ICA 8

Derien Weatherspoon

2023-03-24

Load libraries and Packages

```
library(ggcorrplot)
```

```
## Loading required package: ggplot2
```

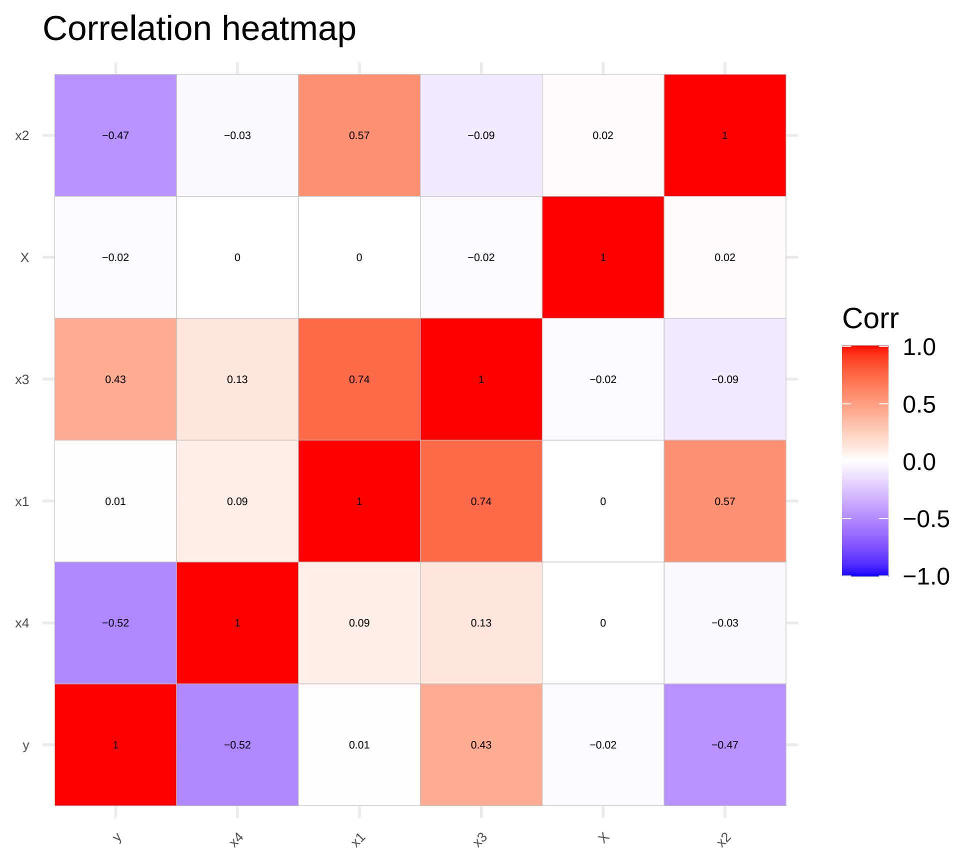
```
df <- read.csv("sampregdata.csv")
head(df)
```

```
##   X        x1        x2         x3         x4          y
## 1 1  7.331693  9.660958 -2.9818692  5.2142838 -2.3118781
## 2 2 16.888609  9.546231 18.4404586 11.8040181  0.9943068
## 3 3  8.280643  6.442062 -0.2257908  6.4932109 -5.2538756
## 4 4  3.827778  3.741506 -2.8506637 -0.2264199 20.0567977
## 5 5  9.675971  4.385008  8.3445758  5.9790432  9.5779140
## 6 6 10.527904 -3.061234 13.4503736 11.8688427 -2.6474849
```

Question 1

Open the file and look at the correlation matrix. Which x variable correlates the most strongly with y? ("Strongest" = highest absolute value of correlation)

```
ggcorrplot(cor(df), lab_size = 1.5, tl.cex = 5, lab = T, title = "Correlation heatmap",  hc.order = TRUE
```




```
round(cor(df),
  digits = 2 # rounded to 2 decimals
)
```

```
##         X   x1    x2    x3    x4     y
## X    1.00 0.00  0.02 -0.02  0.00 -0.02
## x1   0.00 1.00  0.57  0.74  0.09  0.01
## x2   0.02 0.57  1.00 -0.09 -0.03 -0.47
## x3  -0.02 0.74 -0.09  1.00  0.13  0.43
## x4   0.00 0.09 -0.03  0.13  1.00 -0.52
## y   -0.02 0.01 -0.47  0.43 -0.52  1.00
```

The variable x4 correlates the most with variable y, with an absolute value of .52.

Question 2

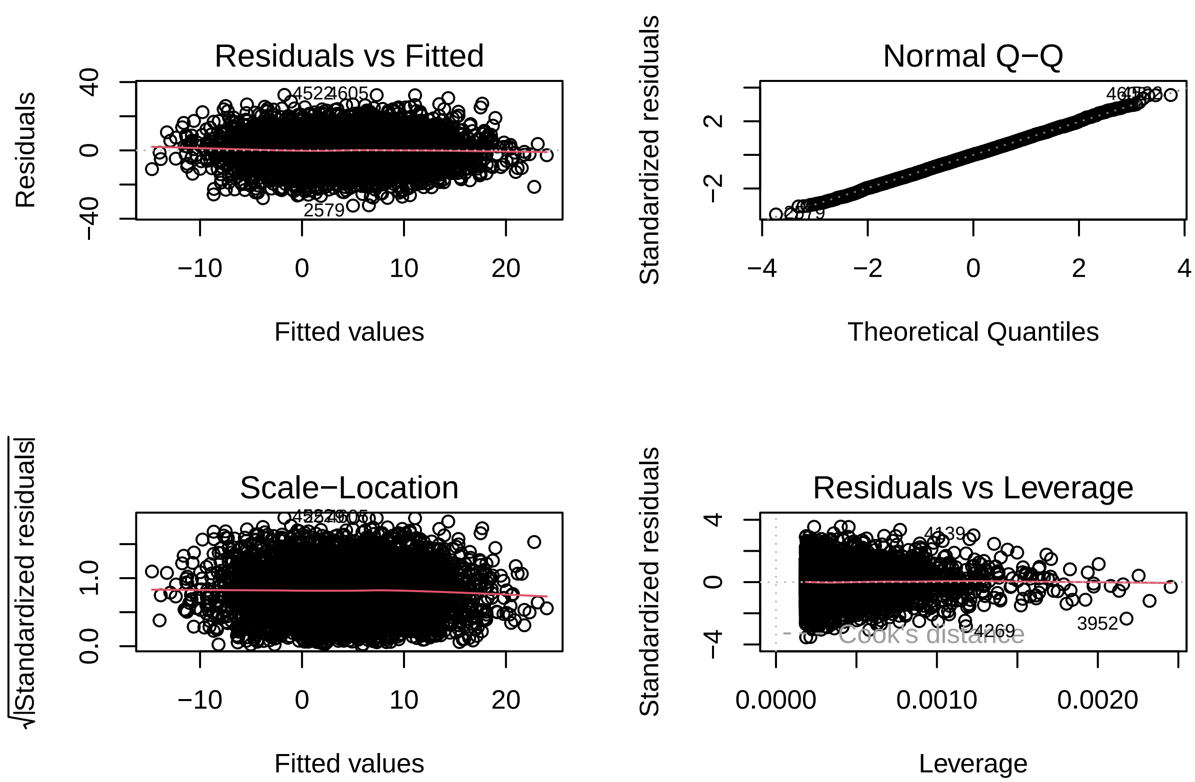
Build a linear regression model for y with that x. Evaluate the residuals by: a. Looking at a normal probability plot (qqnorm()) b. Plot the residuals vs. the actual y value.

```
lm.fit <- lm(y ~ x4, data = df)
summary(lm.fit)
```

```
##
## Call:
```

```
## lm(formula = y ~ x4, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -32.335  -6.065   0.060   6.038  32.475
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.75071    0.15776   55.47   <2e-16 ***
## x4          -1.22446    0.02708  -45.22   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.124 on 5392 degrees of freedom
## Multiple R-squared:  0.275,  Adjusted R-squared:  0.2749
## F-statistic:  2045 on 1 and 5392 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,2))
plot(lm.fit)
```



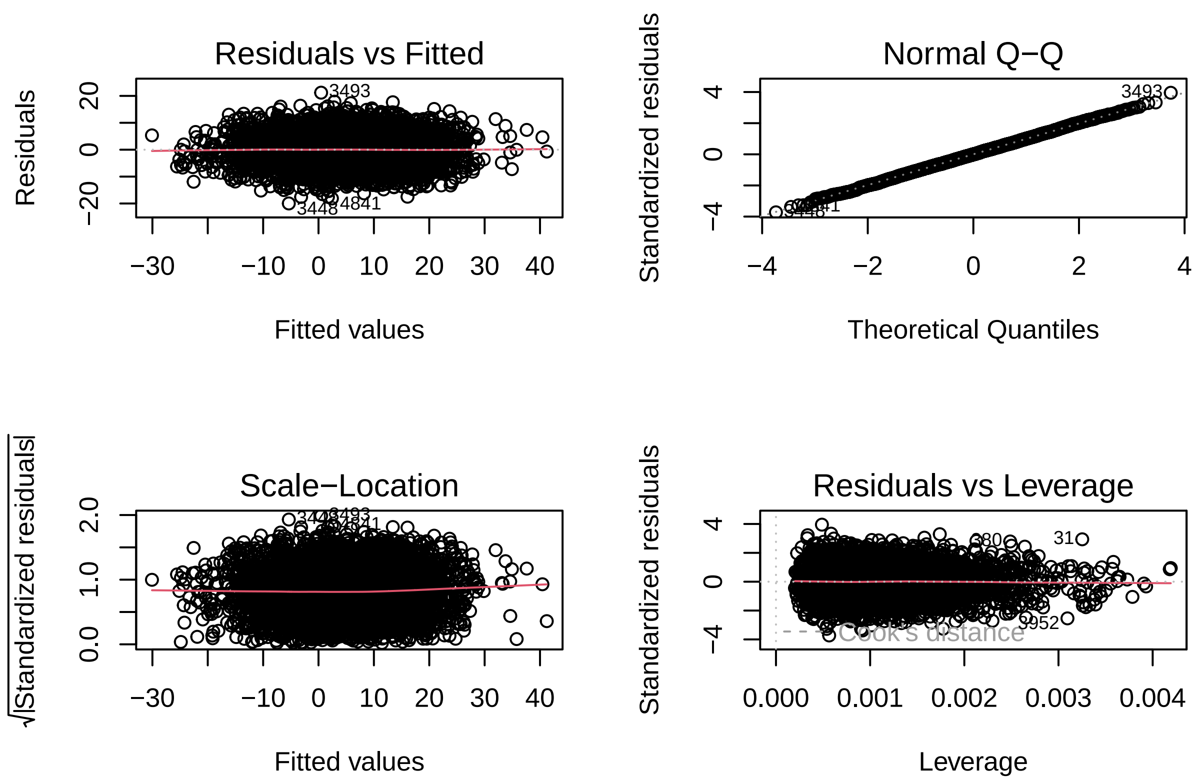
Question 3

Now, build a multiple regression model with all of the x's. How is the fit?

```r
lm.full <- lm(y ~ ., data = df)
summary(lm.full)
```

```
##
## Call:
## lm(formula = y ~ ., data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.9794  -3.5663  -0.0217   3.5915  21.1953
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.591e+01  2.744e-01  57.967   <2e-16 ***
## X           -2.280e-07  4.694e-05  -0.005    0.996
## x1          -1.337e+00  6.100e-02 -21.918   <2e-16 ***
## x2           4.727e-02  3.897e-02   1.213    0.225
## x3           1.204e+00  3.149e-02  38.218   <2e-16 ***
## x4          -1.387e+00  1.607e-02 -86.292   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.364 on 5388 degrees of freedom
## Multiple R-squared:  0.7496, Adjusted R-squared:  0.7493
## F-statistic:  3225 on 5 and 5388 DF,  p-value: < 2.2e-16
```

```r
par(mfrow=c(2,2))
plot(lm.full)
```



Adjusted R^2 has increased, but that is expected since we added all variables. The residuals plots are nearly identical.

Question 4

For this model, for the coefficient with the lowest (absolute) t-value, construct a 95% confidence interval for the coefficient.

```r
# The lowest is x2.
confint(lm.full, "x2")
```

```
##           2.5 %    97.5 %
## x2 -0.02912321 0.1236705
```

Question 5

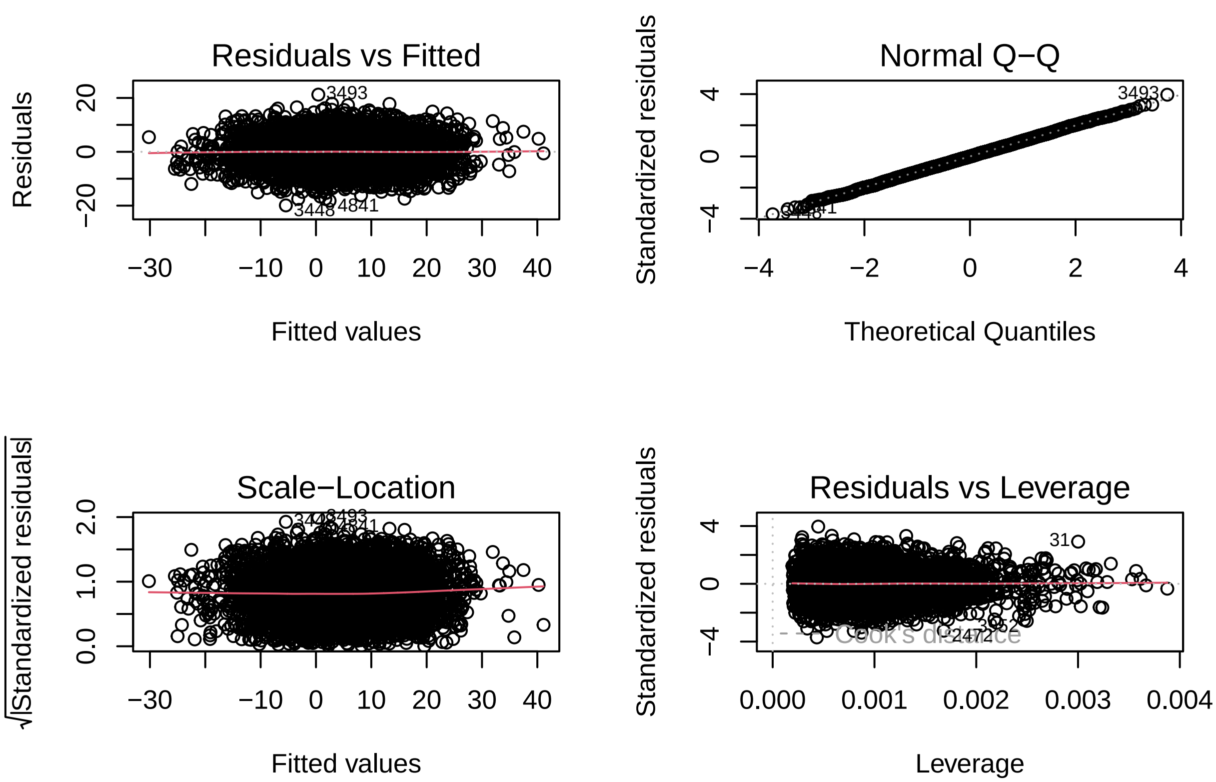
Now, create a linear regression model without this variable with the worst fit. How does the fit compare to the previous model?

```r
lm.fit_2 <- lm(y ~. - x2, data = df)
summary(lm.fit_2)
```

```
##
## Call:
```

```
## lm(formula = y ~ . - x2, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.9050  -3.5740  -0.0004   3.5987  21.2365
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.572e+01  2.271e-01  69.206   <2e-16 ***
## X           -1.346e-06  4.694e-05  -0.029    0.977
## x1          -1.266e+00  1.843e-02 -68.697   <2e-16 ***
## x3           1.168e+00  1.160e-02 100.676   <2e-16 ***
## x4          -1.388e+00  1.606e-02 -86.425   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.365 on 5389 degrees of freedom
## Multiple R-squared:  0.7495, Adjusted R-squared:  0.7493
## F-statistic:  4031 on 4 and 5389 DF,  p-value: < 2.2e-16
```

```r
par(mfrow=c(2,2))
plot(lm.fit_2)
```



Even with the least correlated variable removed, everything still looks pretty similar.