IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Mathematical Modelling of Engineering Problems
Vol. 9, No. 5, October, 2022, pp. 1217-1224
Journal homepage: http://iieta.org/journals/mmep

A Comparative Study of Regression Machine Learning Algorithms: Tradeoff Between Accuracy and Computational Complexity

Dunia Abas Gzar[1*], Ali Majeed Mahmood[1], Maythem K. Abbas[2]

[1] Control and Systems Engineering Department, University of Technology-Iraq, Al-Sina'a St., Baghdad 10066, Iraq
[2] Department of Internet Engineering and Computer Science, Lee Kong Chian Faculty of Engineering and Science, Universiti Tunku Abdul Rahman, Sungai Long Campus Kajang, Selangor 43400, Malaysia

Corresponding Author Email: cse.20.18@grad.uotechnology.edu.iq

https://doi.org/10.18280/mmep.090508

Received: 25 July 2022
Accepted: 21 October 2022

Keywords:
artificial intelligent, machine learning, support vector regression, random forest, linear regression, neural network

ABSTRACT

The computational complexity of Machine Learning is a mathematical study of the possibilities for efficient learning by computers which is the determination of looking for the best methods to solve a problem. The accuracy of a regression model's predictions must be reported as an error. According to the researchers, the most problematic issue is the lack of a properly defined machine learning assessment. In this research, Various types of machine learning regression algorithms, namely, Linear Regression, Support Vector Regression, Random Forest Regression, and Multilayer Perceptron Neural Network have been used to process and analyze the collected data in terms of comparison of their accuracy and the computational complexity. The applied dataset was collected using IoT sensors seeking an appropriate algorithm that is the fittest to the collected data to design a model system that represents the goal of specific future applications. The result shows that the Random Forest regression has the highest computational complexity and highest accuracy depending on the calculated error metrics (Mean Square Error, Mean Absolute Error, and R Squared score) which are (0.0002, 0.005, and 0.995) respectively. Based on that, Random Forest Regression will be adapted and implemented with the structure of a planned design system.

1. INTRODUCTION

Linear Regression (LR), Support Vector Regression (SVR), Random Forest Regression (RFR), and multilayer perceptron (MLP) are the machine learning regression models used in this study for data processing and analysis (collected data from IoT sensors node). In many proposed systems, various types of Machine Learning Regression Algorithms (MLAs) have been used in many different applications to process and analyze data [1]. The goal is to show the computational difficulty of each of them and to encourage the use of efficient learning techniques. The concept of computational complexity is offered to aid in the explanation of how efficiently learnable things are. In general, an algorithm's efficiency may be determined by the time it takes to execute as a function of the amount of input [2]. To determine the efficiency of an algorithm can be used Big-O Notation (Computational Complexity). Big-O notation is a measure for determining the complexity of algorithms. It denotes the connection between the algorithm's input and the steps necessary to run it. It's frequently used to assess the efficiency of different algorithms by determining how much memory is required and how long it takes to execute them.

Some of the Reality of Computational Complexity:

• Model complexity, duration, and speed are proportional to the magnitude of the input data.

• While Big O Notation is important for calculating an algorithm's efficiency in terms of time and space, many other factors influence how long it takes to train a machine learning model.

• Some research shows that as accuracy increases, the computation complexity also increases accordingly as shown in Figure 1 [2]. The complexity of an algorithm is determined by the line code, and it might be O(n), O(n2), O(log n), and other values. Denote the following: n is the number of training examples (rows), d denotes the number of dimensions (columns), and k denotes the number of neighbors [3].

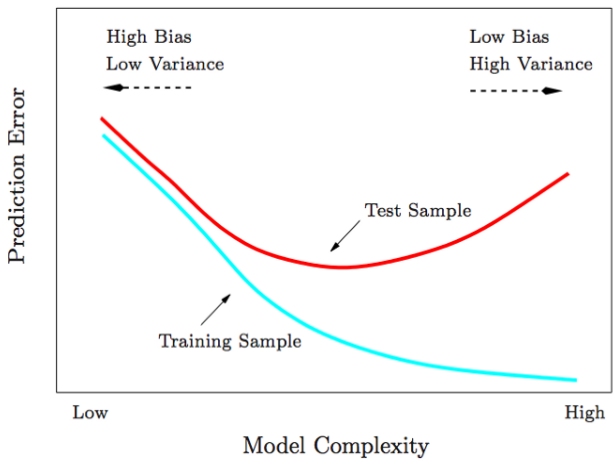

Figure 1. The relationship between complexity and accuracy [2]

2. RELATED WORKS

This section displays some of the related work that deployed machine learning techniques considering the model accuracy.

Shah et al. [4] used machine learning approaches to model and produce empirical equations for the surface water quality of the upper Indus river basin over 30 years. The goal is to identify the most trustworthy model that can correctly predict river water quality. Nash-Sutcliffe Efficiency (NSE), Root Mean Square Error (RMSE), R Squared (R^2), and Mean Absolute Error (MAE) were some of the evaluation metrics utilized to rate the performance of the models. Gene expression programming outweighs other techniques in terms of accuracy. Cheng et al. [5] focused on using of molecular-orbital-based clustering (MOB). Regression Clustering (RC) is applied in this work. The outcome demonstrates that the combined RC/LR/RFC and RC/GPR/RFC MOB-ML implementations are found to produce good prediction accuracy with noticeably shorter wall-clock training times. Jumin et al. [6] This work offers a precise model based on machine learning techniques to forecast tropospheric ozone concentration in some of Malaysia's cities. The proposed models were created utilizing three years' worth of historical data for various factors as input to forecast tropospheric ozone levels for the next 24 and 12 hours. Investigated machine learning methods include boosted decision trees, neural networks, and linear regression. For all stations, boosted decision tree techniques outperformed linear regression and neural network algorithms. Using the 12-hour dataset instead of the 24-hour dataset, where R^2 values were equal to 0.91, 0.88, and 0.87 for the three analyzed stations, increased the performance of the suggested model.

Likewise. Many other related works consider the complexity of ML algorithms. For example, Majeed [7] explains that due to the distributed storage of the needed algorithm or the memory space, the suggested WSN-MLP architecture has a low space complexity. The number of patterns in a dataset and the number of neurons in the hidden and output layers are shown to significantly impact time complexity Sun et al. [8] suggest that FPTC and the coefficient ω were proposed to measure each classifier's running time. The FPTC of five common classifiers (kNN, LR, CART, RF, and SVM) was created. Mbaabu [9] provides an efficient random forest-based feature selection approach for improving MLA performance while processing big and complicated datasets. The proposed approach considerably decreases the processing time of MLAs without significantly lowering accuracy. The results of the simulations support the suggested algorithm's efficacy and effectiveness.

Donges et al. [10] show that the time complexity of a linear system described by a covariance matrix is O (logN) or O (1). These findings show that the circuit can solve linear systems quickly in a wide range of applications, indicating that IMC is a promising choice for future big data and machine learning accelerators. Ray [11] demonstrates that the complexity of KNN and SVM is analyzed, and other complexity-reducing methodologies are then examined. It is also suggested to use a hybrid method that combines KNN and SVM to use the possibilities of the two techniques.

3. THE PROPOSED EVALUATION APPROACH

In this research, Machine Learning Regression (MLR) models were used to investigate the fittest one which can be applied to designing smart systems for predication the health of plants. The investigation started by using IoT sensors to collect data from a farm to form a dataset that can be used to train the MLR. The investigation of this work depends mostly on calculating the processing time complexity represented by calculating the elapsed time and then calculating the accuracy for each algorithm, which can be assisted in the decision of choosing the model that can be implemented in the structure of the suggested system that can monitor and control a farm with the highest precision in terms of predication health of plant and decision-making accuracy. In the proposed evaluation strategy, four metrics are used to evaluate the Point of Interest (POI) regression machine learning algorithms, which are Mean Squared Error (MSE), Root Mean Squared Error (MAE), and the Elapsed time MATLAB function for processing time measurement. As shown in Figure 2, the evaluation of each algorithm relies on the accuracy of prediction and the corresponding processing overhead.

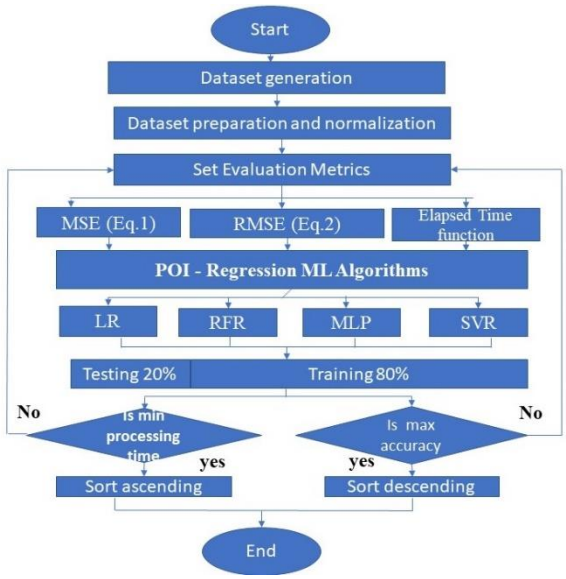

Figure 2. The proposed evaluation strategy

4. MACHINE LEARNING

Machine Learning (ML) model learns the relationship between the dataset's input (independent) features and the target (dependent) feature to make future predictions [12]. To test the model's performance, a new equivalent dataset with only the input characteristics is supplied, and the goal of the model is to predict the target variable using the knowledge gathered during training. The expected values are compared to the actual target values using an appropriate performance metric [11, 13].

Its techniques employ computational approaches to learn directly from datasets rather than relying on pre-programmed equations as a model. As the number of training samples provided rises, the algorithms adjust to improve their performance [13]. ML has broadly been used for computation processing based on experiences for improving and increasing accuracy for both performance and prediction [14]. Due to its capacity to offer accurate estimated input-output data with solid correlations, machine learning has seen a lot of use in remote sensing, which opens up a lot of possibilities for biophysical parameter retrievals and applications. Predication can be applied or data through experiences. Regression aims

to find the line which represents the relationship between data with a minimum vertical distance between data points and the regression line. Various application regression algorithms can be used for such as prediction purposes, forecasting tasks, and time-series techniques to find the most accurate relationship between variables [15]. Linear, nonlinear, Gaussian process regression model (GPRM), support vector machine (SVM) regression, generalized linear model (GLM), decision tree (DT), ensemble approaches, and neural networks are the most well-known regression algorithms. In regression methods, several equations are employed, which are discussed below.

Mean Squared Error (MSE): For prediction tasks, MSE considers the most regression measurement. It is the squared difference between the actual and projected numbers. MSE is differentiable and has a curved shape, making optimizations easier. Large error values are penalized by MSE which is represented by Eq. (1) below:

$$MSE = (\frac{1}{n}) \sum_{i=1}^{n} (yi - \hat{y}) \quad (1)$$

where, Y_i is the actual target value. $\hat{y}$ is thepredicted target value; n is the total number of data points.

Root Mean Squared Error (RMSE) Equation: The difference between the root square of real and predicted costs is calculated using the RMSE method. The absolute degree of fit is measured by the RMSE which is represented by the equation below. As a result, RMSE calculates the spread of the residuals as shown in Eq. (2):

$$RMSE = \sqrt{(\frac{1}{n}) \sum_{i=1}^{n} (yi - \hat{y})} \quad (2)$$

4.1 Linear regression algorithm

Linear Regression Algorithm (LRA) is a type of regression model used to predict parameters for forecasting by adding constant value (bias term) to normal weight variables of certain applications. This type of regression has a specific number of parameters (weights) which represent the number of features. With this algorithm, the best fit line represents the value of the total prediction with an error close to zero. The distance between the prediction value and an actual one of this regression is called an error.

As an algorithm is important to identify its efficiency for specific data to process and analysis them with less time to train the model and make a decision as fast as possible. The temporal complexity of training and evaluation may vary greatly. Parametric models, such as linear regression, might take a long time to train but can be tested quickly (although linear regression has a long training time, they are efficient during test time [9]. Table 1 shown below, demonstrates the linear regression algorithm's advantages and disadvantages. The linear regression algorithm flowchart shows in Figure 3. The equations [16], and pseudo-code of linear regression are shown below:

$$Y = a + bX \quad (3)$$

where, Y is the dependent variable; X is the independent; b is the slope of the line and a is the y-intercept.

$$a = \frac{(\sum y)(\sum x2) - (\sum x)(\sum xy)}{n(\sum x2) - (\sum x)2} \quad (4)$$

$$a = \frac{n(\sum xy) - (\sum x)(\sum y)}{n(\sum x2) - (\sum x)2} \quad (5)$$

Table 1. Demonstrates some advantages and disadvantages of LRA [17]

Advantage	Disadvantage
LR is simple and easy to implementation	LR technique outliers have a huge effect on regression and boundaries
Less complex algorithm compared to other	LR assumes a linear relationship between dependent and independent variables. It assumes independence between attributes
LR is susceptible to overfitting which can be avoided by using regularization (L1&L2) techniques and cross-validation.	LR does not have a complete description of the relationship among variables.

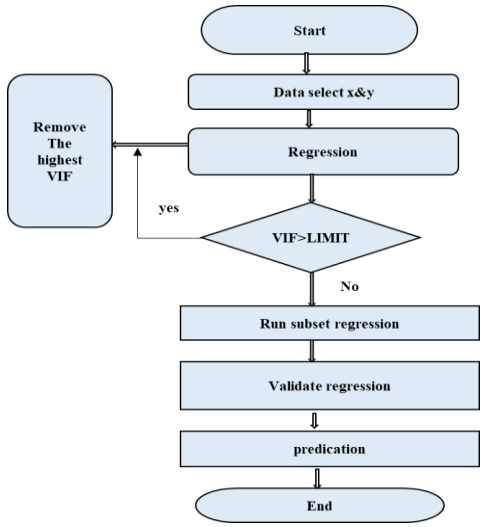

Figure 3. Flowchart linear regression algorithm [18]

Algorithm 1: Pseudocode of LR [19]

Input: Cost- function (x,y, theta)
Output: mapping function

```
m=length of y
h-x=Dot product of x and theta
algorithm (np. Square (h. x-y)/(2*m)
return j
gradient  descent (x, Y, theta, alpha, num  iters):
m=number of samples
p=X
t=theta
j=[]
For I in rang (number of iterations):
Cost=compute Cost m (P,Y,T)
j .append(cost)
h_x=np.dot (p, t)
err=h_x-Y
for f in range (theta.size):
t [f]= t[f]- alpha/m
*(np.sum((np.dot(p[:,f] .T, err))))
#Print (t)
Return j,t
End
```

4.2 Support vector regression algorithm

Support Vector Regression (SVR) is a supervised learning model. SVR can be presented in both linear and nonlinear regressions. Its goal of applying linear regression is to reduce the error between the prediction value and actual data [20]. So, the attempt of Support Vector Regression to reduce the errors and avoiding from exceeding the threshold. An SVR model has several hyper-parameters that determine the performance of the model as follows below:

• Kernel: A function for associating lower-dimensional data with higher-dimensional data.

• Hyper level: A line that divides data, regression support vector, the regression problem is a line that assists the hyperplane in predicting continuous or goal values.

• Restrictions on determination: Boundaries are effectively the super plane's determination limit. Carrier vectors can be found both inside and outside of the barrier. The hyperplane with the greatest number of points in the boundary is used to find the optimal line.

• The data points nearest to the decision border are called support vectors. The minimal or minimum distance between points is used. After training, calculate the loss.

The loss between the actual values in the testing dataset and the predicted values can be calculated using a cost function called RMSE which is represented by Eq. (2). The RMSE of a formula determines the absolute fit of data to find the fittest of actual data to predicted values. With the small value of RMSE refers to a better fit and accurate measure for determining prediction models. Table 2 shows some cons and pros of this algorithm. Figure 4 demonstrates the SVR algorithm. As well as, the pseudo-code of this algorithm is mentioned below:

Algorithm 2: Pseudocode of SVR [21]

Input: Datasets with p* variables and binary outcome
Output: Ranked list of variables according to their relevance

Train the SVM model:
P←p*:
variables
SVM p←SVM with the optimized tuning parameters for the p variables and observations in Data;
wp←calculates weight vector of the SVM (w_{p1}, ..., w_{pp});
rank. Criteria ←(w^2_{p1}, ..., w^2_{pp})
min. rank. Criteria←variable with the lowest value in rank. calculator;
Remove min. rank. Criteria from Data;
$Rank_p$ ←min.rank.criteria;
p←p-1;
end
Rank ←variable in Data∉($Rank_{p2}$, ..., $Rank_{p*}$);
Return ($Rank_{p2}$, ..., $Rank_{p*}$)

Table 2. The advantage and disadvantages of SVR [22]

Advantage	Disadvantage
Provide a better result even with a small amount of information	There is difficulty in choosing the appropriate kernel solution function.
Works well with unstructured data	When the dataset is large, the training process is long.
Solve complex problems	Difficult to interpret because of personal factors and variable weights.
Relatively good scaling with high dimensional data	The weight of the variable is not constant which contributes each variable to the output is variant.

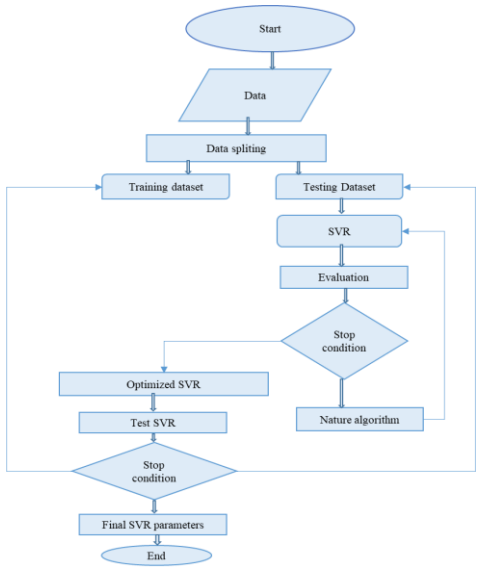

Figure 4. Flowchart of SVR algorithm [23]

4.3 Random forest algorithm

RFR is a supervised machine learning technique that is frequently used to solve classification and regression issues [24]. It is creating a decision tree with different samples and voting on the categorization and mean for regressions by a majority vote. Considerably, one of the most important features of the Random Forest algorithm is the ability to process datasets that contain continuous variables, as in the case of regression, and categorical variables, as in the case of classification.

RF is created from subsets of data and the final output is created by averaging our majority ranking. It is relatively slower in computing processing. RF doesn't use any formula, the result provides by averaging selected observations that build a decision tree [25]. Table 3 explains some of the cons and pros of this algorithm, also Figure 5 of the same algorithm mentioned. The pseudo-code of this algorithm is mentioned below.

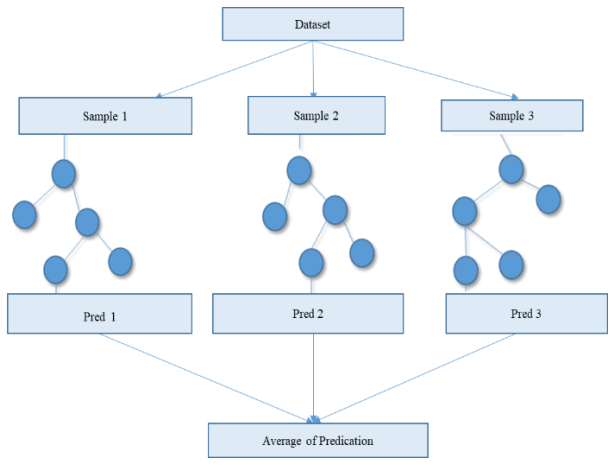

Figure 5. Explains RFR algorithm [26]

The random forest method has the following steps:

Step 1: In RF, n random records are chosen at random from a data collection of k records.

Step 2: For each sample, individual decision trees are built.

Step 3: Each decision tree produces a result.

Step 4: For classification and regression, the final output is based on majority voting or averaging, accordingly.

The math model of this algorithm is many that are used in many types of research. For example, the Gini Index is used as a cost function to evaluate splits in the dataset as shown in Eq. (6) [22].

$$Gini(t) = 1 - \sum_{i=0}^{c-1} \left[p\left(\frac{i}{t}\right) \right]^2 \quad (6)$$

Algorithm 3: Pseudocode of RFR [27]

Input: N is the input patterns space
Output: To generate c classifiers:
for i=1 to c do
Randomly sample the training data D with replacement to produce D_i
Create a root node, N_i containing D_i
Call Build Tree (N_i)
end for
Build Tree (N):
If N contains instances of only one class, then return
else
Randomly select x% of the possible splitting feature in N
Select the feature F with the highest information gain to split on
Create f child nodes of $N, N_i, \ldots, N_f$, where F has f possible values($F_i, \ldots, F_f$)
For i=1 to f do
Set the content of N to D_i, where D is all instances in N that match
F
Call Build Tree (N_i)
end for
end if

Important Features of Random Forest:

1. Diversity Individual trees do not take into consideration all features, variables, or characteristics, and each tree is unique.

2. The Dimensionality Curse is a term that refers to the phenomenon of having more than one dimension. Because not all trees describe all features, there is a reduction in feature space.

3. Parallelization is number three. Various data and qualities are used to build each tree independently. This means you can use the CPU to create a random forest to your heart's content.

4. Split your training and testing. There is no need to divide the train and test data in a random forest because 30% of the data does not appear in the decision tree.

Table 3. Displays the advantages and disadvantages of (RFR) [28]

Advantage	Disadvantage
One of the most accurate algorithm	RFs have been observed to overfit for some data.
Efficient on large data	A large number of data make the algorithm slow for real-time prediction.
Handle a lot of variables (powerful)	Need to choose the number of trees
Generate forests can be saved for future use and able to deal with non-linear problems.	Trained model hard for human interpretation

4.4 Multilayer perceptron algorithm (MPA)

Artificial neural networks with a multi-layer perceptron design are the most complicated algorithm [29]. Multiple layers of the perceptron make up the majority of it. Stock analysis, image identification, spam detection, and election voting prediction are all tasks that the multilayer perceptron (MLP) is utilized. MLPs also learn by adjusting the weights of their perception to achieve a low error rate on the training data. This has traditionally been done using the back propagation technique, which seeks to reduce the MSE. The backpropagation (BP) algorithm is used to display a multilayer perceptron (MLP) algorithm which can be trained with a BP algorithm [30]. On the other hand, backpropagation with a hidden layer between the input and output layer may generalize the ordinary perception and may also estimate any continuous function to a reasonable degree of accuracy with a small number of hidden layers, as stated [27]. Figure 6 perception of main layers of MLP (input layer, hidden layer, output layer). Bellow Table 4 explains the cons and pros of this algorithm, as well as Figure 7 of MLP steps of an algorithm, which is mentioned down below. There are some of the MLP equations which are represented by Eqns. (6), and (7), (8) [31] down below as well. Figure 6 shows the algorithm steps of the MLP algorithm and the pseudo-code of the same algorithm mentioned as well.

$$h_i^{(1)} = \phi^1\left(\sum_j wij^1 x_j + b_i^1\right) \quad (7)$$

$$h_i^{(2)} = \phi^2\left(\sum_j wij^2 hj^1 + b_i^2\right) \quad (8)$$

$$y_i = \phi^3\left(\sum_j wij^3 hj^2 + b_i^3\right) \quad (9)$$

where, x_j=input units; y_i=the output unit as; $H^{(n)}{}_i$=the units of the hidden layer.

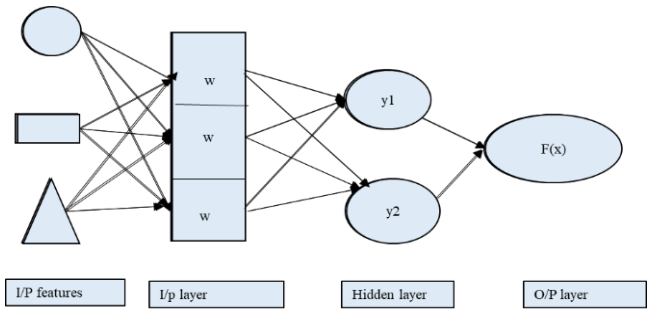

Figure 6. Shows the layers of the MLP algorithm [25]

Table 4. Some advantages and disadvantages of (MLP) [28]

Advantage	Disadvantage
Can be applied to complex nonlinear problems	It is not known to what extent is each independent variable affected by the independent variable.
It deals very well with large data.	Computation is difficult and time-consuming.
Faster in providing prediction after training	The proper function depends on training quality.

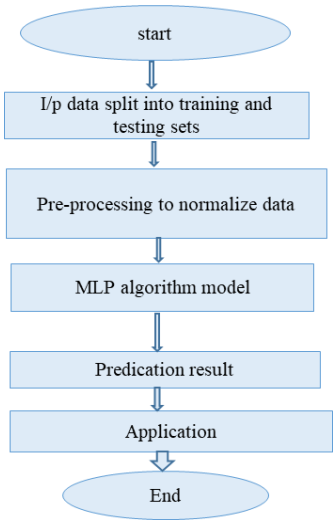

Figure 7. Steps of MLP algorithm [32]

Algorithm 4: Pseudocode of Multilayer Perceptron [26]

```
Input: Choose an initial weight vector ∝w
Output: initialize minimization approach
while the error did not converge do
for all(∝x, ~d) ∈D do
apply ∝x to ϕnetwork and calculate the network output

calculate ðe (∝x)
end for
calculate ðE (E)
for all weights summing over all training patterns
perform one update step of the minimization approach
end while
End
```

5. DATA GENERATION FROM AGRICULTURAL FIELD

The applied dataset in this study is collected using an IoT system that was implemented in a citrus orchard. The hardware system consists of ESP32 integrated WiFi and IoT sensors which are implemented in the field of citrus plants in Karbala city which is located in the middle area of Iraq country. The dataset contains 745 samples which are collected in April 2022 and validated by agricultural experts and an online forecasting station. The collected scenario had done by monitoring the field and recording its parameters such as temperature, humidity, light intensity, and soil moisture. It was collected every thirty minutes day and night times for almost one month by sensors and sent to IoT AWS cloud which is set up on PC windows as a software system part [33]. The aim of measuring the parameters from the farm is to use them in training MLR algorithms to structure a highly advanced system, that is able for monitoring and controlling the health of the plants.

6. EVALUATION AND DISCUSSION

Based on the findings, which show the results of gathered data (collected by IoT sensors) that are analyzed using four types of machine learning regression algorithms in Table 5. As shown, SVR displays the least accuracy with less time to process since it is highly qualified to solve complicated issues. RFR is the most accurate of the others because the random forest method is a vast number of decision trees that are precise to the 'forest' via bagging or bootstrap aggregation. Bagging is a meta-algorithm that improves machine learning algorithms' accuracy. Through an ensemble of techniques. A random forest method overcomes the drawbacks of the decision tree technique. It decreases dataset over-fitting and improves accuracy over-fitting recasts without requiring multiple packages over-fitting that is because the Random Forest algorithm combines many decision trees to produce a more accurate and reliable forecast. That causes time to process data longer as shown in Table 5. In addition, the time processing (computational complexity) has a direct proportion to the accuracy. Based on the result that is shown in Table 5, the least value of each MSE, MAE, and the highest R^2 score for RFR will be implemented in the suggested system which is a smart agriculture system for predicting the health of plants that is required high-precision consideration.

LR is considered to be the best model that is because, without an optimization technique (such as Gradient descent), the cost function of linear regression must be determined through repetitions of weight combinations [34]. As a result, calculation time is proportional to the number of weights and, of course, the quantity of training data. MLP outcomes are as expected, displaying the second least accuracy and long processing time. That is because one of its features which its computation is difficult and time-consuming. Table 6. Below shows the general computational complexity functions of some machine learning algorithms where n=number of training examples, m=number of features, n=number of support vectors=number of neighbors k'=number of trees.

Table 5. Performance evaluation of different ML regression algorithms with the collected dataset

Algorithms	Processing Time	MSE	MAE	R^2 Score
LR	0.001	0.0031	0.039	0.943
RFR	0.211	0.0002	0.005	0.995
MLP	0.189	0.0032	0.037	0.942
SVR	0.008	0.0034	0.045	0.939

Table 6. Shows the big-O notation for various types of machine learning regression algorithms [35]

Algorithm	Time complexity	Test time	Space complexity
LR	$O(n*m^2+m^3)$	O(m)	O(m)
SVR	$O(n^2)$	O(n'*m)	O(n*m)
RFR	O(k'*n*log(n)*m)	O(m*k')	O(k'*depth of tree)
MLP	$O(2^n)$	N/A	N/A

7. CONCLUSIONS

The algorithm's runtime complexity is critical because, at the end of the training, the model on unseen data is assessed by computing the model's accuracy. It aids in the selection of a better model system by assisting in the discovery of a suitable one that is data-fit. When there are several answers to a question, the answer is determined by comparing their runtime and error. However, time constraints may not the only consideration that should make. Consider if the code is readable enough and how much memory it will require. When

developing code, there are typically several options for getting to a solution. In addition, knowing Big O notation is useful for creating algorithms. It aids in determining if the algorithm is speeding up or slowing down. Hence, comparing several machine learning algorithms to determine which one is the most effective aspect in selecting the appropriate MLA. The result shows that RFR displays the most accuracy in processing collected data, and it is the most efficient to fit or analyzed and provides less error than other algorithms that are used in this study. Also, the number of trees and predictors affects the wall clock time for predicting Random Forest regression in which the result also shows that RFR got the highest time processing data. Hence, there is a trade-off between the computational complexity and the accuracy as the complexity computational increases the accuracy increases as well. Future work will be focused on evaluating different ML algorithms and the work may be extended to involve the classification techniques of ML.

REFERENCES

[1] Nasser, A.R., Mahmood, A.M. (2021). Cloud-based Parkinson's disease diagnosis using machine learning. Mathematical Modelling of Engineering Problems, 8(6): 915-922. https://doi.org/10.18280/mmep.080610

[2] Sajee, A. (2020). Model complexity, accuracy and interpretability. Medium. https://towardsdatascience.com/model-complexity-accuracy-and-interpretability-59888e69ab3d.

[3] Serpen, G., Gao, Z. (2014). Complexity analysis of multilayer perceptron neural network embedded into a wireless sensor network. Procedia Computer Science, 36: 192-197. https://doi.org/10.1016/j.procs.2014.09.078

[4] Shah, M.I., Javed, M.F., Abunama, T. (2021). Proposed formulation of surface water quality and modelling using gene expression, machine learning, and regression techniques. Environmental Science and Pollution Research, 28(11): 13202-13220. https://doi.org/10.1007/s11356-020-11490-9

[5] Cheng, L., Kovachki, N.B., Welborn, M., Miller III, T.F. (2019). Regression clustering for improved accuracy and training costs with molecular-orbital-based machine learning. Journal of Chemical Theory and Computation, 15(12): 6668-6677. https://doi.org/10.1021/acs.jctc.9b00884

[6] Jumin, E., Zaini, N., Ahmed, A.N., Abdullah, S., Ismail, M., Sherif, M., Sefelnasr, A., El-Shafie, A. (2020). Machine learning versus linear regression modelling approach for accurate ozone concentrations prediction. Engineering Applications of Computational Fluid Mechanics, 14(1): 713-725. https://doi.org/10.1080/19942060.2020.1758792

[7] Majeed, A. (2019). Improving time complexity and accuracy of the machine learning algorithms through selection of highly weighted top k features from complex datasets. Annals of Data Science, 6(4): 599-621. https://doi.org/10.1007/s40745-019-00217-4

[8] Sun, Z., Pedretti, G., Mannocci, P., Ambrosi, E., Bricalli, A., Ielmini, D. (2020). Time complexity of in-memory solution of linear systems. IEEE Transactions on Electron Devices, 67(7): 2945-2951. https://doi.org/10.1109/TED.2020.2992435

[9] Mbaabu, O. (2020). Introduction to the random forest in machine learning. Berreskuratua-(e) tik https://www.section.io/engineering-education/introduction-to-random-forest-in-machine-learning.

[10] Donges, N., Contributor, E. (2022). Random forest classifier: A complete guide to how it works in machine learning. https://builtin.com/data-science/random-forest-algorithm.

[11] Ray, S. (2021). An analysis of computational complexity and accuracy of two supervised machine learning algorithms—K-nearest neighbor and support vector machine. In Data Management, Analytics and Innovation, pp. 335-347. https://doi.org/10.1007/978-981-15-5616-6_24

[12] Nasser, A., Al-Khazraji, H. (2022). A hybrid of convolutional neural network and long short-term memory network approach to predictive maintenance. International Journal of Electrical & Computer Engineering, 12(1): 721-730. http://dx.doi.org/10.11591/ijece.v12i1.pp721-730

[13] Croy, M.H. (2022). How to calculate time complexity with big o notation, medium. https://medium.com/dataseries/how-to-calculate-time-complexity-with-big-o-notation-9afe33aa4c46.

[14] Mohri, M., Rostamizadeh, A., Talwalkar, A. (2018). Foundations of Machine Learning. MIT Press.

[15] Linear Regression-Researchgate.net. https://researchgate.net/figure/Flow-chart-describing-methodology-LRA-Linear-Regression-analysis_fig3_311002915, accessed on Jun. 06, 2022.

[16] Salim, C., Mitton, N. (2020). Machine learning based data reduction in WSN for smart agriculture. In International Conference on Advanced Information Networking and Applications, pp. 127-138. https://doi.org/10.1007/978-3-030-44041-1_12

[17] Sonmez, H., Gokceoglu, C., Nefeslioglu, H.A., Kayabasi, A. (2006). Estimation of rock modulus: For intact rocks with an artificial neural network and for rock masses with a new empirical equation. International Journal of Rock Mechanics and Mining Sciences, 43(2): 224-235. https://doi.org/10.1016/j.ijrmms.2005.06.007

[18] Linear Regression: Simple Steps, Video. Find the Equation, Coefficient, Slope. Statistics How To, https://www.statisticshowto.com/probability-and-statistics/regression-analysis/find-a-linear-regression-equation.

[19] Guliyev, N.J., Ismailov, V.E. (2018). Approximation capability of two hidden layer feedforward neural networks with fixed weights. Neurocomputing, 316: 262-269. https://doi.org/10.1016/j.neucom.2018.07.075

[20] Houssein, E.H., Dirar, M., Abualigah, L., Mohamed, W.M. (2022). An efficient equilibrium optimizer with support vector regression for stock market prediction. Neural Computing and Applications, 34(4): 3165-3200. https://doi.org/10.1007/s00521-021-06580-9

[21] da Costa, L.A.L., Kunst, R., de Freitas, E.P. (2022). Intelligent resource sharing to enable quality of service for network clients: the trade-off between accuracy and complexity. Computing, 104(5): 1219-1231. https://doi.org/10.1007/s00607-021-01042-5

[22] The Flowchart of Random Forest (RF) for Regression. https://www.researchgate.net/figure/The-flowchart-of-random-forest-RF-for-regression-adapted-from-Rodriguez-Galiano-et_fig3_303835073, accessed on Jun.

6, 2022.
[23] Bhise, K. (2022). Linear Regression-Machine Learning. Medium. https://medium.com/analytics-vidhya/linear-regression-machine-learning-ef8b8899922a.
[24] Khudhur, S.D., Khudhur, D.D. (2022). IgG-IgM antibodies are bad antibody antibodies based on antibodies-based machining models. TELKOMNIKA (Telecommunication Computing Electronics and Control), 20(2): 340-347. http://doi.org/10.12928/telkomnika.v20i2.21649
[25] Sanz, H., Valim, C., Vegas, E., Oller, J.M., Reverter, F. (2018). SVM-RFE: Selection and visualization of the most relevant features through non-linear kernels. BMC Bioinformatics, 19(1): 1-18. https://doi.org/10.1186/s12859-018-2451-4
[26] Random Forests-The Math of Intelligence (Week 7). llSourcell. (2017). https://github.com/llSourcell/random_forests/blob/master/Random%20Forests%20.ipynb, accessed on Jul. 22, 2022.
[27] The Flowchart of Random Forest (RF) for Regression. https://www.researchgate.net/figure/The-flowchart-of-random-forest-RF-for-regression-adapted-from-Rodriguez-Galiano-et_fig3_303835073, accessed on Jun. 06, 2022.
[28] Akintola, A.G., Balogun, A.O., Lafenwa-Balogun, F.B., Mojeed, H.A. (2018). Comparative analysis of selected heterogeneous classifiers for software defects prediction using filter-based feature selection methods. FUOYE Journal of Engineering and Technology, 3(1): 134-137. http://dx.doi.org/10.46792/fuoyejet.v3i1.178
[29] Hassan, H.J., Taqi, A.K. (2016). An algorithm for face recognition based on isolated image points with neural network. International Journal of Computer Applications, 150(2):1-5.
[30] Akkar, H.A., Haddad, S.Q.G. (2020). Diagnosis of lung cancer disease based on back-propagation artificial neural network algorithm. Engineering and Technology Journal, 38(3): 184-196. http://dx.doi.org/10.30684/etj.v38i3B.1666
[31] Comparison of Multi-class Classification Algorithms on Early Diagnosis. https://www.researchgate.net/publication/338950098_Comparison_of_Multi-class_Classification_Algorithms_on_Early_Diagnosis_of_Heart_Diseases, accessed on Jun. 06, 2022
[32] Prediction Flowchart of Multilayer Perceptron Network. Download. https://www.researchgate.net/figure/Prediction-flowchart-of-multilayer-perceptron-network-7_fig1_328201921, accessed on Jun 06, 2022.
[33] Nasser, A.R., Hasan, A.M., Humaidi, A.J., Alkhayyat, A., Alzubaidi, L., Fadhel, M.A., Santamaría, J., Duan, Y. (2021). IoT and cloud computing in health-care: A new wearable device and cloud-based deep learning algorithm for monitoring of diabetes. Electronics, 10(21): 2719. https://doi.org/10.3390/electronics10212719
[34] CSC 311 Fall 2021: Introduction to Machine Learning. CSC311 Fall. (2021). https://www.cs.to22, 2022du/~rgrosse/courses/csc311_f21/, accessed on Jul. 22, 2022.
[35] Computational Complexity of Machine Learning Models - II: Data Science and Machine Learning. Kaggle. https://www.kaggle.com/general/263127, accessed on Jun. 06, 2022.