Customer Churn Analysis

Jonathan Ho

2024-09-01

Customer Churn Study: Part-1

1.1 Data Preprocessing

```r
# Import data
telcom_churn <- read_csv("Telco_Customer_Churn.csv")

# Check data types for customerID, Churn and SeniorCitizen columns
str(telcom_churn[c('customerID','Churn', 'SeniorCitizen')])
```

```
## tibble [7,043 x 3] (S3: tbl_df/tbl/data.frame)
##  $ customerID   : chr [1:7043] "7590-VHVEG" "5575-GNVDE" "3668-QPYBK" "7795-CFOCW" ...
##  $ Churn        : chr [1:7043] "No" "No" "Yes" "No" ...
##  $ SeniorCitizen: num [1:7043] 0 0 0 0 0 0 0 0 0 0 ...
```

```r
# Look at unique values for Churn and SeniorCitizen columns
print('unique values for Churn and SeniorCitizen columns')
```

```
## [1] "unique values for Churn and SeniorCitizen columns"
```

```r
unique(telcom_churn$Churn)
```

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

```r
unique(telcom_churn$SeniorCitizen)
```

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

```r
# Change datatypes for Churn and SeniorCitizen columns to factors
telcom_churn$Churn <- as.factor(telcom_churn$Churn)
telcom_churn$SeniorCitizen <- as.factor(telcom_churn$SeniorCitizen)

# Check data types again for Churn and SeniorCitizen columns
str(telcom_churn[c('Churn', 'SeniorCitizen')])
```

```
## tibble [7,043 x 2] (S3: tbl_df/tbl/data.frame)
##  $ Churn        : Factor w/ 2 levels "No","Yes": 1 1 2 1 2 2 1 1 2 1 ...
##  $ SeniorCitizen: Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

```r
# Check for missing values
colSums(is.na(telcom_churn)) # There are no missing values for customerID, Churn and SeniorCitizen.
```

```
##       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
```

```r
# We are good to proceed with the analysis
```

The datatypes for Churn and SeniorCitizen were changed to factors. There were also no missing values for columns customerID, Churn and SeniorCitizen. We thus proceed with the analysis.

1.2 Probability of customer churn

```
# Probability of customer churn
pi_hat <- mean(telcom_churn$Churn == "Yes")
pi_hat
```

```
## [1] 0.2653699
```

```
# Total number of customers
n <- nrow(telcom_churn)

# Critical value for 95% confidence
Z <- qnorm(p = 1-0.05/2, mean = 0, sd = 1)

# Lower bound
lower_bound <- pi_hat - Z*sqrt((pi_hat*(1-pi_hat))/(n+Z^2))
upper_bound <- pi_hat + Z*sqrt((pi_hat*(1-pi_hat))/(n+Z^2))

agresti_coull_ci <- c(lower_bound, upper_bound)
agresti_coull_ci
```

```
## [1] 0.2550610 0.2756787
```

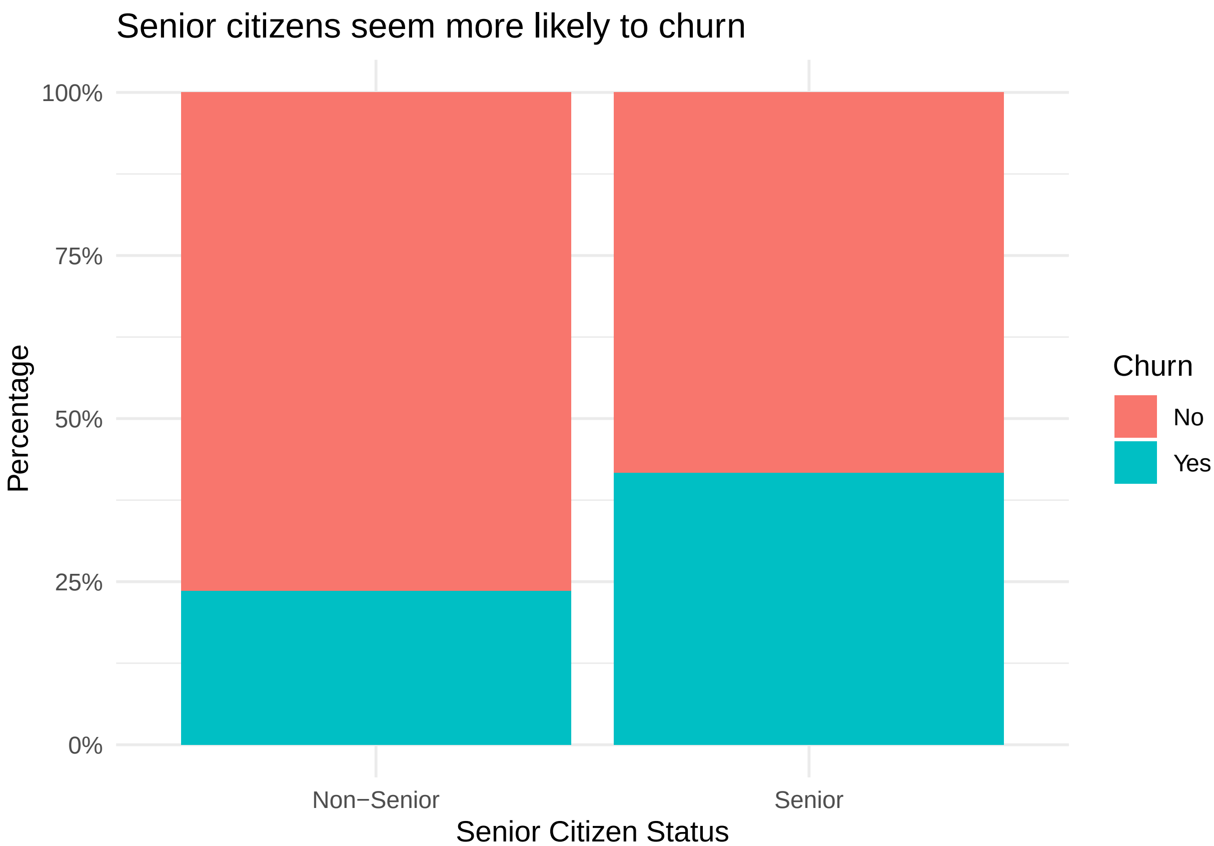
The probability of a customer churning, $\hat{\pi}$, is 0.265 (3 s.f.). The confidence interval is $(0.255, 0.276)$. This means that we are 95% confident that the true probability of a customer churning lies between 25.5% and 27.6%. Since the confidence interval does not include zero, we can say that $\hat{\pi}$ is statistically different from zero.

1.3 Comparison between senior and non-senior customers

```r
library(ggplot2)

senior_compare_plot <- ggplot(data = telcom_churn, aes(x = SeniorCitizen, fill = Churn)) +
  geom_bar(position = 'fill') +
  labs(title = 'Senior citizens seem more likely to churn', x = 'Senior Citizen Status',
       y = 'Percentage', fill = 'Churn') +
  scale_x_discrete(labels = c("Non-Senior", "Senior")) +
  scale_y_continuous(labels = scales::percent) +
  theme_minimal()

senior_compare_plot
```



As seen in the plot, a larger proportion of senior citizens churn.

1.4 Contingency table

```r
library(tidyverse)
library(stargazer)
contingency_table <- table(telcom_churn$SeniorCitizen, telcom_churn$Churn)
churn_probabilities <- prop.table(contingency_table, margin = 1)

# Convert the contingency table and churn probabilities to data frames for stargazer
contingency_df <- as.data.frame.matrix(contingency_table)
churn_probabilities_df <- as.data.frame.matrix(churn_probabilities)

# Set the row names for the data frames
rownames(contingency_df) <- c("Non-Senior", "Senior")
rownames(churn_probabilities_df) <- c("Non-Senior", "Senior")

# Rename the columns from "No" and "Yes" to "No Churn" and "Churn"
colnames(contingency_df) <- c("No Churn", "Churn")
colnames(churn_probabilities_df) <- c("No Churn", "Churn")

# Use stargazer to display the contingency table with row names on the left
stargazer(contingency_df, type = "latex", summary = FALSE,
          title = "Contingency Table: Senior Citizen vs Churn",
          rownames = TRUE,
          header = FALSE)
```

Table 1: Contingency Table: Senior Citizen vs Churn

	No Churn	Churn
Non-Senior	4,508	1,393
Senior	666	476

```r
# Use stargazer to display the churn probabilities table with row names on the left
stargazer(churn_probabilities_df, type = "latex", summary = FALSE,
          title = "Churn Probabilities: Senior Citizen vs Churn",
          rownames = TRUE,
          header = FALSE)
```

Table 2: Churn Probabilities: Senior Citizen vs Churn

	No Churn	Churn
Non-Senior	0.764	0.236
Senior	0.583	0.417

The probabilities do seem quite different, with Seniors about twice as likely to churn than Non-Seniors.

1.5 Confidence intervals for the difference of two probabilities

```r
n1 <- sum(contingency_table['1', ])  # Senior (row '1')
n2 <- sum(contingency_table['0', ])  # Non-Senior (row '0')
pi1_hat <- churn_probabilities['1', 'Yes']  # Proportion of seniors who churned
pi2_hat <- churn_probabilities['0', 'Yes']  # Proportion of non-seniors who churned

difference <- pi1_hat - pi2_hat

difference_wald_lower_bound <- difference -
  Z*sqrt((pi1_hat*(1-pi1_hat))/n1 + (pi2_hat*(1-pi2_hat))/n2)

difference_wald_upper_bound <- difference +
  Z*sqrt((pi1_hat*(1-pi1_hat))/n1 + (pi2_hat*(1-pi2_hat))/n2)

difference_agrestic_lower_bound <- difference -
  Z*sqrt((pi1_hat*(1-pi1_hat))/(n1+2) + (pi2_hat*(1-pi2_hat))/(n2+2))

difference_agrestic_upper_bound <- difference +
  Z*sqrt((pi1_hat*(1-pi1_hat))/(n1+2) + (pi2_hat*(1-pi2_hat))/(n2+2))

# Calculating Wald CI for difference
difference_wald_ci <- c(difference_wald_lower_bound, difference_wald_upper_bound)

# Calculating Agresti-Caffo CI for difference
difference_agresti_ci <- c(difference_agrestic_lower_bound,
                           difference_agrestic_upper_bound)
difference_wald_ci
```

```
## [1] 0.1501720 0.2113298
```

```r
difference_agresti_ci
```

```
## [1] 0.1501961 0.2113058
```

The Wald confidence interval for $\hat{\pi}_1 - \hat{\pi}_2$ is (0.1501720, 0.2113298). Zero is not within this interval, indicating that we can state with 95% confidence that seniors are more likely than non-seniors to churn.

The Agresti-Caffo confidence interval for $\hat{\pi}_1 - \hat{\pi}_2$ is (0.1501961, 0.2113058). Zero is also not within this interval, indicating that we can state with 95% confidence that seniors are more likely than non-seniors to churn.

Both methods yielded similar confidence intervals, and the same conclusion that seniors are more likely than non-seniors to churn.

1.6 Test for the difference of two probabilities

```r
n_plus <- n1 + n2
w_plus <- sum(contingency_table[, 'Yes'])
pi_bar <- w_plus / n_plus

# Calculate Z0 and p-value
Z0 <- (pi1_hat - pi2_hat) / sqrt(pi_bar * (1 - pi_bar) * ((1/n1) + (1/n2)))
p_value <- 2 * (1 - pnorm(abs(Z0)))
```

Using the Two-Sample Z-Test for Proportions, the Z-statistic Z_0 is 12.66302, with a p-value of 0 (< 0.05). Thus the difference in probabilities is highly significant.

1.7 Relative risks

```r
# Calculate relative risk
rr <- pi1_hat/pi2_hat
# Calculate log relative risk
log_rr <- log(rr)

w1 <- sum(contingency_table['1', 'Yes' ])  # Senior who churned (row '1')
w2 <- sum(contingency_table['0', 'Yes'])  # Non-Senior who churned (row '0')

# Calculate variance of log of relative risk
var_log_rr <- 1/w1 - 1/n1 +1/w2 - 1/n2

# Calculating Wald confidence interval for relative risk
rr_wald_ci_lower_bound <- exp(log_rr - Z*sqrt(var_log_rr))
rr_wald_ci_upper_bound <- exp(log_rr + Z*sqrt(var_log_rr))
rr_wald_ci <- c(rr_wald_ci_lower_bound,rr_wald_ci_upper_bound)
rr
```

```
## [1] 1.765694
```

```r
rr_wald_ci
```

```
## [1] 1.625802 1.917622
```

The probability of churning is 1.77 times as large for seniors than for non-seniors, with a 95% confidence interval ranging from 1.63 to 1.92. This is consistent with the findings in the previous sections, that seniors are more likely to churn than non-seniors.

1.8 Odds ratios

```r
# Calculating odds of senior churning
odds_senior_churn <- pi1_hat/(1 - pi1_hat)
# Calculating odds of non-senior churning
odds_non_senior_churn <- pi2_hat/(1 - pi2_hat)

#Calculating odds ratio, and log of odds ratio
odds_ratio <- odds_senior_churn/odds_non_senior_churn
log_odds_ratio <- log(odds_ratio)

# Calculating confidence interval for odds ratio
odds_ratio_ci_lower_bound <- exp(log_odds_ratio - Z*sqrt(1/w1+1/(n1-w1)+1/w2+1/(n2-w2)))
odds_ratio_ci_upper_bound <- exp(log_odds_ratio + Z*sqrt(1/w1+1/(n1-w1)+1/w2+1/(n2-w2)))
odds_ratio_ci <- c(odds_ratio_ci_lower_bound, odds_ratio_ci_upper_bound)
odds_ratio
```

```
## [1] 2.312946
```

```r
odds_ratio_ci
```

```
## [1] 2.026745 2.639563
```

The odds of a senior customer churning is 0.715 (3 s.f.), which is higher than the odds of a non-senior customer churning, which is 0.309 (3.s.f.).

The odds ratio is 2.31 (3 s.f.), with a 95% confidence interval of (2.03, 2.64) (3.s.f.). This means that the estimated odds of a customer churning is 2.31 times as large in the seniors group than in the non-seniors group, and we are 95% confident that the true odds ratio is between 2.03 and 2.64.

Customer Churn Study: Part-2

2.1 Data Preprocessing

```r
# Import data
telcom_churn <- read_csv("Telco_Customer_Churn.csv")

# Check data types for customerID, Churn, tenure, MonthlyCharges, and TotalCharges columns
str(telcom_churn[c('customerID','Churn', 'tenure', 'MonthlyCharges','TotalCharges')])
```

```
## tibble [7,043 x 5] (S3: tbl_df/tbl/data.frame)
##  $ customerID    : chr [1:7043] "7590-VHVEG" "5575-GNVDE" "3668-QPYBK" "7795-CFOCW" ...
##  $ Churn         : chr [1:7043] "No" "No" "Yes" "No" ...
##  $ tenure        : num [1:7043] 1 34 2 45 2 8 22 10 28 62 ...
##  $ MonthlyCharges: num [1:7043] 29.9 57 53.9 42.3 70.7 ...
##  $ TotalCharges  : num [1:7043] 29.9 1889.5 108.2 1840.8 151.7 ...
```

```r
table(telcom_churn$Churn)
```

```
##
##   No  Yes
## 5174 1869
```

```r
# Change datatypes for Churn to numeric. 0 for No, 1 for Yes
telcom_churn$Churn <- ifelse(telcom_churn$Churn == "Yes", 1, 0)

# Check for missing values
colSums(is.na(telcom_churn))
```

```
##       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
```

```r
# We have missing values for TotalCharges. Upon inspection, it is because these rows
# have '0' tenure.
# We will thus changes these missing values to 0.

telcom_churn$TotalCharges[is.na(telcom_churn$TotalCharges)] <- 0

# Check for missing values again
colSums(is.na(telcom_churn)) #No more missing values. We are good to proceed with analysis.
```

```
##       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                0
##            Churn
##                0
```

```r
# Check data types again
str(telcom_churn[c('customerID','Churn', 'tenure', 'MonthlyCharges','TotalCharges')])
```

```
## tibble [7,043 x 5] (S3: tbl_df/tbl/data.frame)
##  $ customerID    : chr [1:7043] "7590-VHVEG" "5575-GNVDE" "3668-QPYBK" "7795-CFOCW" ...
##  $ Churn         : num [1:7043] 0 0 1 0 1 1 0 0 1 0 ...
##  $ tenure        : num [1:7043] 1 34 2 45 2 8 22 10 28 62 ...
##  $ MonthlyCharges: num [1:7043] 29.9 57 53.9 42.3 70.7 ...
##  $ TotalCharges  : num [1:7043] 29.9 1889.5 108.2 1840.8 151.7 ...
```

The datatypes for Churn was changed to numeric; 0 for No, 1 for Yes. There were missing values for TotalCharges. Upon inspection, it is because these rows have '0' tenure; these customers were not on the service for long enough to have a TotalCharge. We will thus changes these missing values of TotalCharges to 0, and proceed with the analysis.

2.2 Maximum Likelihood

$$\pi_i = \frac{e^{\alpha+\beta\times\text{Tenure}_i}}{1+e^{\alpha+\beta\times\text{Tenure}_i}}$$
$$L(\alpha,\beta) = \prod_{i=1}^{n} \pi_i^{y_i}\left(1-\pi_i\right)^{1-y_i}$$

Thus,

$$\begin{aligned}
L(\alpha,\beta \mid \text{Data}) &= \prod_{i=1}^{n}\left(\frac{e^{\alpha+\beta\times\text{Tenure}_i}}{1+e^{\alpha+\beta\times\text{Tenure}_i}}\right)^{y_i}\left(1-\frac{e^{\alpha+\beta\times\text{Tenure}_i}}{1+e^{\alpha+\beta\times\text{Tenure}_i}}\right)^{1-y_i} \\
&= \prod_{i=1}^{n}\left(\frac{e^{\alpha+\beta\times\text{Tenure}_i}}{1+e^{\alpha+\beta\times\text{Tenure}_i}}\right)^{y_i}\left(\frac{1}{1+e^{\alpha+\beta\times\text{Tenure}_i}}\right)^{1-y_i} \\
&= \prod_{i=1}^{n}\left(\frac{e^{y_i(\alpha+\beta\times\text{Tenure}_i)}}{1+e^{\alpha+\beta\times\text{Tenure}_i}}\right)
\end{aligned}$$

2.3 Write and compute the log-likelihood

$$
\begin{aligned}
-\log\left(L(\alpha,\beta \mid \text{Data})\right) &= -\log\left(\prod_{i=1}^{n} \pi_i^{y_i}\left(1-\pi_i\right)^{1-y_i}\right) \\
&= -\sum_{i=1}^{n}\left(y_i \log \pi_i + (1-y_i)\log\left(1-\pi_i\right)\right) \\
&= -\sum_{i=1}^{n}\left(y_i \log\left(\frac{e^{\alpha+\beta\times\text{Tenure}_i}}{1+e^{\alpha+\beta\times\text{Tenure}_i}}\right) + (1-y_i)\log\left(1-\frac{e^{\alpha+\beta\times\text{Tenure}_i}}{1+e^{\alpha+\beta\times\text{Tenure}_i}}\right)\right) \\
&= -\sum_{i=1}^{n}\left(y_i \log\left(\frac{e^{\alpha+\beta\times\text{Tenure}_i}}{1+e^{\alpha+\beta\times\text{Tenure}_i}}\right) + (1-y_i)\log\left(\frac{1}{1+e^{\alpha+\beta\times\text{Tenure}_i}}\right)\right) \\
&= -\sum_{i=1}^{n}\big(y_i(\log e^{\alpha+\beta\times\text{Tenure}_i} - \log\left(1+e^{\alpha+\beta\times\text{Tenure}_i}\right)) + \log[(1+e^{\alpha+\beta\times\text{Tenure}_i})^{-1}] \\
&\quad -y_i \log\left[(1+e^{\alpha+\beta\times\text{Tenure}_i})^{-1}\right]\big) \\
&= -\sum_{i=1}^{n}\big(y_i(\alpha+\beta\times\text{Tenure}_i) - \log(1+e^{\alpha+\beta\times\text{Tenure}_i})\big)
\end{aligned}
$$

```r
# Create function for negative log-likelihood
neg_log_likelihood <- function(parameters, tenure, churn) {
  alpha <- parameters[1]
  beta <- parameters[2]

  pi_i <- exp(alpha + beta*tenure)/(1+exp(alpha + beta*tenure))

  log_likelihood <- sum(churn * log(pi_i) + (1- churn)*log(1- pi_i))

  return(-log_likelihood)
}
```

2.4 Compute the MLE of parameters

```r
# Use optim() function to find lowest possible value of negative log-likelihood
initial_values <- c(0,0)
result <- optim(
  par = initial_values,
  fn = neg_log_likelihood,
  tenure = telcom_churn$tenure,
  churn = telcom_churn$Churn,
)

print(result$par)
```

```
## [1]  0.02731012 -0.03877087
```

Thus, the values of the parameters for our MLE model are $\alpha = 0.02731012$ and $\beta = -0.03877087$.

2.5 Calculate a confidence interval

```
# Running optim again, with hessian matrix this time
initial_values <- c(0,0)
result <- optim(
  par = initial_values,
  fn = neg_log_likelihood,
  tenure = telcom_churn$tenure,
  churn = telcom_churn$Churn,
  hessian = TRUE
)

# Extract alpha and betas
alpha_mle <- result$par[1]
beta_mle <- result$par[2]

# Find variance of alpha and beta
cov_matrix <-  solve(result$hessian)
alpha_var <- cov_matrix[1,1]
beta_var <- cov_matrix[2,2]
alpha_var
```

```
## [1] 0.00178225
```

```
beta_var
```

```
## [1] 1.973791e-06
```

```
# Find standard errors of alpha and beta
alpha_se <- sqrt(alpha_var)
beta_se <- sqrt(beta_var)

# Create Z variable to store 1.96
Z <- qnorm(0.975)

# Create confidence intervals for alpha and beta
alpha_ci <- c(alpha_mle - Z*alpha_se, alpha_mle + Z*alpha_se)
beta_ci <- c(beta_mle - Z*beta_se, beta_mle + Z*beta_se)
alpha_ci
```

```
## [1] -0.0554331  0.1100533
```

```
beta_ci
```

```
## [1] -0.04152446 -0.03601728
```

The variance for α is 0.00178225. The 95% confidence interval for α is $(-0.0554331, 0.1100533)$ which includes zero. Thus, α is not statistically different than zero.

The variance for β is 1.973791e-06. The 95% confidence interval for β is $(-0.04152446, -0.03601728)$ which does not include zero. Thus, β is statistically different than zero.

2.6 Model comparison

```r
# Use glm to create model with tenure
logistic_model <- glm(formula = Churn ~ tenure,
                      family = binomial(link = "logit"),
                      data = telcom_churn)

summary(logistic_model)
```

```
##
## Call:
## glm(formula = Churn ~ tenure, family = binomial(link = "logit"),
##     data = telcom_churn)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.027313   0.042220   0.647    0.518
## tenure      -0.038767   0.001405 -27.589   <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: 8150.1  on 7042  degrees of freedom
## Residual deviance: 7191.9  on 7041  degrees of freedom
## AIC: 7195.9
##
## Number of Fisher Scoring iterations: 4
```

We see that α is 0.027313 (p-value = 0.518), and is not statistically different from zero. This value of $\alpha$is extremely close to our value of 0.02731012 that we obtained from the `optim()` function, and consistent with the fact that our 95% confidence interval for α included zero.

We also see that β is -0.038767 (p-value<2e-16), and is highly statistically significantly different from zero. This is also extremely close to our value of -0.03877087 that we obtained from the `optim()` function, and is consistent with the fact that our 95% confidence interval for β did not include zero.

The results align as both MLE through `optim()` and logistic regression through `glm()` are finding the parameters that maximize the log-likelihood of the observed data. MLE through `optim()` is simply modeling the log-odds of the outcome as a linear model. Slight differences are due to small differences in numerical optimization.

2.7 Extended Model, with Linear Effects

```r
# Create extended model with tenure + MonthlyCharges + TotalCharges
extended_model <- glm(formula = Churn ~ tenure + MonthlyCharges + TotalCharges,
                      family = binomial(link = "logit"),
                      data = telcom_churn)

summary(extended_model)
```

```
##
## Call:
## glm(formula = Churn ~ tenure + MonthlyCharges + TotalCharges,
##     family = binomial(link = "logit"), data = telcom_churn)
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -1.620e+00  1.171e-01  -13.83   <2e-16 ***
## tenure         -6.636e-02  5.437e-03  -12.21   <2e-16 ***
## MonthlyCharges  3.037e-02  1.715e-03   17.71   <2e-16 ***
## TotalCharges    1.384e-04  6.126e-05    2.26   0.0238 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8150.1  on 7042  degrees of freedom
## Residual deviance: 6389.2  on 7039  degrees of freedom
## AIC: 6397.2
##
## Number of Fisher Scoring iterations: 6
```

Our extended model is thus

$$\text{logit}(P(Churn)) = -0.06636 \times \text{Tenure} + 0.03037 \times \text{MonthlyCharges} + 0.0001384 \times \text{TotalCharges} - 1.62$$

with all estimates statistically significant (p-values < 0.05).

-0.06636 for Tenure's coefficient indicates that for every additional unit of Tenure, the odds of a customer churning decreases by $e^{0.06636}$. This means that longer tenure reduces the likelihood of churn, holding other variables constant.

0.03037 for MonthlyCharges' coefficient indicates that for every additional unit of MonthlyCharges, the odds of a customer churning increases by $e^{0.03037}$ This means that higher MonthlyCharges increases the likelihood of churn, holding other variables constant.

0.0001384 for TotalCharges' coefficient indicates that for every additional unit of TotalCharges, the odds of a customer churning increases by $e^{0.0001384}$ This means that higher TotalCharges increases the likelihood of churn, holding other variables constant.

The intercept -1.62 represents the log-odds of churn when all other independent variables (Tenure, MonthlyCharges, and TotalCharges) are equal to zero. It also indicates that in a hypothetical scenario with zero tenure, monthly charges, and total charges, the baseline probability of churn would be approximately

$$\frac{e^{-1.62}}{1 + e^{-1.62}} = 0.165$$

, to three significant figures.

2.8 Likelihood Ratio Tests

```r
# Import car
library(car)

# Run LR test
Anova(extended_model, test = "LR")
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: Churn
##                LR Chisq Df Pr(>Chisq)
## tenure           187.36  1    < 2e-16 ***
## MonthlyCharges   348.10  1    < 2e-16 ***
## TotalCharges       5.19  1    0.02277 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We see that the coefficients for all explanatory variables tenure, MonthlyCharges, and TotalCharges are statistically significant (< 0.05), indicating that they are all significant in predicting the probability of churn.

2.9 Effect of change in Monthly payments

```r
# Find standard deviation of MonthlyCharges
MonthlyCharges_sd <- sd(telcom_churn$MonthlyCharges)

# Extract coefficient of MonthlyCharges from extended_model
MonthlyCharges_coef <- coef(extended_model)["MonthlyCharges"]

# Find the increase in OR for one SD increase in MonthlyCharges
OR_SD_increase <- exp(MonthlyCharges_coef * MonthlyCharges_sd)
MonthlyCharges_sd
```

```
## [1] 30.09005
```

```r
OR_SD_increase
```

```
## MonthlyCharges
##       2.493668
```

```r
#Find Wald CI
MonthlyCharges_coef_se <- summary(extended_model)$coefficients["MonthlyCharges", "Std. Error"]
wald_ci_lower <- exp(MonthlyCharges_sd*MonthlyCharges_coef -
                       MonthlyCharges_sd*Z*MonthlyCharges_coef_se)
wald_ci_upper <- exp(MonthlyCharges_sd*MonthlyCharges_coef +
                       MonthlyCharges_sd*Z*MonthlyCharges_coef_se)
wald_ci <- c(wald_ci_lower,wald_ci_upper)
wald_ci
```

```
## MonthlyCharges MonthlyCharges
##       2.253821       2.759039
```

The odds of a customer churning increases by 2.49, with a 95% confidence interval of (2.253821, 2.759039) when MonthlyCharges increase by one standard deviation of approximately $30.09.

2.10 Confidence Interval for the Probability of Success

```r
### First calculate the Wald confidence interval

# Calculate mean values for tenure, MonthlyCharges, and TotalCharges
tenure_mean <- mean(telcom_churn$tenure)
MonthlyCharges_mean <- mean(telcom_churn$MonthlyCharges)
TotalCharges_mean <- mean(telcom_churn$TotalCharges)

# Create a data frame with the average values
average_values <- data.frame(tenure = tenure_mean,
                             MonthlyCharges = MonthlyCharges_mean,
                             TotalCharges = TotalCharges_mean)

# Predict the probability for the mean values (log-odds scale)
predicted_log_odds <- predict(extended_model, newdata = average_values, type = "link")

# Get the standard error for the log-odds
predicted_se <- predict(extended_model, newdata = average_values,
                        type = "link", se.fit = TRUE)$se.fit

# Calculate the 95% confidence interval for the log-odds
Z <- qnorm(0.975) # 95% confidence
lower_log_odds <- predicted_log_odds - Z * predicted_se
upper_log_odds <- predicted_log_odds + Z * predicted_se

# onvert log-odds to probability using the correct formula
predicted_prob <- 1 / (1 + exp(-predicted_log_odds))  # Predicted probability
ci_lower <- 1 / (1 + exp(-lower_log_odds))            # Lower bound probability
ci_upper <- 1 / (1 + exp(-upper_log_odds))            # Upper bound probability

predicted_prob
```

```
##        1
## 0.184607
```

```r
c(ci_lower, ci_upper)
```

```
##         1         1
## 0.1722661 0.1976208
```

```r
### Use mcprofile package to calculate profile likelihood confidence interval
K <- matrix(c(1, tenure_mean, MonthlyCharges_mean, TotalCharges_mean), nrow = 1)

# Calculate the profile likelihood for the linear combination
linear.combo <- mcprofile(object = extended_model, CM = K)

# Get the profile likelihood confidence interval
ci.logit.profile <- confint(object = linear.combo, level = 0.95)

# Convert the log-odds confidence interval to probability
prob_ci <- exp(ci.logit.profile$confint) / (1 + exp(ci.logit.profile$confint))

# Print the probability confidence interval
print(prob_ci)
```

```
##       lower     upper
## 1 0.1720363 0.1973745
```

We first calculate the Wald Confidence Interval to be (0.1722661, 0.1976208). We also used the `mcprofile` package and calculated the profile likelihood confidence interval to be (0.1720363, 0.1973745). They are extremely similar, and thus we use the profile likelihood confidence interval. We thus conclude that the predicted probability of a customer churning for the mean Tenure, MonthlyCharges and TotalCharges is 0.184607 with a 95% confidence interval of (0.1720363, 0.1973745).

Customer Churn Study: Part-3

3.1 Data Preprocessing

```r
# Import data
telcom_churn <- read_csv("Telco_Customer_Churn.csv")
telcom_churn <- as.data.frame(telcom_churn)
# Check data types for all variables
str(telcom_churn)
```

```
## 'data.frame':    7043 obs. of  21 variables:
##  $ customerID      : chr  "7590-VHVEG" "5575-GNVDE" "3668-QPYBK" "7795-CFOCW" ...
##  $ gender          : chr  "Female" "Male" "Male" "Male" ...
##  $ SeniorCitizen   : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Partner         : chr  "Yes" "No" "No" "No" ...
##  $ Dependents      : chr  "No" "No" "No" "No" ...
##  $ tenure          : num  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" ...
```

```r
# Change datatypes for Gender, SeniorCitizen, Partner, Dependents, PhoneService,
# MultipleLInes, InternetService, OnlineSecurity, OnlineBackup, DeviceProtection
# TechSupport, StreamingTV, StreamingMovies, Contract, PaperlessBilling, PaymentMethod to factors

cols_to_factor <- c('gender', 'SeniorCitizen', 'Partner', 'Dependents', 'PhoneService',
                    'MultipleLines', 'InternetService', 'OnlineSecurity', 'OnlineBackup',
                    'DeviceProtection', 'TechSupport', 'StreamingTV', 'StreamingMovies',
                    'Contract', 'PaperlessBilling', 'PaymentMethod')

telcom_churn[cols_to_factor] <- lapply(telcom_churn[cols_to_factor], as.factor)

# Change datatypes for Churn to numeric. 0 for No, 1 for Yes
telcom_churn$Churn <- ifelse(telcom_churn$Churn == "Yes", 1, 0)

# Set reference for gender
telcom_churn$gender<-relevel(telcom_churn$gender, ref="Male")

# Check for missing values
colSums(is.na(telcom_churn))
```

```
##       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
```

```r
# We have missing values for TotalCharges. Upon inspection, it is because these rows
# have '0' tenure. # We will thus changes these missing values to 0.

telcom_churn$TotalCharges[is.na(telcom_churn$TotalCharges)] <- 0

# Check for missing values again
colSums(is.na(telcom_churn)) #No more missing values. We are good to proceed with analysis.
```

```
##       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                0 
##            Churn 
##                0
```

```r
# Check data types again
str(telcom_churn)
```

```
## 'data.frame':    7043 obs. of  21 variables:
##  $ customerID      : chr  "7590-VHVEG" "5575-GNVDE" "3668-QPYBK" "7795-CFOCW" ...
##  $ gender          : Factor w/ 2 levels "Male","Female": 2 1 1 1 2 2 1 2 2 1 ...
##  $ SeniorCitizen   : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Partner         : Factor w/ 2 levels "No","Yes": 2 1 1 1 1 1 1 1 2 1 ...
##  $ Dependents      : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 2 1 1 2 ...
##  $ tenure          : num  1 34 2 45 2 8 22 10 28 62 ...
##  $ PhoneService    : Factor w/ 2 levels "No","Yes": 1 2 2 1 2 2 2 1 2 2 ...
##  $ MultipleLines   : Factor w/ 3 levels "No","No phone service",..: 2 1 1 2 1 3 3 2 3 1 ...
##  $ InternetService : Factor w/ 3 levels "DSL","Fiber optic",..: 1 1 1 1 2 2 2 1 2 1 ...
##  $ OnlineSecurity  : Factor w/ 3 levels "No","No internet service",..: 1 3 3 3 1 1 1 3 1 3 ...
##  $ OnlineBackup    : Factor w/ 3 levels "No","No internet service",..: 3 1 3 1 1 1 3 1 1 3 ...
##  $ DeviceProtection: Factor w/ 3 levels "No","No internet service",..: 1 3 1 3 1 3 1 1 3 1 ...
##  $ TechSupport     : Factor w/ 3 levels "No","No internet service",..: 1 1 1 3 1 1 1 1 3 1 ...
##  $ StreamingTV     : Factor w/ 3 levels "No","No internet service",..: 1 1 1 1 1 3 3 1 3 1 ...
##  $ StreamingMovies : Factor w/ 3 levels "No","No internet service",..: 1 1 1 1 1 3 1 1 3 1 ...
##  $ Contract        : Factor w/ 3 levels "Month-to-month",..: 1 2 1 2 1 1 1 1 1 2 ...
##  $ PaperlessBilling: Factor w/ 2 levels "No","Yes": 2 1 2 1 2 2 2 1 2 1 ...
##  $ PaymentMethod   : Factor w/ 4 levels "Bank transfer (automatic)",..: 3 4 4 1 3 3 2 4 3 1 ...
##  $ MonthlyCharges  : num  29.9 57 53.9 42.3 70.7 ...
##  $ TotalCharges    : num  29.9 1889.5 108.2 1840.8 151.7 ...
```

```
##  $ Churn           : num  0 0 1 0 1 1 0 0 1 0 ...
```

The datatypes for Gender, SeniorCitizen, Partner, Dependents, PhoneService, MultipleLInes, InternetService, OnlineSecurity, OnlineBackup, DeviceProtection, TechSupport, StreamingTV, StreamingMovies, Contract, PaperlessBilling, and PaymentMethod were change to factor. Churn was changed to numeric; 0 for No, 1 for Yes. There were missing values for TotalCharges. Upon inspection, it is because these rows have '0' tenure; these customers were not on the service for long enough to have a TotalCharge. We will thus changes these missing values of TotalCharges to 0, and proceed with the analysis.

3.2 Estimate a logistic regression

```
m1 <- glm(formula = Churn ~ tenure + MonthlyCharges + TotalCharges + SeniorCitizen +
            gender, data = telcom_churn, family = binomial(link = "logit"))
m2 <- glm(formula = Churn ~ tenure + MonthlyCharges + TotalCharges + SeniorCitizen +
            gender + I(tenure^2) + I(MonthlyCharges^2) + I(TotalCharges^2),
          data = telcom_churn, family = binomial(link = "logit"))
m3 <- glm(formula = Churn ~ tenure + MonthlyCharges + TotalCharges + SeniorCitizen +
           gender + I(tenure^2) + I(MonthlyCharges^2) + I(TotalCharges^2) +
           SeniorCitizen:tenure+SeniorCitizen:MonthlyCharges+SeniorCitizen:TotalCharges+
           gender:tenure+gender:MonthlyCharges+gender:TotalCharges,
         data = telcom_churn, family = binomial(link = "logit"))
```

```
stargazer(m1, m2, m3,
          title = "Comparing Logistic Regression Models",
          align = TRUE,
          type = "latex",
          star.cutoffs = c(0.05, 0.01, 0.001),
          header = FALSE)
```

Table 3: Comparing Logistic Regression Models

	Dependent variable:		
	Churn		
	(1)	(2)	(3)
tenure	-0.067^{***}	-0.123^{***}	-0.130^{***}
	(0.005)	(0.013)	(0.014)
MonthlyCharges	0.028^{***}	0.024^{***}	0.018^{**}
	(0.002)	(0.007)	(0.007)
TotalCharges	0.0001^{*}	0.001^{***}	0.001^{***}
	(0.0001)	(0.0002)	(0.0002)
SeniorCitizen1	0.633^{***}	0.638^{***}	1.499^{***}
	(0.079)	(0.080)	(0.399)
genderFemale	0.003	0.006	-0.229
	(0.062)	(0.062)	(0.234)
I(tenureˆ2)		0.001^{***}	0.001^{***}
		(0.0001)	(0.0001)
I(MonthlyChargesˆ2)		0.00003	0.0001
		(0.0001)	(0.0001)
I(TotalChargesˆ2)		-0.00000^{***}	-0.00000^{***}
		(0.00000)	(0.00000)
tenure:SeniorCitizen1			0.013
			(0.013)
MonthlyCharges:SeniorCitizen1			-0.013^{*}
			(0.005)
TotalCharges:SeniorCitizen1			-0.0001
			(0.0002)
tenure:genderFemale			0.009
			(0.010)
MonthlyCharges:genderFemale			0.006
			(0.003)
TotalCharges:genderFemale			-0.0002
			(0.0001)
Constant	-1.605^{***}	-1.278^{***}	-1.151^{***}
	(0.121)	(0.199)	(0.225)
Observations	7,043	7,043	7,043
Log Likelihood	-3,162.904	-3,145.941	-3,133.919
Akaike Inf. Crit.	6,337.808	6,309.882	6,297.838
Note:			*p<0.05; **p<0.01; ***p<0.001

3.3 Test a hypothesis: linear effects

```r
Anova(mod = m1, test = 'LR')
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: Churn
##                LR Chisq Df Pr(>Chisq)
## tenure          189.176  1  < 2.2e-16 ***
## MonthlyCharges  294.245  1  < 2.2e-16 ***
## TotalCharges      5.537  1    0.01862 *
## SeniorCitizen    63.351  1   1.73e-15 ***
## gender            0.002  1    0.96426
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The variables tenure, MonthlyCharges, TotalCharges, and SeniorCitizen are statistically significant, meaning that there is sufficient evidence indicating that including them in our model helps us better predict the probability of Churn. The variable gender, however, is not statistically significant. This means that there is insufficient evidence indicating that including gender in our model helps us better predict the probability of Churn.

3.4 Test a hypothesis: Non linear effect

```
Anova(m2, test = "LR")
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: Churn
##                     LR Chisq Df Pr(>Chisq)
## tenure               100.286  1  < 2.2e-16 ***
## MonthlyCharges        13.206  1  0.0002791 ***
## TotalCharges          12.812  1  0.0003443 ***
## SeniorCitizen         63.724  1  1.431e-15 ***
## gender                 0.009  1  0.9233482
## I(tenure^2)           31.982  1  1.556e-08 ***
## I(MonthlyCharges^2)    0.373  1  0.5415239
## I(TotalCharges^2)     15.583  1  7.897e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Running the LRT for Model 2, we see that the quadratic terms for tenure and TotalCharges are statistically significant and should be included in our model, even after we have included the linear terms tenure, MonthlyCharges, and TotalCharges. We also note that the quadratic term for MonthlyCharges is not statistically significant after including the linear term MonthlyCharges.

```
Anova(m3, test = "LR")
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: Churn
##                              LR Chisq Df Pr(>Chisq)
## tenure                        101.709  1  < 2.2e-16 ***
## MonthlyCharges                 11.666  1  0.0006365 ***
## TotalCharges                   14.123  1  0.0001712 ***
## SeniorCitizen                  64.228  1  1.108e-15 ***
## gender                          0.019  1  0.8905351
## I(tenure^2)                    29.810  1  4.765e-08 ***
## I(MonthlyCharges^2)             1.421  1  0.2331931
## I(TotalCharges^2)              15.832  1  6.921e-05 ***
## tenure:SeniorCitizen            0.871  1  0.3507512
## MonthlyCharges:SeniorCitizen    5.804  1  0.0159865 *
## TotalCharges:SeniorCitizen      0.227  1  0.6336539
## tenure:gender                   0.761  1  0.3828609
## MonthlyCharges:gender           3.061  1  0.0801781 .
## TotalCharges:gender             2.990  1  0.0837831 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Running the LRT for Model 3, we see that the quadratic terms for tenure and TotalCharges are still statistically significant and should be included in our model, even after we have included the linear terms tenure, MonthlyCharges, and TotalCharges and all the various interaction terms. We also note that the quadratic term for MonthlyCharges is not statistically significant after including the linear term MonthlyCharges and interaction terms that include MonthlyCharges.

3.5 Test a hypothesis: Total effect of gender

```r
Anova(m3, test = "LR")
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: Churn
##                              LR Chisq Df Pr(>Chisq)
## tenure                        101.709  1  < 2.2e-16 ***
## MonthlyCharges                 11.666  1  0.0006365 ***
## TotalCharges                   14.123  1  0.0001712 ***
## SeniorCitizen                  64.228  1  1.108e-15 ***
## gender                          0.019  1  0.8905351
## I(tenure^2)                    29.810  1  4.765e-08 ***
## I(MonthlyCharges^2)             1.421  1  0.2331931
## I(TotalCharges^2)              15.832  1  6.921e-05 ***
## tenure:SeniorCitizen            0.871  1  0.3507512
## MonthlyCharges:SeniorCitizen    5.804  1  0.0159865 *
## TotalCharges:SeniorCitizen      0.227  1  0.6336539
## tenure:gender                   0.761  1  0.3828609
## MonthlyCharges:gender           3.061  1  0.0801781 .
## TotalCharges:gender             2.990  1  0.0837831 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Running the LRT on Model 3, we see that the main effect of gender on churn is not significant. We also note that the interaction effects involving gender are not significant at the 0.05 level, but two of them MonthlyCharges:gender and TotalCharges:gender are weakly significant at a 0.1 level. However, at 0.05 level, we thus conclude that gender, together with the interaction terms involving gender, are all not statistically significant in helping us predict churn.

3.6 Senior V.S. non-senior customers

```
m4 <- glm(formula = Churn ~ tenure + MonthlyCharges + TotalCharges + SeniorCitizen +
            I(tenure^2) + I(TotalCharges^2) +
           SeniorCitizen:MonthlyCharges,
         data = telcom_churn, family = binomial(link = "logit"))

tenure_avg = mean(telcom_churn$tenure)
MonthlyCharges_avg = mean(telcom_churn$MonthlyCharges)
TotalCharges_avg = mean(telcom_churn$TotalCharges)

average_values_senior_compare <- data.frame(tenure = tenure_avg,
                             MonthlyCharges = MonthlyCharges_avg,
                             TotalCharges = TotalCharges_avg, SeniorCitizen = c("1","0"))

predicted_prob <- predict(m4, newdata = average_values_senior_compare, type = "response")
names(predicted_prob) <- c("Senior", "Non-Senior")
predicted_prob
```

```
##     Senior Non-Senior
##  0.2814961  0.1482713
```

The probability of a Senior Citizen with average tenure, MonthlyCharges and TotalCharges churning is 0.2814961, and the probability of a Non-Senior Citizen with the same averages churning is 0.1482713. The relative risk is thus 1.90, which means that churning is 1.90 times as likely for Senior Citizens with average tenure, MonthlyCharges and TotalCharges than for non-Senior Citizens with average tenure, MonthlyCharges and TotalCharges.

3.7 Construct a confidence interval

```
Z <- qnorm(0.975)
df_predict_1 <- data.frame(tenure = 55.00,
                           MonthlyCharges = 89.86,
                           TotalCharges = 3794.7, SeniorCitizen = "0")

df_predict_2 <- data.frame(tenure = 29.00,
                           MonthlyCharges = 18.25,
                           TotalCharges = 401.4, SeniorCitizen = "1")

predict_1 <- predict(m4, newdata = df_predict_1, type = "response", se = TRUE)
predict_2 <- predict(m4, newdata = df_predict_2, type = "response", se = TRUE)

ci_predict_1 <- c(predict_1$fit - Z*predict_1$se, predict_1$fit + Z*predict_1$se)
ci_predict_2 <- c(predict_2$fit - Z*predict_2$se, predict_2$fit + Z*predict_2$se)

predict_1$fit
```

```
##         1
## 0.1249331
```

```
ci_predict_1
```

```
##         1         1
## 0.1045881 0.1452781
```

```
predict_2$fit
```

```
##          1
## 0.09161642
```

```
ci_predict_2
```

```
##          1          1
## 0.05017314 0.13305970
```

For a customer with the profile tenure = 55.00, MonthlyCharges = 89.86, TotalCharges = 3794.7, SeniorCitizen = "No", the probability of churn is 0.1249331 with a 95% confidence interval that it is in the range (0.1045881, 0.1452781).

For a customer with the profile tenure = 29.00, MonthlyCharges = 18.25, TotalCharges = 401.4, SeniorCitizen = "Yes", the probability of churn is 0.09161642 with a 95% confidence interval that it is in the range (0.05017314, 0.13305970).