Polynomial Integrated PLS Regression for Predicting Corrosion Inhibition Efficiency of Ionic Liquids

Petrus Praditya Aswangga [1]*, Muhammad Akrom [2]**
* Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro
** Research Center for Quantum Computing and Material Informatics, Universitas Dian Nuswantoro
111202113772@mhs.dinus.ac.id [1], m.akrom@dsn.dinus.ac.id [2]

Article Info

Article history:

Received 2025-07-09
Revised 2025-07-15
Accepted 2025-07-19

Keyword:

Corrosion Inhibitor,
Ionic Liquid,
Polynomial Function,
Machine Learning,
Partial Least Squares.

ABSTRACT

Corrosion degrades and weakens metal surfaces, leading to structural failure and significant safety hazards across various sectors. Data driven machine learning offers a rapid, cost-effective alternative to the expensive and time consuming traditional experimental methods by predicting inhibitor performance computationally. This study addresses the challenge of accurately predicting corrosion inhibition efficiency (CIE) of ionic liquid compounds. Integration of a polynomial function, especially in higher degrees, inevitably grows the dimensionality and escalates multicollinearity, but it captures deeper nonlinear interactions that the original variables alone would miss. To counterbalance this curse of dimensionality, Partial Least Squares (PLS) Regression was applied after polynomial integration to project the high-dimensional variables into a smaller set of predictors. Besides PLS, Gradient Boosting Regressor (GBR) and Support Vector Regressor (SVR) models were also developed to establish baseline performance. Although these polynomial integrated models outperformed their baseline version, the Polynomial Integrated PLS outperformed their predictive performance, yielding R^2, RMSE, and MAPE of 0.73, 4.730, and 3.73%, respectively. The result of this study highlights that the integration of a polynomial function can improve the predictive performance of PLS for corrosion inhibitors.

I. Introduction

Corrosion is a chemical deterioration of a metal surface as rust, through reactions with water (H_2O) and oxygen (O_2) in the environment. This degradation compromises the integrity of the metal, impairing the functionality of machinery and structural components or causing their failure. The global cost of corrosion mitigation itself amounts to approximately US$2.5 trillion [1]. However, the consequences of the corrosion can be reduced using corrosion inhibitors.

In corrosion prevention, inhibitors serve as the first layer of defence by preventing contact between the metal surface and corrosive agents. Corrosion inhibitors include a range of compounds, among which ionic liquids are particularly promising. Ionic liquids are regarded as green and sustainable inhibitors, preferred due to their non-toxic nature [2]. Nevertheless, conventional experiments to find effective corrosion inhibitors are often time and resource-intensive [3]. In recent years, ML has been widely applied in the study of corrosion inhibitor efficacy, demonstrating its effectiveness in predicting corrosion inhibition efficiency (CIE) [4], [5]. CIE prediction plays a key role in industrial decision-making, as it directly affects material durability, maintenance schedules, and operational safety across various sectors. Furthermore, ML-based approaches help reduce the cost and time associated with conventional experiments [6].

Machine learning is a subset of artificial intelligence. It is capable of learning the relationships that exist in a dataset independently. Using the ionic liquids dataset, supervised ML models can be used to predict corrosion inhibition efficiency. In a previous study, Quadri et al., developed Multiple Linear Regression (MLR) and Multilayer Perceptron Neural Network (MPLNN) models to predict the efficacy of ionic liquids as corrosion inhibitors. MLPNN model outperformed MLR, achieving RMSE and MAPE values of 5.407 and 5.781, respectively [7]. While the model achieved a decent performance, it did not explicitly address the multicollinearity inherent in the dataset and did not explore controlled transformation to capture the nonlinear relationship. This leaves room for further improvement.

A previous study demonstrated that integrating a polynomial function into SVR outperformed the K-Nearest Neighbors (KNN) and Random Forest (RF) models [8]. Likewise, other studies have shown that the Gradient Boosting Regressor (GBR) performance increased as the integration of a polynomial function was applied [9], [10]. However, the integration of a polynomial function brought multicollinearity and higher-dimensional data. To address these complications, a modelling approach that both reduces dimensionality and handles correlated variables is required.

Partial Least Squares (PLS) is a quick, efficient, and optimal regression whenever the number of explanatory variables is high and correlated with each other [6]. PLS can handle high-dimensional datasets with numerous and correlated variables used for prediction by projecting them onto a smaller set of predictors. Therefore, the primary objective of this study is to establish a highly accurate predictive model for estimating the CIE of ionic liquid compounds using the polynomial integrated PLS model.

II. METHOD

The proposed workflow in this study consists of data collection, data preprocessing, data splitting, modelling, integrating a polynomial function, and evaluation as the final step to assess the performance of machine learning models, as depicted in Figure 1.

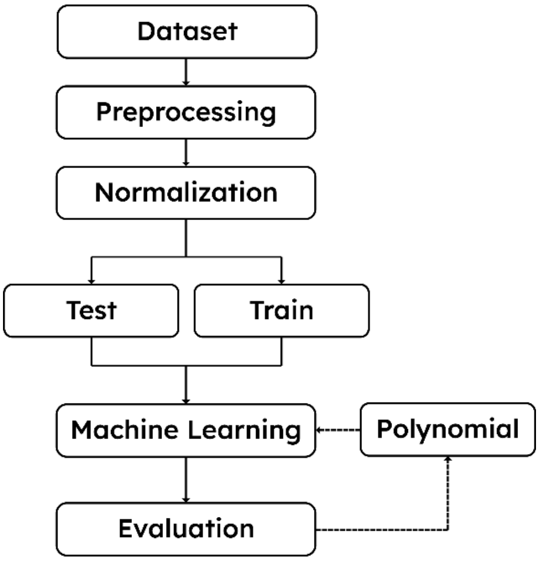

Figure 1. Proposed Workflow

A. Dataset

The dataset employed in this study is a published 30 ionic liquid compounds and 14 descriptors from the literature [7]. All descriptors were primarily derived from Density Functional Theory (DFT), except for concentration (Conc) and temperature (Temp), which were obtained from experimental data. Those collected DFT descriptors include the highest and the lowest occupied molecular orbital energy (HOMO and LUMO), the energy gap (ΔE), dipole moment (μ), ionization potential (IP), electron affinity (EA), electronegativity (χ), global hardness (η), global softness (σ), fraction of electrons transferred (ΔN), and electrophilicity index (ω). The following equation was used for ω [11]:

$$\omega = \frac{(IP + EA)^2}{8(IP + EA)} \tag{1}$$

B. Preprocessing

In the preprocessing step, data cleaning was done to ensure there were no duplicates and missing values. The Yeo-Johnson transformation was applied to HOMO, IP, and ΔN. This transformation effectively reduced the skewness of these features. Later, RobustScaler was applied to reduce the effect of outliers that exist in the dataset. RobustScaler scales the data based on the quartile range, making it less sensitive to outliers [12].

C. Feature Selection

All independent variables were subjected to Pearson and Spearman correlation tests after applying the Yeo-Johnson transformation as an approach to improve the accuracy of the model [13]. Although the transformation increased the correlation coefficient (r), all remained below 0.5, which is considered weak. However, several variables showed improved statistical significance. Variables were retained as candidates were those with statistical significance with a p-value ≤ 0.05 or borderline significance [14]. This selection was based on significance in either Pearson or Spearman test, as the detailed correlation coefficient and p-values are depicted in Table I.

Feature selection plays an integral part in developing a machine learning model [13]. Therefore, 9 variables were selected during filtering process were the temperature (Temp), the total energy (TE), the highest occupied molecular energy (HOMO), the lowest occupied molecular energy (LUMO), the ionization potential (IP), the electron affinity (EA), the electronegativity (χ), the fraction of electrons transferred (ΔN), and the electrophilicity index (ω).

D. Polynomial Integration

A polynomial function is a mathematical expression formed as a finite sum of terms in which the independent variables (x) and the target are modelled as an n-th degree polynomial in x [15]. This function extends the capacity of a model to fit more complex data patterns. Unlike simple linear models, a polynomial function represents a nonlinear relationship in the observed data, even though the parameters are linear [15]. Previous studies reported that the integration of polynomial functions enhanced the predictive performance of machine learning models [3], [8]–[10]. The integration of a polynomial function was done to capture deeper nonlinear interactions that the original variables would miss.

TABLE I
FEATURE CORRELATION

Feature	Correlation Test	Before Yeo-Johnson		After Yeo-Johnson	
		r	p-value	r	p-value
Conc	Pearson	0.074	0.698	0.108	0.571
	Spearman	0.016	0.934	—	—
Temp	Pearson	0.272	0.146	0.270	0.149
	Spearman	0.396	0.030	—	—
TE	Pearson	0.247	0.188	0.321	0.083
	Spearman	0.341	0.065	—	—
HOMO	Pearson	0.350	0.058	0.369	0.045
	Spearman	0.315	0.090	—	—
LUMO	Pearson	0.150	0.430	0.263	0.160
	Spearman	0.416	0.022	—	—
ΔE	Pearson	-0.125	0.510	-0.227	0.227
	Spearman	-0.148	0.437	—	—
μ	Pearson	-0.044	0.816	-0.163	0.391
	Spearman	-0.205	0.278	—	—
IP	Pearson	-0.350	0.058	-0.369	0.045
	Spearman	-0.416	0.090	—	—
EA	Pearson	-0.150	0.430	-0.263	0.160
	Spearman	-0.416	0.022	—	—
χ	Pearson	-0.382	0.037	-0.361	0.050
	Spearman	-0.417	0.022	—	—
η	Pearson	-0.169	0.372	-0.218	0.246
	Spearman	-0.113	0.551	—	—
σ	Pearson	-0.003	0.956	0.209	0.268
	Spearman	0.114	0.548	—	—
ΔN	Pearson	0.095	0.616	0.317	0.088
	Spearman	0.280	0.134	—	—
ω	Pearson	-0.382	0.037	-0.359	0.051
	Spearman	-0.417	0.022	—	—

E. Modelling and Evaluation

Machine learning (ML) is a subfield of artificial intelligence that can learn by itself with the given dataset. In this study, ML is used to capture the correlation that exists between the independent variables (descriptors) and the target (CIE) [16]. The dataset was split into partitions, with the optimum ratio of 85:15 for training and testing, giving 25 training samples and 5 testing samples, determined through trial and error. This split provides a balance between model learning and unbiased evaluation. Given the relatively small sample size and the high variance of the target variable, cross-validation was intentionally not applied, as preliminary experiments showed it produced unstable and unreliable metrics [17], [18]. Instead, a single random split was used to preserve the structure of the dataset and ensure consistent evaluation.

To evaluate the effectiveness of the polynomial integration on PLS, we also implemented and compared it with GBR and SVR models. Both GBR and SVR were developed as they have shown improved predictive performance after the integration of a polynomial function, specifically in predicting CIE. As part of this process, the degree of the polynomial function was determined empirically through trial and error according to the highest R^2 value, as this metric directly reflects the proportion of variance in the target.

PLS model, in contrast, was specifically addressed to handle numerous and correlated variables resulting from polynomial integration by projecting it into a smaller set of predictors. As the polynomial degree increases, the number of variables starts to grow rapidly, introducing greater complexity and multicollinearity into the dataset. PLS enables effective dimensionality reduction by extracting a set of latent components that capture the maximum covariance between the explanatory variables and the target [19]. This makes PLS particularly well suited for handling the complex pattern introduced by the integration of a polynomial function.

After these models have been trained, the final step is to evaluate them with evaluation metrics, such as correlation coefficient (R^2), root mean squared error (RMSE), and mean absolute percentage error (MAPE). R^2 ranges from 0 to 1 and shows how good a model understands the data [20], giving higher values a better result. RMSE calculates the mean of the squared error, but is more sensitive to outliers [21], where a lower value is preferable. MAPE shows the mean of absolute error in percentage, good for direct interpretation [22]. Together, these metrics collectively offer a well-rounded evaluation of the predictive performance of the model.

III. RESULTS AND DISCUSSIONS

A polynomial function was implemented into the Gradient Boosting Regressor (GBR), Support Vector Regression (SVR), and Partial Least Squares (PLS) models to enhance their predictive performance. As presented in Table II, the integration of a polynomial function led to measurable improvements in R^2, RMSE, and MAPE across all three models. Notably, the substantial increase in R^2 values in all three models highlights how the integration of a polynomial function greatly improves their ability to explain variance in corrosion inhibition efficiency (CIE). Overall, it demonstrates that integrating a polynomial function not only boosts the prediction accuracy but also enables the models to capture the underlying relationships within the ionic liquid compounds more effectively.

TABLE II
MODEL PERFORMANCE BEFORE AND AFTER POLYNOMIAL

Model	Before Polynomial			After Polynomial		
	R^2	RMSE	MAPE	R^2	RMSE	MAPE
PLS	0.51	6.383	6.24%	0.73	4.730	3.73%
SVR	0.46	6.703	7.04%	0.56	6.083	6.24%
GBR	0.43	6.928	7.23%	0.58	5.445	6.59%

The evaluation shows that the polynomial integration has successfully improved the predictive performance of all models in estimating the CIE of ionic liquid compounds. Notably, PLS outperformed GBR and SVR, especially after the integration of the polynomial function, with scores of R^2, RMSE, and MAPE at 0.73, 4.730, and 3.73%, respectively. With eight latent components and a fourth-degree polynomial, PLS effectively captured the underlying monotonic relationships between predictors and the target while reducing dimensionality. Compared to its baseline version, the Polynomial Integrated PLS showed a notable improvement in R^2 from 0.51 to 0.73, while RMSE and MAPE dropped from 6.383 to 4.730 and from 6.24% to 3.73%, respectively. This performance gain reflects the model's ability to capture nonlinear relationships introduced by the polynomial function. Despite the added complexity, PLS manages to maintain generalization by reducing the numerous variables into a smaller set of predictors. In contrast, the baseline model, which relies solely on linear terms, is more limited in capturing the underlying patterns in data.

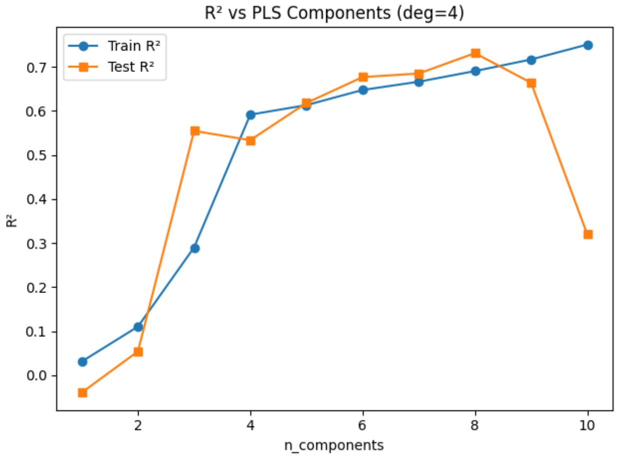

Figure 2. Overfitting Check on PLS Latent Components

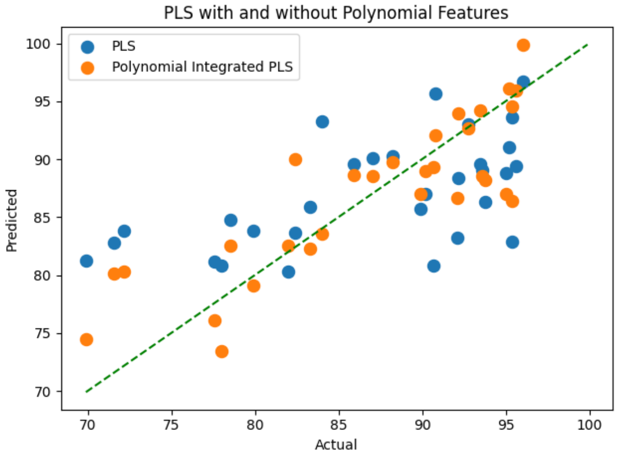

Figure 3. Predicted vs Actual of PLS with and without Polynomial

Figure 2 demonstrates that beyond eight latent components, the model begins to overfit when the training R^2 rises to 0.73 at ten components, while the test R^2 fails to follow the trend by falling sharply to 0.32. This gap indicates that with nine and ten latent components, the model is fitting noise in the training set rather than the true signal. Thus, eight components offer the best balance between model complexity and predictive and generalization.

This improvement is further confirmed by the comparison depicted in Figure 3, where the Polynomial Integrated PLS with eight latent components and a fourth-degree polynomial produces data points that lie substantially closer to the regression line than the baseline PLS model. These visual results combined confirm the observed increases in R^2 alongside reductions in RMSE and MAPE, demonstrating that the polynomial function significantly improves the predictive performance of the PLS model.

The predicted and actual CIE for all three models after the polynomial integration are visualized in Figure 4-6, where the blue dots represent training data, the orange dots represent test data, and the green dotted line indicates the line of perfect prediction. The closer the dots lie to this line, the better the model performance is. Among all three models, the Polynomial Integrated PLS (Figure 4) shows predictions are tightly clustered around the ideal line, reflecting the smallest residuals and the most consistent generalization. Meanwhile,

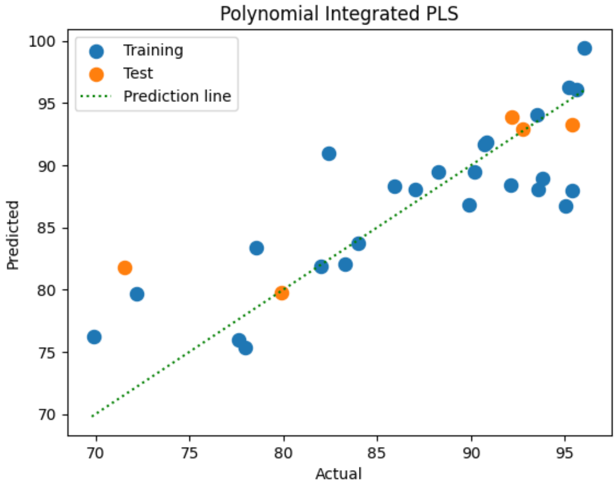

Figure 4. Predicted vs Actual of Polynomial Integrated PLS

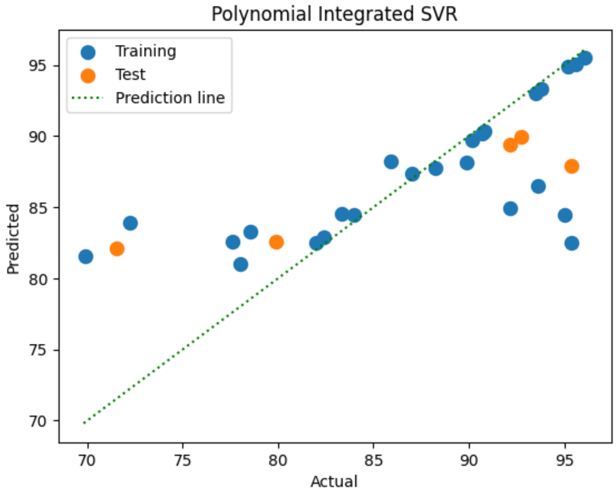

Figure 5. Predicted vs Actual of Polynomial Integrated SVR

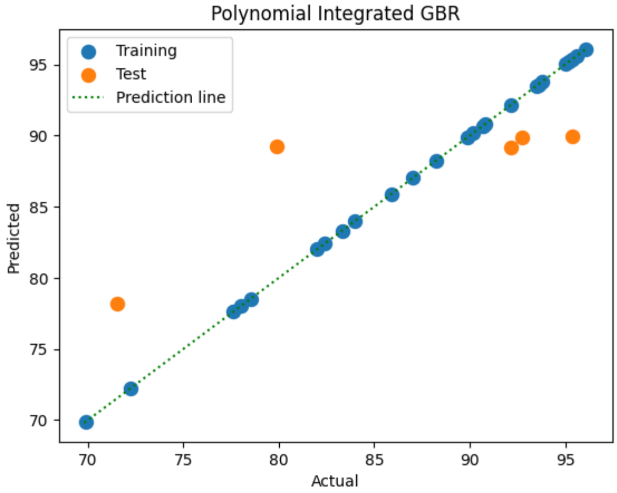

Figure 6. Predicted vs Actual of Polynomial Integrated GBR

the Polynomial Integrated SVR (Figure 5) displays a wider scatter, with noticeable deviations at lower CIE values. In contrast, the Polynomial Integrated GBR (Figure 6) indicates a tendency towards overfitting, with training data closely aligned to the ideal line while the testing data spread away from the line.

Although all three models tend to overpredict at lower CIE values, the Polynomial Integrated PLS remains the most accurate and reliable overall, as reflected by its tighter alignment with the ideal prediction line and superior evaluation metrics. From Table II, it appears that GBR performs better than SVR in terms of evaluation metrics after polynomial integration. However, as illustrated in Figure 3-5, GBR shows signs of overfitting. Considering both the metrics and the visual analysis, the overall order of model performance is PLS > SVR > GBR.

This study also benchmarks its results against the previous study on the same ionic liquids dataset [7]. As shown in Table III, the previous study developed a Multilayer Perceptron Neural Network (MLPNN) model, whereas the proposed model in this study delivers superior predictive performance with lower RMSE and MAPE. While the previous study did not report the R^2, the proposed model achieves strong performance across all evaluation metrics. This direct comparison also demonstrates that the integration of a polynomial function surpasses both conventional regression models and the neural network approach in predicting corrosion inhibition efficiency of ionic liquids.

TABLE III
MLPNN AND POLY PLS MODEL PERFORMANCE

Metrics	MLPNN	Polynomial Integrated PLS
R^2	—	0.73
RMSE	5.407	4.730
MAPE	5.781	3.73%

While all three models demonstrated improved predictive performance, there are important limitations to consider, especially given the small dataset used in this study. A limited number of samples can restrict the diversity of patterns available for training, making all models more susceptible to overfitting or unstable generalization when applied to broader or noisier datasets. PLS, although effective in handling multicollinearity and high dimensionality, may struggle when the underlying relationships are highly nonlinear beyond what latent components can capture. SVR is sensitive to hyperparameter tuning and feature scaling, which can lead to inconsistent performance in the presence of noise or shifted data distributions. GBR, on the other hand, tends to overfit when the dataset is small, as it is not better than memorizing training data rather than learning a generalizable pattern. These factors highlight that the performance reported here is closely tied to the characteristics of the dataset and should be interpreted with caution when transferring to larger or more heterogeneous datasets.

Beyond the three models evaluated in this study, other machine learning or deep learning approaches could also be considered in future work to further handle complex relationships and improve predictive performance. Techniques such as ensemble learning or neural network architectures may capture richer nonlinear patterns when larger datasets become available, although they often come with higher computational cost and reduced interpretability.

In addition, virtual sample generation (VSG) have been shown to enhance the predictive performance for small datasets. For instance, PSO-VSG has been applied and shown promising results in generating meaningful virtual samples [23]. VSG approaches have also successfully demonstrated to improve the prediction performance, especially for CIE [24], [25]. These prior studies show that VSG methods can effectively enrich limited datasets and enhance model generalization. Although this study did not implement such techniques, their reported success highlights promising directions for future work, where combining Polynomial Integrated models with VSG strategies could further improve predictive performance on small and material datasets.

IV. CONCLUSION

The integration of a polynomial function proved to be an effective strategy to enhance the predictive performance of Partial Least Squares (PLS) Regression model, yielding R^2, RMSE, and MAPE of 0.73, 4.730, and 3.73%, respectively. Among the evaluated models, PLS outperformed both Gradient Boosting Regressor (GBR) and Support Vector Regressor (SVR) models in terms of all metrics used, showing consistent superiority both before and after polynomial integration. The optimized configuration with eight latent components demonstrated a superior generalization, as evidenced by minimal residuals and tight gaps between the prediction and experimental CIE. Furthermore, this study benchmarks its results against a previous work utilizing the same ionic liquids dataset. The earlier study developed MLPNN model, yet did not report R^2 for its best-performing model. In contrast, the proposed model shows competitive results with lower RMSE and MAPE. Overfitting analysis confirmed that with the fourth-degree polynomial, PLS strikes the optimal balance between model complexity and predictive reliability.

These findings not only demonstrate the efficacy of the polynomial function integration but also open up potential directions for its broader application. The enhanced performance observed in the PLS model suggests that similar integration strategies may be beneficial when applied to other machine learning models, particularly those dealing with high-dimensional and material datasets or other data where multicollinearity is a concern. In addition, the proposed PLS model offers a promising strategy for virtual sample generation (VSG). Specifically, to get the number of latent components and use it as a reference point for navigating the reduced space and generating meaningful virtual samples.

REFERENCES

[1] M. Akrom *et al.*, "Artificial Intelligence Berbasis QSPR Dalam Kajian Inhibitor Korosi," *JoMMiT J. Multi Media dan IT*, vol. 7, no. 1, pp. 015–020, 2023, doi: 10.46961/jommit.v7i1.721.

[2] C. Verma, E. E. Ebenso, and M. A. Quraishi, "Ionic liquids as green and sustainable corrosion inhibitors for metals and alloys: An overview," *J. Mol. Liq.*, vol. 233, no. 2016, pp. 403–414, 2017, doi: 10.1016/j.molliq.2017.02.111.

[3] S. Budi, M. Akrom, G. A. Trisnapradika, T. Sutojo, and W. A. E. Prabowo, "Optimization of Polynomial Functions on the NuSVR Algorithm Based on Machine Learning: Case Studies on Regression Datasets," *Sci. J. Informatics*, vol. 10, no. 2, pp. 151–158, 2023, doi: 10.15294/sji.v10i2.43929.

[4] V. F. Adiprasetya, M. Akrom, and G. Alfa Trisnapradika, "Investigasi Efisiensi Penghambatan Korosi Senyawa Quinoxaline Berbasis Machine Learning A Study on the Corrosion Inhibition Efficiency of Quinoxaline Compounds Utilizing Machine Learning," *J. Ilm. Tek. Kim.*, vol. 21, no. 2, pp. 2460–8203, 2024.

[5] W. Herowati *et al.*, "Prediction of Corrosion Inhibition Efficiency Based on Machine Learning for Pyrimidine Compounds: A Comparative Study of Linear and Non-linear Algorithms," *KnE Eng.*, vol. 2024, pp. 68–77, 2024, doi: 10.18502/keg.v6i1.15350.

[6] A. H. Alamri and N. Alhazmi, "Development of data driven machine learning models for the prediction and design of pyrimidine corrosion inhibitors," *J. Saudi Chem. Soc.*, vol. 26, no. 6, p. 101536, 2022, doi: 10.1016/j.jscs.2022.101536.

[7] T. W. Quadri *et al.*, "Multilayer perceptron neural network-based QSAR models for the assessment and prediction of corrosion inhibition performances of ionic liquids," *Comput. Mater. Sci.*, vol. 214, no. August, 2022, doi: 10.1016/j.commatsci.2022.111753.

[8] S. Budi *et al.*, "Implementation of Polynomial Functions to Improve the Accuracy of Machine Learning Models in Predicting the Corrosion Inhibition Efficiency of Pyridine-Quinoline Compounds as Corrosion Inhibitors," *KnE Eng.*, vol. 2024, pp. 78–87, 2024, doi: 10.18502/keg.v6i1.15351.

[9] N. V. Putranto, M. Akrom, and G. A. Trinapradika, "Implementasi Fungsi Polinomial pada Algoritma Gradient Boosting Regressor: Studi Regresi pada Dataset Obat-Obatan Kadaluarsa Sebagai Material Antikorosi," *J. Teknol. dan Manaj. Inform.*, vol. 9, no. 2, pp. 172–182, 2023, doi: 10.26905/jtmi.v9i2.11192.

[10] B. J. Rana, N. A. Setiyanto, and M. Akrom, "Prediction of Corrosion Inhibitor Efficiency Based on Quinoxaline Compounds Using Polynomial Regression," *J. Appl. Informatics Comput.*, vol. 9, no. 2, pp. 376–381, 2025, doi: 10.30871/jaic.v9i2.9031.

[11] I. B. Obot, D. D. Macdonald, and Z. M. Gasem, "Density functional theory (DFT) as a powerful tool for designing new organic corrosion inhibitors: Part 1: An overview," *Corros. Sci.*, vol. 99, pp. 1–30, 2015, doi: 10.1016/j.corsci.2015.01.037.

[12] L. B. V. de Amorim, G. D. C. Cavalcanti, and R. M. O. Cruz, "The choice of scaling technique matters for classification performance," *Appl. Soft Comput.*, vol. 133, pp. 1–37, 2023, doi: 10.1016/j.asoc.2022.109924.

[13] M. A. Siddiqi and W. Pak, "Optimizing filter-based feature selection method flow for intrusion detection system," *Electron.*, vol. 9, no. 12, pp. 1–18, 2020, doi: 10.3390/electronics9122114.

[14] H. Akoglu, "User's guide to correlation coefficients," *Turkish J. Emerg. Med.*, vol. 18, no. 3, pp. 91–93, 2018, doi: 10.1016/j.tjem.2018.08.001.

[15] A. K. Nandi, "Data Modeling with Polynomial Representations and Autoregressive Time-Series Representations, and Their Connections," *IEEE Access*, vol. 8, pp. 110412–110424, 2020, doi: 10.1109/ACCESS.2020.3000860.

[16] P. Xu, X. Ji, M. Li, and W. Lu, "Small data machine learning in materials science," *npj Comput. Mater.*, vol. 9, no. 1, pp. 1–15, 2023, doi: 10.1038/s41524-023-01000-z.

[17] S. Bates, T. Hastie, and R. Tibshirani, "Cross-Validation: What Does It Estimate and How Well Does It Do It?," *J. Am. Stat. Assoc.*, vol. 119, no. 546, pp. 1434–1445, 2024, doi: 10.1080/01621459.2023.2197686.

[18] G. Varoquaux, "Cross-validation failure: Small sample sizes lead to large error bars," *Neuroimage*, vol. 180, pp. 68–77, 2018, doi: 10.1016/j.neuroimage.2017.06.061.

[19] V. González, R. Giraldo, and V. Leiva, "PLS1-MD: A partial least squares regression algorithm for solving missing data problems," *Chemom. Intell. Lab. Syst.*, vol. 240, no. March, p. 104876, 2023, doi: 10.1016/j.chemolab.2023.104876.

[20] J. Gao, " R-Squared (R 2) – How much variation is explained? ," *Res. Methods Med. Heal. Sci.*, vol. 5, no. 4, pp. 104–109, 2024, doi: 10.1177/26320843231186398.

[21] C. Miller, T. Portlock, D. M. Nyaga, and J. M. O'Sullivan, "A review of model evaluation metrics for machine learning in genetics and genomics," *Front. Bioinforma.*, vol. 4, no. September, pp. 1–13, 2024, doi: 10.3389/fbinf.2024.1457619.

[22] D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation," *PeerJ Comput. Sci.*, vol. 7, pp. 1–24, 2021, doi: 10.7717/PEERJ-CS.623.

[23] A. Maqbool, A. Khalad, and N. Z. Khan, "Prediction of corrosion rate for friction stir processed WE43 alloy by combining PSO-based virtual sample generation and machine learning," *J. Magnes. Alloy.*, vol. 12, no. 4, pp. 1518–1528, 2024, doi: 10.1016/j.jma.2024.04.012.

[24] M. Akrom, S. Rustad, and H. K. Dipojono, "A machine learning approach to predict the efficiency of corrosion inhibition by natural product-based organic inhibitors," *Phys. Scr.*, vol. 99, no. 3, pp. 1–15, 2024, doi: 10.1088/1402-4896/ad28a9.

[25] I. P. Aldiansah and M. Akrom, "Effect of Virtual Sample Generation in Predicting Corrosion Inhibition Efficiency on Pyridazine," *J. Appl. Informatics Comput.*, vol. 9, no. 2, pp. 382–389, 2025, doi: 10.30871/jaic.v9i2.9131.