HW11

Luke Geel

4/26/2021

```
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.0 
──
```

```
## ✓ ggplot2 3.3.2     ✓ purrr   0.3.4
## ✓ tibble  3.0.3     ✓ dplyr   1.0.2
## ✓ tidyr   1.1.2     ✓ stringr 1.4.0
## ✓ readr   1.3.1     ✓ forcats 0.5.0
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() 
──
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

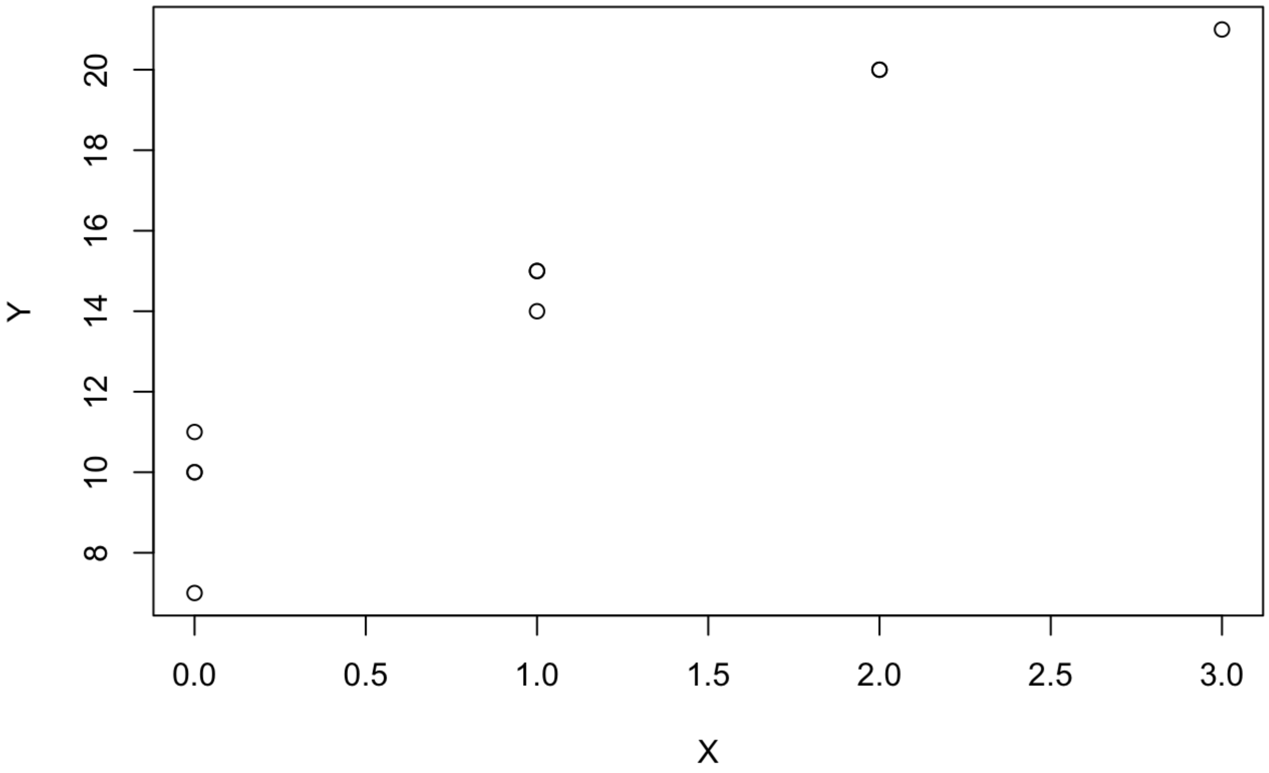
```
loadRData <- function(fileName){
  load(fileName)
  get(ls()[ls() != "fileName"])
}
```

*3.5. Refer to Airfreight breakage Problem 1.21.

```
breakage <- loadRData("/Users/lukegeel/Downloads/breakage_spring2021.RData")
```

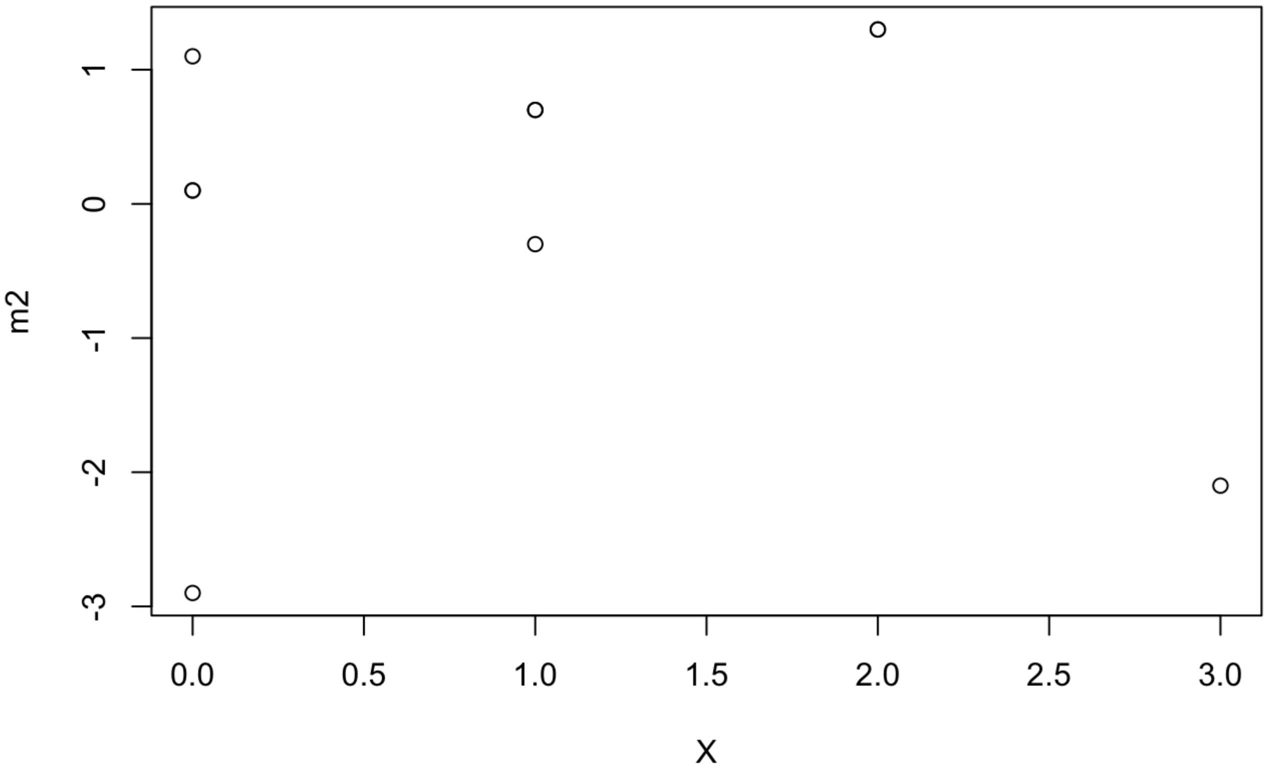
b. The cases are given in time order. Prepare a time plot for the number of transfers. Is any systematic pattern evident in your plot? Discuss. Yes, based on the plot is appears as though as X increases, so does Y.

```
X <- breakage$X
Y <- breakage$Y
plot(X,Y)
```



d. Plot the residuals ei against Xi to ascertain whether any departures from regression model (2.1) are evident. What is your conclusion? Based on the residual plot, there are some departures from the regression where it’s evident that they don’t aling with the model well. Both (0,7) and (3,21) have large residuals meaning they don’t follow the model well.

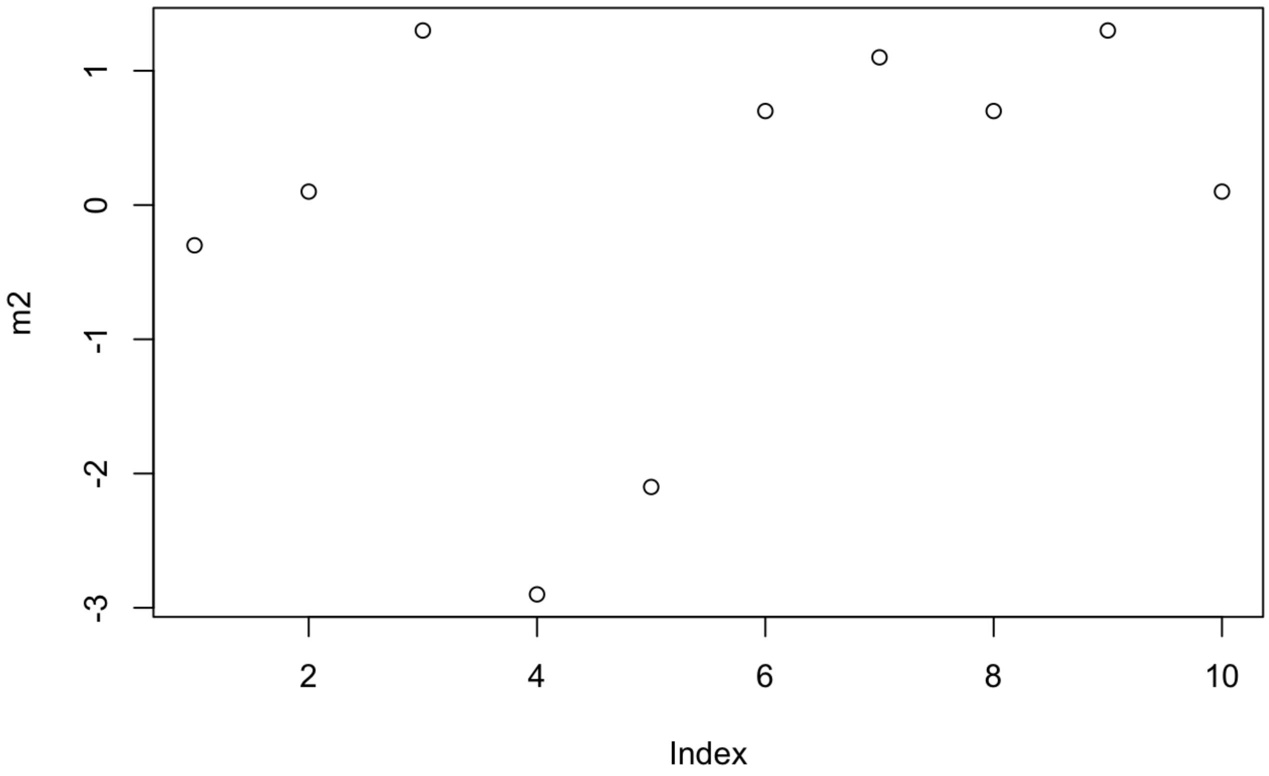
```
m1 <- lm(Y~X)
m2 <- resid(m1)
plot(X,m2)
```



f. Prepare a time plot of the residuals.

What information is provided by your plot? A time plot of residuals indicated a linear or non-linear time-related trend effect. In this case it’s a linear time-related effect.

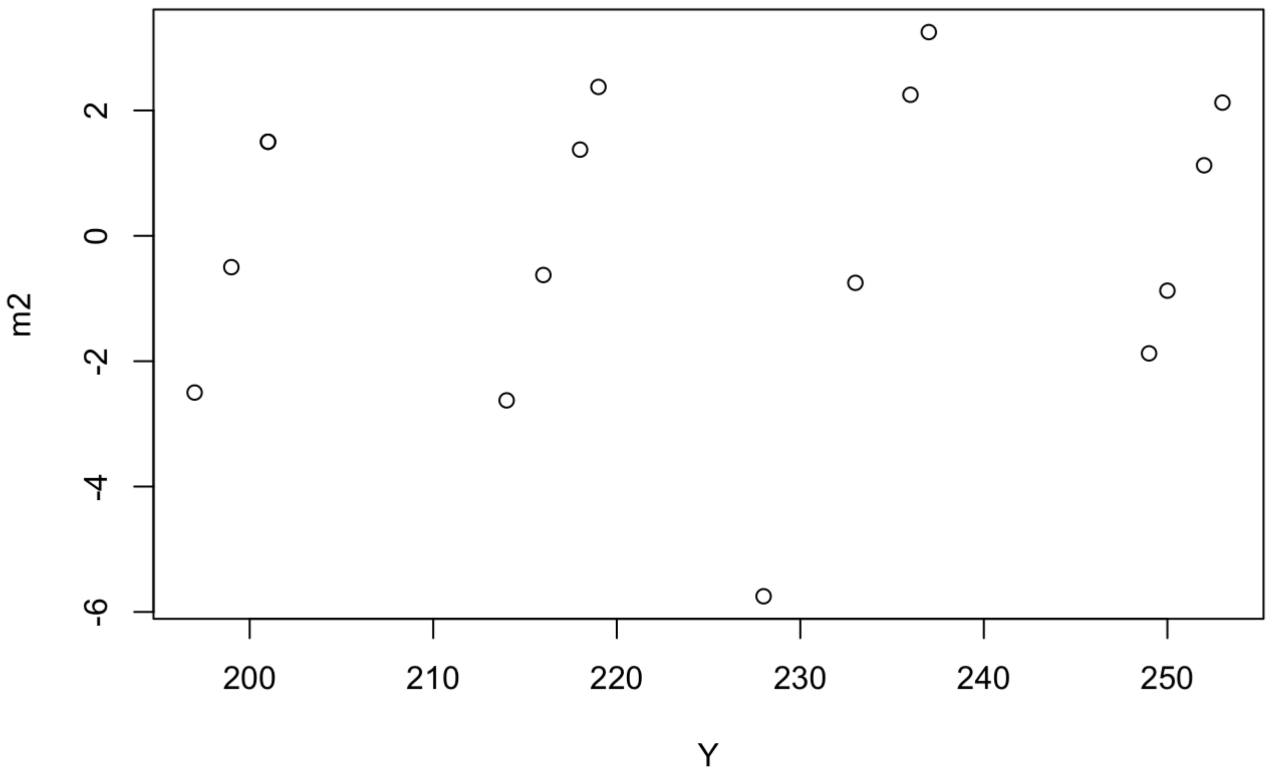
```
plot(m2)
```



3.6. Refer to Plastic hardness Problem 1.22 b. Plot the residuals ei against the fitted values Y; to ascertain whether any departures from regression model (2.1) are evident. State your findings. This plot is very interesting. There are 4 clusters, each with 3 or 4 points and it seems as though each cluster has 1 or 2 points with a residual near 0 and 1 point with a positive residual and 1 point with a negative residual. The point that’s departed furthest from the model would be (32,228)

```
plastic <- loadRData("/Users/lukegeel/Downloads/plastic_spring2021.RData")
X <- plastic$X
Y <- plastic$Y
m1 <- lm(Y~X)
```

```
m2 <- resid(m1)
plot(Y,m2)
```

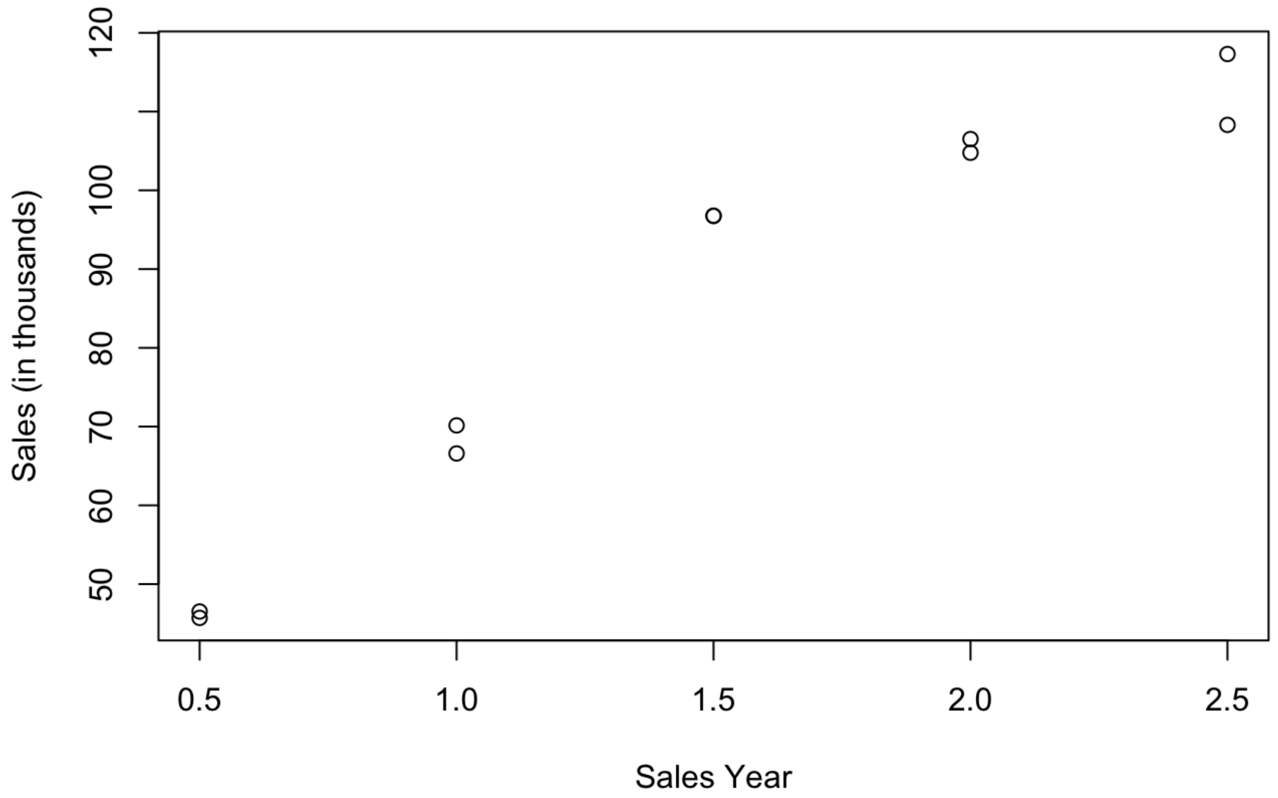


*3.17. Sales growth. A marketing researcher studied annual sales of a product that had been introduced 10 years ago. The data are as follows, where X is the year (coded) and Y is sales in thousands

```
sales <- loadRData("/Users/lukegeel/Downloads/sales_spring2021.RData")
```

a. Prepare a scatter plot of the data. Does a linear relation appear adequate here? Yes, there appears to be a linear relationship.

```
X <- sales$X
Y <- sales$Y
plot(X,Y, xlab = "Sales Year", ylab = "Sales (in thousands)")
```



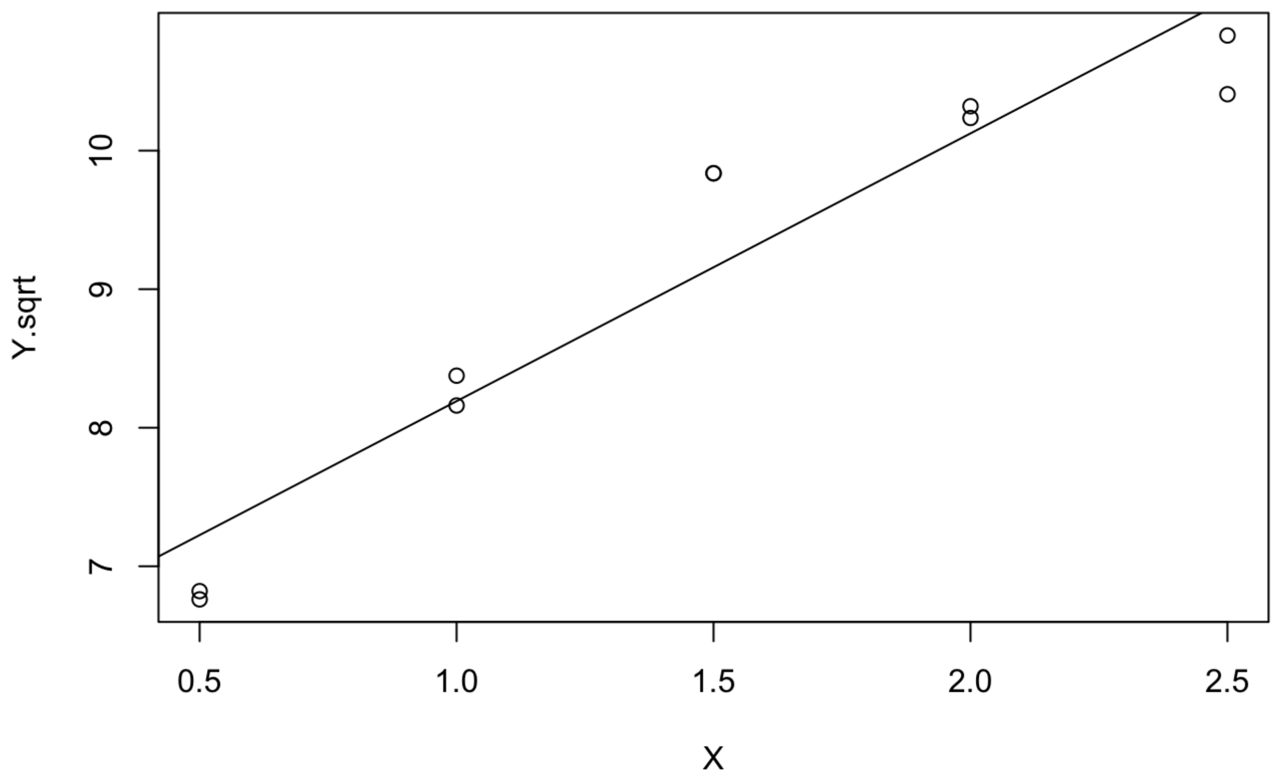
c. Use the transformation Y’ = sqrt(Y) and obtain the estimated linear regression function for the transformed data. sqrt(Y)= 1.933X + 6.258

```
Y.sqrt <-  sqrt(Y)
lm(Y.sqrt~X)
```

```
##
## Call:
## lm(formula = Y.sqrt ~ X)
##
## Coefficients:
## (Intercept)            X
##       6.258        1.933
```

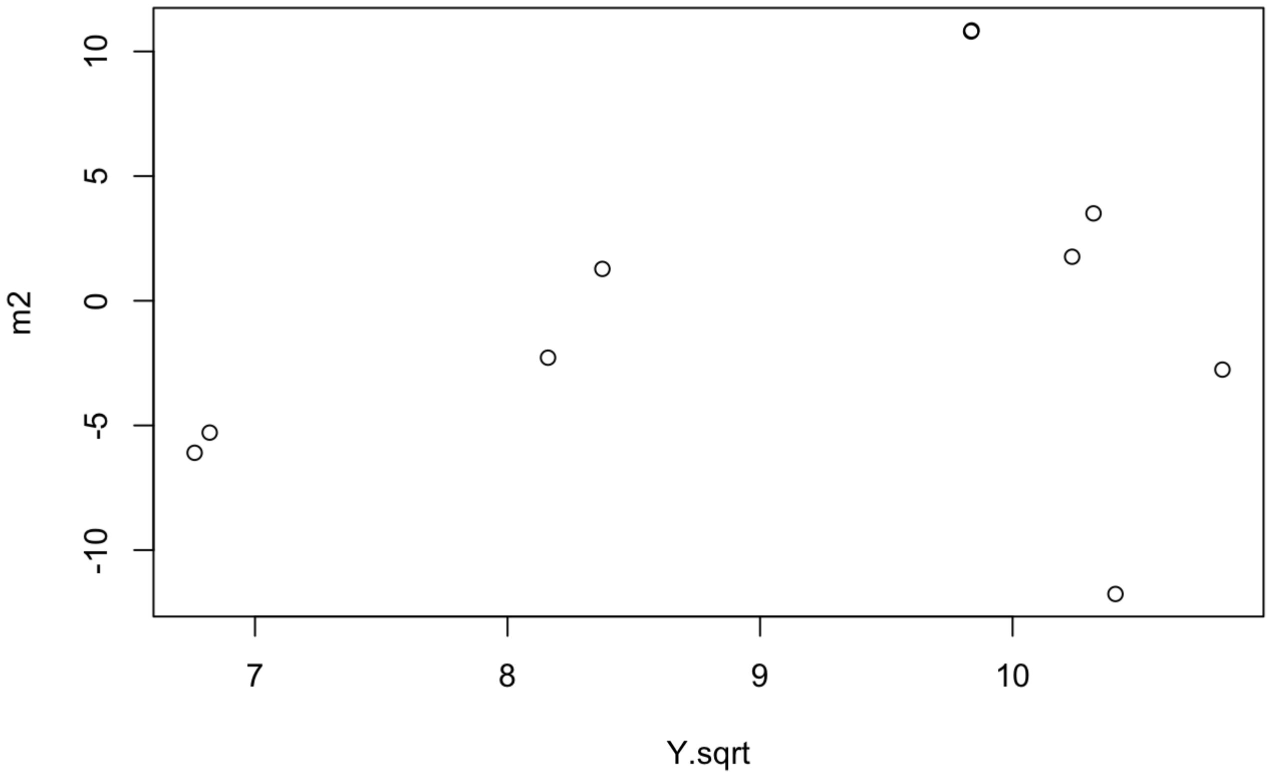
d. Plot the estimated regression line and the transformed data. Does the regression line appear to be a good fit to the transformed data? Yes, the regression line appears to be a great fit to the transformed data.

```
plot(X, Y.sqrt)
abline(lm(Y.sqrt~X))
```



e. Obtain the residuals and plot them against the fitted values. What do your plots show? This plot points to the error in the linear regression aligning with the differences between the expected and observed values. Additionally, since the sum of the residuals is zero it supports the use of the transformation for regression analysis.

```
m1 <- lm(Y~X)
m2 <- resid(m1)
plot(Y.sqrt,m2)
```



f. Express the estimated regression function in the original units Y = 34.13X + 34.74

```
lm(Y~X)
```

```
##
## Call:
## lm(formula = Y ~ X)
##
## Coefficients:
```

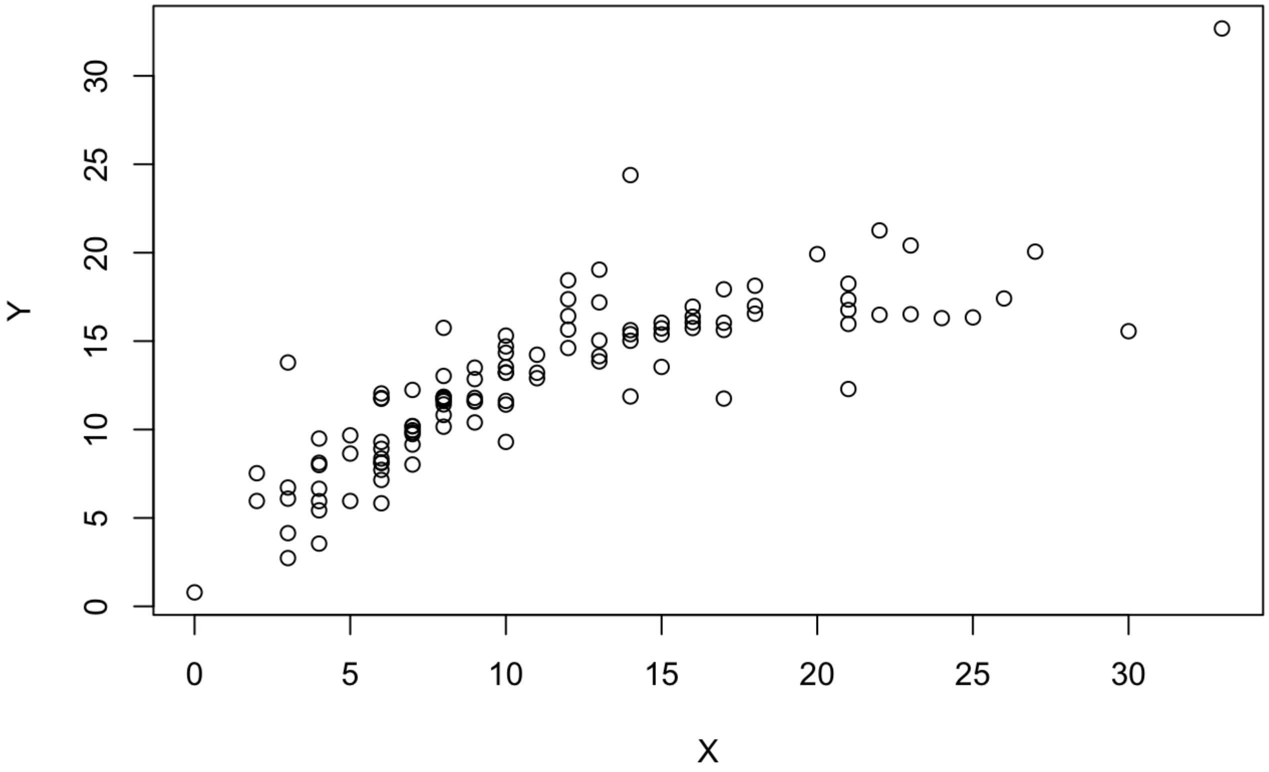
```
## (Intercept)            X
##       34.74        34.13
```

3.18. Production time. In a manufacturing study, the production times for III recent production runs were obtained. The table below lists for each run the production time in hours (Y) and the production lot size (X).

```
production <- loadRData("/Users/lukegeel/Downloads/production_time_spring2021.RData")
```

a. Prepare a scatter plot of the data. Does a linear relation appear adequate here? Would a transformation on X or Y be more appropriate here? Why? A linear relation appears to be adeqate here but a transformation on X would be more approproate due to the outliers.

```
X <- production$X
Y <- production$Y
plot(X,Y)
```



b. Use the transformation X’ = sqrtX and obtain the estimated linear regression function for the transformed data. Y = 3.99sqrt(X) - 0.0318

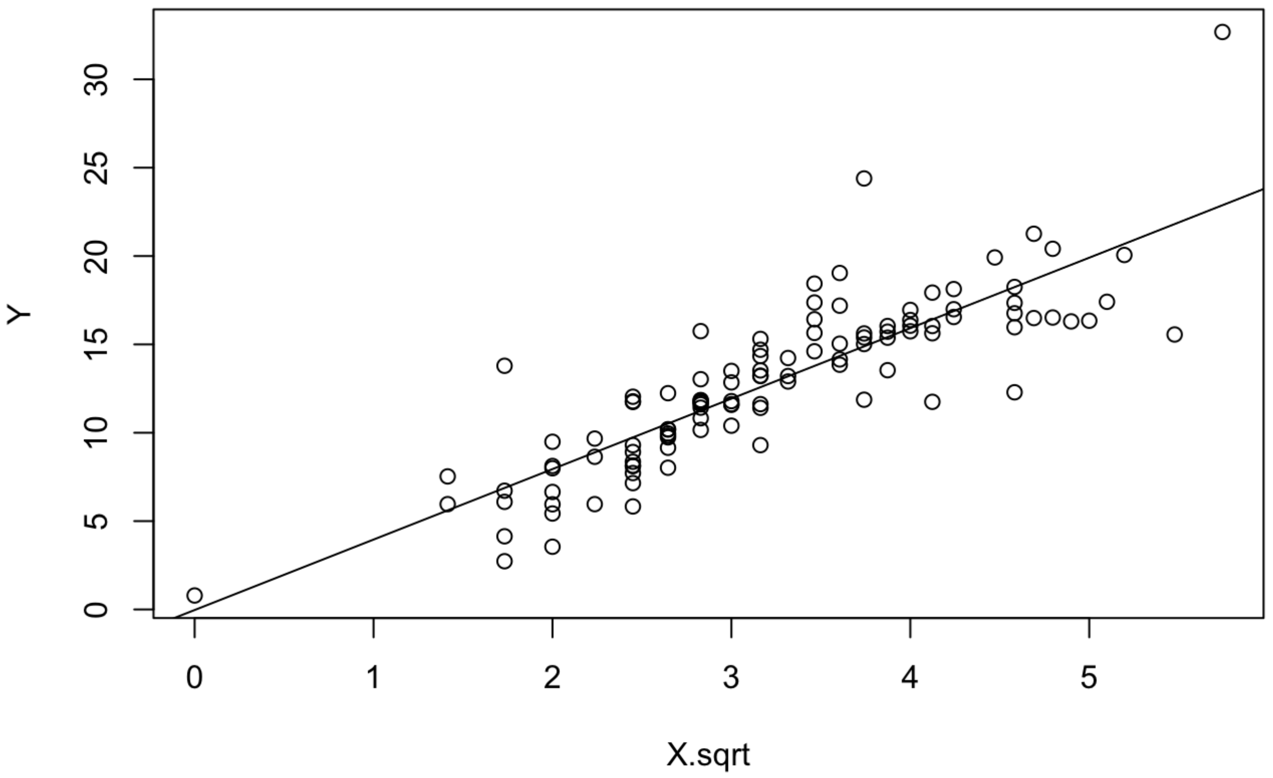
```
X.sqrt <- sqrt(X)
lm(Y~X.sqrt)
```

```
##
## Call:
## lm(formula = Y ~ X.sqrt)
##
## Coefficients:
```

```
## (Intercept)      X.sqrt
##     -0.0318      3.9890
```

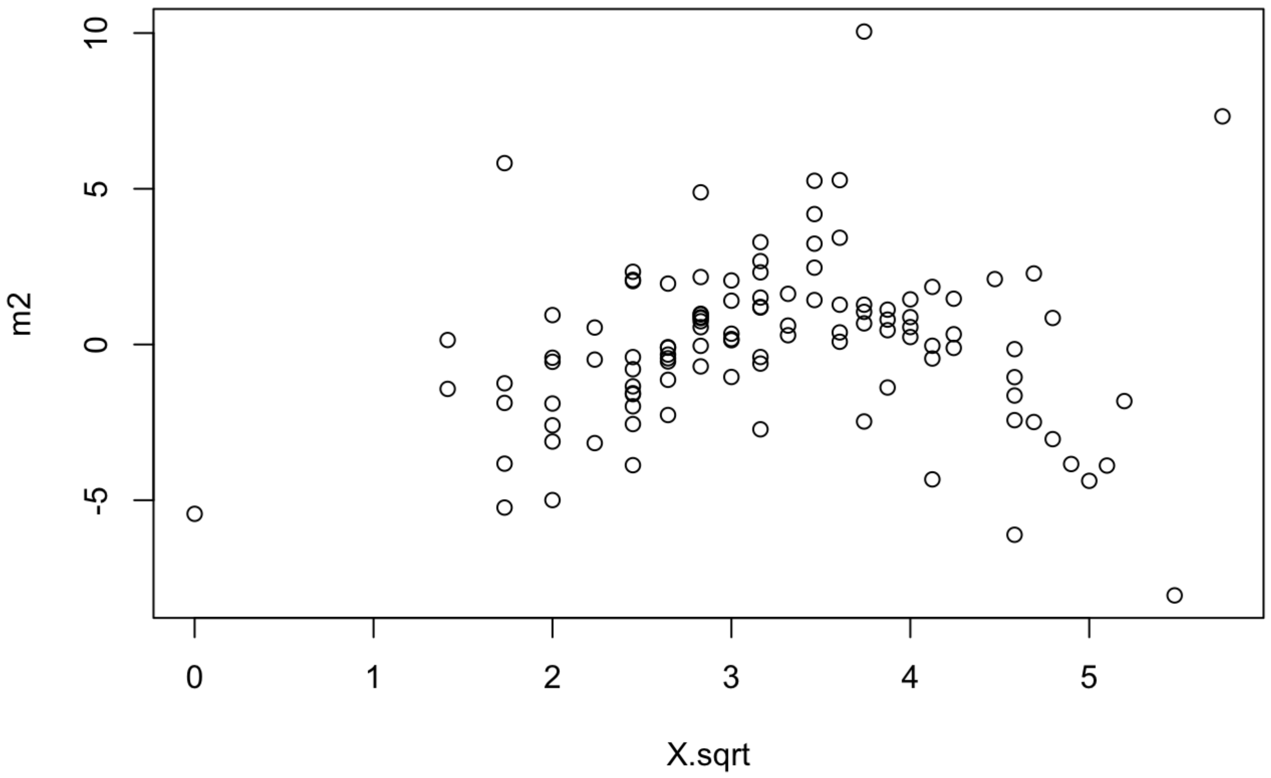
c. Plot the estimated regression line and the transformed data. Does the regression line appear to be a good fit to the transformed data? Yes, the regression line appears to be a great fit for the data

```
plot(X.sqrt,Y)
abline(lm(Y~X.sqrt))
```



d. Obtain the residuals and plot them against the fitted values. What do your plots show? This plot shows that when sqrt(x) is between 2 and 5 the model fits the data well but there are some outliers when sqrt(x) is 0 and greater than 5.

```
m1 <- lm(Y~X)
m2 <- resid(m1)
plot(X.sqrt,m2)
```



e. Express the estimated regression function in the original units. Y = 0.58X + 6.23

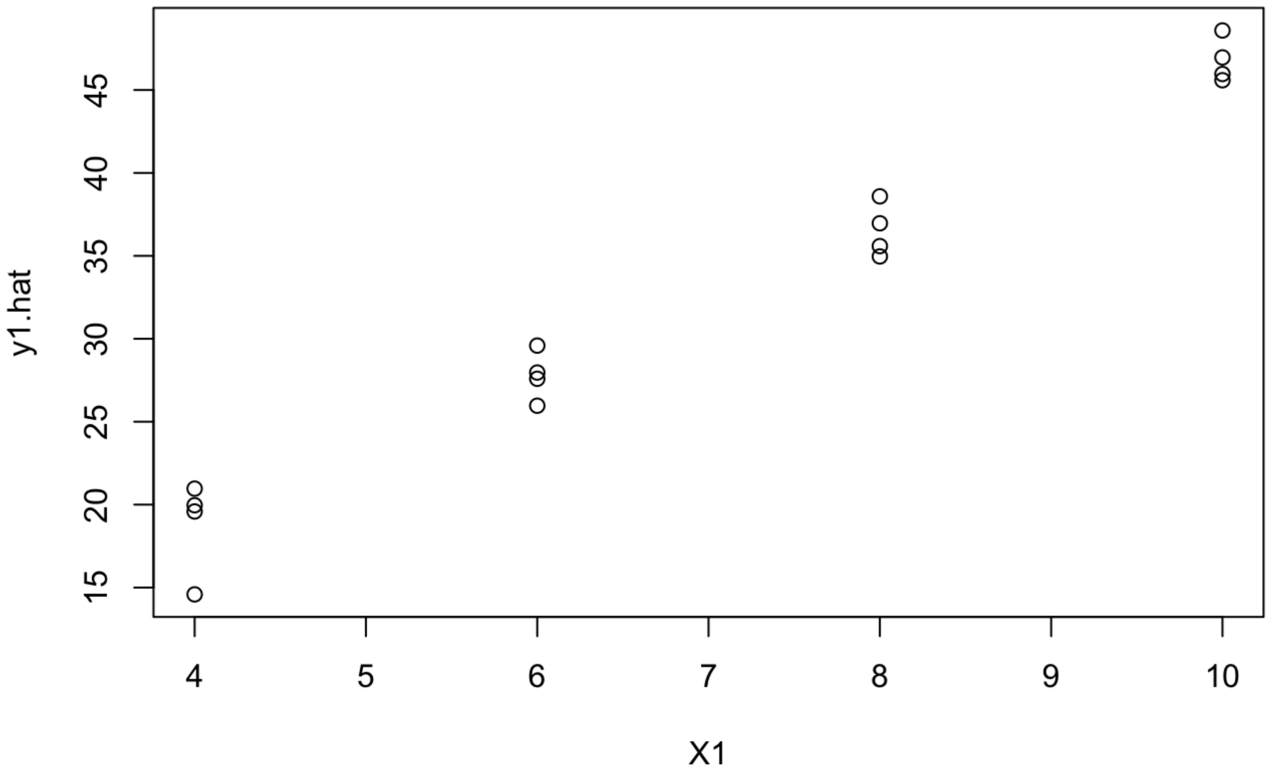
```
lm(Y~X)
```

```
##
## Call:
## lm(formula = Y ~ X)
##
## Coefficients:
## (Intercept)            X
##      6.2272       0.5796
```

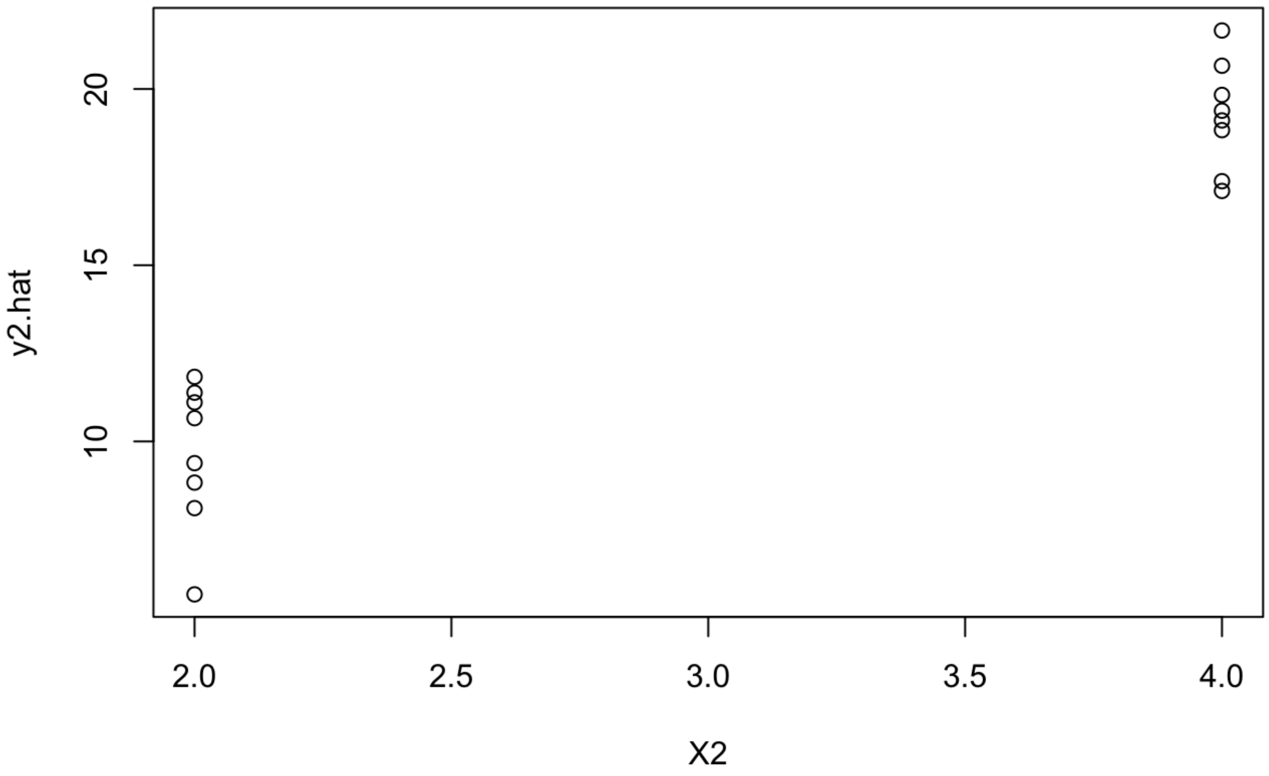
10.5. Refer to Brand preference Problem 6.5b. a. plot the Yi −b0 −b2Xi2 against Xi1 and Yi − b0 − b1Xi1 against Xi2, where b0, b1, and b2 are the estimated regression coefficients obtained by fitting a normal errors multiple linear regression model to the response Y with X1 and X2 as predictors

```
brand <- loadRData("/Users/lukegeel/Downloads/brand_preference_spring2021.RData")
X1 <- brand$X1
X2 <- brand$X2
Y <- brand$Y
lm <- lm(Y~X1+X2)
b0=34.788
b1=4.638
b2=4.812

y1.hat = Y - b0 - b2*X2
y2.hat = Y - b0 - b1*X1
plot(X1, y1.hat)
```



```
plot(X2, y2.hat)
```



b. Do your plots in part (a) suggest that the regression relatioLlships in the fitted regression function in problem 6.5b are inappropriate for any of the predictor variables’? Explain.
No. Based on the plots both predictor variables are appropriate for the function as the sum of residuals for both are zero.

10.6. Refer to Grocery retailer problem 6.9.

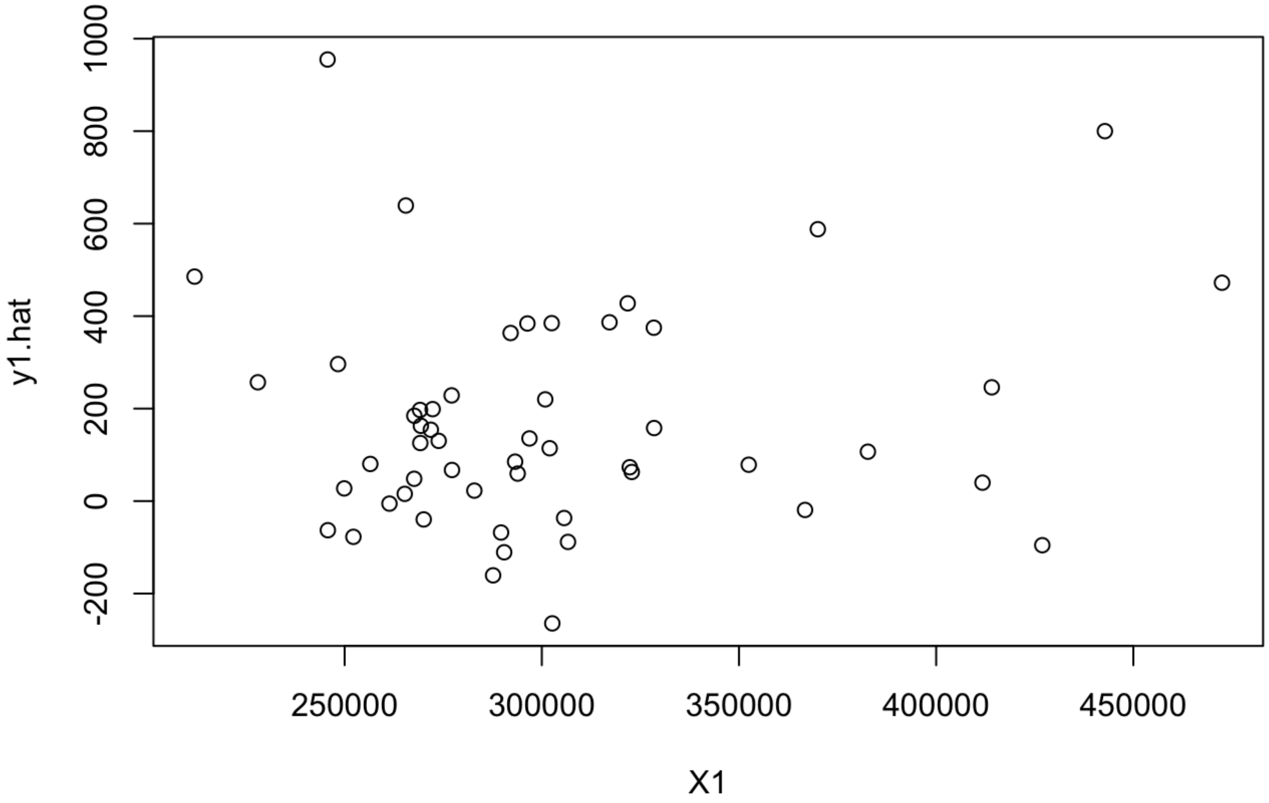
```
grocery <- loadRData("/Users/lukegeel/Downloads/grocery_spring2021.RData")
```

b. plot the Yi −b0 −b2Xi2 against Xi1 and Yi − b0 − b1Xi1 against Xi2, where b0, b1, and b2 are the estimated regression coefficients obtained by fitting a normal errors multiple linear regression model to the response Y with X1 and X2 as predictors.

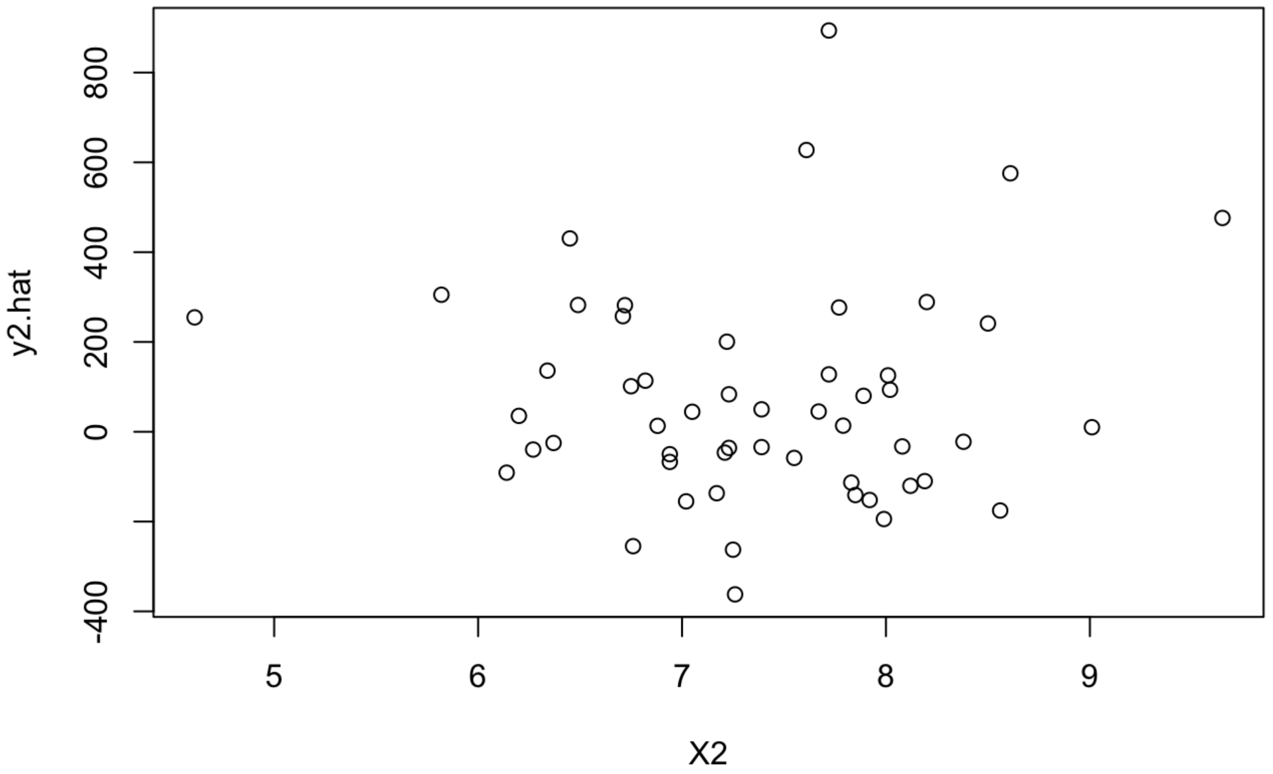
```
X1 <- grocery$X1
X2 <- grocery$X2
Y <- grocery$Y
lm <- lm(Y~X1+X2)
lm
```

```
##
## Call:
## lm(formula = Y ~ X1 + X2)
##
## Coefficients:
## (Intercept)           X1           X2
##   4.134e+03    5.590e-04    9.851e+00
```

```
b0=4.134e+03
b1=5.590e-04
b2 = 9.851e+00
y1.hat = Y - b0 - b2*X2
y2.hat = Y - b0 - b1*X1
plot(X1, y1.hat)
```



```
plot(X2, y2.hat)
```



c. Do your plots in part (a) suggest that the regression relationships in the fitted regressior function in part (a) are innapriopriate for any of the predictor variables? Explain. Yes. Based on the plots it seems as though the zum of residuals for X2 is not zero meaning that it is inappropriate for the function.

9.15. Kidney function. Creatinine clearance (Y) is an important measure of kidney function, but is difficult to obtain in a elinical ofllce setting because it requires 24-hour urine collection. To determine whether this measure can be predicted from some data that are easily available, a kidney specialist obtained the data th1lt fOllow for 33 male subjects. The predictor vari,lbles are serum creatinine concentration (Xt>, age (X2), and weight (X,;).

```
kidney <- loadRData("/Users/lukegeel/Downloads/kidney_spring2021.RData")
```

c. Fit the multiple regression function containing the three predictor variables as first-order terms. Does it appear that all predictor variables should be retained? According to the results, X1 is the best at predicting and needs to be retained. X2 isn't as good as X1 but can still be retained however X3 doesn't do a good job and does not need to be retained.

```
Y <- kidney$Y
X1 <- kidney$X1
X2 <- kidney$X2
X3 <- kidney$X3
m1 <- lm(Y~X1+X2+X3)
anova(m1)
```

```
## Analysis of Variance Table
##
## Response: Y
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## X1         1 19413.7 19413.7 178.240 6.507e-14 ***
## X2         1  2565.1  2565.1  23.550 3.818e-05 ***
## X3         1  3644.8  3644.8  33.463 2.879e-06 ***
## Residuals 29  3158.6   108.9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(m1)
```

```
##
## Call:
## lm(formula = Y ~ X1 + X2 + X3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.823  -7.644   1.214   9.541  15.483
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 111.4487    12.3775   9.004 6.74e-10 ***
## X1          -40.5661     4.6917  -8.646 1.60e-09 ***
## X2           -0.6561     0.1185  -5.538 5.69e-06 ***
## X3            0.8330     0.1440   5.785 2.88e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.44 on 29 degrees of freedom
## Multiple R-squared:  0.8903, Adjusted R-squared:  0.8789
## F-statistic: 78.42 on 3 and 29 DF,  p-value: 5.082e-14
```

```
summary(lm(Y~X1))
```

```
##
## Call:
## lm(formula = Y ~ X1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -37.810 -11.326   2.674  10.965  38.416
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  153.004      9.072  16.865  < 2e-16 ***
## X1           -54.839      6.842  -8.015 4.75e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.38 on 31 degrees of freedom
## Multiple R-squared:  0.6745, Adjusted R-squared:  0.664
## F-statistic: 64.24 on 1 and 31 DF,  p-value: 4.749e-09
```

```
summary(lm(Y~X2))
```

```
##
## Call:
## lm(formula = Y ~ X2)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -36.23 -22.19   4.71  21.50  41.14
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  145.701     13.546  10.756 5.46e-12 ***
## X2            -1.094      0.231  -4.737 4.56e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 23.21 on 31 degrees of freedom
## Multiple R-squared:  0.4199, Adjusted R-squared:  0.4012
## F-statistic: 22.44 on 1 and 31 DF,  p-value: 4.557e-05
```

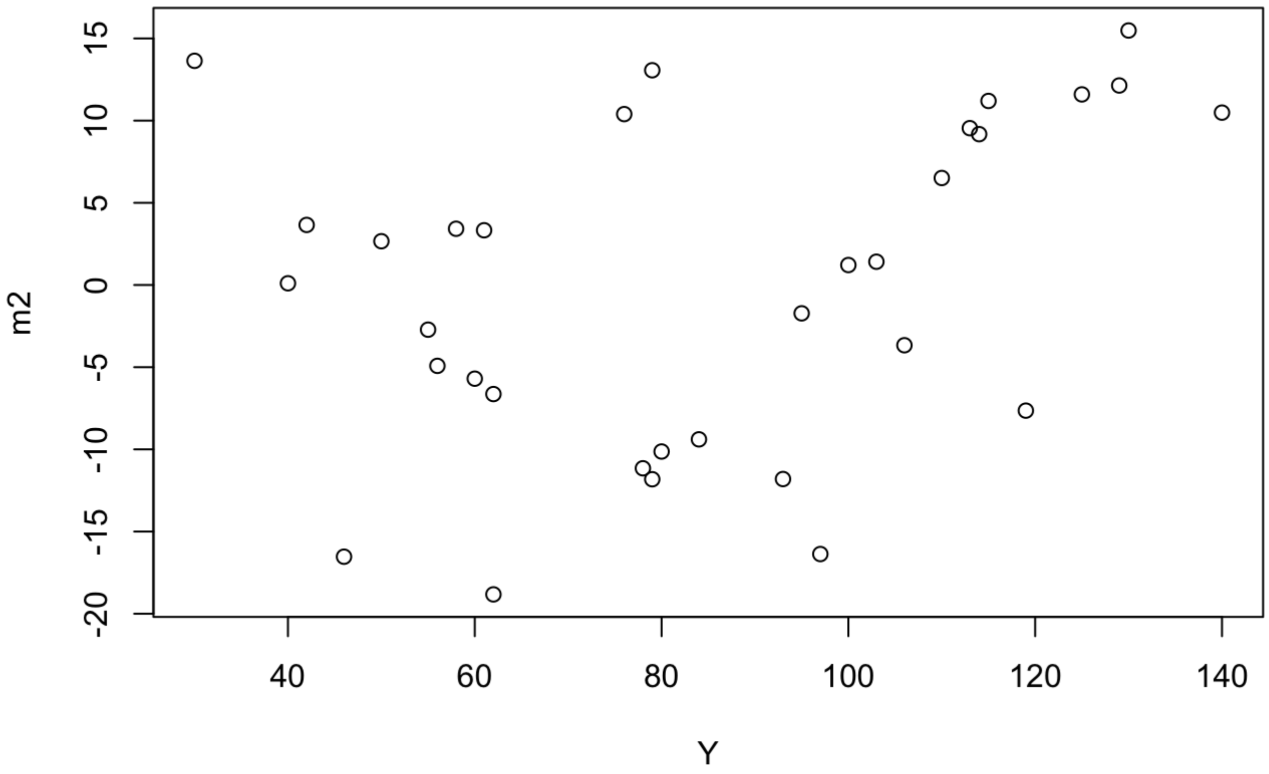
```
summary(lm(Y~X3))
```

```
##
## Call:
## lm(formula = Y ~ X3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -52.172 -23.483  -4.931  17.586  48.862
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  19.4138    28.2134   0.688   0.4965
## X3            0.8966     0.3830   2.341   0.0259 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

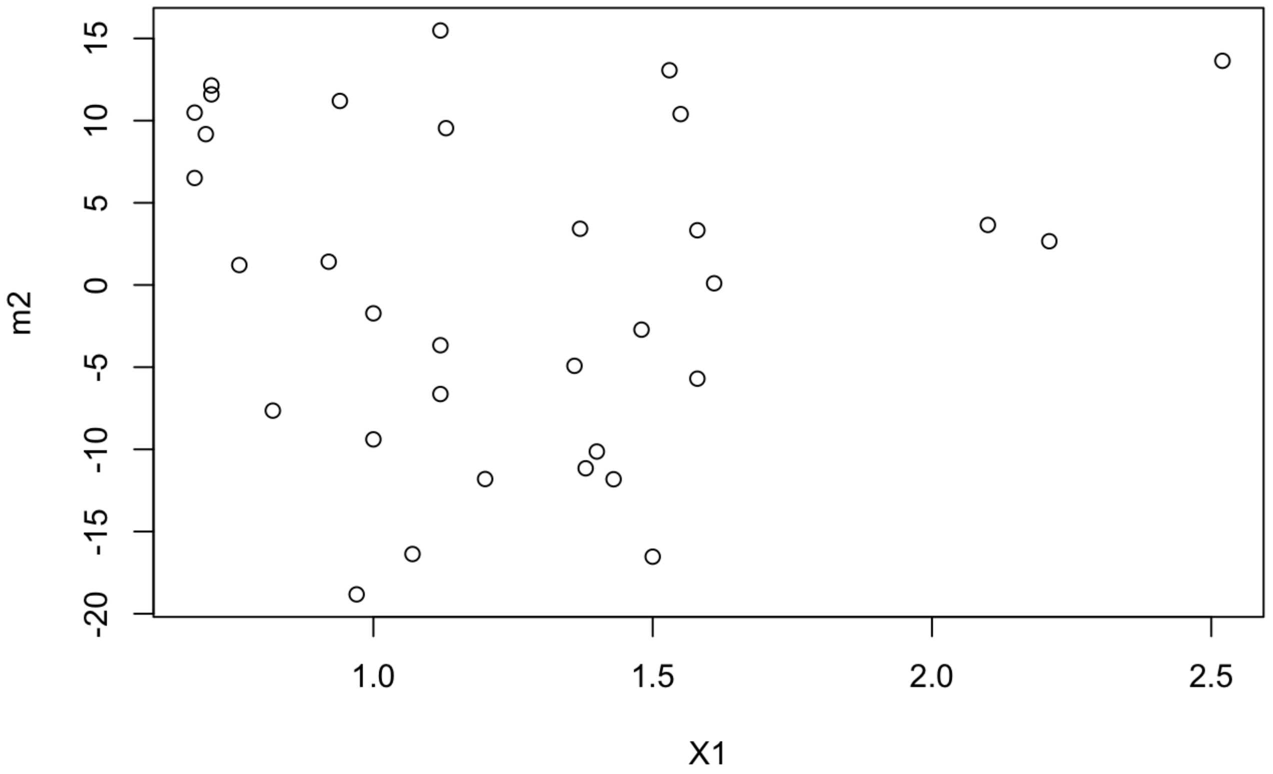
```
##
## Residual standard error: 28.09 on 31 degrees of freedom
## Multiple R-squared:  0.1502, Adjusted R-squared:  0.1228
## F-statistic: 5.479 on 1 and 31 DF,  p-value: 0.02586
```

*10.21. Refer to Kidney function Problem 9.15 and the regression model fitted in part (c). b. Obtain the residuals and plot them seperatley against Y and each of the predictor variables.

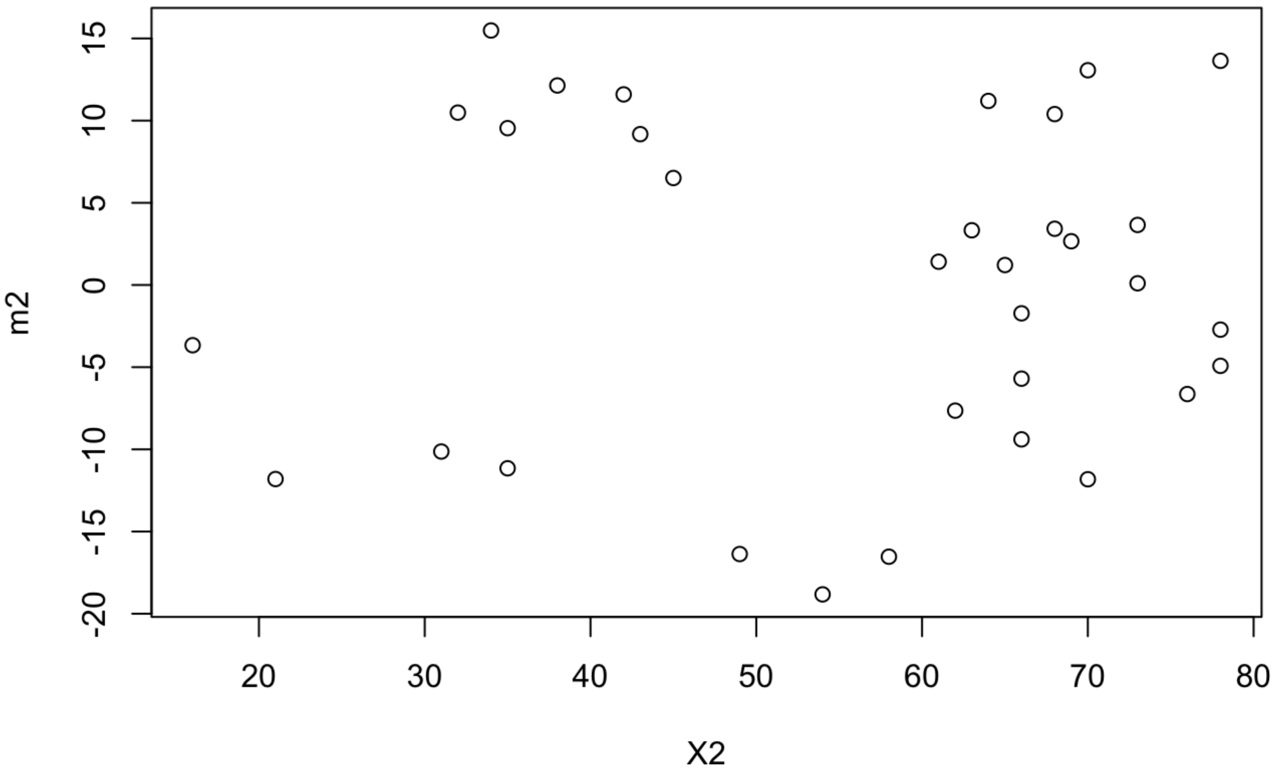
```
kidney <- loadRData("/Users/lukegeel/Downloads/kidney_spring2021.RData")
Y <- kidney$Y
X1 <- kidney$X1
X2 <- kidney$X2
X3 <- kidney$X3
m1 <- lm(Y~X1+X2+X3)
m2 <- resid(m1)
plot(Y,m2)
```



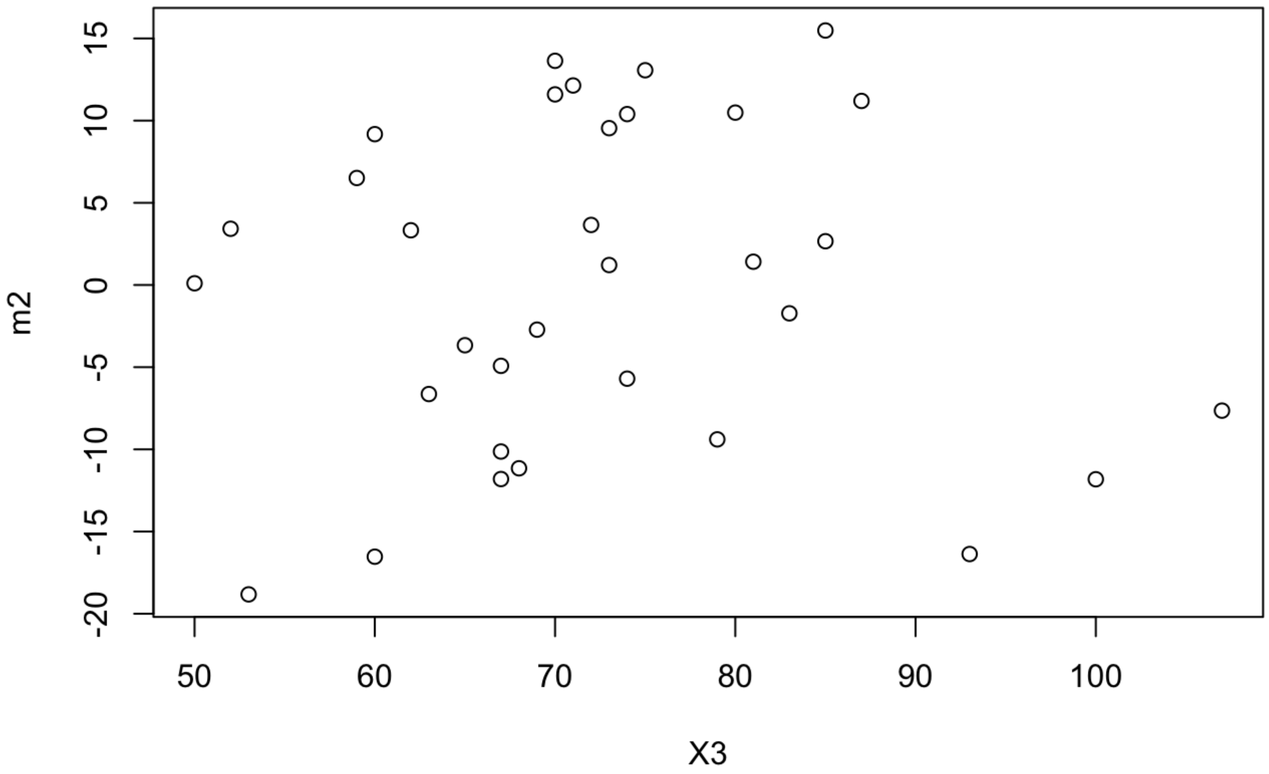
```
plot(X1,m2)
```



```
plot(X2,m2)
```



```
plot(X3,m2)
```



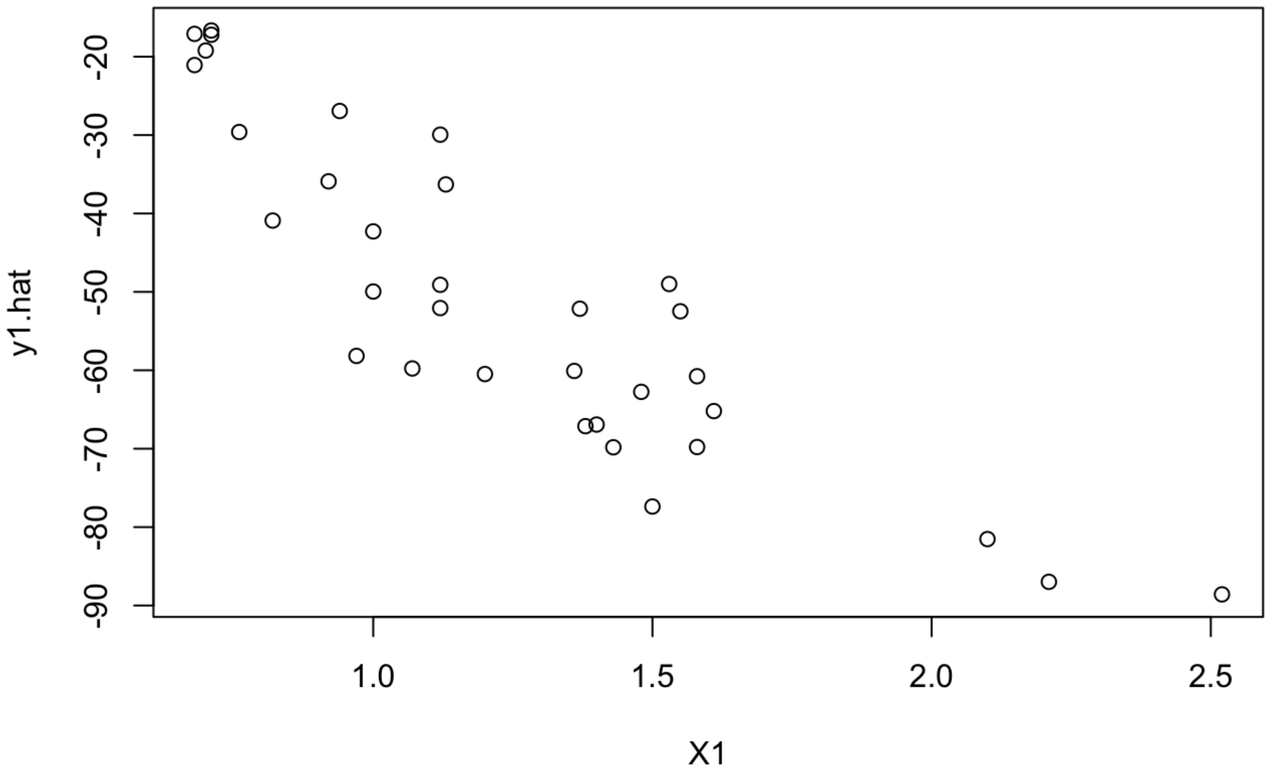
c. Plot Yi − b0 − b2Xi2 − b3Xi3 against Xi1, Yi − b0 − b1Xi1 − b3Xi3 against Xi2, and Yi − b0 − b1Xi1 − b2Xi2 against Xi3 where b0, b1, b2, and b3 are the estimated regression coefficients obtained by fitting a normal errors multiple linear regression model to the response Y with X1, X2, X3 as predictors.

```
lm(Y~X1+X2+X3)
```

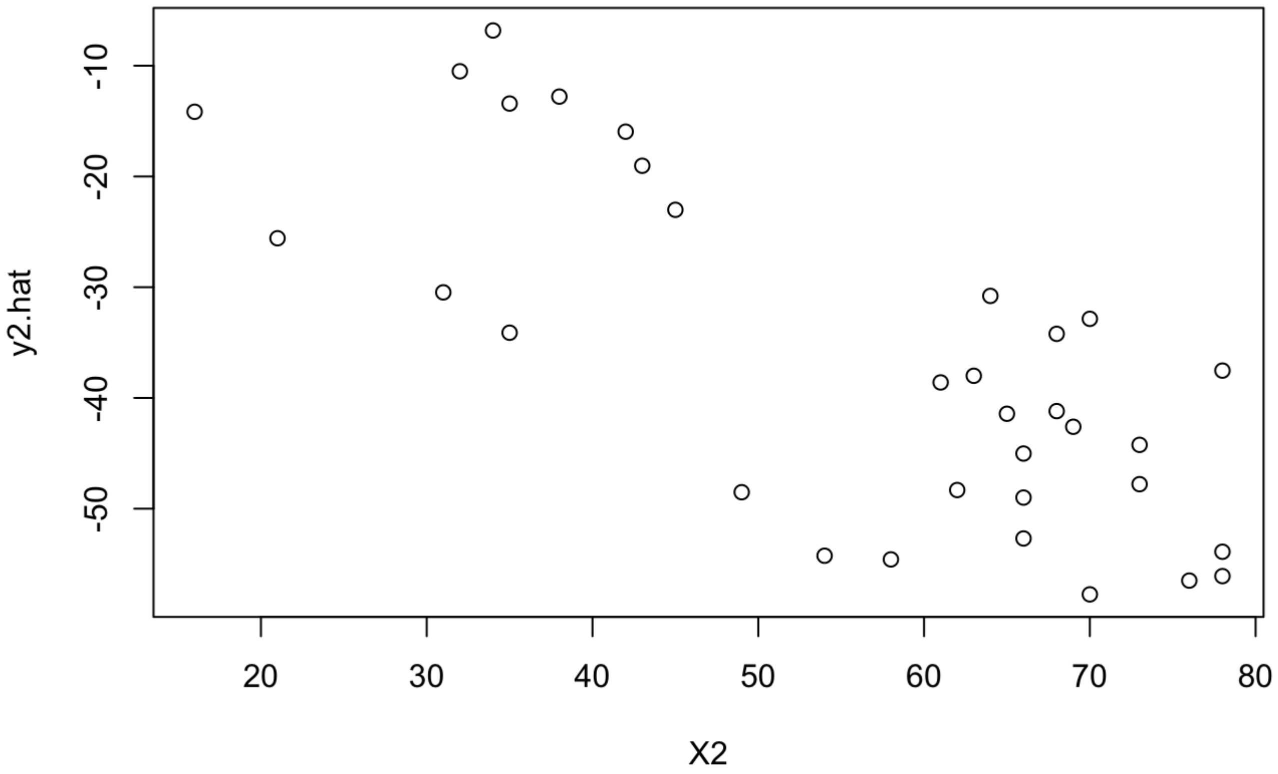
```
##
## Call:
## lm(formula = Y ~ X1 + X2 + X3)
##
## Coefficients:
```

```
## (Intercept)          X1          X2          X3
##    111.4487    -40.5661     -0.6561      0.8330
```

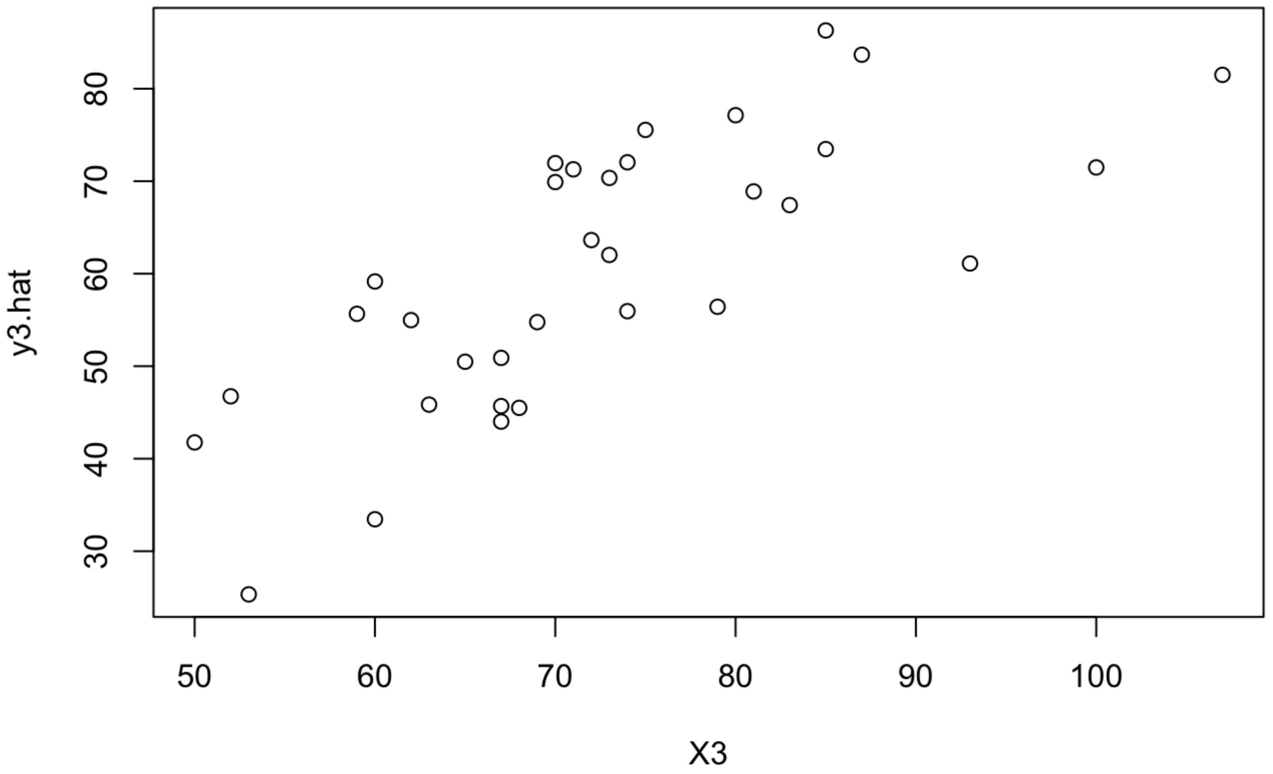
```
b0=111.4487
b1=-40.5661
b2=-0.6561
b3=0.833
y1.hat = Y - b0 - b2*X2-b3*X3
y2.hat = Y - b0 - b1*X1-b3*X3
y3.hat = Y - b0 - b1*X1-b2*X2
plot(X1, y1.hat)
```



```
plot(X2, y2.hat)
```



```
plot(X3,y3.hat)
```



d. Do the plots in parts (b) and (c)

suggest that the regression model should be modified? Based on the plots, I believe that the model would benefit if it's modified. Specifically, X1 seems to have some outliers so if we transformed X1 to say sqrt(X1) the model might be more accurate. Same can be said for the other predictors, just happens that X1 had the most intense outliers.

*10.22. Refer to Kidney function Problems 9.15 and 10.21. Theoretical arguments suggest use of the following regression function: E(lnY) = B0 +B1lnX1+B2ln(140-X2)+B3lnX3 a. Fit the regression function based on theoretical considerations.

```
e=2.71828
Yp=log(Y,e)
X1p=log(X1,e)
```

```
X2p=log(140-X2,e)
X3p=log(X3,e)
lm(Yp~X1p+X2p+X3p)
```

```
##
## Call:
## lm(formula = Yp ~ X1p + X2p + X3p)
##
## Coefficients:
## (Intercept)          X1p          X2p          X3p
##     -2.0250      -0.7092       0.7088       0.7921
```

b. Obtain the residuals and plot them seperately against Y and each predictor variable in the fitted model. Have the difficulties noted in Problem 10.21 now largely been eliminated? I am getting some errors with the transformations of the variables but from what I can tell yes, the difficulties from 10.21 have been largely eliminated.

```
Yp=log(Y,e)
X1p=log(X1,e)
X2p=log(140-X2,e)
X3p=log(X3,e)
m1 <- lm(Yp~X1p+X2p+X3p)
m2 <- resid(m1)
#plot(Yp,m2)
#plot(X1p,m2)
#plot(X2p,m2)
#plot(X3p,m2)
```