STAT 425

# Generalized Least Squares (GLS)

# Generalized Least Squares

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

Suppose $\mathbf{e} \sim N_n(\mathbf{0}, \Sigma)$, where $\Sigma$ is the variance-covariance matrix. We will consider two cases:

- $\Sigma$ known (this is an idealized case from which we can get some insight)
- $\Sigma$ unknown (e.g. regression with time series data, spatial data, etc.)

We will discuss some examples and R code

# GLS: $\Sigma$ known

- Assume $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{e}$ and $\mathbf{e} \sim N_n(\mathbf{0}, \Sigma)$ where $\Sigma$ is a known, symmetric, positive definite covariance matrix.
- Transform this problem back to Ordinary Least-Squares (OLS). Write $\Sigma = SS^\top$ where we assume $S^{-1}$ exists. We could use, for example, the Cholesky decomposition from linear algebra to obtain $S$. Multiply the model equation by $S^{-1}$ on both sides:

$$\begin{aligned} S^{-1}\mathbf{y} &= S^{-1}(\mathbf{X}\boldsymbol{\beta} + \mathbf{e}) \\ \mathbf{y}^* &= \mathbf{X}^*\boldsymbol{\beta} + \mathbf{e}^* \\ \mathbf{e}^* &\sim (S^{-1}\mathbf{0}, S^{-1}\Sigma(S^{-1})^\top) = N(\mathbf{0}, \mathbf{I}) \end{aligned}$$

- Now we can solve for $\boldsymbol{\beta}$ using OLS:

$$\begin{aligned}
\mathbf{y}^* &= \mathbf{X}^*\boldsymbol{\beta} + \mathbf{e}^*, \quad \mathbf{y}^* = S^{-1}\mathbf{y}, \quad \mathbf{X}^* = S^{-1}\mathbf{X} \\
\hat{\boldsymbol{\beta}} &= [\mathbf{X}^{*\top}\mathbf{X}^*]^{-1}\mathbf{X}^{*\top}\mathbf{y}^* \\
&= (\mathbf{X}^\top(S^{-1})^\top S^{-1}\mathbf{X})^{-1}\mathbf{X}^\top(S^{-1})^\top S^{-1}\mathbf{y} \\
&= (\mathbf{X}^\top\Sigma^{-1}\mathbf{X})^{-1}\mathbf{X}^\top\Sigma^{-1}\mathbf{y}
\end{aligned}$$

- Note that the solution minimizes:

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

# Weighted Least Squares (WLS)

- Suppose that $\Sigma$ is a diagonal matrix of unequal error variances:

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

- The GLS estimate of $\boldsymbol{\beta}$ minimizes:

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

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

- Note that the errors are weighted by $1/\sigma_i^2$: 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.

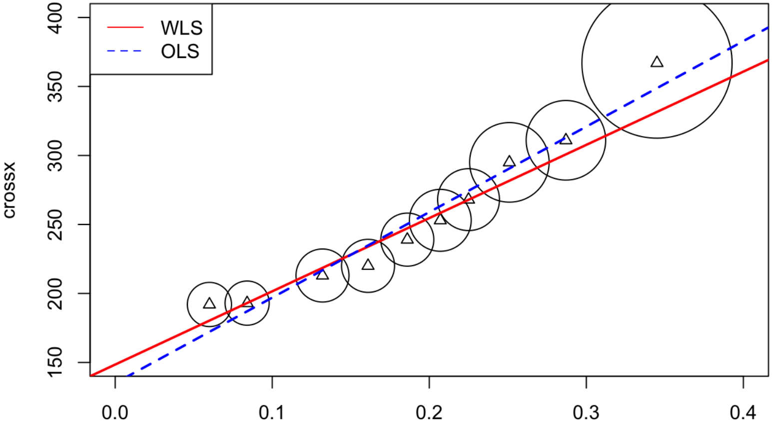
```
data("strongx")
names(strongx)
```

```
## [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 $\beta$ equals

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

Minimizing the RSS to solve for $\boldsymbol{\beta}$ 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 $y_i$:

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

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

# Maximum Likelihood Estimation when $\Sigma$ is known

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

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

- Therefore the MLE is given by

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

# Generalized Least-Squares: $\Sigma$ unknown

How about using the following iterative approach?

1. Start with some initial guess of $\Sigma$
2. Use $\Sigma$ to estimate $\boldsymbol{\beta}$
3. Use residuals (since we have known $\boldsymbol{\beta}$) to estimate $\Sigma$
4. Iterate until convergence

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

Usually, based on the application, we can assume a particular structure for $\Sigma$ that does not involve too many parameters. Then we can model $\boldsymbol{\beta}$ and $\Sigma$ simultaneously. For example , for AR(1) times series (auto-regressive model of order 1), the structure of $\Sigma$ would be:

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

$\Sigma$ as a function of $\rho$ and $\sigma^2$.
Use the **nlme** package in **R**

# Example with auto-correlated errors

## Time series data

- Longley's Economic Regression Data: A data frame with 7 economical variables, observed yearly from 1947 to 1962 (n=16).
- 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: 'noninstitutionalized' population ≥ 14 years of age.
- Year
- Employed: number of people employed.

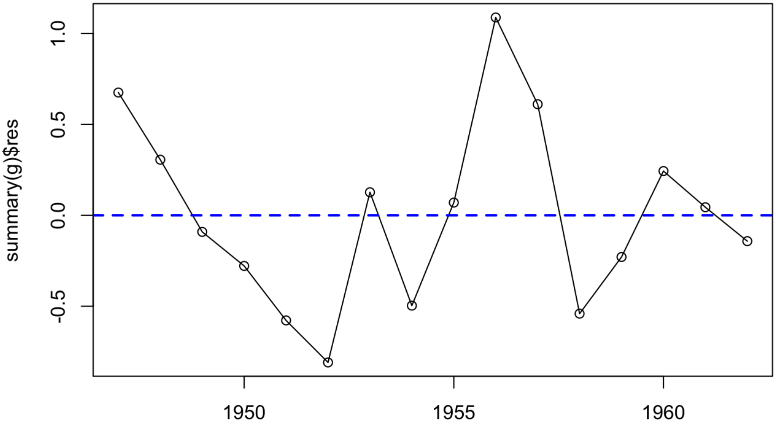
```r
library(faraway)
data("longley")
head(longley)
```

```
##      GNP.deflator     GNP Unemployed Armed.Forces Population Year Employed
## 1947         83.0 234.289      235.6        159.0    107.608 1947   60.323
## 1948         88.5 259.426      232.5        145.6    108.632 1948   61.122
## 1949         88.2 258.054      368.2        161.6    109.773 1949   60.171
## 1950         89.5 284.599      335.1        165.0    110.929 1950   61.187
## 1951         96.2 328.975      209.9        309.9    112.075 1951   63.221
## 1952         98.1 346.999      193.2        359.4    113.270 1952   63.639
```

# Example with auto-correlated errors

Residuals after fitting the model:

g = lm(Employed ∼ GNP + Population, data=longley)



# Test for autocorrelation

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

```
dwtest(g)
```

```
##
##  Durbin-Watson test
##
## data:  g
## DW = 1.3015, p-value = 0.02245
## alternative hypothesis: true autocorrelation is greater than 0
```

```
#D-W test shows the errors are signifficantly correlated
#Solution: Fit a Regression with autocorrelated errors.
library(nlme)
g = gls(Employed ~ GNP + Population, correlation = corAR1(form= ~ Year), data=longley)
summary(g)
```

Use function **gls** from library **nlme**