Homework 02-131

Dharynka Tapia

2022-10-14

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.4
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(tidymodels)
```

```
## -- Attaching packages -------------------------------------- tidymodels 1.0.0 --
## v broom        1.0.1     v rsample      1.1.0
## v dials        1.0.0     v tune         1.0.0
## v infer        1.0.3     v workflows    1.1.0
## v modeldata    1.0.1     v workflowsets 1.0.0
## v parsnip      1.0.2     v yardstick    1.1.0
## v recipes      1.0.1
## -- Conflicts ----------------------------------------- tidymodels_conflicts() --
## x scales::discard() masks purrr::discard()
## x dplyr::filter()   masks stats::filter()
## x recipes::fixed()  masks stringr::fixed()
## x dplyr::lag()      masks stats::lag()
## x yardstick::spec() masks readr::spec()
## x recipes::step()   masks stats::step()
## * Use suppressPackageStartupMessages() to eliminate package startup messages
```

```
library(dbplyr)
```

```
##
## Attaching package: 'dbplyr'
##
## The following objects are masked from 'package:dplyr':
##
##     ident, sql
```

```
library(readr)
library(yardstick)
library(ggplot2)
library(magrittr)
```

```
##
## Attaching package: 'magrittr'
##
## The following object is masked from 'package:purrr':
##
##     set_names
##
## The following object is masked from 'package:tidyr':
##
##     extract
```

```
abalone = read_csv("C:\\Users\\dhary\\Desktop\\homework-2\\data\\abalone.csv")
```

```
## Rows: 4177 Columns: 9
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): type
## dbl (8): longest_shell, diameter, height, whole_weight, shucked_weight, visc...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Question 1: Your goal is to predict abalone age, which is calculated as the number of rings plus 1.5. Notice there currently is no age variable in that data set. Add age to the data set

Assess and describe the distribution

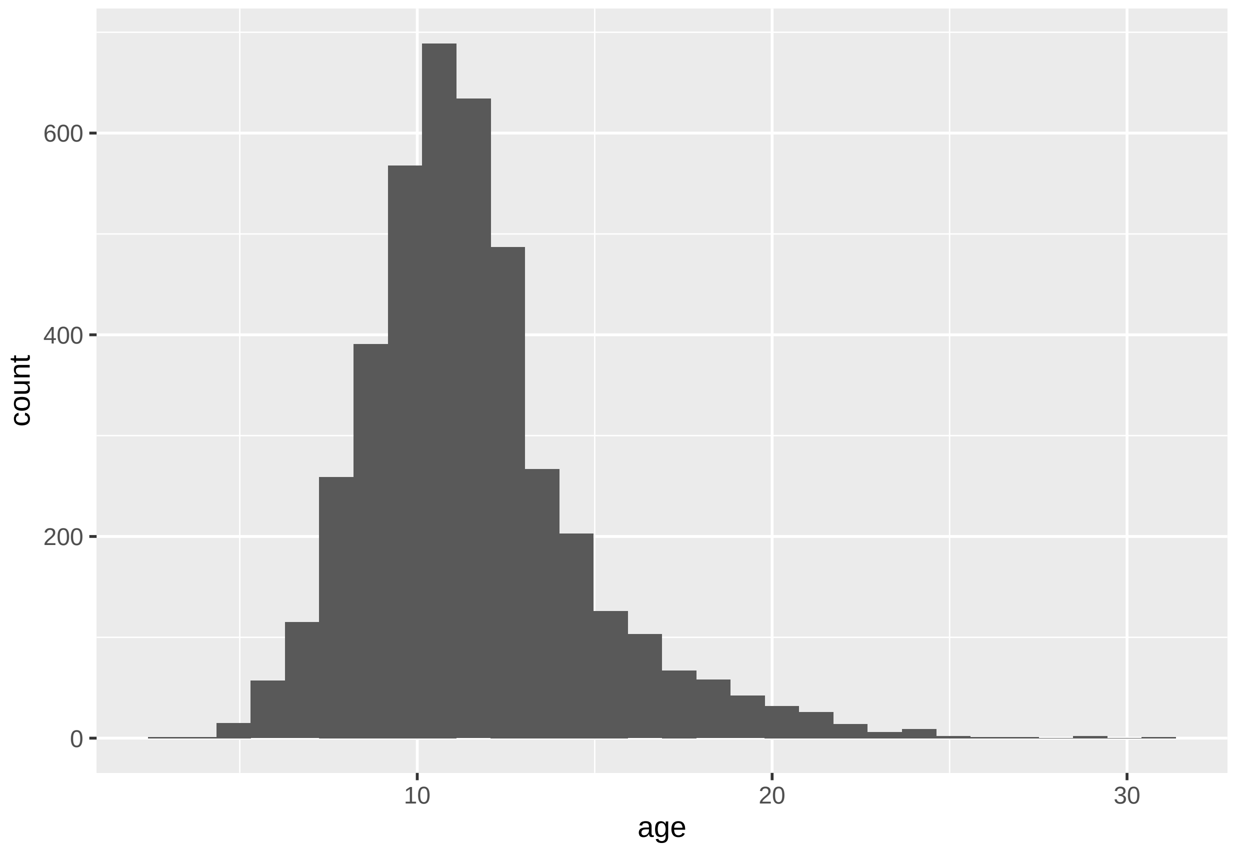
```
abalone = mutate(abalone, age = rings + 1.5)

head(abalone)
```

```
## # A tibble: 6 x 10
##   type  longest_shell diame~1 height whole~2 shuck~3 visce~4 shell~5 rings   age
##   <chr>         <dbl>   <dbl>  <dbl>   <dbl>   <dbl>   <dbl>   <dbl> <dbl> <dbl>
## 1 M             0.455   0.365  0.095   0.514  0.224   0.101    0.15     15  16.5
## 2 M             0.35    0.265  0.09    0.226  0.0995  0.0485   0.07      7   8.5
## 3 F             0.53    0.42   0.135   0.677  0.256   0.142    0.21      9  10.5
## 4 M             0.44    0.365  0.125   0.516  0.216   0.114    0.155    10  11.5
## 5 I             0.33    0.255  0.08    0.205  0.0895  0.0395   0.055     7   8.5
## 6 I             0.425   0.3    0.095   0.352  0.141   0.0775   0.12      8   9.5
## # ... with abbreviated variable names 1: diameter, 2: whole_weight,
## #   3: shucked_weight, 4: viscera_weight, 5: shell_weight
```

```
ggplot(abalone, aes(x=age))+geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



The distribution of age appears to be a positively skewed distribution, with more abalones aging around 8-12.

Question 2: Split the above data into a training set and a testing set. Use stratifies sampling, You should decide on appropriate percentages for splitting the data.

```
set.seed(2313)

abalones_split =  initial_split(abalone, prop = .80, strata = age)

abalones_train = training(abalones_split)
abalones_test = testing(abalones_split)
```

Question 3: Using the training data, create a recipe predicting the outcome variable, age, with all other predictor variables. Note that you should not include rings to predict age. Explain why you shouldn't use rings to predict age

We shouldn't use rings to predict age because rings is what is being used to account for age + 1.5

```
abalone_recipe = recipe(age ~ type + longest_shell + diameter + height + whole_weight
                        + shucked_weight + viscera_weight + shell_weight,
                        data = abalones_train)%>%
  step_dummy(all_nominal_predictors()) %>%
  step_interact(terms = ~starts_with("type"): shucked_weight + longest_shell: diameter
                + shucked_weight: shell_weight) %>%
  step_normalize(all_nominal_predictors())
```

Question 4: Create and store a linear regression object using the "lm" engine.

```
abalone_lm_model = linear_reg() %>%
  set_engine("lm")
```

Question 5: Now 1. Set up an empty workflow 2. Add the model you created in Question 4 and 3. Add the recipe that you created in Question 3

```
abalone_lm_wflow = workflow()%>%
  add_model(abalone_lm_model)%>%
  add_recipe(abalone_recipe)
```

Question 6: Use your fit() object to predict the age of hypothetical female abalone with longest_shell = 0.50, diameter = 0.10, height = 0.30, whole_weight = 4, shucked_weight = 1, viscera_weight = 2, shell_weight = 1

```
abalone_lm_fit = fit(abalone_lm_wflow, abalones_train)

abalone_lm_fit%>%
  extract_fit_parsnip() %>%
  tidy()
```

```
## # A tibble: 14 x 5
##    term                          estimate std.error statistic  p.value
##    <chr>                            <dbl>     <dbl>     <dbl>    <dbl>
##  1 (Intercept)                     4.76       0.683    6.97   3.70e-12
##  2 longest_shell                   1.36       2.36     0.578  5.63e- 1
##  3 diameter                       19.7        3.20     6.16   8.12e-10
##  4 height                         13.5        2.51     5.40   7.10e- 8
##  5 whole_weight                    9.86       0.818   12.1    7.97e-33
##  6 shucked_weight                -19.5        1.14   -17.1    9.00e-63
##  7 viscera_weight                 -9.07       1.45    -6.24   4.99e-10
##  8 shell_weight                    9.65       1.58     6.11   1.11e- 9
##  9 type_I                         -2.17       0.251   -8.67   6.45e-18
## 10 type_M                         -0.520      0.215   -2.42   1.58e- 2
## 11 type_I_x_shucked_weight         5.08       0.773    6.57   5.87e-11
## 12 type_M_x_shucked_weight         1.15       0.439    2.63   8.70e- 3
## 13 longest_shell_x_diameter      -24.0        4.25    -5.65   1.76e- 8
## 14 shucked_weight_x_shell_weight   0.0234     1.68     0.0139 9.89e- 1
```

```
new_abalone = data.frame(type = 'F', longest_shell = 0.50, diameter = 0.10, height = 0.30,
                         whole_weight = 4, shucked_weight = 1, viscera_weight = 2,
                         shell_weight = 1 )

predict(abalone_lm_fit, new_data = new_abalone)
```

```
## # A tibble: 1 x 1
##   .pred
##   <dbl>
## 1  21.8
```

The predicted age of a hypothetical female abalone with longest_shell = 0.50, diameter = 0.10, height = 0.30, whole_weight = 4, shucked_weight = 1, viscera_weight = 2, shell_weight = 1 is 22 years.

Question 7 Now you want to assess your model's performance. To do this, use the yardstick package:

1. Create a metric set that includes R^2 RMSE (Root mean squared error) and MAE(mean absolute error)

```
abalone_metrics = metric_set(rmse,rsq,mae)
```

2. Use predict() and bind_cols() to create a tibble of your model's predicted values from the training data along with the actual observed ages (these are needed to assess your model's performance)

```
abalone_train_res = predict(abalone_lm_fit, new_data = abalones_train %>% select(-age))

abalone_train_res %>%
  head()
```

```
## # A tibble: 6 x 1
##   .pred
##   <dbl>
## 1  9.57
## 2  8.04
## 3  9.20
## 4  9.61
## 5  9.99
## 6 10.8
```

```
abalone_train_res = bind_cols(abalone_train_res, abalones_train %>% select(age))

abalone_train_res %>%
  head()
```

```
## # A tibble: 6 x 2
##   .pred   age
##   <dbl> <dbl>
## 1  9.57   8.5
## 2  8.04   8.5
## 3  9.20   9.5
## 4  9.61   8.5
## 5  9.99   9.5
## 6 10.8    9.5
```

3. Finally, apply your metric set to the tibble, report the result and interpret the Rˆ2 value

```
abalone_metrics(abalone_train_res, truth = age, estimate = .pred)
```

```
## # A tibble: 3 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rmse    standard       2.17
## 2 rsq     standard       0.557
## 3 mae     standard       1.56
```

Because our Rˆ2 is right in the middle it suggest that our model may not be the best model to use to predict the age of Abalones.