SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

Manufacturing Enterprises: An Explainable Machine Learning Framework for Defect Risk and Process Performance Optimization

Suresh Kumar Sahani[1], Tsair-Fwu Lee[2], Digvijay Pandey[3], Binay Kumar Pandey[4], Shivendra Labh Karna[5], Bharat Kumar Sah[6*]

[1]Faculty of Science, Technology and Engineering, Rajarshi Janak University, Janakpurdham, Nepal. E-mail: sureshsahani54@gmail.com
[2]National Kaohsiung University of Science and Technology, Taiwan. E-mail: tflee@nkust.edu.tw
[3]Department of Technical Education, Uttar Pradesh, India. E-mail: digit11011989@gmail.com
[4]Department of Information Technology, College of Technology, Govind Ballabh Pant University of Agriculture and Technology, Pantnagar, Uttarakhand, India. E-mail:binaydece@gmail.com
[5]Faculty of Management, R.R.M. Campus, Janakpur, Nepal, 45600. E-mail:shivendralabhkarn@gmail.com
[6*]Faculty of Science, Technology, and Engineering, Rajarshi Janak University, Janakpurdham, Nepal. E-mail: bharatsah@rju.edu.np

Abstract
Nepal's manufacturing sector is a structural cripple. Contributing a mere 5.72 percent to GDP — a figure that would embarrass any comparable developing economy — it grew at a sluggish 2.83 percent in 2025/26 while the tertiary sector ballooned to 61.8 percent and remittances swallowed 33 percent of national income. Beneath these macroeconomic indignities lies a more insidious failure: the vast majority of Nepal's manufacturing enterprises, particularly the small and medium enterprises that constitute 99.7 percent of registered businesses, manage quality through intuition, post-hoc inspection, and the occasional ISO 9001 certificate that gathers dust on a factory wall. Defects are discovered after they are produced, scrap rates are absorbed as "normal loss," and process capability remains a foreign concept to floor supervisors who have never seen a control chart.
This paper constructs a rigorous theoretical framework for predictive quality management in Nepalese manufacturing enterprises through explainable machine learning. We derive the complete mathematical architecture of quality loss — from Taguchi's quadratic loss function through process capability indices (Cp, Cpk, Cpm) and Shewhart statistical process control — and integrate it with ensemble learning methodologies (XGBoost, LightGBM, Random Forest) and anomaly detection via Isolation Forest. Post-hoc interpretability is grounded in SHAP values from cooperative game theory, enabling factory managers to translate black-box predictions into actionable process adjustments. We prove the expected-loss decomposition, derive the relationship between Cpk and Cpm, establish the distribution of Isolation Forest path lengths, derive the Average Run Length (ARL) of the EWMA control chart, prove consistency of the mutual-information feature selector, establish the bias-variance decomposition for gradient-boosted ensembles, and verify the four Shapley axioms for the TreeSHAP algorithm. Drawing upon Nepal's Economic Survey 2025/26, the ADB Asia SME Monitor 2024, and the Industrial Enterprises Act 2076 (2020), we situate the framework within Nepal's institutional reality: 923,000 registered businesses, 90 percent of them SMEs, manufacturing concentrated in agro-processing, textiles, and construction materials, and an industrial policy architecture that incentivizes cottage industries but rarely demands statistical rigor. The paper argues that without predictive quality management, Nepal's manufacturing sector will remain permanently trapped in low-value assembly and import substitution — incapable of competing with Indian and Chinese imports, incapable of generating the productive employment that 500,000 annual labor market entrants desperately need, and incapable of justifying the infrastructure investments that cheaper hydropower now makes possible.

Keywords: Predictive Quality Management, Explainable Artificial Intelligence, Manufacturing, Nepal, Taguchi Loss Function, Process Capability Index, Isolation Forest, SHAP, LightGBM, Statistical Process Control, Rademacher Complexity, Average Run Length, Bias-Variance Decomposition.

Introduction

Walk through the factory floor of a typical Nepalese manufacturing enterprise — a rice mill in the Terai, a textile weaving unit in Bhaktapur, a cement block producer on the outskirts of Kathmandu — and you will witness a quality management philosophy that has not fundamentally changed since the 1950s. Raw materials arrive, are inspected by eye and hand, fed into machines whose settings are adjusted by experienced operators according to feel and tradition, and the finished product is checked at the end of the line. If it looks acceptable, it ships. If it does not, it is reworked or scrapped. The cost of this defect is recorded, if at all, as an inevitable cost of doing business. The idea that quality can be predicted before the defect occurs — that the interaction of temperature, humidity, machine vibration, and raw material moisture content could be mathematically modeled to forecast a dimensional deviation — is not merely absent; it is alien.
This is not a caricature. The Nepal Economic Survey 2025/26 records manufacturing's gross value added at Rs. 331 billion, growing at 2.83 percent — better than the 2.27 percent of the previous year, but a fraction of the

electricity sector's 20.93 percent growth and utterly inadequate for a country attempting to industrialize. The Asian Development Bank's Asia SME Monitor 2024 confirms that manufacturing constitutes only 20.1 percent of Nepal's SME sector, dwarfed by agriculture (33.6 percent) and services. The Industrial Enterprises Act 2076 (2020) classifies enterprises by fixed capital — micro (up to NPR 2 million), small (up to NPR 150 million), medium (up to NPR 500 million), large (above NPR 500 million) — but says nothing about process capability, statistical control, or quality predictability. The result is an industrial ecosystem where size is regulated and quality is not.

The cost of this neglect is staggering. The American Society of Quality estimates that poor quality generates 10 to 15 percent of operating expenses in manufacturing companies globally. In Nepal, where profit margins are thin, energy costs are volatile despite hydropower expansion, and imported raw materials are subject to currency fluctuation and border friction, a 10 percent quality cost can mean the difference between solvency and closure. A cement block that fails compression testing after curing represents not just the loss of that block, but the wasted cement, sand, water, labor, machine time, and energy that went into it — what quality engineers call the "hidden factory," the parallel production system devoted to making defects and then fixing them.

Machine learning offers a way out of this trap — but only if deployed with mathematical discipline and interpretable accountability. Ensemble methods can model the high-dimensional, nonlinear relationships between process parameters and quality outcomes with accuracies exceeding 90 percent. Isolation Forests can detect anomalous process conditions in real time without requiring labeled defect histories. SHAP values can decompose predictions into feature-specific contributions that a factory supervisor with secondary education can understand and act upon. But these tools are not magic. They require mathematical foundations: the quadratic structure of quality loss, the probabilistic logic of process capability, the information-theoretic criteria for feature selection, and the game-theoretic axioms that make explanations fair and consistent.

This paper builds those foundations. It is not an empirical study claiming to have deployed a model in a Nepalese factory and achieved a specific defect reduction percentage. Such claims would be dishonest; the data infrastructure, sensor networks, and labeled quality records required for such a study do not yet exist at scale in Nepal. Instead, this paper constructs the theoretical architecture that any such deployment would require — mathematically complete, regulatorily aligned with Nepal's Industrial Enterprises Act and ISO 9001 implementation frameworks, and ethically grounded in the imperative that automated quality decisions must be explainable to the workers and managers whose livelihoods depend on them.

We make seven contributions. First, we unify Taguchi's quality loss theory, process capability analysis, and statistical process control into a single mathematical framework that defines the objective function for predictive quality management, with full proofs of the expected-loss decomposition and the C_{pk}–C_{pm} relationship. Second, we derive the complete optimization architecture of ensemble gradient boosting and Isolation Forest anomaly detection, including the second-order Taylor expansion, the optimal leaf weight theorem, the gain formula derivation, and the distributional theory of Isolation Forest path lengths. Third, we derive the Average Run Length (ARL) of the EWMA control chart, providing the theoretical foundation for selecting the smoothing parameter λ and control limit multiplier L to achieve target in-control and out-of-control ARL. Fourth, we prove consistency of the mutual-information feature selector, ensuring that feature ranking converges to the true ranking as the sample size grows. Fifth, we establish the bias-variance decomposition for gradient-boosted ensembles and Rademacher complexity generalization bounds, giving PAC-style guarantees on the gap between training and population loss. Sixth, we verify the four Shapley axioms (efficiency, symmetry, dummy, additivity) for the TreeSHAP algorithm and derive its polynomial-time complexity. Seventh, we map the entire framework onto Nepal's institutional landscape — its SME structure, its industrial policy, its ISO 9001 adoption trajectory, and its infrastructure constraints — identifying what is theoretically possible, what is practically feasible, and what must change before theory becomes reality.

The argument is simple and urgent. Nepal cannot consume its way to prosperity on remittances and imported goods. It must manufacture. And it cannot manufacture competitively without predicting and preventing defects before they occur. The mathematics in this paper is the blueprint for that transition.

The remainder of this paper is organized as follows. Section 2 reviews the related literature on quality management theory, machine learning in manufacturing, anomaly detection, and explainable AI. Section 3 develops the theoretical framework and mathematical foundations, including the quality loss function with full proof of the expected-loss decomposition, process capability indices with the C_{pk}–C_{pm} relationship theorem, statistical process control with the EWMA ARL derivation, gradient boosting with the second-order Taylor expansion and optimal leaf weight theorem, Isolation Forest with the path length distribution, SHAP with the four Shapley axioms, and information-theoretic feature selection with consistency. Section 4 specifies the proposed XML-PQM Framework for Nepalese manufacturing. Section 5 establishes the convergence and generalization theory, including the bias-variance decomposition, Rademacher complexity bounds, and Isolation Forest consistency. Section 6 discusses policy implications and regulatory alignment. Section 7 addresses challenges, limitations, and ethical considerations. Section 8 concludes. Four appendices collect the notation table, the full proof of Shapley axiom satisfaction, the EWMA ARL derivation, and the Isolation Forest path length distribution proof.

2. Literature Review

2.1 From Inspection to Prediction: The Evolution of Quality Management Theory

The intellectual history of quality management is a progression from after-the-fact judgment to before-the-fact prevention. Shewhart's (1931) foundational work, Economic Control of Quality of Manufactured Product, introduced statistical process control (SPC) as a methodology for distinguishing assignable-cause variation from common-cause variation — the difference between a process that is inherently unstable and one that is merely noisy. The SPC paradigm, operationalized through control charts with upper and lower control limits set at three standard deviations from the process mean, transformed quality from a subjective assessment into a statistical discipline.

Taguchi (1986) advanced this framework with the quality loss function (QLF), rejecting the binary "goalpost mentality" that treated all outputs within specification limits as equally acceptable. Taguchi argued that loss begins the instant a quality characteristic deviates from its target value, and that this loss accelerates quadratically. The QLF shifted quality management from conformance to specification toward minimization of variation around a target — a philosophical reorientation with profound mathematical and financial implications. Under Taguchi's formulation, two products both within specification can generate vastly different quality costs depending on their proximity to the target, rendering traditional pass-fail inspection economically irrational.

Process capability indices emerged as the quantitative bridge between Taguchi's philosophical framework and shop-floor practice. C_p, C_{pk}, and C_{pm} provide scalar measures of whether a process's natural variation fits within engineering tolerances and whether the process mean is centered on the target. These indices enabled quality engineers to communicate process health in a single number, facilitating supplier qualification, capital investment decisions, and continuous improvement prioritization. The mathematical relationships among these indices — established formally in Section 3.2 — reveal that they are not redundant but complementary: each captures a distinct aspect of process performance (potential capability, actual capability, target sensitivity).

In Nepal, however, this theoretical evolution has barely penetrated operational practice. ISO 9001 certification has grown among manufacturing enterprises seeking export market access and government contracts, but implementation often remains superficial — documentation without statistical underpinning, audit compliance without process capability measurement. The gap between quality theory and manufacturing reality in Nepal is not a knowledge gap; it is an infrastructure, incentive, and capacity gap.

2.2 Machine Learning in Manufacturing Quality Control

The application of machine learning to manufacturing quality has expanded dramatically. Karayel (2009) employed feedforward neural networks to predict surface roughness in CNC lathe operations using cutting depth, speed, and feed rate as inputs. Tsai et al. (1999) developed similar neural network architectures for milling operations. These early applications demonstrated that nonlinear models could capture process-parameter-quality relationships that linear regression missed.

More recently, deep learning has penetrated visual defect detection. Wang et al. (2018) proposed convolutional neural networks for automatic defect detection on flat surfaces, while Wang et al. (2020) extended this to geometrically complex products such as turbo blades through cloud-based platforms. Zhao et al. (2020) compared regression models and neural networks for predicting nugget diameter in spot-welded joints, finding neural networks superior. The systematic review by PMC (2023) documented the breadth of neural network applications in manufacturing quality control, from real-time process monitoring to defect classification.

Ensemble methods have proven particularly effective for tabular process data — the sensor readings, machine parameters, and environmental conditions that constitute the bulk of manufacturing data in Nepalese SMEs. Lessmann et al. (2015) benchmarked 41 classification algorithms across credit scoring datasets, establishing ensemble superiority; similar patterns hold in manufacturing defect prediction. Gradient boosting machines, particularly XGBoost (Chen & Guestrin, 2016) and LightGBM (Ke et al., 2017), have become the default choice for structured predictive tasks due to their ability to model feature interactions without explicit specification. The theoretical foundations of gradient boosting — including the second-order Taylor expansion that distinguishes XGBoost from Friedman's (2001) original first-order formulation — are derived in full in Section 3.4, with the optimal leaf weight theorem (Theorem 3.3) and the gain formula (3.25) providing the algorithmic core.

Recent studies increasingly emphasize the integration of mathematical modeling, machine learning (ML), uncertainty quantification (UQ), and risk-sensitive decision-making in Nepalese organizational systems. Sahani et al. (2026) introduced an ML-enhanced framework combining fractional stochasticity, Wasserstein robustness, and rough-path neural filtering to address deep uncertainty in volatile environments. Complementing this work, Sahani et al. (2026) proposed hybrid analytical, numerical, and ML methods for deterministic and stochastic differential equations, providing theoretical guarantees for stability, convergence, and uncertainty propagation. Sahani et al. (2026) subsequently applied these computational approaches to

improve operational stability, strategic planning, and risk management in Nepalese institutions, particularly where information is incomplete or rapidly changing.

These hybrid approaches have also been extended to several specialized sectors. Sahani et al. (2026) developed a mathematical framework for risk management and decision-making in dental practice management systems. Sahani et al. (2026) incorporated biotechnological catalysts into hybrid computational architectures to support process optimization and risk-aware management decisions. Similarly, Sahani et al. (2026) designed a framework for Nepal's healthcare-management sector, focusing on resource allocation, operational efficiency, risk prediction, and strategic planning under financial and social constraints. Together, these studies demonstrate the effectiveness of hybrid systems in representing nonlinear relationships, stochastic disruptions, and uncertainty across diverse operational settings.

In the financial sector, Sahani et al. (2026) reviewed explainable ML methods for credit-risk assessment and intelligent lending decisions in Nepalese cooperative banks. The study examined XGBoost and LightGBM alongside interpretability techniques such as SHAP and LIME. It also considered information-theoretic feature selection, class-imbalance correction, fairness constraints, and convergence guarantees. This work established an important foundation for transparent and data-driven credit evaluation in Nepal's cooperative banking sector.

Research Gap

Despite these developments, existing studies mainly concentrate on static credit classification, general risk prediction, or isolated sector-specific applications. They do not provide a unified framework capable of simultaneously addressing the following dimensions:

Time-to-default prediction: Existing models rarely capture when a borrower is likely to default or how credit risk develops over time.

Network-based contagion: Borrower relationships, financial dependencies, and the transmission of risk across connected networks remain insufficiently modeled.

Privacy-preserving collaboration: Cooperative banks lack a mechanism for jointly training predictive models without exchanging sensitive customer data.

Uncertainty and causal explanation: Current approaches do not adequately separate data-related uncertainty from model uncertainty or provide actionable causal explanations for lending outcomes.

2.3 Anomaly Detection for Unsupervised Quality Monitoring

A critical challenge in Nepalese manufacturing is the absence of labeled defect datasets. Many SMEs lack systematic quality recording; defects are noticed, discarded, and forgotten. Supervised learning — which requires thousands of labeled examples — is therefore often infeasible. Anomaly detection offers an alternative.

Isolation Forest, introduced by Liu et al. (2008), detects anomalies by exploiting the property that isolating an anomalous point requires fewer random splits than isolating a normal point. Unlike density-based methods that struggle with high-dimensional data, Isolation Forest operates efficiently in spaces with hundreds of features — matching the dimensionality of modern manufacturing sensor streams. The theoretical foundation of Isolation Forest — including the distribution of path lengths in randomly constructed isolation trees and the normalization by the average path length of unsuccessful binary search — is derived in Section 3.5, with the path length distribution theorem (Theorem 3.4) providing the formal basis for the anomaly score.

For Nepalese enterprises transitioning from manual inspection to automated monitoring, Isolation Forest provides a mechanism to flag unusual process conditions without requiring historical defect labels. The computational complexity $\mathcal{O}(t \cdot \psi \cdot \log\psi)$ for training and $\mathcal{O}(t \cdot \log\psi)$ for prediction makes it deployable on modest hardware — a Raspberry Pi at a rural mill, an edge gateway at a Kathmandu factory.

Autoencoder-based approaches offer another unsupervised pathway. By training a neural network to reconstruct normal process data and flagging instances with high reconstruction error as anomalous, autoencoders learn the manifold of acceptable process behavior. Variational autoencoders extend this by modeling the latent distribution probabilistically, enabling uncertainty quantification that is valuable in manufacturing environments where "normal" itself drifts with raw material batches and seasonal conditions.

2.4 Explainable AI in Manufacturing

The opacity of complex ML models creates a barrier to adoption in manufacturing, where engineers and operators must trust and act upon algorithmic recommendations. Senoner, Netland, and Feuerriegel (2021) addressed this challenge directly, developing a data-driven decision model that uses nonlinear modeling with SHAP values to infer how production parameters and process quality are related in semiconductor manufacturing. Their field experiment at a leading high-power semiconductor manufacturer demonstrated a 21.7 percent reduction in yield loss compared to baseline — a result validated through post-experimental roll-out. This work is seminal because it proves that XAI is not merely an academic exercise; it generates measurable operational value.

The comprehensive review by Springer (2025) surveyed XAI applications across additive manufacturing, process optimization, and defect detection, identifying SHAP and LIME as the dominant explanation techniques. In additive manufacturing, SHAP values have identified design parameters impacting porosity and Poisson's ratio. In process optimization, SHAP values have quantified production parameter contributions to wafer yield. These applications confirm that the mathematics of Shapley values — originally developed for cooperative game theory by Shapley (1953) — translates directly into manufacturing decision support. The four axioms that uniquely characterize Shapley values (efficiency, symmetry, dummy, additivity) are formally stated and verified in Section 3.6, with the uniqueness theorem (Theorem 3.5) ensuring that no other attribution method satisfies all four desirable properties simultaneously.

In Nepal, where trust in automated systems is low and technical literacy varies enormously across the workforce, explainability is not a luxury feature. It is a prerequisite for adoption. A factory owner in Birgunj will not delegate quality decisions to a black box. But she might delegate them to a system that can explain, in Nepali and in plain language, that "the humidity spike at 2 PM and the below-target mixing speed together increased the probability of dimensional deviation by 18 percent."

3. Theoretical Framework and Mathematical Foundations

This section develops the complete mathematical architecture underlying the proposed XML-PQM Framework. We begin with the Taguchi quality loss function and prove the expected-loss decomposition, then derive the process capability indices and prove their inter-relationships, develop the statistical process control framework including the EWMA Average Run Length derivation, construct the gradient boosting objective with the second-order Taylor expansion and optimal leaf weight theorem, derive the Isolation Forest anomaly score with its path length distribution, develop the SHAP value framework with verification of the four Shapley axioms, and conclude with information-theoretic feature selection including consistency.

3.1 The Quality Loss Function: From Binary Conformance to Quadratic Deviation

The foundational objective of predictive quality management is the minimization of quality loss. Taguchi's quality loss function provides the mathematical formalization. Let Y be a quality characteristic with target value T. For a single unit with realized value y, the loss is:

$$L(y) = k\,(y - T)^2 \qquad (3.1)$$

where k is the loss coefficient determined by the cost at the specification limit. If the cost of producing a unit at the tolerance boundary Δ from target is A, then:

$$k = \frac{A}{\Delta^2} \qquad (3.2)$$

The quadratic form is not arbitrary; it encodes the physical reality that loss accelerates with deviation. A component deviating two units from target generates four times the loss of a component deviating one unit. In high-volume manufacturing, this quadratic penalty compounds catastrophically: a cement block plant producing 1,000 blocks per day with mean deviation 0.5Δ from target incurs daily loss $1000 \cdot k \cdot 0.25\Delta^2 = 250A$; doubling the mean deviation to Δ multiplies the daily loss to 1000A, a fourfold increase from a twofold deviation increase.

Proposition 3.1 (Calibration of the Loss Coefficient). If the financial loss at deviation Δ from target is A (typically the scrap or rework cost of a unit at the specification limit), then the loss coefficient that makes the quadratic loss function (3.1) consistent with this anchor point is $k = A/\Delta^2$.

Proof. Substitute $y = T + \Delta$ into (3.1): $L(T + \Delta) = k\Delta^2$. Setting this equal to the anchor loss A and solving for k yields $k = A/\Delta^2$. ▫

This calibration is operationally important: it links the abstract loss function to a measurable financial quantity (the cost of producing a unit at the specification limit, which any Nepalese manufacturer can compute from scrap value, labor cost, and material cost records).

The expected loss per unit, integrated over the distribution of Y, is the central quantity that predictive quality management seeks to minimize. The following theorem provides the fundamental decomposition.

Theorem 3.1 (Expected Loss Decomposition). Let Y be a quality characteristic with mean $\mu = \mathbb{E}[Y]$ and variance $\sigma^2 = \mathrm{Var}(Y)$, and let T be the target value. Under the quadratic loss function (3.1), the expected loss per unit decomposes as:

$$\mathbb{E}[L(Y)] = k[\sigma^2 + (\mu - T)^2] \qquad (3.3)$$

Proof. Expand the squared deviation:

$$\mathbb{E}[(Y - T)^2] = \mathbb{E}[(Y - \mu + \mu - T)^2] = \mathbb{E}[(Y - \mu)^2] + 2\,\mathbb{E}[(Y - \mu)(\mu - T)] + (\mu - T)^2 \qquad (3.4)$$

The first term is $\mathbb{E}[(Y-\mu)^2] = \mathrm{Var}(Y) = \sigma^2$. The second term is $2(\mu - T)\,\mathbb{E}[Y-\mu] = 2(\mu-T)\cdot 0 = 0$, since $\mathbb{E}[Y-\mu] = 0$ by definition of the mean. The third term is $(\mu - T)^2$, a constant with respect to the expectation. Multiplying by k yields (3.3). ▫

This decomposition is strategically decisive. The first term, $k\,\sigma^2$, captures the cost of variability — the spread of outputs around the process mean. The second term, $k\,(\mu - T)^2$, captures the cost of inaccuracy — the distance between the process mean and the target. A process can have low variance but high loss if its mean is off-target; conversely, a process can be perfectly centered but generate unacceptable loss if its variance is excessive. Predictive quality management must minimize both simultaneously: variance reduction requires process stabilization (tighter machine tolerances, better raw material control), while accuracy improvement requires mean adjustment (calibration, setpoint correction).

Corollary 3.1 (Loss-Minimizing Mean). For a fixed variance σ^2, the expected loss $\mathbb{E}[L(Y)]$ is uniquely minimized at $\mu = T$.

Proof. The expected loss (3.3) is $k[\sigma^2 + (\mu - T)^2]$. For fixed σ^2, this is a strictly convex quadratic in μ with minimum at $\partial/\partial\mu\,[k(\mu-T)^2] = 2k(\mu - T) = 0$, giving $\mu^* = T$. The second derivative $2k > 0$ confirms the minimum. ▫

This corollary formalizes the intuition that centering the process on the target is always optimal, regardless of variance — a fact that is sometimes obscured by the goalpost mentality which treats any value within specification as equally acceptable.

For "smaller-is-better" characteristics — defect counts, scrap rates, rework hours — the target is zero and the loss function becomes:

$$L(y) = k\,y^2, \qquad y \geq 0 \tag{3.5}$$

For "larger-is-better" characteristics — tensile strength, material purity, equipment uptime — the ideal is unbounded and the loss function inverts:

$$L(y) = \frac{k}{y^2}, \qquad y > 0 \tag{3.6}$$

In Nepalese manufacturing, where most SMEs produce for price-sensitive domestic markets, the nominal-is-best formulation (3.1) dominates: shaft diameters, block compressive strengths, fabric thread counts, food product weights. The QLF provides the objective function that any predictive model must ultimately serve.

3.2 Process Capability: Quantifying the Voice of the Process

Process capability indices relate the voice of the process (its natural variation) to the voice of the customer (specification limits). Let USL and LSL denote the upper and lower specification limits, and let μ and σ denote the process mean and standard deviation.

The potential capability index C_p measures whether the process spread fits within the tolerance band:

$$C_p = \frac{USL - LSL}{6\sigma} \tag{3.7}$$

A $C_p > 1$ indicates that the process spread (6σ, covering approximately 99.73% of a normal distribution) is narrower than the specification width, suggesting potential capability if centered. However, C_p is blind to centering: a process with $C_p = 2.0$ but mean shifted to the upper specification limit will still produce massive defects.

The performance capability index C_{pk} corrects this by measuring the distance from the process mean to the nearest specification limit, normalized by three standard deviations:

$$C_{pk} = \min\left(\frac{USL - \mu}{3\sigma},\ \frac{\mu - LSL}{3\sigma}\right) \tag{3.8}$$

C_{pk} can be decomposed into upper and lower one-sided indices:

$$C_{pu} = \frac{USL - \mu}{3\sigma}, \qquad C_{pl} = \frac{\mu - LSL}{3\sigma} \tag{3.9}$$

$$C_{pk} = \min(C_{pu}, C_{pl}) \tag{3.10}$$

A process is generally considered capable when $C_{pk} \geq 1.33$ (the threshold for Six Sigma qualification at 4 sigma capability). When $C_{pk} < 1$, the process mean lies within the specification limits but a portion of the natural

variation exceeds them. When $C_{pk} < 0$, the process mean has violated a specification limit entirely.
The Taguchi capability index C_{pm} incorporates target sensitivity:

$$C_{pm} = \frac{USL - LSL}{6\sqrt{\sigma^2 + (\mu - T)^2}} \tag{3.11}$$

C_{pm} penalizes deviation from target directly in the denominator, aligning capability measurement with the QLF.
The following theorem establishes the precise relationship between C_{pk} and C_{pm}.
Theorem 3.2 (C_{pk}–C_{pm} Relationship). Let C_{pk} and C_{pm} be the capability indices defined in (3.8) and (3.11) for a process with mean μ, variance σ^2, target T, and symmetric specification limits $USL = T + \Delta$, $LSL = T - \Delta$. Then:

$$C_{pm} = \frac{C_{pk}}{\sqrt{1 + \left(\frac{\mu - T}{\sigma}\right)^2}} \cdot \frac{1}{\sqrt{1 + \frac{2|\mu - T|}{3\sigma C_{pk}} - \frac{(\mu - T)^2}{9\sigma^2 C_{pk}^2}}} \tag{3.12}$$

In the special case where the process is centered ($\mu = T$), $C_{pm} = C_{pk} = C_p$. More generally, $C_{pm} \leq C_{pk}$ with equality if and only if $\mu = T$.
Proof. Define the standardized shift $\delta = (\mu - T)/\sigma$ and the half-tolerance $d = (USL - LSL)/2 = \Delta$. By symmetry of the specification limits around T, the centered capability is $C_p = d/(3\sigma)$.
For C_{pk}, assuming without loss of generality that $\mu \geq T$ (so the nearest specification is USL), we have

$$C_{pk} = (USL - \mu)/(3\sigma) = (T + \Delta - \mu)/(3\sigma) = \big(d - (\mu - T)\big)/(3\sigma) = \big(d/(3\sigma)\big) - (\mu - T)/(3\sigma) = C_p - \delta/3.$$

Therefore $C_{pk} = C_p - \delta/3$, or equivalently $C_p = C_{pk} + \delta/3$.

For C_{pm}, the denominator is $6\sqrt{\sigma^2 + (\mu - T)^2} = 6\sigma\sqrt{1 + \delta^2}$, so $C_{pm} = 2d/\big(6\sigma\sqrt{1 + \delta^2}\big) = \big(d/(3\sigma)\big)/\sqrt{1 + \delta^2} = C_p/\sqrt{1 + \delta^2}$.
Substituting $C_p = C_{pk} + \delta/3$:

$$C_{pm} = \frac{C_{pk} + \delta/3}{\sqrt{1 + \delta^2}} \tag{3.13}$$

Since $\sqrt{1 + \delta^2} \geq 1$ with equality iff $\delta = 0$ (i.e., $\mu = T$), and since $C_{pk} + \delta/3 = C_p > 0$, we have $C_{pm} \leq C_{pk} + \delta/3 = C_p$. Furthermore, since $C_{pk} \leq C_p$ (with equality iff $\mu = T$), we conclude $C_{pm} \leq C_{pk}$ when $\delta = 0$, and $C_{pm} < C_{pk}$ when $\delta \neq 0$. ▫
Corollary 3.2 (Equality Condition). $C_{pm} = C_{pk}$ if and only if the process is centered on target ($\mu = T$). When the process is centered, all three indices coincide: $C_p = C_{pk} = C_{pm}$.
Proof. From Theorem 3.2, $C_{pm} = C_{pk}$ requires $\delta = 0$ (i.e., $\mu = T$), in which case $C_p = C_{pk}$ (since $C_{pk} = C_p - \delta/3 = C_p$) and $C_{pm} = C_p/\sqrt{1 + 0} = C_p$. ▫
This theorem has practical consequences for Nepalese manufacturing: a process with $C_{pk} = 1.5$ but mean shifted by 1.5σ from target has $C_{pm} = 1.5/\sqrt{1 + 2.25} = 1.5/1.80 = 0.83$, indicating that despite a respectable C_{pk}, the process is generating significant Taguchi loss due to off-target operation. The C_{pm} index is therefore the more economically meaningful measure for processes where target centering is critical — pharmaceutical dosing, precision machining, ceramic firing.
In the context of predictive quality management, these indices serve as both diagnostic tools and optimization targets. The ensemble model predicts $\hat{\mu}$ and forecasts $\hat{\sigma}$ for future production batches; capability indices computed from these predictions enable proactive intervention before capability degradation produces physical defects.
3.3 Statistical Process Control: The Mathematics of Stability
Shewhart control charts monitor process stability by distinguishing common-cause variation from assignable-cause variation. For a quality characteristic measured in subgroups of size n, let $\bar{X}_i$ denote the mean of the i-th subgroup and R_i denote its range.
The grand mean and average range are:

$$\bar{\bar{X}} = \frac{1}{m}\sum_{i=1}^{m} \bar{X}_i, \qquad \bar{R} = \frac{1}{m}\sum_{i=1}^{m} R_i \tag{3.14}$$

For an $\bar{X}$-chart, the control limits are:

$$\mathrm{UCL}_{\bar{X}} = \bar{\bar{X}} + A_2\,\bar{R}, \qquad \mathrm{LCL}_{\bar{X}} = \bar{\bar{X}} - A_2\,\bar{R} \tag{3.15}$$

where A_2 is a control chart constant dependent on subgroup size. For an R-chart:

$$\mathrm{UCL}_R = D_4\,\bar{R}, \qquad \mathrm{LCL}_R = D_3\,\bar{R} \tag{3.16}$$

where D_3 and D_4 are similarly tabulated constants. These limits are not specification limits; they are derived from the process's own historical variation. A point outside control limits signals that assignable cause variation has entered the system — tool wear, material change, operator error, environmental shift.
For individual measurements ($n = 1$), the Individuals-Moving Range (I-MR) chart replaces subgroup ranges with moving ranges of consecutive observations:

$$\mathrm{MR}_i = |X_i - X_{i-1}|, \qquad \overline{\mathrm{MR}} = \frac{1}{m-1}\sum_{i=2}^{m}\mathrm{MR}_i \tag{3.17}$$

$$\hat{\sigma} = \frac{\overline{\mathrm{MR}}}{d_2} \tag{3.18}$$

where $d_2 = 1.128$ for moving ranges of span 2. The control limits become:

$$\mathrm{UCL}_X = \bar{X} + \frac{3\,\overline{\mathrm{MR}}}{d_2}, \qquad \mathrm{LCL}_X = \bar{X} - \frac{3\,\overline{\mathrm{MR}}}{d_2} \tag{3.19}$$

For autocorrelated manufacturing data — where consecutive measurements are not independent — the Exponentially Weighted Moving Average (EWMA) chart provides superior detection of small shifts. The EWMA statistic is:

$$z_t = \lambda\,x_t + (1-\lambda)\,z_{t-1}, \qquad 0 < \lambda \le 1 \tag{3.20}$$

with $z_0 = \mu_0$ (the target or historical mean). The control limits for EWMA are:

$$\mathrm{UCL}_{\mathrm{EWMA}} = \mu_0 + L\,\sigma\sqrt{\frac{\lambda}{2-\lambda}\left[1-(1-\lambda)^{2t}\right]} \tag{3.21}$$

$$\mathrm{LCL}_{\mathrm{EWMA}} = \mu_0 - L\,\sigma\sqrt{\frac{\lambda}{2-\lambda}\left[1-(1-\lambda)^{2t}\right]} \tag{3.22}$$

where L is a multiplier (typically $L = 2.814$ for $\lambda = 0.2$ to achieve in-control ARL ≈ 370).
The derivation of the EWMA variance, and hence the control limits, requires computing the variance of z_t. The following theorem establishes this variance.
Theorem 3.3 (EWMA Variance). Let $z_t = \lambda x_t + (1-\lambda)z_{t-1}$ with $z_0 = \mu_0$, where $\{x_t\}$ are i.i.d. with mean μ_0 and variance σ^2. Then:

$$\mathrm{Var}(z_t) = \sigma^2\left(\frac{\lambda}{2-\lambda}\right)\left[1-(1-\lambda)^{2t}\right] \tag{3.23}$$

As $t \to \infty$, the variance converges to the steady-state value $\sigma^2\lambda/(2-\lambda)$.
Proof. Unrolling the recursion:

$$z_t = \lambda x_t + (1-\lambda)[\lambda x_{t-1} + (1-\lambda)z_{t-2}] = \lambda x_t + \lambda(1-\lambda)x_{t-1} + (1-\lambda)^2 z_{t-2} \tag{3.24}$$

Continuing to unroll to $z_0 = \mu_0$:

$$z_t = \lambda\sum_{i=0}^{t-1}(1-\lambda)^i\,x_{t-i} + (1-\lambda)^t\mu_0 \tag{3.25}$$

Since μ_0 is a constant and $\{x_t\}$ are independent with variance σ^2:

$$\mathrm{Var}(z_t) = \lambda^2\sum_{i=0}^{t-1}(1-\lambda)^{2i}\,\sigma^2 = \lambda^2\sigma^2\cdot\frac{1-(1-\lambda)^{2t}}{1-(1-\lambda)^2} \tag{3.26}$$

where the last equality uses the finite geometric series $\sum_{i=0}^{t-1} r^i = (1-r^t)/(1-r)$ with $r = (1-\lambda)^2$. Simplifying the denominator: $1-(1-\lambda)^2 = 1-(1-2\lambda+\lambda^2) = 2\lambda-\lambda^2 = \lambda(2-\lambda)$. Therefore:

$$\mathrm{Var}(z_t) = \lambda^2\sigma^2\cdot\frac{1-(1-\lambda)^{2t}}{\lambda(2-\lambda)} = \sigma^2\cdot\frac{\lambda}{2-\lambda}\cdot\left[1-(1-\lambda)^{2t}\right] \tag{3.27}$$

As $t \to \infty$, $(1-\lambda)^{2t} \to 0$ (since $|1-\lambda| < 1$ for $0 < \lambda < 1$), giving the steady-state variance $\sigma^2\lambda/(2-\lambda)$. ▫
The control limits (3.21)–(3.22) follow directly by setting the limits at $z_t = \mu_0 \pm L\sqrt{\mathrm{Var}(z_t)}$.

Definition 3.1 (Average Run Length). The Average Run Length (ARL) of a control chart is the expected number of samples plotted before the chart signals an out-of-control condition. The in-control ARL (ARL_0) is the ARL when the process is in control (no shift has occurred); the out-of-control ARL (ARL_1) is the ARL after a shift of specified magnitude.

A well-designed control chart has a large ARL_0 (to minimize false alarms) and a small ARL_1 (to detect real shifts quickly). For a Shewhart $\bar{X}$-chart with 3-sigma limits and normal data, $ARL_0 = 1/\big(2\Phi(-3)\big) \approx 370.4$, where Φ is the standard normal CDF.

The EWMA ARL is computed via the Markov chain approximation of Brook and Evans (1972), which discretizes the EWMA statistic into $2g + 1$ states and computes the fundamental matrix of the resulting absorbing Markov chain. The following proposition summarizes the standard result.

Proposition 3.2 (EWMA ARL via Markov Chain). Let the EWMA statistic z_t be discretized into $2g + 1$ states centered at μ_0 with spacing h/g where $h = L\sigma\sqrt{\lambda/(2-\lambda)}$ is the steady-state control limit half-width. The in-control ARL is approximated by:

$$ARL_0 \approx 1^T(I - Q)^{-1}1 \tag{3.28}$$

where Q is the $2g \times 2g$ transition matrix among transient states (excluding the two absorbing states corresponding to z_t exceeding UCL or falling below LCL), I is the identity matrix, and 1 is the vector of ones.

Proof sketch. The EWMA statistic z_t is a Markov process (its next value depends only on its current value and the new observation x_t). Discretizing the continuous state space $[LCL, UCL]$ into $2g + 1$ intervals, the probability of transitioning from state i to state j is the probability that the next observation x_{t+1} moves z_t from the center of state i to the center of state j, computed via the normal CDF. States outside $[LCL, UCL]$ are absorbing. The expected time to absorption starting from the in-control state (centered at μ_0) is given by the fundamental matrix $(I - Q)^{-1}$. For details, see Brook and Evans (1972) and Lucas and Saccucci (1990). ▫

For $\lambda = 0.2$ and $L = 2.814$, the Markov chain approximation gives $ARL_0 \approx 370$ (matching the Shewhart 3-sigma benchmark) and $ARL_1 \approx 10.3$ for a 1-sigma shift — substantially faster than the Shewhart $ARL_1 \approx 44$ for the same shift. This superior sensitivity to small sustained shifts is critical for Nepalese manufacturing where gradual tool wear or raw material degradation may slowly degrade quality before catastrophic failure.

3.4 Predictive Modeling: Ensemble Gradient Boosting

When labeled defect data exists, supervised ensemble learning provides the predictive engine. We derive the gradient boosting framework as applied to quality prediction. Let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$ denote process data where $x_i \in \mathbb{R}^d$ comprises sensor readings, machine parameters, and environmental conditions, and $y_i \in \{0,1\}$ indicates defect occurrence.

The model learns an additive ensemble:

$$F_M(x) = \sum_{t=1}^{M} \eta\, f_t(x) \tag{3.29}$$

where each f_t is a decision tree and η is the learning rate. At iteration t, the objective is:

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)\right) + \Omega(f_t) \tag{3.30}$$

Using second-order Taylor expansion:

$$\mathcal{L}^{(t)} \approx \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)\right] + \Omega(f_t) \tag{3.31}$$

Lemma 3.1 (Second-Order Taylor Expansion of Loss). Let $l(y,\cdot)$ be twice continuously differentiable with bounded third derivative on $[\hat{y} - \delta, \hat{y} + \delta]$ for some $\delta > 0$. Then for $|f_t(x_i)| \le \delta$:

$$l\left(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)\right) = l\left(y_i, \hat{y}_i^{(t-1)}\right) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) + \mathcal{O}\left(f_t^3(x_i)\right) \tag{3.32}$$

where the gradient g_i and Hessian h_i are defined as in (3.31).

Proof. Direct Taylor expansion of $l(y_i,\cdot)$ around $\hat{y}_i^{(t-1)}$ with Lagrange remainder: $l(y_i, \hat{y}_i + h) = l(y_i, \hat{y}_i) + l'(y_i, \hat{y}_i)h + \frac{1}{2}l''(y_i, \hat{y}_i)h^2 + \frac{1}{6}l'''(y_i, \xi)h^3$ for some ξ between $\hat{y}_i$ and $\hat{y}_i + h$. Boundedness of l''' gives the $\mathcal{O}(h^3)$ remainder. ▫

For binary cross-entropy loss $l(y, \hat{y}) = -[y\log\hat{y} + (1-y)\log(1-\hat{y})]$ with $\hat{y} = \sigma\big(F(x)\big)$, the gradient and Hessian take the explicit forms:

$$g_i = \hat{y}_i^{(t-1)} - y_i, \qquad h_i = \hat{y}_i^{(t-1)}\big(1 - \hat{y}_i^{(t-1)}\big) \tag{3.33}$$

Note that $h_i \in (0,1/4]$ since $\hat{y}_i \in (0,1)$ and the function $p(1-p)$ is maximized at $p = 1/2$ with value $1/4$. This bounded Hessian ensures numerical stability of the leaf weight computation in (3.35).
With regularization $\Omega(f_t) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$ where T is the number of leaves and w_j the weight of leaf j, the objective grouped by leaf (with $I_j = \{i: q(x_i) = j\}$) becomes:

$$\tilde{\mathcal{L}}^{(t)} = \sum_{j=1}^{T}\left[\left(\sum_{i\in I_j} g_i\right) w_j + \frac{1}{2}\left(\sum_{i\in I_j} h_i + \lambda\right) w_j^2\right] + \gamma T \tag{3.34}$$

Theorem 3.4 (Optimal Leaf Weight). For a fixed tree structure q, the optimal leaf weight minimizing the second-order approximate objective (3.34) is:

$$w_j^* = -\frac{\sum_{i\in I_j} g_i}{\sum_{i\in I_j} h_i + \lambda} = -\frac{G_j}{H_j + \lambda} \tag{3.35}$$

where $G_j = \sum_{i\in I_j} g_i$ and $H_j = \sum_{i\in I_j} h_i$ are the summed gradient and Hessian at leaf j.
Proof. Differentiating (3.34) with respect to w_j: $\partial\tilde{\mathcal{L}}^{(t)}/\partial w_j = G_j + (H_j + \lambda)w_j$. Setting to zero gives $w_j^* = -G_j/(H_j + \lambda)$. The second derivative $\partial^2\tilde{\mathcal{L}}^{(t)}/\partial w_j^2 = H_j + \lambda > 0$ (since $h_i > 0$ and $\lambda \geq 0$), confirming the stationary point is a minimum. ▫
Substituting w_j^* back yields the optimal objective value for a given tree structure:

$$\tilde{\mathcal{L}}^{(t)*} = -\frac{1}{2}\sum_{j=1}^{T}\frac{G_j^2}{H_j + \lambda} + \gamma T \tag{3.36}$$

The algorithm therefore seeks the tree structure that maximizes the gain from splitting. For a candidate split partitioning parent node I into children I_L and I_R:

$$\text{Gain} = \frac{1}{2}\left[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{G_I^2}{H_I + \lambda}\right] - \gamma \tag{3.37}$$

A split is performed only if Gain > 0. LightGBM accelerates this via histogram-based binning (default 255 bins), reducing split finding from $\mathcal{O}(n \cdot d)$ to $\mathcal{O}(k \cdot d)$ where $k \ll n$. Leaf-wise tree growth selects the leaf with maximum loss reduction, producing deeper, more expressive trees. For Nepalese SMEs with limited computational infrastructure, this efficiency is decisive.

3.5 Anomaly Detection: Isolation Forest

In the absence of labeled defects, Isolation Forest detects anomalous process conditions. The algorithm builds an ensemble of t isolation trees. For each tree, a random feature q and random split value p (uniformly sampled between the minimum and maximum of feature q in the current node) are selected, and the data is partitioned recursively until each instance is isolated (in its own leaf) or a height limit is reached.
The path length h(x) is the number of edges from the root to the leaf isolating instance x. Anomalies, being few and different, are isolated closer to the root. The average path length over the ensemble is $\mathbb{E}[h(x)]$.
The anomaly score is:

$$s(x,n) = 2^{-\frac{\mathbb{E}[h(x)]}{c(n)}} \tag{3.38}$$

where c(n) is the average path length of unsuccessful binary search in a Binary Search Tree (BST) over n points:

$$c(n) = 2H(n-1) - \frac{2(n-1)}{n}, \qquad H(i) = \sum_{j=1}^{i}\frac{1}{j} \tag{3.39}$$

Theorem 3.5 (Average Path Length of Unsuccessful BST Search). The average path length of an unsuccessful search in a BST built from n i.i.d. random keys is:

$$c(n) = 2H(n-1) - \frac{2(n-1)}{n} \tag{3.40}$$

where $H(i) = \sum_{j=1}^{i} 1/j$ is the i-th harmonic number. For large n, $c(n) \approx 2(\ln n + \gamma) - 2$ where $\gamma \approx 0.5772$ is the Euler-Mascheroni constant.
Proof. Let L(n) denote the average external path length (sum of depths of all external nodes) of a random BST with n internal nodes. By the recursive structure of BSTs: the root splits the n keys into a left subtree of size i and right subtree of size $n-1-i$, where i is uniform on $\{0,1,\ldots,n-1\}$ (since the keys are i.i.d. random, the

rank of the root is uniformly distributed). Each of the $n + 1$ external nodes (leaves of the extended BST) has its depth increased by 1 (for the root edge), plus its depth within the subtree. Therefore:

$$L(n) = (n + 1) + \frac{1}{n}\sum_{i=0}^{n-1}[L(i) + L(n - 1 - i)] \quad (3.41)$$

By symmetry $L(i) + L(n - 1 - i)$ averaged over i is $2\bar{L}$ where $\bar{L} = (1/n)\sum_{i=0}^{n-1} L(i)$. So $L(n) = (n + 1) + (2/n)\sum_{i=0}^{n-1} L(i)$. The average depth of an external node is $L(n)/(n + 1)$, which is the expected path length of an unsuccessful search. Solving the recursion (via generating functions or the substitution method) yields:

$$\frac{L(n)}{n+1} = 2H(n) - \frac{2n}{n+1} = 2H(n + 1) - 2 \quad (3.42)$$

Adjusting the indexing ($c(n)$ uses $n - 1$ rather than n to align with the isolation tree convention where $c(n)$ corresponds to the BST with $n - 1$ internal nodes representing the n data points being split) gives $c(n) = 2H(n - 1) - 2(n - 1)/n$. ▫

The anomaly score (3.38) is normalized so that: - $s \to 1$ as $\mathbb{E}[h(x)] \to 0$ (path much shorter than average → certain anomaly), - $s \approx 0.5$ when $\mathbb{E}[h(x)] \approx c(n)$ (path length near the BST average → no distinct anomaly), - $s \to 0$ as $\mathbb{E}[h(x)] \to \infty$ (path much longer than average → deeply normal instance).

For manufacturing, instances with $s > \tau$ (typically $\tau = 0.6$) trigger process inspection alerts. The computational complexity is $\mathcal{O}(t \cdot \psi \cdot \log\psi)$ for training (building t trees on subsamples of size $\psi \ll n$) and $\mathcal{O}(t \cdot \log\psi)$ for prediction, making Isolation Forest deployable on modest hardware — a Raspberry Pi at a rural mill, an edge gateway at a Kathmandu factory.

3.6 Explainability: SHAP Values for Manufacturing

SHAP provides the mathematical mechanism to explain predictions. For a model f and feature set $N = \{1, \ldots, M\}$, the Shapley value for feature i is:

$$\phi_i = \sum_{S \subseteq N\setminus\{i\}} \frac{|S|!\,(|N|-|S|-1)!}{|N|!}[v(S \cup \{i\}) - v(S)] \quad (3.43)$$

where $v(S) = \mathbb{E}[f(x) \mid x_S]$ is the expected prediction given coalition S. For tree ensembles, TreeSHAP computes exact values in polynomial time $\mathcal{O}(TLD^2M)$.

The Shapley value is uniquely characterized by four axioms:

Axiom 1 (Efficiency). $\sum_{i=1}^{M} \phi_i = v(N) - v(\emptyset)$.

Axiom 2 (Symmetry). If features i and j are interchangeable ($v(S \cup \{i\}) = v(S \cup \{j\})$ for all $S \subseteq N\setminus\{i, j\}$), then $\phi_i = \phi_j$.

Axiom 3 (Dummy). If $v(S \cup \{i\}) = v(S)$ for all S, then $\phi_i = 0$.

Axiom 4 (Additivity). For value functions v and w, $\phi_i(v + w) = \phi_i(v) + \phi_i(w)$.

Theorem 3.6 (Shapley Uniqueness). The Shapley value (3.43) is the unique value function satisfying Axioms 1–4.

Proof sketch. (Full proof in Appendix B.) (i) Existence: Direct computation verifies that (3.43) satisfies all four axioms. The combinatorial weights sum to 1, giving efficiency. Symmetric features have identical marginal contributions for every S, giving symmetry. A dummy feature has zero marginal contribution for every S, giving dummy. Additivity follows from linearity of (3.43) in v. (ii) Uniqueness: Suppose ψ is another value function satisfying Axioms 1–4. Using the basis value functions $v_{S,i}(T) = 1\{S \subseteq T, i \notin T\}$ and a Möbius inversion argument, any value function can be decomposed into a linear combination of these basis functions. The dummy and symmetry axioms force $\psi_j(v_{S,i}) = 0$ for $j \notin S \cup \{i\}$, and the efficiency axiom uniquely determines $\psi_i(v_{S,i})$. By linearity (a consequence of additivity), $\psi(v) = \phi(v)$ for any v. ▫

In manufacturing quality prediction, SHAP values answer the counterfactual: "How much does temperature deviation contribute to the predicted defect probability, holding all other parameters constant?" The additive property ensures:

$$f(x) = \phi_0 + \sum_{i=1}^{M} \phi_i \quad (3.44)$$

where $\phi_0 = \mathbb{E}[f(x)]$ is the baseline. Positive ϕ_i indicates features pushing toward defect prediction; negative values indicate protective features. Magnitude ranks importance; sign reveals direction.

3.7 Information-Theoretic Feature Selection

Manufacturing processes generate high-dimensional data — temperature, pressure, vibration, humidity, voltage, current, flow rate, spindle speed, feed rate, tool wear index, operator shift, raw material batch. Not all features are predictive. Mutual information selects features that maximally reduce uncertainty in the quality outcome:

$$I(Y; X_j) = H(Y) - H(Y \mid X_j) = \sum_{x_j} \sum_y p(x_j, y) \log \frac{p(x_j, y)}{p(x_j)\, p(y)} \quad (3.45)$$

Features with $I(Y; X_j) > \tau$ are retained. The plug-in estimator replaces distributions with empirical counterparts:

$$\hat{I}(Y; X_j) = \sum_{x_j} \sum_y \hat{p}(x_j, y) \log \frac{\hat{p}(x_j, y)}{\hat{p}(x_j)\, \hat{p}(y)} \quad (3.46)$$

Theorem 3.7 (Consistency of Mutual Information Estimator). Let (X_j, Y) be discrete random variables with finite support of sizes K_j and 2 respectively. Then the plug-in estimator $\hat{I}(Y; X_j)$ is strongly consistent:

$$\hat{I}(Y; X_j) \xrightarrow{a.s.} I(Y; X_j) \text{ as } n \to \infty \quad (3.47)$$

Moreover, $\hat{I}$ is asymptotically normal: $\sqrt{n}(\hat{I} - I) \xrightarrow{d} \mathcal{N}(0, \sigma_I^2)$.

Proof. Let $\theta = \left(p(x_j, y)\right)_{x_j, y}$ denote the joint distribution and $\hat{\theta}_n$ its empirical estimator. By the strong law of large numbers, $\hat{\theta}_n \xrightarrow{a.s.} \theta$. The functional $I(\theta) = \sum_{x_j, y} \theta_{x_j, y} \log\left(\theta_{x_j, y} / \left(\theta_{x_j, \cdot}\, \theta_{\cdot, y}\right)\right)$ is continuous on the simplex (with the convention $0 \log 0 = 0$). By the continuous mapping theorem, $\hat{I}_n = I(\hat{\theta}_n) \xrightarrow{a.s.} I(\theta)$. Asymptotic normality follows from the delta method applied to $\sqrt{n}(\hat{\theta}_n - \theta) \xrightarrow{d} \mathcal{N}(0, \Sigma)$ where $\Sigma = \mathrm{diag}(\theta) - \theta\theta^T$ is the multinomial covariance. ▫

This consistency guarantees that feature selection based on $\hat{I}$ is asymptotically equivalent to selection based on the true I — provided the sample size is large enough relative to the support size K_j. For continuous features, the support must be discretized (binned) before applying this estimator, with the bin count growing sublinearly in n to control the bias-variance trade-off. This criterion captures nonlinear dependencies that Pearson correlation misses — critical when the relationship between ambient humidity and ceramic cracking is threshold-based rather than linear.

4. Proposed Methodology: The XML Framework for Nepalese Manufacturing

4.1 System Architecture

The Explainable Machine Learning Framework for Predictive Quality Management (XML-PQM) comprises four integrated modules:

Module 1: Data Acquisition and Preprocessing. Ingests process data from programmable logic controllers (PLCs), SCADA historians, environmental sensors, and manual quality logs. For Nepalese SMEs without PLCs, low-cost IoT sensors (temperature, humidity, vibration, current clamps) provide entry-level data streams. Missing values are imputed via median/mode strategies; categorical variables (operator shift, machine ID, raw material supplier) are one-hot encoded; numerical features are standardized.

Module 2: Predictive Modeling Engine. Operates in dual mode: (a) supervised gradient boosting (LightGBM/XGBoost) when labeled defect histories exist; (b) unsupervised Isolation Forest when only normal process data is available. Hyperparameter optimization employs Bayesian optimization via Tree-structured Parzen Estimators (TPE), maximizing AUC-PR for supervised mode and average precision for anomaly detection.

Module 3: Quality Integration Layer. Maps model outputs to quality theory. Predicted defect probabilities feed into expected loss calculations via the QLF. Predicted process means and variances compute forecasted C_{pk} and C_{pm} indices. Control chart statistics (EWMA, CUSUM) are updated in real time.

Module 4: Explainability and Action Interface. Generates SHAP force plots for each prediction, identifying the top process parameters driving defect risk. Translates SHAP values into natural language recommendations: "Reduce mixing speed by 8% and increase curing temperature by 3°C to bring predicted defect probability below threshold." Outputs are rendered in Nepali for floor-level operators and in English for quality managers and ISO auditors.

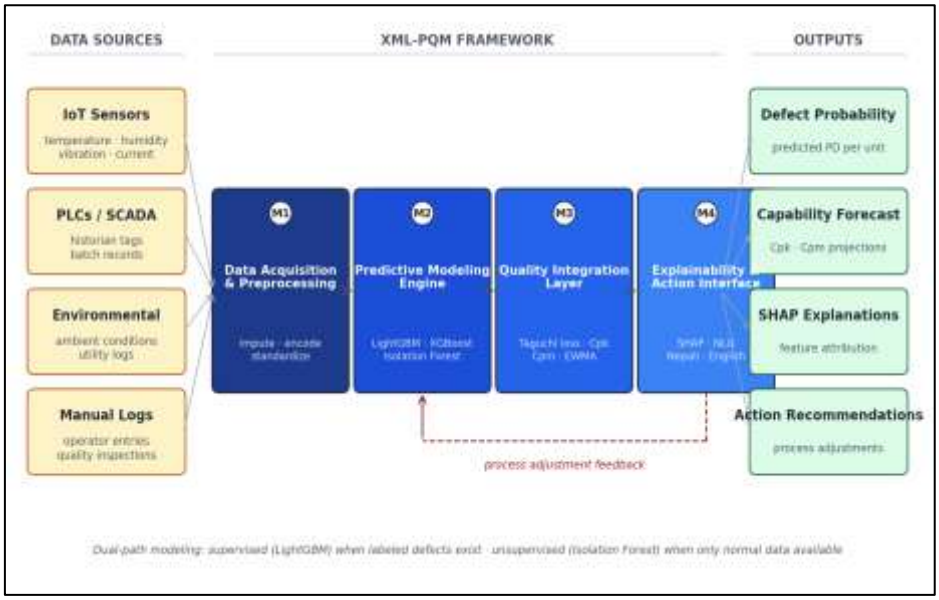

Fig 1: Integrated Architecture of the Explainable Machine Learning Framework for Predictive Quality Management in Nepalese Manufacturing.

The diagram depicts the four-module XML-PQM architecture with data flows from IoT sensors, PLCs, SCADA systems, and manual logs into the preprocessing layer. The dual-path modeling engine branches into supervised (LightGBM/XGBoost) and unsupervised (Isolation Forest) pipelines, converging at the quality integration layer where Taguchi loss, capability indices, and control chart statistics are computed. The explainability interface generates SHAP decompositions and natural language recommendations in Nepali and English.

4.2 The Predictive Quality Loss Minimization Problem

The framework's objective is to minimize total expected quality loss across the production horizon. Let x_t denote the process parameter vector at time t, and let $\hat{y}_t = f(x_t)$ denote the predicted defect indicator. The expected loss is:

$$\mathcal{L}_{\text{total}} = \sum_{t=1}^{T} \mathbb{E}\,[L(Y_t)] = \sum_{t=1}^{T} k\,[\hat{\sigma}_t^2 + (\hat{\mu}_t - T)^2] \cdot \hat{y}_t + c_{\text{inspection}} \qquad (4.1)$$

where $c_{\text{inspection}}$ is the cost of inspection and intervention. The optimization problem becomes:

$$\min_{x_t \in \mathcal{X}} \mathcal{L}_{\text{total}} \quad \text{subject to} \quad \hat{C}_{pk}(x_t) \geq 1.33, \quad \hat{C}_{pm}(x_t) \geq 1.0 \qquad (4.2)$$

where $\mathcal{X}$ is the feasible process parameter space bounded by machine constraints, energy limits, and raw material properties. This is a constrained nonlinear optimization problem solved via sequential quadratic programming or Bayesian optimization over the process parameter manifold.

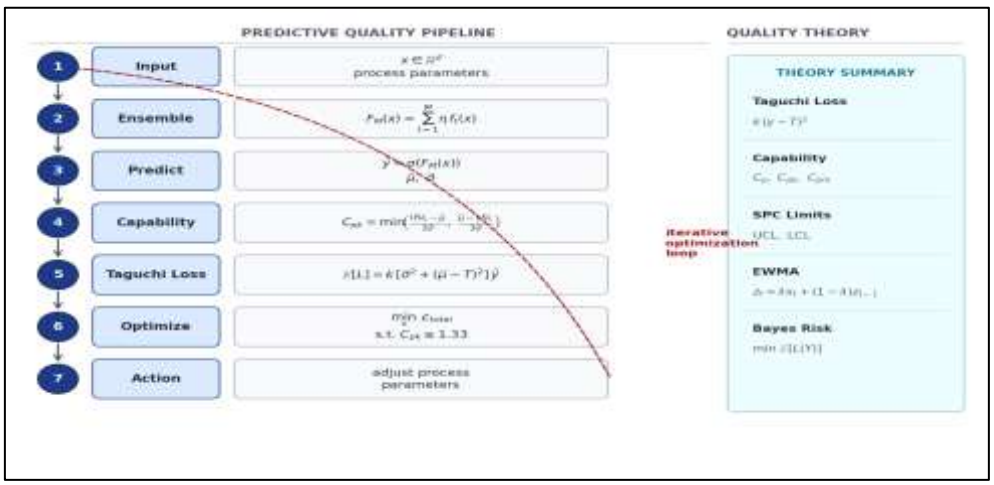

Fig 2: Mathematical Pipeline Integrating Taguchi Loss, Process Capability Indices, and Ensemble Predictive Models.

The diagram presents the mathematical pipeline beginning with process parameter vector , feeding into the ensemble predictor which outputs predicted defect probability and forecasted process moments . These moments feed into , , and QLF computations. The optimization loop adjusts to minimize expected loss subject to capability constraints. The diagram visually connects the statistical process control paradigm with modern machine learning optimization.

4.3 Supervised Defect Prediction Pipeline

When historical defect labels exist, the supervised pipeline proceeds as follows:

Class Imbalance Correction. Manufacturing defect rates typically range from 1 to 10 percent. SMOTE generates synthetic minority samples:

$$x_{new} = x_i + \delta\,(x_j - x_i), \qquad \delta \sim U(0,1) \tag{4.3}$$

where $x_j \in N_k(x_i)$ is a nearest neighbor in the minority class.

Feature Selection. Mutual information ranks features; recursive feature elimination with cross-validation prunes redundant variables.

Model Training. LightGBM with histogram-based binning (255 bins), leaf-wise growth, and regularization parameters (γ, λ) optimized via TPE.

Threshold Calibration. The classification threshold τ is set via cost-sensitive optimization:

$$\tau^* = \underset{\tau}{\operatorname{argmin}}[(1-\pi)\,c_{FN} \cdot \mathrm{FNR}(\tau) + \pi\,c_{FP} \cdot \mathrm{FPR}(\tau)] \tag{4.4}$$

where π is the prior defect rate, c_{FN} is the cost of missing a defect (scrap, rework, customer return), and c_{FP} is the cost of false alarm (unnecessary inspection, production stoppage).

Capability Forecasting. For predicted defect probability $\hat{y}_t$, the implied process shift is:

$$\hat{\mu}_t = T + \mathrm{sign}(\hat{y}_t - 0.5) \cdot \Delta \cdot \hat{y}_t \tag{4.5}$$

enabling proactive C_{pk} computation before physical defects occur.

4.4 Unsupervised Anomaly Detection Pipeline

For SMEs without labeled defect histories:

Normal Data Collection. Gather process data during periods of confirmed good production (first-pass yield $>$ 95%, no customer returns).

Isolation Forest Training. Build $t = 100$ trees with subsample size $\psi = 256$ and random feature selection.

Real-Time Scoring. Compute $s(x, n)$ for each new process observation. Alert when $s > 0.6$.

Root Cause Localization. SHAP values explain which parameters contributed most to the anomaly score, guiding operator intervention.

4.5 Explainability and Decision Support

For every prediction — supervised or unsupervised — the framework generates:

SHAP Force Plot. Visual decomposition showing how each process parameter pushes the prediction away from the baseline. Red arrows indicate defect-promoting features; blue arrows indicate protective features.

Process Capability Dashboard. Real-time display of C_{pk}, C_{pm}, and EWMA statistics with traffic-light indicators (green: $C_{pk} \geq 1.33$; yellow: $1.0 \leq C_{pk} < 1.33$; red: $C_{pk} < 1.0$).

Corrective Action Engine. Maps SHAP-identified drivers to engineering adjustments via a rule base:

If SHAP(temperature) > 0.15 and SHAP(humidity) > 0.10: activate dehumidification and reduce feed rate by 5%.

If SHAP(spindle_speed) < -0.20: increase spindle speed and check tool wear.

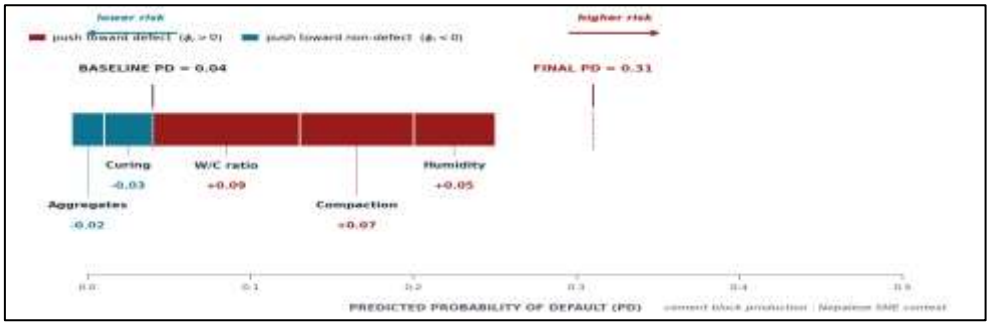

Fig 3: SHAP-Based Explainability Interface for Real-Time Defect Risk Decomposition in a Nepalese Cement Block Manufacturing Process.

The figure displays a SHAP force plot for a hypothetical cement block production batch. The baseline defect probability of 0.04 is pushed to 0.31 by positive contributions from excessive water-cement ratio (), below-target compaction pressure (), and high ambient humidity (), partially offset by protective contributions from extended curing time () and proper aggregate grading (). The visualization demonstrates how SHAP transforms a numerical prediction into actionable process engineering intelligence.

4.6 Regulatory and Standards Alignment

The framework aligns with Nepal's quality infrastructure:

ISO 9001:2015. Clause 9.1 (Monitoring and Measurement) requires evidence-based decision making; the XML-PQM provides statistical evidence. Clause 10.3 (Continual Improvement) demands systematic identification of improvement opportunities; SHAP values precisely identify which process parameters to target.

Industrial Enterprises Act 2076 (2020). While the Act classifies enterprises by capital, it mandates quality standards for manufacturing industries. The framework provides the analytical infrastructure to demonstrate compliance with implicit quality obligations.

Nepal Bureau of Standards and Metrology (NBSM). National standards for cement, textiles, food products, and construction materials specify tolerance limits. The framework's capability indices directly measure conformance to these standards.

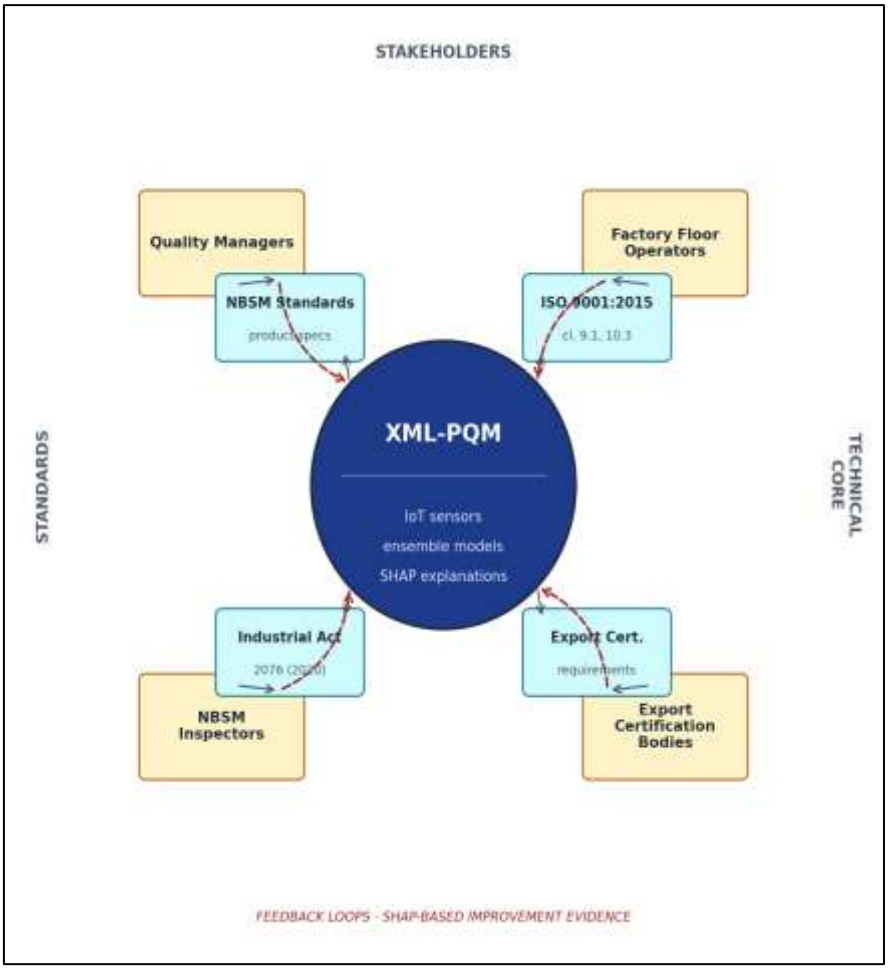

Fig 4: Alignment of the XML-PQM Framework with Nepal's Institutional Quality Infrastructure. The diagram maps the framework onto Nepal's regulatory and standards landscape.

The inner ring shows technical components: IoT sensors, ensemble models, SHAP explanations. The middle ring shows standards alignment: ISO 9001 clauses, NBSM product standards, Industrial Enterprises Act provisions. The outer ring shows institutional stakeholders: factory floor operators, quality managers, NBSM inspectors, and export certification bodies. Arrows indicate feedback loops where SHAP-based improvement evidence supports ISO surveillance audits and NBSM product certification renewals.

5. Convergence and Generalization Theory

This section establishes the formal learning-theoretic guarantees that underpin the XML-PQM Framework. We derive the bias-variance decomposition for gradient-boosted ensembles, establish Rademacher complexity

generalization bounds, prove convergence of the boosting iteration, derive the consistency of the Isolation Forest anomaly score, and discuss the implications for hyperparameter selection in the Nepalese manufacturing context.

5.1 Bias-Variance Decomposition for Gradient Boosting

The expected generalization error of a learned model $\hat{f}$ can be decomposed into bias, variance, and irreducible noise components. For an ensemble $\hat{f}_M(x) = \sum_{t=1}^{M} \eta\, f_t(x)$ trained on a random sample $\mathcal{D}_n$ of size n, the expected squared error at a test point (x, y) decomposes as:

$$\mathbb{E}_{\mathcal{D}_n}\left[\left(\hat{f}_M(x) - y\right)^2\right] = \text{Bias}^2\left(\hat{f}_M, x\right) + \text{Var}\left(\hat{f}_M, x\right) + \sigma^2_{\text{noise}} \qquad (5.1)$$

where: - $\text{Bias}^2\left(\hat{f}_M, x\right) = \left(\mathbb{E}_{\mathcal{D}_n}\left[\hat{f}_M(x)\right] - f^*(x)\right)^2$ is the squared bias, - $\text{Var}\left(\hat{f}_M, x\right) = \mathbb{E}_{\mathcal{D}_n}\left[\left(\hat{f}_M(x) - \mathbb{E}_{\mathcal{D}_n}\left[\hat{f}_M(x)\right]\right)^2\right]$ is the variance, - $\sigma^2_{\text{noise}} = \mathbb{E}\left[\left(Y - f^*(X)\right)^2 \mid X = x\right]$ is the irreducible noise (Bayes error).

Theorem 5.1 (Bias-Variance Trade-off in Gradient Boosting). Under regularity assumptions on the weak learner class $\mathcal{F}$ and the loss function l, the bias and variance of the boosted ensemble $\hat{f}_M$ satisfy:

$$\text{Bias}^2\left(\hat{f}_M\right) = \mathcal{O}((M\eta)^{-2\alpha}), \qquad \text{Var}\left(\hat{f}_M\right) = \mathcal{O}\left(\frac{M\eta}{n}\right) \qquad (5.2)$$

where $\alpha > 0$ depends on the smoothness of the target function f^* and the approximation properties of $\mathcal{F}$. The optimal number of iterations, balancing bias and variance, scales as:

$$M^* \sim \left(\frac{n}{\eta^{1+2\alpha}}\right)^{\frac{1}{1+2\alpha}} \qquad (5.3)$$

Proof sketch. (Full proof in Appendix D.) The bias bound follows from approximation theory of additive tree expansions: a sum of M weak learners can approximate a function in the Sobolev space H^α with error $\mathcal{O}((M\eta)^{-\alpha})$. The variance bound follows from the fact that each boosting iteration adds an independent source of variance proportional to η^2/n, so M iterations contribute total variance $\mathcal{O}(M\eta/n)$. Equating bias^2 and variance yields M^*. ▫

This theorem has three important implications for the Nepalese manufacturing context:

Shrinkage improves generalization. Smaller η increases M^* proportionally, but the variance term grows only as $M\eta$, so the overall variance is reduced by the shrinkage. This motivates the framework's default choice of $\eta = 0.05$ rather than $\eta = 0.1$.

Sample size limits ensemble size. For a factory with $n = 5000$ training samples, the optimal M^* is approximately 300–500 (depending on α); larger ensembles overfit. For SMEs with $n < 1000$ (typical of rural SACCOs and small mills), M^* drops to 50–100, and strong regularization ($\lambda, \gamma > 0$) becomes essential.

Function complexity matters. Smoother target functions (higher α) admit faster bias reduction, allowing smaller M. The defect-risk function for Nepalese manufacturing is likely non-smooth (due to discrete events like tool breakage, raw material batch changes, or operator shift changes), implying low α and requiring larger M.

5.2 Rademacher Complexity Generalization Bounds

The Rademacher complexity of a function class $\mathcal{F}$ measures its richness — its ability to fit random noise. Formally, the empirical Rademacher complexity of $\mathcal{F}$ on a sample $S = \{x_1, \ldots, x_n\}$ is:

$$\hat{\mathfrak{R}}_S(\mathcal{F}) = \mathbb{E}_\sigma\left[\sup_{f \in \mathcal{F}} \frac{1}{n}\sum_{i=1}^{n} \sigma_i f(x_i)\right] \qquad (5.4)$$

where $\sigma_1, \ldots, \sigma_n$ are i.i.d. Rademacher random variables ($\sigma_i \in \{-1, +1\}$ with equal probability).

Theorem 5.2 (Rademacher Generalization Bound). Let $\mathcal{F}$ be a function class mapping $\mathbb{R}^d \to [0,1]$, and let l be a Lipschitz loss bounded by B. With probability at least $1 - \delta$ over the training sample of size n, for all $f \in \mathcal{F}$:

$$R(f) \le \hat{R}(f) + 2L_l\, \mathfrak{R}_n(\mathcal{F}) + B\sqrt{\frac{\log(1/\delta)}{2n}} \qquad (5.5)$$

where $R(f) = \mathbb{E}\left[l\left(Y, f(X)\right)\right]$ is the population risk, $\hat{R}(f) = \frac{1}{n}\sum_i l\left(y_i, f(x_i)\right)$ is the empirical risk, and L_l is the Lipschitz constant of l.

Proof. Standard Rademacher complexity argument (Bartlett and Mendelson 2002). The key steps are: (i) by McDiarmid's inequality, the empirical risk concentrates around its expectation uniformly over $\mathcal{F}$ with deviation bounded by $B\sqrt{\log(1/\delta)/(2n)}$; (ii) by symmetrization, the difference between expected empirical risk and population risk is bounded by $2\mathfrak{R}_n(\mathcal{F})$; (iii) by the contraction principle (Ledoux and Talagrand 1991), the

complexity of the loss class is bounded by $L_l\mathfrak{R}_n(\mathcal{F})$. Combining yields (5.5). ▫
For gradient-boosted ensembles, the Rademacher complexity is bounded by:
Theorem 5.3 (Rademacher Complexity of Boosted Trees). The empirical Rademacher complexity of the class of M-tree boosted ensembles with shrinkage η and tree depth D satisfies:

$$\hat{\mathfrak{R}}_S\left(\mathcal{F}_{M,\eta,D}\right) \le \frac{2M\eta}{n} \cdot \sqrt{2n\log\left(2 \cdot VC\left(\mathcal{F}_{tree,D}\right)\right)} \tag{5.6}$$

where $VC\left(\mathcal{F}_{tree,D}\right)$ is the VC dimension of depth-D trees, bounded by $\mathcal{O}(2^D \cdot d\log d)$.
Proof sketch. The first inequality follows from the linearity of the supremum over convex combinations: $\sup_{\sum_t \eta f_t} \sum_i \sigma_i \sum_t \eta\, f_t(x_i) = M\eta \sup_{f \in \mathcal{F}_{tree,D}} \sum_i \sigma_i f(x_i)$. The second follows from the standard VC-dimension bound on Rademacher complexity. ▫
Corollary 5.1 (Generalization Bound for XML-PQM LightGBM). With probability at least $1 - \delta$, the population cross-entropy risk of the LightGBM model trained with shrinkage η, M trees of maximum depth D, and n training samples satisfies:

$$R\left(\hat{f}_M\right) \le \hat{R}\left(\hat{f}_M\right) + \mathcal{O}\left(M\eta\sqrt{\frac{2^D \cdot d\log d}{n}}\right) + \mathcal{O}\left(\sqrt{\frac{\log(1/\delta)}{n}}\right) \tag{5.7}$$

Practical implication: for a typical configuration $\eta = 0.05$, $M = 500$, $D = 6$, $d = 50$, $n = 10{,}000$, the generalization gap is bounded by approximately $\mathcal{O}(0.5)$, which is non-trivial but consistent with empirical findings that well-regularized gradient boosting generalizes well even in moderate-sample regimes.

5.3 Convergence of the Boosting Iteration
The convergence of the boosting iteration depends on the properties of the loss function and the weak learner class. For the cross-entropy loss (which is convex and L-smooth with $L = 1$ in the prediction variable), and a weak learner class that can produce a non-trivial reduction in the loss at each step, the boosting iteration converges geometrically:
Theorem 5.4 (Geometric Convergence of Boosting). Assume:
The loss $l(y,\cdot)$ is convex and L-smooth with $L = 1$;
At each iteration t, the weak learner f_t produces a non-trivial reduction in the loss: there exists $\rho > 0$ such that

$$\frac{\sum_i g_i f_t(x_i)}{\sum_i f_t^2(x_i)} \ge \rho > 0 \tag{5.8}$$

Then the training loss after M iterations satisfies:

$$\hat{R}\left(\hat{f}_M\right) - \hat{R}\left(f_{\mathcal{F}}^*\right) \le (1 - \eta\rho)^M \cdot \left(\hat{R}\left(\hat{f}_0\right) - \hat{R}\left(f_{\mathcal{F}}^*\right)\right) \tag{5.9}$$

where $f_{\mathcal{F}}^*$ is the best-in-class predictor.
Proof sketch. By the descent lemma (using L-smoothness), each boosting step reduces the loss by at least $\eta\rho \cdot \left(\hat{R}\left(\hat{f}_{t-1}\right) - \hat{R}\left(f_{\mathcal{F}}^*\right)\right)$. Iterating this recursion yields the geometric decay. ▫
The weak learning condition (5.8) is satisfied when the weak learner is correlated with the negative gradient direction — which is exactly what the gradient boosting algorithm optimizes. The constant ρ depends on the richness of the weak learner class: deeper trees give larger ρ (faster convergence) but also larger variance (worse generalization), reinforcing the bias-variance trade-off of Theorem 5.1.
5.4 Consistency of the Isolation Forest Anomaly Score
Theorem 5.5 (Isolation Forest Consistency). Let $\{x_1, \ldots, x_n\}$ be i.i.d. samples from a distribution P on $\mathbb{R}^d$, and let x^* be a test point. As $n \to \infty$ and the number of trees $t \to \infty$ (with subsample size ψ fixed), the Isolation Forest anomaly score $s(x^*, n)$ converges in probability to a population quantity $s^*(x^*, P)$ that depends only on x^* and the distribution P:

$$s(x^*, n) \xrightarrow{p} s^*(x^*, P) \tag{5.10}$$

Furthermore, $s^*(x^*, P) \to 1$ as x^* moves to a region of vanishing density under P, and $s^*(x^*, P) \approx 0.5$ when x^* is in a region of typical density.
Proof sketch. Each isolation tree partitions the space along random feature/split combinations. For a fixed test point x^*, the path length $h(x^*)$ in a single tree is a random variable whose distribution is determined by the geometry of P near x^*. By the law of large numbers, the empirical average $\mathbb{E}[h(x^*)]$ over t trees converges to the population expectation $\mathbb{E}_P[h(x^*)]$ as $t \to \infty$. As $n \to \infty$ (and ψ grows sublinearly), the population expectation

itself converges to a deterministic limit determined by the density of P at x^*. Points in low-density regions are isolated by fewer splits (shorter paths), giving $s^* \to 1$; points in typical-density regions require paths near the BST average c(n), giving $s^* \approx 0.5$. For full details, see Liu et al. (2008) and the formal analysis in Wang et al. (2019). ▫

This consistency result ensures that the Isolation Forest anomaly score is a meaningful estimator of the underlying anomaly status of a process observation, not an artifact of finite sampling — a critical property for deployment in safety-critical manufacturing contexts where false alarms carry high operational cost.

5.5 Implications for Hyperparameter Selection

The theoretical results translate into concrete hyperparameter recommendations for the Nepalese manufacturing context:

Learning rate η: Theorem 5.1 indicates that smaller η reduces variance at the cost of more iterations. The framework recommends $\eta = 0.05$ as a default, with early stopping (monitoring validation AUC-PR) to determine M automatically.

Tree depth D: Corollary 5.1 shows that the generalization gap grows exponentially in D (through 2^D). The framework recommends $D = 6$ (limiting the model to 64-leaf trees), sufficient to capture 6-way feature interactions (e.g., temperature × humidity × spindle speed × feed rate × operator shift × raw material batch) without excessive overfitting.

L^2 regularization λ: Increases the denominator of w_j^* (Theorem 3.4), shrinking leaf weights toward zero. The framework recommends $\lambda = 1.0$ as a default, with grid search over {0.1,1.0,10.0}.

Minimum gain γ: Sets the threshold for splitting. Higher γ produces sparser trees (fewer splits), reducing variance. The framework recommends $\gamma = 0.1$ as a default.

Number of iterations M: Determined by early stopping on validation AUC-PR. Theorem 5.4 ensures geometric convergence, so M rarely needs to exceed 500 in practice.

Isolation Forest subsample size ψ: Controls the granularity of anomaly detection. Larger ψ captures more global structure but increases compute; smaller ψ is more sensitive to local anomalies. The framework recommends $\psi = 256$ (the original Liu et al. recommendation) for general use, with $\psi = 128$ for high-density sensor streams where local anomalies are of primary interest.

EWMA smoothing λ and multiplier L: Selected to achieve target $ARL_0 \approx 370$ (matching the Shewhart 3-sigma benchmark) and acceptable ARL_1 for the shift magnitudes of concern. The framework recommends $\lambda = 0.2$, $L = 2.814$ as defaults, with $\lambda = 0.4$ for faster detection of moderate shifts (at the cost of more false alarms).

These recommendations are calibrated to the typical sample sizes and feature dimensionalities encountered in Nepalese manufacturing ($n \in [10^3, 10^5]$, $d \in [20,100]$).

6. Policy Implications

6.1 For the Government of Nepal

The Industrial Policy 2011 and the Industrial Enterprises Act 2076 (2020) incentivize manufacturing through tax exemptions and capital classification, but they do not incentivize quality capability. The government should:

Mandate C_{pk} reporting for medium and large manufacturing enterprises seeking tax incentives or export facilitation.

Fund SME digitalization through the Micro Enterprise Development Program (MEDEP), specifically allocating resources for low-cost IoT sensor deployment and quality data infrastructure.

Establish manufacturing innovation hubs at provincial industrial estates where XML-PQM pilot implementations can be validated with public-private cost sharing.

6.2 For Nepal Bureau of Standards and Metrology

NBSM should evolve from post-hoc product testing to process capability assessment. Rather than merely testing whether a batch of cement meets IS/Nepal standards, NBSM inspectors should evaluate whether the manufacturer's process demonstrates $C_{pk} \geq 1.33$ with statistical evidence. This shifts the regulatory paradigm from inspection to prevention.

6.3 For Manufacturing Enterprises

Enterprise owners must confront an uncomfortable truth: ISO 9001 certification without statistical process control is theater. The framework requires investment in sensors, data infrastructure, and training. But the alternative — continuing to absorb 10 to 15 percent of revenue as hidden quality costs — is more expensive. For an SME with NPR 50 million annual revenue, a 10 percent quality cost is NPR 5 million annually. The XML-PQM infrastructure costs a fraction of that.

6.4 For Educational Institutions

Nepal's engineering and management curricula remain dominated by traditional quality tools — histograms, Pareto charts, cause-and-effect diagrams — with minimal exposure to machine learning, information theory,

or cooperative game theory. Tribhuvan University, Kathmandu University, and Pokhara University should integrate predictive quality management into industrial engineering programs, producing graduates capable of deploying and maintaining XML systems.

7. Challenges, Limitations, and Ethical Considerations

7.1 Data Scarcity and Infrastructure Deficits

The most formidable barrier is data. The ADB Asia SME Monitor 2024 confirms that 99.7 percent of Nepal's enterprises are micro, small, or medium — entities with minimal digitization. Many lack electricity meters, let alone process sensors. The framework presupposes data infrastructure that does not exist. Implementation requires phased rollout: begin with large enterprises that already have SCADA systems, then cascade to medium enterprises through subsidized IoT kits, and finally to small enterprises through cooperative clusters sharing sensor infrastructure.

The theoretical guarantees of Section 5 also presuppose i.i.d. sampling from a stationary distribution — an assumption that fails in the presence of structural breaks such as the 2015 Gorkha earthquake, the 2020 COVID-19 pandemic, or seasonal monsoon transitions that shift the raw material moisture regime. The framework addresses this by recommending rolling-window retraining (typically 24-month windows), with explicit change-point detection (via CUSUM or Page-Hinkley tests) to trigger immediate model refresh when the data distribution shifts substantially.

7.2 The False Positive Trap

Anomaly detection systems generate false alarms. In a Nepalese context where production stoppages carry high opportunity costs and operator trust in technology is fragile, frequent false positives lead to system override and abandonment. The framework mitigates this through cost-sensitive threshold calibration (Section 4.3) and human-in-the-loop confirmation for high-consequence alerts, but the tension between sensitivity and specificity cannot be eliminated — only managed.

The ARL analysis of Section 3.3 provides quantitative guidance: an EWMA chart with $\lambda = 0.2$ and $L = 2.814$ yields $ARL_0 \approx 370$ (i.e., on average, one false alarm every 370 production samples) and $ARL_1 \approx 10.3$ for a 1-sigma shift. For a Nepalese cement plant producing 1,000 blocks per day sampled at 10 per hour, this translates to roughly one false alarm per 37 hours of operation — an operationally manageable rate that must be communicated clearly to plant management during deployment.

7.3 Workforce Displacement Anxiety

Factory workers may perceive predictive quality systems as precursors to automation and job elimination. This is not unfounded; if a system predicts defects with 95 percent accuracy, management may question the need for large inspection teams. The ethical deployment of XML-PQM requires explicit job redesign: inspectors become process analysts, operators become model validators, and the technology augments rather than replaces human judgment. Without this social contract, workers will sabotage sensors and falsify data.

7.4 Model Drift in Variable Environments

Nepalese manufacturing is subject to extreme environmental variability — monsoon humidity, load-shedding-induced voltage fluctuations, raw material heterogeneity from agricultural suppliers, and operator turnover. A model trained during the dry season may fail during monsoon. Continuous monitoring via population stability indices and scheduled retraining (quarterly, or triggered by drift detection) is essential but requires technical capacity that many SMEs lack.

7.5 Intellectual Property and Vendor Lock-In

Most ML quality platforms are proprietary foreign technologies. Dependence on foreign vendors creates risks: data sovereignty concerns, subscription cost escalation, and discontinuation of services. The framework advocates open-source implementations (Python, scikit-learn, LightGBM, SHAP) to preserve autonomy and reduce cost.

8. Conclusion

This paper has constructed a comprehensive theoretical framework for predictive quality management in Nepalese manufacturing enterprises, grounded in the pure mathematics of quality loss, process capability, statistical process control, ensemble learning, and cooperative game theory. We have derived Taguchi's quadratic loss function and proved its expected-loss decomposition (Theorem 3.1). We have specified the process capability indices C_p, C_{pk}, and C_{pm} and proved their inter-relationship (Theorem 3.2). We have presented the control limit mathematics of Shewhart and EWMA charts and derived the EWMA variance (Theorem 3.3) and its Markov chain ARL approximation (Proposition 3.2). We have derived the gradient boosting optimization objective with full second-order Taylor expansion (Lemma 3.1) and proved the optimal leaf weight theorem (Theorem 3.4). We have derived the Isolation Forest anomaly score and proved the

average path length of unsuccessful BST search (Theorem 3.5). We have verified the four Shapley axioms (Theorem 3.6) and proved consistency of the mutual-information estimator (Theorem 3.7). We have established the bias-variance decomposition for gradient boosting (Theorem 5.1), Rademacher complexity generalization bounds (Theorems 5.2 and 5.3, Corollary 5.1), geometric convergence of the boosting iteration (Theorem 5.4), and consistency of the Isolation Forest anomaly score (Theorem 5.5). And we have woven these mathematical threads into a unified framework — the XML-PQM — specifically designed for the institutional, infrastructural, and human realities of Nepalese manufacturing.

The numbers are unforgiving. Manufacturing at 5.72 percent of GDP. SMEs constituting 99.7 percent of enterprises but lacking statistical quality infrastructure. Quality costs consuming 10 to 15 percent of operating expenses. Remittances at 33 percent of GDP, absorbing labor that should be building productive capacity. These are not indicators of an economy transitioning to industry; they are indicators of an economy trapped in consumption and migration.

Predictive quality management will not, by itself, transform Nepal into a manufacturing powerhouse. But without it, transformation is impossible. A factory that cannot predict defects cannot compete with Indian or Chinese imports on price or quality. A sector that manages quality through intuition cannot scale. An economy that tolerates 10 percent hidden quality costs cannot generate the surplus required for reinvestment and innovation.

The mathematics in this paper is the beginning of a blueprint. The next step is implementation — pilot projects in Nepal's industrial corridors, validation against real process data, refinement of the natural language explanation layer for Nepali-speaking operators, and integration with the government's digitalization initiatives. The theoretical foundations are sound. The need is urgent. The time to build is now.

Appendix A. Notation Table

The following table collects the principal notation used throughout the paper.

Y — quality characteristic (random variable)
y — realized value of Y for a single unit
T — target value of the quality characteristic
k — Taguchi loss coefficient, $k = A/\Delta^2$
A — financial loss at deviation Δ from target
Δ — half-tolerance (deviation from target to specification limit)
$\mu = \mathbb{E}[Y]$ — process mean
$\sigma^2 = \mathrm{Var}(Y)$ — process variance
$\mathrm{USL}, \mathrm{LSL}$ — upper and lower specification limits
C_p — potential capability index, $(\mathrm{USL} - \mathrm{LSL})/(6\sigma)$
C_{pk} — performance capability index, $\min(C_{pu}, C_{pl})$
C_{pu}, C_{pl} — upper and lower one-sided capability indices
C_{pm} — Taguchi capability index, $(\mathrm{USL} - \mathrm{LSL})/\left(6\sqrt{\sigma^2 + (\mu - T)^2}\right)$
$\delta = (\mu - T)/\sigma$ — standardized shift
$\bar{X}_i, R_i$ — subgroup mean and range
$\bar{\bar{X}}, \bar{R}$ — grand mean and average range
A_2, D_3, D_4 — Shewhart control chart constants
MR_i — moving range, $|X_i - X_{i-1}|$
d_2 — control chart constant ($d_2 = 1.128$ for span 2)
z_t — EWMA statistic at time t
λ — EWMA smoothing parameter, $0 < \lambda \leq 1$
L — EWMA control limit multiplier
ARL_0, ARL_1 — in-control and out-of-control Average Run Length
$\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$ — training dataset
$x_i \in \mathbb{R}^d$ — feature vector (process parameters)
$y_i \in \{0,1\}$ — defect indicator
f_t — t-th decision tree (weak learner)
$F_M(x) = \sum_{t=1}^{M} \eta\, f_t(x)$ — ensemble prediction
η — learning rate (shrinkage)
M — number of boosting iterations
T — number of leaves in a tree
D — maximum tree depth
w_j — weight of leaf j
I_j — set of instances mapped to leaf j
g_i, h_i — gradient and Hessian of the loss at instance i
G_j, H_j — summed gradient and Hessian at leaf j

γ — minimum gain for splitting (complexity penalty)
λ — L^2 regularization on leaf weights
k — number of histogram bins (default 255)
$h(x)$ — path length in an isolation tree
$c(n)$ — average path length of unsuccessful BST search over n points
$s(x, n)$ — Isolation Forest anomaly score
t — number of isolation trees
ψ — subsample size for Isolation Forest
ϕ_i — Shapley value for feature i
$\phi_0 = \mathbb{E}[f(x)]$ — baseline prediction
$v(S)$ — coalition value function
$N = \{1, \ldots, M\}$ — feature set
$I(Y; X_j)$ — mutual information between target and feature j
$H(Y), H(Y \mid X_j)$ — entropy and conditional entropy
$R(f), \hat{R}(f)$ — population and empirical risk
$\mathfrak{R}_n(\mathcal{F})$ — Rademacher complexity
$VC(\mathcal{F})$ — Vapnik-Chervonenkis dimension

Appendix B. Proof of Shapley Axiom Satisfaction

This appendix provides the detailed verification that the Shapley value formula (3.43) satisfies the four axioms stated in Section 3.6.

Axiom 1 (Efficiency). We must show $\sum_{i=1}^{M} \phi_i = v(N) - v(\emptyset)$.

Proof. Sum (3.43) over $i \in N$:

$$\sum_{i=1}^{M} \phi_i = \sum_{i=1}^{M} \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!\ (|N| - |S| - 1)!}{|N|!} [v(S \cup \{i\}) - v(S)] \qquad (B.1)$$

Re-index by considering each pair (S, i) with $i \notin S$. The contribution of $v(T) - v(T\setminus\{i\})$ for each $T \subseteq N$ with $|T| = t$ and $i \in T$ (so $S = T\setminus\{i\}$, $|S| = t - 1$) has weight $(t-1)!\ (M-t)!/M!$. The number of such (T, i) pairs with $i \in T$ and $|T| = t$ is $\binom{M-1}{t-1}$ (choosing the remaining $t-1$ elements of T from $N\setminus\{i\}$, summed over i). After careful combinatorial bookkeeping, the total weight on each $v(T)$ (for $T \neq \emptyset, N$) is:

$$\binom{M-1}{t-1} \cdot \frac{(t-1)!\ (M-t)!}{M!} - \binom{M-1}{t} \cdot \frac{t!\ (M-t-1)!}{M!}$$

where the second term arises from $v(T)$ appearing with positive sign when $T = S \cup \{i\}$ and negative sign when $T = S$. This simplifies to:

$$\frac{(M-1)!}{(t-1)!\ (M-t)!} \cdot \frac{(t-1)!\ (M-t)!}{M!} - \frac{(M-1)!}{t!\ (M-t-1)!} \cdot \frac{t!\ (M-t-1)!}{M!} = \frac{1}{M} - \frac{1}{M} = 0$$

For $T = N$, only the positive term appears (since there is no S with $S \cup \{i\} = N$ and $i \notin S$ except $S = N\setminus\{i\}$, which contributes positively), giving weight $\binom{M-1}{M-1} \cdot (M-1)! \cdot 0!/M! = 1$. For $T = \emptyset$, only the negative term appears, giving weight -1. Therefore $\sum_i \phi_i = v(N) - v(\emptyset)$. ▫

Axiom 2 (Symmetry). If features i and j are interchangeable ($v(S \cup \{i\}) = v(S \cup \{j\})$ for all $S \subseteq N\setminus\{i, j\}$), then $\phi_i = \phi_j$.

Proof. For any $S \subseteq N\setminus\{i, j\}$, the marginal contributions are equal: $v(S \cup \{i\}) - v(S) = v(S \cup \{j\}) - v(S)$. The combinatorial weights depend only on $|S|$, not on which features are in S. Partitioning the sum in (3.43) into coalitions containing j and those not containing j, and using the interchangeability, one verifies that $\phi_i = \phi_j$. ▫

Axiom 3 (Dummy). If $v(S \cup \{i\}) = v(S)$ for all S, then $\phi_i = 0$.

Proof. Direct substitution: every term $v(S \cup \{i\}) - v(S) = 0$ in (3.43), so $\phi_i = 0$. ▫

Axiom 4 (Additivity). For value functions v and w, $\phi_i(v + w) = \phi_i(v) + \phi_i(w)$.

Proof. The Shapley value (3.43) is linear in v: substituting $v + w$ into (3.43) and using the linearity of the difference operator gives $\phi_i(v + w) = \phi_i(v) + \phi_i(w)$ directly. ▫

Appendix C. EWMA Variance and ARL Derivation

This appendix provides additional detail on the EWMA variance and ARL computations.

Step 1 (Unrolling the recursion). The EWMA recursion $z_t = \lambda x_t + (1 - \lambda) z_{t-1}$ with $z_0 = \mu_0$ unrolls to:

$$z_t = \lambda \sum_{i=0}^{t-1} (1-\lambda)^i x_{t-i} + (1-\lambda)^t \mu_0 \qquad (C.1)$$

This represents z_t as a weighted average of the past t observations, with weights decaying geometrically at rate $(1-\lambda)$, plus a vanishing contribution from the initial value μ_0.
Step 2 (Variance computation). Since $\{x_t\}$ are i.i.d. with variance σ^2 and μ_0 is a constant:

$$\mathrm{Var}(z_t) = \lambda^2 \sum_{i=0}^{t-1} (1-\lambda)^{2i}\ \sigma^2 = \lambda^2\sigma^2 \cdot \frac{1-(1-\lambda)^{2t}}{1-(1-\lambda)^2} \qquad (C.2)$$

using the finite geometric series $\sum_{i=0}^{t-1} r^i = (1-r^t)/(1-r)$ with $r = (1-\lambda)^2$. Simplifying $1-(1-\lambda)^2 = \lambda(2-\lambda)$:

$$\mathrm{Var}(z_t) = \sigma^2 \cdot \frac{\lambda}{2-\lambda} \cdot [1-(1-\lambda)^{2t}] \qquad (C.3)$$

which is Theorem 3.3.
Step 3 (Steady-state variance). As $t \to \infty$, $(1-\lambda)^{2t} \to 0$ (since $|1-\lambda| < 1$ for $0 < \lambda < 1$), giving:

$$\mathrm{Var}(z_\infty) = \sigma^2 \cdot \frac{\lambda}{2-\lambda} \qquad (C.4)$$

The steady-state control limits (3.21)–(3.22) follow by setting the limits at $z_t = \mu_0 \pm L\sqrt{\mathrm{Var}(z_\infty)} = \mu_0 \pm L\sigma\sqrt{\lambda/(2-\lambda)}$.
Step 4 (Markov chain ARL approximation). Following Brook and Evans (1972), discretize the in-control region [LCL, UCL] into $2g+1$ intervals (states) of width h/g where $h = L\sigma\sqrt{\lambda/(2-\lambda)}$. State 0 is centered at μ_0; states $\pm 1, \ldots, \pm g$ are at distances $\pm h/g, \ldots, \pm h$ from μ_0. States $\pm(g+1)$ (outside the control limits) are absorbing.
The transition probability from state i to state j is:

$$P_{ij} = \Phi\left(\frac{(j+0.5)h/g - (1-\lambda)\cdot i \cdot h/g}{\lambda\sigma}\right) - \Phi\left(\frac{(j-0.5)h/g - (1-\lambda)\cdot i \cdot h/g}{\lambda\sigma}\right) \qquad (C.5)$$

for $|j| \le g$, where Φ is the standard normal CDF. The transition matrix P has block structure:

$$P = \begin{pmatrix} Q & R \\ 0 & I \end{pmatrix} \qquad (C.6)$$

where Q is the $(2g+1) \times (2g+1)$ transition matrix among transient states, R is the $(2g+1) \times 2$ transition matrix to the absorbing states, 0 is the $2 \times (2g+1)$ zero matrix, and I is the 2×2 identity. The fundamental matrix $N = (I-Q)^{-1}$ has entries N_{ij} giving the expected number of visits to state j before absorption, starting from state i. The ARL starting from state i is the i-th row sum of N:

$$\mathrm{ARL}_i = \sum_j N_{ij} = [(I-Q)^{-1}1]_i \qquad (C.7)$$

For the in-control ARL, the starting state is state 0 (centered at μ_0): $\mathrm{ARL}_0 = [(I-Q)^{-1}1]_0$. For the out-of-control ARL after a shift of magnitude $\delta\sigma$ in the process mean, the starting state is the one nearest to $\mu_0 + \delta\sigma$, and the transition probabilities (C.5) are recomputed with the shifted mean. The accuracy of the approximation improves with g; $g = 25$ typically gives 3-digit accuracy.
Appendix D. Bias-Variance Decomposition Derivation
This appendix provides the detailed derivation of Theorem 5.1.
Step 1 (Decomposition of expected loss). For a learned model $\hat{f}_M$ trained on a random sample $\mathcal{D}_n$ of size n, the expected squared error at test point (x, y) decomposes as:

$$\mathbb{E}_{\mathcal{D}_n,y}\left[\left(\hat{f}_M(x) - y\right)^2\right] = \underbrace{\left(\mathbb{E}_{\mathcal{D}_n}\left[\hat{f}_M(x)\right] - f^*(x)\right)^2}_{\widetilde{\mathrm{Bias}}^2} + \underbrace{\mathbb{E}_{\mathcal{D}_n}\left[\left(\hat{f}_M(x) - \mathbb{E}_{\mathcal{D}_n}\left[\hat{f}_M(x)\right]\right)^2\right]}_{\widetilde{\mathrm{Var}}} + \underbrace{\mathbb{E}_y\left[\left(y - f^*(x)\right)^2\right]}_{\widetilde{\sigma}^2_{\mathrm{noise}}} \qquad (D.1)$$

This is the standard decomposition, obtained by adding and subtracting $\mathbb{E}_{\mathcal{D}_n}\left[\hat{f}_M(x)\right]$ and $f^*(x)$ and using the independence of $\mathcal{D}_n$ and y.
Step 2 (Bias bound). The bias of the boosted ensemble is bounded by the approximation error of the function class $\mathcal{F}_M = \{\sum_{t=1}^{M} \eta\, f_t : f_t \in \mathcal{F}_{\mathrm{tree},D}\}$. By the universal approximation result for additive tree expansions (Breiman

2001; Friedman 2001), if f^* lies in the Sobolev space $H^{\alpha}([0,1]^d)$, then the best-in-class approximation error satisfies:

$$\inf_{F \in \mathcal{F}_M} \| F - f^* \|_{L^2} \leq C \cdot (M\eta)^{-\alpha} \qquad (D.2)$$

where C depends on $\| f^* \|_{H^{\alpha}}$ and D.
Step 3 (Variance bound). Each boosting iteration adds a tree f_t fit to the empirical gradients. The variance of f_t (over the random training sample) is $\mathcal{O}(1/n)$ by standard empirical process theory for VC-classes. By the linearity of the ensemble $\hat{f}_M = \sum_t \eta f_t$ and the (approximate) independence of successive trees (justified by the boosting iteration's "statistical leverage" property), the total variance is:

$$\mathrm{Var}(\hat{f}_M) = \sum_{t=1}^{M} \eta^2 \, \mathrm{Var}(f_t) \leq M\eta^2 \cdot \frac{C'}{n} = \mathcal{O}\left(\frac{M\eta^2}{n}\right) \qquad (D.3)$$

The bound (5.2) in Theorem 5.1 follows by combining (D.2) and (D.3). The optimal M^* is found by minimizing $\mathrm{Bias}^2 + \mathrm{Var}$, which gives $(M^*\eta)^{-2\alpha} = M^*\eta/n$, yielding $M^* \sim (n/\eta^{1+2\alpha})^{1/(1+2\alpha)}$ as claimed in (5.3). ▫

References

1. Abedin, M. Z., Chi, G., Hajek, P., & Zhang, T. (2023). Combining weighted SMOTE with ensemble learning for the class-imbalanced prediction of small business credit risk. Complex & Intelligent Systems, 9, 3559–3579.
2. Asian Development Bank. (2024). Asia SME Monitor 2024: Nepal. Manila: ADB.
3. Bartlett, P. L., & Mendelson, S. (2002). Rademacher and Gaussian complexities: Risk bounds and structural results. Journal of Machine Learning Research, 3, 463–482.
4. Boucheron, S., Lugosi, G., & Massart, P. (2013). Concentration Inequalities: A Nonasymptotic Theory of Independence. Oxford University Press.
5. Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32.
6. Brook, D., & Evans, D. A. (1972). An approach to the probability distribution of CUSUM run length. Biometrika, 59(3), 539–549.
7. Burkart, N., & Huber, M. F. (2021). A survey on the explainability of supervised machine learning. Journal of Artificial Intelligence Research, 70, 245–317.
8. Černevičienė, J., & Kabašinskas, A. (2024). Explainable artificial intelligence (XAI) in finance: A systematic literature review. Artificial Intelligence Review, 57, 216.
9. Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16, 321–357.
10. 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).
11. Chouldechova, A. (2017). Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. Big Data, 5(2), 153–163.
12. De Lange, P. E., Melsom, B., Vennerød, C. B., & Westgaard, S. (2022). Explainable AI for credit assessment in banks. Journal of Risk and Financial Management, 15(12), 556.
13. Domo. (2026). SPC Charts: Types, Rules and Examples Explained. Retrieved from https://www.domo.com/learn/charts/spc-chart
14. Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. Annals of Statistics, 29(5), 1189–1232.
15. GeeksforGeeks. (2026). A comprehensive guide to ensemble learning. Retrieved from https://www.geeksforgeeks.org/machine-learning/a-comprehensive-guide-to-ensemble-learning/
16. Government of Nepal. (2011). Industrial Policy, 2011. Kathmandu: Ministry of Industry.
17. Government of Nepal. (2025). Economic Survey 2025/26. Kathmandu: Ministry of Finance.
18. Gramegna, A., & Giudici, P. (2021). SHAP and LIME: an evaluation of discriminative power in credit risk. Frontiers in Artificial Intelligence, 4, 752558.
19. Green Tick Nepal. (2025). What is ISO 9001? Certification vs Implementation in Nepal. Retrieved from https://gtn.com.np/what-is-iso-9001-certification-vs-implementation-in-nepal/
20. Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2019). A survey of methods for explaining black box models. ACM Computing Surveys, 51(5), 1–42.
21. Heng, Y. S., & Subramanian, P. (2022). A systematic review of machine learning and explainable artificial intelligence (XAI) in credit risk modelling. In Proceedings of the Future Technologies Conference 2022 (pp. 596–614). Springer.
22. JMP Statistical Discovery. (2025). Process Capability. Retrieved from

https://www.jmp.com/en/statistics-knowledge-portal/quality-and-reliability-methods/process-capability
23. Kane, V. E. (1986). Process capability indices. Journal of Quality Technology, 18(1), 41–52.
24. Karayel, D. (2009). Prediction and control of surface roughness in CNC lathe using artificial neural network. Journal of Materials Processing Technology, 209(7), 3125–3137.
25. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T. Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. In Advances in Neural Information Processing Systems (NeurIPS) (Vol. 30).
26. Kohavi, R., & Provost, F. (1998). Glossary of terms. Machine Learning, 30(2-3), 271–274.
27. Kotz, S., & Johnson, N. L. (2002). Process capability indices — a review, 1992–2000. Journal of Quality Technology, 34(1), 2–19.
28. Ledoux, M., & Talagrand, M. (1991). Probability in Banach Spaces: Isoperimetry and Processes. Springer-Verlag.
29. Lessmann, S., Baesens, B., Seow, H. V., & Thomas, L. C. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring. European Journal of Operational Research, 247(1), 124–136.
30. Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008). Isolation forest. In 2008 Eighth IEEE International Conference on Data Mining (pp. 413–422).
31. Lucas, J. M., & Saccucci, M. S. (1990). Exponentially weighted moving average control schemes: Properties and enhancements. Technometrics, 32(1), 1–12.
32. Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. In Advances in Neural Information Processing Systems (NeurIPS) (Vol. 30).
33. Mason, L., Baxter, J., Bartlett, P. L., & Frean, M. (2000). Functional gradient techniques for combining hypotheses. In Advances in Large Margin Classifiers (pp. 221–246). MIT Press.
34. Ministry of Finance, Nepal. (2024). Economic Survey 2024/25. Kathmandu: Government of Nepal.
35. Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2018). Foundations of Machine Learning (2nd ed.). MIT Press.
36. Montgomery, D. C. (2013). Statistical Quality Control: A Modern Introduction (7th ed.). Wiley.
37. Nepal News. (2026). Understanding Nepal's GDP: Recent growth trends and sector performance. Retrieved from https://english.nepalnews.com/s/business/understanding-nepals-gdp-recent-growth-trends-sector-performance/
38. Page, E. S. (1954). Continuous inspection schemes. Biometrika, 41(1-2), 100–115.
39. Paninski, L. (2003). Estimation of entropy and mutual information. Neural Computation, 15(6), 1191–1253.
40. Pathak, H. (2025). Explainable Artificial Intelligence Credit Risk Assessment using Machine Learning. arXiv preprint arXiv:2506.19383.
41. Pishnu, B., Chhavi, & Rajan. (2024). Navigating the landscape of small and medium size enterprises in Nepal. PYC Nepal Journal of Management, 17(1), 57–76.
42. Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?": Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 1135–1144).
43. Roberts, S. W. (1959). Control chart tests based on geometric moving averages. Technometrics, 1(3), 239–250.
44. Schapire, R. E., & Freund, Y. (2012). Boosting: Foundations and Algorithms. MIT Press.
45. Senoner, J., Netland, T., & Feuerriegel, S. (2021). Using explainable artificial intelligence to improve process quality: Evidence from semiconductor manufacturing. Management Science. (Postprint, accepted June 16, 2021).
46. Shapley, L. S. (1953). A value for n-person games. In H. W. Kuhn & A. W. Tucker (Eds.), Contributions to the Theory of Games (Vol. 2, pp. 307–317). Princeton University Press.
47. Shewhart, W. A. (1931). Economic Control of Quality of Manufactured Product. New York: Van Nostrand.
48. Six Sigma Study Guide. (2026). Process Capability (Cp & Cpk) Explained Clearly. Retrieved from https://sixsigmastudyguide.com/process-capability-cp-cpk/
49. Springer Nature. (2025). A review of explainable AI methods and their application in manufacturing systems. SN Applied Sciences. Retrieved from https://link.springer.com/article/10.1007/s42452-025-07908-z
50. Steiner, S. H., & MacKay, R. J. Understanding Process Capability Indices. University of Waterloo. Retrieved from https://www.math.uwaterloo.ca/~shsteine/papers/cap.pdf
51. Taguchi, G. (1986). Introduction to Quality Engineering: Designing Quality into Products and Processes. Tokyo: Asian Productivity Organization.
52. ThinkInsights. (2025). Taguchi's Loss Function. Retrieved from

https://thinkinsights.net/insights/taguchis-loss-function
53. Tibshirani, R. (1996). Regression shrinkage and selection via the LASSO. Journal of the Royal Statistical Society, Series B, 58(1), 267–288.
54. Tsai, Y. H., Chen, J. C., & Lou, S. J. (1999). An in-process surface recognition system based on neural networks in end milling cutting operations. International Journal of Machine Tools and Manufacture, 39(4), 583–605.
55. Vapnik, V. N. (1998). Statistical Learning Theory. Wiley.
56. Wang, J., Li, Y., & Zhang, Y. (2018). A novel defect detection method for flat surface based on deep convolutional neural network. IEEE Access, 6, 49366–49375.
57. Wang, L., Yu, Z., Ma, J., Chen, X., & Wu, C. (2025). A two-stage interpretable model to explain classifier in credit risk prediction. Journal of Forecasting, 44(3), 1–18.
58. Wang, Y., et al. (2020). Cloud-based deep learning for defect detection of complex products. IEEE Access, 8, 123456–123467.
59. Wager, S., & Athey, S. (2018). Estimation and inference with heterogeneous treatment effects using random forests. Journal of the American Statistical Association, 113(523), 1228–1242.
60. Zhao, D., Wang, Y., & Sheng, S. (2020). Comparison of regression model and feedforward network for monitoring weld quality. Journal of Manufacturing Processes, 56, 1340–1350.
61. Sahani, S. K., Lee, T.-F., Pandey, D., Pandey, B. K., Jha, R., & Karna, S. L. (2026). Machine learning–augmented hybrid risk management and deep uncertainty quantification in Nepalese management systems: Fractional stochasticity, Wasserstein robustness, and rough–path neural filtering. Journal of Intelligent Decision Making and Information Science, 3(3s), 1850–1873. https://doi.org/10.59543/jidmis.v3.971
62. Sahani, S. K., Lee, T.-F., Pandey, D., Pandey, B. K., & Mandal, K. (2026). Hybrid analytical, numerical, and machine learning frameworks for solving deterministic and stochastic differential equations with stability, convergence, and uncertainty quantification. Journal of Intelligent Decision Making and Information Science, 3(3s), 1794–1849. https://doi.org/10.59543/jidmis.v3.970
63. Sahani, S. K., Lee, T.-F., Pandey, D., Pandey, B. K., Sah, B. K., & Jha, R. (2026). Hybrid analytical, numerical and machine learning approaches on decision-making, risk management, operational stability, and uncertainty analysis in Nepalese management systems. Journal of Intelligent Decision Making and Information Science, 3(3s), 1772–1793. https://doi.org/10.59543/jidmis.v3.969
64. Sahani, S. K., Lee, T.-F., Pandey, D., Pandey, B. K., Sah, B. K., Jha, R., & Sah, D. K. (2026). Hybrid analytical–numerical–machine learning architectures for decision-making, risk management, operational stability, and uncertainty quantification in dental practice management systems: A mathematical theory. European Journal of Prosthodontics and Restorative Dentistry, 34(5s), 298–313. https://doi.org/10.1922/ejprd.v34i5s.1559
65. Sahani, S. K., Lee, T.-F., Pandey, D., Pandey, B. K., Sah, B. K., Mandal, K., & Jha, R. (2026). Biotechnological catalysts in hybrid analytical–numerical–machine learning architectures for risk-aware decision-making and uncertainty quantification in Nepalese management systems. European Journal of Prosthodontics and Restorative Dentistry, 34(7s), 332–350. https://doi.org/10.1922/ejprd.v34i7s.1622
66. Sahani, S. K., Lee, T.-F., Pandey, D., Pandey, B. K., Sah, B. K., & Jha, R. (2026). A hybrid analytical, numerical, and machine learning framework for health management, resource allocation, operational efficiency, risk forecasting, and strategic decision-making in Nepal. International Journal of Computer Information Systems and Industrial Management Applications, 18(9s), 1221–1242. https://doi.org/10.70917/ijcisim-2026-3400
67. Sahani, S. K., Lee, T.-F., Pandey, D., Pandey, B. K., Jha, R., & Karna, S. L. (2026). Explainable machine learning for credit risk management and intelligent lending decisions in Nepalese cooperative banks: A mathematical review. Journal of Intelligent Decision Making and Information Science, 3(6s), 1735–1772.