Titanic: logistic regression with R

Jintawee.s

Review Data

```
library(titanic)
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
```

Drop NA (missing values)

```
titanic_train <- na.omit(titanic_train)
nrow(titanic_train)
```

```
## [1] 714
```

Convert sex to factor

```
titanic_train$Sex = as.factor(titanic_train$Sex)
str(titanic_train)
```

```
## 'data.frame':    714 obs. of  12 variables:
##  $ PassengerId: int  1 2 3 4 5 7 8 9 10 11 ...
##  $ Survived   : int  0 1 1 1 0 0 0 1 1 1 ...
```

```
##  $ Pclass     : int  3 1 3 1 3 1 3 3 2 3 ...
##  $ Name       : chr  "Braund, Mr. Owen Harris" "Cumings, Mrs. John Bradley (Florence Briggs Thayer)"
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 1 1 1 ...
##  $ Age        : num  22 38 26 35 35 54 2 27 14 4 ...
##  $ SibSp      : int  1 1 0 1 0 0 3 0 1 1 ...
##  $ Parch      : int  0 0 0 0 0 0 1 2 0 1 ...
##  $ Ticket     : chr  "A/5 21171" "PC 17599" "STON/O2. 3101282" "113803" ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : chr  "" "C85" "" "C123" ...
##  $ Embarked   : chr  "S" "C" "S" "S" ...
##  - attr(*, "na.action")= 'omit' Named int [1:177] 6 18 20 27 29 30 32 33 37 43 ...
##   ..- attr(*, "names")= chr [1:177] "6" "18" "20" "27" ...
```

Split Data

```
set.seed(19)
n <- nrow(titanic_train)
id <- sample(1:n, size=n*0.7) ## 70% train 30% test
train_data <- titanic_train[id, ]
test_data <- titanic_train[-id, ]
```

Train Model

```
model_train <- glm(Survived ~ Pclass + Age + Sex, data = train_data, family="binomial")
summary(model_train)
```

```
##
## Call:
## glm(formula = Survived ~ Pclass + Age + Sex, family = "binomial",
##     data = train_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7120  -0.6777  -0.4067   0.6206   2.4457
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.944545   0.593776   8.327  < 2e-16 ***
## Pclass      -1.217577   0.164808  -7.388 1.49e-13 ***
## Age         -0.037541   0.009051  -4.148 3.36e-05 ***
## Sexmale     -2.541612   0.248926 -10.210  < 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: 673.56  on 498  degrees of freedom
## Residual deviance: 450.39  on 495  degrees of freedom
## AIC: 458.39
##
## Number of Fisher Scoring iterations: 5
```

Predict and Evaluate Model

```
train_data$prob_survived <- predict(model_train, type="response")
train_data$pred_survived <- ifelse(train_data$prob_survived >= 0.5, 1, 0)
```

Confusion matrix

```
conM_train <- table(train_data$pred_survived, train_data$Survived,
              dnn=c("Predicted", "Actual"))
```

Model_train Evaluation

```
Acc_train <- (conM_train[1 ,1] + conM_train[2 ,2]) / sum(conM_train)
Pre_train <- conM_train[2 ,2]/ (conM_train[2 ,1] + conM_train[2 ,2])
Re_train <- conM_train[2 ,2]/ (conM_train[1 ,2] + conM_train[2 ,2])

F1_train <- 2*((Pre_train*Re_train) / (Pre_train + Re_train))

cat("Accuracy:", Acc_train, "\nPrecision:", Pre_train, "\nRecall:", Re_train, "\nF1:", F1_train)
```

```
## Accuracy: 0.7955912
## Precision: 0.7659574
## Recall: 0.7128713
## F1: 0.7384615
```

Test Model

```
model_test <- glm(Survived ~ Pclass + Age + Sex, data = test_data, family="binomial")
summary(model_test)
```

```
##
## Call:
## glm(formula = Survived ~ Pclass + Age + Sex, family = "binomial",
##     data = test_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5818  -0.7057  -0.3728   0.6513   2.5023
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  5.35058    0.94342   5.671 1.42e-08 ***
## Pclass      -1.45558    0.26246  -5.546 2.92e-08 ***
## Age         -0.03497    0.01425  -2.455   0.0141 *
## Sexmale     -2.53124    0.38358  -6.599 4.14e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 290.94  on 214  degrees of freedom
## Residual deviance: 195.83  on 211  degrees of freedom
## AIC: 203.83
##
## Number of Fisher Scoring iterations: 5
```

Predict and Evaluate Model

```
test_data$prob_survived <- predict(model_test, type="response")
test_data$pred_survived <- ifelse(test_data$prob_survived >= 0.5, 1, 0)
```

Confusion matrix

```
conM_test <- table(test_data$pred_survived, test_data$Survived,
                   dnn=c("Predicted", "Actual"))
```

Model_train Evaluation

```
Acc_test <- (conM_test[1 ,1] + conM_test[2 ,2]) / sum(conM_test)
Pre_test <- conM_test[2 ,2]/ (conM_test[2 ,1] + conM_test[2 ,2])
Re_test <- conM_test[2 ,2]/ (conM_test[1 ,2] + conM_test[2 ,2])

F1_test <- 2*((Pre_test*Re_test) / (Pre_test + Re_test))

cat("Accuracy:", Acc_test, "\nPrecision:", Pre_test, "\nRecall:", Re_test, "\nF1:", F1_test)
```

```
## Accuracy: 0.7953488
## Precision: 0.7444444
## Recall: 0.7613636
## F1: 0.752809
```