

# Training, validazione e test Riduzione delle feature

Vincenzo Bonifaci

IN550 – Machine Learning

•

# Training set e test set

Separiamo **a caso** i dati di esempio a nostra disposizione in due insiemi:



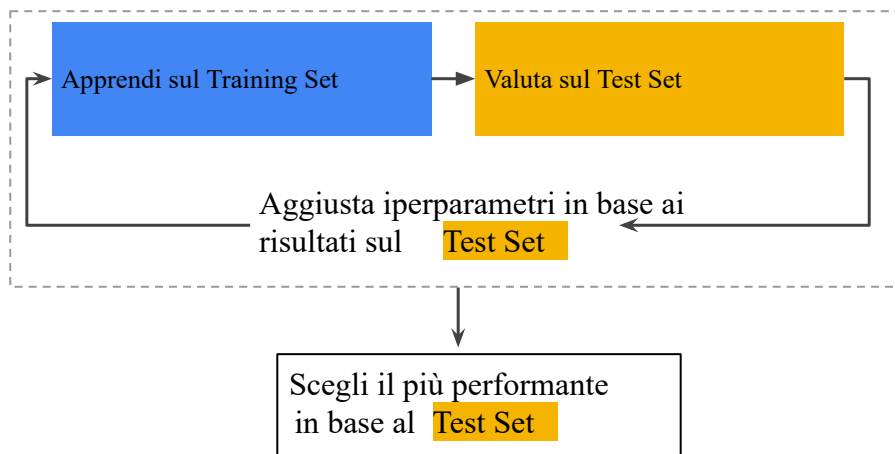
# Impostazione degli iperparametri

Molti metodi di apprendimento richiedono di specificare **iperparametri**:

- $\eta$  e  $T$  negli algoritmi basati su Gradient Descent
- $K$  nei metodi  $K$ -Nearest Neighbor
- $\lambda$  nei metodi con regolarizzazione
- ...

Quale metodologia per selezionare i valori degli iperparametri?

# Un possibile schema di lavoro?

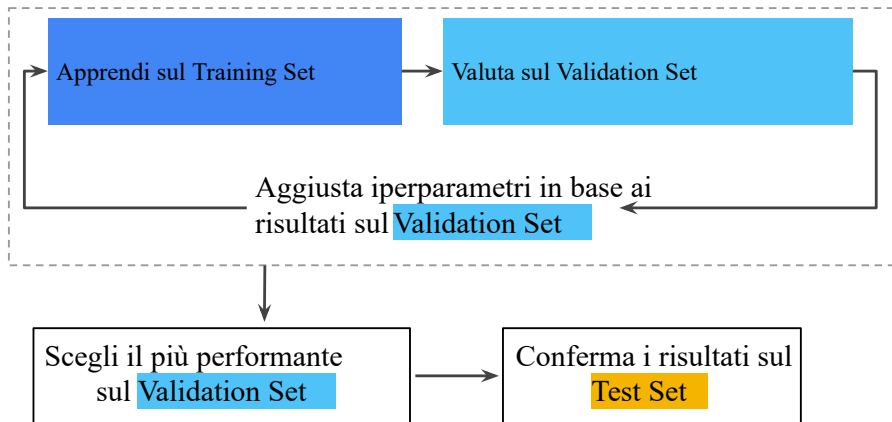


## Il validation set

Separiamo **a caso** i dati di esempio a nostra disposizione in **tre** insiemi:



# Uso di un validation set



# Dimensionamento dei vari insiemi

- Validation set e test set devono essere **sufficientemente** grandi da poter stimare con la precisione desiderata il rischio atteso
- Compatibilmente col punto precedente, il training set deve essere il più grande possibile; sarà sempre il più grande dei tre insiemi

## Dimensione del test set e stima del rischio atteso

## Teorema (Dimensione del test set)

Sia  $h$  un'ipotesi e si consideri una funzione costo a valori in  $[0, b]$ . Allora per ogni  $\delta \in (0, 1)$ , con probabilità almeno  $1 - \delta$  sulla scelta di un test set  $T$  di dimensione  $m_T$  si ha

$$\frac{1}{m_T} \sum_i \ell(h, (x^{(i)}, y^{(i)})) \uparrow \quad |L_T(h) - L_D(h)| \leq b \underbrace{\sqrt{\frac{\log(2/\delta)}{2m_T}}}_{\epsilon} \leq b \sqrt{\frac{1/100}{2m_T}} \rightarrow \frac{1}{m_T} \approx \left(\frac{1}{100}\right)^2 \downarrow m_T \approx 10000$$

Interpretazione: l'errore con cui il test set stima il rischio atteso decresce con la radice quadrata della dimensione del test set

⇒ per una stima di  $L_D(h) \pm 1\%$  è sufficiente  $m_T$  dell'ordine di 10,000

$$E_{(x,y) \sim D} [\ell(h, (x,y))]$$



## Dimensionamento dei vari insiemi in pratica

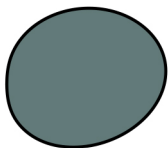
- Per dataset **piccoli** o **medi** ( $\approx 100 - 10,000$  osservazioni) è spesso usata una suddivisione 70%-30% oppure 60%-20%-20%:  
70% per il training set, 30% per il test set, o  
60% per il training set, 20% per il validation set, 20% per il test set
- Per dataset **grandi** ( $\approx 100,000 - 1,000,000$  osservazioni e oltre) può essere sufficiente mantenere validation set e test set intorno alle 10,000 o 100,000 osservazioni



Esempio: 1,000,000 osservazioni

98% training set, 1% validation set, 1% test set può essere accettabile

# Validazione incrociata [Cross-validation]

Se i dati scarseggiano, possiamo fare a meno del validation set?



 training  
 validation



fold 1



fold 2



fold 3

# Validazione incrociata [Cross-validation]

$K$  intero positivo (valore spesso utilizzato:  $K = 10$ )

## $K$ -fold Cross Validation ( $K$ -CV)

- 1 Partiziona il training set  $S$  in  $K$  sottoinsiemi (*fold*)  $S_1, \dots, S_K$
- 2 Ripeti per  $i = 1, \dots, K$ :
  - Apprendi un'ipotesi  $h_i$  usando tutti i dati in  $S$  **tranne quelli in  $S_i$**
  - Stima il rischio atteso di  $h_i$  usando i dati in  $S_i$ :  $L_{S_i}(h_i)$
- 3 Restituisci, come stima del rischio atteso dell'ipotesi appresa su  $S$ , il rischio medio di validazione:

$$\frac{1}{K} \sum_{i=1}^K L_{S_i}(h_i)$$

**Avvertenza:** la validazione incrociata è un'euristica che spesso funziona bene in pratica, ma raramente è supportata da garanzie teoriche

# Validazione incrociata per la selezione di un modello

La validazione incrociata permette anche di **selezionare** gli iperparametri

## $K$ -CV per la selezione degli iperparametri

**Input:** training set  $S$ , insieme di iperparametri  $\Theta$ , algoritmo  $A$ , intero  $K$

- 1 Partiziona il training set  $S$  in  $K$  sottoinsiemi (*fold*)  $S_1, \dots, S_K$
- 2 Per ogni  $\theta \in \Theta$ :
  - Per  $i = 1, \dots, K$ :  $h_{i,\theta} = A(S \setminus S_i; \theta)$
  - $\text{risk}(\theta) = \frac{1}{K} \sum_{i=1}^K L_{S_i}(h_{i,\theta})$
- 3  $\theta^* = \text{argmin}_{\theta} \text{risk}(\theta)$
- 4  $h^* = A(S; \theta^*)$

# Riduzione delle feature

# Perché ridurre le feature?

Supponiamo di avere  $d$  variabili di input e  $m$  osservazioni

Ridurre il numero di predittori ( $d$ ) può essere importante per:

- Tenere sotto controllo la **varianza** (specialmente quando  $d > m$ )
- Migliorare l'**interpretabilità** del modello

# Metodi di riduzione delle feature

- *Selezione*: selezioniamo un **sottoinsieme** di  $d' < d$  predittori
- *Shrinkage (regolarizzazione)*: penalizziamo modelli in cui **tanti** predittori hanno **grande** influenza sulla predizione
- *Proiezione*: proiettiamo i  $d$  predittori su un **sottospazio**  $d'$ -dimensionale con  $d' < d$

**NB.** Qui illustreremo i metodi nel contesto della regressione lineare, ma i principi sono generali

## Metodi di selezione



# Selezione: Best Subset Selection

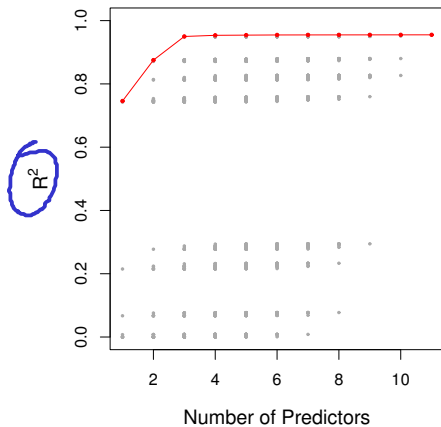
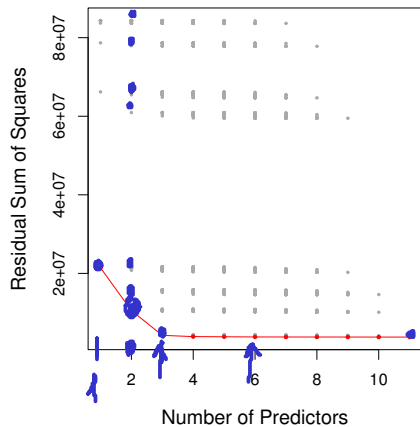
## Best Subset Selection

- 1**  $h_0 \leftarrow$  miglior ipotesi **costante**, cioè con 0 predittori  
Per esempio, nella regressione lineare,

$$h_0(x) = \frac{1}{m} \sum_{i=1}^m y^{(i)}$$

- 2** Per  $k = 1, 2, \dots, d$ ,
- Apprendi tutte le  $\binom{d}{k}$  ipotesi consistenti di  $k$  predittori
  - $h_k \leftarrow$  ipotesi col miglior **rischio empirico** tra queste
- 3** Restituisci l'ipotesi  $h \in \{h_0, h_1, \dots, h_d\}$  col minimo **rischio atteso** stimato sul validation set (o con la validazione incrociata)

# Esempio di Best Subset Selection



# Best Subset vs. selezione passo-passo

- Il numero totale di ipotesi considerate è  $2^d$ :  
inapplicabile se  $d$  è molto grande
- Se  $d$  è grande, lo spazio di ricerca è enorme e può dare luogo ad **overfitting**
- Per questo motivo, un'alternativa è costituita dai metodi di *selezione passo-passo*

# Selezione passo-passo in avanti

- Inizia con un'ipotesi senza predittori
- Aggiungi un predittore alla volta: quello che fornisce il **massimo incremento** della qualità del fit *(mass. decrem. del rischio emp.)*
- Restituisci l'ipotesi col minimo rischio atteso stimato (sul validation set) tra tutte le  $d + 1$  ipotesi costruite

# Selezione passo-passo in avanti

## Forward Stepwise Selection

- 1  $h_0 \leftarrow$  miglior ipotesi costante, con 0 predittori
  - Per  $k = 0, \dots, d - 1$ , considera tutte le  $d - k$  ipotesi con un predittore **in più** di  $h_k$
  - $h_k \leftarrow$  ipotesi col miglior **rischio empirico** tra queste
- 2 Restituisci l'ipotesi  $h \in \{h_0, h_1, \dots, h_d\}$  col minimo **rischio atteso** stimato sul validation set (o con la validazione incrociata)

# Vantaggi e svantaggi di Forward Stepwise Selection

- Ipotesi esplorate:  $d + 1$  anziché  $2^d$
- **Non** garantisce il miglior modello tra i  $2^d$  possibili

## Esempio: Credit dataset

$k$	Best Subset	Forward Stepwise
1	rating	rating
2	rating, income	rating, income
3	rating, income, student	rating, income, student
4	<u>cards</u> , <u>income</u> , <u>student</u> , <u>limit</u>	rating, income, student, limit

# Metodi di shrinkage e regolarizzazione



# Regularizzazione

Come cercare automaticamente un equilibrio tra bias e varianza?

Una *funzione di regolarizzazione* è una funzione  $R : \mathcal{H} \rightarrow \mathbb{R}_+$

$R(h)$  è una qualche misura di complessità dell'ipotesi  $h$

## Regularized Loss Minimization (RLM)

Dato un insieme di esempi  $S$ , cerca una regola di predizione  $h$  che minimizzi il rischio empirico di  $h$  su  $S$ , più  $R(h)$ :

$$\min_{h \in \mathcal{H}} L_S(h) + R(h)$$

Se l'ipotesi  $h$  è codificata dai coefficienti  $w \in \mathbb{R}^{d+1}$ , scriveremo anche  $R(w)$  (in tal caso  $R$  è vista come funzione definita su  $\mathbb{R}^{d+1}$  anziché su  $\mathcal{H}$ )

# Regressione Ridge

$w = (w_0, w_1, \dots, w_d)$  include il termine costante  $w_0$   
 $\omega = (w_1, w_2, \dots, w_d)$  non lo include

## Regolarizzazione $\ell_2$ (di Tikhonov)

$$R(w) = \lambda(w_1^2 + w_2^2 + \dots + w_d^2) = \lambda \|\omega\|_2^2 \quad (\lambda \geq 0)$$

## ⇒ Regressione Ridge

$$\min_{w \in \mathbb{R}^{d+1}} L_S(h_w) + \lambda \|\omega\|^2 = \min_{w \in \mathbb{R}^{d+1}} \frac{1}{m} \|Xw - y\|^2 + \lambda \|\omega\|^2$$

■ Per  $\lambda = 0$ , coincide con il metodo dei minimi quadrati

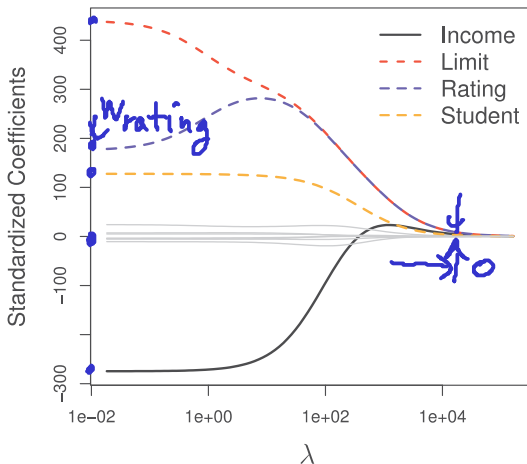
■ Per  $\lambda \rightarrow \infty$ , i coefficienti  $w_k$  tendono a 0

■ Ammette soluzione in forma chiusa:  $w^* = (X^T X + \lambda m \cdot I_0)^{-1} X^T y$

Handwritten notes and diagram:

- A blue arrow points from the  $\lambda \|\omega\|^2$  term in the equation above to the  $d+1$  dimensionality of the  $I_0$  matrix.
- The matrix  $I_0$  is defined as a  $(d+1) \times (d+1)$  identity matrix with the first row and column zeroed out:
 
$$I_0 = \begin{pmatrix} 0 & & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & \\ & & & & 0 \end{pmatrix}$$

# Regressione Ridge



Valori dei coefficienti di una regressione ridge in funzione di  $\lambda$

# Regressione regolarizzata e standardizzazione delle variabili

Nella regressione regolarizzata è importante *standardizzare* i predittori, centrandoli sulla media e scalandoli per la deviazione standard:

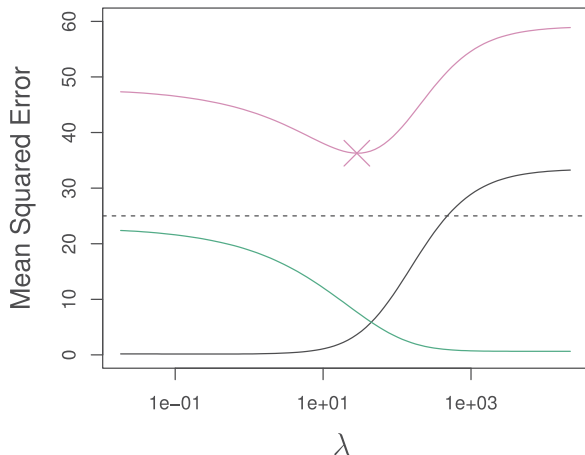
## Standardizzazione di una variabile

Se  $v$  è un predittore, con esempi di valore  $v_1, \dots, v_m$  e media  $\bar{v}$ , poniamo

$$v \leftarrow \frac{v - \bar{v}}{\sqrt{\frac{1}{m} \sum_{i=1}^m (v_i - \bar{v})^2}}$$

Questo non era necessario nella regressione lineare, perché lì i coefficienti erano *equivarianti* rispetto alla scala: moltiplicare  $v$  per  $c$  scalava il coefficiente corrispondente di  $1/c$

# Regressione Ridge



$Bias^2$  (nero), varianza (verde) e MSE di test (viola) per una regressione ridge in funzione di  $\lambda$

# Regressione LASSO

## Regolarizzazione $\ell_1$

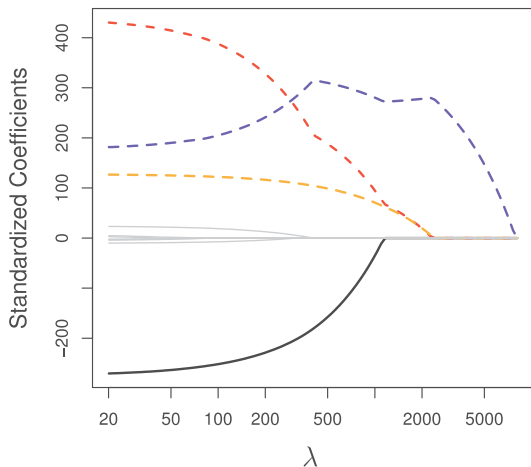
$$R(w) = \lambda(|w_1| + |w_2| + \dots + |w_d|) = \lambda \|\omega\|_1 \quad (\lambda \geq 0)$$

## ⇒ Regressione LASSO

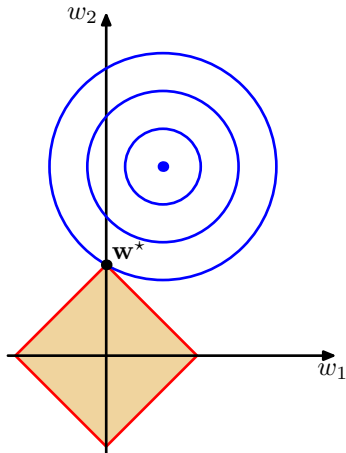
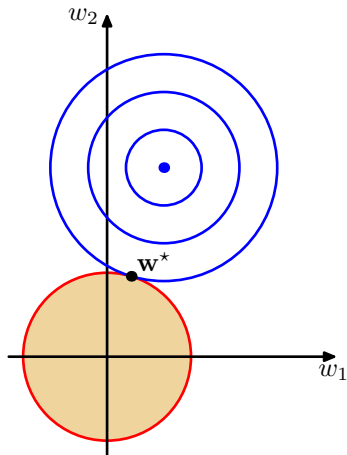
$$\min_{w \in \mathbb{R}^{d+1}} L_S(h_w) + \lambda \|\omega\|_1 = \min_{w \in \mathbb{R}^{d+1}} \frac{1}{m} \|Xw - y\|^2 + \lambda \|\omega\|_1$$

- Per  $\lambda = 0$ , coincide con il metodo dei minimi quadrati
- Per  $\lambda \rightarrow \infty$ , i coefficienti  $w_k$  tendono a 0
- Per  $\lambda$  crescente, alcuni coefficienti diventano **esattamente** pari a 0  
(⇒ incentiva modelli *sparsi*)

# Regressione LASSO

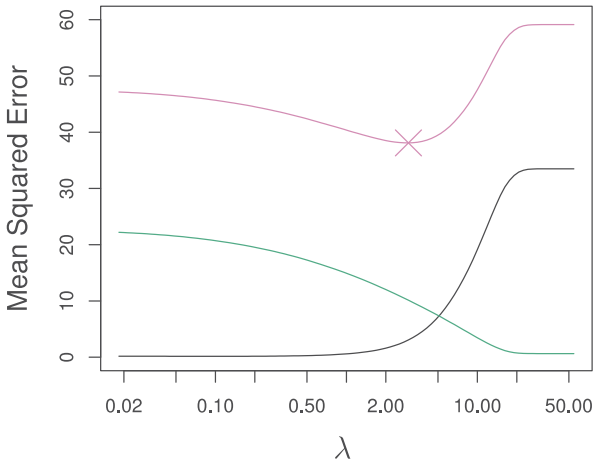


Valori dei coefficienti di una regressione LASSO in funzione di  $\lambda$

Ridge ( $\lambda \|w\|_2^2$ ) vs. LASSO ( $\lambda \|w\|_1$ )



# Regressione LASSO



Bias<sup>2</sup> (nero), varianza (verde) e MSE di test (viola) per una regressione LASSO in funzione di  $\lambda$

# Metodi di regressione regolarizzata in scikit-learn

	Iperparametri	Interfaccia scikit-learn
Ridge (diretto)	$\lambda$	Ridge
Ridge (SGD)	$\eta, T, \lambda$	SGDRegressor(penalty='l2')
LASSO (SGD)	$\eta, T, \lambda$	SGDRegressor(penalty='l1')

## Metodi di proiezione

Parleremo dei metodi di proiezione nel contesto dell'apprendimento non supervisionato

Esempio tipico: Principal Component Analysis

