PLSC 503 – Spring 2017
“Variances”

February 28, 2017

Variances: A Generalization

Start with:

$$Y_i = \mathbf{X}_i\boldsymbol{\beta} + u_i$$

with:

$$\mathrm{Var}(u_i) = \sigma^2/w_i$$

with w_{iu} <u>known</u>.

Weighted Least Squares

WLS now minimizes:

$$\text{RSS} = \sum_{i=1}^{N} w_i (Y_i - \mathbf{X}_i \boldsymbol{\beta}).$$

which gives:

$$\begin{aligned} \hat{\boldsymbol{\beta}}_{WLS} &= [\mathbf{X}'(\sigma^2 \boldsymbol{\Omega})^{-1}\mathbf{X}]^{-1}\mathbf{X}'(\sigma^2 \boldsymbol{\Omega})^{-1}\mathbf{Y} \\ &= [\mathbf{X}'\mathbf{W}^{-1}\mathbf{X}]^{-1}\mathbf{X}'\mathbf{W}^{-1}\mathbf{Y} \end{aligned}$$

where:

$$\mathbf{W} = \begin{bmatrix} \frac{\sigma^2}{w_1} & 0 & \cdots & 0 \\ 0 & \frac{\sigma^2}{w_2} & \cdots & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & \frac{\sigma^2}{w_N} \end{bmatrix}$$

Getting to Know WLS

The variance-covariance matrix is:

$$\begin{aligned}\text{Var}(\hat{\beta}_{WLS}) &= \sigma^2(\mathbf{X}'\Omega^{-1}\mathbf{X})^{-1} \\ &\equiv (\mathbf{X}'\mathbf{W}^{-1}\mathbf{X})^{-1}\end{aligned}$$

A common case is:

$$\text{Var}(u_i) = \sigma^2 \frac{1}{N_i}$$

where N_i is the number of observations upon which (aggregate) observation i is based.

"Robust" Variance Estimators

Recall that, if $\sigma_i^2 \neq \sigma_j^2 \forall i \neq j$,

$$\begin{aligned} \text{Var}(\beta_{\text{Het.}}) &= (\mathbf{X}'\mathbf{X})^{-1}(\mathbf{X}'\mathbf{W}^{-1}\mathbf{X})(\mathbf{X}'\mathbf{X})^{-1} \\ &= (\mathbf{X}'\mathbf{X})^{-1}\,\mathbf{Q}\,(\mathbf{X}'\mathbf{X})^{-1} \end{aligned}$$

where $\mathbf{Q} = (\mathbf{X}'\mathbf{W}^{-1}\mathbf{X})$ and $\mathbf{W} = \sigma^2\Omega$.

We can rewrite $\mathbf{Q}$ as

$$\begin{aligned} \mathbf{Q} &= \sigma^2(\mathbf{X}'\Omega^{-1}\mathbf{X}) \\ &= \sum_{i=1}^{N} \sigma_i^2 \mathbf{X}_i \mathbf{X}_i' \end{aligned}$$

Huber's Insight

Estimate $\hat{\mathbf{Q}}$ as:

$$\widehat{\mathbf{Q}} = \sum_{i=1}^{N} \hat{u}_i^2 \mathbf{X}_i \mathbf{X}_i'$$

Yields:

$$\begin{aligned}
\widehat{\mathsf{Var}(\beta)}_{\mathsf{Robust}} &= (\mathbf{X}'\mathbf{X})^{-1}(\mathbf{X}'\widehat{\mathbf{Q}}^{-1}\mathbf{X})(\mathbf{X}'\mathbf{X})^{-1} \\
&= (\mathbf{X}'\mathbf{X})^{-1}\left[\mathbf{X}'\left(\sum_{i=1}^{N} \hat{u}_i^2 \mathbf{X}_i \mathbf{X}_i'\right)^{-1}\mathbf{X}\right](\mathbf{X}'\mathbf{X})^{-1}
\end{aligned}$$

Practical Things

"Robust" VCV estimates:

- are heteroscedasticity-consistent, but
- are biased in small samples, and
- are less efficient than "naive" estimates when $\mathrm{Var}(u) = \sigma^2 \mathbf{I}$.

"Clustering"

Huber / White ????????? WLS / GLS

I know very little about my error variances...

I know a great deal about my error variances...

“Clustering”

A common case:

$$Y_{ij} = \mathbf{X}_{ij}\boldsymbol{\beta} + u_{ij}$$

with

$$\sigma^2_{ij} = \sigma^2_{ik}.$$

“Robust, clustered” estimator:

$$\widehat{\mathrm{Var}(\boldsymbol{\beta})}_{\mathrm{Clustered}} = (\mathbf{X}'\mathbf{X})^{-1}\left\{\mathbf{X}'\left[\sum_{i=1}^{N}\left(\sum_{j=1}^{n_j}\hat{u}^2_{ij}\mathbf{X}_{ij}\mathbf{X}'_{ij}\right)\right]^{-1}\mathbf{X}\right\}(\mathbf{X}'\mathbf{X})^{-1}$$

“Real-Data” Example

```
> Justices<-read.csv("Justices.csv")
> attach(Justices)
> summary(Justices)
     name               score               civrts           econs      
 Length:31          Min.   :-1.0000   Min.   :19.80   Min.   :34.60  
 Class :character   1st Qu.:-0.4700   1st Qu.:35.90   1st Qu.:43.85  
 Mode  :character   Median : 0.3300   Median :43.70   Median :50.20  
                    Mean   : 0.1210   Mean   :51.42   Mean   :55.75  
                    3rd Qu.: 0.6250   3rd Qu.:75.55   3rd Qu.:66.65  
                    Max.   : 1.0000   Max.   :88.90   Max.   :81.70  
  Neditorials        eratio           scoresq          lnNedit     
 Min.   : 2.000   Min.   : 0.5000   Min.   :0.0000   Min.   :0.6931  
 1st Qu.: 4.000   1st Qu.: 0.7083   1st Qu.:0.1936   1st Qu.:1.3863  
 Median : 6.000   Median : 1.0000   Median :0.2500   Median :1.7918  
 Mean   : 8.742   Mean   : 2.0242   Mean   :0.4599   Mean   :1.8442  
 3rd Qu.:11.500   3rd Qu.: 2.5000   3rd Qu.:0.8281   3rd Qu.:2.4414  
 Max.   :47.000   Max.   :11.7500   Max.   :1.0000   Max.   :3.8501  
```

OLS...

```
> OLSfit<-with(Justices, lm(civrts~score))
> summary(OLSfit)

Call:
lm(formula = civrts ~ score)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   48.810      2.852  17.113  < 2e-16 ***
score         21.544      4.206   5.122 1.81e-05 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 15.63 on 29 degrees of freedom
Multiple R-squared: 0.475,Adjusted R-squared: 0.4569
F-statistic: 26.24 on 1 and 29 DF,  p-value: 1.806e-05
```

WLS, Weighting by ln(N of Editorials)

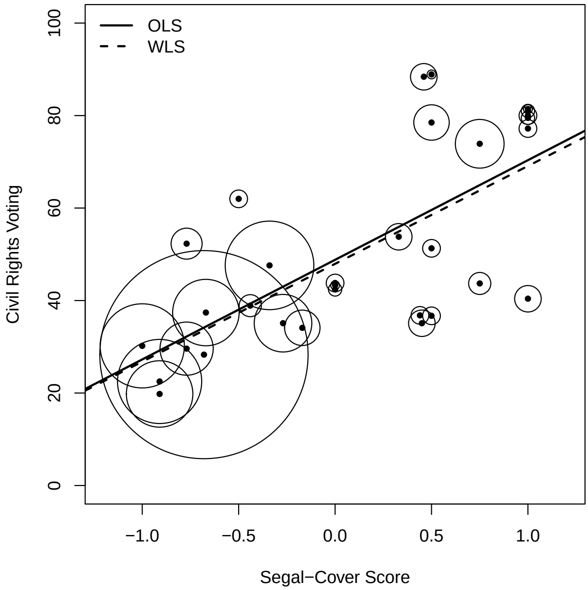
```
> WLSfit<-with(Justices, lm(civrts~score,weights=lnNedit))
> summary(WLSfit)

Call:
lm(formula = civrts ~ score, weights = lnNedit)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   47.936      2.600  18.439  < 2e-16 ***
score         21.158      3.797   5.572 5.18e-06 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 19.59 on 29 degrees of freedom
Multiple R-squared: 0.5171,Adjusted R-squared: 0.5004
F-statistic: 31.05 on 1 and 29 DF,  p-value: 5.179e-06
```

Figure: Plot of `civrts` Against `score`, Weighted by `Neditorials`



“Robust” Standard Errors

```
> library(car)
> hccm(OLSfit, type="hc1")
            (Intercept)     score
(Intercept)    6.963921  2.929622
score          2.929622 13.931212


> library(rms)
> OLSfit2<-ols(civrts~score, x=TRUE, y=TRUE)
> RobSEs<-robcov(OLSfit2)
> RobSEs

Linear Regression Model

ols(formula = civrts ~ score, x = TRUE, y = TRUE)

         n Model L.R.       d.f.          R2      Sigma
        31      19.97          1       0.475      15.63

Residuals:
    Min      1Q  Median      3Q     Max
-29.954  -8.088  -2.120   9.396  29.680

Coefficients:
          Value Std. Error      t  Pr(>|t|)
Intercept 48.81      2.552 19.123 0.000e+00
score     21.54      3.610  5.968 1.739e-06

Residual standard error: 15.63 on 29 degrees of freedom
Adjusted R-Squared: 0.4569
```