Global Certificate Course in Electricity Price Forecasting

# Advanced Computational Tools For Forecasting

Artificial Neural Network (ANN)
Concept: A computational model inspired by the structure of biological neurons.
Related terms: deep learning, feed-forward network, backpropagation.
Explanation: ANNs consist of layers of interconnected nodes that transform input data through weighted connections and activation functions. In electricity price forecasting, ANNs capture non-linear relationships between demand, generation mix, weather, and market variables.
Example: A three-layer ANN predicts day-ahead market clearing prices using historical load, temperature, and fuel cost data.
Practical application: Integration with real-time data streams to update price forecasts every hour.
Challenges: Requires large training datasets, prone to over-fitting, and model interpretability is limited.

Autoregressive Integrated Moving Average (ARIMA)
Concept: A statistical time-series model that combines autoregression, differencing, and moving-average components.
Related terms: seasonal ARIMA (SARIMA), Box-Jenkins methodology.
Explanation: ARIMA models capture linear dependencies and trends in historical price series. The "I" (integrated) part removes non-stationarity by differencing the series.
Example: An ARIMA(2,1,1) model forecasts hourly electricity prices by modeling the differenced price series with two lagged terms and one moving-average term.
Practical application: Short-term price prediction where computational speed is critical.
Challenges: Limited in handling non-linear patterns, requires careful stationarity testing, and model selection can be time-consuming.

Bayesian Inference
Concept: A probabilistic framework that updates the probability of a hypothesis as more evidence becomes available.
Related terms: Markov Chain Monte Carlo (MCMC), prior distribution, posterior distribution.
Explanation: In price forecasting, Bayesian methods incorporate prior knowledge (e.g., expert beliefs about fuel price volatility) and combine it with observed data to generate predictive distributions rather than point forecasts.
Example: A Bayesian hierarchical model estimates regional price spikes by linking local price series to a national prior.
Practical application: Quantifying forecast uncertainty for risk-adjusted bidding strategies.
Challenges: Computationally intensive for high-dimensional data, requires specification of appropriate

priors, and convergence diagnostics can be complex.

Boosted Regression Trees (BRT)
Concept: An ensemble learning technique that combines many weak decision trees to produce a strong predictive model.
Related terms: gradient boosting, learning rate, tree depth.
Explanation: Each successive tree focuses on the residual errors of the previous ensemble, improving accuracy for non-linear and interaction effects in price data.
Example: A BRT model predicts intraday price volatility using features such as renewable output, load forecast error, and market sentiment indicators.
Practical application: Real-time price alerts that adapt to changing market conditions.
Challenges: Sensitive to over-fitting if the number of trees is too large, requires careful hyper-parameter tuning, and can be less transparent than linear models.

Convolutional Neural Network (CNN)
Concept: A deep learning architecture primarily used for processing grid-like data such as images or spatial matrices.
Related terms: kernel, pooling layer, feature map.
Explanation: In electricity markets, CNNs can extract spatial patterns from geographically distributed data (e.g., generation capacity maps, weather radar images) and embed them into price forecasts.
Example: A CNN processes a 2-D temperature field to predict regional price spikes caused by heat waves.
Practical application: Enhancing forecasts for markets where locational marginal pricing depends heavily on spatial congestion.
Challenges: Requires large labeled datasets, high computational resources, and careful design of input representations.

Copula Models
Concept: Statistical tools that model complex dependencies between multiple random variables by linking marginal distributions to a joint multivariate distribution.
Related terms: Gaussian copula, t-copula, tail dependence.
Explanation: Copulas allow separate modeling of marginal behavior (e.g., individual price series) and their dependence structure, which is crucial for joint risk assessment of multiple market zones.
Example: A t-copula captures the strong co-movement of peak prices in neighboring bidding zones during system stress.
Practical application: Portfolio optimization for cross-border electricity traders.
Challenges: Selecting the appropriate copula family, estimating parameters in high dimensions, and ensuring numerical stability.

Dynamic Factor Model (DFM)
Concept: A multivariate time-series approach that reduces a large set of observed variables to a few

unobserved common factors.
Related terms: principal component analysis (PCA), state-space representation, Kalman filter.
Explanation: DFMs capture shared dynamics such as overall market sentiment or system-wide supply constraints that drive electricity prices across regions.
Example: A DFM extracts three latent factors from a panel of hourly prices, load forecasts, and fuel price indices, then uses them to forecast next-day prices.
Practical application: Reducing dimensionality for large-scale forecasting models involving many market zones.
Challenges: Factor identification can be ambiguous, model may miss idiosyncratic shocks, and estimation can be sensitive to outliers.

Ensemble Forecasting
Concept: The combination of multiple independent models to produce a single, often more accurate, prediction.
Related terms: model averaging, stacking, bagging.
Explanation: By aggregating forecasts from ARIMA, ANN, BRT, and other methods, ensembles mitigate individual model biases and improve robustness.
Example: A weighted average of a SARIMA, a Gradient Boosting Machine, and a Long Short-Term Memory network yields lower mean absolute error on a validation set.
Practical application: Daily price forecasts for market participants who require both accuracy and reliability.
Challenges: Determining optimal weights, managing computational overhead, and ensuring diversity among component models.

Extreme Gradient Boosting (XGBoost)
Concept: An optimized implementation of gradient boosting that emphasizes speed and performance.
Related terms: regularization, tree pruning, parallel processing.
Explanation: XGBoost builds additive trees while controlling over-fitting through L1/L2 regularization, making it well-suited for large electricity price datasets with many features.
Example: An XGBoost model predicts hourly day-ahead prices using lagged price, wind forecast error, and market participant order flow.
Practical application: Automated trading algorithms that need rapid retraining as new data arrive.
Challenges: Hyper-parameter tuning can be extensive, and the model may become opaque to regulators.

Feature Engineering
Concept: The process of creating, selecting, and transforming variables to improve model performance.
Related terms: lag features, interaction terms, dimensionality reduction.
Explanation: In electricity price forecasting, features may include time-of-day dummies, rolling averages of load, weather-derived indices, and market sentiment scores. Proper engineering captures domain knowledge and reduces noise.
Example: Constructing a “price spread” feature that measures the difference between peak and off-peak

prices in the previous week.
Practical application: Enhancing the predictive power of machine-learning models without increasing complexity.
Challenges: Risk of data leakage, high-dimensional feature spaces leading to over-fitting, and the need for continuous updates as market structures evolve.

Gaussian Process Regression (GPR)
Concept: A non-parametric Bayesian approach that defines a distribution over functions and provides predictive mean and variance.
Related terms: kernel function, covariance matrix, hyper-parameter optimization.
Explanation: GPR offers flexible modeling of non-linear price dynamics while delivering uncertainty estimates, which are valuable for risk-aware decision making.
Example: Using a radial basis function kernel, a GPR model predicts the probability distribution of next-hour prices given recent load and renewable output.
Practical application: Scenario analysis for capacity planning where confidence intervals guide investment.
Challenges: Computational cost scales cubically with data size, making it unsuitable for very large datasets without approximation techniques.

Gradient Boosting Machine (GBM)
Concept: An ensemble technique that builds models sequentially, each correcting errors of its predecessor.
Related terms: learning rate, shrinkage, loss function.
Explanation: GBM can handle heterogeneous data types and capture complex interactions, making it effective for price series with mixed categorical (e.g., market zone) and continuous (e.g., temperature) inputs.
Example: A GBM forecasts day-ahead prices by minimizing the Huber loss to reduce sensitivity to outliers caused by sudden plant outages.
Practical application: Generating price forecasts for settlement calculations in wholesale markets.
Challenges: Requires careful cross-validation to avoid over-fitting, and training time can be significant for large feature sets.

Hybrid Modeling
Concept: The integration of two or more distinct modeling paradigms—such as statistical and machine-learning methods—into a single forecasting framework.
Related terms: model coupling, cascade architecture, residual learning.
Explanation: Hybrid models exploit the strengths of each component; for instance, an ARIMA captures linear trends while an ANN models residual non-linearities.
Example: A two-stage hybrid where SARIMA forecasts the baseline price, and a Long Short-Term Memory network predicts the remaining error term.
Practical application: Improving forecast accuracy in markets with both predictable seasonal patterns and abrupt, non-linear shocks.

Challenges: Increased system complexity, higher computational demands, and the need for seamless data flow between components.

Long Short-Term Memory (LSTM)
Concept: A recurrent neural network architecture designed to learn long-range dependencies in sequential data.
Related terms: cell state, gate mechanisms, vanishing gradient.
Explanation: LSTMs retain information over extended horizons, making them suitable for capturing temporal patterns such as weekly cycles and multi-day price trends.
Example: An LSTM network predicts 24-hour ahead prices using a sliding window of the past 168 hours of load, price, and wind generation data.
Practical application: Real-time price forecasting for automated bidding in intraday markets.
Challenges: Requires substantial training data, susceptible to over-fitting if network depth is excessive, and hyper-parameter selection (e.g., number of units) is non-trivial.

Markov Switching Model (MSM)
Concept: A time-series model that allows parameters to shift between distinct regimes according to a Markov process.
Related terms: regime-dependent variance, transition probability matrix, latent states.
Explanation: Electricity prices often exhibit high-volatility spikes and low-volatility periods; MSM captures these by switching between “normal” and “spike” regimes.
Example: A two-state MSM estimates separate AR coefficients for each regime and computes the probability of being in a spike regime at each hour.
Practical application: Risk management tools that trigger hedging actions when the spike-regime probability exceeds a threshold.
Challenges: Estimating transition probabilities can be unstable, especially with limited extreme events, and model identification may suffer from local optima.

Monte Carlo Simulation
Concept: A computational technique that uses random sampling to approximate the distribution of possible outcomes.
Related terms: scenario generation, variance reduction, pseudo-random numbers.
Explanation: In price forecasting, Monte Carlo methods generate a large set of price paths based on stochastic models, enabling probabilistic assessment of future market states.
Example: Simulating 10,000 possible day-ahead price trajectories using a mean-reverting jump-diffusion process to evaluate expected revenue for a power plant.
Practical application: Quantifying Value-At-Risk (VaR) for electricity portfolios.
Challenges: Computationally intensive, results depend on the quality of the underlying stochastic model, and convergence may require many iterations.

Neural-Ordinary Differential Equations (Neural ODEs)
Concept: A framework that treats the hidden layers of a neural network as the solution to an ordinary differential equation parameterized by a neural net.
Related terms: continuous-depth models, adaptive solvers, adjoint sensitivity.
Explanation: Neural ODEs provide a flexible way to model continuous-time dynamics of electricity prices, especially when observations are irregularly spaced.
Example: A Neural ODE learns the evolution of price volatility over a day, allowing forecasts at any desired time resolution.
Practical application: High-frequency trading systems that require sub-hourly price predictions.
Challenges: Requires specialized solvers, training can be unstable if the ODE dynamics are stiff, and interpretability remains limited.

Principal Component Analysis (PCA)
Concept: A linear dimensionality-reduction technique that transforms correlated variables into a set of orthogonal components ordered by variance explained.
Related terms: eigenvectors, explained variance, loading matrix.
Explanation: PCA is often used to compress large sets of correlated market variables—such as prices across many nodes—into a few factors that retain most information.
Example: Reducing 50 regional price series to the first three principal components, which are then used as inputs for a downstream regression model.
Practical application: Simplifying model inputs for real-time forecasting platforms with limited computational bandwidth.
Challenges: Assumes linear relationships, may discard important non-linear information, and components can be difficult to interpret economically.

Probabilistic Forecasting
Concept: The generation of full predictive distributions rather than single point estimates.
Related terms: prediction interval, quantile regression, distributional loss.
Explanation: Probabilistic forecasts convey the range of likely price outcomes, enabling market participants to assess risk and make informed bidding decisions.
Example: Using quantile regression forests to produce the 5th, 50th, and 95th percentile forecasts of hourly prices.
Practical application: Designing risk-adjusted bidding strategies that consider the probability of price spikes.
Challenges: Requires calibration of predictive intervals, evaluation metrics such as CRPS are less familiar to practitioners, and computational load can increase.

Quantile Regression
Concept: A regression technique that estimates conditional quantiles of the response variable, allowing asymmetric error modeling.
Related terms: pinball loss, asymmetric Laplace distribution, conditional quantile function.

Explanation: In electricity markets, price distributions are often skewed; quantile regression directly models the tails, providing valuable information for extreme-event planning.
Example: A linear quantile regression model predicts the 90th percentile of day-ahead prices based on forecasted wind generation and fuel cost.
Practical application: Setting reserve margins that reflect the likelihood of high price events.
Challenges: Estimating multiple quantiles simultaneously can be computationally demanding, and model stability may suffer with limited extreme observations.

Reinforcement Learning (RL)
Concept: A machine-learning paradigm where an agent learns to make sequential decisions by maximizing cumulative reward through interaction with an environment.
Related terms: Markov Decision Process (MDP), policy gradient, exploration-exploitation.
Explanation: RL agents can be trained to place bids or schedule generation in response to price forecasts, learning strategies that adapt to market dynamics.
Example: A Deep Q-Network learns optimal bidding policies in a simulated day-ahead market, receiving rewards based on profit and penalty for price violations.
Practical application: Automated trading bots that continuously refine strategies as market conditions evolve.
Challenges: Requires realistic market simulators, risk of over-fitting to simulated data, and convergence can be slow.

Support Vector Regression (SVR)
Concept: A kernel-based regression method that finds a function within a specified error margin (ε-insensitive tube) while maximizing flatness.
Related terms: kernel trick, regularization parameter C, epsilon-insensitive loss.
Explanation: SVR handles non-linear relationships in price data by mapping inputs into high-dimensional feature spaces, often achieving good performance with limited data.
Example: An SVR with a radial basis function kernel predicts hourly prices using lagged load, fuel price, and renewable forecasts.
Practical application: Short-term forecasting for markets where data availability is constrained.
Challenges: Selection of kernel parameters is critical, training scales poorly with large datasets, and model interpretability is modest.

Time-Series Decomposition
Concept: The process of separating a series into trend, seasonal, and residual components.
Related terms: additive model, multiplicative model, STL (Seasonal-Trend decomposition using Loess).
Explanation: Decomposition clarifies underlying patterns in electricity prices, facilitating targeted modeling (e.g., applying ARIMA to the residual after removing seasonality).
Example: Applying STL to a 5-year hourly price series yields a weekly seasonal component, a slowly varying trend, and a noisy remainder used for error modeling.

Practical application: Improving forecast accuracy by modeling each component with the most appropriate technique.
Challenges: Choice between additive and multiplicative forms, handling of irregular holidays, and potential leakage if decomposition uses future information.

Transfer Learning
Concept: The technique of leveraging knowledge gained from training on one domain to improve performance on a related, but different, domain.
Related terms: pre-training, fine-tuning, domain adaptation.
Explanation: A model trained on a well-documented market (e.g., European day-ahead prices) can be fine-tuned with limited data from an emerging market, accelerating model development.
Example: A pre-trained CNN on satellite weather imagery is adapted to forecast price impacts of cloud cover in a new region.
Practical application: Rapid deployment of forecasting tools in markets with scarce historical data.
Challenges: Mismatch between source and target distributions can cause negative transfer, and identifying transferable layers requires expertise.

Wavelet Transform
Concept: A mathematical tool that decomposes a signal into localized frequency components using scaled and shifted wavelet functions.
Related terms: discrete wavelet transform (DWT), mother wavelet, multi-resolution analysis.
Explanation: Wavelets isolate transient spikes and periodic patterns in electricity price series, enabling models to focus on relevant frequency bands.
Example: Applying a Daubechies 4 wavelet to de-noise hourly price data before feeding it to an ANN.
Practical application: Enhancing forecast robustness during sudden market shocks caused by plant outages.
Challenges: Selection of appropriate wavelet family and decomposition level, and reconstruction errors may affect downstream models.

Weather-Adjusted Load Forecasting
Concept: The integration of meteorological variables into load prediction models to improve accuracy.
Related terms: temperature-elasticity, degree-day index, climate normalization.
Explanation: Since electricity demand is highly sensitive to temperature, humidity, and wind speed, incorporating these factors refines the price forecast that depends on load.
Example: A linear regression model uses heating-degree-days and cooling-degree-days as predictors for residential load, which feeds into a downstream price model.
Practical application: Adjusting day-ahead price forecasts during heatwaves or cold snaps.
Challenges: Weather forecast errors propagate to load and price predictions, and extreme weather events may lie outside historical patterns.

Zero-Inflated Models

Concept: Statistical models designed for count data that contain an excess of zero observations.
Related terms: zero-inflated Poisson (ZIP), hurdle model, over-dispersion.
Explanation: In certain markets, price spikes may be rare, leading to many zero-change periods; zero-inflated models capture both the probability of a zero change and the distribution of non-zero changes.
Example: A zero-inflated Gaussian model predicts hourly price changes, separating the "no-change" probability from the continuous distribution of actual changes.
Practical application: Improving forecast calibration for markets with frequent price plateaus.
Challenges: Parameter estimation can be unstable, especially when the proportion of zeros varies over time, and model selection between ZIP and hurdle approaches requires careful testing.

Hybrid ARIMA-ANN Model
Concept: A specific combination where an ARIMA model captures linear components and an ANN models the residual non-linear patterns.
Related terms: residual learning, stacked modeling, error correction.
Explanation: The two-stage approach first forecasts the price series with ARIMA, then feeds the residuals into an ANN for refinement, often yielding lower error metrics than either model alone.
Example: An ARIMA(1,1,1) predicts the baseline, while a feed-forward ANN with three hidden nodes reduces the mean absolute percentage error by 12 % on a test set.
Practical application: Day-ahead price forecasting where both trend and sudden non-linear shocks are present.
Challenges: Requires synchronization of training windows, risk of error propagation from the first stage, and increased computational workload.

Kernel Density Estimation (KDE)
Concept: A non-parametric technique for estimating the probability density function of a random variable.
Related terms: bandwidth selection, Gaussian kernel, smooth density.
Explanation: KDE provides a flexible way to model the empirical distribution of price residuals, which can be used to generate realistic Monte Carlo scenarios.
Example: Estimating the density of hourly price forecast errors using a Gaussian kernel with Silverman's rule-of-thumb bandwidth.
Practical application: Constructing scenario trees for stochastic optimization of generation dispatch.
Challenges: Choice of bandwidth critically affects smoothness, high-dimensional KDE suffers from the curse of dimensionality, and computational cost rises with sample size.

Long-Memory Processes
Concept: Time-series processes whose autocorrelation function decays hyperbolically, indicating persistent dependence over long horizons.
Related terms: fractional integration, Hurst exponent, ARFIMA.
Explanation: Electricity price series sometimes exhibit long-memory behavior, especially in markets with

structural constraints; models that account for this can improve multi-day forecasts.
Example: An ARFIMA(1,d,0) model with d = 0.35 captures the slow decay of autocorrelation in weekly average prices.
Practical application: Strategic planning for hedging contracts that span several weeks.
Challenges: Estimating the fractional differencing parameter accurately, and long-memory models may be sensitive to structural breaks.

Multivariate GARCH (MGARCH)
Concept: An extension of the Generalized Autoregressive Conditional Heteroskedasticity model to multiple correlated series.
Related terms: BEKK model, dynamic conditional correlation (DCC), volatility spillover.
Explanation: MGARCH captures time-varying volatility and cross-market contagion, essential for assessing joint price risk across interconnected bidding zones.
Example: A DCC-GARCH model estimates evolving correlations between German and French day-ahead prices, revealing higher co-movement during peak demand periods.
Practical application: Portfolio risk assessment for cross-border electricity traders.
Challenges: Parameter estimation is computationally intensive, and model convergence may fail with many series.

Neural Architecture Search (NAS)
Concept: An automated process that explores a space of neural network structures to identify optimal architectures for a given task.
Related terms: search space, reinforcement-learning controller, gradient-based search.
Explanation: NAS can discover specialized network topologies—such as hybrid convolution-recurrent models—that outperform manually designed architectures for electricity price forecasting.
Example: Using a reinforcement-learning agent to propose and evaluate candidate models, the resulting architecture achieves a 5 % reduction in RMSE compared with a baseline LSTM.
Practical application: Deploying high-performance models without extensive manual experimentation.
Challenges: Requires substantial computational resources, risk of over-fitting to the validation set, and discovered architectures may be difficult to interpret.

Quantitative Risk Management (QRM)
Concept: The systematic use of statistical and mathematical models to measure, monitor, and control financial risk.
Related terms: Value-At-Risk (VaR), Conditional VaR (CVaR), stress testing.
Explanation: In electricity markets, QRM leverages price forecasts and their uncertainty to assess exposure, set risk limits, and allocate capital.
Example: Computing 1-day 95 % VaR for a portfolio of forward contracts using probabilistic forecasts generated by a Bayesian neural network.
Practical application: Regulatory reporting and internal risk dashboards for utilities and traders.

Challenges: Dependent on the quality of forecast distributions, model risk from misspecified assumptions, and computational burden for large portfolios.

Recursive Feature Elimination (RFE)
Concept: A wrapper method that iteratively removes the least important features based on model performance.
Related terms: feature ranking, cross-validation, model-agnostic importance.
Explanation: RFE helps identify a parsimonious set of variables that retain predictive power, reducing over-fitting and computation time in complex forecasting models.
Example: Applying RFE with a Random Forest estimator selects 12 out of 50 candidate features for a day-ahead price model, improving out-of-sample MAE by 8 %.
Practical application: Streamlining feature pipelines for real-time forecasting services.
Challenges: Computationally expensive for large feature spaces, results can vary with the underlying estimator, and may discard variables that become important under regime changes.

Stochastic Volatility Models
Concept: Models that treat volatility itself as a random process, often using latent variables.
Related terms: Heston model, GARCH-type volatility, latent volatility.
Explanation: Electricity prices exhibit abrupt volatility spikes; stochastic volatility models capture this by allowing variance to evolve stochastically, improving density forecasts.
Example: A stochastic volatility model with a mean-reverting variance process generates predictive intervals that better cover observed price spikes.
Practical application: Pricing of volatility-linked derivatives such as electricity options.
Challenges: Parameter estimation requires advanced techniques (e.g., particle filtering), and calibration can be sensitive to outlier events.

TensorFlow Probability (TFP)
Concept: A library that extends TensorFlow with probabilistic modeling and statistical inference capabilities.
Related terms: variational inference, probabilistic layers, Monte Carlo integration.
Explanation: TFP enables building deep Bayesian models—such as Bayesian LSTMs—that produce calibrated predictive distributions for electricity prices.
Example: Implementing a Bayesian neural network with TFP that outputs mean and variance for each forecasted hour, facilitating risk-aware bidding.
Practical application: Embedding uncertainty quantification directly into production-grade forecasting pipelines.
Challenges: Steeper learning curve for practitioners unfamiliar with probabilistic programming, and increased training time compared with deterministic networks.

Unsupervised Clustering of Price Regimes
Concept: The identification of distinct market states without labeled data, using algorithms that group

similar observations.
Related terms: K-means, Gaussian mixture models (GMM), spectral clustering.
Explanation: Clustering historical price patterns reveals regimes such as “normal,” “high-renewable,” and “constrained,” which can be used as inputs for regime-dependent forecasting models.
Example: A GMM with three components classifies daily price curves, and each regime feeds a specialized ARIMA model.
Practical application: Adaptive forecasting that switches models based on detected regime.
Challenges: Determining the optimal number of clusters, handling overlapping regimes, and ensuring clusters remain stable over time.

Variational Autoencoder (VAE)
Concept: A generative neural network that learns a low-dimensional latent representation of data while enforcing a probabilistic structure.
Related terms: encoder-decoder, KL divergence, latent space sampling.
Explanation: VAEs can compress high-dimensional price and weather data into latent variables that capture essential dynamics, which are then used for downstream forecasting.
Example: Training a VAE on 5-year hourly price-weather matrices, then feeding the 10-dimensional latent vectors into a Gradient Boosting model for next-day price prediction.
Practical application: Reducing data storage and transmission costs for distributed forecasting agents.
Challenges: Balancing reconstruction loss with regularization, risk of posterior collapse, and latent variables may lack direct interpretability.

Wavelet-Based Neural Networks
Concept: Hybrid models that first decompose time series using wavelets and then feed the resulting coefficients into a neural network.
Related terms: multi-resolution analysis, coefficients as features, denoising.
Explanation: Wavelet decomposition isolates high-frequency spikes and low-frequency trends; the neural network learns separate mappings for each frequency band, enhancing forecast accuracy.
Example: Applying a Daubechies 8 wavelet to split hourly price data into approximation and detail components, then training two parallel feed-forward networks whose outputs are summed.
Practical application: Improving robustness of price forecasts during sudden renewable output fluctuations.
Challenges: Selecting appropriate wavelet family and decomposition level, and increased model complexity can hinder real-time deployment.

X-GBoost with Quantile Loss
Concept: An adaptation of XGBoost that optimizes the pinball loss function to directly estimate conditional quantiles.
Related terms: gradient quantile regression, asymmetric loss, prediction intervals.
Explanation: By minimizing quantile loss, the model produces forecasts for specific percentiles (e.g., 5 % and 95 %), delivering a full predictive interval without separate post-processing.

Example: Training XGBoost with a 0.9 quantile loss yields the 90th-percentile price forecast, useful for risk-averse bidding strategies.
Practical application: Real-time generation of confidence bands for intraday market participants.
Challenges: Requires careful tuning of learning rate and tree depth for each quantile, and the model may produce crossing quantiles if not constrained.

Zero-Shot Learning for New Market Zones
Concept: A method that enables a model to make predictions for previously unseen categories by leveraging auxiliary information.
Related terms: semantic embeddings, attribute-based classification, domain generalization.
Explanation: When a new bidding zone is introduced, zero-shot learning uses features such as grid topology and generation mix to infer price behavior without historical price data.
Example: Mapping a new zone's characteristics to a latent space shared with existing zones, then applying a pre-trained price forecasting model to generate initial forecasts.
Practical application: Rapidly providing price estimates for newly formed market regions or after regulatory boundary changes.
Challenges: Quality of auxiliary descriptors heavily influences performance, and model confidence may be low until real data become available.

Cross-Validation for Time Series
Concept: A validation technique that respects temporal ordering, avoiding leakage of future information into training folds.
Related terms: rolling-origin evaluation, blocked cross-validation, expanding window.
Explanation: Proper cross-validation ensures that forecast performance metrics reflect realistic out-of-sample conditions, which is critical for electricity price models that exhibit strong autocorrelation.
Example: Using a rolling window of 30 days for training and the subsequent 7 days for testing, repeated across the dataset to compute average MAE.
Practical application: Model selection and hyper-parameter tuning for production-grade forecasting systems.
Challenges: Computational cost increases with many folds, and choosing window sizes balances bias and variance.

Dynamic Bayesian Networks (DBN)
Concept: Probabilistic graphical models that represent temporal dependencies among variables using directed acyclic graphs.
Related terms: hidden Markov model, belief propagation, temporal inference.
Explanation: DBNs can model the joint evolution of price, load, and weather variables, capturing both causal and stochastic relationships over time.
Example: A DBN with nodes for temperature, load, and price at each hour, where price depends on the previous hour's price and current load.

Practical application: Scenario generation for stochastic optimization of generation schedules.
Challenges: Exact inference is often intractable for large networks, requiring approximate methods that may lose accuracy.

Energy-Based Models (EBM)
Concept: A class of models that define an energy function over configurations, with lower energy corresponding to higher probability.
Related terms: contrastive divergence, score matching, deep energy models.
Explanation: EBMs can learn complex joint distributions of price and exogenous variables without explicit likelihood, useful for generating realistic price samples.
Example: Training an EBM on historical price-weather pairs, then sampling from the learned energy landscape to create synthetic price scenarios for Monte Carlo analysis.
Practical application: Augmenting limited historical data with realistic synthetic observations for model training.
Challenges: Training stability issues, difficulty in evaluating normalization constants, and limited support in mainstream libraries.

Hybrid SARIMA-LSTM Model
Concept: A combined approach where a Seasonal ARIMA model captures linear seasonal patterns and an LSTM models the remaining non-linear dynamics.
Related terms: seasonal decomposition, residual LSTM, stacked forecasting.
Explanation: SARIMA handles weekly and annual cycles common in electricity prices, while LSTM learns complex interactions such as weather-driven spikes.
Example: A SARIMA(0,1,1)(1,0,1)[24] generates a baseline, and