~~~
---
title: "RMarkdown Week 8 & 9"
author: "Dipika Sharma"
date: May 16, 2021
output:

  pdf_document: default
  html_document: default
  word_document: default
bibliography: bibliography.bib
---

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML,
PDF, and MS Word documents. For more details on using R Markdown see
<http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content
as well as the output of any embedded R code chunks within the document. You can embed an
R code chunk like this:

## Add Citations

* R for Everyone [@lander2014r]
* Discovering Statistics Using R [@field2012discovering]

## Assignment 06

Set the working directory to the root of your DSC 520 directory
Load the `data/r4ds/heights.csv` to

```{r include=TRUE}

## Set the working directory to the root of your DSC 520 directory
setwd("/Users/dipikasharma/R_Projects/DSC520")

## Load the `data/r4ds/heights.csv` to
heights_df <- read.csv("data/r4ds/heights.csv")

```

Fit a linear model using the `age` variable as the predictor and `earn` as the outcome.
View the summary of your model using `summary()`
~~~

```{r include=TRUE}
## Load the ggplot2 library
library(ggplot2)

## Fit a linear model using the `age` variable as the predictor and `earn` as the outcome
age_lm <- lm(earn~age, data = heights_df)

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

Creating predictions using `predict()`

```{r include=TRUE}
## Creating predictions using `predict()`
age_predict_df <- data.frame(earn = predict(age_lm, heights_df), age = heights_df$age)
#age_predict_df
```

Plot the predictions against the original data

```{r include=TRUE}
##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))

```

Compute deviation (i.e. residuals)

```{r include=TRUE}
mean_earn <- mean(heights_df$earn)
mean_earn
```

Corrected Sum of Squares Total

```{r include=TRUE}
sst <- sum((mean_earn - heights_df$earn)^2)
sst
```

Corrected Sum of Squares for Model

~~~
```{r include=TRUE}
ssm <- sum((mean_earn - age_predict_df$earn)^2)
ssm
```
~~~

Residuals

~~~
```{r include=TRUE}
residuals <- heights_df$earn - age_predict_df$earn
residuals
```
~~~

Sum of Squares for Error

~~~
```{r include=TRUE}
sse <- sum(residuals^2)
sse
```
~~~

R Squared

~~~
```{r include=TRUE}
#R Squared R^2 = SSM\SST
r_squared <- (ssm/sst)
r_squared
```
~~~

Number of observations

~~~
```{r include=TRUE}
n <- nrow(heights_df)
n
```
~~~

Number of regression parameters

~~~
```{r include=TRUE}
p <- 2
p
```
~~~

Corrected Degrees of Freedom for Model

~~~
```{r include=TRUE}
#Corrected Degrees of Freedom for Model (p-1)
dfm <- (p-1)
dfm
```
~~~

Degrees of Freedom for Error

~~~
```{r include=TRUE}
#Degrees of Freedom for Error (n-p)
dfe <- (n-p)
dfe
```
~~~

Corrected Degrees of Freedom Total

~~~
```{r include=TRUE}
#Corrected Degrees of Freedom Total:   DFT = n - 1
dft <- (n-1)
dft
```
~~~

Mean of Squares for Model

~~~
```{r include=TRUE}
#Mean of Squares for Model:   MSM = SSM / DFM
msm <- (ssm/dfm)
msm
```
~~~

Mean of Squares for Error

~~~
```{r include=TRUE}
#Mean of Squares for Error:   MSE = SSE / DFE
mse <- (sse/dfe)
mse
```
~~~

Mean of Squares Total

~~~
```{r include=TRUE}
#Mean of Squares Total:   MST = SST / DFT
mst <- (sst/dft)
~~~

~~~
mst
```

F Statistic

```{r include=TRUE}
#F Statistic F = MSM/MSE
f_score <- (msm/mse)
f_score
```

Adjusted R Squared R2

```{r include=TRUE}
#Adjusted R Squared R2 = 1 - (1 - R2)(n - 1) / (n - p)
adjusted_r_squared <- (1 - (1 - r_squared)*(n - 1) / (n - p))
adjusted_r_squared
```

Calculate the pvalue from the F distribution

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



## Assignment 07

Set the working directory to the root of your DSC 520 directory
Load the `data/r4ds/heights.csv` to

```{r include=TRUE}
setwd("/Users/dipikasharma/R_Projects/DSC520")

## Load the `data/r4ds/heights.csv` to
heights_df <- read.csv("data/r4ds/heights.csv")

```

Fit a linear model

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

~~~
earn_lm
```
~~~

View the summary of your model

~~~
```{r include=TRUE}
summary(earn_lm)
```
~~~

Predicted Model

~~~
```{r include=TRUE}
predicted_df <- data.frame(
  earn = predict(earn_lm, heights_df),
  ed=heights_df$ed, race=heights_df$race, height=heights_df$height,
  age=heights_df$age, sex=heights_df$sex
)
#predicted_df
```
~~~

Compute deviation (i.e. residuals)

~~~
```{r include=TRUE}
mean_earn <- mean(heights_df$earn)
mean_earn
```
~~~

Corrected Sum of Squares Total

~~~
```{r include=TRUE}
sst <- sum((mean_earn - heights_df$earn)^2)
sst
```
~~~

Corrected Sum of Squares for Model

~~~
```{r include=TRUE}
ssm <- sum((mean_earn - predicted_df$earn)^2)
ssm
```
~~~

Residuals

~~~
```{r include=TRUE}
residuals <- (heights_df$earn - predicted_df$earn)
~~~

~~~
residuals
```
~~~

Sum of Squares for Error

~~~
```{r include=TRUE}
sse <- sum(residuals^2)
sse
```
~~~

R Squared

~~~
```{r include=TRUE}
r_squared <- (ssm/sst)
r_squared
```
~~~

Number of observations

~~~
```{r include=TRUE}
n <- nrow(heights_df)
n
```
~~~

Number of regression paramaters

~~~
```{r include=TRUE}
p <- 8
p
```
~~~

Corrected Degrees of Freedom for Model

~~~
```{r include=TRUE}
dfm <- (p-1)
dfm
```
~~~

Degrees of Freedom for Error

~~~
```{r include=TRUE}
dfe <- (n-p)
dfe
```
~~~

Corrected Degrees of Freedom Total

~~~
```{r include=TRUE}
#Corrected Degrees of Freedom Total:   DFT = n - 1
dft <- (n-1)
dft
```
~~~

Mean of Squares for Model

~~~
```{r include=TRUE}
# Mean of Squares for Model:   MSM = SSM / DFM
msm <- (ssm/dfm)
msm
```
~~~

Mean of Squares for Error

~~~
```{r include=TRUE}
# Mean of Squares for Error:   MSE = SSE / DFE
mse <- (sse/dfe)
mse
```
~~~

Mean of Squares Total

~~~
```{r include=TRUE}
# Mean of Squares Total:   MST = SST / DFT
mst <- (sst/dft)
mst
```
~~~

F Statistic

~~~
```{r include=TRUE}
f_score <- (msm/mse)
f_score
```
~~~

Adjusted R Squared R2

~~~
```{r include=TRUE}
# Adjusted R Squared R2 = 1 - (1 - R2)(n - 1) / (n - p)
~~~

~~~
adjusted_r_squared <- (1 - (1 - r_squared)*(n - 1) / (n - p))
adjusted_r_squared
```

## Housing Data

```{r include=TRUE}
## Set the working directory to the root of your DSC 520 directory
setwd("/Users/dipikasharma/R_Projects/DSC520")
library(readxl)
housing_df <- read_excel("data/week-7-housing.xlsx")
housing_df
#unique(housing_df$ctyname)
```



## 3a. i. 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.

```{r include=TRUE}
housing_df$ctyname <- ifelse(is.na(housing_df$ctyname), housing_df$postalctyn,
housing_df$ctyname)
#housing_df
#unique(housing_df$ctyname)

housing_df <- subset(housing_df, select = -sale_warning)
housing_df
```

##  Complete the following:
##  Explain any transformations or modifications you made to the dataset

Answer: After reading the data I found that sale_warning and ctyname has null values.
sale_warning has only 18% of the rows with actual data in housing dataset. rest of the 82% of
rows are showing NULL. Adding zero instead of null value will not solve the problem and can
lead to wrong prediction. Better way is to remove this column from dataset in order to avoid
miscalculation. ctyname has 50% of rows with actual values and rest 50% with NULL values, but
we have another column postalctyn, we can either remove this column or can replace NULL
values in ctyname column with postalctyn value. I thought of removing ctyname first but then
realized that we have different values in ctyname Column and have same value for all address
in postalctyn. so i think it is better idea to just relace null values in ctyname with postalctyn.
~~~

## 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.

```{r include=TRUE}
SP_FL_lm <- lm(`Sale Price`~sq_ft_lot, data = housing_df)
SP_FL_lm
SP_other_lm <- lm(`Sale Price`~building_grade+square_feet_total_living , data = housing_df)
SP_other_lm
```

## Explain the basis for your additional predictor selections.

Answer: I am using the building_grade and square_feet_total_living as the predictor variable because i think that the change in building grade or square_feet_total_living will affect the sale price of the house.

# iii Execute a summary() function on two variables defined in the previous step to compare the model results.

```{r include=TRUE}
summary(SP_FL_lm)
summary(SP_other_lm)
```

## 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?

Answer: R2 is used to determine to what extent the variance of one variable explain the variance of the other variable. Adjusted R2 is the modified version of R2, it is adjusted for number of predictor variable. if the new term improves the model from what it is expected then adjusted R2 increases and decreases when predictor variable improves the model less then what it is expected.

By compary the R2 and adjusted R2 of both the variable I found that the relationship between model and multiple independent variables (building grade and square_feet_total_living) is better than the relationship between model and independent variable square ft lot. Looking at the R2 and adjusted R2 of sale price and predictor variables building grade and square_feet_total_living we can understand that the increase in these predictor variable will show the change in dependent variable 'sale price'

yes, by adding the predictor variable building grade and square_feet_total_living, we found 20% variation in 'sale price'

## 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?

```{r include=TRUE}
library(lm.beta)
lmbeta_SP <- lm.beta(SP_FL_lm)
lmbeta_sp_other <- lm.beta(SP_other_lm)
lmbeta_SP
lmbeta_sp_other
```

Answer: Standard coefficient is uded to find out which of the independent variable in multiple regression model have greater effect on the dependent variables. By looking at standard coeffcient of all variable we can see that square_feet_total_living has most effect on dependent variable 'Sale Price' compare to the others independent variables.

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

```{r include=TRUE}
summary(SP_FL_lm)
summary(SP_other_lm)

confint(SP_other_lm, 'sq_ft_lot', level=0.95)

confint(SP_other_lm, level=0.95)
```

Answer: Looking at the parameter values we can say that the confidence interval of building grade (35164.8 to 52185.6) signifies the range in which the true population parameter lies at a 95% level of confidence And the confidence interval of square_feet_total_living (140 to 159) signifies the range in which the true population parameter lies at a 95% level of confidence.

## 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.

```{r include=TRUE}
summary(SP_FL_lm)

summary(SP_other_lm)
library(car)
compareCoefs(SP_FL_lm, SP_other_lm)
```

~~~
```
~~~

Answer: When I compared the R2 and Adjusted R2 of both model simple regression model and multiple regression model where I am using two independent variables building grades and square feet living total, I found that the values of multiple regression model is higher and it is expected to always choose the model which has higher Adjusted R2. Adjusted R2 increases only if new term improves the model more than would be expected by chance.

```{r include=TRUE}
anova(SP_FL_lm, SP_other_lm)
```

As we can see from the p-value, both models are slightly different. but since the RSS df is less in model 2 so we can say model 2 is better.

## 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.

```{r include=TRUE}
HousingOrg <-
  lm(`Sale Price`~square_feet_total_living+bath_3qtr_count+bath_full_count+bath_half_count+bedrooms+building_grade+lat+lon+present_use+sale_instrument+sale_reason+sq_ft_lot+year_built+year_renovated+zip5,
     data=housing_df)
summary(HousingOrg)
library(car)
outlierTest(HousingOrg)
outlierTest(SP_FL_lm)
outlierTest(SP_other_lm)
```

The original data had line 6430 and with the adjusted model we have row 4649 is listed. so the updated data frames with out the outlier rows look like below:

```{r include=TRUE}
HousingOrg_out <- housing_df[-c(11992,6430,6438,6437,6431,6436,6441,6432,6442,6433,4649),]
str(HousingOrg_out)
```

Creating above 2 models with Housing data set without the outliers

```{r include=TRUE}
```

~~~~
model3 <- lm(`Sale Price`~sq_ft_lot, data = HousingOrg_out)
summary(model3)

model4 <- lm(`Sale Price`~building_grade+square_feet_total_living, data = HousingOrg_out)
summary(model4)
```

## 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.

```{r include=TRUE}
HousingOrg_out$standardized.residuals <- rstandard(model4)
HousingOrg_out$studentized.residuals <- rstudent(model4)
HousingOrg_out$cooks.distance <- cooks.distance(model4)
HousingOrg_out$dfbeta <- dfbeta(model4)
HousingOrg_out$leverage <- hatvalues(model4)
HousingOrg_out$covariance.ratios <- covratio(model4)
str(HousingOrg_out)
```

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

```{r include=TRUE}
HousingOrg_out$large.residual <- HousingOrg_out$standardized.residuals > 2 |
HousingOrg_out$studentized.residuals < -2
str(HousingOrg_out)
```

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

```{r include=TRUE}
sum(HousingOrg_out$large.residual)

HousingOrg_out[HousingOrg_out$large.residual , c("Sale Price", "square_feet_total_living",
"bath_full_count", "bath_half_count", "bath_3qtr_count", "bedrooms", "sq_ft_lot")]
```

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

```{r include=TRUE}
HousingOrg_out[HousingOrg_out$large.residual , c("leverage" ,
"cooks.distance","covariance.ratios") ]
```
~~~~

As we can none of the values in cook’s distance is greater than 1 or even closer to 1, so we can say none of the cases is having an undue influence on the model. Lets calculate the average of top 4 leverage which will be equal to 4/21.66=5.415e-04 and we can see all the cases are within boundary of the 4 times the average of 5.415e-04 and many cases are close to 3 times the average. We know the covariance ration should be between [1 + 4(leverage average)] and [1 - 4(leverage average] wg=hich will give us [1 + 4(4/21.66)] = 1.00216 and [1-3(4/12865)] = 0.9978 i.e. the range is 0.978 to 1.00216. Most of the cases lies between these boundaries. From above theries we can conclude that the Cook's distance can raise no or little cause for alarm.

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

```{r include=TRUE}
dwt(model4)

```

We can see the DW value is 0.55087 from which we ca conclude that the value is within the limits.

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

```{r include=TRUE}
vif(model4)
1/vif(model4)
mean(vif(model4))
```

We can stat after seeing the result of above function that the largest vif(2.25) which is not greater than 10. we got 0.44 as tolerance values and the mean of the vif is 2.25 We can conclude from these observation that there is no collinearity within the data.

## 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.

```{r include=TRUE}
plot(model4)

library(ggplot2)

hist(HousingOrg_out$studentized.residuals)
```

~~~
```
~~~

Lets look at the fitted values against residual plot we understand that the values are evenly distributed around 0. It is save to assume that this is linear graph. we do not see any funnel type data as random variables do not show finite variances so we can say this model does not show homoscedasticity. By looking at the histogram of final model it looks similar to bell shape. so the data is not skewed and it is not biased.

## 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?

~~~
```{r include=TRUE}
with(HousingOrg_out, hist(standardized.residuals, scale="frequency", breaks="Sturges",
col="red",
                   xlab="Studentized Residuals"))

housing_df$studentized.residuals <- rstudent(SP_other_lm)
hist(housing_df$studentized.residuals)

```
~~~

Looking at the histgram we can see it is bell shaped plot so the designed model is not skewed or not biased. In above plots I used the housing original data and the housing data without the outliers and both show the bell shape plot, i think it is save to believe that both sample and population model are not biased model.

## References