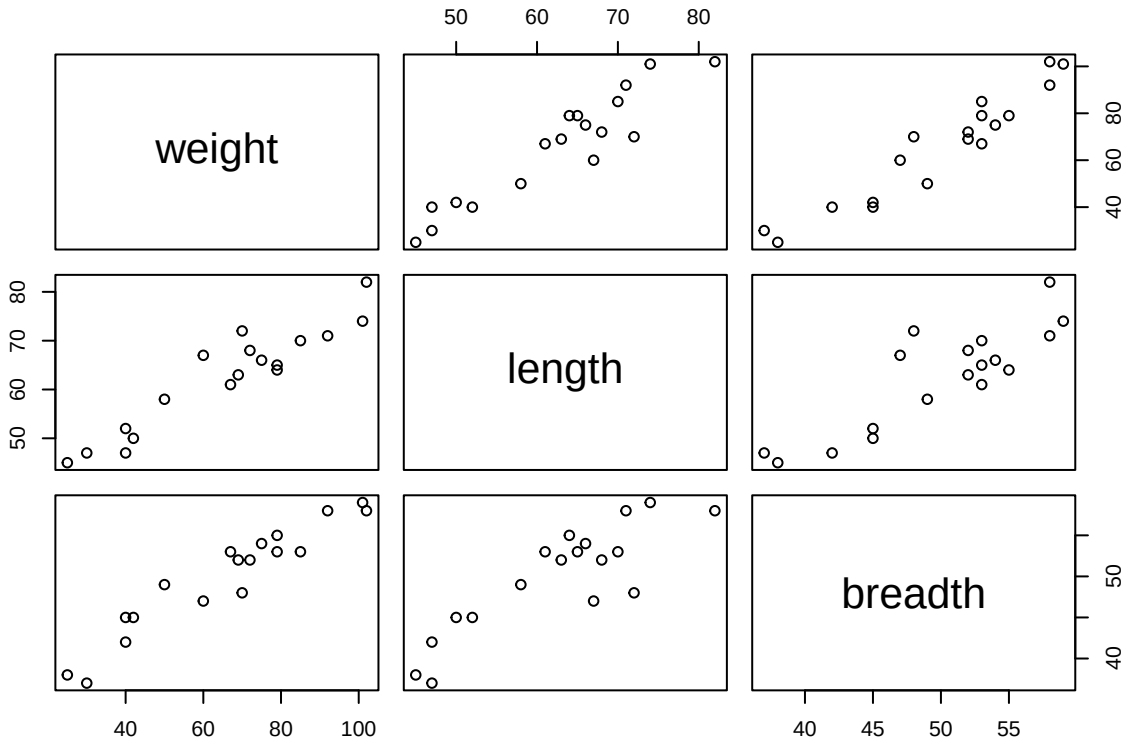
lab4math158

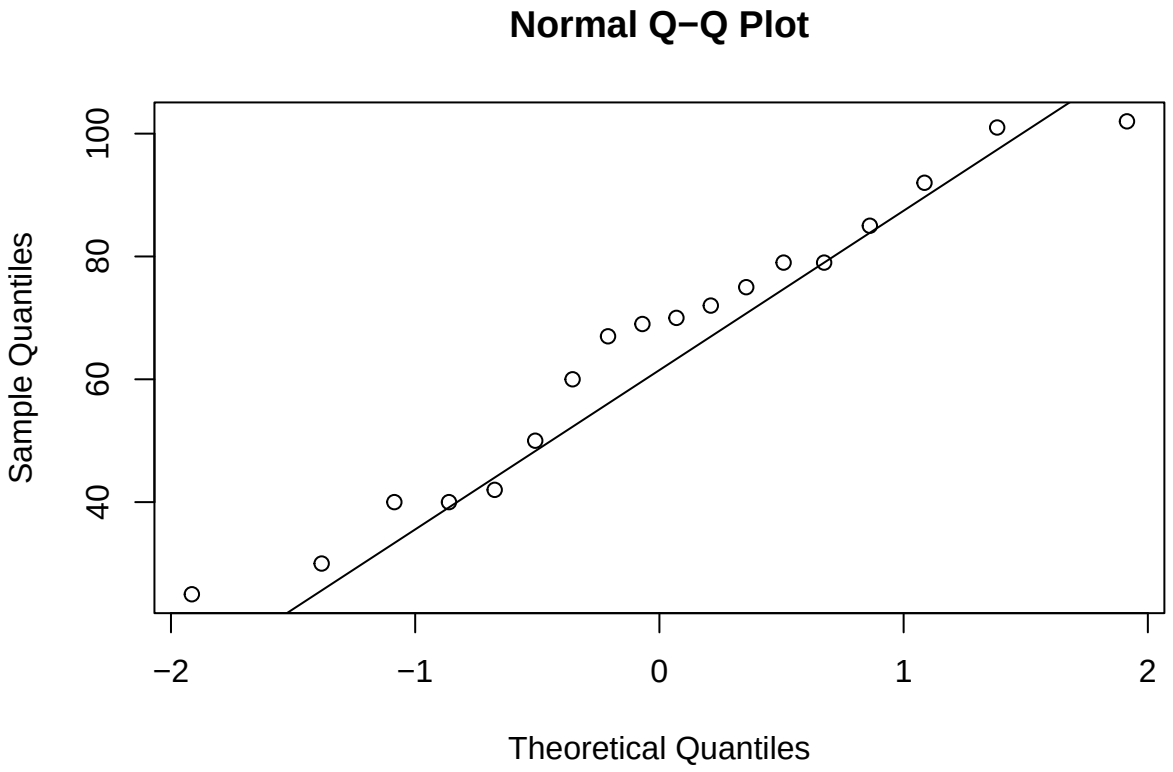
Joshua Jansen-Montoya

2022-10-25

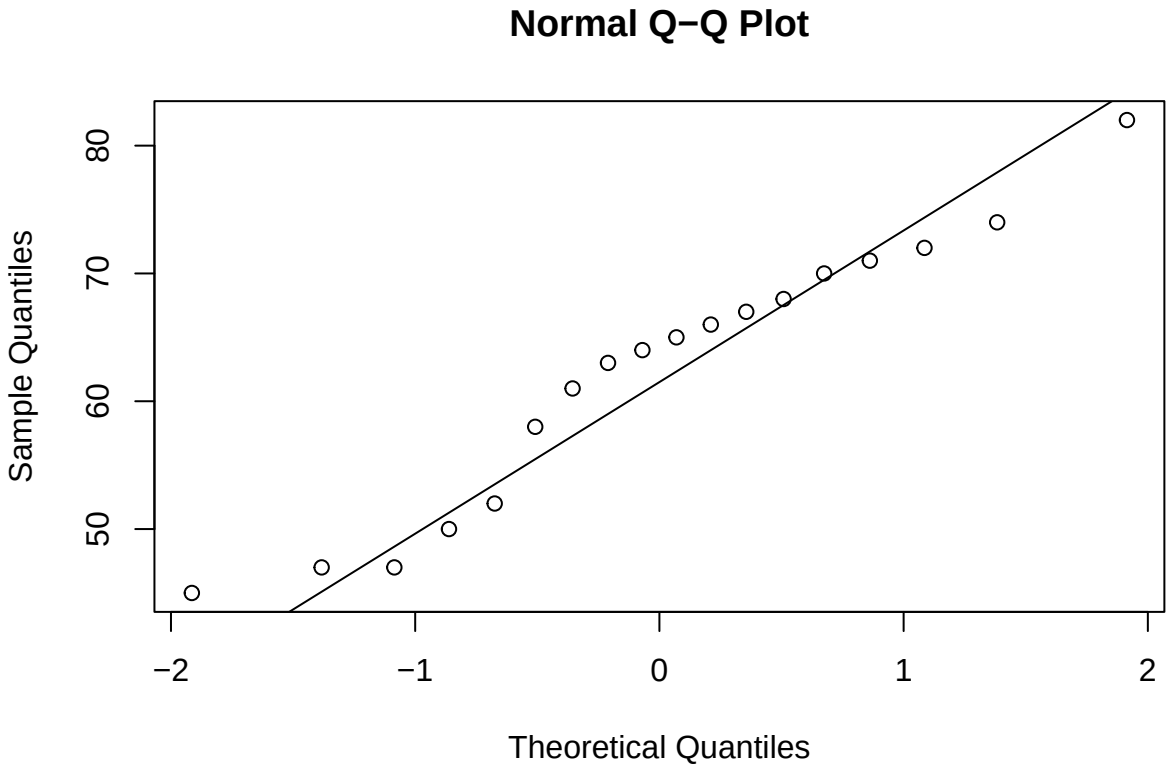
```
potato <- read.table("potato", header=T)
plot(potato)
```



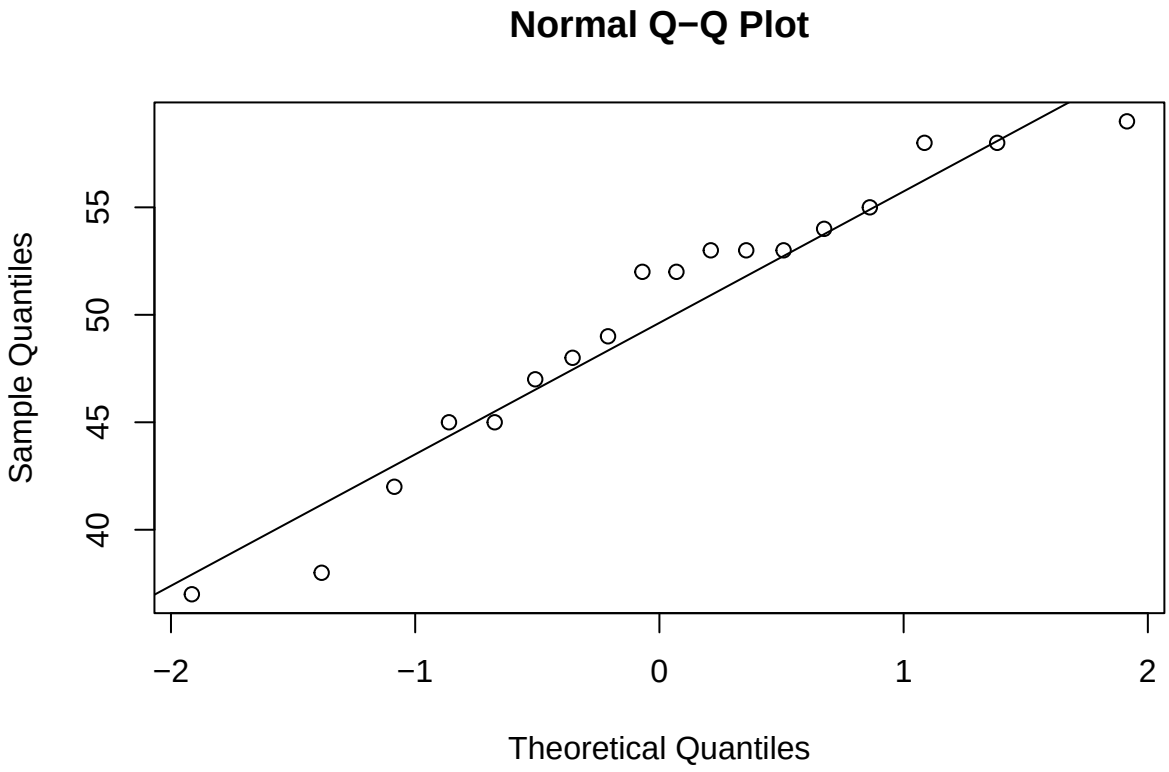

```
qqnorm(potato$weight)
qqline(potato$weight)
```




```
qqnorm(potato$length)
qqline(potato$length)
```




```
qqnorm(potato$breadth)
qqline(potato$breadth)
```




```
  shapiro.test(potato$weight)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  potato$weight
## W = 0.9568, p-value = 0.5411
```

```
  shapiro.test(potato$length)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  potato$length
## W = 0.94898, p-value = 0.4092
```

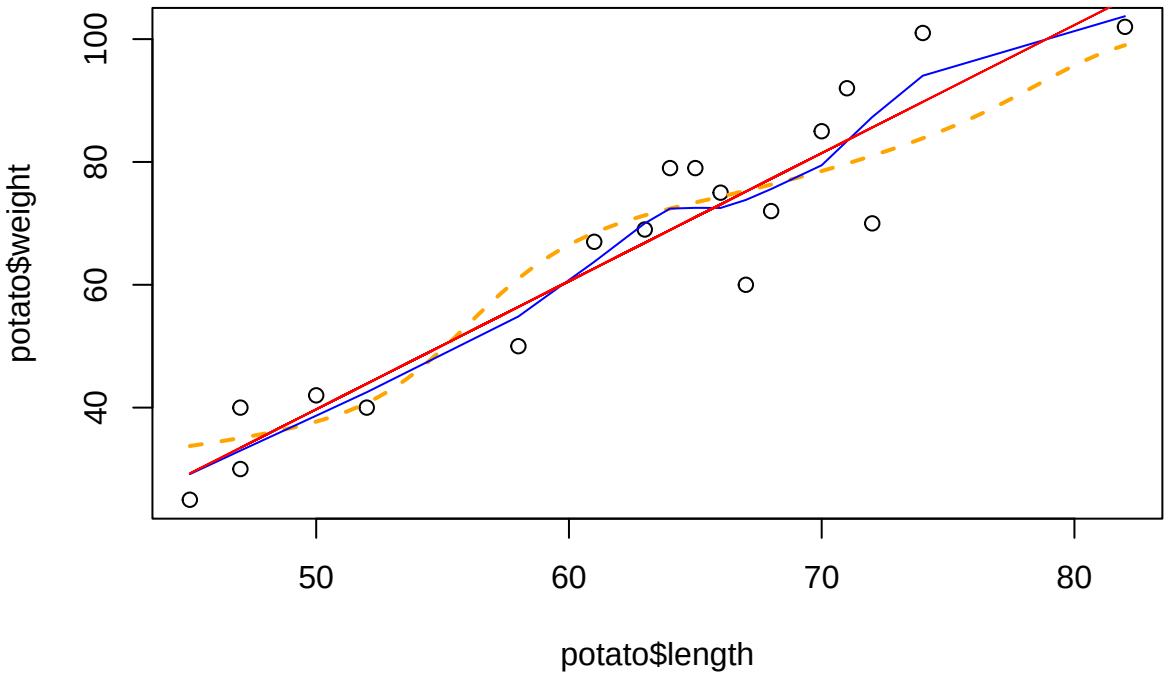
```
  shapiro.test(potato$breadth)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  potato$breadth
## W = 0.94066, p-value = 0.2974
```

Looking at these plots, it appears that there exists linear relationships between each of our variables, and thus, it would make sense that a linear model could be used to summarize these relationships. Similarly, going off of our Shapiro-Wilks test and from our qqnorm plots, we can see that each of our columns in our data set are normal and thus sufficiently meet the pre-requisite for our linear models. Now, we can use the models as follows,

```
  plot(potato$length,potato$weight)
  lines(ksmooth(potato$length,potato$weight,"normal",bandwidth=12),lty=2,lwd=2,col="orange")
```

```
mm <-loess(potato$weight~ potato$length,span=0.55,degree=1)
lines(sort(potato$length),mm$fit[sort.list(potato$length)],col="blue")
mm <-lm(weight ~ length,data=potato)
lines(potato$length,mm$fit,col="red")
```



For the bandwidth, we chose the value bandwidth = 12 since it provided a moderately linear fit along with demonstrating some of the non-linear movements to the data. We felt that this most accurately reflected the trends in our data. Similarly with the span = 0.55, we felt that this choice also did well to show that the data was not perfectly linear, but that it would trend well with a linear model fit. This makes us more confident about moving onto linear models for weight as a functin of length and a function of breadth since we were able to find numbers for span and breadth that would mimic those of a linear function that were not choices that would otherwise force a linear shape to our data. Thus, we can see that there is a an underlying linear structure. This may have been an unnecessary data transformation since we could also see the underlying linear structure through the linear model and by inspecting the R^2 value. ## Ordinary Least Squares Now, we can solve our OLS by hand as follows,

```
X <-cbind(1,potato$length,potato$breadth)
dim(X)
```

```
## [1] 18  3
```

```
coeff <-solve(t(X)%*%X)%*%(t(X)%*%potato$weight)
coeff
```

```
##             [,1]
## [1,] -97.383549
## [2,]   1.012151
## [3,]   1.999188
```

```
lmod <- lm(weight ~ length + breadth, data = potato)
summary(lmod)
```

```
##
## Call:
## lm(formula = weight ~ length + breadth, data = potato)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.2814 -3.0796 -0.3766  3.8902  5.8466
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -97.3835     8.7997 -11.067 1.30e-08 ***
## length        1.0122     0.2129   4.754 0.000256 ***
## breadth       1.9992     0.3425   5.838 3.27e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.646 on 15 degrees of freedom
## Multiple R-squared:  0.9649, Adjusted R-squared:  0.9603
## F-statistic: 206.3 on 2 and 15 DF,  p-value: 1.223e-11
```

```
  diagonals = diag(X)
  diagonals
```

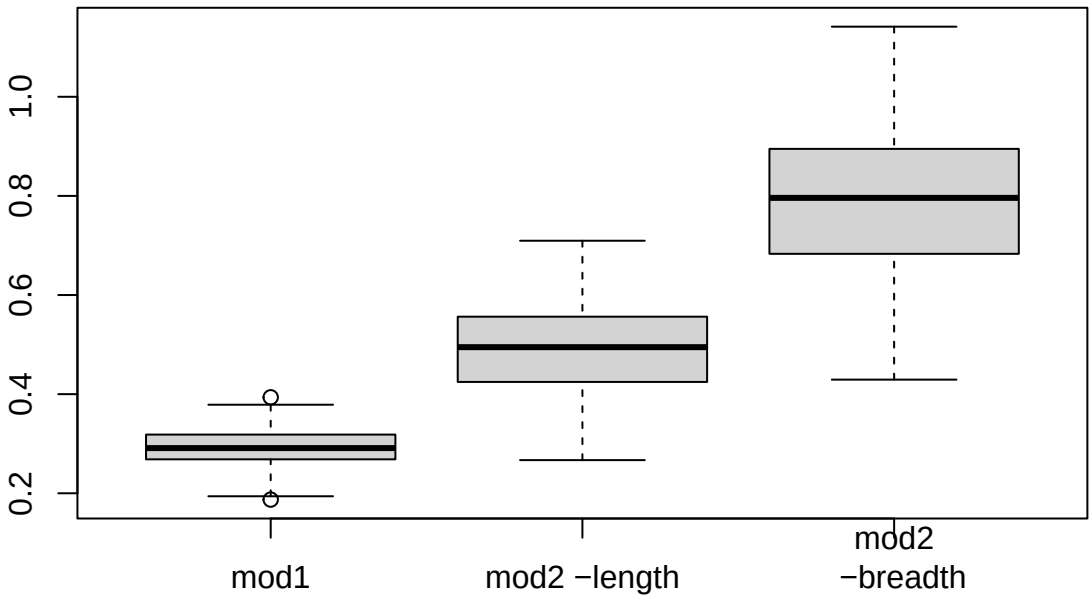
```
## [1]  1 52 47
```

We can see that the coefficients for both of our methods of calculation agree with one another. Looking at our diagonals, we can see that we obtain square rooted value of 1, $\sqrt{52}$, and $\sqrt{47}$. We can note that these are not the same as what we had found in our linear model. The off-diagonal elements represent the covariance in our model.

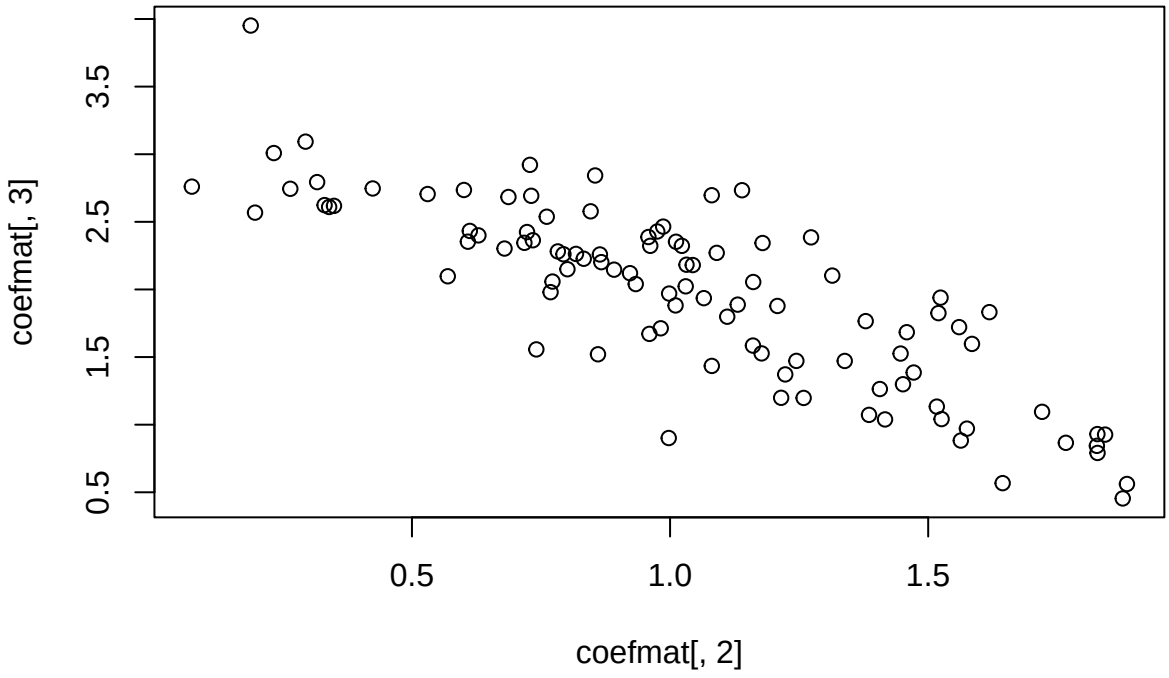
Colinearity

Using the given code, we can see,

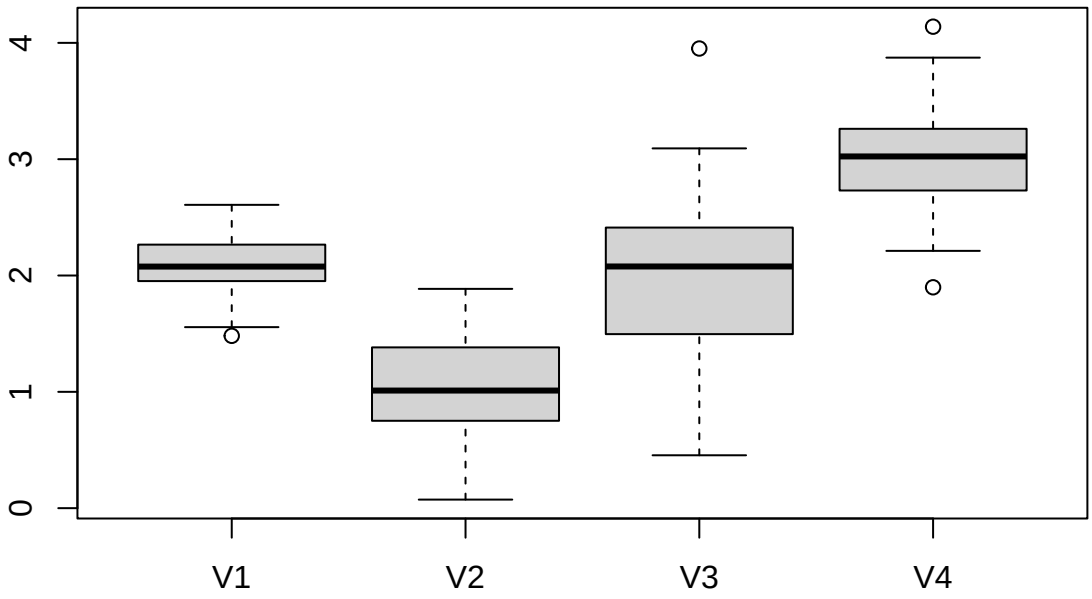
```
  coefmat <-matrix(0,100,3)
  sdcoefmat <-matrix(0,100,3)
  # to record the 3 set of coefficient estimates of
  # model 1 (1 coefficient) and model 2 (2 coefficients).
  for (kk in (1:100)) {
  newerror <-rnorm(18,sd=10)
  mod1 <-summary(lm(potato$weight+newerror ~ potato$length))
  mod2 <-summary(lm(potato$weight+newerror ~ potato$length+potato$breadth))
  coefmat[kk,1] <-mod1$coef[2,1]
  coefmat[kk,2:3] <-mod2$coef[2:3,1]
  sdcoefmat[kk,1] <-mod1$coef[2,2]
  sdcoefmat[kk,2:3] <-mod2$coef[2:3,2] }
  # a boxplot of the standard errors for the model coefficients in model 1 and
  # 2 respectively
  boxplot(as.data.frame(sdcoefmat), names=c("mod1", "mod2 -length", "mod2
  -breadth"))
```




```
# Relationship between model coefficients "length" and "breadth" in model 2.
plot(coefmat[,2],coefmat[,3])
```

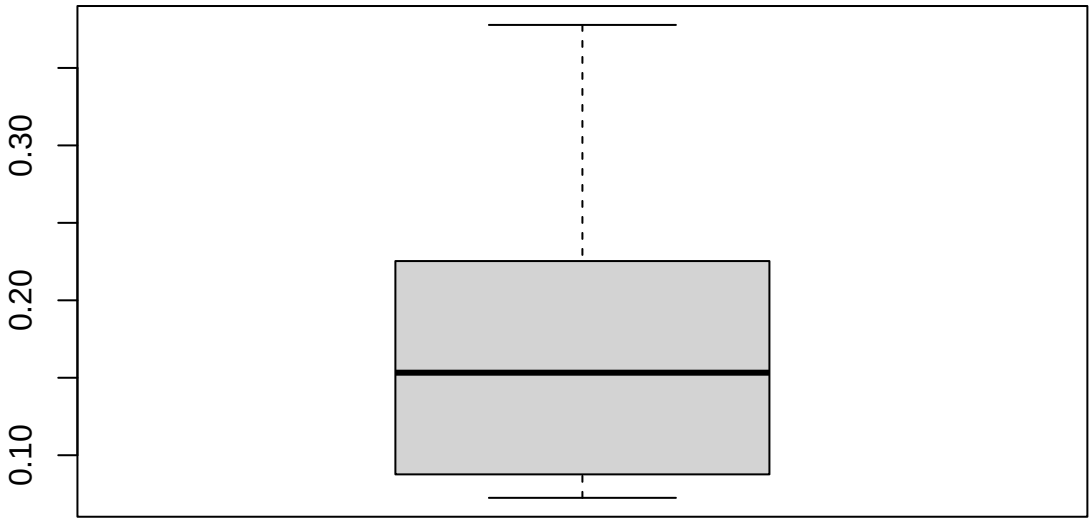



```
# Boxplots of the coefficients in model 1, model 2 (length and breadth), and
# the sum of the 2 coefficients in model 2.
boxplot(as.data.frame(cbind(coefmat,coefmat[,2]+coefmat[,3])))
```



Thus, we can see that there appears to be collinearity between the two coefficients as indicated by the scatter plot, but that there does not seem to be major overlap between the coefficients we generated across our models. ## Leverage We can identify the points with high leverage using the following code,

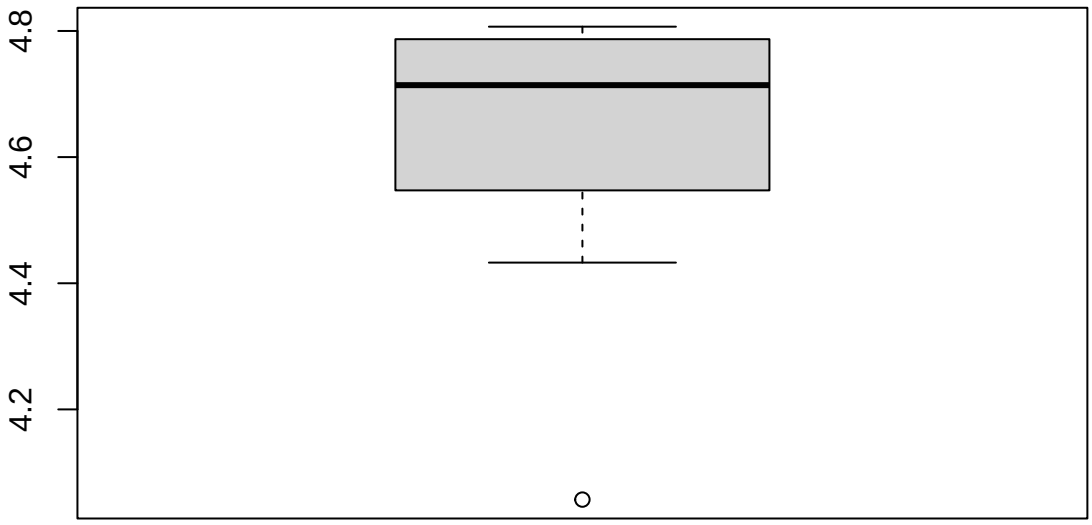
```
influence <- lm.influence(lmod, do.coef = TRUE)
?lm.influence
boxplot(influence$hat)
```




```
influence$hat > 2*mean(influence$hat)
```

```
##     1     2     3     4     5     6     7     8     9    10    11    12    13
##  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##    14    15    16    17    18
## FALSE FALSE FALSE FALSE FALSE
```

```
boxplot(influence$sigma)
```




```
  influence$sigma[abs(influence$sigma) > 2]
```

```
##        1        2        3        4        5        6        7        8
## 4.783546 4.575226 4.620121 4.806803 4.432735 4.057018 4.786990 4.801311
##        9       10       11       12       13       14       15       16
## 4.633016 4.547260 4.487960 4.714788 4.713391 4.794394 4.796409 4.762668
##       17       18
## 4.784822 4.525935
```

Looking at the hat values, we can see that observations 13, 14, 24, 25, 26, and 27 are all points that have high leverage, that we do not seem to have any outliers that correspond to residual standard deviations of greater than 2, and because of this, we do not have any influential points. Now, removing the observations we found to have high leverage, we obtain that,

```
  reducedPotato <- potato[-c(13, 14, 24, 25, 26, 27), ]
  reducedPotatoMod <- lm(weight ~ length + breadth, data = reducedPotato)
  summary(reducedPotatoMod)
```

```
##
## Call:
## lm(formula = weight ~ length + breadth, data = reducedPotato)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.7632 -3.0874  0.5823  4.3454  5.9542
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -98.5836     9.3352 -10.560 9.52e-08 ***
## length        0.9555     0.2335   4.091  0.00127 **
## breadth       2.1006     0.3796   5.533 9.65e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.867 on 13 degrees of freedom
## Multiple R-squared:  0.9666, Adjusted R-squared:  0.9614
## F-statistic:   188 on 2 and 13 DF,  p-value: 2.546e-10
```

We can see that we obtain a slightly better R^2, and similarly, that our coefficients change by ≈ 0.2, and that

we lose a level of significance for our length predictor variable, but that the Intercept and breadth remain highly significant to the overall model.

Testing

Now, we can construct the 95% CI on our original model as follows,

```
  confint(lmod, level=0.95)
```

```
##                    2.5 %     97.5 %
## (Intercept) -116.1396053 -78.627494
## length         0.5583736   1.465928
## breadth        1.2692455   2.729130
```

Looking at our confidence intervals, we can see that it appears that if we were to drop one of our predictors, it would be length since it contributes less to our overall model. Doing so, we can compare a reduced model with our main model as follows,

```
  reducedMod <- lm(weight ~ breadth, data = reducedPotato)
  anova(reducedMod, reducedPotatoMod)
```

```
## Analysis of Variance Table
##
## Model 1: weight ~ breadth
## Model 2: weight ~ length + breadth
##   Res.Df    RSS Df Sum of Sq      F   Pr(>F)
## 1     14 704.34
## 2     13 307.91  1    396.43 16.737 0.001274 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Looking at our results, we can see that as we have a Pr(>F) of less than 0.05, we can conclude that our reduced model is a better fit and that therefore, we can use our reduced model that does not include breadth as a predictor variable.

Summary

From the our statistical analysis of the potato dataset, we were able to find that collinearity was a problem within our the predictor variables within our dataset from which, we were able to analyze our data set further through using anova and different linear models to conclude that we could use a reduced model using just breadth as a predictor variables instead of the full model. We were also able to find that we could indeed use a linear model, as the requirements for a linear model that we needed were met by all parts of our data set including normality as shown by the Shapiro-Wilks test. We were also able to note that there were a couple of points of high leverage that after removing, we were able to construct a better fitting linear model. We did not find any clear violations of our assumptions for the linear model and thus, we trust in our results from the linear model.