Homework 3

Shuhei Kaneko

2022/10/28

```
library(tidyverse)
library(tidymodels)
library(ggplot2)
library(corrplot)
library(MASS)
library(discrim)
library(klaR)
library(yardstick)
```

```
titanic <- read_csv("titanic.csv")

titanic <- titanic %>%
  mutate(survived = as.factor(survived)) %>%
  mutate(pclass = as.factor(pclass))
```

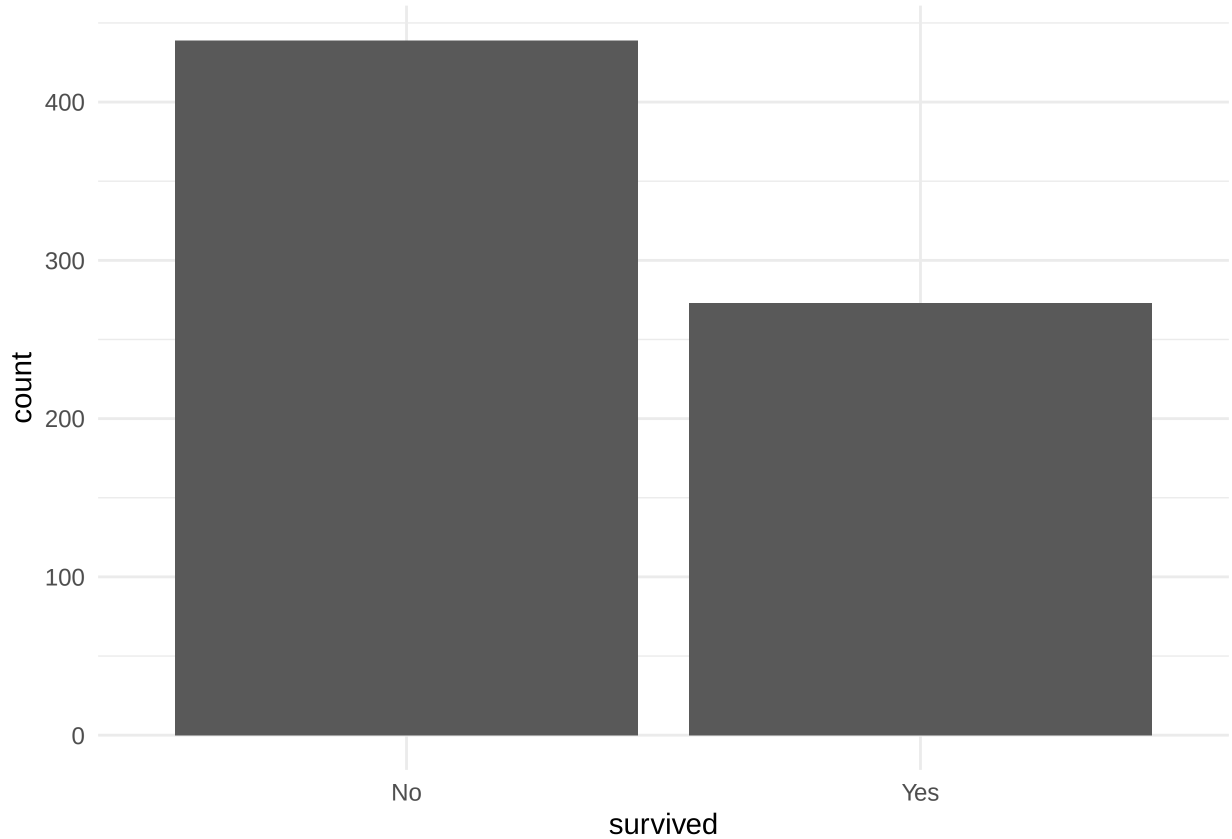
Question 1

```
set.seed(10)

titanic_split <- initial_split(titanic, prop = 0.80,
                               strata = survived)
titanic_train <- training(titanic_split)
titanic_test <- testing(titanic_split)
```

The stratification based on outcome will equate the fraction of y between training and test sample. Without using it, the fraction of y in test data could be very different from that in training data, which give us poor performance of our model.

Question 2

```
titanic_train2 <- titanic_train %>%
  mutate(survived_dummy = ifelse(survived == "Yes", 1, 0))

hist_survived <- titanic_train %>%
  ggplot(aes(survived)) +
  geom_bar() +
  theme_minimal()
plot(hist_survived)
```



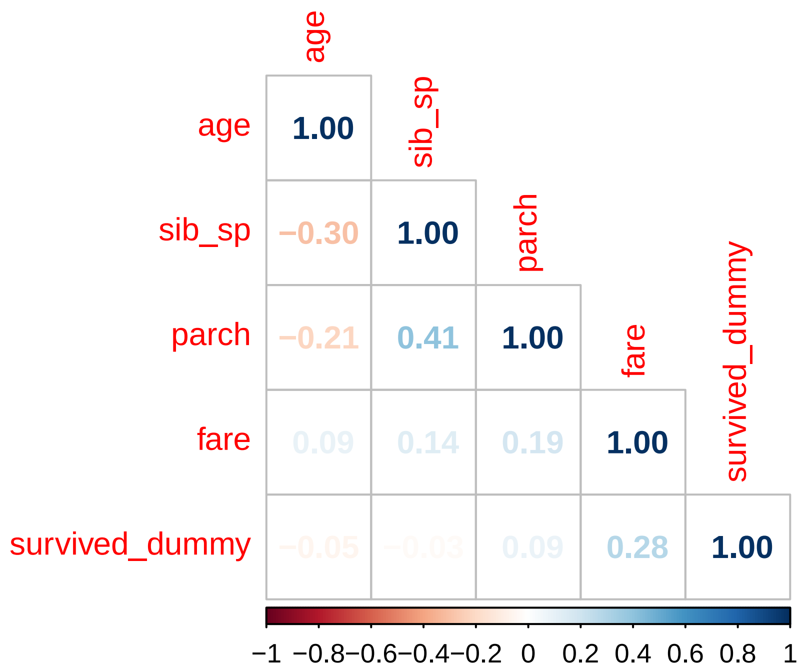

```
summary(tibble(titanic_train2$survived_dummy))
```

```
##  titanic_train2$survived_dummy
##  Min.   :0.0000
##  1st Qu.:0.0000
##  Median :0.0000
##  Mean   :0.3834
##  3rd Qu.:1.0000
##  Max.   :1.0000
```

Around 38% of passengers survived.

Question 3

```
titanic_train2 %>%
  dplyr::select(age, sib_sp, parch, fare, survived_dummy) %>%
  cor(use = "complete.obs") %>%
  corrplot(method = "number", type = "lower")
```



More expensive fare have positive correlation with survival probability. It is noteworthy that passenger age have negative correlation with # of sibling and spouse, or # of parents and children. # of sibling and spouse and # of parents and children has a positive correlation.

Question 4

```
titanic_recipe <- recipe(survived ~ pclass + sex + age + sib_sp + parch + fare, data = titanic_train) %>%
  step_impute_linear(age, impute_with = imp_vars(pclass, sex, sib_sp, parch, fare)) %>%
  step_dummy(all_nominal_predictors()) %>%
  step_interact(terms = ~ starts_with("sex"):fare) %>%
  step_interact(terms = ~ age:fare)
```

Question 5

```
# Engine
log_reg <- logistic_reg() %>%
  set_engine("glm") %>%
  set_mode("classification")

# Workflow
log_wkflow <- workflow() %>%
  add_model(log_reg) %>%
  add_recipe(titanic_recipe)

log_fit <- fit(log_wkflow, titanic_train)
```

Question 6

```
# Engine
lda <- discrim_linear() %>%
  set_engine("MASS") %>%
  set_mode("classification")

# Workflow
lda_wkflow <- workflow() %>%
  add_model(lda) %>%
  add_recipe(titanic_recipe)

lda_fit <- fit(lda_wkflow, titanic_train)
```

Question 7

```
# Engine
qda <- discrim_quad() %>%
  set_engine("MASS") %>%
  set_mode("classification")

# Workflow
qda_wkflow <- workflow() %>%
  add_model(qda) %>%
  add_recipe(titanic_recipe)

qda_fit <- fit(qda_wkflow, titanic_train)
```

Question 8

```
nb <- naive_Bayes() %>%
  set_mode("classification") %>%
  set_engine("klaR") %>%
  set_args(usekernel = FALSE)

nb_wkflow <- workflow() %>%
  add_model(nb) %>%
  add_recipe(titanic_recipe)

nb_fit <- fit(nb_wkflow, titanic_train)
```

Question 9

```
pred_log <- predict(log_fit, new_data = titanic_train, type = "prob")
pred_lda <- predict(lda_fit, new_data = titanic_train, type = "prob")
pred_qda <- predict(qda_fit, new_data = titanic_train, type = "prob")
pred_nb <- predict(nb_fit, new_data = titanic_train, type = "prob")
```

```
predictions <- bind_cols(pred_log[,2], pred_lda[,2], pred_qda[,2], pred_nb[,2])
```

```
## New names:
## * `.pred_Yes` -> `.pred_Yes...1`
## * `.pred_Yes` -> `.pred_Yes...2`
```

```
## * `.pred_Yes` -> `.pred_Yes...3`
## * `.pred_Yes` -> `.pred_Yes...4`
```

```
colnames(predictions) <- c("Logistic", "LDA", "QDA", "NB")

predictions
```

```
## # A tibble: 712 x 4
##    Logistic    LDA         QDA         NB
##       <dbl>  <dbl>       <dbl>      <dbl>
##  1   0.0664 0.0390 0.00280     0.00839
##  2   0.0945 0.0547 0.00381     0.00858
##  3   0.334  0.263  0.0664      0.510
##  4   0.0930 0.0592 0.0000513   0.000116
##  5   0.153  0.0877 0.00632     0.00952
##  6   0.787  0.844  0.501       0.340
##  7   0.0533 0.0375 0.000000193 0.00000121
##  8   0.466  0.582  0.186       0.264
##  9   0.531  0.643  0.000360    0.00501
## 10   0.0943 0.0544 0.00376     0.00881
## # ... with 702 more rows
```

- Logistic

```
augment(log_fit, new_data = titanic_train) %>%
  conf_mat(truth = survived, estimate = .pred_class)
```

```
##           Truth
## Prediction  No Yes
##        No  391  78
##        Yes  48 195
```

- LDA

```
augment(lda_fit, new_data = titanic_train) %>%
  conf_mat(truth = survived, estimate = .pred_class)
```

```
##           Truth
## Prediction  No Yes
##        No  385  86
##        Yes  54 187
```

- QDA

```
augment(qda_fit, new_data = titanic_train) %>%
  conf_mat(truth = survived, estimate = .pred_class)
```

```
##           Truth
## Prediction  No Yes
##        No  412 126
##        Yes  27 147
```

- Naive Bayes

```
augment(nb_fit, new_data = titanic_train) %>%
  conf_mat(truth = survived, estimate = .pred_class)
```

```
##           Truth
## Prediction  No Yes
##        No  407 128
```

```
##        Yes  32 145
```

Accuracy meausre of each method

- Logistic

```
log_acc <- augment(log_fit, new_data = titanic_train) %>%
  accuracy(truth = survived, estimate = .pred_class)
log_acc
```

```
## # A tibble: 1 x 3
##   .metric  .estimator .estimate
##   <chr>    <chr>          <dbl>
## 1 accuracy binary         0.823
```

- LDA

```
lda_acc <- augment(lda_fit, new_data = titanic_train) %>%
  accuracy(truth = survived, estimate = .pred_class)
lda_acc
```

```
## # A tibble: 1 x 3
##   .metric  .estimator .estimate
##   <chr>    <chr>          <dbl>
## 1 accuracy binary         0.803
```

- QDA

```
qda_acc <- augment(qda_fit, new_data = titanic_train) %>%
  accuracy(truth = survived, estimate = .pred_class)

qda_acc
```

```
## # A tibble: 1 x 3
##   .metric  .estimator .estimate
##   <chr>    <chr>          <dbl>
## 1 accuracy binary         0.785
```

- Naive Bayes

```
nb_acc <- augment(nb_fit, new_data = titanic_train) %>%
  accuracy(truth = survived, estimate = .pred_class)

nb_acc
```

```
## # A tibble: 1 x 3
##   .metric  .estimator .estimate
##   <chr>    <chr>          <dbl>
## 1 accuracy binary         0.775
```

```
accuracies <- c(log_acc$.estimate, lda_acc$.estimate,
                qda_acc$.estimate, nb_acc$.estimate)
models <- c("Logistic", "LDA", "QDA", "Naive Bayes")
results <- tibble(accuracies = accuracies, models = models)
results %>%
  arrange(-accuracies)
```

```
## # A tibble: 4 x 2
##   accuracies models
##        <dbl> <chr>
```

```
## 1       0.823 Logistic
## 2       0.803 LDA
## 3       0.785 QDA
## 4       0.775 Naive Bayes
```

The above results suggest that Logistic regression is the best model among four strategies.

Question 10

- Prediction for testing data

```r
log_acc_test <- augment(log_fit, new_data = titanic_test) %>%
  accuracy(truth = survived, estimate = .pred_class)
log_acc_test
```

```
## # A tibble: 1 x 3
##   .metric  .estimator .estimate
##   <chr>    <chr>          <dbl>
## 1 accuracy binary         0.771
```

Accuracy is about 77%

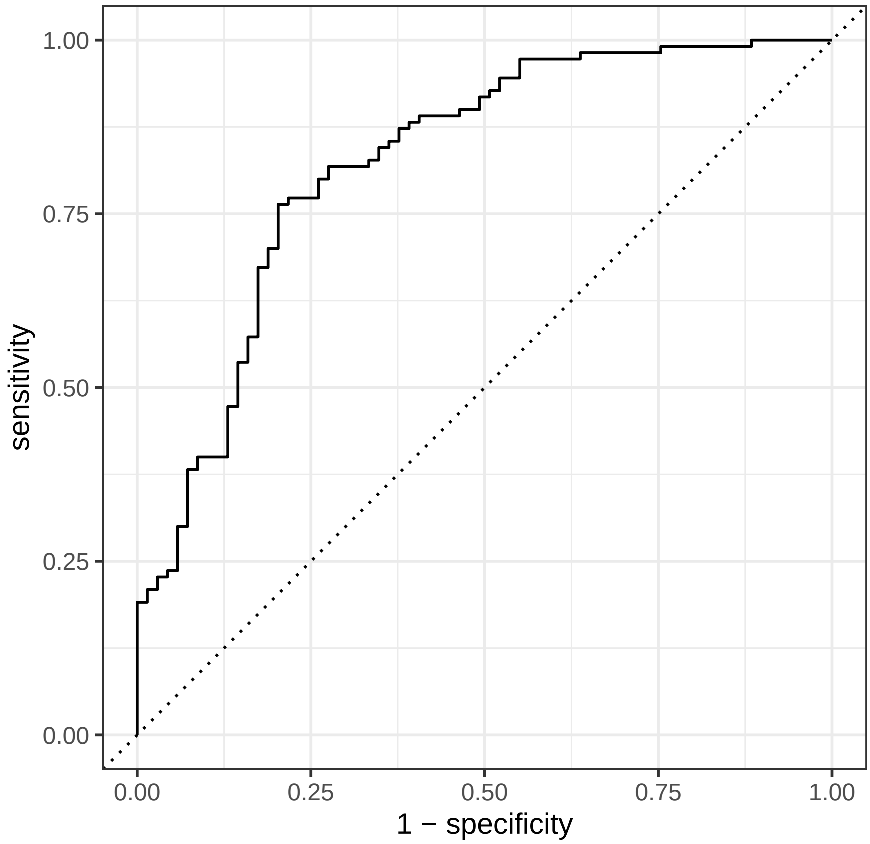
- Confusion matrix

```r
augment(log_fit, new_data = titanic_test) %>%
  conf_mat(truth = survived, estimate = .pred_class)
```

```
##           Truth
## Prediction No Yes
##        No  95  26
##        Yes 15  43
```

- ROC curve and AUC

```r
augment(log_fit, new_data = titanic_test) %>%
  roc_curve(survived, .pred_No) %>%
  autoplot()
```




```
augment(log_fit, new_data = titanic_test) %>%
  roc_auc(survived, .pred_No)
```

```
## # A tibble: 1 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 roc_auc binary         0.824
```

AUC is 0.8238.

Question 11

$$
\begin{aligned}
p &= \frac{e^z}{1+e^z} \\
&\Rightarrow p(1+e^z) = e^z \\
&\Rightarrow p = e^z(1-p) \\
&\Rightarrow e^z = \frac{p}{1-p} \\
&\Rightarrow z(p) = \ln\left(\frac{p}{1-p}\right)
\end{aligned}
$$

Question 12

Increase in x_1 by two will induce increase in $log(odds)$ by $2\beta_1$. Therefore, odds increase by $e^{2\beta_1}$.

When β_1 is negative, z approaches to $-\infty$ as $x_1 \to \infty$, that is, p approaches to 0. On the other hand, as $x_1 \to -\infty$, p approaches to 1.