G-Tech : Jurnal Teknologi Terapan
Vol. 10, No. 3, July 2026, p. 1255-1264
E-ISSN: 2623-064X | P-ISSN: 2580-8737

Classification of Hotel Maintenance Levels Using Principal Component Analysis and Support Vector Machine

Bagus Gilang Pratama[1]✉, Sely Novita Sari[2], Rizal Maulana[3], Zainul Arifin[4], Annisa Fauziah[5]

[1, 4] Electrical Engineering Study Program, Faculty of Engineering and Planning, Institut Teknologi Nasional Yogyakarta (ITNY), Indonesia
[2, 3,5] Civil EngineeringStudy Program, Faculty of Engineering and Planning, Institut Teknologi Nasional Yogyakarta (ITNY), Indonesia

✉ **Corresponding Author :** Bagus Gilang Pratama (e-mail: bagusgilangp@itny.ac.id)

Article Information

Article History

Received : June 16, 2026
Revised : July 05, 2026
Published : July 20, 2026

***Keywords*:**

Support Vector Machine;
Principal Component Analysis;
Multi-class Classification;
Hotel Building Maintenance;
Machine Learning.

ABSTRACT

Hotel building maintenance in tourism-driven regions such as the Special Region of Yogyakarta is essential for maintaining service quality and building reliability. However, conventional assessment methods often depend on subjective expert judgment, which may be time-consuming and inconsistent. This study evaluates the performance of Support Vector Machine (SVM) combined with Principal Component Analysis (PCA) to classify hotel building maintenance levels in Yogyakarta into five categories. The dataset includes 175 samples and 11 variables covering architectural, structural, mechanical, electrical, outdoor space, and housekeeping aspects, with naturally imbalanced class distribution. Data were normalized using MinMaxScaler and reduced to nine principal components, explaining 93.41% of cumulative variance. Three SVM kernels polynomial, radial basis function (RBF), and sigmoid were tested using a 70:30 training–testing split with default scikit-learn hyperparameters. The sigmoid kernel produced the best performance, achieving 90.57% accuracy, 91.32% precision, 90.57% recall, and 90.53% F1-score, outperforming RBF and polynomial kernels. Stratified 5-Fold cross-validation showed an average accuracy of 81.71% ± 9.66%. The results indicate that PCA-SVM with a sigmoid kernel is effective for automated hotel building maintenance classification.

INTRODUCTION

Hotel buildings constitute important tourism infrastructure assets that contribute significantly to economic activities, particularly in tourism-oriented regions such as the Special Region of Yogyakarta. To ensure that building functions and services remain optimal, regular maintenance and upkeep activities are required in accordance with national regulations governing building management. In Indonesia, these activities are regulated by Ministerial Regulation No. 24/PRT/M/2008 (Sari et al., 2024), which prescribes periodic maintenance and care of building assets, including architectural, structural, mechanical, electrical, outdoor space, and housekeeping components. Effective hotel maintenance management directly influences guest satisfaction, service continuity, and operational efficiency (Atkin & Bildsten, 2017).

The assessment of hotel building maintenance levels is commonly conducted through manual expert evaluation based on visual inspection and professional judgment. Although this

This is an open access article under the CC BY license

method is widely used, it has several limitations: results may be influenced by assessor subjectivity, evaluations require considerable time, and different experts may produce inconsistent conclusions. As a result, the allocation of maintenance budgets and decisions regarding facility management can become suboptimal, particularly in hotels with high occupancy that require rapid and consistent maintenance decision-making. A quantitative, data-driven approach capable of mapping building conditions into maintenance-level categories in an objective, consistent, and reproducible manner is therefore needed. Automatic classification based on machine learning is one promising approach that has been widely applied in facility management and building condition assessment domains (Han et al., 2022).

A potential machine learning method for addressing this need is the Support Vector Machine (SVM). SVM is a supervised classification algorithm that constructs a decision boundary by determining the most appropriate hyperplane to separate data classes. By maximizing the margin between classes, SVM is able to improve classification stability, limit overfitting, and maintain strong generalization performance, especially when applied to small and medium-sized datasets (Tsiamis et al., 2023). For non-linear classification tasks, SVM applies a kernel function to transform the original data into a higher-dimensional feature space. This transformation allows patterns that cannot be separated linearly in the original space to become more distinguishable and separable in the transformed space (Valkenborg et al., 2023). Some commonly used kernels include polynomials, radial base functions (RBF), and sigmoids, each of which has different characteristics in mapping feature spaces (Ben-Hur & Weston, 2010).

When the number of features is large and potentially multicollinear, dimensionality reduction methods, such as Principal Component Analysis (PCA) can be combined with classification algorithms to retain important information while lowering the complexity of the feature space (Jolliffe & Cadima, 2016). In the context of hotel building maintenance assessment in Yogyakarta, an assessment instrument has been developed to evaluate maintenance implementation across various building components, ranging from architectural elements to housekeeping aspects (Sari et al., 2024). Based on the same assessment instrument, a previous study employed an Artificial Neural Network (ANN) model to estimate the implementation level of hotel building maintenance in Yogyakarta, with encouraging predictive performance (Pratama et al., 2025). Subsequent research further investigated the use of PCA-based indicator reduction to evaluate its effect on ANN performance in classifying hotel maintenance levels using a comparable dataset (Pratama et al., 2026). Although ANN-based approaches have been explored both with and without PCA, the performance of the Support Vector Machine algorithm combined with PCA in the context of hotel building maintenance assessment in Yogyakarta has not been widely studied. This is the research gap that is the focus of this study.

Although prior ANN-based studies have demonstrated feasibility, several practical limitations of ANN motivate the exploration of alternative algorithms. First, ANN training and re-training are computationally intensive (approximately 20 seconds per fold for the previous studies), which becomes prohibitive when the model must be periodically re-trained with newly collected hotel data. Second, ANN requires careful hyperparameter tuning including learning rate, batch size, epoch count, and layer architecture each of which materially affects performance. Third, ANN typically offers limited interpretability, which is a significant barrier to adoption by facility managers who must justify maintenance-related decisions to hotel management. Fourth, single ANN models may exhibit higher variance across cross-validation folds compared with simpler geometric classifiers. In contrast, SVM offers a maximum-margin geometric approach with fewer hyperparameters, faster training, and a mathematically transparent formulation. The critical question, then, is whether SVM combined with PCA can approximate ANN performance closely enough to justify its practical advantages for real-world deployment in hotel maintenance decision-support systems. This constitutes the research gap addressed by the present study. Based on the above background and identified research gap, the specific objectives of this study are:

1. To evaluate the performance of SVM with three different kernel functions polynomial, RBF, and sigmoid in classifying hotel building maintenance levels in Yogyakarta into five ordinal categories;
2. To analyze the contribution of PCA-based dimensionality reduction to model performance and to justify the selected number of principal components based on cumulative-variance retention;
3. To assess model stability and generalization capability using Stratified 5-Fold cross-validation; and
4. To provide a critical comparison with the previously reported ANN-based approach in order to position PCA-SVM as a viable, lightweight alternative for automated hotel building maintenance classification systems.

RESEARCH METHODS

This study uses a computational experimental quantitative approach. The experiment was carried out by implementing the SVM algorithm on the assessment dataset of hotel building maintenance and maintenance in Yogyakarta. The entire modeling process was carried out using the Python programming language with the scikit-learn library (Raschka et al., 2020) for SVM, PCA, and MinMaxScaler, supported by pandas and numpy libraries for data manipulation and matplotlib and seaborn for visualization of results. The research flow broadly includes the stages of data collection, pre-processing, data sharing, SVM modeling, and evaluation.

Dataset

The dataset used consisted of 175 observation samples of the assessment of hotel building maintenance and maintenance in the Special Region of Yogyakarta area with 11 input variables (features) and one target variable. The assessment instrument used as the source of the features in this study is based on a maintenance evaluation framework that covers various building components and maintenance aspects (Sari et al., 2024). The eleven features represent the assessment scores on the maintenance and maintenance aspects of the six building components as presented in Table 1. Each feature has a value range of [0.1] which is the normalized score of the inspection assessment results. The target variable is in the form of maintenance level classes which are divided into five categories (Class 0 to Class 4).

The class distribution is naturally imbalanced, reflecting real-world hotel maintenance patterns in Yogyakarta: Class 0 (very low) accounts for 13.1% of samples, Class 1 (low) 17.1%, Class 2 (moderate) 39.4% as the majority class, Class 3 (high) 18.9%, and Class 4 (very high) 11.4% as the minority class. This imbalance is not an artifact but reflects field reality most hotels fall in the moderate maintenance category, while extreme conditions (very low or very high maintenance) are less common. This distribution characteristic informs the choice of evaluation metrics, motivating the use of macro-average and per-class metrics in addition to overall accuracy.

Table 1. List of Hotel Building Maintenance and Maintenance Assessment Features

No	Feature Name	Building Components
1	Architectural Maintenance	Architecture
2	Architectural Care	Architecture
3	Structure Maintenance	Structure
4	Structural Care	Structure
5	Mechanical Maintenance and Maintenance	Mechanical
6	Electrical Maintenance	Electric
7	Electrical Care	Electric

8	Outdoor Space Maintenance	Outdoor Space
9	Outdoor Space Care	Outdoor Space
10	Housekeeping Maintenance	Housekeeping
11	Housekeeping Care	Housekeeping

Data Pre-processing

The pre-processing stage is carried out in two steps. First, normalize the data using MinMaxScaler which transforms each feature to the range [0, 1] with the formula x_norm = (x − x_min) / (x_max − x_min). The normalization stage is important because SVMs with distance-based kernels and PCAs are sensitive to feature scale (Douglass, 2020). Second, dimension reduction uses PCA with the number of main components set at nine components.

The PCA works by projecting the data in the directions with the largest variants so that the resulting components are orthogonal and uncorrelated (Jolliffe & Cadima, 2016). Empirical evaluation showed that seven principal components retained approximately 84.9% of the total variance, eight components retained approximately 89.4%, and nine components retained 93.41%, exceeding the commonly used 90% cumulative-variance threshold. Using more than nine components, such as ten or eleven, would retain only marginal additional information while reducing the benefit of dimensionality reduction. Conversely, using fewer than nine components, such as seven or eight, could risk losing relevant data structures, particularly those associated with minority classes. Therefore, nine components were selected as a balanced and justifiable trade-off between information preservation and feature-space simplification. This configuration is also consistent with the prior ANN-PCA study conducted on similar data, thereby enabling a fair methodological comparison with previous findings (Pratama et al., 2026).

Data Splitting

The dataset was split into training and test data with a 70:30 ratio using random_state = 0 to ensure reproducibility. This split resulted in 122 training samples and 53 test samples. Stratified sampling was applied to preserve the class distribution in both subsets. The split was performed after normalization and PCA had been fitted only on the training data and then applied to the test data, avoiding data leakage.

SVM Model Configuration and Hyperparameter Selection

The Support Vector Machine algorithm works by looking for an optimal hyperplane that maximizes the margins between classes (Pisner & Schnyer, 2020; Tsiamis et al., 2023). For multi-class problems, scikit-learn defaults to a one-vs-one strategy. Effective SVM classification can be achieved through a systematic evaluation of kernel functions and regularization parameters to identify the most appropriate model configuration (Ben-Hur & Weston, 2010). Referring to these recommendations, this study tested three SVM kernels with the following configurations: (1) polynomial kernels with degrees 3, gamma = auto, and C = 1.0; (2) RBF kernels with gamma = scale and C = 1.0; and (3) sigmoid kernels with gamma = scale and C = 1.0. The regulation parameter value of C = 1.0 is selected as the default setting of scikit-learn which is commonly used as the starting point for the test. The function form of each kernel can be written as follows (Valkenborg et al., 2023):

$$\text{Kernel polynomial: } K(x, y) = (\gamma \cdot x^{T}y + r)^{d}, \text{ where d = polynomial degrees.} \quad (1)$$

$$\text{Kernel RBF: } K(x, y) = \exp(-\gamma \cdot \|x - y\|^{2}). \quad (2)$$

$$\text{Kernel sigmoid: } K(x, y) = \tanh(\gamma \cdot x^{T}y + r). \quad (3)$$

Regarding hyperparameter selection, the regularization parameter C, gamma, and polynomial degree were deliberately set to scikit-learn's default values (C = 1.0, gamma = 'scale', degree = 3). This methodological choice was made for two principal reasons. First, using default parameters across all three kernels ensures a fair kernel-level comparison in which observed

performance differences reflect the intrinsic behavior of each kernel rather than differential tuning effort a critical requirement for the primary research objective of characterizing kernel behavior on this class of data. Second, the default values in scikit-learn are calibrated to provide reasonable baseline performance across a wide range of problems (Rajamani & Iyer, 2023; Tran et al., 2022) and are widely adopted as a starting point in comparative kernel studies. It is acknowledged that subsequent hyperparameter optimization (e.g., via grid search or Bayesian optimization) on the best-performing kernel is likely to further improve absolute performance; this optimization is intentionally scoped as future work rather than being interleaved with the comparative analysis, in order to preserve the interpretability of the comparison presented here.

Evaluation Metrics

The model performance was assessed using four evaluation metrics: accuracy, precision, recall, and F1-score (Ruppert, 2004). Accuracy represents the proportion of correctly classified instances among all predictions, while precision indicates the proportion of correct predictions within each predicted class. Recall reflects the model's ability to identify samples belonging to a given class, and the F1-score provides a balanced measure by combining precision and recall. In this multi-class classification task, the metrics were computed using a weighted average according to the number of samples in each class. In addition, a confusion matrix was employed to examine the distribution and pattern of classification errors across classes. To assess the robustness and generalization ability of the model while reducing dependency on a single train–test split, a 5-fold Stratified K-Fold cross-validation procedure was applied (Tsamardinos et al., 2018). In each fold, both normalization and PCA were performed only on the training subset to prevent data leakage.

RESULTS AND DISCUSSION

PCA-based Dimensionality Reduction Results

The application of PCA to the normalized training data (122 samples, 11 features) produced nine principal components with a cumulative variance of 93.41%. The variance explained by each component is presented in Figure 1. The first component (PC1) captured the largest contribution at approximately 43.8% of variance, while the second component (PC2) accounted for approximately 10.5%. Components three through nine contributed progressively smaller portions of variance, together reaching the 93.41% cumulative threshold.

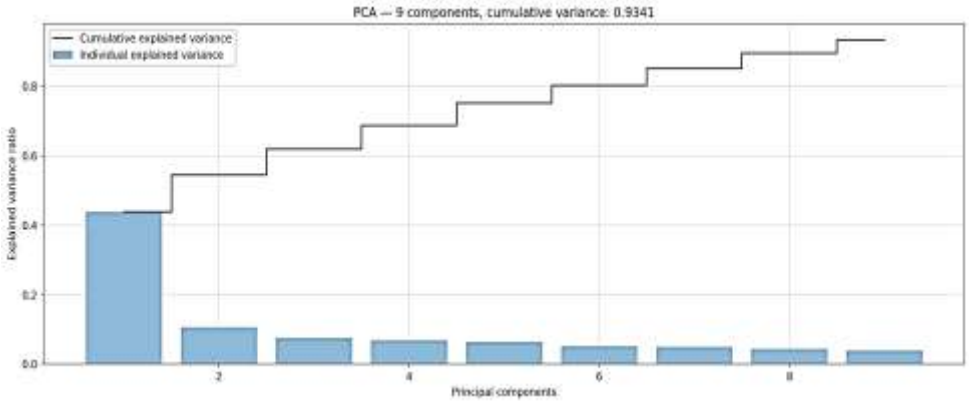

Figure 1. Variance described by each of the main components of PCA (cumulative 93.41%).

PC1, which absorbs nearly half of the total variance, indicates the presence of a dominant latent dimension in the data, most likely representing overall maintenance quality that correlates positively across all indicators. Hotels with well-managed maintenance tend to score consistently high across all components, including architecture, structure, MEP, outdoor space, and housekeeping, while poorly managed hotels tend to score consistently low. Meanwhile, PC2 likely represents a contrast between structural components, such as architecture and structure, and operational components, such as mechanical, electrical, and housekeeping. This component captures hotels where the physical structure is well maintained but MEP systems lag behind, or

vice versa. These findings support the effectiveness of PCA in reducing hotel maintenance indicators while preserving meaningful data structures on similar hotel maintenance datasets (Pratama et al., 2026).

Best SVM Kernel Selection

Testing against three SVM kernels yielded very different performance as shown in Table 2. The polynomial kernel provides the lowest performance with an accuracy of 39.62% indicating the kernel's inability to map data to feature spaces that properly separate classes. A very low precision value (0.157) indicates that the model tends to predict only one or two classes so that most classes are not selected as predictions. This is thought to be due to the form of third-degree polynomial decision limits that are too rigid for data distribution after PCA. The RBF kernel provides much better performance with an accuracy of 83.02%, but it is still below the sigmoid kernel which achieved an accuracy of 90.57%.

Table 2. Comparison of SVM Performance with Three Kernels on Test Data

Kernel	Akurasi	Presisi	Recall	F1-Score
Polynomial (d=3)	0,3962	0,1570	0,3962	0,2249
RBF	0,8302	0,8337	0,8302	0,8301
Sigmoid	0,9057	0,9132	0,9057	0,9053

Beyond the numerical differences, the observed kernel performance can be explained through the interaction between each kernel's mathematical structure and the characteristics of the dataset after PCA. The following paragraphs discuss the scientific reasoning underlying each kernel's behavior. The failure of the polynomial kernel can be understood from both theoretical and empirical perspectives. Theoretically, the polynomial kernel maps input features into a very high-dimensional feature space containing all monomial combinations up to degree three. For a training set of only 122 samples with 9 dimensions after PCA, this induced feature space becomes vastly over-parameterized relative to the sample count, producing the classical 'curse of dimensionality' effect (Burges, 1998). Empirically, this manifests as the model retaining 118 of 122 training samples (approximately 97%) as support vectors an indication that the model has failed to identify a compact discriminative structure and is instead effectively memorizing the training data. Consequently, precision collapses to 0.157 because the classifier degenerates into predicting only one or two dominant classes, and recall on minority classes approaches zero. Increasing the degree or using an alternative polynomial form would not resolve this issue without proportionally increasing the sample size.

The RBF kernel performs substantially better because its isotropic Gaussian similarity structure effectively captures local neighborhoods, and it is generally regarded as a robust default choice (Luts et al., 2010). However, its performance still falls short of the sigmoid kernel for two structural reasons. First, after PCA transformation the data occupies a compressed range on each principal component axis, so pairwise Euclidean distances between samples become relatively small; in this regime, the RBF kernel with gamma = 'scale' behaves increasingly like a linear kernel, partially losing its non-linear advantage. Second and more importantly the target variable has an ordinal structure (Class $0 < 1 < 2 < 3 < 4$), which is not naturally suited to isotropic radial basis functions that treat all feature-space directions equally. This mismatch between the isotropic kernel geometry and the anisotropic ordinal class structure produces suboptimal class separation, particularly near ordinal class boundaries.

The sigmoid kernel's superior performance is attributable to a mathematical equivalence that is directly relevant to this dataset: the sigmoid transformation $\tanh(\gamma x^T y + r)$ implements the same activation function as a single-hidden-layer neural network. In effect, SVM-sigmoid performs a neural-network-like feature transformation while retaining SVM's maximum-margin optimization framework, combining the representational flexibility of a shallow neural network with the geometric regularization of the margin principle. This mechanism explains why the sigmoid kernel produced performance comparable to the strong ANN-based results reported in the same problem

domain, as both approaches rely on similar non-linear transformation characteristics (Pratama et al., 2025, 2026). Furthermore, the sigmoid kernel produced the fewest support vectors, with 103 out of 122 samples, or approximately 84%, among the three tested kernels. This indicates a more parsimonious decision function and stronger generalization potential.

The distribution of support vectors across classes for the sigmoid kernel was 14 for Class 0, 20 for Class 1, 32 for Class 2, 21 for Class 3, and 16 for Class 4. Classes at the extremes, namely Class 0 and Class 4, required fewer support vectors, while the middle classes, namely Classes 1 and 3, required proportionally more support vectors to define their decision boundaries. This pattern is consistent with the intuition that adjacent classes are more difficult to separate than extreme classes.

Sigmoid Kernel SVM Performance Analysis

The performance of SVM with the sigmoid kernel on the test set is presented in detail through the confusion matrix in Figure 2 and the classification report in Table 3. Of the 53 test samples, 48 were correctly classified and 5 were misclassified.

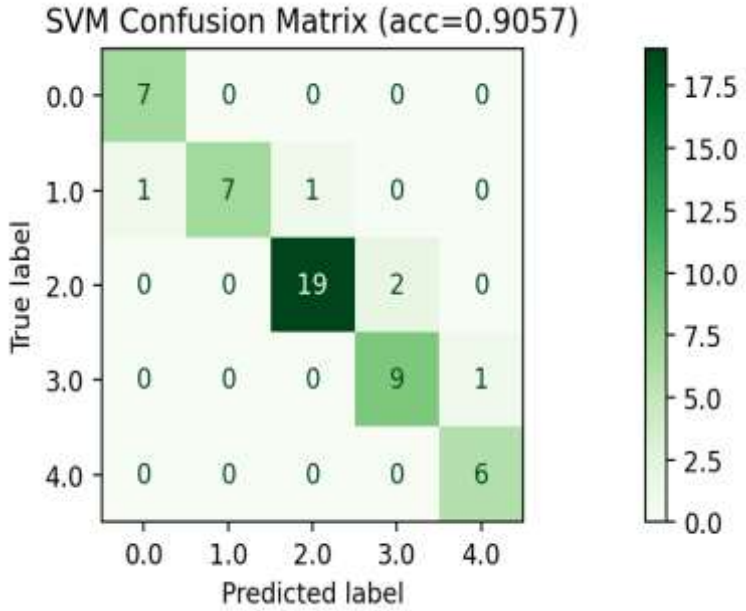

Figure 2. Confusion Matrix SVM Sigmoid Kernel on Test Data (accuracy = 0.9057).

Table 3. SVM Sigmoid Kernel Classification Report on Test Data

Classes	Precision	Recall	F1-Score	Support
0	0,88	1,00	0,93	7
1	1,00	0,78	0,88	9
2	0,95	0,90	0,93	21
3	0,82	0,90	0,86	10
4	0,86	1,00	0,92	6
Accuracy	—	—	0,91	53
Macro avg	0,90	0,92	0,90	53
Weighted avg	0,91	0,91	0,91	53

Class 0 and Class 4 achieved perfect recall (1.00), meaning all samples in these two classes were correctly identified. This indicates that extreme conditions (lowest and highest maintenance levels) have distinctive patterns that are easily separated by the model. Class 1 exhibited the lowest recall (0.78) because two of nine samples in this class were misclassified into neighboring classes

(one to Class 0 and one to Class 2). Classes 2 and 3 achieved a recall of 0.90 each, with two Class 2 samples misclassified as Class 3 and one Class 3 sample misclassified as Class 4.

The error pattern reveals that misclassifications occur exclusively between adjacent classes (0↔1, 2↔3, 3↔4). This indicates that decision boundaries between adjacent classes exhibit gray areas, a characteristic feature of ordinal data. No cases of non-adjacent misclassification (e.g., Class 0 predicted as Class 3) were observed, which shows that the model has correctly captured the ordinal structure of maintenance levels. This is a valuable property for facility management applications, where errors between adjacent maintenance levels have limited operational impact, whereas large jumps between non-adjacent levels could lead to significant misallocation of maintenance resources.

Stability Test with Stratified K-Fold Cross-Validation

To test the model's stability and generalization beyond a single data split, Stratified 5-Fold cross-validation was performed on the sigmoid kernel SVM (Tsamardinos et al., 2018). The accuracy of each fold is presented in Table 4. Accuracy ranged from 71.43% on Fold 3 to 97.14% on Fold 5, with an average of 81.71% and a standard deviation of 9.66%.

Table 4. Akurasi SVM Kernel Sigmoid pada Stratified 5-Fold Cross-Validation

Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Average ± SD
0,8857	0,7714	0,7143	0,7429	0,9714	0,8171 ± 0,0966

The average accuracy of 81.71% in cross-validation was lower than the accuracy in single test data of 90.57%. This difference of about 8.9 percentage points indicates that the results on the single test data may be slightly optimistic, and a more realistic estimate of the model's generalized performance is in the range of 71% to 91% considering the range ± one standard deviation. Performance estimates derived from a single data partition may be subject to substantial variability, whereas cross-validation generally yields more stable and reliable estimates of predictive performance (Tsamardinos et al., 2018). The relatively high variability (SD = 9.66%) suggests that the sigmoid kernel SVM in this dataset is quite sensitive to data partitioning, which can be triggered by two main factors: the relatively small size of the dataset and the unbalanced distribution of the class. When hard-to-classify samples are concentrated in a particular fold, the selection of the support vector and the position of the hyperplane can change significantly so that performance decreases in that fold (Valkenborg et al., 2023).

Fold 5 which provides an accuracy of 97.14% indicates that on a given data distribution, the SVM sigmoid kernel is capable of achieving very high performance. In contrast, the Fold 3, which only reaches 71.43%, gives an idea of the lower limit of the model's performance when faced with more challenging data sharing. These findings reinforce the importance of reporting model performance through cross-validation and not just through one specific data sharing (Han et al., 2022).

Research Limitations

Several limitations need to be conveyed. First, the size of the dataset is still relatively small (175 samples) and limited to hotels in the Yogyakarta area, so the generalization of the findings to the wider hotel population or to other types of buildings needs to be further tested. Second, the distribution of classes is unbalanced with Class 2 as the majority (about 40% of the test data) and Class 4 as the minority (about 11% of the test data), which has the potential to affect the stability of the model especially on cross-validation. One of the most widely used techniques for addressing class imbalance is the Synthetic Minority Over-sampling Technique (SMOTE), which generates synthetic samples for minority classes to improve class distribution during model training (Chawla et al., 2002). Third, SVM hyperparameter tuning was conducted within a limited search space, involving only three kernel functions and a single regularization parameter value of $C = 1.0$. A more comprehensive grid search involving combinations of C, *gamma*, and other kernel-specific

parameters may improve model performance (Douglass, 2020). Fourth, this study has not directly compared the performance of SVM-PCA with the ANN-PCA approach that has been studied by Pratama et al. (2026); Experimental comparisons of the same data will provide recommendations of the best method more convincingly.

CONCLUSION

This study demonstrated that the integration of Support Vector Machine (SVM) and Principal Component Analysis (PCA) can be effectively used to classify hotel building maintenance levels in Yogyakarta into five categories. PCA successfully reduced the original variables into nine principal components while preserving 93.41% of the total data variance, indicating efficient dimensionality reduction without significant information loss. Among the tested kernels, the sigmoid kernel achieved the best performance with high accuracy, precision, recall, and F1-score, outperforming the RBF and polynomial kernels. The model also showed reasonably stable performance based on cross-validation results, although some sensitivity to data partitioning remained. Overall, the proposed PCA-SVM model provides a lightweight, interpretable, and practical solution for automated hotel building maintenance classification, making it suitable for facility management applications.

For future research, it is recommended to expand the dataset by including hotels from different regions and accommodation types to improve model generalization. Further improvements can be achieved by applying systematic hyperparameter optimization and implementing class-balancing techniques to enhance prediction performance, especially for minority classes. Additionally, future studies should compare the PCA-SVM model with other machine-learning approaches under consistent experimental settings and explore the use of ensemble and ordinal classification methods to better capture the sequential nature of maintenance levels and improve overall classification accuracy.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge LPPMI ITNY (Institut Teknologi Nasional Yogyakarta) for providing financial support through the ITNY Research Program for the 2026 Fiscal Year, which enabled the completion of this study.

REFERENCES

Atkin, B., & Bildsten, L. (2017). A future for facility management. *Construction Innovation*, *17*(2), 116–124. https://doi.org/10.1108/CI-11-2016-0059

Ben-Hur, A., & Weston, J. (2010). *A User's Guide to Support Vector Machines* (pp. 223–239). https://doi.org/10.1007/978-1-60327-241-4_13

Burges, C. J. C. (1998). A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery*, *2*(2), 121–167. https://doi.org/10.1023/A:1009715923555

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. https://doi.org/10.1613/jair.953

Douglass, M. J. J. (2020). Book Review: Hands-on Machine Learning with Scikit-Learn, Keras, and Tensorflow, 2nd edition by Aurélien Géron. *Physical and Engineering Sciences in Medicine*, *43*(3), 1135–1136. https://doi.org/10.1007/s13246-020-00913-z

Han, J., Pei, J., & Tong, H. (2022). *Data Mining: Concepts and Techniques*. Morgan Kaufmann.

Jolliffe, I. T., & Cadima, J. (2016). Principal component analysis: a review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *374*(2065), 20150202. https://doi.org/10.1098/rsta.2015.0202

Luts, J., Ojeda, F., Van de Plas, R., De Moor, B., Van Huffel, S., & Suykens, J. A. K. (2010). A tutorial on support vector machine-based methods for classification problems in chemometrics. *Analytica Chimica Acta*, *665*(2), 129–145. https://doi.org/10.1016/j.aca.2010.03.030

Pisner, D. A., & Schnyer, D. M. (2020). Support vector machine. In *Machine Learning* (pp. 101–121). Elsevier. https://doi.org/10.1016/B978-0-12-815739-8.00006-7

Pratama, B. G., Sari, S. N., & Maulana, R. (2025). Implementasi Artificial Neural Network Untuk Prediksi Pelaksanaan Pemeliharaan Hotel di Yogyakarta. *STORAGE: Jurnal Ilmiah Teknik Dan Ilmu Komputer*, *4*(4), 277–286. https://doi.org/10.55123/storage.v4i4.6022

Pratama, B. G., Sari, S. N., & Maulana, R. (2026). Evaluating the effectiveness of PCA-based indicator reduction on ANN performance for hotel maintenance classification in Yogyakarta. *E3S Web of Conferences*, *688*, 03006. https://doi.org/10.1051/e3sconf/202668803006

Rajamani, S. K., & Iyer, R. S. (2023). *Machine Learning-Based Mobile Applications Using Python and Scikit-Learn* (pp. 282–306). https://doi.org/10.4018/978-1-6684-8582-8.ch016

Raschka, S., Patterson, J., & Nolet, C. (2020). Machine Learning in Python: Main Developments and Technology Trends in Data Science, Machine Learning, and Artificial Intelligence. *Information*, *11*(4), 193. https://doi.org/10.3390/info11040193

Ruppert, D. (2004). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. *Journal of the American Statistical Association*, *99*(466), 567–567. https://doi.org/10.1198/jasa.2004.s339

Sari, S. N., Maulana, R., Hermawan, A., & Ardian, O. H. (2024). Penilaian Pelaksanaan Pemeliharaan dan Perawatan Bangunan Hotel di Sleman Yogyakarta. *G-Tech: Jurnal Teknologi Terapan*, *8*(3), 1769–1778. https://doi.org/10.33379/gtech.v8i3.4556

Tran, M., Panchal, S., Chauhan, V., Brahmbhatt, N., Mevawalla, A., Fraser, R., & Fowler, M. (2022). <scp>Python-based scikit-learn</scp> machine learning models for thermal and electrical performance prediction of <scp>high-capacity</scp> lithium-ion battery. *International Journal of Energy Research*, *46*(2), 786–794. https://doi.org/10.1002/er.7202

Tsamardinos, I., Greasidou, E., & Borboudakis, G. (2018). Bootstrapping the out-of-sample predictions for efficient and accurate cross-validation. *Machine Learning*, *107*(12), 1895–1922. https://doi.org/10.1007/s10994-018-5714-4

Tsiamis, A., Ziemann, I., Matni, N., & Pappas, G. J. (2023). Statistical Learning Theory for Control: A Finite-Sample Perspective. *IEEE Control Systems*, *43*(6), 67–97. https://doi.org/10.1109/MCS.2023.3310345

Valkenborg, D., Rousseau, A.-J., Geubbelmans, M., & Burzykowski, T. (2023). Support vector machines. *American Journal of Orthodontics and Dentofacial Orthopedics*, *164*(5), 754–757. https://doi.org/10.1016/j.ajodo.2023.08.003