Lab 3: Homework #3 Preparation

Wednesday Bushong

2/22/2017

Preliminaries

In this lab we'll be learning about outlier removal. We'll visually inspect our data to see if anything seems out of place, and then we'll learn about some more quantitative measures of outlier-ness. Finally, we'll either remove or Winsorize outliers and see how it affects our regression results.

```
library(foreign)
library(car) # for Boxplot() function
library(ggplot2) # for nice plotting functions
theme_set(theme_classic())
source("../Lab2/showmelm_WB.R")

d <- read.spss("Lab3Data.sav", to.data.frame = TRUE)
d <- d[complete.cases(d), ] # remove rows with NAs
d$case.number <- rownames(d)
```

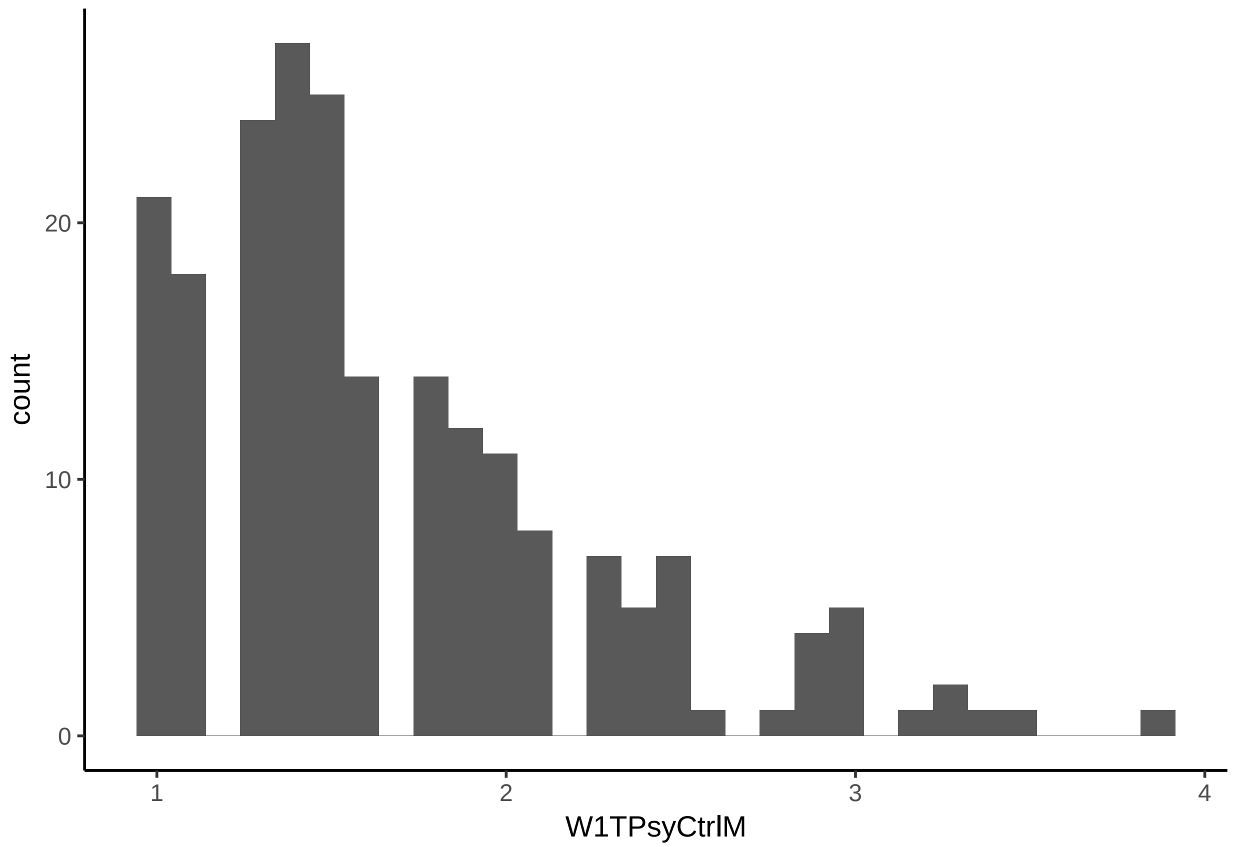
Visually Inspecting Potential Outliers

Univariate

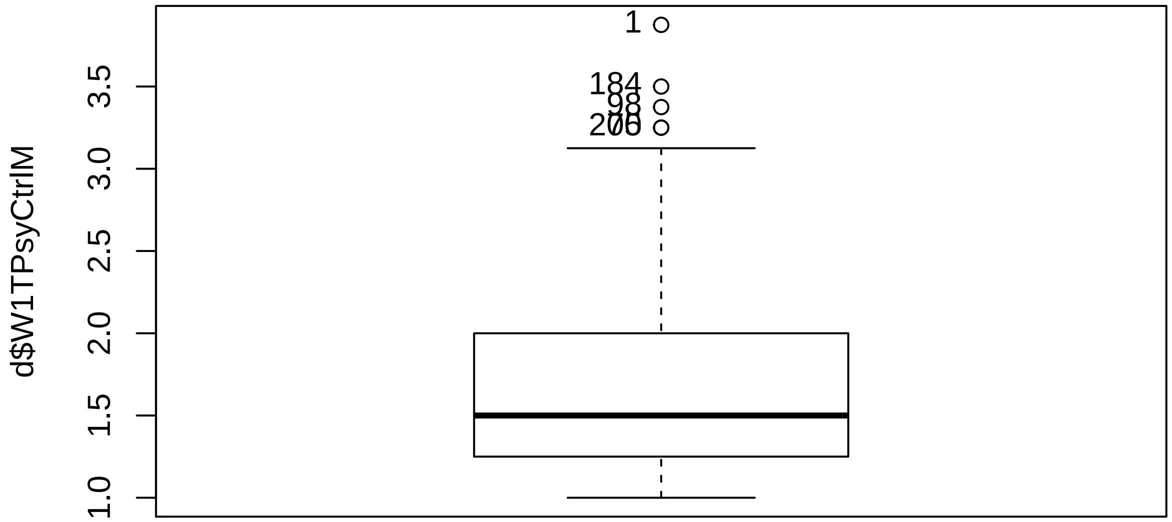
To identify univariate outliers, we can use histograms and boxplots to look for values that seem out of place:

```
p.hist <- ggplot(d, aes(x = W1TPsyCtrlM)) +
  geom_histogram()
p.hist
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



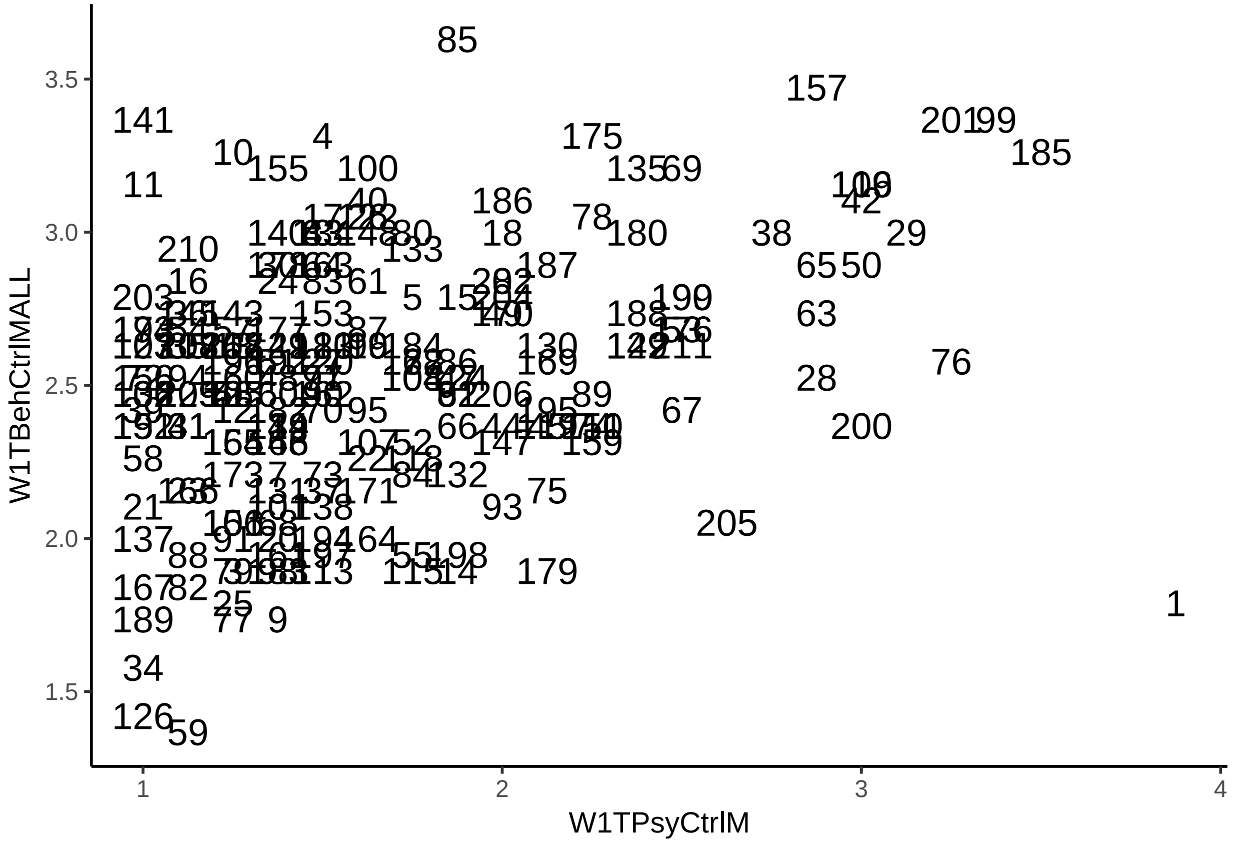
```
p.box <- Boxplot(d$W1TPsyCtrlM)
```



Bivariate

To visually inspect whether there are bivariate outliers, we'll make a scatterplot. We can also label the points by their ID number so we can easily keep track of which cases seem to be outliers:

```r
p.scatter <- ggplot(d, aes(x = W1TPsyCtrlM, y = W1TBehCtrlMALL, label = case.number)) +
  geom_text(size = 5)
p.scatter
```



Regression Diagnostics for Outlier Identification

First, we will fit a regression:

```
m <- lm(W1TNegIntM ~ W1TPsyCtrlM + W1TBehCtrlMALL, d)
```

The car package in R has a function ls.diag() which computes a number of regression diagnostics from a model object:

```
m.diag <- ls.diag(m)
names(m.diag) # what's in this object?
```

```
##  [1] "std.dev"      "hat"          "std.res"      "stud.res"
##  [5] "cooks"        "dfits"        "correlation"  "std.err"
##  [9] "cov.scaled"   "cov.unscaled"
```

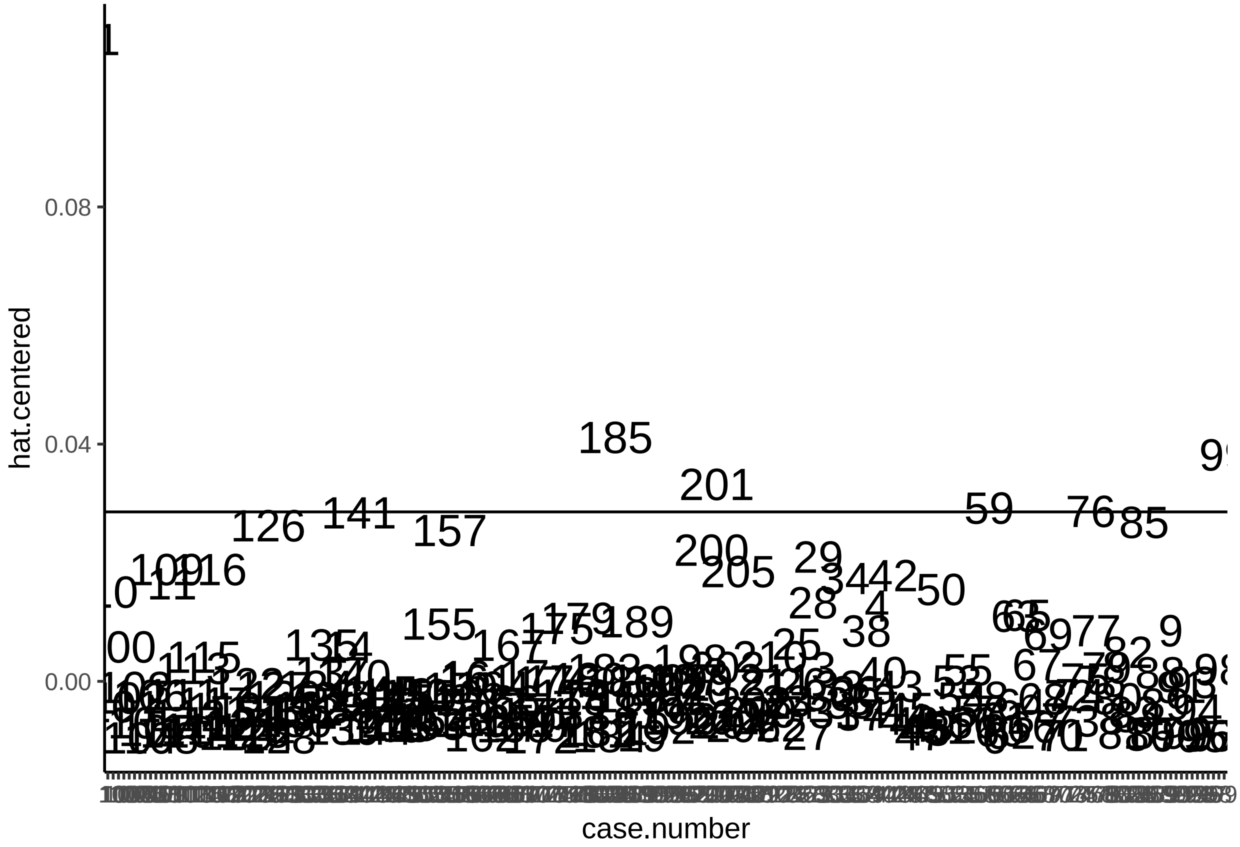
Leverage

Leverage is a measure of how "extreme" values of the combination of independent variables are for each case. We can visually inspect the leverage values of each of the cases and use some rule-of-thumb cutoffs to decide whether we need to remove any of them:

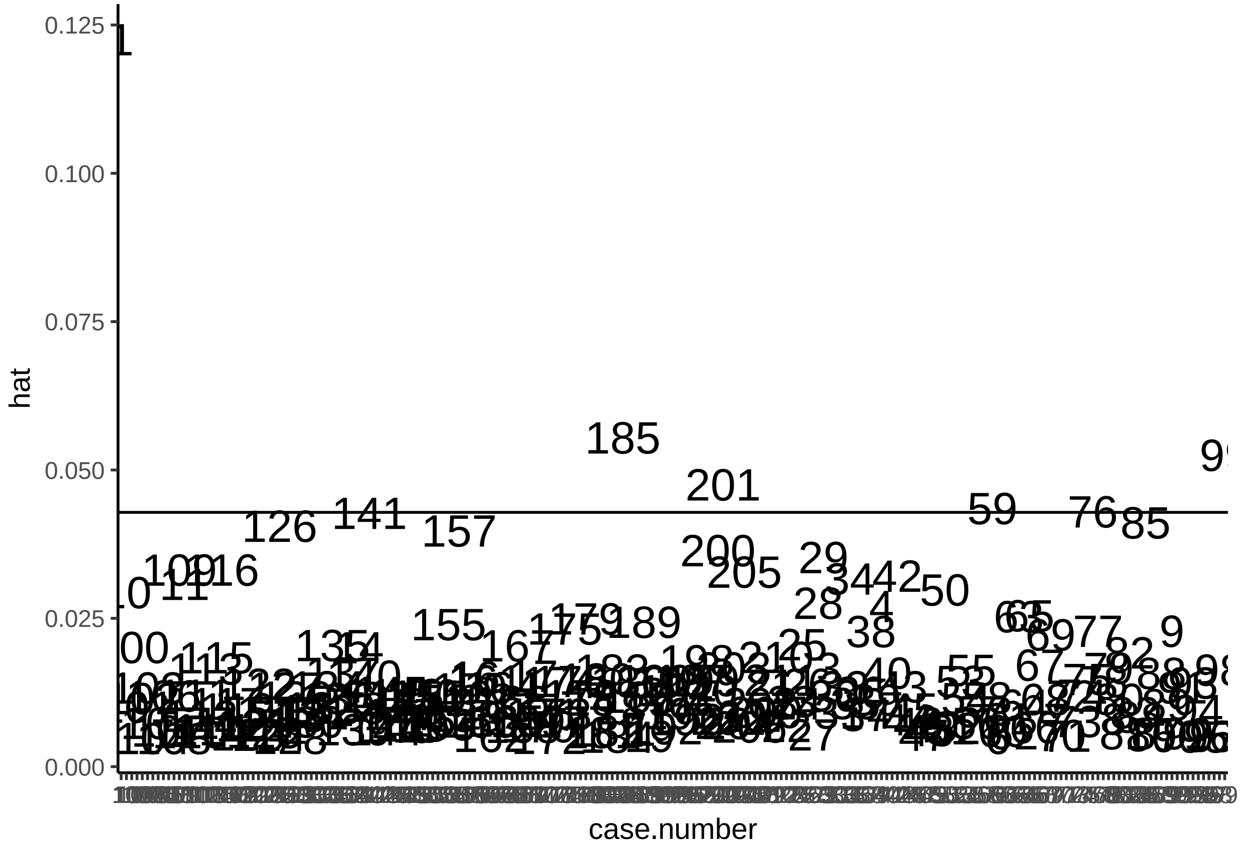
```
d$hat <- m.diag$hat
d$hat.centered <- m.diag$hat - mean(m.diag$hat)

## Two rules-of-thumb leverage cutoffs:
## 2 * (# independent variables) / (# observations) [3 for "small" samples]
## 2 * (mean hat value) [3 for "small" samples]
cutoff.centeredvals <- (3 * 2) / nrow(d)
cutoff.uncenteredvals <- 3 * mean(d$hat)

levplot1 <- ggplot(d, aes(x = case.number, y = hat.centered, label = case.number)) +
  geom_text(size = 6) +
  geom_hline(yintercept = cutoff.centeredvals)
levplot1
```



```
levplot2 <- ggplot(d, aes(x = case.number, label = case.number, y = hat)) +
  geom_text(size = 6) +
  geom_hline(yintercept = cutoff.uncenteredvals)
levplot2
```



Discrepancy

Discrepancy is the distance between predicted and observed values on the dependent variable using externally studentized residuals. The studentized residuals can be treated as t-values and we can perform a t-test on them to see whether the most extreme values are significantly different from predicted.

```
d$stud.res <- m.diag$stud.res

# significance of _largest_ studentized residual can be tested with a t test,
# where t = stud.res value, df = n - k - 1, and alpha = .05 / n
# (k = number of IVs; n = number of cases)
# since we already have the t-value, we will use the dt() function, which gives us the probability of a
highest.t.val <- max(d$stud.res)
p.val <- dt(highest.t.val, df = nrow(d) - 2 - 1)
```

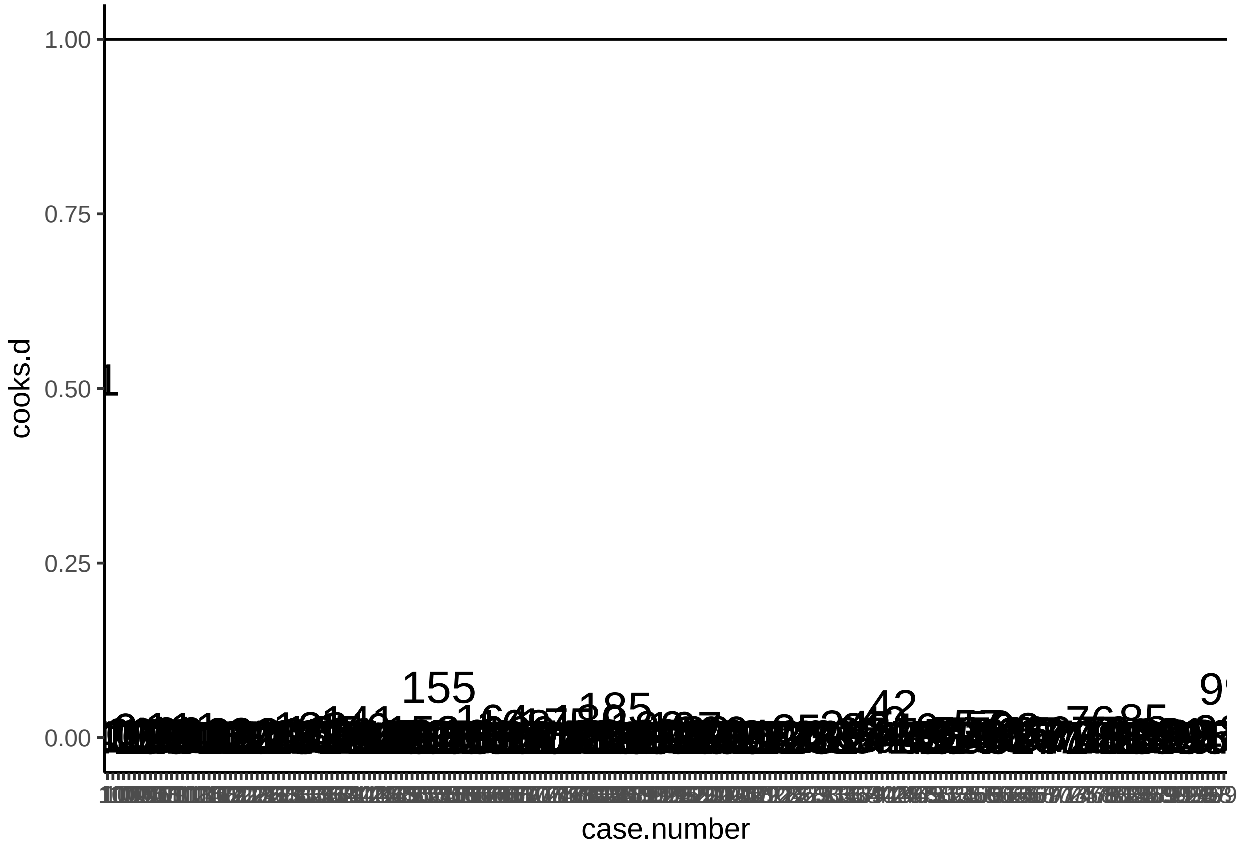
Influence on Regression Estimates

Global Influence: Cook's D

Cook's D is a measure of how much an individual case influences predicted Y-values. A general rule-of-thumb cutoff is 1.

```
d$cooks.d <- m.diag$cooks
```

```r
p.cooks.d <- ggplot(d, aes(x = case.number, y = cooks.d, label = case.number)) +
  geom_text(size = 6) +
  geom_hline(yintercept = 1)
p.cooks.d
```



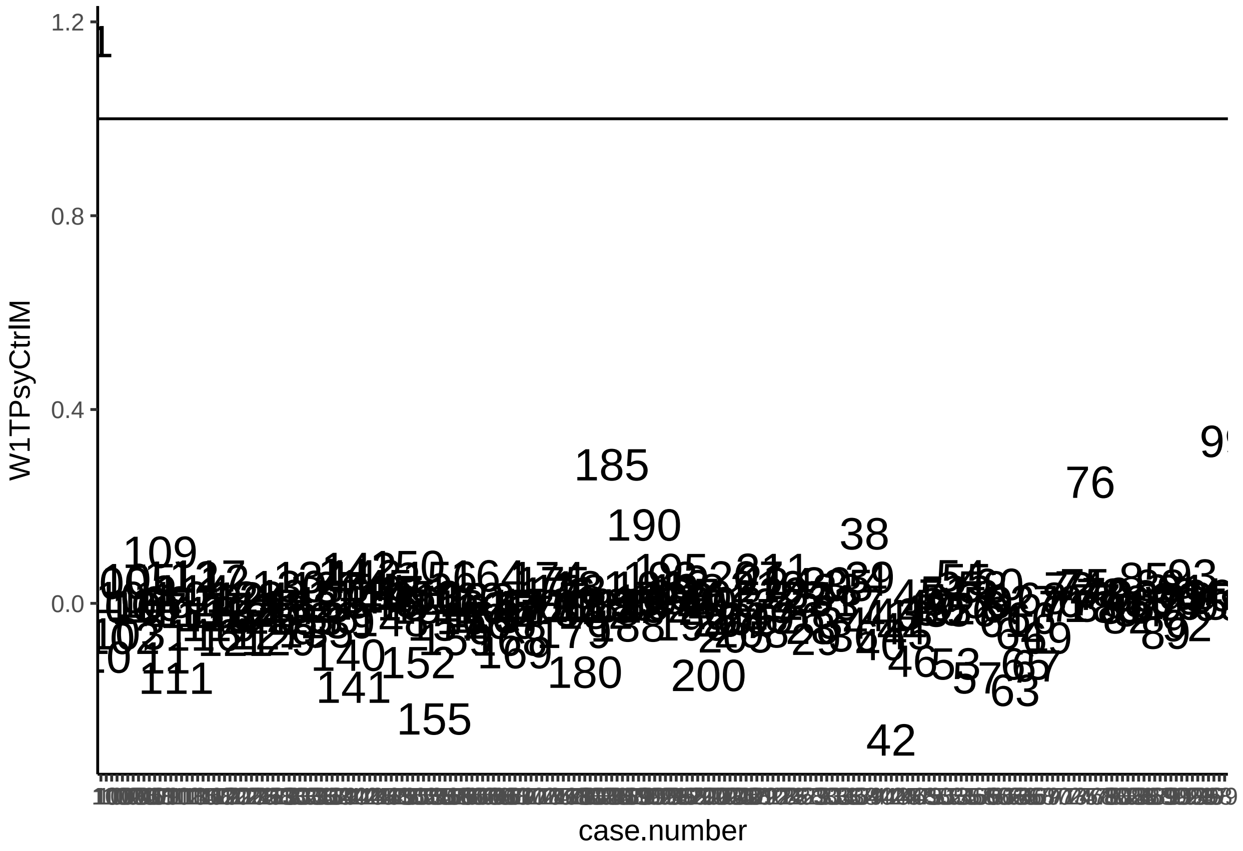
Specific Influence

DF Beta is a measure of how an indvidual case affects the coefficient estimate (beta) for a particular predictor. A rule of thumb cutoff for DF Betas is 1.

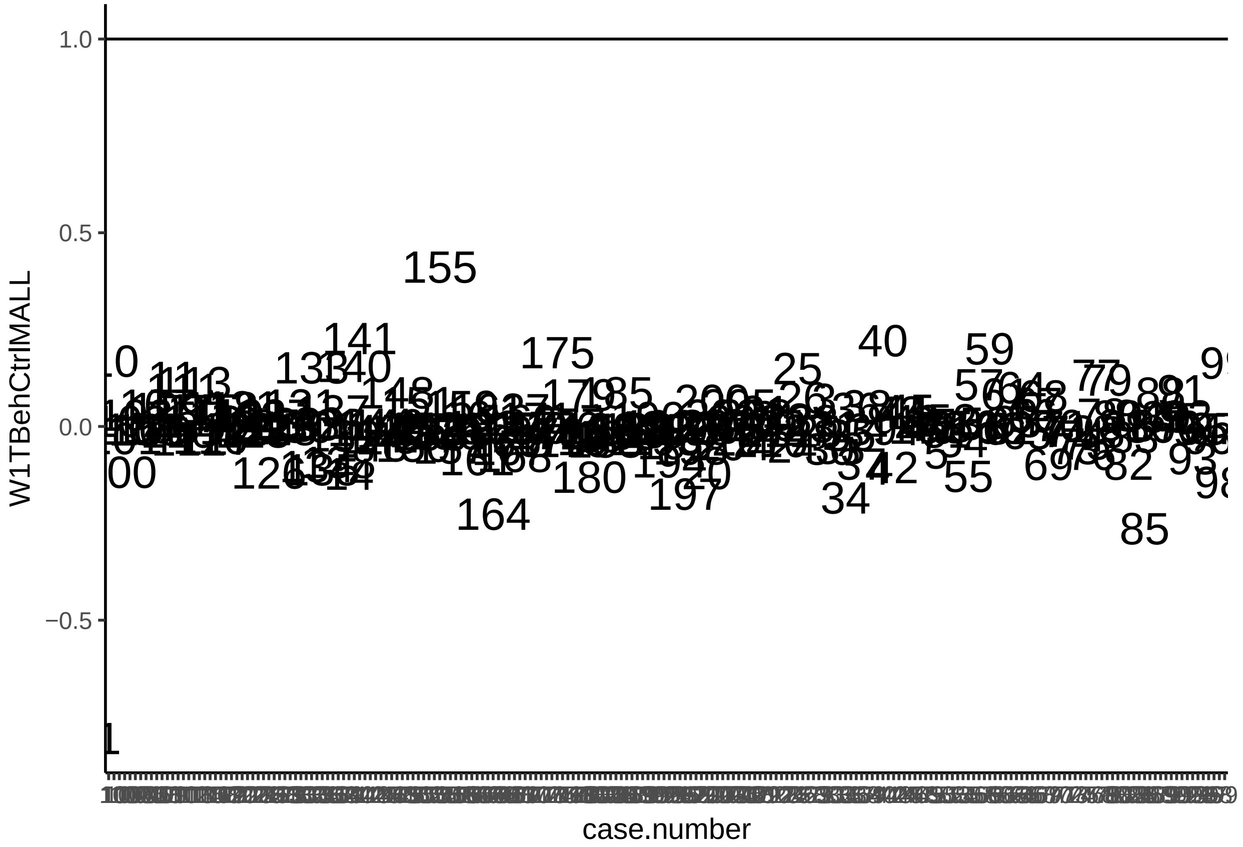
```r
dfbeta.vals <- as.data.frame(dfbetas(m))
dfbeta.vals$case.number <- rownames(dfbeta.vals)
head(dfbeta.vals, 10)
```

```
##     (Intercept)  W1TPsyCtrlM W1TBehCtrlMALL case.number
## 1   0.310069168  1.160613041   -0.803763415           1
## 2  -0.020571932  0.037528428   -0.005335561           2
## 3  -0.087879223  0.012819946    0.071531380           3
## 4   0.079920784  0.049205412   -0.108190559           4
## 5   0.048033472  0.008575664   -0.066702327           5
## 6  -0.001422495 -0.005493425   -0.004982037           6
## 7   0.016797806 -0.004955420   -0.011344758           7
## 8   0.013607762 -0.007014940   -0.013159420           8
## 9  -0.029050555 -0.001546158    0.026751836           9
## 10 -0.110304266 -0.109715284    0.170239295          10
```

```
p.betas.1 <- ggplot(dfbeta.vals, aes(x = case.number, label = case.number, y = W1TPsyCtrlM)) +
  geom_text(size = 6) +
  geom_hline(yintercept = 1)
p.betas.1
```



```
p.betas.2 <- ggplot(dfbeta.vals, aes(x = case.number, label = case.number, y = W1TBehCtrlMALL)) +
  geom_text(size = 6) +
  geom_hline(yintercept = 1)
p.betas.2
```



Removing Outliers

One way of dealing with outliers is to outright remove them. Since we have seen throughout this lab that the FamilyID 1 observation is a clear outlier, let's remove it by subsetting:

```
d.outlier.removed <- subset(d, FamilyID != 1)
```

Winsorizing Outliers

Alternatively, we might want to instead of removing a case entirely, reassigning it a more "reasonable" value. There are many different ways to do this, but here we will reassign the outlier to be the 5th or 95th percentile (depending on whether the outlier is high or low on the value).

```
d.winsorized <- d
W1TNegIntM.quantile95 <- quantile(d.winsorized$W1TNegIntM, probs = c(0.95))
W1TPsyCtrlM.quantile95 <- quantile(d.winsorized$W1TPsyCtrlM, probs = c(0.95))

d.winsorized$W1TNegIntM[d.winsorized$FamilyID == "1"] <- W1TNegIntM.quantile95
d.winsorized$W1TPsyCtrlM[d.winsorized$FamilyID == "1"] <- W1TPsyCtrlM.quantile95
```

Checking how removing/Winsorizing a case affects regression results

Now, let's see how regression results are affected by removing or Winsorizing our outlier.

```
m.removed <- lm(W1TNegIntM ~ W1TPsyCtrlM + W1TBehCtrlMALL, d.outlier.removed)
m.winsorized <- lm(W1TNegIntM ~ W1TPsyCtrlM + W1TBehCtrlMALL, d.winsorized)
summary(m)
```

```
## 
## Call:
## lm(formula = W1TNegIntM ~ W1TPsyCtrlM + W1TBehCtrlMALL, data = d)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -1.4819 -0.4802 -0.1279  0.4396  2.0491 
## 
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)    
## (Intercept)     1.25738    0.28991   4.337 2.25e-05 ***
## W1TPsyCtrlM     0.86634    0.08306  10.430  < 2e-16 ***
## W1TBehCtrlMALL -0.27767    0.11808  -2.351   0.0196 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.6592 on 207 degrees of freedom
## Multiple R-squared:  0.3467, Adjusted R-squared:  0.3404 
## F-statistic: 54.93 on 2 and 207 DF,  p-value: < 2.2e-16
```

```
summary(m.removed)
```

```
## 
## Call:
## lm(formula = W1TNegIntM ~ W1TPsyCtrlM + W1TBehCtrlMALL, data = d.outlier.removed)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -1.4486 -0.4788 -0.1363  0.4093  1.8637 
## 
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)    
## (Intercept)      1.1697     0.2839   4.119  5.5e-05 ***
## W1TPsyCtrlM      0.7723     0.0856   9.022  < 2e-16 ***
## W1TBehCtrlMALL  -0.1851     0.1183  -1.564    0.119    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.643 on 206 degrees of freedom
## Multiple R-squared:  0.2921, Adjusted R-squared:  0.2852 
## F-statistic:  42.5 on 2 and 206 DF,  p-value: 3.518e-16
```

```
summary(m.winsorized)
```

```
## 
## Call:
## lm(formula = W1TNegIntM ~ W1TPsyCtrlM + W1TBehCtrlMALL, data = d.winsorized)
## 
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.4504 -0.4816 -0.1233  0.4012  1.8686
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     1.17983    0.28171   4.188 4.16e-05 ***
## W1TPsyCtrlM     0.77839    0.08345   9.327  < 2e-16 ***
## W1TBehCtrlMALL -0.19270    0.11587  -1.663   0.0978 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6416 on 207 degrees of freedom
## Multiple R-squared:  0.3022, Adjusted R-squared:  0.2954
## F-statistic: 44.82 on 2 and 207 DF,  p-value: < 2.2e-16
```

As we can see, our regression results change quite drastically if we remove or Winsorize just one single case.