Riflessioni sulla caratterizzazione metrologica dei sistemi di Machine Learning per la classificazione – Seconda parte

Luca Mari, Dario Petri

Tutto_Misure, 4, 2025

[3.11.25]

Nel numero precedente di Tutto_Misure abbiamo proposto un inquadramento generale e alcune prime riflessioni a proposito di come la scienza della misurazione potrebbe contribuire allo sviluppo dei sistemi di Machine Learning impiegati per la classificazione: la crescente diffusione di tali sistemi nella nostra società rende sempre più importante valutare in modo appropriato la qualità dell'informazione che essi producono, e a questo scopo la cultura metrologica può avere un ruolo significativo. A partire dalla distinzione tra valutazione della qualità del comportamento dei sistemi e valutazione della qualità dei risultati di tale comportamento, ci siamo chiesti come il concetto stesso di incertezza di misura potrebbe essere adattato al contesto della classificazione, e quindi all'obiettivo della qualificazione dei risultati che i sistemi di Machine Learning per la classificazione producono. Proseguiamo qui la riflessione.

Abbiamo fondato la nostra esplorazione sull'*analogia strutturale tra misurazione e classificazione*, entrambi processi finalizzati a produrre informazione su proprietà di oggetti mediante valori di proprietà:

– quando grazie a una misurazione arriviamo a concludere, per esempio, che la lunghezza di un certo oggetto è 1,2345 m intendiamo produrre l'informazione che quell'oggetto ha una lunghezza che è quella di 1,2345 volte il metro, scelta come lunghezza di riferimento;

– quando grazie a una classificazione arriviamo a concludere, per esempio, che la forma di un certo testo è "3" nell'insieme {"0", "1", "2", …, "9"} intendiamo produrre l'informazione che quel testo ha una forma che è quella del "3" nel sistema di numerazione arabo, scelto come insieme di cifre di riferimento.

I sistemi di misura e i sistemi di classificazione, tradizionali o di Machine Learning, incorporano questa analogia e la operazionalizzano, implementando una funzione di valutazione f: $X \rightarrow Y$, da insiemi di entità da valutare (X) a insiemi di valori (Y). La trasformazione $f(x) = y$ si può dunque leggere:

– nel caso di una misurazione, come l'informazione che il misurando x ha valore $y = f(x)$;

– nel caso di una classificazione, come l'informazione che la proprietà da classificare x è classificata nella classe identificata dal valore $y = f(x)$ (si noti che il valore y porta l'informazione sulla classe, e non la classe in quanto tale: pur di sapere che la cifra "3" si legge in italiano "tre", l'informazione che un certo testo è classificato da y = "3" oppure da y = "tre" è la stessa, perché in questo caso la classe identificata da "3" e da "tre" è la stessa),

con ciò riconoscendo che misurazioni e classificazioni sono casi di *valutazioni*, nel senso tecnico (e non valoriale / etico) di processi finalizzati ad attribuire valori alle entità considerate.

In accordo a questa semplice presentazione, la qualità del processo di valutazione è una caratteristica della funzione di valutazione f: migliore – in un senso da definire – è la funzione f, e migliore sarà il processo di valutazione stesso.

Come si costruisce dunque una funzione di valutazione appropriata? Anche in questo sistemi di misura e sistemi di Machine Learning impiegati per la classificazione manifestano interessanti analogie. Per entrambi, la funzione f è tipicamente il risultato di un processo guidato da dati (*data driven* in inglese), che si assumono disponibili e di qualità sufficiente, nella forma di un insieme di coppie "di riferimento" (x_i^*, y_i^*): il valore y_i^* di una grandezza x_i^* di un campione di misura o l'identificatore y_i^* della classe di una proprietà di riferimento classificata x_i^* (nel contesto del Machine Learning gli identificatori y_i^* sono spesso chiamati "etichette", *label* in inglese).

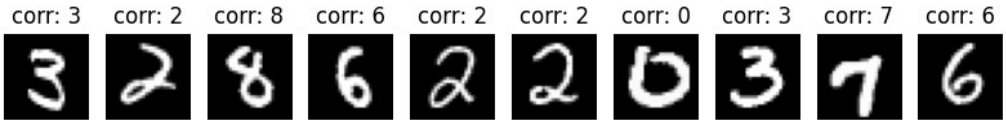
Una funzione di valutazione caratterizza il comportamento del sistema che la implementa:

– nel caso di un sistema di misura, si costruisce la funzione di valutazione tarando il sistema (e lo stesso vale per un sistema di classificazione tradizionale, pur di ammettere di poter usare la terminologia della metrologia anche nel caso della valutazione di proprietà non quantitative);

– nel caso di un sistema di Machine Learning per la classificazione, si costruisce la funzione di valutazione addestrando il sistema (il processo chiamato *training* in inglese; per enfatizzare la disponibilità delle etichette y_i^*, più specificamente, e dal punto di vista del sistema, lo si chiama "apprendimento supervisionato", *supervised learning* in inglese).

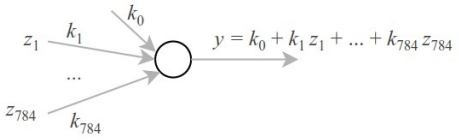
La strategia più ovvia è di costruzione estensionale: si definisce f per punti, dunque $y_i^* = f(x_i^*)$, così che la funzione non è altro che uno strumento di memorizzazione dei dati di riferimento, e in quanto tale può essere presentata in una tabella (una *lookup table* in inglese) a due colonne, per valori di X e Y rispettivamente, e tante righe quante coppie sono disponibili. Questa semplicità ha però la conseguenza che f rimane indefinita per ogni proprietà x che non coincide con una proprietà di riferimento x_i^*, e perciò in questo caso l'addestramento di un sistema di Machine Learning è appunto solo una memorizzazione: il sistema impara a ripetere, ma non a riconoscere le caratteristiche invarianti proprie di ogni classe (che ogni testo da classificare come "3" contiene due archi sovrapposti aperti a sinistra, ecc.). Come si dice, il sistema non impara a generalizzare.
Quando gli insiemi X e Y sono dotati di sufficiente struttura algebrica, questa strategia di costruzione per punti può essere migliorata, assumendo di interpolare la funzione f per gli argomenti x che non sono presenti nell'insieme di riferimento, in accordo a ipotesi sulle proprietà della funzione (per esempio la continuità) e sulla sua forma analitica. Nel caso più ovvio, si assume che f sia lineare, e dunque una funzione parametrica, a 2 parametri se sia X sia Y sono scalari, così che due coppie di punti distinti sono sufficienti per definire interamente la funzione.
È l'opzione tipicamente adottata nella taratura dei sistemi di misura più semplici, ma che non può essere impiegata per i sistemi di classificazione per i quali, per la definizione stessa di classificazione, l'insieme Y non ha alcuna struttura algebrica, essendo appunto solo un insieme di identificatori di classi (a proposito dell'esempio che stiamo usando, non ci si lasci trarre in inganno dal fatto che sui numeri è definita una ricca struttura algebrica: è vero che 1 è minore di 2, che la distanza tra 1 e 2 è la stessa che c'è fra 4 e 5, e così via, ma il nostro esempio ha a che vedere con la forma delle cifre, per poterle riconoscere in testi scritti, e la forma dell'"1" non è né minore né maggiore della forma del "2", ma solo diversa da questa).
Siamo dunque costretti ad ammettere che, per andare oltre la banale memorizzazione dei dati nell'insieme di riferimento, la funzione di valutazione di un sistema per la classificazione deve essere costruita in accordo a una strategia di qualche altro genere.

Il Machine Learning sviluppa una strategia alternativa alla memorizzazione: mantiene l'idea di operare con funzioni di valutazione parametriche, compensando l'assenza di struttura algebrica negli insiemi X e Y su cui si opera con l'impiego di funzioni con un grande numero di parametri. Per comprendere il senso di questa strategia, lavoriamo sull'esempio già menzionato di cifre da riconoscere, a partire dal noto dataset MNIST (https://en.wikipedia.org/wiki/MNIST_database), un insieme di riferimento con complessivamente 70000 immagini di singole cifre x_i^* scritte a mano, da persone diverse con stili di scrittura diversi, e contenente per ogni immagine la cifra corretta y_i^*.
Ecco l'esempio di 10 di queste immagini, scelte casualmente, con la cifra corretta corrispondente:



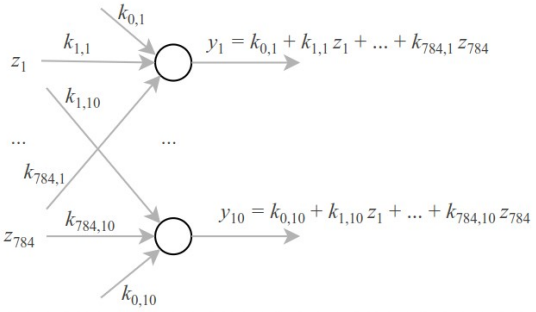
Il dataset MNIST rende disponibili queste immagini già in formato digitale, come matrici di 28 per 28 pixel in scala di grigi, con un byte per ogni pixel: la funzione di valutazione deve perciò associare una cifra, nell'insieme {"0", "1", "2", …, "9"}, a ogni matrice di 28 per 28 = 784 numeri tra 0 e 255. La costruzione di tale funzione è dunque un problema esclusivamente informazionale, perché la fase di acquisizione è avvenuta offline, e senza più nulla di empirico, come invece sarebbe stato se le immagini fossero state da riprendere mediante una telecamera: è questa una differenza importante rispetto alla taratura di uno strumento di misura, che prevede come ingresso il segnale fornito da un sensore.

Una soluzione semplice è di ricorrere anche in questo caso a una funzione lineare, dunque con 784+1 parametri k_i, $y=k_0+\sum k_i z_i$, che calcola il valore y associato a un'immagine a partire dai colori z_i dei punti dell'immagine stessa. Lo si può interpretare come un singolo "neurone" con 784 input z_1, ..., z_{784}, uno per ogni pixel, 784 parametri k_1, ..., k_{784}, uno per ogni input, e 1 "bias" k_0:



Si noti: se stessimo applicando una strategia di memorizzazione, avremmo bisogno di inserire in una tabella 784 numeri + 1 cifra per ogni immagine; qui stiamo invece cercando di trovare i valori numerici di 785 parametri che ci consentano di classificare correttamente decine di migliaia di immagini diverse. Questa funzione di valutazione opera perciò come un sistema di *compressione di dati*.

Considerata in una prospettiva metrologica, questa soluzione al nostro problema di classificazione corrisponde a una misurazione che produce un singolo valore misurato, ma che sappiamo può essere resa più informativa producendo una distribuzione di probabilità sui possibili valori del misurando. Analogamente, nel nostro esempio la funzione di valutazione può essere modificata in modo che produca non una cifra, ma una distribuzione di probabilità definita sull'insieme delle 10 cifre. Potrebbe essere una ancora semplice rete neurale, lineare, senza strati interni e *fully connected*, in cui cioè ognuno dei 784 input è connesso a ognuno dei 10 output:



Si tratta perciò di una funzione con 7850 parametri, un numero rilevante (benché di molti ordini di grandezza minore del numero di parametri delle reti alla base degli attuali chatbot), che consente di ottenere buoni risultati nella classificazione delle immagini, anche con un processo di addestramento – cioè di ottimizzazione del valore dei parametri della funzione – che richiede relativamente poche ripetizioni.
E proprio in una prospettiva metrologica la disponibilità di distribuzioni di probabilità sui valori possibili, invece che solo di singoli valori, apre interessanti opzioni.
Il più ovvio indice di qualità del comportamento di un classificatore è la sua *accuratezza*, definita in questo caso come la percentuale di classificazioni corrette sul totale di classificazioni calcolate (e si noti con ciò l'analogia con l'accuratezza di uno strumento di misura: in entrambi i casi, per valutare l'accuratezza si assume la conoscenza di un valore di riferimento – operazionalmente trattato come valore vero – della proprietà valutata, in questo caso la cifra effettivamente scritta nell'immagine).
Nonostante i limiti imposti dalla scelta di considerare una classe di funzioni lineari, la rete descritta sopra può essere addestrata fino a raggiungere un'accuratezza di almeno l'80%: è un buon risultato (con reti più complesse si ottiene un'accuratezza superiore al 99% per questo dataset, ma questo è un altro discorso), ma ciò significa comunque che statisticamente 1 immagine su 5 viene classificata in modo non corretto. Se stessimo trattando di chatbot, diremmo che, sì, la conversazione è generalmente corretta, ma spesso contiene "allucinazioni".
È però plausibile che molti di questi errori siano dovuti non solo alla relativamente limitata qualità della funzione di valutazione, ma anche alla nostra decisione di identificare comunque una cifra per ogni immagine da classificare: a prescindere da quale distribuzione la funzione abbia prodotto, per calcolare l'accuratezza della classificazione abbiamo sempre e solo preso la cifra calcolata come la più probabile, cioè

la moda della distribuzione. In altri termini, potrebbe essere che il classificatore commetta così tanti errori *anche perché gli abbiamo imposto di darci comunque una risposta*, anche quando sarebbe più cauto che la risposta fosse "non sono abbastanza certo del risultato".
Dopo aver costruito e addestrato una rete, come mostrato sopra, abbiamo messo alla prova questa ipotesi.
Prima di tutto, abbiamo ripetuto per 100 volte, in condizioni di indipendenza statistica, il calcolo dell'accuratezza della classificazione su campioni di 1000 immagini, ottenendo un'accuratezza media dell'87% (con una deviazione standard campionaria dell'1%, a indicazione di un dato affidabile), e dunque con un tasso di errore del 13%.
A questo punto abbiamo caratterizzato le condizioni di certezza di una classificazione in funzione di due iperparametri:
– w_1: la probabilità minima della moda della distribuzione;
– w_2: la differenza minima tra la probabilità della moda e la seconda probabilità più grande della distribuzione.
In questa versione modificata della funzione di classificazione, un'immagine viene classificata, scegliendo la moda della distribuzione, solo se la probabilità della moda della distribuzione è maggiore di w_1 e la differenza tra le due probabilità più grandi della distribuzione è maggiore di w_2; altrimenti, riconoscendo che ci si trova in una condizione di non sufficiente certezza, il risultato è di ammissione di indecisione.
Ecco qualche esempio dei risultati così ottenuti:

Scenari	A	B	C	D
w_1 (probabilità minima della moda)	0	0,4	0,5	0,6
w_2 (differenza minima tra le due probabilità più grandi)	0	0,1	0,2	0,3
accuratezza %	87	79	71	63
casi di indecisione %	0	17	26	36
errore %	13	4	3	1

Come si vede, il tasso di errore dipende in modo significativo dalla scelta di aggiungere una classe di indecisione e dal criterio con cui tale classe viene identificata. Confrontiamo i due scenari estremi: A, in cui non si ammette indecisione e dunque si ottiene comunque una classificazione, e D, in cui il criterio di decisione è particolarmente stringente e quindi oltre un terzo delle immagini non è classificato perché la valutazione ottenuta non è considerata abbastanza certa. A questo incremento dei casi di indecisione corrisponde però, in modo complementare, una riduzione drastica del tasso di errore, che passa dal 13% nello scenario A all'1% nello scenario D.
Troviamo dunque qui un'indicazione chiara sul valore di un trattamento esplicito dell'incertezza nel *decision making* anche a proposito dei sistemi di Machine Learning per la classificazione: *quando si ammette di non sapere perché non si è abbastanza certi, si commettono meno errori*.

È lo stesso genere di conclusione a cui è giunto un recente articolo di OpenAI a proposito delle "allucinazioni" dei chatbot (https://openai.com/it-IT/index/why-language-models-hallucinate), e da cui citiamo: "Le allucinazioni persistono in parte perché gli attuali metodi di valutazione forniscono incentivi sbagliati. Sebbene le valutazioni di per sé non causino direttamente allucinazioni, la maggior parte delle valutazioni misura le prestazioni del modello in modo tale da incoraggiare le supposizioni piuttosto che l'onestà riguardo all'incertezza. Consideralo come un test a scelta multipla. Se non conosci la risposta ma provi a indovinare, potresti essere fortunato e indovinare. Se non indichi una risposta, invece, ottieni automaticamente uno zero. Analogamente, quando i modelli vengono valutati solo in base alla precisione, ovvero alla percentuale di domande a cui rispondono correttamente, essi sono indotti a tirare a indovinare piuttosto che dire 'non lo so'."
L'*onestà riguardo all'incertezza*… una questione su cui la metrologia ha tanto da insegnare.