MA678 Homework 2

9/20/2022

myName <- "JingjianGao"

11.5

Residuals and predictions: The folder `Pyth` contains outcome y and predictors x_1, x_2 for 40 data points, with a further 20 points with the predictors but no observed outcome. Save the file to your working directory, then read it into R using `read.table()`.

(a)

Use R to fit a linear regression model predicting y from x_1, x_2, using the first 40 data points in the file. Summarize the inferences and check the fit of your model.

```
pyth <- read.table("/Users/billg/Desktop/MA 678 Data/Pyth.csv",header=TRUE)
RegPyth <- glm(y~x1+x2,data=pyth)
summary(RegPyth)
```

```
##
## Call:
## glm(formula = y ~ x1 + x2, data = pyth)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.9585  -0.5865  -0.3356   0.3973   2.8548
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.31513    0.38769   3.392  0.00166 **
## x1           0.51481    0.04590  11.216 1.84e-13 ***
## x2           0.80692    0.02434  33.148  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.8100599)
##
##     Null deviance: 1086.897  on 39  degrees of freedom
## Residual deviance:   29.972  on 37  degrees of freedom
##   (20 observations deleted due to missingness)
## AIC: 109.97
##
## Number of Fisher Scoring iterations: 2
```

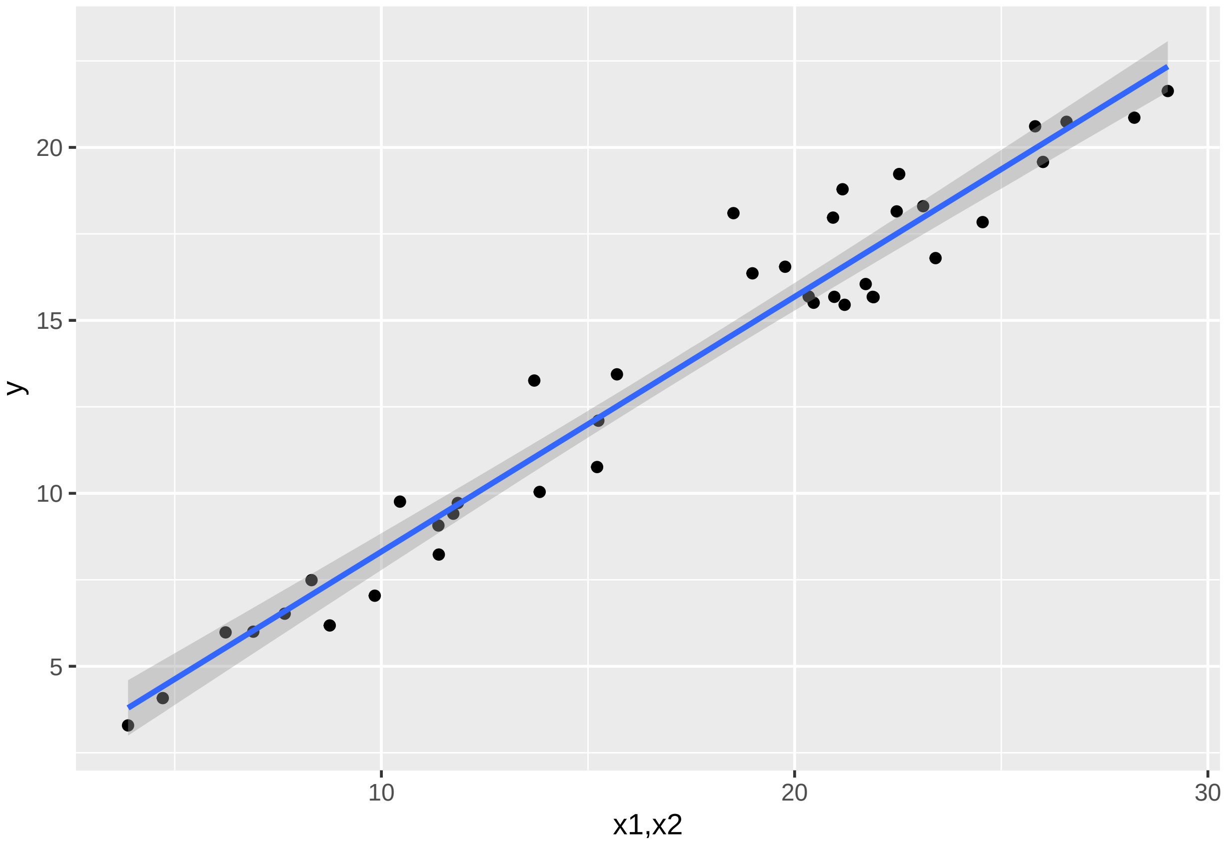
```
#This Model Fits Well.#
```

(b)

Display the estimated model graphically as in Figure 11.2

```r
library(ggplot2)
ggplotpyth <- ggplot(pyth)
ggplotpyth+aes(x=x1+x2,y)+geom_point()+xlab("x1,x2")+ylab("y")+geom_smooth(method="glm",se=T)
```

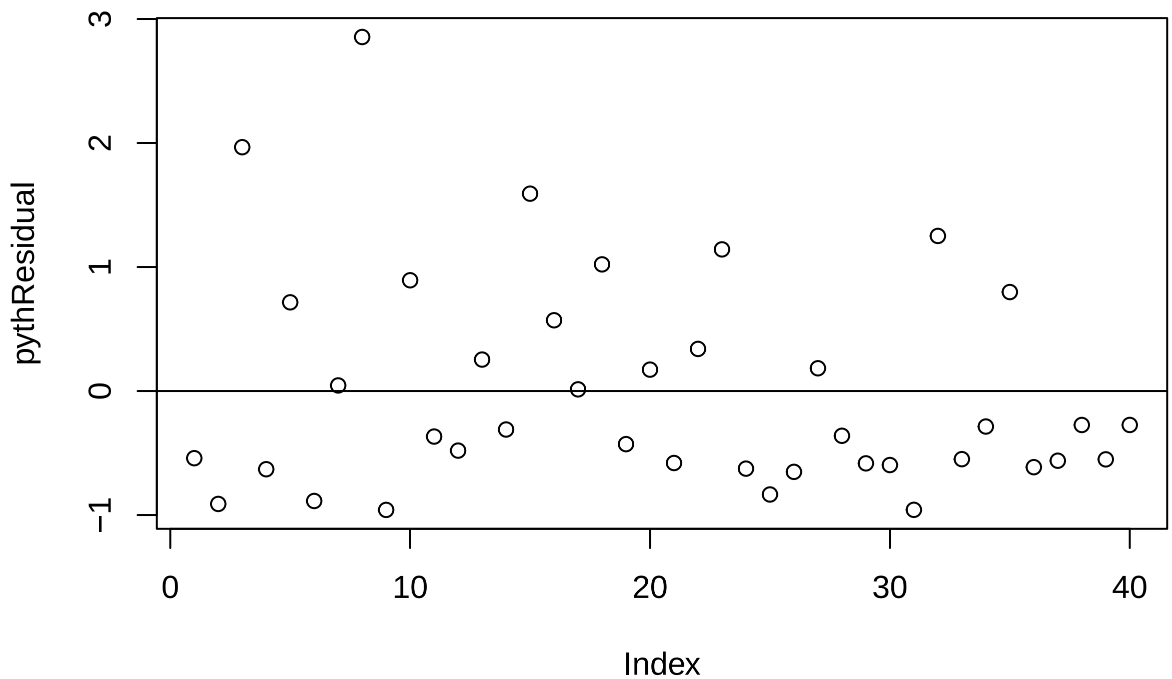
```
## `geom_smooth()` using formula 'y ~ x'
## Warning: Removed 20 rows containing non-finite values (stat_smooth).
## Warning: Removed 20 rows containing missing values (geom_point).
```



(c)

Make a residual plot for this model. Do the assumptions appear to be met?

```r
pythResidual <- resid(RegPyth)
plot(pythResidual)
abline(0,0)
```




```
#"The plot is not distributed normaly, so it does not appear to meet the assumptions"#
#Assumptions:Expectation=0,Variance=sigma^2,Covariance=0#
```

(d)

Make predictions for the remaining 20 data points in the file. How confident do you feel about these predictions?

```
first40 <- head(pyth,40)
remain20 <- tail(pyth,20)
RegPredict <- glm(y~x1+x2,data=pyth)
predict_points <- predict(RegPredict,newdata=remain20)
predict_points
```

```
##        41        42        43        44        45        46        47        48
## 14.812484 19.142865  5.916816 10.530475 19.012485 13.398863  4.829144  9.145767
##        49        50        51        52        53        54        55        56
##  5.892489 12.338639 18.908561 16.064649  8.963122 14.972786  5.859744  7.374900
##        57        58        59        60
##  4.535267 15.133280  9.100899 16.084900
```

12.5

Logarithmic transformation and regression: Consider the following regression:

$$\log(\text{weight}) = -3.8 + 2.1 \log(\text{height}) + \text{error},$$

with errors that have standard deviation 0.25. Weights are in pounds and heights are in inches.

(a)

Fill in the blanks: Approximately 68% of the people will have weights within a factor of ________ and ________ of their predicted values from the regression.

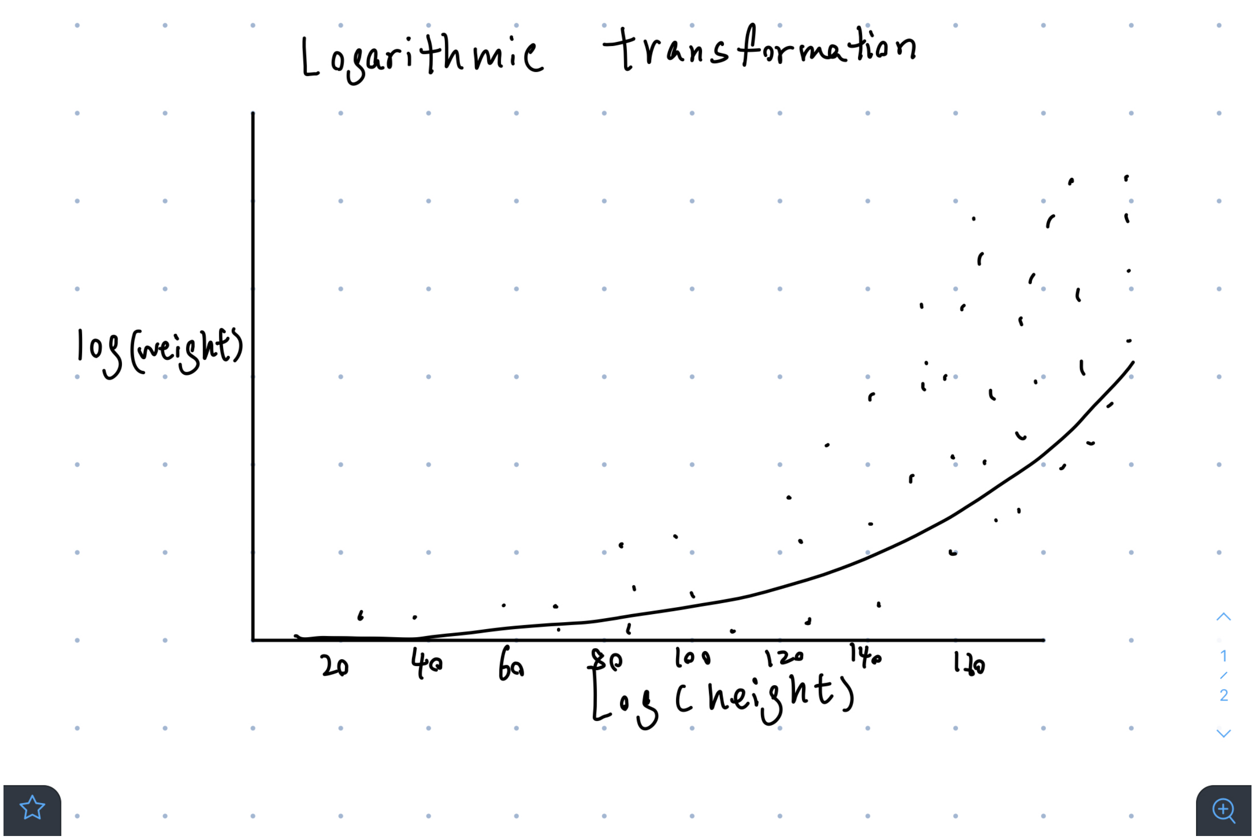
#68 95 99.7 rule tells us that 68% of the population is within one standard deviation of the mean.# #Therefore, Approximately 68% of the people will have weights within a factor of 1.3 and 0.25 of their predicted values from the regression.# #exp(0.25)=1.284

(b)

Using pen and paper, sketch the regression line and scatterplot of log(weight) versus log(height) that make sense and are consistent with the fitted model. Be sure to label the axes of your graph.

#The function is log(weight)=-3.8+2.1*log(height)

```
library(knitr)
knitr::include_graphics("/Users/billg/Desktop/MA 678 Data/Logarithmic Graph.jpeg")
```



12.6

Logarithmic transformations: The folder `Pollution` contains mortality rates and various environmental factors from 60 US metropolitan areas. For this exercise we shall model mortality rate given nitric oxides, sulfur dioxide, and hydrocarbons as inputs. this model is an extreme oversimplication, as it combines all sources of mortality and does not adjust for crucial factors such as age and smoking. We use it to illustrate log transformation in regression.

(a)

Create a scatterplot of mortality rate versus level of nitric oxides. Do you think linear regression will fit these data well? Fit the regression and evaluate a residual plot from the regression.

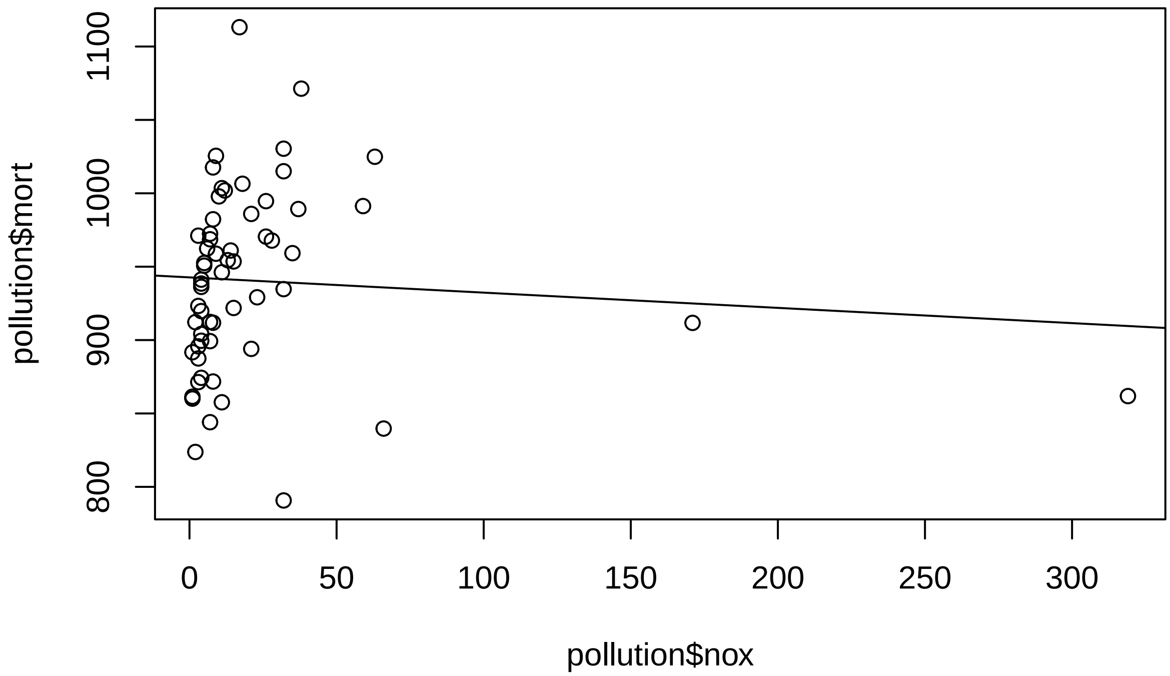
```r
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.2      v forcats 0.5.2
## v purrr   0.3.4
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

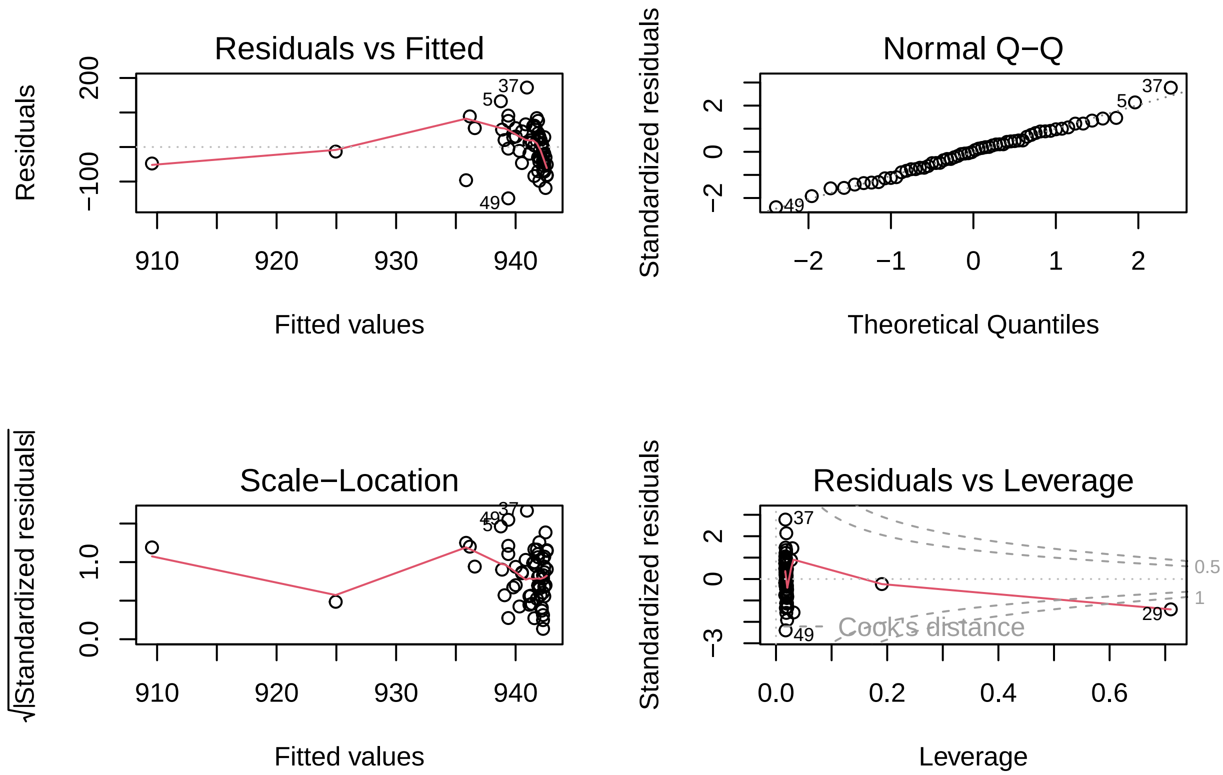
```r
library(foreign)
pollution <- read.dta("http://www.stat.columbia.edu/~gelman/arm/examples/pollution/pollution.dta")
summary(pollution)
```

```
##       prec            jant            jult            ovr65
##  Min.   :10.00   Min.   :12.00   Min.   :63.00   Min.   : 5.600
##  1st Qu.:32.75   1st Qu.:27.00   1st Qu.:72.00   1st Qu.: 7.675
##  Median :38.00   Median :31.50   Median :74.00   Median : 9.000
##  Mean   :37.37   Mean   :33.98   Mean   :74.58   Mean   : 8.798
##  3rd Qu.:43.25   3rd Qu.:40.00   3rd Qu.:77.25   3rd Qu.: 9.700
##  Max.   :60.00   Max.   :67.00   Max.   :85.00   Max.   :11.800
##       popn            educ            hous            dens           nonw
##  Min.   :2.920   Min.   : 9.00   Min.   :66.80   Min.   :1441   Min.   : 0.80
##  1st Qu.:3.210   1st Qu.:10.40   1st Qu.:78.38   1st Qu.:3104   1st Qu.: 4.95
##  Median :3.265   Median :11.05   Median :81.15   Median :3567   Median :10.40
##  Mean   :3.263   Mean   :10.97   Mean   :80.91   Mean   :3876   Mean   :11.87
##  3rd Qu.:3.360   3rd Qu.:11.50   3rd Qu.:83.60   3rd Qu.:4520   3rd Qu.:15.65
##  Max.   :3.530   Max.   :12.30   Max.   :90.70   Max.   :9699   Max.   :38.50
##      wwdrk            poor             hc              nox
##  Min.   :33.80   Min.   : 9.40   Min.   :  1.00   Min.   :  1.00
##  1st Qu.:43.25   1st Qu.:12.00   1st Qu.:  7.00   1st Qu.:  4.00
##  Median :45.50   Median :13.20   Median : 14.50   Median :  9.00
##  Mean   :46.08   Mean   :14.37   Mean   : 37.85   Mean   : 22.65
##  3rd Qu.:49.52   3rd Qu.:15.15   3rd Qu.: 30.25   3rd Qu.: 23.75
##  Max.   :59.70   Max.   :26.40   Max.   :648.00   Max.   :319.00
##       so2             humid            mort
##  Min.   :  1.00   Min.   :38.00   Min.   : 790.7
##  1st Qu.: 11.00   1st Qu.:55.00   1st Qu.: 898.4
##  Median : 30.00   Median :57.00   Median : 943.7
##  Mean   : 53.77   Mean   :57.67   Mean   : 940.4
##  3rd Qu.: 69.00   3rd Qu.:60.00   3rd Qu.: 983.2
##  Max.   :278.00   Max.   :73.00   Max.   :1113.2
```

```r
plot(pollution$nox,pollution$mort)
Regpollution <- lm(mort~nox,data=pollution)
abline(Regpollution)
```




```r
par(mfrow=c(2,2))
plot(Regpollution)
```




```r
"I think linear regression will not fit these data well. The residual plot is not random."
```

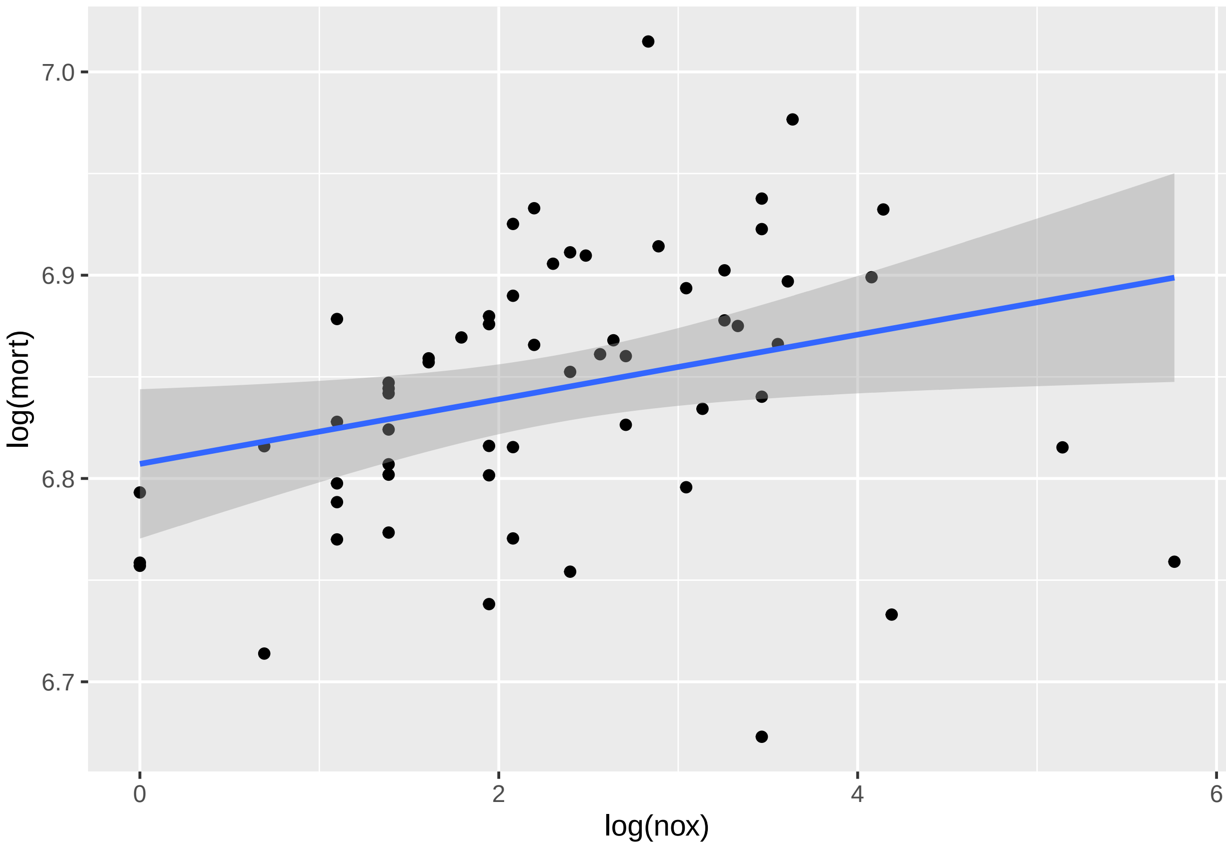
```
## [1] "I think linear regression will not fit these data well. The residual plot is not random."
```

(b)

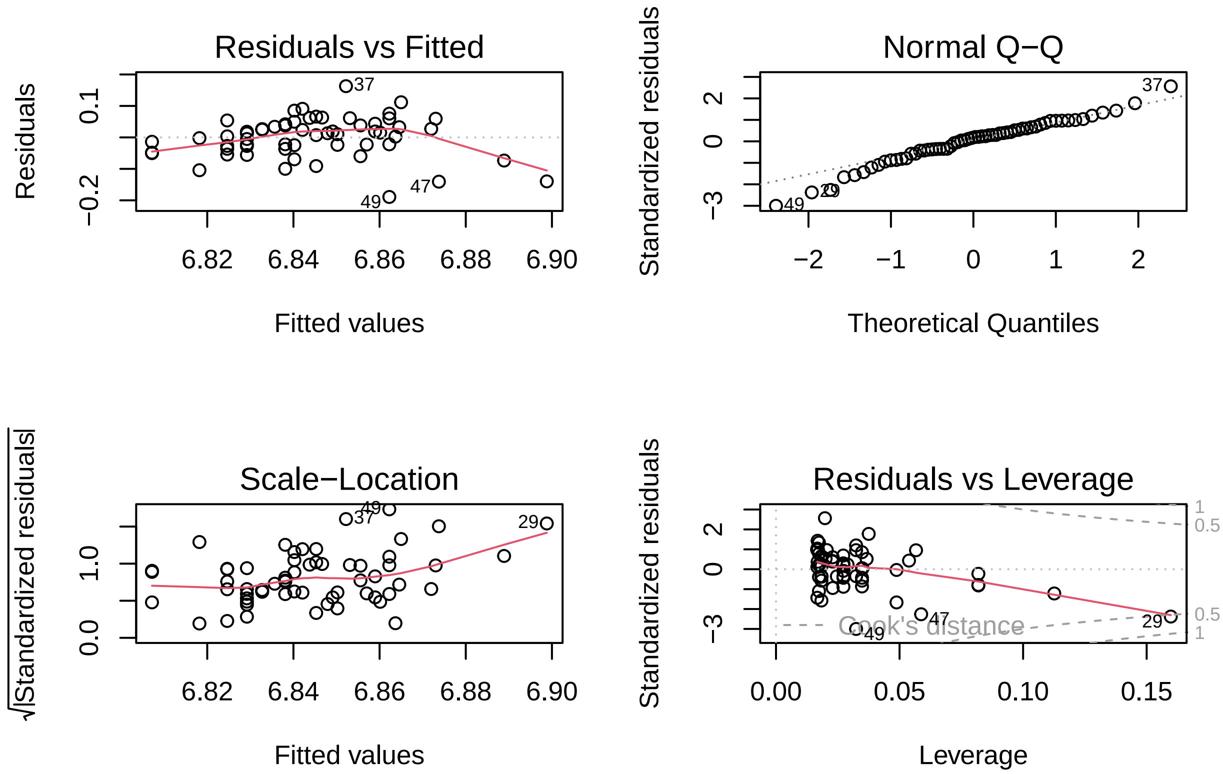
Find an appropriate reansformation that will result in data more appropriate for linear regression. Fit a regression to the transformed data and evaluate the new residual plot.

```
Regpollution2 <- lm(log(pollution$mort)~log(pollution$nox),data=pollution)

ggplot(data=pollution, aes(x=log(nox), y=log(mort))) + geom_point() +
  geom_smooth(method="lm", formula=y ~ x)
```




```
par(mfrow=c(2,2))
plot(Regpollution2)
```




```r
"The new Residual Plot is so much better since the points are spreaded out."
```

```
## [1] "The new Residual Plot is so much better since the points are spreaded out."
```

```r
exp(6.81)
```

```
## [1] 906.8708
```

(c)

Interpret the slope coefficient from the model you chose in (b)

#The average morality rate is 906.81 #For each 1% of change in nox, the morality rate changes 2%

(d)

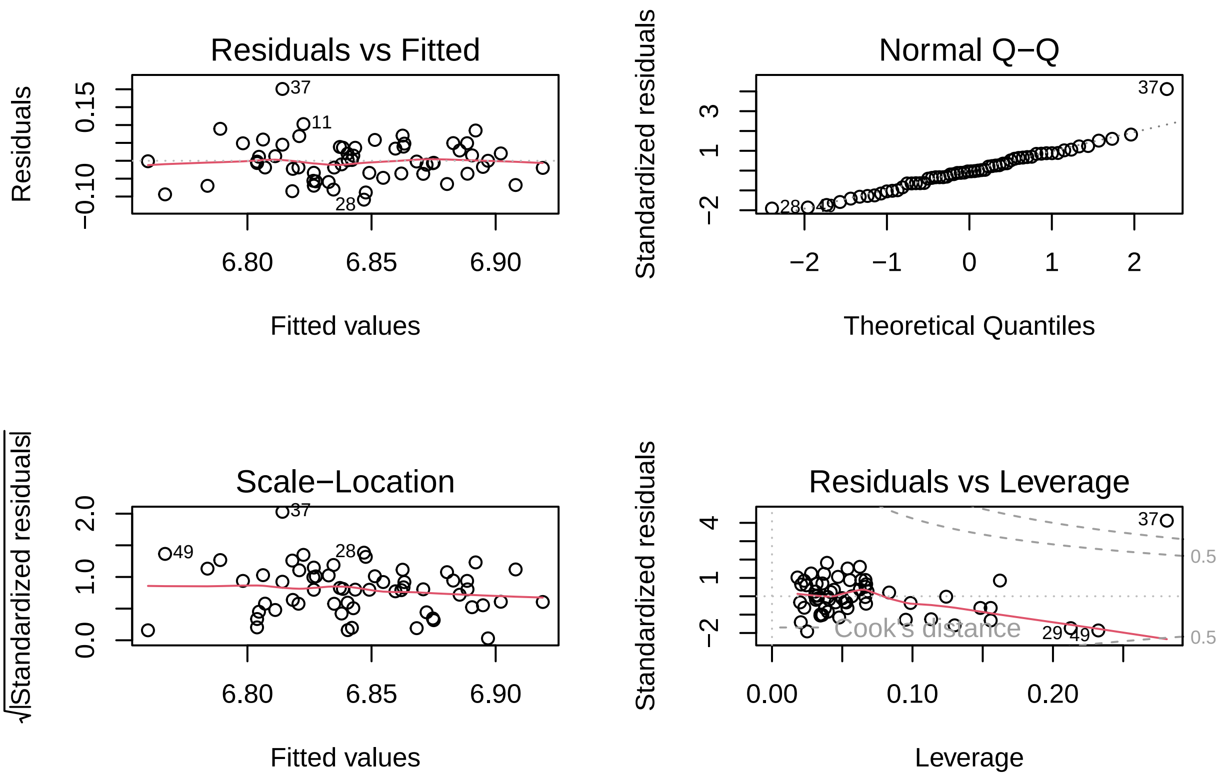
Now fit a model predicting mortality rate using levels of nitric oxides, sulfur dioxide, and hydrocarbons as inputs. Use appropriate transformation when helpful. Plot the fitted regression model and interpret the coefficients.

```r
Regpollution3 <- lm(log(pollution$mort)~log(pollution$nox)+
                      log(pollution$hc)+log(pollution$so2),data=pollution)
summary(Regpollution3)
```

```
##
## Call:
## lm(formula = log(pollution$mort) ~ log(pollution$nox) + log(pollution$hc) +
##     log(pollution$so2), data = pollution)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.10874 -0.03574 -0.00218  0.03709  0.20085
```

```
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          6.826749   0.022701 300.726  < 2e-16 ***
## log(pollution$nox)   0.059837   0.023021   2.599  0.01192 *
## log(pollution$hc)   -0.060812   0.020553  -2.959  0.00452 **
## log(pollution$so2)   0.014309   0.007584   1.887  0.06436 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05753 on 56 degrees of freedom
## Multiple R-squared:  0.2852, Adjusted R-squared:  0.2469
## F-statistic: 7.449 on 3 and 56 DF,  p-value: 0.0002777
```

```
par(mfrow=c(2,2))
plot(Regpollution3)
```




```
""
```

```
## [1] ""
```

(e)

Cross validate: fit the model you chose above to the first half of the data and then predict for the second half. You used all the data to construct the model in (d), so this is not really cross validation, but it gives a sense of how the steps of cross validation can be implemented.

```
firsthalf <- head(pollution,30)
secondhalf <- tail(pollution,30)
Regpollution4 <- lm(log(pollution$mort)~log(pollution$nox)+
                      log(pollution$hc)+log(pollution$so2),pollution)
```

```r
predict_pollution <- predict(Regpollution4,newdata=secondhalf)
```

```
## Warning: 'newdata' had 30 rows but variables found have 60 rows
```

```r
predict_pollution
```

```
##        1        2        3        4        5        6        7        8
## 6.861994 6.890497 6.875035 6.820883 6.891924 6.888491 6.908041 6.820577
##        9       10       11       12       13       14       15       16
## 6.851339 6.834941 6.822568 6.882914 6.894884 6.859607 6.806299 6.826749
##       17       18       19       20       21       22       23       24
## 6.840414 6.826565 6.868182 6.798192 6.826749 6.827562 6.789061 6.807100
##       25       26       27       28       29       30       31       32
## 6.814092 6.837239 6.811203 6.846958 6.847682 6.896912 6.885478 6.759940
##       33       34       35       36       37       38       39       40
## 6.870828 6.834676 6.874710 6.837939 6.814103 6.862872 6.902068 6.918980
##       41       42       43       44       45       46       47       48
## 6.818284 6.842099 6.888640 6.862506 6.832725 6.840341 6.818088 6.849101
##       49       50       51       52       53       54       55       56
## 6.766832 6.803877 6.854694 6.842658 6.843425 6.804664 6.838466 6.783921
##       57       58       59       60
## 6.863229 6.803907 6.880411 6.872165
```

12.7

Cross validation comparison of models with different transformations of outcomes: when we compare models with transformed continuous outcomes, we must take into account how the nonlinear transformation warps the continuous outcomes. Follow the procedure used to compare models for the mesquite bushes example on page 202.

(a)

Compare models for earnings and for log(earnings) given height and sex as shown in page 84 and 192. Use `earnk` and `log(earnk)` as outcomes.

```r
library(rstanarm)
```

```
## Loading required package: Rcpp
## This is rstanarm version 2.21.3
## - See https://mc-stan.org/rstanarm/articles/priors for changes to default priors!
## - Default priors may change, so it's safest to specify priors, even if equivalent to the defaults.
## - For execution on a local, multicore CPU with excess RAM we recommend calling
##   options(mc.cores = parallel::detectCores())
```

```r
earnings <- read.csv("/Users/billg/Desktop/MA 678 Data/earnings.csv")
#Regearnings <-stan_glm(earn~height+male,data=earnings)
#loo_1 <- loo(Regearnings)
#earnk <- kfold(Regearnings,K=10)
#earnk
#Regearnings2 <-stan_glm(log(earn)~log(height)+log(male),data=earnings)
#loo_2 <- loo(Regearnings2)
#log(earnk) <- kfold(Regearnings2,K=10)
```

(b)

Compare models from other exercises in this chapter.

#The models are similar. Some are just simpler. Logarithmic transformations are great"

12.8

Log-log transformations: Suppose that, for a certain population of animals, we can predict log weight from log height as follows:

- An animal that is 50 centimeters tall is predicted to weigh 10 kg.
- Every increase of 1% in height corresponds to a predicted increase of 2% in weight.
- The weights of approximately 95% of the animals fall within a factor of 1.1 of predicted values.

(a)

Give the equation of the regression line and the residual standard deviation of the regression.

#The equation should be: log(weight)=2*log(height)+log(10)-2*long(50)+error -> log(weight)=-5.5+2*log(height)+error #Since 95% of the animals fall within a factor of 1.1 of predicted values,error is between -0.095 and 0.095.Then the residual standard deviation would be 0.0486.

(b)

Suppose the standard deviation of log weights is 20% in this population. What, then, is the R^2 of the regression model described here?

#R^2=1-(0.0486/0.2)=0.757

12.9

Linear and logarithmic transformations: For a study of congressional elections, you would like a measure of the relative amount of money raised by each of the two major-party candidates in each district. Suppose that you know the amount of money raised by each candidate; label these dollar values D_i and R_i. You would like to combine these into a single variable that can be included as an input variable into a model predicting vote share for the Democrats. Discuss the advantages and disadvantages of the following measures:

(a)

The simple difference, $D_i - R_i$

#The advantage of this measure is that the difference is easy to get and it's centered at zero. But this measure will not mean the same when D_i and R_i become larger.

(b)

The ratio, D_i/R_i

#The ratio is not recommended because if the republics party raise way more money than Democrats, the measure will approach to zero. Answers will be various.

(c)

The difference on the logarithmic scale, $\log D_i - \log R_i$

#This measure is similar to part (a), better than (a), since there is a less severe increase or decrease, and is not much affected by the outliers.

(d)

The relative proportion, $D_i/(D_i + R_i)$.

#This measure is better than part (b). The relative proportion is not much affected if the money raised by republics party is very large.

12.11

Elasticity: An economist runs a regression examining the relations between the average price of cigarettes, P, and the quantity purchased, Q, across a large sample of counties in the United States, assuming the functional form, $\log Q = \alpha + \beta \log P$. Suppose the estimate for β is 0.3. Interpret this coefficient.

#With the logarithmic scale, for every 1% change in the average price of cigarettes, there is a 0.3% change is the total cigarette quality purchased.

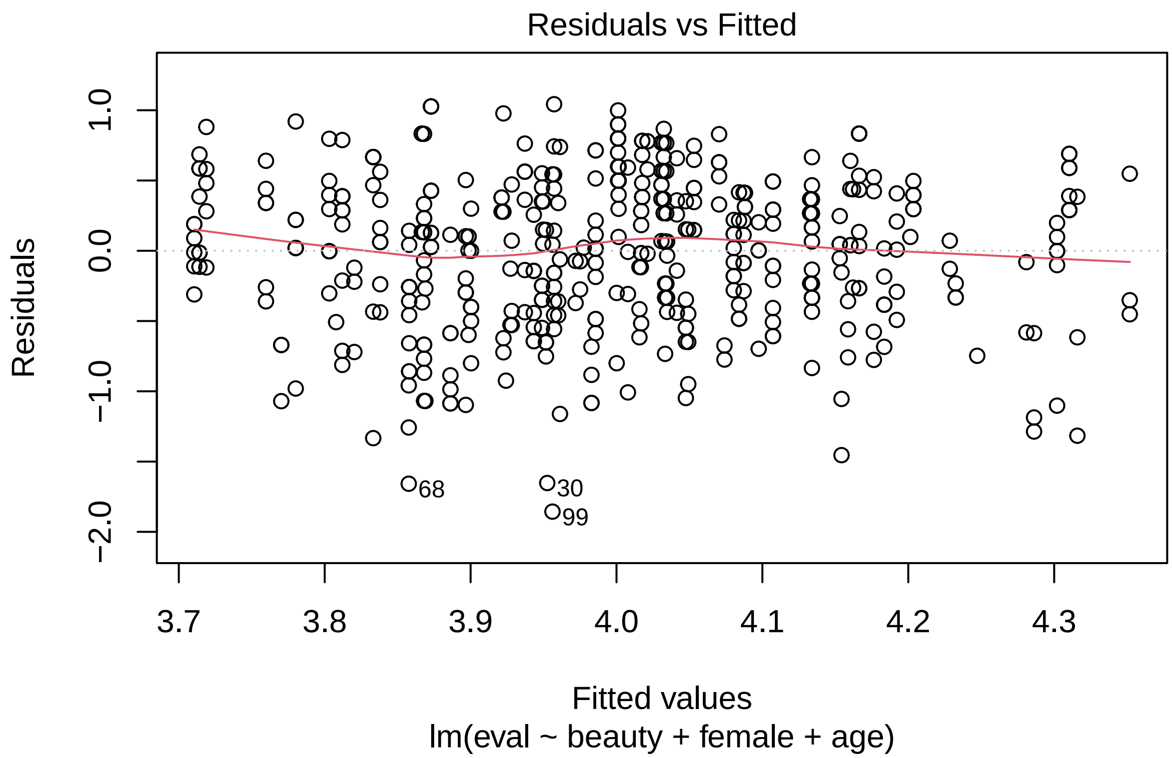
12.13

Building regression models: Return to the teaching evaluations data from Exercise 10.6. Fit regression models predicting evaluations given many of the inputs in the dataset. Consider interactions, combinations of predictors, and transformations, as appropriate. Consider several models, discuss in detail the final model that you choose, and also explain why you chose it rather than the others you had considered.

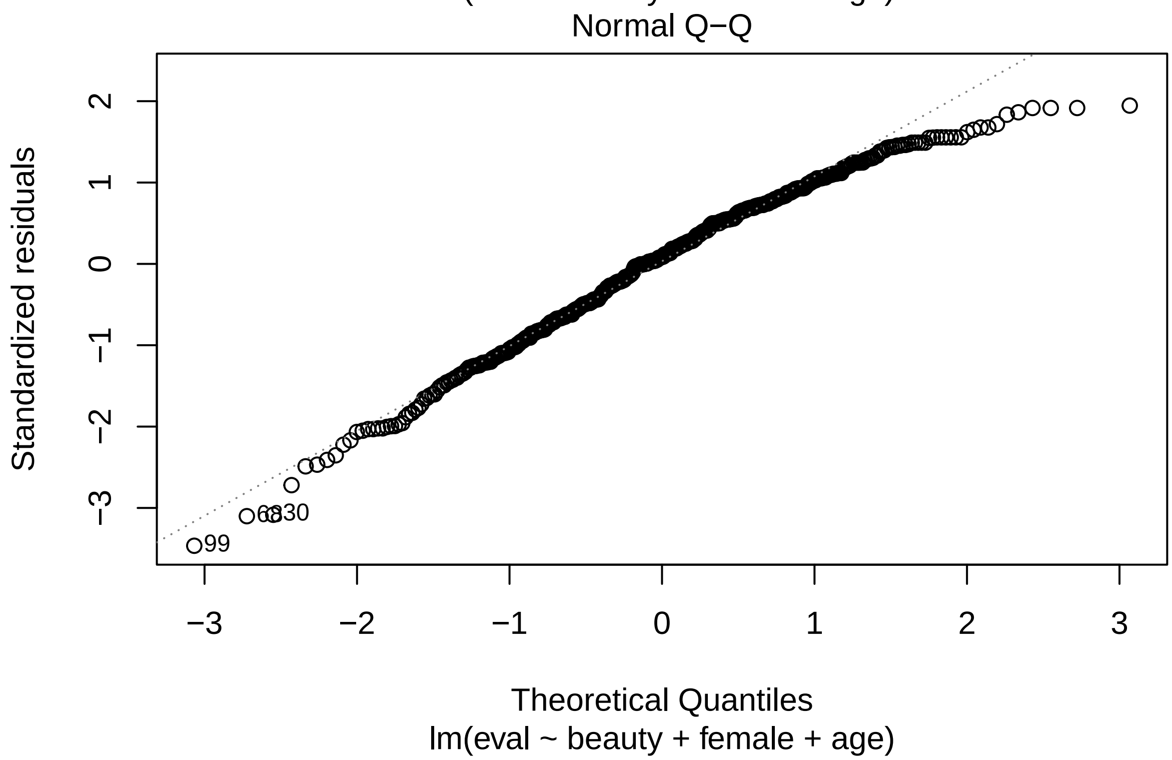
```
beauty <- read.csv("/Users/billg/Desktop/MA 678 Data/beauty.txt")
head(beauty)
```

```
##   eval     beauty female age minority nonenglish lower course_id.
## 1  4.3  0.2015666      1  36        1          0     0        3\\
## 2  4.5 -0.8260813      0  59        0          0     0        0\\
## 3  3.7 -0.6603327      0  51        0          0     0        4\\
## 4  4.3 -0.7663125      1  40        0          0     0        2\\
## 5  4.4  1.4214450      1  31        0          0     0        0\\
## 6  4.2  0.5002196      0  62        0          0     0        0\\
```

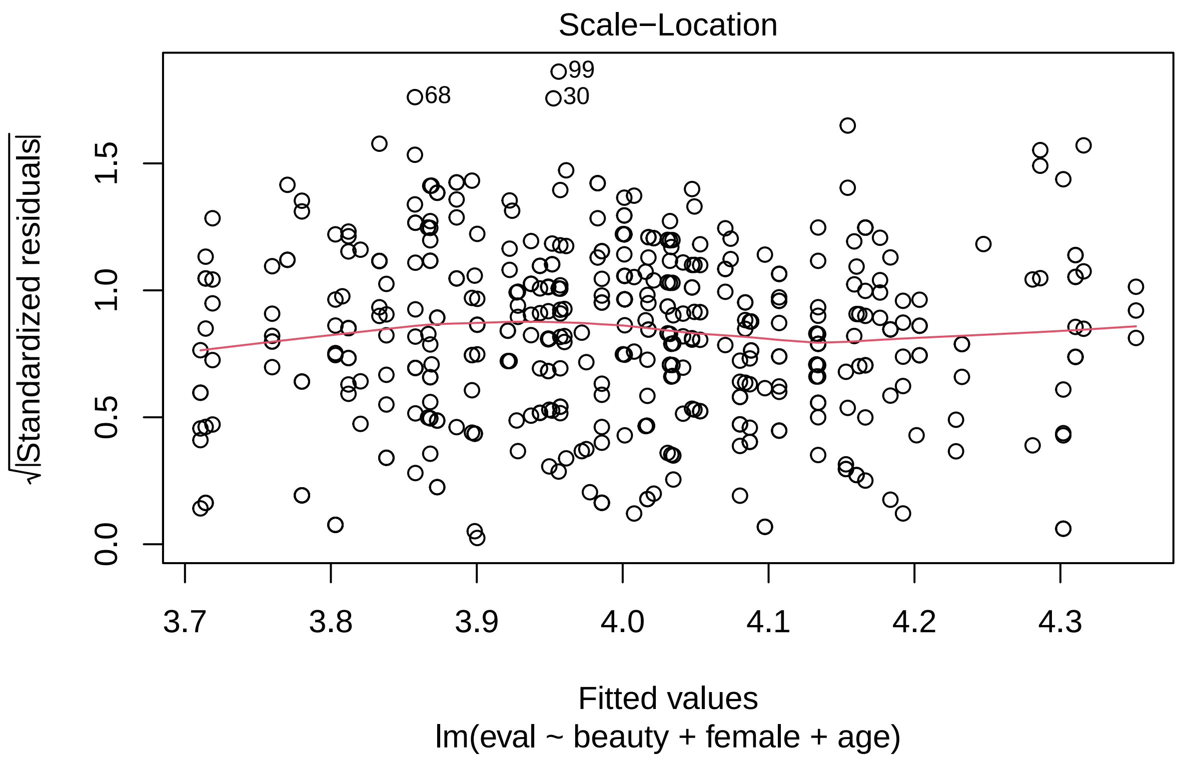
```
reg10.6a <- lm(eval~beauty+female+age,data=beauty)
plot(reg10.6a)
```

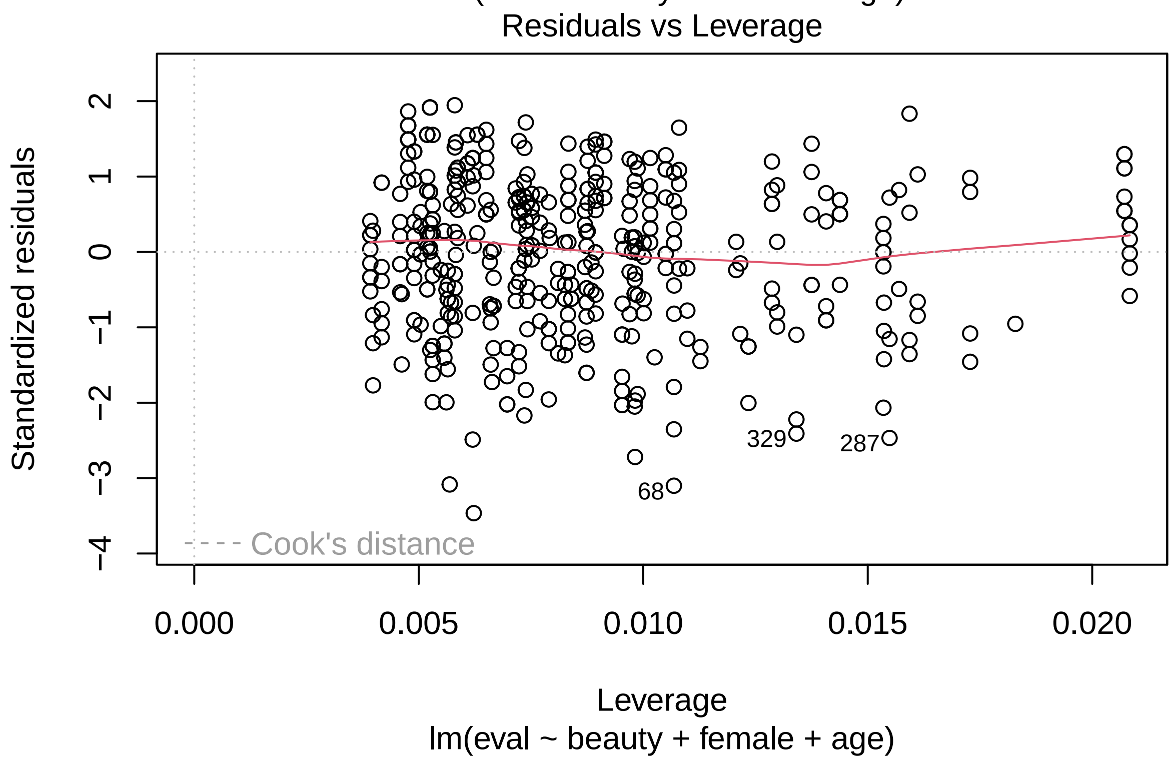


Residuals vs Fitted
Residuals
1.0
0.0
−1.0
−2.0
68
30
99
3.7
3.8
3.9
4.0
4.1
4.2
4.3
Fitted values
lm(eval ~ beauty + female + age)

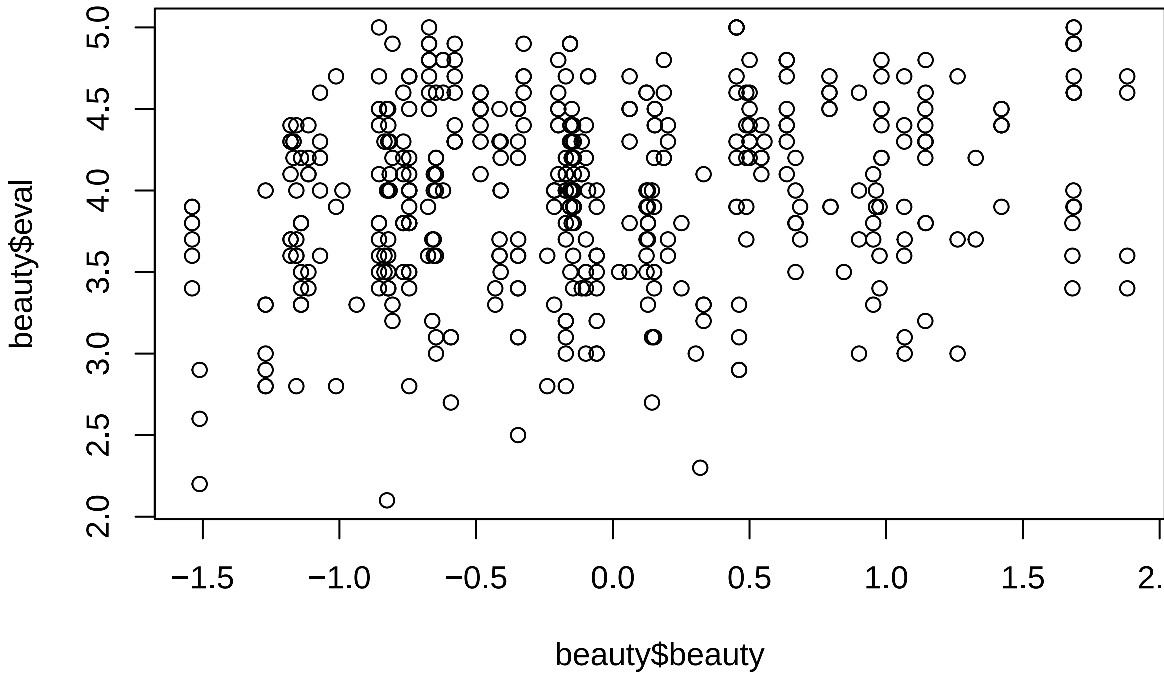


Normal Q−Q
Standardized residuals
2
1
0
−1
−2
−3
99
68
30
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(eval ~ beauty + female + age)

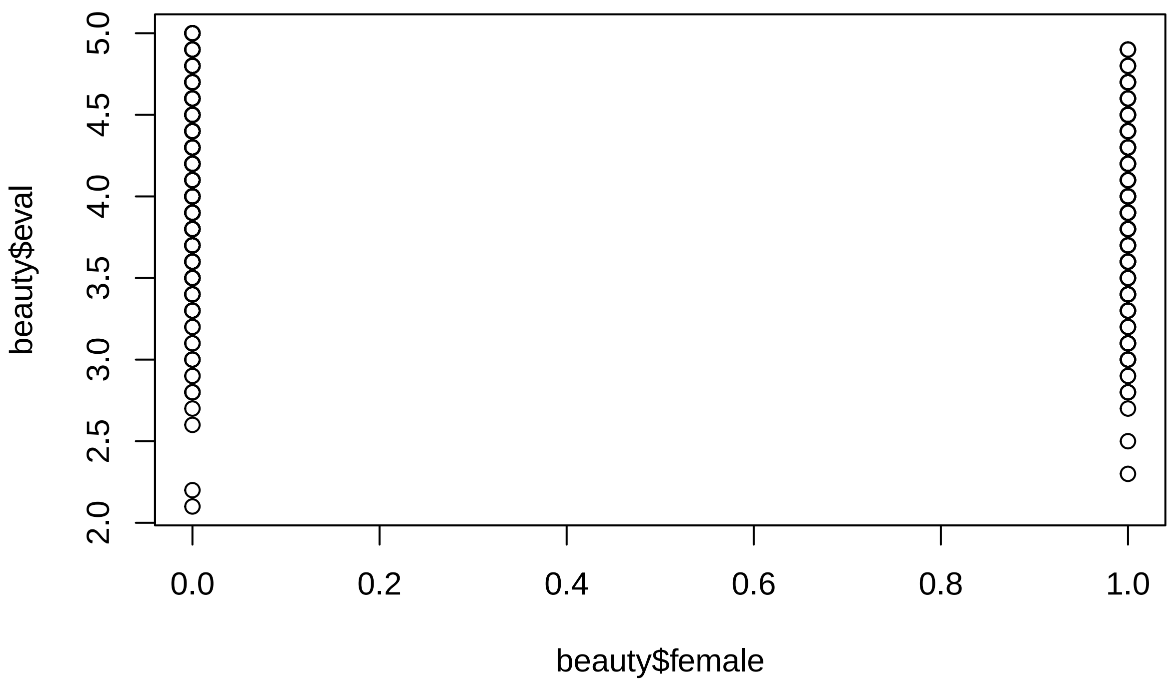





```
secondhalfbeauty <- tail(beauty,40)
predict_eval <- predict(reg10.6a,newdata=secondhalfbeauty)
plot(beauty$beauty, beauty$eval)
```

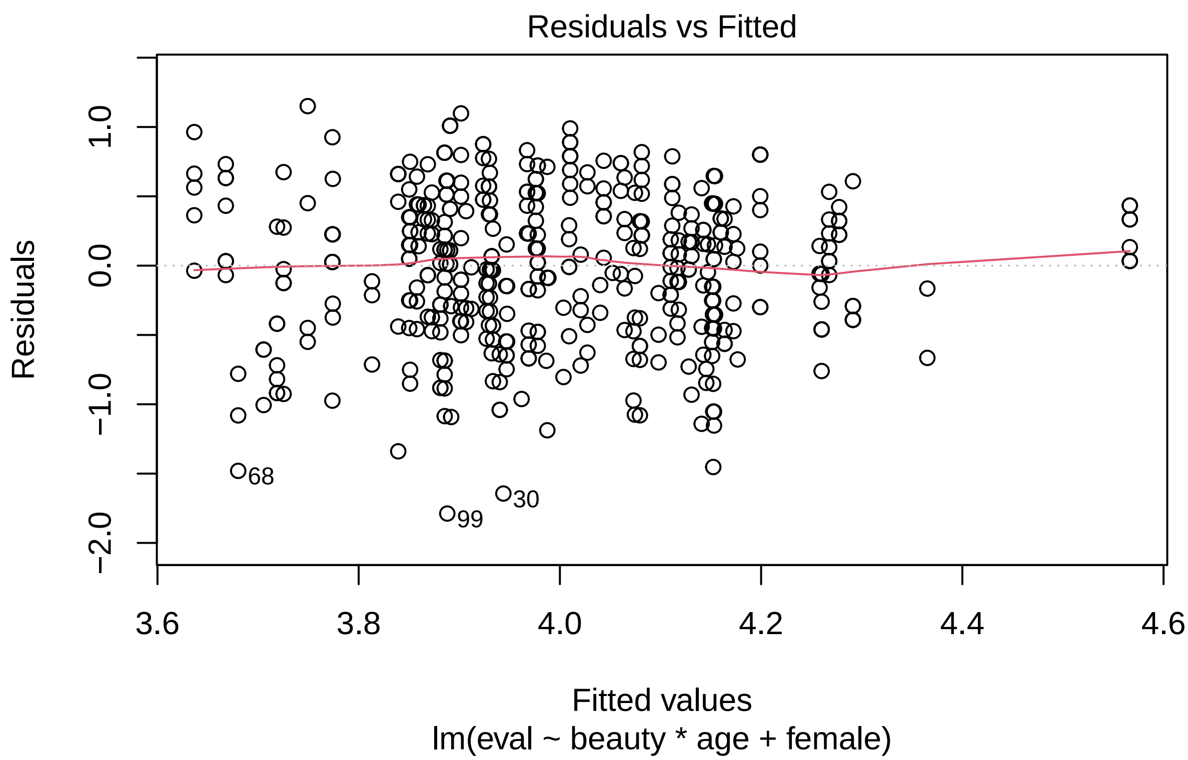


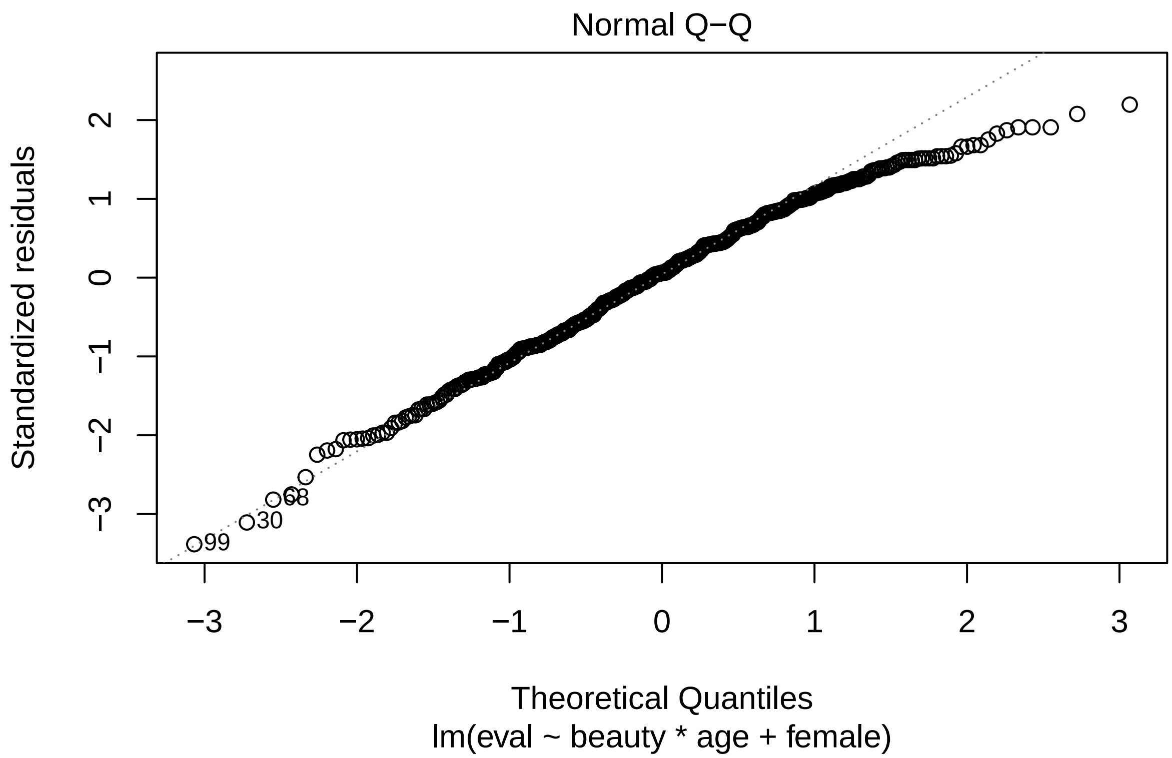

```
plot(beauty$female,beauty$eval)
```

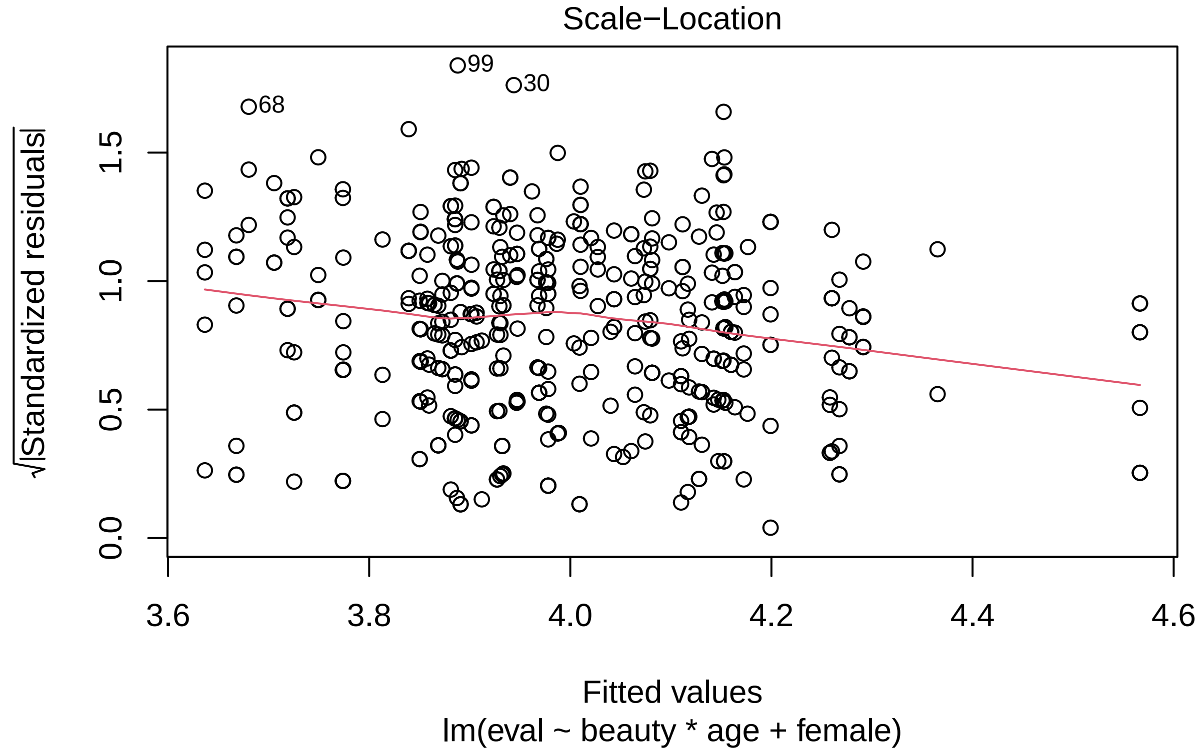


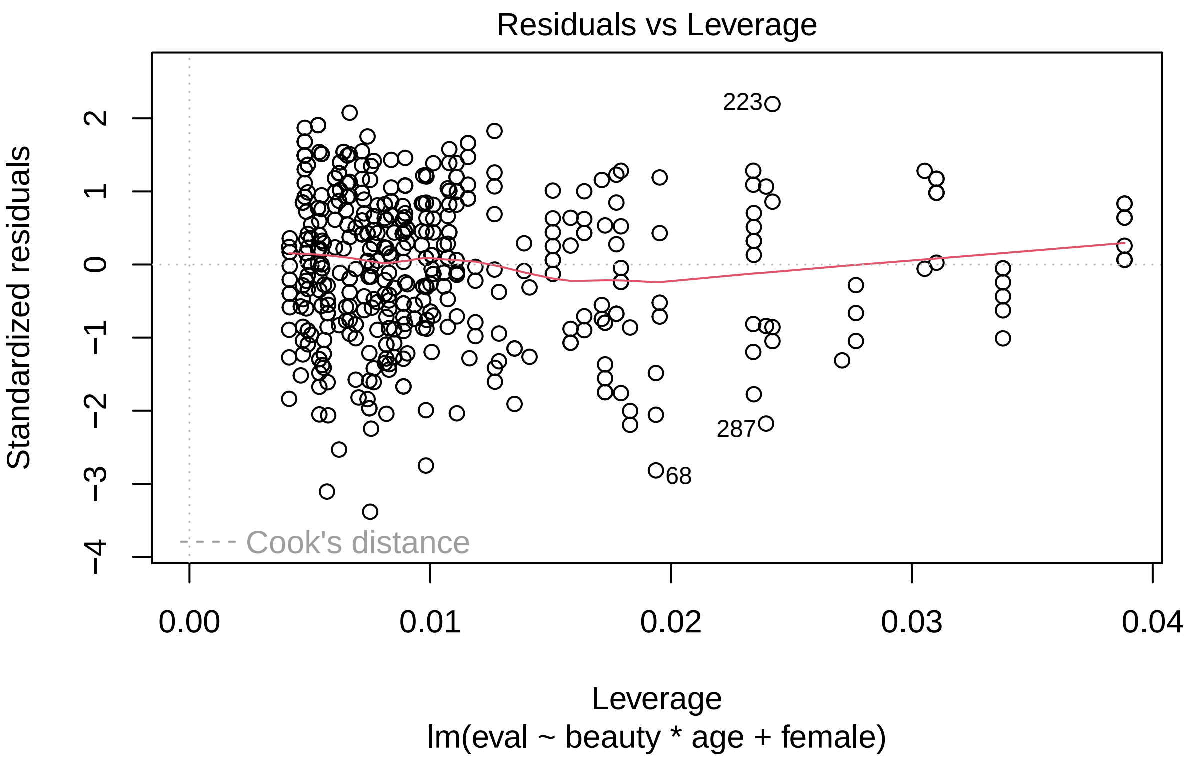

```
plot(beauty$age,beauty$eval)
```


```r
reg10.6b <- lm(eval~beauty*age+female,data=beauty)
plot(reg10.6b)
```




Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(eval ~ beauty * age + female)
99
30
68
−3
−2
−1
0
1
2
3


Scale−Location
√|Standardized residuals|
Fitted values
lm(eval ~ beauty * age + female)
99
30
68
0.0
0.5
1.0
1.5
3.6
3.8
4.0
4.2
4.4
4.6




```
#I would choose the linear model since it's mostly accurate with stable graph.
```

12.14

Prediction from a fitted regression: Consider one of the fitted models for mesquite leaves, for example `fit_4`, in Section 12.6. Suppose you wish to use this model to make inferences about the average mesquite yield in a new set of trees whose predictors are in data frame called new_trees. Give R code to obtain an estimate and standard error for this population average. You do not need to make the prediction; just give the code.

```
#fit_4 <- stan_glm(formula = log(weight) ~ log(canopy_volume) + log(canopy_area)
# + log(canopy_shape) + log(total_height) + log(density) + group, data=mesquite)
#Predict_new_trees <- predict(fit_4,newdata=new_trees)
```