Lab Clustering n°4

Exercice 1 – DBSCAN :

Part 1 (Theory)

1. Prerequisites :

   Consider n points in space to be clustered, P and Q two points, $m \in N$ and $\epsilon \in R$:

   - P is called core-point if at least m points are within distance ϵ (including P itself).
   - Q is directly reachable from a core-point P if Q is within distance ϵ from P.
   - Q is reachable from a core-point P if there is a suit $(P_i)_i$ of k points where $P_0 = P$ and $P_k = Q$ and $\forall i \in [0, k-1]$, P_{i+1} is directly reachable from P_i. This suit of points is called a path between P and Q.
   - All point unreachable from any other point are called outliers.

2. Questions

   - Considering that that the DBSCAN algorithm depends on the definitions given in the previous paragraph, what are the parameters of the DBSCAN clustering algorithm?

   - Consider n equidistant points belonging to the same circle with a r radius and C the center of this circle. Let d be the distance between these n points (d = $P_0P_1 = P_1P_2 = .. = P_{n-1}P_n = P_nP_0 < r$) . How many core points will we have if we fix the couple (m, ϵ) to :

- $(2, d_1)$ with $d_1 < d$
- $(3, d)$
- (n, r)

- Let P,Q be two points where P is a core point and Q is reachable from P. Then the points forming the path between P and Q are core points ?
  - True
  - False

Part 2 (Algorithm)

1. Pseudo – code:

```
** initialization **
Number_clusters = 0
For each point P
        Label(P) = null

** clustering **
For each point P
        If Label(P) != null; continue (next iteration)
        List N = Neighboors(P, ϵ)
        If length(N) < m:
                Label(P) = noise
                continue

        Number_Clusters ++
        Label(P) = Number_Clusters
        List N = N \ {P}
        while N != []
                Q = N[0]
                N = N [1:]
```

```
            if Label(Q) = noise
                Label(Q) = Number_Clusters
                continue
            If label(Q) != null
                continue
            Label(Q) = Number_Clusters
            List M = Neighboors(Q,ϵ)
            if length(M) >= m
                N = N + M
        End while
    End For
```

2. Create the function Neighboors(Point, ϵ) which returns the list of directly reachable points from the point given as parameter.

3. Implement the entire algorithm.

4. In which part of the algorithm can we change the distance function?

5. Change the algorithm then the code in order to make the Distance function another parameter of the algorithm.

6. Create a randomly generated set of points and test your algorithm.