Report Metodi Informatici per la Gestione Aziendale

Appello Settembre 2024 - Progetto base

Richard Rabi, n° matricola 869353

Obiettivo

Gli obiettivi principali del progetto sono:

1. **Analisi esplorativa dei dati** per ottenere una visione complessiva attraverso statistiche descrittive e analisi delle correlazioni.
2. **Ottimizzazione dell'algoritmo K-NN** testando diverse combinazioni di parametri (similarità, valore di K, user/item based) e selezionando la configurazione migliore basata su metriche di performance (MSE e RMSE).
3. **Completamento della matrice di rating** utilizzando la configurazione ottimale di K-NN.
4. **Segmentazione degli utenti** tramite clustering K-means con cosine similarity, per raggruppare gli utenti in base alle preferenze.
5. **Generazione di raccomandazioni personalizzate** (top-N items) per ciascun utente basate sui rating predetti.
6. **Confronto tra K-NN e Matrix Factorization** riempiendo la matrice di rating e valutando i risultati in termini di MSE e RMSE.

Descrizione dei dati

Il dataset rappresenta un insieme di recensione di libri.

Ci sono 10 colonne che rappresentano:

- **Rating** (float): valutazione del prodotto (da 1.0 a 5.0)
- **Title** (str): titolo della recensione dell'utente

- **Text** (str): testo della recensione dell'utente
- **Images** (lista): immagini che gli utenti pubblicano dopo aver ricevuto il prodotto. Ogni immagine ha dimensioni diverse (piccola, media, grande), rappresentate rispettivamente da small_image_url, medium_image_url e large_image_url
- **asin** (str): ID del prodotto
- **parent_asin** (str): ID principale del prodotto
- **user_id** (str): ID del recensore
- **timestamp** (int): data della recensione (tempo unix)
- **verified_purchase** (boolean): verifica dell'acquisto da parte dell'utente
- **helpful_vote** (int): voti utili della recensione

Esempio di un record

```
{'rating': [5.0],
'title': ['Updated: after 1st arrived damaged this one is per
'text': ['Updated: after first book arrived very damaged the
                    arrived in perfect condition.'],
'images': [[]],
'asin': ['0593235657'],
'parent_asin': ['0593235657'],
'user_id': ['AFKZENTNBQ7A7V7UXW5JJI6UGRYQ'],
'timestamp': [1640629604904],
'helpful_vote': [1],
'verified_purchase': [True]}
```

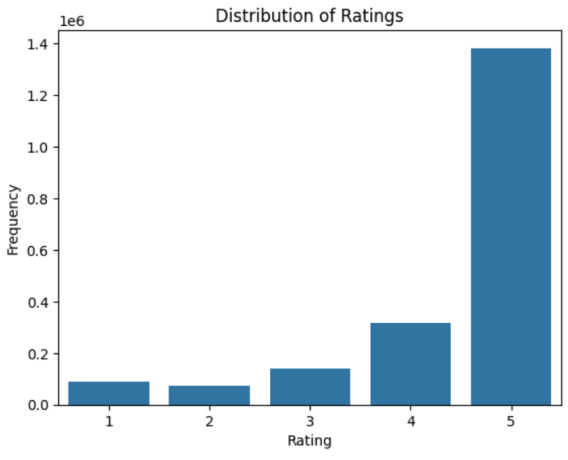
Risultati Analisi Esplorativa

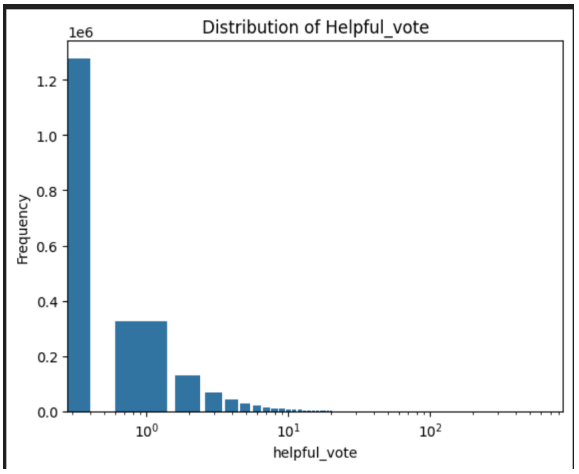
Sono stati caricati tutti i dati del dataset, per una maggiore facilità a livello computazionale i risultati di queste analisi sono state fatte selezionando in modo casuale da tutto il dataset 2 milioni di record.

Statistiche Descrittive

	rating	timestamp	helpful_vote
count	2.000000e+06	2.000000e+06	2.000000e+06
mean	4.415590e+00	1.457760e+12	1.796586e+00
std	1.065272e+00	1.414786e+11	1.823601e+01
min	1.000000e+00	8.507974e+11	0.000000e+00
25%	4.000000e+00	1.390685e+12	0.000000e+00
50%	5.000000e+00	1.468104e+12	0.000000e+00
75%	5.000000e+00	1.558505e+12	1.000000e+00
max	5.000000e+00	1.694473e+12	1.652600e+04

Alcune statistiche descrittive delle variabili numeriche del dataset. Le statistiche descrittive su timestamp sono poco informative a causa della natura della variabile.





- **Rating**: La distribuzione delle valutazioni è fortemente sbilanciata verso il valore massimo (5.0), suggerendo che le recensioni sono prevalentemente positive.
- **Timestamp**: La maggior parte delle recensioni è recente, con un numero crescente negli ultimi anni.
- **Helpful Vote**: La maggior parte delle recensioni non riceve voti utili, con poche recensioni che accumulano un numero elevato di voti.

Identificazione della configurazione ottimale dell'algoritmo K-NN per la predizione dei rating

Per poter avere dei dati piu accurati prendiamo dal nostro dataset:

Utenti con che ha fatto più di 30 recensioni e libri che hanno ricevuto più 30 recensioni, ottenendo:

	rating	timestamp	helpful_vote
count	1045.000000	1.045000e+03	1045.000000
mean	4.176077	1.414329e+12	6.324402
std	0.993080	1.548275e+11	51.617060
min	1.000000	9.494678e+11	0.000000
25%	4.000000	1.327508e+12	0.000000
50%	4.000000	1.438137e+12	0.000000
75%	5.000000	1.523711e+12	2.000000
max	5.000000	1.682090e+12	1249.000000

- **Valutazioni**: Le valutazioni tendono ad essere molto positive, con la maggior parte delle recensioni che ottengono 4 o 5 stelle.
- **Voti utili**: Sebbene molte recensioni non ricevano voti utili (mediana pari a 0), ci sono alcune recensioni che ricevono un numero eccezionalmente elevato di voti utili, come indicato dal valore massimo di 1249.

- **Distribuzione temporale**: Le recensioni coprono un arco temporale significativo, dal 2000 al 2023, con un aumento nel numero di recensioni negli ultimi anni.

Grid Search e Ottimizzazione del K-NN

Parametri dei numero di vicini da esplorare : 10, 20, 30, 40, 50, 70, 80, 100.

Metriche di similarità: cosine, Pearson.
User based: True, false.

Per configurazione Ottimale del K-NN effettuiamo su questi dati filtrati una Grid Search per l'algoritmo K-NN ottenendo:

Best RMSE: 0.9925
Best MSE: 0.9861
Best parameters for RMSE: {'k': 10, 'sim_options': {'name': 'cosine', 'user_based': True}}
Best parameters for MSE: {'k': 10, 'sim_options': {'name': 'cosine', 'user_based': True}}

Concludendo che il miglior modello trovato utilizza:

- **K-NN con** `k=10` vicini.
- **Similarità coseno** per misurare la similarità tra utenti.
- **Basato sugli utenti** (`user_based=True`), il che significa che le raccomandazioni si basano su utenti con preferenze simili.
- Il modello ha prodotto errori relativamente bassi, con un RMSE vicino a 1, indicando che le sue predizioni di rating sono piuttosto accurate.

Filling della matrice di rating con la configurazione ottimale

Prendiamo la miglior configurazione ottimale e la utilizziamo per popolare la matrice.

Per la matrice di filling è stata creata una una copia del matrix delle valutazioni originali e siamo andati ad aggiungere le nuove righe con le righe predette.

Alcune previsioni:

Prediction for user AETF2GTK4HYAQMDR7Q66AUQWKL4A and item 0761463275: 4.2
Prediction for user AETF2GTK4HYAQMDR7Q66AUQWKL4A and item 0399155341: 4.2
Prediction for user AETF2GTK4HYAQMDR7Q66AUQWKL4A and item B00JO8PEN2: 4.2
Prediction for user AETF2GTK4HYAQMDR7Q66AUQWKL4A and item 0440221668: 3.0

Trasformazione dei valori predetti in pivot table :

```
prediction_matrix_pivot = prediction_matrix.pivot_table(index='user_id', columns='asin', values='rating', aggfunc='mean').round(1)
prediction_matrix_pivot.head(20)
```
✓ 0.2s

asin	0060248025	0060254920	0060530944	0060887184	0060899220	0060936223	0060938455	0061120073	0061122416	0061124958	...	B08KH8YT2S	B08SLW9ZL1	B08WCFQVR8	B08WGR3YSK
user_id															
AE27NHMO2VWDSCCEP4A53GTX7YIA	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	3.0	4.2	4.2
AE2C6ZDFCI2EW5EDM53RYKP7WHNQ	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE2FGO5WED4FEC53K57X4DHCEPIQ	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE2TPEBBWEDH4MPALT3JZZATUSVQ	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE2UE4MFTXIDF6K3CJGS6DY5L2FA	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE2W3CFIRW42PHPYRNRUCIR4X6BA	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE32HKESSYBEQJID7PWYW3PUUP7Q	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE3DHH4ITYD373F5H5N6WLOQFZBA	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE3DICKCDVG5C6AQQRWRICN7OFSA	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE3EAPQJCJI4AIY7I7SDAWLVSZOQ	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE3GRCJPL7IXD22AJODJBC6OGZDA	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE3VKA2EQVPVCNNMRWA5CI3ZSKWA	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE43JDTOAHNTZIMRP5ROITQP2RVQ	4.2	4.2	4.2	4.2	4.2	5.0	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE47GPOIIKR2AQMLQPRAHZ6C7IDQ	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE4N4QNMKXOAAJYRBDQMTITDFQSA	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE64KJRO3QOHPHK47RHSD4LDELHQ	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE6NGIV5TDARQTF36EZTKLBDDIWA	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE6PG6NUBEQ7AS3MTWZJEB3VUVDQ	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE6TDCUOKCSCRL5ZHWKEL7QHOA3A	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2
AE6U2ZCGCBRDU3TT3PVH44ITRLOA	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	4.2	...	4.2	4.2	4.2	4.2

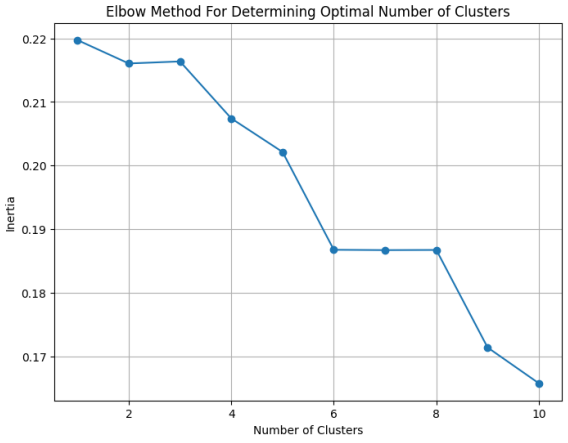
Otteniamo valori molto simili dovuto allo sbilanciamento dei dati verso il voto 4-5.

Segmentazione degli utenti in base alle preferenze

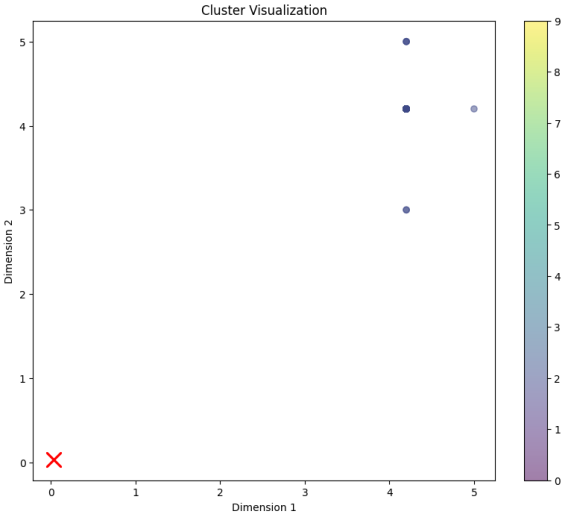
Algoritmo di clustering K-MEANS con cosine similarity.

Per trovare il numero di clustering ho utilizzato l'algoritmo **K-Means** con la **similarità coseno**. Le funzioni eseguono il clustering sui dati, determinano il numero ottimale di cluster tramite il **Metodo del Gomito**.

L'algoritmo K-means è stato applicato sulla matrice di similarità per segmentare gli utenti in 10 cluster distinti.



Andamento del grafico: Il metodo del gomito cerca il punto in cui l'inertia non diminuisce più in modo significativo all'aumentare del numero di cluster. In questo grafico, sembra che il "gomito" sia attorno a 6 cluster. Questo suggerisce che 6 cluster potrebbero essere un buon punto di partenza, dato che oltre questo valore non ci sono riduzioni significative dell'inertia.



Distribuzione dei cluster: Nel secondo grafico, i cluster sembrano ben distinti, con chiari centri di cluster (croci rosse). Alcuni cluster sembrano più densi e concentrati rispetto ad altri, il che può indicare che i dati in quei cluster sono più omogenei.

Creazione per ogni utente della lista degli n items (top k items) da consigliare

Per ogni utente generiamo le migliori
n raccomandazioni per ciascun utente in base alle predizioni dei rating.
In particolare il sistema crea un dizionario che associa a ogni utente i
top n prodotti con i rating predetti più alti, ordinandoli in ordine decrescente.

- Ordina i prodotti predetti per l'utente in base al rating stimato (in ordine decrescente).
- Seleziona i primi **n** prodotti con i rating più alti.

Esempio Top 10 recommendations for user **AEPMHVACMQKZNMSLWIFQYC5T6LZA**:

	user_id	asin	rating
14217	AEPMHVACMQKZNMSLWIFQYC5T6LZA	1594633665	5.000000
858	AEPMHVACMQKZNMSLWIFQYC5T6LZA	0307744434	5.000000
177	AEPMHVACMQKZNMSLWIFQYC5T6LZA	B077GRGYJ4	5.000000
14607	AEPMHVACMQKZNMSLWIFQYC5T6LZA	1476729093	5.000000
14391	AEPMHVACMQKZNMSLWIFQYC5T6LZA	0684801221	5.000000
14057	AEPMHVACMQKZNMSLWIFQYC5T6LZA	0062073486	5.000000
14165	AEPMHVACMQKZNMSLWIFQYC5T6LZA	014310974X	5.000000
14380	AEPMHVACMQKZNMSLWIFQYC5T6LZA	0142410705	5.000000
14548	AEPMHVACMQKZNMSLWIFQYC5T6LZA	0385741278	4.176077
14541	AEPMHVACMQKZNMSLWIFQYC5T6LZA	B06XNPKBL9	4.176077

Filling della matrice di rating attraverso Matrix Factorization in aggiunta a K-NN e confronto dei risultati ottenuti in termini di MSE e RMSE

1. MSE (Mean Squared Error)

- **KNN - MSE: 1.2242**
- **SVD - MSE: 1.1628**

L'MSE del modello SVD è leggermente inferiore rispetto al modello KNN, il che significa che, in media, SVD predice i rating in modo più accurato rispetto a KNN.

2. RMSE (Root Mean Squared Error)

- **KNN - RMSE: 1.1064**
- **SVD - RMSE: 1.0783**

Anche in questo caso, l'RMSE di SVD è più basso rispetto a KNN, suggerendo che il modello SVD fa predizioni più accurate.

Conclusione:

SVD offre una migliore performance in termini di accuratezza rispetto a KNN sia per MSE che per RMSE. La differenza non è molto grande, ma indica che il modello basato sulla factorizzazione della matrice (SVD) gestisce meglio la predizione dei rating rispetto al KNN, probabilmente perché SVD cattura meglio le relazioni latenti tra utenti e prodotti, mentre KNN si basa esclusivamente sulla similarità diretta tra utenti o item.