Machine Learning usando o R
Lab04 - Previsão de Churn com Regressão Logística

Pedro Colangelo

Abstract

A rotatividade de clientes ocorre quando clientes ou assinantes param de fazer negócios com uma empresa ou serviço, também conhecido como atrito com clientes. Também é referido como perda de clientes ou simplesmente *churn*. Um setor no qual as taxas de cancelamento são particularmente úteis é o setor de telecomunicações. Vamos prever a rotatividade de clientes usando um conjunto de dados de telecomunicações disponível no site da IBM, com base em um modelo de regressão logística.

Introdução

O propósito deste laboratório é o de utilizar a capacidade preditiva de classificação da regressão logística para se criar um modelo que, baseado em determinadas características de cada indivíduo, possa definir a priori se este indivíduo irá "dar churn" (deixar de ser cliente da empresa).

Como base, será utilizado um trabalho de laboratório anterior que teve como objeto de estudo a mesma base de dados ("Telco-Customer-Churn.csv") aqui utilizada. Nele, o trabalho da análise exploratória de dados (etapa determinante para admitir quais variáveis iriam ser escolhidas como preditores para o modelo de regressão logística) já foi realizado e será reproduzido na íntegra na seção seguinte do documento.

Análise Exploratória de Dados

Importação das Bibliotecas Necessárias

```
library(ggplot2)
library(gridExtra)
library(readr)
```

Importação dos Dados

```
dados <- read.csv('Telco-Customer-Churn.csv')
```

Declaração das Variáveis

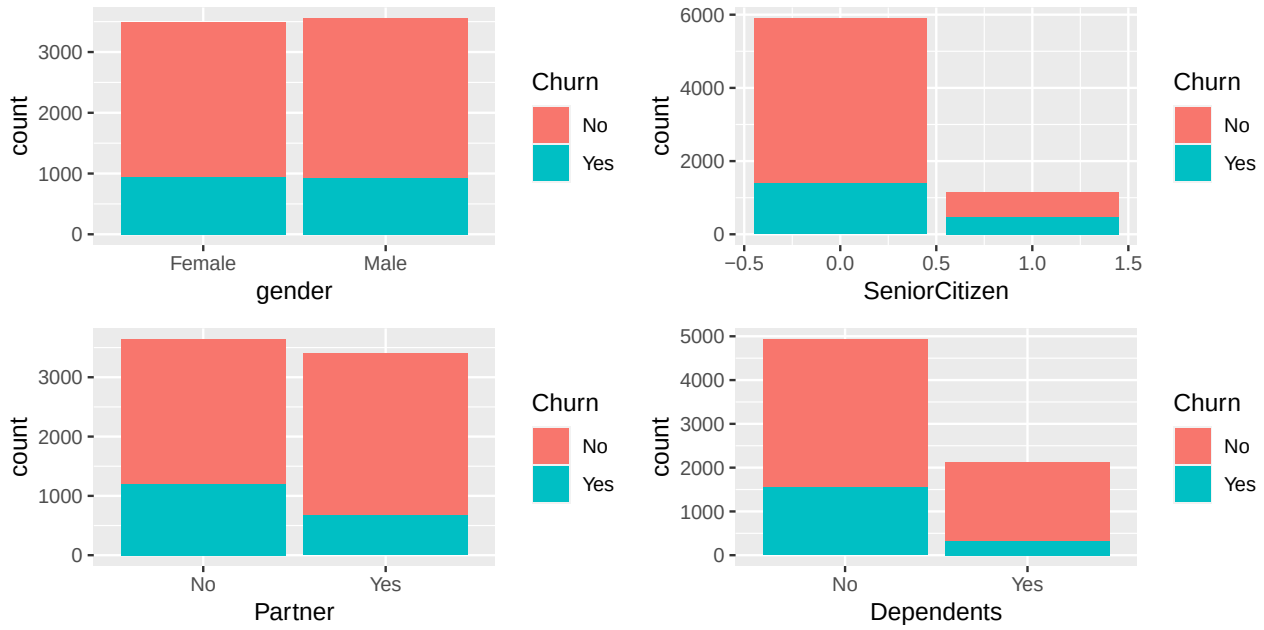
```
g1 <- ggplot(dados, aes(x = gender, fill = Churn))+
  geom_bar(position = 'stack')


g2 <- ggplot(dados, aes(x = SeniorCitizen, fill = Churn))+
  geom_bar(position = 'stack')


g3 <- ggplot(dados, aes(x = Partner, fill = Churn))+
  geom_bar(position = 'stack')


g4 <- ggplot(dados, aes(x = Dependents, fill = Churn))+
  geom_bar(position = 'stack')


g5 <- ggplot(dados, aes(x = PhoneService, fill = Churn))+
  geom_bar(position = 'stack')

g6 <- ggplot(dados, aes(x = MultipleLines, fill = Churn))+
  geom_bar(position = 'stack')

g7 <- ggplot(dados, aes(x = InternetService, fill = Churn))+
  geom_bar(position = 'stack')
```

```
g8 <- ggplot(dados, aes(x = OnlineSecurity, fill = Churn))+
  geom_bar(position = 'stack')

g9 <- ggplot(dados, aes(x = OnlineBackup, fill = Churn))+
  geom_bar(position = 'stack')

g10 <- ggplot(dados, aes(x = DeviceProtection, fill = Churn))+
  geom_bar(position = 'stack')

g11 <- ggplot(dados, aes(x = TechSupport, fill = Churn))+
  geom_bar(position = 'stack')

g12 <- ggplot(dados, aes(x = StreamingTV, fill = Churn))+
  geom_bar(position = 'stack')

g13 <- ggplot(dados, aes(x = StreamingMovies, fill = Churn))+
  geom_bar(position = 'stack')

g14 <- ggplot(dados, aes(x = Contract, fill = Churn))+
  geom_bar(position = 'stack')

g15 <- ggplot(dados, aes(x = PaperlessBilling, fill = Churn))+
  geom_bar(position = 'stack')

g16 <- ggplot(dados, aes(x = PaymentMethod, fill = Churn))+
  geom_bar(position = 'stack')+
  coord_flip()

g17 <- ggplot(dados,
       aes(x = Churn,
       y = tenure))+
  geom_boxplot()+
  labs(title = 'Distribuição de tempo como cliente por churn')

g18 <- ggplot(dados,
       aes(x = tenure,
           fill = Churn))+
  geom_density(alpha = 0.4)+
  labs(title = 'Distribuição de tempo como cliente por churn')

g19 <- ggplot(dados,
       aes(x = Churn,
           y = MonthlyCharges))+
  geom_boxplot()+
  labs(title = 'Distribuição de cobranças mensais por churn')

g20 <- ggplot(dados,
       aes(x = MonthlyCharges,
           fill = Churn))+
  geom_density(alpha = 0.4)+
  labs(title = 'Distribuição de cobranças mensais por churn')

g21 <- ggplot(dados,
```

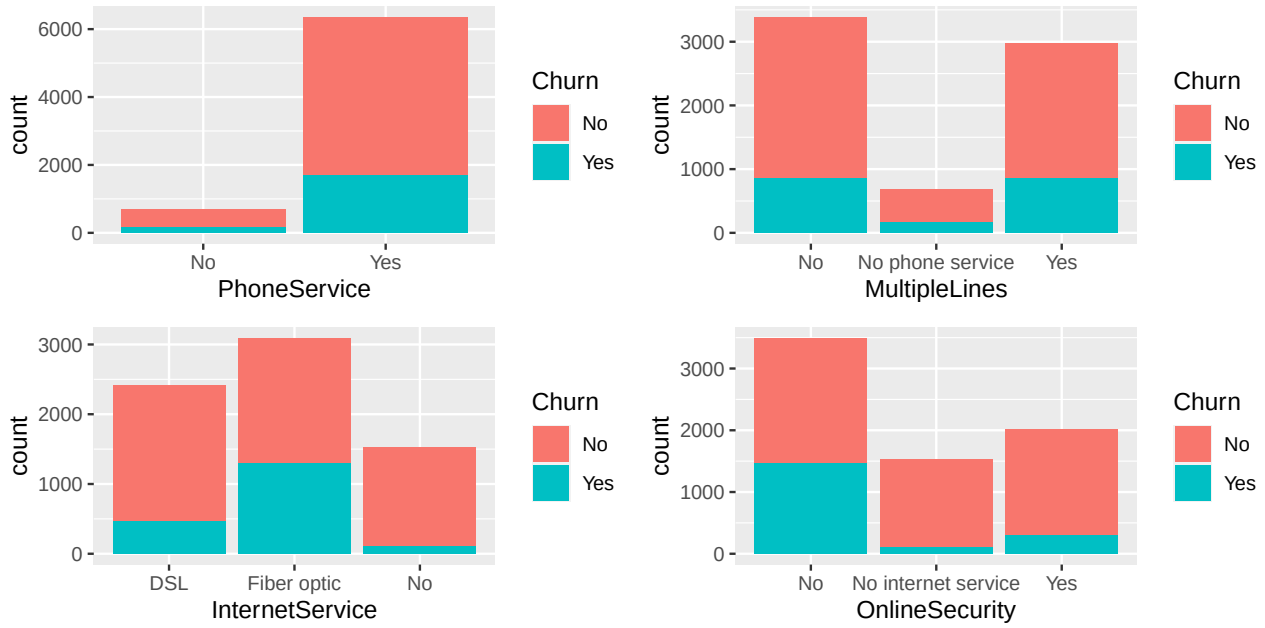
```
       aes(x = Churn,
           y = TotalCharges))+
  geom_boxplot()+
  labs(title = 'Distribuição de cobranças totais por churn')

g22 <- ggplot(dados,
       aes(x = TotalCharges,
           fill = Churn))+
  geom_density(alpha = 0.4)+
  labs(title = 'Distribuição de cobranças totais por churn')
```

Exibição dos Gráficos

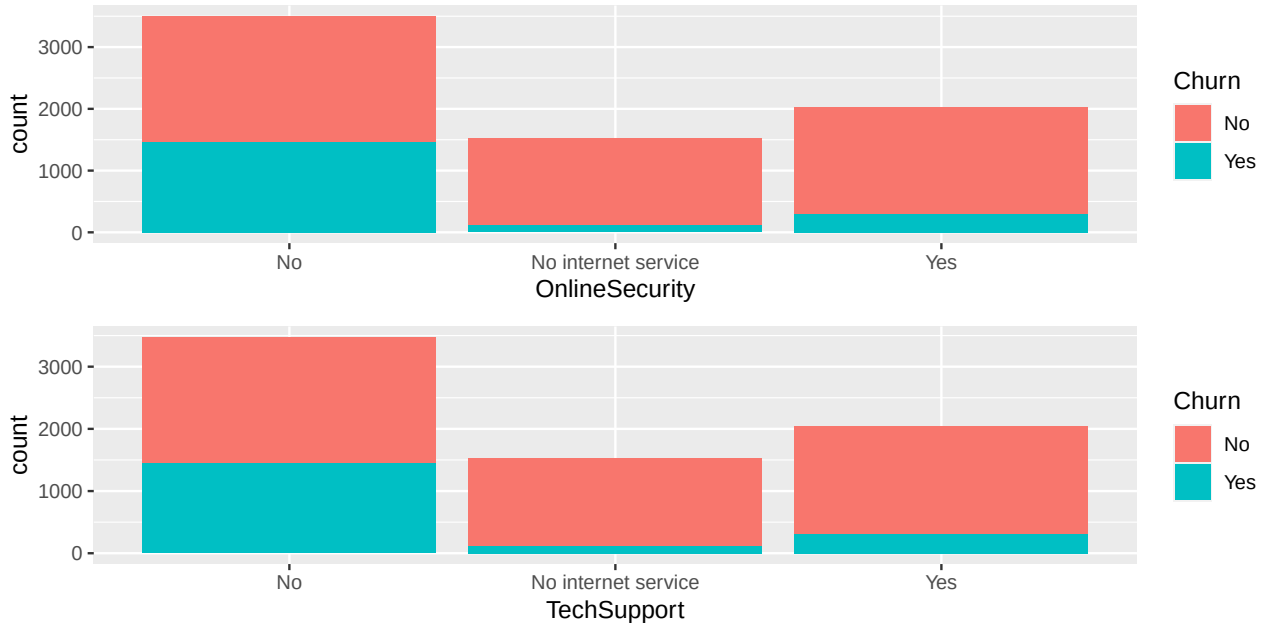
```
grid.arrange(g1, g2, g3, g4)
```



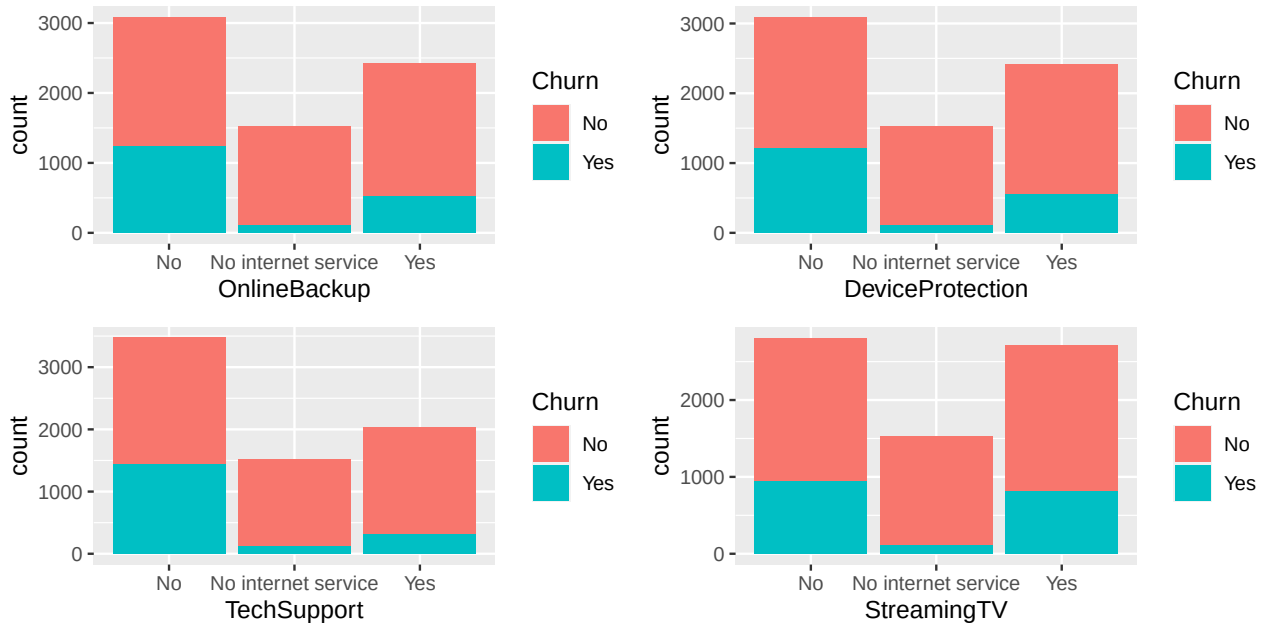

```
grid.arrange(g5, g6, g7, g8)
```



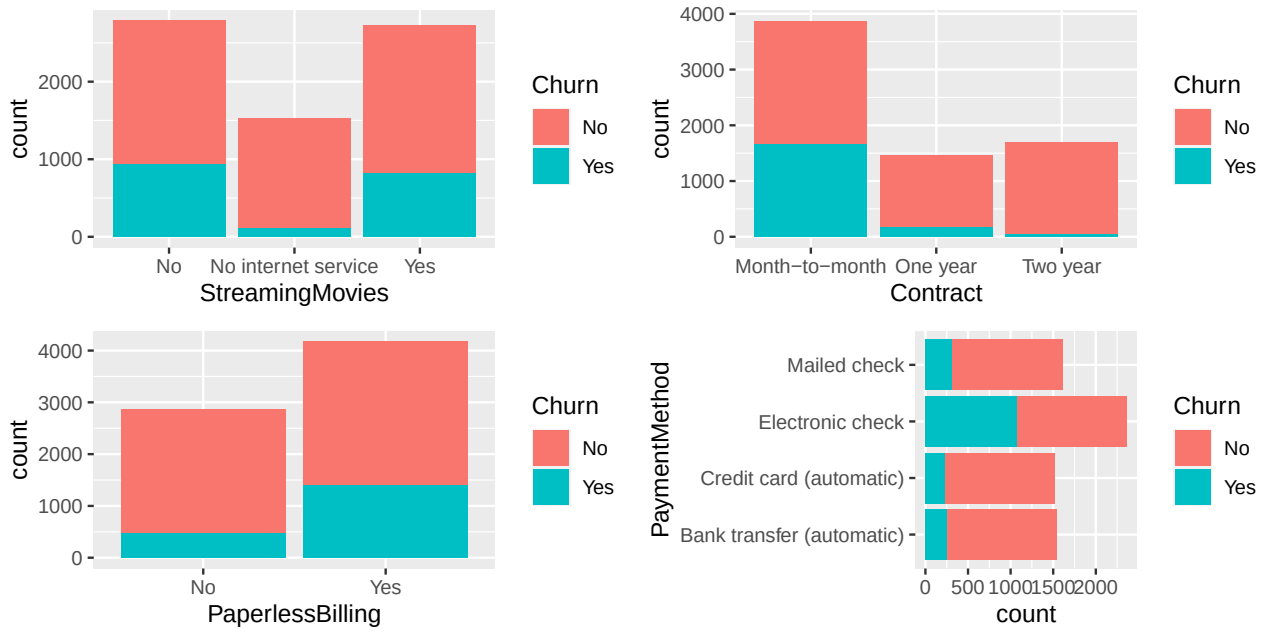

```
grid.arrange(g8, g11)
```



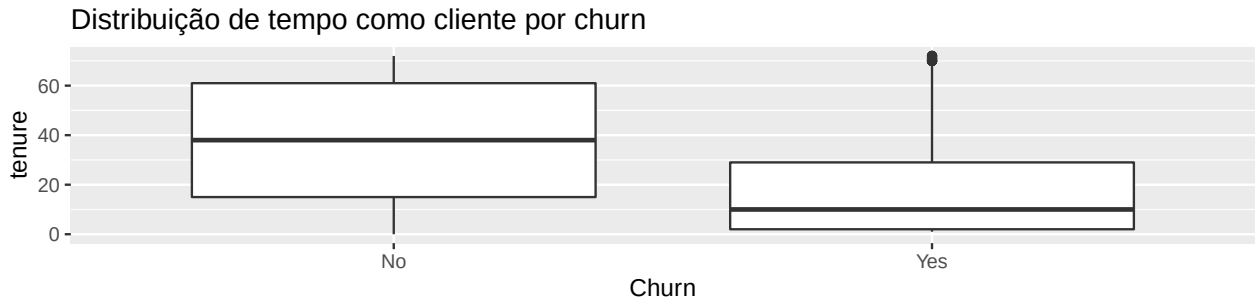

```
grid.arrange(g9, g10, g11, g12)
```

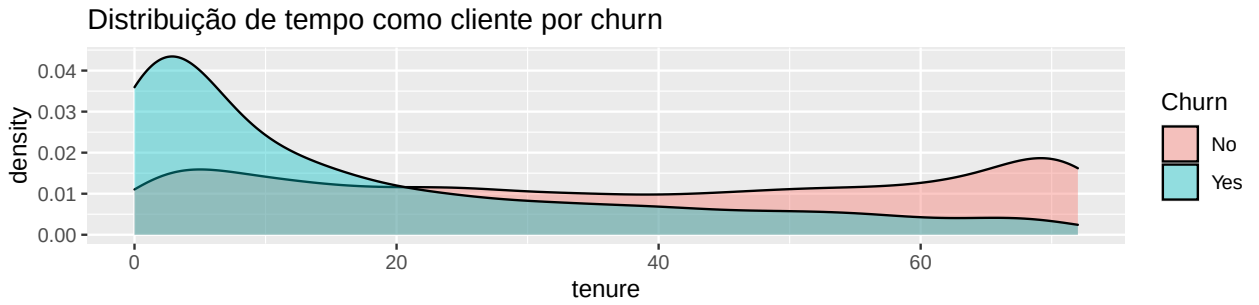



```
grid.arrange(g13, g14, g15, g16)
```

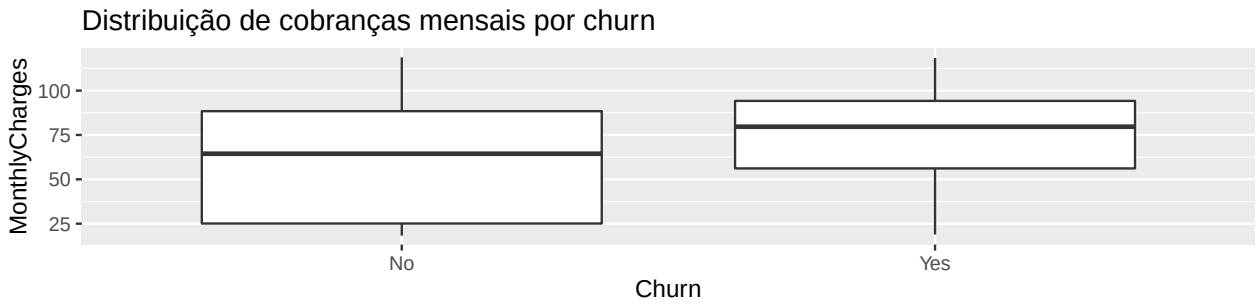


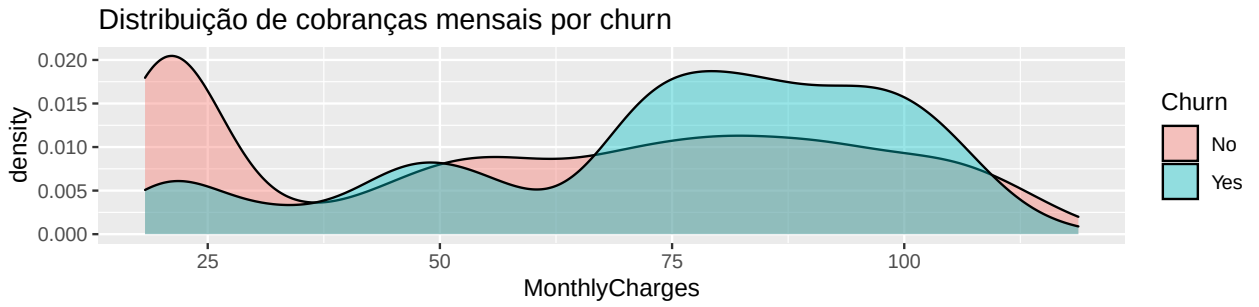

```
grid.arrange(g17, g18)
```



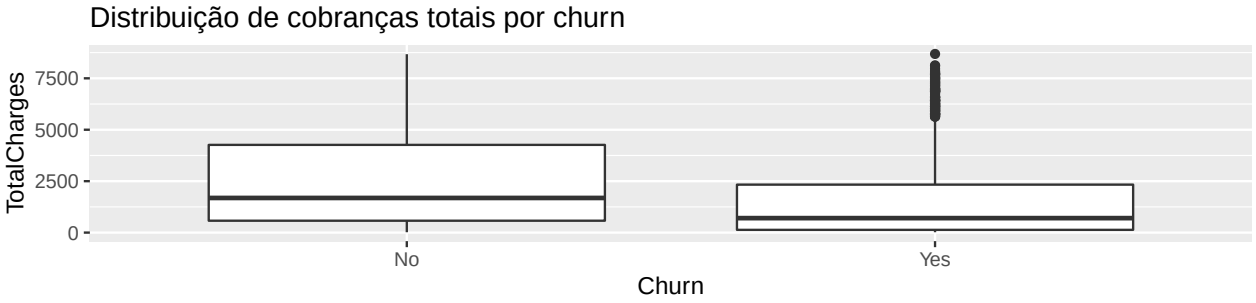


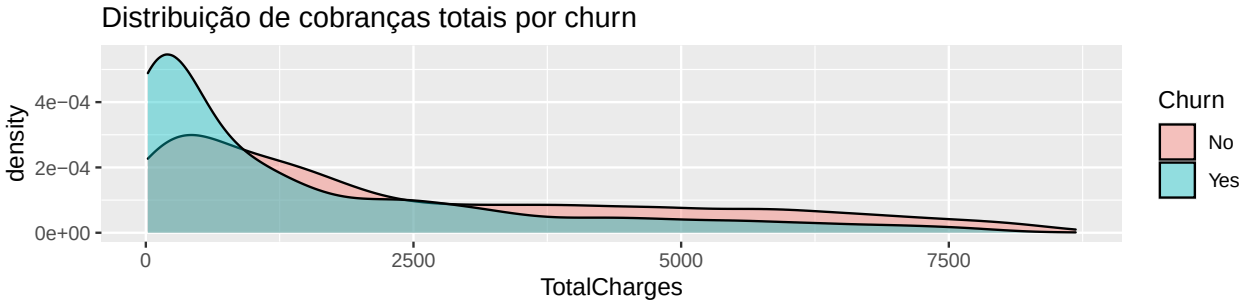

```
grid.arrange(g19, g20)
```






```
grid.arrange(g21, g22)
```





Variáveis de Destaque na Análise Exploratória

Variáveis categóricas: (i) método de pagamento escolhido, (ii) tipo de contrato, (iii) suporte técnico e (iv) segurança online.

Variáveis quantitativas: (v) tempo decorrido como cliente e (vi) cobranças mensais.

Embora outras variáveis não mencionadas também aparentem possuir determinado grau de capacidade preditiva, estas seis foram as elencadas por aparentarem maior influência do que as demais. Como em um modelo é desejável que tomemos as escolhas mais parcimoniosas possíveis quanto à quantidade de preditores, iremos nos ater a estas seis variáveis apenas.

Preparando os Dados para o Modelo

Importação das Bibliotecas Necessárias

```
library(caret)
library(car)
library(rcompanion)
library(performance)
library(OddsPlotty)
library(ggthemes)
library(ggplot2)
```

Novo Dataset Contendo Apenas as Variáveis de Interesse

```
df_logistic <- data.frame(dados[,21],
                          dados[,18],
                          dados[,16],
                          dados[,13],
                          dados[,10],
```

```
                          dados[,6],
                          dados[,19])

colnames(df_logistic) <- c('churn', 'payment_method',
                           'contract', 'tech_support',
                           'online_security', 'tenure',
                           'monthly_charges')

### Alteração dos factors da variável churn para
### 0's e 1's

df_logistic$churn <- ifelse(df_logistic$churn == 'Yes',
                            1,
                            0)

### Converter variáveis que são factor mas contam como
### character

df_logistic[,c(1, 2, 3, 4, 5)] <- lapply(df_logistic[, c(1, 2, 3, 4, 5)],
                                         as.factor)
```

Implementação do Modelo

```
## Separação do dataset de treino para o dataset de
## teste na proporção de 75%-25% respectivamente usando
## a função sample()

### Setando uma seed para tornar o modelo replicável

set.seed(123)

split <- sort(sample(nrow(df_logistic), nrow(df_logistic)*0.75))

training_set <- df_logistic[split,]
testing_set <- df_logistic[-split,]

## Padronização das variáveis contínuas do dataset
## para que o modelo não se confunda por conta das
## diferentes escalas

training_set['tenure'] <- scale(training_set['tenure'])

training_set['monthly_charges'] <- scale(training_set['monthly_charges'])

testing_set['tenure'] <- scale(testing_set['tenure'])

testing_set['monthly_charges'] <- scale(testing_set['monthly_charges'])

## Declaração do modelo de regressão logística

logit_churn <- glm(churn ~ ., data = training_set,
                   family = binomial(link = 'logit'))
```

```
summary(logit_churn)
```

```
##
## Call:
## glm(formula = churn ~ ., family = binomial(link = "logit"), data = training_set)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7362  -0.6843  -0.2956   0.7904   3.1113
##
## Coefficients: (1 not defined because of singularities)
##                                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)                         -0.69536    0.10515  -6.613 3.76e-11 ***
## payment_methodCredit card (automatic) -0.08110    0.12887  -0.629 0.529144
## payment_methodElectronic check       0.39654    0.10624   3.733 0.000190 ***
## payment_methodMailed check          -0.04950    0.12780  -0.387 0.698499
## contractOne year                    -0.79034    0.11829  -6.682 2.36e-11 ***
## contractTwo year                    -1.68952    0.20332  -8.309  < 2e-16 ***
## tech_supportNo internet service     -0.56187    0.16322  -3.442 0.000577 ***
## tech_supportYes                     -0.49818    0.09525  -5.230 1.69e-07 ***
## online_securityNo internet service        NA         NA      NA       NA
## online_securityYes                  -0.56688    0.09433  -6.010 1.86e-09 ***
## tenure                              -0.74179    0.05887 -12.601  < 2e-16 ***
## monthly_charges                      0.67501    0.06124  11.023  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6171.2  on 5281  degrees of freedom
## Residual deviance: 4552.0  on 5271  degrees of freedom
## AIC: 4574
##
## Number of Fisher Scoring iterations: 6
```

```
## A partir do valor relevante dos z values e
## do p-valor significativo a 0,01% da maioria
## das variáveis, sugere-se que estas foram uma boa
## escolha como preditores da variável dependente Churn
```

Avaliar a Precisão do Modelo

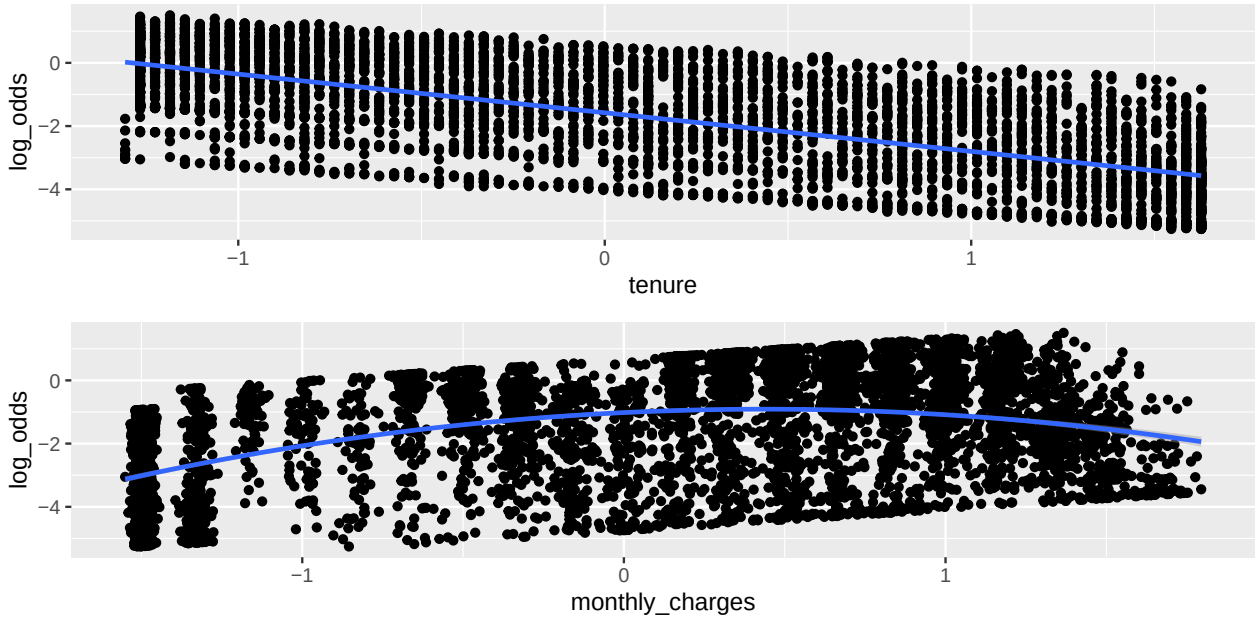
```
### Prever os resultados do modelo logit_churn aplicado
### ao testing_set

predicted_values <- predict(logit_churn,
                            newdata = testing_set,
                            type = 'response')

predicted_values <- ifelse(predicted_values > 0.5,
                           1,
                           0)

true_values <- as.numeric(summary(predicted_values == testing_set$churn)[['TRUE']])
```

```
accuracy <- true_values/nrow(testing_set)

### A acurácia é de 80.47%
```

Verificar Possíveis Violações de Premissas do Modelo de Regressão Logística

Linearidade dos Parâmetros

```
## Verificar a condição de linearidade entre as variáveis
## contínuas e o log das chances do modelo utilizando
## uma análise visual através de um gráfico de dispersão

log_odds <- logit_churn$linear.predictors

linearity_plot <- data.frame(log_odds = log_odds,
                             tenure = training_set$tenure,
                             monthly_charges = training_set$monthly_charges)

ggplot_tenure <- ggplot(linearity_plot, aes(x = tenure, y = log_odds)) +
  geom_point() +
  geom_smooth(method = 'lm')

ggplot_monthly_charges <- ggplot(linearity_plot, aes(x = monthly_charges, y = log_odds)) +
  geom_point() +
  geom_smooth(method = 'lm', formula = y ~ x + I(x^2))

grid.arrange(ggplot_tenure, ggplot_monthly_charges)
```




```
## Talvez uma outra especificação envolvendo
## o quadrado da variável monthly_charges descreva
## melhor essa relação não-linear
```

```r
## Nova especificação do modelo

logit_churn2 <- glm(churn ~ payment_method + contract + tech_support + online_security + tenure + monthl
                    family = binomial(link = 'logit'),
                    data = training_set)

summary(logit_churn2)
```

```
##
## Call:
## glm(formula = churn ~ payment_method + contract + tech_support +
##     online_security + tenure + monthly_charges + I(monthly_charges^2),
##     family = binomial(link = "logit"), data = training_set)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8756  -0.6900  -0.2964   0.7743   3.1788
##
## Coefficients: (1 not defined because of singularities)
##                                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)                          -0.89939    0.11323  -7.943 1.98e-15 ***
## payment_methodCredit card (automatic) -0.07436    0.12939  -0.575   0.5655
## payment_methodElectronic check         0.37770    0.10669   3.540   0.0004 ***
## payment_methodMailed check            -0.06751    0.12848  -0.525   0.5993
## contractOne year                      -0.79414    0.11936  -6.653 2.87e-11 ***
## contractTwo year                      -1.70660    0.20479  -8.333  < 2e-16 ***
## tech_supportNo internet service       -1.36595    0.22277  -6.132 8.70e-10 ***
## tech_supportYes                       -0.54537    0.09661  -5.645 1.65e-08 ***
## online_securityNo internet service          NA         NA      NA       NA
## online_securityYes                    -0.57075    0.09497  -6.010 1.86e-09 ***
## tenure                                -0.81399    0.06143 -13.251  < 2e-16 ***
## monthly_charges                        0.56448    0.06396   8.826  < 2e-16 ***
## I(monthly_charges^2)                   0.36600    0.07046   5.195 2.05e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6171.2  on 5281  degrees of freedom
## Residual deviance: 4525.2  on 5270  degrees of freedom
## AIC: 4549.2
##
## Number of Fisher Scoring iterations: 6
```

```r
predicted_values2 <- predict(logit_churn2,
                             newdata = testing_set,
                            type = 'response')

predicted_values2 <- ifelse(predicted_values2 > 0.5,
                            1,
                            0)

true_values2 <- as.numeric(summary(predicted_values2 == testing_set$churn)[['TRUE']])
```

```
accuracy2 <- true_values2/nrow(testing_set)

## Acurácia de 81,26%
```

Ausência de Multicolinearidade Perfeita

```
## Verificando se o grau de multicolinearidade
## entre as variáveis representa um potencial
## problema para a nossa regressão logística,
## um erro foi entregue. De fato deve haver variáveis
## cujos valores se aproximam de uma multicolinearidade
## perfeita. A checagem foi através de um erro da
## função vif.default(), que não está sendo reprodu-
## zido aqui por um erro ao tentar usar o knit
## do R Markdown.
```

```
## Vamos verificar com a função alias().

alias(logit_churn2)
```

Model : churn ~ payment_method + contract + tech_support + online_security + tenure + monthly_charges + I(monthly_charges^2)

Complete : (Intercept) online_securityNo internet service 0
payment_methodCredit card (automatic) online_securityNo internet service 0
payment_methodElectronic check online_securityNo internet service 0
payment_methodMailed check contractOne year online_securityNo internet service 0 0
contractTwo year online_securityNo internet service 0
tech_supportNo internet service online_securityNo internet service 1
tech_supportYes online_securityYes tenure online_securityNo internet service 0 0 0
monthly_charges I(monthly_charges^2) online_securityNo internet service 0 0

```
## Foi detectada uma correlação alta entre
## as dummies "online_security" e "tech_support".
## Vamos ter certeza analisando a estatística
## V's Cramer

tb <- xtabs(~training_set$online_security + training_set$tech_support)

cramerV(tb)
```

Cramer V 0.7302

```
## De fato, trata-se de uma correlação alta visto que
## 0.7302 se aproxima e muito de uma
## multicolinearidade perfeita (o valor 1).
## Vamos retirar a variável "online_security",
## visto que anteriormente já tivemos um
## problema por um factor dele não estar presente
## no conjunto de treino e o coeficiente resultante
## ser NA.
```

```
## Terceira especificação do modelo

logit_churn3 <- glm(churn ~ payment_method + contract +
```

```
                      tech_support + tenure + monthly_charges + I(monthly_charges^2),
                    family = binomial(link = 'logit'),
                    data = training_set)

summary(logit_churn3)
```

```
##
## Call:
## glm(formula = churn ~ payment_method + contract + tech_support +
##     tenure + monthly_charges + I(monthly_charges^2), family = binomial(link = "logit"),
##     data = training_set)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8829  -0.6959  -0.3060   0.7887   3.2030
##
## Coefficients:
##                                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)                          -1.05946    0.10985  -9.644  < 2e-16 ***
## payment_methodCredit card (automatic) -0.06329    0.12869  -0.492    0.623
## payment_methodElectronic check         0.42430    0.10591   4.006 6.17e-05 ***
## payment_methodMailed check            -0.07071    0.12760  -0.554    0.579
## contractOne year                      -0.85043    0.11861  -7.170 7.51e-13 ***
## contractTwo year                      -1.80687    0.20406  -8.854  < 2e-16 ***
## tech_supportNo internet service       -1.22000    0.22138  -5.511 3.57e-08 ***
## tech_supportYes                       -0.57770    0.09598  -6.019 1.76e-09 ***
## tenure                                -0.85643    0.06089 -14.064  < 2e-16 ***
## monthly_charges                        0.56709    0.06390   8.875  < 2e-16 ***
## I(monthly_charges^2)                   0.36287    0.06984   5.196 2.04e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6171.2  on 5281  degrees of freedom
## Residual deviance: 4562.2  on 5271  degrees of freedom
## AIC: 4584.2
##
## Number of Fisher Scoring iterations: 6
```

```
predicted_values3 <- predict(logit_churn3,
                             newdata = testing_set,
                             type = 'response')

predicted_values3 <- ifelse(predicted_values3 > 0.5,
                            1,
                            0)

true_values3 <- as.numeric(summary(predicted_values3 == testing_set$churn)[['TRUE']])

accuracy3 <- true_values3/nrow(testing_set)

accuracy3
```

```
## [1] 0.8131743
```

```
## Acurácia de 81,32%
```

Ausência de Outliers Influentes

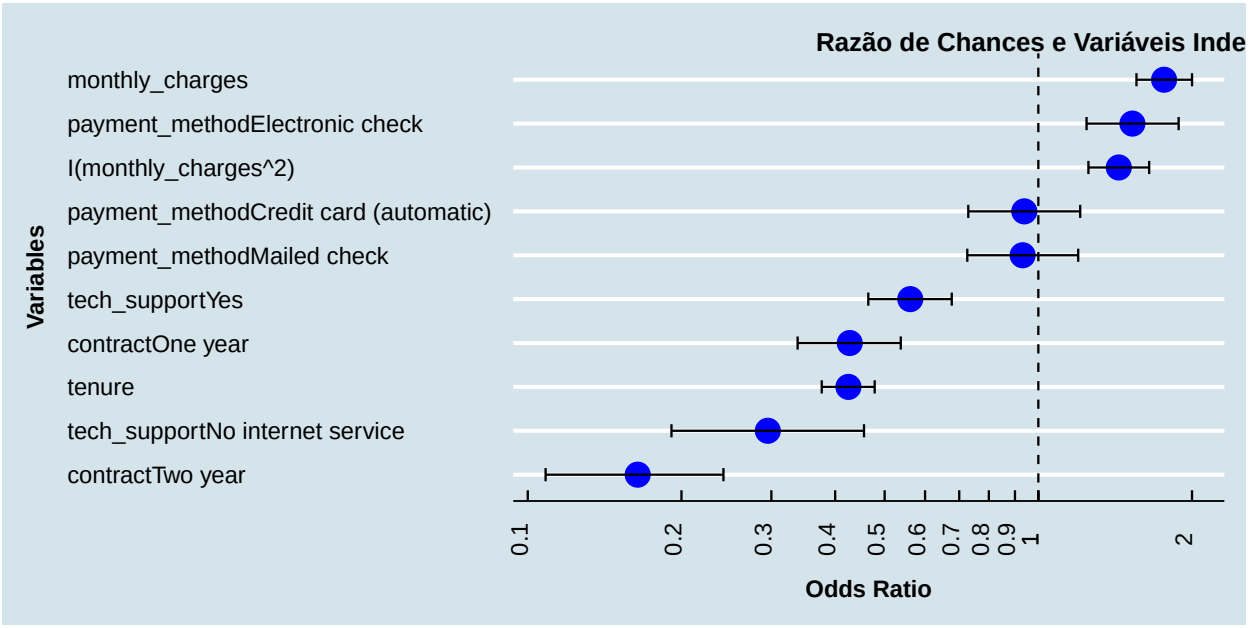
```
## Verificar se há outliers que possam estar sendo
## demasiadamente influentes na calibração do nosso
## modelo

check_outliers(logit_churn3,
               method = c('cook', 'iqr', 'zscore'))
```

```
## OK: No outliers detected.
## - Based on the following methods and thresholds: cook (3.09), iqr (1.7), zscore (0.94).
## - For variable: (Whole model)
```

```
## Ausência de outliers influentes confirmada.
```

Plotagem do Modelo

```
logit_churn3_plot <- odds_plot(logit_churn3,
                    title = 'Razão de Chances e Variáveis Independentes')

plot <- logit_churn3_plot$odds_plot

plot + ggthemes::theme_economist() + theme(axis.text.x = element_text(angle = 90)) + theme(axis.title.x
```



Considerações Finais

Após três especificações do modelo, chegamos a uma versão final dele com um grau de acurácia aperfeiçoado em relação às especificações anteriores.

A primeira especificação ocorreu utilizando seis variáveis preditivas: (i) “payment_method”, (ii) “contract”, (iii) “tech_support”, (iv) “online_security”, (v) “tenure” e (vi) “monthly_charges”.

Julgou-se necessária uma alteração por conta da relação não linear existente entre os log odds e a variável preditiva “monthly_charges”. Por isso, na segunda especificação, foi adotado um termo “I(monthly_charges^2)” para captar esta não linearidade.

A terceira e última especificação procurou corrigir um problema de multicolinearidade próxima de perfeita entre duas variáveis: “tech_support” e “online_security”. Como “online_security” chegou a apresentar alguns problemas por conta da forma com que foram separados os conjuntos de treino e de teste, ela foi a variável escolhida para ser removida.

Ainda foi checada a possibilidade de outliers influentes estarem afetando os parâmetros do nosso modelo, mas para três tipos de teste diferentes (Cook’s Distance, Interquartile Range e Z-Score), não foi detectado nenhum outlier de influência.

Respectivamente, a acurácia dos três modelos foi de 80,47%, 81,26% e 81,32%.