Problem Set #12

Econ 103

Part I – Problems from the Textbook

Since I have assigned two even-numbered problems from the book, I will post solutions to these along with those for the "Additional Problems."

12-2, 12-3, 13-5, 13-12, 14-1, 14-3, 14-5

For 13-5, part (e) you can download the data from my website as follows:

```
data.url <- 'http://www.ditraglia.com/econ103/ex_13_5.csv'
election <- read.csv(data.url)
head(election)
```

```
##   year    y x1  x2
## 1 1946  7.3 32 -40
## 2 1950  2.0 43 100
## 3 1954  2.3 65 -10
## 4 1958  5.9 56 -10
## 5 1962 -0.8 67  60
## 6 1966  1.7 48 100
```

Similarly, the data for 14-5 can be downloaded as follows:

```
data.url <- 'http://www.ditraglia.com/econ103/ex_14_5.csv'
bpdata <- read.csv(data.url)
head(bpdata)
```

```
##   D WEIGHT BP
## 1 0    180 81
## 2 0    150 75
## 3 0    210 83
## 4 0    140 74
```

```
## 5 0    160 72
## 6 0    160 80
```

Part II – Additional Problems

1. This question is based on the dataset on child test scores and mother characteristics we studied during our final lecture of the semester. Before working on this question, make sure you've installed the package `arm` in RStudio. You can download the data from:

   `www.ditraglia.com/econ103/child_test_data.csv`

   The columns contained in this dataset are as follows:

Variable Name	Description
`kid.score`	Child's Test Score at Age 3
`mom.age`	Age of Mother at Birth of Child
`mom.hs`	Mother Completed High School? (1 = Yes)
`mom.iq`	Mother's IQ Score

   (a) Run a regression of `kid.score` on `mom.age`. Plot both the data and the fitted regression line, making sure to label the axes. Interpret the results. At what age do you recommend mothers give birth? What assumptions must you make to justify your recommendation?

   (b) Augment your model from part (a) by allowing a different intercept for children whose mother completed high school. Plot the data along with the regression lines for each group (those whose mother completed high school and those whose mother did not). Interpret your results and compare them to those you got in part (a).

   (c) Now allow different slopes as well as intercepts for each group (those whose mother completed high school and those whose mother did not). Plot the data and the regression lines for each group and interpret your results.

2. This example is based on 12-1 from WW4, but has been adapted somewhat for you to carry out in R. Suppose that the following expression gives the true relationship, i.e. the long-run average, between corn yield in tons per acre (Y) and the amount of fertilizer used in hundreds of pounds per acre (X)

$$Y = 2.40 + 0.30X$$

This means that the population regression parameters are $\beta_0 = 2.40$ and $\beta_1 = 0.30$. Normally we don't know these parameters but rather use data to estimate them. In this question, however, we will pretend that we know these parameters and carry out a Monte Carlo simulation to understand how sampling variability works in the context of regression.

(a) Write an R function called `y.plus.noise` that takes as its input a vector `x` of X-values and returns the corresponding Y values from the above equation *plus a standard normal error term*. The error term should be a *different* random number for each element.

(b) Define `x.test <- 0:12`, a vector containing all the integers from 0 to 12. Test our your function from part (a) by inputting `x.test` and assigning the result to `y.sim`. Make a plot of the function $Y = 2.40 + 0.30X$ along with the points `x.test` and `y.sim`. Try repeating this: you should get a different result because the error terms are *random*.

(c) Run a regression of `y.sim` on `x.test` using the R command `lm(y.sim  x.test)`. Compare the estimated coefficients to the population regression parameters from above by adding the *fitted* regression line to your plot from part (b) using code similar to the following:

```
reg <- lm(y.sim ~ x.test)
estimates <- coefficients(reg)
a.estimate <- estimates[1]
b.estimate <- estimates[2]
abline(a = a.estimate, b = b.estimate, lty = 2)
```

The command `coefficients` extracts the estimated regression coefficients as a numeric vector, while `abline` plots a line based on its intercept (a) and slope (b). Setting the parameter `lty = 2` gives a dashed line. If you repeat part (b) followed by part (c) what changes in the picture and what stays the same?

(d) Adapting the code from above, write a function called `slope.sim` that takes as its input a vector `x` of X-values and then does the following:

Step 1 Create a vector `y.sim` as above.

Step 2 Regress `y.sim` on `x` and call the result `reg`.

Step 3 Return the *estimated slope coefficient* from `reg`.

(e) To simulate the sampling distribution of the estimated regression slope parameter using the population regression given above, use the function `replicate` to call your function `slope.sim` 1000 times and store the result in a vector called `b.sim`. In each of these replications, use `x.test` as the input for `slope.sim`.

(f) Calculate the mean standard deviation of the vector `b.sim` and plot a histogram. Explain your results.