

OdeSIS: in-depth analysis of outlier detection positioning using median absolute deviation

Alvincent E. Danganan

Department of Computer Science, College of Computer Studies, Tarlac State University, Tarlac City, Philippines

Article Info

Article history:

Received Sep 14, 2025

Revised Aug 17, 2026

Accepted Aug 27, 2026

Keywords:

Algorithm

Clustering

Data mining

K-means

Outlier

ABSTRACT

Outliers are easy to overlook, but they quietly skew data analysis and the decisions built on it. This study examined how the median absolute deviation (MAD) algorithm performs when applied before or after K-means clustering. To test this, the study used a synthetic dataset with five deliberately inserted outliers and an augmented version of the Iris setosa dataset containing three anomalies. MAD was applied first, then again after clustering with two and three clusters. The results were clear. When MAD ran first, it detected every outlier with 100% accuracy rate and zero false positives. After clustering, its performance slipped, false positives appeared, and it failed to flag genuinely present outliers. K-means is likely the culprit. Its mean-based centroids get dragged toward anomalous points, so outliers get absorbed into clusters where they no longer look unusual, and MAD stops flagging them. These findings suggest MAD works best as a pre-processing step. Detecting anomalies before clustering improves outlier identification, preserves the underlying data, and prevents stray points from steering cluster formation.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Alvincent E. Danganan

Department of Computer Science, College of Computer Study, Tarlac State University

Tarlac City, Philippines

Email: avdanganan@tsu.edu.ph

1. INTRODUCTION

Advancements in technology have led to the emergence of different new fields. Various areas, such as engineering, health, and business, generate and collect large amounts of data every day [1]. The process of discovering useful knowledge from databases, data warehouses, and other information repositories is known as data mining [2]. Data mining is a vital area that manages and extracts the required information from these enormous amounts of data. Clustering is one of the many data mining techniques that is widely used in the knowledge discovery process. Clustering is an unsupervised technique that lets similar objects naturally fall into the same group. The idea is simple: things that belong together should end up together, because they resemble each other more closely than they resemble objects in other clusters [3]. Clustering is a core technique in data analysis, with applications spanning fields like data mining, machine learning, pattern recognition, and image analysis [4]. Many studies in clustering have largely relied on the K-means algorithm [5]–[7], a method that's widely favored and well-proven thanks to its straightforward nature, efficiency, and ability to handle large amounts of data. However, most of the clustering algorithms are sensitive to outliers. An outlier is a data point or observation that departs significantly from the rest of the observations. It is a technique used to identify abnormalities in each dataset. Outliers can mess with the clustering results. Outliers can distort the cluster structure, so the groups that emerge may misrepresent what the data actually look like [8]. Tracing outliers has essential applications in data cleaning and the mining of abnormal points for fraud detection, intrusion detection, network sensors, stock market analysis, and marketing. Various

clustering algorithms have been proposed to address outliers and noise. K-means clustering with outlier removal does two things at once: it groups the data and detects and removes outliers [9]. A new k-means variant handles outliers through a self-adaptive weighting scheme that continually adjusts point weights as the algorithm runs, so noisy observations steadily lose influence, yielding more robust, faithful clusters [10]. A study [11] proposed an enhanced K-means algorithm that integrates cluster refinement with anomaly detection. The method uses Euclidean distance between data points and their cluster centroids to identify potential outliers, which are subsequently removed before the centroids are recalculated. This iterative process reduces intra-cluster variance and aims to produce tighter and better-defined clusters than traditional K-means clustering.

Outliers are a quiet problem in clustering analysis. A single stray point can pull on the data's underlying structure and keep clusters from forming as tight, well-separated groups. For that reason, catching and dealing with outliers is something analysts cannot afford to skip. But there is a genuine question about when detection should occur. An outlier can be flagged before clustering begins or spotted only after an initial run of the algorithm; the two approaches do not necessarily yield the same cluster structure. The sensible way to settle it is empirical: compare the quality of the clusters produced with and without the outliers removed, and let the numbers show whether removing them actually improves cohesion and separation [12].

In this regard, the study aims to analyze how effective and accurate the median absolute deviation (MAD) method is at detecting outliers before clustering, or whether it is more reliable to apply it during or after clustering. The goal is to pinpoint which of these approaches helps us identify outliers more precisely, especially in datasets where the clusters themselves might be quite complex, perhaps having two or more distinct groups.

This research will also examine how well the MAD method performs when outliers are either clearly separate from the main data groups or, more challenging, hidden right inside those clusters. By studying how MAD behaves in these varied situations, the study intends to offer concrete, evidence-based insights into how the placement of outlier detection strategically affects the overall quality of data segmentation and flags unusual points.

This investigation is sparked by earlier work in [13], which highlighted how crucial the methodological placement of outlier detection can be. Building on that foundation, our current study plans to offer a more detailed and technical deep dive, with the hope that our findings can help shape best practices in statistical modeling, data preparation, and robust clustering techniques.

2. METHOD

2.1. Operational framework

In this section, the study illustrates two workflows for the clustering application, as depicted in Figures 1 and 2. Figure 1 demonstrates the clustering workflow, in which outlier detection is applied before clustering the datasets. In Figure 2, clustering is performed first, followed by outlier detection in the datasets.

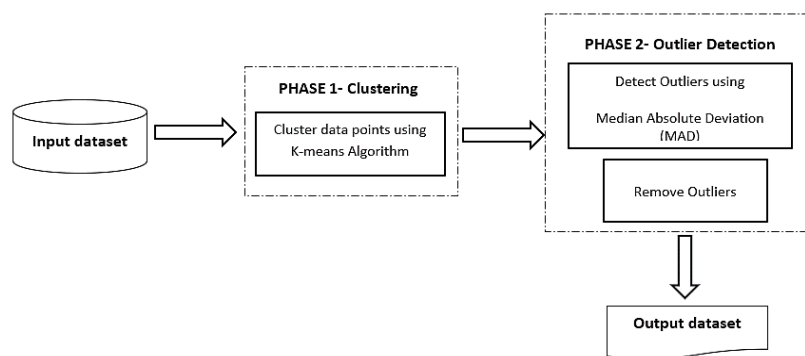


Figure 1. Implementation of outlier detection after clustering of data

2.2. Outlier detection using median absolute deviation

In this section, the MAD process is discussed. To calculate MAD: first, the collected data objects are ranked in ascending order. Then the median of the data series is calculated. Subsequently, the calculated median is subtracted from each data object to obtain the absolute value. These absolute deviations are then sorted in ascending order to determine the MAD. Finally, the MAD is multiplied by a constant $b=1.4826$ [13] to obtain the MAD value. The MAD formula [13] is defined in (1):

$$MAD = b M_i(|x_i - M_j(x_j)|) \quad (1)$$

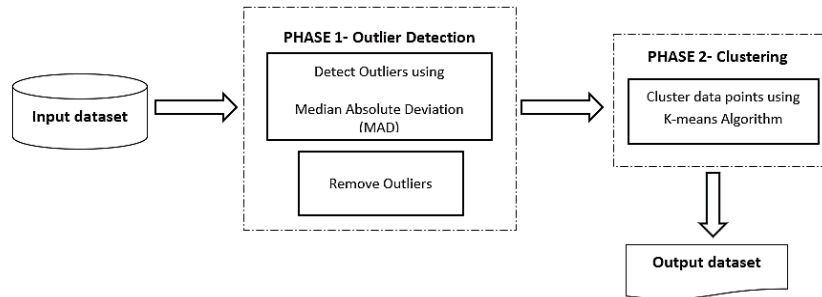


Figure 2. Implementation of outlier detection before clustering of data

To detect outliers, a criterion is calculated as the median plus or minus a threshold (± 2 , ± 2.5 , or ± 3) multiplied by the MAD. Typically, a threshold of 2.5 is recommended as the default for outlier detection. In (2) provides the specific criterion value corresponding to this approach.

$$M + 2.5 * MAD \text{ or } M - 2.5 * MAD \quad (2)$$

2.3. Clustering using the K-means algorithm

The K-means algorithm aims to partition data objects into k clusters, with each object assigned to the cluster closest to it. In this section, the researchers detail the workings of the K-means algorithm:

Initially, the user specifies the number k of clusters, and the algorithm randomly generates initial cluster centroids. Subsequently, it computes the Euclidean distance between each data object and each initial centroid. Each data object is then assigned to the nearest cluster centroid, followed by recalculation of the centroids. This iterative process continues until the assignments of data objects stabilize. The Euclidean distance formula [14] is defined in (3):

$$d_{Euclidian}(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (3)$$

2.4. Accuracy measurement

To assess how accurately MAD outlier detection results aligned with the ground truth, this study calculated recall, precision, and F-measure [14] based on correctly identified true outliers. The calculations for precision and recall focused on correctly identifying genuine outliers and accounting for any false positives. If there are many false positives, precision will naturally be low. Conversely, recall gauges how effective the MAD outlier detection method was at catching most, if not all, of the true outliers while keeping false negatives to a minimum. A low recall signifies that MAD missed a significant number of actual outliers (false negatives). Equation lays out the formula used for precision (4) and equation provides the formula for recall (5).

$$\text{Precision} = \frac{TP}{TP + FP} \quad (4)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (5)$$

Let's break down how the study classifies the predictions: a true positive means MAD correctly identified something as an outlier when it genuinely was one. A false positive occurs when MAD predicts that something is an outlier, but it isn't in reality. Lastly, a false negative is when MAD fails to identify an actual outlier, mistakenly predicting it is not one.

To get a comprehensive picture of both precision and recall, the study used the F-measure, also known as the F1 score. This score combines precision and recall by calculating their weighted harmonic mean. A higher F1 score indicates excellent detection accuracy, with a perfect score of 1 signifying flawless detection and a score of 0 representing the worst possible outcome. The calculation of the F-measure is given by (6):

$$F_1 \text{ Score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (6)$$

3. RESULTS AND DISCUSSION

In this section, two experiments were conducted to assess the algorithm's precision in identifying the exact locations of outliers in the datasets, either before or after clustering.

3.1. Experiment 1

For experiment 1, this study uses synthetic datasets to assess how accurately the outlier detection process identifies unusual data, both before and after clustering. A study [15] uses synthetic datasets for testing because they provide complete control over the data-generation process. This control enables precise scenario design and accurate evaluation of algorithm performance under known conditions.

Each of these datasets had 50 features and purposely included 5 outliers. We added these anomalies specifically to test whether the detection method could correctly identify nonconforming data. Figure 3 shows how the algorithm visualizes the data; the circled data points clearly highlight the deliberately inserted abnormalities.

In Figure 4, we've illustrated the algorithm's visualization of detected abnormalities in the data. This really underscores its strength in accurately identifying these anomalies before any clustering or grouping of the regular data points. Isolating these anomalies early on results in a more efficient clustering process that isn't negatively affected by any outliers. Figure 5 displays a far cleaner dataset, with the algorithm having successfully isolated the abnormalities found.

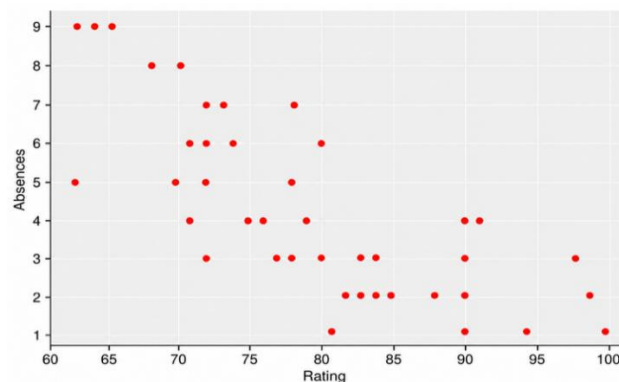


Figure 3. Scatter plot of synthetic datasets

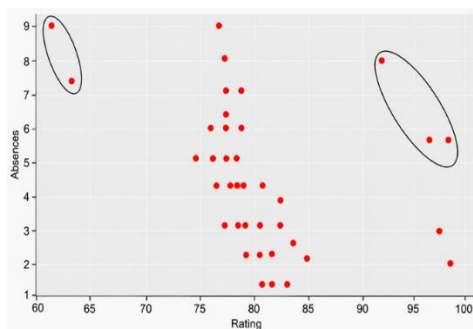


Figure 4. Scatter plot of synthetic datasets with identified outliers prior to clustering

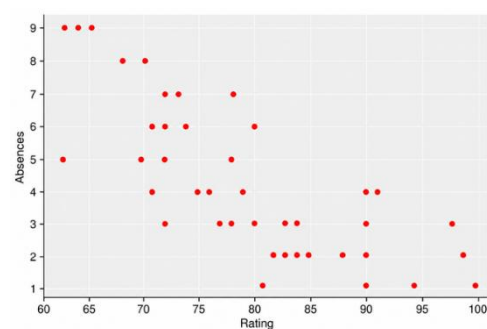


Figure 5. Scatter plot of synthetic datasets with isolated outliers

Looking at Figure 6 illustrates the algorithm's clustering process, which clearly results in two distinct groups. With the abnormalities removed, the algorithm achieves much better clustering, resulting in more precise, well-separated groups. Figure 7 shows the initial visualization of two clusters before any abnormalities were removed from the data. As the figure clearly illustrates, the anomalies we inserted ended up within the clusters the algorithm created.

This time, we're testing how well the algorithm can spot oddities in the data. Figure 8 clearly shows that the algorithm is quite sensitive to noise, often pulling those noisy bits right into the clusters. Since

clustering is performed before noise isolation, the algorithm effectively identifies true outliers. However, it falsely tagged normal data points as anomalies (red circled data points), even when they're perfectly valid observations.

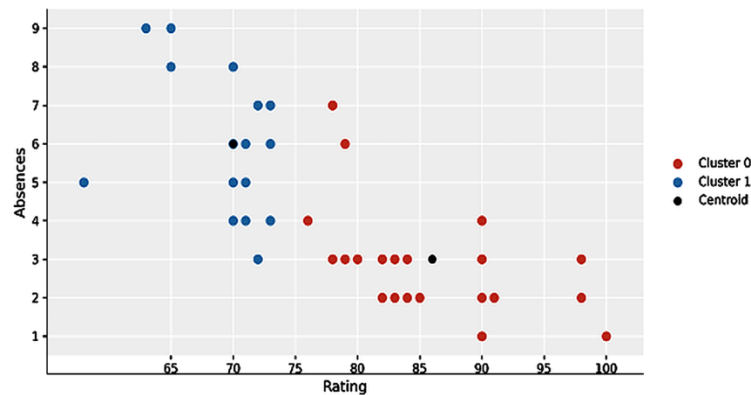


Figure 6. Scatter plot of synthetic datasets with two (2) clusters

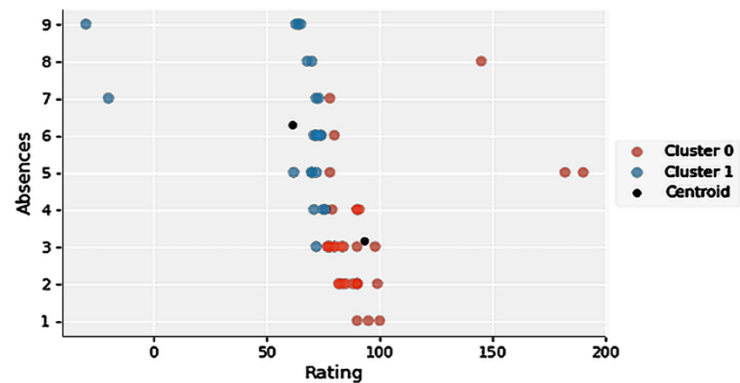


Figure 7. Scatter plot of synthetic datasets with two (2) clusters and outliers

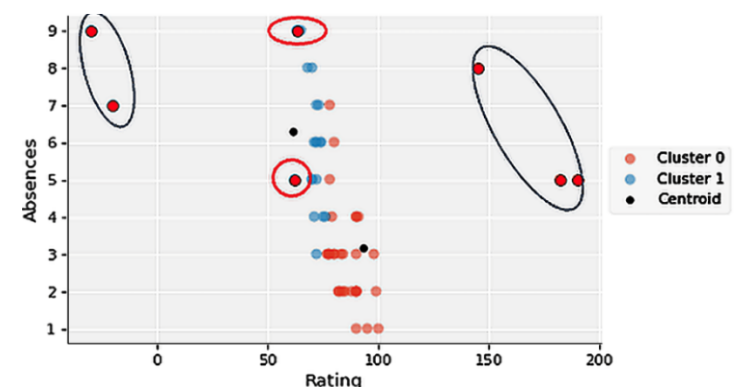


Figure 8. Scatter plot of synthetic datasets with two (2) clusters and identified outliers

Another test was conducted, and this time the algorithm created three clusters. Figure 9 shows how these clusters are formed. What's clear is that even the extreme values still got pulled into the clusters. As a result, those oddities in the data influenced the cluster centers, causing them to be classified as normal data points.

Figure 10 clearly illustrates that, because of those odd data points mixed into the clusters, the algorithm mistakenly identified normal data points as outliers (red-circled data points). It becomes challenging for the algorithm to pinpoint the true outliers when the added anomalies fall within the clusters.

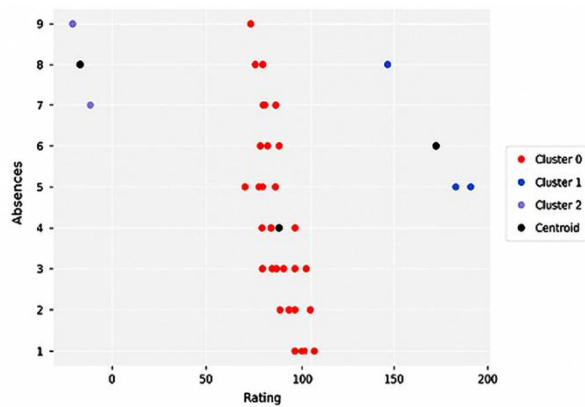


Figure 9. Scatter plot of synthetic datasets with three (3) clusters and outliers

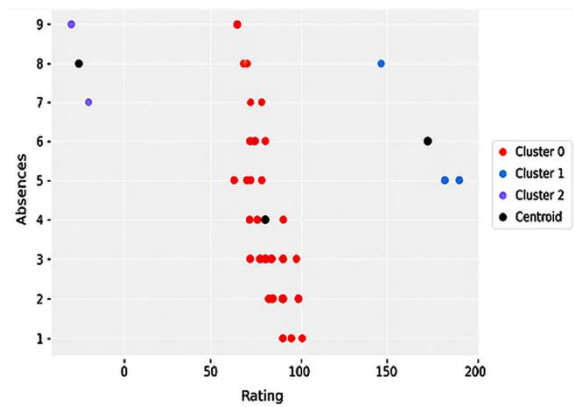


Figure 10. Scatter plot of synthetic datasets with three (3) clusters and identified outliers

For this initial test, we wanted to see whether the algorithm should identify anomalies before or after cluster formation. Using synthetic datasets, we discovered that it was far more accurate at picking out abnormalities when that process happened before any grouping, rather than finding the unusual data after it has been clustered.

3.2. Experiment 2

This second experimentation involved testing the algorithm with real datasets. The study aimed to determine whether using genuine data would alter the initial findings and how the algorithm would perform under those conditions. The study worked with the well-known Iris dataset, which we obtained from the UCI Machine Learning Repository [16], [17]. This benchmark dataset is widely used [16], [18], [19] in machine learning and comprises 150 instances, equally divided among three Iris flower species: *Iris setosa*, *Iris versicolor*, and *Iris virginica*. Each flower is characterized by four numerical measurements: sepal length, sepal width, petal length, and petal width.

The focus of this experiment was exclusively on the *Iris setosa* [20], [21] subset, which contains 50 samples. This particular class is often considered clean and distinct from the other species, with no inherent outliers. To rigorously test our outlier detection method, we intentionally introduced three synthetic outliers into this *setosa* subset. These artificial anomalies were added specifically to assess how sensitive and effective our model would be at identifying deviations within an otherwise pristine dataset.

Figure 11 shows the *Iris setosa* data after the algorithm has done its work, with artificial anomalies already woven into the dataset. The three circled spots mark exactly where those anomalies were planted. This visualization matters because it visualizes how the algorithm treats the injected outliers, whether it flags them, absorbs them into the clusters, or lets them pass unnoticed. In short, it gives a concrete basis for judging how well the algorithm recognizes and responds to the anomalies it was never meant to encounter.

After loading the *Iris setosa* subset and visualizing the data, we put the algorithm to the test to identify unusual entries. As Figure 12 illustrates, the algorithm successfully flagged every anomaly before the cluster formed.

We ran another test, this time assessing the algorithm's performance with a different cluster arrangement. The first step was to create two clusters to assess whether the algorithm could accurately identify anomalies in the data. Figure 13 illustrates the outcomes, showing precisely how the model pinpointed these anomalies. As you can see in Figure 13, while the algorithm successfully recognized the genuine outlier, it also incorrectly flagged two regular data points (circled in red) as outliers (a false positive).

Another test was conducted, in Figure 14, the algorithm was tested using a three-cluster arrangement. The results indicate that, much like the previous experiment, the algorithm incorrectly flagged a regular data point as an anomaly. And this misidentified point originated from a different cluster. While the algorithm successfully identified the genuine outliers, it also mistakenly labeled a non-outlier as abnormal.

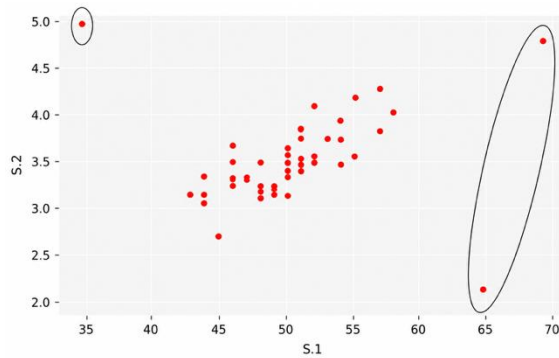


Figure 11. Scatter plot of Iris setosa subset

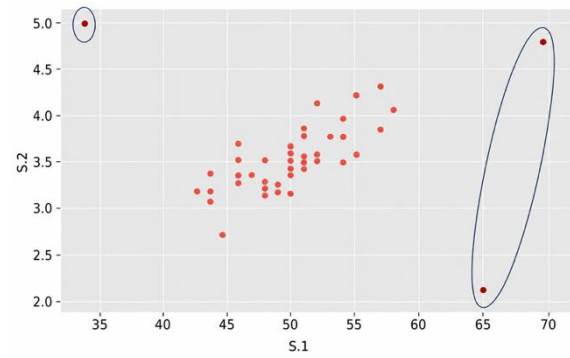


Figure 12. Scatter plot of Iris setosa subset with identified outliers

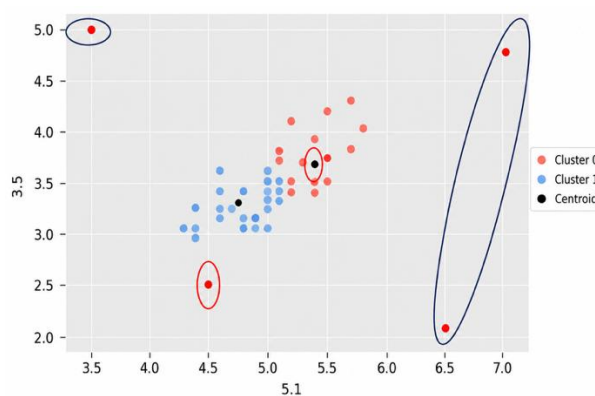


Figure 13. Scatter plot of synthetic datasets with two (2) clusters and identified outliers

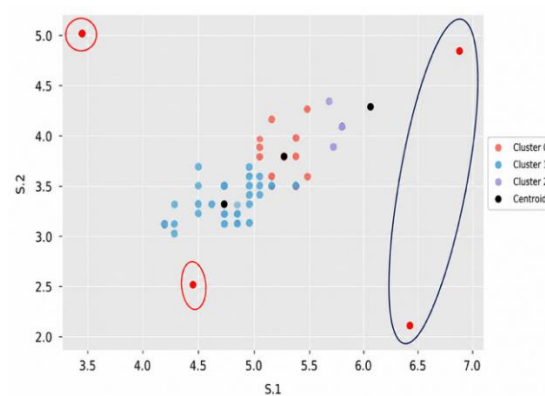


Figure 14. Scatter plot of synthetic datasets with three (3) clusters and identified outliers

Table 1 clearly shows that applying the MAD method prior to clustering significantly improved outlier-identification accuracy across the datasets. The results from this series of experiments provide strong evidence that detecting outliers before clustering is an effective approach across various datasets.

Table 1. Accuracy results

Datasets	Anomaly recognition applied	Precision	Recall	F-measure
Synthetics	Pre-clustering stage	1.0	1.0	1.0
	Post-clustering stage (2 clusters)	0.714	1.0	0.833
	Post-clustering stage (3 clusters)	0.0	0.0	0.0
Iris setosa	Pre-clustering stage	1.0	1.0	1.0
	Post-clustering stage (2 clusters)	0.6	1.0	0.75
	Post-clustering stage (3 clusters)	0.6	1.0	0.75

These findings align with earlier work showing that K-means struggles with outliers because its centroids are mean-based, and extreme observations can pull them off-center [22], [23]. Past studies have tried to get around this by removing outliers beforehand or tinkering with the algorithm itself so anomalous points carry less weight during cluster formation [24], [25]. What the present study offers is something simpler, keeping K-means untouched and running MAD ahead of it as a preprocessing step. Under the conditions tested, this detected every injected outlier with 100% accuracy and no false positives. The practical takeaway is that when anomalies are flagged before clustering, K-means can form its clusters on data that hasn't been distorted by those outliers. There is no need to modify the clustering algorithm at all; the conventional K-means procedure stays in place, and MAD simply does the cleaning first.

That said, the scope here was limited; the findings rest on a synthetic dataset and an augmented Iris setosa dataset with artificially introduced anomalies. Whether MAD-before-clustering holds up in messier, real-world conditions remains an open question, so future work on larger datasets with naturally occurring

outliers and with clustering algorithms beyond K-means would go a long way toward confirming how generalizable this approach really is.

4. CONCLUSION

This study showed that the MAD algorithm performs better when it is allowed to run before clustering. Under the conditions tested in the study, MAD flagged every injected outlier with 100% accuracy and produced no false positives when applied up front. The results changed after K-means clustering, false positives began to appear, and in some cases, genuine anomalies were missed altogether. What this suggests is that treating MAD as a preprocessing step does two things at once: it makes anomaly detection more reliable and prevents stray observations from pulling K-means's centroids toward them during cluster formation.

There is a practical lesson here for anyone using mean-based clustering: a quick robust statistical screen on the data beforehand can spare you the trouble of outliers quietly steering your clusters. The study also adds something to the broader conversation about how robust statistical methods and clustering techniques interact. Still, the study would not oversell the results. The evidence rests on a synthetic dataset and the Iris setosa data, both with artificially planted anomalies, and only K-means was considered. Real datasets are messier than that, so it is hard to say with confidence how MAD would behave in more complex settings.

That is exactly where future work should point. Testing MAD on larger, real-world datasets with naturally occurring outliers and pairing it with other algorithms such as DBSCAN, hierarchical clustering, and Gaussian mixture models would tell us a lot more. It would also be worth comparing MAD head-to-head with other robust anomaly-detection techniques to see whether its edge holds up across different methods, datasets, and clustering environments, or fades as conditions change.

FUNDING INFORMATION

The funding for the publication of this manuscript is from Tarlac State University

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Alvincent E. Danganan	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓

C : **C**onceptualization

M : **M**ethodology

So : **S**oftware

Va : **V**alidation

Fo : **F**ormal analysis

I : **I**nterpretation

R : **R**esources

D : **D**ata Curation

O : **O**riginal Draft

E : **E**xperimentation

Vi : **V**isualization

Su : **S**upervision

P : **P**roject administration

Fu : **F**unding acquisition

CONFLICT OF INTEREST STATEMENT

The authors declare no conflict of interest and have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

DATA AVAILABILITY

The data used for the testing and experimentation is available at <https://archive.ics.uci.edu> repository.

REFERENCES

- [1] K. Neha and M. Y. Reddy, "A study on applications of data mining," *International Journal of Scientific and Technology Research*, vol. 9, no. 2, pp. 3385–3388, 2020.
- [2] C. Sirichanya and K. Kraisak, "Semantic data mining in the information age: a systematic review," *International Journal of Intelligent Systems*, vol. 36, no. 8, pp. 3880–3916, Aug. 2021, doi: 10.1002/int.22443.
- [3] M. Wang, T. Yao, and G. I. Allen, "Supervised convex clustering," *Biometrics*, vol. 79, no. 4, pp. 3846–3858, Mar. 2023, doi: 10.1111/biom.12500.

- 10.1111/biom.13860.
- [4] S. M. F. D. S. Mustapha, "High-dimensional data analysis using parameter free algorithm data point positioning analysis," *Applied Sciences*, vol. 14, no. 10, p. 4231, May 2024, doi: 10.3390/app14104231.
 - [5] K. Tabianan, S. Velu, and V. Ravi, "K-means clustering approach for intelligent customer segmentation using customer purchase behavior data," *Sustainability*, vol. 14, no. 12, p. 7243, Jun. 2022, doi: 10.3390/su14127243.
 - [6] F. Guevara-Viejó, J. D. Valenzuela-Cobos, P. Vicente-Galindo, and P. Galindo-Villardón, "Application of k-means clustering algorithm to commercial parameters of pleurotus spp. cultivated on representative agricultural wastes from province of guayas," *Journal of Fungi*, vol. 7, no. 7, p. 537, Jul. 2021, doi: 10.3390/jof7070537.
 - [7] W. Azizah, A. I. Purnamasari, A. Bahtiar, and Kaslani, "Clustering analysis of administrative service types using k-means (study case: village bojongsalam)," *Journal of Artificial Intelligence and Engineering Applications (JAIEA)*, vol. 4, no. 2, pp. 1303–1310, Feb. 2025, doi: 10.59934/jaiea.v4i2.867.
 - [8] H. Liu, J. Li, Y. Wu, and Y. Fu, "Clustering With Outlier Removal," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 6, pp. 2369–2379, 1 June 2021, doi: 10.1109/TKDE.2019.2954317.
 - [9] G. Gan, "A k-means algorithm with automatic outlier detection," *Electronics*, vol. 14, no. 9, p. 1723, Apr. 2025, doi: 10.3390/electronics14091723.
 - [10] C. Chenghu and A. Thammano, "A self-adaptive weights for K-means classification algorithm," *HighTech and Innovation Journal*, vol. 6, no. 3, pp. 903–917, 2025, doi: 10.28991/HIJ-2025-06-03-010.
 - [11] V. Shorewala, "Notice of Removal: Anomaly Detection and Improvement of Clusters using Enhanced K-Means Algorithm," in *2021 5th International Conference on Computer, Communication and Signal Processing (ICCCSP)*, Chennai, India, 2021, pp. 115–121, doi: 10.1109/ICCCSP52374.2021.9465539.
 - [12] A. Nowak-Brzezińska and I. Gaibei, "How the outliers influence the quality of clustering?," *Entropy*, vol. 24, no. 7, p. 917, 2022, doi: 10.3390/e24070917.
 - [13] A. E. Danganan, A. M. Sison, and R. P. Medina, "An improved overlapping clustering algorithm to detect outlier," *Indonesian Journal of Electrical Engineering and Informatics (IJEI)*, vol. 6, no. 4, pp. 401–409, Dec. 2018, doi: 10.52549/ijeie.v6i4.499.
 - [14] S. Ochella, M. Shafiee, and C. Sansom, "Adopting machine learning and condition monitoring P-F curves in determining and prioritizing high-value assets for life extension," *Expert Systems with Applications*, vol. 176, 2021, doi: 10.1016/j.eswa.2021.114897.
 - [15] P. Strnad, L. Švarc, and P. Berka, "Synthetic dataset generator for anomaly detection in a university environment," *Intelligent Data Analysis*, vol. 27, no. 2, pp. 417–422, Mar. 2023, doi: 10.3233/IDA-216511.
 - [16] A. Iqbal, "Performance evaluation of machine learning algorithms for iris dataset classification," *Journal of Advances in Science and Technology*, vol. 21, no. 1, pp. 130–133, Mar. 2024, doi: 10.29070/8z0jnj19.
 - [17] A. Anderies, J. A. R. W. Tchin, P. H. Putro, Y. P. Darmawan, and A. A. S. Gunawan, "Prediction of heart disease uci dataset using machine learning algorithms," *Engineering, Mathematics and Computer Science (EMACS) Journal*, vol. 4, no. 3, pp. 87–93, 2022, doi: 10.21512/emacsjournal.v4i3.8683.
 - [18] P. Khatiwada, M. Subedi, A. Chatterjee, and M. W. Gerdes, "Automated human activity recognition by colliding bodies optimization-based optimal feature selection with recurrent neural network secure digital health infrastructure view project smart cardiovascular monitoring system view project automated human activ," *CoRR*, Oct. 2020, doi: 10.20944/preprints202010.0367.v1.
 - [19] A. Unwin and K. Kleinman, "The iris data set: in search of the source of virginica," *Significance*, vol. 18, no. 6, pp. 26–29, Nov. 2021, doi: 10.1111/1740-9713.01589.
 - [20] U. Gowtham, U. Ushasandhya, A. Bharadwaj, M. K. Reddy, and D. Prathyusha, "Iris dataset classification using 3 machine learning algos," *International Journal of Scientific Research and Engineering Development*, vol. 8, no. 2, pp. 1350–1357, 2025, doi: 10.5281/zenodo.15241079.
 - [21] L. Dicks and D. J. Wales, "Evolution of K-means solution landscapes with the addition of dataset outliers and a robust clustering comparison measure for their analysis," *arXiv*, 2023, doi: 10.48550/arXiv.2306.14346.
 - [22] J. Liao, X. Wu, Y. Wu, and J. Shu, "K-nndp: k-means algorithm based on nearest neighbor density peak optimization and outlier removal," *Knowledge-Based Systems*, vol. 294, p. 111742, Jun. 2024, doi: 10.1016/j.knosys.2024.111742.
 - [23] S. D. Veeramachaneni and R. Garimella, "Constrained centroid clustering: A novel approach for compact and structured partitioning," *arXiv*, 2025, doi: 10.48550/arXiv.2508.12758.
 - [24] N. H. M. M. Shrifan, M. F. Akbar, and N. A. M. Isa, "An adaptive outlier removal aided k-means clustering algorithm," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 8, pp. 6365–6376, Sep. 2022, doi: 10.1016/j.jksuci.2021.07.003.
 - [25] X. Chen, C. Park, X. Gao, and B. Kim, "Robust Model Design by Comparative Evaluation of Clustering Algorithms," *IEEE Access*, vol. 11, pp. 88135–88151, 2023, doi: 10.1109/ACCESS.2023.3306023.

BIOGRAPHY OF AUTHOR



Dr. Alvincent E. Danganan    has been a faculty member of Tarlac State University College of Computer Studies, Philippines since 2003. He served as the chairperson of the Department of Information Systems from 2013 to 2016. He was also designated as the college extension service chairperson for the same period and has conducted projects and presentations which have been awarded at the institutional level. He also served as the Chairperson of the Department of Computer Science from 2019–2020. Currently, he is the dean of Tarlac State University College of Computer Studies. His research interest includes data mining, machine learning, and artificial intelligence. He has authored publications on data mining in Scopus-indexed journals. He can be contacted at email: avdanganan@tsu.edu.ph.