Generalized Least Squares

Lecture 11

Alexandra Chronopoulou

ILLINOIS

COLLEGE OF LIBERAL ARTS & SCIENCES

Department of Statistics
101 Illini Hall, MC-374
725 S. Wright St.
Champaign, IL 61820-5710

Learning objectives

In this lecture we will discuss:

- Generalized Least Squares: Σ known
- Generalized Least Squarest: Σ unknown

Generalized Least Squares

What do we do if the errors are correlated or heteroscedastic?

Suppose $\varepsilon \sim \mathcal{N}_n(\mathbf{0}, \Sigma)$, where Σ is the variance-covariance matrix.

We will consider two cases:

- Σ known: this is an idealized case from which we can get some insight.
- Σ unknown

Σ known

Linear Model

$$\mathbf{y} = \mathbf{X}\beta + \varepsilon$$

where $\varepsilon \sim \mathcal{N}_n(\mathbf{0}, \Sigma)$ and Σ is a *known*, *symmetric*, *positive definite* covariance matrix.

When the errors are heteroscedastic or correlated:

Transform this problem back to *Ordinary Least-Squares (OLS)*:

1. Assume S^{-1} exists and write

$$\Sigma = SS^\top$$

   We could use, for example, the *Cholesky decomposition* from linear algebra to obtain S[1].

[1] **R** will do this for us

GLS: Σ known

2. Multiply the model equation by S^{-1} on both sides:

$$\begin{aligned}
\mathbf{y} &= \mathbf{X}\beta + \varepsilon \\
S^{-1}\mathbf{y} &= S^{-1}(\mathbf{X}\beta + \varepsilon) \\
\underbrace{S^{-1}\mathbf{y}}_{:=\mathbf{y}^*} &= \underbrace{S^{-1}\mathbf{X}}_{:=\mathbf{X}^*}\beta + \underbrace{S^{-1}\varepsilon}_{:=\varepsilon^*} \\
\mathbf{y}^* &= \mathbf{X}^*\beta + \varepsilon^*
\end{aligned}$$

This implies that

$$\varepsilon^* \sim \mathcal{N}\left(S^{-1}\mathbf{0}, \underbrace{S^{-1}\Sigma(S^{-1})^\top}_{=\text{Identity}}\right) = \mathcal{N}(\mathbf{0}, \mathbf{I})$$

since $S^{-1}\Sigma(S^{-1})^\top = S^{-1}SS^\top(S^{-1})^\top = \mathbf{I}$.

3. For the transformed model, we can solve for β using OLS:

$$\mathbf{y}^* = \mathbf{X}^*\beta + \varepsilon^*,$$

where $\mathbf{y}^* = S^{-1}\mathbf{y}$, $\mathbf{X}^* = S^{-1}\mathbf{X}$.
So, the estimator for β computes as

$$\begin{aligned}\hat{\beta} &= (\mathbf{X}^{*\top}\mathbf{X}^*)^{-1}\mathbf{X}^{*\top}\mathbf{y}^* \\ &= (\mathbf{X}^\top \underbrace{(S^{-1})^\top S^{-1}}_{=\Sigma^{-1}} \mathbf{X})^{-1}\mathbf{X}^\top \underbrace{(S^{-1})^\top S^{-1}}_{=\Sigma^{-1}} \mathbf{y} \\ &= (\mathbf{X}^\top \Sigma^{-1}\mathbf{X})^{-1}\mathbf{X}^\top \Sigma^{-1}\mathbf{y}\end{aligned}$$

Note that the solution we obtained minimizes:

$$||\mathbf{y}^* - \mathbf{X}^*\beta||^2 = (\mathbf{y} - \mathbf{X}\beta)^\top \Sigma^{-1}(\mathbf{y} - \mathbf{X}\beta)$$

Weighted Least Squares (WLS)

- Suppose that Σ is a diagonal matrix of unequal error variances:

$$\Sigma = diag(\sigma_1^2, \sigma_2^2, \ldots, \sigma_n^2)$$

- The GLS estimate of β minimizes:

$$(\mathbf{y} - \mathbf{X}\beta)^\top \Sigma^{-1} (\mathbf{y} - \mathbf{X}\beta) = \sum_{i=1}^{n} \frac{(y_i - \mathbf{x}_i^\top \beta)^2}{\sigma_i^2}$$

  This problem is known as the Weighted Least-Squares (WLS).

- Note that the errors are weighted by

$$w_i = \frac{1}{\sigma_i^2}$$

  smaller weights for samples with larger variances.

WLS Example

strongx data set from the Faraway library

A large number of observations taken for each *momentum* measurement, allows to have a good estimate of the standard deviation *sd* for each value of the response crossx at each energy level. We can use $weights = 1/sd^2$ as a parameter in the lm(.) call.

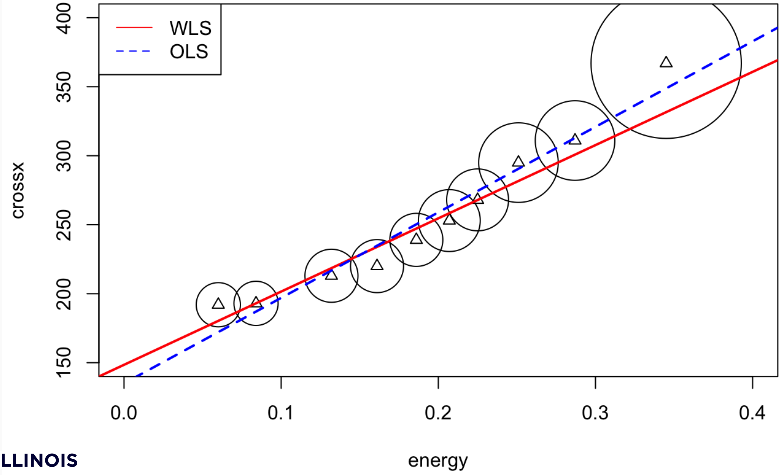
```
data("strongx")
names(strongx)
```

```
## [1] "momentum" "energy"   "crossx"   "sd"
```

```
g=lm(crossx ~ energy, strongx, weights=1/sd^2)
summary(g)
```

OLS vs. WLS

The WLS line departs from values with higher variance (smaller weights)



WLS Special case: Replicated Observations

- Suppose we collected multiple observations for each $\mathbf{x}_i$. We use double subscripts to indicate the replicate observations:

$$(\mathbf{x}_i, y_{i1}, y_{i2}, \ldots, y_{in_i})$$

- Let y_i denote the average of the n_i observations sharing $\mathbf{x}_i$. Then the residual sum of squares for β equals

$$\sum_{i=1}^{n} \sum_{j=1}^{n_i} (y_{ij} - \mathbf{x}_i^\top \beta)^2 = \sum_{i=1}^{n} n_i (y_i - \mathbf{x}_i^\top \beta)^2 + \sum_{i=1}^{n} \sum_{j=1}^{n_i} (y_{ij} - y_i)^2$$

WLS Special case: Replicated Observations

- Minimizing the RSS to solve for β is the same as minimizing the first term on the right only (why?). Because $Var(y_i) = \sigma^2/n_i$, we use WLS on the y_i:

$$\hat{\beta} = \arg\min_{\beta} \sum_{i=1}^{n} n_i (y_i - \mathbf{x}_i^\top \beta)^2$$

- In **R**: Use weights in the lm(.) function: lm(y_i ∼..., weights=n_i,...)

Maximum Likelihood Estimation when Σ is known

- Model: $\mathbf{y} \sim N_n(\mathbf{X}\beta, \Sigma)$
- Log-likelihood:

$$\begin{aligned}
&\log(p(\mathbf{y}|\beta, \Sigma)) \\
&\quad = \log \left\{ \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{y} - \mathbf{X}\beta)^\top \Sigma^{-1}(\mathbf{y} - \mathbf{X}\beta) \right] \right\} \\
&\quad = -\frac{1}{2}(\mathbf{y} - \mathbf{X}\beta)^\top \Sigma^{-1}(\mathbf{y} - \mathbf{X}\beta) + \mathit{Constant}.
\end{aligned}$$

- Therefore the MLE is given by

$$\hat{\beta}_{mle} = \arg\min_{\beta} (\mathbf{y} - \mathbf{X}\beta)^\top \Sigma^{-1}(\mathbf{y} - \mathbf{X}\beta)$$

Σ unknown

When the Variance is Unknown...

- When the variances are known, or even known up to a proportionality constant, the use of Weighted Least Squares with weights

$$w_i = k\,\frac{1}{\sigma_i^2}, \text{ where } k \text{ is a proportionality constant}$$

  would be straightforward.

- Unfortunately, one can rarely has knowledge of the variances σ_i^2.

- We are forced to use estimates of the variances.

Estimation of Variance/Standard Deviation Function

The variance of the error terms ε_i, denoted by σ_i^2 can be expressed as

$$\sigma_i^2 = \mathbb{E}\left(\varepsilon_i^2\right) - \Big(\mathbb{E}(\varepsilon_i)\Big)^2$$

Since we assume that $\mathbb{E}(\varepsilon_i) = 0$, we have

$$\sigma_i^2 = \mathbb{E}\left(\varepsilon_i^2\right)$$

This implies that

- the squared residual r_i^2 is an estimator of σ_i^2, or
- the absolute residual $|r_i|$ is an estimator of the standard deviation σ_i.

Estimate Variance Function

1. Fit a regression model using OLS, and obtain the residuals r_i.
2. Regress the squared residuals r_i^2 against the appropriate predictor variables.

Estimate Standard Deviation Function

1. Fit a regression model using OLS, and obtain the residuals r_i.
2. Regress the absolute residuals $|r_i|$ against the appropriate predictor variables.

After the variance or standard deviation function is estimated, the fitted values from this function are used to obtain the estimated weights:

- Denote $\hat{s}_i$ be the fitted value from standard deviation function

$$w_i = \frac{1}{(\hat{s}_i)^2}$$

- Denote $\hat{v}_i$ be the fitted value from variance function

$$w_i = \frac{1}{\hat{v}_i}$$

The estimated variances are then placed in the variance-covariance matrix Σ and the regression coefficients are estimated using the Weighted Least Squares method.

Blood Pressure Data Example

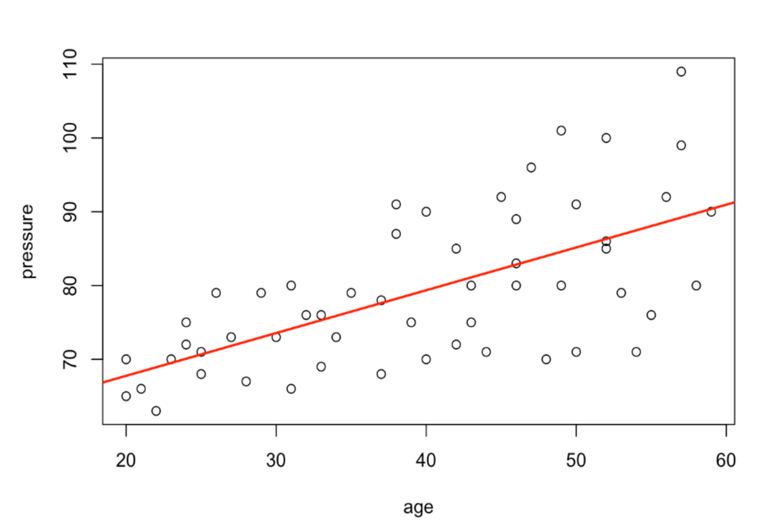
A health researcher interested in studying the relationship between diastolic blood pressure and age among healthy women 20 to 60 years old, collected data on 54 subjects.

```r
pressure <- read.table("blood_pressure.txt", header=FALSE)
names(pressure)=c("age", "pressure")
head(pressure)
```

```
##   age pressure
## 1  27       73
## 2  21       66
## 3  22       63
## 4  24       75
## 5  25       71
## 6  23       70
```

Blood Pressure Data Example

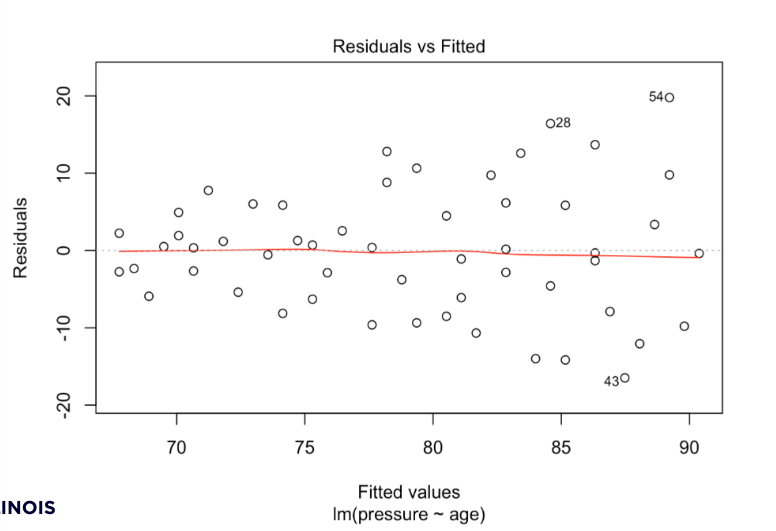
– We start by fitting a linear model between *Blood Pressure* and *Age*:



ILLINOIS

Blood Pressure Data Example

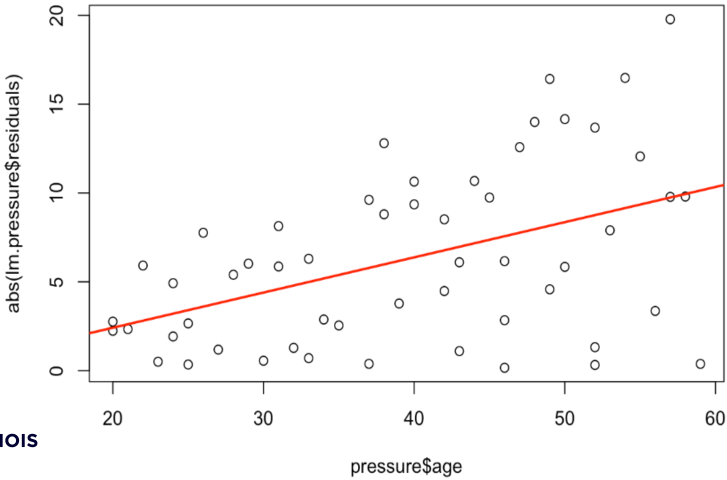
– If we look at the fitted values vs. residuals plot, we observe that the variance increases with *Age*.



Blood Pressure Data Example

- We estimate the standard deviation function, by regressing the *Absolute Residuals* against *Age*.

Absolute *Residuals* vs. *Age*



Blood Pressure Data Example

- The estimated weights are

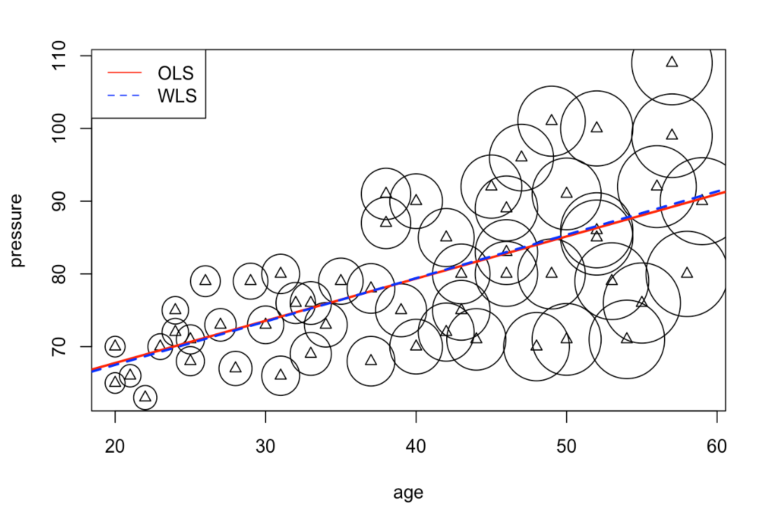
$$w_i = \frac{1}{\hat{s}_i^2}$$

  where $\hat{s}_i$ are the estimated standard deviations of via the regression of the absolute residuals against age.

- We fit a weighted regression function using the w_is as weights.

Blood Pressure Data Example

WLS vs. OLS



Blood Pressure Data Example

Variance Estimators of βs

```
vcov(lm.pressure)
```

```
##             (Intercept)          age
## (Intercept)  15.9494301 -0.371977563
## age          -0.3719776  0.009399527
```

```
vcov(lm.pressure.weights)
```

```
##             (Intercept)          age
## (Intercept)   6.3550256 -0.189363636
## age          -0.1893636  0.006278666
```

How about using the following iterative approach?

1. Start with some initial guess of Σ
2. Use Σ to estimate β
3. Use residuals (since we have known β) to estimate Σ
4. Iterate until convergence.

It looks like a good idea; however the methods will not work if we do not assume some structure about Σ (too many parameters to be estimated).

Generalized Least-Squares: Σ unknown

- Based on the application, we can assume a particular structure for Σ that does not involve too many parameters.
- Then, we can model β and Σ simultaneously.
- For example , for AR(1) times series (auto-regressive model of order 1), the structure of Σ would be:

$$\Sigma = \sigma^2 \begin{pmatrix} 1 & \rho & \rho^2 & \rho^3 & \dots \\ \rho & 1 & \rho & \rho^2 & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \rho^{n-1} & \rho^{n-2} & \dots & \dots & 1 \end{pmatrix}$$

- Σ as a function of ρ and σ^2.

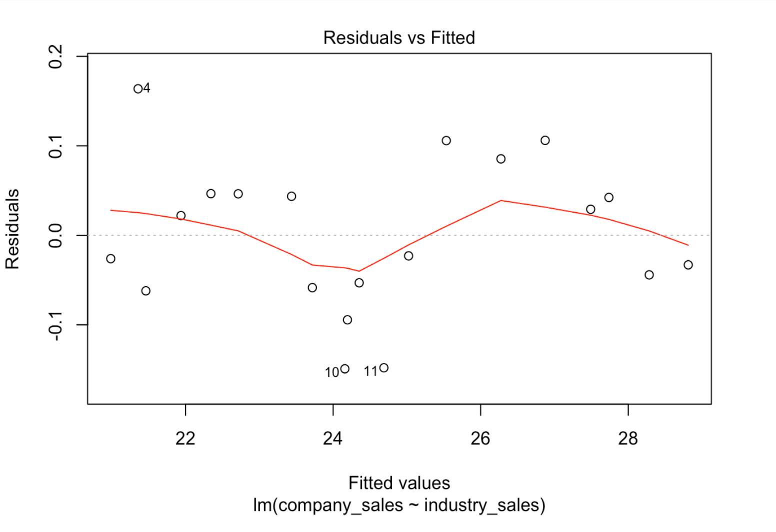
Time series data

A company wants to predict its sales by using industry sales that are available from the industry's trade association, as a predictor.

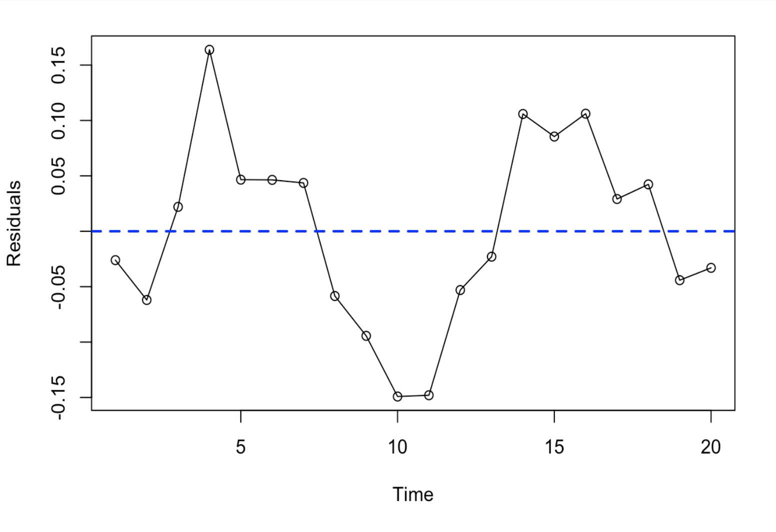
```
sales <- read.table("Sales.txt", header=FALSE)
names(sales)=c("company_sales", "industry_sales")
sales$index = seq(1:dim(sales)[1])
head(sales)
```

```
##   company_sales industry_sales index
## 1         20.96          127.3     1
## 2         21.40          130.0     2
## 3         21.96          132.7     3
## 4         21.52          129.4     4
## 5         22.39          135.0     5
## 6         22.76          137.1     6
```

Example with auto-correlated errors



Example with auto-correlated errors



Test for Auto-Correlation

- Use Durbin-Watson test from the `lmtest` library to test autocorrelation.
- Null hypothesis: Errors are not auto-correlated

```
dwtest(lm.sales)
```

```
##
##  Durbin-Watson test
##
## data:  lm.sales
## DW = 0.73473, p-value = 0.0001748
## alternative hypothesis: true autocorrelation is greater than 0
```

The null hypothesis is rejected, which means that the errors are auto-correlated.

What model should we fit next??

Regression Model with Correlated Errors

```
library(nlme)
lm.sales.cor = gls(company_sales~industry_sales, correlation = corAR1(form= ~ index), data=sales)
summary(lm.sales.cor)
```

```
## Generalized least squares fit by REML
##   Model: company_sales ~ industry_sales
##   Data: sales
##         AIC       BIC   logLik
##   -31.74311 -28.18162 19.87156
##
## Correlation Structure: AR(1)
##  Formula: ~index
##  Parameter estimate(s):
## Phi
##   1
##
## Coefficients:
##                     Value Std.Error  t-value p-value
## (Intercept)    -0.3189197 2041.6945 -0.00016  0.9999
## industry_sales  0.1684878    0.0051 33.06272  0.0000
##
##  Correlation:
##                (Intr)
## industry_sales 0
##
## Standardized residuals:
##           Min            Q1           Med            Q3           Max
## -9.036061e-05 -4.156415e-05 -3.013053e-06  8.080346e-05  1.091922e-04
##
## Residual standard error: 2041.694
## Degrees of freedom: 20 total; 18 residual
```