# MULTIPLE REGRESSION FOR PREDICTIVE MODELING

## Part I: Research Question

**How does the likelyhood of customer churn depend on tenure and data usage?**

**The goal of this research project is to determine how much customer churn is affected by the length of time a customer has been with their service provider, as well as a customer's monthly data usage.**

## Part II: Method Justification

### According to "Logistic Regression: A Brief Primer" by Jill C. Stoltzfus:

**"Basic assumptions that must be met for logistic regression include independence of errors, linearity in the logit for continuous variables, absence of multicollinearity, and lack of strongly influential outliers. Additionally, there should be an adequate number of events per independent variable to avoid an overfit model, with commonly recommended minimum “rules of thumb” ranging from 10 to 20 events per covariate." (Stoltzfus, 2011).**

**This problem is clearly a binary classification problem. Customer churn can take on one of two values, either 1 or 0. I chose R for this project for several reasons. R has several built in fucntions for dealing with problems like this. The glm function in R specifically takes a non-linear model and fits it to the input data. It can fit binomial, gaussian, Gamma, inverse, poisson, quasi, quasibinomial, and quasipoisson functions. R was also built to handle large data sets making it an ideal candidate to deal with a problem like the one I am trying to solve. R is easy to use, it is extremely flexible thanks to the immense number of add-on packages it supports, and it features a wide range of graphical options to make for easier data analysis.**

**I used logistic regression for this specifc problem, rather then linear regression, because the output must be between 0 and 1.**

## Part III: Data Preparation

My goals for data preparation were to remove unnecessary variables, and then scale the remaining variables so I could use logistic regression on my data. I removed the columns with the -c command in R. Then, I replaced all of the Yes/No values with 1/0. Next, I divided the individual elements of each column by their max value to scale the columns between 0 and 1.

In my initial data preparation, I removed all unnecessary columns. This left me with the following columns: Churn, Outage sec per week, Yearly equip failure, Techie, Tenure, Monthly Charge, Bandwidth GB Year, Satisfaction Score, Port modem, Tablet, Phone, Multiple, Online Security, Online Backup, Device Protection, Tech Support, Streaming TV, Streaming Movies, and Paperless Billing.

Next, I created a new column called unsatisfaction Score, which is created using the formula 1 minus the Statisfaction Score. This new column was needed because a customer who is satisfied with their service is less likely to switch providers. I also added the 11 columns that represented add-on features, i.e., Port modem, Tablet, Phone, Multiple, Online Security, Online Backup, Device Protection, Tech Support, Streaming TV, Streaming Movies, and Paperless Billing, then divided the result by 11. I named this new column Features Score. I also created a column called nonfeatures_score, which uses the formula 1 minus the features_score.

The following is the code used to prepare/clean the data:

In [1]:
```
install.packages("readr");
install.packages("ggplot2");
install.packages("dplyr");
install.packages("broom");
install.packages("ggpubr");
install.packages("MASS");
install.packages("repr")
install.packages("caret", dependencies = TRUE)
```

```
package 'readr' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\ContactTracer\AppData\Local\Temp\Rtmpe4UuzP\downloaded_packages
package 'ggplot2' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\ContactTracer\AppData\Local\Temp\Rtmpe4UuzP\downloaded_packages
package 'dplyr' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\ContactTracer\AppData\Local\Temp\Rtmpe4UuzP\downloaded_packages
package 'broom' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\ContactTracer\AppData\Local\Temp\Rtmpe4UuzP\downloaded_packages
package 'ggpubr' successfully unpacked and MD5 sums checked
```

```
The downloaded binary packages are in
        C:\Users\ContactTracer\AppData\Local\Temp\Rtmpe4UuzP\downloaded_packages
package 'MASS' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\ContactTracer\AppData\Local\Temp\Rtmpe4UuzP\downloaded_packages
package 'repr' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\ContactTracer\AppData\Local\Temp\Rtmpe4UuzP\downloaded_packages
package 'caret' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\ContactTracer\AppData\Local\Temp\Rtmpe4UuzP\downloaded_packages
```

In [2]:
```
library(readr)
library(ggplot2)
library(dplyr)
library(broom)
library(MASS)
library(repr)
library(caret)
```

```
Warning message:
"package 'readr' was built under R version 3.6.3"Warning message:
"package 'ggplot2' was built under R version 3.6.3"Warning message:
"package 'dplyr' was built under R version 3.6.3"
Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union

Warning message:
"package 'broom' was built under R version 3.6.3"Warning message:
"package 'MASS' was built under R version 3.6.3"
Attaching package: 'MASS'

The following object is masked from 'package:dplyr':

    select

Warning message:
"package 'repr' was built under R version 3.6.3"Warning message:
"package 'caret' was built under R version 3.6.3"Loading required package: lattice
Warning message:
"package 'lattice' was built under R version 3.6.3"
```

In [3]:
```
data1 <- read_csv("churn_clean_new.csv");
```

```
-- Column specification --------------------------------------------------------
--------------------------------
cols(
  .default = col_character(),
  CaseOrder = col_double(),
  Zip = col_double(),
  Lat = col_double(),
  Lng = col_double(),
  Population = col_double(),
  Children = col_double(),
```

```
  Age = col_double(),
  Income = col_double(),
  Churn_num = col_double(),
  Outage_sec_perweek = col_double(),
  Email = col_double(),
  Contacts = col_double(),
  Yearly_equip_failure = col_double(),
  Tenure = col_double(),
  MonthlyCharge = col_double(),
  Bandwidth_GB_Year = col_double(),
  item1 = col_double(),
  item2 = col_double(),
  item3 = col_double(),
  item4 = col_double()
  # ... with 5 more columns
)
i Use `spec()` for the full column specifications.
```

In [4]:
```
spec(data1)
```

```
cols(
  CaseOrder = col_double(),
  Customer_id = col_character(),
  Interaction = col_character(),
  UID = col_character(),
  City = col_character(),
  State = col_character(),
  County = col_character(),
  Zip = col_double(),
  Lat = col_double(),
  Lng = col_double(),
  Population = col_double(),
  Area = col_character(),
  TimeZone = col_character(),
  Job = col_character(),
  Children = col_double(),
  Age = col_double(),
  Income = col_double(),
  Marital = col_character(),
  Gender = col_character(),
  Churn = col_character(),
  Churn_num = col_double(),
  Outage_sec_perweek = col_double(),
  Email = col_double(),
  Contacts = col_double(),
  Yearly_equip_failure = col_double(),
  Techie = col_character(),
  Contract = col_character(),
  Port_modem = col_character(),
  Tablet = col_character(),
  InternetService = col_character(),
  Phone = col_character(),
  Multiple = col_character(),
  OnlineSecurity = col_character(),
  OnlineBackup = col_character(),
  DeviceProtection = col_character(),
  TechSupport = col_character(),
  StreamingTV = col_character(),
  StreamingMovies = col_character(),
  PaperlessBilling = col_character(),
  PaymentMethod = col_character(),
  Tenure = col_double(),
  MonthlyCharge = col_double(),
  Bandwidth_GB_Year = col_double(),
```

```
  item1 = col_double(),
  item2 = col_double(),
  item3 = col_double(),
  item4 = col_double(),
  item5 = col_double(),
  item6 = col_double(),
  item7 = col_double(),
  item8 = col_double(),
  SatisfactionScore = col_double()
)
```

In [5]:
```
new_data1<-dplyr::select(data1, -c(1,2,3,4,5,6,7))
```

In [6]:
```
new_data2<-dplyr::select(new_data1, -c(11,12,14,15,16,17,18,19,20,21,22))
```

In [7]:
```
unstatisfaction_score<-1-new_data2$SatisfactionScore
```

In [8]:
```
new_data2$unstatisfaction_score<-unstatisfaction_score
```

In [9]:
```
new_data1$Techie<-ifelse(new_data1$Techie=='Yes', 1,0)

new_data1$Port_modem<-ifelse(new_data1$Port_modem=='Yes', 1,0)

new_data1$Tablet<-ifelse(new_data1$Tablet=='Yes', 1,0)

new_data1$Phone<-ifelse(new_data1$Phone=='Yes', 1,0)

new_data1$Multiple<-ifelse(new_data1$Multiple=='Yes', 1,0)

new_data1$OnlineSecurity<-ifelse(new_data1$OnlineSecurity=='Yes', 1,0)

new_data1$OnlineBackup<-ifelse(new_data1$OnlineBackup=='Yes', 1,0)

new_data1$DeviceProtection<-ifelse(new_data1$DeviceProtection=='Yes', 1,0)

new_data1$TechSupport<-ifelse(new_data1$TechSupport=='Yes', 1,0)

new_data1$StreamingTV<-ifelse(new_data1$StreamingTV=='Yes', 1,0)

new_data1$StreamingMovies<-ifelse(new_data1$StreamingMovies=='Yes', 1,0)

new_data1$PaperlessBilling<-ifelse(new_data1$PaperlessBilling=='Yes', 1,0)
```

In [10]:
```
new_data1$features_score = new_data1$Port_modem + new_data1$Tablet + new_data1$Phone +
```

In [11]:
```
new_data2<-dplyr::select(new_data1, -c(1,2,3,4,5,6,7,8,9,10))
```

In [12]:
```
new_data2<-dplyr::select(new_data2, -c(1,2,4,5,6,7,8,9,10))
```

In [13]:
```
new_data2$Churn<-ifelse(new_data2$Churn=='Yes', 1,0)
```

In [14]:
```
new_data2<-dplyr::select(new_data2, -c(2,3,4,5,6,7,8,9,10))
```

In [15]:
```
new_data2<-dplyr::select(new_data2, -c(2,3,4,5,9,10,11,12,13,14,15,16))
```

In [16]:
```
summary(new_data2)
```

```
     Churn           Tenure       MonthlyCharge    Bandwidth_GB_Year
 Min.   :0.000   Min.   : 1.000   Min.   : 79.98   Min.   : 155.5
 1st Qu.:0.000   1st Qu.: 7.918   1st Qu.:139.98   1st Qu.:1236.5
 Median :0.000   Median :35.431   Median :167.48   Median :3279.5
 Mean   :0.265   Mean   :34.526   Mean   :172.62   Mean   :3392.3
 3rd Qu.:1.000   3rd Qu.:61.480   3rd Qu.:200.73   3rd Qu.:5586.1
 Max.   :1.000   Max.   :71.999   Max.   :290.16   Max.   :7159.0
 SatisfactionScore features_score
 Min.   :1.625     Min.   : 0.000
 1st Qu.:3.125     1st Qu.: 4.000
 Median :3.500     Median : 5.000
 Mean   :3.497     Mean   : 5.342
 3rd Qu.:3.875     3rd Qu.: 6.000
 Max.   :5.625     Max.   :11.000
```

## Next, I divided each element of each column by the max of that column to scale all values out of 1.

In [17]:
```
new_data2$features_score<-new_data2$features_score/11
new_data2$Tenure<-new_data2$Tenure/71.999
new_data2$MonthlyCharge<-new_data2$MonthlyCharge/290.16
new_data2$Bandwidth_GB_Year<-new_data2$Bandwidth_GB_Year/7159
new_data2$SatisfactionScore<-new_data2$SatisfactionScore/5.625
```

In [18]:
```
unstatisfaction_score<-1-new_data2$SatisfactionScore
new_data2$unstatisfaction_score<-unstatisfaction_score
nonfeatures_score<-1-new_data2$features_score
new_data2$nonfeatures_score<-nonfeatures_score
```

In [19]:
```
head(new_data2)
```

A tibble: 6 × 8

| Churn | Tenure | MonthlyCharge | Bandwidth_GB_Year | SatisfactionScore | features_score | unstatisfactior |
|---|---|---|---|---|---|---|
| <dbl> | <dbl> | <dbl> | <dbl> | <dbl> | <dbl> | |
| 0 | 0.09438344 | 0.5943463 | 0.12634951 | 0.7333333 | 0.6363636 | 0.2 |
| 1 | 0.01606524 | 0.8362026 | 0.11188473 | 0.6222222 | 0.6363636 | 0.3 |
| 0 | 0.21881060 | 0.5512393 | 0.28701033 | 0.6000000 | 0.4545455 | 0.4 |
| 0 | 0.23732589 | 0.4134162 | 0.30235779 | 0.6444444 | 0.3636364 | 0.3 |
| 1 | 0.02320826 | 0.5167780 | 0.03792337 | 0.7111111 | 0.2727273 | 0.2 |
| 0 | 0.09723737 | 0.6376058 | 0.14518201 | 0.5333333 | 0.6363636 | 0.4 |

## After the dataset was cleaned and prepared, I exported it to CSV following this guide: How to Import an Excel File into R (*Data to Fish*, 2020)

In [129…
```
write.csv(new_data2,"C:\\Users\\ContactTracer\\Desktop\\CleanedDataset.csv", row.names
```

In [20]:
```
NROW(new_data2)
```

10000

## Next, I found the measures of central tendency for the traget variable:

In [21]:
```
mean(new_data2$Churn)
```

0.265

## I created an initial model featuring all remaining predictor variables. This model features 10,000 observations and 5 variables. The target variable is the Churn. The mean of the Churn is 26.5. This shows that 26.5% of customers switched during the year. That is an alarmingly high number. The initial model will feature the following predictor variables, Tenure, MonthlyCharge, Bandwidth_GB_Year, Unstatisfaction_score, and nonfeatures_score. Finally, I selected the best predictor variables to create a reduced model.

# Part IV: Model Comparison and Analysis

## Model featuring all predictor variables

## I used the following source for the code to create the model: Logistic Regression Essentials in R. (kassambara , 2018)

In [22]:
```
glm.fit <- glm(Churn ~ Tenure+MonthlyCharge+Bandwidth_GB_Year+unstatisfaction_score+non
```

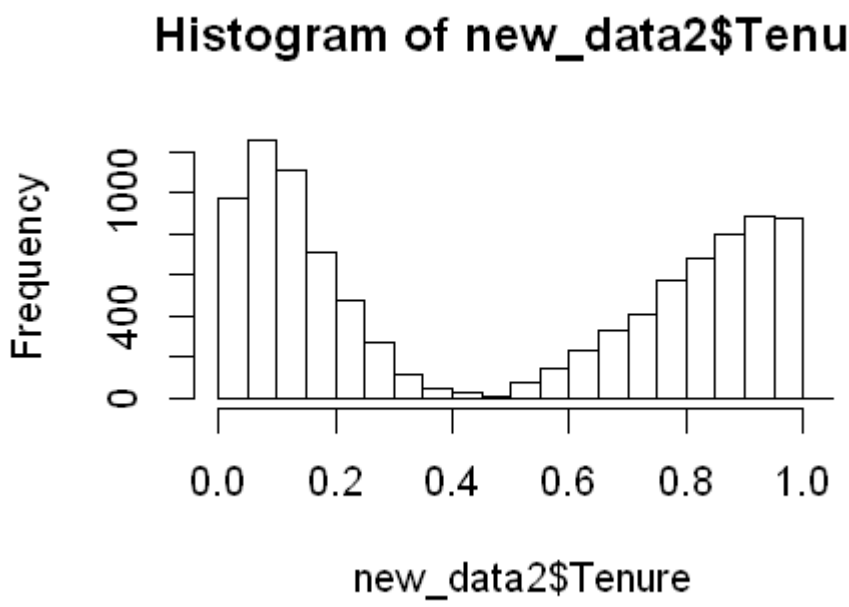
In [23]:
```
summary(glm.fit)
```

```
Call:
glm(formula = Churn ~ Tenure + MonthlyCharge + Bandwidth_GB_Year +
    unstatisfaction_score + nonfeatures_score, family = binomial,
    data = new_data2)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.7633  -0.5187  -0.1541   0.2724   3.3213

Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)            -6.3189     0.3230 -19.562  < 2e-16 ***
Tenure                -19.9335     0.8219 -24.253  < 2e-16 ***
MonthlyCharge           8.3895     0.3201  26.207  < 2e-16 ***
Bandwidth_GB_Year      17.2921     0.9407  18.383  < 2e-16 ***
unstatisfaction_score   0.5431     0.3244   1.674  0.09414 .
nonfeatures_score       0.7606     0.2869   2.652  0.00801 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 11564.4  on 9999  degrees of freedom
Residual deviance:  6475.4  on 9994  degrees of freedom
AIC: 6487.4

Number of Fisher Scoring iterations: 6
```

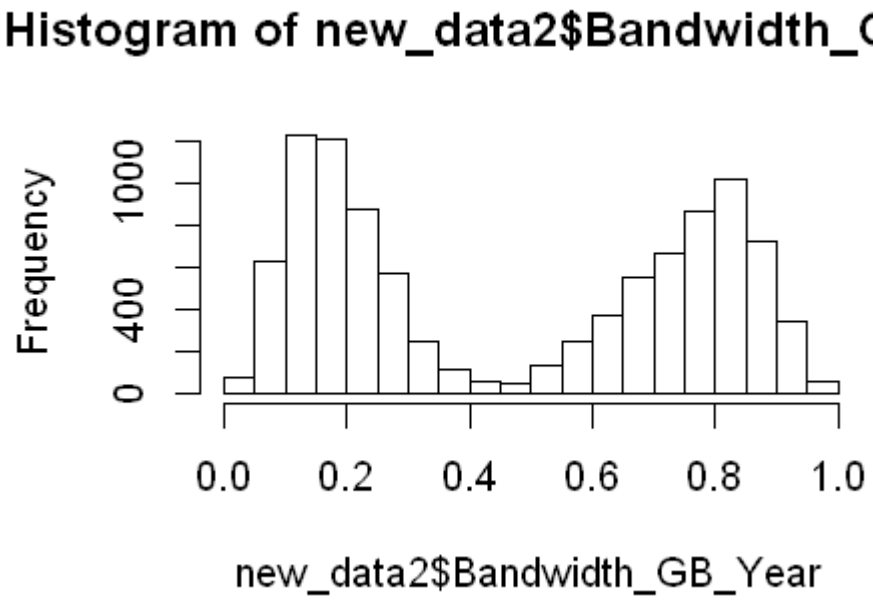
## The summary statistics show that Tenure and Bandwidth_GB_Year

are the most significant variables. Next, I checked both variables to make sure that they are good choices for the reduced model.

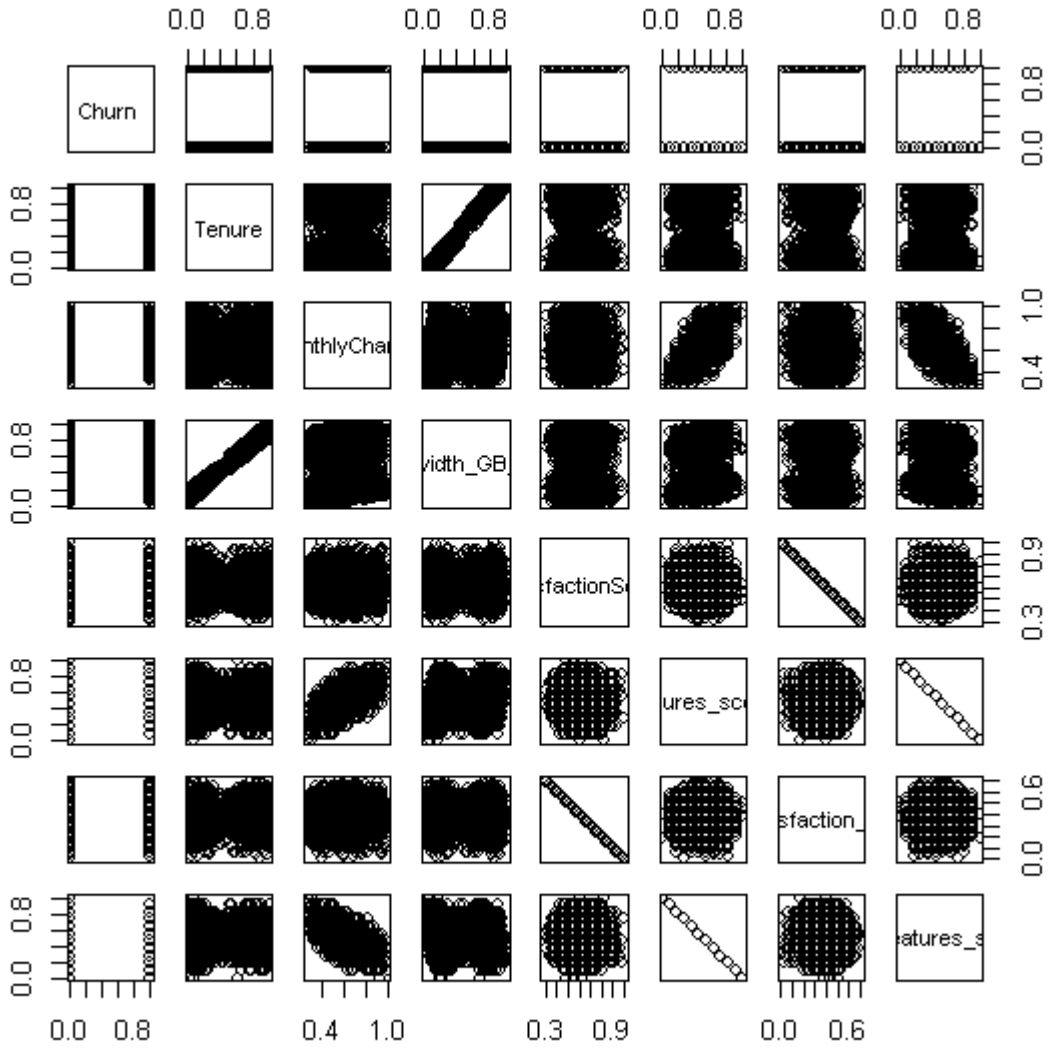
In [24]:
```
options(repr.plot.width=4, repr.plot.height=3)
hist(new_data2$Tenure)
```




In [25]:
```
options(repr.plot.width=4, repr.plot.height=3)
hist(new_data2$Bandwidth_GB_Year)
```




In [26]:
```
options(repr.plot.width=5, repr.plot.height=5)
pairs(new_data2)
```




In [27]:
```
cor(new_data2[,])
```

A matrix: 8 × 8 of type dbl

| | Churn | Tenure | MonthlyCharge | Bandwidth_GB_Year | SatisfactionScore |
|---|---|---|---|---|---|
| **Churn** | 1.00000000 | -0.485475027 | 0.3729378909 | -0.441668690 | -0.0121609834 |
| **Tenure** | -0.48547503 | 1.000000000 | -0.0033368104 | 0.991495192 | -0.0021700910 |
| **MonthlyCharge** | 0.37293789 | -0.003336810 | 1.0000000000 | 0.060406431 | -0.0005683693 |
| **Bandwidth_GB_Year** | -0.44166869 | 0.991495192 | 0.0604064308 | 1.000000000 | -0.0029826166 |
| **SatisfactionScore** | -0.01216098 | -0.002170091 | -0.0005683693 | -0.002982617 | 1.0000000000 |
| **features_score** | 0.23935312 | -0.001936955 | 0.6473223665 | 0.053469267 | 0.0098614105 |
| **unstatisfaction_score** | 0.01216098 | 0.002170091 | 0.0005683693 | 0.002982617 | -1.0000000000 |
| **nonfeatures_score** | -0.23935312 | 0.001936955 | -0.6473223665 | -0.053469267 | -0.0098614105 |

## The cor function in R shows the variables with the largest absolute value of correlation are Bandwidth_GB_Year and Tenure. I used these two variables in my reduced model.

## Creating the reduced model with only two predictors

### I used Tenure and Bandwidth_GB_Year for the reduced model due to the significance of both variables having the largest absolute value.

In [28]:
```
new_data3<-dplyr::select(new_data2, -c(3,5,6,7,8))
```

## This is the final cleaned data set used for the reduced model.

In [29]:
```
head(new_data3)
```

A tibble: 6 × 3

| Churn | Tenure | Bandwidth_GB_Year |
|---|---|---|
| <dbl> | <dbl> | <dbl> |
| 0 | 0.09438344 | 0.12634951 |
| 1 | 0.01606524 | 0.11188473 |
| 0 | 0.21881060 | 0.28701033 |
| 0 | 0.23732589 | 0.30235779 |
| 1 | 0.02320826 | 0.03792337 |
| 0 | 0.09723737 | 0.14518201 |

## Next, I created the code for the reduced model.

In [30]:
```
glm.fit1 <- glm(Churn ~ Tenure+Bandwidth_GB_Year, data = new_data3, family = binomial)
```

In [31]:
```
summary(glm.fit1)
```

```
Call:
glm(formula = Churn ~ Tenure + Bandwidth_GB_Year, family = binomial,
    data = new_data3)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2918  -0.6005  -0.2299   0.4901   3.1735

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)        -1.62463    0.07101  -22.88   <2e-16 ***
Tenure            -26.83513    0.73292  -36.61   <2e-16 ***
Bandwidth_GB_Year  26.80820    0.82711   32.41   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 11564.4  on 9999  degrees of freedom
Residual deviance:  7553.1  on 9997  degrees of freedom
AIC: 7559.1
```

```
Number of Fisher Scoring iterations: 6
```

In [32]:
```
confint(glm.fit1)
```

```
Waiting for profiling to be done...
```

A matrix: 3 × 2 of type dbl

| | 2.5 % | 97.5 % |
|---|---|---|
| **(Intercept)** | -1.764618 | -1.486237 |
| **Tenure** | -28.285492 | -25.412143 |
| **Bandwidth_GB_Year** | 25.200229 | 28.442832 |

## The confidence interval shows that both values are within the 95% confidence interval so we can feel good about our choice of predictor variables.

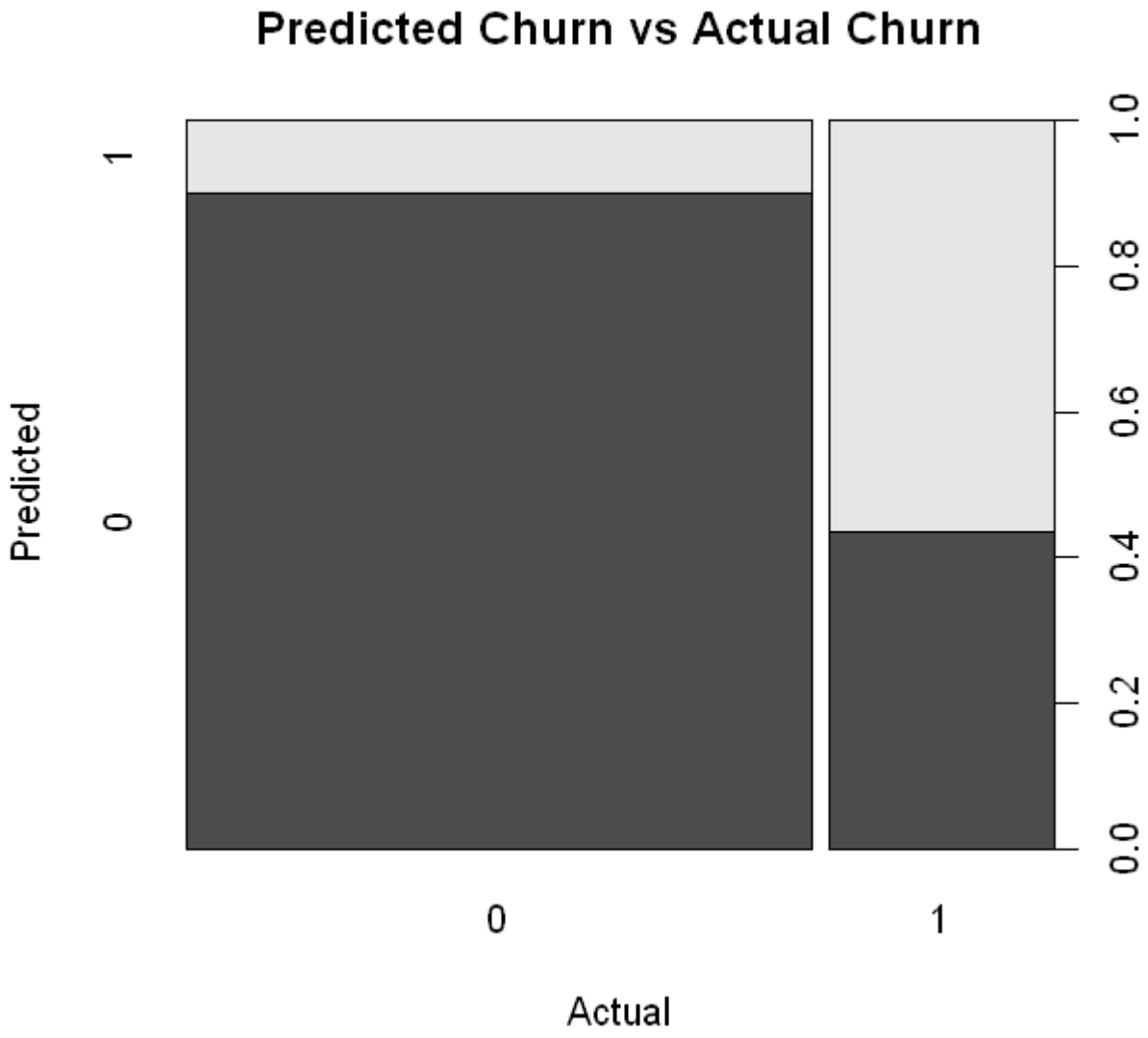
## Next, I created a confusion matrix to assess how accurate the reduced model is in predicting if a customer will switch providers. What is a Confusion Matrix in Machine Learning. Machine Learning Mastery. (Brownlee, 2020)

In [33]:
```
glm.fit1Pred <- round(predict(glm.fit1, new_data3, type="response"))
```

In [34]:
```
predictedChurn <- factor(glm.fit1Pred)
```

In [35]:
```
actualChurn <- factor(new_data3$Churn)
```

In [36]:
```
plot(actualChurn, predictedChurn, main="Predicted Churn vs Actual Churn",
    xlab="Actual",ylab="Predicted")
```




In [193…
```
glm.fit1CM <- confusionMatrix(data = predictedChurn, reference = actualChurn)
```

In [194…
```
glm.fit1CM
```

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 6608 1153
         1  742 1497

               Accuracy : 0.8105
                 95% CI : (0.8027, 0.8181)
    No Information Rate : 0.735
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.4882

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.8990
            Specificity : 0.5649
         Pos Pred Value : 0.8514
```

```
         Neg Pred Value : 0.6686
             Prevalence : 0.7350
         Detection Rate : 0.6608
   Detection Prevalence : 0.7761
      Balanced Accuracy : 0.7320

       'Positive' Class : 0
```

### The reduced model is over 81% accurate in predicting if a customer will switch providers.

### Comparing the initial and reduced models

### Initial model:

### Coefficients:

### Estimate Std. Error z value Pr(>|z|)

### (Intercept) -6.3189 0.3230 -19.562 < 2e-16 ***

### Tenure -19.9335 0.8219 -24.253 < 2e-16 ***

### MonthlyCharge 8.3895 0.3201 26.207 < 2e-16 ***

### Bandwidth_GB_Year 17.2921 0.9407 18.383 < 2e-16 ***

### unstatisfaction_score 0.5431 0.3244 1.674 0.09414 .

### nonfeatures_score 0.7606 0.2869 2.652 0.00801 **

### Reduced model:

### Coefficients:

### Estimate Std. Error z value Pr(>|z|)

### (Intercept) -1.62463 0.07101 -22.88 <2e-16 ***

### Tenure -26.83513 0.73292 -36.61 <2e-16 ***

### Bandwidth_GB_Year 26.80820 0.82711 32.41 <2e-16 ***

### Comparing the models shows that the coefficents with the highest absolute value of the significance are much greater in the reduced model.

# Residual Plot

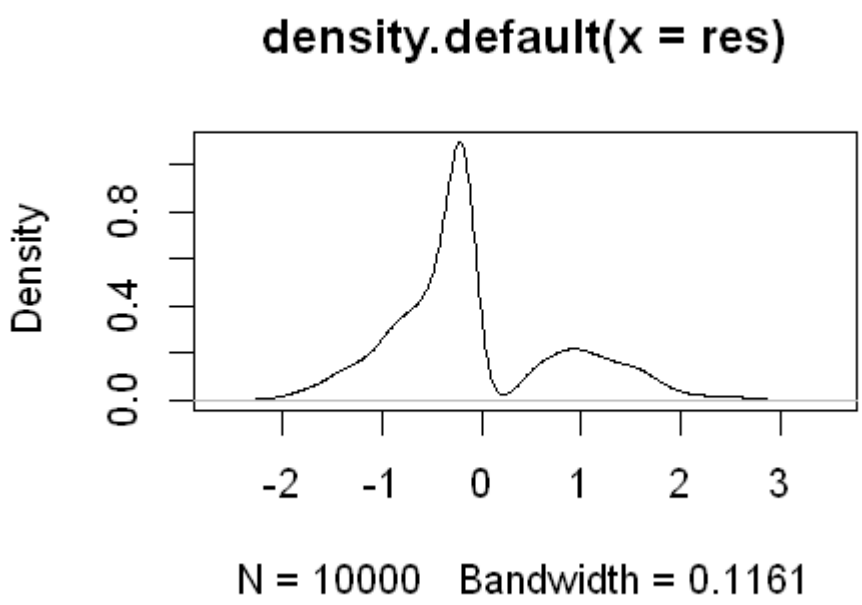
In [58]:
```
res <- resid(glm.fit1)
```

In [3]:
```
options(repr.plot.width=5, repr.plot.height=4)
plot(fitted(glm.fit), res)
```

```
abline(0,0)
```

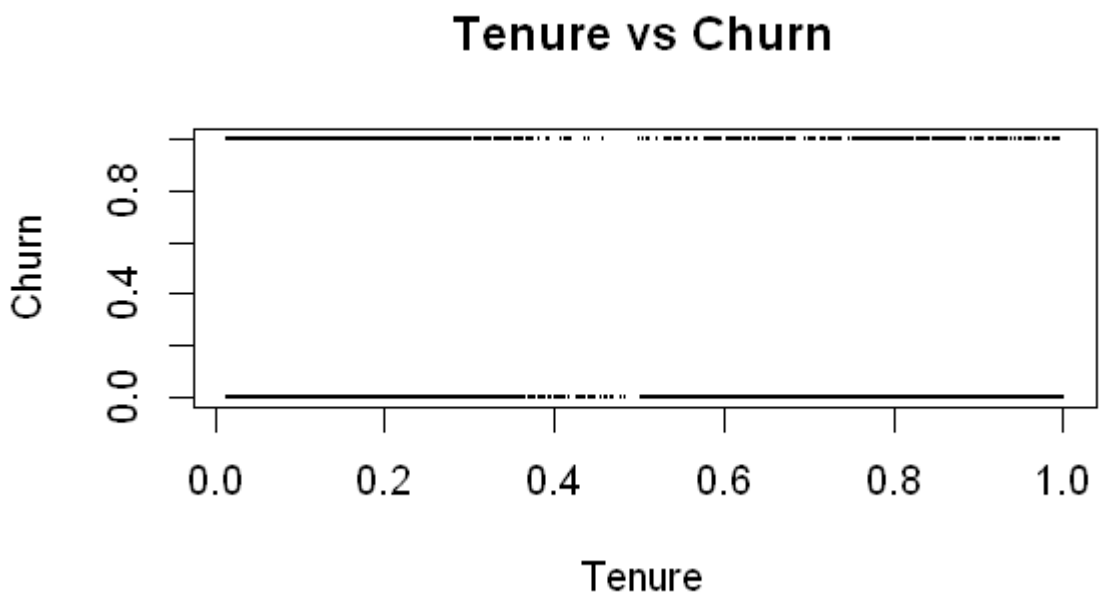
```
Error: object of type 'closure' is not subsettable
Traceback:

1. plot(fitted(glm.fit), res)
2. fitted(glm.fit)
3. fitted.default(glm.fit)
```

In [60]:
```
options(repr.plot.width=4, repr.plot.height=3)
plot(density(res))
```



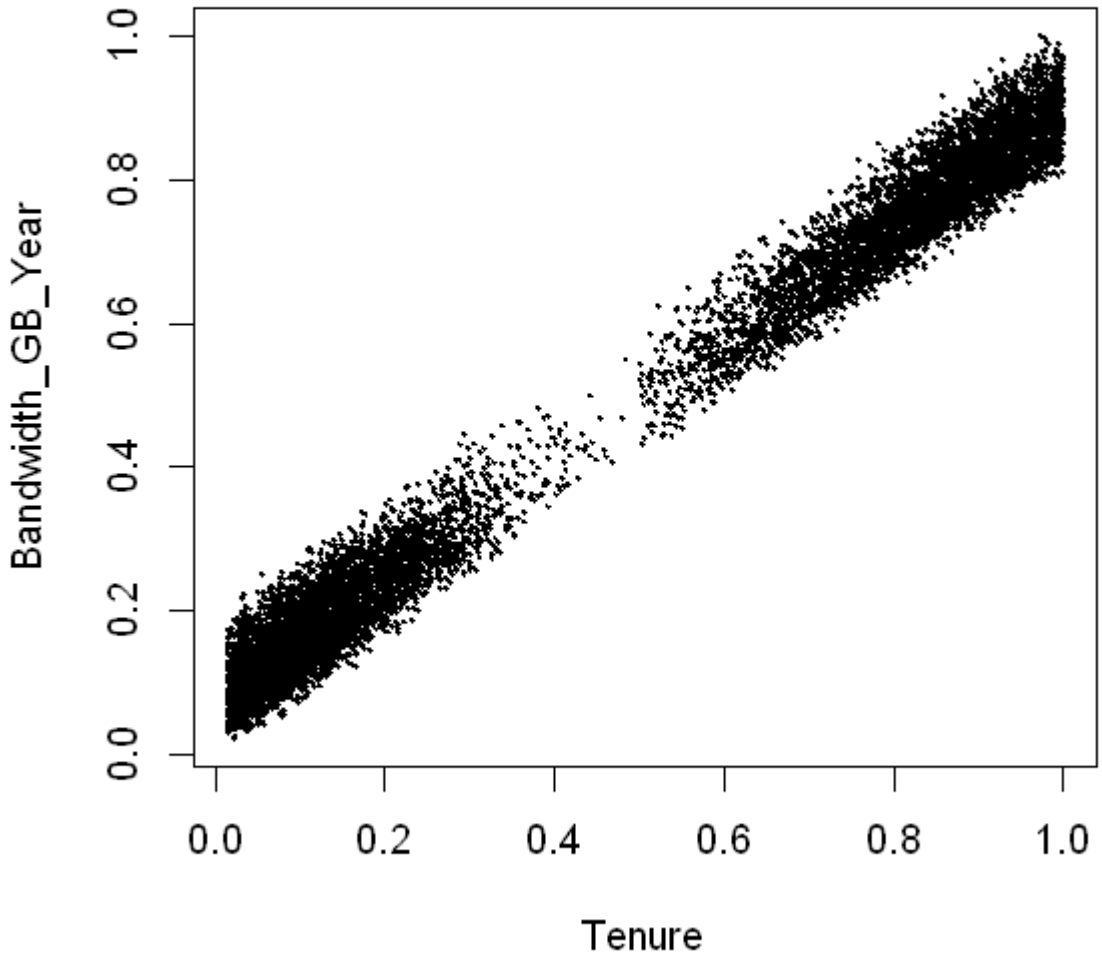

## The density of the residuals is close to normally distributed, which implies a good choice of model.

In [128…
```
options(repr.plot.width=5, repr.plot.height=3)
plot(new_data3$Churn~new_data3$Tenure, main="Tenure vs Churn", xlab="Tenure", ylab="Chu
```




**The distibution of Churn vs Tenure shows that as Tenure increases Churn decreases. The points are more densly packed to the left and less densely packed to the right. This fits exactly with what was seen in the initial and reduced models.**

In [75]:
```
plot(new_data3[,2:3], pch=10, cex=0.2)
```




Here, the relationship between Bandwidth_GB_Year and Tenure is shown. The longer someone is with a provider the more data they use per year.

# Part V: Data Summary and Implications

The equation for the reduced model is: churn = -1.6 - 26.8*Tenure* + *26.8* Bandwidth_GB_Year

The coefficents show that a customer is less likely to switch providers the longer they are with one company, and more likely to switch the higher their data usage is.

Interpreting the data shows the longer someone is with one provider, the more likely they are to stay. People become set in their ways and they do not want to change things once they become accustomed to something. In addition, people who use a large amount of data are more likely to switch providers, as they are

more likely to get large overage fees, thus making them upset with their provider.

According to the model fit, the significance of the coefficents and the intercept are all three stars, which is the highest possible significance score. This shows the model is very significant.

# Limitations of the data:

This data set covers 10000 customers and we have no way to know how representative this set is compared to the full set of all customers. What was the selection criteria? We also have no time information included with this dataset. Was this data collected over one month, one year? Losing 25% of your customers every year is very different than losing 25% every month.

## Course of Action

I would recommend the following course of action based on the results of the created model. First, I would recommend an increase in customer loyalty programs. This would help keep customers with their provider longer and would have a positive effect on decreasing customer churn. Secondly, I would recommend adding an unlimited data tier. Customers who go over their data limit are far more likely to switch, so offering an unlimited tier would be one way to combat that. Another way would be to offer customers a grace period once per year where there would be no overage charge the first time a customer exceeds their data limit.

# References:


Stoltzfus, J. (2011, October 13). Logistic Regression: A Brief Primer. Retrieved December 04, 2020, from https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1553-2712.2011.01185.x

Halilovic, I. (2019, May 01). Markdown for Jupyter notebooks cheatsheet. Retrieved December 04, 2020, from https://medium.com/@ingeh/markdown-for-jupyter-notebooks-cheatsheet-386c05aeebed

Kassambara, Leo, Sara, Mbugua, F. (2018, March 10). Multiple Linear Regression in R. Retrieved December 04, 2020, from http://www.sthda.com/english/articles/40-regression-analysis/168-multiple-linear-regression-in-r/

de Vries, Andres Resizing plots in the R kernel for Jupyter notebooks. (n.d.). Retrieved December 04, 2020, from https://blog.revolutionanalytics.com/2015/09/resizing-plots-in-the-r-kernel-for-jupyter-notebooks.html

kassambara, Leo, Logistic Regression Essentials in R. (2018, March 11). Articles - STHDA. http://www.sthda.com/english/articles/36-classification-methods-essentials/151-logistic-regression-essentials-in-r/

How to Export DataFrame to CSV in R. Data to Fish. (2020, February 8) https://datatofish.com/export-dataframe-to-csv-in-r/

Brownlee, J. What is a Confusion Matrix in Machine Learning. Machine Learning Mastery. (2020, August 15) https://machinelearningmastery.com/confusion-matrix-machine-learning/