Homework Assignment 2

Reynaldo Perez

October 16, 2022

```r
library(tidyverse)  # Load 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   1.2.0      v stringr 1.4.0
## v readr   2.1.2      v forcats 0.5.2

## Warning: package 'tidyr' was built under R version 4.0.5

## Warning: package 'readr' was built under R version 4.0.5

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(tidymodels)  # Load tidymodels
```

```
## -- Attaching packages -------------------------------------- tidymodels 1.0.0 --
## v broom        1.0.1     v rsample      1.1.0
## v dials        1.0.0     v tune         1.0.1
## 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()
## * Search for functions across packages at https://www.tidymodels.org/find/
```

```r
abalone <- read.csv("/Users/reynaldoperez/Downloads/homework-2-2/data/abalone.csv")  # Read the data se

names(abalone)  # See the names and number of columns of the data set
```

```
## [1] "type"           "longest_shell"  "diameter"       "height"
## [5] "whole_weight"   "shucked_weight" "viscera_weight" "shell_weight"
## [9] "rings"
```

Q1) Let's add a new variable, named *age*, to the data set.

```r
age <- abalone$rings + 1.5  # Calculate age
```

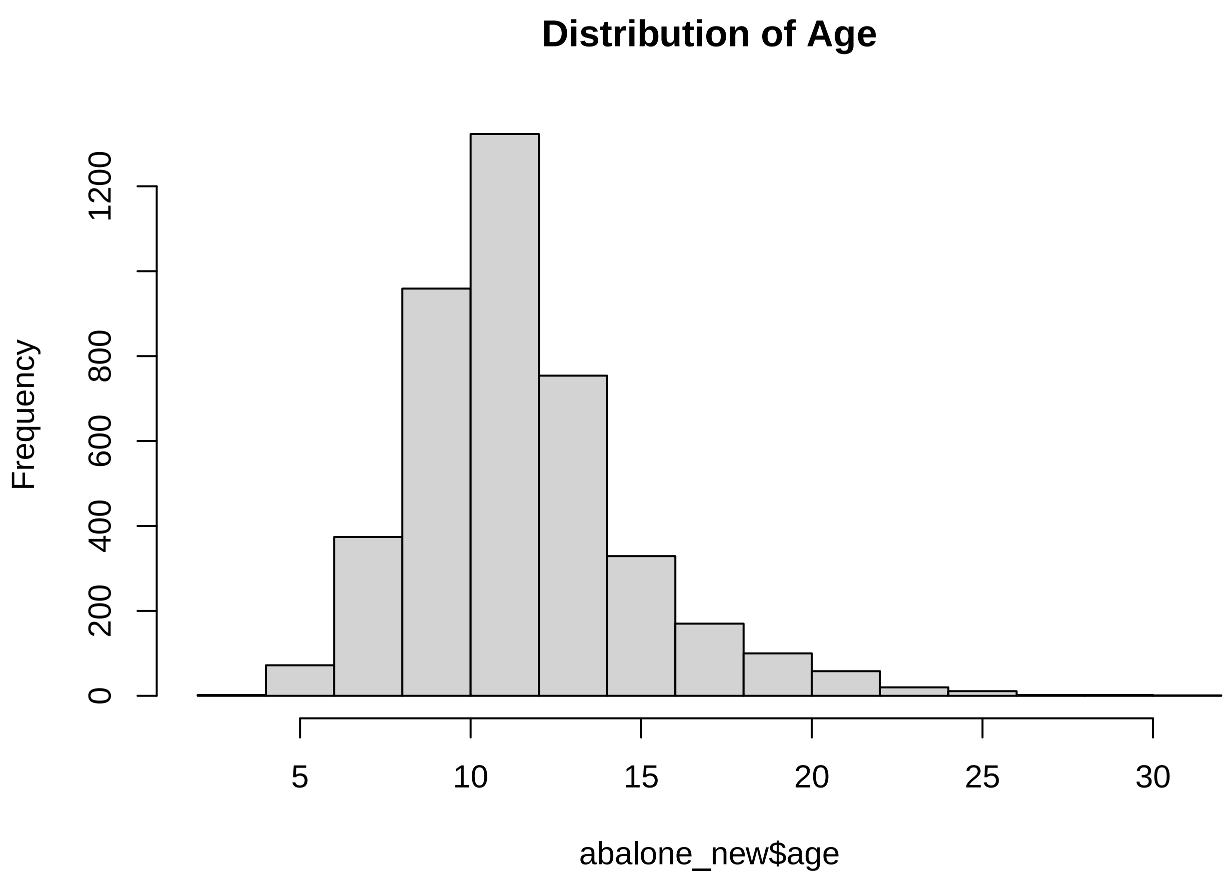
```r
abalone_new <- cbind(abalone, age)  # Add new variable to the dataset

head(abalone_new)  # Check
```

```
##   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
```

Now, let us assess the distribution of *age*:

```r
hist(abalone_new$age, breaks = "Sturges", main = paste("Distribution of Age"))
```



As one can see, the distribution of *age* is slightly skewed to the left, with the highest peak at between 10 to ~12 years.

Q2) We will now split the abalone data into a training set and a testing set. We will use stratified sampling.

```
set.seed(1115)

abalone_split <- initial_split(abalone_new, prop = 0.75, strata = age)

abalone_train <- training(abalone_split)
abalone_test <- testing(abalone_split)
```

Q3) Let us create a recipe for predicting the outcome variable, *age*:

```
simple_abalone_recipe <- recipe(age ~ ., data = abalone_train)

simple_abalone_recipe
```

```
## Recipe
##
## Inputs:
##
##       role #variables
##    outcome          1
##  predictor          9
```

Now, we will complete the recipe:

```
abalone_recipe <- recipe(age ~ type + longest_shell + diameter + height + whole_weight + shucked_weight
  step_dummy_multi_choice(starts_with("type")) %>%
  prep() %>%
  step_interact(terms = ~type_M:shucked_weight) %>%
  step_interact(terms = ~type_F:shucked_weight) %>%
  step_interact(terms = ~type_I:shucked_weight) %>%
  step_interact(terms = ~longest_shell:diameter) %>%
  step_interact(terms = ~shucked_weight:shell_weight) %>%
  step_center(all_predictors()) %>%
  step_scale(all_predictors())
```

Hence, our recipe is finished. Note that we did not include the *rings* variable in our recipe. This is because obtaining the number of rings is a very time-consuming task, and the other observed measurements would help predict the age much faster.

Q4) Now, we will create and store a linear regression object:

```
lm_model <- linear_reg() %>%
  set_engine("lm")
```

Q5) We will now develop an empty workflow, and add the model and recipe we created in the previous questions:

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

Q6) Let's now use the *fit*() object to predict the age of a hypothetical female abalone with the given information.

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

lm_fit %>%
  extract_fit_parsnip() %>%
  tidy()
```

```
## # A tibble: 16 x 5
##    term                          estimate std.error statistic   p.value
##    <chr>                            <dbl>     <dbl>     <dbl>     <dbl>
##  1 (Intercept)                     11.4      0.0380   301.     0
##  2 longest_shell                    0.563    0.291      1.93   5.32e- 2
##  3 diameter                         2.34     0.320      7.30   3.71e-13
##  4 height                           0.218    0.0703     3.09   2.00e- 3
##  5 whole_weight                     5.19     0.389     13.3    1.47e-39
##  6 shucked_weight                  -3.50     0.268    -13.1    5.69e-38
##  7 viscera_weight                  -0.936    0.158     -5.93   3.43e- 9
##  8 shell_weight                     1.67     0.218      7.67   2.26e-14
##  9 type_F                           0.314    0.103      3.06   2.24e- 3
## 10 type_I                          -0.607    0.103     -5.88   4.48e- 9
## 11 type_M                          NA       NA         NA     NA
## 12 type_M_x_shucked_weight         -0.641    0.177     -3.62   2.94e- 4
## 13 type_F_x_shucked_weight         -0.941    0.177     -5.32   1.11e- 7
## 14 type_I_x_shucked_weight         NA       NA         NA     NA
## 15 longest_shell_x_diameter        -3.20     0.410     -7.82   7.34e-15
## 16 shucked_weight_x_shell_weight   -0.189    0.205     -0.923  3.56e- 1
```

```
x0 <- data.frame(type = "type_F", longest_shell = 0.5, diameter = 0.1, height = 0.3, whole_weight = 4, s
x0  # Display data frame
```

```
##     type longest_shell diameter height whole_weight shucked_weight
## 1 type_F           0.5      0.1    0.3            4              1
##   viscera_weight shell_weight
## 1              2            1
```

```
## predict.lm(lm_fit, new_data = x0)  # Predicted age, but received error saying model cannot include N
```

Q7) Now, we will assess our model's performance.

```
library(yardstick)

abalone_train_res <- predict(lm_fit, new_data = abalone_train %>% select(-age))  # Develop predicted va
```

```
## Warning in predict.lm(object = object$fit, newdata = new_data, type =
## "response"): prediction from a rank-deficient fit may be misleading
```

```
abalone_train_res %>%
  head()
```

```
## # A tibble: 6 x 1
##   .pred
##   <dbl>
## 1  9.45
## 2  8.17
## 3  9.46
## 4  9.93
## 5 10.4
## 6 10.0
```

Now, we will develop the metric sets:

```
abalone_metrics <- metric_set(rmse, rsq, mae)
## abalone_metrics(abalone_train_res, truth = age, estimate = .pred)  # Error saying length of "truth" a
```

Then, create a tibble of the model's predicted values:

```
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  9.45   8.5
## 2  8.17   8.5
## 3  9.46   9.5
## 4  9.93   8.5
## 5 10.4    8.5
## 6 10.0    9.5
```

As one can see, the predicted value is not that far off the actual value of age. The R^2 value we calculated is the percentage amount that the variability observed in *age* is explained by the regression model.