# Titanic survives with logistic regression in R

Pichayud Udomsud

2023-11-01

## Use titanic dataset for train logistic regression

### Explain variables in titanic data set

| Variable | Explain |
|---|---|
| PassengerId | A unique identifier for each passenger. |
| Survived | A binary variable indicating whether the passenger survived (1) or not (0). |
| Pclass | The passenger class (1, 2, or 3). |
| Name | The full name of the passenger. |
| Sex | The gender of the passenger (male or female). |
| Age | The age of the passenger in years. |
| SibSp | The number of siblings/spouses aboard the Titanic. |
| Fare | The fare paid for the ticket. |
| Cabin | The cabin number (if applicable). |
| Embarked | The port of embarkation (C = Cherbourg, Q = Queenstown, S = Southampton). |
| Home.Dest | The passenger's home destination. |
| AgeCategory | A categorical variable based on the passenger's age (Child, Adult). |
| Title | The passenger's title (Mr, Miss, Mrs, etc.). |
| FamilySize | A categorical variable based on the size of the passenger's family (1, 2, 3+). |
| FareBand | A categorical variable based on the fare paid for the ticket (1, 2, 3, 4, 5, 6). |

### Install library tidyverse , patchwork

```
library(tidyverse)
library(titanic)
```

### Check data

```
head(titanic_train)
```

```
##   PassengerId Survived Pclass
## 1           1        0      3
## 2           2        1      1
## 3           3        1      3
## 4           4        1      1
## 5           5        0      3
## 6           6        0      3
##                                                   Name    Sex Age SibSp Parch
## 1                              Braund, Mr. Owen Harris   male  22     1     0
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1     0
## 3                               Heikkinen, Miss. Laina female  26     0     0
## 4        Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1     0
## 5                             Allen, Mr. William Henry   male  35     0     0
## 6                                     Moran, Mr. James   male  NA     0     0
##             Ticket    Fare Cabin Embarked
## 1        A/5 21171  7.2500              S
## 2         PC 17599 71.2833   C85        C
## 3 STON/O2. 3101282  7.9250              S
## 4           113803 53.1000  C123        S
## 5           373450  8.0500              S
## 6           330877  8.4583              Q
```

It seems that this data has many variables, and some variables have NA values. Therefore, we need to clean the data first.

## Clean data that have NA (Null)

```
cat("Data before cleaned :",nrow(titanic_train))
```

```
## Data before cleaned : 891
```

```
titanic_clean <- na.omit(titanic_train)
cat("\nData after cleaned  :",nrow(titanic_clean))
```

```
##
## Data after cleaned  : 714
```

After cleaning the data,we will have complete all columns that can be used to train the model. This means that there will be no missing values or invalid entries in the data, which will improve the accuracy of the model.

## Check data which is cleaned that its still have NA

```
titanic_clean %>%
  summarise(sum(is.na(.)))
```

```
##   sum(is.na(.))
## 1             0
```

We can summarize that there are no missing values (NA) in the dataset

## Before split data for test model

### Check variable that can affect survive

```
glimpse(titanic_clean)
```

```
## Rows: 714
## Columns: 12
## $ PassengerId <int> 1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 19~
## $ Survived    <int> 0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 1~
## $ Pclass      <int> 3, 1, 3, 1, 3, 1, 3, 3, 2, 3, 1, 3, 3, 3, 2, 3, 3, 2, 2, 3~
## $ Name        <chr> "Braund, Mr. Owen Harris", "Cumings, Mrs. John Bradley (Fl~
## $ Sex         <chr> "male", "female", "female", "female", "male", "male", "mal~
## $ Age         <dbl> 22, 38, 26, 35, 35, 54, 2, 27, 14, 4, 58, 20, 39, 14, 55, ~
## $ SibSp       <int> 1, 1, 0, 1, 0, 0, 3, 0, 1, 1, 0, 0, 1, 0, 0, 4, 1, 0, 0, 0~
## $ Parch       <int> 0, 0, 0, 0, 0, 0, 1, 2, 0, 1, 0, 0, 5, 0, 0, 1, 0, 0, 0, 0~
## $ Ticket      <chr> "A/5 21171", "PC 17599", "STON/O2. 3101282", "113803", "37~
## $ Fare        <dbl> 7.2500, 71.2833, 7.9250, 53.1000, 8.0500, 51.8625, 21.0750~
## $ Cabin       <chr> "", "C85", "", "C123", "", "E46", "", "", "", "G6", "C103"~
## $ Embarked    <chr> "S", "C", "S", "S", "S", "S", "S", "S", "C", "S", "S", "S"~
```

## Change gender from string to factor

```
titanic_clean$Sex <- factor(titanic_clean$Sex ,
                            level = c("male","female"),
                            labels = c(0,1))
titanic_clean$Sex
```

```
##   [1] 0 1 1 1 0 0 0 1 1 1 1 0 0 1 1 0 1 0 0 1 0 1 1 0 0 0 0 0 0 1 1 1 1 1 1 1 0
##  [38] 0 1 1 0 1 0 1 0 0 1 0 0 1 0 1 0 0 1 0 0 0 0 0 1 0 0 0 1 1 0 1 0 0 0 0 0 0
##  [75] 0 0 1 0 1 0 0 0 0 1 0 0 1 0 1 1 0 0 0 0 1 0 0 1 0 0 0 0 0 0 1 1 0 0 1 0 0
## [112] 0 1 1 0 0 0 0 1 0 0 0 1 0 0 0 1 0 0 1 0 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 1
## [149] 0 0 0 1 0 1 0 1 1 0 1 0 0 0 0 1 0 0 1 0 0 1 0 0 1 1 0 1 0 0 0 0 0 0 0 0 0
## [186] 1 0 0 1 0 0 1 0 0 0 0 0 0 1 1 0 0 1 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 0 1 0 1
## [223] 1 0 1 0 0 0 0 0 0 0 0 1 1 1 0 1 0 0 1 1 0 0 1 0 1 1 1 1 0 0 1 1 0 1 1 0 0
## [260] 1 1 1 0 1 1 1 0 0 0 0 1 0 0 0 1 0 0 0 1 1 0 0 0 0 0 0 1 1 0 0 1 0 0 1 1 0
## [297] 0 0 0 1 1 0 0 0 1 1 0 1 0 0 1 1 0 0 0 1 1 0 1 0 0 1 0 0 1 0 1 0 0 0 0 1 0
## [334] 1 1 0 1 0 0 1 0 1 1 0 0 1 0 0 1 1 1 0 0 1 0 0 1 0 1 0 1 0 0 0 0 0 0 1 0 0
## [371] 0 0 0 0 1 0 1 1 1 0 0 0 1 0 0 1 0 1 0 0 0 0 0 0 0 1 1 0 0 1 1 1 0 1 0 0 0
## [408] 0 1 0 0 1 1 0 1 0 1 0 1 0 0 1 0 1 1 0 1 1 1 1 1 0 0 0 1 0 0 0 0 0 1 0 1 1
## [445] 1 0 0 0 0 1 0 0 1 0 0 0 1 1 0 1 1 0 0 1 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 1
## [482] 1 1 0 1 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0 1 1 0 0 1 0 1 1 1 0 0 0 1 1 0 1 0
## [519] 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0
## [556] 1 0 1 0 0 0 1 0 1 1 0 0 0 0 1 1 0 1 0 0 0 0 0 1 0 1 1 0 0 0 0 1 0 0 1 0 0
## [593] 0 0 1 0 0 1 0 0 0 1 0 0 0 0 1 0 0 1 0 1 1 0 0 0 1 1 0 1 1 1 1 0 0 0 1 0 0
## [630] 0 0 0 0 1 1 0 1 0 1 0 0 0 0 0 1 0 1 0 0 0 1 0 1 0 0 0 1 0 0 1 0 0 1 1 0 0
## [667] 0 1 0 0 0 0 1 0 0 0 0 0 0 0 1 1 1 1 1 0 1 0 0 1 0 1 1 0 0 0 1 0 0 1 1 0 0
## [704] 1 1 0 1 0 0 1 0 1 0 0
## Levels: 0 1
```

## Split data for test model

### Use sample for sampling titanic passenger

### Use 75% for train model and 25% for test model

```
set.seed(10)
allrow <- nrow(titanic_clean)
titanic_random <- sample(allrow, size = allrow*0.75)
titanic_for_train <- titanic_clean[titanic_random,]
titanic_for_test <- titanic_clean[-titanic_random,]
```

## Use logistic regression for train model

### Use pclass , sex , age , cabin and check inside

```
logis_model <- glm(Survived ~ Pclass + Sex + Age + Fare , data = titanic_for_train , family =  "binomial
summary(logis_model)
```

```
##
## Call:
## glm(formula = Survived ~ Pclass + Sex + Age + Fare, family = "binomial",
##     data = titanic_for_train)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.8231323  0.5983850   4.718 2.38e-06 ***
## Pclass      -1.3090713  0.1821361  -7.187 6.61e-13 ***
## Sex1         2.4326623  0.2366561  10.279  < 2e-16 ***
## Age         -0.0402958  0.0086518  -4.657 3.20e-06 ***
## Fare        -0.0009783  0.0025964  -0.377    0.706
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 730.55  on 534  degrees of freedom
## Residual deviance: 505.31  on 530  degrees of freedom
## AIC: 515.31
##
## Number of Fisher Scoring iterations: 4
```

## Change regression into probability and set threshold for split between dead or alive

```
predic_train <- predict(logis_model,type = "response")
titanic_for_train$predict <- if_else(predic_train < 0.5 , 0 , 1)
```

### Test model

```
predic_test  <- predict(logis_model , newdata = titanic_for_test, type = "response")
titanic_for_test$predict <- if_else(predic_test < 0.5 , 0 , 1)
df <- data.frame(train = mean(titanic_for_train$Survived == titanic_for_train$predict),
                 test = mean(titanic_for_test$Survived == titanic_for_test$predict))
df
```

```
##   train      test
## 1   0.8 0.8212291
```

We can observe that the logistic model has the ability to train on unseen data and does not suffer from overfitting.

### Use confusion matrix for explain the model

```
conM <- table(titanic_for_train$predict,titanic_for_train$Survived,
              dnn = c("Predicted","Actual"))
conM
```

```
##          Actual
## Predicted   0   1
##         0 253  54
##         1  53 175
```

```
## Acc
Acc <- (conM[1,1] + conM[2,2]) / sum(conM)
## Precision
Precision <- conM[2,2]/ sum(conM[2,])
## Recall
Recall <- conM[2,2] / sum(conM[,2])
## f-1
f1 <- (2*(Precision*Recall)/(Precision+Recall))
cat("\nAccuracy   :", Acc,
    "\nPrecision  :", Precision,
    "\nRecall     :", Recall,
    "\nf1         :", f1)
```

```
##
## Accuracy   : 0.8
## Precision  : 0.7675439
## Recall     : 0.7641921
## f1         : 0.7658643
```

## Summary

This trained model can predict survival on the Titanic using variables like Pclass, Sex, Age, and Fare with an accuracy ranging from 76% to 80% as depicted in the confusion matrix.