# CMPT 459
# Assignment 3

Professor Martin Ester

Submission By: Muhammad Arslan
Student ID: 301296874

**Assignment.3.1a)**

**Prove the following property: If an itemset is not frequent in DB and not frequent in ΔDB, then it cannot be frequent in in DB U ΔDB**

Item X is infrequent in a data set if the support of item set X in dataset D is less than some min-support. The support is given by the formula:

$$\sup(X, DB) = \frac{|\{T\epsilon\ DB\ |X \subseteq T\}|}{|DB|}$$

Let |DB| = **c1** and |ΔDB| = **c2**

Let **x1** be the $|\{T\epsilon\ DB\ |X \subseteq T\}|$
Let **x2** be the $|\{T\epsilon\ \Delta DB\ |X \subseteq T\}|$

Let **s** be the minimum support.

Using support formula, the two equations we get from
**Itemset is not frequent in DB and not frequent in ΔDB**
**then it cannot be frequent in in DB U ΔDB**

If $\frac{x1}{c1} < s$ …(1) AND $\frac{x2}{c2} < s$ …(2)

Then, prove $\frac{x1+x2}{c1+c2} < s$

Proving the equations mathematically:

From (1), we get $x1 < c1 \cdot s$
From (2), we get $x2 < c2 \cdot s$

Adding the two inequalities

$x1 + x2 < c1 \cdot s + c2 \cdot s$
$x1 + x2 < s(c1 + c2)$

$$\frac{x1 + x2}{c1 + c2} < s$$

**Assignment3.1b)**
From the above proof we know that if an item set is infrequent in DB and ΔDB, then it is infrequent in DB U ΔDB. From this we can imply that if an item set is frequent in either DB or ΔDB, then it might be frequent in DB U ΔDB.

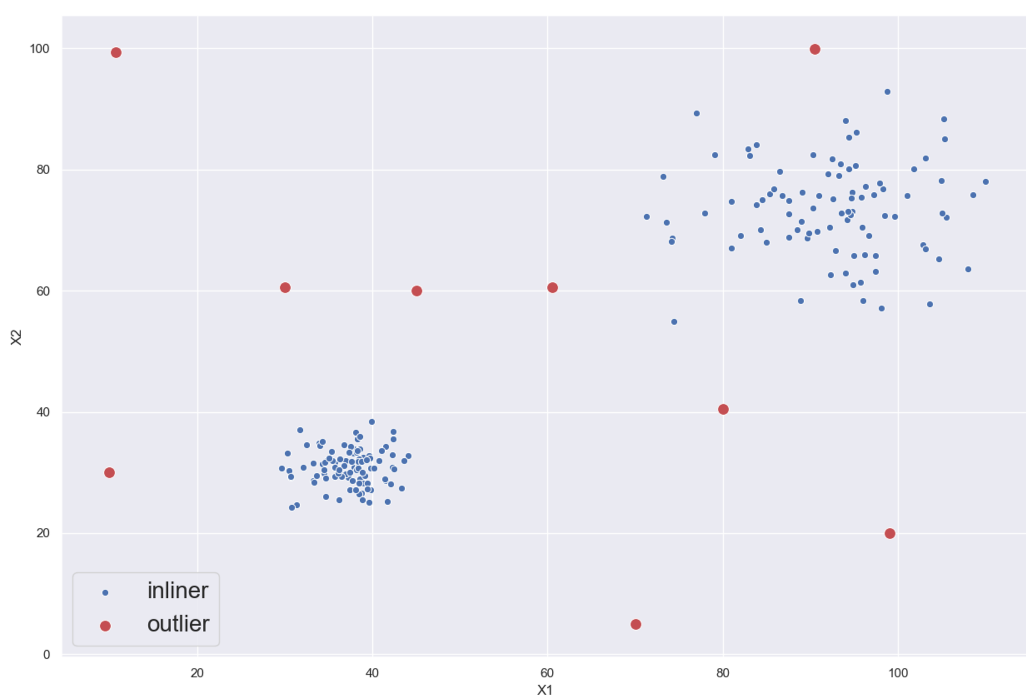
So, when counting support in DB, we should count the item sets that are frequent in ΔDB and not in DB. This is because we know that it is infrequent in DB and by above implication, we might get it as a frequent itemset in DB U ΔDB, if it is frequent in ΔDB.
Similarly, when counting support in ΔDB, we should count the item sets that are frequent in DB and not in ΔDB. This step of algorithm did not return some itemset as frequent in ΔDB but if it is frequent in DB, then we might get it as a frequent item set

Assignment3.2

The small values for k do not work well with this algorithm. The values of k between 2 and 9, we see a lot of fluctuations in the result. For k=2, the algorithms show a lot of false positives, and that is expected because cluster are generally larger than 2 data points. Picking up k between 5 to 9 gives a relatively good result on the provided data, but we would still expect cluster to be larger than that. The smaller the k, the smaller is the neighborhood and LOF assigned to each point gets higher with lower k. A k-value of 15 is found to be decent for this data and also if there is new data coming up. A k-neighbourhood of 15 can expected where cluster happens. For current dataset, the results remain same for k between 10 and 40.

The choice of threshold is generally a smaller value bringing clusters together and assigning high LOF to datapoints away from cluster. And also varies a little with different k values. With k=15, 75% data points are greater than LOF values above 1. We cannot obviously choose 1 here because 75% of data cannot be outliers. And only 4.5% of datapoints are greater than LOF = 2. This gives a more promising results and also looking at the scatter plot below, we can see the red points are certainly outliers.

So, to conclude, it all depends on the kind of data we have. It is generally a good choice to start with k as 10 and go up from there to see and the best results. And for threshold, start with at least 1 and see what percentage of values are detected as outliers and increase the threshold to get good results. Visualisation certainly help in choosing these values. LOF removes any bias of dense or sparse cluster. But it's the job of data scientist to decide what number of data points is a cluster in data set being worked on.



**References:**

[Breunig, Kriegel, Sander & Ng 2000]
Breunig, Markus M ; Kriegel, Hans-Peter ; Ng, Raymond T ; Sander, Jörg
LOF: Identifying Density-Based Local Outliers. SIGMOD record, 2000-06, Vol.29 (2), p.93-104
SFU library resources

https://stackoverflow.com/questions/57107729/how-to-compute-multiple-euclidean-distances-of-all-points-in-a-dataset