Homework 2

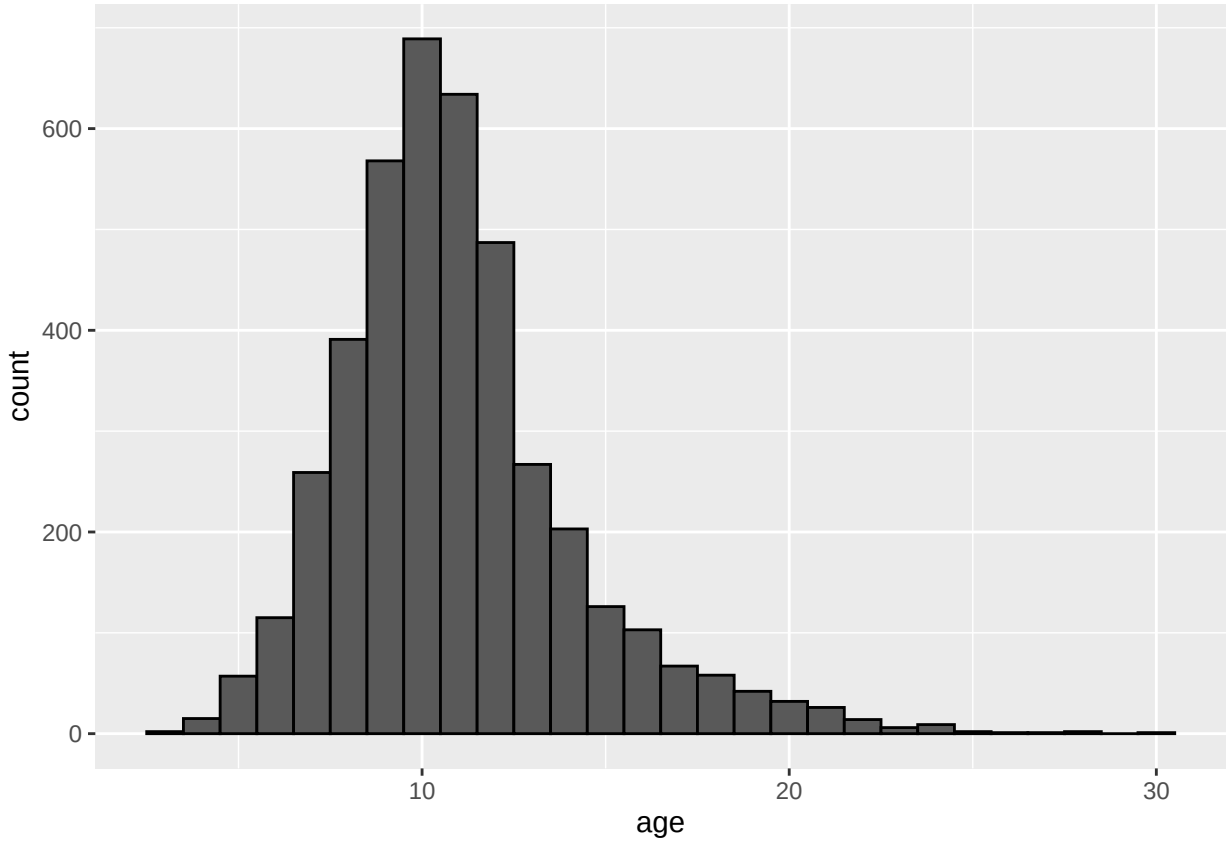
Linear Regression

```
library(tidyverse)
library(tidymodels)
```

```
abalone <- read_csv("homework-2/data/abalone.csv", show_col_types = FALSE)
```

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

ggplot(abalone, aes(x = age)) +
  geom_histogram(color = "black", binwidth = 1)
```



Question 1:

```
mean(abalone$age)
```

```
## [1] 11.43368
```

```
median(abalone$age)
```

```
## [1] 10.5
```

```
var(abalone$age)
```

```
## [1] 10.39527
```

```
range(abalone$age)
```

```
## [1]  2.5 30.5
```

The variable age follows a poisson distribution with a mean of 11.43 and a median of 10.5, and a variance of 10.4 within a range of 2.5 to 30.5.

```
set.seed(0714)

#split the data
abalone_split <- initial_split(abalone, prop = 0.80,
                               strata = age)
#create train and test set
abalone_train <- training(abalone_split)
abalone_test <- testing(abalone_split)
```

Question 2:

Question 3: For this recipe we shouldn't include the variable rings, because both the variables are highly correlated, because rings was used to create the variable age. Given age is created by adding 1.5 to rings, as observations for rings go down, so do the observations for age.

```
#create recipe with training data
abalone_train_recipe <-
  recipe(age ~ type + longest_shell + diameter +
           height + whole_weight + shucked_weight +
           viscera_weight + shell_weight,
         data = abalone_train) %>%
  #create dummy code
  step_dummy(all_nominal_predictors()) %>%
  #create interactions
  step_interact(~ starts_with("type"):shucked_weight) %>%
  step_interact(~ longest_shell:diameter) %>%
  step_interact(~ shucked_weight:shell_weight) %>%
  #center and scale all predictors
  step_normalize(all_predictors())
```

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

Question 4:

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

lm_fit <- fit(lm_wflow, abalone_train)

abalone_lm <- lm_fit %>%
  extract_fit_parsnip() %>%
  tidy()

abalone_lm
```

Question 5:

```
## # A tibble: 14 x 5
##    term                          estimate std.error statistic  p.value
##    <chr>                            <dbl>     <dbl>     <dbl>    <dbl>
##  1 (Intercept)                     11.4       0.0370   309.    0
##  2 longest_shell                    0.324     0.285      1.14  2.56e- 1
##  3 diameter                         1.85      0.315      5.89  4.32e- 9
##  4 height                           0.537     0.0969     5.54  3.33e- 8
##  5 whole_weight                     4.34      0.398     10.9   3.01e-27
##  6 shucked_weight                  -4.11      0.250    -16.4   1.84e-58
##  7 viscera_weight                  -0.812     0.159     -5.12  3.26e- 7
##  8 shell_weight                     1.53      0.221      6.93  5.03e-12
##  9 type_I                          -0.964     0.114     -8.44  4.69e-17
## 10 type_M                          -0.266     0.103     -2.59  9.60e- 3
## 11 type_I_x_shucked_weight          0.514     0.0856     6.01  2.12e- 9
## 12 type_M_x_shucked_weight          0.279     0.107      2.60  9.42e- 3
## 13 longest_shell_x_diameter        -2.32      0.406     -5.70  1.30e- 8
## 14 shucked_weight_x_shell_weight   -0.0652    0.203     -0.321 7.48e- 1
```

```
df <- data.frame(longest_shell = 0.50, diameter = 0.10,
                 height = 0.30, whole_weight = 4,
                 shucked_weight = 1, viscera_weight = 2,
                 shell_weight = 1, type = "F")
predict(lm_fit, new_data = df)
```

Question 6:

```
## # A tibble: 1 x 1
##   .pred
##   <dbl>
## 1  23.1
```

```
#install.packages("yardstick")
library(yardstick)

#create metric set
abalone_metrics <- metric_set(rmse, rsq, mae)

#use predict()
abalone_train_res <- predict(lm_fit, new_data = abalone_train %>%
                               select(-age))

#use bind_cols()
abalone_train_res <- bind_cols(abalone_train_res, abalone_train %>%
                                 select(age))

abalone_train_res
```

Question 7:

```
## # A tibble: 3,340 x 2
##    .pred   age
##    <dbl> <dbl>
##  1  9.52   8.5
##  2  8.03   8.5
##  3  9.59   8.5
##  4 10.3    8.5
##  5 10.0    9.5
##  6 10.8    9.5
##  7  6.22   6.5
##  8  5.88   6.5
##  9  8.58   8.5
## 10 11.8    8.5
## # ... with 3,330 more rows
```

```
#apply metric set to tibble
abalone_metrics(abalone_train_res, truth = age, estimate = .pred)
```

```
## # A tibble: 3 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rmse    standard       2.13
## 2 rsq     standard       0.556
## 3 mae     standard       1.53
```

The R^2 in this model is 0.556, meaning that 55.6% of the training data set fit the model.