Text

Description automatically generated

**Course:** 2021-0707 IST687 Introduction to Data Science

**Name:** Jake Conard, Natali Newman, Andrews Frimpong, Greg Lee

**Assignment:** Final Project Group 3

**Date due:** 14September 2021

**Date submitted:** 14 September 2021

Table of Contents

[Introduction 3](#_Toc82503527)

[Captured Dataset Variables 3](#_Toc82503528)

[Business Question 3](#_Toc82503529)

[Materials & methods 3](#_Toc82503530)

[Segment Analysis Results 4](#_Toc82503531)

[Segment Analysis 1: Analysis of the Contract, Payment month and Charge 4](#_Toc82503532)

[Segment Analysis 2: Additional Services Analysis 6](#_Toc82503533)

[Segment Analysis 3: Customer Demographics 9](#_Toc82503534)

[Segment Analysis 4: Further Analysis including Tenure, Monthly and Total Charges 11](#_Toc82503535)

[Descriptive Statistics and Visualizations 11](#_Toc82503536)

[Probit Regression 13](#_Toc82503537)

[Kernel Support Vector Machine (KSVM) Learning 13](#_Toc82503538)

[Conclusion and Recommendations 14](#_Toc82503539)

[Project Reflection 15](#_Toc82503540)

[R code 15](#_Toc82503541)

[Segment 1: Analysis of the Contract, Payment month and Charge 15](#_Toc82503542)

[Segment 2: Additional Services Analysis 17](#_Toc82503543)

[Segment 3: Customer Demographics R code 19](#_Toc82503544)

[Segment 4: Tenure, Charges and Further Analysis Code 22](#_Toc82503545)

# Introduction

We are part of a data science team for one of the largest telecommunication companies in the United States. We have been capturing data on our customers for the last several months to determine the reasons that customers are leaving our company. We have captured 19 attributes on 7,043 customers to help determine the reason that customers are leaving our company. This report highlights our analysis of the data and provides some recommendations that we believe will improve the company’s overall customer churn rates. The dataset that we utilized for this report can be found at the following link (<https://www.kaggle.com/blastchar/telco-customer-churn>).

## Captured Dataset Variables

* Gender – Is the customer male or female
* Senior Citizen – Is the customer a senior citizen or not
* Partner – Does the customer have a partner that they share the service with
* Dependents – Does the customer have dependents
* Tenure – What is the tenure of the customer with our company in months
* Phone Service – Does the customer also have a phone service
* Multiple Lines – Does the customer have multiple lines
* Internet Service – Does the customer have internet service
* Online Security – Does the customer have online security
* Online Backup – Does the customer have online backup
* Device Protection – Does the customer have device protection
* Tech Support – Does the customer have tech support
* Streaming TV – Does the customer have streaming TV
* Streaming Movies – Does the customer have streaming movies
* Contract Type – What is the customer contract type (month-to-month / annual)
* Paperless Billing – Does the customer utilize paperless billing
* Payment Method – What is the payment method (check, automatic, credit card, etc)
* Monthly Charges – What are the monthly charges for the customer
* Total Charges – What are the total charges to date for the customer

## Business Question

Our primary question was simple- what attributes contribute to the likelihood of customer churn (leaving the company)?

## Materials & methods

Data from the website above was read into the statistical software program RStudio. Using RStudio, we explored this business question using the data as the guide to provide insight. Various tools were used including descriptive statistics, inferential statistics, visualizations, and probit regression. A kernel support vector machine (KSVM) model was tested but further analysis would be required to gain legitimate insight in using this model as a prediction of churn. The output of these methods is described in detail below.

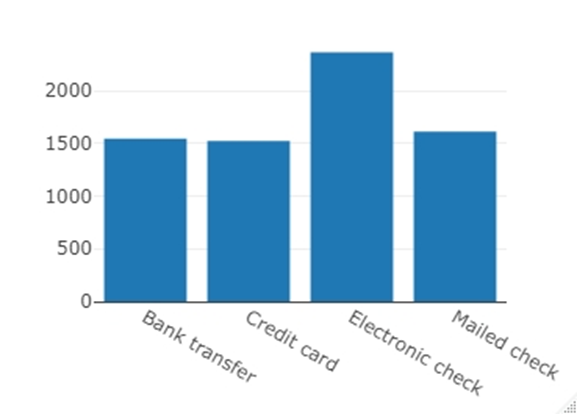
# Segment Analysis Results

The variables captured for the customer churn analysis are split into 4 segments: Customer Demographics, Additional Services, Billing and Contract

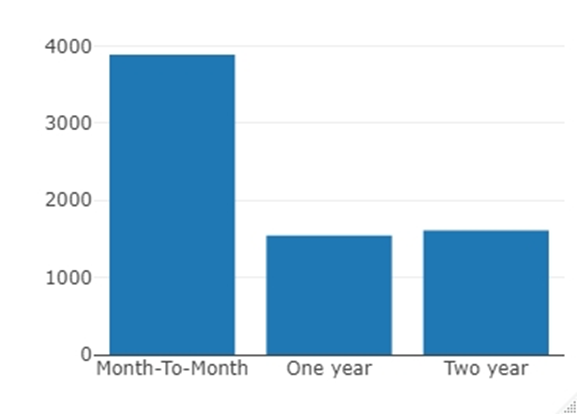
## Segment Analysis 1: Analysis of the Contract, Payment month and Charge

The analysis helps us to understand the behavior of customers on the type of contract, payment month and charge for the service provided by the company. It also helps us to understand the impact that it has on revenue of the company.

*Customer count by payment method*

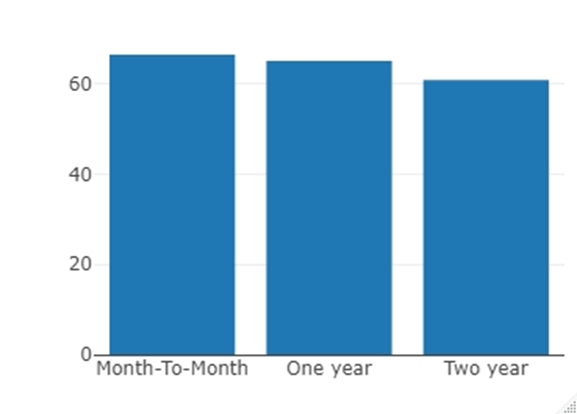


The above chat shows that over 2000 customers prefer paying for the service using electronic check compared to a little over 1,500 using either bank transfer, credit card or mailed check, which makes it the most accepted means of payment with customers. This can be attributed to a lot of factors which need further analysis.

*Customers Count by type of Contract*  


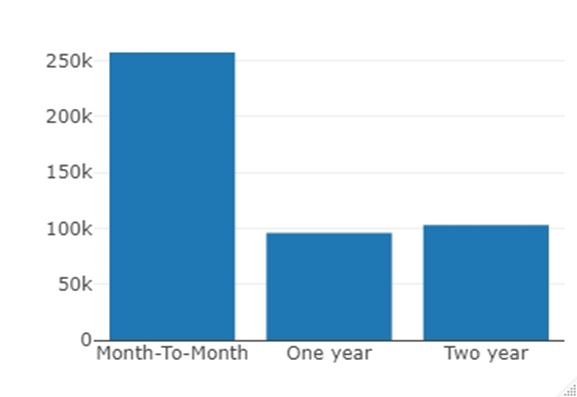
The analysis shows that close to 4000 customers sign up for the month-to-month contract compared to around 1,500 each that opted for the one and two years contract. The analysis clearly indicates that customers prefer the month-to-month contract because of luck of trust, it can also be that the cost of one and two years contract compared to the month-to-month are the same in annual terms.

*Average Charges Per Contract Type*

**

The average charges per contract type show that regardless of the option a customer selected, they all pay the same amount. From customer per contract type analysis, we found out that customers prefer the month-to-month contract and the average charges per contract type analysis gives us a strong indication why customers prefer the month-to-month contract.

*Total Charges Per Contract*

**

The above analysis shows that most of the company’s revenue comes from the month-to-month contract, compared to that of the one and two years contract. This shows that the company is not doing well convincing customers to sign on to the one and two years contract. The implication for the company in the long-term is that there is no guarantee that customers will renew their subscription in the coming month, and this can affect the company form making a revenue projection into the future and how to long-term plan using a vital indicator such revenue streams*.*

## Segment Analysis 2: Additional Services Analysis

The services that the customer had included in their account provided some vital datapoints when it came to determining the customer churn. Of all the additional service datapoints that were analyzed the most meaningful in determine the likelihood of customer churn were the internet service type, whether the customer had online security service or not, and whether the customer had the technical support service or not.

* *Internet Service*
  + A screenshot of a computer screen

    Description automatically generated with low confidence
  + Of the customers that had the Fiber Optic internet service option over 41% of those customers left the company during the period that the data was captured. This above average churn for one specific type of internet service and likely the most expensive is alarming.
  + Chart

    Description automatically generated
  + The fact that the density plots are almost on top of each other in the above chart show that almost 50% of the customers with Fiber Optic internet left the company during the period in which the data was captured. This is highly abnormal and should be further investigated and will be highlighted in the conclusion and recommendations section later in the report.
* *Online Security Service*
  + Graphical user interface, table

    Description automatically generated
  + Of the customers that had the Online Security service option over 41% of those customers left the company during the period that the data was captured. This above average churn for one specific type of additional service is alarming.
  + Chart, line chart

    Description automatically generated
  + The fact that the density plots are almost on top of each other in the above chart show that almost 50% of the customers without the Online Security service left the company during the period in which the data was captured. This is highly abnormal and should be further investigated and will be highlighted in the conclusion and recommendations section later in the report.
* *Technical Support Service*
  + Table

    Description automatically generated
  + Of the customers that had the Tech Support service option over 42% of those customers left the company during the period that the data was captured. This above average churn for one specific type of additional service is alarming.
  + Chart, line chart

    Description automatically generated
  + The fact that the density plots are almost on top of each other in the above chart show that almost 50% of the customers without the Tech Support service left the company during the period in which the data was captured. This is highly abnormal and should be further investigated and will be highlighted in the conclusion and recommendations section later in the report.

## Segment Analysis 3: Customer Demographics

Customer demographics are captured across the following variables:

* Gender (Female or Male)
* Senior Citizen (Yes or No)
* Partner (Yes or No)
* Dependents (Yes or No).

Table 3.1 shows initial analysis of the demographic data set and highlights counts of customers by demographic with corresponding churn volume, churn rate and average tenure.

Yellow highlight = top 3 based on highest churn rate

Green highlight = top 3 based on highest avg tenure

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Demographic** | **Total Population** | **Churn** | **Churn Rate** | **Average Tenure** |
| Male | 3555 | 930 | 0.261603376 | 32.49535865 |
| Female | 3488 | 939 | 0.269208716 | 32.24455275 |
| Senior | 1142 | 476 | 0.416812609 | 33.29597198 |
| nonSenior | 5901 | 1393 | 0.236061684 | 32.19217082 |
| Partner | 3402 | 669 | 0.19664903 | 42.01763668 |
| noPartner | 3641 | 1200 | 0.329579786 | 23.35786872 |
| Dependents | 2110 | 326 | 0.15450237 | 38.36824645 |
| noDependent | 4933 | 1543 | 0.312791405 | 29.80600041 |

***Table 3.1 - customer demographics by population, churn, and average tenure***

A quick scan of the results in table 3.1 show that “noPartner” and “noDependent” have the lowest tenure and 2 of the 3 highest churn rates across all demographic groups. The Senior demographic group raises some questions as they have the highest churn rate while also carrying the third highest tenure. Further investigation is necessary to understand high churn rate and avg tenure within the Senior demographic group.

The relationship between customer churn, size of demographic population and tenure is presented in graph 3.2.

Chart, scatter chart

Description automatically generated

***Graph 3.2 – customer churn plotted against tenure and demographic group population***

Investigation of the displayed results quickly highlights two areas for further investigation:

1. noPartner & noDependents group have churn rates above the average rate for the fullPopulation
2. noPartner & noDependents represent some of the largest demographic segments within the sample population; decreasing churn within these segments can have significant impact on reduction of the overall customer churn rate.
3. Seniors group is an outlier with above average tenure but also the highest churn rate, raising questions of possible sampling error.

## Segment Analysis 4: Further Analysis including Tenure, Monthly and Total Charges

### Descriptive Statistics and Visualizations

*Output values from R:*

|  |  |  |  |
| --- | --- | --- | --- |
| **Descriptive Statistics** | **Tenure** | **Monthly Charges** | **Total Charges** |
| mean | 32.371 | 64.762 | 2281.917 |
| median | 29.000 | 70.350 | 1397.475 |
| min | 0 | 18.25 | 18.8 |
| max | 72 | 118.75 | 8684.8 |
| stdev | 24.55948 | 30.09005 | 2265.27 |
| skewness | 0.2394887 | -0.2204775 | 0.9635838 |

Most notably, in the descriptive statistics we learned that 1) the highest tenure in this company is 72 months (about 6 years), whereas 2) the average is around 32.37 months (about 2.7 years), and 3) the monthly charges are around $64.76 on average. Below are histograms of tenure (left) and monthly charges (right).

Chart, histogram

Description automatically generated Chart, histogram

Description automatically generated

Chart, bar chart

Description automatically generated

Figure 1- Telco Churn count (within the last month). Note that Churn rate = ~26.5%.

Chart, bar chart

Description automatically generated

Figure 2- Churn rate is greater in Fiber optic internet services users (represented in blue) than DSL or No internet service.

Chart, bar chart

Description automatically generated

Figure 3- Churn represented by contract type. Month-to-month has the highest churn rate (represented in orange).

Chart, scatter chart

Description automatically generated

Figure 4- Churn is viewed green. This figure shows greater churn in customers with lower tenure and higher monthly charges, as seen in the top left corner of the figure. Also note the higher churn rate in customers with tenure less than a couple months, regardless of monthly charge.

### Probit Regression

For the Telco Customer data, the output variable (Churn) is a binary dependent variable. Therefore, linear regression is not an appropriate model for this dataset. Here, we ran probit regression, focusing on the middle of our data. When we ran the probit regression using generalized linear modeling, the following output was returned:

Text

Description automatically generated with low confidence

It appears from the probit regression that ‘SeniorCitizen’, ‘Dependents’, ‘tenure’, ‘PhoneService’, ‘Paperlessbilling’, and ‘MonthlyCharges’ are significant variables as related to Churn.

Chart, line chart

Description automatically generated

Tenure was chosen as the significant variable to explore in a regression plot and can be seen in the figure above. We can see that as tenure increases, the churn decreases.

### Kernel Support Vector Machine (KSVM) Learning

We ran a KSVM machine learning approach using a random index of two-thirds of the data as training data and one-third as testing data from a subset of the Telco Customer dataset using the continuous variables (tenure, MonthlyCharges, TotalCharges) as well as the binary independent variables (Partner, Dependents, gender, PhoneService, PaperlessBilling) and Churn as the variable to predict. Below is the output:

Graphical user interface, text, application

Description automatically generated

And the KSVM testing data was plotted:

A picture containing chart

Description automatically generated

This tells a similar story to the scatter plot above. Higher monthly charges and lower tenure show the greatest churn.

# Conclusion and Recommendations

After our data science team analyzed the customer churn data for the company, we conclude that there are practices that our company could put in place to lower customer churn. Based on the datapoints that were captured and the data analysis performed we offer the below recommendations as opportunities to minimize the customer churn from our company:

1. Investigate the Fiber Optic internet service to see if the price is too expensive or if the service is under performing because many customers with this service churned in the period of time when the data was captured.
2. Explore the month-to-month contracts and why they are not wanting to commit to a longer term. Possibly incentivize month-to-month customers to commit to a longer term contract. Many customers churn within the first year, so focus efforts on newer customers.
3. Additionally, ‘NoDependent’ and ‘NoPartner’ demographic groups are more likely to churn and should be targeted with incentives or promotions that reward longer contract terms.
4. Further investigate the customers that have high monthly charges, especially those that are tenure less than a year.
5. The average charge for the three-payment method is almost the same and we recommend the company increase the price for the month-to-month contract to make it unattractive or reduce the charge for the other two contract option as an incentive to move the customers from the month-to-month or any new subscriber. Same approach should use for the payment method.
6. Recommend that we investigate the technical support benefits and if that is something that we should offer to all customers regardless of service type.
7. Recommend that we investigate the online security benefits and if that is something that we should offer to all customers regardless of service type.
8. Recommend looking at the cost benefit analysis of adding online security and technical support to all accounts to alleviate churn based on the absence of these services.

# Project Reflection

One challenge with this dataset is that it contained many binary attributes which we have not worked with much before in R. This also caused a challenge when working with an attribute that contained three variables, which could not be explored in models requiring a numerical input. It may have been more beneficial to use a dataset with more continuous variables to maximize the learning objectives of this class.

# R code

## Segment 1: Analysis of the Contract, Payment month and Charge

install.packages("tidyverse")

library(tidyverse)

install.packages("plyr")

library(plyr)

install.packages("sqldf")

library("sqldf")

install.packages("plotly")

library("plotly")

TelcoC<-read.csv("C:/Users/andre/Desktop/Syracuse Uinversity/Certificate of Advance Study in Data Science/IST 687\_Introduction to Data Science/Final Project/project/TelcoCustomer.csv")

View(TelcoC)

str(TelcoC)

MyData<-as.data.frame(TelcoC)

cols <- c(1, 16:20)

mydata\_analysis<-MyData[,cols]

View(mydata\_analysis)

head(mydata\_analysis)

str(mydata\_analysis)

#No of Customer Group By Paperless Billing

customer\_paperless\_type<-tapply(mydata\_analysis$PaperlessBilling, MyData$PaperlessBilling, count)

view(customer\_paperless\_type)

#Type of Payment Method used by Customers

type\_paymentmethod<-tapply(mydata\_analysis$PaymentMethod, mydata\_analysis$PaymentMethod, FUN = function(x) length(unique(x)))

#Customer Count By Payment Method

sqldf("select PaymentMethod, count(customerID)CustomerIDCount from mydata\_analysis group by PaymentMethod ")

Paymentmothed<-c("Bank transfer","Credit card","Electronic check","Mailed check")

CustomersPerPM<-c(1544,1522,2365,1612)

df<-data.frame(Paymentmothed,CusttomersPerPM)

df

plot\_ly(x=Paymentmothed,y=CusttomersPerPM,type='bar')

#Customers Count by type of Contract

sqldf("select contract, count(customerID)CustomerIDCount from mydata\_analysis group by PaymentMethod")

ContractType<-c("One year","Month-To-Month","Two year")

CustomersPerCT<-c(1544,3887,1612)

df<-data.frame(ContractType,CustomersPerCT)

df

plot\_ly(x=ContractType,y=CustomersPerCT,type='bar')

#Average Charges Per Contract Between One Year, Month-To-Month and Two year

sqldf("select avg(MonthlyCharges), Contract from mydata\_analysis where Contract IN('One year','Month-to-month', 'Two year') group by Contract ")

ContractType<-c("Month-To-Month","One year","Two year")

AvgMonthlyCharges<-c(66.39849 ,65.04861,60.77041)

df<-data.frame(ContractType,AvgMonthlyCharges)

df

plot\_ly(x=ContractType,y=AvgMonthlyCharges,type='bar')

#Total Charges for the month

sqldf("select sum(TotalCharges)TotalCharges from mydata\_analysis")

TotalCharges

1 16056169

#Total Charges Per Contract

sqldf("select sum(MonthlyCharges)TotalChargesPerContract, Contract from mydata\_analysis where Contract IN(select distinct Contract from mydata\_analysis ) group by Contract ")

ContractType<-c("Month-To-Month","One year","Two year")

TotalChargesPerContract<-c(257294.1 ,95816.6,103005.8)

df<-data.frame(ContractType,TotalChargesPerContract)

df

plot\_ly(x=ContractType,y=TotalChargesPerContract,type='bar')

## Segment 2: Additional Services Analysis

# IST687 Group Project -- Telco Churn Dataset [Kaggle]

# https://www.kaggle.com/blastchar/telco-customer-churn

# Jake Conard

# I am looking at the effect on Churn based on the various

# services that a customer has on the Telco plan to include

# Phone

# Multiple Lines

# Internet

# Online Security

# Online Backup

# Device Protection

# Tech Support

# Streaming TV

# Streaming Movies

# Read in dataset from a .csv

testFrame <- read.csv("C:\\Users\\jncon\\OneDrive - Syracuse University\\Desktop\\Syracuse\\IST687\\Project\\project.csv.csv")

# Check the dataframe to ensure structure.

str(testFrame)

# Utilize the tapply function to transform the data to new dataframes

# this will help see the impact of each attribute for churn

# and help with the plotting.

MultLineVchurn <- data.frame(tapply(testFrame$customerID, list(testFrame$MultipleLines,testFrame$Churn), length))

IntServiceVchurn <- data.frame(tapply(testFrame$customerID, list(testFrame$InternetService,testFrame$Churn), length))

OnlineSecVchurn <- data.frame(tapply(testFrame$customerID, list(testFrame$OnlineSecurity,testFrame$Churn), length))

OnlineBackVchurn <- data.frame(tapply(testFrame$customerID, list(testFrame$OnlineBackup,testFrame$Churn), length))

TechSupportVchurn <- data.frame(tapply(testFrame$customerID, list(testFrame$TechSupport,testFrame$Churn), length))

StreamingTV\_Vchurn <- data.frame(tapply(testFrame$customerID, list(testFrame$StreamingTV,testFrame$Churn), length))

StreamingMovieVchurn <- data.frame(tapply(testFrame$customerID, list(testFrame$StreamingMovies,testFrame$Churn), length))

#Install the plotting packages

install.packages("ggplot2")

library(ggplot2)

library(dplyr)

# Create density plots to quickly identify the Churn visually in a

# density plot for each of the attributes.

testFrame %>% ggplot(aes(x=InternetService,fill=Churn))+ geom\_density(alpha=0.8)+scale\_fill\_manual(values=c('pink','purple'))+labs(title='Internet Service split churn vs non churn' )

testFrame %>% ggplot(aes(x=PhoneService,fill=Churn))+ geom\_density(alpha=0.8)+scale\_fill\_manual(values=c('pink','purple'))+labs(title='Phone Service split churn vs non churn' )

testFrame %>% ggplot(aes(x=MultipleLines,fill=Churn))+ geom\_density(alpha=0.8)+scale\_fill\_manual(values=c('pink','purple'))+labs(title='Multiple Lines split churn vs non churn' )

testFrame %>% ggplot(aes(x=OnlineSecurity,fill=Churn))+ geom\_density(alpha=0.8)+scale\_fill\_manual(values=c('pink','purple'))+labs(title='Online Security split churn vs non churn' )

testFrame %>% ggplot(aes(x=DeviceProtection,fill=Churn))+ geom\_density(alpha=0.8)+scale\_fill\_manual(values=c('pink','purple'))+labs(title='Device Protection split churn vs non churn' )

testFrame %>% ggplot(aes(x=TechSupport,fill=Churn))+ geom\_density(alpha=0.8)+scale\_fill\_manual(values=c('pink','purple'))+labs(title='Tech Support split churn vs non churn' )

testFrame %>% ggplot(aes(x=StreamingTV,fill=Churn))+ geom\_density(alpha=0.8)+scale\_fill\_manual(values=c('pink','purple'))+labs(title='Streaming TV split churn vs non churn' )

testFrame %>% ggplot(aes(x=StreamingMovies,fill=Churn))+ geom\_density(alpha=0.8)+scale\_fill\_manual(values=c('pink','purple'))+labs(title='Streaming Movies split churn vs non churn' )

## Segment 3: Customer Demographics R code

## 1 - load telco churn data set

# 1.1 - load libraries

library(ggplot2) # plot graphs

library(reshape2) # melt function

library(dplyr) # filter

library(kernlab) # modeling (ksvm)

# library(sqldf) # to create analytic summary data set

# 1.2 - load data set

setwd("C:\\Users\\grego\\Downloads")

df\_Telco <- read.csv("WA\_Fn-UseC\_-Telco-Customer-Churn.csv")

# 1.3 - explore and prep data set

dim(df\_Telco)

str(df\_Telco)

summary(df\_Telco)

df\_Telco$SeniorCitizen[df\_Telco$SeniorCitizen ==0] <- "No"

df\_Telco$SeniorCitizen[df\_Telco$SeniorCitizen ==1] <- "Yes"

# 1.4 - check for NA

#Check for NA in data set

any(is.na(df\_Telco)) # do any NA's exist? returns T or F

#How many NA?

length(df\_Telco[df\_Telco=='NA'])

#returns list of column names from data frame

colnames(df\_Telco)[colSums(is.na(df\_Telco))>0]

#check observations with NA

df\_na<- df\_Telco[df\_Telco$tenure==0,]

# 1.5 - create demographics subset of data

# Demographic Variables: customerID, gender, SeniorCitizen, Partner, Dependents, tenure, Churn

df\_Demog <- df\_Telco[,-7:-20]

str(df\_Demog)

summary(df\_Demog)

any(is.na(df\_Demog)) # check for NA

## 2 - preliminary analysis and descriptive stats on Customer Demographics dataset

# 2.1 - tapply analysis by

# gender

tapply(df\_Demog$Churn,list(df\_Demog$Churn,df\_Demog$gender),length)

# Senior Citizen

tapply(df\_Demog$Churn,list(df\_Demog$Churn,df\_Demog$SeniorCitizen),length)

# Partner

tapply(df\_Demog$Churn,list(df\_Demog$Churn,df\_Demog$Partner),length)

# Dependent

tapply(df\_Demog$Churn,list(df\_Demog$Churn,df\_Demog$Dependents),length)

# 2.2 - Sample Population Info (previous month)

# create vectors by demographic variable for analysis

v\_churn <- df\_Demog$Churn # vector of sample pop churn

v\_gender <- df\_Demog$gender

v\_senior <- df\_Demog$SeniorCitizen

v\_partner <- df\_Demog$Partner

v\_dependent <- df\_Demog$Dependents

# Sample Population Churn Rate

full\_Population <- dim(df\_Demog)[1]

pop\_churned <- sum(v\_churn == "Yes")

stayed <- sum(v\_churn == "No")

sample\_Pop\_Churn\_rate <- pop\_churned/length(v\_churn) #.26537

# Avg sample pop tenure

sample\_Pop\_Avg\_Tenure <- sum(df\_Demog$tenure)/dim(df\_Demog)[1] #32.37115

# 2.3 - Demographic Churn rates

# 2.3.1 - by Gender

ds <-data.frame(v\_churn,v\_gender)

total\_f <- dim(ds %>% filter(v\_gender == "Female"))[1]

total\_m <- dim(ds %>% filter(v\_gender == "Male"))[1]

churn\_f <- dim(ds %>% filter(v\_churn == "Yes" & v\_gender == "Female"))[1]

churn\_m <- dim(ds %>% filter(v\_churn == "Yes" & v\_gender == "Male"))[1]

churn\_rate\_f <- churn\_f/total\_f

churn\_rate\_m <- churn\_m/total\_m

sub\_df <- df\_Demog %>% filter(gender=="Female")

avg\_tenure\_f <- sum(sub\_df$tenure)/total\_f

sub\_df <- df\_Demog %>% filter(gender=="Male")

avg\_tenure\_m <- sum(sub\_df$tenure)/total\_m

# 2.3.2 - by Senior Citizen

ds <-data.frame(v\_churn,v\_senior)

total\_s <- dim(ds %>% filter(v\_senior == "Yes"))[1]

total\_ns <- dim(ds %>% filter(v\_senior == "No"))[1]

churn\_s <- dim(ds %>% filter(v\_churn == "Yes" & v\_senior == "Yes"))[1]

churn\_ns <- dim(ds %>% filter(v\_churn == "Yes" & v\_senior == "No"))[1]

churn\_rate\_s <- churn\_s/total\_s

churn\_rate\_ns <- churn\_ns/total\_ns

sub\_df <- df\_Demog %>% filter(SeniorCitizen=="Yes")

avg\_tenure\_s <- sum(sub\_df$tenure)/total\_s

sub\_df <- df\_Demog %>% filter(SeniorCitizen=="No")

avg\_tenure\_ns <- sum(sub\_df$tenure)/total\_ns

# 2.3.3 - by Partner

ds <-data.frame(v\_churn,v\_partner)

total\_p <- dim(ds %>% filter(v\_partner == "Yes"))[1]

total\_np <- dim(ds %>% filter(v\_partner == "No"))[1]

churn\_p <- dim(ds %>% filter(v\_churn == "Yes" & v\_partner == "Yes"))[1]

churn\_np <- dim(ds %>% filter(v\_churn == "Yes" & v\_partner == "No"))[1]

churn\_rate\_p <- churn\_p/total\_p

churn\_rate\_np <- churn\_np/total\_np

sub\_df <- df\_Demog %>% filter(Partner=="Yes")

avg\_tenure\_p <- sum(sub\_df$tenure)/total\_p

sub\_df <- df\_Demog %>% filter(Partner=="No")

avg\_tenure\_np <- sum(sub\_df$tenure)/total\_np

# 2.3.4 - by Dependent

ds <-data.frame(v\_churn,v\_dependent)

total\_d <- dim(ds %>% filter(v\_dependent == "Yes"))[1]

total\_nd <- dim(ds %>% filter(v\_dependent == "No"))[1]

churn\_d <- dim(ds %>% filter(v\_churn == "Yes" & v\_dependent == "Yes"))[1]

churn\_nd <- dim(ds %>% filter(v\_churn == "Yes" & v\_dependent == "No"))[1]

churn\_rate\_d <- churn\_d/total\_d

churn\_rate\_nd <- churn\_nd/total\_nd

sub\_df <- df\_Demog %>% filter(Dependents=="Yes")

avg\_tenure\_d <- sum(sub\_df$tenure)/total\_d

sub\_df <- df\_Demog %>% filter(Dependents=="No")

avg\_tenure\_nd <- sum(sub\_df$tenure)/total\_nd

## 3 - create analytic data set and descriptive stats by demographic

# 3.1 - tabulate output for variables by demographic type

df\_output <- data.frame(c("2-Male","2-Female","3-Senior","3-nonSenior","4-Partner","4-noPartner","5-Dependents","5-noDependents","1-fullPopulation")

,c(total\_m,total\_f,total\_s,total\_ns,total\_p,total\_np,total\_d,total\_nd,full\_Population)

,c(churn\_m,churn\_f,churn\_s,churn\_ns,churn\_p,churn\_np,churn\_d,churn\_nd,pop\_churned)

,c(churn\_rate\_m,churn\_rate\_f,churn\_rate\_s,churn\_rate\_ns,churn\_rate\_p,churn\_rate\_np,churn\_rate\_d,churn\_rate\_nd,sample\_Pop\_Churn\_rate)

,c(avg\_tenure\_m,avg\_tenure\_f,avg\_tenure\_s,avg\_tenure\_ns,avg\_tenure\_p,avg\_tenure\_np,avg\_tenure\_d,avg\_tenure\_nd,sample\_Pop\_Avg\_Tenure)

)

colnames(df\_output)<-c("Demographic","Total\_Pop","Churn\_Pop","Churn\_Rate","Avg\_Tenure")

# 3.2 Write dt\_output to file

write.csv(df\_output,"C:\\Users\\grego\\Downloads\\df\_output.csv", row.names = FALSE)

## 4 - Plot churn by demographic group and tenure

d\_plot<- ggplot(df\_output) + geom\_point(aes(x=Demographic, y=Churn\_Rate, size=Total\_Pop, color=Avg\_Tenure))

d\_plot <- d\_plot + ggtitle("Customer Churn by Tenure") + theme(plot.title = element\_text(hjust = 0.5)) + xlab("Demographic") + ylab("Churn Rate")

d\_plot <- d\_plot + theme(axis.text.x = element\_text(angle = 90))

d\_plot

## Segment 4: Tenure, Charges and Further Analysis Code

#read in the data, source: https://www.kaggle.com/blastchar/telco-customer-churn

TelcoChurn <- read.csv("/Users/natalinewman/Desktop/WA\_Fn-UseC\_-Telco-Customer-Churn.csv")

View(TelcoChurn)

nrow(TelcoChurn)

ncol(TelcoChurn)

str(TelcoChurn)

dim(TelcoChurn)

summary(TelcoChurn)

colnames(TelcoChurn)

#Clean the data, check for NAs--------------------------------------------------

any(is.na(TelcoChurn))

length(TelcoChurn[TelcoChurn=='NA'])

colnames(TelcoChurn)[colSums(is.na(TelcoChurn))>0]

median(TelcoChurn$TotalCharges, na.rm=TRUE)

mean(TelcoChurn$TotalCharges, na.rm=TRUE)

sd(TelcoChurn$TotalCharges, na.rm=TRUE)

hist(TelcoChurn$TotalCharges, na.rm=TRUE)

#replace the NAs in TotalCharges with the median of that column because the data is heavily right skewed

TelcoChurn$TotalCharges[is.na(TelcoChurn$TotalCharges)] <- median(TelcoChurn$TotalCharges, na.rm=TRUE)

head(TotalCharges, 50)

# explore the data ------------------------------------------------------------

head(TelcoChurn [order(TelcoChurn$MonthlyCharges) ,])

summary(TelcoChurn$MonthlyCharges)

head (TelcoChurn [order(-TelcoChurn$MonthlyCharges <=20) ,] , )

#T <- tapply(tenure, (TelcoChurn [which(TelcoChurn$MonthlyCharges <= 25) , ]), length) #nope

#T

##

#function for descriptive statistics--------------------------------------------

DescStat <- function (x){

#statistical measurements

a <- mean(x)

b <- median(x)

c <- min(x)

d <- max(x)

e <- sd(x)

f <- quantile (x, probs = c(0.05, 0.95))

g <- skewness(x, na.rm = FALSE)

#Print the results

cat("mean:",a,"\nmedian:",b, "\nmin:", c, "\nmax:", d, "\nstdev:", e, "\nquantile ( 0.05, 0.95):", f, "\nskewness:", g)

}

library(moments)

#end function for descriptive statistics-------------------------------------------------

library(moments)

DescStat(tenure)

hist(tenure)

DescStat(MonthlyCharges)

hist(MonthlyCharges)

DescStat(TotalCharges)

hist(TotalCharges)

#put data frame variables into their own variables for tapply use-----------------------

customerID <- TelcoChurn$customerID

Partner <- TelcoChurn$Partner

Churn <- TelcoChurn$Churn

PhoneService <- TelcoChurn$PhoneService

InternetService <- TelcoChurn$InternetService

Contract <- TelcoChurn$Contract

PaymentMethod <- TelcoChurn$PaymentMethod

MonthlyCharges <- TelcoChurn$MonthlyCharges

TotalCharges <- TelcoChurn$TotalCharges

gender <- TelcoChurn$gender

tenure <- TelcoChurn$tenure

C <- tapply(customerID, Churn, length)

C

barplot(C, col=c("green4","red"), main = "Telco Customer Churn", ylab = "count")

#FALSE(No) TRUE(Yes)

#5174 1869

#t.test(1869, 5174)

#t.test(1869, 5174, alternative = c("greater"), mu = 0, paired = FALSE, conf.level = 0.95,)

#calculate the percent of total

PercOfTotal <- function (myVector, myString)

{

a<-length (myVector [myVector == myString])

b<-a/length(myVector)

return(b)

}

PercOfTotal(Churn, "Yes")

#[1] 0.2653699

P <- tapply(customerID, list(Partner, Churn =='Yes'), length)

P

barplot(P, col=c("orange","blue"), xlab = "Orange = No Partner, Blue = Yes Partner", main="Churn rate by Partner")

Ph<-tapply(customerID, list(PhoneService,Churn=='Yes'), length)

barplot(Ph, col=c("orange","blue"), xlab = "Orange = No Phone Service, Blue = Yes Phone Service", main="Churn rate by Phone Service")

Int <- tapply(customerID, list(InternetService,Churn=='Yes'), length)

Int

barplot(Int, col=c("orange","blue", "green"), xlab="Churn", main="Churn rate by Internet Service Type", legend=TRUE)

tapply(customerID, list(Contract,Churn=='Yes'), length)

tapply(customerID, list(PaymentMethod,Churn=='Yes'), length)

tapply(customerID, list(gender,Churn=='Yes'), length)

temp <- tapply(gender, list(gender,Churn=='Yes'), length)

temp <- tapply(tenure, list(tenure,Churn=='Yes'), length)

temp <- tapply(MonthlyCharges, list(MonthlyCharges,Churn=='Yes'), length)

barplot(temp)

#function to convert columns to numeric more easily, but it doesn't work

#Txt2Num <- function(df, x, txt, newcol) {

# df$newcol <- ifelse(df$x==txt,1,0)

# df$newcol <- as.numeric(df$newcol)

#}

#Txt2Num(TelcoChurn, SeniorCitizen, Yes, seniorcitizen)

############################# Visualizations ################################

#take a look at the data via a histogram

library(ggplot2)

ghist <- ggplot(TelcoChurn, aes(x=tenure)) + geom\_histogram(bins=10, color="black", fill="steelblue")

ghist <- ghist + ditch\_the\_axes + ggtitle("Telco Customers Tenure (in months)")

ghist

#monthly charges histogram

ghist <- ggplot(TelcoChurn, aes(x=MonthlyCharges)) + geom\_histogram(bins=10, color="black", fill="steelblue")

ghist <- ghist + ditch\_the\_axes + ggtitle("Telco Customers Monthly Charges")

ghist

#total charges histogram

ghist <- ggplot(TelcoChurn, aes(x=TotalCharges)) + geom\_histogram(bins=10, color="black", fill="steelblue")

ghist <- ghist + ditch\_the\_axes + ggtitle("Telco Customers Total Charges")

ghist

#Look at the data via a scatter chart

library(ggplot2)

gscatter <- ggplot(TelcoChurn) + geom\_point(aes(x=tenure, y=MonthlyCharges, color=Churn)) + ggtitle("Customer Churn by Tenure and Monthly Charges")

gscatter <- gscatter + ditch\_the\_axes

gscatter

###################convert the 'Contract' column text to 0, 1, and 2: ###################

str(TelcoChurn)

#TelcoChurn$contract <- as.factor(TelcoChurn$Contract) #this doesn't work

#TelcoChurn$contract <- as.numeric(TelcoChurn$Contract) #this doesn't work

#TelcoChurn$contract <- strtoi(TelcoChurn$Contract, base = 0L) #this doesn't work

#TelcoChurn$contract <- recode(TelcoChurn$Contract, 'Month-to-month'=0, 'One year'=1, 'Two year'=2) #this doesn't work

#TelcoChurn$contract <- switch(TelcoChurn$Contract, "Month-to-month"=0, "One year"=1, "Two year"=2) #this also doesn't work…

#TelcoChurn$contract <- ifelse(TelcoChurn$Contract=="One year" | TelcoChurn$Contract=="Two year", 1, 0) #also doesn't work

#TelcoChurn$contract <- ifelse(TelcoChurn$Contract=="Two year" && TelcoChurn$Contract!="One year", 2, 0) #doesn't work

#trying again to get the Contract column coded into 0, 1, 2 still…

#contract\_month <- tapply(Contract, Contract=="Month-to-month", length)

#contract\_oneyr <- tapply(Contract, Contract=="One year", length)

#contract\_twoyr <- tapply(Contract, Contract=="Two year", length) #nope

############## end try to convert three categories to numerical ####################

#################convert the yes/no or male/female columns to 0s and 1s

View(TelcoChurn)

TelcoChurn$Churn <- ifelse(TelcoChurn$Churn=="Yes",1,0)

TelcoChurn$Churn <- as.numeric(TelcoChurn$Churn)

TelcoChurn$Partner <- ifelse(TelcoChurn$Partner=="Yes",1,0)

TelcoChurn$Partner <- as.numeric(TelcoChurn$Partner)

TelcoChurn$Dependents <- ifelse(TelcoChurn$Dependents=="Yes",1,0)

TelcoChurn$Dependents <- as.numeric(TelcoChurn$Dependents)

TelcoChurn$gender <- ifelse(TelcoChurn$gender=="Female",1, 0)

TelcoChurn$gender <- as.numeric(TelcoChurn$gender)

TelcoChurn$PhoneService <- ifelse(TelcoChurn$PhoneService=="Yes",1,0)

TelcoChurn$PhoneService <- as.numeric(TelcoChurn$PhoneService)

TelcoChurn$PaperlessBilling <- ifelse(TelcoChurn$PaperlessBilling=="Yes",1,0)

TelcoChurn$PaperlessBilling <- as.numeric(TelcoChurn$PaperlessBilling)

#binary list = c(Partner, Dependents, gender, PhoneService, PaperlessBilling)

View(TelcoChurn)

#remove the CustomerID column

TelcoChurn <- TelcoChurn[ , -1]

#remove the non-binary character columns

TelcoChurn <- TelcoChurn[ , -7:-15]

TelcoChurn <- TelcoChurn[ , -8]

str(TelcoChurn)

TelcochurnNum <- TelcoChurn

View(TelcochurnNum)

########################## LINEAR REGRESSION ##################################

#Telco.lm <- lm(formula = Churn ~ ., data=TelcoChurn)

#summary(Telco.lm)

#now view only significant variables

#Telco.lm.sig <- lm(formula = Churn ~ SeniorCitizen + tenure + MultipleLines + Contract + PaperlessBilling + PaymentMethod + TotalCharges, data=TelcoChurn)

#summary(Telco.lm.sig)

#but, the data is non-linear, so this is not appropriate

##################### PROBABILITY (PROBIT) REGRESSION ##########################

#logit/probit regression of binary output

Telco.probit <- glm(Churn ~ tenure, family=binomial(probit), data=TelcochurnNum)

summary(Telco.probit)

#define new data frame that contains predictor variable

testdata <- data.frame(tenure=seq(min(TelcochurnNum$tenure), max(TelcochurnNum$tenure)), len=100)

#use fitted model to predict values of vs

testdata$Churn = predict(Telco.probit, testdata, type="response")

#plot logistic regression curve

plot(Churn ~ tenure, data=TelcochurnNum, col="steelblue", main="Probit regression")

lines(Churn ~ tenure, testdata, lwd=2)

# plotting code taken from: https://www.statology.org/plot-logistic-regression-in-r/

###################### end probit regression ##################################

#list churn/ no churn by Contract type

Churn\_contract <- tapply(Churn, list(Contract, Churn=="Yes"), length)

barplot(Churn\_contract, col=c("orange","blue", "green"), xlab = "Churn", main="Churn by Contract", ylab="count", legend=TRUE)

barplot(Ph, col=c("orange","blue"), xlab = "Orange = No Phone Service, Blue = Yes Phone Service", main="Churn rate by Phone Service", legend=TRUE)

library(ggplot2)

gbar <- ggplot(TelcoChurn, aes(x=Contract)) + geom\_bar()

gbar <- gbar + ggtitle("Telco Customers by Contract") + theme(plot.title = element\_text(hjust = 0.5))

gbar

############################ MACHINE LEARNING ##############################

library(ggplot2)

library(arules)

library(arulesViz)

library(e1071)

library(caret)

library(kernlab)

library(gridExtra)

#----------first, split your data set into a training set and a testing data set:

randIndex <- sample(1:dim(TelcochurnNum)[1]) #create a list/vector variable, the [1] is essentially the row names(row numbers)

summary(randIndex) #verify indices in randIndex

length(randIndex) #verify indices in randIndex

#floor takes a single numeric argument x and returns a numeric vector containing the largest integers not greater than the corresponding elements of x.

cutpoint2\_3 <- floor(2 \* dim(TelcochurnNum)[1]/3) #create 2/3 cutpoint

TCtraindata <- TelcoChurn[randIndex[1:cutpoint2\_3] , ]

TCtestdata <- TelcoChurn[randIndex[(cutpoint2\_3+1):dim(TelcochurnNum)[1]] , ]

str(TCtraindata)

head(TCtraindata)

str(TCtestdata)

head(TCtestdata)

#------------------------------------now convert Churn from string into factors:

#training data

#TCtraindata$Churn <- as.factor(TCtraindata$Churn) #it says factors will be ignored, so nvm...

#testing data

#TCtestdata$Churn <- as.factor(TCtestdata$Churn) #it says factors will be ignored, so nvm...

str(TCtestdata)

summary(TCtestdata)

#-----------------------------------------next, build a linear prediction model:

#remove customerID from the data frames

#TCtraindata <- TCtraindata[ , -1]

#TCtestdata <- TCtestdata[ , -1]

model <- lm(Churn ~ ., data=TCtraindata)

summary(model)

lmPred <- predict(model, TCtestdata)

str(lmPred) #this may take a while

compTable <- data.frame(TCtestdata[ , 1], lmPred) #now put the actuals and predicted churn values together in a data frame to see how well your machine learning did

colnames(compTable) <- c("test", "Predict")

head(compTable)

sqrt(mean((compTable$test-compTable$Predict)^2)) #now see the difference between your training data (actuals) compared to the predicted

#lm plot

compTable$error <- abs(compTable$test - compTable$Predict)

Plot3 <- data.frame(compTable$error, TCtestdata$tenure, TCtestdata$MonthlyCharges)

colnames(Plot3) <- c("error","tenure", "MonthlyCharges")

lm.plot <- ggplot(Plot3, aes(x=tenure, y=MonthlyCharges)) + geom\_point(aes(size=error, color=error)) + ggtitle("lm")

lm.plot

#----------------------------------------------------------------

#First, the data has to be pre-processed to be coded into 0, 1, 2, etc.

Churn <- TelcochurnNum$Churn

svmOutput <-ksvm(Churn ~ ., data=TCtraindata, kernel="rbfdot",

kpar="automatic",C=10, cross=10, prob.model=TRUE)

svmOutput #check the output

svmPred <- predict(svmOutput, TCtestdata, type = "response")

head(svmPred)

#put the original test data and ksvm prediction output in a table together

compTable <- data.frame(TCtestdata[,1], svmPred[,1])

# change the column names to "test" and "Pred"

colnames(compTable) <- c("test","Pred")

# compute the Root Mean Squared Error

sqrt(mean((compTable$test-compTable$Pred)^2))

# compute absolute error for each case

compTable$error <- abs(compTable$test - compTable$Pred)

# create a new dataframe contains error, tempreture and wind

svmPlot <- data.frame(compTable$error, TCtestdata$tenure, TCtestdata$MonthlyCharges, TCtestdata$Churn)

# assign column names

colnames(svmPlot) <- c("error","Tenure","MonthlyCharges", "Churn")

head(svmPlot)

# plot result using ggplot, setting "Tenure" as x-axis and "MonthlyCharges" as y-axis

plot.ksvm <- ggplot(svmPlot, aes(x=Tenure,y=MonthlyCharges)) + geom\_point(aes(size=error, color=Churn))

plot.ksvm <- plot.ksvm + ggtitle("Telco Prediction using KSVM") + theme(plot.title=element\_text(hjust=.5))

plot.ksvm

#########################################################################

library(gdata)

library(ggplot2)

library(openintro)

library(ggmap)

library(readxl)

#Function To remove axis formats from the heatmaps (the background grid)

ditch\_the\_axes <- theme(

axis.line = element\_blank(),

axis.ticks = element\_blank(),

panel.border = element\_blank(),

panel.grid = element\_blank(),

)