Generalized Least Squares

Math 463, Spring 2017, University of Oregon

David A. Levin

University of Oregon

June 2, 2017

Generalized Least Squares (GLS)

- Suppose $\varepsilon \sim \sigma^2 \Sigma$.
- Write $\Sigma = SS^T$. (Note that if Σ has full rank, we can write $\Sigma = U \Lambda U^T$, in which case $S = U \sqrt{\Lambda} U^T$.) If

$$Y = X\beta + \varepsilon\,,$$

then

$$\underbrace{S^{-1} Y}_{Y'} = \underbrace{S^{-1} X}_{X'} \beta + S^{-1} \varepsilon\,.$$

Note that

$$\begin{aligned} \mathrm{Var}(Y') &= \mathrm{Var}(S^{-1} Y) \\ &= S^{-1} \mathrm{Var}(Y) (S^{-1})^T \\ &= \sigma^2 S^{-1} S S^T (S^{-1})^T \\ &= \sigma^2 I\,, \end{aligned}$$

so $Y' = X'\beta + \varepsilon'$, where $\{\varepsilon'_i\}$ are i.i.d. $N(0,1)$ errors.

Example with possible correlation between errors

J. W. Longley (1967) An appraisal of least-squares programs from the point of view of the user. Journal of the American Statistical Association, 62, 819–841.

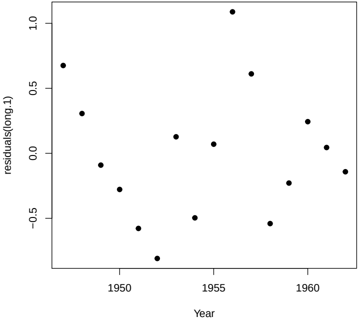
7 economical variables, observed yearly from 1947 to 1962 ($n = 16$):

variable	description
GNP.deflator	GNP implicit price deflator (1954=100)
GNP	Gross National Product
Unemployed	number of unemployed
Armed.Forces	number of people in the armed forces.
Population	non-institutionalized population 14 years of age or older
Year	the year (time)
Employed	number of people employed

Fit a model by regressing Employed on GNP and Population

Residuals vs. time

```
> long.1 = lm(Employed~GNP+Population, data=longley)
> plot(residuals(long.1)~Year, data=longley,pch=19)
```



Appears to be dependence of ε_i on ε_{i-1}.

- Model the errors as

$$\varepsilon_{i+1} = \rho\varepsilon_i + \delta_{i+1},$$

  where $\{\delta_i\}$ is a sequence of i.i.d. $N(0, \sigma^2)$ errors.
- Set

$$\Sigma_{ij} = \mathrm{Cor}(\varepsilon_i, \varepsilon_j) = \rho^{|i-j|}\,.$$

- Estimate ρ by the sample correlation between

$$(\hat{\varepsilon}_2, \ldots, \hat{\varepsilon}_n)$$

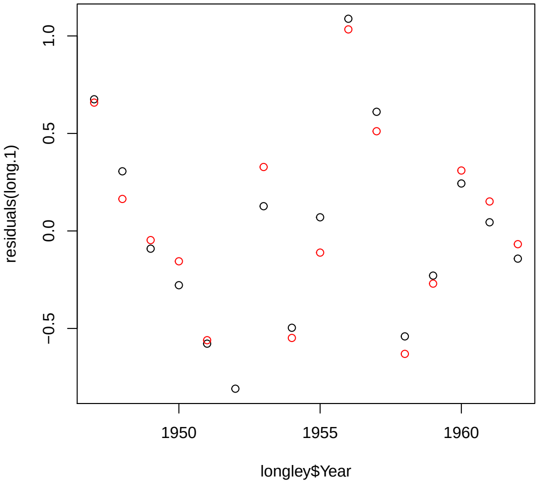
  and

$$(\hat{\varepsilon}_1, \ldots, \hat{\varepsilon}_{n-1})\,.$$

  In this case, $\hat{\rho} = 0.310$.

```
> X = model.matrix(long.1)
> rhoest <- function(v){
+   rc = cor(v[-1],v[-16])
+   return(rc)
+ }
> f <- function(rh){
+   Ma = diag(16)
+   Ma = rh^abs(row(Ma)-col(Ma))
+   ea <- eigen(Ma)
+   Va <- ea$vectors
+   Sa = Va%*%diag(I(sqrt(1/ea$values)))%*%t(Va)
+   Sai = Va%*%diag(sqrt(ea$values))%*%t(Va)
+   SYa = Sa%*%longley$Employed
+   SXa = Sa%*%X
+   return(list(lm(SYa~SXa-1),Sai))
+ }
> rhov = 1:11
> rhov[1] = rhoest(residuals(long.1))
> for(i in 1:10){
+   rh = rhov[i]
+   newf = f(rhov[i])
+   rhov[i+1] = rhoest(newf[[2]]%*%residuals(newf[[1]]))
+ }
> rhov

 [1] 0.3104092 0.3564161 0.3644994 0.3659364 0.3661923 0.3662378 0.3662460
 [8] 0.3662474 0.3662477 0.3662477 0.3662477
```

```
> plot(residuals(long.1)~longley$Year)
> points(residuals(newf[[1]])~longley$Year, col="red")
```



Weighted Least Squares

Suppose that

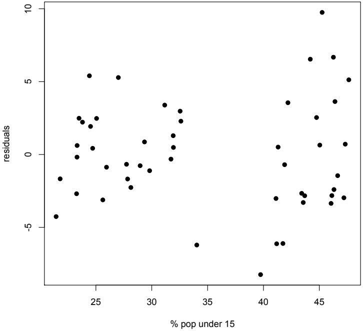
$$\Sigma = \begin{bmatrix} 1/w_1 & 0 & \cdots & 0 \\ 0 & 1/w_2 & \cdots & 0 \\ \vdots & \cdots & \ddots & \vdots \\ 0 & \cdots & \cdots & 1/w_n \end{bmatrix}$$

Then

$$S^{-1} = \begin{bmatrix} \sqrt{w_1} & 0 & \cdots & 0 \\ 0 & \sqrt{w_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & \sqrt{w_n} \end{bmatrix}$$

Savings data-set: savings rate is regressed on

- percent population under 15,
- percent population over 75,
- per-capita disposable income,
- percent growth rate of per-capita disposable income.



Estimate two variances: pop15 < 35 and pop15 ≥ 35.
$\hat{\sigma}_1 = 7.42, \hat{\sigma}_2 = 20.66$.

```
f2 <- lm(sr~.,data=savings, weights=w)
```

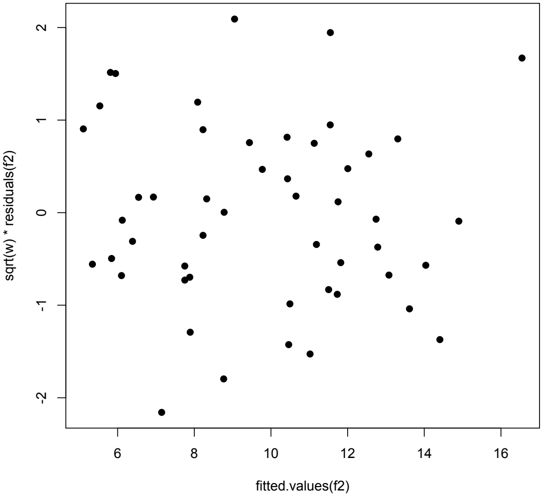
Note that

$$\mathrm{Var}(Y_i - E(Y_i)) = \sigma^2 / w_i,$$

so

$$\mathrm{S.D.}(\sqrt{w_i}(Y_i - EY_i)) = \sigma\,.$$

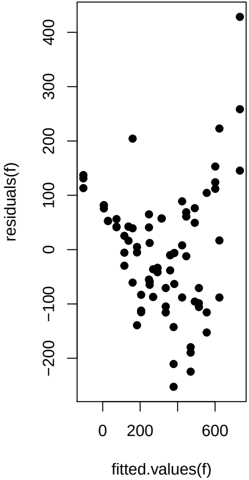
So should check if the spread of $\sqrt{w_i} \times \hat{\varepsilon}_i$ depends on i.

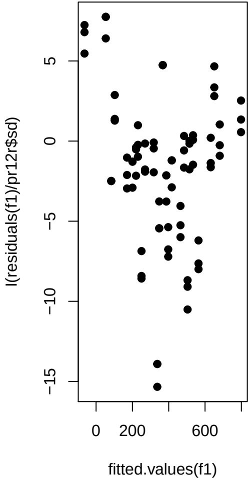

sqrt(w) * residuals(f2)
fitted.values(f2)
2
1
0
-1
-2
6
8
10
12
14
16

Dealing with stacked data frames

```
> library(gdata)
> library(xtable)
> pr12 = read.xls("/Users/dlevin/Dropbox/COURSES/MATH463_S16/SLIDES/GLS/data-prob-5-12.XLS")[1:6]
> names(pr12)[4:6] = c("y.1","y.2","y.3")
> # always rename columns with this convention if you are going to reshape them
> pr12[1:3,]

  x1 x2 x3 y.1 y.2 y.3
1 -1 -1 -1  34  10  28
2  0 -1 -1 115 116 130
3  1 -1 -1 192 186 263

> pr12$sd =  apply(pr12[4:6],1,sd)+1
> pr12r = reshape(pr12,varying=c("y.1","y.2","y.3"),direction="long")
> pr12r[c(1:2,28:29,55:56),]

    x1 x2 x3        sd time   y id
1.1 -1 -1 -1 13.489996    1  34  1
2.1  0 -1 -1  9.386497    1 115  2
1.2 -1 -1 -1 13.489996    2  10  1
2.2  0 -1 -1  9.386497    2 116  2
1.3 -1 -1 -1 13.489996    3  28  1
2.3  0 -1 -1  9.386497    3 130  2

> f = lm(y~x1+x2+x3,data=pr12r)
> par(mfrow=c(1,2))
> plot(residuals(f)~fitted.values(f),pch=19)
> f1 = lm(y~x1+x2+x3,data=pr12r,weights=I(1/sd^2))
> plot(I(residuals(f1)/pr12r$sd)~fitted.values(f1),pch=19)

> xtable(rbind(coef(f),coef(f1)))
```

	(Intercept)	x1	x2	x3
1	314.67	177.00	109.43	131.46
2	367.26	166.69	117.08	147.17


residuals(f)
400
300
200
100
0
−100
−200
0
200
600
fitted.values(f)


I(residuals(f1)/pr12r$sd)
5
0
−5
−10
−15
0
200
600
fitted.values(f1)