PSTAT_131_HW2

Branson Enani

2022-10-10

```r
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v tibble  3.1.6      v dplyr   1.0.10
## v tidyr   1.2.0      v stringr 1.4.0
## v readr   2.1.2      v forcats 0.5.1
## v purrr   0.3.4

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

```r
library(tidymodels)
```

```
## -- Attaching packages -------------------------------------- tidymodels 0.2.0 --

## v broom        0.8.0     v rsample      0.1.1
## v dials        0.1.1     v tune         0.2.0
## v infer        1.0.0     v workflows    0.2.6
## v modeldata    0.1.1     v workflowsets 0.2.1
## v parsnip      0.2.1     v yardstick    0.0.9
## v recipes      0.2.0

## -- 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()
## * Learn how to get started at https://www.tidymodels.org/start/
```

```r
library(corrplot)
```

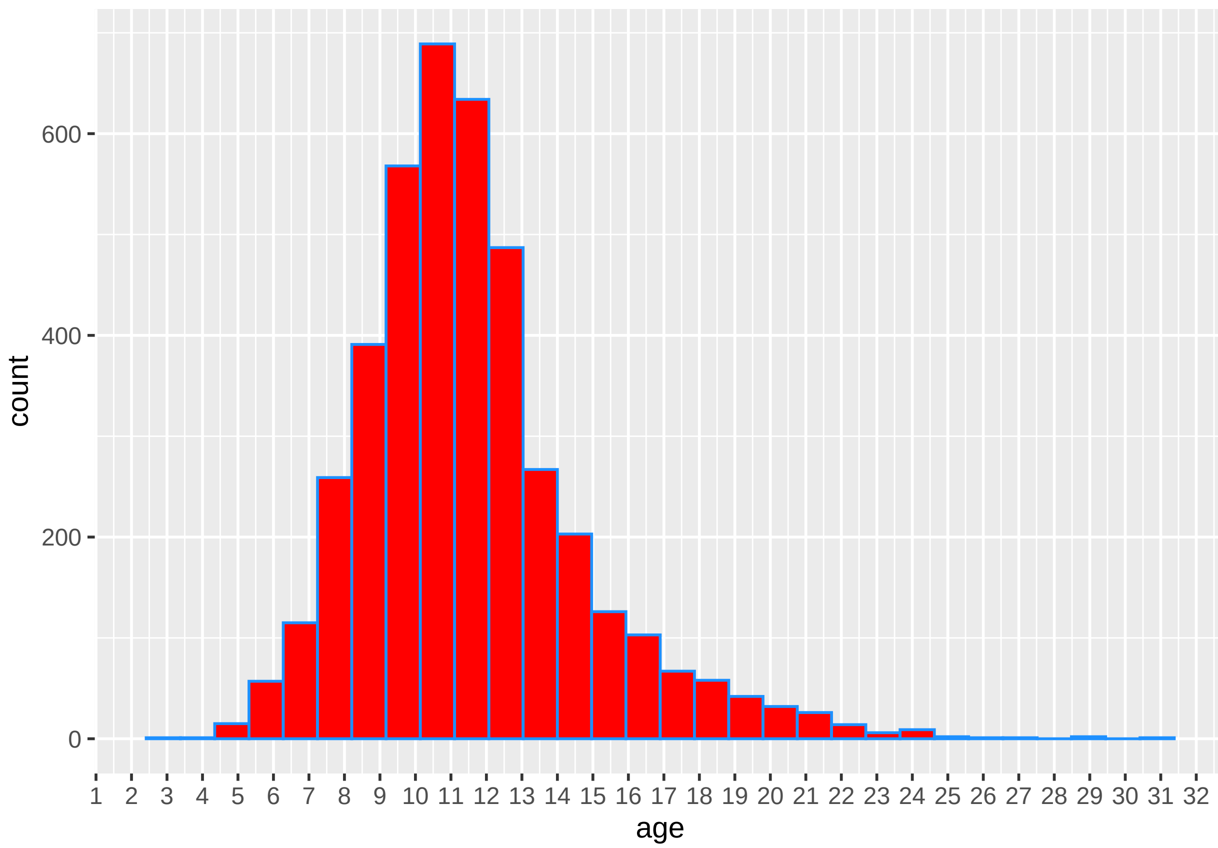
```
## corrplot 0.92 loaded
```

```r
library(ggthemes)
abalone <- read.csv('/Users/kerouac/Downloads/homework-2/data/abalone.csv')
```

1

```r
age <- 1.5+(abalone$rings)
abalone$Age <- age
ggplot2::ggplot(data = abalone, aes(x=age ))+geom_histogram(color = 'DodgerBlue', fill = 'Red')+scale_x_
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



We can see from this plot that the distribution of x has a degree of right skewness with the majority of observations of Age being between 7 and 15 years.

2

```r
set.seed(0808)
abalone_split <- initial_split(abalone, prop = 0.80,
                               strata = Age)
abalone_train <- training(abalone_split)
abalone_test <- testing(abalone_split)
```

3

```r
abalone_recipe <- recipe( Age ~., data = subset(abalone_train, select = -rings))

abalone_recipe%>%
  step_scale()
```

```
## Recipe
##
## Inputs:
##
##       role #variables
##    outcome          1
##  predictor          8
##
## Operations:
##
## Scaling for <none>
```

```r
abalone_recipe%>%
  step_center()
```

```
## Recipe
##
## Inputs:
##
##       role #variables
##    outcome          1
##  predictor          8
##
## Operations:
##
## Centering for <none>
```

```r
#
# abalone_recipe%>%
#   step_normalize(all_numeric_predictors( ))

abalone_recipe%>%
  step_interact(terms = type~shucked_weight)
```

```
## Recipe
##
## Inputs:
##
##       role #variables
##    outcome          1
##  predictor          8
##
## Operations:
##
## Interactions with type, shucked_weight
```

```r
abalone_recipe%>%
   step_interact(terms = longest_shell~diameter)
```

```
## Recipe
##
## Inputs:
##
##       role #variables
##    outcome          1
##  predictor          8
##
## Operations:
##
## Interactions with longest_shell, diameter
```

```r
abalone_recipe%>%
   step_interact(terms = shucked_weight~shell_weight)
```

```
## Recipe
##
## Inputs:
##
##       role #variables
##    outcome          1
##  predictor          8
##
## Operations:
##
## Interactions with shucked_weight, shell_weight
```

```r
abalone_recipe%>%
  step_dummy(all_nominal_predictors())
```

```
## Recipe
##
## Inputs:
##
##       role #variables
##    outcome          1
##  predictor          8
##
## Operations:
##
## Dummy variables from all_nominal_predictors()
```

The reason that we do not want to use rings to predict age is because there is colinearity between the two of these variables. As the number of rings increases, the age will also increase at a constant rate so there is no use in using rings in our formula since we already know the exact relationship between the two.

4

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

5

```r
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: 10 x 5
##    term           estimate std.error statistic   p.value
##    <chr>             <dbl>     <dbl>     <dbl>     <dbl>
##  1 (Intercept)      5.40      0.324    16.7    5.46e- 60
##  2 typeI           -0.823     0.113    -7.29   3.89e- 13
##  3 typeM            0.0269    0.0920    0.293  7.70e-  1
##  4 longest_shell    0.186     2.07      0.0901 9.28e-  1
##  5 diameter        10.8       2.54      4.26   2.13e-  5
##  6 height           9.49      1.60      5.95   2.94e-  9
##  7 whole_weight     9.28      0.796    11.7    8.84e- 31
##  8 shucked_weight -20.8       0.919   -22.7    7.41e-106
##  9 viscera_weight -10.2       1.41     -7.21   6.71e- 13
## 10 shell_weight     8.67      1.23      7.07   1.85e- 12
```

6

```r
predict(lm_fit, new_data = data.frame(type = 'F', longest_shell = 0.5, diameter = 0.1, height = 0.3, whe
```

```
## # A tibble: 1 x 1
##   .pred
##   <dbl>
## 1  14.0
```

7

```r
abalone_train_res <- predict(lm_fit, new_data = abalone_train %>% select(-Age))
abalone_train_res <- bind_cols(abalone_train_res, abalone_train %>% select(Age))
```

```r
rmse(abalone_train_res, truth = Age, estimate = .pred)
```

```
## # A tibble: 1 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rmse    standard        2.17
```

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

```
## # A tibble: 3 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rmse    standard       2.17
## 2 rsq     standard       0.539
## 3 mae     standard       1.57
```