WHEN DO HEAVY TAILS HELP? IDENTIFIABILITY LIMITS AND PREDICTIVE UNCERTAINTY IN LÉVY-DRIVEN NEURAL SDES FOR ELECTRICITY MARKETS

Anonymous authors
Paper under double-blind review

ABSTRACT

Electricity markets are increasingly shaped by renewable generation, whose intermittency introduces heavy-tailed price shocks that standard forecasting models neither capture nor bound. Existing approaches either fit econometric models with Gaussian assumptions or apply generic sequence architectures with no account of the dynamical limits that constrain predictability under jump-dominated conditions. We introduce LévySDE-Net, a neural stochastic differential equation in which drift and diffusion are parameterized by neural networks and driven by α-stable Lévy motion, trained via a data-fitting loss on the readout together with an adversarial term that shapes the learned diffusion's magnitude. We pair this model with phase-space reconstruction and maximal Lyapunov exponent estimation to characterize a market's intrinsic predictability horizon, treating positive exponents as finite-sample measures of trajectory divergence rather than evidence of deterministic chaos. Evaluated on Nord Pool electricity price areas, LévySDE-Net and its Gaussian-forced counterpart both substantially outperform econometric and recurrent baselines at short horizons, but heavy-tailed forcing confers no consistent point-forecast advantage over Gaussian forcing – a result we trace, via a synthetic identifiability analysis, to a genuine confound between the Lévy stability parameter and the diffusion scale under discretely observed data, rather than to an absence of dynamical difference between the two forcing mechanisms. We further show that this confound does not resolve under coarser observation, and if anything modestly sharpens, ruling out simple undersampling as the explanation. Instead, Lévy forcing's distinguishing empirical effect is on predictive uncertainty: it systematically widens ensemble spread and interval width relative to Gaussian forcing. In the wind-driven price area, we validate this directly against an independently estimated predictability horizon: iterative rollout shows Lévy- and Gaussian-forced ensembles diverging at a timescale consistent with, and if anything preceding, the area's estimated Lyapunov time. Our results indicate that heavy-tailed stochastic forcing's principal value in this setting lies not in point accuracy but in producing forecast uncertainty that reflects a market's underlying dynamical instability – and that claims of a robust point-accuracy advantage for heavy-tailed neural SDEs should be read against the identifiability limits we characterize here.

1 INTRODUCTION

Renewable generation has made electricity prices among the hardest financial series to forecast. Wind-driven oversupply, transmission-congestion spikes, and hydro-reservoir buffering each produce abrupt, heavy-tailed price shocks whose severity differs systematically across price areas rather than being a single property of "electricity prices" (1; 9). Forecast error in such markets grows quickly with horizon, and small perturbations in state can diverge rapidly even when the system is stochastic rather than chaotic (2; 3) – a natural test bed for a broader question in time-series forecasting: when does forecast error reflect a model's limits, and when does it reflect limits on predictability itself (4)?

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Recent work has made real progress on both sides of this problem. Neural ODEs and SDEs learn continuous-time dynamics with flexible drift and diffusion functions (14; 15), and SDE-Net adds an adversarial scheme that shapes the learned diffusion for uncertainty estimation (7). Classical jump-diffusion models capture electricity price spikes, but by fitting parametric jump processes directly to the price series rather than learning drift and diffusion from data (12). Only one line of work combines the two: (23) drive a neural SDE with α-stable noise and pair it with Lyapunov diagnostics on stock indices, but their claim that heavy-tailed forcing improves point forecasts rests on single-run comparisons, and Lyapunov time is used only to compare one model's accuracy inside and outside its own estimated horizon. It remains unclear whether heavy-tailed forcing helps at all once seed variability is accounted for, and what it changes if not accuracy.

We therefore ask two questions. **(Q1)** Does replacing Gaussian forcing with α-stable Lévy forcing improve point forecasts in electricity markets with different jump mechanisms? **(Q2)** If not, what does heavy-tailed forcing change, and is that change related to a market's intrinsic predictability horizon, measured by the Lyapunov time of the price dynamics? Here the Lyapunov exponent is read as a finite-sample measure of trajectory divergence in a stochastic system, not as evidence of low-dimensional chaos.

To answer them, we introduce LévySDE-Net, a neural SDE whose drift and diffusion are neural networks driven by α-stable Lévy motion, trained with an adversarial diffusion loss and no explicit likelihood, and compare it against a Gaussian counterpart identical except for the noise law ($\alpha = 2$), so any difference is attributable to the driver alone. Both models have 1,145 parameters, roughly $15\times$ fewer than our LSTM baseline. We evaluate on hourly day-ahead prices from three Nord Pool areas chosen for distinct jump mechanisms (DK1, SE4, NO4), with five seeds per model and per-seed Diebold–Mariano tests, and estimate each area's Lyapunov time by delay-embedding the price differences.

Our findings are as follows. Both neural SDEs outperform econometric baselines at every horizon tested, reducing MSE by roughly 14% at DK1 and 8–12% at SE4, but the Lévy and Gaussian variants differ by under 1% in relative MSE, with a sign that varies by horizon and seed, and accuracy is nearly invariant to α. What heavy-tailed forcing does change is the ensemble: at DK1 and SE4 it widens the 95% interval by about $2.3\times$ relative to the Gaussian model (NO4 pending). At DK1, iterative rollout to $3h$ shows the Lévy-to-Gaussian spread ratio crossing a fixed threshold at about 67 hours, before the independently estimated Lyapunov time of 114 hours. A synthetic identifiability study adds a caution for reading α physically: α is unbiased in expectation across 40-seed recovery experiments, but its errors correlate with the diffusion scale (Pearson r up to 0.41), so we treat α as a modelling mechanism rather than an estimated quantity.

Contributions. (1) We introduce LévySDE-Net, a $1{,}145$-parameter Lévy-driven neural SDE trained without a likelihood or SDE residual, with a matched Gaussian control isolating the effect of the noise law. **(2)** We test Q1 across three Nord Pool areas with per-seed Diebold–Mariano tests and find no robust point-forecast advantage for heavy-tailed forcing, despite a $\sim$14% MSE gain over econometric baselines at DK1 that both drivers share. **(3)** We characterise recovery of α on Lévy–Ornstein–Uhlenbeck data over five stability values and two sampling frequencies, showing α–σ error coupling that limits how α can be interpreted. **(4)** We show Lévy forcing changes predictive uncertainty rather than accuracy, widening ensemble spread by $\sim 2.3\times$ at DK1 and SE4, and compare its rollout behaviour with the Lyapunov time in the wind-driven DK1 area.

2 Related Work

Electricity price forecasting and jump-diffusion models. (10; 11) Existing work on jump-diffusion models for electricity prices typically fits parametric jump processes (e.g. compound Poisson or Lévy-driven models) directly to price series (12; 13), rather than learning drift and diffusion as flexible neural functions; we build on the latter, neural-SDE line of work instead.

Neural SDEs and continuous-time deep learning. Continuous-time neural architectures emerged from interpreting residual networks as discretizations of underlying differential equations, formalized by Neural ODEs (14). This framework was extended to stochastic settings through Neural SDEs, which incorporate diffusion-driven noise to model stochastic dynamics from data (15). Ex-

tensions to hybrid and discontinuous systems, such as Neural Jump SDEs, explicitly model both continuous evolution and abrupt transitions (16).

A central line of work develops *latent SDE* models, where stochastic dynamics evolve in a learned latent space and are trained via variational inference. Early approaches combine stochastic adjoint methods with variational objectives for scalable learning (17), while subsequent work emphasizes interpretable continuous-time representations of latent stochastic dynamics (18). More recent developments improve expressivity and scalability through structured latent representations and alternative training objectives that avoid expensive simulation-based inference (19; 20), as well as probabilistic frameworks for modeling uncertainty in sequential data (21).

In parallel, stochastic differential equations have become central to modern deep generative modeling. Score-based generative models reinterpret diffusion processes as SDEs and learn time-dependent score functions to model complex data distributions (22).

Adversarial and uncertainty-aware SDE training. Our training procedure builds directly on SDE-Net (7), which trains drift and diffusion networks via a data-fitting loss and an adversarial loss that shapes the diffusion network to distinguish in- from out-of-distribution inputs. We adapt this scheme to point regression under α-stable, rather than Gaussian, driving noise, repurposing the diffusion network's adversarially-shaped output to represent heavy-tailed forecast uncertainty rather than for OOD detection. (23) make a closely related adaptation in their LDE-Net model – also an α-stable Lévy-driven neural SDE with an adversarial scheme derived from SDE-Net, paired with the same phase-space reconstruction and Lyapunov diagnostics (Cao's E_1/E_2, the Wolf method) – applied to stock indices rather than electricity markets; the overlap is close enough that we initially adopted the name LDE-Net before renaming to LévySDE-Net to avoid conflation. Two gaps in their evaluation are central to our contribution: their claim that Lévy-driven forcing outperforms Gaussian SDE-Net on point accuracy rests on single-run estimates with no seed variability or significance testing – exactly what our synthetic identifiability analysis (Section 4.3) and per-seed Diebold–Mariano testing (Section 4.4) are designed to interrogate, and we find the opposite once seed-level variability is accounted for – and they use Lyapunov time only to compare one model's accuracy inside versus outside its own estimated horizon, rather than testing whether two forcing mechanisms diverge at a horizon consistent with that estimate, which our Experiment 4.8 does directly.

This adversarial scheme is a distinct design choice from physics-informed approaches that embed a discretized-SDE consistency residual in the training objective (24; 25; 26), requiring no loss-weighting hyperparameters ($\omega_D, \omega_F, \omega_B$) or curriculum-style annealing, and reducing optimization to two standard losses trained jointly throughout (Appendix E). Closest in spirit is (26), which applies an SDE-constrained Lévy-driven neural SDE under a predictability-aware framing to exchange-rate forecasting; we differ in domain (electricity markets, with their negative-price and renewable-driven jump structure), training scheme (adversarial rather than residual-constrained), and scope (adding the identifiability analysis of Section 4.3 and the rollout-versus-Lyapunov-time validation of Section 4.8, neither present in that work).

Limitations and positioning. Most existing neural SDE approaches rely on diffusion-based (Gaussian) stochastic forcing and therefore model dynamics through continuous perturbations driven by Wiener processes. While effective for smooth stochastic systems, such formulations are not well-suited to the heavy-tailed shocks and abrupt discontinuities commonly observed in electricity spot prices, particularly in renewable-dominated or congestion-constrained markets. Recent work has further highlighted challenges associated with stochastic noise estimation and instability in latent SDE training (27).

Building on this literature, we introduce a predictability-aware forecasting framework based on an α-stable Lévy driver, enabling the modeling of discontinuous and heavy-tailed stochastic dynamics in electricity markets. Beyond forecasting accuracy alone, we explicitly connect stochastic forcing to trajectory divergence through Lyapunov-based diagnostics, and – unlike much of the neural-SDE literature, which reports point-accuracy comparisons alone – test directly, via iterative rollout, whether an independently estimated Lyapunov time is empirically consistent with the point at which heavy-tailed and Gaussian forcing produce distinguishable forecasts.

Limitations and positioning. Most existing neural SDE approaches rely on diffusion-based (Gaussian) stochastic forcing and therefore model dynamics through continuous perturbations driven by Wiener processes. While effective for smooth stochastic systems, such formulations are not well-

suited to the heavy-tailed shocks and abrupt discontinuities commonly observed in electricity spot prices, particularly in renewable-dominated or congestion-constrained markets. Recent work has further highlighted challenges associated with stochastic noise estimation and instability in latent SDE training (27).

Building on this literature, we introduce a predictability-aware forecasting framework based on an α-stable Lévy driver, enabling the modeling of discontinuous and heavy-tailed stochastic dynamics in electricity markets. Beyond forecasting accuracy alone, we explicitly connect stochastic forcing to trajectory divergence through Lyapunov-based diagnostics, and – unlike much of the neural-SDE literature, which reports point-accuracy comparisons alone – test directly, via iterative rollout, whether an independently estimated Lyapunov time is empirically consistent with the point at which heavy-tailed and Gaussian forcing produce distinguishable forecasts.

3 METHOD

3.1 OVERVIEW: PREDICTABILITY-AWARE LÉVY NEURAL SDES

Electricity price forecasting is constrained not only by model capacity, but by properties intrinsic to the underlying market: renewable-driven intermittency, transmission congestion, and hydro buffering each impose a distinct severity and frequency of heavy-tailed price shocks. We propose a unified framework combining: (i) a Lévy-driven neural stochastic differential equation (LévySDE-Net), (ii) an adversarial training scheme that calibrates the diffusion network's magnitude without an explicit likelihood or residual objective, and (iii) Lyapunov-based diagnostics that estimate each market's practical predictability horizon.

Notation. We model a scalar price-difference process $X_t \in \mathbb{R}$ ($X_t = P_t - P_{t-1}$, Section 4) with drift and diffusion parameterisations $f_{\theta_f} : \mathbb{R}^d \to \mathbb{R}^d$, $g_{\theta_g} : \mathbb{R}^d \to \mathbb{R}$, acting on a d-dimensional delay-embedded input window (Section 3.4).

3.2 MODULE I: LÉVY-DRIVEN STOCHASTIC DYNAMICS

We model electricity price differences as a pure-jump Itô–Lévy semimartingale:

$$X_t = X_0 + \int_0^t f(X_s, s; \theta_f)\, ds + \int_0^t g(X_s, s; \theta_g)\, dL_s^\alpha, \tag{1}$$

where L_t^α is a symmetric α-stable Lévy process, $1 < \alpha \leq 2$ ($\alpha = 2$ the Gaussian limit). Unlike Brownian-driven neural SDEs, this formulation represents heavy-tailed forcing and discontinuous price movements directly, rather than smoothing them into Gaussian increments (11; 12; 13). The Lévy driver is a distributional model for non-Gaussian increments, not an explicit jump-detection mechanism. Full Lévy–Itô representation and CMS sampling details are given in Appendix F.

3.3 MODULE II: ADVERSARIAL DIFFUSION-CALIBRATION TRAINING

Rather than embedding a discretized-SDE consistency residual into the training objective – an approach with a well-known ill-posedness under α-stable forcing, since the population second moment of L_k is undefined for $\alpha < 2$ (28) – we adapt the adversarial training scheme of SDE-Net (7), which trains drift and diffusion jointly via two separate losses rather than a single composite residual.

Data-fitting loss. The downsampling and drift parameters are trained by mean-squared error between the readout $\hat{y} = \overline{\hat{X}_K}$ – the mean over the final SDE state's coordinates after $K = 6$ Euler steps – and the observed target:

$$L_{\text{data}}(\theta_h, \theta_f) = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2. \tag{2}$$

This loss alone anchors the point forecast; it carries no information about tail behaviour, which is instead shaped by the adversarial loss below.

Adversarial diffusion loss. The diffusion network is not trained to match a population moment of the driving noise. Instead, following (7), it is trained as a binary classifier distinguishing real training inputs from Gaussian-perturbed inputs $x' = x + 2\varepsilon$, $\varepsilon \sim \mathcal{N}(0, I)$:

$$L_{\text{diff}}(\theta_g) = \text{BCE}(g_{\theta_g}(h_\theta(x)), \text{real}) + \text{BCE}(g_{\theta_g}(h_\theta(x')), \text{fake}). \tag{3}$$

The diffusion network's sigmoid output is scaled by a fixed constant $\sigma = 0.5$ to produce the forward pass's per-step diffusion magnitude; this magnitude is what widens or narrows the ensemble at inference time – in the spirit of repeated stochastic forward passes as an uncertainty estimate (29; 30) – and it is here, not in the drift, that heavy-tailed forcing shapes forecast uncertainty.

Optimisation. L_{data} and L_{diff} are optimised jointly by two separate SGD optimisers, updated in alternation at every training step: $\text{lr} = 10^{-4}$, momentum 0.9, weight decay 5×10^{-4} for (θ_h, θ_f); $\text{lr} = 10^{-2}$, momentum 0.9, weight decay 5×10^{-4} for θ_g. Gradient norms for (θ_h, θ_f) are clipped at 5.0. We train full-batch for 100 epochs of 50 inner iterations each ($5{,}000$ gradient steps per optimiser). Unlike a composite residual objective, this scheme requires no loss-weighting coefficients and no curriculum-style annealing schedule: both losses are active, at full weight, from the first step.

Forward pass. The forward pass is itself the Euler–Maruyama discretisation of Eq. equation 1, with $\Delta t = 1/6$ and α-stable increments L_k sampled fresh at every step (31):

$$\hat{X}_{k+1} = \hat{X}_k + f(\hat{X}_k)\,\Delta t + \sigma\, g(\hat{X}_0)\,(\Delta t)^{1/\alpha}\, L_k. \tag{4}$$

Because this same forward pass produces both training predictions and the ensemble samples used at test time (Section 4), no separate sampling procedure is required for uncertainty estimation: an ensemble is simply N independent calls to Eq. equation 4.

3.4 Module III: Operational definition of instability

We define instability operationally as exponential divergence of nearby trajectories, quantified by the maximal Lyapunov exponent $\lambda_{\max}$, estimated with a Wolf-type procedure (2) on delay-embedded price-difference trajectories, with embedding dimension selected via Cao's E_1/E_2 criterion (8). In the proposed framework, instability emerges from the interaction between α-stable perturbations (Module I) and the nonlinear propagation dynamics learned by f_{θ_f}: heavy-tailed increments perturb the state, and the drift function determines how these perturbations evolve and amplify. The corresponding Lyapunov time

$$h = \frac{1}{\lambda_{\max}} \tag{5}$$

is an empirical predictability timescale, not a strict deterministic-forecasting bound.

Positive $\lambda_{\max}$ is not interpreted as evidence of low-dimensional deterministic chaos. Electricity price-difference dynamics are predominantly stochastic ($E_2 \approx 1$ under Cao's criterion) across all price areas tested (Section 4); positive exponents therefore reflect trajectory divergence from stochastic forcing and nonlinear amplification, not a chaotic attractor.

3.5 Scope and interpretation

This framework does not assume electricity price dynamics are governed by deterministic chaos. It treats predictability as a finite-horizon property, constrained by stochastic instability rather than model capacity alone. Short-horizon forecasting performance reflects local predictability; long-horizon degradation reflects intrinsic uncertainty amplification. The Lyapunov time is therefore an operational estimate of the horizon over which forecasts remain practically informative – a claim we test directly, rather than leave as interpretation alone, via iterative rollout in Section 4.8.

4 Experiments

4.1 Experimental design

Price areas and data. We evaluate LévySDE-Net on hourly Nord Pool day-ahead spot prices for three bidding zones representing distinct market regimes: DK1 (Western Denmark, 30 Aug

2015–30 Apr 2021, 49,677 observations), SE4 (Southern Sweden), and NO4 (a Norwegian bidding zone); SE4 and NO4 are drawn from the Open Power System Data (OPSD) package (ENTSO-E Transparency Platform, Jan 2015–Sep 2020). All series are hourly, in EUR/MWh, chronologically sorted, and de-duplicated by keeping the most recent observation at any repeated timestamp.

Transform and splits. Because spot prices can be zero or negative (1.18% of DK1's observations), the standard log-return transform is undefined; we work directly with price differences $r_t = P_t - P_{t-1}$ (32; 33). Each series is split chronologically 80/20 (train/test) and standardized using training-set statistics only. We report MSE, RMSE, MAE, and SMAPE (with ε-stabilization for near-zero differences); full preprocessing details are in Appendix A–C.

Baselines. We compare against a zero-change baseline, AR(1), GARCH(1,1), an LSTM (hidden dim 64, 16,705 parameters), and a Gaussian SDE-Net variant – identical to LévySDE-Net in architecture and training but with $\alpha = 2$ (standard Brownian motion) in place of the α-stable driver, isolating the effect of the noise law. Both SDE variants have 1,145 parameters on a $d = 8$ delay-embedded input window, $\sim 15\times$ smaller than the LSTM, so no advantage can be attributed to additional capacity. AR(1)/GARCH(1,1) are deterministic; LSTM and both SDE variants are reported as mean $\pm$ std over five seeds.

Table 1: Model classes used in the forecasting comparison.

Model	Purpose	Parameters	Variability
Zero baseline	No-change forecast	–	Deterministic
AR(1)	Linear econometric benchmark	2	Deterministic
GARCH(1,1)	Volatility-based financial benchmark	4	Deterministic
LSTM	Sequential neural baseline (hidden dim 64)	16,705	5 seeds
Gaussian SDE-Net	Neural SDE, Gaussian noise ($\alpha = 2$)	1,145	5 seeds
LévySDE-Net	Neural SDE, α-stable Lévy forcing ($\alpha = 1.5$)	1,145	5 seeds

4.2 Experiment 1: Tail-heaviness justification

Before comparing forecasting models, we establish independently whether each area's data exhibits heavy-tailed behaviour: for each area we fit Gaussian and α-stable distributions to the price-difference series and report the stability index $\hat{\alpha}$, excess kurtosis, the log-likelihood gap $\Delta L = L_{\text{stable}} - L_{\text{Gaussian}}$, and jump frequency $J = \#\{t : |r_t| > 3\sigma\}/N$.

Table 2: Tail-heaviness diagnostics by price area. Larger ΔL, higher kurtosis, and lower $\hat{\alpha}$ indicate stronger heavy-tailed / jump-dominant behaviour.

Area	$\hat{\alpha}$	Excess Kurtosis	ΔL (stable − Gaussian)	Jump Frequency J
NO4	0.758	1252.39	62175.11	0.93%
SE4	0.983	114.465	32564.11	1.99%
DK1	1.013	90.573	27963.74	1.65%

These diagnostics do not resolve to a single jump-dominance ordering: NO4 is most heavy-tailed on $\hat{\alpha}$, kurtosis, and ΔL, but least on jump frequency – an unusual combination consistent with a small number of extreme outliers rather than uniformly heavier tails, and warranting a data-quality check we have not yet completed. We therefore do not treat Table 2 as a settled ranking, and any comparison elsewhere in the paper that orders areas by jump-dominance should be read as directional only. A formal test of whether the Lévy-versus-Gaussian uncertainty effect (Experiment 6) scales with jump-dominance as measured here is left to future work (Section 5); we do not report such an analysis in this paper.

4.3 Experiment 2: Synthetic identifiability analysis

Before interpreting any empirical Lévy-versus-Gaussian comparison, we characterize how reliably α can be recovered from discrete-time data, using synthetic trajectories from a Lévy-driven Ornstein–Uhlenbeck process $dX_t = -\theta X_t\, dt + \sigma\, dL_t^{(\alpha)}$ with known (θ, σ, α), estimating α and σ *jointly* via

maximum likelihood (fixing σ independently would implicitly assume tail behaviour that need not hold).

Across 40 seeds per setting and $\alpha_{\text{true}} \in \{1.2, 1.5, 1.8, 1.9, 1.95\}$, recovery at hourly sampling is unbiased (95% CIs include zero throughout; Table 16) but imprecise: $\text{std}(\hat{\alpha})$ ranges from ≈ 0.027 to 0.046, with seed-level errors reaching ≈ 0.10 – largest at $\alpha_{\text{true}} = 1.5$, not near the Gaussian boundary. We trace this to a *compensation effect* between α and σ: seed-level errors in the two are correlated ($r = 0.41$ at $\alpha_{\text{true}} = 1.2$, falling to $r = 0.04$ at 1.95; Table 18) – strongest, not weakest, away from the Gaussian boundary, contrary to a naive "confound sharpens near $\alpha = 2$" intuition. Repeating at $720\times$ coarser (monthly-equivalent) sampling leaves recovery variance comparable but raises the compensation correlation at every tested α_{true} and introduces a small bias at $\alpha_{\text{true}} = 1.9$ ($p = 0.005$); coarser sampling therefore does not resolve the identifiability limit and may modestly worsen it (full tables in Appendix J).

These results indicate α is recoverable in expectation but not precisely identifiable from a single realization – a reference bound on the estimation problem itself, not a direct characterization of the model's own amortized α-head. This directly informs Experiment 3: LévySDE-Net ($\alpha = 1.5$) and Gaussian SDE-Net ($\alpha = 2.0$) differ by only $\Delta\alpha = 0.5$, comparable to the recovery error found here, so seed-dependent significance in that comparison is plausibly an estimation-theoretic artifact rather than evidence against a genuine dynamical difference between the two forcing mechanisms.

4.4 Experiment 3: One-step-ahead forecasting

We estimate DK1's effective predictability horizon via phase-space reconstruction and a Wolf-type Lyapunov exponent: $\tau = 1$, $m = 8$ (selected on an earlier percentage-return transformation, not yet independently re-checked on the price-difference series; Appendix G), giving $h \approx 114$ hours and motivating the multi-horizon extension below.

Table 3: One-step-ahead forecasting results, DK1. Mean $\pm$ std over five seeds where applicable. Lower is better.

Model	MSE	RMSE	MAE	SMAPE
Zero baseline	1.876899	1.370000	0.811798	166.863
AR(1)	1.874359	1.369072	0.813559	198.019
GARCH(1,1)	1.877010	1.370040	0.811826	166.543
LSTM	1.892150 $\pm$ 0.041401	1.375488 $\pm$ 0.015086	0.834416 $\pm$ 0.006280	**123.777** $\pm$ 0.909
Gaussian SDE-Net	1.618980 $\pm$ 0.012792	1.272384 $\pm$ 0.005035	0.743450 $\pm$ 0.003184	128.683 $\pm$ 1.855
LévySDE-Net	**1.617658** $\pm$ 0.015060	**1.271861** $\pm$ 0.005934	**0.742781** $\pm$ 0.003044	128.572 $\pm$ 1.962

Both SDE variants reduce MSE by $\approx$14% relative to the zero, AR(1), and GARCH baselines and outperform the LSTM on every metric but SMAPE, whose instability near zero-crossing series is illustrated by AR(1)'s poor SMAPE (198.0) despite competitive MSE. LévySDE-Net and Gaussian SDE-Net are nearly tied in aggregate, but per-seed Diebold–Mariano tests (Appendix H.3, Table 14) show two of five seeds significant at 5% ($p = 0.034, 0.015$), both favouring Gaussian – a modest, direction-consistent edge rather than seed noise, and, per Experiment 2, plausibly reflecting the identifiability limit rather than architecture. This is consistent with one-step forecasting operating in a locally stable regime where perturbations have limited time to propagate, so shared structural components dominate over tail behaviour; whether a robust separation emerges at longer horizons is examined next.

4.5 Experiment 4: Sensitivity to the Lévy stability parameter

We train LévySDE-Net at $\alpha \in \{1.2, 1.4, 1.5, 1.6, 1.8, 2.0\}$ with all else fixed, to isolate whether accuracy is driven by tail-heaviness or by the architecture itself. DK1 results (Appendix I) are essentially flat from $\alpha = 1.4$ through 2.0 (MSE varying by $<$0.1%), with a sharp departure at $\alpha = 1.2$ (MSE up $\approx$15% relative to $\alpha = 1.4$, seed-level std roughly tenfold higher) – a pattern more consistent with training instability under strongly heavy-tailed forcing than a smooth accuracy gradient in α, though this result has not yet been independently re-verified and should be treated cautiously.

4.6 EXPERIMENT 5: MULTI-HORIZON OUT-OF-SAMPLE FORECASTING

We extend to direct multi-horizon forecasts $q \in \{1, 2, 3, 4\}$ using the same chronological split.

Table 4: Multi-horizon forecasting results, DK1. $q = 1$ from Table 3.

Horizon q	Model	MSE	RMSE	MAE	SMAPE
6*1	Zero baseline	1.876899	1.370000	0.811798	166.863
	AR(1)	1.874359	1.369072	0.813559	198.019
	GARCH(1,1)	1.877010	1.370040	0.811826	166.543
	LSTM	1.892150 ± 0.041401	1.375488 ± 0.015086	0.834416 ± 0.006280	123.777 ± 0.909
	Gaussian SDE-Net	1.618980 ± 0.012792	1.272384 ± 0.005035	0.743450 ± 0.003184	128.683 ± 1.855
	LévySDE-Net	1.617658 ± 0.015060	1.271861 ± 0.005934	0.742781 ± 0.003044	128.572 ± 1.962
6*2	Zero baseline	1.876244	1.369760	0.811588	166.860
	AR(1)	1.873724	1.368840	0.813353	198.019
	GARCH(1,1)	1.876354	1.369801	0.811616	166.540
	LSTM	1.804213 ± 0.030963	1.343170 ± 0.011526	0.847947 ± 0.010458	129.169 ± 1.123
	Gaussian SDE-Net	1.704450 ± 0.019017	1.305530 ± 0.007287	0.785151 ± 0.003198	139.876 ± 2.322
	LévySDE-Net	1.707247 ± 0.018937	1.306601 ± 0.007253	0.785069 ± 0.003265	140.066 ± 2.415
6*3	Zero baseline	1.876385	1.369812	0.811601	166.857
	AR(1)	1.873864	1.368891	0.813368	198.018
	GARCH(1,1)	1.876496	1.369852	0.811628	166.536
	LSTM	1.941750 ± 0.018615	1.393454 ± 0.006690	0.881194 ± 0.007239	132.703 ± 1.028
	Gaussian SDE-Net	1.703679 ± 0.013833	1.305242 ± 0.005299	0.788436 ± 0.003122	141.519 ± 1.680
	LévySDE-Net	1.706499 ± 0.014083	1.306322 ± 0.005397	0.788177 ± 0.002837	141.737 ± 1.623
6*4	Zero baseline	1.876088	1.369704	0.811461	166.854
	AR(1)	1.873546	1.368775	0.813224	198.018
	GARCH(1,1)	1.876199	1.369744	0.811489	166.534
	LSTM	1.956042 ± 0.010961	1.398581 ± 0.003925	0.876885 ± 0.002209	132.621 ± 0.727
	Gaussian SDE-Net	1.712222 ± 0.008908	1.308516 ± 0.003405	0.788993 ± 0.003366	141.842 ± 1.212
	LévySDE-Net	1.714752 ± 0.009716	1.309481 ± 0.003711	0.788785 ± 0.003776	142.060 ± 1.107

The econometric baselines stay essentially flat across all four horizons, gaining no information from a longer window. Both SDE variants substantially outperform them at every horizon (≈9–14% MSE reduction) and degrade in a specific, non-gradual pattern: a single step from $q = 1$ to $q = 2$ (MSE ≈1.62 to ≈1.70), then essentially flat through $q = 4$ – a short transition in effective predictability rather than continuous widening. The LSTM instead degrades non-monotonically, competitive at $q = 2$ but worst of all models by $q = 4$. The Lévy-versus-Gaussian gap stays small and sign-inconsistent across horizons (LévySDE-Net ahead at $q = 1$; Gaussian ahead at $q = 2$–4), in all cases under 1% relative MSE. Per-seed Diebold–Mariano tests between the two forcing mechanisms (Appendix H.3, Table 14) show this gap is horizon-dependent: two of five seeds are individually significant at $q = 1$ and again at $q = 2$ (both favouring Gaussian), but none are significant at $q = 3$ or $q = 4$. The modest Gaussian-favouring edge visible at short horizons is therefore not a stable, horizon-independent effect – it disappears by $q = 3$, consistent with the identifiability limit of Experiment 2 rather than a robust structural advantage for either forcing mechanism at longer horizons.

4.7 EXPERIMENT 6: PREDICTIVE UNCERTAINTY ANALYSIS

Since point-forecast accuracy is comparable between forcing mechanisms, we test whether Lévy forcing's distinguishing effect is instead on predictive uncertainty, generating an ensemble of stochastic forward passes per test point (size not yet justified by a convergence check) and computing ensemble spread and 95% interval width at $q \in \{1, 2, 4\}$ for DK1 and SE4.

Table 5: Predictive uncertainty by area and forcing mechanism (95% interval width).

Area	Model	Avg. Ensemble Spread	Width ($q = 1$)	Width ($q = 2$)	Width ($q = 4$)
2*DK1	Gaussian SDE-Net	0.0071	0.0274 ± 0.0014	0.0275 ± 0.0018	0.0270 ± 0.0023
	LévySDE-Net	0.0325	0.0634 ± 0.0033	0.0636 ± 0.0044	0.0627 ± 0.0052
2*SE4	Gaussian SDE-Net	0.0054	0.0206 ± 0.0016	0.0210 ± 0.0021	0.0211 ± 0.0023
	LévySDE-Net	0.0260	0.0473 ± 0.0036	0.0483 ± 0.0048	0.0482 ± 0.0052

At both areas, LévySDE-Net produces substantially wider intervals and greater spread than Gaussian SDE-Net, with a strikingly consistent width ratio ($2.31\times$ at DK1, $2.30\times$ at SE4 at $q = 1$) despite the areas' different absolute uncertainty scales – suggesting the effect is a stable architectural property rather than area-specific. We flag one tension rather than resolve it: SE4 shows lower absolute spread than DK1 at both models despite scoring higher on every diagnostic in Table 2, which cuts against treating DK1 as straightforwardly more jump-dominant than SE4.

4.8 Experiment 7: Iterative rollout versus the estimated Lyapunov time (DK1)

To validate the predictability-horizon interpretation directly, we evaluate DK1 under iterative rollout well beyond its estimated Lyapunov time $h \approx 114$h ($\tau = 1, m = 8$), testing whether the two forcing mechanisms diverge at a horizon consistent with that estimate. We define the *rollout divergence horizon* as the first horizon at which the LévySDE-Net/Gaussian ensemble-spread ratio $\rho(t)$ exceeds $\rho^\star = 4.0$ (a threshold not yet justified on independent grounds), interpolating linearly if the crossing falls between evaluated horizons.

Table 6: Ensemble spread (mean $\pm$ std) by rollout horizon, DK1, relative to $h \approx 114$h.

	$h/2$ (57h)	h (114h)	$3h$ (344h)
LévySDE-Net	0.03694 ± 0.00502	0.03829 ± 0.00563	0.04152 ± 0.00624
Gaussian SDE-Net	0.00928 ± 0.00078	0.00934 ± 0.00079	0.00952 ± 0.00082
Ratio $\rho(t)$	3.98	4.10	4.36

LévySDE-Net's spread and its variability both grow across this range, while the Gaussian baseline stays flat and tight; $\rho(t)$ rises from 3.98 to 4.36, crossing $\rho^\star$ between $h/2$ and h and placing the rollout divergence horizon at ≈67 hours – *before* the estimated h, with $\rho(t)$ continuing to climb rather than plateau (Table 7).

Table 7: Estimated Lyapunov time vs. empirical rollout divergence horizon, DK1.

Area	Estimated Lyapunov Time (hrs)	Rollout Divergence Horizon (hrs)	Consistent?
DK1	114	67	Divergence precedes estimated Lyapunov time

This is consistent with h marking a meaningful, if approximate, predictability boundary for DK1. We treat it as a targeted confirmation on the one area tested, not as evidence that the relationship holds uniformly across areas of differing jump-dominance; testing that is left to future work (Section 5).

5 Conclusions, limitations, and future work

LévySDE-Net provides a stable, lightweight framework for electricity price dynamics under heavy-tailed stochastic forcing, with the Lyapunov-based analysis offering an operational interpretation of forecast degradation as stochastic instability rather than model misspecification. Most predictive gain comes from the neural SDE architecture itself: Lévy and Gaussian forcing achieve comparable point-forecast accuracy (Experiments 3, 5), while Lévy forcing's distinguishing contribution is systematically wider predictive intervals and ensemble spread (Experiment 6) – suggesting its primary value lies in uncertainty representation rather than point accuracy. Identifiability limitations bearing on α and the scope of the Lyapunov-based interpretation are discussed further in Appendix L.2 and Appendix L.3.

Several items remain open. The uncertainty analysis is not yet paired with a calibration study, since interval width depends on fixed diffusion-scale constants rather than the target series' own variance. The uncertainty analysis has not yet been run for NO4, and the rollout validation has not yet been run for SE4 or NO4, so no cross-area jump-dominance-scaling claim is made or reported here. The ensemble size used throughout has not been justified by a convergence check, and Experiment 2's identifiability analysis characterizes a standard maximum-likelihood estimator rather than the model's own amortized α-head – characterizing the latter directly remains open.

AI USE STATEMENT

In this work, we used generative AI tools for editing and polishing manuscript text (grammar, clarity, structural consistency across sections). We have not used generative AI tools to generate research ideas, to write the paper's core scientific claims or interpretations independently of author direction, or to conduct literature review or citation discovery; these were performed by the authors. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work uses only publicly available electricity market data (Nord Pool day-ahead spot prices; the OPSD time-series package sourced from the ENTSO-E Transparency Platform) and does not involve human subjects, personally identifiable information, or data requiring IRB review. We are not aware of any conflicts of interest or sponsorship arrangements bearing on this work.

The main ethical consideration we identify is the potential for dual use: forecasting and uncertainty-quantification tools of the kind proposed here could in principle inform strategic bidding behaviour that reduces rather than improves efficiency in electricity markets, particularly where generation is already concentrated, and over-reliance on model outputs in dispatch or trading decisions in what is safety-critical infrastructure carries downstream risk if deployed without appropriate human oversight and validation on conditions (e.g. extreme scarcity events) not represented in the training data. We discuss these considerations, along with the model's identifiability and single-area validation limitations relevant to responsible deployment, in Appendix L.

REPRODUCIBILITY STATEMENT

We take several steps to support reproducibility. Anonymized source code covering data preprocessing, model training (LévySDE-Net and the Gaussian SDE-Net baseline), one-step and multi-horizon forecasting evaluation, Diebold–Mariano significance testing, the α-sensitivity sweep, phase-space reconstruction and Lyapunov exponent estimation, and the synthetic identifiability recovery experiments is available at `https://anonymous.4open.science/r/levy-sde-electricity-4CF9/`. Full experimental details – data sources and preprocessing (Appendix A), model architecture, hyperparameters, and the training procedure (Appendix D, Appendix E), evaluation metrics and the Diebold–Mariano test statistic (Appendix C), phase-space reconstruction and Lyapunov exponent estimation (Appendix G), and the synthetic recovery experiment's data-generating process and estimation procedure (Appendix J) – are provided so that all reported results can be independently verified. All neural models are trained with fixed random seeds across Python, NumPy, and PyTorch, and with deterministic algorithms enabled throughout (Appendix K), so that results are reproducible run to run on identical hardware and software versions.

REFERENCES

K. A. Kasianova. Electricity spot price dynamics comparison in the European and Siberian price zones of Russia using a stochastic volatility model. *Economics and Mathematical Methods*, 60(1), 2024. URL `https://api.semanticscholar.org/CorpusID:272814047`.

Alan Wolf, Jack B. Swift, Harry L. Swinney, and John A. Vastano. Determining Lyapunov exponents from a time series. *Physica D: Nonlinear Phenomena*, 16:285–317, 1985. URL `https://api.semanticscholar.org/CorpusID:14411384`.

Qiwei Yao and Howell Tong. Quantifying the influence of initial values on nonlinear prediction. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 56:701–725, 1994. URL `https://api.semanticscholar.org/CorpusID:122073384`.

Lin Zhao, Jian Dong, Ruo-Jing Chen, Yifeng Wang, Yicheng Jin, and Yi Zhao. Perceiving unpredictability for new energy power and electricity consumption forecasting. *Entropy*, 28, 2026. URL `https://api.semanticscholar.org/CorpusID:284563429`.

Dragutin T. Mihailović, Slavica Malinović-Milićević, Jeong-Hoon Hanc, and Vijay Singh. Complexity and chaotic behavior of the U.S. rivers and estimation of their prediction horizon. 2023. URL https://api.semanticscholar.org/CorpusID:256390244.

Vignesh Gopakumar, Ander Gray, Daniel Giles, Lorenzo Zanisi, Matt J. Kusner, Timo Betcke, S. Pamela, and Marc Peter Deisenroth. Learning physical operators using neural operators. *arXiv preprint arXiv:2602.23113*, 2026. URL https://api.semanticscholar.org/CorpusID:286082727.

Lingkai Kong, Jimeng Sun, and Chao Zhang. SDE-Net: equipping deep neural networks with uncertainty estimates. In *International Conference on Machine Learning*, 2020. URL https://api.semanticscholar.org/CorpusID:221082095.

Liang-Yue Cao. Practical method for determining the minimum embedding dimension of a scalar time series. *Physica D: Nonlinear Phenomena*, 110:43–50, 1997. URL https://api.semanticscholar.org/CorpusID:121713604.

Rafał Weron. Electricity price forecasting: a review of the state-of-the-art with a look into the future. *International Journal of Forecasting*, 30(4):1030–1081, 2014. https://doi.org/10.1016/j.ijforecast.2014.08.008.

Xia Chen. Advanced approaches for electricity market price forecasting. 2010. URL https://api.semanticscholar.org/CorpusID:150851905.

Svetlana Borovkova and Ferry Jaya Permana. Modelling electricity prices by the potential jump-diffusion. In A. N. Shiryaev, M. R. Grossinho, P. E. Oliveira, and M. L. Esquível, editors, *Stochastic Finance*, pages 239–263. Springer, Boston, MA, 2006. URL https://api.semanticscholar.org/CorpusID:150530861.

Álvaro Cartea and Marcelo G. Figueroa. Pricing in electricity markets: a mean reverting jump diffusion model with seasonality. *Applied Mathematical Finance*, 12:313–335, 2005. URL https://api.semanticscholar.org/CorpusID:54048471.

Thilo Meyer-Brandis and Peter Tankov. Multi-factor jump-diffusion models of electricity prices. *International Journal of Theoretical and Applied Finance*, 11:503–528, 2008. URL https://api.semanticscholar.org/CorpusID:37886497.

Tian Qi Chen, Yulia Rubanova, Jesse Bettencourt, and David Kristjanson Duvenaud. Neural ordinary differential equations. *arXiv preprint arXiv:1806.07366*, 2018. URL https://api.semanticscholar.org/CorpusID:49310446.

Xuanqing Liu, Tesi Xiao, Si Si, Qin Cao, Sanjiv Kumar, and Cho-Jui Hsieh. Neural SDE: stabilizing neural ODE networks with stochastic noise. *arXiv preprint arXiv:1906.02355*, 2019. URL https://api.semanticscholar.org/CorpusID:174802756.

Junteng Jia and Austin R. Benson. Neural jump stochastic differential equations. *arXiv preprint arXiv:1905.10403*, 2019. URL https://api.semanticscholar.org/CorpusID:166228516.

Xuechen Li, Ting-Kam Leonard Wong, Ricky T. Q. Chen, and David Kristjanson Duvenaud. Scalable gradients for stochastic differential equations. In *International Conference on Artificial Intelligence and Statistics*, 2020. URL https://api.semanticscholar.org/CorpusID:209862121.

Lea Duncker, Gergo Bohner, Julien Boussard, and Maneesh Sahani. Learning interpretable continuous-time models of latent stochastic dynamical systems. In *International Conference on Machine Learning*, 2019. URL https://api.semanticscholar.org/CorpusID:60441161.

Sebastian Zeng, Florian Graf, and Roland Kwitt. Latent SDEs on homogeneous spaces. *arXiv preprint arXiv:2306.16248*, 2023.

Grigory Bartosh, Dmitry Vetrov, and Christian A. Naesseth. SDE matching: scalable and simulation-free training of latent stochastic differential equations. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*. PMLR, 2025. URL https://arxiv.org/abs/2502.02472.

James Rice. Stochastic deep learning: a probabilistic framework for modeling uncertainty in structured temporal data. *arXiv preprint arXiv:2601.05227*, 2026. URL https://api.semanticscholar.org/CorpusID:284544684.

Yang Song, Jascha Narain Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations (ICLR)*, 2021. URL https://api.semanticscholar.org/CorpusID:227209335.

Luxuan Yang, Ting Gao, Yubin Lu, Jinqiao Duan, and Tao Liu. Neural network stochastic differential equation models with applications to financial data forecasting. *Applied Mathematical Modelling*, 115:279–299, 2023. URL https://api.semanticscholar.org/CorpusID:250311878.

Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

Liu Yang, Xuhui Meng, and George Em Karniadakis. B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data. *Journal of Computational Physics*, 425:109913, 2021.

N'Adoi Aboagye and Saralees Nadarajah. SDE-constrained Lévy-driven neural SDEs for predictability-aware exchange rate forecasting. *Journal of Risk and Financial Management*, 19(6):432, 2026. URL https://doi.org/10.3390/jrfm19060432.

Linus Heck, Maximilian Gelbrecht, Michael T. Schaub, and Niklas Boers. Improving the noise estimation of latent neural stochastic differential equations. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 35(6), 2024. URL https://api.semanticscholar.org/CorpusID:274982620.

Gennady Samorodnitsky and Murad S. Taqqu. *Stable Non-Gaussian Random Processes: Stochastic Models with Infinite Variance*. Chapman & Hall, New York, 1994.

Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian approximation: representing model uncertainty in deep learning. In *International Conference on Machine Learning*, 2015. URL https://api.semanticscholar.org/CorpusID:160705.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, 2016. URL https://api.semanticscholar.org/CorpusID:6294674.

John M. Chambers, Colin L. Mallows, and Bart W. Stuck. A method for simulating stable random variables. *Journal of the American Statistical Association*, 71(354):340–344, 1976. URL https://api.semanticscholar.org/CorpusID:121641397.

Stefan Schneider. Power spot price models with negative prices. *The Journal of Energy Markets*, 4(4):77–102, 2011.

Enzo Fanone, Andrea Gamba, and Marcel Prokopczuk. The case of negative day-ahead electricity prices. *Energy Economics*, 35:22–34, 2013. https://doi.org/10.1016/j.eneco.2011.12.006.

Jesus Lago, Grzegorz Marcjasz, Bart De Schutter, and Rafał Weron. Forecasting day-ahead electricity prices: a review of state-of-the-art algorithms, best practices and an open-access benchmark. *Applied Energy*, 293:116983, 2021. https://doi.org/10.1016/j.apenergy.2021.116983.

A Data and Preprocessing

We use hourly Nordpool electricity spot price series (EUR/MWh) for three price areas: DK1, SE4, and NO4. Because electricity spot prices P_t can be zero or negative – arising during periods of oversupply, when producers pay to offload excess generation – the standard log-return transform $\log(P_t/P_{t-1})$ is undefined or numerically unstable for this series. We therefore work directly with price differences,

$$r_t = P_t - P_{t-1},$$

which are well-defined for all values in the series and are the standard transform used for electricity spot price modeling for this reason.

All series are aligned to a common hourly calendar (`HourUTC`). Rows with unparseable prices or timestamps are dropped; duplicate timestamps are resolved by keeping the last observation.

Each series is standardized to zero mean and unit variance using statistics estimated from the training window only, independently per price area. The resulting transformation is applied unchanged to the test set to prevent look-ahead bias and ensure consistent out-of-sample evaluation.

B Data Splits and Forecasting Setup

All experiments use strictly chronological train–test splits to avoid look-ahead bias. Each price area's series is divided into 80% training data and 20% test data, with the most recent 20% of observations held out for evaluation.

We evaluate both one-step-ahead and multi-horizon forecasting performance. Inputs are constructed via delay-coordinate phase-space reconstruction (Appendix G): for embedding dimension $d = 8$ and delay $\tau = 1$, the input vector at time t_i is

$$\mathbf{X}(t_i) = \left[r_{t_i}, r_{t_i-\tau}, \ldots, r_{t_i-(d-1)\tau}\right],$$

and for a forecasting horizon T, the target is $r_{t_i+T+(d-1)\tau}$.

Multi-horizon forecasts are evaluated across the set of horizons

$$T \in \{1, 2, 3, 4\}.$$

C Evaluation Metrics and Statistical Testing

Forecast accuracy is evaluated using standard pointwise error metrics:

- Mean Squared Error (MSE)
- Root Mean Squared Error (RMSE)
- Mean Absolute Error (MAE)
- Symmetric Mean Absolute Percentage Error (SMAPE), with a small ε-stabilization term in the denominator to avoid numerical instability for near-zero price differences.

These metrics are computed on out-of-sample forecasts and averaged over the test set. For neural SDE-based models, results are reported as mean $\pm$ standard deviation across five random seeds; deterministic baselines (Zero, AR(1), GARCH(1,1)) are reported without variability.

C.1 Diebold–Mariano Test

To assess statistical significance of forecast differences, we employ the Diebold–Mariano (DM) test, which evaluates the null hypothesis of equal predictive accuracy between two competing models.

Let $e_{1,t}$ and $e_{2,t}$ denote forecast errors from two models at time t. The DM test is based on the loss differential

$$d_t = |e_{1,t}|^p - |e_{2,t}|^p,$$

using $p = 2$ (squared-error loss). The test statistic is

$$\text{DM} = \frac{\bar{d}}{\sqrt{\hat{\sigma}_d^2/T}},$$

where $\bar{d}$ is the sample mean of d_t, T is the number of observations, and $\hat{\sigma}_d^2$ is estimated as

$$\hat{\sigma}_d^2 = \gamma_0 + 2\sum_{k=1}^{h-1}\gamma_k, \qquad \gamma_k = \frac{1}{T}\sum_{t=1}^{T-k}(d_t - \bar{d})(d_{t+k} - \bar{d}),$$

a Newey-West-style long-run variance estimator with truncation lag $h - 1$. A small-sample correction (Harvey–Leybourne–Newbold) is applied to the resulting statistic, and two-sided p-values are computed from the $t(T - 1)$ reference distribution.

D Experimental Details

This section provides detailed information about the experimental setup, preprocessing, model configurations, hyperparameters, and evaluation protocols for the experiments presented in the main text. Implementations are available at `https://anonymous.4open.science/r/levy-sde-electricity-4CF9/`.

D.1 General Setup

Unless otherwise specified, all experiments use the following setup:

- **Framework:** PyTorch.
- **Data:** Hourly Nordpool electricity spot price series for three price areas: DK1, SE4, and NO4.
- **Transformation:** Raw spot prices are transformed into price differences, $r_t = P_t - P_{t-1}$ (Appendix A).
- **Scaling:** Standardization to zero mean and unit variance, fitted only on the training split, applied independently per area.
- **Split:** Chronological 80% training and 20% test split.
- **Input construction:** Delay-coordinate phase-space reconstruction with embedding dimension $d = 8$ and delay $\tau = 1$ (Appendix B, Appendix G).
- **Forecast horizons:** $T \in \{1, 2, 3, 4\}$ for multi-horizon evaluation.
- **Seeds:** Neural models are evaluated over five random seeds: $\{0, 1, 2, 3, 4\}$.
- **Metrics:** MSE, RMSE, MAE, and SMAPE (with ε-stabilization for near-zero price differences).

D.2 LévySDE-Net Configuration

The proposed LévySDE-Net models the electricity price-difference process using a neural stochastic differential equation with Lévy-driven noise, following the SDE-Net architecture of Kong et al. (2020) adapted for point regression rather than out-of-distribution detection.

Architecture.

- **Downsampling layer:** A single linear layer, seq_len $\to$ seq_len.
- **Drift network:** A single linear layer, seq_len $\to$ seq_len, followed by ReLU.
- **Diffusion network:** Linear(seq_len $\to$ 100) $\to$ ReLU $\to$ Linear(100 $\to$ 1) $\to$ sigmoid.
- **Layer depth:** 6 (i.e. 6 Euler steps per forward pass, $\Delta t = 1/6$).
- **Diffusion scale:** the diffusion network's sigmoid output (bounded in $[0, 1]$) is scaled by a *fixed* constant $\sigma = 0.5$; the diffusion scale itself is not a free, tuned hyperparameter.

- **Default Lévy stability parameter:** $\alpha = 1.5$; Gaussian SDE-Net baseline uses $\alpha = 2.0$ (the exact Gaussian limit), with skewness $\beta = 0$ (symmetric driver) throughout.
- **Lévy scale:** 0.1 (the noise-generating distribution's scale parameter, distinct from σ above).

No separate trajectory-noise or physics-loss residual clip exists – neither the forward-pass noise nor any residual term is clipped in the implementation.

Training procedure. Training uses *two* separate SGD optimizers, following the original SDE-Net's adversarial training scheme:

- **Optimizer F** (downsampling layer + drift network): SGD, lr $= 10^{-4}$, momentum 0.9, weight decay 5×10^{-4}.
- **Optimizer G** (diffusion network): SGD, lr $= 10^{-2}$, momentum 0.9, weight decay 5×10^{-4}.
- **Gradient clipping:** max norm 5.0, applied only to the downsampling and drift parameters (optimizer F).
- **Batch size:** full-batch (the entire training split is used as a single batch each iteration).
- **Epochs / inner iterations:** 100 epochs $\times$ 50 inner iterations per epoch (5,000 total gradient steps per optimizer, per training run).

The training objective is *not* a physics-informed residual loss. It combines two separate losses, optimized in alternation within each inner iteration:

1. **Data loss** (optimizer F): mean-squared error between the target and the readout $\hat{y} = \overline{\text{out}}$, the mean over the final SDE state's coordinates after the full forward pass:

$$L_{\text{data}} = \frac{1}{n} \sum_i (y_i - \hat{y}_i)^2.$$

2. **Adversarial diffusion loss** (optimizer G): a binary cross-entropy loss that trains the diffusion network to distinguish real training inputs (label "real") from Gaussian-perturbed inputs $x' = x + 2\varepsilon$, $\varepsilon \sim \mathcal{N}(0, 1)$ (label "fake"), following the original SDE-Net's out-of-distribution-detection training scheme, here repurposed to shape the learned diffusion magnitude rather than to detect OOD inputs at test time:

$$L_{\text{diffusion}} = \text{BCE}(\hat{p}(x), \text{real}) + \text{BCE}(\hat{p}(x'), \text{fake}).$$

D.3 Baseline Configurations

The following baselines are used for comparison:

- **Zero baseline:** Predicts the last training-set value (constant forecast).
- **AR(1):** First-order autoregressive model fitted on the training segment via `statsmodels`.
- **GARCH(1,1):** Conditional volatility model (constant mean, GARCH(1,1) variance), fitted by maximum likelihood via the `arch` package; series rescaled by $\times 100$ for optimizer stability and rescaled back after forecasting.
- **LSTM:** Single-layer LSTM, hidden dimension 64, linear readout on the final hidden state; trained with Adam (lr $= 10^{-3}$), 100 epochs, batch size 64, MSE loss.
- **Gaussian SDE-Net:** Identical architecture and training procedure to LévySDE-Net (above), with $\alpha = 2.0$.

All baselines use identical preprocessing and train/test splits. For multi-horizon experiments, each model is retrained independently at each horizon $T \in \{1, 2, 3, 4\}$ using that horizon's own phase-space-reconstructed data (i.e. no horizon-wise model selection is performed; every model is evaluated at every horizon).

D.4 Multi-Horizon Forecasting Protocol

For a horizon T, the forecasting task is constructed via phase-space reconstruction (Appendix B) as:

$$\mathbf{X}(t_i) = \left[r_{t_i}, \ldots, r_{t_i-(d-1)\tau}\right] \;\mapsto\; r_{t_i+T+(d-1)\tau},$$

with $d = 8$, $\tau = 1$. We evaluate $T \in \{1, 2, 3, 4\}$. All input-target pairs are constructed chronologically; models are trained only on the training portion and evaluated on the held-out test segment.

D.5 Phase-Space Reconstruction and Predictability Diagnostics

Full method details, the embedding-dimension provenance caveat, and the resulting diagnostics table are provided in Appendix G (Table 8). We report these diagnostics for DK1 only in this paper; the corresponding phase-space reconstruction and Lyapunov estimation have not yet been run for SE4 or NO4.

D.6 Sensitivity Analysis

To assess the role of the Lévy stability parameter, LévySDE-Net is trained with:

$$\alpha \in \{1.2, 1.4, 1.5, 1.6, 1.8, 2.0\}.$$

All other model components, optimiser settings, preprocessing steps, and train–test splits are held fixed. This isolates whether predictive performance is driven by the heavy-tailed stochastic driver or by the neural SDE architecture. We report this sweep for DK1 only (Appendix I); it has not yet been run for SE4 or NO4.

D.7 Synthetic Recovery Experiment

For identifiability analysis, we simulate trajectories from a Lévy-driven Ornstein–Uhlenbeck process:

$$dX_t = -\theta X_t \, dt + \sigma \, dL_t^{(\alpha)}.$$

The true parameter α is fixed, and the estimation procedure is evaluated over multiple seeds. Recovery accuracy is reported using the signed error $\hat{\alpha} - \alpha_{\text{true}}$ and the compensation correlation with $\hat{\sigma}$ (see Appendix J for full results).

D.8 Statistical Testing

We use Diebold–Mariano tests to compare each forecasting model against the zero-difference baseline, and paired Diebold–Mariano tests at the per-seed level to compare LévySDE-Net against the Gaussian SDE-Net baseline. The null hypothesis is equal predictive accuracy; see Appendix C for the test statistic and the correction to its stated multi-horizon behavior (currently, DM calls in the codebase use $h = 1$ uniformly rather than $h = T$, so no autocorrelation correction is presently applied at $T > 1$ – pending resolution per Appendix C).

D.9 Compute Environment

All experiments are run on CPU, with `torch.use_deterministic_algorithms(True)` and `torch.set_num_threads(1)` enabled for run-to-run reproducibility. A single training run (one model, one α, one horizon, one seed; 100 epochs $\times$ 50 inner iterations) takes approximately 9 minutes. The full α-sensitivity sweep (6 values of $\alpha \times 5$ seeds $= 30$ runs, one-step-ahead only) is estimated at under 5 hours sequential on this hardware. No specialised hardware or distributed computing resources are required, though the full experimental grid across all three price areas represents a non-trivial sequential compute budget.

D.10 Reproducibility

Random seeds are fixed for Python, NumPy, and PyTorch (`random.seed`, `np.random.seed`, `torch.manual_seed`), together with the determinism settings noted above. The repository includes scripts for preprocessing, one-step forecasting, multi-horizon forecasting, Diebold–Mariano

testing, α-sensitivity analysis, trajectory-rollout divergence testing, and synthetic recovery experiments.

E LÉVYSDE-NET ARCHITECTURE AND TRAINING

E.1 DRIFT AND DIFFUSION PARAMETERIZATION

The drift function $f(\cdot)$ and diffusion function $g(\cdot)$ are parameterized by neural networks (Appendix D for the exact layer specification). The resulting model conceptually defines a Lévy-driven neural stochastic differential equation of the form

$$X_t = X_0 + \int_0^t f(X_s, s; \theta_f)\, ds + \int_0^t g(X_s, s; \theta_g)\, dL_t^\alpha, \tag{6}$$

where L_t^α is a symmetric α-stable Lévy process with stability parameter $1 < \alpha \leq 2$ ($\alpha = 2$ recovering the Gaussian/Brownian limit used by the Gaussian SDE-Net baseline).

This formulation admits the standard Lévy–Itô decomposition, in which the driving noise is represented through a compensated Poisson random measure $\tilde{N}(ds, du)$:

$$X_t = X_0 + \int_0^t f(X_s, s; \theta_f)\, ds + \int_0^t \int_{|u|>0} g(X_s, s; \theta_g)\, u\, \tilde{N}(ds, du), \tag{7}$$

with associated α-stable Lévy measure $\nu(du) = C_\alpha |u|^{-1-\alpha} du$, C_α a normalizing constant. This representation is presented as motivating mathematical background for the discrete-time model actually implemented (Appendix F), not as a claim that the training procedure targets a continuous-time likelihood or residual derived from it.

E.2 TRAINING OBJECTIVE

The model is *not* trained via a physics-consistency residual on Equation 6. Training instead uses the composite adversarial scheme described in full in Appendix D.2: a mean-squared data-fitting loss on the readout, optimized jointly with an adversarial binary cross-entropy loss that shapes the diffusion network's output magnitude by training it to distinguish real inputs from Gaussian-perturbed ones. We refer to that appendix for the exact loss functions, optimizers, and hyperparameters, rather than duplicating them here.

E.3 OPTIMISATION AND STABILISATION

Training uses two separate SGD optimizers (one for the downsampling/drift parameters, one for the diffusion parameters; see Appendix D.2 for learning rates and momentum), each with weight decay. Gradient clipping (max norm 5.0) is applied to the downsampling/drift parameters to mitigate instability arising from heavy-tailed Lévy perturbations propagating through the forward pass.

Because the forward pass draws fresh α-stable increments at every call (Appendix F), each training iteration and each evaluation-time forward pass uses independently resampled noise rather than a fixed realization – this is what enables the ensemble-based uncertainty analysis in the main text (Experiments 6 and 7), which draws repeated stochastic forward passes from a single trained model.

F LÉVY NOISE MODELLING AND NUMERICAL SCHEME

F.1 α-STABLE LÉVY SAMPLING

Symmetric α-stable Lévy increments are generated via `scipy.stats.levy_stable`, which implements the Chambers–Mallows–Stuck (CMS) sampling procedure. The Lévy driver is parameterized by the stability parameter $\alpha \in (1, 2]$ and skewness $\beta = 0$ (symmetric), with smaller α corresponding to heavier-tailed stochastic forcing. See Appendix E for the Lévy–Itô representation this driver corresponds to conceptually.

F.2 Euler–Maruyama Discretisation of the Forward Pass

The model's forward pass implements an Euler–Maruyama discretization directly: starting from the downsampled input x_0, the state is propagated through 6 steps ($\Delta t = 1/6$) as

$$x_{k+1} = x_k + f(x_k; \theta_f)\,\Delta t + \sigma \cdot g_0 \cdot (\Delta t)^{1/\alpha}\, L_k, \tag{8}$$

where L_k are freshly-sampled i.i.d. symmetric α-stable increments (resampled at every step of every forward pass, not fixed across steps or reused across calls), g_0 is the diffusion network's output evaluated once at $k = 0$ and held fixed across the six steps, and $\sigma = 0.5$ is a fixed scale constant (Appendix D.2). This discretization is used to generate the model's predictions – it is the forward pass itself, not a separate residual term used inside the training loss.

F.3 Numerical Stability

α-stable processes with $\alpha < 2$ can exhibit large, discontinuous increments, which can destabilize naive optimization. Gradient clipping (Appendix E) mitigates this for the downsampling/drift parameters. No explicit clipping is applied to the noise increments L_k themselves or to any residual term, since no residual term exists in the implemented objective.

Positive Lyapunov exponents estimated from reconstructed phase-space trajectories (Appendix G) should be interpreted cautiously: since the underlying price-difference series are predominantly stochastic ($E_2 \approx 1$ under Cao's criterion), the resulting Lyapunov diagnostics represent finite-sample measures of effective trajectory divergence rather than evidence of low-dimensional deterministic chaos.

G Dynamical Analysis and Lyapunov Estimation

G.1 Phase-Space Reconstruction

To analyse effective instability and predictability limits, we reconstruct delay-coordinate phase spaces from the one-dimensional electricity price-difference series for each price area. The reconstructed state vectors are defined as

$$\mathbf{X}(t_i) = \left[r_{t_i}, r_{t_i-\tau}, \ldots, r_{t_i-(m-1)\tau}\right]. \tag{9}$$

The embedding delay τ is selected using the Average Mutual Information (AMI) criterion; for DK1, AMI decays rapidly and we adopt $\tau = 1$.

The embedding dimension m is estimated using Cao's E_1/E_2 criterion: the optimal dimension is selected as the smallest value for which $E_1(m)$ stabilizes, while $E_2(m) \approx 1$ indicates predominantly stochastic, high-dimensional dynamics rather than low-dimensional deterministic chaos. For DK1, this procedure – applied to an earlier percentage-return transformation of the series, rather than the price-difference series used throughout the rest of this paper – selected $m = 8$; this value is used for DK1 throughout the main text and this appendix. This selection has not yet been independently re-checked by applying Cao's criterion directly to DK1's own price-difference series, and we report it as the working value used in the main results rather than as a fully re-validated choice. We report only DK1's phase-space and Lyapunov diagnostics in this paper (Table 8); the corresponding diagnostics for SE4 and NO4 have not yet been run and are not reported.

G.2 Lyapunov Exponent Estimation

The maximal Lyapunov exponent $\lambda_{\max}$ is estimated using a Wolf-type nearest-neighbour orbit-divergence procedure applied to the reconstructed trajectories. Positive values of $\lambda_{\max}$ indicate effective instability through exponential separation of nearby trajectories.

The corresponding Lyapunov time

$$h = \frac{1}{\lambda_{\max}}$$

is interpreted as an empirical predictability timescale – the rate at which forecast uncertainty amplifies with horizon – rather than a strict deterministic forecasting boundary.

Importantly, positive Lyapunov exponents are not interpreted here as evidence of low-dimensional deterministic chaos. Since DK1's price-difference dynamics are predominantly stochastic ($E_2 \approx 1$ under Cao's criterion), the estimated exponent should instead be viewed as a finite-sample measure of trajectory divergence arising from the interaction between stochastic forcing and nonlinear dynamics – consistent with the framing adopted throughout the main text (Experiments 3 and 7).

Table 8: Phase-space reconstruction and predictability diagnostics for DK1. The delay parameter τ is estimated using average mutual information (AMI), while the embedding dimension m_{opt} is selected using Cao's E_1/E_2 criterion (see the caveat above regarding its provenance). The maximal Lyapunov exponent $\lambda_{\max}$ is reported together with the associated predictability horizon $h = 1/\lambda_{\max}$; $\lambda_{\max}$ here is computed directly from h for internal consistency with the value used throughout the main text.

Price Area	τ	m_{opt}	$\lambda_{\max}$ (nat/sample)	$h = 1/\lambda_{\max}$ (hours)
DK1	1	8	0.00877	114.00

For DK1, $E_2(m) \approx 1$, supporting the interpretation of the price-difference dynamics as stochastic and high-dimensional rather than low-dimensional deterministic chaos. This is the only price area for which we report a Lyapunov-time estimate in this paper; the iterative-rollout validation of this estimate (main text, Experiment 7) finds that the Lévy-versus-Gaussian ensemble-spread ratio crosses a fixed divergence threshold at $\approx$67 hours – before h – and continues to diverge well beyond it, consistent with h marking a meaningful, if approximate, predictability boundary for this area. We do not report comparable estimates for SE4 or NO4, and do not make any claim about how DK1's predictability horizon compares to theirs.

H Additional Experimental Results

This section provides supplementary results for SE4 and NO4. DK1 results are reported in full in the main text (Experiments 3, 5, and 7); this appendix extends the one-step and multi-horizon forecasting comparisons to the remaining two areas. Per-seed Diebold–Mariano testing, both against the zero baseline and directly between LévySDE-Net and Gaussian SDE-Net, is reported for DK1 only (Appendix H.3); extending it to SE4 and NO4 is left as an open item.

H.1 One-Step Forecasting Results

This section reports one-step-ahead forecasting results for SE4 and NO4, using the same model set, training procedure, and evaluation protocol as DK1's one-step comparison (Section 4.4). Results for neural models are reported as mean $\pm$ standard deviation over five random seeds; deterministic econometric models are reported without variability estimates. Lower values are better.

H.1.1 SE4

Table 9: One-step-ahead forecasting results for Nordpool SE4 electricity spot price differences. Neural models are reported as mean $\pm$ standard deviation over five random seeds. Lower is better.

	Model	MSE	RMSE	MAE	SMAPE
	Zero baseline	1.7036	1.3052	0.7252	145.31
	AR(1)	1.6399	1.2806	0.6576	199.93
!	GARCH(1,1)	1.6401	1.2807	0.6568	175.79
	LSTM	1.5780 $\pm$ 0.0411	1.2561 $\pm$ 0.0163	0.6432 $\pm$ 0.0089	117.35 $\pm$ 1.68
	Gaussian SDE-Net	1.5034 $\pm$ 0.0102	1.2261 $\pm$ 0.0042	0.6367 $\pm$ 0.0032	137.99 $\pm$ 2.07
	LévySDE-Net	1.5067 $\pm$ 0.0115	1.2275 $\pm$ 0.0047	0.6370 $\pm$ 0.0035	138.03 $\pm$ 2.13

Both neural SDE variants substantially outperform the econometric and naive baselines: LévySDE-Net and Gaussian SDE-Net reduce MSE by approximately 12% relative to AR(1) (versus DK1's $\approx$14%), and both outperform the LSTM baseline on MSE, RMSE, and MAE, though as at DK1 the LSTM's SMAPE advantage should be read cautiously given SMAPE's instability near zero-crossing series.

LévySDE-Net and Gaussian SDE-Net again achieve nearly identical aggregate performance, with LévySDE-Net numerically behind Gaussian SDE-Net by a small margin (1.5067 vs. 1.5034 MSE, well under one seed-level standard deviation) – the same direction as DK1's one-step result (Gaussian mildly favoured). The per-seed separation is markedly weaker than at DK1, however: none of the five seeds shows an individually significant difference between LévySDE-Net and Gaussian SDE-Net at the 5% level at $q = 1$ for SE4, compared to four of five seeds significant (with inconsistent sign) at DK1's one-step horizon. Whether this reflects SE4's own tail-heaviness relative to DK1's, or simply ordinary sampling variability across only five seeds, cannot be resolved from this one-step result alone; we do not draw a jump-dominance-scaling conclusion from it here.

H.1.2 NO4

Table 10: One-step-ahead forecasting results for Nordpool NO4 electricity spot price differences. Neural models are reported as mean ± standard deviation over five random seeds. Lower is better.

	Model	MSE	RMSE	MAE	SMAPE
	Zero baseline	0.3962	0.6295	0.4978	161.48
	AR(1)	0.1940	0.4404	0.1953	199.16
!	GARCH(1,1)	0.1940	0.4404	0.1950	169.81
	LSTM	0.1753 ± 0.0064	0.4187 ± 0.0077	0.1847 ± 0.0087	123.07 ± 6.23
	Gaussian SDE-Net	0.1720 ± 0.0039	0.4147 ± 0.0048	0.1922 ± 0.0028	143.50 ± 3.18
	LévySDE-Net	0.1721 ± 0.0040	0.4148 ± 0.0048	0.1922 ± 0.0029	143.55 ± 3.13

Both neural SDE variants substantially outperform the naive, econometric, and recurrent baselines: LévySDE-Net and Gaussian SDE-Net reduce MSE by approximately 11% relative to AR(1) – somewhat smaller than DK1's ≈14% and broadly in line with SE4's ≈12% – and both again outperform the LSTM on MSE, RMSE, and MAE, with the LSTM's SMAPE advantage read with the same caution noted above.

LévySDE-Net and Gaussian SDE-Net achieve almost identical performance at NO4: 0.1721 vs. 0.1720 MSE, a difference of under 0.05% and far smaller than either model's own seed-level standard deviation (≈0.004) – the smallest aggregate gap of the three areas at one-step-ahead. We do not have per-seed Diebold–Mariano results for NO4 (Appendix H introduction), so we do not know whether this near-tie masks seed-level significant differences the way it did at DK1 and SE4; we note the aggregate result without treating it as conclusive on its own.

H.2 Multi-Horizon Forecasting Results

We report multi-horizon out-of-sample forecasting results for SE4 and NO4, using the same protocol as DK1's multi-horizon comparison (Section 4.6): horizons $T \in \{1, 2, 3, 4\}$, each model retrained independently at each horizon.

H.2.1 SE4

Table 11: Multi-horizon out-of-sample forecasting results for Nordpool SE4 electricity spot price differences. Lower is better. Neural models are reported as mean ± standard deviation across five random seeds.

Horizon T	Model	MSE	RMSE	MAE	SMAPE
6*1	Zero baseline	1.7036	1.3052	0.7252	145.31
	AR(1)	1.6399	1.2806	0.6576	199.93
	GARCH(1,1)	1.6401	1.2807	0.6568	175.79
	LSTM	1.5780 ± 0.0411	1.2561 ± 0.0163	0.6432 ± 0.0089	117.35 ± 1.68
	Gaussian SDE-Net	1.5034 ± 0.0102	1.2261 ± 0.0042	0.6367 ± 0.0032	137.99 ± 2.07
	LévySDE-Net	1.5067 ± 0.0115	1.2275 ± 0.0047	0.6370 ± 0.0035	138.03 ± 2.13
6*2	Zero baseline	1.7037	1.3053	0.7252	145.32
	AR(1)	1.6401	1.2806	0.6577	199.92
	GARCH(1,1)	1.6402	1.2807	0.6569	175.80
	LSTM	1.5440 ± 0.0203	1.2426 ± 0.0081	0.6646 ± 0.0056	122.38 ± 0.87
	Gaussian SDE-Net	1.5826 ± 0.0163	1.2580 ± 0.0065	0.6717 ± 0.0046	149.74 ± 2.33
!	LévySDE-Net	1.5863 ± 0.0161	1.2595 ± 0.0064	0.6718 ± 0.0042	149.83 ± 2.31
6*3	Zero baseline	1.7039	1.3053	0.7253	145.32
	AR(1)	1.6402	1.2807	0.6578	199.92
	GARCH(1,1)	1.6404	1.2808	0.6569	175.80
	LSTM	1.5659 ± 0.0974	1.2509 ± 0.0387	0.6634 ± 0.0155	123.91 ± 0.35
	Gaussian SDE-Net	1.5686 ± 0.0176	1.2524 ± 0.0070	0.6686 ± 0.0079	149.22 ± 1.72
	LévySDE-Net	1.5705 ± 0.0171	1.2532 ± 0.0068	0.6689 ± 0.0077	149.35 ± 1.70
6*4	Zero baseline	1.7041	1.3054	0.7253	145.32
	AR(1)	1.6404	1.2808	0.6578	199.95
	GARCH(1,1)	1.6406	1.2808	0.6570	175.80
	LSTM	1.4781 ± 0.0427	1.2157 ± 0.0176	0.6459 ± 0.0101	123.13 ± 0.67
	Gaussian SDE-Net	1.5906 ± 0.0119	1.2612 ± 0.0047	0.6694 ± 0.0076	150.52 ± 2.11
	LévySDE-Net	1.5897 ± 0.0106	1.2608 ± 0.0042	0.6690 ± 0.0074	150.65 ± 2.10

Degradation with horizon at SE4 follows a broadly similar shape to DK1: both SDE variants show their largest single increase in error between $T = 1$ and $T = 2$ (MSE rising from ≈1.50 to ≈1.58), after which performance is roughly flat from $T = 2$ through $T = 4$ (1.58–1.59 throughout) – the same "step up then plateau" pattern reported for DK1. The econometric baselines are again essentially flat across all four horizons, consistent with hourly price differences behaving close to a stationary, weakly-dependent series at SE4 as well.

The Lévy-versus-Gaussian gap remains small and sign-inconsistent across horizons, as at DK1: Gaussian SDE-Net is marginally ahead at $T = 1$ and $T = 2$, essentially tied at $T = 3$, and LévySDE-Net is marginally ahead at $T = 4$ (1.5897 vs. 1.5906) – in every case well under 1% relative MSE. Given the LSTM's anomalous $T = 3/T = 4$ behaviour noted above, we do not draw a comparative claim about the SDE variants' degradation pattern relative to the LSTM's at SE4 until that instability is resolved.

H.2.2 NO4

Table 12: Multi-horizon out-of-sample forecasting results for Nordpool NO4 electricity spot price differences. Lower is better. Neural models are reported as mean ± standard deviation across five random seeds.

Horizon T	Model	MSE	RMSE	MAE	SMAPE
6*1	Zero baseline	0.3962	0.6295	0.4978	161.48
	AR(1)	0.1940	0.4404	0.1953	199.16
	GARCH(1,1)	0.1940	0.4404	0.1950	169.81
	LSTM	0.1753 ± 0.0064	0.4187 ± 0.0077	0.1847 ± 0.0087	123.07 ± 6.23
	Gaussian SDE-Net	0.1720 ± 0.0039	0.4147 ± 0.0048	0.1922 ± 0.0028	143.50 ± 3.18
	LévySDE-Net	0.1721 ± 0.0040	0.4148 ± 0.0048	0.1922 ± 0.0029	143.55 ± 3.13
6*2	Zero baseline	0.3962	0.6295	0.4978	161.47
	AR(1)	0.1940	0.4404	0.1953	199.14
	GARCH(1,1)	0.1940	0.4405	0.1950	169.80
	LSTM	0.2393 ± 0.0522	0.4871 ± 0.0509	0.1984 ± 0.0066	132.70 ± 4.10
	Gaussian SDE-Net	0.1895 ± 0.0046	0.4353 ± 0.0053	0.2058 ± 0.0034	152.31 ± 3.62
!	LévySDE-Net	0.1899 ± 0.0042	0.4358 ± 0.0048	0.2058 ± 0.0033	152.35 ± 3.60
6*3	Zero baseline	0.3963	0.6295	0.4978	161.48
	AR(1)	0.1940	0.4405	0.1953	199.14
	GARCH(1,1)	0.1940	0.4405	0.1950	169.80
	LSTM	0.1906 ± 0.0130	0.4364 ± 0.0147	0.2032 ± 0.0086	134.15 ± 1.45
	Gaussian SDE-Net	0.1881 ± 0.0046	0.4337 ± 0.0053	0.2051 ± 0.0042	152.23 ± 3.26
	LévySDE-Net	0.1881 ± 0.0046	0.4337 ± 0.0053	0.2051 ± 0.0042	152.28 ± 3.21
6*4	Zero baseline	0.3963	0.6295	0.4978	161.49
	AR(1)	0.1940	0.4405	0.1953	199.17
	GARCH(1,1)	0.1940	0.4405	0.1950	169.80
	LSTM	0.2061 ± 0.0164	0.4537 ± 0.0179	0.1998 ± 0.0044	135.87 ± 4.42
	Gaussian SDE-Net	0.1902 ± 0.0043	0.4361 ± 0.0049	0.2042 ± 0.0039	153.20 ± 3.53
	LévySDE-Net	0.1903 ± 0.0042	0.4363 ± 0.0048	0.2041 ± 0.0038	153.21 ± 3.46

Degradation with horizon at NO4 follows the same qualitative shape reported for DK1 and SE4: both SDE variants show their largest single increase in error between $T = 1$ and $T = 2$ (MSE rising from ≈0.172 to ≈0.190, a ≈10% relative step), after which performance is essentially flat from $T = 2$ through $T = 4$ (≈0.188–0.190 throughout). The econometric baselines are again flat across all four horizons, consistent with the DK1/SE4 pattern.

The Lévy-versus-Gaussian gap remains small and sign-inconsistent across horizons, as at DK1 and SE4: Gaussian SDE-Net is marginally ahead at $T = 1$, $T = 3$, and $T = 4$, and LévySDE-Net is marginally ahead at $T = 2$ (0.1895 vs. 0.1899) – in every case well under 1% relative MSE, the smallest aggregate gap of the three areas at every horizon tested. We do not draw a jump-dominance interpretation from this: as noted in Section 4.2, NO4's tail-heaviness diagnostics do not resolve to a consistent ranking against SE4 and DK1, so we report the point-accuracy pattern here without attributing it to NO4's position in a jump-dominance ordering.

Figure 1: One-step-ahead forecast comparison for SE4. The upper panel compares the observed price-difference trajectory against the zero baseline and LévySDE-Net forecasts; the lower panel shows forecast residuals relative to the true series.

Figure 2: One-step-ahead forecast comparison for NO4. The upper panel compares the observed price-difference trajectory against the zero baseline and LévySDE-Net forecasts; the lower panel shows forecast residuals relative to the true series.

H.3 Diebold–Mariano Tests: Against the Zero Baseline and Between Forcing Mechanisms

We report two sets of per-seed Diebold–Mariano tests for DK1, across all four horizons: each model against the zero-difference baseline (Table 13), and a direct paired test between LévySDE-Net and

Gaussian SDE-Net (Table 14), which is the comparison the main text's significance claims (Section 4.4, Section 4.6) are actually based on. We report per-seed results directly rather than a single pooled statistic per horizon, since each of the five neural-model training runs constitutes an independent realization rather than repeated measurements of one underlying series. This is the only area for which we report per-seed DM tests in this appendix; extending the same analysis to SE4 and NO4 is left as an open item.

Table 13: Per-seed Diebold–Mariano test results for DK1, comparing LévySDE-Net and Gaussian SDE-Net against the zero-difference baseline, across all four horizons and five random seeds.

Horizon	Seed	Comparison	DM Statistic	p-value
1	0	LévySDE-Net vs. Zero	-8.133	< 0.001
1	0	Gaussian SDE-Net vs. Zero	-8.082	< 0.001
1	1	LévySDE-Net vs. Zero	-6.790	< 0.001
1	1	Gaussian SDE-Net vs. Zero	-6.698	< 0.001
1	2	LévySDE-Net vs. Zero	-9.162	< 0.001
1	2	Gaussian SDE-Net vs. Zero	-9.126	< 0.001
1	3	LévySDE-Net vs. Zero	-7.689	< 0.001
1	3	Gaussian SDE-Net vs. Zero	-10.675	< 0.001
1	4	LévySDE-Net vs. Zero	-6.261	< 0.001
1	4	Gaussian SDE-Net vs. Zero	-6.663	< 0.001
2	0	LévySDE-Net vs. Zero	-14.647	< 0.001
2	0	Gaussian SDE-Net vs. Zero	-14.732	< 0.001
2	1	LévySDE-Net vs. Zero	-12.063	< 0.001
2	1	Gaussian SDE-Net vs. Zero	-12.236	< 0.001
2	2	LévySDE-Net vs. Zero	-17.350	< 0.001
2	2	Gaussian SDE-Net vs. Zero	-17.366	< 0.001
2	3	LévySDE-Net vs. Zero	-15.899	< 0.001
2	3	Gaussian SDE-Net vs. Zero	-15.889	< 0.001
2	4	LévySDE-Net vs. Zero	-12.262	< 0.001
! 2	4	Gaussian SDE-Net vs. Zero	-12.221	< 0.001
3	0	LévySDE-Net vs. Zero	-11.255	< 0.001
3	0	Gaussian SDE-Net vs. Zero	-14.907	< 0.001
3	1	LévySDE-Net vs. Zero	-13.780	< 0.001
3	1	Gaussian SDE-Net vs. Zero	-13.933	< 0.001
3	2	LévySDE-Net vs. Zero	-17.817	< 0.001
3	2	Gaussian SDE-Net vs. Zero	-17.605	< 0.001
3	3	LévySDE-Net vs. Zero	-15.623	< 0.001
3	3	Gaussian SDE-Net vs. Zero	-15.490	< 0.001
3	4	LévySDE-Net vs. Zero	-13.654	< 0.001
3	4	Gaussian SDE-Net vs. Zero	-13.438	< 0.001
4	0	LévySDE-Net vs. Zero	-14.096	< 0.001
4	0	Gaussian SDE-Net vs. Zero	-13.898	< 0.001
4	1	LévySDE-Net vs. Zero	-14.205	< 0.001
4	1	Gaussian SDE-Net vs. Zero	-14.142	< 0.001
4	2	LévySDE-Net vs. Zero	-16.498	< 0.001
4	2	Gaussian SDE-Net vs. Zero	-16.511	< 0.001
4	3	LévySDE-Net vs. Zero	-12.168	< 0.001
4	3	Gaussian SDE-Net vs. Zero	-13.582	< 0.001
4	4	LévySDE-Net vs. Zero	-13.219	< 0.001
4	4	Gaussian SDE-Net vs. Zero	-13.458	< 0.001

Every seed at every horizon rejects the null of equal predictive accuracy against the zero-difference baseline for both LévySDE-Net and Gaussian SDE-Net ($p < 0.001$ throughout; Table 13), confirming that the neural SDE architecture's advantage over the naive baseline is robust across seeds and horizons at DK1. The open question is not whether the neural SDE approach helps – it clearly and consistently does – but whether the specific choice of driving noise matters, addressed directly next.

Table 14: Per-seed Diebold–Mariano test results for DK1, LévySDE-Net versus Gaussian SDE-Net directly, across all four horizons and five random seeds. Positive DM statistics favour Gaussian SDE-Net; negative favour LévySDE-Net.

Horizon	Seed	DM Statistic	p-value
1	0	2.120	0.034
1	1	0.156	0.876
1	2	2.439	0.015
1	3	1.513	0.130
1	4	1.139	0.255
2	0	0.929	0.353
2	1	2.397	0.017
2	2	3.219	0.001
2	3	1.759	0.079
2	4	0.370	0.712
3	0	1.089	0.276
3	1	1.834	0.067
3	2	1.128	0.259
3	3	0.584	0.559
3	4	−1.270	0.204
4	0	0.473	0.636
4	1	0.678	0.498
4	2	1.424	0.155
4	3	1.145	0.252
4	4	1.600	0.110

At the one-step horizon ($q = 1$), two of five seeds show a significant difference between LévySDE-Net and Gaussian SDE-Net at the 5% level ($p = 0.034$, $p = 0.015$), both favouring Gaussian; a third seed is marginal ($p = 0.130$). At $q = 2$, the same pattern holds – two of five seeds significant ($p = 0.017$, $p = 0.001$), again favouring Gaussian. By $q = 3$ and $q = 4$, however, no seed reaches significance at either horizon: the modest, direction-consistent Gaussian edge visible at short horizons is therefore not a stable, horizon-independent effect. It is concentrated at $q = 1$–2 and disappears by $q = 3$, consistent with the identifiability limit of Experiment 2 rather than a robust structural advantage for either forcing mechanism at longer horizons.

I Sensitivity and Ablation Studies

I.1 Sensitivity to the Lévy Stability Parameter

The main text reports the DK1 α-sensitivity results and their qualitative conclusions (Experiment 4, Section 4.5). We report DK1 only in this appendix: the corresponding α-sensitivity sweep has not yet been run for SE4 or NO4, so no comparison across areas is made here.

Table 15: Sensitivity of LévySDE-Net to the Lévy stability parameter α for DK1. Results are mean $\pm$ std over five seeds. Lower is better.

α	MSE	RMSE	MAE	SMAPE
1.2	2.374276 ± 0.135598	1.540362 ± 0.044152	0.913592 ± 0.033503	165.168 ± 13.958
1.4	2.056289 ± 0.011875	1.433972 ± 0.004147	0.836687 ± 0.003768	127.622 ± 1.600
1.5	2.054333 ± 0.011267	1.433290 ± 0.003933	0.837135 ± 0.003599	127.257 ± 1.045
1.6	2.054493 ± 0.011205	1.433346 ± 0.003911	0.837540 ± 0.003568	127.178 ± 0.971
1.8	2.054632 ± 0.011101	1.433395 ± 0.003874	0.837794 ± 0.003615	127.142 ± 0.944
2.0	2.054637 ± 0.011122	1.433396 ± 0.003881	0.837814 ± 0.003628	127.137 ± 0.945

Even under this unverified run, the table shows a pattern worth noting once confirmed: performance is essentially flat from $\alpha = 1.4$ through $\alpha = 2.0$ (MSE varying by under 0.1% across this range), with a sharp departure at $\alpha = 1.2$ – MSE rising by ≈15% relative to $\alpha = 1.4$, accompanied by a roughly tenfold increase in seed-level standard deviation (0.136 vs. ≈0.011 elsewhere). This is consistent with training instability at strongly heavy-tailed forcing rather than a smooth

accuracy gradient in α, but we do not treat this as established until the verification above is resolved.==========

Taken together with the DK1 results in the main text, the sensitivity analysis across all three price areas supports a consistent interpretation once complete:

J Synthetic Recovery and Identifiability Analysis

To assess identifiability of the Lévy stability parameter α and the robustness of the proposed training procedure, we conduct controlled experiments on synthetic data generated from known stochastic differential equations. We simulate trajectories from an Ornstein–Uhlenbeck process driven by α-stable Lévy noise:

$$dX_t = -\theta X_t \, dt + \sigma \, dL_t^{(\alpha)},$$

where $L_t^{(\alpha)}$ is a symmetric α-stable process ($\beta = 0$). The true parameters are fixed at $\theta = 0.1$ and $\sigma = 1.0$, and trajectories of length $N = 2000$ are simulated via Euler–Maruyama at hourly frequency to match the empirical forecasting setup, with a coarser, monthly-equivalent frequency ($720\times$ the hourly step, matching the calendar-month conversion at hourly base resolution: 24×30) used separately below to test whether sampling density affects identifiability.

Estimation procedure. Given a simulated trajectory, we first fit an AR(1)-style discretization of the OU drift by ordinary least squares to obtain residual innovations, then estimate the Lévy stability parameter α and the diffusion scale σ *jointly* via a two-dimensional maximum-likelihood optimization over the α-stable log-density. Joint estimation is important here: fixing the scale independently of α (e.g. via a Gaussian-consistent moment estimator) implicitly assumes tail behavior that need not hold under heavy-tailed forcing, and can bias the recovered α. To avoid artificially truncating the estimate at the edge of the valid parameter space, α is optimized in an unconstrained reparametrization (a logistic map into $(1.01, 2.0)$) rather than via a hard box constraint; scale is optimized in log-space to enforce positivity. The optimizer is run from three restarts with initial α guesses spread across the valid range – including one near the upper boundary – and the best-likelihood result across restarts is retained, since single-start optimization near a parameter boundary is prone to poor local optima that would otherwise distort the boundary-clipping analysis below. Because full-sample likelihood evaluation is computationally expensive, the likelihood used inside the optimizer is computed on a random subsample of 1200 residual points per run.

We evaluate recovery across 40 random seeds for each of five true stability parameters, $\alpha_{\text{true}} \in \{1.2, 1.5, 1.8, 1.9, 1.95\}$ — extending toward the Gaussian boundary ($\alpha \to 2$), where the theoretical argument for identifiability breakdown is sharpest — and test whether any residual bias in $\hat{\alpha}$ is statistically distinguishable from zero via a 95% confidence interval and a one-sample t-test on the signed error $\hat{\alpha} - \alpha_{\text{true}}$. This grid brackets both stability values used in the main empirical comparison ($\alpha = 1.5$ for LévySDE-Net, $\alpha = 2.0$ for the Gaussian SDE-Net baseline), so the recovery and compensation results below can be read directly against that comparison.

Table 16: Synthetic recovery results for the Lévy stability parameter α, joint (α, σ) MLE, 40 random seeds per setting, hourly sampling frequency. $\hat{\alpha}$ is the mean recovered value; the 95% CI is on the mean signed bias; *frac. boundary* is the fraction of seeds whose estimate landed within 0.005 of the $\alpha = 2$ boundary.

α_{true}	$\hat{\alpha}$ (mean)	$\text{std}(\hat{\alpha})$	Mean bias	95% CI	p-value	Frac. boundary
1.20	1.2068	0.0372	0.0068	[−0.0047, 0.0183]	0.253	0.000
1.50	1.4959	0.0460	−0.0041	[−0.0184, 0.0101]	0.574	0.000
1.80	1.8055	0.0337	0.0055	[−0.0049, 0.0160]	0.305	0.000
1.90	1.9017	0.0324	0.0017	[−0.0083, 0.0118]	0.737	0.000
1.95	1.9527	0.0270	0.0027	[−0.0057, 0.0111]	0.531	0.025

Table 17: Synthetic recovery results for the Lévy stability parameter α, joint (α, σ) MLE, 40 random seeds per setting, monthly-equivalent ($720\times$ coarser) sampling frequency. Columns as in Table 16.

α_{true}	$\hat{\alpha}$ (mean)	$\text{std}(\hat{\alpha})$	Mean bias	95% CI	p-value	Frac. boundary
1.20	1.2059	0.0393	0.0059	[−0.0063, 0.0181]	0.348	0.000
1.50	1.5110	0.0564	0.0110	[−0.0065, 0.0285]	0.224	0.000
1.80	1.8015	0.0438	0.0015	[−0.0121, 0.0150]	0.834	0.000
1.90	1.9142	0.0298	0.0142	[0.0049, 0.0234]	**0.005**	0.000
1.95	1.9551	0.0277	0.0051	[−0.0035, 0.0137]	0.251	0.125

Across the ten (setting $\times$ frequency) combinations tested, recovery is unbiased at the 5% level in nine of ten cases; the exception is $\alpha_{\text{true}} = 1.9$ under monthly-equivalent sampling, where a small positive bias (0.014) is statistically distinguishable from zero ($p = 0.005$). This is an isolated result – the same setting shows no bias at hourly sampling, and no bias appears at any other α_{true} under either frequency – and we report it as such rather than as evidence of a systematic bias affecting the estimator generally.

The compensation effect. Recovery imprecision is substantial at both sampling frequencies: $\text{std}(\hat{\alpha})$ ranges from 0.027 to 0.046 at hourly sampling and 0.030 to 0.056 at monthly-equivalent sampling (Tables 16 and 17). We trace this to a compensation effect between the recovered α and σ: seed-level errors in the two parameters are positively correlated at every tested setting and both sampling frequencies (Table 18), consistent with the optimizer partially explaining heavy-tailed fluctuations via inflated diffusion scale rather than correctly attributing them to tail-heaviness. At hourly sampling, this correlation is strongest at $\alpha_{\text{true}} = 1.2$ ($r = 0.410$) and weakens toward the Gaussian boundary ($r = 0.041$ at $\alpha_{\text{true}} = 1.95$) – the correlation does not sharpen near $\alpha = 2$ at hourly sampling, contrary to a naive expectation that the confound would be worst exactly where heavy-tailed and Gaussian forcing are hardest to distinguish.

Table 18: Compensation effect: correlation between α-error and σ-error, with 95% CI (Fisher z), at hourly and monthly-equivalent sampling frequency.

α_{true}	Hourly: r [95% CI]	Monthly: r [95% CI]
1.20	0.410 [0.113, 0.640]	0.553 [0.292, 0.737]
1.50	0.369 [0.065, 0.611]	0.404 [0.105, 0.635]
1.80	0.324 [0.014, 0.577]	0.420 [0.125, 0.647]
1.90	0.072 [−0.245, 0.375]	0.202 [−0.117, 0.483]
1.95	0.041 [−0.274, 0.348]	0.168 [−0.152, 0.455]

Ruling out discretization as the cause. A natural hypothesis is that recovery imprecision reflects the model observing only a discrete, coarsened view of an intrinsically continuous-time jump process, and that finer sampling would resolve α more precisely. To test this, we repeat the full recovery experiment at a $720\times$ coarser, monthly-equivalent observation spacing (holding the number of observations fixed, so that any change in recovery quality is attributable to sampling frequency rather than sample size) and compare $\text{std}(\hat{\alpha})$ and the compensation correlation between hourly and monthly sampling. The hypothesis is not supported in the direction that would be most favorable to the estimator: recovery variance is broadly comparable in magnitude across frequencies (inflating by roughly 6–30% at three of five settings under coarser sampling, and slightly lower at $\alpha_{\text{true}} = 1.9$), but the compensation correlation is higher under monthly-equivalent sampling than hourly sampling at every one of the five tested settings (Table 18), and a small but significant bias appears at $\alpha_{\text{true}} = 1.9$ under monthly sampling that is absent at hourly sampling. Coarser sampling therefore does not resolve the identifiability limit, and by several measures modestly worsens it. This is broadly, though not exactly, consistent with the self-similarity of symmetric α-stable laws under aggregation: a sum of k i.i.d. α-stable increments is itself α-stable with rescaled scale, which is consistent with recovery variance not degrading substantially under coarser sampling. However, because the underlying process is a mean-reverting Lévy-Ornstein-Uhlenbeck process rather than a pure α-stable Lévy motion, exact self-similarity is not guaranteed: the substantially larger recovered diffusion scale under monthly-equivalent sampling (e.g. $\hat{\sigma} \approx 5$–6 at $\alpha_{\text{true}} = 1.2$, versus ≈ 1 at hourly sampling) is consistent with the aggregated observations increasingly reflecting the process's stationary marginal

distribution rather than its instantaneous innovations, and the α–σ confound sharpens somewhat as a consequence, rather than remaining invariant.

Boundary behavior near $\alpha \to 2$. As α_{true} approaches the Gaussian limit, a fraction of seed-level estimates land at or within numerical precision of the boundary $\alpha = 2$ under maximum likelihood, and this fraction increases under coarser sampling: at $\alpha_{\text{true}} = 1.95$, 2.5% of seeds (1 of 40) land at the boundary under hourly sampling, rising to 12.5% (5 of 40) under monthly-equivalent sampling; no boundary clipping occurs at any other tested α_{true} under either frequency (Tables 16 and 17). Under the unconstrained reparametrization and multi-start optimization described above, these are genuine finite-sample MLE optima rather than artifacts of a hard constraint or a missed local optimum, and the increase under coarser sampling is consistent with coarser observation providing somewhat less, not more, information to distinguish a heavy-tailed process from an exactly Gaussian one near the boundary.

Scope of this diagnostic. This experiment establishes a reference identifiability bound for the Lévy-driven OU class under a standard, well-specified statistical estimator; it is independent of, and should not be conflated with, the accuracy of the model's own amortized α-estimation head, which we do not directly probe here. It should be read as a statement about the difficulty of the underlying estimation problem, and a floor against which the model's own estimates can be judged, rather than a direct characterization of the model's in-context inference. We therefore do not treat α as a precisely identifiable quantity from a single realization; a point estimate — from this diagnostic or from the model at inference time — should be interpreted alongside its estimation uncertainty and its coupling to $\hat{\sigma}$, rather than as an exact, independently identifiable value. In the main forecasting experiments, α is not estimated from the electricity price data — it is fixed as an architectural hyperparameter ($\alpha = 1.5$ for LévySDE-Net, $\alpha = 2.0$ for the Gaussian SDE-Net baseline).

K Compute and Reproducibility

All experiments are implemented in PyTorch, run on CPU , with `torch.use_deterministic_algorithms(True)` and `torch.set_num_threads(1)` enabled throughout for run-to-run reproducibility.

Random seeds are fixed across Python (`random.seed`), NumPy (`np.random.seed`), and PyTorch (`torch.manual_seed`) for every training run.

A single training run (one model, one α, one horizon, one seed; 100 epochs $\times$ 50 inner iterations) takes approximately 9 minutes. The full α-sensitivity sweep for one price area (5 values of $\alpha \times 4$ horizons $\times$ 5 seeds $= 100$ runs) is estimated at approximately 15 hours sequential on this hardware. No specialised hardware or distributed computing resources are required, though the full experimental grid across three price areas, multiple horizons, and the ensemble-based uncertainty experiments represents a non-trivial sequential compute budget.

L Societal Impact

The proposed framework has the potential to improve short-horizon electricity price forecasting and uncertainty quantification, particularly for market participants, grid operators, and retail suppliers who must manage exposure to volatile spot prices driven by weather-dependent generation (wind) and transmission constraints. By providing calibrated uncertainty estimates alongside point forecasts, the framework may support better-informed operational and risk-management decisions – for example, in scheduling flexible demand or dispatchable generation around periods of anticipated price volatility.

However, use of such models in real-world electricity markets carries risks. Electricity markets are safety-critical infrastructure: incorrect forecasts or over-reliance on model outputs in dispatch or trading decisions could contribute to grid instability, unexpected price spikes passed on to consumers, or poor risk management by market participants. Forecasting tools of this kind could in principle also be used to inform strategic bidding behaviour in ways that reduce rather than improve market efficiency, particularly in already-concentrated generation markets.

These risks highlight the importance of responsible deployment, including appropriate human oversight, robustness checks across market conditions not represented in the training data (e.g. extreme scarcity events), and a clear understanding of the model's limitations – particularly its weak identifiability of α (Appendix J) and its currently single-area validation of the predictability-horizon claim (Section 4.8) – when applied in practice.

L.1 N.1 Numerical Stability of the Forward Pass

For α-stable Lévy drivers with $\alpha < 2$, higher-order moments may be theoretically undefined (28), which complicates interpretation of any mean-squared quantity computed from the process in the continuous-time limit. In the implemented model, this is relevant to the readout loss L_{data} (Appendix D), which is a finite-sample mean-squared error over discretized, forward-simulated trajectories rather than a population moment of the continuous-time process. Numerical stability during training is provided by gradient clipping (max norm 5.0) on the drift and downsampling parameters, together with the diffusion network's bounded (sigmoid) output; no separate clipping is applied to the noise increments themselves or to any residual term. Consequently, the model's forecasts should be interpreted as an empirical, discrete-time approximation to Lévy-driven dynamics rather than as an exact simulation or an exact likelihood-based inference procedure for the continuous-time process.

L.2 N.2 Identifiability Limitations

The synthetic recovery experiments (Appendix J) indicate partial non-identifiability of the Lévy stability parameter, with different combinations of diffusion scaling and tail-heaviness producing similar observed dynamics. This non-identifiability manifests as a compensation effect between the recovered stability parameter and diffusion scale that is, at hourly sampling, *strongest* away from the Gaussian boundary ($r = 0.410$ at $\alpha_{\text{true}} = 1.2$) and *weakens* toward $\alpha = 2$ ($r = 0.041$ at $\alpha_{\text{true}} = 1.95$; Appendix J, Table 18) – the opposite of a naive expectation that the confound would sharpen near the Gaussian limit. Separately, a nontrivial fraction of maximum-likelihood estimates land at the boundary of the parameter space specifically near $\alpha \to 2$ (Appendix J, Tables 16 and 17). Point estimates of α, whether from this diagnostic or from the model at inference time, should therefore be interpreted alongside their estimation uncertainty rather than as exact physical quantities. In the main forecasting experiments, α is not estimated from the electricity price data – it is fixed as an architectural hyperparameter ($\alpha = 1.5$ for LévySDE-Net, $\alpha = 2.0$ for the Gaussian SDE-Net baseline), so this identifiability limitation bears on how α *would* be interpreted if estimated, not on the validity of the fixed-hyperparameter comparisons reported in the main text.

L.3 N.3 Predictability-Horizon Scope

The Lyapunov-based predictability analysis should not be interpreted as a strict deterministic-chaos diagnostic. Since electricity price-difference dynamics are predominantly stochastic ($E_2 \approx 1$ under Cao's criterion; Appendix G), estimated trajectory divergence reflects both nonlinear drift amplification and stochastic perturbations rather than low-dimensional deterministic chaos. Accordingly, the Lyapunov time is interpreted operationally as an empirical predictability timescale rather than as a formal invariant of deterministic chaos.

For DK1, this interpretation has been tested directly rather than left untested: the iterative-rollout experiment (Section 4.8) extends forecast evaluation to $3\times$ the estimated Lyapunov time (≈ 344 hours, against $h \approx 114$ hours), and finds that the Lévy-versus-Gaussian ensemble-spread ratio crosses a fixed divergence threshold at ≈ 67 hours – *before* h – and continues to widen well beyond it rather than plateauing. This is consistent with h marking a meaningful, if approximate, boundary in predictive behaviour for DK1, and is evidence *for* the Lyapunov-based interpretation at this horizon regime, not merely an untested prediction of it.

This finding is currently established for DK1 only; the same rollout validation has not yet been run for SE4 or NO4 (Section 4.8, main text), and the divergence-threshold choice ($\rho^\star = 4.0$) used to define the crossing point in that experiment is itself a design choice not yet independently justified (Section 4.8). We therefore treat the DK1 rollout result as a targeted, illustrative confirmation on the area where the effect is expected to be strongest, rather than as evidence that the same relationship

holds uniformly across areas of differing jump-dominance; testing that more broadly is left to future work (Section 5), and we do not report such an analysis in this paper. Separately, our multi-horizon evaluation (Experiment 5) is formulated as direct T-step regression rather than iterative rollout of the learned SDE; Lyapunov divergence concerns the separation of nearby trajectories under iterated dynamics, which is a related but distinct statistical object from fixed-horizon direct-regression error, and the two should not be conflated.

L.4 N.4 Symmetric Driver Assumption

Electricity spot prices are themselves structurally asymmetric: scarcity events can produce very large upward price spikes, while oversupply is bounded below by a floor near zero or negative prices (generators paying to offload excess generation), rather than symmetric shocks in either direction. LévySDE-Net's Lévy driver is symmetric ($\beta = 0$; Appendix D), which does not explicitly model this directional asymmetry. We adopt the symmetric case for tractability and consistency with the identifiability analysis in Appendix J, which assumes $\beta = 0$ throughout; extending both the model and the identifiability results to a skewed ($\beta \neq 0$) driver – better matched to the upward-skewed spike behaviour of scarcity-driven price movements, particularly in wind-dominated areas like DK1 – is a natural direction for future work.

Future work may also investigate settings with more extreme jump-dominance than any of the three areas evaluated here – for example, price areas with even higher renewable penetration or more frequent transmission congestion events – and test, statistically rather than only directionally, whether the magnitude of the Lévy-versus-Gaussian uncertainty effect scales with jump-dominance as measured in Table 2 (Section 5); we do not report such an analysis in this paper. Additional directions include explicit regime-switching dynamics, further probabilistic calibration analysis beyond Experiment 6, skewed ($\beta \neq 0$) Lévy drivers better matched to directional scarcity-driven jumps, and neural jump-diffusion formulations with stronger theoretical identifiability guarantees than the amortized α-estimation approach discussed in Appendix J.