PSTAT231 HW3 Cheng Ye

Cheng Ye

2022-10-30

```r
library(tidymodels)
```

```
## -- Attaching packages -------------------------------------- tidymodels 1.0.0 --
```

```
## v broom        1.0.1      v recipes      1.0.1
## v dials        1.0.0      v rsample      1.1.0
## v dplyr        1.0.10     v tibble       3.1.8
## v ggplot2      3.3.6      v tidyr        1.2.1
## v infer        1.0.3      v tune         1.0.1
## v modeldata    1.0.1      v workflows    1.1.0
## v parsnip      1.0.2      v workflowsets 1.0.0
## v purrr        0.3.4      v yardstick    1.1.0
```

```
## -- Conflicts ----------------------------------------- tidymodels_conflicts() --
## x purrr::discard() masks scales::discard()
## x dplyr::filter()  masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## x recipes::step()  masks stats::step()
## * Use suppressPackageStartupMessages() to eliminate package startup messages
```

```r
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
```

```
## v readr   2.1.2     v forcats 0.5.2
## v stringr 1.4.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x readr::col_factor() masks scales::col_factor()
## x purrr::discard()    masks scales::discard()
## x dplyr::filter()     masks stats::filter()
## x stringr::fixed()    masks recipes::fixed()
## x dplyr::lag()        masks stats::lag()
## x readr::spec()       masks yardstick::spec()
```

```r
library(ggplot2)
library(ggthemes)
library(corrr)
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
library(discrim)
```

```
##
## Attaching package: 'discrim'
##
## The following object is masked from 'package:dials':
##
##     smoothness
```

```
library(poissonreg)
library(klaR)
```

```
## Loading required package: MASS
##
## Attaching package: 'MASS'
##
## The following object is masked from 'package:dplyr':
##
##     select
```

```
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
##
## Attaching package: 'pROC'
##
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
tidymodels_prefer()
#Load required packages
```

```
titanic <-read.csv("C:/Cheng Ye/UCSB/PSTAT 231/HW/homework-3/data/titanic.csv") %>%
  mutate(survived = factor(survived,
                           levels = c("Yes", "No")),
         pclass = factor(pclass))
# Loading required dataset
head(titanic) # visualize data set
```

```
##   passenger_id survived pclass
## 1            1       No      3
## 2            2      Yes      1
## 3            3      Yes      3
## 4            4      Yes      1
## 5            5       No      3
## 6            6       No      3
##                                                  name    sex age sib_sp 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  <NA>        S
## 2         PC 17599 71.2833   C85        C
## 3 STON/O2. 3101282  7.9250  <NA>        S
## 4           113803 53.1000  C123        S
## 5           373450  8.0500  <NA>        S
## 6           330877  8.4583  <NA>        Q
```

```
set.seed(231) # could be any number
```

Question 1

```
titanic_split <- initial_split(titanic, prop = 0.80, strata = survived) # split the data and str
atified on survived,train 80%, test 20%
titanic_train <- training(titanic_split) # training the dataset
titanic_test <- testing(titanic_split) # testing the dataset
dim(titanic_train)
```

```
## [1] 712  12
```

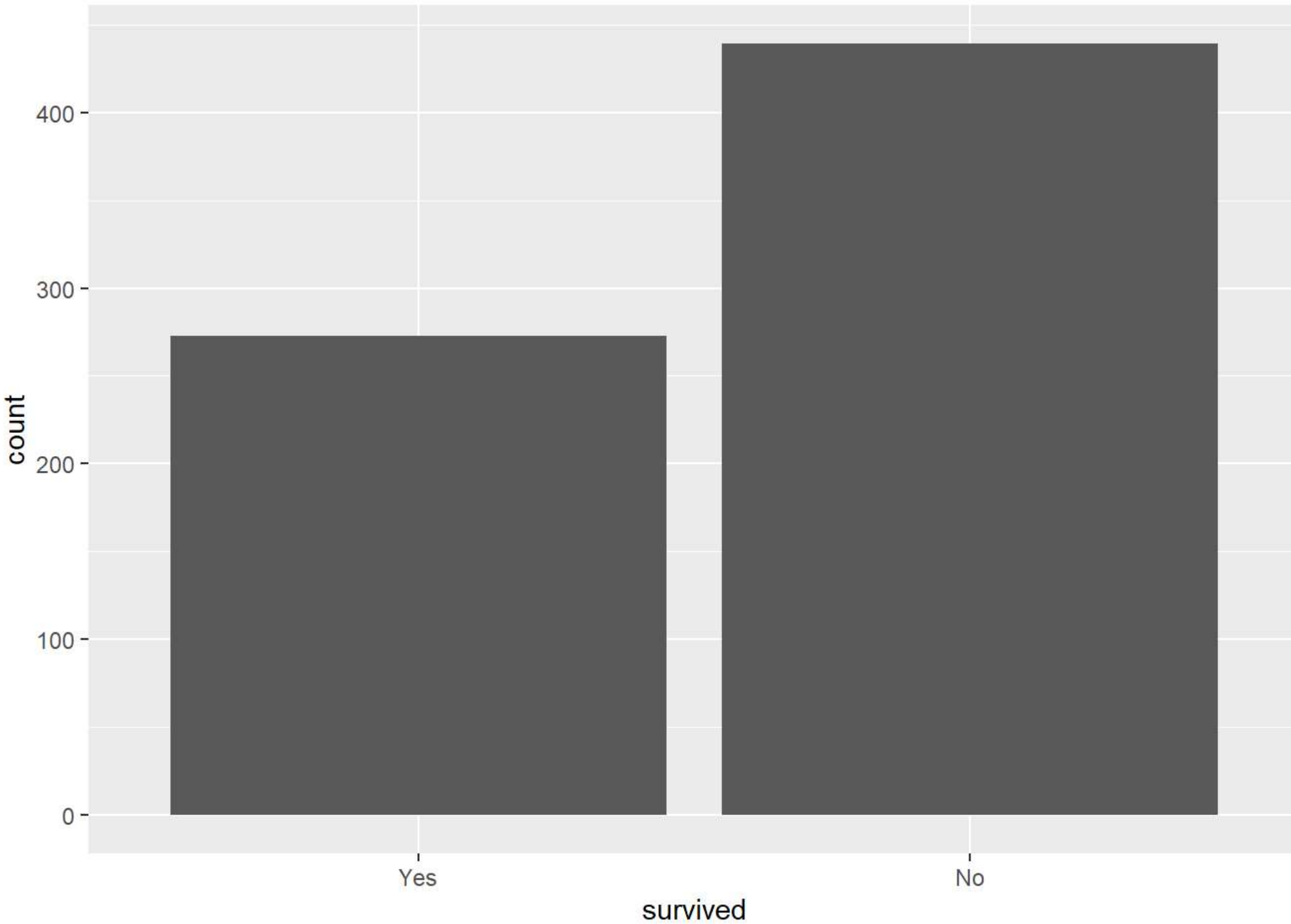
```
dim(titanic_test)
```

```
## [1] 179  12
```

```
# As the outcome variable is imbalanced for this dataset, using stratified sampling method for t
his data allows every subgroup in the population receives proper representation.
```

Question 2

```
ggplot(titanic_train, aes(survived))+geom_bar()
```

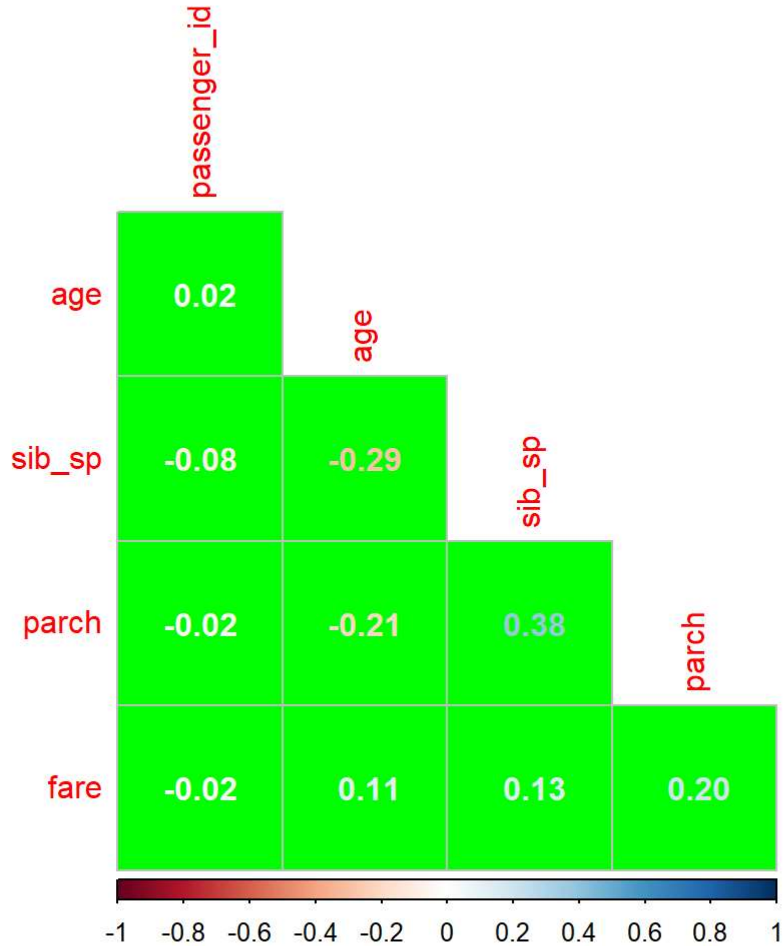



```
# Using barchart to visualize the training dataset, we could see that there is slight class imba
lance where one class, survived = yes, contains significantly fewer samples than the other clas
s, survived = no. From the graph, it looks like a binomial distribution
```

Question 3

```
titanic_train %>% select(is.numeric, -c(survived, name, sex, ticket, cabin, embarked)) %>% cor(u
se = "complete.obs") %>% corrplot(type = "lower", diag=FALSE, bg ="green", method = "number")
```

```
## Warning: Predicate functions must be wrapped in `where()`.
##
##   # Bad
##   data %>% select(is.numeric)
##
##   # Good
##   data %>% select(where(is.numeric))
##
## i Please update your code.
## This message is displayed once per session.
```




```
# From the graph, we could deduce that age and sib_sp are negatively correlated, parch and age a
re also negatively correlated; parch and sib_sp are postively correlated, fare and parch are pos
tively correlated
```

Question 4

```
titanic_survived_recipe <- recipe(survived ~ pclass + sex + age + sib_sp + parch + fare, titanic
_train) %>%
  step_impute_linear(age) %>%
  step_dummy(all_nominal_predictors()) %>%
  step_interact(terms = ~ starts_with('sex'):fare+age:fare)
summary(titanic_survived_recipe)
```

```
## # A tibble: 7 x 4
##   variable type    role      source
##   <chr>    <chr>   <chr>     <chr>
## 1 pclass   nominal predictor original
## 2 sex      nominal predictor original
## 3 age      numeric predictor original
## 4 sib_sp   numeric predictor original
## 5 parch    numeric predictor original
## 6 fare     numeric predictor original
## 7 survived nominal outcome   original
```

Question 5

```
log_reg <- logistic_reg() %>%
  set_engine("glm") %>%
  set_mode("classification")
log_wkflow <- workflow() %>%
  add_model(log_reg) %>%
  add_recipe(titanic_survived_recipe)
log_fit <- fit(log_wkflow, titanic_train)
```

Question 6

```
lda_mod <- discrim_linear() %>%
  set_engine('MASS') %>%
  set_mode('classification')
lda_wkflow <- workflow() %>%
  add_model(lda_mod) %>%
  add_recipe(titanic_survived_recipe)
lda_fit <- fit(lda_wkflow, titanic_train)
```

Question 7

```
qda_mod <- discrim_quad() %>%
  set_engine('MASS') %>%
  set_mode('classification')
qda_wkflow <- workflow() %>%
  add_model(qda_mod) %>%
  add_recipe(titanic_survived_recipe)
qda_fit <- fit(qda_wkflow, titanic_train)
```

Question 8

```
nb_mod <- naive_Bayes() %>%
  set_mode("classification") %>%
  set_engine("klaR") %>%
  set_args(usekernel = FALSE)
nb_wkflow <- workflow() %>%
  add_model(nb_mod) %>%
  add_recipe(titanic_survived_recipe)
nb_fit <- fit(nb_wkflow, titanic_train)
```

Question 9

```
log_acc <- predict(log_fit, new_data = titanic_train, type = "class") %>%
  bind_cols(titanic_train %>% select(survived)) %>%
  accuracy(truth = survived, estimate = .pred_class)
lda_acc <- predict(lda_fit, new_data = titanic_train, type = "class") %>%
  bind_cols(titanic_train %>% select(survived)) %>%
  accuracy(truth = survived, estimate = .pred_class)
qda_acc <- predict(qda_fit, new_data = titanic_train, type = "class") %>%
  bind_cols(titanic_train %>% select(survived)) %>%
  accuracy(truth = survived, estimate = .pred_class)
nb_acc <- predict(nb_fit, new_data = titanic_train, type = "class")%>%
  bind_cols(titanic_train %>% select(survived)) %>%
  accuracy(truth = survived, estimate = .pred_class)
```

```
## Warning in FUN(X[[i]], ...): Numerical 0 probability for all classes with
## observation 1
```

```
## Warning in FUN(X[[i]], ...): Numerical 0 probability for all classes with
## observation 2
```

```
## Warning in FUN(X[[i]], ...): Numerical 0 probability for all classes with
## observation 3
```

```
## Warning in FUN(X[[i]], ...): Numerical 0 probability for all classes with
## observation 4
```

```
## Warning in FUN(X[[i]], ...): Numerical 0 probability for all classes with
## observation 5
```

```
## Warning in FUN(X[[i]], ...): Numerical 0 probability for all classes with
## observation 6
```

```
## Warning in FUN(X[[i]], ...): Numerical 0 probability for all classes with
## observation 7
```

```
## Warning in FUN(X[[i]], ...): Numerical 0 probability for all classes with
## observation 708
```

```
## Warning in FUN(X[[i]], ...): Numerical 0 probability for all classes with
## observation 709
```

```
## Warning in FUN(X[[i]], ...): Numerical 0 probability for all classes with
## observation 710
```

```
## Warning in FUN(X[[i]], ...): Numerical 0 probability for all classes with
## observation 711
```

```
## Warning in FUN(X[[i]], ...): Numerical 0 probability for all classes with
## observation 712
```

```
results <- bind_rows(log_acc, lda_acc, qda_acc, nb_acc) %>%
  tibble() %>%mutate(model = c("Logistic Regression","Linear Discriminant Aanalysis", "Quadratic
Discriminant Analysis","Naive Bayes")) %>%
  select(model, .estimate)
results
```

```
## # A tibble: 4 x 2
##   model                              .estimate
##   <chr>                                  <dbl>
## 1 Logistic Regression                    0.810
## 2 Linear Discriminant Aanalysis          0.803
## 3 Quadratic Discriminant Analysis        0.813
## 4 Naive Bayes                            0.792
```

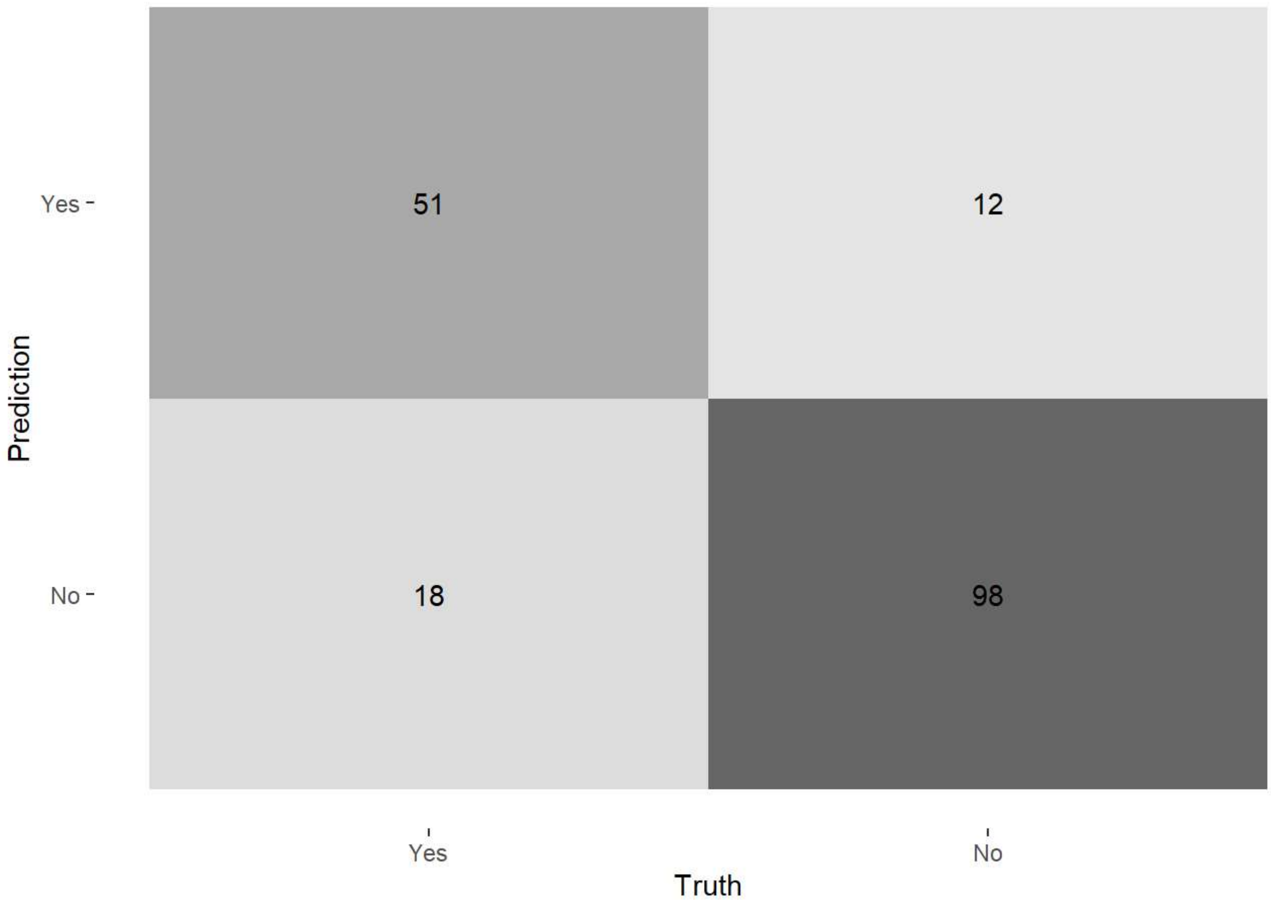
```
#From observing the results, the Logistic Regression method performed the best on the training d
ata
```

Question 10

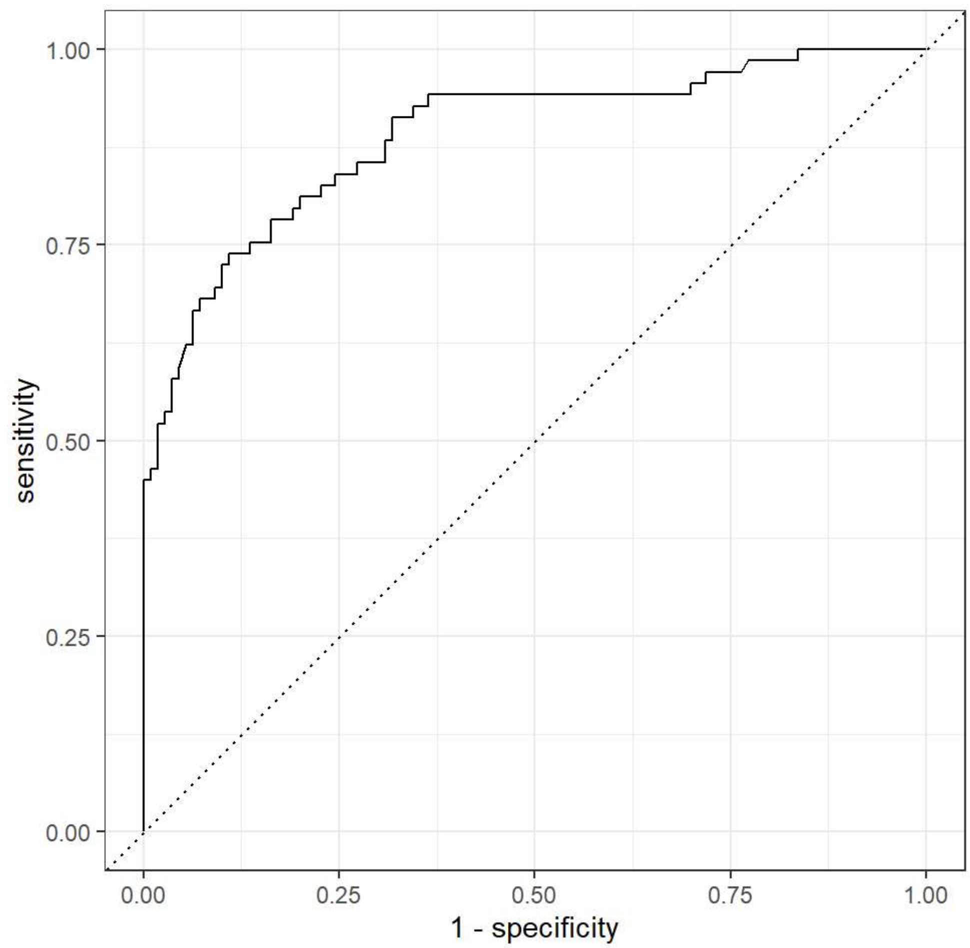
```
log_test <- fit(log_wkflow, titanic_test)
predict(log_test, new_data = titanic_test, type = "class") %>%
  bind_cols(titanic_test %>% select(survived)) %>%
  accuracy(truth = survived, estimate = .pred_class)
```

```
## # A tibble: 1 x 3
##   .metric  .estimator .estimate
##   <chr>    <chr>          <dbl>
## 1 accuracy binary         0.832
```

```
augment(log_test, new_data = titanic_test) %>%
  conf_mat(truth = survived, estimate = .pred_class) %>%
  autoplot(type = "heatmap")
```




```
augment(log_test, new_data = titanic_test) %>%
  roc_curve(survived, .pred_Yes) %>%
  autoplot()
```




```
#The model has accuracy 0.8268156, which is close to 1, hence the model performed pretty well, i
ts accuracy slightly increased on the testing data, this specifies that the model fitting is wel
l. The cause of such results might be the model being mechanistic thus having lower variance.
```

Question 11

```
# Denote P(z) = y
# Then y+y*e**(z)=e**z %>% y=e**z - y*e**z %>% y = (1-y)*e**z
# Then e**z =y/(1-y) %>% log(e**z)=log(y/(1-y))
# Then z = log(y/(1-y)) %>% z(p) = log(p/(1-p))
# Q.E.D.
```

Question 12

```
#If we increase X_1 by 2, the odds of outcome would increase by e**(2*beta_1) times.
#If beta_1  is negative, as X_1 approaches infinity, p/1-p approaches 0, as X_1 approaches negat
ive infinity , p/1-p approaches 1
```