Stat/Math 415 Homework 6

Due Friday Nov 1; Joseph Sepich (jps6444)

1 Problem 7.6-3

```
midterms <- c(70, 74, 80, 84, 80, 67, 70, 64, 74, 82)
finals <- c(87, 79, 88, 98, 96, 73, 83, 79, 91, 94)

meanx <- sum(midterms) / 10
meany <- sum(finals) / 10

beta <- sum((finals - meany) * (midterms - meanx)) / sum((midterms - meanx)^2)

epsilon <- finals - meany - beta * (midterms - meanx)
variance_est <- sum(epsilon^2) / 10

print(paste0('Sample mean x: ',meanx))
```

```
## [1] "Sample mean x: 74.5"
```

```
print(paste0('Sample mean y(alpha): ',meany))
```

```
## [1] "Sample mean y(alpha): 86.8"
```

```
print(paste0('Beta: ',beta))
```

```
## [1] "Beta: 1.01568154402895"
```

```
print(paste0('SigmaSquared: ',variance_est))
```

```
## [1] "SigmaSquared: 17.9998069963812"
```

Suppose x denotes the midterm score, which indicates the final score y. This gives us the least squares regression line:

$$y = 86.8 + 1.0157(x - 74.5) = 11.132 + 1.0157x$$

1.1 Part a

To find a 95% confidence interval we use the following equation:

$$\hat{\alpha} + \hat{\beta}(x - \overline{x}) + -t_{\alpha/2}\sqrt{\frac{n}{n-2}\hat{\sigma^2}(\frac{1}{n} + \frac{(x-\overline{x})^2}{\Sigma_{i=1}^n (x_i - \overline{x})^2})}$$

Looking up the value for $t_{0.025}$ in the t distribution table for n - 2 = 10 - 2 = 8 degrees of freedom, we get t = 2.306. We plug the values into our equation for each value of x we want to get a CI for and we get:

```r
xVals <- c(68, 75, 82)
n <- 10

for (x_val in xVals) {
  y_hat <- meany + beta*(x_val - meanx)
  t <- 2.306
  sqrt_coefficient <- (n * variance_est) / (n-2) #n * sigmasquared / n-2
  sqrt_val <- (1 / n) + ((x_val - meanx)^2 / (sum((midterms - meanx)^2)))

  lower_bound <- y_hat - (t * sqrt(sqrt_coefficient * sqrt_val))
  upper_bound <- y_hat + (t * sqrt(sqrt_coefficient * sqrt_val))

  print(paste0('Lower bound for ',x_val,' is ',lower_bound))
  print(paste0('Upper bound for ',x_val,' is ',upper_bound))
}
```

```
## [1] "Lower bound for 68 is 75.2827850670534"
## [1] "Upper bound for 68 is 85.1133548605703"
## [1] "Lower bound for 75 is 83.8384438112727"
## [1] "Upper bound for 75 is 90.7772377327562"
## [1] "Lower bound for 82 is 89.1071380547519"
## [1] "Upper bound for 82 is 99.7280851056824"
```

This makes the following Confidence Intervals:

- x=68 y=[75.283, 85.113]
- x=75 y=[83.838, 90.777]
- x=82 y=[89.107, 99.728]

1.2 Part b

For the prediction interval we do the same thing, but add in variance due to random error resulting in the following equation:

$$\hat{\alpha} + \hat{\beta}(x - \overline{x}) + -t_{\alpha/2}\sqrt{\frac{n}{n-2}\hat{\sigma^2}(1 + \frac{1}{n} + \frac{(x - \overline{x})^2}{\Sigma_{i=1}^{n}(x_i - \overline{x})^2})}$$

```r
for (x_val in xVals) {
  y_hat <- meany + beta*(x_val - meanx)
  t <- 2.306
  sqrt_coefficient <- (n * variance_est) / (n-2) #n * sigmasquared / n-2

  # DIFFERENCE HERE FROM PART A
  sqrt_val <- 1 + (1 / n) + ((x_val - meanx)^2 / (sum((midterms - meanx)^2)))

  lower_bound <- y_hat - (t * sqrt(sqrt_coefficient * sqrt_val))
  upper_bound <- y_hat + (t * sqrt(sqrt_coefficient * sqrt_val))

  print(paste0('Lower bound for ',x_val,' is ',lower_bound))
  print(paste0('Upper bound for ',x_val,' is ',upper_bound))
}
```

```
## [1] "Lower bound for 68 is 68.20617467105"
## [1] "Upper bound for 68 is 92.1899652565736"
## [1] "Lower bound for 75 is 75.8325524371854"
## [1] "Upper bound for 75 is 98.7831291068436"
## [1] "Lower bound for 82 is 82.2583905459186"
## [1] "Upper bound for 82 is 106.576832614516"
```

This makes the following Prediction Intervals:

- x=68 y=[68.206, 92.190]
- x=75 y=[75.833, 98.783]
- x=82 y=[82.258, 106.577]

2 Problem 7.6-7

```r
x <- c(32, 23, 23, 23, 26, 30, 17, 20, 17, 18, 26, 16, 21, 24, 30)
y <- c(28, 25, 24, 32, 31, 27, 23, 30, 18, 18, 32, 22, 28, 31, 26)

n <- 15

meanx <- sum(x) / n
meany <- sum(y) / n

beta <- sum((y - meany) * (x - meanx)) / sum((x - meanx)^2)

epsilon <- y - meany - beta * (x - meanx)
variance_est <- sum(epsilon^2) / n

print(paste0('Sample mean x: ',meanx))
```

```
## [1] "Sample mean x: 23.0666666666667"
```

```r
print(paste0('Sample mean y(alpha): ',meany))
```

```
## [1] "Sample mean y(alpha): 26.3333333333333"
```

```r
print(paste0('Beta: ',beta))
```

```
## [1] "Beta: 0.506163615988046"
```

```r
print(paste0('SigmaSquared: ',variance_est))
```

```
## [1] "SigmaSquared: 14.1257626696551"
```

Here x denotes the social science score, which indicates the natural science score y. This gives us the least squares regression line:

$$y = 26.333 + 0.506(x - 23.067) = 14.658 + 0.506x$$

2.1 Part a

To find a 95% confidence interval we use the following equation:

$$\hat{\alpha} + \hat{\beta}(x - \overline{x}) + -t_{\alpha/2}\sqrt{\frac{n}{n-2}\hat{\sigma^2}(\frac{1}{n} + \frac{(x-\overline{x})^2}{\Sigma_{i=1}^{n}(x_i - \overline{x})^2})}$$

Looking up the value for $t_{0.025}$ in the t distribution table for n - 2 = 15 - 2 = 13 degrees of freedom, we get t = 2.160. We plug the values into our equation for each value of x we want to get a CI for and we get:

```
xVals <- c(17, 20, 23, 26, 29)

for (x_val in xVals) {
  y_hat <- meany + beta*(x_val - meanx)
  t <- 2.160
  sqrt_coefficient <- (n * variance_est) / (n-2) #n * sigmasquared / n-2
  sqrt_val <- (1 / n) + ((x_val - meanx)^2 / (sum((x - meanx)^2)))

  lower_bound <- y_hat - (t * sqrt(sqrt_coefficient * sqrt_val))
  upper_bound <- y_hat + (t * sqrt(sqrt_coefficient * sqrt_val))

  print(paste0('Lower bound for ',x_val,' is ',lower_bound))
  print(paste0('Upper bound for ',x_val,' is ',upper_bound))
}
```

```
## [1] "Lower bound for 17 is 19.6694485629091"
## [1] "Upper bound for 17 is 26.8557662297693"
## [1] "Lower bound for 20 is 22.1215422087615"
## [1] "Upper bound for 20 is 27.4406542798451"
## [1] "Lower bound for 23 is 24.047795485293"
## [1] "Upper bound for 23 is 28.5513826992419"
## [1] "Lower bound for 26 is 25.1907621389503"
## [1] "Upper bound for 26 is 30.4453977415129"
## [1] "Lower bound for 29 is 25.7911636962327"
## [1] "Upper bound for 29 is 32.8819778801588"
```

This makes the following Confidence Intervals:

- x=17 y=[19.669, 26.856]
- x=20 y=[22.122, 27.441]
- x=23 y=[24.048, 28.551]
- x=26 y=[25.191, 30.445]
- x=29 y=[25.791, 32.882]

2.2 Part b

For the prediction interval we do the same thing, but add in variance due to random error resulting in the following equation:

$$\hat{\alpha} + \hat{\beta}(x - \overline{x}) + -t_{\alpha/2}\sqrt{\frac{n}{n-2}\hat{\sigma^2}(1 + \frac{1}{n} + \frac{(x-\overline{x})^2}{\Sigma_{i=1}^{n}(x_i - \overline{x})^2})}$$

Looking up the value for $t_{0.05}$ in the t distribution table for n - 2 = 15 - 2 = 13 degrees of freedom, we get t = 1.771 We plug the values into our equation for each value of x we want to get a CI for and we get:

```
for (x_val in xVals) {
  y_hat <- meany + beta*(x_val - meanx)
  t <- 1.771
  sqrt_coefficient <- (n * variance_est) / (n-2) #n * sigmasquared / n-2

  # DIFFERENCE HERE FROM PART A
  sqrt_val <- 1 + (1 / n) + ((x_val - meanx)^2 / (sum((x - meanx)^2)))

  lower_bound <- y_hat - (t * sqrt(sqrt_coefficient * sqrt_val))
  upper_bound <- y_hat + (t * sqrt(sqrt_coefficient * sqrt_val))

  print(paste0('Lower bound for ',x_val,' is ',lower_bound))
  print(paste0('Upper bound for ',x_val,' is ',upper_bound))
}
```

```
## [1] "Lower bound for 17 is 15.5295632472784"
## [1] "Upper bound for 17 is 30.9956515453999"
## [1] "Lower bound for 20 is 17.3060940185538"
## [1] "Upper bound for 20 is 32.2561024700529"
## [1] "Lower bound for 23 is 18.9151865376899"
## [1] "Upper bound for 23 is 33.683991646845"
## [1] "Lower bound for 26 is 20.3507436982672"
## [1] "Upper bound for 26 is 35.285416182196"
## [1] "Lower bound for 29 is 21.6183574870175"
## [1] "Upper bound for 29 is 37.054784089374"
```

This makes the following Prediction Intervals:

- x=17 y=[15.530, 30.996]
- x=20 y=[17.306, 32.256]
- x=23 y=[18.915, 33.684]
- x=26 y=[20.351, 35.285]
- x=29 y=[21.618, 37.055]

3 Problem 8.3-1

3.1 Part a

Recall that α represents the probability of rejecting H_0 given that H_0 is true. We can find this result from our binomial distribution:

$$P(Y \leq 6|p = 0.08) = \Sigma_{k=0}^{6}(\frac{100!}{k! * (100-k)!} * 0.08^k * (1-0.08)^{(n-k)})$$

```
k <- seq(from=0, to=6, by=1)
n <- 100
p <- 0.08
```

```r
alpha <- sum((factorial(n) / (factorial(k) * factorial(n-k)))*p^k*(1-p)^(n-k))
paste0('Alpha value is: ', alpha)
```

```
## [1] "Alpha value is: 0.303155991468686"
```

The significance level α of the test is **0.3032**.

3.2 Part b

The probablity of a type two error is also denoted at β. Beta would be defined as the probability of no rejecting H_0 given that H_1 is true. We can find this result from our binomial distribution:

$$P(Y \geq 7|p = 0.04) = 1 - \Sigma_{k=0}^{6}(\frac{100!}{k! * (100-k)!} * 0.04^k * (1-0.04)^{(n-k)})$$

```r
k <- seq(from=0, to=6, by=1)
n <- 100
p <- 0.04

beta <- 1 - sum((factorial(n) / (factorial(k) * factorial(n-k)))*p^k*(1-p)^(n-k))
paste0('Beta value is: ', beta)
```

```
## [1] "Beta value is: 0.10639231790457"
```

The probability of a Type II error, if in fact p=0.04 is **0.1064**.

4 Problem 8.3-3

5 Problem 8.3-7

6 Problem 8.3-11