**Homework-9**

|  |
| --- |
| Team#: \_6\_\_  Team Member-1:\_\_\_Luis\_\_Fuentes\_\_\_\_\_\_\_\_\_\_\_Member’s Contribution (in %) \_33\_\_  Team Member-2:\_\_\_Bhagwat Dhakal\_ \_\_\_\_\_\_\_\_Member’s Contribution (in %) \_33\_\_  Team Member-3:\_\_\_Ishmam Rahman\_\_\_\_\_\_\_\_\_Member’s Contribution (in %) \_33\_\_ |
| **Submission**   1. Answer the following questions 2. **Problem-1 and Problem-2:** Rename your doc file ““HW8\_YourTeam#” (e.g., HW8\_Team1.doc). 3. **Problem-3** Rename your R file ““HW9\_YourTeam#” HW9\_Team1.rmd/HW9\_Team1.ipynb). 4. Upload the files to Canvas. |

**Problem-1: K-Means Clustering [30 points]**

Suppose that the data mining task is to cluster the following points (with (x, y) representing location): P1(20, 100), P2(20, 50), P3(80, 40), P4(50, 80), P5(70, 50), P6(60, 40), P7(10, 20), P8(40, 90). The distance function is Euclidean distance.

Suppose initially centroids are P1, P4, and P7. Use the k-means algorithm to compute:

1. **The three clusters (K = 3) and their centers after the first iteration. Compute the total Sum of Squared Error for this iteration. [10 points]**

Answer:  
Part 1: First Iteration

C1: P1(20, 100)

C2: P4(50, 80)

C3: P7(10, 20)

Calculating distances

1. P1(20, 100):

- Distance to C1: 0

- Distance to C2: sqrt(50-20)^2 + (80-100)^2) = sqrt(900 + 400) = sqrt(1300) = 36.06

- Distance to C3: 80.62

- Closest: C1

2. P2(20, 50):

- Distance to C1: 50

- Distance to C2: 42.43

- Distance to C3: 31.62

- Closest: C3

3. P3(80, 40):

- Distance to C1: 84.85

- Distance to C2: 50

- Distance to C3: 72.80

- Closest: C2

4. P4(50, 80):

- Distance to C1: 36.06

- Distance to C2: 0

- Distance to C3: 72.11

- Closest: C2

5. P5(70, 50):

- Distance to C1: 70.71

- Distance to C2: 36.06

- Distance to C3: 67.08

- Closest: C2

6. P6(60, 40):

- Distance to C1: 72.11

- Distance to C2: 41.23

- Distance to C3: 53.85

- Closest: C2

7. P7(10, 20):

- Distance to C1: 80.62

- Distance to C2: 72.11

- Distance to C3: 0

- Closest: C3

8. P8(40, 90):

- Distance to C1: 22.36

- Distance to C2: 14.14

- Distance to C3: 76.16

- Closest: C2

Cluster Assignments after 1st iteration:

- C1 :P1

- C2 :P3, P4, P5, P6, P8

- C3 :P2, P7

Updating centroids by calculating the mean of each cluster

- New C1: Mean of P1 = P1 = (20, 100)

- New C2: Mean of P3, P4, P5, P6, P8

- x: (80 + 50 + 70 + 60 + 40)/5 = 300/5 = 60

- y: (40 + 80 + 50 + 40 + 90)/5 = 300/5 = 60

- So, new C2: (60, 60)

- New C3: Mean of P2,P7

- x: (20 + 10)/2 = 15

- y: (50 + 20)/2 = 35

- So, new C3: (15, 35)

New Centroids after 1st iteration:

- C1: (20, 100)

- C2: (60, 60)

- C3: (15, 35)

Calculating SSE for the first iteration

- Cluster 1:

- P1 to C1: 0

- Cluster 2:

- P3 to C2: sqrt((80-60)^2 + (40-60)^2) = sqrt(400 + 400) = sqrt(800) ≈ 28.28^2 = 800

- P4 to C2: 500

- P5 to C2: 200

- P6 to C2: 400

- P8 to C2: 1300

- Total for Cluster 2: 800 + 500 + 200 + 400 + 1300 = 3200

- Cluster 3:

- P2 to C3: 250

- P7 to C3: 250

- Total for Cluster 3: 250 + 250 = 500

Total SSE after 1st iteration: 0+3200+ 500 = 3700

1. **The three clusters (K = 3) and their centers after the second iteration. Compute the total Sum of Squared Error for this iteration. [10 points]**

Answer:

Current Centroids

- C1: (20, 100)

- C2: (60, 60)

- C3: (15, 35)

Reassign points to the nearest centroid.

1. P1(20, 100)

- Distance to C1: 0

- Distance to C2: 56.57

- Distance to C3: 65.19

- Closest: C1

2. P2(20, 50)

- Distance to C1: 50

- Distance to C2: 41.23

- Distance to C3: 15.81

- Closest: C3

3. P3(80, 40)

- Distance to C1:85

- Distance to C2: 28.28

- Distance to C3: 65.19

- Closest: C2

4. P4(50, 80):

- Distance to C1: 36.06

- Distance to C2: 22.36

- Distance to C3: 57.01

- Closest: C2

5. P5(70, 50):

- Distance to C1:.71

- Distance to C2: 14.14

- Distance to C3: 57.01

- Closest: C2

6. P6(60, 40):

- Distance to C1: 72.11

- Distance to C2: 20

- Distance to C3: 45.28

- Closest: C2

7. P7(10, 20):

- Distance to C1: 80.62

- Distance to C2: 64.03

- Distance to C3: 15.81

- Closest: C3

8. P8(40, 90):

- Distance to C1: 22.36

- Distance to C2: 36.06

- Distance to C3: 60.41

- Closest: C1

Cluster Assignments after 2nd iteration:

C1: (20,100): P1, P8

C2: (60,60): P3, P4, P5, P6

C3: (15,35): P2, P7

Step 2: Updating centroids by calculating the mean of each cluster.

- New C1: Mean of P1,P8

- x: (20 + 40)/2 = 30

- y: (100 + 90)/2 = 95

- new C1: (30, 95)

- New C2: Mean of P3,P4,P5,P6

- x: 65

- y: 52.5

- new C2: (65, 52.5)

- New C3: Mean of P2,P7

- x:15

- y: 35

- So, new C3: (15, 35)

New Centroids after 2nd iteration:

- C1: (30, 95)

- C2: (65, 52.5)

- C3: (15, 35)

Calculating SSE for the second iteration

- Cluster 1:

- P1 to C1: sqrt((20-30)^2 + (100-95)^2) = sqrt(100 + 25) = sqrt(125) = 11.18^2 = 125

- P8 to C1: 125

- Total for Cluster 1: 125 + 125 = 250

- Cluster 2:

- P3 to C2: 381.25

- P4 to C2: 981.25

- P5 to C2: 31.25

- P6 to C2: 181.25

- Total for Cluster 2: 381.25 + 981.25 + 31.25 + 181.25 = 1575

- Cluster 3:

- P2 to C3: 250

- P7 to C3: 250

- Total for Cluster 3: 250 + 250 = 500

Total SSE after 2nd iteration: 250 (C1) + 1575 (C2) + 500 (C3) = 2325

1. **Between the first iteration and the second iteration, which one produces optimal clusters? Justify your answer. [10 points]**

Answer:   
First Iteration

- SSE: 3700

- Clusters:

- C1: P1

- C2: P3, P4, P5, P6, P8

- C3: P2, P7

Second Iteration

- SSE: 2325

- Clusters:

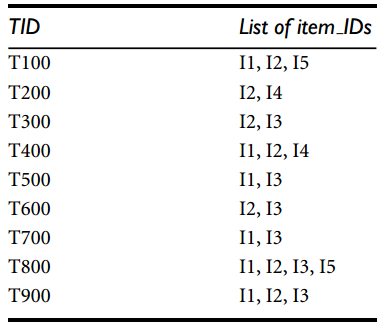
- Cluster 1: P1, P8

- Cluster 2: P3, P4, P5, P6

- Cluster 3: P2, P7

The second iteration produces more optimal clusters as it has a lower SSE, indicating better cohesion within clusters and better separation of data

**Problem-2: Market Basket Analysis [30 points]**

The following table shows transactional data for a company.

1. **What is the maximum number of association rules that can be extracted from this data (including rules that have zero support)? [5 points]**

Answer: total number of items: n=5

∑4​(k5​)(25−k−1)=(15​)(24−1)+(25​)(23−1)+(35​)(22−1)+(45​)(21−1)=5(15)+10(7)+10(3)+5(1)=75+70+30+5

=180

1. **What is the maximum number of frequent itemsets that can be extracted (assuming min support count >= 0)? [5 points]**

Answer: 2^n - 1 = 2^5 – 1 = 31

1. **Apply Apriori algorithm to generate the candidate itemsets and frequent itemsets for a set of k-itemsets where k = 1 to 3. Min support count ≥ 2. [10 points]**

Answer:   
**1-itemsets (count ≥ 2)**

Count of each item:

* I1: 6 (T100, T400, T500, T700, T800, T900)
* I2: 7 (T100, T200, T300, T400, T600, T800, T900)
* I3: 6 (T300, T500, T600, T700, T800, T900)
* I4: 2 (T200, T400)
* I5: 2 (T100, T800)

So, L1:

* L1 = {I1}, {I2}, {I3}, {I4}, {I5}

**Step 2: 2-itemsets**

Generate all pairs from L1:

Count support:

* {I1, I2} → 4 (T100, T400, T800, T900)
* {I1, I3} → 4 (T500, T700, T800, T900)
* {I1, I4} → 1 (T400)
* {I1, I5} → 2 (T100, T800)
* {I2, I3} → 4 (T300, T600, T800, T900)
* {I2, I4} → 2 (T200, T400)
* {I2, I5} → 2 (T100, T800)
* {I3, I4} → 0
* {I3, I5} → 1 (T800)
* {I4, I5} → 0

**So, (L2):**

* {I1, I2}, {I1, I3}, {I1, I5}, {I2, I3}, {I2, I4}, {I2, I5}

**3-itemsets (combine frequent 2-itemsets)**

Only combinations where all subsets are frequent:

Candidate 3-itemsets (C3):

* {I1, I2, I3} → appears in T800, T900 → support = 2
* {I1, I2, I5} → T100, T800 → support = 2

Other combinations don't meet support or have infrequent subsets.

**Frequent 3-itemsets (L3):**

* {I1, I2, I3}, {I1, I2, I5}

**Final Frequent Itemsets:**

L1 = {I1}, {I2}, {I3}, {I4}, {I5}

L2={I1,I2},{I1,I3},{I1,I5},{I2,I3},{I2,I4},{I2,I5}  
 L3 = {I1, I2, I3}, {I1, I2, I5}

1. **What are the association rules that can be generated from the set of the 3-itemsets calculated in the above question? Compute confidence of each rule. Given that *minconf* ≥ 70%, which rules will be considered for association mining. [10 points]**

Answer:

Rules from {I1, I2, I3} (support=2):

• - I1 ∧ I2 ⇒ I3 → Conf = 2/4 = 50%

• - I1 ∧ I3 ⇒ I2 → Conf = 2/4 = 50%

• - I2 ∧ I3 ⇒ I1 → Conf = 2/4 = 50%

• - I1 ⇒ I2 ∧ I3 → Conf = 2/6 = 33%

• - I2 ⇒ I1 ∧ I3 → Conf = 2/7 ≈ 28.6%

• - I3 ⇒ I1 ∧ I2 → Conf = 2/6 ≈ 33%

Rules from {I1, I2, I5} (support=2):

• - I1 ∧ I2 ⇒ I5 → Conf = 2/4 = 50%

• - I1 ∧ I5 ⇒ I2 → Conf = 2/2 = 100%

• - I2 ∧ I5 ⇒ I1 → Conf = 2/2 = 100%

• - I1 ⇒ I2 ∧ I5 → Conf = 2/6 = 33%

• - I2 ⇒ I1 ∧ I5 → Conf = 2/7 ≈ 28.6%

• - I5 ⇒ I1 ∧ I2 → Conf = 2/2 = 100%

**Valid Rules (Confidence ≥ 70%):**

**• - I1 ∧ I5 ⇒ I2 (100%)**

**• - I2 ∧ I5 ⇒ I1 (100%)**

**• - I5 ⇒ I1 ∧ I2 (100%)**

**Problem-3: R-Programming [40 points]**

A wholesale customer dataset is available in the homework folder. The dataset refers to clients of a wholesale distributor. It includes the annual spending in monetary units (m.u.) on diverse product categories. Your task is to apply k-means clustering to find some interesting groupings from the data. The attribute information is as per the following:

1. CHANNEL: Wholesale purchase or Retail purchase
2. REGION: Region1, Region2, or Region3
3. FRESH: annual spending (m.u.) on fresh products
4. MILK: annual spending (m.u.) on milk products
5. GROCERY: annual spending (m.u.)on grocery products
6. FROZEN: annual spending (m.u.)on frozen products
7. DETERGENTS\_PAPER: annual spending (m.u.) on detergents & paper products
8. DELICATESSEN: annual spending (m.u.)on and delicatessen products

Specifically, develop an R script to

1. Read-in the dataset. **[5 points]**
2. Apply k-means algorithm. **[15 points]**
3. Play with the k value to find some interesting groupings **[10 points]**.
4. Write a brief report to explain each of the findings **[10 points]**.

:Answer for 4 is on the code file.