36-401 Class Notes
Robust Regression II

Method	OLS	L1	Huber	Bisquare
$\rho(u)$	u^2	$\lvert u\rvert$	u^2 if $\lvert u\rvert \leq c$ [1] $c(2\lvert u\rvert - c)$ if $\lvert u\rvert > c$	$(c^2/3)(1-[1-(u/c)^2]^3)$ if $\lvert u\rvert \leq c$ [2] $c^2/3$ if $\lvert u\rvert > c$
$w = \psi(u)/u$	1 [3]	$1/\lvert u\rvert$	1 if $\lvert u\rvert \leq c$ $c/\lvert u\rvert$ if $\lvert u\rvert > c$	$[1-(u/c)^2]^2$ if $\lvert u\rvert \leq c$ 0 if $\lvert u\rvert > c$

[1] c is a tuning parameter usually set equal to 1.345 for the Huber function.
[2] c is a tuning parameter usually set equal to 4.685 for the Bisquare function.
[3] Ignoring any multiplicative constants...

1. Robust Standard Errors

   - Standard errors are based on data following the correct model:

     $$Y_i = \mathbf{x_i}\boldsymbol{\beta} + \epsilon_i \quad \epsilon_i \sim \text{i.i.d, Var}(\epsilon_i) = \sigma^2$$

   - Variance-covariance matrix for the coefficient estimates $\mathbf{B}$ is based on:

     $$S = \sigma^2 \frac{E[\psi^2(\epsilon)]}{E[\psi'(\epsilon)]^2}(X^T X)^{-1}$$

     where $\psi(u) = \rho'(u)$. Somewhat complex to implement, often done by computer.
   - Since coefficient estimates, $\mathbf{B}$, are asymptotically normal and unbiased, tests and confidence intervals OK in "asymptopia"

2. Using Robust Estimation

   - Discrepancies between OLS and robust regression suggest outlier problems
   - Robust residuals and/or weights help identify outliers
   - Robust results should be used if it appears that the true distribution of the residuals is heavy tailed.
   - For only a few outliers, it is often better to just report OLS results with and without the outliers.
   - Robust regression remains vulnerable to curvilinearity, leverage (unusual X values), and multicollinearity.
   - High leverage cases can control a robust regression that uses OLS starting values because small residual cases don't get down-weighted.

3. Bounded influence regression – down weight high leverage (potential)

   One possibility: replace the weights w_i at each step in IRLS with $w_i \cdot w_i^H$, where

$$w_i^H = \begin{cases} 1 & \text{if } h_{ii} \le c^H \\ c^H/h_{ii} & \text{if } h_{ii} > c^H \end{cases}$$

   c^H could be chosen to be the 90^{th} percentile of the h_{ii}'s, for example.

4. Summary

   - Robust regression sample-to-sample variation of coefficients and predictions is smaller than for OLS if the errors follow a heavy-tailed distribution (and are larger under a Gaussian distribution).

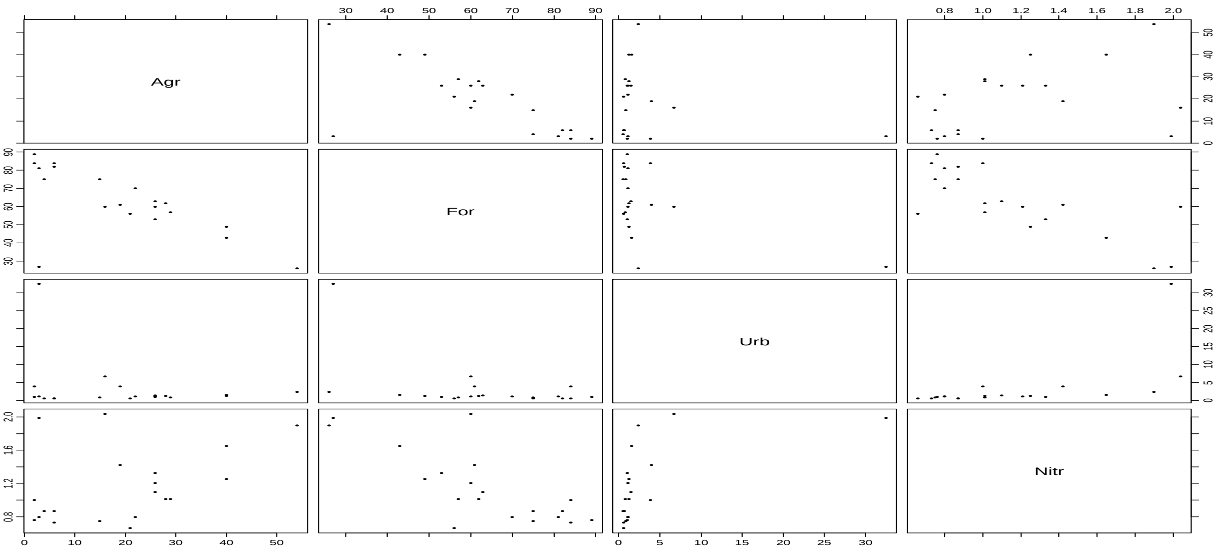
   - Robust regression can be used as a check on OLS.

   - If we need quick rough-and-ready answers or we believe the tails are heavy, robust regression results can be used for parameter testing, prediction, etc.

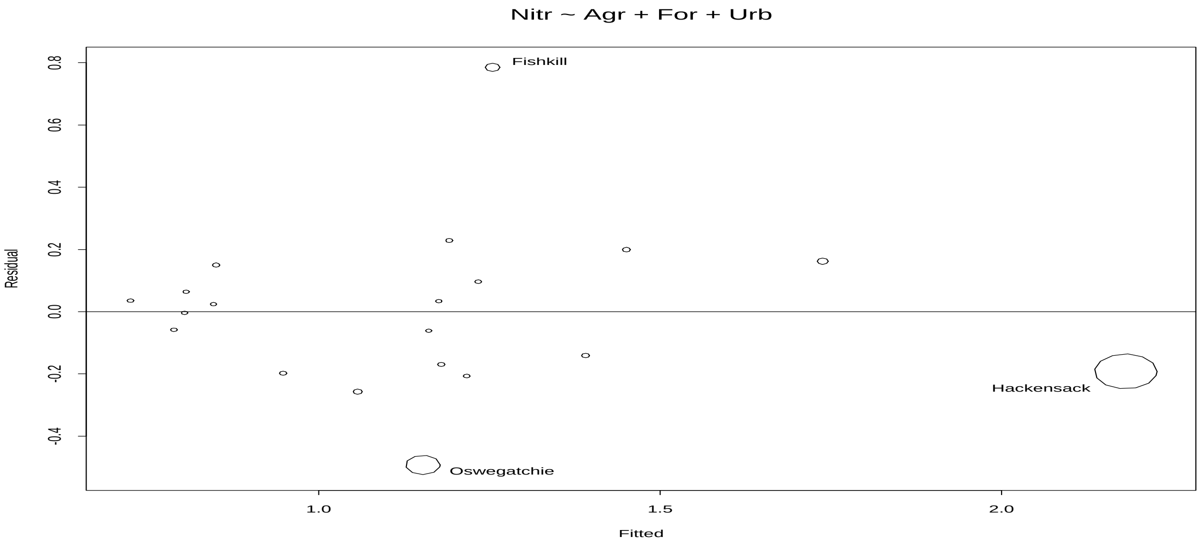
   - Exploratory data analysis, choosing transformations, leverage checks, multicollinearity checks, and examination of residuals are still required.

Example

```
# RwG p. 213, problem 6-8:10   River basin nitrogen vs. land use percentages

river_read.table("river.dat",T)
dim(river)   # 20 4
river[1:5,]
#             Agr For   Urb Nitr
#       Olean  26  63  1.49 1.10
#   Cassadaga  29  57  0.79 1.01
#       Oatka  54  26  2.38 1.90
#   Neversink   2  84  3.88 1.00
# Hackensack    3  27 32.51 1.99

pairs(river)
```




```
# OLS
rslt_lm(Nitr~Agr+For+Urb, river)
summary(rslt)
# Coefficients: Value Std. Error t value Pr(>|t|)
# (Intercept)  1.4286  1.2927     1.1051  0.2854
#         Agr  0.0085  0.0158     0.5376  0.5982
#         For -0.0084  0.0145    -0.5840  0.5674
#         Urb  0.0294  0.0276     1.0638  0.3032
#
# Residual standard error: 0.2802 on 16 degrees of freedom
# Multiple R-Squared: 0.6531
# F-statistic: 10.04 on 3 and 16 degrees of freedom, the p-value is 0.0005818

# Check Cook's distance and DFBETAS
CookPlot(rslt, id=T)
```




```
inf_influence(rslt)
4/nrow(river)  # 0.2
inf$cook[c(5,7,19)]
# Hackensack  Fishkill Oswegatchie
#   59.55654 0.1419857   0.5197304

2/sqrt(nrow(river))  # 0.45
inf$dfbetas[c(5,7,19),]
#            (Intercept)         Agr         For        Urb
#  Hackensack  -0.3547048  0.69002984  0.62373589 -7.7945257
#    Fishkill   0.0776236 -0.06365847 -0.03540836  0.1140718
# Oswegatchie  -1.5213296  1.42595105  1.49082737  1.4823534

# Check leverage (potential)
2*length(rslt$coef)/nrow(river)  # 0.4
cat(row.names(river)[inf$hat>0.4], round(inf$hat[inf$hat>0.4],2), "\n")
# Hackensack 0.96

# OLS without Hackensack
rsltX_lm(Nitr~Agr+For+Urb, river, subset=row.names(river)!="Hackensack")
# Coefficients: Value Std. Error t value Pr(>|t|)
# (Intercept)  1.6986  0.7626     2.2273  0.0417
#         Agr  0.0021  0.0094     0.2214  0.8278
#         For -0.0138  0.0086    -1.6056  0.1292
#         Urb  0.1562  0.0279     5.5903  0.0001
# Residual standard error: 0.165 on 15 degrees of freedom
# Multiple R-Squared: 0.8589

collin(river[row.names(river)!="Hackensack",1:3], 0.1)
# Agr 0.919  For 0.921  Urb 0.206

# Robust regression
library("MASS")  # Required once per Splus session!!
```

```
# Exclude Hackensack; it probably won't be down-weighted
rrsltX_rlm(Nitr~Agr+For+Urb, river, subset=row.names(river)!="Hackensack")
summary(rrsltX)
# Coefficients: Value Std. Error t value
# (Intercept)  2.5057  0.6144     4.0782
#         Agr -0.0071  0.0076    -0.9428
#         For -0.0225  0.0069    -3.2624
#         Urb  0.1350  0.0225     5.9976
#
# Residual standard error: 0.1315 on 15 degrees of freedom
# Multiple R-Squared: NA
# F-statistic: on and degrees of freedom, the p-value is NA
#
# Correlation of Coefficients:
#     (Intercept)     Agr     For
# Agr -0.9739
# For -0.9947      0.9589
# Urb -0.5005      0.4353  0.4537

par(mfrow=c(2,2), oma=c(1,1,3,1))
plot(rrsltX$fit, rrsltX$res, xlab="Fit", ylab="Residual")
abline(h=0)
qqn(rrsltX$res, main="")
plot(rrsltX$w,  rrsltX$res, xlab="Weight", ylab="Residual")
abline(h=0)
plot(rrsltX$fit,  rrsltX$w, ylab="Weight", xlab="Fit")
abline(h=0)
```