Jurnal Teknologi dan Manajemen Industri Terapan (JTMIT) Vol. 5, No. 3, September 2026 pp. 1683 - 1690
P-ISSN: 2829-0232 E-ISSN: 2829-0038

Predictive Maintenance of Government Official Vehicles Using Linear Regression and Decision Tree Machine Learning Models

Esa Fauzi[1], Hervin Avionilova[2]
[1,2)] Program Studi Teknik Informatika, Fakultas Teknologi, Universitas Widyatama
Jalan Cikutra No. 204A, Sukapada, Kecamatan Cibeunying Kidul, Kota Bandung, Jawa Barat 40125
Email: esa.fauzi@widyatama.ac.id , avionilova.hervin@widyatama.ac.id

ABSTRAK

Penerapan teknologi machine learning dalam predictive maintenance kendaraan dinas bertujuan meningkatkan produktivitas dalam memprediksi biaya dan jadwal pemeliharaan, serta penggantian suku cadang kendaraan. Penelitian ini menggunakan machine learning untuk menganalisis data historis pemeliharaan kendaraan dinas dan memprediksi kebutuhan pemeliharaan di masa mendatang. Model machine learning yang digunakan untuk memprediksi tanggal dan biaya servis adalah regresi linier, sedangkan decision tree digunakan untuk prediksi suku cadang. Tahapan penelitian meliputi pengumpulan data, prapemrosesan data, pelatihan model, evaluasi model, dan implementasi model. Model yang dihasilkan diharapkan dapat memberikan prediksi yang akurat mengenai biaya dan jadwal pemeliharaan, serta penggantian suku cadang kendaraan, sehingga membantu mengoptimalkan pengelolaan kendaraan dinas. Hasil penelitian ini diharapkan memberikan kontribusi yang signifikan dalam mengoptimalkan manajemen pemeliharaan dan alokasi anggaran pemeliharaan kendaraan dinas.

Kata kunci: pemeliharaan prediktif, kendaraan dinas, pembelajaran mesin, regresi linier, pohon keputusan

ABSTRACT

The application of machine learning technology in predictive maintenance for service vehicles aims to enhance productivity in predicting maintenance costs and schedules, as well as the replacement of vehicle spare parts. This research employs machine learning to analyze historical maintenance data of service vehicles and predict future maintenance needs. The machine learning models used for predicting service dates and costs are linear regression, while decision trees are used for spare part predictions. The research stages include data collection, data preprocessing, model training, model evaluation, and model implementation. The resulting model is expected to provide accurate predictions regarding maintenance costs and schedules, as well as vehicle spare parts replacement, thereby helping to optimize the management of service vehicles. The results of this research are expected to make a significant contribution to optimizing the maintenance management and budget allocation for service vehicle maintenance.

Keywords: *predictive maintenance, official vehicles, machine learning, linear regression, decision tree*

Introduction

The poor condition of official vehicles that are unfit for use can disrupt public services and even lead to accidents that endanger the lives of drivers and passengers. Official vehicles, which are routinely used to support various government operations, require timely and appropriate maintenance to function optimally. Currently, the repair of official vehicles is carried out reactively - meaning repairs are only performed after a vehicle has already experienced damage. The budgeting for official vehicle maintenance has not been based on scientific studies, resulting in suboptimal maintenance budgets [1] [2] [3] [4]. This leads to uncertainty in budget planning and maintenance execution, which has the potential to disrupt the operation of official vehicles.

Previous studies have explored various approaches to implementing predictive maintenance using machine learning technology. For example, a study conducted by Sharma and Kalra [13] in the journal "Predictive Maintenance for Commercial Vehicles Tyres Using Machine Learning" demonstrated that machine learning-based regression models can effectively predict the remaining lifespan of commercial vehicle tires. Additionally, research by Sengupta et al. [16] in "Predictive Maintenance of Armoured Vehicles using Machine Learning Approaches" showed that machine learning models can provide accurate predictions for the maintenance needs of armored vehicles. Both studies highlight the importance of utilizing historical data and machine learning algorithms to improve vehicle maintenance predictions [5].

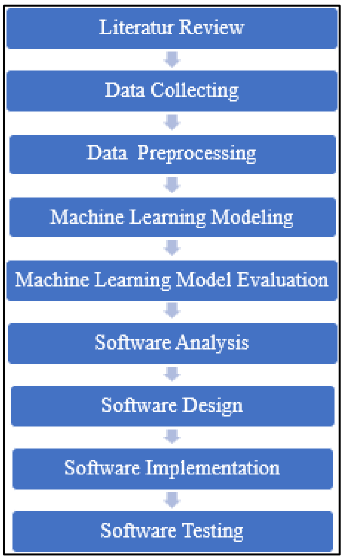
Predictive maintenance for connected vehicle fleets has recently drawn growing attention in Industry 4.0 and industrial IoT settings, with hybrid deep learning and unsupervised anomaly-detection approaches applied to fleet management scenarios [6], [7]. Cost, however, has received far less attention than failure detection in this literature. Where cost has been modelled, it has typically been for large infrastructure assets or production equipment rather than vehicle fleets [8], and studies that do address maintenance expenditure show that accurate cost forecasting supports budget control and more efficient resource allocation [9]. While previous work has applied machine learning to official vehicles, it has been limited to classifying vehicle condition rather than forecasting the operational parameters needed for budget planning [6]. This research therefore aims to develop a predictive maintenance model that simultaneously forecasts maintenance schedules, costs, and spare part replacements for official vehicles. Unlike prior studies that focused on commercial or military vehicles, or that predicted only failure conditions, this research targets government official vehicles and explicitly incorporates service cost prediction, which directly supports maintenance budget planning. In practical terms, the resulting model is intended to give fleet administrators three concrete capabilities: a defensible basis for preparing annual maintenance budget estimates rather than relying on historical averages, advance visibility of upcoming service events so that vehicle availability can be planned around them, and early identification of spare part requirements to support procurement scheduling. Collectively, these shift official vehicle maintenance from a reactive posture, in which repairs follow breakdowns, towards a predictive one in which interventions are planned before failure occurs.

Taken together, these studies share a common limitation relevant to the present work: each predicts a physical condition or a remaining-life estimate for a component, whether tyre wear on commercial vehicles, sensor-derived failure states on armoured vehicles, or engine-component condition. None forecasts the operational and financial parameters that a government fleet administrator actually needs to prepare a maintenance budget, namely when the next service will fall due, how much it will cost, and which spare parts will be required. This study addresses that gap directly by predicting all three simultaneously for government official vehicles.

Research Method

The research approach used is a quantitative approach to develop and test the predictive maintenance model for official vehicles using machine learning, while the waterfall method is applied for software development. The research stages are illustrated in Gambar 1.

Each stage feeds a defined output into the next. The literature review establishes the methodological basis for algorithm and metric selection. Data collection yields the raw maintenance record set from BPKAD, which becomes the input for preprocessing. Preprocessing produces a cleaned, encoded, and standardised dataset in array form, which is the direct input to model training. Model training produces three serialised model artefacts, one for spare part classification and two for the regression targets. Model evaluation takes these artefacts together with the held-out test partition and produces the performance metrics reported in this paper, which determine whether a model is accepted for deployment. Finally, system implementation embeds the accepted artefacts in the vehicle maintenance application, whose output is the predictive maintenance page presented to end users. Because each stage depends on the validated output of the preceding one, a deficiency detected at the evaluation stage requires returning to preprocessing or training rather than proceeding to implementation.



Gambar 1. Research Stages

Quantitative is a systematic scientific investigation of components, phenomena, and the causality of their relationships. It is defined as a systematic examination of phenomena by collecting measurable data through statistical, mathematical, or computational techniques. Quantitative research is largely conducted using statistical methods to collect quantitative data from research studies. In this research method, researchers and statisticians utilize mathematical frameworks and theories related to the quantity in question [7].

The Waterfall Method is a systematic and structured software development model, where each stage must be completed sequentially before proceeding to the next phase. The stages in this method include requirement analysis, system design, implementation, testing, deployment, and maintenance. Pressman explains that this model is suitable for projects with well-defined and stable requirements from the beginning, minimizing the likelihood of changes during the development process. Although this method has weaknesses in handling changes that occur mid-process, the Waterfall model remains an effective approach for projects with a well-defined scope and limited development time [18]. Simulation studies of the Waterfall lifecycle confirm that its sequential structure performs predictably when requirements are stable, reinforcing its suitability for this research [10].

Result and Discussion

Data Collecting

The data source for this research comes from vehicle maintenance records at the Regional Financial and Asset Management Agency (BPKAD) of West Java Province. The dataset covers the vehicle maintenance period from 2020 to 2024, consisting of 3,832 records. The dataset structure is shown in Tabel 1.

Tabel 1. Dataset Structure

Column	Data Type	Description
no	Integer	Sequential data number
no_polisi	String	Police number
merk	String	Brand vehicle
transmisi	String	Transmision type (manual/automatic)
tahun_kendaraan	Integer	Year of vehicle production
tanggal_service	Date	Date of last vehicle service
tanggal_bayar	Date	Date of service payment
biaya_service	Float	Cost incurred for service
bengkel	String	Name of the service workshop
kategori_kendaraan	String	Vehicle category (two-wheel, four-wheel, official four-wheel)
spare_part	String	Spare parts replaced during service

Data Preprocessing

Raw maintenance records cannot be fed directly into a machine learning model. Incomplete entries and attributes measured on different scales must first be resolved, since both affect the quality of the resulting predictions. Missing values in categorical attributes were handled before training, as unresolved gaps degrade classifier accuracy [11]. Numerical attributes were standardised so that features with larger numeric ranges, such as service cost, do not dominate features with smaller ranges, such as vehicle production year [12].

The preprocessing stages carried out were: handling missing values, converting date columns to numeric format, identifying numerical and categorical columns, standardising numerical columns, encoding categorical columns, transforming the dataset, converting the data to array format, and saving the processed dataset. Categorical columns were encoded prior to model training, since the choice of encoding technique (e.g., one-hot, label, or frequency encoding) has a notable influence on how well the transformed data represents the underlying categories [13].

Machine Learning Modeling

Spare part replacement is a multi-class classification problem: for a given service event, the model must select which part is likely to be replaced from a set of possible categories. Decision trees are well suited to this task because they handle categorical predictors directly and produce rules that maintenance staff can inspect, and they have been applied to spare part decision-making and purchase-quantity prediction in manufacturing settings [14], [15]. The main model used for spare part prediction is therefore the Decision Tree Classifier from sklearn.tree [16]. This model is trained with data that has undergone an encoding process. The dataset is split into a training set (80%) and a testing set (20%) using train_test_split. The trained model is saved in the file model_spare_part.pkl.

The choice of a single decision tree over ensemble alternatives such as Random Forest, Gradient Boosting, or XGBoost was deliberate and reflects the operational context of this study rather than a search for maximum predictive accuracy. Three considerations governed the selection. First, interpretability: a single tree yields an explicit and auditable decision path, which matters in a public-sector setting where a predicted spare part requirement may need to be justified to procurement and audit functions; ensemble methods aggregate many trees and do not expose an equivalent traceable rule. Second, the nature of the predictors: the dataset consists largely of low-cardinality categorical attributes such as vehicle brand, transmission type, and vehicle category, which a decision tree partitions directly without requiring the additional variance reduction that ensembles are designed to provide. Third, deployment cost: a single tree trains and infers quickly and produces a compact serialised artefact, which suits integration into an existing web-based maintenance application maintained by a small technical team. The same reasoning applies to the use of Linear Regression for the two regression targets, where a transparent coefficient-based relationship between vehicle attributes and service date or cost is more readily explained to budget planners than an ensemble output. Comparative benchmarking of these models against ensemble and boosting alternatives was not performed in this study and is identified as a direction for further work.

Linear Regression is used for two predictions [17] [18]:

- Prediction of tanggal_service: Predicting when the vehicle will be serviced next.
- Prediction of biaya_service: Predicting the cost to be incurred for vehicle servicing.

The data is split into a training set (80%) and a testing set (20%) using train_test_split(). An 80:20 training-testing split was adopted, consistent with common practice for balancing sufficient training data against a testing set large enough to evaluate generalisation performance [19]. The Linear Regression model is trained using fit(). The trained model is saved in the file model_tanggal_service.pkl and model_biaya_service.pkl.

Machine Learning Model Evaluation

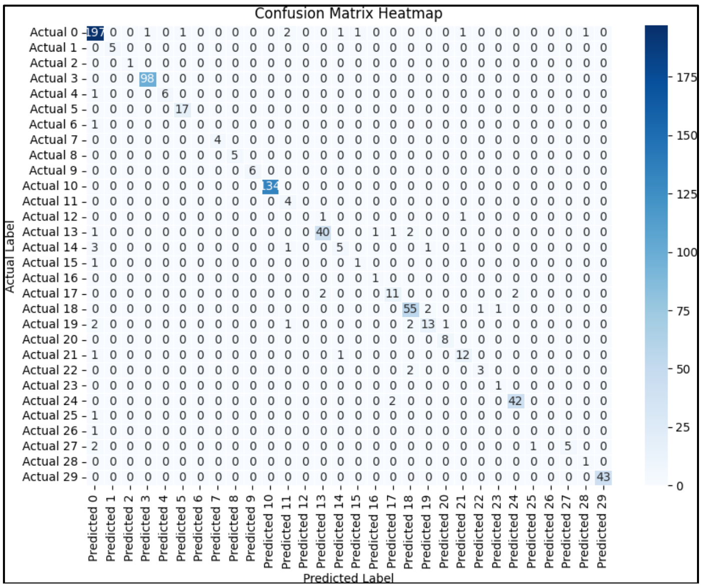
Because the spare part prediction task is a multi-class classification problem, accuracy alone is not sufficient to describe model performance; class-level metrics are required to reveal how the model behaves on labels with few samples [20]. The evaluation of the machine learning model for spare part prediction therefore uses accuracy, confusion matrix, precision, recall, and F1-score. Below are the results for each evaluation metric:

1. Accuracy.

The evaluation of the machine learning model for spare part prediction using accuracy_score resulted in a value of 0.9361147327249022, meaning that the model has an accuracy rate of 93% in predicting the spare parts that need to be replaced.

2. Confusion Matrix.

The evaluation of the machine learning model for spare part prediction uses a confusion matrix in the form of a heatmap, which visualizes the comparison between actual labels and predicted labels. The confusion matrix heatmap visualization is shown in Gambar 2.



Gambar 2. Confusion Matrix Heatmap

The conclusion from the confusion matrix is that most predictions are located on the main diagonal, indicating that the model can correctly classify the majority of labels.

3. Precision, Recall, and F-1 Score.

The table of precision, recall, and F1-score values is shown in the following Tabel 2.

Tabel 2. Precision, Recall, and F1-Score

Label	Precision	Recall	F1-Score
0	0.933649	0.960976	0.947115
1	1.000000	1.000000	1.000000
2	1.000000	1.000000	1.000000
3	0.989899	1.000000	0.994924
4	1.000000	0.857143	0.923077
5	0.944444	1.000000	0.971429
6	0.000000	0.000000	0.000000
7	1.000000	1.000000	1.000000

8	1.000000	1.000000	1.000000
9	1.000000	1.000000	1.000000
10	1.000000	1.000000	1.000000
11	0.500000	1.000000	0.666667
12	0.000000	0.000000	0.000000
13	0.930233	0.888889	0.909091
14	0.714286	0.454545	0.555556
15	0.500000	0.500000	0.500000
16	0.500000	1.000000	0.666667
17	0.785714	0.733333	0.758621
18	0.901639	0.932203	0.916667
19	0.812500	0.684211	0.742857
20	0.888889	1.000000	0.941176
21	0.800000	0.857143	0.827586
22	0.750000	0.600000	0.666667
23	0.500000	1.000000	0.666667
24	0.954545	0.954545	0.954545
25	0.000000	0.000000	0.000000
26	0.000000	0.000000	0.000000
27	1.000000	0.625000	0.769231
28	0.500000	1.000000	0.666667
29	1.000000	1.000000	1.000000

The overall conclusion is that the model achieves high precision and recall for the majority of the thirty spare part labels, and the F1-Score indicates balanced performance across these classes. Performance is not uniform, however. Four labels (6, 12, 25, and 26) record a precision, recall, and F1-Score of zero, meaning the model failed to correctly identify any instance of these classes, while a further seven labels fall below an F1-Score of 0.7. This pattern is consistent with class imbalance in the training data: spare parts that are replaced only rarely provide too few examples for the classifier to learn a reliable decision boundary. The 93% overall accuracy therefore reflects strong performance on frequently replaced spare parts, and should not be read as uniform reliability across every spare part category.

Several strategies could address this imbalance in subsequent work. Resampling the training partition, either by oversampling the minority spare part classes or by synthetic minority oversampling, would give the classifier more examples from which to learn a boundary for rarely replaced parts. Class weighting offers an alternative that does not alter the data distribution, applying a higher misclassification penalty to minority classes during training so that the loss function no longer favours the dominant categories. A third option is to consolidate the label space: spare parts that are functionally related and individually rare could be merged into broader categories, trading granularity for classes with enough observations to be learned reliably. Given that the current label set contains thirty categories drawn from a dataset of 3,832 records, category consolidation may be the most practical of the three in this setting, since it addresses the root cause of insufficient per-class observations rather than compensating for it during training. These strategies were not applied in the present study and represent a clear avenue for improvement.

For the two regression tasks, model performance is measured using MSE, RMSE, MAE, and R^2, a combination commonly adopted when comparing regression models because each metric captures a different aspect of error: MSE and RMSE penalise large deviations, MAE reports the average error in the original units, and R^2 indicates how much of the variability in the data the model explains [21]. Evaluation of the machine learning model for service date prediction uses these four metrics, and the results are shown in Tabel 3.

Tabel 3. Evaluation Metrics Service Date

No	Metrics	Result	Conclusion
1	MSE	3.6896636781 987992e-06	The difference between the model's predictions and the actual values is very small, meaning the Model can predict the service date with high accuracy.
2	RMSE	0.0019208497 281668858	The average prediction error of the model is close to zero days, which means the model is very accurate in predicting the service date.
3	MAE	0.0006744450 934286468	The average absolute difference between the predictions and actual values is only around 0.00067 days, which means the model is highly accurate in predicting the service date.
4	R^2	0.999999999 9848894	The R^2 value, which is close to 1, means the model can explain almost all of the variability in the service date data very well.

Evaluation of the machine learning model for service cost prediction using MSE, RMSE, MAE, and R2. Below are the results for each evaluation metric shown in Tabel 4. The R^2 values obtained for both regression targets, approximately 0.99999999999 in each case, warrant caution rather than confidence. Error values this close to zero are atypical for real-world maintenance data, where genuine variability in service intervals and costs would be expected to leave a visible residual. The most likely explanation is target leakage: if a predictor available at training time is itself derived from, or deterministically related to, the target variable, the model can reproduce the target almost exactly without having learned any

Jurnal Teknologi dan Manajemen Industri Terapan (JTMIT) Vol. 5, No. 3, September 2026 pp. 1683 - 1690
P-ISSN: 2829-0232 E-ISSN: 2829-0038

generalisable relationship. In this dataset the presence of both tanggal_service and tanggal_bayar, which are closely coupled in practice, is a plausible source of such leakage for the service date target. A second possibility is overfitting to a single hold-out partition, which a single 80:20 split cannot rule out. Three checks are therefore required before these figures can be treated as evidence of predictive capability: an audit of the feature set to confirm that no predictor encodes the target, k-fold cross-validation to establish whether performance is stable across different partitions, and validation on maintenance records from a period or agency outside the training data. Until these are carried out, the reported R^2 values should be read as an indication that the evaluation setup requires verification, not as a measure of the model's generalisation ability.

Tabel 4. Evaluation Metrics Service Cost

No	Metrics	Result	Conclusion
1	MSE	72.441419 11226842	The mean squared error in predicting service costs is approximately 73 cost units (rupiah). Th value is relatively small because service costs range from thousands to millions (rupiah Therefore, an MSE of 73 indicates that the model is highly accurate in predicting the service cos
2	RMSE	8.5112524 99618868	The average prediction error in the original units is approximately 8.51 cost units (rupiah meaning that most predictions fall within ±8.51 of the actual value.
3	MAE	2.9113347 56278066	The average absolute difference between predictions and actual values is only about 2.91 cc units (rupiah), indicating that the model is highly accurate in predicting service costs.
4	R^2	0.9999999 999901579	The R^2 value close to 1 means that the model can explain almost all the variability in servi cost data very well.

Implementation Predictive Maintenance Page

The Predictive Maintenance page is a page that displays the results of vehicle maintenance predictions, including the estimated service date, estimated service cost, and estimated spare parts. The estimated service date and service cost use a linear regression machine learning model, while the estimated spare parts use a decision tree machine learning model. The machine learning models that have been created are integrated with the vehicle maintenance application, following standard practice for deploying trained models into production systems so that predictions can be served to end users [22]. The predictive maintenance page follows dashboard design practices intended to support maintenance planners and operators with clear, actionable insights drawn from model outputs [23].

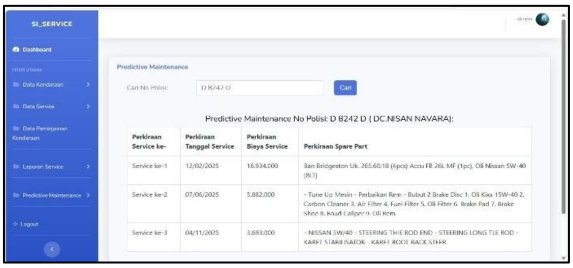
On the predictive maintenance page, in the 'Search by License Plate' column, the user enters the vehicle's license plate number in the 'Search by License Plate' field, then clicks search. This will display a table of predictive maintenance results, including the 'Estimated Service No.' column (which shows the order of the predicted service), the 'Estimated Service Date' column (which displays the predicted next service date, set for the year 2025), the 'Estimated Service Cost' column (which displays the estimated service cost), and the 'Estimated Spare Parts' column (which displays the predicted spare parts that need to be replaced). A screenshot of the predictive maintenance page can be seen in the Gambar 3.

In operational use, the page supports two distinct decision processes. At the operational level, a fleet administrator can query an individual vehicle before assigning it to a duty, using the predicted service date to avoid dispatching a vehicle that is approaching a service interval, and the predicted spare part requirement to confirm with the workshop that the necessary component is available before the vehicle is sent in. This shortens the interval between a service becoming due and the work being completed. At the planning level, aggregating the predicted service costs across the fleet for a forthcoming fiscal period produces a bottom-up maintenance budget estimate that is traceable to individual vehicles, in contrast to the current practice of budgeting from historical totals. The predicted spare part categories can likewise be aggregated to inform procurement volumes ahead of the period, reducing the incidence of parts being ordered reactively after a breakdown. Realising these benefits depends on the accuracy caveats discussed in the evaluation section being resolved first, since a budget built on unvalidated cost predictions would carry the model's error into the financial plan.

Black Box Testing

The developed system is tested to ensure that all features function as planned. System testing uses the black-box testing method, a software testing approach that examines the application's functionality without considering the system's internal structure or source code, focusing instead on inputs and outputs to verify that the application behaves according to specifications and user requirements [24].

The results of testing all black box testing scenarios on the Login page, Dashboard, Vehicle Data, Service Data, Vehicle Borrowing Data, Service Reports, and Predictive Maintenance pages show that all the scenarios executed produced outputs as expected.



Gambar 3. Predictive Maintenance Page

Conclusion

Based on the research conducted, the following conclusions can be drawn: A machine learning model has been developed using Decision Tree for spare part prediction and Linear Regression for predicting service dates and service costs. The evaluation of the machine learning model for spare part prediction shows good results with an accuracy of 93%. The evaluation of the machine learning model for service cost prediction produced an MSE of 72.44141911226842, an RMSE of 8.511252499618868, an MAE of 2.911334756278066, and an R^2 value of 0.9999999999901579. These results indicate that the service cost prediction model is highly accurate, with a very low error rate.

The machine learning model for service date prediction produced an MSE of 3.6896636781987992e-06, an RMSE of 0.0019208497281668858, an MAE of 0.0006744450934286468, and an R^2 value of 0.9999999999848894. These results indicate that the service date prediction model has excellent performance with an extremely low error rate. Based on this, the machine learning model for service date prediction can accurately determine when a vehicle needs maintenance before a failure occurs, shifting maintenance from a reactive system (waiting for damage to occur) to a proactive system (preventing damage before it happens).

To improve the performance of the government vehicle predictive maintenance system, it is recommended that the developed machine learning model be continuously updated with the latest historical data to maintain high accuracy, since predictive models are susceptible to performance degradation over time as the underlying data distribution shifts (data drift) or the relationship between features and the target changes (concept drift) [25]. Additionally, incorporating other features such as road conditions, vehicle usage patterns, and external factors like weather can be a strategic step in optimizing predictive maintenance for government vehicles.

This study has several limitations that bound the interpretation of its results. The dataset is drawn from a single agency over a five-year period, so the models have not been tested against maintenance practices, workshop pricing, or vehicle compositions outside that context. The feature set is confined to attributes recorded in existing administrative maintenance records and therefore omits variables that plausibly drive maintenance need, including cumulative mileage, vehicle usage intensity, driver behaviour, road conditions, and environmental exposure. Spare part prediction is unreliable for rarely replaced categories, as the class-level metrics show, and the regression results require the leakage and cross-validation checks identified earlier before they can be relied upon. Further work should therefore prioritise, in order: verifying the regression evaluation setup, enriching the feature set with usage and environmental variables, addressing class imbalance in the spare part labels, benchmarking the selected algorithms against ensemble alternatives, and establishing a periodic retraining schedule so that the deployed models remain aligned with the current data distribution as vehicle composition and maintenance costs change over time.

References

[1] A. Shaf, D. Retnoningsih, H. Khusnuliawati, P. Studi Informatika, F. Sains, and T. dan Kesehatan, "Sistem Pengelolaan Kendaraan Dinas Di Pemerintah Kota Salatiga," *Jurnal Gaung Informatika*, vol. 13, no. 2, Jul. 2020, doi: 10.47942/GI.V13I2.539.

[2] D. Jaelani, D. Hadi Kushendar, and S. Tinggi Ilmu Administrasi, "Implementasi Kebijakan Pemeliharaan Kendaraan Dinas Operasional Pada Biro Umum Sekretariat Daerah Provinsi Jawa Barat," *Moderat : Jurnal Ilmiah Ilmu Pemerintahan*, vol. 9, no. 4, pp. 705–720, Nov. 2023, doi: 10.25157/MODERAT.V9I4.3495.

[3] F. Khaerunnisa and C. W. Hoerudin, "Pengelolaan Kendaraan Dinas Dalam Mewujudkan Tertib Administrasi Pada Badan Pengelolaan Keuangan dan Aset Daerah Kabupaten Bogor," *Ministrate: Jurnal Birokrasi dan Pemerintahan Daerah*, vol. 4, no. 2, pp. 72–82, Aug. 2022, doi: 10.15575/JBPD.V4I2.19420.

[4] N. Hasdyna and A. Arafat, "Implementasi Sistem Informasi Monitoring Kendaraan Dinas Terintegrasi Pada Bank Indonesia Lhokseumawe," *Informatics Journal*, vol. 5, no. 2, 2020.

[5] M. Paolanti, L. Romeo, A. Felicetti, A. Mancini, E. Frontoni, and J. Loncarski, "Machine Learning approach for Predictive Maintenance in Industry 4.0," *2018 14th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications, MESA 2018*, Aug. 2018, doi: 10.1109/MESA.2018.8449150.

[6] A. Chaudhuri and S. K. Ghosh, "Predictive maintenance of vehicle fleets through hybrid deep learning-based ensemble methods for industrial IoT datasets," *Log. J. IGPL*, vol. 32, no. 4, pp. 671–687, Jul. 2024, doi: 10.1093/JIGPAL/JZAE017.

[7] A. Giannoulidis and A. Gounaris, "A context-aware unsupervised predictive maintenance solution for fleet management," *Journal of Intelligent Information Systems 2022 60:2*, vol. 60, no. 2, pp. 521–547, Sep. 2022, doi: 10.1007/S10844-022-00744-2.

[8] C. Jia, J. An, S. Ma, and X. Dai, "Machine learning approaches for predicting preventive maintenance costs of expressways in Xinjiang," *PLoS One*, vol. 21, no. 6, p. e0349595, Jun. 2026, doi: 10.1371/JOURNAL.PONE.0349595.

[9] Y. Lv, X. Guo, Q. Zhou, L. Qian, and J. Liu, "Predictive maintenance decision-making for variable faults with non-equivalent costs of fault severities," *Advanced Engineering Informatics*, vol. 56, p. 102011, Apr. 2023, doi: 10.1016/J.AEI.2023.102011.

[10] A. Saravanos and M. X. Curinga, "Simulating the Software Development Lifecycle: The Waterfall Model," *Applied System Innovation 2023, Vol. 6, Page 108*, vol. 6, no. 6, p. 108, Nov. 2023, doi: 10.3390/ASI6060108.

[11] M. Ishaq, S. Zahir, L. Iftikhar, M. F. Bulbul, S. Rho, and M. Y. Lee, "Machine Learning Based Missing Data Imputation in Categorical Datasets," *IEEE Access*, vol. 12, pp. 88332–88344, 2024, doi: 10.1109/ACCESS.2024.3411817.

[12] D. U. Ozsahin, M. Taiwo Mustapha, A. S. Mubarak, Z. Said Ameen, and B. Uzun, "Impact of feature scaling on machine learning models for the diagnosis of diabetes," *Proceedings - 2022 International Conference on Artificial Intelligence in Everything, AIE 2022*, pp. 87–94, 2022, doi: 10.1109/AIE57029.2022.00024.

[13] N. Kosaraju, S. R. Sankepally, and K. Mallikharjuna Rao, "Categorical Data: Need, Encoding, Selection of Encoding Method and Its Emergence in Machine Learning Models—A Practical Review Study on Heart Disease Prediction Dataset Using Pearson Correlation," *Lecture Notes in Networks and Systems*, vol. 551, pp. 369–382, 2023, doi: 10.1007/978-981-19-6631-6_26/SAVE-RESEARCH.

[14] Patlisan, "Optimasi Akurasi Model Decision Tree Menggunakan Random Forest Regression Untuk Prediksi Kuantitas Pembelian Barang Pada Perusahaan Manufaktur," *Jurnal SIMETRIS*, vol. 14, no. 2, 2023.

[15] X. Li, S. Li, L. Han, and Y. Qian, "Research on Enterprise Spare Parts Production Decision System Based on Decision Tree," *IEEE Information Technology and Mechatronics Engineering Conference, ITOEC*, no. 2025, pp. 1158–1163, 2025, doi: 10.1109/ITOEC63606.2025.10968355.

[16] A. Géron, "Hands-on machine learning with Scikit-Learn, Keras and TensorFlow : concepts, tools, and techniques to build intelligent systems," p. 834, 2023.

[17] M. Riza Alifi *et al.*, "Penerapan Algoritma Regresi Linier pada Prediksi Tarif Influencer Media Sosial," *Journal of Information System Research*, vol. 4, no. 1, pp. 210–218, 2022, doi: 10.47065/josh.v4i1.2361.

[18] T. Indarwati, T. Irawati, and E. Rimawati, "Penggunaan Metode Linear Regression Untuk Prediksi Penjualan Smartphone," *Jurnal Teknologi Informasi dan Komunikasi (TIKomSiN)*, vol. 6, no. 2, Jan. 2018, doi: 10.30646/TIKOMSIN.V6I2.369.

[19] M. Sivakumar, S. Parthasarathy, and T. Padmapriya, "Trade-off between training and testing ratio in machine learning for medical image processing," *PeerJ Comput. Sci.*, vol. 10, p. e2245, Sep. 2024, doi: 10.7717/PEERJ-CS.2245/SUPP-6.

[20] O. Rainio, J. Teuho, and R. Klén, "Evaluation metrics and statistical tests for machine learning," *Scientific Reports 2024 14:1*, vol. 14, no. 1, pp. 6086-, Mar. 2024, doi: 10.1038/s41598-024-56706-x.

[21] B. Sekeroglu, Y. K. Ever, K. Dimililer, and F. Al-Turjman, "Comparative Evaluation and Comprehensive Analysis of Machine Learning Models for Regression Problems," *Data Intell.*, vol. 4, no. 3, pp. 620–652, Jul. 2022, doi: 10.1162/DINT_A_00155.

[22] M. N. Chowdary, B. Sankeerth, C. K. Chowdary, and M. Gupta, "Accelerating the Machine Learning Model Deployment using MLOps," in *Journal of Physics: Conference Series*, Institute of Physics, 2022. doi: 10.1088/1742-6596/2327/1/012027.

[23] I. J. Bristy, M. Tabassum, M. I. Islam, and Md. N. Hasan, "IoT-Driven Predictive Maintenance Dashboards in Industrial Operations," *Saudi Journal of Engineering and Technology*, vol. 10, no. 09, pp. 457–466, Sep. 2025, doi: 10.36348/sjet.2025.v10i09.009.

[24] A. Fikri, H. Hozairi, and M. Muhsi, "Analisis Pengujian Sistem Informasi Mui Kabupaten Pamekasan Menggunakan Metode Blackbox Functional Testing," *Jurnal Mnemonic*, vol. 5, no. 2, pp. 158–164, Aug. 2022, doi: 10.36040/MNEMONIC.V5I2.4803.

[25] Sandeep Bharadwaj Mannapur, "Understanding Data Drift and Concept Drift in Machine Learning Systems," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 11, no. 1, pp. 318–330, Jan. 2025, doi: 10.32628/cseit25111239.