Ingénierie des Systèmes d'Information
Vol. 29, No. 4, August, 2024, pp. 1397-1403
Journal homepage: http://iieta.org/journals/isi

A Machine Learning Approach on Outlier Removal for Decision Tree Regression Method

Agus Sihabuddin*, Nur Rokhman, Erwin Eko Wahyudi

Department of Computer Science and Electronics, Gadjah Mada University, Yogyakarta 55281, Indonesia

Corresponding Author Email: a_sihabudin@ugm.ac.id

https://doi.org/10.18280/isi.290414

Received: 20 November 2023
Revised: 15 May 2024
Accepted: 2 June 2024
Available online: 21 August 2024

Keywords:
outlier removal, Isolation Forest, decision tree regression, supervised learning, machine learning

ABSTRACT

Outliers can occur in application areas, adversely affecting the prediction method's performance. Outliers can be removed by using robust statistical algorithms. However, statistical methods have limitations in capturing the outlier for high-dimensional data. Approaches using Machine Learning (ML) are offered as they develop rapidly due to their excellent interpretability and strong generalization capabilities. So, ML is popular in detecting or eliminating outliers to increase the accuracy of forecasting methods, such as Isolation Forest (IF), an unsupervised outlier detection strategy using a collective approach to calculate the isolation score for every data point. This research objective is to improve the prediction accuracy of the Decision Tree Regression (DTR) method by proposing an IF as an ML-based outlier removal method. The proposed method was tested by two Air Quality Index (AQI) dataset that contained outliers with Mean Absolute Error (MAE), R-Square, and Root Mean Square Error (RMSE) as the accuracy measurements. The results showed that the proposed method outperforms previous studies.

1. INTRODUCTION

Outliers are defined by objects that are few and diverge from the majority object [1, 2]. Outliers in data sets can occur due to systematic measurement errors and missing covariates [3]. In contrast to noise, defined as misclassification (class noise) or attribute error (attribute noise), outliers are a broader concept encompassing inconsistent data arising from natural population or process variation [4]. In other words, an outlier is a rare and unexpected occurrence that is very different from a regular occurrence [5].

Outliers in datasets can adversely affect machine learning method performance [6]. There are many approaches to detecting the presence of outliers, and each has its advantages. Two widely used approaches are descriptive statistics (Interquartile Range [7], Tukey's Method [8, 9], Z-Score [10], Studentized Residuals [11], Cook's Distance [12], Mahalanobis Distance [13], M Huber [14], Local Outlier Factor (LOF) [15-17], and Minimum Covariance Determinant (MCD) [17]); and Machine Learning (ML) clustering (k-Means [18], Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [19], hierarchical clustering [18], k-Means++ [20], DBSCAN++ [19], k-Medoids [21], One-Class SVM [15, 17], and Isolation Forest (IF)). These methods have limitations in detecting outliers in high-dimensional data that show complex patterns. The distribution fitting method also has several limitations in data partition [22]. Ideally, to ensure that the simulation provides the most accurate representation of expected reality, this distribution is intended to reflect the stochastic nature of the activity. However, these distributions are often chosen arbitrarily and are based on distribution classes usually found in the statistical literature, which may not align with the actual project characteristics [23].

However, ML algorithms can be used to overcome these limitations [22, 24]. One of the ML popular methods for removing outliers is the Isolation Forest [25-27]. The Isolation Forest can determine anomaly scores by collecting particular trees, the so-called isolation trees [28]. The advantage of this is its computational efficiency on high dimensional data [29]. This algorithm is used for anomaly detection and is characterized by its linear time complexity, showing superior detection capability on perceptual data [30].

This research objective is to improve the accuracy of the prediction of the Decision Tree Regression (DTR) method by integrating outlier removal using the Isolation Forest method. DTR is a widely used ML method for prediction [28]. DTR is like a classification tree with roots, nodes, and leaves. DTR is a robust ML algorithm that offers outstanding advantages such as transparency, simplicity, and versatility in handling different data types. However, there are also limitations associated with DTR, such as the risk of overfitting [29, 30]. Overfitting occurs when training data are too complex, noisy, incomplete, etc., [30, 31]. Overfitting due to outliers is often problematic in regression and classification models [32, 33].

This paper consists of the following parts. The first section introduces the outliers and methods for outlier detection. Isolation Forest is used as an alternative for outlier removal, and DTR is used as the regression method. Section 2 describes the methodology used: decision tree regression, Isolation Forest, and the proposed method. Then, section three presents the experimental results and discussion. Finally, Section 4 presents the paper's result, conclusion, and future work.

2. RESEARCH METHODS

This research uses a modified DTR supervised learning approach by integrating the Isolation Forest method as outlier removal in data pre-processing. In addition, this research focuses on improving prediction accuracy in supervised learning.

2.1 Decision tree regression

Decision Tree (DT) is used for both classification and regressive analyses [31, 32]. This method is advantageous when dealing with decision-related problems [31]. DT works by continuously dividing the input data at each branch and creating a prediction method at each part (node) based on the target value (output). This division results in a visual representation of a decision tree comprising branches and nodes. The first or internal node is the tree's root, with outward edges, while the others are called leaves. Each node within this tree framework executes a binary judgment that distinguishes one or more categories from the rest [33]. The Decision Tree Regression (DTR) is based on the DT method and creates a prediction method as a tree structure [34]. Figure 1 illustrates the DT structure. The characteristics of an item are analyzed in DTR, and a tree-shaped method is employed to make precise prognostications for forthcoming data and relevant continuous results [35-39].

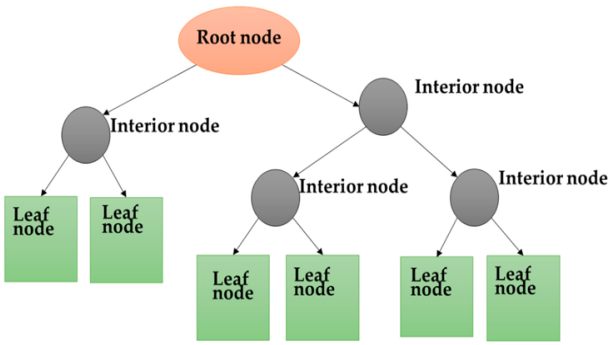

Figure 1. The decision tree structure [31]

In a regression problem, let $X = X_1, X_2, \ldots, X_{pn}$ be variables of predictor where pn is the predictor variable's total number. Let n and $Y = Y_1, Y_2, \ldots, Y_n$ be the number of observations and a target variable that takes continuous values. The vf is a feature variable and th is a value threshold [36]. Let t and $\gamma = (vf, th_t)$ be a node and candidate split, respectively.

$$Q_l(\gamma) = (x, y) |\ x_{vf} \leq th_t \tag{1}$$

Eq. (1) illustrates Q_1, that the decision tree's left branch is determined by dividing the data into potential split candidates.

$$Q_r(\gamma) = (x, y) |\ x_{vf} > th_t \tag{2}$$

Eq. (2) illustrates Q_r, denoted as the right side in the decision tree, is determined by dividing the data into γ potential splits. Furthermore, Eq. (2) can be defined as $Q_r(\gamma) = \frac{Q}{Q_1(\gamma)}$. $\bar{Y}_t$, representing the average of predicted value at the terminal nodes. Assume n to be the number of samples present at the current node.

$$\bar{Y}_t = \frac{1}{n} \sum_{i \in n} Y_i \tag{3}$$

Eq. (3) illustrates the computation of the average estimated value at the terminal nodes.

2.2 Isolation Forest

The Isolation Forest is a collection of binary trees, called Isolation Trees, designed to isolate data points [28]. The algorithm generates individual isolation trees that merge into an ensemble method, the Isolation Forest. The tree creation depends on the decisions determined by the data set format [40]. The Isolation Forest functions optimally with huge datasets as it has the time complexity of a linear function and low memory overhead [41]. So, the Isolation Forest technique is an unsupervised approach to outlier detection from a collective-based method, where an isolation score is calculated for every data point [42]. Briefly, the distribution is split multiple times through Isolation Forests at random domain values, and then the number of splits required to isolate each point is counted. Points that require less splitting are more likely to be outliers. The outlier score is determined by the number of necessary splits or output functions from numerous repetitions of this process [43]. To determine how an instance is unique compared to other cases based on their respective path lengths, it is essential to calculate the outlier score mathematically represented by Eq. (4), and Eq. (5) is used to evaluate the isolated trees average path length [40, 44, 45].

$$s(x, n) = 2^{-\frac{E(h(x))}{c(n)}} \tag{4}$$

$$c(n) = 2H(n-1) - \frac{2(n-1)}{n} \tag{5}$$

With an anomaly score of $s(x, n)$is; n is the dataset size; the average path length of the instance x over a tree collection $E(h(x))$ and an average path length of an unsuccessful search in a binary search tree given n samples $c(n)$. $H(n-1)$ is a harmonic number and can be approximated by $ln(n-1) + 0.5772156649$. Anomaly scores range from 0 to 1. Scores close to 1 indicate that an anomaly has been detected, scores below 0.5 indicate normal data and values close to 0.5 indicate no obvious anomaly.

2.3 Proposed method

The proposed method of the paper consists of several steps, which are explained below. The first step is data acquisition. At this stage, the data required for analysis is collected. The data preprocessing involves missing data imputation and feature selection.

The preprocessing method applied is similar to that described by Van et al. [29]. To handle the issue of missing data values, the K-Nearest Neighbours Imputer technique (KNNImputer) is applied as a method to fill in these empty values. The KNNImputer replaces missing values by evaluating target values from its nearest neighbors. In this approach, the missing values are filled using an approximation of the target value calculated through the average of the k-

nearest neighbor data values. The number of neighbors considered is determined by the *n_neighbors* parameter in the KNNImputer, which is in this study using the parameter value $n\_neighbors$ equals 3 as used in previous research [29], is utilized, meaning the algorithm will consider three nearest neighbor data points to fill in the missing values.

Then, the second step is outlier removal from the dataset. At this stage, the outliers are detected using Isolation Forest, the ML-based outlier detection method, and then removed from the dataset in the third step. This process aims to clean the data from deviant values that can affect prediction accuracy. In the fourth step, the data was divided into two groups: train data and test data. Training data is used to train the method and analyze patterns in the data, while test data is used to test the performance of the process that has been created.

The fifth step is to generate a DTR method. A decision tree regression model is constructed using the *sci-kit-learn* library, with the following parameters specified: *random_state* = 0, *max_depth* = 6, *max_leaf_nodes* = 100. Then, training data cleaned from outliers is used to train the DTR method. The method's performance is validated using K-fold cross-validation. Finally, a comparison of the results is performed based on Mean Absolute Error (MAE), R-Square (R^2), and Root Mean Square Error (RMSE) and compared to other previous research and methods.

This entire process forms a framework or methodology that can be used to analyze and improve calculation results. Algorithm 1 presents the proposed algorithm in this research.

Algorithm 1: Machine Learning-Based Outlier Removal DTR

Input: Data Set
Output: Decision Tree Regression Method
Process:

1. Input data set.
2. Outlier detection from the dataset using Isolation Forest ML-based outlier detection.
3. Remove outliers from the dataset.
4. Split data into train and test sets.
5. Generate a DTR method.
6. Train the DTR method.
7. Validate the method's performance using K-fold cross-validation.
8. Evaluate the method using MAE, R^2, and RMSE.

The devised method is implemented on the datasets [46]. In that paper, two datasets are utilized, namely the Air Quality Index (AQI) dataset provided by the Central Pollution Control Board (Dataset 1) and Open Government Data (Dataset 2) India.

The first dataset presents 29,531 daily samples recording the average AQI from January 2015 to June 2020. There are 12 significant environmental pollutant variable values, including PM10, PM2.5, Carbon Monoxide (CO), Ozone (O3), Nitrogen Dioxide (NO2), NOx, NO, Sulphur Dioxide (SO2), NH3, Benzene, Toluene, and Xylene. However, out of these 12 variables, only the most relevant ones will be selected through a feature selection stage to analyze AQI values.

Meanwhile, the Air Quality Index (AQI) Dataset 2 from Open Government Data contains 1,574 samples taken every hour in January 2020. This dataset is more focused, presenting AQI values and six other major pollutants, namely PM10, PM2.5, Ozone (O3), Nitrogen Dioxide (NO2), Sulphur Dioxide (SO2), and Carbon Monoxide (CO). These six variables are the focus of the second dataset for analysis and modeling related to air quality.

The features selected to predict the AQI value are performed by analyzing Pearson's Correlation Coefficient (PCC) between the target value and 12 pollutant variables, as shown in Table 1. The variables chosen as features to predict the AQI value must have a correlation value of at least 0.45 or higher. Therefore, the prediction analysis will consider only variables with a significant relationship with the AQI value. These features in Dataset 1 are used as features in Dataset 2 [29].

Table 1. Feature selection

	Pollutant Variable	Coefficient Correlation	Include in the Feature
1	PM2.5	0.65	Yes
2	PM10	0.80	Yes
3	Ozone	0.19	No
4	Nitrogen Dioxide	0.54	Yes
5	NOx	0.48	Yes
6	NO	0.45	Yes
7	Sulfur Dioxide	0.49	Yes
8	Carbon Monoxide	0.68	Yes
9	NH3	0.25	No
10	Benzene	0.04	No
11	Toluene	0.28	No
12	Xylene	0.16	No

In the implementation, the dataset is split using a 75:25 ratio, where 75% of the data is designated for method training, and the remaining 25% is used for assessing the method's performance. The algorithmic method employed in this research is implemented using Python 3.11.6, along with Pandas 2.1.1, NumPy 1.26.1, and Scikit-learn 1.3.1.

2.4 Evaluation method

The MAE is a metric used to evaluate the regression method. It calculates the mean of the predicted errors over all instances to give the final score and assesses the variation between the value of the predicted instance and the actual value [47, 48]. This is simple to measure and less sensitive to outlying values [49]. Eq. (6) is a description of the metrics used in this research work [47, 50].

$$MAE = \frac{\sum_{i=1}^{n} abs\left(y_i - \lambda\left(x_i\right)\right)}{n} \tag{6}$$

With y_i is the data ground truth value for x_i, $\lambda\ (x_i)$ is the predicted value for a data x_i, x_i is the number of data.

The R^2 or coefficient of determination is a statistical measure quantifying uncertainty from 0 to 1. A value of 1 shows a strong correlation between estimated and measured values [51]. R^2 is given by Eq. (7) [34].

$$R^2 = \frac{\sum_{i=1}^{n}\left(y_i - y_m\right)\left(\lambda\left(x_i\right) - \lambda\left(x_m\right)\right)}{\sqrt{\left(\sum_{i=1}^{n}\left(y_i - y_m\right)^2\right)\left(\sum_{i=1}^{n}\left(\lambda\left(x_i\right) - \lambda\left(x_m\right)\right)^2\right)}} \tag{7}$$

where, y_m and $\lambda\ (x_m)$ are the mean of the actual and predicted values.

RMSE is the root mean square error of the prediction versus the observation. RMSE is shown by Eq. (8) [34].

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{\left(y_i - \lambda(x_i)\right)^2}{n}} \quad (8)$$

3. RESULT AND DISCUSSION

This section explains the proposed method applied to two datasets started by outlier detection and removal using Isolation Forest and three other common comparison methods: MCD, LOF, and one-class SVM.

Table 2 shows the results of processing Dataset 1 and Dataset 2 using the proposed method with an Isolation Forest threshold of 0.1 and three other methods. There are 16 attributes, of which seven attributes were selected in Dataset 1. Dataset 1 has 29531 instances and 2953 outliers detected. Meanwhile, in Dataset 2, there are 14 attributes, of which eight attributes were selected. Dataset 2 has 1565 instances and 157 outliers detected. Thus, the proposed method can see outliers in both datasets using some of the attributes selected from the existing attributes. The number of outliers detected is also proportional to the size of each dataset.

The next stage, after outlier removal, is applied to the DTR to do the regression for the training and testing data. Table 3 shows the results of the analysis of two datasets, which were analyzed using four different methods mentioned previously. The three evaluation metrics are used to measure the accuracy performance of the methods used: MAE, R^2, and RMSE. The DTR (Lightweight ML) did not show the training data accuracy parameters because there was no data from the previous research [29].

Table 3 reveals that the proposed method displayed superior outcomes compared to previous research and three other standard outlier methods.

The proposed method training accuracy parameter outperforms all other methods for all MAE, R2, and RMSE for Datasets 1 and 2. The training accuracy parameter compared to the testing data showed almost no difference for all methods, especially for the proposed method. It means that the model is not overfitting and does not yet need the L1 or L2 regularisation.

In the Dataset 1 testing result, the proposed method showed the best performance, with the lowest MAE of 21.7104 and the lowest RMSE of 33.0481, although the R2 value was 0.8095, which is not the best number among the other methods.

The standard DTR method has the highest R^2, 0.8943, but its MAE and RMSE are higher than the proposed method. The Local Outlier Factor-DTR, Minimum Covariance Distance-DTR, and One Class SVM-DTR methods have lower R^2 values than standard DTR, indicating that integrating these outlier detection techniques only sometimes results in improvements in the context of this dataset. Moreover, their MAE and RMSE values are higher than the proposed method.

For Dataset 2, the proposed method again shows superior performance with a very low MAE of 1.679, a low RMSE of 4.6822, and a very high R^2 value of 0.9976. The proposed method indicates that it is very effective in handling this dataset. Standard DTR also shows excellent results with an R^2 of 0.9964, but its MAE and RMSE are higher than the proposed method. DTR variations integrating outlier detection techniques show reduced performance compared to standard DTR, with slightly lower values of R^2 and higher values of MAE and RMSE.

The model performance analysis concluded that the proposed method consistently performs better in both datasets than standard DTR and its variations that use outlier detection techniques, as presented in Table 3. This shows the importance of selecting and adapting the proper method for each dataset type to achieve optimal results.

This result indicates that outlier removal at leaf nodes in the learning procedure enhances conventional DTR's efficacy and resolves the dataset's outlier issue. Furthermore, the findings of this experimental analysis show that the proposed method can predict the value of the AQI dataset well, so the results of this research can improve the results of previous research conducted [29].

Figure 2 shows a comparative graph contrasting the actual and predicted values of AQI Dataset 1. The X-axis indicates the sample quantity, whereas the Y-axis indicates the genuine AQI values. The plot shows the predicted values in red plus signs and the actual values in yellow lines connected by dots.

Table 2. Regression dataset and outliers observation

Dataset	Attribute	Attribute Used	Instance	Outlier Removed			
				Isolation Forest	Minimum Covariance Determinant	Local Outlier Factor	One-Class SVM
Dataset 1	16	7	29531	2953	296	510	293
Dataset 2	14	8	1565	157	16	77	16

Table 3. The model performance result

Dataset	Method	Training			Testing		
		MAE	R^2	RMSE	MAE	R^2	RMSE
Dataset 1	DTR (Lightweight ML)				22.6105	0.8943	34.5898
	Local Outlier Factor-DTR	27.8009	0.8452	51.0231	28.5061	0.7991	57.2998
	Minimum Covariance Determinant-DTR	26.2130	0.8193	46.0683	26.4652	0.8209	46.2176
	One Class SVM-DTR	27.7972	0.8126	53.9411	27.9215	0.8170	56.7355
	Proposed method	22.2618	0.8016	44.0269	**21.7104**	0.8095	**33.0481**
Dataset 2	DTR (Lightweight ML)				3.2459	0.9964	5.9360
	Local Outlier Factor-DTR	4.7824	0.9919	8.5864	4.0539	0.9951	6.6878
	Minimum Covariance Determinant-DTR	5.2230	0.9909	9.2531	4.6875	0.9938	7.8248
	One Class SVM-DTR	4.6790	0.9929	8.1549	4.7677	0.994	7.5649
	Proposed method	2.9707	0.9945	6.7589	**1.679**	**0.9976**	**4.6822**

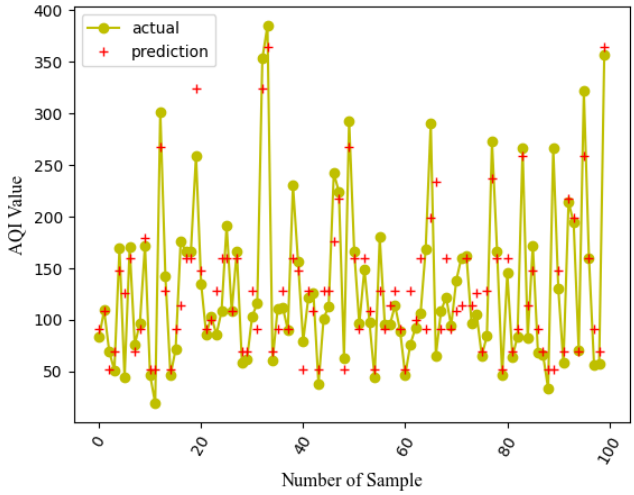

Figure 2. Actual and predicted value of AQI Dataset 1 with Isolation Forest outlier removal

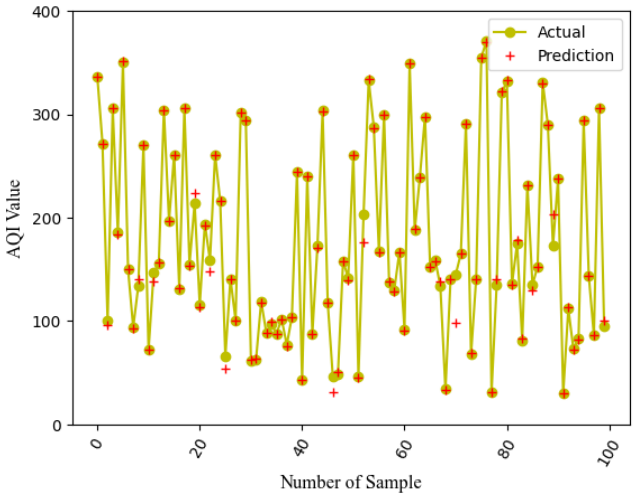

Figure 3. Actual and predicted value of AQI Dataset 2 with Isolation Forest outlier removal

Figure 3 is a comparison plot of predictions and actuals from AQI Dataset 2. In particular, the results obtained in Figure 3, compared to Figure 2, show a discrepancy. Precise updates are necessary to enhance the predicted values for AQI Dataset 1.

4. CONCLUSIONS

The DTR method integrated with Isolation Forest outlier removal in this research shows increased prediction accuracy. The DTR method with outlier removal at leaf nodes helps improve the performance of conventional DTR. Based on the discussion, outliers can affect the accuracy of regression performance. Therefore, detecting and handling outliers is essential to ensure accurate analysis results and effective machine-learning methods.

Future research could explore alternative outlier removal techniques, such as DBSCAN, DBSCAN++, kmeans, kmeans++, and other methods. The DTR can be modified using XGBOOST, which is renowned for its good performance, time complexity, and low memory requirements. Researchers must carefully consider the characteristics of the data set and the advantages and limitations of each method before choosing the most appropriate method for detecting and handling outliers.

ACKNOWLEDGMENT

This work was partially supported by the Department of Computer Science and Electronics, Universitas Gadjah Mada, under the Publication Funding Year 2024.

REFERENCES

[1] Du, X., Zuo, E., Chu, Z., He, Z., Yu, J. (2023). Fluctuation-based outlier detection. Scientific Reports, 13(1): 2408. https://doi.org/10.1038/s41598-023-29549-1

[2] Zhu, B., Jiao, J., Steinhardt, J. (2022). Generalized resilience and robust statistics. The Annals of Statistics, 50(4): 2256-2283. https://doi.org/10.1214/22-AOS2186

[3] Salgado, C.M., Azevedo, C., Proença, H., Vieira, S.M. (2016). Noise versus outliers. In: Secondary Analysis of Electronic Health Records. Springer, Cham, pp. 163-183. https://doi.org/10.1007/978-3-319-43742-2_14

[4] Alimohammadi, H., Chen, S.N. (2022). Performance evaluation of outlier detection techniques in production timeseries: A systematic review and meta-analysis. Expert Systems with Applications, 191: 116371. https://doi.org/10.1016/j.eswa.2021.116371

[5] Bijoy, M.B., Pebbeti, B.P., Manoj, A.S., Fathaah, S.A., Raut, A., Pournami, P.N., Jayaraj, P.B. (2023). Deep cleaner—A few shot image dataset cleaner using supervised contrastive learning. IEEE Access, 11: 18727-18738. https://doi.org/10.1109/ACCESS.2023.3247500

[6] Dobos, D., Nguyen, T.T., Dang, T., Wilson, A., Corbett, H., McCall, J., Stockton, P. (2023). A comparative study of anomaly detection methods for gross error detection problems. Computers & Chemical Engineering, 175: 108263. https://doi.org/10.1016/j.compchemeng.2023.108263

[7] Zhang, L., Zhong, H., Wei, B., Fan, J., Huang, J., Li, Y., Liu, W. (2022). Establishing reference values for peripheral blood lymphocyte subsets of healthy children in China using a single platform. Journal of Immunology Research, 2022(1): 5603566. https://doi.org/10.1155/2022/5603566

[8] Iftikhar Hussain, A.D.I.L., Zaman, A. (2020). Outliers detection in skewed distributions: Split sample skewness based boxplot. Economic Computation and Economic Cybernetics Studies and Research, (3): 279-296. https://doi.org/10.24818/18423264/54.3.20.17

[9] Li, Z., Xu, R., Luo, X., Cao, X., Sun, H. (2023). Short-term wind power prediction based on modal reconstruction and CNN-BiLSTM. Energy Reports, 9: 6449-6460. https://doi.org/10.1016/j.egyr.2023.06.005

[10] LIM, F., WONG, L., YAP, H., YOW, K. (2023). Identifying outlier subjects in bioavailability trials using generalized studentized residuals. Sains Malaysiana, 52(5): 1581-1593. http://doi.org/10.17576/jsm-2023-5205-19

[11] Dekker, V., Schweikert, K. (2021). A comparison of different data-driven procedures to determine the bunching window. Public Finance Review, 49(2): 262-

293. https://doi.org/10.1177/1091142121993055
[12] Sardashti, A., Nazari, J. (2023). A learning-based approach to fault detection and fault-tolerant control of permanent magnet DC motors. Journal of Engineering and Applied Science, 70(1): 109. https://doi.org/10.1186/s44147-023-00279-5
[13] Basalamah, S., Sihabuddin, A. (2024). A huber m-estimator algorithm and decision tree regression approach to improve the prediction performance of datasets with outlier. International Journal of Intelligent Engineering & Systems, 17(1): 1-9. https://doi.org/10.22266/ijies2024.0229.01
[14] Kumar, M., Saifi, Z., Krishnananda, S.D. (2023). Decoding the physiological response of plants to stress using deep learning for forecasting crop loss due to abiotic, biotic, and climatic variables. Scientific Reports, 13(1): 8598. https://doi.org/10.1038/s41598-023-35285-3
[15] Celdrán, A.H., Sánchez, P.M.S., von der Assen, J., Shushack, D., Gómez, A.L.P., Bovet, G., Pérez, G.M., Stiller, B. (2023). Behavioral fingerprinting to detect ransomware in resource-constrained devices. Computers & Security, 135: 103510. https://doi.org/10.1016/j.cose.2023.103510
[16] Karasmanoglou, A., Antonakakis, M., Zervakis, M. (2023). ECG-based semi-supervised anomaly detection for early detection and monitoring of epileptic seizures. International Journal of Environmental Research and Public Health, 20(6): 5000. https://doi.org/10.3390/ijerph20065000
[17] Ma, J., Zhang, H., Yang, S., Jiang, J., Li, G. (2023). An improved robust sparse convex clustering. Tsinghua Science and Technology, 28(6): 989-998. https://doi.org/10.26599/TST.2022.9010046
[18] Fuhnwi, G.S., Agbaje, J.O., Oshinubi, K., Peter, O.J. (2023). An empirical study on anomaly detection using density-based and representative-based clustering algorithms. Journal of the Nigerian Society of Physical Sciences, 5(2): 1364. https://doi.org/10.46481/jnsps.2023.1364
[19] Liu, X., Gong, W., Shang, L., Li, X., Gong, Z. (2023). Remote sensing image target detection and recognition based on yolov5. Remote Sensing, 15(18): 4459. https://doi.org/10.3390/rs15184459
[20] Mieczyńska, M., Czarnowski, I. (2021). Impact of distance measures on the performance of AIS data clustering. Computer Systems Science & Engineering, 36(1): 69-82. http://doi.org/10.32604/csse.2021.014327
[21] Buck, L., Schmidt, T., Feist, M., Schwarzfischer, P., Kube, D., Oefner, P.J., Zacharias, H.U., Altenbuchinger, M., Dettmer, K., Gronwald, W., Spang, R. (2023). Anomaly detection in mixed high-dimensional molecular data. Bioinformatics, 39(8): btad501. https://doi.org/10.1093/bioinformatics/btad501
[22] Chen, Z., Peng, Z., Zou, X., Sun, H. (2022). Deep learning based anomaly detection for muti-dimensional time series: A survey. In: Lu, W., Zhang, Y., Wen, W., Yan, H., Li, C. (eds) Cyber Security. CNCERT 2021. Communications in Computer and Information Science. Springer, Singapore, pp. 71-92. https://doi.org/10.1007/978-981-16-9229-1_5
[23] Vanhoucke, M., Batselier, J. (2019). Fitting activity distributions using human partitioning and statistical calibration. Computers & Industrial Engineering, 129: 126-135. https://doi.org/10.1016/j.cie.2019.01.037
[24] Buschjäger, S., Honysz, P.J., Morik, K. (2022). Randomized outlier detection with trees. International Journal of Data Science and Analytics, 13(2): 91-104. https://doi.org/10.1007/s41060-020-00238-w
[25] Barbariol, T., Susto, G.A. (2022). TiWS-iForest: Isolation Forest in weakly supervised and tiny ML scenarios. Information Sciences, 610: 126-143. https://doi.org/10.1016/j.ins.2022.07.129
[26] Heigl, M., Anand, K.A., Urmann, A., Fiala, D., Schramm, M., Hable, R. (2021). On the improvement of the Isolation Forest algorithm for outlier detection with streaming data. Electronics, 10(13): 1534. https://doi.org/10.3390/electronics10131534
[27] Chen, J., Zhang, J., Qian, R., Yuan, J., Ren, Y. (2023). An anomaly detection method for wireless sensor networks based on the improved Isolation Forest. Applied Sciences, 13(2): 702. https://doi.org/10.3390/app13020702
[28] Wang, X., Ping, W., Al-Shati, A.S. (2023). Numerical simulation of ozonation in hollow-fiber membranes for wastewater treatment. Engineering Applications of Artificial Intelligence, 123: 106380. https://doi.org/10.1016/j.engappai.2023.106380
[29] Van, N.H., Van Thanh, P., Tran, D.N., Tran, D.T. (2023). A new model of air quality prediction using lightweight machine learning. International Journal of Environmental Science and Technology, 20(3): 2983-2994. https://doi.org/10.1007/s13762-022-04185-w
[30] Ravi, T., Sathish Kumar, K., Dhanamjayulu, C., Khan, B. (2023). Utilization of stockwell transform and random forest algorithm for efficient detection and classification of power quality disturbances. Journal of Electrical and Computer Engineering, 2023(1): 6615662. https://doi.org/10.1155/2023/6615662
[31] Han, J. (2022). System optimization of talent life cycle management platform based on decision tree model. Journal of Mathematics, 2022(1): 2231112. https://doi.org/10.1155/2022/2231112
[32] Rath, K., Rügamer, D., Bischl, B., von Toussaint, U., Albert, C.G. (2023). Dependent state space Student-t processes for imputation and data augmentation in plasma diagnostics. Contributions to Plasma Physics, 63(5-6): e202200175. https://doi.org/10.1002/ctpp.202200175
[33] Sheng, S., Wang, X. (2023). Network traffic anomaly detection method based on chaotic neural network. Alexandria Engineering Journal, 77: 567-579. https://doi.org/10.1016/j.aej.2023.07.019
[34] Kassim, N.M., Santhiran, S., Alkahtani, A.A., Islam, M.A., Tiong, S.K., Mohd Yusof, M.Y., Amin, N. (2023). An adaptive decision tree regression modeling for the output power of large-scale solar (LSS) farm forecasting. Sustainability, 15(18): 13521. https://doi.org/10.3390/su151813521
[35] Rao, N.S.S.V.S., Thangaraj, S.J.J. (2023). Flight ticket prediction using random forest regressor compared with decision rree regressor. In 2023 Eighth International Conference on Science Technology Engineering and Mathematics (ICONSTEM), Chennai, India, pp. 1-5. https://doi.org/10.1109/ICONSTEM56934.2023.10142260
[36] Bouras, Y., Li, L. (2023). Prediction of high-temperature creep in concrete using supervised machine learning

algorithms. Construction and Building Materials, 400: 132828. https://doi.org/10.1016/j.conbuildmat.2023.132828
[37] Guo, S., Xu, J. (2021). CPRQ: Cost prediction for range queries in moving object databases. ISPRS International Journal of Geo-Information, 10(7): 468. https://doi.org/10.3390/ijgi10070468
[38] Ajay, R., Joel, R.S., Prakash, P.O. (2023). Analyzing and predicting the sales forecasting using modified random forest and decision tree algorithm. In 2023 8th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, pp. 1649-1654. https://doi.org/10.1109/ICCES57224.2023.10192723
[39] Pekel, E. (2020). Estimation of soil moisture using decision tree regression. Theoretical and Applied Climatology, 139(3): 1111-1119. https://doi.org/10.1007/s00704-019-03048-8
[40] Bao, J., Adcock, J., Li, S., Jiang, Y. (2023). Enhancing quality control of chip seal construction through machine learning-based analysis of surface macrotexture metrics. Lubricants, 11(9): 409. https://doi.org/10.3390/lubricants11090409
[41] Mohy-Eddine, M., Guezzaz, A., Benkirane, S., Azrour, M., Farhaoui, Y. (2023). An ensemble learning based intrusion detection model for industrial IoT security. Big Data Mining and Analytics, 6(3): 273-287. https://doi.org/10.26599/BDMA.2022.9020032
[42] Liu, Z., Zhao, X., Tian, Y., Tan, J. (2023). Development of compositional-based models for prediction of heavy crude oil viscosity: Application in reservoir simulations. Journal of Molecular Liquids, 389: 122918. https://doi.org/10.1016/j.molliq.2023.122918
[43] Gregg, J.T., Moore, J.H. (2023). STAR_outliers: A python package that separates univariate outliers from non-normal distributions. BioData Mining, 16(1): 25. https://doi.org/10.1186/s13040-023-00342-0
[44] Călin, A.D., Coroiu, A.M., Mureşan, H.B. (2023). Analysis of preprocessing techniques for missing data in the prediction of sunflower yield in response to the effects of climate change. Applied Sciences, 13(13): 7415. https://doi.org/10.3390/app13137415
[45] Zheng, H., Hu, Q., Yang, C., Mei, Q., Wang, P., Li, K. (2023). Identification of spoofing ships from automatic identification system data via trajectory segmentation and Isolation Forest. Journal of Marine Science and Engineering, 11(8): 1516. https://doi.org/10.3390/jmse11081516
[46] Central Pollution Control Board, Ministry of Environment, Forest and Climate Change Government of India. https://cpcb.nic.in/air-pollution/. Second link to download DATA1: https://www.kaggle.com/rohanrao/air-quality-data-in-india.
[47] Oloyede, A., Ozuomba, S., Asuquo, P., Olatomiwa, L., Longe, O.M. (2023). Data-driven techniques for temperature data prediction: Big data analytics approach. Environmental Monitoring and Assessment, 195(2): 343. https://doi.org/10.1007/s10661-023-10961-z
[48] Tipu, R.K., Suman, Batra, V. (2024). Enhancing prediction accuracy of workability and compressive strength of high-performance concrete through extended dataset and improved machine learning models. Asian Journal of Civil Engineering, 25(1): 197-218. https://doi.org/10.1007/s42107-023-00768-1
[49] Shakeera, S., Jyothi, V.B.N., Venkataraman, H. (2023). ML-based techniques for prediction of ocean currents for underwater vehicles. In 2023 11th International Symposium on Electronic Systems Devices and Computing (ESDC), Sri City, India, pp. 1-6. https://doi.org/10.1109/ESDC56251.2023.10149859
[50] Wudil, Y.S., Imam, A., Gondal, M.A., Ahmad, U.F., Al-Osta, M.A. (2023). Application of machine learning regressors in estimating the thermoelectric performance of Bi2Te3-based materials. Sensors and Actuators A: Physical, 351: 114193. https://doi.org/10.1016/j.sna.2023.114193
[51] Rahman, M.M., Shakeri, M., Tiong, S.K., Khatun, F., Amin, N., Pasupuleti, J., Hasan, M.K. (2021). Prospective methodologies in hybrid renewable energy systems for energy prediction using artificial neural networks. Sustainability, 13(4): 2393. https://doi.org/10.3390/su13042393