Chapter 3 HW

Nehemya McCarter-Ribakoff

28 February 2017

Conceptual Questions

Exercise 1: Describe the null hypotheses to which the p-values given in Table 3.4 correspond. Explain what conclusions you can draw based on these p-values. Your explanation should be phrased in terms of sales, TV, radio, and newspaper, rather than in terms of the coefficients of the linear model.

Null Hypotheses:

TV: There is no relationship between TV advertising and sales. The p-value is below 0.01, so we may reject the null hypothesis. That is, there is some relationship between TV ads and sales.

radio: There is no relationship between radio advertising and sales. The p-value is below 0.01, so we may reject the null hypothesis. That is, there is some relationship between radio ads and sales.

newspaper: There is not relationship between newspaper advertising and sales. The p-value is about 0.86, which is quite high. Thus, we may retain the null hypothesis. There is no relationship between newspaper ads and sales.

Exercise 3: suppose we have a data set with five predictors, X1 = GPA, X2 = IQ, X3 = Gender (1 for Female and 0 for Male), X4 = Interaction between GPA and IQ, and X5 = Interaction between GPA and Gender. The response is starting salary after graduation (in thousands of dollars). suppose we use least squares to fit the model, and get $\beta_0 = 50$, $\beta_1 = 20$, $\beta_2 = 0.07$, $\beta_3 = 35$, $\beta_4 = 0.01$, $\beta_5 = -10$.

Our intercept coefficient is 50, so our average salary is $50,000.

male: (gender = 0) $50 + 20\ k_1 + 0.07\ k_2 + 0.01(k_1 * k_2)$

female: (gender = 1) $50 + 20\ k_1 + 0.07\ k_2 + 35 + 0.01(k_1 * k_2) - 10k_1$

(a) Which answer is correct, and why?

i. For a fixed value of IQ and GPA, males earn more on average than females.

ii. For a fixed value of IQ and GPA, females earn more on average than males.

iii. For a fixed value of IQ and GPA, males earn more on average than females provided that the GPA is high enough.

iv. For a fixed value of IQ and GPA, females earn more on average than males provided that the GPA is high enough.

Let’s take a look at output of each function for some given values of IQ and GPA.

GPA (k_1): 4.0

IQ (k_2): 120

male: (gender = 0) 50 + 20(4.0) + 0.07(120) + 0.01(4.0 * 120)

female: (gender = 1) 50 + 20(4.0) + 0.07(120) + 35 + 0.01(4.0 * 120) - 10(4.0)

male = 143.2 female = 138.2

At equal values for IQ and GPA, males earn more than females. i is correct, ii is false. Since males already earn more than females, iii is also false.

Interaction between GPA and gender has a negative coefficient, so iv does not make sense.

(b) Predict the salary of a female with IQ of 110 and a GPA of 4.0.

We simply multiply the predictors to the model's corresponding coefficients.

```
50 + (20 * 4.0) + (0.07 * 110) + 35 + (0.01 * 4.0 * 110) - (10 * 4.0)
```

We expect her to have a salary of $137,10.

(c) True or false: Since the coefficient for the GPA/IQ interaction term is very small, there is very little evidence of an interaction effect. Justify your answer.

This is false. A coefficient tells us the intensity of the interaction. It has little bearing on its statistical significance. We must look at the p-value of the coefficient to determine this.

Applied Questions

Exercise 10: This question should be answered using the Carseats data set.

```
library(ISLR)
# Let's get the names so we know what we're doing
names(Carseats)
```

```
##  [1] "Sales"       "CompPrice"   "Income"      "Advertising" "Population"
##  [6] "Price"       "ShelveLoc"   "Age"         "Education"   "Urban"
## [11] "US"
```

```
#Note that unit sales are in $1,000.
```

(a) Fit a multiple regression model to predict Sales using Price, Urban, and US.

```
lm.fit = lm(Carseats$Sales~Carseats$Price+Carseats$Urban+Carseats$US)
lm.fit
```

```
##
## Call:
## lm(formula = Carseats$Sales ~ Carseats$Price + Carseats$Urban +
##     Carseats$US)
##
## Coefficients:
##      (Intercept)     Carseats$Price  Carseats$UrbanYes
##         13.04347           -0.05446           -0.02192
##    Carseats$USYes
##          1.20057
```

(b) Provide an interpretation of each coefficient in the model. Be careful—some of the variables in the model are qualitative!

Our Intercept is 13.04 – when all predictors are averaged, we can expect sales of $13,040

The coefficient of price is -0.05446. As price increases, sales marginally decrease.

Stores in urban locations exhibit marginally lower sales. Important to note that -0.022 is quite a small coefficient.

There is a small, positive relationship between a U.S. store location and an increase in sales.

(c) Write out the model in equation form, being careful to handle the qualitative variables properly.

Sales = β_0 + β_1Price + β_2UrbanYes + β_3USYes

(d) For which of the predictors can you reject the null hypothesis H0 : β_j = 0?

We must look at the p-values

```
summary(lm.fit)
```

```
##
## Call:
## lm(formula = Carseats$Sales ~ Carseats$Price + Carseats$Urban +
##     Carseats$US)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.9206 -1.6220 -0.0564  1.5786  7.0581
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       13.043469   0.651012  20.036  < 2e-16 ***
## Carseats$Price    -0.054459   0.005242 -10.389  < 2e-16 ***
## Carseats$UrbanYes -0.021916   0.271650  -0.081    0.936
## Carseats$USYes     1.200573   0.259042   4.635 4.86e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.472 on 396 degrees of freedom
## Multiple R-squared:  0.2393, Adjusted R-squared:  0.2335
## F-statistic: 41.52 on 3 and 396 DF,  p-value: < 2.2e-16
```

We can reject the null hypotheses for Price and USYes, as these p-values are both 2^{-16}

(e) On the basis of your response to the previous question, fit a smaller model that only uses the predictors for which there is evidence of association with the outcome.

```
lm.smallFit = lm(Carseats$Sales~Carseats$Price + Carseats$US)
```

(f) How well do the models in (a) and (e) fit the data?

```
summary(lm.fit)$sigma #RSE fit
```

```
## [1] 2.472492
```

```
summary(lm.smallFit)$sigma #RSE smallFit
```

```
## [1] 2.469397
```

```
summary(lm.fit)$r.sq #R^2 fit
```

```
## [1] 0.2392754
```

```
summary(lm.smallFit)$r.sq #R^2 smallFit
```

```
## [1] 0.2392629
```

The models fit the data about the same, though the original fit has a larger R_2, so it is a slightly better fit. Both are much closer to 0 than 1, so neither is a very good fit.

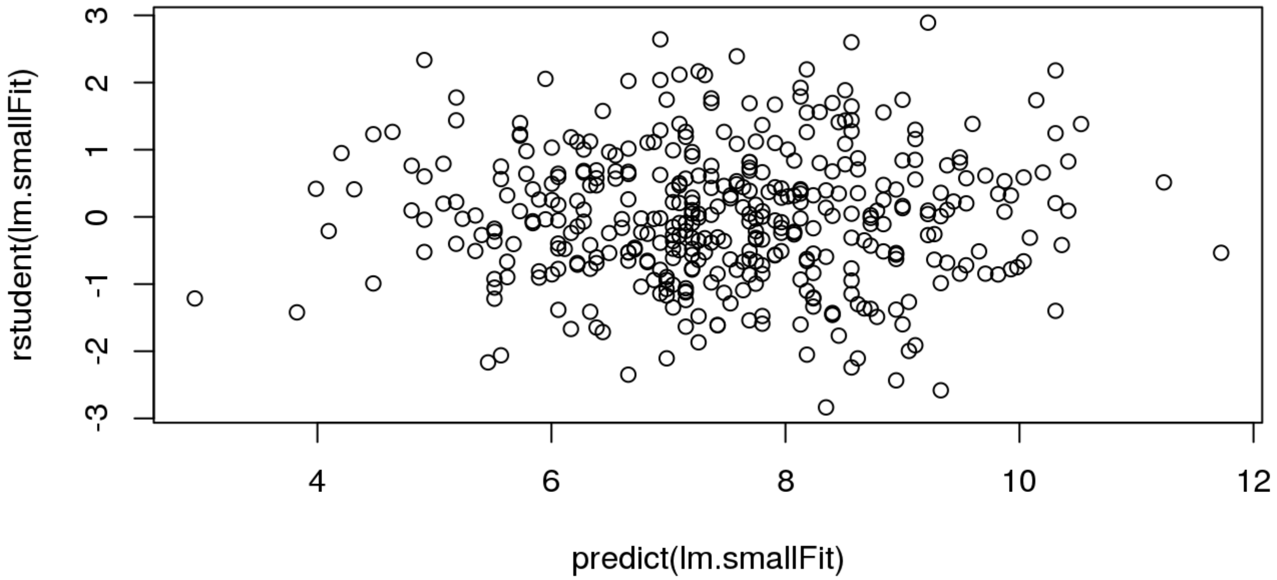
(g) Using the model from (e), obtain 95% confidence intervals for the coefficient(s).

```
confint(lm.smallFit, level = 0.95)
```

```
##                       2.5 %      97.5 %
## (Intercept)    11.79032020 14.27126531
## Carseats$Price -0.06475984 -0.04419543
## Carseats$USYes  0.69151957  1.70776632
```

(h) Is there evidence of outliers or high leverage observations in the model from (e)?

```
plot(predict(lm.smallFit), rstudent(lm.smallFit))
```



Studentized residuals fall between -3 and 3, so we do not appear to have outliers.

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.

seed(1)

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

```
x = rnorm(100, mean = 0, sd = 1)
```

(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, mean = 0, sd = sqrt(0.25))
```

(c) Using x and eps, generate a vector y according to the model

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

What is the length of the vector y? What are the values of β_0 and β_1 in this linear model?

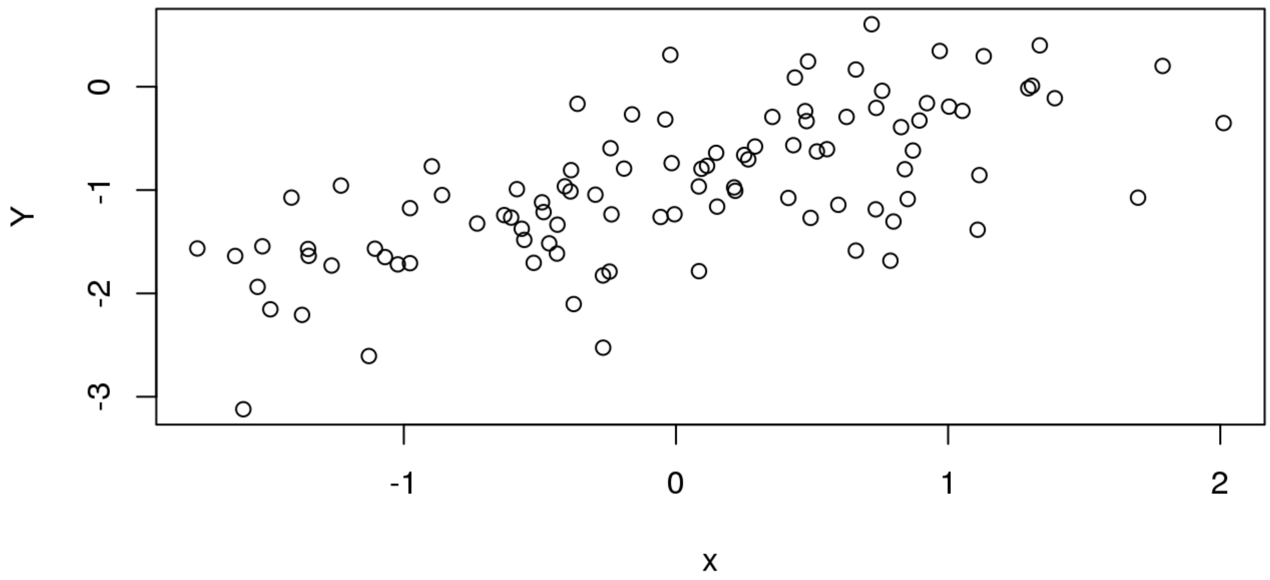
```
Y = -1 + (0.5 * x) + eps
length(Y)
```

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

The length of Y is 100. β_0 = -1 and β_1 = 0.5

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

```
plot(x, Y)
```



There is a general positive relationship between x and Y

(e) Fit a least squares linear model to predict y using x. Comment on the model obtained. How do β_0 and β_1 compare to β^$_0$ and β^$_1$?

```
ls.fit = lm(Y~x)
summary(ls.fit)
```

```
##
## Call:
## lm(formula = Y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.42030 -0.29084  0.07494  0.29626  1.27702
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.95658    0.05402 -17.707  < 2e-16 ***
## x            0.55439    0.06240   8.885 3.18e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5402 on 98 degrees of freedom
## Multiple R-squared:  0.4461, Adjusted R-squared:  0.4405
## F-statistic: 78.94 on 1 and 98 DF,  p-value: 3.175e-14
```

β^$_0$ = -0.97274

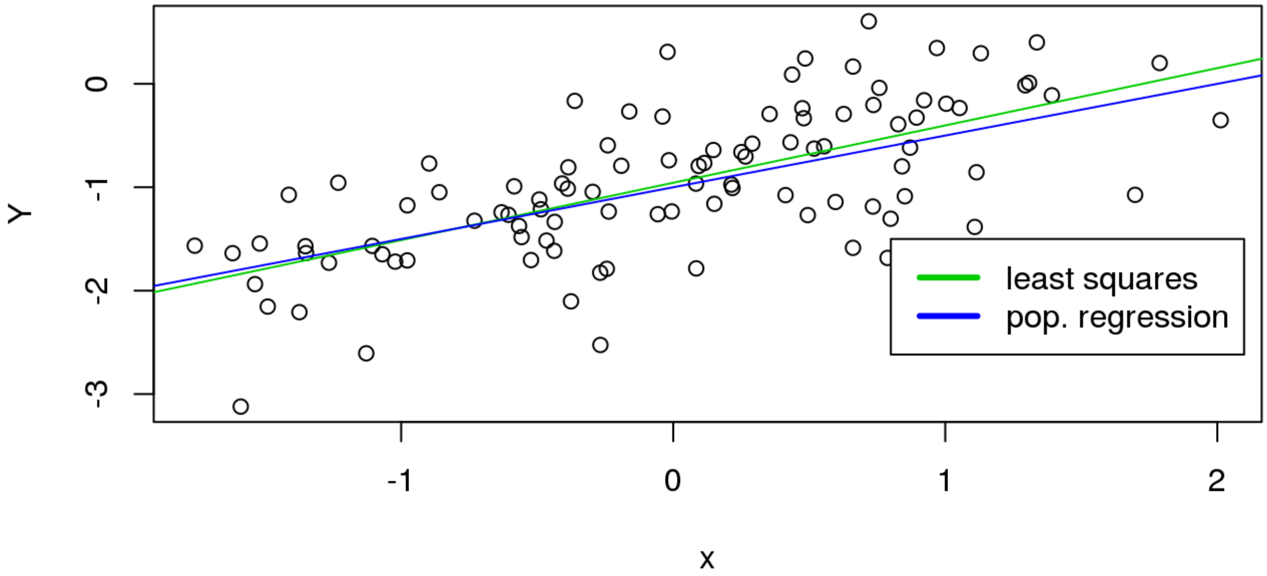
β^$_1$ = 0.51947

β^$_0$ is closer to 0 than β_0

β^$_1$ > β_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)
abline(ls.fit, col = 3)
abline(-1, 0.5, col = 4)
legend(0.8,-1.5, legend = c("least squares", "pop. regression"), col=3:4, lwd = 3)
```



(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.

```
lm.fit2 = lm(Y~x + x^2)
summary(lm.fit2)
```

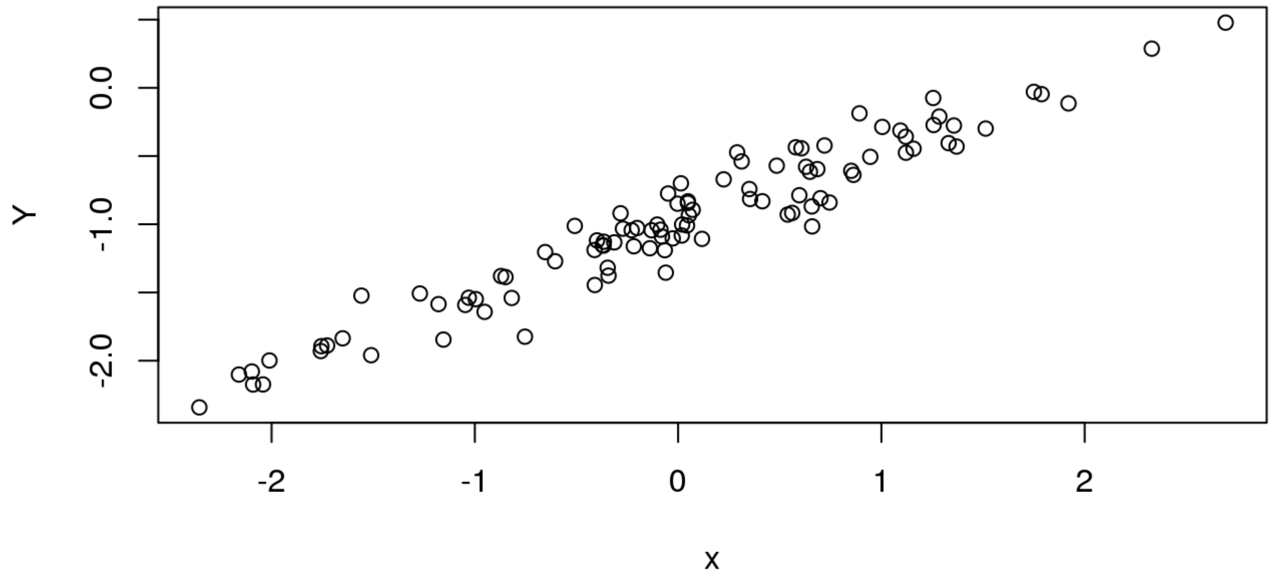
```
##
## Call:
## lm(formula = Y ~ x + x^2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.42030 -0.29084  0.07494  0.29626  1.27702
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.95658    0.05402 -17.707  < 2e-16 ***
## x            0.55439    0.06240   8.885 3.18e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5402 on 98 degrees of freedom
## Multiple R-squared:  0.4461, Adjusted R-squared:  0.4405
## F-statistic: 78.94 on 1 and 98 DF,  p-value: 3.175e-14
```

Based on the provided statistics, the quadratic term has no effect on the model fit. RSE, R^2, and F-statistic are all identical in the original and the quadratic model.

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

sd = sqrt(var), so let’s try decreasing sd. we will divide eps’ sd by 10

```
x = rnorm(100, mean = 0, sd = 1)
eps = rnorm(100, mean = 0, sd = sqrt(0.25 / 10))
Y = -1 + (0.5 * x) + eps
# (d) create a scatterplot, comment on observations
plot(x, Y)
```



The shape here is decidedly more linear and uniform. It more cloesly represents a simple regression line.

```
# (e) fit a least squares linear regression model. Comment
# on the model obtained. How do β^0 and β^1 compare to
# β0 and β1?

ls.fitLessNoise = lm(Y~x)
summary(ls.fitLessNoise)
```

```
##
## Call:
## lm(formula = Y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.43341 -0.10206  0.02323  0.07332  0.35969
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.98808    0.01542  -64.08   <2e-16 ***
## x            0.53428    0.01482   36.05   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1542 on 98 degrees of freedom
## Multiple R-squared:  0.9299, Adjusted R-squared:  0.9292
## F-statistic:  1300 on 1 and 98 DF,  p-value: < 2.2e-16
```

For reference,

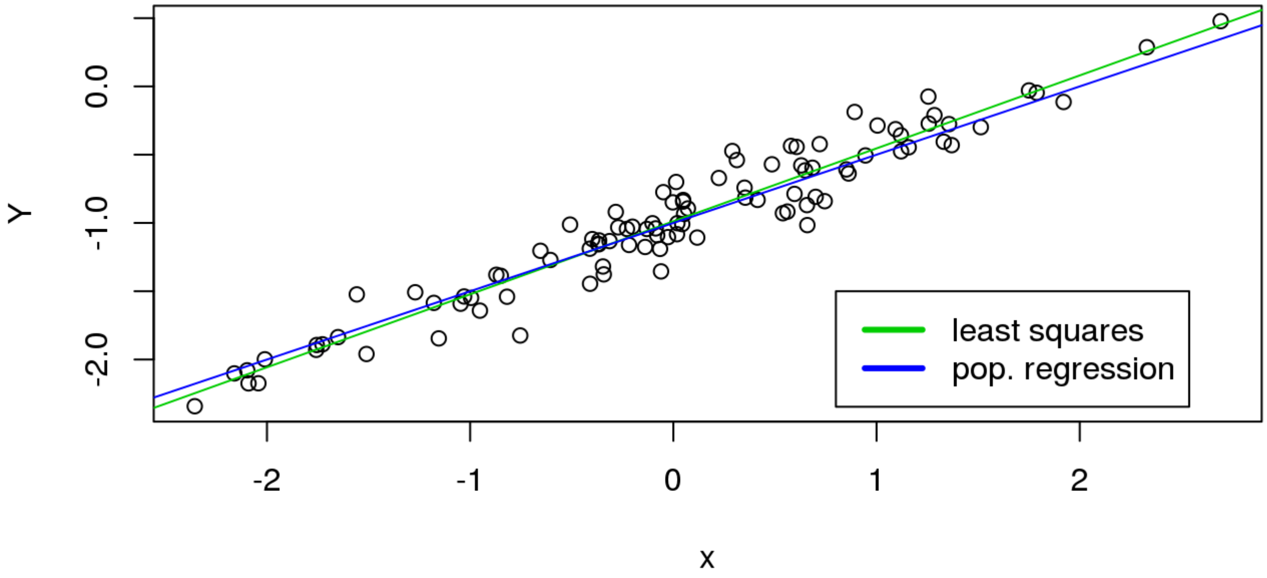
β_0 = -1

β_1 = 0.5

Here, $\beta\text{^}_0$ = -1.00456 and $\beta\text{^}_1$ = 0.49701

Both values have decreased, with $\beta\text{^}_1$ getting closer to 0.

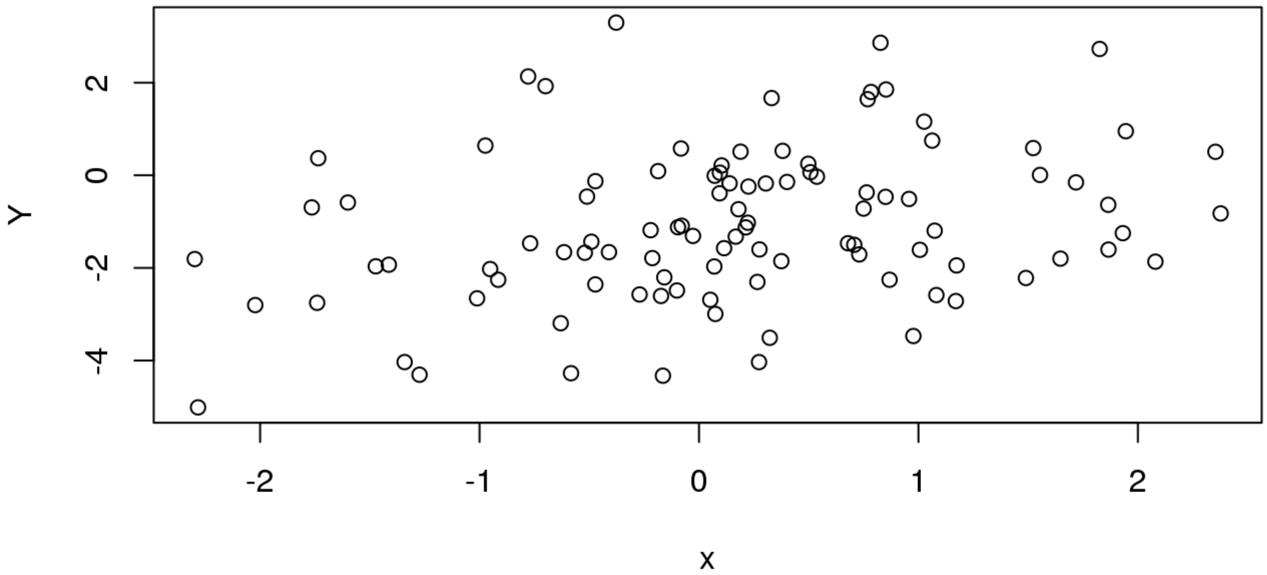
```
# (f) Display the least squares line on the scatterplot obtained in (d).
# Draw the population regression line on the plot, in a different
# color. Use the legend() command to create an appropriate legend.

plot(x, Y)
abline(ls.fitLessNoise, col = 3)
abline(-1, 0.5, col = 4)
legend(0.8 ,-1.5, legend = c("least squares", "pop. regression"), col=3:4, lwd = 3)
```



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

We will multiply eps' sd by 10

```
x = rnorm(100, mean = 0, sd = 1)
eps = rnorm(100, mean = 0, sd = sqrt(2.5))
Y = -1 + (0.5 * x) + eps
# (d) create a scatterplot, comment on observations
plot(x, Y)
```



The data now has virtually no shape. There is no discernible relationship between x and Y

```
# (e) fit a least squares linear regression model. Comment
# on the model obtained. How do β^0 and β^1 compare to
# β0 and β1?

ls.fitMoreNoise = lm(Y~x)
summary(ls.fitMoreNoise)
```

```
##
## Call:
## lm(formula = Y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.1142 -1.0060 -0.1078  1.0207  4.6100
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.1345     0.1621  -6.998 3.24e-10 ***
## x             0.4758     0.1566   3.038  0.00306 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.599 on 98 degrees of freedom
## Multiple R-squared:  0.08605,    Adjusted R-squared:  0.07673
## F-statistic: 9.227 on 1 and 98 DF,  p-value: 0.003056
```

For reference,

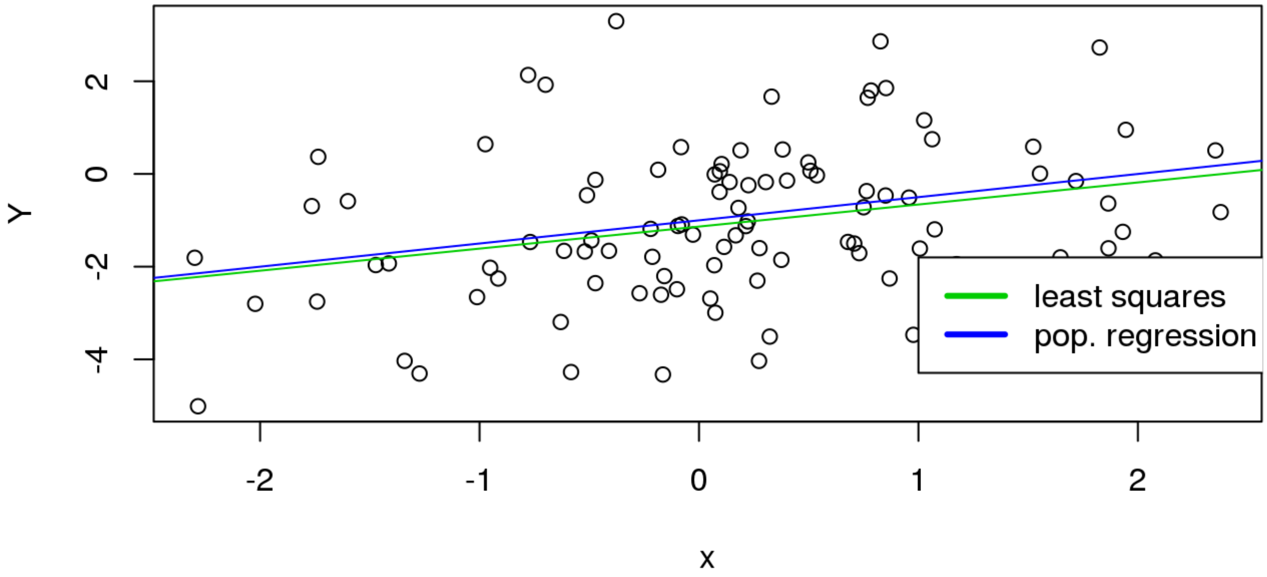
β_0 = -1 and β_1 = 0.5

$\beta\hat{}_0$ = -1.03724

$\beta\hat{}_1$ = 0.01037

$\beta\hat{}_1$ is significantly closer to 0, while $\beta\hat{}_0$ has descender further in the negative direction.

```
# (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)
abline(ls.fitMoreNoise, col = 3)
abline(-1, 0.5, col = 4)
legend(1,-1.8, legend = c("least squares", "pop. regression"), col=3:4, lwd = 3)
```



(j) What are the confidence intervals for β_0 and β_1 based on the original data set, the noisier data set, and the less noisy data set? Comment on your results.

Original data set

```
confint(ls.fit)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.0637834 -0.8493709
## x            0.4305646  0.6782151
```

Less noisy data set

```
confint(ls.fitLessNoise)
```

```
##                   2.5 %     97.5 %
## (Intercept) -1.0186759 -0.9574784
## x            0.5048702  0.5636830
```

More noisy data set

```
confint(ls.fitMoreNoise)
```

```
##                   2.5 %     97.5 %
## (Intercept) -1.4562372 -0.8127991
## x            0.1649511  0.7865687
```

As we’d expect, the less noisy data set has the most narrow confidence interval, since its data have all been squeezed together. The noisy data set has the widest interval, because its data is scattered everywhere. The original set falls somewhere in between.

Teamwork report

Team member	Conceptual	Applied	Contribution %
Nehemya	Yes	Yes	100%
Total			100%