CSC4008 - Assignment 4

Yohandi

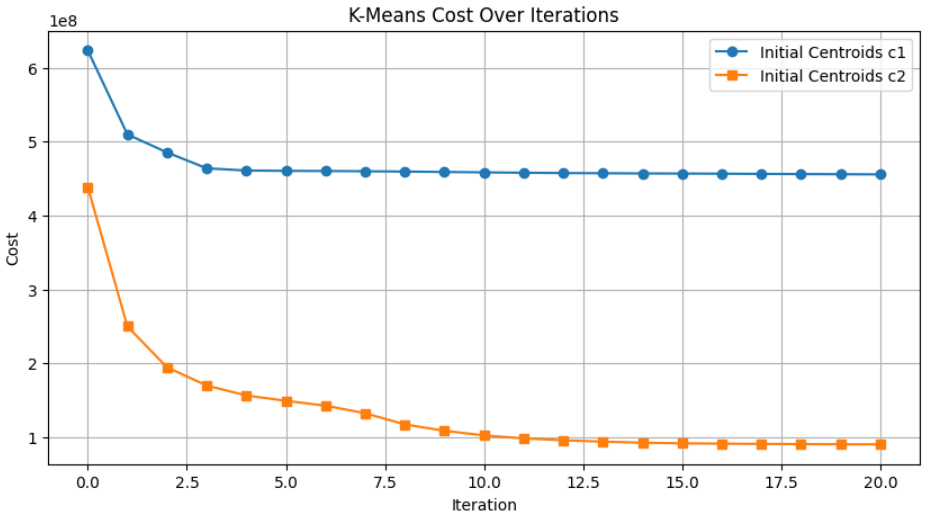
December 18, 2023

1. Implementing k-means on Spark

Part (a)

```
Iteration 0: Cost = 623660345.3064115
Iteration 1: Cost = 509862908.2975454
Iteration 2: Cost = 485480681.8720084
Iteration 3: Cost = 463997011.68501294
Iteration 4: Cost = 460969266.5729968
Iteration 5: Cost = 460537847.9827684
Iteration 6: Cost = 460313099.6535447
Iteration 7: Cost = 460003523.8894081
Iteration 8: Cost = 459570539.3177352
Iteration 9: Cost = 459021103.3422909
Iteration 10: Cost = 458490656.1919812
Iteration 11: Cost = 457944232.5879751
Iteration 12: Cost = 457558005.1986773
Iteration 13: Cost = 457290136.35230196
Iteration 14: Cost = 457050555.05956286
Iteration 15: Cost = 456892235.61535585
Iteration 16: Cost = 456703630.7370345
Iteration 17: Cost = 456404203.01897514
Iteration 18: Cost = 456177800.541994
Iteration 19: Cost = 455986871.0273468
Iteration 20: Cost = 455729268.3551448
Iteration 0: Cost = 438747790.02791756
Iteration 1: Cost = 249803933.62600276
Iteration 2: Cost = 194494814.40631256
Iteration 3: Cost = 169804841.4515432
Iteration 4: Cost = 156295748.8062759
Iteration 5: Cost = 149094208.10896596
Iteration 6: Cost = 142508531.61961532
Iteration 7: Cost = 132303869.40652987
Iteration 8: Cost = 117170969.83719075
Iteration 9: Cost = 108547377.17857003
Iteration 10: Cost = 102237203.3179959
Iteration 11: Cost = 98278015.74975666
Iteration 12: Cost = 95630226.12177408
Iteration 13: Cost = 93793314.051193
Iteration 14: Cost = 92377131.96821073
Iteration 15: Cost = 91541606.25423889
Iteration 16: Cost = 91045573.83042458
Iteration 17: Cost = 90752240.10140811
```

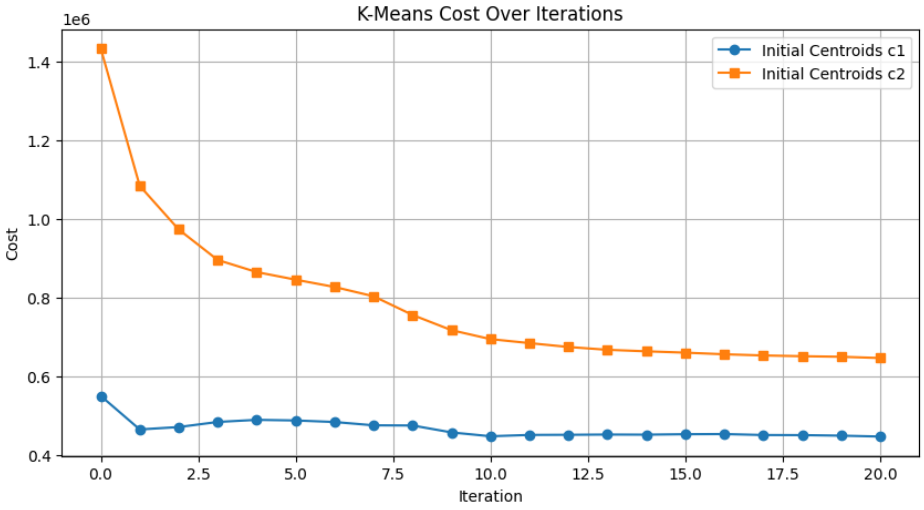
```
Iteration 18: Cost = 90470170.18122731
Iteration 19: Cost = 90216416.1756313
Iteration 20: Cost = 90162390.91041416
percentage change (%) for C1: 26.48391714456056
percentage change (%) for C2: 76.69795594605947
```



Part (b)

```
Iteration 0: Cost = 550117.1419999995
Iteration 1: Cost = 464829.2684039448
Iteration 2: Cost = 470934.15384668094
Iteration 3: Cost = 483874.81628509343
Iteration 4: Cost = 489234.2347883463
Iteration 5: Cost = 487664.6926267904
Iteration 6: Cost = 483718.6659285149
Iteration 7: Cost = 475337.9476330566
Iteration 8: Cost = 474871.96654965664
Iteration 9: Cost = 457244.7897417528
Iteration 10: Cost = 447493.1956040521
Iteration 11: Cost = 450891.8358047706
Iteration 12: Cost = 451232.57747569657
Iteration 13: Cost = 451860.12588546576
Iteration 14: Cost = 451567.2235891488
Iteration 15: Cost = 452710.05209994374
Iteration 16: Cost = 453078.22696184996
Iteration 17: Cost = 450646.1355620941
Iteration 18: Cost = 450419.9701134367
Iteration 19: Cost = 449009.59037188545
Iteration 20: Cost = 446771.2835417304
Iteration 0: Cost = 1433739.3099999954
Iteration 1: Cost = 1084488.7769648773
Iteration 2: Cost = 973431.7146620404
Iteration 3: Cost = 895934.5925630709
Iteration 4: Cost = 865128.3352940814
Iteration 5: Cost = 845846.647031348
Iteration 6: Cost = 827219.5827561249
Iteration 7: Cost = 803590.3456011118
```

```
Iteration 8: Cost = 756039.5172761207
Iteration 9: Cost = 717332.9025432297
Iteration 10: Cost = 694587.9252526882
Iteration 11: Cost = 684444.5019967903
Iteration 12: Cost = 674574.7475478561
Iteration 13: Cost = 667409.4699160281
Iteration 14: Cost = 663556.6278215044
Iteration 15: Cost = 660162.7772287563
Iteration 16: Cost = 656041.3222947121
Iteration 17: Cost = 653036.7540731612
Iteration 18: Cost = 651112.4262522729
Iteration 19: Cost = 649689.0131843555
Iteration 20: Cost = 646481.1586157621
percentage change (%) for C1: 18.65492611679923
percentage change (%) for C2: 51.554099102389095
```



2. Recommender Systems

Part (a):

The non-normalized user similarity matrix T is defined as $T = R \cdot R^T$. The element T_{ij} of matrix T represents the number of items that both user i and user j like. This is because each entry in $R \cdot R^T$ is a dot product of two user's ratings, which counts the number of items that both users have in common (i.e., both users have a '1' for the same item).

$$T_{ij} = \sum_k R_{ik} \cdot R_{jk}$$

This means that T_{ij} is the sum of products of ratings from user i and user j across all items. If $i \neq j$, then T_{ij} represents the intersection of items liked by both users, which can be related to the concept of node degrees and paths in a bipartite graph. Then, when $i = j$, it simply denotes the number of items liked by user i.

Part (b):

Let's start with the definition of cosine similarity and how it relates to the matrices R, P, and Q.

Cosine similarity between two vectors **a** and **b** is given by:

$$cos - sim(\mathbf{a}, \mathbf{b}) = \frac{\mathbf{a} \cdot \mathbf{b}}{\|\mathbf{a}\|\|\mathbf{b}\|}$$

Where $\mathbf{a} \cdot \mathbf{b}$ is the dot product of vectors $\mathbf{a}$ and $\mathbf{b}$, and $\|\mathbf{a}\|$ and $\|\mathbf{b}\|$ are the Euclidean norms (magnitudes) of the vectors.

In the context of the item similarity matrix S_I, each vector $\mathbf{a}$ and $\mathbf{b}$ corresponds to the rows of matrix R representing item ratings by all users. The normalization factors come from the diagonal matrix Q, where each diagonal element Q_{ii} is the square root of the number of users that liked item i. Hence, $Q^{-1/2}$ normalizes the vectors to unit length in terms of cosine similarity.

The element-wise calculation of S_I is as follows:

$$(S_I)_{ij} = \frac{(R^T R)_{ij}}{\sqrt{(Q)_{ii}}\sqrt{(Q)_{jj}}}$$

Which can be rewritten using matrix operations as:

$$S_I = Q^{-1/2} R^T R Q^{-1/2}$$

The matrix S_U for user similarity follows the same logic, with P being the diagonal matrix where each diagonal element P_{ii} is the square root of the number of items liked by user i. Thus, $P^{-1/2}$ normalizes the user rating vectors.

The element-wise calculation of S_U would be:

$$(S_U)_{ij} = \frac{(RR^T)_{ij}}{\sqrt{(P)_{ii}}\sqrt{(P)_{jj}}}$$

And in matrix notation:

$$S_U = P^{-1/2} R R^T P^{-1/2}$$

These matrix operations ensure that the similarities are scaled between 0 and 1, representing the cosine of the angle between the rating vectors, which is the essence of cosine similarity in collaborative filtering.

Part (c):

The recommendation matrix Γ is computed differently for user-user and item-item collaborative filtering. For the item-item case, as given by the hint, it is:

$$\Gamma = RQ^{-1/2} R^T R Q^{-1/2}$$

This expression computes the predicted ratings by weighing the similarities between items and the user's existing ratings.

For the user-user collaborative filtering case, the expression for Γ would be analogous, considering the similarities between users:

$$\Gamma = P^{-1/2} R R^T P^{-1/2} R$$

This predicts a user's rating for an item based on the weighted average of ratings from similar users.

These are the expressions derived from the given information and the hint for the item-item case. If you have more specific questions about each part or need further assistance with the LaTeX code, feel free to ask!

Part (d):

```
User to User Collaborative Filtering:
FOX 28 News at 10pm
Family Guy
2009 NCAA Basketball Tournament
NBC 4 at Eleven
Two and a Half Men
Item to Item Collaborative Filtering:
```

FOX 28 News at 10pm
Family Guy
NBC 4 at Eleven
2009 NCAA Basketball Tournament
Access Hollywood