Data and Web Science Group
Prof. Dr. Christian Bizer
B6 – B1.15
68159 Mannheim

Data Mining – FSS 2021

Exercise 2: Cluster Analysis

2.1. Analyzing the Customer Data Set

1. Import the *customers* data set into RapidMiner. The customer data set is provided in ILIAS as an Excel file.
2. **Cluster the dataset using K-Means clustering. Experiment with different K values. Which values do make sense?**
   Solution:

   - Cluster on the attributes ItemsBought and ItemsReturned
   - Visualize (Scatter) the clustering using ItemsBought and ItemsReturned for the axes and the attribute cluster for the color dimension.

   Conclusion:

   - K=3 makes sense as there are 3 distinct groups:
     - Customers that buy a lot and keep the products (Best customers)
     - Customers that buy a lot but also return some products (OK customers)
     - Customers that don’t buy a lot and also return most products (Likely bad customers)

   What does the clustering tell you concerning your product portfolio?

   Solution:
   - Join the clustering again with the original data using a join operator (inner join over id attribute).
   - View data and order by cluster

   Conclusion:

   - You should stop selling product 2435, as most customers return this product and don't keep it (likely because of bad product quality).

   What does the clustering tell you concerning your marketing efforts in different regions?

   Solution:

   - View data and order by cluster

Conclusion:

- All customers from the Best Customers cluster live in Zip-Code region 1 and 2.

3. **Cluster the data set using Agglomerative Hierarchical Clustering. What does the dendrogram tell you concerning your customer groups?**
   Conclusion:

   - The diagram shows that the "likely bad customers" have a high distance to the other clusters ("ok customers" and the "good customers"). This is shown by the large distance (y-scale) before those both cluster are joint.

4. **Flatten the hierarchical clustering so that you get 3 or 4 customer groups. Use the *MixedMeasures* with *MixedEuclideanDistance*. Name these groups with appropriate labels.**
   Solution:

   - View data and order by cluster

   Conclusion:

   - 3 Groups: Similar the K-Means with K=3.
   - 4 Groups: The cluster of the good customers is divided when using 4 clusters
     - Customers buying a lot and keeping (almost) everything (Best Customers)
     - Customers buying a lot and keeping most of the products (Good Customers)
     - Customers buying a lot but returning a significant number of products (Average Customers)
     - Customers returning almost all products (likely bad Customers)

2.2. Analyzing the Students Data Set

1. **Aggregate the students' data set by student and calculate the average mark and the average number of attended classes.**
   Solution:

   - Use XML importer or retrieve data from repository
   - Aggregate Data by mark and classes, set group by name

2. **Cluster the data set using the K-Means algorithm.**
   Does one attribute dominate the clustering? What can you do about this? Assign suitable labels to your clusters.
   Solution:

   - View data and order by cluster
   - Have a look at the value of the average attended classes

   Conclusion:

   - Clustering is dominated by the number of average attended classes
   - Avoid by normalizing this value to the data range of marks [1..5] (or normalize both between 0..1).
   - There are 3 different groups of students (K=3):

- Students attending often and getting good marks (good students)
- Students attending almost never and getting bad marks (likely bad students)
- Students attending seldom and getting good marks (high-flyer)

3. **Cluster the data set using Agglomerative Hierarchical Clustering. Experiment with different setting for calculating the cluster similarity. What is a good setting?**
4. **What does the dendrogram tell you about the distances between the different groups of students?**
   Solution 3 & 4:

   - Single link works best for this set of data. The outstanding "high-flyer" cluster is joined on top level with the other clusters.
   - Note: When trying to use 4 clusters SingleLink and AverageLink Split up "bad students" into different clusters. CompleteLink splits up the good Students into "very good" and "average students" which could be a better clustering for this set of data.

2.3. Clustering the Iris Data Set

1. **Cluster the Iris data set using different algorithms and parameter settings.**
2. **Does it make sense to normalize the data before applying the algorithms?**
   Conclusion: Although the ranges of the four parameters do not differ that much (-2.41 up to 3.104) a normalization makes sense.

3. **Try to choose an algorithm and parameter setting the reproduces the original division into the three different species.**
   Conclusion: Iris-Setosa is almost with any algorithm easy to extract as own cluster. Iris-virginica and Iris-versicolor have overlapping parameter ranges. A good approximation is using K=3 // NumericalMeasures - CamberraDistance. Here only 9 are wrong assigned.

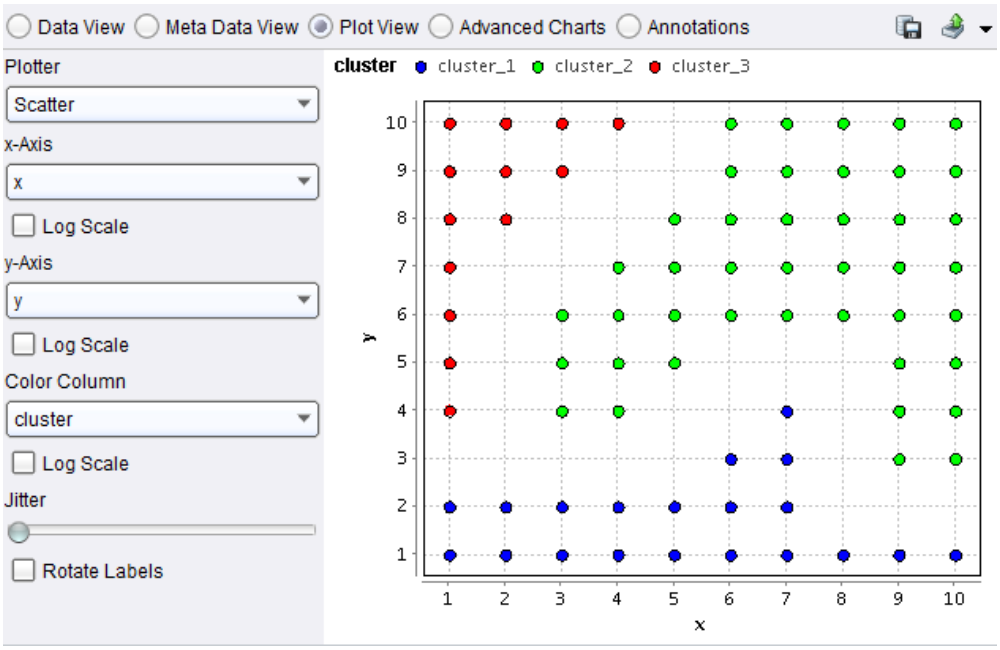
2.4. Clustering the Geo Data Set

1. **Within the geo data set (provided in ILIAS) the coordinates (x & y) of housings of inhabitants of an area are collected. Have a look at the data and visualize it using the Plot View**
   Solution: See first Exercise about plotting.

2. **Cluster the data using k-Means (k=3). Do the clusters represent the original areas?**
   Solution: Although its clearly recognizable that there are 3 different areas described by the geo data set, k-Means is not able to reproduce this areas as the areas have no representative centroid (are not round shaped) which leads to miss-clustered items.

3. **Apply DBSCAN and play around with the epsilon. Can you reproduce the original areas using this cluster algorithm?**
   Solution: Applying DBScan to the data-set does not work using the default set-up (epsilon = 1.0), as the data set only has integer coordinates (minimum distance 1).

   Conclusion: Increasing the epsilon to 2.0 and reduce minimum points to 3 lets DBScan reproduce the original areas.



2.5. Clustering the Zoo Data Set

1. **The zoo data set describes 101 animals using 18 different attributes. The data set is provided in ILIAS as an ARFF file. Import the Zoo data set into your local repository using the operators "Read ARFF" and "Store".**
2. **Cluster the data set using Agglomerative Hierarchical Clustering. Experiment with different parameter settings in order to generate a nice species tree.**
   Solution: Use Agglomerative Clustering with SingleLink and NominalMeasures with NominalDistance. Filter out the attributes: eggs, legs, predator and aquatic.

   Conclusion: It is possible to create a good hierarchy on top level. Going deeper species get mixed up within their clusters e.g. reptiles and amphibians.

3. **Try to assign appropriate species names to the clusters at the upper levels.**
   Conclusion: Clusters on top levels:

   - Cluster 195: Mammals
   - Cluster 199: Non-Mammals
   - Cluster 191: Fish
   - Cluster 186: Birds
   - Cluster 196: Reptiles & Amphibians
   - Cluster 184: Invertebrate
   - Cluster 178: Insects

   Possible cluster names on lower levels:

   - Cluster 195: Mammals
     - Cluster 165: Tail-less Mammals
     - Cluster 194: Mammals with tail
       - Cluster 189 Domestic Mammals
       - Cluster 193:Non-Domestic Mammals
         - Cluster 180: Mammals with fins
   - Cluster 199: Non-Mammals

- Cluster 191: Fish
    - Cluster 7: Domestic Fish
    - Cluster 163: Large Non-Domestic Fish
    - Cluster 181: Small Non-Domestic Fish
- Cluster 186: Birds
    - Cluster 181: Small Birds
    - Cluster 163: Large Birds
- Cluster 196: Reptiles & Amphibians
- Cluster 184: Invertebrate
- Cluster 178: Insects
    - Cluster 143: Non-Sliders
    - Cluster 174: Sliders

NOTE: Please note that this is only a possible solution for the task. There may be more hierarchical clustering solutions which may make sense.