Parte I

Apprendimento supervisionato

1 INTRODUZIONE

1.1 OBIETTIVI E APPLICAZIONI DEI METODI DI MACHINE LEARNING

Per learning si intende il processo di individuare, a partire da un insieme di dati, un modello che li descriva. Per esempio, nel caso del supervised learning si vuole individuare una corrispondenza tra input e output. Un modo per fare ciò è postulare l'esistenza di un qualche tipo di meccanismo parametrico per la generazione dei dati, di cui non conosciamo però i valori esatti dei parametri. Questo processo fa tipicamente riferimento a tecniche di tipo statistico.

L'estrazione di leggi generali a partire da un insieme di dati osservati viene denominata *induzione*, e si contrappone alla *deduzione* in cui, a partire da leggi generali, si vuole prevedere il valore di un insieme di variabili.

L'induzione è il meccanismo fondamentale alla base del metodo scientifico, in cui si vuole derivare leggi generali (tipicamente descritte in linguaggio matematico) a partire dall'osservazione di fenomeni, osservazione che comprende la misurazione di un insieme di variabili, e quindi l'acquisizione di dati che descrivono i fenomeni osservati.

Il modello ottenuto potrà quindi essere utilizzato per effettuare previsioni rispetto a ulteriori dati, e il processo complessivo nel quale, a partire da un insieme di osservazioni, si vuole effettuare previsioni rispetto a nuove situazioni prende il nome di *inferenza*.

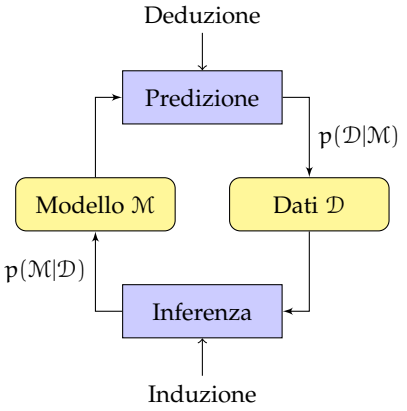

Figura 1: Schema generale di machine learning

1.2 CLASSIFICAZIONE DEI PROBLEMI DI LEARNING

I problemi di learning vengono divisi in due tipologie principali: *supervised learning* e *unsupervised learning*.

1.2.1 Supervised learning

Il supervised learning è il problema più studiato nel machine learning. Esso si pone l'obiettivo di prevedere, dato un elemento di cui si conoscono un insieme di parametri (*features*), il valore di un diverso parametro di *output* relativo all'elemento stesso.

Per far ciò, nel supervised learning viene definito (mediante apprendimento da insiemi di esempi) un *modello*.

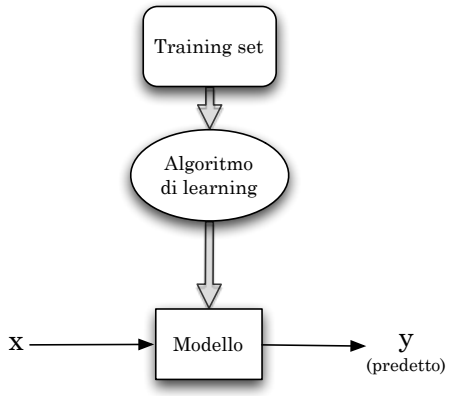

Figura 2: Schema generale di supervised learning

Il problema è definito a partire da un universo di elementi (descritti dai valori assunti da un insieme $\mathbf{x}$ di *features* considerate come input del problema): ad ogni elemento è poi associato un valore y di output (o *target*). Quel che si vuole è, a partire dalla conoscenza di un insieme $\mathcal{T}$ di elementi (denominato *training set*), ciascuno descritto da una coppia $(\mathbf{x}_i, y_i)$, dove $\mathbf{x}_i$ è il vettore dei valori delle d features $x_i^{(1)}, x_i^{(2)}, \ldots, x_i^{(d)}$ e y_i è il relativo output, derivare un *modello* della relazione (sconosciuta) tra features e valori di output, che consenta, dato un nuovo elemento $\mathbf{x}$, di predire il corrispondente valore di output $\mathbf{y}$. Ad esempio, problemi di questo tipo possono essere: predire la presenza o meno di una malattia in presenza dei risultati di un insieme di analisi cliniche, prevedere una quotazione sulla borsa di domani a partire dall'andamento dei giorno precedenti, prevedere il possibile gradimento di un film da parte di uno spettatore a partire dalle preferenze rispetto ad altri film già visti, etc.

Si noti che i valori assunti dalle singoli features possono essere di varia natura:

- Quantitativi: forniscono la misura di una grandezza
- Qualitativi: specificano una classe di appartenenza
- Qualitativi ordinati: specificano l'appartenenza ad un intervallo

Allo stesso modo, i valori di output possono essere:

- Quantitativi: in questo caso il valore restituito è la predizione di una misura, e si parla di *regressione*. Se y è un vettore, si parla di *regressione multivariata*.
- Qualitativi: in questo caso il valore restituito è l'assegnazione ad una classe (categoria), e si parla di *classificazione* o *pattern recognition*. In

particolare, se il numero di possibili classi è 2, si parla di *classificazione binaria*, altrimenti di *classificazione multi-classe*.

- Qualitativi ordinati: in questo caso si parla di *regressione ordinale*.

In figura 3 viene mostrato a sinistra un training set con features in $\mathbb{R}^2$ e classificazione binaria (rosso/nero), Ad esempio, i due assi potrebbero riportare valori di insulina e colesterolo e le classi potrebbero essere SANO (nero) e MALATO (malato), per un insieme di pazienti osservati. Al centro è mostrato un esempio di classificatore lineare: una separazione delle due classi mediante una linea retta. Si noti che tre elementi del training set (due rossi e un bianco) sono classificati male. A destra, le classi sono separate da una curva più complessa: non ci sono errori di classificazione, ma la separazione potrebbe essere eccessivamente dipendente dal training set (overfitting).

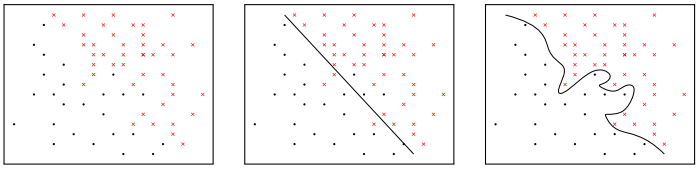

Figura 3: Esempio di classificazione binaria

Per la rappresentazione del training set, assumiamo che siano disponibili n osservazioni ognuna data dal valore di d features. Allora:

- per ogni osservazione, abbiamo un vettore $\mathbf{x}_i$ ($1 \leqslant i \leqslant n$) di dimensione d: $x_i^{(j)}$ ($1 \leqslant j \leqslant d$) indica il valore della j-esima feature nell'osservazione $\mathbf{x}_i$
- una feature classificatoria (discreta) z che può assumere i valori $1, \ldots, K$ sarà in generale rappresentata per mezzo di K variabili $z_1, \ldots, z_K$ in modo tale che $x = i$ se e solo se $z_i = 1, z_j = 0\ \forall j \neq i$.
- abbiamo inoltre un un vettore $\mathbf{y}$ di dimensione n, dove y_i ($1 \leqslant i \leqslant n$) indica il valore dell'output corrispondente alla i-ma osservazione

Il tutto è rappresentabile da una coppia: matrice $\mathbf{X}$ delle osservazioni - vettore $\mathbf{y}$ dei risultati

$$\begin{bmatrix} - & \mathbf{x}_1^T & - \\ - & \mathbf{x}_2^T & - \\ & \vdots & \\ - & \mathbf{x}_n^T & - \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

dove quindi si ha che (per ogni $1 \leqslant i \leqslant n, 1 \leqslant j \leqslant d$) $x_{ij} = x_i^{(j)}$.

$$\begin{pmatrix} x_{11} & x_{21} & \cdots & x_{d1} \\ x_{12} & x_{22} & \cdots & x_{d2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1n} & x_{2n} & \cdots & x_{dn} \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

Per brevità, indicheremo a volte la coppia $\mathbf{X}, \mathbf{y}$ (e quindi il training set) come $\mathcal{T}$.

Un approccio comune al problema (e che sarà considerato nel seguito) è quello di derivare dal training set una funzione h tale che $y(\mathbf{x})$ sia una buona (in qualche modo da definire) previsione del valore sconosciuto y.

Un approccio alternativo è derivare, a partire dal training set, una distribuzione di probabilità $p(y|\mathbf{x})$ per l'output, che consenta ad esempio di ottenere anche dei valori di confidenza. Quella che in effetti verrà utilizzata è la distribuzione a posteriori $p(y|\mathbf{x}, \mathcal{T})$, dove "a posteriori" indica successivamente all'osservazione del training set.

In effetti, la distribuzione che fornisce una descrizione completa della situazione è la distribuzione di probabilità congiunta $p(\mathbf{x}, y)$, che nel caso vogliamo derivare a partire dal training set, e quindi come $p(\mathbf{x}, y|\mathcal{T})$. Ricordando che

$$p(\mathbf{x}, y) = p(y|\mathbf{x})p(\mathbf{x})$$

vediamo come determinare $p(y|\mathbf{x})$ equivale a determinare la probabilità congiunta nel caso in cui la probabilità $p(\mathbf{x})$ degli input è assunta uniforme.

Una volta derivato un modello, una possibile misura della sua qualità nell'effettuare predizioni è la *verosimiglianza a posteriori*, che misura, su un insieme di test di elementi (features e output), la probabilità che il modello assegna ai valori corretti, a partire dalle features

$$L(\mathcal{M}|\mathcal{T}) = \prod_{i=1}^{n} p(y_i|\mathbf{x}_i, \mathcal{T}, \mathcal{M})$$

dove n è la dimensione del test set. Tipicamente, viene considerato, per semplicità il logaritmo (in una qualche base) di tale grandezza, che prende il nome di *log-likelihood*

$$l(\mathcal{M}|\mathcal{T}) = \sum_{i=1}^{n} \log p(y_i|\mathbf{x}_i, \mathcal{T}, \mathcal{M})$$

Una misura più semplice della qualità del modello è naturalmente la percentuale di errore.

1.2.2 Modelli parametrici e non parametrici

In un *modello parametrico*, il modello stesso è preventivamente caratterizzato da un vettore θ di parametri: quel che quindi accade è che viene ipotizzato che la relazione tra features e input sia rappresentabile all'interno di una famiglia di relazioni (modelli) parametrici rispetto a θ, e quindi tali che una assegnazione di valori a θ definisca uno specifico modello della famiglia. Gli elementi nel training set sono successivamente utilizzati per derivare tale assegnazione di valori ai parametri (o una distribuzione di probabilità per tali valori), dopo di che non sono più utilizzati.

Il processo di derivare $p(\theta|\mathcal{T})$ è detto *apprendimento* (o *learning*). Spesso, per semplicità, invece di derivare la distribuzione di θ, viene stimato un "buon" valore $\hat{\theta}(\mathcal{T})$ per i parametri, mediante una *stima di punto*.

Data $p(\theta|\mathcal{T})$, la previsione su y può essere formulata in termini di valore atteso

$$p(y|\mathbf{x}, \mathcal{T}) = \int p(y|\mathbf{x}, \theta)p(\theta|\mathcal{T})d\theta$$

In presenza della stima di punto $\hat{\theta}(\mathcal{T})$, la previsione può essere approssimata come

$$p(y|\mathbf{x}, \mathcal{T}) \approx p(y|\mathbf{x}, \hat{\theta}(\mathcal{T}))$$

In un *modello non parametrico*, il numero di parametri cresce con la dimensione del training set: sostanzialmente, ogni singola previsione, in questo caso, richiede l'utilizzo dell'intero training set. Un esempio di approccio non parametrico sono i classificatori di tipo *nearest neighbor*, in cui la previsione y di $\mathbf{x}$ è determinata ponendola uguale al valore y_i dell'elemento $\mathbf{x}_i$ del training set più vicino a $\mathbf{x}$.

1.2.3 Modelli generativi e discriminativi

Possiamo considerare due modi per apprendere la distribuzione $p(y|\mathbf{x})$ da utilizzare per la previsione:

1. effettuare direttamente una stima di $p(y|\mathbf{x})$ dal training set; questo approccio è detto *discriminativo*, perchè, a partire da $\mathcal{T}$, viene derivata una caratterizzazione dell'output in funzione delle features, in modo tale da discriminare, dato un elemento, il più probabile tra i possibili valori dell'output

2. effettuare una stima di $p(\mathbf{x}|y)$ dal training set; questo approccio è detto *generativo*, perchè, a partire da $\mathcal{T}$, viene derivata una caratterizzazione delle features in funzione dell'output, in modo tale da generare, dato un possibile output, un elemento che con buona probabilità potrà essere associato a quell'output. Sostanzialmente, in un approccio generativo, viene derivato, per ogni possibile output, un modello (sotto forma di distribuzione di probabilità) degli elementi associati a quell'output. Il teorema di Bayes consente, a partire da $p(\mathbf{x}|y)$, e nota la distribuzione a priori $p(y)$, di ottenere $p(y|\mathbf{x})$, in quanto

$$p(y|\mathbf{x}) = \frac{p(\mathbf{x}|y)p(y)}{p(\mathbf{x})}$$

1.2.4 Unsupervised learning

Nel caso del learning unsupervised i dati consistono soltanto di un insieme di features $\mathbf{x}_i$, senza variabili di output y_i. Quel che si vuole è individuare un modello che si adatti ai dati, al fine di scoprire proprietà interessanti di essi.

Ad esempio, un algoritmo di unsupervised learning applicato ai dati a sinistra nella figura 4 (che rappresentano coppie altezza-peso) ci può mostrare l'esistenza di due *cluster* nella distribuzione, corrispondenti a maschi e femmine, e l'assegnazione dei vari elementi all'uno o all'altro cluster. I due cluster sono mostrati, in colori diversi, a destra nella figura.

In un problema di unsupervised learning, in assenza di un valore corretto di output che funga da riferimento, non è possibile misurare la correttezza della soluzione trovata rispetto ad una soluzione corretta (almeno nel training set e nel test set). Quel che è possibile effettuare è una valutazione del modello ottenuto, sulla base della likelihood media di tutti gli n elementi dell'insieme

$$L(\mathcal{M}|\mathcal{T}) = \frac{1}{n}\sum_{i=1}^{n} p(\mathbf{x}_i|\mathcal{M},\mathcal{T})$$

che misura la verosimiglianza dell'insieme di elementi considerato, rispetto al modello ottenuto.

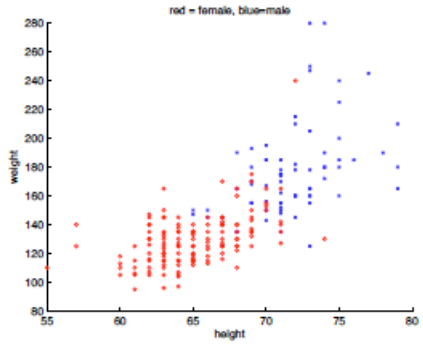

Figura 4: Esempio di classificazione binaria

2 REGRESSIONE

2.1 MODELLI LINEARI DI REGRESSIONE

Come detto, nel problema della regressione, dato un vettore di input $\mathbf{x}^T = (x_1, x_2, \ldots, x_d)$, si vuole predire l'output corrispondente individuando una funzione $y()$ tale che $y(\mathbf{x})$ non sia troppo diverso dal valore di output y associato a $\mathbf{x}$.

Nel caso di modelli lineari, considerato qui, ci limitiamo a funzioni $y()$ composte da una combinazione lineare dei valori in $\mathbf{x}$. Quindi, quel che si vuole, dato un vettore di input $\mathbf{x} = (x_1, x_2, \ldots, x_d)^T$, è predire l'output corrispondente mediante un modello del tipo:

$$y = y(x_1, \ldots, x_d) = w_0 + \sum_{j=1}^{d} x_j w_j$$

Se consideriamo il vettore $\overline{\mathbf{x}} = (1, x_1, x_2, \ldots, x_d)^T$ allora possiamo scrivere la relazione in forma matriciale:

$$y = y(\mathbf{x}) = \boldsymbol{w}^T \overline{\mathbf{x}}$$

con

$$\boldsymbol{w} = (w_0, w_1, \ldots, w_d)^T$$

Se l'output non è scalare, ma è un vettore $\mathbf{y}$ di dimensione q, allora vogliamo predire ogni singola componente $y_1, y_2, \ldots, y_q$ dell'output mediante un modello lineare:

$$y_l = h_l(x_1, \ldots, x_d) = w_{0l} + \sum_{j=0}^{d} w_{jl} x_j$$

In questo caso, abbiamo una matrice $\boldsymbol{W}$ di dimensioni $(d+1) \times q$ e la relazione in forma matriciale è data da:

$$y = \boldsymbol{W}^T \overline{\mathbf{x}}$$

Esempio: Costo anelli di diamanti rispetto a carati (sul mercato di Singapore).

Taglia in carati	Costo in SGD (dollari di Singapore), 1996
0.17	355
0.16	328
0.17	350
0.18	325
0.25	642
0.16	342
⋮	⋮

Vogliamo trovare una buona previsione del costo di un anello con diamante di 0.32 carati.

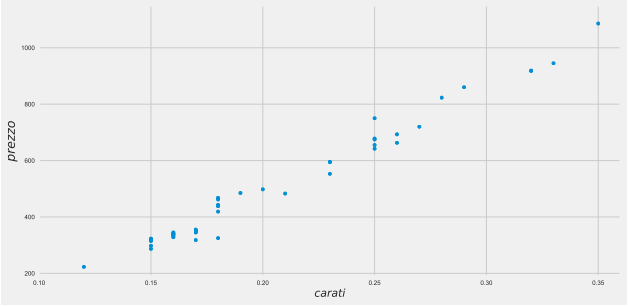

Figura 5: Dati osservati per l'esempio

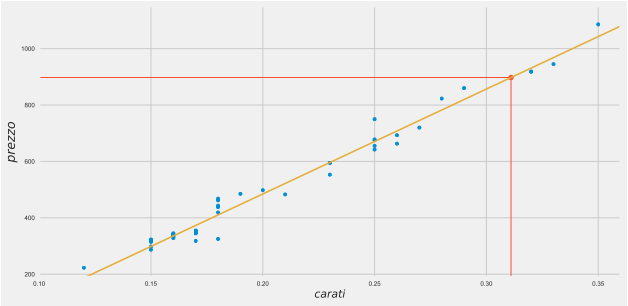

Figura 6: Retta di regressione lineare per l'esempio

È possibile derivare il modello lineare con $\hat{w}_0 = -259.6, \hat{w}_1 = 3721$, corrispondente alla retta $y = 3721x - 259.6$. In Figura 6 è mostrato il dataset di **??** con la retta trovata, utilizzata per effettuare la previsione del prezzo relativo a un diamante di 0.311 carati, che risulta pari a 897.6 dollari.

Come decidere i valori dei parametri $\boldsymbol{w} = (w_0, w_1)$ del modello? Il valore $\hat{\boldsymbol{w}}$ prescelto per $\boldsymbol{w}$ è quello che minimizza una *funzione di costo* preventivamente definita sui valori y_i osservati nel training set $\mathcal{T} = (\mathbf{X}, \mathbf{y})$ e su quelli predetti $\boldsymbol{w}^T \bar{\boldsymbol{x}}_i$ (dove $\bar{\boldsymbol{x}}_i = (1, x_i^{(1)}, \ldots, x_i^{(d)})^T$) .

La più diffusa funzione di costo è la semisomma dei quadrati dei residui (dove il coefficiente moltiplicativo 1/2 è inserito soltanto per semplicità nei calcoli successivi):

$$\begin{aligned} C(\boldsymbol{w}; \mathcal{T}) &= \frac{1}{2} \sum_{i=1}^{n} (y(\boldsymbol{x}_i) - y_i)^2 = \frac{1}{2} \sum_{i=1}^{n} (\boldsymbol{w}^T \bar{\boldsymbol{x}}_i - y_i)^2 \\ &= \frac{1}{2} \sum_{i=1}^{n} (w_0 + \sum_{j=1}^{d} w_j x_i^{(j)} - y_i)^2 \end{aligned}$$

Assumiamo per semplicità che ci sia una sola feature $x^{(1)} = x$, e quindi che

$$y = y(x) = w_0 + w_1 x$$

Allora,

$$C(\boldsymbol{w}; \mathcal{T}) = \frac{1}{2} \sum_{i=1}^{n} (y(\boldsymbol{x}_i) - y_i)^2 = \frac{1}{2} \sum_{i=1}^{n} (w_0 + x_i w_1 - y_i)^2$$

I valori di $\hat{w}_0$ e $\hat{w}_1$ possono essere ottenuti in modo semplice utilizzando il calcolo differenziale. Derivando rispetto a w_0 e ponendo la derivata pari a 0 otteniamo

$$\frac{\partial C(\boldsymbol{w};\mathcal{T})}{\partial w_0} = \sum_{i=1}^{n} (w_0 + x_i w_1 - y_i) = 0$$

e quindi, dividendo per n[1],

$$w_0 + \frac{w_1}{n}\sum_{i=1}^{n} x_i - \frac{1}{n}\sum_{i=1}^{n} y_i = w_0 + \overline{x} w_1 - \overline{y} = 0$$

Effettuando lo stesso rispetto a w_1 abbiamo

$$\frac{\partial C(\boldsymbol{w};\mathcal{T})}{\partial w_1} = \sum_{i=1}^{n} (w_0 + x_i w_1 - y_i) x_i = 0$$

e quindi, dividendo ancora per n[2],

$$\frac{w_0}{n}\sum_{i=1}^{n} x_i + \frac{w_1}{n}\sum_{i=1}^{n} x_i^2 - \frac{1}{n}\sum_{i=1}^{n} y_i x_i = w_0\overline{x} + w_1\overline{xx} - \overline{xy} = 0$$

Risolvendo il sistema lineare composto dalle due equazioni, otteniamo facilmente:

$$w_1 = \frac{\overline{x}\,\overline{y} - \overline{xy}}{\overline{x}^2 - \overline{xx}} \qquad \text{e} \qquad w_0 = \frac{\overline{xy}\,\overline{x} - \overline{y}\,\overline{xx}}{\overline{x}^2 - \overline{xx}}$$

Il fatto che $C(\boldsymbol{w};\mathcal{T})$ venga effettivamente minimizzato per tali valori è immediatamente verificabile osservando che le derivate seconde rispetto a w_0 e a w_1 sono rispettivamente [3]

$$\frac{\partial C(\boldsymbol{w};\mathcal{T})}{\partial w_0} = 1 \qquad \text{e} \qquad \frac{\partial C(\boldsymbol{w};\mathcal{T})}{\partial w_1} = \overline{xx}$$

e quindi positive ovunque.

2.1.1 Soluzione analitica

Ricordiamo che dato un array di variabili $\boldsymbol{x} = (x_1, \ldots, x_d)$ definite su $\mathbb{R}$, il *gradiente* di una funzione $f(\boldsymbol{x}) : \mathbb{R}^d \mapsto \mathbb{R}$, indicato come $\nabla_{\boldsymbol{x}} f(\boldsymbol{x})$ (o equivalentemente come $\frac{\partial f(\boldsymbol{x})}{\partial \boldsymbol{x}}$), è definito come un vettore di funzioni da $\mathbb{R}^d$ a $\mathbb{R}$ tale che

$$\nabla f(\boldsymbol{x}) = \left(\frac{\partial f(\boldsymbol{x})}{\partial x_1}, \frac{\partial f(\boldsymbol{x})}{\partial x_2}, \ldots, \frac{\partial f(\boldsymbol{x})}{\partial x_d}\right)^T \quad \text{e quindi } \nabla f(\boldsymbol{x})_i = \frac{\partial f(\boldsymbol{x})}{\partial x_i}$$

nel seguito, qualora il contesto chiarisca le variabili di derivazione $\boldsymbol{x}$, scriveremo per brevità $\nabla f(\boldsymbol{x})$.

La definizione si estende immediatamente al caso di una matrice $\boldsymbol{A}$ di $m \times n$ variabili reali e di una funzione $f : \mathbb{R}^{m\times n} \mapsto \mathbb{R}$, nel qual caso

$$\nabla f(\boldsymbol{A})_{ij} = \frac{\partial f(\boldsymbol{A})}{\partial a_{ij}}$$

Ricordiamo inoltre che, data una matrice quadrata $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, la *traccia* $\operatorname{Tr}\boldsymbol{A}$ di $\boldsymbol{A}$ è la somma degli elementi sulla diagonale principale

$$\operatorname{Tr}\boldsymbol{A} = \sum_{i=1}^{n} a_{ii}$$

1 $\overline{y}$ e $\overline{x}$ indicano le medie aritmetiche di $(y_1, \ldots, y_n)$ e di $(x_1, \ldots, x_n)$
2 $\overline{xy}$ e $\overline{xx}$ indicano le medie aritmetiche degli insiemi $(x_1y_1, \ldots, x_ny_n)$ e $(x_1^2, \ldots, x_n^2)$
3 Si ricorda che una funzione $y = f(x)$ ha un minimo in x_0 se $f'(x_0) = 0$ e $f''(x_0) > 0$

Infine, osservando che $\nabla_{A^T} f(A) = (\nabla_A f(A))^T$ e che $\nabla_A \operatorname{Tr}(ABA^TC) = CAB + C^TAB^T$ possiamo ottenere che

$$\begin{aligned}\nabla_{A^T} \operatorname{Tr}(ABA^TC) &= (\nabla_A \operatorname{Tr}(ABA^TC))^T = (CAB + C^TAB^T)^T \\ &= B^TA^TC^T + BA^TC\end{aligned}$$

Osserviamo ora che, se X è la matrice $n \times d$ delle osservazioni (e $\overline{X}$ è la matrice $n \times (d+1)$ ottenuta da essa anteponendo una colonna di tutti elementi unitari alle altre) e y il vettore dei corrispondenti valori, allora

$$\overline{X}w - y = \begin{bmatrix} w^T\overline{x}_1 \\ w^T\overline{x}_2 \\ \vdots \\ w^T\overline{x}_n \end{bmatrix} - \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} w^T\overline{x}_1 - y_1 \\ w^T\overline{x}_2 - y_2 \\ \vdots \\ w^T\overline{x}_n - y_n \end{bmatrix}$$

Dato che, in generale, per ogni vettore $z = (z_1, \ldots, z_m)^T$ si ha che $z^Tz = \sum_{i=1}^m z_i^2$, allora

$$\frac{1}{2}(\overline{X}w - y)^T(\overline{X}w - y) = \frac{1}{2}\sum_{i=1}^n (w^T\overline{x}_i - y_i)^2 = C(w; \mathcal{T})$$

Calcolare il minimo di $C(w; \mathcal{T})$ richiede il calcolo delle derivate rispetto a $w_0, \ldots, w_d$, e quindi il calcolo del gradiente di $C(w; \mathcal{T})$ rispetto al vettore w

Calcolo di $\nabla_w C(w; \mathcal{T})$

Il gradiente deriva nel modo seguente:

$$\begin{aligned}\nabla C(w; \mathcal{T}) &= \nabla\left(\frac{1}{2}(\overline{X}w - y)^T(\overline{X}w - y)\right) && (1)\\ &= \frac{1}{2}\nabla(w^T\overline{X}^T\overline{X}w - w^T\overline{X}^Ty - y^T\overline{X}w + y^Ty) && (2)\\ &= \frac{1}{2}\nabla \operatorname{Tr}(w^T\overline{X}^T\overline{X}w - w^T\overline{X}^Ty - y^T\overline{X}w + y^Ty) && (3)\\ &= \frac{1}{2}\nabla(\operatorname{Tr}(w^T\overline{X}^T\overline{X}w) - 2\operatorname{Tr}(y^T\overline{X}w)) && (4)\\ &= \frac{1}{2}(\overline{X}^T\overline{X}w + \overline{X}^T\overline{X}w - 2\overline{X}^Ty) && (5)\\ &= \overline{X}^T\overline{X}w - \overline{X}^Ty && (6)\end{aligned}$$

dove (2) deriva ricordando che $(A+B)^T = A^T + B^T$ e che $(AB)^T = B^TA^T$, (3) deriva in quanto la traccia di un numero reale è il numero stesso. La (4) risulta in quanto $\operatorname{Tr}(w^T\overline{X}^Ty) = \operatorname{Tr}(w^T\overline{X}^Ty)^T = \operatorname{Tr}(y^T\overline{X}w)$ e $\operatorname{Tr}(y^Ty)$ è un numero reale, indipendente da w, per cui $\nabla \operatorname{Tr}(y^Ty) = 0$. Si ottiene poi la (5) in quanto $\nabla_A \operatorname{Tr}(AB) = \nabla_A \operatorname{Tr}(BA) = B^T$ e inoltre, come mostrato sopra,

$$\nabla_{A^T} \operatorname{Tr}(ABA^TC) = B^TA^TC^T + BA^TC$$

e, ponendo $A = w^T$, $B = \overline{X}^T\overline{X} = B^T$ e $C = I$ (dove I è la matrice identità, in cui $i_{jk} = 0$ se $j \neq k$ e $i_{kk} = 1$), si ottiene

$$\nabla \operatorname{Tr}(w^T\overline{X}^T\overline{X}wI) = \nabla \operatorname{Tr}(w^T\overline{X}^T\overline{X}w) = \overline{X}^T\overline{X}w + \overline{X}^T\overline{X}w.$$

Affinché le derivate siano pari a 0, abbiamo che deve essere $\overline{X}^T\overline{X}w = \overline{X}^Ty$ e quindi, che

$$w = (\overline{X}^T\overline{X})^{-1}\overline{X}^Ty$$

dove evidentemente assumiamo che $\overline{X}^T\overline{X}$ sia non singolare, e quindi che tutte le sue colonne siano linearmente indipendenti.

Esiste un'interpretazione geometrica della derivazione di w da X e y: infatti, assumendo in modo naturale che $n > d$, abbiamo che le $d+1$ colonne

di $\overline{X}$ (ognuna delle quali è un vettore di n elementi reali) hanno uno span (sottospazio generato) di dimensione pari al rango di $\overline{X}$, e quindi a $d+1$ se $\overline{X}$ è non singolare.

Se indichiamo $\overline{X}w$ come y', allora possiamo osservare che il minimo di $C(w;\mathcal{T})$ si ha per

$$0 = \overline{X}^T\overline{X}w - \overline{X}^T y = \overline{X}^T(y' - y)$$

il che è vero se e solo se $y' - y$ è ortogonale rispetto a tutte le colonne di $\overline{X}$, e quindi allo span di tali colonne. Detto altrimenti, y' (e quindi $\overline{X}^T w$) è la proiezione di y sullo span delle colonne di $\overline{X}$, e può essere espresso come combinazione lineare delle colonne di $\overline{X}$. I coefficienti della combinazione lineare sono proprio gli elementi di w.

Di seguito è illustrato un esempio in cui $n = 3, d = 1$: i due vettori c_1, c_2 corrispondono alle due colonne di $\overline{X}$ e generano un sottospazio di dimensione 2. Come viene mostrato, il vettore y' è la proiezione di y su tale piano.

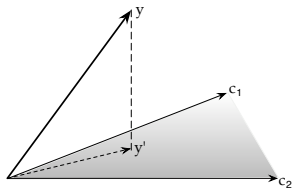

Figura 7: Interpretazione geometrica della regressione lineare per $\overline{X}^T\overline{X}$ non singolare

Si noti che, se $\overline{X}^T\overline{X}$ è singolare allora $|\overline{X}^T\overline{X}| = 0$ e, dato che $|\overline{X}^T| = |\overline{X}|$, allora $|\overline{X}| = 0$ e anche $\overline{X}$ è singolare. Quindi, se $\overline{X}^T\overline{X}$ è singolare, le $d+1$ colonne di $\overline{X}$ non sono linearmente indipendenti e il loro span ha dimensione minore di $d+1$. La figura seguente illustra questo caso, ancora per $n = 3, d = 1$, in cui le due colonne c_1, c_2 sono linearmente dipendenti, e hanno uno span di dimensione 1.

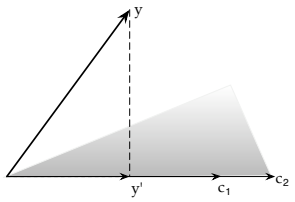

Figura 8: Interpretazione geometrica della regressione lineare per $\overline{X}^T\overline{X}$ singolare

Anche in questo caso, y' è comunque la proiezione di y sullo span delle colonne di $\overline{X}$: a differenza del caso precedente, y' può essere espresso in diversi modi in termini di combinazione lineare delle colonne di $\overline{X}$, e quindi corrisponde a diverse soluzioni per w.

2.1.2 Giustificazione probabilistica dei minimi quadrati

Perchè utilizzare proprio i minimi quadrati come funzione di costo? Una risposta può essere formulata sulla base di considerazioni probabilistiche.

Supponiamo che, dato un input $\boldsymbol{x}$ il cui output corretto è y, il valore predetto dal nostro modello sia $y(\boldsymbol{x})$: definiamo allora l'errore della previsione del modello come

$$\varepsilon = y(\boldsymbol{x}) - y = \boldsymbol{w}^T\boldsymbol{x} - y$$

dove ε rappresenta l'effetto sia di fenomeni non modellati che di rumore casuale.

Supponiamo che ε derivi da numerosi effetti indipendenti, e sia distribuito secondo una distribuzione gaussiana di media $\mu = 0$ e varianza σ^2, e quindi

$$p(\varepsilon) = \frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{\varepsilon^2}{2\sigma^2}\right)$$

Da ciò deriva che

$$p(y \mid \boldsymbol{x}, \boldsymbol{w}, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{(\boldsymbol{w}^T\boldsymbol{x} - y)^2}{2\sigma^2}\right)$$

Il modo comune di considerare questa espressione è in termini di funzione di y: in particolare, in questo caso, dato $\boldsymbol{w}$, ad ogni ogni valore di $\boldsymbol{x}$ è associata una funzione di y (che in particolare esprime una probabilità). Possiamo però considerare la stessa espressione come definizione di una funzione di $\boldsymbol{w}$: in questo caso, dato $\boldsymbol{x}$, ad ogni valore di y è associata una funzione di $\boldsymbol{w}$, denominata *verosimiglianza (likelihood)*.

$$L(\boldsymbol{w}; \boldsymbol{x}, y, \sigma^2) = p(y \mid \boldsymbol{x}, \boldsymbol{w}, \sigma^2)$$

Dato un training set $\mathcal{T} = (\mathbf{X}, \boldsymbol{y})$, se assumiamo che gli errori ε_i associati alle varie osservazioni siano mutuamente indipendenti, e quindi lo siano anche i valori y_i condizionati dalle osservazioni $\boldsymbol{x}_i$ otteniamo

$$\begin{aligned} L(\boldsymbol{w}) = p(\boldsymbol{y}|\mathbf{X}, \boldsymbol{w}, \sigma^2) &= \prod_{i=1}^{n} p(y_i \mid \boldsymbol{x}_i, \boldsymbol{w}, \sigma^2) \\ &= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{(y_i - \boldsymbol{w}^T\boldsymbol{x}_i)^2}{2\sigma^2}\right) \end{aligned}$$

Dato il modello precedente, ci chiediamo quale sia, nel caso in cui $\mathbf{X}$ e $\boldsymbol{y}$ siano conosciuti (come si assume siano), la migliore scelta di $\boldsymbol{w}$. A tal fine, applichiamo il *principio di massima verosimiglianza (maximum likelihood)*, secondo il quale conviene scegliere il valore di $\boldsymbol{w}$ che massimizza $L(\boldsymbol{w})$.

Si noti che, dato che la funzione logaritmo è monotona, per qualunque funzione $f(x)$ i suoi punti di massimo (o di minimo) corrispondono ai massimi (o ai minimi) di $\log f(x)$. Massimizzare $L(\boldsymbol{w})$ equivale quindi a massimizzare la *log-likelihood* $l(\boldsymbol{w}) = \log L(\boldsymbol{w})$ (la base del logaritmo è irrilevante). Nel nostro caso

$$\begin{aligned} l(\boldsymbol{w}) &= \sum_{i=1}^{n} \log\left(\frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{(y_i - \boldsymbol{w}^T\boldsymbol{x}_i)^2}{2\sigma^2}\right)\right) \\ &= n \log\frac{1}{\sqrt{2\pi}\sigma} - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \boldsymbol{w}^T\boldsymbol{x}_i)^2 \end{aligned}$$

Dato che il primo addendo $n \log\frac{1}{\sqrt{2\pi}\sigma}$ è costante rispetto a $\boldsymbol{w}$, massimizzare $l(\boldsymbol{w})$ equivale quindi a minimizzare i minimi quadrati

$$\frac{1}{2}\sum_{i=1}^{n}(y_i - \boldsymbol{w}^T\boldsymbol{x}_i)^2$$

2.1.3 Funzioni di base

Fino ad ora abbiamo considerato modelli definiti come combinazione lineare delle features. In effetti, questo approccio può essere generalizzato al caso in cui viene effettuata una combinazione lineare di un insieme predefinito di funzioni non lineari sulle features, dette *funzioni di base*. In questo caso cioè assumiamo di avere a disposizione $m-1$ funzioni $\phi_1, \ldots, \phi_{m-1}$, definite da $\mathbb{R}^d$ a $\mathbb{R}$, e consideriamo il modello:

$$y = y(x_1, \ldots, x_d) = w_0 + \sum_{j=1}^{m-1} w_j \phi_j(x_1, \ldots, x_d)$$

che, definendo $\mathbf{x} = (x_1, \ldots, x_d)^T$ e ponendo $\phi_0(\mathbf{x}) = 1$ per ogni $\mathbf{x}$, può essere scritto

$$y = \sum_{j=0}^{m-1} w_j \phi_j(\mathbf{x}) = \boldsymbol{w}^T \boldsymbol{\phi}(\mathbf{x})$$

dove $\boldsymbol{\phi}(\mathbf{x}) = (\phi_0(\mathbf{x}), \phi_1(\mathbf{x}), \ldots, \phi_{m-1}(\mathbf{x}))^T$

Un primo esempio di funzioni base è dato dalle funzioni polinomiali $\phi_i(x) = x^i$, altri casi sono le gaussiane

$$\phi_i(x) = exp\left(-\frac{(x-\mu_i)^2}{2\sigma^2}\right)$$

o le sigmoidali

$$\phi_i(x) = \frac{1}{1 + exp\left(\frac{-(x-\mu_i)}{\sigma}\right)}$$

Nelle figure 9, 10 e 11 sono mostrate famiglie di funzioni polinomiali, gaussiane e sigmoidali.

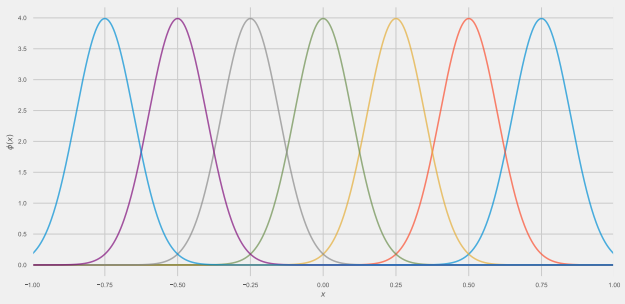

Figura 9: Funzioni base gaussiane

Si noti che, mentre le funzioni polinomiali sono tendenzialmente globali, nel senso che, per una variazione di x, cambia il valore restituito da tutte le funzioni, ciò non è vero per le gaussiane e le sigmoidali: infatti, per tali famiglie, soltanto un numero limitato di funzioni cambiano in modo significativo il proprio valore, per piccole variazioni di x.

Come nel caso di combinazioni lineari delle features, possiamo cercare di ottenere una buona scelta dei parametri in $\boldsymbol{w}$ minimizzando una funzione di costo (*loss function*), che possiamo assumere sia la usuale funzione "somma dei quadrati".

Consideriamo nuovamente che $y = y(\mathbf{x}) + \varepsilon$, con ε variabile casuale distribuita secondo una gaussiana di media 0 e varianza σ^2. Allora possiamo scrivere:

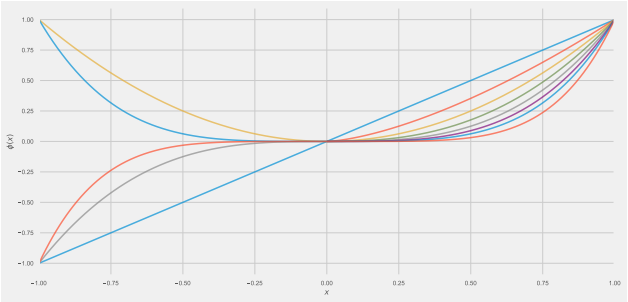

Figura 10: Funzioni base polinomiali

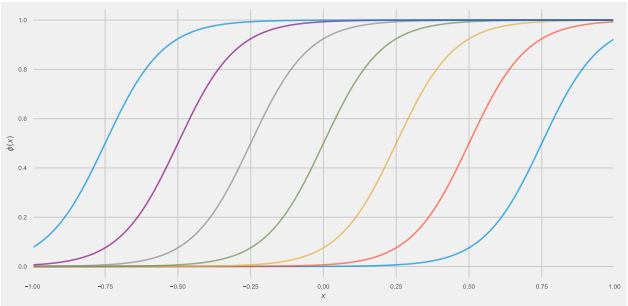

Figura 11: Funzioni base sigmoidali

$$
\begin{aligned}
p(y|x, w, \sigma^2) &= \frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{(y - y(x))^2}{2\sigma^2}\right) \\
&= \frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{(y - w^T \phi(x))^2}{2\sigma^2}\right)
\end{aligned}
$$

Chiaramente, dato che il valore atteso dell'errore è 0, il valore atteso di y sarà, per ipotesi, proprio il valore $y(x)$ restituito dal modello.

$$E[y|x] = \int t p(t|x) dt = y(x)$$

Dato un training set X, y, e assunto nuovamente che gli errori ε_i associati alle varie osservazioni siano mutuamente indipendenti, avremo allora che, per quanto riguarda la likelihood

$$L(w) = \prod_{i=1}^{n} p(y_i \mid x_i, w, \sigma^2) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{(y_i - w^T \phi(x_i))^2}{2\sigma^2}\right)$$

e, passando alla log-likelihood

$$
\begin{aligned}
l(w) &= \sum_{i=1}^{n} \log\left(\frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{(y_i - w^T \phi(x_i))^2}{2\sigma^2}\right)\right) \\
&= n \log \frac{1}{\sqrt{2\pi}\sigma} - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - w^T \phi(x_i))^2
\end{aligned}
$$

La massimizzazione di $l(w)$ come funzione di w equivale alla minimizzazione di

$$\frac{1}{2} \sum_{i=1}^{n} (y_i - w^T \phi(x_i))^2$$

Ripercorrendo quanto mostrato nella sezione 2.1.1 e considerando, al posto della matrice $\overline{\mathbf{X}}$, la matrice

$$\mathbf{\Phi} = \begin{bmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \phi_2(\mathbf{x}_1) & \cdots & \phi_{m-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & \phi_2(\mathbf{x}_2) & \cdots & \phi_{m-1}(\mathbf{x}_1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_n) & \phi_1(\mathbf{x}_n) & \phi_2(\mathbf{x}_n) & \cdots & \phi_{m-1}(\mathbf{x}_n) \end{bmatrix}$$

tale cioè che $\Phi_{ij} = \phi_{j-1}(\mathbf{x}_i)$, otteniamo che

$$\boldsymbol{w} = (\mathbf{\Phi}^T\mathbf{\Phi})^{-1}\mathbf{\Phi}^T\mathbf{y}$$

Il termine $(\mathbf{\Phi}^T\mathbf{\Phi})^{-1}\mathbf{\Phi}^T$ può essere visto come una generalizzazione della matrice inversa al caso di matrici non quadrate (e in generale non invertibili). Si noti infatti che, se $\mathbf{\Phi}$ è quadrata (e quindi $n = m$) e invertibile, allora $(\mathbf{\Phi}^T\mathbf{\Phi})^{-1}\mathbf{\Phi}^T = \mathbf{\Phi}^{-1}(\mathbf{\Phi}^T)^{-1}\mathbf{\Phi}^T = \mathbf{\Phi}^{-1}$.

La conoscenza del training set ci permette anche di effettuare la stima di σ^2 mediante massimizzazione della likelihood (in particolare della log-likelihood). In particolare, effettueremo la massimizzazione di $l(\boldsymbol{w})$ rispetto a $\gamma = \frac{1}{\sigma^2}$, dal che risulta:

$$\begin{aligned} \frac{\partial l(\boldsymbol{w})}{\partial \gamma} &= \frac{\partial}{\partial \gamma}\left(n \log \frac{\sqrt{\gamma}}{\sqrt{2\pi}} - \frac{\gamma}{2}\sum_{i=1}^{n}(y_i - \boldsymbol{w}^T\boldsymbol{\phi}(\mathbf{x}_i))^2\right) \\ &= \frac{\partial}{\partial \gamma}\left(\frac{n}{2}\log\gamma - \frac{n}{2}\log 2\pi - \frac{\gamma}{2}\sum_{i=1}^{n}(y_i - \boldsymbol{w}^T\boldsymbol{\phi}(\mathbf{x}_i))^2\right) \\ &= \frac{1}{2}\left(\frac{n}{\gamma} - \sum_{i=1}^{n}(y_i - \boldsymbol{w}^T\boldsymbol{\phi}(\mathbf{x}_i))^2\right) \end{aligned}$$

e, ponendo la derivata pari a 0, otteniamo

$$\sigma^2 = \frac{1}{\gamma} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \boldsymbol{w}^T\boldsymbol{\phi}(\mathbf{x}_i))^2$$

e quindi risulta che la migliore stima di σ^2 è data dalla varianza dei valori effettivi rispetto ai valori forniti dal modello.

2.1.4 Overfitting e regolazione

In generale, è possibile aumentare la rispondenza del modello rispetto ai dati nel training set aumentando opportunamente la complessità del modello stesso, ad esempio incrementando il numero di parametri.

Ad esempio, i dati corrispondenti ai punti in figura 12 sono derivati aggiungendo del rumore casuale alla funzione $y = \sin x \cos x$, riportata in figura. In particolare, i valori x_i di input sono stati generati da una distribuzione uniforme in $(0, 1)$: per ogni x_i il corrispondente valore y_i è stato ottenuto aggiungendo al valore della funzione $\sin 2\pi x_i$ una componente di rumore derivata da una distribuzione gaussiana con media $\mu = 0$ e deviazione standard $\sigma = 0,1$.

Vogliamo definire un modello che approssimi al meglio la funzione a partire dai 10 punti del training set. Consideriamo modelli definiti su funzioni base polinomiali

$$y(\mathbf{x}) = \sum_{i=0}^{d} w_i x^i$$

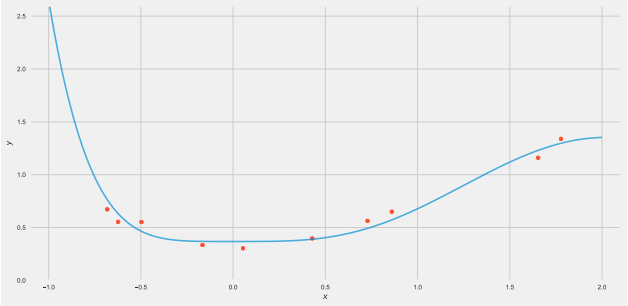

Figura 12: Training set da $y = \frac{1}{e} + e^{-x}x^3 \sin x$

con un valore d opportuno e parametri $w_0, \ldots, w_d$ derivati dal training set utilizzando le tecniche esposte sopra, basate sulla massimizzazione della likelihood (o sulla minimizzazione della funzione di costo "quadrati degli errori").

Nelle figure 13, 14, 15 sono riportati i polinomi approssimanti nei casi $d = 1$ (la normale regressione lineare), $d = 3$ e $d = 9$.

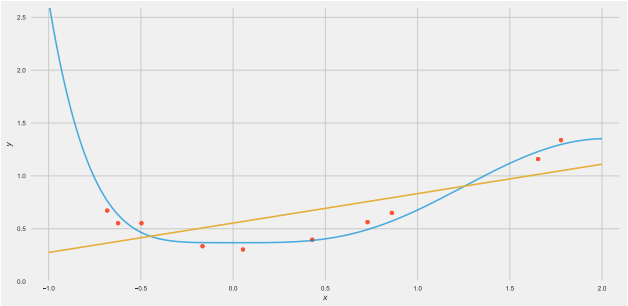

Figura 13: Approssimazione lineare

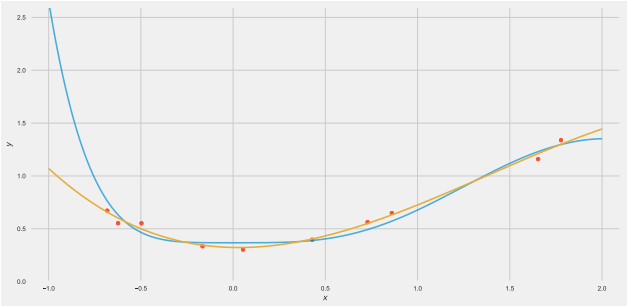

Figura 14: Approssimazione cubica

Come si può vedere, al crescere di d i polinomi approssimano sempre meglio i punti nel training set, fino a quando per $d = 9$ il numero di gradi di libertà del modello (10, tanti quanti i coefficienti $w_0, \ldots, w_9$) permette di ottenere una curva che passa esattamente per i punti del training set. Ciò è possibile in quanto per ogni insieme di n punti nel piano, esiste un polinomio di grado n che li attraversa. Si può osservare però che in tal caso, e in generale al crescere del grado, ci possono essere scostamenti anche molto

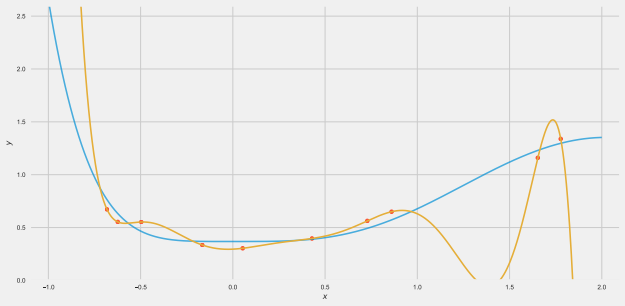

Figura 15: Approssimazione con polinomio di grado 9

importanti tra i valori della funzione e quelli del polinomio approssimante, soprattutto nelle regioni più lontane rispetto ai punti presenti nel training set. Come si può vedere in Figura 15, il polinomio di grado 9, per poter attraversare tutti i punti del training set, presenta delle marcate oscillazioni, risultano, complessivamente in una approssimazione povera della funzione $y = e^{-1} + e^{-x}x^3 \sin x$. Questo fenomeno, per il quale il tentativo di approssimare in modo molto preciso, oltre una certa soglia, i dati del training set (che a loro volta, per la presenza di rumore, sono soltanto un'approssimazione della funzione soggiacente) fa sì che l'approssimazione della funzione stessa diventi povera, viene denominato *overfitting*.

L'effetto dell'overfitting è visibile nella figura 16, nella quale sono stati riportati i valori della funzione di costo

$$E_{RMS} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - y(x_i))^2}{n}}$$

in cui la normalizzazione rispetto a n consente di comparare la funzione calcolata su insiemi di dimensione diversa. In figura viene riportato l'andamento del costo al crescere di d, calcolato sul training set di 10 punti e su un *validation set* (o *test set*) di 20 punti generati con la stessa procedura utilizzata per generare il training set. La funzione E_{RMS}, applicata sul validation set, ci fornisce una misura dell'errore indotto dalle scelte effettuate sulla base del training set (errore che, se calcolato sul training set, può essere arbitrariamente ridotto dando luogo a overfitting): tale errore viene denominato *errore di validazione*.

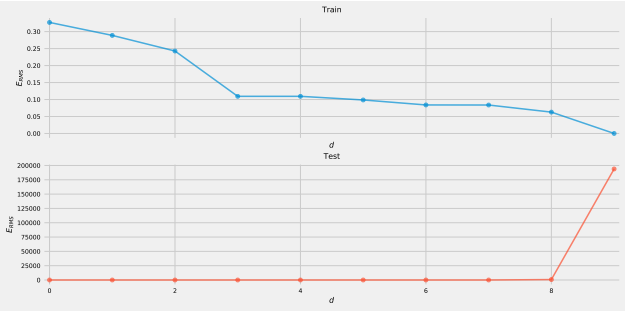

Figura 16: Valori della funzione di costo E_{RMS} per il training set

Come si può vedere, l'errore nell'approssimazione del training set diminuisce (come già osservato) al crescere di d, fino ad annullarsi per $d = 9$.

Al contrario, l'approssimazione del validation set, l'errore di validazione, peggiora da un certo punto in poi per effetto dell'overfitting.

In modo duale, si può osservare come, dato un modello con un certo numero di parametri (10, corrispondenti a $d = 9$ nel nostro caso), il fenomeno dell'overfitting vada a svanire al crescere della dimensione del training set. Nella figura 17, sono riportati i polinomi di 9 grado derivati da training set di 10, 15, 20 e 50 punti, rispettivamente, e si può notare come, al crescere della dimensione del training set, migliori l'approssimazione della funzione.

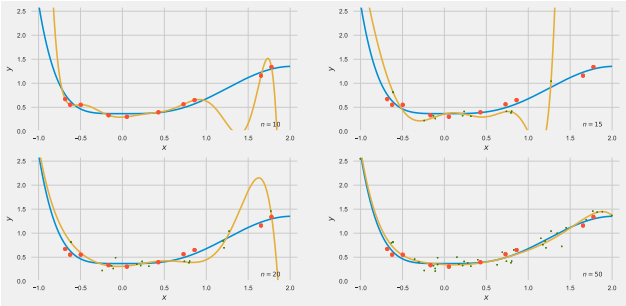

Figura 17: Overfitting al crescere del training set

Un ulteriore effetto dell'overfitting è l'assegnazione di valori numerici elevati ai coefficienti del modello, valori necessari per ottenere l'andamento della funzione che attraversa (o approssima fortemente) i punti nel training set. Se osserviamo i valori dei coefficienti per i modelli precedenti, riportati qui sotto, vediamo come al crescere di d tali valori possono essere decisamente elevati.

	$d = 0$	$d = 1$	$d = 3$	$d = 9$
$\hat{w}_0$	0.65	0.55	0.32	0.3
$\hat{w}_1$		0.28	-0.03	0.08
$\hat{w}_2$			0.58	1.49
$\hat{w}_3$			-0.14	-3.59
$\hat{w}_4$				-2.46
$\hat{w}_5$				15.11
$\hat{w}_6$				-5.56
$\hat{w}_7$				-16.12
$\hat{w}_8$				15.02
$\hat{w}_9$				-3.64

Per limitare l'effetto dell'overfitting, può essere opportuno aggiungere alla funzione di costo un termine di *regolazione* che tenda a far rimanere sufficientemente limitati i valori dei coefficienti in gioco:

$$\tilde{C}(\boldsymbol{w}; \mathcal{T}) = C(\boldsymbol{w}; \mathcal{T}) + \lambda \mathcal{E}(\boldsymbol{w})$$

il coefficiente λ prende il nome di *coefficiente di regolazione*.

Ad esempio, possiamo definire la funzione di costo da minimizzare come

$$\begin{aligned}\tilde{C}(\boldsymbol{w}; \mathcal{T}) &= \frac{1}{2} \sum_{i=1}^{n} (\boldsymbol{w}^T \phi_i(\boldsymbol{x}) - y_i)^2 + \frac{\lambda}{2} \sum_{j=0}^{M} w_j^2 \\ &= \frac{1}{2} (\boldsymbol{\Phi w} - \boldsymbol{y})^T (\boldsymbol{\Phi w} - \boldsymbol{y}) + \frac{\lambda}{2} \boldsymbol{w}^T \boldsymbol{w}\end{aligned}$$

Si può mostrare, applicando lo stesso procedimento descritto sopra per il caso $C(\boldsymbol{w};\mathcal{T}) = \frac{1}{2}(\boldsymbol{X}\boldsymbol{w}-\boldsymbol{y})^T(\boldsymbol{X}\boldsymbol{w}-\boldsymbol{y})$, che il minimo si ha pe

$$\boldsymbol{w} = (\lambda \mathbf{I} + \boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\boldsymbol{y}$$

In tabella riportiamo i coefficienti ottenuti minimizzando $\tilde{C}(\boldsymbol{w};\mathcal{T})$ per diversi valori di λ, nell'esempio precedente, con $d = 9$.

	$\lambda = 0$	$\lambda = 10^{-4}$	$\lambda = 10^{-2}$	$\lambda = 10$
$\hat{w}_0$	0.3	0.3	0.32	0.21
$\hat{w}_1$	0.08	-0.14	-0.09	-0.01
$\hat{w}_2$	1.49	0.91	0.55	0.08
$\hat{w}_3$	-3.59	0.12	0.0	0.01
$\hat{w}_4$	-2.46	-1.06	0.06	0.04
$\hat{w}_5$	15.11	0.63	-0.03	0.01
$\hat{w}_6$	-5.56	0.61	-0.03	0.02
$\hat{w}_7$	-16.12	-1.26	-0.04	0.01
$\hat{w}_8$	15.02	0.76	-0.0	0.01
$\hat{w}_9$	-3.64	-0.16	0.01	-0.01

Nella figura sottostante vengono poi visualizzate le relative curve di approssimazione

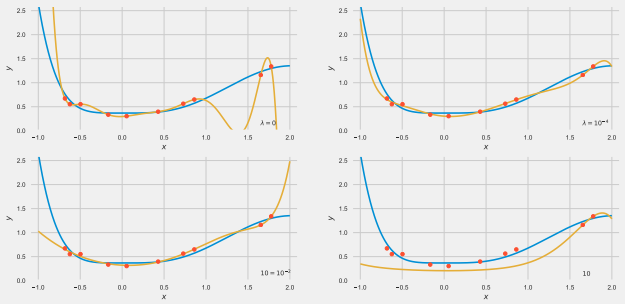

Figura 18: Effetto della regolazione sull'overfitting

Come si può vedere, ponendo $\ln\lambda = -5$ l'overfitting è scomparso e la funzione polinomiale trovata appare una approssimazione molto migliore di quella corrispondente a $\lambda = 0$. Al tempo stesso, vediamo come, per valori di λ più grandi (come per il caso $\ln\lambda = 2$ in figura), l'effetto del termine di regolazione diventa troppo consistente, rispetto a quello determinato dalla funzione di costo, e l'approssimazione peggiora.

Tutto ciò può essere visualizzato nella figura 19, dove viene riportato l'andamento della funzione E_{RMS} per i training set e validation set descritti sopra, nel caso $d = 9$, al crescere di λ. Come si può verificare, l'introduzione del termine di errore ($\lambda \neq 0$) porta a una diminuzione iniziale sostanziale dell'errore sul validation set, facendolo poi crescere successivamente, per valori più grandi di λ. Al tempo stesso, come ci si può aspettare, l'errore rispetto al training set aumenta in modo monotono al crescere di λ.

Il termine di regolazione introdotto sopra può essere generalizzato come

$$\frac{\lambda}{2}\sum_{j=0}^{d}|w_j|^q$$

che, al variare di q, fornisce termini di regolazione diversi, tra i quali abbiamo sopra considerato quello corrispondente al caso $q = 2$.

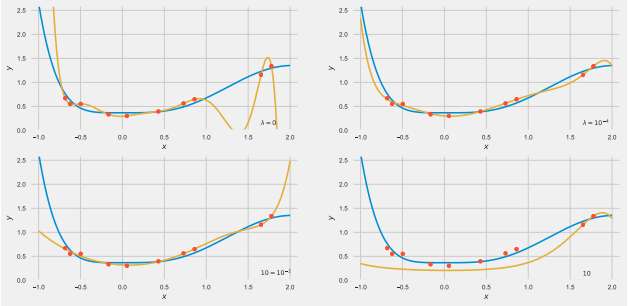

Figura 19: Errore in regressione lineare con regolazione, in funzione di λ

Una certa importanza riveste anche il caso $q = 1$, noto come *lasso* in letteratura, il quale presenta la proprietà che, per λ sufficientemente grande, alcuni termini w_i sono portati a0, conducendo ad una approssimazione *sparsa*, in cui soltanto alcune funzioni base concorrono.

Al fine di individuare il miglior valore per λ, un approccio molto comune è quello di verificare un insieme $\lambda_1, \lambda_2, \ldots, \lambda_t$ di possibili valori di λ determinando iterativamente per ogni λ_i il corrispondente modello (e quindi i coefficienti $\boldsymbol{w}$), utilizzando il training set, valutando l'errore di validazione per tale modello, usando il validation set, e selezionando il valore di λ che fornisce un modello il cui errore di validazione è minimo (tra quelli considerati).

2.1.5 La decomposizione errore sistematico-varianza

Come visto, nell'effettuare una regressione lineare il primo passo consiste nello scegliere uno specifico estimatore $y(\mathbf{x})$ del valore y per ogni input $\mathbf{x}$. Oltre a ciò, è necessario definire anche una funzione di costo $\mathcal{C}(y, y(\mathbf{x}))$: il rischio nella scelta di $y(\mathbf{x})$ come stimatore è allora dato dal costo medio rispetto a tutti i possibili elementi (features e output)

$$E_{\mathbf{x},y}[C(y, y(\mathbf{x}))] = \int\int C(y, y(\mathbf{x}))p(\mathbf{x}, y)d\mathbf{x}dy$$

Una scelta comune, come visto, per la funzione di costo è la funzione di costo quadratica

$$C(y, y(\mathbf{x})) = (y(\mathbf{x}) - y)^2$$

Vogliamo scegliere uno stimatore $y(\mathbf{x})$ che minimizzi il rischio.

Selezione dello stimatore $y(x)$

Osserviamo che

$$\begin{aligned} C(y, y(\mathbf{x})) &= (y(\mathbf{x}) - y)^2 = ((y(\mathbf{x}) - E_y[y|\mathbf{x}]) + (E_y[y|\mathbf{x}] - y))^2 \\ &= (y(\mathbf{x}) - E_y[y|\mathbf{x}])^2 + 2(y(\mathbf{x}) - E_y[y|\mathbf{x}])(E_y[y|\mathbf{x}] - y) \\ &\quad + (y - E_y[y|\mathbf{x}])^2 \end{aligned}$$

dove $E_y[y|\mathbf{x}]$ è il valore medio di y osservato in corrispondenza a più osservazioni dell'elemento con features $\mathbf{x}$ (si ricordi la presenza di rumore, che fa sì che tale valore possa essere diverso).

Quindi,

$$\begin{aligned}
E_{x,y}[C(y, y(x))] &= \int_x \int_y C(y, y(x)) p(x, y)\, dx\, dy \\
&= \int_x \int_y (y(x) - E_y[y|x])^2 p(x, y)\, dx\, dy \\
&\quad + \int_x \int_y 2(y(x) - E_y[y|x])(E_y[y|x] - y)\, dx\, dy \\
&\quad + \int_x \int_y (y - E_y[y|x])^2 p(x, y)\, dx\, dy
\end{aligned}$$

Allora,

$$\begin{aligned}
&\int_x \int_y (y(x) - E_y[y|x])^2 p(x, y)\, dx\, dy = \int_x \left(\int_y (y(x) - E_y[y|x])^2 p(x, y)\, dy \right) dx \\
&= \int_x \left((y(x) - E_y[y|x])^2 \int_y p(x, y)\, dy \right) dx = \int_x (y(x) - E_y[y|x])^2 p(x)\, dx
\end{aligned}$$

Inoltre, dato che $p(x, y) = p(y|x)p(x)$,

$$\begin{aligned}
&\int_x \int_y 2(y(x) - E_y[y|x])(E_y[y|x] - y) p(x, y)\, dx\, dy \\
&= 2 \int_x \int_y (y(x) - E_y[y|x])(E_y[y|x] - y) p(y|x) p(x)\, dx\, dy \\
&= 2 \int_x \left(\int_y (y(x) - E_y[y|x])(E_y[y|x] - y) p(y|x)\, dy \right) p(x)\, dx \\
&= 2 \int_x (y(x) - E_y[y|x]) \left(\int_y (E_y[y|x] - y) p(y|x)\, dy \right) p(x)\, dx \\
&= 2 \int_x (y(x) - E_y[y|x]) \left(\int_y E_y[y|x] p(y|x)\, dy - \int_y y p(y|x)\, dy \right) p(x)\, dx \\
&= 2 \int_x (y(x) - E_y[y|x]) \left(E_y[y|x] \int_y p(y|x)\, dy - E_y[y|x] \right) p(x)\, dx = 0
\end{aligned}$$

Infine,

$$\begin{aligned}
&\int_x \int_y (y - E_y[y|x])^2 p(x, y)\, dx\, dy = \int_x \int_y (y - E_y[y|x])^2 p(y|x) p(x)\, dx\, dy \\
&= \int_x \left(\int_y (y - E_y[y|x])^2 p(y|x)\, dy \right) p(x)\, dx = \int_x \sigma^2_{y|x} p(x)\, dx
\end{aligned}$$

Da tutto ciò deriva che il costo atteso è

$$\begin{aligned}
E_{x,y}[C(y, y(x))] &= \int_x (y(x) - E_y[y|x])^2 p(x)\, dx + \int_x \sigma^2_{y|x} p(x)\, dx \\
&= E_x[(y(x) - E_y[y|x])^2] + E_x[\sigma^2_{y|x}]
\end{aligned}$$

Esaminando la struttura del costo atteso

$$E_{x,y}[C(y, y(x))] = E_x[(y(x) - E_y[y|x])^2] + E_x[\sigma^2_{y|x}]$$

osserviamo che il primo addendo $E_x[(y(x) - E_y[y|x])^2]$ descrive il contributo all'errore dato dalla scelta di $y(x)$, e viene minimizzato (in effetti azzerato) scegliendo come stimatore la media condizionata, $E_y[y|x]$: si osservi però che la funzione di x $E_y[y|x]$, denominata *funzione di regressione*, è però sconosciuta. Il secondo addendo rappresenta la varianza media di y al variare di x, è indipendente da $y(x)$, e descrive il rumore nei dati: esso non può quindi essere modificato dalla scelta di $y(x)$, ed è intrinseco ai dati.

La funzione di regressione $E_y[y|x]$ è, come detto sopra, sconosciuta e può essere derivata a partire dai dati di training. In linea di principio, avendo disponibilità di una quantità indefinita di dati di training (oltre che di adeguata capacità di calcolo), tale funzione potrebbe essere derivata in modo arbitrariamente preciso ma, in presenza di dati di training limitati, la conoscenza della funzione di regressione sarà necessariamente approssimata

e, quindi, la valutazione del costo medio dovrà tener conto, come vedremo ora, di questa approssimazione.

Consideriamo un insieme di training set, ognuno composto di n elementi estratti indipendentemente sulla distribuzione $p(\mathbf{x}, y)$. A partire da un particolare training set $\mathcal{T}$, l'algoritmo di learning utilizzato deriverà uno stimatore $y(\mathbf{x}, \mathcal{T})$: le prestazioni dell'algoritmo di learning saranno allora misurate in termini di valore atteso, rispetto al training set considerato, di $y(\mathbf{x}, \mathcal{T})$: indichiamo con $E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})]$ tale valore atteso.

Consideriamo ora il primo termine $E_{\mathbf{x}}[(y(\mathbf{x}) - E_y[y|\mathbf{x}])^2]$ nell'espressione del costo atteso, e valutiamone il valore atteso al variare di $\mathcal{T}$.

Selezione dello stimatore $y(x)$

Il termine $E_{\mathbf{x}}[(y(\mathbf{x}) - E_y[y|\mathbf{x}])^2]$ prende, per un particolare training set $\mathcal{T}$, la forma

$$\begin{aligned}
&\int_{\mathbf{x}} (y(\mathbf{x}, \mathcal{T}) - E_y[y|\mathbf{x}])^2 p(\mathbf{x}) d\mathbf{x} \\
&= \int_{\mathbf{x}} (y(\mathbf{x},) - E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})] + E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})] - E_y[y|\mathbf{x}])^2 \, p(\mathbf{x}) d\mathbf{x} \\
&= \int_{\mathbf{x}} (y(\mathbf{x}, \mathcal{T}) - E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})])^2 p(\mathbf{x}) d\mathbf{x} + \int_{\mathbf{x}} (E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})] - E_y[y|\mathbf{x}])^2 p(\mathbf{x}) d\mathbf{x} \\
&\quad + 2 \int_{\mathbf{x}} (y(\mathbf{x}, \mathcal{T}) - E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})])(E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})] - E_y[y|\mathbf{x}]) p(\mathbf{x}) d\mathbf{x}
\end{aligned}$$

e, passando al valore atteso di tale espressione rispetto a $\mathcal{T}$,

$$\begin{aligned}
&\int_{\mathbf{x}} E_{\mathcal{T}}[(y(\mathbf{x}, \mathcal{T}) - E_y[y|\mathbf{x}])^2] p(\mathbf{x}) d\mathbf{x} \\
&= \int_{\mathbf{x}} E_{\mathcal{T}}[(y(\mathbf{x}, \mathcal{T}) - E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})])^2] p(\mathbf{x}) d\mathbf{x} \\
&\quad + \int_{\mathbf{x}} E_{\mathcal{T}}[(E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})] - E_y[y|\mathbf{x}])^2] p(\mathbf{x}) d\mathbf{x} \\
&\quad + 2 \int_{\mathbf{x}} E_{\mathcal{T}}[(y(\mathbf{x}, \mathcal{T}) - E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})])(E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})] - E_y[y|\mathbf{x}])] p(\mathbf{x}) d\mathbf{x} \\
&= \int_{\mathbf{x}} E_{\mathcal{T}}[(y(\mathbf{x}, \mathcal{T}) - E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})])^2] p(\mathbf{x}) d\mathbf{x} \\
&\quad + \int_{\mathbf{x}} (E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})] - E_y[y|\mathbf{x}])^2 p(\mathbf{x}) d\mathbf{x} \\
&\quad + 2 \int_{\mathbf{x}} (E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})] - E_y[y|\mathbf{x}]) E_{\mathcal{T}}[(y(\mathbf{x}, \mathcal{T}) - E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})])] p(\mathbf{x}) d\mathbf{x} \\
&= \int_{\mathbf{x}} E_{\mathcal{T}}[(y(\mathbf{x}, \mathcal{T}) - E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})])^2] p(\mathbf{x}) d\mathbf{x} + \int_{\mathbf{x}} (E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})] - E_y[y|\mathbf{x}])^2 p(\mathbf{x}) d\mathbf{x}
\end{aligned}$$

Ottenuta l'uguaglianza

$$\begin{aligned}
&\int_{\mathbf{x}} E_{\mathcal{T}}[(y(\mathbf{x}, \mathcal{T}) - E_y[y|\mathbf{x}])^2] p(\mathbf{x}) d\mathbf{x} \\
&= \int_{\mathbf{x}} E_{\mathcal{T}}[(y(\mathbf{x}, \mathcal{T}) - E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})])^2] p(\mathbf{x}) d\mathbf{x} + \int_{\mathbf{x}} (E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})] - E_y[y|\mathbf{x}])^2 p(\mathbf{x}) d\mathbf{x}
\end{aligned}$$

indichiamo il primo termine

$$\int_{\mathbf{x}} (E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})] - E_y[y|\mathbf{x}])^2 p(\mathbf{x}) d\mathbf{x}$$

come *errore sistematico* (o *bias*): esso misura il quadrato della differenza attesa (sull'elemento $\mathbf{x}$) tra il valore restituito dallo stimatore derivato in media applicando l'algoritmo di learning ai diversi training set e il valore effettivo della funzione di regressione: sostanzialmente, esso fornisce quindi una indicazione della rispondenza dello stimatore derivato rispetto alla funzione di regressione.

Il secondo termine

$$\int_{\mathbf{x}} E_{\mathcal{T}}[(y(\mathbf{x}, \mathcal{T}) - E_{\mathcal{T}}[y(\mathbf{x}, \mathcal{T})])^2] p(\mathbf{x}) d\mathbf{x}$$

è detto *varianza*, e misura di quanto le soluzioni fornite dagli stimatori derivati da diversi training set variano intorno alla loro media: esso indica quindi quanto siano diversi stimatori derivati da diversi training set.

L'obiettivo che ci poniamo è evidentemente quello di minimizzare la somma "*errore sistematico+varianza*": possiamo però renderci conto dell'esistenza di un trade-off tra queste due grandezze.

Ad esempio, consideriamo il caso della funzione $y = \sin 2\pi x$ e supponiamo che $y(\mathbf{x})$ sia derivabile da un training set di 25 punti, estratto da una collezione di $L = 100$ diversi insiemi di punti con ascissa generata uniformemente in $(0, 1)$ e ordinata derivata aggiungendo un rumore gaussiano al valore $\sin 2\pi x$. Supponiamo che, per ogni training set $\mathcal{T}_i$, la funzione $h_i(\mathbf{x})$ venga derivata minimizzando la funzione di costo regolarizzata

$$\tilde{C}(\boldsymbol{w}; \mathcal{T}) = \frac{1}{2}(\boldsymbol{\Phi w} - \mathbf{y})^T(\boldsymbol{\Phi w} - \mathbf{y}) + \frac{\lambda}{2}\boldsymbol{w}^T\boldsymbol{w}$$

La figura 20 mostra a sinistra un possibile andamento delle 100 funzioni $h_i(\mathbf{x})$ derivate a partire dai 100 training set $\mathcal{T}_i$, $i = 1, \ldots, 100$ ponendo $\ln \lambda = 2.6$, e a destra la funzione risultante dalla media delle $h_i(\mathbf{x})$, insieme alla $y = \sin 2\pi x$.

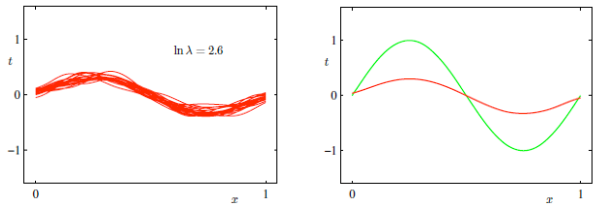

Figura 20: Trade-off errore sistematico/varianza

Come si può vedere, le funzioni $h_i(\mathbf{x})$ tendono a non discostarsi molto l'una dall'altra (varianza bassa), ma la loro media è piuttosto lontana dalla funzione da approssimare (errore sistematico alto). Mostriamo poi le stesse funzioni derivate con $\ln \lambda = -0.31$ e con $\ln \lambda = -2.4$.

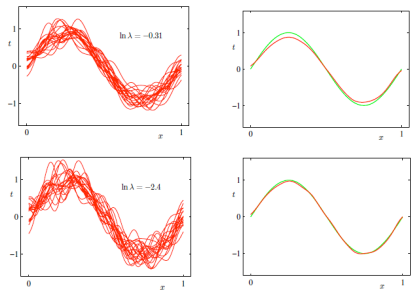

Figura 21: Ancora sul trade-off errore sistematico/varianza

Possiamo osservare come, al diminuire di λ, la varianza aumenti (le curve $h_i(\mathbf{x})$ tendono ad essere diverse tra loro), mentre l'errore sistematico diminuisce (la loro media approssima meglio $y = \sin 2\pi x$). Ricordiamo comunque come la presenza del termine di rumore $E_{\mathbf{x}}[\sigma^2_{y|\mathbf{x}}]$ ponga un limite alla approssimabilità di $y = \sin 2\pi x$.

Nella figura 22 riportato l'andamento dell'errore sistematico (linea continua), della varianza (linea tratteggiata) e della loro somma (linea a puntini)

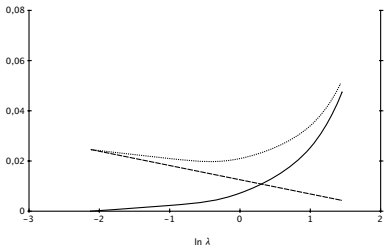

Figura 22: Andamento di errore sistematico e varianza al crescere di λ nell'esempio considerato

al crescere di λ: possiamo notare che, in effetti, l'errore sistematico aumenta al crescere del coefficiente di regolazione, mentre la varianza diminuisce. La loro somma presenta un minimo in corrispondenza al valore ottimo di λ.

2.1.6 Regressione lineare e statistica bayesiana

Come osservato in precedenza, l'utilizzo del criterio di massima verosimiglianza per la determinazione dei parametri del modello tende tipicamente ad essere soggetto al problema dell'overfitting, in particolare in presenza di training set di dimensioni non elevate (in rapporto al numero di parametri).

Per controllare la complessità del modello, un approccio di tipo bayesiano, invece di fare uso del termine di regolazione $\mathcal{E}(\boldsymbol{w})$, introduce una distribuzione che rappresenta la nostra valutazione a priori (prima dell'osservazione del training set) dei valori dei coefficienti $\boldsymbol{w}$

$$p(\boldsymbol{w}|\alpha) = \prod_{i=0}^{M} \left(\frac{\alpha}{2\pi}\right)^{1/2} exp\left(-\frac{\alpha}{2} w_i^2\right)$$

che equivale ad assumere che $\boldsymbol{w}$ sia distribuito secondo una gaussiana multivariata

$$f(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{n/2} \mid \boldsymbol{\Sigma} \mid^{1/2}} exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)$$

con parametri

$$\boldsymbol{\mu} = \mathbf{0} \qquad\qquad \boldsymbol{\Sigma} = \begin{bmatrix} \alpha^{-1} & 0 & \cdots & 0 \\ 0 & \alpha^{-1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \alpha^{-1} \end{bmatrix}$$

di una situazione, quindi, in cui si assume che i coefficienti $\boldsymbol{w}$ siano tra loro indipendenti e tutti distribuiti allo stesso modo, secondo una gaussiana a media 0 e varianza $\sigma^2 = \alpha^{-1}$. Come si può osservare, la distribuzione a priori è definita in modo parametrico, utilizzando l'iperparametro α, che, essendo inversamente proporzionale alla varianza, misura il grado di certezza nella valutazione dei valori di $\boldsymbol{w}$.

In precedenza abbiamo visto come, data la funzione di costo e il coefficiente di regolazione, sia possibile effettuare una stima di punto per i coefficienti $\boldsymbol{w}$, vale a dire determinare un valore per tali coefficienti. Quel che ora è invece possibile fare, in presenza di una distribuzione a priori sui valori dei coefficienti e della likelihood

$$p(\mathbf{y}|\boldsymbol{\Phi}, \boldsymbol{w}, \sigma) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{(y_i - \boldsymbol{w}^T \boldsymbol{\phi}(x_i))^2}{2\sigma^2}\right)$$

è determinare una distribuzione a posteriori per i coefficienti $\boldsymbol{w}$ applicando il teorema di Bayes

$$p(\boldsymbol{w}|\boldsymbol{y}, \boldsymbol{\Phi}, \alpha, \sigma) = \frac{p(\boldsymbol{y}|\boldsymbol{\Phi}, \boldsymbol{w}, \sigma)p(\boldsymbol{w}|\alpha)}{p(\boldsymbol{y}|\boldsymbol{\Phi}, \alpha, \sigma)} \propto p(\boldsymbol{y}|\boldsymbol{\Phi}, \boldsymbol{w}, \sigma)p(\boldsymbol{w}|\alpha)$$

Dato che sia $p(\boldsymbol{y}|\boldsymbol{\Phi}, \boldsymbol{w}, \sigma)$ che $p(\boldsymbol{w}|\alpha)$ sono gaussiane, allora anche $p(\boldsymbol{w}|\boldsymbol{y}, \boldsymbol{\Phi}, \alpha, \sigma)$ è gaussiana e, ricordando che se $\boldsymbol{x}, \boldsymbol{y}$ sono tali che

$$\boldsymbol{x} \sim \mathit{Normal}(\boldsymbol{\mu}, \boldsymbol{\Sigma}_1) \qquad \boldsymbol{y}|\boldsymbol{x} \sim \mathit{Normal}(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}, \boldsymbol{\Sigma}_2)$$

allora

$$\boldsymbol{x}|\boldsymbol{y} \sim \mathit{Normal}(\overline{\boldsymbol{\mu}}, \overline{\boldsymbol{\Sigma}})$$

con

$$\begin{aligned}
\overline{\boldsymbol{\mu}} &= (\boldsymbol{\Sigma}_1^{-1} + \boldsymbol{A}^T\boldsymbol{\Sigma}_2^{-1}\boldsymbol{A})^{-1}(\boldsymbol{A}^T\boldsymbol{\Sigma}_2^{-1}(\boldsymbol{y} - \boldsymbol{b}) + \boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}) \\
\overline{\boldsymbol{\Sigma}} &= (\boldsymbol{\Sigma}_1^{-1} + \boldsymbol{A}^T\boldsymbol{\Sigma}_2^{-1}\boldsymbol{A})^{-1}
\end{aligned}$$

osserviamo che, nel nostro caso, $\boldsymbol{x} = \boldsymbol{w}$, $\boldsymbol{A} = \boldsymbol{\Phi}$, $\boldsymbol{b} = 0$, $\boldsymbol{\Sigma}_2 = \sigma^2\boldsymbol{I}$, $\boldsymbol{\mu} = 0$, $\boldsymbol{\Sigma}_1 = \alpha^{-1}\boldsymbol{I}$, otteniamo che $p(\boldsymbol{w}|\boldsymbol{y}, \boldsymbol{\Phi}, \alpha, \sigma)$ ha matrice di covarianza

$$\begin{aligned}
\overline{\boldsymbol{\Sigma}} &= (\alpha\boldsymbol{I} + \boldsymbol{\Phi}^T\sigma^{-2}\boldsymbol{I}\boldsymbol{\Phi})^{-1} \\
&= (\alpha\boldsymbol{I} + \sigma^{-2}\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}
\end{aligned}$$

e vettore delle medie

$$\begin{aligned}
\overline{\boldsymbol{\mu}} &= (\alpha\boldsymbol{I} + \boldsymbol{\Phi}^T\sigma^{-2}\boldsymbol{I}\boldsymbol{\Phi})^{-1}(\boldsymbol{\Phi}^T\sigma^{-2}\boldsymbol{I}\boldsymbol{y} + \alpha\boldsymbol{I}0) \\
&= (\alpha\boldsymbol{I} + \sigma^{-2}\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\sigma^{-2}\boldsymbol{\Phi}^T\boldsymbol{y} \\
&= \sigma^{-2}(\alpha\boldsymbol{I} + \sigma^{-2}\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\boldsymbol{y} \\
&= \sigma^{-2}\overline{\boldsymbol{\Sigma}}\boldsymbol{\Phi}^T\boldsymbol{y}
\end{aligned}$$

Una volta nota la distribuzione a posteriori dei coefficienti $\boldsymbol{w}$, possiamo pensare di considerare come valori da assegnare a tali coefficienti i valori corrispondenti al vettore che massimizza la probabilità rispetto alla distribuzione $p(\boldsymbol{w}|\boldsymbol{y}, \boldsymbol{\Phi}, \alpha, \sigma)$ stessa: tale vettore è detto *MAP (maximum a posteriori)*.

Individuare il massimo di $p(\boldsymbol{w}|\boldsymbol{y}, \boldsymbol{\Phi}, \alpha, \sigma)$ rispetto a $\boldsymbol{w}$ equivale, dato che il denominatore nel teorema di Bayes è indipendente da $\boldsymbol{w}$, a massimizzare il numeratore, vale a dire il prodotto tra $p(\boldsymbol{y}|\boldsymbol{\Phi}, \boldsymbol{w}, \sigma)$ e $p(\boldsymbol{w}|\alpha)$ o, in modo equivalente, a minimizzare il negativo del logaritmo del prodotto:

$$-\log p(\boldsymbol{y}|\boldsymbol{\Phi}, \boldsymbol{w}, \sigma) - \log p(\boldsymbol{w}|\alpha)$$

non considerando i termini costanti rispetto a $\boldsymbol{w}$, che non hanno effetto ai fini della minimizzazione, otteniamo

$$E_{MAP}(\boldsymbol{w}) = \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \boldsymbol{w}^T\boldsymbol{\phi}(x_i)) + \frac{\alpha}{2}\boldsymbol{w}^T\boldsymbol{w}$$

come si può vedere, ciò equivale ad applicare un coefficiente di regolazione $\lambda = \sigma^2\alpha$.

In realtà, è possibile anche considerare distribuzioni a priori gaussiane generali $\boldsymbol{w} \sim \mathit{Normal}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$: in questo caso, si ha ancora che $p(\boldsymbol{w}|\boldsymbol{y}, \boldsymbol{\Phi}, \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0, \sigma)$ ha distribuzione gaussiana. Osservando che, rispetto al caso generale, $\boldsymbol{x} = \boldsymbol{w}$, $\boldsymbol{A} = \boldsymbol{\Phi}$, $\boldsymbol{b} = 0$, $\boldsymbol{\Sigma}_2 = \sigma^2\boldsymbol{I}$, $\boldsymbol{\mu} = \boldsymbol{\mu}_0$, $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_0$, otteniamo che $p(\boldsymbol{w}|\boldsymbol{y}, \boldsymbol{\Phi}, \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0, \sigma)$ ha matrice di covarianza

$$\overline{\boldsymbol{\Sigma}} = (\boldsymbol{\Sigma}_0^{-1} + \boldsymbol{\Phi}^T\sigma^{-2}\boldsymbol{I}\boldsymbol{\Phi})^{-1} = (\boldsymbol{\Sigma}_0^{-1} + \sigma^{-2}\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}$$

e vettore delle medie

$$\begin{aligned}\overline{\boldsymbol{\mu}} &= (\boldsymbol{\Sigma}_0^{-1} + \boldsymbol{\Phi}^T \sigma^{-2} \mathbf{I} \boldsymbol{\Phi})^{-1} (\boldsymbol{\Phi}^T \sigma^{-2} \mathbf{I} \mathbf{y} + \boldsymbol{\Sigma}_0^{-1} \boldsymbol{\mu}_0) \\ &= (\boldsymbol{\Sigma}_0^{-1} + \sigma^{-2} \boldsymbol{\Phi}^T \boldsymbol{\Phi})^{-1} (\sigma^{-2} \boldsymbol{\Phi}^T \mathbf{y} + \boldsymbol{\Sigma}_0^{-1} \boldsymbol{\mu}_0) \\ &= \overline{\boldsymbol{\Sigma}} (\sigma^{-2} \boldsymbol{\Phi}^T \mathbf{y} + \boldsymbol{\Sigma}_0^{-1} \boldsymbol{\mu}_0)\end{aligned}$$

L'approccio bayesiano può essere applicato anche in una situazione in cui la conoscenza del training set avviene in modo incrementale, un elemento dopo l'altro (*sequential learning*).

Partiamo dall'osservazione che, se la distribuzione a priori e la likelihood sono gaussiane, la distribuzione a posteriori rimane anch'essa gaussiana. Inoltre, se consideriamo due training set indipendenti T_1, T_2, che assumiamo vengano conosciuti in sequenza, abbiamo

$$p(\boldsymbol{w}|T_1, T_2) \propto p(T_1, T_2|\boldsymbol{w}) p(\boldsymbol{w}) = p(T_2|\boldsymbol{w}) p(T_1|\boldsymbol{w}) p(\boldsymbol{w}) \propto p(T_2|\boldsymbol{w}) p(\boldsymbol{w}|T_1)$$

dove $p(\boldsymbol{w})$ è la distribuzione a priori, prima dell'osservazione dei training set, $p(\boldsymbol{w}|T_1)$ è la distribuzione a posteriori dell'osservazione di T_1 e $p(T_2|\boldsymbol{w})$ è la likelihood di T_2. Quindi, la distribuzione a posteriori dell'osservazione di T_1 può essere utilizzata come distribuzione a priori, insieme alla likelihood del nuovo training set osservato, per una nuova applicazione del metodo. In generale, per una successione $T_1, \ldots, T_n$ di training set osservati, avremo

$$\begin{aligned}&p(\boldsymbol{w}|T_1, \ldots T_n) \propto p(T_n|\boldsymbol{w}) p(\boldsymbol{w}|T_1, \ldots T_{n-1}) \\ &p(\boldsymbol{w}|T_1, \ldots T_{n-1}) \propto p(T_{n-1}|\boldsymbol{w}) p(\boldsymbol{w}|T_1, \ldots T_{n-2}) \\ &\ldots \\ &p(\boldsymbol{w}|T_1) \propto p(T_1|\boldsymbol{w}) p(\boldsymbol{w})\end{aligned}$$

Consideriamo ad esempio il caso di una variabile di input x, una variabile di output y e una regressione lineare $y(x, w_0, w_1) = w_0 + w_1 x$. Nel nostro esempio, i dati sono generati dalla funzione $y = a_0 + a_1 x$ (con $a_0 = -0.3$ e $a_1 = 0.5$) scegliendo i valori x in modo uniforme in $[-1, 1]$, valutando y e aggiungendo rumore gaussiano con $\mu = 0$ e $\sigma = 0.2$.

Assumiamo, seguendo quanto esposto sopra, che la distribuzione a priori $p(w_0, w_1)$ dei coefficienti sia una gaussiana bivariata con $\boldsymbol{\mu} = \mathbf{0}$ e $\boldsymbol{\Sigma} = \sigma^2 \mathbf{I} = 0.04\mathbf{I}$: in figura **??** sono mostrate la distribuzione e un insieme di 6 rette campionate da essa.

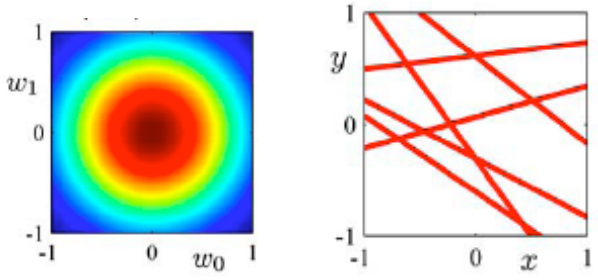

Figura 23: Distribuzione a priori di w_0 e w_1 e rette campionate da essa

In figura 24 è illustrato come tale distribuzione, e il conseguente campionamento su di essa di rette, vari al crescere del numero di punti osservati.

L'osservazione di un elemento (x_1, y_1) del training set (corrispondente al punto contrassegnato da un cerchio nell'immagine a destra in prima riga

nella **??**), determina una distribuzione a posteriori per $p(w_0, w_1|x_1, y_1)$ mostrata a sinistra nella stessa riga: a destra è fornito un insieme di 6 rette campionate su tale distribuzione.

L'osservazione di un ulteriore elemento (x_2, y_2) del training set (corrispondente all'ulteriore punto contrassegnato da un cerchio nell'immagine a destra in seconda riga di figura **??**), determina una distribuzione a posteriori per $p(w_0, w_1|x_1, y_1, x_2, y_2)$ mostrata a sinistra nella stessa riga: a destra è fornito un insieme di 6 rette campionate su tale distribuzione.

Dopo l'osservazione di n elementi $(x_1, y_1), \ldots (x_n, y_n)$ del training set (corrispondenti ai punti contrassegnati da un cerchio nell'immagine a destra nell'ultima riga di figura **??**), è stata determinata una distribuzione a posteriori per $p(w_0, w_1|x_1, y_1, \ldots, x_n, y_n)$ mostrata a sinistra nella stessa riga: a destra è fornito un insieme di 6 rette campionate su tale distribuzione.

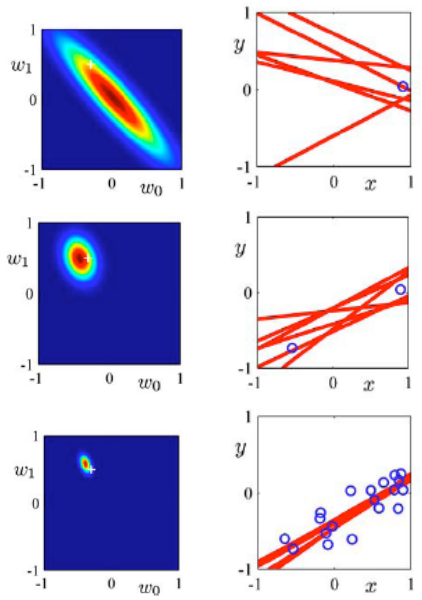

Figura 24: Andamento della distribuzione di w_0 e w_1 e delle rette campionate da essa

Come si può osservare, al crescere del numero di elementi osservati la distribuzione di w_0 e w_1 tende sempre più a spostarsi verso una media corrispondente al punto a_0, a_1 e, al tempo stesso, la varianza diminuisce, concentrando la distribuzione intorno alla media. Allo stesso tempo, in conseguenza di ciò, le rette estratte campionando la distribuzione tendono sempre più a concentrarsi intorno ad una unica soluzione, che si avvicina alla retta $y = a_0 + a_1 x$.

In figura **??** lo stesso processo viene mostrato per un problema di regressione lineare con funzioni di base gaussiane. La prima colonna mostra il training set, con gli elementi osservati evidenziati; la seconda colonna mostra la distribuzione per due coefficienti w_0, w_1 relativi alle gaussiane evidenziate nella prima colonna; la terza colonna mostra campioni estratti dalla distribuzione attuale dei coefficienti, con la soluzione corrispondente alla MAP.

Come visto, nell'approccio classico (frequentista) il valore $\hat{\boldsymbol{w}}_{LS}$ per i coefficienti $\boldsymbol{w}$ viene appreso effettuando una stima puntuale mediante minimizzazione della funzione quadratica di costo (e equivalentemente, di massi-

mizzazione della likelihood), eventualmente regolarizzata: tale valore viene poi utilizzato, dato un valore $\mathbf{x}$, per effettuare la previsione $y = \hat{\boldsymbol{w}}_{LS}^{T}\bar{\mathbf{x}}$.

Nell'approccio bayesiano, così come esaminato finora, viene determinata la distribuzione a posteriori $p(\boldsymbol{w}|\mathbf{y}, \boldsymbol{\Phi}, \alpha, \sigma)$ e su questa effettuata ancora una stima puntuale, valutando il valore atteso $\hat{\boldsymbol{w}}_{MAP}$ della distribuzione a posteriori (MAP): come osservato sopra, i due approcci sono equivalenti, in quanto $\hat{\boldsymbol{w}}_{MAP}$ corrisponde a $\hat{\boldsymbol{w}}_{LS}$ nel caso $\lambda = \sigma^2\alpha$. La predizione, dato un valore $\mathbf{x}$, è una distribuzione di probabilità gaussiana $p(y|\hat{\boldsymbol{w}}_{MAP}, \sigma^2)$ per y, con media $\boldsymbol{w}_{MAP}\mathbf{x}^{T}$ e varianza σ^2. La media di tale distribuzione è comunque pari al valore calcolato secondo l'approccio classico.

Come si può vedere, la distribuzione di probabilità per y non è derivata direttamente dalla distribuzione a posteriori $p(\boldsymbol{w}|\mathbf{y}, \boldsymbol{\Phi}, \alpha, \sigma)$, ma è invece costruita a partire dal valore atteso di tale distribuzione e dalla varianza del rumore intrinseco nei dati, assunta per ipotesi.

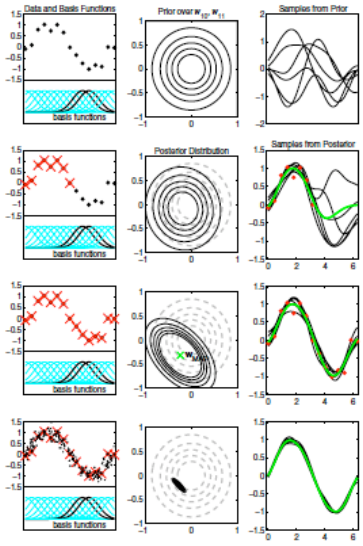

Figura 25: Apprendimento sequenziale per regressione lineare con funzioni di base gaussiane

Un approccio completamente bayesiano opera invece definendo la distribuzione di probabilità predittiva per y, ottenuta come distribuzione marginale integrando rispetto a $\boldsymbol{w}$ il prodotto della distribuzione condizionata (di y rispetto a $\boldsymbol{w}$) $p(y|\mathbf{x}, \boldsymbol{w}, \sigma^2)$, che caratterizza la probabilità di y una volta fissati i coefficienti $\boldsymbol{w}$ (e, naturalmente, i valori $\mathbf{x}$ delle features), e della distribuzione di $\boldsymbol{w}$ a posteriori della conoscenza del training set, $p(\boldsymbol{w}|\mathbf{y}, \boldsymbol{\Phi}, \alpha, \sigma^2)$. Il prodotto risultante definisce la probabilità congiunta di y e $\boldsymbol{w}$ che, marginalizzato rispetto a $\boldsymbol{w}$, fornisce la distribuzione voluta

$$p(y|\mathbf{x}, \mathbf{y}, \boldsymbol{\Phi}, \alpha, \sigma^2) = \int p(y|\mathbf{x}, \boldsymbol{w}, \sigma^2)p(\boldsymbol{w}|\mathbf{y}, \boldsymbol{\Phi}, \alpha, \sigma^2)d\boldsymbol{w}$$

Dato che $p(y|\mathbf{x}, \boldsymbol{w}, \sigma)$ è gaussiana, così come anche $p(\boldsymbol{w}|\mathbf{y}, \boldsymbol{\Phi}, \alpha, \sigma)$ (assunto che sia gaussiana la distribuzione a priori $p(\boldsymbol{w}|\alpha)$), e in particolare

$$y|\mathbf{x}, \boldsymbol{w}, \sigma \sim Normal(\boldsymbol{\phi}(\mathbf{x})^{T}\boldsymbol{w}, \sigma^2)$$
$$\boldsymbol{w}|\mathbf{y}, \boldsymbol{\Phi}, \alpha, \sigma \sim Normal(\sigma^{-2}\boldsymbol{\Sigma}\boldsymbol{\Phi}^{T}\mathbf{y}, \boldsymbol{\Sigma})$$

dove $\mathbf{\Sigma} = (\alpha\mathbf{I} + \sigma^{-2}\mathbf{\Phi}^T\mathbf{\Phi})^{-1}$, allora anche $p(y|\mathbf{x}, \mathbf{y}, \mathbf{\Phi}, \alpha, \sigma^2)$ è gaussiana e, ricordando che se $\mathbf{x}, \mathbf{y}$ sono tali che

$$\mathbf{x} \sim Normal(\boldsymbol{\mu}, \mathbf{\Sigma}_1) \qquad \mathbf{y}|\mathbf{x} \sim Normal(\mathbf{A}\mathbf{x} + \mathbf{b}, \mathbf{\Sigma}_2)$$

allora

$$\mathbf{y} \sim Normal(\overline{\boldsymbol{\mu}}, \overline{\mathbf{\Sigma}})$$

con

$$\overline{\boldsymbol{\mu}} = \mathbf{A}\boldsymbol{\mu} + \mathbf{b} \qquad \overline{\mathbf{\Sigma}} = \mathbf{\Sigma}_2 + \mathbf{A}\mathbf{\Sigma}_1\mathbf{A}^T$$

osserviamo che, nel nostro caso, $\mathbf{x} = \boldsymbol{w}$, $\mathbf{A} = \boldsymbol{\phi}(\mathbf{x})^T$, $\mathbf{b} = 0$, $\mathbf{\Sigma}_2 = \sigma^2$, $\boldsymbol{\mu} = \sigma^{-2}\mathbf{\Sigma}_1\mathbf{\Phi}^T\mathbf{y}$, $\mathbf{\Sigma}_1 = (\alpha\mathbf{I} + \sigma^{-2}\mathbf{\Phi}^T\mathbf{\Phi})^{-1}$, otteniamo che $p(y|\mathbf{x}, \mathbf{y}, \mathbf{\Phi}, \alpha, \sigma^2)$ ha varianza

$$\overline{\sigma}^2 = \sigma^2 + \boldsymbol{\phi}(\mathbf{x})^T\mathbf{\Sigma}_1\boldsymbol{\phi}(\mathbf{x})$$

e media

$$\overline{\mu} = \boldsymbol{\phi}(\mathbf{x})^T\sigma^{-2}\mathbf{\Sigma}_1\mathbf{\Phi}^T\mathbf{y} = \sigma^{-2}\boldsymbol{\phi}(\mathbf{x})^T\mathbf{\Sigma}_1\mathbf{\Phi}^T\mathbf{y}$$

Dove nella caratterizzazione della varianza il termine σ^2 è relativo all'incertezza intrinseca nei dati osservati, mentre il termine $\boldsymbol{\phi}(\mathbf{x})^T\mathbf{\Sigma}_1\boldsymbol{\phi}(\mathbf{x})$ riflette l'incertezza rispetto ai valori derivati per i coefficienti $\boldsymbol{w}$.

È possibile dimostrare che, al crescere del numero di elementi nel training set, il secondo termine della varianza diminuisce, e quindi la distribuzione tende a concentrarsi intorno al valore previsto per y. Al limite, al crescere indefinito della dimensione del training set, il solo primo termine rimane significativo, mostrando che la sola incertezza rimanente è quella relativa ai dati osservati.

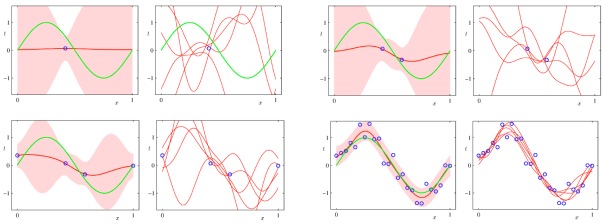

Figura 26: Esempio di distribuzione predittiva

In figura 26, sono illustrati esempi di distribuzioni predittive relative alla funzione $y = \sin 2\pi x$, utilizzando un modello con 9 funzioni base gaussiane, considerando 1,2,4,25 elementi nel training set. Nelle immagini a sinistra, sono mostrati, oltre agli elementi nel training set (generati casualmente e aggiungendo rumore gaussiano al valore corretto), l'andamento della media della distribuzione predittiva (curva in colore rosso) e della varianza della distribuzione stessa (la regione a sfondo rosa si estende fino a una deviazione standard dalla media). Si noti come l'incertezza della predizione (così come la media) dipenda da x e sia minore nelle vicinanze degli elementi del training set, e come l'incertezza diminuisca al crescere del training set.

Le immagini a destra mostrano, per gli stessi elementi nel training set, gli andamenti di 5 possibili curve approssimanti $y = \sin 2\pi x$, generate mediante campionamento sulla distribuzione a posteriori $p(\boldsymbol{w}|\mathbf{y}, \mathbf{\Phi}, \alpha, \sigma^2)$. Si può vedere che, coerentemente a quanto mostrato nelle immagini a sinistra, al crescere del training set l'incertezza diminuisce, e le varie curve campionate tendono ad approssimare sempre meglio la curva iniziale.

Osservando ora che il valore medio della distribuzione predittiva, utilizzato come previsione dell'output (dati i valori delle features) può essere riscritto come

$$y(\mathbf{x}) = \sigma^{-2}\boldsymbol{\phi}(\mathbf{x})^T\boldsymbol{\Sigma}_1\boldsymbol{\Phi}^T\mathbf{y} = \sum_{i=1}^{n}\sigma^{-2}\boldsymbol{\phi}(\mathbf{x})^T\boldsymbol{\Sigma}_1\boldsymbol{\phi}(\mathbf{x}_i)y_i$$

e quindi, il valore derivato può essere visto come combinazione lineare dei valori $\mathbf{y}$ degli elementi del training set

$$y(\mathbf{x}) = \sum_{i=1}^{n}\kappa(\mathbf{x},\mathbf{x}_i)y_i$$

dove la funzione di $\mathbf{x}$ e $\mathbf{x}_i$

$$\kappa(\mathbf{x},\mathbf{x}_i) = \sigma^{-2}\boldsymbol{\phi}(\mathbf{x})^T\boldsymbol{\Sigma}_1\boldsymbol{\phi}(\mathbf{x}_i)$$

è denominata *kernel equivalente*.

In figura 27 è mostrato, a destra, l'andamento sul piano (x, x_i) di un esempio di funzione di kernel equivalente generata da un training set di 200 elementi. A sinistra, l'andamento della curva per tre diversi valori di x. Come si può vedere, la funzione di kernel tende a dare, per la determinazione del valore $y(x)$, maggiore rilevanza ai valori y_i relativi a elementi x_i vicini a x, effettuando una *localizzazione* delle funzioni base intorno ai punti del training set.

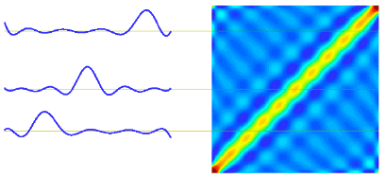

Figura 27: Esempio di funzione di kernel equivalente per funzione di base gaussiana e una sola feature, con esempio per tre valori di x

Ciò è anche evidenziato considerando la covarianza tra i valori $y(\mathbf{x})$ e $y(\mathbf{x}')$ derivati per due elementi $\mathbf{x}, \mathbf{x}'$:

$$\begin{aligned}
Cov[\boldsymbol{\phi}(\mathbf{x})^T\mathbf{w}, \mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}')] &= E[\boldsymbol{\phi}(\mathbf{x})^T\mathbf{w}\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}')] - E[\boldsymbol{\phi}(\mathbf{x})^T\mathbf{w}]E[\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}')] \\
&= \boldsymbol{\phi}(\mathbf{x})^T E[\mathbf{w}\mathbf{w}^T]\boldsymbol{\phi}(\mathbf{x}') - \boldsymbol{\phi}(\mathbf{x})^T E[\mathbf{w}]E[\mathbf{w}^T]\boldsymbol{\phi}(\mathbf{x}') \\
&= \boldsymbol{\phi}(\mathbf{x})^T \left(E[\mathbf{w}\mathbf{w}^T] - E[\mathbf{w}]E[\mathbf{w}^T]\right)\boldsymbol{\phi}(\mathbf{x}') \\
&= \boldsymbol{\phi}(\mathbf{x})^T Cov[\mathbf{w}\mathbf{w}^T]\boldsymbol{\phi}(\mathbf{x}') \\
&= \boldsymbol{\phi}(\mathbf{x})^T\boldsymbol{\Sigma}_1\boldsymbol{\phi}(\mathbf{x}') = \sigma^2\kappa(\mathbf{x},\mathbf{x}')
\end{aligned}$$

Dal che possiamo osservare come la covarianza (e quindi la relazione tra i valori di output calcolati per due elementi) sia proporzionale al valore della funzione di kernel equivalente, il che ci dice che elementi vicini avranno valori molto correlati, mentre per elementi lontani i valori calcolati saranno sostanzialmente indipendenti.

In figura 28 viene mostrato come anche i kernel equivalenti per funzioni di base diverse (polinomiali e sigmoidali) abbiano la stessa proprietà di localizzazione intorno a x (in questo caso considerando $x = 0$).

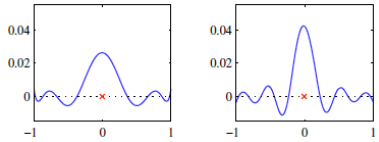

Figura 28: Esempio di funzioni di kernel equivalente per funzioni di base polinomiale (a sinistra) e sigmoidale (a destra)

Parte II

Reti neurali

Parte III

Apprendimento non supervisionato

Parte IV

Apprendimento per rinforzo

Parte V

Machine learning e big data

Parte VI

Appendici