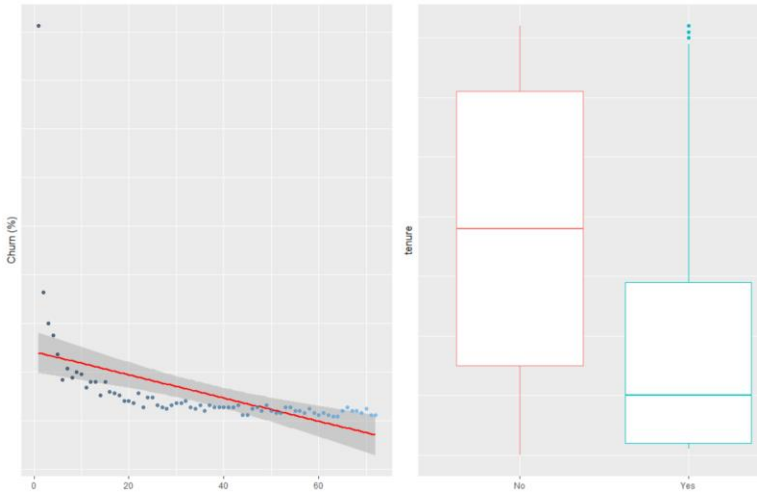
- Tenure
- Monthly charges
- Total charges

There are only three numerical columns: tenure, monthly charges and total charges. The probability density distribution can be estimate using ggplot function.

Code snippets:

```
plot_grid(
  data %>%
    filter(Churn == "Yes") %>%
    group_by(tenure) %>%
    summarize(
      n = n()
    ) %>%
```

Results:



Observation:

1. From the plots above we can conclude that, recent clients are more likely to churn. Tenure and Monthly Charges are probably important features.

2.4.3 Categorical features distribution

This dataset has 16 categorical features:

1. Six binary features (Yes/No)

2. Nine features with three unique values each (categories)

3. One feature with four unique values

2.4.3a

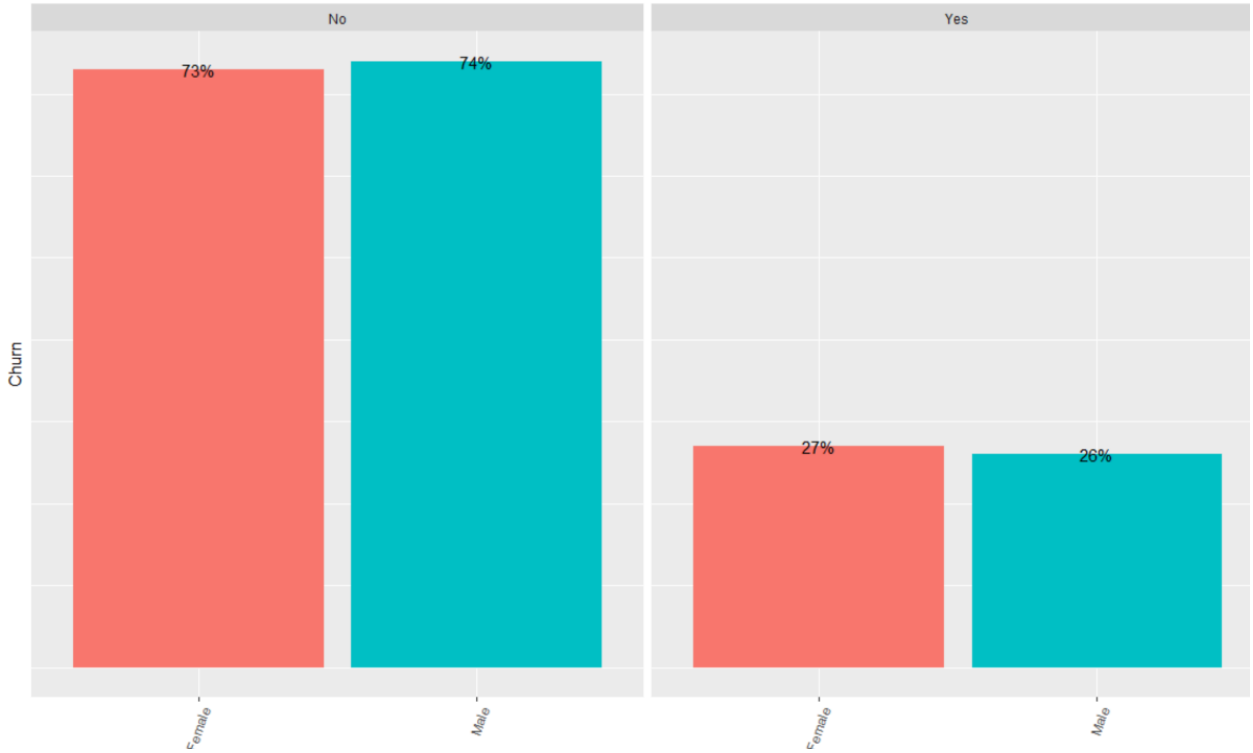
- Gender
- Senior Citizen

Code snippets:

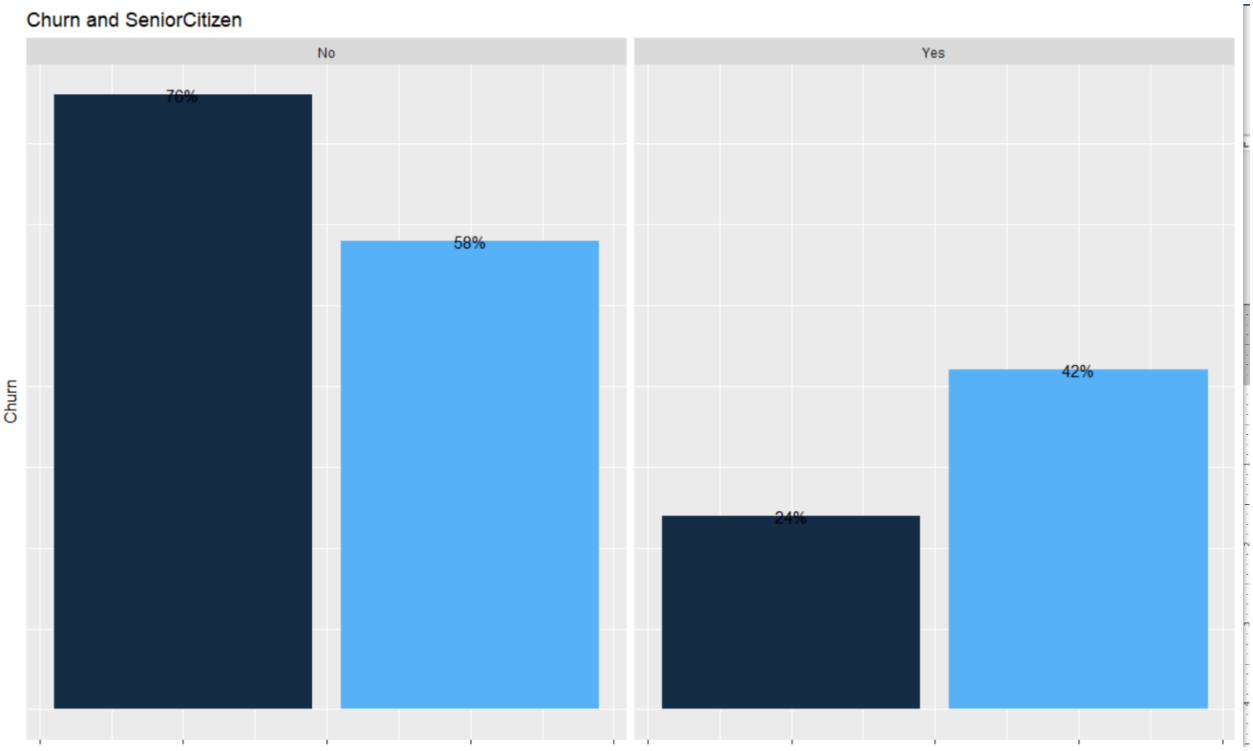
```
# Save plot in a variable so plots can be displayed sequentially
p <- ggplot(
  data = a, aes_string(
    x = colnames(a[1]), y = colnames(a[4]), fill = colnames(a[1])
  )
) +
```

Results:

Churn and gender



Churn and SeniorCitizen



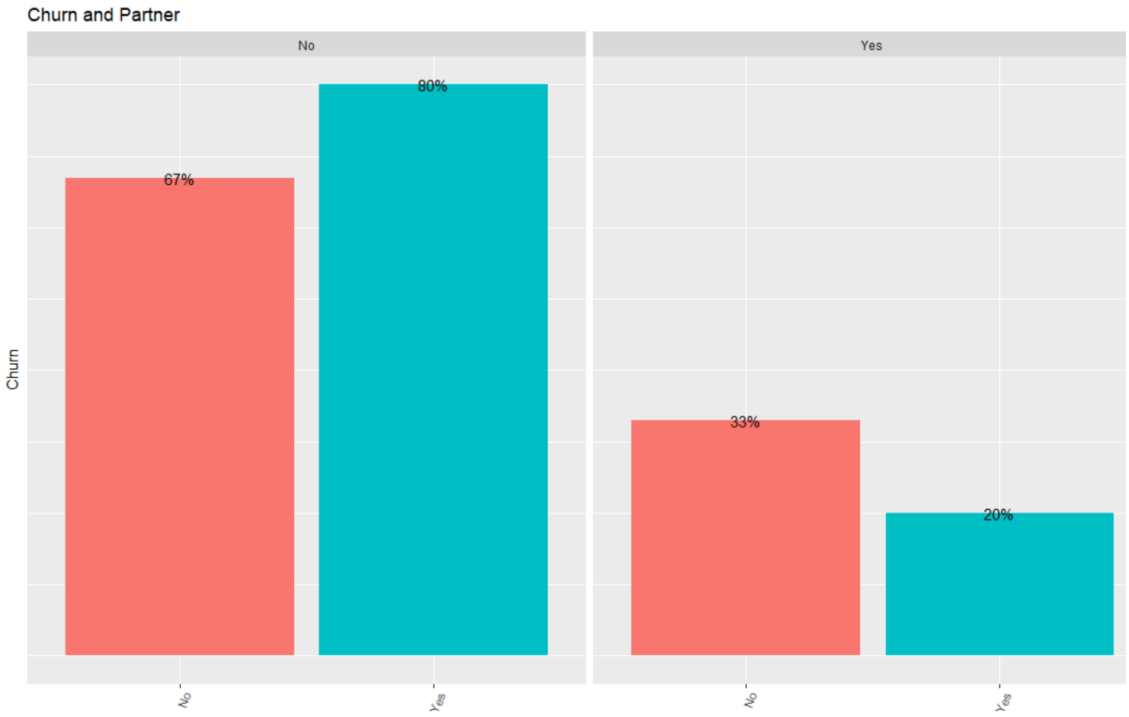
Observation:

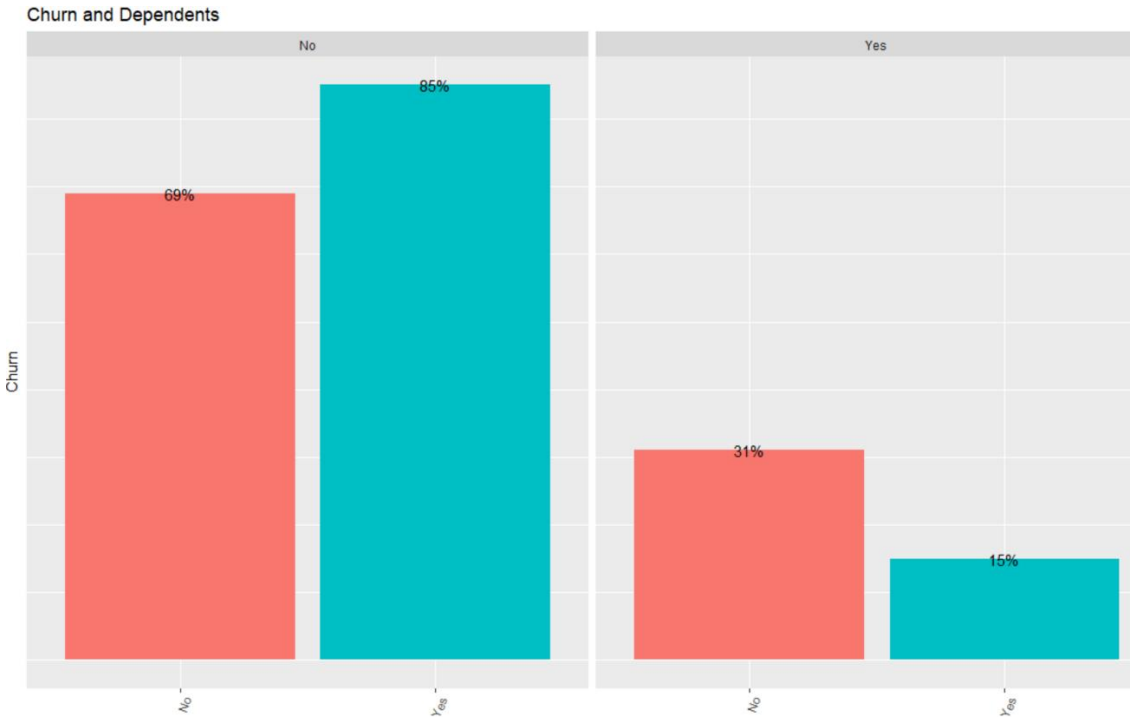
1. Gender is not an indicative of churn.

2. Senior Citizens are only 24% of customers, but they have a much lower churn rate: 42% against 58% for non-senior customers.

2.4.3b

- Dependents
- Partner





Observation:

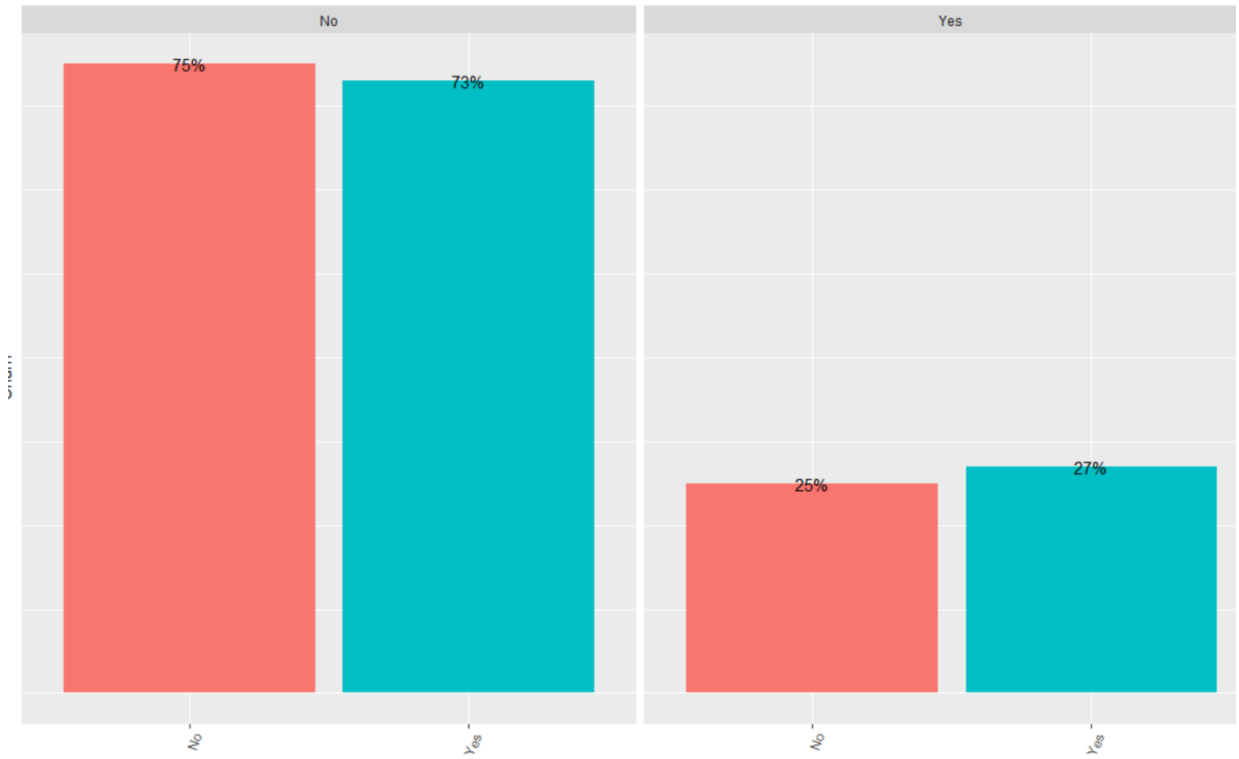
1. Customers that doesn't have partners are more likely to churn

2. Customers without dependents are also more likely to churn

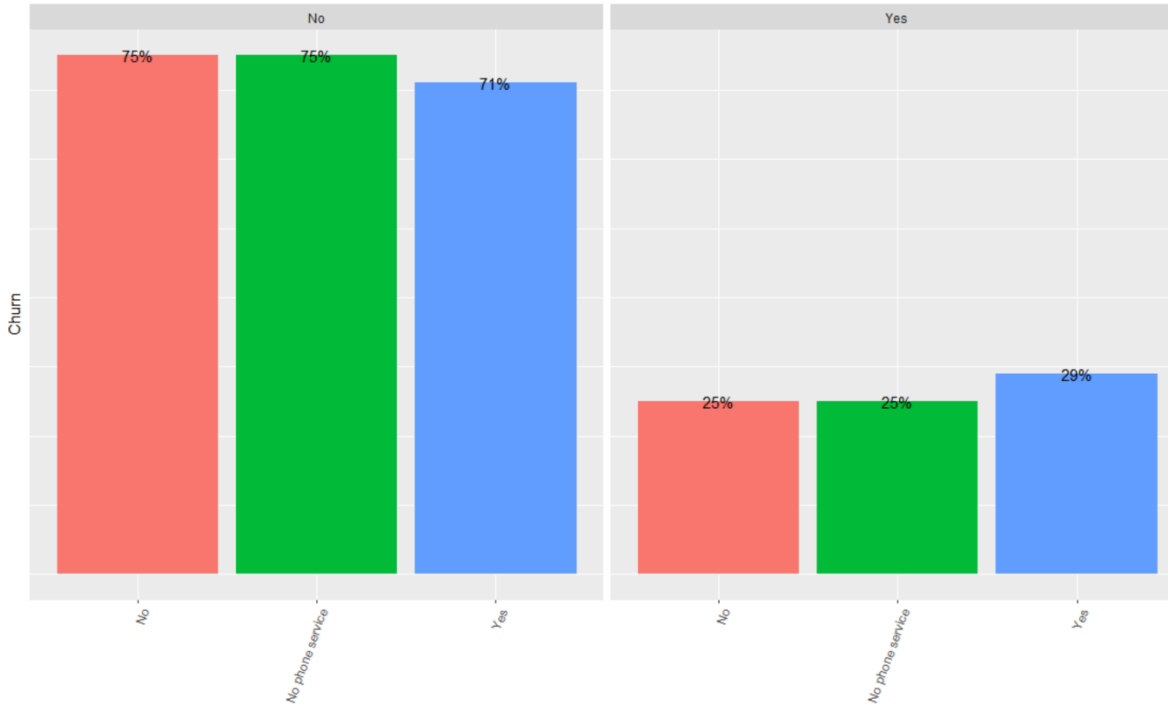
2.4.3c

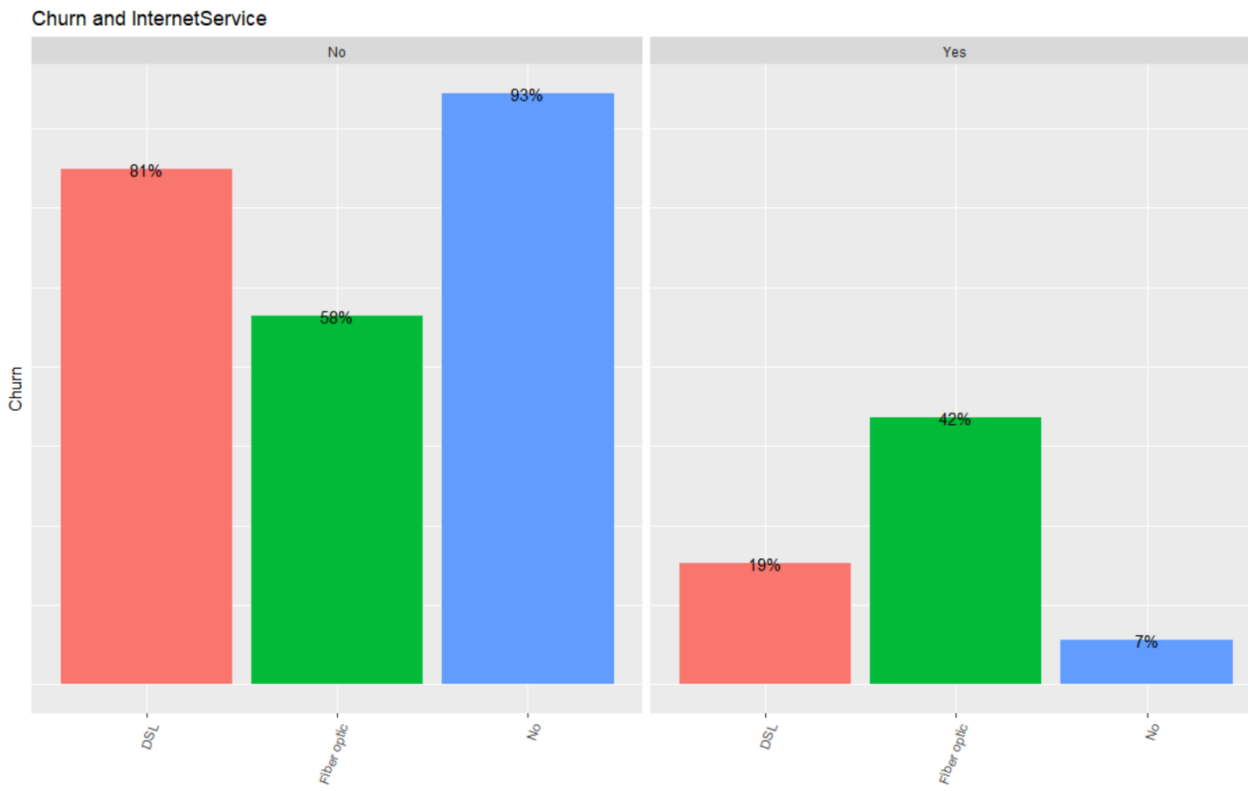
- Phone Service
- Internet Service
- Multiple Line

Churn and PhoneService



Churn and MultipleLines





Observation:

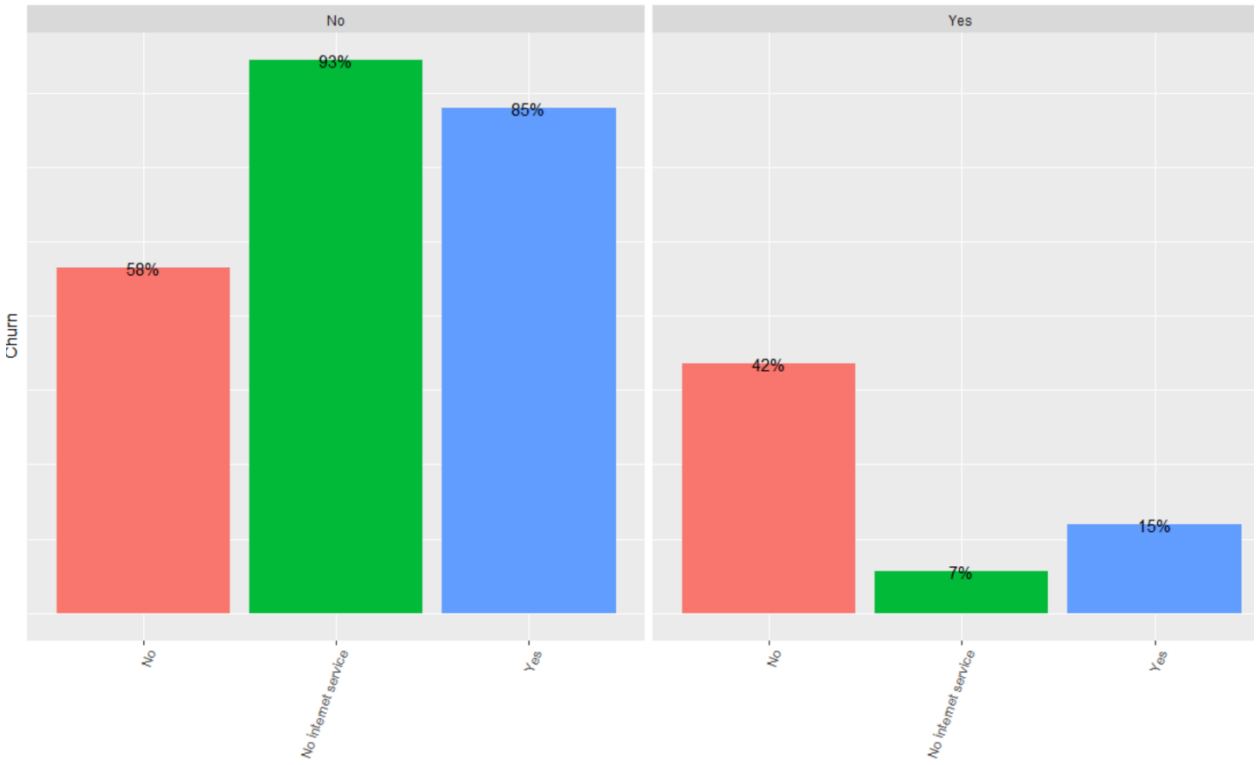
1. Multiple lines and Phone service has little influence on churn rate.

2. Customers with multiple lines have a slightly higher churn rate

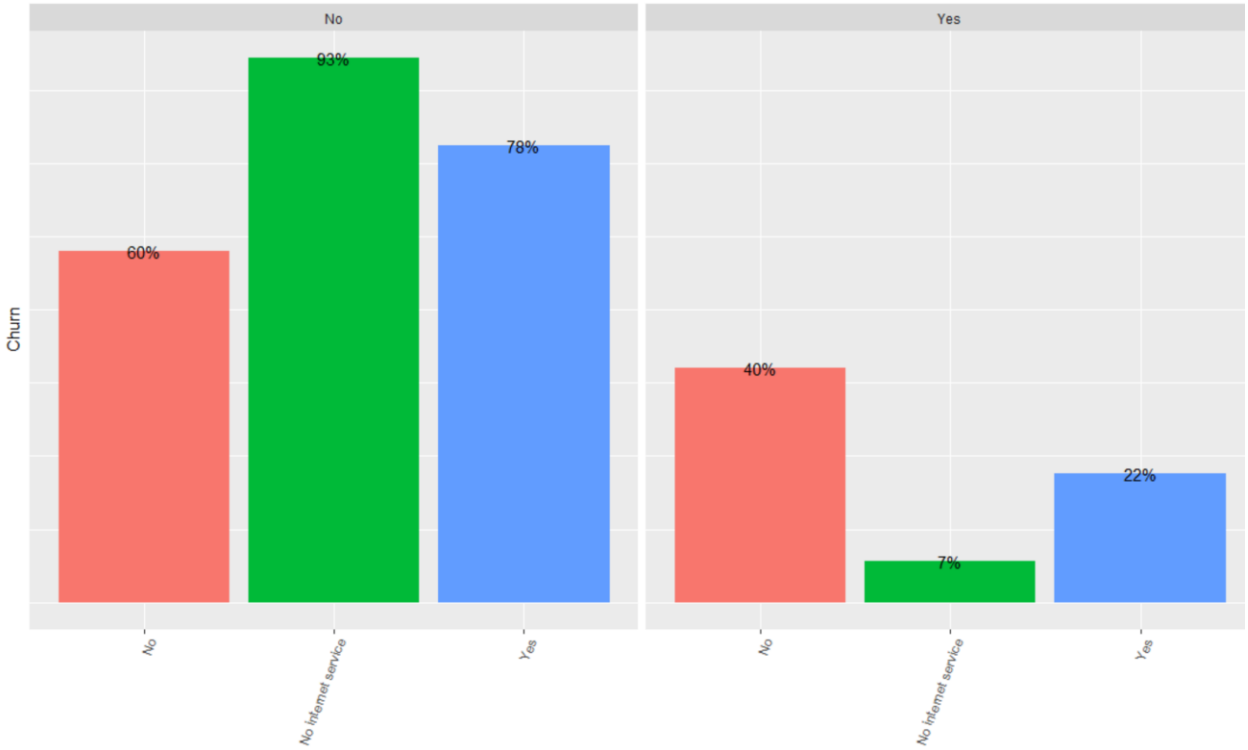
2.4.3d

- Additional service

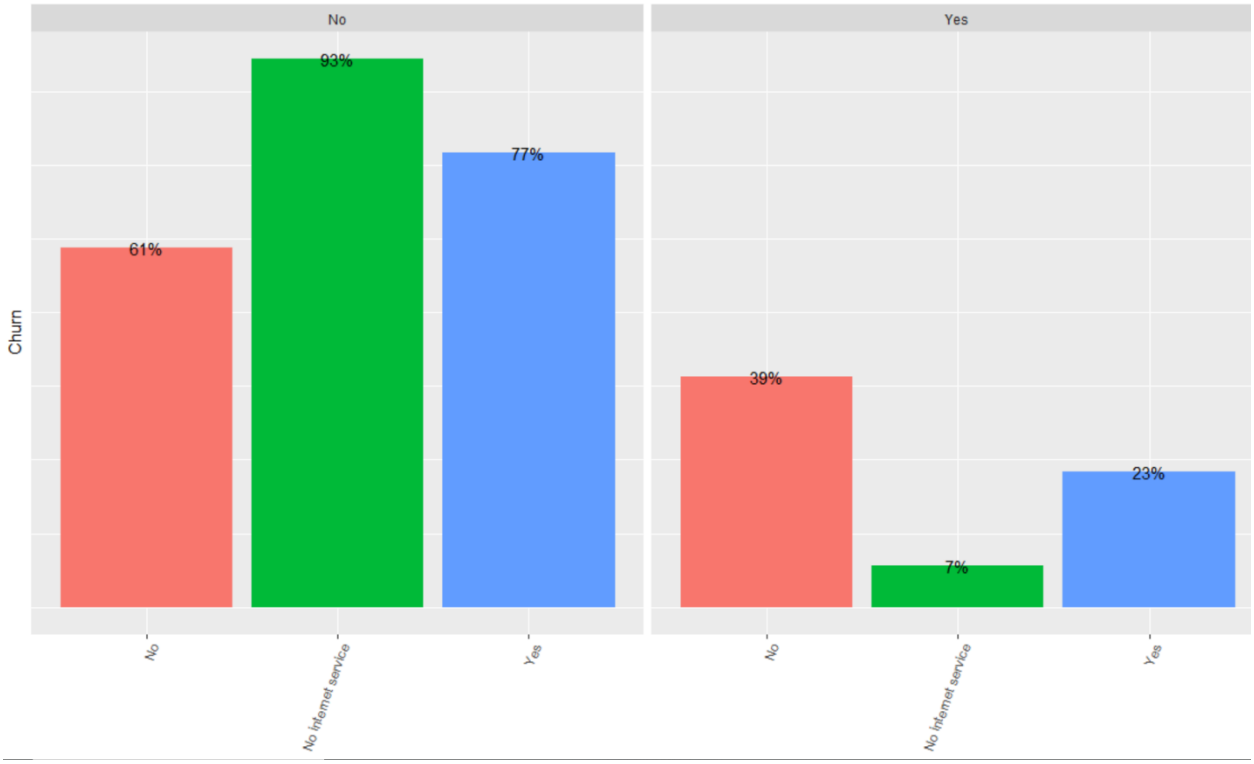
Churn and OnlineSecurity



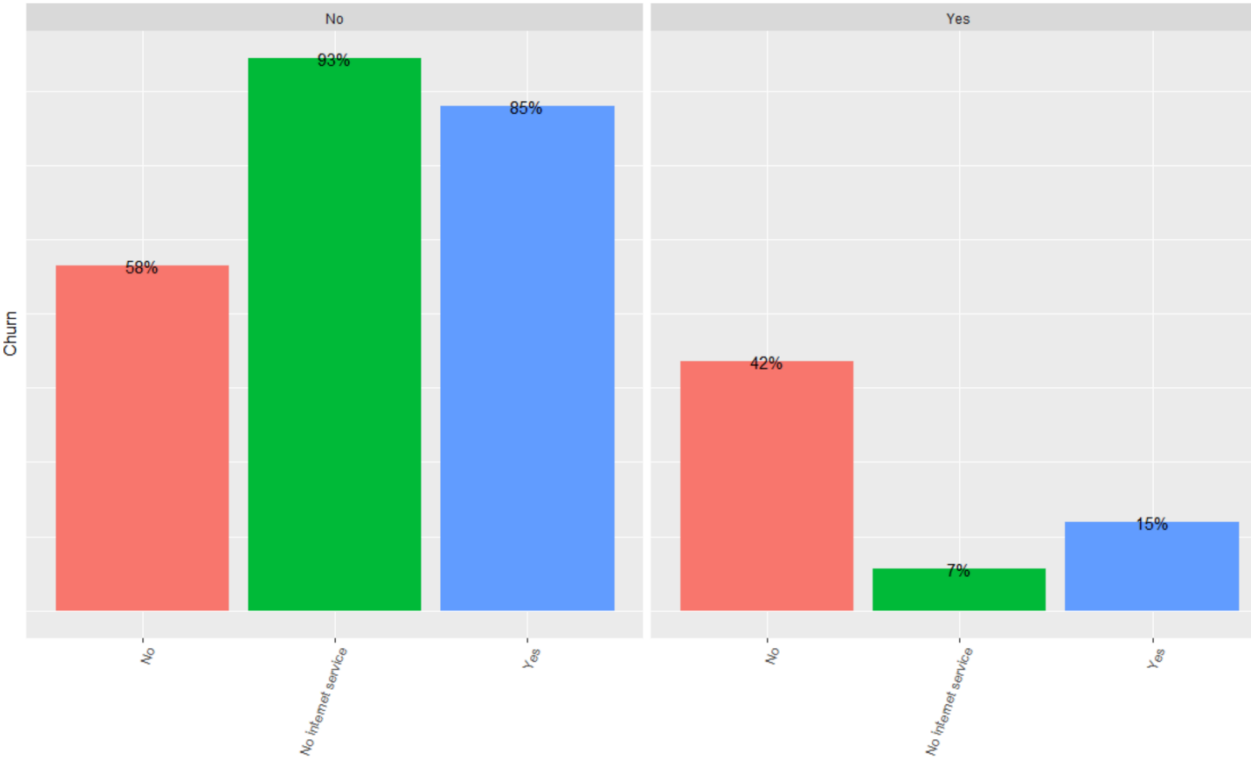
Churn and OnlineBackup



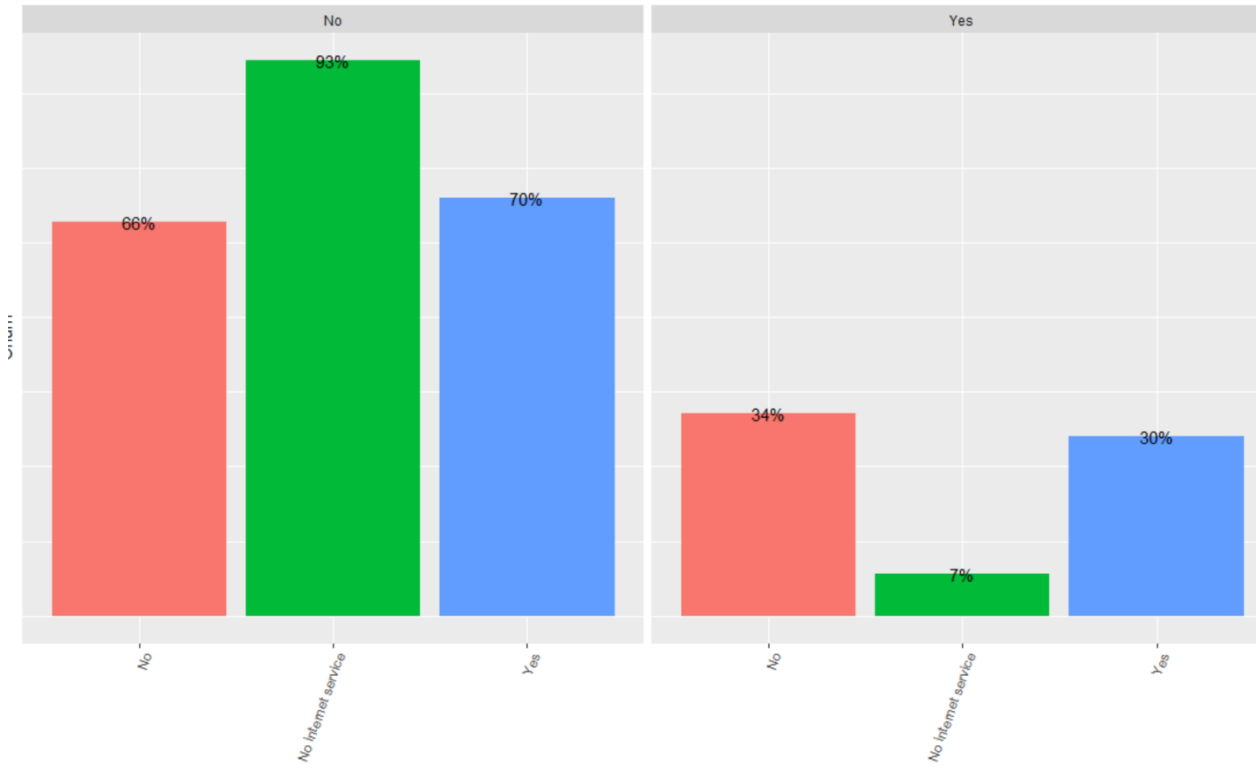
Churn and DeviceProtection



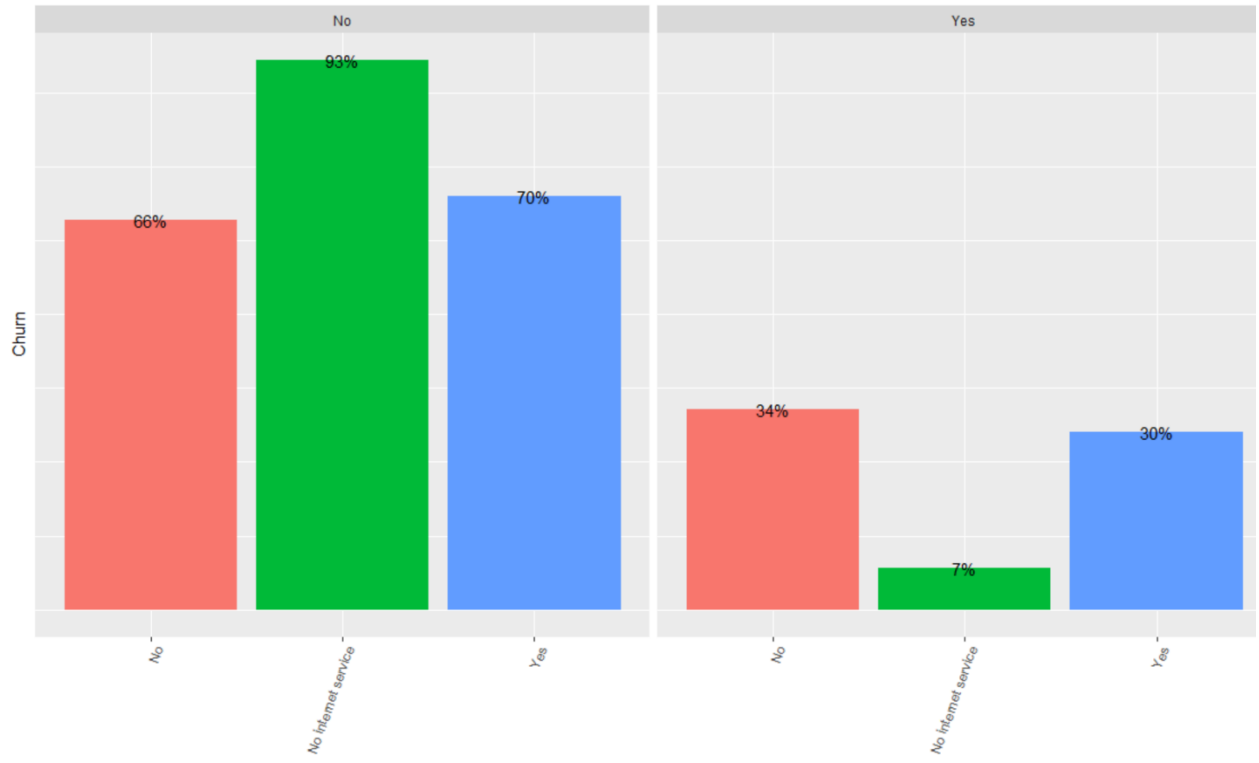
Churn and TechSupport



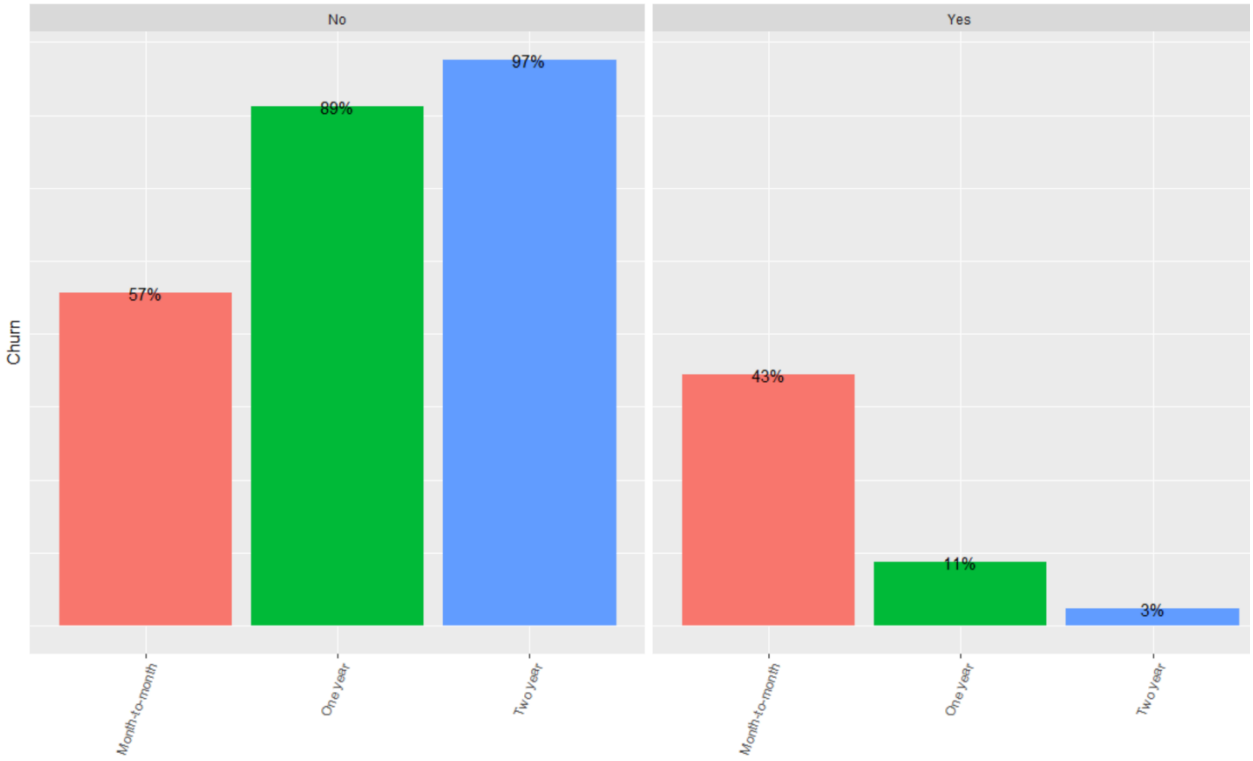
Churn and StreamingTV



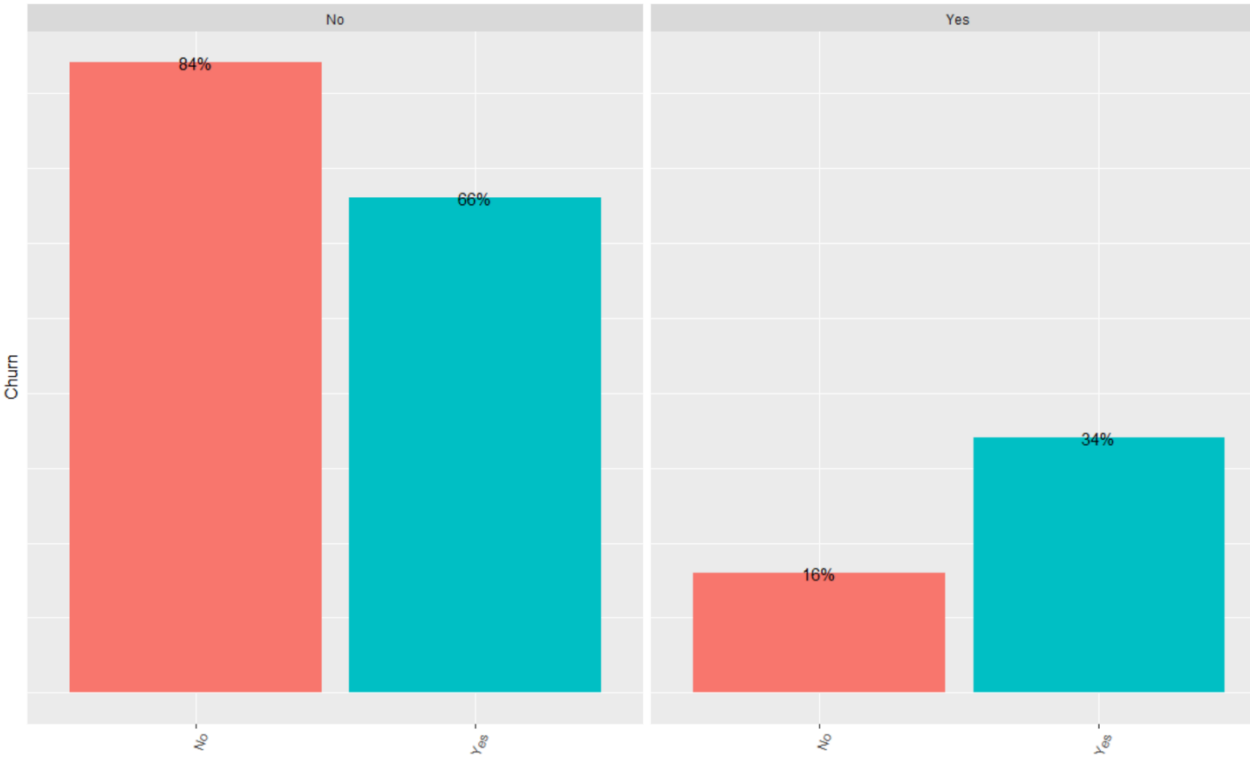
Churn and StreamingMovies

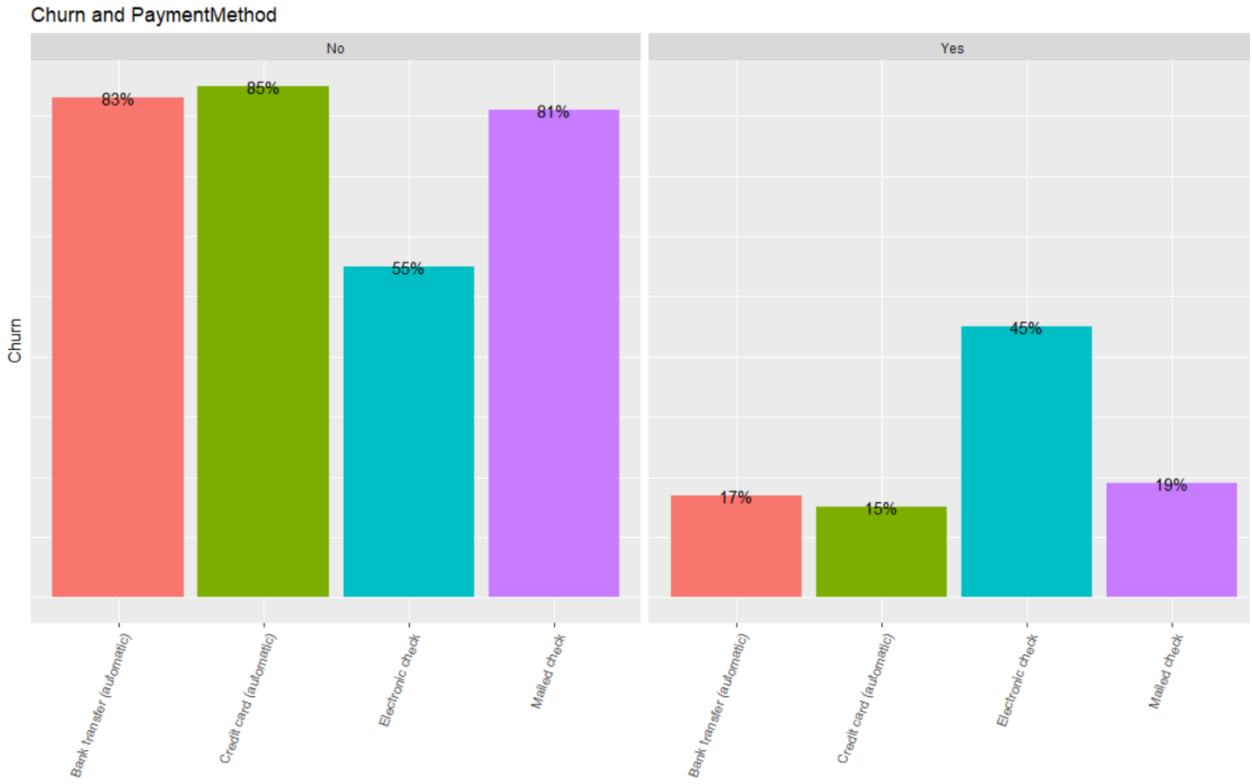


Churn and Contract



Churn and PaperlessBilling





Observation:

1. Online security service, online backup, Tech supports and Device Protection have a great impact on churn rate for people using internet, if don't have those services, these people are more likely to churn. However, people who have not internet service will not be affected by these online services.

2. Streaming TV and Streaming Movie have little influence on churning rate.

3. The longer time people keep their contract, the more likely they are going to stay to the same provider. There is a huge gap in charges between customers that churn and those that don't with respect to Mailed Check

4. people using Electrical check are more likely to churn.The preferred payment method is Electronic check with around 35% of customers. This method also has a very high churn rate.

3.BUILD THE MODEL

3.1 Use Naïve Bayes

3.2.1 Split train dataset and test train

In this section, we can simulate this scenario by dividing our data into two portions: a training dataset that will be used to build the Naïve Bayes classification and a test dataset that will be used to estimate the predictive accuracy of the model.

Using the data extraction methods, Managing and Understanding Data, we will split the data frame into data_train and data_test:

Code snippets:

```
#TRAIN, TEST & SPLIT
#Data splicing basically involves splitti
#set seed
n <- nrow(data)
n_train <- round(0.8*n)
n_train
set.seed(2020)
train_indices <- sample(1:n, n_train)
data_train <- data[train_indices, ]
data_test <- data[-train_indices, ]
```

3.1.2 Train the model

Equipped with our training data and labels vector, we are now ready to classify.

We now have nearly everything that we need to apply the k-NN algorithm to this data. We've split our data into training and test datasets, each with exactly the same numeric features.

Now we can use the knn() function to classify the test data:

Code snippets:

```
#Apply model
bayes <- naiveBayes(Churn~., data = data_train, laplace = 1)
```

3.2 Evaluation Model

After using a classification, evaluation is one of the important parts to do, so we predict the churn from test model and check the accuracy comparing with the real price and visualize the residuals through plot the difference . To evaluate its performance, we use the ROC Curve and ConfusionMatric.

Code snippets:

```
#Evalusate model(confusionMatrix)
pred <- predict(bayes, data_test)
confusionMatrix(pred, data_test$Churn)

#Evalusate model
rawpred <- predict(bayes, data_test, type = "raw")
ptest <- prediction(rawpred[,2], data_test$Churn)
perf <- performance(ptest, "tpr", "fpr")
plot(perf, colorize = T)

performance(ptest, "auc")@y.values
```

```
#Evalusate model(confusionMatrix)
pred <- predict(bayes, data_test)
confusionMatrix(pred, data_test$Churn)
```

Result:

```
Confusion Matrix and Statistics

          Reference
Prediction  No Yes
       No  692  76
       Yes 341 300

               Accuracy : 0.704
                 95% CI : (0.6794, 0.7278)
    No Information Rate : 0.7331
    P-Value [Acc > NIR] : 0.9934

                  Kappa : 0.3821

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.6699
            Specificity : 0.7979
         Pos Pred Value : 0.9010
         Neg Pred Value : 0.4680
             Prevalence : 0.7331
         Detection Rate : 0.4911
   Detection Prevalence : 0.5451
      Balanced Accuracy : 0.7339

       'Positive' Class : No
```

```
> performance(ptest, "auc")@y.values
[[1]]
[1] 0.8007907
```

Observation:

1. The cell percentages in the table indicate the proportion of values that fall into four categories. The top-left cell indicates the true negative results. These 692 of 1409 values are cases where the customer churn and the Naïve Bayes algorithm correctly identified it as such. The bottom-right cell indicates the true positive results, where the classifier define the customer still use this service.is malignant. A total of 300 of 1409 predictions were true positives.

2. The accuracy of the model using Confusion Matrix is 70.4%, The next step is to find the area under the curve, The AUC of the model is 80%. which is higher than the Confusion Matrix.

4.CONCLUSION

From the above example, we can see that Naïve Bayes can be used for customer churn analysis for this particular dataset equally fine.Throughout the analysis, I have learned several important things:

Features such as tenure, Contract, PaperlessBilling, MonthlyCharges and InternetService appear to play a role in customer churn. There does not seem to be a relationship between gender and churn. Customers in a month-to-month contract, with PaperlessBilling and are within 12 months tenure, are more likely to churn; On the other hand, customers with one or two year contract, with longer than 12 months tenure, that are not using PaperlessBilling, are less likely to churn.

5.REFERENCE

1. Predict Customer Churn with R

https://towardsdatascience.com/predict-customer-churn-with-r-9e62357d47b4

2. Credit Card Fraud Detection: KNN & Naive Bayes

https://www.kaggle.com/yuridias/credit-card-fraud-detection-knn-naive-bayes