ComTech: Computer, Mathematics and Engineering Applications, 17(2), December 2026, 99−116 P-ISSN: 2087-1244
DOI: 10.21512/comtech.v17i2.14076 E-ISSN: 2476-907X

Evaluation of Tree-Based Ensemble Models Using Socioeconomic Data

Riechie[1]; Ferry Vincenttius Ferdinand[2]; Feliks Victor Parningotan Samosir[3]*

[1,2]Department of Mathematics, Faculty of Science and Technology, Universitas Pelita Harapan, Tangerang, Indonesia 15811
[3]Department of Informatics, Faculty of Information Technology, Universitas Pelita Harapan, Tangerang, Indonesia15811
[1]riechiemistery@gmail.com ; [2]ferry.vincenttius@uph.edu ; [3]feliks.parningotan@uph.edu

Received: 24th July 2025/ **Revised:** 12th January 2026/ **Accepted:** 28th January 2026

How to Cite: Riechie, Ferdinand, F. V., & Samosir, F. V. P. (2026). Evaluation of tree-based ensemble models using socioeconomic data. *ComTech: Computer, Mathematics and Engineering Applications, 17*(2), 99−116. https://doi.org/10.21512/comtech.v17i2.14076

Abstract **-** Air quality significantly impacts public well-being and environmental sustainability, with socioeconomic factors known to influence air quality levels. However, existing AQI prediction studies predominantly rely on meteorological or pollutant variables, while the predictive capability of socioeconomic indicators remains underexplored. This study evaluates the effectiveness of tree-based ensemble learning techniques, including bagging, boosting, stacking, and multi-layer stacking, to predict air quality index (AQI) in Singapore using monthly socioeconomic indicators (34 variables) collected in Singapore from April 2014 until July 2024, encompassing transportation, tourism, industrialization, and price-related variables. Various tree-based algorithms, including Random Forest, ExtraTrees, XGBoost, and LightGBM, were evaluated. Stacking and multi-layer stacking models were also implemented with Linear Regression as the meta-learner. Feature engineering guided by the Pearson correlation coefficient was employed to optimize input selection. To optimize performance, hyperparameter tuning and model architectures were explored. The stacking model achieved the best predictive performance, with MAE, RMSE, and MAPE values of 13.68, 22.73, and 0.1359, respectively. Bagging models outperformed boosting models on the small dataset, while multi-layer stacking added computational complexity without significant performance gains. The results indicate that stacking captures complementary predictive patterns in socioeconomic data more effectively than deeper ensemble architectures, which offer limited benefits under data-constrained conditions. Furthermore, the findings show that socioeconomic variables provide sufficient predictive information to support AQI forecasting. Overall, this study highlights the potential of integrating socioeconomic data and offers methodological insights for future air quality research and applications.

Keywords**:** air quality, socioeconomic, ensemble learning, bagging, boosting

I. INTRODUCTION

Air quality is a critical determinant of human health and environmental sustainability, with escalating air pollution posing severe threats to public well-being and ecosystems worldwide (Manisalidis et al., 2020). To quantify and communicate air quality levels and their potential health impacts, the Air Quality Index (AQI) has been widely adopted as a standardized metric. Among the potential factors that affect AQI are socioeconomic factors, such as economic growth, urbanization, industrial activity, and demographic trends, are reported to significantly influence regional air pollution exposure (Jiao et al., 2018). However, the integration of these socioeconomic aspects into AQI prediction models remains underexplored, despite their potential to reveal deeper insights into the determinants of air quality variations.

The advancement of machine learning for analyzing real-world datasets has revolutionized predictive modeling, offering unprecedented accuracy and efficiency, as evidenced by the use of machine learning in research by Issah et al. (2023), Putri et al. (2023), and Gafarov et al. (2022). Among these, ensemble learning techniques, such as bagging, boosting, and stacking, have emerged as powerful tools to enhance model performance by combining the

*Corresponding Author

strengths of multiple base learners (Mienye & Sun, 2022). While considerable progress has been made in predicting AQI using machine learning models trained on meteorological and pollutant data, as shown by Méndez et al. (2023), Mao et al. (2021), and Gurumoorthy et al. (2023), the role of socioeconomic aspects as predictors in these models has not been sufficiently investigated.

Thus, a clear research gap exists: despite the growing use of machine learning for AQI prediction, the predictive capability of socioeconomic indicators remains insufficiently examined. This study addresses this gap by evaluating advanced tree-based ensemble methods for AQI prediction using socioeconomic data as input features. Singapore is selected as the case study due to the availability of extensive and publicly accessible socioeconomic datasets. The novelty of this work lies in its systematic evaluation of tree-based ensemble models with varying levels of complexity using socioeconomic data alone, providing empirical insights into their effectiveness for AQI prediction under data-constrained conditions. In doing so, the study contributes methodological evidence on the suitability of ensemble learning strategies for socioeconomic-based AQI forecasting, with implications for data-driven air quality monitoring and urban policy support.

The literature on tree-based ensemble learning methods will be reviewed to justify their use in this study further. Tree-based ensemble learning algorithms, such as Random Forest (RF), Gradient Boosting Decision Trees (GBDT), and XGBoost (XGB), have been extensively used in prediction tasks due to their ability to capture complex nonlinear relationships and handle high-dimensional datasets (Hu & Li, 2022; Zhang et al., 2022). These characteristics are highly relevant to socioeconomic datasets, which typically comprise diverse indicators with complex, non-additive interactions.

Several studies from non-AQI domains further demonstrate the suitability of these models for structured, tabular data. Ghiasi and Zendehboudi (2021) demonstrated the effectiveness of bagging algorithms (RF and ExtraTrees (ET)) in breast cancer classification, achieving superior diagnostic performance and highlighting their robustness to noise and limited sample sizes. Yoon et al. (2023) showed that boosting-based models, such as XGB and LightGBM (LGB), outperformed traditional regression approaches for predicting plant metabolites from hyperspectral data, highlighting their effectiveness at extracting predictive signals from complex, multivariate inputs. Saied et al. (2023) evaluated tree-based ensemble methods for network intrusion detection, finding that RF achieved near-perfect accuracy (0.999991), with other methods such as Bagging Meta Classifier (BMC) and ADB also performing strongly. Although these studies are conducted outside the AQI domain, the underlying data characteristics closely resemble those of socioeconomic datasets used for air quality prediction.

In addition to bagging and boosting, stacking has emerged as a powerful ensemble learning technique that combines predictions from multiple base models through a meta-model to enhance accuracy and robustness (Menshawy et al., 2023). Stacking has proven effective across various domains, including health diagnostics, environmental modeling, and sentiment analysis, where complex variable interactions demand advanced predictive frameworks, as shown by Kumar et al. (2022), Abayomi-Alli et al. (2022), Koopialipoor et al. (2022), and Lin et al. (2022). Tree-based stacking ensembles, in particular, often outperform other tree-based ensemble methods. For example, Alazba and Aljamaan (2022) demonstrated the superiority of a stacked ensemble model for software defect prediction, utilizing fine-tuned tree-based ensembles. Similarly, Rashid et al. (2022) showed that a tree-based stacked model outperformed individual models in network intrusion detection. Extending this approach, Shafieian and Zulkernine (2023) proposed a multi-layer stacked ensemble model for network intrusion detection, achieving an exceptional F1-score of 0.99, the highest among all evaluated models.

As mentioned previously, the utilization of machine learning for AQI prediction has become increasingly widespread. Kumar and Pande (2023) analyzed air pollution data from 23 Indian cities. They found that the XGB model outperformed other machine learning methods, a finding further supported by Van et al. (2023), who demonstrated XGB's effectiveness in predicting AQI across different regions in India. Similarly, Ravindiran et al. (2023) compared several tree-based models, including RF, XGB, LGB, CatBoost (CB), and ADB, and found that the CB model achieved the highest performance, attaining an R^2 of 0.9998. These studies confirm the suitability of tree-based ensemble learning methods for AQI prediction; however, most rely on meteorological and pollutant datasets, overlooking the established link between air quality and socioeconomic factors, as highlighted by Wang et al. (2022) in the United States and China.

To address the limitations of prior AQI studies that primarily rely on meteorological and pollutant variables, this study investigates the use of socioeconomic data as alternative predictive inputs within a tree-based ensemble learning framework. The analysis is conducted using monthly socioeconomic indicators collected in Singapore, covering transportation, tourism, industrial activity, and price-related factors. Bagging methods, including RF and ET, and boosting methods, such as XGB and LGB, will be employed. To further enhance predictive performance, stacking and multi-layer stacking will be implemented, utilizing the previous tree-based algorithms as base learners and Linear Regression (LR) as the meta-learner. Feature engineering will be guided by the Pearson Correlation Coefficient (PCC) to reduce dataset complexity, ensuring that the models effectively capture relationships between socioeconomic factors and air quality. Accordingly,

the objective of this study is to evaluate the predictive performance of different tree-based ensemble learning strategies for AQI forecasting using socioeconomic data, with Singapore serving as a case study.

II. METHODS

The dataset used in this study comprises Singapore's AQI values as the target variable and 34 socioeconomic variables as predictors. Unlike many regions that use the standard AQI, Singapore employs the Pollutant Standards Index (PSI), a modified version tailored to its local air quality monitoring needs. For consistency, the term PSI will be used throughout this paper in place of AQI when referring to Singapore's air quality values. The socioeconomic variables can be categorized into four main aspects: Transportation, Tourism, Industrialization, and Economy & Prices. The dataset spans April 2014 until July 2024. The PSI data was obtained from data.gov.sg (National Environment Agency, 2025), and all socioeconomic variables were sourced from the SingStat Website (Department of Statistics Singapore, 2025). Model implementations were conducted in Python 3.10.12 using Google Colab.

Prior to inputting the dataset into the machine learning models, a series of pre-processing steps are performed to ensure data quality, validity, and readiness for analysis. First, Data Cleaning is performed. The PSI dataset consists of air quality values from five regions in Singapore: North, North-East, East, West, and Central. To represent the overall PSI value for Singapore on a given date, the maximum value among the five regions is selected. While the PSI dataset was initially collected at an hourly frequency, the socioeconomic variables are available monthly. To address this discrepancy in data frequency, the PSI dataset was aggregated to a monthly level, using the maximum hourly PSI value for each month to represent the monthly PSI value. The resulting dataset consists of 124 rows of data instances and 35 columns. The socioeconomic variables collected for this study are presented in Table 1 (see Appendices).

Second, Feature Engineering is performed. High-dimensional datasets often suffer from multicollinearity among input variables, where certain features may carry redundant information. Although tree-based models are relatively robust to multicollinearity, addressing this issue remains a valuable step to improve dataset validity, reduce model complexity, and enhance interpretability through feature engineering techniques (Sharma et al., 2024; Singh et al., 2022). Previous studies have demonstrated that utilizing the Pearson Correlation Coefficient (PCC) to guide feature engineering or feature selection can effectively improve model accuracy and reduce computational time (S et al., 2022; Zhiyong et al., 2019). Given a set of samples $\{(x_1, y_1), \ldots, (x_m, y_m)\}$, where m denotes the number of observations, the formula of the sample PCC, denoted as $r_{x,y}$, is given by Equation (1),

$$r_{x,y} = \frac{\sum_{i=1}^{m}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{m}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{m}(y_i - \bar{y})^2}} \quad (1)$$

where $\bar{x}$ and $\bar{y}$ represent the means of variables x and y, respectively. In this study, the feature engineering process involves mathematical operations such as addition, subtraction, multiplication, and division of two or more input variables. The goal is to reduce the PCC values among the input variables while strengthening their correlation with the target variable, PSI. The criteria for this process are as follows: the PCC between any pair of input variables must not exceed 0.9, and the PCC between each input variable and the target variable (PSI) must not fall below 0.1. This procedure is applied to each category of the socioeconomic variables, ensuring that each category retains at least two representative variables. This process successfully reduces the number of input variables from 34 to 13 while increasing their correlation with PSI. Figure 1 (see Appendices) illustrates the PCC heatmap of the original feature set. At the same time, the resulting feature-engineered variables, along with their corresponding mathematical operations and absolute PCC values relative to PSI, are summarized in Table 2 (see Appendices).

Finally, data scaling is performed using normalization in this study to ensure that all input variables are on the same scale. Normalization, also known as min-max scaling, maps the values of a variable to the range [0,1]. Let x_i^j represent the i-th instance of the j-th input variable, and $\vec{x}^j = \{x_1^j, \ldots, x_m^j\}$ denote the set of all values for the j-th input variable. The maximum and minimum values of $\vec{x}^j$ are denoted as $\max(\vec{x}^j)$ and $\min(\vec{x}^j)$, respectively. The formula to compute the normalized value for each instance is given by Equation (2),

$$x_i^j = \frac{x_i^j - \min(\vec{x}^j)}{\max(\vec{x}^j) - \min(\vec{x}^j)} \quad (2)$$

Ensemble learning is a method that combines the predictions of multiple machine learning models, referred to as base learners or learners, to solve the same task. By aggregating the outputs of individual models, ensemble learning aims to reduce variance, improve predictive performance, and enhance the overall robustness and generalization ability of the model (Saied et al., 2023). Tree-based ensemble methods involve using Decision Tree (DT) models as base learners within the ensemble framework. A DT is a machine learning model structured hierarchically like a tree, employing a top-down, greedy approach to split data based on feature values. The model comprises a root node (starting point), interior nodes (decision points), and leaf nodes (terminal nodes), all connected by branches. Each leaf node represents the model's final prediction.

There are various methods for combining the predictions of base learners in ensemble learning. The simplest approaches involve voting for classification tasks and averaging for regression tasks. However, more advanced ensemble techniques, such as bagging, boosting, and stacking, provide significant improvements by enhancing model performance and robustness. These advanced techniques will be discussed in detail in the following sections.

Bootstrap Aggregating (Bagging) is an ensemble learning technique that leverages bootstrap sampling and aggregation to improve predictive performance and reduce variance. Given T base learners, bagging generates T bootstrap samples, each a random subset drawn with replacement from the original dataset. Each bootstrap sample is then used to train a base learner in parallel. The predictions from all base learners are subsequently aggregated, using voting for classification tasks or averaging for regression tasks, to produce the final output. The tree-based bagging algorithms employed in this study are Random Forest (RF) and ExtraTrees (ET). This selection is informed by the work of Shafieian and Zulkernine (2023), who used these algorithms as representative bagging methods in their research.

Random Forest (RF) is an ensemble learning method that combines multiple Decision Trees (DTs) using the bagging approach. DTs, while effective, are prone to high variance, and the primary objective of RF is to mitigate this variance by aggregating the predictions of multiple trees. RF is particularly well-suited for unstable base learners, where individual models are sensitive to changes in the training data. However, when all DTs are trained on the same set of input variables, their predictions can become highly correlated, reducing the effectiveness of the ensemble. To address this, RF introduces a technique to decorrelate the trees by limiting the number of input variables considered at each node split. This strategy prevents a small subset of predictors from dominating the tree structure, resulting in more diverse and robust predictions.

Extremely Randomized Trees (ExtraTrees, ET) is a variation of the RF algorithm, distinguished by two primary differences. First, in RF, node splitting is determined by selecting the split points that maximize the purity of the resulting regions. In contrast, ET performs node splitting by randomly selecting the split points, introducing additional randomness to the model. Second, unlike RF, which employs bootstrap sampling to generate subsets of the training data, ET uses the entire dataset to train each DT model. These differences make ET computationally more efficient while enhancing its ability to capture diverse patterns in the data.

Boosting is an ensemble learning technique that sequentially combines multiple base learners, with each subsequent learner focusing on instances where the previous model made the largest errors. A widely used variant of boosting is gradient boosting, which computes residuals and their gradients to optimize model performance iteratively. The tree-based gradient boosting algorithms utilized in this study are XGBoost (XGB) and LightGBM (LGB). XGB was chosen for its demonstrated efficiency and consistent performance, as evidenced by its strong track record in numerous machine learning competitions, including Kaggle (Chen & Guestrin, 2016). LGB was selected for its superior ability to handle high-dimensional datasets and its computational efficiency in both speed and memory usage, often surpassing XGB in these aspects (Ke et al., 2017).

eXtreme Gradient Boosting (XGBoost, XGB) is an advanced and efficient implementation of the gradient boosting algorithm that incorporates regularization parameters to enhance the model's generalization. This improvement is achieved by modifying the objective function of the traditional gradient boosting framework. Given a dataset with m instances, the regularized objective function for XGB is defined by Equation (3),

$$L_t\left(f_t(x_i)\right) = \sum_{i=1}^{m} L\left(y_i, f_t(x_i)\right) + \sum_{i=1}^{T} \Omega(f_t), \quad (3)$$

where $f_t(x_i)$ is the prediction of the i-th instance using t trees, $L(*)$ represents the loss function, and $\Omega(f_t)$ is the regularization parameter. Additionally, XGB is designed to handle sparse data, supports parallelized computations for improved efficiency, and mitigates the risk of overfitting through advanced pruning techniques. These characteristics enable XGBoost to maintain strong predictive accuracy and robustness across diverse data conditions.

The Light Gradient Boosting Machine (LightGBM, LGB) is a gradient boosting algorithm that employs Decision Trees (DTs) as base learners and is optimized for speed and memory efficiency through a histogram-based learning approach. Unlike other tree-based models that typically rely on level-wise tree growth, LGB uses a best-first strategy that prioritizes the most significant splits. Furthermore, LGB incorporates advanced techniques such as Gradient-Based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB) to reduce computational complexity and enhance overall efficiency. GOSS prioritizes training instances with the largest gradients to ensure efficient learning, while EFB bundles mutually exclusive features, effectively reducing data dimensionality without sacrificing information. These features make LGB particularly well-suited for large-scale and high-dimensional datasets.

Stacked Generalization (Stacking) is an ensemble learning technique that trains a new learner, referred to as the meta-learner, to combine the outputs of the base learners, also known as first-level learners. Unlike bagging and boosting, which typically utilize homogeneous base learners, stacking often employs heterogeneous base learners (i.e., models from different algorithms), leveraging their complementary strengths to improve predictive performance. To mitigate the risk of overfitting during meta-learner training, a k-fold cross-validation approach is employed, with

k set to 10 in this study. The meta-learners will then use out-of-fold predictions from the base learners as inputs. Figure 2 (see Appendices) illustrates the general framework of the stacking method.

A variation of the stacking method is the multi-layer stacking approach, which consists of more than two layers of learners. In this model, the predictions generated by each layer serve as input variables for the subsequent layer. The final layer comprises a single meta-learner that combines the outputs of the preceding layer to produce the final prediction. Figure 3 (see Appendices) illustrates the general framework of a multi-layer stacking model with *P* layers.

Theoretically, any machine learning algorithm can be employed as a learner in any layer of a stacking model. However, to align with the objective of this study of analyzing the comparative performance of tree-based ensemble learning methods, the learners in this study are restricted to tree-based algorithms, including RF, ET, XGB, and LGB. An exception is made for the meta-learner, which utilizes Linear Regression (LR), as supported by the findings of Alazba and Aljamaan (2022).

The evaluation metrics used in this study includes Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE). Let m represent the number of instances to predict y_i,denote the actual output of the i-th instance, and $h(x_i)$ indicate the predicted output of the i-th instance. Then, Equations (4) to (6) calculates MAE, RMSE, and MAPE respectively:

$$MAE = \frac{1}{m}\sum_{i=1}^{m}|y_i - h(x_i)|, \quad (4)$$

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(y_i - h(x_i)\right)^2}, \quad (5)$$

$$MAPE = \frac{1}{m}\sum_{i=1}^{m}\left|\frac{y_i - h(x_i)}{y_i}\right|. \quad (6)$$

The performance of a machine learning model is highly dependent on the selection of its hyperparameters; therefore, this study conducts a comprehensive analysis of the impact of hyperparameter tuning on each model's performance. For the stacking model, the primary hyperparameters to be tuned involve the architecture of the model, including the selection of learners for each layer. The hyperparameters for the first-level learners are set to their optimal values for each model, while the second-level learner is further fine-tuned. The hyperparameters specific to each tree-based model are detailed in Table 3 (see Appendices), while the architectures to be explored for the stacking and multi-layer stacking models are presented in Table 4 (see Appendices).

Typically, datasets are divided into fixed training and testing sets to evaluate a model's performance. However, due to the limited size of the dataset in this study, 10-fold cross-validation will be employed to maximize data utilization and identify more robust models, as recommended by Allgaier and Pryss (2024). Although the dataset is indexed over time, this study does not adopt a time-series modeling framework; instead, each observation is treated as an independent instance representing a snapshot of socioeconomic conditions to evaluate the predictive capability of socioeconomic indicators rather than modeling temporal dependencies. Under this assumption, standard cross-validation is considered appropriate for model evaluation. For each configuration of the stacking and multi-layer stacking models, LR, RF, ET, XGB, and LGB will be used as meta-learners. Model performance for each hyperparameter combination will be assessed using a grid search. The evaluation criteria include predictive performance (based on the mean of the predefined evaluation metrics), model stability (measured by the standard deviation (SD) across folds), and computational efficiency (evaluated through fit time in seconds). Model stability refers to the consistency of a model's predictive performance across different folds of the dataset.

III. RESULTS AND DISCUSSIONS

This section presents the key findings obtained in this study. First, we analyze the impact of various hyperparameters on model performance, then we provide a comparative evaluation of the best-performing configurations for each model. To illustrate the effects of hyperparameter tuning, performance metrics are aggregated and averaged for each hyperparameter value, enabling clear visualization of its influence on model performance.

Figure 4 (see Appendices) illustrates the impact of various hyperparameters on the performance of the RF model. This analysis initially focuses on the first three rows of the figure, with the fourth row addressed later. The first row shows the average MAE, the second row the average SD MAE, and the third row the average fit time. The most influential hyperparameters for predictive performance and stability are min_samples_leaf and criterion, while n_estimators has the greatest impact on training time. As max_depth increases, MAE worsens, whereas increasing min_samples_leaf reduces MAE, suggesting that simpler models perform better on this small dataset, as complex models may overfit. This suggests that overly complex tree structures may overfit socioeconomic features that already embed aggregated temporal and structural information.

The criterion reveals that absolute error outperforms squared error in predictive performance but at the cost of reduced stability and increased training time. For n_estimators, performance plateaus beyond 100 trees, while training time rises significantly. Lastly, limiting features via max_features improves accuracy and stability and reduces training time, indicating that aggressive feature usage may introduce noise rather than additional predictive signal.

The impact of hyperparameters on the mean and SD MAPE of the RF model closely mirrors the patterns observed for MAE and will not be shown.

Similarly, mean RMSE exhibits trends comparable to MAE and MAPE, but slight differences are observed in SD RMSE, as illustrated in the fourth row of Figure 4 (see Appendices). Specifically, increasing min_samples_leaf raises SD RMSE, while using all features minimizes SD RMSE. These differences arise because RMSE is more sensitive to large errors, indicating that simpler bagging models, though generally more accurate, are more susceptible to extreme deviations, a property that is particularly relevant when modeling AQI using indirect socioeconomic proxies. The effects of hyperparameters on the ET model are nearly identical to those of RF and thus are not visualized separately, as Figure 4 (see Appendices) sufficiently represents the trends for both models.

Figure 5 (see Appendices) illustrates the impact of various hyperparameters on the performance of the XGB model. The first row presents the average MAE, the second the average SD MAE, and the third the average mean fit time. Among the hyperparameters, max_depth is the most influential, with larger values significantly increasing mean MAE and training time. The best performance is observed with 100 trees, indicating that, as with bagging models, simpler XGB configurations improve predictive performance on socioeconomic data but may reduce stability. A smaller learning rate yields superior predictive performance and stability, though at the cost of increased training time due to slower convergence. For regularization parameters, lower reg_alpha and higher reg_lambda reduce errors; reg_lambda = 1.5 yields the best performance.

Figure 6 (see Appendices) shows the impact of hyperparameters on LGB model performance, with the rows corresponding to those in Figure 6 (see Appendices). The results reinforce the preference for simpler models, where a larger minimum child samples and a smaller number of leaves result in lower errors, improved stability, and reduced computational time. Notably, discretizing input variables into smaller bins does not benefit the model and may omit valuable information. Other evaluation metrics for XGB and LGB exhibit similar patterns and are therefore omitted.

Overall, these results indicate that when socioeconomic variables are used as the sole predictors, ensemble models with restrained complexity offer a more favorable balance among accuracy, stability, and efficiency. This finding underscores the importance of aligning model capacity with data characteristics, particularly in socioeconomic-based AQI prediction settings where data availability and granularity are inherently limited.

The combination of first-level learners does not exhibit a consistent or dominant effect on the stacking model's predictive performance or stability. This suggests that the overall architecture of the stacking model has a greater influence on its performance than the specific choice of first-level learners. However, the combination of XGB and LGB as first-level learners achieves the fastest training time, as illustrated in Figure 7 (see Appendices). This suggests that homogeneous boosting-based configurations may offer efficiency advantages without materially affecting predictive outcomes.

Figure 8 (see Appendices) shows the impact of the meta-learner on the stacking model's predictive performance. Bagging models (RF and ET) yield lower mean errors across all evaluation metrics compared to LR and boosting models (XGB and LGB). However, as shown in Figure 9 (see Appendices), they also exhibit the highest SD on MAE and RMSE, highlighting a trade-off between accuracy and stability when selecting the meta-learner. The training time of the stacking model does not vary significantly across meta-learners and is therefore omitted. The similarity in training time across meta-learners indicates that accuracy–stability considerations, rather than computational cost, should guide meta-learner selection in this context.

The combinations of first-level and second-level learners show no significant impact on the predictive performance or stability of the multi-layer model. However, using bagging models (RF and ET) as first-level learners and boosting models (XGB and LGB) as second-level learners results in the fastest training time, as shown in Figure 10 (see Appendices). This suggests that bagging models are more efficient at fitting high-dimensional data, making them advantageous for improving computational efficiency when used as first-level learners.

The impact of meta-learners on the predictive performance of the multi-layer stacking model mirrors the patterns observed in the stacking model and is therefore omitted. However, differences in model stability are evident, particularly for XGB, which exhibits the highest SD across all evaluation metrics, as shown in Figure 11 (see Appendices). This suggests that while XGB may achieve competitive predictive performance, its use as a meta-learner in multi-layer stacking models introduces greater variability.

Figure 12 (see Appendices) illustrates the impact of meta-learners on the averaged fit time of the multi-layer stacking model. Using ET as the meta-learner yields the fastest training time, highlighting its efficiency with low-dimensional datasets, since it is trained on the predictions of two algorithms from the previous layer. This finding further supports ET's suitability as a second-level learner in multi-layer stacking models.

Table 5 (see Appendices) summarizes the optimal hyperparameters yielding the lowest errors for each model in this study. Given that three evaluation metrics are used (MAE, RMSE, and MAPE), the optimal hyperparameters are determined by majority voting. If all metrics indicate different optimal values, the hyperparameters that minimize MAE are selected as the final configuration.

The best predictive performance for each model is presented in Figure 13 (see Appendices). The stacking model outperformed all others across all evaluation metrics, achieving mean MAE, RMSE, and MAPE values of 13.68, 22.73, and 0.1359, respectively. This result indicates that stacking is

particularly effective at aggregating complementary information from multiple base learners when modeling AQI using socioeconomic data. In contrast, the multi-layer stacking model failed to outperform the stacking model, with mean MAE, RMSE, and MAPE values of 13.97, 22.91, and 0.1382, respectively. This may be attributed to the small dataset size, which likely limited the multi-layer model's ability to capture complex relationships, introducing noise rather than valuable insights. Bagging and boosting models performed worse than both stacking and multi-layer stacking models, further emphasizing the advantages of combining base learners in a hierarchical structure. In general, bagging consistently outperforms boosting, indicating its stronger suitability for small-sample socioeconomic datasets.

Figure 14 (see Appendices) presents the stability of each model. Although boosting models show the worst predictive performance, they achieve the lowest SD for MAE, RMSE, and MAPE, with LGB recording the lowest SD for MAE (6.34) and MAPE (0.0248), and XGB achieving the lowest SD for RMSE (12.04). These results highlight the superior stability of boosting models compared to others, underscoring the trade-off between model accuracy and stability.

Table 6 (see Appendices) summarizes the best computational time for each model. Boosting models train faster than bagging models, with LGB achieving the fastest mean fit time of 0.02 seconds. In contrast, stacking models significantly increase training time, with mean fit times of 2.24 seconds for stacking and 6.65 seconds for multi-layer stacking. These results underscore the trade-off between model accuracy and computational efficiency.

Compared to previous studies that focused on meteorological or pollutant data, such as those by Méndez et al. (2023), Mao et al. (2021), and Gurumoorthy et al. (2023), this research highlights the predictive potential of socioeconomic aspects alone. While boosting models like XGBoost have performed well in prior studies (K. Kumar & Pande, 2023; Ravindiran et al., 2023), our results suggest that simpler ensemble methods, such as bagging, may be more suitable for limited data and non-traditional inputs. Similarly, our results show that a basic stacking approach outperforms a multi-layered stacking model, which contrasts with the findings of Shafieian and Zulkernine (2023). Furthermore, studies such as Wang et al. (2022) have identified strong correlations between socioeconomic status and air quality in China and the U.S., yet predictive integration of these factors remains limited. This study contributes to addressing that gap by empirically demonstrating the feasibility of using socioeconomic variables as reliable predictors for AQI modeling.

A major strength of this study lies in its exclusive focus on socioeconomic features, which broadens the conventional scope of AQI prediction and demonstrates the value of an underexplored data source. The use of multiple ensemble models, supported by a robust feature engineering process based on Pearson correlation thresholds, further enhances the model's predictive performance and reliability. While the analysis does not aim to identify which specific socioeconomic variables exert the strongest influence, the results nonetheless carry important policy implications. In urban contexts where dense air-quality sensor networks are unavailable or costly to maintain, predictive models based on publicly available socioeconomic data can serve as a complementary decision-support tool. Such models may help policymakers anticipate periods of deteriorating air quality, prioritize monitoring efforts, and design proactive mitigation strategies. Additionally, these insights can support public health planning by informing early warning systems, guiding the allocation of healthcare resources, and enabling targeted interventions for communities that are socioeconomically vulnerable to air pollution exposure.

However, several limitations should be acknowledged. First, the dataset is relatively small due to limited availability of high-frequency socioeconomic data. This constraint may have limited the effectiveness of deeper ensemble architectures such as multi-layer stacking and necessitated the use of a single cross-validation framework for hyperparameter tuning, model selection, and performance evaluation to maximize data utilization. Feature engineering and selection based on the Pearson correlation coefficient were also conducted prior to cross-validation for computational feasibility, which may introduce optimistic bias due to potential information leakage.

Second, the dataset is restricted to Singapore because comparable socioeconomic data are not publicly available for many other regions. This limits the generalizability of the findings to different geographic or socioeconomic contexts. In addition, the intentional exclusion of meteorological and pollutant variables limits direct comparability with conventional AQI prediction models that rely on environmental inputs, as this study focuses on assessing the predictive potential of socioeconomic indicators.

Finally, model performance was evaluated using cross-validation, which provides a robust estimate under data constraints but does not fully reflect real-world deployment scenarios involving future, unseen data. Moreover, while predictive performance was emphasized, model interpretability could be further enhanced by incorporating Explainable AI (XAI) techniques to yield more actionable policy insights.

Future research could explore hybrid models that integrate both socioeconomic and environmental variables to improve robustness and predictive power. Expanding the dataset to include data from other regions or gaining access to daily socioeconomic indicators may enhance the model’s generalizability. As temporal dependencies are not explicitly modeled in this study, future studies could adopt temporal hold-out validation strategies or real-time testing to assess out-of-sample performance in practical deployment scenarios better. Fold-wise feature selection during

cross-validation should also be applied to ensure a stricter separation between training and validation data. Incorporating XAI frameworks such as SHAP or LIME could also provide clearer insights into model decisions, helping city authorities understand the drivers of poor air quality. Finally, exploring the performance of advanced architectures, such as attention-based models and graph neural networks, could open new directions for high-resolution, urban-scale AQI forecasting.

IV. CONCLUSIONS

This study addresses a key gap in air quality modeling by systematically evaluating tree-based ensemble learning methods using socioeconomic indicators as the sole predictors of AQI. Unlike most prior studies that rely on meteorological or pollutant data, this work demonstrates that socioeconomic variables alone can support meaningful AQI prediction when paired with appropriate ensemble architectures. The results show that a well-designed stacking model provides the most effective balance between predictive accuracy, stability, and computational cost in a data-constrained setting. In contrast, additional architectural complexity, such as multi-layer stacking, offers limited benefit under such conditions.

Beyond model comparison, the key contribution of this study lies in its methodological insight: model complexity should be aligned with data characteristics rather than assumed to yield superior performance. This finding has practical relevance for cities where high-frequency environmental measurements are unavailable, but socioeconomic data are publicly accessible. In such contexts, ensemble-based predictive frameworks can serve as complementary tools for air quality monitoring, early warning systems, and evidence-informed urban planning.

Future research may extend this framework by integrating additional predictors, such as meteorological variables, to enhance model accuracy and robustness. Exploring alternative ensemble techniques, such as hybrid models, and utilizing larger datasets could further assess the scalability and performance of complex models like multi-layer stacking. Advanced feature engineering approaches, including Principal Component Analysis (PCA) or autoencoders, may also be investigated to optimize input variable selection. Additionally, applying Explainable AI (XAI) methods is strongly recommended to improve model interpretability and provide actionable insights for policymakers and government agencies. Overall, this study provides empirical guidance for selecting ensemble learning strategies in alternative data-driven AQI prediction and supports the broader integration of socioeconomic information into urban environmental analytics.

AUTHOR CONTRIBUTIONS

Conceived and designed the analysis, R., F. V. F. and F. V. P. S.; Collected the data, R.; Contributed data or analysis tools, R., F. V. F. and F. V. P. S.; Performed the analysis, R.; Wrote the paper, R.; Other contribution, F. V. F. and F. V. P. S.

DATA AVAILABILITY

Data derived from public domain resources. The data that supports the findings of the research are available in Air-Quality at https://github.com/Riechie19/Air-Quality. These data were derived from the following resources available in the public domain: https://data.gov.sg/, https://www.singstat.gov.sg/. The data that support the findings of the research are available from the corresponding author, FVP Samosir, upon reasonable request.

REFERENCES

Abayomi-Alli, O. O., Damaševičius, R., Maskeliūnas, R., & Misra, S. (2022). An ensemble learning model for COVID-19 detection from blood test samples. *Sensors*, *22*(6), 2224. https://doi.org/10.3390/s22062224

Alazba, A., & Aljamaan, H. (2022). Software defect prediction using stacking generalization of optimized tree-based ensembles. *Applied Sciences*, *12*(9), 4577. https://doi.org/10.3390/app12094577

Allgaier, J., & Pryss, R. (2024). Cross-validation visualized: A narrative guide to advanced methods. *Machine Learning and Knowledge Extraction*, *6*(2), 1378–1388. https://doi.org/10.3390/make6020065

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794.

Department of Statistics Singapore. (2025). *SingStat Website*. Retrieved August 8, 2025, from https://www.singstat.gov.sg/

Gafarov, F., Berdnikov, A., & Ustin, P. (2022). Online social network user performance prediction by graph neural networks. *International Journal of Advances in Intelligent Informatics*, *8*(3), 285. https://doi.org/10.26555/ijain.v8i3.859

Ghiasi, M. M., & Zendehboudi, S. (2021). Application of decision tree-based ensemble learning in the classification of breast cancer. *Computers in Biology and Medicine*, *128*, 104089. https://doi.org/10.1016/j.compbiomed.2020.104089

Gurumoorthy, S., Kokku, A. K., Falkowski-Gilski, P., & Divakarachari, P. B. (2023). Effective air quality prediction using reinforced swarm optimization and bi-directional gatedrecurrent unit. *Sustainability*, *15*(14), 11454. https://doi.org/10.3390/su151411454

Hu, L., & Li, L. (2022). Using tree-based machine learning

for health studies: Literature review and case series. *International Journal of Environmental Research and Public Health*, *19*(23), 16080. https://doi.org/10.3390/ijerph192316080

Issah, I., Appiah, O., Appiahene, P., & Inusah, F. (2023). A systematic review of the literature on machine learning application of determining the attributes influencing academic performance. *Decision Analytics Journal*, *7*, 100204. https://doi.org/10.1016/j.dajour.2023.100204

Jiao, K., Xu, M., & Liu, M. (2018). Health status and air pollution related socioeconomic concerns in urban China. *International Journal for Equity in Health*, *17*(1), 18. https://doi.org/10.1186/s12939-018-0719-y

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 3149–3157.

Koopialipoor, M., Asteris, P. G., Salih Mohammed, A., Alexakis, D. E., Mamou, A., & Armaghani, D. J. (2022). Introducing stacking machine learning approaches for the prediction of rock deformation. *Transportation Geotechnics*, *34*, 100756. https://doi.org/10.1016/j.trgeo.2022.100756

Kumar, K., & Pande, B. P. (2023). Air pollution prediction with machine learning: a case study of Indian cities. *International Journal of Environmental Science and Technology*, *20*(5), 5333–5348. https://doi.org/10.1007/s13762-022-04241-5

Kumar, M., Singhal, S., Shekhar, S., Sharma, B., & Srivastava, G. (2022). Optimized stacking ensemble learning model for breast cancer detection and classification using machine learning. *Sustainability*, *14*(21), 13998. https://doi.org/10.3390/su142113998

Lin, S.-Y., Kung, Y.-C., & Leu, F.-Y. (2022). Predictive intelligence in harmful news identification by BERT-based ensemble learning model with text sentiment analysis. *Information Processing & Management*, *59*(2), 102872. https://doi.org/10.1016/j.ipm.2022.102872

Manisalidis, I., Stavropoulou, E., Stavropoulos, A., & Bezirtzoglou, E. (2020). Environmental and health impacts of air pollution: A review. *Frontiers in Public Health*, *8*. https://doi.org/10.3389/fpubh.2020.00014

Mao, W., Wang, W., Jiao, L., Zhao, S., & Liu, A. (2021). Modeling air quality prediction using a deep learning approach: Method optimization and evaluation. *Sustainable Cities and Society*, *65*, 102567. https://doi.org/10.1016/j.scs.2020.102567

Méndez, M., Merayo, M. G., & Núñez, M. (2023). Machine learning algorithms to forecast air quality: a survey. *Artificial Intelligence Review*, *56*(9), 10031–10066. https://doi.org/10.1007/s10462-023-10424-4

Menshawy, L., Eid, A. H., & Abdel-Kader, R. F. (2023). Ensemble deep models for COVID-19 pandemic classification using chest X-ray images via different fusion techniques. *International Journal of Advances in Intelligent Informatics*, *9*(1), 51. https://doi.org/10.26555/ijain.v9i1.922

Mienye, I. D., & Sun, Y. (2022). A survey of ensemble learning: Concepts, algorithms, applications, and prospects. *IEEE Access*, *10*, 99129–99149. https://doi.org/10.1109/ACCESS.2022.3207287

National Environment Agency. (2025). *Singapore's open data portal*. Retrieved August 8, 2025, from https://data.gov.sg/datasets?agencies=National+Environment+Agency+(NEA)&resultId=d_e1058d6974c877257e32048ab128ad83&page=1

Putri, N. S. F., Wibawa, A. P., Ar Rasyid, H., Nafalski, A., & Hasyim, U. R. (2023). Boosting and bagging classification for computer science journal. *International Journal of Advances in Intelligent Informatics*, *9*(1), 27. https://doi.org/10.26555/ijain.v9i1.985

Rashid, M., Kamruzzaman, J., Imam, T., Wibowo, S., & Gordon, S. (2022). A tree-based stacking ensemble technique with feature selection for network intrusion detection. *Applied Intelligence*, *52*(9), 9768–9781. https://doi.org/10.1007/s10489-021-02968-1

Ravindiran, G., Hayder, G., Kanagarathinam, K., Alagumalai, A., & Sonne, C. (2023). Air quality prediction by machine learning models: A predictive study on the Indian coastal city of Visakhapatnam. *Chemosphere*, *338*, 139518. https://doi.org/10.1016/j.chemosphere.2023.139518

Saied, M., Guirguis, S., & Madbouly, M. (2023). A comparative analysis of using ensemble trees for botnet detection and classification in IoT. *Scientific Reports*, *13*(1), 21632. https://doi.org/10.1038/s41598-023-48681-6

Shafieian, S., & Zulkernine, M. (2023). Multi-layer stacking ensemble learners for low footprint network intrusion detection. *Complex & Intelligent Systems*, *9*(4), 3787–3799. https://doi.org/10.1007/s40747-022-00809-3

Sharma, A., Priya, T. H., & Naidu, V. (2024). Optimizing bearing health condition monitoring: exploring correlation feature selection algorithm. *Engineering Research Express*, *6*(2), 025511. https://doi.org/10.1088/2631-8695/ad3380

Singh, P., Pal, G. K., & Gangwar, S. (2022). Prediction of cardiovascular disease using feature selection techniques. *International Journal of Computer Theory and Engineering*, *14*(3), 97–103. https://doi.org/10.7763/IJCTE.2022.V14.1316

Van, N. H., Van Thanh, P., Tran, D. N., & Tran, D.-T. (2023). A new model of air quality prediction using lightweight machine learning. *International Journal of Environmental Science and Technology*, *20*(3), 2983–2994. https://doi.org/10.1007/s13762-022-04185-w

Wang, Y., Wang, Y., Xu, H., Zhao, Y., & Marshall, J. D. (2022). Ambient Air Pollution and Socioeconomic Status in China. *Environmental Health Perspectives*, *130*(6). https://doi.org/10.1289/EHP9872

Yoon, H. I., Lee, H., Yang, J.-S., Choi, J.-H., Jung, D.-H., Park, Y. J., Park, J.-E., Kim, S. M., & Park, S. H. (2023). Predicting models for plant metabolites based on PLSR, AdaBoost, XGBoost, and LightGBM algorithms using hyperspectral imaging of brassica juncea. *Agriculture*, *13*(8), 1477. https://doi.org/10.3390/agriculture13081477

Zhang, Y., Liu, J., & Shen, W. (2022). A Review of ensemble learning algorithms used in remote sensing applications. *Applied Sciences*, *12*(17), 8654. https://doi.org/10.3390/app12178654

Zhiyong, L., Hongdong, Z., Ruili, Z., Kewen, X., Qiang, G., & Yuhai, L. (2019). Fault identification method of diesel engine in light of pearson correlation coefficient diagram and orthogonal vibration signals. *Mathematical Problems in Engineering*, *2019*(1). https://doi.org/10.1155/2019/2837580

APPENDICES

Table 1 Socioeconomic Variables Used as Input Features in This Study

Label	Variable Names	Category	Label	Variable Names	Category
x_1	Month	-	x_{18}	FnB Sales Index in Chained Volume	Industrialization
x_2	Number of Public Holiday	-	x_{19}	FnB Sales Index at Current Price	Industrialization
x_3	Cars	Transportation	x_{20}	Food & Beverage Sales Value	Industrialization
x_4	Private Hire Cars	Transportation	x_{21}	Consumer Price Index	Economy & Prices
x_5	Taxis	Transportation	x_{22}	MAS Core Inflation Measure	Economy & Prices
x_6	Buses	Transportation	x_{23}	Services Inflation Measure	Economy & Prices
x_7	Motorcycles & Scooters	Transportation	x_{24}	Retail & Other Goods Inflation Measure	Economy & Prices
x_8	Goods & Other Vehicles	Transportation	x_{25}	Electricity & Gas Inflation Measure	Economy & Prices
x_9	Average Daily Ridership	Transportation	x_{26}	Price of Diesel (Per Litre)	Economy & Prices
x_{10}	International Visitor Arrivals	Tourism	x_{27}	Price of Petrol 98 Octane	Economy & Prices
x_{11}	Visitor Average Length of Stay	Tourism	x_{28}	Price of Petrol 95 Octane	Economy & Prices
x_{12}	Outbound Departures	Tourism	x_{29}	Price of Petrol 92 Octane	Economy & Prices
x_{13}	Available Hotel Room-Nights	Tourism	x_{30}	Price of LPG (Per KG)	Economy & Prices
x_{14}	Index of Industrial Production	Industrialization	x_{31}	Import Price Index	Economy & Prices
x_{15}	Retail Sales Index in Chained Volume	Industrialization	x_{32}	Export Price Index	Economy & Prices
x_{16}	Retail Sales Index at Current Price	Industrialization	x_{33}	Singapore Manufactured Products Price Index	Economy & Prices
x_{17}	Retail Sales Value	Industrialization	x_{34}	Domestic Supply Price Index	Economy & Prices

Table 2 Feature Engineered Input Variables Derived from Socioeconomic Data

Label	Variable Names	Category	Operation	Absolute PCC (PSI)
x_1'	Goods & Other Vehicles	Transportation	x_8	0.156935
x_2'	Taxis & Private Hire Cars difference	Transportaion	abs(x_4 - x_5)	0.302729
x_3'	Buses & Scooters ratio	Transportation	x_7 / x_6	0.171329
x_4'	LoS & Hotel Mult	Tourism	x_{11} * x_{13}	0.14357
x_5'	Arrivals & Departures ratio	Tourism	x_{12} / x_{10}	0.143663
x_6'	IIP	Industrialization	x_{14}	0.271306
x_7'	FBSI CV	Industrialization	x_{18}	0.196686
x_8'	RSV & RSI CP Mult	Industrialization	x_{16} * x_{17}	0.102679
x_9'	MAS Core Inflation Measure	Economy & Prices	x_{22}	0.198879
x_{10}'	Price of LPG (Per KG)	Economy & Prices	x_{30}	0.331961
x_{11}'	Petrol & Diesel Ratio	Economy & Prices	x_{28} / x_{26}	0.281864
x_{12}'	IPI & EPI difference	Economy & Prices	abs(x_{31} - x_{32})	0.269978
x_{13}'	SPMPI & DSPI Ratio	Economy & Prices	x_{33} / x_{34}	0.214352

Table 3 Hyperparameters for Tree-Based Models to be Tuned Using Grid Search

Model	Hyperparameters	Values
RF (Random Forest) & ET (ExtraTrees)	n_estimators	[100, 200, …,500]
	max_depth	[4, 6 , 8, 10]
	min_samples_leaf	[1, 6, 11, 20]
	criterion	[squared error, absolute error]
	max_features	[None, sqrt(n), log2(n)]
XGB (XGBoost)	n_estimators	[100, 200, …, 500]
	max_depth	[4, 6, 8, 10]
	learning_rate	[0.05, 0.1, 0.3]
	reg_lambda	[0.5, 1, 1.5]
	reg_alpha	[0, 0.1, 0.5]
LGB (LightGBM)	num_iterations	[100, 200, …, 500]
	min_child_samples	[1, 6, 11, 20]
	num_leaves	[10, 31, 250, 500]
	learning_rate	[0.05, 0.1, 0.3]
	max_bin	[10, 50, 90, 255]

Table 4 Architectures for Stacking and Multi-Layer Stacking Models to be Tuned Using Grid Search

Combination	Stacking	Multi-Layer Stacking	
	First-Level Learner	First-Level Learner	Second-Level Learner
1	RF, ET	RF, ET	XGB, LGB
2	RF, XGB	RF, XGB	ET, LGB
3	RF, LGB	RF, LGB	ET, XGB
4	ET, XGB	ET, XGB	RF, LGB
5	ET, LGB	ET, LGB	RF, XGB
6	XGB, LGB	XGB, LGB	RF, ET
7	RF, ET, XGB		
8	RF, ET, LGB		
9	RF, XGB, LGB		
10	ET, XGB, LGB		
11	RF, ET, XGB, LGB		

Table 5 Optimal Hyperparameters for Each Model

Bagging		Boosting		Stacking	
Hyperparameters	Value	Hyperparameters	Value	Architecture	Value
RF		*XGB*		*stacking*	
n_estimators	100	n_estimators	100	first-level learner	RF, ET
max_depth	4	max_depth	4	meta learner	RF
min_samples_leaf	20	learning_rate	0.05		
criterion	absolute_error	reg_lambda	1.5		
max_features	None	reg_alpha	0		
ET		*LGB*		*multi-layer stacking*	
n_estimators	100	num_iterations	100	first-level learner	RF, ET
max_depth	6	min_child_samples	20	second-level learner	XGB, LGB
min_samples_leaf	11	num_leaves	10	meta learner	ET
criterion	absolute_error	learning_rate	0.05		
max_features	None	max_bin	255		

Table 6 Best Computational Time for All Models

Model	Mean Fit Time (in Seconds)
RF	0.20
ET	0.13
XGB	0.10
LGB	0.02
Stacking	2.24
Multi-Layer Stacking	6.65

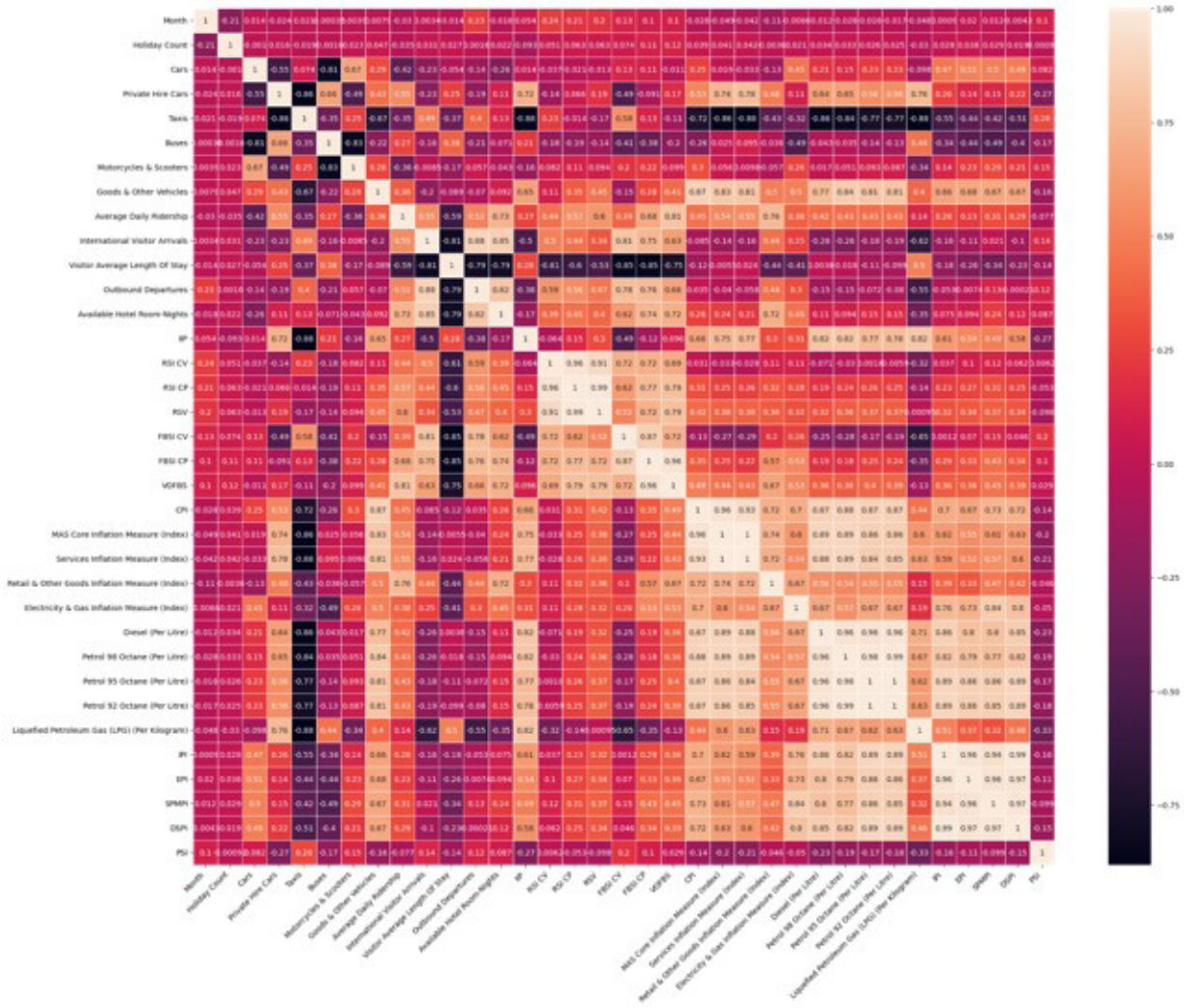


Figure 1 Heatmap of Pearson Correlation Coefficient (PCC) Values Between Variables

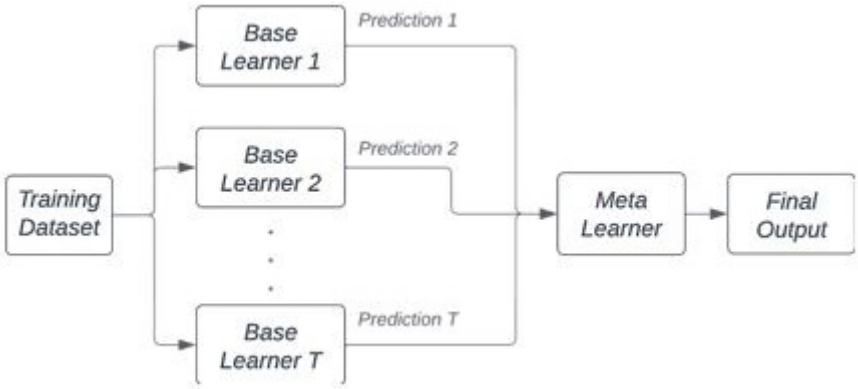


Figure 2 General Framework of Stacking Model.
First-level Learners Generate Predictions that are Used as Inputs for a Meta-Learner

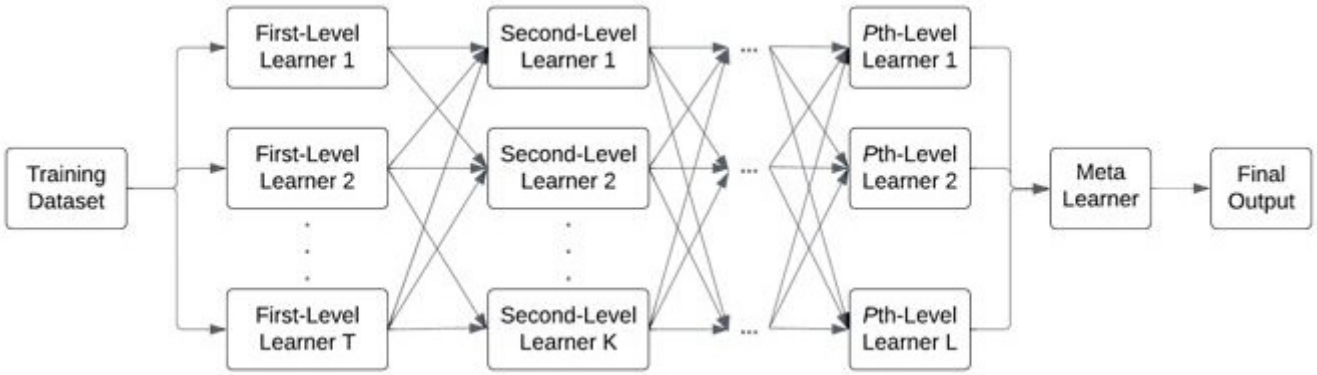


Figure 3 General Framework of Multiple Layered Stacking Method.
Learners of Each Layer Generate Predictions that are Used as Inputs for Learners in the Subsequent Layer

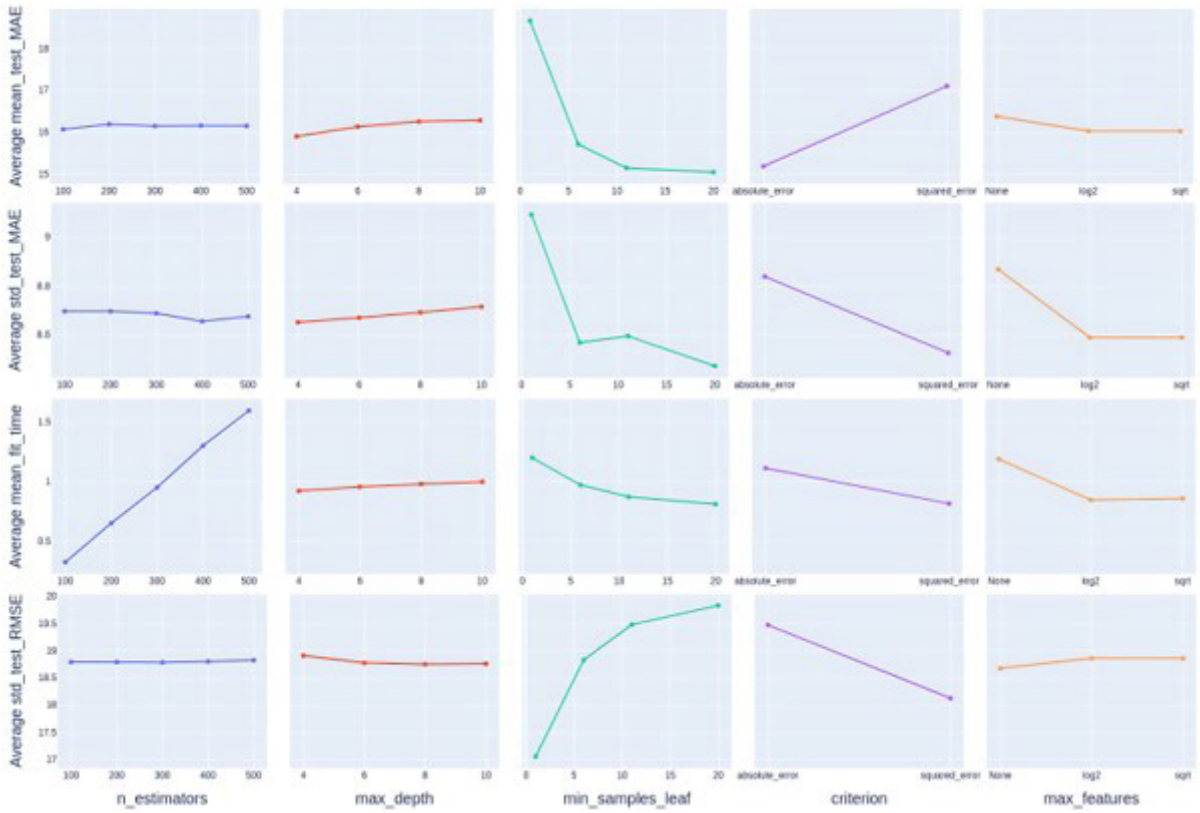


Figure 4 Impact of Hyperparameters on RF (Random Forest) Model Performance.
The Rows Represent Mean MAE (Mean Absolute Error), SD of MAE (Mean Absolute Error),
Mean Fit Time, and Mean RMSE (Root Mean Squared Error) Across 10-Fold Cross-Validation

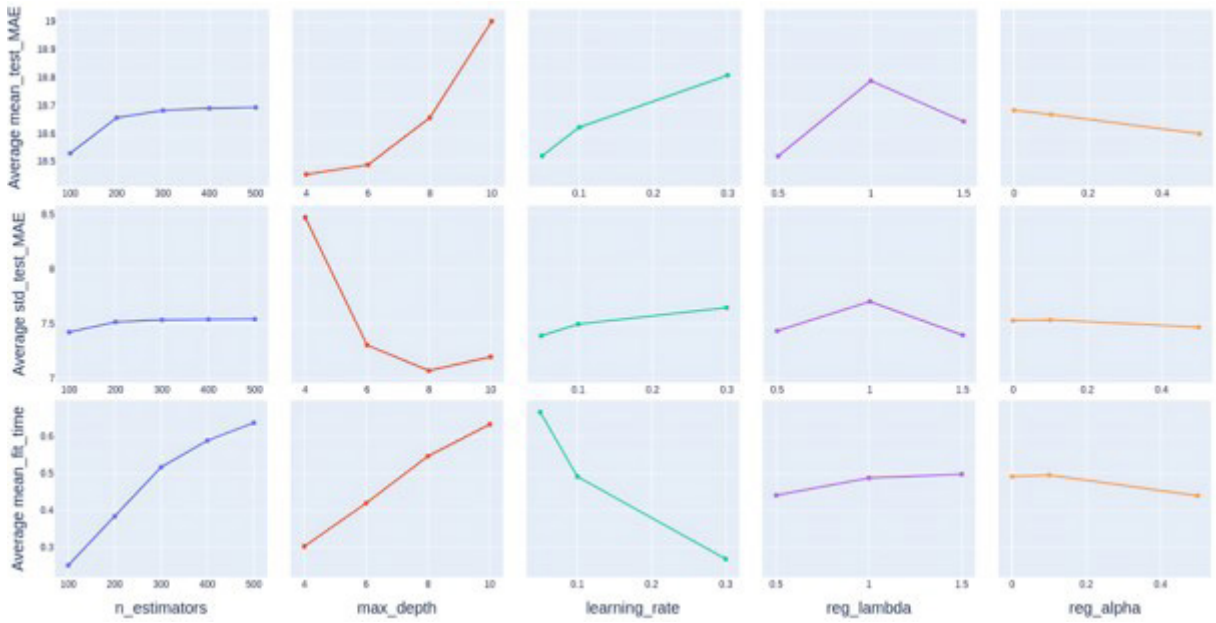


Figure 5 Impact of Hyperparameters on XGB (XGBoost) Model Performance.
The Rows Represent Mean MAE, SD of MAE, and Mean Fit Time Across 10-Fold Cross-Validation

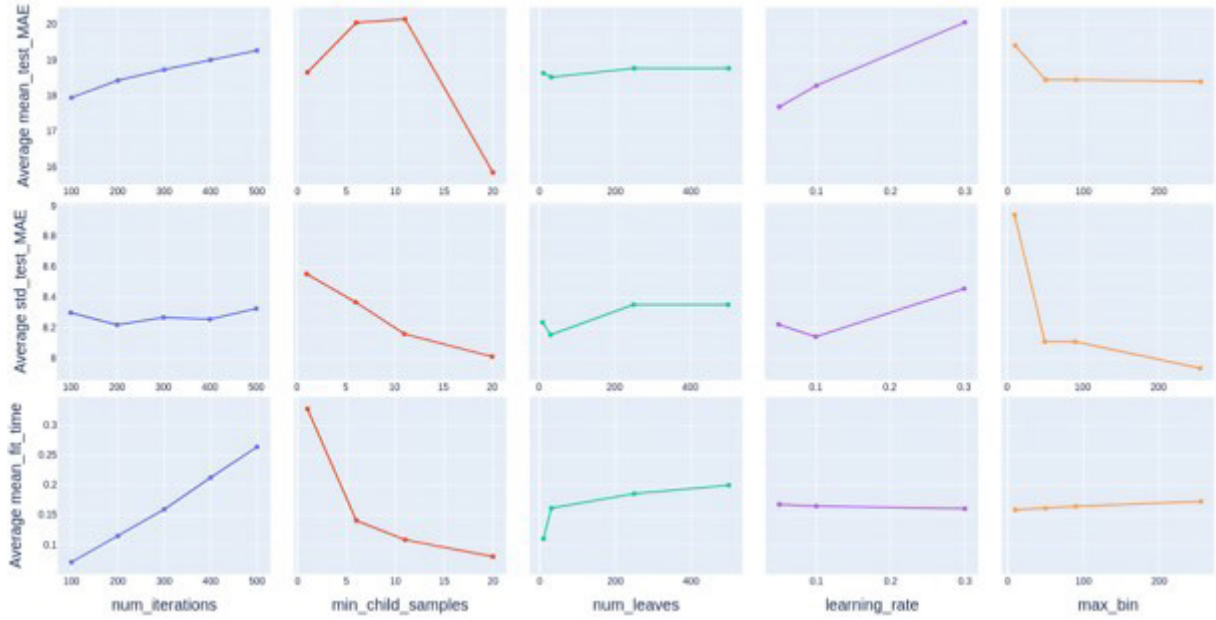


Figure 6 Impact of Hyperparameter on LGB (LightGBM) Model Performance.
The Rows Represent Mean MAE, SD of MAE, and Mean Fit Time Across 10-Fold Cross-Validation.

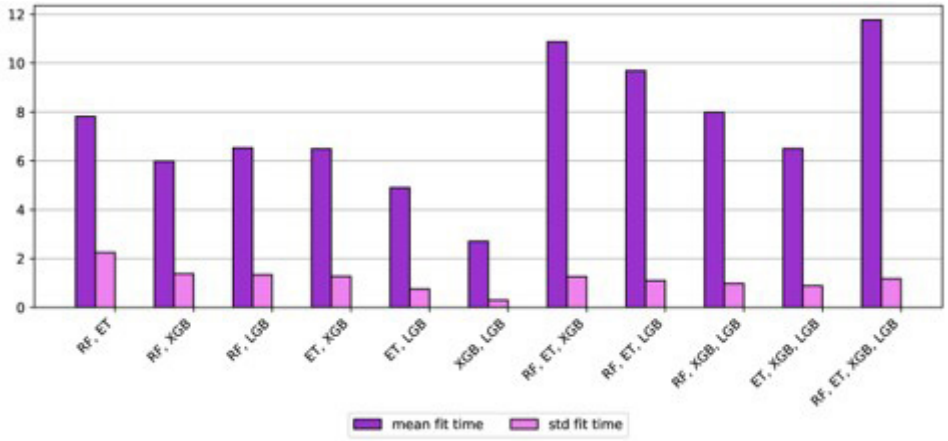


Figure 7 Impact of First-Level Learners on Stacking Model's
Average Fit Time (in Seconds) Across 10-Fold Cross-Validation

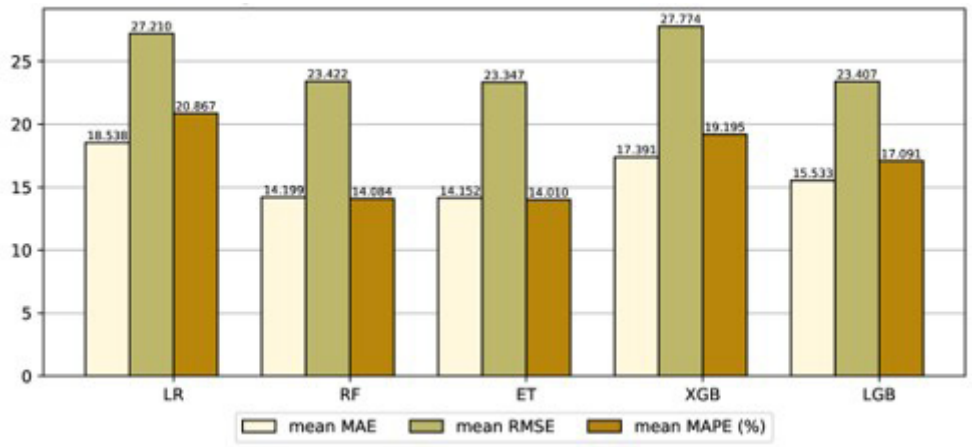


Figure 8 Impact of Meta-Learner on Stacking Model Predictive Performance.
Values Represent Averages of Mean MAE, RMSE, and MAPE Across 10-Fold Cross-Validation

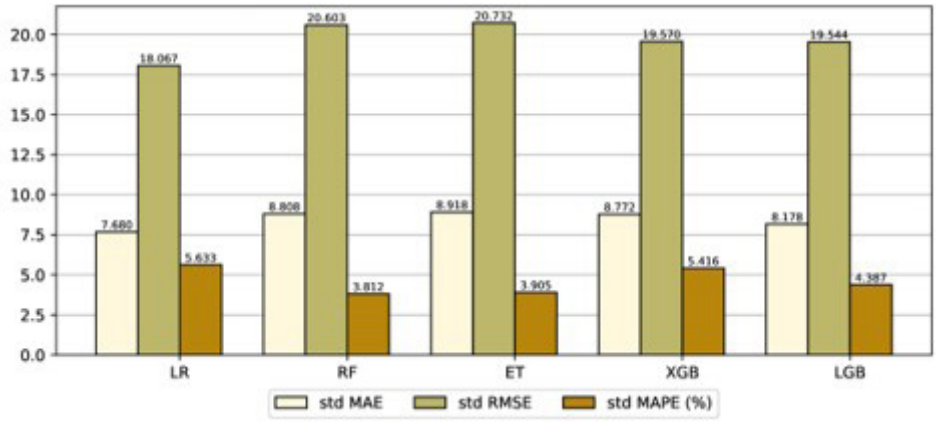


Figure 9 Impact of Meta-Learner on Stacking Model Stability.
Values Represent Averages of SD of MAE, RMSE,
and MAPE Across 10-Fold Cross-Validation

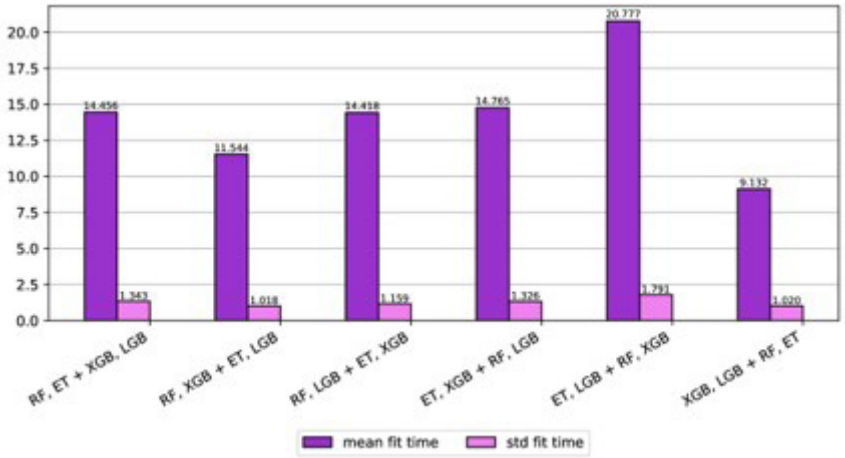


Figure 10 Impact of First and Second-Level Learners on Multi-Layer Stacking
Model Average Fit Time (in Seconds) Across 10-Fold Cross Validation

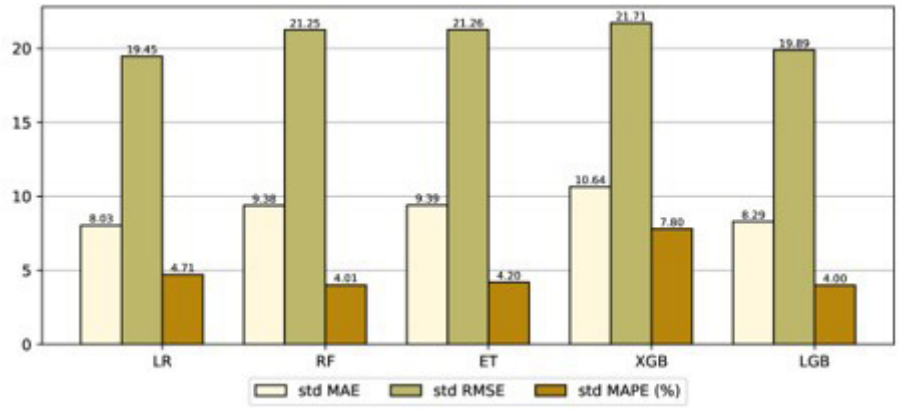


Figure 11 Impact of Meta-Learner on Multi-Layer Stacking Model Stability.
Values Represent Averages of SD of MAE, RMSE, and MAPE Across 10-Fold Cross Validation

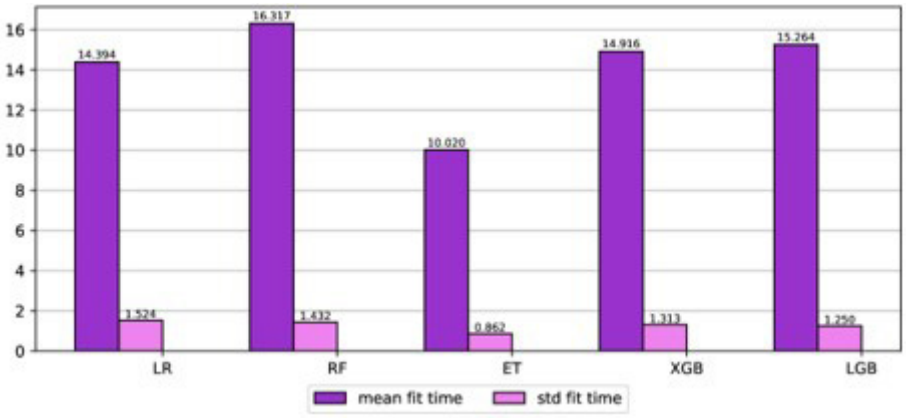


Figure 12 Impact of Meta-Learner on Multi-Layer Stacking Model
Average Fit Time (in Seconds) Across 10-Fold Cross Validation

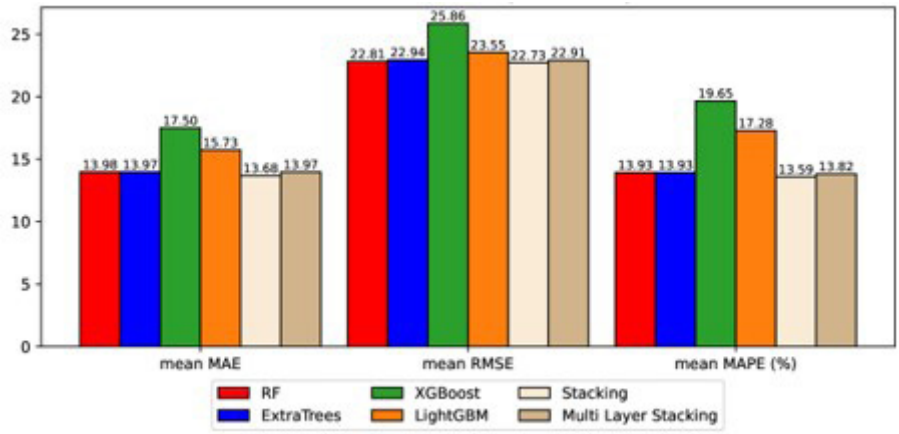


Figure 13 Best predictive performance for all models.
Values Represent Averages of Mean MAE, RMSE, and MAPE Across 10-Fold Cross Validation

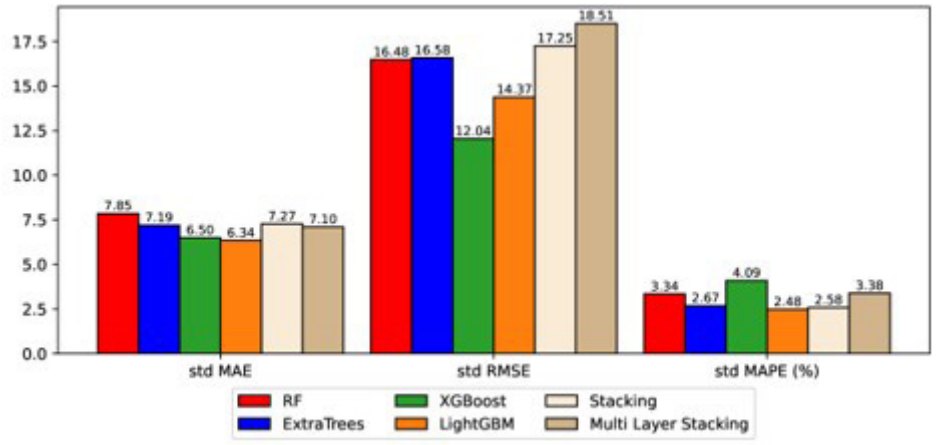


Figure 14 Best Stability for All Models. Values Represent
Averages of Mean MAE, RMSE, and MAPE Across 10-Fold Cross Validation