PROGETTO DI INGEGNERIA DELLA CONOSCENZA

Gruppo:

Abbattista Marianna MAT. 663721

Balestrucci Pier Felice MAT. 668705

Lanotte Michele MAT. 661569

Musti Luca MAT.666755

INDICE

INTRODUZIONE

Il progetto nasce con l'intento di predire se una paziente possa ripresentare in futuro un cancro al seno in base ad elementi classificanti la malattia. A tal proposito sono state adottate una serie di tecniche sia di apprendimento supervisionato sia non supervisionato. Inoltre, è stata modellata un'ontologia di dominio che offre una rappresentazione formale e concettualizzata della realtà presa in esame, affinché sia relazionabile con altre ontologie già esistenti e possa venire interrogata.

Il dataset(https://archive.ics.uci.edu/ml/datasets/Breast+Cancer) utilizzato prevede le seguenti features:

1. Class(target): no-recurrence-events, recurrence-events
2. age: 10-19, 20-29, 30-39, 40-49, 50-59, 60-69, 70-79, 80-89, 90-99.
3. menopause: lt40, ge40, premeno.
4. tumor-size: 0-4, 5-9, 10-14, 15-19, 20-24, 25-29, 30-34, 35-39, 40-44, 45-49, 50-54, 55-59.
5. inv-nodes: 0-2, 3-5, 6-8, 9-11, 12-14, 15-17, 18-20, 21-23, 24-26, 27-29, 30-32, 33-35, 36-39.
6. node-caps: yes, no.
7. deg-malig: 1, 2, 3.
8. breast: left, right.
9. breast-quad: left-up, left-low, right-up, right-low, central.
10. irradiat: yes, no.

2. APPRENDIMENTO SUPERVISIONATO

Con l'Apprendimento Supervisionato cerchiamo di costruire un modello partendo da dei dati di addestramento etichettati, con i quali cerchiamo di fare previsioni su dati non disponibili o futuri. Con *Supervisione* si intende quindi che nel nostro insieme dei campioni (o dataset), i segnali di output desiderati sono già noti poiché precedentemente etichettati. In questo tipo di apprendimento, basato su *etichette delle classi discrete*, avremo quindi un compito basato su tecniche di classificazione.

E' stato necessario usare un algoritmo che bilanciasse il dataset, perché la proporzione tra le due classi di esempi era circa del 2:1. Abbiamo utilizzato lo "SMOTE", che prevede un ridimensionamento della classe di esempi maggiore e minore. Nel KNN si mostra come questa applicazione abbia inferito positivamente sulla classificazione.

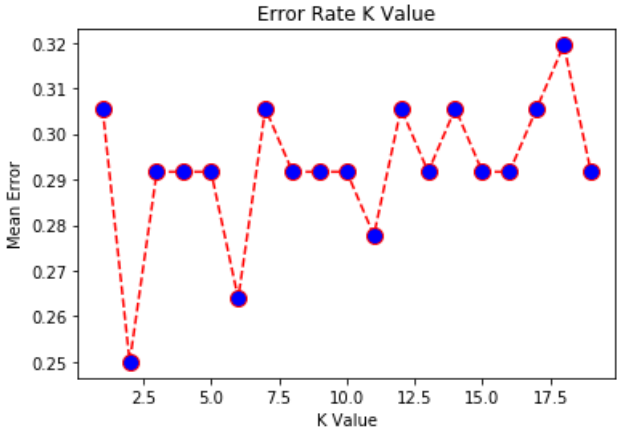
Per ogni algoritmo di apprendimento supervisionato sono stati prodotti i seguenti grafici:

- ROC Curve
- Precision-Recall Curve
- Bar Chart di varianza e deviazione standard
- Matrice di Confusione

Per la maggior parte degli algoritmi è stata usata la tecnica della cross-validation per rilevare possibili problemi di sovra-adattamento. A tal proposito si riportano i valori del punteggio medio (cross-val-score), della varianza, dev. Standard su cinque iterate.

2.1 K-nearest-neighbour

Il K-Nearest-neighbour è un algoritmo di apprendimento supervisionato che consiste nell'individuare i k esempi più vicini a quello che si intende classificare, a quest'ultimo viene quindi attribuita la categoria "più ricorrente" tra i k esempi più vicini.



Il grafico proposto di sopra, fornisce un suggerimento sulla scelta del numero di vicini per minimizzare l'errore medio. Si evince dal grafico che con numero di vicini = 2 l'errore medio minimo commesso è di 0.25.

Dalla classificazione dell'algoritmo sono stati prodotti i seguenti classification report pre e post applicazione dell'algoritmo SMOTE.

PRE-SMOTE

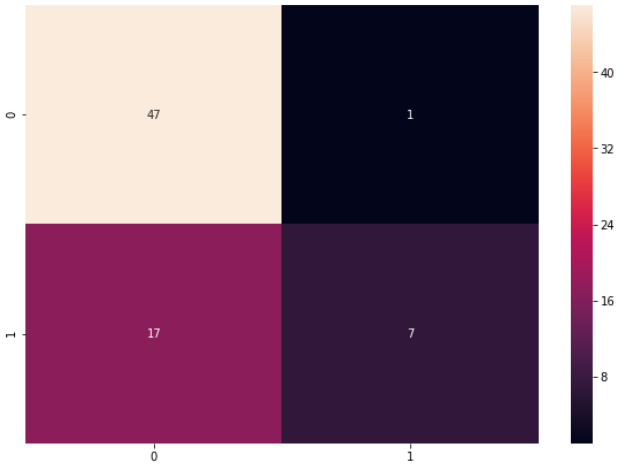
Clasification report:

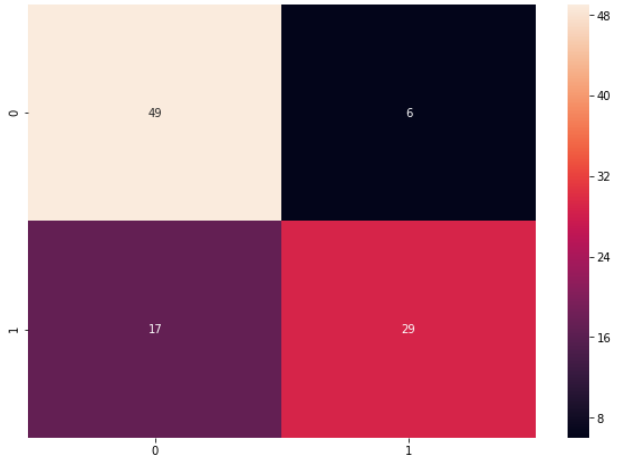
	precision	recall	f1-score	support
0	0.73	0.98	0.84	48
1	0.88	0.29	0.44	24
micro avg	0.75	0.75	0.75	72
macro avg	0.80	0.64	0.64	72
weighted avg	0.78	0.75	0.71	72

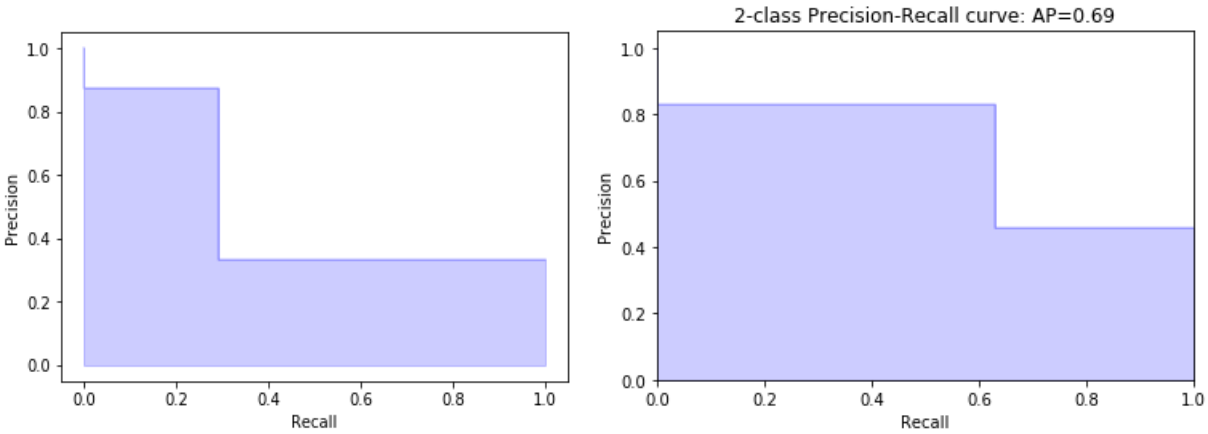
POST-SMOTE

Clasification report:

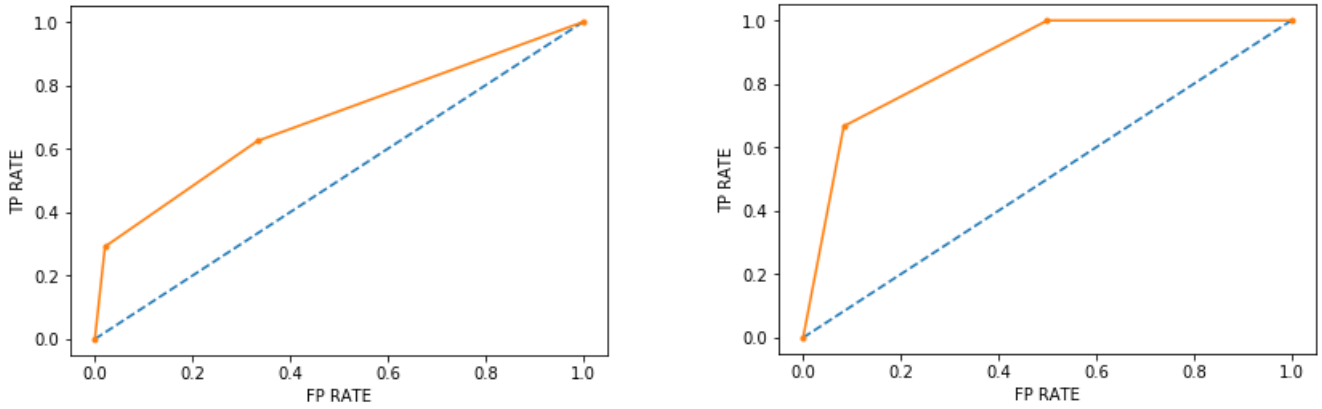
	precision	recall	f1-score	support
0	0.74	0.89	0.81	55
1	0.83	0.63	0.72	46
micro avg	0.77	0.77	0.77	101
macro avg	0.79	0.76	0.76	101
weighted avg	0.78	0.77	0.77	101



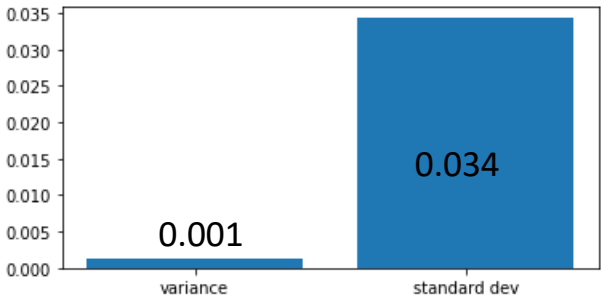


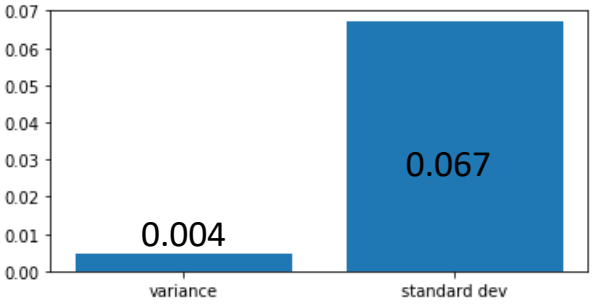


Analizzando i due grafici di precision-recall, si può notare come, a seguito dell'ottimizzazione del nostro dataset, vi è un netto miglioramento della precision all'aumentare della recall. Come conseguenza anche l'average-precision ne risente, infatti essa è pari 0.491 prima dell'ottimizzazione e subisce un incremento di +0.20 dopo l'ottimizzazione stessa. Inoltre l'accuracy è del 0.75.



La curva ROC (Receiver Operating Characteristics) dell'AUC (Area Under The Curve) è una delle metriche di valutazione più importanti per il controllo delle prestazioni di qualsiasi modello di classificazione. Il ROC è una curva di probabilità e l'AUC rappresenta la misura della separabilità. Indica quanti modelli è in grado di distinguere tra le classi. Maggiore è l'AUC, migliore è il modello nel predire 0 come 0 e 1 come 1. Nel nostro caso l'AUC è pari a 0.688 prima dell'ottimizzazione del dataset con l'algoritmo SMOTE, e subisce un incremento di +0.187 dopo l'ottimizzazione.





Per quanto riguarda la cross validation sul classificatore del KNN i dati ottenuti sono:

pre-SMOTE

cv_scores mean ---> 0.717
cv_score variance ---> 0.0011
cv_score dev standard ---> 0.034

post-SMOTE

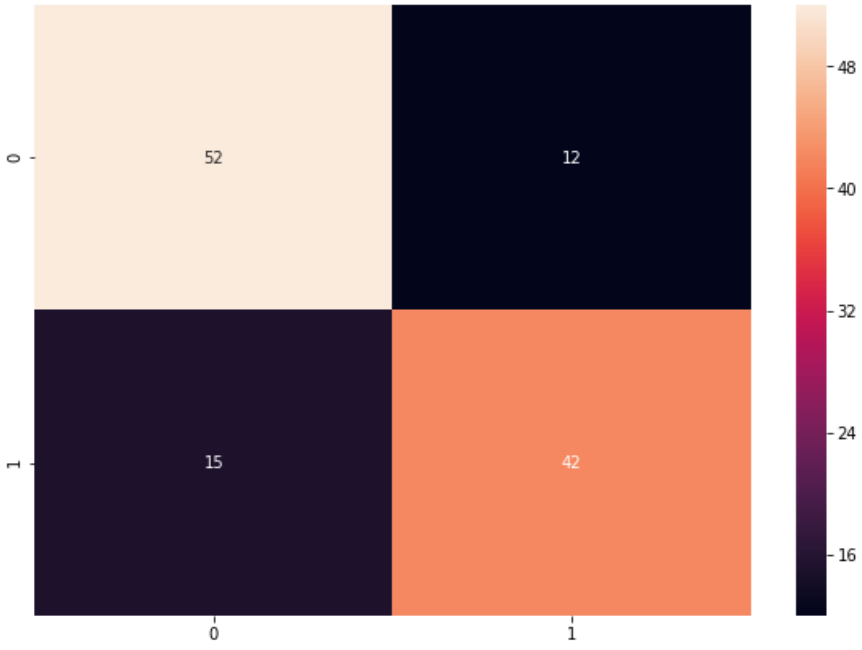
cv_scores mean ---> 0.774
cv_score variance ---> 0.0045
cv_score dev standard ---> 0.067

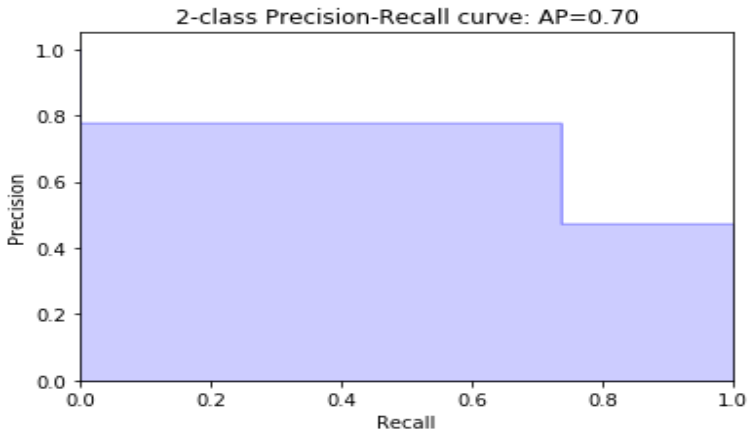
2.2 Random forest

Il random forest è un modello composito costituito da molti alberi di decisione, ognuno dei quali fornisce una predizione. Esse vengono poi, combinate allo scopo di ottenere una previsione complessiva della foresta per un dato esempio. La predizione di ciascun albero può essere ottenuta o, attraverso la media delle predizioni di un albero per ogni esempio o, attraverso un meccanismo di votazione in cui tutti gli alberi votano la propria classificazione più probabile e l'esempio col maggior numero di voti sarà scelto come predizione finale.

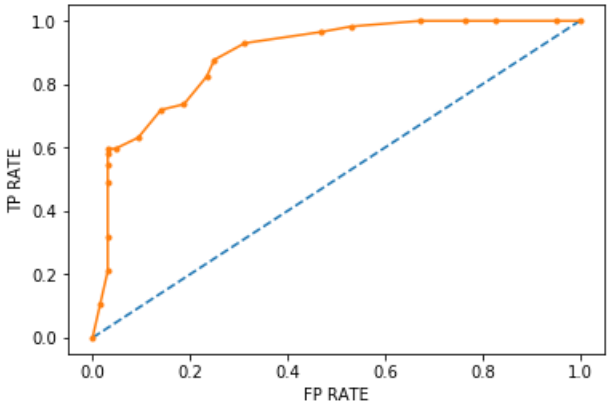
Clasification report:

	precision	recall	f1-score	support
0	0.78	0.81	0.79	64
1	0.78	0.74	0.76	57
micro avg	0.78	0.78	0.78	121
macro avg	0.78	0.77	0.78	121
weighted avg	0.78	0.78	0.78	121

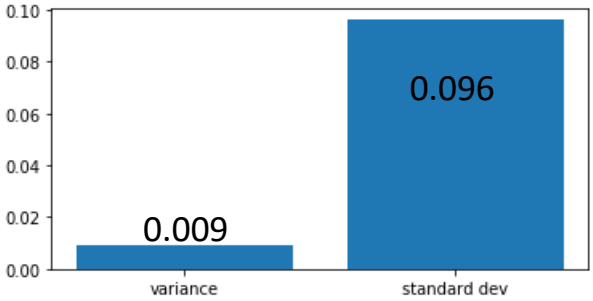




L' average precision è pari a 0.697 mentre l'accuratezza è uguale a 0.777.



L'algoritmo Random Forest, secondo la curva ROC, è in grado di differenziare abbastanza bene le due classi (il valore AUC è pari a 0.890).



Per quanto riguarda la cross validation sul classificatore i dati ottenuti sono:

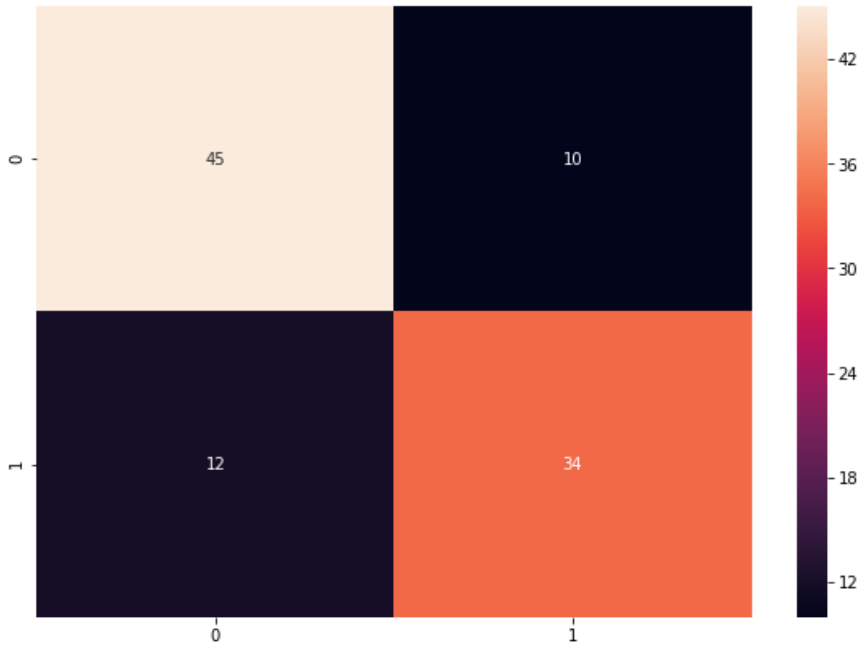
```
cv_scores mean ---> 0.7271341463414634
cv_score variance ---> 0.009199620761451524
cv_score dev standard ---> 0.09591465352828797
```

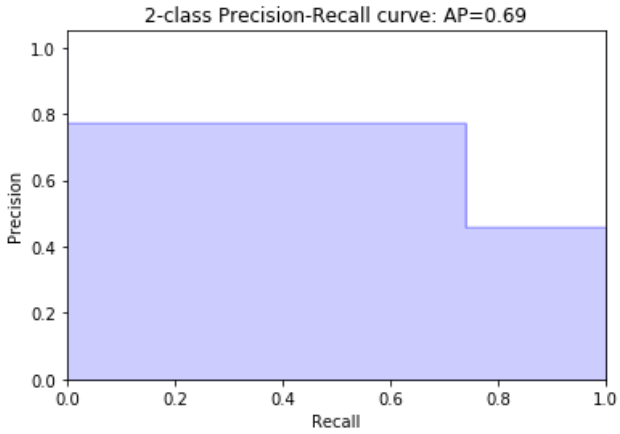
2.3 Support-Vector Machines

Un modello SVM è una rappresentazione degli esempi come punti nello spazio, mappati in modo tale che gli esempi appartenenti alle due diverse categorie siano chiaramente separati da uno spazio il più possibile ampio. I nuovi esempi sono quindi mappati nello stesso spazio e la predizione della categoria alla quale appartengono viene fatta sulla base del lato nel quale ricade.

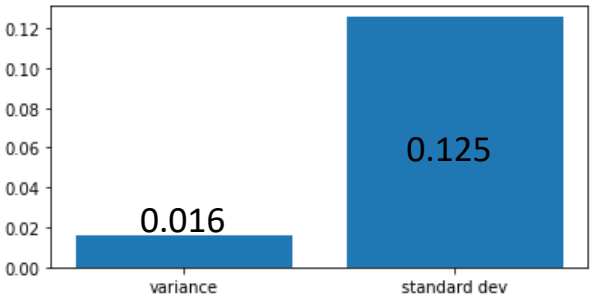
Clasification report:

	precision	recall	f1-score	support
0	0.79	0.82	0.80	55
1	0.77	0.74	0.76	46
micro avg	0.78	0.78	0.78	101
macro avg	0.78	0.78	0.78	101
weighted avg	0.78	0.78	0.78	101





Il grafico di precision-recall mostra un buon andamento della precisione al variare della recall con un grado di average precision pari a 0.69 e una accuracy pari a 0.78



Per quanto riguarda la cross validation sul classificatore i dati ottenuti sono:

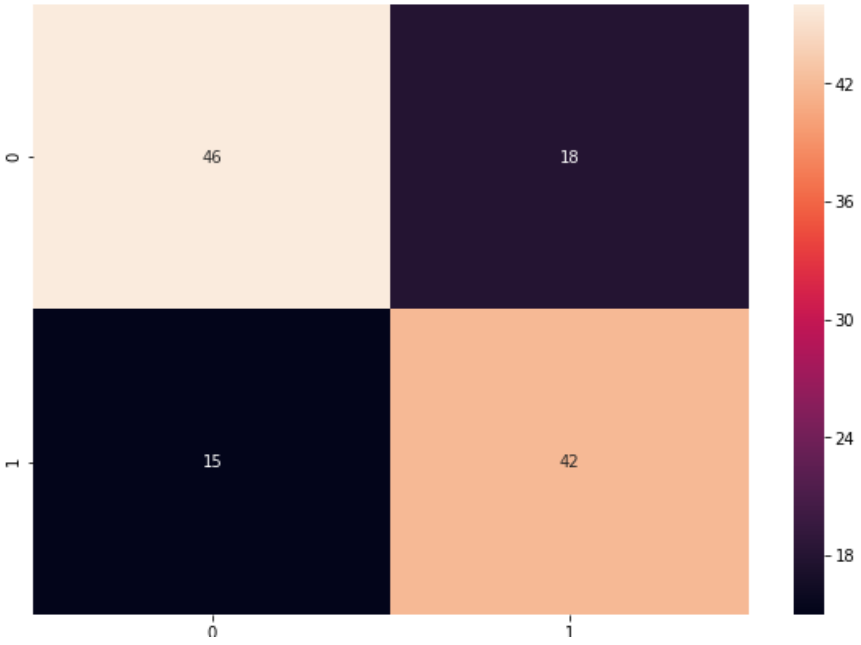
```
cv_scores mean ---> 0.6647560975609756
cv_score variance ---> 0.01570633551457466
cv_score dev standard ---> 0.125324919766879
```

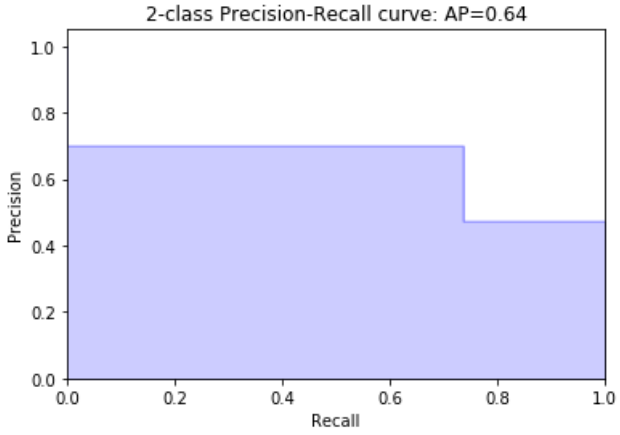
2.4 Multinomial Naive Bayes

I classificatori “naive Bayes” sono una famiglia di semplici classificatori probabilistici basati sull’applicazione del teorema di Bayes con una forte assunzione (ingenua) di indipedenza tra le feature. Con un modello di eventi multinomiali, gli esempi (vettori di feature) rappresentano le frequenze con cui certi eventi sono stati generati da una distribuzione polinomiale (p1, …, pn) dove pi è la probabilità che l’evento i si verifichi.

Clasification report:

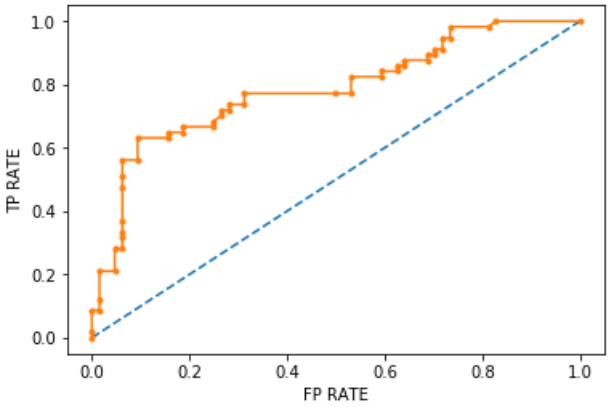
	precision	recall	f1-score	support
0	0.75	0.72	0.74	64
1	0.70	0.74	0.72	57
micro avg	0.73	0.73	0.73	121
macro avg	0.73	0.73	0.73	121
weighted avg	0.73	0.73	0.73	121



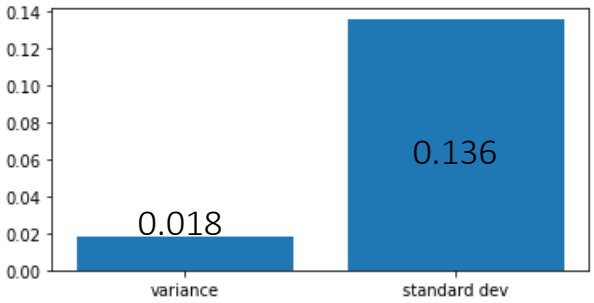


Il Multinomial Naive Bayes propone un'accuratezza del 73% (0.727) e una average precision di 0.639.

Tra gli algoritmi trattati fin ora, questo è quello che propone una precision, al variare della recall, leggermente più bassa. Nonostante l'uso dell'algoritmo SMOTE, l'average-precision è pari a 0.640.



Per quanto riguarda l'AUC, abbiamo un valore pari a 0.785.



Per quanto riguarda la cross validation (con cv=5) sul classificatore i dati ottenuti sono:

```
cv_scores mean ---> 0.6444512195121951
cv_score variance ---> 0.018386875371802495
cv_score dev standard ---> 0.13559821301109576
```

2.5 Neural network

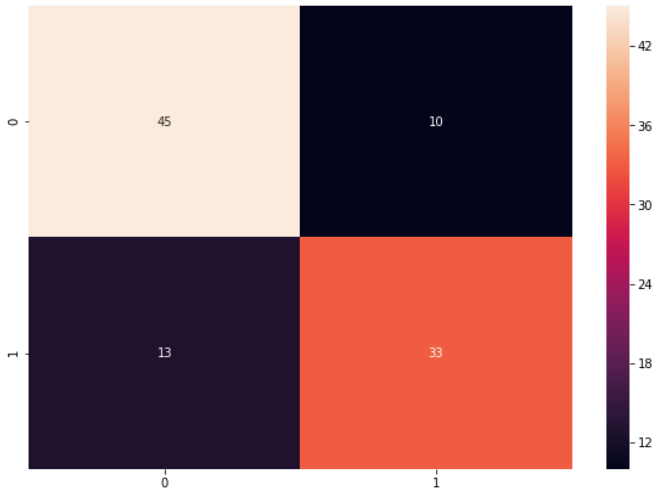
Una rete neurale" è un modello matematico/informatico di calcolo basato sulle reti neurali biologiche. Tale modello è costituito da un gruppo di interconnessioni di informazioni costituite da neuroni artificiali e processi che utilizzano un approccio di connessionismo di calcolo. Nella maggior parte dei casi una rete neurale artificiale è un sistema adattivo che cambia la sua struttura basata su informazioni esterne o interne che scorrono attraverso la rete durante la fase di apprendimento.

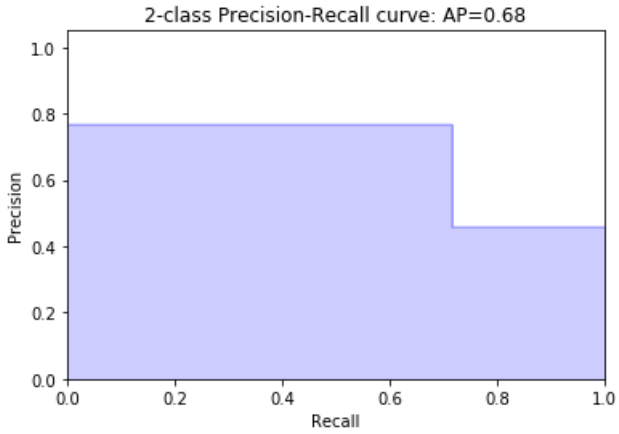
Nel nostro caso la nostra rete prevede una struttura sequenziale a tre livelli:

uno di input(costituito da 41 ingressi), uno nascosto(costituito da 17 neuroni artificiali) e uno di output(costituito da un solo neurone, abbiamo quindi una classificazione di tipo “single-class”) che restituisce un valore tra 0(esempio non appartenente alla categoria “recurrence-events” e 1 (esempio appartenente alla categoria “recurrence-events”)

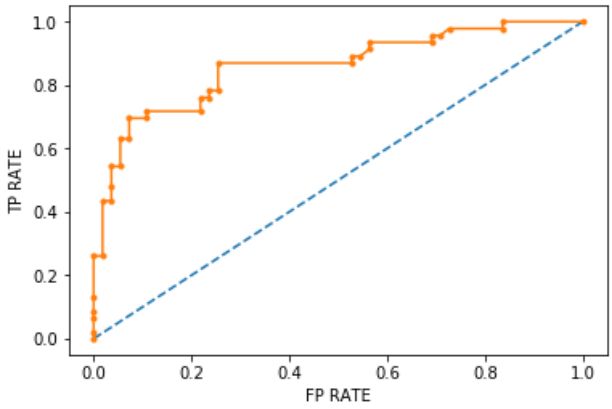
Clasification report:

	precision	recall	f1-score	support
0	0.78	0.82	0.80	55
1	0.77	0.72	0.74	46
micro avg	0.77	0.77	0.77	101
macro avg	0.77	0.77	0.77	101
weighted avg	0.77	0.77	0.77	101

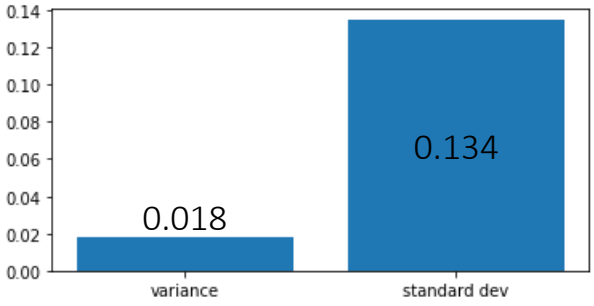




Il meccanismo di apprendimento supervisionato basato su rete neurale, prevede un'average precision pari a 0.679, un'accuratezza (0.77) in linea con le altre tecniche trattate e una f1 di 0.742.



Nella rete neurale, l'AUC è pari a 0.860.



Per quanto riguarda la cross validation sul classificatore i dati ottenuti sono:

```
cv_scores mean ---> 0.639567903086635
cv_score variance ---> 0.018067486318166993
cv_score dev standard ---> 0.13441535000946503
```

2.6 Tabella riassuntiva

ALGORITMO	*ACCURATEZZA*	*VARIANZA*	*DEV.STANDARD*	*F1*	*AVERAGE-PRECISION*	*AUC*
K-Nearest-neighbour	0.75 0.772(con SMOTE)	0.001 0.005(con SMOTE)	0.034 0.067(con SMOTE)	0.438 0.716 (con SMOTE)	0.491 0.691(con SMOTE)	0.688 0.875 (con SMOTE)
Random forest	0.777	0.009	0.096	0.757	0.697	0.890
Support-vector machines	0.782	0.016	0.125	0.756	0.690	/
Multinomial naive Bayes	0.727	0.018	0.136	0.718	0.640	0.785
Neural network	0.772	0.018	0.134	0.742	0.679	0.860

3. APPRENDIMENTO NON SUPERVISIONATO

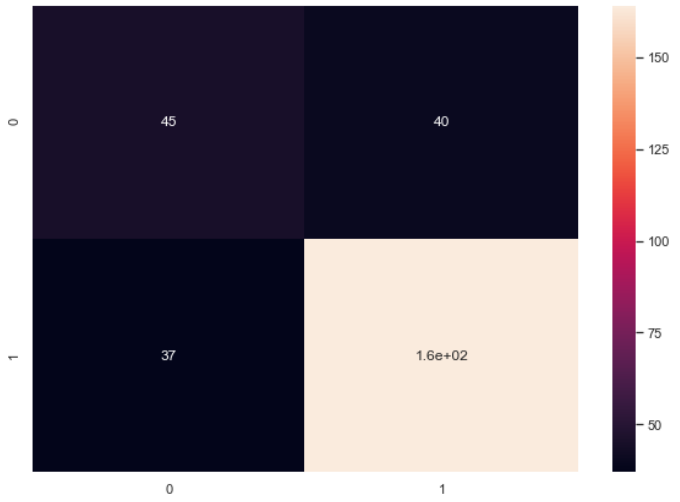
L'apprendimento non supervisionato è una tecnica di apprendimento automatico che consiste nel fornire al sistema informatico una serie di input (esperienza del sistema) che egli riclassificherà ed organizzerà sulla base di caratteristiche comuni per cercare di effettuare ragionamenti e previsioni sugli input successivi. Al contrario dell'apprendimento supervisionato, durante l'apprendimento vengono forniti all'apprendista solo esempi non annotati, in quanto le classi non sono note a priori ma devono essere apprese automaticamente.

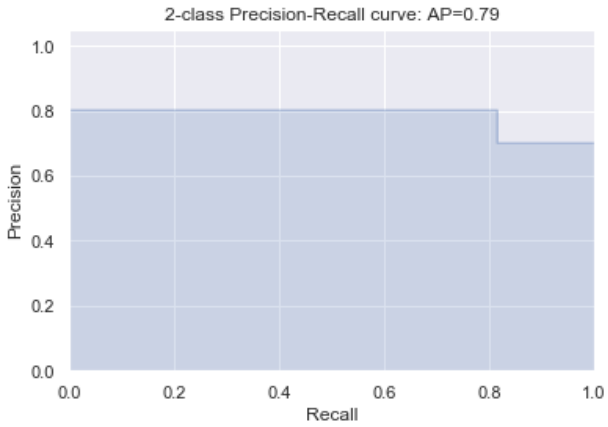
3.1 K-Means

L'algoritmo K-means è un algoritmo di hard-clustering partizionale che permette di suddividere un insieme di oggetti in K(nel nostro caso K=2) gruppi sulla base dei loro attributi. Si assume che gli attributi degli oggetti possano essere rappresentati come vettori, e che quindi formino uno spazio vettoriale. Ogni cluster viene identificato mediante un centroide. L'algoritmo segue una procedura iterativa. Inizialmente crea K partizioni e assegna ad ogni partizione i punti d'ingresso o casualmente o usando alcune informazioni euristiche. Quindi calcola il centroide di ogni gruppo. Costruisce quindi una nuova partizione associando ogni punto d'ingresso al cluster il cui centroide è più vicino ad esso. Quindi vengono ricalcolati i centroidi per i nuovi cluster e così via, finché l'algoritmo non converge.

Clasification report:

	precision	recall	f1-score	support
0	0.55	0.53	0.54	85
1	0.80	0.82	0.81	201
micro avg	0.73	0.73	0.73	286
macro avg	0.68	0.67	0.67	286
weighted avg	0.73	0.73	0.73	286





Il K-Means offre un buon grado di precisione nel predire quali sono gli esempi corrispondenti alla categoria “no-recurrence-events”, non è ottimale invece nel predire quali sono gli esempi della categoria “recurrence events”. Ciò nonostante, il K-Means garantisce un’accuratezza del 73% (0.731) e un’average-precision pari a 0.785.

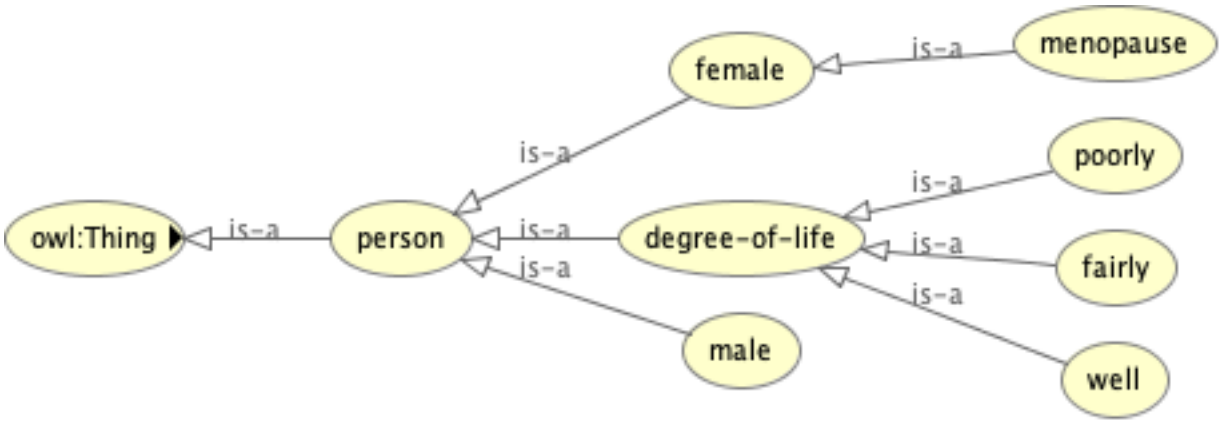
4. ONTOLOGIE

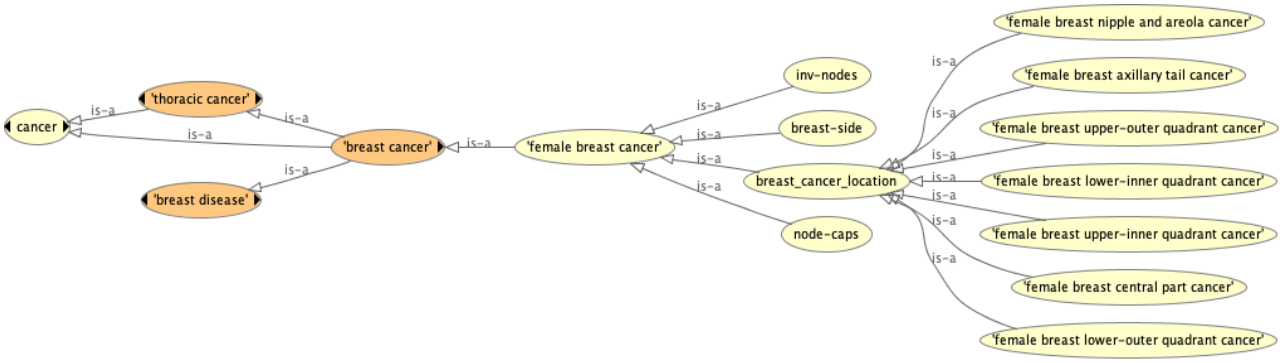
Un' ontologia è la specificazione dei significati dei simboli in un sistema informatico. La specifica formale è importante per l'interoperabilità semantica, ovvero l'abilità di basi di conoscenza differenti di operare insieme ad un livello semantico tale che i significati dei simboli sono rispettati. L'ontologia viene descritta come "una specificazione di una concettualizzazione". Una rappresentazione formale di un insieme di conoscenze è una concettualizzazione, ossia un insieme di oggetti, concetti e relazioni fra di essi che esistono in una particolare area d'interesse.

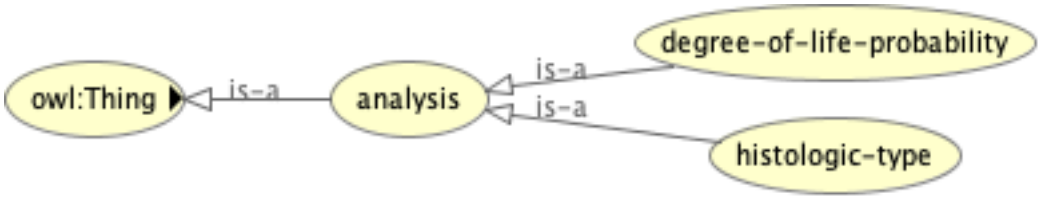
Nella modellizzazione della nostra ontologia ci siamo avvalsi del programma Protégé. Seguendo le linee guida, abbiamo usato una ontologia già esistente di malattie (http://www.obofoundry.org/ontology/doid.html) , a cui abbiamo integrato il nostro modello allo scopo di relazionare le stesse. Poiché ritenevamo il nostro dataset insufficiente nel dominio, abbiamo integrato un altro dataset esterno di tumori primari (https://archive.ics.uci.edu/ml/datasets/primary+tumor - correlato al nostro) da cui abbiamo estratto le feature più adatte alle nostre esigenze. Abbiamo così ottenuto un nuovo mondo in cui sono presenti, non solo le malattie già riportate, ma anche quelle dei due dataset congiunti, e le loro caratteristiche. Sono state aggiunte delle proprietà di oggetto e di dati con un dominio di appartenenza e un loro range. Il tutto cercando di mantenere una certa coerenza.

Da questo è possibile interrogare l'ontologia e ottenere nuove informazioni prima non reperibili. Usando il tab di Protègé "DL query" abbiamo ottenuto la probabilità di benessere di una donna affetta da cancro di una certa età. Un uso di questo tipo può trovare valenza aggiungendo un'ulteriore colonna al dataset originale contenente la probabilità di una persona di essere in uno stato di salute buono in relazione all'età della stessa. (Si è usato come ragionatore – reasoner – HermiT)

Di seguito viene riportata come è stata modellata l'ontologia:



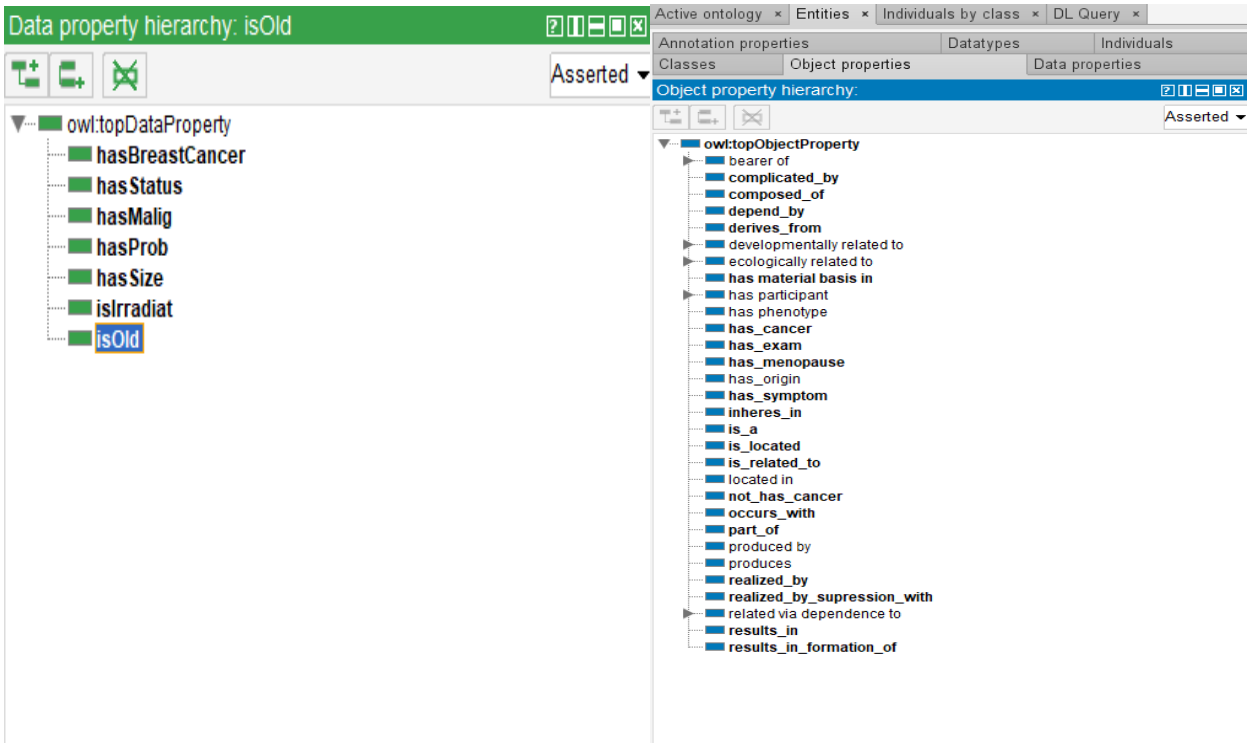




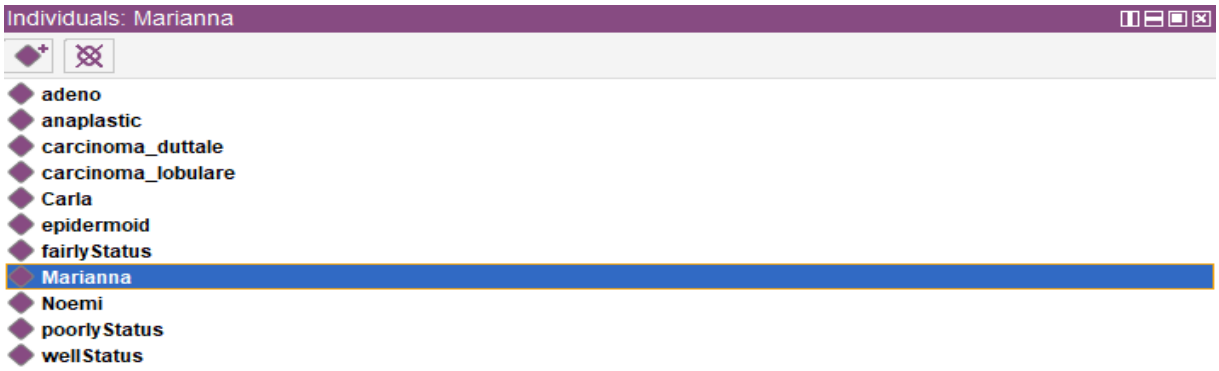
Queste entità sono in relazione tra di loro grazie a delle proprietà sia di oggetto che di data.

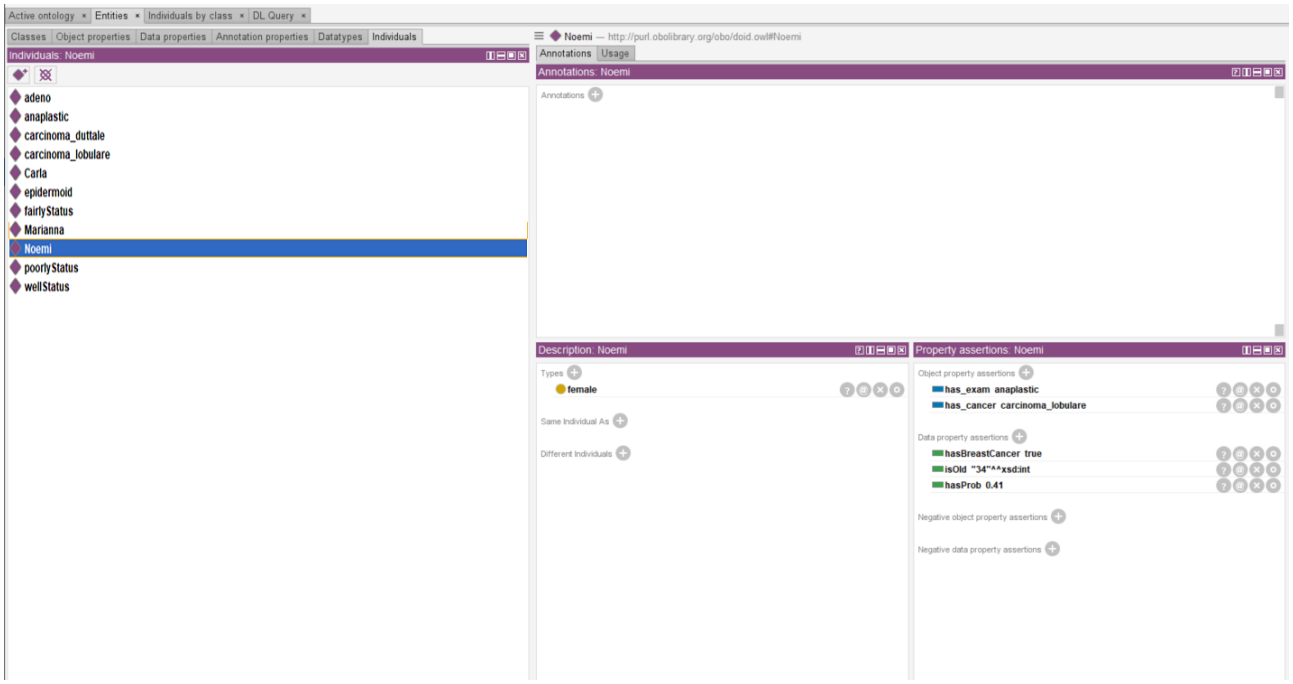
Una object property permette di mettere in relazione due individui, siano essi di classi distinte o della stessa classe.

Una data property permette di mettere in relazione un individuo con un valore di tipo primitivo



Inoltre, per alcune entità si sono create delle istanze. Alcune ad esempio per individuare una tipologia di esame come quella istologica, altre per individuare istanze di persone a cui attribuire ad esempio una malattia e le relazioni con essa.

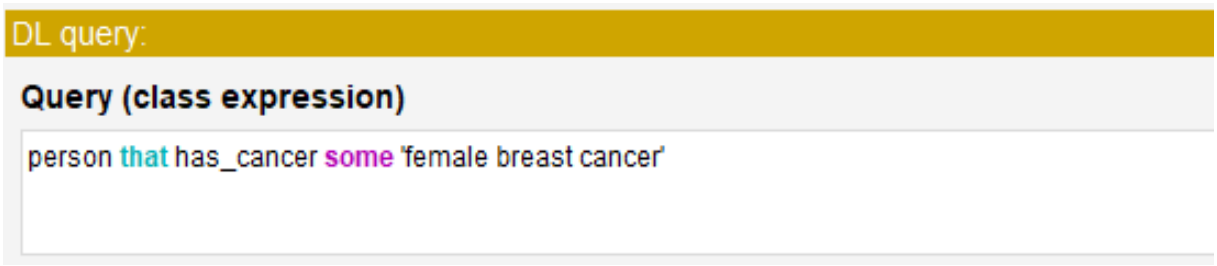




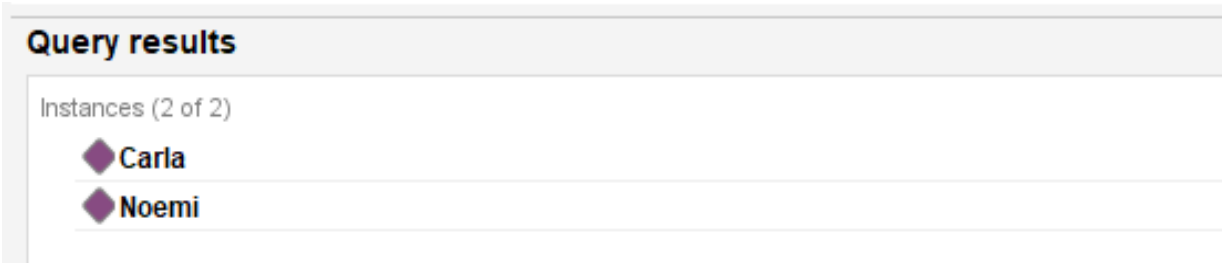
Successivamente sono state formulate delle query per interrogare l'ontologia. Si è partiti con query semplici come per esempio cercare persone affette da un tumore al seno e a quali esami istologici si sono sottoposte fino a query più complesse che permettono di ottenere la lista di persone con un dato stato di salute, ottenuto dalla combinazione di più attributi inerenti a ciascuna persona.

Le query formulate sono le seguenti:

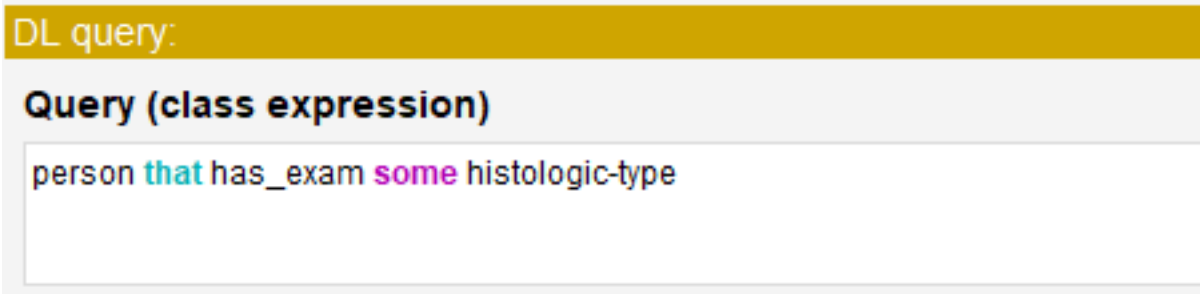
1. Query: Ottenere tutte le persone che sono affette da un qualsiasi tumore al seno



Result Set

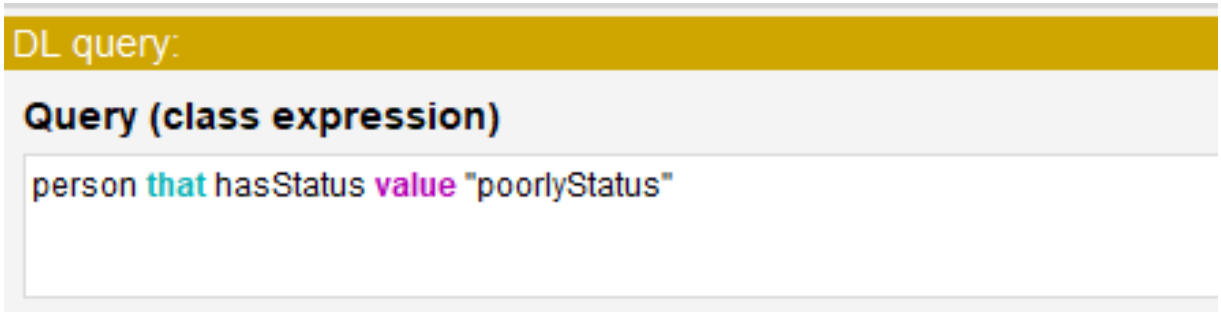


2. Query: Ottenere tutte le persone che si sono sottoposte ad un esame istologico

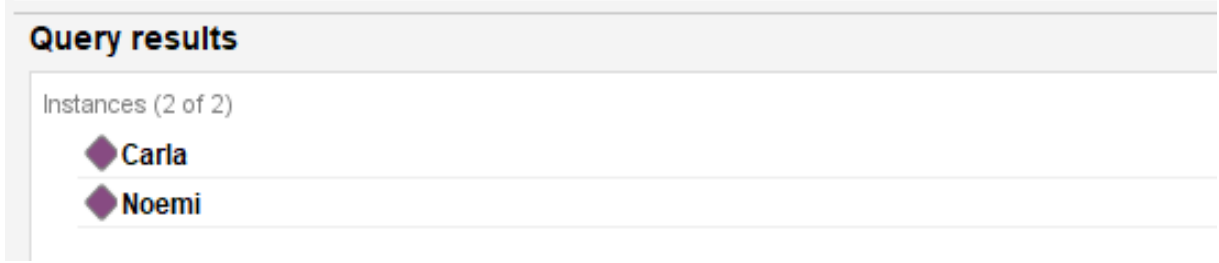


Result Set

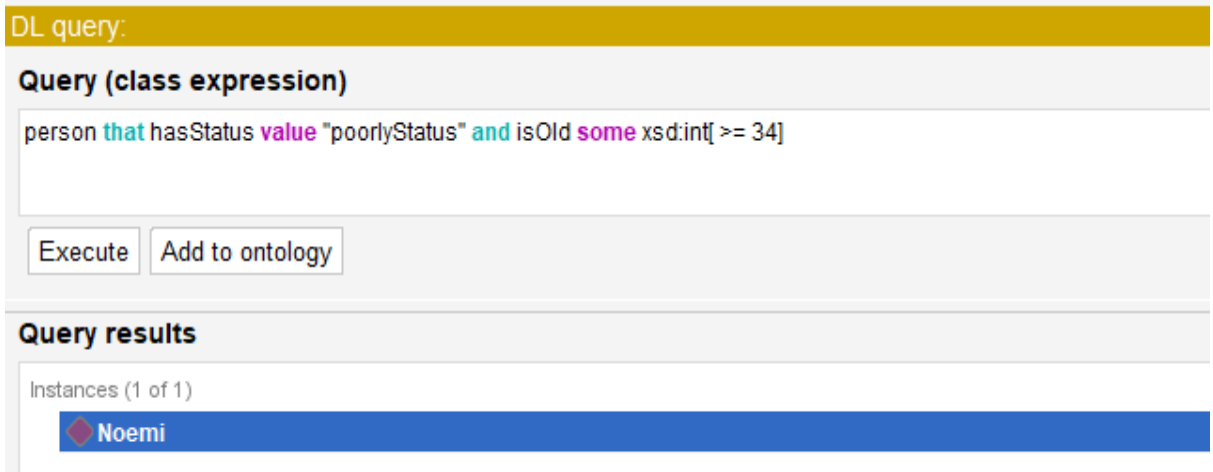
3. Query: Ottenere tutte le persone con uno stato di salute basso



Result Set



4. Query: Ottenere tutte le persone con età maggiore di 34 con uno stato di salute basso



In particolare le ultime due query sono state eseguite dal reasoner conoscendo solamente l'età della persona e la probabilità ad esso associata per lo stato di salute. Dalla combinazione di questi due dati il reasoner attribuisce automaticamente lo stato di salute ad una persona.