

Esercitazione 8

1. Utilizzare i codici Python **collaborative_filtering.py** e **recommendation_data.py**, disponibili sulla piattaforma.
 - a. Eseguire il programma **collaborative_filtering** e verificare che il risultato sia quello atteso, confrontando l'output con quello illustrato nella Tabella 2.2 di [1]. Stampare anche gli indici di similarità di tutti gli utenti per Toby e il prodotto tra l'indice di similarità e ogni film.
 - b. Utilizzare la distanza euclidea per misurare la similarità tra gli utenti. Confrontare il risultato ottenuto rispetto al caso precedente.
 - c. Modificare il codice usando la seguente stima di valutazione

$$\hat{r}_{ki} = \mu_k + \frac{\sum_{j \in N(i)} \text{sim}(k, j)(r_{ji} - \mu_j)}{\sum_{j \in N(i)} \text{sim}(k, j)},$$

dove \hat{r}_{ki} è la raccomandazione fornita all'utente k per il prodotto i ,
 μ_k è il voto medio dell'utente k , r_{ji} è il voto espresso dall'utente j per il prodotto i ,
 $N(i)$ è l'insieme di tutti gli utenti che hanno votato il prodotto i ,
 $\text{sim}(i, j)$ è la similarità tra gli utenti i e j , misurata secondo la formula di Pearson.

Confrontare i risultati ottenuti rispetto all'esercizio 1.

[1] Toby Segaran, *Programming collective intelligence: building smart web 2.0 applications*. O' Reilly Media, Inc.", 2007. Una copia elettronica del testo è disponibile sulla piattaforma e-learning.

2. Si consideri il sistema di raccomandazione basato sulla SVD troncata implementato nel codice Python **SVD_rs.py**. Esso è formulato sulla base delle valutazioni di alcuni utenti per sei stagioni:

	Ryne	Erin	Nathan	Pete
season 1	5	5	0	5
season 2	5	0	3	4
season 3	3	4	0	3
season 4	0	0	5	3
season 5	5	4	4	5
season 6	5	4	5	5

- a. Nel codice, sulla base della SVD troncata con $k=2$, sono rappresentati gli utenti e i prodotti in uno spazio a due dimensioni. In tale spazio si lavora per fornire raccomandazioni al nuovo utente Luke. Luke aveva fornito queste valutazioni: [5 5 0 0 0 5]. Quali utenti risultano più simili a Luke?
- b. Si ripeta l'esercizio con $k=3$ e $k=4$. Quali utenti risultano più simili a Luke?
3. Utilizzare il codice **compression_svd.py** disponibile sulla piattaforma, calcolare la porzione di energia conservata con $k=5, 20, 100$, mediante la formula

$$\frac{\sum_{j=1}^k \sigma_j}{\sum_{j=1}^n \sigma_j}$$

dove σ_i sono i valori singolari della matrice X ed n è il numero totale dei valori singolari.

FACOLTATIVO. Determinare il valore minimo di k per cui si conserva l'80% dell'energia totale.