

# Machine Learning Regression Techniques for Modeling Complex Industrial Systems: A Comprehensive Summary

(Authors Details)

**Sanjay Nakharu Prasad Kumar**  
George Washington University, USA

## Abstract

Machine learning regression techniques have become essential for modeling complex industrial systems in the industry 4.0 era. This summary reviews key regression methods—from linear models to deep learning—and examines their applications across manufacturing, energy, chemical processing, and predictive maintenance. We discuss how different techniques balance accuracy, interpretability, and computational requirements, and highlight real-world implementations that have achieved substantial efficiency gains. The review also addresses critical challenges including data quality, model interpretability, and generalization, while exploring future directions such as physics-informed models, federated learning, and explainable AI.

**Keywords:** Machine Learning, Regression, Industrial Systems, Predictive Maintenance, Manufacturing, Process Modeling, Ensemble Learning, Deep Learning, Industry 4.0

**DOI:** 10.21590/ijhit.06.01-3.07

## 1. Introduction

Industries are experiencing a fundamental transformation toward automation and data-driven decision-making, with machine learning emerging as a critical enabler for optimizing industrial processes. Traditional rule-based automation systems often struggled with the variability and complexity of modern manufacturing environments. In contrast, ML-driven systems learn from historical and real-time data to recognize patterns, predict outcomes, and continuously improve without explicit reprogramming.

The rise of Industry 4.0, characterized by the Industrial Internet of Things (IIoT), cloud computing, and ubiquitous sensing, has accelerated this transformation by generating massive volumes of data. Machine learning regression techniques—methods for predicting continuous outcomes based on input features—have become critical for extracting actionable insights from these complex datasets to drive improvements in efficiency, cost, and reliability.

Regression analysis provides a quantitative means to link process inputs to performance outputs, helping engineers answer questions like: How do changes in operating conditions affect product quality or yield? While historically simple regression models were used in process engineering, complex industrial systems often exhibit nonlinear, high-dimensional relationships that require more sophisticated approaches. Today's arsenal ranges from regularized linear models to

nonlinear kernel-based methods, ensemble algorithms, and deep learning models that automatically learn rich representations from large-scale data.

## 2. Overview of Regression Methods

In machine learning, regression refers to predictive modeling techniques for estimating a continuous output variable given a set of input variables. In industrial systems, regression tasks are ubiquitous examples include predicting product quality metrics, sensor readings, remaining useful life of equipment, or energy consumption levels based on various process parameters.

A wide range of regression methods exist, differing in their model assumptions and capacity to fit complex patterns:

- **Linear Regression:** The simplest form, assuming a linear relationship between inputs and outputs. Offers interpretability and low complexity but cannot capture nonlinear dependencies.
- **Regularized Regression:** Extensions like ridge (L2 penalty), lasso (L1 penalty), and elastic net help prevent overfitting in high-dimensional settings while maintaining interpretability.
- **Nonlinear Methods:** Support vector regression (SVR) and Gaussian process regression (GPR) use kernel functions or Bayesian inference to model complex functions with flexibility.
- **Ensemble Methods:** Techniques like random forests and gradient boosting (XGBoost, LightGBM) combine multiple decision trees to achieve higher predictive accuracy by aggregating predictions.
- **Deep Learning:** Multi-layer neural networks (MLPs, CNNs, RNNs, LSTMs) automatically learn intricate feature representations from large datasets, achieving state-of-the-art results in many domains.

### 2.1 Comparative Overview

Category	Examples	Key Strengths	Main Limitations
<b>Linear and regularized</b>	Linear regression, Ridge, Lasso, Elastic Net	Interpretable coefficients, fast training, good baseline, low data needs	Cannot capture nonlinearity, may underfit complex processes
<b>Nonlinear (Kernel-Based)</b>	SVR, GPR, k-NN	Models complex nonlinear functions, GPR provides uncertainty estimates	Higher computational cost, less interpretable, scaling challenges
<b>Ensemble (Tree-Based)</b>	Random Forests, XGBoost, LightGBM	High accuracy on diverse data, robust to noise, captures interactions	Black-box nature, requires tuning, limited extrapolation
<b>Deep Learning</b>	MLP, CNN, RNN, LSTM	Learns complex high-dimensional patterns, state-of-the-art results	Requires large datasets, long training time, difficult to interpret

In practice, industrial ML engineers must choose an appropriate technique based on factors like data size and quality, complexity of relationships, real-time requirements, and the need for model transparency.

### 3. Linear and Regularized Regression

Linear regression assumes a relationship of the form  $y = \beta_0 + \beta_1x_1 + \dots + \beta_px_p + \varepsilon$ , where coefficients are learned from data. Despite its simplicity, linear regression has been a workhorse in industrial contexts due to its interpretability—engineers can readily understand each input factor's contribution—and low data requirements. Multiple linear regression has historically underpinned design of experiments in manufacturing, helping identify how process parameters affect outcomes.

However, straightforward linear models struggle when inputs have strong interactions or nonlinear effects, which is common in complex physical processes. Regularized regression techniques address these limitations while retaining interpretability:

**Ridge Regression (L2):** Adds a proportional penalty to the squared magnitude of coefficients, shrinking them toward zero and stabilizing estimates in the presence of correlated variables.

**Lasso Regression (L1):** Adds an absolute value penalty, forcing many coefficients exactly to zero and performing automatic feature selection by identifying the most influential predictors.

**Elastic Net:** Combines L1 and L2 penalties to balance ridge's stability and lasso's sparsity.

These regularized methods are vital in high-dimensional industrial data where hundreds or thousands of sensor variables may be recorded. For instance, in semiconductor manufacturing, applying lasso regression helped identify critical tool signals from thousands of candidates, significantly simplifying predictive models without sacrificing accuracy.

Linear models can be extended using polynomial or other basis function expansions to model mild nonlinearity, though regularization becomes even more important as dimensionality grows. These techniques offer a robust baseline for many industrial problems—they are computationally efficient and interpretable, though their linear assumption limits their ability to capture complex nonlinear process behavior.

### 4. Nonlinear Regression

Many industrial systems exhibit inherently nonlinear relationships between inputs and outputs. Two versatile and widely used nonlinear regression techniques are support vector regression and Gaussian process regression.

#### 4.1 Support Vector Regression (SVR)

SVR extends support vector machines to regression problems, using kernel functions to implicitly map data into higher-dimensional feature spaces where linear regression is performed. This

kernel trick enables SVR to model highly nonlinear functions while maximizing margin to avoid overfitting.

SVR is effective on smaller to medium-sized datasets and can handle complex relationships with appropriate kernel choice (Gaussian RBF, polynomial, etc.). In semiconductor fabrication, SVR models have predicted critical dimensions based on process variables, capturing nonlinear trends that linear models could not. The method inherently includes regularization via margin maximization, but models are less interpretable than linear approaches, and computational cost grows with the number of support vectors.

#### **4.2 Gaussian Process Regression (GPR)**

GPR takes a probabilistic non-parametric approach, treating prediction as a Bayesian framework problem. A Gaussian process defines prior over functions that is updated to a posterior when training data are observed. GPR yields both predictions and uncertainty estimate valuable features for high-stakes industrial applications where knowing prediction confidence is important.

GPR with suitable kernels can model extremely complex relationships and sometimes outperform neural networks on moderate-sized datasets. In semiconductor process modeling, GPR consistently surpassed linear regression and neural network baselines in predictive accuracy while tracking concept drift through uncertainty bands. Hybrid approaches combining CNNs with GPR have also shown success, with CNNs extracting features from complex sensor signals and GPR producing final quality predictions with uncertainty quantification.

The strengths of GPR are its flexibility and probabilistic outputs; limitations include poor scalability to very large datasets (computational cost grows cubically with samples) and the requirement to choose appropriate kernel functions.

Other nonlinear methods include k-Nearest Neighbors regression and multivariate adaptive regression splines (MARS). In practice, SVR and GPR are prominent standalone nonlinear techniques in industrial case studies, especially when dataset size is moderate and insight into model behavior is desired.

### **5. Ensemble Regression Methods**

Ensemble methods combine predictions of multiple base models to produce more robust and accurate overall models. The two main paradigms are bagging (bootstrap aggregating) and boosting

#### **Random Forests**

Random forests use bagging ensembles of decision trees, where each tree is trained on random subsets of data and features. This randomness de-correlates trees and averaging many predictions reduces variance and improves generalization. Random forests can model complex nonlinear relationships, handle high-dimensional data, and are robust to outliers and noise—all useful properties for industrial datasets.

An added benefit is intrinsic importance measures, which can offer interpretability insights. In semiconductor manufacturing, random forests have achieved approximately 97% classification accuracy in identifying defective versus normal wafers.

### **Gradient Boosting Machines**

Gradient boosting builds ensembles sequentially, where each new tree corrects errors of the current ensemble. Optimized implementations like XGBoost, LightGBM, and CatBoost have become go-to tools in machine learning competitions and industrial applications. Boosting algorithms tend to achieve higher accuracy than bagging for many problems, especially with careful tuning, since trees concentrate on difficult-to-predict cases.

In industrial contexts, tree-based ensemble models have achieved excellent performance. They readily accommodate mixed data types and missing values, and scale well to large datasets. For example, gradient-boosted trees have been used to predict defect density and pinpoint influential process factors. By applying SHAP (Shapley Additive Explanations) to boosting models, engineers can interpret which sensor signals contribute most to yield losses.

Ensembles have also excelled in predictive maintenance, with studies integrating Random Forest, XGBoost, and LightGBM to predict machine failure probabilities, leveraging the ensemble's ability to handle diverse sensor inputs. Compared to single models like SVR or neural networks, ensembles often deliver competitive or superior accuracy on tabular industrial data without requiring extensive parameterization.

The downside is that resulting models are "black boxes" with potentially hundreds of trees, making direct interpretation difficult (though feature importance or SHAP analysis helps). Ensembles may also struggle with extrapolation beyond training data ranges.

Overall, ensemble regression methods have become a staple of industrial machine learning due to strong performance and robustness. They strike a good balance: more flexible than linear models but easier to train and often less data-hungry than deep neural networks.

## **6. Deep Learning-Based Regression**

Deep learning has increasingly been applied to regression problems in industrial domains, especially those involving very complex relationships or high-dimensional sensor data. Deep neural networks with multiple layers can automatically extract and combine features through hidden layers, making them extremely powerful for modeling complex systems when sufficient training data are available.

### **6.1 Feed-Forward Neural Networks (MLPs)**

Multi-layer perceptrons are fully connected networks where each layer processes weighted sums of previous outputs through nonlinear activations. MLPs can approximate virtually any continuous function given enough neurons. They have been used for mapping process conditions to quality measures or sensor readings to machine health indices. A 2- or 3-layer neural network

can capture moderate nonlinearities and interactions, outperforming linear models in capturing nonlinear process effects.

## **6.2 Convolutional Neural Networks (CNNs)**

CNNs specialize in spatial data like images and have achieved remarkable success in visual inspection tasks. In industrial settings, CNN-based regression applies when inputs are images—for example, predicting quality metrics from product images or detecting defect severity. Case studies show that image-based deep learning can detect manufacturing defects in real time with higher accuracy than human inspectors. CNNs have also been used to analyze vibration spectrograms or thermal images for predictive models.

## **6.3 Recurrent Neural Networks (RNNs)**

RNNs, particularly LSTM and GRU architectures, are designed for sequence data, making them suitable for time-series problems like forecasting and predictive maintenance. Many industrial processes generate time-series sensor data, and RNN-based regression can teach temporal dependencies. LSTM networks have been employed to predict remaining useful life of engines or forecast short-term load demand in energy grids, capturing temporal patterns that static models miss.

## **6.4 Hybrid Architectures**

Deep learning allows creative combinations. CNN-LSTM hybrids have been built for predictive maintenance, feeding CNN-extracted features from sensor signals into LSTMs to model temporal behavior. Recent comparisons found that CNN-LSTM hybrids achieved the best performance (96% accuracy, 95% F1-score) in predicting machine failures, outperforming either architecture alone. Autoencoders are also used in anomaly detection, and deep networks have been fused with other methods like Gaussian processes.

However, challenges that remain deep learning models are often considered "black boxes" due to lack of interpretability, which can hinder trust and adoption. They typically require large, labeled datasets, collecting sufficiently diverse and annotated industrial data can be difficult and expensive. Overfitting risks exist if networks are too complex for available data. Training and deploying deep models can be resource-intensive, potentially requiring specialized hardware for real-time inference.

Despite these challenges, deep learning adoption is increasing markedly, with a general shift toward real-time, edge-deployed models in manufacturing contexts. Deep learning-based regression represents the state-of-the-art for many complex prediction tasks in industry, with ongoing research devoted to making these models more efficient and interpretable.

## **7. Comparative Evaluation**

Each category of regression technique offers distinct advantages, and performance varies widely depending on problem characteristics and data:

**Linear vs. Nonlinear Performance:** Nonlinear ML models generally outperform simple linear models on complex problems involving interactions or threshold effects. Studies consistently show Gaussian process regression and neural networks significantly outperforming linear regression on semiconductor process datasets due to their ability to capture nonlinear dependencies. However, when relationships are approximately linear or data are limited, linear models with regularization can perform comparably while being simpler and more robust.

**Regularized Linear vs. Tree Ensembles:** Ensembles can model nonlinearities that linear models cannot and often win contests in accuracy on diverse industrial datasets. However, linear models with lasso can perform feature selection, providing sparser, more interpretable models. Ensembles may be overfit if not tuned, but techniques like cross-validation and early stopping help mitigate this. Training linear models is extremely fast even on large data, whereas ensembles are moderately heavier but still far faster than deep networks.

**Support Vector vs. Ensemble vs. Neural Network:** Comparative studies in manufacturing provide useful insights. One virtual metrology study found that lasso had the highest error due to its restricted form, neural networks improved accuracy by fitting nonlinear patterns but required careful tuning, and Gaussian processes achieved the best accuracy with uncertainty estimates but were most computationally intensive. In many competitions and benchmarks on industrial data, gradient boosted trees and deep neural networks are often top performers.

**Interpretability and User Trust:** Simpler models are far easier for domain experts to trust and understand. In critical safety industries, a slight drop in accuracy might be acceptable for a transparent model. Techniques like SHAP values can be applied to tree ensembles and neural networks to explain predictions. However, a persistent trade-off exists between interpretability and performance—engineers often prefer simpler, more transparent models for routine use.

**Data Regimes:** The amount of training data strongly influences which model is appropriate. For small data (dozens to hundreds of samples), linear or SVR/GPR models are more reliable. For medium data (thousands of samples), tree ensembles and shallow neural networks work well. For very large data (millions of samples), deep learning excels. Real-world industrial datasets often have high dimensionality but not always high volume, favoring methods that handle high-dimensional data with regularization.

In summary, no single regression technique is universally best for all industrial problems. A sensible approach is often to benchmark multiple models on validation datasets and consider practical aspects beyond raw accuracy. Increasingly, hybrid solutions are used to capitalize on complementary strengths of different methods.

## **8. Industrial Applications**

### **8.1 Manufacturing**

ML regression improves production quality, yield, and efficiency across manufacturing operations:

**Quality Control and Inspection:** High-resolution cameras and sensors capture product data, and ML models predict quality metrics or detect defects. CNNs analyzing images of manufactured parts have achieved real-time defect detection with higher accuracy and consistency than manual inspection. Regression models output continuous quality scores or measurements that help decide if parts meet specifications.

**Process Modeling and Optimization:** Regression models serve as surrogate models or virtual sensors for complex manufacturing processes. In semiconductor wafer fabrication, regression models predict critical outputs like film thickness from in situ sensor readings, enabling near-real-time estimation without direct measurement. Multiple methods (linear, lasso, neural network, GPR) have been evaluated for virtual metrology, significantly outperforming static approaches.

**Yield and Throughput Prediction:** ML regression forecasts final yield of production batches based on early-process data or predicts production throughput under various settings. In one deployment, an ensemble of ML models in a digital twin framework for an oil refinery's catalytic cracking unit achieved a 0.5% increase in light oil yield by recommending better operating conditions—translating to substantial economic gain at scale.

**Smart Manufacturing/Industry 4.0:** Regression models deployed alongside IoT devices provide instant predictions. Edge-deployed deep learning models continuously monitor machines and output tool wear levels or remaining life estimates, allowing control systems to adjust parameters or schedule maintenance at optimal times.

## 8.2 Energy Systems

**Load Forecasting:** Utilities need to predict electricity demand on various timescales to inform generation scheduling. Models including support vector regression, random forests, and LSTM neural networks have shown improved accuracy by incorporating weather data, calendar effects, and smart grid sensor measurements. Accurate forecasts translate into more efficient generation dispatch and reduced need for expensive peaking power.

**Smart Grid Management:** Power grids instrumented with sensors use ML regression models to predict potential failures or inefficiencies. Models can predict transformer temperature or health indices hours or days in advance based on loading and ambient conditions, enabling proactive load reduction or scheduled maintenance. ML contributes significantly by predicting equipment failures and optimizing energy distribution.

**Renewable Energy Forecasting:** Regression techniques help forecast wind farm and solar panel output. Gaussian process regression has been applied to wind speed/power forecasting to provide not only point forecasts but also uncertainty ranges. Improved forecasts allow better balance of supply and demand and reduce the need for spinning reserves.

**Energy Management:** Industrial plants and commercial buildings use regression models to predict energy consumption of HVAC systems and industrial equipment, then optimize settings to reduce usage during peak price hours or participate in demand response programs. Neural

networks modeling building cooling loads as functions of weather, occupancy, and time enable control systems to adjust thermostats in advance of peak periods.

### **8.3 Chemical Processing**

**Soft Sensors:** ML regression models infer hard-to-measure quality variables (product composition, impurity levels, reaction yields) from easy-to-measure signals (temperatures, pressures, flow rates). These virtual sensors provide near-real-time quality estimates, enabling tighter control and reducing lab sampling frequency.

**Process Modeling and Optimization:** Regression models (neural networks, SVR) predict reaction yields or conversions as functions of feed properties and operating conditions. These models can be coupled with optimization algorithms to find set-points maximizing yield or minimizing energy usage. Hybrid models combining fundamental physics-based models with ML regression have improved model robustness and acceptance.

**Fault Detection and Diagnosis:** Multivariate regression and related techniques form the basis of many process monitoring systems. Significant deviations in predicted versus actual values indicate faults. Gradient boosted regression trees with SHAP analysis have uncovered that specific process fluctuations strongly drive yield variability, guiding process adjustments that reduce variability.

**Predictive Maintenance:** Random forest models predict failure probabilities of pumps, compressors, and heat exchangers based on vibration and flow data. Combined feature selection and regression techniques achieve reliable early warning with manageable false alarms.

### **8.4 Predictive Maintenance**

Predictive maintenance uses data-driven methods to predict equipment faults or remaining useful life so maintenance can be scheduled just in time—reducing unexpected breakdowns and avoiding unnecessary routine maintenance.

**Remaining Useful Life (RUL) Prediction:** Given historical run-to-failure data, regression models output estimated time until failure or maintenance need. Modern approaches use LSTM neural networks ingesting time series of sensor readings to output RUL directly. Hybrid CNN-LSTM models predicting RUL of engines or turbofan systems have achieved over 95% prediction accuracy, learning complex patterns of how sensor signals evolve as machines wear.

**Health Scoring:** Random forests output continuous "health scores" for motors, which can be thresholded to decide when to intervene. Gradient boosting regression predicts torque and vibration behavior of drivetrain components under different conditions to infer wear levels and plan maintenance.

**Wide Asset Coverage:** ML regression models in predictive maintenance have been implemented for motors, pumps, bearings, gearboxes, turbines, engines, and more. In aviation, researchers used

regression to predict aircraft engine remaining life. In renewable energy, AI-based prognostic maintenance techniques forecast failures in wind turbine gearboxes or power converters.

Real-world implementations report sizable reductions in unplanned downtime. One manufacturing plant implementing ML-based predictive maintenance using IoT sensor data and ensemble models cut downtime by approximately 20% while minimizing unnecessary preventive replacements.

## 9. Challenges and Future Directions

### Key Challenges

**Data Quality and Quantity:** Industrial datasets can be noisy, have missing values, or become misaligned in time. Many processes don't generate millions of independent samples, and failures might be extremely rare. Substantial effort must go into data preprocessing, outlier detection, denoising, and normalization. Concept drift occurs as equipment ages or processes change, requiring continuous data collection and periodic retraining.

**Model Interpretability and User Trust:** High-performing regression models like ensembles and neural networks are complex black boxes. Lack of transparency can be a barrier in industrial domains where engineers and managers are cautious about acting on ML recommendations without understanding rationale. Regulatory or safety requirements may demand explainability.

**Generalization and Transferability:** Models often face challenges generalizing beyond conditions or assets they were trained on due to tool-to-tool or machine-to-machine variations. Retraining from scratch for each scenario is expensive. Researchers are investigating transfer learning and domain adaptation to enable models to adjust to new tools or recipes with minimal new data.

**Computational Constraints:** Many industrial applications require low-latency predictions on edge devices or existing control system hardware. Complex models like deep networks can be computationally heavy, driving needs for model compression, optimization, or simpler surrogate models. Real-time requirements are especially stringent in control applications.

**Imbalanced and Limited Failure Data:** For predictive maintenance and quality anomaly detection, models might see thousands of hours of normal operation but only a handful of failure instances. Strategies include oversampling minority classes, cost-sensitive learning, two-stage models, and using physics-based simulation or generative models to create additional training examples for underrepresented scenarios.

**Organizational and Human Challenges:** Engineers may need training to use and maintain ML models. There can be resistance to relying on algorithm outputs over traditional heuristics. Cross-disciplinary skill sets combining physical process understanding with ML tools are necessary. AutoML and easier-to-use frameworks can help lower barriers.

## **Future Research Directions**

**Hybrid Physics-ML Models:** Integrating first-principles knowledge with machine learning shows promise. Physics-informed neural networks (PINNs) are trained on data while satisfying differential equations representing known physics. In manufacturing, hybrid models combining mechanical physics equations with learned components have achieved extremely high prediction accuracy. By embedding physics, these models are more interpretable and require less training data.

**Real-Time and Edge Analytics:** Movement of ML computation to the edge will continue, with future regression models being smaller and more energy efficient. Coupled with digital twins, these models enable real-time simulation and optimization in parallel with operations. Federated learning allows training across multiple edge devices collaboratively without centralizing data, addressing privacy and security concerns.

**Explainable and Ethical AI:** More focus on making ML models explainable by design, including work on explainable boosting machines that remain interpretable while maintaining accuracy. Industry 5.0 themes emphasize human-centric and ethically responsible AI, meaning models should be accurate, fair, transparent, and aligned with human operators.

**Automated ML and Simplified Integration:** AutoML systems that handle model selection, hyperparameter tuning, and validation will enable non-ML experts to deploy regression models. Simplified interfaces integrated into existing industrial software would allow engineers to create predictive models from historical data through intuitive interactions.

**Robustness and Cybersecurity:** As industrial ML models become core to operations, robustness against adversarial conditions or cyber-attacks becomes critical. Future models might incorporate defensive mechanisms, and research on verifying and validating ML models for safety will gain importance in certification of AI-driven control systems.

## **10. Conclusion**

Machine learning regression techniques have emerged as indispensable tools for modeling and optimizing complex industrial systems. This review surveyed a spectrum of methods—from linear and regularized models to nonlinear kernel methods, ensemble approaches, and deep learning discussions, which discussed their respective roles, strengths, and limitations in industrial applications.

Key takeaways include the importance of choosing appropriate model complexity for given problems, balancing accuracy with interpretability and data availability. Even relatively simple models can provide valuable insights into process behavior. Advanced techniques like ensemble trees and deep neural networks often significantly outperform traditional models in capturing subtle, high-dimensional relationships inherent to modern industrial systems.

Across manufacturing, energy, chemical processing, and maintenance domains, ML-driven regression solves real-world problems: improving product quality through predictive modeling,

reducing downtime via accurate equipment failure predictions, optimizing process parameters for yield and efficiency gains, and enabling new capabilities like virtual sensors and digital twins for real-time decision support. Many industrial operations report tangible benefits such as cost savings, productivity improvements, and enhanced reliability after integrating ML-based prediction systems.

The journey to widespread adoption continues, with challenges including data quality management, maintaining model performance over time and across changing conditions, and providing explainability to build user trust. Active research in hybrid modeling combining physics with ML promises accurate and interpretable models. Edge and federated learning approaches aim to bring ML closer to machines for real-time intelligence. Explainable AI techniques increasingly open the "black box," making advanced models more transparent to engineers.

Machine learning regression techniques have already demonstrated significant value in modeling complex industrial processes, and their importance will only grow in the Industry 4.0 era and beyond. By combining data-driven methods with domain knowledge and addressing current limitations, industries can achieve greater efficiency, safety, and innovation. The continued cross-pollination between machine learning and industrial engineering through collaborative research and practical deployments will pave the way for smarter factories, smarter grids, and more resilient industrial ecosystems.

## References:

1. Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32. <https://doi.org/10.1023/A:1010933404324>
2. Burkov, A. (2019). *The hundred-page machine learning book*. Andriy Burkov.
3. Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). <https://doi.org/10.1145/2939672.2939785>
4. Chollet, F. (2018). *Deep learning with Python*. Manning Publications.
5. Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), 273–297. <https://doi.org/10.1007/BF00994018>
6. Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5), 1189–1232. <https://doi.org/10.1214/aos/1013203451>
7. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.
8. Hastie, T., Tibshirani, R., & Friedman, J. (2017). *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed.). Springer. <https://doi.org/10.1007/978-0-387-84858-7>
9. Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1), 55–67. <https://doi.org/10.1080/00401706.1970.10488634>
10. Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245), 255–260. <https://doi.org/10.1126/science.aaa8415>

11. Khan, S., Yairi, T., & Khan, S. A. (2019). A review on the application of deep learning in system health management. *Mechanical Systems and Signal Processing*, 107, 241–265. <https://doi.org/10.1016/j.ymsp.2018.02.040>
12. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444. <https://doi.org/10.1038/nature14539>
13. Venkata Krishna Bharadwaj Parasaram. (2021). Explainable Machine Learning Models for Improving Decision Making in Project Portfolio Management. *Darpan International Research Analysis*, 9(1), 12–21. <https://doi.org/10.36676/dira.v9.i1.188>
14. Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems* (pp. 4765–4774).
15. Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian processes for machine learning*. MIT Press.
16. Schölkopf, B., Smola, A. J., Williamson, R. C., & Bartlett, P. L. (2000). New support vector algorithms. *Neural Computation*, 12(5), 1207–1245. <https://doi.org/10.1162/089976600300015565>
17. Shickel, B., Tighe, P. J., Bihorac, A., & Rashidi, P. (2018). Deep EHR: A survey of recent advances in deep learning techniques for electronic health record analysis. *IEEE Journal of Biomedical and Health Informatics*, 22(5), 1589–1604. <https://doi.org/10.1109/JBHI.2017.2767063>