Uppsala University
Department of Information Technology
Kjell Orsborn

DATA MINING I - 1DL360

Assignment 2 - k-Means and DBSCAN clustering

1 k-Means and DBSCAN clustering

In this assignment we focus on two different clustering techniques, namely k-Means clustering and DBSCAN.

The k-Means clustering method is a partitional clustering method. It is one of the most commonly used clustering methods as it is quite easy to understand and implement. It applies a proximity measure for forming the clustering.

DBSCAN [1] is a density-based clustering method that applies a density measure for forming the clustering.

In this assignment you will study both algorithms and examine their characteristics on two different 4-dimensional data sets. In this case, the data sets are completely synthetically generated. The reason for this is that they include specific hidden characteristics to be revealed by you in the assignment.

You will use an implementation of k-Means and DBSCAN that is implemented in the AmosMiner (AmoxMiner for Mac OS and Linux) application using the Amos II database management system. The instructions to install Amos II and AmosMiner are provided on the assignments home page. It can be used either in the computer labs at Uppsala University (Windows) or on your own laptop or desktop computer (Windows, Mac OS and Linux supported).

2 Preparation

We suggest that you read about k-Means and DBSCAN in Chapter 7 in Tan et al. [Tan06]. As for Assignment 1, it also useful to study the sections on normalization and proximity measures that are treated in Chapter 2 in Tan et al.

You can find and download the AmosMiner and Amos II system from the assignments home page and you will also get a chance to familiarize yourself with those systems in the introductory tutorials that you are recommended to attend. There is also one introductory tutorial to this assignment, which is not mandatory, but advisable to attend.

We also strongly recommend you to run the tutorials for Amos II and AmosMiner before you start with the actual assignment.

3 Assignment

You will find the following two data files for this assignment in the data directory under AmosMiner:

`clusterdata1.nt` including 150 rows that each include 4 fields.

`clusterdata2.nt` including 200 rows that each include 4 fields.

For both files this means that each row represents a data object through 4 attribute values, as a vector of values: {`attribute 1, attribute 2, attribute 3, attribute 4`}.

You should perform cluster analysis on both these data sets using the k-Means and DBSCAN clustering algorithms implemented within the AmosMiner system. Furthermore, you should experiment with the different parameters used in those algorithms to see how they influence the clusterings. The different steps in the assignment are described in more detail below. There is also a result form (available on the assignments home page) to be filled in by you and handed in to your assistant. Also

see Section 4 on more details on examination and how you should report the assignment.

Under the section for Assignment 2 on the assignment course page, you will find the script file for the this assignment named 'a2.osql'. This file includes queries and calls to predefined functions for importing data, preprocessing and normalization of data, and for performing and tuning k-Means and DBSCAN clustering.

You should explore the various phases of k-Means and DBSCAN clustering analysis by interactively exploring and modifying the query language statements found in the script file. This is done most easily by opening the script file in a text editor (e.g. Notepad in Windows and TextEdit in Mac OS), copy a query expression from the editor and paste it into the AmosMiner terminal window were it will be executed.

That is, you should do the following:

START-UP

Once you have started the AmosMiner system (follow the instructions on the assignments home page) and opened your `a2.osql` script file, you can start exploring the various aspects of k-Means and DBSCAN clustering by the copy-and-paste approach described above.

1. DATA DEFINITIONS AND IMPORT

   - Define the stored functions
   - Do steps 2-3 for each of the datasets: `clusterdata1.nt` and `clusterdata2.nt`

2. K-MEANS CLUSTERING

   (a) Preliminary Analysis of Number of Clusters
   Determine the number of clusters for k-Means to discover. You should experiment with different numbers of clusters to be identified with k-Means.

   Visualize data the 4-dimensional dataset using different projections such as:

- projecting original data to different 2- and 3-dimensional subspaces
- projecting to eigenvectors (Principal Component Analysis), as shown in the `a2.osql` script file.

(b) Choosing Initial Centroids

Generate k initial centroids as for example by selecting a random sample of the dataset.

(c) Single k-Means Analysis

Obtain the set of final centroids. Assign each point to a cluster defined by a final centroid, and evaluate:

- graphically, using different projections, and by
- computing sum of squared error (SSE).

(d) Data Normalization

Try different normalization methods. You can experiment with different normalization methods as described in Assignment 1.

Note that the samples (i.e. the initial centroids) and the final centroids will also be generated in a normalized scale. Remember to transform the results (an assignment of data points to the clusters) in the original scale, in order to visualize and compare SSE.

(e) Multiple k-Means Analysis

Generate a number of different sets of initial centroids. Use a stored function to store them.

Now run k-Means for each of them and generate the corresponding sets of final centroids.

Choose the best set of initial centroids based on SSE for the resulting clustering. Save this clustering and visualize the clusters.

3. DBSCAN CLUSTERING

   (a) DBSCAN Analysis

   - Define the Eps and MinPts parameters.

   - Build the directly-density-reachable (DDR) table and the set of core points. Count the number of core points discovered.

   - Run DBSCAN with the DDR table and core points.

   - Visualize the result (use different projections), and compute the SSE.

   (b) Data Normalization

   - Try different normalization methods to see how its influencing the DBSCAN algorithm.

   (c) Visually Selecting Eps and MinPts

   - Define a selector function that selects a k-th smallest number from a set of numbers.

   - Plot a k-Nearest distance curve for given number k. Try different k. Select the optimal Eps and MinPts parameter based on that.

   - Repeat Step 3a with the optimal parameters.

4. COMPARING THE ALGORITHMS

   (a) Which algorithm have you found to be most suitable for clustering of each dataset?

   (b) Which parameters (including initial centroids) did you use?

You should motivate your answers and explain your decisions and reasoning.

You can also find these instructions and hints in the TODO list of the `a2.osql` script file.

You are also referred to the Amos II manual for further reading, which is available through the lab course home page. We suggest that you read section 2.6 about Collections. You might find the following sections especially useful: `avg`, `stdev`, and `count` in 2.6.1, `groupby` in 2.6.2, and `aggv`.

4 Examination

At the examination, you might be asked to execute your implementations in AmosMiner by running your script file.

We further expect you to present the outcome of your analysis when it comes to choice of different parameter values including:

- number of clusters, i.e. parameter `k` (KMEANS).
- set of initial centroids saved, for instance, to the files `vic1.nt` and `vic2.nt` (KMEANS).
- `eps` parameter (DBSCAN).
- `minpts` parameter (DBSCAN).

Be prepared to answer questions regarding topics such as:

- A brief description of how the data was preprocessed and why.
- Which normalization, if any, results in better clustering (determine this visually, you cannot compare SSE values)?
- How did you choose your `k` parameter in KMEANS?
- How will the selection of centroids influence the solution in KMEANS?
- How did you choose your `eps` and `minpts` parameters in (DBSCAN) and how might this choice influence your results?
- If it is known a priori (beforehand) that each attribute has different importance, how can that knowledge be used in the pre-processing to improve the classication performance?

References

[EK96] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, pages 226-231, 1996 (The paper is available on the lab course homepage).

[Tan06] Tan, P-N, Steinbach, M. and Kumar, V.: Introduction to Data Mining, Addison-Wesley, 2006.