Matthew Reynolds
Oren Mangourbi
DS4635
24 March 2023

Assignment #2

Exercise #3:

A) III is correct. The starting salary for males is higher than for females on average if $50 + 20*GPA \geq 85 + 10$.

B) $50 + 20 * 4 + 0.07 * 110 + 35 * 1 + 0.01 * 4 * 110 – 10 * 4 * 1$
$= 137.1$

C) False. We aren't given the error between GPA & IQ so we cannot be sure whether it's significant.

Exercise #4:

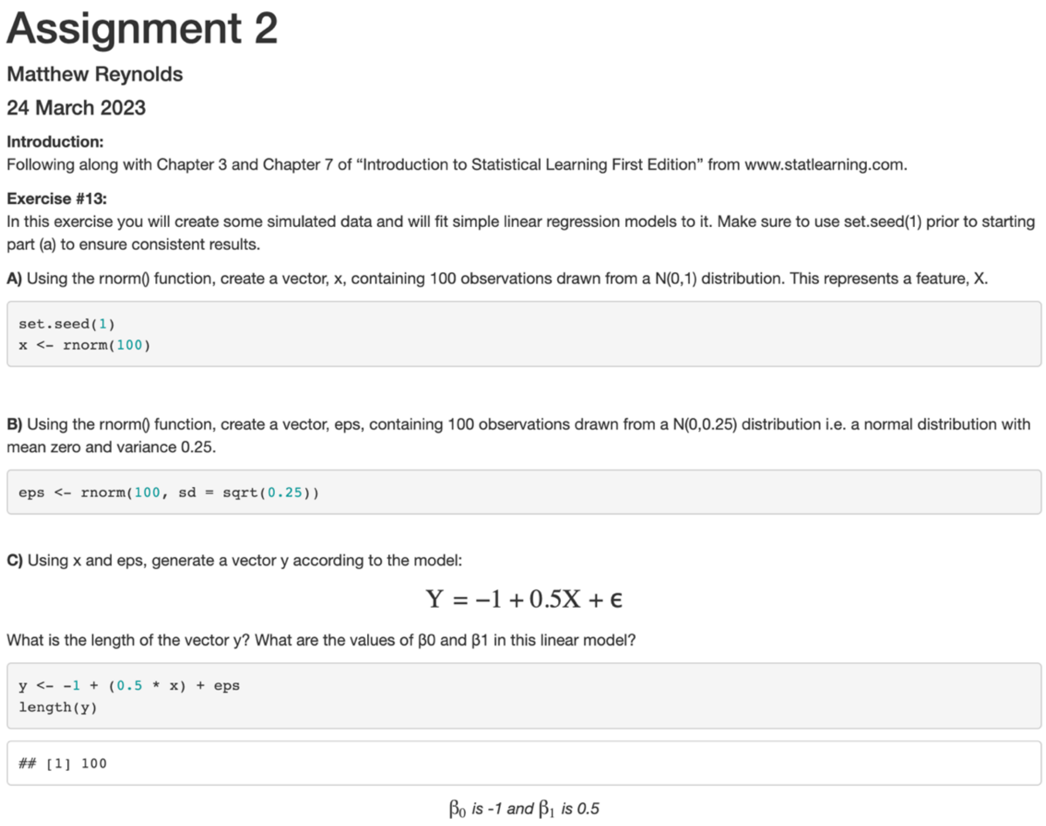
A) The polynomial regression would have a lower training RSS rather than the linear regression because it could make a better fit.

B) The polynomial regression would have a better test RSS since the over trained model would have greater error.

C) The polynomial regression would have the lower training RSS because its high flexibility.

D) We can't really tell since we don't have a visual/graph or statement to tell how linear the graph is. So it depends, if its linear, the linear regression test RSS would be lower but if its cubic then the polynomial test RSS would be lower.

13)



Assignment 2

Matthew Reynolds

24 March 2023

Introduction:
Following along with Chapter 3 and Chapter 7 of “Introduction to Statistical Learning First Edition” from www.statlearning.com.

Exercise #13:
In this exercise you will create some simulated data and will fit simple linear regression models to it. Make sure to use set.seed(1) prior to starting part (a) to ensure consistent results.

A) Using the rnorm() function, create a vector, x, containing 100 observations drawn from a N(0,1) distribution. This represents a feature, X.

```
set.seed(1)
x <- rnorm(100)
```

B) Using the rnorm() function, create a vector, eps, containing 100 observations drawn from a N(0,0.25) distribution i.e. a normal distribution with mean zero and variance 0.25.

```
eps <- rnorm(100, sd = sqrt(0.25))
```

C) Using x and eps, generate a vector y according to the model:

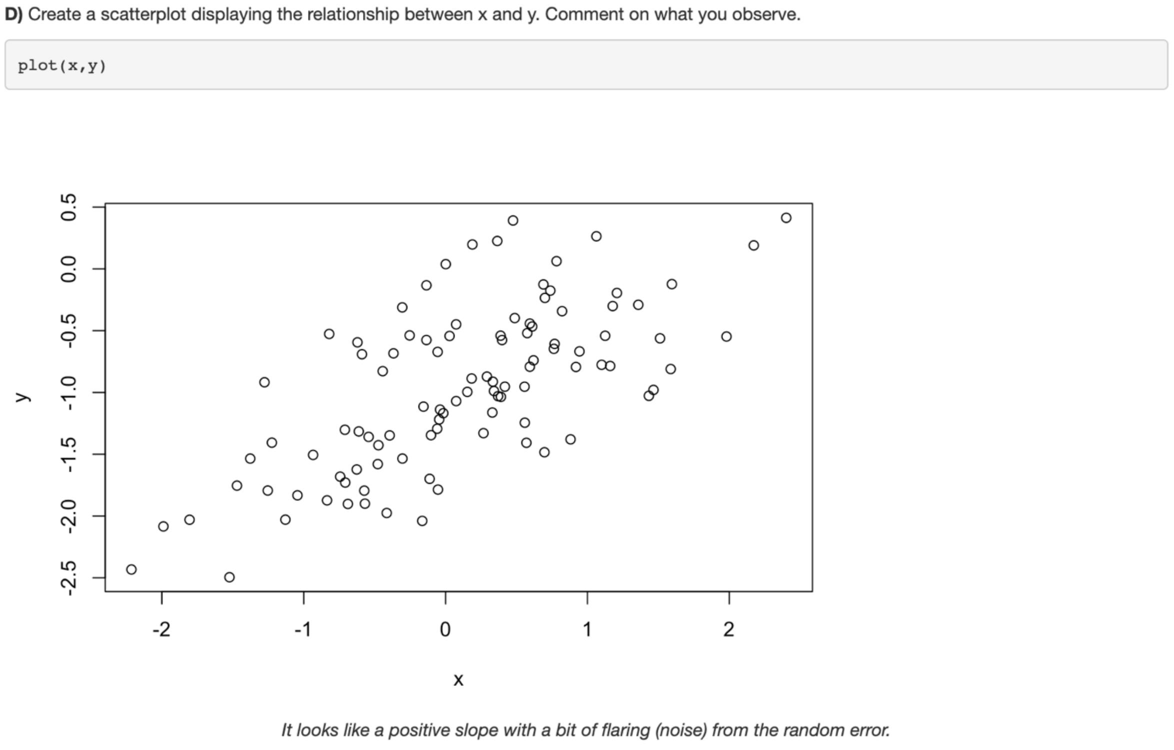
$$Y = -1 + 0.5X + \epsilon$$

What is the length of the vector y? What are the values of β0 and β1 in this linear model?

```
y <- -1 + (0.5 * x) + eps
length(y)
```

```
## [1] 100
```

β_0 is -1 and β_1 is 0.5



D) Create a scatterplot displaying the relationship between x and y. Comment on what you observe.

```
plot(x,y)
```

It looks like a positive slope with a bit of flaring (noise) from the random error.

E) Fit a least squares linear model to predict y using x. Comment on the model obtained. How do βˆ0 and βˆ1 compare to β0 and β1?

```
leastSquares <- lm(y ~ x)
summary(leastSquares)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.93842 -0.30688 -0.06975  0.26970  1.17309
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.01885    0.04849 -21.010  < 2e-16 ***
## x            0.49947    0.05386   9.273 4.58e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4814 on 98 degrees of freedom
## Multiple R-squared:  0.4674, Adjusted R-squared:  0.4619
## F-statistic: 85.99 on 1 and 98 DF,  p-value: 4.583e-15
```

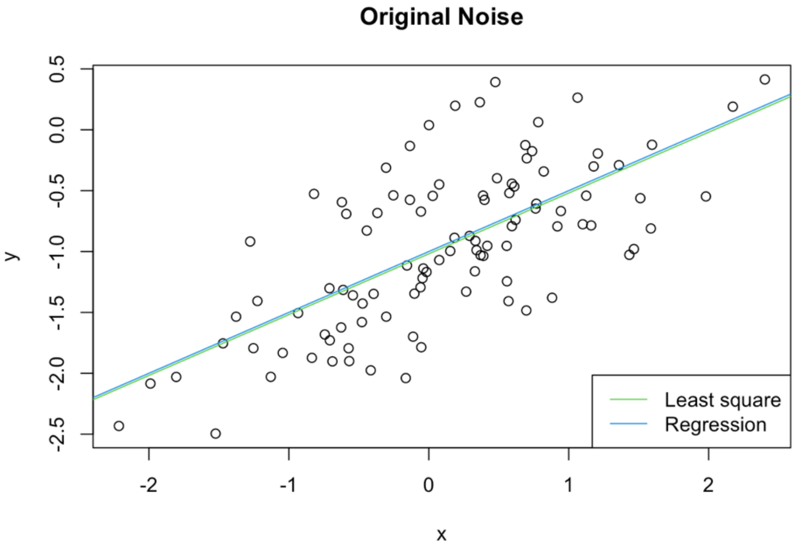
This model has a quite large F-statistic at 85.99 with a essentially 0 p-value.

F) Display the least squares line on the scatterplot obtained in (d). Draw the population regression line on the plot, in a different color. Use the legend() command to create an appropriate legend.

```
plot(x, y, main = "Original Noise")
abline(leastSquares, col = 3)
abline(-1, 0.5, col = 4)
legend("bottomright", c("Least square", "Regression"), col = c("3", "4"), lty = c(1, 1))
```

F) Display the least squares line on the scatterplot obtained in (d). Draw the population regression line on the plot, in a different color. Use the legend() command to create an appropriate legend.

```
plot(x, y, main = "Original Noise")
abline(leastSquares, col = 3)
abline(-1, 0.5, col = 4)
legend("bottomright", c("Least square", "Regression"), col = c("3", "4"), lty = c(1, 1))
```



G) Now fit a polynomial regression model that predicts y using x and x^2. Is there evidence that the quadratic term improves the model fit? Explain your answer.

```
polyReg <- lm(y ~ x + I(x^2))
summary(polyReg)
```

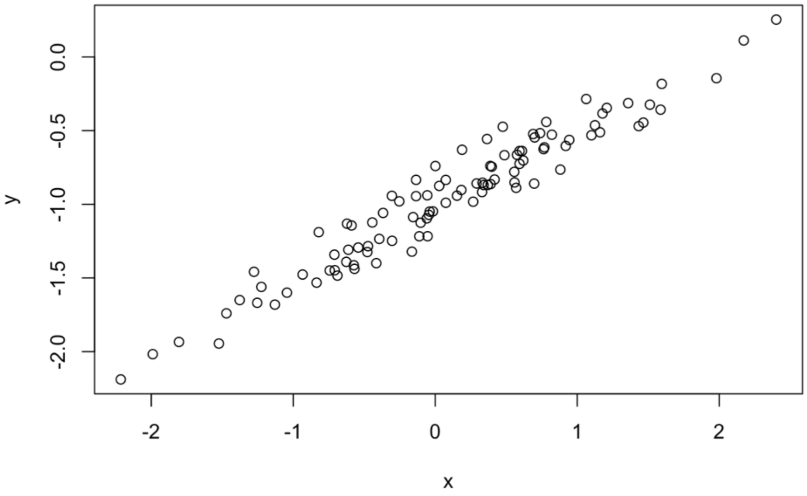
```
##
## Call:
## lm(formula = y ~ x + I(x^2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.98252 -0.31270 -0.06441  0.29014  1.13500
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.97164    0.05883 -16.517  < 2e-16 ***
## x            0.50858    0.05399   9.420  2.4e-15 ***
## I(x^2)      -0.05946    0.04238  -1.403    0.164
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.479 on 97 degrees of freedom
## Multiple R-squared:  0.4779, Adjusted R-squared:  0.4672
## F-statistic:  44.4 on 2 and 97 DF,  p-value: 2.038e-14
```

x^2 *is not significant since it doesn't achieve a p-value lower than 0.05.*

H) Repeat (a)–(f) after modifying the data generation process in such a way that there is less noise in the data. The model (3.39) should remain the same. You can do this by decreasing the vari- ance of the normal distribution used to generate the error term ε in (b). Describe your results.

```
set.seed(1)
x <- rnorm(100)
eps <- rnorm(100, sd = (sqrt(0.25) * 0.25))
y <- -1 + (0.5 * x) + eps
```

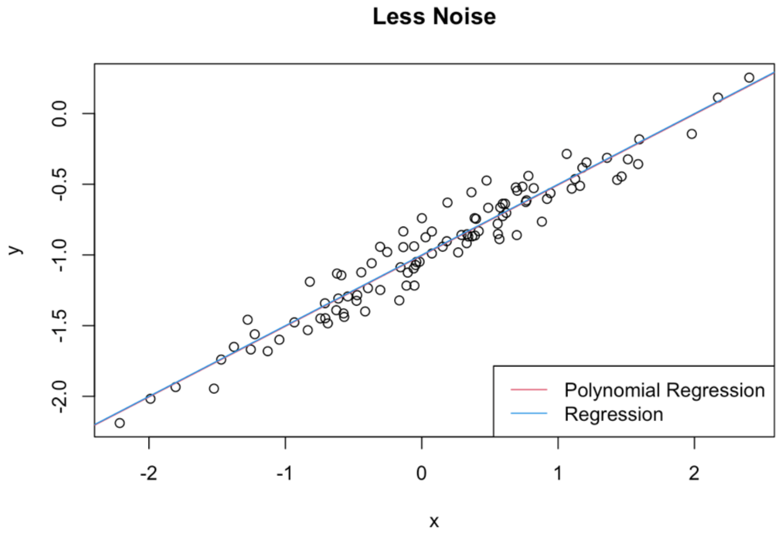
```
plot(x,y)
```



```
model <- lm(y ~ x)
summary(model)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.23461 -0.07672 -0.01744  0.06742  0.29327
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.00471    0.01212  -82.87   <2e-16 ***
## x            0.49987    0.01347   37.12   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1203 on 98 degrees of freedom
## Multiple R-squared:  0.9336, Adjusted R-squared:  0.9329
## F-statistic:  1378 on 1 and 98 DF,  p-value: < 2.2e-16
```

```
plot(x,y, main = "Less Noise")
abline(model, col = 2)
abline(-1, 0.5, col = 4)
legend("bottomright", c("Polynomial Regression", "Regression"), col = c("2", "4"), lty = c(1, 1))
```

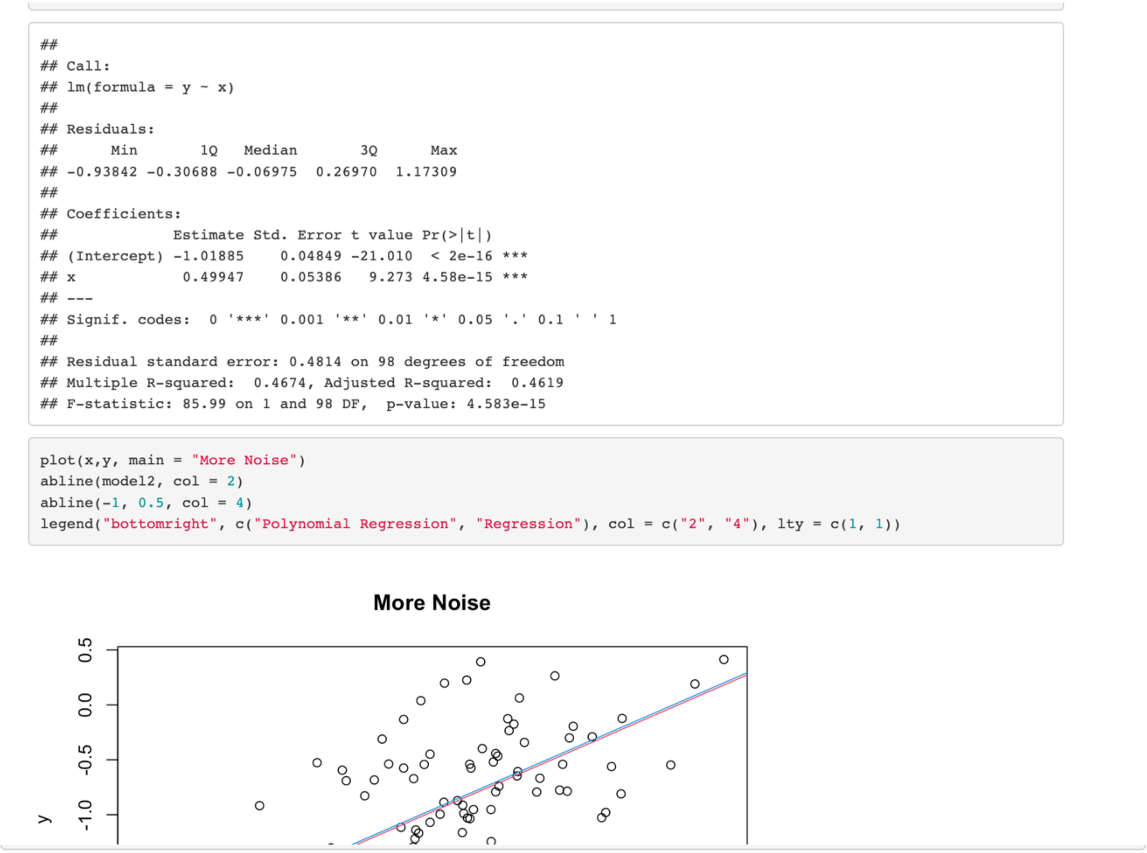


I reduced the noise by multiplying my standard deviation by 0.25. The regression lines are near identical to each other.

l) Repeat (a)–(f) after modifying the data generation process in such a way that there is more noise in the data. The model (3.39) should remain the same. You can do this by increasing the variance of the normal distribution used to generate the error term ε in (b). Describe your results.

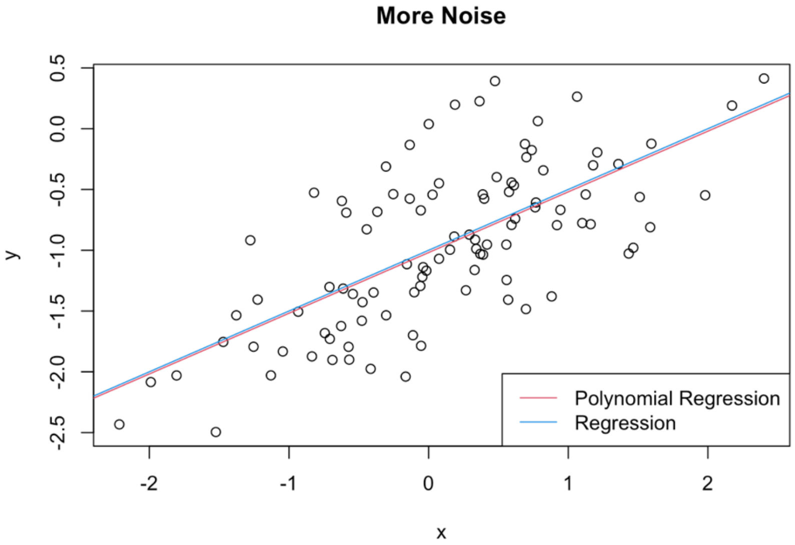
```
set.seed(1)
x <- rnorm(100)
eps <- rnorm(100, sd = 0.5)
y <- -1 + (0.5 * x) + eps
```

```
plot(x,y)
```



```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.93842 -0.30688 -0.06975  0.26970  1.17309
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.01885    0.04849 -21.010  < 2e-16 ***
## x            0.49947    0.05386   9.273 4.58e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4814 on 98 degrees of freedom
## Multiple R-squared:  0.4674, Adjusted R-squared:  0.4619
## F-statistic: 85.99 on 1 and 98 DF,  p-value: 4.583e-15
```

```
plot(x,y, main = "More Noise")
abline(model2, col = 2)
abline(-1, 0.5, col = 4)
legend("bottomright", c("Polynomial Regression", "Regression"), col = c("2", "4"), lty = c(1, 1))
```

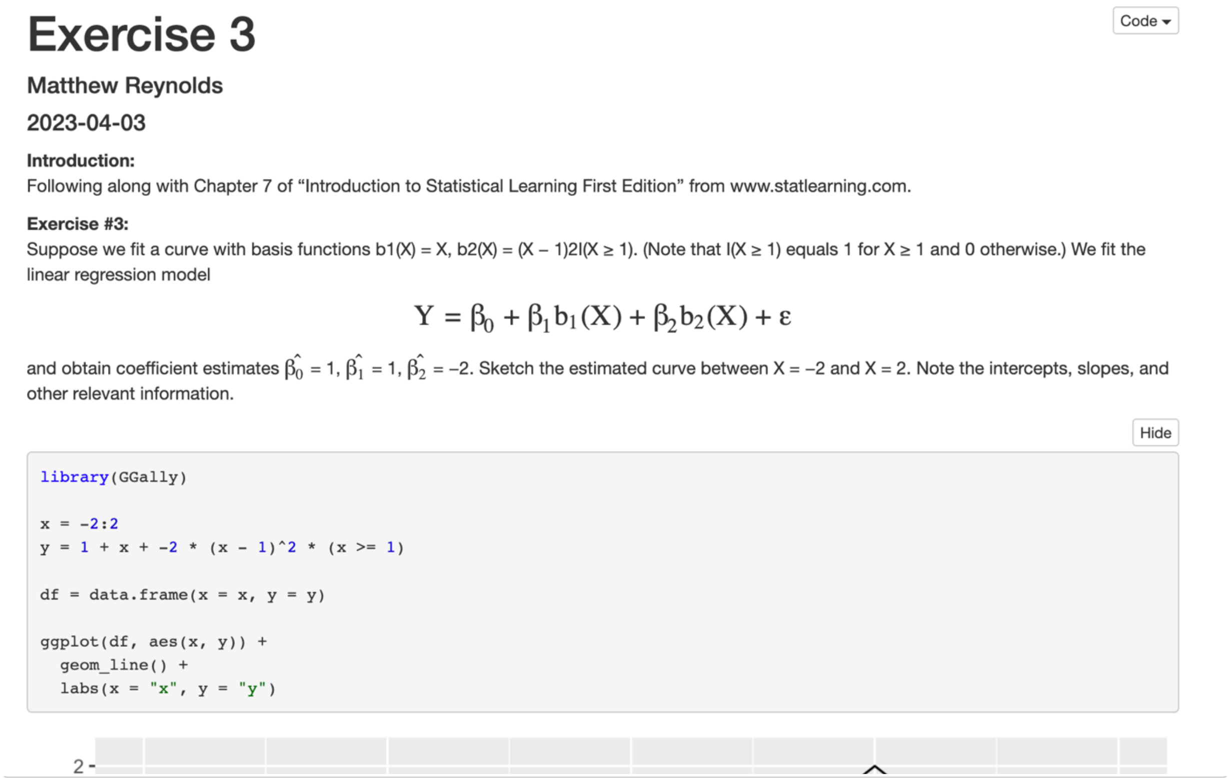


I increased the noise by multiplying my standard deviation by 4. The regression lines are near identical to each other.

J) What are the confidence intervals for β0 and β1 based on the original data set, the noisier data set, and the less noisy data set? Comment on your results.

This part was removed from the assignment after I asked in class for the Cross-Validation R demo.

Exercise #3)



Exercise 3

Matthew Reynolds

2023-04-03

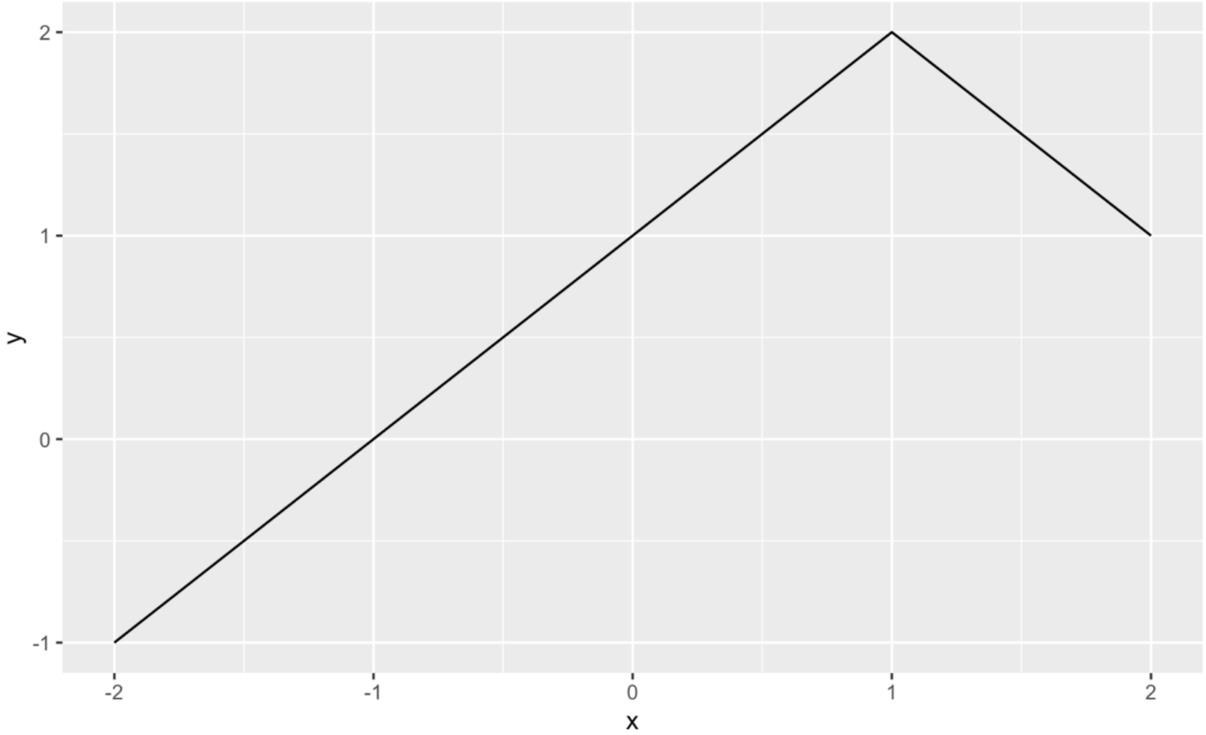
Introduction:
Following along with Chapter 7 of “Introduction to Statistical Learning First Edition” from www.statlearning.com.

Exercise #3:
Suppose we fit a curve with basis functions b1(X) = X, b2(X) = (X – 1)2I(X ≥ 1). (Note that I(X ≥ 1) equals 1 for X ≥ 1 and 0 otherwise.) We fit the linear regression model

$$Y = \beta_0 + \beta_1 b_1(X) + \beta_2 b_2(X) + \varepsilon$$

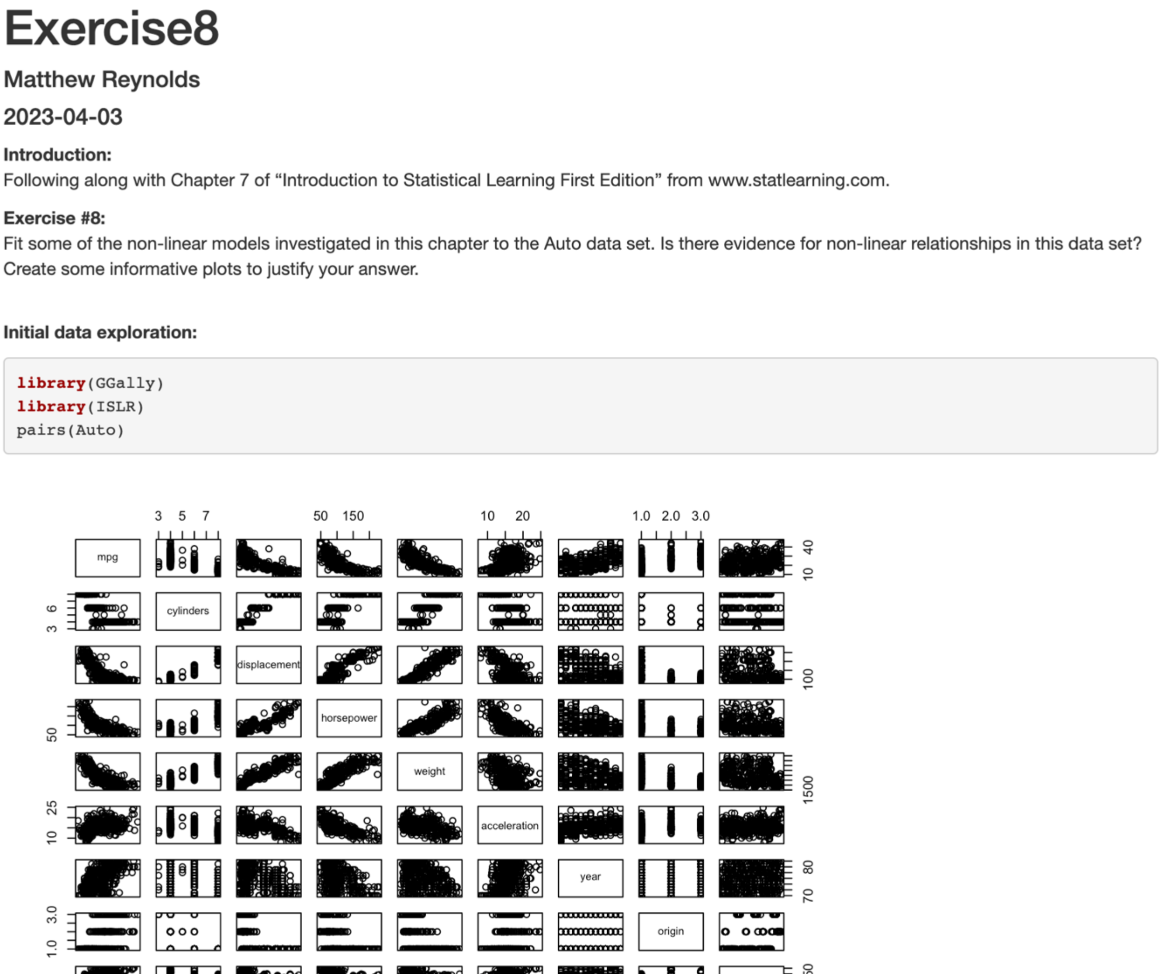
and obtain coefficient estimates $\hat{\beta_0} = 1$, $\hat{\beta_1} = 1$, $\hat{\beta_2} = -2$. Sketch the estimated curve between X = −2 and X = 2. Note the intercepts, slopes, and other relevant information.

```
library(GGally)

x = -2:2
y = 1 + x + -2 * (x - 1)^2 * (x >= 1)

df = data.frame(x = x, y = y)

ggplot(df, aes(x, y)) +
  geom_line() +
  labs(x = "x", y = "y")
```



By graphing the curve, I find that the curve is linear from x = -2:1., $y = 1 + x$. *And the curve is quadratic from x = 1:2,* $y = 1 + x - 2(x - 1)^2$

Exercise 8)



Exercise8

Matthew Reynolds

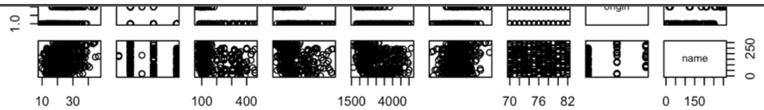
2023-04-03

Introduction:
Following along with Chapter 7 of “Introduction to Statistical Learning First Edition” from www.statlearning.com.

Exercise #8:
Fit some of the non-linear models investigated in this chapter to the Auto data set. Is there evidence for non-linear relationships in this data set? Create some informative plots to justify your answer.

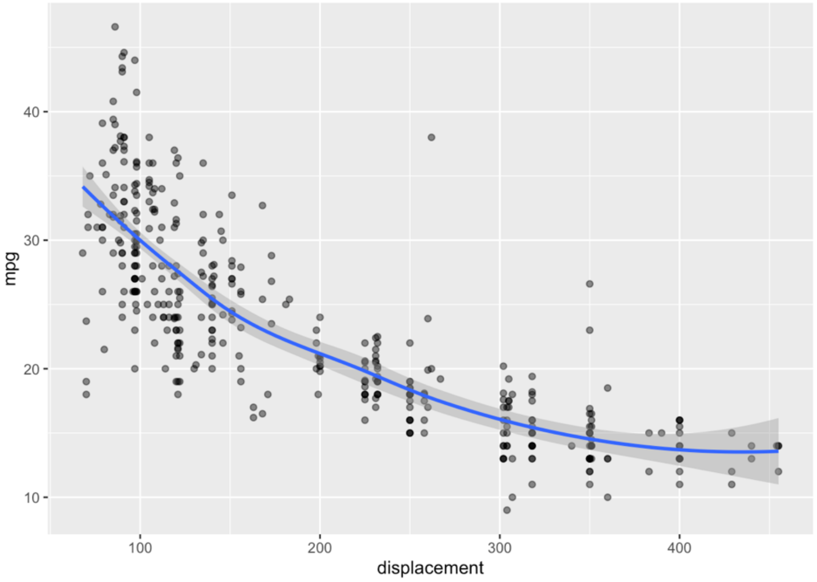
Initial data exploration:

```
library(GGally)
library(ISLR)
pairs(Auto)
```

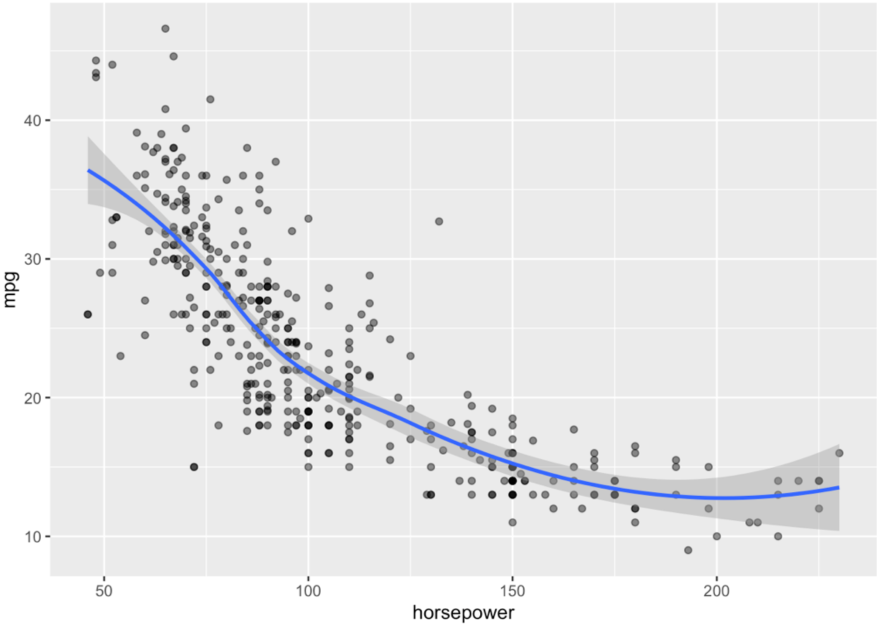


It would appear that the correlation between MPG and seemingly everything is non-linear.

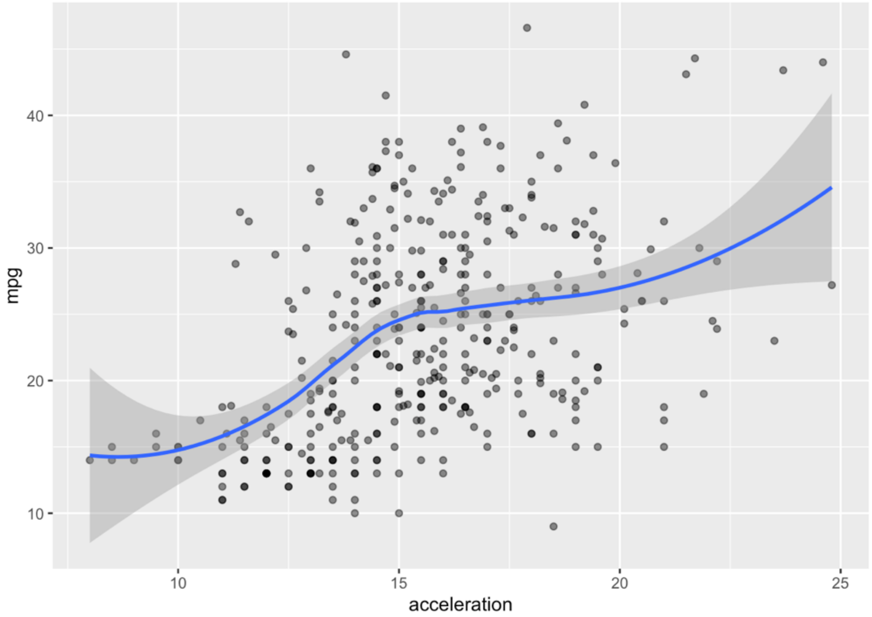
```
ggplot(Auto, aes(x = displacement, y = mpg)) +
  geom_point(alpha = 0.5) +
  geom_smooth()
```



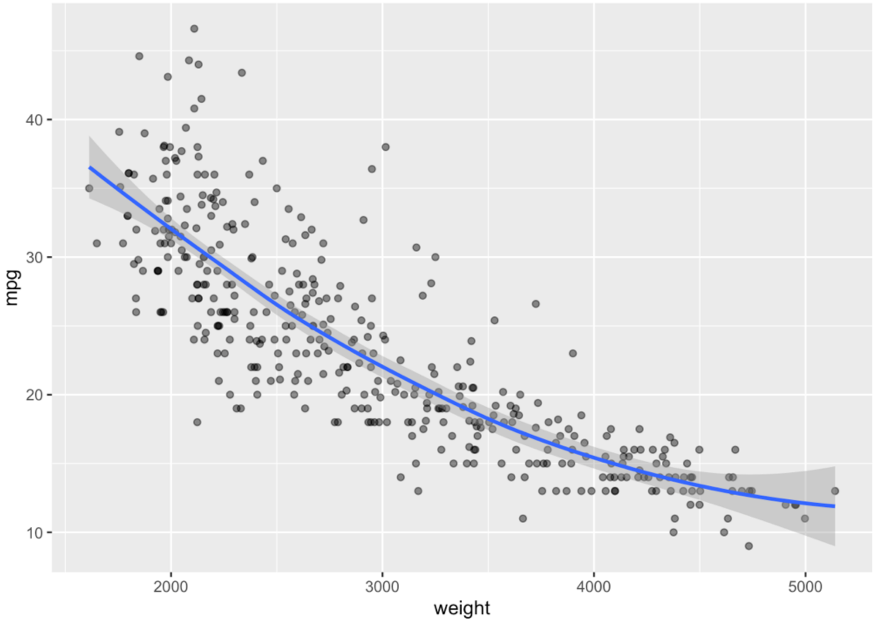
```
ggplot(Auto, aes(x = horsepower, y = mpg)) +
  geom_point(alpha = 0.5) +
  geom_smooth()
```



```
ggplot(Auto, aes(x = acceleration, y = mpg)) +
  geom_point(alpha = 0.5) +
  geom_smooth()
```

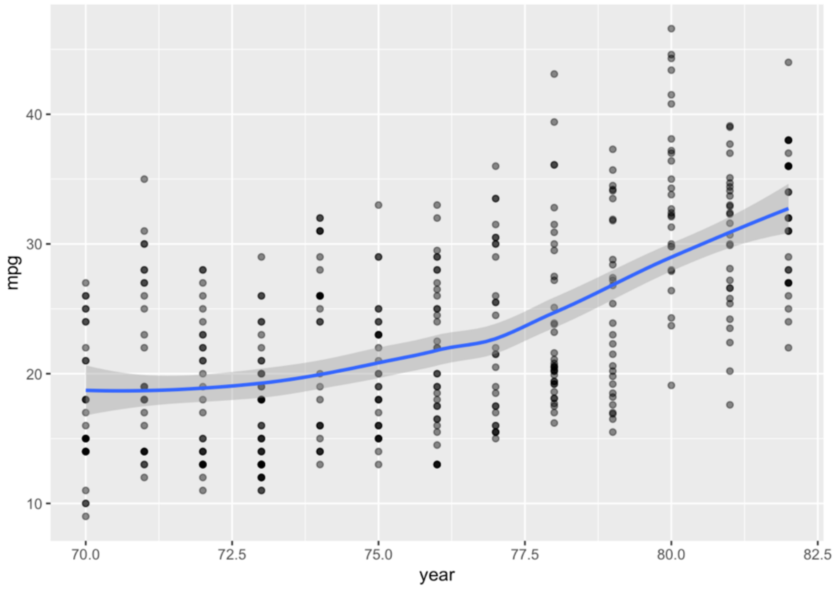


```
ggplot(Auto, aes(x = weight, y = mpg)) +
  geom_point(alpha = 0.5) +
  geom_smooth()
```



acceleration

```
ggplot(Auto, aes(x = year, y = mpg)) +
  geom_point(alpha = 0.5) +
  geom_smooth()
```



Answer:

I would definetly say there’s non linear relationships in this data. All of those graphs are clearly non linear. Especially the mpg x acceleration (second from bottom).