Hybrid Thermodynamic and Machine Learning Workflow for Forecasting Turbine Efficiency at Kamojang 3 Geothermal Power Plant

Meigi Rama[1], Sihana Sihana[1], Dwi Joko Suroso[1]

[1]Faculty of Technology, Departement of Nuclear Engineering and Physical Engineering, Universitas Gajah Mada, Yogyakarta, Indonesia
meigirama@mail.ugm.co.id

Abstract. Reliable operation of geothermal power plants depends on stable turbine performance and early detection of degradation. This study builds a thermodynamics-informed, data-driven workflow in Orange to forecast hourly isentropic turbine efficiency from operational logs. The dataset comprises 1,487 hourly records covering the steam, condenser, cooling-water, and generator subsystems. Missing entries were handled by interpolation at the hourly cadence. Thermodynamic consistency was enforced with IAPWS-97 to compute inlet and exhaust states and to form the efficiency target. The observed efficiency ranges from 0.774 to 0.853, approximately 77.4 to 85.3 percent, with a stable trajectory over the window. Five learners were benchmarked on a chronological training split using five-fold cross validation, and Gradient Boosting was selected based on the highest mean R^2. Evaluation on the held-out test period achieved R^2 0.926, RMSE 0.003, MAE 0.002, and MAPE 0.200 percent, which are small relative to the series spread. Combining first-principles thermodynamics with supervised learning in a reproducible Orange pipeline yields accurate and explainable efficiency forecasts that support condition monitoring and timely maintenance planning in geothermal power plants.

Keywords: Geothermal Power Plant, Efficiency, Turbine, Forecasting, Work Flow, Machine Learning, Predictive Maintenance.

1 Introduction

Geothermal energy is one of the major sustainable energy resources, and it can be used to provide continuous electricity with low emissions [1],[2]. As of 2024, Indonesia operates geothermal power plants with a total installed capacity of approximately 2,597 MW, while the Java–Madura–Bali system contributes about 1,210 MW with a capacity factor of 98.36%, which indicates the high electricity demand served by these plants. Given their role in providing baseload power and maintaining high availability, these plants require a dedicated maintenance strategy to ensure sustained reliability.

The Kamojang geothermal area operates five power plants with a total installed capacity of 235 MW, and Kamojang 3 contributes 55 MW, serving as the reference unit

R. Adrian et al. (eds.), *Proceedings of the International Conference on Sustainable Energy: Toward Energy Transition and Net-Zero Emission (ICOSE 2025)*, Atlantis Highlights in Sustainable Development 4,
https://doi.org/10.2991/978-94-6463-944-5_28

for this study. Even though it operates under stable pressure and temperature conditions, contaminants and thermal stress still cause gradual wear on turbine components such as nozzles and blades [3]. The existing maintenance program for the turbine runs on a time-based schedule, whereas anomalies can occur either before or after the scheduled time. In practice, maintenance actions may be performed early when anomalies are detected or postponed if the turbine remains fit for operation [5],[7],[13]. One of the indicators of turbine health is the isentropic efficiency.

This study builds a workflow using Orange to predict hourly turbine isentropic efficiency from operating data. The thermodynamic values are checked with IAPWS-97 to ensure the target data used for efficiency calculation are accurate. The models are trained using a time-based data split, then tested on a separate data period. The model results are evaluated using statistical metrics to confirm the prediction accuracy. The workflow provides a practical tool to support maintenance planning, as predictive maintenance is more effective than time-based maintenance [6],[8],[14], allowing maintenance actions to be executed at the appropriate time in geothermal power plants.

2 Method and Materials

2.1 Data Preparation

The dataset was assembled from geothermal power plant logs that cover three subsystems, which are the main steam system including turbine parameters, the main cooling water system, and the generator system. The logs span from July 1 to August 31, 2017, with hourly sampling. Features were selected from these logs based on operational understanding, focusing on parameters related to turbine efficiency and conditions that may indicate degradation.

The feature set comprised the following variables:

- Turbine rotational speed in (rpm)
- Main steam flow (t/h)
- Main steam pressure (bar)
- Main steam temperature (°C)
- Steam chest pressure (bar)
- Exhaust steam temperature (°C)
- Condenser level (mm)
- Condenser pressure (mbara)
- Cooling water inlet temperature (°C)
- Cooling water outlet temperature (°C)
- Cooling water outlet flow (t/h)
- Generator load (MW)

Data checking was done before modeling. Since the logs were manually recorded, the data were checked manually to prevent typing errors. The workflow did not use the *Outlier* widget because it might automatically remove outlier data and break the time order. Missing values were handled using the *Interpolation* widget along the logged sequence for each variable, which keeps the time order intact.

The Interpolation widget provides several methods for handling missing data, such as linear, mean or mode, nearest point, and cubic spline interpolation. Among these, the nearest point interpolation method was selected because it fills missing entries using the value from the closest available timestamp. This preserves the original data pattern and avoids artificial smoothing that could distort short-term operational behaviour.

This approach is well-suited for the operational data, where each record reflects the actual condition of the system at the time of measurement. By keeping the dataset consistent and physically meaningful, the interpolated data can be used reliably for further analysis and model training.

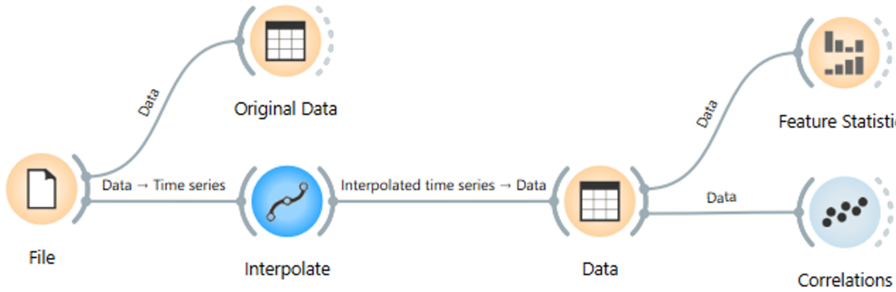

Fig. 1. Data preparation workflow.

Data screening was also performed using the Feature Statistics and Correlations widgets to review basic statistics, value ranges, and relationships between parameters before further calculation and modeling.

2.2 Efficiency Calculation

Environment setup for thermodynamics properties in Orange. The IAPWS-97 library was used to obtain steam and water properties for the efficiency calculation. This ensures accurate and consistent thermodynamic values across pressure and temperature ranges without depending on manual estimation or external lookup tables. Since the standalone version of Orange uses an embedded Python environment that does not include IAPWS by default, the library was installed in the user site directory and linked to Orange through a path adjustment in the Python Script widget.

Before the main process runs, a short verification script checks whether the IAPWS package is available in the environment. If the package is missing, it notifies the user to install it manually through the Orange Python environment. This setup keeps Orange unchanged, does not require administrator access, and allows specific library versions to be used consistently across different runs.

Unit Conversion and Additional Parameters. After the thermodynamic environment was properly configured, the recorded plant data were converted into consistent SI units to ensure accuracy in further calculations. Pressures were converted to MPa, temperatures to Kelvin, and steam flow rates to kilograms per second. These conversions keep all parameters comparable and physically consistent across the dataset.

The converted variables and their corresponding units are listed below:

Table 1. Converted Variables

Variable	Description	Unit
P_1	Steam Chest Pressure	Mpa
P_2	Condenser Pressure	MPa
T_1	Main Steam Temperature	K
T_2	Exhaust Steam Temperature	K
$\dot{m}$	Main Steam Flow	kg/s

From these standardized variables, two additional parameters were calculated as performance indicators:

- Pressure Ratio

$$\text{pr} = \frac{P_1}{P_2} \tag{1}$$

- Specific Steam Consumption

$$\text{ssc} = \frac{\dot{m}}{P_e} \times 3600 \tag{2}$$

The pressure ratio (P_2/P_1) describes the expansion between the turbine inlet and condenser, while the specific steam consumption (SSC) indicates the amount of steam needed to generate one kilowatt-hour of power. Both parameters provide supporting information for evaluating turbine behavior and overall efficiency.

Thermodynamic properties framework. With the data already converted into consistent units, the workflow calculates steam properties using the IAPWS-97 library to ensure thermodynamic accuracy before efficiency modeling. The calculation covers the main thermodynamic properties required for turbine efficiency analysis, including enthalpy, entropy, and saturation temperature derived from the converted pressure and temperature data.

Table X lists the thermodynamic variables generated from this step, which are later used in the turbine efficiency calculation.

Table 2. Generated variables

Variable	Description	Unit
h_1	Inlet enthalpy at P_1 and T_1	kJ/kg
s_1	Inlet entropy at P_1 and T_1	kJ/kg·K
h_{2s}	Isentropic exhaust enthalpy at P_2 with $s = s_1$	K
Tsat_out_K	Saturation temperature at P_2	K
T_2_valid	Flag when T_2 within the acceptable range	-

A plausibility check is also applied to verify whether the measured exhaust temperature is within an acceptable range of the calculated saturation temperature. This step helps identify unrealistic or inconsistent data before further processing.

Isentropic Efficiency Calculation. Using the thermodynamic results obtained in the previous step, the workflow calculates the turbine isentropic efficiency. The process starts by selecting the actual exhaust enthalpy (h_{2a}) before evaluating the overall efficiency.

The actual exhaust enthalpy is calculated using two alternative paths, depending on data availability. When turbine power output (Pe) and steam flow rate ($\dot{m}$) are available, the efficiency derived from power is obtained from

$$\eta_{from\_power} = \frac{w_{actual}}{h1 - h2_s} \tag{3}$$

and the corresponding actual exhaust enthalpy is calculated as

$$h_{2a} = h_1 - \eta_{from\_power}(h_1 - h_{2s}) \tag{4}$$

If the exhaust temperature (T_2) is valid and within a reasonable range of the saturation temperature at the same pressure, the exhaust enthalpy is determined from the thermodynamic state (P_2, T_2). When neither the power nor temperature data are available, a conservative approximation is applied:

$$h_{2a} \approx h_{2s} + 5 \tag{5}$$

After the actual exhaust enthalpy is defined, the isentropic efficiency of the turbine is calculated as

$$\eta_{\_turb\_isentropic} = \frac{h1 - h2_a}{h1 - h2_s} \tag{6}$$

and expressed in percentage form as:

$$\eta_{\_turb\_pct} = 100 \times \eta_{\_turb\_isentropic} \tag{7}$$

Non-physical values, such as negative or greater than 1, are automatically constrained within a valid range to maintain physical consistency throughout the calculation.

Python script output. The last step creates a new data table that combines the original measurements with all converted and calculated results. This process is done using the Python Script widget, as shown in Fig. 2, where the input data are connected to the script and generated as an augmented table.

The script defines new variables and adds them to the original dataset. These include the converted parameters, thermodynamic properties, and turbine efficiency results. All variables are numeric, arranged by row, and use the same units as in previous steps. The output keeps the same structure as the input so it can be used directly for later analysis or model training.

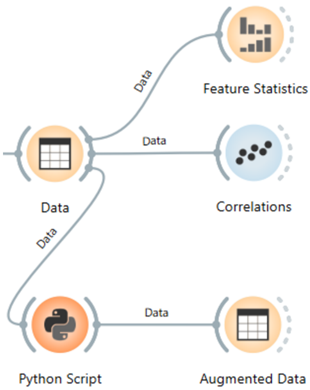

Fig. 2. Python script and output workflow.

The final output includes these variable groups:

- Main parameter : P1_MPa, P2_MPa, T1_K, T2_K
- Steam properties : h1_kJkg, s1_kJkg, h2s_kJkg
- Validation checks : Tsat_out_K, T2_valid
- Exhaust references : h2a_kJkg, eta_from_power
- Final efficiencies : eta_turb_isentropic, eta_turb_pct
- Supporting Data : Pe_MW, mdot_kg_s, pr, ssc, P1_from_chest, T1_from_main

2.3 Machine Learning

Data Partitioning and Target Definition. The augmented table was divided into training and testing sets while maintaining the time order of the measurements. The split used the *Select Rows* widget instead of the *Data Sampler*, which randomizes rows and breaks the time sequence. Because the model was designed for time-series forecasting, keeping the data in time order was essential.

A single cut-off timestamp was manually set to create an 80% training portion and a 20% testing portion. Data earlier than the cut-off formed the training set, while data later than the cut-off formed the testing set. Since the condition applied in the *Select Rows* widget was *greater than*, the training set labelled as *unmatched data* and the testing set as *matching data*. This approach prevents data leakage and keeps the split reproducible because the cut-off time is explicitly recorded.

Next, the data were processed in the Select Columns widget to define the prediction target, which was the turbine isentropic efficiency. Variables that could cause data leakage, such as *eta_turb_pct* and *eta_from_power*, were excluded because both were calculated directly from the actual power output and final turbine efficiency. Keeping these variables as the inputs would make the model indirectly learn information from the target variable during training, which leads to misleadingly high accuracy and poor generalization when applied to new data.

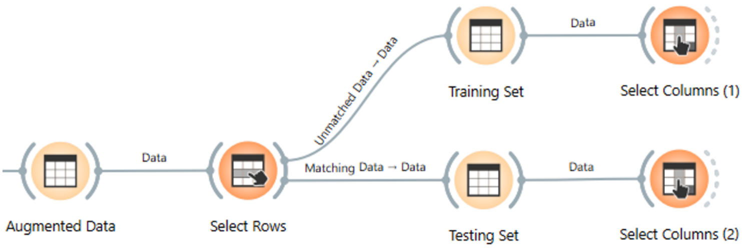

Fig. 3. Data partitioning and target definition workflow.

Model Benchmarking and Selection. The training data were evaluated in Orange using the Test and Score widget to compare several machine learning models. The comparison was performed through cross-validation to ensure consistent results across data folds. The number of folds was adjusted as needed to balance accuracy and computational time.

Five learning algorithms were tested: *AdaBoost*, *Gradient Boosting*, *Random Forest*, *Support Vector Machine* (*SVM*), and *Decision Tree*. Each model's performance was assessed using several regression metrics, with R^2 as the main indicator since values closer to one represent better prediction accuracy. Additional metrics such as MSE, RMSE, MAE, and MAPE were used to measure error magnitude and percentage deviation.

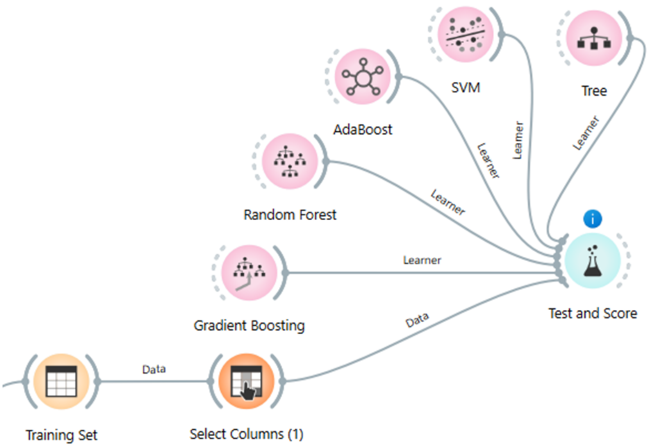

Fig. 4. Model benchmarking workflow.

This process was done entirely on the training data to avoid bias, while cross-validation provided a fair comparison and variance estimate. The model with the highest R^2 score was then selected for final training and prediction.

Final Training and Prediction. After benchmarking, the best-performing model was retrained using the same training data to build the final predictor. The same preprocessing steps and data layout were applied to the testing branch to keep both parts consistent and prevent data leakage. The final model was then connected to the Predictions widget to generate forecasts of turbine isentropic efficiency (*η_turb_isentropic*) on the testing data that were kept separate from training.

Error inspection was performed directly within the *Predictions* widget using the *Difference* view to compare predicted and actual values. Overall model performance was summarised in the same interface using R^2 as the main indicator, while RMSE, MAE, and MAPE were also reviewed to evaluate the error magnitude.

Interpretability Analysis. Model interpretation was performed using the *Feature Importance* and *Explain Model* widgets which were connected to the trained model. Both were applied to the testing dataset to describe how each input parameter affected the predicted turbine isentropic efficiency.

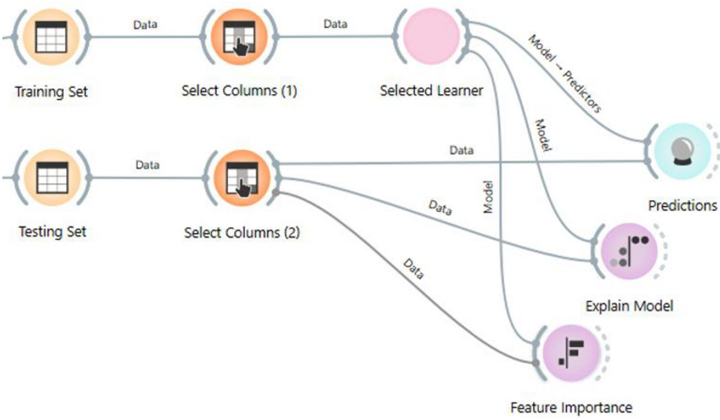

Fig. 5. Prediction and interpretability workflow.

The *Feature Importance* widget measures how much each feature contributes to the model by checking how the prediction accuracy changes when that feature's values are shuffled. The results are shown as a bar chart, where the size of the change in R^2 represents how much influence the feature has on the prediction.

While Feature Importance gives an overall view of which parameters are most influential, the *Explain Model* widget provides a more detailed view of how those parameters behave in each prediction. It visualizes the influence of individual features using a point-distribution (beeswarm) plot, where each dot represents one data sample. Its position on the horizontal axis shows whether it increases or decreases the predicted efficiency, while its color indicates the size of the feature value.

3 Result and Discussion

3.1 Dataset Coverage and Exploratory Statistics

From 1 July to 31 August 2017, the dataset contained 1,487 hourly records across 13 operational variables. About 0.1% of the entries were missing and were resolved using the Interpolate widget while maintaining the original structure. Since the data were recorded at hourly intervals and plant parameters normally vary smoothly over time, interpolation was suitable for filling short gaps. The dataset showed no long-term gaps, so no further imputation was required.

Fig. 6. Interpolation result.

Post-interpolation inspection showed that the plant operated within normal and realistic ranges. Turbine speed was steady around 3004 rpm (2988–3028), main steam pressure near 6.00 bar (5.9–6.8), condenser pressure around 120 mbar(a) (98–151), and generator load near 55.4 MW (45.1–56.8). These values confirmed consistent operation and supported later efficiency analysis without any signs of sensor or equipment issues.

Table 3. Statistic of interpolated data.

Feature	Mode	Mean	Median	Dispersion	Min.	Max.
Turbine Speed	3000	3003.9	3004	0.00164254	2988	3028
Main Steam Flow	422	421.532	422	0.012194	351	431
Main Steam Press	6.0	6.0002	6	0.00379234	5.9	6.8
Main Steam Temp	168.0	167.868	168	0.00557546	164	171
Steam Chest Press	5.40	5.40825	5.4	0.00648096	5.3	5.5
Exhaust Steam Temp	53.0	53.5328	53.4	0.0287213	48	57.2
Condenser Level	20	-7.53194	-10	-4.80953	-98	92
Condenser Press	121	120.41	121	0.0543113	98	151
Condenser Water Inlet Temp	30.0	30.2973	30.2	0.0300252	27	33
Condenser Water Outlet Temp	49.0	49.2145	49	0.0211772	45.5	52
Condenser Water Outlet Flow	12.20	12.2536	12.2	0.0214766	11.2	13.5
Generator Load	55.7	55.4094	55.5	0.0135838	45.1	56.8

Linear relationships among the main variables were checked using Pearson correlation. The strongest pairs were cooling-water outlet versus inlet temperature (+0.907), outlet versus exhaust steam temperature (+0.843), and inlet versus exhaust steam temperature (+0.843). These results match the expected condenser-side thermal interaction, confirming the dataset's internal consistency. No new variables were added at this stage.

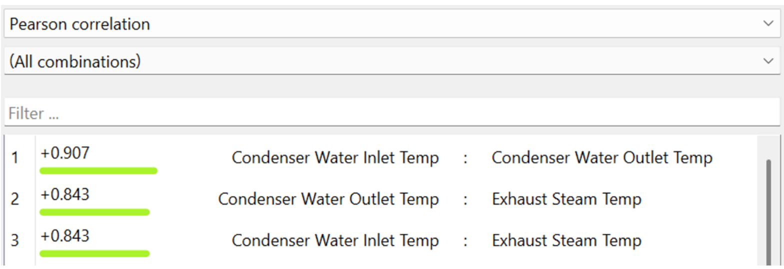

Fig. 7. Top three highest correlations features.

3.2 Thermodynamic States and Efficiency Results

The verified dataset was then used to analyse the thermodynamic states that describe the turbine operating conditions. All parameters reflected normal plant behaviour with stable values across the main process variables, forming the basis for subsequent efficiency calculations.

Steam-chest pressure averaged 0.541 MPa (range 0.530–0.550 MPa), while condenser pressure averaged 0.012 MPa (range 0.010–0.015 MPa). The pressure ratio (pr) was centred at 0.022, with a median of 0.022 and a range of 0.019–0.028. Main steam temperature averaged 441.018 K (range 437.150–444.150 K), and exhaust temperature averaged 326.683 K (range 321.150–330.350 K). The condenser saturation temperature averaged 322.612 K, corresponding to a mean superheat of about 118.540 K. Steam flow was steady at 117.092 kg/s (equivalent to 421.532 t/h), and generator load averaged 55.409 MW, confirming stable operation at hourly resolution.

Table 4. Statistic of converted operational states.

Feature	Mode	Mean	Median	Dispersion	Min.	Max.
P1_MPa	0.54	0.540825	0.54	0.00648096	0.53	0.55
P2_MPa	0.0121	0.012041	0.0121	0.0543113	0.0098	0.0151
T1_K	441.15	441.018	441.15	0.00212223	437.15	444.15
T2_K	326.15	326.683	326.55	0.00470649	321.15	330.35
Tsat_out_K	322.736	322.612	322.736	0.00344783	318.563	327.258
mdot_kg_s	117.222	117.092	117.222	0.012194	97.5	119.722
Pe_MW_norm	55.7	55.4094	55.5	0.0135838	45.1	56.8

Thermodynamic properties calculated using IAPWS-97 behaved as expected. Inlet enthalpy (h_1) averaged 2782.390 kJ/kg, inlet entropy (s_1) averaged 6.865 kJ/kg·K (range 6.845–6.884), and isentropic exhaust enthalpy (h_{2s}) averaged 2197.030 kJ/kg·K. The selected actual exhaust enthalpy (h_2a) averaged 2299.500 kJ/kg (range 2281.610–2326.310 kJ/kg). The plausibility flag T_2 valid equaled 1 across the dataset, confirming that all exhaust temperatures were within reasonable physical bounds.

Table 5. Statistic of thermodynamic properties and calculation

Feature	Mode	Mean	Median	Dispersion	Min.	Max.
h1_kJkg	2782.76	2782.39	2782.61	0.000764564	2773.5	2789.62
s1_kJkgK	6.86659	6.86509	6.86503	0.000804649	6.84552	6.8843
h2s_kJkg	2197.63	2197.03	2197.63	0.00325161	2169.93	2226.77
h2a_kJkg	2293.28	2299.5	2299.49	0.0027834	2281.61	2326.31
eta_turb_isentropic	0.829946	0.824988	0.826343	0.00998283	0.774494	0.852905
T2_valid	1	1	1	0	1	1

The isentropic efficiency was calculated from the thermodynamic states defined by IAPWS-97. The inlet state provided the values of h1 and s1, while the ideal exhaust at the same entropy gave h_{2s}. The actual exhaust enthalpy h_{2a} was selected based on the method explained earlier. The resulting efficiency remained within a reasonable range, with a mean value of 0.825, median 0.826, and range 0.774–0.853. The hourly results showed stable behaviour that matched turbine operation. When condenser pressure increased, the available energy drop became smaller, causing efficiency to decrease. In contrast, when the turbine operated under higher thermal energy conditions, the efficiency improved due to the larger energy potential between the inlet and exhaust sides.

3.3 Predictive Modelling

Benchmarking and model selection. Model benchmarking used the *Test and Score* widget with five-fold cross-validation on the training data. The main measure was R^2, while RMSE, MAE, and MAPE were used to show the size of prediction errors. Five models which are *Gradient Boosting*, *AdaBoost*, *Random Forest*, *Decision Tree*, and *SVM* were compared. The final choice was the model with the highest R^2 and stable results across folds.

Table 6. Benchmarking result

Model	MSE	RMSE	MAE	MAPE	R2
Gradient Boosting	0.000	0.001	0.001	0.097	0.953
AdaBoost	0.000	0.002	0.001	0.107	0.920
Random Forest	0.000	0.002	0.001	0.120	0.915
Tree	0.000	0.003	0.002	0.190	0.817
SVM	0.000	0.015	0.013	1.568	-4.421

Gradient Boosting achieved the best result with $R^2 = 0.953$ and very small RMSE and MAE (around 0.001). This shows that the model accurately learned the relationship between turbine operating parameters and efficiency while keeping prediction errors minimal.

Gradient Boosting achieved the best result with $R^2 = 0.953$ and very small RMSE and MAE (about 0.001), showing that it could capture the relationship between turbine parameters and efficiency with high precision. *AdaBoost* and *Random Forest* followed with $R^2 \approx 0.920$ and 0.915; both performed well but with slightly larger errors. *AdaBoost* is more sensitive to residual noise, while *Random Forest* tends to underfit gradual nonlinear patterns due to averaging many deep trees. The single *Decision Tree*, with $R^2 = 0.817$, showed limited flexibility because it uses a single hierarchy of splits and cannot represent complex variable interactions. Meanwhile, the *Support Vector Machine* scored $R^2 = -4.421$, indicating its default settings were not suitable for this dataset; without kernel tuning or feature scaling, it failed to capture the nonlinear relationships among turbine parameters.

Overall, *Gradient Boosting* provided the best balance of accuracy and stability, making it the most suitable model for final evaluation and interpretability analysis.

Model Tuning and Performance Evaluation. Following the benchmarking stage, *Gradient Boosting* was selected as the best-performing learner based on its superior R^2 and error consistency. To further improve its predictive stability, a detailed manual tuning process was carried out using Orange's *Gradient Boosting* (scikit-learn) implementation. The default configuration, which consisted of 100 trees, a learning rate of 0.100, and full sampling, was used as the baseline for comparison.

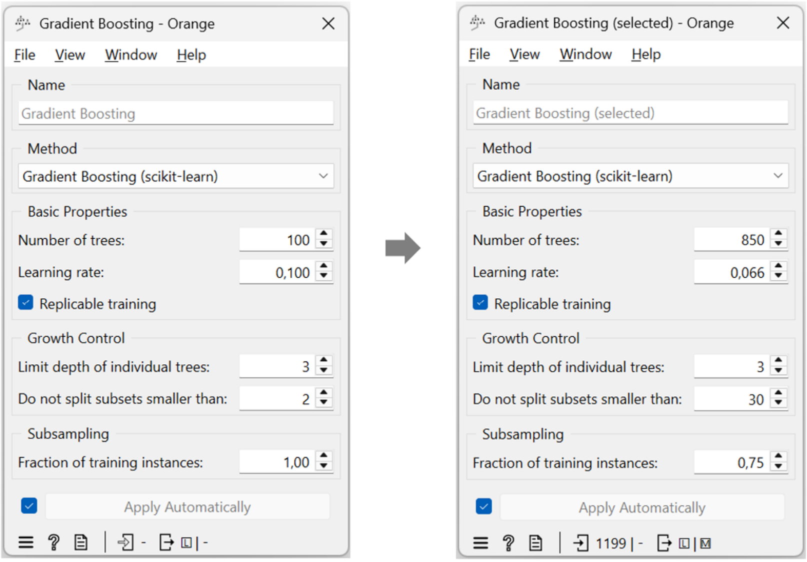

Fig. 8. Parameter optimization.

The tuning process aimed to balance model flexibility with robustness against noise. The learning rate was reduced to 0.066, allowing the model to make smaller and more controlled updates during training. This adjustment ensured smoother convergence and reduced the risk of overfitting. The number of trees was expanded substantially from 100 to 850, giving the ensemble enough capacity to capture nonlinear interactions between turbine parameters.

To maintain interpretability and prevent excessive depth growth, the maximum tree depth was kept at three. Meanwhile, the minimum number of samples required for a branch split was increased from two to thirty, strengthening statistical validity in each decision node and minimizing spurious splits from small fluctuations in plant measurements. Subsampling was also applied at 0.75, meaning that each iteration used 75% of the training data. This introduces mild randomness between trees, improving the model's ability to generalize across unseen data.

Figure 8 compares the default and tuned configurations, showing the progression from a simple baseline to a more adaptive and stable model. With these optimized parameters, the *Gradient Boosting* model achieved $R^2 = 0.926$ on the testing split, with RMSE = 0.003, MAE = 0.002, and MAPE = 0.200. Although slightly lower than its cross-validation mean ($R^2 = 0.953$), the test performance remains strong and consistent for hourly data. The small residuals and narrow prediction spread show that the model captures the key turbine behavior consistently over time.

Overall, the tuned configuration demonstrates improved accuracy, reduced error variance, and reliable reproducibility. This balance between model precision and robustness confirms *Gradient Boosting* as a suitable and efficient predictor for turbine efficiency forecasting in geothermal plant operations.

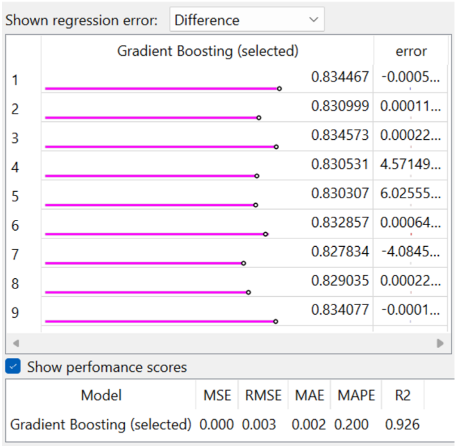

Fig. 9. Prediction result.

Model Interpretability and Physical Consistency. With the tuned Gradient Boosting model fixed and the test configuration mirroring the training setup, the next step was to examine which input variables contributed most to the predictions. Model interpretability was explored through two views: Explain Model for local contributions and Feature Importance for global ranking.

As shown in Figure X, the Explain Model widget visualizes each prediction as a sum of signed feature effects, where positive values increase the predicted efficiency and negative values reduce it. The beeswarm plot combines these effects for all records, with horizontal position showing the impact magnitude and color representing the feature value. In this visualization, the pressure ratio (pr) clearly dominates and follows physical intuition—higher pr corresponds to higher backpressure, which lowers turbine efficiency.

The specific steam consumption (ssc) follows as a key driver, where lower values indicate better performance due to reduced steam usage per generated power. Higher actual exhaust enthalpy (h2a_kJkg) shifts points left, reflecting a smaller enthalpy drop across the turbine, while higher isentropic exhaust enthalpy (h2s_kJkg) shifts points right, indicating an idealized larger enthalpy drop and higher efficiency. Other variables such as Tsat_out_K, P2_MPa, and condenser pressure show minor, centered impacts, meaning they influence efficiency less directly in this operating window.

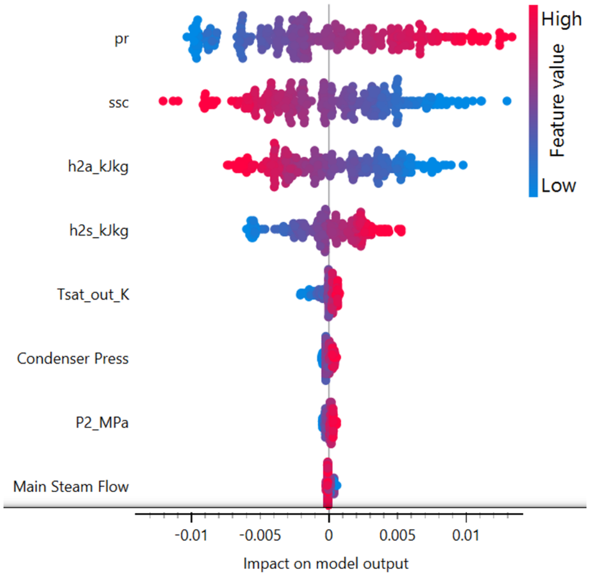

Fig. 10. Explain model.

Complementing the local analysis, Figure Y presents the Feature Importance results, which quantify how much each variable affects model performance when shuffled. The largest decrease in R^2 occurred when pr was permuted, confirming it as the most influential predictor. The next tier includes ssc, h2a_kJkg, and h2s_kJkg, aligning with the local effects observed earlier. Meanwhile, Tsat_out_K, P2_MPa, and condenser pressure contribute marginally once the main thermodynamic terms are accounted for.

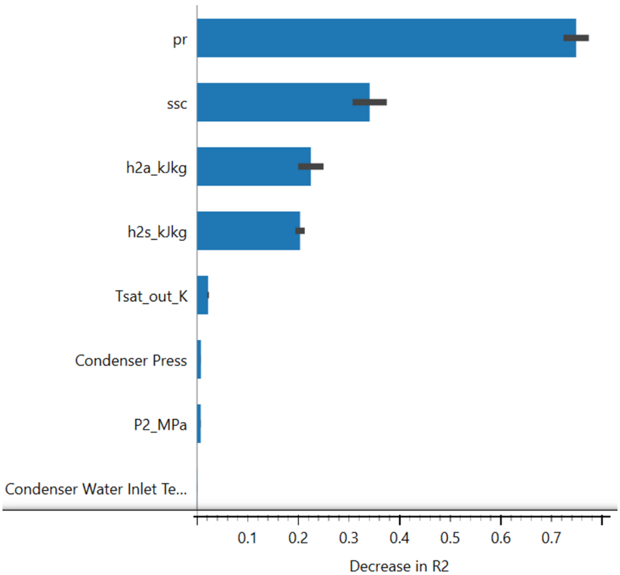

Fig. 8. Feature importance.

Together, these two interpretability views confirm that the model's behavior aligns with turbine physics. The Gradient Boosting model primarily captured backpressure effects and energy conversion represented by enthalpy and steam consumption, rather than learning statistical artifacts. This consistency between model output and physical reasoning supports its reliability for interpreting turbine performance and monitoring efficiency trends.

Summary. In summary the workflow produced a reliable efficiency signal. Interpolation preserved coverage, IAPWS-97 states were consistent, and the observed isentropic efficiency lay between 0.774 and 0.853. *Gradient Boosting* generalised best on the test period with R^2 0.926 and low absolute errors. Explain Model and permutation importance both pointed to pressure ratio as the main driver, with enthalpy terms and work per flow as secondary effects. The approach is suitable for condition monitoring, with wider validation across units and seasons left for future work.

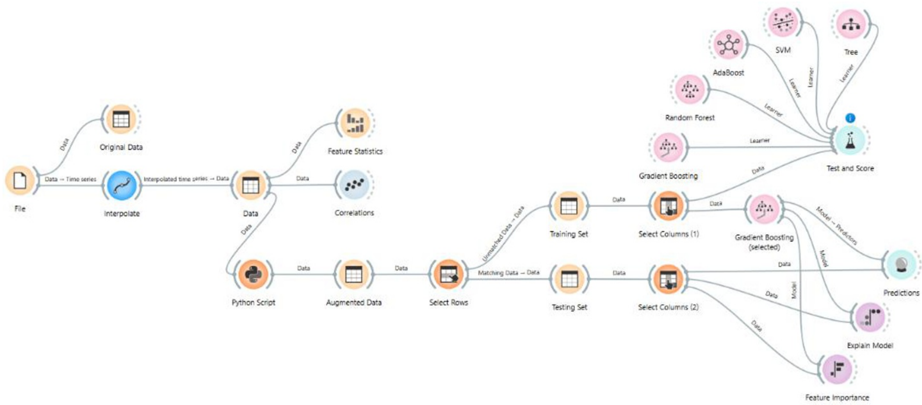

Fig. 9. End-to-end Orange workflow.

4 Conclusion

This study delivers a reproducible physics-informed workflow that forecasts hourly isentropic turbine efficiency from routine plant logs. IAPWS-97 provides consistent inlet and exhaust states so the target is physically meaningful. The Orange pipeline preserves time order, benchmarks multiple learners, and selects a stable model. *Gradient Boosting* maintains strong agreement on a chronological holdout with R^2=0.926 and small absolute errors, while explanations align with turbine fundamentals, pressure ratio dominates and enthalpy terms and work per flow provide secondary modulation [1]. The predictor is therefore accurate and technically interpretable.

The resulting artefacts, a live efficiency trace and transparent feature attributions, could credibly underpin predictive and condition-based maintenance [5],[6]. Plants could use the trace to build baselines and control bands for efficiency, detect drift with enough lead time to plan work, and check recovery after interventions [5]. The attributions could focus troubleshooting on backpressure or superheat when efficiency softens, while persistent residual patterns could surface sensor or tagging issues before they mislead operations [1]. Because the workflow is compact and auditable, it could be transferred to other units with modest adaptation.

In the wider context of sustainable energy, the workflow could help geothermal units stay closer to best achievable efficiency, which may reduce steam use per megawatt, lower the frequency of unplanned outages, and sustain a higher capacity factor [1],[2]. Such operational steadiness would likely ease the integration of variable renewables and contribute to energy-transition and net-zero objectives.

The study covers a single unit over a stable window. Future work may extend validation across seasons and additional units, include exogenous signals where available, calibrate uncertainty and action thresholds, and run the pipeline online with linkage to the maintenance system [5],[12]. With these steps the approach could become a plant-ready module that strengthens reliability while supporting low-emission geothermal generation.

Acknowledgments. We gratefully acknowledge PLN Indonesia Power for providing access to operational data from Kamojang Unit 3 and for permitting the use of aggregated results for research and publication. We thank Universitas Gadjah Mada, Department of Nuclear Engineering and Engineering Physics (DTNTF), for academic collaboration and supervision, including guidance from the author's academic advisor.

This study is part of the author's Master's program at Universitas Gadjah Mada and fulfills a scholarship commitment to PT PLN. Scholarship support from PT PLN is gratefully acknowledged. Responsibility for the analysis, interpretations, and any remaining errors rests with the authors.

Disclosure of Interests. The authors declare no competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. Access to operational data from PLN Indonesia Power and scholarship support from PT PLN are acknowledged in the Acknowledgments. Neither organization influenced the study design, data analysis, interpretation, or the decision to submit this manuscript.

References

1. DiPippo, R.: Geothermal power plants: Evolution and performance assessments. Geothermics 53, 291–307 (2015).
2. Moya, D., et al.: Geothermal energy: Power plant technology and direct heat applications. Renewable and Sustainable Energy Reviews 94, 889–901 (2018).
3. Rudiyanto, B., et al.: Preliminary analysis of dry-steam geothermal power plant by employing exergy assessment: Case study in Kamojang Geothermal Power Plant, Indonesia. Case Studies in Thermal Engineering 10, 292–301 (2017).
4. Gu, H.-H., et al.: Data-physics-model based fatigue reliability assessment methodology for high-temperature components and its application in steam turbine rotor. Reliability Engineering & System Safety 241, 109633 (2024).

5. Huang, C., et al.: Prognostics and health management for predictive maintenance: A review. Journal of Manufacturing Systems 75, 78–101 (2024).
6. Cheng, W., et al.: Diagnostics and prognostics in power plants: A systematic review. Reliability Engineering & System Safety, 110663 (2024).
7. Hiruta, T., et al.: A design method of data analytics process for condition based maintenance. CIRP Annals 68(1), 145–148 (2019).
8. Ling, W., et al.: Efficient data-driven models for prediction and optimization of geothermal power plant operations. Geothermics 119, 102924 (2024).
9. Que, Z., Xu, Z.: A data-driven health prognostics approach for steam turbines based on XGBoost and DTW. IEEE Access 7, 93131–93138 (2019).
10. IAPWS: Revised Release on the IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam (IAPWS-IF97). International Association for the Properties of Water and Steam (2012). URL: https://www.iapws.org, last accessed 2025/09/29
11. Hagmeyer, S., Zeiler, P.: A comparative study on methods for fusing data-driven and physics-based models for hybrid remaining useful life prediction of air filters. IEEE Access 11, 35737–35753 (2023).
12. Li, H., et al.: A review on physics-informed data-driven remaining useful life prediction: Challenges and opportunities. Mechanical Systems and Signal Processing 209, 111120 (2024).
13. Arena, S., et al.: A novel decision support system for managing predictive maintenance strategies based on machine learning approaches. Safety Science 146, 105529 (2022).
14. Mitici, M., et al.: Dynamic predictive maintenance for multiple components using data-driven probabilistic RUL prognostics: The case of turbofan engines. Reliability Engineering & System Safety 234, 109199 (2023).
15. Breiman, L.: Random forests. Machine Learning 45, 5–32 (2001)
16. PT PLN (Persero). (2025). Rencana Usaha Penyediaan Tenaga Listrik (RUPTL) PLN 2025–2034. PT PLN (Persero), Jakarta. Retrieved from https://web.pln.co.id/statics/uploads/2025/06/b967d-ruptl-pln-2025-2034-pub-.pdf