Homework 2

PSTAT 131/231

Contents

Linear Regression

For this lab, we will be working with a data set from the UCI (University of California, Irvine) Machine Learning repository (see website here). The full data set consists of $4,177$ observations of abalone in Tasmania. (Fun fact: Tasmania supplies about 25% of the yearly world abalone harvest.)

The age of an abalone is typically determined by cutting the shell open and counting the number of rings with a microscope. The purpose of this data set is to determine whether abalone age (**number of rings + 1.5**) can be accurately predicted using other, easier-to-obtain information about the abalone.

The full abalone data set is located in the `\data` subdirectory. Read it into *R* using `read_csv()`. Take a moment to read through the codebook (`abalone_codebook.txt`) and familiarize yourself with the variable definitions.

Make sure you load the `tidyverse` and `tidymodels`!

```
library(ggplot2)
library(tidyverse)
library(tidymodels)
library(corrplot)
library(ggthemes)
tidymodels_prefer()
tidymodels_packages()
```

```
##  [1] "broom"        "cli"          "conflicted"   "dials"        "dplyr"
##  [6] "ggplot2"      "hardhat"      "infer"        "modeldata"    "parsnip"
## [11] "purrr"        "recipes"      "rlang"        "rsample"      "rstudioapi"
## [16] "tibble"       "tidyr"        "tune"         "workflows"    "workflowsets"
## [21] "yardstick"    "tidymodels"
```

```
abalone <- read.csv("abalone.csv")
head(abalone)
```

```
##   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
## 1        0.150    15
## 2        0.070     7
## 3        0.210     9
## 4        0.155    10
## 5        0.055     7
## 6        0.120     8
```

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.

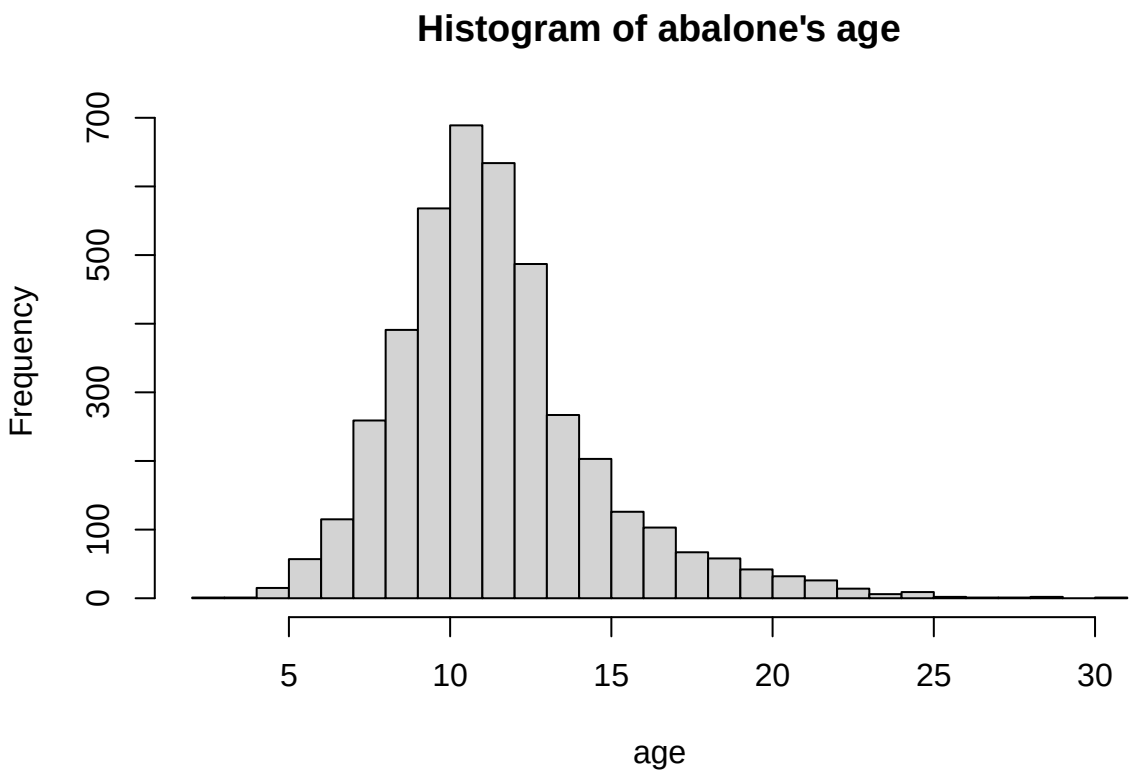
Assess and describe the distribution of `age`.

```
abalone_new <- abalone %>%
  mutate(age= rings+1.5)

head(abalone_new)
```

```
##   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
```

```
hist(abalone_new$age, xlab = "age",breaks =20, main = "Histogram of abalone's age")
```



From the histogram we can see that the age of abalone is uneven distribute and the overall shape of the distribution is skewed to the left with a clear mode around age 11, most abalone's age is between 8-13 and there has outlier around 30.

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(2022)
abalone_split <- initial_split(abalone_new, prop = 0.80, strata = age)
abalone_train <- training(abalone_split)
abalone_test <- testing(abalone_split)
head(abalone_train)
```

```
##    type longest_shell diameter height whole_weight shucked_weight
## 2     M         0.350    0.265  0.090       0.2255         0.0995
## 5     I         0.330    0.255  0.080       0.2050         0.0895
## 6     I         0.425    0.300  0.095       0.3515         0.1410
## 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
## 2          0.0485        0.070     7 8.5
## 5          0.0395        0.055     7 8.5
## 6          0.0775        0.120     8 9.5
## 19         0.0430        0.100     7 8.5
```

```
## 36         0.1240        0.125     8 9.5
## 38         0.1165        0.145     8 9.5
```

```
head(abalone_test)
```

```
##    type longest_shell diameter height whole_weight shucked_weight
## 12    M         0.430     0.35  0.110       0.4060         0.1675
## 17    I         0.355     0.28  0.085       0.2905         0.0950
## 18    F         0.440     0.34  0.100       0.4510         0.1880
## 23    F         0.565     0.44  0.155       0.9395         0.4275
## 35    F         0.705     0.55  0.200       1.7095         0.6330
## 40    M         0.355     0.29  0.090       0.3275         0.1340
##    viscera_weight shell_weight rings  age
## 12         0.0810        0.135    10 11.5
## 17         0.0395        0.115     7  8.5
## 18         0.0870        0.130    10 11.5
## 23         0.2140        0.270    12 13.5
## 35         0.4115        0.490    13 14.5
## 40         0.0860        0.090     9 10.5
```

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`.

Steps for your recipe:

1. dummy code any categorical predictors
2. create interactions between
    - `type` and `shucked_weight`,
    - `longest_shell` and `diameter`,
    - `shucked_weight` and `shell_weight`
3. center all predictors, and
4. scale all predictors.

You'll need to investigate the `tidymodels` documentation to find the appropriate step functions to use.

```
abalone_recipe <- abalone_train %>%
  recipe(age ~ type +longest_shell + diameter + height +
           whole_weight + shucked_weight + viscera_weight +
           shell_weight) %>%
  step_dummy(all_nominal_predictors())  %>%
  step_interact(terms = ~ starts_with("type"):shucked_weight+
                  longest_shell:diameter+
                  shucked_weight:shell_weight) %>%
  step_center(all_predictors()) %>%
  step_scale(all_predictors())
```

Since we calculate the age by using rings plus 1.5 which means the variable age and variable rings are dependent, and the purpose of this data set is to determine whether abalone age (number of rings + 1.5) can be accurately predicted using other, easier-to-obtain information about the abalone. So we want to use independent variable to do the prediction.

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)
```

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.

```
# fit the linear model to the training set
lm_fit <- fit(lm_wflow, abalone_train)
```

```
# Predict female abalone with longest_shell = 0.50, diameter = 0.10, height = 0.30, whole_weight = 4, s
abalone_hypo <- 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,new_data = abalone_hypo)
```

```
## # A tibble: 1 x 1
##   .pred
##   <dbl>
## 1  26.4
```

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).

```
# generates predicted values for age for each observation in the training set:
abalone_train_res <- predict(lm_fit, new_data = abalone_train %>% select(-age))

# attach a column with the actual observed age observations:
abalone_train_res <- bind_cols(abalone_train_res, abalone_train %>% select(age))
abalone_train_res %>%
  head()
```

```
## # A tibble: 6 x 2
##   .pred   age
##   <dbl> <dbl>
## 1  9.57   8.5
## 2  8.09   8.5
## 3  9.32   9.5
## 4 10.5    8.5
## 5 10.1    9.5
## 6 10.9    9.5
```

3. Finally, apply your metric set to the tibble, report the results, and interpret the R^2 value.

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

```
## # A tibble: 3 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rmse    standard       2.14
## 2 rsq     standard       0.557
## 3 mae     standard       1.55
```

rmse: The root mean squared error is equal to 2.1412871.Since we are predicting the age of the abalone, the value 2.14 kind is 'big' to us.
mae: The magnitude of difference between the prediction of an observation and the true value of that observation is 1.5567.
R^2: The R^2 represent how well the regression model fits the observed data. Since the rsq=0.5567, we cannot say the regression model fits well.