COMPOUNDING ENERGY LTD

Working Paper Series • CE-NOTE-2025-03

Training equals production

Why we never train on synthetic hindcasts, and what this buys in calibration drift

Dr Christopher T. M. Clack
Founder, Compounding Energy Ltd
christopher@compoundingenergy.com

August 2025

Abstract. Most operational energy-modelling stacks are trained against synthetic hindcasts: re-runs of the upstream weather model, the upstream price model, or the upstream load model on historical inputs, with the actual realised conditions hand-corrected back into the training data. The resulting calibration is a fit against a reconstruction, not against the data the model will encounter in production. Compounding Energy's position is opposite: every shadow forecast that customers see is itself the next training cycle's input data. Training and production share an artefact, not a copy. This note explains why this discipline matters, what it buys in calibration drift, and what it costs to maintain.

Keywords: model serving, training drift, hindcast, operational ML, energy forecasting

The hindcast problem

Operational energy forecasting models — prices, generation, demand — are calibrated against historical data. The standard pipeline takes a model class, fits it to a historical span ending some weeks before the present, validates it against a held-out span, and deploys the trained artefact into production. Periodically the cycle repeats: the historical span is extended forwards, the model is re-fitted, the new artefact is re-deployed.

The defect in the standard pipeline is that the historical span is almost always reconstructed. Weather inputs that the production model would have seen are replaced by the upstream forecast model's hindcast over the same period. Realised dispatch is replaced by the system operator's settlement reconciliation. Price formation is replaced by post-hoc cleared market data. The reconstructions are usually closer to truth than the original forecasts; the training data is therefore systematically smoother, less noisy, and more internally consistent than what the production model receives at inference time.

The consequence is that the trained model has learned to fit a version of reality the production model will never see. The gap is not statistical: the production inputs are a non-stationary, non-i.i.d. slice of a manifold the training data does not adequately cover. The production model degrades — silently, slowly, with no detectable failure mode — because the conditions it operates under drift away from the conditions it was calibrated on.

Our position

We commit to the discipline that the training data for cycle $k+1$ is exactly the production input stream that produced the shadow forecasts in cycle k. The shadow forecasts our customers see are not a polished version of our calibration data; they are the calibration data. The training cycle has no privileged hindcast access. The training set is a dated, immutable, append-only log of every input the production system received.

Three corollaries follow.

We never replace input with hindcast. If a weather model issued a 0600 Z forecast that turned out to be wrong by 4 m/s on a storm event, that forecast — not the corrected reanalysis — is the input to the next training cycle's calibration. The error is in the data because the error was in production; correcting it post-hoc would teach the model to assume corrections it does not have at inference.

We never use the system operator's settlement data as training input for forecasting models. Settlement data are reconciled and balanced; production data from the trading-floor view are not. Training on settlement leaks information into the calibration that the production model will never have access to. The price-forecasting model is trained on the within-day price history exactly as it appeared at the time, not on the cleaned export from the settlement system.

We never re-run a model on past data with a model version that did not exist at the time. If we deployed model v0.3 in February and v0.4 in May, the training data for v0.5 includes v0.4 production output for the May-onward window and v0.3 output for the February-to-May window, not v0.4 retroactively re-run over the entire period. The training-equals-production rule is observed across model versions as well as across time.

What this buys

Three benefits compound.

Calibration drift is bounded by realised drift. Because the training input distribution is exactly the production input distribution, the model's calibration residual is the actual drift it accumulates in service. There is no hidden drift attributable to a synthetic hindcast that has shifted from the production stream. When we report a forecast skill metric on shadow runs, the same metric on the next training cycle's validation split will be within sampling noise of it, by construction.

The validation split is operationally meaningful. In the standard pipeline, the validation split is a hindcast span that the model has not seen during training but has been reconstructed in the same way as the training span. In our pipeline, the validation split is the most recent month of actual production shadows. Performance on the validation split is the performance the customer would have seen in that month. There is no extra translation step from validation to deployment.

Customers can verify our claims. Every shadow run we produce is logged with the same input fingerprint that will appear in the next training cycle. Customers who maintain their own hold-out sets can demonstrate that our reported skill metrics on those sets match the metrics we would report internally. The calibration claim is reproducible end-to-end.

What this costs

The discipline imposes three constraints.

We cannot use synthetic data augmentation. Augmenting the training set with synthetic samples (counterfactual weather, perturbed market scenarios, synthetic regime shifts) is forbidden because those samples are not in the production input distribution. We give up some sample efficiency in exchange for the calibration guarantee.

We must instrument production exhaustively. Every input the production model receives must be logged in a form that survives the training cycle. Input fingerprints, version stamps, and timezone-disciplined timestamps are non-negotiable. The data engineering effort is materially greater than a standard pipeline.

We accept slower iteration. A new model variant cannot be calibrated on a synthetic backfill of the production input stream; it must be deployed in shadow mode for at least one full retraining cycle (typically 30–60 days) before it can be promoted to a customer-facing role. Aggressive iteration cycles that retrofit historical data through new architectures are not available to us.

Where this position came from

The position is a response to a specific failure mode we have repeatedly observed in commercial energy-forecasting tools. A vendor reports backtest skill metrics that are 20–40% better than the realised production performance the customer sees in their first month of shadow operation. The cause is almost always a training-production mismatch: the backtest was constructed against a hindcast that smoothed over the very features the production stream is most uncertain about.

The discipline is straightforward to articulate. It is operationally costly to maintain. The benefit is precisely the absence of a calibration-drift surprise. For a customer placing capital allocation decisions on the back of a model's output, that absence is the only signal worth paying for.

What this means for product evaluation

If you are evaluating a Compounding Energy product against an alternative, ask the alternative's vendor three questions:

1. What is the training input source: the production input stream as it appeared at the time, or a hindcast/reconstruction?
2. How is the validation split constructed: as a recent production-shadow window, or as a held-out historical hindcast span?
3. What guarantee can you give that the skill metrics in the published backtest will reproduce on the customer's first month of shadow operation, and what is the recourse if they do not?

We answer all three identically across every product in the Compounding Energy stack. CompoundVision, CEGridSight, CEForesight, and CESentinel all run the training-equals-production discipline. The CEAtlas product is an exception only in the trivial sense that its inputs are static (geospatial layers) and there is no streaming production input to retrain on; the discipline is simply vacuous there.

This is a Compounding Energy Ltd position paper. It is non-technical and intended for product-evaluation discussions with customers and investors. Comments to `christopher@compoundingenergy.com`.