Week 8 - 9

Kimberly Cable

05-14-2022

Assignment 6

```
##    earn   height    sex ed age  race
## 1 50000 74.42444   male 16  45 white
## 2 60000 65.53754 female 16  58 white
## 3 30000 63.62920 female 16  29 white
## 4 50000 63.10856 female 16  91 other
## 5 51000 63.40248 female 17  39 white
## 6  9000 64.39951 female 15  26 white
```

Fit a linear model using the `age` variable as the predictor and `earn` as the outcome

lm(outcome ~ predictors, data = dataframe)

```
age_lm <-  lm(earn ~ age, data = heights_df)
```

View the summary of your model using `summary()`

```
summary(age_lm)
```

```
##
## Call:
## lm(formula = earn ~ age, data = heights_df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -25098 -12622  -3667   6883 177579
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 19041.53    1571.26  12.119  < 2e-16 ***
## age            99.41      35.46   2.804  0.00514 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19420 on 1190 degrees of freedom
## Multiple R-squared:  0.006561,   Adjusted R-squared:  0.005727
## F-statistic:  7.86 on 1 and 1190 DF,  p-value: 0.005137
```

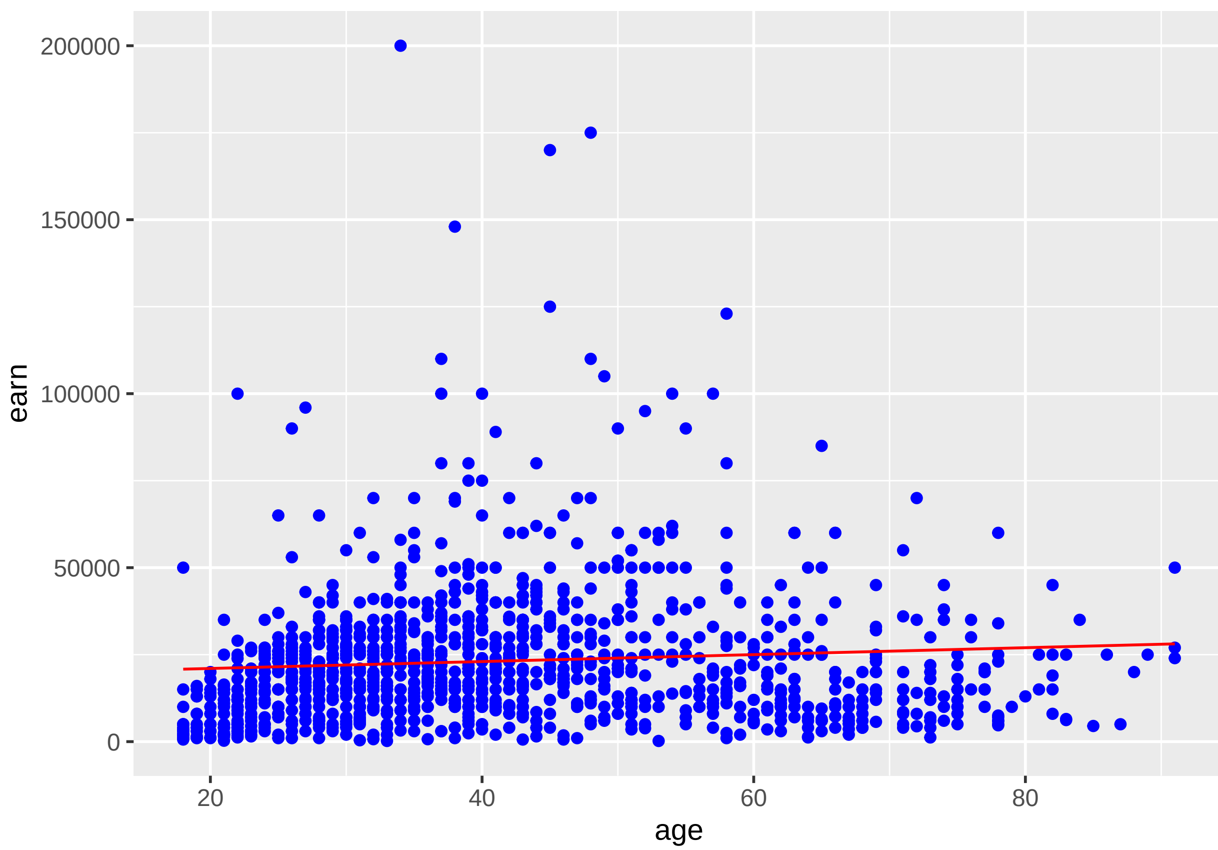
Creating predictions using `predict()`

```
age_predict_df <- data.frame(earn = predict(age_lm, heights_df), age = heights_df$age)
head(age_predict_df)
```

```
##       earn age
## 1 23514.79  45
## 2 24807.06  58
## 3 21924.29  29
## 4 28087.45  91
## 5 22918.35  39
## 6 21626.08  26
```

Plot the predictions against the original data

```
ggplot(data = heights_df, aes(y = earn, x = age)) +
  geom_point(color = 'blue') +
  geom_line(color='red',data = age_predict_df, aes(y = earn, x = age))
```




```
mean_earn <- mean(heights_df$earn)
mean_earn
```

```
## [1] 23154.77
```

Corrected Sum of Squares Total

```
sst <- sum((mean_earn - heights_df$earn)^2)
sst
```

```
## [1] 451591883937
```

Corrected Sum of Squares for Model

```
ssm <- sum((mean_earn - age_predict_df$earn)^2)
ssm
```

```
## [1] 2963111900
```

Residuals

```
residuals <- heights_df$earn - age_predict_df$earn
head(residuals)
```

```
## [1]  26485.214  35192.939   8075.707  21912.549  28081.649 -12626.076
```

Sum of Squares for Error

```
sse <- sum(residuals^2)
sse
```

```
## [1] 448628772037
```

R Squared R^2 = SSM/SST

```
r_squared <- ssm / sst
r_squared
```

```
## [1] 0.006561482
```

Number of observations

```
n <- nrow(heights_df)
n
```

```
## [1] 1192
```

Number of regression parameters

```r
p <- 2
```

Corrected Degrees of Freedom for Model (p-1)

```r
dfm <- p - 1
```

Degrees of Freedom for Error (n-p)

```r
dfe <- n - p
```

Corrected Degrees of Freedom Total: DFT = n - 1

```r
dft <- n - 1
```

Mean of Squares for Model: MSM = SSM / DFM

```r
msm <- ssm / dfm
msm
```

```
## [1] 2963111900
```

Mean of Squares for Error: MSE = SSE / DFE

```r
mse <- sse / dfe
mse
```

```
## [1] 376998968
```

Mean of Squares Total: MST = SST / DFT

```r
mst <- sst / dft
mst
```

```
## [1] 379170348
```

F Statistic F = MSM/MSE

```r
f_score <- msm / mse
f_score
```

```
## [1] 7.859735
```

Adjusted R Squared R2 = 1 - (1 - R2)(n - 1) / (n - p)

```r
adjusted_r_squared <- 1 - ((1 - r_squared) * dft) / dfe
adjusted_r_squared
```

```
## [1] 0.005726659
```

Calculate the p-value from the F distribution

```r
p_value <- pf(f_score, dfm, dft, lower.tail = F)
p_value
```

```
## [1] 0.005136826
```

Assignment 7

Load the `data/r4ds/heights.csv` to

```
##    earn   height    sex ed age  race
## 1 50000 74.42444   male 16  45 white
## 2 60000 65.53754 female 16  58 white
## 3 30000 63.62920 female 16  29 white
## 4 50000 63.10856 female 16  91 other
## 5 51000 63.40248 female 17  39 white
## 6  9000 64.39951 female 15  26 white
```

Fit a linear model

```r
earn_lm <-  lm(earn ~ height + sex + ed + age + race, data = heights_df)
earn_lm
```

```
##
## Call:
## lm(formula = earn ~ height + sex + ed + age + race, data = heights_df)
##
## Coefficients:
##  (Intercept)        height       sexmale            ed           age
##     -41478.5         202.5       10325.6        2768.4         178.3
## racehispanic     raceother     racewhite
##      -1414.3         371.0        2432.5
```

View the summary of your model

```r
summary(earn_lm)
```

```
##
## Call:
## lm(formula = earn ~ height + sex + ed + age + race, data = heights_df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -39423  -9827  -2208   6157 158723
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -41478.4    12409.4  -3.342 0.000856 ***
## height          202.5      185.6   1.091 0.275420
## sexmale       10325.6     1424.5   7.249 7.57e-13 ***
## ed             2768.4      209.9  13.190  < 2e-16 ***
## age             178.3       32.2   5.537 3.78e-08 ***
## racehispanic  -1414.3     2685.2  -0.527 0.598507
## raceother       371.0     3837.0   0.097 0.922983
## racewhite      2432.5     1723.9   1.411 0.158489
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17250 on 1184 degrees of freedom
## Multiple R-squared:  0.2199, Adjusted R-squared:  0.2153
## F-statistic: 47.68 on 7 and 1184 DF,  p-value: < 2.2e-16
```

```r
predicted_df <- data.frame(
  earn = predict(earn_lm, heights_df),
  ed = heights_df$age,
  race = heights_df$race,
  height = heights_df$height,
  age = heights_df$age,
  sex = heights_df$sex
  )
head(predicted_df)
```

```
##       earn ed  race   height age    sex
## 1 38666.11 45 white 74.42444  45   male
## 2 28859.09 58 white 65.53754  58 female
## 3 23301.90 29 white 63.62920  29 female
## 4 32189.84 91 other 63.10856  91 female
## 5 27807.39 39 white 63.40248  39 female
## 6 20154.60 26 white 64.39951  26 female
```

Compute deviation (i.e. residuals)

```r
mean_earn <- mean(heights_df$earn)
mean_earn
```

```
## [1] 23154.77
```

Corrected Sum of Squares Total

```r
sst <- sum((mean_earn - heights_df$earn)^2)
sst
```

```
## [1] 451591883937
```

Corrected Sum of Squares for Model

```r
ssm <- sum((mean_earn - predicted_df$earn)^2)
ssm
```

```
## [1] 99302918657
```

Residuals

```r
residuals <- heights_df$earn - predicted_df$earn
head(residuals)
```

```
## [1]  11333.891  31140.911   6698.099  17810.165  23192.610 -11154.599
```

Sum of Squares for Error

```r
sse <- sum(residuals^2)
sse
```

```
## [1] 3.52289e+11
```

R Squared

```r
r_squared <- ssm / sst
r_squared
```

```
## [1] 0.2198953
```

Number of observations

```r
n <- nrow(heights_df)
n
```

```
## [1] 1192
```

Number of regression parameters

```r
p <- 8
```

Corrected Degrees of Freedom for Model

```r
dfm <- p - 1
dfm
```

```
## [1] 7
```

Degrees of Freedom for Error

```r
dfe <- n - p
dfe
```

```
## [1] 1184
```

Corrected Degrees of Freedom Total: DFT = n - 1

```r
dft <- n - 1
dft
```

```
## [1] 1191
```

Mean of Squares for Model: MSM = SSM / DFM

```r
msm <- ssm / dfm
msm
```

```
## [1] 14186131237
```

Mean of Squares for Error: MSE = SSE / DFE

```
mse <- sse / dfe
mse
```

```
## [1] 297541356
```

Mean of Squares Total: MST = SST / DFT

```
mst <- sst / dft
mst
```

```
## [1] 379170348
```

F Statistic

```
f_score <- msm / mse
f_score
```

```
## [1] 47.67785
```

Adjusted R Squared R2 = 1 - (1 - R2)(n - 1) / (n - p)

```
adjusted_r_squared <- 1 - ((1 - r_squared) * dft) / dfe
adjusted_r_squared
```

```
## [1] 0.2152832
```

Housing Data

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

## Loading required package: lattice

## Loading required package: survival
```

```
## Loading required package: Formula

##
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:dplyr':
##
##     src, summarize

## The following objects are masked from 'package:base':
##
##     format.pval, units
```

Data for this assignment is focused on real estate transactions recorded from 1964 to 2016 and can be found in Housing.xlsx. Using your skills in statistical correlation, multiple regression, and R programming, you are interested in the following variables: Sale Price and several other possible predictors.

If you worked with the Housing dataset in previous week – you are in luck, you likely have already found any issues in the dataset and made the necessary transformations. If not, you will want to take some time looking at the data with all your new skills and identifying if you have any clean up that needs to happen.

```
## # A tibble: 4 x 24
##   `Sale Date`         `Sale Price` sale_reason sale_instrument sale_warning
##   <dttm>                     <dbl>       <dbl>           <dbl> <chr>
## 1 2006-01-03 00:00:00       698000           1               3 <NA>
## 2 2006-01-03 00:00:00       649990           1               3 <NA>
## 3 2006-01-03 00:00:00       572500           1               3 <NA>
## 4 2006-01-03 00:00:00       420000           1               3 <NA>
## # ... with 19 more variables: sitetype <chr>, addr_full <chr>, zip5 <dbl>,
## #   ctyname <chr>, postalctyn <chr>, lon <dbl>, lat <dbl>,
## #   building_grade <dbl>, square_feet_total_living <dbl>, bedrooms <dbl>,
## #   bath_full_count <dbl>, bath_half_count <dbl>, bath_3qtr_count <dbl>,
## #   year_built <dbl>, year_renovated <dbl>, current_zoning <chr>,
## #   sq_ft_lot <dbl>, prop_type <chr>, present_use <dbl>
```

i Explain any transformations or modifications you made to the dataset

```
## # A tibble: 4 x 25
##   Sale_Date           Sale_Price sale_reason sale_instrument sale_warning
##   <dttm>                   <dbl>       <dbl>           <dbl> <chr>
## 1 2006-01-03 00:00:00     698000           1               3 <NA>
## 2 2006-01-03 00:00:00     649990           1               3 <NA>
## 3 2006-01-03 00:00:00     572500           1               3 <NA>
## 4 2006-01-03 00:00:00     420000           1               3 <NA>
## # ... with 20 more variables: sitetype <chr>, addr_full <chr>, zip5 <dbl>,
## #   ctyname <chr>, postalctyn <chr>, lon <dbl>, lat <dbl>,
## #   building_grade <dbl>, square_feet_total_living <dbl>, bedrooms <dbl>,
## #   bath_full_count <dbl>, bath_half_count <dbl>, bath_3qtr_count <dbl>,
## #   year_built <dbl>, year_renovated <dbl>, current_zoning <chr>,
## #   sq_ft_lot <dbl>, prop_type <chr>, present_use <dbl>, total_bath_count <dbl>
```

ii: Create two variables; one that will contain the variables Sale Price and Square Foot of Lot (same variables used from previous assignment on simple regression) and one that will contain Sale Price and several additional predictors of your choice. Explain the basis for your additional predictor selections.

For my additional predictors, I chose zip5, total_bath_count, year_built, square_feet_total_living, and bedroom because they traditionally are used to calculate the price of a house

```
## # A tibble: 5 x 2
##   Sale_Price sq_ft_lot
##        <dbl>     <dbl>
## 1     698000      6635
## 2     649990      5570
## 3     572500      8444
## 4     420000      9600
## 5     369900      7526
```

```
## # A tibble: 5 x 6
##   Sale_Price  zip5 total_bath_count year_built square_feet_total_living bedrooms
##        <dbl> <dbl>            <dbl>      <dbl>                    <dbl>    <dbl>
## 1     698000 98052             2.5        2003                     2810        4
## 2     649990 98052             2.75       2006                     2880        4
## 3     572500 98052             2.25       1987                     2770        4
## 4     420000 98052             1.75       1968                     1620        3
## 5     369900 98052             1.75       1980                     1440        3
```

iii: Execute a summary() function on two variables defined in the previous step to compare the model results. What are the R2 and Adjusted R2 statistics? Explain what these results tell you about the overall model. Did the inclusion of the additional predictors help explain any large variations found in Sale Price?

```
##
## Call:
## lm(formula = Sale_Price ~ sq_ft_lot, data = price_sqft)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2016064  -194842   -63293    91565  3735109
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 6.418e+05  3.800e+03  168.90   <2e-16 ***
## sq_ft_lot   8.510e-01  6.217e-02   13.69   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 401500 on 12863 degrees of freedom
## Multiple R-squared:  0.01435,    Adjusted R-squared:  0.01428
## F-statistic: 187.3 on 1 and 12863 DF,  p-value: < 2.2e-16
```

```
##
##  Pearson's product-moment correlation
##
```

```
## data:  price_sqft$Sale_Price and price_sqft$sq_ft_lot
## t = 13.687, df = 12863, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1027447 0.1368093
## sample estimates:
##       cor
## 0.1198122
```

The linear regression model for Sales Price and Square Feet per Lot is an R^2 = 0.01435 tells us Sq feet per lot only accounts for 1.4% of the variation in the Sales Price.

```
##
## Call:
## lm(formula = Sale_Price ~ zip5 + total_bath_count + year_built,
##     data = price_predictors, subset = +square_feet_total_living +
##         bedrooms)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -884708 -155614  -63512   50750 2606400
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -1.235e+09  1.695e+08  -7.289 3.30e-13 ***
## zip5              1.256e+04  1.729e+03   7.264 3.96e-13 ***
## total_bath_count  9.716e+04  5.316e+03  18.277  < 2e-16 ***
## year_built        2.182e+03  2.247e+02   9.707  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 410600 on 12856 degrees of freedom
##   (5 observations deleted due to missingness)
## Multiple R-squared:  0.05006,    Adjusted R-squared:  0.04984
## F-statistic: 225.8 on 3 and 12856 DF,  p-value: < 2.2e-16
```

The linear regression model for Sales Price, zip5, total_bath_count, year_built, square_feet_total_living, and bedrooms is an R^2 = 0.05 tells us that adding other variables raises the percentage to 5% in the Sales Price.

This tells me that there are a lot more variables that go into the Sales Price of a house.

iv: Considering the parameters of the multiple regression model you have created. What are the standardized betas for each parameter and what do the values indicate?

v: Calculate the confidence intervals for the parameters in your model and explain what the results indicate.

vi: Assess the improvement of the new model compared to your original model (simple regression model) by testing whether this change is significant by performing an analysis of variance.

vii: Perform casewise diagnostics to identify outliers and/or influential cases, storing each function's output in a dataframe assigned to a unique variable name.

viii: Calculate the standardized residuals using the appropriate command, specifying those that are +-2, storing the results of large residuals in a variable you create.

ix: Use the appropriate function to show the sum of large residuals.

x: Which specific variables have large residuals (only cases that evaluate as TRUE)?

xi: Investigate further by calculating the leverage, cooks distance, and covariance rations. Comment on all cases that are problematics.

xii: Perform the necessary calculations to assess the assumption of independence and state if the condition is met or not.

xiii: Perform the necessary calculations to assess the assumption of no multicollinearity and state if the condition is met or not.

xiv: Visually check the assumptions related to the residuals using the plot() and hist() functions. Summarize what each graph is informing you of and if any anomalies are present.

xv: Overall, is this regression model unbiased? If an unbiased regression model, what does this tell us about the sample vs. the entire population model?