

# Machine Learning Algorithms for Fault Diagnosis

## AdaBoost

Concept: Adaptive Boosting algorithm that combines multiple weak learners to form a strong classifier.

Related terms: boosting, weak learner, ensemble

Explanation: AdaBoost iteratively trains a series of simple classifiers (often decision stumps) on weighted versions of the training data. After each round, mis-classified samples receive higher weights, forcing the next learner to focus on harder cases. The final prediction is a weighted vote of all learners, where each learner's weight reflects its accuracy.

Example: In a printed-circuit-board (PCB) fault detection task, an AdaBoost model may use voltage-level features from test points; each weak learner distinguishes between "open circuit" and "short circuit" based on a single feature, and the ensemble yields high overall detection accuracy.

Practical application: Used for classifying component failures in solder-joint inspection systems, where rapid training on limited labeled data is needed.

Challenges: Sensitive to noisy data and outliers; requires careful tuning of the number of boosting rounds to avoid over-fitting.

## Artificial Neural Network (ANN)

Concept: Computational model inspired by biological neurons, consisting of layers of interconnected nodes that learn representations from data.

Related terms: deep learning, perceptron, back-propagation

Explanation: An ANN maps inputs to outputs through weighted connections and non-linear activation functions. During training, the back-propagation algorithm adjusts weights to minimize a loss function, enabling the network to capture complex non-linear relationships. In fault diagnosis, the network can learn patterns linking sensor readings to specific failure modes.

Example: A feed-forward ANN trained on temperature, current, and vibration data from a power supply can predict imminent capacitor degradation with a probability score for each degradation class.

Practical application: Deployed on embedded diagnostic modules that continuously monitor critical components and trigger maintenance alerts when the network indicates a high risk of failure.

Challenges: Requires a large labeled dataset for reliable training; prone to over-fitting if network depth is excessive relative to data size; interpretability is limited compared with rule-based methods.

## Autoencoder

Concept: Unsupervised neural architecture that learns to reconstruct its input, thereby extracting a compressed latent representation.

Related terms: dimensionality reduction, reconstruction error, bottleneck

Explanation: An autoencoder consists of an encoder that maps the input to a low-dimensional code and a decoder that reconstructs the input from this code. Training minimizes the difference between original and reconstructed data. For fault diagnosis, the reconstruction error can serve as an anomaly indicator: normal operating data are reconstructed accurately, while faulty data produce larger errors.

Example: In a motor-control board, an autoencoder trained on normal vibration spectra will flag a new vibration pattern with a high reconstruction error, suggesting bearing wear.

Practical application: Integrated into real-time monitoring systems to provide unsupervised detection of novel faults without needing labeled failure data.

Challenges: Selecting appropriate architecture size; ensuring the latent space captures fault-relevant features rather than irrelevant variations; sensitivity to scaling and preprocessing.

### Bagging

Concept: Bootstrap Aggregating technique that builds multiple models on random subsets of data and aggregates their predictions.

Related terms: bootstrap, ensemble, variance reduction

Explanation: Bagging generates several training sets by sampling with replacement from the original dataset. Each subset trains an independent base learner (often decision trees). The final prediction is obtained by majority voting (classification) or averaging (regression). This reduces model variance and improves robustness to noise.

Example: A bagged ensemble of ten decision trees classifies resistor failures based on voltage drop and temperature readings, achieving higher stability than any single tree.

Practical application: Used in portable diagnostic tools where limited computational resources benefit from the simplicity of decision trees while still achieving reliable performance.

Challenges: Increased computational cost during inference due to multiple models; diminishing returns if base learners are already low-variance.

### Bayesian Network

Concept: Probabilistic graphical model representing variables and their conditional dependencies via a directed acyclic graph.

Related terms: probabilistic inference, DAG, belief propagation

Explanation: Nodes denote random variables (e.g., sensor measurements), and edges encode conditional relationships. The joint probability distribution factorizes according to the graph structure, allowing efficient inference of posterior probabilities given evidence. In fault diagnosis, Bayesian networks can model causal links between component health states and observable symptoms.

Example: A network for a power-converter includes nodes for "input voltage anomaly," "output ripple," and "capacitor health." Observing high ripple updates the belief that the capacitor is degrading.

Practical application: Embedded in diagnostic expert systems that provide technicians with probability-based fault hypotheses, supporting decision-making under uncertainty.

Challenges: Requires expert knowledge to define structure and conditional probability tables; learning parameters from limited fault data can be difficult; inference can become computationally intensive for large networks.

### Classification

Concept: Supervised learning task that assigns discrete labels to input instances based on learned patterns.

Related terms: supervised learning, label, multiclass

Explanation: A classifier learns a decision boundary that separates data points belonging to different classes. In electronics repair, classes often correspond to specific fault types such as "open circuit," "short circuit,"

“component drift,” or “no fault.” Performance is measured using metrics like accuracy, precision, recall, and F1-score.

Example: A support-vector-machine classifier distinguishes between defective and functional diodes using forward voltage and reverse leakage current as features.

Practical application: Automated test equipment (ATE) uses classification models to sort devices into “pass” or “fail” bins, reducing manual inspection time.

Challenges: Imbalanced class distributions common in failure data (few fault examples vs. many normal examples) can bias the model; feature selection is critical to avoid irrelevant inputs degrading performance.

### Convolutional Neural Network (CNN)

Concept: Deep learning architecture specialized for processing grid-like data, especially images or spectrograms, using convolutional filters.

Related terms: feature maps, pooling, kernels

Explanation: CNNs apply learned filters that slide across the input, capturing local patterns such as edges or frequency components. Subsequent pooling layers reduce spatial dimensions, providing translation invariance. For fault diagnosis, visual representations of signals (e.g., time-frequency plots) are fed into CNNs, which automatically learn discriminative features.

Example: A CNN trained on spectrograms of acoustic emissions from a PCB identifies solder-joint cracks, outperforming handcrafted feature classifiers.

Practical application: Integrated into smartphone-based diagnostic apps that capture audio of device operation and classify faults on-device.

Challenges: Requires substantial labeled image data; computationally demanding for real-time embedded deployment; interpretability of learned filters can be opaque.

### Cross-validation

Concept: Model evaluation technique that partitions data into multiple training and validation folds to assess generalization performance.

Related terms: k-fold, hold-out, overfitting

Explanation: In k-fold cross-validation, the dataset is divided into k equal parts; each part serves once as a validation set while the remaining k-1 parts train the model. Results across folds are averaged, providing a more reliable estimate of model performance than a single train-test split.

Example: Using 5-fold cross-validation, a random-forest fault detector for voltage regulators achieves an average accuracy of 94 % with a standard deviation of 1.2 %.

Practical application: Helps select hyperparameters (e.g., tree depth, learning rate) for algorithms used in automated test benches, ensuring robust performance across production batches.

Challenges: Increases training time proportionally to the number of folds; care must be taken to preserve temporal ordering for time-dependent data to avoid leakage.

### Decision Tree

Concept: Tree-structured model that recursively splits data based on feature thresholds to predict class labels.

Related terms: entropy, Gini impurity, leaf node

Explanation: At each node, the algorithm selects the feature and split point that maximally reduces impurity

(e.g., using information gain). The process continues until stopping criteria are met (e.g., depth limit or minimum samples per leaf). Decision trees are interpretable, as the path from root to leaf provides a rule set.

Example: A tree for diagnosing LED driver failures uses thresholds on output current and temperature to decide whether the fault is "over-current," "thermal shutdown," or "normal."

Practical application: Embedded in low-power microcontrollers for on-board diagnostics, where the simple rule set can be executed without floating-point hardware.

Challenges: Prone to over-fitting, especially with noisy data; small changes in data can lead to drastically different tree structures (high variance).

### Deep Learning

Concept: Subfield of machine learning that employs neural networks with many layers to learn hierarchical representations from raw data.

Related terms: layer, representation learning, end-to-end

Explanation: Deep models automatically discover features at multiple levels of abstraction, reducing reliance on manual feature engineering. In fault diagnosis, deep networks can ingest raw sensor streams (voltage, current, acoustic) and output fault probabilities directly.

Example: A deep residual network (ResNet) processes raw waveform samples from a power-supply test point and classifies faults into "normal," "capacitor leak," and "inductor short."

Practical application: Deployed on edge AI accelerators attached to test equipment, enabling real-time classification without sending data to a cloud server.

Challenges: Needs large annotated datasets; training can be resource-intensive; risk of "black-box" behavior that hinders regulatory acceptance.

### Ensemble Learning

Concept: Strategy that combines predictions from multiple models to improve overall performance compared with any single model.

Related terms: stacking, voting, diversity

Explanation: Ensembles exploit the principle that different models make different errors; by aggregating them (e.g., majority voting, weighted averaging), the combined system reduces variance, bias, or both.

Techniques include bagging, boosting, and stacking, each promoting diversity in distinct ways.

Example: An ensemble consisting of a random forest, a gradient-boosted tree, and a support-vector-machine achieves 97 % fault detection accuracy on a dataset of faulty and functional MOSFETs, surpassing each constituent model.

Practical application: Used in high-throughput manufacturing lines where a single model's misclassification could cause costly rework; the ensemble provides a safety net.

Challenges: Increased computational load during inference; managing model interoperability and ensuring consistent input preprocessing across members.

### Extreme Gradient Boosting (XGBoost)

Concept: Optimized implementation of gradient boosting that builds additive tree models using second-order gradient information.

Related terms: boosting, regularization, tree learner

**Explanation:** XGBoost sequentially adds decision trees that predict the residuals of the current ensemble. It employs a regularized objective function that penalizes model complexity, improving generalization. Parallel processing and tree pruning make it fast and scalable.

**Example:** An XGBoost classifier trained on voltage-ripple, temperature, and load-current features identifies intermittent power-supply failures with an AUC of 0.98.

**Practical application:** Integrated into diagnostic software for field-replaceable units (FRUs) where rapid model updates are needed as new fault data become available.

**Challenges:** Sensitive to hyperparameter settings (e.g., learning rate, max depth); may over-fit if trees become too deep relative to the amount of fault data.

### Feature Extraction

**Concept:** Process of transforming raw sensor data into informative variables that capture underlying patterns relevant to fault detection.

**Related terms:** signal processing, domain knowledge, dimensionality reduction

**Explanation:** Techniques include statistical descriptors (mean, variance), frequency-domain measures (FFT peaks, spectral entropy), time-frequency methods (wavelet coefficients), and learned embeddings from autoencoders. Effective features enhance classifier performance and reduce training time.

**Example:** For a switching regulator, extracting the dominant frequency component from the output ripple and the variance of the control-loop current yields features that separate "normal" and "oscillatory" fault states.

**Practical application:** Feature extraction pipelines are implemented in firmware of test instruments, allowing on-device preprocessing before sending data to a cloud-based model.

**Challenges:** Selecting features that are robust to environmental variations (temperature, supply noise); risk of discarding subtle fault signatures when overly aggressive dimensionality reduction is applied.

### Fuzzy Logic

**Concept:** Reasoning framework that handles uncertainty by allowing partial membership of variables in linguistic sets.

**Related terms:** membership function, inference engine, rule base

**Explanation:** In fuzzy systems, inputs are mapped to degrees of truth (0–1) using membership functions (e.g., "high temperature"). A set of IF-THEN rules combines these degrees to produce an output that can be defuzzified into a crisp diagnostic decision. This approach mimics expert reasoning and tolerates imprecise measurements.

**Example:** A fuzzy controller evaluates "voltage deviation" and "temperature rise" to infer a fault severity level ranging from "nominal" to "critical."

**Practical application:** Deployed in legacy equipment where adding a full machine-learning pipeline is infeasible, but a rule-based expert system can still provide automated diagnostics.

**Challenges:** Designing appropriate membership functions and rule sets requires domain expertise; scaling to many inputs can lead to rule explosion and reduced interpretability.

### Gaussian Naive Bayes

**Concept:** Probabilistic classifier that assumes feature independence and models each feature's distribution as Gaussian.

Related terms: Bayes theorem, conditional independence, likelihood

Explanation: For each class, the model estimates mean and variance of each continuous feature, then computes the posterior probability of the class given a new observation using Bayes' rule. Despite its strong independence assumption, it often performs well on high-dimensional data with limited training samples.

Example: Using temperature, current, and vibration amplitude as Gaussian features, the classifier predicts whether a power-module is "healthy," "over-heated," or "mechanically damaged."

Practical application: Suitable for on-chip diagnostic modules where memory and compute constraints preclude more complex models.

Challenges: Violation of independence assumption can degrade accuracy; Gaussian assumption may not hold for skewed sensor distributions, requiring preprocessing or alternative distributions.

### Isolation Forest

Concept: Unsupervised anomaly detection algorithm that isolates observations by randomly partitioning data using binary trees.

Related terms: anomaly score, tree depth, contamination

Explanation: Anomalies are those that require fewer random splits to isolate, resulting in shorter average path lengths across the forest. The algorithm builds many isolation trees on subsamples of the data, then assigns an anomaly score based on the average depth.

Example: Applying Isolation Forest to a dataset of current-waveform snapshots from a fleet of inverters flags a few rare waveforms with high anomaly scores, prompting further inspection for possible sensor drift.

Practical application: Integrated into continuous-monitoring dashboards for remote IoT devices, providing early warning of abnormal behavior without labeled fault data.

Challenges: Choice of contamination parameter (expected proportion of anomalies) influences detection threshold; may produce false positives when normal operating conditions vary widely.

### Jaccard Index

Concept: Similarity metric that quantifies the overlap between two sets, defined as the size of the intersection divided by the size of the union.

Related terms: set similarity, IoU, binary classification

Explanation: In fault diagnosis, the Jaccard Index can evaluate the agreement between predicted fault regions (e.g., segmented thermal images) and ground-truth annotations. A higher index indicates better correspondence.

Example: For a thermal-image segmentation model that identifies hot spots on a power board, a Jaccard score of 0.82 demonstrates strong overlap with manually labeled defect areas.

Practical application: Used as an evaluation metric for image-based fault localization models during model validation.

Challenges: Sensitive to class imbalance; when the fault region is very small, a small misalignment can cause a large drop in the index.

### K-Nearest Neighbors (KNN)

Concept: Instance-based learning algorithm that classifies a query point based on the majority label among its k closest training samples.

Related terms: distance metric, lazy learning, curse of dimensionality

Explanation: The algorithm stores the entire training set and computes distances (e.g., Euclidean) at prediction time. The value of  $k$  controls the trade-off between bias and variance. Feature scaling is essential because distance calculations are sensitive to magnitude.

Example: A KNN model with  $k = 3$  classifies voltage-ripple patterns into "normal," "capacitor aging," or "inductor saturation" by comparing to a library of labeled waveforms.

Practical application: Useful for quick prototyping on diagnostic laptops where the dataset fits in memory and real-time predictions are acceptable.

Challenges: Computationally expensive for large datasets; performance degrades in high-dimensional spaces unless dimensionality reduction is applied.

### Logistic Regression

Concept: Linear model that predicts the probability of a binary outcome using the logistic (sigmoid) function.

Related terms: odds ratio, maximum likelihood, regularization

Explanation: The model computes a weighted sum of input features, passes it through the sigmoid to obtain a probability between 0 and 1, and applies a threshold to decide class membership. Regularization (L1 or L2) prevents over-fitting by penalizing large coefficients.

Example: Predicting whether a voltage regulator will fail within the next 100 hours based on temperature trend and load-current variance, yielding a failure probability of 0.73.

Practical application: Frequently embedded in firmware for health-monitoring modules that need transparent decision logic and low computational overhead.

Challenges: Assumes linear relationship between features and log-odds; may underperform when fault patterns are highly non-linear.

### Multilayer Perceptron (MLP)

Concept: Feed-forward neural network composed of one or more hidden layers, each fully connected to the previous layer.

Related terms: hidden units, activation function, back-propagation

Explanation: MLPs learn non-linear mappings by applying weighted sums and non-linear activations (e.g., ReLU, tanh) at each layer. The network is trained via gradient descent to minimize a loss function. In fault diagnosis, MLPs can model complex interactions among sensor readings.

Example: An MLP with two hidden layers (64 and 32 neurons) classifies PCB board states into "good," "solder-joint crack," or "component misplacement" using voltage, current, and acoustic features.

Practical application: Implemented on programmable logic controllers (PLCs) that support neural inference libraries, enabling on-site fault classification without external computation.

Challenges: Requires careful hyperparameter tuning (learning rate, number of neurons); susceptible to over-fitting on small fault datasets; training may be slower than tree-based methods.

### Naive Bayes

Concept: Family of simple probabilistic classifiers that assume conditional independence among features given the class label.

Related terms: posterior probability, prior, likelihood

Explanation: The model computes the product of individual feature likelihoods for each class and multiplies

by the class prior, then selects the class with the highest posterior probability. Despite its naive assumptions, it often performs well in high-dimensional settings and is extremely fast to train and predict.

Example: Using binary features indicating whether certain voltage thresholds are exceeded, a Naive Bayes model predicts "short circuit" vs. "open circuit" in a connector test.

Practical application: Deployed in low-power diagnostic chips where rapid inference is essential and memory for storing complex models is limited.

Challenges: Independence assumption rarely holds in real sensor data, potentially reducing accuracy; smoothing techniques (e.g., Laplace) are needed to handle zero-frequency features.

### One-Class SVM

Concept: Unsupervised learning algorithm that learns a decision boundary enclosing the majority of data points, treating all others as outliers.

Related terms: kernel trick, support vectors, novelty detection

Explanation: The algorithm maps data into a high-dimensional feature space using a kernel (commonly RBF) and finds the hyperplane that maximally separates the data from the origin. Points lying outside this boundary receive an anomaly label. Useful when only normal operation data are available.

Example: Training a One-Class SVM on normal current-waveform samples from a power-module allows detection of abnormal waveforms caused by component fatigue.

Practical application: Integrated into firmware of legacy devices that cannot store extensive fault libraries but can learn the normal operating envelope during a calibration phase.

Challenges: Sensitive to kernel parameters ( $\gamma$ ) and the  $\nu$  parameter (controlling trade-off between outlier fraction and margin); may misclassify rare but legitimate variations as faults.

### Principal Component Analysis (PCA)

Concept: Linear dimensionality-reduction technique that projects data onto orthogonal axes (principal components) capturing maximal variance.

Related terms: eigenvectors, covariance matrix, reconstruction error

Explanation: By computing the eigenvectors of the data covariance matrix, PCA identifies directions along which the data vary most. Projecting onto the top-k components reduces dimensionality while preserving most of the information. In fault diagnosis, PCA can highlight anomalous deviations from the normal subspace.

Example: After applying PCA to a set of temperature-current feature vectors, the first two components explain 95 % of variance; a new observation with a large projection on the third component signals a potential fault.

Practical application: Used as a preprocessing step for clustering algorithms that group similar fault signatures, improving computational efficiency.

Challenges: Captures only linear correlations; nonlinear relationships may be missed; components can be difficult to interpret in terms of physical sensor meanings.

### Quadratic Discriminant Analysis (QDA)

Concept: Classification method that models each class with its own covariance matrix, allowing quadratic decision boundaries.

Related terms: Gaussian assumption, discriminant function, Bayes classifier

**Explanation:** QDA assumes that data from each class follow a multivariate Gaussian distribution with distinct means and covariances. The resulting discriminant function includes quadratic terms, enabling more flexible separation than linear discriminant analysis (LDA).

**Example:** Classifying power-supply failures into "over-voltage," "under-voltage," and "oscillation" using voltage, current, and temperature features; QDA captures the curved boundaries observed in the data.

**Practical application:** Applied in diagnostic suites where fault classes exhibit different variabilities, such as when some faults cause high variance in temperature while others affect voltage stability.

**Challenges:** Requires estimating a full covariance matrix for each class, which can be unstable with limited training samples; prone to over-fitting in high-dimensional spaces.

### Random Forest

**Concept:** Ensemble of decision trees built on bootstrapped samples and random subsets of features, aggregating predictions by majority vote.

**Related terms:** bagging, feature randomness, out-of-bag error

**Explanation:** Each tree is trained on a different bootstrap sample, and at each split only a random subset of features is considered, promoting diversity. The out-of-bag (OOB) error provides an internal estimate of generalization performance without a separate validation set.

**Example:** A Random Forest with 200 trees classifies capacitor health into "good," "degraded," and "failed" using capacitance, ESR, and temperature features, achieving 96% overall accuracy.

**Practical application:** Frequently used in manufacturing test stations where robustness to sensor noise and ease of interpretability (feature importance) are valuable.

**Challenges:** Large ensembles increase memory usage; individual trees are still prone to bias if the underlying features are weakly informative.

### Support Vector Machine (SVM)

**Concept:** Supervised learning algorithm that finds the hyperplane maximizing the margin between classes in a (possibly transformed) feature space.

**Related terms:** margin, kernel trick, support vectors

**Explanation:** By solving a convex optimization problem, SVM determines the decision boundary that separates classes with the largest possible distance. Kernels (linear, polynomial, RBF) enable non-linear separation by implicitly mapping inputs to higher-dimensional spaces.

**Example:** An RBF-kernel SVM distinguishes between normal and intermittent short-circuit faults in a switching regulator based on high-frequency noise spectra.

**Practical application:** Adopted in diagnostic software for high-precision equipment where the number of fault classes is modest but the feature space is complex.

**Challenges:** Training time scales quadratically with the number of samples, making it less suitable for very large datasets; selecting an appropriate kernel and regularization parameter (C) requires careful cross-validation.

### Time-Series Forecasting

**Concept:** Predictive modeling of future values of a sequence of observations based on past behavior.

**Related terms:** autoregressive, moving average, seasonality

**Explanation:** Methods range from classical statistical models (ARIMA, exponential smoothing) to recurrent

neural networks (LSTM, GRU). In fault diagnosis, forecasting can predict when a parameter (e.g., temperature) will cross a failure threshold, enabling proactive maintenance.

Example: An LSTM network trained on hourly temperature readings from a power converter predicts a temperature surge 8 hours ahead, allowing a scheduled shutdown before overheating occurs.

Practical application: Integrated into predictive-maintenance platforms that schedule service visits for field-deployed electronics based on forecasted degradation trends.

Challenges: Requires sufficient historical data; non-stationary behavior due to operational mode changes can degrade forecast accuracy; model complexity must be balanced against real-time constraints.

### Unsupervised Learning

Concept: Machine-learning paradigm that discovers patterns in data without using labeled outcomes.

Related terms: clustering, dimensionality reduction, anomaly detection

Explanation: Algorithms such as k-means, hierarchical clustering, DBSCAN, and autoencoders identify structure, group similar instances, or detect outliers. In fault diagnosis, unsupervised methods are valuable when failure examples are scarce or unknown.

Example: Applying k-means clustering to feature vectors derived from voltage and current waveforms reveals three clusters corresponding to "normal operation," "soft fault," and "hard fault," even though no prior labels were supplied.

Practical application: Used during early product development to explore potential fault modes before a comprehensive labeled dataset is built.

Challenges: Determining the correct number of clusters or appropriate distance metrics can be ambiguous; results may be sensitive to feature scaling and noise.

### Variance Inflation Factor (VIF)

Concept: Metric that quantifies multicollinearity among predictor variables in a regression model.

Related terms: multicollinearity, regression diagnostics, tolerance

Explanation: VIF for a given feature is calculated as  $1 / (1 - R^2)$  where  $R^2$  is the coefficient of determination when that feature is regressed on all others. High VIF values (commonly  $> 5$  or  $> 10$ ) indicate that the feature is highly correlated with others, potentially inflating variance of coefficient estimates.

Example: In a logistic-regression fault model, temperature and current have  $VIF = 12$ , suggesting that one should be removed or combined to improve model stability.

Practical application: Employed during feature selection for fault-diagnosis models to ensure that each input contributes independent information, enhancing interpretability and reducing over-fitting.

Challenges: Does not address non-linear relationships; removal of correlated features may discard useful information if not handled carefully.

### Wavelet Transform

Concept: Time-frequency analysis technique that decomposes a signal into localized wavelet coefficients at multiple scales.

Related terms: continuous wavelet, discrete wavelet, mother wavelet

Explanation: By convolving the signal with scaled and shifted versions of a mother wavelet, the transform captures both frequency content and temporal location, making it ideal for transient-rich fault signatures (e.g., spikes, bursts). The resulting coefficients can be used as features for classification.

Example: Using the Daubechies-4 wavelet, the decomposition of a voltage waveform reveals high-energy coefficients at the 3rd level, indicating a sudden load-step fault.

Practical application: Implemented in firmware of high-speed digital oscilloscopes that automatically extract wavelet features for on-device fault classification.

Challenges: Choice of mother wavelet and decomposition level affects sensitivity; computational load can be significant for high-resolution signals.

### X-GBoost

Concept: Advanced gradient-boosting library that provides regularization, parallel processing, and handling of missing values.

Related terms: boosting, tree ensemble, regularization

Explanation: X-GBoost builds trees sequentially, each trying to correct the residuals of the combined previous trees. It incorporates L1/L2 regularization on leaf weights, shrinkage (learning rate), and column subsampling to reduce over-fitting. The system can automatically learn optimal split points, making it highly efficient.

Example: An X-GBoost model trained on 10 000 samples of voltage, current, and temperature measurements achieves 98 % precision in detecting intermittent short-circuit events.

Practical application: Deployed in cloud-based diagnostic services that process large volumes of telemetry from field devices, providing rapid model updates as new fault data arrive.

Challenges: Hyperparameter tuning (max depth, learning rate, gamma) can be complex; model size may become large, requiring compression techniques for edge deployment.

### Zero-Shot Learning

Concept: Learning paradigm where a model can recognize classes it has never seen during training by leveraging auxiliary semantic information.

Related terms: semantic embedding, attribute vectors, transfer learning

Explanation: The model learns a mapping from visual or sensor features to a shared semantic space (e.g., attribute descriptions). When a new fault type appears, its semantic description enables the model to infer its representation and classify instances despite lacking explicit training examples.

Example: Using textual descriptions of "capacitor leakage" and "inductor saturation," a zero-shot model can identify these faults in new devices based on sensor patterns, even though the training set contained only "open circuit" and "short circuit" examples.

Practical application: Valuable for rapidly evolving product lines where new failure modes emerge faster than labeled datasets can be assembled, allowing diagnostic systems to adapt without retraining from scratch.

Challenges: Requires high-quality semantic descriptors; performance typically lags behind fully supervised models; risk of semantic mismatch leading to misclassification.

### Variance Threshold

Concept: Simple feature-selection method that removes features whose variance does not meet a specified threshold.

Related terms: feature selection, low variance filter, dimensionality reduction

Explanation: Features with near-constant values across samples contribute little information for

discrimination and can be discarded. Setting a variance threshold (e.g., 0.01) eliminates such redundant features, simplifying models and reducing computational load.

Example: In a dataset of sensor readings, a temperature sensor that always reports 25 °C (variance  $\approx 0$ ) is removed before training a fault classifier.

Practical application: Used as a preprocessing step in automated test equipment pipelines to streamline the feature set before applying more sophisticated algorithms.

Challenges: May inadvertently discard features that have low variance overall but become highly informative under certain operating conditions; threshold selection can be arbitrary.

### Weighted Voting

Concept: Ensemble aggregation technique where each model's vote is multiplied by a weight reflecting its confidence or performance.

Related terms: ensemble, confidence score, model weighting

Explanation: Instead of simple majority voting, weighted voting assigns higher influence to models that historically perform better on certain classes or data regions. Weights can be static (derived from validation accuracy) or dynamic (based on per-sample confidence).

Example: In a three-model ensemble (Random Forest, SVM, XGBoost) for diagnosing MOSFET failures, the XGBoost model receives weight = 0.5, Random Forest = 0.3, and SVM = 0.2, reflecting its superior validation F1-score.

Practical application: Improves reliability of diagnostic decisions in safety-critical systems where a single model's error could have severe consequences.

Challenges: Determining optimal weights adds an extra optimization layer; if weights are poorly estimated, the ensemble may amplify the weaknesses of a dominant model.

### Y-Axis Normalization

Concept: Preprocessing step that scales the values of a signal's Y-axis (amplitude) to a common range, often [0, 1] or [-1, 1].

Related terms: min-max scaling, standardization, amplitude normalization

Explanation: Normalization removes amplitude differences caused by varying sensor gains or measurement conditions, ensuring that subsequent feature extraction focuses on shape rather than scale. This is especially important when comparing waveforms from different devices or test setups.

Example: Normalizing the voltage waveforms from three different power supplies before feeding them to a CNN prevents the network from learning trivial amplitude-based distinctions.

Practical application: Implemented in data-acquisition firmware that automatically rescales raw ADC values prior to storage or transmission.

Challenges: If absolute amplitude carries diagnostic significance (e.g., an over-voltage fault), excessive normalization may obscure the fault signature; careful documentation of scaling factors is required for traceability.

### Z-Score Normalization

Concept: Statistical scaling method that transforms a feature to have zero mean and unit variance.

Related terms: standardization, mean centering, variance scaling

Explanation: For each feature, the Z-score is computed as  $(\text{value} - \text{mean}) / \text{standard deviation}$ . This ensures

that features with different units or ranges contribute equally to distance-based algorithms (e.g., KNN, SVM).

Example: Applying Z-score normalization to temperature, current, and vibration features before training a logistic-regression fault detector improves convergence and classification performance.

Practical application: Commonly used in preprocessing pipelines for machine-learning models deployed on diagnostic tablets that ingest heterogeneous sensor data.

Challenges: Requires accurate estimation of mean and standard deviation from representative training data; outliers can distort the scaling, necessitating robust statistics or outlier removal beforehand.