

- multiple regression (linear regression con multiple features)

polynomial regression

caso generale

$$y_i = w_0 + w_1 x_i + w_2 x_i^2 + \cdots + w_p x_i^p + \epsilon_i \quad \text{notazione matriciale}$$

Il caso generale, con un solo input x_i , è il seguente:

$$\begin{aligned} y_i &= w_0 \phi_0(x_i) + w_1 \phi_1(x_i) + \cdots + w_D \phi_D(x_i) + \epsilon_i \\ &= \sum_{j=0}^D w_j \phi_j(x_i) + \epsilon_i \end{aligned}$$

Il caso generale, che ha in input un vettore \mathbf{x}_i , è pertanto il seguente:

$$\begin{aligned} y_i &= w_0 \phi_0(\mathbf{x}_i) + w_1 \phi_1(\mathbf{x}_i) + \cdots + w_D \phi_D(\mathbf{x}_i) + \epsilon_i \\ &= \sum_{j=0}^D w_j \phi_j(\mathbf{x}_i) + \epsilon_i \end{aligned}$$

costo da minimizzare, la **RSS** definita come segue, a partire da N osservazioni disponibili:

$$\text{RSS}(\mathbf{w}) = \sum_{i=1}^N (y_i - \hat{y}_i)^2 = \sum_{i=1}^N [y_i - (w_0 \phi_0(\mathbf{x}_i) + \cdots + w_D \phi_D(\mathbf{x}_i))]^2$$

$$\begin{aligned} y_i &= [w_0 \ w_1 \ \cdots \ w_D] \cdot \begin{bmatrix} \phi_0(\mathbf{x}_i) \\ \phi_1(\mathbf{x}_i) \\ \vdots \\ \phi_D(\mathbf{x}_i) \end{bmatrix} + \epsilon_i \\ y_i &= [\phi_0(\mathbf{x}_i) \ \phi_1(\mathbf{x}_i) \ \cdots \ \phi_D(\mathbf{x}_i)] \cdot \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_D \end{bmatrix} + \epsilon_i \\ y_i &= \sum_{j=0}^D w_j \phi_j(\mathbf{x}_i) + \epsilon_i = \mathbf{w}^T \cdot \phi(\mathbf{x}_i) + \epsilon_i = \phi^T(\mathbf{x}_i) \cdot \mathbf{w} + \epsilon_i \end{aligned}$$

$$\mathbf{y} = \Phi \cdot \mathbf{w} + \boldsymbol{\epsilon} \Rightarrow \boldsymbol{\epsilon} = \mathbf{y} - \Phi \mathbf{w}$$

$$\begin{aligned} \text{RSS}(\mathbf{w}) &= \sum_{i=1}^N [y_i - \phi^T(\mathbf{x}_i) \cdot \mathbf{w}]^2 \\ &= \sum_{i=1}^N \epsilon_i^2 = \boldsymbol{\epsilon}^T \cdot \boldsymbol{\epsilon} \end{aligned}$$

$$\text{RSS}(\mathbf{w}) = (\mathbf{y} - \Phi \mathbf{w})^T (\mathbf{y} - \Phi \mathbf{w})$$

$$\nabla \text{RSS}(\mathbf{w}) = \nabla[(\mathbf{y} - \Phi \mathbf{w})^T (\mathbf{y} - \Phi \mathbf{w})] = -2 \Phi^T (\mathbf{y} - \Phi \mathbf{w})$$

minimizzazione

forma chiusa

(1)

$$\nabla \text{RSS}(\mathbf{w}) = -2 \Phi^T (\mathbf{y} - \Phi \mathbf{w}) = \mathbf{0} \rightarrow \hat{\mathbf{w}} = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{y}$$

gradient descent

(2)

$$\mathbf{w}^{(t+1)} \leftarrow \mathbf{w}^{(t)} - \alpha \cdot \nabla \text{RSS}(\mathbf{w}^{(t)})$$

In pratica, possiamo terminare l'elaborazione quando:

$$\|\nabla \text{RSS}(\mathbf{w}^{(t)})\|_2 \leq \epsilon$$

$$\mathbf{w}^{(1)} = \mathbf{0} \text{ (oppure lo inizializziamo in modo casuale)}$$

$$t = 1$$

$$\text{while } \|\nabla \text{RSS}(\mathbf{w}^{(t)})\|_2 > \epsilon$$

$$\text{for } j = 0, 1, \dots, D$$

$$\text{derivata_parziale}[j] = -2 \sum_{i=1}^N \phi_j(\mathbf{x}_i) [y_i - \hat{y}_i(\mathbf{w}^{(t)})]$$

$$w_j^{(t+1)} \leftarrow w_j^{(t)} - \alpha * \text{derivata_parziale}[j]$$

$$t \leftarrow t + 1$$

Valutazione regression

• loss function

$$L[y, \hat{f}_{\hat{w}}(\mathbf{x})]$$

La funzione di Loss può essere ad esempio definita come
Errore Assoluto (Absolute Error):

$$L[y, \hat{f}_{\hat{w}}(\mathbf{x})] = |y - \hat{f}_{\hat{w}}(\mathbf{x})|$$

oppure come Errore Quadratico (Squared Error):

$$L[y, \hat{f}_{\hat{w}}(\mathbf{x})] = [y - \hat{f}_{\hat{w}}(\mathbf{x})]^2$$

• training error

- Definizione di una Loss Function (absolute error, squared error, ecc.)

- Calcolo del Training Error come "average loss", definito sugli N punti di training:

$$\text{Training Error} = \frac{1}{N} \cdot \sum_{i=1}^N L[y_i, \hat{f}_{\hat{w}}(\mathbf{x}_i)]$$

• generalisation error

Formalmente, possiamo definire il Generalization (o True) Error come segue:

$$\text{Generalization Error} = E_{x,y}[L(y, \hat{f}_{\hat{w}}(\mathbf{x}))]$$

ossia come l'average value della funzione Loss, calcolato su tutte le possibili coppie (x, y) pesate in base alla loro probabilità di comparire nella zona.

non è calcolabile

• test error

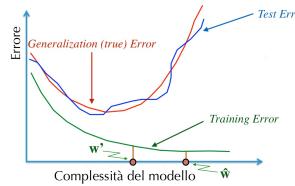
Definito come average loss sui punti dell'insieme di test:

$$\text{Test Error} = \frac{1}{N_{\text{test}}} \cdot \sum_{i \in \text{test}} L[y_i, \hat{f}_{\hat{w}}(\mathbf{x}_i)]$$

• overfitting

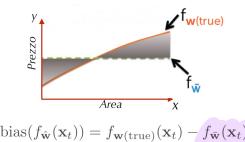
Dato un modello con parametri \hat{w} , si ha overfitting se esiste un modello con i parametri stimati w' tale che:

- training error(\hat{w}) < training error(w')
- true error(\hat{w}) > true error(w')



Sorgenti di errore: noise, bias, variance

Il bias è definito come la differenza tra la **funzione media** e la true function:

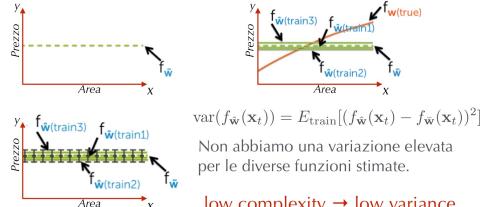


$$\text{bias}(f_{\hat{w}}(\mathbf{x}_t)) = f_{\text{true}}(\mathbf{x}_t) - f_{\hat{w}}(\mathbf{x}_t)$$

E' in sostanza una valutazione di quanto il mio modello si adatti alla true function.

low complexity → high bias

Per introdurre il concetto di varianza nella regression, dobbiamo considerare quanto le varie funzioni f stimate differiscono dalla funzione media. Vediamo ad esempio il caso di modello costante:

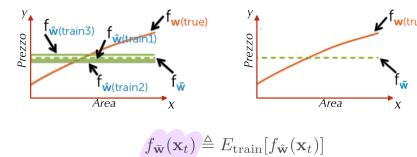


$$\text{var}(f_{\hat{w}}(\mathbf{x}_t)) = E_{\text{train}}[(f_{\hat{w}}(\mathbf{x}_t) - f_{\hat{w}}(\mathbf{x}_t))^2]$$

Non abbiamo una variazione elevata per le diverse funzioni stimate.

low complexity → low variance

Con :



Il bias per modelli "high order" è invece in genere basso:

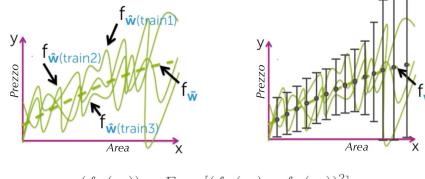
$$f_{\hat{w}}(\mathbf{x}_t) \triangleq E_{\text{train}}[f_{\hat{w}}(\mathbf{x}_t)]$$

$$\text{bias}(f_{\hat{w}}(\mathbf{x}_t)) = f_{\text{true}}(\mathbf{x}_t) - f_{\hat{w}}(\mathbf{x}_t)$$



high complexity → low bias

Se consideriamo le funzioni f stimate per i possibili training set:

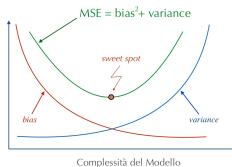


$$\text{var}(f_{\hat{w}}(\mathbf{x}_t)) = E_{\text{train}}[(f_{\hat{w}}(\mathbf{x}_t) - f_{\hat{w}}(\mathbf{x}_t))^2]$$

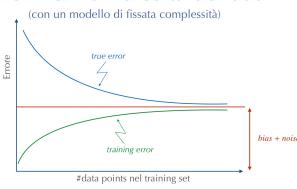
Questa volta la variazione è elevata.

high complexity → high variance

Bias-Variance Tradeoff



Errori vs. numerosità dei dati



Andamento del True Error:

Se abbiamo pochi punti nel training set l'errore è alto, perché la funzione f (fitted function) non stima bene la "true relationship" tra x e y .

Aumentando i punti l'errore diminuisce.

Al limite, esso tende ad un valore uguale a: bias + noise. Questo perché, anche se avessimo tutte le osservazioni possibili, il modello potrebbe non essere sufficientemente flessibile per catturare perfettamente la "true relationship" (questa è la nostra definizione di bias).

A ciò si aggiunge il noise che, come sappiamo, non possiamo controllare.

WORKFLOW (NO PEAKING)

1. Model selection

Per ogni modello di complessità λ :

- stima dei parametri \hat{w}_λ sul training set
- valutazione delle prestazioni sul validation set
- scelta del parametro $\lambda (\lambda^*)$ che comporta il più basso errore sul validation set

2. Model assessment

Calcolo del test error (usando dunque il test set) con \hat{w}_{λ^*} per approssimare il Generalization Error.