CONFORMAL INFERENCE FOR ENERGY PRICE FORECASTING

Frederik V. Svane (s224766)
Marcus P. L. Christoffersen (s224750)

ABSTRACT

Energy price forecasting requires reliable uncertainty estimates for effective risk management. Standard conformal prediction provides guaranteed coverage but assumes exchangeable data, which is violated in time series settings. We apply online conformal methods, Online Gradient Descent (OGD) and Error-Quantified Conformal Inference (ECI), to English electricity spot price forecasting during the 2022 energy crisis, using both LSTM and ARIMAX models. Our results show that ECI achieves 5.6% narrower prediction intervals than OGD when paired with the LSTM model (101 vs 107 GBP/MWh average width) while maintaining near-target coverage (88.6% vs 90.2%). However, this advantage diminishes with weaker base models, confirming that conformal methods amplify rather than compensate for prediction quality. These findings suggest ECI is preferable when strong base models are available.

1. INTRODUCTION

Energy price forecasting, like other investment domains, benefits immensely from uncertainty estimates for proper risk management. Conformal prediction is able to provide these prediction intervals, with guaranteed coverage. Setting a miscoverage rate of $\alpha = 0.1$ means we expect 90% of true values to fall within our intervals. However, standard conformal methods assume exchangeable data, which is violated in time series, where observations are sequentially dependent/inexchangeable.

Extensions to standard conformal inference, such as Adaptive Conformal Inference (ACI) and Online Gradient Descent (OGD) [1], have been developed to address this problem of assumed exchangeability, by dynamically adjusting intervals based on binary coverage feedback. Error-Quantified Conformal Inference (ECI) [2] extends these methods further, by incorporating the *magnitude* of prediction errors, instead of mere binary miss/cover feedback.

In this project, we apply these conformal methods to English electricity spot price forecasting using 2022 hourly data with weather and consumption features. We implement two forecasting models, ARIMAX and LSTM, to evaluate how conformal methods perform across models of varying quality and to demonstrate the model-agnostic nature of conformal inference [3]. Specifically, we compare OGD and ECI for constructing prediction intervals. This allows us to explore how online conformal prediction performs on real energy data from the 2022 energy crisis, where the market experienced increased volatility and distribution shifts.
Our implementation is available at [4].

2. METHODS

2.1. Conformal Prediction

Conformal prediction constructs prediction intervals by comparing new observations to a calibration set [3]. Given a point prediction $\hat{y}_t$ and a true value y_t, we define a nonconformity score $s_t = |y_t - \hat{y}_t|$ measuring how "unusual" the observation is. A prediction interval is then constructed as $[\hat{y}_t - q_t, \hat{y}_t + q_t]$, where q_t is a threshold chosen such that the interval covers the true value with probability $1 - \alpha$.

Standard conformal prediction assumes exchangeability, meaning the observations can be reordered without affecting the underlying statistical relationships. This assumption obviously fails for time series data, where observations are strictly chronologically ordered.

Additionally, the underlying distribution may shift over time, meaning the statistical properties of prices in summer differ from those during winter, and likewise for crisis periods compared to stable periods. So a calibration set from the past may not represent future conditions, causing coverage guarantees to break down.

2.2. Online Conformal Methods

Online conformal methods address non-exchangeability by updating the threshold q_t dynamically based on observed coverage. The Online Gradient Descent (OGD) method uses the update rule

$$q_{t+1} = q_t + \eta \cdot (\mathrm{err}_t - \alpha) \tag{1}$$

where η is a learning rate and the binary variable $\mathrm{err}_t \in \{0, 1\}$ indicates whether the true value fell outside the interval. When coverage is too low ($\mathrm{err}_t = 1$ more often than α), the threshold increases, widening intervals. When coverage is too high, the threshold decreases, tightening intervals.

Error-Quantified Conformal Inference (ECI) extends OGD by incorporating the magnitude of prediction errors. The update rule becomes

$$q_{t+1} = q_t + \eta \cdot [(\text{err}_t - \alpha) + (s_t - q_t) \cdot f'(s_t - q_t)] \quad (2)$$

where s_t is the nonconformity score and f is a "shaping" function, such as a sigmoid or a Gaussian kernel. As such, the additional term $(s_t - q_t) \cdot f'(s_t - q_t)$ adjusts the update based on how far the score was from the threshold. The choice of shaping function comes with considerations: the sigmoid saturates for large deviations, giving extreme misses similar corrections to moderate ones. The Gaussian kernel goes further, actively dampening corrections for extreme deviations, treating them as potential outliers that should have less influence on the threshold.

We further evaluate two variants of ECI. ECI-cutoff only applies the error-quantification term when deviations exceed an adaptive threshold h_t computed from recent score ranges, otherwise reverting to binary feedback like OGD. ECI-integral replaces the single-step update with a weighted sum over all past observations, using exponentially decaying weights to emphasize recent data over past data.

2.3. Data

We use hourly electricity market data from England spanning February 1, 2022 to February 1, 2023, comprising 8,760 observations. The target variable is the day-ahead spot price (GBP/MWh), while the raw features include wind power forecasts from a wind farm near London (MW), temperature forecasts (°C), historical normal temperatures (°C), and timestamps.

This period coincides with the European energy crisis, characterized by significant price volatility and distribution shifts. These conditions stress-test conformal methods designed for non-stationary environments.

2.3.1. Feature engineering

With only a single year of data, capturing seasonal patterns directly is challenging. To address this, we engineer additional features from the available measurements.

From the timestamp, we derive hour of day (1–24), day of year (1–365), week number (1–52), and month (1–12). To represent annual seasonality without discontinuities, we encode day of year cyclically:

$$\text{day\_sin} = \sin\left(\frac{2\pi \cdot \text{day}}{365}\right), \quad \text{day\_cos} = \cos\left(\frac{2\pi \cdot \text{day}}{365}\right)$$

We also construct features based on energy market dynamics. Following WHO guidelines that recommend a minimum indoor temperature of 18°C [5], we define heating demand as $\max(0, 18 - \text{temperature forecast})$. We then multiply this by consumption forecasts, based on the hypothesis that cold weather combined with high demand creates price pressure.

Additionally, we compute temperature deviation from historical norms, based on the intuition that unusual weather may drive price volatility.

Finally, we include lagged spot price $(t - 1)$ as a feature, reflecting the realistic constraint that only past prices are observable at prediction time.

This yields 14 input features in total. The data is split 80/20 into training and test sets, preserving temporal order to ensure future data is not used during training.

2.4. Forecasting Models

We use two forecasting models to generate point predictions, representing deep learning and classical time series approaches.

2.4.1. Deep Learning Model

Sequential data like this calls for a recurrent architecture. Standard feedforward networks treat each input independently, failing to capture positional relationships within a sequence. They also require fixed input dimensions, making them unsuitable for variable-length sequences. Recurrent neural networks (RNNs) address both issues by processing inputs sequentially and maintaining a hidden state that carries information across time steps.

However, traditional RNNs suffer from vanishing or exploding gradients. During backpropagation, gradients in early layers are computed via the chain rule, resulting in a product across all time steps:

$$\frac{\partial \mathcal{L}}{\partial h_0} = \prod_{t=1}^{T} \frac{\partial h_t}{\partial h_{t-1}} \cdot \frac{\partial \mathcal{L}}{\partial h_T} \quad (3)$$

For long sequences, this product either explodes or vanishes. Ie, if the intermediate gradients are consistently greater than one, the product blows up. If they are consistently smaller than one, it goes to zero.

LSTMs address this through a gating mechanism. The cell state acts as a highway that allows gradients to flow across many time steps with minimal transformation. The forget gate controls what information to discard, the input gate controls what new information to store, and the output gate controls what to expose to the next layer. These gates use sigmoid activations to produce values in $[0, 1]$, enabling smooth gradient flow during training.

We considered adding attention mechanisms or moving to a full Transformer architecture, but opted against this for practical reasons. Both require substantially more data to train effectively, and our dataset spans only a single year. They also introduce computational overhead that slows down iteration during experimentation. Given these constraints, a standard

LSTM provides a reasonable balance between expressiveness and trainability.

Our implementation uses PyTorch and consists of two stacked LSTM layers with 64 hidden units each, followed by a linear output layer. The network processes sequences of 48 hourly observations (two days), with only the final hidden state passed to the output layer to produce a point prediction for the next hour.

We tune sequence length and number of training epochs via grid search. For sequence length, we tested windows ranging from 1 day (24 hours) to 14 days (336 hours), with 48 hours performing best. We use chronological cross-validation with an expanding window to prevent future information from leaking into training.

2.4.2. *Classical time series model*

For comparison with the deep learning approach, we implement an ARIMAX model (AutoRegressive Integrated Moving Average with eXogenous variables). ARIMAX combines autoregressive terms using the previous p observations, differencing of order d to achieve stationarity, moving average terms using the previous q forecast errors, and exogenous variables X. The full model is:

$$y_t = c + \sum_{i=1}^{p} \phi_i y_{t-i} + \sum_{j=1}^{q} \theta_j \epsilon_{t-j} + \beta^T X_t + \epsilon_t \qquad (4)$$

We use the auto_arima function from pmdarima to select the optimal order, constraining the search to $p, q \in [0, 5]$ and using the Bayesian Information Criterion. The search identified order (2, 1, 0) as optimal. We do not use seasonal ARIMA, as short-term seasonal patterns are already captured through our engineered time features, and the increased complexity and computational overhead is not justified given our limited data.

3. EXPERIMENTS & RESULTS

We conduct experiments to evaluate how well online conformal methods construct prediction intervals for energy price forecasting. Our focus is on comparing OGD and ECI across different forecasting models and understanding their performance.

3.1. Experimental Setup

We compare OGD and ECI on both ARIMAX and LSTM models. ECI is evaluated in three variants: basic, cutoff and integral each with sigmoid or Gaussian shaping functions. Performance is measured using coverage rate (the proportion of true values falling within prediction intervals), average interval width, avg median. The target coverage is 90%, corresponding to miscoverage rate $\alpha = 0.1$.

Hyperparameters are selected through random search over 10,000 configurations. For OGD, we sample learning rate $\eta \in [0.01, 10]$ and initial threshold $q_0 \in [5, 100]$. For ECI, we additionally sample sigmoid scaling parameter $c \in [0.1, 10]$ and select between sigmoid and Gaussian shaping functions. The cutoff variant of ECI uses a rolling window for adaptive thresholding, with window length sampled from $\{20, \ldots, 200\}$ and Gaussian bandwidth $h \in [0.3, 2.0]$. From the evaluated configurations, we select the one that minimizes average interval width while achieving coverage between 88% and 92%.

The dataset is split 80/20 meaning the training data spans February 1 to November 20, 2022 (7,026 observations), and test data spans November 20, 2022 to February 1, 2023 (1,757 observations). This split ensures the test period includes the winter months of the energy crisis, providing a test of how well the conformal methods handle distribution shifts.

3.2. Results

We present results comparing OGD and ECI variants across both forecasting models, examining both visual behavior and quantitative performance metrics.

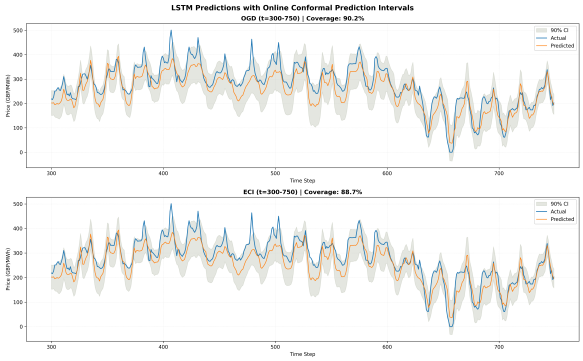

Fig. 1. LSTM results with Confidence Intervals for both OGD and ECI

Figure 1 shows LSTM results over timesteps 300-750. Both OGD and ECI track actual prices closely, with intervals widening during the volatile period around timesteps 400-600, then tightening during stable periods toward timestep 750. ECI intervals appear slightly narrower throughout, suggesting more adaptive behavior.

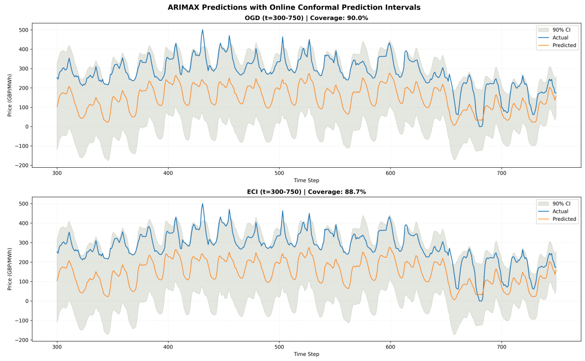

Fig. 2. ARIMAX results with Confidence Intervals for both OGD and ECI

Figure 2 shows ARIMAX results over timesteps 300-750. Both OGD and ECI produce consistently wide intervals throughout the test period, with larger prediction deviations during the high-volatility period around timesteps 400-600, reflecting the classical model's higher prediction uncertainty.

Table 1. Performance comparison of conformal methods. Coverage (%), average width, and median width.

Method /	ARIMAX			LSTM		
Model	Cov.	Avg.	Med.	Cov.	Avg.	Med.
	(%)	W.	W.	(%)	W.	W.
OGD	**90.0**	222	198	**90.2**	107	106
ECI	88.7	**219**	**197**	88.6	**101**	**101**
ECI-cut.	88.7	220	197	88.6	101	104
ECI-int.	86.0	220	196	82.2	92	97

OGD achieves coverage closer to the 90% target (90.0-90.2%) compared to ECI variants (86.0-88.7%). However, ECI produces narrower intervals, particularly with the LSTM model where basic ECI achieves 101 GBP/MWh compared to OGD's 107 GBP/MWh. This presents a practical tradeoff: OGD provides slightly more reliable coverage in this case, while ECI offers tighter intervals at the cost of slightly more frequent misses. By incorporating error magnitude rather than binary feedback, ECI adaptively tightens intervals during stable periods (visible in Figure 1 during the stable period toward timestep 750), making it the superior choice when base model quality is high.

The ARIMAX results reveal that interval width is primarily driven by base model quality. ARIMAX ECI shows roughly double the average width (219 GBP/MWh) compared to LSTM ECI (101 GBP/MWh) due to the model's larger prediction errors. When base predictions are poor, error-quantification provides minimal advantage over binary feedback, with ARIMAX methods differing by only 3 GBP/MWh. This underscores that conformal methods amplify base model quality rather than compensate for it.

The lower coverage of ECI may be partially explained by the choice of shaping function. Energy prices exhibit sudden spikes that, while meaningful market events, are treated as outliers by the shaping functions. The sigmoid function does not widen intervals aggressively for large deviations, and the Gaussian kernel actively dampens corrections for extreme errors. In a domain where spikes carry important information, this leads to insufficient interval widths and more frequent misses.

4. DISCUSSION

The primary limitation is base model quality. More training data spanning multiple years would enable better seasonal modeling and predictions, allowing conformal methods to construct tighter intervals.

The ECI variants show limitations in this volatile setting. ECI-cutoff provides minimal improvement because the adaptive threshold offers little benefit with high volatility. ECI-integral achieves the worst coverage and has $O(t)$ computational complexity. We did not extensively tune its hyperparameters due to poor initial results, suggesting past errors may carry limited information for this data.

The choice of shaping function affects performance. Both sigmoid and Gaussian functions dampen corrections for extreme deviations, but energy price spikes are meaningful events that should widen intervals. A linear shaping function would preserve corrections for large errors rather than dampening them. Finally, more sophisticated hyperparameter optimization such as Bayesian optimization could improve the coverage-width trade-off.

5. CONCLUSION

This project evaluated online conformal prediction methods for energy price forecasting during the 2022 European energy crisis. Our results demonstrate that Error-Quantified Conformal Inference (ECI) achieves 5.6% narrower intervals than Online Gradient Descent (OGD) when paired with high-quality base models (LSTM: 101 vs 107 GBP/MWh), while maintaining near-target coverage (88.6% vs 90.2%). However, this advantage diminishes with weaker prediction models, as can be seen by ARIMAX which showed only 1.4% width reduction between OGD and ECI, thereby confirming that conformal methods amplify rather than compensate for base model quality. While our single-year dataset limits seasonal modeling and test sequence length, these findings suggest ECI is the superior choice for uncertainty quantification when strong predictive models are available.

6. REFERENCES

[1] Isaac Gibbs and Emmanuel Candès, "Adaptive conformal inference under distribution shift," 2021.

[2] Junxi Wu, Dongjian Hu, Yajie Bao, Shu-Tao Xia, and Changliang Zou, "Error-quantified conformal inference for time series," 2025.

[3] Anastasios N. Angelopoulos and Stephen Bates, "A gentle introduction to conformal prediction and distribution-free uncertainty quantification," 2022.

[4] Frederik V. Svane and Marcus P. L. Christoffersen, "Conformal prediction for energy price forecasting," `https://github.com/FrederikvSvane/Conformal-Prediction-for-Energy-Price-Forecasting`, 2025.

[5] World Health Organization, *WHO Housing and Health Guidelines*, World Health Organization, Geneva, 2018.

DECLARATION OF USE OF GENERATIVE AI

This declaration **must** be filled out and included as the **final page** of the document. The questions apply to all parts of the work, including research, project writing, and coding.

- I/we have used generative AI tools: yes

If you answered *yes*, please complete the following sections. List the generative AI tools you have used:

- Claude by Anthropic AI
- Claude Code by Anthropic AI

Describe how the tools were used:

What did you use the tool(s) for?

We used the named AI tools for brainstorming- and evaluating ideas, proofreading and correcting parts of the text in the report, bugfixing and generating non-essential, utility code, such as plotting data with matplotlib.

At what stage(s) of the process did you use the tool(s)?

We used the named AI tools during the research-, implementation and proofreading stages of the process.

How did you use or incorporate the generated output?

Some generated output we simply read and responded to, in order to guide our thinking, in a similar way to what we would have done to a TA or professor. Some coding output, such as matplotlib plotting code, we ran directly. And grammatical corrections to sections in our report, we included in the report.