



Sistema di Supporto alle Decisioni per la Salute Mentale Studentesca

Integrazione di Machine Learning, Web Semantico, Prolog e
Reti Bayesiane

Giuseppe Lucarelli

Matricola: 801275

g.lucarelli19@studenti.uniba.it

REPO:

[https://github.com/lucarellig/Progetto-Salute-Mentale-
Studentesca-ICON-25-
26/tree/main/Progetto_Depressione_Studenti](https://github.com/lucarellig/Progetto-Salute-Mentale-Studentesca-ICON-25-26/tree/main/Progetto_Depressione_Studenti)

AA 2025/26

Indice

0. Introduzione.....	3
0.1 Il Dominio di Interesse: La Salute Mentale Studentesca	3
0.2 Sommario: Architettura del Knowledge-Based System (KBS)	3
0.3 Elenco Argomenti di Interesse.....	4
1. Apprendimento Supervisionato e Ingegneria dei Dati	5
1.1 Sommario.....	5
1.2 Strumenti utilizzati.....	5
1.3 Decisioni di Progetto	5
1.4 Valutazione	6
2. Rappresentazione della Conoscenza e Sistemi a Regole (DSS)	10
2.1 Sommario.....	10
2.2 Strumenti utilizzati.....	10
2.3 Decisioni di Progetto	10
2.4 Valutazione	11
3. Ragionamento Probabilistico in Condizioni di Incertezza	12
3.1 Sommario.....	12
3.2 Strumenti utilizzati.....	12
3.3 Decisioni di Progetto e Scelte Modificate	12
3.4 Valutazione	13
4. Conclusioni.....	15
4.1 Possibili Sviluppi (Problematiche non affrontate).....	15
4.2 Riferimenti Bibliografici	15

0. Introduzione

0.1 Il Dominio di Interesse:

La Salute Mentale Studentesca

Il benessere psicologico degli studenti universitari (spesso indicato come *Student Depression*) è un tema complesso e influenzato da una molteplicità di fattori. L'insorgenza di disturbi depressivi, infatti, raramente dipende da una singola causa, ma deriva quasi sempre dall'interazione di aspetti ambientali, socio-economici (come lo stress finanziario) e legati alla pressione per i risultati accademici. A questi si aggiungono fattori di stile di vita, come la qualità del sonno e l'alimentazione, o eventuali predisposizioni genetiche familiari.

Riconoscere precocemente questo mix di elementi per poter fornire un supporto mirato (che sia uno sportello psicologico, un aiuto economico o un tutoraggio) rappresenta una bella sfida per le università. In questo contesto, usare un approccio puramente statistico mostra dei limiti: per prendere decisioni sulla salute non basta una semplice previsione numerica, ma serve un sistema in grado di spiegare il perché delle proprie scelte (*Explainable AI*) e di distinguere le vere relazioni di causa-effetto dalle normali correlazioni statistiche.

0.2 Sommario:

Architettura del Knowledge-Based System (KBS)

Per far fronte a questa complessità, il progetto illustra lo sviluppo di un *Knowledge-Based System* (KBS) ibrido, pensato per operare come un *Decision Support System* (DSS) con scopi preventivi. Invece di limitarci a utilizzare classificatori statistici, abbiamo adottato un'architettura Neuro-Simbolica che combina in un'unica pipeline l'apprendimento automatico, la logica formale e il ragionamento probabilistico.

Il flusso di lavoro si sviluppa in questo modo: i dati grezzi passano innanzitutto per un modulo basato su Ontologie (OWL 2), dove un reasoner deduce nuove conoscenze e fattori di rischio latenti (*Background Knowledge*), arricchendo dal punto di vista semantico le informazioni di partenza. I dati così potenziati vengono passati ai modelli di Apprendimento Supervisionato (*Machine Learning*), che si occupano di estrarre i pattern e classificare il rischio di depressione. Parallelamente, per gestire l'incertezza e capire le cause del fenomeno, si sfrutta una Rete Bayesiana Causale, molto

utile per effettuare simulazioni avanzate (analisi "What-If") e osservare l'effetto di compensazione. Infine, le predizioni statistiche e le deduzioni logiche vengono convogliate in un Sistema Esperto scritto in Prolog. Quest'ultimo agisce come decisore finale: applica le policy dell'Ateneo (ad esempio consigliando l'accesso a sportelli di ascolto o l'indirizzamento verso borse di studio) e fa da "rete di sicurezza" per arginare i falsi negativi generati dal machine learning.

0.3 Elenco Argomenti di Interesse

Il progetto tocca in modo trasversale diverse competenze previste dal programma del corso, organizzate in tre aree tematiche principali:

- **Argomento 1: Apprendimento Supervisionato e Ingegneria dei Dati.**

Questo modulo copre tutta la parte di analisi esplorativa (EDA), preparazione dei dati (gestione dei valori mancanti e discretizzazione) e l'addestramento comparativo di vari modelli (come la Random Forest o la Regressione Logistica). Lo scopo è quantificare, tramite *Stratified Cross-Validation*, quanto l'iniezione della Background Knowledge ontologica riesca a migliorare i risultati predittivi, prestando un'attenzione particolare alla *Feature Importance* e alla minimizzazione dei Falsi Negativi.

- **Argomento 2: Rappresentazione della Conoscenza e Sistemi a Regole (DSS).**

Riguarda la componente simbolica e deduttiva del progetto. Comprende la progettazione della base di conoscenza in OWL, l'uso del ragionatore HermiT (basato sull'algoritmo *Hypertableau*) per inferire le classi di rischio e lo sviluppo vero e proprio del sistema a regole in Prolog (basato su clausole di Horn). Questo sistema viene poi interrogato dinamicamente a runtime per assegnare i protocolli di intervento agli studenti.

- **Argomento 3: Ragionamento Probabilistico in Condizioni di Incertezza.**

Si concentra sull'impiego delle Reti Bayesiane discrete. Viene proposto un confronto critico tra un approccio di apprendimento automatico della struttura dai dati (*Structure Learning* tramite Hill Climbing) e un approccio di modellazione causale guidato dall'esperto (*Expert-Driven*). Infine, l'inferenza probabilistica (*Variable Elimination*) viene usata per

dimostrare come il sistema riesca a simulare meccanismi preventivi come l'effetto di compensazione ("Explaining Away").

1. Apprendimento Supervisionato e Ingegneria dei Dati

1.1 Sommario

L'obiettivo di questa fase è estrarre pattern utili dallo *Student Depression Dataset*, che ha come variabile target Depression (0=Sano, 1=Depresso). Il lavoro è stato diviso in due passaggi: prima una fase di *Data Engineering* per pulire i dati e fare un'analisi esplorativa (EDA), e poi l'addestramento di quattro algoritmi di classificazione (Albero di Decisione, Random Forest, K-NN, Regressione Logistica).

Lo scopo principale è capire se e quanto l'aggiunta di *Background Knowledge* (la conoscenza semantica estratta successivamente dall'ontologia) migliori le performance dei modelli, cercando soprattutto di limitare gli errori clinici.

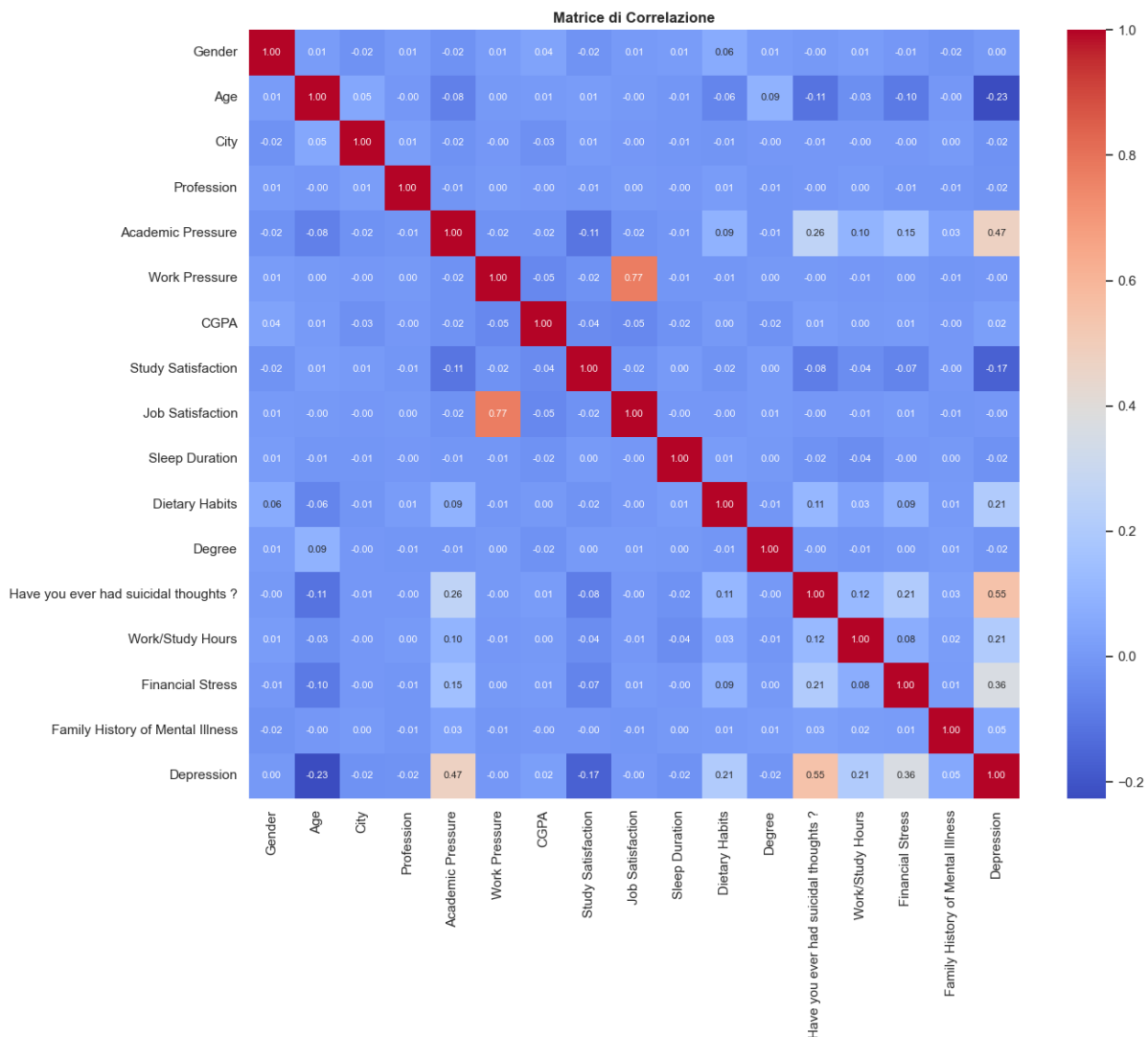
1.2 Strumenti utilizzati

Tutto il codice è stato scritto in Python. Per la gestione dei dati e l'ingegneria delle feature è stata usata la libreria pandas, mentre i grafici dell'analisi esplorativa sono stati creati con matplotlib e seaborn. Infine, per l'addestramento dei modelli, il partizionamento del dataset e il *Label Encoding* ci si è affidati a scikit-learn.

1.3 Decisioni di Progetto

Lavorando con dati clinici e universitari, sono state prese alcune decisioni mirate per la preparazione del dataset:

- **Valori mancanti (Missing Values):**
Per gestire i record incompleti nelle colonne numeriche, si è scelto di usare la mediana al posto della classica media. Questo approccio aiuta a non falsare i dati nel caso in cui ci siano valori anomali (ad esempio, studenti che dichiarano un numero irrealistico di ore di studio giornaliero).
- **Mantenimento delle variabili poco correlate:**
Dalla matrice di correlazione di Pearson è emerso che variabili come *Job Satisfaction* e *Work Pressure* non avevano quasi alcun legame



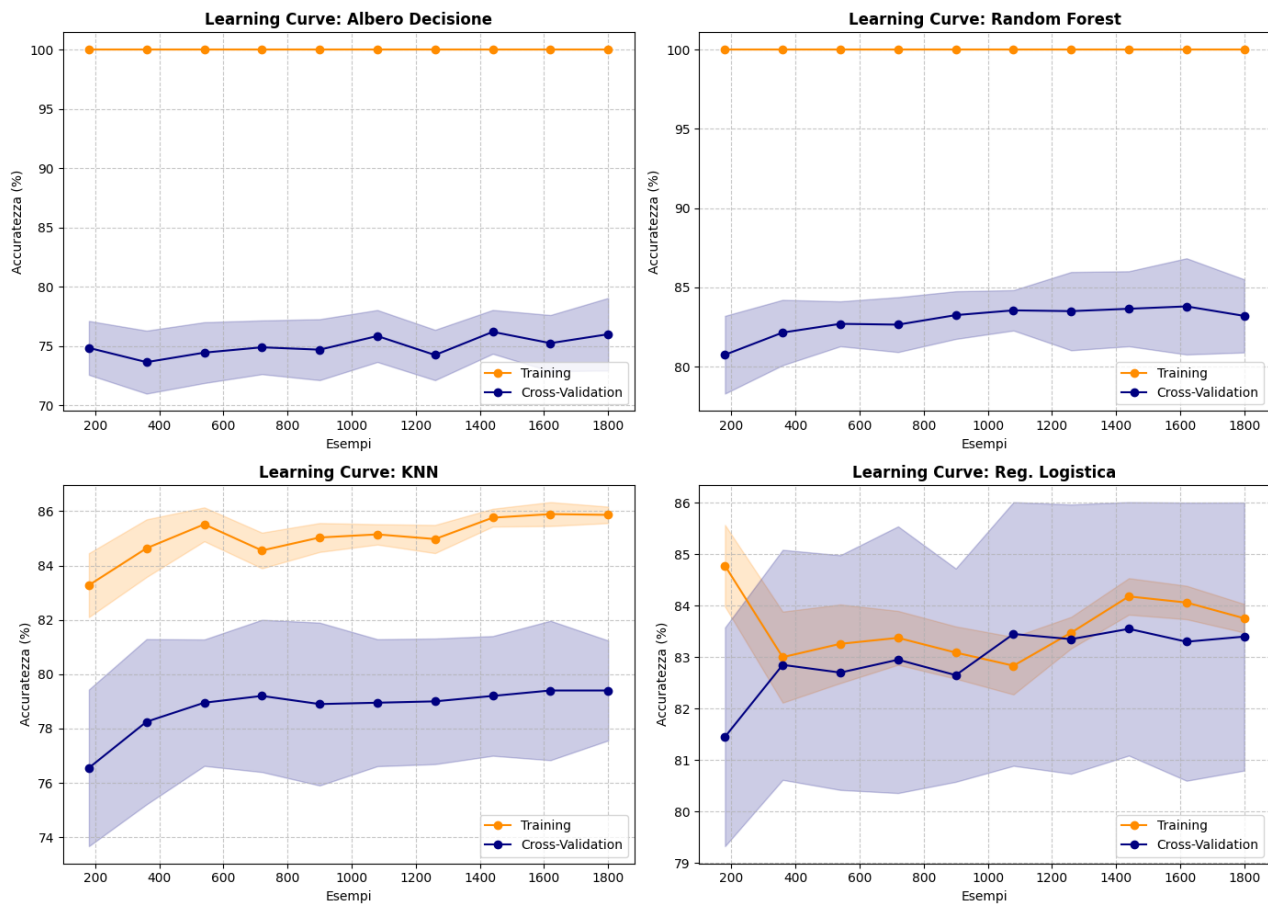
lineare col target. Invece di scartarle subito con una *Feature Selection*, si è deciso di tenerle per testare la capacità pratica della Random Forest di ignorare in automatico le feature inutili (rumore informativo).

- **Label Encoding:** Tutte le risposte testuali ai questionari sono state trasformate in numeri sequenziali. Era un passaggio obbligato per far funzionare correttamente algoritmi come il K-NN, che si basano matematicamente sul calcolo delle distanze nello spazio.

1.4 Valutazione

Per evitare l'overfitting e avere stime affidabili, i modelli sono stati testati utilizzando una 10-Fold Stratified Cross-Validation. A ulteriore conferma della bontà dell'addestramento, l'analisi della curva di apprendimento (Learning

Curve) dimostra che, sebbene la Random Forest tenda fisiologicamente a memorizzare il training set (raggiungendo il 100% di accuratezza), la curva della Cross-Validation cresce e si stabilizza correttamente all'aumentare dei campioni, confermando un'ottima e stabile capacità di generalizzazione sui nuovi dati.



Di fatto è stato condotto un esperimento A/B: un primo addestramento sul dataset di base pre-processato (Test A)

--- VALUTAZIONE: DATI GREZZI (NO ONTOLOGIA) ---				
Modello	Accuracy	Precision	Recall	F1-Score
Albero Decisione	76.00 ±3.06%	75.38 ±3.16%	75.19 ±3.16%	75.12 ±3.14%
Random Forest	83.20 ±2.30%	82.92 ±2.18%	82.21 ±2.81%	82.41 ±2.61%
KNN	79.40 ±1.84%	79.20 ±1.78%	77.69 ±2.23%	78.15 ±2.15%
Reg. Logistica	83.40 ±2.61%	83.08 ±2.52%	82.46 ±3.14%	82.63 ±2.90%

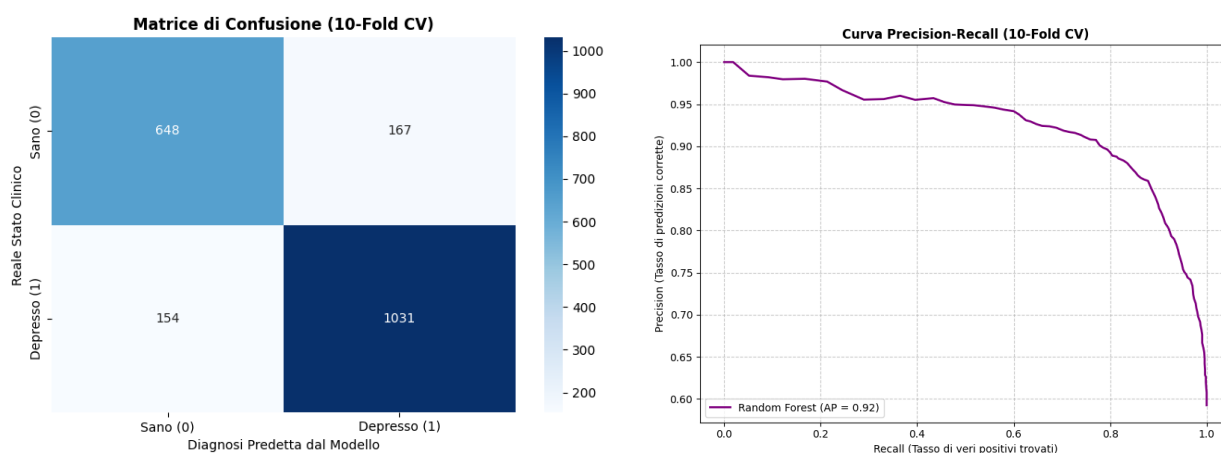
e uno sul dataset arricchito con la nuova variabile logica Onto_Rischio_Alto (Test B).

--- VALUTAZIONE: DATI CON ONTOBK ---

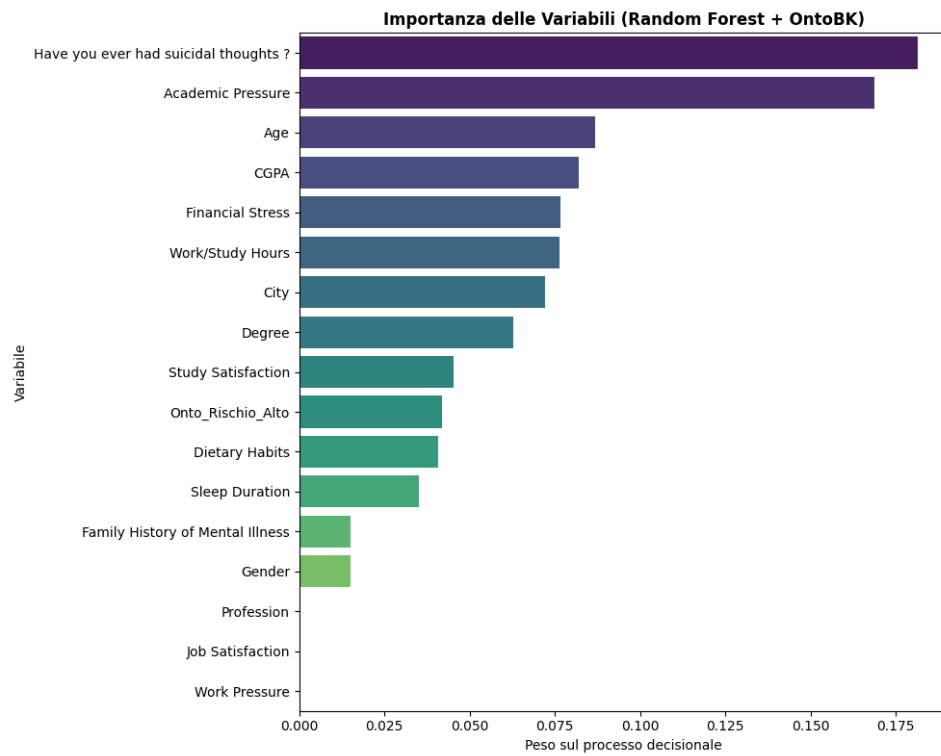
Modello	Accuracy	Precision	Recall	F1-Score
Albero Decisione	75.95 \pm 2.60%	75.27 \pm 2.71%	74.91 \pm 2.82%	74.95 \pm 2.79%
Random Forest	83.95 \pm 1.96%	83.56 \pm 2.10%	83.25 \pm 2.07%	83.32 \pm 2.06%
KNN	79.75 \pm 1.93%	79.55 \pm 2.05%	78.12 \pm 2.14%	78.57 \pm 2.12%
Reg. Logistica	83.55 \pm 2.72%	83.24 \pm 2.64%	82.62 \pm 3.24%	82.79 \pm 3.01%

I risultati indicano che l'uso della Background Knowledge semantica rende le predizioni più stabili e precise. Il modello migliore è risultato la Random Forest, con un'accuratezza media dell'83.95% (\pm 1.96%).

Tuttavia, trattandosi di prevenzione sanitaria, guardare solo all'accuratezza globale non basta. La priorità del sistema doveva essere l'abbassamento dei Falsi Negativi, poiché non diagnosticare uno studente effettivamente depresso rappresenta un errore molto grave. Osservando la matrice di confusione affiancata alla curva Precision-Recall, si nota che la Random Forest potenziata dall'ontologia riesce a isolare bene i casi patologici. In particolare, la curva PR dimostra il raggiungimento di un ottimo compromesso, garantendo un valore di *Recall* molto solido senza far crollare la *Precision* (evitando così troppi falsi allarmi agli sportelli di supporto).



Infine, per capire come il modello prende le sue decisioni (Explainable AI), è stata estratta la Gini Feature Importance. Il grafico conferma che, sebbene i sintomi clinici diretti (come i pregressi pensieri suicidi) o lo stress accademico dominino fisiologicamente le decisioni, la nuova variabile dedotta logicamente dall'ontologia (Onto_Rischio_Alto) emerge come una feature ingegnerizzata di grande rilevanza. Supera persino parametri fisici e di stile di vita fondamentali come le abitudini alimentari e le ore di sonno, dimostrando che il Machine Learning trae un beneficio concreto dall'iniezione di conoscenza semantica esterna.



2. Rappresentazione della Conoscenza e Sistemi a Regole (DSS)

2.1 Sommario

I modelli di Machine Learning (ML) puri presentano il limite di comportarsi come "scatole nere", faticando a cogliere la semantica profonda del dominio clinico. Questo modulo integra due sottosistemi per superare tale limite: una *Knowledge Base* (KB) in logica descrittiva, progettata per dedurre fattori di rischio latenti, e un motore inferenziale a regole (clausole di Horn) che funge da *Decision Support System* (DSS). L'obiettivo è trasformare le previsioni statistiche e le deduzioni semantiche in protocolli operativi di intervento (una sorta di vero e proprio "trriage" universitario preventivo).

2.2 Strumenti utilizzati

L'ontologia di dominio è stata sviluppata in Python utilizzando la libreria *owlready2*. Per il ragionamento automatico (*Reasoning*) sulle classi si è impiegato **HermiT**, un reasoner basato sull'algoritmo *Hypertableau*. Il sistema normativo e decisionale è stato invece scritto in **Prolog**. L'integrazione tra la pipeline Python e il motore Prolog per il passaggio dinamico dei dati a runtime è stata gestita tramite l'interfaccia *pyswip*.

2.3 Decisioni di Progetto

La costruzione della KB e delle regole decisionali ha richiesto alcune scelte architetturali precise:

- **Classe definita per Equivalenza Logica:** Per evitare di usare l'ontologia come un semplice database relazionale, l'appartenenza alle classi di rischio non è stata assegnata in modo statico. Nel modulo OWL (TBox), l'appartenenza alla classe *StudenteAdAltoRischio* è definita per Equivalenza Logica (\equiv). Di conseguenza, un individuo viene sussunto nella classe dal reasoner solo se soddisfa l'intersezione (\cap) di specifiche condizioni critiche (ad esempio, alto stress finanziario unito a un'elevata pressione accademica).
- **Gestione della Complessità Computazionale:** Durante la fase di *reparenting* è emersa una criticità hardware. I ragionatori per OWL 2 hanno un costo computazionale nel caso pessimo pari a **NEXPTIME-complete**. Elaborare simultaneamente tutte le oltre 27.000 istanze dell'ABox avrebbe saturato la memoria (Out-Of-Memory) bloccando

l'esecuzione. Si è quindi optato per un campionamento stratificato del dataset riducendolo a 2000 istanze, un compromesso che ha garantito la trattabilità logica del problema in tempi di calcolo brevi.

- **Base di Conoscenza Dinamica e Protocollo DELTA:** Il motore Prolog non interroga una KB statica, ma riceve i "fatti" dinamicamente a runtime (tramite il predicato `assertz`), incrociando l'output del ML e dell'ontologia. Nelle regole implementate, il **Protocollo DELTA** svolge una funzione cruciale di *Safety Net* (rete di sicurezza). Questa regola si innesca quando il modello ML classifica erroneamente uno studente come "sano", ma l'ontologia rileva al contempo una combinazione di fattori ambientali altamente critica. In questo scenario, il Prolog scavalca il "punto cieco" del classificatore statistico e prescrive comunque un colloquio preventivo, riducendo drasticamente il rischio di falsi negativi clinici.

2.4 Valutazione

Il sistema ha dimostrato di poter arricchire i dati in ingresso e generare azioni correttive in modo deterministico. Nella prima fase, Hermit ha valutato correttamente gli assiomi popolandolo la nuova feature semantica `Onto_Rischio_Alto` all'interno del dataset.

Nella fase finale, il motore Prolog ha processato i profili degli studenti restituendo output decisionali univoci. Incrociando le regole logiche, il sistema ha assegnato con successo interventi mirati: dal *Protocollo ALFA* per gestire le emergenze assolute, al *Protocollo BETA* per indirizzare verso i bandi per il Diritto allo Studio gli studenti il cui disagio aveva una forte radice economica.

ID PAZIENTE	PROFILO (ML / Ontologia / Stress)	DECISIONE DSS
Studente #896	ML:Alto / Onto:Critico / Fin:Alto	PROTOCOLLO ALFA -> Rischio severo. Attivazione immediata sportello psicologico e tutorato accademico.
Studente #178	ML:Alto / Onto:Sicuro / Fin:Alto	PROTOCOLLO BETA -> Rischio clinico associato a disagio economico. Indirizzare bando DSU (Diritto allo Studio).
Studente #1196	ML:Alto / Onto:Sicuro / Fin:Basso	PROTOCOLLO GAMMA -> Rischio clinico isolato. Inserimento in lista di monitoraggio e invio questionario screening.
Studente #1317	ML:Basso / Onto:Critico / Fin:Alto	PROTOCOLLO DELTA -> Rischio latente rilevato dall'Ontologia (Falso Negativo ML). Consigliato colloquio preventivo.
Studente #507	ML:Basso / Onto:Sicuro / Fin:Basso	PROTOCOLLO OMEGA -> Nessuna criticità. Invio newsletter standard su benessere studentesco e gestione del sonno.

L'architettura proposta rispetta così il paradigma dell'*Explainable AI* richiesto in ambito medico, offrendo una giustificazione formale, tracciabile e basata su regole logiche per ogni singola decisione intrapresa.

3. Ragionamento Probabilistico in Condizioni di Incertezza

3.1 Sommario

L'apprendimento supervisionato (Sezione 1) e le logiche descrittive (Sezione 2) funzionano molto bene per classificare e dedurre informazioni in modo esatto, ma presentano un limite: non gestiscono nativamente l'incertezza e non spiegano le vere relazioni di causa-effetto tra i sintomi clinici. Per analizzare l'eziologia del problema (cioè le sue cause), in questo modulo è stata implementata una Rete Bayesiana Discreta (DAG - *Directed Acyclic Graph*). Lo scopo non è più solo classificare, ma poter fare interrogazioni probabilistiche (le cosiddette analisi "What-If"), aggiornando le stime di rischio clinico man mano che si inseriscono nuove evidenze sul paziente.

3.2 Strumenti utilizzati

Il modello probabilistico è stato sviluppato in Python sfruttando la libreria pgmpy. Questo strumento è stato impiegato per stimare i parametri statistici (tramite *Maximum Likelihood*), per tentare l'apprendimento automatico della struttura della rete (*Structure Learning* tramite *Hill Climbing* e score BIC) e per fare inferenza con l'algoritmo esatto di *Variable Elimination*. Per disegnare e formattare i grafi sono state usate le librerie networkx e matplotlib.

3.3 Decisioni di Progetto e Scelte Modificate

La costruzione della Rete Bayesiana non è stata lineare, ma ha richiesto vari tentativi e un cambio di approccio in corso d'opera:

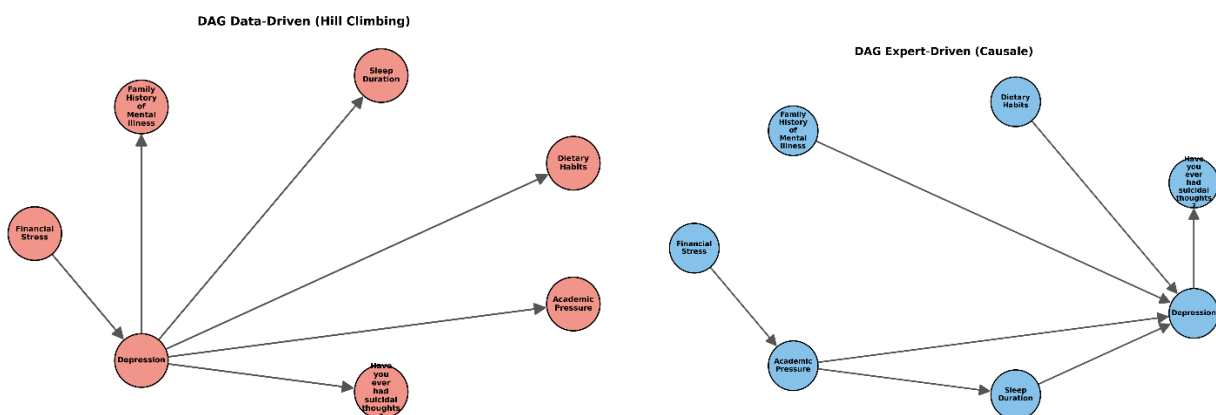
- **Selezione delle variabili:** Per non incappare nell'esplosione combinatoria tipica degli algoritmi di ricerca probabilistica, la rete non è stata costruita sull'intero dataset, ma limitata a 7 nodi chiave. Si è scelto deliberatamente di non selezionare le variabili basandosi solo sulla *Feature Importance* della Random Forest (che tende a premiare i sintomi finali e prettamente clinici, come i pensieri suicidi), ma di focalizzarsi sulle cause originarie del disagio. Il campionamento ha quindi incluso nodi ambientali scatenanti (come lo stress accademico), affiancati a fattori biologico-comportamentali e genetici (come il sonno, la dieta e la storia familiare). Sfruttando i dati già discretizzati nella Fase

1, l'algoritmo ha potuto valutare le combinazioni causali esplorando lo spazio di ricerca in modo rapido, convergendo all'ottimo globale in 6 iterazioni.

- **Il problema dell'approccio Data-Driven:** L'idea di partenza era far costruire l'intera struttura del grafo all'algoritmo dai dati (approccio *Data-Driven*). Purtroppo, la rete generata dall'Hill Climbing non aveva alcun senso clinico. L'algoritmo, limitato dalle Classi di Equivalenza di Markov, ha confuso la forte correlazione statistica con la vera causalità temporale. Di conseguenza, nel grafo generato in automatico, il nodo Depression è diventato la radice, indicando la situazione assurda per cui lo stato depressivo fosse la causa della storia genetica dello studente o del suo carico di studio.
- **Il passaggio all'Expert-Driven:** Davanti a questo limite evidente dell'IA puramente statistica, si è deciso di scartare l'apprendimento automatico della struttura e di passare a un approccio *Expert-Driven*. Gli archi del grafo sono stati inseriti manualmente per rispettare il reale decorso medico: i fattori ambientali e genetici puntano (come genitori) verso la depressione, che a sua volta agisce come causa verso sintomi più gravi, come i pensieri suicidi. La libreria pgmpy è stata quindi usata solo per il *Parameter Learning*, ovvero per compilare le Tabelle delle Probabilità Condizionate (CPT) partendo dal dataset di training.

3.4 Valutazione

Mettendo a confronto i due grafi, emerge chiaramente perché in ambito medico non ci si possa affidare solo a metriche puramente matematiche per definire la struttura eziologica di un sistema.



Testando la rete finale (quella Expert-Driven) con l'algoritmo di Variable Elimination, il modello si è dimostrato capace non solo di calcolare le

probabilità diagnostiche, ma anche di simulare il fenomeno clinico della compensazione (*Explaining Away*). Ad esempio, inserendo nel sistema l'evidenza isolata di uno stress accademico massimo, la rete stima un rischio di depressione dell'85.43%. Tuttavia, aggiungendo l'informazione protettiva che lo studente dorme più di 8 ore a notte e mangia in modo sano, la probabilità marginale crolla drasticamente al 50.00% (evidenziando una riduzione del rischio clinico di oltre 35 punti percentuali). Questo risultato empirico e matematico dimostra che il modello bayesiano ha "imparato" a riconoscere l'effetto cuscinetto dei fattori ambientali positivi, simulando in modo logico il ragionamento differenziale che farebbe un professionista umano.

```
=====
CONFRONTO INFERENZA E DIMOSTRAZIONE 'EXPLAINING AWAY'
=====
SCENARIO A: Paziente con Pressione Accademica Massima (5.0)
-> Rischio Depressione Stimato : 85.43%

SCENARIO B: Pressione Massima (5.0) + Sonno > 8h + Dieta Sana
-> Rischio Depressione Stimato : 50.00%

-----
EFFETTO COMPENSAZIONE (Explaining Away) DIMOSTRATO:
Il rischio e' crollato del 35.43% grazie ai fattori ambientali positivi.
=====
```

4. Conclusioni

Il progetto ha esplorato la progettazione e lo sviluppo di un *Knowledge-Based System* ibrido, applicato al delicato ambito della salute mentale studentesca. Mettendo insieme tre diversi paradigmi dell'Intelligenza Artificiale, si è cercato di superare i limiti tipici dei modelli puramente statistici arrivando a strutturare un *Decision Support System* capace non solo di fare previsioni, ma anche di motivarle.

L'integrazione tra i vari moduli ha portato a un flusso di lavoro completo: il *Machine Learning* si è occupato di estrarre i pattern latenti dai dati storici; la logica descrittiva, unita alle regole Prolog, ha permesso di inserire nel sistema la conoscenza di dominio e di definire protocolli di sicurezza (come il Protocollo DELTA, pensato per arginare i falsi negativi del classificatore); infine, le Reti Bayesiane hanno fornito un modello formale per gestire l'incertezza e le reali relazioni di causa-effetto. Nel suo insieme, l'architettura implementata si allinea ai principi della *Explainable AI* (XAI), offrendo uno strumento che fornisce indicazioni diagnostiche e operative tracciabili.

4.1 Possibili Sviluppi (Problematiche non affrontate)

In fase di progettazione si era valutata l'idea di impiegare tecniche di *Inductive Logic Programming* (ILP) per far apprendere automaticamente le clausole Prolog analizzando direttamente i dati del dataset. Alla fine questa strada non è stata implementata per una precisa scelta logica: i protocolli universitari (come i requisiti per l'assegnazione di un bando DSU) sono norme amministrative rigide, non pattern deducibili dalla distribuzione statistica dei sintomi di un paziente.

Tuttavia, uno sviluppo futuro molto interessante potrebbe prevedere l'uso dell'ILP non sui dati clinici, ma sugli storici amministrativi (ad esempio i log delle segreterie). Questo permetterebbe al sistema di estrarre in automatico le logiche latenti con cui l'Ateneo ha storicamente gestito le borse di studio, per poi tradurle in nuove regole Prolog da aggiungere alla base di conoscenza. Altre possibili estensioni riguardano l'ampliamento dell'ontologia per valutare l'impatto specifico dei singoli corsi di laurea sullo stress accademico.

4.2 Riferimenti Bibliografici

Il presente progetto implementa a livello pratico i modelli teorici descritti nei seguenti moduli del programma d'esame:

- **Apprendimento Supervisionato (Cap. 7):** Riferimento teorico per i modelli di classificazione, la gestione dell'overfitting (Cross-Validation) e i modelli composti (Random Forest) utilizzati nella *Sezione Argomento 1*.
- **Rappresentazione e Ragionamento Relazionale (Cap. 15):** Riferimento per il calcolo dei predicati, le dimostrazioni e i simboli di funzione alla base del Sistema Esperto in Prolog (*Sezione Argomento 2*).
- **Ontologie: Knowledge Graph (Cap. 16):** Riferimento per la modellazione delle classi, l'uso delle Logiche Descrittive e del Web Semantico (OWL) per l'estrazione della Background Knowledge (*Sezione Argomento 2*).
- **Ragionamento e Incertezza (Cap. 9) & Apprendimento e Incertezza (Cap. 10):** Riferimento teorico per le Belief Network, l'inferenza probabilistica (Cap. 9) e le metriche di Structure Learning (Cap. 10) applicate nell'analisi causale della *Sezione Argomento 3*.