Libyan Journal of Applied and Contemporary Sciences
ISSN
E: 3136-3709
L: 3136-3695
Vol 1: No 2

Forecasting or Contemporaneous Estimation? A Leakage-Aware Industrial Energy Audit with Uncertainty Quantification

Esam Miftah Abdulnabi Aboudoumat [1*] ; Nabeel Faraj Amhimmid Abdullah [1] ; Ashraf Faraj Saed Albarki [2] ; Faraj Ahmed Mohammed Alfarisi [1]

[1]Department of Computer, College of Science and Technology Qumins, Libya Affiliations
[2]Department of Computer Science, College of Arts and Sciences Qumins, University of Benghazi, Libya

[*]Email corresponding author's: esam.mouftah@gmail.com

Received: 28/6/2026 Accepted: 18/8/2026 Published: 20/8/2026

Abstract

Short-term industrial energy forecasts can assist load planning only when all predictors are available at the forecast issuance time. This research audits 15-minute and 60-minute forecasting with 35,040 real observations from a South Korean steel plant. We reconstruct a continuous 15-min timeline, define a deployable 39-feature protocol using measurements available no later than the forecast origin, and compare it to two diagnostic protocols that admit target-time sensors. We compare ridge regression, Random Forest and XGBoost to persistence and daily and weekly seasonal-naive baselines for chronological and random 70/15/15 train/calibration/test splits. The clean chronological Random Forest obtained mean absolute error (MAE) of 3.829 kWh at 15 minutes and 7.484 kWh at 60 minutes, which corresponds to a decrease of MAE relative to persistence by 23.35% and 35.31%. All paired moving-block bootstrap intervals against the operational baselines did not include zero. Target-time measurements tell a different story, cutting XGBoost MAE by 82.72% and 90.12%. Sub-1-kWh results of that kind describe contemporaneous estimation, not deployable forecasting. Target-time CO2 was a strong proxy, 0.988 correlated with energy use, and 97.69% of its values equaled rounded usage multiplied by 0.00045. The marginal coverage of the split-conformal intervals sat near its nominal level. Peak-load coverage at the 90% setting did not: 51.99% and 44.70%. Feature-availability semantics and conditional uncertainty assessment were more important for our interpretation than the choice between strong tree learners.

Keyword : industrial energy forecasting; data leakage; temporal evaluation; conformal prediction; uncertainty quantification; Random Forest; XGBoost; smart manufacturing
Doi : https://doi.org/10.64943/ljacs.2026.010204

INTRODUCTION

Plant-level load forecasts are useful only if they can be issued before the demand they are meant to anticipate. In this study, the key question is therefore not simply whether a regression model predicts well, but whether every predictor exists when the forecast is made. A sensor value recorded at the target timestamp is unavailable to a forecast issued 15 or 60 minutes earlier, even when the predictor and target appear in the same row of the stored dataset.

The distinction is important for industrial datasets in which energy use, reactive power, power factor, emissions, and a contemporaneous load label are logged together. A random row split can further obscure the operational meaning of evaluation by mixing seasons and operating regimes across partitions. General discussions of leakage define the problem as information crossing the learn-predict boundary [1], while recent evidence shows that leakage can produce irreproducible or practically

invalid machine-learning claims across scientific domains [2]. Forecast evaluation is especially exposed because temporal preprocessing, feature extraction, partitioning, and horizon definition can each allow future information to enter the model [3].

We study this issue using the Steel Industry Energy Consumption dataset [4], which contains real 15-minute measurements from DAEWOO Steel in Gwangyang, South Korea. The associated work established the dataset and demonstrated the value of data-mining models for industrial energy prediction [5]. We do not propose another algorithmic benchmark in isolation. Instead, we ask whether the task is forecasting or contemporaneous estimation under three explicit feature-availability protocols.

Our study makes four contributions. First, we reconstruct the dataset timeline and document a midnight-encoding issue that creates 365 backward jumps under naive timestamp parsing. Second, we compare a deployable past-only feature set with two non-deployable target-time sensor sets and quantify the resulting optimism. Third, we evaluate point forecasts using chronological partitions, operational baselines, peak-load errors, and paired moving-block bootstrap intervals. Fourth, we audit split-conformal intervals both marginally and during high-load periods, showing that near-nominal overall coverage can conceal severe conditional undercoverage.

We deliberately avoid two overclaims. The random split is not assumed to be uniformly optimistic; its effect is measured and found to depend on the forecast horizon. Likewise, near-nominal empirical conformal coverage is not treated as a formal guarantee for this dependent, shifted time series. These restrictions preserve the difference between an empirical audit and a theorem about deployment.

RELATED WORK

Industrial energy prediction

The Steel Industry Energy Consumption dataset contains energy, electrical, temporal, and operating-load variables collected from a smart small-scale steel facility [4]. Sathishkumar et al. [5] used the data to investigate data-analytic energy prediction for an industrial building. This work made an important real dataset available and motivated subsequent model comparisons. However, a row-oriented table does not by itself specify the forecast origin or whether same-row measurements are available before the target. Our analysis treats availability as an explicit experimental factor.

Forecast validation should match the time at which an operational decision would actually be made. Tashman [6] argued for genuine out-of-sample tests, while Bergmeir and Benitez [7] clarified the settings in which cross-validation can still be defensible for time-series predictors. Cerqueira et al. [8] showed empirically that the choice of performance-estimation scheme interacts with nonstationarity. Recent regional work on monthly oil-production forecasting likewise found that explicitly modelling seasonal temporal structure improved forecast performance, with SARIMA outperforming ARIMA when seasonal patterns were present [9]. Hewamalage et al. [3] brought these issues together, identifying weak baselines, unsuitable accuracy measures, and leakage introduced during preprocessing or horizon construction as recurring sources of misleading forecast results. For that reason, our operational conclusions rely on the chronological split; the random split is retained only as a diagnostic.

Leakage and feature availability

Leakage was formally defined by Kaufman et al. [1] who proposed a strict learn-predict separation. This is no small implementation detail: Kapoor and Narayanan [2] show that it can invalidate the target estimand and lead to reproducibility failures. Forecasting complicates this further. A feature can be valid for one operational design and not for another, and CO2 is the clean example here, as we have the measurement taken at the forecast origin but not the one logged at the future target time. We label each feature by its availability relative to the forecast origin, not by its name.

This is not a standard ablation. Removing the target-time variables does not test whether they are informative, which is already clear, and origin-time CO2 alone is second in the 15 minute SHAP

ordering. The audit tests if the forecasting interpretation of an accuracy result is maintained after the learn-predict border is imposed.

Models, metrics, and dependent comparisons

We include Ridge regression [10] as a regularized linear reference, Random Forest [11] as a bagged non-linear learner and XGBoost [12] as a boosted tree learner. TreeSHAP describes the trained clean XGBoost model [13]. We report MAE, RMSE, symmetric MAPE (sMAPE) and R-squared. As discussed by Hyndman and Koehler [14], percentage error measures need care when the target can be zero, and the energy target here runs down to 0.00 kWh. MAE is our primary ranking measure for that reason; sMAPE is supplementary. Resampling individual errors one at a time would throw away the serial dependence. We instead compare paired absolute-error sequences using a moving-block bootstrap with one-day blocks, following the general block-resampling principle for stationary observations developed by Kunsch [15]. The resulting percentile intervals are descriptive for the fixed test period and do not establish external validity.

Distribution-free uncertainty for time series

Conformal prediction can wrap a point predictor with distribution-free predictive intervals under exchangeability [16,17]. Conformalized quantile regression adapts interval width to heteroscedasticity [18]. Extensions address covariate shift [19], dynamic time series [20], adaptive time-series coverage [21], and non-exchangeability [22]. Our implementation uses the simpler split-conformal residual interval because it is transparent and computationally light. We explicitly test where that simplicity fails: conditional coverage at high load. Since chronological calibration and test observations are not exchangeable in the usual sense, we report empirical coverage rather than claiming exact finite-sample validity.

MATERIAL AND METHODS

Data source

The public dataset is distributed by the UCI Machine Learning Repository under CC BY 4.0 [4] and is also linked to the original industrial-energy study [5]. There are 35,040 observations across 11 raw columns: energy use in kWh, lagging and leading reactive power, CO2, lagging and leading power factors, seconds from midnight, week status, day of week, and load type. Nothing is missing and no row is duplicated.

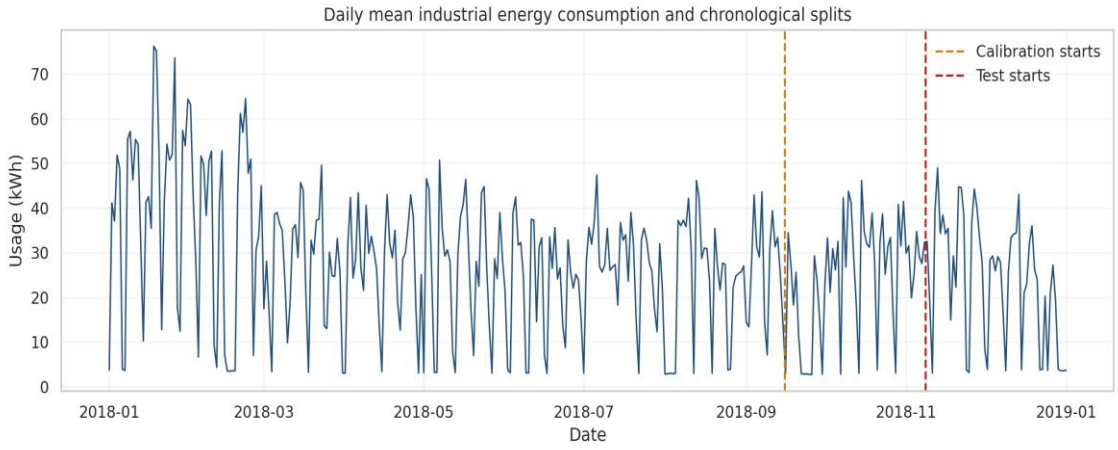

Figure 1. Daily energy consumption, with the chronological train, calibration and test regions marked.

Timestamp reconstruction

Parsing the supplied date string directly would produce 365 backward jumps, one for each day. These were found when we were checking date integrity and ordering prior to fitting anything, not from a

model misbehaving downstream. It is caused by an encoding quirk: the last observation of each day has the time 00:00, but keeps the previous date. Nothing is wrong with the measurements themselves, only with their order. We rebuild the index as a continuous 15-minute sequence running from 2018-01-01 00:15 to 2019-01-01 00:00, and the reconstructed time-of-day then matches the supplied seconds-from-midnight variable on all 35,040 rows.

Table 1. Dataset integrity and timestamp audit.

Audit item	Result
Observations	35,040
Raw columns	11
Sampling interval	15 minutes
Missing values	0
Duplicate rows	0
Energy mean	27.387 kWh
Energy standard deviation	33.444 kWh
Energy range	0.00-157.18 kWh
Naive timestamp backward jumps	365
Reconstructed clock/NSM agreement	100%

Target and forecast origins

Let *y_tau* denote energy use at target time *tau*. For horizon *h* in {1, 4}, corresponding to 15 and 60 minutes, the forecast origin is *t = tau - h*. A deployable forecast has the form

$$y\_hat\_(t+h) = f\_h(X\_available\_at\_or\_before\_t, C\_(t+h)),$$

where *C_(t+h)* contains only calendar information known in advance. Target-time sensor measurements are excluded because they are observed after the forecast is issued.

CO_2 proxy audit

CO2 and energy use have a Pearson correlation of 0.9882. More importantly, 97.694% of the rows satisfy *CO2 = round(Usage_kWh x 0.00045, 2)*. CO2 has only eight distinct values, compared with 3,343 energy values. This pattern makes target-time CO2 a strong proxy for the target, although origin-time CO2 remains a legitimate predictor.

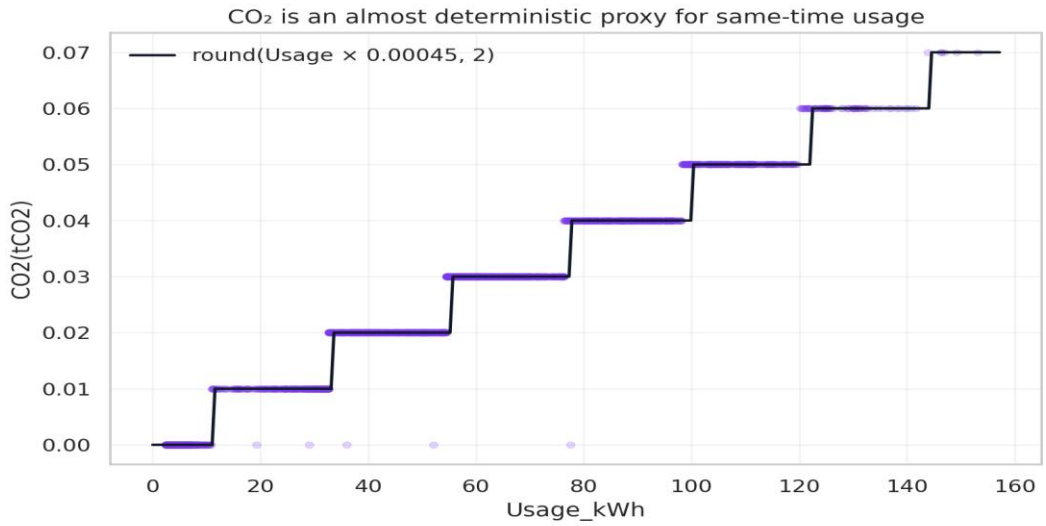

Figure 2. Relationship between energy consumption and CO2, including the rounded usage-based proxy pattern.

Experimental Design

Feature-availability protocols

We construct three protocols indexed by target time. The clean protocol contains 39 features. Seven describe the target calendar using sine/cosine encodings for time slot, day of week, and day of year plus a weekend indicator. Energy-history features include usage at the forecast origin; lags 1, 2, 4, 12, and 24 steps before the origin; target-relative lags of 24 hours, 48 hours, and seven days; rolling means, standard deviations, minima, and maxima over 1 hour, 3 hours, 24 hours, and seven days; and 1-hour and 24-hour differences. The remaining clean features are the five numerical sensor readings shifted to the forecast origin.

Table 2. Feature-availability protocols.

Protocol	Feature count	Added information	Operational status
Clean past-only	39	Calendar plus energy and sensors available no later than origin	Deployable
Target sensors without CO2	46	Five target-time electrical readings and target load-type indicators	Diagnostic only
All target-time features	47	Previous protocol plus target-time CO2	Diagnostic only

The target load label is placed in the diagnostic set because its same-row value describes the operating state at the target instant. Kept in the clean set, it would act as a future observation without saying so.

Partitions

For the main experiment, we preserve time order and allocate 70% of the usable observations to training, 15% to calibration, and 15% to testing. After feature construction, the 15-minute task contains 34,368 targets. The training segment runs from 2018-01-08 00:15 to 2018-09-15 14:15 and contains 24,057 observations; the next 5,155 observations form the calibration segment through 2018-11-08 07:00, and the final 5,156 observations form the test segment through 2019-01-01 00:00. At the four-step horizon, two additional early targets are unavailable because of the longer look-ahead, while the final test period remains unchanged.

As a sensitivity check, we also create a reproducible random 70/15/15 partition using the fixed seed. Its role is limited to showing how the conclusions change when chronology is removed. We do not use

these scores to support operational claims because a random partition mixes earlier and later operating periods rather than forecasting a future period from past data.

Learners and baselines

Ridge uses standardised inputs and *alpha = 10*. Random Forest uses 240 trees, all depths available, 0.8 feature sampling, and four worker processes. XGBoost uses 450 trees, depth 6, learning rate 0.04, subsample 0.85, column subsample 0.85, squared-error objective, and histogram tree construction. The fixed random seed is 42. Hyperparameters were selected as stable, moderate-complexity settings; no exhaustive search was conducted.

Three baselines represent operational alternatives: persistence uses the latest origin value, the daily seasonal-naive forecast uses the value 96 intervals earlier, and the weekly seasonal-naive forecast uses the value 672 intervals earlier. Including simple baselines prevents a complex model from being judged only against weaker learned models [3].

Evaluation measures

For *n* test observations, MAE and RMSE are

$$MAE = (1/n)\ sum_i\ |y_i - y\_hat_i|,$$

$$RMSE = sqrt((1/n)\ sum_i\ (y_i - y\_hat_i)^2).$$

We also report sMAPE and R-squared. Peak-load MAE is calculated on test observations above the 90th percentile of the training target, which equals 82.48 kWh at both horizons. This fixed training-derived threshold prevents test information from defining the subgroup.

Paired moving-block bootstrap

For each best learned model and baseline, we compute the paired series $d_i = |e\_model,i| - |e\_baseline,i|$. We sample contiguous blocks of 96 observations, wrap blocks as required until a bootstrap sample reaches the test length, and calculate the mean paired difference. The 2.5th and 97.5th percentiles over 2,000 replicates form the reported 95% interval. Negative values favour the learned model.

Split-conformal intervals

For a fitted point model, the calibration nonconformity score is $s_i = |y_i - y\_hat_i|$. For target miscoverage *alpha*, we use the finite-sample corrected calibration quantile and construct

$$C\_alpha(x) = [y\_hat(x) - q\_(1-alpha),\ y\_hat(x) + q\_(1-alpha)].$$

We evaluate nominal 90% and 95% intervals on the chronological test set. In addition to overall coverage and mean width, we report peak coverage using the same training-derived threshold. The intervals are an empirical baseline, not a formal time-series guarantee.

Interpretation and reproducibility

TreeSHAP values are computed for the clean 15-minute XGBoost model [13]. Everything runs on Python 3.12: NumPy 2.3.5, pandas 2.2.3, scikit-learn 1.8.0, XGBoost 3.0.5, SHAP 0.49.1, Matplotlib 3.10.8. The raw file hashes to SHA-256

9b1cee6f9cb9cd9df2b95814ca90a9a2ff15b7f5f1fba0fae3c643e82072eacc.

RESULTS AND DISCUSSION

Clean chronological forecasts

Across the clean chronological tests, Random Forest produced the smallest overall MAE and RMSE at both horizons. At 15 minutes, its MAE was 3.829 kWh and its RMSE was 8.012 kWh, with R-squared of 0.934. XGBoost was close on aggregate error and gave the lower peak-load MAE, 15.283 kWh compared with 15.788 kWh for Random Forest. The same overall ordering remained at 60 minutes: Random Forest reached MAE 7.484 kWh and R-squared 0.809, whereas XGBoost again had the slightly smaller peak-load MAE.

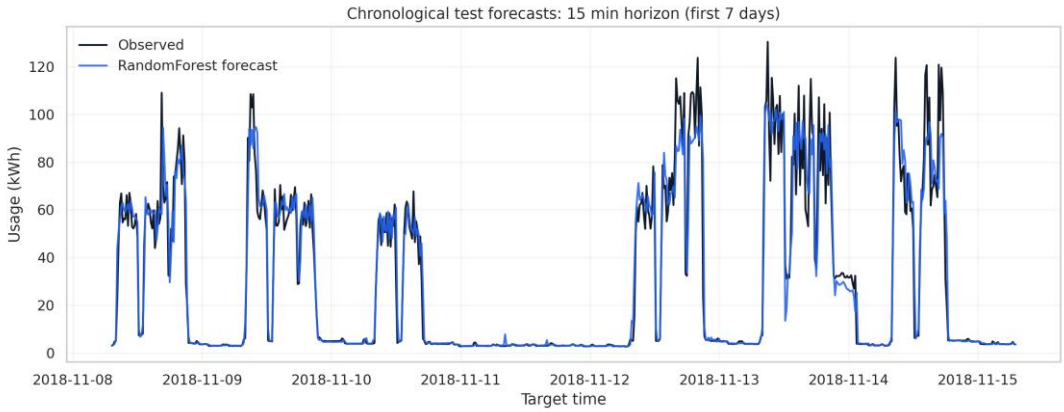

Figure 3. Clean chronological 15-minute test predictions for learned models and baselines.

Table 3. Clean chronological point-forecast results.

Horizon	Model	MAE (kWh)	RMSE (kWh)	sMAPE (%)	R-squared	Peak MAE (kWh)
15 min	Random Forest	3.829	8.012	13.004	0.934	15.788
15 min	XGBoost	4.012	8.122	15.118	0.933	15.283
15 min	Persistence	4.995	11.639	14.110	0.861	20.206
15 min	Ridge	6.177	10.133	46.235	0.895	20.250
15 min	Seasonal naive 7d	12.814	24.403	39.277	0.391	44.664
15 min	Seasonal naive 24h	14.617	26.997	47.516	0.255	44.929
60 min	Random Forest	7.484	13.660	31.184	0.809	25.663
60 min	XGBoost	8.172	14.155	37.883	0.795	24.492
60 min	Ridge	10.626	15.721	70.369	0.747	32.509
60 min	Persistence	11.570	24.003	34.630	0.411	36.934
60 min	Seasonal naive 7d	12.814	24.403	39.277	0.391	44.664
60 min	Seasonal naive 24h	14.617	26.997	47.516	0.255	44.929

Ridge's large sMAPE is driven mainly by the low-load part of the test set, where percentage errors become unstable. We therefore do not use sMAPE to determine the model ranking; MAE and RMSE remain directly interpretable in kWh.

Bootstrap comparisons

All paired 95% bootstrap intervals are below zero. Against persistence, Random Forest reduces MAE by 1.166 kWh (23.35%) at 15 minutes and 4.086 kWh (35.31%) at 60 minutes. Improvements relative to the daily and weekly seasonal baselines are larger.

Table 4. Paired moving-block bootstrap comparisons for Random Forest.

Horizon	Baseline	Model-baseline MAE difference	95% interval	Relative MAE reduction
15 min	Persistence	-1.166	[-1.531, -0.826]	23.35%
15 min	Seasonal naive 24h	-10.788	[-13.151, -8.983]	73.81%
15 min	Seasonal naive 7d	-8.985	[-10.584, -7.163]	70.12%
60 min	Persistence	-4.086	[-5.491, -2.571]	35.31%

Horizon	Baseline	Model-baseline MAE difference	95% interval	Relative MAE reduction
60 min	Seasonal naive 24h	-7.133	[-9.640, -5.194]	48.80%
60 min	Seasonal naive 7d	-5.329	[-7.025, -3.442]	41.59%

These intervals support a difference on this held-out period. They do not remove the limitation that only one facility-year is available.

Feature-availability leakage

The protocol comparison produces the largest effect in the study. For XGBoost, adding target-time electrical sensors and load indicators reduces MAE from 4.012 to 0.868 kWh at 15 minutes and from 8.172 to 1.088 kWh at 60 minutes. Adding target-time CO2 reduces the values further to 0.693 and 0.808 kWh.

Table 5. Effect of target-time feature availability on MAE.

Horizon	Model	Clean past-only MAE	Target sensors, no CO2	All target-time features	Apparent reduction from clean
15 min	Ridge	6.177	2.649	1.819	70.55%
15 min	Random Forest	3.829	0.977	0.962	74.87%
15 min	XGBoost	4.012	0.868	0.693	82.72%
60 min	Ridge	10.626	4.822	2.115	80.10%
60 min	Random Forest	7.484	1.113	0.973	87.00%
60 min	XGBoost	8.172	1.088	0.808	90.12%

For XGBoost, CO2 provides an additional 20.10% MAE reduction at 15 minutes and 25.74% at 60 minutes after the other target-time variables are present. These numbers should not be interpreted as forecast improvements. They quantify how strongly a contemporaneous estimation task can outperform a real forecast when target-time availability is ignored.

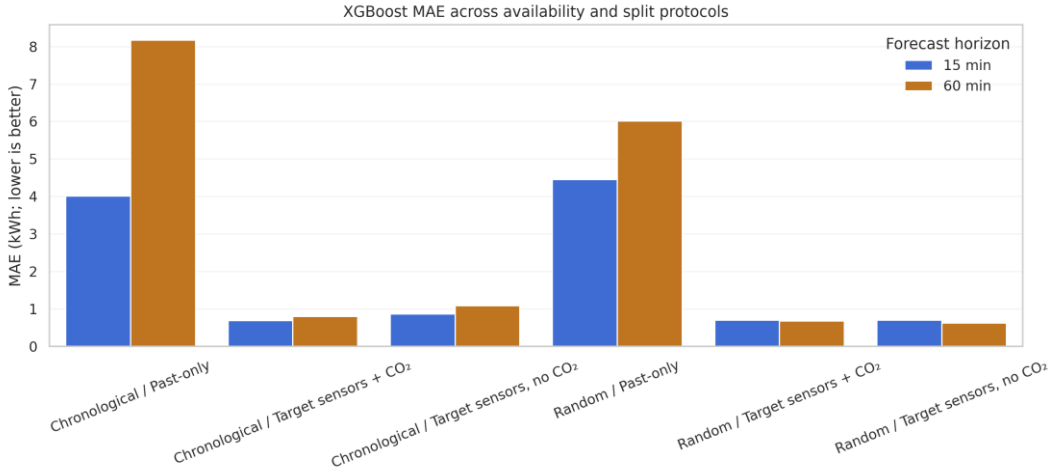

Figure 4. MAE by horizon, split design, feature-availability protocol, and model.

Random versus chronological partitioning

Random splitting does not produce a uniform direction of bias. At 15 minutes, random-split MAE is higher for Ridge, Random Forest, and XGBoost by 7.85%, 14.86%, and 11.00%. At 60 minutes, it is 2.98% higher for Ridge but 24.46% and 26.42% lower for the two tree models. Therefore, the scientific issue is not that random splitting must always improve scores. It changes which operating regimes are represented in each partition and can alter conclusions in a horizon- and model-dependent way.

Table 6. Sensitivity of clean-model MAE to random versus chronological splitting.

Horizon	Model	Chronological MAE	Random MAE	Random change vs chronological
15 min	Ridge	6.177	6.662	+7.85%
15 min	Random Forest	3.829	4.398	+14.86%
15 min	XGBoost	4.012	4.453	+11.00%
60 min	Ridge	10.626	10.942	+2.98%
60 min	Random Forest	7.484	5.654	-24.46%
60 min	XGBoost	8.172	6.013	-26.42%

The target mean declines from 28.719 kWh in training to 24.553 kWh in testing, and the 90th percentile declines from 82.48 to 69.425 kWh. The descriptive two-sample Kolmogorov-Smirnov statistic is 0.161 and the Wasserstein distance is approximately 4.46 kWh. We do not attach an iid p-value to the KS statistic because the observations are serially dependent.

Conformal interval audit

The Random Forest split-conformal intervals obtain empirical marginal coverage close to their nominal levels. At 15 minutes, 90% and 95% intervals cover 91.29% and 95.85% of test targets. At 60 minutes, coverage is 89.72% and 94.78%. However, the peak-load subgroup is substantially undercovered.

Table 7. Random Forest split-conformal coverage audit.

Horizon	Nominal coverage	Empirical coverage	Mean interval width (kWh)	Peak coverage
15 min	90%	91.29%	27.367	51.99%
15 min	95%	95.85%	40.631	67.88%
60 min	90%	89.72%	43.022	44.70%
60 min	95%	94.78%	60.724	62.25%

At nominal 90%, peak coverage is only 51.99% at 15 minutes and 44.70% at 60 minutes. The fixed symmetric residual quantile is therefore too narrow when energy use is high. Good marginal calibration does not imply operationally adequate protection for the most consequential subgroup.

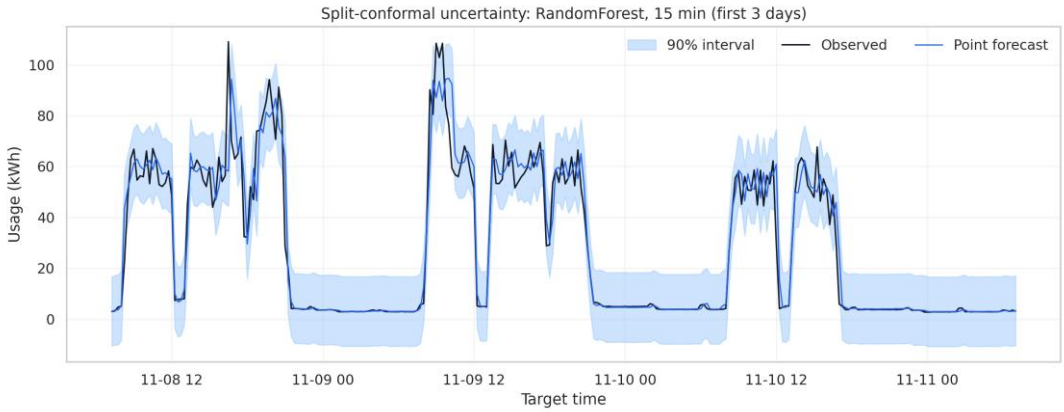

Figure 5. Random Forest 90% and 95% split-conformal intervals for the 15-minute horizon.

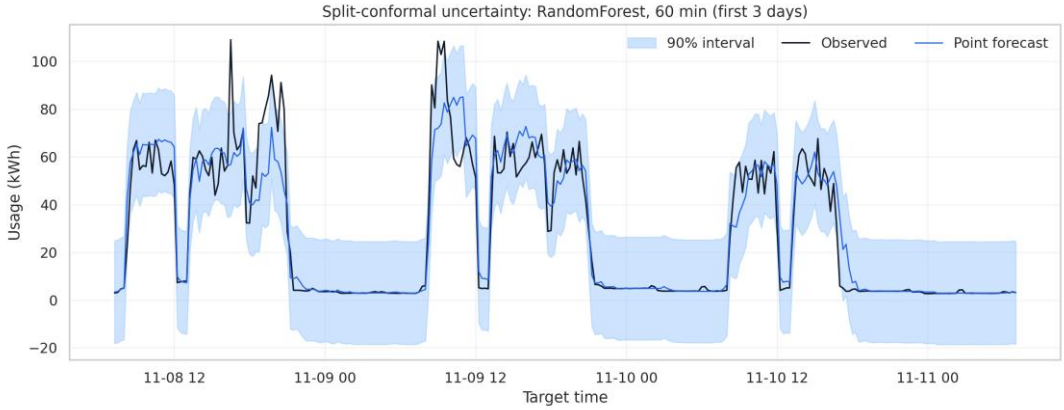

Figure 6. Random Forest 90% and 95% split-conformal intervals for the 60-minute horizon.

Model interpretation

TreeSHAP ranks usage at the forecast origin as the dominant 15-minute XGBoost feature, with mean absolute SHAP 21.291. Origin-time CO2 comes second at 4.010. Behind it sit the previous usage lag, the target time-of-day cosine, the weekly lag and the daily lag, an ordering consistent with a load process that is strongly persistent and nonlinear at once.

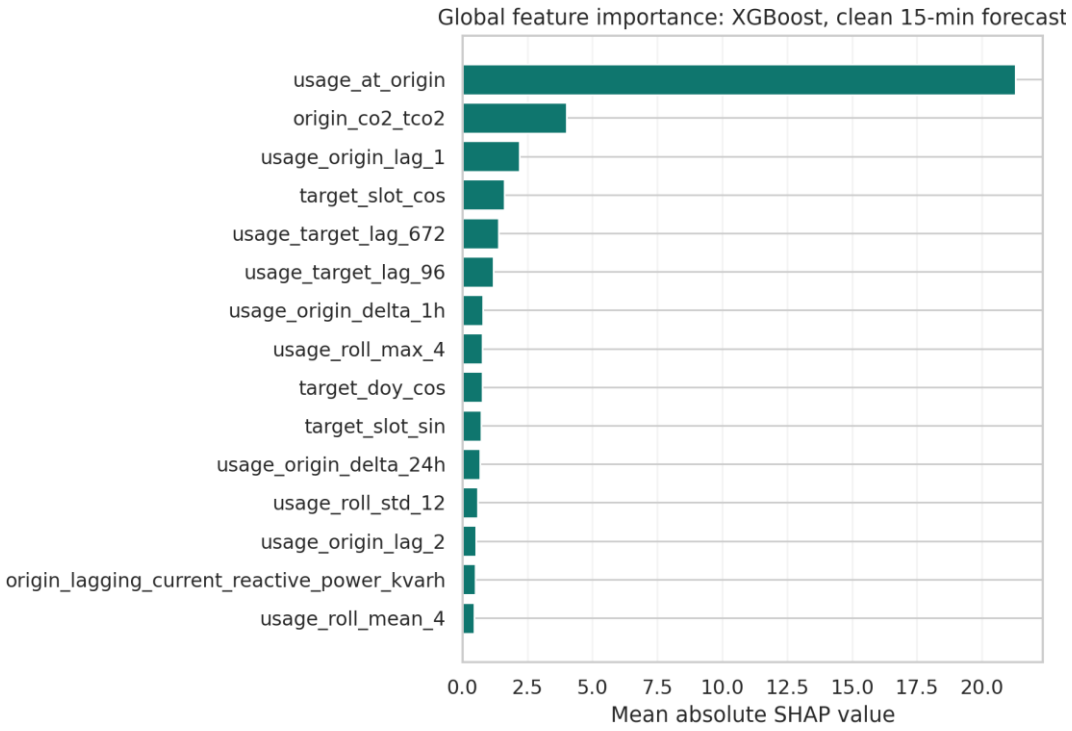

Figure 7. Mean absolute TreeSHAP importance for the clean 15-minute XGBoost model.

The distinction between origin and target remains essential. Origin-time CO2 is legitimately available and can capture the current operating regime. Target-time CO2 is future information under both forecast horizons.

Discussion and Implications

Accuracy is conditional on task semantics

The clean Random Forest result supports a useful short-term prediction claim: when only origin-available measurements and known calendar information are used, it improves over persistence and seasonal-naive baselines on a later test period. The contemporaneous protocols support a different claim: sensors recorded with the target make the current energy state easy to estimate. Both statements can be true, but they answer different questions.

The scale of the protocol effect exceeds the difference between Random Forest and XGBoost. At 15 minutes, the clean models differ by only 0.183 kWh MAE, while changing from clean to all target-time features lowers XGBoost MAE by 3.318 kWh. At 60 minutes, the corresponding protocol reduction is 7.364 kWh. Model selection without an availability specification can therefore optimise the wrong estimand with high numerical precision.

Why random splitting behaved inconsistently

Random splitting combines high- and low-demand periods throughout training and test, whereas the chronological split evaluates a later, lower-load regime. A random test may be more difficult at 15 minutes because it has more dispersed operating conditions than the last chronological segment. Random training with wider regime coverage can help nonlinear models interpolate at 60 minutes. This interpretation agrees with the measured shift, but is not a causal decomposition. The strong conclusion is that random split changes the question and should not replace a deployment aligned test.

Marginal coverage can conceal peak risk

Marginal coverage gives a reassuring but incomplete picture. At the 15-minute horizon, the nominal 90% interval covers 91.29% of all test observations, yet coverage falls to 51.99% when evaluation is restricted to the training-defined peak subgroup; the corresponding 60-minute peak coverage is lower still. Inspection of the errors shows that residual magnitude increases with load, so one symmetric calibration radius cannot represent low-load and peak regimes equally well.

Several extensions could address this pattern. Conformalized quantile regression [18], weighted conformal methods under shift [19], EnbPI [20], adaptive conformal procedures [21], and methods designed beyond exchangeability [22] each offer a different way to relax the fixed-width assumption. A load-stratified or Mondrian variant is also operationally plausible, provided that the strata and calibration rules are fixed from training and calibration data before test outcomes are examined. A useful next comparison would report both interval efficiency and peak coverage under rolling-origin evaluation.

Operational implications

We translate the audit into three implementation checks that do not depend on the learner. First, each candidate feature should carry an explicit availability timestamp or a documented latency rule. Second, offline evaluation should construct the feature vector from the forecast origin rather than selecting values from an already assembled target row; this is the practical distinction between our 39-feature forecasting protocol and the 47-feature diagnostic protocol. Third, monitoring should separate aggregate performance from peak-load behaviour for both point error and interval coverage. These checks can be applied before a model is considered for deployment.

LIMITATIONS

External validity. The evidence comes from one plant and one calendar year: 35,040 records from a South Korean steel facility in 2018. The study does not capture changes in production schedules, tariffs, weather, equipment, or sensor conventions that may occur at other facilities. Accordingly, the reported MAE values should not be interpreted as transferable performance estimates.

Construct validity. We define deployment as a forecast issued 15 or 60 minutes before the target using sensors available at the forecast origin. Under a different operational definition, such as nowcasting or sensors with different reporting latency, the admissible feature set would change. The target-time protocols are therefore invalid for the future-forecasting task defined in Section 3.3, but not for every possible industrial prediction task.

Internal validity. Hyperparameters were fixed in advance rather than tuned to maximise the reported scores: Ridge alpha = 10, Random Forest with 240 trees, XGBoost with 450 trees at depth 6, and seed 42 throughout. This limits claims about the best attainable learner, but it also avoids repeatedly adapting the comparison to the calibration split. The experiment is intended to audit task definitions, not to establish a state-of-the-art leaderboard.

Statistical validity. The bootstrap results depend on the chosen resampling scheme: one-day blocks of 96 observations and 2,000 replicates, with dependence treated as approximately stationary within a block. Different block choices could alter the interval estimates. Split-conformal inference has a separate limitation because its classical validity relies on exchangeability, which a chronologically shifted series need not satisfy. We therefore report empirical coverage on the held-out period and do not claim exact distribution-free validity.

Subgroup definition. Peak load is defined by the training-set 90th percentile, 82.48 kWh. This is an analysis threshold rather than a universal operating limit; another facility could instead use a tariff trigger, an equipment rating, or a cost-sensitive threshold. Conditional coverage may also differ across operating regimes that were not isolated by this definition.

Interpretability. SHAP values describe the fitted predictor rather than a causal mechanism. Because energy lags and electrical sensor readings are correlated, importance can be distributed across related variables, so individual attributions should be interpreted in that context.

REPRODUCIBILITY AND AVAILABILITY

To support replication, the released package contains the attributed UCI data and its hash, the executable experiment code, pinned dependency versions, saved predictions, metric tables, bootstrap and conformal outputs, SHAP values, figures, and environment metadata. The experiment uses seed 42; any remaining variation should be limited to library-level parallel numerical behaviour. The full experiment is launched with one command:

python src/run_experiments.py --bootstrap-replicates 2000

CONCLUSION

The audit separates two prediction tasks that can look identical in a row-oriented industrial dataset. With only information available at the forecast origin and a chronological holdout, Random Forest achieved MAE 3.829 kWh at 15 minutes and 7.484 kWh at 60 minutes, and every paired block-bootstrap interval against the operational baselines remained below zero. When target-time measurements were admitted, XGBoost MAE fell by as much as 90.12%; that result reflects contemporaneous estimation rather than a genuine future forecast. The random-split sensitivity analysis also changed direction across horizons. Uncertainty evaluation revealed a second distinction: the 15-minute 90% split-conformal interval attained 91.29% marginal coverage but only 51.99% coverage on the peak-load subgroup. The main methodological finding is therefore that feature availability and subgroup-specific uncertainty assessment can alter the scientific interpretation more than the choice between two competitive tree learners. Generalisation beyond this plant-year remains untested, and load-adaptive conformal methods under rolling deployment are a natural next step.

Compliance with Ethical Standards

This analysis uses only the public UCI industrial dataset; it contains no human-subject or personal records and involved no intervention, so ethical approval was not required. The dataset is licensed under Creative Commons Attribution 4.0 [4].

Disclosure of Conflict of Interest

The authors declare that they have no conflict of interest.

REFERENCES

1. Kaufman, S., Rosset, S., Perlich, C., & Stitelman, O. (2012). Leakage in data mining: Formulation, detection, and avoidance. *ACM Transactions on Knowledge Discovery from Data, 6*(4), Article 15, 1-21. https://doi.org/10.1145/2382577.2382579
2. Kapoor, S., & Narayanan, A. (2023). Leakage and the reproducibility crisis in machine-learning-based science. *Patterns*, *4*(9), Article 100804. https://doi.org/10.1016/j.patter.2023.100804
3. Hewamalage, H., Ackermann, K., & Bergmeir, C. (2023). Forecast evaluation for data scientists: Common pitfalls and best practices. *Data Mining and Knowledge Discovery, 37*, 788-832. https://doi.org/10.1007/s10618-022-00894-5
4. Sathishkumar, V. E., Shin, C., & Cho, Y. (2021b). *Steel Industry Energy Consumption* [Data set]. UCI Machine Learning Repository. https://doi.org/10.24432/C52G8C
5. Sathishkumar, V. E., Shin, C., & Cho, Y. (2021a). Efficient energy consumption prediction model for a data analytic-enabled industry building in a smart city. *Building Research & Information, 49*(1), 127-143. https://doi.org/10.1080/09613218.2020.1809983
6. Tashman, L. J. (2000). Out-of-sample tests of forecasting accuracy: An analysis and review. *International Journal of Forecasting, 16*(4), 437-450. https://doi.org/10.1016/S0169-2070(00)00065-0
7. Bergmeir, C., & Benitez, J. M. (2012). On the use of cross-validation for time series predictor evaluation. *Information Sciences, 191*, 192-213. https://doi.org/10.1016/j.ins.2011.12.028

8. Cerqueira, V., Torgo, L., & Mozetic, I. (2020). Evaluating time series forecasting models: An empirical study on performance estimation methods. *Machine Learning, 109*, 1997-2028. https://doi.org/10.1007/s10994-020-05910-7

9. Al-Sharif, H. M. A.-M. (2025). Analysis and evaluation of ARIMA and SARIMA models performance in time series forecasting: An applied study. *Libyan Journal of Medical and Applied Sciences, 3*(2), 146-157. https://doi.org/10.64943/ljmas.v3i2.89

10. Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics, 12*(1), 55-67. https://doi.org/10.1080/00401706.1970.10488634

11. Breiman, L. (2001). Random forests. *Machine Learning, 45*(1), 5-32. https://doi.org/10.1023/A:1010933404324

12. Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785-794). https://doi.org/10.1145/2939672.2939785

13. Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems* (Vol. 30, pp. 4765-4774).

14. Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting, 22*(4), 679-688. https://doi.org/10.1016/j.ijforecast.2006.03.001

15. Kunsch, H. R. (1989). The jackknife and the bootstrap for general stationary observations. *The Annals of Statistics, 17*(3), 1217-1241. https://doi.org/10.1214/aos/1176347265

16. Angelopoulos, A. N., & Bates, S. (2023). Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning, 16*(4), 494-591. https://doi.org/10.1561/2200000101

17. Lei, J., G'Sell, M., Rinaldo, A., Tibshirani, R. J., & Wasserman, L. (2018). Distribution-free predictive inference for regression. *Journal of the American Statistical Association, 113*(523), 1094-1111. https://doi.org/10.1080/01621459.2017.1307116

18. Romano, Y., Patterson, E., & Candes, E. J. (2019). Conformalized quantile regression. In *Advances in Neural Information Processing Systems* (Vol. 32, pp. 3538-3548).

19. Tibshirani, R. J., Barber, R. F., Candes, E. J., & Ramdas, A. (2019). Conformal prediction under covariate shift. In *Advances in Neural Information Processing Systems* (Vol. 32).

20. Xu, C., & Xie, Y. (2021). Conformal prediction interval for dynamic time-series. In *Proceedings of the 38th International Conference on Machine Learning* (Vol. 139, pp. 11559-11569). PMLR.

21. Zaffran, M., Feron, O., Goude, Y., Josse, J., & Dieuleveut, A. (2022). Adaptive conformal predictions for time series. In *Proceedings of the 39th International Conference on Machine Learning* (Vol. 162, pp. 25834-25866). PMLR.

22. Barber, R. F., Candes, E. J., Ramdas, A., & Tibshirani, R. J. (2023). Conformal prediction beyond exchangeability. *The Annals of Statistics, 51*(2), 816-845. https://doi.org/10.1214/23-AOS2276

التنبؤ أم القياس المتزامن؟ تدقيق واعٍ بتسرب البيانات للطاقة الصناعية مع تقدير عدم اليقين

الملخص

تكون دقة التنبؤ باستهلاك الطاقة الصناعية ذات معنى فقط عندما تكون جميع الخصائص المستخدمة متاحة فعلياً لحظة إصدار التنبؤ. نقدّم تدقيقاً واعياً بتسرب البيانات للتنبؤ بعد 15 و60 دقيقة باستخدام 35,040 مشاهدة فعلية من منشأة لصناعة الصلب في كوريا الجنوبية. أُعيد بناء التسلسل الزمني، وعُرّف بروتوكول قابل للتطبيق يضم 39 خاصية لا تتجاوز لحظة إصدار التنبؤ، ثم قورن ببروتوكولين تشخيصيين يستخدمان قياسات الزمن المستهدف. حققت الغابة العشوائية متوسط خطأ مطلق قدره 3.829 kWh عند أفق 15 دقيقة و7.484 kWh عند أفق 60 دقيقة، مع خفض الخطأ مقارنة بخط الاستمرارية بنسبة 23.35% و35.31%. في المقابل، أدى إدخال قياسات الزمن المستهدف إلى خفض ظاهري لخطأ XGBoost بنسبة تصل إلى 90.12%، وهو ما يصف تقديراً متزامناً لا تنبؤاً مستقبلياً. حققت فواصل conformal تغطية كلية قريبة من المستوى الاسمي، لكنها عانت من نقص شديد في تغطية الأحمال المرتفعة. تؤكد النتائج أن تعريف زمن توفر الخصائص وتقييم عدم اليقين الشرطي عنصران أساسيان في صحة الاستنتاج العلمي.

الكلمات المفتاحية: التنبؤ بالطاقة الصناعية؛ تسرب البيانات؛ التقييم الزمني؛ التنبؤ المطابق؛ تقدير عدم اليقين؛ الغابة العش