pset5

Peter

2023-10-13

```
library(Stat2Data)
library(mosaic)
```

```
## Registered S3 method overwritten by 'mosaic':
##   method                           from
##   fortify.SpatialPolygonsDataFrame ggplot2
```

```
##
## The 'mosaic' package masks several functions from core packages in order to add
## additional features.  The original behavior of these functions should not be affected by this.
```

```
##
## Attaching package: 'mosaic'
```

```
## The following objects are masked from 'package:dplyr':
##
##     count, do, tally
```

```
## The following object is masked from 'package:Matrix':
##
##     mean
```

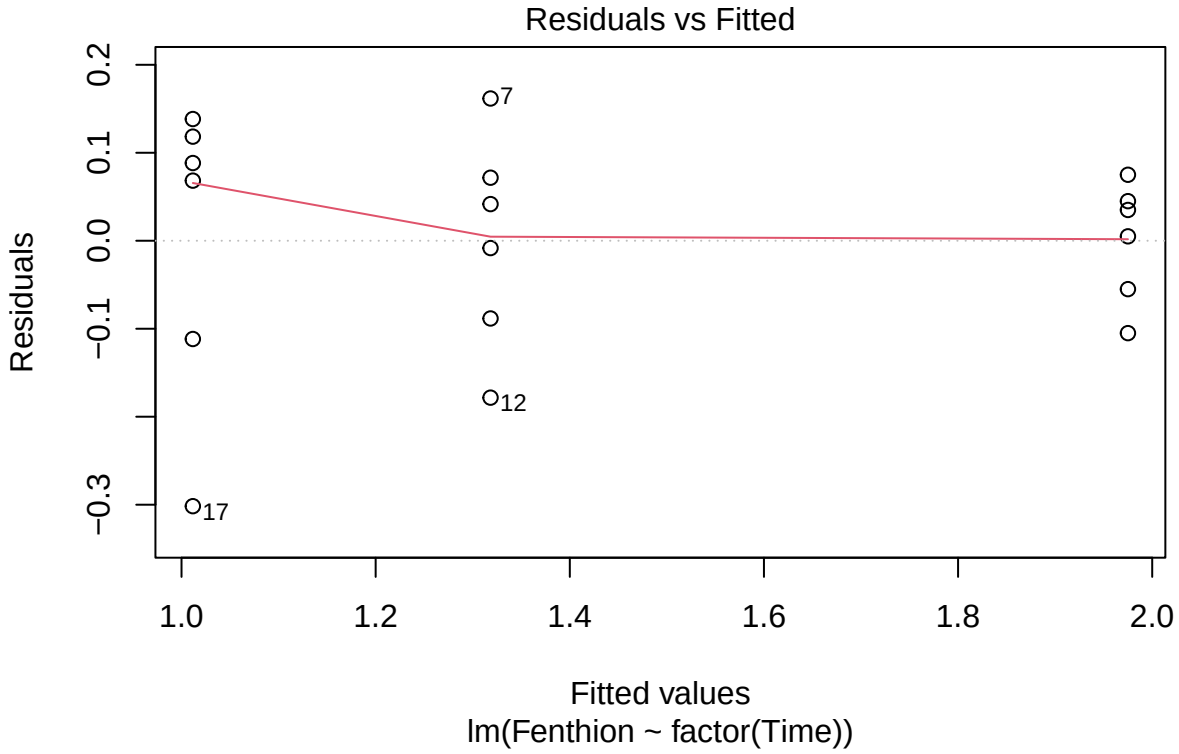
```
## The following object is masked from 'package:ggplot2':
##
##     stat
```

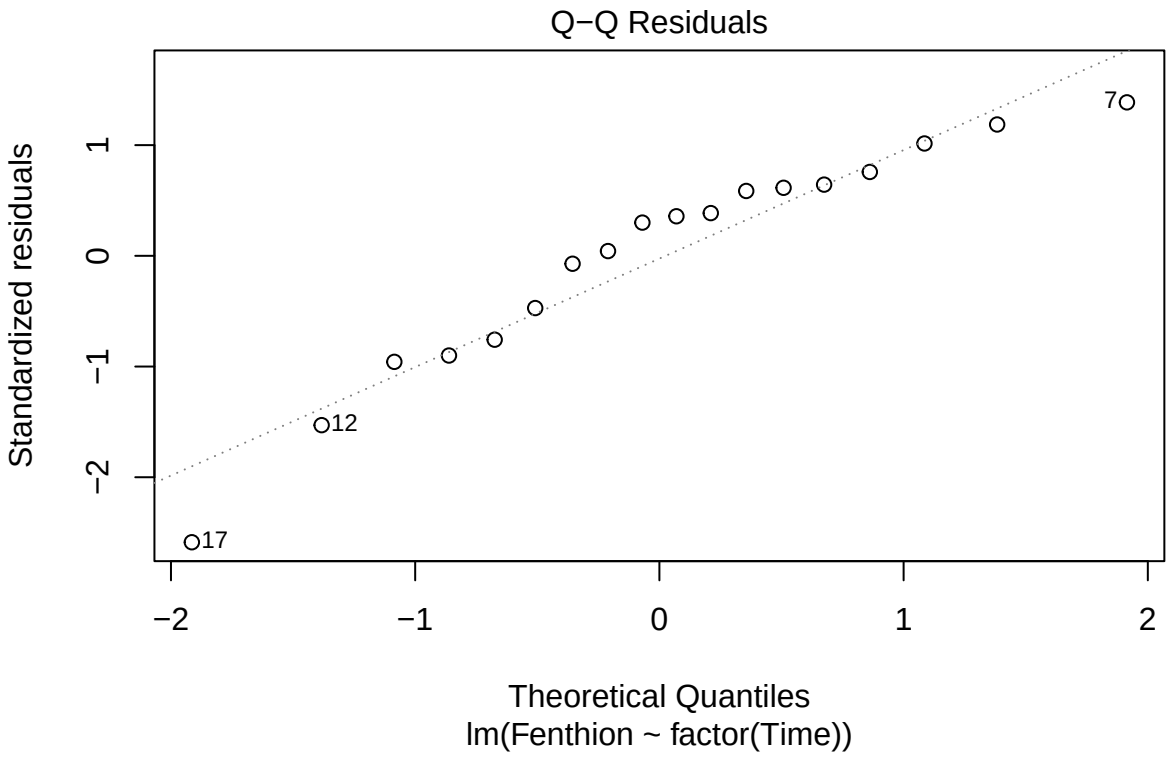
```
## The following objects are masked from 'package:stats':
##
##     binom.test, cor, cor.test, cov, fivenum, IQR, median, prop.test,
##     quantile, sd, t.test, var
```

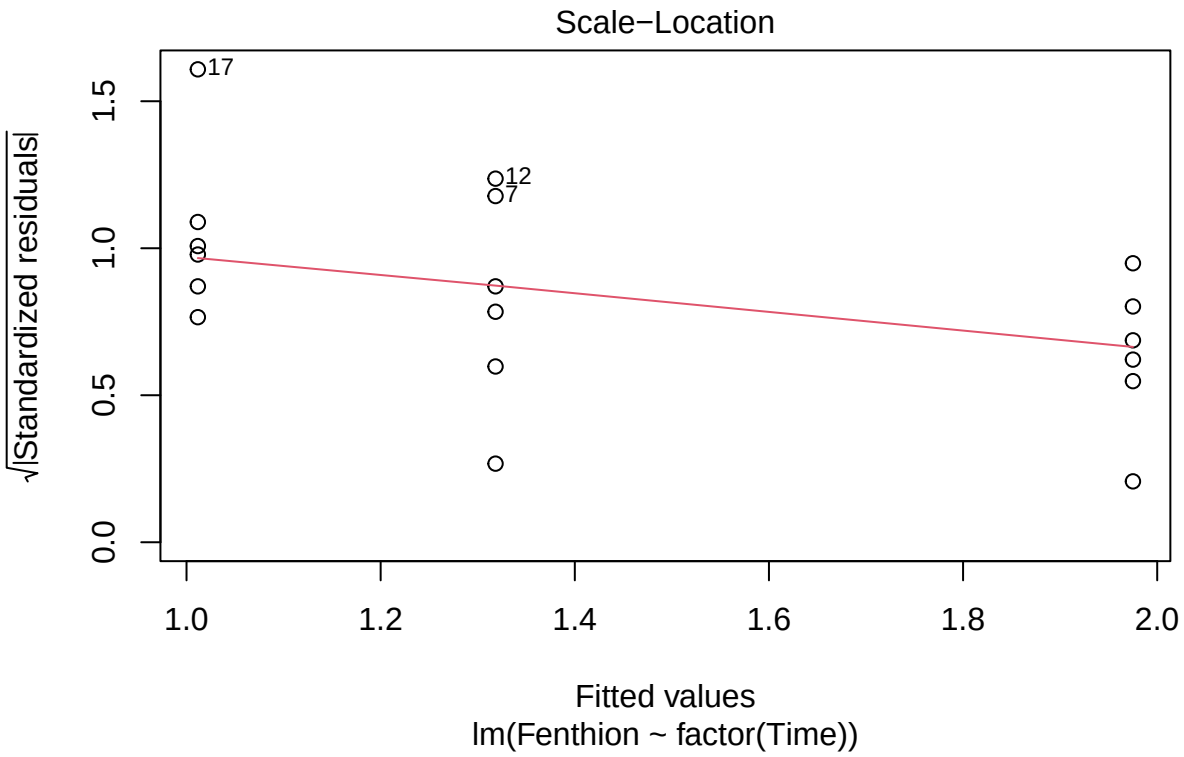
```
## The following objects are masked from 'package:base':
##
##     max, mean, min, prod, range, sample, sum
```

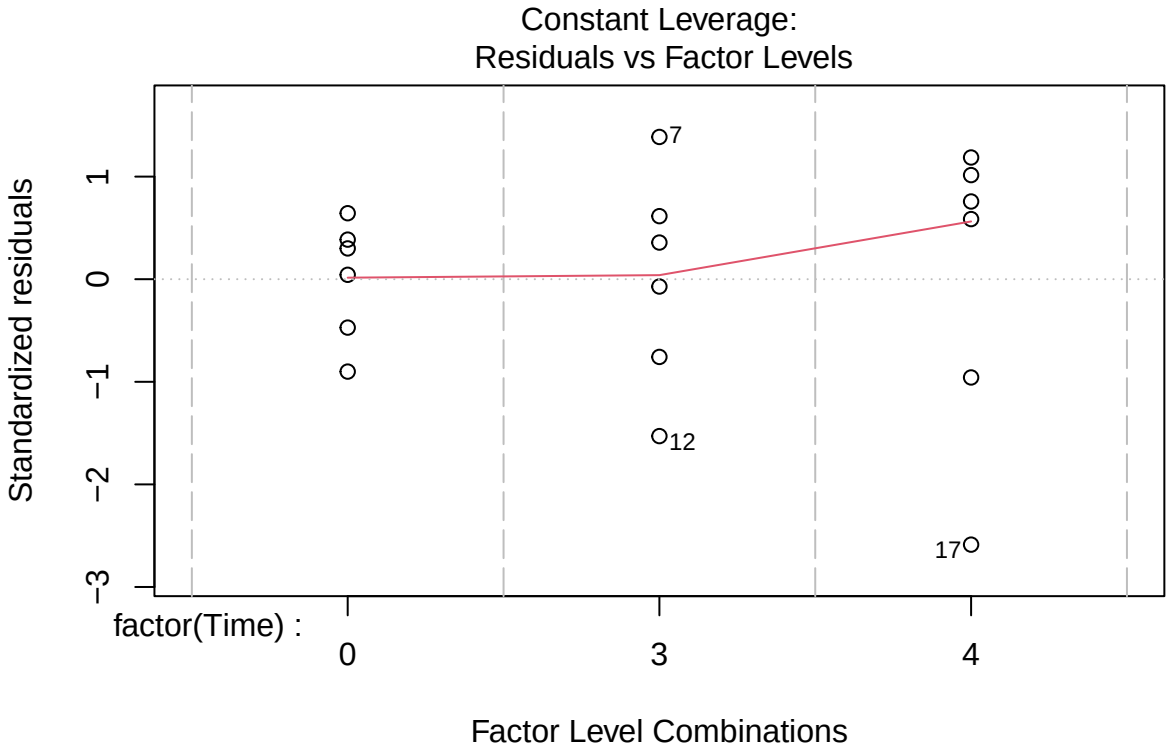
```
#5.12:

#a: Self reported speeds, observational study, not a randomized experiment,
#b: The ones who reported that they listened to heavy metal are more likely to report
#speeding, this can be extended to other similar schools.
```

```r
#5.44 (only a/b)

#5.44a:
data("Olives")
#Fenthion is the response, time is the explanatory, because it's the amount of fenthion
#left on the plant after a certain amount of time.

#5.44b:
timeee <- lm(Fenthion~factor(Time), data = Olives)
plot(timeee)
```




Q−Q Residuals
Standardized residuals
−2
−1
0
1
−2
−1
0
1
2
17
12
7
Theoretical Quantiles
lm(Fenthion ~ factor(Time))


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
17
12
7
1.0
1.2
1.4
1.6
1.8
2.0
Fitted values
lm(Fenthion ~ factor(Time))



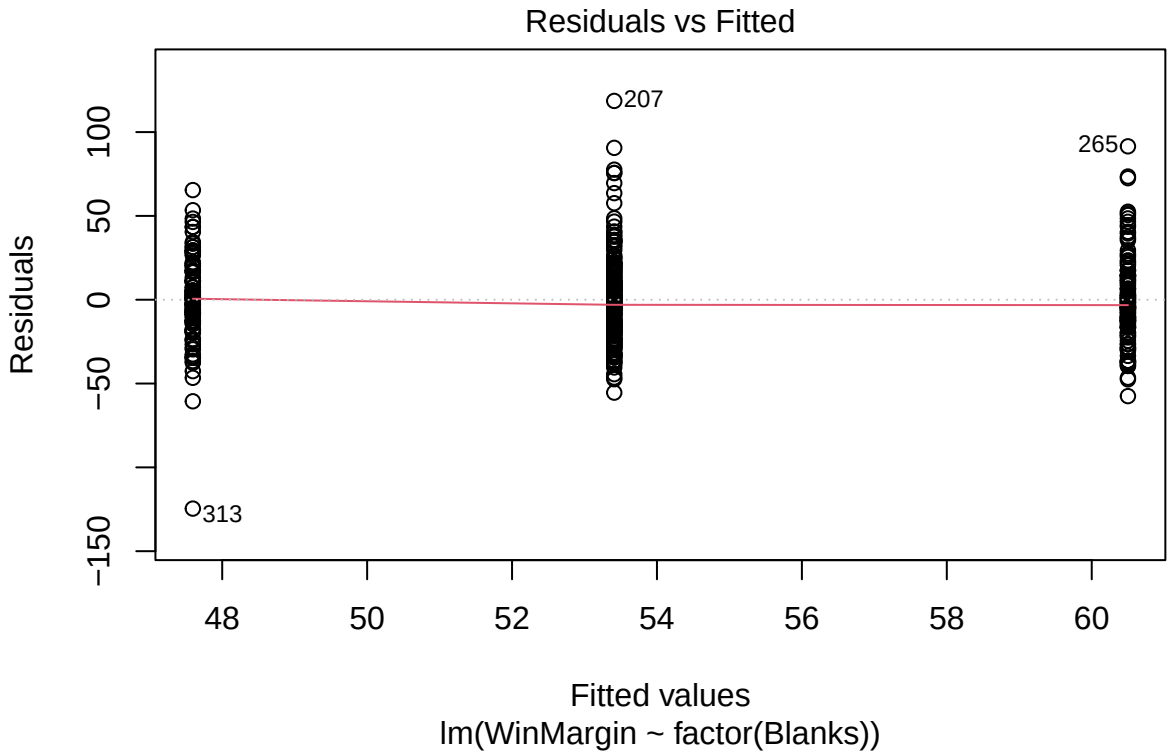

```
favstats(Fenthion~factor(Time), data = Olives)
```

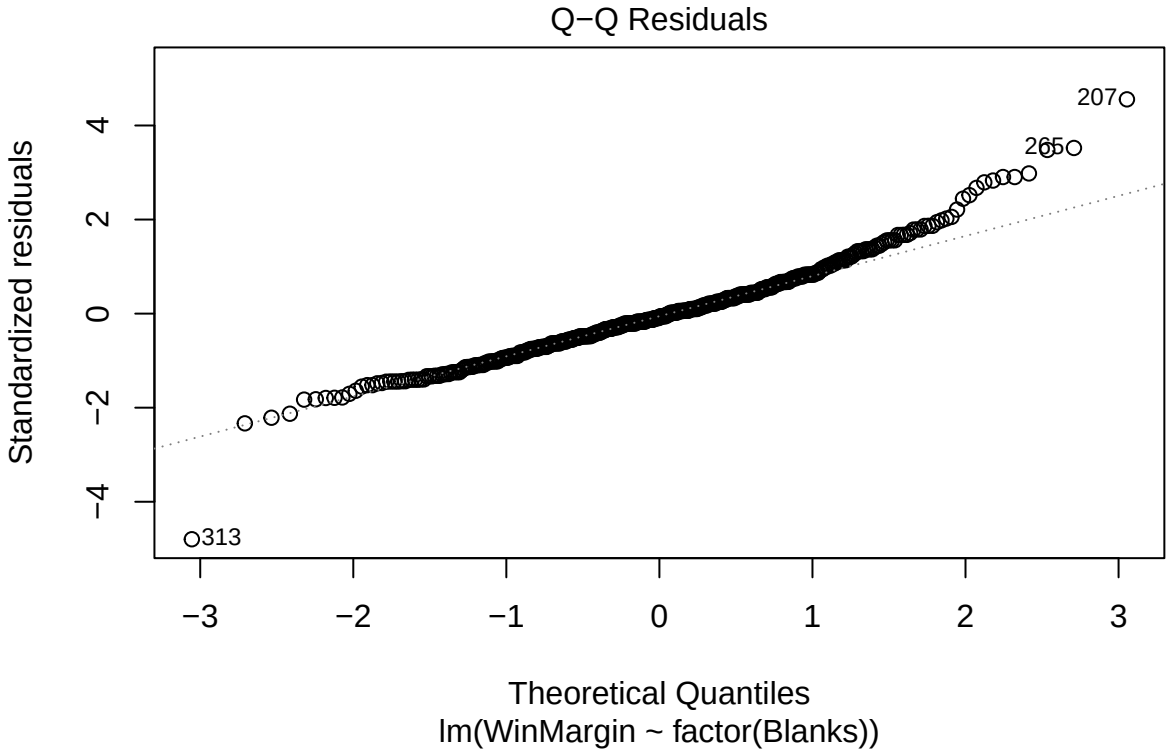
```
##   factor(Time)  min    Q1 median     Q3  max     mean         sd n missing
## 1            0 1.87 1.935  1.995 2.0175 2.05 1.975000 0.06774954 6       0
## 2            3 1.14 1.250  1.335 1.3825 1.48 1.318333 0.12056810 6       0
## 3            4 0.71 0.945  1.090 1.1225 1.15 1.011667 0.17267503 6       0
```

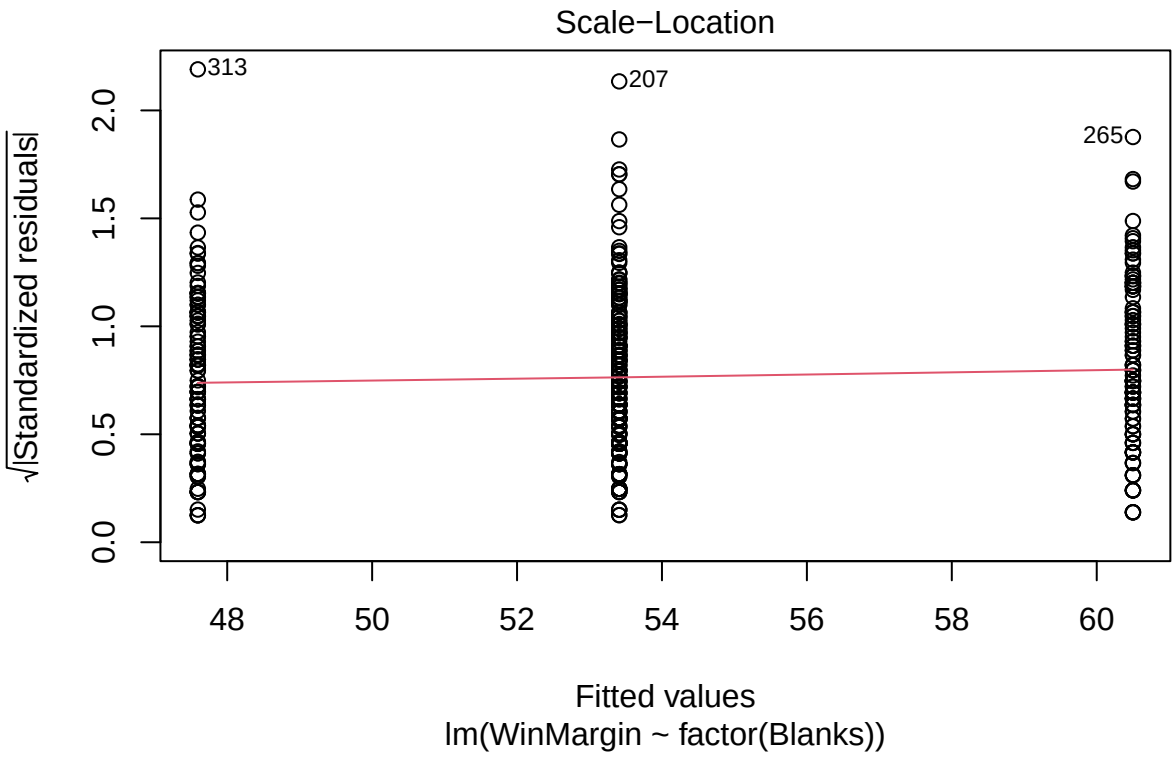
```
#Constant and additive: Yes the treatment are constant and additive based on the context.

#Errors zero-mean: According to the resids vs fits plot, yes the errors are 0 mean visually.

#Errors have same variance: This condition is not met, our lowest SD (.0678) * 2 is
#NOT greater than the largest SD (.173). The errors do NOT have the same variance.

#Errors are normal: With so few observations it's hard to tell, but there is a slight
#curve in the Q-Q residuals plot, with pts 17 and 7 being fairly concerning. I
#would say that normality is not met, or at least very questionable.

#errors are independent: The errors might not be independent because the amount of
#fenthion oil is dependent on the amount there was on the plant at a different time.
```

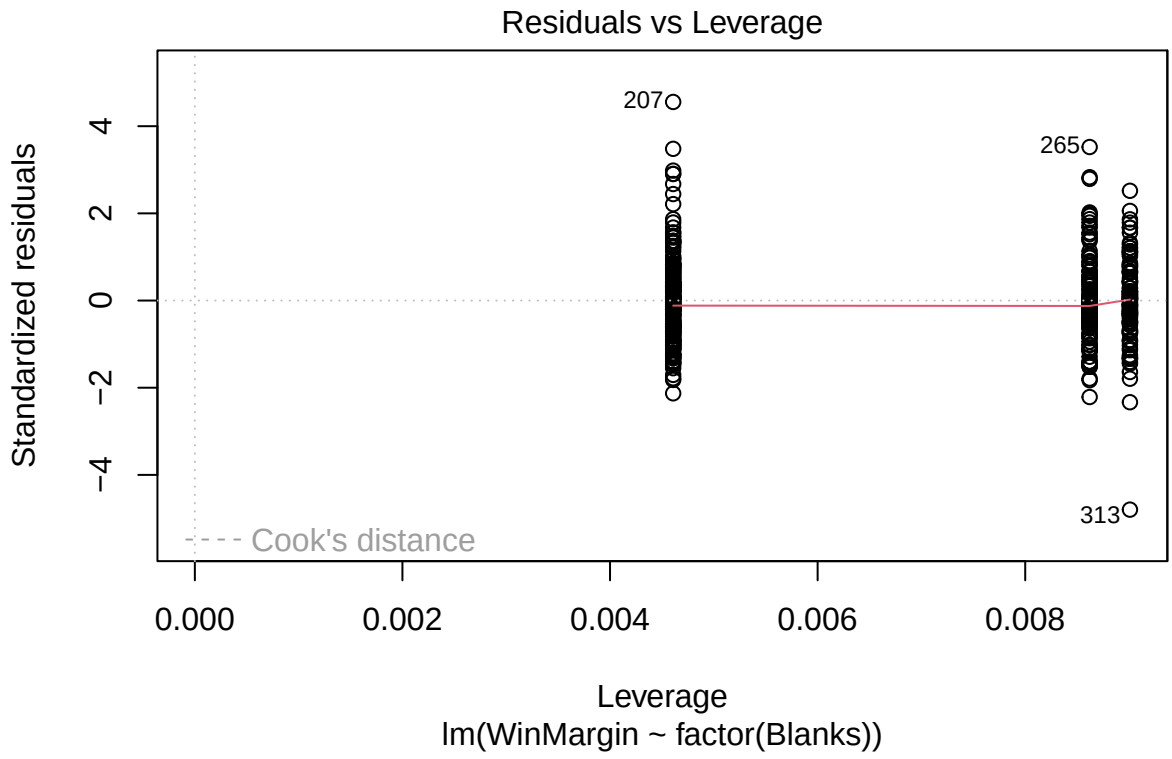
```
#5.52:
data("WordsWithFriends")
methoddd <- lm(WinMargin~factor(Blanks), data = WordsWithFriends)
```

```r
#5.52a:
plot(methoddd)
```




Q−Q Residuals
Standardized residuals
4
2
0
−2
−4
207
265
313
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(WinMargin ~ factor(Blanks))


Scale−Location
313
207
265
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
48
50
52
54
56
58
60
Fitted values
lm(WinMargin ~ factor(Blanks))




```
#Constant and additive: Yes the treatment are constant and additive based on the context.

#Errors zero-mean: According to the resids vs fits plot, yes the errors are 0 mean visually.

#Errors have same variance: The smallest SD (25.7) * 2 is greater than the biggest SD
#(27.2). This condition is satisfied.

#Errors are normal: The Q-Q resids plot looks slightly concerning, there are quite a few
#points on each tail that suggest non-normality, maybe a transformation of the data is needed?

#errors are independent: The errors are probably independent, one could try to get
#less/more blank tiles based on a previous game, but I assume its different players.
favstats(WinMargin~factor(Blanks), data = WordsWithFriends)
```

```
##   factor(Blanks) min Q1 median Q3 max     mean       sd   n missing
## 1        0blanks -77 35   48.0 64 113 47.59459 25.70298 111       0
## 2         1blank  -2 36   51.0 64 172 53.41014 25.70078 217       0
## 3        2blanks   3 44   56.5 75 152 60.50000 27.16407 116       0
```

```
#5.52b:
#Ho: mean0=mean1=mean2 or alpha0=alpha1=alpha2=0, for how # of blanks affects win margin
#Ha: At least one alpha isn't equal to 0.

methoddd <- lm(WinMargin~factor(Blanks), data = WordsWithFriends)
anova(methoddd)
```

```
## Analysis of Variance Table
##
## Response: WinMargin
##                 Df Sum Sq Mean Sq F value   Pr(>F)
## factor(Blanks)   2   9514  4757.2  6.9884 0.001028 **
## Residuals      441 300202   680.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#with an F=6.9884, df= (2,441) pval = .001028<0.05 we can reject the null hypothesis
#that the # of blanks has no affect on win margin (alpha values 0). We can conclude
#that the # of blanks does have a significant affect on win margin.
```