(F94) Metodi statistici per l'apprendimento

Consistenza e algoritmi nonparametrici

Docente: *Nicolò Cesa-Bianchi* versione 22 marzo 2017

Fissiamo un modello statistico (D, η) per classificazione binaria. Per ogni algoritmo di apprendimento A, denotiamo con $A(S_m)$ il classificatore prodotto da A quando il training set S_m contenente m esempi è fornito in input. Denotiamo con $\mathrm{er}\big(A(S_m)\big)$ il rischio di questo classificatore rispetto al modello (D, η). Quindi, se $A(S_m) = h$ allora $\mathrm{er}\big(A(S_m)\big) = \mathbb{P}\big(h(\boldsymbol{X}) \neq Y\big)$. Infine, indichiamo con $\mathrm{er}(f^*)$ il rischio del classificatore Bayesiano ottimo f^*, sempre rispetto a (D, η).

L'algoritmo A è detto essere consistente rispetto a (D, q) quando

$$\lim_{m\to\infty} \mathbb{E}\Big[\mathrm{er}\big(A(S_m)\big)\Big] = \mathrm{er}(f^*)$$

dove il valore atteso è rispetto all'estrazione del training set S_m da (D, η). Quindi, la consistenza è una proprietà asintotica che certifica la capacità dell'algoritmo di raggiungere in media le prestazioni del Bayesiano ottimo al crescere del training set. La consistenza forte è invece definita come

$$\lim_{m\to\infty} \mathrm{er}\big(A(S_m)\big) = \mathrm{er}(f^*) \qquad \text{con probabilità 1}$$

rispetto all'estrazione del training set S_m da (D, η). Quindi chiediamo che A raggiunga le prestazioni di f^* non solo in media, ma con probabilità arbitrariamente alta al crescere del training set.

Nearest Neighbor è consistente per ogni (D, η) quando k è scelto in modo opportuno. Se indichiamo con $k-\mathrm{NN}$ l'algoritmo k-Nearest Neighbor, allora per ogni (D, η) vale

$$\lim_{m\to\infty} \mathbb{E}\Big[\mathrm{er}\big(k-\mathrm{NN}(S_m)\big)\Big] \leq \mathrm{er}(f^*) + 2\sqrt{\frac{\mathrm{er}(f^*)}{k}}\ . \tag{1}$$

Perciò per ogni k fissato $k-\mathrm{NN}$ non è consistente. Si noti che per $k = 1$, otteniamo

$$\lim_{m\to\infty} \mathbb{E}\Big[\mathrm{er}\big(k-\mathrm{NN}(S_m)\big)\Big] \leq \mathrm{er}(f^*) + 2\sqrt{\mathrm{er}(f^*)}\ .$$

Questo maggiorante è peggiore di $2\mathrm{er}(f^*)$, che otteniamo come limite per $m \to \infty$ nel maggiorante di 1-NN

$$\mathbb{E}\big[\mathrm{er}(\widehat{h}_S)\big] \leq 2\,\mathrm{er}(f^*) + \mathcal{O}\big(m^{-1/(d+1)}\big)$$

dimostrato in precedenza sotto l'assunzione

$$\big|\eta(\boldsymbol{x}) - \eta(\boldsymbol{x}')\big| \leq c\,\big\|\boldsymbol{x} - \boldsymbol{x}'\big\| \qquad \text{per ogni } \boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}\ . \tag{2}$$

Se in (1) facciamo crescere k non troppo velocemente, cioè se $k = k_m$ tale che $k_m \to \infty$ e $k_m = o(m)$, allora è possibile dimostrare che $k-\mathrm{NN}$ diventa fortemente consistente,

$$\lim_{m\to\infty} \mathrm{er}\big(k_m-\mathrm{NN}(S_m)\big) = \mathrm{er}(f^*) \qquad \text{con probabilità 1.}$$

Sotto determinate assunzioni su (D, η), è possibile dimostrare che un particolare algoritmo di costruzione di predittori ad albero è anch'esso fortemente consistente (dettagli omessi).

Il motivo per cui un algoritmo consistente può risultare in pratica peggiore di uno non consistente è legato alla velocità di convergenza. Infatti, la consistenza per ogni modello (D, η) può essere pagata con una velocità di convergenza *arbitrariamente lenta* al Bayes error, come stabilito dal seguente risultato.

Teorema 1 (No Free Lunch) *Sia $a_1, a_2, \ldots$ una sequenza convergente a zero di numeri positivi tali che $\frac{1}{16} \geq a_1 \geq a_2 \geq \cdots$. Per ogni algoritmo di apprendimento A esiste un modello statistico (D, η) tale che $\mathrm{er}(f^*) = 0$ e simultaneamente $\mathbb{E}\big[\mathrm{er}\big(A(S_m)\big)\big] \geq a_m$ per ogni $m \geq 1$.*

Si noti che se $\mathrm{er}(f^*) = 0$ allora dev'essere $\eta(\boldsymbol{x}) \in \{0, 1\}$ per ogni $\boldsymbol{x}$. Ovvero η è discontinua e la condizione (2) non può essere soddisfatta.

Viceversa, per ogni classe $\mathcal{H}$ di predittori sappiamo che l'algoritmo A che minimizza il rischio empirico soddisfa

$$\mathrm{er}\big(A(S_m)\big) \leq \inf_{h \in \mathcal{H}} \mathrm{er}(h) + \sqrt{\frac{2}{m} \ln \frac{2|\mathcal{H}|}{\delta}}$$

con probabilità almeno $1 - \delta$ rispetto all'estrazione di S_m e per qualunque modello (D, η).

Quindi, la questione può essere posta in questi termini: se abbiamo informazioni a priori su (D, η) allora possiamo cercare A consistente che converga a $\mathrm{er}(f^*)$ rapidamente. Se invece (D, η) è completamente ignota, allora conviene focalizzarsi su un algoritmo A che lavori in modo efficiente su una classe $\mathcal{H}$ di modelli che sia abbastanza ampia per approssimare decentemente $\mathrm{er}(f^*)$ (controllo dell'errore di bias) ma non troppo ampia da compromettere la convergenza a $\inf_{h \in \mathcal{H}} \mathrm{er}(h)$ (controllo dell'errore di varianza).

Gli algoritmi consistenti, come k-NN e gli algoritmi per i predittori ad albero, vengono anche detti algoritmi **nonparametrici**. Questa terminologia si riferisce al fatto che i classificatori prodotti non sono rappresentabili con un numero predeterminato di parametri (come ad esempio succede per i classificatori lineari). Infatti, la struttura di un classificatore nonparametrico non è fissata, ma viene determinata dai dati di training (si pensi alla costruzione di un predittore ad albero guidata dal training set).