Homework 3

PSTAT 131/231

Contents

```
library(tidymodels)
library(tidyverse)
library(corrplot)
library(discrim)
library(poissonreg)
library(corrr)
library(klaR)
titanic <- read.csv("titanic.csv")
titanic$survived <- factor(titanic$survived, levels = c("Yes", "No"))
titanic$pclass <- factor(titanic$pclass)
head(titanic)
```

```
##   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
```

Question1 Split the data, stratifying on the outcome variable, `survived`. You should choose the proportions to split the data into. Verify that the training and testing data sets have the appropriate number of observations. Take a look at the training data and note any potential issues, such as missing data.

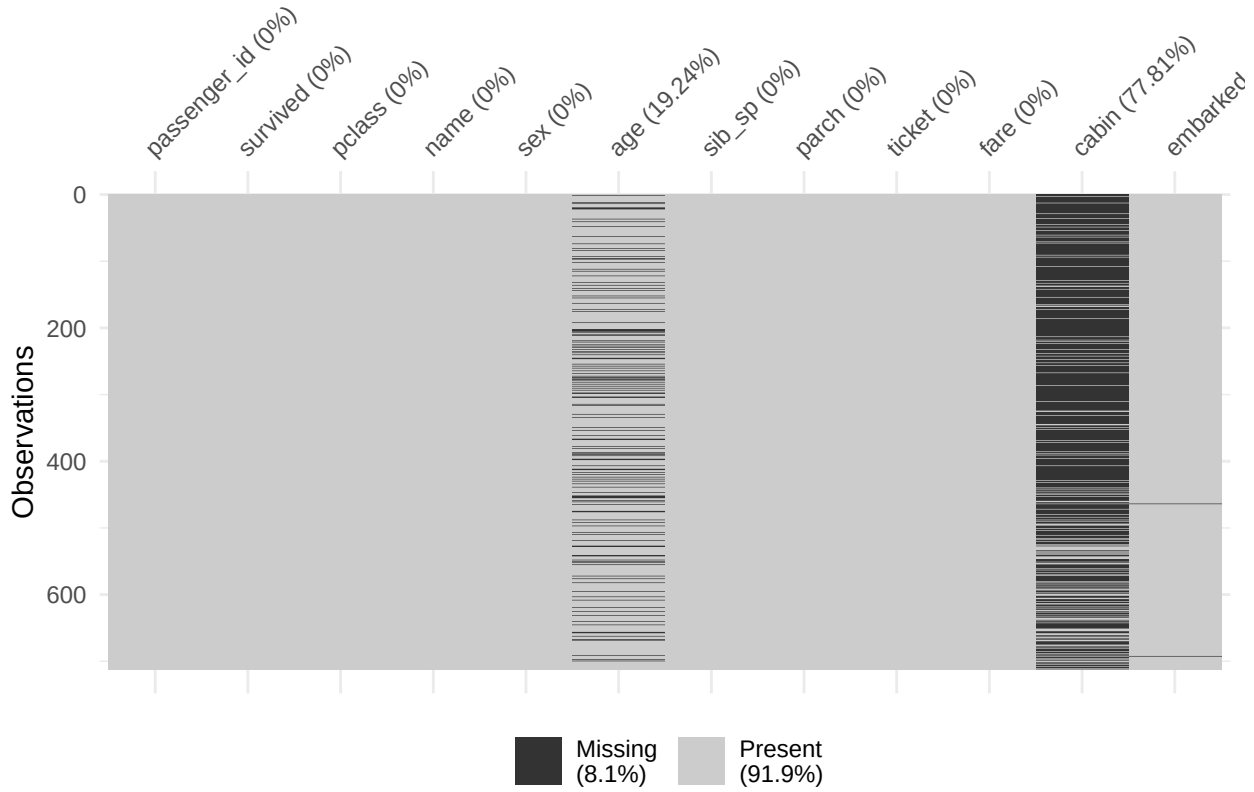
Why is it a good idea to use stratified sampling for this data?

```
set.seed(1979)
titanic_split <- initial_split(titanic, prop = 0.8,
                               strata = survived)
titanic_train <- training(titanic_split)
titanic_test <- testing(titanic_split)
c(nrow(titanic_train),nrow(titanic_test))
```

```
## [1] 712 179
```

The training data sets have 712 observation and testing data sets have 179 observation. They both have the appropriate number of observation

```
library(naniar)
vis_miss(titanic_train)
```

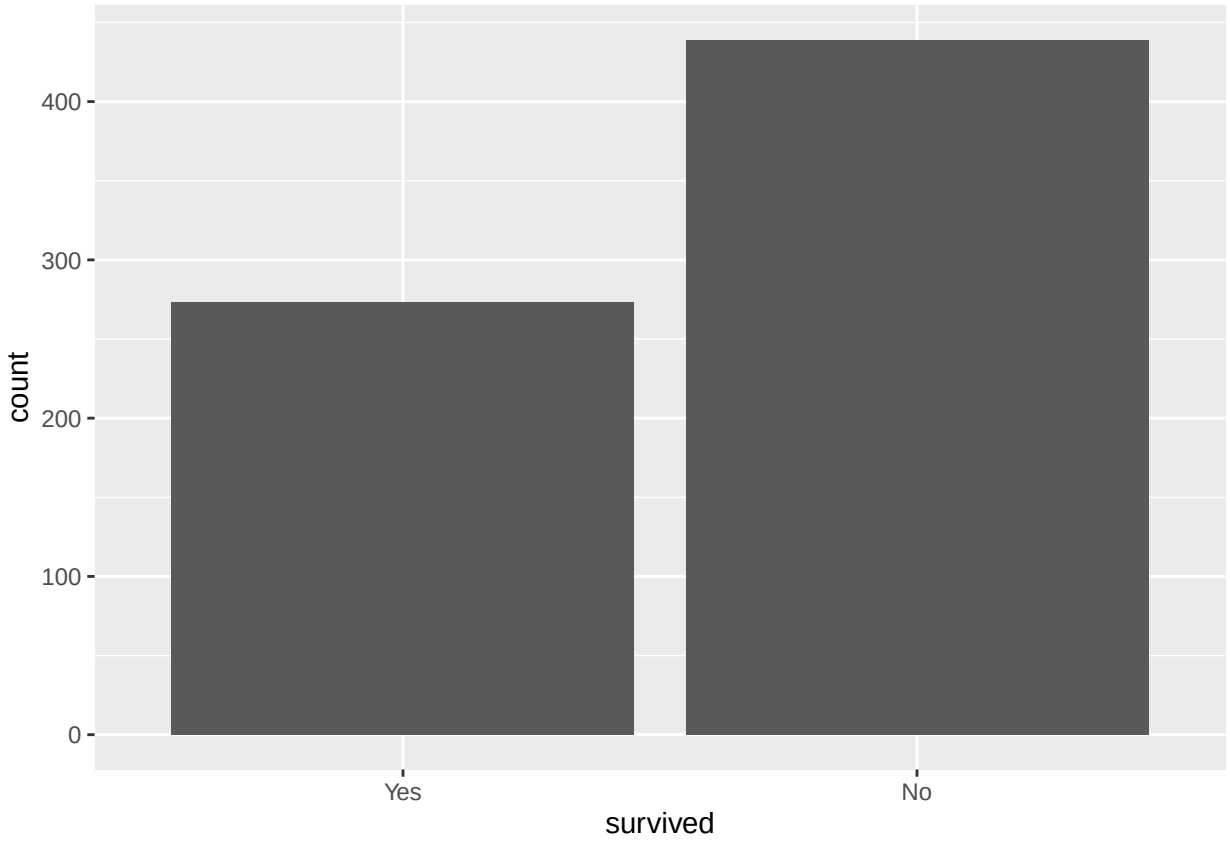


We have 77.81% missing value in cabin and 19.24% in age.

Why is it a good idea to use stratified sampling for this data? + Stratified sampling can make a representative amount of each level (No/Yes) of variable (survived) is included in the training and testing data set.

Question2 Using the **training** data set, explore/describe the distribution of the outcome variable `survived`.

```
titanic_train %>%
  ggplot(aes(x = survived)) +
  geom_bar()
```

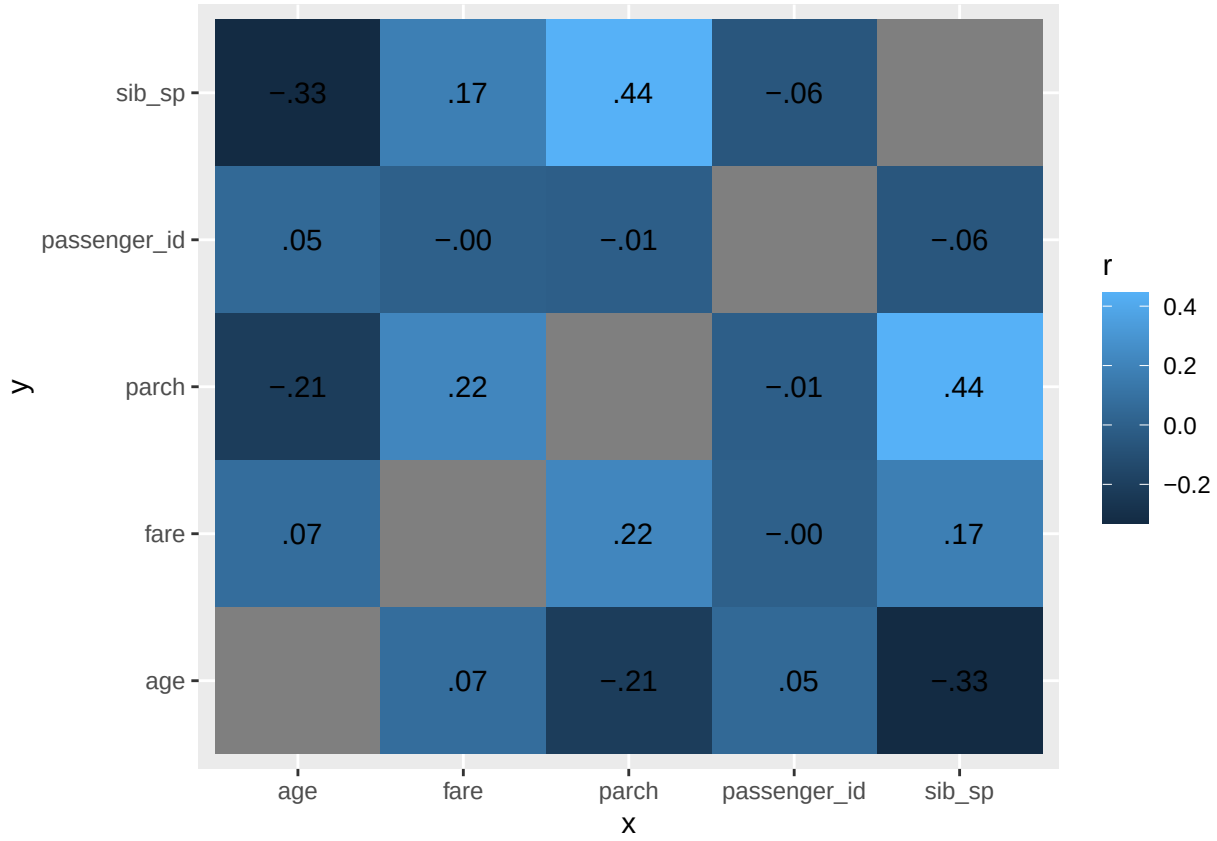


There are more died passenger than survived, with 40% dead and 60% survived.

Question3 Using the **training** data set, create a correlation matrix of all continuous variables. Create a visualization of the matrix, and describe any patterns you see. Are any predictors correlated with each other? Which ones, and in which direction?

```
library(corrr)
cor_titanic <- titanic_train %>%
  select_if(is.numeric) %>%
  correlate()

cor_titanic %>%
  stretch() %>%
  ggplot(aes(x, y, fill = r)) +
  geom_tile() +
  geom_text(aes(label = as.character(fashion(r))))
```



We can also use ggplot and the geom_tile() function to create a heatmap-style correlation plot Parch is positive correlated with sib_sp with correlation 0.44 Parch is positive correlated with fare with correlation 0.22 Sib_sp is negative correlated with age with correlation -0.33 Sib_sp is positive correlated with fare with correlation 0.17 Most of variables are not correlated with each other with correlation close to 0

Question4 Using the **training** data, create a recipe predicting the outcome variable `survived`. Include the following predictors: ticket class, sex, age, number of siblings or spouses aboard, number of parents or children aboard, and passenger fare.

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

Question5 Specify a **logistic regression** model for classification using the `"glm"` engine. Then create a workflow. Add your model and the appropriate recipe. Finally, use `fit()` to apply your workflow to the **training** data.

```
log_model <- logistic_reg() %>%
  set_engine("glm")

log_wflow <- workflow() %>%
  add_model(log_model) %>%
  add_recipe(simple_titanic_recipe)
```

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

Question6 **Repeat Question 5**, but this time specify a linear discriminant analysis model for classification using the `"MASS"` engine.

```
lda_mod <- discrim_linear() %>%
  set_mode("classification") %>%
  set_engine("MASS")

lda_wkflow <- workflow() %>%
  add_model(lda_mod) %>%
  add_recipe(simple_titanic_recipe)

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

Question7 **Repeat Question 5**, but this time specify a quadratic discriminant analysis model for classification using the `"MASS"` engine.

```
qda_mod <- discrim_quad() %>%
  set_mode("classification") %>%
  set_engine("MASS")

qda_wkflow <- workflow() %>%
  add_model(qda_mod) %>%
  add_recipe(simple_titanic_recipe)

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

Question8 **Repeat Question 5**, but this time specify a naive Bayes model for classification using the `"klaR"` engine. Set the `usekernel` argument to `FALSE`.

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

nb_wkflow <- workflow() %>%
  add_model(nb_mod) %>%
  add_recipe(simple_titanic_recipe)

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

Question9 Now you've fit four different models to your training data.

Use `predict()` and `bind_cols()` to generate predictions using each of these 4 models and your **training** data. Then use the *accuracy* metric to assess the performance of each of the four models.

Which model achieved the highest accuracy on the training data?

```
predict(log_fit, new_data = titanic_train, type = "prob") %>%
  bind_cols(titanic_train)
```

```
## # A tibble: 712 x 14
##    .pred_Yes .pred_No passenger_id survived pclass name       sex     age sib_sp
##        <dbl>    <dbl>        <int> <fct>    <fct>  <chr>      <chr> <dbl>  <int>
##  1    0.112     0.888            1 No       3      Braund, M~ male     22      1
##  2    0.0756    0.924            5 No       3      Allen, Mr~ male     35      0
##  3    0.111     0.889            6 No       3      Moran, Mr~ male     NA      0
##  4    0.324     0.676            7 No       1      McCarthy,~ male     54      0
##  5    0.134     0.866            8 No       3      Palsson, ~ male      2      3
##  6    0.182     0.818           13 No       3      Saunderco~ male     20      0
##  7    0.0495    0.950           14 No       3      Andersson~ male     39      1
##  8    0.794     0.206           15 No       3      Vestrom, ~ fema~    14      0
##  9    0.0774    0.923           17 No       3      Rice, Mas~ male      2      4
## 10    0.463     0.537           19 No       3      Vander Pl~ fema~    31      1
## # ... with 702 more rows, and 5 more variables: parch <int>, ticket <chr>,
## #   fare <dbl>, cabin <chr>, embarked <chr>
```

```
predict(qda_fit, new_data = titanic_train, type = "prob") %>%
  bind_cols(titanic_train)
```

```
## # A tibble: 712 x 14
##      .pred_Yes .pred_No passenger_id survived pclass name     sex     age sib_sp
##          <dbl>    <dbl>        <int> <fct>    <fct>  <chr>    <chr> <dbl>  <int>
##  1 0.00549        0.995            1 No       3      Braund,~ male     22      1
##  2 0.00373        0.996            5 No       3      Allen, ~ male     35      0
##  3 0.00525        0.995            6 No       3      Moran, ~ male     NA      0
##  4 0.135          0.865            7 No       1      McCarth~ male     54      0
##  5 0.000124       1.00             8 No       3      Palsson~ male      2      3
##  6 0.00875        0.991           13 No       3      Saunder~ male     20      0
##  7 0.550          0.450           14 No       3      Anderss~ male     39      1
##  8 0.503          0.497           15 No       3      Vestrom~ fema~    14      0
##  9 0.000000685    1.00            17 No       3      Rice, M~ male      2      4
## 10 0.232          0.768           19 No       3      Vander ~ fema~    31      1
## # ... with 702 more rows, and 5 more variables: parch <int>, ticket <chr>,
## #   fare <dbl>, cabin <chr>, embarked <chr>
```

```
predict(lda_fit, new_data = titanic_train, type = "prob") %>%
  bind_cols(titanic_train)
```

```
## # A tibble: 712 x 14
##    .pred_Yes .pred_No passenger_id survived pclass name       sex     age sib_sp
##        <dbl>    <dbl>        <int> <fct>    <fct>  <chr>      <chr> <dbl>  <int>
##  1    0.0694    0.931            1 No       3      Braund, M~ male     22      1
##  2    0.0481    0.952            5 No       3      Allen, Mr~ male     35      0
##  3    0.0688    0.931            6 No       3      Moran, Mr~ male     NA      0
##  4    0.260     0.740            7 No       1      McCarthy,~ male     54      0
##  5    0.0878    0.912            8 No       3      Palsson, ~ male      2      3
##  6    0.110     0.890           13 No       3      Saunderco~ male     20      0
##  7    0.0322    0.968           14 No       3      Andersson~ male     39      1
##  8    0.841     0.159           15 No       3      Vestrom, ~ fema~    14      0
```

```
##  9    0.0551    0.945           17 No       3      Rice, Mas~ male      2      4
## 10    0.567     0.433           19 No       3      Vander Pl~ fema~    31      1
## # ... with 702 more rows, and 5 more variables: parch <int>, ticket <chr>,
## #   fare <dbl>, cabin <chr>, embarked <chr>
```

```
# naive Bayes model
predict(nb_fit, new_data = titanic_train, type = "prob") %>%
  bind_cols(titanic_train)
```

```
## # A tibble: 712 x 14
##     .pred_Yes .pred_No passenger_id survived pclass name      sex     age sib_sp
##         <dbl>    <dbl>        <int> <fct>    <fct>  <chr>     <chr> <dbl>  <int>
##  1 0.0155        0.984            1 No       3      Braund, ~ male     22      1
##  2 0.0152        0.985            5 No       3      Allen, M~ male     35      0
##  3 0.0158        0.984            6 No       3      Moran, M~ male     NA      0
##  4 0.573         0.427            7 No       1      McCarthy~ male     54      0
##  5 0.000278      1.00             8 No       3      Palsson,~ male      2      3
##  6 0.0176        0.982           13 No       3      Saunderc~ male     20      0
##  7 0.0960        0.904           14 No       3      Andersso~ male     39      1
##  8 0.475         0.525           15 No       3      Vestrom,~ fema~    14      0
##  9 0.00000328    1.00            17 No       3      Rice, Ma~ male      2      4
## 10 0.383         0.617           19 No       3      Vander P~ fema~    31      1
## # ... with 702 more rows, and 5 more variables: parch <int>, ticket <chr>,
## #   fare <dbl>, cabin <chr>, embarked <chr>
```

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

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

```
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         0.760
```

```
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         0.760
```

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

```
## # A tibble: 4 x 2
##   accuracies models
##        <dbl> <chr>
## 1      0.808 Logistic Regression
## 2      0.799 LDA
## 3      0.760 Naive Bayes
## 4      0.760 QDA
```

Logistic Regression achieved highest accuracy on the training data

Question 10 Fit the model with the highest training accuracy to the **testing** data. Report the accuracy of the model on the **testing** data.

Again using the **testing** data, create a confusion matrix and visualize it. Plot an ROC curve and calculate the area under it (AUC).

How did the model perform? Compare its training and testing accuracies. If the values differ, why do you think this is so?

```
predict(log_fit, new_data = titanic_test, type = "prob")
```

```
## # A tibble: 179 x 2
##    .pred_Yes .pred_No
##        <dbl>    <dbl>
##  1    0.819     0.181
##  2    0.219     0.781
##  3    0.415     0.585
##  4    0.0322    0.968
##  5    0.324     0.676
##  6    0.111     0.889
##  7    0.757     0.243
##  8    0.292     0.708
##  9    0.382     0.618
## 10    0.814     0.186
## # ... with 169 more rows
```

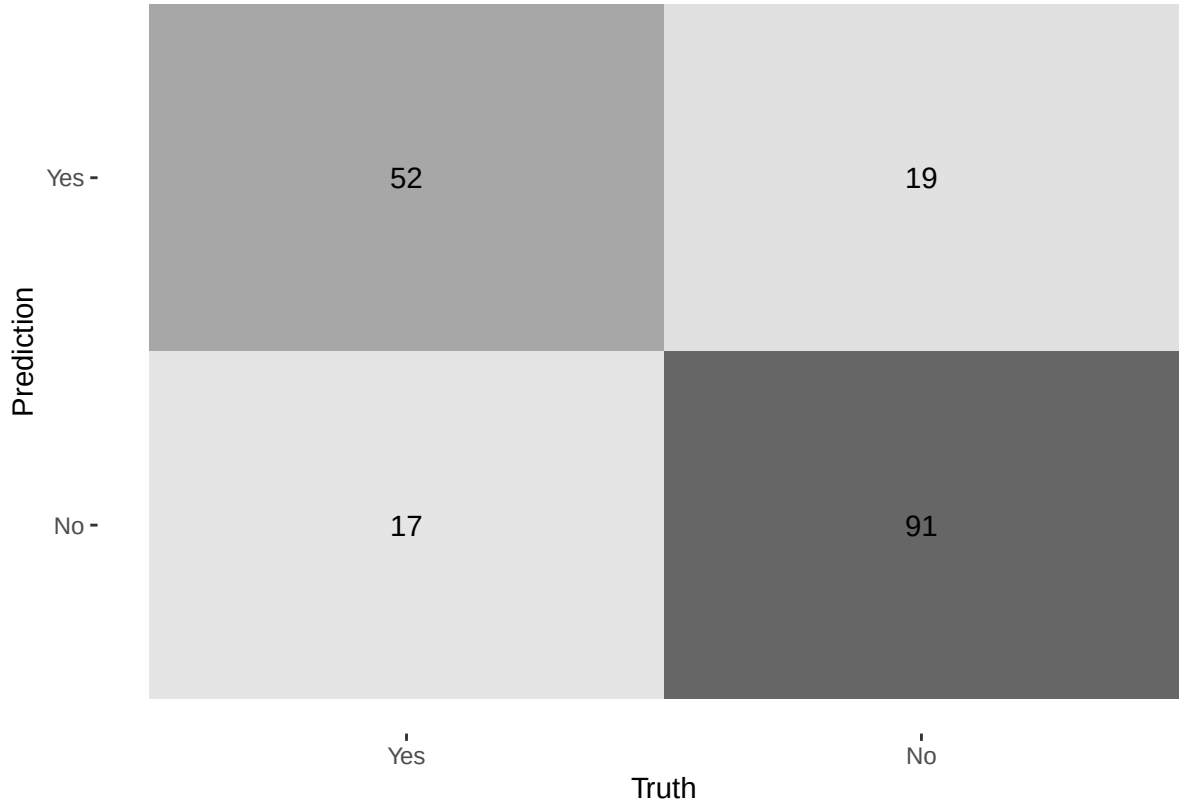
```
multi_metric <- metric_set(accuracy, sensitivity, specificity)

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

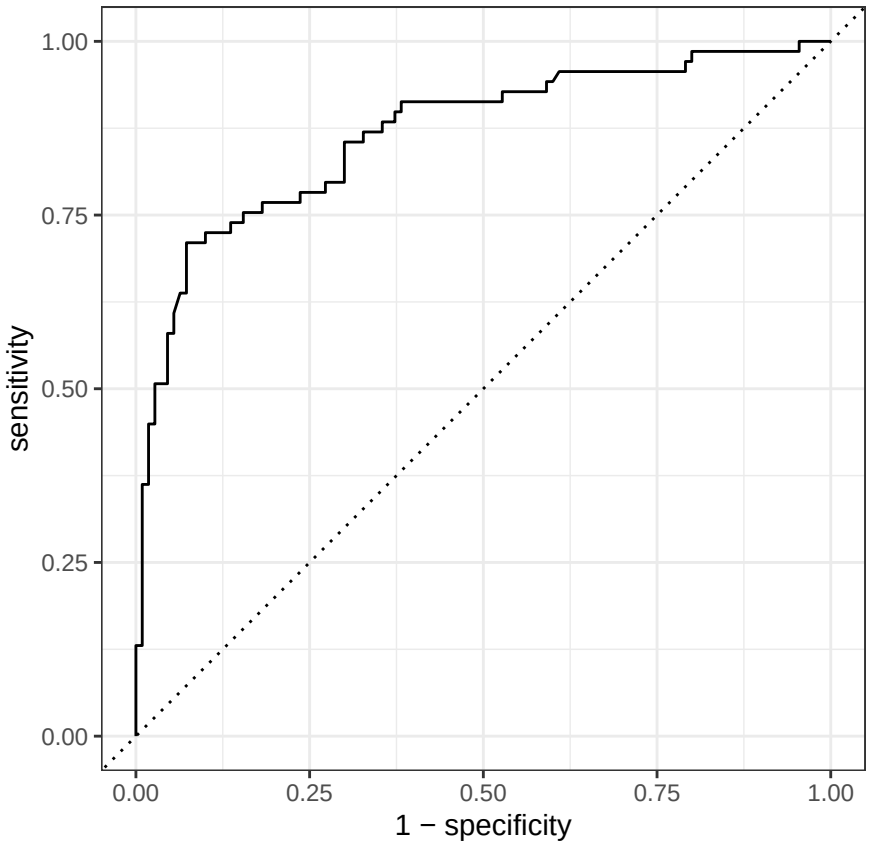
```
## # A tibble: 3 x 3
##   .metric     .estimator .estimate
##   <chr>       <chr>          <dbl>
## 1 accuracy    binary         0.799
## 2 sensitivity binary         0.754
## 3 specificity binary         0.827
```

The accuracy of the model on the testing data is 0.7988827

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




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




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

```
## # A tibble: 1 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 roc_auc binary         0.867
```

The model preforms well, and the area under ROC-curve is 0.86. The accuracy between training and testing is similar and both close to 80.