

# Anime Base

## Documentazione Progetto

Thomas Carella, MAT. 724810, [t.carella5@studenti.uniba.it](mailto:t.carella5@studenti.uniba.it)

<https://github.com/ThomasCarella/Anime-Base>

AA 2021-2022

## Introduzione

Il progetto consiste in una **base di conoscenza** che permette di avere conoscenza di un dominio di interesse degli anime. Questo obiettivo viene svolto attraverso un modello di **classificazione** con **predizioni** sulla categoria dei ratings degli anime da parte degli utenti online, ed un'**ontologia** che ne descrive il dominio specificandone alcuni dettagli che l'utente vuole sapere. È inoltre presente un sistema di **recommender system** basato su *content based filtering* e *collaborative filtering* che permette di avere raccomandazione in base al nome di un anime o ai generi preferiti. Per ultimo, è anche presente un sistema che determina se l'anime è stato un successo o no usando diversi **modelli di apprendimento** (supervisionati e non supervisionati).

## Sommario

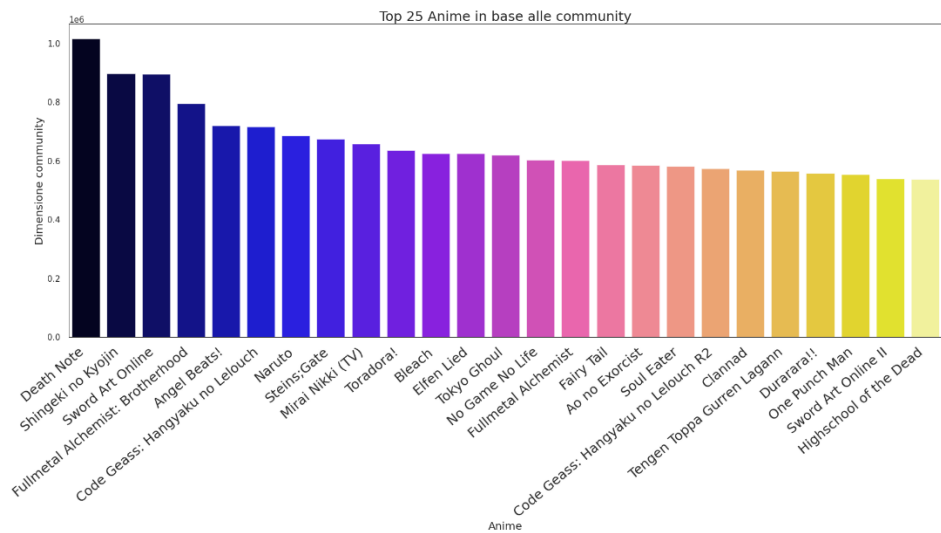
Il dataset usato (<https://www.kaggle.com/datasets/CooperUnion/anime-recommendations-database>) è stato pulito di alcuni degli elementi non utilizzabili e/o vuoti. Contiene al suo interno due file, denominati rispettivamente anime e rating.

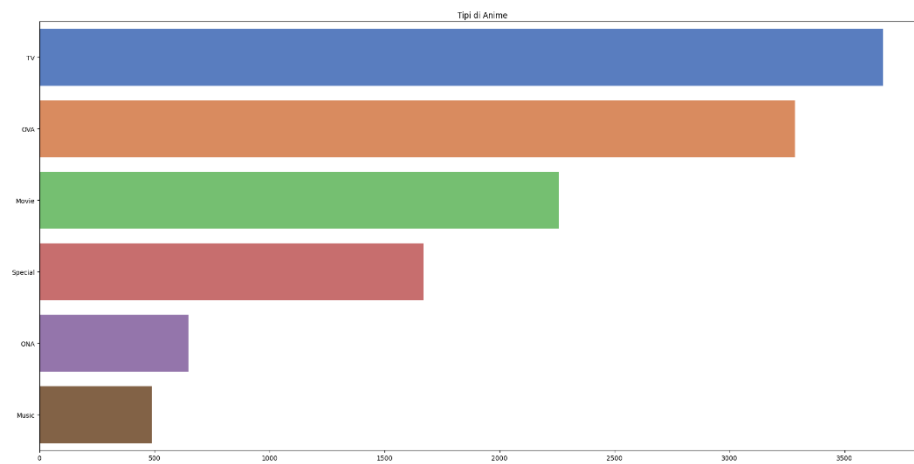
Il file Anime.csv contiene i campi:

- **anime\_id**: id univoco dell'anime (gli id sono estratti dal sito myanimelist.net);
- **name**: il nome completo dell'anime;
- **genre**: lista di generi separati da una virgola;
- **type**: film, serie TV, puntate speciali (OVA), etc...;
- **episodes**: quanti episodi contiene (1 se è un film);
- **rating**: punteggio medio da 1 a 10;
- **members**: numero dei membri della community che si trovano in un "gruppo" in cui si parla di quell'anime;

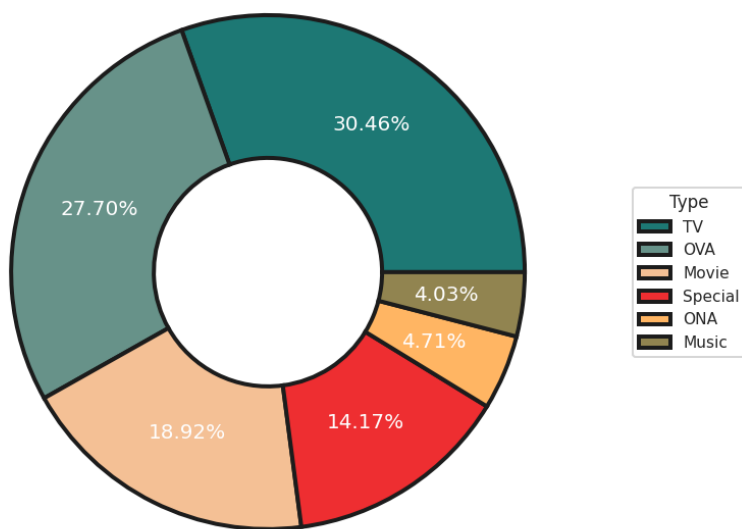
Il dataset Rating.csv contiene i campi

- **user\_id**: identificativo casuale per un utente;
- **anime\_id**: l'anime che l'utente ha votato;
- **rating**: punteggio da 1 a 10 che l'utente ha assegnato all'anime (-1 se l'utente lo ha visto ma non ha assegnato un punteggio);





Distribuzione tipi di anime



## Elenco argomenti di interesse

- Sistemi basati su conoscenza: Ontologia
- Ragionamento su individui e relazioni (knowledge base)
- Apprendimento supervisionato
- Apprendimento non Supervisionato: hard clustering (k-Means)

# Ontologia

## Sommario

L'ontologia, in informatica, è un modello di rappresentazione formale della realtà e della conoscenza. È una struttura di dati che consente di descrivere le entità (oggetti, concetti, ecc.) e le loro relazioni in un determinato dominio di conoscenza. [15]

## Strumenti utilizzati

L'ontologia usata per manipolare la KBS è stata interamente costruita con python, per migliorare la coerenza con il codice già presente e per facilità d'uso. Il dominio illustrato all'interno dell'ontologia stessa e del grafico rappresenta un dominio degli anime e degli utenti che li votano. In questo contesto l'utente, incuriosito dal sapere di che cosa tratta, visualizza i generi, il tipo, il ranking ed i voti degli utenti ad esso collegati.

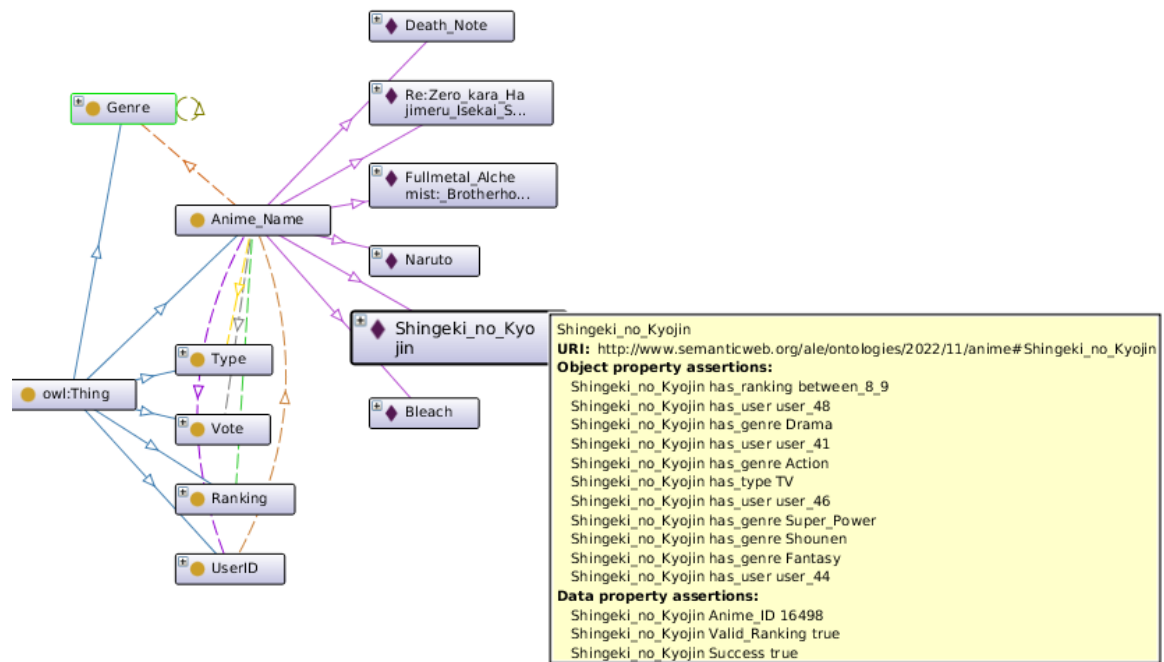
La realizzazione dell'ontologia (creata con **Protégé**) è stata divisa in classi la cui la maggior parte rappresenta una categoria legata ad un genere, il tipo, assieme al ranking. Vi è anche quello dell'utente assieme ai voti inseriti.

Oltre alla creazione delle classi, sono state create anche una serie di proprietà: object-properties e data-properties. Queste proprietà aiutano a relazionare due individui di classi uguali o diverse (object-properties) oppure permettono di relazionare un individuo al loro tipo di dato primitivo (data-properties).

## Decisioni di progetto

Per la consultazione in Python dell'ontologia, è stata utilizzata la libreria **Owlready2** [16], attraverso la quale è stato possibile, in maniera semplice e comprensibile, interrogare l'ontologia precedentemente creata con il tool Protégé. Nell'esecuzione del programma si potrà scegliere tra la visualizzazione delle classi, delle proprietà di oggetto, delle proprietà dei dati e della visualizzazione di alcune query d'esempio.

## Valutazione



```

Enter your choice : 10

WELCOME TO THE ONTOLOGY

Select what you would like to explore:

1) Classes view
2) Object properties view
3) Data properties view
4) Example query view
5) Exit Ontology

Enter your choice here:
1

classes present in the ontology:

[anime.UserID, anime.Anime_Name, anime.Genre, anime.Ranking, anime.Type, anime.Vote]

Would you like to explore any particular class better?

1) UserID
2) Anime_Name
3) Ranking
4) Genre
5) Type
6) Vote

Enter your choice here:
1

List of UserIDs present:

[anime.UserID, anime.user_33, anime.user_34, anime.user_38, anime.user_8, anime.user_3, anime.user_39, anime.user_40, anime.user_1, anime.user_13, anime.user_41, anime.user_44, anime.user_46, anime.user_48, anime.user_10, anime.user_11, anime.user_12, anime.user_17]

Would you like to go back or continue?
Back (yes) Continue (no)

yes
Select what you would like to explore:

1) Classes view
2) Object properties view
3) Data properties view
4) Example query view
5) Exit Ontology

Enter your choice here:
2

Object properties present in the ontology:

[anime.has_anime_name, anime.has_user, anime.has_genre, anime.has_ranking, anime.has_sibling_genre, anime.has_type, anime.has_vote]

Select what you would like to explore:

1) Classes view
2) Object properties view
3) Data properties view
4) Example query view
5) Exit Ontology

Enter your choice here:
3

Properties of the data present in the ontology:

[anime.Anime_ID, anime.Success, anime.Valid_Ranking]

Select what you would like to explore:

```



Select what you would like to explore:

- 1) Classes view
- 2) Object properties view
- 3) Data properties view
- 4) Example query view
- 5) Exit Ontology

Enter your choice here:

3

Properties of the data present in the ontology:

[anime.Anime\_ID, anime.Success, anime.Valid\_Ranking]

Select what you would like to explore:

- 1) Classes view
- 2) Object properties view
- 3) Data properties view
- 4) Example query view
- 5) Exit Ontology

Enter your choice here:

4

Example queries:

List of anime featuring the 'Fantasy' category:

[anime.Shingeki\_no\_Kyojin, anime.Fullmetal\_Alchemist:\_Brotherhood, anime.Re:Zero\_kara\_Hajimeru\_Isekai\_Seikatsu]

List of anime featuring the Ranking 'between 8-9':

[anime.Death\_Note, anime.Shingeki\_no\_Kyojin, anime.Re:Zero\_kara\_Hajimeru\_Isekai\_Seikatsu]

List of animex featuring the user 'user\_38':

[anime.Bleach, anime.Death\_Note, anime.Re:Zero\_kara\_Hajimeru\_Isekai\_Seikatsu]

Select what you would like to explore:

- 1) Classes view
- 2) Object properties view
- 3) Data properties view
- 4) Example query view
- 5) Exit Ontology

Enter your choice here:

5

# Ragionamento su individui e relazioni (knowledge base)

## Sommario

La knowledge base (o KB) è definibile come un insieme di assiomi, cioè delle proposizioni che possono essere asserite essere vere che costituiscono dunque un ambiente volto a facilitare la raccolta, l'organizzazione e la distribuzione della conoscenza.

In merito al nostro caso di studio, la KB viene utilizzata per consentire all'utente di porre domande inerenti al dominio approfondito.

## Strumenti utilizzati

Per la gestione degli assiomi all'interno della KB è stato usato python, così come per il popolamento, che avviene direttamente dal dataset. Così facendo, nel caso in cui il dataset venisse aggiornato con nuovi dati, si aggiornerebbe direttamente anche la KB.

## Decisioni di progetto

Nella costruzione della KB, sono stati aggiunti le seguenti regole:

- Dato il nome di un anime, restituisce i dettagli come ranking e tipo (TV, serie, etc...)
- Dato il nome di un anime e un genere, restituisce una risposta positiva o negativa se il genere appartiene all'anime o meno
- Mostra i generi che appartengono all'anime
- Dato i nomi di due anime, restituisce vero o falso se questi hanno almeno un genere in comune tra loro
- Dato il nome di un anime che ci piace e uno che vorremmo sapere se ci può interessare, restituisce la probabilità che l'anime inserito possa interessarci
- Dato un certo numero di generi, restituisce il numero di anime collegati a quei generi

L'ultima regola permette la completa navigabilità degli anime in base a una combinazione di generi. Questa regola è stata ritenuta specialmente utile all'interno del progetto, in quanto nuova nell'uso e utile per navigare i dati all'interno di una KB decentemente grande.

All'interno del progetto è stata usata **l'assunzione del nome unico** (unique name assumption). È un'assunzione diversa sui nomi in una base di conoscenza, secondo cui due nomi diversi denotano sempre due individui diversi.

## Valutazione

```
Enter your choice : 4
Enter anime name as example Naruto : Naruto
  Anime name  Rating Type
0      Naruto    7.81   TV

Enter your choice : 5
Enter anime name : Naruto
Select one genres
1 Action
2 Adventure
3 Cars
4 Comedy
5 Dementia
6 Demons
7 Drama
8 Ecchi
9 Fantasy
10 Game
11 Harem
12 Hentai
13 Historical
14 Horror
15 Josei
16 Kids
17 Magic
18 Martial Arts
19 Mecha
20 Military
21 Music
22 Mystery
23 Parody
24 Police
25 Psychological
26 Romance
27 Samurai
28 School
29 Sci-Fi
30 Seinen
31 Shoujo
32 Shoujo Ai
33 Shounen
34 Shounen Ai
35 Slice of Life
36 Space
37 Sports
38 Super Power
39 Supernatural
40 Thriller
41 Vampire
42 Yaoi
43 Yuri
If you enter a number that does not match a listed gender, the question will be repeated
Enter your choice : 1
No

Enter your choice : 6
Enter anime name : Naruto
Genre :
Action
Comedy
Martial Arts
Shounen
Super Power

Enter your choice : 7
Enter first anime name : Naruto
Enter second anime name : Gintama
Yes
```

## Funzione 8

```

39      Thriller          28
40      Vampire          47
41      Yaoi             4
42      Yuri             3
Insert anime genre/type : Adventure
      Anime Genres      Number of anime
0      Action           871
1      Adventure         871
2      Cars              4
3      Comedy            321
4      Dementia          1
5      Demons            33
6      Drama             146
7      Ecchi             44
8      Fantasy           319
9      Game              33
10     Harem             8
11     Hentai            4
12     Historical         45
13     Horror            18
14     Kids              79
15     Magic              91
16     Martial Arts       55
17     Mecha             154
18     Military           57
19     Music              6
20     Mystery           31
21     Parody            10
22     Police            23
23     Psychological      7
24     Romance           76
25     Samurai           23
26     School            15
27     Sci-Fi            332
28     Seinen            30
29     Shoujo            10
30     Shounen           337
31     Slice of Life      5
32     Space              72
33     Sports            10
34     Super Power        98
35     Supernatural       68
36     Thriller           2
37     Vampire            5
Insert anime genre/type : Vampire
      Anime Genres      Number of anime
0      Action            5
1      Adventure          5
2      Comedy            2
3      Demons            1
4      Fantasy           1
5      Horror            1
6      Magic             1
7      Sci-Fi            1
8      Shounen           2
9      Supernatural       3
10     Vampire           5
Insert anime genre/type : Details
There is not a genre/type with this name
Warning : There is only 1 type for anime
Insert anime genre/type : DETAILS
      Name                                     Genres      Type      rating
93      JoJo no Kimyou na Bouken (TV)         Action, Adventure, Shounen, Supernatural, Vampire      TV      8.51
914     JoJo no Kimyou na Bouken: Phantom Blood      Action, Adventure, Horror, Shounen, Vampire      Movie      7.77
3768     Master Mosquiton                       Action, Adventure, Comedy, Demons, Supernatura...      OVA      6.89
4349     Wild Arms: Twilight Venom               Action, Adventure, Fantasy, Magic, Sci-Fi, Vam...      TV      6.75
4501     Master Mosquiton 99                     Action, Adventure, Comedy, Supernatural, Vampire      TV      6.71
Insert anime genre/type : STOP

Enter your choice : 9
Enter first anime name(that you like it) : Naruto
Enter second anime name(that you want know if you can like it) : Gintama
68%

```

# Apprendimento supervisionato

## Sommario

L'apprendimento supervisionato viene utilizzato nel progetto per fornire raccomandazioni all'utente di anime che potrebbero essere d'interesse. Il grado di interesse viene determinato in base agli anime guardati e il tasso di successo di quelli consigliati e in base al ranking.

I metodi utilizzati a tale scopo sono:

- Algoritmi di classificazione e regressione (lineari e non)
- Algoritmi basati sugli alberi di decisione
- Algoritmi basati sulle reti neurali
- Algoritmi di ragionamento basato su casi (CBR): kNN, distanze
- Algoritmi statistici: tf-idf

## Strumenti utilizzati

### Gaussian naive classifier

Rientra nei classificatori probabilistici, sono una famiglia di semplici classificatori probabilistici basati sull'applicazione del teorema di Bayes con una forte assunzione (ingenua) di indipendenza tra le feature. Questo avviene con tutti i classificatori Naive Bayes, i quali presuppongono che il valore di una particolare caratteristica sia indipendente dal valore di qualsiasi altra feature, data la variabile di classe. [1]

### Logistic regression

Modello di regressione statico, che converte i valori tra 0 e 1. In questo modello, le probabilità che descrivono il possibile risultato di un singolo tentativo sono modellati usando una funzione logistica. [2]

### K-nn classifier

In questo classificatore un oggetto è ordinato da un voto di pluralità con l'oggetto assegnato alla classe più comune tra i suoi  $k$  vicini più vicini [3]

### Decision tree classifier

Il metodo più utilizzato. Si basa su un albero di classificatori dove ogni nodo è associato ad una domanda su una feature [4]

### Random forest classifier

Modello composto da molti alberi di decisione, in cui le feature su cui vengono fatte le predizioni sono permutate a random. Da questo nodo dipartono tanti archi quanti sono i possibili valori che la caratteristica può assumere, fino a raggiungere le foglie che indicano la categoria associata alla decisione. [5]

### MLP classifier

È un classificatore che appartiene alle reti neurali, che allena il modello tramite back-propagation ed agisce a più strati. Con questo modello viene ottimizzata la funzione di perdita logaritmica utilizzando l'algoritmo L-BFGS oppure la discesa del gradiente stocastico. Quest'ultimo è stato scelto dal gruppo in quanto è relativamente più veloce durante la fase di allenamento. [6]

### Gradient Boosting Classifier

Algoritmo che rientra nelle tecniche di apprendimento automatico, ma implementato per classificazione supervisionata. [7]

### TF-IDF

È una funzione utilizzata per misurare l'importanza di un termine rispetto ad un documento o ad una collezione di documenti. Tale funzione aumenta proporzionalmente al numero di volte che il termine è contenuto nel documento, ma cresce in maniera inversamente proporzionale con la frequenza del termine nella collezione [8]

### Distanza euclidea

La distanza euclidea è una distanza tra due punti, in particolare è una misura della lunghezza del segmento avente per estremi i due punti. Usando questa distanza, lo spazio euclideo diventa uno spazio metrico. I metodi basati sulla distanza danno priorità agli elementi con i valori più bassi per poter riconoscere una possibile similarità tra di essi. [9]

### Correlazione di pearson

L'indice di correlazione di Pearson è una tecnica per investigare un'eventuale relazione di linearità tra due quantitativi, che siano continui. È una misura legata alla forza e alla direzione di una relazione lineare. [10] È tra gli algoritmi più veloci per il sistema di raccomandazioni.

### Random forest

Ogni albero in una *random forest* impara da un campione casuale di dati. I campioni vengono disegnati con la sostituzione, nota come *bootstrap*, il che significa che alcuni campioni verranno utilizzati più volte in un singolo albero [?]. È stato utile al progetto in quanto facile da implementare ed è possibile avere gli iperparametri predefiniti ottimali. Questo perché producono un buon risultato di previsione, consentendo anche un notevole miglioramento di essi, e di conseguenza della previsione. [11]

### Support vector

L'algoritmo SVM può essere impiegato per problemi di classificazione multiclasse, utilizzando la metodologia **one-vs-one**.

Oltre a eseguire la classificazione lineare, le SVM possono eseguire in modo efficiente una classificazione non lineare utilizzando quello che viene chiamato il **trucco del kernel**, mappando implicitamente i loro input in spazi di funzionalità ad alta dimensione. [12]

### Bagging classifier

Un classificatore Bagging è un **meta stimatore** d'insieme che adatta i classificatori di base ciascuno su sottoinsiemi casuali del set di dati originale e quindi aggrega le loro previsioni individuali (votando o calcolando la media) per formare una previsione finale. [13]

### Decisioni di progetto

Scegliendo la prima funzione si avvia una funzione che lavora con il collaborative filtering ed content based filtering. Prende in input il nome di un anime che l'utente conosce. Dopodiché avvia uno dopo l'altro i diversi algoritmi (similarità di coseno, distanza euclidea e correlazione di Pearson, tf-idf per il content based filtering, mentre per il collaborative filtering l'algoritmo knn) per trovare quattro liste di dieci anime simili che potrebbero interessare all'utente. Per il content based filtering viene visualizzata la scarsità del dataset.

Scegliendo la seconda funzione si avvia un recommended system basato su content based filtering in base ai generi (e alle combinazioni che ne scaturiscono). La scelta delle metriche usate è stata fatta tra la similarità di coseno, correlazione di Pearson e distanza euclidea. La più performante in tempo di esecuzione e ai dati consigliati è stata la Correlazione di Pearson.

Successivamente, dopo aver ottenuto i 10 anime più simili all'anime dato in input, viene eseguita una classificazione relativamente al ranking degli anime. Per fare ciò, è stato usato il modello knn su un modello addestrato con divisione del dataset del training e test dell'80% e 20% rispettivamente. Al modello ottenuto viene dato un **classification report** per poter includere più metriche di valutazione. Infine, viene calcolato il **ROC-AUC** score.

Al fine di aumentare il ROC-AUC, è stato usato il RandomizedSearchCv per fare Hyperparameters tuning. La ricerca è stata fatta 15 volte al fine di trovare i parametri giusti per migliorare il ROC-AUC score.

Le ultime 3 funzioni, al di fuori dello scopo del programma ma utili all'interno del progetto sono:

- Una funzione che, dato in input il nome di un anime, usa diversi algoritmi (gaussian naive classifier, knn, decision tree, mlp, gradient boosting) e restituisce se l'anime ha avuto successo o meno.
- Una funzione che, dato in input il nome di un anime, usa dei modelli di test per restituire in output le possibilità di successo di quel determinato anime
- Una funzione che, restituisce un report basato sull'intero modello di test con le percentuali di successo (precision; recall; f1-score; support)

## Valutazioni

```
Enter your choice : 1
Do you want proceed with a name test : Shingeki no Kyojin a.k.a. Attack of titan ?
Enter y for a yes, something else for no : y
Using Collaborative Filtering, the Top 10 most similar anime are :
Warning : to reduce the use of ram,
the size of the database will be reduced, anime with at least 500 votes will be considered
0: Shingeki no Kyojin, with distance of 0.0:
1: Sword Art Online, with distance of 0.1335946872659557:
2: Angel Beats!, with distance of 0.14903078429088036:
3: No Game No Life, with distance of 0.16254528367480707:
4: Hataraku Maou-sama!, with distance of 0.1638429001300072:
5: Steins;Gate, with distance of 0.16728424722076785:
6: Highschool of the Dead, with distance of 0.16957425002499732:
7: Toradora!, with distance of 0.1752054636541328:
8: Psycho-Pass, with distance of 0.17726390458904436:
9: Death Note, with distance of 0.18132084001891173:
10: Code Geass: Hangyaku no Lelouch, with distance of 0.18198153072037426:
Time to calculate it : 0.51 seconds

Using Cosine Similarty
Using Content based filtering, the Top 10 most similar anime are :

      Anime name      Rating
0      Shingeki no Kyojin OVA      7.88
1      Shingeki no Kyojin Movie 2: Jiyuu no Tsubasa      7.75
2      Shingeki no Kyojin Movie 1: Guren no Yumiya      7.70
3      Shingeki no Kyojin: Ano Hi Kara      6.96
4      One Piece      8.58
5      One Piece: Episode of Merry - Mou Hitori no Na...      8.29
6      One Piece: Episode of Nami - Koukaishi no Nami...      8.27
7      One Piece: Episode of Sabo - 3 Kyoudai no Kizu...      7.78
8      Dragon Ball Specials      6.96
9      Saint Seiya: Meiou Hades Elysion-hen      7.82
Time to calculate it : 10.43 seconds
Using Euclidean Distances
Using Content based filtering, the Top 10 most similar anime are :

      Anime name      Rating
0      Suzumiya Haruhi no Shoushitsu      8.81
1      Monogatari Series: Second Season      8.80
2      Death Note      8.71
3      Ping Pong The Animation      8.67
4      Boku dake ga Inai Machi      8.65
5      Steins;Gate Movie: Fuka Ryouiki no Déjà vu      8.61
6      Yuri!!! on Ice      8.61
7      Uchuu Kyoudai      8.59
8      Bakemono no Ko      8.58
9      Usagi Drop      8.56
Time to calculate it : 5.15 seconds
Using Pearson Similarty
Using Content based filtering, the Top 10 most similar anime are :

      Anime name      Rating
0      Shingeki no Kyojin      8.54
1      Shingeki no Kyojin OVA      7.88
2      Shingeki no Kyojin Movie 2: Jiyuu no Tsubasa      7.75
3      Shingeki no Kyojin Movie 1: Guren no Yumiya      7.70
4      Shingeki no Kyojin: Ano Hi Kara      6.96
...      ...      ...
12012      Sakura Trick      7.20
12013      Suzumiya Haruhi no Yuuutsu      8.06
12014      Boku wa Tomodachi ga Sukunai      7.57
12015      Kagerou Daze: In a Days      7.29
12016      Suzumiya Haruhi no Yuuutsu (2009)      7.28

[12017 rows x 2 columns]
Time to calculate it : 2.18 seconds
Sparsity of the dataset: 78.50594011424991 %
```



## Funzione 2

```
Using Content based filtering, the Top 10 most similar anime are :

      Anime name  Rating
0      Shingeki no Kyojin  8.54
1      Shingeki no Kyojin OVA  7.88
2      Shingeki no Kyojin Movie 2: Jiyuu no Tsubasa  7.75
3      Shingeki no Kyojin Movie 1: Guren no Yumiya  7.78
4      Shingeki no Kyojin: Ano Hi Kara  6.96
...
12012      Sakura Trick  7.28
12013      Suzumiya Haruhi no Yuuutsu  8.06
12014      Boku wa Tomodachi ga Sukunai  7.57
12015      Kagerou Daze: In a Days  7.29
12016      Suzumiya Haruhi no Yuuutsu (2009)  7.28

[12017 rows x 2 columns]
Time to calculate it : 2.12 seconds

Initial model composition with basic hyperparameters

Predictions of the first 10 elements: [0 0 0 0 7 0 0 7 0 0]
Actual values: [0 0 0 0 7 7 7 8 0 7]
Model evaluation...

Classification report:
      precision    recall  f1-score   support

0         0.76      0.87      0.81      1621
7         0.58      0.37      0.42       658
8         0.48      0.19      0.27       128
9         0.00      0.00      0.00         5

 accuracy      0.78      2484
 macro avg      0.43      0.36      0.38      2484
weighted avg      0.67      0.78      0.68      2484

ROC score: 0.6671887548624621
Accuracy is low, let's try to improve the quality of the predictions

WITH GRID SEARCH:

Best weights: distance
Best metric: manhattan
Best n_neighbors: 22
Let's recompose the model using the new hyperparameters...
Predictions of the top 10 items in the ranking category: [0 0 0 0 7 0 0 7 0 7] Actual values: [0 0 0 0 7 7 7 8 0 7]
Classification report:
      precision    recall  f1-score   support

0         0.76      0.98      0.83      1621
7         0.55      0.36      0.43       658
8         0.43      0.28      0.27       128
9         0.00      0.00      0.00         5

 accuracy      0.72      2484
 macro avg      0.44      0.37      0.38      2484
weighted avg      0.69      0.72      0.69      2484

ROC score: 0.6768277912862548
We have increased the accuracy of our model
Now we can proceed to the recommendation phase...
The 10 anime most similar to the one proposed with a prediction on the ranking:

Name : Kimi no Na wa.
Rating : 9.37
Prediction Rating 6

Name : Fullmetal Alchemist: Brotherhood
Rating : 9.26
Prediction Rating 9
```

funzioni extra:

funzione 11

```

10 Game
11 Harem
12 Hentai
13 Historical
14 Horror
15 Josei
16 Kids
17 Magic
18 Martial Arts
19 Mecha
20 Military
21 Music
22 Mystery
23 Parody
24 Police
25 Psychological
26 Romance
27 Samurai
28 School
29 Sci-Fi
30 Seinen
31 Shoujo
32 Shoujo Ai
33 Shounen
34 Shounen Ai
35 Slice of Life
36 Space
37 Sports
38 Super Power
39 Supernatural
40 Thriller
41 Vampire
42 Yaoi
43 Yuri
44 test with genres of Shingeki no Kyojin a.k.a. Attack of titan
Enter 0 to stop selection, enter 45 if you do not want this type of recommendation
Enter your choice : 44
Processing data...
Random Forest Classifier recommendation:
Warning : the maximum depth of the tree is limited to 10 in this particular test

                                Anime name
961 Shingeki no Kyojin Movie 2: Jiyuu no Tsubasa
Time to calculate it : 1.40 seconds

Processing data...
Support Vector Classifier recommendation:

                                Anime name
961 Shingeki no Kyojin Movie 2: Jiyuu no Tsubasa
Time to calculate it : 15.41 seconds

Processing data...
Bagging classifier using k-nearest neighbors vote recommendation:

                                Anime name
961 Shingeki no Kyojin Movie 2: Jiyuu no Tsubasa
Time to calculate it : 8.09 seconds

Processing data...
Bagging classifier using Decision Tree Classifier recommendation:
Warning : the maximum depth of the tree is limited to 10 in this particular test

                                Anime name
1009 Shingeki no Kyojin Movie 1: Guren no Yumiya
Time to calculate it : 2.51 seconds

```

## Funzione 12

```
Enter your choice : 12
Enter anime name as example Gintama : Gintama
For Gaussian Naive Bayes Classifier :
Yes
For Logistic Regression :
Yes
For KNeighbors Classifier :
Yes
For Decision Tree Classifier :
Yes
For Neural Network :
Yes
For Random Forest Classifier :
Yes
For Gradient Boosting Classifier :
Yes
```

### Funzione 13:

```
Enter your choice : 13
1 mean Success
0 means Flop

For Gaussian Naive Bayes Classifier :
      precision    recall  f1-score   support

     0       0.81     0.85     0.83     4015
     1       0.67     0.61     0.64     1994

 accuracy      0.77     0.77     0.77     6009
 macro avg     0.74     0.73     0.73     6009
weighted avg     0.76     0.77     0.77     6009

For Logistic Regression :
      precision    recall  f1-score   support

     0       1.00     1.00     1.00     4015
     1       0.99     0.99     0.99     1994

 accuracy      0.99     0.99     0.99     6009
 macro avg     0.99     0.99     0.99     6009
weighted avg     0.99     0.99     0.99     6009

For KNeighbors Classifier :
      precision    recall  f1-score   support

     0       0.83     0.91     0.87     4015
     1       0.78     0.64     0.70     1994

 accuracy      0.82     0.82     0.82     6009
 macro avg     0.81     0.77     0.79     6009
weighted avg     0.82     0.82     0.81     6009

For Decision Tree Classifier :
      precision    recall  f1-score   support

     0       1.00     1.00     1.00     4015
     1       1.00     1.00     1.00     1994

 accuracy      1.00     1.00     1.00     6009
 macro avg     1.00     1.00     1.00     6009
weighted avg     1.00     1.00     1.00     6009

For Neural Network :
      precision    recall  f1-score   support

     0       0.99     0.99     0.99     4015
     1       0.98     0.98     0.98     1994

 accuracy      0.99     0.99     0.99     6009
 macro avg     0.98     0.98     0.98     6009
weighted avg     0.99     0.99     0.99     6009

For Random Forest Classifier :
      precision    recall  f1-score   support

     0       1.00     1.00     1.00     4015
     1       1.00     1.00     1.00     1994

 accuracy      1.00     1.00     1.00     6009
 macro avg     1.00     1.00     1.00     6009
weighted avg     1.00     1.00     1.00     6009
```

## Apprendimento non Supervisionato: hard clustering (k-Means)

### Sommario

È stato usato anche un algoritmo di apprendimento non supervisionato senza feature-target che utilizza il metodo del clustering. La funzione prende in input il nome di un anime e restituisce i 10 più simili

### Strumenti utilizzati

L'algoritmo che lo implementa nel nostro caso è il k-Means. Questo utilizza un hard-clustering partizionale (che divide le classi in cluster) che permette di suddividere un insieme di oggetti in K gruppi sulla base dei loro attributi. Questo algoritmo mira a scegliere i centroidi che minimizzano l'inerzia, o il criterio della **somma dei quadrati all'interno del cluster**. [14]

### Decisioni di progetto

L'algoritmo è implementato nella libreria **sklearn.cluster** di **scikit** ed è stato testato con i parametri standard del metodo.

### Valutazione

Rispetto agli algoritmi supervisionati, questo risulta (nel dataset che utilizziamo) più lento, in quanto l'intero dataset deve essere diviso in mille cluster per avere un risultato soddisfacente. Ovviamente, tutto questo a discapito della velocità.

```
Enter your choice : 3
Enter anime name as example Gintama : Gintama
Similar anime using unsupervised learning :
Gintama
Detective Conan Movie 15: Quarter of Silence
Raiyantsuuri no Uta
Hayate no Gotoku! Heaven Is a Place on Earth
Hoshizora no Violin
Document Taiyou no Kiba Diagram
Jiu Se Lu
W: Wish Omake
Wakaranai Buta
Big Mouth Do Do
Soukou Kyojin Z-Knight
The Primitives: Bongo and Grunge
Wei Qi Shao Nian
Xi Yang Yang Yu Hui Tai Lang: Zhi Hu Hu Sheng Wei
Floating Material
Wana: Hakudaku Mamire no Houkago
```

## Conclusioni

Il programma esegue molteplici operazioni al fine di consigliare gli anime all'utente e poter porre una estrema navigabilità dei dati attraverso le caratteristiche degli anime, le quali sono intrecciate tra di loro (es.: un'anime può possedere da 1 a 43 generi).

## Riferimenti Bibliografici

- [1] [https://scikit-learn.org/stable/modules/naive\\_bayes.html#gaussian-naive-bayes](https://scikit-learn.org/stable/modules/naive_bayes.html#gaussian-naive-bayes)
- [2] [https://scikit-learn.org/stable/modules/linear\\_model.html#logistic-regression](https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression)
- [3] <https://scikit-learn.org/stable/modules/neighbors.html#classification>
- [4] <https://scikit-learn.org/stable/modules/tree.html#decision-trees>
- [5] <https://scikit-learn.org/stable/modules/ensemble.html#random-forests>
- [6] [https://scikit-learn.org/stable/modules/neural\\_networks\\_supervised.html#classification](https://scikit-learn.org/stable/modules/neural_networks_supervised.html#classification)
- [7] [https://scikit-learn.org/stable/auto\\_examples/applications/plot\\_cyclical\\_feature\\_engineering.html#gradient-boosting](https://scikit-learn.org/stable/auto_examples/applications/plot_cyclical_feature_engineering.html#gradient-boosting)
- [8] [https://scikit-learn.org/stable/modules/feature\\_extraction.html](https://scikit-learn.org/stable/modules/feature_extraction.html)
- [9] <https://scikit-learn.org/stable/modules/metrics.html#metrics>
- [10] <https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.pearsonr.html>

- [11] <https://scikit-learn.org/stable/modules/ensemble.html#random-forests>
- [12] <https://scikit-learn.org/stable/modules/svm.html#support-vector-machines>
- [13] <https://scikit-learn.org/stable/modules/ensemble.html#bagging-meta-estimator>
- [14] <https://scikit-learn.org/stable/modules/clustering.html#k-means><https://scikit-learn.org/stable/modules/clustering.html#k-means>
- [15] <https://artint.info/2e/html/ArtInt2e.Ch14.S3.html>
- [16] <https://pypi.org/project/Owlready2/>