UNIVERSITÀ DEGLI STUDI DI PADOVA

METODI STATISTICI PER LA BIOINGEGNERIA

A.A. 2024-2025

Prof. Alessandra Bertoldo
Ing. Mattia De Francisci, Ing. Claudia Tarricone

METODI DI SHRINKAGE

Dati a disposizione e obiettivo del laboratorio:

- I dati a disposizione per il laboratorio sono stati raccolti per uno studio di genetica di 500 soggetti.

- L'obiettivo è utilizzare metodi di shrinkage per determinare quali geni influenzano l'outcome da misurare (ultima colonna della matrice data) ed eventualmente scartare i geni che meno contribuiscono alla relazione lineare ipotizzata

- I dati sono contenuti nel file ***data_lab9.mat****, sono presenti:*

  data: Matrice 2D di dimensioni 500x101 (500 soggetti, 101 variabili: le prime 100 saranno le nostre variabili indipendenti e l'ultima la variabile dipendente)
  N.B. lo z-score è già stato effettuato

1) LINEAR REGRESSION

1. Fissare il random number generator (***rng***) con le istruzioni:

   ```
   rng('default');
   rng(1) ;
   ```

2. Caricare i dati (***data_lab9.mat***) e indicare con n il numero di soggetti
3. Creare la variabile X contenente le variabili indipendenti (da 1 a end-1) e Y contenente la variabile dipendente (ultima colonna)
4. Stimare con il metodo dei minimi quadrati lineari i coefficienti beta.
5. Rappresentare in un plot:

   - a sinistra → l'andamento di Y attraverso i soggetti con sovrapposta la stima del modello
   - a destra → scatter plot del dato (Y) vs. predizione (Y_hat)

2) RIDGE REGRESSION

6. Definire il vettore contenente i valori del coefficiente di regolarizzazione (**lambda_ridge**) da testare. Usare la function **`logspace`** per definire un vettore di 20 punti in scala logaritmica da 10^{-3} a 10^{3}

7. Fittare i dati tramite Ridge Regression. Per la scelta del parametro di regolarizzazione, eseguire una Leave-One-Out-Cross-Validation (LOOCV), selezionando il valore del coefficiente di regolarizzazione che minimizza il valore di MSE (Mean Squared Error), pari alla media della somma dei residui.

Implementazione LOOCV (pseudo-codice):

per ogni valore del parametro di regolarizzazione j-esimo (lambda_ridge)
    per ogni soggetto i-esimo
        training set = tutti i soggetti meno il soggetto i-esimo
        calcolare le stime dei parametri con

$$\beta^{Ridge} = \left(X^T X + \lambda I_{pxp}\right)^{-1} X^T Y$$

        stima(soggetto i-esimo) = predizione del soggetto i-esimo
    end
    calcolo dell'MSE per il valore j-esimo
end

→ La matrice identità I_{pxp} in Matlab si può creare con la function **`eye`**

2) RIDGE REGRESSION

8. Selezionare il lambda ottimo (**lambda_OPT**) corrispondente al minimo valore di MSE. Plottare l'andamento di MSE al variare del parametro di regolarizzazione evidenziando il lambda ottimo.

9. Considerando tutti i soggetti, stimare i valori finali di beta (**B_ridge**) usando il parametro di regolarizzazione selezionato. Calcolare la predizione del modello (**Y_hat_ridge**) e il coefficiente di determinazione tra Y e la predizione (**R2**).

10. Rappresentare in un plot:

- a sinistra → l'andamento di Y attraverso i soggetti con sovrapposta la stima del modello
- a destra → scatter plot del dato (Y) vs. predizione (Y_hat)

11. [BONUS] Ripetere i punti precedenti usando 10 fold CV. Per la figura di MSE, mostrare i risultati sia per LOOCV, sia per 10 fold CV.

3) LASSO REGRESSION

12. Fittare i dati tramite LASSO Regression (function **`lasso`**). Settare una nuova griglia di valori per lambda (lambda_vec=logspace(-1,0.5,200)).

Nella function **`lasso`** impostare: 'CV'=10 (10 Fold Cross-Validation), 'Alpha' = 1, 'lambda' = lambda_vec. Identificare, sulla base del valore di MSE, il valore ottimo di lambda (**lambda_OPT_lasso**). Visualizzare l'andamento di MSE al variare di lambda tramite **`lassoPlot`**.

13. Considerando tutti i soggetti, stimare i valori finali di beta (refit del modello) usando il **lambda_OPT_lasso** selezionato. Calcolare la predizione del modello e il coefficiente di determinazione tra Y e la predizione.

14. Rappresentare in un plot:

- a sinistra → l'andamento di Y attraverso i soggetti con sovrapposta la stima del modello
- a destra → scatter plot del dato (Y) vs. predizione (Y_hat)

4) ELASTIC-NET REGRESSION

15. Fittare i dati tramite ELASTIC-NET Regression (**`lasso`**). Settare la griglia di valori per lambda come nel caso Lasso (lambda_vec =logspace(-1,0.5,200)).

Nella function **`lasso`** impostare: 'CV'=10 (10 Fold Cross-Validation), 'Alpha' = 0.5, 'lambda' = lambda_vec. Identificare, sulla base del valore di MSE, il valore ottimo di lambda (**lambda_OPT_lasso**). Visualizzare l'andamento di MSE al variare di lambda tramite **`lassoPlot`**.

16. Considerando tutti i soggetti, stimare i valori finali di beta (refit del modello) usando il parametro di regolarizzazione selezionato. Calcolare la predizione del modello e il coefficiente di determinazione tra Y e la predizione.

17. Stampare sulla Command Window (**`disp`**) i valori minimi di MSE ottenuti con Lasso ed ElasticNet. Quale dei due descrive meglio i dati?

18. Rappresentare in un plot:

- a sinistra → l'andamento di Y attraverso i soggetti con sovrapposta la stima del modello
- a destra → scatter plot del dato (Y) vs. predizione (Y_hat)

5) CONFRONTO TRA I METODI

19. Controllare i valori assunti dai beta con i 4 metodi:

- **Regressione lineare**
- **Ridge regression**
- **Lasso**
- **Elastic Net**

Quanti coefficienti beta sono uguali a zero per ogni metodo? Il risultato è atteso?