**D209 Data Mining 1**

Nicole Reiswig

College of Information Technology, Western Governors University

Festus Elleh

January 14, 2024

**Table of Contents**

**Part I: Research Question**

A.

1.  Research question 3

2.  Data analysis goal 3

**Part II: Method Justification**

B.

1.  Classification method expected outcomes 3

2.  Classification method assumption 3

3.  Packages and libraries 4

**Part III: Data Preparation**

C.

1.  Goal 4

2.  Variables 4

3.  Data Preparation 7

4.  Cleaned data set 8

**Part IV: Analysis**

D.

1.  Split data, training, and test set 8

2.  Analysis technique 11

3.  Code 12

**Part V: Data Summary and Implications**

E.

1.  Mean Squared Error 51

2.  Results and implications 51

3.  Limitation 51

4.  Recommendation 52

**Part VI: Demonstration**

F.  Panopto video 52

G.  Sources 52

# Predictive Analysis

**A.  Part I: Research Question**

**1.** The research question that is to be answered in this data mining report is, “Is it possible to predict Tenure and identify key contributors using the Lasso Regression method and current available data?”

**2.** The goal of this data mining report and classification analysis is to identify major predictors of length of Tenure, determine which factors it is based on, and determine if Tenure can be predicated based on the trained model. This project aims to forecast factors affecting Tenure so that appropriate actions can be taken to lengthen Tenure. In this project, we will clean, evaluate, and create a supervised machine-learning model.

**B. Part II: Method Justification**

**1.** Lasso Regression is an analysis method used to analyze the research question, “Is it possible to predict Tenure and identify key contributors using the Lasso Regression method and current available data?” Lasso Regression was chosen as the best method for this analysis because it performs both variable selection and regularization to improve prediction accuracy for the statistical model. Lasso stands for least absolute shrinkage and selection operator. It is a good model to use for this analysis because it will automatically remove features that have little to no impact on the model. Lasso regression is good to use on large data sets with continuous features.

**2.** Assumptions of Lasso Regression are that it is assumed that linear model coefficients are sparse or non-zero (Wikipedia, 2024). Lasso Regression has the same assumptions as linear regression being, linearity, independence, homoscedasticity, and normality.

**3.** The R Script in R Studios is the programming language utilized to perform this classification analysis. The libraries and packages used are tidyverse which is essential to data science collection, ggplot2 which is used to generate plots, dplyr which is essential for data frame manipulation, caret which is a predictive model package, e1071 which is required for confusion matrix, corrr to check correlation, ROCR which is required to create a performance curve, rpart which is used for building classification and regression trees, data.table used for reading and manipulation of data, glmnet for regression.

**C. Part III: Data Preparation**

**1.** A preprocessing goal of this analysis is to encode the many categorical variables from yes/no to 0/1 utilizing dummy variables to fit the Lasso Regression model for analysis.

**2.**  The dataset variables in the telecom churn database utilized to answer the research question in section A1 utilizing Lasso Regression are as follows, see data class for classification of categorical or numerical variable type:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Variable** | **Data Type** | **Description** | **Example** | **Data Class** |
| Case order | Qualitative | Placeholder variable | 1 | Categorical |
| Customer ID | Qualitative | Unique customer id | K409198 | Categorical |
| Interaction uid | Qualitative | Transaction id | Aa90260b-4141… | Categorical |
| City | Qualitative | Customer residence | Point Baker | Categorical |
| State | Qualitative | Customer residence | Arkansas | Categorical |
| County | Qualitative | Customer residence | Cook | Categorical |
| Zip | Qualitative | Customer residence | 99362 | Categorical |
| Lat | Qualitative | GPS coordinate | 56.251 | Categorical |
| Lng | Qualitative | GPS coordinate | -133.376 | Categorical |
| Population | Quantitative | Population within 1mile of customer | 38 | Numeric |
| Area | Qualitative | Area type | Rural | Categorical |
| Timezone | Qualitative | Customer timezone | America/Chicago | Categorical |
| Job | Qualitative | Customer job | Engineer | Categorical |
| Children | Quantitative | # of children in household | 2 | Numeric |
| Age | Quantitative | Age of customer | 30 | Numeric |
| Education | Qualitative | Customer education | Masters | Categorical |
| Employment | Qualitative | Customer employment | Employed | Categorical |
| Income | Quantitative | Income of customer | 28561 | Numeric |
| Marital | Qualitative | Customer marital status | Single | Categorical |
| Gender | Qualitative | Gender of customer | Female | Categorical |
| Churn | Qualitative | If customer d/c service | Yes | Categorical |
| Outage sec per week | Quantitative | Avg # of outage seconds a week | 6.9 | Numeric |
| Email | Quantitative | # of emails sent to customer in week | 5 | Numeric |
| Contacts | Quantitative | # of times customer contacted tech support | 0 | Numeric |
| Yearly equip failure | Quantitative | # of customer equipment failures a year | 0 | Numeric |
| Techie | Qualitative | Customer is technically inclined | Yes | Categorical |
| Contract | Qualitative | Contract terms | One year | Categorical |
| Port modem | Qualitative | Customer has port modem | Yes | Categorical |
| Tablet | Qualitative | Customer has tablet | Yes | Categorical |
| Internet service | Qualitative | Customers internet service provider | Fiber optics | Categorical |
| Phone | Qualitative | Customer has phone service | Yes | Categorical |
| Multiple | Qualitative | Customer has multiple lines | Yes | Categorical |
| Online security | Qualitative | Customer has add on security | Yes | Categorical |
| Online backup | Qualitative | Customer has add on backup | Yes | Categorical |
| Device protection | Qualitative | Customer has device protection | yes | Categorical |
| Tech support | Qualitative | Customer has add on tech support | Yes | Categorical |
| Streaming tv | Qualitative | Customer has streaming tv | Yes | Categorical |
| Streaming movies | Qualitative | Customer has streaming movies | Yes | Categorical |
| Paperless billing | Qualitative | Customer has paperless billing | Yes | Categorical |
| Payment method | Qualitative | Customers payment method | Check | Categorical |
| Tenure | Quantitative | # of months customer has been with provider | 12 | Numeric |
| Monthly charge | Quantitative | Amount of monthly charge | 171.45 | Numeric |
| Bandwidth gb year | Quantitative | Amount of gb used per year | 904 | Numeric |
| Timely response | Quantitative | Survey response rating importance of timely response | 1 | Categorical |
| Timely fixes | Quantitative | Survey response rating importance of timely fixes | 2 | Categorical |
| Timely replacements | Quantitative | Survey response rating importance of timely replacements | 3 | Categorical |
| Reliability | Quantitative | Survey response rating importance of reliability | 4 | Categorical |
| Options | Quantitative | Survey response rating importance of having options | 5 | Categorical |
| Respectful response | Quantitative | Survey response rating importance of respectful response | 6 | Categorical |
| Courteous exchange | Quantitative | Survey response rating importance of courteous exchange | 7 | Categorical |
| Evidence of active listening | Quantitative | Survey response rating importance of active listening | 8 | Categorical |

**3.** Steps to prepare the dataset for the analysis:

* 1. Import CSV data file into R studio
  2. Profile data utilizing str()
  3. Check dimensions of data utilizing dim()
  4. Load packages needed for each step of the analysis
  5. View data with str() and glimpse()
  6. Check for duplicates using duplicated()
  7. Check for missing values using colSums(is.na()) or vismis()
  8. Rename columns with misspellings or awkward names
  9. Explore data using univariate and bivariate visuals such as
     1. Histograms
     2. Boxplots
     3. Bar plots
     4. Scatterplots
  10. When appropriate, impute, retain, or exclude values
  11. Check the correlation of variables
  12. Encode yes/no variables with dummy variables as 0/1
  13. Export the clean data to CSV. See the attached CSV of clean data.

**4.** See the cleaned data set attached.

**D. Part IV: Analysis**

**1.** Splitting the data

####DF

DF <- Mutate\_Churn2[c('Children',

                      'Age',

                      'Income',

                      'Outage\_sec\_perweek',

                      'Email',

                      'Contacts',

                      'Yearly\_equip\_failure',

                      'Techie',

                      'Port\_modem',

                      'Tablet',

                      'Phone',

                      'Multiple',

                      'OnlineSecurity',

                      'OnlineBackup',

                      'DeviceProtection',

                      'TechSupport',

                      'StreamingTV',

                      'StreamingMovies',

                      'PaperlessBilling',

                      'Churn',

                      'MonthlyCharge',

                      'Bandwidth\_GB\_Year',

                      'Timely\_Response',

                      'Timely\_Fixes',

                      'Timely\_Replacements',

                      'Reliability',

                      'Options',

                      'Respectful\_Response',

                      'Courteous\_Exchange',

                      'Active\_Listening',

                      'Area\_Suburban',

                      'Area\_Urban',

                      'Contract\_OneYear',

                      'Contract\_TwoYear',

                      'Gender\_Male',

                      'Gender\_Nonbinary',

                      'InternetService\_FiberOptic',

                      'InternetService\_None',

                      'Tenure')]

#Partition data, set seed

set.seed(1234)

#Proportion for reproducibility- Train

Train\_prop <- 0.7

#Partition data into training and test sets

Train\_indices <- createDataPartition(DF$Tenure, p = Train\_prop, list = FALSE)

Train\_data <- DF[Train\_indices, ]

Test\_data <- DF[-Train\_indices, ]

#CSV files

write.csv(Train\_data, file = "Training\_data.2.csv", row.names = FALSE)

write.csv(Test\_data, file = "Test\_data.2.csv", row.names = FALSE)

#Load test

Train\_data <- read.csv("Training\_data.2.csv")

Test\_data <- read.csv("Test\_data.2.csv")

#Extract variables

X\_Train <- Train\_data[, c('Children',

                          'Age',

                          'Income',

                          'Outage\_sec\_perweek',

                          'Email',

                          'Contacts',

                          'Yearly\_equip\_failure',

                          'Techie',

                          'Port\_modem',

                          'Tablet',

                          'Phone',

                          'Multiple',

                          'OnlineSecurity',

                          'OnlineBackup',

                          'DeviceProtection',

                          'TechSupport',

                          'StreamingTV',

                          'StreamingMovies',

                          'PaperlessBilling',

                          'Churn',

                          'MonthlyCharge',

                          'Bandwidth\_GB\_Year',

                          'Timely\_Response',

                          'Timely\_Fixes',

                          'Timely\_Replacements',

                          'Reliability',

                          'Options',

                          'Respectful\_Response',

                          'Courteous\_Exchange',

                          'Active\_Listening',

                          'Area\_Suburban',

                          'Area\_Urban',

                          'Contract\_OneYear',

                          'Contract\_TwoYear',

                          'Gender\_Male',

                          'Gender\_Nonbinary',

                          'InternetService\_FiberOptic',

                          'InternetService\_None')]

X\_Test <- Test\_data[, c('Children',

                        'Age',

                        'Income',

                        'Outage\_sec\_perweek',

                        'Email',

                        'Contacts',

                        'Yearly\_equip\_failure',

                        'Techie',

                        'Port\_modem',

                        'Tablet',

                        'Phone',

                        'Multiple',

                        'OnlineSecurity',

                        'OnlineBackup',

                        'DeviceProtection',

                        'TechSupport',

                        'StreamingTV',

                        'StreamingMovies',

                        'PaperlessBilling',

                        'Churn',

                        'MonthlyCharge',

                        'Bandwidth\_GB\_Year',

                        'Timely\_Response',

                        'Timely\_Fixes',

                        'Timely\_Replacements',

                        'Reliability',

                        'Options',

                        'Respectful\_Response',

                        'Courteous\_Exchange',

                        'Active\_Listening',

                        'Area\_Suburban',

                        'Area\_Urban',

                        'Contract\_OneYear',

                        'Contract\_TwoYear',

                        'Gender\_Male',

                        'Gender\_Nonbinary',

                        'InternetService\_FiberOptic',

                        'InternetService\_None')]

#Create train and test sets

Y\_train <- Train\_data$Tenure

Y\_test <- Test\_data$Tenure

#Export Training and Test Data

write.csv(X\_Train, "C:/Users/ntrei/OneDrive/Documents/MSDA/XTrain209.2.csv")

write.csv(X\_Test, "C:/Users/ntrei/OneDrive/Documents/MSDA/XTest209.2.csv")

write.csv(Y\_train, "C:/Users/ntrei/OneDrive/Documents/MSDA/YTrain209.2.csv")

write.csv(Y\_test, "C:/Users/ntrei/OneDrive/Documents/MSDA/YTest209.2.csv")

See the attached split training and test data sets.

2. Output and intermediate calculations

#Lasso Regression [in text citation: (Poulson, B. 2020)]

#Load packages

pacman::p\_load(lars, magrittr, pacman, rio, tidyverse)

#Import as tibble

DF <- import("C:/Users/ntrei/OneDrive/Documents/MSDA/MutateChurn2.209.2.csv") %>%

  as\_tibble() %>%

  print()

#Get descriptives

DF %>% summary()

#Variables need to be the same scale

DF %<>%

  scale() %>%

  as\_tibble()

#Get descriptives again

DF %>% summary()

#Data as matrix for lars package

y <- DF %>% select(Churn) %>% as.matrix()

X <- DF %>% select(-Churn) %>% as.matrix()

#Compute model

fit\_step <- lars(X, y, type = "stepwise")

#Plot results

fit\_step %>% plot()

legend(

  "bottomleft",

  lwd = 2,

  col = (1:nrow(X)),

  legend = colnames(X),

  cex = .7

)

#Lasso regression

Fit\_lasso <- lars(X, y, type = "lasso")

#Plot results

Fit\_lasso %>% plot()

legend(

  "bottomleft",

  lwd = 2,

  col = (1:nrow(X)),

  legend = colnames(X),

  cex = .4

)

#View more results

Fit\_lasso$R2

Fit\_lasso$R2 %>% plot()

Fit\_lasso %>% coef()

#Fit model and calculate MSE and MAE

# Perform k-fold cross-validation to find the optimal lambda value

library(glmnet)

cv\_model <- cv.glmnet(x, y, alpha = 1, type.measure = "mse")

best\_lambda <- cv\_model$lambda.min # optimal lambda value

# Obtain predictions on the test set using the optimal lambda value

lasso\_pred <- predict(cv\_model, s = best\_lambda, newx = x)

# Calculate MSE on the test set

lasso\_mse <- mean((y - lasso\_pred)^2)

lasso\_mse

plot(lasso\_mse)

plot(lasso\_pred)

plot(best\_lambda)

plot(cv\_model)

No intermediate calculations were performed in this analysis.

**3.** See the attached code.

#Check the working directory

getwd()

#Data profiling

str("~/MSDA/churn\_clean")

#Dimensions of data

dim(churn\_clean)

#10000 50

# load packages

#view missng data

library(visdat)

#for reading and manipulation of data

library(data.table)

# used for data manipulation and joining

library(dplyr)

# used for regression

library(glmnet)

# used for plotting

library(ggplot2)

# used for skewness

library(e1071)

# used for combining multiple plots

library(cowplot)

#collection of r packages for data science

library(tidyverse)

#Perform Naive Bayes Classification

library(naivebayes)

#Tools for training regression and classification models

library(caret)

#For loading and unloading packages

library(pacman)

#For pipes

library(magrittr)

#For lasso regression

library(lars)

#View data

glimpse(churn\_clean)

str(churn\_clean)

#Detect duplicates

duplicated("~/MSDA/churn\_clean")

#No duplicates

#Detect missing values

vis\_miss(churn\_clean)

colSums(is.na(churn\_clean))

#No missing values

#Rename column names item 1- 8 [in-text citation: (Zach, 2022)]

colnames(churn\_clean)[colnames(churn\_clean) == 'Item1'] <- 'Timely\_Response'

colnames(churn\_clean)[colnames(churn\_clean) == 'Item2'] <- 'Timely\_Fixes'

colnames(churn\_clean)[colnames(churn\_clean) == 'Item3'] <- 'Timely\_Replacements'

colnames(churn\_clean)[colnames(churn\_clean) == 'Item4'] <- 'Reliability'

colnames(churn\_clean)[colnames(churn\_clean) == 'Item5'] <- 'Options'

colnames(churn\_clean)[colnames(churn\_clean) == 'Item6'] <- 'Respectful\_Response'

colnames(churn\_clean)[colnames(churn\_clean) == 'Item7'] <- 'Courteous\_Exchange'

colnames(churn\_clean)[colnames(churn\_clean) == 'Item8'] <- 'Active\_Listening'

#Verify columns renamed successfully

glimpse(churn\_clean)

#Visualizations

#Boxplot of each variable - detect outliers - Univariate Graphs of each variable

boxplot(churn\_clean$CaseOrder, main = "Case Order")

boxplot(churn\_clean$Zip, main = "Zip Code")

boxplot(churn\_clean$Lat, main = "Lat")

boxplot(churn\_clean$Lng, main = "Lng")

boxplot(churn\_clean$Population, main = "Population")

boxplot(churn\_clean$Children, main = "Children")

boxplot(churn\_clean$Age, main = "Age")

boxplot(churn\_clean$Income, main = "Income")

boxplot(churn\_clean$Outage\_sec\_perweek, main = "Outage Sec Per Week")

boxplot(churn\_clean$Email, main = "Email")

boxplot(churn\_clean$Contacts, main = "Contacts")

boxplot(churn\_clean$Yearly\_equip\_failure, main = "Yearly Equip Failures")

boxplot(churn\_clean$Tenure, main = "Tenure")

boxplot(churn\_clean$MonthlyCharge, main = "Monthly Charge")

boxplot(churn\_clean$Bandwidth\_GB\_Year, main = "Bandwidth GB Year")

#Count and range of variables with outliers

Children <- churn\_clean[which(churn\_clean$Children > 7), ]

str(Children)

Income <- churn\_clean[which(churn\_clean$Income > 100000), ]

str(Income)

Outage <- churn\_clean[which(churn\_clean$Outage\_sec\_perweek > 20), ]

str(Outage)

Outage2 <- churn\_clean[which(churn\_clean$Outage\_sec\_perweek < 0), ]

str(Outage2)

Email <- churn\_clean[which(churn\_clean$Email > 20), ]

str(Email)

Email2 <- churn\_clean[which(churn\_clean$Email < 4), ]

str(Email2)

Contacts <- churn\_clean[which(churn\_clean$Contacts > 5), ]

str(Contacts)

Equipment <- churn\_clean[which(churn\_clean$Yearly\_equip\_failure > 2), ]

str(Equipment)

Monthly <- churn\_clean[which(churn\_clean$MonthlyCharge > 300), ]

str(Monthly)

Timely\_Response <- churn\_clean[which(churn\_clean$Timely\_Response > 5), ]

str(Timely\_Response)

Timely\_Response1 <- churn\_clean[which(churn\_clean$Timely\_Response < 2), ]

str(Timely\_Response1)

Timely\_Fixes <- churn\_clean[which(churn\_clean$Timely\_Fixes > 5), ]

str(Timely\_Fixes)

Timely\_Fixes2 <- churn\_clean[which(churn\_clean$Timely\_Fixes < 2), ]

str(Timely\_Fixes2)

Timely\_Replacements <- churn\_clean[which(churn\_clean$Timely\_Replacements > 5), ]

str(Timely\_Replacements)

Timely\_Replacements3 <- churn\_clean[which(churn\_clean$Timely\_Replacements < 2), ]

str(Timely\_Replacements3)

Reliability <- churn\_clean[which(churn\_clean$Reliability > 5), ]

str(Reliability)

Reliability4 <- churn\_clean[which(churn\_clean$Reliability < 2), ]

str(Reliability4)

Options <- churn\_clean[which(churn\_clean$Options > 5), ]

str(Options)

Options5 <- churn\_clean[which(churn\_clean$Options < 2), ]

str(Options5)

Respectful\_Response <- churn\_clean[which(churn\_clean$Respectful\_Response > 5), ]

str(Respectful\_Response)

Respectful\_Response6 <- churn\_clean[which(churn\_clean$Respectful\_Response < 2), ]

str(Respectful\_Response6)

Courteous\_Exchange <- churn\_clean[which(churn\_clean$Courteous\_Exchange > 5), ]

str(Courteous\_Exchange)

Courteous\_Exchange7 <- churn\_clean[which(churn\_clean$Courteous\_Exchange < 2), ]

str(Courteous\_Exchange7)

Active\_Listening <- churn\_clean[which(churn\_clean$Active\_Listening > 5), ]

str(Active\_Listening)

Active\_Listening8 <- churn\_clean[which(churn\_clean$Active\_Listening < 2), ]

str(Active\_Listening8)

#Summary statistics

summary(churn\_clean)

#Histograms of variables with outliers- Univariate graphs

hist(churn\_clean$Population, col = 'turquoise', main = "Population")

hist(churn\_clean$Children, col = 'turquoise', main = "Children")

hist(churn\_clean$Income, col = 'turquoise', main = "Income")

hist(churn\_clean$Outage\_sec\_perweek, col = 'turquoise', main = "Outage Sec Perweek")

hist(churn\_clean$Email, col = 'turquoise', main = "Email")

hist(churn\_clean$Contacts, col = 'turquoise', main = "Contacts")

hist(churn\_clean$Yearly\_equip\_failure, col = 'turquoise', main = "Yearly Equip Failure")

hist(churn\_clean$Timely\_Response, col = 'turquoise', main = "Timely Response")

hist(churn\_clean$Timely\_Fixes, col = 'turquoise', main = "Timely Fixes")

hist(churn\_clean$Timely\_Replacements, col = 'turquoise', main = "Timely Replacements")

hist(churn\_clean$Reliability, col = 'turquoise', main = "Reliability")

hist(churn\_clean$Options, col = 'turquoise', main = "Options")

hist(churn\_clean$Respectful\_Response, col = 'turquoise', main = "Respectful Response")

hist(churn\_clean$Courteous\_Exchange, col = 'turquoise', main = "Courteous Exchange")

hist(churn\_clean$Active\_Listening, col = 'turquoise', main = "Active Listening")

#Explore data variables- univariate graphs [in-text citation: (R programming 101, n.d.)]

barplot(sort(table(churn\_clean$Area)), col = 'blue', main = "Area")

barplot(sort(table(churn\_clean$TimeZone)), col = 'blue', main = "Timezone")

barplot(sort(table(churn\_clean$Children)), col = 'blue', main = "Number of Children")

barplot(sort(table(churn\_clean$Age)), col = 'blue', main = "Age")

barplot(sort(table(churn\_clean$Income)), col = 'blue', main = "Income")

barplot(sort(table(churn\_clean$Marital)), col = 'blue', main = "Marital")

barplot(sort(table(churn\_clean$Gender)), col = 'blue', main = "Gender")

barplot(sort(table(churn\_clean$Churn)), col = 'blue', main = "Churn")

barplot(sort(table(churn\_clean$Outage\_sec\_perweek)), col = 'blue', main = "Outage Sec Per Week")

barplot(sort(table(churn\_clean$Email)), col = 'blue', main = "Email")

barplot(sort(table(churn\_clean$Contacts)), col = 'blue', main = "Contacts")

barplot(sort(table(churn\_clean$Yearly\_equip\_failure)), col = 'blue', main = "Yearly Equipment Failure")

barplot(sort(table(churn\_clean$Techie)), col = 'blue', main = "Techie")

barplot(sort(table(churn\_clean$Contract)), col = 'blue', main = "Contracts")

barplot(sort(table(churn\_clean$Port\_modem)), col = 'blue', main = "Port Modem")

barplot(sort(table(churn\_clean$Tablet)), col = 'blue', main = "Tablet")

barplot(sort(table(churn\_clean$InternetService)), col = 'blue', main = "Internet Service")

barplot(sort(table(churn\_clean$Phone)), col = 'blue', main = "Phone")

barplot(sort(table(churn\_clean$Multiple)), col = 'blue', main = "Multiple")

barplot(sort(table(churn\_clean$OnlineSecurity)), col = 'blue', main = "Online Security")

barplot(sort(table(churn\_clean$OnlineBackup)), col = 'blue', main = "Online Backup")

barplot(sort(table(churn\_clean$DeviceProtection)), col = 'blue', main = "Device Protection")

barplot(sort(table(churn\_clean$TechSupport)), col = 'blue', main = "Tech Support")

barplot(sort(table(churn\_clean$StreamingTV)), col = 'blue', main = "Streaming TV")

barplot(sort(table(churn\_clean$StreamingMovies)), col = 'blue', main = "Streaming Movies")

barplot(sort(table(churn\_clean$PaperlessBilling)), col = 'blue', main = "Paperless Billing")

barplot(sort(table(churn\_clean$PaymentMethod)), col = 'blue', main = "Payment Method")

barplot(sort(table(churn\_clean$Tenure)), col = 'blue', main = "Tenure")

barplot(sort(table(churn\_clean$MonthlyCharge)), col = 'blue', main = "Monthly Charge")

barplot(sort(table(churn\_clean$Bandwidth\_GB\_Year)), col = 'blue', main = "Bandwidth GB Year")

barplot(sort(table(churn\_clean$Timely\_Response)), main = "Timely Response")

barplot(sort(table(churn\_clean$Timely\_Fixes)), col = 'blue', main = "Timely Fixes")

barplot(sort(table(churn\_clean$Timely\_Replacements)), col = 'blue', main = "Timely Replacements")

barplot(sort(table(churn\_clean$Reliability)), col = 'blue', main = "Reliability")

barplot(sort(table(churn\_clean$Options)), col = 'blue', main = "Options")

barplot(sort(table(churn\_clean$Respectful\_Response)), col = 'blue', main = "Respecftful Response")

barplot(sort(table(churn\_clean$Courteous\_Exchange)), col = 'blue', main = "Courteous Exchange")

barplot(sort(table(churn\_clean$Active\_Listening)), col = 'blue', main = "Active Listening")

#BiVariate Graph of each variable   [in text citation: (R4DS. (2023).)

library(ggplot2)

#Stacked bar chart Tenure, Area

ggplot(churn\_clean, aes(x = Tenure, y = Area, fill = Area)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = Area, fill = Area)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

  formula = y ~ poly(x,2),

  color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = Area)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Area", x = "Length of Tenure", title = "Length of Tenure by Area",

  subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = Area)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by Area")

#Density Ridge

library(ggridges)

ggplot(churn\_clean, aes(x = Tenure, y = Area, fill = Area)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Area") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, Contract

ggplot(churn\_clean, aes(x = Tenure, y = Contract, fill = Contract)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = Contract, fill = Contract)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = Contract)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Contract", x = "Tenure", title = "Length of Tenure by Contract",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = Contract)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by Contract")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = Contract, fill = Contract)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Contract") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, Gender

ggplot(churn\_clean, aes(x = Tenure, y = Gender, fill = Gender)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = Gender, fill = Gender)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = Gender)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Gender", x = "Length of Tenure", title = "Length of Tenure by Gender",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = Gender)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by Gender")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = Gender, fill = Gender)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Gender") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, Marital

ggplot(churn\_clean, aes(x = Tenure, y = Marital, fill = Marital)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = Marital, fill = Marital)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = Marital)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Marital", x = "Length of Tenure", title = "Length of Tenure by Marital Status",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = Marital)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by Marital Status")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = Marital, fill = Marital)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Marital Status") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, Internet Service

ggplot(churn\_clean, aes(x = Tenure, y = InternetService, fill = InternetService)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = InternetService, fill = InternetService)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = InternetService)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Internet Service", x = "Length of Tenure", title = "Length of Tenure by Internet Service Type",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = InternetService)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by Internet Service Type")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = InternetService, fill = InternetService)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Internet Service Type") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, Payment Method

ggplot(churn\_clean, aes(x = Tenure, y = PaymentMethod, fill = PaymentMethod)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = PaymentMethod, fill = PaymentMethod)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = PaymentMethod)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Payment Method", x = "Length of Tenure", title = "Length of Tenure by Payment Method",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = PaymentMethod)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by Payment Method")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = PaymentMethod, fill = PaymentMethod)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Payment Method") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, Churn

ggplot(churn\_clean, aes(x = Tenure, y = Churn, fill = Churn)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = Churn, fill = Churn)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = Churn)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Churn", x = "Length of Tenure", title = "Length of Tenure by Churn",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = Churn)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by Churn")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = Churn, fill = Churn)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Churn") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, Techie

ggplot(churn\_clean, aes(x = Tenure, y = Techie, fill = Techie)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = Techie, fill = Techie)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = Techie)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Techie", x = "Length of Tenure", title = "Length of Tenure by Techie",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = Techie)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by Techie")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = Techie, fill = Techie)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Techie") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, Port Modem

ggplot(churn\_clean, aes(x = Tenure, y = Port\_modem, fill = Port\_modem)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = Port\_modem, fill = Port\_modem)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = Port\_modem)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Port Modem", x = "Length of Tenure", title = "Length of Tenure by Port Modem",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = Port\_modem)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by Port Modem")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = Port\_modem, fill = Port\_modem)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Port Modem") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, Tablet

ggplot(churn\_clean, aes(x = Tenure, y = Tablet, fill = Tablet)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = Tablet, fill = Tablet)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = Tablet)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Tablet", x = "Length of Tenure", title = "Length of Tenure by Tablet",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = Tablet)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by Tablet")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = Tablet, fill = Tablet)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Tablet") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, Phone

ggplot(churn\_clean, aes(x = Tenure, y = Phone, fill = Phone)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = Phone, fill = Phone)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = Phone)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Phone", x = "Length of Tenure", title = "Length of Tenure by Phone",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = Phone)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by Phone")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = Phone, fill = Phone)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Phone") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, Multiple

ggplot(churn\_clean, aes(x = Tenure, y = Multiple, fill = Multiple)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = Multiple, fill = Multiple)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = Multiple)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Multiple", x = "Length of Tenure", title = "Length of Tenure by Multiple",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = Multiple)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by Multiple")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = Multiple, fill = Multiple)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Multiple") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, OnlineSecurity

ggplot(churn\_clean, aes(x = Tenure, y = OnlineSecurity, fill = OnlineSecurity)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = OnlineSecurity, fill = OnlineSecurity)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = OnlineSecurity)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "OnlineSecurity", x = "Length of Tenure", title = "Length of Tenure by OnlineSecurity",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = OnlineSecurity)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by OnlineSecurity")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = OnlineSecurity, fill = OnlineSecurity)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Online Security") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, OnlineBackup

ggplot(churn\_clean, aes(x = Tenure, y = OnlineBackup, fill = OnlineBackup)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = OnlineBackup, fill = OnlineBackup)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = OnlineBackup)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Online Backup", x = "Length of Tenure", title = "Length of Tenure by Online Backup",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = OnlineBackup)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by Online Backup")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = OnlineBackup, fill = OnlineBackup)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Online Backup") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, Device Protection

ggplot(churn\_clean, aes(x = Tenure, y = DeviceProtection, fill = DeviceProtection)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = DeviceProtection, fill = DeviceProtection)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = DeviceProtection)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Device Protection", x = "Length of Tenure", title = "Length of Tenure by Device Protection",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = DeviceProtection)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by Device Protection")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = DeviceProtection, fill = DeviceProtection)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Device Protection") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, TechSupport

ggplot(churn\_clean, aes(x = Tenure, y = TechSupport, fill = TechSupport)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = TechSupport, fill = TechSupport)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = TechSupport)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Tech Support", x = "Length of Tenure", title = "Length of Tenure by Tech Support",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = TechSupport)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by Tech Support")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = TechSupport, fill = TechSupport)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Tech Support") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, StreamingTV

ggplot(churn\_clean, aes(x = Tenure, y = StreamingTV, fill = StreamingTV)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = StreamingTV, fill = StreamingTV)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = StreamingTV)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Streaming TV", x = "Length of Tenure", title = "Length of Tenure by Streaming TV",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = StreamingTV)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by StreamingTV")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = StreamingTV, fill = StreamingTV)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Streaming TV") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, Streaming Movies

ggplot(churn\_clean, aes(x = Tenure, y = StreamingMovies, fill = StreamingMovies)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = StreamingMovies, fill = StreamingMovies)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = StreamingMovies)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Streaming Movies", x = "Length of Tenure", title = "Length of Tenure by Streaming Movies",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = StreamingMovies)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by StreamingMovies")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = StreamingMovies, fill = StreamingMovies)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Streaming Movies") +

  theme(legend.position = "none")

#Stacked bar chart Tenure, PaperlessBilling

ggplot(churn\_clean, aes(x = Tenure, y = PaperlessBilling, fill = PaperlessBilling)) +

  geom\_col(position = "stack")

#Scatter plot

ggplot(churn\_clean, aes(x = Tenure, y = PaperlessBilling, fill = PaperlessBilling)) +

  geom\_point(color = "steelblue") +

  geom\_smooth(method = "lm",

              formula = y ~ poly(x,2),

              color = "indianred2")

#Line plot

ggplot(churn\_clean, aes(x = Tenure, y = PaperlessBilling)) +

  geom\_line(size = 2, color = "lightgrey") +

  geom\_point(size = .5, color = "steelblue") +

  labs(y = "Paperless Billing", x = "Length of Tenure", title = "Length of Tenure by Paperless Billing",

       subtitle = "Telecom Churn Data", caption = "Western Governors University")

#Density plot

ggplot(churn\_clean, aes(x = Tenure, fill = PaperlessBilling)) +

  geom\_density(alpha = 0.4) +

  labs(title = "Length of Tenure by Paperless Billing")

#Density Ridge

ggplot(churn\_clean, aes(x = Tenure, y = PaperlessBilling, fill = PaperlessBilling)) +

  geom\_density\_ridges() +

  theme\_ridges() +

  labs("Length of Tenure by Paperless Billing") +

  theme(legend.position = "none")

#Encode categorical variables [in-text citation: (Statistics Globe, n.d.)]

#Encode Churn

churn\_clean$Churn <- as.character(churn\_clean$Churn)

churn\_clean$Churn[churn\_clean$Churn == "Yes"] <- 1

churn\_clean$Churn[churn\_clean$Churn == "No"] <- 0

churn\_clean$Churn <- as.factor(churn\_clean$Churn)

#Encode Techie

churn\_clean$Techie <- as.character(churn\_clean$Techie)

churn\_clean$Techie[churn\_clean$Techie == "Yes"] <- 1

churn\_clean$Techie[churn\_clean$Techie == "No"] <- 0

churn\_clean$Techie <- as.factor(churn\_clean$Techie)

#Encode Port Modem

churn\_clean$Port\_modem <- as.character(churn\_clean$Port\_modem)

churn\_clean$Port\_modem[churn\_clean$Port\_modem == "Yes"] <- 1

churn\_clean$Port\_modem[churn\_clean$Port\_modem == "No"] <- 0

churn\_clean$Port\_modem <- as.factor(churn\_clean$Port\_modem)

#Encode Port Tablet

churn\_clean$Tablet <- as.character(churn\_clean$Tablet)

churn\_clean$Tablet[churn\_clean$Tablet == "Yes"] <- 1

churn\_clean$Tablet[churn\_clean$Tablet == "No"] <- 0

churn\_clean$Tablet <- as.factor(churn\_clean$Tablet)

#Encode Phone

churn\_clean$Phone <- as.character(churn\_clean$Phone)

churn\_clean$Phone[churn\_clean$Phone == "Yes"] <- 1

churn\_clean$Phone[churn\_clean$Phone == "No"] <- 0

churn\_clean$Phone <- as.factor(churn\_clean$Phone)

#Encode Multiple

churn\_clean$Multiple <- as.character(churn\_clean$Multiple)

churn\_clean$Multiple[churn\_clean$Multiple == "Yes"] <- 1

churn\_clean$Multiple[churn\_clean$Multiple == "No"] <- 0

churn\_clean$Multiple <- as.factor(churn\_clean$Multiple)

#Encode Online Security

churn\_clean$OnlineSecurity <- as.character(churn\_clean$OnlineSecurity)

churn\_clean$OnlineSecurity[churn\_clean$OnlineSecurity == "Yes"] <- 1

churn\_clean$OnlineSecurity[churn\_clean$OnlineSecurity == "No"] <- 0

churn\_clean$OnlineSecurity <- as.factor(churn\_clean$OnlineSecurity)

#Encode Online Backup

churn\_clean$OnlineBackup <- as.character(churn\_clean$OnlineBackup)

churn\_clean$OnlineBackup[churn\_clean$OnlineBackup == "Yes"] <- 1

churn\_clean$OnlineBackup[churn\_clean$OnlineBackup == "No"] <- 0

churn\_clean$OnlineBackup <- as.factor(churn\_clean$OnlineBackup)

#Encode Device Protection

churn\_clean$DeviceProtection <- as.character(churn\_clean$DeviceProtection)

churn\_clean$DeviceProtection[churn\_clean$DeviceProtection == "Yes"] <- 1

churn\_clean$DeviceProtection[churn\_clean$DeviceProtection == "No"] <- 0

churn\_clean$DeviceProtection <- as.factor(churn\_clean$DeviceProtection)

#Encode Tech Support

churn\_clean$TechSupport <- as.character(churn\_clean$TechSupport)

churn\_clean$TechSupport[churn\_clean$TechSupport == "Yes"] <- 1

churn\_clean$TechSupport[churn\_clean$TechSupport == "No"] <- 0

churn\_clean$TechSupport <- as.factor(churn\_clean$TechSupport)

#Encode Streaming TV

churn\_clean$StreamingTV <- as.character(churn\_clean$StreamingTV)

churn\_clean$StreamingTV[churn\_clean$StreamingTV == "Yes"] <- 1

churn\_clean$StreamingTV[churn\_clean$StreamingTV == "No"] <- 0

churn\_clean$StreamingTV <- as.factor(churn\_clean$StreamingTV)

#Encode Streaming Movies

churn\_clean$StreamingMovies <- as.character(churn\_clean$StreamingMovies)

churn\_clean$StreamingMovies[churn\_clean$StreamingMovies == "Yes"] <- 1

churn\_clean$StreamingMovies[churn\_clean$StreamingMovies == "No"] <- 0

churn\_clean$StreamingMovies <- as.factor(churn\_clean$StreamingMovies)

#Encode Paperless Billing

churn\_clean$PaperlessBilling <- as.character(churn\_clean$PaperlessBilling)

churn\_clean$PaperlessBilling[churn\_clean$PaperlessBilling == "Yes"] <- 1

churn\_clean$PaperlessBilling[churn\_clean$PaperlessBilling == "No"] <- 0

churn\_clean$PaperlessBilling <- as.factor(churn\_clean$PaperlessBilling)

#One-Hot Encoding Area

library(fastDummies)

A\_treat <- dummy\_cols(churn\_clean, select\_columns = "Area")

#One-Hot Encoding Marital

M\_treat <- dummy\_cols(churn\_clean, select\_columns = "Marital")

#One-Hot Encoding Gender

G\_treat <- dummy\_cols(churn\_clean, select\_columns = "Gender")

#One-Hot Encoding Contract

C\_treat <- dummy\_cols(churn\_clean, select\_columns = "Contract")

#One-Hot Encoding InternetService

I\_treat <- dummy\_cols(churn\_clean, select\_columns = "InternetService")

#One-Hot Encoding PaymentMethod

P\_treat <- dummy\_cols(churn\_clean, select\_columns = "PaymentMethod")

library(tidyverse)

#Put all df in a list [in-text citation: (Zach, 2021)]

df\_list <- list(churn\_clean, A\_treat, M\_treat, G\_treat, C\_treat, I\_treat, P\_treat)

#Merge all data frames together using mutate[in-text citation: (Zach, 2021)]

library (dplyr)

Mutate\_Churn <- mutate(churn\_clean, A\_treat, C\_treat, G\_treat, I\_treat, M\_treat, P\_treat)

#Drop columns by name that were duplicates to the variables one-hot encoded

Mutate\_Churn1 <- subset(Mutate\_Churn, select = -c(Area, Contract, Gender, InternetService, Marital, PaymentMethod))

#Rename columns with unexpected symbol

colnames(Mutate\_Churn1)[colnames(Mutate\_Churn1) == 'Marital\_Never Married'] <- 'Marital\_NeverMarried'

colnames(Mutate\_Churn1)[colnames(Mutate\_Churn1) == 'Contract\_One year'] <- 'Contract\_OneYear'

colnames(Mutate\_Churn1)[colnames(Mutate\_Churn1) == 'Contract\_Two Year'] <- 'Contract\_TwoYear'

colnames(Mutate\_Churn1)[colnames(Mutate\_Churn1) == 'Contract\_Month-to-month'] <- 'Contract\_Month\_To\_Month'

colnames(Mutate\_Churn1)[colnames(Mutate\_Churn1) == 'InternetService\_Fiber Optic'] <- 'InternetService\_FiberOptic'

colnames(Mutate\_Churn1)[colnames(Mutate\_Churn1) == 'PaymentMethod\_Bank Transfer(automatic)'] <- 'PaymentMethod\_BankTransfer'

colnames(Mutate\_Churn1)[colnames(Mutate\_Churn1) == 'PaymentMethod\_Credit Card (automatic)'] <- 'PaymentMethod\_CreditCard'

colnames(Mutate\_Churn1)[colnames(Mutate\_Churn1) == 'PaymentMethod\_Electronic Check'] <- 'PaymentMethod\_ElectronicCheck'

colnames(Mutate\_Churn1)[colnames(Mutate\_Churn1) == 'PaymentMethod\_Mailed Check'] <- 'PaymentMethod\_MailedCheck'

#Reduce Model Payment Method, Marital as cardinality is too great and we do not want to proliferate

#K-1 method

Mutate\_Churn2 <- subset(Mutate\_Churn1, select = -c(Job, TimeZone, Population,

                                                   Lat, Lng, Zip, County, State,

                                                   City, UID, Interaction, Customer\_id,

                                                   PaymentMethod\_BankTransfer,

                                                   PaymentMethod\_CreditCard,

                                                   PaymentMethod\_ElectronicCheck,

                                                   PaymentMethod\_MailedCheck,

                                                   Marital\_Divorced, Marital\_Married,

                                                   Marital\_NeverMarried, Marital\_Widowed,

                                                   Marital\_Separated, CaseOrder, Area\_Rural,

Contract\_Month\_To\_Month, Gender\_Female,

InternetService\_DSL))

str(Mutate\_Churn2)

#Export clean data

write.csv(Mutate\_Churn2, "C:/Users/ntrei/OneDrive/Documents/MSDA/MutateChurn2.209.2.csv")

#Visualize correlation

library(corrr)

Mutate\_Churn2 %>%

  dplyr::select(where(is.numeric)) %>%

  correlate() %>%

  shave() %>%

  rplot(print\_cor = TRUE) +

  theme(axis.text.x = element\_text(angle = 90, hjust = 1))

str(Mutate\_Churn2)

describe(Mutate\_Churn2)

####DF

DF <- Mutate\_Churn2[c('Children',

                      'Age',

                      'Income',

                      'Outage\_sec\_perweek',

                      'Email',

                      'Contacts',

                      'Yearly\_equip\_failure',

                      'Techie',

                      'Port\_modem',

                      'Tablet',

                      'Phone',

                      'Multiple',

                      'OnlineSecurity',

                      'OnlineBackup',

                      'DeviceProtection',

                      'TechSupport',

                      'StreamingTV',

                      'StreamingMovies',

                      'PaperlessBilling',

                      'Churn',

                      'MonthlyCharge',

                      'Bandwidth\_GB\_Year',

                      'Timely\_Response',

                      'Timely\_Fixes',

                      'Timely\_Replacements',

                      'Reliability',

                      'Options',

                      'Respectful\_Response',

                      'Courteous\_Exchange',

                      'Active\_Listening',

                      'Area\_Suburban',

                      'Area\_Urban',

                      'Contract\_OneYear',

                      'Contract\_TwoYear',

                      'Gender\_Male',

                      'Gender\_Nonbinary',

                      'InternetService\_FiberOptic',

                      'InternetService\_None',

                      'Tenure')]

#Partition data, set seed

set.seed(1234)

#Proportion for reproducibility- Train

Train\_prop <- 0.7

#Partition data into training and test sets

Train\_indices <- createDataPartition(DF$Tenure, p = Train\_prop, list = FALSE)

Train\_data <- DF[Train\_indices, ]

Test\_data <- DF[-Train\_indices, ]

#CSV files

write.csv(Train\_data, file = "Training\_data.2.csv", row.names = FALSE)

write.csv(Test\_data, file = "Test\_data.2.csv", row.names = FALSE)

#Load test

Train\_data <- read.csv("Training\_data.2.csv")

Test\_data <- read.csv("Test\_data.2.csv")

#Extract variables

X\_Train <- Train\_data[, c('Children',

                          'Age',

                          'Income',

                          'Outage\_sec\_perweek',

                          'Email',

                          'Contacts',

                          'Yearly\_equip\_failure',

                          'Techie',

                          'Port\_modem',

                          'Tablet',

                          'Phone',

                          'Multiple',

                          'OnlineSecurity',

                          'OnlineBackup',

                          'DeviceProtection',

                          'TechSupport',

                          'StreamingTV',

                          'StreamingMovies',

                          'PaperlessBilling',

                          'Churn',

                          'MonthlyCharge',

                          'Bandwidth\_GB\_Year',

                          'Timely\_Response',

                          'Timely\_Fixes',

                          'Timely\_Replacements',

                          'Reliability',

                          'Options',

                          'Respectful\_Response',

                          'Courteous\_Exchange',

                          'Active\_Listening',

                          'Area\_Suburban',

                          'Area\_Urban',

                          'Contract\_OneYear',

                          'Contract\_TwoYear',

                          'Gender\_Male',

                          'Gender\_Nonbinary',

                          'InternetService\_FiberOptic',

                          'InternetService\_None')]

X\_Test <- Test\_data[, c('Children',

                        'Age',

                        'Income',

                        'Outage\_sec\_perweek',

                        'Email',

                        'Contacts',

                        'Yearly\_equip\_failure',

                        'Techie',

                        'Port\_modem',

                        'Tablet',

                        'Phone',

                        'Multiple',

                        'OnlineSecurity',

                        'OnlineBackup',

                        'DeviceProtection',

                        'TechSupport',

                        'StreamingTV',

                        'StreamingMovies',

                        'PaperlessBilling',

                        'Churn',

                        'MonthlyCharge',

                        'Bandwidth\_GB\_Year',

                        'Timely\_Response',

                        'Timely\_Fixes',

                        'Timely\_Replacements',

                        'Reliability',

                        'Options',

                        'Respectful\_Response',

                        'Courteous\_Exchange',

                        'Active\_Listening',

                        'Area\_Suburban',

                        'Area\_Urban',

                        'Contract\_OneYear',

                        'Contract\_TwoYear',

                        'Gender\_Male',

                        'Gender\_Nonbinary',

                        'InternetService\_FiberOptic',

                        'InternetService\_None')]

#Create train and test sets

Y\_train <- Train\_data$Tenure

Y\_test <- Test\_data$Tenure

#Export Training and Test Data

write.csv(X\_Train, "C:/Users/ntrei/OneDrive/Documents/MSDA/XTrain209.2.csv")

write.csv(X\_Test, "C:/Users/ntrei/OneDrive/Documents/MSDA/XTest209.2.csv")

write.csv(Y\_train, "C:/Users/ntrei/OneDrive/Documents/MSDA/YTrain209.2.csv")

write.csv(Y\_test, "C:/Users/ntrei/OneDrive/Documents/MSDA/YTest209.2.csv")

#Lasso Regression [in text citation: (Poulson, B. 2020)]

#Load packages

pacman::p\_load(lars, magrittr, pacman, rio, tidyverse)

#Import as tibble

DF <- import("C:/Users/ntrei/OneDrive/Documents/MSDA/MutateChurn2.209.2.csv") %>%

  as\_tibble() %>%

  print()

#Get descriptives

DF %>% summary()

#Variables need to be the same scale

DF %<>%

  scale() %>%

  as\_tibble()

#Get descriptives again

DF %>% summary()

#Data as matrix for lars package

y <- DF %>% select(Churn) %>% as.matrix()

X <- DF %>% select(-Churn) %>% as.matrix()

#Compute model

fit\_step <- lars(X, y, type = "stepwise")

#Plot results

fit\_step %>% plot()

legend(

  "bottomleft",

  lwd = 2,

  col = (1:nrow(X)),

  legend = colnames(X),

  cex = .7

)

#Lasso regression

Fit\_lasso <- lars(X, y, type = "lasso")

#Plot results

Fit\_lasso %>% plot()

legend(

  "bottomleft",

  lwd = 2,

  col = (1:nrow(X)),

  legend = colnames(X),

  cex = .4

)

#View more results

Fit\_lasso$R2

Fit\_lasso$R2 %>% plot()

Fit\_lasso %>% coef()

#Fit model and calculate MSE and MAE

# Perform k-fold cross-validation to find the optimal lambda value

library(glmnet)

cv\_model <- cv.glmnet(x, y, alpha = 1, type.measure = "mse")

best\_lambda <- cv\_model$lambda.min # optimal lambda value

# Obtain predictions on the test set using the optimal lambda value

lasso\_pred <- predict(cv\_model, s = best\_lambda, newx = x)

# Calculate MSE on the test set

lasso\_mse <- mean((y - lasso\_pred)^2)

lasso\_mse

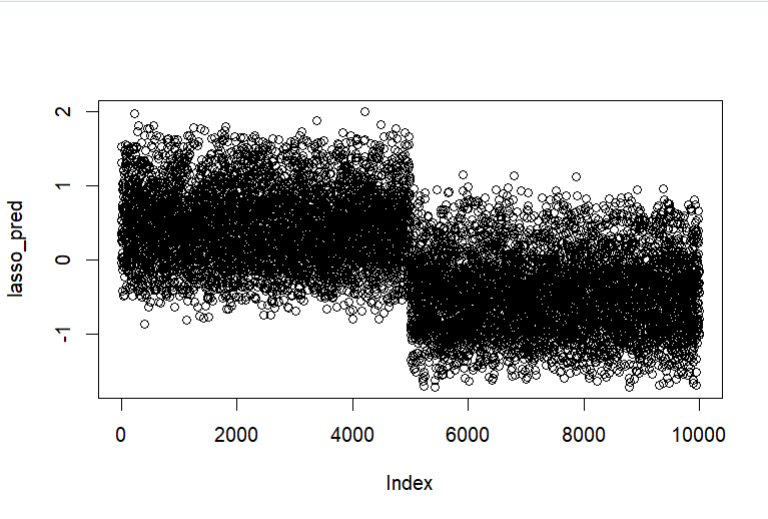
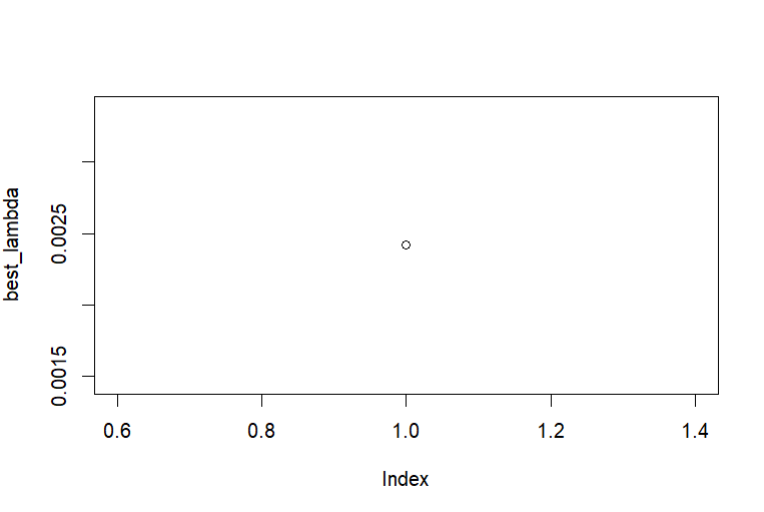
plot(lasso\_mse)

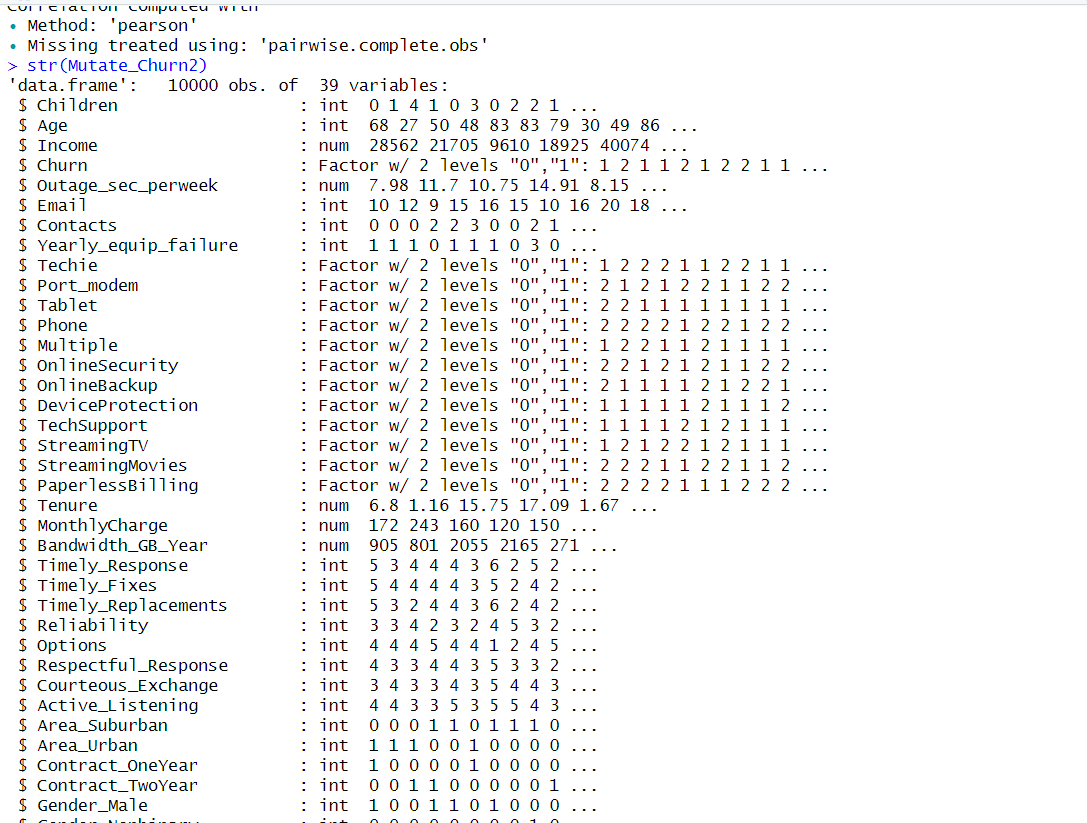
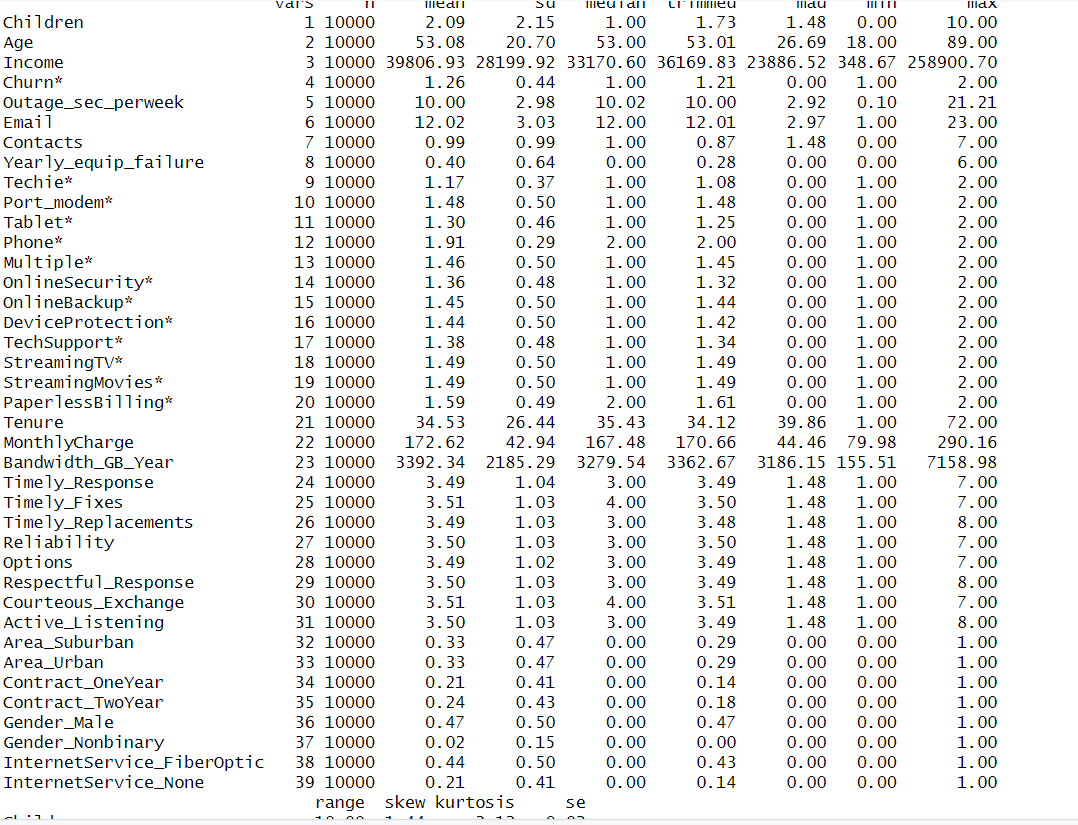
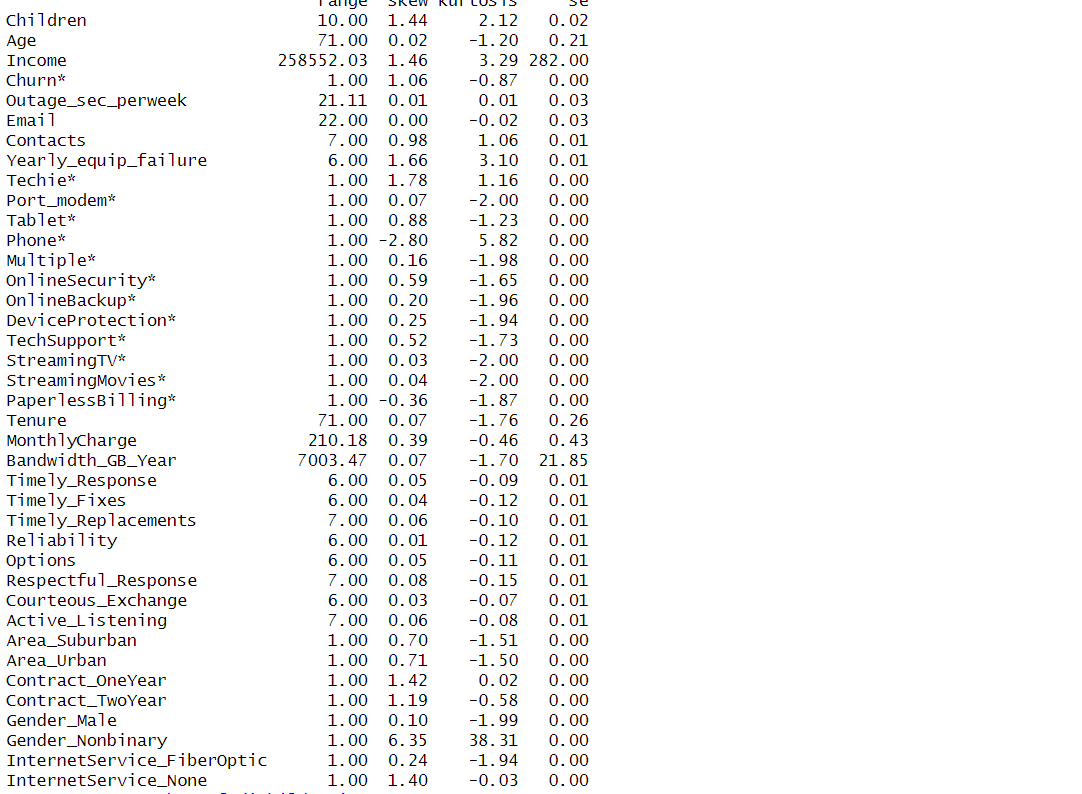
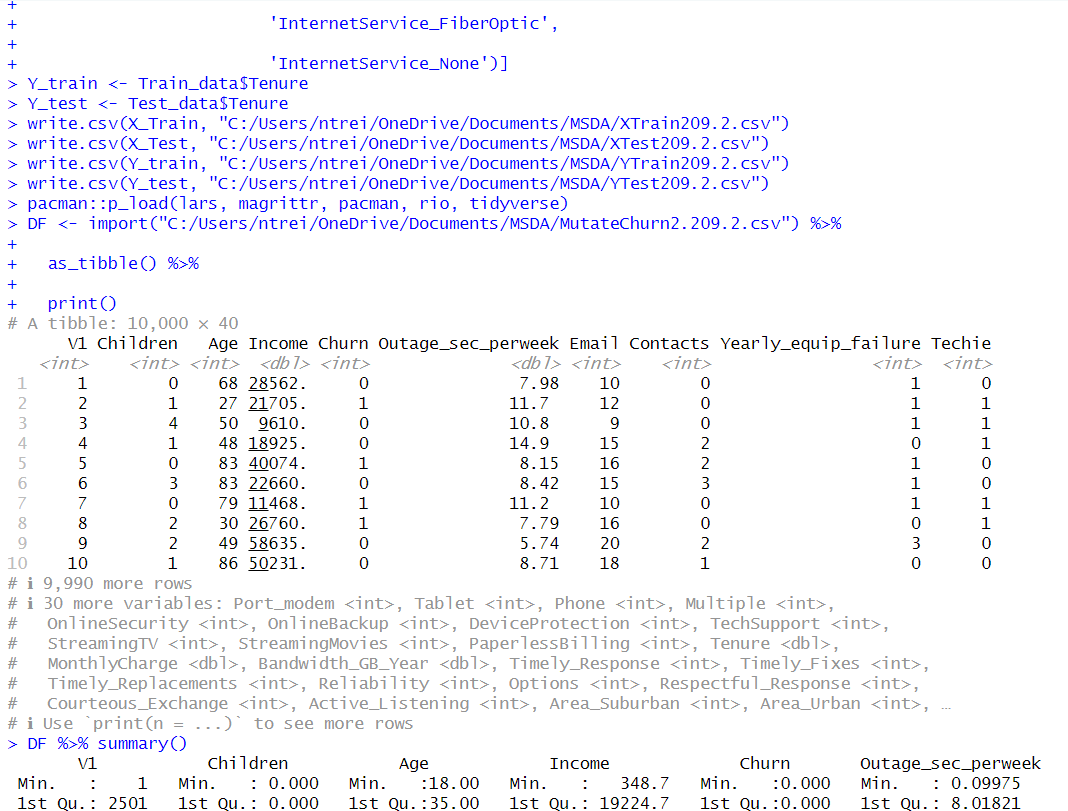
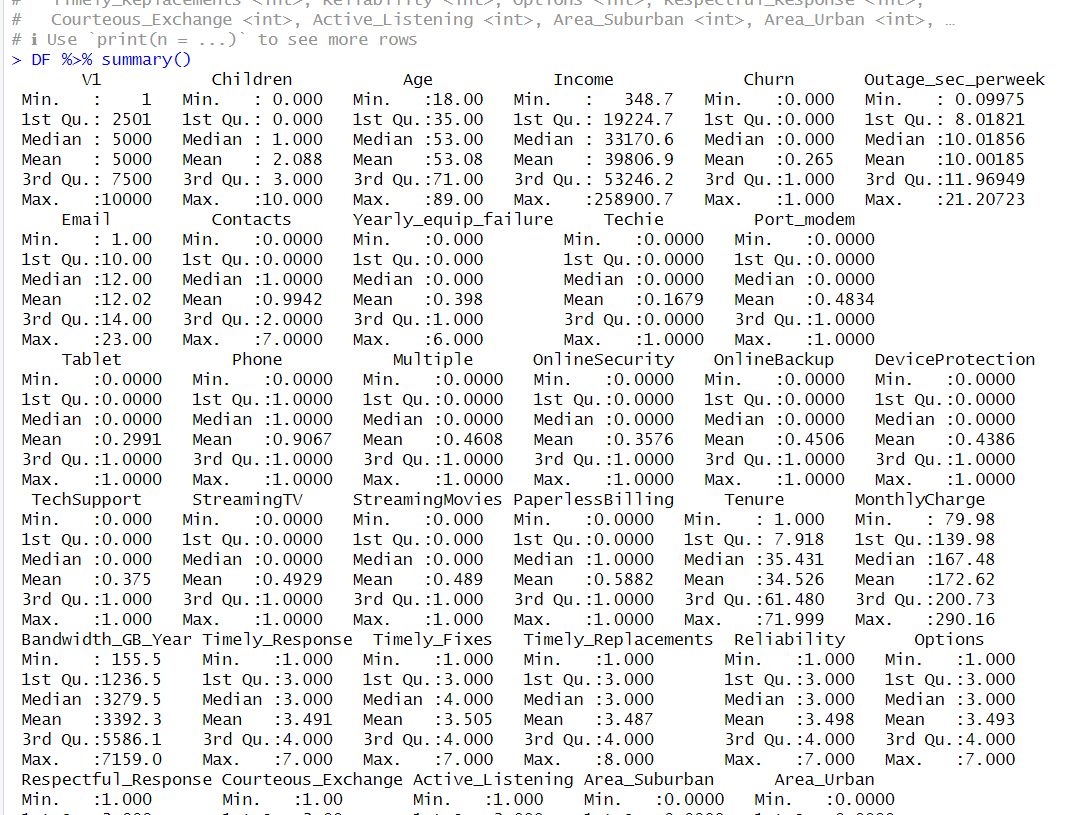
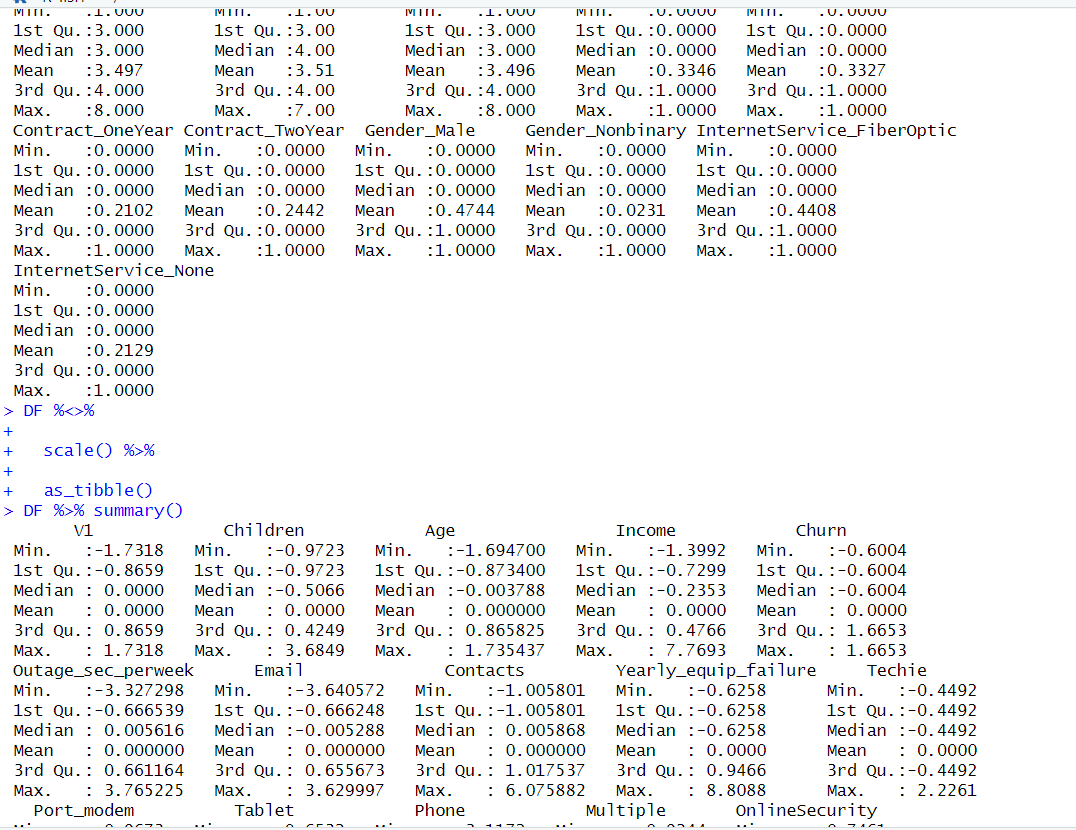
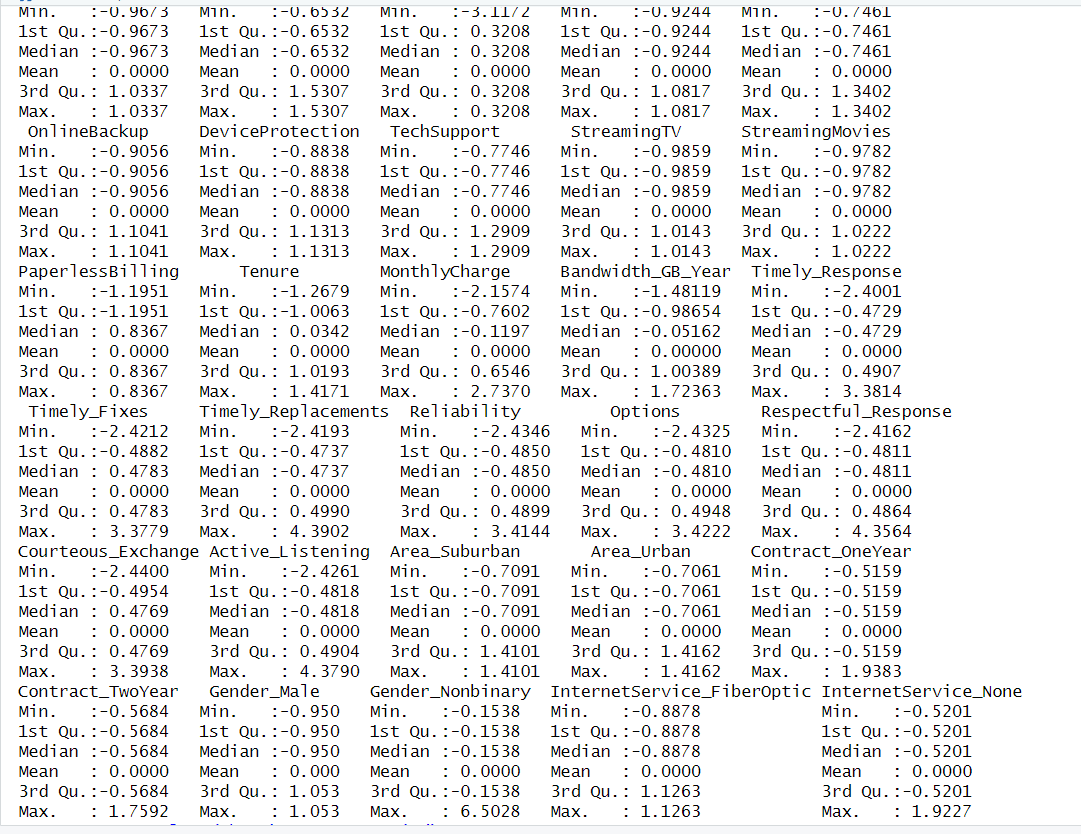
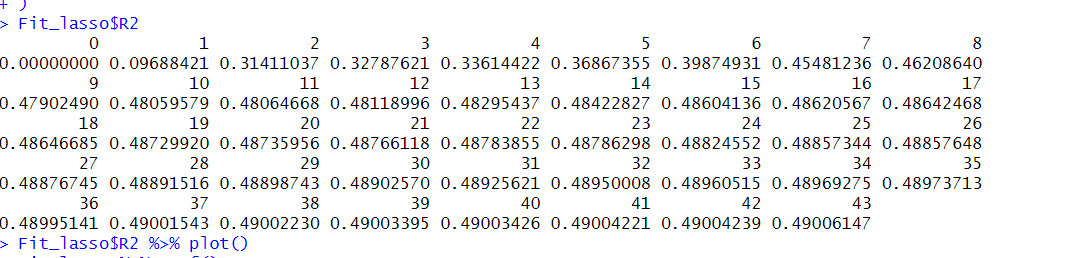
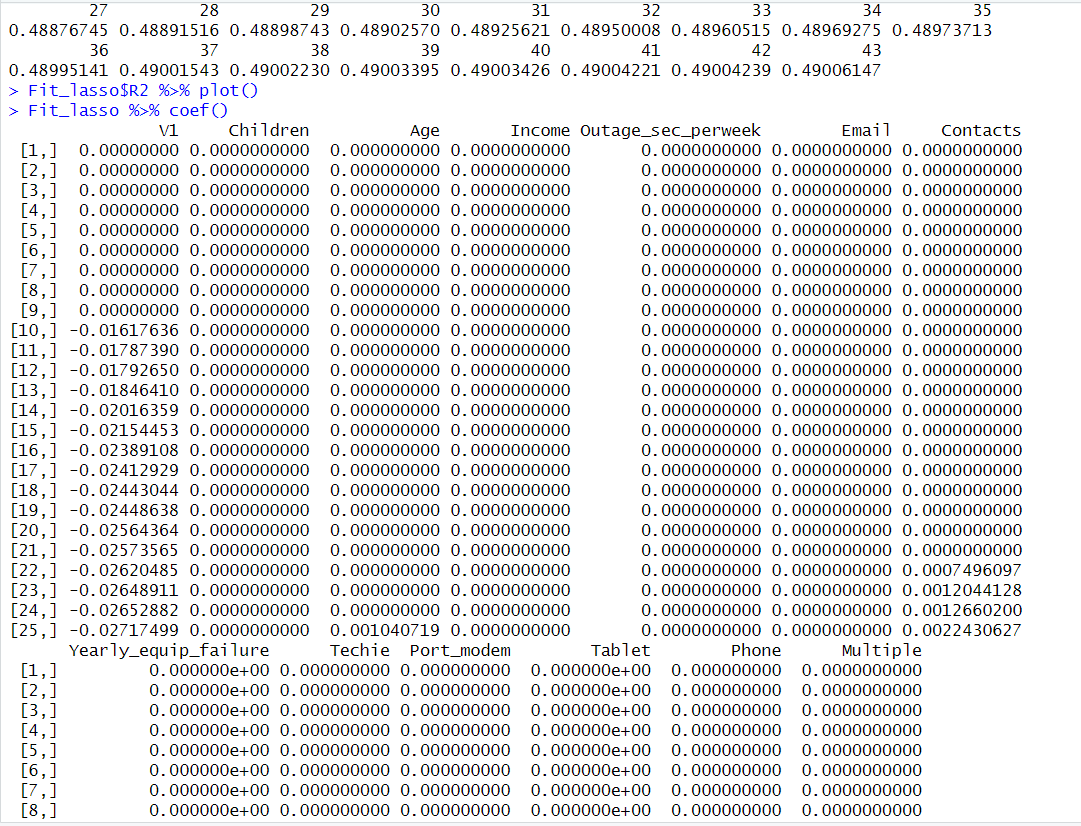
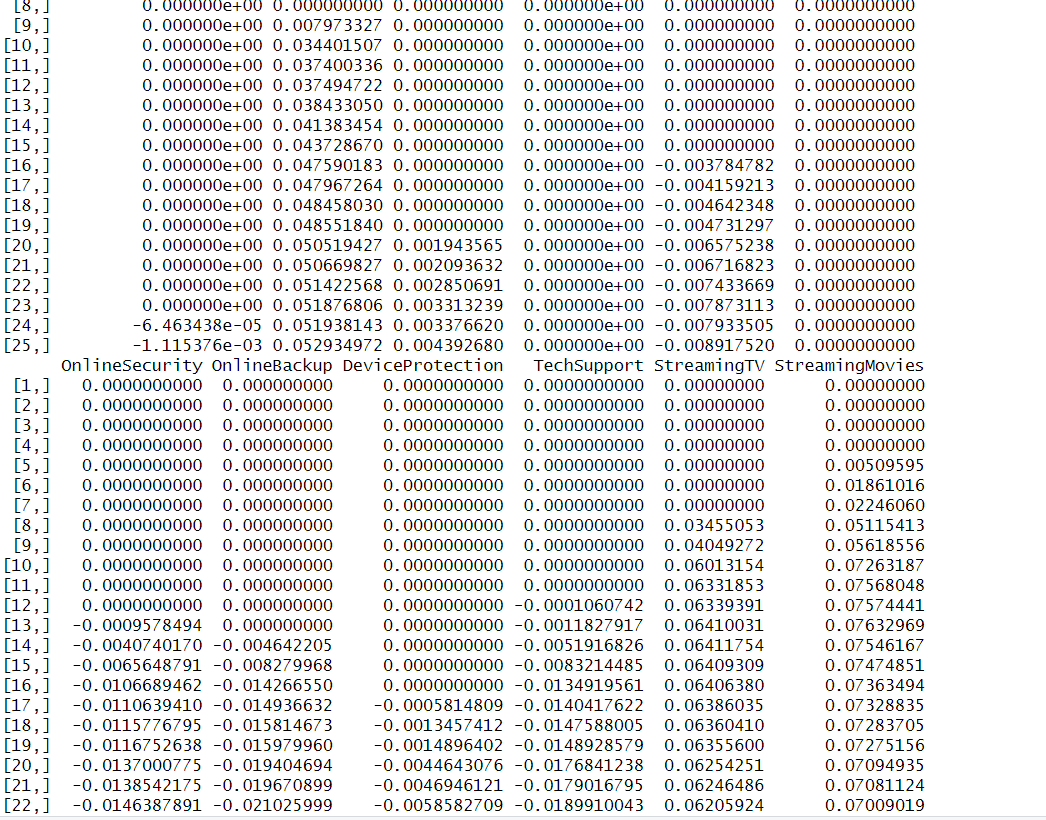
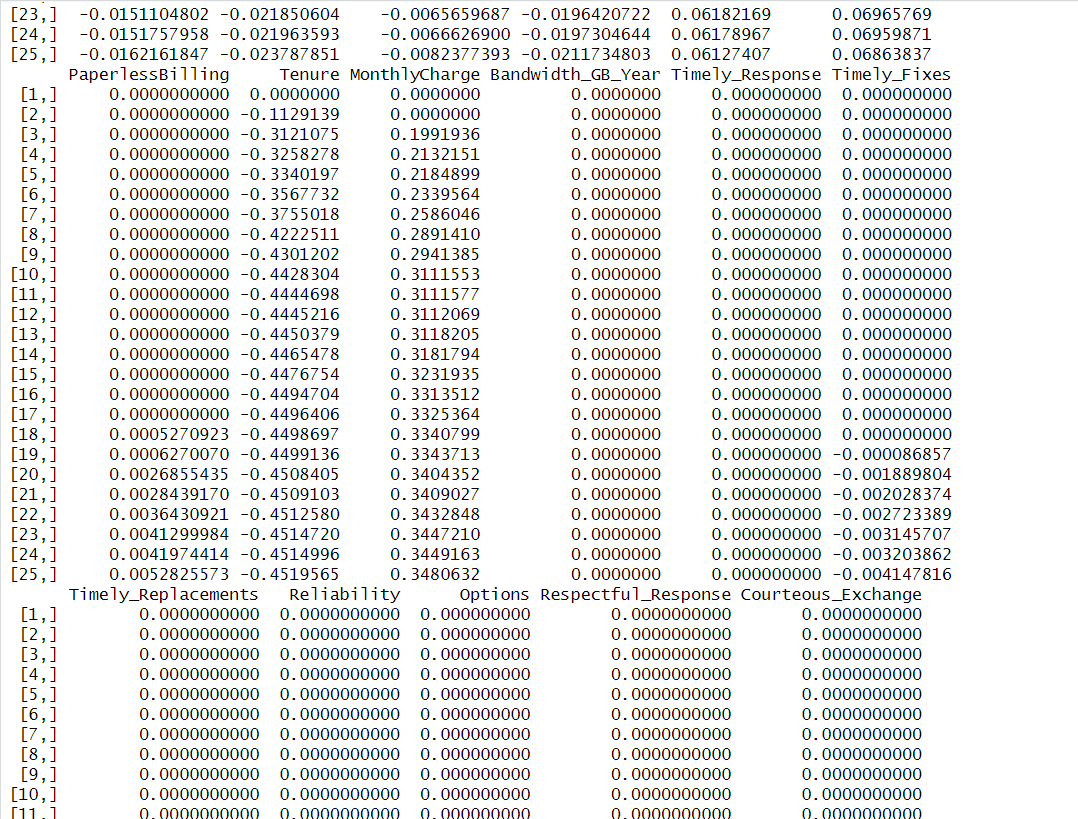
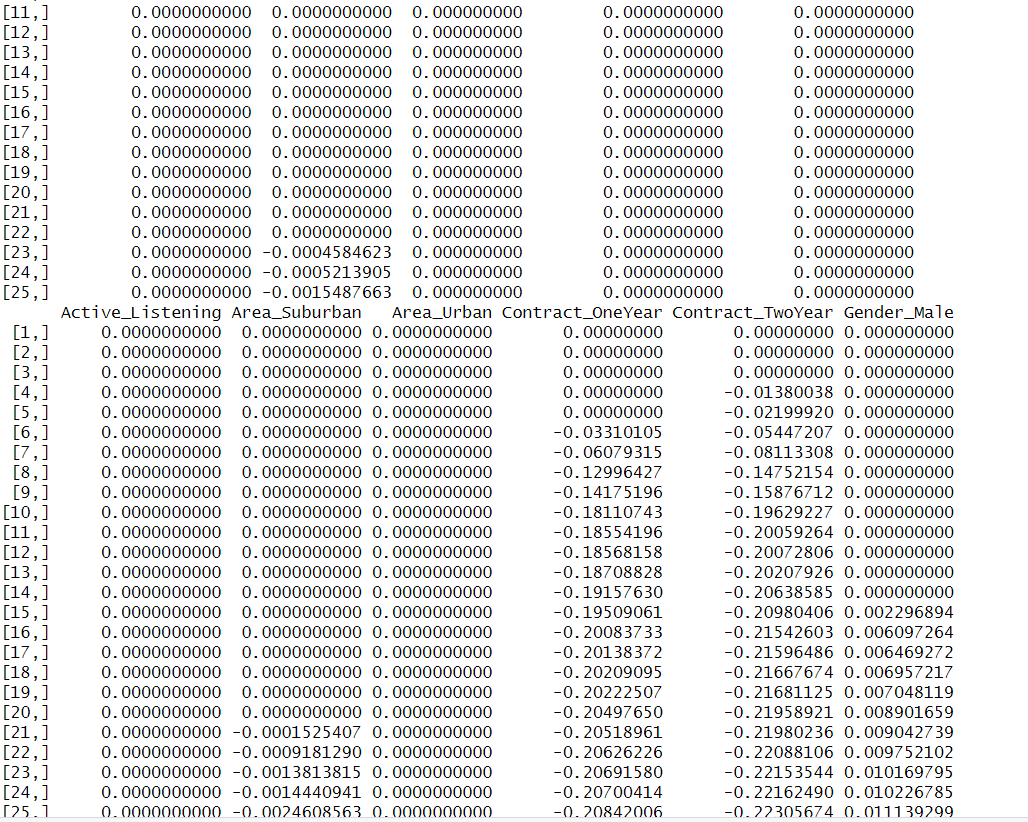
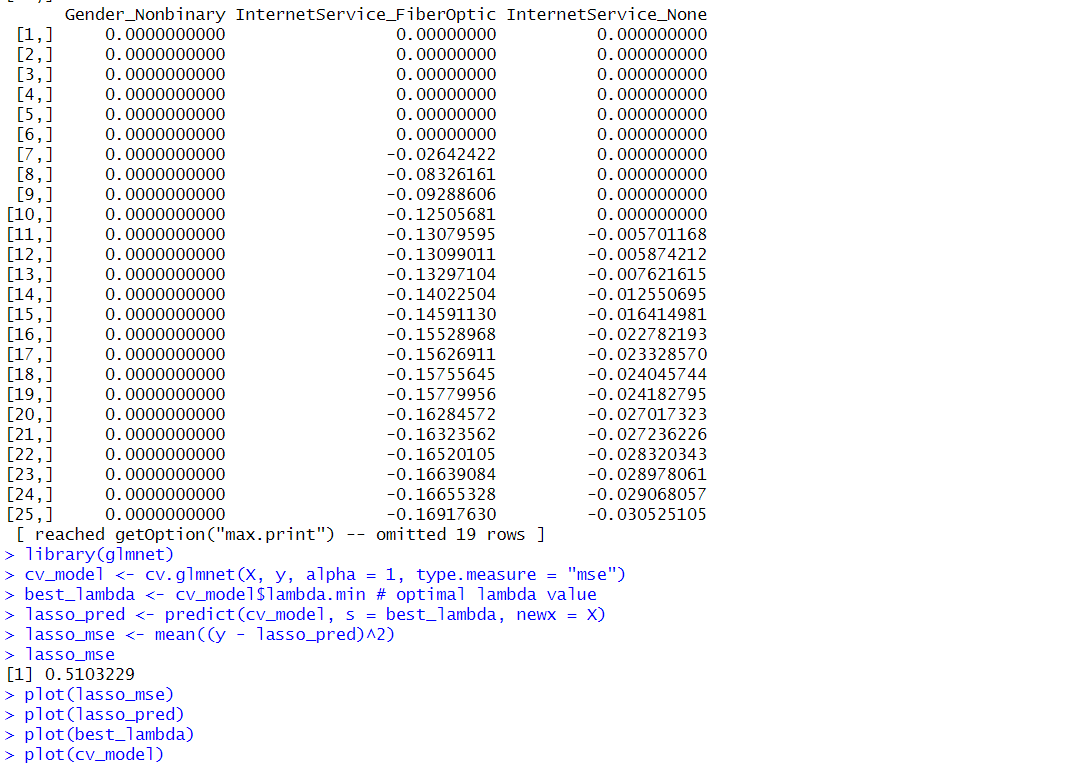
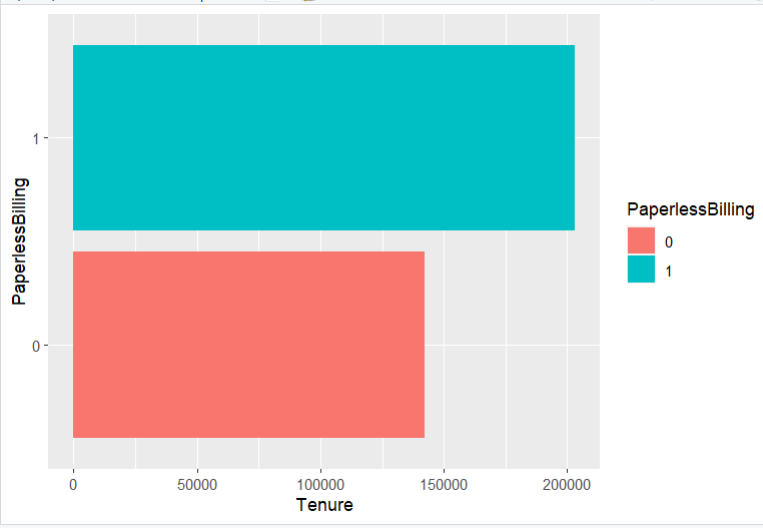
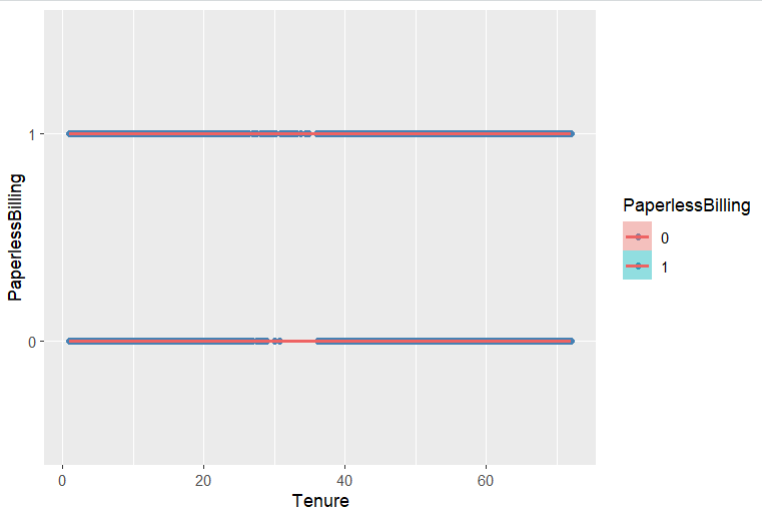
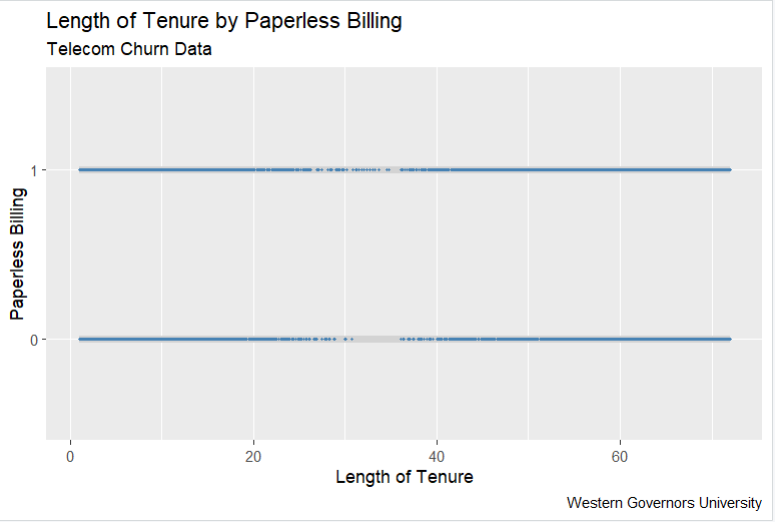
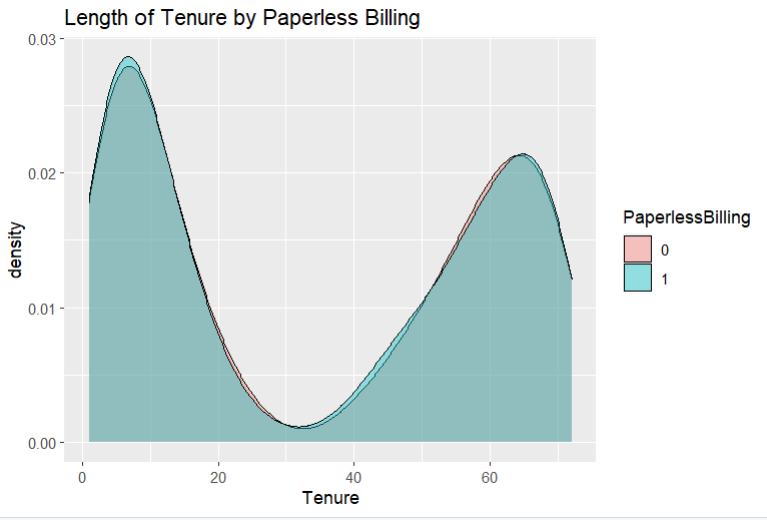
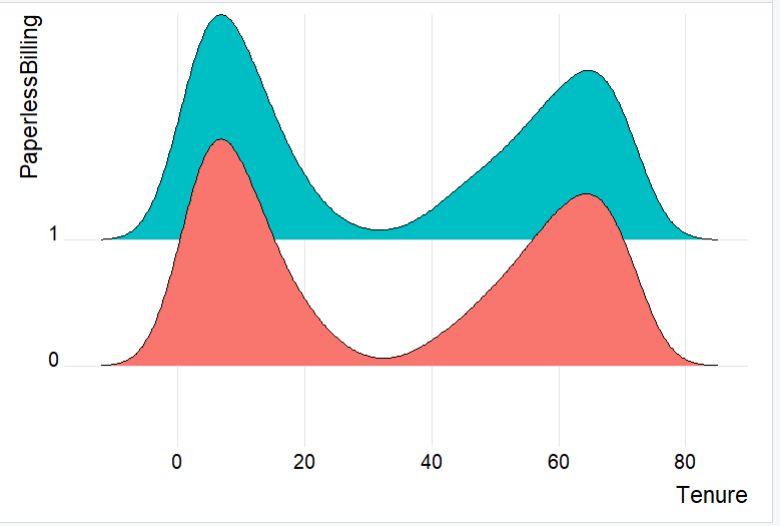
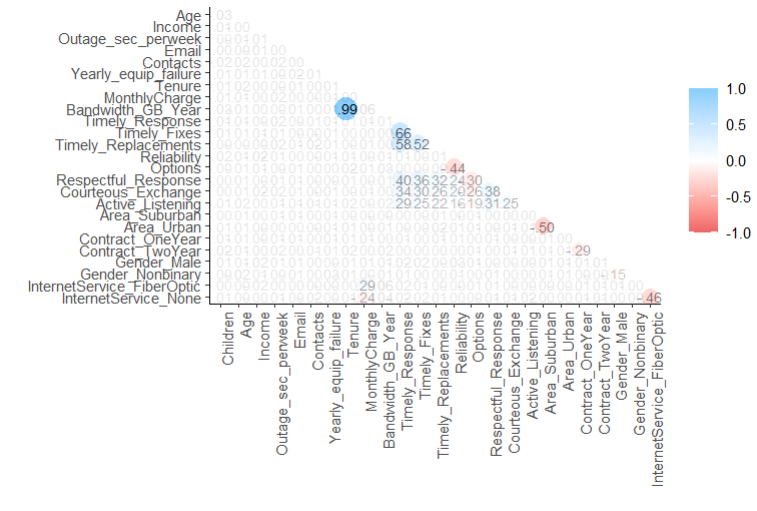
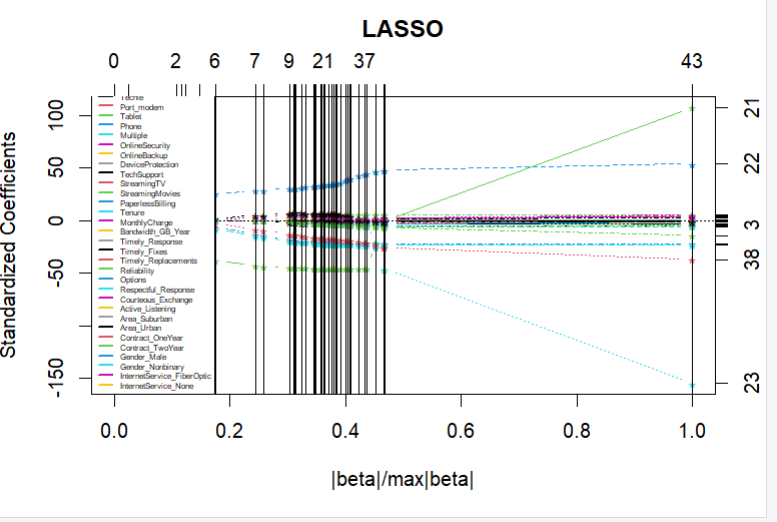
plot(lasso\_pred)

plot(best\_lambda)

plot(cv\_model)

A graph with numbers and lines

Description automatically generatedA graph with dots and numbers

Description automatically generated

**E. Part V: Data Summary and Implications**

**1.** The accuracy score and MSE or mean standard error score which was calculated to be 0.5103229. The MSE score is not as close to zero as I would like but it is not a bad score. The closer to zero the MSE score is, the more accurate the model is.

**2.** The result and implication of the analysis is that Tenure is affected by the independent variables and therefore business decisions can be made to increase the length of customer tenure.

**3.** A limitation of this analysis is that Lasso Regression assumes all independent variables are independent, but this may not always be the case. For example, two independent variables such as income and children or other variables may have a relationship.

**4.** The recommended course of action is to make changes to the business model utilizing this new data to increase the length of tenure of a customer.

A graph with numbers and lines

Description automatically generated

**F.** **Part VI: Demonstration**

**1.** See attached Panopto recording.

**G.** Sources

Wikipedia. (2024). *Lasso regression assumptions.*

<https://www.bing.com/ck/a?!&&p=7cb4047359136802JmltdHM9MTcwNTE5MDQwMCZpZ3VpZD0yZDA4ZDAyYS1kODE2LTY5MzktM2E4MS1jM2Q5ZDk0MDY4ODQmaW5zaWQ9NTQ5Ng&ptn=3&ver=2&hsh=3&fclid=2d08d02a-d816-6939-3a81-c3d9d9406884&psq=lasso+regression+assumptions&u=a1aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvTGFzc29fKHN0YXRpc3RpY3MpIzp-OnRleHQ9VGhlJTIwbGFzc28lMjBtZXRob2QlMjBhc3N1bWVzJTIwdGhhdCUyMHRoZSUyMGNvZWZmaWNpZW50cyUyMG9mLExhc3NvJTIwd2FzJTIwb3JpZ2luYWxseSUyMGZvcm11bGF0ZWQlMjBmb3IlMjBsaW5lYXIlMjByZWdyZXNzaW9uJTIwbW9kZWxzLg&ntb=1>

R4DS. (2023). Data Visualizations with R.

[Data Visualization with R: Bivariate Graphs (datavisr01 4) (youtube.com)](https://www.youtube.com/watch?v=q4hWHd9eR7s)

Poulson, B. (2020). *Predicting outcomes with lasso regression.*

[Predicting outcomes with lasso regression (linkedin.com)](https://www.linkedin.com/learning/r-essential-training-part-2-modeling-data/predicting-outcomes-with-lasso-regression?resume=false&u=2045532)

[R code tutorial: regression -- LASSO (Part 1) (youtube.com)](https://www.youtube.com/watch?v=WnvUd7AbmiU)

Elleh, F. (2023). *D209 Task 1: Expectations and Data Preprocessing- Python.*

[D209 Task 1: Expectations and Data Preprocessing - Python (panopto.com)](https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=7329050b-3de4-412a-9ff4-b0790009d02b)

Rai, B. (2019). *Naive Bayes Classification with R.*

[Naive Bayes Classification with R | Example with Steps (youtube.com)](https://www.youtube.com/watch?v=RLjSQdcg8AM)