Robust regression methods in R

James Uanhoro

March 6, 2018

Check-in

1. Have script?
2. Have data?

Robust regression models in R

What we will cover:

1. Checking assumptions of regression analysis
2. Attempts at managing outliers
3. Attempts at managing heteroskedasticity

My assumptions

Know how to interpret regression coefficients?

Know how to work with data in R - import, data frames, . . . ?

Know what packages are?

Work in a project in RStudio

`File -> New Project`

Projects can help you organize your work when using RStudio.

Create the project in a directory you can find.

You can return to the project by opening RStudio and `File -> Open Project`

Packages to install before continuing

Only need to do this once

```r
install.packages("car") # Plots for diagnostics
install.packages("robustbase") # Robust regression
install.packages("arm") # Fancy regression output
install.packages("sandwich") # Robust variance-covariance
install.packages("lmtest") # Testing regression models
install.packages("hcci") # Wild bootstrapping
install.packages("hett") # t-regression
install.packages("stargazer") # Nice regression tables
```

First dataset for today

```r
edu <- read.csv("education.csv")
```

Region: 1 = Northeastern, 2 = North central, 3 = Southern, 4 = Western

Urban: Number of residents per thousand residing in urban areas in 1970

Income: Per capita personal income in 1973

Under18: Number of residents per thousand under 18 years of age in 1974

Educ: Per capita expenditure on public education in a state, projected for 1975

Summary statistics

```r
summary(edu)
```

```
##      State        Region          Urban           Income
##   AK     : 1   Min.   :1.00   Min.   :322.0   Min.   :344
##   AL     : 1   1st Qu.:2.00   1st Qu.:546.8   1st Qu.:413
##   AR     : 1   Median :3.00   Median :662.5   Median :470
##   AZ     : 1   Mean   :2.66   Mean   :657.8   Mean   :467
##   CA     : 1   3rd Qu.:3.75   3rd Qu.:782.2   3rd Qu.:505
##   CO     : 1   Max.   :4.00   Max.   :909.0   Max.   :588
##   (Other):44
##     Under18           Educ
##   Min.   :287.0   Min.   :208.0
##   1st Qu.:310.8   1st Qu.:234.2
##   Median :324.5   Median :269.5
##   Mean   :325.7   Mean   :284.6
##   3rd Qu.:333.0   3rd Qu.:316.8
##   Max.   :386.0   Max.   :546.0
##
```

Goal

Model `Educ` using Urban, Income and Under18

I usually begin with plots, but most people want to jump right into the model so:

```r
library(arm)
library(car)
(fit.1 <- lm(Educ ~ Urban + Income + Under18, edu))
```

```
##
## Call:
## lm(formula = Educ ~ Urban + Income + Under18, data = edu
##
## Coefficients:
## (Intercept)        Urban       Income      Under18
##  -5.566e+02   -4.269e-03    7.239e-02    1.552e+00
```

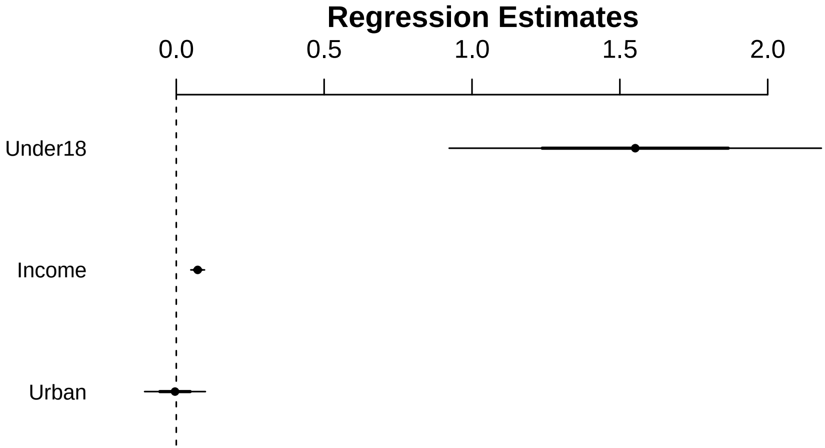
Model summary

```
summary(fit.1)
```

```
##
## Call:
## lm(formula = Educ ~ Urban + Income + Under18, data = edu
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -84.878 -26.878  -3.827  22.246  99.243
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.566e+02  1.232e+02  -4.518 4.34e-05 ***
## Urban       -4.269e-03  5.139e-02  -0.083    0.934
## Income       7.239e-02  1.160e-02   6.239 1.27e-07 ***
## Under18      1.552e+00  3.147e-01   4.932 1.10e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.
```

Coefficients plot

```
coefplot(fit.1) # From arm package
```



A few things about this model

The regression model we have run is technically regression with *Ordinary Least Squares* (OLS) estimation. It has the nice property of being the *best linear unbiased estimator* (BLUE) meaning:

1. Best: It produces the most stable results from one sample to the next among unbiased estimators
2. Unbiased: It recovers population parameters - the truth of the population
3. Linear: It can only capture linear relationships between the outcome and the predictor variables you put into the model

These are the reasons everyone learns about this model.

Some Assumptions

Linearity

Think of personal income. Moving from \$1,000 to \$2,000 each month can make a huge difference in a person's life. Then think a millionare, moving from \$99,000 each month to \$100,000 each month does not impact their lives.

So for certain variables, it makes sense to think of them as impacting other variables **not** in a linear way, but in a multiplicative way. Not unit increases but percentage increases.

Reasonable examples: personal income, country GDP, . . . so it is common to take their logarithm prior to entry into models.

If relationship is not linear and you do not account for this, you are distorting the data.

Exogeneity

Predictor variables are uncorrelated with error term, or else your coefficients are *biased*.

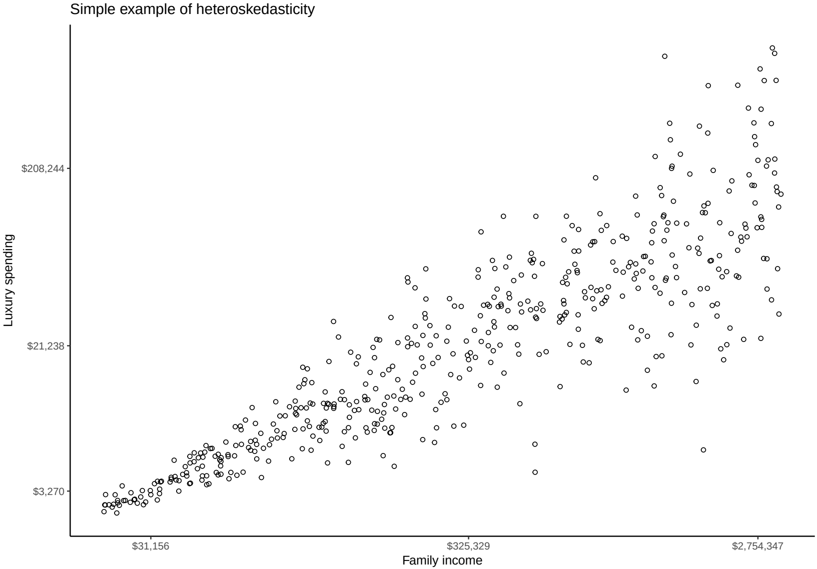
Satisfied if your predictor variable is a grouping variable from a randomized controlled trial.

Practically, this is why we *control* for auxilliary variables in observational studies.

We will not be addressing this today.

Homoskedasticity

The variance of the error is the same for each observation



Homoskedasticity

If violated, you have heteroskedasticity, and cherished p-values are incorrect. Your coefficients are still fine, but no longer efficient - they are less stable from study to study.

We will address this today.

No perfect multicollinearity

Perfect multicollinearity: You can get one of your predictor variables simply by adding two other variables, subtracting, multiplying one variable several times. Simple data mistake. Find offending variable and drop it

Severe multicollinearity: You can predict one of your predictor variables really well from your other predictor variables. Case of overlapping information in predictors.

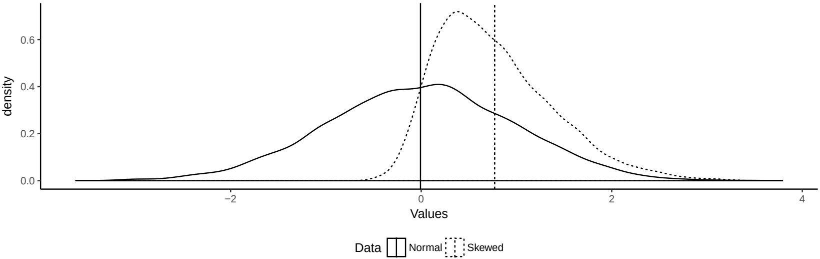
Example: using both exam score at time 1 and time 2 to predict exam score at time 3. When this happens, your coefficients and p-values behave eratically.

We will not address this today. Techniques such as ridge regression, lasso help. They shrink coefficients towards zero. Smaller coefficients mean less variance, and more stability.

Normality of residuals

Technically, you can do without this assumption, especially because of central limit theorem (large sample sizes).

Vertical lines are means. Non−normal residuals may be a sign that your regression coefficients differ from their most likely values



Practically, if the residuals of a few observations are particularly large, they can exert undue influence on your regression equation. And you fail to estimate the most likely values for your coefficients.

We will address this today. The focus of the field of *robust regression*.

Sample is “independent and identically distributed”

Practically, identically distributed means homoskedasticity.

Practically, independent means your sample is a simple random sample.

If violated, there are multilevel (mixed effects models), generalized least squares, and generalized estimating equations approaches.

Diagnostics

Beginning diagnostics

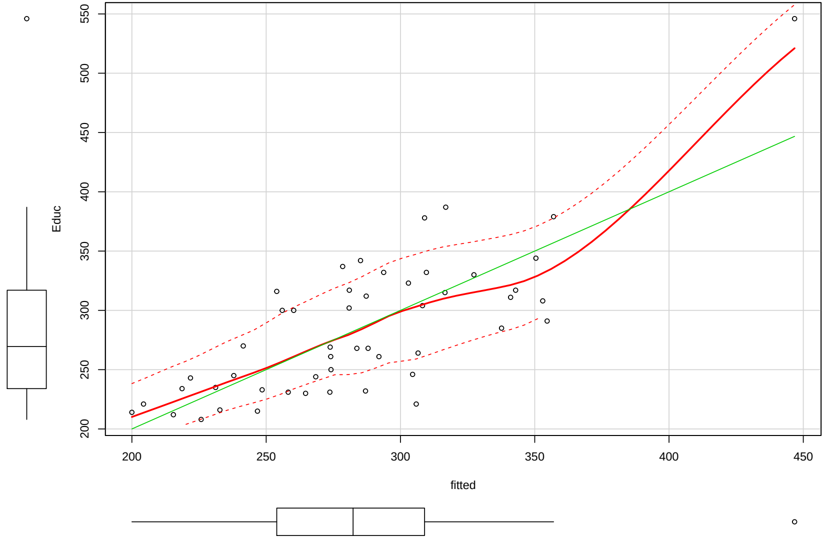
First retrieve the data used to fit the model. Then store model predictions and residuals.

```
model.dat <- fit.1$model
model.dat$residuals <- fit.1$residuals
model.dat$fitted <- fit.1$fitted.values
str(model.dat)
```

```
## 'data.frame':    50 obs. of  6 variables:
##  $ Educ     : int  235 231 270 261 300 317 387 285 300 2
##  $ Urban    : int  508 564 322 846 871 774 856 889 715 7
##  $ Income   : int  3944 4578 4011 5233 4780 5889 5663 57
##  $ Under18  : int  325 323 328 305 303 307 301 310 300 3
##  $ residuals: num  3.83 -42.72 28.53 -30.99 44.01 ...
##  $ fitted   : num  231 274 241 292 256 ...
##  - attr(*, "terms")=Classes 'terms', 'formula'  language
##   .. ..- attr(*, "variables")= language list(Educ, Urban
##   .. ..- attr(*, "factors")= int [1:4, 1:3] 0 1 0 0 0 0
##           attr(*, "dimnames")=List of 2
```

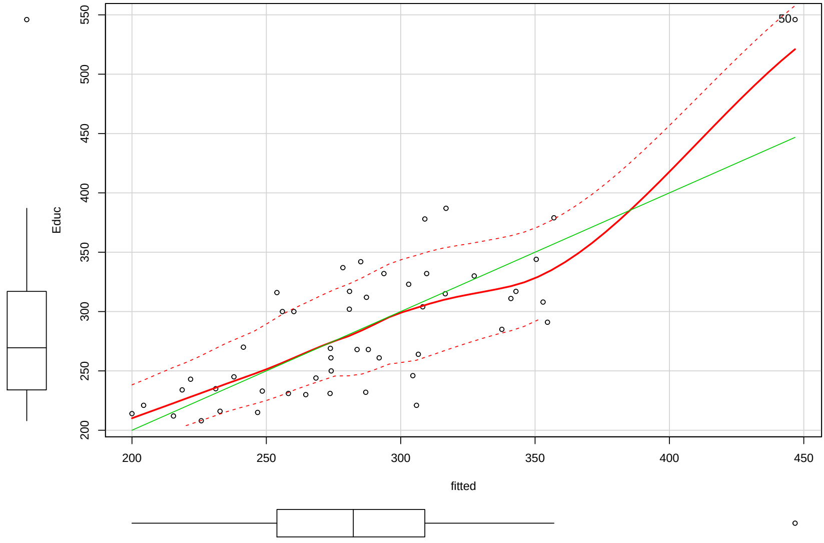
A quick check of linearity

```
scatterplot(Educ ~ fitted, model.dat)
```



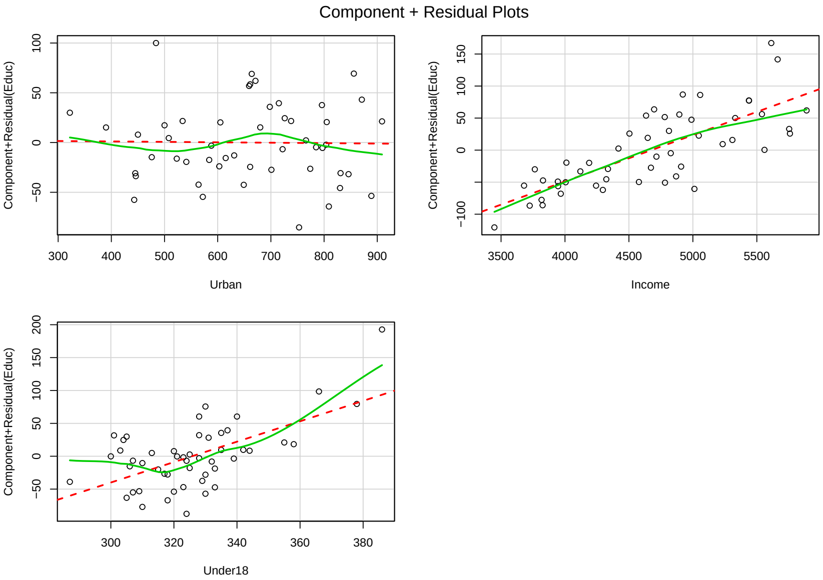
A quick check of linearity

```
scatterplot(Educ ~ fitted, model.dat, id.n = 1)
```



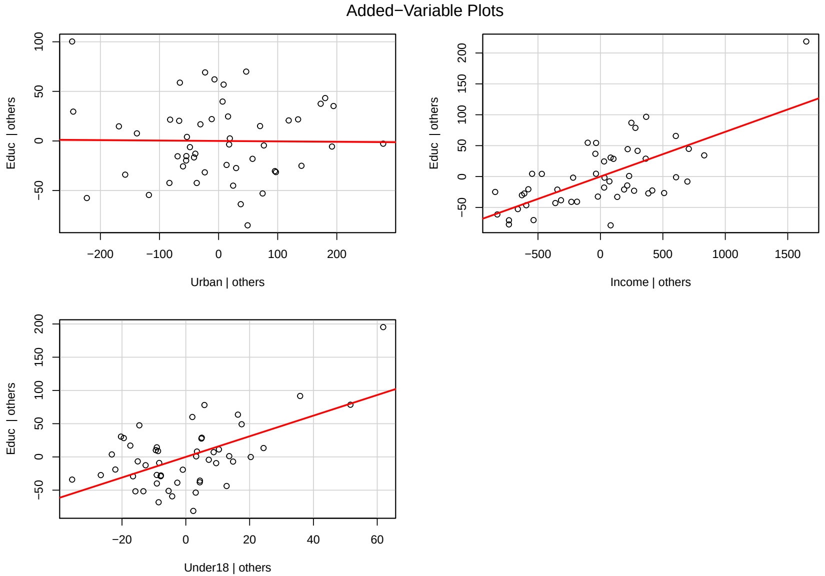
A variable by variable check for linearity

```
crPlots(fit.1)
```



A variable by variable check for general strangeness

```
avPlots(fit.1)
```



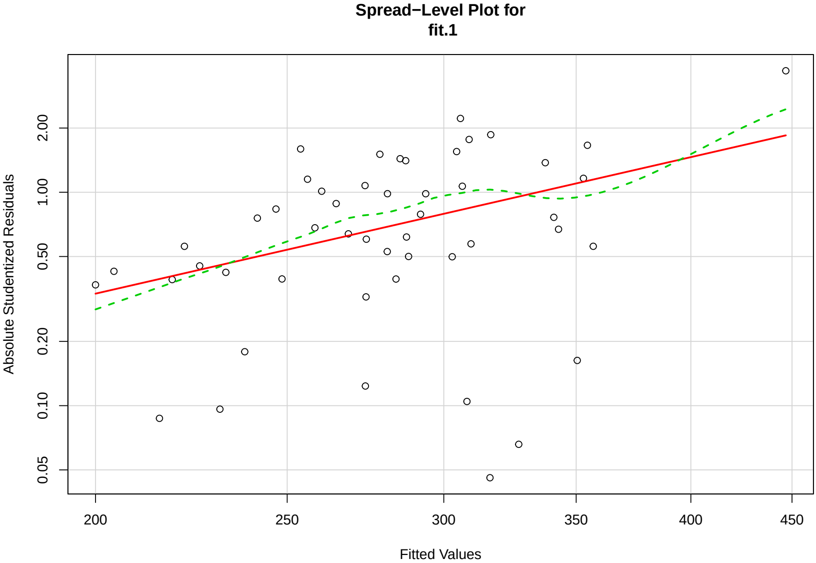
Checking homoskedasticity

```
spreadLevelPlot(fit.1)
```

Studentized residuals are standardized residuals such that if there was homoskedasticity, they would look like a horizontal block that was not related to your fitted values. Do we have a problem here?

Checking homoskedasticity

```
spreadLevelPlot(fit.1)
```



In case we want a statistical test for homoskedasticity

```r
library(lmtest)
bptest(fit.1)
```

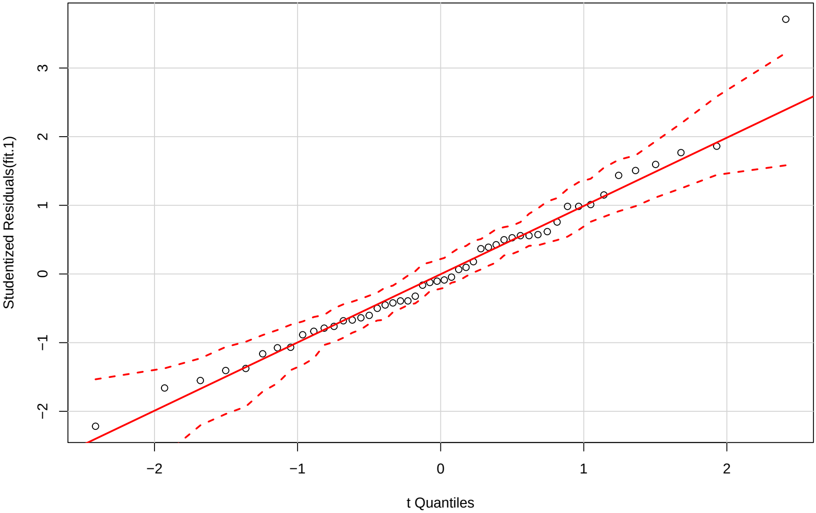
```
##
##  studentized Breusch-Pagan test
##
## data:  fit.1
## BP = 15.59, df = 3, p-value = 0.001376
```

A statistically significant result indicates a violation of homoskedasticity. Additional suggestion that we have a problem.

Can we trust the *p*-values from the regression model?

Checking normality of residuals

```
qqPlot(fit.1)
```



Outlier detection

In a simple linear regression

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i \tag{1}$$

The estimated regression coefficient is essentially:

$$\hat{\beta_1} = \frac{\Sigma(x_i - \bar{x})y_i}{\Sigma(x_i - \bar{x})^2} \tag{2}$$

If a person's distance from the average $(x_i - \bar{x})$ is too large, it can have inordinate influence on the equation. This idea extends to multiple regression, with multiple predictors.

Cases with values close to the mean matter very little during estimation. This is why we pay attention to outliers because they can easily influence the equation.

Identifying outliers

High leverage (“hat values”) relates to cases with extreme values on the predictors.

High influence relates to cases that alter the line greatly. We identify them by dropping them from the model and refitting the model. We then see how much the coefficients (or model predictions) change in their absence.

Influence plot

```
influencePlot(fit.1)
```

We want smaller leverages, the vertical lines correspond to 2 and 3 times the average leverage. Points which exceed this may be extreme on the predictors.

The studentized residuals tell us how far a point is from the regression equation when the equation was fitted without it.

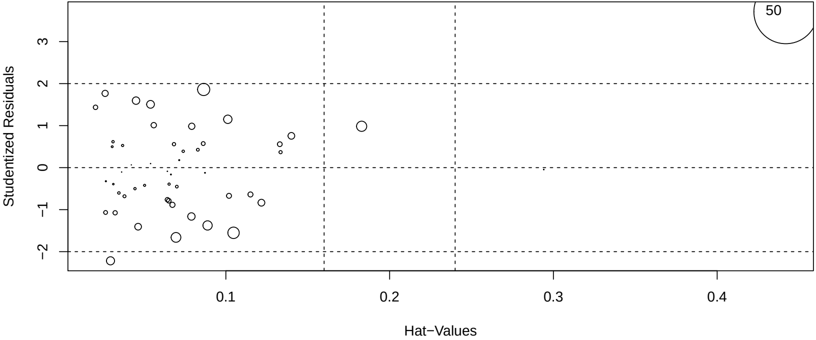
If it is far from the equation, and is an outlier on the predictors, then it can overly influence the equation.

Extreme cases on leverage and studentized residuals make for influential cases - the big circles. A common measure of influence is Cook's Distance. The largest circles have high Cook's Distance.

Do we have see problems cases on the plot?

Influence plot

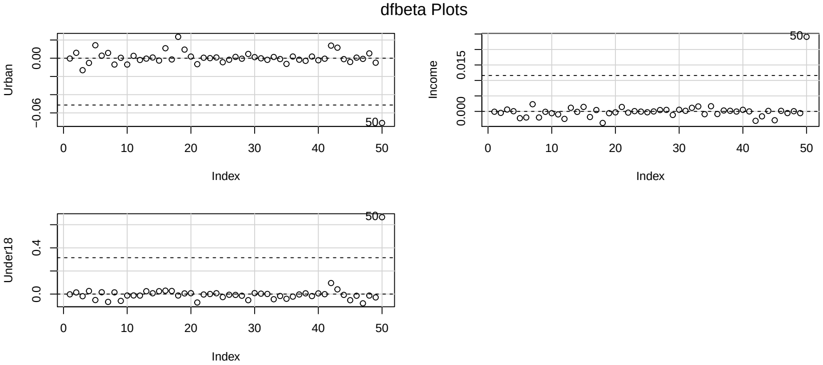
```
influencePlot(fit.1)
```



```
##       StudRes       Hat    CookD
## 50 3.709922 0.4419099 2.132795
```

Additional influence plots - change in coefficients

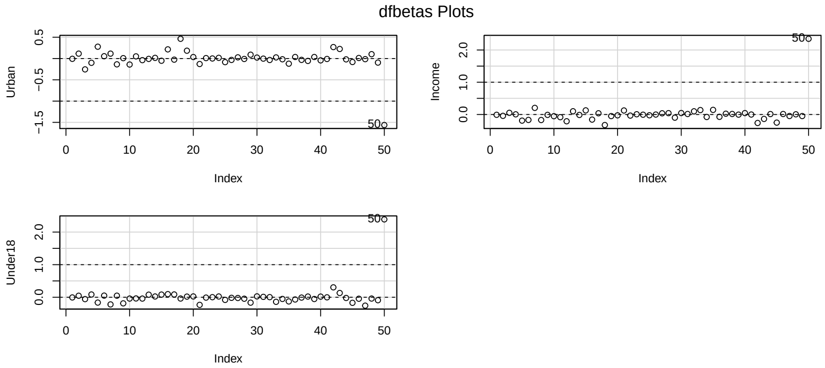
```
dfbetaPlots(fit.1, id.n = 1)
```



Lines represent one standard deviation of the coefficient (standard error).

Additional influence plots - standardized change in coefficients

```r
dfbetasPlots(fit.1, id.n = 1)
```



The evidence is overwhelming against Hawaii

We should drop Hawaii from the regression and fit the model without it to see how the results change.

Use same syntax with subset option

```
fit.2 <- lm(Educ ~ Urban + Income + Under18, edu,
            subset = State != "HI")
```

Let's introduce stargazer

```r
library(stargazer)
stargazer(fit.1, fit.2, type = "html", out = "reg.html")
```

Check for a new file called reg.html in your folder. Open it in your browser.

You can select, copy then paste the table from your browser to Microsoft Word.

Try `?stargazer` to see more options for how to enrich/modify the table. Check https://www.jakeruss.com/cheatsheets/stargazer/ for extensive details.

Robust methods

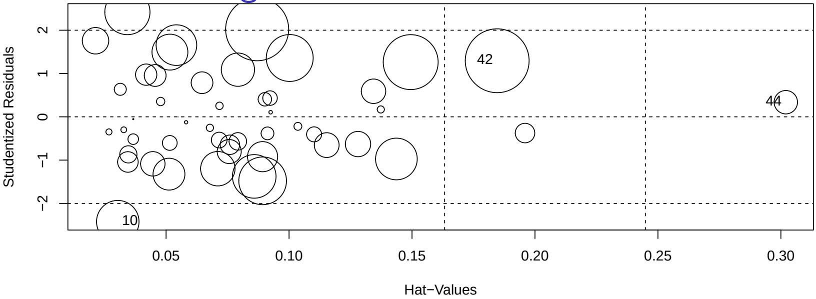
Dropping cases

Some might have problems with dropping a case from the model.

In practice, unless you can justify dropping a case (bad data, different population, . . .), it may not be advisable to drop outlying cases.

Additionally, after dropping cases, we should conduct diagnostics again to see that no new Hawaii's have propped up - you may have to loop through this process several times if you have thousands of cases.

New Hawaiis? using fit.2



```
##        StudRes       Hat      CookD
## 10 -2.4220398 0.0303599 0.04143791
## 42  1.2951332 0.1846406 0.09355313
## 44  0.3376777 0.3020002 0.01258150
```

Though we have large bubbles, note that the uppermost and lowest sections on the right are empty. We do not have any extreme combinations of high leverage and large residuals. We may rest easy for now.

Handling outlying cases - Case weighting

Cases are down-weighted during estimation if they are extreme, so their contribution to the model is lessened.

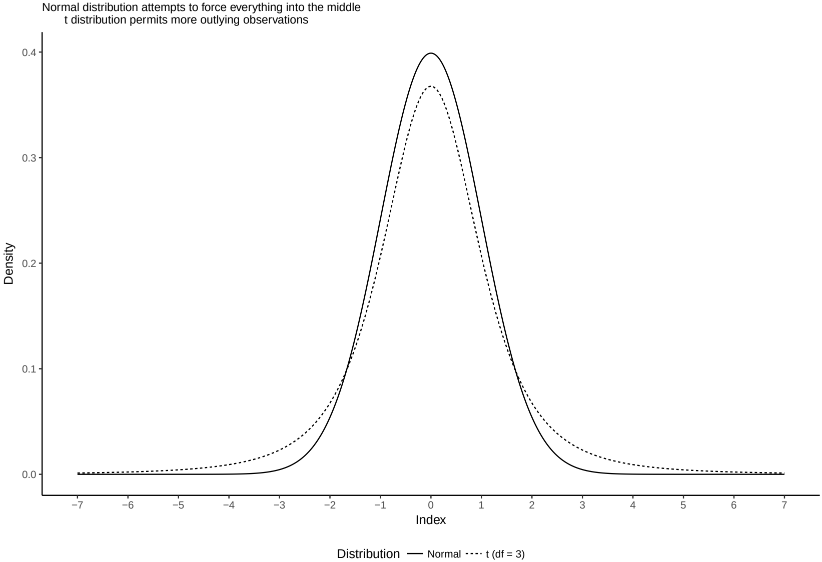
```r
library(robustbase)
fit.rob <- lmrob(Educ ~ Urban + Income + Under18, edu)
stargazer(fit.1, fit.2, fit.rob,
          type = "html", out = "reg.html")
```

Using the default options in `lmrob()` as we have done accounts for both outliers and heteroskedasticity. See `?lmrob` for citations.

In practice, some modellers skip diagnostics and go right to robust alternatives. If the results are markedly different, then they go to diagnostics.

Handling outlying cases - different error distribution

Assuming a different distribution for the errors. If we have extreme outliers, then the normal distribution may not be a great idea.



Regression with t-distributed errors

```
library(hett)
fit.t <- tlm(Educ ~ Urban + Income + Under18,
              ~ Urban + Income + Under18, edu)
```

In this model, the first formula is for the outcome, the second formula is to model the variation in the residuals. So we can attempt to deal with both outliers and heteroskedasticity.

Citation in “English”:

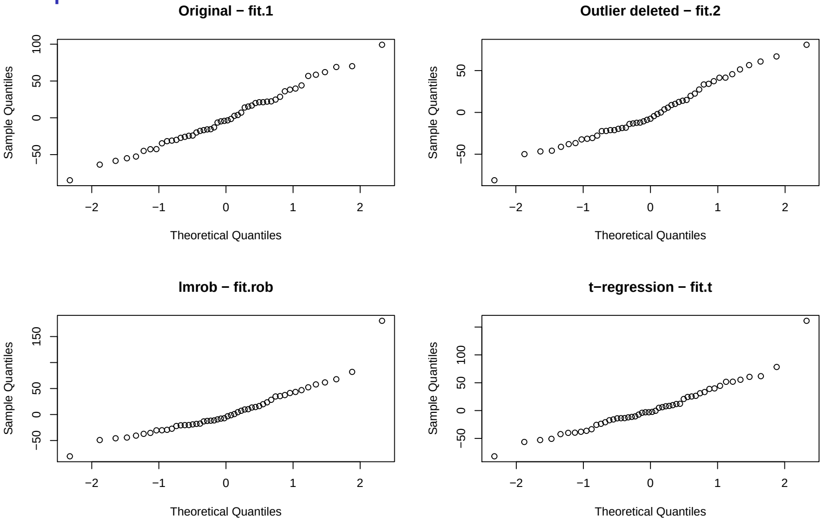
- Section titled: The t distribution instead of the normal in Gelman, A., & Hill, J. (2007). Data analysis using regression and multilevel/hierarchical models.

Regression with t-distributed errors

```r
summary(fit.t)$loc.summary
```

```
##
## Call:
## tlm(lform = Educ ~ Urban + Income + Under18, sform = ~Ur
##     Income + Under18, data = edu)
##
## Residuals:
##     Min        1Q    Median        3Q       Max
## -82.125  -20.037   -2.668   26.175  161.241
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) -313.98324   97.51485  -3.220  0.00235 **
## Urban          0.03519    0.03755   0.937  0.35363
## Income         0.06121    0.01082   5.656 9.45e-07 ***
## Under18        0.87606    0.23507   3.727  0.00053 ***
## ---
```

A comparison of residuals



- In the robust models, outlier has huge residual (lacks influence).
- In the model with case deletion, residuals more normally distributed.

Going on to heteroskedasticity

The deleted model seems to perform well, I'll invoke some theory and delete Hawaii. OLS is familiar to reviewers. I'll be sure to report Hawaii's deletion with justification.

Is there any heteroskekasticity in `fit.2`? Two practical approaches to address heteroskedasticity:

- Use heteroskedasticity-consistent standard errors: They yield correct p-values, especially at large sample sizes.
- Use *wild-bootstrapping* technique: More adequate at smaller sample sizes.

Good citation:

Hausman, J., & Palmer, C. (2012). Heteroskedasticity-robust inference in finite samples. Economics Letters, 116(2), 232–235. https://doi.org/10.1016/j.econlet.2012.02.007

Heteroskedasticity-consistent standard errors (HCSEs)

HCSEs account for heteroskedasticity by using the residuals of observations in the formula for standard errors.

HCSEs do not change your regression coefficients, just your standard errors, hence *p*-values. They are preferably used at large sample sizes, but have the advantage of computational convenience.

There are several types of HCSEs. `HC4` is a good choice based on simulation studies.

```r
library(sandwich)
fit.2.r <- coeftest(fit.2, vcov. = vcovHC, type = "HC4")
stargazer(fit.1, fit.2, fit.rob, fit.2.r,
          type = "html", out = "reg.html")
```

Wild bootstrapping to deal with heteroskedasticity

Bootstrapping is a commonplace tool for statistical inference when mathematical formula for standard errors are not available or may be problematic.

There are many types of bootstrapping, and today, we will look at the wild bootstrap, which is helpful for handling heteroskedasticity especially at small sample sizes.

```r
library(hcci)
```

Wild bootstrapping to deal with heteroskedasticity

The `Tboot()` has a strangeness about it. We need to refit the linear regression model in this way:

```
# First, drop Hawaii
edu.noHI <- edu[-50, ]
fit.2.again <- lm(edu.noHI$Educ ~ edu.noHI$Urban +
                    edu.noHI$Income + edu.noHI$Under18)
fit.hc <- Tboot(fit.2.again, significance = .05, hc = 4,
                  double = TRUE, J = 1000, K = 100)
```

I would only do this when I am done with modelling, and want a final check using an extensive method. `Tboot()` combines HCSEs with bootstrapping to produce a fairly reliable test. However, it is only worth the effort if the sample size is small and our results are borderline (p ~= .05).

Exercises

Voting for Bill (Clinton)

We'll experiment with what we have learned today on the Clinton votes dataset.

Read in the file. Each case is a county with

- Name: County Name
- Vote: Percent voting for Clinton in 1992 (Outcome)
- Age: Median Age
- Savings: Mean Savings($)
- Income: PerCapita Income ($)
- Poverty: Percent in Poverty
- Veterans: Percent Veterans
- Female: Percent Female
- Density: Population Density
- Nursing: Percent in Nursing Homes
- Crime: Crime Index (Per capita)

Run a regression model using variables to predict outcome.

Diagnosing issues

You can work in groups. Share the responsibility of diagnosing problems with the model.

You could either:

- Perform diagnostics: normality of residuals; outliers, homoskedasticity; or
- Try robust models. If there are discrepancies, then back to diagnostics

Make decisions, case deletion? OLS acceptable? See if you can delete cases and reconcile OLS and robust coefficient estimates.

Atlanta schools

We'll experiment with what we have learned today on the Atlanta schools dataset.

Read in the file. Each case is a school with:

- PPC: Per-Pupil Cost (Outcome)
- DA: Average daily Attendance
- MTS: Average Monthly Teacher Salary
- ATT: Percent Attendance
- PTR: Pupil/Teacher ratio

Run a regression model using variables to predict outcome.

Diagnosing issues

You can work in groups. Share the responsibility of diagnosing problems with the model.

You could either:

- Perform diagnostics: normality of residuals; outliers, homoskedasticity; or
- Try robust models. If there are discrepancies, then back to diagnostics

Make decisions, case deletion? OLS acceptable? See if you can delete cases and reconcile OLS and robust coefficient estimates.