BLUE, GLS, OLS e loro applicazioni

Laboratorio di Metodi Computazionali e Statistici (2023/2024)

R. Cardinale, F. Parodi, S. Passaggio
(con contributo per la conversion in Latex di S. Traverso e A. Sciaccaluga)

November 19, 2023

Combinazione di misure: BLUE

Si consideri il problema di combinare N misure della stessa grandezza Y in modo da determinare $\bar{y}$, miglior stima di Y. In generale le misure possono essere correlate (ad es. possono avere incertezze derivanti da parametri comuni o da una comune procedura di estrazione/analisi).

Si abbiano quindi N misure $y_1, y_2, ..., y_N$ della grandezza Y, il cui valore vero è μ. Come costruisco $\bar{y}$, miglior stima di Y ?

Best **L**inear **U**nbiased **E**stimator

$$\bar{y} = \sum_{i=1}^{N} \alpha_i y_i = \boldsymbol{\alpha y} \qquad \boldsymbol{\alpha} = (\alpha_1, \alpha_2, ..., \alpha_N)$$

- è lineare
- è unbiased: $E[\bar{y}] = \mu$

$$E\left[\sum_{i=1}^{N} \alpha_i y_i\right] = \sum_{i=1}^{N} E[\alpha_i y_i] = \sum_{i=1}^{N} \alpha_i E[y_i] = \sum_{i=1}^{N} \alpha_i \mu = \mu \quad \Rightarrow \quad \sum_{i=1}^{N} \alpha_i = 1$$

- ha varianza $\sigma^2_{\bar{y}}(\boldsymbol{\alpha})$ minima (best)

BLUE: minima varianza

Scriviamo la varianza in termini degli α_i

$$\sigma^2(\bar{y}) = Cov[\bar{y}, \bar{y}] = Cov\left[\sum_{i=1}^{N} \alpha_i y_i, \sum_{j=1}^{N} \alpha_j y_j\right]$$

$$= \sum_{i,j=1}^{N} \alpha_i \alpha_j Cov\left[y_i, y_j\right] = \sum_{i,j=1}^{N} \alpha_i \alpha_j V_{ij}$$

che corrisponde, in termini matriciali, a:

$$\sigma^2(\bar{y}) = \boldsymbol{\alpha} V \boldsymbol{\alpha}^T$$

La richiesta di varianza minima determina $\boldsymbol{\alpha}$ e, di conseguenza, l'estimatore migliore.

BLUE: caso 2D

Consideriamo il caso di due sole misure

$$V = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix} \qquad \boldsymbol{\alpha} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix} \xRightarrow{\sum_i \alpha_i = 1} \boldsymbol{\alpha} = \begin{pmatrix} \alpha_1 \\ 1-\alpha_1 \end{pmatrix}$$

$$\sigma_{\bar{y}}^2(\alpha_1) = \begin{pmatrix} \alpha_1 & 1-\alpha_1 \end{pmatrix} \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ 1-\alpha_1 \end{pmatrix}$$

$$\sigma_{\bar{y}}^2(\alpha_1) = \sigma_1^2\alpha_1^2 + 2\rho\sigma_1\sigma_2\alpha_1(1-\alpha_1) + \sigma_2^2(1-\alpha_1)^2$$

Imponiamo varianza minima:

$$\frac{d\sigma_{\bar{y}}^2(\alpha_1)}{d\alpha_1} = 2\sigma_1^2\alpha_1 + 2\rho\sigma_1\sigma_2(1-2\alpha_1) - 2\sigma_2^2(1-\alpha_1) = 0$$

Ottenendo i valori di α_1 e α_2 (e di conseguenza $\sigma^2(\bar{y})$) per cui la varianza è minima $(\hat{\alpha_1}, \hat{\alpha_2})$:

$$\hat{\alpha}_1 = \frac{\sigma_2^2 - \rho\sigma_1\sigma_2}{\sigma_1^2 - 2\rho\sigma_1\sigma_2 + \sigma_2^2} \qquad \hat{\alpha}_2 = \frac{\sigma_1^2 - \rho\sigma_1\sigma_2}{\sigma_1^2 - 2\rho\sigma_1\sigma_2 + \sigma_2^2}$$

$$\sigma_{\bar{y}}^2 = \frac{(1-\rho^2)\sigma_1^2\sigma_2^2}{\sigma_1^2 - 2\rho\sigma_1\sigma_2 + \sigma_2^2}$$

BLUE: caso 2D

Verifichiamo il risultato per σ_y^2

$$
\begin{aligned}
\sigma_y^2(\hat{\alpha}_1) &= \sigma_1^2\hat{\alpha}_1^2 + 2\rho\sigma_1\sigma_2\hat{\alpha}_1(1-\hat{\alpha}_1) + \sigma_2^2(1-\hat{\alpha}_1)^2 \\
&= \frac{\sigma_1^2(\sigma_2^2-\rho\sigma_1\sigma_2)^2}{(\dots)} + \frac{2\rho\sigma_1\sigma_2(\sigma_2^2-\rho\sigma_1\sigma_2)(\sigma_1^2-\rho\sigma_1\sigma_2)}{(\dots)} + \frac{\sigma_2^2(\sigma_1^2-\rho\sigma_1\sigma_2)^2}{(\dots)} \\
&= \sigma_1^2\sigma_2^2\left[\frac{(\sigma_2-\rho\sigma_1)^2 + 2\rho(\sigma_2-\rho\sigma_1)(\sigma_1-\rho\sigma_2) + (\sigma_1-\rho\sigma_2)^2}{(\dots)}\right] \\
&= \sigma_1^2\sigma_2^2\left[\frac{\sigma_2^2 - 2\rho\sigma_1\sigma_2 + \rho^2\sigma_1^2 + 2\rho(\sigma_1\sigma_2 - \rho\sigma_1^2 - \rho\sigma_2^2 + \rho^2\sigma_1\sigma_2) + \sigma_1^2 - 2\rho\sigma_1\sigma_2 + \rho^2\sigma_2^2}{(\dots)}\right] \\
&= \sigma_1^2\sigma_2^2\left[\frac{\sigma_2^2 - 2\rho\sigma_1\sigma_2 + \sigma_1^2 - \rho^2(2\sigma_1^2 - \sigma_2^2 + 2\sigma_1^2 - \sigma_2^2 - 2\rho\sigma_1\sigma_2)}{(\dots)}\right] \\
&= \sigma_1^2\sigma_2^2\left[\frac{(1-\rho^2)(\sigma_1^2+\sigma_2^2-2\rho\sigma_1\sigma_2)}{(\sigma_1^2-2\rho\sigma_1\sigma_2+\sigma_2^2)^2}\right] \\
&= \frac{\sigma_1^2\sigma_2^2(1-\rho^2)}{\sigma_1^2-2\rho\sigma_1\sigma_2+\sigma_2^2}
\end{aligned}
$$

BLUE: caso 2D con misure sono non correlate ($\rho = 0$)

Se le misure non sono correlate si ha $\rho = 0$ e quindi:

$$\bar{y} = \hat{\alpha}_1 y_1 + \hat{\alpha}_2 y_2 = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2} y_1 + \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2} y_2$$

$$\sigma_{\bar{y}}^2(\hat{\alpha}_1) = \frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2 + \sigma_2^2}$$

Dividendo numeratore e denominatore di $\bar{y}$ e σ^2 per $\sigma_1^2 \sigma_2^2$ si ottiene

$$\bar{y} = \frac{\frac{y_1}{\sigma_1^2}}{\frac{1}{{\sigma_1}^2} + \frac{1}{{\sigma_2}^2}} + \frac{\frac{y_2}{\sigma_2^2}}{\frac{1}{{\sigma_1}^2} + \frac{1}{{\sigma_2}^2}} \qquad \sigma_{\bar{y}}^2 = \frac{1}{\frac{1}{{\sigma_1}^2} + \frac{1}{{\sigma_2}^2}}$$

che sono le note formule della media pesata.

BLUE: caso 2D con misure correlate ($\rho \neq 0$)

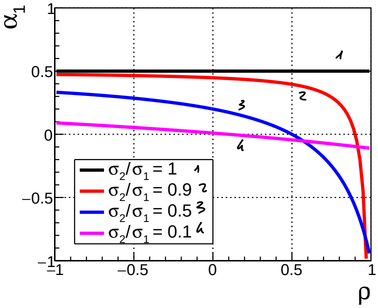
Per fissare le idee consideriamo $\sigma_2 \leq \sigma_1$ e scriviamo tutto in funzione del rapporto $\eta = \sigma_2/\sigma_1$ (dividendo numeratore e denominatore per σ_1^2):

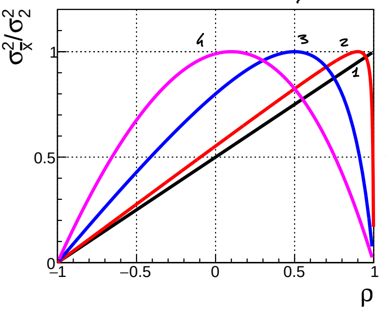
$$\hat{\alpha}_1 = \frac{\eta^2 - \rho\eta}{1 - 2\rho\eta + \eta^2} \qquad \hat{\alpha}_2 = \frac{1 - \rho\eta}{1 - 2\rho\eta + \eta^2}$$

$$\frac{\sigma_{\bar{y}}^2}{\sigma_2^2} = \frac{(1-\rho^2)}{1 - 2\rho\eta + \eta^2}$$

Studiamo l'andamento di $\hat{\alpha}_1$ e $\sigma_{\bar{y}}^2/\sigma_1^2$ in funzione di ρ

se $\rho \in [0{,}5, 1]$
miglior stima sta
fuori dall'intervallo
fra le due misure





paradossale

miglior
stima
sta
dx y_2
xk $\sigma_2 < \sigma_1$

BLUE: caso 2D con misure correlate ($\rho \neq 0$)

Si nota che:

- per $\eta = 1$ cioè $\sigma_2 = \sigma_1$ il coefficiente α_1 è sempre 0.5 (come si può anche verificare esplicitamente per sostituzione) come nel caso non correlato, ma l'errore dipende da ρ
- per $\rho > \sigma_2/\sigma_1$ α_1 diventa negativo e, di conseguenza, α_2 maggiore di uno.

In particolare, quindi, per $\rho > \sigma_2/\sigma_1$:

$$\bar{y} = \alpha_1 y_1 + (1 - \alpha_1) y_2 = y_2 + (y_2 - y_1)|\alpha_1| > y_2$$

Per spiegare questo apparente paradosso analizziamo le due misure: sono molto correlate ($\rho > \sigma_2/\sigma_1$), la misura 2 è più precisa della misura 1. Come si può produrre sperimentalmente la configurazione osservata per cui $y_1 < y_2$? Avviene solo se le due misure hanno deviato dalla stessa parte del valore vero (correlazione positiva) di una quantità proporzionale alla loro incertezza. In questa situazione quindi il valore vero, di cui $\bar{y}$ è un estimatore, deve stare lontano da entrambe (nella stessa direzione), più vicino alla misura più precisa.

Quindi se, $\sigma_2 < \sigma_1$ e $\rho > \sigma_2/\sigma_1$, come nel caso in questione:

analogamente se, $\sigma_1 < \sigma_2$ e $\rho > \sigma_1/\sigma_2$:

BLUE: caso ND

Per la soluzione di BLUE in n dimensione è utile fare uso di uno strumento generale per la ricerca degli estremi di una funzione in presenza di vincoli: i **moltiplicatori di Lagrange**.

Data la funzione f di n incognite e m vincoli (g_i) con $m < n$

$$\begin{cases} f(x_1, x_2, ..., x_n) & \\ g_i(x_1, x_2, ..., x_n) & \text{con i=1}, ..., m \end{cases}$$

esistono λ_i (con $i = 1, ..., m$) tali che:

$$\boldsymbol{\nabla} \left(f - \sum_{i=1}^{m} \lambda_i g_i \right) = 0$$

in ogni estremo vincolato.

BLUE: caso ND

la f da minimizzare é
la varianza dello stimatore

Nel caso di BLUE:

$$\sum_{i=1}^{N} \alpha_i = 1 \Leftarrow \begin{cases} \sigma_Y^2 = \boldsymbol{\alpha} V \boldsymbol{\alpha}^T & \text{funzione da minimizzare} \\ \underset{\text{vettore riga}}{\boldsymbol{\alpha}} \underset{\text{colonna}}{\boldsymbol{u}} = 1 & \text{vincolo con } \boldsymbol{u} = 1, ..., 1 \end{cases}$$

generale

il vincolo

Applicando il metodo dei moltiplicatori di Lagrange (come mostrato in Appendice A) si ricavano le formule risolutive generali

$$\hat{Y} = \boldsymbol{\alpha} Y = \frac{\boldsymbol{u}^T V^{-1} \boldsymbol{Y}}{\boldsymbol{u}^T V^{-1} \boldsymbol{u}}$$
$$\sigma_{\hat{Y}}^2 = (\boldsymbol{u}^T V^{-1} \boldsymbol{u})^{-1}$$

Takeaway messages:

- La media di misure correlate richiede una precisa determinazione della matrice di covarianza e, in particolare, del coefficiente di correlazione ρ
- Non c'è in generale un'assunzione conservativa (ad esempio 100% correlazione)

BLUE vs GLS/OLS

Vogliamo risolvere il problema della media di misure correlate con il metodo generale dei minimi quadrati (General Least Squares).
Impostiamo il problema matriciale in dimensione generica. N misure
Sia:

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}; \qquad E[y_i] = \mu \quad \forall i = 1, \ldots, N; \qquad \boldsymbol{\mu} = \begin{bmatrix} \mu \\ \vdots \\ \mu \end{bmatrix}$$

Sia inoltre V la matrice di convarianza (simmetrica, definita positiva): $V_{ij} = Cov[y_i, y_j]$. Definiamo allora:

$$S(\mu) \equiv \sum_{i,j=1}^{N} (y_i - \mu) V_{ij}^{-1} (y_j - \mu) = (\boldsymbol{y} - \boldsymbol{\mu})^T V^{-1} (\boldsymbol{y} - \boldsymbol{\mu})$$

La precedente nel caso di V diagonale diventa ($V_{ii}^{-1} = 1/\sigma_i^2$) (Ordinary Least Squares):

$$S(\mu) = \sum_{i=1}^{N} \frac{(y_i - \mu)^2}{\sigma_i^2}$$

BLUE vs GLS/OLS

Ora torniamo al caso $N = 2$:

$$V = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix} \implies \det V = \sigma_1^2\sigma_2^2(1-\rho^2)$$

Allora sappiamo che:

$$V^{-1} = \frac{1}{\det V} \begin{bmatrix} \sigma_2^2 & -\rho\sigma_1\sigma_2 \\ -\rho\sigma_1\sigma_2 & \sigma_1^2 \end{bmatrix} = \frac{1}{1-\rho^2} \begin{bmatrix} \frac{1}{\sigma_1^2} & -\frac{\rho}{\sigma_1\sigma_2} \\ -\frac{\rho}{\sigma_1\sigma_2} & \frac{1}{\sigma_2^2} \end{bmatrix}$$

Quindi sostituendo nella definizione di S troviamo:

$$S(\mu) = \frac{1}{1-\rho^2} \left[\frac{(y_1-\mu)^2}{\sigma_1^2} - 2\rho\frac{(y_1-\mu)(y_2-\mu)}{\sigma_1\sigma_2} + \frac{(y_2-\mu)^2}{\sigma_2^2} \right]$$

BLUE vs GLS/OLS

Chiamiamo $\hat{\mu}$ il valore per cui $S(\hat{\mu}) \leq S(\mu)\ \forall \mu$.

$$\begin{aligned}
\frac{dS}{d\mu} &= -\frac{2}{1-\rho^2}\left[\frac{(y_1-\mu)}{\sigma_1^2} - \rho\frac{(y_2-\mu)}{\sigma_1\sigma_2} - \rho\frac{(y_1-\mu)}{\sigma_1\sigma_2} + \frac{(y_2-\mu)}{\sigma_2^2}\right] \\
&= -\frac{2}{1-\rho^2}\left[\frac{\sigma_2^2 y_1 - \sigma_2^2\mu - \rho\sigma_1\sigma_2(y_1+y_2) + 2\rho\sigma_1\sigma_2\mu + \sigma_1^2 y_2 - \sigma_1^2\mu}{\sigma_1^2\sigma_2^2}\right] \\
&= \frac{2}{\sigma_1^2\sigma_2^2(1-\rho^2)}\left[(\sigma_1^2 - 2\rho\sigma_1\sigma_2 + \sigma_2^2)\mu - (\sigma_2^2 y_1 - \rho\sigma_1\sigma_2(y_1+y_2) + \sigma_1^2 y_2)\right] = 0
\end{aligned}$$

Dalla precedente, raccogliendo $\frac{1}{\sigma_1^2\sigma_2^2}$ e osservando che il fattore moltiplicativo non può essere nullo, si trova:

$$\bar{y} = \hat{\mu} = \frac{\sigma_2^2 y_1 - \rho\sigma_1\sigma_2(y_1+y_2) + \sigma_1^2 y_2}{\sigma_1^2 - 2\rho\sigma_1\sigma_2 + \sigma_2^2}$$

La soluzione coincide con quanto trovato con il metodo BLUE.

Si noti come la minimizzazione sia stata effettuata rispetto a μ (valore vero, stimato dalla media) e non rispetto ai coefficienti α_i degli y_i.

GLS: proprietà del minimo

Si noti, nel caso lineare 1D, la derivata seconda di S rispetto al parametro da stimare sia direttamente collegata alla varianza di $\hat{\mu}$ dedotta con BLUE:

$$\frac{d^2S}{d\mu^2} = \frac{2(\sigma_1^2 - 2\rho\sigma_1\sigma_2 + \sigma_2^2)}{\sigma_1^2\sigma_2^2(1-\rho^2)} = \frac{2}{\sigma_{\hat{\mu}}^2}$$

Se infine sviluppiamo S intorno al minimo:

$$S(\mu) = S(\hat{\mu}) + \left.\frac{dS}{d\mu}\right|_{\hat{\mu}}(\mu-\hat{\mu}) + \frac{1}{2}\left.\frac{d^2S}{d\mu^2}\right|_{\hat{\mu}}(\mu-\hat{\mu})^2 = S(\hat{\mu}) + \frac{(\mu-\hat{\mu})^2}{\sigma_{\hat{\mu}}^2}$$

In altri termini calcolando la variazione di ascissa rispetto al minimo che produce una variazione di S di 1, determino $\sigma_{\hat{\mu}}$.

Caso lineare generale: GLS

Consideriamo n misure (x_i, y_i) con $i = 1, \dots, n$ e assumiamo che:

- le y_i siano unbiased, ovvero $E[y_i] = \mu_i$
- μ_i (i "valori veri") siano esprimibili come espressione lineare nei parametri da stimare β_j $(j = 1, \dots, p)$
- le misure x_i siano note esattezza
- le misure y_i siano affette da errore e sia nota la loro matrice di covarianza completa.

Possibili espressioni per la relazione tra i μ_i e i β_i sono:

$$\mu_i = \beta_0 + \beta_1 x_i$$

$$\mu_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2$$

$$\mu_i = \beta_0 + \beta_1 g(x_i) + \beta_2 h(x_i)$$

$\beta_0 = \mu = \mu_i$

GLS

Siccome l'unica ipotesi che abbiamo fatto è la relazione con i parametri sia lineare, generalizzando si può scrivere:

$$\boxed{\boldsymbol{\mu} = X\boldsymbol{\beta}}$$

dove $\boldsymbol{\mu}$ è $n \times 1$, X è $n \times p$, e $\boldsymbol{\beta}$ è $p \times 1$.
Quindi se ora riprendiamo l'espressione di S vista nella sezione precedente:

$$S(\boldsymbol{\beta}) \equiv (\boldsymbol{y} - X\boldsymbol{\beta})^T V^{-1} (\boldsymbol{y} - X\boldsymbol{\beta})$$

Si può dimostrare (vedi Appendice B) che la minimizzazione di S produce:

$$X = \begin{pmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \\ 1 & x_n \end{pmatrix} \qquad \mu_i = \beta_0 + \beta_1 x_i$$

$$\hat{\boldsymbol{\beta}} = \underbrace{(X^T V^{-1} X)^{-1} X^T V^{-1}}_{p \times n} \boldsymbol{y}$$

$$V_{\hat{\boldsymbol{\beta}}} = (X^T V^{-1} X)^{-1}$$

oppure

$$x = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} = \mu \qquad p = 1$$

GLS: Gauss-Markov

Il teorema di Gauss-Markov afferma che:

> "L'estimatore GLS ha varianza minima nella classe degli estimatori lineari, unbiased"

- Non c'è alcuna assunzione sulle distribuzioni di probabilità delle misure (non è necessario che siano gaussiane né che siano identiche per tutti gli y_i).
- L'assunzione di unbiaseness non è invece eliminabile (esistono infatti estimatori biased on varianza minore)

In gergo quindi si può dire che GLS è BLUE.

GLS in 2D senza correlazioni (OLS)

Applicando le formule trovate e ricordando che in 2D:

$$X^T = \begin{pmatrix} 1 & 1 \\ x_0 & x_1 \end{pmatrix} \qquad \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} \qquad V^{-1} = \begin{pmatrix} \frac{1}{\sigma_1^2} & 0 \\ 0 & \frac{1}{\sigma_2^2} \end{pmatrix}$$

e quindi:

$$X^T V^{-1} = \begin{pmatrix} \frac{1}{\sigma_1^2} & \frac{1}{\sigma_1^2} \\ \frac{x_1}{\sigma_1^2} & \frac{x_2}{\sigma_1^2} \end{pmatrix}$$

$$X^T V^{-1} X = \begin{pmatrix} \frac{1}{\sigma_1^2} & \frac{1}{\sigma_1^2} \\ \frac{x_1}{\sigma_1^2} & \frac{x_2}{\sigma_1^2} \end{pmatrix} \begin{pmatrix} 1 & x_1 \\ 1 & x_2 \end{pmatrix} = \begin{pmatrix} [1] & [x] \\ [x] & [x^2] \end{pmatrix}$$

dove con $[x]$ si indica $\sum_i x_i/\sigma_i^2$

GLS in 2D senza correlazioni (OLS)

La matrice di covarianza risulta

$$V_{\hat{\boldsymbol{\beta}}} = \frac{1}{[1][x^2]-[x][x]} \begin{pmatrix} [x^2] & -[x] \\ -[x] & [1] \end{pmatrix}$$

e i valori dei parametri:

$$\begin{aligned}
\hat{\boldsymbol{\beta}} &= \\
&= (X^T V^{-1} X)^{-1} X^T V^{-1} \boldsymbol{y} \\
&= \frac{1}{[1][x^2]-[x][x]} \begin{pmatrix} [x^2] & -[x] \\ -[x] & [1] \end{pmatrix} \begin{pmatrix} \frac{1}{\sigma_1^2} & \frac{1}{\sigma_2^2} \\ \frac{x_1}{\sigma_1^2} & \frac{x_2}{\sigma_2^2} \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \\
&= \frac{1}{[1][x^2]-[x][x]} \begin{pmatrix} [x^2]\frac{1}{\sigma_1^2} - [x]\frac{x_1}{\sigma_1^2} & [x^2]\frac{1}{\sigma_2^2} - [x]\frac{x_2}{\sigma_2^2} \\ -[x]\frac{1}{\sigma_1^2} + [1]\frac{x_1}{\sigma_1^2} & -[x]\frac{1}{\sigma_2^2} + [1]\frac{x_2}{\sigma_2^2} \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \\
&= \frac{1}{[1][x^2]-[x][x]} \begin{pmatrix} [x^2][y] - [x][xy] \\ -[x][y] + [1][xy] \end{pmatrix}
\end{aligned}$$

che sono i noti risultati dei minimi quadrati in caso di funzione di fit lineare, senza errore sulle x

GLS in 2D senza correlazioni (OLS)

Con i risulati noti si può calcolare la varianza di y per qualsiasi valore di x. Questa informazione permette, ad esempio, di disegnare la banda di incertezza sulla funzione fittata.

$$y = \beta_0 + \beta_1 x$$

$$\sigma_y^2 = \sum_{i,j} \frac{df}{d\beta_i} \frac{df}{d\beta_j} V_{\boldsymbol{\beta}}$$

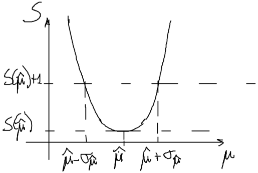
$$\sigma_y^2 = \begin{bmatrix} 1 & x \end{bmatrix} V_{\boldsymbol{\beta}} \begin{bmatrix} 1 \\ x \end{bmatrix} = \begin{bmatrix} \sigma_1^2 + x\rho\sigma_0\sigma_1 & \rho\sigma_0\sigma_1 + x\sigma_1^2 \end{bmatrix} \begin{bmatrix} 1 \\ x \end{bmatrix} = \sigma_0^2 + 2x\rho\sigma_0\sigma_1 + \sigma_1^2 x^2$$

Se il modello di fit non è lineare dei parametri si deve ricorrere alla minimizzazione numerica con l'ausilio di programmi come, ad esempio, MINUIT.

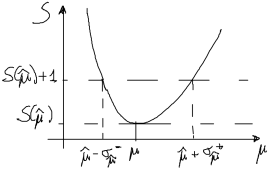
Scelta dei valori iniziali dei parametri è molto rilevante per assicurarsi che il fit converga alla soluzione corretta.

In generale inoltre l'andamento di S intorno al minimo non è parabolico (lo è solo nel caso lineare). Il metodo $\Delta S = 1$ per il calcolo dell'errore seleziona, in generale, errori asimmetrici

Caso lineare



Caso non lineare



Fit di Istogrammi

Consideriamo un istogramma con k bin e n_i conteggi nel bin i, sia $\nu_i(\beta)$. il numero aspettato di conteggi in funzione del vettore di parametri $\boldsymbol{\beta}$. Se ν_i o n_i (a seconda del metodo) non sono troppo piccoli si può utilizzare l'approssimazione gaussiana della distribuzione di Poisson e applicare il metodo dei minimi quadrati.

χ^2 di Pearson:

$$S(\boldsymbol{\beta}) = \sum_{i=1}^{k} \frac{(n_i - \nu_i(\boldsymbol{\beta}))^2}{\nu_i(\boldsymbol{\beta})}$$

χ^2 di Neyman:

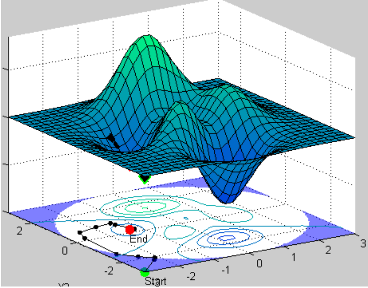
$$S(\boldsymbol{\beta}) = \sum_{i=1}^{k} \frac{(n_i - \nu_i(\boldsymbol{\beta}))^2}{n_i}$$

L'approssimazione si assume valida se tutti gli n_i (o ν_i) sono maggiori di 5.

Una nota sulla nomenclatura: come vedremo in seguito la funzione $S(\boldsymbol{\beta})$ al minimo segue la distribuzioni di χ^2 con $k - n$ gradi di libertà (dove n è il numero di componenti di $\boldsymbol{\beta}$). Da qui la consuetudine di usare l'espressione χ^2 anche per S. Nel corso manterremo invece le due nomenclature distinte.

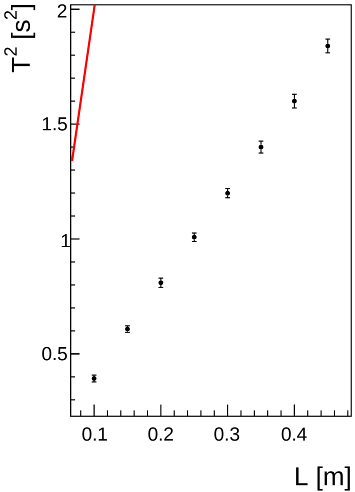
Minimizzazione numerica in ROOT

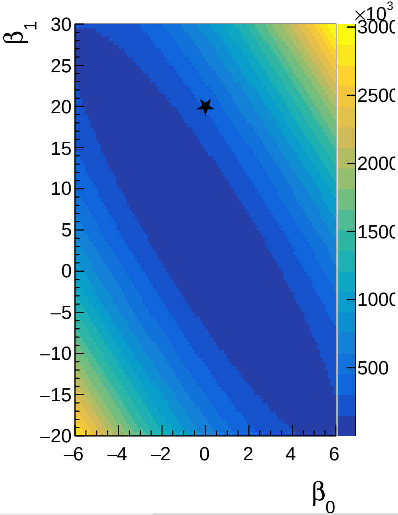
- Per la minimizzazione numerica del χ^2 useremo (ed in parte avete già usato) gli strumenti che ROOT fornisce.
  - In particolare le classi TGraph, TGraphErrors e tutte le classi di istogrammi (TH1X) sono dotate di un metodo
    ```
    void Fit(const char* fun);
    ```
    dove fun è il nome della TF1 che si vuole utilizzare funzione di fit.
- Elemento chiave di questa minimizzazione è il valore iniziale dei parametri
  - Occorre anche avere una stima ragionevole dei valori dei parametri che sono presenti nella funzione.
  - Infatti i programmi di minimizzazione hanno bisogno di un punto di partenza (punto nello spazio dei parametri).
  - Se questo punto di partenza è troppo distante dal minimo fisico la minimizzazione può selezionare un minimo relativo non fisico.



Visualizzazione delle minimizzazione per il fit di una retta

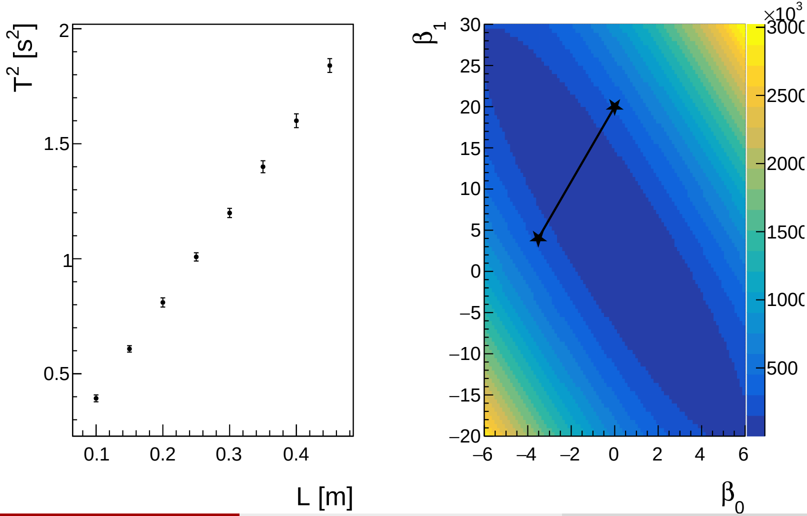
Classico esempio del fit del pendolo (T^2 vs L).





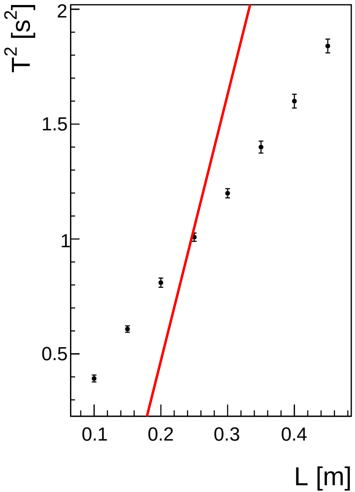
Visualizzazione delle minimizzazione per il fit di una retta

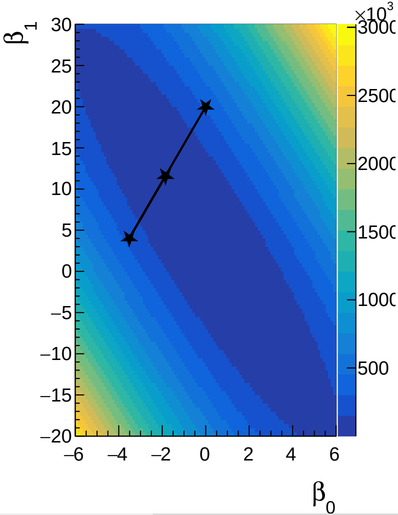
Classico esempio del fit del pendolo (T^2 vs L).



Visualizzazione delle minimizzazione per il fit di una retta

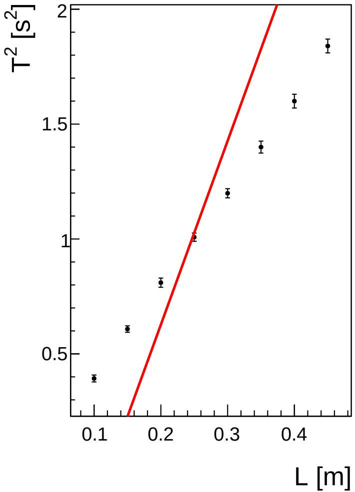
Classico esempio del fit del pendolo (T^2 vs L).

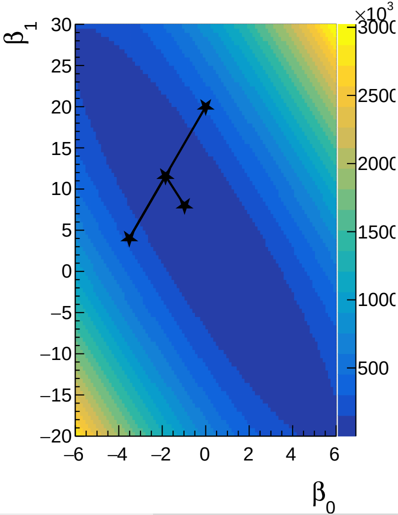




Visualizzazione delle minimizzazione per il fit di una retta

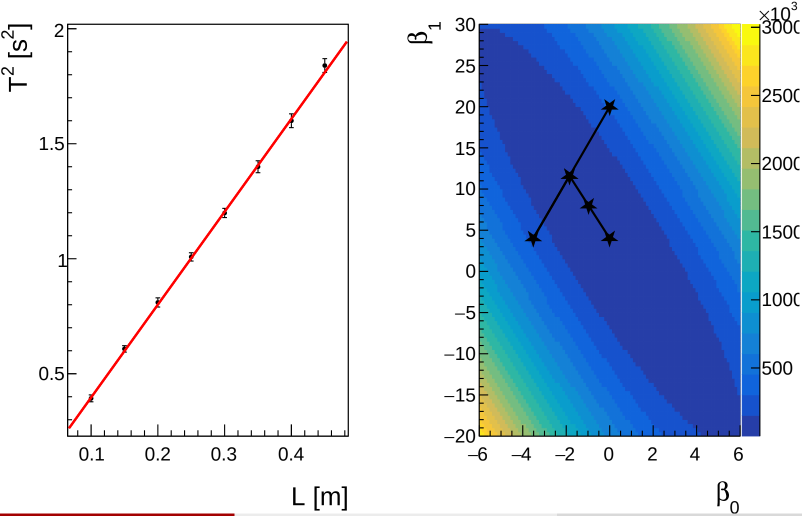
Classico esempio del fit del pendolo (T^2 vs L).



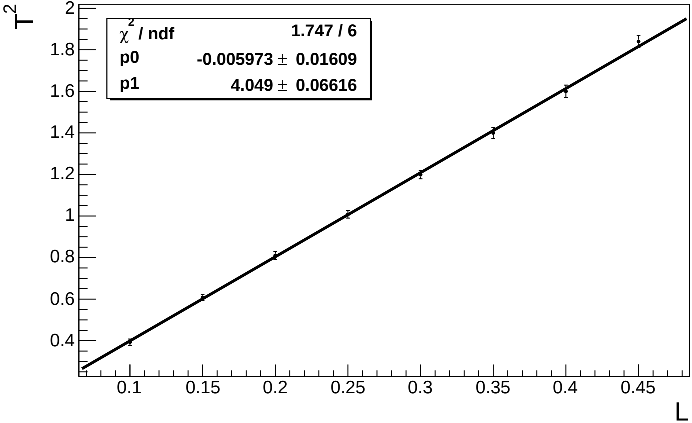


Visualizzazione delle minimizzazione per il fit di una retta

Classico esempio del fit del pendolo (T^2 vs L).



Visualizzazione dei risultati




```
FCN=1.74719 FROM MIGRAD    STATUS=CONVERGED      42 CALLS          43 TOTAL
                            EDM=9.34034e-09    STRATEGY= 1      ERROR MATRIX ACCURATE
EXT  PARAMETER                                     STEP              FIRST
NO.   NAME       VALUE              ERROR          SIZE            DERIVATIVE
  1     p0       -5.97342e-03   1.60871e-02     5.54236e-06      -1.99465e-02
  2     p1        4.04874e+00   6.61605e-02     2.27938e-05      -4.45864e-03
```

Fit con definizione autonoma di S (o di $-log\mathcal{L}$)

- In molti casi è utile poter definire autonomamente la funzione (S o $-log\mathcal{L}$) da minimizzare. Le motivazioni possono essere molteplici:
  - la funzione di fit non è analitica (ad esempio ha componenti che possono essere calcolate solo numericamente);
  - il fit è complesso, ad esempio si vuole descrivere i dati con più funzioni contemporaneamente: fit "simultanei";
  - si desidera avere un maggior controllo del processo di minimizzazione.
- Nel seguito mostreremo, per semplicità, gli strumenti di minimizzazione per il caso lineare.

Minuit

- Minuit è concepito come uno strumento per trovare il minimo di una funzione multiparametrica ed analizzarne la forma intorno al minimo stesso.
- È particolarmente adatto a gestire problemi difficili, compresi quelli che necessitano di una guida per trovare la soluzione corretta.
- Scritto inizialmente in fortran (CERNLIB) è stato tradotto in C++ e inglobato in ROOT. La classe ROOT che implementa MINUIT è TMinuit
- Anche se non è instanziato esplicitamente, TMinuit viene instanziato implicitamente dal metodo Fit. Il puntatore (o oggetto in python)

  `gMinuit`

  è un puntatore (oggetto) globale all'ultimo oggetto TMinuit istanziato.

Definizione χ^2 (C++)

```
namespace data{
  vector<double>  x, y, ex, ey;
}

double fun(const double *x,const double *par){
  return par[1]*(*x)+par[0];
}

void fcn(int &npar,double *gin,double &f,double *par,int iflag){
  f = 0.0;
  for (int i=0;i<data::x.size();i++){
    // Calcolo del Chi2
  }
}
```

- L'utente deve sempre fornire una funzione con questo prototipo
- Significato dei parametri in input/output:
  - `npar` numero di parametri (input)
  - `gin` vettore delle derivate prime della funzione nei parametri (opzionale, output)
  - `f` valore della funzione (output)
  - `par` vettori di parametri (costanti o variabili) (input)
  - `iflag` intero che indica lo stadio di minimizzazione (input)

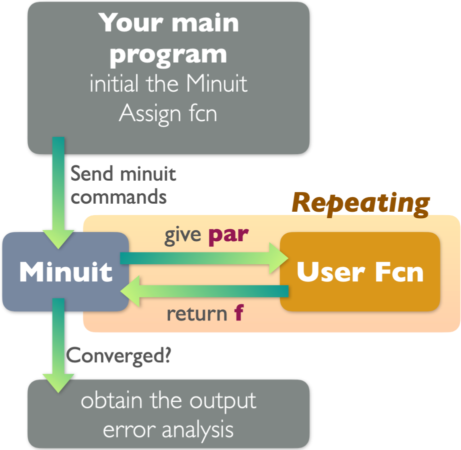
Per iniziare è sufficiente definire `f` sulla base di `par`.

Definizione χ^2 (C++)

```
void fitlin(){
  ifstream file("pendolo.dat");
  double x,y,ex,ey;
  while (file >> x >> y >> ex >> ey){
    data::x.push_back(x); data::y.push_back(y); data::ex.push_back(ex); data::ey.push_back(ey);
  }

  // Define the minimization problem
  // TMinuit *minuit = new TMinuit(...); // numero di parametri
  minuit->SetFCN(fcn);
  // minuit->DefineParameter(indice,"nome",valin,step,min,max);
  // per ogni indice inserisco nome, valore iniziale, step, minimo, massimo del parametro

  // Minimize
  minuit->Command("MIGRAD"); // Comando di minimizzazione

  // Get result
  // minuit->GetParameter(indice,val,eval);
  // per ogni indice estraggo il valore del parametro e del suo errore
  // minuit->GetParameter(indice,val,eval);

}
```

Minuit workflow

- Il programma principale deve inizializzare la classe TMinuit e fornire la funzione da minimizzare
- Si definiscono i parametri (variabili o costanti)
- Si danno comandi a Minuit che procede alla minimizzazione chiamando la funzione fornita.



Opzioni di minimizzazione di Minuit

Comandi in forma di stringhe che possono essere passati al metodo `Command()` (i caratteri minuscoli individuano le lettere che possono essere emesse):

- `MIGrad`: è il metodo di minimizzazione più generale, adatto alla maggior parte degli scopi. È basato sulla ricerca del minimo a partire da un set di valori iniziali sfruttando il criterio del massimo gradiente (pendenza maggiore verso il minimo).
- `HESse`: calcola la matrice degli errori dei parametri (o dei pesi) che corrisponde all'inverso della matrice di covarianza. Il calcolo viene eseguito al minimo (per questo il metodo va chiamato dopo MIGRAD) tramite il calcolo delle derivate seconde con il metodo delle differenze finite. In alcuni casi viene automaticamente chiamata da MIGRAD. Conviene comunque chiamarlo esplicitamente se si desidera una stima precisa degli errori. Si noti che, per sua natura, sfrutta l'approssimazione parabolica per il minimo.
- `MINOs`: calcola la matrice degli errori (eventualemnte asimmetrici) tramite il metodo ΔS o $\Delta log\mathcal{L}$. Deve essere chiamato dopo che il minimo è stato trovato (quindi dopo MIGRAD).

Metodi di Minuit

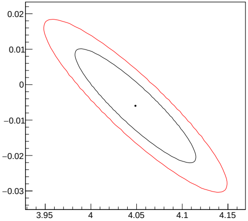
Metodi rilevanti della classe Minuit (oltre a quelli mostrati nell'esempio)

- `SetFCN(fun)`
  definisce la funzione di least squares (S) o di $log\mathcal{L}$
- `SetErrorDef(up)`
  definisce ΔS o $\Delta log\mathcal{L}$ per il problema specifico (attenzione: va comunque fornito anche se non si userà il metodo grafico per il calcolo dell'errore)
- `Command(string)`
  passa a Minuit un comando definito (vedi slides precedente)
- `Contour(npoints,pa1,pa2)`
  ritorna il grafico composto dagli `N` punti del "contorno" (ΔS o $\Delta log\mathcal{L} =$ up) nello spazio dei parametri `pa1,pa2`

  ```
  TObject* Contour (int npoints=10, int pa1=0, int  pa2=1)
  ```

Contour

- Il $\Delta\chi^2$ rispetto al minimo segue la distribuzione del χ^2, quindi ha valori diversi se calcolato per una variabile (1 gradi libertà) o per due variabili (2 gradi di libertà).
- Nel grafico in questione la linea nera identifica $\Delta\chi^2 = 1$ (le proiezioni sugli assi danno l'intervallo di 1 σ (68%) per ciascuna variabili indipendentemente dall'altra) mentre quella rossa $\Delta\chi^2 = 2.3$ (regione che, nello spazio dei due parametri, corrispondete ad una probabilità del 68%)



- Per definire il $\Delta\chi^2$ corrispondente ad un certo numero di σ in più dimensioni occorre risolvere l'equazione trascendente:

$$Prob(\chi^2, ndf) = \frac{2}{\sqrt{\pi}} \int_0^{N\sigma} e^{-t^2} dt$$

dove il secondo membro, riconducibile all'integrale della gaussiana tra $-N\sigma$ e $N\sigma$, è calcolabile con `TMath::Erf(x)`, mentre il primo è calcolabile con `TMath::Prob(chi2,ndf`. In particolare si può calcolare il $\Delta\chi^2$ relativo ad un certo numero N di sigma in $Ndim$ dimensioni risolvendo:

```
N = TMath::ErfInverse(1-TMath::Prob(DeltaChi2,Ndim))*sqrt(2)
```

Appendice A: BLUE in ND

Soluzione di BLUE in ND con moltiplicatori di Lagrange

$$h(\boldsymbol{\alpha}, \lambda) = f(\boldsymbol{\alpha}) + \lambda g(\boldsymbol{\alpha}) = \boldsymbol{\alpha} V \boldsymbol{\alpha}^T + \lambda(\boldsymbol{\alpha}\boldsymbol{u} - 1)$$

$$\begin{cases} \frac{\partial h}{\partial \boldsymbol{\alpha}} = 0 \longrightarrow \frac{\partial h}{\partial \alpha_k} = \frac{\partial}{\partial \alpha_k}\left(\alpha_i V_{ij} \alpha_j + \lambda(\alpha_i u_i - 1)\right) = V_{kj}\alpha_j + \alpha_i V_{ik} + \lambda u_k = 0 \\ \frac{\partial h}{\partial \lambda} = 0 \longrightarrow \frac{\partial h}{\partial \lambda} = (\alpha_i u_i - 1) \end{cases}$$

Ricordando che la matrice V è simmetrica ($V_{ik} = V_{ki}$) e scrivendo in notazione matriciale
def pos $\Rightarrow$ invertibile

$$2V\boldsymbol{\alpha}^T + \lambda \boldsymbol{u} = 0 \quad (1)$$

$$\boldsymbol{\alpha}\boldsymbol{u} - 1 = 0 \quad (2)$$

Risolviamo rispetto a $\boldsymbol{\alpha}$:

$(V^{-1})^T = V^{-1}$

$$V\boldsymbol{\alpha}^T = -\frac{1}{2}\lambda \boldsymbol{u} \qquad \boldsymbol{\alpha}^T = -\frac{1}{2}\lambda V^{-1}\boldsymbol{u} \qquad \boldsymbol{\alpha} = -\frac{1}{2}\lambda \boldsymbol{u}^T V^{-1}$$

Appendice A: BLUE in ND

Sostituiamo in 2):

$$\left(-\frac{1}{2}\lambda \boldsymbol{u}^T V^{-1}\right)\boldsymbol{u} = 1$$

Da cui si ricava $\hat{\lambda}$:

$$\hat{\lambda} = -\frac{2}{\boldsymbol{u}^T V^{-1}\boldsymbol{u}}$$

dà l'indice di ottimalità per $\alpha_1 \dots \alpha_n$

e, per sostituzione:

$$\hat{\boldsymbol{\alpha}} = -\frac{1}{2}\frac{-2}{\boldsymbol{u}^T V^{-1}\boldsymbol{u}} V^{-1}\boldsymbol{u} = \frac{V^{-1}\boldsymbol{u}}{\boldsymbol{u}^T V^{-1}\boldsymbol{u}}$$

$$\sigma^2_{\bar{y}} = \boldsymbol{\alpha} V \boldsymbol{\alpha}^T = \frac{1}{(\boldsymbol{u}^T V^{-1}\boldsymbol{u})^2}\boldsymbol{u}^T V^{-1} V V^{-1}\boldsymbol{u} = \frac{1}{\boldsymbol{u}^T V^{-1}\boldsymbol{u}}$$

Appendice B: GLS in ND

La relazione da minimizzare è:

$$S(\boldsymbol{\beta}) = \sum_{i,j=1}^{N} (y_i - \beta_k X_{ki}) V_{ij}^{-1} (y_j - X_{jl}\beta_l)$$

in forma matriciale

$$\begin{aligned} S(\boldsymbol{\beta}) &= (\mathbf{y} - X\boldsymbol{\beta})^T V^{-1} (\mathbf{y} - X\boldsymbol{\beta}) \\ &= \mathbf{y}^T V^{-1} \mathbf{y} - (X\boldsymbol{\beta})^T V^{-1} \mathbf{y} - \mathbf{y}^T V^{-1} X\boldsymbol{\beta} + (X\boldsymbol{\beta})^T V^{-1} (X\boldsymbol{\beta}) \\ &= \mathbf{y}^T V^{-1} \mathbf{y} - 2(X\boldsymbol{\beta})^T V^{-1} \mathbf{y} - (X\boldsymbol{\beta})^T V^{-1} (X\boldsymbol{\beta}) \qquad \text{perché } (V^{-1})^T = V^{-1} \end{aligned}$$

La derivata rispetta $\boldsymbol{\beta}$ del primo termine è nulla, analizziamo gli altri:

$$\frac{d[(X\boldsymbol{\beta})^T V^{-1} \mathbf{y}]}{d\beta_n} = \frac{d[\beta_l X_{li} V_{ij}^{-1} y_j]}{d\beta_n} = X_{ni} V_{ij}^{-1} y_j \quad \longrightarrow X^T V^{-1} \mathbf{y}$$

$$\begin{aligned} \frac{d[(X\boldsymbol{\beta})^T V^{-1} (X\boldsymbol{\beta})]}{d\beta_n} &= \frac{d[\beta_k (X^T V^{-1} X)_{kl} \beta_l]}{d\beta_n} \\ &= (X^T V^{-1} X)_{nl} \beta_l + \beta_k (X^T V^{-1} X)_{kn} = (X^T V^{-1} X)_{nl} \beta_l + \beta_l (X^T V^{-1} X)_{ln} \\ &\longrightarrow ((X^T V^{-1} X) + (X^T V^{-1} X)^T)\boldsymbol{\beta} = 2(X^T V^{-1} X)\boldsymbol{\beta} \end{aligned}$$

Da cui:

$$-2X^T V^{-1} \mathbf{y} + 2(X^T V^{-1} X)\boldsymbol{\beta} = 0$$

$$\boldsymbol{\beta} = (X^T V^{-1} X)^{-1} X^T V^{-1} \mathbf{y} = L\mathbf{y}$$

Appendice B: GLS in ND

La matrice di covarianza di $\boldsymbol{\beta}$ è data da:

$$
\begin{aligned}
V_\beta = LVL^T &= (X^T V^{-1} X)^{-1} X^T V^{-1} V ((X^T V^{-1} X)^{-1} X^T V^{-1})^T \\
&= (X^T V^{-1} X)^{-1} X^T V^{-1} V V^{-1} X ((X^T V^{-1} X)^{-1})^T \\
&= (X^T V^{-1} X)^{-1} X^T V^{-1} X ((X^T V^{-1} X)^{-1})^T \\
&= (X^T V^{-1} X)^{-1}
\end{aligned}
$$