D208 Predictive Modeling - Task 2

Scott Babcock WGU - MS, Data Analytics

Created: December 22 2024

Load libraries

```
library(dplyr)
library(naniar)
library(fastDummies)
library(ggplot2)
library(broom)
library(outliers)
library(car)
library(corrplot)
library(yardstick)
```

Turn off Scientific Notation

```
options(scipen = 999)
```

Load data set

```
churn_df <- read.csv("churn_clean.csv")
```

A1, Research Question

What factors can predict churn for a customer?

A2, Goals of Analysis

The analysis aims to predict what factors lead to customer churn. The analysis will provide valuable insight into what customers are at risk of discontinuing service. The company can take preemptive measures to ensure the business is maintained.

B1, Assumptions of Logistic Regression Model

The central assumption of logistic regression is that the response variable must have a binary outcome. A binary outcome would have two outcomes: yes/no, 1/0, male/female, and so on. There also must be no multicollinearity amongst the explanatory variables. Multicollinearity occurs when two or more independent variables are highly correlated, which can lead to overfitting the model (Assumptions, 2024). A good way to test for multicollinearity is by calculating VIF (Variable Inflation Factor). A VIF greater than ten would indicate that multicollinearity is present (D208 Webinar, n.d.). The observations should also be independent. The outcome of one observation should not influence the outcome of another. Finally, there must be a large sample size. The sample size should be large enough to ensure reliable inference (Bobbitt, October 13, 2020).

B2, Benefits of Programming Language

The R programming language has a multitude of benefits. It is specifically geared towards statistical analysis, making many analysis phases easier and more efficient. It has a wide array of packages that allow users to accomplish tasks in a single step instead of lengthy coding. One benefit of this analysis was using the 'fastDummies' library for variable re-expression. The library provided the ability to create dummy columns for all specified variables and drop the first category in a single step. Also, creating logistic models is simple and intuitive with base R. With minimal coding, a user can run logistic regression on a target variable against all explanatory variables.

B3, Logistic Regression Justification

Multiple Logistic Regression is an appropriate technique to identify factors that can predict customer churn. The dependent variable must have a binary outcome in logistic regression; in this case, the Churn variable has a binary Yes/No outcome. Running logistic regression on multiple variables helps identify which predictor variables are significant. Through the feature selection process, the variables can be reduced to only meaningful ones when predicting the dependent variable outcome. It is a straightforward way to make predictions.

C1, Data Cleaning Goals & Steps

One should follow specific steps to ensure that data is clean and ready for analysis. First, the data was checked for duplicate records, and none were present. The data was then checked to ensure there were no missing values. No missing values were found, but had there been, values would have been imputed using measures of central tendency such as median. Finally, each variable chosen for the initial model was checked to see if outliers were present. Outliers were identified in five of the numeric variables but ultimately deemed reasonable. Unique values for each categorical variable were viewed to determine how they should be re-expressed. Given that all the categorical variables had a small number of unique values, one hot encoding was chosen for re-expression. The dependent variable, churn, was converted from Yes/No to 1/0 to run a correlation matrix. After these steps were performed, the data was clean and in a good state for analysis.

Check for Duplicates/Missing Values

```
# check for duplicate records [In-text citation:(Getting Started with Duplicates, n.d.)]
sum(duplicated(churn_df))
```

```
## [1] 0
```

```
# check for missing values [In-text citation: (Tierney, n.d.)]
miss_var_summary(churn_df)
```

```
## # A tibble: 50 x 3
##    variable    n_miss pct_miss
##    <chr>        <int>    <num>
##  1 CaseOrder        0        0
##  2 Customer_id      0        0
##  3 Interaction      0        0
##  4 UID              0        0
##  5 City             0        0
##  6 State            0        0
##  7 County           0        0
##  8 Zip              0        0
```

```
##  9 Lat                0        0
## 10 Lng                0        0
## # i 40 more rows
```

Create Analysis Data Frame

```
# create subset for model variables
churn_analysis <-
  churn_df %>%
  select(Churn,
         Age,
         Area,
         Marital,
         Gender,
         Income,
         Outage_sec_perweek,
         Email,
         Contacts,
         Yearly_equip_failure,
         Contract,
         MonthlyCharge,
         Bandwidth_GB_Year)

# view data structure
glimpse(churn_analysis)
```

```
## Rows: 10,000
## Columns: 13
## $ Churn                <chr> "No", "Yes", "No", "No", "Yes", "No", "Yes", "Yes~
## $ Age                  <int> 68, 27, 50, 48, 83, 83, 79, 30, 49, 86, 23, 56, 8~
## $ Area                 <chr> "Urban", "Urban", "Urban", "Suburban", "Suburban"~
## $ Marital              <chr> "Widowed", "Married", "Widowed", "Married", "Sepa~
## $ Gender               <chr> "Male", "Female", "Female", "Male", "Male", "Fema~
## $ Income               <dbl> 28561.99, 21704.77, 9609.57, 18925.23, 40074.19, ~
## $ Outage_sec_perweek   <dbl> 7.978323, 11.699080, 10.752800, 14.913540, 8.1474~
## $ Email                <int> 10, 12, 9, 15, 16, 15, 10, 16, 20, 18, 9, 17, 9, ~
## $ Contacts             <int> 0, 0, 0, 2, 2, 3, 0, 0, 2, 1, 0, 1, 0, 1, 3, 1, 1~
## $ Yearly_equip_failure <int> 1, 1, 1, 0, 1, 1, 1, 0, 3, 0, 2, 1, 0, 0, 0, 0, 0~
## $ Contract             <chr> "One year", "Month-to-month", "Two Year", "Two Ye~
## $ MonthlyCharge        <dbl> 172.45552, 242.63255, 159.94758, 119.95684, 149.9~
## $ Bandwidth_GB_Year    <dbl> 904.5361, 800.9828, 2054.7070, 2164.5794, 271.493~
```

View Unique Values for Categorical Variables

```
unique(churn_analysis$Churn)
```

```
## [1] "No"  "Yes"
```

```
unique(churn_analysis$Area)
```

```
## [1] "Urban"    "Suburban" "Rural"
```

```r
unique(churn_analysis$Marital)
```

```
## [1] "Widowed"       "Married"       "Separated"     "Never Married"
## [5] "Divorced"
```

```r
unique(churn_analysis$Gender)
```

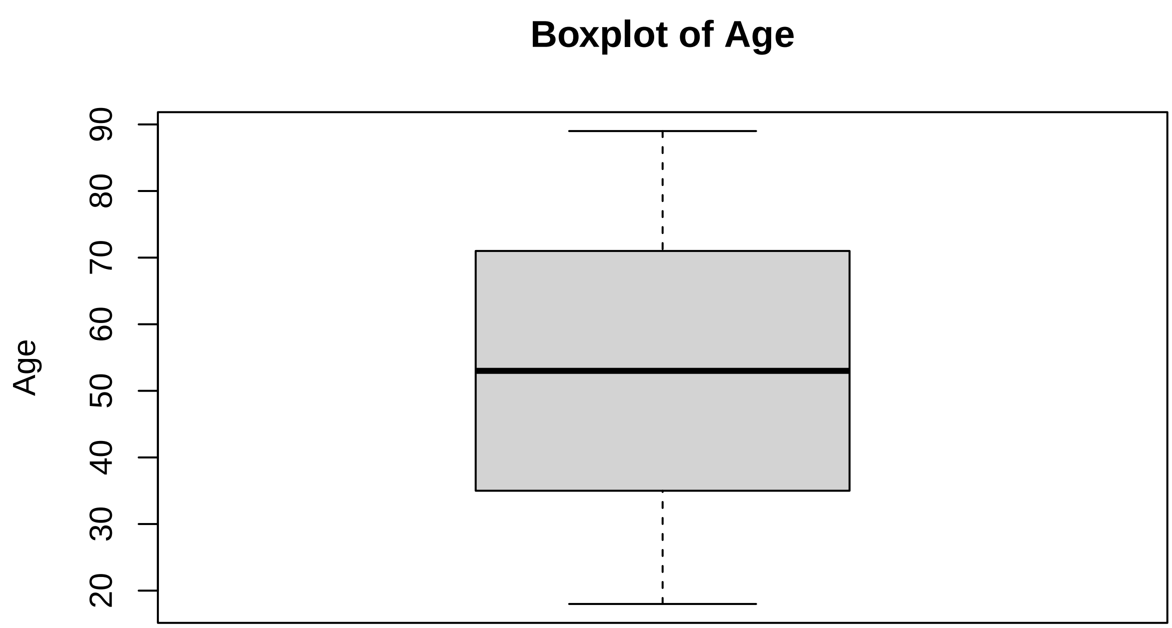
```
## [1] "Male"      "Female"    "Nonbinary"
```

```r
unique(churn_analysis$Contract)
```

```
## [1] "One year"       "Month-to-month" "Two Year"
```

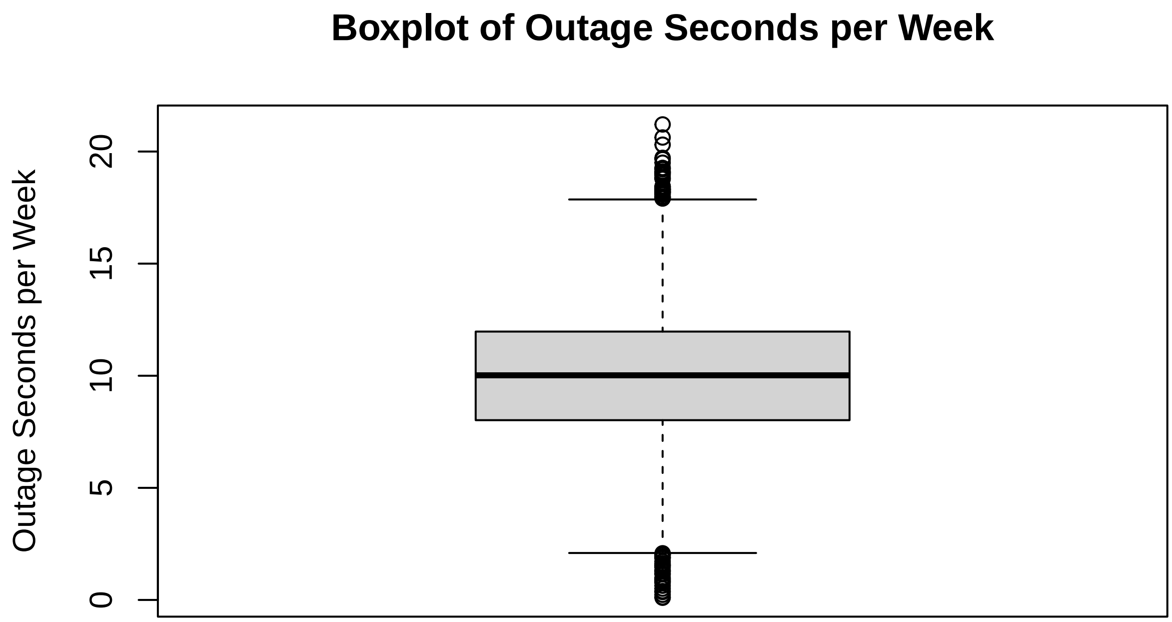
Check for Outliers

```r
# check for outliers in numeric variables
boxplot(churn_analysis$Age,
        ylab = "Age",
        main = "Boxplot of Age")
```

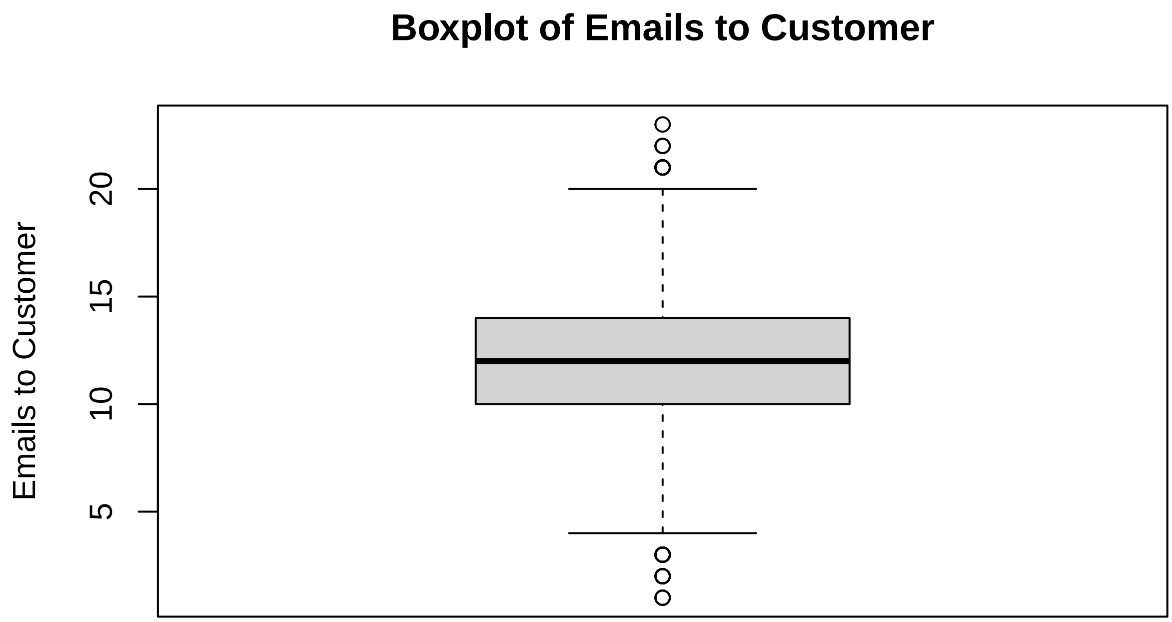



```r
boxplot(churn_analysis$Income,
        ylab = "Income",
        main = "Boxplot of Income")
```

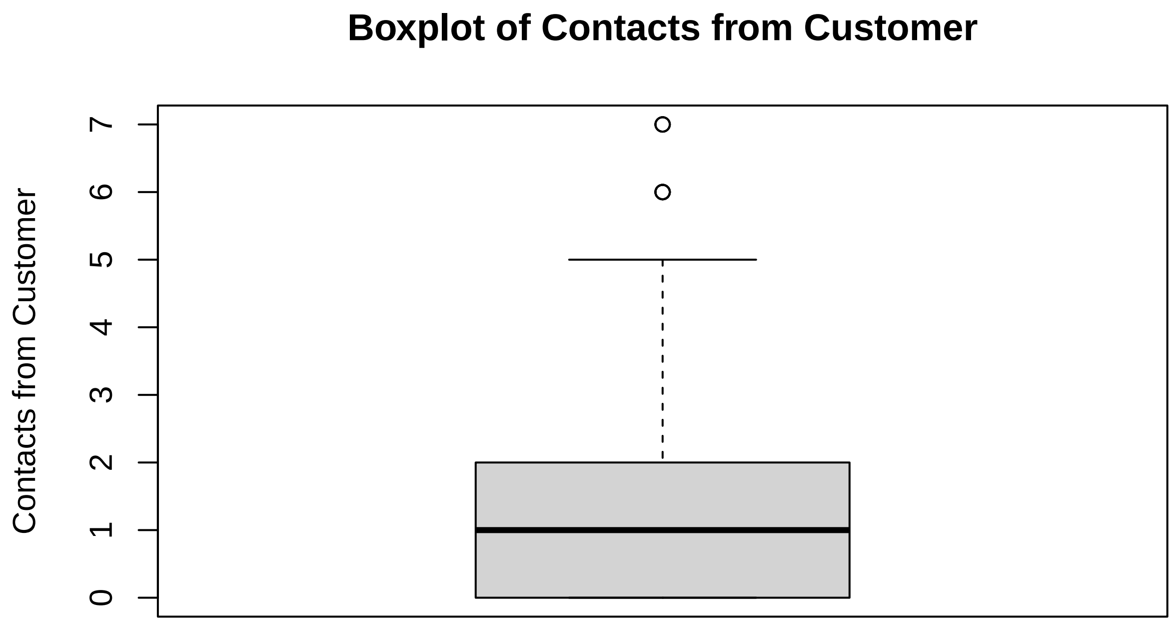

```r
boxplot(churn_analysis$Outage_sec_perweek,
        ylab = "Outage Seconds per Week",
        main = "Boxplot of Outage Seconds per Week")
```



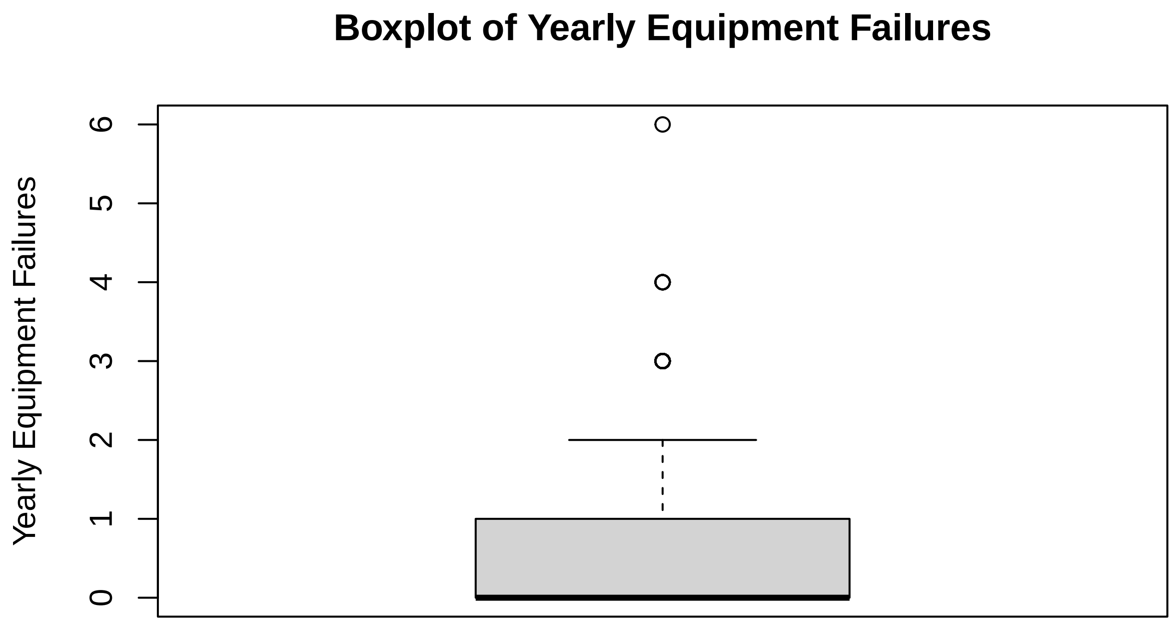

```r
boxplot(churn_analysis$Email,
        ylab = "Emails to Customer",
        main = "Boxplot of Emails to Customer")
```



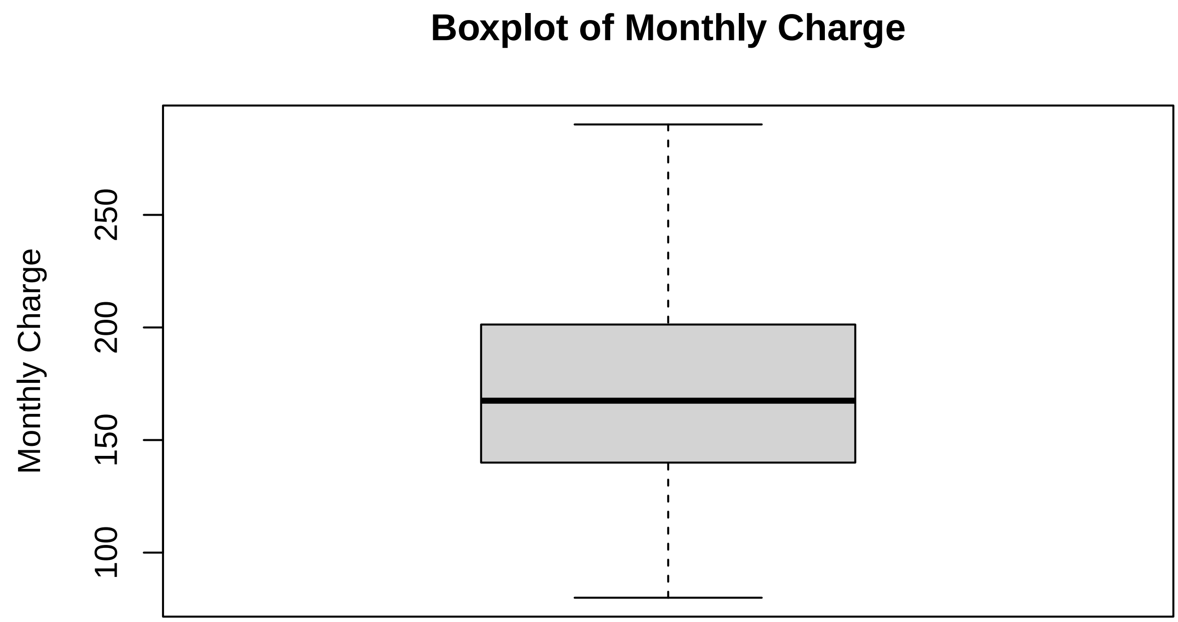

```
boxplot(churn_analysis$Contacts,
        ylab = "Contacts from Customer",
        main = "Boxplot of Contacts from Customer")
```



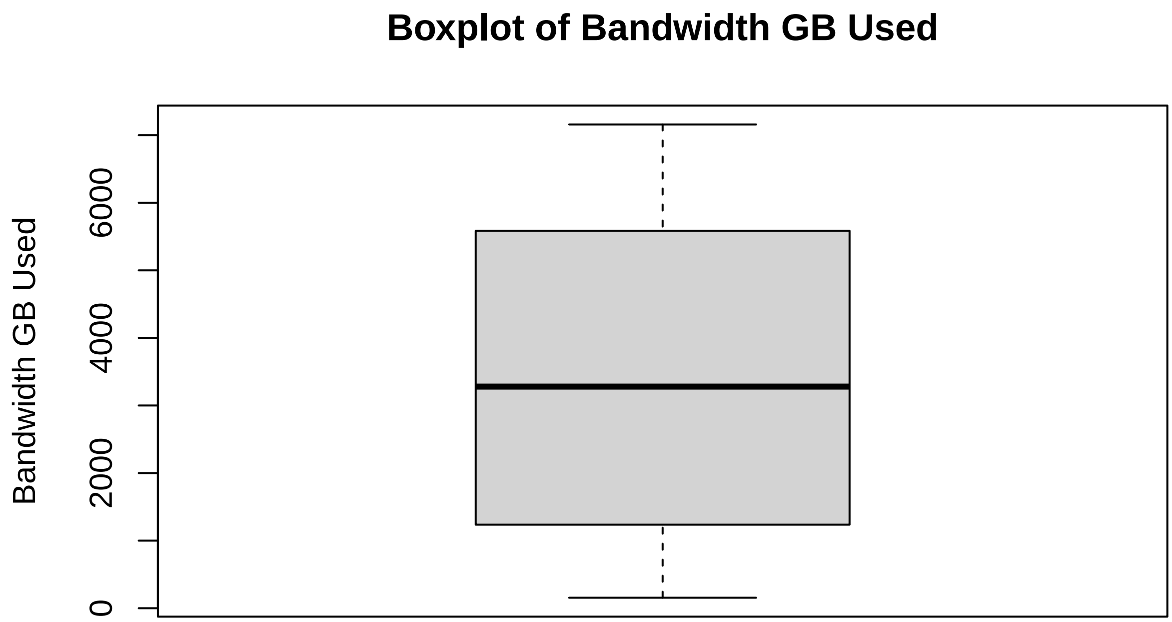

```r
boxplot(churn_analysis$Yearly_equip_failure,
        ylab = "Yearly Equipment Failures",
        main = "Boxplot of Yearly Equipment Failures")
```




```
boxplot(churn_analysis$MonthlyCharge,
        ylab = "Monthly Charge",
        main = "Boxplot of Monthly Charge")
```




```
boxplot(churn_analysis$Bandwidth_GB_Year,
        ylab = "Bandwidth GB Used",
        main = "Boxplot of Bandwidth GB Used")
```



C2, Dependent and Independent Variables

For the initial analysis, 13 variables were chosen from the Churn dataset. The dependent variable, or the response variable, was the Churn variable. Churn is a Yes/No categorical variable that indicates whether the customer has discontinued service within the last month. The analysis will be attempting to predict what leads to a customer churning.

The remaining variables were the independent variables or explanatory variables. Age is numeric and represents the customer's age when they signed up. Area is a categorical variable classified by where the customer lives based on census data. Marital is a categorical variable and represents the marital status of the customer. Gender is a categorical variable that indicates whether the customer identifies as male, female, or non-binary. Income is a self-reported numeric variable that is the customer's annual income. Outage_sec_perweek is a numeric variable representing the average number of seconds per week the customer's neighborhood experienced service outages. The Email variable is numeric and represents the number of emails sent to the customer within the last year. The Contacts variable is numeric and represents the number of times the customer contacted technical support. The Yearly_equip_failure variable is numeric and represents how many times the customer's equipment failed within the last year. The Contract variable is categorical and indicates the type of contract the customer has: month-to-month, one-year, or two-year. The MonthlyCharge variable is numeric and is the average monthly charge for a customer. The Bandwidth_GB_Year variable is numeric and represents how much data the customer has used within the last year.

Quantitative Variables

```r
summary(churn_analysis$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   18.00   35.00   53.00   53.08   71.00   89.00
```

```r
summary(churn_analysis$Income)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##    348.7  19224.7  33170.6  39806.9  53246.2 258900.7
```

```r
summary(churn_analysis$Outage_sec_perweek)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##  0.09975  8.01821 10.01856 10.00185 11.96949 21.20723
```

```r
summary(churn_analysis$Email)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   10.00   12.00   12.02   14.00   23.00
```

```r
summary(churn_analysis$Contacts)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0000  1.0000  0.9942  2.0000  7.0000
```

```r
summary(churn_analysis$Yearly_equip_failure)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   0.398   1.000   6.000
```

```r
summary(churn_analysis$MonthlyCharge)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   79.98  139.98  167.48  172.62  200.73  290.16
```

```r
summary(churn_analysis$Bandwidth_GB_Year)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   155.5  1236.5  3279.5  3392.3  5586.1  7159.0
```

Categorical Variables

```r
table(churn_analysis$Churn)
```

```
##
##   No  Yes
## 7350 2650
```

```r
table(churn_analysis$Area)
```

```
##
##    Rural Suburban    Urban
##     3327     3346     3327
```

```r
table(churn_analysis$Marital)
```

```
##
##      Divorced       Married Never Married     Separated       Widowed
##          2092          1911          1956          2014          2027
```

```r
table(churn_analysis$Gender)
```

```
##
##    Female      Male Nonbinary
##      5025      4744       231
```

```r
table(churn_analysis$Contract)
```

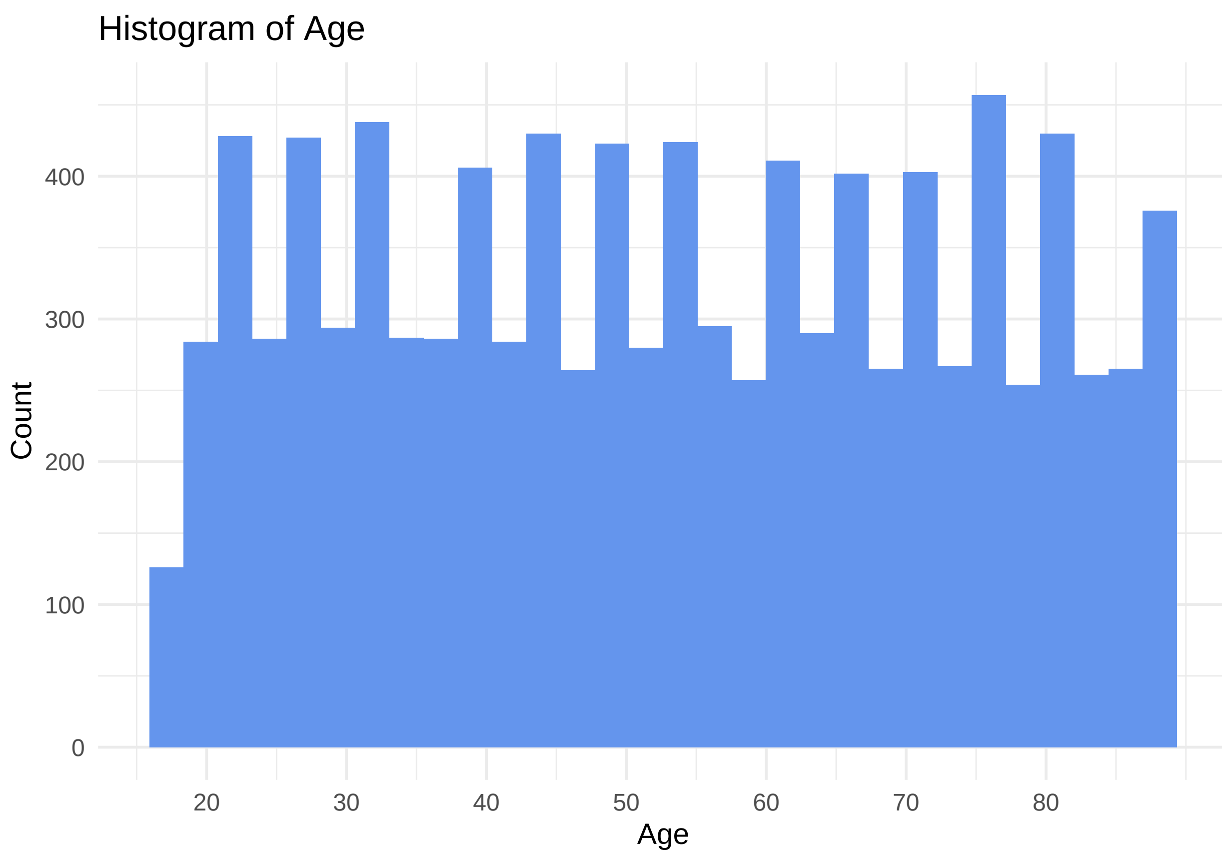
```
##
## Month-to-month       One year       Two Year
##           5456           2102           2442
```

C3, Univariate and Bivariate Visualizations

Univariate and Bivariate visualizations were generated for each explanatory variable and can be found below.

Univariate Visualizations

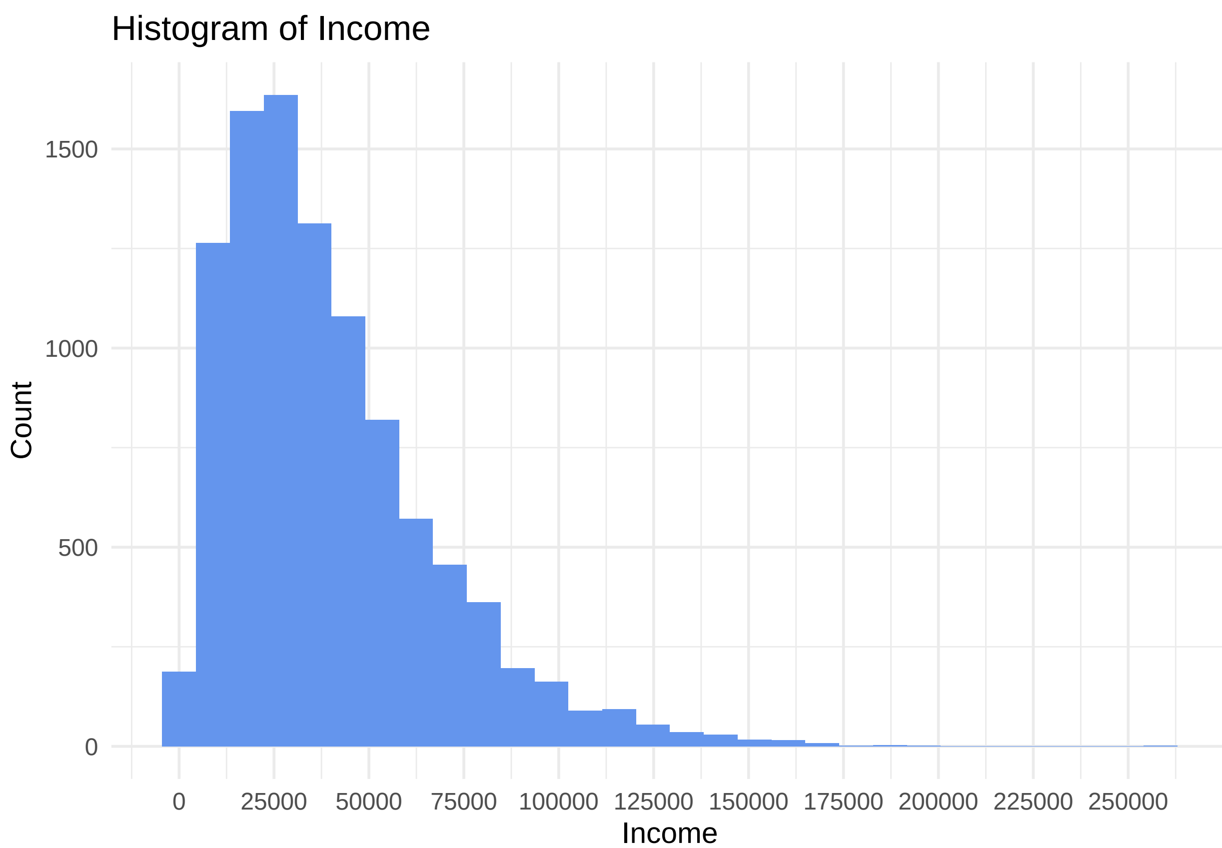
```r
# Quantitative Variables
ggplot(churn_analysis, aes(x = Age))+
  geom_histogram(fill = "cornflowerblue")+
  scale_x_continuous(breaks = seq(0,max(churn_analysis$Age),10))+
  labs(title = "Histogram of Age", y= "Count")+
  theme_minimal()
```




```
summary(churn_analysis$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   18.00   35.00   53.00   53.08   71.00   89.00
```

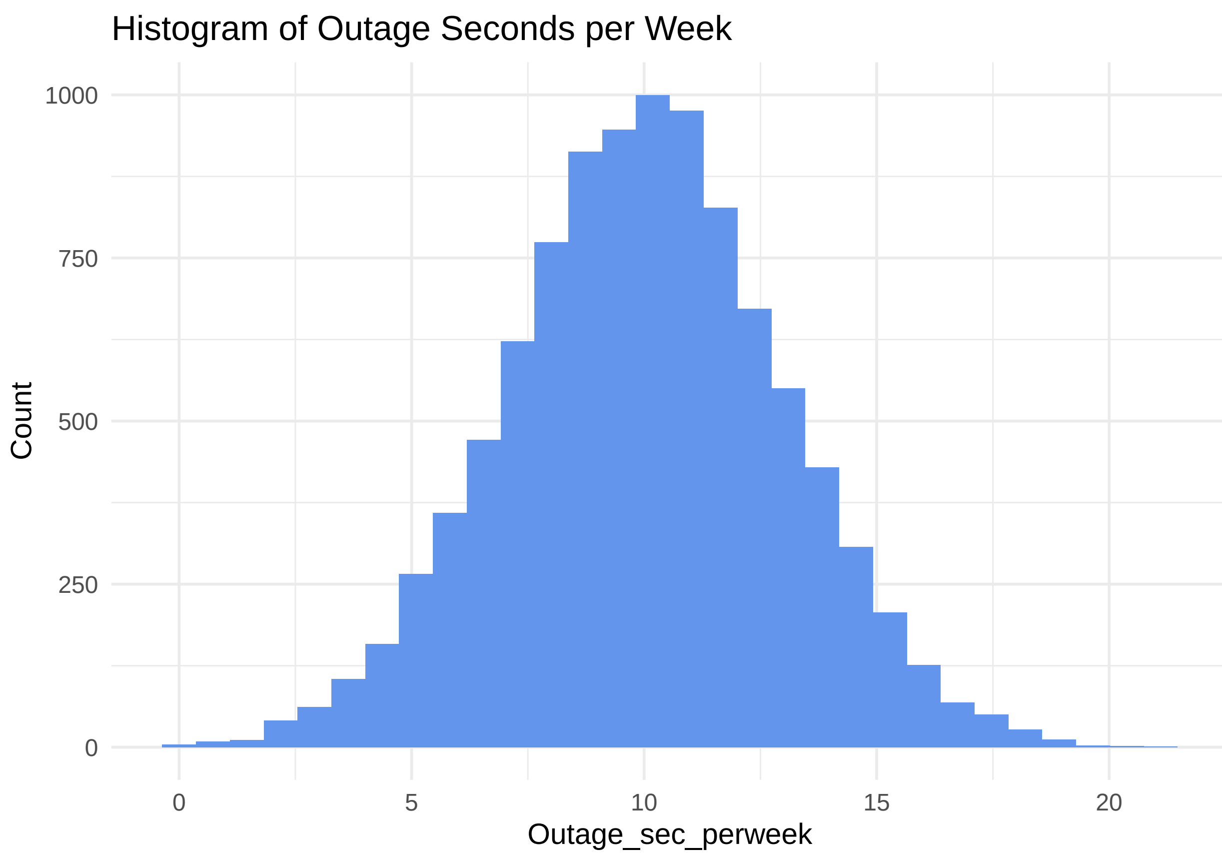
```
ggplot(churn_analysis, aes(x = Income))+
  geom_histogram(fill = "cornflowerblue")+
  scale_x_continuous(breaks = seq(0,max(churn_analysis$Income),25000))+
  labs(title = "Histogram of Income", y= "Count")+
  theme_minimal()
```




```
summary(churn_analysis$Income)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##    348.7  19224.7  33170.6  39806.9  53246.2 258900.7
```

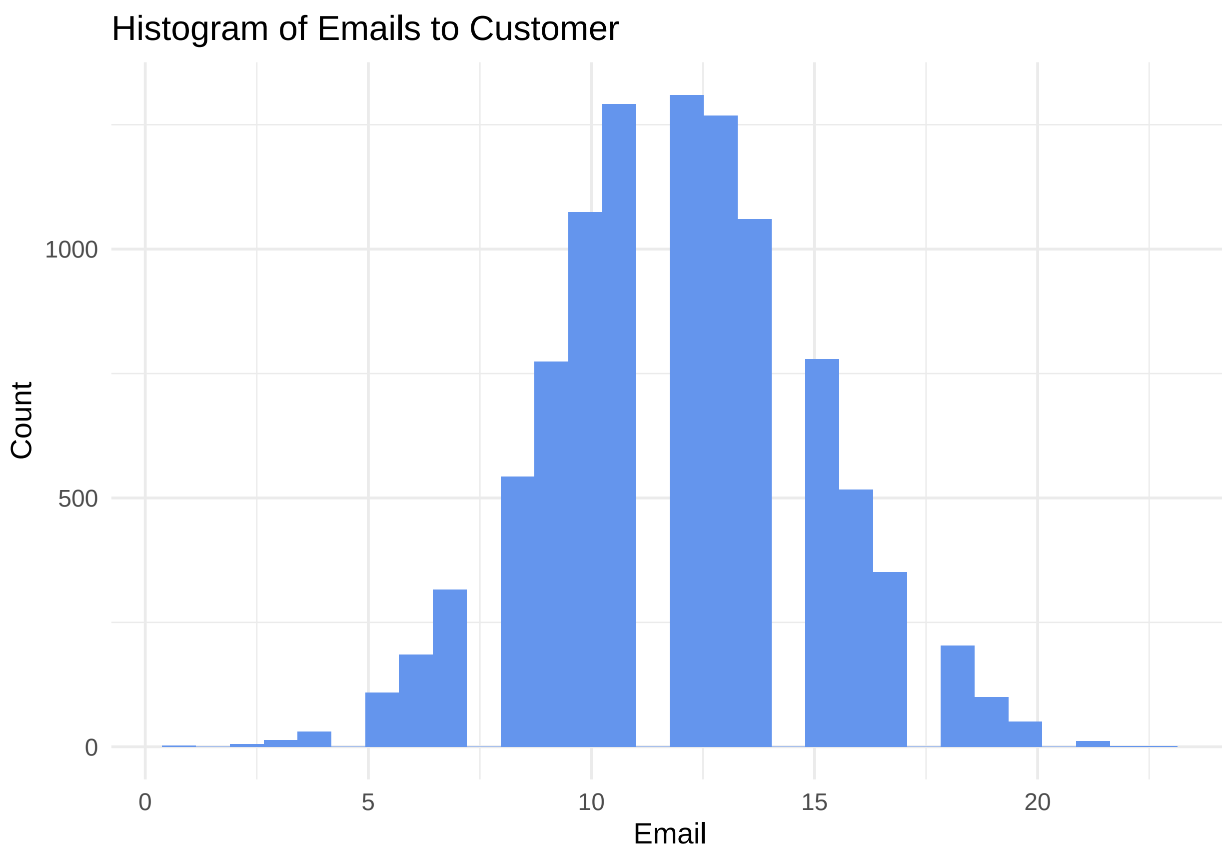
```
ggplot(churn_analysis, aes(x = Outage_sec_perweek))+
  geom_histogram(fill = "cornflowerblue")+
  labs(title = "Histogram of Outage Seconds per Week", y= "Count")+
  theme_minimal()
```




```r
summary(churn_analysis$Outage_sec_perweek)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##  0.09975  8.01821 10.01856 10.00185 11.96949 21.20723
```

```r
ggplot(churn_analysis, aes(x = Email))+
  geom_histogram(fill = "cornflowerblue")+
  labs(title = "Histogram of Emails to Customer", y= "Count")+
  theme_minimal()
```




```
summary(churn_analysis$Email)
```

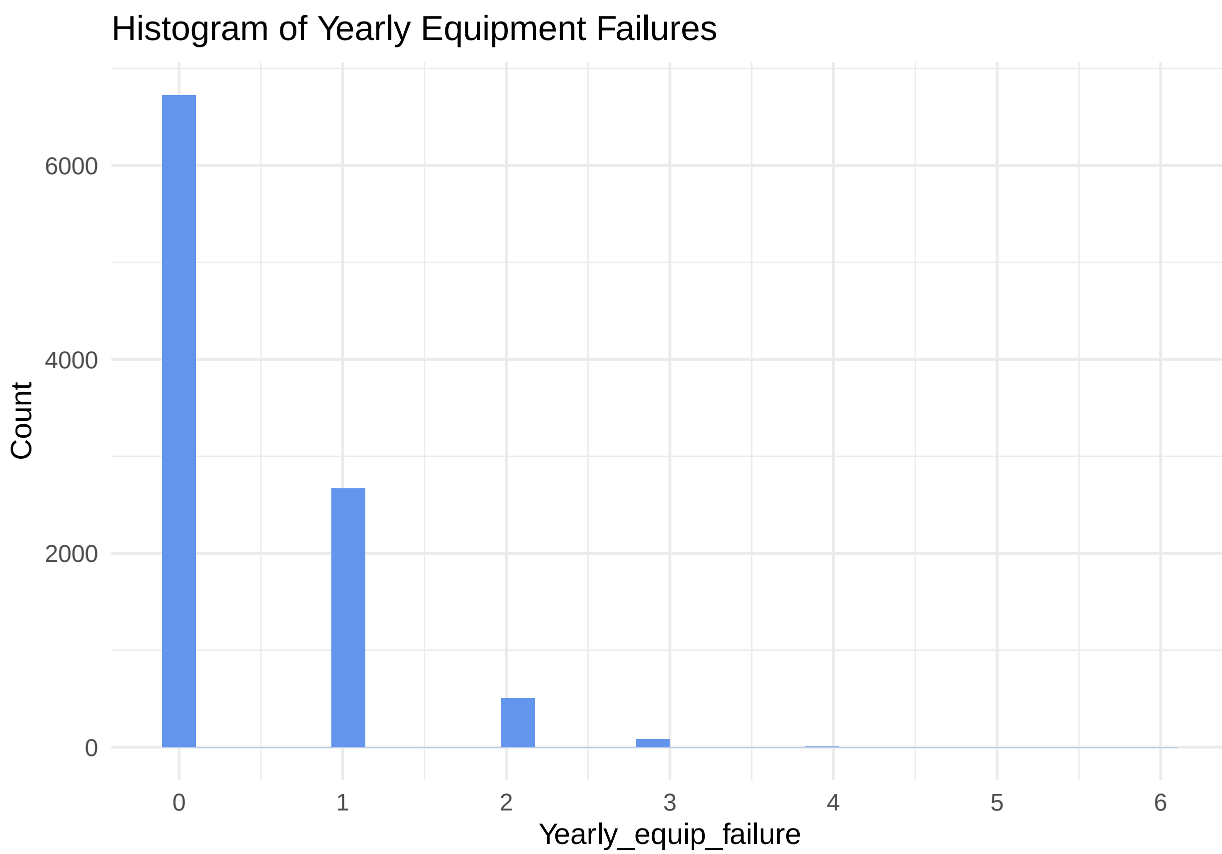
```
##     Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     1.00   10.00   12.00   12.02   14.00   23.00
```

```
ggplot(churn_analysis, aes(x = Contacts))+
  geom_histogram(fill = "cornflowerblue")+
  scale_x_continuous(breaks = seq(0,max(churn_analysis$Contacts),1))+
  labs(title = "Histogram of Contacts from Customer", y= "Count")+
  theme_minimal()
```


```r
summary(churn_analysis$Contacts)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0000  1.0000  0.9942  2.0000  7.0000
```

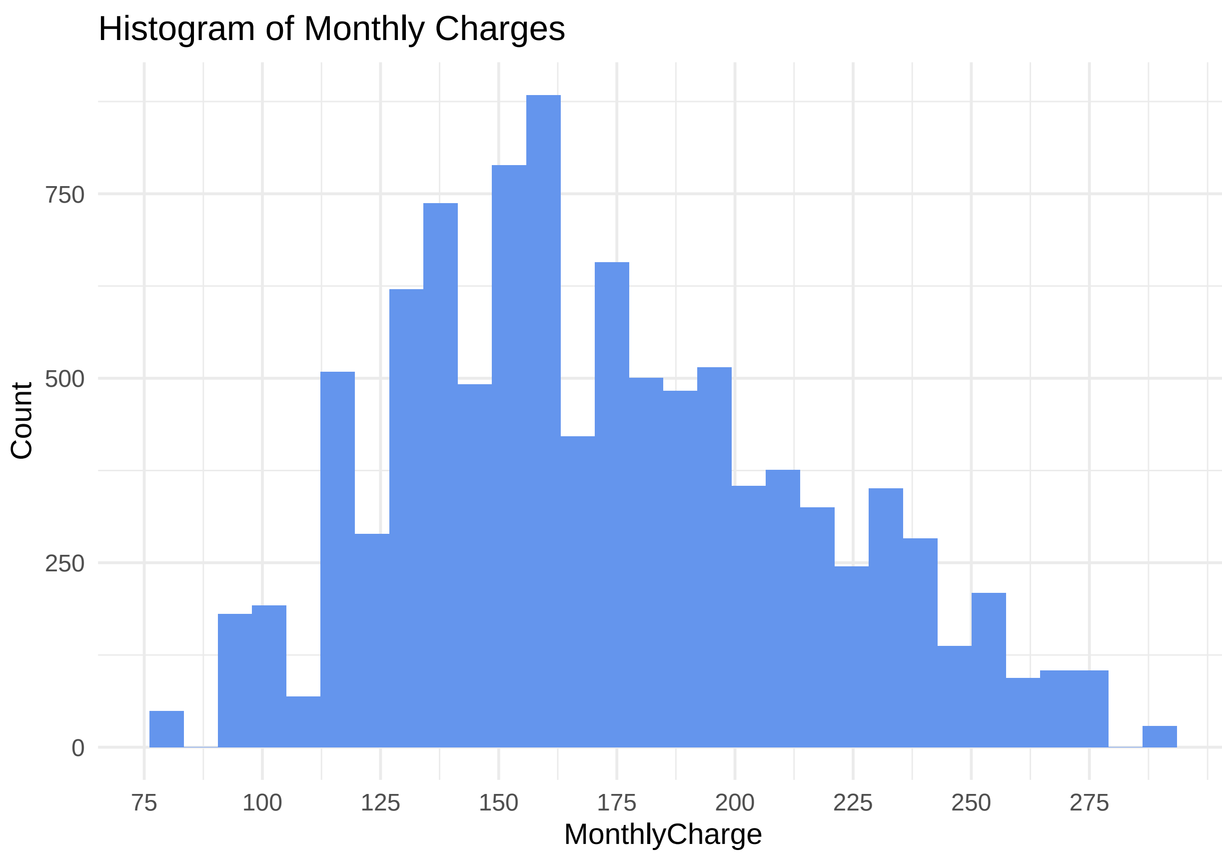
```r
ggplot(churn_analysis, aes(x = Yearly_equip_failure))+
  geom_histogram(fill = "cornflowerblue")+
  scale_x_continuous(breaks = seq(0,max(churn_analysis$Yearly_equip_failure),1))+
  labs(title = "Histogram of Yearly Equipment Failures", y= "Count")+
  theme_minimal()
```




```
summary(churn_analysis$Yearly_equip_failure)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   0.398   1.000   6.000
```

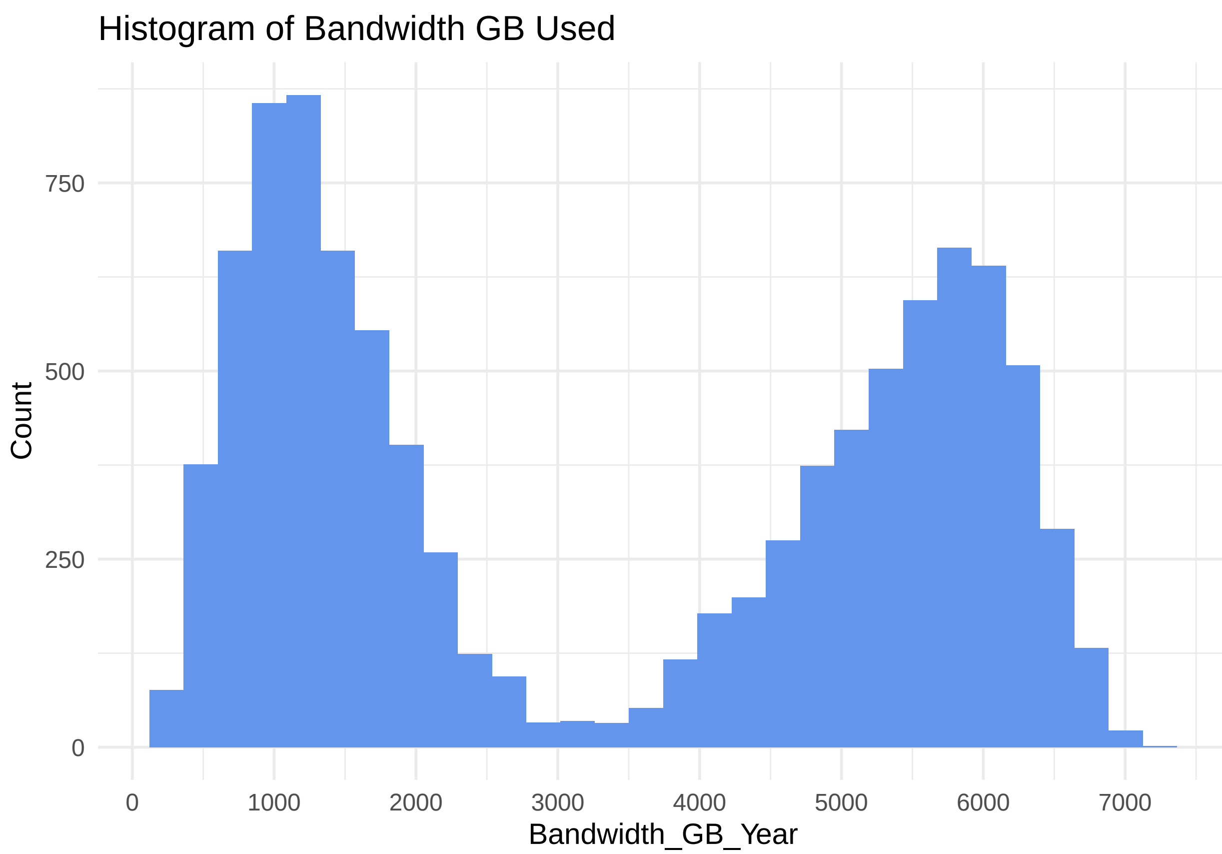
```
ggplot(churn_analysis, aes(x = MonthlyCharge))+
  geom_histogram(fill = "cornflowerblue")+
  scale_x_continuous(breaks = seq(0,max(churn_analysis$MonthlyCharge),25))+
  labs(title = "Histogram of Monthly Charges", y= "Count")+
  theme_minimal()
```




```r
summary(churn_analysis$MonthlyCharge)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   79.98  139.98  167.48  172.62  200.73  290.16
```

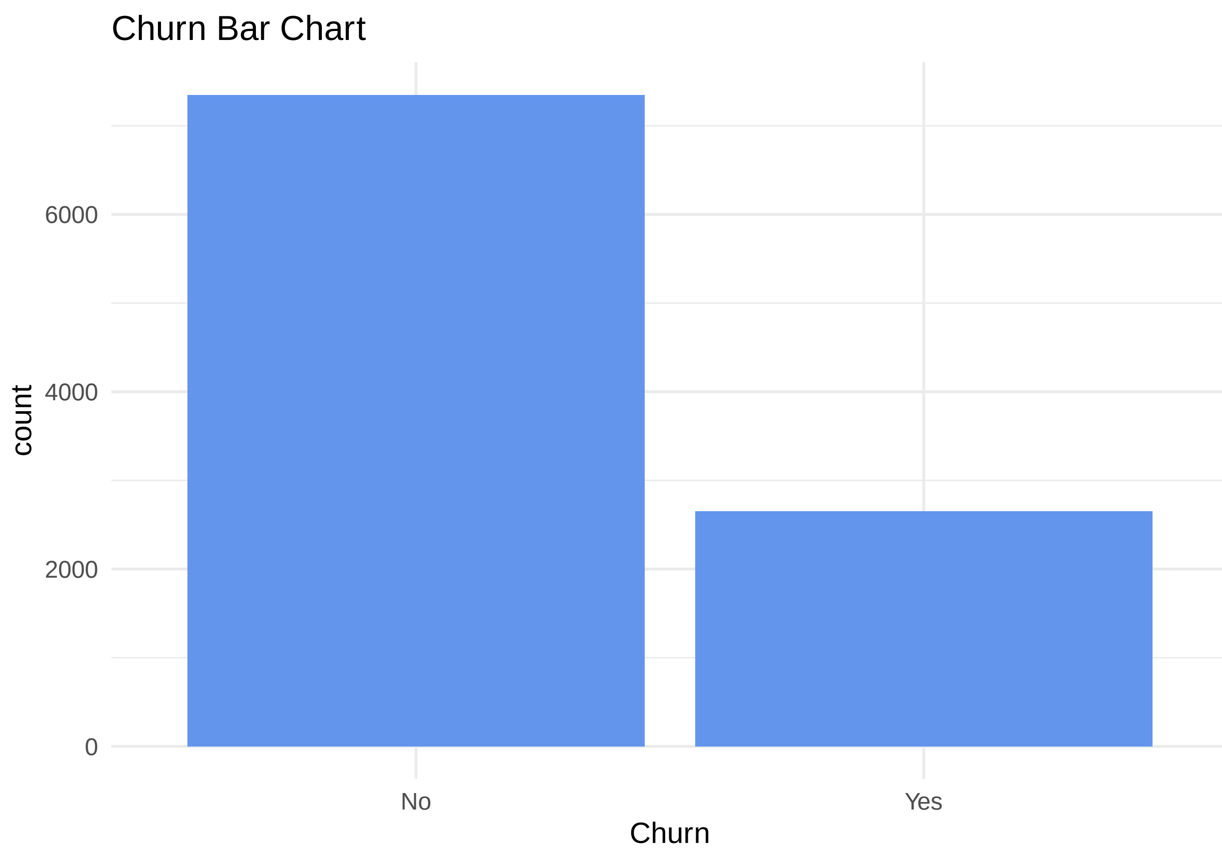
```r
ggplot(churn_analysis, aes(x = Bandwidth_GB_Year))+
  geom_histogram(fill = "cornflowerblue")+
  scale_x_continuous(breaks = seq(0,max(churn_analysis$Bandwidth_GB_Year),1000))+
  labs(title = "Histogram of Bandwidth GB Used", y= "Count")+
  theme_minimal()
```




```r
summary(churn_analysis$Bandwidth_GB_Year)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   155.5  1236.5  3279.5  3392.3  5586.1  7159.0
```

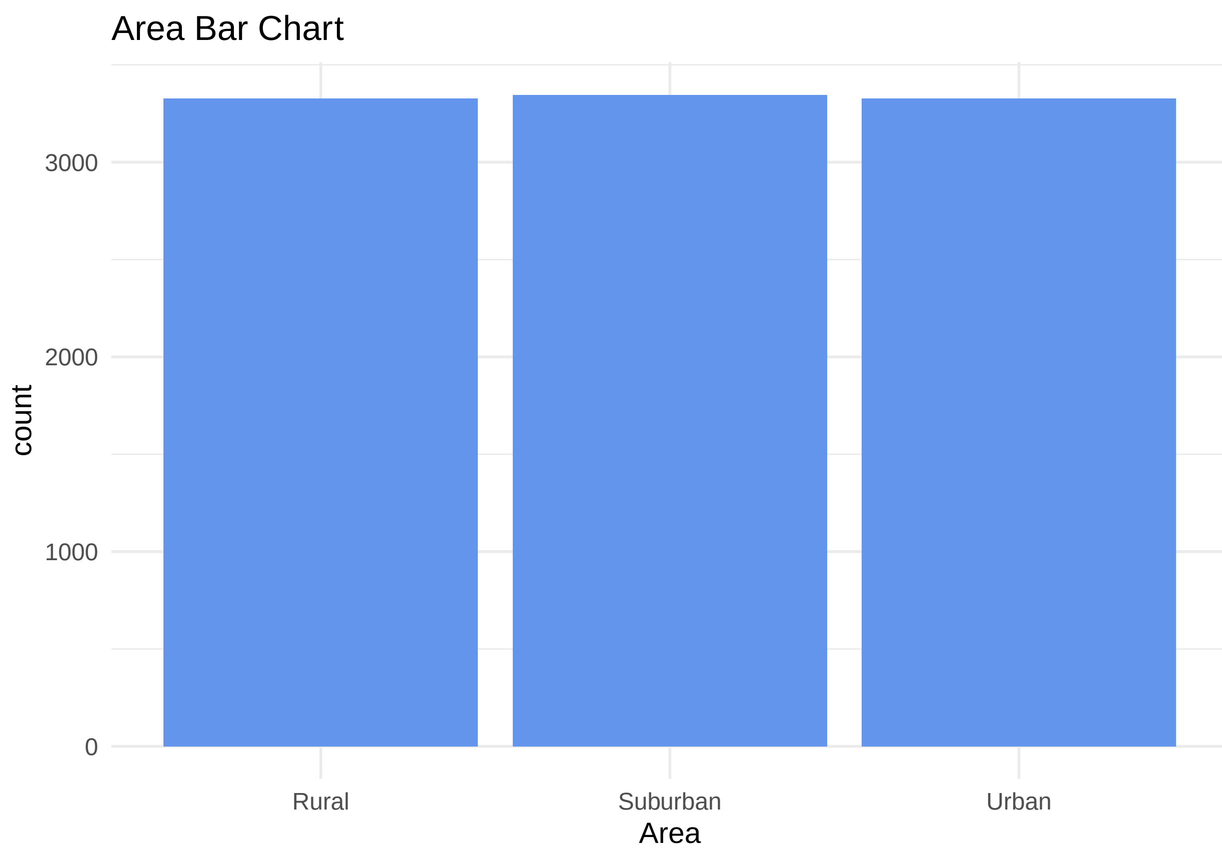
```r
# Categorical Variables
ggplot(churn_analysis, aes(x = Churn))+
  geom_bar(fill = "cornflowerblue")+
  labs(title = "Churn Bar Chart")+
  theme_minimal()
```




```
table(churn_analysis$Churn)
```

```
##
##   No  Yes
## 7350 2650
```

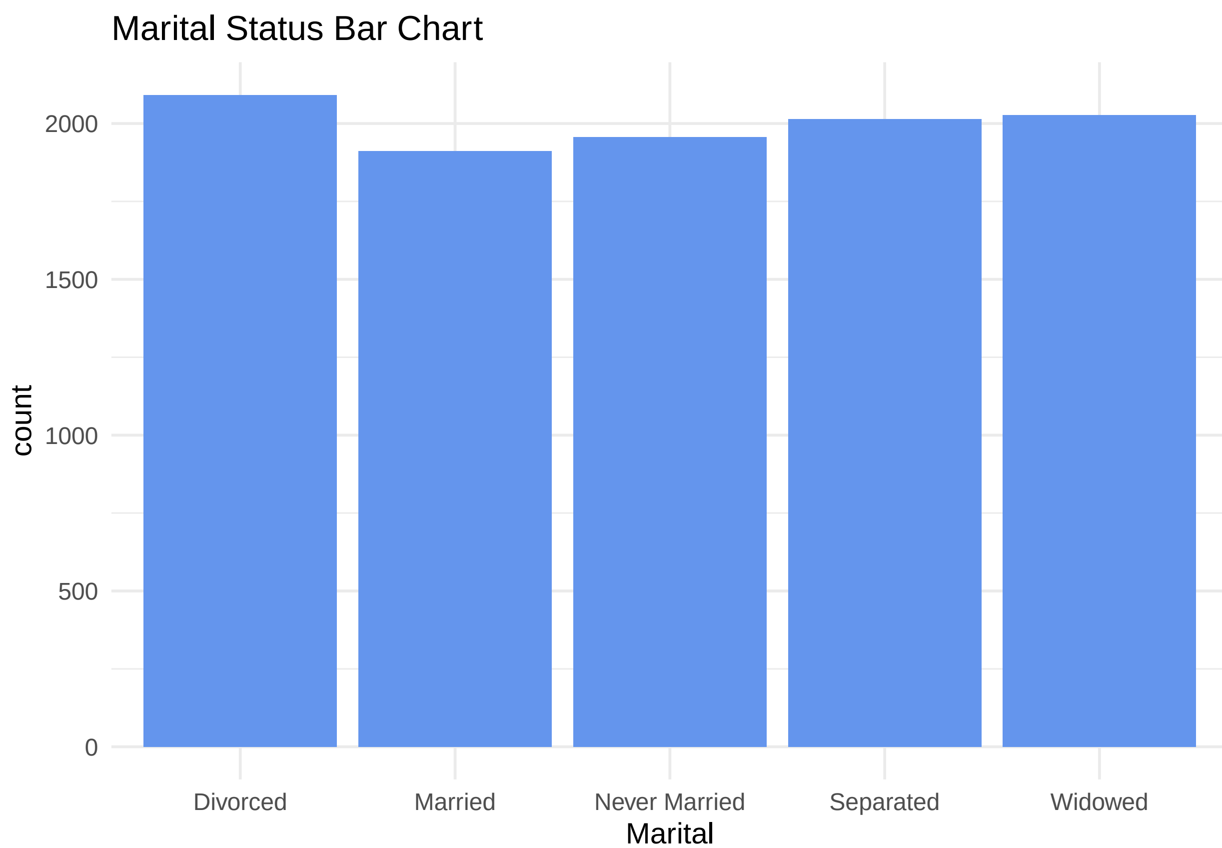
```
ggplot(churn_analysis, aes(x = Area))+
  geom_bar(fill = "cornflowerblue")+
  labs(title = "Area Bar Chart")+
  theme_minimal()
```




```
table(churn_analysis$Area)
```

```
##
##    Rural Suburban    Urban
##     3327     3346     3327
```

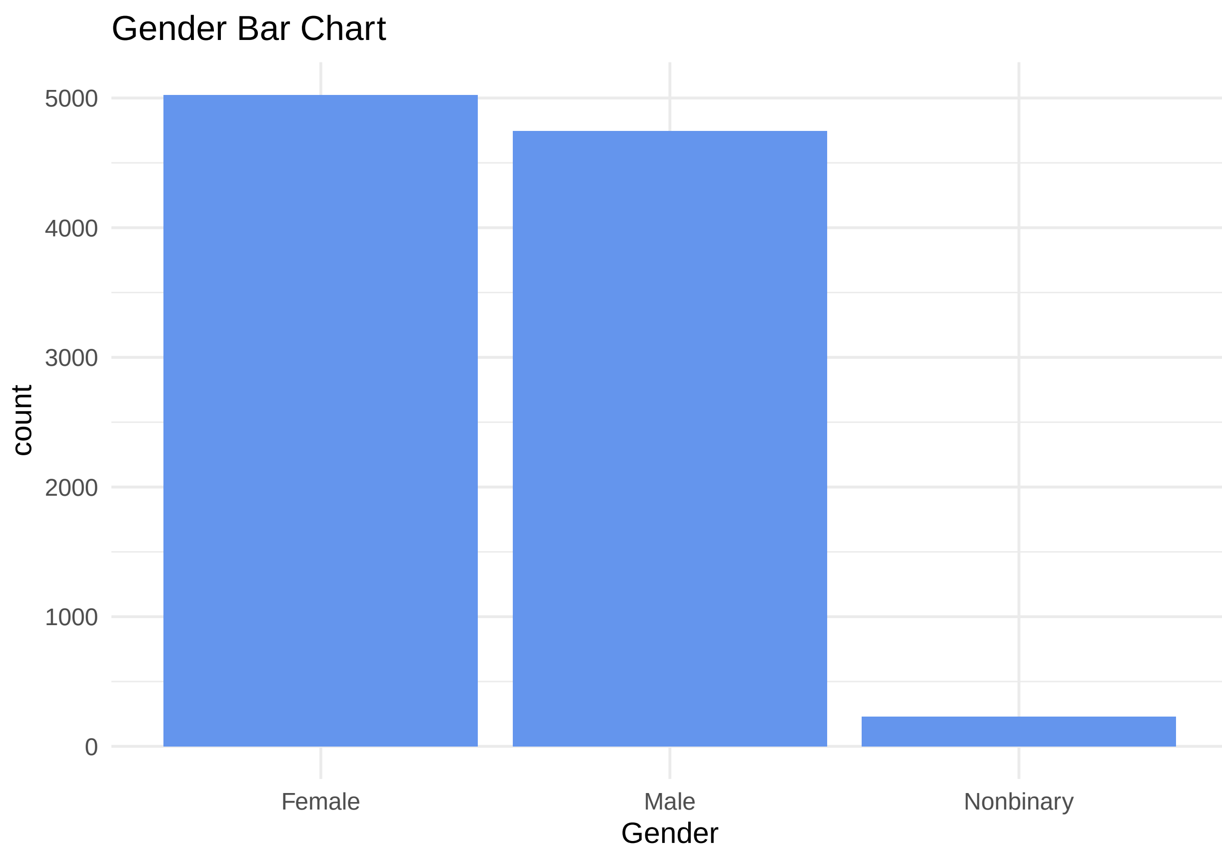
```
ggplot(churn_analysis, aes(x = Marital))+
  geom_bar(fill = "cornflowerblue")+
  labs(title = "Marital Status Bar Chart")+
  theme_minimal()
```




```
table(churn_analysis$Marital)
```

```
##
##      Divorced       Married Never Married     Separated       Widowed
##          2092          1911          1956          2014          2027
```

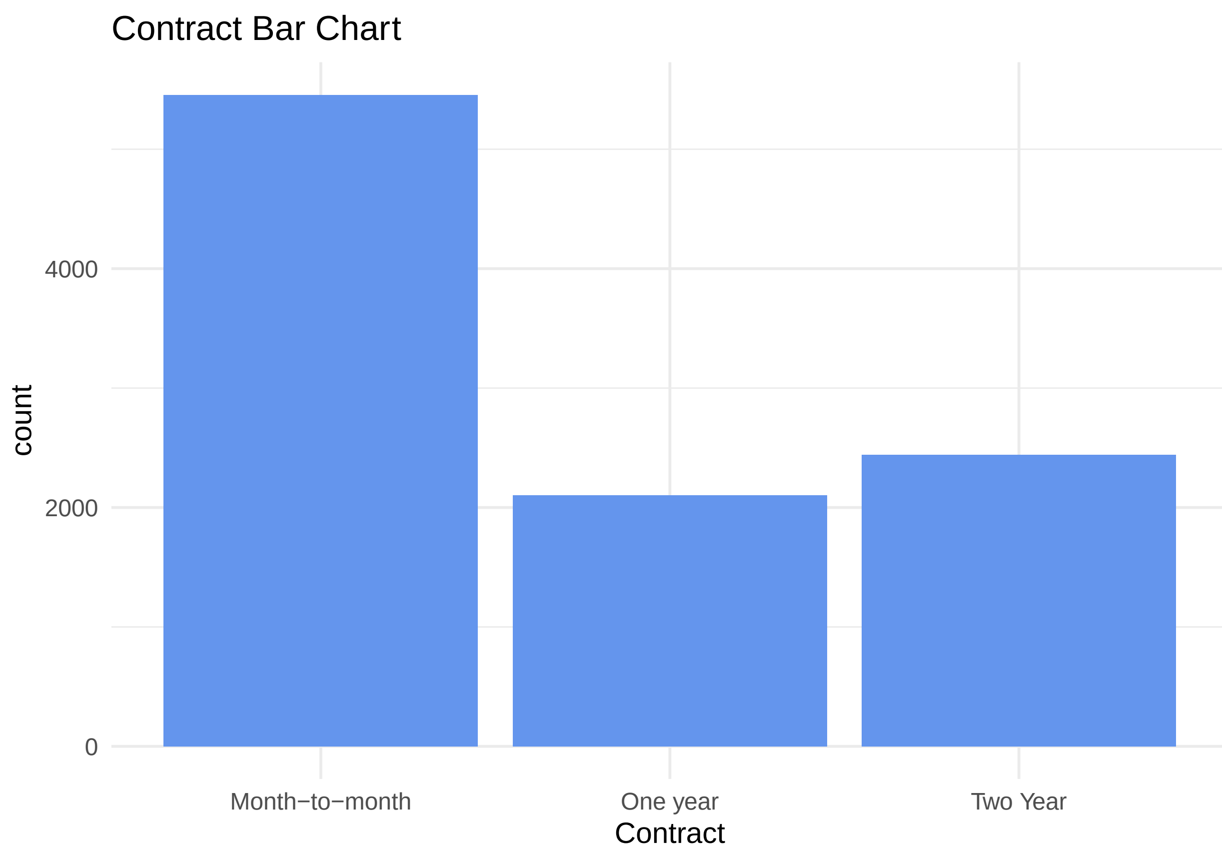
```
ggplot(churn_analysis, aes(x = Gender))+
  geom_bar(fill = "cornflowerblue")+
  labs(title = "Gender Bar Chart")+
  theme_minimal()
```




```
table(churn_analysis$Gender)
```

```
##
##    Female      Male Nonbinary
##      5025      4744       231
```

```
ggplot(churn_analysis, aes(x = Contract))+
  geom_bar(fill = "cornflowerblue")+
  labs(title = "Contract Bar Chart")+
  theme_minimal()
```

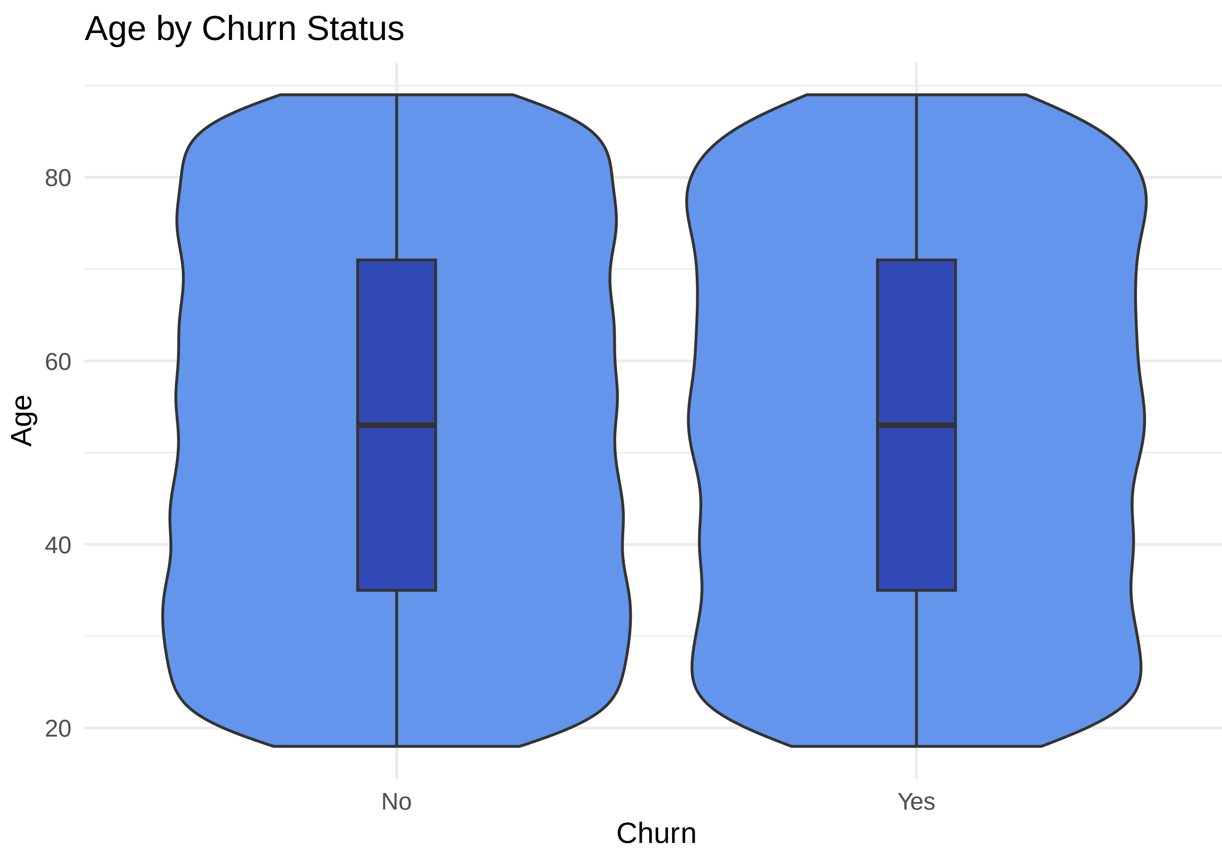



```r
table(churn_analysis$Contract)
```

```
##
## Month-to-month       One year       Two Year
##           5456           2102           2442
```

Bivariate Visualizations

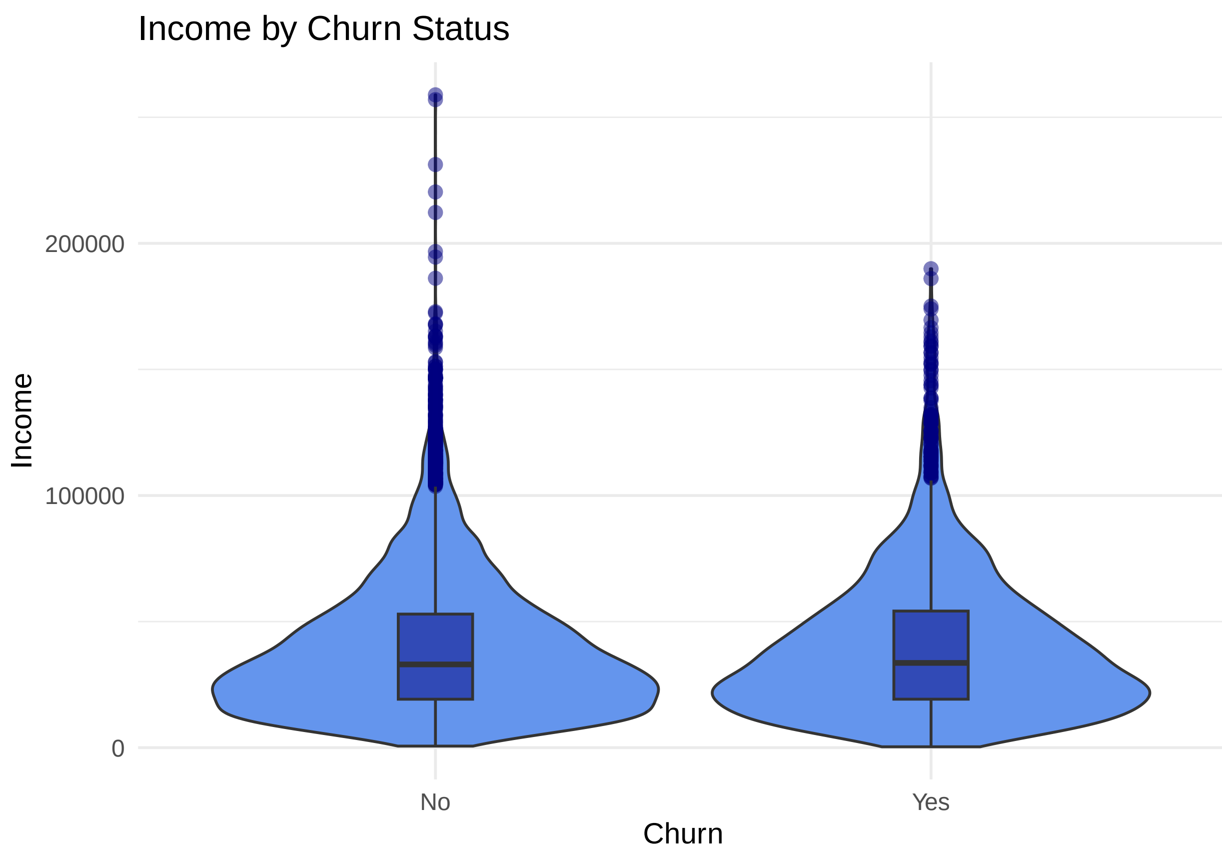
```r
# Categorical vs Quantitative
ggplot(churn_analysis, aes(x = Churn, y = Age))+
  geom_violin(fill = "cornflowerblue")+
  geom_boxplot(width = .15, alpha = 0.5, fill = "navy", outlier.color = "navy", outlier.size = 2)+
  labs(title = "Age by Churn Status")+
  theme_minimal()
```




```
chisq.test(churn_analysis$Age, churn_analysis$Churn)
```

```
##
##  Pearson's Chi-squared test
##
## data:  churn_analysis$Age and churn_analysis$Churn
## X-squared = 61.972, df = 71, p-value = 0.769
```

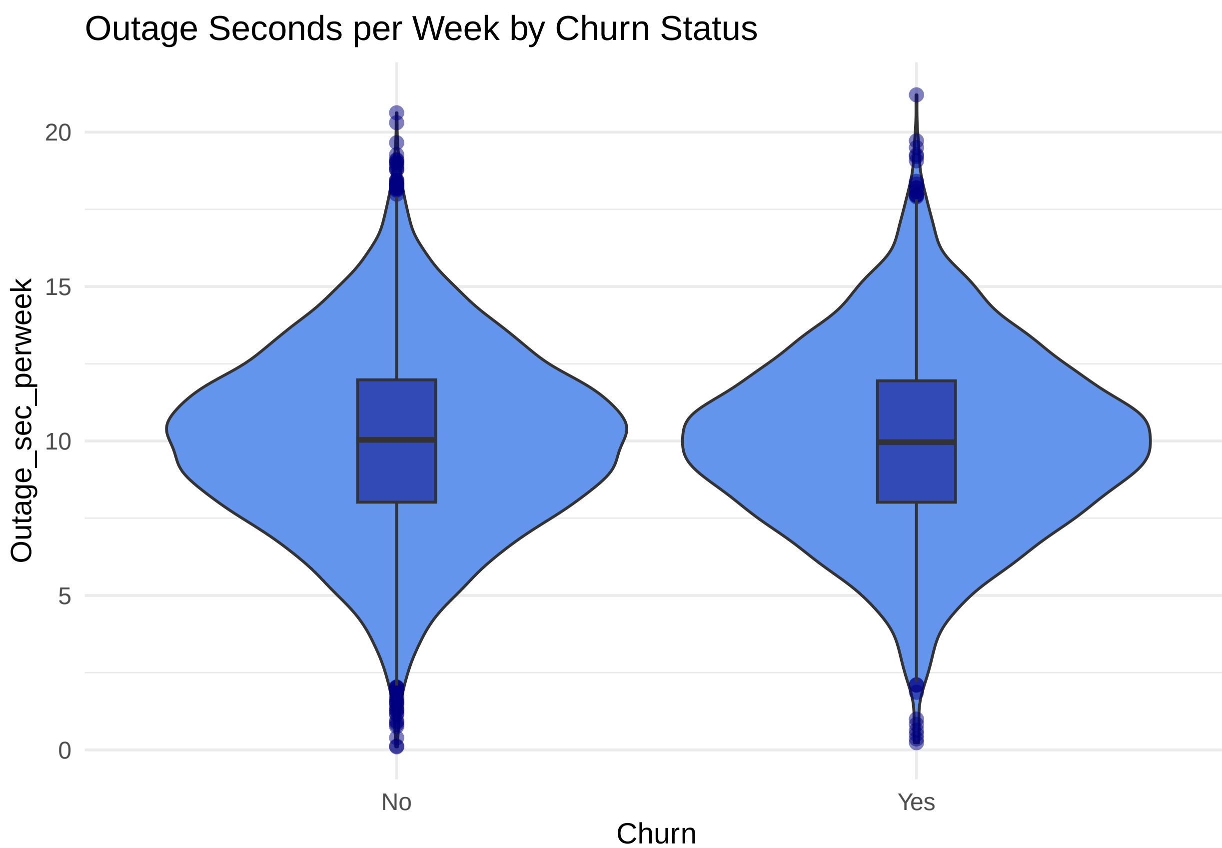
```
ggplot(churn_analysis, aes(x = Churn, y = Income))+
  geom_violin(fill = "cornflowerblue")+
  geom_boxplot(width = .15, alpha = 0.5, fill = "navy", outlier.color = "navy", outlier.size = 2)+
  labs(title = "Income by Churn Status")+
  theme_minimal()
```




```
chisq.test(churn_analysis$Income, churn_analysis$Churn)
```

```
##
##  Pearson's Chi-squared test
##
## data:  churn_analysis$Income and churn_analysis$Churn
## X-squared = 9994.9, df = 9992, p-value = 0.49
```

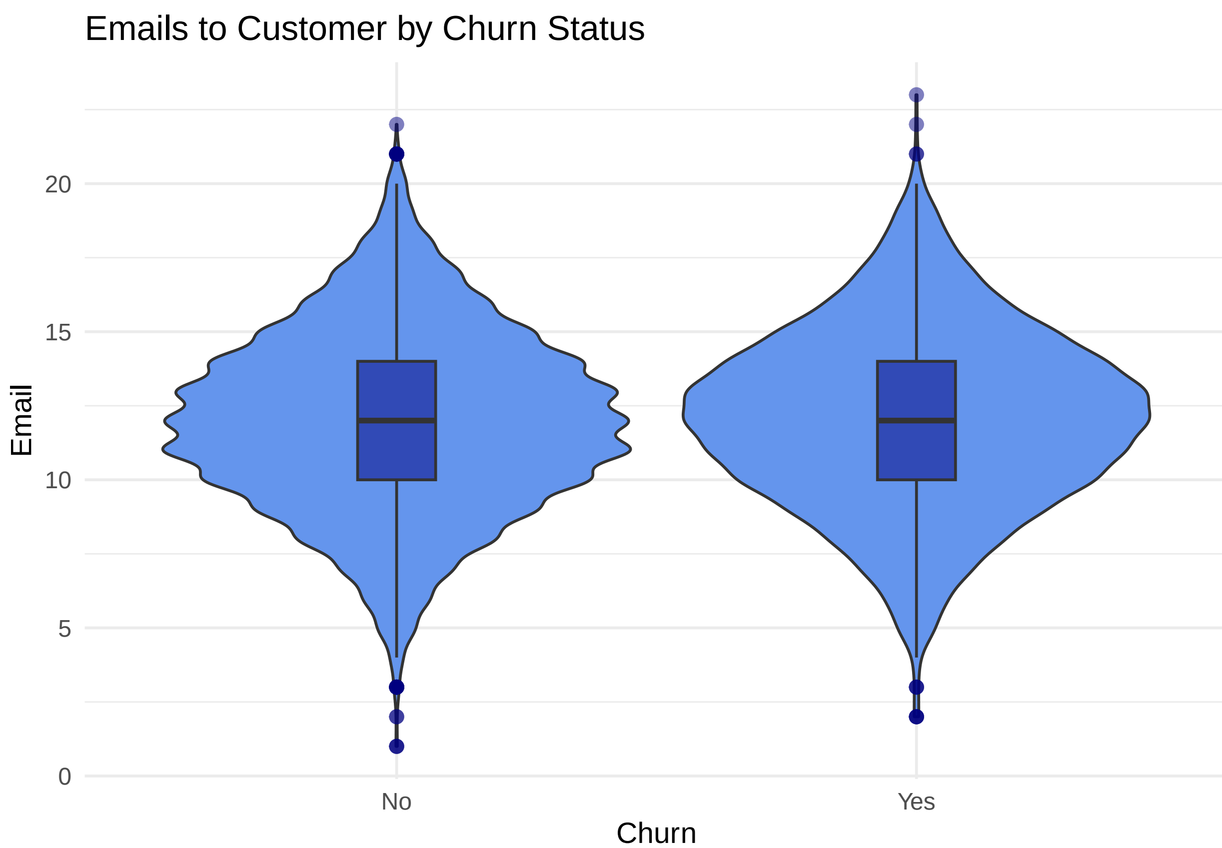
```
ggplot(churn_analysis, aes(x = Churn, y = Outage_sec_perweek))+
  geom_violin(fill = "cornflowerblue")+
  geom_boxplot(width = .15, alpha = 0.5, fill = "navy", outlier.color = "navy", outlier.size = 2)+
  labs(title = "Outage Seconds per Week by Churn Status")+
  theme_minimal()
```




```
chisq.test(churn_analysis$Outage_sec_perweek, churn_analysis$Churn)
```

```
##
##  Pearson's Chi-squared test
##
## data:  churn_analysis$Outage_sec_perweek and churn_analysis$Churn
## X-squared = 9982, df = 9985, p-value = 0.5065
```

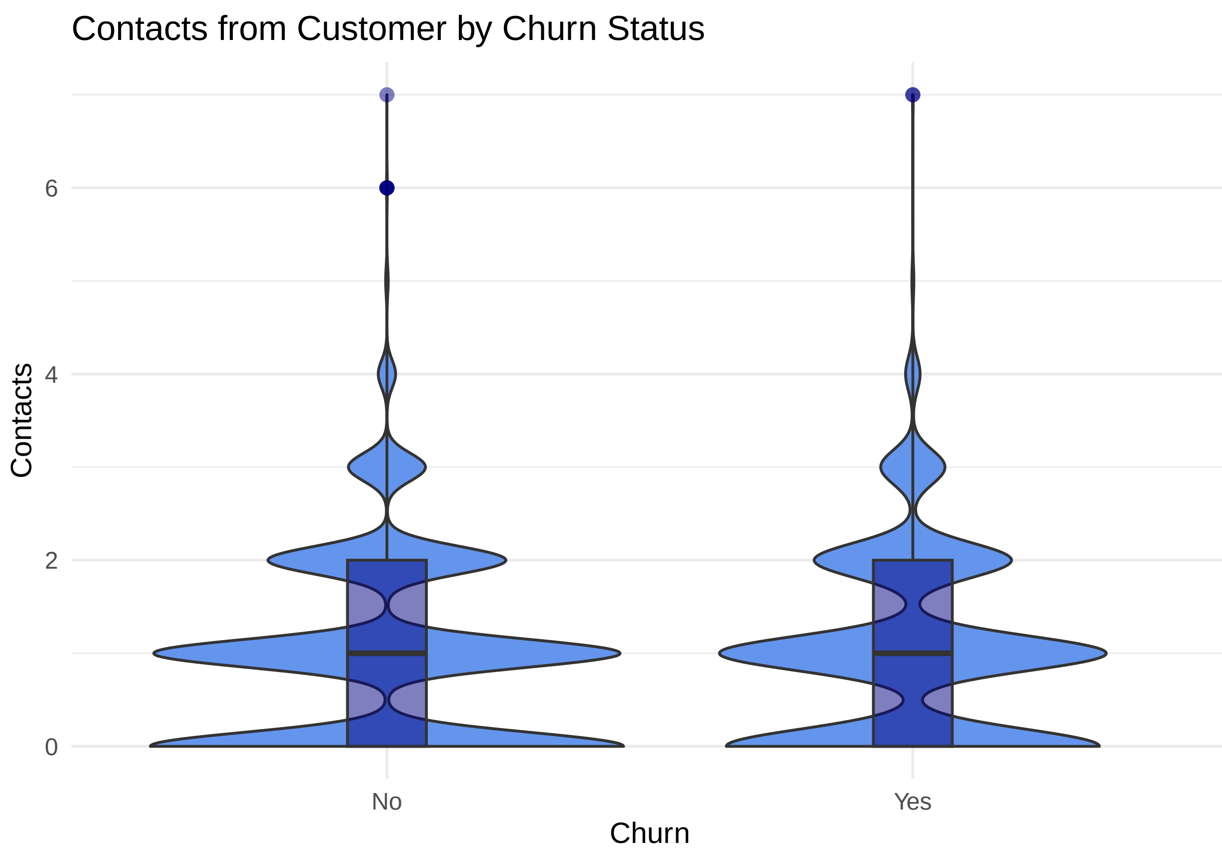
```
ggplot(churn_analysis, aes(x = Churn, y = Email))+
  geom_violin(fill = "cornflowerblue")+
  geom_boxplot(width = .15, alpha = 0.5, fill = "navy", outlier.color = "navy", outlier.size = 2)+
  labs(title = "Emails to Customer by Churn Status")+
  theme_minimal()
```




```
chisq.test(churn_analysis$Email, churn_analysis$Churn)
```

```
##
##  Pearson's Chi-squared test
##
## data:  churn_analysis$Email and churn_analysis$Churn
## X-squared = 23.111, df = 22, p-value = 0.3955
```

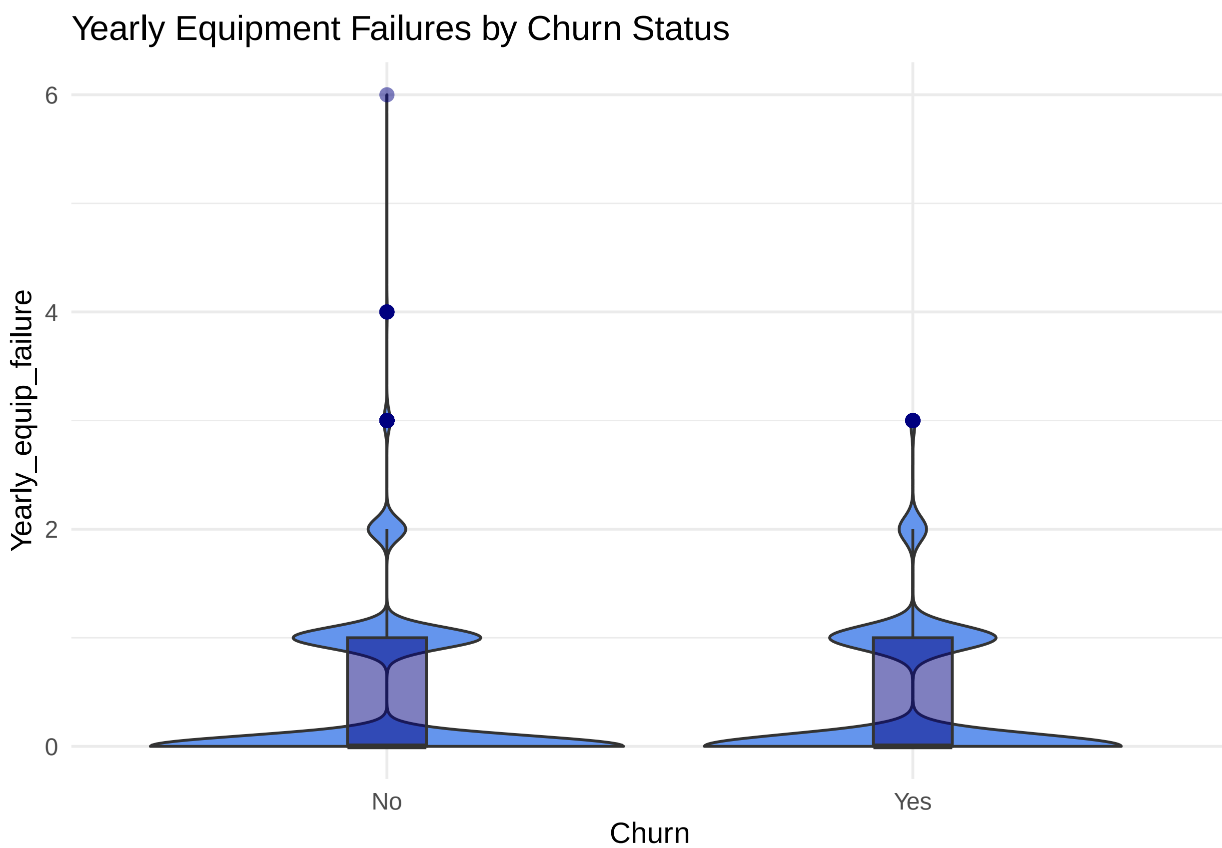
```
ggplot(churn_analysis, aes(x = Churn, y = Contacts))+
  geom_violin(fill = "cornflowerblue")+
  geom_boxplot(width = .15, alpha = 0.5, fill = "navy", outlier.color = "navy", outlier.size = 2)+
  labs(title = "Contacts from Customer by Churn Status")+
  theme_minimal()
```




```
chisq.test(churn_analysis$Contacts, churn_analysis$Churn)
```

```
##
##  Pearson's Chi-squared test
##
## data:  churn_analysis$Contacts and churn_analysis$Churn
## X-squared = 5.5218, df = 7, p-value = 0.5966
```

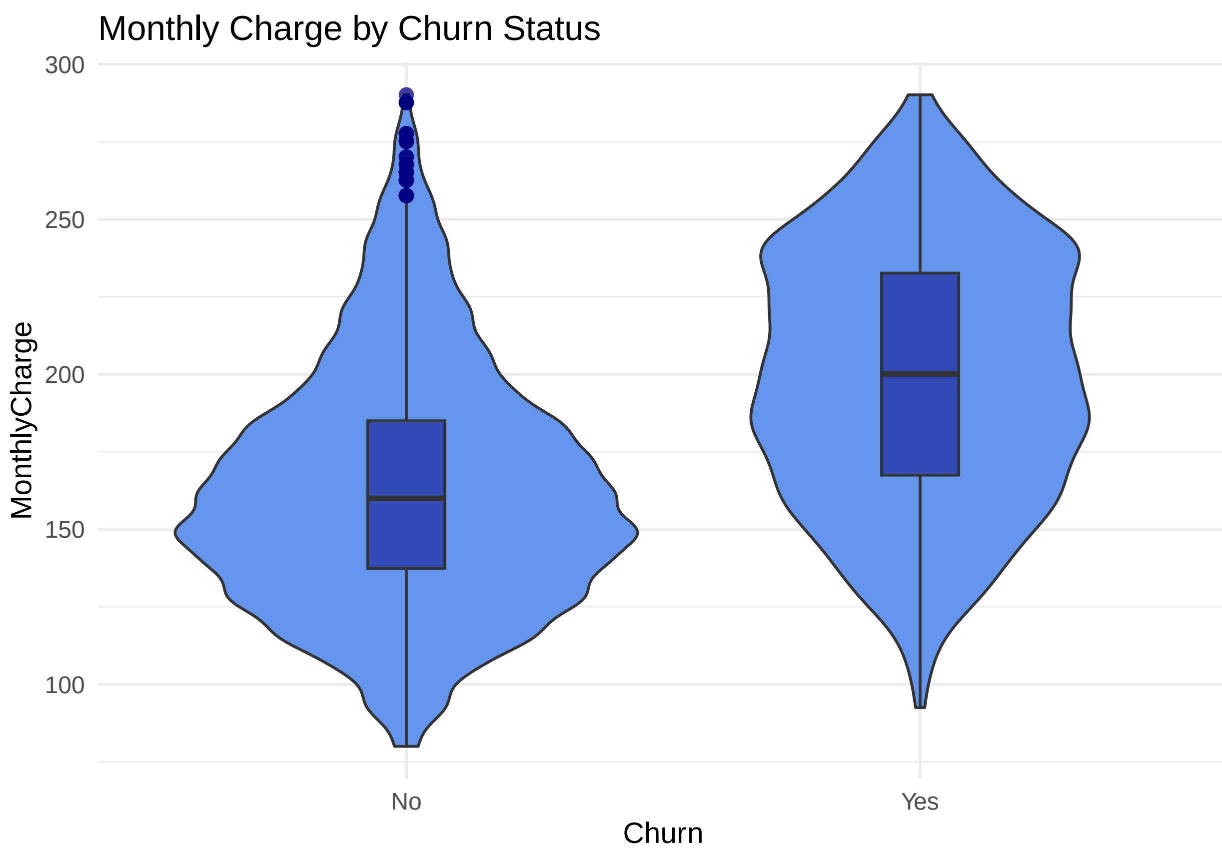
```
ggplot(churn_analysis, aes(x = Churn, y = Yearly_equip_failure))+
  geom_violin(fill = "cornflowerblue")+
  geom_boxplot(width = .15, alpha = 0.5, fill = "navy", outlier.color = "navy", outlier.size = 2)+
  labs(title = "Yearly Equipment Failures by Churn Status")+
  theme_minimal()
```




```
chisq.test(churn_analysis$Yearly_equip_failure, churn_analysis$Churn)
```

```
##
##  Pearson's Chi-squared test
##
## data:  churn_analysis$Yearly_equip_failure and churn_analysis$Churn
## X-squared = 6.9253, df = 5, p-value = 0.2263
```

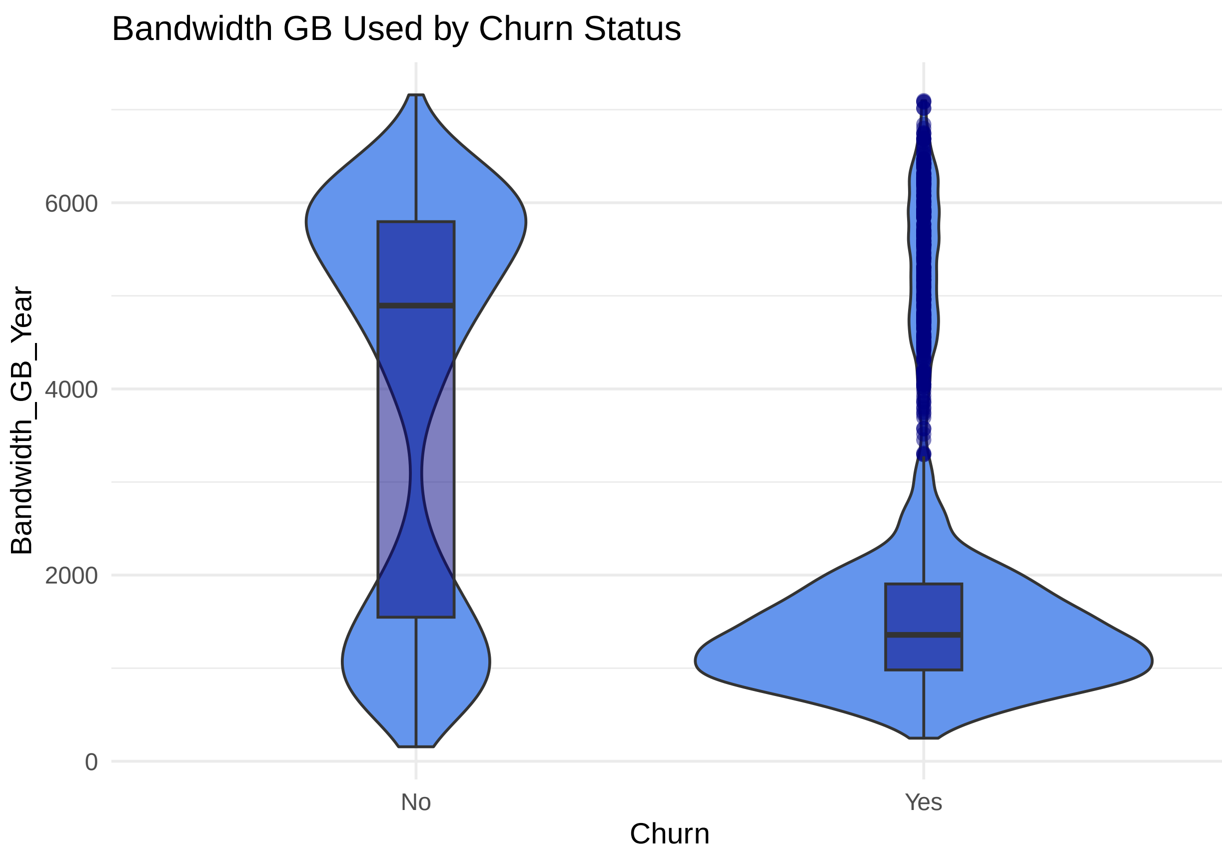
```
ggplot(churn_analysis, aes(x = Churn, y = MonthlyCharge))+
  geom_violin(fill = "cornflowerblue")+
  geom_boxplot(width = .15, alpha = 0.5, fill = "navy", outlier.color = "navy", outlier.size = 2)+
  labs(title = "Monthly Charge by Churn Status")+
  theme_minimal()
```




```
chisq.test(churn_analysis$MonthlyCharge, churn_analysis$Churn)
```

```
##
##  Pearson's Chi-squared test
##
## data:  churn_analysis$MonthlyCharge and churn_analysis$Churn
## X-squared = 3026, df = 749, p-value < 0.00000000000000022
```

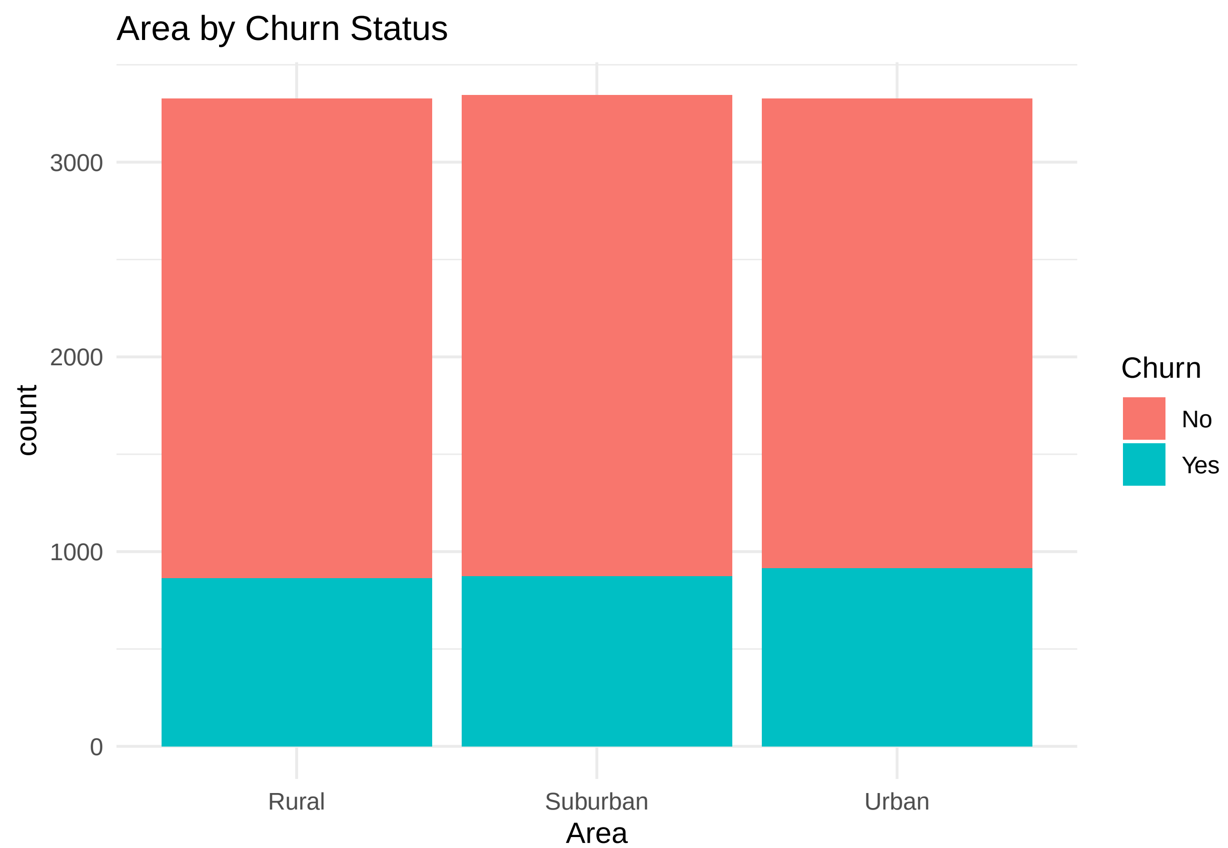
```
ggplot(churn_analysis, aes(x = Churn, y = Bandwidth_GB_Year))+
  geom_violin(fill = "cornflowerblue")+
  geom_boxplot(width = .15, alpha = 0.5, fill = "navy", outlier.color = "navy", outlier.size = 2)+
  labs(title = "Bandwidth GB Used by Churn Status")+
  theme_minimal()
```




```
chisq.test(churn_analysis$Bandwidth_GB_Year, churn_analysis$Churn)
```

```
##
##  Pearson's Chi-squared test
##
## data:  churn_analysis$Bandwidth_GB_Year and churn_analysis$Churn
## X-squared = 10000, df = 9999, p-value = 0.4953
```

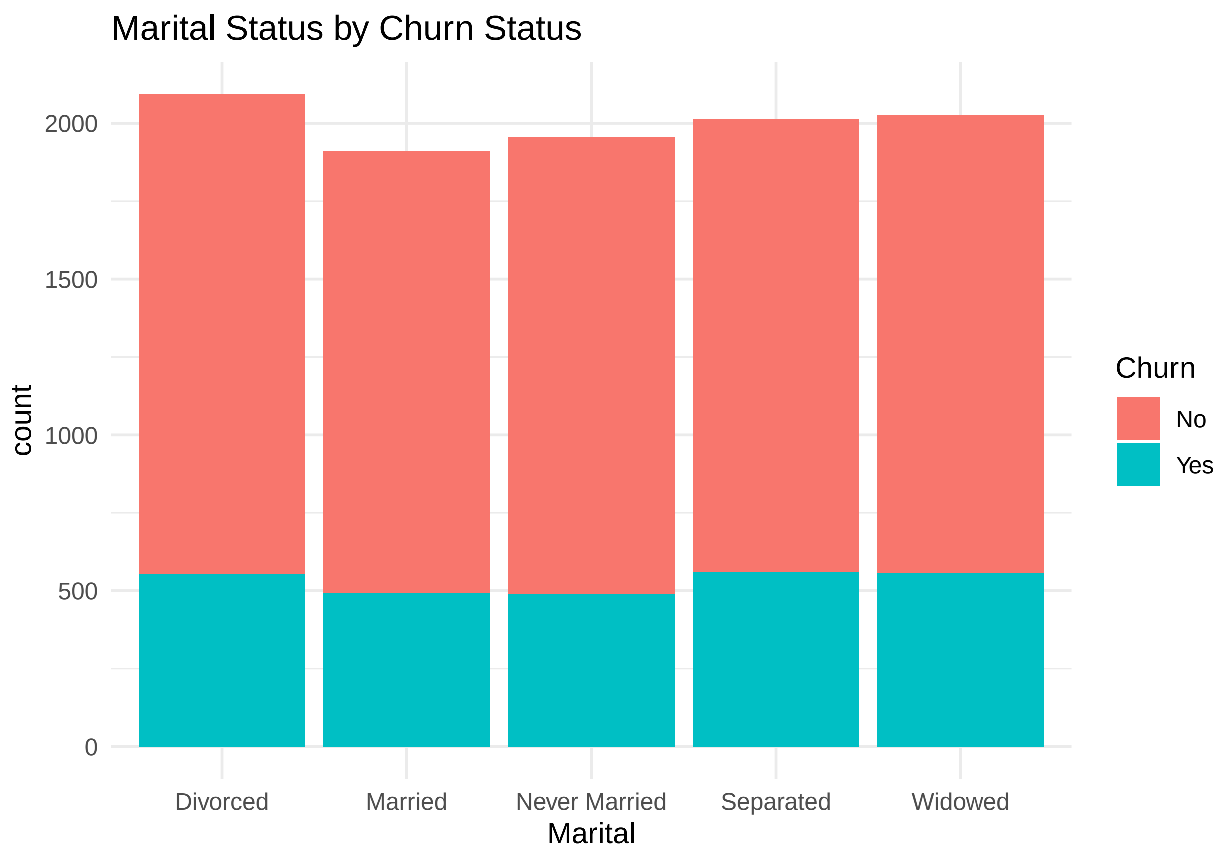
```
# Categorical vs Categorical
ggplot(churn_analysis, aes(x = Area, fill = Churn))+
  geom_bar()+
  labs(title = "Area by Churn Status")+
  theme_minimal()
```




```
chisq.test(churn_analysis$Area, churn_analysis$Churn)
```

```
##
##  Pearson's Chi-squared test
##
## data:  churn_analysis$Area and churn_analysis$Churn
## X-squared = 2.4391, df = 2, p-value = 0.2954
```

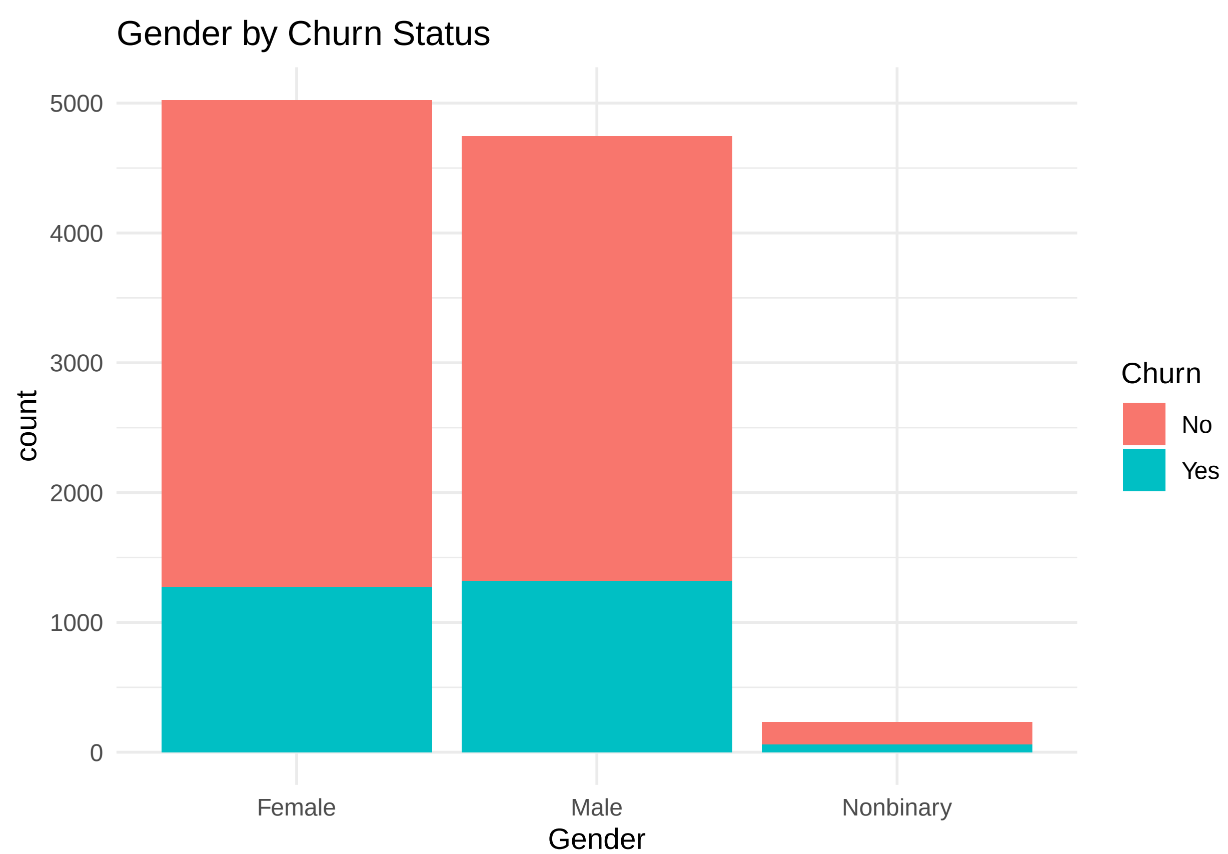
```
ggplot(churn_analysis, aes(x = Marital, fill = Churn))+
  geom_bar()+
  labs(title = "Marital Status by Churn Status")+
  theme_minimal()
```




```
chisq.test(churn_analysis$Marital, churn_analysis$Churn)
```

```
##
##  Pearson's Chi-squared test
##
## data:  churn_analysis$Marital and churn_analysis$Churn
## X-squared = 5.5658, df = 4, p-value = 0.234
```

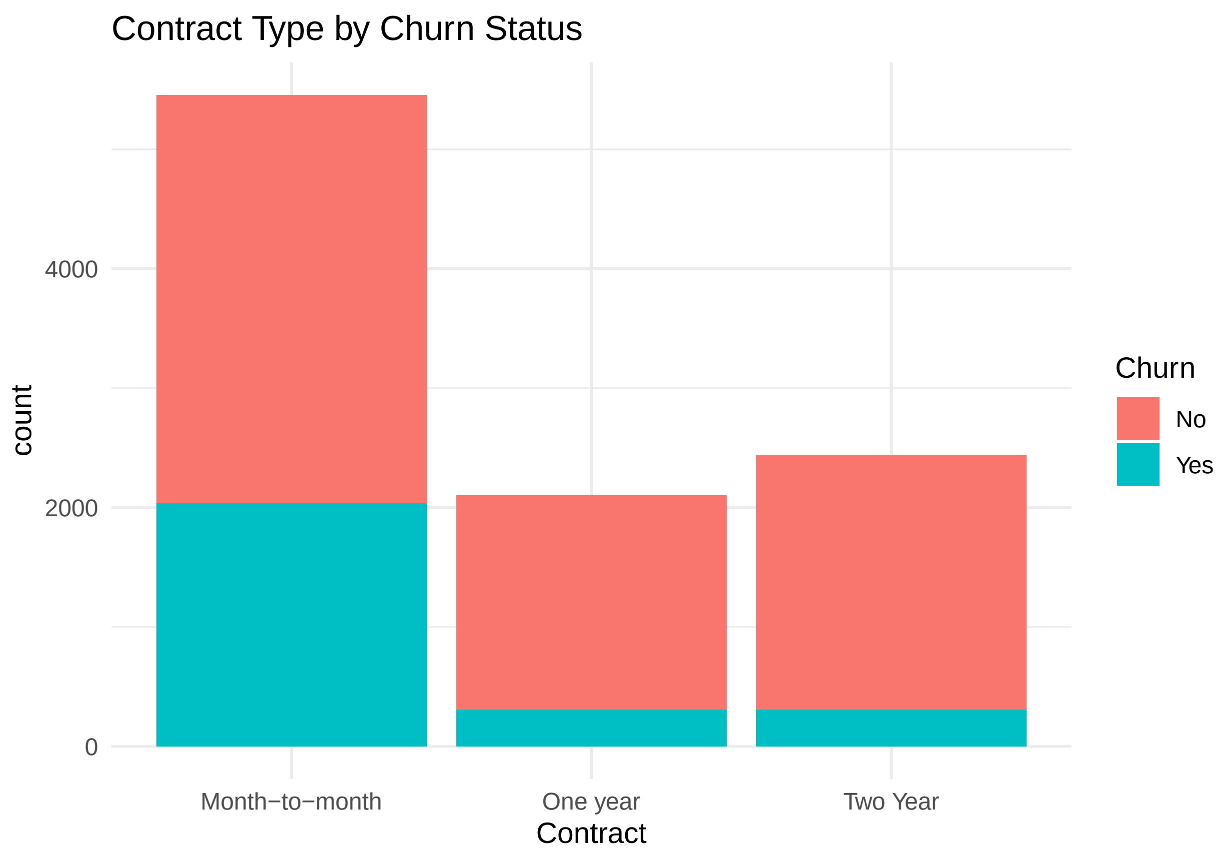
```
ggplot(churn_analysis, aes(x = Gender, fill = Churn))+
  geom_bar()+
  labs(title = "Gender by Churn Status")+
  theme_minimal()
```




```
chisq.test(churn_analysis$Gender, churn_analysis$Churn)
```

```
##
##  Pearson's Chi-squared test
##
## data:  churn_analysis$Gender and churn_analysis$Churn
## X-squared = 7.8801, df = 2, p-value = 0.01945
```

```
ggplot(churn_analysis, aes(x = Contract, fill = Churn))+
  geom_bar()+
  labs(title = "Contract Type by Churn Status")+
  theme_minimal()
```




```r
chisq.test(churn_analysis$Contract, churn_analysis$Churn)
```

```
##
##  Pearson's Chi-squared test
##
## data:  churn_analysis$Contract and churn_analysis$Churn
## X-squared = 718.59, df = 2, p-value < 0.00000000000000022
```

C4, Data Transformation Goals

The data was checked for duplicate records, missing values, and outliers. It was deemed clean in that regard. An analysis data frame was created to isolate the 13 variables used in the regression model. Of the 13 variables, five were categorical. In order to properly perform regression, they needed to be re-expressed as numeric variables. The dependent variable, churn, was also transformed from Yes/No to numeric 1/0. This enabled the creation of a correlation matrix.

One hot encoding method was used to transform the data. This meant that each category needed its own binary column, and to avoid multicollinearity, one of the category columns had to be dropped. The 'dummy_cols' function from the 'fastDummies' library was used to perform the transformation. This package allows the user to create dummy columns for all categorical variables. It lets the user drop the first dummy column and remove the source columns (Kaplan, 2020).

An additional data frame was created for the transformed data so the original could be referred back to if needed. The resulting dataset used in the initial regression model contained 19 variables after transformation. The cleaned and transformed data was written to a CSV file as well. The code executed for the transformation is provided below.

```
# create new df for initial model with re-expressed categorical variables [In-text citation: (Kaplan, 2
churn_initial <- churn_analysis

churn_initial$Churn[churn_initial$Churn == "Yes"] <- 1
churn_initial$Churn[churn_initial$Churn == "No"] <- 0
churn_initial$Churn <- as.numeric(churn_initial$Churn)

churn_initial <- dummy_cols(
  churn_initial,
  select_columns =
    c("Area","Marital","Gender","Contract"),
  remove_first_dummy = TRUE,
  remove_selected_columns = TRUE
)
```

C5, Prepared Data Set

The cleaned and transformed dataset used in the initial multiple logistic regression model was written to a CSV file and is included in the submission.

D1, Initial MLR Model

An initial multiple logistic regression model was created using all 19 variables from the transformed dataset. This included churn as the dependent variable and all remaining features as the independent variables. Eight independent variables had a p-value less than 0.05, which would be deemed significant. The initial model had an AIC of 5848.9.

```
model_initial <- glm(Churn ~ ., data = churn_initial, family = "binomial")
summary(model_initial)
```

```
##
## Call:
## glm(formula = Churn ~ ., family = "binomial", data = churn_initial)
##
## Coefficients:
##                          Estimate  Std. Error z value             Pr(>|z|)
## (Intercept)          -5.239474179 0.273295573 -19.171 < 0.0000000000000002
## Age                  -0.003343172 0.001617020  -2.067             0.038688
## Income                0.000001111 0.000001173   0.947             0.343430
## Outage_sec_perweek   -0.003078095 0.011096911  -0.277             0.781486
## Email                -0.002688606 0.010982837  -0.245             0.806611
## Contacts              0.043775723 0.033145933   1.321             0.186603
## Yearly_equip_failure -0.035708333 0.052683157  -0.678             0.497902
## MonthlyCharge         0.042844985 0.001087560  39.396 < 0.0000000000000002
## Bandwidth_GB_Year    -0.000984748 0.000023789 -41.394 < 0.0000000000000002
## Area_Suburban        -0.019601112 0.081497620  -0.241             0.809934
```

```
## Area_Urban              0.044024304  0.081334107   0.541             0.588316
## Marital_Married         0.021832689  0.105165345   0.208             0.835539
## `Marital_Never Married` -0.060031421  0.105268991  -0.570             0.568497
## Marital_Separated       0.128598232  0.103552725   1.242             0.214287
## Marital_Widowed         0.219350560  0.102637026   2.137             0.032586
## Gender_Male             0.251985270  0.067212986   3.749             0.000178
## Gender_Nonbinary       -0.231771045  0.226721725  -1.022             0.306653
## `Contract_One year`    -2.485743783  0.100948927 -24.624 < 0.0000000000000002
## `Contract_Two Year`    -2.575025647  0.097932783 -26.294 < 0.0000000000000002
##
## (Intercept)             ***
## Age                     *
## Income
## Outage_sec_perweek
## Email
## Contacts
## Yearly_equip_failure
## MonthlyCharge           ***
## Bandwidth_GB_Year       ***
## Area_Suburban
## Area_Urban
## Marital_Married
## `Marital_Never Married`
## Marital_Separated
## Marital_Widowed         *
## Gender_Male             ***
## Gender_Nonbinary
## `Contract_One year`     ***
## `Contract_Two Year`     ***
## ---
## 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:  5810.9  on 9981  degrees of freedom
## AIC: 5848.9
##
## Number of Fisher Scoring iterations: 6
```

D2, Justification of Feature Selection

After running the initial regression model, it was clear that some variables were not adding value to the model based on the resulting p-values. The model was checked to ensure there was no multicollinearity. The VIF function was applied to the initial model, and no variables had an inflation factor greater than 10, which meant there was no multicollinearity present in the model (D208 Webinar, n.d.).

Backward stepwise elimination was then performed as a feature selection technique to reduce the number of variables in the model. This method starts with all variables and removes them one by one until there is no longer a statistically valid reason to remove more variables. The step function utilizes AIC scoring to determine the combination of variables that provide the lowest AIC score. While one of the final variables is not considered significant, it did lead to the model with the lowest AIC score. Thus, the remaining coefficients that make up the reduced regression model and are either statistically significant or are found to add value with their interaction. After stepwise elimination, eight explanatory variables were chosen for the reduced model (Bobbitt, 2019).

```
# check for multicollinearity [In-text citation:(D208 Webinar, n.d.)]
vif(model_initial)
```

```
##                   Age                 Income      Outage_sec_perweek
##              1.007404               1.002899                1.004199
##                 Email               Contacts    Yearly_equip_failure
##              1.002509               1.002805                1.003346
##         MonthlyCharge       Bandwidth_GB_Year           Area_Suburban
##              1.801716               1.726602                1.347056
##            Area_Urban        Marital_Married `Marital_Never Married`
##              1.349011               1.552133                1.553358
##     Marital_Separated        Marital_Widowed             Gender_Male
##              1.576612               1.593950                1.029128
##      Gender_Nonbinary     `Contract_One year`     `Contract_Two Year`
##              1.026082               1.234036                1.243599
```

```
# perform feature selection for reduced model [In-text citation: (Bobbitt, 2019)]
backward_stepwise <- step(model_initial, direction = "backward", scope = formula(model_initial), trace =
backward_stepwise$anova
```

```
##                         Step Df    Deviance Resid. Df Resid. Dev      AIC
## 1                            NA          NA      9981   5810.871 5848.871
## 2          - Marital_Married  1 0.04309533      9982   5810.914 5846.914
## 3            - Area_Suburban  1 0.05679421      9983   5810.971 5844.971
## 4                    - Email  1 0.05646995      9984   5811.027 5843.027
## 5       - Outage_sec_perweek  1 0.07788031      9985   5811.105 5841.105
## 6     - Yearly_equip_failure  1 0.45757051      9986   5811.563 5839.563
## 7  - `Marital_Never Married`  1 0.57863487      9987   5812.141 5838.141
## 8               - Area_Urban  1 0.58459298      9988   5812.726 5836.726
## 9                   - Income  1 0.89220638      9989   5813.618 5835.618
## 10        - Gender_Nonbinary  1 1.13155768      9990   5814.750 5834.750
## 11                - Contacts  1 1.71438755      9991   5816.464 5834.464
```

```
backward_stepwise$coefficients
```

```
##         (Intercept)                 Age       MonthlyCharge   Bandwidth_GB_Year
##        -5.237638096        -0.003321033         0.042793708        -0.000983959
##   Marital_Separated     Marital_Widowed         Gender_Male `Contract_One year`
##         0.142065160         0.229705586         0.263012218        -2.482158780
## `Contract_Two Year`
##        -2.574063034
```

```
# select columns for reduced MLR model
churn_reduced <- churn_initial %>%
  select(
    Churn,
    Age,
    MonthlyCharge,
    Bandwidth_GB_Year,
```

```
    Marital_Separated,
    Marital_Widowed,
    Gender_Male,
    `Contract_One year`,
    `Contract_Two Year`
  )
```

D3, Reduced MLR Model

The initial multiple regression model contained 18 explanatory variables, resulting in an AIC of 5849.9. The reduced model contained eight explanatory variables after checking for multicollinearity and performing backward stepwise elimination. The resulting reduced model produced essentially the same results with fewer features. The AIC of the reduced model was 5834.5. The reduced model is a better fit because it has a lower AIC score.

Initial Model

```
summary(model_initial)
```

```
##
## Call:
## glm(formula = Churn ~ ., family = "binomial", data = churn_initial)
##
## Coefficients:
##                           Estimate   Std. Error z value            Pr(>|z|)
## (Intercept)           -5.239474179  0.273295573 -19.171 < 0.0000000000000002
## Age                   -0.003343172  0.001617020  -2.067             0.038688
## Income                 0.000001111  0.000001173   0.947             0.343430
## Outage_sec_perweek    -0.003078095  0.011096911  -0.277             0.781486
## Email                 -0.002688606  0.010982837  -0.245             0.806611
## Contacts               0.043775723  0.033145933   1.321             0.186603
## Yearly_equip_failure  -0.035708333  0.052683157  -0.678             0.497902
## MonthlyCharge          0.042844985  0.001087560  39.396 < 0.0000000000000002
## Bandwidth_GB_Year     -0.000984748  0.000023789 -41.394 < 0.0000000000000002
## Area_Suburban         -0.019601112  0.081497620  -0.241             0.809934
## Area_Urban             0.044024304  0.081334107   0.541             0.588316
## Marital_Married        0.021832689  0.105165345   0.208             0.835539
## `Marital_Never Married` -0.060031421  0.105268991  -0.570             0.568497
## Marital_Separated      0.128598232  0.103552725   1.242             0.214287
## Marital_Widowed        0.219350560  0.102637026   2.137             0.032586
## Gender_Male            0.251985270  0.067212986   3.749             0.000178
## Gender_Nonbinary      -0.231771045  0.226721725  -1.022             0.306653
## `Contract_One year`   -2.485743783  0.100948927 -24.624 < 0.0000000000000002
## `Contract_Two Year`   -2.575025647  0.097932783 -26.294 < 0.0000000000000002
##
## (Intercept)           ***
## Age                   *
## Income
## Outage_sec_perweek
## Email
## Contacts
```

```
## Yearly_equip_failure
## MonthlyCharge            ***
## Bandwidth_GB_Year        ***
## Area_Suburban
## Area_Urban
## Marital_Married
## `Marital_Never Married`
## Marital_Separated
## Marital_Widowed          *
## Gender_Male              ***
## Gender_Nonbinary
## `Contract_One year`      ***
## `Contract_Two Year`      ***
## ---
## 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:  5810.9  on 9981  degrees of freedom
## AIC: 5848.9
##
## Number of Fisher Scoring iterations: 6
```

Reduced Model

```
model_reduced <- glm(Churn ~ ., data = churn_reduced, family = "binomial")
summary(model_reduced)
```

```
##
## Call:
## glm(formula = Churn ~ ., family = "binomial", data = churn_reduced)
##
## Coefficients:
##                        Estimate  Std. Error z value             Pr(>|z|)
## (Intercept)         -5.23763810  0.18920713 -27.682 < 0.0000000000000002 ***
## Age                 -0.00332103  0.00161415  -2.057              0.03964 *
## MonthlyCharge        0.04279371  0.00108549  39.423 < 0.0000000000000002 ***
## Bandwidth_GB_Year   -0.00098396  0.00002375 -41.425 < 0.0000000000000002 ***
## Marital_Separated    0.14206516  0.08541851   1.663              0.09628 .
## Marital_Widowed      0.22970559  0.08428079   2.725              0.00642 **
## Gender_Male          0.26301222  0.06635820   3.964            0.0000739 ***
## `Contract_One year` -2.48215878  0.10079374 -24.626 < 0.0000000000000002 ***
## `Contract_Two Year` -2.57406303  0.09780423 -26.319 < 0.0000000000000002 ***
## ---
## 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:  5816.5  on 9991  degrees of freedom
## AIC: 5834.5
##
## Number of Fisher Scoring iterations: 6
```

E1, Model Comparison

The AIC score was chosen as an evaluation metric to compare the two models. In the initial model with 18 explanatory variables, the AIC score was 5848.9. After reducing the number of explanatory variables to eight through feature selection, the AIC score was 5834.5. The lowest AIC score between models is considered the best, so the reduced model is the best fit in this case.

E2, Model Output & Calculations

Below is the output of the reduced model, which includes a confusion matrix and accuracy metrics. Calculations are then provided to arrive at the same accuracy output.

```r
summary(model_reduced)
```

```
##
## Call:
## glm(formula = Churn ~ ., family = "binomial", data = churn_reduced)
##
## Coefficients:
##                        Estimate  Std. Error z value             Pr(>|z|)
## (Intercept)         -5.23763810  0.18920713 -27.682 < 0.0000000000000002 ***
## Age                 -0.00332103  0.00161415  -2.057              0.03964 *
## MonthlyCharge        0.04279371  0.00108549  39.423 < 0.0000000000000002 ***
## Bandwidth_GB_Year   -0.00098396  0.00002375 -41.425 < 0.0000000000000002 ***
## Marital_Separated    0.14206516  0.08541851   1.663              0.09628 .
## Marital_Widowed      0.22970559  0.08428079   2.725              0.00642 **
## Gender_Male          0.26301222  0.06635820   3.964            0.0000739 ***
## `Contract_One year` -2.48215878  0.10079374 -24.626 < 0.0000000000000002 ***
## `Contract_Two Year` -2.57406303  0.09780423 -26.319 < 0.0000000000000002 ***
## ---
## 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:  5816.5  on 9991  degrees of freedom
## AIC: 5834.5
##
## Number of Fisher Scoring iterations: 6
```

```r
# create confusion matrix for reduced model
response_actual_reduced <- churn_reduced$Churn
response_predicted_reduced <- round(fitted(model_reduced))
outcomes_reduced <- table(response_predicted_reduced, response_actual_reduced)
outcomes_reduced
```

```
##                           response_actual_reduced
## response_predicted_reduced    0    1
##                          0 6779  775
##                          1  571 1875
```

```r
confusion_reduced <- conf_mat(outcomes_reduced)
summary(confusion_reduced, event_level = "second")
```

```
## # A tibble: 13 x 3
##    .metric              .estimator .estimate
##    <chr>                <chr>          <dbl>
##  1 accuracy             binary         0.865
##  2 kap                  binary         0.646
##  3 sens                 binary         0.708
##  4 spec                 binary         0.922
##  5 ppv                  binary         0.767
##  6 npv                  binary         0.897
##  7 mcc                  binary         0.647
##  8 j_index              binary         0.630
##  9 bal_accuracy         binary         0.815
## 10 detection_prevalence binary         0.245
## 11 precision            binary         0.767
## 12 recall               binary         0.708
## 13 f_meas               binary         0.736
```

While the model accuracy can be retrieved using code, to calculate it manually, the true positives and true negatives are summed and divided by the entire population. Based on the confusion matrix above, the calculation would be (6779 + 1875)/ (6779+1875+775+571), which comes to the same 0.865.

E3, Supporting Code

The code used in the analysis has been included in the submission in a .R file.

F1, Analysis Results

The regression equation for the reduced model can be found below:

P(Churn) = -5.2376 – 0.0033(Age) + 0.0427(MonthlyCharge) – 0.0009(Bandwidth_GB_Year) + 0.1421(Marital_Separated) + 0.2297(Marital_Widowed) + 0.2630(Gender_Male) – 2.4821(Contract_One_Year) – 2.5740(Contract_Two_Year)

The intercept, -5.2376, represents the log odds of churning if all explanatory variables were zero. A unit change in age would result in a -0.0033 decrease in the log odds of churning. A unit change in monthly charges would result in a 0.0427 increase in the log odds of churning. A unit change in bandwidth GB used would result in a 0.0009 decrease in log odds of churning. If a customer had a marital status of separated, it would result in a 0.1421 increase in the log odds of churning. If a customer had a marital status of widowed, it would result in a 0.2297 increase in the log odds of churning. If the customer's gender were male, it would result in a 0.2630 increase in the log odds of churning. A contract length of one year would result in a 2.4821 decrease in the log odds of churning. A contract length of two years would result in a 2.5740 decrease in the log odds of churning. (Bobbitt, October 27, 2020). The reduced model rated out well in terms of accuracy. There were 6,779 true negatives and 1,875 true positives, which resulted in an accuracy of 0.865. The reduced model also had a lower AIC score than the initial model, indicating it is better. All but one of the explanatory variables are statistically significant based on the p-values.

In terms of practical significance, the reduced model could benefit the company. It can help predict which customers are at risk of discontinuing service. The company could use this information to preemptively address areas of concern with the customer to make it more likely that they retain services. Logically, many of these variables would lead to customer churn. If a customer's monthly charge is too high, they may seek another provider. They may not feel they need the same service if they do not use much data. One and two-year contracts make it more likely that a customer will stay with the company.

There are certain limitations of the data analysis. First, many of the explanatory variables are customer-reported variables. It is possible the customer had not provided accurate information to the company. That could lead to misleading or inaccurate results. From a data standpoint, it appeared that some of the data had been cleaned beforehand based on the distribution of the variables. The age variable was a good example of this. If another analyst had imputed missing values with the mean or median, the results may not be as meaningful.

F2, Course of Action

The initial research question was what factors can lead to customer churn, and this analysis has identified several such factors. Having a one or two-year contract reduces the odds of churn, higher monthly charges increase the odds of churn, having a marital status of separated or widowed increases the odds of churn, increases in a customer's age reduce the odds of churn, and being male increases the odds of churn. Most of these make logical sense, and it helps to have the regression back it up.

This model can be helpful for the company. When identifying active customers with these characteristics, the company can take steps to ensure they stay with the company. More variables could be looked at to see if they added any additional value to the model. It is possible that the analysis did not cast a wide enough net when choosing variables for the initial model.

G, Panopto Video

I created a Panopto video recording that covered the execution of the code, a comparison of the initial and reduced models, and an interpretation of the coefficients. The video link can be found in the submission.

H, Sources for Code

Bobbitt, Z. (April 27, 2019). A complete guide to stepwise regression in R. Statology. Retrieved December 8, 2024, from (https://statology.org/stepwise-regression-r/)

Bobbitt, Z. (July 30, 2021). How to turn off scientific notation in R (with examples). Statology. Retrieved December 8, 2024, from (https://www.statology.org/turn-off-scientific-notation-in-r/)

Kaplan, J. (November 28, 2020). Making dummy variables with dummy_cols(). fastDummies. Retrieved December 8, 2024, from (https://jacobkap.github.io/fastDummies/articles/making-dummy-variables.html)

Schork, J. (n.d.). Correlation matrix in R. Statistics Globe. Retrieved December 10, 2024, from (https://statisticsglobe.com/correlation-matrix-in-r)

Soetewey, A. (Aug 11, 2020). Outlier detection in R. Stats and R. Retrieved December 8, 2024, from (https://statsandr.com/blog/outliers-detection-in-r/)

Tierney, N. (n.d.). Dealing with Missing Data in R [MOOC]. DataCamp. (https://app.datacamp.com/learn/courses/dealing-with-missing-data-in-r)

WGU College of Information Technology (n.d.). D208 Predictive Modeling Webinar [Panopto Video]. Western Governors University. (https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=567da34c-96e3-44c7-a160-ae3100f9433d)

WGU College of Information Technology (n.d.). Getting Started with Duplicates [PowerPoint slides]. Western Governors University. (https://westerngovernorsuniversity.sharepoint.com/sites/DataScienceTeam/Shared Documents/Forms/AllItems.aspx?id=%2Fsites%2FDataScienceTeam%2FShared%20Documents%2FGraduate%20

I, Sources for Content

Bobbitt, Z. (April 27, 2019). A complete guide to stepwise regression in R. Statology. Retrieved December 8, 2024, from (https://statology.org/stepwise-regression-r/)

Bobbitt, Z. (October 13, 2020). The 6 Assumptions of Logistic Regression. Statology. Retrieved December 21, 2024, from (https://www.statology.org/assumptions-of-logistic-regression)

Bobbitt, Z. (October 27, 2020). Introduction to logistic regression. Statology. Retrieved December 21, 2024, from (https://www.statology.org/logistic-regression)

Kaplan, J. (November 28, 2020). Making dummy variables with dummy_cols(). fastDummies. Retrieved December 8, 2024, from (https://jacobkap.github.io/fastDummies/articles/making-dummy-variables.html)

WGU College of Information Technology (n.d.). D208 Predictive Modeling Webinar [Panopto Video]. Western Governors University. (https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=567da34c-96e3-44c7-a160-ae3100f9433d)