Titanic Survive prediction

Pitchaporn Paipitak

Library

```
library(titanic)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

Drop NA

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

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

```
titanic_train$Survived <- as.factor(titanic_train$Survived)
```

Split Data

```
set.seed(42)
n <- nrow(titanic_train)
id <- sample(1:n, size = n*0.7)
train_data <- titanic_train[id, ]
test_data <- titanic_train[-id, ]
nrow(train_data)
```

```
## [1] 499
```

```
nrow(test_data)
```

```
## [1] 215
```

Train Model

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

```
##
```

```
## Call:
## glm(formula = Survived ~ Pclass + Sex + Age, family = "binomial",
##     data = train_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8617  -0.6485  -0.3554   0.6129   2.3884
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  5.604600   0.637259   8.795  < 2e-16 ***
## Pclass      -1.443887   0.174955  -8.253  < 2e-16 ***
## Sexmale     -2.739281   0.262607 -10.431  < 2e-16 ***
## Age         -0.041450   0.009522  -4.353 1.34e-05 ***
## ---
## 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: 432.26  on 495  degrees of freedom
## AIC: 440.26
##
## 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 of Train Model

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

```
##           Actual
## Predicted   0   1
##         0 253  44
##         1  57 145
```

Model evaluation

```
trainAccuracy <- (conM_train[1,1] + conM_train[2,2])/sum(conM_train)
trainPrecision <- conM_train[2,2]/(conM_train[2,2]+conM_train[2,1])
trainRecall <- conM_train[2,2]/(conM_train[1,2]+conM_train[2,2])
trainF1 <- 2*(( trainPrecision  * trainRecall  )/( trainPrecision  + trainRecall  ))

cat("Train Model","\nAccuracy: ",  trainAccuracy  , "\nPrecision: ", trainPrecision  , "\nRecall: ",  t
```

```
## Train Model
## Accuracy:  0.7975952
## Precision:  0.7178218
## Recall:  0.7671958
```

```
## F1 score:  0.741688
```

Test Model

```
test_data$prob <- predict(model_train, newdata = test_data, type = "response")
test_data$pred <- ifelse(test_data$prob >= 0.5,1,0)
```

Confusion matrix of Test Model

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

```
##           Actual
## Predicted   0   1
##         0 106  21
##         1  27  61
```

```
testAccuracy   <- (conM_test[1,1] + conM_test[2,2])/sum(conM_test)
testPrecision   <- conM_test[2,2]/(conM_test[2,2]+conM_test[2,1])
testRecall  <- conM_test[2,2]/(conM_test[1,2]+conM_test[2,2])
testF1   <- 2*(( testPrecision  * testRecall  )/( testPrecision   + testRecall  ))

cat("Test Model","\nAccuracy: ",  testAccuracy  , "\nPrecision:", testPrecision , "\nRecall: ",  testRe
```

```
## Test Model
## Accuracy:  0.7767442
## Precision: 0.6931818
## Recall:  0.7439024
## F1 score:  0.7176471
```