hwk_2_131

cristian razo

4/10/2022

1

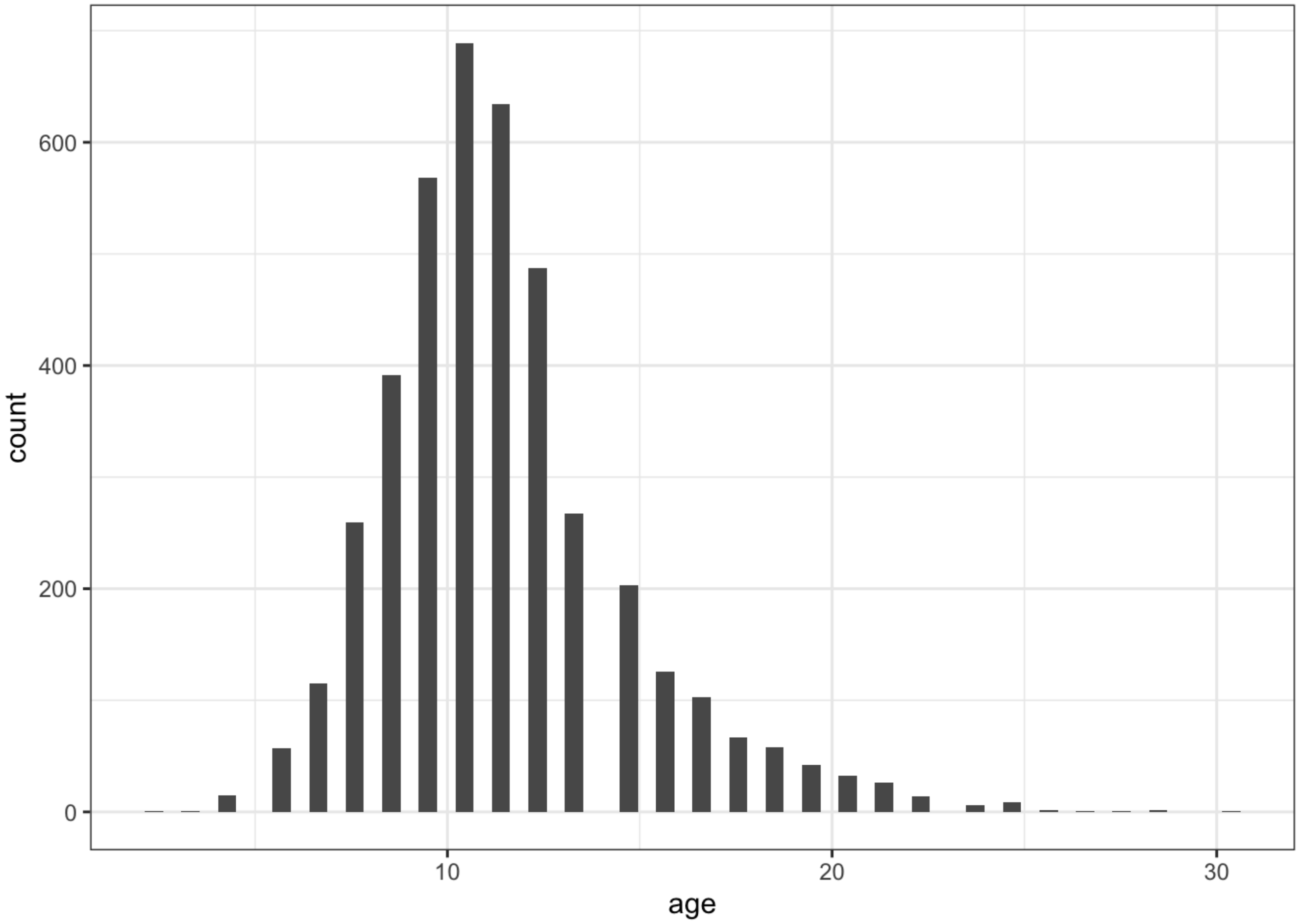
```
library(ggplot2)
library(tidyverse)
library(tidymodels)
library(corrplot)
library(ggthemes)
tidymodels_prefer()
library(readr)
```

```
##loading data set
abalone <- read_csv("Downloads/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.
```

```
abalone[10] <- abalone[9]+1.5
colnames(abalone)[10] <- 'age'

## script for histogram for 60 bins
abalone %>%
  ggplot(aes(x = age)) +
  geom_histogram(bins = 60) +
  theme_bw()
```



I have contrusted a histogram to understand the “age” feature distribution. The visual of the histogram shows us that this is a heavy right tail distribution. This means that most of our data lies in the lower bound of our distribution . It also seems that it almost resembles a normal distribution but there is values in the upper bound preventing it to be . This distribution is more suited to be represented by a T distribution because of its uneven lower and upper bound.

2

```
## Splitting data set to training and testing set that are stratified based on age

set.seed(3435)
abalone_norings=abalone[-9]
abalone_norings
```

```
## # A tibble: 4,177 × 9
##    type  longest_shell diameter height whole_weight shucked_weight
##    <chr>         <dbl>    <dbl>  <dbl>        <dbl>          <dbl>
##  1 M             0.455    0.365  0.095        0.514         0.224
##  2 M             0.35     0.265  0.09         0.226         0.0995
##  3 F             0.53     0.42   0.135        0.677         0.256
##  4 M             0.44     0.365  0.125        0.516         0.216
##  5 I             0.33     0.255  0.08         0.205         0.0895
##  6 I             0.425    0.3    0.095        0.352         0.141
##  7 F             0.53     0.415  0.15         0.778         0.237
##  8 F             0.545    0.425  0.125        0.768         0.294
##  9 M             0.475    0.37   0.125        0.509         0.216
## 10 F             0.55     0.44   0.15         0.894         0.314
## # … with 4,167 more rows, and 3 more variables: viscera_weight <dbl>,
## #   shell_weight <dbl>, age <dbl>
```

```
split <- initial_split(abalone_norings, prop = 0.80, strata = age)
train <- training(split)
test <- testing(split)
```

Made a data frame that doesnt include the column “rings” because age and ring are dependent on each other . The way we were able to compute age is by ring and ring can be treated as its real label. If we included this in the LM then it wont be able to do much learning if it know the true label . It also wont help us understand the relationship between each predictors when we run the LM . That is the reason why we dont include ring column in our LM. We did strafied sampling based on the column age.

3

```
albone_recipe <- recipe(age ~ ., data = train) %>%
  step_dummy(all_nominal()) %>%
  step_interact(terms = ~ starts_with('type') :shucked_weight) %>%
  step_interact(terms= ~ longest_shell:diameter) %>%
  step_interact(terms = ~ shucked_weight:shell_weight) %>%
  step_normalize(all_predictors())
```

The first step in recipe is to make all nominal variables into dummy indicator variabes . The I made three different step interact predictors based on the arguments I gave . The last step included is normalizing all my predictors this enables me tobe able to center and scale all my predictors my standardizing them .

4

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

I created and stored my linear regression with the “LM” engine into a variables to be used later .

5

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

Now I created a workflow where it is able in to integrate the model and recipe I created. The work flow is suited for the albone data file given and the response variable “age”.

6

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

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

```
## # A tibble: 14 × 5
##    term                          estimate std.error statistic  p.value
##    <chr>                            <dbl>     <dbl>     <dbl>    <dbl>
##  1 (Intercept)                   11.4        0.0375  305.     0       
##  2 longest_shell                  0.591      0.286     2.07   3.86e- 2
##  3 diameter                       2.06       0.313     6.61   4.59e-11
##  4 height                         0.236      0.0696    3.39   7.10e- 4
##  5 whole_weight                   4.29       0.387    11.1    4.66e-28
##  6 shucked_weight                -4.06       0.250   -16.2    5.35e-57
##  7 viscera_weight                -0.792      0.158    -5.00   6.12e- 7
##  8 shell_weight                   1.74       0.212     8.20   3.32e-16
##  9 type_I                        -0.942      0.117    -8.07   9.36e-16
## 10 type_M                        -0.239      0.104    -2.29   2.21e- 2
## 11 type_I_x_shucked_weight        0.525      0.0876    5.99   2.26e- 9
## 12 type_M_x_shucked_weight        0.293      0.109     2.68   7.41e- 3
## 13 longest_shell_x_diameter      -2.75       0.396    -6.95   4.32e-12
## 14 shucked_weight_x_shell_weight -0.00330    0.205    -0.0161 9.87e- 1
```

```
## the values were told to inpute into our model
param=data.frame(type='F',longest_shell = 0.50, diameter = 0.10, height = 0.30, whole_we
ight = 4, shucked_weight = 1, viscera_weight = 2, shell_weight = 1)

predict(lm_fit, new_data = param)
```

```
## # A tibble: 1 × 1
##   .pred
##   <dbl>
## 1  23.7
```

Now I am finally to fit my training the and the LM workflow I created for the albone data set. With the input values given to us the model predicted that the age will be 23.7. I can see that the interaction between shucked weight and shell weight have a p value that failed to reject the null hypothesis which suggest its not significant for our LM.

7

```
train_res <- predict(lm_fit, new_data = train %>% select(-age))

train_res <- bind_cols(train_res, train %>% select(age))

metrics=metric_set(rmse,mae)
metrics(train_res, truth = age, estimate = .pred)
```

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

I use the metric RMSE and MAE to check how well the model is able fit the training data.The R^2 helps us understand how much of the data is explained by variance of the bias . The R^2 is over one meaning I have possibly overfitted my data since my model is able to explain much of the error, A good R^2 is known to be in the interval of (.3,.5). All my predictors except the interaction between shucked weight and shell weight passed the p test meaning they are significant to my LM.A transformation on the predictors and removing the insignificant interaction could possibly give us a better R^2 score.