**Machine Learning**

**(Assignment for IITP –BSE Course)**

**A1: Consider the iris data set and perform clustering on it. The iris dataset is of 150 instances**

**represented by 4 features. The last column in the iris data set represents the label information.**

**Group the dataset in an unsupervised way i.e., without considering the label information.**

**Perform the clustering algorithm using the following algorithm i) K‐means, ii) Bottom‐up**

**Hierarchical Clustering (BUHC) using single linkage, and iii) Fuzzy C means (FCM) (nonoverlapping).**

**Provide the number of cluster K as K=3, 4 and 5. Use the Silhouette score and**

**Adjusted Rank Index (ARI) for evaluating the generated clusters. Report the score obtained**

**below in a tabular form.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Silhouette Score | | | ARI | | |
|  | K‐means | BUHC | FCM | K‐means | BUHC | FCM |
| K=3 | 0.4599 | 0.50464 | 0.4584 | 0.6201 | 0.5583 | 0.6303 |
| K=4 | 0.3882 | 0.4067 | 0.4016 | 0.4440 | 0.5522 | 0.5877 |
| K=5 | 0.3423 | 0.3424 | 0.3448 | 0.4324 | 0.5517 | 0.5617 |

**A2:**

**1.Load data on colab from the following GitHub links:**

ratings="https://github.com/couturierc/tutorials/raw/master/recommender\_system/data/ratings.csv"

movies="https://github.com/couturierc/tutorials/raw/master/recommender\_system/data/movies.csv"

df\_ratings = pd.read\_csv(ratings, sep=',')

df\_ratings.columns = ['userId', 'itemId', 'rating', 'timestamp']

df\_movies = pd.read\_csv(movies, sep=',')

df\_movies.columns = ['itemId', 'title', 'genres']

**2.Print first 20 rows for each dataset.**

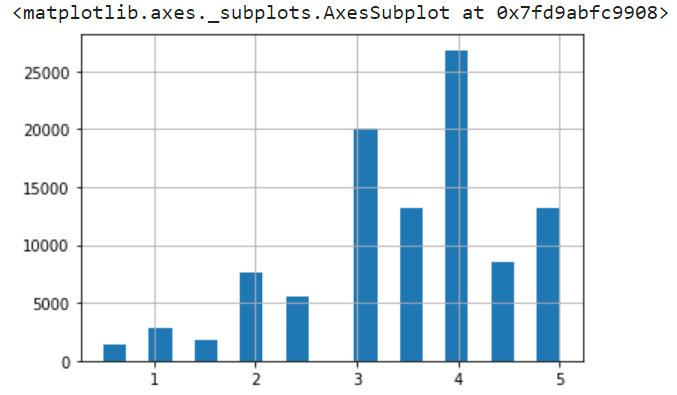
**Movies:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.NO** | **itemId** | **title** | **genres** |
| **0** | 1 | Toy Story (1995) | Adventure|Animation|Children|Comedy|Fantasy |
| **1** | 2 | Jumanji (1995) | Adventure|Children|Fantasy |
| **2** | 3 | Grumpier Old Men (1995) | Comedy|Romance |
| **3** | 4 | Waiting to Exhale (1995) | Comedy|Drama|Romance |
| **4** | 5 | Father of the Bride Part II (1995) | Comedy |
| **5** | 6 | Heat (1995) | Action|Crime|Thriller |
| **6** | 7 | Sabrina (1995) | Comedy|Romance |
| **7** | 8 | Tom and Huck (1995) | Adventure|Children |
| **8** | 9 | Sudden Death (1995) | Action |
| **9** | 10 | GoldenEye (1995) | Action|Adventure|Thriller |
| **10** | 11 | American President, The (1995) | Comedy|Drama|Romance |
| **11** | 12 | Dracula: Dead and Loving It (1995) | Comedy|Horror |
| **12** | 13 | Balto (1995) | Adventure|Animation|Children |
| **13** | 14 | Nixon (1995) | Drama |
| **14** | 15 | Cutthroat Island (1995) | Action|Adventure|Romance |
| **15** | 16 | Casino (1995) | Crime|Drama |
| **16** | 17 | Sense and Sensibility (1995) | Drama|Romance |
| **17** | 18 | Four Rooms (1995) | Comedy |
| **18** | 19 | Ace Ventura: When Nature Calls (1995) | Comedy |
| **19** | 20 | Money Train (1995) | Action|Comedy|Crime|Drama|Thriller |

**Ratings:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.NO** | **userId** | **itemId** | Rating | **timestamp** |
| **0** | 1 | 1 | 4.0 | 964982703 |
| **1** | 1 | 3 | 4.0 | 964981247 |
| **2** | 1 | 6 | 4.0 | 964982224 |
| **3** | 1 | 47 | 5.0 | 964983815 |
| **4** | 1 | 50 | 5.0 | 964982931 |
| **5** | 1 | 70 | 3.0 | 964982400 |
| **6** | 1 | 101 | 5.0 | 964980868 |
| **7** | 1 | 110 | 4.0 | 964982176 |
| **8** | 1 | 151 | 5.0 | 964984041 |
| **9** | 1 | 157 | 5.0 | 964984100 |
| **10** | 1 | 163 | 5.0 | 964983650 |
| **11** | 1 | 216 | 5.0 | 964981208 |
| **12** | 1 | 223 | 3.0 | 964980985 |
| **13** | 1 | 231 | 5.0 | 964981179 |
| **14** | 1 | 235 | 4.0 | 964980908 |
| **15** | 1 | 260 | 5.0 | 964981680 |
| **16** | 1 | 296 | 3.0 | 964982967 |
| **17** | 1 | 316 | 3.0 | 964982310 |
| **18** | 1 | 333 | 5.0 | 964981179 |
| **19** | 1 | 349 | 4.0 | 964982563 |

**3.Give a graphical representation of how the total number of users varies from rating 1 to 5.**



**4.Print max, min, and average ratings for userId 10.**

count 140.000000

mean 3.278571

std 1.175700

min 0.500000

25% 3.000000

50% 3.500000

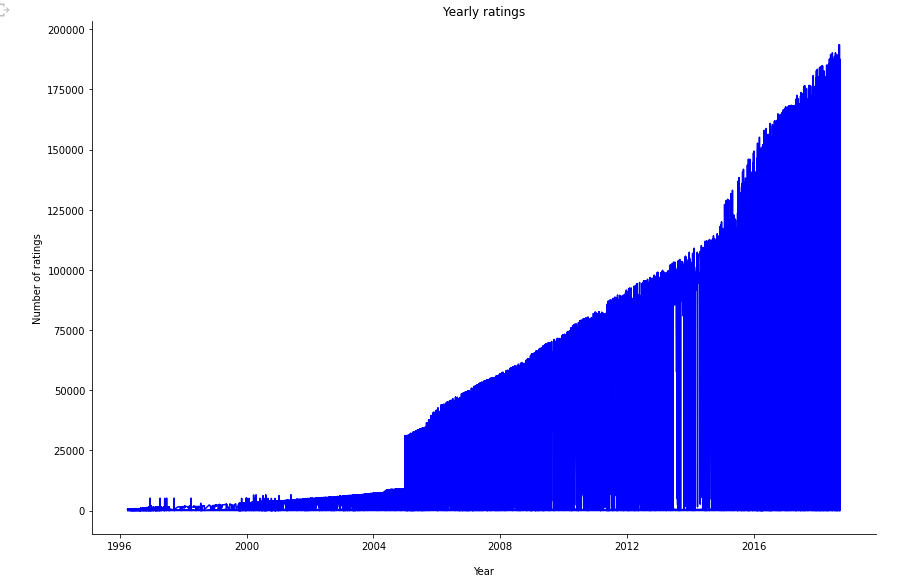
75% 4.000000

max 5.000000

Name: rating, dtype: float64

**5.Give a graphical representation of how the total number of users (based on the number of**

**posted ratings) varies from time to time.**



**6.Convert the Rating data to matrix form where columns indicate ‘userId' and row indicates**

**‘itemId’ and each entry indicates rating values.**

itemId 1 2 3 4 ... 193583 193585 193587 193609

userId ...

1 4.0 NaN 4.0 NaN ... NaN NaN NaN NaN

2 NaN NaN NaN NaN ... NaN NaN NaN NaN

3 NaN NaN NaN NaN ... NaN NaN NaN NaN

4 NaN NaN NaN NaN ... NaN NaN NaN NaN

5 4.0 NaN NaN NaN ... NaN NaN NaN NaN

... ... ... ... ... ... ... ... ... ...

606 2.5 NaN NaN NaN ... NaN NaN NaN NaN

607 4.0 NaN NaN NaN ... NaN NaN NaN NaN

608 2.5 2.0 2.0 NaN ... NaN NaN NaN NaN

609 3.0 NaN NaN NaN ... NaN NaN NaN NaN

610 5.0 NaN NaN NaN ... NaN NaN NaN NaN

[610 rows x 9724 columns]

**7.For instance, print rating for userId=10 for movies with itemId 1 to 30.**

itemId

1 0.0

2 0.0

3 0.0

4 0.0

5 0.0

6 0.0

7 0.0

8 0.0

9 0.0

10 0.0

11 0.0

12 0.0

13 0.0

14 0.0

15 0.0

16 0.0

17 0.0

18 0.0

19 0.0

20 0.0

21 0.0

22 0.0

23 0.0

24 0.0

25 0.0

26 0.0

27 0.0

28 0.0

29 0.0

30 0.0

Name: 10, dtype: float64

**8.Save the movie ids for userId: 21.**

Int64Index([ 1, 2, 3, 4, 5, 6, 7, 8,

9, 10,

...

193565, 193567, 193571, 193573, 193579, 193581, 193583, 193585,

193587, 193609],

dtype='int64', name='itemId', length=9724)

**9.Model 1:** Apply Approximate SVD with stochastic gradient descend (SGD) on the dataset with

n\_factors = 10, # number of factors

alpha = .01, # learning rate

n\_epochs = 3, # number of iteration of the SGD procedure

MAE: 2.4

**Model 2:** Apply Approximate SVD with stochastic gradient descend (SGD) on the dataset with

n\_factors = 20, # number of factors

alpha = .01, # learning rate

n\_epochs = 6, # number of iteration of the SGD procedure

MAE: 3.4

**Model 3:** Apply Approximate SVD with stochastic gradient descend (SGD) on the dataset with

n\_factors = 40, # number of factors

alpha = .01, # learning rate

n\_epochs = 18, # number of iteration of the SGD procedure

MAE: 3.8