S510 HW2

Erick Castillo

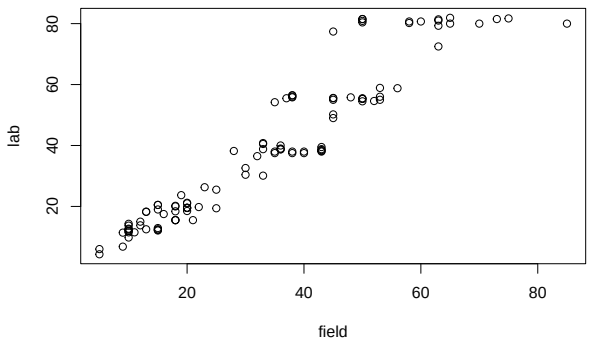
9/28/2021

Problem 1

This problem uses the "Pipeline" data set in the "alr4" library.

A. Create a scatterplot using the Field and Lab variables.

```
field <- pipeline$Field
lab <- pipeline$Lab

plot(field, lab)
```



It appears that the simple linear regression model would be adequate for the data; however, it's important to note that without a transformation, the residuals may have a fanning pattern. By looking at the above plot, it's clear that the observations are becoming more spread as the x-variable increases.

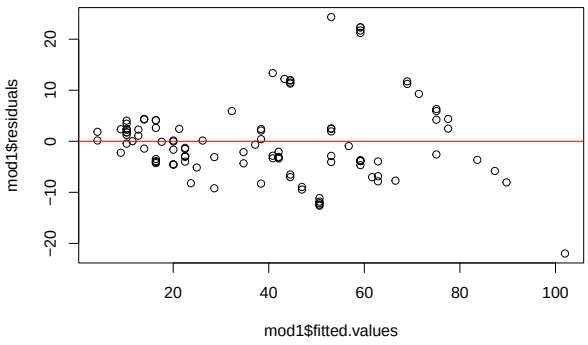
B. Now fit the simple linear regression (SLR) model, get the residual plot, and summarize.

```
mod1 <- lm(lab~field)
summary(mod1)
```

```
##
## Call:
## lm(formula = lab ~ field)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -21.985  -4.072  -1.431   2.504  24.334
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.96750    1.57479  -1.249    0.214
## field        1.22297    0.04107  29.778   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.865 on 105 degrees of freedom
## Multiple R-squared:  0.8941, Adjusted R-squared:  0.8931
## F-statistic: 886.7 on 1 and 105 DF,  p-value: < 2.2e-16
```

```
plot(mod1$fitted.values, mod1$residuals)
abline(0,0, col = 'red')
```



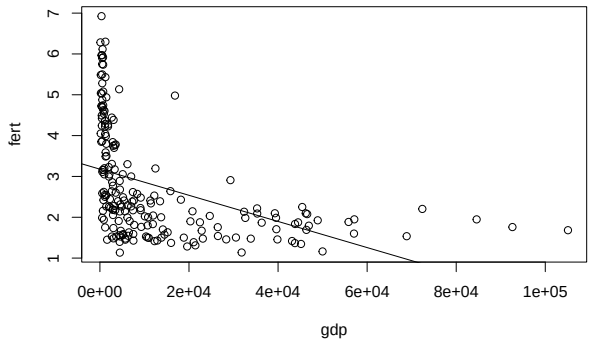
According to the output of the summary() function, the model appears to fit very well; however, when analyzing the plot of fitted values against the residuals, it becomes clear that the equal variance assumption fails. There is a very clear fanning pattern, meaning that the residuals are getting farther away from zero as the fitted values increase.

Problem 2

This problem uses the "UN11" data from the "alr4" package.

A. Plot fertility against ppgdp. Fit a SLR model, and the least squares line onto the plot.

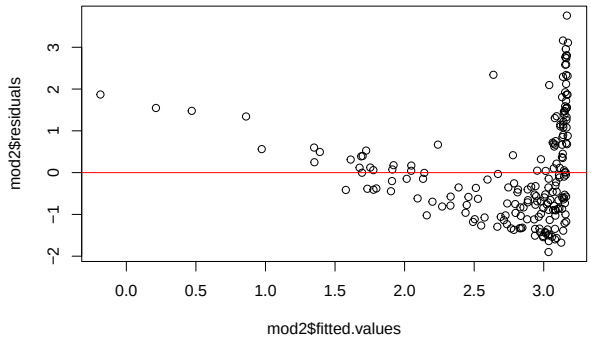
```
fert <- UN11$fertility
gdp <- UN11$ppgdp

plot(gdp, fert)

mod2 <- lm(fert~gdp)
abline(mod2)
```



This does not appear to be a good model because a SLR is not enough to fit the data. It appears that the scatter plot follows a pattern that is similar to those of rational functions, that is $f(x) = \frac{1}{x}$.

B. Plot the residuals against the fitted values.

```
plot(mod2$fitted.values, mod2$residuals)
abline(0,0, col = 'red')
```



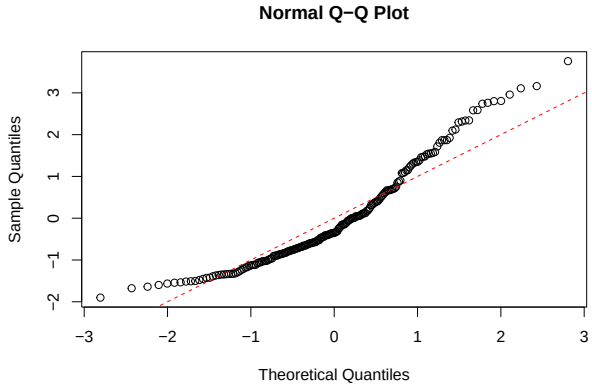
There are two very clear issues with the above residual vs. fit plot:

- The error terms do not appear to have a constant variance.
- There is a lack of linearity present in the model. A line would not be the best fit for this data.

With the above points, it is clear that a transformation might be necessary on the data.

C. Study the normality assumption using a Q-Q probability plot.

```
qqnorm(mod2$residuals)
abline(0,1, col = 'red', lty = 2)
```



It is clear that the SLR model does not satisfy the normality assumption. The residuals do not fall in line with the theoretical quantiles of the normal distribution. The left tail of the above plot is light, while the right tail is heavy.

D. Use the Shapiro-Wilk test on R to check if the residuals are normally distributed.

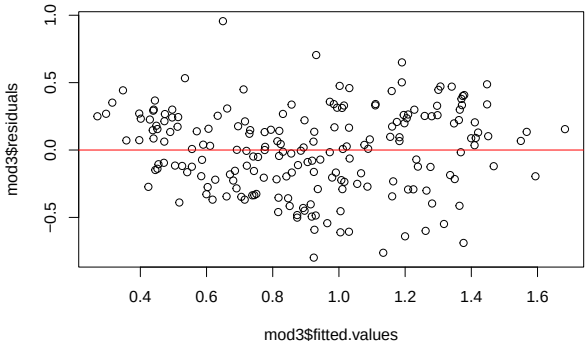
```
shapiro.test(mod2$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  mod2$residuals
## W = 0.92844, p-value = 2.708e-08
```

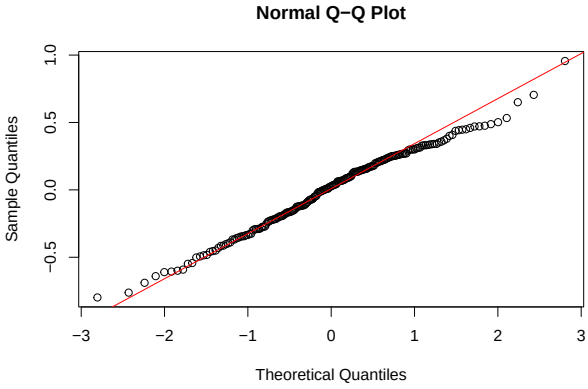
The p-value for the Shapiro Test is $2.7 * 10^{-0.8}$. I would reject H_0 at $\alpha = 0.05$. This means that there is very strong evidence to suggest that the residuals are not normally distributed. This agrees with the conclusion that I made in part C of this problem.

E. Apply the necessary transformation to the data. Use a residual vs fit plot and QQ-plot to verify that the model has been improved.

```
y.trans2 <- log(fert)
x.trans1 <- log(gdp)

mod3 <- lm(y.trans2~x.trans1)

plot(mod3$fitted.values, mod3$residuals)
abline(0,0, col = 'red')
```




```
qqnorm(mod3$residuals)
qqline(mod3$residuals, col = 'red')
```




```
shapiro.test(mod3$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  mod3$residuals
## W = 0.99048, p-value = 0.2128
```

Creating a linear model using a log() transformation on both the explanatory and response variable generated the above plots.

This QQ-plot looks better than the original. The residuals stop falling in line with the theoretical quantiles of the normal distribution as they reach the right hand side; however the Shapiro-Wilk test indicates that we should fail to reject the null hypothesis, meaning that the residuals do seem to somewhat follow the normal distribution.

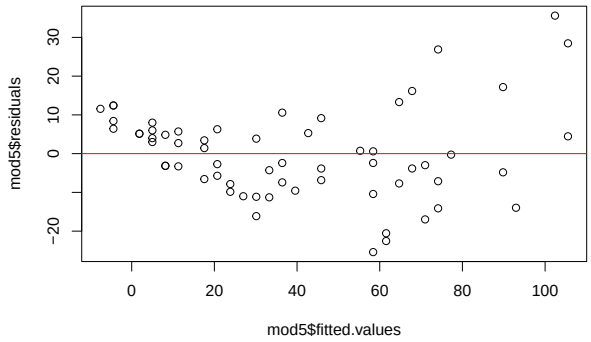
This residual vs fitted values plot looks better than before. The errors appear random, and their variance appears constant.

Problem 3

This problem uses the **stopping** data in the alr4 package.

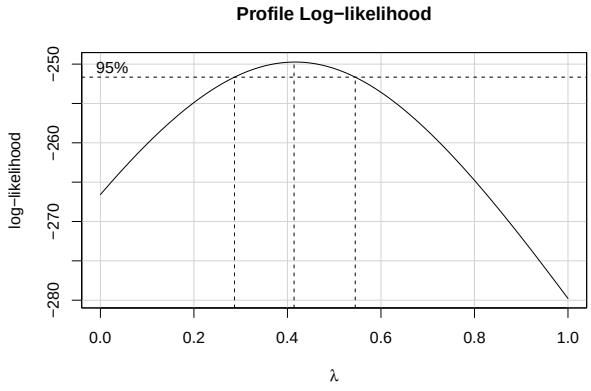
A. Using Speed as the only predictor and Distance as the response, find an appropriate transformation on Distance that can linearize the regression.

```
mod5 <- lm(Distance~Speed, data = stopping)

plot(mod5$fitted.values, mod5$residuals)
abline(0,0, col = 'red')
```



Notice that a transformation appears to be necessary because the residuals have a quadratic type of pattern.

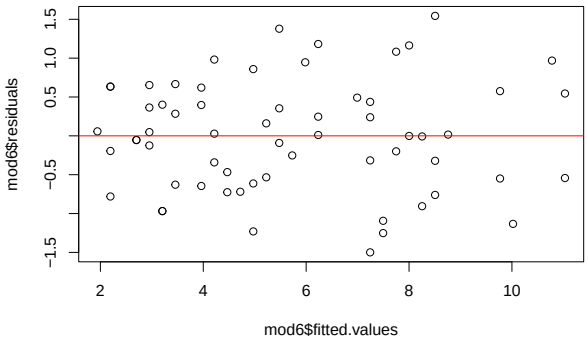
```
# use of Box-Cox to find lambda for Y
mod.boxcox = boxCox(mod5, lambda = seq(0, 1, length = 10))
```




```
mod.boxcox$x[which.max(mod.boxcox$y)]
```

```
## [1] 0.4141414
```

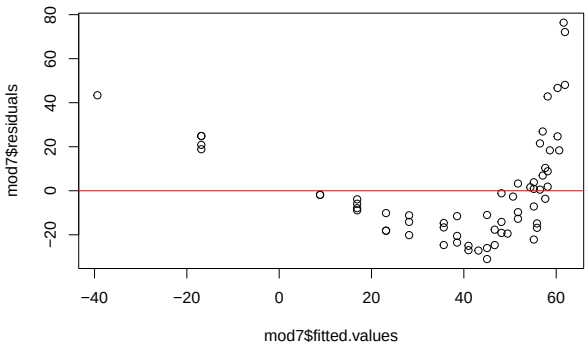
```r
# applying lambda = 0.5
Distance1 <- stopping$Distance^0.5
mod6 <- lm(Distance1~Speed, data = stopping)

plot(mod6$fitted.values, mod6$residuals)
abline(0,0, col = 'red')
```



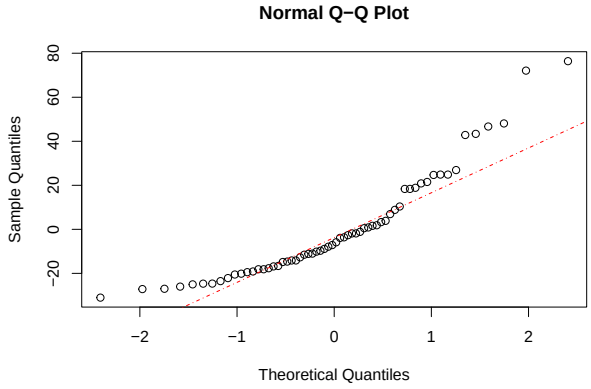
Applying a transformation of $\lambda = 0.5$ to the Distance response variable results in the above fitted values vs. residuals plot. This plot is much better than the first as there is no visible pattern in the residuals. The equal variance assumption also appears to be met.

B. Transform the predictor Speed using $\lambda = \{-1, 0, 1\}$ and show that neither of these are suitable transformations.

```r
mod7 <- lm(Distance~I(Speed^-1), data = stopping) # lambda of -1
mod8 <- lm(Distance~1, data = stopping)  # lambda of 0
mod9 <- lm(Distance~Speed, data = stopping)       # lambda of 1

plot(mod7$fitted.values, mod7$residuals)
abline(0,0, col = 'red')
```

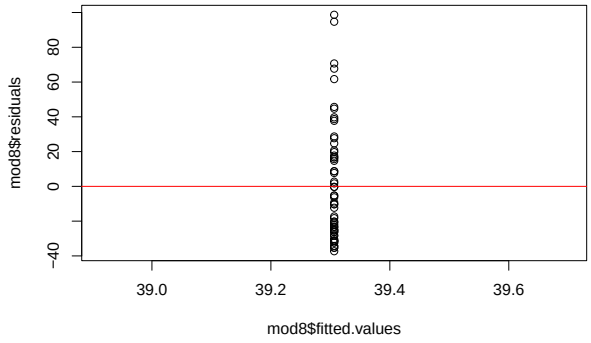



```r
qqnorm(mod7$residuals)
qqline(mod7$residuals, col = 'red', lty = 4)
```

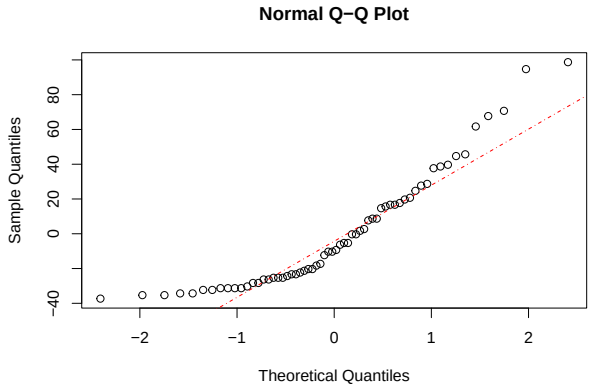


The above output corresponds to the transformation where $\lambda = -1$. This is not a good transformation because the fitted values vs. residuals plot has a very apparent non-linear pattern. There appears to be a strange cluster on the right hand side of this plot that has a quadratic shape. The QQ-Normal plot also shows that normality doesn't appear to be satisfied in this case.

```
plot(mod8$fitted.values, mod8$residuals)
abline(0,0, col = 'red')
```

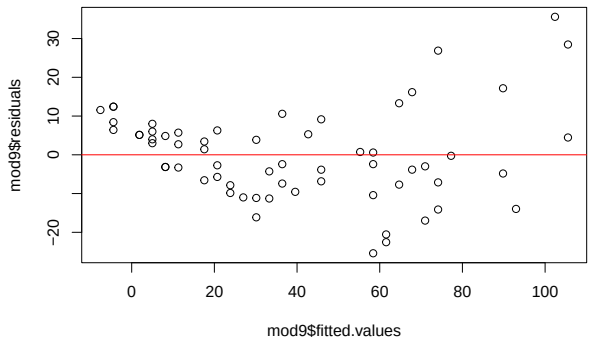



```
qqnorm(mod8$residuals)
qqline(mod8$residuals, col = 'red', lty = 4)
```

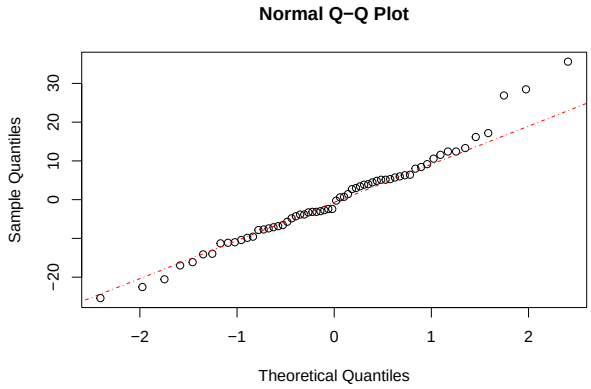


The above output corresponds to the transformation where $\lambda = 0$. This is not a good model because there is a clear lack of linearity. This residual vs. fitted plot is just a vertical line, meaning that predictors are missing. The QQ-plot also appears to not satisfy the normality assumption.

```
plot(mod9$fitted.values, mod9$residuals)
abline(0,0, col = 'red')
```




```
qqnorm(mod9$residuals)
qqline(mod9$residuals, col = 'red', lty = 4)
```

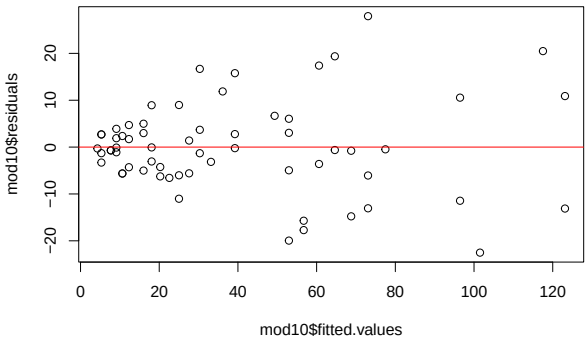


The above plot corresponds to the transformation where $\lambda = 1$, essentially leaving the predictor variable unchanged. The normality assumption appears to be met by looking at the QQ-plot; however, the fitted values vs. residuals plot appears to lack linearity. There appears to be a slight quadratic trend in the errors. The errors also appear to be fanning out, meaning that the variance is not constant.

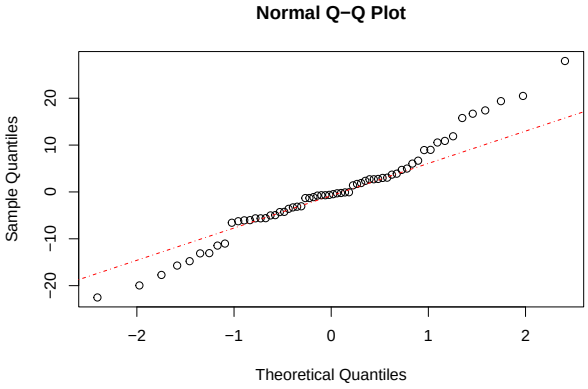
None of the above transformations on only the predictor variable are satisfactory.

C. Show that $\lambda = 2$ would be a good transformation on the predictor variable Speed.

```
mod10 <- lm(Distance~Speed+I(Speed^2), data = stopping)

plot(mod10$fitted.values, mod10$residuals)
abline(0,0, col = 'red')
```

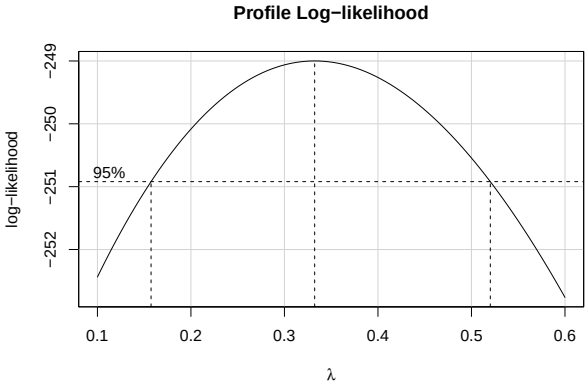



```
qqnorm(mod10$residuals)
qqline(mod10$residuals, col = 'red', lty = 4)
```



Notice that there is a fanning pattern present in the fitted values vs residuals plot, meaning that a Y transformation is necessary.

```
bc1 <- boxCox(mod10, lambda = seq(0.1, 0.6, length = 10))
```

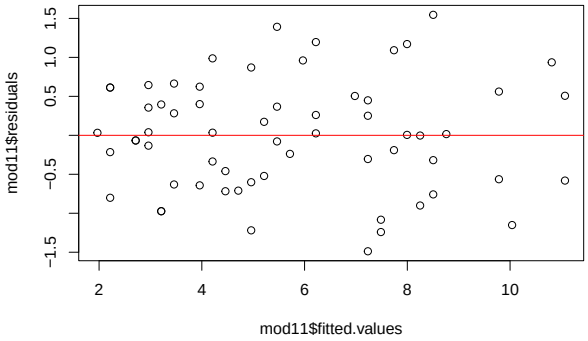


```
bc1$x[which.max(bc1$y)]
```

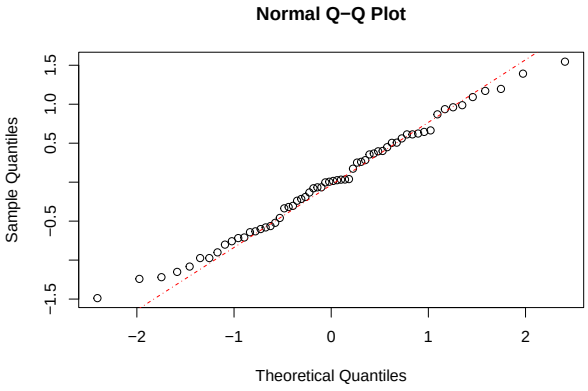
```
## [1] 0.3323232
```

Because 0.5 is in the above 95% confidence interval, we can use $\lambda = 0.5$ for the transformation on Y.

```
dist1 <- stopping$Distance^0.5
mod11 <- lm(dist1~Speed + I(Speed^2), data = stopping)

plot(mod11$fitted.values, mod11$residuals)
abline(0,0, col = 'red')
```




```
qqnorm(mod11$residuals)
qqline(mod11$residuals, col = 'red', lty = 4)
```



With the above transformation done on Y, we can see that the residual vs fit plot is much better, and that normality is kept.