DATA 603 HW 3

Kane Smith

2022-11-29

Contents

Problem 1

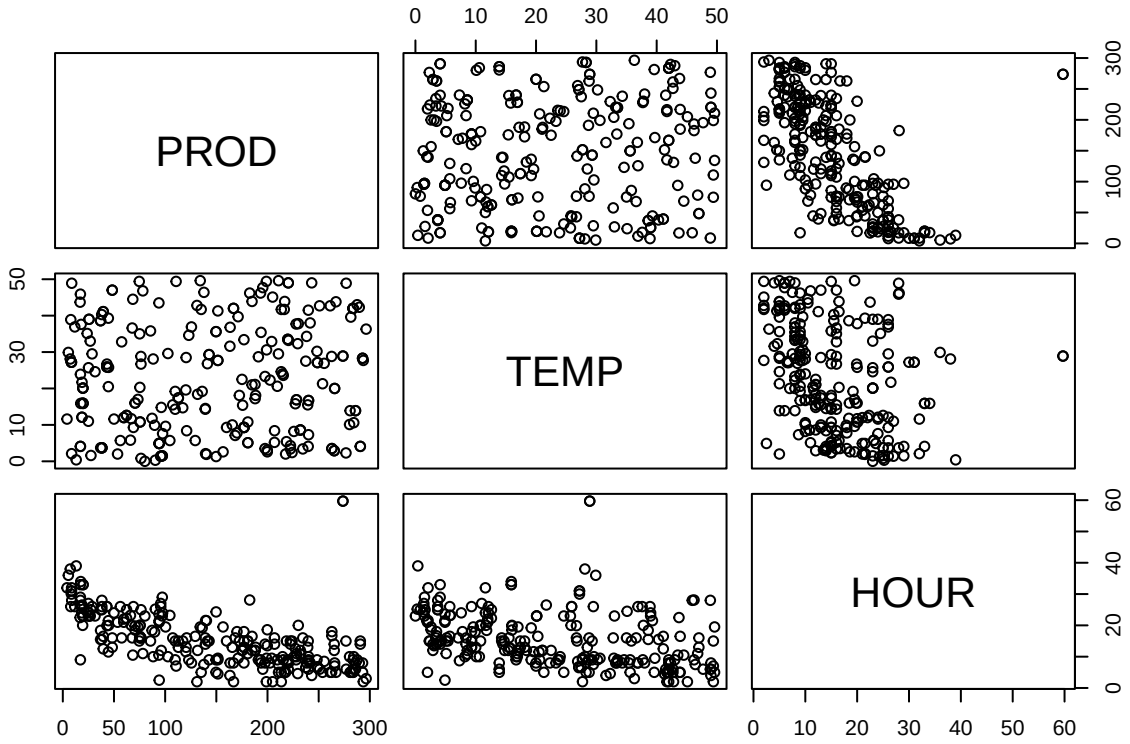
```
# Read in CSV file
water =read.csv("water.csv", header = TRUE)
```

a)

The model given is $\widehat{USAGE} = \hat{\beta}_0 PROD + \hat{\beta}_1 TEMP + \hat{\beta}_2 HOUR + \hat{\beta}_3 (PROD * TEMP) + \hat{\beta}_4 (PROD * HOUR)$.

When testing for multicollinearity, we do not include interaction terms.

```
q1_firstorder <- lm(USAGE~PROD+TEMP+HOUR, data = water)
pairs(~PROD+TEMP+HOUR, data=water)
```




```
vif(q1_firstorder)
```

```
##     PROD     TEMP     HOUR
## 1.645210 1.173827 1.854801
```

From our output, we get VIFs for all variables between 1 and 2. This means there is low multicollinearity between the predictors in this model and there is no issue with the multicollinearity assumption.

b)

We will first create a residual plot to visually see the distribution of residuals vs fitted values of our model.

```
q1_full <-  lm(USAGE~PROD+TEMP+HOUR+ PROD:TEMP+PROD:HOUR, data = water)
ggplot(q1_full, aes(x=.fitted, y=.resid)) +
geom_point() +
geom_hline(yintercept = 0) +
geom_smooth()+
ggtitle("Residual plot: Residual vs Fitted values")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

Testing for heteroscedasticity using the Breusch-Pagan test:

Null hypothesis: heteroscedasticity is not present ($H_0 : \sigma_1^2 = \sigma_2^2 = ... = \sigma_n^2$)

Alternative hypothesis: heteroscedasticity is present (H_a : at least one σ_i^2 is different from the others)

We will set the alpha value to 0.05.

```
bptest(q1_full)
```

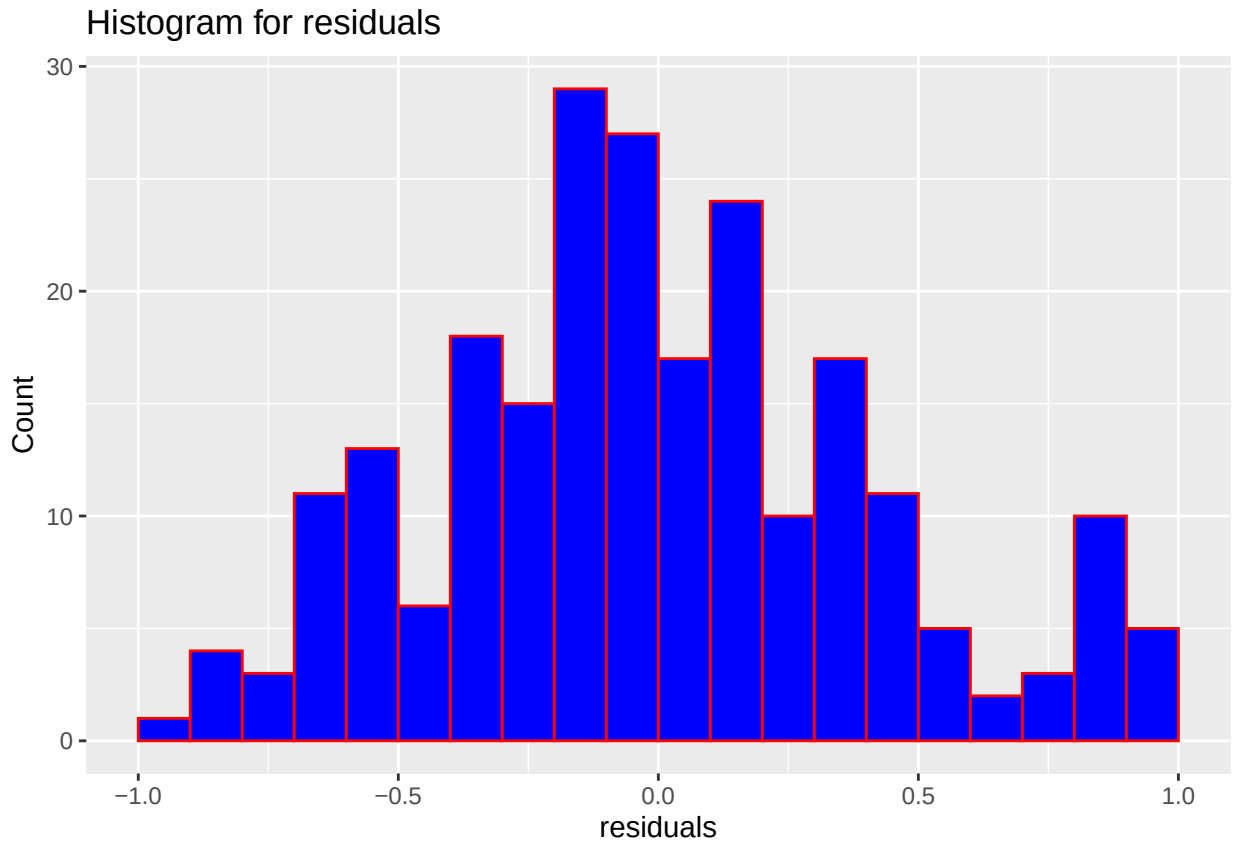
```
##
##  studentized Breusch-Pagan test
##
## data:  q1_full
## BP = 2.0057, df = 5, p-value = 0.8484
```

From the output of our test, we get a p-value of 0.8484 which is greater than 0.05. This means we fail to reject the null hypothesis that there is homoscedasticity and conclude with a significance level of 0.05 that our model is homoscedastic. This means that there does not appear to do a problem with the homoscedasticity assumption.

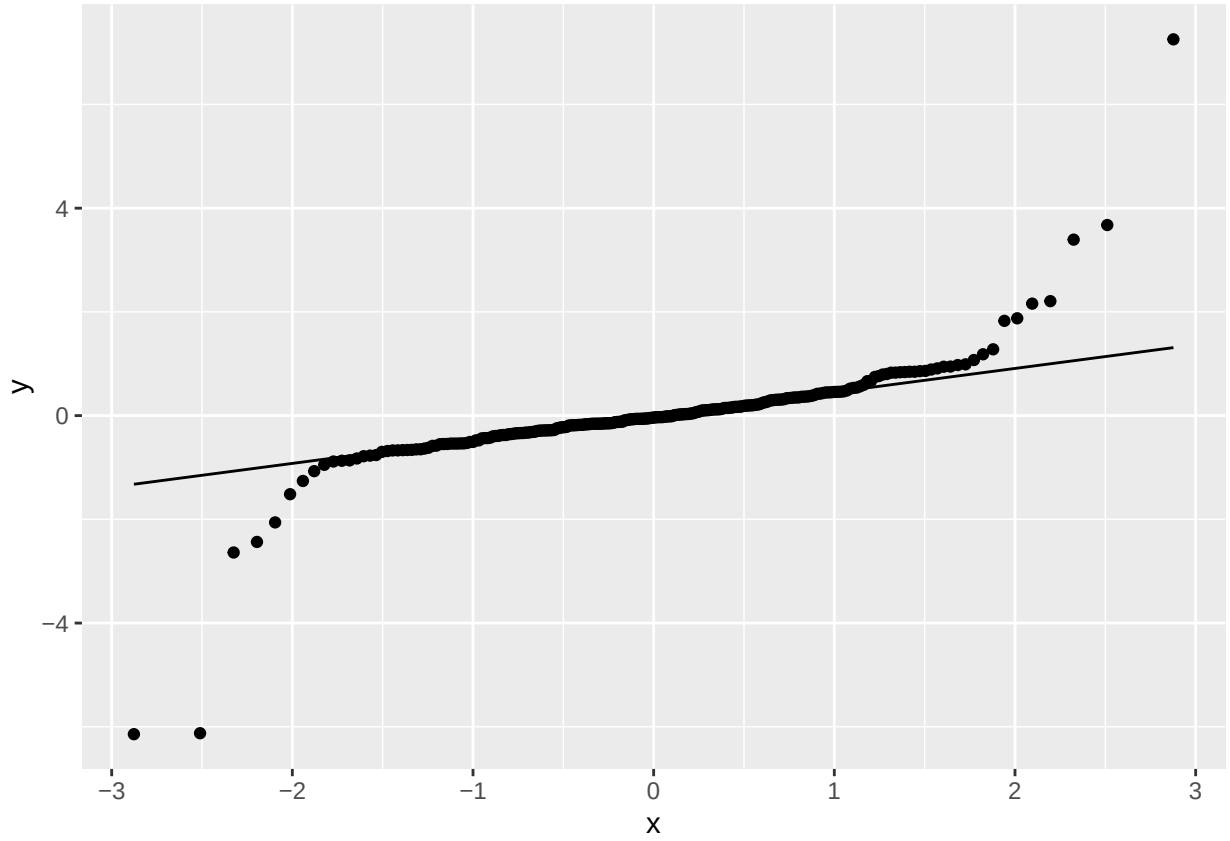
c)

```
# Histogram of residuals
ggplot(data=q1_full, aes(residuals(q1_full))) +
```

```
geom_histogram(breaks = seq(-1,1,by=0.1), col="red", fill="blue") +
labs(title="Histogram for residuals") +
labs(x="residuals", y="Count")
```




```
# Q-Q Plot
ggplot(q1_full, aes(sample=q1_full$residuals)) +
stat_qq() +
stat_qq_line()
```



Testing for normality using the Shapiro-Wilk test:

Null hypothesis: the sample data are significantly normally distributed

Alternative hypothesis: the sample data are not significantly normally distributed

We will set the alpha value to 0.05.

```
shapiro.test(residuals(q1_full))
```

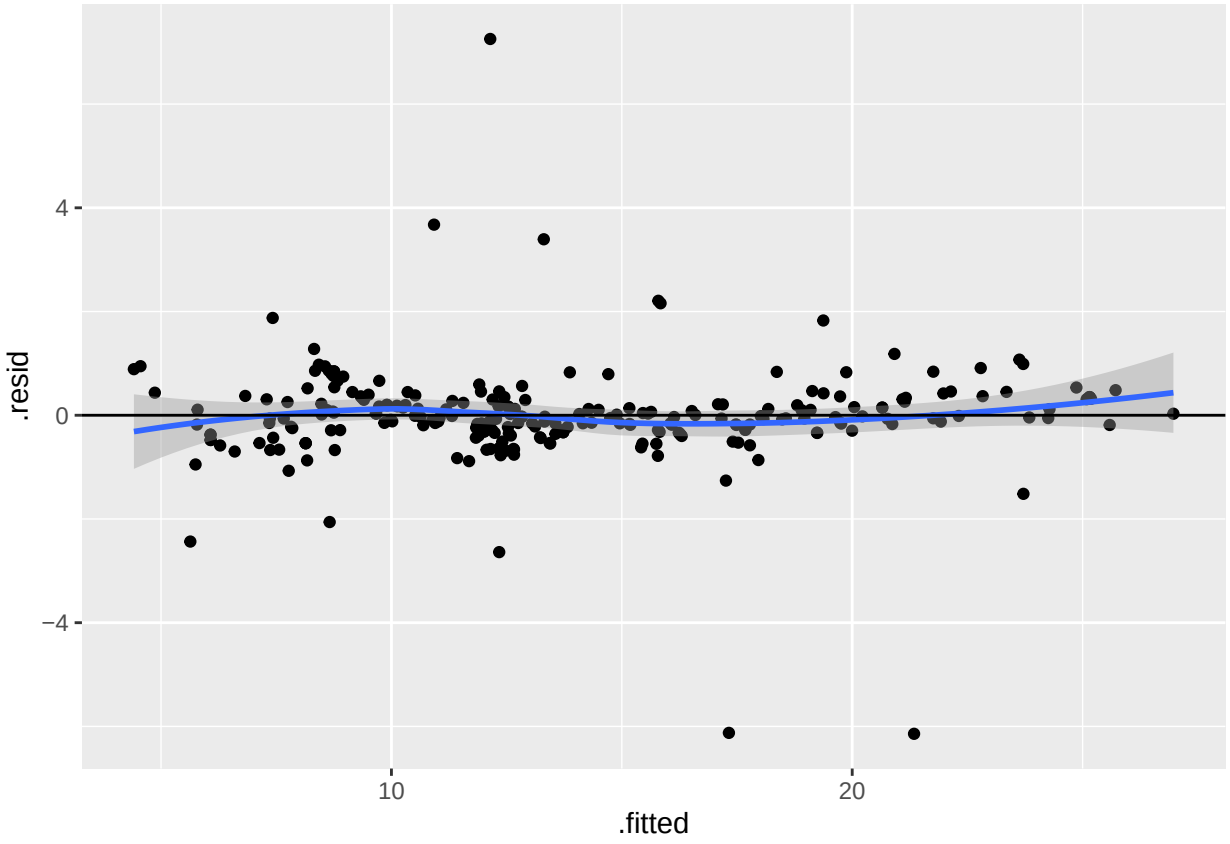
```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(q1_full)
## W = 0.67655, p-value < 0.00000000000000022
```

From the output of our test, we get a p-value of 0.00000000000000022 which is less than 0.05. This means we can reject our null hypothesis that our sample data is significantly normally distributed and conclude with a significance level of 0.05 that our sample data is not normally distributed. This means that there is a problem with the normality assumption.

d)

```
ggplot(q1_full, aes(x=.fitted, y=.resid)) +
geom_point() + geom_smooth()+
geom_hline(yintercept = 0)
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Looking at the plot of residuals vs. predicted $\hat{Y}$, the there is a linear relationship between the predictors in our model and the response variable as the data in the plot do not seem to diverge from the line very much. Therefore, it does not seem like there is a problem with the linearity assumption.

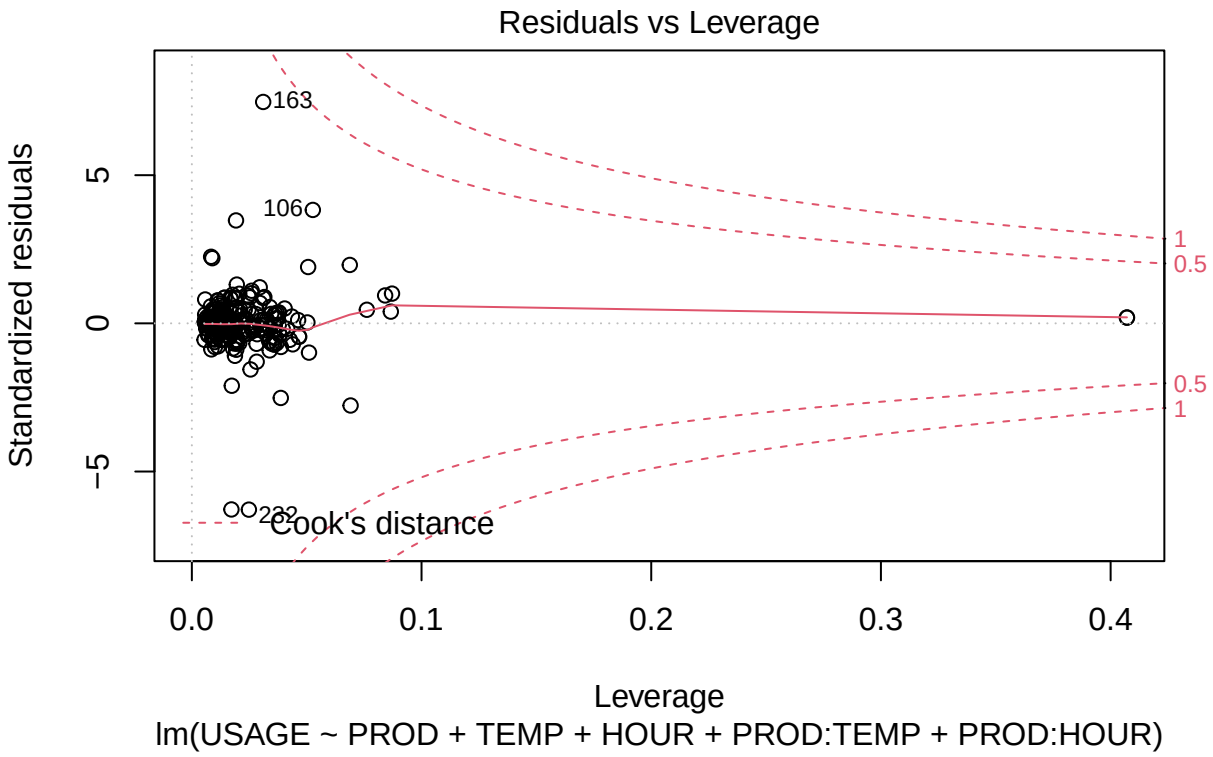
e)

```
# Get points where cooks distance is greater than 1
water[cooks.distance(q1_full)>1,]
```

```
## [1] PROD  TEMP  HOUR  USAGE DAYS
## <0 rows> (or 0-length row.names)
```

From our output, there does not seem to be any data points with a cook's distance greater than 1 which means there are no influential outliers in our data. We will check this conclusion with a residual vs. leverage plot:

```
plot(q1_full,which=5)
```



From our residual vs. leverage plot, we can see that there is indeed no data points with a cook's distance greater than 1. In fact, all points have a cook's distance less than 0.5. So therefore we can conclude that we have no problems with influential outliers in our data.

f)

Based on our conclusions from parts (a)-(e), our model meets all assumptions except the normality assumption since the model failed the Shapiro-Wilk test. To fix the normality issue, we will likely need to add more variables into our model. This could mean adding already existing variables, creating new variables through transformations, or collecting more data.

Problem 2

```
# Read in CSV file
kbi =read.csv("KBI.csv", header = TRUE)
```

a)

Before we begin our tests, we will fit the model given to us:

```
q2_fit <- lm(BURDEN~MEM+SOCIALSU+CGDUR, data=kbi)
```

Testing for normality using the Shapiro-Wilk test:

Null hypothesis: the sample data are significantly normally distributed

Alternative hypothesis: the sample data are not significantly normally distributed

We will set the alpha value to 0.05.

```
shapiro.test(residuals(q2_fit))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(q2_fit)
## W = 0.98407, p-value = 0.2716
```

From the output of our test, we get a p-value of 0.2716 which is greater than 0.05. This means we fail to reject our null hypothesis that our sample data is significantly normally distributed and conclude with a significance level of 0.05 that our sample data is normally distributed. This means that there is no problem with the normality assumption.

Testing for heteroscedasticity using the Breusch-Pagan test:

Null hypothesis: heteroscedasticity is not present ($H_0 : \sigma_1^2 = \sigma_2^2 = ... = \sigma_n^2$)

Alternative hypothesis: heteroscedasticity is present (H_a : at least one σ_i^2 is different from the others)

We will set the alpha value to 0.05.

```
bptest(q2_fit)
```

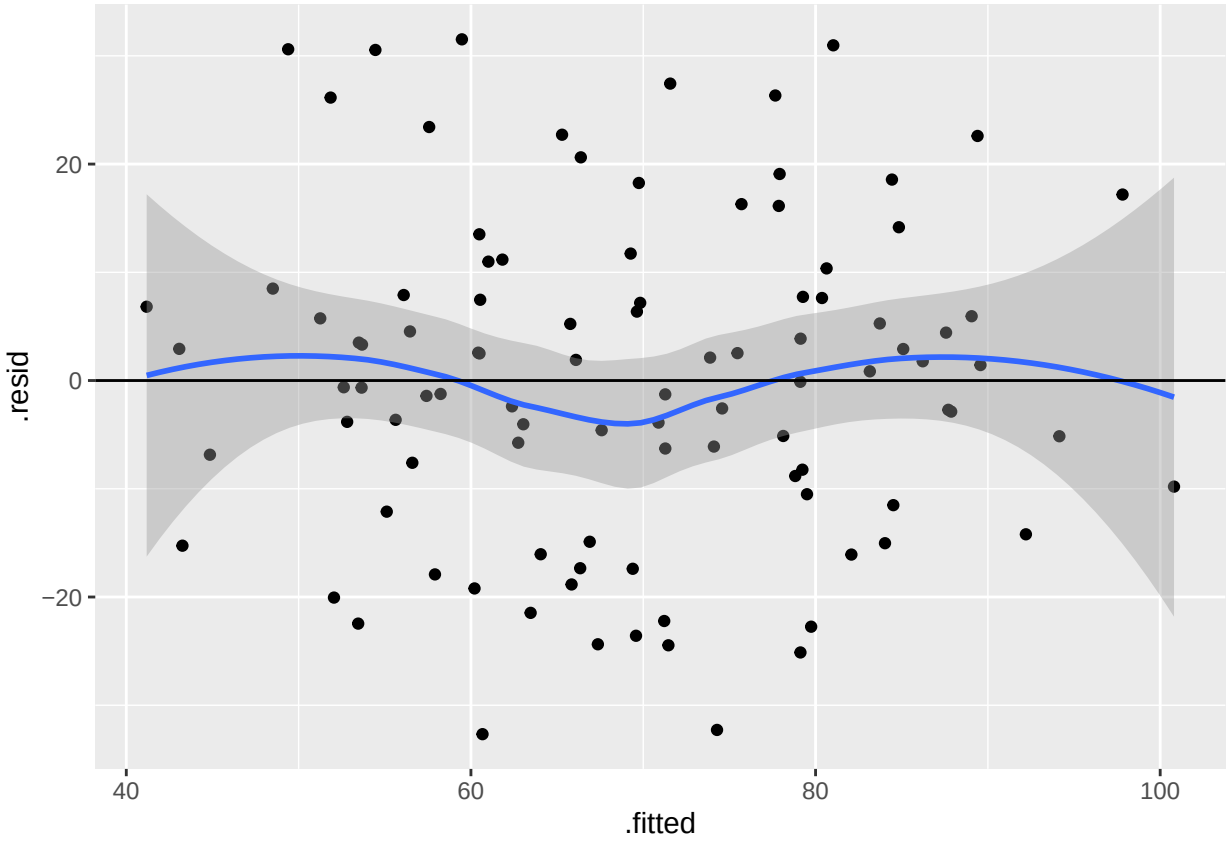
```
##
##  studentized Breusch-Pagan test
##
## data:  q2_fit
## BP = 2.0208, df = 3, p-value = 0.5681
```

From the output of our test, we get a p-value of 0.5681 which is greater than 0.05. This means we fail to reject the null hypothesis that there is homoscedasticity and conclude with a significance level of 0.05 that our model is homoscedastic. This means that there does not appear to do a problem with the homoscedasticity assumption.

Testing for linearity using a residuals vs predicted $\hat{Y}$ plot:

```
ggplot(q2_fit, aes(x=.fitted, y=.resid)) +
geom_point() + geom_smooth()+
geom_hline(yintercept = 0)
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Looking at this plot, our model does seem to be linear.

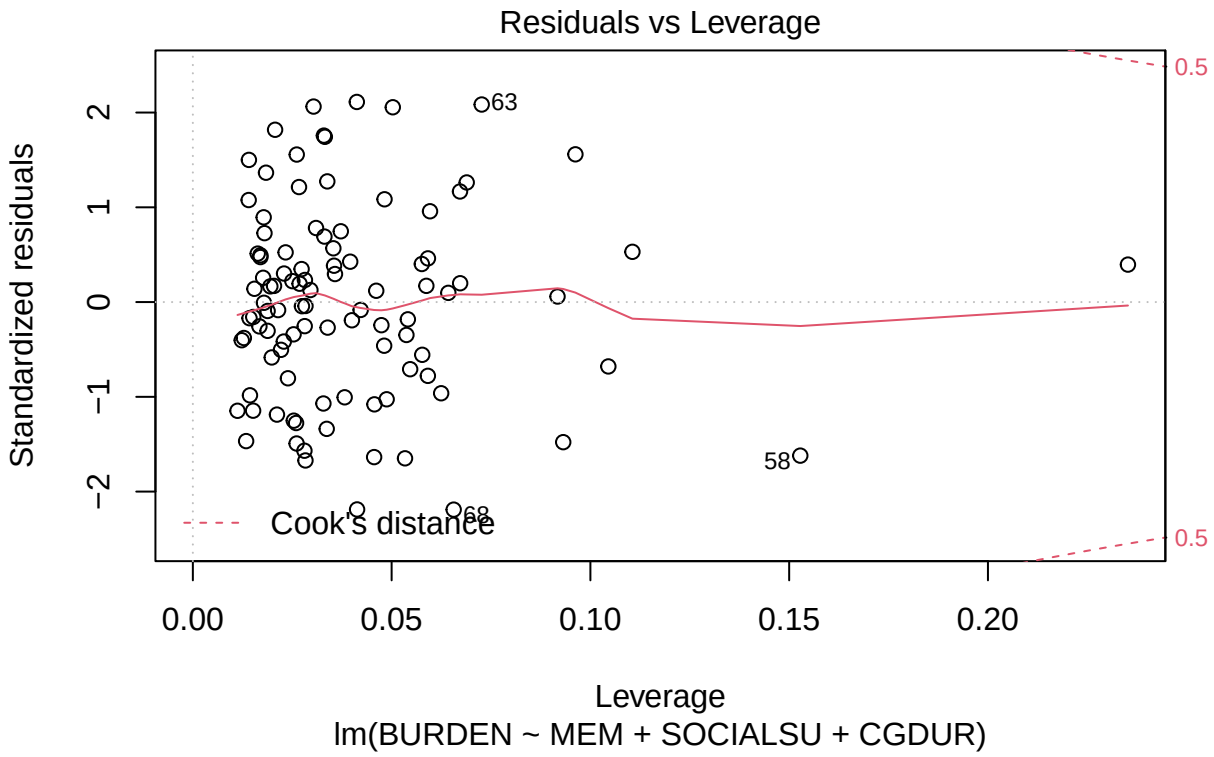
b)

```
lev=hatvalues(q2_fit)
p = length(coef(q2_fit))
n = nrow(kbi)
outlier3p = lev[lev>(3*p/n)]
outlier3p
```

```
##        58        71
## 0.1527990 0.2352185
```

From our output, there seems to be two data points, row 58 and row 71, with a leverage greater than $3p/n$. We will check this with a residual vs. leverage plot:

```
plot(q2_fit,which=5)
```



We will remove row 58 and row 71, as it is considered an influential outlier.

```
kbi_2 <- kbi[-c(58, 71), ]
```

c)

```
q2_fit2 <- lm(BURDEN~MEM+SOCIALSU+CGDUR, data=kbi_2)
summary(q2_fit)
```

```
##
## Call:
## lm(formula = BURDEN ~ MEM + SOCIALSU + CGDUR, data = kbi)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -32.672  -9.977   0.367   7.774  31.523
##
## Coefficients:
##             Estimate Std. Error t value            Pr(>|t|)
## (Intercept) 115.53922   12.36816   9.342 0.00000000000000386 ***
## MEM           0.56612    0.10232   5.533 0.00000027252958394 ***
## SOCIALSU     -0.49237    0.08930  -5.514 0.00000029562624399 ***
## CGDUR         0.12168    0.06486   1.876              0.0637 .
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.25 on 96 degrees of freedom
## Multiple R-squared:  0.4397, Adjusted R-squared:  0.4222
## F-statistic: 25.12 on 3 and 96 DF,  p-value: 0.000000000004433
```

```
summary(q2_fit2)
```

```
##
## Call:
## lm(formula = BURDEN ~ MEM + SOCIALSU + CGDUR, data = kbi_2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -34.714  -8.844  -0.156   8.064  32.455
##
## Coefficients:
##              Estimate Std. Error t value           Pr(>|t|)
## (Intercept) 115.05690   13.04046   8.823 0.0000000000000584 ***
## MEM           0.55803    0.10517   5.306 0.0000007439454900 ***
## SOCIALSU     -0.49423    0.09168  -5.391 0.0000005195507867 ***
## CGDUR         0.16150    0.07544   2.141             0.0349 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.19 on 94 degrees of freedom
## Multiple R-squared:  0.4476, Adjusted R-squared:   0.43
## F-statistic: 25.39 on 3 and 94 DF,  p-value: 0.000000000004056
```

Original data model: $115.539 + 0.566MEM - 0.49237SOCIALSU + 0.121CDUR$

We get an adjusted R-squared value of 0.4222 and a RMSE of 15.25.

New data model: $115.05690 + 0.55803MEM - 0.49423SOCIALSU + 0.16150CDUR$

We get an adjusted R-squared value of 0.43 and a RMSE of 15.19.

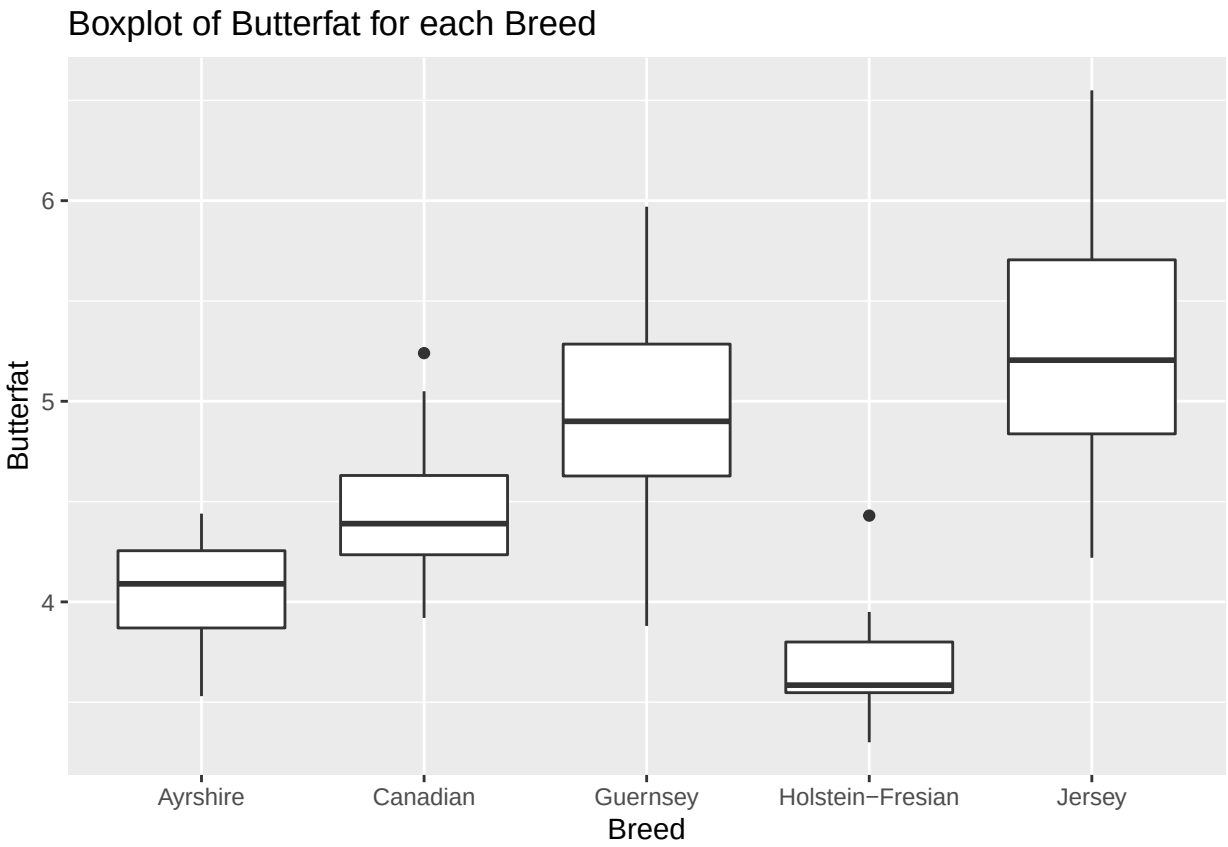
By removed row 58, the intercept and coefficients in our model changed slightly. The adjusted R-squared increased by (0.43 - 0.4222) = 0.0078 and our RMSE decreased by (15.25 - 15.19) = 0.06. So by removing row 58 from the data, our model improved overall.

Problem 3

```
# Read in CSV file
butterfat =read.csv("butterfat.csv", header = TRUE)
```

a)

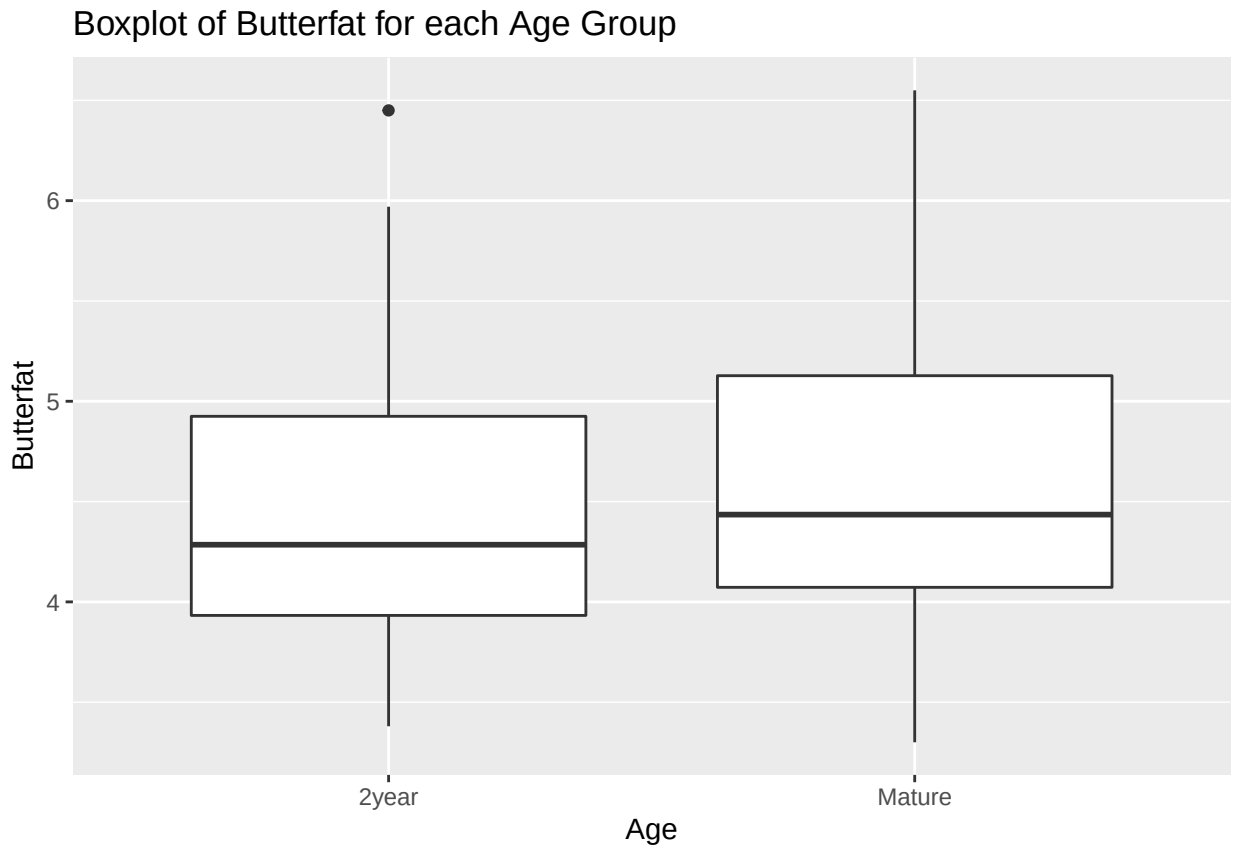
```
ggplot(data = butterfat) + geom_boxplot(aes(x=Breed, y=Butterfat)) + ggtitle("Boxplot of Butterfat for
```



Holstein-Fresian has the lowest mean butterfat content and also the tightest interqartile range. Jersey cows have the highest mean butterfat content along with the largest interquartile range.

Ranking the breeds in terms of mean butter fat content from highest to lowest: Jersey, Guernsey, Canadian, Ayrshire and Holstein-Fresian. This ranking is pretty much the same for the tightness of interquartile range from leats tight to most tight.

```
ggplot(data = butterfat)+ geom_boxplot(aes(x=Age, y=Butterfat))+ggtitle("Boxplot of Butterfat for each
```



The mean butterfat content for mature cows seems to be slightly higher compared to 2year cows. The interquartile range is approximately the same size when comparing the two age groups.

b)

```
q3_fit <- lm(Butterfat~Age+Breed, data=butterfat)
q3_interaction <-  lm(Butterfat~(Age+Breed)^2, data=butterfat)
summary(q3_fit)
```

```
##
## Call:
## lm(formula = Butterfat ~ Age + Breed, data = butterfat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.0202 -0.2373 -0.0640  0.2617  1.2098
##
## Coefficients:
##                         Estimate Std. Error t value             Pr(>|t|)
## (Intercept)              4.00770    0.10135  39.541 < 0.0000000000000002 ***
## AgeMature                0.10460    0.08276   1.264              0.20937
## BreedCanadian            0.37850    0.13085   2.893              0.00475 **
## BreedGuernsey            0.89000    0.13085   6.802  0.00000000094806446 ***
## BreedHolstein-Fresian -0.39050    0.13085  -2.984              0.00362 **
```

```
## BreedJersey            1.23250    0.13085   9.419  0.00000000000000316 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4138 on 94 degrees of freedom
## Multiple R-squared:  0.6825, Adjusted R-squared:  0.6656
## F-statistic: 40.41 on 5 and 94 DF,  p-value: < 0.00000000000000022
```

```
summary(q3_interaction)
```

```
##
## Call:
## lm(formula = Butterfat ~ (Age + Breed)^2, data = butterfat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.0190 -0.2720 -0.0430  0.2372  1.3170
##
## Coefficients:
##                                 Estimate Std. Error t value
## (Intercept)                       3.9660     0.1316  30.143
## AgeMature                         0.1880     0.1861   1.010
## BreedCanadian                     0.5220     0.1861   2.805
## BreedGuernsey                     0.9330     0.1861   5.014
## BreedHolstein-Fresian            -0.3030     0.1861  -1.628
## BreedJersey                       1.1670     0.1861   6.272
## AgeMature:BreedCanadian          -0.2870     0.2631  -1.091
## AgeMature:BreedGuernsey          -0.0860     0.2631  -0.327
## AgeMature:BreedHolstein-Fresian  -0.1750     0.2631  -0.665
## AgeMature:BreedJersey             0.1310     0.2631   0.498
##                                           Pr(>|t|)
## (Intercept)                     < 0.0000000000000002 ***
## AgeMature                                 0.31503
## BreedCanadian                             0.00616 **
## BreedGuernsey                       0.0000026536 ***
## BreedHolstein-Fresian                     0.10693
## BreedJersey                         0.0000000122 ***
## AgeMature:BreedCanadian                   0.27834
## AgeMature:BreedGuernsey                   0.74457
## AgeMature:BreedHolstein-Fresian           0.50773
## AgeMature:BreedJersey                     0.61982
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4161 on 90 degrees of freedom
## Multiple R-squared:  0.6926, Adjusted R-squared:  0.6619
## F-statistic: 22.53 on 9 and 90 DF,  p-value: < 0.00000000000000022
```

I would not keep age in my model for predicting butterfat content. This is because age is not statistically significant per individual t-test, and none of it's interactions are statistically significant either.

```
q3_final <- lm(Butterfat~Breed, data=butterfat)
summary(q3_final)
```

```
##
## Call:
## lm(formula = Butterfat ~ Breed, data = butterfat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.07250 -0.27213 -0.05125  0.22363  1.25750
##
## Coefficients:
##                       Estimate Std. Error t value             Pr(>|t|)
## (Intercept)            4.06000    0.09281  43.743 < 0.0000000000000002 ***
## BreedCanadian          0.37850    0.13126   2.884              0.00486 **
## BreedGuernsey          0.89000    0.13126   6.780  0.00000000100941928 ***
## BreedHolstein-Fresian -0.39050    0.13126  -2.975              0.00371 **
## BreedJersey            1.23250    0.13126   9.390  0.00000000000000333 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4151 on 95 degrees of freedom
## Multiple R-squared:  0.6771, Adjusted R-squared:  0.6635
## F-statistic:  49.8 on 4 and 95 DF,  p-value: < 0.00000000000000022
```

Therefore, our predictive model will be: $\widehat{Butterfat} = 4.06000 + 0.37850(Breed_{Canadian}) + 0.89000(Breed_{Guernsey}) - 0.39050(Breed_{Holstein-Fresian}) + 1.23250(Breed_{Jersey})$

c)

Testing for normality using the Shapiro-Wilk test:

Null hypothesis: the sample data are significantly normally distributed

Alternative hypothesis: the sample data are not significantly normally distributed

We will set the alpha value to 0.05.

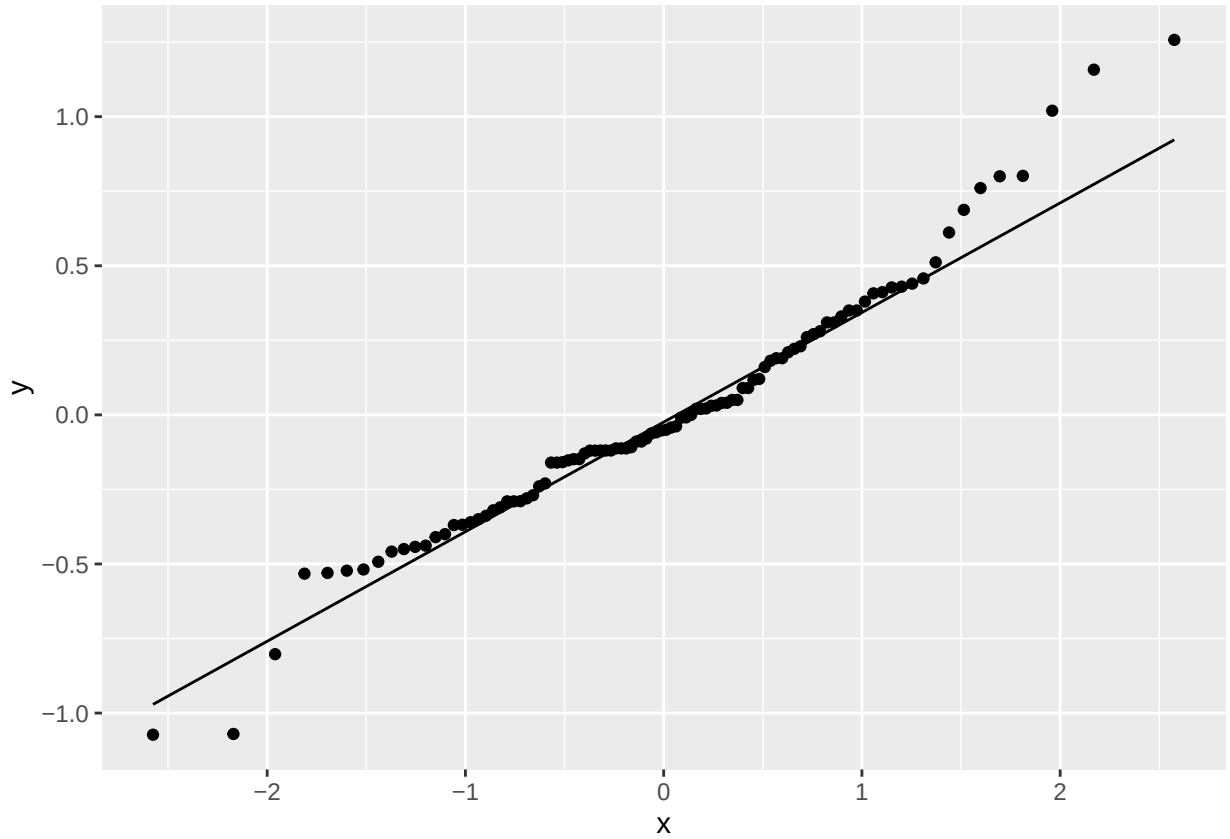
```
shapiro.test(residuals(q3_final))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(q3_final)
## W = 0.96805, p-value = 0.01571
```

From the output of our test, we get a p-value of 0.01571 which is less than 0.05. This means we can reject our null hypothesis that our sample data is significantly normally distributed and conclude with a significance level of 0.05 that our sample data is not normally distributed. This means that there is a problem with the normality assumption.

Plotting a Q-Q plot:

```
ggplot(q3_final, aes(sample=q3_final$residuals)) +
stat_qq() +
stat_qq_line()
```



Looking at our Q-Q plot, it does make sense that we would reject the null hypothesis of our sample data is significantly normally distributed in our Shapiro-Wilk test as it does look like a good amount of the points are diverging from the normaility line.

Testing for heteroscedasticity (non-constant variance) using the Breusch-Pagan test:

Null hypothesis: heteroscedasticity is not present ($H_0 : \sigma_1^2 = \sigma_2^2 = ... = \sigma_n^2$)

Alternative hypothesis: heteroscedasticity is present (H_a : at least one σ_i^2 is different from the others)

We will set the alpha value to 0.05.

```
bptest(q3_final)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  q3_final
## BP = 13.389, df = 4, p-value = 0.009525
```

From the output of our test, we get a p-value of 0.009525 which is less than 0.05. This means we can reject the null hypothesis that there is homoscedasticity and conclude with a significance level of 0.05 that our

model is not homoscedastic. This means that there does appear to do a problem with the homoscedasticity assumption.

Conclusion: Based on the tests conducted above, the data does not seem to be normally distributed since we failed the Shapiro-Wilk test and the model seems to be heteroscedastic since we failed the Breusch-Pagan test.

d)

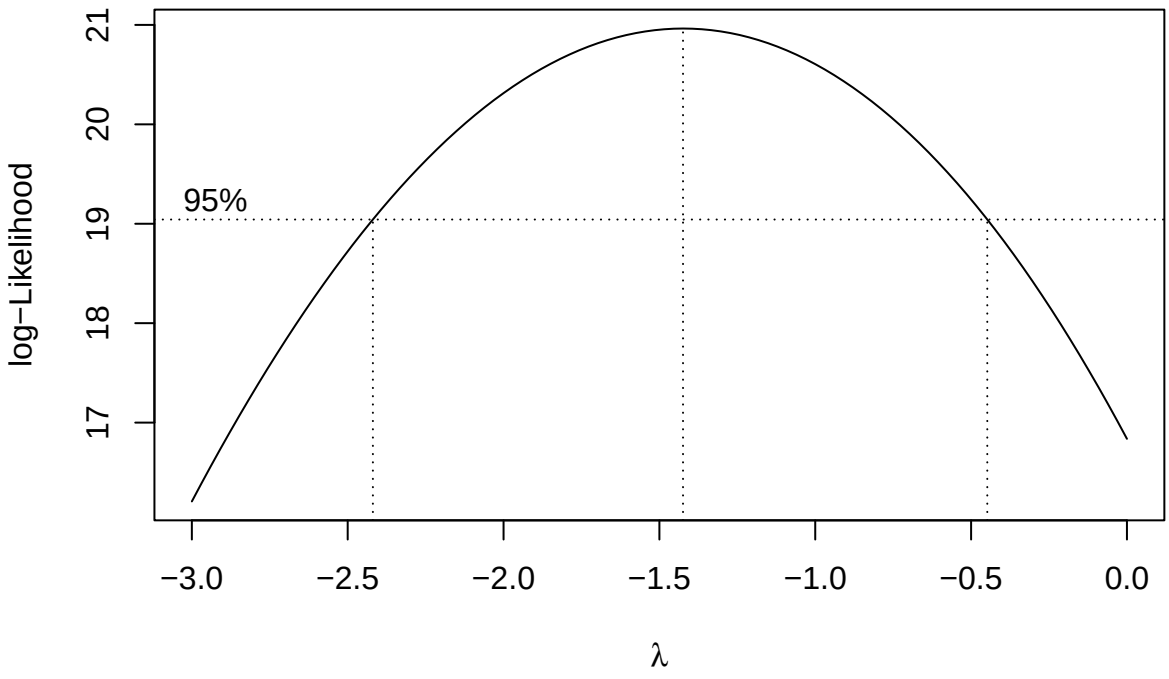
To fix the issues with our model identified in part (c), we will do a Box-Cox transformation. First, we should confirm that our responce variable is always positive.

```
butterfat[butterfat["Butterfat"]<0,]
```

```
## [1] Butterfat Breed     Age
## <0 rows> (or 0-length row.names)
```

We get no rows where butterfat is less than 0, so we can continue with the Box-Cox transformation.

```
bc=boxcox(q3_final,lambda=seq(-3,0))
```




```
bestlambda=bc$x[which(bc$y==max(bc$y))]
bestlambda
```

```
## [1] -1.424242
```

From above, the best lambda is approximately -1.424242.

```
bcmodel = lm((((Butterfat^(-1.424242))-1)/-1.424242)~Breed,data=butterfat)
summary(q3_final)
```

```
##
## Call:
## lm(formula = Butterfat ~ Breed, data = butterfat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.07250 -0.27213 -0.05125  0.22363  1.25750
##
## Coefficients:
##                       Estimate Std. Error t value             Pr(>|t|)
## (Intercept)            4.06000    0.09281  43.743 < 0.0000000000000002 ***
## BreedCanadian          0.37850    0.13126   2.884              0.00486 **
## BreedGuernsey          0.89000    0.13126   6.780  0.00000000100941928 ***
## BreedHolstein-Fresian -0.39050    0.13126  -2.975              0.00371 **
## BreedJersey            1.23250    0.13126   9.390  0.00000000000000333 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4151 on 95 degrees of freedom
## Multiple R-squared:  0.6771, Adjusted R-squared:  0.6635
## F-statistic:  49.8 on 4 and 95 DF,  p-value: < 0.00000000000000022
```

```
summary(bcmodel)
```

```
##
## Call:
## lm(formula = (((Butterfat^(-1.424242)) - 1)/-1.424242) ~ Breed,
##     data = butterfat)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0286891 -0.0060992  0.0000438  0.0073002  0.0267700
##
## Coefficients:
##                       Estimate Std. Error t value             Pr(>|t|)
## (Intercept)           0.606021   0.002235 271.165 < 0.0000000000000002 ***
## BreedCanadian         0.011144   0.003161   3.526             0.000652 ***
## BreedGuernsey         0.022989   0.003161   7.274  0.00000000009892312 ***
## BreedHolstein-Fresian -0.014954   0.003161  -4.732  0.00000775058571188 ***
## BreedJersey           0.029329   0.003161   9.280  0.00000000000000573 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.009995 on 95 degrees of freedom
## Multiple R-squared:  0.7274, Adjusted R-squared:  0.7159
## F-statistic: 63.36 on 4 and 95 DF,  p-value: < 0.00000000000000022
```

```
(0.4151 - 0.009995)/0.4151
```

```
## [1] 0.9759215
```

Comparing our Box-Cox transformed model with our original model fitted in part (b):

Original model: $\widehat{Butterfat} = 4.06000 + 0.37850(Breed_{Canadian}) + 0.89000(Breed_{Guernsey}) - 0.39050(Breed_{Holstein-Fresian}) + 1.23250(Breed_{Jersey})$

Box-Cox model: $\widehat{Butterfat} = 0.606021 + 0.011144(Breed_{Canadian}) + 0.022989(Breed_{Guernsey}) - 0.014954(Breed_{Holstein-Fresian}) + 0.029329(Breed_{Jersey})$

Our Box-Cox model has an adjusted R-squared of 0.6988 which is (0.7159 - 0.6635) = 0.0524 or 7.89% larger than our original fitted model. We also get an RMSE of 0.05462 which is (0.4151 - 0.009995) = 0.36048, or 97.59% smaller than our original model. Therefore, our Box-Cox transformed model is superior in both regards.

e)

We will now do a diagnostics analysis like in part (c) on our Box-Cox model.

Testing for normality using the Shapiro-Wilk test:

Null hypothesis: the sample data are significantly normally distributed

Alternative hypothesis: the sample data are not significantly normally distributed

We will set the alpha value to 0.05.

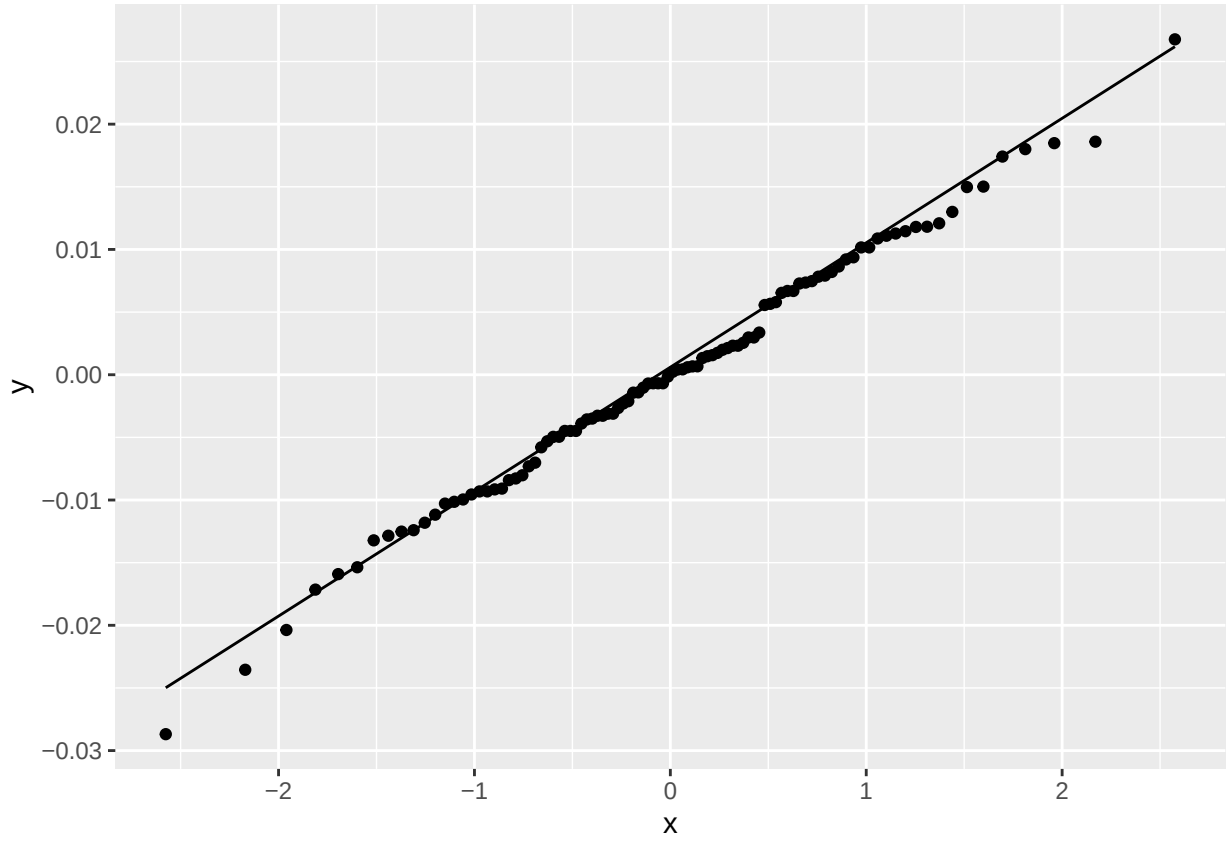
```
shapiro.test(residuals(bcmodel))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(bcmodel)
## W = 0.99449, p-value = 0.9596
```

From the output of our test, we get a p-value of 0.2578 which is greater than 0.05. This means we fail to reject our null hypothesis that our sample data is significantly normally distributed and conclude with a significance level of 0.05 that our sample data is not normally distributed. This means that there is no longer a problem with the normality assumption.

Plotting a Q-Q Plot:

```
ggplot(bcmodel, aes(sample=bcmodel$residuals)) +
stat_qq() +
stat_qq_line()
```



Testing for heteroscedasticity (non-constant variance) using the Breusch-Pagan test:

Null hypothesis: heteroscedasticity is not present ($H_0 : \sigma_1^2 = \sigma_2^2 = ... = \sigma_n^2$)

Alternative hypothesis: heteroscedasticity is present (H_a : at least one σ_i^2 is different from the others)

We will set the alpha value to 0.05.

```
bptest(bcmodel)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  bcmodel
## BP = 0.58064, df = 4, p-value = 0.9652
```

From the output of our test, we get a p-value of 0.2689 which is greater than 0.05. This means we fail to reject the null hypothesis that there is homoscedasticity and conclude with a significance level of 0.05 that our model is in fact homoscedastic. This means that there no longer appears to do a problem with the homoscedasticity assumption.

From the diagnostics analysis conducted above, it seems like the Box-Cox transformation fixed the issues with our model assumptions since our model now passes the Breusch-Pagan and Shapiro-Wilk test.

Problem 4

```r
# Read in CSV file
vibration =read.csv("vibration.csv", header = TRUE)
```

a)

The response variable is the amount of motor vibration (measured in microns).

The experimental unit is the motors.

b)

The treatment is the bearing used in the motor.

There are 5 treatment levels because there are 5 different brands of bearings.

c)

Null hypothesis: $H_0 : \mu_1 = \mu_2 = \mu_3 = \mu_4 = \mu_5$

Null hypothesis: H_a : At least one μ_i is different for $i = 1, 2, 3, 4, 5$

We will set the alpha value to 0.05.

```r
CRD <- aov(vibration~brand, data=vibration)
summary(CRD)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## brand        4  30.86   7.714   8.444 0.000187 ***
## Residuals   25  22.84   0.914
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

From our ANOVA output, we get a F-calc of 8.444 and a p-value of 0.000187. Since the p-value is less than our set alpha value of 0.05, we can reject the null hypothesis that the mean amount of motor vibration is the same for all brands of bearings. Therefore, we can conclude with a significance level of 0.05 that at least one of the mean motor vibrations is different for the five brands of bearings.

d)

```r
vib_fit <- lm(vibration~brand, data=vibration)
reg_df = 4
res_df = nrow(vibration) - reg_df - 1
total_df = reg_df + res_df
ssr = sum((vib_fit$fitted.values - mean(vibration$vibration))^2)
sse = sum((vib_fit$fitted.values-vibration$vibration)^2)
sst = sse + ssr
msr = ssr/reg_df
mse = sse/res_df
f_calc = msr/mse
col_1 <- c("Regression", "Residual", "Total")
```

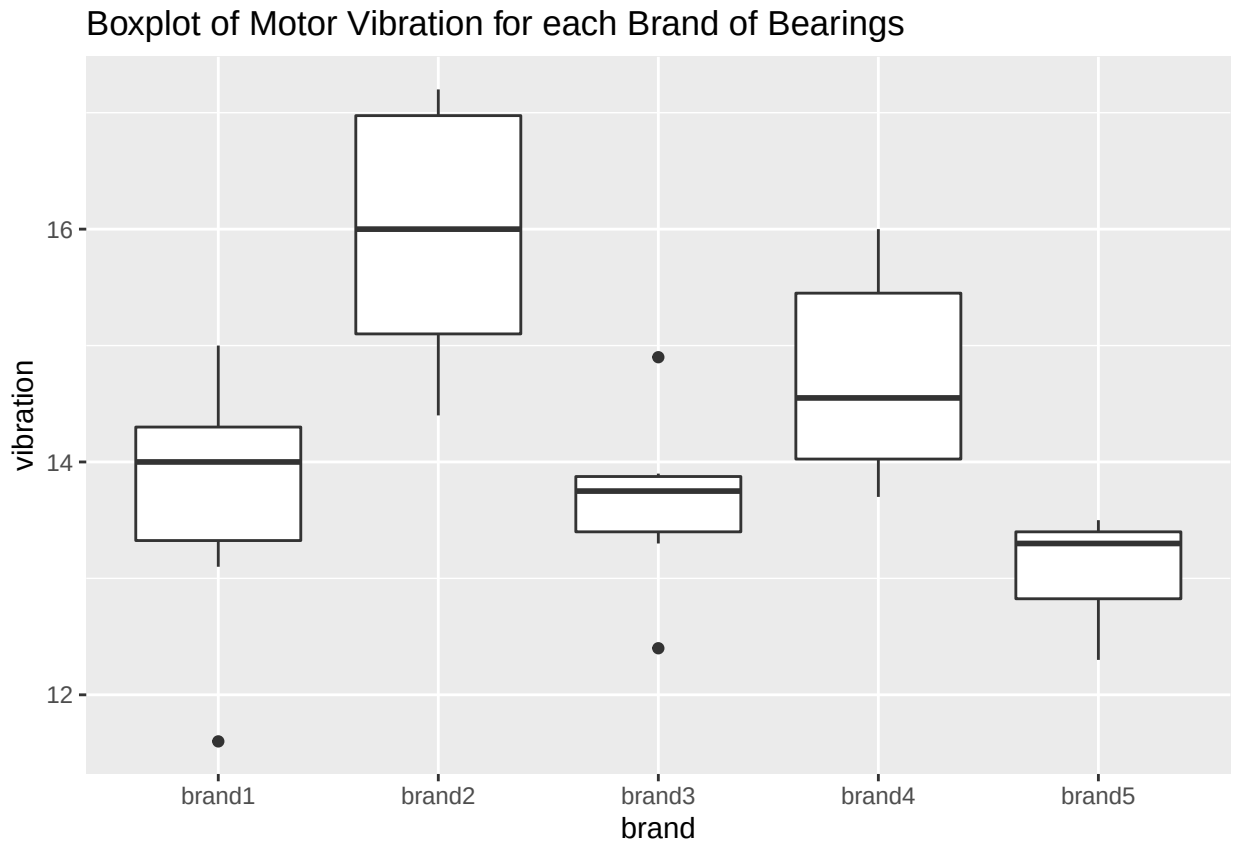
```r
col_2 <- c(reg_df, res_df, total_df)
col_3 <- c(ssr, sse, sst)
col_4 <- c(msr, mse, "")
col_5 <- c(f_calc, "", "")
anova_df <- data.frame(col_1, col_2,col_3,col_4, col_5)
colnames(anova_df) <- c("Source of Variation", "Df", "Sum of Squares", "Mean Square", "F-Statistic")
knitr::kable(anova_df, caption = "ANOVA Table")
```

Table 1: ANOVA Table

Source of Variation	Df	Sum of Squares	Mean Square	F-Statistic
Regression	4	30.85533	7.71383333333332	8.44395387871268
Residual	25	22.83833	0.913533333333334	
Total	29	53.69367		

e)

```r
ggplot(data = vibration) + geom_boxplot(aes(x=brand, y=vibration)) + ggtitle("Boxplot of Motor Vibratio
```



Based on the boxplots above, it seems like there may be some influential outliers for brand 1 and brand 3. To see for sure, we should calculate their cook's distance.

```r
# Maybe remove this
vibration[cooks.distance(vib_fit)>0.1,]
```

```
##    vibration  brand
## 6       11.6 brand1
## 11      14.4 brand2
```

f)

Before we do any pairwise tests, we should conduct a global F-test to be certain that at least one of the columns is statistically significant:

```r
vibration_null <- lm(vibration~1, data=vibration)
anova(vibration_null, vib_fit)
```

```
## Analysis of Variance Table
##
## Model 1: vibration ~ 1
## Model 2: vibration ~ brand
##   Res.Df    RSS Df Sum of Sq     F    Pr(>F)
## 1     29 53.694
## 2     25 22.838  4    30.855 8.444 0.0001871 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We get an F-stat of 8.444 and a p-value of 0.0001871 meaning that we can say that at least one μ_{brand} is significant. So we can go ahead with the pairwise tests. For all tests we will set our alpha value to 0.05.

Unadjusted Pairwise t-test:

```r
pairwise.t.test(vibration$vibration,vibration$brand, p.adj = "none")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  vibration$vibration and vibration$brand
##
##        brand1  brand2   brand3  brand4
## brand2 0.00038 -        -       -
## brand3 0.97615 0.00035  -       -
## brand4 0.06865 0.03689  0.06464 -
## brand5 0.28728 0.000023 0.30058 0.00618
##
## P value adjustment method: none
```

From the output, we get the following groups:

Brand 1: a Brand 2: b Brand 3: a Brand 4: a Brand 5: a

Adjusted Pairwise t-test:

Bonferroni Adjustment:

```
pairwise.t.test(vibration$vibration,vibration$brand, p.adj = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  vibration$vibration and vibration$brand
##
##        brand1  brand2  brand3  brand4
## brand2 0.00376 -       -       -
## brand3 1.00000 0.00348 -       -
## brand4 0.68648 0.36891 0.64642 -
## brand5 1.00000 0.00023 1.00000 0.06184
##
## P value adjustment method: bonferroni
```

From the output, we get the following groups:

Brand 1: a Brand 2: b Brand 3: a Brand 4: a Brand 5: a

Holm Adjustment:

```
pairwise.t.test(vibration$vibration,vibration$brand, p.adj = "holm")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  vibration$vibration and vibration$brand
##
##        brand1  brand2  brand3  brand4
## brand2 0.00313 -       -       -
## brand3 0.97615 0.00313 -       -
## brand4 0.32321 0.22134 0.32321 -
## brand5 0.86183 0.00023 0.86183 0.04329
##
## P value adjustment method: holm
```

From the output, we get the following groups: Brand 1: a Brand 2: b Brand 3: a Brand 4: ab Brand 5: a

Holm is the superior t-test out of the three we have conducted, so we will defer to it's output.

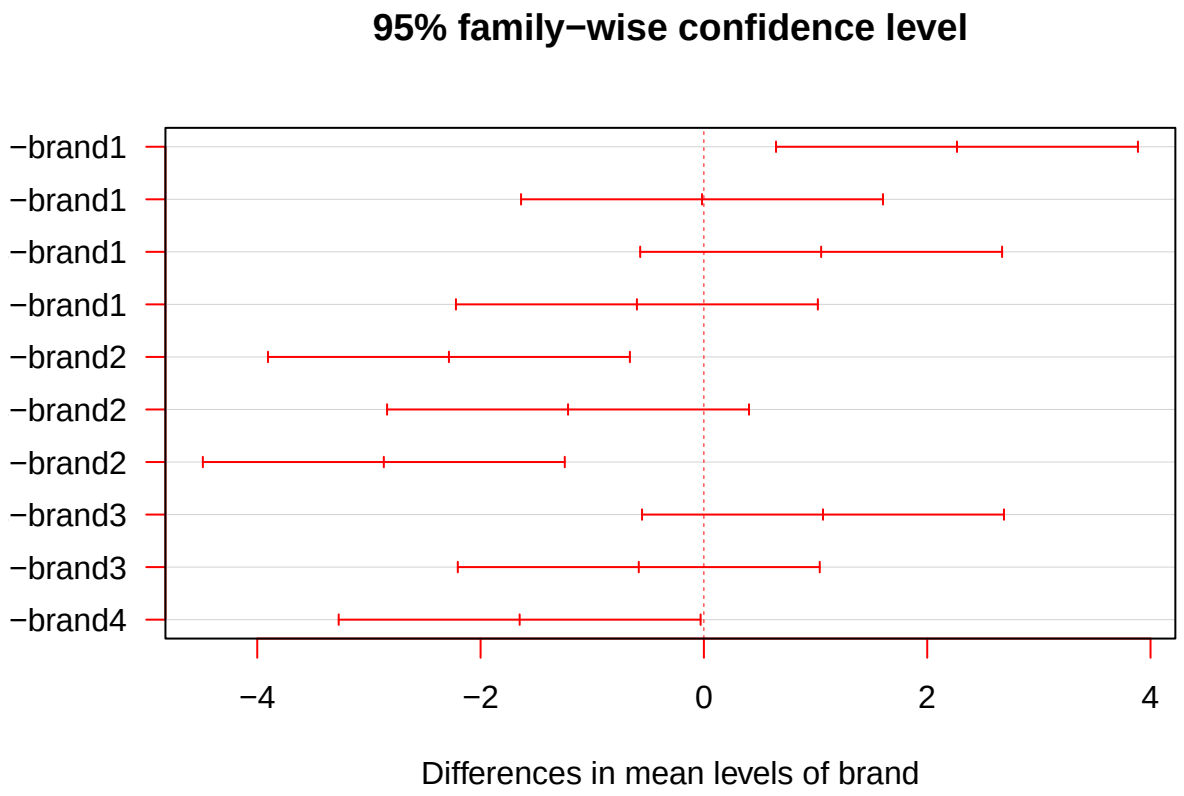
Tukey HSD Test:

```
TukeyHSD(CRD, conf.level = 0.95)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = vibration ~ brand, data = vibration)
##
## $brand
##                       diff        lwr        upr     p adj
## brand2-brand1  2.26666667  0.6460270  3.8873064 0.0031588
## brand3-brand1 -0.01666667 -1.6373064  1.6039730 0.9999998
```

```
## brand4-brand1  1.05000000 -0.5706397  2.6706397 0.3418272
## brand5-brand1 -0.60000000 -2.2206397  1.0206397 0.8112981
## brand3-brand2 -2.28333333 -3.9039730 -0.6626936 0.0029299
## brand4-brand2 -1.21666667 -2.8373064  0.4039730 0.2106883
## brand5-brand2 -2.86666667 -4.4873064 -1.2460270 0.0002024
## brand4-brand3  1.06666667 -0.5539730  2.6873064 0.3268245
## brand5-brand3 -0.58333333 -2.2039730  1.0373064 0.8262091
## brand5-brand4 -1.65000000 -3.2706397 -0.0293603 0.0445279
```

```
plot(TukeyHSD(CRD, conf.level = 0.95),las=1, col = "red")
```



From the output, we get the following groups: Brand 1: a Brand 2: b Brand 3: ab Brand 4: ab Brand 5: a

Newman-Keuls Test:

```
print(SNK.test(CRD,"brand",group=TRUE))
```

```
## $statistics
##     MSerror Df     Mean       CV
##   0.9135333 25 14.22333 6.719869
##
## $parameters
##   test name.t ntr alpha
##    SNK  brand   5  0.05
##
## $snk
```

```
##      Table CriticalRange
## 2 2.912627      1.136505
## 3 3.522566      1.374503
## 4 3.889997      1.517874
## 5 4.153363      1.620640
##
## $means
##        vibration       std r  Min  Max    Q25   Q50    Q75
## brand1  13.68333 1.1940128 6 11.6 15.0 13.325 14.00 14.300
## brand2  15.95000 1.1674759 6 14.4 17.2 15.100 16.00 16.975
## brand3  13.66667 0.8164966 6 12.4 14.9 13.400 13.75 13.875
## brand4  14.73333 0.9395034 6 13.7 16.0 14.025 14.55 15.450
## brand5  13.08333 0.4792355 6 12.3 13.5 12.825 13.30 13.400
##
## $comparison
## NULL
##
## $groups
##        vibration groups
## brand2  15.95000      a
## brand4  14.73333      b
## brand1  13.68333     bc
## brand3  13.66667     bc
## brand5  13.08333      c
##
## attr(,"class")
## [1] "group"
```

Brand 2 & Brand 4 & Brand 5 are different from each other.

Scheffe Test:

```
scheffe.test(CRD,"brand", group=TRUE,console=TRUE)
```

```
##
## Study: CRD ~ "brand"
##
## Scheffe Test for vibration
##
## Mean Square Error  : 0.9135333
##
## brand,  means
##
##        vibration       std r  Min  Max
## brand1  13.68333 1.1940128 6 11.6 15.0
## brand2  15.95000 1.1674759 6 14.4 17.2
## brand3  13.66667 0.8164966 6 12.4 14.9
## brand4  14.73333 0.9395034 6 13.7 16.0
## brand5  13.08333 0.4792355 6 12.3 13.5
##
## Alpha: 0.05 ; DF Error: 25
## Critical Value of F: 2.75871
##
## Minimum Significant Difference: 1.833094
```

```
##
## Means with the same letter are not significantly different.
##
##        vibration groups
## brand2  15.95000      a
## brand4  14.73333     ab
## brand1  13.68333      b
## brand3  13.66667      b
## brand5  13.08333      b
```

Brand 2 is different from Brand 1 & Brand 3 & Brand 5

Final Conclusions:

```
brands <- c("Brand 1","Brand 2","Brand 3","Brand 4","Brand 5")
tests <- c("Unadjusted  t-test", "Bonferroni", "Holm", "Tukey HSD", "Newman-Keuls", "Scheffe")
unadjusted <- c("a", "b", "a", "a", "a")
bonferroni <- c("a", "b", "a", "a", "a")
holm <- c("a", "b", "a", "ab", "a")
tukey <- c("a", "b", "ab", "ab", "a")
newman <- c("a", "b", "bc", "bc", "c")
scheffe <- c("a", "ab", "b", "b", "b")
final_conc <- data.frame(brands,unadjusted,bonferroni,holm,tukey,newman,scheffe )
final_conc
```

```
##    brands unadjusted bonferroni holm tukey newman scheffe
## 1 Brand 1          a          a    a     a      a       a
## 2 Brand 2          b          b    b     b      b      ab
## 3 Brand 3          a          a    a    ab     bc       b
## 4 Brand 4          a          a   ab    ab     bc       b
## 5 Brand 5          a          a    a     a      c       b
```

The above data frame displays all of the outputs for the tests we conducted. Our final conclusion will depend on what test we choose.

g)

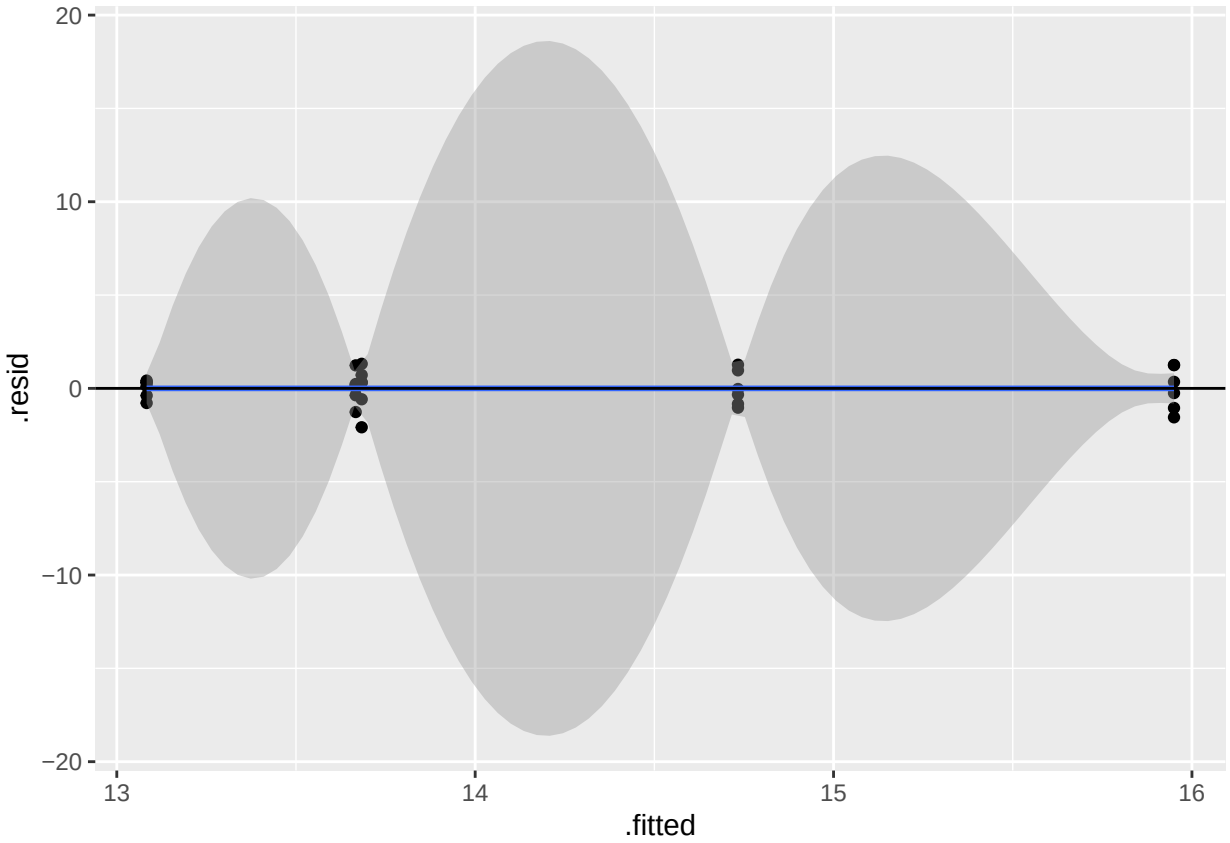
There are three basic assumption of CRD model:

1. The error terms are independent of each other
2. The error terms are normally distributed with a true mean of 0
3. The error terms have constant variance (homoscedastic)
4. Plotting Residuals vs. Fitted values:

If the residuals are independent of each other, there should be no obvious patterns in the plot.

```
ggplot(vib_fit, aes(x=.fitted, y=.resid)) +
geom_point() + geom_smooth()+
geom_hline(yintercept = 0)
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Looking at the plot above, it seems that the residuals are evenly distributed, meaning that the residuals are independant from each other.

2. Testing for normality using the Shapiro-Wilk test:

Null hypothesis: the sample data are significantly normally distributed

Alternative hypothesis: the sample data are not significantly normally distributed

We will set the alpha value to 0.05.

```
shapiro.test(residuals(vib_fit))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(vib_fit)
## W = 0.95996, p-value = 0.3091
```

From the output of our test, we get a p-value of 0.3091 which is greater than 0.05. This means we fail to reject our null hypothesis that our sample data is significantly normally distributed and conclude with a significance level of 0.05 that our sample data is not normally distributed. This means that there is no problem with the normality assumption.

3. Testing for heteroscedasticity (non-constant variance) using the Breusch-Pagan test:

Null hypothesis: heteroscedasticity is not present ($H_0 : \sigma_1^2 = \sigma_2^2 = ... = \sigma_n^2$)

Alternative hypothesis: heteroscedasticity is present (H_a : at least one σ_i^2 is different from the others)

We will set the alpha value to 0.05.

```
bptest(vib_fit)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  vib_fit
## BP = 4.5697, df = 4, p-value = 0.3344
```

From the output of our test, we get a p-value of 0.3344 which is greater than 0.05. This means we fail to reject the null hypothesis that there is homoscedasticity and conclude with a significance level of 0.05 that our model is homoscedastic. This means that there does not appear to do a problem with the homoscedasticity assumption.

Final Conclusion:

Based on the above tests on our model's basic assumptions, we can conclude that our model meets the assumption of independent error terms, normality and homoscedasticity. Therefore there are no issues with our model in regards to the assumptions and we do not need to do anything else.