

Comparative Analysis of Machine Learning and Deep Learning Models for PM2.5 and PM10 Time Series Forecasting Using the SISANAPAS Web Platform

Dwi Indra Prasetyo^{1*}, Agustina Rachmawardani², Bayu Satrio³, Derby Brylian Adinara⁴

^{1,2,3,4}MCG-Instrumentation Study Program, State College of Meteorology, Climatology, and Geophysics, Tangerang, Banten, Indonesia

E-mail: ^{1*}dwi.indra.prasetyo2003@gmail.com, ²agustina.rahmawardani@stmkg.ac.id, ³strbayu1404@gmail.com, ⁴adinara.cmpt@gmail.com

(Received: 21 Jul 2025, revised: 18 Aug 2025, accepted: 20 Aug 2025)

Abstract

Particulate matter (PM2.5 and PM10) pollution remains a significant environmental concern in Indonesia. This study employed the SISANAPAS web platform to compare machine learning (ML) and deep learning (DL) algorithms for PM2.5 and PM10 forecasting. Using 3-hourly data from Cibeureum (January-October 2024), which underwent comprehensive pre-processing (K-Nearest Neighbors imputation, Z-score outlier removal, Min-Max scaling, feature engineering), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Random Forest, and XGBoost models were evaluated. XGBoost Regression provided the most accurate forecasts, with an R^2 of 0.85, RMSE of 5.05 $\mu\text{g}/\text{m}^3$, and MAE of 3.19 $\mu\text{g}/\text{m}^3$ for PM2.5, and an R^2 of 0.85, RMSE of 9.71 $\mu\text{g}/\text{m}^3$, and MAE of 6.55 $\mu\text{g}/\text{m}^3$ for PM10. These results, significantly outperforming LSTM and GRU, highlight XGBoost's potential for reliable air quality prediction and demonstrate SISANAPAS as a valuable tool for environmental data analysis, crucial for informing public health and environmental policies.

Keywords: Air Quality, Particulate Matter, Time Series Forecasting, Machine Learning, Deep Learning.

I. INTRODUCTION

Air quality in Indonesia, particularly concerning particulate matter (PM) pollution, poses significant environmental and public health challenges. Rapid urbanization and increased industrial activities in urban areas have led to a notable decline in air quality [1]. According to the World Air Quality Report by IQAir from 2018 to 2023, Indonesia ranked as the country with the highest air pollution levels in Southeast Asia [2]. The main contributing factors to air pollution in Indonesia are the transportation sector, industrial activities, and biomass burning, which have been identified as the most significant sources [3].

Particulate Matter (PM) refers to a mixture of microscopic solid or liquid particles suspended in the air, specifically PM2.5 and PM10, which are classified by their aerodynamic diameters, with PM2.5 being particles with a diameter of 2.5 micrometers or smaller and PM10 encompassing particles of 10 micrometers or smaller [4].

In terms of air quality standards, the *Technical Assistance Document for the Reporting of Daily Air Quality – the Air Quality Index* provides a structured framework for classifying

Air Quality Index (AQI) levels, as shown in Table 1. Published by the Environmental Protection Agency (EPA), this document outlines the methodology used to interpret and report daily concentrations of air pollutants.

Table 1. AQI Categories Based on 24-Hour PM2.5 and PM10 Concentrations [5]

AQI Category	PM2.5 ($\mu\text{g}/\text{m}^3$)	PM10 ($\mu\text{g}/\text{m}^3$)
Good	0.0 – 9.0	0 – 54
Moderate	9.1 – 35.4	55 – 154
Unhealthy for Sensitive Groups	35.5 – 55.4	155 – 254
Unhealthy	55.5 – 125.4	255 – 354
Very Unhealthy	125.5 – 225.4	355 – 424
Hazardous	225.5+	425+

The health implications of PM2.5 and PM10 exposure are substantial, with strong links established to cardiovascular and respiratory diseases. Evidence indicates that long-term exposure to PM2.5 correlates with increased rates of mortality from cardiovascular conditions and respiratory illnesses, as

demonstrated in multiple studies [6]–[8]. Specifically, PM2.5 penetration into the lungs can incite inflammatory responses that lead to chronic conditions, including chronic obstructive pulmonary disease (COPD) [9].

The potential of machine learning (ML) and deep learning (DL) techniques for addressing air pollution, particularly PM2.5 and PM10 levels, is substantial and continuously evolving, as evidenced by a multitude of recent studies. The application of models such as Long Short-Term Memory (LSTM), Random Forest (RF) and XGBoost has demonstrated notable improvements in predicting particulate matter concentrations. These models address specific air quality challenges, including inaccurate forecasting, limited spatial-temporal resolution, and the complexities of source apportionment, which can impede timely public health warnings and policy responses [10]–[12]. For instance, Tsai et al. demonstrated that the use of LSTM neural networks for PM2.5 forecasting yields satisfactory results, particularly when pollutant levels are not influenced by transboundary pollution or climate factors, achieving a minimum MAE of 1.644 and a maximum of 5.543 for one-hour forecasts; moreover, the model shows promising accuracy for long-term predictions [13]. Amelia's study successfully identified the Seasonal ARIMA (0,1,1)(1,1,1) [12] model as more accurate than Triple Exponential Smoothing for forecasting PM2.5 levels in Jakarta, with an average percentage difference of -11.52% compared to actual data, based on secondary air quality reports from IQAir covering the period 2018–2021 [14]. Moreover, Zaini et al. found that the EMD-BiLSTM ensemble achieved the best PM2.5 forecasting performance, reducing LSTM's RMSE by up to 4.3% and GRU's by up to 17.35%, while CNN-LSTM performed the worst with an RMSE of 12.0066 $\mu\text{g}/\text{m}^3$ and MAPE of 38.48% [15]. Furizal et al. propose a hybrid GRU-ARIMA approach for PM10 AQI forecasting in Jakarta, where GRU handles time series prediction while ARIMA imputes missing data, achieving MSE 15.6457 and RMSE 3.2170, with results showing decreased accuracy at longer timesteps but successfully capturing seasonal PM10 peaks (May–October) to inform real-time air quality management decisions [16]. Altinçöp et al. study found that RF consistently outperforms Neural Networks in predicting air pollution, with MAE for PM10 in Şişli at 1.225 (RF) compared to 1.8985 (Neural Network), and for CO in Dilovası at 1.717 (RF) versus 2.5132 (Neural Network), while Neural Networks slightly outperform RF for CO in Şişli with MAE of 1.3168 versus 1.717 [17]. Alfasanah et al. study on PM2.5 monitoring in Jakarta revealed that XGBoost Regression achieved superior training performance with an RMSE of 9.87 compared to Support Vector Regression's 17.16, yet demonstrated reduced testing accuracy with an RMSE of 20.38 while SVR maintained more stable performance at 18.85, with the integrated XGBoost-EWMA approach emerging as the optimal solution by detecting merely one out-of-control point in Phase II and exhibiting excellent control in Phase I, thereby establishing an effective monitoring framework with minimal false alarms for urban air quality surveillance [18].

The SISANAPAS web platform serves as a sophisticated web-based platform designed to facilitate air quality data modeling, empowering researchers to develop time series prediction models without the necessity of programming proficiency. Constructed with advanced technologies including PHP Laravel for server-side operations, JavaScript for dynamic user interactions, and Python for backend data processing, the platform ensures seamless experience for its users. Key features of SISANAPAS encompass comprehensive data input capabilities accompanied by robust database storage, coupled with intuitive data exploration through the ydata library [19]. Users can efficiently engage in vital tasks such as data pre-processing, model training, evaluation, parameter tuning, and the subsequent saving of models—all facilitated through an accessible and user-friendly interface. This integration of functionalities within SISANAPAS significantly enhances the research community's ability to conduct air quality modeling, thereby aiming to improve environmental data insights. Figure 1 illustrates the system design of SISANAPAS, depicting the flow from the input page to exploratory data analysis, pre-processing, training and evaluation, hyperparameter tuning, and completion, all managed by a controller within the Laravel framework, with Python scripts handling backend processing and a database storing CSV data.

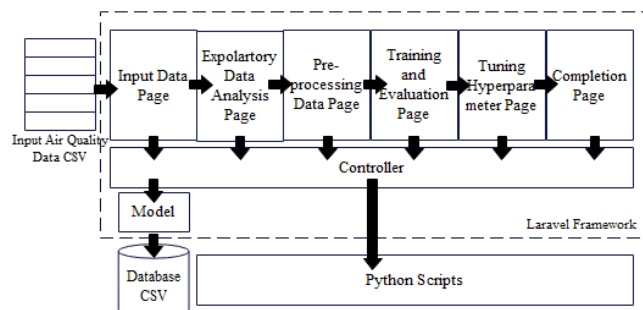


Figure 1. System Design of SISANAPAS Web Platform

This study advances the state of the art in air quality forecasting by using the SISANAPAS web platform to overcome the common programming barrier in model comparison. This tool facilitates a direct, code-free evaluation of leading models, enabling us to identify the most accurate algorithm while validating a more accessible research workflow.

This paper presents a comparative analysis of machine learning and deep learning algorithms for air quality prediction, utilizing the SISANAPAS website for time series modeling of PM2.5 and PM10 pollutants. We evaluate the performance of LSTM, GRU, Random Forest Time Series, and XGBoost Regression through multiple metrics, implementing hyperparameter tuning to optimize predictive accuracy. The study aims to inform evidence-based policy interventions through precise air quality forecasts. Our findings contribute to advancing machine learning applications in environmental monitoring, demonstrating how optimized algorithmic approaches can enhance predictive capabilities for critical air quality parameters.

II. RESEARCH METHODOLOGY

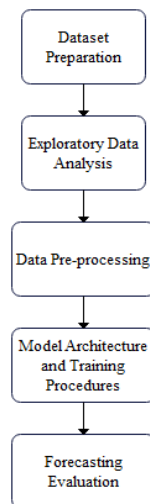


Figure 2. Workflow of the Research Methodology

Figure 2 outlines the structured, linear workflow of the research methodology employed in the study. This process starts with dataset preparation, where the initial data is collected and organized. The next step is exploratory data analysis, which involves examining the dataset to understand its primary characteristics, identify patterns, and detect anomalies. Following this is data pre-processing, a crucial stage where the data is cleaned, transformed, and made suitable for modeling. The fourth stage, model architecture and training procedures, involves selecting the appropriate machine learning models, defining their structures, and training them on the pre-processed data. The final step is forecasting evaluation, where the performance of the trained models is rigorously assessed using various statistical metrics to determine their predictive accuracy and identify the best-performing model.

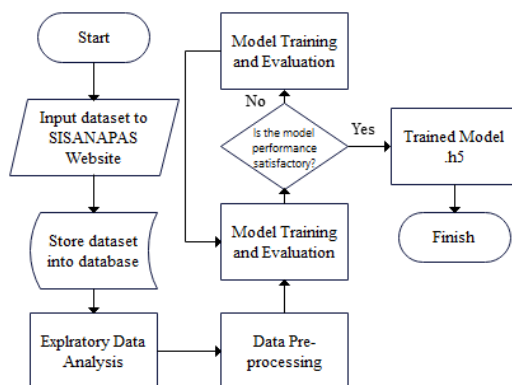


Figure 3. Workflow of SISANAPAS Web-Based Modeling Methodology

Figure 3 illustrates the operational workflow of the SISANAPAS web platform, detailing the user's journey from data submission to model finalization. The process begins when a user inputs a dataset, which is then stored in the system's database. Following this, the platform facilitates exploratory data analysis to uncover initial insights and data pre-processing to clean and prepare the data. The core of the

workflow is an iterative cycle of model training and evaluation. A decision point assesses whether the model's performance is satisfactory; if not, the user can re-engage in the training and evaluation phase, presumably to tune parameters or select different options. Once the performance meets the required standards, the trained model is saved as an .h5 file, concluding the process.

A. Dataset Preparation

This study utilizes a dataset sourced from the Cibeureum Air Pollution Monitoring Post, an operational unit under the coordination of the Citeko Bogor Meteorological Station, managed by the Indonesian Agency for Meteorology, Climatology, and Geophysics (BMKG). The monitoring post is tasked with observing, processing, and disseminating atmospheric chemistry data, including greenhouse gases (CO₂, CH₄), reactive gases (CO, O₃, NO₂, SO₂), suspended particulate matter (SPM), precipitation quality, and aerosols (PM10, PM2.5). The dataset, which comprises a total of 2,448 data points for each parameter at a 3-hour resolution, was retrieved directly from the local computer system at the Cibeureum Air Pollution Monitoring Post and spans a 10-month period from January 2024 to October 2024. The parameters selected for analysis include PM10 and PM2.5, key pollutants in Indonesia frequently associated with health impacts and air quality [20]. The data were input through the SISANAPAS website's data input page and stored in the system's database for subsequent analysis.

B. Exploratory Data Analysis

After the dataset is stored in the SISANAPAS website's database, this study proceeds to the exploratory data analysis (EDA) phase. The system employs the Ydata library to generate detailed descriptive statistics, including measures of central tendency and dispersion for the dataset. Visualizations such as correlation matrices are produced to illustrate relationships between parameters, alongside plots to identify missing values.

C. Data Pre-processing

Following the exploratory data analysis, this study advances to the data pre-processing stage through the SISANAPAS website interface. The system offers several pre-processing options, including data cleaning to eliminate outliers and address missing values, ensuring the dataset's integrity. Additionally, this study may apply data imputation to handle remaining missing values and implement data transformation techniques, such as scaling, to standardize the dataset for modeling purposes.

D. Model Architecture and Training Procedures

This study implements and compares several time-series forecasting models, specifically, Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Random Forest Time Series, and XGBoost Regression.

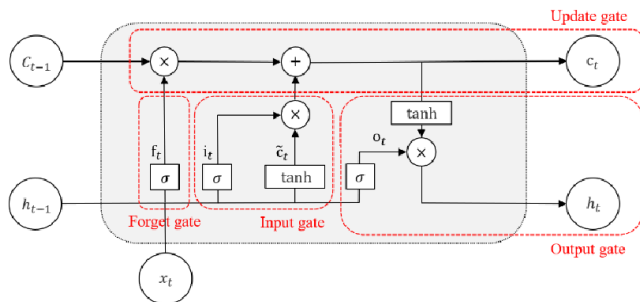


Figure 4. Architecture of the LSTM Model [21]

Figure 4 illustrates the architecture of the LSTM model employed in the modeling process of this study. Long Short-Term Memory (LSTM) neural networks represent a significant advancement in the field of recurrent neural networks (RNNs), addressing the challenge posed by long-term dependencies in time series data. LSTMs are characterized by their unique architecture, which includes memory cells and three types of gates: input, forget, and output gates. This design allows LSTMs to maintain information over extended sequences, enabling them to selectively retain or discard information as required, thus mitigating the vanishing gradient problem commonly associated with traditional RNNs [22].

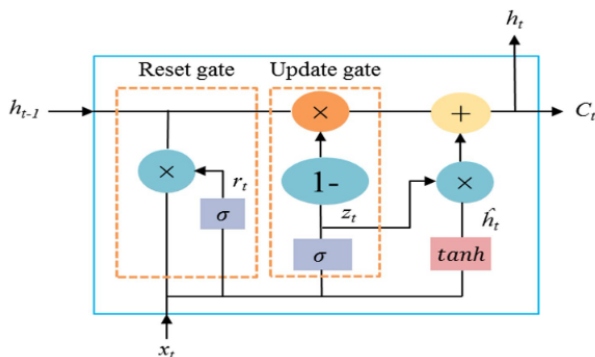


Figure 5. Architecture of the GRU Model [23]

The Gated Recurrent Unit (GRU) is a type of artificial neural network categorized under Recurrent Neural Networks (RNNs), designed to address the limitations of traditional RNNs, particularly in capturing long-term dependencies in sequential data, with a simpler structure compared to Long Short-Term Memory (LSTM) networks [24]. Its architecture is illustrated in Figure 5.

The utilization of ensemble methods can enhance the model's generalization capability and reduce predictive uncertainty [25]. Random Forest (RF) is a powerful ensemble machine learning technique predominantly utilized for classification and regression tasks, including time series forecasting. Built upon the fundamental principle of decision tree modeling, RF creates multiple decision trees during training and merges their outputs to enhance predictive accuracy and control overfitting through bagging, a method that combines predictions from multiple models to improve robustness [26].

Building on the concept of ensemble techniques, Extreme Gradient Boosting (XGBoost) is also a highly efficient machine learning algorithm tailored for regression and classification tasks, including time series forecasting. At its core, XGBoost combines the principles of gradient boosting and decision trees, employing an ensemble method that sequentially integrates multiple models to optimize predictions while employing regularization techniques to enhance model generalization and prevent overfitting [27].

The training process involves fitting each model to the pre-processed dataset, which is divided into three main subsets: training data, validation data, and testing data. Specifically, 75% of the total data is allocated as training data, utilized to train the model in recognizing patterns within the data, while 10% is designated as validation data to evaluate model performance during training and assist in selecting the optimal hyperparameters. The remaining 15% of the data is reserved as testing data to assess the model's generalization ability on unseen data. This structured data splitting, followed by hyperparameter tuning, optimizes model performance and enables a thorough evaluation of each model's effectiveness in forecasting PM2.5 and PM10 concentrations, facilitating the identification of the most suitable model for the dataset analyzed in this study.

E. Forecasting Evaluation

This study evaluates the models using key metrics, including Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and the coefficient of determination (R^2). The system ranks the models based on these metrics, identifying the best-performing model.

The Mean Absolute Error (MAE) is defined as the average of the absolute differences between predicted and observed values, expressed mathematically as Equation 1 [28]. The MAE provides a direct measure of average error magnitude, which is particularly beneficial for comprehending model performance in a straightforward manner.

Furthermore, the Root Mean Square Error (RMSE) is another crucial metric expressed as Equation 2. RMSE is particularly sensitive to larger errors due to the squaring of differences, thus emphasizing substantial forecasting discrepancies [29]. Lastly, the Coefficient of Determination (R^2) assesses the proportion of variance in the dependent variable explained by the independent variables, articulated as Equation 3 [30]. R^2 serves a pivotal role in elucidating model fit and predictive power, thus guiding researchers in model selection [31].

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \tag{1}$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \tag{2}$$

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \bar{y})^2}{\sum_{i=1}^n (y_i - \hat{y}_i)^2} \tag{3}$$

III. RESULT AND DISCUSSION

A. Dataset Characteristics and Exploratory Data Analysis Insight

Table 2. Exploratory Data Analysis of PM10 and PM2.5 Datasets

Metrics	PM2.5	PM10
Missing	954	677
Missing (%)	5.1%	3.6%
Mean	33.73	16.16
Minimum	-989.9	-40.9
Maximum	790.2	472.8
Zeros	502	518
Zeros (%)	2.7%	2.8%
Q1	13.3	5.5
Median (Q2)	27.9	12.9
Q3	47.8	23.3
Interquartile Range (IQR)	34.5	17.8
Standard Deviation (SD)	28.05	14.66
Coefficient of Variation (CV)	0.80	0.90
Kurtosis	45.58	94.16
Mean Absolute Deviation (MAD)	17.37	8.42
Skewness	4.18	2.93

Table 2 presents the exploratory data analysis (EDA) of the PM10 and PM2.5 datasets, revealing characteristics that require preprocessing data. PM10 dataset spans a broad range of values, from -989.9 to 790.2 $\mu\text{g}/\text{m}^3$, with 1.5% negative entries (275 observations) and 2.7% zero values (502 observations), while PM2.5 ranges from -40.9 to 472.8 $\mu\text{g}/\text{m}^3$, with 1.2% negative entries (223 observations) and 2.8% zero values (518 observations). These negative and zero values are physically implausible for particulate matter concentrations, indicating potential sensor errors or data logging issues that necessitate correction, such as replacing negative values with zeros. Furthermore, missing data is a concern, with PM10 exhibiting 5.1% missing values (954 entries) and PM2.5 showing 3.6% (677 entries), which could bias subsequent analyses unless mitigated through appropriate imputation techniques.

Further examination of the distributional properties reveals substantial challenges that necessitate data transformation for modeling. Both datasets exhibit high skewness (PM10: 2.93; PM2.5: 4.18) and elevated kurtosis (PM10: 45.58; PM2.5: 94.16), indicating right-skewed distributions with heavy tails and extreme outliers, as evidenced by the wide ranges (PM10: 1780.1 $\mu\text{g}/\text{m}^3$; PM2.5: 513.7 $\mu\text{g}/\text{m}^3$) and significant standard deviations (PM10: 28.06 $\mu\text{g}/\text{m}^3$; PM2.5: 14.67 $\mu\text{g}/\text{m}^3$). These characteristics are visually confirmed by the frequency distribution histograms, where Figure 6 illustrates the highly skewed distribution of PM10 data, and Figure 7 depicts a similar pattern for PM2.5 data, with both showing a concentration of values at the lower end and a long tail extending to extreme values.

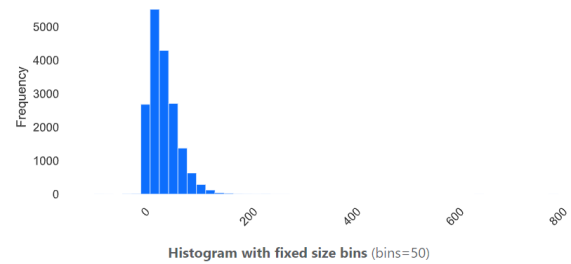


Figure 6. Frequency Distribution Histogram of PM10 Data

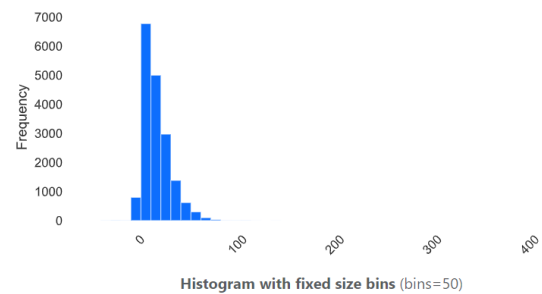


Figure 7. Frequency Distribution Histogram of PM2.5 Data

The third quartile (Q3) values—47.8 $\mu\text{g}/\text{m}^3$ for PM10 and 23.3 $\mu\text{g}/\text{m}^3$ for PM2.5—are substantially lower than their respective maximum values of 790.2 $\mu\text{g}/\text{m}^3$ and 472.8 $\mu\text{g}/\text{m}^3$, indicating the presence of extreme outliers that skew the data distribution. This discrepancy is further emphasized by the interquartile range (IQR) values of 34.5 $\mu\text{g}/\text{m}^3$ for PM10 and 17.8 $\mu\text{g}/\text{m}^3$ for PM2.5, which represent the spread of the middle 50% of the data; the large gaps between Q3 and the maximum values suggest that a small proportion of observations are driving the upper tail, as visually supported by the time series plots in Figure 8 for PM10 and Figure 9 for PM2.5, spanning 2019 to 2025, where sporadic peaks likely correspond to these outliers.

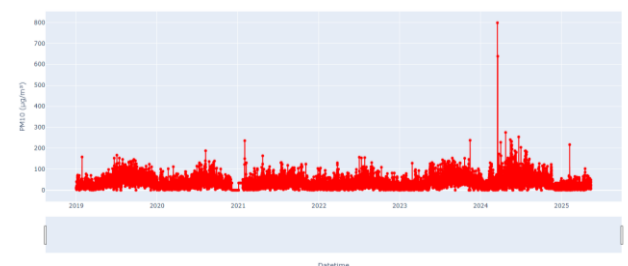


Figure 8. Time Series Plot of PM10 Concentrations From 2019 to 2025

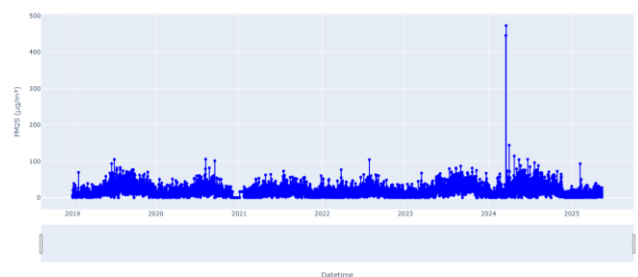


Figure 9. Time Series Plot of PM2.5 Concentrations From 2019 to 2025

The strong positive correlation between PM10 and PM2.5, with a Pearson correlation coefficient of 0.973, underscores a near-linear relationship where variations in one parameter are closely aligned with the other, as visually represented in the heatmap depicted in Figure 10. This high correlation indicates a significant shared variance, likely attributable to common pollution sources such as vehicular emissions or industrial processes. Given this robust interdependence, feature engineering can be effectively implemented, utilizing both PM10 and PM2.5 as input features to predict either parameter individually, leveraging the redundant information to enhance the accuracy and efficiency of air quality forecasting models.

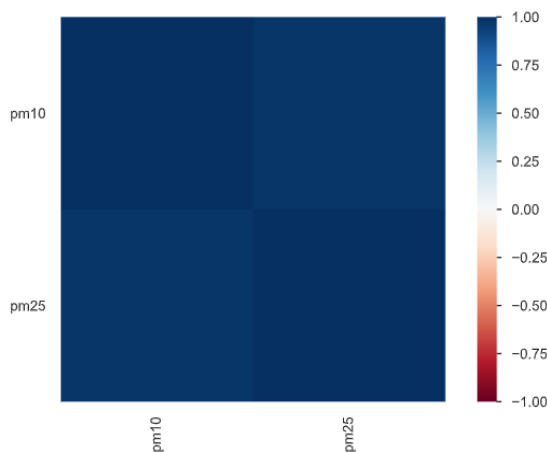


Figure 10. Heatmap Showing the Pearson Correlation Between PM2.5 and PM10 Concentrations

B. Data Pre-processing

Based on the results of the Exploratory Data Analysis, the dataset requires data imputation to address missing values, including negative values and zeros, to ensure data integrity for subsequent analyses. To this end, imputation is performed using the KNN imputation method with five neighbors (neighbors=5), a uniform weight function, and the nan_euclidean metric, which is well-suited for handling missing data in multivariate datasets. The KNN imputation method is one of the simplest and most popular approaches for addressing missing data, particularly in scenarios involving multiple variables, and has been widely implemented for imputing missing values in datasets with complex patterns. The KNN-I algorithm operates by identifying observations similar to those with missing values, leveraging the proximity of data points to estimate and fill in the gaps [32], [33].

To address the presence of outliers in the PM10 and PM2.5 datasets, as observed in the time series plots in Figure 7 and Figure 8, the Z-score method is employed to identify and remove extreme values that deviate significantly from the mean. The Z-score method calculates the number of standard deviations a data point is from the mean, enabling the elimination of outliers beyond a threshold of $|Z| > 3$, thereby mitigating the influence of extreme values on subsequent analyses. After applying the Z-score method to handle outliers, the cleaned datasets are visualized in the time series plots shown in Figure 11 for PM10 and Figure 12 for PM2.5, where

the maximum values are now reduced to 115.4 $\mu\text{g}/\text{m}^3$ for PM10 and 58.8 $\mu\text{g}/\text{m}^3$ for PM2.5, reflecting a more stable and representative range of concentrations.

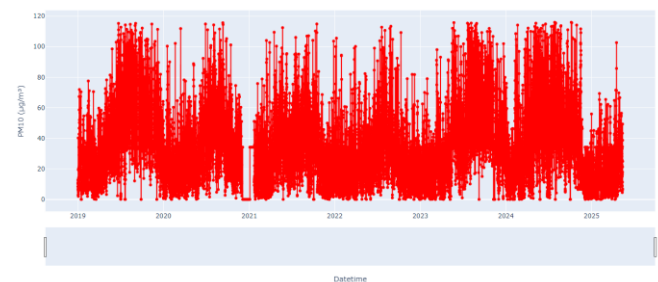


Figure 11. Time Series Plot of PM10 Concentrations From 2019 to 2025 After Outlier Removal Using the Z-Score Method

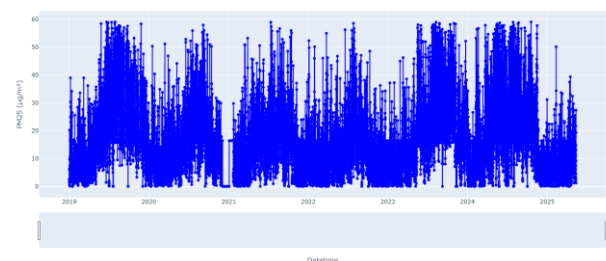


Figure 12. Time Series Plot of PM2.5 Concentrations From 2019 to 2025 After Outlier Removal Using the Z-score Method

To enhance the performance and effectiveness of the model, feature scaling is implemented using the Min-Max scaling method. Min-Max normalization is a data processing technique that transforms values within a dataset into a specific range, typically between 0 and 1, ensuring a uniform scale across all features (variables) to prevent features with larger values from exerting excessive dominance or influence on the analysis or models utilizing the data. The primary objective of this normalization is to standardize the dataset, thereby improving model convergence and accuracy. This technique transforms the range of the input data into 0–1 using Equation 4.

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (4)$$

Given the high correlation of 0.973 between the PM10 and PM2.5 datasets, indicating a strong linear relationship, both features are utilized to predict a single target feature, leveraging their interdependence to enhance predictive accuracy. To this end, feature engineering is performed, resulting in a set of 12 features to predict the two target features, PM2.5 and PM10. The engineered features include temporal components—namely 'hour', 'day', 'month', and 'weekday'—to capture time-based patterns, as well as lagged variables such as 'pm25_lag1', 'pm10_lag1', 'pm25_lag2', and 'pm10_lag2', which account for the previous time steps of PM2.5 and PM10 concentrations. Additionally, rolling mean

features, 'pm25_rolling_mean' and 'pm10_rolling_mean', are incorporated to represent the short-term trends in the data, thereby providing a comprehensive feature set that captures both temporal dynamics and historical patterns.

C. Forecasting Performance and Evaluation

1. Performance Evaluation of the Long Short-Term Memory (LSTM) Model

The LSTM model, detailed in Table 3, was configured with a two-layer architecture, comprising 128 units in the first LSTM layer followed by 64 units in the second. This structure aims to capture temporal dependencies of varying complexities within the 7-day input sequences ($\text{seq_length} = 7$) across 12 features. To mitigate overfitting, a substantial Dropout rate of 0.4 was applied after each LSTM layer, and L2 regularization (0.01) was implemented on both LSTM layers and the subsequent Dense layer (64 units). Notably, the 'relu' activation function was utilized within these layers. The model employed the Adam optimizer with a standard learning rate of 0.001. Training was conducted over a maximum of 100 epochs with a batch size of 32, incorporating Early Stopping ($\text{patience}=20$) and ReduceLROnPlateau ($\text{factor}=0.5$, $\text{patience}=5$) callbacks to ensure convergence and prevent prolonged training without improvement. Despite these measures, the model yielded moderate performance (Table 3), suggesting that while regularization likely prevented severe overfitting, the chosen architecture or the Adam optimizer settings might not have been fully optimal for extracting the predictive signals, resulting in R^2 values in the 0.62-0.67 range.

Table 3. Validation and Test Metrics for PM2.5 and PM10 Using LSTM

Metrics	RMSE	MAE	R^2
Validation PM2.5	8.43	6.05	0.62
Validation PM10	16.60	12.14	0.58
Test PM2.5	7.73	5.18	0.66
Test PM10	14.35	10.16	0.67

2. Performance Evaluation of the Gated Recurrent Unit (GRU) Model

Similar to the LSTM, the GRU model (Table 4) utilized a two-layer structure (128 then 64 units) and identical sequence length (7), Dropout (0.4), L2 regularization (0.01), Dense layer size (64), activation functions ('relu'), and training protocols (100 epochs, batch size 32, Early Stopping, ReduceLROnPlateau). The primary distinguishing factor was the optimizer: Stochastic Gradient Descent (SGD) was employed, configured with a learning rate of 0.01, decay of $1e-5$, momentum of 0.9, and Nesterov acceleration. Despite the change in optimization strategy, the GRU model's performance metrics (Table 4) were remarkably similar to those of the LSTM model. This indicates that for this specific problem and dataset structure, the architectural limitations or the nature of the data likely played a more significant role in determining performance than the choice between Adam and Nesterov-accelerated SGD, with both models achieving comparable, moderate R^2 scores (0.57-0.68).

Table 4. Validation and Test Metrics for PM2.5 and PM10 Using GRU

Metrics	RMSE	MAE	R^2
Validation PM2.5	8.40	6.02	0.62
Validation PM10	16.63	12.15	0.57
Test PM2.5	7.71	5.21	0.67
Test PM10	14.27	10.13	0.68

3. Performance Evaluation of the Random Forest Time Series Model

The Random Forest model (Table 5) employed a markedly different, non-sequential approach. Its configuration was relatively straightforward, primarily defined by $n_estimators$ set to 100. This parameter dictates that 100 individual decision trees were constructed and their predictions averaged to yield the final forecast. A random_state of 42 was set to ensure the reproducibility of results. Despite its structural simplicity compared to the deep learning models, Random Forest achieved significantly superior performance. The high R^2 values (0.79-0.85) demonstrate its strong capability in capturing the underlying relationships within the features, even when presented as a time-series-derived tabular dataset. This suggests that an ensemble-of-trees approach was highly effective for this regression task, outperforming the RNNs.

Table 5. Validation and Test Metrics for PM2.5 and PM10 Using Random Forest Time Series

Metrics	RMSE	MAE	R^2
Validation PM2.5	5.41	3.70	0.84
Validation PM10	11.54	8.27	0.79
Test PM2.5	5.11	3.1	0.85
Test PM10	10.07	6.64	0.83

4. Performance Evaluation of the XGBoost Regression Model

The XGBoost model (Table 6) utilized a gradient boosting framework, configured for regression with the 'reg:squarederror' objective function. Like Random Forest, it built an ensemble of 100 estimators ($n_estimators$). Key hyperparameters included a learning_rate of 0.1, which moderates the contribution of each new tree, and a max_depth of 5, limiting the complexity of individual trees to prevent overfitting. A random_state of 42 was also used for consistency. The XGBoost model delivered the highest performance overall, particularly in achieving the lowest error metrics (RMSE, MAE) and the highest R^2 values (up to 0.85). This superior performance likely stems from the boosting mechanism, where subsequent trees are built to correct the errors of prior ones, combined with the regularization effects implicitly managed by the learning rate and maximum depth. These settings allowed the model to effectively capture complex patterns while maintaining good generalization, positioning it as the most robust model in this comparative study.

Table 6. Validation and Test Metrics for PM2.5 and PM10 Using XGBoost Regression

Metrics	RMSE	MAE	R ²
Validation PM2.5	5.36	3.73	0.84
Validation PM10	11.10	8.00	0.81
Test PM2.5	5.05	3.19	0.85
Test PM10	9.71	6.55	0.85

5. Analysis of the Best Performing Model

Based on a comprehensive evaluation, the XGBoost Regression model unequivocally stands out as the most effective for forecasting PM2.5 and PM10 in this study. It achieved the lowest Root Mean Square Error (RMSE) (5.05 for PM2.5; 9.71 for PM10) and Mean Absolute Error (MAE) (3.19 for PM2.5; 6.55 for PM10), coupled with the highest coefficient of determination (R²) of 0.85 for both pollutants. This statistical superiority confirms its strong predictive capability. While its performance is closely followed by Random Forest and significantly surpasses the LSTM and GRU models, its optimized configuration (n_estimators=100, learning_rate=0.1, max_depth=5) appears best suited for this task.

Table 7. Comparison of Actual and Predicted PM2.5 using Best Performing Model

Datetime	Actual	Predicted	Error
2025-05-13 6:00:00	13.2	14.9	1.7
2025-05-13 9:00:00	27.4	20.4	7.0
2025-05-13 12:00:00	19.3	20.7	1.4
2025-05-13 15:00:00	9.2	11.0	1.8
2025-05-13 18:00:00	1.2	8.0	6.8

Table 8. Comparison of Actual and Predicted PM10 using Best Performing Model

Datetime	Actual	Predicted	Error
2025-05-13 6:00:00	21.2	24.4	3.2
2025-05-13 9:00:00	49.3	35.8	13.5
2025-05-13 12:00:00	38.7	43.3	4.6
2025-05-13 15:00:00	24.9	23.1	1.8
2025-05-13 18:00:00	6.8	22.2	15.4

Despite its overall robustness and strong visual correlation in Figure 13, presented in Tables 7 and 8 reveals that while many predictions are highly accurate with errors as low as 1.4 µg/m³ (for PM2.5) and 1.8 or 3.2 µg/m³ (for PM10), instances with larger deviations do occur. We now observe errors as high as 7.0 µg/m³ for PM2.5, and particularly 13.5 µg/m³ and 15.4 µg/m³ for PM10 at 9:00 and 18:00 respectively. This is consistent with an R² value of 0.85 – indicating a strong, yet not perfect, correlation, and highlighting that predicting PM10 can sometimes involve larger errors. Nevertheless, these occasional discrepancies do not diminish the overall conclusion. The combination of leading statistical metrics and the demonstrated ability to capture key pollutant dynamics confirms that the XGBoost Regression model offers the most reliable and accurate forecasting performance among the models tested for this specific air quality dataset.

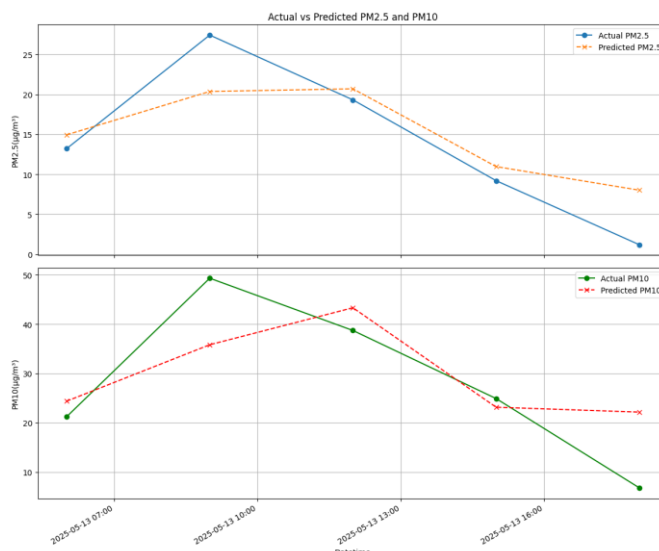


Figure 13. Time Series Plot of Best Performing Model Predictions vs. Actual PM2.5 and PM10 Data

IV. CONCLUSION

Based on a comparative analysis using the SISANAPAS platform, this study evaluated Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Random Forest Time Series, and XGBoost Regression models for forecasting PM2.5 and PM10 concentrations. The results unequivocally identified XGBoost Regression as the superior model, achieving the lowest Root Mean Square Error (RMSE) (5.05 for PM2.5; 9.71 for PM10), the lowest Mean Absolute Error (MAE) (3.19 for PM2.5; 6.55 for PM10), and the highest coefficient of determination (R²) of 0.85 for both pollutants. While Random Forest also performed strongly, significantly outpacing the deep learning models, both LSTM and GRU delivered only moderate performance despite optimization, with R² values ranging from 0.57 to 0.68.

The strong performance of XGBoost underscores the criticality of meticulous data pre-processing, including K-Nearest Neighbors (KNN) imputation and Z-score outlier removal, alongside effective feature engineering. Although the 0.85 R² value indicates a robust model, granular analysis revealed occasional larger prediction deviations, confirming that while highly reliable, perfect correlation was not achieved. Nevertheless, this study confirms XGBoost Regression as the most dependable and accurate approach among those tested for this dataset and validates the SISANAPAS platform's utility, offering valuable insights for evidence-based air quality management policies.

REFERENCES

- [1] A. Rachmawardani, D. Prabowo, Nardi, K. L. Toruan, S. Abdurachman, and D. B. Adinara, "Designing a Cost-Effective IoT Air Quality Sensor for Real-Time Data Monitoring," *2024 IEEE Int. Conf. Comput. ICOCO 2024*, pp. 219–224, 2024, doi:

- 10.1109/ICOCO62848.2024.10928212.
- [2] IQAir, "World Air Quality Report 2023," *IQAir*, pp. 1–45, 2023, [Online]. Available: <https://www.iqair.com/world-most-polluted-countries>
- [3] Z. C. Winni and S. Mataram, "Perancangan Zine Penggunaan Transportasi Umum Guna Mengurangi Polusi Udara Bagi Dewasa Di Jakarta," *Tuturrupe*, vol. 6, no. 2, pp. 101–110, 2024, doi: 10.24167/tuturrupe.v6i2.11373.
- [4] U.S. Environmental Protection Agency, "Particulate Matter (PM) Basics." Accessed: May 07, 2025. [Online]. Available: <https://www.epa.gov/pm-pollution/particulate-matter-pm-basics#PM>
- [5] U.S. Environmental Protection Agency, "Technical Assistance Document for the Reporting of Daily Air Quality – the Air Quality Index (AQI)," Research Triangle Park, NC, 2024.
- [6] H. Kim *et al.*, "Cardiovascular effects of long-term exposure to air pollution: a population-based study with 900 845 person-years of follow-up," *J. Am. Heart Assoc.*, vol. 6, no. 11, 2017, doi: 10.1161/jaha.117.007170.
- [7] S. M. Daryanoosh, G. Goudarzi, M. J. Mohammadi, H. Armin, Y. O. Khaniabadi, and S. Sadeghi, "Exposure to particulate matter and its health impacts (an airq approach)," *Arch. Hyg. Sci.*, vol. 6, no. 1, pp. 88–95, 2017, doi: 10.29252/archhygsci.6.1.88.
- [8] J. D. Sacks *et al.*, "Particulate matter–induced health effects: who is susceptible?," *Environ. Health Perspect.*, vol. 119, no. 4, pp. 446–454, 2011, doi: 10.1289/ehp.1002255.
- [9] J. Li *et al.*, "Major air pollutants and risk of copd exacerbations: a systematic review and meta-analysis," *Int. J. Chron. Obstruct. Pulmon. Dis.*, vol. Volume 11, pp. 3079–3091, 2016, doi: 10.2147/copd.s122282.
- [10] W. Yang, G. Tang, Y. Hao, and J. Wang, "A novel framework for forecasting, evaluation and early-warning for the influence of PM10 on public health," *Atmosphere (Basel)*, vol. 12, no. 8, pp. 1–20, 2021, doi: 10.3390/atmos12081020.
- [11] P. Sekula, Z. Ustrnul, A. Bokwa, B. Bochenek, and M. Zimnoch, "Random Forests Assessment of the Role of Atmospheric Circulation in PM10 in an Urban Area with Complex Topography," *Sustain.*, vol. 14, no. 6, pp. 1–43, 2022, doi: 10.3390/su14063388.
- [12] A. Pant, S. Sharma, and K. Pant, "Evaluation of Machine Learning Algorithms for Air Quality Index (AQI) Prediction," *J. Reliab. Stat. Stud.*, vol. 16, no. 2, pp. 229–242, 2023, doi: 10.13052/jrss0974-8024.1621.
- [13] Y. T. Tsai, Y. R. Zeng, and Y. S. Chang, "Air pollution forecasting using rnn with lstm," *Proc. - IEEE 16th Int. Conf. Dependable, Auton. Secur. Comput. IEEE 16th Int. Conf. Pervasive Intell. Comput. IEEE 4th Int. Conf. Big Data Intell. Comput. IEEE 3*, pp. 1068–1073, 2018, doi: 10.1109/DASC/PiCom/DataCom/CyberSciTec.2018.00178.
- [14] R. Amelia, Guskarnali, R. G. Mahardika, C. R. Niani, and N. Lewaherilla, "Predicting particulate matter PM2.5 using the exponential smoothing and Seasonal ARIMA with R studio," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 1108, no. 1, 2022, doi: 10.1088/1755-1315/1108/1/012079.
- [15] N. Zaini, L. W. Ean, A. N. Ahmed, M. Abdul Malek, and M. F. Chow, "PM2.5 forecasting for an urban area based on deep learning and decomposition method," *Sci. Rep.*, vol. 12, no. 1, pp. 1–13, 2022, doi: 10.1038/s41598-022-21769-1.
- [16] Furizal, A. Ma'arif, I. Suwarno, A. Masitha, L. Aulia, and A. N. Sharkawy, "Real-Time Mechanism Based on Deep Learning Approaches for Analyzing the Impact of Future Timestep Forecasts on Actual Air Quality Index of PM10," *Results Eng.*, vol. 24, no. November, 2024, doi: 10.1016/j.rineng.2024.103434.
- [17] H. Altınçöp and A. B. Oktay, "Air Pollution Forecasting with Random Forest Time Series Analysis," *2018 Int. Conf. Artif. Intell. Data Process. IDAP 2018*, pp. 8–12, 2019, doi: 10.1109/IDAP.2018.8620768.
- [18] Z. Alfasanah, M. Z. H. Niam, S. Wardiani, M. Ahsan, and M. H. Lee, "Monitoring air quality index with EWMA and individual charts using XGBoost and SVR residuals," *MethodsX*, vol. 14, no. August 2024, 2025, doi: 10.1016/j.mex.2024.103107.
- [19] YData, "YData Profiling Documentation." 2024. [Online]. Available: <https://docs.profiling.ydata.ai/>
- [20] IQAir, "Indonesia Air Quality Index (AQI) and Air Pollution Information." 2025. [Online]. Available: <https://www.iqair.com/indonesia>
- [21] S. Arifin *et al.*, "Long short-term memory (lstm): trends and future research potential," *Int. J. Emerg. Technol. Adv. Eng.*, vol. 13, no. 5, pp. 24–34, 2023, doi: 10.46338/ijetae0523_04.
- [22] A. Sherstinsky, "Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network," *Phys. D Nonlinear Phenom.*, vol. 404, no. March, pp. 1–43, 2020, doi: 10.1016/j.physd.2019.132306.
- [23] S. Mohsen, "Recognition of human activity using GRU deep learning algorithm," *Multimed. Tools Appl.*, vol. 82, no. 30, pp. 47733–47749, 2023, doi: 10.1007/s11042-023-15571-y.
- [24] Y. Heryadi, *Dasar-Dasar Deep Learning dan Implementasinya*, I. Yogyakarta: Gava Media, 2021.
- [25] A. Rachmawardani, S. K. Wijaya, and A. Shopaheluwakan, "Sistem Peringatan Dini Banjir Berbasis Machine Learning: Studi Literatur," *METHOMIKA J. Manaj. Inform. dan Komputerisasi Akunt.*, vol. 6, no. 6, pp. 188–198, 2022, doi: 10.46880/jmika.vol6no2.pp188-198.
- [26] M. Diaz Resquin *et al.*, "A machine learning approach to address air quality changes during the COVID-19 lockdown in Buenos Aires, Argentina," *Earth Syst. Sci. Data*, vol. 15, no. 1, pp. 189–209, 2023, doi: 10.5194/essd-15-189-2023.
- [27] C. X. Lv, S. Y. An, B. J. Qiao, and W. Wu, "Time series analysis of hemorrhagic fever with renal syndrome in



- mainland China by using an XGBoost forecasting model,” *BMC Infect. Dis.*, vol. 21, no. 1, pp. 1–13, 2021, doi: 10.1186/s12879-021-06503-y.
- [28] A. Al Mamun, M. Sohel, N. Mohammad, M. S. Haque Sunny, D. R. Dipta, and E. Hossain, “A Comprehensive Review of the Load Forecasting Techniques Using Single and Hybrid Predictive Models,” *IEEE Access*, vol. 8, pp. 134911–134939, 2020, doi: 10.1109/ACCESS.2020.3010702.
- [29] G. E. Saulnier, J. C. Castro, and C. B. Cook, “Forecasting inpatient glycemic control: extension of damped trend methods to subpopulations,” *Futur. Sci. OA*, vol. 6, no. 10, 2020, doi: 10.2144/fsoa-2020-0096.
- [30] T. Esaki, “Appropriate evaluation measurements for regression models,” *Chem-Bio Informatics J.*, vol. 21, no. 0, pp. 59–69, 2021, doi: 10.1273/cbij.21.59.
- [31] S. Inayati, N. Iriawan, and I. Irhamah, “A markov switching autoregressive model with time-varying parameters,” *Forecasting*, vol. 6, no. 3, pp. 568–590, 2024, doi: 10.3390/forecast6030031.
- [32] A. F. Sallaby and A. Azlan, “Analysis of Missing Value Imputation Application with K-Nearest Neighbor (K-NN) Algorithm in Dataset,” *Int. J. Informatics Comput. Sci.*, vol. 5, no. 2, pp. 141–144, 2021.
- [33] I. J. Fadillah and C. D. Puspita, “Pemanfaatan Metode Weighted K-Nearest Neighbor Imputation (Weighted Knni) Untuk Mengatasi Missing Data,” in *Seminar Nasional Official Statistics*, 2020, pp. 511–518.