

# SEMINARIO DEL 01/04

*Ruggieri Andrea*

*Stranieri Francesco*

*MAD Lab*

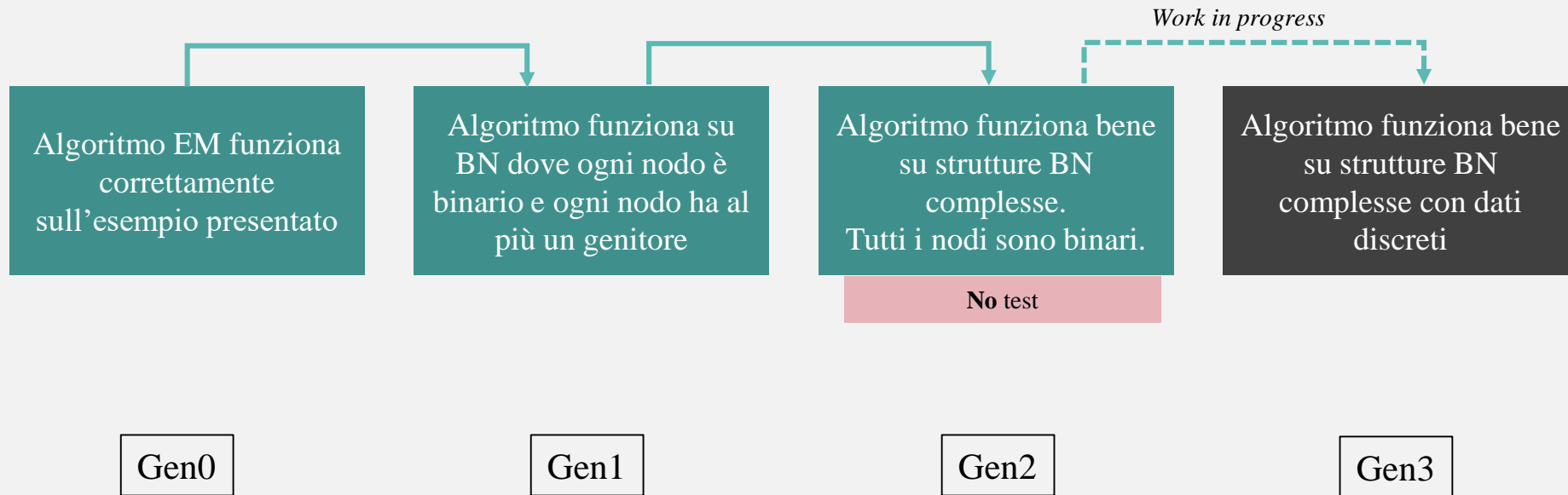


# Introduzione

*Review*

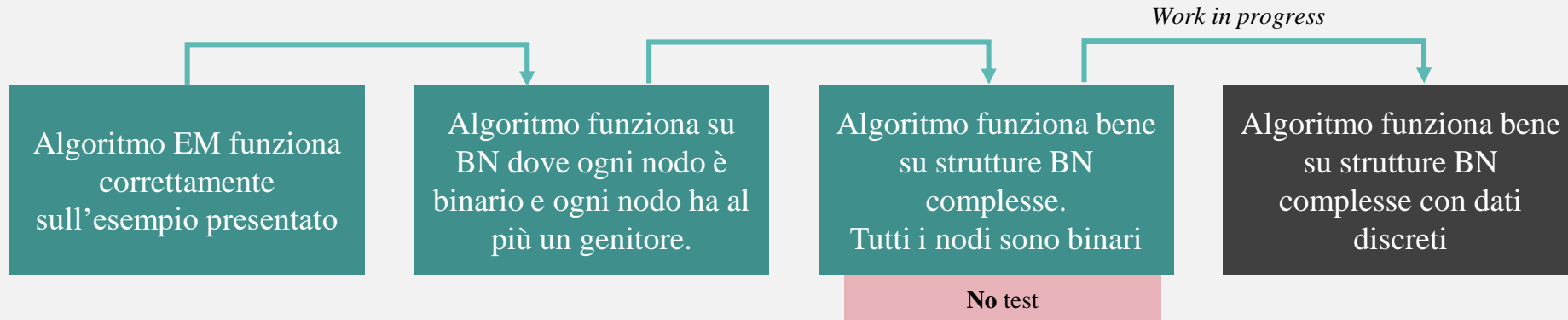
# EVOLUZIONE DEL PROGETTO

ULTIMO MEETING

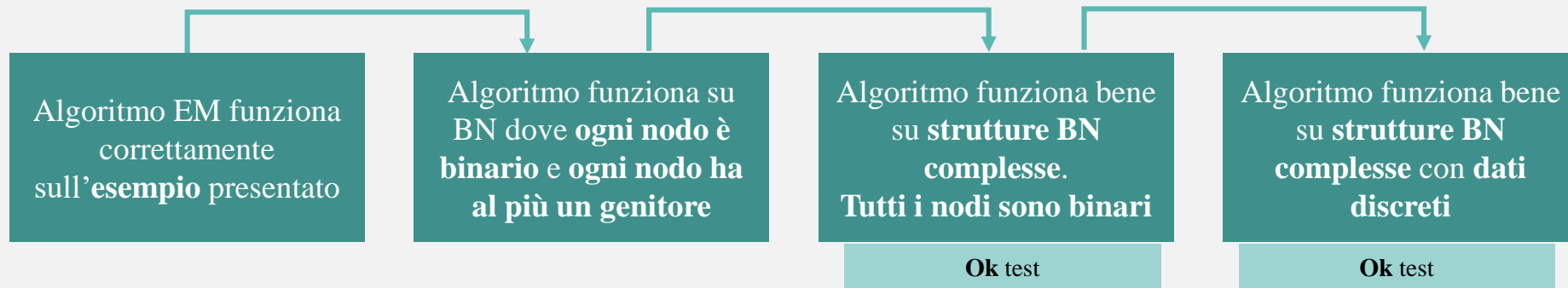


# EVOLUZIONE DEL PROGETTO

## ULTIMO MEETING



## ORA



**N.B.:** Un test ha riportato che valori inconsistenti sono ottenuti quando all'interno di un dato sono presenti tanti missing values. **DA CORREGGERE**

# TRE FASI

## FASE 1: TESTING

- Gen2 testata su 7 diversi tipi di dataset;
- Sono stati testati sia casi limite sia casi normali;
- Ogni test ha uno scopo ben preciso volte a valutare la correttezza di particolari **funzionalità**;
- Alla fine degli esperimenti sono state individuate **sette inconsistenze** che sono state risolte apportando modifiche direttamente sullo script.

## FASE 2: IMPLEMENTAZIONE GEN3 (alg. EM su dati discreti)

- **Idea chiave:** memorizzare informazioni riguardante i dati mancanti all'interno di una particolare **struttura** a tabella;
- Estendere i passi di **Expectation** e di **Maximisation** in modo da lavorare su dati discreti.

## FASE 3: TEST E MIGLIORAMENTI FINALI DEL CODICE

- Esecuzione dell'algoritmo su una struttura estremamente complessa (esempio reale: **PlayTennis**);
- Miglioramento e ottimizzazione del codice.

# Test Eseguiti

*Presentazione dei test*

*Analisi dei risultati*

# INTRODUZIONE

## *APPROCCIO METODOLOGICO*

I risultati sono stati confrontati con **EM manuale (GEN2)** ed eventualmente sono stati eseguiti **calcoli a mano**;

Due tipi di **confronti** sono stati effettuati:

- Con le CPT al termine del passo di Maximisation;
- Con i dati rimpiazzati.

Siamo quindi passati all'interpretazione dei risultati, cercando di individuare **inconsistenze**:

- Errori;
- Valori diversi rispetto a *bnlean*;
- Valori non attesi.

Per ogni inconsistenza trovata abbiamo approfondito sulla possibile causa:

- Debug step-by-step;
- Eseguendo calcoli manuali;
- Eseguendo altri esperimenti con dataset simili.

# TEST 1 – TANTI DATI

Dataset

A	0	0	M	0	1	0	M	1	1	0	1	0	0	1	1	1	0	M	0	0	0
B	1	0	0	M	0	1	1	0	1	M	1	1	0	0	M	1	M	1	1	1	0
C	0	M	1	0	0	0	1	M	M	1	1	0	1	1	1	M	1	0	1	1	0

Struttura della rete

[A] [B|A] [C|A.B]

Risultati ottenuti

	D2	D3	D4	D7	D8	D9	D10	D15	D16	D17	D18
GEN2	1	0	1	1	1	1	1	1	1	1	0
BNLE ARN	1	0	1	0	1	1	1	1	1	1	0

% valori rimpiazzati correttamente: 91%

Aspetti testati

Osservati tanti dati. Sono attesi poche probabilità vicine a 1.

Confronto con *bnlearn*.

Errori risolti

L'indicizzazione sulle CPT ottenuta attraverso il metodo *expand.grid* risultava essere invertita e faceva ottenere probabilità finali invertite. Errore trovato dopo aver eseguito debug manuale sull'esempio. **RISOLTO**

**In Maximisation:** il vettore *column\_cpt* contenente gli indici delle colonne delle CPT da aggiornare, veniva aggiornata ad ogni step in maniera errata. Errore trovato dopo aver eseguito a mano i calcoli e aver identificato l'inversione delle probabilità nelle CPT.

**RISOLTO**

**In Maximisation:** mancavano dei controlli da fare ai nodi parents qualora il nodo da computare consisteva di 2 o più genitori.

**RISOLTO**



# TEST 2 – RETE COMPLESSA

Dataset

A	0	NA	1	1	1
B	1	1	0	NA	0
C	0	0	1	0	0
D	NA	1	0	0	0
E	0	0	1	1	NA

Struttura della rete

[A] [B|A] [C|A] [D|C] [E|A:D:B]

Risultati ottenuti

	Data 1	Data 2	Data 4	Data5
GEN2	0	1	0	1
BNLEARN	0	1	0	1

% valori rimpiazzati correttamente: 100%

Aspetti testati

Osservati i pochi dati, alcune probabilità nelle CPT devono essere uguali a 1.

La CPT del nodo E deve essere aggiornata correttamente.

Confronto con *bnlearn*.

Errori risolti

**In Expectation:** risultati inconsistenti nel nodo E per il dato 5.

## RISOLTO

**Causa:** indicizzazione sbagliata delle CPT in caso di tanti genitori. Questo errore si manifestava solo in casi rari e con particolarissimi condizioni. Il codice di Expectation è stato riscritto in larga percentuale per correggere questo errore.

**In Maximisation:** errore se il numero di esempi presenti all'interno del Dataset è troppo basso (problema divisione per 0). RISOLTO

# TEST 4 – CASO LIMITE #1

## Dataset

<b>A</b>	1	NA	1	NA	1
<b>B</b>	1	1	1	1	NA
<b>C</b>	NA	1	1	1	1

## Struttura della rete

[A] [B|A] [C|B]

## Risultati ottenuti

	Data 1	Data 2	Data 4	Data5
<b>GEN2</b>	1	1	1	1
<b>BNLEARN</b>	1	1	1	1

% valori rimpiazzati correttamente: 100%

## Aspetti testati

Tutti i valori del dataset osservati sono pari a 1.

Ci si aspetta che tutti i valori saranno rimpiazzati con 1.

Confronto con *bnlearn*.

## Errori risolti

Nessun errore è stato riscontrato.

# TEST 5 – CASO LIMITE #2

## Dataset

<b>A</b>	0	1	0	0	1	1	0
<b>B</b>	NA	NA	1	NA	NA	NA	NA
<b>C</b>	1	0	0	0	0	0	1

## Struttura della rete

[A] [B|A] [C|B]

## Risultati ottenuti

**Validazione manuale:** allo step numero 1 i risultati sono uguali ai calcoli.

	Data 1	Data 2	Data 4	Data 5	Data 6	Data 7
<b>GEN2</b>	1	0	1	0	0	1
<b>BNLEARN</b>	1	1	1	1	1	1

% valori rimpiazzati correttamente: 50%

## Aspetti testati

Il nodo B prevede la presenza di valori Missing.

Confronto con *bnlearn*.

## Errori risolti

**Risultato inconsistente con *bnlearn*:** calcoli eseguiti a mano e riportati nella slide successiva per valutare la correttezza di gen2.

# TEST 5 – CASO LIMITE #2

TEST 5

A	0	1	0	0	1	1	0
B	NA	NA	1	NA	NA	NA	NA
C	1	0	0	0	0	0	1

Passo 1 - Expectation

Data 1, Data 7

$$P(B=0|\theta) = \alpha \cdot 0,5 \cdot 0,3 \cdot 0,7 = \alpha \cdot 0,105 = 0,46$$

$$P(B=1|\theta) = \alpha \cdot 0,5 \cdot 0,6 \cdot 0,3 = \alpha \cdot 0,09 = 0,53$$

Data 2, Data 5, Data 6

$$P(B=0|\theta) = \alpha \cdot 0,5 \cdot 0,6 \cdot 0,3 = \alpha \cdot 0,09 = 0,5$$

$$P(B=1|\theta) = \alpha \cdot 0,5 \cdot 0,3 \cdot 0,6 = \alpha \cdot 0,09 = 0,5$$

Data 4

$$P(B=0|\theta) = \alpha \cdot 0,5 \cdot 0,3 \cdot 0,3 = \alpha \cdot 0,045 = 0,2$$

$$P(B=1|\theta) = \alpha \cdot 0,5 \cdot 0,6 \cdot 0,3 = \alpha \cdot 0,09 = 0,8$$

Passo 1 - Maximisation

CPT A = 0  $\rightarrow 1+1+1+1/7 = 0,5714$   $\rightarrow$  nuova max

CPT A = 1  $\rightarrow 1+1+1+1/7 = 0,4285$

CPT B  $\rightarrow P(B=0|A=0) = 0,283$   $P(B=0|A=1) = 0,5$

CPT C  $\rightarrow P(C=0|B=0) = 0,64$   $P(C=0|B=1) = 0,755$

Passo 2 - Expectation

Data 1, Data 7

$$P(B=0|\theta) = \alpha \cdot 0,5714 \cdot 0,283 \cdot 0,3544 = \alpha \cdot 0,05714$$

$$P(B=1|\theta) = \alpha \cdot 0,5714 \cdot 0,716 \cdot 0,244 = \alpha \cdot 0,05714$$

Passo 2 - Expectation

Data 1, Data 7

$$P(B=0|\theta) = \alpha \cdot 0,5714 \cdot 0,283 \cdot 0,3544 = \alpha \cdot 0,05714$$

$$P(B=1|\theta) = \alpha \cdot 0,5714 \cdot 0,716 \cdot 0,244 = \alpha \cdot 0,05714$$

Data 2, Data 5, Data 6

$$P(B=0|\theta) = \alpha \cdot 0,4285 \cdot 0,5 \cdot 0,6455 = \alpha \cdot 0,14285$$

$$P(B=1|\theta) = \alpha \cdot 0,4285 \cdot 0,5 \cdot 0,755 = \alpha \cdot 0,15714$$

Data 4

$$P(B=0|\theta) = \alpha \cdot 0,5714 \cdot 0,283 \cdot 0,6455 = \alpha \cdot 0,25246$$

$$P(B=1|\theta) = \alpha \cdot 0,5714 \cdot 0,716 \cdot 0,7557 = \alpha \cdot 0,74754$$

[CPT validate in un foglio a parte (MAXIMISATION)]

Passo 3 - Expectation

Data 1, 7

$$P(B=0|\theta) = \alpha \cdot 0,5714 \cdot 0,2453 \cdot 0,308 = \alpha \cdot 0,2673$$

$$P(B=1|\theta) = \alpha \cdot 0,5714 \cdot 0,7546 \cdot 0,274 = \alpha \cdot 0,7327$$

Data 2, 5, 6

$$P(B=0|\theta) = \alpha \cdot 0,4285 \cdot 0,4606 \cdot 0,6915 = \alpha \cdot 0,4486$$

$$P(B=1|\theta) = \alpha \cdot 0,4285 \cdot 0,5393 \cdot 0,7258 = \alpha \cdot 0,5513$$

Data 4

$$P(B=0|\theta) = \alpha \cdot 0,5714 \cdot 0,2453 \cdot 0,6915 = \alpha \cdot 0,2365$$

$$P(B=1|\theta) = \alpha \cdot 0,5714 \cdot 0,7546 \cdot 0,7258 = \alpha \cdot 0,7635$$

Passo 4 - Expectation

~~Data~~ Nou COMPUTO PIU Data 1, 4, 7

Data 2, 5, 6

$$P(B=0|\theta) = \alpha \cdot 0,5714 \cdot 0,1830 \cdot 0,747 = \alpha \cdot 0,4648$$

$$P(B=1|\theta) = \alpha \cdot 0,5714 \cdot 0,8069 \cdot 0,7 = \alpha \cdot 0,5351$$

Passo 5 - Expectation

Data 2, 5, 6

$$P(B=0|\theta) = \alpha \cdot 0,5714 \cdot 0,134 \cdot 0,826 = \alpha \cdot 0,5166$$

$$P(B=1|\theta) = \alpha \cdot 0,5714 \cdot 0,865 \cdot 0,671 = \alpha \cdot 0,4833$$

Andando avanti per altre iterazioni

$P(B=0|\theta) > P(B=1|\theta)$  SEMPRE

Final Data

A	0	1	0	0	1	1	0
B	1	0	1	1	0	0	1
C	1	0	0	0	0	0	1

**N.B.:** Andando avanti in iterazioni successivi  $P(B=0|\theta)$  tende sempre a essere vicino a 0.5.

La validazione manuale conferma che i risultati riportati da GEN2 sono corretti. Tuttavia, è possibile notare che le probabilità calcolate in corrispondenza dei dati 2,5 e 6 sono molto vicini allo 0.5 e un metodo di inferenza approssimata come EM *bnlearn* potrebbe sbagliare.

# TEST 6 – MISSING VALUES RARI

## Dataset

A	0	0	1	1	0	0	1	1
B	1	1	0	1	0	0	0	1
C	1	0	NA	0	1	0	NA	1

## Struttura della rete

[A] [B|A] [C|B:A]

## Risultati ottenuti

**Validazione manuale:** allo step numero 1 i risultati sono uguali ai calcoli.

	Data 3	Data 7
GEN2	1	1
BNLEARN	1	1

% valori rimpiazzati correttamente: 100%

## Aspetti testati

I due missing values dovrebbero assumere lo stesso valore.

Dati i pochi dati mancanti, l'algoritmo si dovrebbe arrestare alle prime iterazioni.

CPT di A e B dovrebbero rimanere invariate e non cambiare mai.

## Modalità di testing

Validazione attraverso calcoli manuali.

Confronto con *bnlearn*.

## Errori risolti

Tutti i requisiti soddisfatti e calcoli corretti dopo la validazione manuale.

# TEST 7 – RETE PARTICOLARE

Dataset

A	0	1	0	NA
B	1	NA	1	1
C	0	0	1	1
D	1	0	0	1
E	1	0	1	1
F	NA	1	1	0

Struttura della rete

[A] [B] [C] [D] [E|A:B:C:D][F|A:D:E]

Risultati ottenuti

**Validazione manuale:** allo step numero 1 i risultati sono uguali ai calcoli

	Data 1	Data 2	Data 4
GEN2	0	1	1
BNLEARN	0	1	0

% valori rimpiazzati correttamente: 66%

Aspetti testati

Considerato i pochi dati, ci si aspetta che un numero limitato di probabilità di CPT varia.

Dati i pochi dati mancanti, l'algoritmo si dovrebbe arrestare alle prime iterazioni.

Modalità di testing

Validazione attraverso calcoli manuali.

Confronto con *bnlearn*.

Errori risolti

**Risultato inconsistente con bnlearn:** validazione fatta a mano step by step attraverso esecuzione del codice in modalità debug.

# TEST 8 – RISULTATI UGUALI

## Dataset

<b>A</b>	0	1	NA	1	0
<b>B</b>	1	0	1	NA	1
<b>C</b>	NA	0	0	0	1

## Struttura della rete

[A] [B|A] [C|B]

## Risultati ottenuti

Confronto tra i risultati ottenuti tra le varie generazioni.

	Data 1	Data 3	Data 4
<b>GEN1</b>	0	0	0
<b>GEN2</b>	0	0	0
<b>GEN3</b>	0	0	0

% valori rimpiazzati correttamente: 100%

## Aspetti testati

Risultati (dati rimpiazzati e CPT) devono essere uguali per gen1, gen2 e gen3.

Dopo aver effettuato calcoli a mano ci si aspetta che tutti i dati mancanti vengano rimpiazzati con 0.

## Errori risolti

Nessun errore è stato riscontrato.

# Gen3

(alg. EM su dati discreti)

*Presentazione Gen3*



# EM MANUALE – GEN3

Si tratta di un'estensione di GEN2 e ha l'obiettivo di permettere l'esecuzione dell'algoritmo EM su dati discreti.

GEN3 aggiunge a GEN2 tre strutture dati:

1. **table\_posterior\_prob**: struttura in grado di memorizzare tutte le probabilità a posteriori dei dati mancanti. Questa struttura viene aggiornata nella fase di Expectation e utilizzata nella fase di Maximization;
2. **matrix\_to\_compute**: struttura in grado di memorizzare ogni combinazione di variabili e di indicizzare queste combinazioni con le CPT;
3. **compute\_probabilities**: questa matrice è collegata direttamente a `matrix_to_compute` (stesso numero di riga e stesso indice). Questa struttura è presente sia in Expectation sia in Maximisation ma con due scopi diversi:
  - in Expectation, memorizza le informazioni parziali delle probabilità a posteriori;
  - in Maximisation, memorizza i calcoli parziali da salvare successivamente nelle CPT.

# EM MANUALE – GEN3

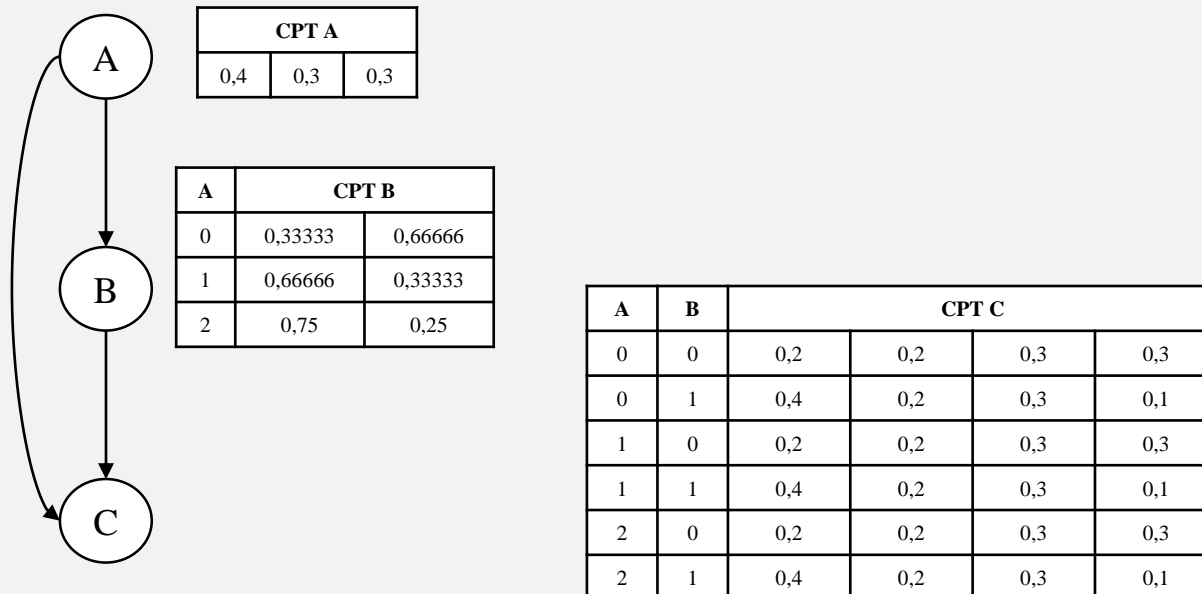
## ESEMPIO PRATICO

Sia fissato il seguente dataset:

	Data_1	Data_2	Data_3	Data_4	Data_5	Data_6
A	1	0	2	1	NA	0
B	1	0	NA	1	0	1
C	0	3	2	NA	1	3

$A = \{0,1,2\}$   
 $B = \{0,1\}$   
 $C = \{0,1,2,3\}$

Sia fissata anche la seguente rete:



# EM MANUALE – GEN3

## ESEMPIO PRATICO

Si definisce ora la prima tabella: `table_posterior_prob`

Node	Value	Data_3	Data_4	Data_5
A	0	0	0	NA
A	1	0	0	NA
A	2	0	0	NA
B	0	NA	0	0
B	1	NA	0	0
C	0	0	NA	0
C	1	0	NA	0
C	2	0	NA	0
C	3	0	NA	0

Lista dei dati in cui è presente almeno un missing value.

Cella che conterrà la probabilità a posteriori:  $P(A = 2 \mid \text{param})$  calcolata attraverso inferenza esatta.

Dato osservato.

**Regola testata:** la somma di tutte le probabilità deve sempre essere uguale a 1.

# EM MANUALE – GEN3

## ESEMPIO PRATICO

Seconda struttura: **matrix\_to\_compute**

Punto di partenza: CPT

A	B	CPT C			
0	0	0,2	0,2	0,3	0,3
0	1	0,4	0,2	0,3	0,1
1	0	0,2	0,2	0,3	0,3
1	1	0,4	0,2	0,3	0,1
2	0	0,2	0,2	0,3	0,3
2	1	0,4	0,2	0,3	0,1

**matrix\_to\_compute**

ID_ROW	A	B
1	0	0
2	0	1
3	1	0
4	1	1
5	2	0
6	2	1

**matrix\_to\_compute** ha lo scopo di replicare la parte più a sinistra della CPT in modo da memorizzare ogni possibile combinazione delle variabili coinvolte.

Tale tabella faciliterà l'**indicizzazione** con le CPT, superando alcuni problemi sorti nell'implementazione di GEN2.

# EM MANUALE – GEN3

## ESEMPIO PRATICO

Seconda struttura: **compute\_probabilities**

Punto di partenza: CPT

A	B	CPT C			
0	0	0,2	0,2	0,3	0,3
0	1	0,4	0,2	0,3	0,1
1	0	0,5	0,2	0,1	0,2
1	1	0,25	0,1	0,4	0,25
2	0	0,3	0,1	0,2	0,4
2	1	0,35	0,15	0,25	0,25

**compute\_probabilities**

ID_ROW	0	1	2	3
1	0,2	0,2	0,3	0,3
2	0,4	0,2	0,3	0,1
3	0,5	0,2	0,1	0,2
4	0,25	0,1	0,4	0,25
5	0,3	0,1	0,2	0,4
6	0,35	0,15	0,25	0,25

**compute\_probabilities** ha lo scopo di replicare la parte più a sinistra della CPT in modo da contenere ogni valore che poi verrà usato per aggiornare le CPT.

Particolare attenzione deve essere posta sulla chiave **ID\_ROW** il cui valore si mappa direttamente sulla matrice **matrix\_to\_compute**.

**ATTENZIONE:** la spiegazione appena riportata di `compute_probabilities` è solo la base del ragionamento applicato per implementare la GEN3 e superare problemi sorti in GEN2.

Nella realtà questa tabella, nello step di Maximisation è stata estesa leggermente per contenere un altro tipo di informazione, ossia la colonna **denominatore** che servirà come controllo del fatto che tutta la procedura implementata sia corretta.

# EM MANUALE – GEN3

## ESEMPIO PRATICO

Note aggiuntive sull'implementazione:

- Il join tra `matrix_to_compute` e `compute_probabilities`, forma la struttura tabellare di una CPT come mostrato nella slide precedente.
- Una colonna di una CPT in *bnlearn* equivale a una riga di `compute_probabilities`

Conditional probability table:

A		0	1	2
B				
0	0.333333	0.666667	0.750000	0.250000
1	0.666667	0.333333	0.250000	0.750000

	0	1
1	0,33333	0,66666
2	0,66666	0,33333
3	0,75	0,25

- Le due informazioni sono state separate in due strutture particolari per tre motivi:
  1. **Efficienza e accesso alle informazioni:** ci sono delle situazioni in cui è necessario conoscere solo un'informazione e definire un'unica tabella può essere inefficiente;
  2. **Testing:** avere due strutture logicamente separate permette di testare in maniera più efficiente se ciò che è stato implementato risulta essere corretto;
  3. **Maggiore facilità nell'accesso alle CPT** utilizzando solo `compute_probabilities`.

# GEN3 - ITERAZIONE 1

## STEP 1- Expectation

Node	Value	Data_3	Data_4	Data_5
A	0	0	0	0,299
A	1	0	0	0,4485
A	2	0	0	0,2523
B	0	0,7058	0	0
B	1	0,2941	0	0
C	0	0	0,25	0
C	1	0	0,1	0
C	2	0	0,4	0
C	3	0	0,25	0

## STEP 2- Updating

	Data_1	Data_2	Data_3	Data_4	Data_5	Data_6
A	1	0	2	1	1	0
B	1	0	0	1	0	1
C	0	3	2	2	1	3

## STEP 3- Maximisation

```

$A
A      [,1]
0 0.3831776
1 0.4080997
2 0.2087227

$B
A      0      1      2
B 0 0.5650407 0.1832061 0.7651449
  1 0.4349593 0.8167939 0.2348551

$C
, , A = 0
      B
C      0 1
0 0.0000000 0
1 0.2302158 0
2 0.0000000 0
3 0.7697842 1

, , A = 1
      B
C      0 1
0 0 0.625
1 1 0.050
2 0 0.200
3 0 0.125

, , A = 2
      B
C      0 1
0 0.0000000 0
1 0.2633391 0
2 0.7366609 1
3 0.0000000 0
    
```

# Risultati Finali

## e miglioramento del codice

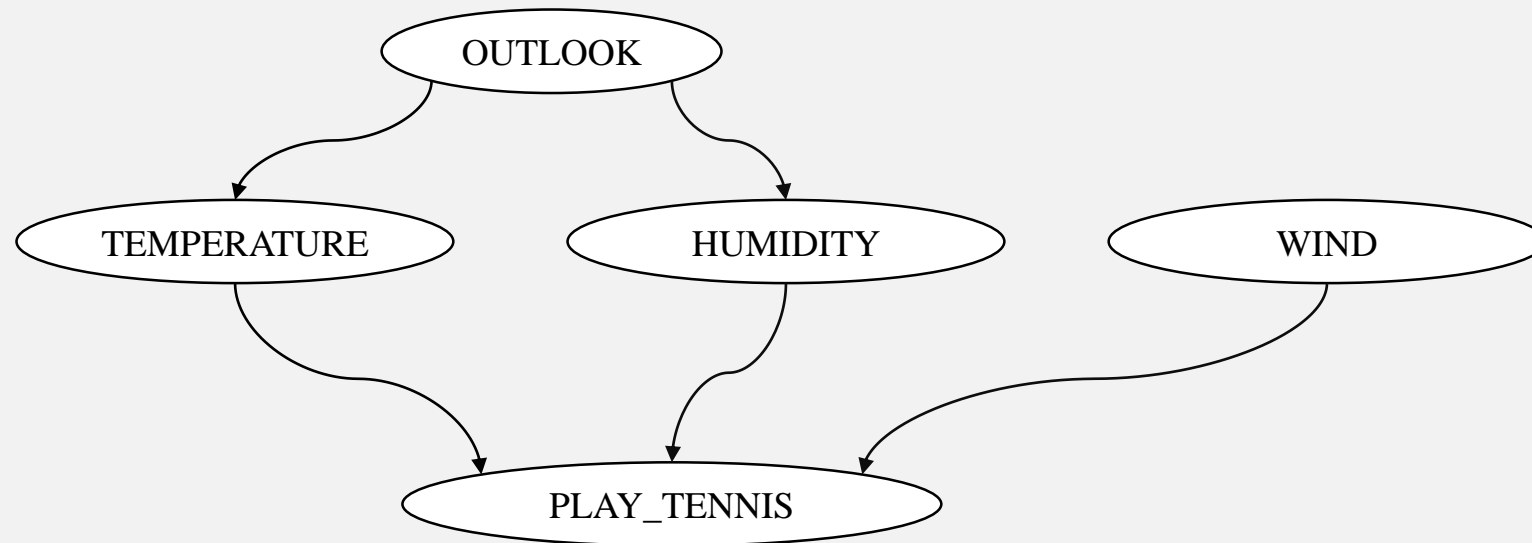
*Presentazione risultati GEN3*



# EM MANUALE – GEN3

L'esempio visto precedentemente è solo introduttivo sul funzionamento di EM manuale GEN3, vediamo ora un esempio più complesso prendendo in input il seguente dataset.

	Data_1	Data_2	Data_3	Data_4	Data_5	Data_6	Data_7	Data_8	Data_9	Data_10	Data_11	Data_12	Data_13	Data_14
OUTLOOK	0	0	NA	2	2	2	1	0	0	2	0	1	NA	2
TEMPERATURE	2	2	2	1	0	0	0	1	NA	1	1	1	2	1
HUMIDITY	2	2	2	NA	1	1	1	2	1	1	1	2	1	2
WIND	0	1	0	0	0	1	1	0	0	0	NA	1	0	1
PLAY_TENNIS	0	0	1	1	1	0	1	0	1	1	1	1	1	0



OUTLOOK = {0,1,2}  
TEMPERATURE = {0,1,2}  
HUMIDITY = {0,1,2}  
WIND = {0,1}  
PLAY\_TENNIS = {0,1}

# GEN3 - ITERAZIONE 1 (1)

## Expectation

Node	Value	Data_3	Data_4	Data_9	Data_11	Data_13
HUMIDITY	0	0	0.378378378378378	0	0	0
HUMIDITY	1	0	0.405405405405405	0	0	0
HUMIDITY	2	0	0.216216216216216	0	0	0
OUTLOOK	0	0.484793973804036	0	0	0	0.400529456842301
OUTLOOK	1	0.283321153521839	0	0	0	0.312100875461533
OUTLOOK	2	0.231884872674124	0	0	0	0.287369667696166
PLAY_TENNIS	0	0	0	0	0	0
PLAY_TENNIS	1	0	0	0	0	0
TEMPERATURE	0	0	0	0.177777777777778	0	0
TEMPERATURE	1	0	0	0.277777777777778	0	0
TEMPERATURE	2	0	0	0.544444444444444	0	0
WIND	0	0	0	0	0.695652173913044	0
WIND	1	0	0	0	0.304347826086957	0

## Updating

	Data_1	Data_2	Data_3	Data_4	Data_5	Data_6	Data_7	Data_8	Data_9	Data_10	Data_11	Data_12	Data_13	Data_14
OUTLOOK	0	0	0	2	2	2	1	0	0	2	0	1	0	2
TEMPERATURE	2	2	2	1	0	0	0	1	2	1	1	1	2	1
HUMIDITY	2	2	2	1	1	1	1	2	1	1	1	2	1	2
WIND	0	1	0	0	0	1	1	0	0	0	0	1	0	1
PLAY_TENNIS	0	0	1	1	1	0	1	0	1	1	1	1	1	0

# GEN3 - ITERAZIONE 1 (2)

## Maximisation

Parameters of node HUMIDITY (multinomial distribution)

Conditional probability table:

HUMIDITY	OUTLOOK		
	0	1	2
0	0.00000000	0.00000000	0.06855607
1	0.40788403	0.50554432	0.66907135
2	0.59211597	0.49445568	0.26237259

Parameters of node OUTLOOK (multinomial distribution)

Conditional probability table:

OUTLOOK	A
0	0.4203802
1	0.1853873
2	0.3942325

Parameters of node TEMPERATURE (multinomial distribution)

Conditional probability table:

TEMPERATURE	OUTLOOK		
	0	1	2
0	0.03020697	0.38529379	0.36236778
1	0.38702678	0.38529379	0.54355167
2	0.58276625	0.22941241	0.09408056

Parameters of node WIND (multinomial distribution)

Conditional probability table:

WIND	A
0	0.621118
1	0.378882

Parameters of node PLAY\_TENNIS (multinomial distribution)

Conditional probability table:

, , TEMPERATURE = 0, WIND = 0			
PLAY_TENNIS	HUMIDITY		
	0	1	2
0	0.6000000	0.0000000	0.4000000
1	0.4000000	1.0000000	0.6000000
, , TEMPERATURE = 1, WIND = 0			
PLAY_TENNIS	HUMIDITY		
	0	1	2
0	0.0000000	0.0000000	0.8222222
1	1.0000000	1.0000000	0.1777778
, , TEMPERATURE = 2, WIND = 0			
PLAY_TENNIS	HUMIDITY		
	0	1	2
0	0.5000000	0.0000000	0.5000000
1	0.5000000	1.0000000	0.5000000
, , TEMPERATURE = 0, WIND = 1			
PLAY_TENNIS	HUMIDITY		
	0	1	2
0	0.5500000	0.5000000	0.6000000
1	0.4500000	0.5000000	0.4000000
, , TEMPERATURE = 1, WIND = 1			
PLAY_TENNIS	HUMIDITY		
	0	1	2
0	0.3500000	0.0000000	0.5000000
1	0.6500000	1.0000000	0.5000000
, , TEMPERATURE = 2, WIND = 1			
PLAY_TENNIS	HUMIDITY		
	0	1	2
0	0.5000000	0.2500000	1.0000000
1	0.5000000	0.7500000	0.0000000

# GEN3 - ITERAZIONE 2

## Expectation

Node	Value	Data_3	Data_4	Data_9	Data_11	Data_13
HUMIDITY	0	0	0.0874136981673394	0	0	0
HUMIDITY	1	0	0.85311197490565	0	0	0
HUMIDITY	2	0	0.0594743269270103	0	0	0
OUTLOOK	0	0.825044278652587	0	0	0	0.683287343301183
OUTLOOK	1	0.119607392114927	0	0	0	0.147023270838318
OUTLOOK	2	0.0553483292324859	0	0	0	0.169689385860499
PLAY_TENNIS	0	0	0	0	0	0
PLAY_TENNIS	1	0	0	0	0	0
TEMPERATURE	0	0	0	0.0302069682104547	0	0
TEMPERATURE	1	0	0	0.38702678019645	0	0
TEMPERATURE	2	0	0	0.582766251593095	0	0
WIND	0	0	0	0	0.62111801242236	0
WIND	1	0	0	0	0.37888198757764	0

## Updating

	Data_1	Data_2	Data_3	Data_4	Data_5	Data_6	Data_7	Data_8	Data_9	Data_10	Data_11	Data_12	Data_13	Data_14
OUTLOOK	0	0	0	2	2	2	1	0	0	2	0	1	0	2
TEMPERATURE	2	2	2	1	0	0	0	1	2	1	1	1	2	1
HUMIDITY	2	2	2	1	1	1	1	2	1	1	1	2	1	2
WIND	0	1	0	0	0	1	1	0	0	0	0	1	0	1
PLAY_TENNIS	0	0	1	1	1	0	1	0	1	1	1	1	1	0

# GEN3 – RISULTATO FINALE

Eseguiamo tutte le iterazioni successive fino ad arrivare alla fine dello script:

- il programma termina alla 12-esima iterazione fissando il parametro  $\alpha = 0.001$ ;
- i dati finali risultano essere i seguenti:

	Data_1	Data_2	Data_3	Data_4	Data_5	Data_6	Data_7	Data_8	Data_9	Data_10	Data_11	Data_12	Data_13	Data_14
OUTLOOK	0	0	0	2	2	2	1	0	0	2	0	1	0	2
TEMPERATURE	2	2	2	1	0	0	0	1	2	1	1	1	2	1
HUMIDITY	2	2	2	1	1	1	1	2	1	1	1	2	1	2
WIND	0	1	0	0	0	1	1	0	0	0	0	1	0	1
PLAY_TENNIS	0	0	1	1	1	0	1	0	1	1	1	1	1	0

- I dati cerchiati, inizialmente cerchiati sono stati rimpiazzati.

# EM MANUALE – GEN3:

## MIGLIORAMENTI

Il metodo dello *stopping criteria* è stato riscritto da zero in modo da essere maggiormente **comprensibile** e **adatto** per qualunque tipo di rete bayesiana discreta.

```
#works very well on every BN structures
stopping_criteria <- function(alpha) {
  delta = 0
  for (node in names(bn)){
    for (value in 1:length(cpt[[node]])){
      differenza = abs(cpt[[node]][value] - cpt_last[[node]][value])
      delta = delta + differenza
    }
  }

  if (delta > alpha){
    return(FALSE)
  }else{
    return(TRUE)
  }
}
```

# Limiti e Sviluppi Futuri

*Limiti dello script*

# EM MANUALE – LIMITI

Lo script soffre ancora di alcuni problemi:

- In alcuni casi i risultati sono inconsistenti se all'interno di un dato sono presenti più valori mancanti;
- Contenuti online limitati per validare l'algoritmo anche con strutture esistenti.