Homework 2
COL 761

- Due: **21st Sept 6:00PM**.
- **30%** penalty for each late day.
- You are free to use any programming language but it must compile in an UNIX-based OS. Please start early. For submissions in C++, you must fix the optimization flag to **O3** [gcc ver 5.3.1]
- You can do the homework in **groups of three**. Only **one zip file** must be submitted per team.
- You should upload all your code to a Github repo. The Github page should maintain your HW1 implementations as well as the implementation of HW2. The goal is to create a single data mining library over this course. **You are not allowed to make any changes after the deadline**. Otherwise, you will be penalized as per the late submission policy.

Implement *k-means, DBSCAN* and *OPTICS*. The dataset would be in the following format, where each line corresponds to an *n*-dimensional point. Each dimension is separated by a space. The number of dimensions and lines can be of any value till 5 and ~1-million points respectively. The feature values are floating point numbers.

```
3 4 5 …
1 7 8 …
…
```

- You need to provide a script with the following functionalities
  - `sh compile.sh`: Clones your Github repo and compiles your code with respect to all implementations.
  - `sh <rollno>.sh -kmeans <k> <filename>`: should execute *k-means* algorithm with *k* as the number of clusters and produce the cluster assignment of each data point. The `<filename>` indicates the name of the dataset to read the points from.
  - `sh <rollno>.sh -dbscan -<minPts> <epsilon> <filename>`: should execute *DBSCAN* and produce the list of cluster assignment of each data point.
  - `sh <rollno>.sh -optics -<minPts> <epsilon> <filename>`: should execute *OPTICS* and plot the reachability data using `matplotlib`. You do not need to provide the cluster assignment for *OPTICS* though you are welcome to implement them on your own for verification purposes.
  - The filename indicates the name of the dataset to read the points from.

- Output format for *DBSCAN* and *k-means*. It should produce files called `dbscan.txt` and `kmeans.txt` respectively. Each file should be of the following format:

```
#<cluster ID>
<Point1 line no>
<Point 2 line no>
```

```
…
#<cluster ID>
<Point 1 line no>
<Point 2 line no>
…
```

In other words, the "#" indicates the start of a cluster ID, followed by the line number of all points belonging to the cluster. The line numbers start from 0 and can be treated as an ID of each point. For *DBSCAN*, group all outliers under the special cluster ID "`#outlier`". `<cluster ID>` should be assigned integer values only starting from 0.

1. Correctness
   a. Correct implementation of *k-means* [5 points]
   b. Correct implementation of *DBSCAN* [10 points]
   c. Correct implementation of *OPTICS* (we will evaluate the reachability plot) [10 points]
2. Create a single synthetic dataset of 100,000 2-dimensional points that showcases why *DBSCAN* is better than *k-means* and why *OPTICS* is better than *DBSCAN*. You will be assigned a demo slot to explain and your grade will depend on how comprehensively you can bring out the strengths of each technique. Your demo must be short, precise, clear, and comprehensive [40 points]
3. Efficiency: Competition based on running time for *DBSCAN* implementation. You must produce a correct implementation to be eligible for this competition. Do not use multi-threaded programming. For C implementations, you are only allowed O3 flag. The fastest would get full points. If your algorithm is `X`% slower than the fastest, then you would get `X`% of full points. [20 points]

Submission Instructions:

1. There should be only **one** submission per group.
2. Submit **one .zip** file. The name of the file should be the entry number of any **one** member of your team. On unzipping the file, it should produce one folder. The folder should have the same name as the zip file. This folder should contain all the source files and scripts required. In addition, provide a `README.txt` that contains the entry numbers of **all** team members.
3. Since your submissions will be auto-graded, it is essential to ensure your submissions conform to the format specified.

Anti-Plagiarism Policy:

Plagiarism will result in an F-grade in the course.