Lab 3
732A75: Association Analysis -2

Aman Nayak (amana551)
Mahmood Siddique(mahsi404)

Objective:

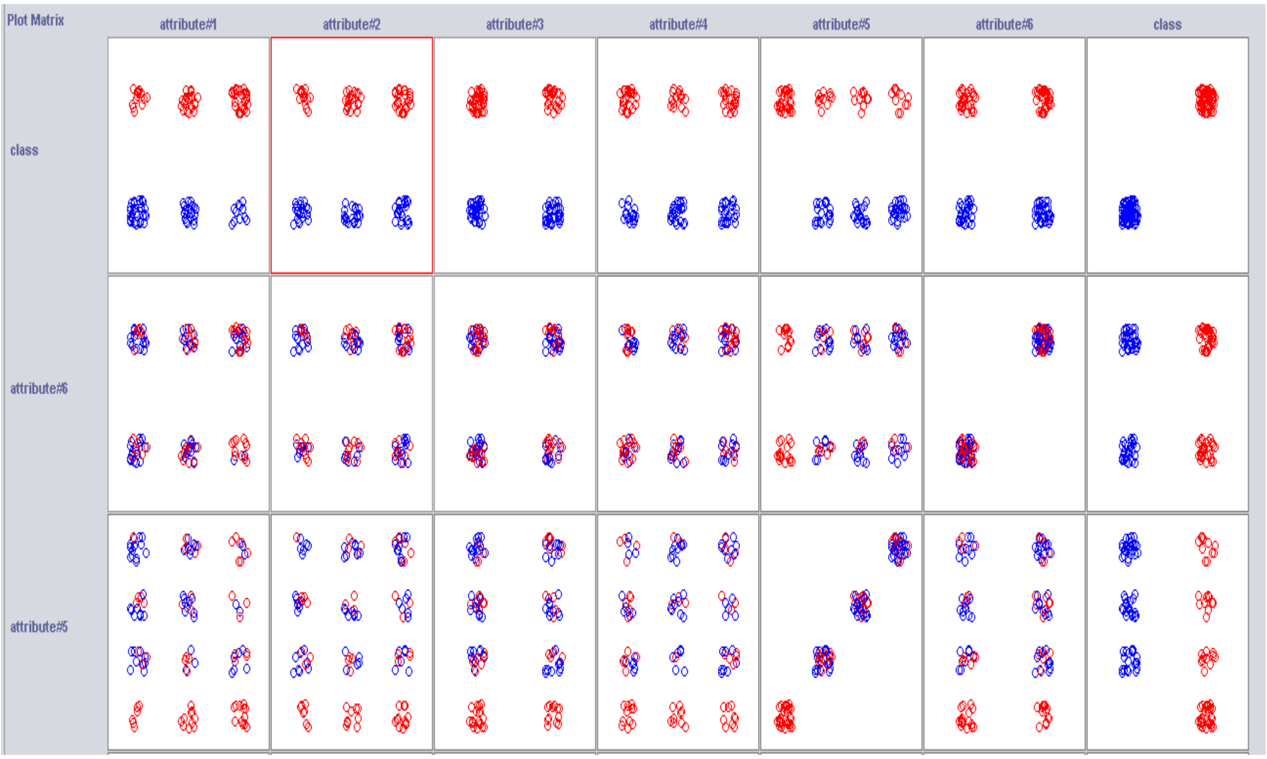
Appreciate the importance of the distance metric used within the clustering algorithm, e.g. the clusters found by the algorithm may not correspond to the clustering that is most natural for us human beings.

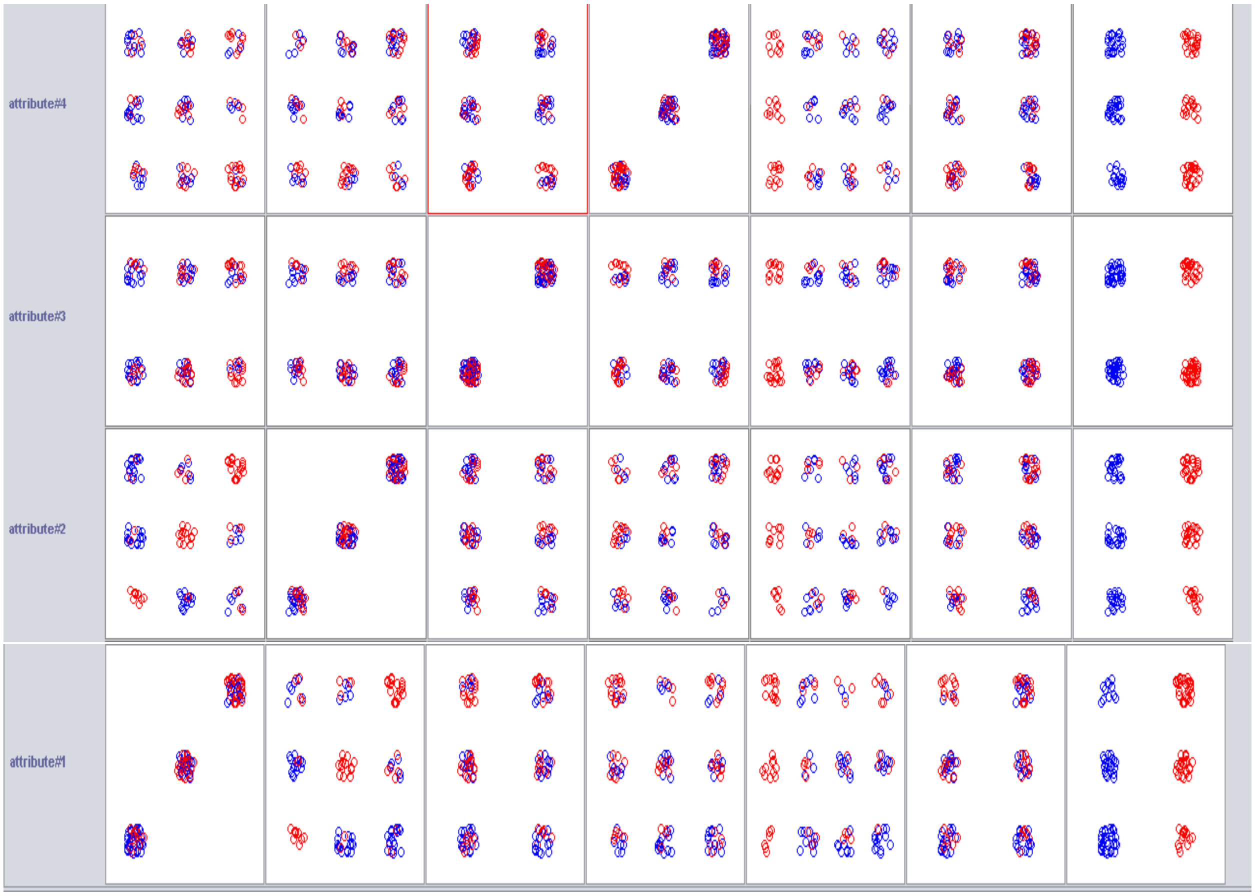
Data: We have monk1 dataset which is artificial dataset with 124 instances, each described by 6 discrete attributes and a binary class attribute.

Clustering Analysis

Clustering algorithm is well known unsupervised learning method for harnessing insights from unlabeled data and grouping data in similar group. We are analyzing data with below different clustering algorithms:

Visual representation of given data:





It can be seen from visual representation of data that attributes have strong correlation and are grouped for such data clustering is not a good option for analysis, we will run different clustering algorithms to validate our visual analysis:

Case 1: SimpleKMeans

1. **Config Setting:**

Clustering Algorithm	**SimpleKMeans**
Number of Cluster	**2**
Classes to cluster evaluation	**(Nom) class**
Seed	**10**

```
=== Run information ===

Scheme:       weka.clusterers.SimpleKMeans -init 0 -max-candidates 100 -periodic-pruning 10000 -min-
density 2.0 -t1 -1.25 -t2 -1.0 -N 2 -A "weka.core.EuclideanDistance -R first-last" -I 500 -num-slots 1 -S 10
Relation:     monk1
Instances:    124
Attributes:   7
              attribute#1
              attribute#2
              attribute#3
              attribute#4
              attribute#5
              attribute#6
Ignored:
              class
Test mode:    Classes to clusters evaluation on training data

=== Clustering model (full training set) ===


kMeans
======

Number of iterations: 3
Within cluster sum of squared errors: 358.0

Initial starting points (random):

Cluster 0: 1,2,1,1,1,2
Cluster 1: 3,2,2,3,2,1

Missing values globally replaced with mean/mode

Final cluster centroids:
                          Cluster#
Attribute      Full Data          0          1
                 (124.0)     (77.0)     (47.0)
==============================================
attribute#1            1          1          3
attribute#2            3          2          3
attribute#3            1          1          2
attribute#4            3          1          3
attribute#5            4          4          2
attribute#6            2          2          1




Time taken to build model (full training data) : 0 seconds

=== Model and evaluation on training set ===

Clustered Instances

0       77 ( 62%)
1       47 ( 38%)
```

```
Class attribute: class
Classes to Clusters:

  0  1  <-- assigned to cluster
 40 22 | 0
 37 25 | 1

Cluster 0 <-- 0
Cluster 1 <-- 1

Incorrectly clustered instances : 59.0       47.5806 %
```

2. Config Setting:

Clustering Algorithm	**SimpleKMeans**
Number of Cluster	**4**
Classes to cluster evaluation	**(Nom) class**
Seed	**10**

```
=== Run information ===

Scheme:       weka.clusterers.SimpleKMeans -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 4 -A "weka.core.EuclideanDistance -R first-last" -I 500 -num-slots 1 -S 10
Relation:     monk1
Instances:    124
Attributes:   7
              attribute#1
              attribute#2
              attribute#3
              attribute#4
              attribute#5
              attribute#6
Ignored:
              class
Test mode:    Classes to clusters evaluation on training data

=== Clustering model (full training set) ===


kMeans
======

Number of iterations: 3
Within cluster sum of squared errors: 293.0

Initial starting points (random):
```

```
Cluster 0: 1,2,1,1,1,2
Cluster 1: 3,2,2,3,2,1
Cluster 2: 2,3,1,2,1,1
Cluster 3: 2,3,1,3,1,2

Missing values globally replaced with mean/mode

Final cluster centroids:
                    Cluster#
Attribute     Full Data        0        1        2        3
            (124.0)    (50.0)    (36.0)    (24.0)    (14.0)
==================================================================
attribute#1         1        1        3        2        2
attribute#2         3        2        1        3        3
attribute#3         1        1        2        1        1
attribute#4         3        1        3        2        3
attribute#5         4        3        2        1        1
attribute#6         2        2        1        1        2




Time taken to build model (full training data) : 0 seconds

=== Model and evaluation on training set ===

Clustered Instances

0      50 ( 40%)
1      36 ( 29%)
2      24 ( 19%)
3      14 ( 11%)


Class attribute: class
Classes to Clusters:

  0  1  2  3  <-- assigned to cluster
 29 17 11  5 | 0
 21 19 13  9 | 1

Cluster 0 <-- 0
Cluster 1 <-- 1
Cluster 2 <-- No class
Cluster 3 <-- No class

Incorrectly clustered instances : 76.0      61.2903 %
```

3. Config Setting:

Clustering Algorithm	**SimpleKMeans**
Number of Cluster	**3**
Classes to cluster evaluation	**(Nom) class**
Seed	**10**

```
=== Run information ===

Scheme:       weka.clusterers.SimpleKMeans -init 0 -max-candidates 100 -periodic-pruning 10000 -min-
density 2.0 -t1 -1.25 -t2 -1.0 -N 3 -A "weka.core.EuclideanDistance -R first-last" -I 500 -num-slots 1 -S 10
Relation:     monk1
Instances:    124
Attributes:   7
              attribute#1
              attribute#2
              attribute#3
              attribute#4
              attribute#5
              attribute#6
Ignored:
              class
Test mode:    Classes to clusters evaluation on training data

=== Clustering model (full training set) ===


kMeans
======

Number of iterations: 3
Within cluster sum of squared errors: 314.0

Initial starting points (random):

Cluster 0: 1,2,1,1,1,2
Cluster 1: 3,2,2,3,2,1
Cluster 2: 2,3,1,2,1,1

Missing values globally replaced with mean/mode

Final cluster centroids:
                         Cluster#
Attribute     Full Data         0         1         2
                (124.0)    (59.0)    (38.0)    (27.0)
=====================================================
attribute#1           1         1         3         2
attribute#2           3         2         1         3
```

```
attribute#3          1         1         2         1
attribute#4          3         1         3         2
attribute#5          4         3         2         1
attribute#6          2         2         1         1



Time taken to build model (full training data) : 0 seconds

=== Model and evaluation on training set ===

Clustered Instances

0       59 ( 48%)
1       38 ( 31%)
2       27 ( 22%)


Class attribute: class
Classes to Clusters:

  0  1  2  <-- assigned to cluster
 33 17 12 | 0
 26 21 15 | 1

Cluster 0 <-- 0
Cluster 1 <-- 1
Cluster 2 <-- No class

Incorrectly clustered instances : 70.0       56.4516 %
```

Case 2: MakeDensityBasedClusterer

1. **Config Setting:**

Clustering Algorithm	**MakeDensityBasedClusterer**
Number of Cluster	**2**
Classes to cluster evaluation	**(Nom) class**
Seed	**10**
Standard Deviation	**1.0E-6**

```
=== Run information ===
Scheme:       weka.clusterers.MakeDensityBasedClusterer -M 1.0E-6 -W weka.clusterers.SimpleKMeans --
-init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 2 -A
"weka.core.EuclideanDistance -R first-last" -I 500 -num-slots 1 -S 10
Relation:     monk1
Instances:    124
Attributes:   7
              attribute#1
              attribute#2
              attribute#3
              attribute#4
              attribute#5
              attribute#6
Ignored:
              class
Test mode:    Classes to clusters evaluation on training data

=== Clustering model (full training set) ===

MakeDensityBasedClusterer:

Wrapped clusterer:
kMeans
======

Number of iterations: 3
Within cluster sum of squared errors: 358.0

Initial starting points (random):

Cluster 0: 1,2,1,1,1,2
Cluster 1: 3,2,2,3,2,1

Missing values globally replaced with mean/mode

Final cluster centroids:
                          Cluster#
Attribute      Full Data          0          1
                (124.0)     (77.0)     (47.0)
================================================
attribute#1            1          1          3
attribute#2            3          2          3
attribute#3            1          1          2
attribute#4            3          1          3
attribute#5            4          4          2
attribute#6            2          2          1
```

```
Fitted estimators (with ML estimates of variance):

Cluster: 0 Prior probability: 0.619

Attribute: attribute#1
Discrete Estimator. Counts =  35 30 15  (Total = 80)
Attribute: attribute#2
Discrete Estimator. Counts =  22 36 22  (Total = 80)
Attribute: attribute#3
Discrete Estimator. Counts =  53 26  (Total = 79)
Attribute: attribute#4
Discrete Estimator. Counts =  33 26 21  (Total = 80)
Attribute: attribute#5
Discrete Estimator. Counts =  18 15 20 28  (Total = 81)
Attribute: attribute#6
Discrete Estimator. Counts =  26 53  (Total = 79)

Cluster: 1 Prior probability: 0.381

Attribute: attribute#1
Discrete Estimator. Counts =  12 14 24  (Total = 50)
Attribute: attribute#2
Discrete Estimator. Counts =  15 8 27  (Total = 50)
Attribute: attribute#3
Discrete Estimator. Counts =  14 35  (Total = 49)
Attribute: attribute#4
Discrete Estimator. Counts =  11 15 24  (Total = 50)
Attribute: attribute#5
Discrete Estimator. Counts =  13 18 12 8  (Total = 51)
Attribute: attribute#6
Discrete Estimator. Counts =  32 17  (Total = 49)


Time taken to build model (full training data) : 0 seconds

=== Model and evaluation on training set ===

Clustered Instances

0      83 ( 67%)
1      41 ( 33%)


Log likelihood: -6.09856


Class attribute: class
```

```
Classes to Clusters:

  0  1  <-- assigned to cluster
 44 18 | 0
 39 23 | 1

Cluster 0 <-- 0
Cluster 1 <-- 1

Incorrectly clustered instances : 57.0       45.9677 %
```

2. Config Setting:

Clustering Algorithm	**MakeDensityBasedClusterer**
Number of Cluster	**3**
Classes to cluster evaluation	**(Nom) class**
Seed	**10**
Standard Deviation	**1.0E-6**

```
=== Run information ===

Scheme:       weka.clusterers.MakeDensityBasedClusterer -M 1.0E-6 -W weka.clusterers.SimpleKMeans --
-init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 3 -A
"weka.core.EuclideanDistance -R first-last" -I 500 -num-slots 1 -S 10
Relation:     monk1
Instances:    124
Attributes:   7
              attribute#1
              attribute#2
              attribute#3
              attribute#4
              attribute#5
              attribute#6
Ignored:
              class
Test mode:    Classes to clusters evaluation on training data

=== Clustering model (full training set) ===

MakeDensityBasedClusterer:

Wrapped clusterer:
kMeans
======

Number of iterations: 3
Within cluster sum of squared errors: 314.0

Initial starting points (random):
```

```
Cluster 0: 1,2,1,1,1,2
Cluster 1: 3,2,2,3,2,1
Cluster 2: 2,3,1,2,1,1

Missing values globally replaced with mean/mode

Final cluster centroids:
                      Cluster#
Attribute     Full Data         0          1          2
              (124.0)     (59.0)     (38.0)     (27.0)
===========================================================
attribute#1          1          1          3          2
attribute#2          3          2          1          3
attribute#3          1          1          2          1
attribute#4          3          1          3          2
attribute#5          4          3          2          1
attribute#6          2          2          1          1



Fitted estimators (with ML estimates of variance):

Cluster: 0 Prior probability: 0.4724

Attribute: attribute#1
Discrete Estimator. Counts =  32 17 13  (Total = 62)
Attribute: attribute#2
Discrete Estimator. Counts =  9 34 19  (Total = 62)
Attribute: attribute#3
Discrete Estimator. Counts =  37 24  (Total = 61)
Attribute: attribute#4
Discrete Estimator. Counts =  30 14 18  (Total = 62)
Attribute: attribute#5
Discrete Estimator. Counts =  8 13 23 19  (Total = 63)
Attribute: attribute#6
Discrete Estimator. Counts =  16 45  (Total = 61)

Cluster: 1 Prior probability: 0.3071

Attribute: attribute#1
Discrete Estimator. Counts =  9 10 22  (Total = 41)
Attribute: attribute#2
Discrete Estimator. Counts =  21 7 13  (Total = 41)
Attribute: attribute#3
Discrete Estimator. Counts =  9 31  (Total = 40)
Attribute: attribute#4
Discrete Estimator. Counts =  11 9 21  (Total = 41)
```

```
Attribute: attribute#5
Discrete Estimator. Counts =  9 17 6 10  (Total = 42)
Attribute: attribute#6
Discrete Estimator. Counts =  24 16  (Total = 40)

Cluster: 2 Prior probability: 0.2205

Attribute: attribute#1
Discrete Estimator. Counts =  7 18 5  (Total = 30)
Attribute: attribute#2
Discrete Estimator. Counts =  8 4 18  (Total = 30)
Attribute: attribute#3
Discrete Estimator. Counts =  22 7  (Total = 29)
Attribute: attribute#4
Discrete Estimator. Counts =  4 19 7  (Total = 30)
Attribute: attribute#5
Discrete Estimator. Counts =  15 4 4 8  (Total = 31)
Attribute: attribute#6
Discrete Estimator. Counts =  19 10  (Total = 29)


Time taken to build model (full training data) : 0 seconds

=== Model and evaluation on training set ===

Clustered Instances

0       60 ( 48%)
1       39 ( 31%)
2       25 ( 20%)


Log likelihood: -6.09108


Class attribute: class
Classes to Clusters:

  0  1  2  <-- assigned to cluster
 35 16 11 | 0
 25 23 14 | 1

Cluster 0 <-- 0
Cluster 1 <-- 1
Cluster 2 <-- No class

Incorrectly clustered instances : 66.0       53.2258 %
```

3. **Config Setting:**

Clustering Algorithm	**MakeDensityBasedClusterer**
Number of Cluster	**4**
Classes to cluster evaluation	**(Nom) class**
Seed	**10**
Standard Deviation	**1.0E-6**

```
=== Run information ===

Scheme:       weka.clusterers.MakeDensityBasedClusterer -M 1.0E-6 -W weka.clusterers.SimpleKMeans --
-init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 4 -A
"weka.core.EuclideanDistance -R first-last" -I 500 -num-slots 1 -S 10
Relation:     monk1
Instances:    124
Attributes:   7
              attribute#1
              attribute#2
              attribute#3
              attribute#4
              attribute#5
              attribute#6
Ignored:
              class
Test mode:    Classes to clusters evaluation on training data

=== Clustering model (full training set) ===

MakeDensityBasedClusterer:

Wrapped clusterer:
kMeans
======

Number of iterations: 3
Within cluster sum of squared errors: 293.0

Initial starting points (random):

Cluster 0: 1,2,1,1,1,2
Cluster 1: 3,2,2,3,2,1
Cluster 2: 2,3,1,2,1,1
Cluster 3: 2,3,1,3,1,2

Missing values globally replaced with mean/mode

Final cluster centroids:
                          Cluster#
Attribute     Full Data        0         1         2         3
```

```
               (124.0)   (50.0)   (36.0)   (24.0)   (14.0)
==================================================================
attribute#1          1        1        3        2        2
attribute#2          3        2        1        3        3
attribute#3          1        1        2        1        1
attribute#4          3        1        3        2        3
attribute#5          4        3        2        1        1
attribute#6          2        2        1        1        2



Fitted estimators (with ML estimates of variance):

Cluster: 0 Prior probability: 0.3984

Attribute: attribute#1
Discrete Estimator. Counts =  28 13 12  (Total = 53)
Attribute: attribute#2
Discrete Estimator. Counts =  9 31 13  (Total = 53)
Attribute: attribute#3
Discrete Estimator. Counts =  30 22  (Total = 52)
Attribute: attribute#4
Discrete Estimator. Counts =  30 14 9  (Total = 53)
Attribute: attribute#5
Discrete Estimator. Counts =  6 11 22 15  (Total = 54)
Attribute: attribute#6
Discrete Estimator. Counts =  16 36  (Total = 52)

Cluster: 1 Prior probability: 0.2891

Attribute: attribute#1
Discrete Estimator. Counts =  9 10 20  (Total = 39)
Attribute: attribute#2
Discrete Estimator. Counts =  21 7 11  (Total = 39)
Attribute: attribute#3
Discrete Estimator. Counts =  8 30  (Total = 38)
Attribute: attribute#4
Discrete Estimator. Counts =  11 9 19  (Total = 39)
Attribute: attribute#5
Discrete Estimator. Counts =  8 16 6 10  (Total = 40)
Attribute: attribute#6
Discrete Estimator. Counts =  24 14  (Total = 38)

Cluster: 2 Prior probability: 0.1953

Attribute: attribute#1
Discrete Estimator. Counts =  7 16 4  (Total = 27)
Attribute: attribute#2
```

```
Discrete Estimator. Counts =  8 4 15  (Total = 27)
Attribute: attribute#3
Discrete Estimator. Counts =  19 7  (Total = 26)
Attribute: attribute#4
Discrete Estimator. Counts =  4 19 4  (Total = 27)
Attribute: attribute#5
Discrete Estimator. Counts =  13 4 4 7  (Total = 28)
Attribute: attribute#6
Discrete Estimator. Counts =  19 7  (Total = 26)

Cluster: 3 Prior probability: 0.1172

Attribute: attribute#1
Discrete Estimator. Counts =  5 7 5  (Total = 17)
Attribute: attribute#2
Discrete Estimator. Counts =  1 4 12  (Total = 17)
Attribute: attribute#3
Discrete Estimator. Counts =  12 4  (Total = 16)
Attribute: attribute#4
Discrete Estimator. Counts =  1 1 15  (Total = 17)
Attribute: attribute#5
Discrete Estimator. Counts =  6 4 2 6  (Total = 18)
Attribute: attribute#6
Discrete Estimator. Counts =  1 15  (Total = 16)


Time taken to build model (full training data) : 0 seconds

=== Model and evaluation on training set ===

Clustered Instances

0       51 ( 41%)
1       35 ( 28%)
2       23 ( 19%)
3       15 ( 12%)


Log likelihood: -6.06035


Class attribute: class
Classes to Clusters:

  0  1  2  3  <-- assigned to cluster
 28 17 12  5 | 0
 23 18 11 10 | 1
```

```
Cluster 0 <-- 0
Cluster 1 <-- 1
Cluster 2 <-- No class
Cluster 3 <-- No class

Incorrectly clustered instances : 78.0      62.9032 %
```

<u>Analysis of Clustering Algorithm performance**</u>**

With different numbers of clusters and different clustering algorithms, the misclassification rate (which is 1- Accuracy rate) is high. Thus, our first visual analysis was correct that the clustering algorithm will not perform optimally here as we have a high correlation in data among different attributes, which makes it difficult to segregate data into different classes based on Euclidean or density. Also, selected algorithms work well with continuous data while we were provided with discrete data here.

Table of Misclassification:

Clustering Algorithm	Number of Clusters	Misclassification Rate
SimpleKMeans	2	47.5806 %
SimpleKMeans	3	56.4516 %
SimpleKMeans	4	61.2903 %
MakeDensityBasedClusterer	2	45.9677 %
MakeDensityBasedClusterer	3	53.2258 %
MakeDensityBasedClusterer	4	62.9032 %

Association Analysis

In this part of analysis, we are going to use Apriori Algorithm along with SimpleKMeans clustering for analysis.

Config Setting:

Association Algorithm	**Apriori**
Number of Cluster	**2**
Classes to cluster evaluation	**(Nom) class**
Seed	**10**
Minimum Support	**0.05**

```
=== Run information ===

Scheme:     weka.associations.Apriori -N 19 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.05 -S -1.0 -c -1
Relation:    monk1-weka.filters.unsupervised.attribute.AddCluster-Wweka.clusterers.SimpleKMeans -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 2 -A "weka.core.EuclideanDistance -R first-last" -I 500 -num-slots 1 -S 10-weka.filters.unsupervised.attribute.Remove-R8
Instances:    124
Attributes:   7
```

```
            attribute#1
            attribute#2
            attribute#3
            attribute#4
            attribute#5
            attribute#6
            class
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.05 (6 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 19

Generated sets of large itemsets:

Size of set of large itemsets L(1): 19

Size of set of large itemsets L(2): 151

Size of set of large itemsets L(3): 378

Size of set of large itemsets L(4): 125

Size of set of large itemsets L(5): 6

Best rules found:

 1. attribute#5=1 29 ==> class=1 29    <conf:(1)> lift:(2) lev:(0.12) [14] conv:(14.5)
 2. attribute#1=3 attribute#2=3 17 ==> class=1 17    <conf:(1)> lift:(2) lev:(0.07) [8] conv:(8.5)
 3. attribute#3=1 attribute#5=1 17 ==> class=1 17    <conf:(1)> lift:(2) lev:(0.07) [8] conv:(8.5)
 4. attribute#5=1 attribute#6=1 16 ==> class=1 16    <conf:(1)> lift:(2) lev:(0.06) [8] conv:(8)
 5. attribute#1=2 attribute#2=2 15 ==> class=1 15    <conf:(1)> lift:(2) lev:(0.06) [7] conv:(7.5)
 6. attribute#1=3 attribute#5=1 13 ==> class=1 13    <conf:(1)> lift:(2) lev:(0.05) [6] conv:(6.5)
 7. attribute#5=1 attribute#6=2 13 ==> class=1 13    <conf:(1)> lift:(2) lev:(0.05) [6] conv:(6.5)
 8. attribute#2=3 attribute#5=1 12 ==> class=1 12    <conf:(1)> lift:(2) lev:(0.05) [6] conv:(6)
 9. attribute#3=2 attribute#5=1 12 ==> class=1 12    <conf:(1)> lift:(2) lev:(0.05) [6] conv:(6)
10. attribute#1=3 attribute#2=3 attribute#6=2 12 ==> class=1 12    <conf:(1)> lift:(2) lev:(0.05) [6] conv:(6)
11. attribute#4=1 attribute#5=1 11 ==> class=1 11    <conf:(1)> lift:(2) lev:(0.04) [5] conv:(5.5)
12. attribute#1=2 attribute#5=1 10 ==> class=1 10    <conf:(1)> lift:(2) lev:(0.04) [5] conv:(5)
13. attribute#2=2 attribute#5=1 10 ==> class=1 10    <conf:(1)> lift:(2) lev:(0.04) [5] conv:(5)
14. attribute#1=1 attribute#2=1 9 ==> class=1 9    <conf:(1)> lift:(2) lev:(0.04) [4] conv:(4.5)
15. attribute#4=2 attribute#5=1 9 ==> class=1 9    <conf:(1)> lift:(2) lev:(0.04) [4] conv:(4.5)
16. attribute#4=3 attribute#5=1 9 ==> class=1 9    <conf:(1)> lift:(2) lev:(0.04) [4] conv:(4.5)
17. attribute#1=2 attribute#2=2 attribute#3=1 9 ==> class=1 9    <conf:(1)> lift:(2) lev:(0.04) [4] conv:(4.5)
```

```
18. attribute#1=3 attribute#2=3 attribute#3=1 9 ==> class=1 9    <conf:(1)> lift:(2) lev:(0.04) [4] conv:(4.5)
19. attribute#3=1 attribute#5=1 attribute#6=1 9 ==> class=1 9    <conf:(1)> lift:(2) lev:(0.04) [4] conv:(4.5)
```

Filtering obtained best rules against redundant one:

```
attribute#5=1 29 ==> class=1 29                       <conf:(1)> lift:(2) lev:(0.12) [14] conv:(14.5).
attribute#1=3 attribute#2=3 17 ==> class=1 17    <conf:(1)> lift:(2) lev:(0.07) [8] conv:(8.5)     .
attribute#5=1 attribute#6=1 16 ==> class=1 16    <conf:(1)> lift:(2) lev:(0.06) [8] conv:(8          .
attribute#4=2 attribute#5=1 9 ==> class=1 9         <conf:(1)> lift:(2) lev:(0.04) [4] conv:(4.5)     .
```

Conclusion:

Looking at the above optimal result, we can say that attributes are closely related; thus, clustering does not give the desired outcome and therefore using Apriori Algorithm can be useful for such data set analysis.