

Smoking study

Studio e analisi dei dati di soggetti fumatori e non

Natasha Fabrizio - Matricola: 717446

Email: *n.fabrizio@studenti.uniba.it*

Francesco Saverio Cassano - Matricola: 716133

Email: *f.cassano45@studenti.uniba.it*

Progetto di Ingegneria della Conoscenza

2021-2022

Indice

1	Introduzione	3
2	Requisiti funzionali	3
2.1	Liberie utilizzate	3
2.2	Installazione e avvio	3
3	Dataset	4
3.1	Preprocessing del dataset	4
3.1.1	Feature dicotomiche	4
3.1.2	Features presenti nel dataset	4
3.2	Panoramica dei dati	5
3.3	Bilanciamento delle classi	6
4	Apprendimento Supervisionato	7
4.1	Scelta del modello	7
4.2	Verifica dell'importanza delle features	10
4.3	Creazione della rete bayesiana	11
4.3.1	Calcolo della probabilità	12
4.4	Interfaccia per l'interazione dell'utente con la knowledge base	16

1 Introduzione

Il sistema è in grado di prevedere se un soggetto è potenzialmente un fumatore o meno, a seconda dei valori riscontrati nel dataset preso in considerazione.

Inoltre, l'utilizzatore del programma potrà inserire dei valori, (non necessariamente tutti) inerenti alle analisi, per poter comprendere se risulta un soggetto potenzialmente fumatore o meno; in caso affermativo, potrà ulteriormente decidere se ricevere o meno, un suggerimento su quali valori migliorare per non risultare più un fumatore.

2 Requisiti funzionali

La realizzazione del progetto è stata effettuata interamente in Python in quanto, tale linguaggio, risulta il più idoneo per la trattazione e analisi di dati; inoltre, è stato utilizzando come ambiente di lavoro l'IDE PyCharm 2022.

2.1 Liberie utilizzate

Le librerie Python utilizzate nel progetto, sono le seguenti:

- **Matplotlib**: usata per la visualizzazione di tutti i grafici, presenti nel progetto.
- **Numpy**: usata per la visualizzazione di tutti i grafici, presenti nel progetto.
- **Pandas**: usata per l'importazione del Dataset in formato ".csv".
- **Pgmpy**: usata per la creazione della rete bayesiana..
- **Scikit-learn**: usata per applicare i concetti del Machine Learning.
- **Warnings**: usata per la gestione dei messaggi di warnings del sistema.

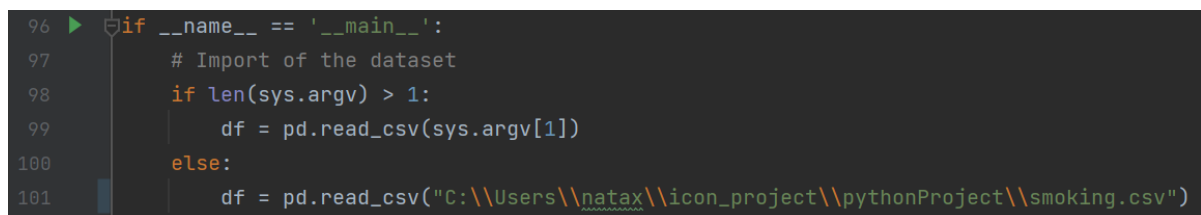
2.2 Installazione e avvio

Per poter installare, correttamente, le librerie utilizzate:

- aprire il terminale e navigare fino alla cartella dove è presente il file *"main.py"*
- inserire il seguente comando : *"pip install -r requirements.txt"*.

Per l'avvio del programma, sarà possibile trascinare il file *"smoking.csv"* sopra il file *"main.py"* oppure attraverso l'utilizzo del terminale, quest'ultimo aperto sul percorso dove è presente il file *"main.py"* e successivamente, eseguire il comando *"python main.py smoking.csv"*.

Eventualmente, è possibile modificare il percorso di lettura del file .csv dal file *"main.py"*, come segue da immagine:



```
96 if __name__ == '__main__':
97     # Import of the dataset
98     if len(sys.argv) > 1:
99         df = pd.read_csv(sys.argv[1])
100     else:
101         df = pd.read_csv("C:\\Users\\natax\\icon_project\\pythonProject\\smoking.csv")
```

Figura 1: Modifica del percorso di lettura del file csv a riga 101

3 Dataset

Il dataset utilizzato, "*Body signal of smoking*", consiste in una raccolta di dati di segnali biologici sanitari di base. L'obiettivo è quello di determinare la presenza o l'assenza del fumo attraverso segnali biologici. Link sorgente: [clicca qui](#)

Display (partial) of the dataframe:										
	gender	age	height(cm)	weight(kg)	waist(cm)	eyesight(left)	eyesight(right)	hearing(left)	hearing(right)	
0	1	40	155	60	81.3	1.2	1.0	1.0	1.0	
1	1	40	160	60	81.0	0.8	0.6	1.0	1.0	
2	0	55	170	60	80.0	0.8	0.8	1.0	1.0	
3	0	40	165	70	88.0	1.5	1.5	1.0	1.0	
4	1	40	155	60	86.0	1.0	1.0	1.0	1.0	
	systolic	relaxation	fasting blood sugar	Cholesterol	triglyceride	HDL	LDL			
	114.0	73.0	94.0	215.0	82.0	73.0	126.0			
	119.0	70.0	130.0	192.0	115.0	42.0	127.0			
	138.0	86.0	89.0	242.0	182.0	55.0	151.0			
	100.0	60.0	96.0	322.0	254.0	45.0	226.0			
	120.0	74.0	80.0	184.0	74.0	62.0	107.0			
	hemoglobin	Urine protein	serum creatinine	AST	ALT	Gtp	oral	dental caries	tartar	smoking
	12.9	1.0	0.7	18.0	19.0	27.0	1	0	1	0
	12.7	1.0	0.6	22.0	19.0	18.0	1	0	1	0
	15.8	1.0	1.0	21.0	16.0	22.0	1	0	0	1
	14.7	1.0	1.0	19.0	26.0	18.0	1	0	1	0
	12.5	1.0	0.6	16.0	14.0	22.0	1	0	0	0

Figura 2: Anteprima del dataset

3.1 Preprocessing del dataset

Nella fase di preprocessing, il dataset viene modificato in modo da poterlo utilizzare correttamente. Il dataset non presenta problemi di mancanza dei dati.

3.1.1 Feature dicotomiche

Una feature dicotomica è una feature che presenta soltanto due modalità. Sono dicotomiche feature con valori come {si, no}, oppure {vero, falso}. In questo caso verranno assegnati i valori {0,1} rispettivamente alle due modalità, verrà sostituito il valore iniziale della modalità con il nuovo valore assegnato. Nel nostro caso, le features dicotomiche sono:

- **tartar**: {vero, falso} in {0,1}
- **oral**: {vero, falso} in {0,1}
- **gender**: {vero, falso} in {0,1}

3.1.2 Features presenti nel dataset

Per ridurre la complessità, si è scelto di eliminare la feature *ID*, presente nel dataset, in quanto ha significatività bassa o tendendete a 0, essendo un semplice indice del dataset. Qui sotto sono elencante le features principali del dataset dopo la features selection:

- **gender**
- **age** : 5 anni di intervallo
- **height(cm)** : 5 anni di intervallo
- **weight(kg)** : 5 anni di intervallo
- **waist(cm)** : lunghezza circonferenza vita
- **eyesight(left)**

- eyesight(right)
- hearing(left)
- hearing(right)
- systolic : pressione sanguigna
- relaxation : pressione sanguigna
- fasting blood sugar
- cholesterol : totale
- triglyceride
- HDL : tipo di colesterolo
- LDL : tipo di colesterolo
- hemoglobin
- urine protein
- serum creatinine
- AST : tipo glutammico ossalacetico transaminasi
- ALT : tipo glutammico ossalacetico transaminasi
- Gtp
- oral : stato dell'esame orale
- dental caries
- tartar : stato tartaro
- smoking

3.2 Panoramica dei dati

Mediante la stampa di una tabella, viene mostrato a schermo una panoramica inerente alle info del dataset.

Info dataset:									
	gender	age	height(cm)	weight(kg)	waist(cm)	eyesight(left)	eyesight(right)	hearing(left)	
count	7001.000000	7001.000000	7001.000000	7001.000000	7001.000000	7001.000000	7001.000000	7001.000000	
mean	0.371661	44.213684	164.545779	65.735609	81.998829	1.008256	1.011827	1.024139	
std	0.483283	12.115199	9.272506	12.829996	9.175379	0.414061	0.514026	0.153493	
min	0.000000	20.000000	135.000000	30.000000	51.000000	0.100000	0.100000	1.000000	
25%	0.000000	40.000000	160.000000	55.000000	76.000000	0.800000	0.800000	1.000000	
50%	0.000000	40.000000	165.000000	65.000000	82.000000	1.000000	1.000000	1.000000	
75%	1.000000	55.000000	170.000000	75.000000	88.000000	1.200000	1.200000	1.000000	
max	1.000000	85.000000	190.000000	120.000000	116.000000	9.900000	9.900000	2.000000	

hearing(right)	systolic	relaxation	fasting blood sugar	Cholesterol	triglyceride	HDL	LDL	hemoglobin	
7001.000000	7001.000000	7001.000000	7001.000000	7001.000000	7001.000000	7001.000000	7001.000000	7001.000000	
1.028996	121.535495	75.915441	99.206113	197.27039	126.520354	57.32681	115.063785	14.610113	
0.167887	13.599619	9.611540	21.087883	36.14813	71.402842	14.36465	35.440412	1.566799	
1.000000	82.000000	49.000000	51.000000	96.00000	19.000000	18.00000	9.000000	4.900000	
1.000000	112.000000	70.000000	89.000000	172.00000	74.000000	47.00000	92.000000	13.500000	
1.000000	120.000000	76.000000	96.000000	105.00000	108.000000	55.00000	113.000000	14.800000	
1.000000	130.000000	82.000000	103.000000	220.00000	160.000000	66.00000	136.000000	15.700000	
2.000000	220.000000	134.000000	475.000000	373.00000	999.000000	128.00000	910.000000	19.300000	

urine protein	serum creatinine	AST	ALT	Gtp	oral	dental caries	tartar	smoking	
7001.000000	7001.000000	7001.000000	7001.000000	7001.000000	7001.0	7001.000000	7001.000000	7001.000000	
1.088702	0.885588	26.110127	27.011141	39.515069	1.0	0.210256	0.555778	0.361091	
0.409166	0.230885	23.084120	37.956003	49.434818	0.0	0.407519	0.496915	0.480351	
1.000000	0.100000	6.000000	1.000000	6.000000	1.0	0.000000	0.000000	0.000000	
1.000000	0.700000	19.000000	15.000000	17.000000	1.0	0.000000	0.000000	0.000000	
1.000000	0.900000	23.000000	21.000000	25.000000	1.0	0.000000	1.000000	0.000000	
1.000000	1.000000	28.000000	30.000000	43.000000	1.0	0.000000	1.000000	1.000000	
6.000000	10.300000	1311.000000	2062.000000	999.000000	1.0	1.000000	1.000000	1.000000	

Figura 3: Informazioni inerenti al dataset

3.3 Bilanciamento delle classi

Attraverso un grafico, verifichiamo se il dataset sia ben bilanciato, in modo da riuscire ad avere ottimi risultati durante l'apprendimento.

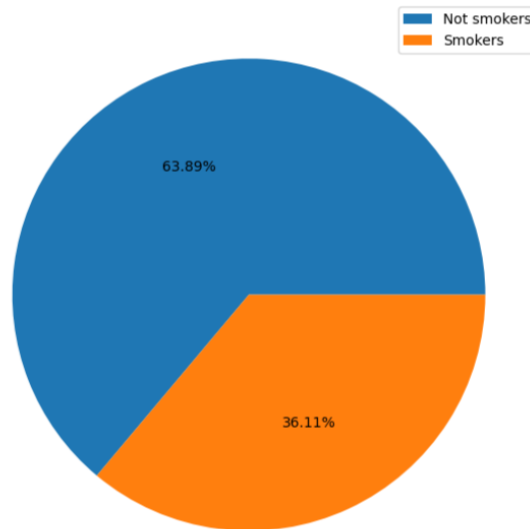


Figura 4: Grafico di occorrenze di fumatori e non fumatori

Dal grafico è possibile notare uno sbilanciamento delle classi.

Classi squilibrate mettono fuori gioco la *"precisione"*. Questo è un problema sorprendentemente comune nell'apprendimento automatico (in particolare nella classificazione), che si verifica in set di dati con un rapporto sproporzionato di osservazioni in ciascuna classe.

La precisione standard non misura più in modo affidabile le prestazioni, il che rende l'addestramento del modello molto più complicato.

Vi sono diversi modi per poter risolvere il problema dello sbilanciamento delle classi. La soluzione per cui abbiamo optato è quella di utilizzare l'*oversampling*. Per far ciò abbiamo individuato la classe di maggioranza e di minoranza ed abbiamo effettuato un resampling facendo combaciare le occorrenze. Ottenendo, così, classi bilanciate:

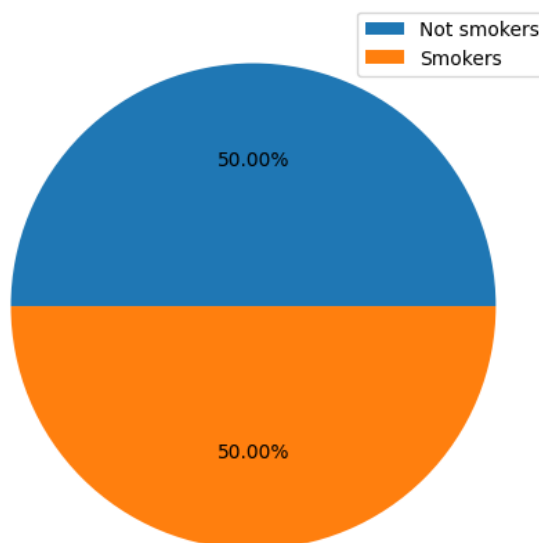


Figura 5: Grafico di occorrenze di fumatori e non fumatori dopo oversampling

```
Not smokers: 4473 (% 63.89)
Smokers: 2528 (% 36.11)

Value after Oversampling:

Not smokers: 4473 (% 50.00)
Smokers: 4473 (% 50.00)
```

Figura 6: Bilanciamento classi

4 Apprendimento Supervisionato

4.1 Scelta del modello

Per questo tipo di apprendimento abbiamo usato vari modelli per poi identificare quale fosse quello più adatto al nostro dataset.

I modelli valutati sono:

- **KNN (K-Nearest Neighbors)**
 - Il K-Nearest Neighbors, è un algoritmo utilizzato nel riconoscimento di pattern per la classificazione di oggetti basandosi sulle caratteristiche degli oggetti vicini a quello considerato.
- **Decision Tree**
 - Il Decision Tree è un classificatore con struttura ad albero (alberi di decisione), in cui ogni nodo può essere o foglia o nodo interno: se foglia, indica il valore della classe assegnata all'istanza; se nodo interno, specifica il test effettuato su un attributo. Per ciascun valore assunto da un attributo in un test, l'algoritmo crea un ramo e il relativo sottoalbero.
- **Random Forest**
 - Il Random Forest è un classificatore d'insieme ottenuto dall'aggregazione tramite bagging di alberi di decisione. Esso si pone come soluzione che minimizza l'overfitting del training set rispetto agli alberi di decisione.
- **SVC (Support-Vector Classification)**
 - SVC è un modello di apprendimento per la regressione e la classificazione. Dato un insieme di esempi per l'addestramento, ognuno dei quali etichettato con la classe di appartenenza fra le due possibili classi, un algoritmo di addestramento per le SVC costruisce un modello che assegna i nuovi esempi a una delle due classi, ottenendo quindi un classificatore lineare binario non probabilistico.

Classificatori Naïve Bayes

- **BernoulliNB (Bernoulli Naïve Bayes)**

- Questo classificatore è simile al multinomiale naive bayes ma i predittori sono variabili booleane. I parametri che usiamo per prevedere la variabile di classe occupano solo i valori sì o no.

- **GaussianNB (Gaussian Naive Bayes)**

- Gaussian Naive Bayes è una variante di Naive Bayes che segue la distribuzione normale gaussiana e supporta dati continui.

Sui suddetti abbiamo eseguito il **K-Fold Cross Validation**, per verificare quale di esse fosse il più attendibile. In particolare, il **RepeatedKFold**, con 5 ripetizioni.

Inoltre, le metriche di performance utilizzate nella valutazione sono:

- accuratezza
- precisione
- richiamo
- F1-score

Segue che i risultati ottenuti dalla valutazione sono:

	model	accuracy	precision	recall	f1score
0	KNN	0.732253	0.710351	0.811280	0.757468
1	DecisionTree	0.836221	0.801534	0.906725	0.757468
2	RandomForest	0.877585	0.831917	0.955531	0.889450
3	SVM	0.750699	0.723684	0.835141	0.775428
4	BernoulliNB	0.553382	0.555157	0.671367	0.607757
5	GaussianNB	0.664058	0.727016	0.557484	0.631062

Figura 7: Performance classificatori (Eseguiti 5 ri-avvi con piccole differenze di 1-2%)

In particolare:

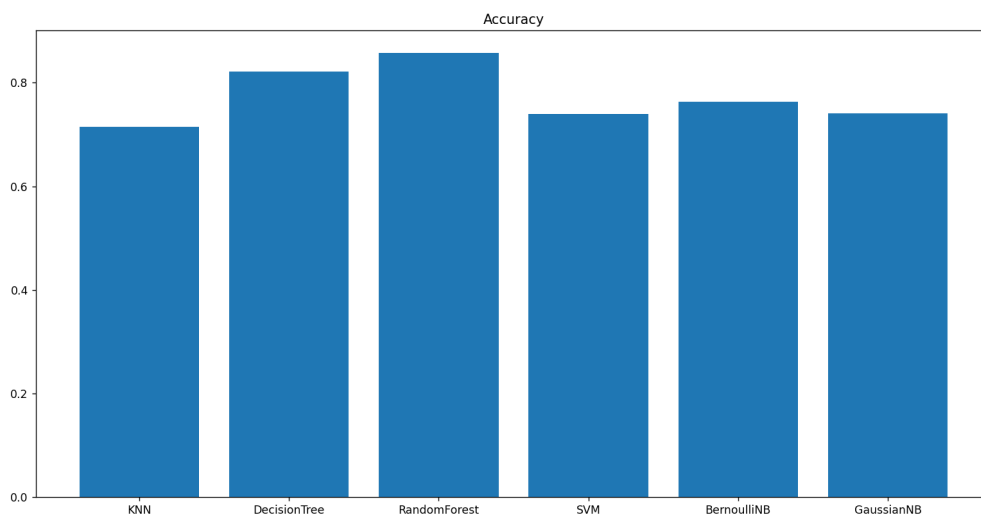


Figura 8: Grafico metrica accuratezza

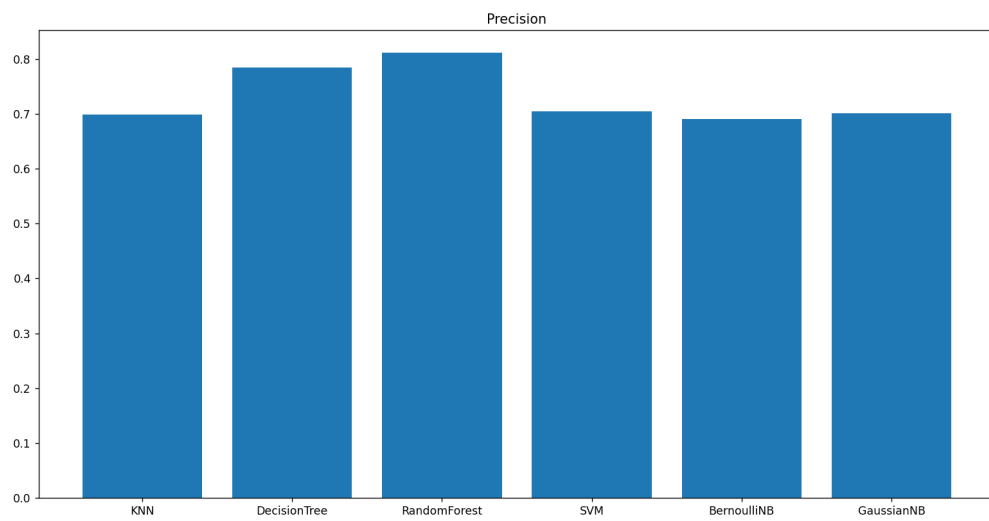


Figura 9: Grafico metrica precisione

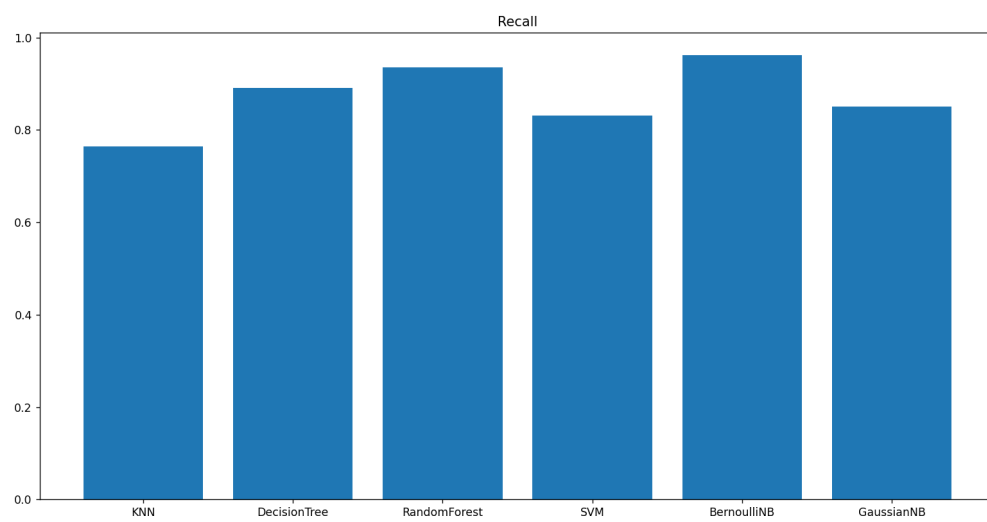


Figura 10: Grafico metrica richiamo

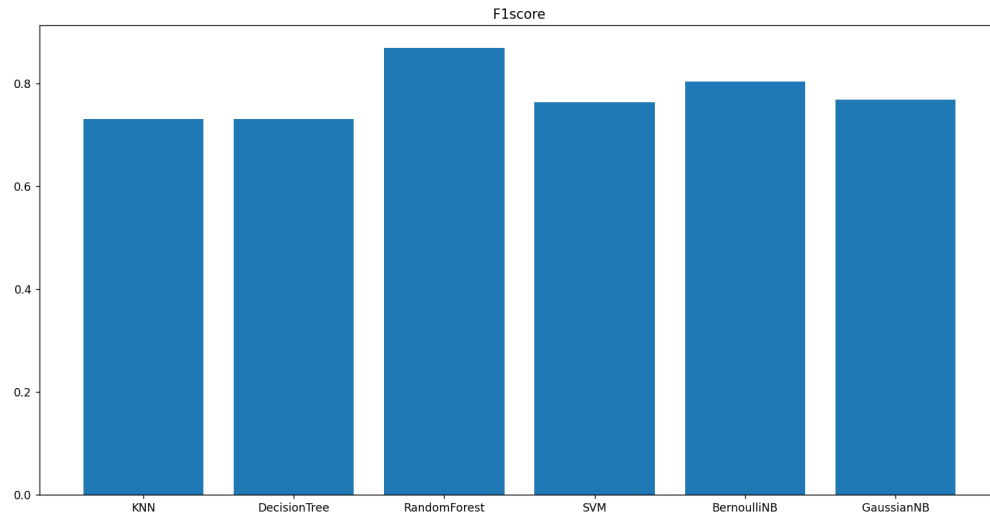


Figura 11: Grafico metrica F1-score

Basandoci su tali performance, in particolare su quella del **F1-score**, abbiamo riscontrato che il classificatore migliore risulta essere il Random Forest.

4.2 Verifica dell'importanza delle features

A seguito dell'analisi effettuata in precedenza, abbiamo generato un grafico che estrae le features più importanti derivanti proprio dal Random Forest:

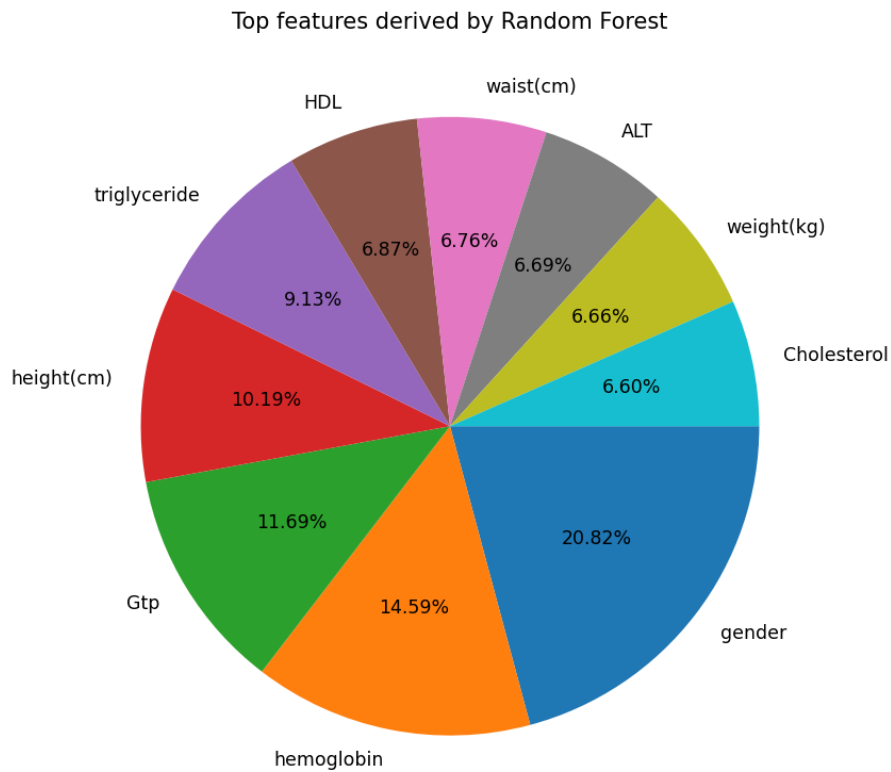


Figura 12: Risultati classificatore Random Forest

Possiamo notare che il genere, l'emoglobina e il gtp sono le caratteristiche mediche predittive più importanti per la diagnostica di un soggetto potenzialmente fumatore o meno.

4.3 Creazione della rete bayesiana

Abbiamo scelto di implementare una rete bayesiana per poter effettuare delle interrogazione per verificare le probabilità delle features; utilizzando come metodo di scoring il $K2score$ e come stimatore il *MaximumLikeliHoodEstimato*. In questo modo la predizione avverrà in base alle nostre features.

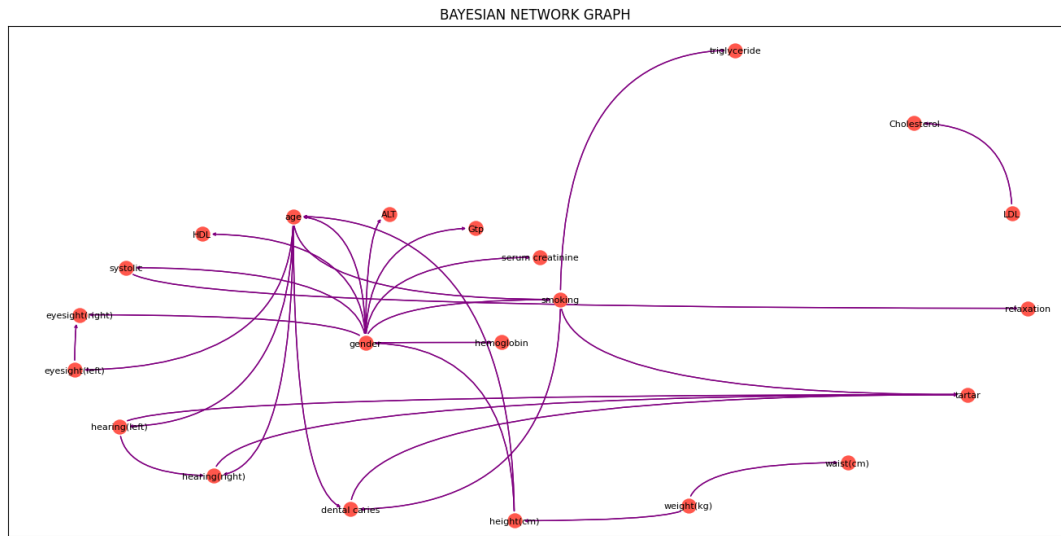


Figura 13: Grafo della rete bayesiana

4.3.1 Calcolo della probabilità

Sfruttando la rete bayesiana precedentemente creata, calcoliamo le probabilità per le features *gender* e *smoking*.

Gender Dalle osservazioni precedenti è emerso che la feature *gender* influenza diverse features, ragion per cui osserviamo con delle interrogazioni tali influenze.

Data una variabile X, solo alcune variabili influenzano direttamente il suo valore.

Le variabili che influenzano localmente sono dette *Markov Blanket*.

Le Markov blanket di *gender*, quindi, sono:

```
Markov blanket for "gender"
['Gtp', 'height(cm)', 'hemoglobin', 'age', 'HDL', 'serum creatinine', 'systolic', 'smoking', 'eyesight(right)', 'ALT', 'eyesight(left)']
```

Figura 14: Markov blanket di *gender*

Tenendo in considerazione le Markov blanket di *gender*, abbiamo effettuato interrogazioni sulla rete con un soggetto femminile ed uno maschile non fumatori.

Probability control of features that affect gender:
Test on a non-smoker female subject

gender	phi(gender)
gender(0)	0.0069
gender(1)	0.9931

Test on a non-smoker male subject

gender	phi(gender)
gender(0)	0.9840
gender(1)	0.0160

Figura 15: Probabilità ottenute

Per ottenere tali dati, abbiamo effettuato le seguenti query:

1. contenente i dati di un soggetto femminile;
2. contenente i dati di un soggetto maschile.

```
woman = data.query(show_progress=False, variables=['gender'], evidence={'age': 40, 'height(cm)': 160,
    'weight(kg)': 65, 'Gtp': 34, 'HDL': 55,
    'serum creatinine': 0, 'systolic': 126,
    'ALT': 15, 'eyesight(left)': 1,
    'eyesight(right)': 0, 'hemoglobin': 13,
    'smoking': 0})

man = data.query(show_progress=False, variables=['gender'], evidence={'age': 75, 'height(cm)': 160,
    'weight(kg)': 70, 'Gtp': 17, 'HDL': 71,
    'serum creatinine': 0, 'systolic': 128,
    'ALT': 16,
    'eyesight(left)': 0, 'eyesight(right)': 1,
    'hemoglobin': 15, 'smoking': 0})
```

Figura 16: Query effettuate

Abbiamo effettuato le query sopra indicate, considerando i soggetti, questa volta, come fumatori. Ottenendo come risultati:

Test on a smoker female subject		
gender	phi(gender)	
gender(0)	0.1414	
gender(1)	0.8586	

Test on a smoker male subject		
gender	phi(gender)	
gender(0)	1.0000	
gender(1)	0.0000	

Figura 17: Probabilità ottenute

Smoking Le Markov Blanket di *smoking*, invece, sono:

```
Markov blanket for "smoking"
['hearing(left)', 'age', 'triglyceride', 'dental caries', 'gender', 'hearing(right)', 'tartar']
```

Figura 18: Markov blanket di *smoking*

Sfruttando la rete bayesiana, calcoliamo la probabilità per un soggetto presumibilmente non fumatore (0) ed uno fumatore (1) di esserlo o meno. Il soggetto di genere femminile preso in analisi ha età, altezza e peso standard, ovvero età: 20, altezza: 170, peso: 60.

Probability for a potentially non-smoker:	
smoking	phi(smoking)
smoking(0)	0.6944
smoking(1)	0.3056

Probability for a potential smoker:	
smoking	phi(smoking)
smoking(0)	0.3400
smoking(1)	0.6600

Figura 19: Probabilità ottenute

Per ottenere tali dati, abbiamo effettuato due query:

1. contenente i dati di una persona non fumatrice;
2. contenente i dati di una persona fumatrice.

```
notSmoker = data.query(show_progress=False, variables=['smoking'],
                      evidence={'age': 20, 'height(cm)': 170, 'weight(kg)': 60, 'gender': 1, 'hearing(left)': 0,
                                'hearing(right)': 0, 'triglyceride': 113, 'dental caries': 0, 'tartar': 0})

smoker = data.query(show_progress=False, variables=['smoking'],
                    evidence={'age': 20, 'height(cm)': 170, 'weight(kg)': 60, 'gender': 1, 'triglyceride': 151,
                              'hearing(left)': 1, 'hearing(right)': 1, 'dental caries': 1, 'tartar': 1})
```

Figura 20: Query effettuate

Per verificare l'effettiva rilevanza delle features all'interno della rete bayesiana, abbiamo effettuato anche dei test sui soggetti presi in considerazione; riscontrando che le features che influenzano maggiormente risultano essere: *triglyceride*, *dental caries*, *tartar*. Ottenendo come probabilità:

Test on Subject potentially smoker:		
smoking	phi(smoking)	
smoking(0)	0.6328	
smoking(1)	0.3672	

Test on Potentially non-smoker subject:		
smoking	phi(smoking)	
smoking(0)	0.4076	
smoking(1)	0.5924	

Figura 21: Probabilità ottenute dai test

```
TestSmoker = data.query(show_progress=False, variables=['smoking'],
                        evidence={'age': 20, 'height(cm)': 170, 'weight(kg)': 60, 'gender': 1, 'triglyceride': 120,
                                'hearing(left)': 1, 'hearing(right)': 1, 'dental caries': 0, 'tartar': 0})
TestNotSmoker = data.query(show_progress=False, variables=['smoking'],
                           evidence={'age': 20, 'height(cm)': 170, 'weight(kg)': 60, 'gender': 1,
                                   'triglyceride': 148, 'hearing(left)': 1, 'hearing(right)': 1,
                                   'dental caries': 1, 'tartar': 1})
```

Figura 22: Query effettuate per i test

4.4 Interfaccia per l'interazione dell'utente con la knowledge base

```

Welcome to our system!

It allows you to predict whether, taken of the subjects, they are smokers or not.

Do you want to enter your data for a prediction? - Y/N? - (Typing 'n' close program)

```

Figura 23: Interazione con la knowledge base

L'utente può decidere di interagire con la knowledge base effettuando interrogazioni su quest'ultima. Per poterlo fare, bisogna inserire i dati relativi alla sua età, altezza, peso e genere obbligatoriamente (rispettando i valori ed i controlli segnalati, ove dovuti). Per i restanti valori, quali triglyceride, hearing(left), hearing(right), dental caries, tartar, l'utente se non è a conoscenza di uno di essi (o di tutti) può scegliere di non inserirli e quindi effettuare una interrogazione "parziale". Nell'eventualità che l'utente inserisca valori non validi o fuori range indicato, viene segnalato l'errore.


```

Do you want to enter your data for a prediction? - Y/N? - (Typing 'n' close program)
y
Please insert:
['age', 'height(cm)', 'weight(kg)', 'gender', 'hearing(left)', 'hearing(right)', 'triglyceride', 'dental caries', 'tartar']
Age - height(cm) - weight(kg) - gender are obligatory to enter!
The range of allowed values are multiples of 5
The minimum acceptable " age " value is: 20 The maximum is: 85
Insert age value:
20
The range of allowed values are multiples of 5
The minimum acceptable " height(cm) " value is: 135 The maximum is: 190
Insert height(cm) value:
170
The range of allowed values are multiples of 5
The minimum acceptable " weight(kg) " value is: 30 The maximum is: 120
Insert weight(kg) value:
60
Insert gender value (0 = Man, 1 = Woman)
0
Insert hearing(left) value (if you don't have the value, enter -1):
1
Insert hearing(right) value (if you don't have the value, enter -1):
1
Insert triglyceride value (if you don't have the value, enter -1):
131
Insert dental caries value (0 = No, 1 = Yes, -1 = Data not available):
1
Insert tartar value (0 = No, 1 = Yes, -1 = Data not available):
1

```

Figura 24: Inserimento dei valori

Successivamente, viene visualizzata la probabilità risultante dai valori impostati dall'utente.

```

+-----+-----+
| smoking | phi(smoking) |
+=====+=====+
| smoking(0) | 0.2016 |
+-----+-----+
| smoking(1) | 0.7984 |
+-----+-----+

```

Figura 25: Probabilità a seguito dei valori inseriti dall'utente

Nel caso in cui la probabilità di essere un fumatore dovesse risultare superiore al 50%, il sistema chiede all'utente se desidera ricevere suggerimenti, ottenuti mediante una predizione, di valori ai quali ambire per poter risultare un non fumatore; tali valori sono ottenuti partendo dai valori di età, altezza, peso e genere. Se il sistema dovesse trovare una combinazione valida, ovvero con probabilità inferiore al 50%, viene visualizzata come segue:

```

Want to know what values to improve not to be to no longer be considered a smoker? - Y/N
y
The search will last 60.0 seconds. If no ideal combinations are found, no data will be displayed.
Finding combination...
Suggested values:
height(cm) tartar age hearing(right) smoking triglyceride hearing(left) gender dental caries weight(kg)
0 170 1 20 1 0 59 1 0 0 60
New probability based on suggested values
+-----+-----+
| smoking | phi(smoking) |
+=====+=====+
| smoking(0) | 0.5693 |
+-----+-----+
| smoking(1) | 0.4307 |
+-----+-----+
Do you want to enter your data for a prediction? - Y/N? - (Typing 'n' close program)

```

Figura 26: Valori suggeriti e nuova probabilità

Riferimenti bibliografici

- [1] Kompremos. (2020). Classificatore Naive Bayes.
<https://kompremos.com/it/classificatore-naive-bayes/>
- [2] Elite Data Science. (2022). How to Handle Imbalanced Classes in Machine Learning.
<https://elitedatascience.com/imbalanced-classes>
- [3] D. Poole, A. Mackworth: Artificial Intelligence: Foundations of Computational Agents. Cambridge University Press. 2nd Ed. [Ch.8]
<http://artint.info/2e/html/ArtInt2e.Ch8.html>
- [4] Documentazione libreria *scikit-learn*.
https://scikit-learn.org/stable/user_guide.html
- [5] Documentazione libreria *pgmpy*.
<https://pgmpy.org/>