C744 Data Mining and Analytics 2

```
library(xlsx)
library(ggplot2)
library(plyr)
library(corrplot)
library(gridExtra)
library(MASS)
library(effects)
library(FactoMineR)
library(factoextra)
```

Tool selection

A. Why R?

R has many benefits that make it an ideal choice for this analysis. R is open source with a large community for support. This allows for the creation of community packages like FactoMineR, where developers who use the software write packages designed to do their work (Tufféry, 2011, pg 124). FactoMineR and ggplot2 are two packages that will be used in this analysis designed under the open source GNU-GPL license. R is also a statistical language, designed specifically for this type of analysis with visualization libraries like ggplot2 available (Data Flair, 2019).

B. Goal of analysis

The goal of this analysis will be to find potential indicators that explain why customers leave a telecommunications company for their cable competitors. This will be accomplished by identifying what customers who leave have in common and how they differ from those that don't. These factors will be combined, ranked and scored to determine which have the strongest prediction power. As exra value to the prediction, variables that describe customers will be reviewing, increasing the knowledge of this population's relationship to the telco.

C. Which methods?

A summary of the dataset supplied is below.

```
custData <- read.table('initial data set.csv', header=TRUE, sep=',')
str(custData)
```

```
## 'data.frame':    7043 obs. of  21 variables:
##  $ customerID      : chr  "7590-VHVEG" "5575-GNVDE" "3668-QPYBK" "7795-
CFOCW" ...
##  $ gender          : chr  "Female" "Male" "Male" "Male" ...
##  $ SeniorCitizen   : int  0 0 0 0 0 0 0 0 0 0 ...
```

```
##  $ Partner         : chr  "Yes" "No" "No" "No" ...
##  $ Dependents      : chr  "No" "No" "No" "No" ...
##  $ tenure          : int  1 34 2 45 2 8 22 10 28 62 ...
##  $ PhoneService    : chr  "No" "Yes" "Yes" "No" ...
##  $ MultipleLines   : chr  "No phone service" "No" "No" "No phone service"
...
##  $ InternetService : chr  "DSL" "DSL" "DSL" "DSL" ...
##  $ OnlineSecurity  : chr  "No" "Yes" "Yes" "Yes" ...
##  $ OnlineBackup    : chr  "Yes" "No" "Yes" "No" ...
##  $ DeviceProtection: chr  "No" "Yes" "No" "Yes" ...
##  $ TechSupport     : chr  "No" "No" "No" "Yes" ...
##  $ StreamingTV     : chr  "No" "No" "No" "No" ...
##  $ StreamingMovies : chr  "No" "No" "No" "No" ...
##  $ Contract        : chr  "Month-to-month" "One year" "Month-to-month"
"One year" ...
##  $ PaperlessBilling: chr  "Yes" "No" "Yes" "No" ...
##  $ PaymentMethod   : chr  "Electronic check" "Mailed check" "Mailed check"
"Bank transfer (automatic)" ...
##  $ MonthlyCharges  : num  29.9 57 53.9 42.3 70.7 ...
##  $ TotalCharges    : num  29.9 1889.5 108.2 1840.8 151.7 ...
##  $ Churn           : chr  "No" "No" "Yes" "No" ...
```

Most of these variables will be transformed into binary factor variables, however, there are other continuous variables that will be examined. The dependent variable will be churn as this is a direct representation of whether a customer remains with this organization or not. Analysis methods chosen will need to be suited to mixed data.

Logistic regression will be used as a non-descriptive method for this analysis. Binary logistic regression is appropriate for a prediction of a binary variable based on one or more continuous or binary variables (Tufféry, 2011). For this analysis, our target variable will be the binomial factor variable Churn based on a set of other variables of mixed types.

Multiple correspondence analysis (MCA) will be used as a descriptive method for this analysis. MCA is a type of factor analysis where the goal is reduce the dimensions of a problem while retaining as much information as possible (Tuffery, 2011). MCA is especially useful when an analysis is needed of multiple qualitative variables.

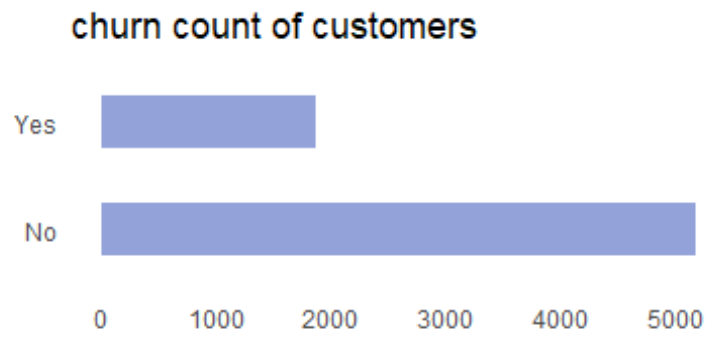
Data exploration

D. Target variable

The target variable of this analysis is churn. As the goal to to explain customer attrition, this variable is the best indicator of a customer's status with the company.

```
ggplot(custData, aes(y=Churn)) +
  ggtitle('churn count of customers') +
  geom_bar(aes(x=..count..), width=0.5, fill='#2748b3', alpha=0.5) +
  ylab('') + xlab('') +
```

```
  theme_minimal() +
  theme(panel.grid.major=element_blank(), panel.grid.minor=element_blank())
```

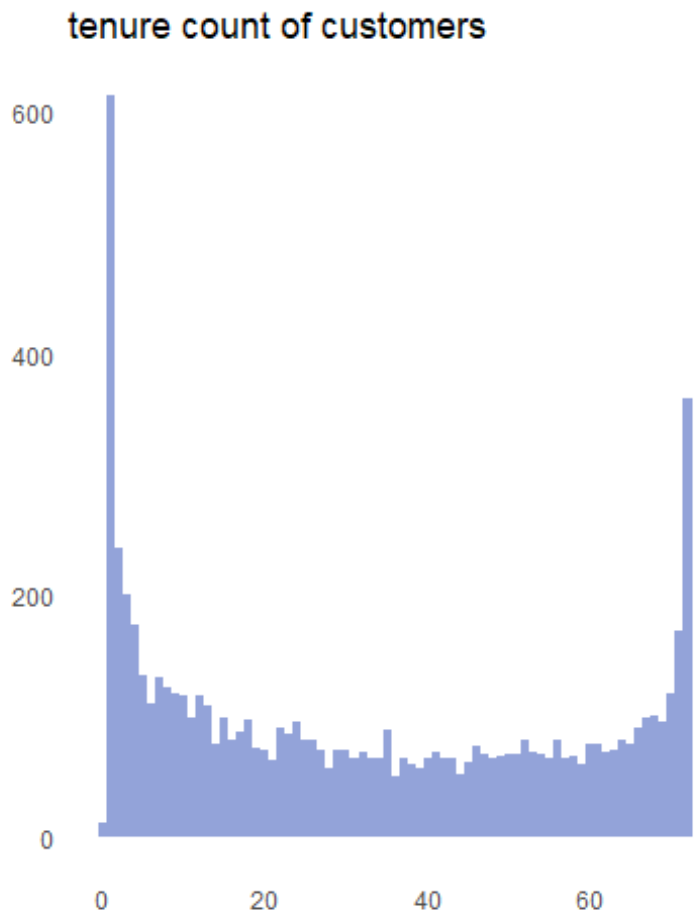


Churn is a binomial factor variable with yes and no as levels.

E. Independent predictor variable

One of the independent predictor variables available for this analysis is tenure. This is the number of months a customer has been active with the company. Tenure is a continuous, quantitative variable.

```
ggplot(custData, aes(x=tenure)) +
  geom_histogram(binwidth=1, fill='#2748b3', alpha=0.5) +
  ggtitle('tenure count of customers') +
  ylab('') + xlab('') +
  theme_minimal() +
  theme(panel.grid.major=element_blank(), panel.grid.minor=element_blank())
```



Tenure has a bi modal distribution and will likely need to be binned during the analysis.

F. Data manipulation goals

This data set will need to be cleaned and transformed to complete this analysis.

Cleaning will start by searching for aberrant or missing values. Decisions will be made on how to proceed based on which variables have missing data and how many records have those variables missing. Preference will be given to ignoring missing values when possible, then deleting records if a small enough percentage of them have missing values. As a last resort, inference will be used to estimate the missing or aberrant values.

Many of the variables will also need to be transformed to be used in the methods selected. For example, “no internet service” and “no” effectively have the same meaning in the OnlineBackup variable. This also happens with MultipleLines and “no phone service”. The variables where this can be applied will be transformed to binary factor variables with yes/no values.

New variables will be created as transformations of existing variables. For example, tenure will be binned. Some variables may also be removed if they are found to be strongly correlated with other variables.

G. Statistical identity

This data set includes many different data types. customerID serves as a unique identifier for each record.

```
sum(duplicated(custData$customerID))
```

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

Churn will be the dependent variable. As stated previously, Churn is a binomial factor variable with two levels. The value of Churn is the phenomenon to be predicted.

The data include three independent, continuous variables: MonthlyCharges, TotalCharges and Tenure (Tufféry, 2011)

Finally, the remaining 16 variables are qualitative categorical variables with a varying number of levels. Each of these and their unique values are below.

```
discCat <- custData[,c(2:5, 7:18)]
sapply(discCat, unique)
```

```
## $gender
## [1] "Female" "Male"
##
## $SeniorCitizen
## [1] 0 1
##
## $Partner
## [1] "Yes" "No"
##
## $Dependents
## [1] "No"  "Yes"
##
## $PhoneService
## [1] "No"  "Yes"
##
## $MultipleLines
## [1] "No phone service" "No"               "Yes"
##
## $InternetService
## [1] "DSL"         "Fiber optic" "No"
##
## $OnlineSecurity
## [1] "No"                  "Yes"                 "No internet service"
##
## $OnlineBackup
## [1] "Yes"                 "No"                  "No internet service"
##
## $DeviceProtection
## [1] "No"                  "Yes"                 "No internet service"
##
```

```
## $TechSupport
## [1] "No"                  "Yes"                 "No internet service"
##
## $StreamingTV
## [1] "No"                  "Yes"                 "No internet service"
##
## $StreamingMovies
## [1] "No"                  "Yes"                 "No internet service"
##
## $Contract
## [1] "Month-to-month" "One year"       "Two year"
##
## $PaperlessBilling
## [1] "Yes" "No"
##
## $PaymentMethod
## [1] "Electronic check"          "Mailed check"
## [3] "Bank transfer (automatic)" "Credit card (automatic)"
```

H. Clean the data

The first step in cleaning will be to identify variables with missing values.

```
sapply(custData, function(x) sum(is.na(x)))
```

```
##       customerID           gender    SeniorCitizen          Partner
##                0                0                0                0
##       Dependents           tenure     PhoneService    MultipleLines
##                0                0                0                0
##  InternetService   OnlineSecurity     OnlineBackup DeviceProtection
##                0                0                0                0
##      TechSupport      StreamingTV  StreamingMovies         Contract
##                0                0                0                0
## PaperlessBilling    PaymentMethod   MonthlyCharges     TotalCharges
##                0                0                0               11
##            Churn
##                0
```

Only TotalCharges has missing values and only in 0.15% of the total observations. If this variable is not removed by the end of the data cleaning, these observations will be removed.

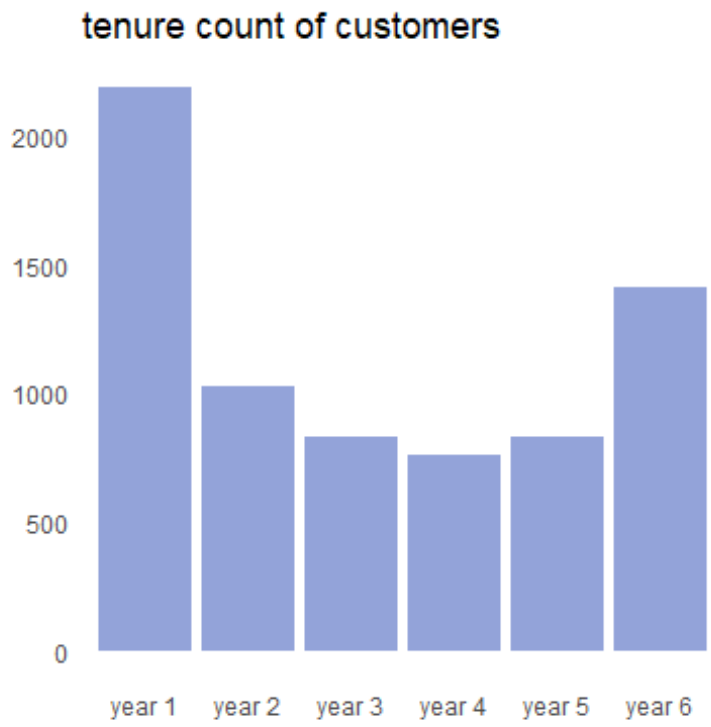
“No phone service” and “No internet service” will need to be converted to just “No”. These are functionally equivalent and combining them will limit the levels on these factors.

```
custData[custData=="No phone service"] <- "No"
custData[custData=="No internet service"] <- "No"
discCat <- custData[,c(2:5, 7:18)]
sapply(discCat, unique)
```

```
## $gender
## [1] "Female" "Male"
##
## $SeniorCitizen
## [1] 0 1
##
## $Partner
## [1] "Yes" "No"
##
## $Dependents
## [1] "No"  "Yes"
##
## $PhoneService
## [1] "No"  "Yes"
##
## $MultipleLines
## [1] "No"  "Yes"
##
## $InternetService
## [1] "DSL"         "Fiber optic" "No"
##
## $OnlineSecurity
## [1] "No"  "Yes"
##
## $OnlineBackup
## [1] "Yes" "No"
##
## $DeviceProtection
## [1] "No"  "Yes"
##
## $TechSupport
## [1] "No"  "Yes"
##
## $StreamingTV
## [1] "No"  "Yes"
##
## $StreamingMovies
## [1] "No"  "Yes"
##
## $Contract
## [1] "Month-to-month" "One year"       "Two year"
##
## $PaperlessBilling
## [1] "Yes" "No"
##
## $PaymentMethod
## [1] "Electronic check"          "Mailed check"
## [3] "Bank transfer (automatic)" "Credit card (automatic)"
```

Tenure can be binned by years. This will limit the levels of this factor while maintaining the spread.

```
min(custData$tenure)

## [1] 0

max(custData$tenure)

## [1] 72

tenureBins <- function(tenure){
    if (tenure <= 12)
      {return('year 1')}
    if (tenure <= 24)
      {return('year 2')}
    if (tenure <= 36)
      {return('year 3')}
    if (tenure <= 48)
      {return('year 4')}
    if (tenure <= 60)
      {return('year 5')}
    return('year 6')
}

custData$tenureBin <- sapply(custData$tenure, tenureBins)

## (Li, 2017)

ggplot(custData, aes(x=tenureBin)) +
  geom_histogram(fill='#2748b3', alpha=0.5, stat="count") +
  ggtitle('tenure count of customers') +
  ylab('') + xlab('') +
  theme_minimal() +
  theme(panel.grid.major=element_blank(), panel.grid.minor=element_blank())
```

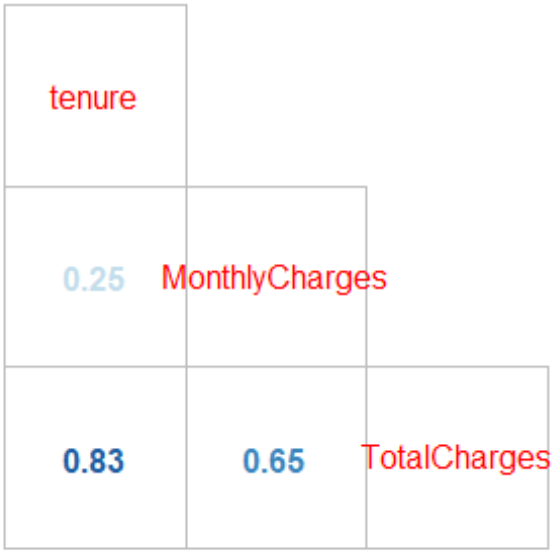


Although not strictly necessary, SeniorCitizen can also be converted to yes/no instead of 0/1 to match the other factor variables.

```
custData$SeniorCitizen[custData$SeniorCitizen==0] <- "No"
custData$SeniorCitizen[custData$SeniorCitizen==1] <- "Yes"
```

Removing strongly correlated variables will ensure that these attributes don't have an undue weight in any models generated.

```
numerics <- sapply(custData, is.numeric)
custData <- custData[complete.cases(custData), ]
matrix <- cor(custData[,numerics])
# (li, 2017)
corrplot(matrix, tl.pos='d', cl.pos='n', method='number', type='lower')
```



The correlation between tenure and TotalCharges is more than 0.8 and very risky (Tufféry, 2011). As TotalCharges also has missing values, it will be removed in favor of tenure.

The next step in cleaning this data will be to remove the customerID. It won't be used as identifying an individual customer will not be necessary. Tenure was transformed into a factor variable with years as levels and is also no longer needed.

```
custData <- within(custData, rm(customerID, tenure, TotalCharges))
```

The last step will be to convert the various factor variables into factors so that R will handle them appropriately.

```
factorCol <- c(1:16,18,19)
custData[factorCol] <- lapply(custData[factorCol], as.factor)
```

Data Analysis

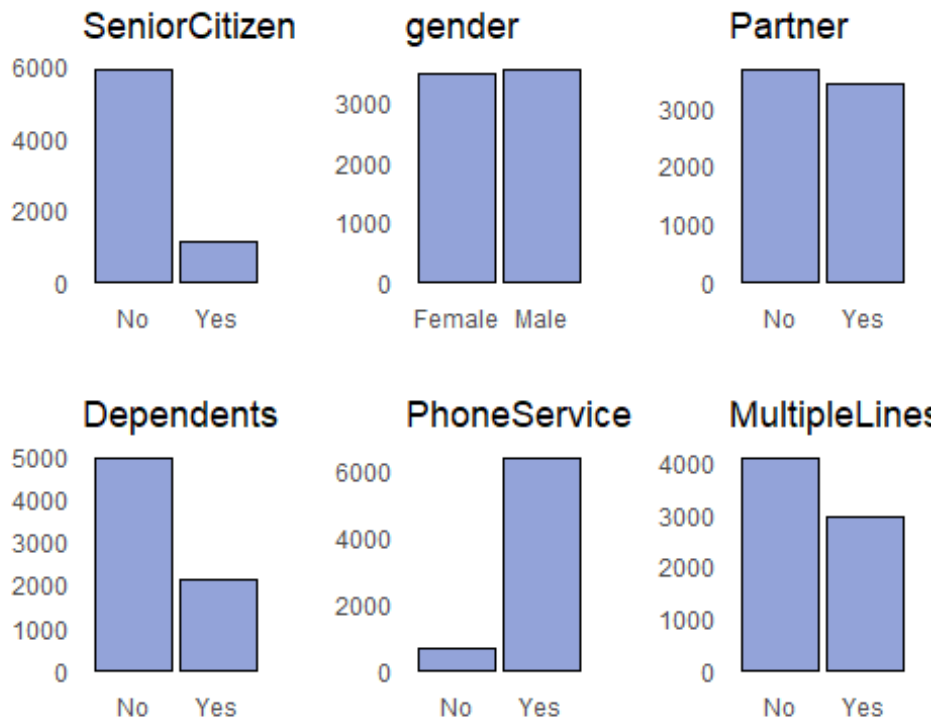
I. Univariate variable distributions

Each of the remaining variables will be visualized to identify their distribution. Histograms and density plots show a variable's distribution well. Churn and tenure have both already been visualized.

```
seniorCitizenPlot <- ggplot(custData, aes(x=SeniorCitizen)) +
      ggtitle('SeniorCitizen') +
      geom_histogram(stat="count", fill='#2748b3', color="black", alpha=0.5)
```

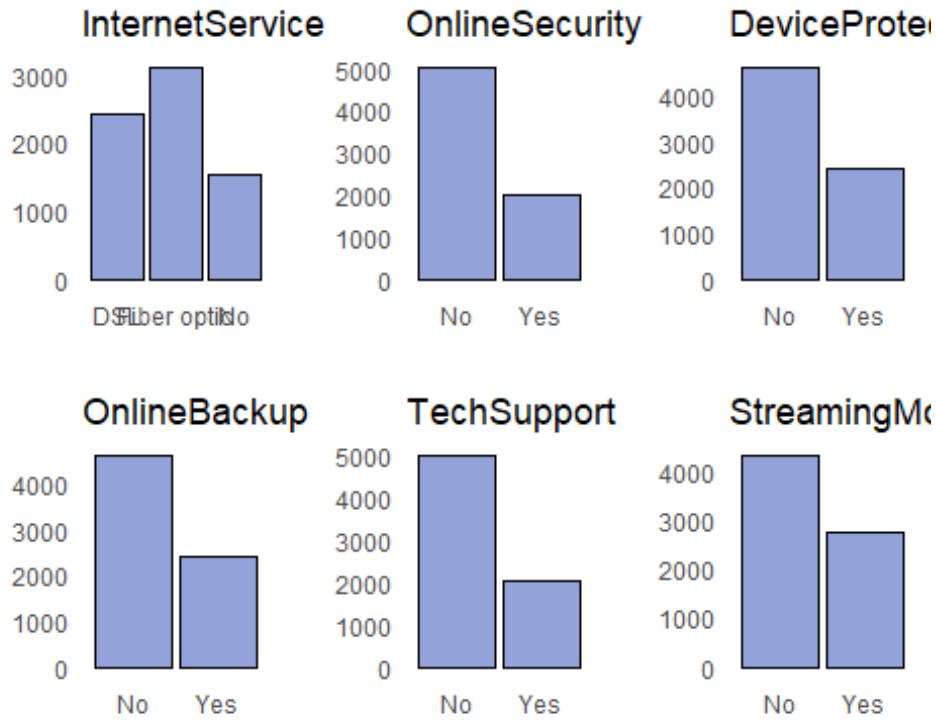
```
+
      ylab('') + xlab('') +
      theme_minimal() +
      theme(panel.grid.major=element_blank(),
panel.grid.minor=element_blank())

genderPlot <- ggplot(custData, aes(x=gender)) +
      ggtitle('gender') +
      geom_histogram(stat="count", fill='#2748b3', color="black", alpha=0.5)
+
      ylab('') + xlab('') +
      theme_minimal() +
      theme(panel.grid.major=element_blank(),
panel.grid.minor=element_blank())

partnerPlot <- ggplot(custData, aes(x=Partner)) +
      ggtitle('Partner') +
      geom_histogram(stat="count", fill='#2748b3', color="black", alpha=0.5)
+
      ylab('') + xlab('') +
      theme_minimal() +
      theme(panel.grid.major=element_blank(),
panel.grid.minor=element_blank())

dependentsPlot <- ggplot(custData, aes(x=Dependents)) +
      ggtitle('Dependents') +
      geom_histogram(stat="count", fill='#2748b3', color="black", alpha=0.5)
+
      ylab('') + xlab('') +
      theme_minimal() +
      theme(panel.grid.major=element_blank(),
panel.grid.minor=element_blank())

phoneServicePlot <- ggplot(custData, aes(x=PhoneService)) +
      ggtitle('PhoneService') +
      geom_histogram(stat="count", fill='#2748b3', color="black", alpha=0.5)
+
      ylab('') + xlab('') +
      theme_minimal() +
      theme(panel.grid.major=element_blank(),
panel.grid.minor=element_blank())

multipleLinesPlot <- ggplot(custData, aes(x=MultipleLines)) +
      ggtitle('MultipleLines') +
      geom_histogram(stat="count", fill='#2748b3', color="black", alpha=0.5)
+
      ylab('') + xlab('') +
      theme_minimal() +
      theme(panel.grid.major=element_blank(),
```

```
panel.grid.minor=element_blank())

grid.arrange(seniorCitizenPlot, genderPlot, partnerPlot,
             dependentsPlot, phoneServicePlot, multipleLinesPlot,
             ncol=3)
```



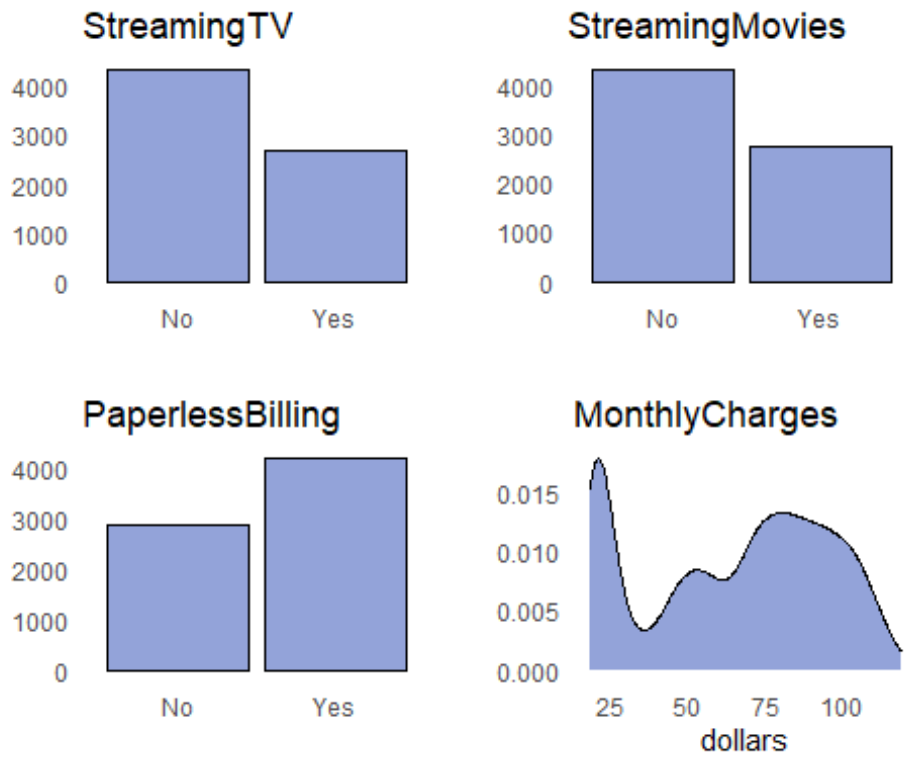

```
internetServicePlot <- ggplot(custData, aes(x=InternetService)) +
      ggtitle('InternetService') +
      geom_histogram(stat="count", fill='#2748b3', color="black", alpha=0.5)
+
      ylab('') + xlab('') +
      theme_minimal() +
      theme(panel.grid.major=element_blank(),
panel.grid.minor=element_blank())

onlineSecurityPlot <- ggplot(custData, aes(x=OnlineSecurity)) +
      ggtitle('OnlineSecurity') +
      geom_histogram(stat="count", fill='#2748b3', color="black", alpha=0.5)
+
      ylab('') + xlab('') +
      theme_minimal() +
      theme(panel.grid.major=element_blank(),
panel.grid.minor=element_blank())

deviceProtectionPlot <- ggplot(custData, aes(x=DeviceProtection)) +
      ggtitle('DeviceProtection') +
```

```
      geom_histogram(stat="count", fill='#2748b3', color="black", alpha=0.5)
+
      ylab('') + xlab('') +
      theme_minimal() +
      theme(panel.grid.major=element_blank(),
panel.grid.minor=element_blank())

onlineBackupPlot <- ggplot(custData, aes(x=OnlineBackup)) +
      ggtitle('OnlineBackup') +
      geom_histogram(stat="count", fill='#2748b3', color="black", alpha=0.5)
+
      ylab('') + xlab('') +
      theme_minimal() +
      theme(panel.grid.major=element_blank(),
panel.grid.minor=element_blank())

techSupportPlot <- ggplot(custData, aes(x=TechSupport)) +
      ggtitle('TechSupport') +
      geom_histogram(stat="count", fill='#2748b3', color="black", alpha=0.5)
+
      ylab('') + xlab('') +
      theme_minimal() +
      theme(panel.grid.major=element_blank(),
panel.grid.minor=element_blank())

streamingMoviesPlot <- ggplot(custData, aes(x=StreamingMovies)) +
      ggtitle('StreamingMovies') +
      geom_histogram(stat="count", fill='#2748b3', color="black", alpha=0.5)
+
      ylab('') + xlab('') +
      theme_minimal() +
      theme(panel.grid.major=element_blank(),
panel.grid.minor=element_blank())

grid.arrange(internetServicePlot, onlineSecurityPlot, deviceProtectionPlot,
             onlineBackupPlot, techSupportPlot, streamingMoviesPlot,
             ncol=3)
```




```
streamingTVplot <- ggplot(custData, aes(x=StreamingTV)) +
      ggtitle('StreamingTV') +
      geom_histogram(stat="count", fill='#2748b3', color="black", alpha=0.5)
+
      ylab('') + xlab('') +
      theme_minimal() +
      theme(panel.grid.major=element_blank(),
panel.grid.minor=element_blank())

paperlessBillingplot <- ggplot(custData, aes(x=PaperlessBilling)) +
      ggtitle('PaperlessBilling') +
      geom_histogram(stat="count", fill='#2748b3', color="black", alpha=0.5)
+
      ylab('') + xlab('') +
      theme_minimal() +
      theme(panel.grid.major=element_blank(),
panel.grid.minor=element_blank())

monthlyChargesPlot <- ggplot(custData, aes(x=MonthlyCharges)) +
  ggtitle("MonthlyCharges") +
  xlab("dollars") + ylab("") +
  geom_density(fill='#2748b3', alpha=0.5) +
  theme_minimal() +
  theme(panel.grid.major=element_blank(), panel.grid.minor=element_blank())

grid.arrange(streamingTVplot, streamingMoviesPlot, paperlessBillingplot,
```

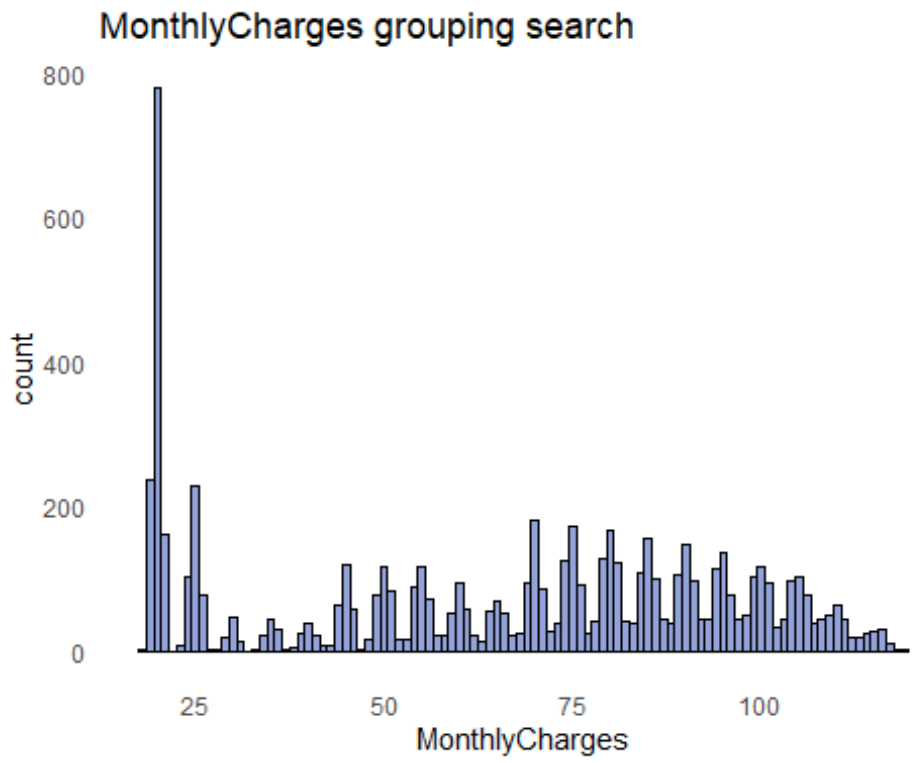
```
            monthlyChargesPlot,
            ncol=2)
```




```
summary(custData$MonthlyCharges)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   18.25   35.59   70.35   64.80   89.86  118.75
```

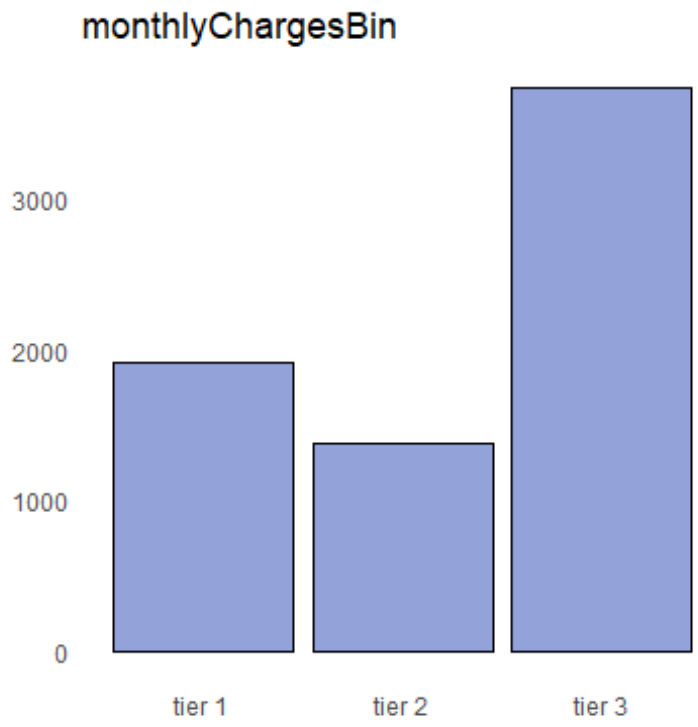
For MonthlyCharges, there are no outliers. However, there are 3 distinct groups. It should be possible to convert MonthlyCharges into a factor variable by using a histogram to find the appropriate group boundaries.

```
ggplot(custData, aes(x=MonthlyCharges)) +
  geom_histogram(binwidth = 1, fill='#2748b3', color="black", alpha=0.5) +
  ggtitle("MonthlyCharges grouping search") +
  theme_minimal() +
  theme(panel.grid.major=element_blank(), panel.grid.minor=element_blank())
```

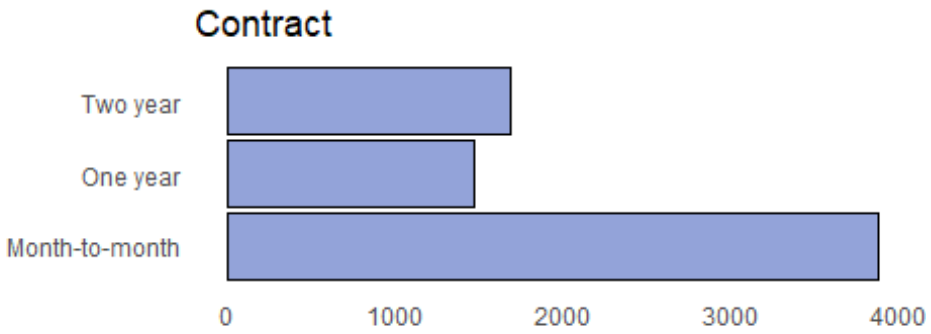


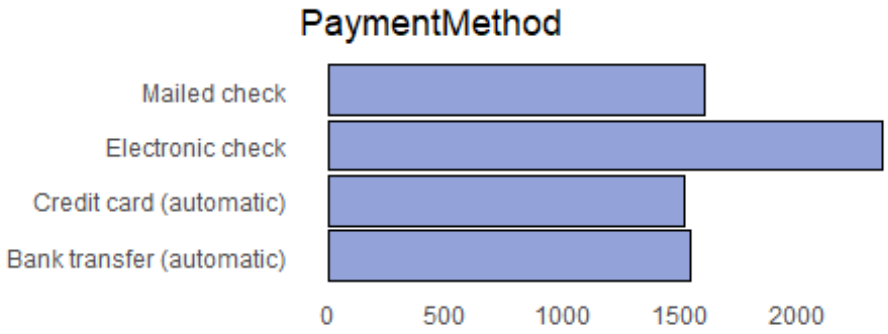
The groupings break around $44 and $69.

```
mcBins <- function(MonthlyCharges){
    if (MonthlyCharges < 44)
      {return('tier 1')}
    if (MonthlyCharges < 69)
      {return('tier 2')}
    return('tier 3')
}

custData$monthlyChargesBin <- sapply(custData$MonthlyCharges, mcBins)
custData$monthlyChargesBin <- as.factor(custData$monthlyChargesBin)
custData <- within(custData, rm(MonthlyCharges))

ggplot(custData, aes(x=monthlyChargesBin)) +
  ggtitle('monthlyChargesBin') +
  geom_histogram(stat="count", fill='#2748b3', color="black", alpha=0.5) +
  ylab('') + xlab('') +
  theme_minimal() +
  theme(panel.grid.major=element_blank(), panel.grid.minor=element_blank())
```




```
contractPlot <- ggplot(custData, aes(x=Contract)) +
      ggtitle('Contract') +
      geom_histogram(stat="count", fill='#2748b3', color="black", alpha=0.5)
+
      ylab('') + xlab('') +
      coord_flip() +
      theme_minimal() +
      theme(panel.grid.major=element_blank(),
panel.grid.minor=element_blank())

paymentMethodPlot <- ggplot(custData, aes(x=PaymentMethod)) +
      ggtitle('PaymentMethod') +
      geom_histogram(stat="count", fill='#2748b3', color="black", alpha=0.5)
+
      ylab('') + xlab('') +
      coord_flip() +
      theme_minimal() +
      theme(panel.grid.major=element_blank(),
panel.grid.minor=element_blank())

grid.arrange(contractPlot, paymentMethodPlot)
```





J. Bivariate variable statistics

For the bivariate analysis, each of the independent categorical variables will be compared the the dependent variable, churn using a chi-squared test for independence.

```
chisq.test(custData$gender, custData$Churn, correct=FALSE)

##
##  Pearson's Chi-squared test
##
## data:  custData$gender and custData$Churn
## X-squared = 0.51341, df = 1, p-value = 0.4737

chisq.test(custData$SeniorCitizen, custData$Churn, correct=FALSE)

##
##  Pearson's Chi-squared test
##
## data:  custData$SeniorCitizen and custData$Churn
## X-squared = 159.36, df = 1, p-value < 2.2e-16

chisq.test(custData$Partner, custData$Churn, correct=FALSE)

##
##  Pearson's Chi-squared test
##
## data:  custData$Partner and custData$Churn
## X-squared = 158.18, df = 1, p-value < 2.2e-16
```

```r
chisq.test(custData$Dependents, custData$Churn, correct=FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  custData$Dependents and custData$Churn
## X-squared = 187.13, df = 1, p-value < 2.2e-16
```

```r
chisq.test(custData$PhoneService, custData$Churn, correct=FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  custData$PhoneService and custData$Churn
## X-squared = 0.9612, df = 1, p-value = 0.3269
```

```r
chisq.test(custData$MultipleLines, custData$Churn, correct=FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  custData$MultipleLines and custData$Churn
## X-squared = 11.27, df = 1, p-value = 0.0007879
```

```r
chisq.test(custData$InternetService, custData$Churn, correct=FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  custData$InternetService and custData$Churn
## X-squared = 728.7, df = 2, p-value < 2.2e-16
```

```r
chisq.test(custData$OnlineSecurity, custData$Churn, correct=FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  custData$OnlineSecurity and custData$Churn
## X-squared = 206.27, df = 1, p-value < 2.2e-16
```

```r
chisq.test(custData$OnlineBackup, custData$Churn, correct=FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  custData$OnlineBackup and custData$Churn
## X-squared = 47.638, df = 1, p-value = 5.127e-12
```

```r
chisq.test(custData$DeviceProtection, custData$Churn, correct=FALSE)
```

```
##
##  Pearson's Chi-squared test
##
```

```
## data:  custData$DeviceProtection and custData$Churn
## X-squared = 30.81, df = 1, p-value = 2.845e-08
```

```
chisq.test(custData$TechSupport, custData$Churn, correct=FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  custData$TechSupport and custData$Churn
## X-squared = 190.79, df = 1, p-value < 2.2e-16
```

```
chisq.test(custData$StreamingTV, custData$Churn, correct=FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  custData$StreamingTV and custData$Churn
## X-squared = 28.135, df = 1, p-value = 1.131e-07
```

```
chisq.test(custData$StreamingMovies, custData$Churn, correct=FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  custData$StreamingMovies and custData$Churn
## X-squared = 26.046, df = 1, p-value = 3.334e-07
```

```
chisq.test(custData$Contract, custData$Churn, correct=FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  custData$Contract and custData$Churn
## X-squared = 1179.5, df = 2, p-value < 2.2e-16
```

```
chisq.test(custData$PaperlessBilling, custData$Churn, correct=FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  custData$PaperlessBilling and custData$Churn
## X-squared = 257.76, df = 1, p-value < 2.2e-16
```

```
chisq.test(custData$PaymentMethod, custData$Churn, correct=FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  custData$PaymentMethod and custData$Churn
## X-squared = 645.43, df = 3, p-value < 2.2e-16
```

```
chisq.test(custData$tenureBin, custData$Churn, correct=FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  custData$tenureBin and custData$Churn
## X-squared = 881.76, df = 5, p-value < 2.2e-16
```

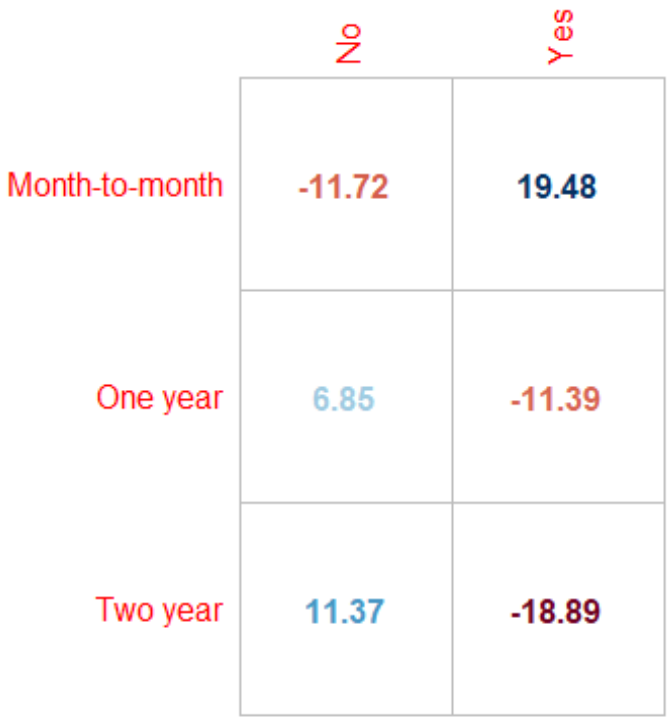
```
chisq.test(custData$monthlyChargesBin, custData$Churn, correct=FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  custData$monthlyChargesBin and custData$Churn
## X-squared = 373.09, df = 2, p-value < 2.2e-16
```

Churn looks to be dependent on most of the remaining variables, with the exception of gender and PhoneService. These variables will remain in the analysis as its possible a multi-variate analysis will find different results.

The relationship between contract and churn is visualized below. Correlation plots on the residuals will show which cells from the contingency tables contributed most to X-squared (STHDA, 2016).

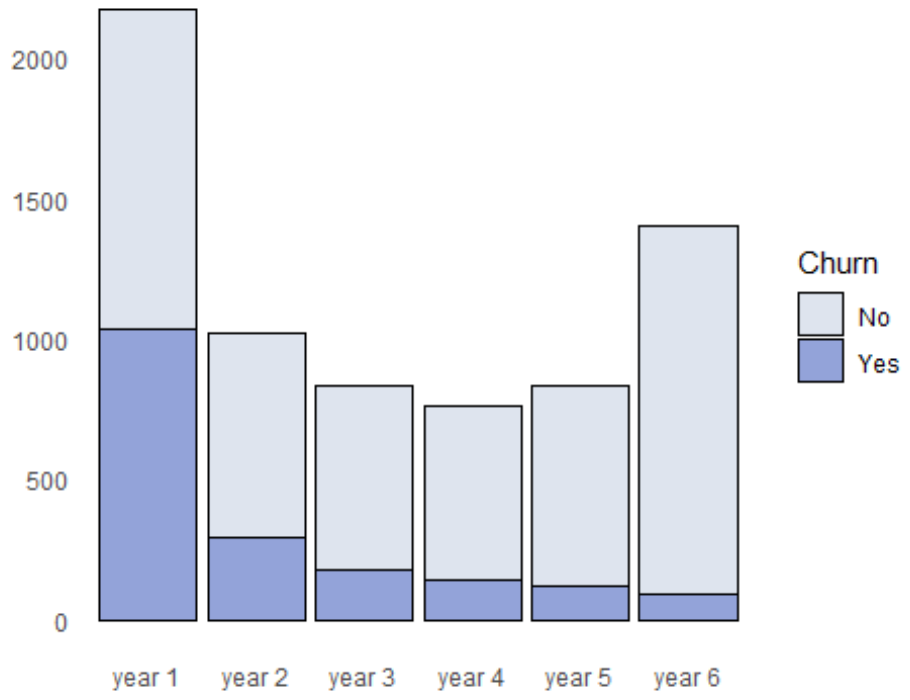
```
chi <- chisq.test(custData$Contract, custData$Churn, correct=FALSE)
corrplot(chi$residuals, is.corr=FALSE, method='number', cl.pos='n')
```



Not surprisingly, customers in some form of contract tend to stay and customers not in a contract are more likely to churn. These types of plots will be easiest to read and

understand when the degrees of freedom is small. For a result with more degrees of freedom, such as tenureBin, stacked bars can help aid in comparisons.

```
ggplot(custData, aes(fill=Churn, y=..count.., x=tenureBin)) +
  geom_bar(position='stack', stat='count', alpha=0.5, color='black') +
  scale_fill_manual(values=c('#bdc9de','#2748b3')) +
  xlab('') + ylab('') +
  theme_minimal() +
  theme(panel.grid.major=element_blank(), panel.grid.minor=element_blank())
```



It seems that the longer a customer is with the organization, the less likely they are to churn. The first year is especially susceptible to churn.

K. Methods applied

Logistic regression

The analytic method will begin by applyint R's GLM function to the data while retaining a portion to test with as well. The data can be split into a training set to train the model and then a test set to test the accuracy. A 70/30 split was used.

```
row70 <- floor(0.7 * nrow(custData))
set.seed(2020) #for repeatability
trainRows <- sample(seq_len(nrow(custData)), size=row70)

trainSet <- custData[trainRows,]
testSet <- custData[-trainRows,]
```

```r
logReg <- glm(Churn ~ ., data=trainSet, family='binomial')
summary(logReg)
```

```
##
## Call:
## glm(formula = Churn ~ ., family = "binomial", data = trainSet)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8601  -0.6838  -0.3001   0.6936   3.0401
##
## Coefficients:
##                                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)                        -0.22923    0.19872  -1.154 0.248681
## genderMale                         -0.01287    0.07694  -0.167 0.867164
## SeniorCitizenYes                    0.18885    0.09985   1.891 0.058586
.
## PartnerYes                         -0.04549    0.09193  -0.495 0.620704
## DependentsYes                      -0.06374    0.10521  -0.606 0.544603
## PhoneServiceYes                    -0.40295    0.24747  -1.628 0.103460
## MultipleLinesYes                    0.18913    0.09299   2.034 0.041975
*
## InternetServiceFiber optic          1.05655    0.21082   5.012 5.40e-07
***
## InternetServiceNo                  -0.92741    0.30577  -3.033 0.002421
**
## OnlineSecurityYes                  -0.34212    0.09858  -3.470 0.000520
***
## OnlineBackupYes                    -0.14064    0.09076  -1.550 0.121250
## DeviceProtectionYes                -0.02690    0.09365  -0.287 0.773955
## TechSupportYes                     -0.35701    0.10167  -3.512 0.000445
***
## StreamingTVYes                      0.22140    0.09616   2.302 0.021310
*
## StreamingMoviesYes                  0.34056    0.09581   3.554 0.000379
***
## ContractOne year                   -0.83752    0.12951  -6.467 9.99e-11
***
## ContractTwo year                   -1.60745    0.20856  -7.707 1.28e-14
***
## PaperlessBillingYes                 0.29117    0.08922   3.264 0.001100
**
## PaymentMethodCredit card (automatic) -0.07506    0.13475  -0.557 0.577506
## PaymentMethodElectronic check       0.27220    0.11101   2.452 0.014209
*
## PaymentMethodMailed check           0.00866    0.13636   0.064 0.949364
## tenureBinyear 2                    -0.87823    0.11508  -7.632 2.32e-14
***
## tenureBinyear 3                    -1.19871    0.13607  -8.809  < 2e-16
```

```
***
## tenureBinyear 4                       -1.15573    0.15346  -7.531 5.04e-14
***
## tenureBinyear 5                       -1.36761    0.16629  -8.224  < 2e-16
***
## tenureBinyear 6                       -1.56660    0.20038  -7.818 5.36e-15
***
## monthlyChargesBintier 2                0.01329    0.26856   0.049 0.960524
## monthlyChargesBintier 3               -0.20372    0.36951  -0.551 0.581407
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5691.3  on 4921  degrees of freedom
## Residual deviance: 4161.9  on 4894  degrees of freedom
## AIC: 4217.9
##
## Number of Fisher Scoring iterations: 6
```

There are many variables that have very little impact impact on the model. The median of the deviance residuals is close to 0 and the first and 3rd quartiles are very close to symmetrical. However, the min and max are not close to symmetrical. The initial look at this model is a reasonably good fit, but could be improved (starmer, 2018). An r-squared approximation and associated p-value will be paired with the AIC score to make sure any additional steps are improvements to the fit. Prediction accuracy on the data partition held back for testing will also be reviewed.

```
ll.null <- logReg$null.deviance/-2
ll.proposed <- logReg$deviance/-2
print(paste("McFadden's pseudo r-squared: ", (ll.null - ll.proposed) /
ll.null))

## [1] "McFadden's pseudo r-squared:  0.268737782707466"

print(paste('p-value: ',
     1 - pchisq(2*(ll.proposed - ll.null), df=(length(logReg$coefficients)-
1))))

## [1] "p-value:  0"

testSet$Churn <- as.character(testSet$Churn)
testSet$Churn[testSet$Churn=="No"] <- "0"
testSet$Churn[testSet$Churn=="Yes"] <- "1"
fitted <- predict(logReg, newdata=testSet, type='response')
fitted <- ifelse(fitted > 0.5, 1, 0)
errorRate <- mean(fitted != testSet$Churn)
print(paste('model accuracy: ', 1 - errorRate)) # (Li, 2017)

## [1] "model accuracy:  0.814691943127962"
```

With so many variables having low coefficients and high p-values in the model, an ANOVA test can be run on the deviances to estimate how much each variable contributes to the model.

```
anova(logReg, test='Chisq')
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: Churn
##
## Terms added sequentially (first to last)
##
##
##                   Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                                4921     5691.3
## gender             1     0.70      4920     5690.6  0.403804
## SeniorCitizen      1    92.51      4919     5598.1 < 2.2e-16 ***
## Partner            1   106.42      4918     5491.7 < 2.2e-16 ***
## Dependents         1    24.72      4917     5467.0 6.629e-07 ***
## PhoneService       1     2.06      4916     5464.9  0.151299
## MultipleLines      1     3.17      4915     5461.8  0.074808 .
## InternetService    2   490.97      4913     4970.8 < 2.2e-16 ***
## OnlineSecurity     1   151.81      4912     4819.0 < 2.2e-16 ***
## OnlineBackup       1    68.81      4911     4750.2 < 2.2e-16 ***
## DeviceProtection   1    47.97      4910     4702.2 4.328e-12 ***
## TechSupport        1    79.95      4909     4622.2 < 2.2e-16 ***
## StreamingTV        1     1.21      4908     4621.0  0.270435
## StreamingMovies    1     1.61      4907     4619.4  0.204878
## Contract           2   278.44      4905     4341.0 < 2.2e-16 ***
## PaperlessBilling   1    10.69      4904     4330.3  0.001076 **
## PaymentMethod      3    28.74      4901     4301.5 2.542e-06 ***
## tenureBin          5   138.75      4896     4162.8 < 2.2e-16 ***
## monthlyChargesBin  2     0.92      4894     4161.9  0.630456
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

In the model, InternetService clears the most deviance from the null model. Contract, OnlineSecurity and tenureBin also have strong predictor power. A model with just these 4 predictors may be easier to understand and therefore more valuable if it has a similar predictive power.

```
trimTrain <- trainSet[c(7,8,14,18,17)]
trimTest <- testSet[c(7,8,14,18,17)]

logRegTrim <- glm(Churn ~ ., data=trimTrain, family='binomial')
summary(logRegTrim)
```

```
##
## Call:
## glm(formula = Churn ~ ., family = "binomial", data = trimTrain)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5626  -0.6803  -0.3348   0.8361   3.0002
##
## Coefficients:
##                             Estimate Std. Error z value Pr(>|z|)
## (Intercept)                -0.21369    0.08116  -2.633  0.00846 **
## InternetServiceFiber optic  1.08491    0.08850  12.258  < 2e-16 ***
## InternetServiceNo          -1.21998    0.14401  -8.472  < 2e-16 ***
## OnlineSecurityYes          -0.46157    0.09607  -4.804 1.55e-06 ***
## ContractOne year           -0.92649    0.12510  -7.406 1.30e-13 ***
## ContractTwo year           -1.82727    0.20091  -9.095  < 2e-16 ***
## tenureBinyear 2            -0.82705    0.10997  -7.521 5.44e-14 ***
## tenureBinyear 3            -1.13214    0.12641  -8.956  < 2e-16 ***
## tenureBinyear 4            -1.05538    0.14166  -7.450 9.32e-14 ***
## tenureBinyear 5            -1.22839    0.15056  -8.159 3.38e-16 ***
## tenureBinyear 6            -1.43563    0.17714  -8.105 5.29e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5691.3  on 4921  degrees of freedom
## Residual deviance: 4262.0  on 4911  degrees of freedom
## AIC: 4284
##
## Number of Fisher Scoring iterations: 6
```

```
ll.null <- logReg$null.deviance/-2
ll.proposed <- logReg$deviance/-2
print(paste("McFadden's pseudo r-squared: ", (ll.null - ll.proposed) /
ll.null))
```

```
## [1] "McFadden's pseudo r-squared:  0.268737782707466"
```

```
print(paste('p-value: ',
     1 - pchisq(2*(ll.proposed - ll.null),
                df=(length(logRegTrim$coefficients)-1))))
```

```
## [1] "p-value:  0"
```

```
fitted <- predict(logRegTrim, newdata=trimTest, type='response')
fitted <- ifelse(fitted > 0.5, 1, 0)
errorRate <- mean(fitted != trimTest$Churn)
print(paste('model accuracy: ', 1 - errorRate)) # (Li, 2017)
```

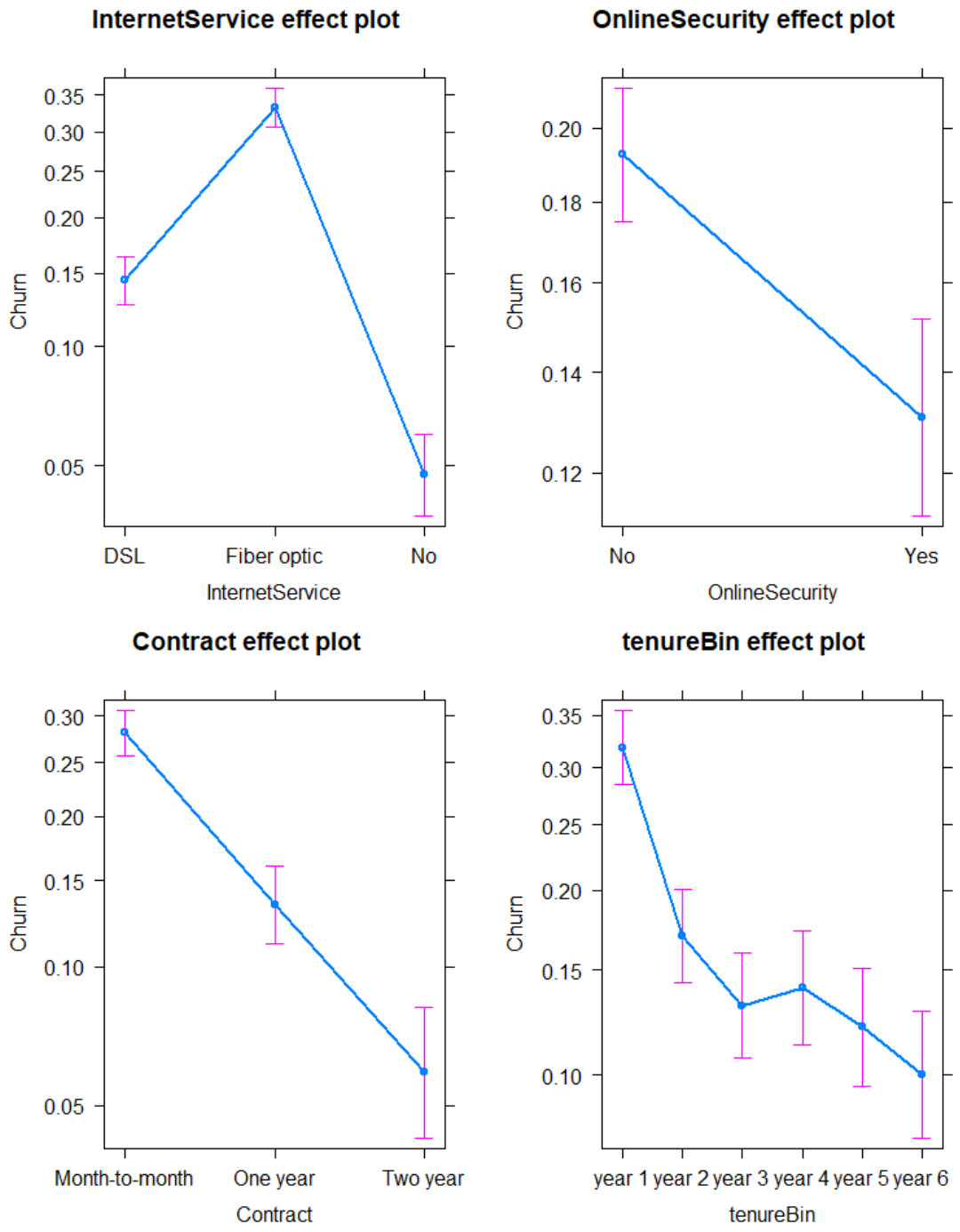
```
## [1] "model accuracy:  0.799526066350711"
```

```
anova(logRegTrim, test='Chisq')
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: Churn
##
## Terms added sequentially (first to last)
##
##
##                 Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                             4921     5691.3
## InternetService  2   549.44      4919     5141.9 < 2.2e-16 ***
## OnlineSecurity   1   208.33      4918     4933.6 < 2.2e-16 ***
## Contract         2   510.97      4916     4422.6 < 2.2e-16 ***
## tenureBin        5   160.59      4911     4262.0 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This model has slightly tighter deviance residuals, a very similar AIC and McFadden's pseudo r-squared. The prediction accuracty is only 1.5 percentage points lower the data set aside for testing. This simpler model may ultimately be more valuable as it is easier to understand. Visualizing the effect of the probability for each level of the predictor variables can also help to explain the model.

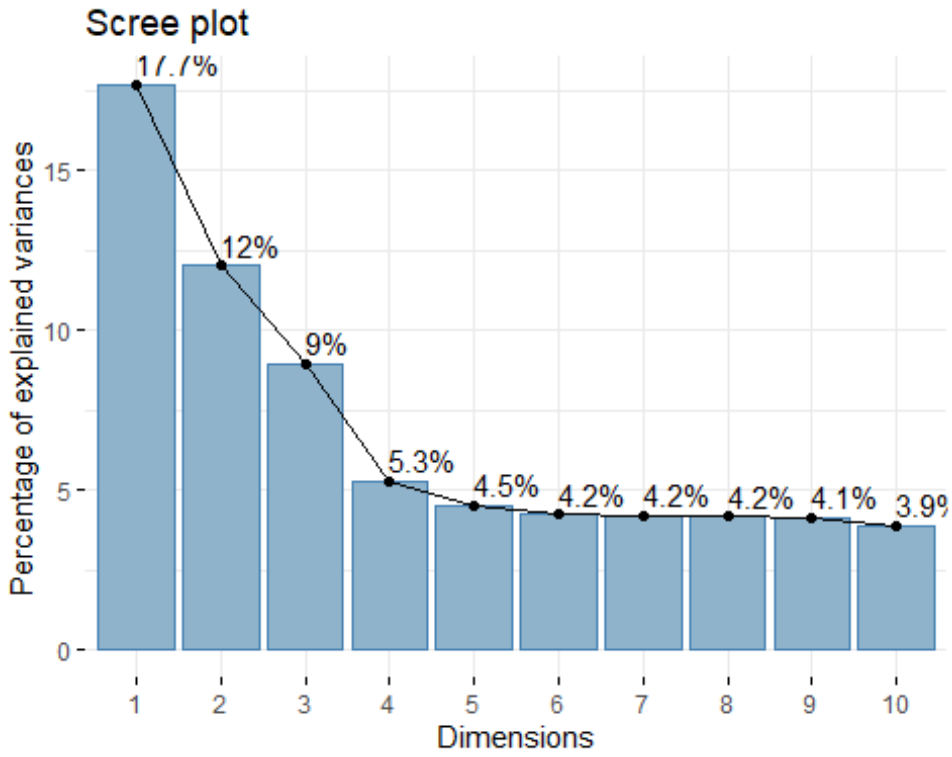
```
plot(allEffects(logRegTrim))
```



Multiple correspondence analysis

One style of MCA allows qualitative variables to be divided into two categories: active and supplemental. This data has a natural partition on demographic and telco relationship variables. Gender, SeniorCitizen, Partner and Dependents will used as supplemental variables and the remaining variables will be active. Since 15 variables will be interpreted, any dimensions with less than 7% of the total variance can be discarded (Housson, 2016).

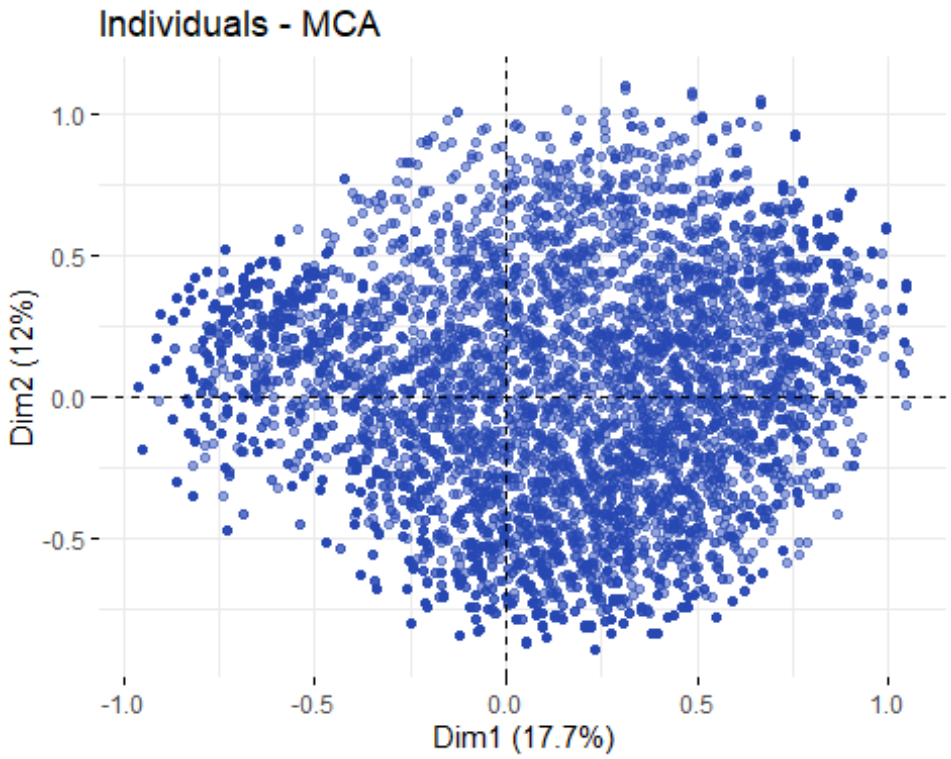
```
custFactor <- MCA(custData, quali.sup=c(1:4))

fviz_eig(custFactor, addlabels=TRUE, barfill='#8FB3CB')

## Registered S3 methods overwritten by 'car':
##   method                           from
##   influence.merMod                 lme4
##   cooks.distance.influence.merMod  lme4
##   dfbeta.influence.merMod          lme4
##   dfbetas.influence.merMod         lme4
```



In this case, the only pair of dimensions where both have more than 7% of the total variance is dimensions 1 and 2. These will be the focus of the rest of this part of the analysis.

For MCA, it is important that the dimensions are centered and distributed in a uniform cloud around zero. The point cloud of individuals will show this (STHDA, 2017).

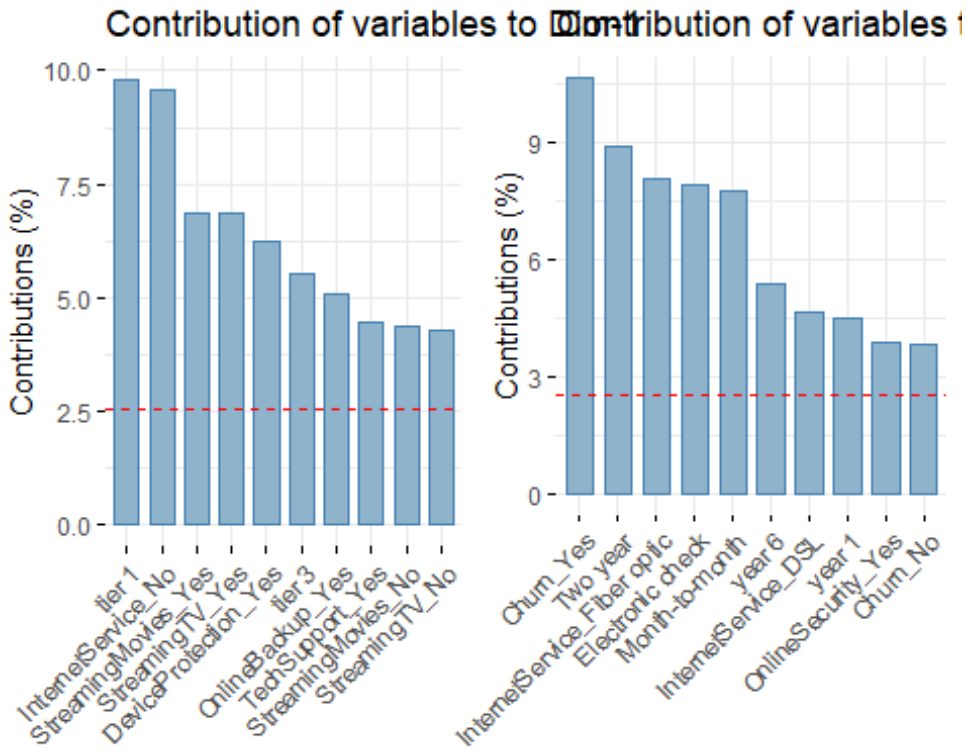
```
fviz_mca_ind(custFactor, label='none', col.ind='#2748b3', alpha.ind=0.5)
```



The point cloud seems to be centered and uniformly distributed across dimensions one and two.

Viewing each variable's contribution to these dimensions may give additional insight. Only the 10 variables with the largest contribution are shown in the figures below.

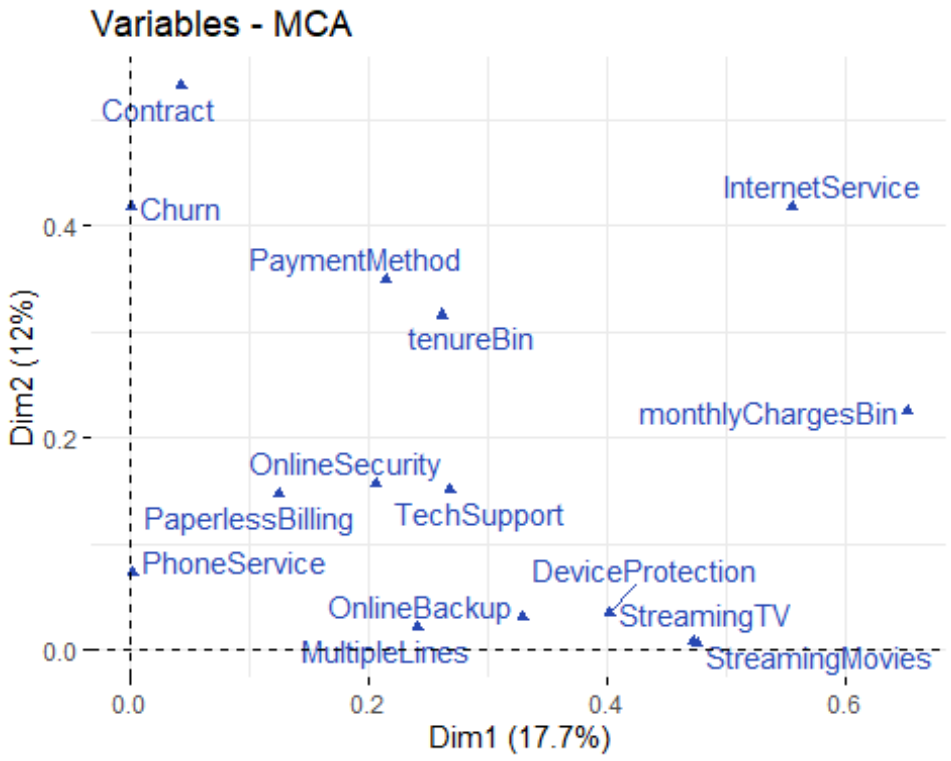
```
dim1 <- fviz_contrib(custFactor, choice='var', top=10, axes=1,
fill='#8FB3CB')
dim2 <- fviz_contrib(custFactor, choice='var', top=10, axes=2,
fill='#8FB3CB')
grid.arrange(dim1, dim2, ncol=2)
```



customers with the lowest tier of monthly charges are associated to those without internet service on the first dimension. For the second dimension, customers who have churned are associated to those with a two year contract.

The correlation between these variables and these dimensions can also be plotted to compare the correlation of each variable to both dimensions. The supplemental variables have been suppressed as they do not contribute to the dimensions.

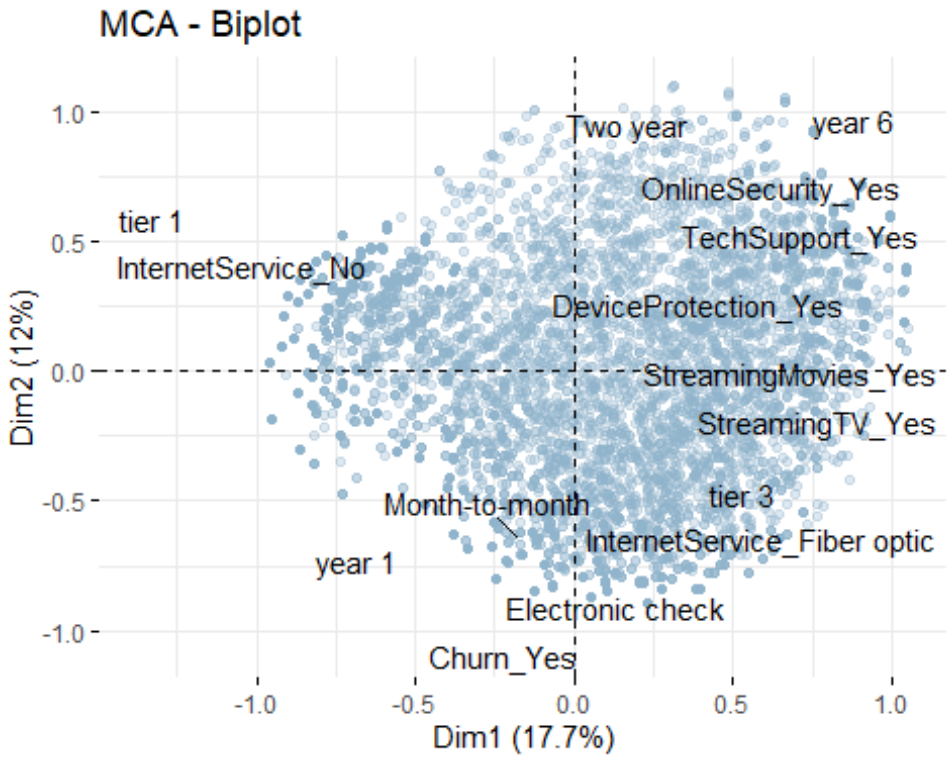
```
fviz_mca_var(custFactor,
             choice = "mca.cor",
             col.var='#2748b3',
             repel=TRUE,
             select.var=list(name=colnames(custData[5:19]))
            )
```



Contract plays a strong role in dimension 2, but is relatively weak in dimension 1. InternetService is is significant in both dimensions.

A biplot of individuals and categories will help to visualize these dimensions as well.

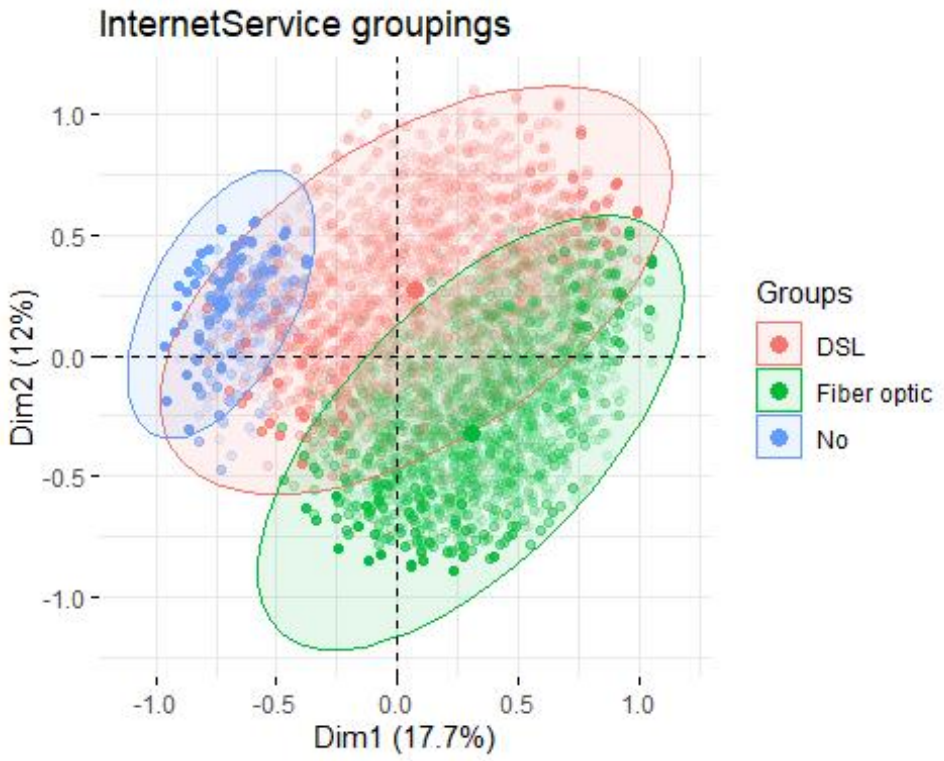
```
fviz_mca_biplot(custFactor, geom.ind='point', col.ind='#8FB3CB',
alpha.ind=0.3,
                geom.var='text', repel=TRUE, label='var',
                invisible='quali.sup', select.var=list(contrib=15),
                col.var='black')
```



This plot shows the similarity of individuals and variables in this set. The distance between points represents how similar the points are.

Finally, single variables can also be examined in this form. In the plot below, the ellipses represent the different levels of the InternetService variable.

```
grouping <- as.factor(custData[,"InternetService"])
fviz_mca_ind(custFactor, habillage=grouping, addEllipses=TRUE, label='none',
             alpha.ind=0.1, title='InternetService groupings')
```



L. Method justification

As the goal was to identify which customers are leaving the telco and attempt to mitigate continued customer loss, a predictive method was needed to identify customers that were at risk leaving so extra care could be given to maintain them. Logistic regression was selected as a non-descriptive and predictive method to attempt to predict the value of Churn. Logistic regression was especially suited to this task as it performs well on binary factors and is considered more transparent and easier to explain than other methods (Tufféry, 2011). Multiple linear regression is better suited for continuous variables, where the majority of the variables in the given set were discrete factors. A neural network may also have been appropriate for this prediction task, however, they are not well suited to categorical variables without some additional preparation, have difficulty with large numbers of variables and lack the transparency of logistic regression. Explaining how the network converged can also be difficult, making the task of troubleshooting or fine-tuning them not as transparent as logistic regression.

As part of identifying characteristics of customer who leave the telco, component analysis could also add value in that it segments customers who have similarities and describes the customer base in ways that may not be initially understood. The data did include several different data types, making Multiple Correspondence Analysis (MCA) the correct choice for component analysis (Tufféry, 2011). Principal Component Analysis (PCA) requires continuous and qualitative variables. Many of the given variables were discrete factors. Clustering analysis methods may also be useful, however, they do require continuous variables. Categorical data must first be transformed through MCA. This work may be continued using these methods. Association analysis would also be appropriate in the

presence of data from many more customers (Tufféry, 2011). Given the parameters of this task, MCA was the correct initial choice to describe the telco's customers.

M. Visualization justification

Univariate analysis

Bar charts and histograms show the distribution of individual across levels of factor variables well. Single stacked bars or pie charts also have this same goal, but these can distort the proportion of the whole each level represents and make comparisons between the levels difficult. Density plots represent the distribution of continuous values well as they highlight the area under their curves, aiding with the intuitive understanding of counts similarly to a bar chart. Line plots also visualize continuous variables well, but it can be difficult to estimate the slope or area under the curve at a glance with line plots.

Bi-variate analysis

Stacked bar charts are especially useful for comparing 2 categorical variables as they quickly show the distribution of a variable in reference to another. Scatter plots are also available, but work best with at least one continuous variable. Jitter has to be introduced when categorical variables are used in a scatter plot and this can be easy to misinterpret. Occasionally, when there are very few categories available in a categorical variable, a heat table is also appropriate as it is easy to tell the difference and understand the scale between a few plainly written numbers. As the number of categories grow, a heat table becomes more cluttered and other methods are needed. Pie charts and donut charts are also sometimes used in specific circumstances, but it can be difficult to distinguish the ratio between categories on these chart types as the area of a wedge or circle is less intuitive for most people than the length of a rectangle.

Logistic Regression

The effects plot was chosen to highlight the differences in the predicted probabilities from the various levels of the factors included. As a bonus, the plot also includes the 95% error bars to gauge the uncertainty of the estimate (Ford, 2016). The conventional signmoid curve visualization would have also been appropriate, but these can get overloaded and difficult to inperpret as variables are added to the model. Adding confidence interval bands compounds this.

MCA

The scree plot was selected for a similar reson to the bar charts in the univariate analysis. This goal is to display what part each dimension has in the total variance. The line and labels were added to highlight the 7% cutoff for dimension usefulness. Pie or donut charts would have been especially inappropriate here as the many of the wedges would have been too small to see. The point cloud, bi-plot and InternetService grouping scatterplots were selected as this type of plot is especially useful in bivariate comparisons where the distribution of variables is important. Individual histograms, bar charts, or area charts would have possibly been better fits for any one of this visualizations. However, when

taken as a whole, the scatterplot provides a continuous view to explore and identify relationships.

Data summary

N. Was the data discriminating?

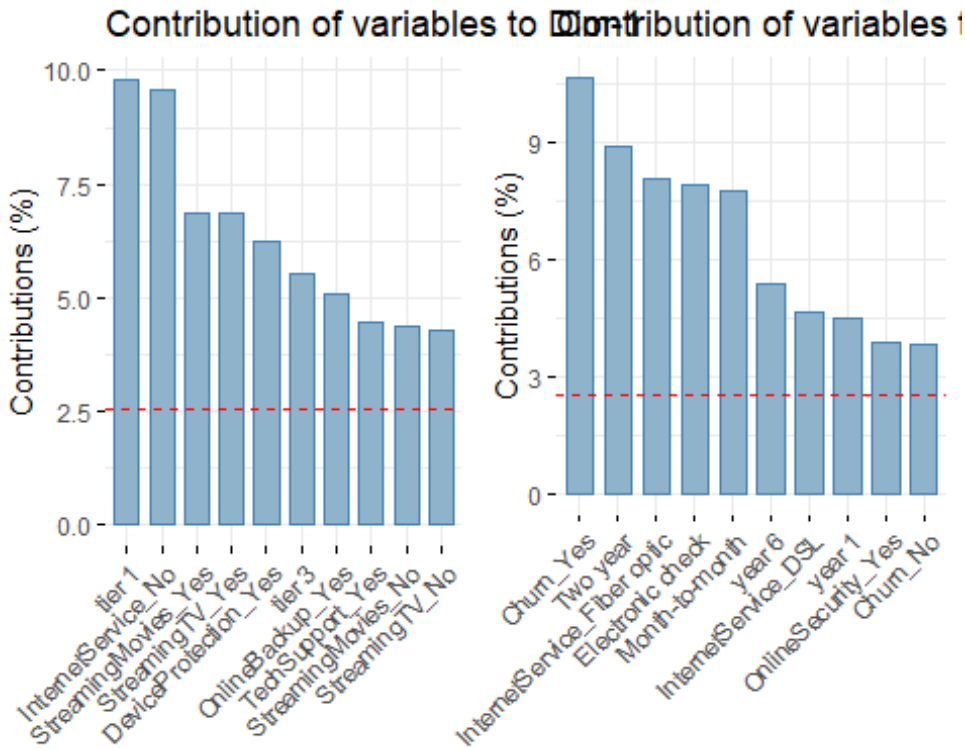
This data was discriminating as seen in the analysis of deviance tables from the logistic regression. The final model included four discriminating variables with significant p-values.

```
anova(logRegTrim, test='Chisq')
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: Churn
##
## Terms added sequentially (first to last)
##
##
##                 Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                              4921     5691.3
## InternetService  2   549.44      4919     5141.9 < 2.2e-16 ***
## OnlineSecurity   1   208.33      4918     4933.6 < 2.2e-16 ***
## Contract         2   510.97      4916     4422.6 < 2.2e-16 ***
## tenureBin        5   160.59      4911     4262.0 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The phenomenon of similarities between customers was found and outlined during the MCA analysis.

```
dim1 <- fviz_contrib(custFactor, choice='var', top=10, axes=1,
fill='#8FB3CB')
dim2 <- fviz_contrib(custFactor, choice='var', top=10, axes=2,
fill='#8FB3CB')
grid.arrange(dim1, dim2, ncol=2)
```



Customers with the lowest monthly charges also do not have internet service.

O. Detect interactions and strong predictors

Chi-squared tests were used to detect interactions between variables. These tests compared the observed frequencies of interaction between two qualitative variables and the expected frequencies if they were independent. A high test statistic indicates there is some interaction between those variables. Contract and tenure were both found to have strong interactions with churn. The p-value of these test statistics were significant.

```
chisq.test(custData$Contract, custData$Churn, correct=FALSE)

##
##  Pearson's Chi-squared test
##
## data:  custData$Contract and custData$Churn
## X-squared = 1179.5, df = 2, p-value < 2.2e-16

chisq.test(custData$tenureBin, custData$Churn, correct=FALSE)

##
##  Pearson's Chi-squared test
##
## data:  custData$tenureBin and custData$Churn
## X-squared = 881.76, df = 5, p-value < 2.2e-16
```

To select the most important predictor variables, an Analysis of Deviance was run on the initial logistic regression model. The variables with the highest explained deviance that had an appropriate p-value were selected and all other variables were trimmed from the model. The Analysis of Deviance for the resulting model was then checked and all variables were found to be significant.

Analysis of Deviance before trimming

```
anova(logReg, test='Chisq')
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: Churn
##
## Terms added sequentially (first to last)
##
##
##                   Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                                4921     5691.3
## gender             1     0.70      4920     5690.6  0.403804
## SeniorCitizen      1    92.51      4919     5598.1 < 2.2e-16 ***
## Partner            1   106.42      4918     5491.7 < 2.2e-16 ***
## Dependents         1    24.72      4917     5467.0 6.629e-07 ***
## PhoneService       1     2.06      4916     5464.9  0.151299
## MultipleLines      1     3.17      4915     5461.8  0.074808 .
## InternetService    2   490.97      4913     4970.8 < 2.2e-16 ***
## OnlineSecurity     1   151.81      4912     4819.0 < 2.2e-16 ***
## OnlineBackup       1    68.81      4911     4750.2 < 2.2e-16 ***
## DeviceProtection   1    47.97      4910     4702.2 4.328e-12 ***
## TechSupport        1    79.95      4909     4622.2 < 2.2e-16 ***
## StreamingTV        1     1.21      4908     4621.0  0.270435
## StreamingMovies    1     1.61      4907     4619.4  0.204878
## Contract           2   278.44      4905     4341.0 < 2.2e-16 ***
## PaperlessBilling   1    10.69      4904     4330.3  0.001076 **
## PaymentMethod      3    28.74      4901     4301.5 2.542e-06 ***
## tenureBin          5   138.75      4896     4162.8 < 2.2e-16 ***
## monthlyChargesBin  2     0.92      4894     4161.9  0.630456
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Analysis of Deviance after trimming.

```
anova(logRegTrim, test='Chisq')
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: Churn
```

```
##
## Terms added sequentially (first to last)
##
##
##                  Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                              4921     5691.3
## InternetService  2   549.44      4919     5141.9 < 2.2e-16 ***
## OnlineSecurity   1   208.33      4918     4933.6 < 2.2e-16 ***
## Contract         2   510.97      4916     4422.6 < 2.2e-16 ***
## tenureBin        5   160.59      4911     4262.0 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

References

Data Flair (2019) Why Learn R? 10 Handy Reason to Learn R programming Language. Retrieved from https://data-flair.training/blogs/why-learn-r/

Ford, Clay (2016), Visualizing the Effects of Logistic Regression Retrieved from https://data.library.virginia.edu/visualizing-the-effects-of-logistic-regression/

Housson, Fracois, (2016) Multiple Correspondence Analysis Playlist [Video Files]. Retrieved from https://www.youtube.com/watch?v=gZ_7WWEVlTg&list=PLnZgp6epRBbTVjKd_-KPhaGWLE7K7InL6

Li, Susan (2017), Predict Customer Churn - Logistic Regression, Decision Tree and Random Forest, Retrieved from https://datascienceplus.com/predict-customer-churn-logistic-regression-decision-tree-and-random-forest/

Starmer, Josh (2018) Logistic Regression Playlist [Video Files]. Retrieved from https://www.youtube.com/watch?v=xxFYro8QuXA&list=PLblh5JKOoLUKxzEP5HA2d-Li7IJkHfXSe&index=4

STHDA (2016), Chi-Square Test of Independence in R, Retrieved from http://www.sthda.com/english/wiki/chi-square-test-of-independence-in-r

STHDA (2017), MCA - Multiple Correspondence Analysis in R: Essentials Retrieved from https://goo.gl/ve3WBa

Tufféry, S. (2011). Data Mining and Statistics for Decision Making [VitalSource Bookshelf version]. Retrieved from vbk://9780470979280

Cleaned data set

Export of the cleaned data set.

```
write.xlsx2(custData, 'finalData.xlsx', sheetName = 'final', col.name = TRUE,
            row.names = FALSE)
```