Dr. Eick

COSC 3337

Problem Set2 Fall 2023

Clustering, Similarity Assessment and Outlier Detection

Deadlines; Task 3 will be due on November 6, and Task 4 will be due on November 20.

Last updated: October 23, 2023 at 11a

Responsible TA: Janet

Task 3: Clustering with K-Means and DBSCAN

Individual Task

Third Draft

A graph of different colored dots

Description automatically generated

Fig. 1: Complex8 DBSCAN Clustering Result

Weight: 55 points

**Learning Objectives**:

1. Learn to use popular clustering algorithms, namely K-means and DBSCAN
2. Learn how to summarize and interpret clustering results
3. Learn to write analysis and evaluation functions which operate on the top of clustering algorithms and clustering results
4. Learning how to interpret unsupervised data mining results

**Datasets**: In Task you will be using the Complex8 dataset which is a 2D spatial dataset that can be found at <http://www2.cs.uh.edu/~ceick/DM/Complex8.csv> and the Basel Weather Dataset (BWD for short, which you can be found at: <https://www2.cs.uh.edu/~ceick/white/Basel_Weather.csv> ); the last attribute of each dataset denotes a class variable which should be ignored when clustering the data sets—however, the class variable will be used in the post analysis of the clusters generated by running K-means and DBSCAN. Moreover, additionally ignore the DATE and MONTH attributes when clustering BWD.

The Basel Weather Dataset has the the following attributes:

DATE / nominal / Each record has a date starting from 01/01/2000 to 01/01/2010

MONTH / nominal / -- / 1 to 12

cloud\_cover / nominal / %/ 0 to 10, The fraction or percentage of the sky that is covered by clouds at a specific location and time

global\_rediation / continuous / W/m2/ The solar radiation that reaches the Earth's surface

precipitation / continuous / mm / Amount of rainfall

subshine / continuous / hours / The amount of time during which the sun is visible and unobscured by clouds or other atmospheric obstructions.

temp\_mean / continuous / celcicus / Average temperture of the day

temp\_min / continuous / celcicus / Minimum temperture of the day

temp\_max / continuous / celcicus / Maximum temperture of the day

humidity / continuous / % /

humidity\_class / ordinal/ ---/ Low, Mid, and High (derived from Humidity attribute; see below)

3 Examples in the Weather Prediction Dataset:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 20000101 | 1 | 7 | 0.82 | 1.34 | 0 | -15.2 | -17 | -13.4 | 0.89 | Mid |
| 20000102 | 1 | 5 | 0.6 | 0.39 | 2.8 | -13.7 | -15 | -12.3 | 0.86 | Low |
| 20000110 | 1 | 8 | 0.25 | 0.38 | 0 | -13.3 | -15 | -11.6 | 0.97 | High |

**Task 3 Subtasks:**

1. Write an function[[1]](#footnote-1) purity(a,b,outliers=FALSE) that computes the purity of a clustering result based on an apriori given set of class lables, where *a* gives the assignment of objects in O to clusters, and *b* is the “ground truth”. Purity is defined as follows: Let

O be a dataset

X={C1,…,Ck} be a clustering of O with Ci ⊆O (for i=1,…,k), C1∪…∪Ck ⊆O and Ci∩Cj=∅ (for i≠ j)

PUR(X)= (number\_of\_majority\_class\_examples(X)/(total\_number\_examples\_in\_clusters(X))

If the used clustering algorithm supports outliers, outliers should be ignored in purity computations; if you use R-clustering algorithms, you can assume that cluster 0 contains all the outliers, and clusters 1,2,…,k represent “true” clusters. If the parameter outliers is set to FALSE, the function just returns a floting point number of the observed purity, if parameter outliers is set to T the function returns a vector: (<purity>,<percentage\_of\_outliers); e.g. if the function returns (0.98, 0.2) this would indicate that the purity is 98%, but 20% of the objects in dataset O have been classified as outliers.\*

1. Run K-means for k=8 and k=12 twice for the Complex8 dataset. Also compute their purity using the function you developed in Task a. Visualize and compare the obtained four clusterings! \*\*\*
2. Develop a search procedure that looks for the “best” clustering by exploring different settings for the (MinPoints, epsilon) parameters of DBSCAN for the Complex8 dataset. The procedure maximizes purity of the obtained clustering, subject to the following constraints:
   * 1. There should be between 2 and 18 clusters
     2. The percentage of outliers should be 10% or less.

The procedure returns the “best” DBSCAN clustering found and the accomplished purity as its result[[2]](#footnote-2); please limit the number of tested (MinPoints, epsilon)-pairs tested to 5000 in your implementation! Explain how your automated parameter selection method works and demonstrate your automated procedure using an example! Report and visualize the best clustering you found. \*\*\*\*\*

Alternatively, you could manually search for the “best” clustering and report and visualize the best clustering; however, we will lose some points if you do not create a search procdure. \*\*

Remark: Students who found a higher purity clustering will be given higher scores; if two students found 100% purity clusters, but one student had 9% outliers and the other had 5% outliers the student with the 5% outliers will get a slightly higher score. Moreover, the student who finds a 100% purity clusters with the fewest number of outliers will be awarded the “*2023 DS1 Clustering Award*” and will win a door price; if no 100% purity clustering are found, the student with the highest purity clustering will win the door price.

1. Run K-means for k=3 and k=5 for the BWD. Report the cluster centroids, the SSE, and compute the purity of the obtained clustering results. Briefly interpret and compare the obtained results. \*\*\*\*
2. Obtain a DBSCAN clustring , having between 2 and 15 clusters with less than 20% outliers for the BWD. Report its purity and its percentage of outliers. Moreover, visualize your clustering result in the humidity×temp\_mean attribute subspace. Interpret the visualization! \*\*\*

**Deliverables for Task 3:**

1. A Report[[3]](#footnote-3) which contains all deliverables for the 5 subtasks of Task 3.
2. An Appendix which describes how to run the procedure that you developed for Task e, if you developed such a procedure.
3. An Appendix which contains the software/code you developed as part of Task 3.

**Task 3 Submission Guidelines:**

1. Name your python/R files to **COSC3337F23-PS2T3-Firstname-Lastname.ipynb** or any other appropriate extension.
2. Name the pdf copy of your report **COSC3337F23-PS2T3-Report-Firstname-Lastname.pdf** carefully.
3. Create a folder and name it **COSC3337F23-PS2T3-Firstname-Lastname**.The folder should contain both python/R file and pdf copy of your report named correctly. Compress (zip) the folder and submit it to MS TEAMS.
4. Upload the zipped folder to the Assignment tab in MS Teams **before the deadline**.

Task4: Outlier Detection for the Basel Weather Dataset

Individual Task

Second Draft

Weight: 48 points



Figure 2: Some Unusual Weather

**Learning Objectives**:

1. Learn how to create distance functions
2. Learn how to develop distance-based outlier detection detection techniques
3. Learning how to interpret, understand, and evaluate outlier detection results

In this task you will be developing distance-based outlier detection technique for the Basel Weather Dataset, you already used for Task 3; the objective is to find “*unusual weather days*” in this dataset. However, in this task we restrict ourselves to a subset of the dataset, called RBWD, focusing on the following attributes: Date (serves as an identifier), precipitation, sunshine, temp\_mean and humidity.

Subtasks:

1. Design a “good” distance function for RBWD!
2. Design and implement a distance-based outlier detection technique for RBWD! The technique if applied to the RBWD should add a column to the examples in the dataset named OLS (Outlier Score) which contains a single number which measures the strength of our belief that the particular example is an outlier.
3. Apply the outlier detection technique to the RBWD; as distance-based outlier detection techniques use hyper parameters: apply your technique 3 times to the dataset using 3 different hyper parameter settings, obtaining three augmented datasets with the OLS column added.
4. Sort the three obtained augmented datasets using the OLS attribute. Discuss the top 6 examples of each augmented dataset; explain why you believe the particular examples were viewed as likely outlier candidate. Also discuss the bottom 2 examples in each augmented dataset; try to explain why these two examples were rated to be “most normal” ones.\*\*\*\*
5. Briefly assess how well your outlier detection technique worked.
6. Deliver a 2-page single spaced report which summarizes the main findings of Task 4.

**Deliverables for Task 4:**

1. Report
2. File with constains the software you developed for Task4.

**Task 4 Submission Guidelines:**

1. Name your python/R files to **COSC3337F23-PS2T4-Firstname-Lastname.ipynb** or any other appropriate extension.
2. Name the pdf copy of your report **COSC3337F23-PS2T4-Report-Firstname-Lastname.pdf** carefully.
3. Create a folder and name it **COSC3337F23-PS2T4-Firstname-Lastname**.The folder should contain both python/R file and pdf copy of your report named correctly. Compress (zip) the folder and submit it to MS TEAMS.
4. Upload the zipped folder to the Assignment tab in MS Teams **before the deadline**.

1. This function could be an R-function, a Python function or any other function. You might find some implementation of this function online; it is okay to use those implementations, as long as you acknowledge in your report what you use, and not all software you find on the internet is running properly. [↑](#footnote-ref-1)
2. It should report the number of clusters obtained and the percentage of outliers as well. [↑](#footnote-ref-2)
3. Single-spaced; please use an 11-point or 12-point font! [↑](#footnote-ref-3)