Performance and Accuracy Trade-Off Analysis of Techniques for Anomaly Detection in IoT Sensors

Paulo Silas Severo de Souza*, Wagner dos Santos Marques*,
Fábio Diniz Rossi*, Guilherme da Cunha Rodrigues† and Rodrigo N. Calheiros‡
*Federal Institute of Education, Science, and Technology Farroupilha (IFFAR)
Alegrete, Brazil
Email: paulo.souza@email.com
†Federal Institute of Education, Science, and Technology Sul-Riograndense (IFSUL)
Charqueadas, Brazil
‡School of Computing, Engineering and Mathematics
Western Sydney University, Australia

***Abstract*—IoT environments are typically composed of hundreds of geographically distributed sensors. Usually, these sensors are not physically protected from unauthorized access, which makes them vulnerable to exploitation where they can be manipulated to send incorrect data. The identification of such compromised sensors can be helpful in the process of exclusion or verification by administrators. To perform the detection of anomalous sensors, several algorithms can be used. However, based on the algorithm used, this evaluation may be delayed or can be inaccurate. Therefore, to detect sensors with different behavior compared to others, we evaluated the trade-off between performance and accuracy of different anomalies detection algorithms. The results showed that Mahalanobis Distance could improve the trade-off between detecting multiple anomalous sensors at execution time and accuracy to avoid false-positives.**

I. Introduction

The popularization and low-cost of electronic devices such as sensors and actuators and networking technologies, along with the rise of the large-scale environment such as clouds enable the storage and analysis of a large amount of data in real time. This set of capabilities drove the emergence of the Internet of Things (IoT).

IoT [1] refers to implementation of machine-to-machine communication (M2M), and this paradigm is supported at the infrastructure level by a dynamic network infrastructure with self-configuration capabilities at execution time based on interoperable communication standards. Moreover, "things" have physical and virtual identities, attributes, and functions; are integrated within a network; and often communicating with data users and the environment. In most cases, sensors are seen as "things" dispersed in the environment, which send data via the network that are received and analyzed in real time.

There are several applications of IoT that can be critical and should be closelly monitored and verified, such as eHealth sensors that monitor vital signs of patients and administer doses of medicament on demand [2]; sensors that measure soil moisture and control soil irrigation in agricultural applications [3]; sensors that warn about unauthorized presence of personnel in restricted areas and trigger invasion alarms [4]. Any of these applications can be potentially harmful if one or more sensors are compromised, and this problem becomes harder to manage as the number of sensors increases to be managed increase.

In this context, security is an imperative subject on the Internet of Things scenario. Nonetheless, ensuring the security of sensors and the data the generate is not a trivial task since IoT sensors are usually connected to untrusted networks. Hence, several studies have been focused on the development of Intrusion Detection Systems (IDS), which analyzes the behavior of attacks, to avoid similar future attacks. Most IDS can only detect known attacks, giving space to anomaly detection techniques, which have been used together to IDS to improve the security of IoT devices [5] [6].

To address this limitation, several studies have used machine learning algorithms to classify, quantify, and assess IoT sensors that search for anomalies [7] [8] [9]. They consist of clustering algorithms that aim to partition observations in groups where each observation belongs to the nearest group average. However, some of these algorithms have limitations concerning execution time or accuracy when using large amounts of data.

Our proposal is to use machine learning algorithms [10] to analyze security issues in data sent by sensors in an IoT environment. Previous work already adopted similar algorithms to address such matters, as in Bovet et.al and Shahriar et. al [11] [12]. However, most of them use machine learning algorithms on static data series. We propose evaluating a time series, wherein freshly generated data is sent in time intervals and analyzed in real time. Furthermore, no previous studies have investigated the trade-off between performance and accuracy for anomaly detection algorithms on IoT sensor data with the intention of showing which option is the best to improve such trade-off.

The applied scenario of this study is the analysis of data generated by sensors allocated to a grain storage. Five widely-used anomaly detection algorithms are evaluated. They search for anomalous values for the monitored metrics, and they are evaluated in terms of time to perform the analysis and the accuracy of the result in terms of false-positives.

This paper is organized as follows: Section 2 presents a

978-1-5090-5124-3/17/$31.00 ©2017 IEEE

Authorized licensed use limited to: Pontificia Universidade Catolica do Rio Grande do Sul (PUC/RS). Downloaded on December 25,2023 at 00:38:17 UTC from IEEE Xplore. Restrictions apply.

background about Internet of Things. In Section 3, anomaly detection algorithms are listed and discussed. In Section 4 we present a test scenario, experiments, and discussion about the results. Section 5 contains conclusions and future work.

II. Internet of Things

The Internet evolved to be used is different contexts, as its capacity grew. Early uses of the Internet were in the form with a huge computer network with little attention to its users and applications. After that, it the raise of Web 2.0, the focus was on the people using the network. Recently, the focus started to change to support a network of interconnected things, where "things" can be computers or absolutely everything that is networked with the rest of the "things" in the world. The increased accessibility, the reduction in cost of network transmission media, and the relative ease of connection of devices has driven a plenty of organizations such as governments, businesses, and individuals to collect data from IoT devices.

Figure 1 shows a typical Internet of Things ecosystem. In Figure 1, we highlight the IoT environment which has three main components, namely, sensors, Internet and, analysis. Any object that can send data over the network can be considered a sensor. Due to the exponential growth in the number of such sensors, there is a significant amount of data traveling via wireless networks, and it requires suitable types of networks and offloading techniques to support such a demand without causing overhead on network channels.

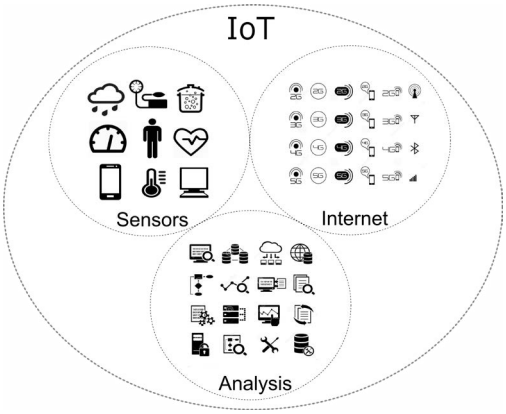

Figure 1. Interoperability among IoT components.

The analysis of data obtained from sensors is used to monitor and manage both environments and situations. Moreover, it generates runtime responses aiming to modify an environment or situation in order to adapt it based on the application goal, for example.

Currently, cloud services called AaaS (Analytics-as-a-Service) [13] can handle a large volume of data due to the elasticity capabilities, and generate responses quickly. However, there are several limitations inherent to the Internet of Things due to overexposure of the sensors and data, ranging from privacy issues to the sensor data change with the intention to compromise the decision-making process on actuators on IoT environment.

IoT provides suitable solutions to multiple contexts, such as urban illumination, waste management, and eHealth. However, these solutions require robust security measures, such as anomaly detection techniques, to maintain its trust, since IoT sensors can be compromised to transmit altered data, which could bring serious problems (e.g., compromised eHealth sensors could send wrong patient's glucose level to doctors, leading wrong medication prescriptions). Consequently, various algorithms are applied in order to detect sensors that may have been compromised.

As these algorithms differ in terms of capability and computing demand, in the next section we review anomaly detection algorithms aiming to draw a coherent landscape in this significant area.

III. Anomaly detection algorithms

Through advances in the area of data science, which allows the creation of knowledge by the analysis of massive quantities of data, datasets have been used to analyze phenomena, trends, and even to make predictions related to several processes. In this context, it is important to recognize data points whose values deviate from the other observations in the sample.

Those abnormal instances are called outliers or anomalous instances and can represent errors during the data collection process or variations in the sample. Figure 2 shows an example of data analysis, where outliers were found and highlighted.

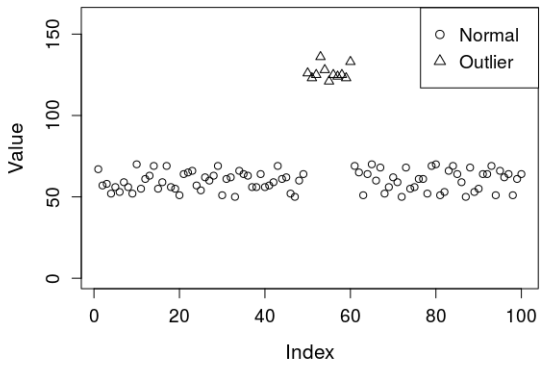

Figure 2. Dataset presenting some outliers, which were detected and highlighted.

According to the importance of anomaly detection on datasets, several techniques have been developed, using technologies related to several areas, such as machine learning, to provide efficient solutions based on outliers detection.

Those anomaly detection techniques can focus on different aspects, such as type of variables (multivariate or univariate), dimensions (spatial, temporal, or spatiotemporal), and type of analysis (online, offline, or both). Regarding machine learning techniques, they can be organized into 3 groups concerning to the way it leads to the feedback in the analysis process:

- *Supervised learning techniques* that receives predefined classes from the user (i.e., a finite training set with the classification of each of its instances). Thus, those techniques can organize the next instances into the available classes taking into consideration the original classification. This kind of technique is

Authorized licensed use limited to: Pontificia Universidade Catolica do Rio Grande do Sul (PUC/RS). Downloaded on December 25,2023 at 00:38:17 UTC from IEEE Xplore. Restrictions apply.

recommended to scenarios in which the user knows how to classify the data correctly.

- *Reinforcement learning techniques*, in which the algorithm automatically classifies the data, and the user gives to it a feedback which is used by the algorithm to learn how to increase its accuracy. These techniques can be used when the user cannot previously organize the data, but can reward or punish the algorithm based on its actions.
- *Unsupervised learning techniques* that automatically classifies the data without the user influence, taking into consideration the similarity between the instances in order to organize it into classes or groups (also known as clusters). These techniques are useful in scenarios where users are not able to help the algorithm in the classification process.

In order to refine our analysis, we focused only on unsupervised learning techniques with support to multivariate data analysis. Among the existing proposals, we focus on the following:

- K-means: It is useful in several studies to detect anomalies, grouping anomalous data on specific clusters, separated from normal instances [14], [15].
- Isolation Forest: Basedon the principle that anomalies occur in small number and are "distant" from normal datapoints in the attributed space [16].
- K-means–: an enhanced version of K-means, with more robustness. Developed for anomaly detection on large datasets [17].
- LOF: Developed for detecting anomalies in large databases. However, it has considerable computational cost [18].
- Mahalanobis Distance: It is applied to detect anomalies in different contexts, such as hyperspectral imaging and LED packages [19], [20].

In the next subsections, we highlight each of the above anomaly detection techniques.

A. K-means

K-means [21] is a non-supervised classification algorithm that performs analyses and comparisons among numerical values present in a data series. Such algorithm allows classification into clusters, which consist of data classes with similar characteristics. The number of classes that the algorithm can find out over a pre-classification dataset consist of k.

To generate the classes and classify the occurrences of similar data within each class, the algorithm makes a comparison between each value and another instance through the Euclidean distance. The way and the cost to calculate this distance depends on the number of attributes provided by the data. As the algorithm iterates, the value of each centroid is refined by averaging the values of each attribute from each event that belongs to this centroid. Thus, the algorithm generates k centroid and places the datapoints according to its distance from the centroid.

B. Isolation Forest

Isolation Forest [16] is a machine learning technique that allows an analysis of linear time complexity and low memory usage. This means that in the worst case, the time follows the growth of data to be processed, and low memory usage also impacts indirectly on performance. In our context, isolating means separating an instance of the total of instances. If we consider that an anomaly consists of an instance of different behavior from the others, this is more susceptible to isolation. Anomalies have two characteristics: they are the minority within a data set and they have a value or attribute very different from other normal instances. Based on this, Isolation Forest creates random trees sub-sets of data, and anomalies are isolated closer to the root, whereas normal points are separated deeper in the tree.

Anomalies are more susceptible to isolation and therefore have short path lengths. The number of partitions needed to separate one point is equal to the path length from the root node to the leaf node. The average size of paths of anomaly and normal instances converge when the number of trees increases. Since each partition is randomly generated, individual trees are produced with different sets of partitions. This shows that anomalies are having shorter run lengths than normal instances. Therefore, for dimensional problems that contain a large number of irrelevant attributes, Isolation Forest can achieve great performance in detecting certain types of anomalies.

C. K-Means–

K-means have been used in several studies to detect anomalies. Nonetheless, this technique can be considered insufficient to detect anomalies in some scenarios considering its outliers detection sensitivity. This factor can influence the final result negatively, since k-means can present some limitations, especially in large datasets, which include failing to detect all anomaly instances, presenting normal observations as anomalies, or even recognizing outliers as normal instances.

Hence, Chawla and Gionis [17] introduced a new anomaly detection method based on k-means, called k-means–. Since one of the most known limitations of k-means is its sensitivity to outliers (i.e., one outlier may cause significant changes in the mean and in the standard deviation of the sample), the authors designed such technique in order to define the number of clusters in a unified way, simultaneously forming clusters and tracking outliers aiming to avoid classification problems, such as false-positives and false-negatives.

Moreover, k-means– presents more robustness than k-means, which allows its usage on large datasets. In order to validate the efficiency of its approach, the authors applied it on a dataset with 5 decades of data related to hurricanes occurred in the Atlantic Ocean. The results showed that k-means– was more capable of detecting outliers in comparison to the classical nearest-neighbor approach [17].

D. LOF

The Local Outlier Factor (LOF) is an algorithm that analyzes the distance between a point and its nearest neighbors (that is called local density). The parameter k defines the

Authorized licensed use limited to: Pontificia Universidade Catolica do Rio Grande do Sul (PUC/RS). Downloaded on December 25,2023 at 00:38:17 UTC from IEEE Xplore. Restrictions apply.

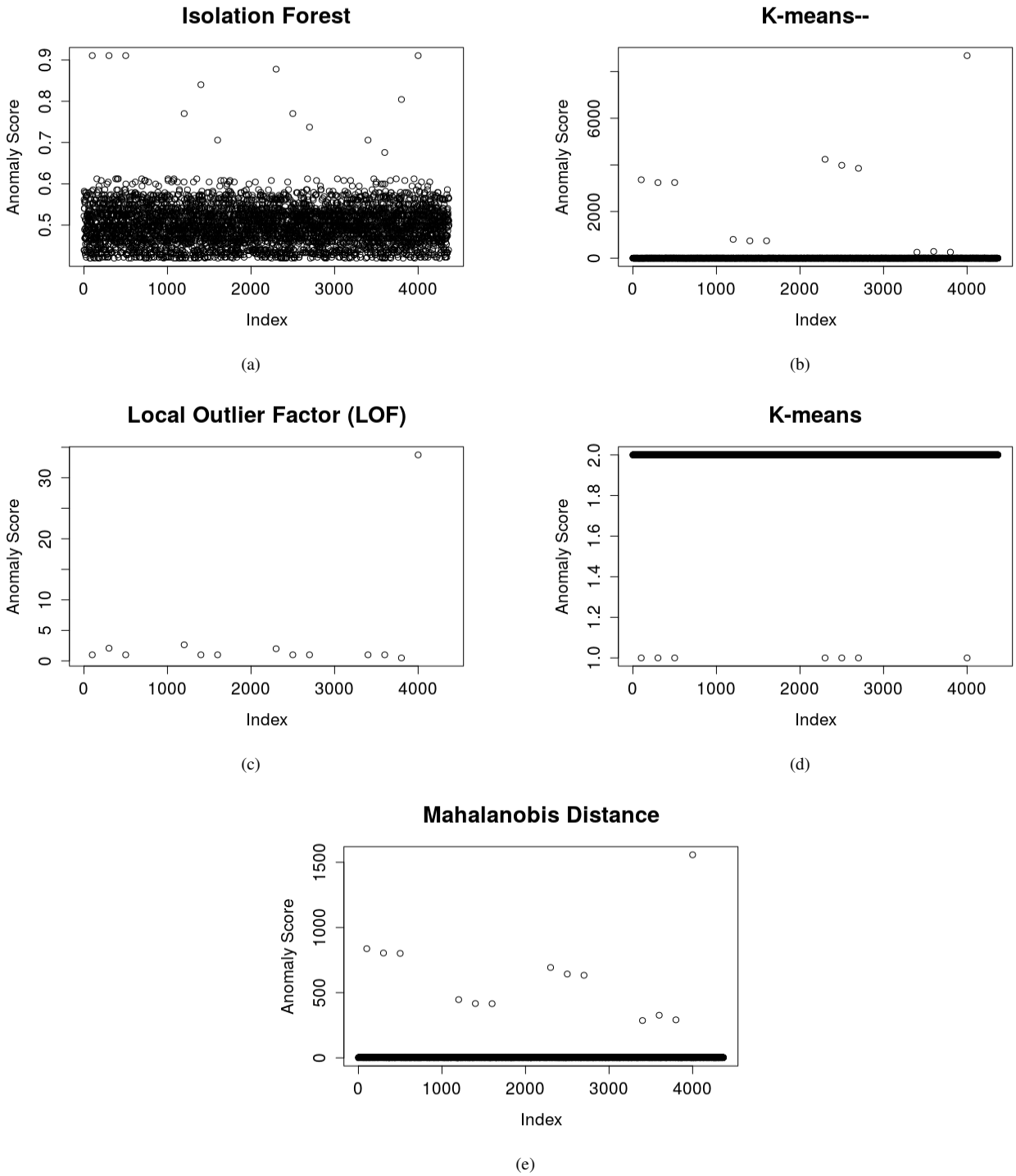

Figure 3. Accuracy results of the analyzed techniques. All algorithms, except K-means (whose results are shown on Figure 3(d)), were able to detect all anomalous instances.

number of neighbors the algorithm will consider to evaluate the density of an instance. Thus, the algorithm performance decreases as the value of *k* neighbors to be considered in the density calculation increases.

Furthermore, LOF can detect anomalies in an intelligent way by calculating the density dynamically, i.e., the algorithm analyzes the sample dynamically, so an instance that would be considered an outlier in a dense dataset can also be considered a normal instance within a sparse dataset. Thus, the algorithm compares the density of different points, identifying regions with similar densities on the sample, and separate the instances which whose density has a deviation in comparison with the k-th data points near from it, which are considered outliers [18].

Authorized licensed use limited to: Pontificia Universidade Catolica do Rio Grande do Sul (PUC/RS). Downloaded on December 25,2023 at 00:38:17 UTC from IEEE Xplore. Restrictions apply.

E. Mahalanobis Distance

The Mahalanobis Distance is a statistic metric created by Mahalanobis [22] that calculates the distance between a point and the centroid of a multivariate conjunct. Thus, it is possible to analyze if an observation belongs to a distribution or not. Differently from Euclidian distance, Mahalanobis Distance takes into account the covariance matrix (based on the known items of each class) in order to calculate the distance of the analyzed point from the rest of the sample population and classifies it into the group with the minimal distance.

In a multivariate analysis scenario, this measure does not take into account the distance between an attribute by time, but analyzes all variables at the same time, calculating the correlation between them. This is very relevant since a data point can be considered normal from the perspective of a single variable, but can constitute a multivariate outlier.

In this section, we presented and discussed the algorithms and techniques. In the next section, we perform experiments aiming to compare and evaluate them in the context of anomaly detection in IoT sensors.

IV. Experiments

In order to examine the performance and accuracy of the chosen anomaly detection techniques, we applied them to a dataset with 4 attributes (temperature, humidity, atmospheric pressure, and CO_2 level) from a grain storage, collected hourly during the first semester of 2016 (4368 measurements in total). These attributes were chosen because they are the most important for adequate storage of grains in silos. In those scenarios, control such variables is an important task to conserve the integrity of the grains (e.g., temperatures lower than 15°C can prevent the appearing of fungi and bacteria on the grains) [23]. Moreover, one of those variables can influence the others (e.g., humidity have a significant impact on the temperature variation) [24]. In such dataset, there were 2 types of anomalies:

- Isolated anomalies, that occurred in a single attribute (this type appeared 3 times in each attribute).
- Anomalies that occurred in all attributes at the same time (appeared one time during the measurements).

Regarding the implementation of the algorithms, in Isolation Forest we set it to fully deterministic mode. To define the number of clusters used by k-means and k-means–, we applied the silhouette method, which suggested the use of 2 clusters. In order to refine the results of k-means, we also used the parameters *nstart = 100* and *iter.max = 20*. Moreover, we implemented LOF using the parameter *k = 1*, which sets the kth-distance to be used to calculate the LOFs. The accuracy results of the algorithms are presented in Figure 3. The only algorithm that was not able to detect all anomalies was k-means (Figure 3 (d)), which, despite have found the generalized anomaly, only found 6 out of 9 isolated anomaly instances. All other algorithms were capable of detecting all the anomaly instances.

The results showed that the majority of the evaluated algorithms were able to detect univariate and multivariate outliers (e.i., abnormal values on one attribute and abnormal values on two or more attributes). In this sense, to define the best algorithm to our case, we also analyzed the performance of each one, since this can be determinant on the algorithm's efficiency in some scenarios.

Table I. Elapsed time by the algorithms on the analysis of the dataset.

Algorithm	Elapsed Time
Isolation Forest	0.513s
Mahalanobis Distance	0.003s
K-means	0.429s
K-means–	1.802s
Local Outlier Factor (LOF)	10.574s

Regarding performance, Mahalanobis Distance was the best, with 0.426s of difference from the second place (k-means). Those results can be a consequence of Mahalanobis capability to search outliers considering all variables simultaneously (considering that all anomalies have a considerable deviance of value from all other normal variables). Even k-means– also having an approach the allows it to analyze the dataset and classify the data within clusters simultaneously, this algorithm was not able to finish the analysis process with the same performance as Mahalanobis Distance.

The worst algorithm in terms of performance was LOF, which took 10.574s to analyze the dataset. The LOF results may have been caused by the method characteristics, where the resulting values are difficult to interpret because there is no clear rule to define an outlier. The information regarding the algorithms performance is presented on Table I.

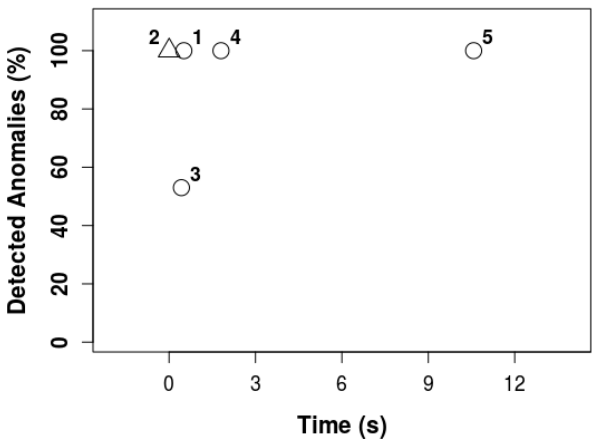

Figure 4. Performance and accuracy trade-off. Although Isolation Forest (1), K-means– (4), and LOF (5) have obtained the maximum accuracy results, Mahalanobis Distance (2) was the best algorithm to improve such trade-off. K-means (3) showed the worst accuracy results.

Therefore, by comparing the algorithms regarding performance and accuracy, we conclude that, for our case study and dataset, Mahalanobis Distance presents the best trade-off between performance and accuracy when compared to the other algorithms. Such result can be seen in Figure 4.

Authorized licensed use limited to: Pontificia Universidade Catolica do Rio Grande do Sul (PUC/RS). Downloaded on December 25,2023 at 00:38:17 UTC from IEEE Xplore. Restrictions apply.

V. Conclusion and Future Work

The increased utilization of Internet-connected sensors in our daily lives has greatly facilitated our way of life. This is occurring not only in everyday matters, such as a simple check of the weather, but also on issues that affect the health, production, locomotion, and many others that affect us directly or indirectly. On health issues, we can now take advantage of wearable sensors, which can monitor vital signs, send alerts, and even administer medications. In crop production, sensors can control several important environmental metrics for the development of plants, such as temperature and light, acting on actuators that can regulate these metrics to such an environment. In either case, compromised sensors can send wrong values, impacting in decision-making processes of the Internet of Things ecosystem.

Therefore, algorithms that can analyze the information sent by a set of sensors in real time and accurately become relevant. The main problem of these algorithms is the trade-off between performance and accuracy. In this context, algorithms that are quick to detect anomalies may not always provide the best accuracy. Algorithms that have an excellent accuracy may be too slow to detect anomalies in real time. Therefore, this paper evaluates the trade-off between performance and accuracy of such algorithms, through data evaluation coming from real sensors in a grain storage scenario.

Results showed that Mahalanobis Distance was the best algorithm in both aspects evaluated, namely, performance and accuracy, being able to detect all outliers in a fraction of second, what demonstrates its efficiency in multivariate analysis. As a future work, we intend to isolate anomalous sensor by consensus algorithm so that these do not influence the decisions of the actuators that maintain a satisfactory environment for the grain storage.

References

[1] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," *Future Gener. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, Sep. 2013.

[2] V. Chandel, A. Sinharay, N. Ahmed, and A. Ghose, "Exploiting imu sensors for iot enabled health monitoring," in *Proceedings of the First Workshop on IoT-enabled Healthcare and Wellness Technologies and Systems*, ser. IoT of Health '16. New York, NY, USA: ACM, 2016, pp. 21–22.

[3] T.-C. Lu, L.-R. Huang, Y. Lee, K.-J. Tsai, Y.-T. Liao, N.-C. Cheng, Y.-H. Chu, Y.-H. Tsai, F.-C. Chen, and T.-C. Chiueh, "Invited - wireless sensor nodes for environmental monitoring in internet of things," in *Proceedings of the 53rd Annual Design Automation Conference*, ser. DAC '16. New York, NY, USA: ACM, 2016, pp. 3:1–3:5.

[4] J. Obermaier and M. Hutle, "Analyzing the security and privacy of cloud-based video surveillance systems," in *Proceedings of the 2Nd ACM International Workshop on IoT Privacy, Trust, and Security*, ser. IoTPTS '16. New York, NY, USA: ACM, 2016, pp. 22–28.

[5] H. Sedjelmaci, S. M. Senouci, and M. Al-Bahri, "A lightweight anomaly detection technique for low-resource iot devices: A game-theoretic methodology," in *2016 IEEE International Conference on Communications (ICC)*. IEEE, 2016, pp. 1–6.

[6] B. Arrignton, L. Barnett, R. Rufus, and A. Esterline, "Behavioral modeling intrusion detection system (bmids) using internet of things (iot) behavior-based anomaly detection via immunity-inspired algorithms," 2016.

[7] J. Takahashi, D. Shioiri, Y. Shida, Y. Kobana, R. Suzuki, Y. Kobayashi, N. Isoyama, G. Lopez, and Y. Tobe, "Clustering for road damage locations obtained by smartphone accelerometers," in *Proceedings of the Second International Conference on IoT in Urban Space*, ser. Urb-IoT '16. New York, NY, USA: ACM, 2016, pp. 89–91.

[8] K. Nishi, K. Tsubouchi, and M. Shimosaka, "Extracting land-use patterns using location data from smartphones," in *Proceedings of the First International Conference on IoT in Urban Space*, ser. URB-IOT '14. ICST, Brussels, Belgium, Belgium: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2014, pp. 38–43.

[9] J. De Melo Borges, T. Riedel, and M. Beigl, "Urban anomaly detection: A use-case for participatory infra-structure monitoring," in *Proceedings of the Second International Conference on IoT in Urban Space*, ser. Urb-IoT '16. New York, NY, USA: ACM, 2016, pp. 36–38.

[10] P. K. Manadhata, "Machine learning for enterprise security," in *Proceedings of the 8th ACM Workshop on Artificial Intelligence and Security*, ser. AISec '15. New York, NY, USA: ACM, 2015, pp. 1–1.

[11] G. Bovet, A. Ridi, and J. Hennebert, "Virtual things for machine learning applications," in *Proceedings of the 5th International Workshop on Web of Things*, ser. WoT '14. New York, NY, USA: ACM, 2014, pp. 4–9.

[12] M. S. Shahriar and M. S. Rahman, "Urban sensing and smart home energy optimisations: A machine learning approach," in *Proceedings of the 2015 International Workshop on Internet of Things Towards Applications*, ser. IoT-App '15. New York, NY, USA: ACM, 2015, pp. 19–22.

[13] H. W. Wang, "Integrity verification of cloud-hosted data analytics computations," in *Proceedings of the 1st International Workshop on Cloud Intelligence*, ser. Cloud-I '12. New York, NY, USA: ACM, 2012, pp. 5:1–5:4.

[14] D. M. Menon and N. Radhika, "Anomaly detection in smart grid traffic data for home area network," in *2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT)*. IEEE, 2016, pp. 1–4.

[15] H. K. Idrissi, Z. Kartit, A. Kartit, and M. El Marraki, "Ckmsa: an anomaly detection process based on k-means and simulated annealing algorithms," *International Review on Computers and Software (IRECOS)*, vol. 11, no. 1, pp. 42–48, 2016.

[16] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, ser. ICDM '08. Washington, DC, USA: IEEE Computer Society, 2008, pp. 413–422.

[17] S. Chawla and A. Gionis, "k-means-: A unified approach to clustering and outlier detection." in *SDM*. SIAM, 2013, pp. 189–197.

[18] L. Xu, Y.-R. Yeh, Y.-J. Lee, and J. Li, "A hierarchical framework using approximated local outlier factor for efficient anomaly detection," *Procedia Computer Science*, vol. 19, pp. 1174–1181, 2013.

[19] Y. Zhang, B. Du, L. Zhang, and S. Wang, "A low-rank and sparse matrix decomposition-based mahalanobis distance method for hyperspectral anomaly detection," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 3, pp. 1376–1389, 2016.

[20] J. Fan, C. Qian, X. Fan, G. Zhang, and M. Pecht, "In-situ monitoring and anomaly detection for led packages using a mahalanobis distance approach," in *Reliability Systems Engineering (ICRSE), 2015 First International Conference on*. IEEE, 2015, pp. 1–6.

[21] J. Hartigan, *Clustering Algorithms*. New York: John Wiley & Sons Inc., 1975.

[22] P. C. Mahalanobis, "On the generalized distance in statistics," *Proceedings of the National Institute of Sciences (Calcutta)*, vol. 2, pp. 49–55, 1936.

[23] D.-W. Sun and J. Woods, "Low temperature moisture transfer characteristics of wheat in thin layers," *Transactions of the ASAE*, vol. 37, no. 6, pp. 1919–1926, 1994.

[24] S. K. Abbouda, D. Chung, P. Seib, and A. Song, "Heat and mass transfer in stored milo. part i. heat transfer model," *Transactions of the ASAE*, vol. 35, no. 5, pp. 1569–1573, 1992.

Authorized licensed use limited to: Pontificia Universidade Catolica do Rio Grande do Sul (PUC/RS). Downloaded on December 25,2023 at 00:38:17 UTC from IEEE Xplore. Restrictions apply.