



DIPARTIMENTO
DI INFORMATICA

**SISTEMA INTELLIGENTE PER L'ORIENTAMENTO
AGLI STUDI DEL CDL IN INFORMATICA**

DOCENTE: Nicola Fanizzi

CORSO: Ingegneria della Conoscenza (ICon)

ANNO ACCADEMICO DI FREQUENZA: 2020-2021

REALIZZATO DA:

GABRIELE PANTALEO – **Matricola:** 708237 – **E-mail:** g.pantaleo11@studenti.uniba.it

NICOLO' PRODON – **Matricola:** 698255 – **E-mail:** n.prodon@studenti.uniba.it

URL REPOSITORY: https://github.com/NicoloProdon/CdL_Affinity

Sommario

1. Obiettivo progettuale.....	3
2. Sistema esperto	
2.1 Introduzione.....	3
2.2 Implementazione.....	4
2.3 Schema di funzionamento del Sistema Esperto.....	6
3. Rete Bayesiana	
3.1 Introduzione.....	7
3.2 Implementazione.....	8
3.3 Schema di funzionamento della Rete Bayesiana.....	10
4. Esempi e valutazioni	
4.1 Rete bayesiana ideale vs Rete bayesiana con MLE	11
5. Librerie.....	14
6. Conclusioni.....	14

1. Obiettivo progettuale

Il sistema realizzato è in grado di determinare la compatibilità di uno studente (e non) con il CdL in Informatica grazie a una sequenza di domande, la cui risposta dovrà essere positiva ('s') o negativa ('n'). In base alle risposte, verrà determinato se lo studente avrà affinità inesistente, mediamente bassa, mediamente alta o molto alta con il profilo da studente universitario informatico. Il sistema di orientamento interroga lo studente, ponendogli una serie di domande per determinare se quest'ultimo è portato per lo studio dell'informatica in ambito universitario: si parla dunque di *forward chaining*.

2. Sistema esperto

2.1 Introduzione

L'idea alla base del sistema è una regola d'inferenza chiamata *modus ponens* per la quale, sia h la "testa" dell'atomo e $a_1 \wedge \dots \wedge a_m$ il "corpo" della clausola formato da a_i atomi, se $h \leftarrow a_1 \wedge \dots \wedge a_m$ è una clausola definita nella base di conoscenza e ogni a_i è stato derivato, allora h può essere derivato.

Se $m > 0$, la clausola è detta regola; se $m = 0$, il corpo è vuoto e la clausola è detta clausola atomica (o fatto), e tutte le clausole atomiche nella base di conoscenza sono sempre derivate in maniera diretta.

Un sistema esperto è un'applicazione dell'intelligenza artificiale atta alla risoluzione di determinati problemi, cercando di simulare i comportamenti di persone esperte in un determinato dominio. È principalmente composto da:

- una "knowledge base", che rappresenta e memorizza fatti e regole riguardanti il mondo;
- un "inference engine", che si occupa di mettere in pratica le nozioni apprese dalla base di conoscenza;
- una "user interface", che permette una facile interazione tra il sistema e l'utente.

2.2 Implementazione

Il sistema si basa su un sistema esperto realizzato in Python, principalmente con la libreria **Experta**, che permette di associare fatti accaduti a regole relative agli stessi.

Le regole sono formate da due componenti:

- **LHS** (Left-Hand-Side): è l'insieme delle condizioni che si devono verificare affinché la regola venga applicata;
- **RHS** (Right-Hand-Side): è l'insieme di azioni che vengono compiute quando viene applicata la regola.

L'interesse dell'intervistato dipende da una serie di fattori ed è diviso in quattro categorie, quali:

1. Assenza di interesse;

2. Minimo interesse (almeno un fattore):

- Importanza della tecnologia nella vita quotidiana;
- Importanza dell'informatica nel mondo del lavoro;

3. Medio Interesse (include minimo interesse e almeno un fattore):

- Interesse per le materie tecnologico-scientifiche dell'area STEM;
- Interesse per la matematica;
- Aver frequentato un istituto di secondario di settore Informatica;

4. Massimo Interesse (include minimo e medio interesse):

- Se diplomato in informatica, gli argomenti studiati devono essere obbligatoriamente graditi;
- Nel caso di istruzione secondaria diversa dall'istituto tecnico, all'intervistato devono piacere almeno due categorie di materie tra quelle proposte nel sondaggio.

```
# minimum interest questions
@Rule(Fact(question=True))
def ask_min(self):
    self.declare(Fact(life=valid_answer("\n1) Pensi che la tecnologia stia influenzando tanto la tua vita?"))
    self.declare(Fact(work=valid_answer("\n2) L'informatica e' per te importante nel mondo del lavoro?"))
```

Figura 1: Esempio di regola (Rule) e fatto (Fact) all'interno del codice.

Ogni domanda posta all'intervistato, alla quale è associata una **regola**, ha il fine di rilevare in quest'ultimo l'interesse verso uno specifico argomento e, in base alla sua risposta (positiva o negativa), la presenza dell'interesse per tale argomento viene impostata a **True** o **False**; dopodiché, il sistema interpreterà il profilo dell'utente e di conseguenza applicherà altre regole relative ad altri argomenti d'interesse. In alcune occasioni, il sistema porrà all'utente domande diverse in base al tipo di risposte fornite da quest'ultimo.

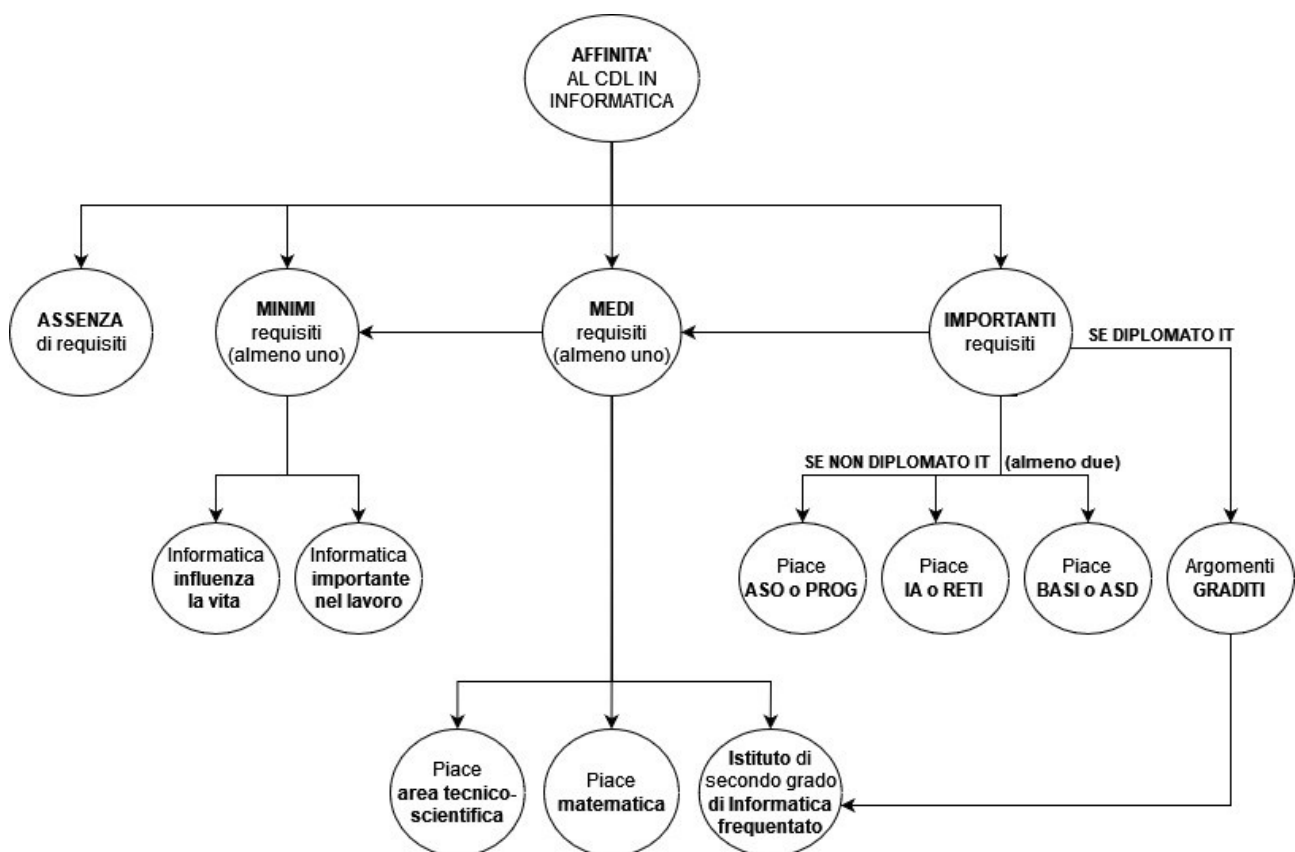
1. Le prime due domande poste all'utente sono di carattere generale:
 - a. se l'utente risponde positivamente ad una delle due domande (o ad entrambe), vorrà dire che l'utente ha un interesse minimo per l'informatica;
 - b. nel caso in cui l'utente risponda negativamente ad entrambe, il programma terminerà.
2. Verificata la presenza di un interesse minimo nell'intervistato, il sistema pone tre domande, leggermente più specifiche, per raggiungere un potenziale livello di interesse maggiore; le prime due domande serviranno a capire se l'utente ha interesse per le materie di tipo scientifico:
 - a. se l'utente risponderà positivamente solo ad una delle prime due domande delle tre che gli verranno poste, l'interesse aumenterà lievemente;
 - b. se l'utente risponderà positivamente alle prime due domande delle tre, l'interesse aumenterà moderatamente;
3. La terza domanda riguarda la formazione dell'utente:
 - a. nel caso in cui l'utente abbia frequentato un istituto tecnico, gli verrà chiesto se ha gradito gli argomenti già precedentemente affrontati:
 - se l'utente ha gradito gli argomenti trattati, l'interesse sarà alto;
 - se l'utente non ha gradito gli argomenti trattati, l'interesse sarà basso;
 - b. nel caso in cui l'utente NON abbia frequentato un istituto tecnico, gli verrà chiesto se conosce gli argomenti cardine che andrà ad affrontare (programmazione e architettura degli elaboratori, reti e intelligenza artificiale, basi di dati, algoritmi e strutture dati):
 - se ha trovato interessanti più di due argomenti, l'interesse è alto;
 - se ha trovato interessante un solo argomento, l'interesse sarà medio;
 - se non ha interesse in nessun argomento, l'interesse sarà basso.
4. Qualora l'utente rispondesse negativamente a tutte e tre le domande per calcolare l'interesse medio, il programma terminerà, poiché l'utente risulterà chiaramente non affine al CdL.

```
# high interest questions
@Rule(AND(Fact(mid_interest=True), Fact(school=True)))
def ask0_high(self):
    self.declare(Fact(feedback0=valid_answer("\n6) Ti sono piaciuti gli argomenti trattati nei 5 anni di scuola?"))

@Rule(AND(Fact(mid_interest=True), Fact(school=False)))
def ask1_high(self):
    self.declare(Fact(feedback1=valid_answer(
        "\n6a) Ti piacerebbe conoscere il mondo della programmazione o l'architettura di una macchina?")))
    self.declare(Fact(feedback2=valid_answer(
        "6b) Ti interessa sviluppare intelligenze artificiali o conoscere il funzionamento delle reti?")))
    self.declare(Fact(feedback3=valid_answer(
        "6c) Ti piacerebbe gestire basi di dati o studiare algoritmi e strutture dati?")))
```

Figura 2: Se c'è interesse medio (*mid_interest*) e lo studente ha frequentato la scuola, gli viene chiesto se ha gradito gli argomenti trattati; altrimenti, gli viene chiesto se ha interesse per gli argomenti cardine del CdL in Informatica.

2.3 Schema di funzionamento del Sistema Esperto



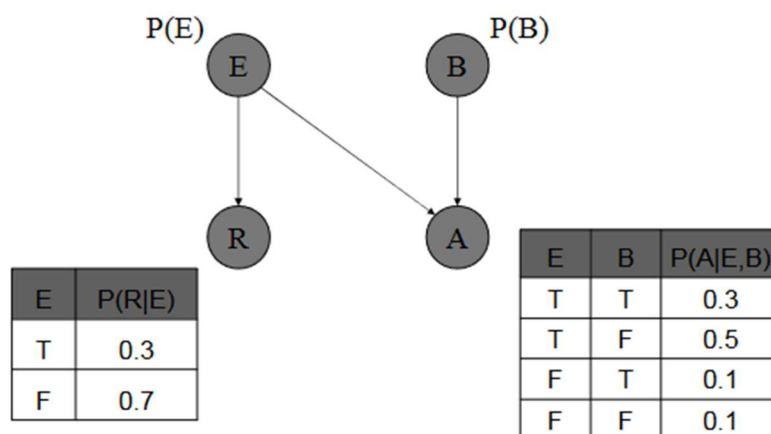
3. Rete bayesiana

3.1 Introduzione

Come secondo modello si è usato una **rete bayesiana**, un Grafo Diretto Aciclico (DAG) definito come segue:

- Ad ogni nodo corrisponde una variabile aleatoria;
- Ogni arco tra due nodi X ed Y rappresenta una influenza diretta tra i due nodi;
- Ad ogni nodo è associata una Tabella delle Probabilità condizionate (CPT) ad eccezione dei nodi radice.

Esempio



Bisogna creare una relazione di ordinamento tra nodi. La rete bayesiana usa le probabilità e il **teorema di Bayes** per poter calcolare la probabilità di un determinato evento, dipendendo (o meno) da altri eventi (probabilità condizionata). La rete bayesiana è una rappresentazione compatta della distribuzione congiunta:

$$P(A_1, A_2, \dots, A_n) = \prod P(A_i | \text{predecessori}(A_i))$$

L'affinità al corso di laurea in Informatica dipende da requisiti, che dipendono a loro volta da specifiche domande. Utilizzando la rete bayesiana possiamo esplicitare la **dipendenza** tra domande e requisiti andando a creare un nodo per ogni requisito e per ogni domanda con degli archi che partono dall'affinità e finiscono nei nodi relativi ai requisiti e poi ancora alle relative domande. In questa maniera, possiamo imporre una relazione d'ordine ed assegnare ad ogni nodo dei genitori (se esistono).

Una volta costruita una rete bayesiana con una sua struttura (la **DAG**) e le sue tabelle per la probabilità condizionata, possiamo inferire la probabilità che un determinato evento si verifichi avendo osservato il verificarsi di alcuni eventi.

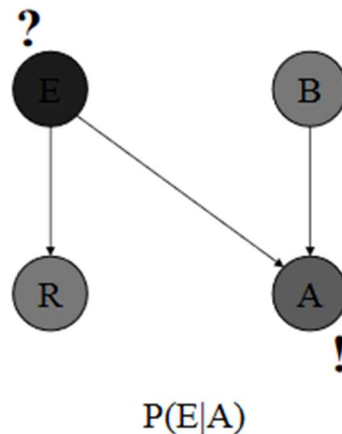
Data una osservazione (evidenza, valori assegnati a delle variabili) calcolare la distribuzione di probabilità per un insieme di variabili di query:

$$P(\text{Query} \mid \text{Evidenza})$$

Ogni variabile può essere considerata sia query che test.

Inferenza Diagnostica:

➤ dagli effetti alle cause



- **inferenza esatta**: si va ad enumerare i mondi coerenti con le osservazioni per poi sfruttare un algoritmo per calcolare la probabilità esatta dell'evento su cui si indaga.

- **inferenza approssimata**: si va a stimare la probabilità di un certo evento.

Nel caso in cui non si conoscano a priori le tabelle delle probabilità condizionate degli eventi, è possibile ricavarle andando ad analizzare i dati e, usando opportuni algoritmi, stimare le probabilità condizionate dei nodi di una DAG.

Per stimare le probabilità, è possibile procedere con lo stimatore di massima verosimiglianza.

3.2 Implementazione

Per implementare la rete bayesiana in Python si è fatto uso di **bnlearn**, una libreria *wrapper* contenente un'altra libreria, **pgmpy**.

La libreria **bnlearn** consente la creazione di una DAG, la creazione ed assegnazione delle tabelle di probabilità condizionate per ciascun nodo (con la classe **TabularCPD**); permette inoltre di inferire le probabilità per un nodo della DAG, andando a segnalare le osservazioni sull'utente tramite il metodo della eliminazione delle variabili. Inoltre, **bnlearn** permette di imparare anche le tabelle delle probabilità, andando a stimare da un dataset.

Nel sistema viene permesso di scegliere come opzione lo **stimatore di massima verosimiglianza** (MLE) che per comprendere meglio descriviamo come:

Supponiamo di avere una variabile casuale osservabile X , per un certo esperimento, che assuma valori in un insieme S .

Supponiamo inoltre che la distribuzione di X dipenda da un parametro ignoto a , suscettibile di assumere valori in uno spazio parametrico A .

Più specificamente, indicheremo con: $f(x | a)$ la funzione di densità di X in x . In generale, sia X che a sono vettori.

La funzione di verosimiglianza L è la funzione che si ottiene invertendo i ruoli di x e a ; ovvero interpretando a come la variabile x come l'informazione nota (cioè il punto di vista della stima): **$L(a | x) = f(x | a)$ per a appartenente a A e x appartenente a S .** Col metodo della massima verosimiglianza, si cerca un valore $u(x)$ del parametro a che massimizzi $L(a | x)$ per ogni x in S . Se riusciamo a trovare tale valore, $u(X)$ è detto stimatore di massima verosimiglianza di a . Con il metodo si cerca di trovare i valori dei parametri che possono aver prodotto con la maggiore probabilità i dati osservati.

Per la creazione della DAG, è stato creato un vettore di coppie, ciascuna formata da due nodi del grafo che indicano un arco che li collega. Creata una rete bayesiana funzionante, bisogna raccogliere le osservazioni dell'utente (i requisiti che pensa di rispettare) ed inferire la probabilità che l'utente sia compatibile con il CdL. Per far ciò, vengono poste delle domande all'utente; ogni requisito osservato avrà valore 0 se assente e 1 se invece è presente. Dopodiché, andiamo ad usare la funzione di inferenza di *bnlearn* che va ad usare l'eliminazione di variabili per ottenere la probabilità di affinità.

In alternativa, se non si dispone delle tabelle di probabilità per ogni nodo, si può apprendere le probabilità usando lo stimatore.

Basta dare una DAG in input combinata con un dataset: ogni riga rappresenta un utente intervistato. Una volta passati questi input, è possibile creare una rete con stimatore di massima verosimiglianza con probabilità apprese e non date.

Valori **TabularCPD** per la valutazione in Rete Bayesiana ideale per quesito:

1) Pensi che la tecnologia stia influenzando tanto la tua vita?

values=[[0.53, 0.47], [0.47, 0.53]]

2) L'informatica e' per te importante nel mondo del lavoro?

values=[[0.53, 0.47], [0.47, 0.53]]

3) Sei interessato alle materie di indirizzo tecnologico-scientifico?

values=[[0.65, 0.35], [0.35, 0.65]]

4) Sei disposto ad approfondire argomenti matematici?

values=[[0.65, 0.35], [0.35, 0.65]]

5) Hai frequentato un istituto secondario di 2° grado, settore Informatica?

values=[[0.65, 0.50], [0.35, 0.50]]

6) Ti piacerebbe conoscere il mondo della programmazione o l'architettura di una macchina?

values=[[0.80, 0.20], [0.20, 0.80]]

7) Ti interessa sviluppare intelligenze artificiali o conoscere il funzionamento delle reti?

values=[[0.80, 0.20], [0.20, 0.80]]

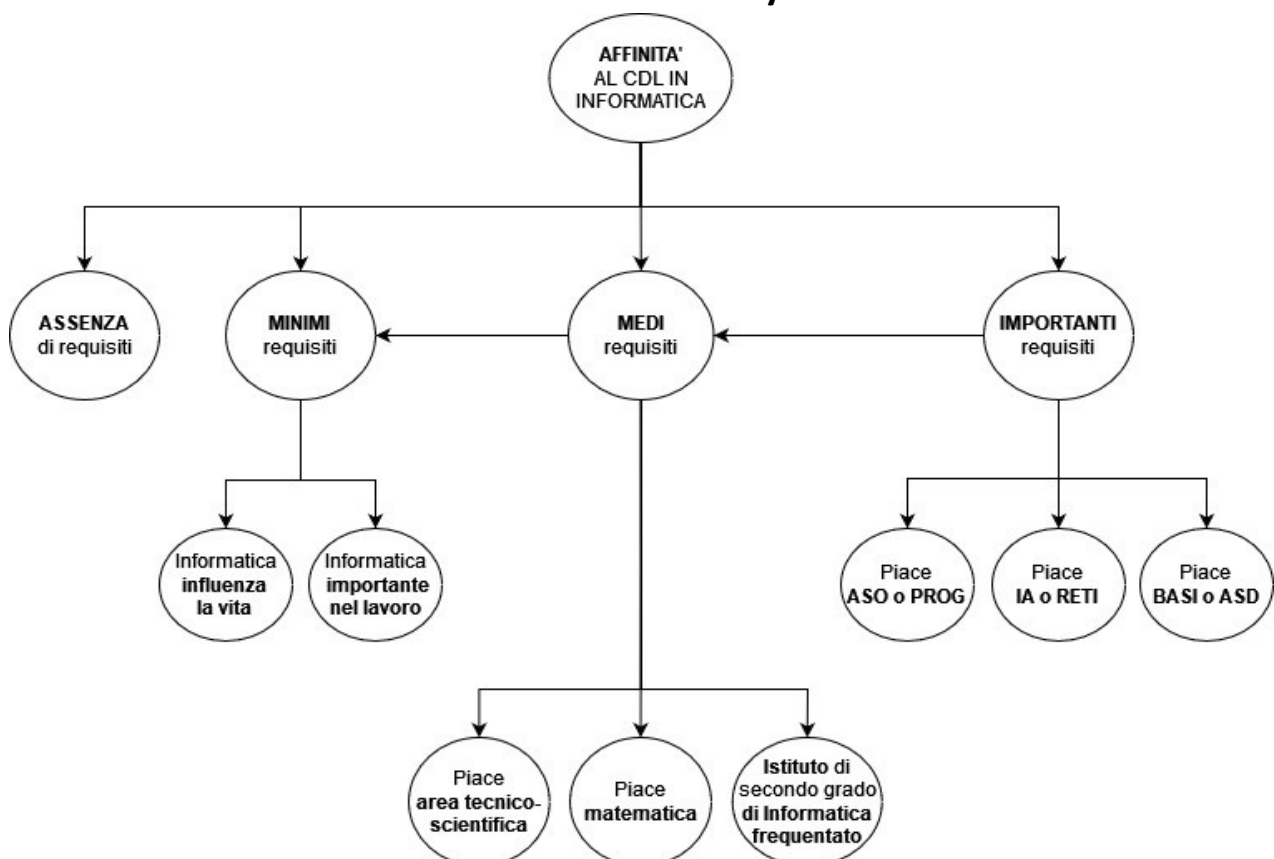
8) Ti piacerebbe gestire basi di dati o studiare algoritmi e strutture dati?

values=[[0.80, 0.20], [0.20, 0.80]]

Si è scelto di definire il livello di affinità come:

1. MOLTO ALTO se $\geq 75\%$;
2. MEDIAMENTE ALTO se $\geq 60\%$ e $< 75\%$;
3. MEDIAMENTE BASSO se $\geq 40\%$ e $< 60\%$;
4. INSUFFICIENTE se $\geq 20\%$ e $< 40\%$;
5. ASSENTE se $< 20\%$;

3.3 Schema di funzionamento della Rete Bayesiana



4. Esempi e valutazioni

4.1 Rete bayesiana ideale vs Rete bayesiana con MLE

```
*Valutazione con rete bayesiana IDEALE*

1) Pensi che la tecnologia stia influenzando tanto la tua vita? (s/n): s
2) L'informatica e' per te importante nel mondo del lavoro? (s/n): n

3) Sei interessato alle materie di indirizzo tecnologico-scientifico? (s/n): s
4) Sei disposto ad approfondire argomenti matematici? (s/n): s
5) Hai frequentato un istituto secondario di 2° grado, settore Informatica? (s/n): n

6) Ti piacerebbe conoscere il mondo della programmazione o l'architettura di una macchina? (s/n): s
7) Ti interessa sviluppare intelligenze artificiali o conoscere il funzionamento delle reti? (s/n): n
8) Ti piacerebbe gestire basi di dati o studiare algoritmi e strutture dati? (s/n): n

La compatibilita' con il CdL in Informatica e' del 39%

**La facolta' soddisfa POCHE tue esigenze, iscrizione NON consigliata**.
```

Fig. 1a L'utente oggetto del questionario ritiene che l'informatica non sia fondamentale in ambito lavorativo, non ha una formazione tecnico-scientifica e non intende approfondire argomenti come l'intelligenza artificiale o le basi di dati; pertanto, il sistema assegna all'utente una bassa percentuale di compatibilità con il CdL di Informatica (pari al 39%), suggerendo giustamente di non iscriversi al CdL in Informatica.

```
*Valutazione con rete bayesiana con STIMATORE DI MASSIMA VEROSOMIGLIANZA*

DATASET CARICATO CON SUCCESSO!

1) Pensi che la tecnologia stia influenzando tanto la tua vita? (s/n): s
2) L'informatica e' per te importante nel mondo del lavoro? (s/n): n

3) Sei interessato alle materie di indirizzo tecnologico-scientifico? (s/n): s
4) Sei disposto ad approfondire argomenti matematici? (s/n): s
5) Hai frequentato un istituto secondario di 2° grado, settore Informatica? (s/n): n

6) Ti piacerebbe conoscere il mondo della programmazione o l'architettura di una macchina? (s/n): s
7) Ti interessa sviluppare intelligenze artificiali o conoscere il funzionamento delle reti? (s/n): n
8) Ti piacerebbe gestire basi di dati o studiare algoritmi e strutture dati? (s/n): n

La compatibilita' con il CdL in Informatica e' del 22%

**La facolta' soddisfa POCHE tue esigenze, iscrizione NON consigliata**.
```

Fig. 1b L'utente si comporta come visto precedentemente nella Fig. 1a, questa volta in una valutazione con rete bayesiana con stimatore di massima verosimiglianza, ottenendo comunque una percentuale bassa (22%) coerente con quella ottenuta in Fig. 1a.

Valutazione con rete bayesiana IDEALE

- 1) Pensi che la tecnologia stia influenzando tanto la tua vita? (s/n): *n*
- 2) L'informatica e' per te importante nel mondo del lavoro? (s/n): *n*
- 3) Sei interessato alle materie di indirizzo tecnologico-scientifico? (s/n): *n*
- 4) Sei disposto ad approfondire argomenti matematici? (s/n): *s*
- 5) Hai frequentato un istituto secondario di 2° grado, settore Informatica? (s/n): *n*
- 6) Ti piacerebbe conoscere il mondo della programmazione o l'architettura di una macchina? (s/n): *s*
- 7) Ti interessa sviluppare intelligenze artificiali o conoscere il funzionamento delle reti? (s/n): *n*
- 8) Ti piacerebbe gestire basi di dati o studiare algoritmi e strutture dati? (s/n): *s*

La compatibilita' con il CdL in Informatica e' del 70%

La facolta' soddisfa mediamente le tue esigenze, iscrizione CONSIGLIATA

Fig. 2a L'utente in questione, benché apparentemente non interessato al settore dell'informatica, ottiene comunque un punteggio medio (70%), poiché è interessato ad argomenti cardine del CdL in Informatica, quali la programmazione e la conoscenza di algoritmi e strutture dati ed è inoltre disposto ad approfondire argomenti matematici.

Valutazione con rete bayesiana con STIMATORE DI MASSIMA VEROSIMIGLIANZA

DATASET CARICATO CON SUCCESSO!

- 1) Pensi che la tecnologia stia influenzando tanto la tua vita? (s/n): *n*
- 2) L'informatica e' per te importante nel mondo del lavoro? (s/n): *n*
- 3) Sei interessato alle materie di indirizzo tecnologico-scientifico? (s/n): *n*
- 4) Sei disposto ad approfondire argomenti matematici? (s/n): *s*
- 5) Hai frequentato un istituto secondario di 2° grado, settore Informatica? (s/n): *n*
- 6) Ti piacerebbe conoscere il mondo della programmazione o l'architettura di una macchina? (s/n): *s*
- 7) Ti interessa sviluppare intelligenze artificiali o conoscere il funzionamento delle reti? (s/n): *n*
- 8) Ti piacerebbe gestire basi di dati o studiare algoritmi e strutture dati? (s/n): *s*

La compatibilita' con il CdL in Informatica e' del 64%

La facolta' soddisfa mediamente le tue esigenze, iscrizione CONSIGLIATA

Fig. 2b L'utente si comporta come visto precedentemente nella Fig. 2a, questa volta in una rete bayesiana con stimatore di massima verosimiglianza, ottenendo ancora una percentuale media (64%) coerente con quella ottenuta in Fig. 2a.

Valutazione con rete bayesiana IDEALE

1) Pensi che la tecnologia stia influenzando tanto la tua vita? (s/n): *n*

2) L'informatica e' per te importante nel mondo del lavoro? (s/n): *n*

3) Sei interessato alle materie di indirizzo tecnologico-scientifico? (s/n): *s*

4) Sei disposto ad approfondire argomenti matematici? (s/n): *n*

5) Hai frequentato un istituto secondario di 2° grado, settore Informatica? (s/n): *n*

6) Ti piacerebbe conoscere il mondo della programmazione o l'architettura di una macchina? (s/n): *s*

7) Ti interessa sviluppare intelligenze artificiali o conoscere il funzionamento delle reti? (s/n): *s*

8) Ti piacerebbe gestire basi di dati o studiare algoritmi e strutture dati? (s/n): *s*

La compatibilita' con il CdL in Informatica e' del 97%

Complimenti! La facolta' soddisfa molto le tue esigenze, iscrizione CALDAMENTE consigliata

Fig. 1c L'utente è interessato alle materie di indirizzo STEM e mostra interesse per quelli che sono alcuni degli argomenti fondamentali del CdL in Informatica; dunque, il sistema consiglia fortemente all'utente di iscriversi a tale CdL, con una percentuale di compatibilità pari al 97%.

Valutazione con rete bayesiana con STIMATORE DI MASSIMA VEROSOMIGLIANZA

DATASET CARICATO CON SUCCESSO!

1) Pensi che la tecnologia stia influenzando tanto la tua vita? (s/n): *n*

2) L'informatica e' per te importante nel mondo del lavoro? (s/n): *n*

3) Sei interessato alle materie di indirizzo tecnologico-scientifico? (s/n): *s*

4) Sei disposto ad approfondire argomenti matematici? (s/n): *n*

5) Hai frequentato un istituto secondario di 2° grado, settore Informatica? (s/n): *n*

6) Ti piacerebbe conoscere il mondo della programmazione o l'architettura di una macchina? (s/n): *s*

7) Ti interessa sviluppare intelligenze artificiali o conoscere il funzionamento delle reti? (s/n): *s*

8) Ti piacerebbe gestire basi di dati o studiare algoritmi e strutture dati? (s/n): *s*

La compatibilita' con il CdL in Informatica e' del 92%

Complimenti! La facolta' soddisfa molto le tue esigenze, iscrizione CALDAMENTE consigliata

Fig. 2c L'utente si comporta come visto precedentemente nella Fig. 1c, questa volta in una rete bayesiana con stimatore di massima verosimiglianza, ottenendo comunque una percentuale ugualmente alta (92%) coerente con quella ottenuta in Fig. 1c.

5. Librerie

Le librerie fondamentali per il corretto funzionamento del programma e contenute all'interno del file requirements.txt sono:

- **bnlearn~0.6.0** : Costruito sulla libreria pgmpy e contiene i pipeline bayesiani più richiesti. E' semplice ed intuitivo ed è focalizzato sull'apprendimento della struttura, dei parametri e l'inferenza.
1. **experta~1.9.4** è una libreria open-source per Python adatta alla costruzione di sistemi esperti.
 2. **pandas~1.4.0** è una libreria open-source per Python che fornisce strutture dati efficienti e flessibili per l'elaborazione e l'analisi dei dati. È particolarmente utile per la manipolazione e l'analisi dei dati tabulari e per la costruzione di dataframe, che sono strutture simili a tabelle in un foglio di calcolo. Con pandas, è possibile eseguire facilmente operazioni complesse come groupby, join, pivot, reshape e molte altre ancora. pandas è molto popolare tra data scientists, data analysts e sviluppatori di software che lavorano con grandi quantità di dati e viene utilizzato in molti progetti di data science e analisi aziendali.
 3. **pgmpy~0.1.17** è una libreria open-source per la modellizzazione probabilistica grafica in Python, tecnica utilizzata per rappresentare e analizzare relazioni causali e probabilistiche tra variabili casuali. La libreria pgmpy fornisce un insieme di strumenti per la creazione, inferenza e manipolazione di modelli probabilistici grafici, che possono essere utilizzati in molte applicazioni, come ad esempio classificazione, regressione e previsione.

6. Conclusioni

I vari modelli utilizzati offrono diverse modalità per calcolare l'affinità.

Il primo analizzato risulta essere molto selettivo e andrebbero ampliate le casistiche di ottenimento del risultato finale. Le reti bayesiane (secondo e terzo modello implementato) riescono a garantire un risultato coerente con la realtà.

In generale andrebbe migliorata l'interazione con l'utente nella fase di output del risultato, in cui si potrebbero dare informazioni più dettagliate a riguardo.

In futuro, il progetto sviluppato potrà essere utilizzato da più facoltà per l'orientamento ai corsi di laurea universitari, ma anche adattato a diversi contesti d'uso ad esempio per misurare l'affinità di un individuo a una particolare offerta lavorativa, un hobby o qualsiasi altra cosa "diagnosticabile".