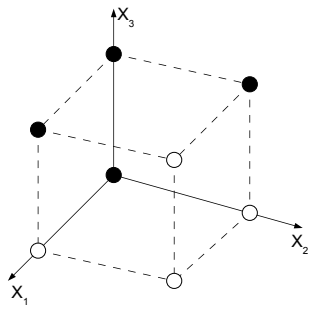
Università degli Studi di Cagliari
Corsi di Laurea Magistrale in Ing. Elettronica

Corso di Intelligenza Artificiale
A.A. 2016/2017

Esercizi sui metodi di apprendimento automatico

1. Si consideri la funzione Booleana di tre variabili rappresentata in figura: i valori delle variabili per cui la funzione assume i valori 1 e 0 sono indicati con cerchi rispettivamente neri e bianchi.



   (a) Costruire un albero di decisione in grado di rappresentare tale funzione.

   (b) Considerando gli otto punti in figura come esempi di un *training set*, calcolare l'entropia corrispondentre alla scelta della variabile X_1 per il nodo radice di un albero di decisione.

2. Determinare l'architettura e i valori dei pesi delle connessioni di una rete di percettroni, in modo da renderla consistente con il seguente *training set* composto da esempi descritti da due attributi X_1, X_2 (numeri reali), a appartenenti a due classi aventi etichette 0 e 1:

   - classe 0: $\{(0.3, 0), (0.7, 0), (0.5, 0.3)\}$
   - classe 1: $\{(0, 0.2), (0.5, 0.7), (1, 0.2)\}$

   Che cosa cambierebbe se la rete neurale dovesse essere composta da unità con funzione d'attivazione "sigmoidale", $g(a) = [1 + \exp(-a)]^{-1}$?

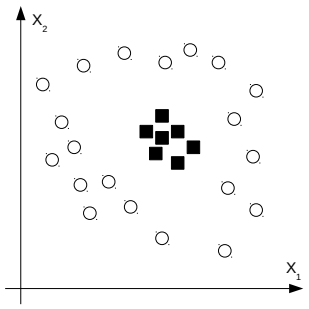
3. Si consideri una rete neurale *feed-forward multi-layer* le cui unità abbiano una funzione di attivazione "sigmoidale", $g(a) = [1 + \exp(-a)]^{-1}$. Si supponga di addestrare la rete con l'algoritmo di *backpropagation* su un *training set* T composto da $n = 100$ esempi, usando come funzione d'errore $E(T) = \frac{1}{2}\sum_{i=1}^{n}(y_i - t_i)^2$, dove y_i indica l'uscita della rete in corrispondenza dell'i-esimo esempio, e t_i il valore desiderato dell'uscita. Si assuma infine che l'algoritmo di *backpropagation* trovi valori dei pesi corrispondenti al minimo globale della funzione d'errore. Se tutti gli esempi fossero identici per i valori degli attributi, ma per ottanta di essi il valore desiderato dell'uscita della rete fosse 1 e per gli altri venti fosse 0, quale etichetta verrebbe assegnata dalla rete a ciascuno di tali esempi?

4. Si consideri un problema di classificazione supervisionata nel quale il *training set* sia composto da sette esempi appartenenti a due classi indicate con le etichette `A` e `B`, e rappresentati da due attributi (numeri reali): $(0.5, 0.5, \texttt{A})$, $(1, 0.75, \texttt{A})$, $(2.0, 0.5, \texttt{A})$, $(1, 0.25, \texttt{B})$, $(2, 0.25, \texttt{B})$, $(2.5, 0.75, \texttt{B})$, $(3, 0.5, \texttt{B})$. Si dica se sia possibile ottenere un'ipotesi consistente con questo insieme di esempi, usando:

   - un albero di decisione,
   - un percettrone,
   - una rete neurale *feed-forward multi-layer* con uno strato nascosto.

   Per ogni risposta affermativa, determinare *quante* diverse ipotesi (alberi di decisione, percettroni, o reti neurali) consistenti con l'insieme di esempi possano essere costruite, e determinare la struttura di una di tali ipotesi.

5. Si consideri il *training set* mostrato in figura, composto da esempi appartenenti a due classi (rappresentate da quadrati neri e cerchi bianchi) e descritti da due attributi X_1 e X_2 (numeri reali).



Si dica se sia possibile costruire un'ipotesi consistente con tale insieme di esempi, usando:

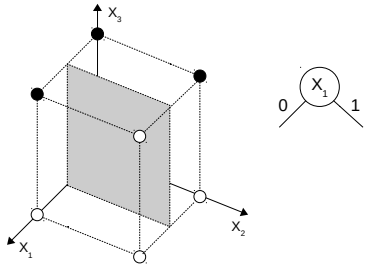
- un albero di decisione,
- un percettrone,
- una rete neurale *feed-forward multi-layer* con uno strato nascosto.

Per ogni risposta affermativa, definire una possibile ipotesi consistente; in particolare, nel caso eventuale di alberi di decisione e reti neurali, si cerchi di minimizzare la loro complessità, intesa come il numero di nodi e di foglie di un albero di decisione, e il numero di unità nascoste di una rete neurale.

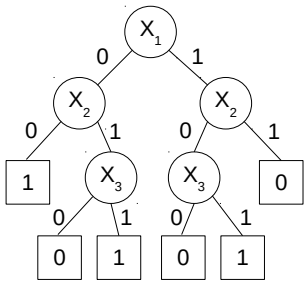
Soluzioni

1. (a) Il procedimento più semplice consiste nel costruire un percorso dal nodo radice a una foglia per ciascuno degli otto valori della funzione: scelto un attributo qualsiasi per il nodo radice, ogni percorso conterrà tre nodi (corrispodenti ai tre attributi, inclusa la radice) e una foglia. Per esempio, il percorso corrispondente a $X_1 = 0, X_2 = 0, X_3 = 0$ codificherà la regola:
`IF` $X_1 = 0$ `AND` $X_2 = 0$ `AND` $X_3 = 0$ `THEN` $F = 1$.

   Un procedimento più efficiente (in termini della dimensione dell'albero di decisione) consiste invece nel cercare di discriminare gli elementi del dominio della funzione in modo analogo agli algoritmi di apprendimento per alberi di decisione. Data la simmetria della funzione, il nodo radice può essere associato indifferentemente a ciascuno dei tre attributi. Per esempio, scegliendo X_1 per il nodo radice, i *test* $X_1 = 0$ e $X_1 = 1$ suddividono gli otto valori del dominio come indicato in figura:



   La costruzione di ciascuno dei sottoalberi in corrispondenza dei successori del nodo radice avrà come obiettivo la separazione dei tre punti corrispondenti a uno dei valori della funzione dall'unico punto corrispondente all'altro valore. Per esempio, per $X_1 = 0$ si dovranno separare i punti $(0, 0, 0)$, $(0, 0, 1)$ e $(0, 1, 1)$, per i quali $F = 1$, dal punto $(0, 1, 0)$, per il quale $F = 0$. A questo scopo è necessario considerare i valori di entrambi gli attributi rimanenti, ma su un solo ramo di ciascun sottoalbero. Un possibile albero di decisione è quindi il seguente:



   (b) Interpretando sia il valore F della funzione che le variabili X_1, X_2 e X_3 come variabili aleatorie, l'espressione dell'entropia corrispondentre alla scelta di X_1 per il nodo radice è la seguente:

$$H(F|X_1) = \mathbb{P}(X_1 = 0) \times H(F|X_1 = 0) + \mathbb{P}(X_1 = 1)H(F|X_1 = 1) \ .$$

   Il valore $H(F|X_1 = k)$, con $k \in \{0, 1\}$, è definito come:

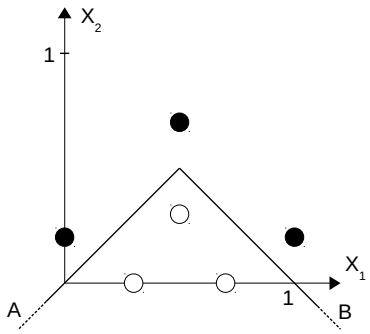
$$\begin{aligned} H(F|X_1 = k) \quad = \quad & -\mathbb{P}(F = 0|X_1 = k) \log_2 P(F = 0|X_1 = k) \\ & -\mathbb{P}(F = 1|X_1 = k) \log_2 P(F = 1|X_1 = k) \ . \end{aligned}$$

   I valori $\mathbb{P}(X_1 = k)$, $k \in \{0, 1\}$, si calcolano come la frazione del numero di elementi del dominio per i quali X_1 assume il valore k (in entrambi i casi si ottiene $4/8 = 0.5$), mentre ciascuno dei valori $\mathbb{P}(F = f|X_1 = k)$, con $f, k \in \{0, 1\}$, si calcola come la frazione di valori del dominio per i quali $F = f$ rispetto a quelli per i quali $X_1 = k$. Per esempio, $\mathbb{P}(F = 0|X_1 = 0) = \frac{1}{4} = 0.25$.

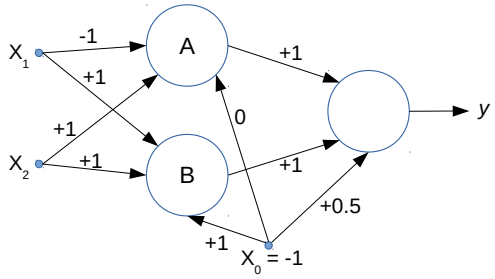
   Si ottiene quindi:

$$\begin{aligned} H(F|X_1) \quad &= \quad \mathbb{P}(X_1 = 0) \times H(F|X_1 = 0) + \mathbb{P}(X_1 = 1)H(F|X_1 = 1) \\ &= \quad 0.5 \times [-0.25 \log_2 0.25 - 0.75 \log_2 0.75] + \\ &\quad\ \ 0.5 \times [-0.75 \log_2 0.75 - 0.25 \log_2 0.25] \\ &\approx \quad 0.81 \ . \end{aligned}$$

2. Gli esempi del *training set* sono mostrati in figura (quelli della classe '0' con cerchi bianchi, quelli della classe '1' con cerchi neri), e possono essere separati dalle due semirette A e B, appartenenti alle rette descritte dalle equazioni $X_2 = X_1$ (A) e $X_2 = 1 - X_1$ (B).



La rete di percettroni in gradi di realizzare tali regioni di decisione è composta da uno strato nascosto contenente due unità. Una possibile scelta per i pesi delle connessioni consiste nel fare in modo che le due unità nascoste abbiano uscita pari a 1 nel sottospazio al di sopra delle due semirette: in questo caso l'unità di uscita dovrà realizzare la funzione logica `OR`. La funzione di attivazione dell'unità nascosta corrispondente alla semiretta A sarà quindi $-X_1 + X_2 \geq 0$, e i corrispondenti pesi delle connessioni di ingresso di tale unità saranno -1 per X_1, $+1$ per X_2, e 0 per la polarizzazione. L'unità nascosta corrispondente alla semiretta B dovrà avere la funzione d'attivazione $X_1 + X_2 - 1 \geq 0$; i pesi delle sue connessioni di ingresso saranno quindi tutti pari a 1. Per realizzare la funzione logica `OR`, i pesi delle connessioni dalle due unità nascoste all'unità di uscita saranno pari a 1, mentre il peso della connessione di polarizzazione sarà pari a 0.5:



Se la rete neurale dovesse essere composta da unità con funzione d'attivazione "sigmoidale", si potrebbero ottenere le stesse regioni di decisione con gli stessi pesi delle connessioni indicate sopra. Infatti in questo caso le uscite delle unità nascoste saranno maggiori di 0.5 nel sottospazio al di sopra delle due semirette, e minori di 0.5 nel sottospazio al di sotto di esse. L'unità di uscita avrà quindi un valore d'ingresso $a > 0.5 - w_0$ nel sottospazio al di sopra delle due semirette, e un valore $a < 0.5 - w_0$ al di sotto di esse, dove w_0 indica il peso della connessione di polarizzazione della stessa unità. Conseguentemente l'uscita della rete neurale, $y = [1 + \exp(-a)]^{-1}$, sarà maggiore di 0.5 (cioè, $a > 0$) nel sottospazio al di sopra delle semirette A e B, e minore di 0.5 ($a < 0$) nel sottospazio al di sotto di esse. Sarà quindi possibile usare la classica funzione di decisione che prevede di assegnare l'etichetta '1' alle istanze per le quali l'uscita della rete sia maggiore o uguale a 0.5, e l'etichetta '0' a tutte le altre.

3. Dato che tutti gli esempi hanno valori identici dei loro attributi, anche l'uscita della rete neurale sarà la stessa in corrispondenza di ogni esempio. Tale valore dell'uscita, indicato con y, può essere ottenuto considerando che in corrispondenza di esso si ha il minimo assoluto della funzione d'errore, data da:

$$E(T) = \frac{1}{2}\left[80 \times (y-1)^2 + 20 \times y^2\right] \ .$$

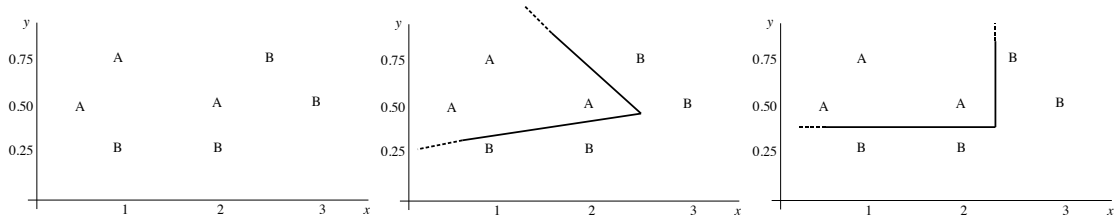
Ponendo a zero la derivata rispetto a y si ottiene:

$$\frac{\mathrm{d}E(T)}{\mathrm{d}y} = 80(y-1) + 20y = 0 \ ,$$

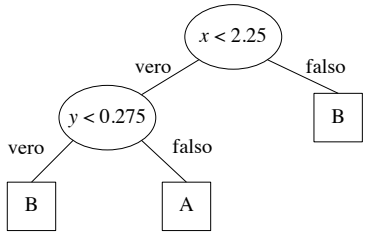
e quindi il minimo assoluto di $E(T)$ si ottiene per $y = 0.8$.

Poiché la funzione di attivazione dell'unità di uscita assume valori nell'intervallo $(0, 1)$, si userà la regola di decisione menzionata nell'esercizio precedente. Quindi la rete neurale considerata assegnerà a *tutti* gli esempi del *training set* l'etichetta '1'. In altre parole, poiché nel *training set* le due classi sono indistinguibili in base ai valori degli attributi considerati, l'ipotesi più vicina alla consistenza è quella che assegna a tutte le istanze l'etichetta della classe più numerosa nel *training set*.

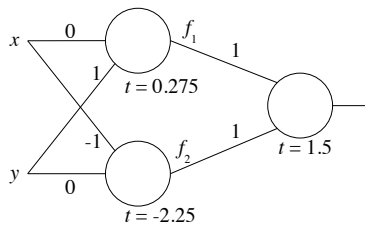
4. Nella figura in basso a sinistra si mostrano gli esempi del *training set* nello spazio dei due attributi, indicati con x e y. È facile vedere che gli esempi delle due classi non sono linearmente separabili, quindi un percettrone non può fornire un'ipotesi consistente. Dato che gli esempi delle due classi possono essere separati usando due semirette, come mostrato nella figura al centro, è possibile ottenere un'ipotesi consistente usando una rete neurale con uno strato nascosto. In particolare, dato che le due semirette possono essere perpendicolari agli assi (come nella figura a destra), è anche possibile usare un albero di decisione.



Un possibile albero di decisione, corrispondente all'ipotesi mostrata in alto a destra, è il seguente:



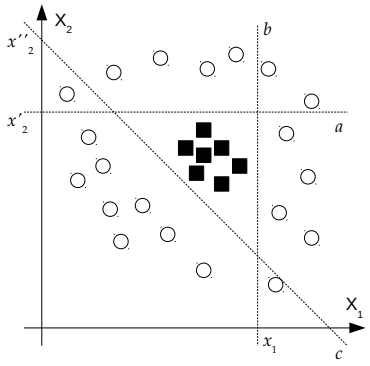
Per quanto riguarda la rete neurale, si osservi prima di tutto che sono sufficienti due unità nascoste, dato che i punti delle due classi possono essere separati usando *due* semirette. Come si è visto nell'esercizio 2, sarà possibile usare sia una rete di percettroni che una rete composta da unità con funzione di attivazione continua. Per semplicità si considerano le regioni di decisione mostrate nella figura in alto a destra. Si assume inoltre di assegnare l'uscita desiderata 1 alla classe `A` e 0 alla classe `B`. Usando una rete di percettroni, le funzioni di attivazione delle due unità nascoste potranno quindi rappresentare i semipiani definiti dalle disequazioni $y - 0.275 \geq 0$ e $-x + 2.25 \geq 0$. In questo modo l'unità di uscita dovrà realizzare la funzione logica `AND`. La rete neurale corrispondente è mostrata nella figura in basso (dove si indica con t il peso della connnessione di polarizzazione).



Il numero di ipotesi consistenti che si possono ottenere sia con un albero di decisione che con una rete neurale è chiaramente infinito. Infatti gli attributi hanno valori continui, ed esistono infinite coppie di semirette, parallele agli assi o meno, che separano gli esempi delle due classi. Si noti anche che una stessa funzione può essere rappresentata da più alberi di decisione, ottenuti cambiando l'ordine con cui gli attributi vengono considerati nei diversi percorsi dalla radice alle foglie; inoltre, gli esempi delle due classi possono essere separati definendo regioni di decisione composte da più di due semirette.

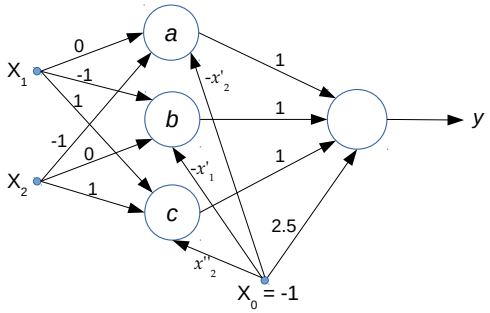
5. Gli esempi delle due classi non sono linearmente separabili, e quindi un percettrone non può rappresentare un'ipotesi consistente.

   Per ottenere un'ipotesi consistente con una rete di percettroni o una rete neurale con funzioni d'attivazione continue è necessario uno strato nascosto contenente almeno tre unità: per separare gli esempi delle due classi è infatti necessaria la combinazione di almeno tre segmenti di retta, come nella figura in basso.

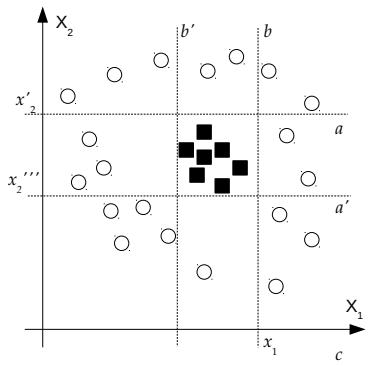


   Per definire rete di percettroni in grado di realizzare le regioni di decisione mostrate in figura, la funzione di attivazione di ognuna delle unità nascoste dovrà corrispondere a una delle rette. Associando la classe dei quadrati neri all'etichetta '1', una possibilità consiste nel fare assumere all'uscita di ciascuna unità nascosta il valore 1 nel semipiano contenente i quadrati neri. In questo modo l'unità di uscita dovrà realizzare la funzione logica `AND` rispetto ai suoi tre ingressi (le uscite delle unità nascoste).

   Le funzioni di attivazione delle unità nascoste dovranno quindi essere pari a 1, in riferimento rispettivamente alle rette a, b e c, nei semipiani definiti dalle seguenti disequazioni: $-X_2 + x'_2 \geq 0$; $-X_1 + x'_1 \geq 0$; $X_1 + X_2 - x''_2 \geq 0$. Per l'unità di uscita si potrà realizzare la funzione `AND` ponendo a 1 i pesi di ciascuna delle connessioni provenienti dalle unità nascoste, e a 2.5 il peso per l'unità di polarizzazione. La rete di percettoni sarà quindi la seguente:



   Infine, dato che le regioni di decisione di un albero di decisione sono composte (in uno spazio a due dimensioni) da rette parallele agli assi, per separare gli esempi delle due classi è necessaria una struttura come quella mostrata in figura (le rette a e b corrispondono a quelle della figura precedente):



Per la costruzione dell'albero di decisione si può osservare che la regione contenente i quadrati neri può essere definita dalla regola: `IF` $X_1 < x_1$ `AND` $X_2 < x_2'$ `AND` $X_1 > x_1'$ `AND` $X_2 > x_2'''$ `THEN` $Y = 1$ Il corrispondente albero di decisione sarà quindi il seguente: