# Project 2 First Cut

Hung Yee Wong a1815836

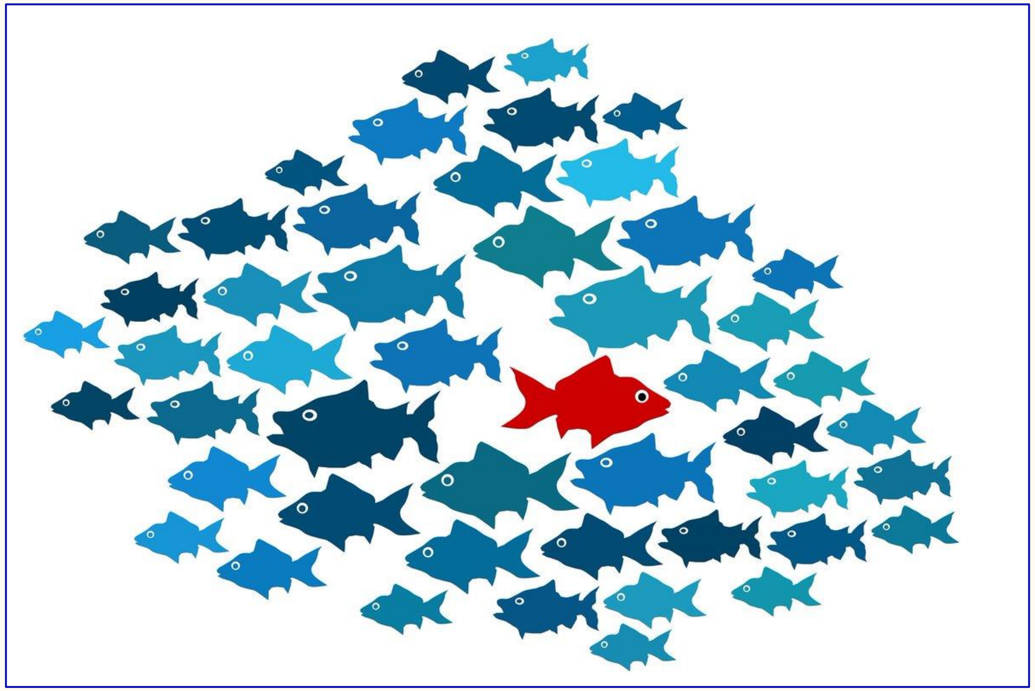
# Topic Introduction (Data Science)

Here's a brief explanation of the topic :

Data science in simple words can be defined as an interdisciplinary field of study that uses data for various research and reporting purposes to derive insights and meaning out of that data

This helps businesses and organizations make informed decisions backed up by data analysis

# The Challenge (more covered later)



Notice that the red fish looks significantly different compared to the other fish !

- During the first project, I carried out data preparation or data cleaning to sort out redundant or missing data

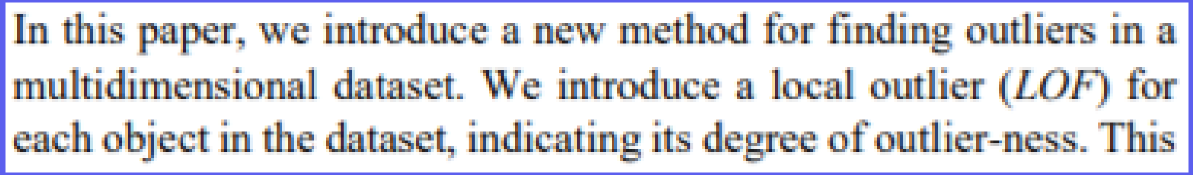
- This brings us to a specific challenge in data cleaning which is outlier detection.

- The picture on the left gives a rough idea of what is an outlier and the aim is to find an accurate outlier detection method

# Review of the first paper

First, we review a highly cited and popular solution, the Local Outlier Factor (LOF) algorithm

Outliers are traditionally considered in binary terms, for example, a point IS or IS NOT an outlier



In this paper, we introduce a new method for finding outliers in a multidimensional dataset. We introduce a local outlier (*LOF*) for each object in the dataset, indicating its degree of outlier-ness. This

This paper aims to show for each point how ‘outlier-like’ or the degree of ‘outlier-ness’ of a specific point.

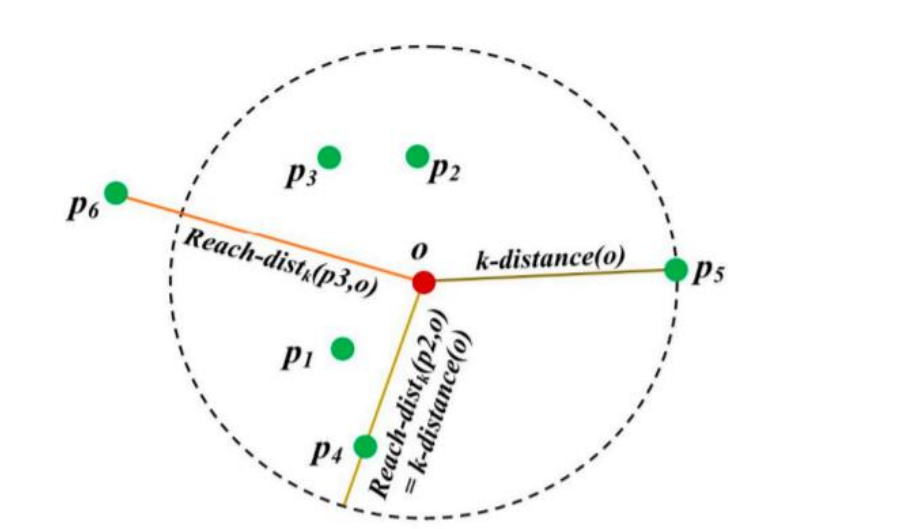
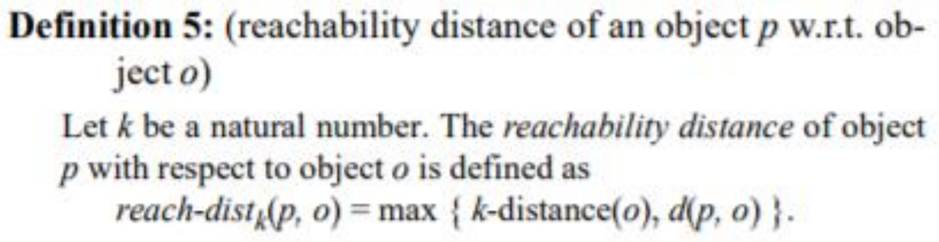


**Figure 4.** The reachability distance for different data points *p* with regard to *o*, when *k* equals 5.

- A K-Nearest Neighbours approach is used, in the example, k = 5, so the 5 closest points to o form a neighbourhood. Then, the k-distance is the maximum distance from the centre, o, to the furthest point in the neighbourhood.

# Continued review of the first paper

- After obtaining the k-distance, reachability distance of a point p, to an object, o, is defined as

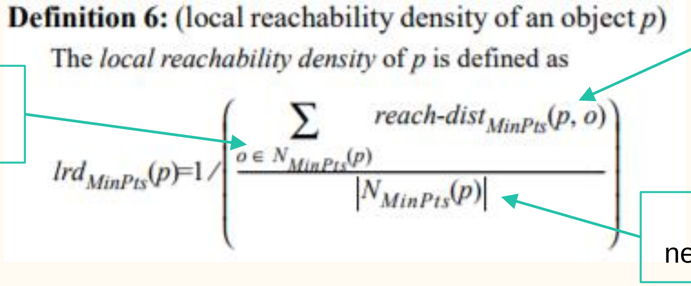


**Definition 5:** (reachability distance of an object *p* w.r.t. object *o*)

Let *k* be a natural number. The *reachability distance* of object *p* with respect to object *o* is defined as

$$reach\text{-}dist_k(p, o) = \max \{ k\text{-distance}(o), d(p, o) \}.$$

- Then using this formula, the reachability density of the point, p, is calculated :



**Definition 6:** (local reachability density of an object *p*)

The *local reachability density* of *p* is defined as

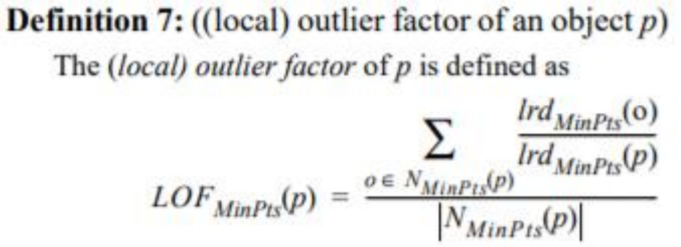
$$lrd_{MinPts}(p) = 1 / \left( \frac{\sum_{o \in N_{MinPts}(p)} reach\text{-}dist_{MinPts}(p, o)}{|N_{MinPts}(p)|} \right)$$

O is one of the K-nearest neighbours of P

Reach-dist of P w.r.t O

Number of K-nearest neighbours

- Finally, the local outlier factor, LOF for the point is calculated :



**Definition 7:** ((local) outlier factor of an object *p*)

The *(local) outlier factor* of *p* is defined as

$$LOF_{MinPts}(p) = \frac{\sum_{o \in N_{MinPts}(p)} \frac{lrd_{MinPts}(o)}{lrd_{MinPts}(p)}}{|N_{MinPts}(p)|}$$

- Where a threshold is set by user to filter outliers, generally close to 1, if larger, considered as outlier

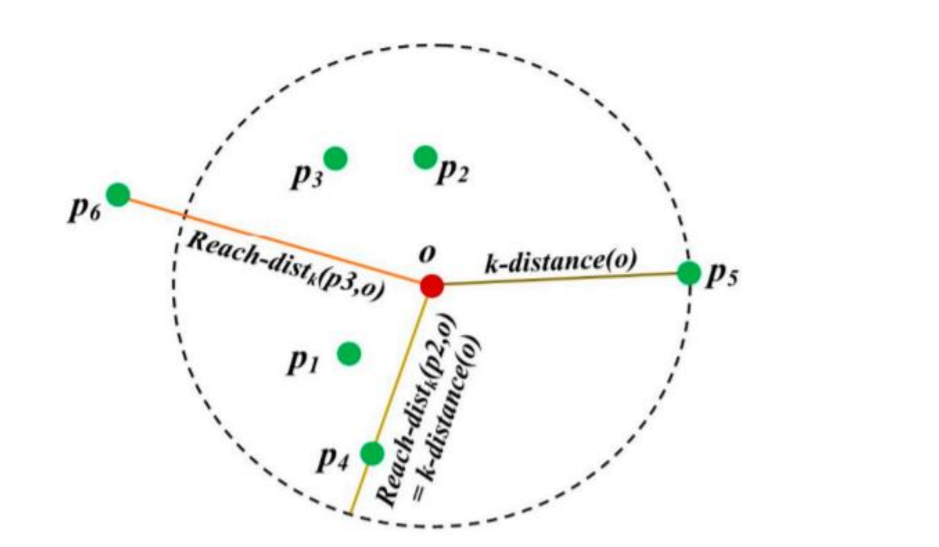
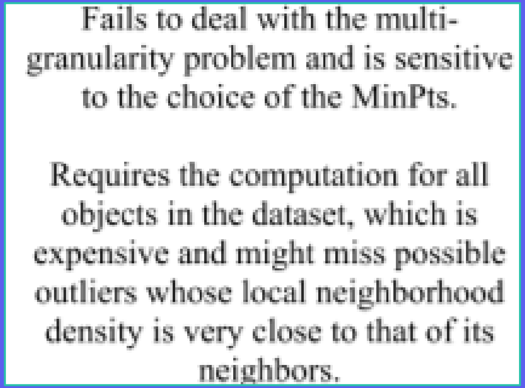


**Figure 4.** The reachability distance for different data points *p* with regard to *o*, when *k* equals 5.

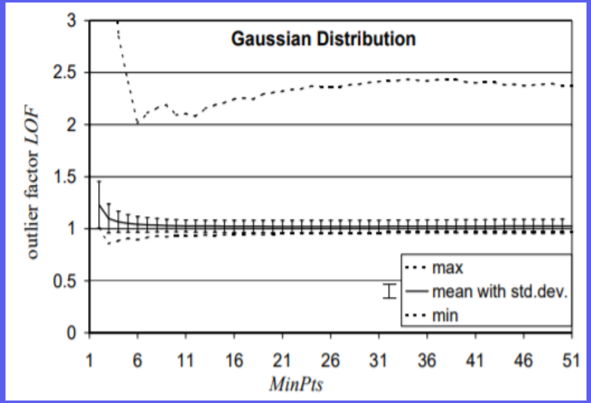
If P is within the neighbourhood of O, the reachability distance is the k-distance(O), else the reachability distance is the distance between P and O itself

# Drawbacks of the first method

- From a survey paper published, this method has a couple of drawbacks

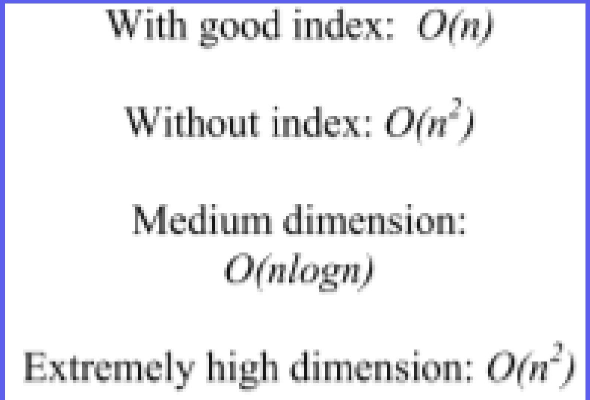

Fails to deal with the multi-granularity problem and is sensitive to the choice of the MinPts.

Requires the computation for all objects in the dataset, which is expensive and might miss possible outliers whose local neighborhood density is very close to that of its neighbors.



First drawback is that the LOF algorithm is very sensitive to the choice of MinPts, which is the number of nearest neigbours of each point, k, determined by the user

Another drawback is that from the runtimes given in another survey, the algorithm is computationally expensive for high dimensions of data


With good index: $O(n)$

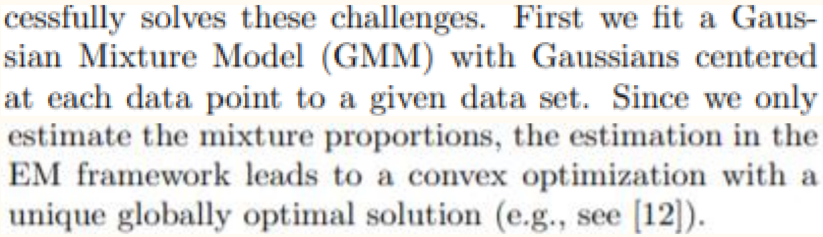
Without index: $O(n^2)$

Medium dimension: $O(nlogn)$

Extremely high dimension: $O(n^2)$

# Review of the second paper

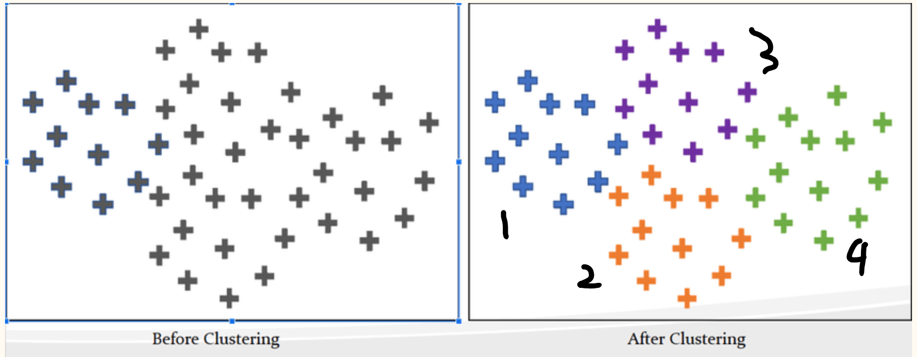
The second paper employs a statistical method which is the use of the Gaussian Mixture Model (GMM).



cessfully solves these challenges. First we fit a Gaussian Mixture Model (GMM) with Gaussians centered at each data point to a given data set. Since we only estimate the mixture proportions, the estimation in the EM framework leads to a convex optimization with a unique globally optimal solution (e.g., see [12]).

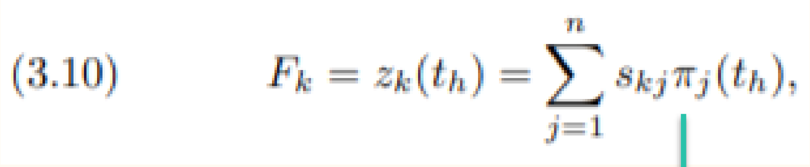
First, an iterative Expectation-Maximization(EM) algorithm is employed to fit the Gaussian Mixture Model to the dataset

What this does is, for a real number k, the probability of each data point being the centre of a gaussian distribution is calculated and reiterated to produce clusters



Before Clustering

After Clustering

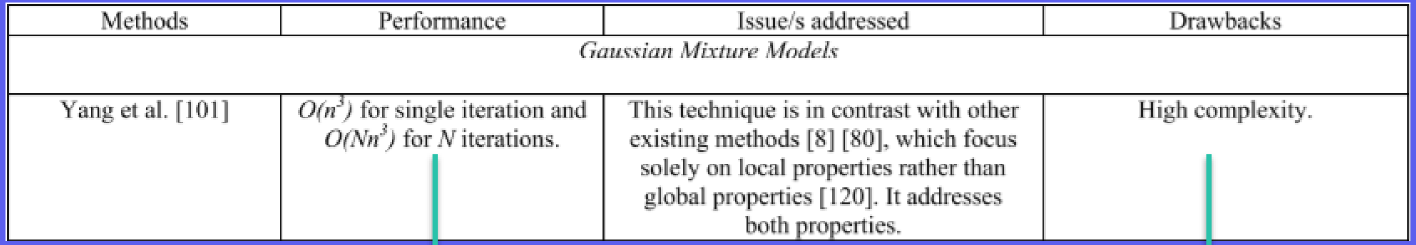


$$(3.10) \qquad F_k = z_k(t_h) = \sum_{j=1}^{n} s_{kj}\pi_j(t_h),$$

Then, the outlier factor for each point is calculated. The smaller this value, the larger probability that is it an outlier

In particular, the term $s_{kj}\pi_j(t_h)$ represents how much point $x_k$ is influenced by point $x_j$ with $s_{kj}$ being the strength of the connection and $\pi_j(t_h)$ measuring the importance of point j.

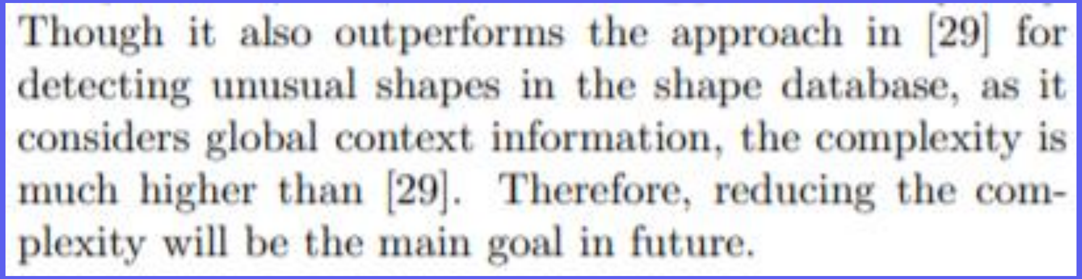
# Drawbacks of the second method



| Methods | Performance | Issue/s addressed | Drawbacks |
|---|---|---|---|
| *Gaussian Mixture Models* | | | |
| Yang et al. [101] | $O(n^3)$ for single iteration and $O(Nn^3)$ for $N$ iterations. | This technique is in contrast with other existing methods [8] [80], which focus solely on local properties rather than global properties [120]. It addresses both properties. | High complexity. |

This is very evident as this method proposes the use of an EM algorithm to fit a GMM which is very domain specific to areas such as Machine Learning and requires a great amount of domain knowledge to implement. Thus, errors are more evident.

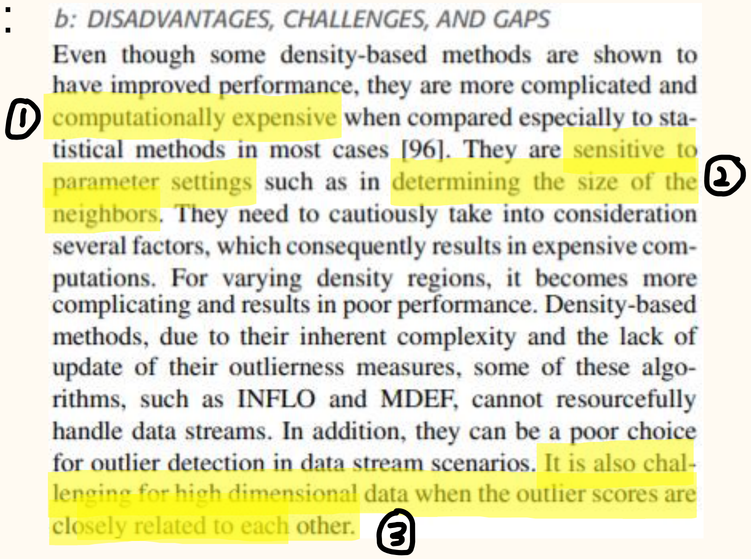
We also note that the performance of this algorithm is generally larger than $O(n^3)$, this makes computation very expensive. In comparison, LOF algorithms normally run at $O(n^2)$.



Though it also outperforms the approach in [29] for detecting unusual shapes in the shape database, as it considers global context information, the complexity is much higher than [29]. Therefore, reducing the complexity will be the main goal in future.

# Solution Motivation

My solution aims to solve some of the challenges faced by the current solutions, especially density based algorithms, we take a look at what a survey says :



*b: DISADVANTAGES, CHALLENGES, AND GAPS*

Even though some density-based methods are shown to have improved performance, they are more complicated and 1 computationally expensive when compared especially to statistical methods in most cases [96]. They are sensitive to parameter settings such as in determining the size of the 2 neighbors. They need to cautiously take into consideration several factors, which consequently results in expensive computations. For varying density regions, it becomes more complicating and results in poor performance. Density-based methods, due to their inherent complexity and the lack of update of their outlierness measures, some of these algorithms, such as INFLO and MDEF, cannot resourcefully handle data streams. In addition, they can be a poor choice for outlier detection in data stream scenarios. It is also challenging for high dimensional data when the outlier scores are closely related to each other. 3

From the survey, we obtain a few common challenges :

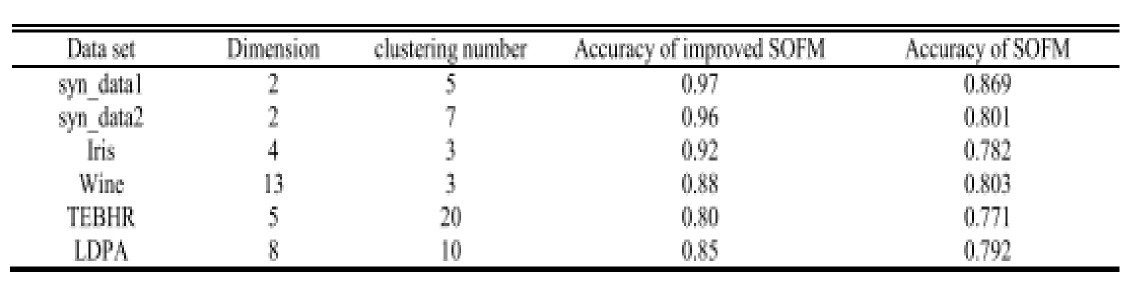
- Computationally expensive
- Sensitive to parameter settings
- Curse of dimensionality

Therefore, to try and solve these problems, we may try to:

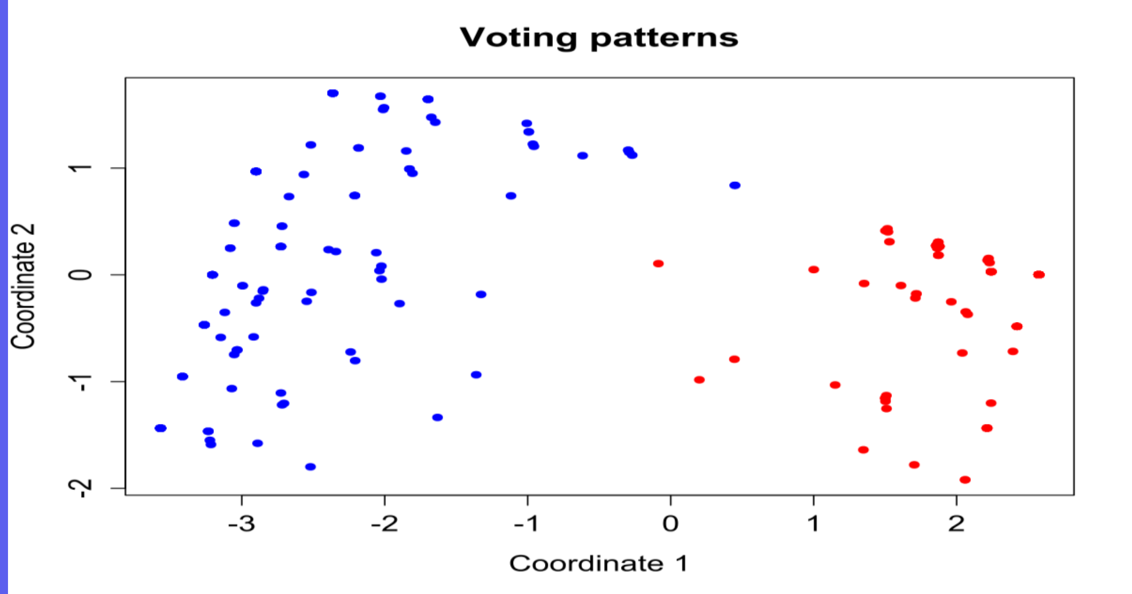
- Reduce the number of iterations needed
- Change the type of distance metric
- Carry out dimension reduction
- Cluster the data

# Proposed Solution

In the pitch, the solution that uses a Self Organizing Feature Map (SOFM) to cluster the data comes with a drawback, for high dimension data, the accuracy is decreased, so features or information may be lost :



| Data set | Dimension | clustering number | Accuracy of improved SOFM | Accuracy of SOFM |
|---|---|---|---|---|
| syn_data1 | 2 | 5 | 0.97 | 0.869 |
| syn_data2 | 2 | 7 | 0.96 | 0.801 |
| Iris | 4 | 3 | 0.92 | 0.782 |
| Wine | 13 | 3 | 0.88 | 0.803 |
| TEBHR | 5 | 20 | 0.80 | 0.771 |
| LDPA | 8 | 10 | 0.85 | 0.792 |



- Thus, I propose the use of Multidimensional Scaling (MDS) coupled with a clustering algorithm
- MDS reduces the dimension of the data while preserving the distances between objects
- Clustering is then done on the MDS output and the result is used as input for our density/distance based algorithms

Lastly, to further reduce the curse of dimensionality, we can try using other suitable distance measurement metrics which may help improve performance.

# Conclusion

In conclusion, even though a lot of research has been done on outlier detection. A one size fits all solution is yet to be found. Further research should be carried out to obtain more generally applicable solutions.

Some future research directions :

- Are there even better dimensionality reduction approaches ?
- Is there a way to accurately determine the parameters needed for each algorithm ?
- Other than dimensionality reduction and distance metric approaches, how do we further reduce the curse of dimensionality ?

# References

[1] M. Breunig, H. Kriegel, R. T. Ng, and J. Sander, ‘‘LOF: Identifying density-based local outliers,’’ ACM SIGMOD Rec., vol. 29, no. 2, pp. 93–104, 2000

[2] X. Yang, L. J. Latecki, and D. Pokrajac, ‘‘Outlier detection with globally optimal exemplar-based GMM,’’ in Proc. SIAM Int. Conf. on Mining (SDM), Apr. 2009, pp. 145–154.

[3]H. Wang, M. J. Bah and M. Hammad, "Progress in Outlier Detection Techniques: A Survey," in IEEE Access, vol. 7, pp. 107964-108000, 2019, doi: 10.1109/ACCESS.2019.2932769.

[4] Alghushairy, O.; Alsini, R.; Soule, T.; Ma, X. A Review of Local Outlier Factor Algorithms for Outlier Detection in Big Data Streams. Big Data Cogn. Comput. 2021, 5, 1. https://doi.org/10.3390/bdcc5010001