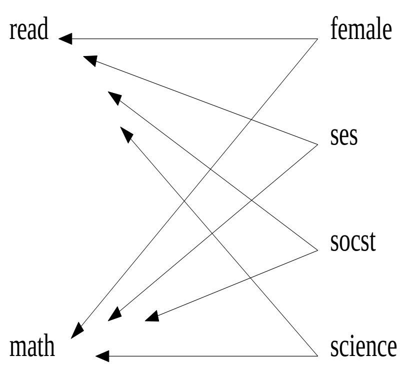
last change 16.01.13 @ 18.00

Summary 1 *multivariate regression, generalized least squares, Kronecker products, SUR regression, aggregation, k-sample problem, MANOVA*

Multivariate Regression

Example:

```
Contains data from http://www.ats.ucla.edu/stat/stata/notes/hsb2.dta
 obs:  200 highschool and beyond (200 cases) vars:  11
------------------------------------------------------------
 1.  id float       %9.0g
 2.  female float   %9.0g gl
 3.  race float     %12.0g rl
 4.  ses float      %9.0g sl <------------social economic status
 5.  schtyp float   %9.0g scl type of school
 6.  prog float     %9.0g sel type of program
 7.  read float     %9.0g reading score
 8.  write float    %9.0g writing score
 9.  math float     %9.0g math score
 10.  science float %9.0g science score
 11.  socst float   %9.0g social studies score
```



$$\texttt{read} = \alpha_1 + \beta_{11}\texttt{female} + \beta_{12}\texttt{ses} + \beta_{13}\texttt{socst} + \beta_{14}\texttt{science} + \epsilon_1$$
$$\texttt{math} = \alpha_2 + \beta_{21}\texttt{female} + \beta_{22}\texttt{ses} + \beta_{23}\texttt{socst} + \beta_{24}\texttt{science} + \epsilon_2$$

nomenclature: simple-multiple-multivariate regression

$\mathbf{y} = [y_1, \ldots, y_p]$, $\mathbf{x} = [x_1, \ldots, x_q]$ row vectors, B matrix $(q \times p)$
model (generic form, one observation)

$$\mathbb{E}\left(\mathbf{y} |\ \mathbf{x}\right) = \mathbf{x}B$$

i.e.

$$y_j = \sum\nolimits_k^q x_k \beta_{kj} + \epsilon_j \ j = 1, \ldots, p$$
$$\epsilon \sim \mathcal{N}_p\left(0, \Sigma\right)$$

Inference (n observations)

$$\text{Data matrix: } \begin{bmatrix} y_{11} & \cdots & y_{1p} & x_{11} & \cdots & x_{1q} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ y_{n1} & \ldots & y_{np} & x_{n1} & \ldots & x_{nq} \end{bmatrix}$$

Model (in matrix notation)

$$\mathbb{E}\left(Y\ |X\right) = XB$$

Vector notation:

$$Y = \Bigg[\underbrace{\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(p)}}_{\text{col}\, vectors}\Bigg] = \begin{bmatrix} y_{11} & \cdots & y_{1p} \\ \vdots & \ddots & \vdots \\ y_{n1} & \ldots & y_{np} \end{bmatrix} = \left[\left.\begin{matrix} \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_n \end{matrix}\right\} \text{row}\, vectors\right]$$

$$X = \left[\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(q)}\right] = \begin{bmatrix} x_{11} & \cdots & x_{1q} \\ \vdots & \ddots & \vdots \\ x_{n1} & \ldots & x_{nq} \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_n \end{bmatrix}$$

$$B = \left[\beta^{(1)}, \ldots, \beta^{(p)}\right] = \begin{bmatrix} \beta_{11} & \cdots & \beta_{1p} \\ \vdots & \ddots & \vdots \\ \beta_{1q} & \ldots & \beta_{pq} \end{bmatrix} = \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_q \end{bmatrix}$$

model (row vectors)

$$\left.\begin{matrix} \mathbf{y}_1, \ldots, \mathbf{y}_n \ \text{ independent} \\ \mathbf{y}_i \sim \mathcal{N}_p(\mathbf{x}_i B, \Sigma) \end{matrix}\right\} \Longrightarrow \left\{\begin{matrix} \mathbf{y}_i = \mathbf{x}_i B + \mathbf{e}_i \ \ i = 1, \ldots, n \\ \mathbf{e}_1, \ldots, \mathbf{e}_n \text{ iid} \sim \mathcal{N}_p(\mathbf{0}, \Sigma) \end{matrix}\right.$$

model (col vectors)

$$\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(p)} \ \ \textbf{not} \text{ independent } \textbf{but} \ \left\{\begin{matrix} \mathbf{y}^{(j)} = X\beta^{(j)} + \mathbf{e}^{(j)}, \ j = 1, \cdots, p \\ \mathbf{e}^{(j)} \sim \mathcal{N}_n\left(0, \sigma_j^2 I\right) \end{matrix}\right.$$

or in matrix notation:

$$Y = XB + E$$

Estimation: OLS & GLS

log likelihood function (one observation)

$$\log f(\mathbf{y}|\ X) \sim -\log|\Sigma| - (\mathbf{y} - XB)\,\Sigma^{-1}\,(\mathbf{y} - XB)'$$

where Σ covariance on each individual

mle for n *iid* observations (ML=LS)

$$\left(\widehat{B},\widehat{\Sigma}\right) = \min\left(n\log|\Sigma| + \sum_{i}^{n}(\mathbf{y}_i - \mathbf{x}_iB)\,\Sigma^{-1}\,(\mathbf{y}_i - \mathbf{x}_iB)'\right)$$

special case: $\boxed{\Sigma = \sigma^2 I}$, σ^2 known. It can be shown that

$$\widehat{B} = (X'X)^{-1}\,X'Y$$

or, since $\widehat{B} = \left[\widehat{\beta}^{(1)},\ldots,\widehat{\beta}^{(p)}\right] = (X'X)^{-1}\,X'\left[\mathbf{y}^{(1)},\ldots,\mathbf{y}^{(p)}\right]$
$\Longrightarrow$

$$\widehat{\beta}^{(j)} = (X'X)^{-1}\,X'\mathbf{y}^{(j)}$$

the MLE (=least squares estimator, **OLS**) can be obtained by running p one

dimensional regressions, one for each component. In this case the model is called **decomposable.**

If σ^2 unknown, estimate (by "corelating the residuals")

$$\widehat{\sigma}^2 = \frac{1}{np}\sum_{j}^{p}\left(\mathbf{y}^{(j)} - X\widehat{\beta}^{(j)}\right)'\left(\mathbf{y}^{(j)} - X\widehat{\beta}^{(j)}\right)$$

case Σ =diag(Σ) similar (diagonally WLS). continue with the general, heteroscedastic, decomposable case

general case: $\boxed{\Sigma \text{ general}}$, *known.* generalized least squares

fundamental **Lemma (GLS):** consider a regression with correlated errors. model

$$\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_{11} & \ldots & x_{1k} \\ \vdots & \ddots & \vdots \\ x_{n1} & \ldots & x_{nk} \end{bmatrix}\begin{bmatrix} \beta_1 \\ \vdots \\ \beta_k \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{bmatrix}$$

$$\mathbf{y} = X\beta + \epsilon$$
$$\epsilon \sim \mathcal{N}_n(0,\Sigma)$$

then the GLS:

$$\widehat{\beta} = \min (\mathbf{y} - X\beta)' \Sigma^{-1} (\mathbf{y} - X\beta) \implies \widehat{\beta} = \left(X'\Sigma^{-1}X\right)^{-1} X'\Sigma^{-1}\mathbf{y}$$

Proof: For the case $\Sigma = \sigma^2 I$ the MLE=OLS: $\widehat{\beta} = (X'X)^{-1} X'\mathbf{y}$

for general Σ we estimate in a transformed model

1. transform error:

$$\epsilon \sim \mathcal{N}(0, \Sigma) \implies \Sigma = AA'$$
$$\epsilon^* = A^{-1}\epsilon \implies \mathbb{V}\epsilon^* = A^{-1}\mathbb{V}\epsilon \ \left(A^{-1}\right)' = I$$

2. transfom model

$$\mathbf{y}^* = A^{-1}\mathbf{y}$$
$$\mathbf{y}^* = A^{-1}X\beta + \epsilon^* = X^*\beta + \epsilon^*$$

in this model the MLE=OLS

$$\widehat{\beta} = (X^{*\prime}X^*)^{-1} X^{*\prime}\mathbf{y}^*$$

3. back transform

$$\widehat{\beta} = (X^{*\prime}X^*)^{-1} X^{*\prime}\mathbf{y}^* = \left(X'\left(A^{-1}\right)'A^{-1}X\right)^{-1} X'\left(A^{-1}\right)'A^{-1}\mathbf{y} \implies$$
$$\widehat{\beta} = \left(X'\Sigma^{-1}X\right)^{-1} X'\Sigma^{-1}\mathbf{y}$$

this is the GLS estimator qed

Note: if Σ is unknown (as is usually the case) estimate it by correlating residuals from univariate OLS regressions (GLS→**Feasible least squares**)
in the model

$$\mathbf{y} = X\beta + \epsilon$$
$$\epsilon \sim \mathcal{N}_k(0, \Sigma)$$

estimate β by OLS

$$\widehat{\beta}_o = (X'X)^{-1} X'\mathbf{y}$$

estimate Σ by

$$\widehat{\Sigma} = \left(\mathbf{y} - X\widehat{\beta}_o\right)\left(\mathbf{y} - X\widehat{\beta}_o\right)'$$

note: this seems a poor estimator to estimate many parameters (correlation). it gets better by reducing the parameters by imposing more structure on correlations (e.g. in blocs).

Apply to general problem: mvReg & SUR

$$Y = XB + E$$

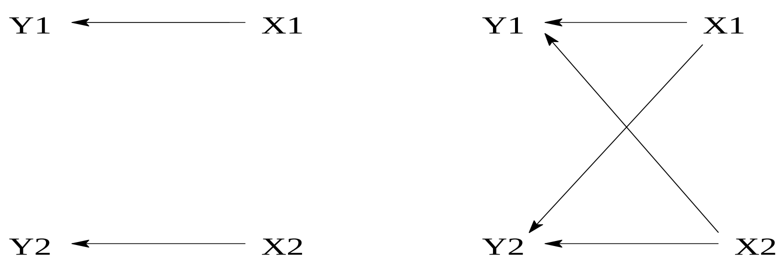
1) **mvReg**(colums)

$$\mathbf{y}^{(j)} = X\beta^{(j)} + \mathbf{e}^{(j)},\ j = 1, \cdots, p$$
$$\mathbf{e}^{(j)} \sim \mathcal{N}_n\left(0, \sigma_j^2 I\right)$$

2) **SUR**: mvReg where the (column) vectors $\beta^{(1)}, \ldots, \beta^{(p)}$ may have some komponents equal to zero ("structural zeroes"). more generally, define a SUR

$$\mathbf{y}^{(j)} = X_j\beta^{(j)} + \mathbf{e}^{(j)},\ j = 1, \cdots, p$$
$$\mathbf{e}^{(j)} \sim \mathcal{N}_n\left(0, \sigma_j^2 I\right)$$

where the matrices $X_1, \ldots, X_p$ may or may not be identical (or completely unrelated) .



Example:
mvReg:

$$y_1 = \beta_{10} + \beta_{11}x_1 + \beta_{21}x_2 + \epsilon_1$$
$$y_2 = \beta_{20} + \beta_{12}x_1 + \beta_{22}x_2 + \epsilon_2$$

where cov$(\epsilon_1, \epsilon_2) \neq 0$
SUR

$$y_1 = \widetilde{\beta}_{10} + \widetilde{\beta}_{11}x_1 + \epsilon_1$$
$$y_2 = \widetilde{\beta}_{20} + \widetilde{\beta}_{22}x_2 + \epsilon_2$$

where cov$(\epsilon_1, \epsilon_2) \neq 0$
in matrix form
mvReg

$$\mathbf{y}_1 = X\beta_1 + \epsilon_1$$
$$\mathbf{y}_2 = X\beta_2 + \epsilon_2$$

SUR

$$\mathbf{y}_1 = X_1\widetilde{\beta}_1 + \epsilon_1$$
$$\mathbf{y}_2 = X_2\widetilde{\beta}_2 + \epsilon_2$$

write the matrix model as write as a vector model by **stacking** the columns of the matrix vertically, using ***Vectorization***. for mvRreg:

$$\underbrace{Y}_{n\times p} = \underbrace{X}_{n\times q}\underbrace{B}_{q\times p} + \underbrace{E}_{n\times p}$$

$\Longleftrightarrow$

$$\mathbf{y}^{(j)} = X\beta^{(j)} + \epsilon^{(j)}, \quad j = 1, \ldots, p$$

where

$$\mathbf{y}^{(j)} = \begin{bmatrix} y_{1j} \\ \vdots \\ y_{nj} \end{bmatrix} \qquad X = \left[\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(j)}\right] = \begin{bmatrix} x_{11} & \cdots & x_{1q} \\ \vdots & \ddots & \vdots \\ x_{n1} & \ldots & x_{nq} \end{bmatrix} \qquad \beta^{(j)} = \begin{bmatrix} \beta_1^{(j)} \\ \vdots \\ \beta_q^{(j)} \end{bmatrix}$$

$$\epsilon^{(j)} = \begin{bmatrix} \epsilon_{1j} \\ \vdots \\ \epsilon_{nj} \end{bmatrix}, \quad \text{where} \qquad \begin{array}{l} \text{cov}(\epsilon_{i_1 j}, \epsilon_{i_2 j}) = 0 \;\; \text{(uncorr property between indiv } i_1 \neq i_2) \\ \text{cov}(\epsilon_{i j_1}, \epsilon_{i j_2}) = \sigma_{j_1 j_2} \neq 0 \;\; \text{(corr properties } j_1, j_2 \text{ within indiv } i) \end{array}$$

To apply the GLS estimator we rewrite p related univariate regression models as one single **stacked** univariate regression model:

$$\begin{bmatrix} \mathbf{y}^{(1)} \\ \vdots \\ \mathbf{y}^{(p)} \end{bmatrix} = \begin{bmatrix} X & & \\ & \ddots & \\ & & X \end{bmatrix} \begin{bmatrix} \beta^{(1)} \\ \vdots \\ \beta^{(p)} \end{bmatrix} + \begin{bmatrix} \epsilon^{(1)} \\ \vdots \\ \epsilon^{(p)} \end{bmatrix}$$

in short (stacked) form

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

for the SUR model we have

$$\begin{bmatrix} \mathbf{y}^{(1)} \\ \vdots \\ \mathbf{y}^{(p)} \end{bmatrix} = \begin{bmatrix} X_1 & & \\ & \ddots & \\ & & X_p \end{bmatrix} \begin{bmatrix} \beta^{(1)} \\ \vdots \\ \beta^{(p)} \end{bmatrix} + \begin{bmatrix} \epsilon^{(1)} \\ \vdots \\ \epsilon^{(p)} \end{bmatrix}$$

next, figure out the covariance matrix $\widetilde{\mathbf{\Sigma}}$ of $\left[\epsilon^{(1)}, \ldots, \epsilon^{(p)}\right]$ and apply the GLS estimator

$$\widehat{\beta} = \left(\widetilde{\mathbf{X}}'\widetilde{\mathbf{\Sigma}}^{-1}\widetilde{\mathbf{X}}\right)^{-1}\widetilde{\mathbf{X}}'\widetilde{\mathbf{\Sigma}}^{-1}\mathbf{y}$$

Theorem: for the mvReg the GLS estimator equals the OLS estimator

$$\widehat{\beta}^{(j)} = \left(X'X\right)^{-1} X'\mathbf{y}^{(j)}$$

$j = 1, \ldots, p$ that means, the multivariate regression model is decomposable into p univariate regression models and ("by unstacking")

$$\widehat{B} = \left(X'X\right)^{-1} X'Y$$

Proof : Covariance structure:

$$\widetilde{\mathbf{\Sigma}} = \begin{bmatrix} \sigma_{11} & \underbrace{0\ldots0}_{n-1} & \sigma_{12} & \\ \vdots & \ddots & 0 & \ddots \\ \sigma_{21} & \ddots & \sigma_{22} & \ddots \\ & \ddots & & \ddots \end{bmatrix} = \begin{bmatrix} \sigma_{11}I & \sigma_{12}I & \\ \sigma_{21}I & \sigma_{22}I & \\ & & \ddots \end{bmatrix} = \Sigma \otimes I$$

use **Kronecker product**: $A = [a_{ij}] \implies A \otimes B = [a_{ij}B]$

write model in Kronecker products:

$$\widetilde{\mathbf{X}} = I \otimes X$$
$$\widetilde{\mathbf{\Sigma}} = \Sigma \otimes I$$

$\implies$ model:

$$\mathbf{y} = (I \otimes X)\,\beta + \epsilon$$
$$\mathbb{V}\epsilon = \Sigma \otimes I$$

$\implies$

$$\widehat{\beta} = \left(\widetilde{\mathbf{X}}'\widetilde{\mathbf{\Sigma}}^{-1}\widetilde{\mathbf{X}}\right)^{-1} \widetilde{\mathbf{X}}'\widetilde{\mathbf{\Sigma}}^{-1}\mathbf{y}$$

rules for Kronecker products:

rule**1**: $\boxed{(A \otimes B)^{-1} = \mathrm{A}^{-1} \otimes \mathrm{B}^{-1}}$

$\Longrightarrow$

$$\widetilde{\mathbf{\Sigma}}^{-1} = \Sigma^{-1} \otimes I = \begin{bmatrix} \sigma^{11} & & \sigma^{12} & \\ & \ddots & & \\ \sigma^{21} & & \sigma^{22} & \\ & & & \ddots \end{bmatrix} = \begin{bmatrix} \sigma^{11} I & \sigma^{12} I & \\ \sigma^{21} I & \sigma^{22} I & \\ & & \ddots \end{bmatrix}$$

rules for Kronecker products:

$$\text{rule}\mathbf{2}: \boxed{(A \otimes B)' = \mathrm{A}' \otimes \mathrm{B}'}$$

$\Longrightarrow$

$$\left.\begin{array}{c} \widetilde{\mathbf{X}} = I \otimes X \\ \widetilde{\mathbf{\Sigma}} = \Sigma \otimes I \\ \widehat{\beta} = \left(\widetilde{\mathbf{X}}' \widetilde{\mathbf{\Sigma}}^{-1} \widetilde{\mathbf{X}}\right)^{-1} \widetilde{\mathbf{X}}' \widetilde{\mathbf{\Sigma}}^{-1} \mathbf{y} \end{array}\right\} \Longrightarrow$$

$$\begin{aligned} \widehat{\beta} &= \left[(I \otimes X)' (\Sigma \otimes I)^{-1} (I \otimes X)\right]^{-1} (I \otimes X)' [\Sigma \otimes I]^{-1} \mathbf{y} \\ &= \left[(I \otimes X') \left(\Sigma^{-1} \otimes I\right) (I \otimes X)\right]^{-1} (I \otimes X)' \left(\Sigma^{-1} \otimes I\right) \mathbf{y} \end{aligned}$$

rules for Kronecker products:

$$\text{rule}\mathbf{3}: \boxed{(A \otimes B)(C \otimes D) = \mathrm{AB} \otimes \mathrm{CD}}$$

$\Longrightarrow$

$$\begin{aligned} \widehat{\beta} &= \left[\left(\Sigma^{-1} \otimes X'\right) (I \otimes X)\right]^{-1} (I \otimes X)' \left(\Sigma^{-1} \otimes I\right) \mathbf{y} && \text{(rule3)} \\ &= \left[\left(\Sigma^{-1} \otimes X'X\right)\right]^{-1} (I \otimes X)' \left(\Sigma^{-1} \otimes I\right) \mathbf{y} && \text{(rule3)} \\ &= \left(\Sigma \otimes (X'X)^{-1}\right) \left(\Sigma^{-1} \otimes X'\right) \mathbf{y} && \text{(rule1, 2)} \\ &= \Sigma\Sigma^{-1} \otimes (X'X)^{-1} X'\mathbf{y} && \text{(rule3)} \\ &= \left(I \otimes (X'X)^{-1} X'\right) \mathbf{y} \end{aligned}$$

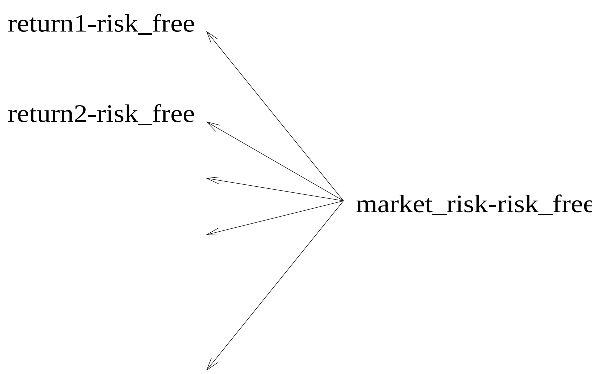
a decoupled system of OLS estimators:

$$\widehat{\beta} = \begin{bmatrix} \widehat{\beta^{(1)}} \\ \vdots \\ \widehat{\beta^{(q)}} \end{bmatrix}$$

qed

Example: CAPM r_{it} : return on stock i at time t. r_{ft} : risk free asset (T-bills) at time t. r_{mt} : market return (stock index) at time t. β_i securities $beta-$coefficient.

$$r_{it} - r_{ft} = \alpha_i + \beta_i \left(r_{mt} - r_{ft}\right) + \epsilon_{it}$$



- It is assumed that the disturbances ϵ_{it} are uncorrelated in time (this is reasonable since they are disturbances of $r_{it} - r_{ft}$, (they are not disturbances of r_{it} which are certainly correlated in (small) time intervals).
- It is not assumed that the disturbances are uncorrelated among stocks (which they are certainly not)
- Under these assumptions it is equally efficient to run separate univaratiate regressions in each stock rather than running one large multivariate regression on the whole portfolio.
- obviously, the estimators in the multivariate regression model and in the separate univariate regression models are equal in this case.

Estimation for SUR

nondecomposable model (can not be broken up into successive one dimensional regressions)

$$\mathbf{y}^{(j)} = X_j \beta^{(j)} + \mathbf{e}^{(j)}, \quad j = 1, \ldots, p$$

where

$$\mathbf{y}^{(j)} = \begin{bmatrix} y_{1j} \\ \vdots \\ y_{nj} \end{bmatrix} \qquad X^{(j)} = \left[\mathbf{x}^{(j;1)}, \ldots, \mathbf{x}^{(j;q_j)}\right] = \begin{bmatrix} x_{11}^{(j)} & \cdots & x_{1q_j}^{(j)} \\ \vdots & \ddots & \vdots \\ x_{n1}^{(j)} & \ldots & x_{nq_j}^{(j)} \end{bmatrix} \qquad \beta^{(j)} = \begin{bmatrix} \beta_1^{(j)} \\ \vdots \\ \beta_{q_j}^{(j)} \end{bmatrix}$$

$$\mathbf{e}^{(j)} = \begin{bmatrix} e_{1j} \\ \vdots \\ e_{nj} \end{bmatrix}, \quad \text{where} \qquad \begin{array}{l} \mathrm{cov}(e_{i_1 j}, e_{i_2 j}) = 0 \;\; \text{(uncorr between indiv)} \\ \mathrm{cov}(e_{i j_1}, e_{i j_2}) \neq 0 \;\; \text{(corr within indiv)} \end{array}$$

Example: *Grunfeld's equation of Investment demand.* I : inverstment, C : capital stock, F : outstanding shares.

$$I(t) = \alpha_0 + \alpha_1 C(t-1) + \alpha_2 F(t-1) + u(t)$$

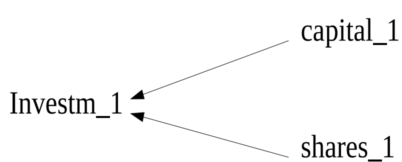
for 2 companies:

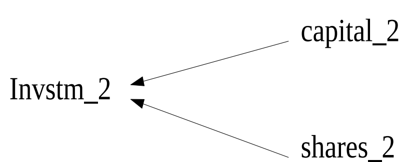
$$\begin{bmatrix} I_1(t) \\ I_2(t) \end{bmatrix} = \begin{bmatrix} 1 & C_1(t-1) & F_1(t-1) & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & C_2(t-1) & F_2(t-1) \end{bmatrix} \begin{bmatrix} \alpha_{01} \\ \alpha_{11} \\ \alpha_{21} \\ \alpha_{02} \\ \alpha_{21} \\ \alpha_{22} \end{bmatrix} + \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

with

$$\mathrm{cov}(u_1, u_2) \neq 0$$

This is a multivariate regression with certain regression coefficients a priori set to zero ("structural zeroes")





$$\begin{bmatrix} I_1^{(1)}(t) \\ \vdots \\ I_n^{(1)}(t) \\ I_1^{(2)}(t) \\ \vdots \\ I_n^{(2)}(t) \end{bmatrix} = \begin{bmatrix} 1 & C_1^{(1)}(t-1) & F_1^{(1)}(t-1) & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & C_n^{(1)}(t-1) & F_n^{(1)}(t-1) & 0 & 0 \\ 1 & 0 & 0 & C_1^{(2)}(t-1) & F_1^{(2)}(t-1) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & C_n^{(2)}(t-1) & F_n^{(2)}(t-1) \end{bmatrix} \begin{bmatrix} \alpha_{01} \\ \alpha_{11} \\ \alpha_{21} \\ \alpha_{02} \\ \alpha_{12} \\ \alpha_{22} \end{bmatrix} + \begin{bmatrix} u_1^{(1)}(t) \\ \vdots \\ u_n^{(1)}(t) \\ u_1^{(2)}(t) \\ \vdots \\ u_n^{(2)}(t) \end{bmatrix}$$

Estimation: iteratively.

1. start with OLS
2. estimate correlation among residual
3. GLS

Alternative solution to SUR estimation: can be considered a multiv reg

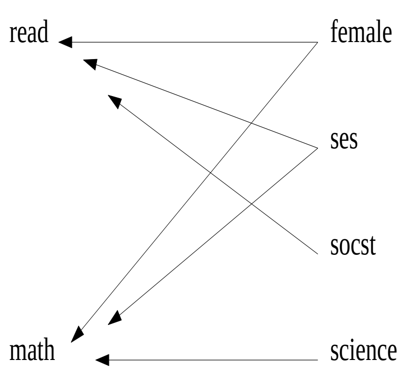
$$Y = XB + E$$

with **structural zeros** of B in certain components. Use **constrained minimization:**

stack

$$\mathbf{y} = \begin{bmatrix} y_1^{(1)} \\ \vdots \\ y_{n_1}^{(n_1)} \\ \vdots \\ y_1^{(p)} \\ \vdots \\ y_{n_p}^{(p)} \end{bmatrix} \qquad X = \begin{bmatrix} X_1 & & \\ & \ddots & \\ & & X_p \end{bmatrix} \qquad B = \begin{bmatrix} \beta_1^{(1)} \\ \vdots \\ \beta_{q_p}^{(p)} \end{bmatrix}$$

Example (cont)



Sur regression

```
sureg (read female ses socst)(math female ses science)
------------------------------------------------------------------------------
        |  Coef.      Std.  Err.     z     P>|z| [95% Conf.  Interval]
---------+--------------------------------------------------------------------
```

read						
female	-1.39969	1.139324	-1.229	0.219	-3.63272	.833343
ses	1.49531	.829882	1.802	0.072	-.13122	3.12181
socst	.51558	.054818	9.405	0.000	.40814	.62302
_cons	24.30038	3.343654	7.268	0.000	17.74694	30.85382
math						
female	1.03163	1.031014	1.001	0.317	-.98912	3.0527
ses	1.65704	.734050	2.257	0.024	.21833	3.0957
science	.50588	.052901	9.563	0.000	.40219	.6095
_cons	21.41615	3.416379	6.269	0.000	14.72017	28.1121

Two separate univ multiple regressions

```
regress read female ses socst
```

read	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
female	-1.51113	1.151079	-1.313	0.191	-3.7812	.758962
ses	1.21837	.839900	1.451	0.148	-.43803	2.874768
socst	.56993	.056830	10.124	0.000	.45890	.680957
_cons	22.19363	3.40042	6.527	0.000	15.48751	28.8997

```
regress math female ses science
```

math	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
female	1.16090	1.041641	1.114	0.266	-.89336	3.2151
ses	1.39964	.742390	1.885	0.061	-.06446	2.8637
science	.57533	.054328	10.590	0.000	.46819	.68247
_cons	18.14428	3.481754	5.211	0.000	11.27777	25.01079

Notes:
The SUR-model approach is more efficient than OLS except when

a. the correlation matrix is close to a diagonal

b. the regressors are highly correlated

moreover,

c. increase in precision of SUR regression dependent on correlation between errors (increasing) and correlation between regressors (decreasing)

d. it can be shown that in some cases the likelihood function is multimodal for small sample sizes (see Drton & Richardson, Biometrika 2004)

Testing SURs: LR-test

$$H_0: \ \beta_1 = \beta_2 = \cdots$$

Application: estimation from aggregated, disaggregated data, **aggregation bias**

$$\begin{aligned} y_1(t) &= \alpha_1 + \beta_1 x_1(t) + \epsilon_1(t) \\ y_2(t) &= \alpha_2 + \beta_2 x_2(t) + \epsilon_2(t) \end{aligned}$$

$\Longrightarrow$

$$\overline{y}(t) = \frac{\beta_1 x_1(t) + \beta_2 x_2(t)}{x_1(t) + x_2(t)} \overline{x}(t) + \alpha_0 + \overline{\epsilon}(t)$$

test for no aggregation bias: H_0: $\beta_1 = \beta_2$.

Notes:

1. we are testing one more general SUR against another (smaller) SUR.
2. Under H_0 is the model $\overline{y}(t) = \beta\overline{x}(t) + \alpha_0 + \overline{\epsilon}(t)$ as **population (macro) model** justified and β can be estimated from the pooled sample. **Linearity on the micro level** is not sufficient to justify linearity on the macro level.
3. As an example conLider $y_i(t) = \alpha_i + \beta_i x_i(t) + \epsilon_i(t)$ on the **household(HH)**-level (micro) and $\overline{y}(t) = \beta\overline{x}(t) + \alpha_0 + \overline{\epsilon}(t)$ on the population (macro) level. Under H_0 we can estimate a macro function from micro-level data
4. one alternative model is to consider $\beta_i = \beta + \delta_i$ with $\delta_1, \delta_2, \ldots$ random with $\mathbb{E}\delta = 0$. (**random coefficient model**, hierarchical model)

Example: Simulation

data generating processes SUR

$$\begin{gathered} y^{(i)} = x^{(i)} + \epsilon^{(i)}, \ i = 1, 2 \\ x^{(1)}, x^{(2)} \text{ iid} \sim \mathcal{N}(0,1) \\ \begin{bmatrix} \epsilon^{(1)} \\ \epsilon^{(2)} \end{bmatrix} \sim \mathcal{N}_2(\mu, \Sigma) \text{ with } \Sigma = \begin{bmatrix} 1 & 0.5 \\ & 1 \end{bmatrix} \end{gathered}$$

model1: SUR

$$y^{(i)} = \alpha^{(i)} + \beta^{(i)} x^{(i)} + \epsilon^{(i)},\ i = 1,2$$
$$\begin{bmatrix} \epsilon^{(1)} \\ \epsilon^{(2)} \end{bmatrix} \sim \mathcal{N}_2\left(\mu, \Sigma\right)$$

model2: indep univariate regressions

$$y^{(i)} = \alpha^{(i)} + \beta^{(i)} x^{(i)} + \epsilon^{(i)},\ i = 1,2$$
$$x^{(i)} \sim \mathcal{N}_1\left(\mu, \sigma_i^2\right),\ \epsilon^{(1)} \perp \epsilon^{(2)}$$

it is seen that the SUR estimates differ in size and standard error from those obtained by fitting separately two univariate regressions.

model3: SUR

$$y^{(i)} = \alpha^{(i)} + \beta_1^{(i)} x^{(1)} + \beta_2^{(i)} x^{(2)} + \epsilon^{(i)},\ i = 1,2$$
$$\begin{bmatrix} \epsilon^{(1)} \\ \epsilon^{(2)} \end{bmatrix} \sim \mathcal{N}_2\left(\mu, \Sigma\right)$$

model4: indep univariate regressions

$$y^{(i)} = \alpha^{(i)} + \beta_1^{(i)} x^{(1)} + \beta_2^{(i)} x^{(2)} + \epsilon^{(i)},\ i = 1,2$$
$$x^{(i)} \sim \mathcal{N}_1\left(\mu, \sigma_i^2\right),\ \epsilon^{(1)} \perp \epsilon^{(2)}$$

model5: mvReg

$$y^{(i)} = \alpha^{(i)} + \beta_1^{(i)} x^{(1)} + \beta_2^{(i)} x^{(2)} + \epsilon^{(i)},\ i = 1,2$$
$$\begin{bmatrix} \epsilon^{(1)} \\ \epsilon^{(2)} \end{bmatrix} \sim \mathcal{N}_2\left(\mu, \Sigma\right)$$

it is seen that estimated parameters and their standard errors are equal in all three models 3),4),5)

```
> #------------------------------------------------------
> #fit bivariate SUR regression model to simulated data
> #compare to two univariate and one multivariate OLS
> #regression models
> #------------------------------------------------------
>
> library(systemfit)
> rm(list=ls())
> set.seed(123)
```

Noise generation

```
> eps1<-rnorm(20); eps2<-eps1+rnorm(20)
```

Model

```
> y1<-(x1<-rnorm(20))+eps1; y2<-(x2<-rnorm(20))+eps2
> print(cor(eps1,eps2))
[1] 0.7352677
```

SUR regression (reduce model with struct zeroes)

```
> eqy1<-y1~x1;  eqy2<-y2~x2
> system<-list(y1=eqy1,y2=eqy2)
>
> fitsur<-systemfit(system,"SUR")
> summary(fitsur)

systemfit results
method: SUR

SUR estimates for 'y1' (equation 1)
Model Formula: y1 ~ x1

            Estimate Std. Error
(Intercept) 0.154621   0.222369
x1          0.877940   0.162171

SUR estimates for 'y2' (equation 2)
Model Formula: y2 ~ x2

            Estimate Std. Error
(Intercept) 0.117323   0.271969
x2          1.224795   0.194997
>
```

Fit separate OLS models (reduced models)

```
> fitols1<-lm(y1~x1)
> fitols2<-lm(y2~x2)
> summary(fitols1)
Coefficients:
            Estimate Std. Error
(Intercept)   0.1552     0.2231
x1            0.8728     0.2376

> summary(fitols2)
Coefficients:
            Estimate Std. Error
(Intercept)   0.1285     0.2731
x2            1.3178     0.2857
>
```

Fit SUR to full model

```
> eqy1<-y1~x1+x2;  eqy2<-y2~x1+x2
> system<-list(y1=eqy1,y2=eqy2)
>
> fitsur<-systemfit(system,"SUR")
> summary(fitsur)
Model Formula: y1 ~ x1 + x2

             Estimate Std. Error
(Intercept) 0.166092   0.229716
x1          0.888700   0.245851
x2          0.105210   0.241868

Model Formula: y2 ~ x1 + x2

              Estimate Std. Error
(Intercept) 0.12797799 0.28231751
x1          0.00566283 0.30214775
x2          1.31867324 0.29725254
>
```

Fit separate OLS (full models)

```
> fitols1<-lm(y1~x1+x2)
> fitols2<-lm(y2~x1+x2)
> summary(fitols1)

Coefficients:
            Estimate Std. Error
(Intercept)   0.1661     0.2297
x1            0.8887     0.2459
x2            0.1052     0.2419

> summary(fitols2)

Coefficients:
            Estimate Std. Error
(Intercept) 0.127978   0.282318
x1          0.005663   0.302148
x2          1.318673   0.297253
>
```

Fit mv OLS full model

```
> y<-cbind(y1,y2)
> out<-lm(y~x1+x2)
> summary(out)
Response y1 :
Coefficients:
            Estimate Std. Error
(Intercept)   0.1661     0.2297
x1            0.8887     0.2459
x2            0.1052     0.2419

Response y2 :
Coefficients:
            Estimate Std. Error
(Intercept) 0.127978   0.282318
x1          0.005663   0.302148
x2          1.318673   0.297253
```