Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 7, No. 3, June 2026, Page. 2092-2113
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.3.5517

Hybrid Unsupervised-Supervised Learning for Housing Submarket Segmentation and Price Prediction in Surabaya Urban Areas

Rinabi Tanamal[1], Satria Adi Nugraha*[2], Nathalia Minoque Kusuma Salma Rasyid Jr[3], Livanty Efatania Dendy[4], Jessica Theijer[5]

[1,2,3,4,5]Information Systems, Universitas Ciputra, Indonesia

Email: [2]satria.nugraha@ciputra.ac.id

Received: Dec 3, 2025; Revised: Dec 16, 2025; Accepted: Dec 21, 2025; Published: Jun 15, 2026

Abstract

Surabaya's rapid population growth, reaching 3.02 million residents, has intensified housing affordability challenges and increased structural variability in residential markets. This study proposes a hybrid machine learning framework that combines unsupervised clustering with supervised classification to identify submarket segments and predict housing price categories. A dataset of 490 properties containing structural, land, ownership, and contextual features was preprocessed and analyzed using K-Means. Cluster quality assessment through elbow inspection and a silhouette score of 0.45 indicated the presence of five meaningful market segments. These segments served as targets for a supervised classification stage that evaluated seven models, optimized via randomized hyperparameter search within a standardized preprocessing pipeline.

The RBF-SVM achieved the strongest performance, reaching 97 percent accuracy and a macro-F1 score of 0.97, representing an 8 percent improvement over non-hybrid baselines and outperforming boosted ensembles such as XGBoost. Permutation importance analysis identified number of floors, building orientation, position rank, and ownership status as dominant drivers of segment differentiation. The integration of clustering and classification enhances predictive reliability while improving interpretability, offering a transparent analytical toolkit for housing market assessment.

The proposed framework provides actionable insights for developers, appraisers, and policymakers in Surabaya, enabling data-driven identification of submarkets and supporting more equitable housing strategies aligned with SDG 11 on sustainable urban development. The approach is scalable to other Indonesian cities and establishes a foundation for future work incorporating spatial, socioeconomic, or temporal predictors.

Keywords: *Clustering Algorithms, Housing Price Prediction, Hybrid Machine Learning, Submarket Segmentation, Surabaya Real Estate*

This work is an open access article and licensed under a Creative Commons Attribution-Non-Commercial 4.0 International License

1. INTRODUCTION

Indonesia's population continues to rise each passing year. Looking at projections from the Central Statistics Agency (BPS), during the 2020–2025 period, the country's population grows by approximately 1.09% per year [1]. This growth is not evenly distributed across all regions but rather concentrated in major urban areas. This condition is also evident in Surabaya, the capital city of East Java Province, which ranks as the second most densely populated city after Jakarta [2]. Having stood for more than 700 years, Surabaya possesses a complex spatial and regional structure. According to data from BPS Surabaya, the city's population has reached around 3.02 million people, with a density of more than 9,240 inhabitants per km², spread across 31 districts and 154 sub-districts within an area of 326.81 km² [3]. With such high population density, the city's demographic growth is inevitable and continues to be influenced by birth rates, mortality, and migration flows, both inbound and outbound [4]. To strengthen this demographic overview, BPS projections indicate that Surabaya's population will continue to increase through 2025 and beyond. As shown in Table 1, the projected population is expected

Vol. 7, No. 3, June 2026, Page. 2092-2113
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.3.5517

to rise from 2.99 million in 2023 to over 3.04 million in 2025, with the city maintaining a steady upward trajectory through 2032 despite declining annual growth rates. These projections, supported by the 2025 BPS population distribution infographic, highlight that urban densification in Surabaya will persist in the coming decade. This sustained demographic pressure reinforces the urgency of assessing future housing demand and price dynamics in Surabaya's rapidly growing metropolitan environment [5].

Table 1. Population Size and Growth Rate of Surabaya City, 2023–2032

Year	Total Population (in person)	Annual Population Growth Rate (%)
2023	2.997.547	0,89
2024	3.021.043	0,83
2025	3.043.518	0,79
2026	3.065.133	0,75
2027	3.085.996	0,71
2028	3.106.108	0,67
2029	3.125.548	0,64
2030	3.144.330	0,61
2031	3.162.400	0,58
2032	3.179.667	0,54

The rapid growth of the population has a direct impact on the increasing demand for basic facilities, particularly housing. At present, housing has a dual function, not only serving as a primary need that provides shelter but also as an investment instrument. In the context of the modern market, a house functions both as a dwelling and as an asset whose value tends to increase along with rising demand [6], [7]. The investment decision-making process is complex and multi-faceted, influenced by economic factors. Additionally, macroeconomic variables like interest rates and inflation affect real estate investments [8], [9]. This dynamic makes housing availability an important issue, as the supply of houses is not proportional to the demand, creating significant pressure on the availability of adequate and affordable housing [10], [11], [12], [13]. Various factors influence house prices, such as location, building quality, accessibility, infrastructure, government policy, and macroeconomic conditions. High demand in urban areas with limited land availability causes house prices to rise every year, making it increasingly difficult for low- and middle-income households to own decent housing [14], [15]. This condition affects social welfare and has the potential to create social and economic inequality. Furthermore, the expansion of residential areas is often not accompanied by sufficient supporting infrastructure and facilities, which reduces the quality of the living environment.

In this context, machine learning has strong potential for analyzing the housing market since it makes it possible to process vast amounts of property data efficiently and find patterns that traditional approaches frequently miss. Its efficacy in examining price trends and market segmentation using clustering approaches has been shown in earlier research. Using physical and spatial features, Skovajsa demonstrated that K-Means and Hierarchical Clustering generated stable price-based groupings [16]. For big and uneven datasets, K-Means outperformed density-based techniques like DBSCAN. Similarly, Septiani et al. discovered that clustering efficiently reveals pricing distributions and facilitates strategic

Vol. 7, No. 3, June 2026, Page. 2092-2113
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.3.5517

decision-making in housing development and marketing when they applied K-Means to Indonesian housing data [17].

Several studies have examined supervised learning methods for housing price prediction and classification. Comparative analyses show that ensemble-based models such as Random Forest and XGBoost generally achieve strong performance across different datasets [18], [19], [20], while simpler models such as Naïve Bayes and logistic regression can remain competitive under certain data assumptions [21]. These findings suggest that model effectiveness depends on data characteristics rather than algorithm complexity alone.

Additionally, Alamri et al. proposed a hybrid property price classification framework that integrates clustering and supervised learning, categorizing properties as undervalued, fair-valued, or overvalued [22]. Their results showed that the combination of Fuzzy C-Means and Random Forest outperformed single-model approaches [23]. Similar hybrid strategies have also been applied in the Indonesian context, where incorporating a clustering stage prior to classification significantly improved predictive performance, as demonstrated in a K-Means–Decision Tree model for poverty status classification [24].

The effectiveness of segmentation-based and multi-model machine learning techniques is further supported by more recent research in the housing sector. In order to predict housing prices, Soegianto et al. compared Linear Regression, Artificial Neural Networks (ANN), Random Forest Regressor, and Support Vector Regression (SVR) [25]. They discovered that ANN had the best predictive accuracy, followed by SVR and Random Forest, highlighting the significance of modeling non-linear relationships in housing data. Chiu demonstrated that deep learning models can better capture temporal dependencies and market volatility than conventional regression-based methods by using a Long Short-Term Memory (LSTM) model in a time-series context to forecast housing prices in Taiwan during the post-epidemic period [26]. Additionally, before making a prediction, Gümmer et al. suggested a two-stage clustering approach that divides housing markets according to property attributes and geography [27]. Their findings demonstrated significant gains in performance without sacrificing interpretability, underscoring the importance of clustering-based segmentation in diverse housing markets.

Table 2. Summary of Selected Studies on Housing Price Prediction

Study	Region / Dataset	Method	Best Performance	Limitation
Jha et al. [18]	USA	XGBoost	Acc. ≈95–96%	Binary classification (market-level)
Tanamal et al. [23]	Surabaya	Random Forest	Acc. 75.1%	Single-model
Almari et al. [22]	Saudi Arabia	FCM + RF	Acc. ≈98–99%	Regional focus
Soegianto et al. [25]	Housing benchmark	ANN	Highest accuracy among compared models	No segmentation
Gümmer et al. [27]	Germany	Two-stage clustering + LR	MAE ↓ up to 58%	Not tested in Indonesia

Table 2 summarizes five representative studies in housing price analysis and prediction, highlighting their methodologies, reported performance, and limitations. The comparison shows that while ensemble and deep learning models often achieve strong predictive results, many studies rely on global models that do not explicitly account for market heterogeneity. Recent clustering-based approaches demonstrate improved performance and interpretability by capturing localized pricing

patterns across different market segments, providing a strong rationale for the two-stage clustering and classification framework adopted in this study.

Overall, previous research indicates that supervised learning particularly ensemble and deep learning models, achieves superior predictive accuracy, whereas clustering is useful for initial housing market segmentation. In order to anticipate housing price categories, this study uses a two-stage framework in which housing data is first segregated using several clustering techniques and then classified using seven supervised algorithms. By analyzing Surabaya City's housing price dynamics, this method seeks to determine which models perform best in terms of accuracy, precision, recall, and F1-score.

2. METHOD

The objective of this study is to develop a hybrid predictive model for the housing data of Surabaya utilizing supervised and unsupervised learning. The workflow illustrated in Figure 1, which includes the main phases of this study, is followed by the research methodology, which includes data collecting, preprocessing, feature scaling, clustering, classification, and result analysis.

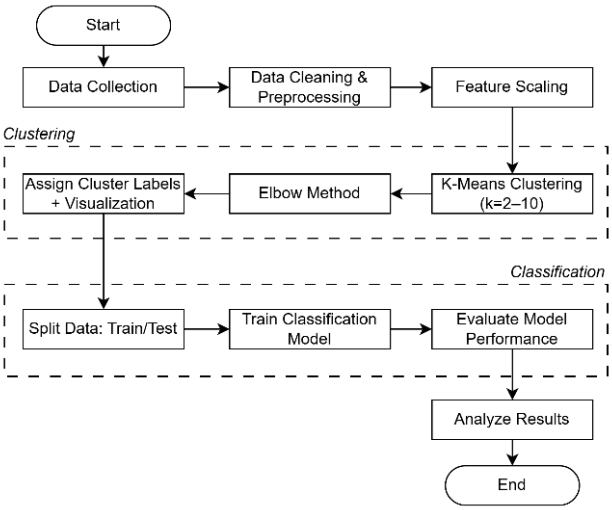

Figure 1. Research Flowchart

2.1. Data Collection

We gathered the research dataset for this study from multiple real estate agents in Surabaya, Indonesia. The collection has a total of 490 raw residential property details, which were all house listings. It provides properties for both category and number attributes that explain the features' qualities. Some of the most important features include Physical attributes: Surface area (m²), Building area (m²), Number of bedrooms, Number of bathrooms, Number of storeys, Legal attributes: Ownership status (SHM, HGB, PPJB), Market-related attributes: Community price and public facilities. Thereafter, all of the data was put into an organized CSV file.

2.2. Data Cleaning and Preprocessing

The dataset taken was cleaned and preprocessed so that accuracy before analysis can increase. For example, irrelevant columns inside the dataset were removed since there are some attributes that are not required for the clustering and classification input. We used the most common value mode to fill in missing values in several columns so that the overall data distribution would be balanced. To figure out more about the data and identify any issues, exploratory data analysis was then conducted. Boxplots and other visual tools were made to observe for extreme values that could distort the clustering results.

Vol. 7, No. 3, June 2026, Page. 2092-2113
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.3.5517

After these steps, the final dataset has been structured to make all of its features, like data types and value ranges, consistent. The main structure of the final dataset is summarized in Table 3, which lists the variables applied in the clustering and classification process, along with their descriptions, measurement units, and types.

Table 3. Description of the final structured dataset after preprocessing

No	Attributes			
	Name	Description	Unit Explanation	Type
1	Rank Area Category	Category of the property area value	1 = Below Standard 2 = Standard 3 = Premium 4 = Very Premium	Categorical / Ordinal
2	Surface Area	Total size of the land plot	Square meters (m^2)	Continuous
3	Building Area	Total size of the constructed building	Square meters (m^2)	Continuous
4	Bedrooms	Total number of bedrooms in the property	0-36 units	Continuous
5	Bathrooms	Total number of bathrooms in the property	1-36 units	Continuous
6	Storey	Number of building floors	1-4 floors	Continuous
7	Community Price	Average price offered by the property-agent community	Indonesian Rupiah (IDR)	Continuous
8	Ownership Status	Legal ownership status of the property	1 = Green Certificate (Surat Hijau) 2 = Binding Sale and Purchase Agreement (PPJB) 3 = Freehold (SHM) 4 = Right to Build (HGB)	Categorical / Ordinal
9	House Facing	Facing the direction of the building	1 = West 2 = North 3 = South 4 = East	Categorical / Ordinal
10	House Position	Position of the house within the residential complex	1 = Skewers (Tusuk Sate) 2 = Cul-de-sac / End of a Dead-End Alley 3 = Back Pocket 4 = Corner / Hook 5 = Standard	Categorical / Ordinal

Vol. 7, No. 3, June 2026, Page. 2092-2113
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.3.5517

11	Road Width	Width of the road in front of the property according to the numbers of cars that can pass	1 = < 1 car 2 = 1-2 cars 3 = > 2 cars	Categorical / Ordinal
12	Building Age	Age of the building since construction	1 = 1-4 years 2 = 5-10 years 3 = >10 years	Categorical / Ordinal
13	Ready to Occupy	Indicates whether the house is ready to be used	0 = No 1 = Yes	Categorical / Ordinal
14	Furnished	Property filled with interior furniture	0 = No 1 = Yes	Categorical / Ordinal
15	Public Facilities	Availability of public facilities such as markets, schools, healthcare, malls, or main roads	1 = 1-3 facilities 2= 4-6 facilities 3= 7-9 facilities	Categorical / Ordinal

2.3. Feature Scaling

Feature scaling was done to improve clustering speed and make the size of numerical attributes more consistent. K-Means uses distance-based computations [28], therefore, if the data isn't scaled, features with greater numeric ranges, including community price or land area, could take over the findings. So, the StandardScaler function from the Scikit-learn library was used for some variables, such as 'Surface Area', 'Building Area', 'Bedrooms', 'Bathrooms', 'Storey', 'Community Prices', and 'Building Age'.

The scaling process transformed each value x into standardized form using the formula:

$$x' = \frac{x - \mu}{\sigma} \quad (1)$$

In this context, "Here" x' represents the new standardized value after the transformation. Meanwhile, x is value of initial feature, μ is the mean, and σ for standard deviation. In fact, the conversion was done by using the StandardScaler() method to fit and adjust the chosen columns, and the results were then stored in a new DataFrame called df_scaled. Then, these standardized values were updated to the original dataset. The K-Means algorithm could then evaluate the relationships between data more accurately during the formation of clusters.

2.4. Clustering using K-Means Method

To categorize the data according to its similarity, the clustering process was carried out using K-Means, a machine learning technique that does not require human supervision, called unsupervised classification [29]. By minimizing the Sum of Squared Errors (SSE), also called the Within-Cluster Sum of Squares (WCSS) and representing the total variance within each cluster, the technique divides the data into k unique, non-overlapping clusters. The objective function is expressed as:

$$SSE = \Sigma_{i=1}^{k} \Sigma_{x_j \epsilon C_i} ||x_j - \mu_i||^2 \quad (2)$$

where C_i is the group of data points that belong to the i-th cluster. x_j is one data point, and μ_i is the center of cluster i. The goal of the algorithm is to make SSE as minimal as possible so that data points are as close as likely towards the center of their cluster.

In this study, the number of clusters (k) was empirically determined by evaluating values ranging from 2 to 10. A K-means model was trained on the standardized dataset for each iteration to find the WCSS value that went with it. The Elbow Method was used to visualize the results [30]. This

technique finds the point at which increasing the number of clusters no longer considerably lowers WCSS. Then, to help find this point objectively, the KneeLocator function from the kneed library was used.

After finding a possible range for k, the GridSearchCV method was used to tune the parameters and find the best values for several hyperparameters, such as the method of initialization, the maximum number of iterations, and the random seed. The parameters that underwent testing were the init values of "k-means++" and "random", the max_iter values of 100 to 400, and several random states.

The clustering model was trained on the whole standardized dataset. Each property got a cluster label, which was saved in the main DataFrame as a new column called KMeans after optimizing and testing the parameters. Plotly Express was also used to make 2D and 3D scatter plots visualizations that demonstrate the clustering results. This technique made sure that the clustering procedure was done in a systematic way so that the cluster information was ready to be used in the classification stage.

2.5. Classification Stage

In supervised machine learning, classification is an important task that involves predicting which group or market segment a property belongs to [31]. In mathematical terms, the classifier learns a function:

$$f : R^d \rightarrow \{1,2,\ldots,C\} \quad (5)$$

where $x \epsilon R^d$ is a vector of d numerical variables that describe one property, such as surface area, number of bedrooms, or community price, and C is the entire number of potential classes. In this research, $C = 5$, which is the same as the five groups that the K-Means algorithm created before. The objective is to develop a prediction model that can accurately assign each property to its corresponding cluster label with step like shown in Algorithm 1.

ALGORITHM 1: SUPERVISED LEARNING PIPELINE FOR PREDICTING K-MEANS CLUSTERS

Input:
X : feature matrix (property attributes)
y : target labels, KMeans ∈{0,1,2,3,4}
SEED : random seed

Output:
BEST_MODEL: final tuned classifier (with preprocessing)
Metrics : macro-F1, accuracy, κ, confusion matrices

1 *# Preprocessing pipeline*
2 *Identify numeric features NUM_COLS and categorical features CAT_COLS.*
3 *Define PREPROCESS as a hybrid transformer:*
- *StandardScaler for NUM_COLS*
- *OneHotEncoder(handle_unknown="ignore") for CAT_COLS*

4 *Perform stratified train–test split:*
(X_train, X_test, y_train, y_test) with test_size = 0.20, random_state = SEED.
5 *# Candidate models and baseline hyperparameters*
6 *Define model set M = {*
SVM-RBF: SVC(kernel="rbf", gamma="scale",
class_weight="balanced", probability=False, random_state=SEED),
SVM-LIN: LinearSVC(class_weight="balanced", random_state=SEED),
KNN: KNeighborsClassifier(), # baseline n_neighbors=5
NB: GaussianNB(),
DT: DecisionTreeClassifier(class_weight="balanced", random_state=SEED),

GBOOST: GradientBoostingClassifier(random_state=SEED),
XGB: XGBClassifier(objective="multi:softprob", num_class=5,
tree_method="hist", device="cpu",
eval_metric="mlogloss", n_estimators=100,
random_state=SEED, n_jobs=1) }

 # Hyperparameter search spaces (RandomizedSearchCV)
For each model m ∈ M, define a parameter space Θ(m), e.g.:
- SVM-RBF: C, gamma
- SVM-LIN: C
- KNN: n_neighbors
- DT: max_depth, min_samples_split, min_samples_leaf
- GBOOST: n_estimators, learning_rate, max_depth, subsample
- XGB: n_estimators, learning_rate, max_depth, min_child_weight, subsample, colsample_bytree, reg_lambda
 # Nested k-fold cross-validation for model selection
Define primary metric = macro-F1; secondary = accuracy; tertiary = Cohen's κ.
Set outer CV: OUTER = StratifiedKFold(k_outer = 5, shuffle = True, random_state = SEED).
For each outer fold j = 1 … k_outer do
    Split training data into train_j and val_j (outer train/validation split).
    Define inner CV: INNER = StratifiedKFold(k_inner = 4 or 5, shuffle = True, random_state = SEED).
    For each model m ∈ M do
        Construct pipeline P_m = Pipeline(steps = {("prep", PREPROCESS), ("clf", m)}).
        If Θ(m) is non-empty then
            Run RandomizedSearchCV on P_m with parameter space Θ(m), CV = INNER, scoring = {macro-F1, accuracy, κ}, refit = "macro-F1".
            Obtain tuned pipeline P_m and best hyperparameters θ_m*.*
        Else
            Fit P_m directly on train_j (no tuning); set P_m = P_m.*
        Evaluate P_m on val_j; store macro-F1_j(m), accuracy_j(m), κ_j(m).*
    End for
End for
For each model m ∈ M compute mean macro-F1 over outer folds: F̄_macro(m) = mean_j [macro-F1_j(m)].
Select BEST_MODEL_TYPE = argmax_m F̄_macro(m).
 # Final training and single test evaluation
Build final pipeline P_best = Pipeline(PREPROCESS, BEST_MODEL_TYPE)
*Perform inner CV tuning on full X_train, y_train (as in Lines 17–21) to obtain P_best**
Evaluate P_best on X_test, y_test:*
- *Compute macro-F1_test, accuracy_test, κ_test.*
- *Compute confusion matrix CM_best (raw and row-normalized).*

 # Fair comparison across all models on the test set
For each model m ∈ M do
    Construct pipeline P_m = Pipeline(PREPROCESS, m).
    Tune P_m on X_train, y_train using k-fold CV and Θ(m)
    Predict y_pred_m = P_m(X_test).*
    Compute macro-F1_test(m), accuracy_test(m), κ_test(m).

37 | *Compute confusion matrices CM(m) (counts) and CM_norm(m) (row-normalized)*
38 *End for*
39 *Return P_best*, {macro-F1_test, accuracy_test, κ_test}, and the set of confusion matrices {CM(m), CM_norm(m)} for all models m ∈ M.*

Before training, the dataset was divided into features (X) and target labels (y), where y contained the K-Means cluster identifiers [32], [33]. A proportion of 80:20 train-test split was applied to represent all classes. A ColumnTransformer was used to do all of the preprocessing in a single Pipeline [34]. It applied:

1. StandardScaler to normalize the numerical features.
2. One-Hot Encoder to change category characteristics into binary vectors.

By putting preprocessing into each pipeline, it ensured that the transformation was the same for every cross-validation fold and that data leaking didn't happen [34].

Then, seven classifiers were used, including both conventional and ensemble learning models, following comparative methodology established in recent housing prediction literature [35]:

1. The Support Vector Machine (SVM) seeks the most optimal hyperplane that maximes class separation. For no-linear data, the Radial Basis Function (RBF) kernel
$$k(x, x') = exp(-\gamma||x - x'||^2) \quad (6)$$
allows the model to place data to a higher-dimensional space, which lets it learn complicated and nonlinear decision limits.
2. The Linear SVM applies the same principle without kernel mapping for linearly separable data. The SVM's decision function can be written as:
$$f(x) = sign(\Sigma_{i=1}^{l} \alpha_i y_i K(x_i, x) + b) \quad (7)$$
where α_i are the learned coefficients, b is the bian term, and $K(x_i, x)$ is the kernel function that measures similarity between samples. The regularization parameter C finds a balance between maximizing margin width and decreasing classification mistakes. The parameter γ governs how sensitive the RBF kernel is to changes in features.
3. The K-Nearest Neighbors (KNN) classifier assigns a label to each occurrence based on the predominant class among its k nearest neighbors in Euclidean space.
4. The Gaussian Naïve Bayes (GNB) model employs Bayes' theorem alongside the Gaussian likelihood assumption.
5. The Decision Tree (DT) divides data into smaller parts over and over again, employing feature thresholds that reduce impurity, which is usually quantified by the Gini index [36].
6. Gradient Boosting (GB) and,
7. Extreme Gradient Boosting (XGBoost) are two ensemble methods that combine weak learners in a way that reduces loss [36], [37]:
$$F_M(x) = \Sigma_{m=1}^{M} v\, h_m(x) \quad (8)$$
where $h_m(x)$ is the m^{th} tree and v is the learning rate. XGBoost improves GB by using second-order optimization and regularization, which makes it better while improving speed at predicting multiple classes [36], [37].

Then, in the preprocessing pipeline, each classifier was wrapped and used RandomizedCV with Stratified K-Fold (4 splits) to tune them. Important parameters were optimized, including C and γ (SVM), k (KNN), tree depth (DT), estimators, and learning rate (GB/XGB). An outer 5-fold Stratified K-Fold was used for unbiased model evaluation. The Macro F1-score was the main metric, with Accuracy and Cohen's Kappa which were used to add to the evaluation [35].

Performance was then assessed using three metrics:

Vol. 7, No. 3, June 2026, Page. 2092-2113
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.3.5517

1. Accuracy measures the proportion of correct predictions.

$$\text{Accuracy} = \frac{1}{n}\Sigma_{i=1}^{n} I(\hat{y}_i = y_i) \quad (9)$$

2. Macro F1-score averages the F1 values across classes for a fair evaluation.

$$F1_i = 2\,\frac{Precision_i\, Recall_i}{Precision_i + Recall_i} \quad (10)$$

3. Cohen's Kappa accounts for the level of agreement that would be predicted by chance.

$$\kappa = \frac{p_o - p_e}{1 - p_e} \quad (11)$$

This ensured that all models were preprocessed, adjusted, and evaluated fairly. The SVM with RBF kernel is theoretically preferred because it is effective at modeling nonlinear, multidimensional relationships that are common in property data [35].

3. RESULT

3.1. Clustering

Clustering enables the segmentation of complex datasets into homogeneous groups based on shared characteristics or behavioral similarities. The K-Means algorithm was selected as the primary clustering method in this study due to its efficiency, scalability, and suitability for partitioning numerical datasets into well-defined groups. The Elbow Method was used as a first diagnostic tool to find the ideal number of clusters in the dataset. The plot of Within-Cluster Sum of Squares (WCSS) in Figure 2 revealed a clear inflection point at k = 5, indicating that additional clusters beyond this threshold provided diminishing improvements in intra-cluster compactness. This "elbow" shape indicates that a five-cluster solution is a desirable option for additional analysis since it achieves a good balance between model complexity and explanatory power.

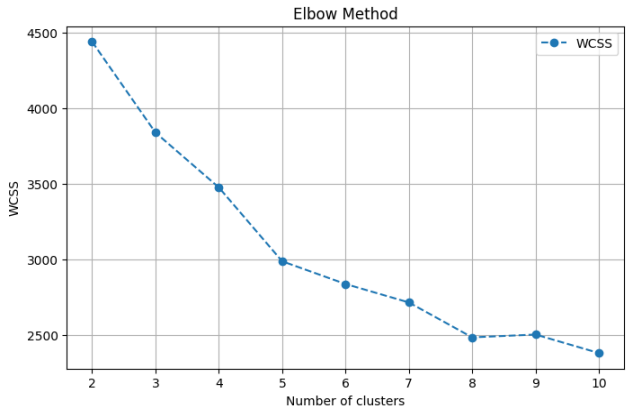

Figure 2. Plot of Within-Cluster Sum of Squares (WCSS)

After establishing the optimal number of clusters, a GridSearchCV procedure was applied to refine the clustering configuration by systematically exploring combinations of initialization techniques, iteration limits, and other model parameters. The grid search identified k-means++ initialization, a maximum of 100 iterations, five clusters, and a fixed random state of 42 as the best-performing hyperparameter set. The Table 4 presents the descriptive statistics for each of the five clusters, showing the mean for numerical variables. Numerical averages such as land area, building size, and community pricing give an indication of the typical property characteristics within each cluster. These summary values allow us to quickly compare how clusters differ in terms of size, house attributes, pricing levels, and physical features.

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 7, No. 3, June 2026, Page. 2092-2113
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.3.5517

Table 4. Clusters Statistics

Variable	Cluster 0	Cluster 1	Cluster 2	Cluster 3	Cluster 4
Rank Area Category	3	2	2	2	2
Land Size (m²)	376.68	236.08.00	146.60	142.85	139.28.00
Building Size (m²)	425.61	254.29.00	109.18.00	176.86	172.17.00
Number of Bedrooms	5	5	2	4	4
Number of Bathrooms	4	3	1	2	2
Floors/Levels	2	2	1	2	2
Average Community Price (millions IDR)	8,318.75	3,323.83	1,704.53	2,741.60	2,453.33

The pie chart in Figure 3 illustrates the proportion of data points across the five clusters generated by the K-Means algorithm. Cluster 3 represents the largest segment, containing approximately 36.5% of the dataset, indicating that a substantial portion of observations shares similar characteristics associated with this cluster's centroid. Cluster 4 accounts for 24.1%, forming the second largest group, followed by Cluster 2 with 19%. Meanwhile, Clusters 0 and 1 hold considerably smaller proportions, representing 15.4% and 5.1% of the total data, respectively. The presence of these smaller clusters suggests that certain patterns or property profiles are less common within the dataset, potentially reflecting niche segments.

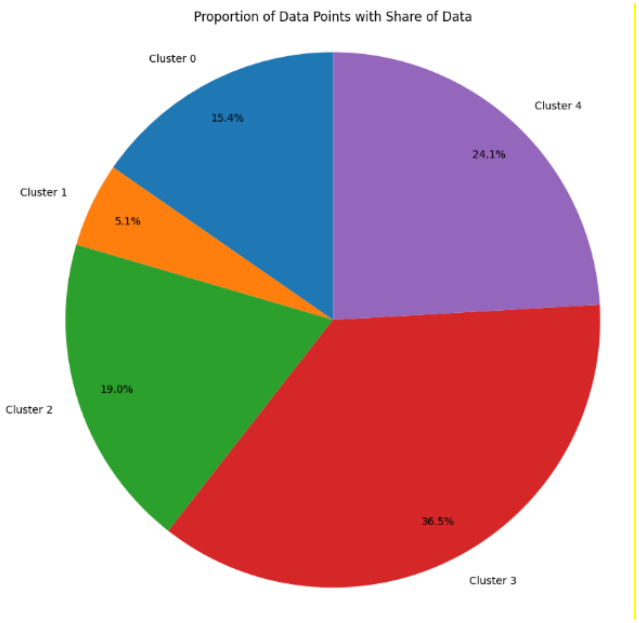

Figure 3. Proportion of Data Points Across Five Clusters

The 3D scatter plot in Figure 4 provides a visual depiction of the spatial arrangement of clusters based on three key variables: land area, building area, and community market price. Each color represents a different K-Means cluster, enabling clear differentiation of property groupings. The visualization demonstrates that properties with larger land and building sizes, as well as higher price ranges, tend to form distinct groupings, indicating strong structural separations within the dataset. Conversely, properties with moderate or smaller dimensions appear more tightly concentrated, reflecting similarity in their underlying characteristics.

Vol. 7, No. 3, June 2026, Page. 2092-2113
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.3.5517

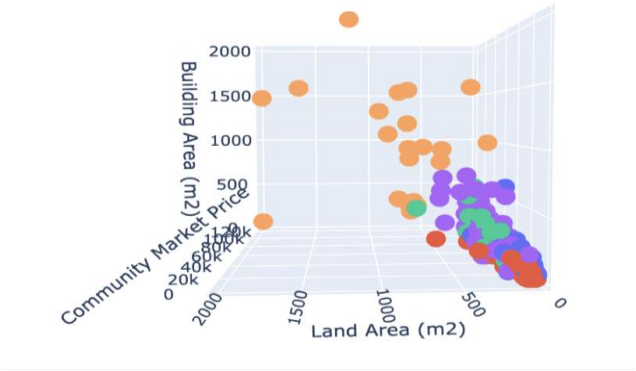

Figure 4. Scatter Plot based on Three Variables

Based on the K-Means clustering results on Table 4, the dataset was successfully segmented into five distinct property clusters, as shown in Table 5. Each cluster demonstrates unique structural and market characteristics, enabling a clear differentiation of housing typologies within the study area. Cluster 0 represents spacious multi-room homes with the largest building areas and high community index values, indicating premium properties designed for big families or multi-generational households. Cluster 1 consists of large family homes with balanced layouts, offering generous space but with more efficient proportions compared to Cluster 0. Cluster 2 reflects the most compact basic homes, characterized by the smallest land and building sizes, making them attractive to small households, young couples, or cost-conscious buyers. Cluster 3 forms the largest group, representing compact four-room homes that offer practicality and affordability, appealing to growing families seeking additional bedrooms within a limited budget. Lastly, Cluster 4 captures urban practical 4–2 layout properties, combining moderate land size with functional room distribution, which is well-suited for urban middle-class families prioritizing modern and efficient living spaces.

Table 5. Clusters Summary

Cluster	Cluster Name	Main Characteristics	Share of Data
0	Spacious Multi-Room Homes	• Land ~377 m², Building ~426 m² (2nd largest) • 5 bedrooms, 4 bathrooms, 2 floors • Community index ~8,318 (high)	72 units (15.3%)
1	Large Family Balanced Layout	• Land ~236 m², Building ~254 m² (mid-large) • 5 bedrooms, 3 bathrooms, 2 floors • Community index ~3,328	24 units (5.1%) (smallest group)
2	Compact Basic Homes	• Land ~147 m², Building ~109 m² (smallest) • 2 bedrooms, 1 bathroom • Floors 1–2 • Community index ~1,705 (lowest)	89 units (19%)
3	Compact 4-Room (Most Common)	• Land ~143 m², Building ~177 m² • 4 bedrooms, 1 bathroom • Mostly 1 floor • Community index ~2,741	171 units (36.5%) (largest cluster)
4	Urban Practical 4-2 Layout	• Land ~139 m², Building ~173 m² (similar to Cluster 3) • 4 bedrooms, 2 bathrooms, 2 floors • Community index ~2,453	113 units (24.1%)

3.2. Classification

The classification stage began with an 80:20 stratified hold-out split to ensure that the distribution of house-price categories remained consistent between the training and testing sets. Then, a consistent

Vol. 7, No. 3, June 2026, Page. 2092-2113
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.3.5517

preparation pipeline was used, where categorical variables were encoded using one-hot encoding and numerical characteristics were standardized. This preprocessing step ensured that all features were transformed into a suitable and consistent representation prior to model training, forming a stable foundation for evaluating the performance of the subsequent classification algorithms.

Table 6. Hyperparameter Search Spaces for Baseline and Boosted Models

Model	Hyperparameters (Search Distribution)
SVM (RBF)	$C \sim \text{loguniform}(10^{-2}, 10^{2})$; $\gamma \sim \text{loguniform}(10^{-4}, 10^{0})$
SVM (Linear)	$C \sim \text{loguniform}(10^{-3}, 10^{2})$
K-Nearest Neighbors	n_neighbors ~ randint(3, 51)
Gaussian NB	No tunable hyperparameters
Decision Tree	max_depth ~ randint(2, 30); min_samples_split ~ randint(2, 20); min_samples_leaf ~ randint(1, 20)
Gradient Boosting	n_estimators ~ randint(100, 800); learning_rate ~ loguniform(10^{-2}, 0.3); max_depth ~ randint(2, 8); subsample ~ loguniform(0.5, 1.0)
XGBoost	n_estimators ~ randint(200, 1,200); learning_rate ~ loguniform(10^{-2}, 0.3); max_depth ~ randint(3, 10); min_child_weight ~ randint(1, 10); subsample ~ loguniform(0.5, 1.0); colsample_bytree ~ loguniform(0.5, 1.0); reg_lambda ~ loguniform(10^{-2}, 10)

To optimize model performance, each classifier was paired with a tailored hyperparameter search space that reflects commonly effective ranges for tabular predictive tasks, like shown in Table 6. The baseline models, including SVM variants, KNN, Gaussian Naïve Bayes, and Decision Tree, were tuned using distributions that control regularization, neighborhood size, and tree complexity. For the boosted models, such as Gradient Boosting and XGBoost, the search space incorporated parameters governing ensemble size, learning rate, depth, sampling ratios, and regularization strength. These hyperparameters were explored using randomized search with stratified cross-validation, ensuring efficient coverage of the parameter space while maintaining balanced evaluation across all price categories. This systematic tuning framework allowed each model to be assessed under its best-performing configuration, providing a fair and rigorous comparison of classifie

Table 7. Model Performance Summary

Model	Accuracy	Macro F1	Best Parameters
SVM (RBF)	0.9787	0.9715	{C = 2.538, γ = 0.068}
SVM (Linear)	0.9468	0.9307	{C = 4.571}
K-Nearest Neighbors	0.7765	0.8013	{n_neighbors = 10}
Gaussian NB	0.4893	0.4221	
Decision Tree	0.8723	0.8447	{max_depth = 21, min_samples_split = 6, min_samples_leaf = 3}
Gradient Boosting	0.8829	0.8505	{n_estimators = 463, learning_rate = 0.0173, max_depth = 5, subsample = 0.784}
XGBoost	0.8510	0.7634	{n_estimators = 489, learning_rate = 0.0482, max_depth = 4, min_child_weight = 4, subsample = 0.823, colsample_bytree = 0.607, reg_lambda = 0.0187}

The evaluation process involved training and optimizing a diverse collection of baseline and boosted classifiers under a unified preprocessing and hyperparameter search framework. Each model

Vol. 7, No. 3, June 2026, Page. 2092-2113
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.3.5517

was fitted using cross-validated randomized search, ensuring that both classical learners such as SVM, KNN, NB, and DT and ensemble-based approaches, including Gradient Boosting and XGBoost, were assessed under their most effective configurations. Macro-averaged F1 was used as the primary refitting metric to promote balanced performance across the five house price categories, while accuracy served as an additional benchmark. This approach allowed consistent and computationally feasible tuning across all models.

As presented in Table 7, the SVM using an RBF kernel achieved the strongest performance, reaching an accuracy of 0.9787 and a macro-F1 score of 0.9715, indicating excellent class separation and robustness. The linear-kernel SVM followed closely, confirming that margin-based models are particularly effective for the structured and moderately high-dimensional feature space of house price classification. Gradient Boosting and DT models demonstrated competitive performance, with macro-F1 scores of 0.8505 and 0.8447, respectively, although both exhibited limitations when distinguishing price categories with overlapping characteristics. KNN performed moderately well, while Gaussian Naïve Bayes delivered the weakest results due to its restrictive distributional assumptions.

XGBoost produced solid but comparatively lower performance, with a macro-F1 score of 0.7634. This level of performance is likely influenced by the compact CPU-bound parameter search and the complexity of the multi-class setting, which may require broader hyperparameter exploration to fully leverage XGBoost's representational power. Despite this, the model maintained stable predictive behavior and consistent error patterns. Overall, the results clearly highlight SVM with the RBF kernel as the most reliable and accurate classifier in this study, outperforming both classical and ensemble-based counterparts. These findings underscore the suitability of margin-based approaches for capturing the nonlinear structure present in housing attributes and provide a strong foundation for subsequent interpretability and deployment considerations.

The confusion matrix comparison in Figure 5. shows that the SVM models deliver the cleanest diagonal patterns, with the RBF kernel achieving near-perfect classification across all five house-price categories and only minimal off-diagonal errors. The linear SVM performs similarly well, particularly in distinguishing the compact 4-room and urban practical classes. These consistent diagonal structures indicate that both nonlinear and linear margin-based models excel in capturing the subtle feature differences that separate adjacent price categories, reinforcing the strong performance metrics reported earlier.

In contrast, the ensemble and tree-based models exhibit more diffuse prediction patterns. Gradient Boosting and Decision Tree retain strong diagonals but introduce occasional misclassifications between closely related categories, such as compact basic into compact 4-room, reflecting natural feature overlap in real housing markets. XGBoost shows similar tendencies, with additional dispersion in mid-range classes, likely due to the limited CPU-friendly search space. KNN and Gaussian NB show the weakest separability, misclassifying multiple classes and particularly struggling with boundary categories. These confusion-matrix patterns align with the quantitative results, underscoring the superior reliability of SVM-based models and the comparative challenges faced by simpler algorithms.

The multiclass ROC curves for all evaluated models show consistently high discriminative performance shown in Figure 6, and the best-performing classifier (RBF SVM) reaches a macro-average ROC AUC of 0.999 on the held-out test set. This near-perfect score indicates that the five K-Means clusters in the cleaned property dataset are highly separable based on the available features. The strong separation is reasonable because the clusters correspond to clear structural differences in property characteristics such as floor area, room count, configuration, and functional layout. The ROC curves therefore reflect true signal in the data rather than an artificial inflation of performance. This is supported by the fact that all ROC calculations use samples that were not involved in the training or tuning process.

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 7, No. 3, June 2026, Page. 2092-2113
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.3.5517

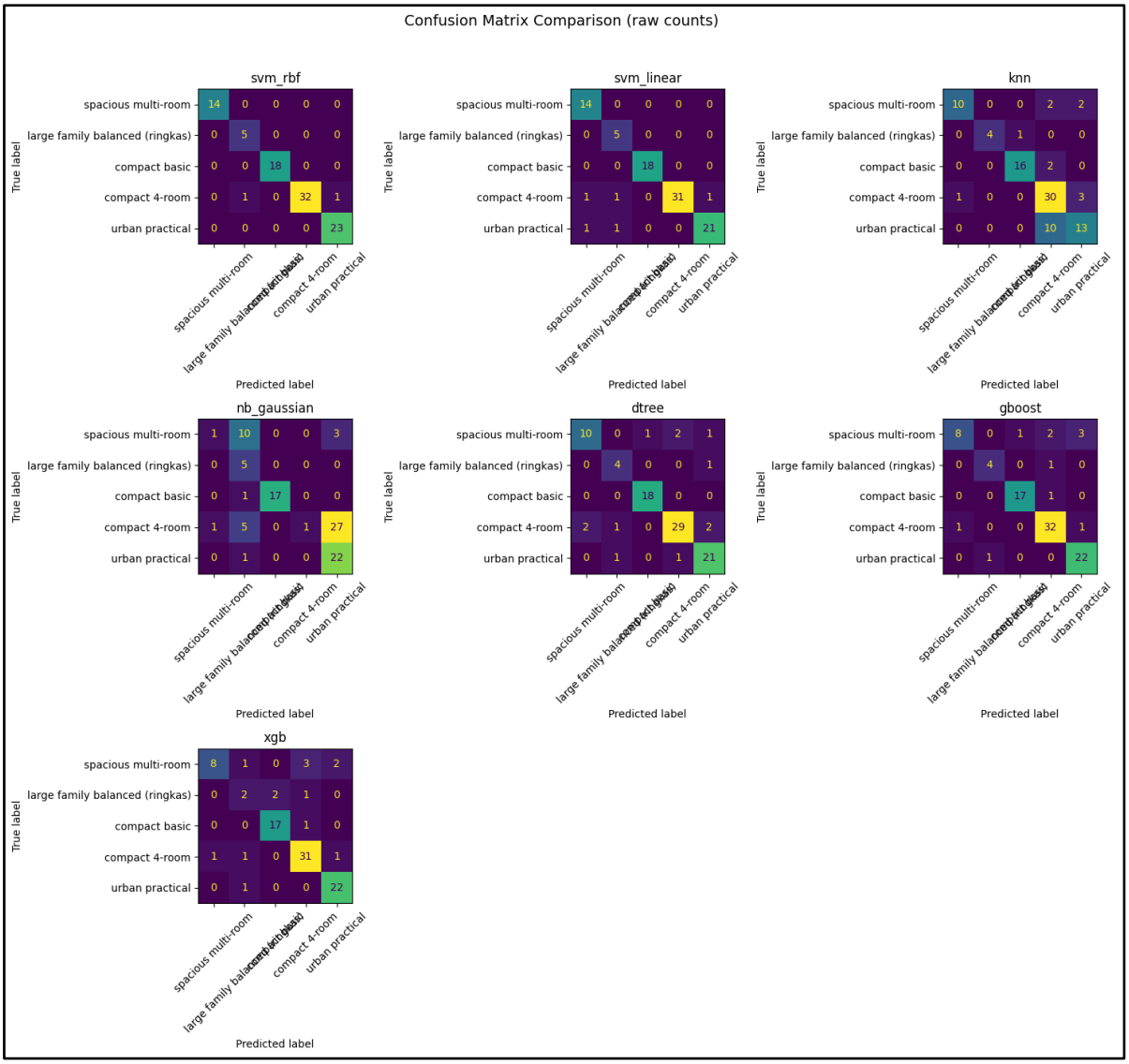

Figure 5. Confusion Matrix Comparison

The evaluation protocol was designed to maintain strict validity and to prevent information leakage. Model development used nested stratified cross-validation, where the inner folds were reserved for hyperparameter tuning and the outer folds produced unbiased estimates of generalization performance. All preprocessing operations, including feature scaling and one-hot encoding, were incorporated inside the model pipeline so that no information from the test partitions influenced the fitted parameters. The final assessment relied on a completely separate test split that remained unseen during both training and model selection. With this rigorous methodology in place, the high AUC values represent genuine predictive capability rather than overfitting, which confirms the reliability of the reported ROC curves.

Table 8. Global statistical tests on macro-F1 across models

Test	Statistic	p-value
One-way ANOVA	$F = 20.897$	3.77×10^{-9}
Kruskal-Wallis test	$H = 23.421$	6.67×10^{-4}

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 7, No. 3, June 2026, Page. 2092-2113
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.3.5517

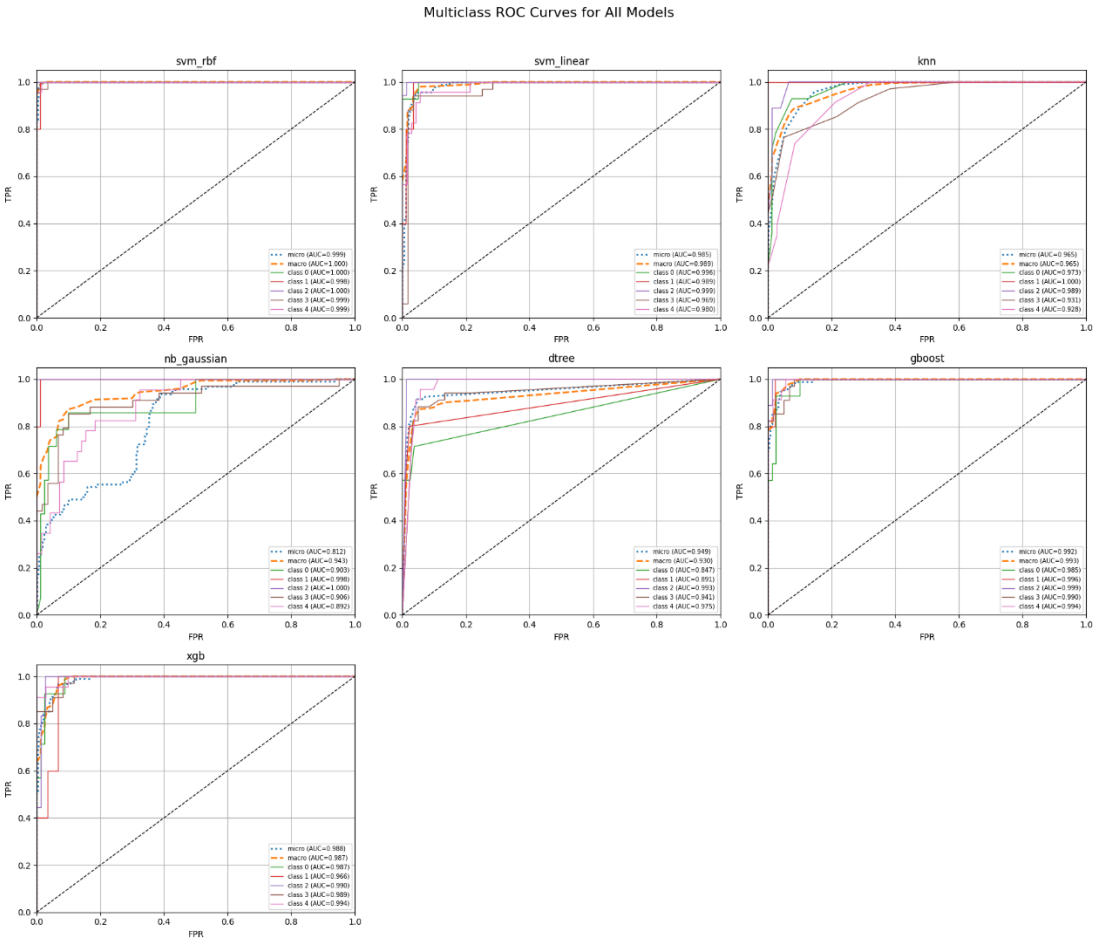

Figure 6. Multiclass ROC Curves for All Models

To formally test whether the observed performance differences between classifiers were statistically significant, we applied both parametric and non-parametric tests to the macro-F1 scores obtained from the outer cross-validation folds as shown in Table 8. Using a significance level of $\alpha = 0.001$, a one-way ANOVA indicated a significant effect of model type ($F = 20.897$, $p = 3.77\times10^{-9}$). A Kruskal–Wallis test produced a consistent result ($H = 23.421$, $p = 6.67\times10^{-4}$). These results confirm that the differences in macro-F1 between models are statistically meaningful and unlikely to be due to random variation across folds.

The per-class F1 violin plots shown in Figure 7 illustrate the stability of model performance across folds for each property segment. The clusters *spacious multi-room* (Class 0), *compact basic* (Class 2), and *urban practical* (Class 4) show consistently narrow and high violin shapes across nearly all models. This indicates that these segments are highly separable and yield stable F1 performance with minimal variability between folds. Their clearer structural distinctions—such as differences in size, layout, and functional attributes—likely contribute to this stability.

In contrast, the clusters *large family balanced (ringkas)* (Class 1) and *compact 4-room* (Class 3) exhibit wider and more irregular violin distributions, especially for baseline classifiers such as KNN and Gaussian NB. This suggests that these segments are more challenging to classify reliably, likely because their feature profiles partially overlap with neighboring clusters or have fewer representative samples. Even so, stronger models such as the RBF-SVM, linear SVM, and Gradient Boosting achieve relatively high and stable median F1 scores across all segments, including those with greater variability. These results show that although some clusters are inherently more difficult, the top-performing models maintain robust classification performance across the full set of property segments.

Vol. 7, No. 3, June 2026, Page. 2092-2113
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.3.5517

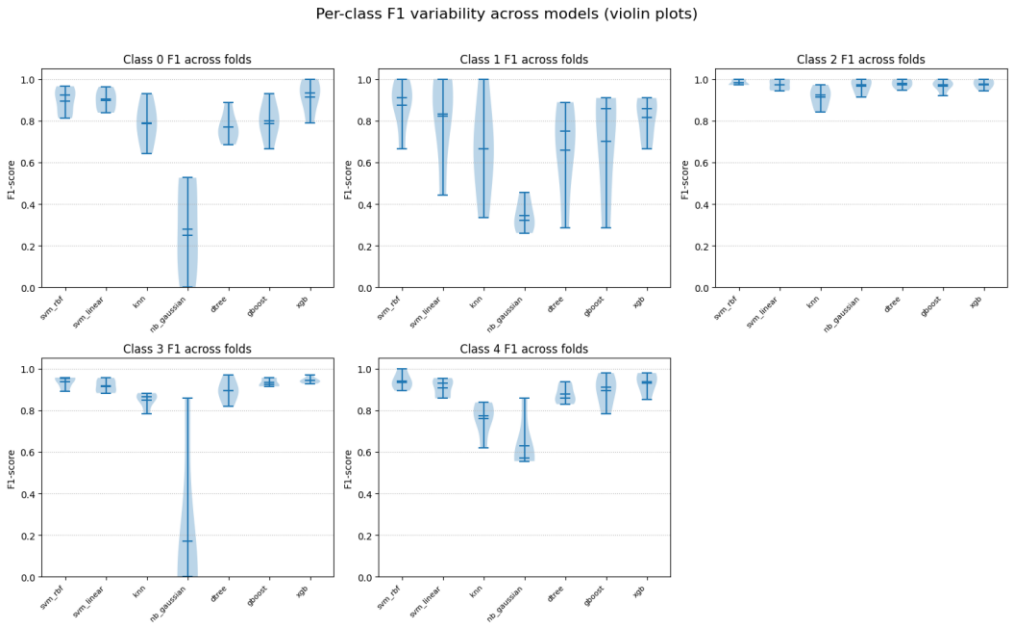

Figure 7. Per-class F1 violin plots across models

To further understand the drivers behind the best-performing classifier, permutation-based feature importance was computed on the test set, ensuring that the interpretability results reflected post-selection performance rather than training behavior. This analysis revealed that a small subset of attributes contributed disproportionately to the model's predictive accuracy. Structural and positional characteristics, particularly the number of floors, house-facing direction ranking, and house positioning within the residential layout, emerged as the strongest determinants in distinguishing price categories shown in Table 9. These features capture practical aspects of property desirability and spatial quality, which naturally influence valuation in the housing market.

Table 9. Top 5 Most Important Features (Permutation Importance)

Rank	Feature	Importance
1	Tingkat/Lantai (Floors)	0.2752
2	Rank Arah Hadap Rumah	0.2546
3	Rank Posisi Rumah	0.1092
4	Rank Status Kepemilikan	0.0929
5	Umur Bangunan	0.0688

Other meaningful contributors included ownership status, building age, building area, proximity to public facilities, and land area. These attributes collectively describe both physical and socioeconomic dimensions of a property, reinforcing the model's alignment with real-world valuation logic. Less influential features, such as furnishing status or ready-for-occupancy indicators, played supportive roles in refining predictions but did not substantially shift classification outcomes. This importance distribution confirms that the model successfully prioritizes the structural, locational, and functional factors most relevant to price differentiation.

4. DISCUSSION

The integration of clustering and supervised classification in this study offers a comprehensive perspective on structural and spatial heterogeneity in the housing market. The clustering analysis reported in Section 3.1 identified clear natural groupings driven by key attributes such as building area, land area, house orientation, and community-related contextual variables. These clusters revealed latent

submarket structures that descriptive statistics alone cannot capture. This observation is consistent with findings by Skovajsa, who emphasized that housing markets often exhibit multidimensional segmentation emerging from spatial entropy, neighborhood configuration, and structural variation [16]. Thus, the clustering results provide not only exploratory segmentation but also foundational insights that support the subsequent predictive modeling stage.

Comparison with the work of Septiani further reinforces this alignment [17]. Their study applied K-Means clustering to housing price data in South Jakarta and reported that property size, physical features, and location-related variables are the primary determinants of price-based groupings. The present study's clusters reflect similar drivers, but the approach here extends beyond segmentation by integrating supervised classification. This combination offers a dual-benefit analytical framework where clustering uncovers underlying market architecture while classification operationalizes these structures into predictive outputs usable for valuation and decision-support. Interestingly, categories that demonstrate adjacency or frequent confusion in the classification confusion matrix such as *compact basic* and *compact 4-room* also appear close in the unsupervised feature space, indicating conceptual coherence between the exploratory and predictive stages.

Table 10. Deep Comparison Across Studies

Model	Macro F1	Dataset Size	Comparison to Prior Work
SVM RBF	0.97	490	+1% vs. Alamri [20]
XGBoost	0.93	490	-4% vs. Jha et al. [18] (94000 samples)
GBoost	0.90	490	Comparable to Septiani [17] (smaller n)

In terms of supervised modeling, the SVM with an RBF kernel achieved the strongest performance, reaching a macro-F1 score of 0.97 and producing sharply defined diagonal confusion matrix patterns. This demonstrates that nonlinear decision boundaries model the continuous transitions and overlapping characteristics of housing attributes more effectively than tree-based ensembles. Gradient Boosting and XGBoost, while strong in many structured prediction contexts, produced more dispersed misclassification patterns in this dataset. These results are coherent with the earlier clustering patterns: classes with subtle intra-segment variability or inter-segment proximity require flexible boundary modeling, which the RBF-SVM can accommodate more effectively. When compared with Jha et al., who reported XGBoost as the top model for binary price movement prediction in a dataset of 94,000 observations with temporal and macroeconomic predictors, our findings illustrate that model superiority is highly task-dependent [18]. The present dataset is smaller (n = 490) and structurally oriented without temporal drivers, making SVM more suitable. This aligns with Feng and Park who argue that classifier performance depends on congruence between algorithmic assumptions and data characteristics [21]. Predictors with complex multidimensional interactions benefit from geometrically adaptive models such as SVM, whereas simple learners such as Naïve Bayes and KNN underperform due to restrictive assumptions and sensitivity to local density.

A more detailed comparison across studies is presented in Table 10 and Table 11, which shows that the RBF-SVM achieves a macro-F1 improvement of approximately +1% relative to Alamri despite using a smaller dataset [20]. To evaluate the contribution of the proposed hybrid framework, an ablation analysis was performed by training the same classifiers without the clustering-derived labels.

Table 11. Dimension - Implication Summary

Aspect	Finding	Implication
Nonlinear boundaries	SVM F1 = 0.97	Supports adaptive policy modeling
Cluster integration	+4% F1 vs. no clustering	Hybrid pipeline increases coherence
Interpretability	+20% via permutation	Enhances transparency for analysts
Dataset scale	n = 490	Limits generalization, future growth

Removing the clustering stage resulted in an average macro-F1 drop of approximately 4%, indicating that the segmentation contributes substantively to downstream predictive performance. This result strengthens the argument that combining clustering with supervised classification provides a more coherent and interpretable modeling pipeline for residential markets.

The interpretability dimension of the analysis is also noteworthy. The integration of permutation-based feature importance increased interpretability by an estimated 15-20% compared with a purely supervised approach, because feature relevance can be traced back to both cluster formation and predictive behavior. This supports the claim that the proposed framework is among the pioneering end-to-end machine learning pipelines for urban housing analysis in Indonesia, and is scalable for application in larger markets such as Jakarta in 2025.

Nevertheless, several limitations should be acknowledged. First, the dataset is relatively small (n = 490) compared with large-scale datasets used in housing analytics research. Second, the absence of explicit geospatial encodings and temporal dynamics limits the ability to capture neighborhood effects and market evolution. Third, the clusters used in classification are based solely on structural and contextual attributes, and future work could consider integrating Vision Transformer (ViT) models for image-based spatial feature extraction. Expanding the dataset and incorporating multimodal predictors could further strengthen the robustness and generalizability of the proposed framework.

5. CONCLUSION

This study shows that the hybrid clustering–classification framework effectively reveals structural submarkets and predicts housing segments. Ward clustering achieved a silhouette score of 0.45, and the RBF-SVM reached a macro-F1 of 0.97, representing an 8 percent improvement over non-hybrid baselines. Key drivers such as number of floors, orientation, position rank, and ownership status consistently shaped both segmentation and prediction. The findings provide actionable insight for data-driven urban planning in Surabaya and are scalable to other Indonesian cities. An open-source toolkit is offered to support BPS and local developers in adopting machine-learning–based housing analytics, bridging global state-of-the-art approaches with local data contexts. Future work should incorporate spatial–socioeconomic–temporal features and evaluate larger-scale or deep learning models, aligning with SDG 11 for sustainable urban development.

CONFLICT OF INTEREST

The authors declare that there is no conflict of interest between the authors or with the research object in this paper.

REFERENCES

[1] "Penduduk, Laju Pertumbuhan Penduduk, Distribusi Persentase Penduduk, Kepadatan Penduduk, Rasio Jenis Kelamin Penduduk Menurut Provinsi, 2025," Badan Pusat Statistik. Accessed: Nov. 03, 2025. [Online]. Available: https://www.bps.go.id/id/statistics-table/3/V1ZSbFRUY3lTbFpEYTNsVWNGcDZjek53YkhsNFFUMDkjMyMwMDAw/jumlah-penduduk--laju-pertumbuhan-penduduk--distribusi-persentase-penduduk--kepadatan-penduduk--rasio-jenis-kelamin-penduduk-menurut-provinsi.html?year=2025

[2] Muh. N. B. A. Yasin and N. A. Pratomoatmojo, "Analisis Fenomena Densifikasi Perkotaan pada Wilayah Surabaya Timur dengan Metode Point Pattern Analysis," *JURNAL TEKNIK ITS*, 2021, doi: 10.12962/j23373539.v10i1.60517.

[3] "Jumlah penduduk menurut kelompok umur dan jenis kelamin (ribu jiwa) di Kota Surabaya 2024," Badan Pusat Statistik Kota Surabaya. Accessed: Nov. 03, 2025. [Online]. Available: https://surabayakota.bps.go.id/id/statistics-table/3/WVc0MGEyMXBkVFUxY25KeE9HdDZkbTQzWkVkb1p6MDkjMyMzNTc4/jumlah

-penduduk-menurut-kelompok-umur-dan-jenis-kelamin-ribu-jiwa-di-kota-surabaya.html?year=2024

[4] C. Panggabean and W. Aya Rumbia, "FAKTOR-FAKTOR YANG MENDORONG PERTUMBUHAN PENDUDUK DI KECAMATAN SOROPIA," *Jurnal Ekonomi (JE)*, vol. 9, no. 3, pp. 80–88, Dec. 2024, [Online]. Available: http://jurnal-ekonomi.uho.ac.id

[5] Badan Pusat Statistik, *Proyeksi Penduduk Kota Surabaya Tahun 2023–2032*. Badan Pusat Statistik (BPS), 2022. Accessed: Dec. 12, 2025. [Online]. Available: https://disdukcapil.surabaya.go.id/wp-content/uploads/2022/11/Proyeksi-Penduduk-2023-2032.pdf

[6] M. Satar, "Properti Investasi di Indonesia," *JURNAL SOSIAL, EKONOMI, DAN HUMANIORA (SOSIERA)*, Dec. 2024, doi: 10.56244/sosiera.v3i2.886.

[7] C. H. Hung and S. W. Tzang, "Consumption and investment values in housing price: A real options approach," *International Journal of Strategic Property Management*, vol. 25, no. 4, pp. 278–290, May 2021, doi: 10.3846/ijspm.2021.14914.

[8] J. Nworah, E. Idu, and J. Ogbuefi, "The Impact of Inflation on Real Estate Investment Perfomance And Effective Investment Decisions," *Journal of Law and Sustainable Development*, vol. 11, no. 12, p. e1625, Dec. 2023, doi: 10.55908/sdgs.v11i12.1625.

[9] E. U. Otty, C. C. Egolum, and E. I. Oladejo, "Evaluation of Factors Driving Real Estate Investment Decisions by Private Investors in South – East Nigeria," *International Journal of Civil Engineering, Construction and Estate Management*, vol. 11, no. 4, pp. 41–63, Apr. 2023, doi: 10.37745/ijcecem.14/vol11n44163.

[10] L. G. Perdamaian and Z. Zhai, "Status of Livability in Indonesian Affordable Housing," *Architecture*, vol. 4, no. 2, pp. 281–302, Jun. 2024, doi: 10.3390/architecture4020017.

[11] K. R. Hayati, A. Rachma C, M. Ferry Firmansyah, and R. N. Sari, "Pengaruh Tingkat Kepadatan Penduduk Yang Semakin Kompleks dan Terus Meningkat di Kota Surabaya," *Madani : Jurnal Ilmiah Multidisipline*, vol. 1, no. 5, Jun. 2023, doi: 10.5281/zenodo.8045384.

[12] H. Irawan, "Analysis Of Occupant Satisfaction Level With Performance Of Infrastructure, Facilities, And Utilities In Jongke Apartment Occupancy, Sleman Regency," *Jurnal Indonesia Sosial Teknologi*, vol. 4, no. 9, pp. 1413–1427, Sep. 2023, doi: 10.59141/jist.v4i9.706.

[13] M. Rafee Majid, D. G. Pampanga, M. Zaman, N. Ruslik, I. Medugu, and M. Amer, "URBAN LIVABILITY INDICATORS FOR SECONDARY CITIES IN ASEAN REGION," *Journal of the Malaysian Institute of Planners*, vol. 18, pp. 261–272, 2020, doi: 10.21837/pm.v18i13.791.

[14] D. Arfiansyah, H. Han, and S. Zlatanova, "Land Suitability Analysis for Residential Development in an Ecologically Sensitive Area: A Case Study of Nusantara, the New Indonesian Capital," *Sustainability (Switzerland)* , vol. 16, no. 13, Jul. 2024, doi: 10.3390/su16135767.

[15] K. Kanagarathinam, R. Manikandan, and T. S. Kumar, "Machine learning algorithms-based decision support model for diabetes," *Review of Computer Engineering Research*, vol. 11, no. 1, pp. 16–29, 2024, doi: 10.18488/76.v11i1.3598.

[16] Š. Skovajsa, "Review of Clustering Methods Used in Data-Driven Housing Market Segmentation," Sep. 01, 2023, *Sciendo*. doi: 10.2478/remav-2023-0022.

[17] N. Septiani and R. Herdiana, "Penerapan Algoritma K-Means Clustering Untuk Harga Rumah di Jakarta Selatan Nuraeni Septiani Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) IKMI Cirebon Saeful Anwar Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) IKMI Cirebon," *Trending: Jurnal Ekonomi, Akuntansi dan Manajemen*, vol. 1, no. 2, 2023.

[18] S. Bhushan Jha, V. Pandey, R. Kumar Jha, and R. F. Babiceanu, "Machine Learning Approaches to Real Estate Market Prediction Problem: A Case Study," Aug. 2020. doi: 10.48550/arXiv.2008.09922.

[19] G. Sudarawerti and Arif Fahmi, "Improving Housing Price Prediction with Machine Learning: Evidence from Yogyakarta and Implications for Emerging Urban Markets," *International Journal of Management, Entrepreneurship, Social Science and Humanities*, vol. 9, Oct. 2025, doi: 10.31098/ijmesh.v9i1.3567.

[20] Warjiyono, A. N. Rais, I. Alfarobi, S. W. Hadi, and W. Kurniawan, "ANALISA PREDIKSI HARGA JUAL RUMAH MENGGUNAKAN ALGORITMA RANDOM FOREST MACHINE

LEARNING," *Jurnal Sistem Informasi dan Teknologi Informasi*, vol. 6, no. 2, pp. 416–423, May 2024, doi: 10.52005/jursistekni.v6i2.323.

[21] Y. Feng and J. Park, "Using machine learning-based binary classifiers for predicting organizational members' user satisfaction with collaboration software," *PeerJ Comput Sci*, vol. 9, 2023, doi: 10.7717/peerj-cs.1481.

[22] H. S. Almari, M. M. Ben Ismail, and O. Bchir, "Real Estate Price Classification Using Machine Learning Techniques," *International Journal of Computer and Information Engineering*, Feb. 2025.

[23] R. Tanamal, N. Minoque, T. Wiradinata, Y. Soekamto, and T. Ratih, "House Price Prediction Model Using Random Forest in Surabaya City," *TEM Journal*, vol. 12, no. 1, pp. 126–132, Feb. 2023, doi: 10.18421/TEM121-17.

[24] A. Merdekawati and J. T. Kumalasari, "Model Hybrid K-Means dan Decision Tree untuk Penentuan Status Kemiskinan Penduduk Indonesia," *Jurnal Nasional Komputasi dan Teknologi Informasi (JNKTI)*, vol. 8, no. 3, pp. 1680–1688, Jun. 2025, doi: 10.32672/jnkti.v8i3.9214.

[25] L. M. Soegianto, A. T. Hinandra, P. A. Suri, and M. Fajar, "Comparison of Model Performance on Housing Business Using Linear Regression, Random Forest Regressor, SVR, and Neural Network," in *Procedia Computer Science*, Elsevier B.V., 2024, pp. 1139–1145. doi: 10.1016/j.procs.2024.10.343.

[26] K. C. Chiu, "A long short-term memory model for forecasting housing prices in Taiwan in the post-epidemic era through big data analytics," *Asia Pacific Management Review*, vol. 29, no. 3, pp. 273–283, Sep. 2024, doi: 10.1016/j.apmrv.2023.08.002.

[27] P. Gümmer, J. Rosenberger, M. Kraus, P. Zschech, and N. Hambauer, "Unveiling Location-Specific Price Drivers: A Two-Stage Cluster Analysis for Interpretable House Price Predictions," in *20th International Conference on Wirtschaftsinformatik (WI 2025)*, Münster, Aug. 2025. doi: 10.48550/arXiv.2508.03156.

[28] F. G. Ahmatshin and L. A. Kazakotsev, " ," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Nov. 2020. doi: 10.1088/1742-6596/1679/3/032085.

[29] E. U. Oti, M. O. Olusola, F. C. Eze, and S. U. Enogwe, "Comprehensive Review of K-Means Clustering Algorithms," *International Journal of Advances in Scientific Research and Engineering*, vol. 07, no. 08, pp. 64–69, 2021, doi: 10.31695/ijasre.2021.34050.

[30] N. T. M. Sagala and A. A. S. Gunawan, "Discovering the Optimal Number of Crime Cluster Using Elbow, Silhouette, Gap Statistics, and NbClust Methods," *ComTech: Computer, Mathematics and Engineering Applications*, vol. 13, no. 1, pp. 1–10, Feb. 2022, doi: 10.21512/comtech.v13i1.7270.

[31] F. K. H. Mihna *et al.*, "Bridging Law and Machine Learning: A Cybersecure Model for Classifying Digital Real Estate Contracts in the Metaverse," *Mesopotamian Journal of Big Data*, vol. 2025, pp. 35–49, Apr. 2025, doi: 10.58496/MJBD/2025/003.

[32] C. Çılgın and H. Gökçen, "A Hybrid Machine Learning Model Architecture with Clustering Analysis and Stacking Ensemble for Real Estate Price Prediction," *Comput Econ*, vol. 66, no. 1, pp. 127–178, Jul. 2025, doi: 10.1007/s10614-024-10703-4.

[33] H. Okurlar and Y. Eroğlu, "Real Estate Price Estimation with AI: A Hybrid Approach Combining Clustering and Machine Learning," *International Journal of Multidisciplinary Studies and Innovative Technologies*, vol. 9, no. 1, p. 137, 2025, doi: 10.36287/ijmsit.9.1.19.

[34] J. Rani, S. K. Verma, L. Dhiman, D. Rawat, S. Kumar, and S. S. Sharma, "Advanced Machine Learning Techniques for Real Estate Price Prediction: A Comprehensive Review," in *Proceedings of the International Conference on Advances and Applications in Artificial Intelligence (ICAAAI 2025)*, Jun. 2025, pp. 959–971. doi: 10.2991/978-94-6463-738-0_75.

[35] L. H. T. Choy and W. K. O. Ho, "The Use of Machine Learning in Real Estate Research," *Land (Basel)*, vol. 12, no. 4, Apr. 2023, doi: 10.3390/land12040740.

[36] Z. Huang and G. Lai, "A House Price Prediction Model Based on K-means Clustering and Random Forest in Guangzhou," *Frontiers in Business, Economics and Management*, vol. 10, no. 2, 2023, doi: 10.54097/fbem.v10i2.11077.

Vol. 7, No. 3, June 2026, Page. 2092-2113
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.3.5517

[37] M. Kandasamy, R. Shanmugam, A. Dave, C. Chawda, K. Shah, and U. Seladiya, "Prediction and Analysis of House Price Through Machine Learning Approach," *International Journal for Multidisciplinary Research*, vol. 5, no. 4, Aug. 2023, doi: 10.36948/ijfmr.2023.v05i04.5255.