Global Certificate Course in Electricity Price Forecasting

Machine Learning Techniques For Load And Price Prediction

ARIMA (AutoRegressive Integrated Moving Average)
Related terms: SARIMA, Box-Jenkins methodology, differencing.
Explanation: A statistical model that captures autocorrelation in time-series data by combining autoregressive (AR) terms, differencing (I) to achieve stationarity, and moving-average (MA) components.
Example: Forecasting hourly electricity demand using past demand values and residual errors.
Practical application: Day-ahead load prediction for utility scheduling.
Challenges: Selecting appropriate order (p, d, q) and handling non-linear patterns common in price series.

ARMAX (AutoRegressive Moving Average with Exogenous Inputs)
Related terms: ARIMA, exogenous variables, multivariate time series.
Explanation: Extends ARIMA by incorporating external regressors (e.g., temperature, fuel price) that influence the target series.
Example: Including weather forecasts as exogenous inputs to improve load forecasts.
Practical application: Short-term price forecasting where market fundamentals affect prices.
Challenges: Identifying relevant exogenous variables and avoiding multicollinearity.

Autoencoder
Related terms: Dimensionality reduction, neural network, reconstruction error.
Explanation: An unsupervised neural network that learns a compressed representation (encoding) of input data and reconstructs it, useful for feature extraction.
Example: Reducing high-dimensional smart-meter data to a lower-dimensional latent space before feeding into a regression model.
Practical application: Denoising load profiles and detecting anomalies.
Challenges: Choosing bottleneck size and preventing over-compression that loses important patterns.

Bagging (Bootstrap Aggregating)
Related terms: Random Forest, ensemble learning, variance reduction.
Explanation: Generates multiple training subsets via bootstrapping, trains a base learner on each, and aggregates predictions (usually by averaging) to improve stability.
Example: Bagging decision trees for robust short-term price forecasts.
Practical application: Reducing sensitivity to outliers in load prediction.
Challenges: Computational cost increases with number of base models.

Bayesian Optimization
Related terms: Hyperparameter tuning, surrogate model, acquisition function.
Explanation: Treats hyperparameter search as a probabilistic problem, iteratively building a surrogate (often Gaussian Process) to predict performance and selecting new points via an acquisition function.
Example: Optimizing learning rate and depth of a Gradient Boosting model for price forecasting.
Practical application: Efficiently finding optimal model configurations with limited training runs.
Challenges: Requires careful choice of prior and may struggle with high-dimensional hyperparameter spaces.

Bayesian Neural Network
Related terms: Uncertainty quantification, variational inference, Monte Carlo dropout.
Explanation: Extends conventional neural networks by placing probability distributions over weights, yielding predictive distributions rather than point estimates.
Example: Providing confidence intervals for day-ahead load forecasts.
Practical application: Risk-aware decision making in market bidding.
Challenges: Computationally intensive training and inference.

Boosting
Related terms: Gradient Boosting, XGBoost, AdaBoost, ensemble methods.
Explanation: Sequentially trains weak learners, each focusing on errors of its predecessor, and combines them into a strong predictor.
Example: Using XGBoost to predict hourly electricity prices based on historical price, demand, and weather features.
Practical application: Capturing complex non-linear relationships in price data.
Challenges: Prone to overfitting if not regularized; requires careful early stopping.

CatBoost
Related terms: Gradient Boosting, categorical encoding, ordered boosting.
Explanation: A gradient-boosting library that handles categorical variables natively and reduces prediction shift via ordered boosting.
Example: Incorporating fuel type categories (coal, gas, renewable) directly into a price forecasting model.
Practical application: Faster development cycles when many categorical market features exist.
Challenges: Memory usage can be high for very large datasets.

Clustering
Related terms: K-means, hierarchical clustering, silhouette score.
Explanation: Unsupervised technique that groups similar observations based on distance metrics, often used for pattern discovery.
Example: Segmenting customers into load-profile groups to build tailored forecasting models.
Practical application: Reducing model complexity by applying a single model per cluster.

Challenges: Determining optimal number of clusters and handling high-dimensional data.

Convolutional Neural Network (CNN)
Related terms: Convolution, pooling, feature maps.
Explanation: Neural architecture that applies convolutional filters to capture local patterns, traditionally used for images but adaptable to time-series via 1-D convolutions.
Example: Learning short-term temporal patterns in price spikes from raw price series.
Practical application: Automated feature extraction from high-frequency market data.
Challenges: Requires sufficient data to avoid over-parameterization; filter design can be non-intuitive for non-spatial data.

Cross-validation
Related terms: K-fold, time-series split, model assessment.
Explanation: Technique for estimating model performance by partitioning data into training and validation sets multiple times.
Example: Using a rolling-origin evaluation to assess a load forecasting model across multiple weeks.
Practical application: Preventing over-optimistic error estimates in model selection.
Challenges: Standard K-fold breaks temporal order; special time-aware splits are needed.

Data Augmentation
Related terms: Synthetic data, oversampling, SMOTE.
Explanation: Generates additional training samples by applying transformations, improving model robustness, especially when data are scarce.
Example: Adding small Gaussian noise to historic load curves to expand the training set.
Practical application: Enhancing model generalization for rare price spikes.
Challenges: Augmented data may not reflect true underlying dynamics, risking bias.

Data Imputation
Related terms: Missing data handling, interpolation, forward fill.
Explanation: Strategies to fill gaps in datasets, essential for continuous time-series modeling.
Example: Using linear interpolation to replace missing hourly price values due to market outages.
Practical application: Maintaining model continuity for real-time forecasting.
Challenges: Imputation can introduce artificial smoothness, affecting variance estimates.

Decision Tree
Related terms: CART, pruning, impurity measures.
Explanation: Tree-structured model that splits data based on feature thresholds to minimize impurity (e.g., Gini, entropy).
Example: Predicting price direction (up/down) using a shallow tree on recent price change and demand.
Practical application: Interpretable baseline for load and price forecasting.

Challenges: High variance; prone to overfitting without pruning or ensemble methods.

Dimensionality Reduction
Related terms: Principal Component Analysis (PCA), t-SNE, autoencoder.
Explanation: Techniques that transform high-dimensional data into a lower-dimensional space while preserving essential information.
Example: Applying PCA to a set of weather variables to reduce multicollinearity before model training.
Practical application: Faster training of complex models and mitigation of the curse of dimensionality.
Challenges: Loss of interpretability; selecting the number of components is non-trivial.

Early Stopping
Related terms: Regularization, validation loss, overfitting prevention.
Explanation: Halts training when performance on a validation set ceases to improve, preventing over-learning.
Example: Stopping LSTM training after 10 epochs with no validation loss reduction.
Practical application: Efficient training of deep networks for price forecasting.
Challenges: Requires a reliable validation set that reflects future data distribution.

Ensemble Kalman Filter (EnKF)
Related terms: Data assimilation, state estimation, Monte Carlo.
Explanation: A recursive Bayesian filter that uses an ensemble of system states to estimate the posterior distribution, suitable for non-linear models.
Example: Updating load forecasts in real-time as new smart-meter readings arrive.
Practical application: Improving short-term forecasts under measurement noise.
Challenges: Requires careful ensemble size selection; computationally intensive for large systems.

Feature Engineering
Related terms: Lag features, rolling statistics, interaction terms.
Explanation: Process of creating informative variables from raw data to improve model performance.
Example: Constructing a 24-hour lag of load and a moving-average of temperature over the past 6 hours.
Practical application: Enhancing predictive power of simple linear models for price forecasting.
Challenges: Risk of data leakage if future information is inadvertently used.

Feature Scaling
Related terms: Normalization, standardization, Min-Max scaling.
Explanation: Rescales features to a common range, facilitating model convergence, especially for gradient-based algorithms.
Example: Scaling price and load values to [0,1] before feeding into a neural network.
Practical application: Stabilizing training of deep models.
Challenges: Scaling parameters must be stored and applied consistently during inference.

Forecast Horizon
Related terms: Short-term, mid-term, long-term, lead time.
Explanation: The future time interval for which predictions are generated, ranging from minutes to years.
Example: A 24-hour ahead forecast for day-ahead market bidding.
Practical application: Aligning model choice (e.g., ARIMA vs. LSTM) with horizon length.
Challenges: Accuracy typically degrades as horizon extends; different features become relevant.

Gaussian Process (GP)
Related terms: Kernel functions, Bayesian non-parametrics, kriging.
Explanation: A probabilistic model that defines a distribution over functions, providing mean predictions and uncertainty estimates.
Example: Modeling the smooth component of electricity price curves with a radial-basis-function kernel.
Practical application: Quantifying forecast uncertainty for risk-aware trading.
Challenges: Computational complexity scales cubically with data size; requires kernel selection.

Gradient Boosting Machine (GBM)
Related terms: Boosting, learning rate, tree depth.
Explanation: Builds an additive model of weak learners (typically decision trees) where each new tree fits the residuals of the combined previous trees.
Example: Using GBM to predict hourly LMPs based on historical prices, load, and renewable output.
Practical application: Capturing non-linear interactions in price formation.
Challenges: Sensitive to hyperparameters; may overfit without regularization.

Grid Search
Related terms: Hyperparameter tuning, exhaustive search, cross-validation.
Explanation: Systematically evaluates a predefined set of hyperparameter combinations to identify the best configuration.
Example: Testing learning rates {0.01,0.05,0.1} and max depths {3,5,7} for a Random Forest.
Practical application: Baseline approach for model optimization.
Challenges: Computationally expensive for large search spaces; may miss optimal regions between grid points.

Hidden Markov Model (HMM)
Related terms: State space model, Markov chain, emission probabilities.
Explanation: Statistical model where observable sequences are generated by underlying hidden states following Markov dynamics.
Example: Modeling regime shifts in electricity prices (e.g., high-volatility vs. low-volatility periods).
Practical application: Detecting price regime changes for dynamic strategy adjustment.
Challenges: Determining appropriate number of hidden states; parameter estimation can be sensitive to initialization.

Hyperparameter
Related terms: Model configuration, tuning, learning rate.
Explanation: Settings external to the learning algorithm that control model capacity, regularization, and optimization behavior.
Example: Number of trees in a Random Forest, dropout rate in a neural network.
Practical application: Adjusted to balance bias-variance trade-off.
Challenges: No universal rule; requires systematic search and validation.

Independent Component Analysis (ICA)
Related terms: Blind source separation, signal decomposition, statistical independence.
Explanation: Decomposes multivariate signals into statistically independent components, often used to isolate underlying sources.
Example: Separating load consumption patterns from noise in aggregated smart-meter data.
Practical application: Enhancing feature extraction for load forecasting.
Challenges: Assumes non-Gaussian sources; component ordering may be ambiguous.

Kalman Filter
Related terms: Linear state-space model, prediction-update cycle, optimal estimator.
Explanation: Recursive algorithm that provides optimal estimates of hidden states in linear Gaussian systems, combining model predictions with noisy observations.
Example: Real-time adjustment of load forecasts as new SCADA measurements arrive.
Practical application: Reducing forecast latency in operational environments.
Challenges: Assumes linearity and Gaussian noise; extensions needed for non-linear dynamics.

K-means Clustering
Related terms: Centroids, Euclidean distance, inertia.
Explanation: Partitions data into K clusters by minimizing within-cluster sum of squares, assigning each point to the nearest centroid.
Example: Grouping daily load curves into typical “weekday”, “weekend”, and “holiday” shapes.
Practical application: Building specialized models per cluster to improve accuracy.
Challenges: Requires pre-specifying K; sensitive to outliers and initial centroid placement.

L1 Regularization (Lasso)
Related terms: Sparsity, feature selection, penalty term.
Explanation: Adds the absolute value of coefficients to the loss function, encouraging many coefficients to become exactly zero.
Example: Selecting the most influential weather variables for a linear price model.
Practical application: Simplifying models for interpretability and reducing overfitting.
Challenges: May arbitrarily drop correlated predictors; selection stability can vary with data splits.

L2 Regularization (Ridge)
Related terms: Weight decay, shrinkage, penalty term.
Explanation: Adds the squared magnitude of coefficients to the loss function, shrinking them toward zero but rarely eliminating them entirely.
Example: Stabilizing a multivariate regression of price on demand and fuel costs.
Practical application: Improves generalization when many correlated features exist.
Challenges: Does not produce sparse solutions; selecting regularization strength is critical.

LightGBM
Related terms: Gradient Boosting, leaf-wise growth, histogram based.
Explanation: An efficient gradient-boosting framework that grows trees leaf-wise and uses histogram binning for speed and memory savings.
Example: Forecasting 15-minute ahead prices with millions of observations.
Practical application: Real-time price prediction where latency matters.
Challenges: Can overfit on small datasets; requires careful tuning of leaf-wise parameters.

Long Short-Term Memory (LSTM)
Related terms: Recurrent Neural Network (RNN), gates, cell state.
Explanation: A type of RNN that introduces input, forget, and output gates to preserve long-range dependencies while mitigating vanishing gradients.
Example: Modeling daily load patterns with an LSTM that ingests past 168 hours of load and weather.
Practical application: Accurate multi-step ahead forecasts for day-ahead markets.
Challenges: Requires substantial training data; hyperparameter selection (layers, units) impacts performance.

MAE (Mean Absolute Error)
Related terms: Evaluation metric, absolute loss, forecast accuracy.
Explanation: Average of absolute differences between predicted and actual values, providing an intuitive measure of average error magnitude.
Example: MAE of 2.3 MW for a 24-hour load forecast.
Practical application: Benchmarking models where penalizing large errors is less critical than overall bias.
Challenges: Does not penalize large outliers as heavily as squared-error metrics.

MAPE (Mean Absolute Percentage Error)
Related terms: Relative error, scaling, forecast evaluation.
Explanation: Mean of absolute percentage differences, expressing error as a proportion of actual values.
Example: MAPE of 4 % for hourly price forecasts.
Practical application: Comparing models across different scales (e.g., low-load vs. high-load periods).
Challenges: Undefined when actual values approach zero; can be skewed by small denominators.

Monte Carlo Simulation

Related terms: Stochastic modeling, scenario generation, probabilistic forecast.
Explanation: Generates a large number of random scenarios based on probability distributions to assess uncertainty and risk.
Example: Simulating 10 000 possible price paths for a given day using estimated volatility.
Practical application: Estimating Value-at-Risk (VaR) for market participants.
Challenges: Computationally heavy; results depend on assumed distributions.

Multi-Layer Perceptron (MLP)
Related terms: Feedforward neural network, hidden layers, activation function.
Explanation: A classic neural network consisting of an input layer, one or more hidden layers, and an output layer, where each neuron applies a weighted sum followed by a non-linear activation.
Example: Predicting next-hour price using an MLP with ReLU activation and three hidden layers.
Practical application: Baseline deep learning model for quick prototyping.
Challenges: Limited ability to capture temporal dependencies without engineered lag features.

Multicollinearity
Related terms: Correlated predictors, variance inflation factor (VIF), linear regression.
Explanation: Situation where two or more explanatory variables are highly linearly related, inflating variance of coefficient estimates.
Example: Temperature and humidity often exhibit multicollinearity in weather-driven load models.
Practical application: Diagnosing and remedying via feature selection or PCA.
Challenges: Can obscure true predictor importance and destabilize regression models.

Neural Architecture Search (NAS)
Related terms: AutoML, hyperparameter optimization, model discovery.
Explanation: Automated process that explores the space of neural network architectures to identify high-performing designs.
Example: Using NAS to discover an optimal combination of convolutional and recurrent layers for price forecasting.
Practical application: Reducing manual effort in designing deep models.
Challenges: Extremely computationally expensive; risk of overfitting to validation set.

Normalization
Related terms: Feature scaling, zero-mean unit-variance, standardization.
Explanation: Adjusts data to have a particular statistical property (commonly zero mean and unit variance), facilitating model training.
Example: Normalizing load values before feeding them into a Support Vector Machine.
Practical application: Improves convergence speed of gradient-based optimizers.
Challenges: Must apply same transformation to future data; outliers can affect mean and variance.

Online Learning
Related terms: Incremental learning, streaming data, concept drift.
Explanation: Model updates occur continuously as new data arrive, without retraining from scratch.
Example: Updating a price prediction model hourly as market data streams in.
Practical application: Maintaining up-to-date forecasts in fast-changing markets.
Challenges: Balancing stability and adaptability; handling noisy updates.

Outlier Detection
Related terms: Anomaly detection, robust statistics, z-score.
Explanation: Identifies observations that deviate markedly from the majority, which may indicate data errors or rare events.
Example: Flagging a sudden 30 % price jump as a potential market anomaly.
Practical application: Cleaning training data to avoid biasing models.
Challenges: Distinguishing genuine extreme events from erroneous measurements.

Overfitting
Related terms: High variance, regularization, model complexity.
Explanation: When a model captures noise in training data, leading to poor generalization on unseen data.
Example: A deep LSTM with many layers that perfectly fits historical price but fails on new days.
Practical application: Recognizing and mitigating via validation, early stopping, or dropout.
Challenges: Detecting overfitting early in time-series contexts where validation data may be limited.

Parametric Model
Related terms: Linear regression, ARIMA, assumptions.
Explanation: Model defined by a finite set of parameters with a predetermined functional form.
Example: A linear regression of price on load and fuel cost with fixed coefficients.
Practical application: Simpler interpretation and faster training.
Challenges: May be too rigid to capture complex non-linear relationships.

Partial Autocorrelation Function (PACF)
Related terms: ACF, lag selection, Box-Jenkins.
Explanation: Measures correlation between a time series and its lag after removing the effects of intermediate lags, aiding in AR order identification.
Example: PACF spikes at lag 24 suggest a daily seasonality component in load data.
Practical application: Guiding ARIMA model specification.
Challenges: Requires stationary series; noisy data can produce ambiguous spikes.

Peak Load
Related terms: Base load, demand curve, capacity planning.
Explanation: The maximum electricity demand observed over a specific period, often dictating system

capacity requirements.
Example: A 45 GW peak observed during a summer afternoon.
Practical application: Forecasting peaks to schedule generation and manage market pricing.
Challenges: Peaks are rare and highly sensitive to weather and behavioral factors.

Permutation Importance
Related terms: Feature importance, model interpretability, SHAP.
Explanation: Assesses the impact of each feature by randomly permuting its values and measuring the resulting drop in model performance.
Example: Determining that temperature has the highest permutation importance for load forecasts.
Practical application: Guiding feature selection and model explanation.
Challenges: Computationally expensive for large datasets; can be biased for correlated features.

Pinball Loss
Related terms: Quantile regression, probabilistic forecast, asymmetric loss.
Explanation: Measures accuracy of quantile forecasts by penalizing under- and over-predictions asymmetrically according to the quantile level.
Example: Pinball loss of 0.12 for the 0.9-quantile price forecast.
Practical application: Evaluating probabilistic price forecasts required for risk management.
Challenges: Requires multiple quantile predictions; interpretation can be less intuitive than MSE.

Power Law Distribution
Related terms: Heavy-tailed, Pareto, extreme events.
Explanation: Statistical distribution where the probability of an event scales as a power of its size, often observed in price spikes.
Example: Modeling the tail of price spikes with a Pareto distribution.
Practical application: Risk assessment for extreme price events.
Challenges: Parameter estimation is sensitive to sample size; tail behavior may change over time.

Probabilistic Forecasting
Related terms: Prediction intervals, quantile regression, ensemble methods.
Explanation: Generates a full distribution (or set of quantiles) for future values rather than a single point estimate.
Example: Producing the 5th, 50th, and 95th percentile forecasts for hourly load.
Practical application: Enabling market participants to hedge against uncertainty.
Challenges: Requires more sophisticated evaluation metrics (e.g., CRPS) and larger training data.

Principal Component Analysis (PCA)
Related terms: Eigenvectors, variance explained, orthogonal transformation.
Explanation: Linear technique that transforms correlated variables into a set of uncorrelated components

ordered by explained variance.
Example: Reducing 10 weather variables to 3 principal components before regression.
Practical application: Mitigating multicollinearity and speeding up training.
Challenges: Components may lack physical interpretability; linear assumption may miss non-linear structures.

Random Forest
Related terms: Bagging, decision trees, feature importance.
Explanation: Ensemble of decorrelated decision trees built on bootstrapped samples, with random feature selection at each split, producing robust predictions.
Example: Predicting day-ahead price using a Random Forest with 500 trees.
Practical application: Handles mixed data types and provides built-in importance scores.
Challenges: Can be memory intensive; less effective for extrapolation beyond training range.

Recurrent Neural Network (RNN)
Related terms: Temporal dependencies, hidden state, vanishing gradient.
Explanation: Neural architecture designed for sequential data, where each step's output depends on previous hidden state.
Example: Modeling hourly price sequences with a simple RNN layer.
Practical application: Capturing short-term temporal patterns.
Challenges: Training instability due to vanishing/exploding gradients; LSTM/GRU often preferred.

Regularization
Related terms: L1, L2, dropout, early stopping.
Explanation: Techniques that add constraints or penalties to the loss function to prevent overfitting and improve generalization.
Example: Applying L2 regularization to the weights of a neural network for price forecasting.
Practical application: Stabilizing models with limited training data.
Challenges: Selecting appropriate regularization strength; overly strong regularization may underfit.

Residual Analysis
Related terms: Autocorrelation of errors, model diagnostics, Ljung-Box test.
Explanation: Examination of the differences between observed and predicted values to assess model adequacy.
Example: Detecting remaining seasonal patterns in residuals of an ARIMA load model.
Practical application: Guiding model refinement and identifying missing dynamics.
Challenges: Requires sufficient residual data; statistical tests may have limited power for short series.

Ridge Regression
Related terms: L2 regularization, shrinkage, multicollinearity.

Explanation: Linear regression variant that penalizes the sum of squared coefficients, reducing variance at the cost of some bias.
Example: Stabilizing price prediction coefficients when fuel price and carbon price are highly correlated.
Practical application: Improves out-of-sample performance for high-dimensional feature sets.
Challenges: Does not perform variable selection; all predictors remain in the model.

Rolling Window Validation
Related terms: Time-series split, walk-forward, model evaluation.
Explanation: Sequentially moves a fixed-size training window forward in time, training and testing at each step to mimic real-world forecasting.
Example: Using a 30-day training window to predict the next 24 hours, then rolling forward by one day.
Practical application: Provides realistic performance estimates for operational models.
Challenges: Computationally demanding; choice of window size influences bias-variance trade-off.

Scenario Analysis
Related terms: What-if modeling, stress testing, Monte Carlo.
Explanation: Evaluates model performance under a set of predefined alternative future conditions.
Example: Assessing load forecasts under a heatwave scenario versus a normal summer day.
Practical application: Planning generation capacity and market strategies under extreme conditions.
Challenges: Defining realistic and comprehensive scenarios; may require additional data sources.

Seasonality
Related terms: Periodic patterns, Fourier terms, seasonal decomposition.
Explanation: Regular, repeating fluctuations in a time series occurring at fixed intervals (e.g., daily, weekly, yearly).
Example: Higher electricity demand every weekday morning due to work-day routines.
Practical application: Incorporating seasonal components in ARIMA or neural network models to improve accuracy.
Challenges: Seasonality can shift over time (e.g., due to policy changes), requiring adaptive modeling.

Shapley Additive Explanations (SHAP)
Related terms: Model interpretability, feature contribution, game theory.
Explanation: Provides consistent, locally accurate attribution values for each feature based on cooperative game theory.
Example: Explaining that a sudden temperature rise contributed +0.8 MW to the load forecast for a specific hour.
Practical application: Building trust with stakeholders by elucidating black-box model decisions.
Challenges: Computationally intensive for large models; approximations may be needed.

Signal-to-Noise Ratio (SNR)

Related terms: Data quality, measurement error, filtering.
Explanation: Ratio of the power of the desired signal to the power of background noise, indicating data reliability.
Example: High SNR in SCADA measurements versus lower SNR in low-voltage smart-meter data.
Practical application: Determining appropriate preprocessing steps (e.g., smoothing) before model training.
Challenges: Estimating noise characteristics in real-world electricity data can be difficult.

Simple Exponential Smoothing (SES)
Related terms: Forecasting, smoothing parameter, level component.
Explanation: Forecasting method that applies exponentially decreasing weights to past observations, suitable for data without trend or seasonality.
Example: Using SES to predict short-term price when the market is relatively stable.
Practical application: Baseline comparison for more complex models.
Challenges: Cannot capture trends or seasonal patterns; smoothing parameter selection is critical.

Single-Step Forecast
Related terms: One-ahead prediction, iterative forecasting, direct method.
Explanation: Predicts the immediate next value in a series, often used as a building block for multi-step forecasts.
Example: Forecasting the price for the next 15-minute interval.
Practical application: Real-time bidding decisions in intraday markets.
Challenges: Errors can accumulate when iteratively feeding predictions back into the model for longer horizons.

Spatial-Temporal Modeling
Related terms: Geostatistics, spatio-temporal kriging, graph neural networks.
Explanation: Incorporates both spatial relationships (e.g., between nodes in a grid) and temporal dynamics into a unified framework.
Example: Predicting nodal LMPs by considering neighboring bus prices and historical trends.
Practical application: Enhancing location-specific price forecasts for congestion management.
Challenges: High dimensionality; requires extensive data on network topology.

State-Space Model
Related terms: Kalman filter, hidden Markov model, dynamic linear model.
Explanation: Represents a system with an unobserved state vector evolving over time and observed measurements linked to that state.
Example: Modeling the hidden demand trend and observed load simultaneously.
Practical application: Provides a flexible framework for incorporating both observed and latent variables.
Challenges: Model identification can be complex; requires assumptions about noise distributions.

Stochastic Gradient Descent (SGD)
Related terms: Mini-batch, learning rate, optimization.
Explanation: Iterative optimization algorithm that updates model parameters using gradients computed on small random subsets of data.
Example: Training a large-scale price prediction neural network with SGD and momentum.
Practical application: Enables training on massive datasets that cannot fit into memory.
Challenges: Sensitive to learning-rate schedule; may converge to noisy minima.

Support Vector Machine (SVM)
Related terms: Kernel trick, margin maximization, hyperplane.
Explanation: Supervised learning algorithm that finds the hyperplane that maximally separates classes (classification) or fits within a tolerance (regression).
Example: Using ε-SVR to predict price deviations from a baseline.
Practical application: Effective for small-to-medium datasets with non-linear relationships when combined with kernels.
Challenges: Scaling to large datasets is difficult; kernel choice heavily influences performance.

Temporal Fusion Transformers (TFT)
Related terms: Attention mechanism, multi-horizon forecasting, variable selection network.
Explanation: Advanced deep learning architecture that combines recurrent layers, attention, and gating mechanisms to handle both static and time-varying inputs for multi-step forecasts.
Example: Forecasting 24-hour ahead prices using recent market data, calendar features, and weather forecasts.
Practical application: State-of-the-art performance on complex electricity price series.
Challenges: Requires substantial computational resources; interpretability is less straightforward than classical models.

Time-Series Cross-validation
Related terms: Rolling origin, walk-forward, evaluation.
Explanation: Validation technique that respects temporal order by training on past data and testing on future data repeatedly.
Example: Using a 12-month training window to predict the following month, then shifting forward month by month.
Practical application: Provides realistic performance estimates for forecasting models.
Challenges: Limited number of folds for short series; may still leak information if seasonal patterns span folds.

Transfer Learning
Related terms: Pre-training, fine-tuning, domain adaptation.
Explanation: Leverages knowledge learned from a source task to improve performance on a related target

task, often by reusing model weights.
Example: Pre-training a neural network on large-scale European price data, then fine-tuning on a specific regional market.
Practical application: Reduces data requirements for new markets with limited historical records.
Challenges: Mismatch between source and target domains can lead to negative transfer.

Trend Component
Related terms: Long-term pattern, detrending, decomposition.
Explanation: Underlying direction (increasing or decreasing) in a time series, separate from seasonal and irregular components.
Example: A gradual rise in average electricity prices due to policy changes.
Practical application: Removing trend before fitting stationary models like ARIMA.
Challenges: Trend may be non-linear; detecting it reliably requires sufficient data length.

t-Distributed Stochastic Neighbor Embedding (t-SNE)
Related terms: Visualization, high-dimensional data, perplexity.
Explanation: Non-linear dimensionality reduction technique that maps high-dimensional data to a low-dimensional space while preserving local structure.
Example: Visualizing clusters of load profiles to identify typical consumption patterns.
Practical application: Exploratory data analysis for feature engineering.
Challenges: Computationally intensive; results can vary with hyperparameters, making reproducibility an issue.

Underfitting
Related terms: High bias, model simplicity, insufficient capacity.
Explanation: When a model is too simple to capture underlying patterns, leading to poor performance on both training and validation data.
Example: Using a linear regression to model highly non-linear price spikes.
Practical application: Diagnosing model inadequacy and prompting the use of more expressive algorithms.
Challenges: Balancing model complexity to avoid swinging into overfitting.

Vanilla RNN
Related terms: Simple recurrent