

# 131 Homework1

Scott Shang (8458655)

April 10, 2022

## Question1

```
library("tidyverse")
```

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

```
## v ggplot2 3.3.5    v purrr  0.3.4
## v tibble  3.1.6    v dplyr  1.0.8
## v tidyr   1.2.0    v stringr 1.4.0
## v readr   2.1.2    v forcats 0.5.1
```

```
## -- Conflicts ----- tidyverse_conflicts() --
```

```
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library("tidymodels")
```

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

```
## v broom      0.7.12    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()
## * Search for functions across packages at https://www.tidymodels.org/find/
```

```
abalone=read_csv('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.
```

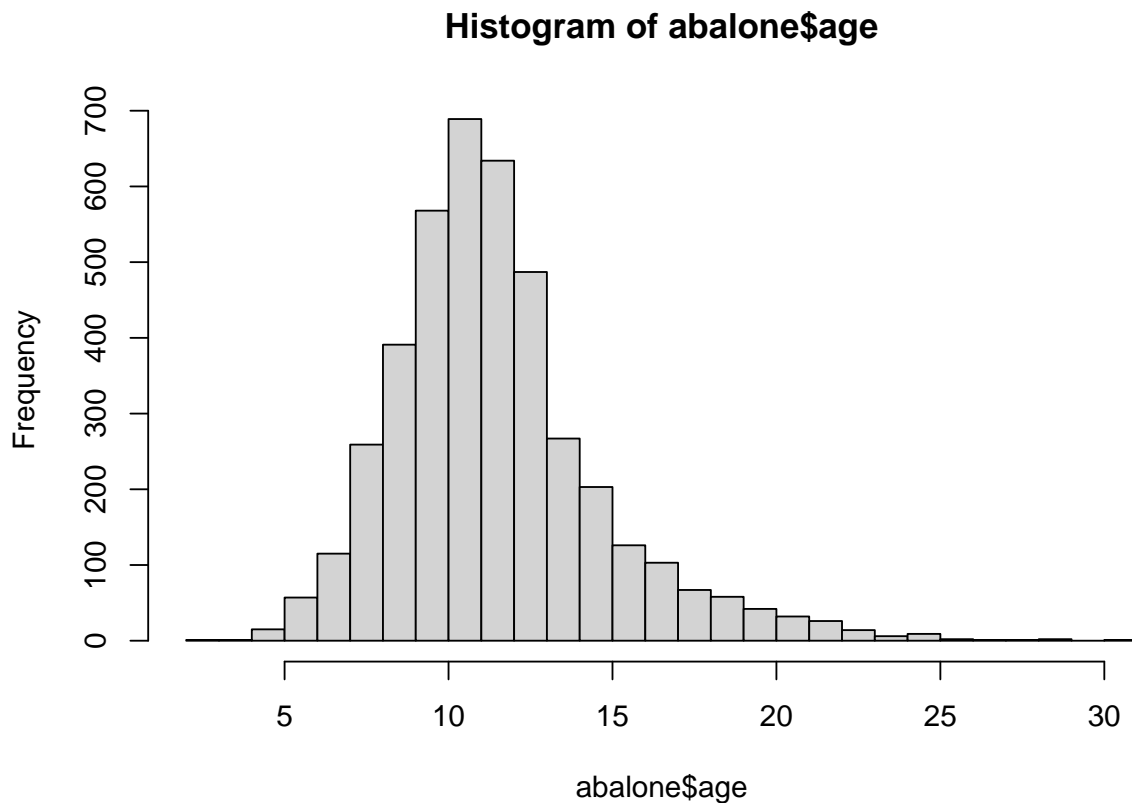
```
head(abalone)
```

```
## # A tibble: 6 x 9
##   type longest_shell diameter height whole_weight shucked_weight viscera_weight
##   <chr>         <dbl>    <dbl> <dbl>         <dbl>         <dbl>         <dbl>
## 1 M           0.455    0.365 0.095         0.514         0.224         0.101
## 2 M           0.35     0.265 0.09          0.226         0.0995        0.0485
## 3 F           0.53     0.42  0.135         0.677         0.256         0.142
## 4 M           0.44     0.365 0.125         0.516         0.216         0.114
## 5 I           0.33     0.255 0.08          0.205         0.0895        0.0395
## 6 I           0.425    0.3    0.095         0.352         0.141         0.0775
## # ... with 2 more variables: shell_weight <dbl>, rings <dbl>
```

```
abalone$age=abalone$rings+1.5
head(abalone)
```

```
## # A tibble: 6 x 10
##   type longest_shell diameter height whole_weight shucked_weight viscera_weight
##   <chr>         <dbl>    <dbl> <dbl>         <dbl>         <dbl>         <dbl>
## 1 M           0.455    0.365 0.095         0.514         0.224         0.101
## 2 M           0.35     0.265 0.09          0.226         0.0995        0.0485
## 3 F           0.53     0.42  0.135         0.677         0.256         0.142
## 4 M           0.44     0.365 0.125         0.516         0.216         0.114
## 5 I           0.33     0.255 0.08          0.205         0.0895        0.0395
## 6 I           0.425    0.3    0.095         0.352         0.141         0.0775
## # ... with 3 more variables: shell_weight <dbl>, rings <dbl>, age <dbl>
```

```
hist(abalone$age,breaks=20)
```



```
summary(abalone$age)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       2.5     9.5    10.5    11.4    12.5    30.5
```

From the histogram above, we notice that the distribution of age is partially normal distributed, but slightly skewed to the right, with minimum 2.5, maximum 30.5, and mean 11.4.

Question2

```
set.seed(1234)
ab_split=initial_split(abalone,prop=0.80,strata=age)
ab_train=training(ab_split)
ab_test=testing(ab_split)
```

Question3

```
ab_recipe=recipe(age~type+longest_shell+diameter+height+whole_weight+shucked_weight+viscera_weight+shell_weight)
  step_dummy(all_nominal_predictors())%>%
  step_interact(terms=~type_I:shucked_weight+type_M:shucked_weight+longest_shell:diameter+shucked_weight:diameter)
  step_normalize(all_predictors())
```

We shouldn't use rings to predict age because age is depends on age, so we won't have the relation between age and other predictors.

#### Question4

```
lm_model=linear_reg()%>%  
  set_engine("lm")
```

#### Question5

```
lm_wflow=workflow() %>%  
  add_model(lm_model) %>%  
  add_recipe(ab_recipe)
```

#### Question6

```
lm_fit=fit(lm_wflow,ab_train)  
  
ab_test=data.frame("type"="F","longest_shell" = 0.50, "diameter" = 0.10, "height" = 0.30, "whole_weight"  
  )  
  
predict(lm_fit,new_data=ab_test)  
  
## # A tibble: 1 x 1  
##   .pred  
##   <dbl>  
## 1  23.5
```

#### Question7

```
ab_metric=metric_set(rsq,rmse,mae)  
ab_pred=predict(lm_fit,new_data=ab_train%>%select(-age))  
ab_pred=bind_cols(ab_pred,ab_train%>%select(age))  
ab_pred
```

```
## # A tibble: 3,340 x 2  
##   .pred age  
##   <dbl> <dbl>  
## 1  9.33  9.5  
## 2  9.83  8.5  
## 3 10.1   9.5  
## 4  6.32  6.5  
## 5  5.82  6.5  
## 6  5.95  5.5  
## 7  8.56  8.5  
## 8  7.72  7.5  
## 9 10.2   8.5  
## 10 12.7  9.5  
## # ... with 3,330 more rows
```

```
ab_metric(ab_pred,truth=age,estimate=.pred)
```

```
## # A tibble: 3 x 3  
##   .metric .estimator .estimate  
##   <chr>   <chr>         <dbl>  
## 1 rsq     standard      0.558  
## 2 rmse    standard      2.15  
## 3 mae     standard      1.55
```