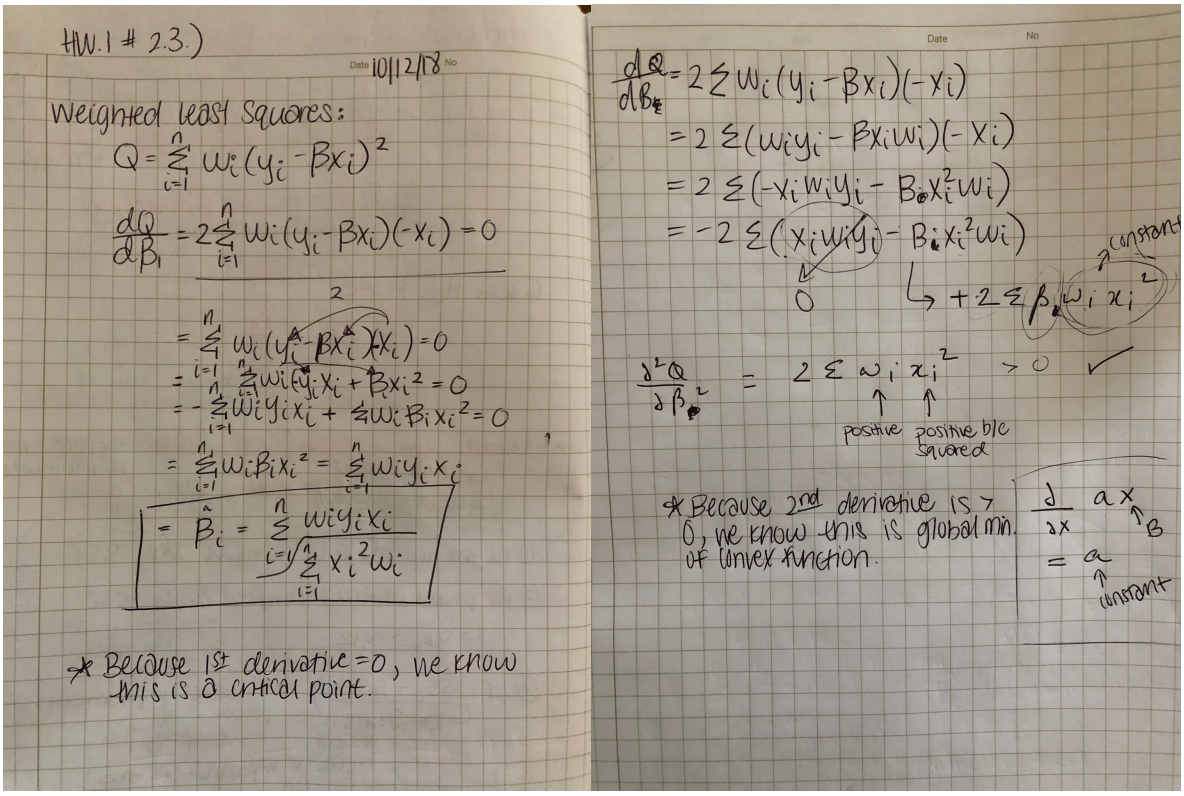
Naomi Kaduwela

Predictive Analytics HW #1: **Chapter 2, Exercises 2.3, 2.8, 2.9, 2.10, 2.11**

2.3 (Weighted least squares) Show that the **weighted least squares (WLS)** estimator of the slope β for regression through the origin obtained by minimizing the LS criterion Q



HW.1 # 2.3.)

Date 10/12/18 No

Weighted least Squares:

$$Q = \sum_{i=1}^{n} w_i (y_i - \beta x_i)^2$$

$$\frac{dQ}{d\beta_1} = 2\sum_{i=1}^{n} w_i (y_i - \beta x_i)(-x_i) = 0$$

$$= \sum_{i=1}^{n} w_i (y_i - \beta x_i)(x_i) = 0$$

$$= \sum_{i=1}^{n} w_i y_i x_i + \beta x_i^2 = 0$$

$$= -\sum_{i=1}^{n} w_i y_i x_i + \sum w_i \beta_i x_i^2 = 0$$

$$= \sum_{i=1}^{n} w_i \beta_i x_i^2 = \sum_{i=1}^{n} w_i y_i x_i$$

$$= \hat{\beta}_i = \frac{\sum_{i=1}^{n} w_i y_i x_i}{\sum_{i=1}^{n} x_i^2 w_i}$$

* Because 1st derivative = 0, we know this is a critical point.

Date No

$$\frac{dQ}{d\beta_i} = 2\sum w_i (y_i - \beta x_i)(-x_i)$$

$$= 2\sum (w_i y_i - \beta x_i w_i)(-x_i)$$

$$= 2\sum (-x_i w_i y_i - \beta_i x_i^2 w_i)$$

$$= -2\sum (x_i w_i y_i - \beta_i x_i^2 w_i)$$

$x_i w_i y_i$ → 0

$\beta_i x_i^2 w_i$ → $+2\sum \beta_i w_i x_i^2$ (β constant)

$$\frac{\partial^2 Q}{\partial \beta_i^2} = 2\sum w_i x_i^2 > 0 \checkmark$$

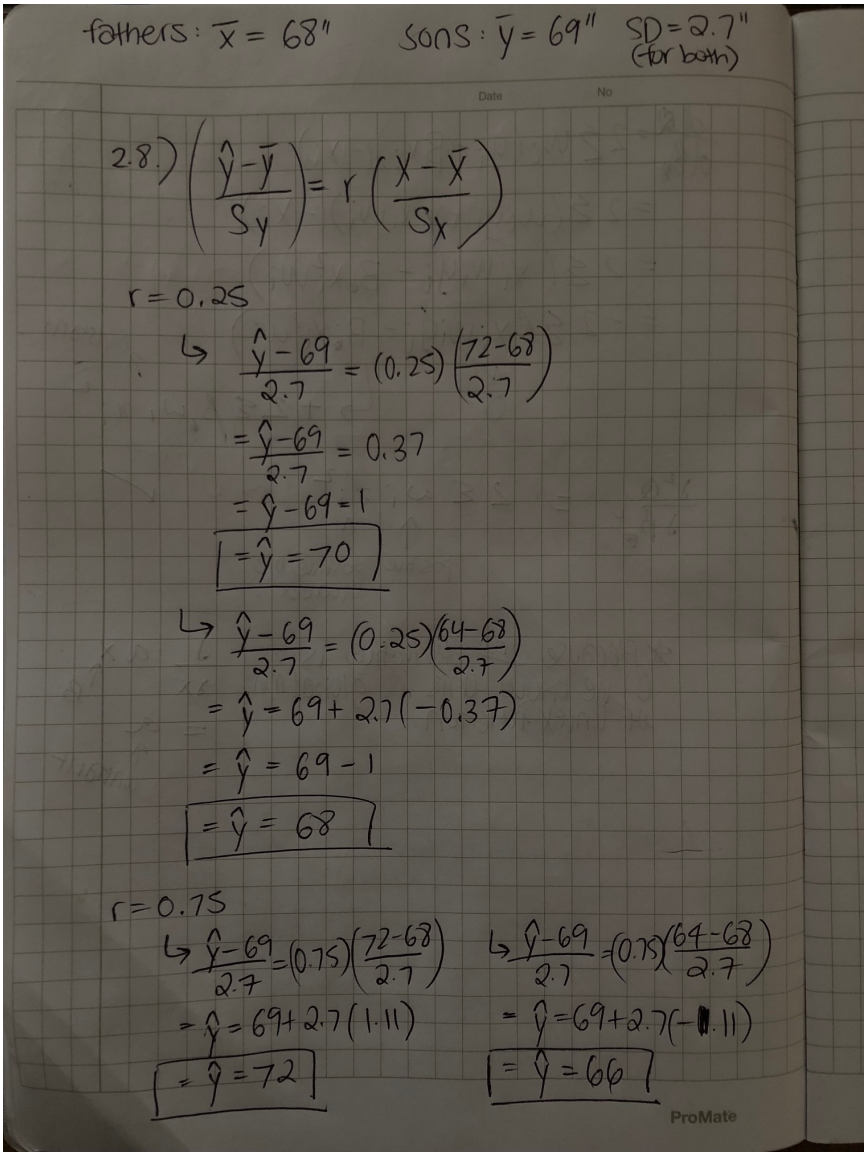
positive, positive b/c squared

* Because 2nd derivative is > 0, we know this is global min. of convex function.

$\frac{\partial}{\partial x} ax^B = a$ (a constant)

2.8 (Regression to the mean) Refer to Example 2.3. For the tall and short father considered in that example, calculate the expected heights of their sons for two different values of the correlation between the fathers' heights and sons' heights: r = 0.25 and r = 0.75. What do you conclude?

As r^2 increases, we see that the son is closer to the father's height than the mean



Predictive Analytics HW 1 CH 2

Naomi Kaduwela

10/12/2018

Chapter 2, Exercises 2.3, 2.8, 2.9, 2.10, 2.11

2.9

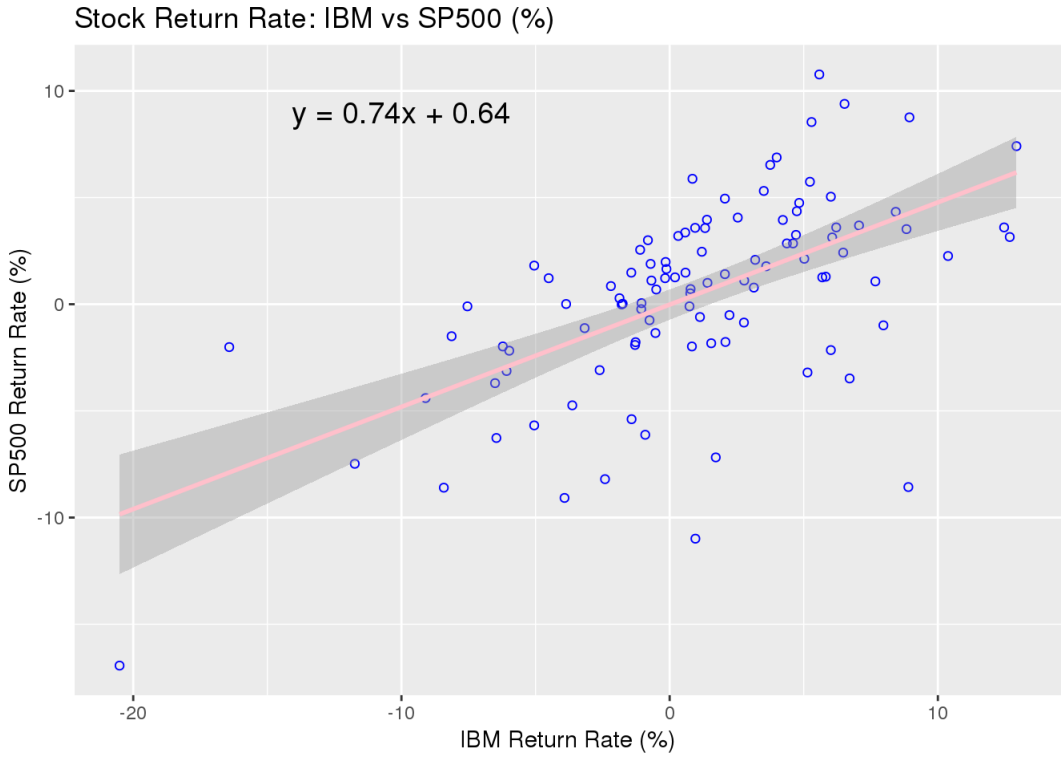
2.9a) Scatter plots of rates of return of IBM versus S&P 500 and Apple versus S&P 500 and comment on them.

From the regression line plotted in the graph, it is clear Apple has a higher y intercept - meaning it starts off higher - and has a bigger slope - meaning it will increase faster than IBM. Looking at the β coefficients, we also note that Apple will have more impact on the S&P 500 than IBM, as it has a larger coefficient.

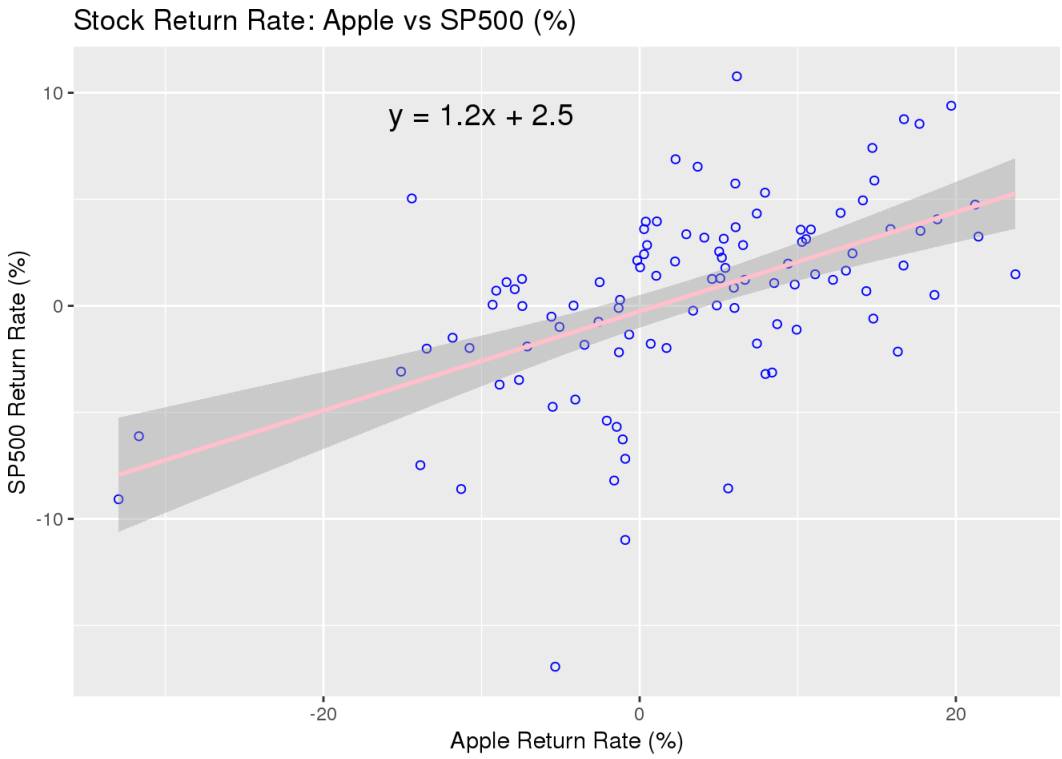
2.9b.) Please see linear model formula β's for IBM and Apple versus S&P 500 printed on chart output below

2.9c.) Calculate sample standard deviations. Calculate correltation matrix. Check β's = RSy/Sx

```
# Import and store data
IBM_apple_data <- read.csv("~/Predictive Analytics/IBM-Apple-SP500 RR Data.csv", stringsAsFactors = FALSE, s
kip = 1)

#Clean Data - Remove unnecessary columns, fix date format, remove % from numbers and convert to numeric form
IBM_apple_data_cleaned <- IBM_apple_data[,-5]
names(IBM_apple_data_cleaned)[2] <- "SP500"
IBM_apple_data_cleaned[, "Date"] <- as.Date(IBM_apple_data_cleaned[, "Date"])
IBM_apple_data_cleaned[, "SP500"] <- as.numeric(sub("%", "", IBM_apple_data_cleaned[, "SP500"], fixed=TRUE))
IBM_apple_data_cleaned[, "IBM"] <- as.numeric(sub("%", "", IBM_apple_data_cleaned[, "IBM"], fixed=TRUE))
IBM_apple_data_cleaned[, "Apple"] <- as.numeric(sub("%", "", IBM_apple_data_cleaned[, "Apple"], fixed=TRUE))

#View(IBM_apple_data_cleaned)
#summary(IBM_apple_data_cleaned)

#Regression Line - IBM
  regressionIBM <- lm(IBM_apple_data_cleaned$IBM ~ IBM_apple_data_cleaned$SP500)
  a <- signif(coef(regressionIBM)[1], digits = 2)
  b <- signif(coef(regressionIBM)[2], digits = 2)
  IBMLM <- paste("y = ",b,"x + ",a, sep="")

#PLOT - IBM
  ggplot(data = IBM_apple_data_cleaned) + aes(x = (IBM_apple_data_cleaned$IBM), y = (IBM_apple_data_cleaned$
SP500)) +
  geom_point(color='blue', shape =1) +
  geom_smooth(method = "lm", color ='pink', se = TRUE)+
  labs(title = "Stock Return Rate: IBM vs SP500 (%) ", x="IBM Return Rate (%)", y="SP500 Return Rate (%)") +
  annotate("text", x = -10, y = 9, label = IBMLM, color="black", size = 5, parse=FALSE)
```



```
#Regression Line - Apple
  regressionApple <- lm(IBM_apple_data_cleaned$Apple ~ IBM_apple_data_cleaned$SP500)
  c <- signif(coef(regressionApple)[1], digits = 2)
  d <- signif(coef(regressionApple)[2], digits = 2)
  appleLM <- paste("y = ",d,"x + ",c, sep="")

#PLOT - Apple
  ggplot(data = IBM_apple_data_cleaned) + aes(x = (IBM_apple_data_cleaned$Apple), y = (IBM_apple_data_cleane
d$SP500)) +
  geom_point(color='blue', shape=1) +
  geom_smooth(method = "lm", color ='pink', se = TRUE)+
  labs(title = "Stock Return Rate: Apple vs SP500 (%)", x="Apple Return Rate (%)", y="SP500 Return Rate (%)"
)+
  annotate("text", x = -10, y = 9, label = appleLM, color="black", size = 5, parse=FALSE)
```




```
# Calculate sample standard deviations (SD's) for SP500, IBM, Apple

  paste("Apple SD= ", sd_apple<- round(sd(IBM_apple_data_cleaned$Apple),2))
```

```
## [1] "Apple SD=  10.31"
```

```
  paste("IBM SD= ", sd_IBM <- round(sd(IBM_apple_data_cleaned$IBM),2))
```

```
## [1] "IBM SD=  5.56"
```

```
  paste("SP500 SD= ", sd_SP500 <- round(sd(IBM_apple_data_cleaned$SP500),2))
```

```
## [1] "SP500 SD=  4.46"
```

```
# Calculate Correlation Matrix
  corrr <- cor(IBM_apple_data_cleaned[, c(2,3,4)])
  print(corrr)
```

```
##           SP500       IBM     Apple
## SP500 1.0000000 0.5974779 0.5382317
## IBM   0.5974779 1.0000000 0.4147253
## Apple 0.5382317 0.4147253 1.0000000
```

```
#Calculate R value
  rIBM <- cor(IBM_apple_data_cleaned$SP500, IBM_apple_data_cleaned$IBM)
  rApple <- cor(IBM_apple_data_cleaned$SP500, IBM_apple_data_cleaned$Apple)

# Check β's = RSy/Sx :
#   r = correlation coeff between SP500 and given stock (IBM, Apple)
#   Sx = sample SD of SP500
#   Sy = sample SD of given stock

# Check IBM β's = RSy/Sx
  if (b == round(rIBM * (sd_IBM / sd_SP500),2)){
    print("IBM Check: Pass")
  }else{
    print("IBM Check: Fail")
  }
```

```
## [1] "IBM Check: Pass"
```

```
# Check Apple β's = RSy/Sx
  if (d == round(rApple * (sd_apple / sd_SP500),1)){
      print("Apple Check: Pass")
  }else{
    print("Apple Check: Fail")
  }
```

```
## [1] "Apple Check: Pass"
```

2.9 d.) Explain based on the statistics calculated how a higher expected return is accompanied by higher volatility of the stock relative to S&P 500.

We see that the SD of Apple (SD = 10.31) is almost double that of IBM (SD = 5.56).

Looking at β's = RSy/Sx, Sy = sample SD of given stock (IBM and Apple). As Sy is in the numerator, a larger SD will result in a larger β

2.10

a.) Estimate the price elasticities of all three steaks. Given that chuck is the least expensive cut and rib eye is the most expensive cut of beef, are the price elasticities of the three cuts in the expected order?

Expensive Scale: Chuck –> PortHse –> RibEye

Steak Estimate: Chuck: -1.3687, PortHse Estimate: -2.6565, RibEye: -1.446

Price elasicty is the change in demand based on the change in price. The higher the price elasticty, the more sensivite the demand based on the price change. The coefficients are not in the expected order, as the PortHse - which is the steak priced in the middle - has the largest coefficient, meaning it would have the highest price elasticity. In theory it would make more sense that the RibEye would be more sensitive.

```
#Read in data file
steakPrice <- read_csv("~/Predictive Analytics/Steak+Prices.CSV")
```

```
## Parsed with column specification:
## cols(
##   Year = col_integer(),
##   Month = col_integer(),
##   `Chuck-Qty` = col_integer(),
##   `Chuck-Price` = col_character(),
##   `PortHse-Qty` = col_integer(),
##   `PortHse-Price` = col_character(),
##   `RibEye-Qty` = col_integer(),
##   `RibEye-Price` = col_character()
## )
```

```
#View(steakPrice)

#Format columns: remove $
steakPrice$`Chuck-Price`<- str_remove(steakPrice$`Chuck-Price` ,'\\$')
steakPrice$`PortHse-Price`<- str_remove(steakPrice$`PortHse-Price`, '\\$')
steakPrice$`RibEye-Price` <- str_remove(steakPrice$`RibEye-Price`, '\\$')

#Format columns: change to numeric
steakPrice$`Chuck-Price` <- as.numeric(steakPrice$`Chuck-Price`)
steakPrice$`PortHse-Price`<- as.numeric(steakPrice$`PortHse-Price`)
steakPrice$`RibEye-Price` <-as.numeric(steakPrice$`RibEye-Price`)
#print(steakPrice)

#Price Elasticity - convert to: lny = lna + blnx --> b= [dy/y] / [dx/x]
#Chuck
regressionChuck <- lm((log(steakPrice$'Chuck-Qty')) ~ (log(steakPrice$'Chuck-Price')), steakPrice)
summary(regressionChuck)
```

```
##
## Call:
## lm(formula = (log(steakPrice$"Chuck-Qty")) ~ (log(steakPrice$"Chuck-Price")),
##     data = steakPrice)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.32463 -0.12036 -0.01714  0.09430  0.49725
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     5.8899     0.2871  20.513  < 2e-16 ***
## log(steakPrice$"Chuck-Price")  -1.3687     0.3199  -4.278 9.44e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1812 on 46 degrees of freedom
## Multiple R-squared:  0.2846, Adjusted R-squared:  0.2691
## F-statistic:  18.3 on 1 and 46 DF,  p-value: 9.441e-05
```

```
#Port Hse
regressionPortHse <- lm((log(steakPrice$'PortHse-Qty')) ~ (log(steakPrice$'PortHse-Price')), steakPrice)
summary(regressionPortHse)
```

```
##
## Call:
## lm(formula = (log(steakPrice$"PortHse-Qty")) ~ (log(steakPrice$"PortHse-Price")),
##     data = steakPrice)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.57655 -0.23544  0.00317  0.23511  0.49991
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      9.1123     0.5136  17.742  < 2e-16 ***
## log(steakPrice$"PortHse-Price")  -2.6565     0.2752  -9.654 1.23e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.283 on 46 degrees of freedom
## Multiple R-squared:  0.6695, Adjusted R-squared:  0.6624
## F-statistic:  93.2 on 1 and 46 DF,  p-value: 1.233e-12
```

```
#RibEye
regressionRibEye <- lm((log(steakPrice$'RibEye-Qty')) ~ (log(steakPrice$'RibEye-Price')), steakPrice)
summary(regressionRibEye)
```

```
##
## Call:
## lm(formula = (log(steakPrice$"RibEye-Qty")) ~ (log(steakPrice$"RibEye-Price")),
##     data = steakPrice)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.54075 -0.21801  0.03995  0.20328  0.70950
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     7.6627     0.7537  10.167 2.39e-13 ***
## log(steakPrice$"RibEye-Price")  -1.4460     0.3731  -3.876 0.000335 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2943 on 46 degrees of freedom
## Multiple R-squared:  0.2462, Adjusted R-squared:  0.2298
## F-statistic: 15.02 on 1 and 46 DF,  p-value: 0.0003352
```

```
#Price Elasticity - calculate demand if price changes by 10%

#Chuck
Chuck_elasticity <- as.numeric(regressionChuck$coefficients[2])
paste("Chuck",Chuck_elasticity * 10)
```

```
## [1] "Chuck -13.6866509356365"
```

```
#Port Hse
Port_elasticity <- as.numeric(regressionPortHse$coefficients[2])
paste("Port Hse",Port_elasticity * 10)
```

```
## [1] "Port Hse -26.5648730236326"
```

```
#Ribeye
Ribeye_elasticity <- as.numeric(regressionRibEye$coefficients[2])
paste("Ribeye",Ribeye_elasticity * 10)
```

```
## [1] "Ribeye -14.4600367651748"
```

2.10 b) Estimate how much the demand will change if the price is increased by 10% for each cut.

The demand will change by the price elasticity * 10.

Chuck = -13.687

Port Hse = -26.565

Ribeye = -14.46

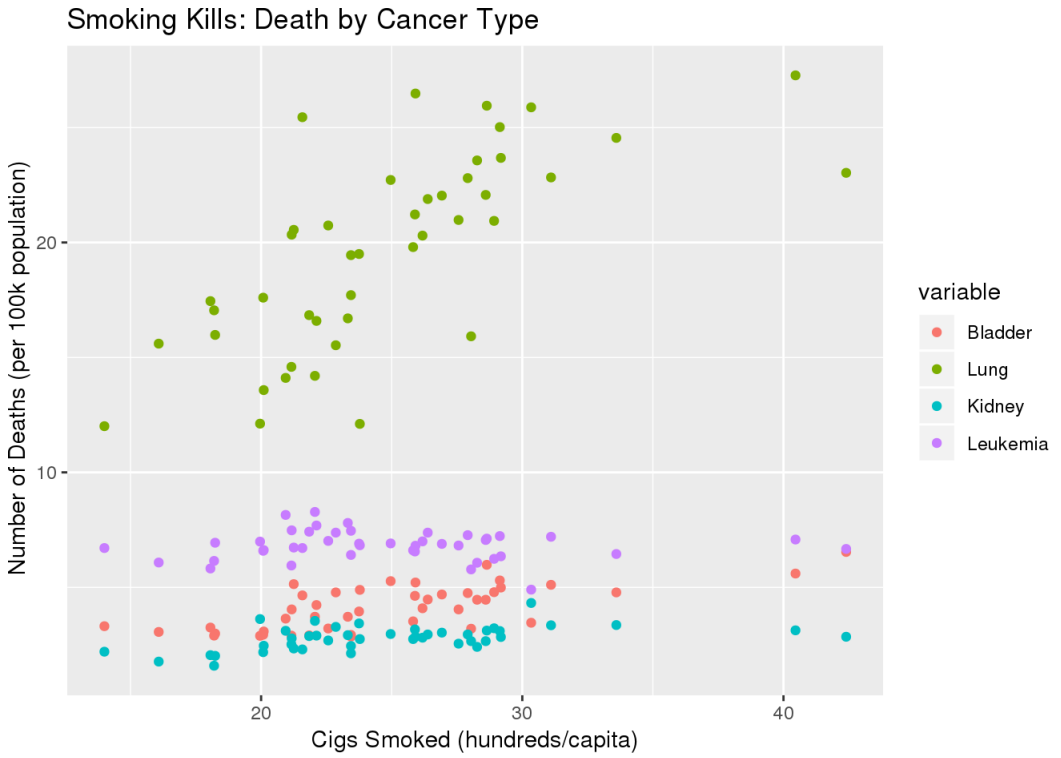
2.11

2.11a.) Scatterplot of deaths due to each type of cancer vs cigs smoked to see what relationships exist and if there are outlieres

It does seem to be linear. No obvious visual outliers.

2.11b.) Perform tests on the correlations to see which type of cancer deaths are most significantly correlated with cig smoking. Bladder has the highest correlation coefficient.

```
  #Read in data
  smokingData <- read.csv("~/Predictive Analytics/smoking-cancer.csv", stringsAsFactors = FALSE)
  #View(smokingData)
  #summary(smokingData)

  #PLOT
  smokingDataLong <- melt(smokingData, id = "Smoke", measure = c("Bladder", "Lung", "Kidney", "Leukemia"))
  ggplot(smokingDataLong, aes(Smoke, value, color = variable)) +
  geom_point(aes(color = factor(variable)))+
  guides(fill = guide_legend(title = "Types of Cancer", title.position = "top")) +
  labs(title = "Smoking Kills: Death by Cancer Type  ", x="Cigs Smoked (hundreds/capita)", y=" Number of Dea
ths (per 100k population)")
```




```
#Calculate Correlation Coefficients - Lung
corrLung <- cor(smokingData$Smoke, smokingData$Lung)
paste("Lung: ", corrLung)
```

```
## [1] "Lung:  0.697402504927529"
```

```
#Calculate Correlation Coefficients - Kidney
corrKidney <- cor(smokingData$Smoke, smokingData$Kidney)
paste("Kidney: ", corrKidney)
```

```
## [1] "Kidney:  0.487389617033565"
```

```
#Calculate Correlation Coefficients - Bladder
corrBladder <- cor(smokingData$Smoke, smokingData$Bladder)
paste("Bladder: ", corrBladder)
```

```
## [1] "Bladder:  0.703621859461442"
```

```
#Calculate Correlation Coefficients - Leukemia
corrLeukemia <- cor(smokingData$Smoke, smokingData$Leukemia)
paste("Leukemia: ", corrLeukemia)
```

```
## [1] "Leukemia:  -0.068481229476639"
```

```
#Look at p values for each correlation cancer with smoking to see significance level

#Lung
corLung <- regressionRibEye <- lm(smokingData$Smoke~ smokingData$Lung, smokingData)
summary(corLung)
```

```
##
## Call:
## lm(formula = smokingData$Smoke ~ smokingData$Lung, data = smokingData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.6630 -2.8167 -0.0832  1.4357 14.3817
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)         6.8473     2.9289   2.338   0.0242 *
## smokingData$Lung    0.9193     0.1458   6.306 1.44e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.042 on 42 degrees of freedom
## Multiple R-squared:  0.4864, Adjusted R-squared:  0.4741
## F-statistic: 39.77 on 1 and 42 DF,  p-value: 1.439e-07
```

```
#Kidney
corKidney <- regressionRibEye <- lm(smokingData$Smoke~ smokingData$Kidney, smokingData)
summary(corKidney)
```

```
##
## Call:
## lm(formula = smokingData$Smoke ~ smokingData$Kidney, data = smokingData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.2737 -2.9842 -0.8092  2.0627 17.1957
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)          10.290      4.110   2.504 0.016270 *
## smokingData$Kidney    5.233      1.447   3.617 0.000792 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.924 on 42 degrees of freedom
## Multiple R-squared:  0.2375, Adjusted R-squared:  0.2194
## F-statistic: 13.09 on 1 and 42 DF,  p-value: 0.0007922
```

```
#Bladder
corBladder <- regressionRibEye <- lm(smokingData$Smoke~ smokingData$Bladder, smokingData)
summary(corBladder)
```

```
##
## Call:
## lm(formula = smokingData$Smoke ~ smokingData$Bladder, data = smokingData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.8048 -3.2558  0.0011  2.0259  9.5358
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)           8.1657     2.6789   3.048  0.00397 **
## smokingData$Bladder   4.0640     0.6333   6.417 9.96e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.007 on 42 degrees of freedom
## Multiple R-squared:  0.4951, Adjusted R-squared:  0.4831
## F-statistic: 41.18 on 1 and 42 DF,  p-value: 9.964e-08
```

```
#Leukemia
corLeukemia <- regressionRibEye <- lm(smokingData$Smoke~ smokingData$Leukemia, smokingData)
summary(corLeukemia)
```

```
##
## Call:
## lm(formula = smokingData$Smoke ~ smokingData$Leukemia, data = smokingData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.986  -3.368  -1.056   2.991  17.390
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)            28.998      9.220   3.145  0.00305 **
## smokingData$Leukemia   -0.598      1.344  -0.445  0.65871
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.626 on 42 degrees of freedom
## Multiple R-squared:  0.00469,    Adjusted R-squared:  -0.01901
## F-statistic: 0.1979 on 1 and 42 DF,  p-value: 0.6587
```

2.11 d.) Which cancers are significantly related with smoking?

Looking at the p values we see that they are all significant except Leukemia

Lung = 1.439e-07

Kidney = 0.0007922

Bladder = 9.96e-08

Leukemia = 0.6587