

Introduzione all'Intelligenza Artificiale – a.a. 2018/19

Seconda prova intermedia

4 Giugno 2019

PARTE I - Domande (punti 7)

La risposta corretta alla maggior parte delle domande abilita la correzione della seconda parte. Fornire risposte brevi, entro una riga di testo.

1. Candidate Elimination è completo nella strategia di ricerca o nello spazio delle ipotesi?
2. Quante ipotesi del Version Space rispondono positivamente con un *unbiased learner* su un nuovo esempio?
3. Scrivere la *loss* del Least Square utilizzata per il modello lineare.
4. Esprimere il Gain Ratio.
5. Fornire due esempi di iper-parametri visti nei modelli presentati nel corso.
6. Definire il clustering.
7. Esprimere un paio dei "sinonimi" utilizzati per descrivere l'approccio K-NN (oltre che "*instance-based*").

PARTE II – Esercizi

Esercizio 1. Costruiamo un modello con 6 dati (punti 14)

Vi sono forniti i seguenti dati (esempi) di training nella forma (input x , output desiderato y) con input in due dimensioni e target $y \in \{-1, +1\}$, quindi come coppie di esempi $[(x_1, x_2), y]$ di valori: $[(0,0), 1]$, $[(0,-1), 1]$, $[(0,2), 1]$, $[(1,0), 1]$, $[(3,0), -1]$, $[(0,-3), -1]$.

- a. Definire il tipo di compito (task) di apprendimento.
- b. Disegnare nel piano cartesiano una possibile soluzione esatta al problema con un modello lineare. Scrivere poi l'equazione del modello individuato (mostrando i valori dei parametri w).
- c. Si provi un passo dell'*algoritmo di discesa del gradiente* con i dati di esempio forniti: si calcolino i valori dei $w(t)$ dopo un passo ($t=1$) assumendo *eta* (*learning rate*) pari a $\frac{1}{2}$, i valori iniziali dei pesi w a 0 al passo $t=0$ e la *loss Least Squares* vista per il modello lineare.
- d. Determinare l'equazione del modello $h(x)$ risultante dal punto c e disegnare la soluzione nel piano cartesiano. Commette errori (e nel caso su quali esempi dell'insieme di dati fornito) rispetto al task originale?
- e. K-NN: Cosa risponde un K-NN nel punto $(-1,-1)$ per $K=1, 2$, e 3 ? Motivare la risposta.
- f. DT: Proporre per via intuitiva (senza calcolo dei Gain) un Decision Tree che risponda correttamente sui dati.
- g. SVM: Proporre una soluzione (direttamente, senza necessità di eseguire algoritmi di learning) con un modello SVM *hard-margin* senza Kernel (per via grafica) e indicare quali siano i vettori di supporto.

Esercizio 2. Dubbi di design (punti 11)

Un collega vi propone delle euristiche per applicare modelli di ML. Discutere le seguenti affermazioni riguardanti possibili scelte di design, ossia dire se considerarle vere o false e motivare la risposta (molto brevemente in una o due righe), alla luce dei principi alla base del ML.

1. In un modello lineare soggetto a una *linear basis expansion* si aumenta la flessibilità del modello.
2. Un modello lineare soggetto a una *linear basis expansion* e regolarizzato con *ridge regression/Tikhonov regularization* non può andare in *overfitting*.
3. Se il numero di esempi di training cresce, il modello può essere più flessibile.
4. Una SVM non può risolvere problemi non linearmente separabili.
5. In un DT limitare il numero dei nodi a un massimo fissato garantisce un buon apprendimento.
6. In una discesa di gradiente non fermarsi presto assicura un buon *learning*.
7. In un DT il numero di nodi non deve essere inferiore alla metà dei dati di training.
8. Per stimare la (futura) capacità predittiva di un modello è bene considerare accuratamente il risultato della model selection senza guardare al risultato in training.
9. In un modello lineare è ragionevole usare un numero di esempi proporzionale al numero di parametri.