Received: April 19, 2023. Revised: August 27, 2023.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

A Huber M-Estimator Algorithm and Decision Tree Regression Approach to Improve the Prediction Performance of Datasets with Outlier

Salsabila Basalamah[1] Agus Sihabuddin[1]*

[1]*Department of Computer Science and Electronics, FMIPA, Gadjah Mada University, Yogyakarta, Indonesia*
* Corresponding author's Email: a_sihabudin@ugm.ac.id@mail.ugm.ac.id

Abstract: Outliers can cause the results of the analysis to be biased. Two approaches to dealing with existing outliers are removing the outliers or modifying the method used. Commonly used methods like machine learning (ML) often require enhanced robustness in predicting outliers. One such method is decision tree regression (DTR). However, the DTR method has limitations as it does not consider outliers and makes predictions at leaf nodes based on central values of the data, which can introduce biases into the results. One of the algorithm that retains outliers is the M-estimator from robust regression. This study proposes a modification of the M-estimator for DTR by using Huber weights on leaf nodes for DTR predictions. We used five regression datasets sourced from UCI. The results are that the dataset with outliers provides better predictions on the concrete dataset, superconductivity dataset, Boston dataset, and Airfoil dataset having the best mean absolute error (MAE) of 3.963, 9.140, 2.021, and 1.644, with QSAR fish toxicity the only exception, where has the best MAE of 0.522 for the outlier remover dataset.

Keywords: Decision tree regression, Outliers, Robust regression, M-estimator, Huber weights.

1. Introduction

An outlier is a case that describes the characteristics of a difference. The existence of outliers [1] in the data set is one of the problems which can cause the data variance to become more prominent, which can cause the analysis results to be biased.

Research on the problem of outliers has started to increase. Many studies have shown the existence of outliers interfering with the analysis results, such as machine learning (ML) analysis meathods [1-3]. ML algorithms tend not to study complex outlier data [4] and are sensitive to outliers [5]. This approach causes research to remove outlier data [1, 2]. Eliminating outliers in the dataset causes the possibility of losing important information because not all outlier data are errors [6].

Another approach is modifying the ML algorithm, which makes it unnecessary to remove outlier data. The ML method was modified [7] in support vector regression with one of the Robust estimates, namely least squares twin. In addition, there is also a modification [3] by adding a probabilistic tree algorithm to one of the ML algorithms, namely the random forest classification.

Another ML method that has not modified the algorithm to be able to make predictions with outlier datasets is the decision tree regression (DTR). The traditional DTR model [8] has weaknesses, such as a tendency to overfit small training datasets and suboptimal performance, as each leaf node relies on a single prediction result obtained from the mean [6] or median [9]. Thus, modifications made to the DT leaf node will provide better predictive results [6]. The DTR method in several studies has poor result accuracy [8] compared to other ML methods. However, some studies show the performance of DTR as the best method [10, 11].

Regression analysis can be used to increase the predictive performance of the DTR leaf nodes. Robust regression is one of the regression methods that resist outlier data [6]. Robust regression has several estimates, one of which is the M-estimator. The traditional M-estimator uses random weighting. However, random weighting can be changed using

Huber weighting, which is more accurate in predicting results. Still, one of the primary limitations of M-estimator is its computational complexity, which can lead to significantly longer computation times than other estimation methods [3].

This study aims to assess the potential improvement in prediction accuracy of DTR when handling data containing outliers by integrating an M-estimator for modeling at the leaf nodes. The advantage of the M-estimator method is its ability to handle data with outliers. The DTR algorithm can partition the data into smaller sub-data and minimize the computation of the large M-estimator [1]. Additionally, we replace the random weighting in the M-estimator with Huber weighting.

This article consists of the following parts. The first chapter briefly introduces the DTR and Regression Robust. Section 2 discusses the research methods. After that, we show the experimental results in section 3. Finally, we conclude and future work the whole paper in section 4.

2. Research methods

The approach method in this study uses supervised learning with modified DTR. Unlike previous research [2], this research does not use data pre-processing. This study conducted supervised learning for predictions.

2.1 Outlier detection

In detecting the existence of outliers data in the dataset used in this study, we can check outliers using Cook's distance method and the studentized residual method. These methods work by considering the residuals from the linear regression analysis, ordinal least square (OLS) estimation [12]. In the regression analysis, an estimated value or prediction results from the feature variable's value. The case of the relationship or influence of two or more feature variables is called multiple regression analysis [13], with the general equation as follows:

$$\hat{y}_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_k x_{ik} + \varepsilon_i, i = 1, \ldots, n \quad (1)$$

where $\hat{y}_i$: Prediction variable labels i ; β_0 : Constant; β_1 : Features parameters; β_k : features variable parameter ik; x_{ik} : features variable ik; x : features variable; and ε_i : Residual variable i.

The relationship with the outlier regression analysis can cause several things, such as large residuals from the model formed, the variance in the data becomes more considerable; the data interval has a wide range. Thus, linear regression, especially OLS, can be used to check outliers by using the residual results of predictions using OLS estimates.

Residual results using OLS estimation, it can be defined as the standardized residual of observation i as in the Eq. (2) [14].

$$I_{is} = \frac{e_i}{MSE\sqrt{(1-h_{ii})}} \quad (2)$$

where e_i is the residual of observation i, h_{ii} is the leverage value, and the MSE value is the mean square error. The results of the standardized residual will then be used to calculate the studentized residual (S_i) value, using Eq. (3).

$$S_i = I_{is}\left(\frac{n-k-2}{n-k-1-I_{is}^2}\right)^{1/2} \quad (3)$$

The calculation as Cook's distance can use in Eq. 4 [14].

$$D_i = \frac{1}{p} r_i^2 \frac{h_{ii}}{1-h_{ii}}, \quad \mathrm{i} = 1,2,3,..,\mathrm{k} \quad (4)$$

where D_i is the product of observation i between the standardized residual r_i , p is the number of variables and the leverage value h_{ii}.

Studentised residual evaluates based on studentised scores, where if the value of studentised is more than 2 or less than -2, then the data is said to be an outlier [15]. In Cook's distance, consider the D value [16]. If the distance value is more than 4 divided by the number of observations, it is the outlier. A data is said to be an outlier if the value is $D_i > \frac{4}{k}$, where k is the number of observations.

2.2 Huber M-estimator

Choosing a robust method for outliers in a data set containing outlier values is essential to get a robust model against outliers [17]. Estimates on robust regression provide a valid prediction for most data [15].

Algorithm 1 uses the objective function to find the weighting function in robust regression. The weighting functions commonly used in robust regression is Huber [18]. One of the robust regression estimates is the M estimate. The smaller the c value in the M-estimator, the more excellent the resistance of the prediction to outliers [15].

The efficiency and breakdown point is an internal terms in the robust regression. Efficiency is to describe how good a robust estimate is. Meanwhile, the breakdown point is the minor fraction or percentage of the data, that is, outliers or data

Algorithm 1. Huber M-estimator prediction

Algorithm: Huber M-estimator prediction
Input: Training data
Output: Model Huber M-estimator prediction
Proses:
1. Input labels (Y) and features (X)
2. Changing the value of the label in vector form
3. Changing feature values in matrix form
4. Transpose the X-matrix
5. Calculating intercept values with OLS estimates
6. The initial predictions with OLS estimator
7. Calculate the error from Step 6
8. Calculating the estimated value of a robust scale (constant value)
9. Calculate the weighted error value
10. Huber weighting (in a matrix)
11. Calculating the intercept value with the M-estimator
12. Calculating predictions of the M-estimator
13. Calculate the error from Step 12
14. Repeat Steps 8-11 until the value of the error metric E converges for the specified number of iterations
15. Return the model to mathematical equations

grouped at the far end of the distribution that the estimator can tolerate without producing biased results [15]. So, the higher an estimate's efficiency and breakdown point, the more resistant a model is in analyzing data containing outliers [19].

The M-estimator is also known as a weighted OLS estimator because in the M-estimator, the used predictions of the OLS estimator as initial predictions. Then this predicted value would get an error and be weighted to get a better predictive value [15]. Furthermore, the error from this prediction uses to predict the M-estimator. The S value in Algorithm 1 has a constant value. In the fourteenth step, stop iterations until the number of convergent errors or reach the maximum iteration limit.

In the M-estimator, additional weighting can be done using Huber weighting. The value of c = 1.345 at Huber provides substantial resistance to outliers and produces a relative efficiency of approximately 95% [15]. This M-estimator makes M-estimator, with Huber weighting, able to make M-estimator have a relative efficiency of roughly 95%.

2.3 Decision tree regression

The DTR is a method with a tree structure included in Supervised Learning. This method can work for classification and regression. The DTR

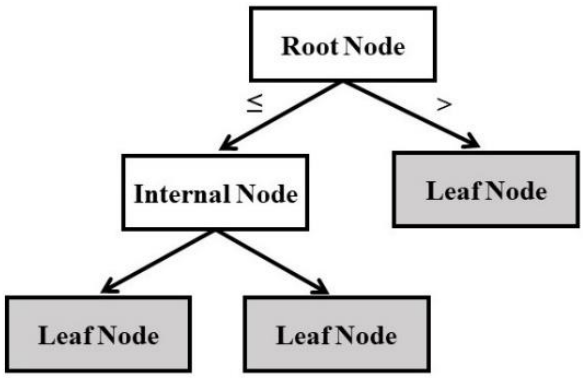

Figure. 1 Decision tree regression structure

divides the data several times according to specific feature threshold values. The process of dividing this data is done to break the data into smaller and more homogeneous label values. Different subsets of the data set of through splitting, each instance belonging to one subset.

Data division is done based on the value of the criteria used. We use the error criteria to build a tree. The error value also determines the best threshold value and the best feature with the smallest error value. The process in the decision tree begins by creating a root node. The last subgroup is the terminal node or leaf node, and the intermediate subset is the internal node or split node [20]. The process at the root node determines the feature that most influences the label to serve as feature splitting. The best threshold values are also selected from each feature to divide the data on that feature. This step also applies to internal nodes.

Various algorithms can use to grow trees and to make leaf node predictions. Determining when the internal node should have broken down into leaf nodes can be done by measuring the error. If the error in the internal node exceeds the leaf node, then additional leaf nodes can do. However, if it's the other way around, the internal node is a leaf node. Several data concentration measures can use to predict the results at each leaf node, namely the mean, median, and so on from the training data at each leaf node.

2.4 Modification decision tree regression

This study proposes modifying the method (Fig. 2) to deal with datasets with outliers. This modification involves an M-estimator with Huber weights. The change focuses on the model building at each DTR leaf node. As in Fig. 2, the leaf node will have a predicted value obtained from the mathematical model. DTR generally uses centralised data for the predicted value at each leaf node.

In building a tree (Algorithm 2), criteria are needed to determine the branching nodes. We use the

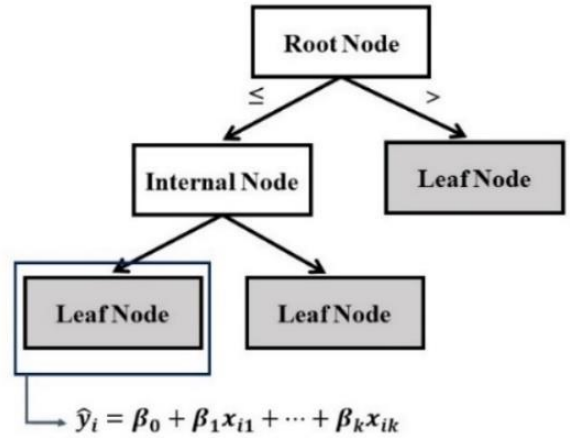

Figure. 2 DTR with Huber M-estimator

Algorithm 2. DTR with Huber M-estimator

Algorithm: Decision Tree Regression with Huber M-Estimator

Input: Training data; criteria Mean Absolute Error; Maximum leaf nodes
Output: Decision Tree Regression with M-Estimator on Leaf nodes

Process:

1. Initialise an empty tree
2. Initialise the criteria function (criteria MAE) and maximum leaf node
3. Run the loop as long as it does not meet the maximum leaf node conditions
4. Find the features and threshold values that provide the best data separation using the criteria
5. Create two new branches of the tree according to the features and threshold values found
6. On the left branch, only data that has a feature value less than or equal to the threshold is processed
7. On the right branch, only data that has a feature value more significant than the threshold is processed
8. Continue dividing the data using 3-7 if it does not meet the maximum leaf node conditions.
9. Create a Huber M-estimator model using sample data trains at each leaf node
10. Return tree

MAE criteria. The research uses recursive binary splitting to divide the numerical and categorical features into binary. Divide data into sub-nodes with the best threshold value. The best threshold value is the value of the data divider with the most petite MAE at a node.

The number of nodes in the leaf node (minimum leaf node) is limited to prevent overfitting [21]. Another limitation that we use is the maximum number of samples in each leaf node built, and this is to make it easier to build the model in the leaf node. Tree building will stop when the limit is reached.

The traditional DTR leaf node uses the average or median value to produce one predicted value for each leaf node. Thus, data that enters the same leaf node will have the same predictive value, so the large number of value labels in the DTRmaking biased results. Making leaf nodes uses regression modelling to make the prediction results more precise and avoid bias in the prediction results. Thus, Algorithm 2 in line 19 is the modification we propose which the leaf node DTR will partition the data and then use robust regression modelling with an Huber M-estimator on the leaf nodes to create a model for each leaf node.

2.5 Evaluation criteria

Test evaluation can do using several methods of performance measurement. The results of the research do expect to achieve the best results. However, it is inevitable that, in practice, it will take more work to achieve 100% correct results. In this study, performance evaluation uses the mean absolute error (MAE). MAE is the average absolute error value of the predicted error results (Eq. (2)). This measurement does not pay attention to the MAE positive or negative values.

$$MAE = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n} \quad (2)$$

The error values are typically not averaged to quantify the magnitude of the errors, considering the variations in both positive and negative values. So, if you add it up, the error value will be small. The error must be made into an absolute number or squared and then averaged to avoid this. The advantage of MAE is that it is resistant to outliers and unaffected by significant errors [22].

3. Result and discussion

This section will explain the five data sets in the study, as shown in Table 1. The data sets in this study came from UCI Machine Learning Repository [23]. To test the performance of the built model, divide the dataset by 80% for training data and 20% for testing data for the five datasets.

Several studies with several different methods have used these five datasets. The concrete compressive strength dataset [24] has been analysed with DT using k-fold cross-validation. The research [25] used the QSAR fish toxicity, concrete

Table 1. Regression dataset

Dataset	Attributes	Instance	Reference
QSAR fish toxicity	7	908	[26] [25]
Concrete	9	1030	[27] [25]
Superconductivity	81	21263	[28]
Boston Housing	14	507	[25] [29]
Airfoil Self Noise	6	1503	[25] [29]

Table 2. Outlier observation

Dataset	Cook's D	Studentised
QSAR fish toxicity	69	51
Concrete	79	51
Superconductivity	754	977
Boston Housing	30	25
Airfoil Self Noise	114	81

compressive strength, Boston housing, and Airfoil self noise datasets, with 20% for data testing.

QSAR fish toxicity dataset is a dataset that predicts acute aquatic toxicity to Pimephales promelas (Fathead Minnow). The number of instances in this dataset is 908, consisting of 6 features and one label, namely LC50. The LC50 variable is the concentration that causes death in 50% of fish. In comparison, the features in this dataset are molecular.

The concrete compressive strength dataset is a dataset for measuring the compressive strength of concrete. Where concrete compressive strength is a function of age and components of the mixture in manufacturing concrete, this dataset consists of 9 attributes, where one attribute is a label, and 8 attributes are features. The concrete compressive strength attribute is a label, while the mixed components are features of essential and additional components. The trigger for outliers is the different components, namely blast furnace slag, fly ash, and superplasticizer. The function of these three components is to affect the strength of the concrete. So, this feature from being not removed because it affects the power of the concrete.

The superconductivity dataset is a condition in some materials at low temperatures, so this dataset can predict critical temperatures where features are chemical elements. Research conducted by [28] used 20 of the 80 features in the Superconductivity dataset. So this study also used 20 features and one label.

The algorithm performance evaluation will be carried out using training for datasets with outliers and datasets that have removed outliers by sampling the same training data in each data set and using it to predict the same test data. Our evaluation executes two algorithms: DTR with M-estimator as the proposed method and M-estimator without DTR as the baseline method. Both algorithms use the distribution of 80% training data and 20% test for the five proposed datasets (see Table 1).

The outlier remover in this study involves additional methods to check the number of outliers in the dataset used. Checking for outlier data in the data set can use Cook's distance analysis [30] and the studentized residual method [31]. The number of outlier observations for these five datasets is presented in Table 2. We will consider the results of checking outliers for outlier removal, which helps evaluate the performance of the proposed method.

The Table 2, in the QSAR fish toxicity dataset, some observations are outliers from Cook's distance. The limit value of this dataset for Cook's distance is 0.004, obtained from 4 divided by the number of cases. Data that passes this value is an outlier observation. There are 69 outliers in the dataset. Meanwhile, the Studentized detection results found 51 objects as outlier observations or around 5.62%.

The concrete compressive strength dataset has 69 outlier instances from the results of checking Cook's distance (see Table 2). Outlier checking results on the Superconductivity dataset using the Cook's distance method using a limiting value of 0.0002 obtained 754 outlier observations, and 977 instances of Studentized Residual in the Superconductivity dataset are outliers.

In the Boston dataset, 30 out of 507 instances are outliers obtained using Cook's distance. Meanwhile, the results of 25 instances of Studentized Residual are outliers. Outlier checking results for the 114 instances of the Airfoil Self Noise dataset are outliers from checking Cook's distance, and 25 instances are outliers from checking studentized residual.

The presence of outliers in the Boston housing dataset led the study [1] to perform outlier removal to improve prediction evaluation results and achieve good outcomes for the utilized Lasso regression analysis. Dealing with outliers in the data can be approached through other methods, such as adding weights to balance outlier observations with standard observations.

This study works by incorporating weighting to overcome outliers in the dataset using M-estimation with Huber weighting. Using an M-estimator provides good results in predicting datasets with outliers, yielding satisfactory evaluation performance [32]. The previous weighting addition has been investigated in [25] using granular scaling and achieved the best accuracy.

Fig. 3 shows how our modification generates a DTR with the M-estimator on the QSAR fish toxicity dataset using 726 training data. The M-estimator

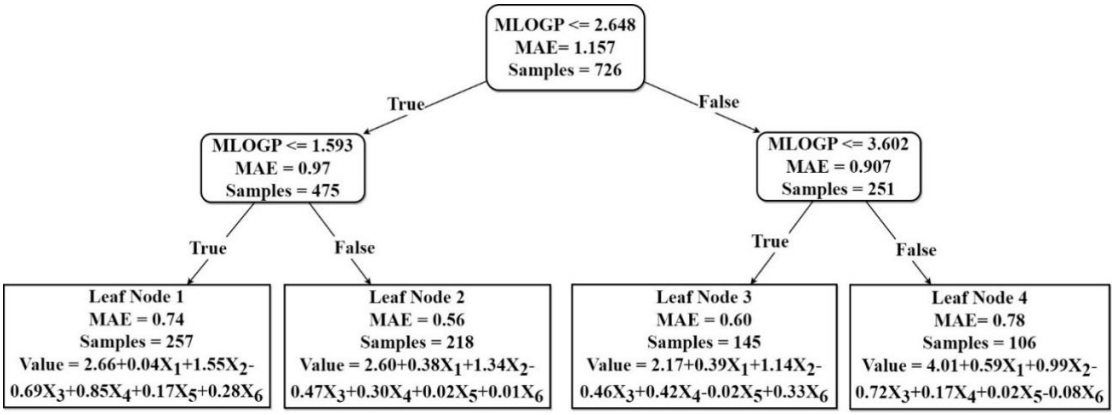

Figure. 3 DTR with M-estimator for predict QSAR fish toxicity

Table 3. Result of comparison between methods

Dataset	Reference	Method	MAE	RMSE
QSAR fish toxicity	[25]	GEN	0.791	-
		Ridge	0.769	-
	Baseline	M-estimator (with outlier)	0.599	0.637
		M-estimator (remove outlier)	0.541	0.491
	Proposed method	DTR M-estimator (with outlier)	0.583	0.622
		DTR M-estimator (remove outlier)	**0.522**	0.498
Concrete	[25]	GEN	4.348	-
		Ridge	8.100	-
	[24]	Decision Tree Regression (k = 10)	5.210	-
	[29]	ABRF	4.677	-
	Baseline	M-estimator (with outlier)	7.476	9.520
		M-estimator (remove outlier)	7.625	9.621
	Proposed method	DTR M-estimator (with outlier)	**3.963**	**31.401**
		DTR M-estimator (remove outlier)	4.666	38.902
Superconductivity	Baseline	M-estimator (with outlier)	14.974	19.635
		M-estimator (remove outlier)	14.611	18.900
	Proposed method	DTR M-estimator (with outlier)	**9.140**	**14.944**
		DTR M-estimator (remove outlier)	10.549	38,640
Boston Housing	[1]	Lasso	-	2.833
	[25]	GEN	3.051	-
		Ridge	3.926	-
	[29]	ABRF	2.520	-
	Baseline	M-estimator (with outlier)	3.028	4.316
		M-estimator (remove outlier)	2.560	3.556
	Proposed method	DTR M-estimator (with outlier)	**2.021**	**2.772**
		DTR M-estimator (remove outlier)	2.181	4.176
Airfoil Self Noise	[25]	GEN	3.288	-
		Ridge	3.545	-
	[29]	ABRF	2.066	-
	Baseline	M-estimator (with outlier)	3.474	4.403
		M-estimator (remove outlier)	3.202	3.996
	Proposed method	DTR M-estimator (with outlier)	**1.644**	**2.527**
		DTR M-estimator (remove outlier)	1.952	2.753

coefficients for each feature show how these contribute to changes in the LC50 predictions of the QSAR fish toxicity dataset tested in our proposed method. We get four leaf nodes in this dataset which are the best for making predictions. In the leaf node 1 model, the highest regression coefficient is 1.55. The intercept value is 2.66, which shows the predicted value when eight features are with zero, then the expected value for LC50 is 2.66. The MAE values the use as an evaluation for each leaf node. The leaf node has a more petite MAE compared to its internal node. Other information is the number of data train samples

at each node. Each dataset has a different number of leaf nodes and a different M-estimator model.

Performance measurement uses the mean absolute error (MAE). As shown in Table 3, the DTR method with the M-estimator outperforms those without DTR. Table 3 shows that without DTR, outlier removal has a superior performance to removing outliers for four datasets concrete datasets, superconductivity datasets, Boston housing datasets, and Airfoil self noise datasets, except for the QSAR fish toxicity dataset. This result shows that the M-estimator learning scheme on leaf nodes helps improve traditional DTR performance and conquer the problem of outliers in the dataset.

Research by [25] used the granular elastic network (GEN) method as one of the methods proposed and tested for several UCI datasets. The GEN method is the best in research [25], used to deal with multicollinearity problems in data. The four datasets in the study [25] are the same as the datasets used in this study QSAR fish toxicity datasets, concrete datasets, Boston housing datasets, and Airfoil self noise datasets.

QSAR has a result for the data test, which is 0.791. The results of [25] are more significant than the findings in this study using the outlier remover dataset with the proposed modification, MAE 0.522. For performance without DTR the M-estimator has more minor results in the dataset with the outlier remover compared to using the outlier dataset. However, the difference in results is less than 0.1, so the proposed method is quite good and can be considered.

A new approach called ABRF (the attention-based random forest) and its modifications for the attention mechanism to apply the random forest (RF) by [29]—the datasets used in this study are Concrete datasets, Boston housing datasets, and Airfoil self noise datasets.

The concrete dataset is one of the datasets used to test the proposed method. The results obtained in Table 3 show that the proposed method can conquer the outlier problem in the concrete dataset by having an MAE of 3.963, the value of the DTR with the M-estimator, without eliminating outlier observations. This dataset was also tested using the DTR with the 10 k-fold methods [24], which obtained an MAE value of 5.210, and the proposed ABRF method [29] received an MAE of 4.677.

The MAE results were not better when compared with the evaluation results of the concrete dataset in this study. Thus, the proposed modifications can excel in the case of Concrete dataset outliers. Testing data with outliers on the superconductivity dataset performs well on the DTR with the M-estimator. The DTR M-estimator obtains an MAE of 9.140 (see Table 3), whereas when eliminating outliers in the dataset, it obtains a larger MAE of 10.549 (see Table 3).

The proposed GEN method [25] and the ABRF method [29] used as much as 20% of the Boston housing dataset for the test, obtaining MAE values of 3,051 and 2,520. Our test better evaluates the proposed DTR M-estimator using a dataset with outliers, with an MAE of 2.021. Our fifth dataset, the Airfoil Self Noise dataset, also better estimates the proposed DTR M-estimator using a dataset with outliers, with an MAE of 1.644. These results also performed better than the GEN method [25] and the ABRF method [29], which have MAE 3.288 and 2.066.

4. Conclusion

We introduce the Huber-weighted M-estimator model for each leaf node of the constructed DTR. We analyse the difference between the dataset with outliers and the dataset remover outliers to prove the effectiveness of the proposed method. Experiments on five commonly used machine learning datasets show that the DTR with the Huber M-estimator outperforms the datasets with outliers, except for the QSAR fish toxicity dataset. In addition, we prove through comparative experiments that the Huber M-estimator inputted with DTR is better than the Huber M-estimator without DTR. This study also obtained better results when compared to previous studies so that this study could optimise the results of the analysis on outlier datasets.

For the development of future research, we can add other outlier datasets in the study to test the Huber M-estimator. Furthermore, the addition of other robust regression estimation methods, such as the MM-estimator, LTS-estimator, and other estimators, can also be carried out.

Acknowledgements

We would also like to thank the Program in Computer Science, Universitas Gadjah Mada, for who have supported this research.

Author contributions

The contributions by the authors for this research are as follows: conceptualization, methodology, formal analysis, Salsabila Basalamah and Agus Sihabuddin; investigation, resources, writing—draft preparation, Salsabila Basalamah; writing—review and editing, validation, supervision, Agus Sihabuddin.

References

[1] S. Sanyal, S. K. Biswas, D. Das, M. Chakraborty, and B. Purkayastha, "Boston House Price Prediction Using Regression Models", *Int. Conf. Intell. Technol.*, pp. 1–6, 2022.

[2] A. Thebelt, J. Wiebe, J. Kronqvist, C. Tsay, and R. Misener, "Maximizing information from chemical engineering data sets: Applications to machine learning", *Chem. Eng. Sci.*, Vol. 252, 2022, doi: 10.1016/j.ces.2022.117469.

[3] L. G. M. Souza and D. C. Santos, "A Performance Comparison of Robust Models in Wind Turbines Power Curve Estimation: A Case Study", *Neural Process. Lett.*, Vol. 54, No. 4, pp. 3375–3400, 2022, doi: 10.1007/s11063-022-10772-2.

[4] Y. Güney, H. Bozdogan, and O. Arslan, "Robust model selection in linear regression models using information complexity", *J. Comput. Appl. Math.*, Vol. 398, p. 113679, 2021, doi: 10.1016/j.cam.2021.113679.

[5] S. Zhang, R. Yao, C. Du, E. Essah, and B. Li, "Analysis of outlier detection rules based on the ASHRAE global thermal comfort database", *Build. Environ.*, Vol. 234, No. March, pp. 1–22, 2023, doi: 10.1016/j.buildenv.2023.110155.

[6] S. Naumets and M. Lu, "Investigation into Explainable Regression Trees for Construction Engineering Applications", *J. Constr. Eng. Manag.*, Vol. 147, No. 8, p. 04021084, Aug. 2021, doi: 10.1061/(ASCE)CO.1943-7862.0002083.

[7] H. Yan, Y. Qi, Q. Ye, and D. J. Yu, "Robust Least Squares Twin Support Vector Regression With Adaptive FOA and PSO for Short-Term Traffic Flow Prediction", *IEEE Trans. Intell. Transp. Syst.*, Vol. 23, No. 9, pp. 14542–14556, Sep. 2022, doi: 10.1109/TITS.2021.3130264.

[8] O. V. Klochko, R. S. Gurevych, V. M. Nagayev, L. Yu Dudorova, and T. P. Zuziak, "Data mining of the healthcare system based on the machine learning model developed in the Microsoft azure machine learning studio", *J. Phys. Conf. Ser.*, Vol. 2288, No. 1, pp. 1–11, 2022, doi: 10.1088/1742-6596/2288/1/012006.

[9] A. Albano, M. Sciandra, and A. Plaia, "A weighted distance-based approach with boosted decision trees for label ranking", *Expert Syst. Appl.*, Vol. 213, No. October 2022, 2023, doi: 10.1016/j.eswa.2022.119000.

[10] A. Ozyilmaz *et al.*, "The Relationship between Health Expenditures and Economic Growth in EU Countries: Empirical Evidence Using Panel Fourier Toda–Yamamoto Causality Test and Regression Models", *Int. J. Environ. Res. Public Health*, Vol. 19, No. 22, 2022, doi: 10.3390/ijerph192215091.

[11] K. M. Hindrayani, T. M. Fahrudin, R. P. Aji, and E. M. Safitri, "Indonesian Stock Price Prediction including Covid19 Era Using Decision Tree Regression", In: *Proc. of 2020 3rd Int. Semin. Res. Inf. Technol. Intell. Syst. ISRITI 2020*, pp. 344–347, 2020, doi: 10.1109/ISRITI51436.2020.9315484.

[12] D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedure*, Third. Florida: Chapman & Hall/ CRC Press, 2004.

[13] H. M. Walpole and R. E. Raymond, *Probability & Statistics for Engineers & Scientists*, Ninth Edit. Boston: Prentice Hall, 2011.

[14] D. J. Olive, *Applied Robust Statistics*. Carbondale: Southern Illinois University Department of Mathematics, 2005.

[15] R. Andersen, *Modern Methods for Robust Regression*. United States of America: SAGE Publications, Inc., 2011.

[16] R. D. Cook, "Detection of Influential Observation in Linear Regression", *JSTOR*, Vol. 19, No. 1, pp. 15–18, 1977, doi: 10.1080/00401706.1977.10489493.

[17] H. Draper and N R. Smith, *Applied Regression Analysis*, Three. New York: John Wiley and sons, 1998.

[18] P. J. Huber, "Robust Estimation of a Location Parameter", *Ann. Math. Stat.*, Vol. 35, No. 1, pp. 73–101, 1964, doi: 10.1214/aoms/1177703732.

[19] P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection*. New York: John Wiley & Sons, 1987.

[20] C. Molnar, *Interpretable Machine Learning. A Guide for Making Black Box Models Explainable*. Lulu.com, 2019.

[21] S. Marsland, *Machine Learning A Probabilistic Perspective*. London: Chapman and Hall/CRC, 2012.

[22] F. Provost and T. Fawcett, *Data Science for Business*. United States of America: O'Reilly Media, 2013.

[23] D. Dua and C. Graff, "UCI machine learning repository", 2017. http://archive.ics.uci.edu/ml.

[24] W. Ahmad, A. Ahmad, K. A. Ostrowski, F. Aslam, P. Joyklad, and P. Zajdel, "Application of Advanced Machine Learning Approaches to Predict the Compressive Strength of Concrete Containing Supplementary Cementitious Materials", *Materials (Basel).*, Vol. 14, No. 19, p. 5762, Oct. 2021, doi: 10.3390/ma14195762.

[25] L. He, Y. Chen, C. Zhong, and K. Wu, "Granular Elastic Network Regression with Stochastic

Gradient Descent", *Mathematics*, Vol. 10, No. 15, pp. 1–15, 2022, doi: 10.3390/math10152628.

[26] M. Shahhosseini, G. Hu, and H. Pham, "Optimizing ensemble weights and hyperparameters of machine learning models for regression problems", *Mach. Learn. with Appl.*, Vol. 7, No. December 2020, p. 100251, 2022, doi: 10.1016/j.mlwa.2022.100251.

[27] S. A. R. Shah, M. Azab, H. M. S. E. Din, O. Barakat, M. K. Anwar, and Y. Bashir, "Predicting Compressive Strength of Blast Furnace Slag and Fly Ash Based Sustainable Concrete Using Machine Learning Techniques: An Application of Advanced Decision-Making Approaches", *Buildings*, Vol. 12, No. 7, p. 914, Jun. 2022, doi: 10.3390/buildings12070914.

[28] Y. Liu *et al.*, "Federated Forest", *IEEE Trans. Big Data*, Vol. 8, No. 3, pp. 843–854, 2022, doi: 10.1109/TBDATA.2020.2992755.

[29] L. V. Utkin and A. V. Konstantinov, "Attention-based random forest and contamination model", *Neural Networks*, Vol. 154, pp. 346–359, 2022, doi: 10.1016/j.neunet.2022.07.029.

[30] C. O. Arimie, E. O. Biu, and M. A. Ijomah, "Outlier Detection and Effects on Modeling", *OALib*, Vol. 07, No. 09, pp. 1–30, 2020, doi: 10.4236/oalib.1106619.

[31] A. M. Baba, H. Midi, M. B. Adam, and N. H. A. A. Rahman, "Detection of Influential Observations in Spatial Regression Model Based on Outliers and Bad Leverage Classification", *Symmetry (Basel).*, Vol. 13, No. 11, p. 2030, Oct. 2021, doi: 10.3390/sym13112030.

[32] E. Bas, "Robust fuzzy regression functions approaches", *Inf. Sci. (Ny).*, Vol. 613, pp. 419–434, 2022, doi: 10.1016/j.ins.2022.09.047.