International Journal on Advanced Science Engineering Information Technology

Vol.14 (2024) No. 6
ISSN: 2088-5334

A Comprehensive Machine Learning Based Modeling of Income Tax Collection

Nghia Chung [a], Thi Mai Thom Do [b], Van Tai Pham [c], Thanh Tan Tran [c], Thi Phuong Thao Nguyen [c], Quang Huy Nguyen [d], Nguyen Van Nguyen [e], Canh Son Nguyen [f], Thanh Nam Dang [g,*]

[a] *Maritime Manning and Training Center, Ho Chi Minh City University of Transport, Ho Chi Minh City, Vietnam*
[b] *Faculty of Financial Management, Vietnam Maritime University, Hai Phong, Vietnam*
[c] *Foreign Trade Faculty, College of Foreign Economic Relations, Ho Chi Minh City, Vietnam*
[d] *Business Administration, College of Foreign Economic Relations, Ho Chi Minh City, Vietnam*
[e] *School of Economics and Law, Tra Vinh University, Tra Vinh, Vietnam*
[f] *Faculty of Economics and Management, Dong Nai Technology University, Bien Hoa City, Vietnam*
[g] *Institute of Maritime, Ho Chi Minh City University of Transport, Ho Chi Minh City, Vietnam*

Corresponding author: [*]nam.dang@ut.edu.vn

***Abstract*—Income tax is one of the important sources of revenue for each country, income tax forecasting is thus one of the important tasks of each country. This work presents a machine learning-based method based on Gross Domestic Product (GDP) and population data to forecast income tax collection. As a result, the violin plot shows the distribution of the data, namely that population values are concentrated around the middle, while GDP has a bimodal distribution, and income tax exhibits a pattern similar to that of the population. On both training and test data, several machine learning models were assessed for accuracy and generalization using Mean Squared Error (MSE), R-squared (R^2), and Mean Absolute Percentage Error (MAPE). With Train MAPE at 2.85% and Test MAPE at 5.53%, Random Forest attained a Train MSE of 25.94 and a Test MSE of 51.00, so indicating good performance but modest overfitting. Although Gradient Boosting had a higher Test MAPE of 6.89% suggesting some overfitting, it scored almost perfect Train MSE of 0.04. While performing poorly on the test data with a Train MSE of 180.50, the Decision Tree fit the training data exactly (Train MSE of 0.00). With Train MSE of 2.32 and Test MSE of 25.49, CatBoost proved constant accuracy over both datasets, it could be considered as the best model for income tax prediction based on GDP and population since it excelled generally in stability and generalization.**

***Keywords*— Machine learning; income tax; gross domestic product; random forest; gradient boosting regression.**

Manuscript received 16 Jan. 2024; revised 7 Aug. 2024; accepted 12 Oct. 2024. Date of publication 31 Dec. 2024.

I. INTRODUCTION

One of Australia's government's primary income sources is tax collected from eligible citizens. It supports infrastructure, education, and healthcare, among other essential public services [1]. The progressive operation of the Australian income tax system results in higher taxes paid as income rises. Income level determines the rates; people pay taxes according to their marginal tax rate. Starting with lower rates for low-income earners and higher rates for those with greater incomes, there are several brackets with varying rates. This system guarantees equitable distribution of tax responsibilities among several income levels. Income tax is handled and collected under the Australian Tax Office [2], [3]. It also guarantees that people and companies satisfy their tax liabilities and promotes compliance. Every year, people must submit a tax return declaring their annual income and any offsets or deductions they qualify for. Australian tax years run from July 1 to June 30. Most taxpayers must turn in their tax returns by the end of October, although in some cases, especially if someone is using a registered tax agent, then extensions are granted [4], [5].

In Australia, income tax covers several kinds of income. This covers salaries from work, company profits, share dividends, rental income, and government payments. Residents' income from outside Australia is also taxed. On the other hand, non-residents pay taxes based on their income generated in Australia [6], [7]. Those who make less than a particular level are exempt from income tax. This guarantees

that tax does not burden low-income earners too much and offers relief for those with minimum income. Determining a person's income tax depends critically on deductions and tax offsets. Deductions cut taxable income, reducing the tax due [8]. Typical deductions include investment costs, charitable organization donations, and job-related expenses. Direct tax offsets lower the tax owing directly and might result in a bigger tax refund. Low-income earners, seniors, and some people in particular circumstances have offsets available. These clauses seek to lower the tax load for qualified groups and bring the tax system fairer [9], [10].

Machine learning plays a transformative role in analyzing income tax data. It enables tax authorities to process vast datasets efficiently. It helps tax authorities to handle enormous databases quickly. Clear patterns in data help to enhance the procedures of decision-making [11]. Through tax data modeling, machine learning systems find fraud and anomalies. This greatly improves accuracy when spotting tax avoidance. Predicting taxpayer behavior benefits, mainly from machine learning models [12]. Given past data trends, algorithms can project tax income. This enables governments to distribute resources more wisely. Predictive modeling also points up audit risk groups. These realizations maximize the use of resources, so lowering unneeded audits and saving time [13]. Models of supervised learning effectively classify income sources. They enable one to differentiate between other income sources and consistent income. This difference helps one properly evaluate tax obligations. Additionally, it accelerates data processing and saves human effort through automated classification. Furthermore, helping in complicated computations like deduction estimation is machine learning [14].

The model of machine learning changes and grows with time. New data inputs help them to improve their accuracy. This iterative learning helps tax systems to be constantly improved. Advanced models spot developing trends in taxpayer actions. Before they go into general, they can spot fresh fraud strategies. Deep learning systems examine income tax data in high dimensions. These models often expose latent insights [15], [16]. Deep learning helps one to detect complex trends in consumer behavior. They favor accurate tax projections to improve government budget planning. Moreover, these realizations enable more exact policy changes. Machine learning helps tax authorities better handle data. Automation reduces administrative load and simplifies repetitious tasks [17], [18]. Algorithms guarantee correct results and lower error margins. They also provide better data security to help safeguard taxpayer data.

This work aims to create an exhaustive machine learning model for income tax collection employing nine contemporary ML methods. The study intends to forecast tax revenues precisely and identify important economic factors using three decades of historical data on income tax collection, population increase, and GDP trends. The uniqueness is in combining several ML techniques to examine long-term tax trends, offering strong predictive insights and practical forecasts. This method may help improve tax planning and policy decisions by providing a data-driven framework for maximizing tax collection in different economic environments and supporting more informed government resource allocation.

II. Materials and Method

A. Machine Learning Methods

This study employs nine machine learning methods to model income tax collection, drawing on three decades of data. Every method offers exceptional benefits in managing the historical trends and economic data pertinent to tax collecting. A brief discussion on each ML is presented:

1) Random Forest

Random Forest (RF) is an ensemble learning technique that builds multiple decision trees and combines them to improve predictive accuracy. It aggregates the predictions by building many trees, each trained on various random samples of the data [19]–[22]. Commonly occurring in single decision trees, overfitting is less likely to occur with this averaging process [23]. RF is beneficial for tax collecting models, including economic indicators like GDP, population, and historical tax revenue, since it can readily handle categorical and continuous variables. Apart from its predictive power [24], [25]. RF provides a means for feature importance ranking, guiding the identification of the most significant variables in tax revenue prediction. This quality is helpful in economic modeling, where knowledge of every element affects tax income collection. RF can show how GDP or population trends affect tax income employing variable importance, helping tax authorities make better decisions and allocate resources. Random Forest offers a strong framework in this work to capture intricate, non-linear interactions between economic variables across time, thus improving the general dependability of tax forecasts.

2) Gradient Boosting Regression

Gradient Boosting Regression (GBR) is a powerful machine learning technique that builds a predictive model sequentially by optimizing a loss function. GBR creates each tree to correct mistakes from the past trees, so producing a more accurate and refined model than conventional ML approaches, which build trees in parallel [26], [27]. GBR can be especially helpful in complex datasets such as income tax data, where economic factors may have non-linear or evolving relationships over time, by allowing GBR to concentrate on difficult-to-predict data points using this iterative approach [28], [29]. This approach is especially appropriate for income tax data since it can detect minor trends and nuances, such as how little GDP changes might affect tax collecting differently over time. Given that even small prediction mistakes can have major financial consequences, such accuracy can be quite important in tax data modeling. Furthermore, providing hyperparameters like learning rate and number of trees, GBR lets one fine-tune to maximize model performance. In this work, GBR will be employed to provide a sophisticated method to capture complex interactions in past tax data and significantly increase the model's predictive capability.

3) Decision Tree

Decision Tree (DT) is one of the simplest and most interpretable ML algorithms, making it an excellent starting point for income tax modeling. The technique forms a tree-like structure of decisions by separating the dataset depending on feature values [30]–[33]. Every split shows a decision

route whereby branches lead to results, representing how income levels, GDP, and population increase affect tax income. Transparency is a feature of decision trees that is especially helpful in economic modeling since it helps legislators better grasp the underlying reasons for tax projections. Although Decision Trees are quite interpretable, particularly in complex datasets, they are also prone to overfitting. Using decision trees as part of ensemble techniques like random forest and gradient boosting helps solve this restriction by aggregating several trees, thus reducing overfitting. Utilizing a hierarchical view of income tax factors. Decision Tree analysis—despite its simplicity—can provide insightful analysis of which variables have the most bearing on tax collecting. In this work, the Decision Tree acts as a baseline model, offering fundamental insights that guide more complex ensemble techniques and helps to build a useful framework for tax revenue prediction depending on economic data [34], [35].

4) Linear Regression

Linear regression (LR) is a fundamental machine-learning technique that model's relationships between variables. It identifies the linear relationship between an independent variable X and a dependent variable Y, so predicting a continuous output [36]–[38]. Finding the best-fit line that reduces the difference between expected and actual values drives LR. This line is computed through the equation $Y = b_0 + b_1X$, where b_0 is the intercept and b_1 is the coefficient of X [39], [40]. Under a technique known as "least squares," the LR model finds these values by minimizing the sum of squared deviations between observed and expected Y values. This method guarantees that the line reflects the data trend as faithfully as feasible. LR holds that X and Y have a linear relationship hence changes in X generate corresponding changes in Y [41], [42]. In cases when this linearity holds and is less complicated than other models, it is efficient. Based on past data, linear regression provides insights into trends and future values that can help forecast results, including costs or sales. It is limited, though, in managing non-linear patterns since these call for more sophisticated approaches [43], [44].

5) Lasso Regression

Lasso regression, or Least Absolute Shrinkage and Selection Operator, is an ML technique employed for regression analysis. It adds a penalty term depending on the absolute value of coefficients and minimizes the sum of squared residuals between actual and forecasted values. Designed as the "L1 regularization," this penalty term drives some coefficients to zero. Lasso thus does both regularizing and variable selection [45], [46]. It simplifies the model by zeroing some coefficients, so choosing the most pertinent features. It makes the model more generalizable by restricting the coefficients, preventing overfitting. This method works well when one wants a sparse model with fewer variables. Lasso can remove extraneous features, unlike ordinary least squares regression, which might include all features, so producing a more straightforward and more interpretable model [47]. It finds this equilibrium by varying the penalty strength across a hyperparameter, sometimes called lambda. One can adjust the degree of sparsity in the model by varying lambda, so optimizing it for prediction accuracy and interpretability over several datasets [48].

6) Ridge Regression

Ridge regression (RR) is instrumental in dealing with multicollinearity and overfitting in regression models. It makes use of a regularization technique to reduce the issue of overfitting. The least squares loss function is modified by adding a penalty term, which is the squared magnitude of the coefficients multiplied by a tuning parameter (λ), to the standard linear regression for improvement [49], [50]. To reduce the complexity of the model, larger values of λ shrink the coefficients toward zero to control the strength of regularization. This type of penalization stabilizes regression coefficients in cases where independent variables exhibit a high degree of correlation. RR improves generalizability by limiting the size of the coefficients, which lessens the model's sensitivity to errors in the training data. It is advantageous when predictor variables show multicollinearity because it more evenly distributes weight among correlated predictors, enhancing prediction and model interpretability [51], [52].

7) Adaptive Boosting Regression

To improve forecasting accuracy, adaptive boosting regression, also known as AdaBoost Regression, is a potent ensemble-type ML technique that builds a series of weak learners, usually decision trees, one after the other until it creates a strong predictive model [53], [54]. In contrast to traditional regression techniques, AdaBoost works to reduce error by repeatedly training learners to accord preference to data points where earlier models have had poor results. To draw attention to these complex cases, the algorithm gives instances with higher residual errors higher weights in each round [55]. Later, learners shift their focus, focusing on samples where the previous model's predictions were less accurate, while the initial weak learner is trained on the complete dataset. The final, more potent model is created by AdaBoost Regression by adding up all of the weak learners' predictions, weighted by each learner's accuracy. This technique is adaptive since it "boosts" performance by continuously learning from errors and improving its strategy with each iteration. AdaBoost can accomplish this by lowering bias and variance and producing a more accurate regression model even on intricate, non-linear data distributions. It is beneficial in situations where conventional models perform poorly because it makes use of the ensemble approach to improve prediction accuracy and stability [56], [57].

8) Extreme Gradient Boosting Regression

Extreme Gradient Boosting Regression (XGBoost) is a ML method that builds a series of decision trees to make accurate predictions [58], [59]. Every tree built by XGBoost aims to fix the errors of the past ones, progressively raising the accuracy of the model. XGBoost incorporates methods to prevent overfitting, which is when a model becomes overly complicated and performs poorly on new data but well on training data [60], [61]. Unlike some other algorithms. It uses parallel computing to speed calculations and adds penalty terms to regulate model complexity. XGBoost also gives great weight to data points that past trees missed, guiding the learning process. XGBoost can capture intricate patterns in data through this focused approach [62]. Its capacity to manage sparse data with missing or zero values, thus making it flexible for a broad spectrum of datasets. Combining these

techniques gives XGBoost a strong tool for regression tasks, where it is used to predict continuous values and is popular for its accuracy and speed in real-world applications [63], [64].

9) Categorical Boosting Regression

Categorical Boosting Regression, also called in short as CatBoost is an ML approach explicitly designed to handle categorical data efficiently. It works within gradient-boosting frameworks by internally managing the categorical features and converting them into numerical representations. Thereby preserving important relationships in the data, unlike some boosting techniques requiring significant preprocessing [65], [66]. This change employs target encoding with ordered boosting, a technique whereby only past events affect the encoding, lowering the probability of overfitting. Aiming to minimize the loss function iteratively, CatBoost creates a set of decision trees, each seeking to fix the error of previous trees. CatBoost's symmetric tree structure is a key feature since it guarantees balanced tree development and helps to reduce prediction times by so improving model stability [67], [68]. CatBoost also includes a dynamic regularizing method to lower overfitting and enhance generalization on fresh data. Many high-dimensional datasets where other gradient boosting techniques may find challenging to capture complex categorical patterns have Catboost become a preferred choice by combining fast performance, reduced need for extensive preprocessing, and effective handling of categorical data.

B. Data collection

For this study, GDP, population, and income tax collection data were sourced from the World Bank's database (*https://www.worldbank.org/en/about/annual-report*) and the Australian Bureau of Statistics (*https://www.abs.gov.au/*). Developing machine learning (ML) models would find the World Bank a suitable source since it offers consistent, thorough worldwide data that is routinely updated. Given their strong link with national income levels and economic capacity, GDP and population data were gathered as main predictors. Target variable income tax collection data directly relates to a nation's economic activities and demographic features by reflecting government revenue earned from taxes. Preprocessing the data guarantees consistency and quality; hence, carefully handling any missing or incomplete entries helps preserve strong model performance. As variables like GDP and population can have different scales across nations and years, the values were also standardized for comparability. A varied and representative dataset spanning several areas and economic situations could be obtained through World Bank data. This method sought to create a model that could provide significant forecasts for nations with different economic profiles and generalize effectively across many settings. Using World Bank data gave the study more legitimacy and guaranteed that the findings could be generally relevant and applicable to economic analysts and legislators.

III. Results and Discussion

A. Correlation among data

The correlation heatmap with an illustration of data scatter illustrates the relationships between population, GDP, and income tax collection in Figure 1. The correlation analysis results are listed in Table 1. With values near one throughout all pairs, the heatmap reveals strong positive correlations between these variables. Particularly, the population shows a strong correlation with GDP (0.95) and income tax collecting (0.96), implying that GDP and tax collecting usually rise as population size rises. With an almost perfect (1) correlation between GDP and income tax collecting, higher GDP is clearly strongly linked with higher income tax collecting. Every couple of variables in the scatter plots of the heatmap shows a linear trend. The linear patterns imply that these variables move proportionately to one another. For instance, income tax collection rises in line with GDP and other factors. The histograms plotted along the diagonal offer individual variances of each variable's distribution. GDP and income tax collection, which vary significantly over the dataset, show a broad range for every variable. This trend shows how closely population size and economic indicators affect tax revenues; each variable seems to support the others in this regard. Close correlations suggest that changes in GDP or population will affect income tax collection consistently.

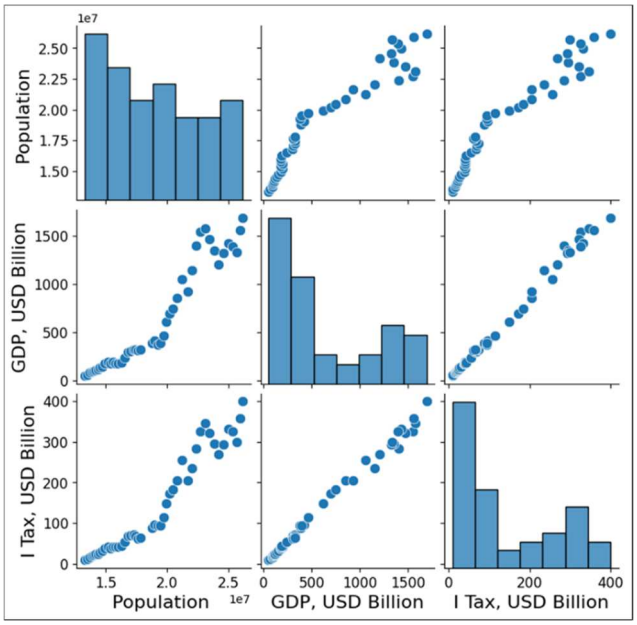

Fig. 1 Pair plots of data

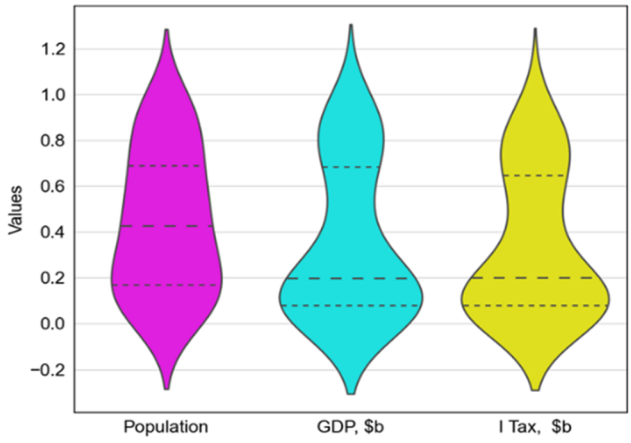

Fig. 2 Normalized violin plots for the data

TABLE I
CORRELATION MATRIX

	Population	GDP, USD Billion	I Tax, USD Billion
Population	1	0.95	0.96
GDP, USD Billion	0.95	1	1
I Tax, USD Billion	0.96	1	1

Scaled for comparability, the violin plot (Figure 2) shows the population, GDP, and income tax (I Tax) variances. Every violin displays the density of values over the range for every variable; wider sections indicate greater density and narrower sections indicate lower density. Represented in magenta, the population variable shows a concentration around the middle range, implying that most population values are modest, with fewer cases at very high or very low levels. Showed in cyan, the GDP variable has a clear bi-modal form that suggests clustering around two primary levels, with a narrower waist indicating lower density in the middle range. This suggests that GDP values tend toward the upper and lower ends and are less fairly distributed. Showing moderate concentration in the middle with less extreme values, the income tax variable in yellow has a distribution similar to that of the population. These forms, taken together, give a sense of the distinct distribution patterns for every variable, illuminating how population, GDP, and income tax vary across the dataset and maybe pointing up different underlying distributions or influencing factors for every variable.

2) Model development and comparison

Table 2 compares several ML-based models applied to predict income tax collection depending on GDP and population. The results include the Mean Squared Error (MSE) on both training and test data, R-squared (R^2) on both sets, and Mean Absolute Percentage Error (MAPE) for both sets. It is important to note that by ensuring the predictions on unseen data, one can evaluate the models' capacity to generalize from the training data to the test data by employing analytical analysis of these measures. With a Train MSE of 25.94 and a Train R^2 of 0.9983, the Random Forest model exhibits outstanding training performance, indicating a great degree of accuracy in fitting the training data.

TABLE II
COMPARATIVE EVALUATION OF ML-BASED MODELS

Model	Train MSE	Test MSE	Train R^2	Test R^2	Train MAPE, %	Test MAPE, %
Random Forest	25.94	51.00	0.9983	0.9958	2.85	5.53
Gradient Boosting	0.04	50.28	1.0000	0.9959	0.21	6.89
Decision Tree	0.00	180.50	1.0000	0.9853	0.00	9.10
Linear Regression	73.59	61.83	0.9951	0.9950	5.25	6.66
Lasso Regression	73.59	61.83	0.9951	0.9950	5.26	6.66
Ridge Regression	73.59	61.83	0.9951	0.9950	5.25	6.66
AdaBoost Regressor	51.41	124.56	0.9966	0.9899	9.45	8.05
XGBoost	0	101.68	1	0.9917	0	6.47
CatBoost	2.32	25.49	0.9998	0.9979	1.86	6.40

On the test set, the Test MSE of 51.00 and Test R^2 of 0.9958, however, show a small decline in performance that suggests some generalization ability but with minor overfitting. Reflecting the relative error rates of the model, the Train MAPE of 2.85% and Test MAPE of 5.53% show modest accuracy on the test data relative to the training data. Figure 3a depicts the comparison of actual and model forecasted values. Figure 4 depicts the model's statistical outcome.

Figure 3b compares actual and model projected values for the GBR model. Figure 4 shows the statistical result of the model. With a Train MSE of 0.04 and Train R^2 of 1, the GBR model performs nearly perfectly on the training set. With a Test MSE of 50.28 and Test R^2 of 0.9959, very near those of the RF model, this model performs poorly on the test set. However, the model's higher Test MAPE of 6.89% indicates a rather more significant relative error in the forecasts. This implies that although Gradient Boosting performs exactly for the training data, it might not be as exact in forecasting the test data as other models. With both Train MSE and Train R^2 values at 0.00 and 1, respectively, suggesting an excellent fit to the training data, the Decision Tree model (Figure 3c) exhibits remarkable performance in training metrics. With a Test MSE of 180.50 and a Test R^2 of 0.9853, far lower than other models, it does poorly on the test data. Since the Decision Tree model fails to generalize from the training data to the test data, the Test MAPE, at 9.10%, is also high and indicates severe overfitting.

Figure 3d compares actual and model projected values in the case of a based model. Figure 4 shows the model's statistically expected result. The LR model shows good general performance with a Train MSE of 73.59 and a Test MSE of 61.83. Train and Test R^2 values on both datasets are about 0.995, indicating good explaining ability. Its MSE values are higher than those of models such as Random Forest and Gradient Boosting, which suggest less exact predictions. Though with somewhat more error than the best models, the MAPE values, at 5.25% on the training set and 6.66% on the test set, show a relatively consistent performance across both sets. With Train MSE and Test MSE both at 73.59 and 61.83, respectively, and Train and Test R^2 values at 0.9951 and 0.9950, Lasso Regression and Ridge Regression models, as depicted in Figure 3e and Figure 3f yield almost precisely the same results to Linear Regression in this case. The MAPE values for training and testing are similar—roughly 5.25% and 6.66% respectively. Though without any performance benefits, these models treat the dataset similarly to Linear Regression, implying that regularization had little effect on outcomes for this dataset.

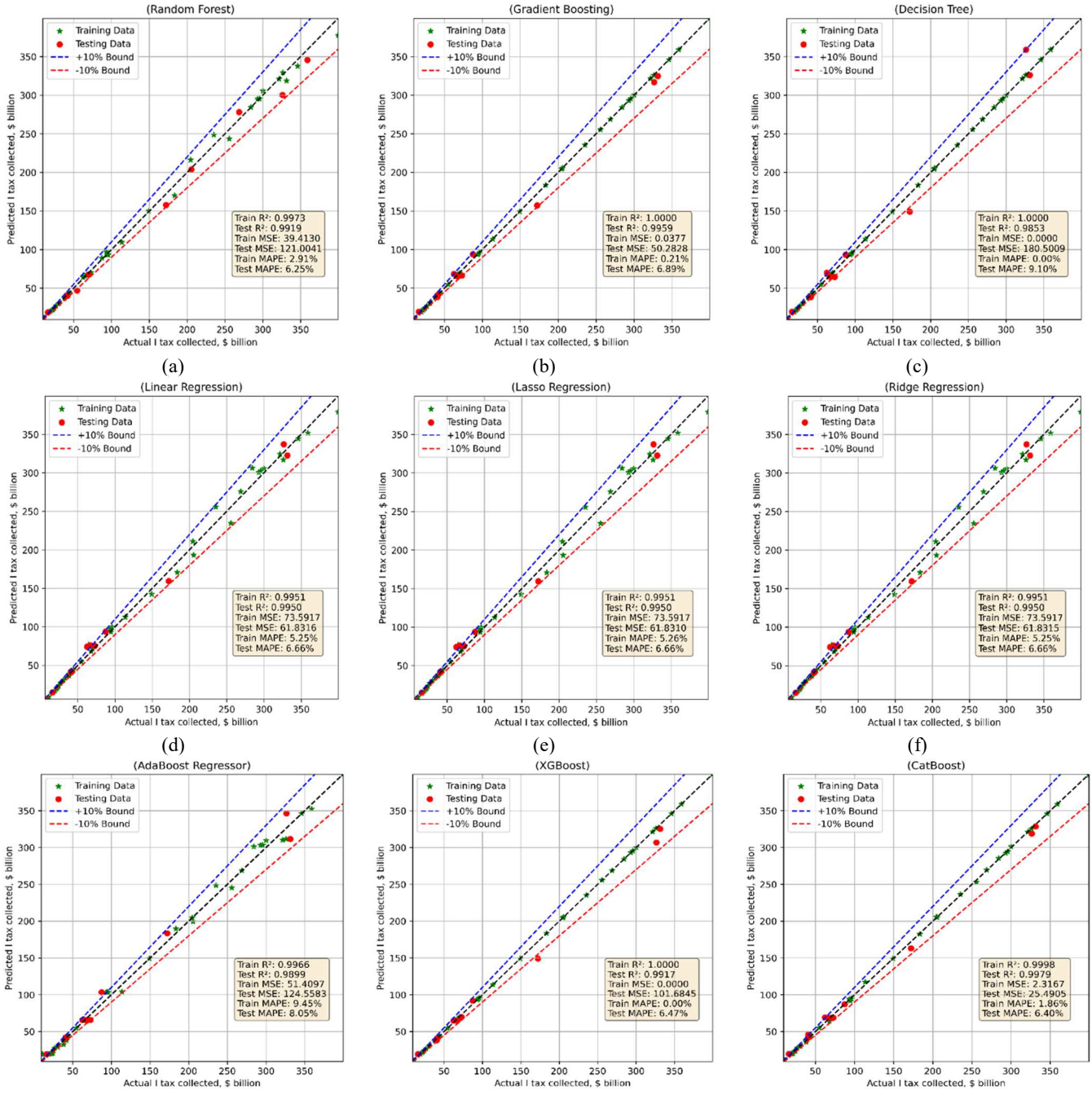

Fig. 3 Actual vs model forecasted Income Tax collected in the case of (a) RF (b) GBR (c) DT (d) LR (e) Lasso (f) (g) RR (h) Adaboost (i) CatBoost regression

Figure 3g compares actual and model projected values for the AdaBoost model. Figure 4 shows the model's statistically derived result. With a Train MSE of 51.41 and Train R^2 of 0.9966, the AdaBoost Regressor shows good training performance, but its test performance falls dramatically with a Train MSE of 124.56 and Train R^2 of 0.9899. AdaBoost's higher Test MAPE of 8.05% suggests it might be overfitting from this disparity. This model is less consistent with test data than other ensemble models, including Random Forest and Gradient Boosting. With both Train MSE and Train R^2 values at 0.00 and 1.0000, the XGBoost model (Figure 3h) perfectly fits the training data. With a Test MSE of 101.68, Test R^2 of 0.9917, and a Test MAPE of 6.47%, its performance is less ideal on the test set. XGBoost detects intricate trends in the training data, but its performance declines on the test set point to some degree of overfitting, though it is less severe than in the Decision Tree.

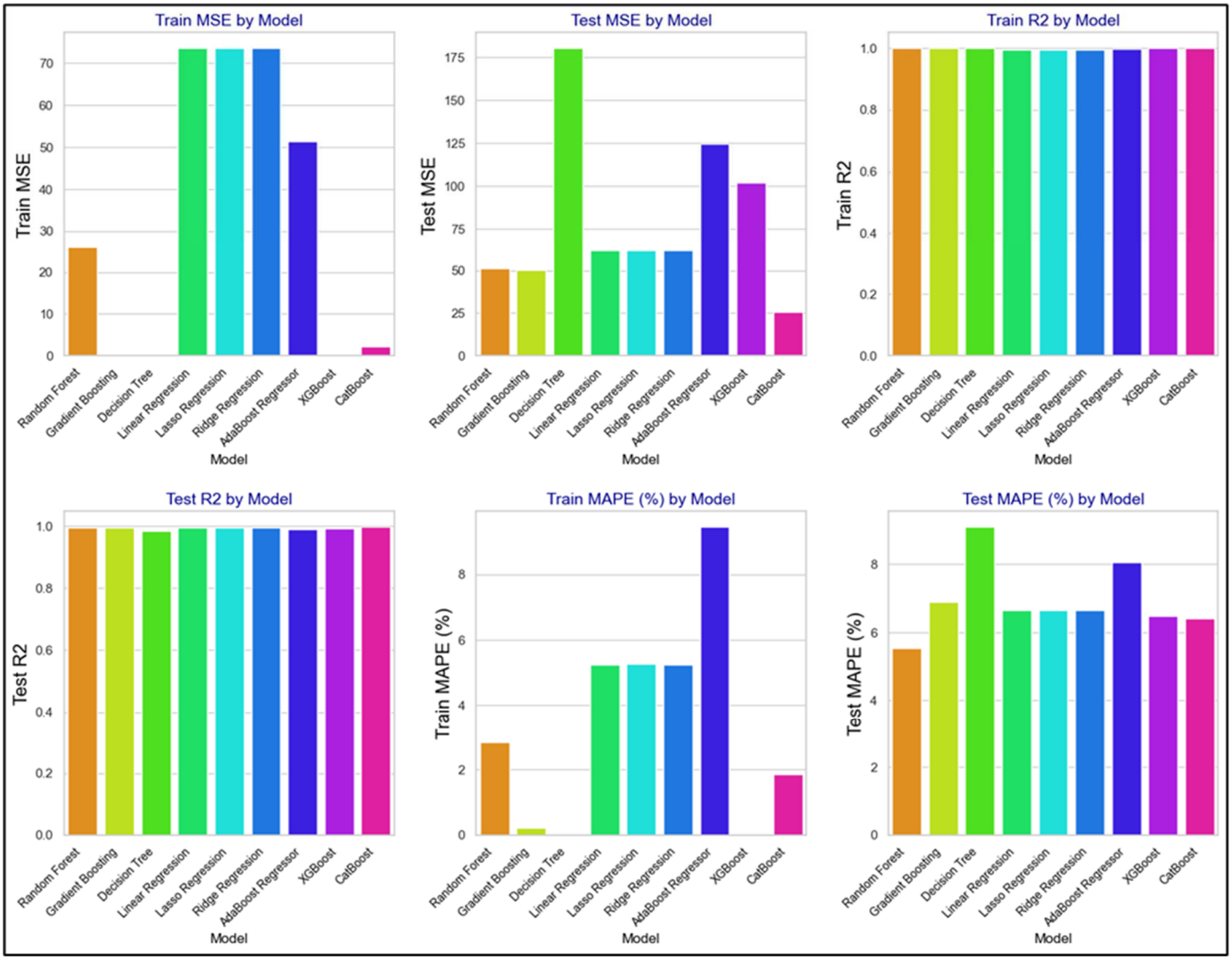

Fig. 4 Statistical comparison of developed models for (a) MSE training (b) MSE training (c) R2 train (d) R2 testing (e) MAPE train (f) MAPE testing

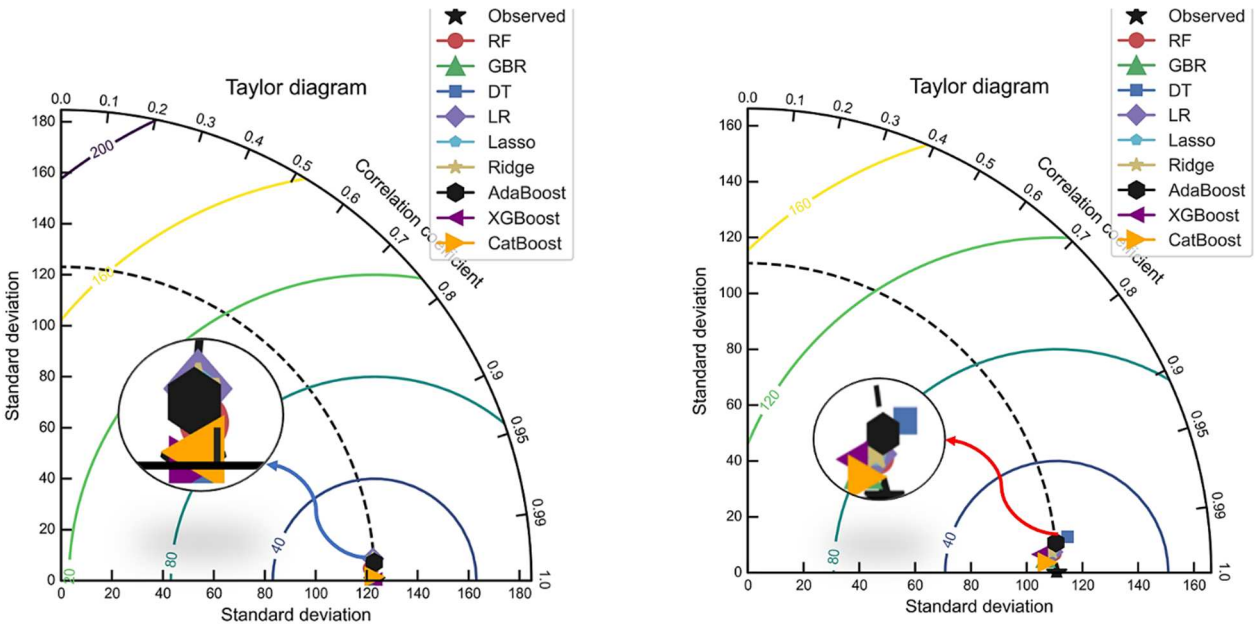

Fig. 5 Taylor diagram of developed models for (a) training and (b) testing phase

With a Train MSE of 2.32 and Train R^2 of 0.9998, the CatBoost model (Figure 3i) exhibits a balanced performance at last and a great fit to the training data. CatBoost shows great generalizing ability on the test set with a Test MSE of 25.49 and Test R^2 of 0.9979. With a Test MAPE of 6.40% among all models, it exhibits constant predictive accuracy and has among the lowest. The stability of CatBoost across training and testing criteria points to its efficient management of data complexity free from overfit. By contrast, the Decision Tree model exhibits extreme overfit with almost perfect training metrics but much worse test set performance. Although they lack the accuracy of advanced ensemble models, linear regression and its regularized forms (Lasso and Ridge) perform rather well. Though both Random Forest and Gradient Boosting exhibit good generalization, their Test MAPE values are somewhat higher than CatBoost. Though it shows more overfitting than Catboost, XGBoost also performs rather well.

CatBoost is the best model among all the ones since its balanced performance over all criteria is outstanding. Indicating it can generalize well; it achieves great accuracy on the test set without compromising training performance. Along with its reduced Test MAPE, CatBoost's lower Test MSE and high-Test R^2 show its capacity to more successfully manage intricate patterns in the dataset than other models. In this sense, CatBoost is the most appropriate model for estimating income tax collecting depending on GDP and population. Taylor's diagram was employed for model comparison, and these also corroborated the statistical evaluation of model in this Catboost was best model, as depicted in Figure 5.

IV. Conclusion

This machine learning-based study on income tax prediction reveals significant differences in model performance. Random Forest performed well, with a Train MSE of 25.94, Test MSE of 51.00, and Test MAPE of 5.53%. Gradient Boosting achieved a near-perfect Train MSE (0.04) but had a Test MAPE of 6.89%, indicating that its predictions were less accurate on new data. Decision Tree fit the training data exactly, with Train MSE of 0.00 and Train R^2 of 1.0000, but performed poorly on the test set, with Test MSE of 180.50 and Test MAPE of 9.10%, signaling severe overfitting. Linear, Lasso, and Ridge Regression models all had similar results, with Train MSE of 73.59 and Test MSE of 61.83, suggesting moderate accuracy but less precision than ensemble models. AdaBoost and XGBoost models both showed overfitting, with higher Test MAPE values at 8.05% and 6.47%, respectively. CatBoost achieved the best overall performance with a Train MSE of 2.32, Test MSE of 25.49, and Test MAPE of 6.40%. Its low error rates and consistent results across both training and test datasets indicate a strong ability to generalize without overfitting. Among all models, CatBoost emerges as the most suitable for predicting income tax collection based on GDP and population, due to its balanced accuracy and effective handling of complex data patterns.

References

[1] S. Nghiem and X.-B. (Benjamin) Vu, "Basic income in Australia: an exploration," *J. Econ. Dev.*, vol. 25, no. 4, pp. 365–376, Nov. 2023, doi: 10.1108/JED-07-2022-0119.

[2] E. Kirchler, A. Niemirowski, and A. Wearing, "Shared subjective views, intent to cooperate and tax compliance: Similarities between Australian taxpayers and tax officers," *J. Econ. Psychol.*, vol. 27, no. 4, pp. 502–517, Aug. 2006, doi: 10.1016/j.joep.2006.01.005.

[3] A. Fullarton and D. Pinto, "Australian Taxation Office Pronouncements: Why Tax Advisers Need to Exercise Caution," *SSRN Electron. J.*, 2022, doi: 10.2139/ssrn.4163859.

[4] A. Stokes and S. Wright, "Does Australia Have A Good Income Tax System?," *Int. Bus. Econ. Res. J.*, vol. 12, no. 5, p. 533, Apr. 2013, doi:10.19030/iber.v12i5.7828.

[5] M. K. Chan, T. Morris, C. Polidano, and H. Vu, "Income and saving responses to tax incentives for private retirement savings," *J. Public Econ.*, vol. 206, p. 104598, Feb. 2022, doi:10.1016/j.jpubeco.2021.104598.

[6] J. L. Hoopes, L. Robinson, and J. Slemrod, "Public tax-return disclosure," *J. Account. Econ.*, vol. 66, no. 1, pp. 142–162, Aug. 2018, doi: 10.1016/j.jacceco.2018.04.001.

[7] T. Sainsbury and R. Breunig, "Tax planning in Australia's income tax system," *Agenda - A J. Policy Anal. Reform*, vol. 27, no. 1, pp. 59–83, Dec. 2020, doi: 10.22459/AG.27.01.2020.03.

[8] J. Pope, "Reform of The Personal Income Tax System in Australia," *Econ. Pap. A J. Appl. Econ. policy*, vol. 24, no. 4, pp. 316–331, Dec. 2005, doi: 10.1111/j.1759-3441.2005.tb01006.x.

[9] R. V. Burkhauser, M. H. Hahn, and R. Wilkins, "Measuring top incomes using tax record data: a cautionary tale from Australia," *J. Econ. Inequal.*, vol. 13, no. 2, pp. 181–205, Jun. 2015, doi:10.1007/s10888-014-9281-z.

[10] A. Tran and Y. H. Zhu, "The impact of adopting IFRS on corporate ETR and book-tax income gap," in *Australian Tax Forum*, 2017, vol. 32, no. 4, pp. 757–792.

[11] A. Howard Miller, "Using unsupervised machine learning to model tax practice learning theory," *Int. J. Eng. Technol.*, vol. 7, no. 2.4, p. 109, Mar. 2018, doi: 10.14419/ijet.v7i2.4.13019.

[12] V. Baghdasaryan, H. Davtyan, A. Sarikyan, and Z. Navasardyan, "Improving Tax Audit Efficiency Using Machine Learning: The Role of Taxpayer's Network Data in Fraud Detection," *Appl. Artif. Intell.*, vol. 36, no. 1, Dec. 2022, doi: 10.1080/08839514.2021.2012002.

[13] N. A. Phong, P. H. Tam, and L. Q. Cuong, "Forecasting Tax Risk by Machine Learning: Case of Firms in Ho Chi Minh City," 2022.

[14] M. Z. Abedin, G. Chi, M. M. Uddin, M. S. Satu, M. I. Khan, and P. Hajek, "Tax Default Prediction Using Feature Transformation-Based Machine Learning," *IEEE Access*, vol. 9, pp. 19864–19881, 2021, doi:10.1109/ACCESS.2020.3048018.

[15] N. Ourdani, M. Chrayah, and N. Aknin, "Towards a new approach to maximize tax collection using machine learning algorithms," *IAES Int. J. Artif. Intell.*, vol. 13, no. 1, p. 737, Mar. 2024, doi:10.11591/ijai.v13.i1.pp737-746.

[16] Olatunji Akinrinola, Wilhelmina Afua Addy, Adeola Olusola Ajayi-Nifise, Olubusola Odeyemi, and Titilola Falaiye, "Application of machine learning in tax prediction: A review with practical approaches," *Glob. J. Eng. Technol. Adv.*, vol. 18, no. 2, pp. 102–117, Feb. 2024, doi: 10.30574/gjeta.2024.18.2.0028.

[17] R. Abdul Rahman, S. Masrom, N. Omar, and M. Zakaria, "An application of machine learning on corporate tax avoidance detection model," *IAES Int. J. Artif. Intell.*, vol. 9, no. 4, p. 721, Dec. 2020, doi:10.11591/ijai.v9.i4.pp721-725.

[18] B. F. Murorunkwere, D. Haughton, J. Nzabanita, F. Kipkogei, and I. Kabano, "Predicting tax fraud using supervised machine learning approach," *African J. Sci. Technol. Innov. Dev.*, vol. 15, no. 6, pp. 731–742, Sep. 2023, doi: 10.1080/20421338.2023.2187930.

[19] T. T. Le, H. C. Le, P. Paramasivam, and N. Chung, "Artificial intelligence applications in solar energy," *JOIV Int. J. Informatics Vis.*, vol. 8, no. 2, pp. 826–844, 2024, doi: 10.62527/joiv.8.2.2686.

[20] T. H. Nguyen, P. Paramasivam, H. C. Le, and D. C. Nguyen, "Harnessing a Better Future: Exploring AI and ML Applications in Renewable Energy," *JOIV Int. J. Informatics Vis.*, vol. 8, no. 1, pp. 55–78, 2024.

[21] Y. A. Seo and J. Cha, "Precipitation Probability Prediction through NWP Bias Correction for South Korea Using Random Forest," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 13, no. 3 SE-Articles, pp. 935–942, Jun. 2023, doi: 10.18517/ijaseit.13.3.18224.

[22] A. Ramadhan, B. Susetyo, and - Indahwati, "Classification Modelling of Random Forest to Identify the Important Factors in Improving the Quality of Education," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 11, no. 2 SE-Articles, pp. 501–507, Apr. 2021, doi:10.18517/ijaseit.11.2.8878.

[23] R. Susetyoko, E. Purwantini, B. N. Iman, and E. Satriyanto, "An Improved Accuracy of Multiclass Random Forest Classifier with

Continuous Attribute Transformation Using Random Percentile Generation," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 13, no. 3 SE-Articles, pp. 943–953, Jun. 2023, doi: 10.18517/ijaseit.13.3.18379.

[24] M. Gholizadeh, M. Jamei, I. Ahmadianfar, and R. Pourrajab, "Prediction of nanofluids viscosity using random forest (RF) approach," *Chemom. Intell. Lab. Syst.*, vol. 201, p. 104010, Jun. 2020, doi:10.1016/j.chemolab.2020.104010.

[25] L. Breiman, "Random Forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.

[26] P. Kumar K, M. Alruqi, H. A. Hanafi, P. Sharma, and V. V. Wanatasanappan, "Effect of particle size on second law of thermodynamics analysis of Al2O3 nanofluid: Application of XGBoost and gradient boosting regression for prognostic analysis," *Int. J. Therm. Sci.*, vol. 197, p. 108825, Mar. 2024, doi:10.1016/j.ijthermalsci.2023.108825.

[27] P. Nie, M. Roccotelli, M. P. Fanti, Z. Ming, and Z. Li, "Prediction of home energy consumption based on gradient boosting regression tree," *Energy Reports*, vol. 7, pp. 1246–1255, Nov. 2021, doi:10.1016/j.egyr.2021.02.006.

[28] R. Nasiboglu and E. Nasibov, "WABL method as a universal defuzzifier in the fuzzy gradient boosting regression model," *Expert Syst. Appl.*, vol. 212, p. 118771, Feb. 2023, doi:10.1016/j.eswa.2022.118771.

[29] T. Wang, S. Hu, and Y. Jiang, "Predicting shared-car use and examining nonlinear effects using gradient boosting regression trees," *Int. J. Sustain. Transp.*, vol. 15, no. 12, pp. 893–907, Oct. 2021, doi:10.1080/15568318.2020.1827316.

[30] A. J. Barid and H. Hadiyanto, "Hyperparameter optimization for hourly PM2.5 pollutant prediction," *J. Emerg. Sci. Eng.*, vol. 2, no. 1, p. e15, Apr. 2024, doi: 10.61435/jese.2024.e15.

[31] P. Paramasivama, K. Naima, and M. Dzida, "Soft computing-based modelling and optimization of NOx emission from a variable compression ratio diesel engine," *J. Emerg. Sci. Eng.*, vol. 2, no. 2, p. e21, Apr. 2024, doi: 10.61435/jese.2024.e21.

[32] M. Yanto, S. Arlis, M. R. Putra, H. Syahputra, and V. Ariandi, "Prediction of Drug Demand Based on Deep Learning Approach and Classification Model," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 13, no. 1 SE-Articles, pp. 357–364, Feb. 2023, doi:10.18517/ijaseit.13.1.17217.

[33] D. Puri, S. Nalbalwar, A. Nandgaonkar, J. Rajput, and A. Wagh, "Identification of Alzheimer's Disease Using Novel Dual Decomposition Technique and Machine Learning Algorithms from EEG Signals," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 13, no. 2 SE-Articles, pp. 658–665, Apr. 2023, doi: 10.18517/ijaseit.13.2.18252.

[34] S. B. Kotsiantis, "Decision trees: a recent overview," *Artif. Intell. Rev.*, vol. 39, no. 4, pp. 261–283, Apr. 2013, doi: 10.1007/s10462-011-9272-4.

[35] J. Abdi, F. Hadavimoghaddam, M. Hadipoor, and A. Hemmati-Sarapardeh, "Modeling of CO2 adsorption capacity by porous metal organic frameworks using advanced decision tree-based models," *Sci. Rep.*, vol. 11, no. 1, p. 24468, Dec. 2021, doi: 10.1038/s41598-021-04168-w.

[36] G. Shanmugasundar, M. Vanitha, R. Čep, V. Kumar, K. Kalita, and M. Ramachandran, "A Comparative Study of Linear, Random Forest and AdaBoost Regressions for Modeling Non-Traditional Machining," *Processes*, vol. 9, no. 11, p. 2015, Nov. 2021, doi: 10.3390/pr9112015.

[37] T. T. Le *et al.*, "Unlocking renewable energy potential: Harnessing machine learning and intelligent algorithms," *Int. J. Renew. Energy Dev.*, vol. 13, no. 4, pp. 783–813, Jul. 2024, doi:10.61435/ijred.2024.60387.

[38] N. S. I. Alsharabi, R. R. Al-Mola, R. E. Slewa Yonan, and Z. Y. Algamal, "Employing Several Methods to Estimate the Generalized Liu Parameter in Multiple Linear Regression Model," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 12, no. 6 SE-Articles, pp. 2386–2390, Dec. 2022, doi: 10.18517/ijaseit.12.6.14789.

[39] O. Vernanda *et al.*, "Correlation of Environmental Factors With Population of Horseshoe Crab (Tachypleus gigas) in Sedati Waters, Sidoarjo District," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 12, no. 2 SE-Articles, pp. 826–833, Apr. 2022, doi: 10.18517/ijaseit.12.2.14958.

[40] Y. Rahmawati, A. F. Sari, and C. Utomo, "The Effect of Consequences in Utilizing Real Estate Investment Trust (REIT) on Property Development," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 13, no. 1 SE-Articles, pp. 173–179, Jan. 2023, doi: 10.18517/ijaseit.13.1.16275.

[41] A. Balal, Y. Pakzad Jafarabadi, A. Demir, M. Igene, M. Giesselmann, and S. Bayne, "Forecasting Solar Power Generation Utilizing Machine Learning Models in Lubbock," *Emerg. Sci. J.*, vol. 7, no. 4, pp. 1052–1062, Jul. 2023, doi: 10.28991/ESJ-2023-07-04-02.

[42] F. E. Tahiri, K. Chikh, and M. Khafallah, "Optimal Management Energy System and Control Strategies for Isolated Hybrid Solar-Wind-Battery-Diesel Power System," *Emerg. Sci. J.*, vol. 5, no. 2, pp. 111–124, Apr. 2021, doi: 10.28991/esj-2021-01262.

[43] S. Pak and T. Oh, "Correlation and Simple Linear Regression," *J. Vet. Clin.*, vol. 27, no. 4, pp. 427–434, 2010.

[44] A. F. Schmidt and C. Finan, "Linear regression and the normality assumption," *J. Clin. Epidemiol.*, vol. 98, pp. 146–151, Jun. 2018, doi:10.1016/j.jclinepi.2017.12.006.

[45] Y. Li, R. Yang, X. Wang, J. Zhu, and N. Song, "Carbon Price Combination Forecasting Model Based on Lasso Regression and Optimal Integration," *Sustain. 2023, Vol. 15, Page 9354*, vol. 15, no. 12, p. 9354, Jun. 2023, doi: 10.3390/SU15129354.

[46] E. Ayyildiz and M. Murat, "A lasso regression-based forecasting model for daily gasoline consumption: Türkiye Case," *Turkish J. Eng.*, vol. 8, no. 1, pp. 162–174, Jan. 2024, doi: 10.31127/tuje.1354501.

[47] P. J. García-Nieto, E. García-Gonzalo, • José, and P. Paredes-Sá Nchez, "Prediction of the critical temperature of a superconductor by using the WOA/MARS, Ridge, Lasso and Elastic-net machine learning techniques," *Neural Comput. Appl.*, vol. 33, doi: 10.1007/s00521-021-06304-z.

[48] A. Kijkarncharoensin and S. Innet, "Consistent Regime-Switching Lasso Model of the Biomass Proximate Analysis Higher Heating Value," *Int. J. Renew. Energy Dev. Vol 12, No 1 January 2023DO - 10.14710/ijred.2023.47831* , Jan. 2023.

[49] L. Firinguetti-Limone and M. Pereira-Barahona, "Bayesian estimation of the shrinkage parameter in ridge regression," *Commun. Stat. - Simul. Comput.*, vol. 49, no. 12, pp. 3314–3327, Dec. 2020, doi:10.1080/03610918.2018.1547395.

[50] Y. Wu, N. Prezhdo, and W. Chu, "Increasing Efficiency of Nonadiabatic Molecular Dynamics by Hamiltonian Interpolation with Kernel Ridge Regression," *J. Phys. Chem. A*, vol. 125, no. 41, pp. 9191–9200, Oct. 2021, doi: 10.1021/acs.jpca.1c05105.

[51] A. Rokem and K. Kay, "Fractional ridge regression: a fast, interpretable reparameterization of ridge regression," *Gigascience*, vol. 9, no. 12, Nov. 2020, doi: 10.1093/gigascience/giaa133.

[52] I. S. Dar, S. Chand, M. Shabbir, and B. M. G. Kibria, "Condition-index based new ridge regression estimator for linear regression model with multicollinearity," *Kuwait J. Sci.*, vol. 50, no. 2, pp. 91–96, Apr. 2023, doi: 10.1016/j.kjs.2023.02.013.

[53] T. T. Le, J. C. Priya, H. C. Le, N. V. L. Le, T. B. N. Nguyen, and D. N. Cao, "Harnessing artificial intelligence for data-driven energy predictive analytics: A systematic survey towards enhancing sustainability," *Int. J. Renew. Energy Dev.*, vol. 13, no. 2, 2024, doi: 10.61435/ijred.2024.60119.

[54] J. Q. Yang and H. Z. Liu, "Application of EMD-Adaboost in wind speed prediction," *Int. J. Data Sci.*, vol. 7, no. 2, p. 164, 2022, doi:10.1504/ijds.2022.126854.

[55] S. Tsiapoki, O. Bahrami, M. W. Häckell, J. P. Lynch, and R. Rolfes, "Combination of damage feature decisions with adaptive boosting for improving the detection performance of a structural health monitoring framework: Validation on an operating wind turbine," *Struct. Heal. Monit.*, vol. 20, no. 2, pp. 637–660, Mar. 2021, doi:10.1177/1475921720909379.

[56] G. A. Busari and D. H. Lim, "Crude oil price prediction: A comparison between AdaBoost-LSTM and AdaBoost-GRU for improving forecasting performance," *Comput. Chem. Eng.*, vol. 155, p. 107513, Dec. 2021, doi: 10.1016/j.compchemeng.2021.107513.

[57] R. Li, H. Sun, X. Wei, W. Ta, and H. Wang, "Lithium Battery State-of-Charge Estimation Based on AdaBoost.Rt-RNN," *Energies*, vol. 15, no. 16, p. 6056, Aug. 2022, doi: 10.3390/en15166056.

[58] N. F. Rozam and M. Riasetiawan, "XGBoost Classifier for DDOS Attack Detection in Software Defined Network Using sFlow Protocol," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 13, no. 2, pp. 718–725, Apr. 2023, doi: 10.18517/ijaseit.13.2.17810.

[59] H. Darmawan, M. Yuliana, and M. Z. S. Hadi, "GRU and XGBoost Performance with Hyperparameter Tuning Using GridSearchCV and Bayesian Optimization on an IoT-Based Weather Prediction System," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 13, no. 3, pp. 848–859, 2023, doi: 10.18517/ijaseit.13.3.18377.

[60] Y. Qiu, J. Zhou, M. Khandelwal, H. Yang, P. Yang, and C. Li, "Performance evaluation of hybrid WOA-XGBoost, GWO-XGBoost and BO-XGBoost models to predict blast-induced ground vibration," *Eng. Comput.*, vol. 38, no. S5, pp. 4145–4162, Dec. 2022, doi:10.1007/s00366-021-01393-9.

[61] B. Akbar, H. Tayara, and K. T. Chong, "Unveiling dominant recombination loss in perovskite solar cells with a XGBoost-based

machine learning approach," *iScience*, vol. 27, no. 3, p. 109200, Mar. 2024, doi: 10.1016/j.isci.2024.109200.

[62] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2016, pp. 785–794, doi: 10.1145/2939672.2939785.

[63] P. Zhang, Y. Jia, and Y. Shang, "Research and application of XGBoost in imbalanced data," *Int. J. Distrib. Sens. Networks*, vol. 18, no. 6, p. 155013292211069, Jun. 2022, doi: 10.1177/15501329221106935.

[64] A. T. Le *et al.*, "Precise Prediction of Biochar Yield and Proximate Analysis by Modern Machine Learning and SHapley Additive exPlanations," *Energy & Fuels*, vol. 37, no. 22, pp. 17310–17327, Nov. 2023, doi: 10.1021/acs.energyfuels.3c02868.

[65] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: unbiased boosting with categorical features," *Adv. Neural Inf. Process. Syst.*, vol. 31, 2018.

[66] S. Ben Jabeur, C. Gharib, S. Mefteh-Wali, and W. Ben Arfi, "CatBoost model and artificial intelligence techniques for corporate failure prediction," *Technol. Forecast. Soc. Change*, vol. 166, p. 120658, May 2021, doi: 10.1016/j.techfore.2021.120658.

[67] Y. Zhang, Z. Zhao, and J. Zheng, "CatBoost: A new approach for estimating daily reference crop evapotranspiration in arid and semi-arid regions of Northern China," *J. Hydrol.*, vol. 588, p. 125087, Sep. 2020, doi: 10.1016/j.jhydrol.2020.125087.

[68] R. Banik and A. Biswas, "Improving Solar PV Prediction Performance with RF-CatBoost Ensemble: A Robust and Complementary Approach," *Renew. Energy Focus*, vol. 46, pp. 207–221, Sep. 2023, doi: 10.1016/j.ref.2023.06.009.