Pstat 131 Homework 2

Yu Tian

Spring 2022-04-10

Linear Regression

```
# Read the full abalone data set into R using read_csv()
abalone <- read.csv(file = 'abalone.csv')
abalone %>% head()
```

```
##   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.

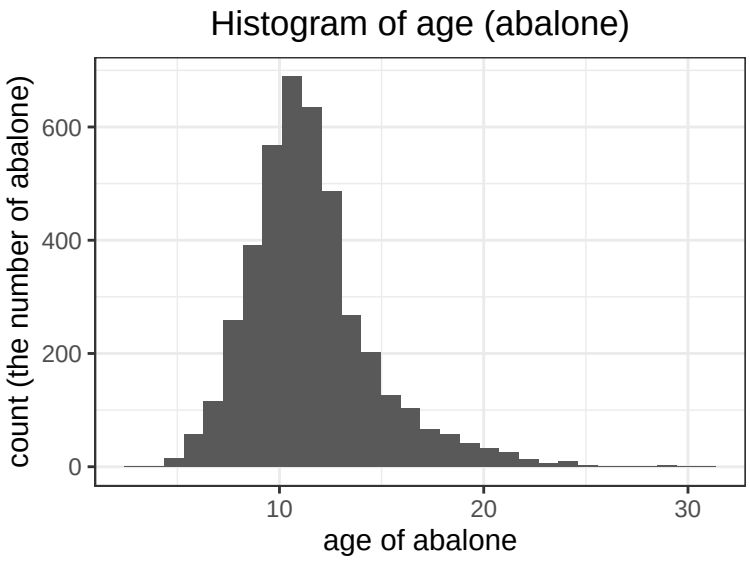
Answer: Q1

```
# create the variable age in the data set
abalone["age"] <- abalone$rings + 1.5
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  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
```

```
#the distribution of age
abalone %>%
  ggplot(aes(x=age)) +
  geom_histogram(bins=30)+
  labs(title="Histogram of age (abalone)",
       x ="age of abalone", y = "count (the number of abalone)") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5))
```



From the histogram, we can find that the distribution is slight skewed, and most abalones' age are distributed around 10. There is a few abalones with age higher than 25.

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.

Answer: Q2

```
# set a seed
set.seed(0623)

# split the abalone data into a training set and a testing set.
abalone_split <- initial_split(abalone, prop = 0.80, strata = age)
abalone_train <- training(abalone_split)
abalone_test <- testing(abalone_split)
dim(abalone)
```

```
## [1] 4177   10
```

```
dim(abalone_train)
```

```
## [1] 3340   10
```

```
dim(abalone_test)
```

```
## [1] 837  10
```

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:

dummy code any categorical predictors

create interactions between

type and shucked_weight, longest_shell and diameter, shucked_weight and shell_weight center all predictors, and scale all predictors.

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

Answer: Q3

We shouldn't use rings to predict age since the new variable "age" is directly calculated from variable "rings".

```
# remove variable ring
new_abalone_train <- abalone_train %>% select(-rings)
# confirm remove rings and show the new data set
head(new_abalone_train)
```

```
##    type longest_shell diameter height whole_weight shucked_weight
## 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
## 43    I         0.240    0.175  0.045       0.0700         0.0315
##    viscera_weight shell_weight age
## 6          0.0775        0.120 9.5
## 17         0.0395        0.115 8.5
## 19         0.0430        0.100 8.5
## 36         0.1240        0.125 9.5
## 38         0.1165        0.145 9.5
## 43         0.0235        0.020 6.5
```

```
abalone_recipe <- recipe(age ~ ., data = new_abalone_train) %>%
  # dummy code any categorical predictors
  step_dummy(all_nominal_predictors()) %>%
  # create interactions between type and shucked_weight
  step_interact(terms = ~type:shucked_weight) %>%
  # create interactions between longest_shell and diameter
  step_interact(terms = ~longest_shell:diameter) %>%
  # create interactions between shucked_weight and shell_weight
  step_interact(terms = ~shucked_weight:shell_weight) %>%
  # center all predictors
  step_center(all_nominal_predictors()) %>%
  # scale all predictors
  step_scale(all_nominal_predictors())
```

```
abalone_recipe
```

```
## Recipe
##
## Inputs:
##
##       role #variables
##    outcome          1
##  predictor          8
##
## Operations:
##
## Dummy variables from all_nominal_predictors()
## Interactions with type:shucked_weight
## Interactions with longest_shell:diameter
## Interactions with shucked_weight:shell_weight
## Centering for all_nominal_predictors()
## Scaling for all_nominal_predictors()
```

Question 4

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

Answer: Q4

```
# create and store a linear regression object
# using the "lm" engine.
lm_model <- linear_reg() %>%
  set_engine("lm")
```

Question 5

Now:

set up an empty workflow, add the model you created in Question 4, and add the recipe that you created in Question 3.

Answer: Q5

```
lm_wflow <- workflow() %>%       # set up an empty workflow
  add_model(lm_model) %>%                # add the model i created in Question 4
  add_recipe(abalone_recipe)             # add the recipe that i created in Question 3
```

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.

Answer: Q6

```
lm_fit <- fit(lm_wflow, new_abalone_train)
lm_fit %>%
  # This returns the parsnip object:
  extract_fit_parsnip() %>%
```

```
  # Now tidy the linear model object:
  tidy()
```

```
## # A tibble: 12 x 5
##    term                          estimate std.error statistic  p.value
##    <chr>                            <dbl>     <dbl>     <dbl>    <dbl>
##  1 (Intercept)                     2.33      0.596      3.91  9.24e- 5
##  2 longest_shell                   7.40      2.34       3.16  1.60e- 3
##  3 diameter                       23.1       3.11       7.45  1.22e-13
##  4 height                          6.26      1.67       3.75  1.77e- 4
##  5 whole_weight                   11.2       0.817     13.6   2.65e-41
##  6 shucked_weight                -19.1       1.09     -17.5   5.38e-66
##  7 viscera_weight                 -9.18      1.46      -6.29  3.50e-10
##  8 shell_weight                   11.0       1.51       7.26  4.82e-13
##  9 type_I                         -0.760     0.116     -6.57  5.78e-11
## 10 type_M                          0.0171    0.0923     0.185 8.53e- 1
## 11 longest_shell_x_diameter      -35.2       4.11      -8.56  1.75e-17
## 12 shucked_weight_x_shell_weight  -2.45      1.68      -1.46  1.44e- 1
```

```
predict_abalone <- data.frame(type = 'F', longest_shell = 0.50, diameter = 0.10, height = 0.30, whole_w
                              )
predict_abalone_res <- predict(lm_fit, new_data = predict_abalone)
predict_abalone_res
```

```
## # A tibble: 1 x 1
##   .pred
##   <dbl>
## 1  24.2
```

the predicted age of a hypothetical female abalone is 24.17389

Question 7

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

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

Answer: Q7

```
abalone_train_res <- predict(lm_fit, new_data = new_abalone_train %>% select(-age))
abalone_train_res
```

```
## # A tibble: 3,340 x 1
##    .pred
##    <dbl>
##  1  9.57
##  2 10.0
##  3 10.3
##  4 10.0
##  5 10.7
##  6  6.39
##  7  5.80
##  8  5.99
```

```
##  9  8.83
## 10 11.4
## # ... with 3,330 more rows
```

```
abalone_train_res <- bind_cols(abalone_train_res, new_abalone_train %>% select(age))
abalone_train_res
```

```
## # A tibble: 3,340 x 2
##    .pred   age
##    <dbl> <dbl>
##  1  9.57   9.5
##  2 10.0    8.5
##  3 10.3    8.5
##  4 10.0    9.5
##  5 10.7    9.5
##  6  6.39   6.5
##  7  5.80   6.5
##  8  5.99   5.5
##  9  8.83   8.5
## 10 11.4    8.5
## # ... with 3,330 more rows
```

```
abalone_metrics <- metric_set(rsq, rmse, mae)
abalone_metrics(abalone_train_res, truth = age,
                estimate = .pred)
```

```
## # A tibble: 3 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rsq     standard       0.558
## 2 rmse    standard       2.16
## 3 mae     standard       1.55
```

In my model,

the $R2$ is 0.5584123 (~ 0.56)

the $RMSE$ is 2.1607721 (~2.16)

the MAE is 1.5509346 (~1.55)

Thus, the R2 value means 56% of response variable fit the regression model and be explained by predictor variable.