STAT4355HW7

This is an R script with the purpose of checking model adequacy and diagnostics on a football data linear model.

Ramesh Kanakala

(a)

```
#load the data
ftbl <- read.csv(file = 'football.csv')
#build linear model
fit <- lm(y~x2+x7+x8, data = ftbl)

#residual analysis
View(fit)
library(MASS)

#standardized residuals
print("standardized residuals:")
```

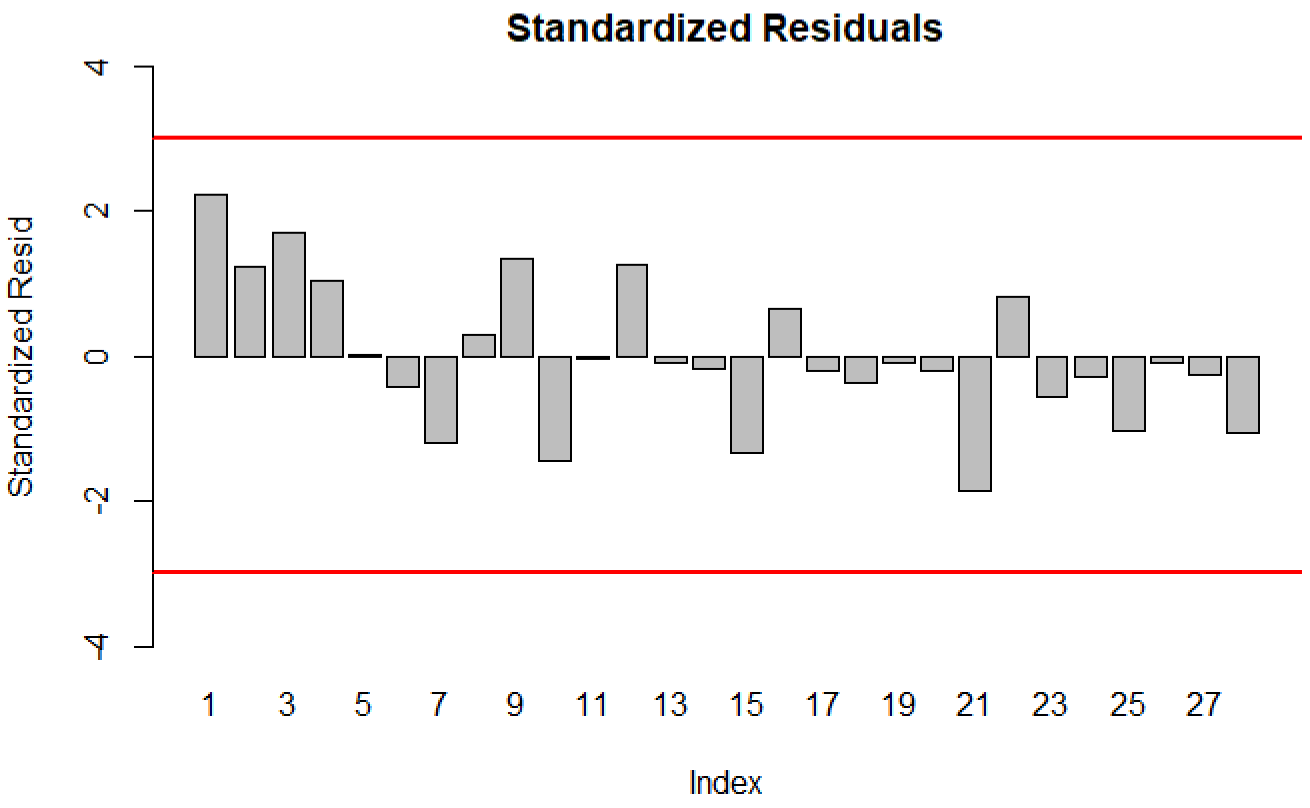
```
[1] "standardized residuals:"
```

```
stdres(fit)
```

```
          1            2            3            4            5            6            7 
 2.231851618  1.225616368  1.702625305  1.029767789  0.006124483 -0.418876221 -1.206836995 
          8            9           10           11           12           13           14 
 0.299328499  1.338032316 -1.441760607 -0.036468456  1.251090093 -0.083851688 -0.160668820 
         15           16           17           18           19           20           21 
-1.335367350  0.644990078 -0.196937383 -0.365011749 -0.078998342 -0.206464327 -1.869940122 
         22           23           24           25           26           27           28 
 0.817274105 -0.551056514 -0.276544687 -1.018586104 -0.094055761 -0.262130195 -1.048746774 
```

```
barplot(height = stdres(fit), names.arg = 1:28,
        main = "Standardized Residuals", xlab = "Index",
        ylab = "Standardized Resid", ylim=c(-4,4))
#Add cutoff values. Either 2 or 3 can be chosen.
abline(h=3, col = "Red", lwd=2)
```

```
abline(h=-3, col = "Red", lwd=2)
```



Hide

```
#studentized residuals
print("studentized residuals:")
```

```
[1] "studentized residuals:"
```

Hide

```
studres(fit)
```

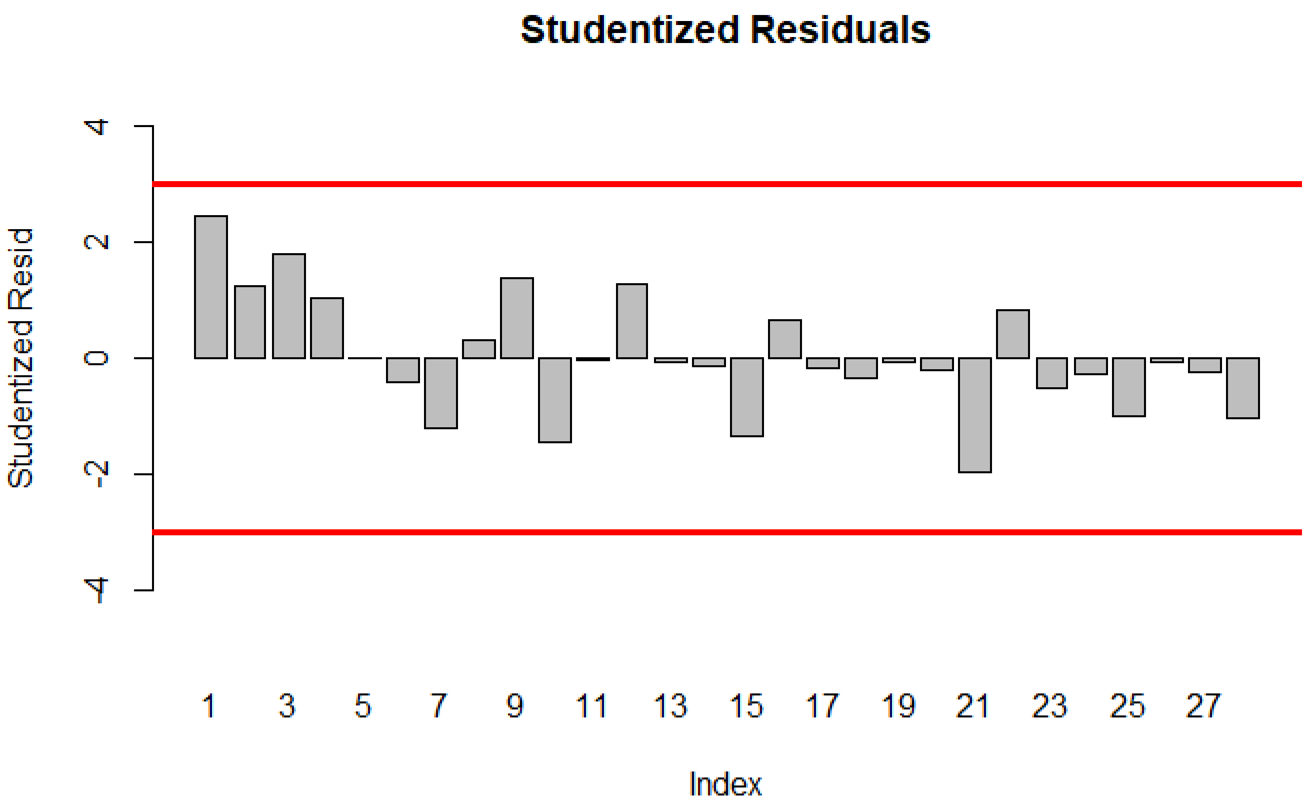
```
           1            2            3            4            5            6            7 
 2.454354223  1.239218310  1.777586702  1.031123075  0.005995537 -0.411563960 -1.218993620 
           8            9           10           11           12           13           14 
 0.293574644  1.361631132 -1.476806719 -0.035701602  1.266752172 -0.082098218 -0.157370596 
          15           16           17           18           19           20           21 
-1.358701256  0.636954384 -0.192946834 -0.358322410 -0.077345090 -0.202296957 -1.980521136 
          22           23           24           25           26           27           28 
 0.811437522 -0.542899513 -0.271154408 -1.019417881 -0.092092392 -0.256979177 -1.051031132 
```

Hide

```
barplot(height = studres(fit), names.arg = 1:28,
        main = "Studentized Residuals", xlab = "Index",
        ylab = "Studentized Resid", ylim=c(-5,5))
#Add cutoff values. Either 2 or 3 can be chosen.
abline(h=3, col = "Red", lwd=3)
```

Hide

```
abline(h=-3, col = "Red", lwd=3)
```



Hide

```

#R-student residuals
print("R-student residuals:")
```

```
[1] "R-student residuals:"
```

Hide

```
RStudent <- rstudent(fit)
RStudent
```

```
           1            2            3            4            5            6            7 
 2.454354223  1.239218310  1.777586702  1.031123075  0.005995537 -0.411563960 -1.218993620 
           8            9           10           11           12           13           14 
 0.293574644  1.361631132 -1.476806719 -0.035701602  1.266752172 -0.082098218 -0.157370596 
          15           16           17           18           19           20           21 
-1.358701256  0.636954384 -0.192946834 -0.358322410 -0.077345090 -0.202296957 -1.980521136 
          22           23           24           25           26           27           28 
 0.811437522 -0.542899513 -0.271154408 -1.019417881 -0.092092392 -0.256979177 -1.051031132 
```

Hide

```
barplot(height = RStudent, names.arg = 1:28,
        main = "R Student Residuals", xlab = "Index",
        ylab = "R Student Resid", ylim=c(-5,5))
cor.level <- 0.05/(2*25)
cor.qt <- qt(cor.level, 21, lower.tail=F)
RStudent> cor.qt
```

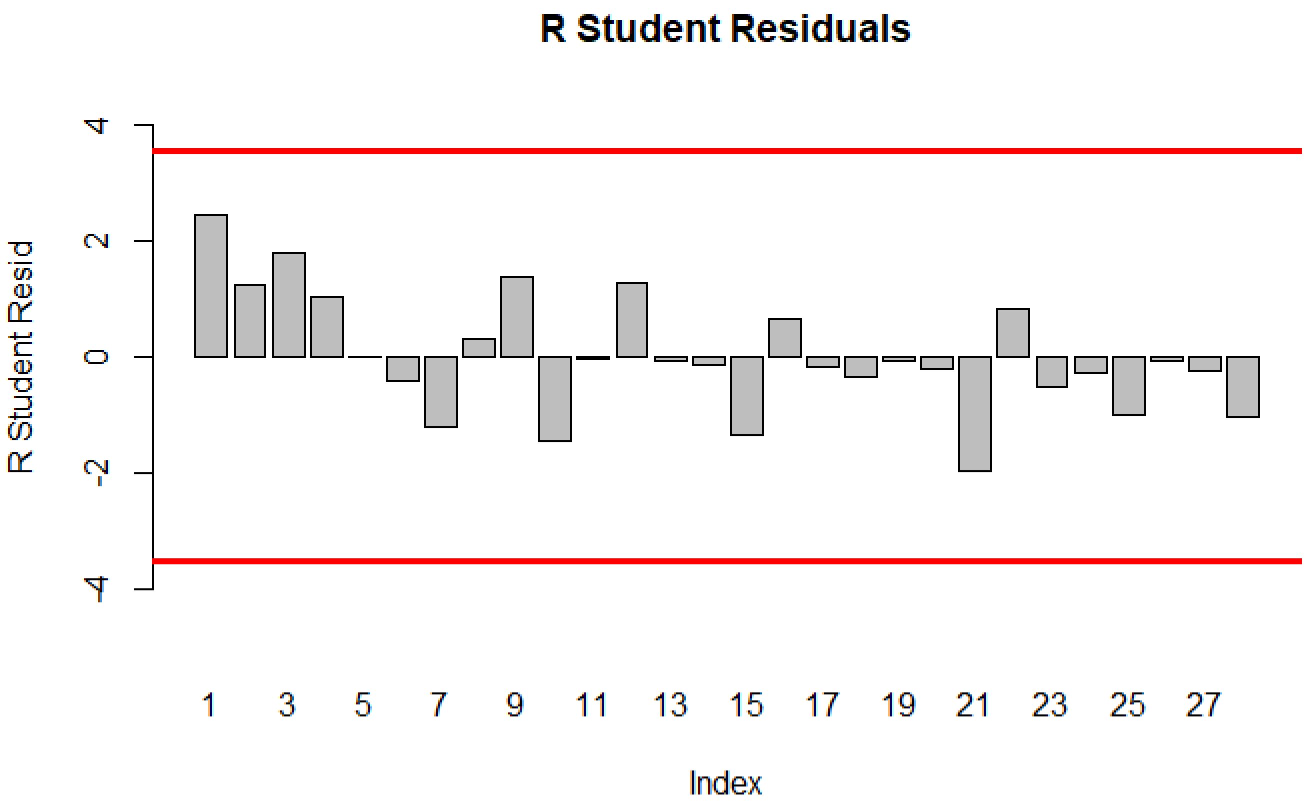
```
    1     2     3     4     5     6     7     8     9    10    11    12    13    14    15    16 
FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE 
   17    18    19    20    21    22    23    24    25    26    27    28 
FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE 
```

Hide

```
abline(h=cor.qt , col = "Red", lwd=3)
```

Hide

```
abline(h=-cor.qt , col = "Red", lwd=3)
```



i. Standardized, studentized, and R-student residuals printed above^
ii. Standardized, studentized, and R-student residuals plotted above^
iii. We see that all residuals fall below the cutoff limits but the first few data points, 7, 9, 10, 16, and 21 stick out with 1 and 21 being the largest.

(b)

Hide

```
#influential analysis
myInf <- influence.measures(fit)
myInf
```

```
Influence measures of
     lm(formula = y ~ x2 + x7 + x8, data = ftbl) :
```

	dfb.1_ <dbl>	dfb.x2 <dbl>	dfb.x7 <dbl>	dfb.x8 <dbl>	dffit <dbl>	cov.r <dbl>	cook.d <dbl>
1	-0.253803050	-0.035359684	2.818317e-01	0.283928685	0.583113786	0.4939373	7.029149e-02
2	0.038132987	0.319290856	-1.008030e-01	-0.065149862	0.458327453	1.0407210	5.136949e-02
3	-0.215883184	-0.165662173	3.679037e-01	0.066068969	0.648881707	0.8028461	9.657121e-02
4	-0.194469937	0.371071932	1.784098e-01	0.052301698	0.449386405	1.1774862	5.035440e-02
5	0.000824700	-0.001228852	2.568992e-05	-0.001641964	0.002924707	1.4677037	2.231451e-06
6	0.054961892	-0.144244400	-5.004646e-02	0.003146865	-0.178447873	1.3677346	8.246307e-03
7	0.101879699	-0.147098179	-1.955443e-01	0.124552875	-0.501947712	1.0794403	6.173783e-02
8	0.024880288	0.059207355	-2.580832e-02	-0.049221625	0.107135887	1.3235318	2.983108e-03
9	-0.138395769	-0.327600702	1.432716e-01	0.346584847	0.625639593	1.0530385	9.449367e-02
10	0.322817662	0.054002137	-3.076018e-01	-0.426020021	-0.548685710	0.9391023	7.173421e-02

1-10 of 28 rows | 1-8 of 9 columns Previous **1** 2 3 Next

Hide

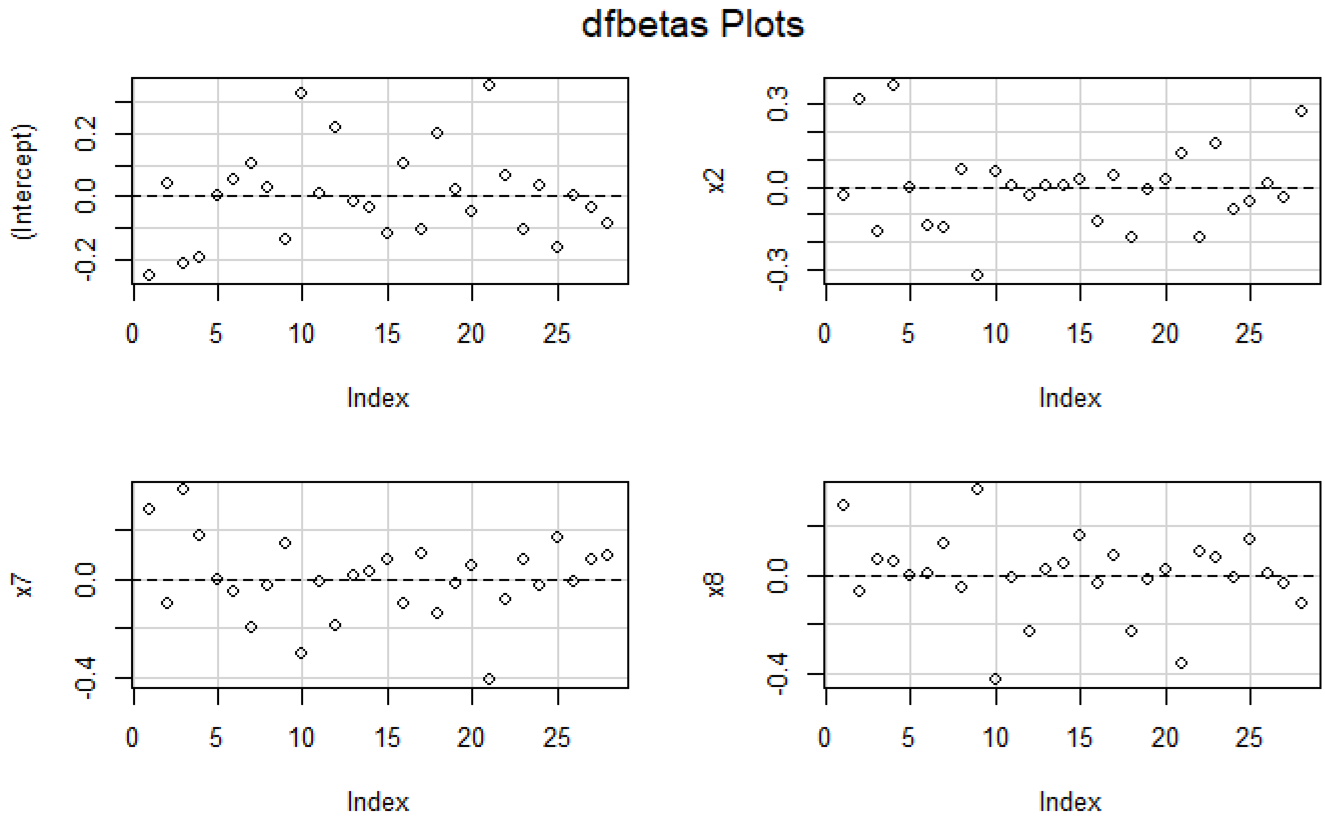
```
summary(myInf)
```

```
Potentially influential observations of
     lm(formula = y ~ x2 + x7 + x8, data = ftbl) :

   dfb.1_ dfb.x2 dfb.x7 dfb.x8 dffit cov.r   cook.d hat 
1  -0.25  -0.04   0.28   0.28   0.58  0.49_*  0.07   0.05
11  0.01   0.01  -0.01  -0.01  -0.02  1.51_*  0.00   0.21
17 -0.10   0.04   0.11   0.08  -0.11  1.59_*  0.00   0.26
18  0.20  -0.19  -0.14  -0.23  -0.29  1.91_*  0.02   0.39
27 -0.04  -0.04   0.08  -0.03  -0.18  1.72_*  0.01   0.32
```

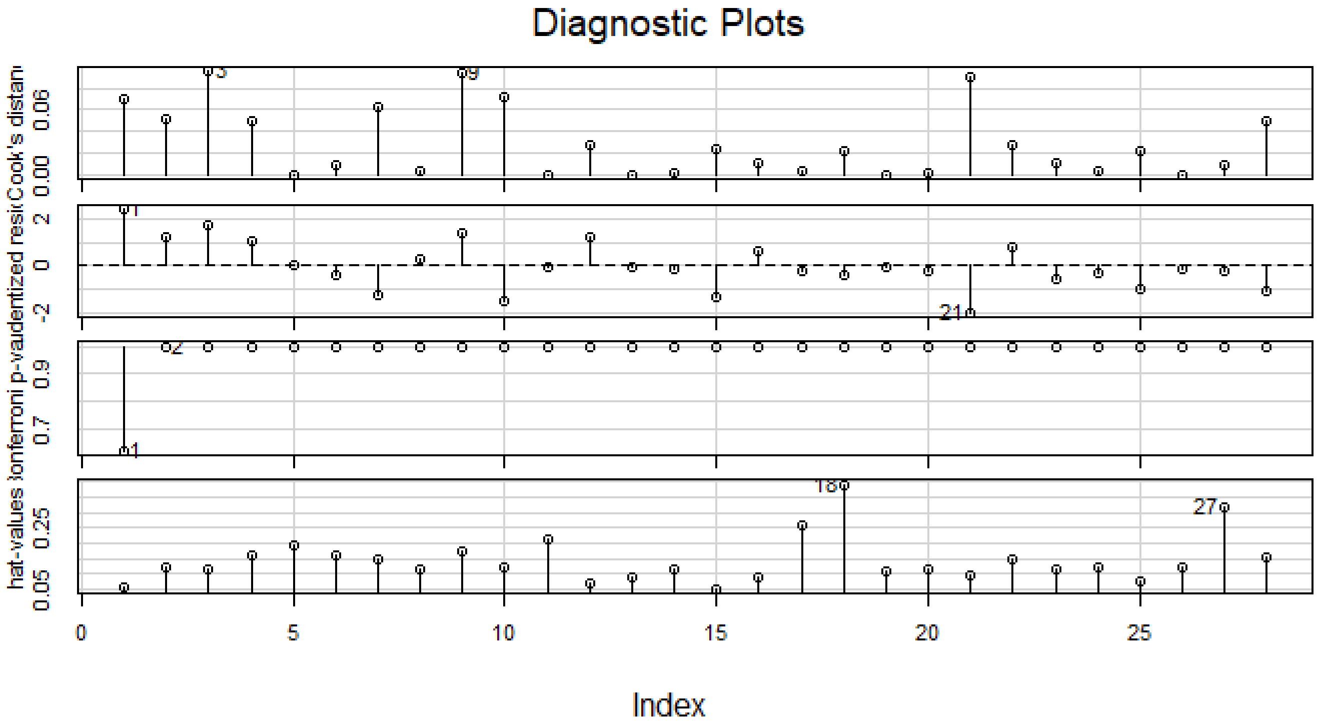
Hide

```
library(car)
dfbetasPlots(fit,intercept=T)
```



Hide

```
influenceIndexPlot(fit)
```



Data points 1, 4, 17, 18, and 27 were flagged to be outliers. Looking at the dfbetas plots, we see that most of the influential values for each predictor are around 0 and all points seem to be inside the cutoffs. Looking at the diagnostic plots, we see again that 1 sticks out, for three of the plots, and 21 sticks out, for the second and somewhat the first. Overall, however, the data has no major issues and seems normal.

(c)

Hide

```
#variance inflation factors
vif(fit)
```

```
      x2       x7       x8 
1.115977 2.097311 2.021254 
```

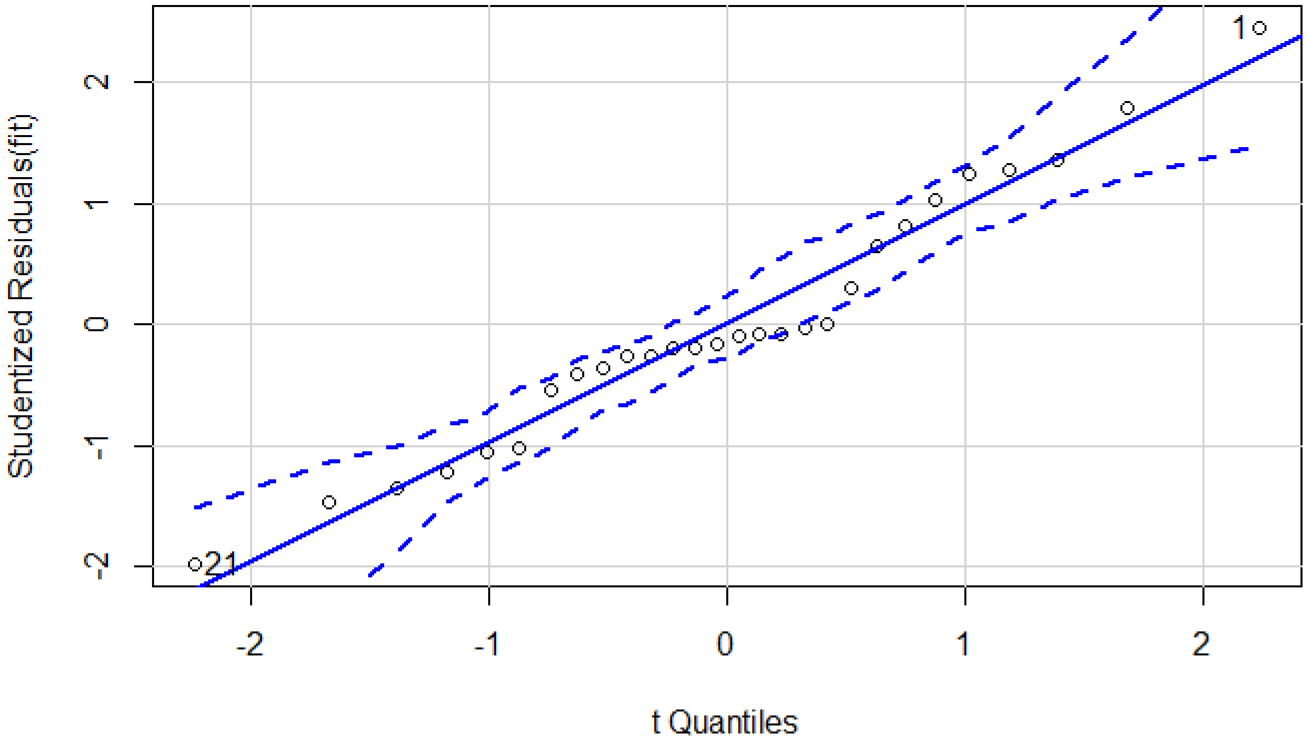
The VIF values are quite low for each predictor, much less than 10 and actually less than 5; there is no multicollinearity because VIFs are less than threshold. x2 has lowest variance inflation and x7 the highest.

(d)

Hide

```
#normal probability plot of residuals
qqPlot(fit)
```

```
[1]  1 21
```

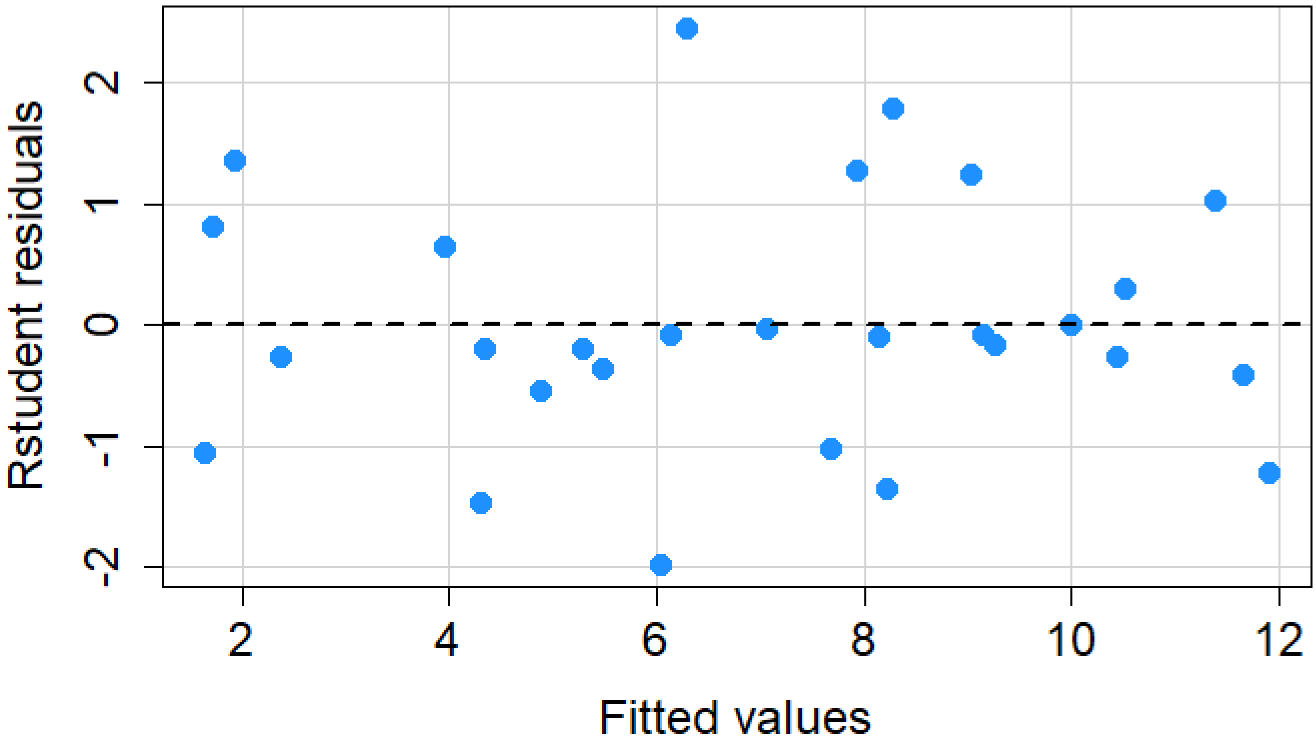


Overall, the normality is not too bad and a few data points are barely outside the curve but I would still go with them. 1 and 21 are flagged but they are also not too bad. I would say the normality assumption is met, but maybe not well met.

(e)

Hide

```
#plot of the residuals versus the fitted values
par(mfrow=c(1,1))
residualPlot(fit, type="rstudent", quadratic=F, col = "dodgerblue",
             pch=16, cex=1.5, cex.axis=1.5, cex.lab=1.5)
```



The residuals are randomly distributed between -3 and 3 forming a horizontal band around the zero line. They are in a random pattern so no residuals stand out; this is satisfactory and we can assume there is likely a linear relationship.

(f)

R code is within this notebook