

Indice

1	Analisi dei dati di probabilità	2
1.1	Problema in esame	2
1.2	Caratteristiche degli eventi di coppia	2
1.2.1	Eventi indipendenti	3
1.2.2	Eventi dipendenti	3
1.3	Evento conosciuto ed evento indovinato	4
1.3.1	Probabilità di rispondere correttamente ad una domanda	5
1.3.2	Il piano	7
2	Rete neurale	12
2.1	Test effettuati	13
2.1.1	Configurazione della rete: 4 neuroni per ciascuno dei 2 layers	14
2.1.2	Configurazione della rete a 2 neuroni per ciascuno dei 2 layers	15
2.1.3	Configurazione della rete a 4 neuroni per 1 layer	18
2.2	Risultati delle previsioni	19

1 Analisi dei dati di probabilità

Durante il periodo 20/05 - 24/05 mi sono occupata di analizzare la probabilità che ha un candidato di rispondere correttamente alle domande in fase di test; valutando le relazioni di dipendenza che possono esistere tra più domande e l'impatto che può assumere la fortuna.

1.1 Problema in esame

Test, sottoposto ad un candidato durante un colloquio, composto da *domande a tripla risposta multipla*.

Nel suddetto documento vengono analizzate le relazioni che intercorrono tra due domande, denominate A e B, a seconda se il candidato risulta in grado di rispondervi correttamente o meno.

1.2 Caratteristiche degli eventi di coppia

Tipi di eventi trattati:

- **Eventi indipendenti;**
- **Eventi dipendenti:**
 - A e B sono strettamente dipendenti;
 - A implica B.
- **Evento conosciuto ed evento indovinato.**

Struttura usata per rappresentare la probabilità degli eventi di coppia:

AB

\wedge

$A\ B$

\vee

Z

con:

- AB rappresenta la probabilità complessiva dell'evento che si verifica sempre;
- A rappresenta la probabilità che permette il verificarsi di A, ma non di B;

- B rappresenta la probabilità che permette il verificarsi di B, ma non di A;
- Z rappresenta la probabilità a zero, l'impossibilità del verificarsi dell'evento.

1.2.1 Eventi indipendenti

A e B sono due domande la quali risposte sono completamente scorrelate tra di loro.

$$\begin{array}{c}
 P(A)P(B) \\
 /\backslash \\
 P(A)(1 - P(B)) \quad (1 - P(A))P(B) \\
 \backslash/ \\
 (1 - P(A))(1 - P(B))
 \end{array}$$

Considerazioni generali La probabilità complessiva nel caso di domande indipendenti A e B viene data da $P(A)$ per $P(B)$.

Se è conosciuta dal candidato la risposta alla domanda A ma non alla domanda B la probabilità di ottenere una risposta corretta è $P(A)$, mentre la probabilità di ottenere una risposta non corretta per B vale $1 - P(B)$. Il ragionamento duale è svolto nel calcolo della probabilità per la risposta corretta alla domanda B ma non ad A.

La probabilità di non ottenere alcuna risposta corretta alle due domande viene calcolata prendendo in considerazione gli eventi contrari a quelli coinvolti. Dunque per A la probabilità che il candidato non conosca la soluzione è $1 - P(B)$, dualmente per B la probabilità è $1 - P(A)$.

1.2.2 Eventi dipendenti

A e B sono due domande fortemente correlate tra di loro se si risponde correttamente ad una delle due domande si risponde correttamente anche all'altra.

$$\begin{array}{c}
 P(A)^2 \\
 /\backslash \\
 0 \quad 0 \\
 \backslash/ \\
 (1 - P(A))^2
 \end{array}$$

Considerazioni generali La probabilità complessiva nel caso di domande dipendenti A e B viene data da $P(A)$ per $P(B)$; ma $P(A) = P(B)$ dunque $P(A)^2 = P(B)^2$.

Conseguentemente se il candidato non conosce la risposta alla domanda A non può conoscere la risposta alla domanda B per cui la probabilità di conoscere uno dei due eventi è pari a 0.

In questo caso la probabilità a 0 è $(1 - P(A))(1 - P(B)) = (1 - P(A))^2$ essendo che $A=B$.

A implica B Se si sa rispondere alla domanda A di conseguenza si è in grado di rispondere anche alla domanda B.

Tuttavia non vale il ragionamento opposto, se si sa rispondere alla domanda B non significa che si è in grado di rispondere alla domanda A.

$$P(A)$$

$$\wedge$$

$$0 \quad P(B) - P(A)$$

$$\vee$$

$$1 - P(B)$$

Considerazioni generali La probabilità complessiva nel caso di domande dipendenti A e B viene data esclusivamente da $P(A)$ in quanto la conoscenza di sia di A che di B è possibile solo se si ha piena conoscenza di A.

Dunque la probabilità che si conosca la risposta alla domanda A ma non a B è impossibile (pari a 0); mentre se si ha conoscenza della domanda B ma non di A la probabilità si stanziava a $P(B) - P(A)$.

La probabilità a zero è $1 - P(B)$ indicatore dell'impossibilità di avere la risposta corretta per A.

1.3 Evento conosciuto ed evento indovinato

Durante un test il candidato deve saper scegliere la risposta, corretta o meno, alla domanda posta. Le variabili che entrano in gioco durante l'esecuzione dell'atto non riguardano esclusivamente la conoscenza personale del singolo. La probabilità di un evento A è data dalla formula:

$$P(A) = P(A_C) + P(A_I)$$

Le variabili in uso sono:

- $P(A_C)$: probabilità che il candidato sappia rispondere alla domanda A correttamente per sua conoscenza;
- $P(A_I)$: probabilità che il candidato sappia rispondere alla domanda A correttamente indovinando.

Per quanto appena definito sopra valgono le seguenti proprietà:

1. $P(B_C|A_C) = 1$
2. $P(B_C|A_I) = P(B_C)$
3. $P(B_I|A_C) = 0$
4. $P(B_I|A_I) = P(B_I)$

1.3.1 Probabilità di rispondere correttamente ad una domanda

Variabili coinvolte:

- $P(A)$: probabilità necessaria perchè si verifichi, per la domanda A, che il candidato dia la risposta corretta. Per la legge dei grandi numeri la frequenza porta alla probabilità.
- S_0 : insieme dei casi in cui in un domanda non viene scartata alcuna risposta dal dominio delle risposte possibili;
- S_1 : insieme dei casi in cui in una domanda viene scartata una risposta dal dominio delle risposte possibili;
- S_2 : insieme dei casi in cui in una domanda vengono scartate due risposte dal dominio delle risposte possibili.
- $P(I)$: probabilità di dare la risposta corretta alla domanda A indovinando;
- $P(C)$: probabilità di dare la risposta corretta alla domanda A per conoscenza.

Sapendo che $P(I) = P(A) - P(C)$ logicamente vale anche $P(A) = P(I) + P(C)$.

Se un candidato non è in grado scartare alcuna risposta dalla domanda ha 1 possibilità su 3 di, indovinando, dare la risposta corretta. Se un candidato invece risulta in grado di scartare una risposta, sbagliata, alla domanda rimane con 1 possibilità su 2 di poter dare la risposta corretta. Se invece, caso ottimo, il candidato ha piena conoscenza della domanda posta risulta in grado di scartare due risposte sbagliate lasciando un'unica risposta possibile, quella esatta. Il ragionamento sopra espresso può venire espresso con la seguente espressione:

$$P(A) = P(S_0)\frac{1}{3} + P(S_1)\frac{1}{2} + P(S_2)$$

Ora individuiamo quale è la probabilità effettiva per un candidato di dare la risposta corretta ad una domanda A.

$$\begin{aligned} 1 &= S_0 + S_1 + S_2 \\ S_0 &= 1 - S_1 - S_2 \end{aligned}$$

Sostituendo:

$$\begin{aligned} P(A) &= (1 - P(S_1) - P(S_2))\frac{1}{3} + P(S_1)\frac{1}{2} + P(S_2) \\ &= \frac{1}{3} - \frac{1}{3}P(S_1) - \frac{1}{3}P(S_2) + \frac{1}{2}P(S_1) + P(S_2) \\ &= \frac{1}{3} + \frac{1}{6}P(S_1) + \frac{2}{3}P(S_2) \end{aligned}$$

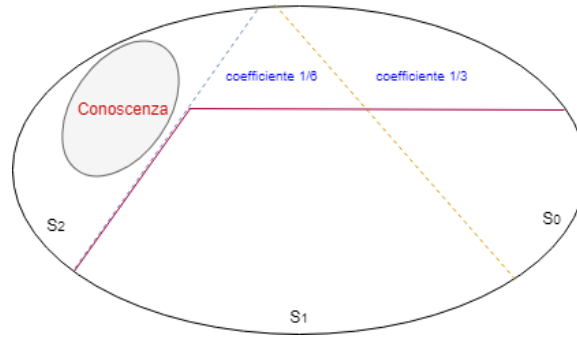


Figura 1: Rappresentazione insiemistica della probabilità di rispondere correttamente ad una domanda: $P(A)$

Considerazioni importanti

In conclusione $P(A) = \frac{1}{3} + \frac{1}{6}P(S_1) + \frac{2}{3}P(S_2)$. Ovvero la probabilità per un candidato di dare in una domanda A la risposta corretta dipende dai seguenti fattori:

- $\frac{1}{3}$: coefficiente che rappresenta la probabilità effettiva per chi non conosce la risposta alla domanda di dare la risposta corretta;
- $\frac{1}{2}P(S_1)$: coefficiente che rappresenta la probabilità effettiva di dare la risposta corretta quando il candidato è in grado di scartare una risposta sbagliata alla domanda;
- $\frac{2}{3}P(S_2)$: coefficiente che rappresenta la probabilità effettiva di dare la risposta corretta quando il candidato è in grado di scartare due risposte sbagliate alla domanda.

Dall'analisi della tipologia di eventi di coppia e dal calcolo della probabilità necessaria per poter rispondere correttamente ad una domanda, si è giunti alla valenza dei seguenti assiomi:

1. Le coppie di domande A e B devono essere fra loro disgiunte, altrimenti si genererebbero situazioni di invalidità dei risultati;
2. Per rispondere correttamente ad una domanda non è necessario che il candidato abbia piena conoscenza di tutti gli argomenti richiesti dall'esame, ma bensì ne risultano sufficienti $n - 1$;
3. La probabilità di conoscere è contenuta all'interno di S_2 , in quanto se un candidato conosce è conseguentemente in grado, da una domanda, di scartare due risposte sbagliate.

1.3.2 Il piano

La probabilità $P(A)$ che un candidato ha in gioco nel momento in cui si approccia a rispondere ad una domanda può venire rappresentata in un piano.

Di seguito viene mostrata l'immagine di un modellino, rappresentativo di $P(A)$, realizzato durante l'analisi.

TODO: foto modello

Ognuno dei tre assi cartesiani rappresenta un insieme dei casi di scarto (S_0 , S_1 , S_2). L'intersezione tra i punti del piano indica la regione accettabile contenente il range di valori assumibili da $P(A)$. Tale punto proiettato su ognuno dei tre assi permette l'individuazione esatta dei coefficienti delle variabili S_0 , S_1 , S_2 .

Ogni porzione del piano viene individuata con la seguente tecnica:

1. Per individuare ogni retta passante per S_0 , S_1 e S_2 è necessario assumere che $S_0 + S_1 + S_2 = 1$;
2. La retta passante per S_0 è rappresentabile per mezzo delle seguenti equazioni:

$$S_0 = 0 \text{ e } S_1 + S_2 = 1$$

In questo modo l'asse S_0 è fissato a 0 e estrapolando S_1 e S_2 da $S_1 = -S_2 + 1$ assumono valori tra (0,1).

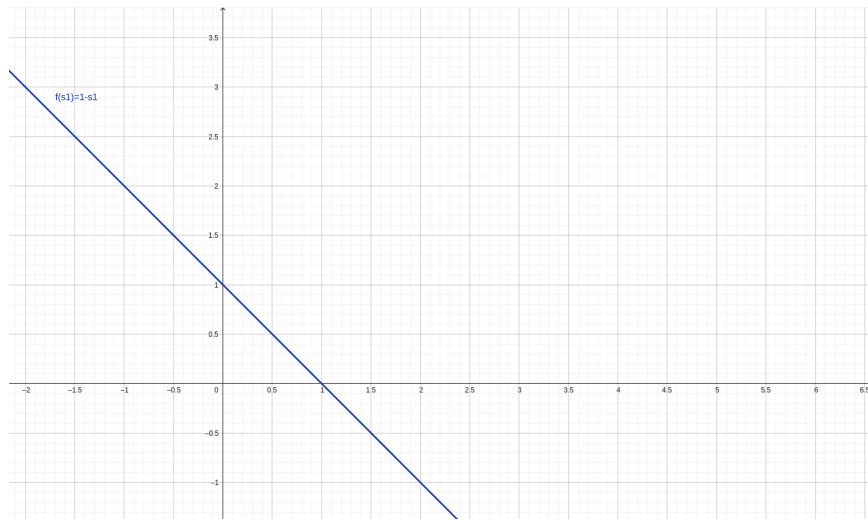


Figura 2: Rappresentazione della retta passante per $S_0 = 0$

3. Il medesimo ragionamento vale per le rette passanti per S_1 e S_2 .

$$S_1 = 0 \text{ e } S_0 + S_2 = 1$$

l'asse S_1 è fissato a 0 e S_0 e S_2 assumono valori tra (0,1).

$$S_2 = 0 \text{ e } S_1 + S_0 = 1$$

l'asse S_2 è fissato a 0 e S_0 e S_2 assumono valori tra (0,1).

4. In questo modo l'unione di tutte le rette passanti per gli assi creano la regione accettabile dei valori di $P(A)$.

Avendo rappresentato il piano si ottiene nei punti di intersezioni fra le tre rette la regione accettabile per $P(A)$. Inoltre è possibile, ora, individuare il fascio di rette che tangenti il piano permettono di affermare se una specifica domanda è, in base alla sua frequenza, ha difficoltà bassa, media, alta per un candidato.

- Se una domanda ha una difficoltà bassa la retta si situa passante per i punti $0 < S_2 \leq 1$ (molto vicino a 1) e $(S_0, S_1) < 0$ (tendenti a 0);
- Se una domanda ha una difficoltà alta la retta si situa passante per i punti $S_2 \leq 0$ (molto vicino a 0), $S_1 < 1$ e $S_0 \leq 1$ (tendente a non scartare alcuna risposta);

- Se una domanda ha una difficoltà media la retta si situa nella parte centrale della regione accettabile, passante per i punti $0 \leq (S_0, S_1, S_2) \leq 1$.

Rappresentazione di P(A)

Vediamo alcuni casi di come le domande possono venire rappresentate sul piano:

La funzione di partenza è:

$$F = \frac{1}{3} + \frac{1}{6}S_1 + \frac{2}{3}S_2$$

Va esplicitato S_1 , i passaggi utili da fare sono i seguenti:

$$\frac{-1}{6}S_1 = \frac{1}{3} + \frac{2}{3}S_2 - F \rightarrow S_1 = -4S_2 - 2 + 6F$$

Essendo che $0 \leq S_2 \leq 1$ usando $S_1 = 1$ e $S_2 = 0$ allora si ottiene che $F = \frac{1}{2} = 0.5$

Quanto appena calcolato può venire rappresentato graficamente impiegando la retta $S_1 = 1 - S_2$ (responsabile di definire una porzione del piano in base alle variabili coinvolte) e mediante la retta $S_1 = -4S_2 - 2 + 6F$ (che permette di calcolare il fascio di rette tangenti alla prima retta).

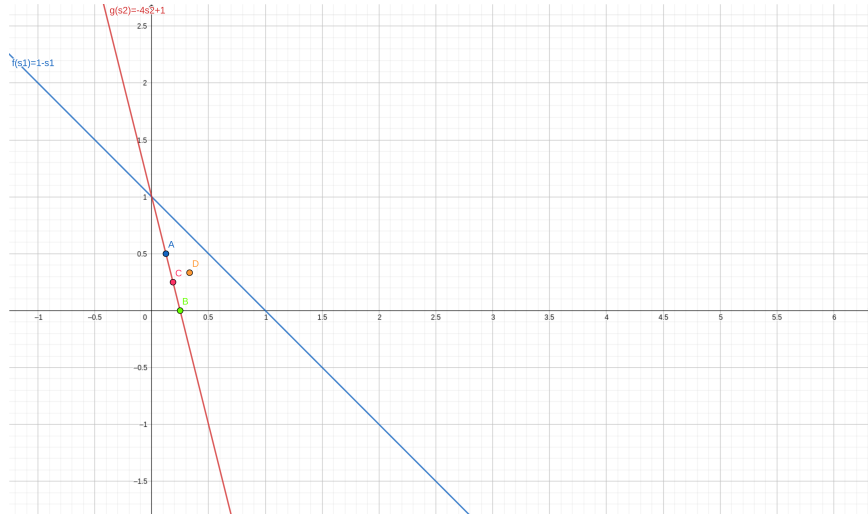


Figura 3: Rappresentazione di P(A) per una frequenza 0.5 proiettata su assi $S_0 = 0$, S_1 e S_2 .

Nella figura sopra sono rappresentati i seguenti significati:

- La linea azzurra rappresenta $S_2 = 1 - S_1$;
- La linea rosa rappresenta la retta tangente $S_1 = -4S_2 + 1$;
- Punto A (blu):

$$S_1 = 0.5 = \frac{1}{2}$$

$$\frac{1}{2} = -4S_2 + 1 \rightarrow 4S_2 = 1 - \frac{1}{2} \rightarrow S_2 = \frac{1}{8}$$

$$S_0 = 1 - \frac{1}{2} - \frac{1}{8} = \frac{3}{8}$$

Ovvero metà dei candidati sottoposti alla domanda sa scartare una delle risposte, lo 0.16% sa dare la risposta corretta e lo 0.36% non sa scartare alcune delle risposte possibili.

- Punto B (verde):

$$S_1 = 0$$

$$S_2 = \frac{1}{4}$$

$$S_0 = 1 - \frac{1}{4} = \frac{3}{4}$$

Ovvero nessun dei candidati sottoposti alla domanda sa scartare una delle risposte, lo 0.25% sa dare la risposta corretta e lo 0.75% non sa scartare alcune delle risposte possibili.

- Punto C (fucsia):

$$S_1 = \frac{1}{4}$$

$$S_2 = \frac{3}{16}$$

$$S_0 = 1 - \frac{1}{4} - \frac{3}{16} = \frac{9}{16}$$

Ovvero lo 0.25% dei candidati sottoposti alla domanda sa scartare una delle risposte, lo 0.19% sa dare la risposta corretta e lo 0.56% non sa scartare alcune delle risposte possibili.

- Punto D (arancione):

$$S_1 = \frac{1}{3}$$

$$S_2 = \frac{1}{3}$$

$$S_0 = 1 - \frac{1}{3} - \frac{1}{3} = \frac{1}{3}$$

Osserviamo che il punto in esame fuoriesce dalla regione delimitata dalla retta tangente di frequenza 0.5 ($S_1 = -4S_2 + 1$). Conseguenza diretta data dall'impossibilità di ottenere una probabilità del 50% sulla domanda con $\frac{1}{3}$ di candidati che sa scartare 2 risposte, $\frac{1}{3}$ che ne sa scartare 1 e $\frac{1}{3}$ nessuna.

Vediamo ulteriori due esempi che permettono di valutare cosa accade nel piano nel caso di una frequenza:

1. Quasi in prossimità di 1;

2. Pari alla soglia minima dell'indovinato.

Il grafico è il seguente:

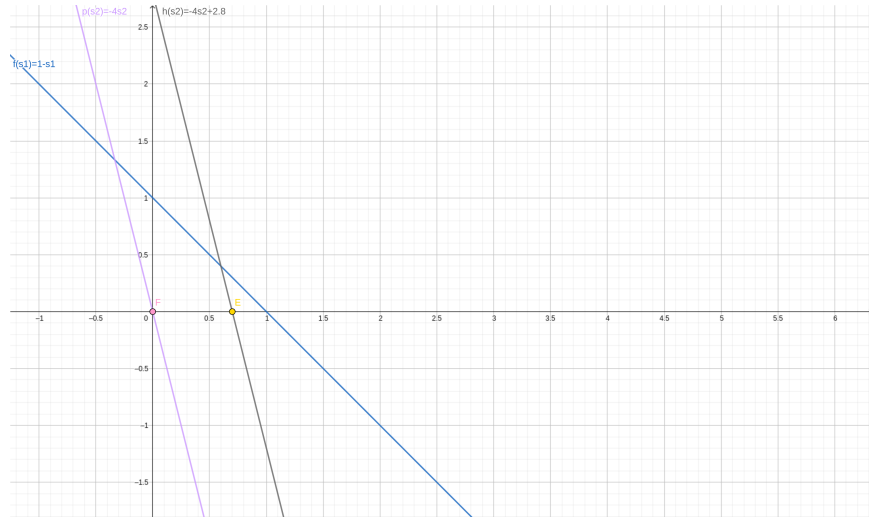


Figura 4: Rappresentazione di $P(A)$ per una frequenza 0.33 e 0.8 proiettate su assi $S_0 = 0$, S_1 e S_2 .

- La linea azzurra mostra la retta tangente con frequenza 0.80%. In questa abbiamo calcolato il punto E (giallo):

$$\begin{aligned} S_1 &= 0 \\ S_2 &= \frac{14}{20} \\ S_0 &= 0 \end{aligned}$$

Quasi la totalità dei candidati ha la conoscenza per poter scartare tutte le risposte sbagliate e dare la risposta giusta alla domanda.

- La linea viola mostra la retta tangente con frequenza 0.33%. In questa abbiamo calcolato il punto F (rosa):

$$\begin{aligned} S_1 &= 0 \\ S_2 &= 0 \\ S_0 &= 1 \end{aligned}$$

Ovvero nessuno dei candidati ha la conoscenza per poter scartare nè una nè due risposte, per cui l'unica possibilità per un candidato di rispondere alla domanda è indovinare. È evidente come se un candidato non sa la risposta ad una domanda ha una probabilità dello 0.33% di poter indovinare la risposta corretta.

2 Rete neurale

La libreria utilizzata per sviluppare la Rete neurale è stata *ConvNetJS*. L'aspetto positivo di tale scelta è stata la semplicità nell'utilizzo del linguaggio javascript; l'aspetto negativo ha riguardato la totale mancanza di mantenibilità della libreria stessa che comporta la scarsità di esempi applicativi, oltre alla documentazione ufficiale, che costringono lo sviluppatore ad una ricerca approfondita personale in un ambiente ove le nozioni si presentano scarse e a continue prove per verificare la validità del codice prodotto.

Questions test - Prevision Neural Net

CONFIGURAZIONE RETE NEURALE

```
layer_defs = [];  
layer_defs.push({type:'input', out_sz:1, out_sy:1, out_depth:6});  
layer_defs.push({type:'fc', num_neurons:2, activation: 'tanh'});  
layer_defs.push({type:'fc', num_neurons:2, activation: 'tanh'});  
layer_defs.push({type:'regression', num_neurons:6});  
  
net = new convnetjs.Net();  
net.makeLayers(layer_defs);  
  
trainer = new convnetjs.SGDTrainer(net, {learning_rate:0.01, momentum:0.1, batch_size:10, l2_decay:0.001});
```

OUTPUT DELLA RETE

```
Richiesta di previsione inoltrata alla rete ...  
Richiesta di previsione accettata  
Il vettore [0,1,0,0,0,0] ha previsione calcolata di  
[-0.14178025275470646,-0.014639237878831693,-0.00131295482446743,0.11790470015357472,0.14410273493990078,0.1329311955165462]  
Rete Neurale in attesa ... Inserire risultati del test...  
Inserimento risposte alle domande del test andato a buon fine  
Ricapitolazione dati inseriti: Risposte ottenute [0,-1,0,0,0,0]  
Inizio allenamento della rete ...  
Richiesta di previsione inoltrata alla rete ...  
Richiesta di previsione accettata  
Il vettore [0,1,1,0,0,0] ha previsione calcolata di  
[-0.9703841414997074,-0.7824502014273779,0.34748891066895563,-0.8448991676224333,-0.5825080715009475,0.29369850449980084]  
Rete Neurale in attesa ...
```

INSERIMENTO DATI DI ADDESTRAMENTO

Number of fields (max 9999): [Insert fields](#)

Input1 Input2 Input3 Input4 Input5 Input6

[save](#)

[start trainer](#)

Vettore di previsione

[prevision trainer](#)

Figura 5: Interfaccia utente della Rete neurale di prova.

Durante il periodo 24/05 - 31/05 mi sono occupata dello sviluppo di una Rete neurale in grado di ricevere in input un training set di dimensione 6 e di restituire una previsione sui dati di apprendimento ricevuti. Il problema che la rete mira ad analizzare è quello discusso nel precedente capitolo *Analisi dei dati di probabilità*

Per agevolare l'apprendimento della rete, ed ottenere delle previsioni stabili mi sono occupata di implementare due metodi di generazione randomica di dati in modo da far apprendere massicciamente la stessa. Il dato prodotto consiste in un vettore di 6 elementi, composto da -1, 0 e 1 con il seguente criterio:

- **-1**: la domanda x è stata posta al candidato che ha risposto in maniera errata;
- **0**: la domanda x non è stata posta al candidato;
- **1**: la domanda x è stata posta al candidato che ha saputo rispondere correttamente.

Il primo metodo che ho sviluppato si occupa di generare un vettore di dati di apprendimento basandosi esclusivamente su come le domande sono interconnesse tra di loro (grazie all'uso di un grafo della conoscenza costruito ad hoc); il secondo metodo ripropone quanto perseguito dal primo metodo con il valore aggiunto di generazione di un profilo randomico di un candidato, che tiene conto della probabilità di risposta ad una domande seguendo la formula $P(A) = \frac{1}{3} + \frac{1}{6}P(S_1) + \frac{2}{3}P(S_2)$.

2.1 Test effettuati

Alcune decisioni che ho preso durante la configurazione della rete riguardano i seguenti settori:

1. Una rete neurale non deve, per fornire dei dati attendibili, possedere un numero di neuroni troppo elevato rispetto al trainset effettuato; altrimenti la previsione ritornerebbe l'identità del vettore di input della stessa, come conseguenza diretta della capacità troppo elevata di immagazzinare dati.
2. I layers, ho deciso, di allenarli mediante tecnica di regressione, che permette l'inserimento in input di una funzione obiettivo e l'ottenimento di un risultato, in output, anche in virgola mobile e composto di tanti elementi quanti sono i neuroni di regressione dichiarati. Per la mia rete di prova è necessario dichiarare 6 neuroni in regressione perchè l'output, appunto, che ci si aspetta dal sistema è di 6 elementi.
3. Per costruire un dataset di dati consistente che permettesse alla rete di imparare qualcosa ho costruito un grafo della conoscenza con lo scopo di mettere in relazione degli argomenti che coinvolgono uno o più domande.

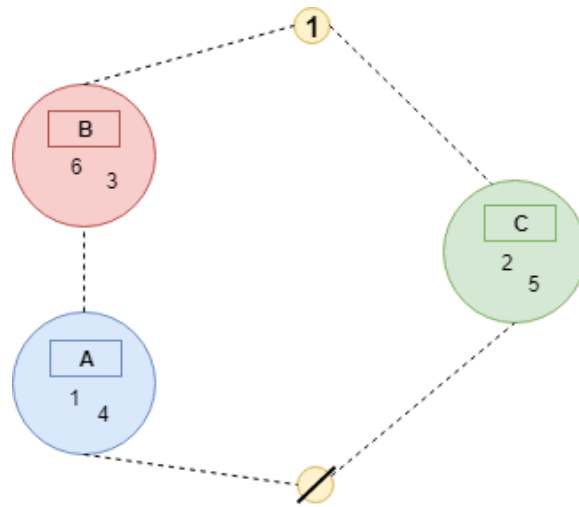


Figura 6: Grafo rappresentante le relazioni esistenti tra il set di domande di prova.

Per svolgere l'apprendimento ogni vettore, facente parte del dataset, viene dato in pasto alla rete che a sua volta provvede alla sua assimilazione come conoscenza mediante la tecnica dell'autoencoder, ovvero la rete impara il vettore riducendone lo spazio occupato.

4. Per creare il dataset ho ritenuto sufficiente generare *2000* vettori di risposta in modo da compiere in maniera esaustivo l'apprendimento della rete.

Il vettore passato in input per svolgere le previsioni è $[0,0,0,0,0,0]$, $[0,0,1,0,1,0]$ e $[0,0,-1,0,0,0]$

Le aspettative riguardano la previsione di risposta di un candidato . Di seguito riporto quanto è stato rilevato in fase di test.

2.1.1 Configurazione della rete: 4 neuroni per ciascuno dei 2 layers

Configurazione della rete utilizzata:

```
layer_defs = [];
layer_defs.push({type:'input', out_sx:1, out_sy:1, out_depth:6});
layer_defs.push({type:'fc', num_neurons:4, activation: 'tanh'});
layer_defs.push({type:'fc', num_neurons:4, activation: 'tanh'});
layer_defs.push({type:'regression', num_neurons:6});

net = new convnetjs.Net();
```

```
net.makeLayers(layer_defs);
```

```
trainer = new convnetjs.SGDTrainer(net, {learning_rate:0.01,  
momentum:0.1, batch_size:10, l2_decay:0.001});
```

I layers utilizzati sono 2 e composti da 4 neuroni.

Training set standard a 4 neuroni per ciascuno dei 2 layers

- Il vettore [0,0,0,0,0,0] ha previsione calcolata di [-0.021598804903572744,-0.11264571886853292,0.17520370837747462]

Appaiono in relazione le domande 1, 2, 5 e 3, 4, 6.

Gli scostamenti tra le coppie 2, 5 e 3, 6 sono consistenti con quelle che sono le relazioni di dipendenza, invece 1, 4 ha una differenza di 0.016 circa che parte da qualche millesimo fino 0.5 Le domande 3 e 6 si dovrebbero presentare con una positività inferiore rispetto a 1 e 4; nel test in analisi questo non viene rispettato da nessuna delle coppie in analisi per differenze che vanno da qualche millesimo fino a 0.018 circa.

Osservazioni

La configurazione testata si compone di 4 neuroni a layer su una base di 2000 test correndo il rischio di avere una rete che apprende troppo e come effetto negativo "veda" addirittura cose che non esistono. A prova di ciò sono i risultati non conformi alle attese. Dunque mi fermo qui con il test di tale configurazione e riducendone il numero di neuroni presenti in ciascun layers e/o il numero di layers presenti.

Le nuove configurazione su cui ho effettuato i test sono esposte nei paragrafi seguenti.

2.1.2 Configurazione della rete a 2 neuroni per ciascuno dei 2 layers

Configurazione della rete utilizzata:

```
layer_defs = [];  
layer_defs.push({type:'input', out_sx:1, out_sy:1, out_depth:6});  
layer_defs.push({type:'fc', num_neurons:2, activation: 'tanh'});  
layer_defs.push({type:'fc', num_neurons:2, activation: 'tanh'});
```

```

layer_defs.push({type:'regression', num_neurons:6});

net = new convnetjs.Net();
net.makeLayers(layer_defs);

trainer = new convnetjs.SGDTrainer(net, {learning_rate:0.01,
momentum:0.1, batch_size:10, l2_decay:0.001});

```

I layers utilizzati sono 2 composti da 2 neuroni.

Training set standard su rete a 2 neuroni per ciascuno dei 2 layers

- Il vettore [0,0,0,0,0,0] ha previsione calcolata di [0.31232372051574936,0.7253754889487585,-0.5051208979797573,0.32075742158673093,0.7324947496336937,-0.4348299972940168]
 Appaiono in relazione le domande 1, 2, 4, 5 e 3, 6.
 Gli scostamenti tra le coppie 1, 4 e 3, 6 e 2, 5 sono consistenti con quelle che sono le relazioni di dipendenza fra le domande. Le domande 3 e 6 si dovrebbero presentare con una positività inferiore rispetto a 1 e 4; in questo test la regola viene rispettata pienamente.
 Dall'immagine canvas e dai dati della previsione si nota come il candidato ha una buona probabilità di saper rispondere alla coppia 1 e 4, e ancora più elevata di saper rispondere correttamente alla coppia 2 e 5; molto bassa di saper rispondere correttamente alle 3 e 6 che sono, appunto, di una difficoltà maggiore rispetto alla coppia 1 e 4.
- Il vettore [0,0,1,0,1,0] ha previsione calcolata di [0.5123144717131076,0.9123354449531641,0.2837937822420923,0.46449868699771607,0.9029832167165894,0.3227303792035435]
 Appaiono in relazione le domande 1, 2, 3 4, 5, 6.
 Gli scostamenti tra le coppie 1, 4 e 3, 6 e 2, 5 sono consistenti con quelle che sono le relazioni di dipendenza fra le domande. Le domande 3 e 6 si dovrebbero presentare con una positività inferiore rispetto a 1 e 4; in questo test la regola viene rispettata pienamente.
 Dall'immagine canvas e dai dati della previsione si nota come il candidato ha un'ottima probabilità di saper rispondere alla coppia 2 e 5 (come imposto dal vettore previsione), buona di saper rispondere alla coppie 3 e 6 (come imposto dal vettore previsione) e più che buona di saper rispondere alle 1 e 4, che sono di una semplicità più elevata rispetto alla 3 e 4.

- Il vettore $[0,0,-1,0,0,0]$ ha previsione calcolata di $[0.3698539826215957, 0.288907514487717, -0.8504159455662308, 0.3663192502433841, 0.2937448801761998, -0.7845589473185985]$
 Appaiono in relazione le domande 1, 2, 4, 5 e 3, 6.
 Gli scostamenti tra le coppie 1, 4 e 3, 6 e 2, 5 sono consistenti con quelle che sono le relazioni di dipendenza fra le domande. Le domande 3 e 6 si dovrebbero presentare con una positività inferiore rispetto a 1 e 4; in questo test la regola viene rispettata pienamente.
 Dall'immagine canvas e dai dati della previsione si nota come il candidato ha una discreta probabilità di saper rispondere alla coppia 2 e 5, un pò meglio di saper rispondere alla coppie 1 e 4 e più di non saper saper rispondere alle 3 e 6 (come imposto dal vettore previsione).

Training set con generazione del profilo di un candidato e calcolo delle probabilità di risposta a 2 neuroni per ciascuno dei 2 layers

- Il vettore $[0,0,0,0,0,0]$ ha previsione calcolata di $[0.057781303506280995, 0.0513731100126314, -0.06600467867066256, 0.029940883111932555, -0.019564515397168573, -0.09570617900597932]$
 Appaiono in relazione le domande 1, 2, 4 e 3, 5, 6.
 Gli scostamenti tra la coppia 1, 4 e 3, 6 sono consistenti con quelle che sono le relazioni di dipendenza fra le domande; invece per la coppia 2 e 5 i segni sono opposti con una differenza di 0.024. Le domande 3 e 6 si dovrebbero presentare con una positività inferiore rispetto a 1 e 4, la regola viene rispettata pienamente. Le anomalie riscontrate sono da ricondurre alla natura del vettore di training che si basa sul calcolo della probabilità di una risposta che sul grafo della conoscenza.
- Il vettore $[0,0,1,0,1,0]$ ha previsione calcolata di $[0.19494624113789977, 0.1712744021266377, 0.577963304906936, 0.781098215373483, 0.3774535909060714, 0.03617314870307162]$
 Appaiono in relazione le domande 1, 2, 3, 4, 5, 6.
 Gli scostamenti tra le coppie 1 e 4 , 2 e 5, 3 e 6 sono consistenti con quelle che sono le relazioni di dipendenza fra le domande. Le domande 3 e 6 si dovrebbero presentare con una positività inferiore rispetto a 1 e 4, la regola non viene rispettata dalla domanda 1 in rapporto con la domanda per una differenza di 0.37 circa. Le anomalie riscontrate sono da ricondurre alla natura del vettore di training che si basa sul calcolo della probabilità di una risposta che sul grafo della conoscenza.
- Il vettore $[0,0,-1,0,0,0]$ ha previsione calcolata di $[0.09845785763965222, 0.015421380649956663, -0.5138068038427066, -0.4853190165287735, -0.22629262719814794, 0.0008152164571250502]$

Appaiono in relazione le domande 1, 2, 6 e 3, 4, 5.

Gli scostamenti tra le coppie 1, 4 e 2, 5 e 3, 6 per una differenza tuttavia trascurabile che oscilla dallo 0.2 allo 0.5. Le domande 3 e 6 si dovrebbero presentare con una positività inferiore rispetto a 1 e 4, la regola non vale per la coppia 6 e 4. Le anomalie riscontrate sono da ricondurre alla natura del vettore di training che si basa sul calcolo della probabilità di una risposta che sul grafo della conoscenza.

Osservazioni

Confrontando i risultati ottenuti dalla rete con i layers impostati a 4 neuroni con quanto emerso dai dati risultanti dalla rete a 2 neuroni emerge come la configurazione a 2 neuroni a layers è sicuramente quella che da i risultati attesi.

Quanto emerso di discordate dal secondo training set è come da aspettative da associare alla natura stessa della creazione del set di dati.

2.1.3 Configurazione della rete a 4 neuroni per 1 layer

Configurazione della rete utilizzata:

```
layer_defs = [];  
layer_defs.push({type:'input', out_sx:1, out_sy:1, out_depth:6});  
layer_defs.push({type:'fc', num_neurons:4, activation: 'tanh'});  
layer_defs.push({type:'regression', num_neurons:6});
```

```
net = new convnetjs.Net();  
net.makeLayers(layer_defs);
```

```
trainer = new convnetjs.SGDTrainer(net, {learning_rate:0.01,  
momentum:0.1, batch_size:10, l2_decay:0.001});
```

Viene utilizzato un unico layer da 4 neuroni.

Training set standard su rete a 4 neuroni per 1 layer

- Il vettore [0,0,0,0,0,0] ha previsione calcolata di
[0.12202628618565468,0.08221724740100582,0.02233631914718809,
0.09586625658118901,0.05558075220027264,0.13443779128784109]

Appaiono in relazione le domande 1, 2, 3, 4, 5, 6.

Gli scostamenti tra le coppie 1, 4 e 3, 6 e 2, 5 sono consistenti con quelle che sono le relazioni di dipendenza fra le domande. Le domande 3 e 6 si dovrebbero presentare con una positività inferiore rispetto

a 1 e 4; in questo test la regola non viene rispettata dalla domanda 6. Dall'immagine canvas e dai dati della previsione si nota come il candidato non ha una buona probabilità di saper rispondere alle domande e la domanda 6 non si presenta conforme alle aspettative.

Osservazioni

Rispetto a quanto osservato nei casi precedenti, ancora la configurazione che rispetta le attese è quella con 2 neuroni per 2 layers.

Tale conclusione ha perfettamente senso in quanto il grafo della conoscenza che ho usato come base per costruire i vettori di apprendimento è composto da 3 nodi (A, B, C) indicanti 3 neuroni; il quarto può venire valutato come un nodo della rete utile per parametri in entrata e in uscita.

Per estendere maggiormente la mia conoscenza della rete, ho provveduto ad aumentare progressivamente il numero di neuroni a layers e osservarne le interazioni. Svolgendo ciò mi sono accorta che il risultato ottenuto dalla previsione era il più possibile vicino al vettore previsione; conseguenza diretta di un numero eccessivo di neuroni dati alla rete per l'apprendimento rispetto al training set svolto, generatrice di una situazione di overfitting e non attendibilità dei dati raccolti. La configurazione a 1 e 2 neuroni invece presenta una buona capacità di previsione in quasi tutti i casi, però tende ad andare in overfitting, come riporto di seguito:

Il vettore [0,0,0,0,0,1] ha previsione calcolata di [0.5613347853884025,0.831067062

Il numero di neuroni non è sufficiente per memorizzare che la domanda 6 deve essere positiva, e comporta a cascata la correttezza anche delle domande 3, 4 e 1. La situazione si presenta simile se il layer con 1 neurone è posto al di sotto.

2.2 Risultati delle previsioni sulla rete del database

Durante la settimana dal 03/06 al 07/06 ho provveduto a montare e configurare la rete neurale che ha come dataset i colloqui ai candidati.

La configurazione che ho utilizzato, basandomi anche sul quanto appreso dalla rete neurale di prova e dal numero di vettori di test utilizzati (1245 vettori x 89) ho aumentato le dimensioni della rete.

```
layer_defs = [];  
layer_defs.push({type:'input', out_sx:1, out_sy:1, out_depth:89});  
layer_defs.push({type:'fc', num_neurons:8, activation: 'tanh'});  
layer_defs.push({type:'fc', num_neurons:4, activation: 'tanh'});  
layer_defs.push({type:'fc', num_neurons:2, activation: 'tanh'});
```

```
layer_defs.push({type:'regression', num_neurons:89});
```

```
net = new convnetjs.Net();  
net.makeLayers(layer_defs);
```

```
trainer = new convnetjs.SGDTrainer(net, {learning_rate:0.01, momentum:0.1, batch_size:100});
```

Ho aggiunto un layer e messo come potenze di 2 il numero di neuroni all'interno di ciascun layers. Devo verificare la bontà di questa mia scelta o se invece mi porta ad una situazione di overfitting.

Analizzando il training set dei vettori ho riscontrato tali correlazioni:

- Solo una piccola parte delle domande presenti in un database vengono svolte durante un colloquio con un candidato, in media una decina su 89 possibili;
- Dalla rete sembrano correlate in positivo le domande