STATISTICAL METHODS FOR MACHINE LEARNING

giosumarin

March 2021

1 Lezione 1

- clustering: raggruppare punti in accordo alla loro similarità (raggruppare clienti per soldi spesi);
- classification: predirre label semantiche associate ai data points (classificare documenti per argomento);
- planning: vogliamo decidere una sequenza di azioni che devono essere fatte per raggiungere un goal (robot che va da quache parte con ostacoli sul percorso o guida autonoma).
- supervised learning: abbiamo laber per degli esempi e imparo a classificare d questi
- unsupervised learning: clustering (label "attaccata" ai data points)

1.1 Label set

- Y label set
- news classification: Y={sport, politica, business, ... }
- predizione stock price: $Y \in \mathbb{R}$
- classification/categorization: Y insieme finito di simboli, $\hat{y} \stackrel{?}{=} y$, con $\hat{y}$ predizione e y valore reale;
- regression: $Y \in \mathbb{R}$, $|\hat{y} - y|$.

1.2 Loss function

$$l(y, \hat{y}) = \begin{cases} 0 \ se \ y = \hat{y} \\ 1 \ altrimenti \end{cases}$$

$Y = \{spam \ (positivo), nonspam \ (negativo)\}$, *binary classification problem*

$$l(y, \hat{y}) = \begin{cases} 2 \ se \ y = nonspam \ e \ \hat{y} = spam \leftarrow \ falso \ positivo \\ 1 \ se \ y = spam \ e \ \hat{y} = nonspam \leftarrow \ falso \ positivo \\ 0 \ altrimenti \end{cases}$$

absolute loss (per regressione): $l(y, \hat{y}) = |\hat{y} - y|$

square loss (per regressione): $l(y, \hat{y}) = (\hat{y} - y)^2$

[ESEMPIO] previsioni meteo: $Y = \{pioggia, asciutto\}$

$\hat{y} =$ *probabilità assegnata a pioggia*; *prediction set:* $Z = \{0, 1\}$

$l(y, \hat{y}) = |\hat{y} - y|$

$$l(y, \hat{y}) = \begin{cases} \ln \frac{1}{\hat{y}} \ se \ y = 1 \\ \ln \frac{1}{1-\hat{y}} \ se \ y = 0 \end{cases}$$

La loss logaritmica ha le seguenti proprietà:

- $\lim_{\hat{y} \to 0^+} l(1, \hat{y}) = \infty$
- $\lim_{\hat{y} \to 1^-} l(0, \hat{y}) = \infty$

2 Lezione 2

2.1 Data Points

X dominio dati, x spesso è codificato convenientemente come vettore di numeri attravero per esempio la one-hot encoding.

$$X = \begin{cases} \mathbb{R}^d \ attributi \ numerici \\ X_1, \ldots, X_d \ attributi \ categorici \end{cases}$$

Possiamo avere anche un mix di diversi attributi.

2.2 Predictor

Un predittore è una funzione che mappa data points in label

$$f : X \to Y, f : X \to \overline{Z}, \overline{Z} \neq Y$$

Dato un ponto x abbiamo quindi

$$\hat{y} = f(x).$$

Quello che vogliore è avere una loss piccola per molti $x \in X$.

2.3 Supervised learning

Abbiamo le coppie (x, y) con x singolo data point e y la sua rispettiva label. Le label possono essere soggettive (annotazioni umane) o ogettive (misurazioni di strumenti).

2.3.1 Training Set

Insieme di esempi su cui effettuiamo l'addestramento; abbiamo quindi un training set in input a un algoritrmo di apprendimento (con la sua loss) e che in output genera un predittore.

2.3.2 Test Set

Insieme di esempi ($\neq$ training set) su cui viene valutata la capacità di generalizzazione di un predittore addestrato sul training set.

2.3.3 Completo

Abbiamo il predittore f uscente dall'algoritmo di apprendimento A usando la funzione di loss l. Abbiamo il test set $(x'_1, y'_1), \ldots, (x'_n, y'_n)$, calcoliamo il nostro test error come

$$\frac{1}{n} \sum_{t}^{n} l(y'_t, f(x'_t)).$$

Il nostro goal è quello di sviluppare una teoria per guidare nel design di A che ci genera predittori con un piccolo test error w.r.t. una loss function.

2.4 Empirical Risk Minimizer

Fisso un insieme F di predittori e una loss function f. Entra quindi il training set (S) in questo ERM (che ha F e l) e abbiamo in output

$$\hat{f} \in arg \min_{f \in F} \hat{l_S}(f).$$

L'idea è di minimizzare il training error in una classe F di predittori. Se $\min_{f \in F} \frac{1}{n} \sum_{t=1}^{n} l(y'_t, f(x'_t))$ è grande siamo in un caso di underfitting.

2.4.1 Esempio

Prendiamo F grande e vediamo cosa succede.

$$X = \{x_1, \ldots, x_5\}, Y = \{-1, 1\}, F \textit{ contiene tutti i classificatori binari}$$
$$|F| = 2^5 = 32, \exists f^* \textit{ t.c. } y_t = f^*(x_t) \textit{ con } t = \{1, \ldots, 5\}$$

Se il training set è formato dai primi 3 data point tutti e 4 i predittori hanno

	x_1	x_2	x_3	x_4	x_5
f^*	-1	1	1	$f^*(x_4)$	$f^*(x_5)$
f^1	-1	1	1	1	1
f^2	-1	1	1	-1	1
f^3	-1	1	1	1	-1
f^4	-1	1	1	-1	-1

lo stesso training error uguale a 0. In questo caso non possiamo decidere quale predittore usare. Chiamo questo caso overfitting.

Possiamo estrapolare la seguente regola da questo esempio (quando F è finito):

$$m \geq \log_2 |F|$$

3 Lezione 3