$ID : u1831807$

1 ***(a)*** The mean birth weight among babies born to women with and without hypertension is calculated by the formula $\frac{1}{N}\sum_{i=1}^{N} x_i$,where in our case, the x_i's are each baby's weight and N is the corresponding size of the sample.

Mean of babyweight **with no** hypertension: 3198.904 (number of instances : 72). Mean of babyweight **with** hypertension 2768.208 (number of instances : 428

The code from R that I used for this is:

```
> Babies_weight_hypertension = mean(courseworkData1$bweight
+ [courseworkData1$hyp == "1"])

> Babies_weight_NO_hypertension = mean(courseworkData1$bweight
+ [courseworkData1$hyp == "0"])
```

1***(b)*** The two parameter estimates are given as follows:

The intercept coefficient is 3198.9 . This means that if the hypertension is equal to 0, then the babies' weights are predicted to equal 3198.9. The coefficient of one unit increase in hypertension is -430.7. This means that if hypertension is present, then the estimated change in the babies' weight is -430.7 i.e if there is hypertension then predicted loss in weight of a baby is 430.7. The linear model in mathematical form is represented by the following equation:

$$y = 3198.9 - 430.7x$$

Where y is the baby weight and $x = \{0, 1\}$ is the hypertension coefficient The code from R that I used for this is:

```
> lm1 = lm(courseworkData1$bweight ~ courseworkData1$hyp,
+ data=courseworkData1)
```

1.***(c)*** The code from R that I used to plot this is:

```
> courseworkData1$symbol_key <- ifelse(courseworkData1$hyp == 0, 8, 20)
+ plot(courseworkData1$bweight ~ courseworkData1$gestwks,
+ xlab="Gestation Period", ylab="Baby weight", pch=courseworkData1$symbol_key)
+ legend("bottomright", title="hyp", legend = c("0", "1"), pch = c(8,20))
```

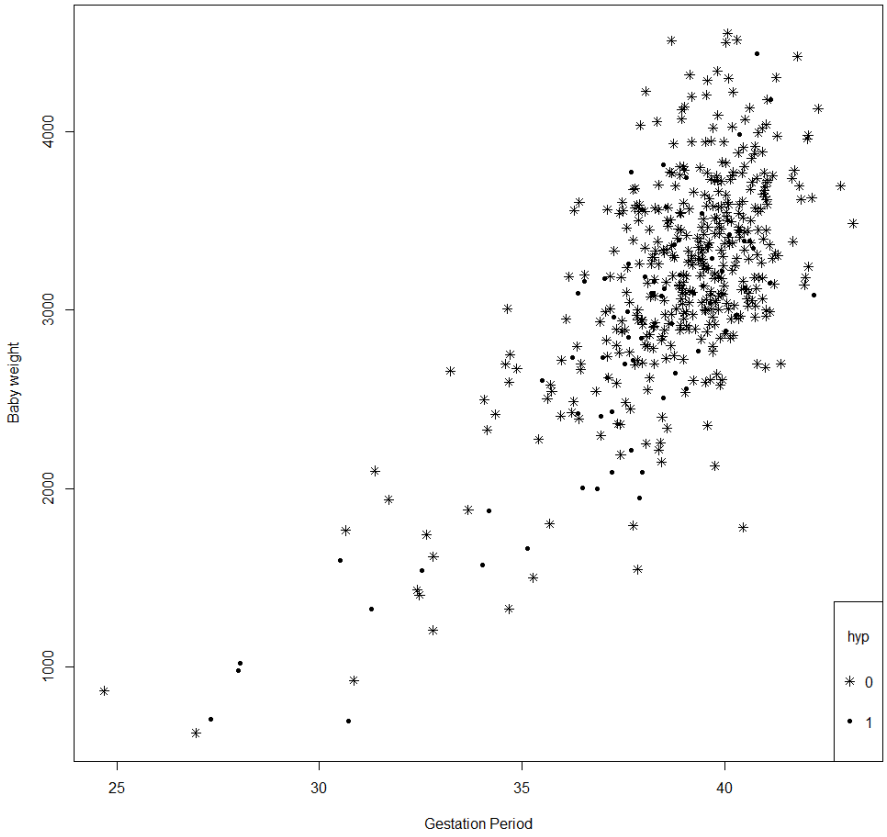

Figure 1: R plot representing the effect hypertension has on the weights of babies

1***(d)*** The hypertension affects the babies' weight less than it affected it in 1(b). The coefficient of 1 unit increase of hypertension is -161.3. In other words, with the presence of hypertension the predicted loss of a baby's weight is 161.3. The reason it is different than the value in 1(b) is because in the previous model we had no other factors(predictor variables) affecting the outcome, and in 1(d), there are 2 more additional variables which have weight on the outcome, hence the weight which hypertension causes to the outcome is less than before. For example, babies born early may be more likely to have hypertension and are also more likely to weigh less. In 1B, this affect was (incorrectly) being attributed to hypertension rather than gestation length.

The code from R that I used for this is:

```
> male = 1*(courseworkData1$sex == "male")
+ lm2 = lm(courseworkData1$bweight ~ courseworkData1$gestwks
```

```
+ + courseworkData1$hyp + male)
```

1(e) The design matrix is represented as follows:

In this model we have $(bw)_i = \alpha + \beta(hyp)_i + \gamma(sex)_i + \delta(gest)_i + \epsilon_i$, so our design matrix X must satisfy $(\mathbf{bw}) = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$

$$\mathbf{X} = \begin{bmatrix} 1 & 38.52 & 0 & 0 \\ 1 & 38.15 & 0 & 0 \\ 1 & 39.79 & 0 & 1 \\ 1 & 38.88 & 1 & 1 \\ 1 & 40.97 & 0 & 0 \end{bmatrix}$$

The first column is the intercept, second column is Gestation period(gestwks) coefficient,third column is Hypertension(hyp) coefficient, fourth column is Sex coefficient, where male's are 1's and females are 0's

The code from R that I used for this is:

```
> bw.na <- bw[is.na(gest)==F]
+ hyp.na <- hyp[is.na(gest)==F]
+ gest.na <- gest[is.na(gest)==F]
+ sex.na <- sex[is.na(gest)==F]
+ sexmale.na <- 1*(sex.na=="male")
+ X <- cbind(rep(1,length(bw.na)), hyp.na, sexmale.na, gest.na)
+ head(X)
```

1***(f)*** Taking the linear model created in 1(d), and checking the summary function of that linear model, we can easily check that the variance of the errors is 190926.2. Furthermore, taking the variance covariance matrix of the linear model, we can check the hypertension variance, which is 3311.9146

The code from R that I used for this is:

```
> vcov(lm2)[2,2] # For the hypertension variance
+ summary(lm2)$sigma^2 # For the variance of the errors
```

1(g) The multiple linear regression model is given by:

$$y = -4337.5 + 191x_1 - 161.3x_2 + 196.8x_3$$

Where -4337.5 is the y-intercept, x_1 denotes the gestation period in weeks,x_2 denotes hypertension value (0 or 1) and x_3 is the indicator whether the baby is a male or female, in which case it takes values 1 or 0 respectively.

The expected birth weight of a female child born at 40 weeks to a mother with hypertension is calculated the following way:

$$y = -4337.5 + 191x_1 - 161.3x_2 + 196.8x_3$$
$$y = -4337.5 + 191(40) - 161.3(1) + 196.8(0)$$
$$y = 3141.2$$

Hence, 3141.2 is the expected birth weight of a female child born at 40 weeks to a mother with hypertension.

The code from R that I used for this is:

```
> coeffs = coefficients(linearMod2)
+ expected_birthweight = coeffs[1] + 1*coeffs[2] + 40*coeffs[3] +0*coeffs[4]
+ expected_birthweight
```

1***(h)*** To check whether hypertension affects the birth weight indirectly by reducing the gestation period, I have created a linear model where gestation period,g is the outcome variable and hypertension,h, and sex , s are the predictor variables. The parameters are 38.86 for the y-intercept and the coefficient for one unit increase in hypertension is -1.43, and 0.12 for the sex coefficient. The reason I include sex in the linear model is to avoid a potential Simpson's paradox (I'd like to make a conclusion for both male's and female's cases). What the hypertension's coefficient means that hypertension does, in fact, reduce gestation period by 1.43 weeks. Mathematical form of linear model:

$$g = 38.86 - 1.43h + 0.128s$$

Furthermore, creating a linear model where birth weight is the outcome variable and the predictor variables are gestation period and sex. We have the following parameters : -4564 as the y-intercept and the coefficient of one unit increase in gestation period is 196.3, and 189.9 for sex. $w = -4564 + 196.3g + 189.9s$ is the linear model. This means that if there is one more week of gestation period then the birth weight increases by 196.3. If there is a 1 week decrease in gestation period, then clearly the weight decreases by 196.3.

So we see that if the gestation period decreases with 1.43, then the birth weight decreases by $1.43(196.3) = 280.709$. So we see that there is an indirect cause of babies being born with less weight due to having lower gestation periods caused by maternal hypertension.

Now I make a model where birth weight is the outcome, but the explanatory variables are sex and hypertension (this tests the direct impact)

$$w = 3087.9 - 448.1h + 215.0s$$

Presence of hypertension decreases the weight by 448.1

Therefore, I would say to the doctor that the direct effect of hypertension on baby weight is stronger than the indirect effect. I would advise more research to be conducted (statistical and medical) in order to make a more concrete conclusion.

The code from R that I used for this is:

```
> lm3 = lm(gestwks ~ hyp + sex)
+ lm4 = lm(bweight ~ gestwks + sex)
+ lm5 = lm(bewight ~ hyp +sex)
```

2***(a)***

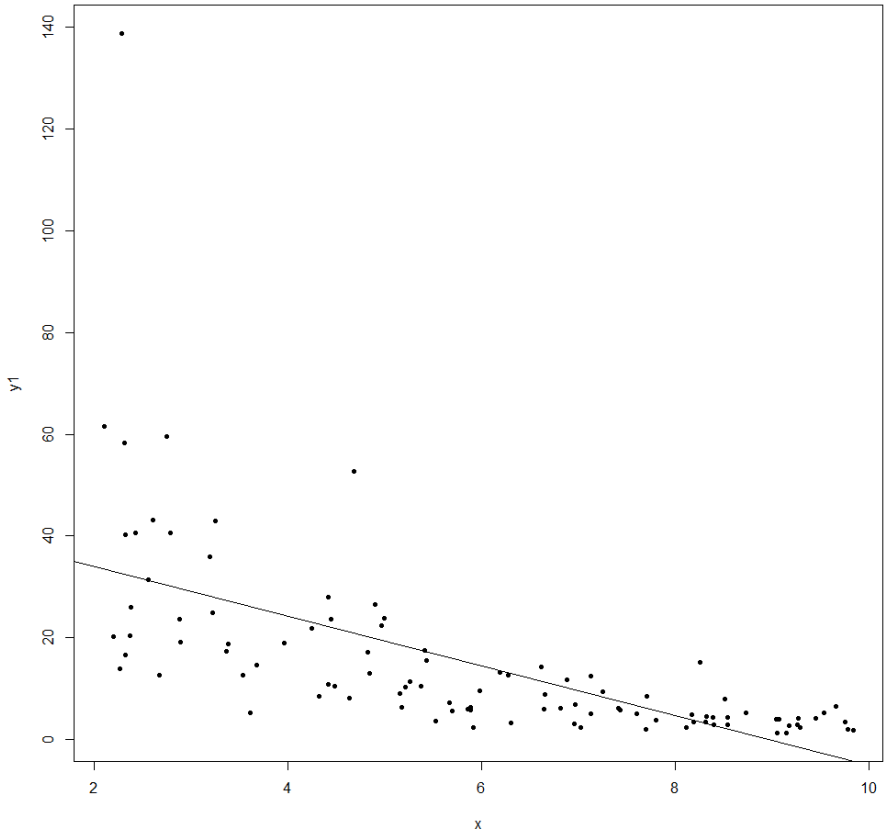

Figure 2: R plot representing the relationship between y_1 as the response and x as the predictor variable, with a fitted line

The code from R that I used to plot this is:

```
> lm = lm(courseworkData2$y1 ~ courseworkData2$x)
+ plot(courseworkData2$y1 ~ courseworkData2$x, pch=20, xlab="x", ylab="y1")
+ abline(lm)
```

2***(b)***

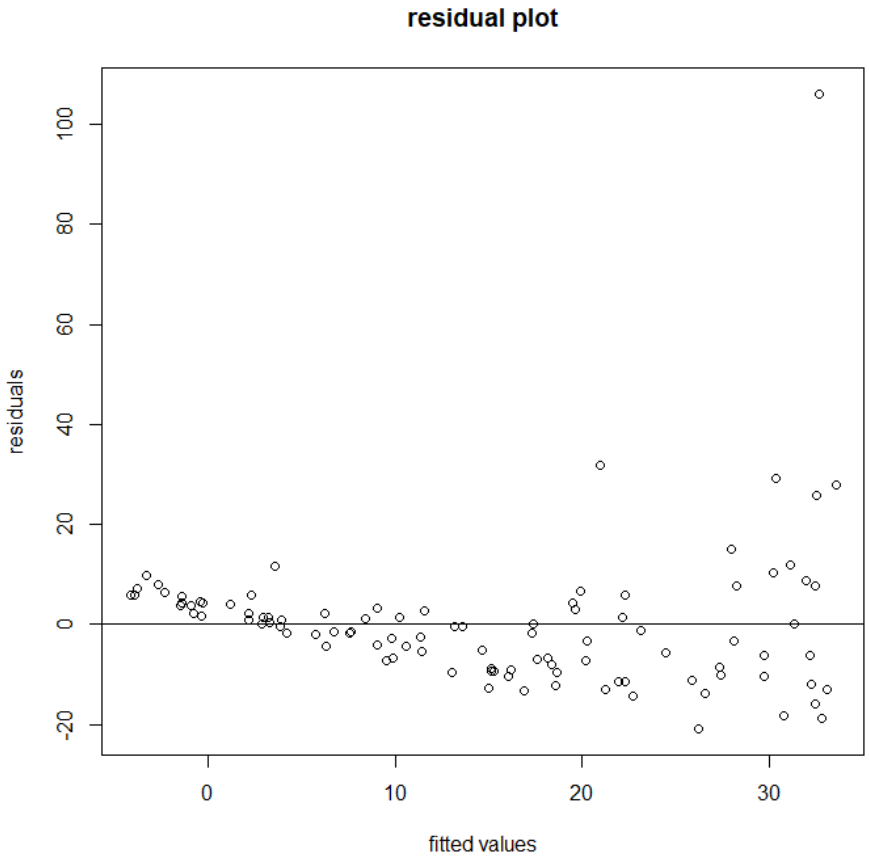

Figure 3: R plot representing the relationship between x and the fitted value of the linear model from 2(a)

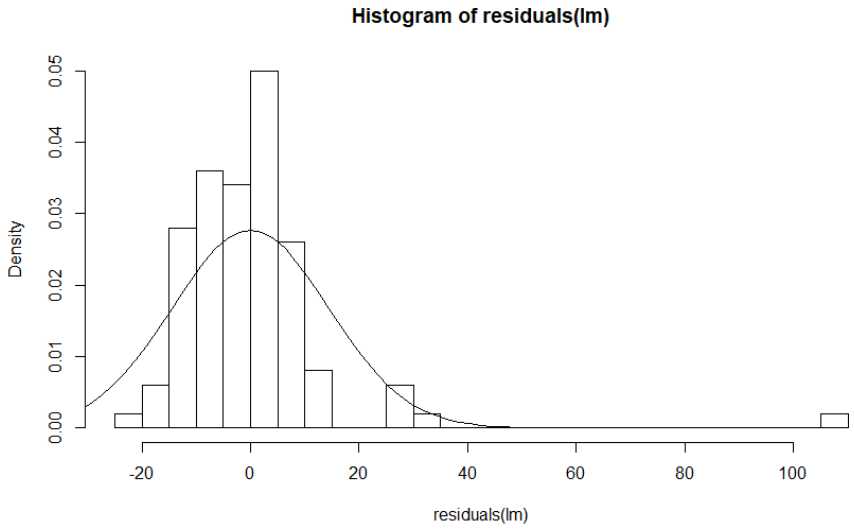

Figure 4: R plot representing the distribution of the residuals of the linear model in 2(a)

As we see above, there is some heteroscedasticity in the residuals. We also see that on the left side of the plot, the values are mainly above the line. And on the right side of the plot, the values are below the line. There is a pattern in our residual plot. This residual plot is not acceptable because the errors appear to have a much greater variance when the fitted value y is large than when y is small, contradicting our assumption of homoscedasticity. There also appears to be a trend that, in the range $y \in [-5, 15]$, the expected value of the error is decreasing with respect to y. This contradicts our assumption that the expected error values are all zero. It is worth noting that there is one outlier, with a y_1 value of 140, but that this is not the reason for the model being unacceptable.

The code from R that I used for this is:

```
> max.resid <- max(abs(residuals(lm)))
+ plot(fitted(lm), residuals(lm) , ylab="residuals", xlab="fitted values",
+ main="residual plot")
+ abline(0,0)

+ hist(residuals(lm), breaks=20, probability=TRUE)
+ max.resid <- max(abs(residuals(lm)))
+ ax <- seq( -max.resid , max.resid , length.out=101)
+ lines(ax, dnorm(ax, 0, sd(residuals(lm))), col="black")
```

2***(c)*** I suggest correcting the model with a transformation of the y_1 variable. The appropriate transformation would be $y_1^{'} = \log(y_1)$

The new residual plot after the transformation is given below.

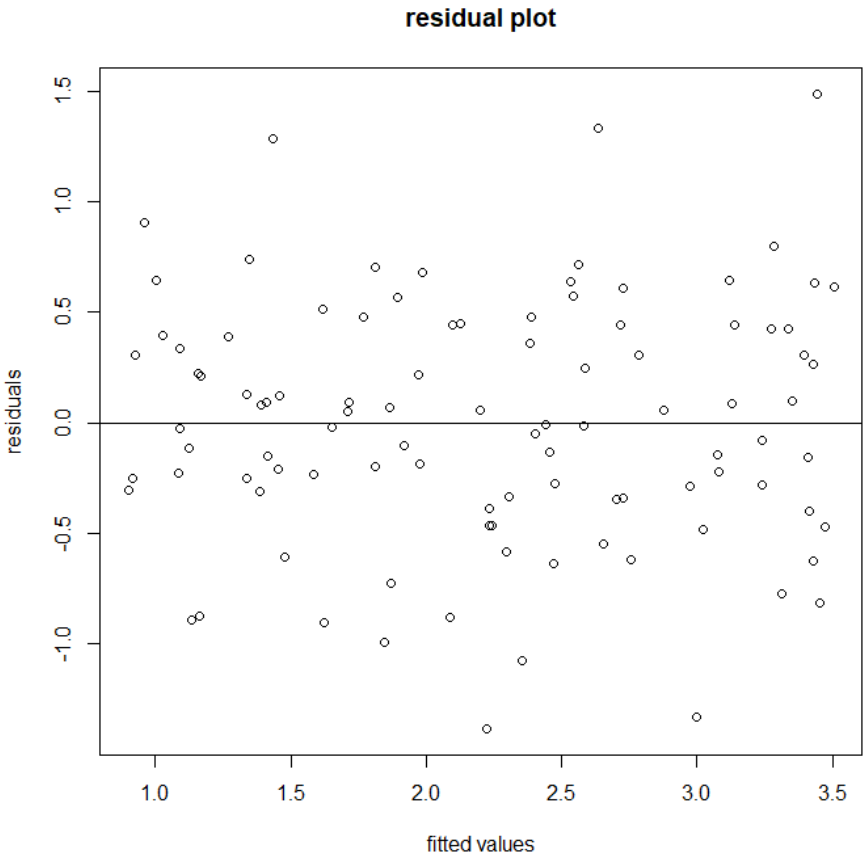

Figure 5: R plot representing the relationship between x and the fitted value of the updated linear model with transformation $y_1^{'}$ =log(y_1)

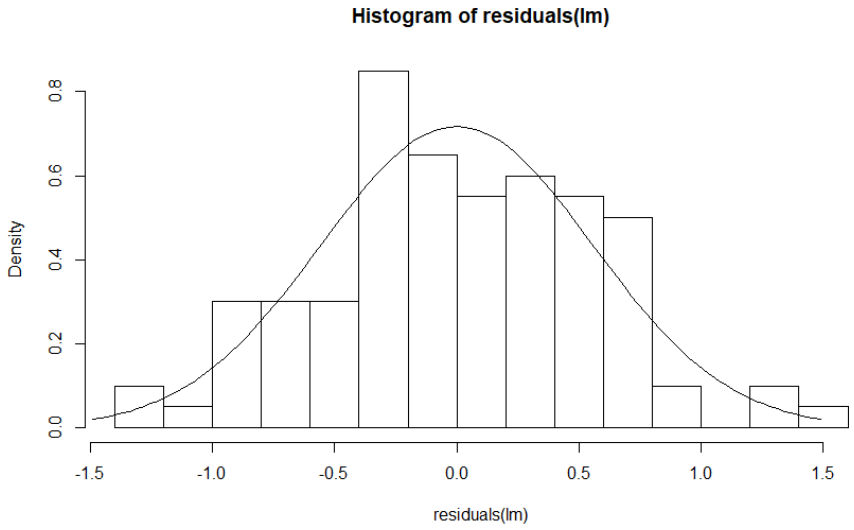

Figure 6: R plot representing density corresponding to the distribution of the residuals of the transformed linear model

We now see that there is not pattern in the residual plot, and the distribution of the residuals is more symmetrically distributed. Hence the plot is acceptable. This residual plot is acceptable because it appears to be homoscalastic and have uniformly an expected value of zero.

The code from R that I used for this is:

```
> lm = lm(log(courseworkData2$y1) ~ courseworkData2$x)
```

2***(d)***

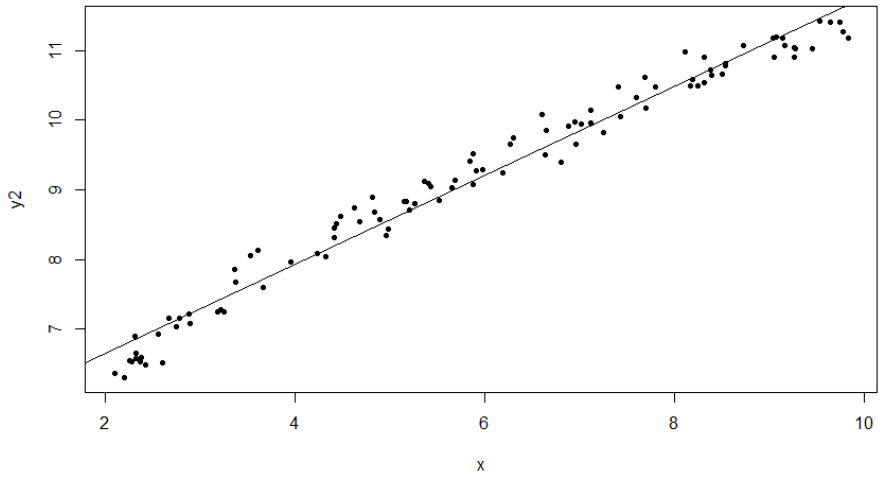

Figure 7: R plot representing the relationship between x and y_2 with a fitted line of the corresponding linear model

Below is the residual plot corresponding to this linear model:

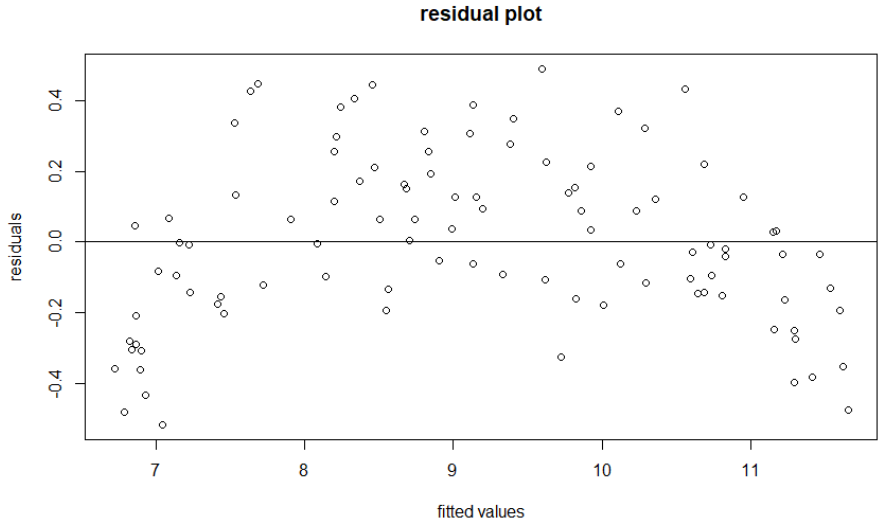

Figure 8: R plot representing the relationship between x and the fitted values of the linear model

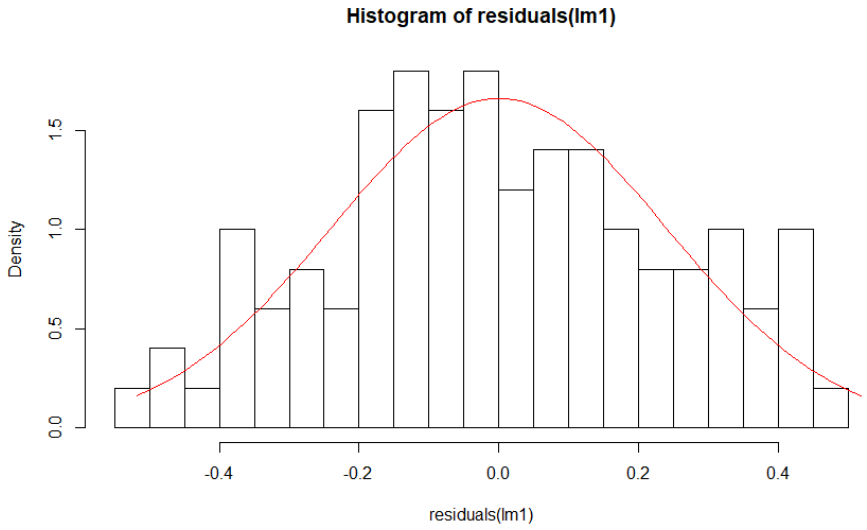

Figure 9: R plot representing the density corresponding to the distribution of the residuals of the linear model

We see that there is a pattern in the plot because clearly the points are distributed in the form of a bell-shaped curve. Also the density is not symmetrically distributed. The model is not acceptable. There is correlation in the residuals, but the variance seems to be constant. Notice that in the scatterplot, the general trend is monotonic. Consequently we can transform the x and/or y values. The error values appear to have negative expected value when the fitted value is in the range $[6.5, 7.5]U[10.5, 11.5]$, and positive expected value when the fitted values are in the range [8,10]. This contradicts our assumption that the expected value of the error is zero uniformly. We can see from the histogram that the variance of the residuals appear to not change with respect to the fitted value (homoscalasticity).

2***(e)*** We make 2 transformations : $y_2^{'} = log_2(y_2)$ and $x^{'} = log(x)$. We then see the following residual plot:

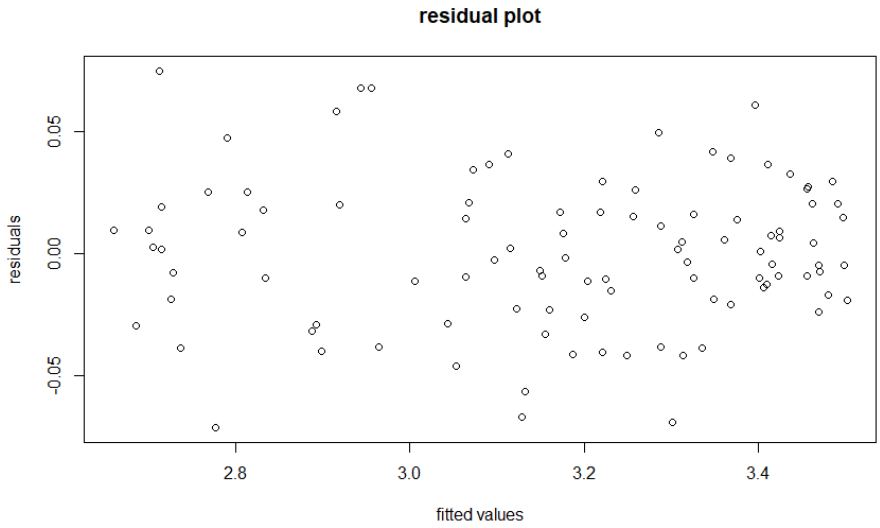

Figure 10: R plot representing the residual plot corresponding to the transformed linear model

We can see that there is no pattern in the residuals.The residuals appear to be homoscedastic and have expected value of zero uniformly. Hence the residual plot is acceptable.

The code from R that I used for this is:

```
> lm1= lm(log2(courseworkData2$y2) ~ log(courseworkData2$x))
+ max.resid <- max(abs(residuals(lm1)))
+ plot(fitted(lm1), residuals(lm1) , ylab="residuals", xlab="fitted values",
+ main="residual plot")
```

2*(f)*

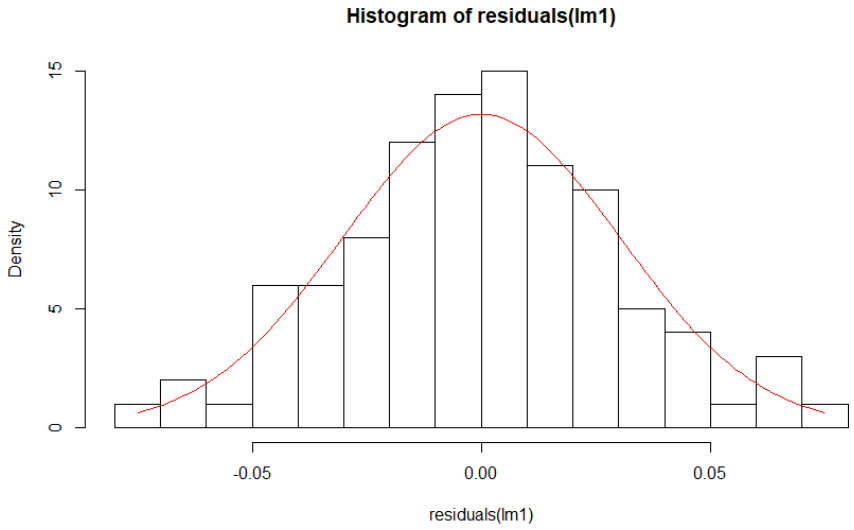

Figure 11: R plot representing the density corresponding to the distribution of the residuals of the tranformed linear model

The code from R that I used for this is:

```
> hist(residuals(lm1), breaks=20, probability=TRUE)
+ max.resid <- max(abs(residuals(lm1)))
+ ax <- seq( -max.resid , max.resid , length.out=101)
+ lines(ax, dnorm(ax, 0, sd(residuals(lm1))), col="red")
```

From the histogram plotted above, we can see that the residuals appear to be normally distributed with mean zero and standard deviation sd(resid.lm6) = 0.03032077