PSTAT 131 HW #2

Finn Stack

4/10/2022

Reading the Abalone data:

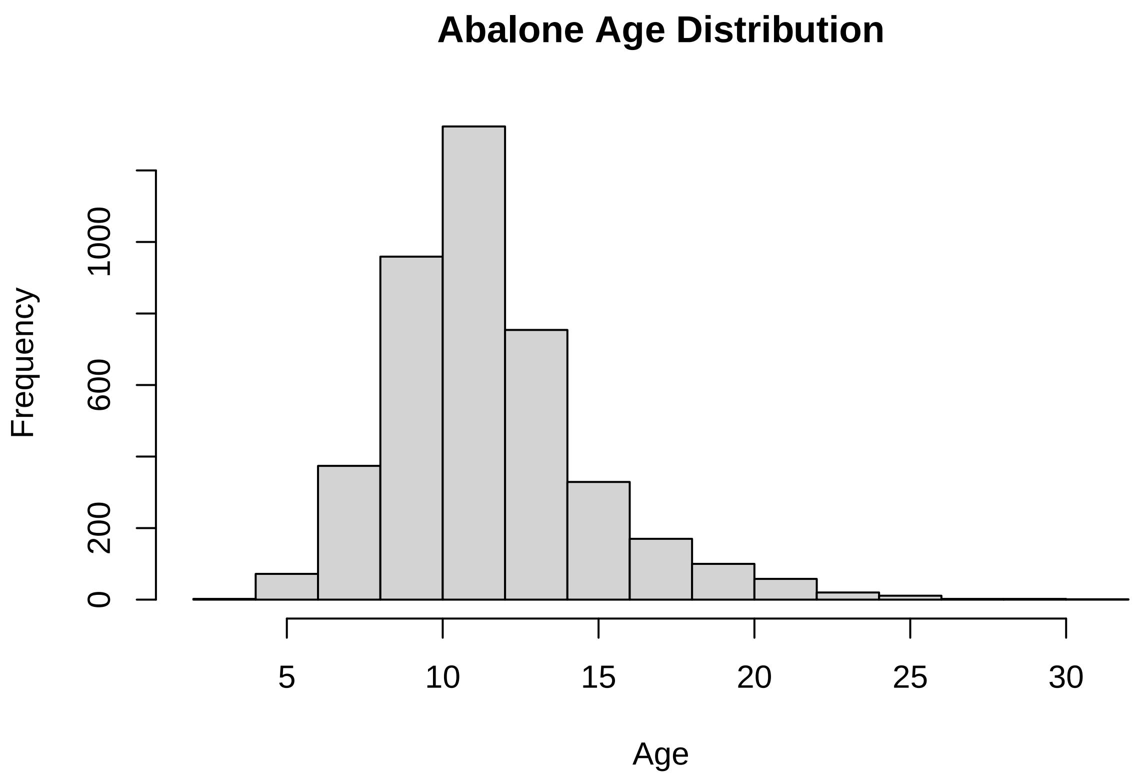
```
abalone <- read.csv("/Users/finnianstack/Desktop/School/PSTAT/PSTAT 131/HW #2/abalone.csv")
```

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 the data set. Add age to the data set.

Assess and describe the distribution of age. **NEED TO DO THIS**

```
##   type longest_shell diameter height whole_weight shucked_weight viscera_weight
## 1    M         0.455    0.365  0.095       0.5140         0.2245         0.1010
## 2    M         0.350    0.265  0.090       0.2255         0.0995         0.0485
## 3    F         0.530    0.420  0.135       0.6770         0.2565         0.1415
## 4    M         0.440    0.365  0.125       0.5160         0.2155         0.1140
## 5    I         0.330    0.255  0.080       0.2050         0.0895         0.0395
## 6    I         0.425    0.300  0.095       0.3515         0.1410         0.0775
##   shell_weight rings  age
## 1        0.150    15 16.5
## 2        0.070     7  8.5
## 3        0.210     9 10.5
## 4        0.155    10 11.5
## 5        0.055     7  8.5
## 6        0.120     8  9.5
```



As we can see from the histogram above, it appears that the age is somewhat normally distributed with approximate 11 years old being the most common.

Question 2

Split the abalone data into a training set and a testing set. Use stratified sampling. You should decide on appropriate percentages for splitting the data.

Remember that you'll need to set a seed at the beginning of the document to reproduce your results.

```
set.seed(1213)

abalone_split <- initial_split(abalone, prop = 0.80,
                               strata = age)
abalone_train <- training(abalone_split)
abalone_test <- testing(abalone_split)
```

```
abalone_recipe <-
  recipe(age ~ type + shucked_weight + longest_shell + diameter +
           shucked_weight + shell_weight, data = abalone) %>%
  step_dummy(all_nominal_predictors()) %>%
  step_scale(all_predictors()) %>% step_center(all_predictors())
```

We should not use rings to predict age because our goal with this is to find a way to predict age in a less-invasive way than cutting the abolone open and counting the rings.

Question 4

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

```
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.

```
lm_wflow <- workflow() %>%
  add_model(lm_model) %>%
  add_recipe(abalone_recipe)
```

```
lm_fit <- fit(lm_wflow, abalone_train)
```

```
lm_fit %>%
  # This returns the parsnip object:
  extract_fit_parsnip() %>%
  # Now tidy the linear model object:
  tidy()
```

```
## # A tibble: 7 x 5
##   term           estimate std.error statistic   p.value
##   <chr>             <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)     11.4       0.0381   300.    0
## 2 shucked_weight  -2.55      0.0960   -26.5   7.75e-141
## 3 longest_shell   -0.123     0.240     -0.512 6.09e-  1
## 4 diameter         1.33      0.244      5.43  6.09e-  8
## 5 shell_weight     2.92      0.100     29.2   4.65e-167
## 6 type_I          -0.471     0.0530    -8.89  9.90e- 19
## 7 type_M          -0.0343    0.0451    -0.761 4.47e-  1
```

Question 6

Use your fit() object to predict the 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.

```
female_test <- 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(lm_fit, female_test)
```

```
## # A tibble: 1 x 1
##   .pred
##   <dbl>
## 1  16.4
```

The prediction of a hypothetical female abalone with the statistics above has an age of 16.392

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 R2, RMSE (root mean squared error), and MAE (mean absolute error)
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).
3. Finally, apply your metric set to the tibble, report the results, and interpret the R2 value.

```
##    type longest_shell diameter height whole_weight shucked_weight
## 5     I         0.330    0.255  0.080       0.2050         0.0895
## 6     I         0.425    0.300  0.095       0.3515         0.1410
## 17    I         0.355    0.280  0.085       0.2905         0.0950
## 19    M         0.365    0.295  0.080       0.2555         0.0970
## 36    M         0.465    0.355  0.105       0.4795         0.2270
## 38    F         0.450    0.355  0.105       0.5225         0.2370
##    viscera_weight shell_weight rings age
## 5          0.0395        0.055     7 8.5
## 6          0.0775        0.120     8 9.5
## 17         0.0395        0.115     7 8.5
## 19         0.0430        0.100     7 8.5
## 36         0.1240        0.125     8 9.5
## 38         0.1165        0.145     8 9.5
```

```
## # A tibble: 6 x 1
##   .pred
##   <dbl>
## 1  8.14
## 2  9.43
## 3  9.65
## 4 10.4
## 5 10.2
## 6 10.6
```

```
## # A tibble: 6 x 11
##   .pred type  longest_shell diameter height whole_weight shucked_weight
##   <dbl> <chr>         <dbl>    <dbl>  <dbl>        <dbl>          <dbl>
## 1  8.14 I             0.33     0.255  0.08         0.205         0.0895
## 2  9.43 I             0.425    0.3    0.095        0.352         0.141
## 3  9.65 I             0.355    0.28   0.085        0.290         0.095
## 4 10.4  M             0.365    0.295  0.08         0.256         0.097
## 5 10.2  M             0.465    0.355  0.105        0.480         0.227
## 6 10.6  F             0.45     0.355  0.105        0.522         0.237
## # ... with 4 more variables: viscera_weight <dbl>, shell_weight <dbl>,
## #   rings <int>, age <dbl>
```

```
## # A tibble: 1 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rmse    standard        2.20
## # A tibble: 3 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rmse    standard       2.20
## 2 rsq     standard       0.521
## 3 mae     standard       1.60
```

Based off of this data and the R squared value of 0.52, we can determine that this did a pretty good job of predicting. I.e. it was a relatively good fit.