HANDOUT - Unusual Points (C4.4)

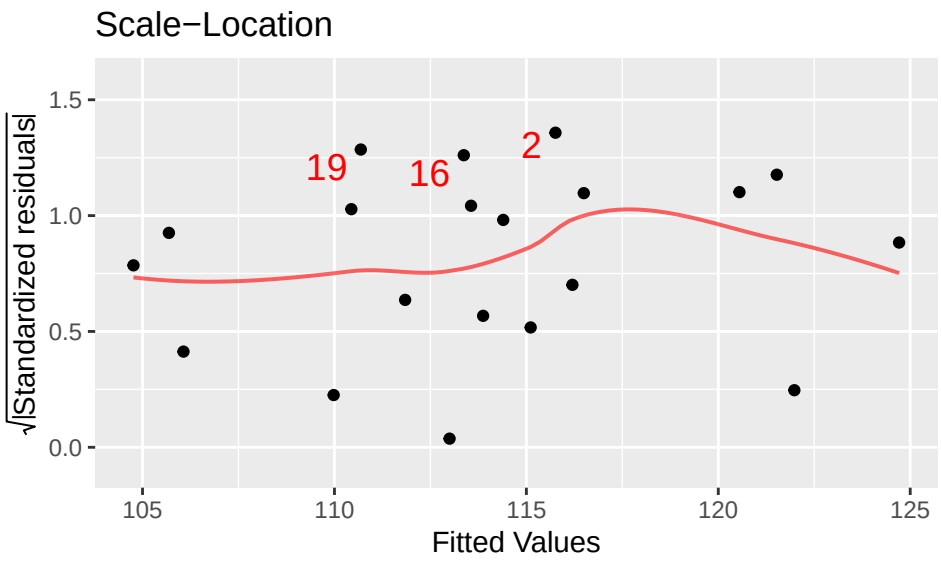
PB.Matheson adapted from S.M. Liao and A.S. Wagaman

R has built-in functions to generate the leverage values (called the *hat* values), the standardized and studentized residuals, and Cook's D values. We will use mplot but new values for the option (which= ?).

We will go back to the model predicting BP and use model fm4 (with Weight, Age and BSA as predictors).

Using option `which = 3` in mplot (shown below) produces a *scale-location* plot which plots the square root of the absolute value of the standardized residuals vs. the fitted values. It looks similar to the our which=1 plot and we can look at linearity and equal variances here as well BUT we are using it here to find unusual points.

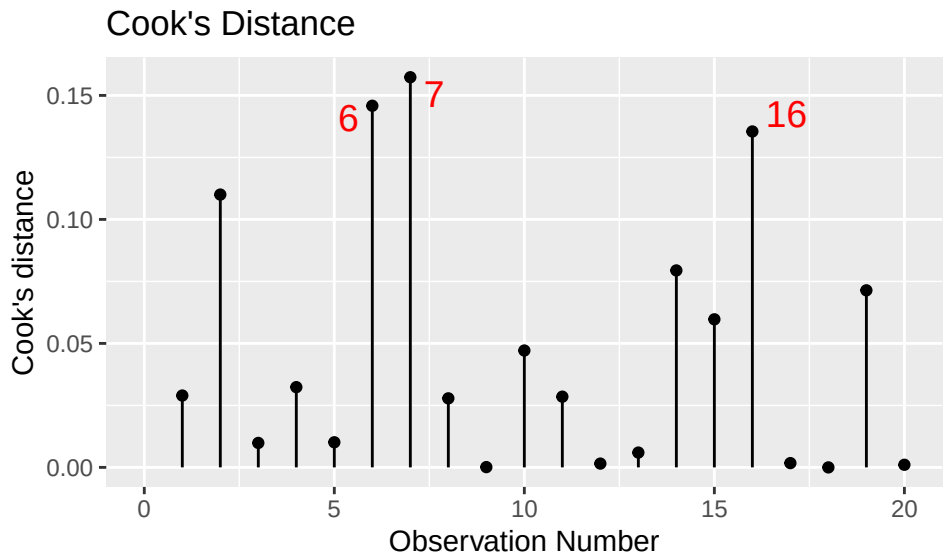
```
mplot(fm4, which = 3)
```



For standardized residuals we want to see if any values are beyond 2 (either below -2 or above +2) to identify potential outliers. For extreme outliers we are looking to find values beyond 3 (either below -3 or above +3). We have to adjust a bit here because the y scale is a square root. Since the square root of 2 is 1.414 (roughly) and square root of 3 is roughly 1.73, these are the values we have to consider in this plot as the cutoffs for potential outliers and extreme outliers, respectively.

The option `which = 4` (shown below) generates a plot showing the Cook's distance values by observation number.

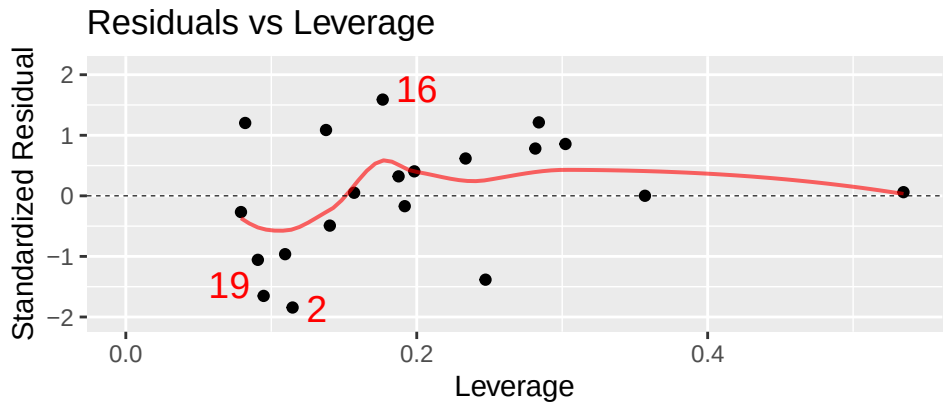
```
mplot(fm4, which = 4)
```



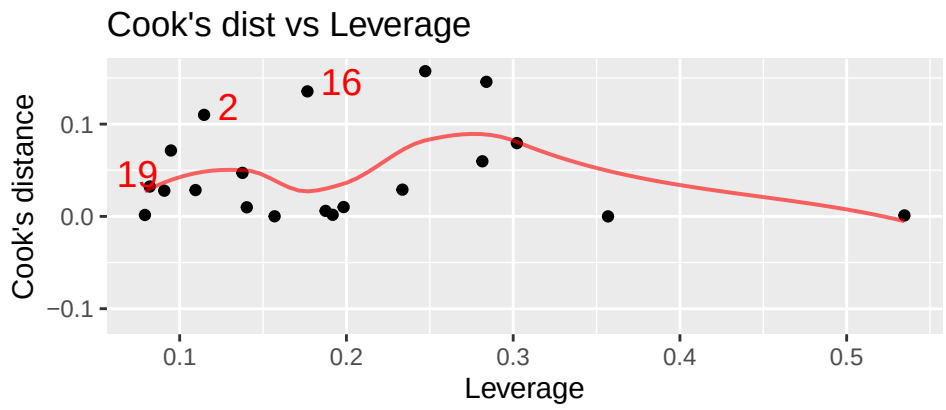
We can examine this plot to see if there are any distances >0.5 which would be a moderately influential point and >1.0 which would be a very influential point. Don't be fooled by spikes, those spikes have to be beyond our rule of thumb cutoffs to require further review.

The next two useful plots `which = 5` shows residuals versus leverage, while `which = 6` shows the Cook's distance values by leverage. While there are some data points that are in keeping with the rest they don't rise to a level of concern because they don't reach the cutoffs of concern.

```
mplot(fm4, which = 5)
```



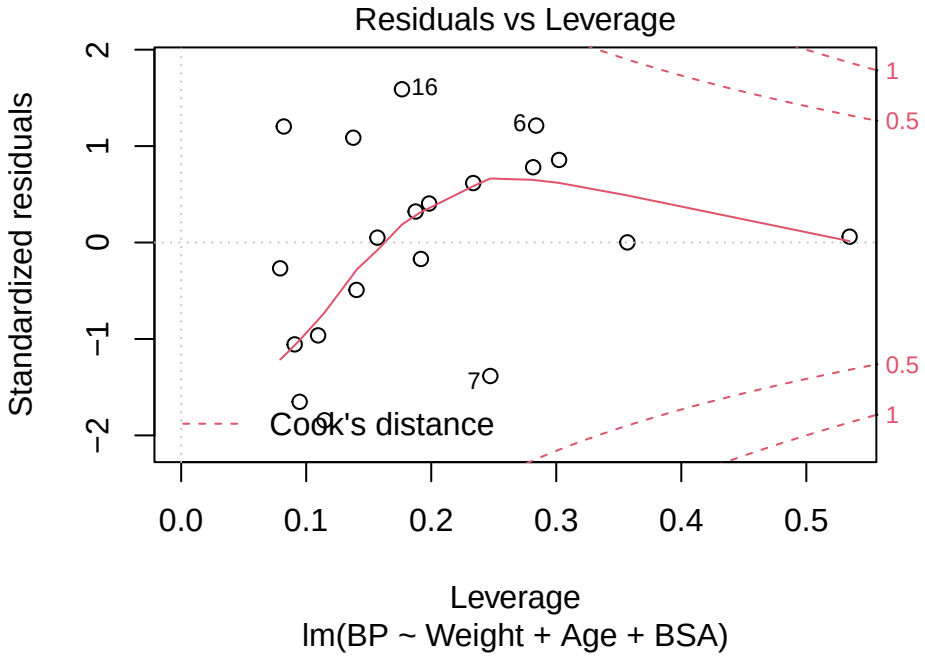

```
mplot(fm4, which = 6)
```



Remember for leverage we have to create the cutoffs based on sample size (for SLR) or sample size and number of predictors (for MLR). The BP data has 20 observations and 3 predictors (age, weight and BSA). Thus we use a cutoff of 2(k+1)/n for moderate leverage which is [2(4)]/20 or 0.4.
For very high leverage we use 3(k+1)/n which is 3(3+1)/20 = 0.6
Look to see if any values have a leverage over 0.4? over 0.6?

A really cool and perhaps even better plots is to use the `plot()` function (instead of `mplot()`) with option `which = 5`. It shows show standardized residuals versus leverage, PLUS additional **contour lines** for the cutoffs of Cook's distance of 0.5 and 1. So you can in fact use this plot to obtain all three measures (Cook's distance, leverage, and standardized residuals) in one plot!

```
plot(fm4, which = 5)
```



In this plot you can see any standardized residual values > |2| using the y axis, leverages over your cutoff using the x axis, and Cook's distance over .5 or 1.0 using the contour lines.

So what? If you have unusual points that are influencing the regression equation you need to examine this

observation more fully.

1) is this data a typo? look at the original data and see if you can fix it.

2) is this data a real value? We've happened to pull in the 7foot 4inch tall NBA player. If so, we probably want to take that observation out, report what we did and why we did it and rerun the analysis. We can't let one extreme value can't drive the entire analysis and change the conclusions.

MORE DETAIL if you want to dig further

How to extract those values/measures from a model

We can easily extract fitted values, residuals (original/standardized/studentized), leverage (hat values), and Cook's distance values via the functions below (where model would equal fm4 here):

- fitted values: `fitted(model)`
- residuals: `residuals(model)`
- standardized residuals: `rstandard(model)`
- studentized residuals: `rstudent(model)`
- leverage: `hatvalues(model)`
- Cook's distance: `cooks.distance(model)`

You could also use the augment function to store them in a new file. Notice we had to add the studentized residuals in the mutate command below to get R to save them since they are not automatically included with augment.

```
nycaug <- augment (fm4) %>%
   mutate(.stu.resid = rstudent(fm4))
names (nycaug)
```

```
##  [1] "BP"         "Weight"     "Age"        "BSA"        ".fitted"
##  [6] ".resid"     ".hat"       ".sigma"     ".cooksd"    ".std.resid"
## [11] ".stu.resid"
```

Now that we have a file (nycaug) with all the values regarding unusualness, we can run summaries on the desired statistics to see what kinds of distributions we have. Remember your cutoff values for each kind of unusual point measure and see if you have values beyond it (>2.0 for std or stu.resid, >0.5 for Cooks Di, >0.4 for leverage).

```
round(favstats(~ .cooksd, data = nycaug), 3) #rounds off to 3 decimals
```

```
##  min    Q1 median    Q3   max  mean    sd  n missing
##    0 0.005  0.029 0.073 0.157 0.048 0.052 20       0
```

```
round(favstats(~ .hat, data = nycaug), 3)
```

```
##    min    Q1 median    Q3   max mean    sd  n missing
##  0.079 0.113  0.182 0.256 0.535  0.2 0.113 20       0
```

```
round(favstats(~ .std.resid, data = nycaug), 3)
```

```
##     min     Q1 median  Q3   max  mean    sd  n missing
##  -1.843 -0.609  0.056 0.8 1.589 0.018 1.001 20       0
```

```
round(favstats(~ .stu.resid, data = nycaug), 3)
```

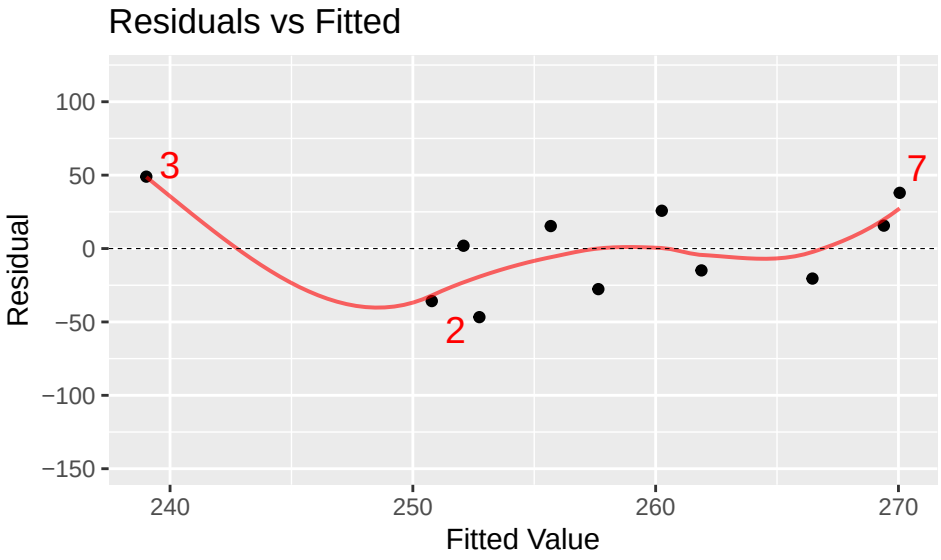
```
##     min   Q1 median   Q3   max  mean    sd  n missing
##  -2.011 -0.6  0.054 0.79 1.677 0.007 1.039 20       0
```

Practice with an example that has more juice!

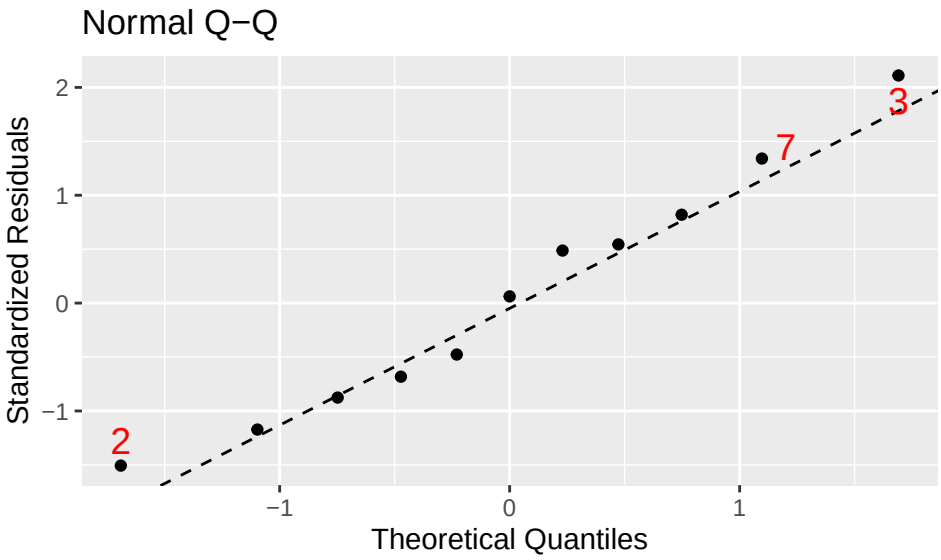
Because this example doesn't have very many unusual points, lets look at another example to see if you can find outliers and influential points in Math Enrollments. This model is predict Spring enrollments from Fall enrollments at Kenyon College in their Mathematics department. Don't forget that your cutoffs for leverage will change!

```
data(MathEnrollment)
model <- lm(Spring ~ Fall, data = MathEnrollment)
mplot(model, which = 1)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

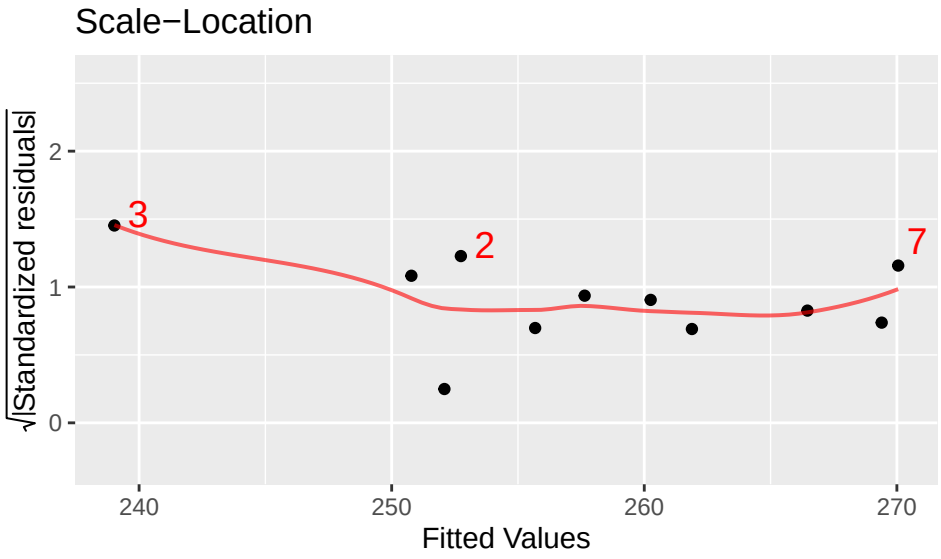



```
mplot(model, which = 2)
```

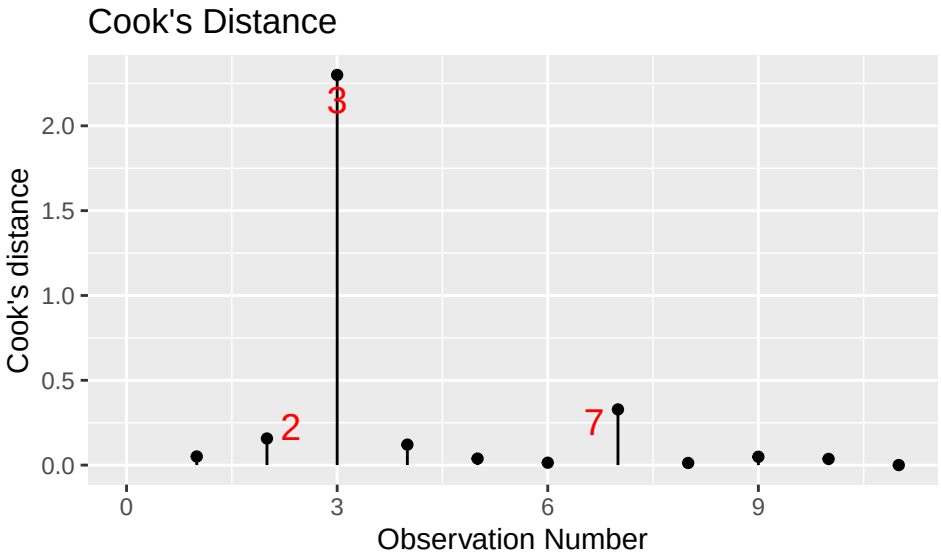


```
mplot(model, which = 3)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

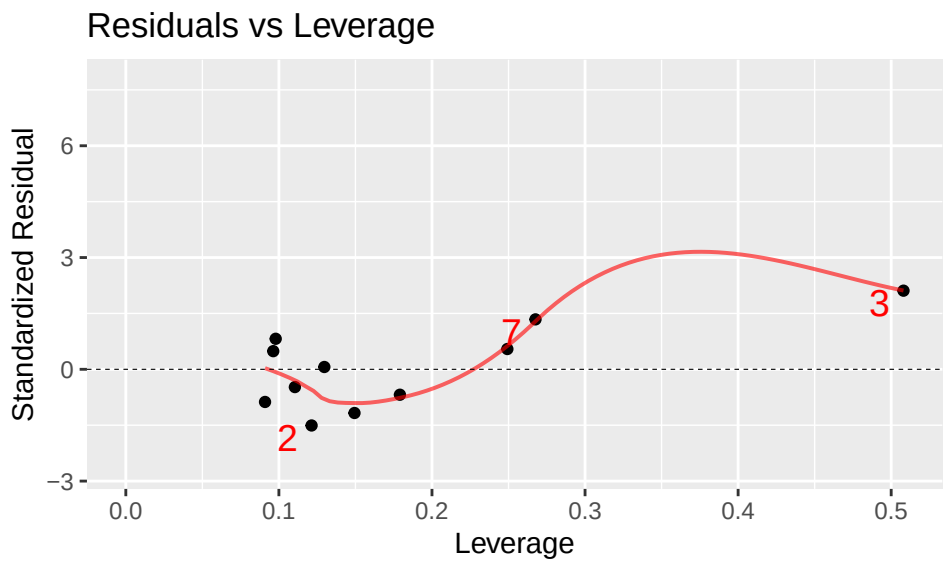



```
mplot(model, which = 4)
```



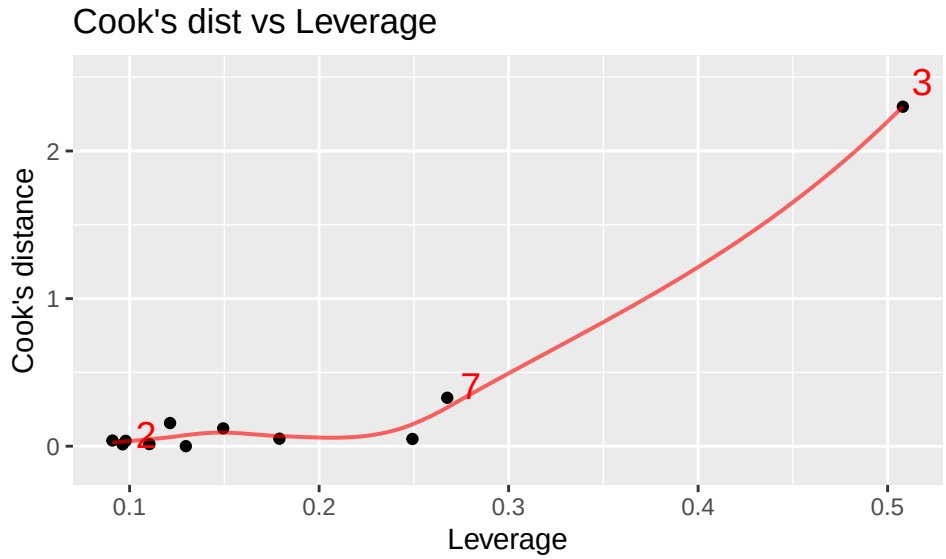

```
mplot(model, which = 5)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

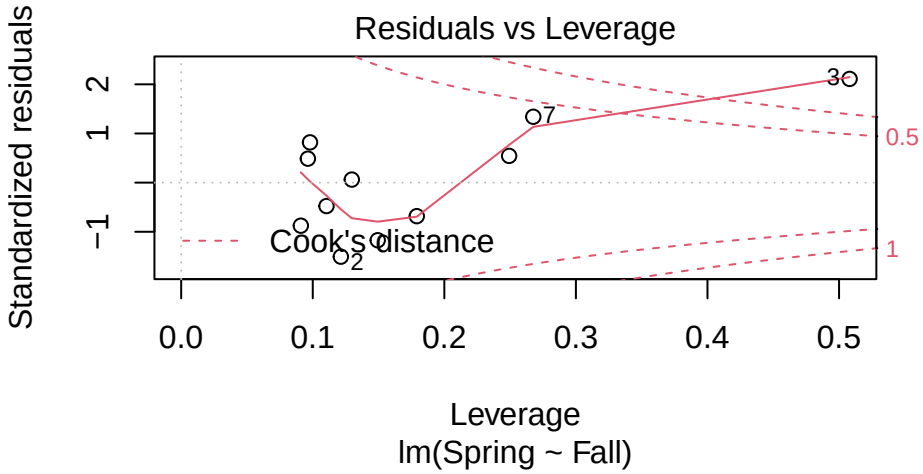



```
mplot(model, which = 6)
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
plot(model, which = 5)
```




```
Mathaug <- augment(model) %>% mutate(.stu.resid = rstudent(model)) #useful!
names(Mathaug)
```

```
## [1] "Spring"     "Fall"       ".fitted"    ".resid"     ".hat"
## [6] ".sigma"     ".cooksd"    ".std.resid" ".stu.resid"
```

Look through this example, and use the plots to help you determine what points would be considered unusual based on the four statistics from this section. Can you see how the different plots can help you spot these quickly?