ComTech: Computer, Mathematics and Engineering Applications, 17(1), June 2026, 39−51
DOI: 10.21512/comtech.v17i1.14218
P-ISSN: 2087-1244
E-ISSN: 2476-907X

Forecasting Food Prices in East Java Using Stacking Ensemble Learning via K-MEANS

Aviolla Terza Damaliana[1]*; Amri Muhaimin[2]; Nabilah Selayanti[3]; Shafira Amanda Putri[4]; Muhammad Nasrudin[5]

[1-5]Data Science, Computer Science, UPN "Veteran" Jawa Timur, Surabaya, Indonesia, 60294

[1]aviolla.terza.sada@upnjatim.ac.id; [2]amri.muhaimin.stat@upnjatim.ac.id; [3]22083010013@student.upnjatim.ac.id; [4]22083010008@student.upnjatim.ac.id; [5]nasrudin.fasilkom@upnjatim.ac.id

Received: 15th August 2025/ **Revised:** 29th December 2025/ **Accepted:** 8th January 2026

How to Cite: Damaliana, A. T., Muhaimin, A., Selayanti, N., Putri, S. A., & Nasrudin, M. (2026). Forecasting food prices in East Java using stacking ensemble learning via K-MEANS. *ComTech: Computer, Mathematics and Engineering Applications, 17*(1), 39−51. https://doi.org/10.21512/comtech.v17i1.14218

IN PRESS

***Abstract* -** Food commodities are essential in developing countries such as Indonesia, and the government regulates food commodity prices in every province. However, price instability issues persist in certain provinces, creating challenges for effective policy control. Data science and statistical techniques play an important role in supporting the government's efforts to monitor and manage food commodity prices. This study proposes the Stackelberg-K-Means method to predict the commodity price index in East Java. The proposed method is a collaborative framework that combines cluster analysis and stacking ensemble learning for time-series prediction. Cluster analysis is conducted first using Dynamic Time Warping as the distance measure, which is suitable for time-series data, resulting in two clusters for each commodity: rice, oil, and flour. The stacking model consists of base learners and a meta-learner. The base learner models include Ridge Regression, Random Forest, and Support Vector Regression, while the meta-learner uses Light Gradient Boosting. Parameter optimization is performed using grid search, and the proposed method is evaluated against AutoARIMA implemented in Python using both training and testing data. The results show that the proposed method outperforms the ARIMA model across all three error metrics: MAPE, MAE, and RMSE. For flour commodities, the scores are 0.042% versus 0.328%, 4.715 versus 37.57, and 6.34 versus 523.99, respectively. For rice commodities, the scores are 0.261% compared to 0.392%, 31.585 compared to 48.142, and 41.92 compared to 56.068. For oil commodities, the scores are 0.185% compared to 0.250%, 33.02 compared to 47.571, and 39.35 compared to 56.060.

Keywords: clustering, ensemble, food commodity, price, time-series

I. INTRODUCTION

Food is a basic human need that supports survival and well-being. In Indonesia, basic daily necessities include rice, shallots, garlic, red chilies, cayenne pepper, beef, chicken, granulated sugar, wheat flour, cooking oil, soybeans, and eggs. These goods are referred to as food commodities (Farisi et al., 2022). For these commodities, prices rise when demand exceeds supply and fall when supply exceeds demand. Such increases or decreases reflect fluctuations in food commodity prices (Putri Z et al., 2024)..

Fluctuating food commodity prices impact economic development and political stability (Wang et al., 2022). In 1998, a significant increase in food commodity prices precipitates a multidimensional crisis in Indonesia that threatens economic and national stability. Given the importance of food commodity prices for economic and political stability, forecasting future prices is crucial because it can mitigate the impact of significant price changes and help stabilize commodity prices (Hasibuan & Novialdi, 2022). Using appropriate forecasting methods is vital, as more accurate forecasts support the government's decision-making in pricing policies.

This study uses the prices of several food commodities in districts and cities in East Java Province, namely medium-grade rice, bulk cooking oil, and wheat flour. East Java is chosen because it is one of the provinces in Indonesia with the most significant production of several leading food commodities

(Mardianto et al., 2023). East Java Province consists of 38 regencies or cities with varying food commodity prices. Forecasting based on the average provincial price across all regions prevents the identification of variations in price patterns and limits the ability to capture hidden patterns in the data (Amatullah et al., 2025). Therefore, this study clusters regencies or cities before conducting forecasting. Another purpose of clustering is to improve the efficiency of government policies and eliminate the need for separate policy management for each region (Zen et al., 2022).

Previous research on time-series clustering of food prices is conducted by Zen et al. (2022), who use cooking oil price data from Indonesia. This study applies Agglomerative Hierarchical Clustering (AHC) as the clustering method and ARIMA as the forecasting method (Zen et al., 2022). A similar approach is also applied by Amatullah et al. (2025) using granulated sugar price data. In addition to food price data, time-series clustering using the same method is applied by Yohansa et al. (2022) to COVID-19 data in DKI Jakarta. That study evaluates the effectiveness of forecasting with and without clustering and shows that clustering enables each sub-district to form a robust structure in representing the data during the clustering process (Yohansa et al., 2022).

In addition to the AHC algorithm for clustering, K-Means can also be used for time-series data. Previous research applies the K-Means algorithm to food commodity price data in Indonesia (Mardianto et al., 2023). However, that study focuses only on the clustering process and does not include a forecasting stage. Several prior studies also use the Dynamic Time Warping (DTW) similarity measure, which has the advantage of comparing two or more time series with different lengths or frequencies without being affected by length variations or time shifts in data patterns (Šťastný et al., 2022).

Another similarity measure applicable to time-series data is Soft-DTW, which previous studies show to outperform the Distance Barycenter Averaging (DBA) similarity measure (Li et al., 2022). Therefore, the novelty of this research lies in applying the K-Means algorithm in the clustering process and comparing DTW and Soft-DTW similarity measures based on their silhouette scores. Previous studies commonly use the ARIMA method (Amatullah et al., 2025; Yohansa et al., 2022; Zen et al., 2022) and several machine learning methods, such as K-nearest neighbor regression and Random Forest (Šťastný et al., 2022). A key limitation of the ARIMA method is that it assumes linear relationships between past values, previous prediction errors, and current data. As a result, ARIMA does not perform optimally when the data are complex and exhibit nonlinear patterns over time (Swaraj et al., 2021). Moreover, commodity price data are influenced by multiple factors that generate complex and nonlinear patterns, making ARIMA-based approaches less effective for this problem (Wang et al., 2022).

Machine learning approaches to forecasting, as implemented by Šťastný et al. (2022), capture nonlinear patterns and offer satisfactory prediction accuracy even for datasets with limited size and dimensionality (Mandal et al., 2023). However, relying on a single machine learning model limits learning flexibility and model robustness (Mandal et al., 2023). To achieve higher accuracy, one effective approach is Stacking Ensemble Learning (Ismail et al., 2023). This method combines multiple base learners and uses a meta-learner to integrate their predictions, enabling more flexible learning and improved predictive performance.

According to Ismail et al. (2023), stacking ensemble learning improves model accuracy by combining the strengths of multiple models and mitigating their individual weaknesses. Another advantage of stacking ensemble learning is its ability to efficiently reduce variance and bias, thereby helping to avoid overfitting (Khan et al., 2022). Based on previous studies, the application of stacking ensemble learning for forecasting on data that are first clustered using K-Means remains limited. Therefore, the novelty of this research lies in integrating K-Means clustering with stacking ensemble learning to forecast food prices in East Java Province, a proposed approach hereinafter referred to as STACKEL K-MEANS.

II. METHODS

This section explains the research methodology, as illustrated in Figure 1. This research aims to develop a robust forecasting methodology through clustering analysis. Initially, cluster analysis is conducted on all attribute data, namely the prices of food commodities such as wheat flour and medium-grade rice in 38 districts or cities in East Java, as well as cooking oil in 37 districts or cities. The data are collected from the website siskaperbapo.jatim-prov.go.id using the Selenium scraping method. The daily data collection period spans from January 1, 2024, to April 30, 2025, resulting in 487 data points per district or city. Based on this collection, wheat flour and medium-grade rice each have 18,006 data points, while cooking oil has 18,019 data points. Initially, the data are collected at the market level in each specific city.

As shown in Figure 1, the second step is data preprocessing. This step involves aggregating the data using the mean value to obtain representative prices. For example, Surabaya has seven recorded markets, and prices from these markets are aggregated to create a single vector price index representing Surabaya. The same aggregation process is applied to the remaining 37 districts or cities in East Java. The third step is cluster analysis, which is conducted for districts or cities in East Java, as illustrated in Figure 2. The clustering process produces new grouped data for each food commodity, resulting in several clusters. Actual prices from districts or cities within the same cluster are averaged, producing daily data for each food

IN PRESS

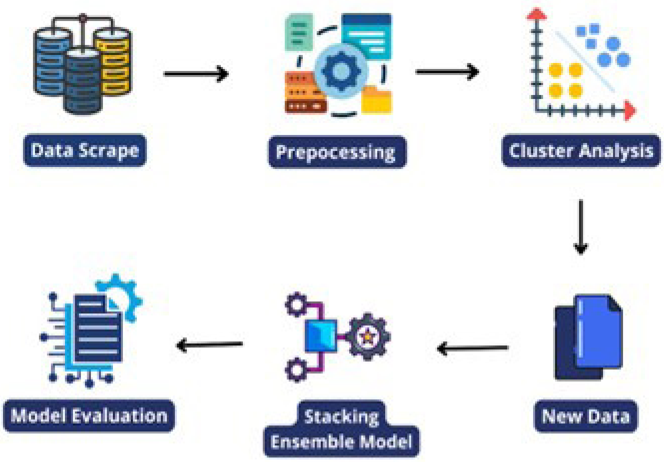
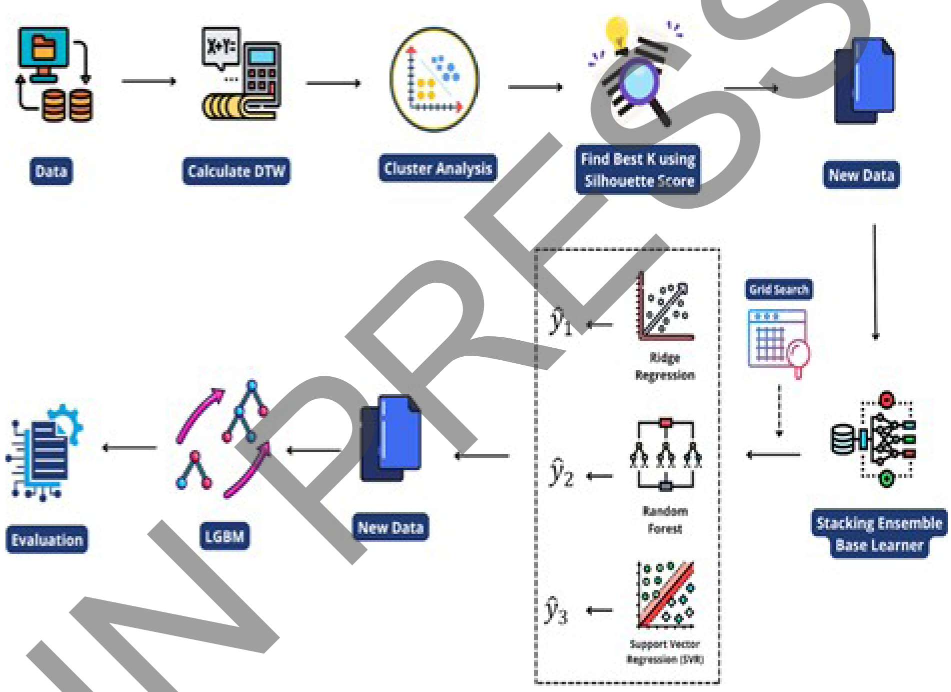


Figure 1 Research Flow Chart



IN PRESS

Figure 2 Proposed Method: STACKEL K-MEANS

commodity cluster that are used in the subsequent forecasting stage.

Before modeling, the newly generated data are divided into training and testing sets. Each dataset formed from the clustering process uses 479 data points for training and 7 data points for testing. These seven testing data points are considered sufficient for forecasting evaluation because the data are recorded on a daily basis. The next stage applies stacking ensemble learning, followed by model evaluation. Detailed explanations of the clustering framework, stacking ensemble learning model, and evaluation metrics are presented in Figures 2 and 3.

As shown in Figure 2, this study applies a time-adaptive distance measure, Dynamic Time Warping (DTW), to perform clustering analysis on time-series data. DTW is a robust method for calculating similarity between two time-series datasets (Sardá-Espinosa, 2019). Its main advantage lies in its ability to flexibly shift and warp the time axis, making it suitable for comparing time-series data that evolve over time (Hegg & Kennedy, 2021).

DTW calculates a cost matrix based on the cumulative dissimilarity between points in two time-series datasets (Herrmann et al., 2023). It then computes the minimum distances required to optimally align the pair of time-series data. The algorithm also generates nonlinear alignments, allowing sections of the time series to be stretched or compressed to achieve the best possible matching (Folgado et al., 2018). The

DTW formula is presented in Equation (1).

Equation (1) defines the cost matrix used to compute the DTW value. The time series are denoted as x_i and y_j, and the distance used is the pointwise distance, which is defined in Equation (2). The minimum terms represent the optimal previous alignment by allowing movement upward, sideways, or diagonally within the matrix. The development of DTW is extensive, and this research also compares DTW with Soft-DTW (Xu et al., 2023).

Soft-DTW uses a soft-minimum function to select the optimal alignment point, which makes the objective function easier to optimize in machine learning models. This property makes Soft-DTW particularly useful when integrating DTW with gradient-based methods, such as neural networks (Cuturi & Blondel, 2017). The Soft-DTW formula is presented in Equation (3).

The soft minimum is calculated by exponentiating the previous cumulative distances and applying a logarithmic transformation. The parameter γ acts as a smoothing factor that controls the softness of the value and can also be interpreted as a regularization parameter (Cuturi & Blondel, 2017). After computing distances using DTW, K-Means clustering is applied to partition the data into several clusters. Using Equation (4), the c_k represents the centroid of the $k\text{−}th$ cluster obtained from the data. In this study, Soft-DTW is employed as the distance measure within the K-Means algorithm. K-Means is chosen because it partitions data based on centroids, and the robustness of DTW-based distances over time makes them well suited for time-series clustering. Furthermore, the Silhouette score, as defined in Equation (5), is used to determine the optimal number of clusters.

The Silhouette measure ($S(i)$) evaluates cluster quality by calculating the difference between intra-cluster similarity and the similarity to the nearest neighboring cluster (Shahapure & Nicholas, 2020). A higher Silhouette value indicates better clustering performance, with values ranging from −1 to 1. Although there is no fixed threshold for an optimal Silhouette score, values greater than 0.4 are commonly considered to indicate well-formed clusters (Maulidya et al., 2024). The groups produced from the clustering analysis are then used to generate new data by aggregating the values within each cluster. These aggregated data serve as the input for the ensemble learning stage.

The ensemble learning approach applied in this study is stacking ensemble learning, which consists of a two-phase modeling process: base learners and a meta-learner (Kwon et al., 2019). The base learner models include Ridge Regression, Random Forest, and Support Vector Regression, while the meta-learner employs the Light Gradient Boosting Method (LGBM). This combination allows the model to capture both linear and nonlinear characteristics in the data, with Ridge Regression modeling linear patterns and the other models addressing nonlinear relationships (James et al., 2013; Muhaimin et al., 2021). The meta-learner is required to efficiently integrate the predictions of the base learners, and LGBM is well suited for this task due to its optimized gradient boosting framework (Ke et al., 2017). In addition, grid search optimization is applied at each modeling stage to determine optimal parameters. Model inputs are derived from lagged data identified using the Partial Autocorrelation Function (PACF) method, and only statistically significant lags based on PACF are used as input variables for the stacking ensemble model (Hassani et al., 2024; Kumar et al., 2024).

As mentioned earlier, Ridge Regression is a linear model that uses regularization to address overfitting and handle multicollinearity. The regularization applied in parameter estimation is based on the Ordinary Least Squares (OLS) framework with an added penalty term. By incorporating regularization, Ridge Regression complements the stacking ensemble model by capturing linear patterns in the data (Renju & Brunda, 2024). The Ridge Regression formula is presented in Equation (6). In Equation (6), the λ represents the regularization parameter used to prevent overfitting, where larger λ values impose stronger penalties and result in smaller coefficient estimates (Pavlou et al., 2024). The objective of Ridge Regression is to minimize the penalized loss function in order to estimate the parameter θ_j (Lukman & Olatunji, 2018).

IN PRESS

$$D(i,j) = distance(x_i, y_j) + min\{D(i-1,j), D(i,j-1), D(i-1,j-1) \tag{1}$$

$$distance(x_i, y_j) = |x_i - y_j| \tag{2}$$

$$softDTW^{\gamma}(\mathbf{x}, \mathbf{x}') = \min_{\pi \in A(\mathbf{x},\mathbf{x}')}{}^{\gamma} \sum_{(i,j) \in \pi} d(x_i, x_j')^2 \min{}^{\gamma}(a_1, \dots, a_n) = -\gamma \log \sum_i e^{-a_i/\gamma} \tag{3}$$

$$cluster(x_i) = dist(x_i, c_k) \tag{4}$$

$$S(i) = \frac{b(i) - a(i)}{(a(i), b(i))} \tag{5}$$

$$J(\theta) = \sum_{i=1}^{m}(y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{n} \theta_j^2$$
$$\theta = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y} \tag{6}$$

Another base learner used in this study is Random Forest. Random Forest is an ensemble learning method whose fundamental component is the decision tree, and it can operate effectively under both linear and nonlinear conditions (Talekar, 2020). This capability allows Random Forest to contribute to the proposed model by providing robust and stable predictive performance. A Random Forest model consists of several key parameters, including the number of trees, the maximum tree depth, and the minimum number of samples per leaf. This method inherently applies an ensemble learning strategy through resampling to improve generalization performance. The Random Forest formulation is presented in Equation (7).

$$\hat{y} = \frac{1}{n_{leaf}} \sum_{i \epsilon leaf} y_i \tag{7}$$

In this study, the Random Forest model produces the final prediction by aggregating the outputs from multiple decision trees. The aggregated result is then normalized by the number of samples in the corresponding leaf node, which is denoted as n_{leaf}. The final base learner used in the stacking ensemble is Support Vector Regression (SVR). SVR is well suited for modeling nonlinear patterns in the data, which is why it is selected to capture complex relationships that cannot be represented by linear models (Suresh et al., 2021). The SVR formulation is presented in Equation (8). This method uses a kernel-based function with a bias term to predict new values

$$f(x) = \langle w, x \rangle + b \tag{8}$$

Support Vector Regression (SVR) employs an epsilon-insensitive loss function that depends on the parameter ε, as defined in Equation (9). This loss function is incorporated into an objective function that is optimized during model training to obtain the best predictive performance. The objective function aims to minimize both the prediction error and model complexity. The SVR objective function consists of a regularization term, denoted as $||w||^2$, which represents the norm of the weight vector and controls model smoothness. In addition, the function includes a parameter c that regulates the trade-off between maximizing the margin and penalizing errors that fall outside the ε-margin. Furthermore, SVR applies the kernel trick to map the input data into a higher-dimensional feature space, enabling the model to generate fitted values for nonlinear relationships.

$$L_\varepsilon(y, \hat{y}) = \begin{cases} 0 & \text{if } |y - \hat{y}| \le \varepsilon \\ |y - \hat{y}| - \varepsilon & \text{if} |y - \hat{y}| > \varepsilon \end{cases} \tag{9}$$

After the modeling process is complete, the model is evaluated using the one-on-one scenario illustrated in Figure 3, and the proposed approach is benchmarked against AutoARIMA implemented in Python. AutoARIMA is a Python-based implementation of the conventional ARIMA procedure that automatically selects the optimal model using the Akaike Information Criterion (AIC). The results produced by STACKEL K-MEANS are compared using evaluation metrics such as MAE, RMSE, and MAPE, which are defined in Equations (10)–(12) (Hodson, 2022). All evaluation metrics are formulated such that lower values indicate better performance. The expected outcome is that STACKEL K-MEANS outperforms AutoARIMA across all evaluated datasets.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}}{y_i}\right| \tag{10}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{11}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}}{y_i}\right| \times 100 \tag{12}$$

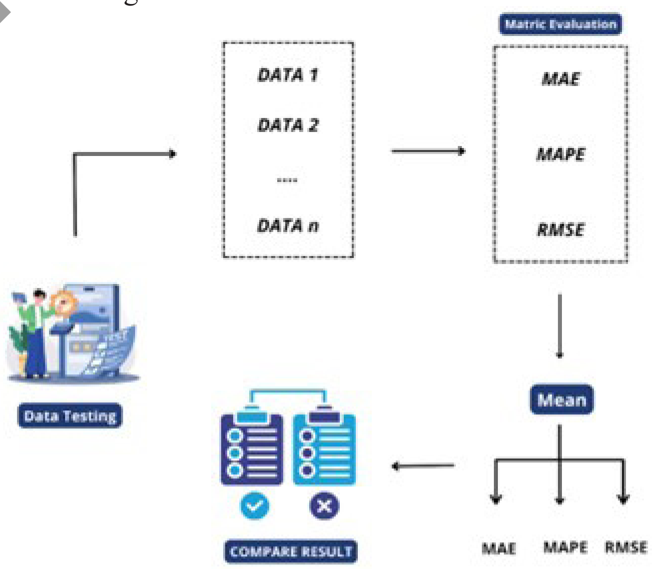


Figure 3 Evaluation Model Scenario

IN PRESS

III. RESULTS AND DISCUSSIONS

In this study, the first step is to cluster time-series price data for flour, medium-grade rice, and cooking oil. In accordance with previous research by Mardianto et al. (2023), the number of clusters is determined in advance, and this study evaluates configurations with two and three clusters. The similarity measures applied are Dynamic Time Warping (DTW) and Soft-DTW, and the data are standardized using the Time Series Scaler with mean–variance normalization. The optimal clustering result is selected based on the highest Silhouette score, and the Silhouette scores for the two- and three-cluster configurations are presented in the following section.

Table 1 shows positive overall Silhouette scores for each food commodity cluster. These results indicate that clustering using the K-Means algorithm effectively separates the data into distinct groups. Based on Table 1, clustering each food commodity price series yields two clusters, with the highest Silhouette values of 0.270 for flour, 0.386 for medium-grade rice, and 0.502 for cooking oil. Table 1 also indicates that Soft-DTW produces the highest Silhouette values for most commodities, except for cooking oil, which achieves its highest Silhouette value using DTW. The best time-series clustering visualizations for flour, medium-grade rice, and cooking oil prices are presented in Figures 4, 5, and 6, respectively.

Table 1 Silhouette Cluster Value

Food Commodity	**DTW**		**Soft-DTW**	
	2	**3**	**2**	**3**
Flour	0.166	0.182	**0.270**	0.188
Medium-grade Rice	0.261	0.133	**0.386**	0.247
Cooking Oil	**0.502**	0.170	0.370	0.342

IN PRESS

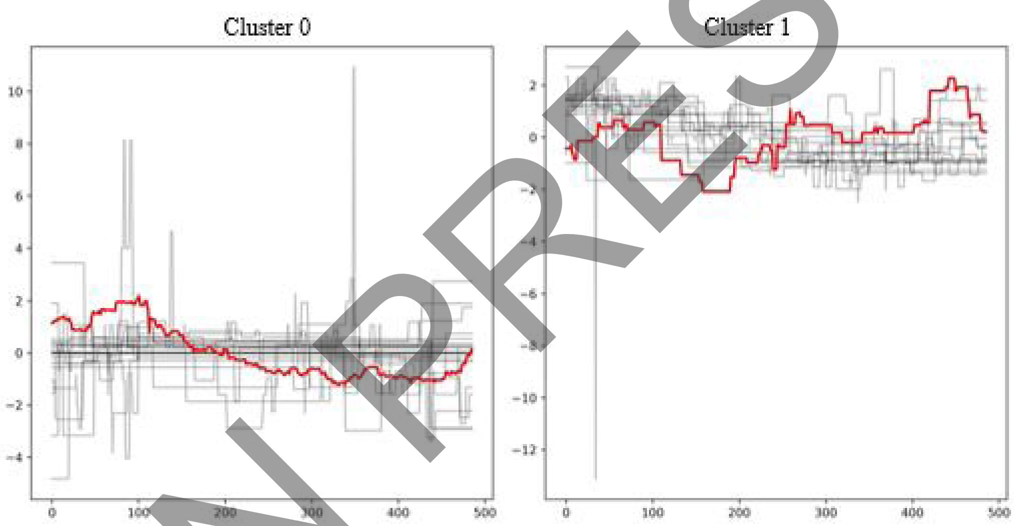


Figure 4 Time Series Clustering Visualization for Flour

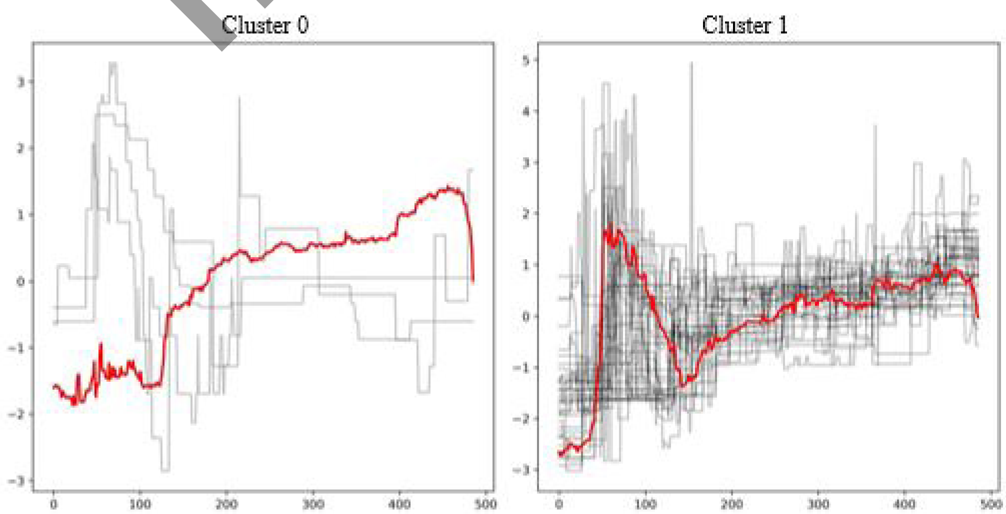


Figure 5 Time Series Clustering Visualization for Medium-Grade Rice

The results shown in Figures 4, 5, and 6 indicate clear differences in data patterns between cluster 0 and cluster 1. In these figures, the x-axis represents the sequence of observations, while the y-axis represents the normalized observation values. In Figure 4, the average time-series values for cluster 0 range from −4 to 10, whereas those for cluster 1 range from −12 to 2. Similarly, Figures 5 and 6 display distinct time-series centroid patterns for each cluster. These visualizations reveal hidden patterns in the time-series data for each cluster and indicate that commodity price data are suitable for clustering. Based on the clustering results, regions are grouped into clusters, where cluster 0 for flour, medium-grade rice, and cooking oil includes 22, 3, and 28 districts or cities, respectively, while cluster 1 includes 16, 35, and 9 districts or cities.

After each commodity cluster is formed, the next step is to calculate the average actual price of the regencies or cities within the same cluster. This aggregated average price data are then used in the forecasting process. Figures 7 and 8 present time-series plots of each commodity, showing the average prices for each cluster. The results indicate that the time-series pattern for cluster 0 fluctuates without a clear upward or downward trend. In contrast, flour prices in cluster 1 tend to decline over time, while medium-grade rice prices in cluster 1 show an increasing trend.

Unlike Figures 7 and 8, Figure 9 shows an increasing time-series pattern for both cluster 0 and cluster 1. In addition, Figures 7, 8, and and 9 show heterogeneous separation between clusters. These results indicate that the average prices differ significantly across clusters, suggesting that each cluster requires a different pricing policy. The next step is to conduct the forecasting process for each cluster and each food commodity price using stacking ensemble learning. Before modeling, lagged input variables are required because the data are univariate. For cluster 0, the significant lags for flour, medium-grade rice, and cooking oil prices are 2, 1, and 7,

IN PRESS

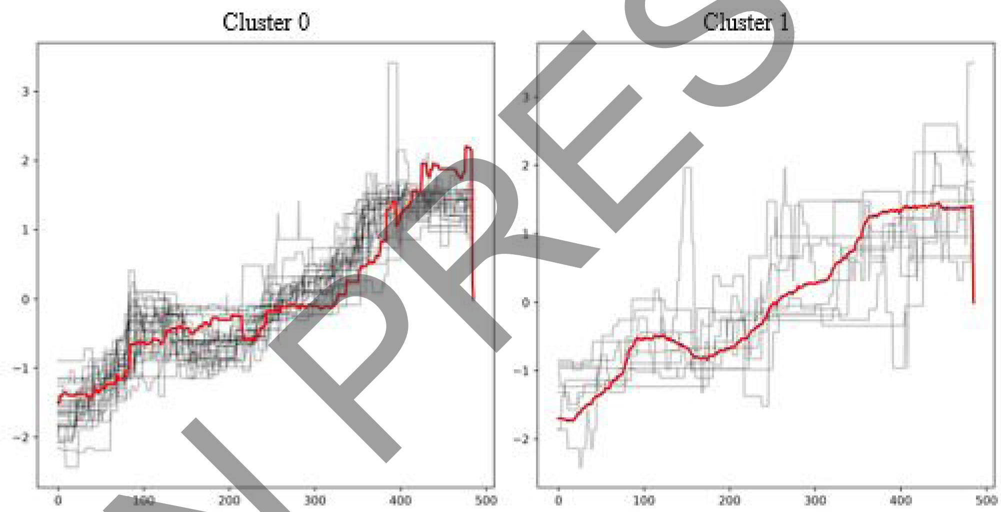


Figure 6 Time Series Clustering Visualization for Cooking Oil

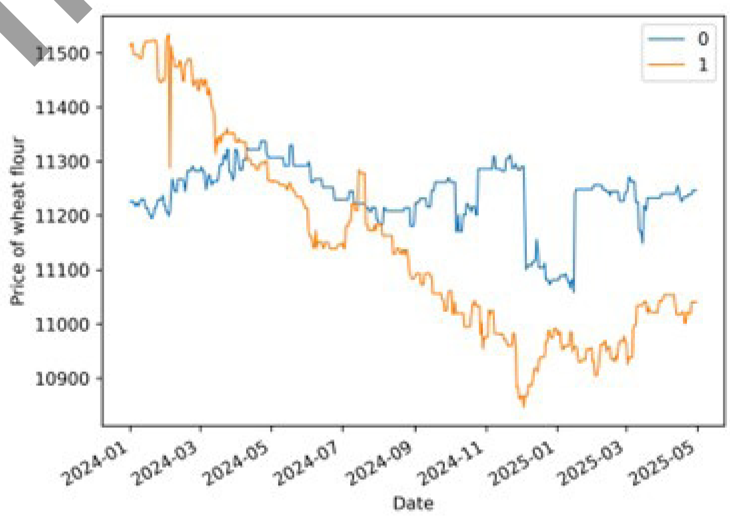


Figure 7 Plot Time Series of Average Wheat Flour Prices per Cluster

respectively. Meanwhile, for cluster 1, the significant lags are 1, 1, and 29. The data are then divided into training and testing sets, with the testing set consisting of the last seven days. The optimal parameters for the base learners and meta-learner for flour, medium-grade rice, and cooking oil price data are presented in Tables 2, 3, and 4.

When modeling the base learners and the meta-learner, optimal parameters are required to achieve satisfactory prediction accuracy. These optimal parameters for both the base learners and the meta-learner are identified using grid search. Using the optimal parameters obtained, the STACKEL K-MEANS model generates predictions on the test data, which are visualized in Figures 10, 11, and 12..

Figure 10 shows that, at a 95% confidence level, most flour price predictions for testing clusters 0 and 1 are close to the actual observed data. These results indicate that the STACKEL K-MEANS model effectively captures the underlying time-series patterns in flour prices. Similar modeling performance is observed in Figures 11 and 12 for medium-grade rice and cooking oil price data, where the predicted values closely follow the actual data.

From Figures 10, 11, and 12, differences in the widths of the confidence intervals are observed across food commodities. A wider confidence interval indicates greater variance between the actual data and the predicted values. Therefore, a model is considered more reliable when the confidence interval is narrower, as shown in Figures 10 and 12. The model evaluation in this study compares the performance of the STACKEL K-MEANS method with AutoARIMA modeling applied to all districts or cities in East Java. This evaluation is conducted by comparing the modeling results on both training and testing data for each food commodity. The AutoARIMA evaluation is based on the average RMSE, MAE, and MAPE values

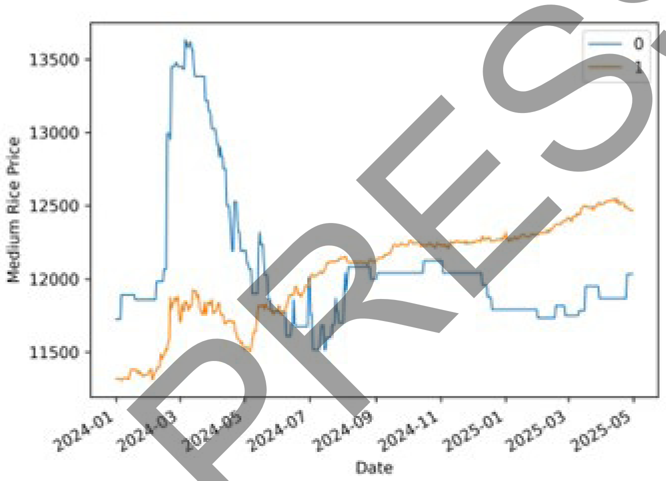


Figure 8 Plot Time Series of Average Medium Rice Prices per Cluster

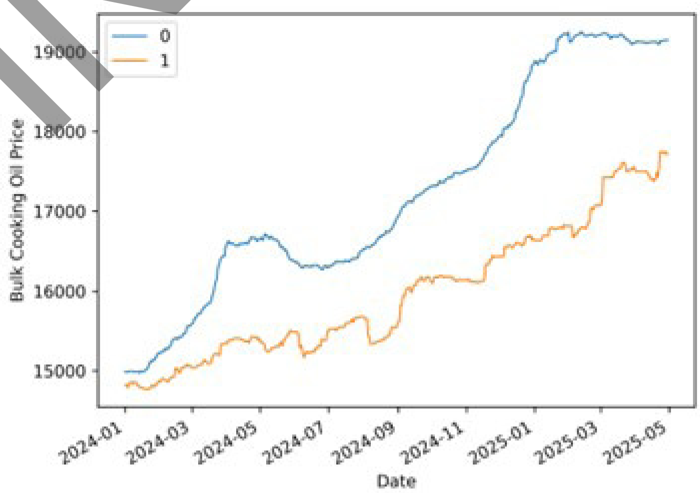


Figure 9 Plot Time Series of Average Bulk Cooking Oil Prices per Cluster

IN PRESS

Table 2 Best Search Grid Parameters for Flour Price Data

Base and Meta Learner	Cluster 0	Cluster 1
Ridge Regression	'alpha': 0.01	'alpha': 1e-15
Random Forest	'max_depth': 5, 'min_samples_split': 3, 'n_estimators': 100	'max_depth': 10, 'min_samples_split': 10, 'n_estimators': 100
SVR	'C': 100, 'epsilon': 0.0001, 'gamma': 'auto', 'kernel': 'rbf'	'C': 1, 'epsilon': 0.0001, 'gamma': 'auto', 'kernel': 'rbf'
LGBM	'learning_rate': 0.5, 'max_depth': 7, 'n_estimators': 100, 'num_leaves': 10	'learning_rate': 0.1, 'max_depth': 7, 'n_estimators': 100, 'num_leaves': 31

Table 3 Best Search Grid Parameters for Medium-Grade Rice Price Data

Base and Meta Learner	Cluster 0	Cluster 1
Ridge Regression	'alpha': 1e-15	'alpha': 1e-15
Random Forest	'max_depth': 5, 'min_samples_split': 5, 'n_estimators': 100	'max_depth': 5, 'min_samples_split': 2, 'n_estimators': 100
SVR	C': 10, 'epsilon': 0.0001, 'gamma': 'scale', 'kernel': 'rbf'	'C': 1000, 'epsilon': 0.0001, 'gamma': 'auto', 'kernel': 'rbf'
LGBM	'learning_rate': 0.5, 'max_depth': 7, 'n_estimators': 10, 'num_leaves': 10	''learning_rate': 0.5, 'max_depth': -1, 'n_estimators': 100, 'num_leaves': 15

Table 4 Best Search Grid Parameters for Cooking Oil Price Data

		Auto ARIMA	STACKEL K-MEANS
Data training	RMSE	1412.930	26.080
	MAE	1411.567	13.925
	MAPE (%)	13.264	0.124
Data testing	RMSE	523.997	6.345
	MAE	37.576	4.715
	MAPE (%)	0.328	**0.042**

IN PRESS

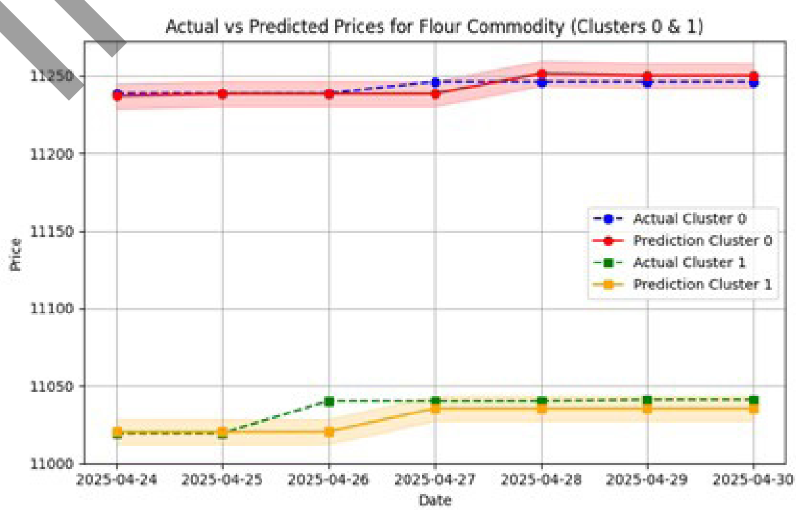


Figure 10 Visualization of Stacking Ensemble Learning Prediction Results on Testing Data for Flour Prices

across 37 to 38 district or city models in East Java. In contrast, the STACKEL K-MEANS evaluation uses the average RMSE, MAE, and MAPE values from the models obtained for clusters 0 and 1. This comparison shows that STACKEL K-MEANS is computationally more efficient than AutoARIMA, as it removes the need to average results from numerous individual regional models.

The evaluation results are presented in Tables 5, 6, and 7. These tables show that the RMSE, MAE, and MAPE values on the testing data for both AutoARIMA and STACKEL K-MEANS are lower than those on the training data. This pattern indicates that the models generalize well and produce reliable predictions. Furthermore, Tables 5, 6, and 7 demonstrate that the RMSE and MAE values on the testing data obtained using the STACKEL K-MEANS method are consistently lower than those produced by AutoARIMA. These results confirm that STACKEL K-MEANS outperforms AutoARIMA in forecasting the prices of all three food commodities.

Tables 5, 6, and 7 show that the MAPE values of the AutoARIMA model for flour, medium-grade rice, and cooking oil prices are 0.392%, 0.392%, and 0.250%, respectively. These MAPE values are all below 10%, indicating that the AutoARIMA model achieves high accuracy in predicting food commodity prices. However, when compared with the STACKEL K-MEANS model, the MAPE values are substantially lower at 0.042%, 0.261%, and 0.185% for flour, medium-grade rice, and cooking oil, respectively. These results indicate that the STACKEL K-MEANS

IN PRESS

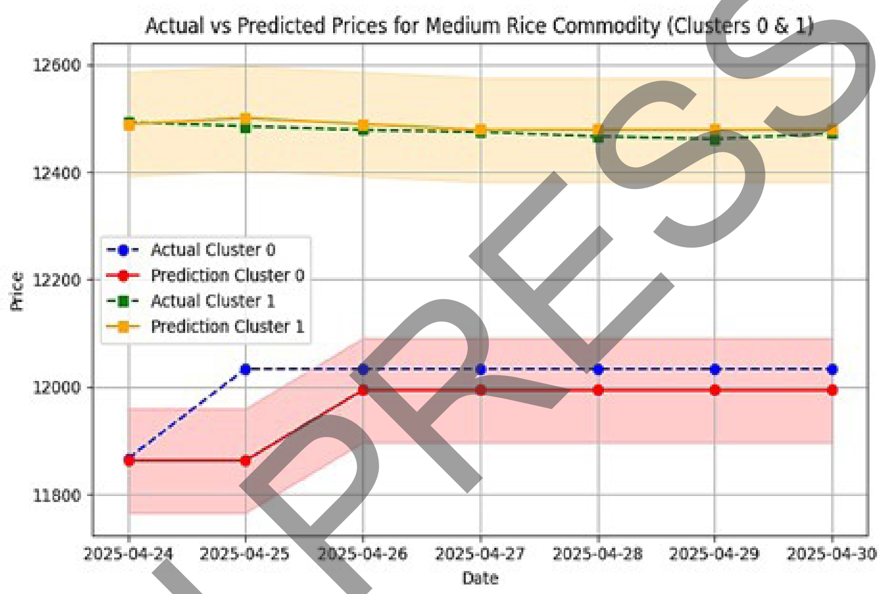


Figure 11 Visualization of Stacking Ensemble Learning Prediction Results on Testing Data for Medium-Grade Rice

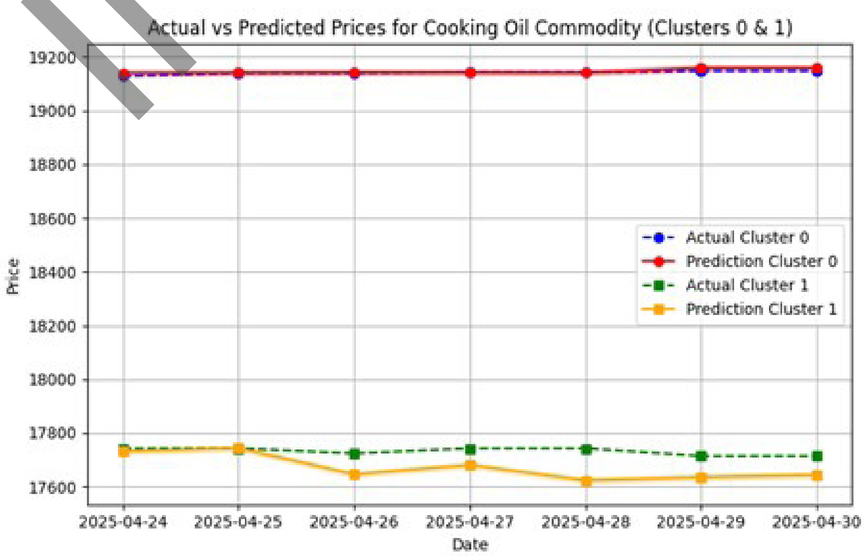


Figure 12 Visualization of Stacking Ensemble Learning Prediction Results on Testing Data for Cooking Oil

Table 5 Model Evaluation on Flour Price Data

		Auto ARIMA	**STACKEL K-MEANS**
Data training	RMSE	1412.930	26.080
	MAE	1411.567	13.925
	MAPE (%)	13.264	0.124
Data testing	RMSE	523.997	6.345
	MAE	37.576	4.715
	MAPE (%)	0.328	**0.042**

Table 6 Model Evaluation on Medium-Grade Rice Data

		AutoARIMA	**STACKEL K-MEANS**
Data training	RMSE	506.133	53.375
	MAE	68.674	27.520
	MAPE (%)	0.598	0.226
Data testing	RMSE	56.068	41.920
	MAE	48.142	31.585
	MAPE (%)	0.392	**0.261**

Table 7 Model Evaluation on Oil Cooking Data

		AutoARIMA	**STACKEL K-MEANS**
Data training	RMSE	684.972	139.150
	MAE	76.912	103.665
	MAPE (%)	0.476	0.651
Data testing	RMSE	56.060	39.345
	MAE	47.571	33.020
	MAPE (%)	0.250	**0.185**

IN PRESS

method provides more accurate price predictions for all three food commodities than the AutoARIMA model. Accordingly, the proposed STACKEL K-MEANS approach demonstrates superior modeling performance compared with the conventional AutoARIMA model.

IV. CONCLUSIONS

Cluster analysis yields two clusters for each commodity in this study. The distance measures used indicate that Soft-DTW is more frequently selected than standard DTW based on clustering performance. In addition, K-Means serves as the primary clustering method and produces robust inter- and intra-cluster separation, as reflected by the Silhouette scores. Based on the clustering results, new datasets are generated by aggregating the sample data within each cluster, resulting in two aggregated datasets per commodity. Stacking ensemble learning is then applied to these aggregated data, producing at least six predictive models to represent price variations across 37–38 districts or cities for the three commodities.

Subsequently, the prediction results obtained from the STACKEL K-MEANS method are evaluated for each city and each commodity. The evaluation scores are then averaged to produce a single performance score for each metric and commodity. The results show that the proposed method consistently outperforms the benchmark AutoARIMA model across all evaluation metrics and commodities, on both training and testing data. Although the performance exceeds that of AutoARIMA, further evaluation and development are still required, such as testing additional evaluation metrics and using alternative datasets. Overall, the proposed framework shows strong potential as a forecasting method.

This study has several limitations that should be considered when interpreting the results. First, the clustering process yields only two clusters for each commodity, which may oversimplify the complexity of real market structures. Second, the evaluation phase relies on selected performance metrics and the available dataset, which may limit the generalizability of the findings to different time periods, regions, or commodities with distinct price dynamics. Third, the specific combination of distance measures, clustering techniques, aggregation strategies, and stacking configurations may introduce model sensitivity.

Therefore, future research should explore alternative evaluation metrics, test different parameter settings during the clustering stage, and apply the framework to additional datasets to enhance external validity. Moreover, expanding the stacking ensemble by incorporating more diverse base learners and relevant external variables could further improve robustness and practical applicability in real-world forecasting scenarios.

AUTHOR CONTRIBUTIONS

Conceived and designed the analysis, A. M.; Collected the data, N. S. and S. A. P.; Performed the analysis, A. T. D.; Wrote the paper, A. T. D. and A. M.; Other contribution, N. S., S. A. P. and M. N.

DATA AVAILABILITY

The data supporting the research findings are available at https://siskaperbapo.jatimprov.go.id/harga-komoditas. These data were derived from the following resources available in the public domain: https://siskaperbapo.jatimprov.go.id/harga-komoditas.

REFERENCES

Amatullah, F. F., Ilmani, E. A., Fitrianto, A., Erfiani, & Jumansyah, L. M. R. D. (2025). Clustering time series forecasting model for grouping provinces in Indonesia based on granulated sugar prices. *Journal of Applied Informatics and Computing (JAIC)*, *9*(1), 121–130. https://doi.org/10.30871/jaic.v9i1.8840

Cuturi, M., & Blondel, M. (2017). Soft-DTW: A differentiable loss function for time-series. *Proceedings of the 34th International Conference on Machine Learning (ICML 2017)*, 894–903. https://doi.org/10.48550/arXiv.1703.01541

Farisi, O. I. R., Jannah, N., & Insania, R. (2022). Prediksi harga komoditas pangan di Indonesia menggunakan backpropagation. *COREAI Jurnal Kecerdasan Buatan, Komputasi dan Teknologi Informasi*, *3*(1), 91–101. https://doi.org/10.33650/coreai.v3i1.4282

Folgado, D., Barandas, M., Matias, R., Martins, R., Carvalho, M., & Gamboa, H. (2018). Time alignment measurement for time series. *Pattern Recognition*, *81*, 268–279. https://doi.org/10.1016/j.patcog.2018.04.003

Hasibuan, L. S., & Novialdi, Y. (2022). Prediksi harga minyak goreng curah dan kemasan menggunakan algoritme Long Short-Term Memory (LSTM). *JIKA: Jurnal Ilmu Komputer dan Aplikasinya*, *9*(2), 149–157. https://doi.org/10.29244/jika.9.2.149-157

Hassani, H., Marvian, L., Yarmohammadi, M., & Yeganegi, M. R. (2024). Unraveling time series dynamics: Evaluating partial autocorrelation function distribution and its implications. *Mathematical and Computational Applications*, *29*(4), 58. https://doi.org/10.3390/mca29040058

Hegg, J. C., & Kennedy, B. P. (2021). Let's do the time warp again: Non-linear time series matching as a tool for sequentially structured data in ecology. *Ecosphere*, *12*(9), e03742. https://doi.org/10.1002/ecs2.3742

Herrmann, M., Tan, C. W., & Webb, G. I. (2023). Parameterizing the cost function of dynamic time warping with application to time series classification. *Data Mining and Knowledge Discovery*, *37*(5), 2024–2045. https://doi.org/10.1007/s10618-023-00926-8

Hodson, T. O. (2022). Root-Mean-Square Error (RMSE) or Mean Absolute Error (MAE): When to use them or not. *Geoscientific Model Development*, *15*(14), 5481–5487. https://doi.org/10.5194/gmd-15-5481-2022

Ismail, W. N., Alsalamah, H. A., & Mohamed, E. (2023). GA-stacking: A new stacking-based ensemble learning method to forecast the COVID-19 outbreak. *Computers, Materials and Continua*, *74*(2), 3945–3976. https://doi.org/10.32604/cmc.2023.031194

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). Moving beyond linearity. In G. James, D. Witten, T. Hastie, & R. Tibshirani (Eds.), *An introduction to statistical learning: With applications in R* (pp. 265–301). Springer. https://doi.org/10.1007/978-1-4614-7138-7_7

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. In I. Guyon, U. von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. V. N. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* (Vol. 30). Curran Associates, Inc. https://api.semanticscholar.org/CorpusID:3815895

Khan, W., Walker, S., & Zeiler, W. (2022). Improved solar photovoltaic energy generation forecast using deep learning-based ensemble stacking approach. *Energy*, *240*, 122812. https://doi.org/10.1016/j.energy.2021.122812

Kumar, S., Pant, M., & Nagar, A. (2024). Forecasting the sugarcane yields based on meteorological data through ensemble learning. *IEEE Access*, *12*, 176539–176553. https://doi.org/10.1109/ACCESS.2024.3502547

Kwon, H., Park, J., & Lee, Y. (2019). Stacking ensemble technique for classifying breast cancer. *Healthcare Informatics Research*, *25*(4), 283–288. https://doi.org/10.4258/hir.2019.25.4.283

Li, Q., Zhang, X., Ma, T., Liu, D., Wang, H., & Hu, W. (2022). A multi-step ahead photovoltaic power forecasting model based on TimeGAN, soft DTW-based K-medoids clustering, and a CNN-GRU hybrid neural network. *Energy Reports*, *8*, 10346–10362. https://doi.org/10.1016/j.egyr.2022.08.180

Lukman, A. F., & Olatunji, A. (2018). Newly proposed estimator for ridge parameter: An application to the Nigerian economy. *Pakistan Journal of Statistics*, *34*(2), 91–98. https://api.semanticscholar.org/CorpusID:172134763

IN PRESS

Mandal, U., Chakraborty, A., Mahato, P., & Das, G. (2023). LinVec: A stacked ensemble machine learning architecture for analysis and forecasting of time-series data. *Indian Journal of Science and Technology*, *16*(8), 570–582. https://doi.org/10.17485/IJST/v16i8.2197

Mardianto, M. F. F., Suliyanto, Effendy, F., Cahyasari, A. D., Purwoko, C. F. F., Aliffia, N., & Simamora, A. N. M. B. (2023, August). Mapping regencies and cities in East Java related food potential using the K-means method. The 8th International Conference and Workshop on Basic and Applied Science *(ICOWOBAS) 2021*. https://doi.org/10.1063/5.0103807

Maulidya, A., Sitorus, Z., Siahaan, A. P. U., & Iqbal, M. (2024). Analysis of increasing student service satisfaction using K-Means clustering algorithm and Gaussian Mixture Models (GMM). *International Journal of Computer Sciences and Mathematics Engineering*, *1*(1), 29–35. https://doi.org/10.61306/ijecom.v3i1.62

Muhaimin, A., Prastyo, D. D., & Lu, H.-S. H. (2021). Forecasting with recurrent neural network in intermittent demand data. *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, 802–809. https://doi.org/10.1109/Confluence51648.2021.9376880

Pavlou, M., Omar, R. Z., & Ambler, G. (2024). Penalized regression methods with modified cross-validation and bootstrap tuning produce better prediction models. *Biometrical Journal*, *66*(5), e202300245. https://doi.org/10.1002/bimj.202300245

Putri Z, R. W., Al Maududi, R., & Hartuti, P. M. (2024). Peramalan harga bahan pangan menggunakan fuzzy time series. *Journal of Science and Technology*, *4*(2), 177–188. https://doi.org/10.15548/jostech.v4i2.9728

Renju, K., & Brunda, V. (2024). Optimizing crop yield prediction through multiple models: An ensemble stacking approach. *International Journal of Data Informatics and Intelligent Computing*, *3*(2), 52–58. https://doi.org/10.59461/ijdiic.v3i2.120

Sardá-Espinosa, A. (2019). Time-series clustering in R using the dtwclust package. *The R Journal*, *11*(1), 22–43. https://doi.org/10.32614/RJ-2019-023

Shahapure, K. R., & Nicholas, C. (2020). Cluster quality analysis using silhouette score. *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*, 747–748. https://doi.org/10.1109/DSAA49011.2020.00096

Šťastný, T., Koudelka, J., Bílková, D., & Marek, L. (2022). Clustering and modelling of the top 30 cryptocurrency prices using dynamic time warping and machine learning methods. *Mathematics*, *10*(19), 3672. https://doi.org/10.3390/math10193672

Suresh, K., Meghana, J., & Pooja, M. E. (2021). Predicting the e-learners learning style by using support vector regression technique. *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, 350–355. https://doi.org/10.1109/ICAIS50930.2021.9396018

Swaraj, A., Verma, K., Kaur, A., Singh, G., Kumar, A., & Melo de Sales, L. (2021). Implementation of stacking based ARIMA model for prediction of COVID-19 cases in India. *Journal of Biomedical Informatics*, *121*. https://doi.org/10.1016/j.jbi.2021.103887

Talekar, B. (2020). A detailed review on decision tree and random forest. *Bioscience Biotechnology Research Communications*, *13*(14), 245–248. https://doi.org/10.21786/bbrc/13.14/57

Wang, J., Wang, Z., Li, X., & Zhou, H. (2022). Artificial bee colony-based combination approach to forecasting agricultural commodity prices. *International Journal of Forecasting*, *38*(1), 21–34. https://doi.org/10.1016/j.ijforecast.2019.08.006

Xu, M., Garg, S., Milford, M., & Gould, S. (2023). Deep declarative dynamic time warping for end-to-end learning of alignment paths. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Yohansa, M., Notodiputro, K. A., & Erfiani. (2022). Dynamic time warping techniques for time series clustering of COVID-19 cases in DKI Jakarta. *ComTech: Computer, Mathematics and Engineering Applications*, *13*(2), 63–73. https://doi.org/10.21512/comtech.v13i2.7413

Zen, M. A., Wahyuningsih, S., & Dani, A. T. R. (2022). Aplikasi pendekatan agglomerative hierarchical time series clustering untuk peramalan data harga minyak goreng di Indonesia. *Seminar Nasional Official Statistics*, 293–302. https://doi.org/10.34123/semnasoffstat.v2022i1.1394

IN PRESS