

Hw9

Manuel Alejandro Garcia Acosta

11/9/2019

```
library(MASS) # Contains function boxcox
setwd('/home/noble_mannu/Documents/PhD/First/STAT_2131_Applied_Statistical_Methods_I/HW9')
```

Homework 9

Exercise 1

Part (a)

Here we regress gamble onto the predictors sex, status, income and verbal and show the summary of the model.

```
Data.a <- read.table('Gambling.txt', header = TRUE)
linMod.a <- lm(gamble ~ ., data = Data.a)
summary(linMod.a)
```

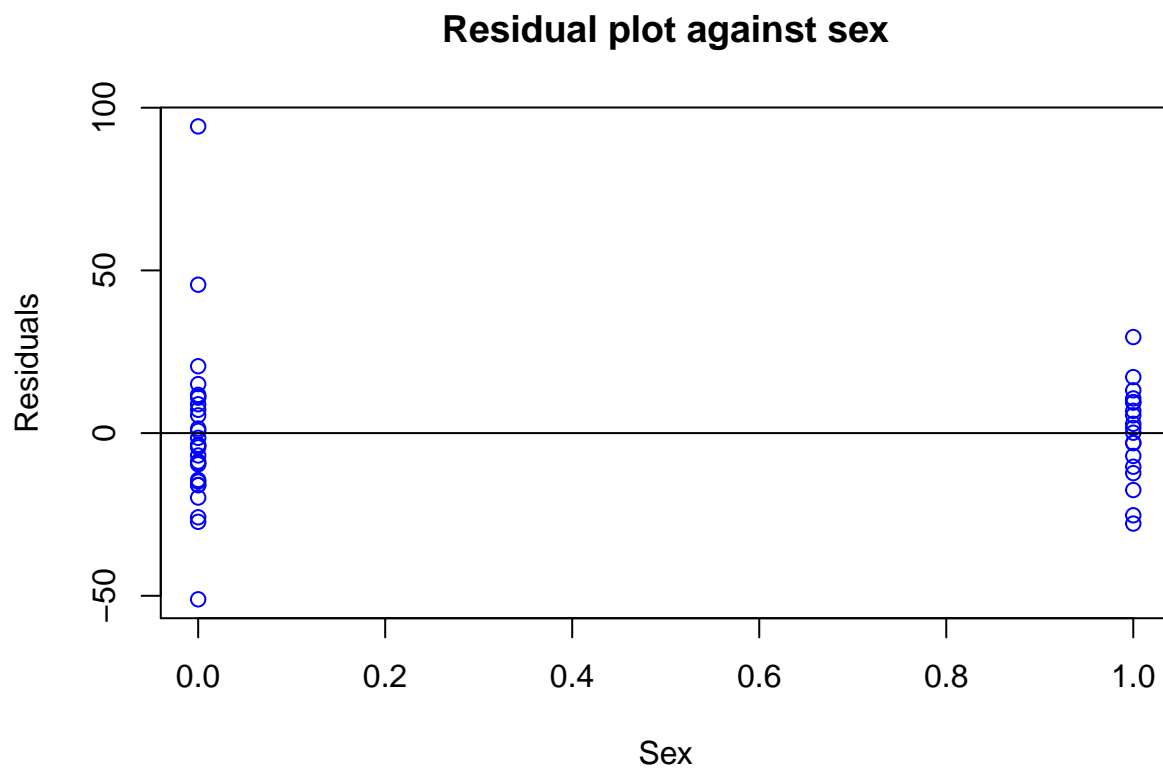
```
##
## Call:
## lm(formula = gamble ~ ., data = Data.a)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -51.082 -11.320  -1.451   9.452  94.252
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  22.55565   17.19680   1.312   0.1968
## sex         -22.11833    8.21111  -2.694   0.0101 *
## status        0.05223    0.28111   0.186   0.8535
## income        4.96198    1.02539   4.839 1.79e-05 ***
## verbal       -2.95949    2.17215  -1.362   0.1803
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 22.69 on 42 degrees of freedom
## Multiple R-squared:  0.5267, Adjusted R-squared:  0.4816
## F-statistic: 11.69 on 4 and 42 DF,  p-value: 1.815e-06
```

Next we plot the residual plots.

Residuals plots

Plot the residuals against sex

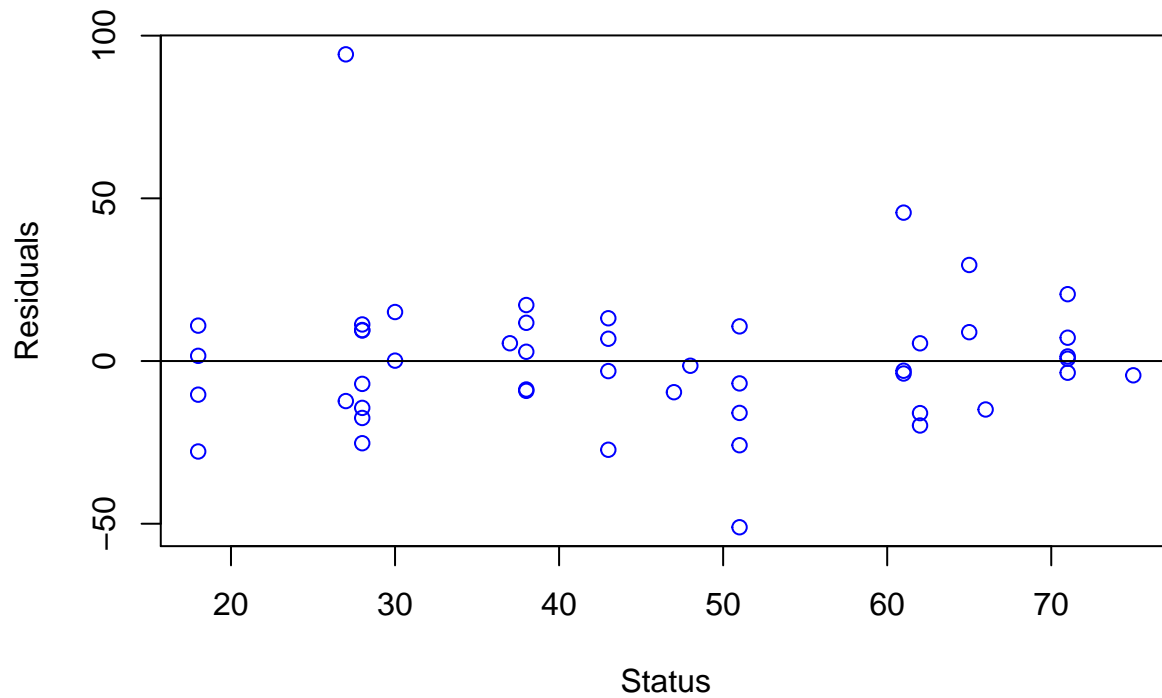
```
plot(Data.a$sex, resid(linMod.a), xlab = "Sex", ylab = "Residuals",
     main = 'Residual plot against sex', col = 'blue')
abline(a=0, b=0)
```



Plot the residuals against status

```
plot(Data.a$status, resid(linMod.a), xlab = "Status", ylab = "Residuals",
     main = 'Residual plot against status', col = 'blue')
abline(a=0, b=0)
```

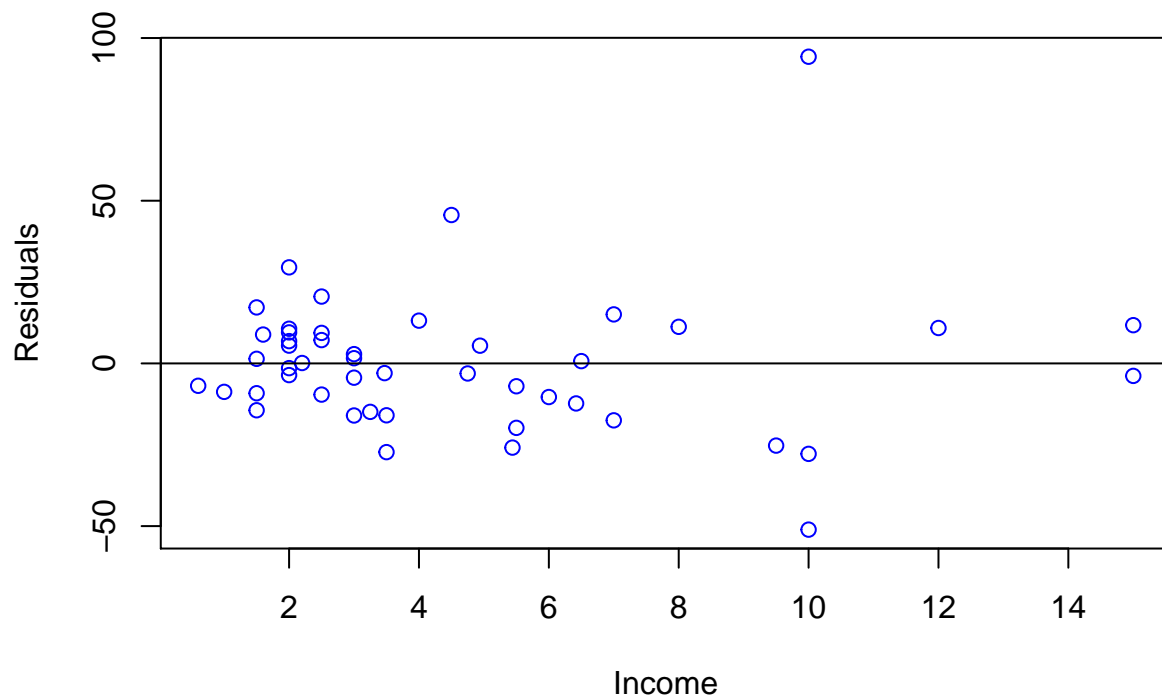
Residual plot against status



Plot the residuals against income

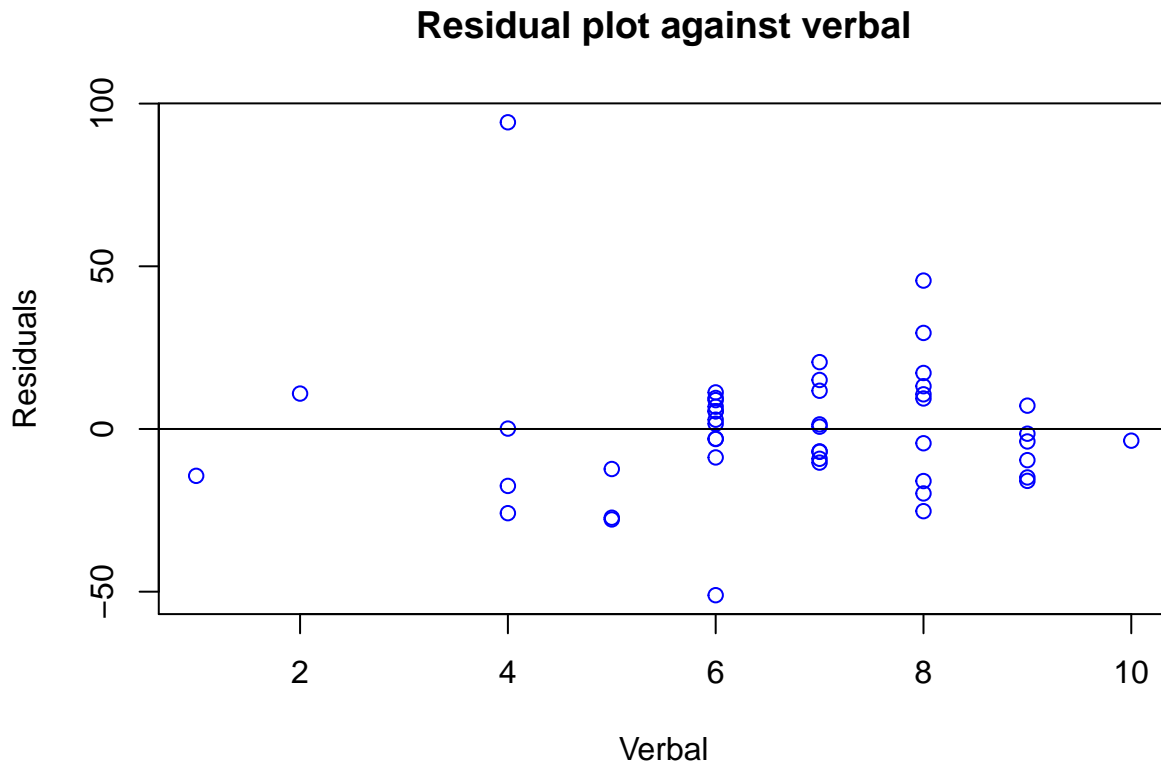
```
plot(Data.a$income, resid(linMod.a), xlab = "Income", ylab = "Residuals",  
     main = 'Residual plot against income', col = 'blue')  
abline(a=0, b=0)
```

Residual plot against income



Plot the residuals against verbal

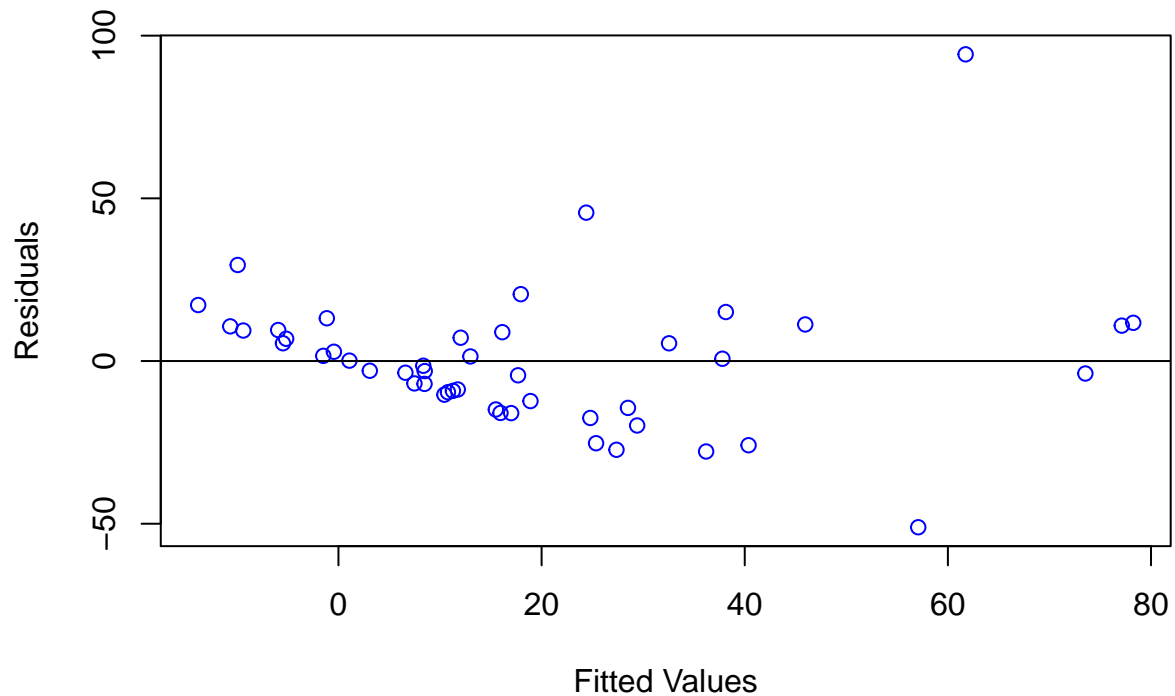
```
plot(Data.a$verbal, resid(linMod.a), xlab = "Verbal", ylab = "Residuals",  
     main = 'Residual plot against verbal', col = 'blue')  
abline(a=0, b=0)
```



Plot the residuals against fitted values

```
plot(linMod.a$fitted.values, resid(linMod.a), xlab = "Fitted Values",  
     ylab = "Residuals", main = 'Residual plot against fitted values',  
     col = 'blue')  
abline(a=0, b=0)
```

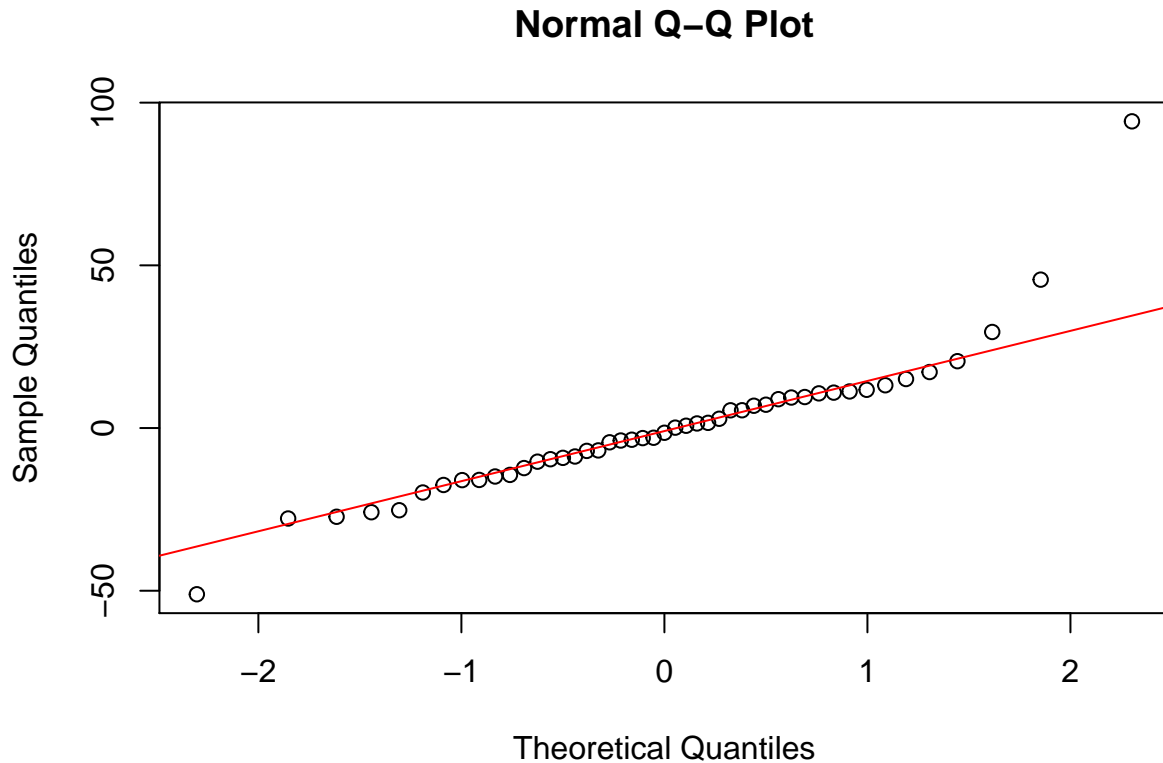
Residual plot against fitted values



By looking at the residual plots with 'income' and 'fitted values' I think that there might be evidence that the constant variance assumption is being violated. With 'income' there seems to be more spread while this variable gets bigger. In addition, I think that the distribution of the points in the plot vs fitted values leads to think that the variance is not constant.

Q-Q Plot

```
qqnorm(residuals(linMod.a))  
qqline(residuals(linMod.a), col="red")
```



In the Q-Q Plot we can see that both tails deviate. I would be a little bit worried that the errors aren't normally distributed because of what we see.

Part (b)

In this part we use Box Cox to suggest a transformation for the response variable. The transformed variable, \tilde{Y} will satisfy the usual mean and variance assumptions.

NOTE: In order to use the `boxcox()` function we add a small value $\delta = 10^{-8}$ to the column *gamble* of our dataset to get such function to work.

```
Data.b <- data.frame(Data.a) # We make a copy of the original data
Data.b$gamble <- Data.b$gamble + 10^-8 # We add 10^-8 to column gamble
linMod.b <- lm(gamble ~ ., data = Data.b) # We run the regression model
summary(linMod.b) # To display the summary of the model
```

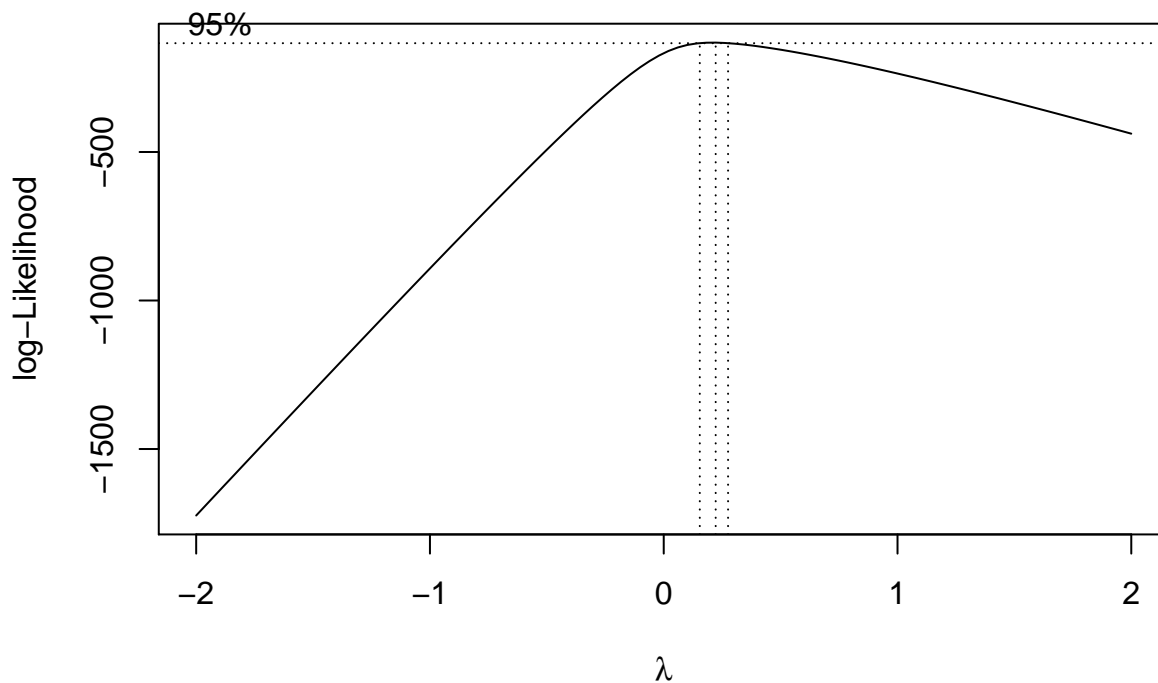
```
##
## Call:
## lm(formula = gamble ~ ., data = Data.b)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -51.082 -11.320  -1.451   9.452  94.252
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  22.55565   17.19680   1.312   0.1968
## sex         -22.11833    8.21111  -2.694   0.0101 *
## status        0.05223    0.28111   0.186   0.8535
## income        4.96198    1.02539   4.839 1.79e-05 ***
```

```
## verbal      -2.95949    2.17215   -1.362    0.1803
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 22.69 on 42 degrees of freedom
## Multiple R-squared:  0.5267, Adjusted R-squared:  0.4816
## F-statistic: 11.69 on 4 and 42 DF,  p-value: 1.815e-06
```

Next we'll run the `boxcox()` function to determine which value for λ is reasonable for the transformation. The transformed data will have the form:

$$\tilde{Y} = \frac{Y^\lambda - 1}{\lambda}$$

```
boxcox.b <- boxcox(linMod.b, plotit = T) # Run boxcox()
```



```
argmax.b <- which.max(boxcox.b$y) # We get the index for argmax(log-likelihood)
lambda.b <- boxcox.b$x[argmax.b] # We get the value for the argmax
```

We get that the best value for λ is:

```
lambda.b
```

```
## [1] 0.2222222
```

Here we obtain \tilde{Y} and we plot it against the residuals of the first model we ran in part(b). In addition, we run another regression model with \tilde{Y} as response.

NOTE: I discussed this problem with the professor and he told me that for simplicity's sake (i.e. being able to explain the transformation if we were required) I should go for $\lambda = 0.25$ instead of the value I got above. So I will use such value rather than $\lambda = 0.2222222$. The reason is that the transformation with the first value will be more easily understood.

$$Y^{0.25} = Y^{\frac{1}{4}} = \sqrt{\sqrt{Y}}$$

```

Y.tilde.b <- (Data.b$gamble^lambda.b - 1)/lambda.b # We transform Y
# Run regression with Y_tilde as response
linMod.b.2 <- lm(Y.tilde.b ~ sex+status+income+verbal, data = Data.b)
summary(linMod.b.2)

##
## Call:
## lm(formula = Y.tilde.b ~ sex + status + income + verbal, data = Data.b)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.727 -1.479  0.069  1.306  4.247
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.96928    1.91690   1.027 0.310147
## sex         -2.12386    0.91528  -2.320 0.025246 *
## status        0.06349    0.03134   2.026 0.049145 *
## income        0.47741    0.11430   4.177 0.000146 ***
## verbal       -0.61955    0.24213  -2.559 0.014197 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.529 on 42 degrees of freedom
## Multiple R-squared:  0.5106, Adjusted R-squared:  0.464
## F-statistic: 10.96 on 4 and 42 DF,  p-value: 3.558e-06

```

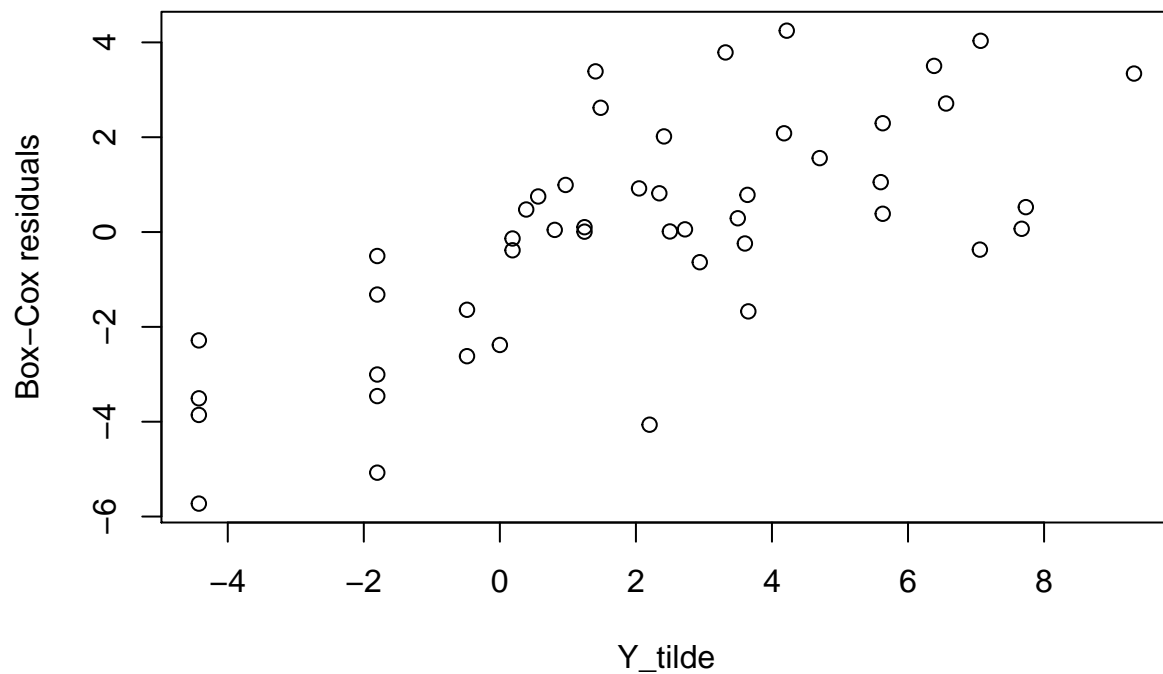
Residual plots under the new model

Here we plot \tilde{Y} vs the estimated residuals.

```

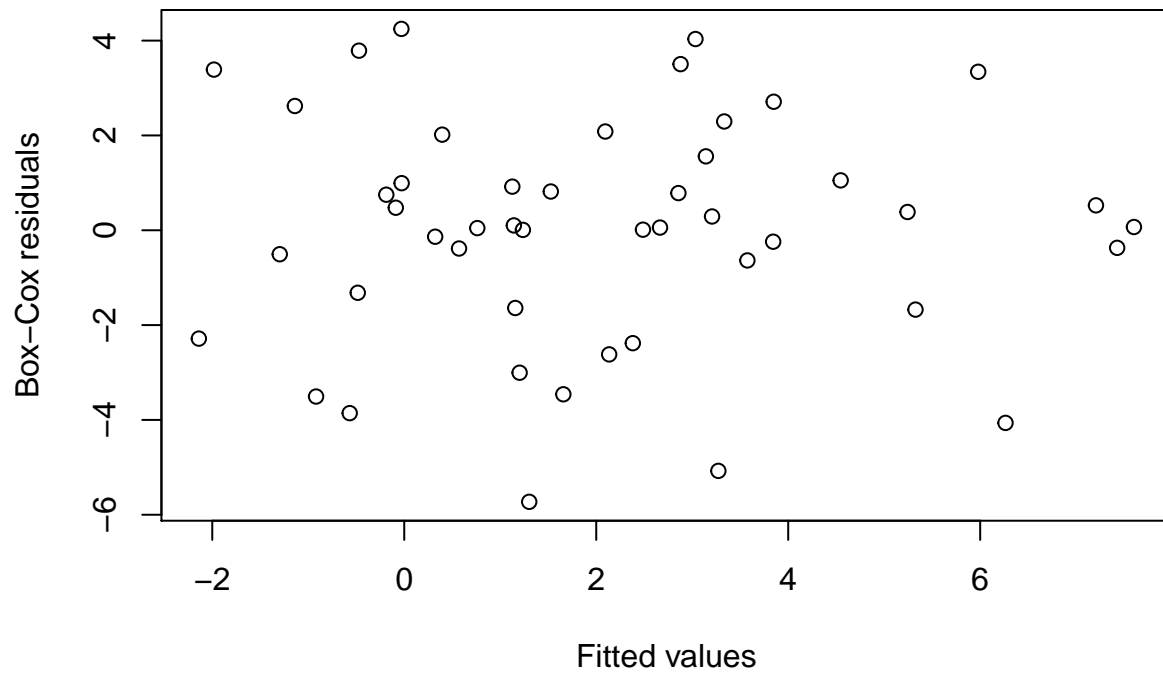
plot(Y.tilde.b, linMod.b.2$residuals, xlab="Y_tilde",
     ylab="Box-Cox residuals")

```

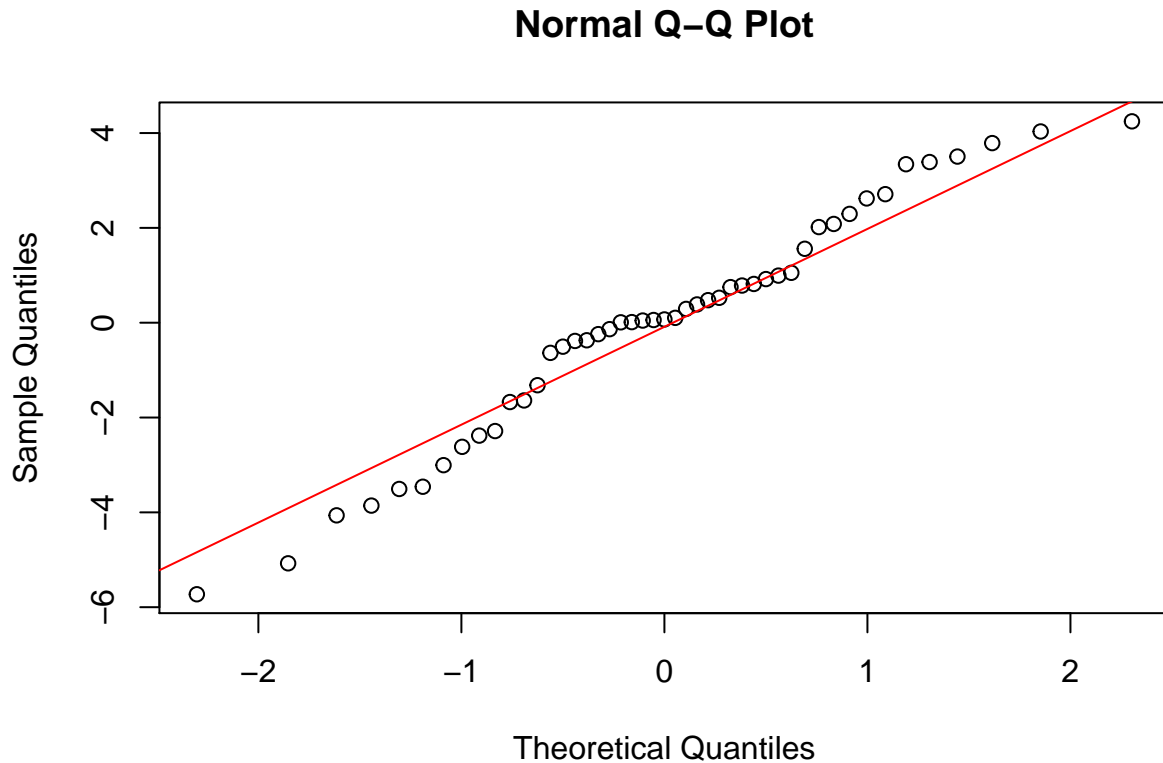
Here we plot the fitted values vs the estimated residuals.

```
plot(linMod.b.2$fitted.values, linMod.b.2$residuals, xlab="Fitted values",
     ylab="Box-Cox residuals")
```



Q-Q Plot under the new model

```
qqnorm(residuals(linMod.b.2))
qqline(residuals(linMod.b.2), col="red")
```



The new model appears to satisfy the constant variance assumption and the residuals also appear to follow a normal distribution.

Part (c)

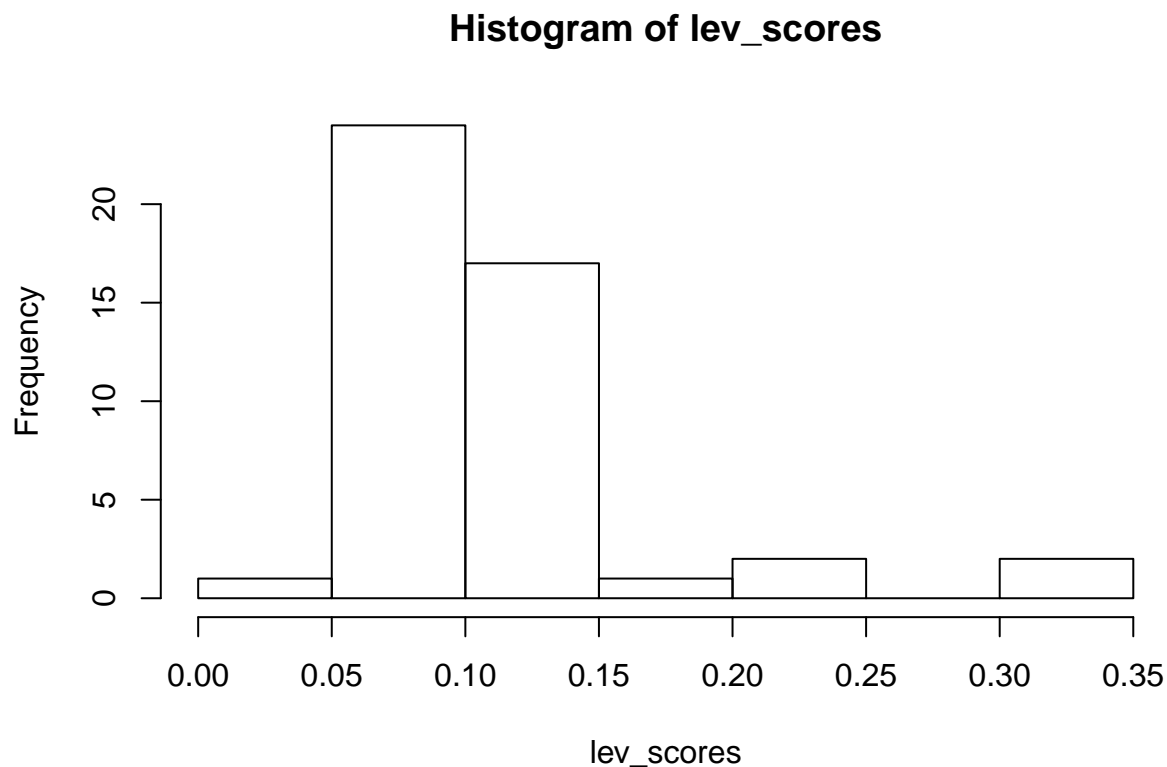
Next we'll compute the hat matrix H for the models we used in part(b).

NOTE: Since we used the same predictors, the hat matrix will be the same for the two models we ran in part(b).

```
intercept <- rep(1, length(Data.b$gamble)) # We create the 1 vector
predictors <- data.matrix(Data.b[,1:4]) # We obtain the columns for the predictors
X <- cbind(intercept, predictors) # We create the X matrix
H <- X %*% solve(t(X) %*% X) %*% t(X) # We create the hat matrix H
```

Remember that the leverage scores are the elements $h_{ii}, i \in \{1, \dots, n\}$ in the diagonal of the hat matrix H . Next we obtain the leverage scores and plot a histogram of them.

```
lev_scores <- diag(H) # We obtain the leverage scores
hist(lev_scores)
```



Part (c.i)

We are concerned about having large leverage scores because they determine influential point in model fitting. In what follows we check if we have large leverage points in this data.

```
# Here we compute the mean leverage h_bar
p <- 5
n <- length(Data.b$gamble)
mean_lev <- p/n
outliers <- which(lev_scores > 2 * mean_lev)
outliers
```

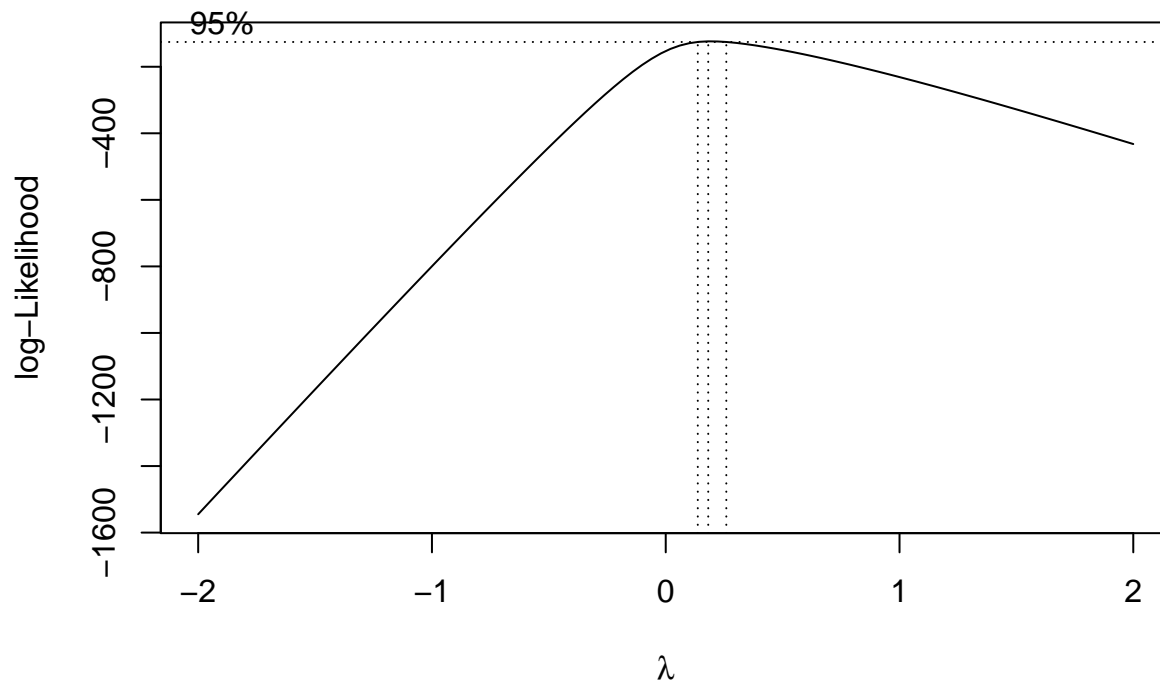
```
## [1] 31 33 35 42
```

It seems that the observations 31,33,35 and 42 are influential points in regression our model.

Part (c.ii)

Here we will reestimate the model from part(b) after removing the points with large leverage scores. I ran again the function boxcox() for the model after deleting the large leverage points to get a new λ for the transformation of Y.

```
# Here we create a dataset dropping the large leverage points
Data.c <- data.frame(Data.b[-c(31,33,35,42),])
# We regress gamble onto the other variables
linMod.c.1 <- lm(gamble ~ . , data = Data.c)
boxcox.c <- boxcox(linMod.c.1, plotit = T) # Run boxcox() again
```



```
argmax.c <- which.max(boxcox.c$y) # We get the index for argmax(log-likelihood)
lambda.c <- boxcox.c$x[argmax.c] # We get the value for the argmax
lambda.c
```

```
## [1] 0.1818182
```

Here we obtained a different λ . We proceed to obtain \tilde{Y} once again and run the model regressing it onto the other four predictors.

```
# We obtain Y_tilde once again
Y.tilde.c <- (Data.c$gamble^lambda.c - 1)/lambda.c
linMod.c.2 <- lm(Y.tilde.c ~ sex+status+income+verbal, data = Data.c)
summary(linMod.c.2)
```

```
##
## Call:
## lm(formula = Y.tilde.c ~ sex + status + income + verbal, data = Data.c)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.2960 -1.5524  0.2807  1.7308  4.1283
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.94838    2.89345   0.673   0.5048
## sex          -2.04640    1.02781  -1.991   0.0537 .
## status         0.06432    0.03410   1.886   0.0669 .
## income         0.44552    0.17956   2.481   0.0176 *
## verbal        -0.64431    0.31508  -2.045   0.0478 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.677 on 38 degrees of freedom
## Multiple R-squared:  0.3909, Adjusted R-squared:  0.3268
```

```
## F-statistic: 6.097 on 4 and 38 DF, p-value: 0.0006834
```

```
summary(linMod.b.2)
```

```
##
## Call:
## lm(formula = Y.tilde.b ~ sex + status + income + verbal, data = Data.b)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.727 -1.479  0.069  1.306  4.247
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.96928    1.91690   1.027 0.310147
## sex         -2.12386    0.91528  -2.320 0.025246 *
## status        0.06349    0.03134   2.026 0.049145 *
## income        0.47741    0.11430   4.177 0.000146 ***
## verbal       -0.61955    0.24213  -2.559 0.014197 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.529 on 42 degrees of freedom
## Multiple R-squared:  0.5106, Adjusted R-squared:  0.464
## F-statistic: 10.96 on 4 and 42 DF, p-value: 3.558e-06
```

Compared to the model in part(b).

Part (d)

```
dis.b <- cooks.distance(linMod.b.2) # Retrieve the cook's distance for all points
cut.b <- qf(.50, df1 = 5, df2 = 47-5)
inf.b <- which(dis.b > cut.b)

dis.c <- cooks.distance(linMod.c.2) # Retrieve the cook's distance for all points
cut.c <- qf(.50, df1 = 5, df2 = 43-5)
inf.c <- which(dis.b > cut.b)
```

None of them seem to be influential points.