Esercitazione 2: Esercizi sul Concept Learning

Francesco Folino

27 Ottobre 2004

Esercizio 1

Problema. Si consideri lo spazio delle istanze costituite da punti interi nel piano (x, y) ed il set di ipotesi consistenti H rappresentate da rettangoli. Piu' precisamente i rettangoli hanno la forma $a \leq x \leq b$, $c \leq x \leq d$ dove a, b, c, d possono essere interi.

1. Si consideri il *Version Space* rispetto all'insieme di esempi positivi (+) e negativi (-) contenuti nel set di training riportato sotto. Qual'e' l'insieme S in questo caso? Scrivere le ipotesi contenute in S e tracciarle sul diagramma. Il Data Set di training D e' cosí fatto:

$$\begin{aligned} D \quad = \quad & \{\langle(1,3),-\rangle, \langle(2,6),-\rangle, \langle(4,4),+\rangle, \langle(5,3),+\rangle, \langle(5,1),-\rangle, \\ & \langle(5,8),-\rangle, \langle(6,5),+\rangle, \langle(9,4),-\rangle\} \end{aligned}$$

2. Qual'e' il limite G nel *Version Space*? Scrivere le ipotesi e disegnarle.

3. Si supponga che il learner possa ora suggerire una nuova istanza (x, y) e chiedere al trainer di classificarla. Suggerire una query che garantisca di ridurre la dimensione del *Version Space* indipendentemente da come il trainer la classifica. Suggerirne una che, invece, non lo fa.

4. Si assuma adesso di voler insegnare questo particolare *target concept*:

$$S \quad = \quad \{3 \leq x \leq 5, 2 \leq x \leq 9\}$$

   Qual'e' il piu' piccolo numero di esempi di training che devono essere forniti all'algoritmo *Candidate Elimination* affiche' questo riesca ad apprendere perfettamente il concetto dato?

Soluzione.

1. Ricordando le definizioni di *Version Space* quale spazio delle ipotesi consistenti ed S quale insieme delle ipotesi consistenti massimamente specifiche, e' facile osservare che S non e' altro che il piu' piccolo rettangolo che racchiude tutte le istanze positive:

$$S \quad = \quad \{4 \leq x \leq 6, 3 \leq x \leq 5\}$$

assumendo che i punti al bordo del rettangolo siano inclusi nell'ipotesi.

2. G e' l'insieme delle ipotesi consistenti massimamente generali. Intuitivamente, nello spazio delle ipotesi considerato, questo puo' essere rappresentato dal rettangolo piu' grande che esclude gli esempi negativi:

$$S \quad = \quad \{(3 \leq x \leq 8, 2 \leq x \leq 7), (2 \leq x \leq 8, 2 \leq x \leq 5)\}$$

3. Ridurre il VS significa rendere piu' piccola la distanza fra S e G. Per fare cio' e' necessario scegliere una query che riesca a discriminare fra le versioni delle ipotesi contenute nel VS. Un'istanza classificata come positiva da alcune ipotesi nel VS e negativa da altre, puo' andare bene. Questo e' equivalente a dire che un qualunque punto (x, y) collocato tra S e G va bene. Se poi l'istanza scelta e' classificata come positiva dal trainer, l'insieme S sara' reso piu' generale. Se, invece, l'istanza verra' classificata come negativa, si otterra' una specializzazione di G. Il punto $(4, 6)$ e' una scelta corretta. Viceversa, qualunque punto interno ad S o esterno a G lasciera' invariato il VS.

4. Il concetto puo' dirsi appreso quando l'insieme S e l'insieme G coincidono in un'unica ipotesi. Definiamo prima di tutto l'insieme massimamente specifico delle ipotesi consistenti S. La costruzione di S dipende dalle istanze classificate come positive. Possiamo provare a usare 2 esempi positivi ai due angoli diagonalmente opposti del rettangolo che rappresenta il concetto da apprendere. Ad esempio, $(3, 9)$ e $(5, 2)$ o alternativamente $(3, 2)$ e $(5, 9)$. Definito S occorre definire G, scegliere cioe' un insieme di punti classificati come negativi. Per delimitare il rettangolo del concetto abbiamo bisogno di 4 esempi negativi (e' banale osservare che 2 non basterebbero perche' lascerebbero spazio libero per diverse ipotesi consistenti). Abbiamo bisogno di 4 esempi negativi che siano i piu' vicini possibili al nostro concetto da apprendere. Una scelta e' $(2, 5), (4, 1), (4, 10), (6, 5)$. La risposta corretta e' che occorrono 6 queries per apprendere il concetto dato.

Esercizio 2

Problema. Si consideri la seguente sequenza di esempi di training positivi e negativi che descrivono il concetto *"coppie di persone che vivono nella stessa casa"*. Ogni esempio di training descrive una coppia *ordinata* di persone, dove ogni persona descritta dal suo sesso (maschio,femmina), dal colore dei capelli (neri,castani,biondi), dall'altezza (alto,medio,basso) e dalla nazionalita' (americana, francese, tedesca, irlandese, indiana, giapponese, portoghese). Si supponga di avere i seguenti dati di training:

$\{\langle\langle maschio, castani, alto, americana\rangle\ \langle femmina, neri, basso, americ.\rangle\rangle, +\}$

$\{\langle\langle maschio, castani, basso, francese\rangle\ \langle femmina, neri, basso, americ.\rangle\rangle, +\}$

$\{\langle\langle maschio, castani, basso, francese\rangle\ \langle femmina, neri, basso, americ.\rangle\rangle, +\}$

$\{\langle\langle femmina, castani, basso, tedesca\rangle\ \langle femmina, neri, basso, indiana\rangle\rangle, -\}$

$\{\langle\langle maschio, castani, basso, irland.\rangle\ \langle femmina, castani, basso, irland.\rangle\rangle, +\}$

Consideriamo uno spazio delle ipotesi definito su queste istanze, in cui ogni ipotesi e' rappresentata da una coppia di tuple di dimensione 4. Inoltre ogni constraint di un certo attributo puo' essere uno specifico valore oppure '?' oppure '$\emptyset$' cosí come nell'esempio *EnjoySport*.

1. Fornire una traccia manuale dell'algoritmo *Candidate Elimination* sulla base del linguaggio delle ipotesi scelto e degli esempi di training forniti. In particolare, mostrare i limiti S e G del *Version Space* dopo aver processato ogni singolo esempio di training.

2. Quante ipotesi distinte nello spazio delle ipotesi sono consistenti con il seguente esempio di training positivo?

   $\{\langle\langle maschio, neri, basso, portogh.\rangle\ \langle femmina, biondi, basso, indiana\rangle\rangle, +\}$

3. Si supponga che il learner abbia incontrato solo l'esempio positivo sopra indicato e che ora voglia sottoporre delle istance al trainer affinche' siano classificate. Fornire una specifica sequenza di queries che assicurino che il learner convergera' ad una singola ipotesi corretta (qualora sia possibile). Fornire la piu' corta sequenza di queries che soddisfi questa esigenza. Quanto deve essere lunga questa sequenza di domande?

4. Si noti che il linguaggio delle ipotesi scelto non puo' esprimere tutti i concetti che possono essere definiti sulle istanze (e' possibile definire insiemi di esempi positivi e negativi per i quali non ci sono corrispondenti descrivibili ipotesi). Se arricchissimo il linguaggio delle ipotesi in maniera tale da esprimere tutti i concetti che possono essere definiti sul linguaggio delle istanze, come cambierebbe la risposta da dare al punto (3)?

Soluzione.

1. Riportiamo di seguito l'esecuzione step-by-step dell'algoritmo *Candidate Elimination*. Supponiamo che arrivino gli esempi di training (1) e (2), il VS sara':

   $$S_0 = \{\langle\langle\emptyset, \emptyset, \emptyset, \emptyset\rangle\ \langle\emptyset, \emptyset, \emptyset, \emptyset\rangle\rangle\}$$

   $$S_1 = \{\langle\langle maschio, castani, alto, americ.\rangle\ \langle femmina, biondi, basso, americ.\rangle\rangle\}$$

   $$S_2 = \{\langle\langle maschio, castani, ?, ?\rangle\ \langle femmina, biondi, basso, americ.\rangle\rangle\}$$

$$G_0 = G_1 = G_2 = \{\langle\langle ?, ?, ?, ?\rangle\ \langle ?, ?, ?, ?\rangle\rangle\}$$

E' facile osservare che, trattandosi di tuple classificate come positive, queste andranno semplicemente a generalizzare l'insieme S. Supponiamo ora che arrivi l'esempio negativo (3). Il VS si modifica cosí:

$$S_3 = \{\langle\langle maschio, castani, ?, ?\rangle\ \langle femmina, biondi, basso, americ.\rangle\rangle\}$$

$$G_3 = \{\langle\langle maschio, ?, ?, ?\rangle\ \langle ?, ?, ?, ?\rangle\rangle\ \langle\langle ?, ?, ?, ?\rangle\ \langle ?, ?, ?, americ.\rangle\rangle\}$$

La costruzione del VS termina con l'ultima istanza di training (4). Pertanto lo spazio delle ipotesi consistenti costruito sui dati di training sara':

$$S_4 = \{\langle\langle maschio, castani, ?, ?\rangle\ \langle femmina, ?, basso, ?\rangle\rangle\}$$

$$G_4 = \{\langle\langle maschio, ?, ?, ?\rangle\ \langle ?, ?, ?, ?\rangle\rangle\}$$

E' bene osservare come l'arrivo dell'istanza (4) comporti la cancellazione di un'ipotesi in G_3. Non e' infatti vero che $\langle\langle ?, ?, ?, ?\rangle\ \langle ?, ?, ?, americana\rangle\rangle$ sia piu' generale di $\langle\langle maschio, castani, ?, ?\rangle\ \langle femmina, ?, basso, ?\rangle\rangle$. Ricordiamo, infatti, che, date due ipotesi h_1 e h_2, $h_1 \geq h_2$ ($h1$ e' piu' generale di h_2) se ogni istanza soddisfatta da h_2 e' soddisfatta anche da h_1.

2. Applicando *Candidate Elimination* all'unica istanza di training indicata nel punto (2) il *Version Space* ottenuto sara':

   $$S_1 = \{\langle\langle maschio, biondi, basso, portogh.\rangle\langle femmina, biondi, alto, indiana\rangle\rangle\}$$

   $$G_1 = \{\langle\langle ?, ?, ?, ?\rangle\ \langle ?, ?, ?, ?\rangle\rangle\}$$

   Per definizione di VS un'ipotesi e' consistente se e' in esso contenuta. Il VS determinato, indica che sono consistenti tutte quelle ipotesi in cui ogni valore o e' quello specificato in $S1$ oppure e' '?'. Ci sono, quindi, 8 attributi ognuno dei quali puo' assumere due valori diversi, percio' il numero totale di ipotesi sara' $(2 * 2 * 2 * 2)(2 * 2 * 2 * 2) = 2^8 = 256$.

3. Qualunque query che si trovi strettamente tra S e G e' una domanda utile per restringere il campo delle ipotesi consistenti. La migliore query possibile e', pero', quella che viene classificata come positiva da una meta' delle ipotesi in VS e come negativa dalla restante meta'. E' la migliore query possibile perche' riesce a dimezzare la dimensione di VS. La piu' corta sequenza di queries e' proprio quella che ad ogni passo dimezza la dimansione di VS. Se riuscissimo a fare cio' convergeremmo al *target concept* in $\lceil \log_2 |VS| \rceil$. Consideriamo, per esempio, la seguente domanda:

   $$\langle\langle femmina, biondi, basso, portogh.\rangle\langle femmina, biondi, basso, indiana\rangle\rangle$$

se il trainer la classificasse come positiva, allora avremmo il VS fatto cosí:

$$S_2 = \{\langle\langle ?, biondi, basso, portogh.\rangle\langle femmina, biondi, alto, indiana\rangle\rangle\}$$

$$G_2 = \{\langle\langle ?, ?, ?, ?\rangle\ \langle ?, ?, ?, ?\rangle\rangle\}$$

Viceversa, se fosse classificata come negativa:

$$S_2 = \{\langle\langle maschio, biondi, basso, portogh.\rangle\langle femmina, biondi, alto, indiana\rangle\rangle\}$$

$$G_2 = \{\langle\langle maschio, ?, ?, ?\rangle\ \langle ?, ?, ?, ?\rangle\rangle\}$$

In entrambe i casi, $|VS| = 2^7$. Tutto quello che abbiamo fatto e' stato fissare il valore di un attributo che puo' essere '?' oppure uno specifico valore (maschio). Il gioco diventa a questo punto banale. Infatti, se ad ogni passo fissiamo il valore di un attributo (lasciando tutti gli altri invariati a parte uno il cui valore viene commutato in '?') in 8 step riusciamo ad apprendere il concetto. Ecco una sequenza di 8 queries che assolvono il compito previsto:

$\langle\langle \mathbf{femmina}, biondi, basso, portogh.\rangle\langle femmina, biondi, alto, indiana\rangle\rangle$

$\langle\langle maschio, \mathbf{castani}, basso, portogh.\rangle\langle femmina, biondi, alto, indiana\rangle\rangle$

$\langle\langle maschio, castani, \mathbf{alto}, portogh.\rangle\langle femmina, biondi, alto, indiana\rangle\rangle$

$\langle\langle maschio, castani, basso, \mathbf{americ.}\rangle\langle femmina, biondi, alto, indiana\rangle\rangle$

$\langle\langle maschio, castani, basso, portogh.\rangle\langle \mathbf{maschio}, biondi, alto, indiana\rangle\rangle$

$\langle\langle maschio, castani, basso, portogh.\rangle\langle femmina, \mathbf{castani}, alto, indiana\rangle\rangle$

$\langle\langle maschio, castani, basso, portogh.\rangle\langle femmina, biondi, alto, \mathbf{americ.}\rangle\rangle$

La lunghezza della sequenza e' semplicemente logaritmica nella dimensione del VS calcolata nel punto (2): $8 = \log_2 2^8$.

4. Il calcolo della cardinalita' dello spazio delle istanze e' banale:

$$\{(size\{a\} * size\{b\} * size\{c\} * size\{d\}) * (size\{a\} * size\{b\} * size\{c\} * size\{d\})\}$$

dove *a,b,c,d* sono gli attributi che costituiscono la tupla. Avremo quindi: $(2 * 3 * 3 * 7) * (2 * 3 * 3 * 7) = 15876$ possibili istanze. Lo spazio di tutte le possibili ipotesi sulle istanze sara': 2^{15876} (il *power set* dello spazio delle istanze). Riuscendo a dimezzare ad ogni passo lo spazio delle ipotesi, e' facile osservare che apprenderemo il concetto con 15876 queries. In sostanza, rispetto al punto (3), per covergere al *target concept* siamo ora costretti a prendere in considerazione tutte le possibili istanze contenute nello spazio delle istanze.

Esercizio 3

Problema. L'algoritmo *Candidate Elimination* e' stato descritto nel caso in cui erano consentite soltanto ipotesi congiuntive, in cui ogni attributo e' specificato da un particolare valore o da un valore arbitrario (?). Chiamiamo questo algoritmo *Alg1*. Una tipica ipotesi nel caso dell'esempio *EnjoySport* puo' essere scritta cosí:

$$\langle Sunny, Warm, ?, Strong, ?, ? \rangle$$

Si consideri la seguente variazione di questo algoritmo. *Alg2* consente l'uso di disgiunzioni interne *binarie*. Cioe' un attributo puo' essere specificato come una OR fra al piu' due valori:

$$\langle Sunny \vee Cloudy, Warm, ?, Strong, ?, ? \rangle$$

Spiegare come questa estensione al linguaggio delle ipotesi modifichi l'algoritmo *Candidate Elimination* classico. Descrivere un ordine parziale dal generale allo specifico ($\geq$) adattato al nuovo spazio delle ipotesi (fornire un metodo per confrontare coppie di ipotesi). Descrivere come aggiornare le ipotesi contenute negli insiemi S e G per ogni possibile esempio. Qual'e' l'effetto che il cambiamento del linguaggio delle ipotesi ha sul *bias* dell'algoritmo *Candidate Elimination*? Stimare la dimensione dello spazio delle ipotesi per *Alg2* sia rispetto ad *Alg1* sia rispetto allo spazio non influenzato (*unbiased*) assumendo F attributi discreti con $V > 2$ valori ciascuno.

Soluzione. Cominciamo con l'osservare che l'algoritmo classico *Candidate Elimination* puo' essere ancora utilizzato nel caso di ipotesi disgiuntive binarie ma abbiamo bisogno di modificare i metodi di generalizzazione di S e di specializzazione di G.

Determinazione dell'ordine dal generale allo specifico: Supponiamo di avere due ipotesi h_k e h_l. Cominciamo con il capire quando un attributo e' piu' generale (meno specifico) di un altro attributo. Sia a_i il valore di un attributo i in h_k (rappresentato come $a_i^{h_k}$). Diremo allora che $a_i^{h_k}$ e' piu' generale di $a_i^{h_l}$ (valore di un attributo in h_l) se e solo se:

- $a_i^{h_k} =?$ oppure
- $a_i^{h_k} = a_i^{h_l}$ oppure
- $a_i^{h_k}$ e' una disgiunzione binaria $v_1 \vee v_2$ e $a_i^{h_k}$ e' uguale a v_1 o a v_2.

Quindi, h_k e' piu' generale di h_l se e solo se i valori di tutti gli attributi in h_k sono piu'generali dei valori dei corrispondenti attributi in h_l.

Aggiornare S: Un'ipotesi h in S deve essere aggiornata affinche' le nuove ipotesi siano generalizzazioni minimali di h. Nella nostra rappresentazione, se un attributo a_i dell'ipotesi h deve essere generalizzato si puo' utilizzare il seguente approccio (assumiamo ci siano almeno 3 valori per ogni attributo):

- se $a_i = \emptyset$ allora occorre aggiornare l'attributo affinche' $a_i = v_{id}$ dove v_{id} e' il valore di tale attributo nell'esempio di training positivo X_d.

- se $a_i = v$ aggiorniamo l'attributo affinche' $a_i = (v \vee v_{id})$ dove v_{id} e' il valore di tale attributo nell'esempio di training positivo X_d.

- se $a_i = (v_1 \vee v_2)$ allora aggiorniamo l'attributo cosí $a_i =?$.

Esempio: Supponiamo che $S = (Sunny, Warm, ?, Strong, ?, ?)$ ed il dato di training sia $(Cloudy, Warm, Normal, Strong, Warm)$. Allora S diventa $S = (Sunny \vee Cloudy, Warm, ?, Strong, ?, ?)$.

Aggiornare G: Un'ipotesi h in G deve essere aggiornata in maniera tale che le nuove ipotesi in esso contenute siano specializzazioni minimali di h. Un atributo a_i puo' essere specializzato nel seguente modo:

- se un attributo $a_i =?$, allora inseriamo in G tutte le combinazioni di $a_i = v_k$ e $a_i = (v_k \vee v_l)$ dove $v_k, v_l \in V_i$ e $v_k \neq v_l, v_k \neq v_{id}, v_l \neq v_{id}$ dove v_{id} e' il valore di tale attributo nell'esempio di training negativo.

- se un attributo $a_i = (v \vee v_{id})$ dove v_{id} e' il valore di tale attributo nell'esempio di training negativo, allora aggiorniamo a_i cosí: $a_i = v$.

Esempio: Assumiamo che G=(?, Warm, Normal, Strong, Warm, Same) e che l'esempio di training negativo sia (Cloudy, Warm, Normal, Strong, Warm, Same). Allora il nuovo insieme G diventa G=(Sunny $\vee$ Overcast, Warm, Normal, Strong, Warm, Same). E' importante osservare che (Sunny, Warm, Normal, Strong, Warm, Same) e (Overcast, Warm, Normal, Strong, Warm, Same) sono entrambe piu' specializzati di (Sunny $\vee$ Overcast, Warm, Normal, Strong, Warm, Same) e quindi non possono essere aggiunti in G nell'algoritmo *Candidate Elimination*.

Questo "nuovo" algoritmo, consentendo la disgiunzione binaria fra valori di attributi, amplia la dimensione dello spazio delle ipotesi ma questo e' ancora troppo piccolo se lo confrontiamo con lo spazio di tutte le possibili ipotesi. La dimensione dello spazio delle ipotesi puo' essere trovato nel seguente modo. Un attributo a_i puo' assumere $|V_i|$ valori. Questo e' equivalente a dire che a_i puo' contenere una delle seguenti rappresentazioni all'interno dell'ipotesi h: $?, v_0, v_1, ..., v_{|V_i|}, (v_0 \vee v_1), (v_0 \vee v_2), ..., (v_{|V_i|-1} \vee v_{V_i})$. Ci sono percio'$(1 + |V_i| + + \frac{|V_i|(|V_i|-1)}{2})$ possibili rappresentazioni per questo attributo. Ricordando che la dimensione dello spazio delle ipotesi nell'algoritmo *Candidate Elimination* e' $1 + size(a_0) * size(a_1) * .... * size(a_n)$, la dimensione dello spazio delle ipotesi nell'algoritmo modificato sara' $1 + \Pi_i(1 + |V_i| + \frac{|V_i|(|V_i|-1)}{2})$.