Predizione della malignità

del cancro al seno

*Gruppo di lavoro*

*Francesca Mantini, MATR. 757122,* [*f.mantini@studenti.uniba.it*](mailto:f.mantini@studenti.uniba.it)

[*https://github.com/frumantini/ProgettoIcon*](https://github.com/frumantini/ProgettoIcon)

*AA 2023-24*

Sommario

[Introduzione 3](#_Toc176943654)

[Sommario 3](#_Toc176943655)

[Elenco argomenti di interesse 3](#_Toc176943656)

[Requisiti funzionali 4](#_Toc176943657)

[Dataset 5](#_Toc176943658)

[Rappresentazione della Conoscenza 8](#_Toc176943659)

[Sommario 8](#_Toc176943660)

[Strumenti utilizzati 8](#_Toc176943661)

[Decisioni di Progetto 8](#_Toc176943662)

[Valutazione 10](#_Toc176943663)

[Apprendimento supervisionato 11](#_Toc176943664)

[Sommario 11](#_Toc176943665)

[Strumenti utilizzati 11](#_Toc176943666)

[Decisioni di Progetto 12](#_Toc176943667)

[Valutazione 15](#_Toc176943668)

[Apprendimento non supervisionato 18](#_Toc176943669)

[Sommario 18](#_Toc176943670)

[Strumenti utilizzati 18](#_Toc176943671)

[Decisioni di Progetto 18](#_Toc176943672)

[Valutazione 19](#_Toc176943673)

[Ragionamento probabilistico 23](#_Toc176943674)

[Sommario 23](#_Toc176943675)

[Strumenti utilizzati 23](#_Toc176943676)

[Decisioni di Progetto 24](#_Toc176943677)

[Valutazione 24](#_Toc176943678)

[Conclusioni 27](#_Toc176943679)

[Riferimenti Bibliografici 27](#_Toc176943680)

# Introduzione

Il cancro al seno rappresenta una delle principali cause di mortalità tra le donne a livello globale. La diagnosi precoce e accurata gioca un ruolo cruciale nel migliorare le prospettive di trattamento e sopravvivenza

Il presente progetto si concentra sullo sviluppo di un sistema avanzato di supporto alle decisioni per la diagnosi del cancro al seno, integrando tecniche di apprendimento automatico, rappresentazione della conoscenza e analisi dei dati. L'obiettivo principale è fornire uno strumento affidabile e preciso per assistere i professionisti medici nella classificazione dei tumori mammari, distinguendo tra casi benigni e maligni.

# Sommario

Il Knowledge-Based System (KBS) sviluppato integra diversi moduli che dimostrano competenze in vari argomenti del corso di Intelligenza Artificiale:

1. Rappresentazione della conoscenza: utilizzo di un'ontologia (BCGO) per formalizzare la conoscenza del dominio (Sezione Rappresentazione della Conoscenza).

2. Ragionamento automatico: implementazione di una rete bayesiana per l'inferenza probabilistica (Sezione Ragionamento Probabilistico).

3. Apprendimento supervisionato: applicazione di algoritmi di machine learning per la classificazione dei tumori (Sezione Apprendimento Supervisionato).

4. Apprendimento non supervisionato: utilizzo di tecniche di clustering per identificare potenziali sottotipi di cancro (Sezione Apprendimento Non Supervisionato).

# Elenco argomenti di interesse

**Integrazione della Conoscenza Semantica (capitolo 16 di** [**[1]**](#rif1)**)**

Prima fase del processo, che consiste nell'integrazione di conoscenza semantica derivata da un'ontologia specializzata sul cancro al seno ([BCGO.owl](https://bioportal.bioontology.org/ontologies/BCGO)). Questa fase, implementata nel modulo semanticIntegration.py, arricchisce i dati clinici con informazioni semantiche estratte dall'ontologia, creando nuove feature che catturano concetti e relazioni rilevanti per la diagnosi del cancro al seno.

**Apprendimento supervisionato (capitolo 7 di** [**[1]**](#rif1)**)**

Questa fase comprende la selezione delle feature più rilevanti, la normalizzazione dei dati e l'addestramento di diversi modelli di classificazione, tra cui K-Nearest Neighbors, Decision Trees, Random Forests e Support Vector Machines.

Per ogni modello addestrato, il sistema genera una serie di output grafici, inclusi matrici di confusione, curve ROC, curve di apprendimento e un riepilogo delle performance, che forniscono una valutazione dettagliata delle prestazioni di ciascun classificatore.

**Apprendimento non supervisionato (capitolo 10 di** [**[1]**](#rif1)**)**

Segue la fase di unsupervised learning, in cui viene utilizzata la tecnica del clustering K-means per identificare una scala di gravità dell’avanzamento del tumore, mostrando la capacità del sistema di scoprire pattern nascosti nei dati.

**Ragionamento probabilistico (capitolo 9 di** [**[1]**](#rif1)**)**

Il progetto si conclude con l’implementazione di una rete bayesiana, in grado di modellare le relazioni probabilistiche tra le caratteristiche del tumore e la sua malignità, consentendo inferenze in condizioni di incertezza.

# Requisiti funzionali

Il progetto è stato interamente sviluppato utilizzando il linguaggio Python, nell'ambiente di sviluppo integrato (IDE) **Visual Studio Code**. Di seguito le librerie utilizzate, con una descrizione delle loro funzionalità principali:

* **Scikit-learn (v. 1.5.1)**: Libreria fondamentale per l'analisi dei dati e l'apprendimento automatico. Contiene strumenti per il pre-processing dei dati, la feature extraction e la feature selection, nonché una vasta gamma di modelli di apprendimento supervisionato e non supervisionato. Inoltre, include metriche per la valutazione delle performance dei modelli.
* **Pandas (v. 2.3.0)**: Fornisce strumenti potenti e flessibili per la manipolazione e l'analisi dei dati, permettendo di organizzare i dati in strutture chiamate DataFrame e di effettuare operazioni avanzate su di essi in modo efficiente.
* **Numpy (v. 2.1.0)**: Questa libreria offre funzionalità avanzate per il calcolo numerico e permette operazioni ad alte prestazioni su array multidimensionali e matrici, facilitando l'elaborazione di dati numerici nello spazio .
* **Scipy (v. 1.11.3)**: Offre funzioni avanzate per il calcolo scientifico, come ottimizzazione, integrazione, interpolazione e operazioni sulle matrici sparse, che estendono le capacità di Numpy.
* **Matplotlib (v. 3.7.2)**: Libreria di base per la creazione di grafici in Python. Permette di generare una vasta gamma di visualizzazioni, come istogrammi, grafici a dispersione e linee temporali.
* **Seaborn (v. 0.13.2)**: Una libreria di visualizzazione dei dati basata su Matplotlib, che consente di creare grafici statistici avanzati, incluse le rappresentazioni delle distribuzioni e le matrici di confusione, utili per l'interpretazione dei risultati del modello.
* **Pgmpy (v. 0.1.21)**: Strumento dedicato alla costruzione e all'analisi di modelli grafici probabilistici (PGM), come reti bayesiane o modelli di Markov.
* **Networkx (v. 3.1)**: Utilizzata per la creazione, la manipolazione e lo studio di strutture di grafi e reti complesse, questa libreria è particolarmente utile per problemi che coinvolgono interconnessioni tra nodi.
* **Rdflib (v. 6.0.2)**: Una libreria per il lavoro con RDF (Resource Description Framework), fondamentale per lo sviluppo di applicazioni semantiche e la gestione di dati collegati.
* **Kneed (v. 0.8.3)**: Un pacchetto utilizzato per rilevare i cosiddetti "gomiti" nelle curve, comunemente impiegato per determinare il numero ottimale di cluster in algoritmi di clustering.
* **Imbalanced-learn (v. 0.11.0)**: Libreria dedicata al trattamento di dataset sbilanciati, fornendo strumenti per il bilanciamento delle classi, come il resampling e il sovra-sottocampionamento.

Per garantire il corretto funzionamento del progetto, è necessario installare tutte le librerie sopra esposte. Ciò può essere fatto eseguendo il comando seguente nel terminale:

*“pip install -r requirements.txt”*

direttamente sulla macchina locale, nel percorso predefinito dell'interprete Python, o in un ambiente virtuale dedicato, che consente di isolare le dipendenze del progetto, evitando conflitti tra librerie e versioni diverse. Nel mio caso ho utilizzato **Anaconda** come segue:

*“conda create --name nome\_ambiente python=3.10”*

*“conda activate nome\_ambiente”*

*“pip install -r requirements.txt”*

# Dataset

Nel presente caso di studio si è scelto di utilizzare il dataset [Breast Cancer Wisconsin (Diagnostic)](https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data/data), un insieme di dati ampiamente riconosciuto nel campo della diagnosi del cancro al seno. Questo dataset contiene 569 istanze estratte da immagini digitali di agoaspirati (FNA) di masse mammarie, le quali descrivono le caratteristiche dei nuclei cellulari presenti in immagini digitalizzate.

**Attributi**

1. **ID**: Numero identificativo univoco
2. **Diagnosi**: M = maligno (212), B = benigno (357)

3-32. **Caratteristiche dei nuclei cellulari**: per ogni nucleo cellulare, vengono calcolate 10 caratteristiche reali, di cui, per ognuna, vengono forniti tre valori (media, errore standard (SE), "worst" o "largest" (media dei tre valori più grandi)).

* raggio: media delle distanze dal centro ai punti sul perimetro,
* texture: deviazione standard dei valori di scala di grigi,
* perimetro,
* area,
* smoothness: variazione locale nelle lunghezze dei raggi,
* compattezza, data da perimetro^2 / area - 1.0,
* concavità: gravità delle porzioni concave del contorno,
* punti concavi: numero di porzioni concave del contorno,
* simmetria,
* dimensione frattale, data da "approssimazione della linea costiera" – 1.

Si è deciso di scegliere questo dataset perché, essendo basato su immagini di FNA reali ed essendo globalmente riconosciuto come affidabile, il dataset riflette accuratamente le sfide della diagnosi clinica, con 30 caratteristiche forniscono una descrizione dettagliata dei nuclei cellulari, permettendo un'analisi approfondita.

# Preprocessing e analisi dei dati

Il modulo main.py rappresenta il punto di ingresso del sistema e coordina l'esecuzione delle diverse fasi del processo di analisi. Analizziamo in dettaglio le sue componenti principali:

La funzione preprocess\_data si occupa di preparare il dataset per l'analisi:

Immagine che contiene testo, schermata, Carattere

Descrizione generata automaticamente

La fase di **preprocessing** consta di: rimozione della colonna 'id', se presente, per evitare che identificatori non pertinenti influenzino l'analisi, conversione della variabile target da categorica ('M' per maligno, 'B' per benigno) a numerica (1 per maligno, 0 per benigno) e, infine, eliminazione di eventuali colonne con tutti i valori mancanti. Si è voluta aggiungere quest’ultima operazione, nonostante il dataset preso in analisi non presenti valori nulli, per rendere il codice adattabile ad altri dataset con formato simile.

La distribuzione delle diagnosi viene visualizzata attraverso un grafico a torta per offrire una panoramica immediata della composizione del dataset.

Immagine che contiene testo, diagramma, schermata, cerchio

Descrizione generata automaticamente

Sebbene leggermente sbilanciato e relativamente piccolo, il dataset permette un'analisi equa di entrambe le classi diagnostiche, motivo per cui si è deciso di non utilizzare tecniche di undersampling o oversampling, che, come verrà analizzato in seguito, aumentano solo l’overfitting, riducendo drasticamente le dimensioni o creando dati fittizi che sovraddestrano il modello.

# Integrazione della Conoscenza Semantica

## Sommario

L’integrazione della conoscenza nel sistema si basa su un'ontologia specializzata per il dominio del cancro al seno, denominata [BCGO (Breast Cancer Grading Ontology)](https://bioportal.bioontology.org/ontologies/BCGO), per arricchire i dati clinici con informazioni semantiche rilevanti. Questa metodologia mira a migliorare la qualità e la profondità delle informazioni disponibili per i successivi processi di apprendimento automatico e analisi predittiva.

La rappresentazione della conoscenza scelta per la Knowledge Base (KB) si basa su un modello ontologico, che fornisce una struttura formale e gerarchica per rappresentare concetti, relazioni e proprietà nel dominio del cancro al seno. L'ontologia BCGO è stata selezionata per la sua completezza e rilevanza nel campo specifico della gradazione del cancro al seno. Questa scelta permette di integrare conoscenze di dominio esperto con i dati clinici, che invece costituiscono la base empirica su cui si applica l'apprendimento.

Il processo di integrazione della conoscenza semantica coinvolge l'estrazione di classi e proprietà rilevanti dall'ontologia, la loro mappatura sui dati clinici esistenti (features estratte da immagini mammografiche) e la creazione di nuove features basate su questa conoscenza. Questo approccio non solo arricchisce il set di dati originale, ma introduce anche una dimensione di interpretabilità e contesto che può essere cruciale per l'analisi e la comprensione dei risultati.

## Strumenti utilizzati

Per l'implementazione e l'utilizzo dell'ontologia, si è utilizzata la libreria RDFLib, un framework che permette di lavorare con dati RDF (Resource Description Framework), il formato standard per la rappresentazione di ontologie e dati semantici.

RDFLib è stata scelta perché permette di caricare ed effettuare il parsing dell'ontologia BCGO dal file OWL (Web Ontology Language), convertendo la struttura dell'ontologia in un oggetto Graph RDF manipolabile. Una volta caricata, l'ontologia può essere interrogata utilizzando query SPARQL o metodi nativi di RDFLib. Questo permette di estrarre informazioni specifiche, come classi, proprietà e relazioni definite nell'ontologia.

Inoltre, questa libreria, facilita la navigazione della struttura gerarchica dell'ontologia, permettendo di risalire le relazioni di sottoclasse per calcolare, ad esempio, la profondità delle classi all'interno dell'ontologia, e l’estrazione di classi e proprietà rilevanti, che vengono poi utilizzate per arricchire i dati clinici con informazioni semantiche.

## Decisioni di Progetto

Prima scelta effettuata è stata quella di estrarre sia le classi che le proprietà dall'ontologia per massimizzare l'informazione semantica disponibile, fattore che permette di catturare non solo la gerarchia concettuale (attraverso le classi) ma anche le relazioni specifiche tra concetti (attraverso le proprietà).

Immagine che contiene testo, schermata

Descrizione generata automaticamente

A questo segue l’implementazione di una funzione per calcolare la profondità di ogni classe nell'ontologia. Questo ci permette di assegnare un peso alle informazioni semantiche basato sulla loro specificità all'interno della struttura ontologica. Infatti, il motivo per cui è importante conoscere la profondità di una classe è quello di capire dove la stessa si trovi rispetto alla radice dell'ontologia. Le classi più vicine a OWL.Thing sono concetti più generali, mentre le classi più lontane sono più specifiche.

Immagine che contiene testo, schermata, Carattere

Descrizione generata automaticamente

Questo è servito da un lato per assicurarsi di evitare il rischio di entrare in cicli infiniti o di errori o eccezioni dovuti alla presenza di una classe senza genitore definito (il calcolo della profondità viene arrestato quando non viene trovato un genitore) e dall’altro, come detto prima, per poter calcolare un punteggio di rilevanza basato sulla profondità di una classe o proprietà nella gerarchia ontologica, per dare più peso alle informazioni più specifiche e dettagliate.



Passo successivo è stato quello di combinare utilizzando una media ponderata tutti i fattori di rischio semantici in un unico indicatore globale, "semantic\_combined\_risk\_factor", di modo da tener conto di diversi aspetti semantici rilevanti per la valutazione del rischio di malignità.

Infine si è deciso di rimuovere automaticamente le colonne con valori costanti per evitare ridondanze e potenziali problemi nell'apprendimento automatico, colonne che durante l’esecuzione del programma sollevavano warning e problematiche di esecuzione.

## Valutazione

La valutazione del componente di rappresentazione della conoscenza si basa principalmente sull'impatto che l'integrazione delle informazioni semantiche ha sulle prestazioni dei modelli di apprendimento automatico.

Uno dei pregi dell’integrazione semantica è stato sicuramente l’aumento della dimensionalità dei dati, con l’aggiunta di 4 nuove colonne semantiche, quali: 'semantic\_risk\_factor\_size', 'semantic\_risk\_factor\_area', 'semantic\_risk\_factor\_texture', 'semantic\_combined\_risk\_factor', fattore che cerca di ridurre il problema precedentemente esposto della dimensione ridotta del dataset, con nuove caratteristiche discriminanti tra i vari pazienti.

Queste nuove features rappresentano aspetti del dominio che potrebbero non essere immediatamente evidenti nei dati clinici grezzi. L'efficacia di questo approccio è supportata dal fatto che 3 su 4 di queste feature semantiche sono state selezionate come rilevanti nel processo successivo di feature selection, indicando il loro valore predittivo.

In conclusione, l'integrazione della conoscenza semantica attraverso l'uso dell'ontologia BCGO ha dimostrato di essere un approccio efficace per migliorare la predizione della malignità del cancro al seno. Questo metodo non solo ha incrementato le prestazioni dei modelli predittivi, ma ha anche arricchito il dataset con informazioni semanticamente rilevanti e interpretabili, dando la possibilità di analisi più approfondite e per una migliore comprensione dei fattori che influenzano la malignità del tumore.

La decisione di calcolare un punteggio di rilevanza basato sulla profondità ontologica permette di ponderare l'importanza dei concetti in modo coerente con la struttura della conoscenza del dominio. Tuttavia, future iterazioni del sistema potrebbero beneficiare di un'analisi più approfondita della relazione tra la struttura ontologica e l'effettivo valore predittivo dei concetti nella pratica clinica.

# Apprendimento supervisionato

## Sommario

L’apprendimento supervisionato è una branca dell’apprendimento automatico in cui l’agente viene addestrato su un insieme di dati con etichette. Si divide in

* **Classificazione**, in cui le etichette rappresentano un dominio di valori discreto (quindi un insieme finito di valori) che la feature di output può assumere. Un caso particolare è la classificazione booleana, in cui le classi sono **true** e **false**.
* **Regressione**, in cui l’etichetta può essere un qualsiasi valore numerico, quindi il dominio della feature di output è continuo.

Obiettivo principale è far sì che l'algoritmo impari una relazione tra gli input e gli output in modo che, una volta addestrato, possa fare previsioni accurate su nuovi dati mai visti prima.

L'apprendimento supervisionato costituisce un componente fondamentale del sistema di supporto alle decisioni per la diagnosi del cancro al seno. La rappresentazione della conoscenza adottata si basa su un approccio ibrido che combina dati numerici derivati da immagini diagnostiche con conoscenza di background (BK) integrata attraverso fattori di rischio semantici.

L'obiettivo è addestrare modelli in grado di apprendere le relazioni tra le caratteristiche dei tumori e la loro classificazione, per poi applicare questa conoscenza a nuovi casi non etichettati. Dunque, nel presente caso di studio l’apprendimento supervisionato è stato utilizzato con scopo di classificazione.

## Strumenti utilizzati

I modelli di apprendimento supervisionato che si è scelto di implementare sono 4 e vengono di seguito analizzati.

K-Nearest Neighbors (KNN): è un algoritmo non parametrico che classifica un'istanza basandosi sulla classe maggioritaria tra i k vicini più prossimi nello spazio delle feature. La distanza tra le istanze è calcolata utilizzando una metrica predefinita, tipicamente la distanza euclidea. La complessità computazionale del KNN è O(n\*d) per la fase di predizione, dove n è il numero di istanze nel training set e d è la dimensionalità delle feature.

Decision Tree: costruisce un albero di decisione dove ogni nodo interno rappresenta un test su un attributo, ogni ramo rappresenta l'esito del test e ogni foglia rappresenta una classe. La costruzione dell'albero avviene attraverso la selezione ricorsiva degli attributi che massimizzano una metrica di impurità (e.g., entropia o indice di Gini). La complessità temporale per la costruzione dell'albero è O(n*m*log(n)), dove n è il numero di istanze e m il numero di feature.

Random Forest: è un ensemble di alberi decisionali. Ogni albero è costruito su un sottoinsieme casuale dei dati e delle feature (bagging). La predizione finale è ottenuta attraverso il voto di maggioranza degli alberi. Questo approccio riduce l'overfitting e aumenta la robustezza del modello. La complessità temporale è O(t*n*m\*log(n)), dove t è il numero di alberi.

Support Vector Machine (SVM): cerca di trovare l'iperpiano che meglio separa le classi nello spazio delle feature, massimizzando il margine tra le classi. Per problemi non linearmente separabili, utilizza il "kernel trick" per mappare i dati in uno spazio di dimensionalità superiore. La complessità computazionale dell'SVM con kernel RBF è O(n^2 \* m) per l'addestramento e O(n\_sv \* m) per la predizione, dove n\_sv è il numero di vettori di supporto.

1. Selezione delle Feature (SelectKBest):
2. Valutazione dei Modelli:

a. Cross-validation: Implementata attraverso StratifiedKFold, suddivide i dati in k sottoinsiemi, mantenendo la proporzione delle classi, per una valutazione più robusta.

b. Metriche di Performance: Accuratezza, precisione, recall e F1-score sono calcolate per una valutazione completa delle performance dei modelli.

1. Visualizzazione: Matplotlib e Seaborn sono utilizzati per creare visualizzazioni informative delle performance dei modelli, incluse matrici di confusione, curve ROC e curve di apprendimento.

## Decisioni di Progetto

**Preparazione dei dati**

La prima fase è stata quella di effettuare la Feature selection con SelectKBest, una tecnica per selezionare le caratteristiche più importanti basata sulla funzione statistica ANOVA (f\_classif). Questo processo seleziona il miglior numero di caratteristiche (k) in base alla performance di un classificatore (in questo caso, il RandomForestClassifier) durante la cross-validation, cioè calcolando un punteggio per ogni feature basato sulla sua correlazione con la variabile target e selezionando poi le feature con i punteggi più alti. Nel processo vengono selezionate 19 features, di cui si presenta la matrice di correlazione, uno strumento statistico che mostra le relazioni tra più variabili. È una tabella in cui ogni cella rappresenta il coefficiente di correlazione tra due variabili. Il coefficiente di correlazione misura la forza e la direzione della relazione lineare tra le variabili.

Immagine che contiene testo, schermata, Parallelo, diagramma

Descrizione generata automaticamente

Da questa si evince la forte correlazione tra raggio, perimetro e aerea, perfettamente in linea con il fatto che descrivono le componenti principali della misurazione della massa tumorale.

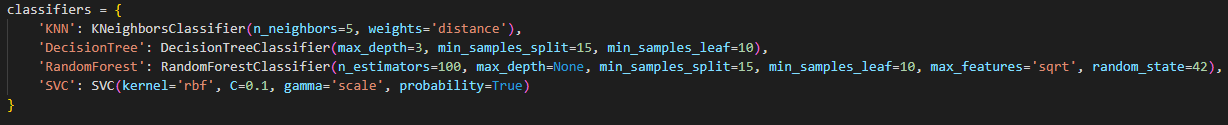
Fase successiva è stata la divisione del dataset in 2 dataset minori, uno per il training (con 455 pazienti) e l’altro per il testing (114 pazienti) e la loro conseguente normalizzazione.

Lo scaler utilizzato è stato MinMaxScaler, che mappa le feature nell'intervallo [0, 1], fondamentale per garantire che tutte le feature contribuiscano equamente all'addestramento del modello, evitando che feature con scale maggiori dominino il processo di apprendimento. Ciò è particolarmente importante per algoritmi sensibili alla scala delle feature, come KNN e SVM. Il MinMaxScaler è stato preferito allo StandardScaler, perché quest’ultimo ha dato problemi di azzeramento di alcuni valori presenti del dataset

**Scelta degli iper-parametri**

Gli iper-parametri sono i parametri di un modello di apprendimento automatico, i quali non vengono appresi durante la fase di addestramento come i normali parametri del modello (es. i pesi di una funzione lineare) ma devono essere necessariamente fissati prima che il modello possa cominciare l’addestramento. La loro scelta influisce sulle prestazioni e sulla complessità del modello. Uno dei compiti più complessi è proprio la scelta degli iper-parametri per i vari modelli.

La scelta è avvenuta manualmente ma ci si prefissa come obiettivo in eventuali sviluppi futuri di utilizzare tecniche come la GridSearchCV o RandomizedSearchCV, che vengono usate per testare automaticamente diverse combinazioni di iperparametri e scegliere i migliori basati sulle performance di cross-validation.



***KNN***:

* n\_neighbors=5: indica il numero di vicini da considerare quando si classifica un punto, rappresenta un compromesso tra la necessità di catturare patterns locali e la robustezza al rumore. La scelta di assegnare un valore pari a 5 è abbastanza comune perché è un compromesso tra avere pochi vicini (che potrebbe portare a un modello rumoroso) e troppi vicini (che potrebbe approssimare troppo).
* weights='distance': specifica come pesare i vicini. Se impostato su 'distance', i vicini più vicini al punto da classificare hanno un peso maggiore rispetto a quelli più lontani.

***Decision* *Tree*:**

* max\_depth=3: imposta la profondità massima dell'albero decisionale, ossia quanti livelli può avere. Un albero con profondità massima di 3 può fare al massimo 3 divisioni. Si è scelto un valore basso per tentare di limitare la complessità del modello e prevenire l'**overfitting**.
* min\_samples\_split=15: numero minimo di campioni richiesti per suddividere un nodo. Questo impedisce che il modello crei suddivisioni con pochi campioni, il che porterebbe a nodi poco significativi, riduce l'eccessiva frammentazione del dataset e mantiene i nodi con più significato statistico.
* min\_samples\_leaf=10: numero minimo di campioni richiesti in una foglia (un nodo che non si suddivide ulteriormente). Questo serve a evitare che il modello finisca con troppe foglie con pochi dati, il che porterebbe a un modello troppo complesso.

***Random* *Forest*:**

* n\_estimators=100: numero di alberi nella foresta. Più alberi si utilizzano, più il modello può essere robusto e ridurre la varianza. 100 è un valore comunemente scelto che, abbastanza grande da stabilizzare il modello ma non troppo grande da diventare inefficiente o portare a costi computazionali eccessivi.
* max\_depth=None: non viene imposta una profondità massima per ogni albero, di modo che ogni albero cresca fino a quando non può più suddividere ulteriormente i nodi (in base ad altri criteri come min\_samples\_split e min\_samples\_leaf). Questa scelta consente di catturare tutte le possibili suddivisioni, riducendo l'errore di bias a costo di aumentare la varianza, ma l'ensemble della foresta mitiga l'overfitting.
* min\_samples\_split=15: come per l'albero decisionale singolo, richiede almeno 15 campioni per suddividere un nodo, riducendo la frammentazione e prevenendo l'overfitting.
* min\_samples\_leaf=10: come per l'albero decisionale singolo, richiede almeno 10 campioni per foglia. Questa scelta aumenta la robustezza degli alberi nel modello di Random Forest, riducendo il rischio di foglie che riflettano eccessivamente il rumore nei dati.
* max\_features='sqrt': indica che, per ogni suddivisione, il modello considera solo la radice quadrata del numero totale di caratteristiche del dataset. Questa strategia introduce casualità nel processo e previene che tutti gli alberi prendano decisioni simili, aumentando la diversità all'interno della foresta. È un iperparametro molto comune nelle Random Forest, poiché aiuta a bilanciare l'accuratezza del modello e la sua varianza.
* random\_state=42: imposta un seme per il generatore di numeri casuali, che garantisce che l'esecuzione del codice produca sempre lo stesso risultato, utile per la riproducibilità degli esperimenti. Il valore assegnato è una convenzione.
* d. Support Vector Machine (SVM): - kernel='rbf': Kernel Radial Basis Function per gestire relazioni non lineari - C=0.1: Regolarizzazione per bilanciare margine e errori di classificazione - gamma='scale': Scala l'influenza di ogni campione in base al numero di feature

***SVM***:

* kernel='rbf' (Radial Basis Function): è una funzione di similarità usata per creare decisioni non lineari. Questo kernel è utile quando i dati non sono linearmente separabili.
* C=0.1: parametro che controlla il compromesso tra una classificazione accurata del training set e la larghezza del margine di separazione (cioè quanto il modello penalizza gli errori di classificazione sui dati di addestramento). Un valore di 0.1 suggerisce che è stato dato più peso alla larghezza del margine piuttosto che all'accuratezza perfetta del training set, il che potrebbe prevenire l'overfitting.
* gamma='scale': iperparametro che definisce quanto influisce un singolo campione sui margini del classificatore. 'scale' è una scelta predefinita e comune, in quanto adatta gamma in base all'inverso del numero di caratteristiche, cercando di evitare sia un valore troppo grande che potrebbe causare overfitting, sia troppo piccolo che potrebbe rendere il modello troppo semplice.
* probability=True: Specifica se il modello deve calcolare le probabilità associate a ciascuna previsione. Questo può rallentare l'addestramento, ma è utile se si vogliono stime probabilistiche piuttosto che semplici classi.

Gli iperparametri scelti sono tutti mirati a bilanciare complessità e rischio di overfitting. Le scelte come i limiti di profondità sugli alberi decisionali, C basso nel SVC, o l'uso di weights='distance' nel KNN hanno come obiettivo quello di evitare modelli troppo complessi, pur mantenendo un buon livello di accuratezza sui dati.

## Valutazione

La valutazione dei modelli è stata condotta utilizzando diverse metriche, ognuna delle quali fornisce informazioni complementari sulle prestazioni dei classificatori.

L’accuratezza del modello è definita come il rapporto tra il numero di esempi correttamente classificati ed il numero di esempi totale

ove TP, TN, FP e FN sono elementi chiave nelle matrici di confusione e rappresentano i diversi risultati possibili di una classificazione binaria. I True Positives e i True Negatives rappresentano il numero di esempi correttamente classificati, rispettivamente, come appartenenti e non alla classe positiva. I False Positives e i False Negatives rappresentano, invece, il numero di esempi erroneamente classificati, rispettivamente, come appartenenti e non alla classe positiva.

Tuttavia, un alto valore di accuratezza non è sufficiente per valutare positivamente un classificatore, ad esempio, quando il test set è molto sbilanciato. Per questo motivo si è presa in considerazione un’altra metrica, detta F1-score o F1-measure, che fornisce una misura complessiva della qualità del classificatore, calcolata come media armonica tra la precisione ed il richiamo:

dove con si indica la precisione, che rappresenta la percentuale degli esempi che viene correttamente classificata, mentre con si indica il richiamo, ovvero la percentuale degli esempi classificati come positivi che sono realmente positivi. Entrambi hanno valori compresi tra 0 ed 1. Idealmente, un classificatore ottimale dovrebbe raggiungere valori di precisione e richiamo entrambi prossimi ad 1.

Altre metriche che si sono calcolate sono:

* l’AUC-ROC (Area Under the Receiver Operating Characteristic Curve), che misura la capacità del modello di discriminare tra le classi. Un valore di 1.0 rappresenta un classificatore perfetto, mentre 0.5 equivale a una classificazione casuale.
* Mean Squared Error (MSE): misura la media dei quadrati degli errori (le differenze tra i valori reali e quelli predetti dal modello). È definito come:

Immagine che contiene Carattere, bianco, design, tipografia

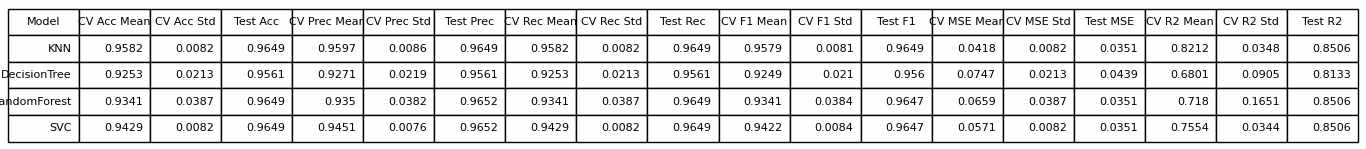
Descrizione generata automaticamente

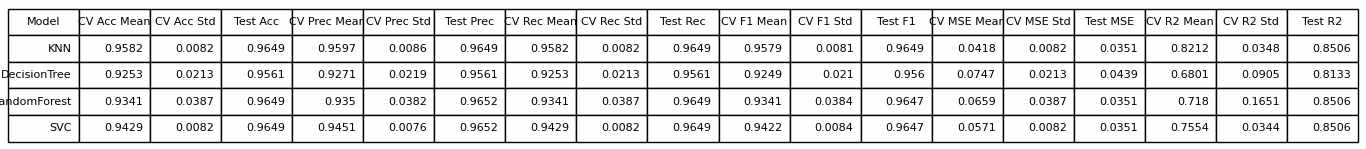
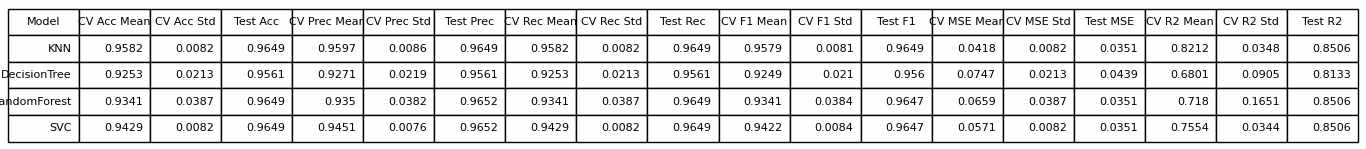
ove n è il numero di osservazioni, yi​ sono i valori reali e ŷi​ sono i valori predetti dal modello.

* R-squared (R2): anche detto coefficiente di determinazione, misura la percentuale di varianza dei dati che è spiegata dal modello. Misura la proporzione della varianza nella variabile dipendente che è prevedibile dalla variabile indipendente. È definito come:

ove SSresidui ​ è la somma dei quadrati degli errori residui, cioè la differenza tra i valori reali e quelli predetti dal modello, mentre SStotale ​ è la somma dei quadrati delle differenze tra i valori reali e la media dei valori reali.

In sintesi, l'MSE valuta l'errore assoluto delle predizioni, mentre l'R² valuta quanto bene il modello spiega i dati.





Dal punto di vista di risultati grafici, sono stati prodotte M*atrici di confusione* per una rapida valutazione visiva degli errori di classificazione, *Curve ROC* per valutare il trade-off tra sensibilità e specificità e *Curve di Apprendimento*, per diagnosticare overfitting o underfitting

Basandoci sui grafici forniti e sui risultati numerici, si procede con un'analisi comparativa delle prestazioni dei quattro modelli implementati

Immagine che contiene testo, diagramma, schermata, linea

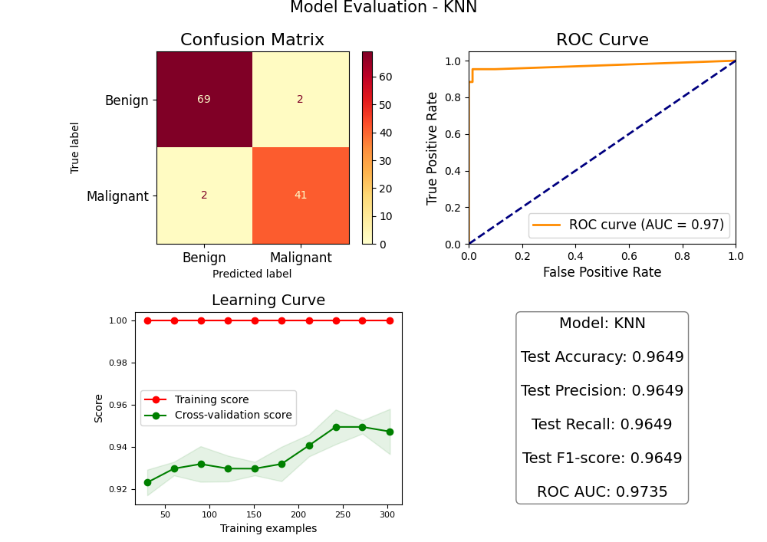
Descrizione generata automaticamente 

Immagine che contiene testo, diagramma, schermata, linea

Descrizione generata automaticamente Immagine che contiene testo, diagramma, schermata, linea

Descrizione generata automaticamente

Tutti i modelli mostrano performance eccellenti, con accuratezze superiori al 94% e AUC-ROC superiori a 0.97, indicando una forte capacità discriminativa tra tumori benigni e maligni. Random Forest, SVC e KNN mostrano performance molto simili in termini di accuratezza, superando leggermente il Decision Tree che, nonostante abbia performance leggermente inferiori, offre il vantaggio di una maggiore interpretabilità.

Le matrici di confusione mostrano una buona capacità di classificazione per entrambe le classi (benigno e maligno), indicando che i modelli non soffrono di bias significativi verso una classe particolare.

L'SVC emerge come il modello più equilibrato, offrendo eccellenti prestazioni con il minor grado di overfitting. Il RandomForest è una valida alternativa, con prestazioni simili ma un po' più di overfitting. Il KNN, nonostante le buone prestazioni sul test set, mostra il maggior grado di overfitting e potrebbe non generalizzare bene su nuovi dati. Il DecisionTree, pur essendo il più interpretabile, potrebbe beneficiare di ulteriore ottimizzazione per ridurre l'overfitting e migliorare le prestazioni.

La scelta tra Random Forest o SVC potrebbe dipendere da considerazioni pratiche come la velocità di inferenza o l'interpretabilità richiesta in un contesto clinico.

Questi risultati supportano l'efficacia dell'approccio adottato, inclusa l'integrazione della conoscenza semantica e le tecniche di preparazione dei dati. Tuttavia, per una validazione completa, sarebbe opportuno testare il sistema su dataset esterni e confrontarlo con le performance dei professionisti medici in contesti reali. Inoltre ci si prefissa come obiettivo in eventuali sviluppi futuri di trovare metodi migliori per riuscire a ridurre in maniera più efficace l’overfitting, che resta un problema non di poco conto.

# Apprendimento non supervisionato

## Sommario

L’apprendimento non supervisionato è una branca dell’apprendimento automatico in cui l’agente viene addestrato su un insieme di dati senza etichette. Esistono due principali tipi di apprendimento non supervisionato:

* Clustering: obiettivo è raggruppare gli elementi del dataset in base a delle somiglianze
* Riduzione della dimensionalità: obiettivo è ridurre il numero di feature utilizzate mantenendo però le informazioni più significative. Un esempio è la *PCA* (*Principal Component Analysis*)

Nel presente caso di studio si è utilizzato l’apprendimento non supervisionato, nello specifico l’algoritmo di hard clustering *KMeans,* è stato utilizzato con lo scopo di effettuare clustering, per raggruppare i casi di tumore in cluster distinti basati sulle loro caratteristiche, senza fare affidamento su etichette predefinite, con lo scopo di creare una scala di gravità dell’avanzamento del tumore.

## Strumenti utilizzati

Il clustering K-means è implementato utilizzando la libreria scikit-learn, una delle più robuste e ampiamente utilizzate per il machine learning in Python. Specificamente, viene impiegata la classe `KMeans` dal modulo `sklearn.cluster`. Questo algoritmo opera iterativamente per assegnare ogni punto dati al cluster il cui centroide è più vicino, minimizzando la somma delle distanze quadratiche all'interno di ciascun cluster.

Per ottimizzare la visualizzazione e l'interpretazione dei risultati, viene applicata l'Analisi delle Componenti Principali (PCA) tramite la classe `PCA` dal modulo `sklearn.decomposition`. Questa tecnica riduce la dimensionalità dei dati, permettendo una rappresentazione visiva efficace dei cluster in due o tre dimensioni.

La libreria matplotlib è utilizzata per la creazione di grafici e visualizzazioni, essenziali per l'interpretazione dei risultati del clustering.

## Decisioni di Progetto

Una delle decisioni chiave nel processo di clustering è la determinazione del numero ottimale di cluster (k). Per affrontare questo problema in modo rigoroso, è stato implementato il metodo del Bayesian Information Criterion (**BIC**). Questo approccio bilancia la complessità del modello (numero di cluster) con la sua capacità di spiegare i dati, evitando l’overfitting, ed è una delle soluzioni più utilizzate per l’identificazione del numero di cluster ideali (si veda pagina 477 di [[1]](#rif1))

Il Bayesian Information Criterion (BIC) è una metrica usata per confrontare diversi modelli statistici, tenendo conto sia della qualità dell'adattamento (quanto bene il modello spiega i dati) sia della complessità del modello (il numero di parametri utilizzati). Il BIC penalizza modelli troppo complessi per evitare overfitting, ovvero quando un modello è troppo aderente ai dati di addestramento e non generalizza bene su nuovi dati.

Il processo di selezione del k ottimale è implementato nella funzione `find\_optimal\_k\_bic()`. Questa funzione utilizza il modello di Mixture Gaussiana (GMM) per calcolare lo score BIC per diversi valori di k, da un minimo di 2 a un massimo di 10 cluster. Il k che minimizza lo score BIC viene selezionato come ottimale.

Parametri chiave per il K-means sono `n\_init=10`, che imposta che l’algoritmo venga eseguito 10 volte con diversi centroidi iniziali per selezionare la migliore inizializzazione, `init='random'` che comporta la scelta casuale dei centroidi iniziali tra i punti dati e `random\_state=42`, che garantisce la riproducibilità dei risultati.

Per la visualizzazione, vengono create rappresentazioni sia bidimensionali che tridimensionali utilizzando le prime due o tre componenti principali derivate dalla PCA.

Ultimo passaggio è l’analisi dei cluster, attraverso la funzione `analyze\_clusters()`, che calcola e presenta le caratteristiche medie di ciascun cluster, identifica le feature più importanti per ogni gruppo e fornisce intuizioni preziose sulle possibili sottocategorie di tumori al seno.

Immagine che contiene testo, schermata, software

Descrizione generata automaticamente

## Valutazione

La valutazione del clustering K-means si basa principalmente sull'analisi qualitativa dei risultati e rivela diversi aspetti interessanti, di seguito affrontati.

L’analisi del grafico del Bayesian Information Criterion (BIC) in funzione del numero di cluster, dimostra che il punto di minimo globale, che rappresenta il miglior compromesso tra complessità del modello e adattamento ai dati, si verifica per k=4. Questo indica che la suddivisione ottimale dei dati prevede quattro cluster distinti.

Immagine che contiene testo, diagramma, linea, Diagramma

Descrizione generata automaticamente

Le visualizzazioni 2D e 3D dei cluster permettono di osservare una buona separazione tra i cluster, specialmente tra il cluster verde (in basso a sinistra) e il cluster viola (sparso principalmente nella parte destra del grafico). I cluster giallo e blu mostrano una certa sovrapposizione, suggerendo caratteristiche simili tra alcuni casi di questi gruppi.

Immagine che contiene testo, schermata, diagramma, Policromia

Descrizione generata automaticamente Immagine che contiene diagramma, testo, schermata, Piano

Descrizione generata automaticamente

La visualizzazione 3D (Image 3) offre un'ulteriore dimensione per l'analisi, confermando la separazione osservata in 2D e mostrando come i cluster si distribuiscono nello spazio tridimensionale. Questa rappresentazione evidenzia ancora meglio la dispersione del cluster viola e la relativa compattezza del cluster verde, indicando che questi sono probabilmente outliers, cioè dati che si discostano significativamente dai cluster principali e non appartengono chiaramente a nessuno dei gruppi individuati dall'algoritmo di clustering.

L'output testuale, inoltre, fornisce informazioni dettagliate sulle caratteristiche medie di ciascun cluster accompagnata da un'interpretazione fornita per ciascun cluster, nel rispetto della coerenza con le caratteristiche osservate, offrendo una categorizzazione intuitiva dei casi di tumore basata principalmente sulle dimensioni e sulla potenziale aggressività:

1. Cluster 0 (skyblue): Caratterizzato da tumori di grandi dimensioni, con i valori più alti per area\_worst, area\_mean, e perimeter\_worst. Questo cluster potrebbe rappresentare i casi più critici e potenzialmente maligni.

Immagine che contiene testo, schermata, Carattere

Descrizione generata automaticamente

1. Cluster 1 (lightcoral): Rappresenta tumori di dimensioni medie, con valori intermedi per le caratteristiche principali.

Immagine che contiene testo, schermata, Carattere

Descrizione generata automaticamente

1. Cluster 2 (yellow): Identifica tumori di dimensioni più piccole, con i valori più bassi per area e perimetro. Questi potrebbero essere i casi meno aggressivi o in fase iniziale.

Immagine che contiene testo, schermata, Carattere

Descrizione generata automaticamente

1. Cluster 3 (purple): Descrive tumori di dimensioni medio-grandi, con valori intermedi tra il cluster 0 e il cluster 1.

Immagine che contiene testo, schermata, Carattere

Descrizione generata automaticamente

In conclusione, l'approccio di clustering non supervisionato ha prodotto risultati robusti e interpretabili, fornendo una suddivisione dei casi di tumore al seno in quattro categorie distinte che potrebbero avere rilevanza clinica significativa. La chiara separazione visiva dei cluster, combinata con le differenze significative nelle caratteristiche medie tra i gruppi, suggerisce che il clustering K-means ha identificato con successo sottogruppi distinti all'interno dei dati.

# Ragionamento probabilistico

## Sommario

Ultima fase del caso di studio è il ragionamento probabilistico, implementato nel sistema attraverso l'uso di una Rete Bayesiana (BN). Questa scelta di rappresentazione della conoscenza permette di modellare esplicitamente le relazioni probabilistiche tra le varie caratteristiche del tumore e la sua malignità. La BN fornisce un framework potente per gestire l'incertezza intrinseca nel dominio medico, consentendo inferenze robuste anche in presenza di informazioni incomplete o incerte.

## Strumenti utilizzati

Il sistema utilizza la libreria pgmpy per l'implementazione della Rete Bayesiana. Questa libreria offre un set completo di strumenti per la creazione, l'addestramento e l'inferenza con modelli grafici probabilistici.

Componenti chiave di pgmpy utilizzati:

- BayesianNetwork: usato per rappresentare e manipolare reti bayesiane, che sono grafi diretti aciclici (DAG) in cui i nodi rappresentano variabili casuali e gli archi rappresentano dipendenze condizionali tra queste variabili.

- HillClimbSearch e K2Score: l'apprendimento della struttura di una rete bayesiana può essere fatto in modo automatico dai dati, utilizzando algoritmi come la ricerca greedy con Hill Climbing e metriche di scoring come K2.

*L’HillClimbSearch* è un algoritmo euristico per apprendere la struttura di una rete bayesiana dai dati. L'idea è quella di partire da una rete casuale o vuota e iterativamente modificare la struttura (aggiungendo, rimuovendo o invertendo archi) in modo da migliorare il punteggio della struttura. In ogni passo, si esplora lo spazio delle strutture vicine alla rete corrente, scegliendo quella che porta al miglioramento maggiore in termini di punteggio.

Il *K2Score*, invece, è una metrica di scoring basata su un'ipotesi bayesiana. Si utilizza per confrontare diverse strutture di rete durante il processo di apprendimento, assegnando un punteggio più alto a quelle reti che meglio spiegano i dati. Assume che l'ordine delle variabili sia noto a priori e che non ci siano cicli.

- BayesianEstimator: è uno dei metodi per stimare tali parametri, ovvero le probabilità condizionali associate a ciascuna variabile data i suoi genitori nella rete, una volta che la struttura della rete è stata definita.

- VariableElimination: una volta che la struttura della rete e i parametri sono stati stimati, è possibile eseguire l'inferenza probabilistica. VariableElimination è un algoritmo esatto di inferenza usato in reti bayesiane. L’inferenza esatta utilizza algoritmi come l'eliminazione delle variabili per calcolare esattamente le probabilità di interesse. È particolarmente adatta per reti di dimensioni moderate, dove possiamo gestire la complessità computazionale.

L'algoritmo lavora eliminando iterativamente variabili non osservate e sommando sulle loro possibili configurazioni. Questo riduce il problema di calcolare probabilità condizionali a un insieme più semplice di operazioni.

## Decisioni di Progetto

Prima fase è il preprocessamento dei dati, in cui si sono utilizzate SelectKBest con il test statistico F-ANOVA per selezionare le k caratteristiche più rilevanti (k è impostato al minimo tra 10 e il numero totale di caratteristiche) e si sono poi discretizzate le caratteristiche continue attraverso KBinsDiscretizer con 3 bin e strategia uniforme.

Si utilizza l'algoritmo Hill-Climbing con lo score K2 per apprendere la struttura della rete, selezionando come parametri “max\_indegree=3”, che limita la complessità della rete e “max\_iter=int(1e4)”, che assicura una ricerca approfondita nello spazio delle strutture.

Viene poi impiegato il BayesianEstimator con prior di tipo "BDeu" (Bayesian Dirichlet equivalent uniform) e equivalent\_sample\_size=10, bilanciando l'influenza dei dati e del prior.

Lo stimatore bayesiano utilizza la probabilità a posteriori per stimare i parametri, integrando le informazioni dei dati con un priore. Questo approccio è particolarmente utile quando abbiamo pochi dati o vogliamo incorporare conoscenza a priori. Un priore comune per la distribuzione delle probabilità condizionali è proprio la distribuzione di Dirichlet, che assegna probabilità a priori alle distribuzioni categoriche. Questo è utile per evitare che le stime delle probabilità vadano a zero per eventi rari nei dati.

Alla visualizzazione della rete segue, come detto, l’utilizzo di VariableElimination per eseguire query probabilistiche sulla rete, che dimostra l'utilità della rete eseguendo diverse inferenze significative:

1. Calcolo della probabilità di cancro maligno date certe caratteristiche del tumore.
2. Identificazione delle caratteristiche più influenti sulla malignità.
3. Analisi di sensibilità per valutare l'impatto di ciascuna caratteristica sulla probabilità di malignità.
4. Inferenza su caratteristiche nascoste, dimostrando la capacità della rete di gestire informazioni mancanti.

## Valutazione

Segue l'analisi della Rete Bayesiana. Si presenta prima di tutto la struttura della rete Bayesiana appresa dai dati.

Immagine che contiene diagramma, linea

Descrizione generata automaticamente

Dalla rappresentazione grafica deduciamo che la variabile target (diagnosi) è direttamente collegata a diverse caratteristiche del tumore, tra cui radius\_worst, perimeter\_worst, e concave points\_worst/mean, che esistono relazioni complesse tra le varie caratteristiche, con molti nodi interconnessi, e che il nodo semantic\_combined\_risk\_factor è ben integrato nella rete, suggerendo che questa caratteristica derivata dall'integrazione semantica ha relazioni significative con altre variabili.

Lo Scenario 1 dimostra la capacità della rete di fornire probabilità di diagnosi basate su evidenze parziali:

Immagine che contiene testo, schermata, Carattere, software

Descrizione generata automaticamente

Per tumori con caratteristiche di grandi dimensioni (Scenario 1a), la rete stima una probabilità del 93.50% di malignità.Per tumori con caratteristiche di piccole dimensioni (Scenario 1b), la probabilità di benignità sale all'89.04%. Questi risultati sono coerenti con le aspettative cliniche e dimostrano la capacità della rete di catturare relazioni significative tra le caratteristiche del tumore e la sua natura.

Analisi dell'influenza delle caratteristiche: Lo Scenario 2 quantifica l'impatto di ciascuna caratteristica sulla diagnosi:

Immagine che contiene testo, schermata, Carattere, design

Descrizione generata automaticamente

concave points\_worst emerge come la caratteristica più influente, seguita da radius\_worst e perimeter\_mean. Inoltre, semantic\_combined\_risk\_factor, pur essendo influente, non è tra le caratteristiche top, suggerendo che le caratteristiche morfologiche dirette hanno un impatto maggiore sulla diagnosi.

Lo Scenario 3 mostra come varia la probabilità di diagnosi al variare di ciascuna caratteristica:

Immagine che contiene testo, schermata, Carattere, design

Descrizione generata automaticamente

concave points\_worst mostra la più ampia gamma di influenza, confermando la sua importanza. Caratteristiche come area\_mean e semantic\_combined\_risk\_factor mostrano un'influenza positiva più marcata rispetto a quella negativa, suggerendo che valori elevati di queste caratteristiche aumentano significativamente la probabilità di malignità.

Scenario 4 dimostra la capacità della rete di inferire valori probabili per caratteristiche non osservate:

Immagine che contiene testo, schermata, Carattere, numero

Descrizione generata automaticamente

Data l'evidenza di grandi dimensioni del tumore, la rete inferisce una probabilità del 67.65% che la concavity\_mean sia alta (valore 2.0). La rete dimostra dunque di essere robusta, fornendo inferenze coerenti in diversi scenari. Le relazioni catturate dalla rete sono interpretabili e allineate con la conoscenza medica sul cancro al seno.

In conclusione, la Rete Bayesiana implementata si dimostra uno strumento potente per il ragionamento probabilistico nel contesto della diagnosi del cancro al seno. Essa cattura efficacemente le relazioni complesse tra le caratteristiche del tumore e fornisce inferenze probabilistiche interpretabili e clinicamente rilevanti.

# Conclusioni

Il progetto ha dimostrato come l'applicazione di tecniche avanzate di machine learning e ingegneria della conoscenza possa migliorare la capacità predittiva e la comprensione delle malattie cardiache. L'uso combinato di integrazione semantica, modelli di classificazione, clustering e reti bayesiane ha permesso di ottenere risultati robusti e di fornire informazioni preziose per la previsione della malignità del cancro al seno

Tuttavia, alcune sfide rimangono, tra cui la gestione della complessità computazionale dei e la riduzione dell’overfitting, un’inferenza più complessa sulle reti bayesiane, nonché la necessità di una migliore interpretazione dei risultati del clustering.

# Riferimenti Bibliografici

**[1]** Poole, D. L., & Mackworth, A. K. (2023). Artificial Intelligence: Foundations of Computational Agents (3rd ed.). Cambridge University Press.