(F94) Metodi statistici per l'apprendimento

Introduzione

Docente: *Nicolò Cesa-Bianchi* versione 7 aprile 2018

I sistemi di apprendimento automatico possono essere utilizzati per risolvere diversi tipi di problemi di inferenza. Per esempio: categorizzare o partizionare dei dati, predire una serie temporale, pianificare una sequenza di azioni. In questo corso ci focalizzeremo su sistemi di apprendimento automatico il cui scopo è imparare delle funzioni che associano un'etichetta y ad un dato $\boldsymbol{x}$. Una volta apprese, queste funzioni possono essere utilizzare per categorizzare un documento o un'immagine. Oppure predire quale annuncio pubblicitario è più probabile venga cliccato dal visitatore di un sito. O anche predire il reddito di un individuo sulla base di indicatori del suo stile di vita. O ancora diagnosticare una malattia sulla base della cartella clinica del paziente.

Si noti che le etichette y implicitamente definite in questi esempi sono di due tipi diversi: etichette simboliche, come le categorie di un documento o le malattie e etichette numeriche, come il reddito. Nel primo caso parliamo di un problema di categorizzazione (o classificazione) con insieme di etichette $\mathcal{Y}$ (p.es., $\mathcal{Y} = \{\text{sport}, \text{politica}, \text{spettacolo}\}$). Nel secondo caso parliamo invece di un problema di regressione, dove l'insieme di etichette è contenuto nei reali $\mathbb{R}$. In un problema di classificazione, generalmente gli errori non sono graduati: se prediciamo la categoria sbagliata commettiamo un errore, qualunque sia la classe predetta. In regressione, invece, dove le etichette hanno un valore numerico, l'errore è graduabile a seconda della distanza fra il valore della classe predetta e quello della classe corretta.

Per valutare la bontà di una predizione in un problema di classificazione o regressione si utilizza una **funzione di perdita** non negativa ℓ che misura la discrepanza fra etichetta predetta ed etichetta vera. Se per il dato $\boldsymbol{x}$ l'etichetta corretta è y, la predizione $\widehat{y}$ viene valutata con $\ell(y, \widehat{y}) \geq 0$. In un problema di classificazione la funzione di perdita più tipicamente utilizzata è quella zero-uno:

$$\ell(y, \widehat{y}) = \begin{cases} 0 & \text{se } y = \widehat{y}, \\ 1 & \text{altrimenti.} \end{cases}$$

In alcuni casi, come ad esempio il riconoscimento di mail spam dove $\mathcal{Y} = \{\text{spam}, \text{nonspam}\}$, possiamo penalizzare un falso positivo (ovvero una mail non spam erroneamente classificata come spam) rispetto ad un falso negativo (ovvero una mail spam non classificata come tale). Ad esempio,

$$\ell(y, \widehat{y}) = \begin{cases} 2 & \text{se } y = \text{nonspam e } \widehat{y} = \text{spam}, \\ 1 & \text{se } y = \text{spam e } \widehat{y} = \text{nonspam}, \\ 0 & \text{altrimenti.} \end{cases}$$

In un problema di regressione, invece, funzioni di perdita tipicamente utilizzate sono la perdita assoluta $\ell(y, \widehat{y}) = |y - \widehat{y}|$ e la perdita quadratica $\ell(y, \widehat{y}) = (y - \widehat{y})^2$. Si noti che queste funzioni hanno un significato solo per etichette numeriche e i loro valori crescono all'aumentare della distanza fra y e $\widehat{y}$.

In certi casi, può essere comodo predire in un insieme $\mathcal{Z}$ diverso da $\mathcal{Y}$. Per esempio, si consideri il problema di assegnare una probabilità $\widehat{y} \in (0, 1)$ all'evento $y =$ "domani piove" (e quindi implicitamente assegnare probabilità $1 - \widehat{y}$ all'evento complementare $y =$ "domani non piove"). In questo

caso, $\mathcal{Y} = \{$"domani piove", "domani non piove"$\}$ e $\mathcal{Z} = (0, 1)$. Identificando "domani piove" con 1 e "domani non piove" con 0, possiamo usare una funzione di perdita per regressione come la perdita di tipo assoluto $\ell(y, \widehat{y}) = |y - \widehat{y}| \in (0, 1)$. Per ampliare il codominio della funzione di perdita, in modo da punire maggiormente predizioni che si scostano troppo dalla realtà, possiamo invece usare la perdita logaritmica,

$$\ell(y, \widehat{y}) = \begin{cases} \ln \frac{1}{\widehat{y}} & \text{se } y = \text{"domani piove"}, \\ \ln \frac{1}{1-\widehat{y}} & \text{se } y = \text{"domani non piove"}. \end{cases}$$

Si noti che, a differenza della perdita assoluta,

$$\lim_{\widehat{y} \to 0^+} \ell(+1, \widehat{y}) = \lim_{\widehat{y} \to 1^-} \ell(-1, \widehat{y}) = \infty \ .$$

Questo in pratica scoraggia fortemente il predittore dal generare predizioni $\widehat{y}$ "troppo certe", ovvero troppo vicine a zero o uno, in quanto essere potrebbero dar luogo a valori arbitrariamente alti della funzione di perdita.

Il dato $\boldsymbol{x}$ è tipicamente un record di una base di dati. In molti casi è possibile codificare il record in un vettore di numeri, un formato che si presta bene a essere analizzato in termini geometrici. Per esempio, tutte le volte che il dato è composto da un insieme di quantità omogenee, come i pixel di un'immagine, è naturale rappresentarlo come un vettore in $\mathbb{R}^d$ (dove d sarebbe il numero di pixel nel caso dell'immagine). In altri casi, questa codifica è un po' più laboriosa. Per esempio, un documento può essere rappresentato come un vettore le cui coordinate sono le parole di un dizionario e i cui valori sono la frequenza con la quale la parola corrispondente alla coordinata appare nel documento. In altri casi ancora la codifica vettoriale può essere forzata. Per esempio, in una cartella clinica tipicamente compaiono attributi numerici non omogenei, come CAP ed età. Oppure attributi simbolici, come il sesso, non rappresentabili su un asse cartesiano. In questo corso tratteremo prevalentemente la situazione in cui il dato può essere naturalmente rappresentato come un vettore di numeri $\boldsymbol{x} \in \mathbb{R}^d$. Quando scriviamo $\boldsymbol{x} \in \mathcal{X}$ significa che non insistiamo sul fatto che $\boldsymbol{x}$ sia un vettore di numeri, ma assumiamo più genericamente che sia un record di una base di dati.

Un classificatore per un problema di classificazione è una funzione $f : \mathcal{X} \to \mathcal{Y}$ (oppure $f : \mathcal{X} \to \mathcal{Z}$ se le predizioni appartengono ad un insieme $\mathcal{Z}$ diverso da $\mathcal{Y}$). Se $\mathcal{Y}$ contiene due sole etichette, per esempio $\mathcal{Y} = \{\text{spam}, \text{nonspam}\}$, allora parliamo di un problema di classificazione binaria e assumiamo convenzionalmente $\mathcal{Y} = \{-1, +1\}$ come abbiamo fatto nell'esempio della pioggia. Analogamente, un regressore per un problema di regressione è una funzione $f : \mathcal{X} \to \mathbb{R}$.

Anche se sono state studiate varie modalità di apprendimento, in questo corso ci concentriamo su una in particolare: la modalità di apprendimento per esempi (o apprendimento supervisionato). Un **esempio** è una coppia $(\boldsymbol{x}, y)$ composta da un dato $\boldsymbol{x}$ e dalla sua etichetta y. L'etichetta y è quella che riteniamo corretta per quel dato. Per esempio, il reddito effettivamente percepito da un individuo, oppure la categoria semantica che un lettore associerebbe ad un documento. Un **training set** è un insieme (o più correttamente multinsieme) $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_m, y_m)$ di esempi. Un algoritmo di apprendimento tramite esempi è un algoritmo che riceve in input un training set e fornisce in output un classificatore oppure un regressore.

Per stimare la capacità predittiva di un classificatore o regressore, che è la cosa alla quale siamo in ultima analisi interessati, si utilizza tipicamente un insieme di esempi chiamato test set. Un test

set è un insieme $(\boldsymbol{x}'_1, y'_1), \ldots, (\boldsymbol{x}'_n, y'_n)$ di esempi a cui l'algoritmo di apprendimento non ha accesso. Training set e test set sono tipicamente costruiti assieme, tramite un singolo processo di raccolta ed annotazione dei dati. La suddivisione dei dati etichettati in training e test viene poi fatta a posteriori, in genere mediante una selezione casuale.

Dato un classificatore o regressore f, stimiamo la capacità predittiva di f attraverso il **test error**

$$\frac{1}{n} \sum_{t=1}^{n} \ell\big(y'_t, f(\boldsymbol{x}'_t)\big) \ .$$

Il test error serve a stimare il comportamento del predittore "sul campo", ovvero su dati non precedentemente osservati. Il nostro scopo è quindi formulare una teoria che ci permetta di sviluppare algoritmi di apprendimento in grado di generare predittori con basso test error.

Dato che l'unica informazione che l'algoritmo riceve in ingresso è il training set $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_m, y_m)$, un approccio ovvio allo sviluppo di algoritmi di apprendimento si basa sull'assunzione che il training error

$$\widehat{\mathrm{er}}(f) = \frac{1}{m} \sum_{t=1}^{m} \ell\big(y_t, f(\boldsymbol{x}_t)\big)$$

di un classificatore o regressore f sia correlato al suo test error.

Sia $\mathcal{F}$ un insieme dato di classificatori o regressori. Il metodo di **minimizzazione del rischio empirico** (ERM, empirical risk minimization) indica l'algoritmo di apprendimento che sceglie la funzione in $\mathcal{F}$ che minimizza il training error,

$$\widehat{f} = \operatorname*{argmin}_{f \in \mathcal{F}} \widehat{\mathrm{er}}(f) \ .$$

Putroppo non è detto che questa strategia funzioni sempre. Per esempio, se il training set è molto piccolo rispetto a $\mathcal{F}$, è possibile trovare tante funzioni in $\mathcal{F}$ con basso training error ma con test error molto diversi. In questo caso, un algoritmo di apprendimento non riuscirà a scegliere un predittore con test error basso esaminando il suo comportamento sul training set. Per evitare che ciò avvenga, il training set dovrebbe essere abbastanza grande per riuscire a distinguere i predittori buoni da quelli cattivi.

Il fenomeno per il quale un algoritmo di apprendimento tende a generare predittori con basso training error e alto test error prende il nome di **overfitting**. Il controllo dell'overfitting è uno dei temi chiave nell'apprendimento automatico. Il rischio di overfitting si verifica soprattutto in caso di dati affetti da "rumore".

In generale, diciamo che un dataset è affetto da rumore quando un'istanza $\boldsymbol{x}$ può comparire (nel training set o nel test set) a volte con un'etichetta e a volte con un'altra. La presenza del rumore può avere due cause (anche concomitanti):

1. L'etichetta è assegnata da una persona che esamina il dato ed esprime un giudizio soggettivo. Pensiamo al caso di documenti che trattano di più argomenti e di conseguenza posso essere categorizzati in modo diverso da persone diverse.

2. L'istanza $\boldsymbol{x}$ non contiene abbastanza informazioni. Per esempio, supponiamo che $\boldsymbol{x}$ rappresenti delle misure atmosferiche (temperatura, pressione, umidità) e $y \in \{-1, +1\}$ indichi se il giorno successivo piove o meno. È molto probabile che queste misure non siano sufficienti a determinare l'etichetta, quindi a parità di valori $\boldsymbol{x}$ rilevati potrebbe seguire sia un giorno di pioggia che un giorno di sole.

Quando c'è rumore possiamo parlare di etichette "tipiche" e "anomale". Nell'esempio dei documenti, l'etichetta di un documento è tipica se corrisponde alla categoria che la maggior parte delle persone assegnerebbe a quel documento ed è anomala se vale il viceversa. Nel caso delle previsioni metereologiche, un'etichetta tipica è la condizione atmosferica (pioggia o sole) che storicamente si è verificata con maggior frequenza il giorno successivo quello in cui un certo insieme di valori è stato osservato. Un buon algoritmo di apprendimento dovrebbe scegliere un classificatore (o regressore) che minimizza il training error sui punti con etichette tipiche, ignorando quelli con etichette anomale. Dato che a priori non è noto quali siano i punti anomali, si può procedere vincolando l'algoritmo di apprendimento a scegliere classificatori o regressori di complessità sufficientemente bassa, sperando che ciò lo costringa a ignorare esattamente i punti anomali del training set.

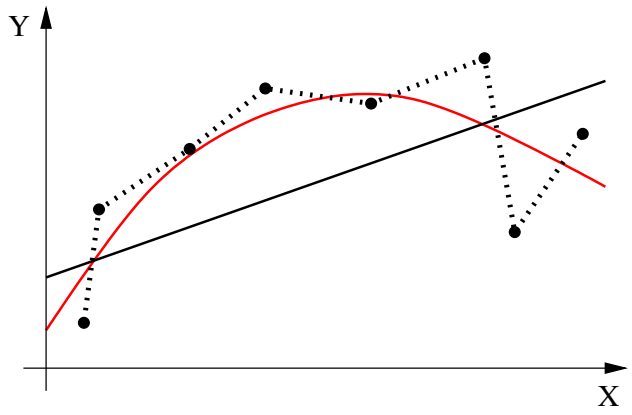

Figura 1: Controllo dell'overfitting in un problema di regressione unidimensionale. Il training set è formato dagli otto punti $(x_t, y_t) \in \mathbb{R}^2$ indicati in nero in figura. Le curve rappresentano predittori della forma $f : \mathbb{R} \to \mathbb{R}$ per il problema di regressione consistente nel predire il valore della coordinata y associata a punti x nel test set (non mostrati in figura). La curva in colore rappresenta il predittore che minimizza il test error. Le altre curve rappresentano predittori di complessità inferiore (linea retta, underfitting) o superiore (linee tratteggiate, overfitting) alla complessità del predittore ottimo.

Nel caso di problemi di regressione, un semplice esempio di overfitting è fornito dalla Figura 1.