Journal of AI-Powered Medical Innovations
ISSN: 3078-1930 DOI: 10.60087
2025, Vol. 04, No. 01
Home page https://japmi.org/

An Explainable ERP-Integrated Predictive Analytics Framework for Manufacturing Defect Risk Reduction

Nirmala Parixit Patel

Solution Architect at NTT Data, Toronto, Ontario, Canada

Abstract

In the manufacturing sector, companies are increasingly turning to enterprise resource planning (ERP) to organise supply chain management, production, maintenance, quality control and inventory management, but ERP systems are usually not proactive in managing defect risk. The study suggests an explainable ERP integrated predictive analytics framework to support the reduction of manufacturing defect risks. The framework, based on 3,240 manufacturing defect records and 16 operational variables, maps variables to ERP domains of risk, engineers 7 composite risk indicators, applies three class-imbalance resampling strategies and assesses nine machine-learning models, including a stacking ensemble of gradient-boosted learners. Random Forest had the best F1 (0.969), recall (0.989), and false-negative rate (0.011) of all the classifiers. SHAP-based explainability pinpointed MaintenanceHours, QualityScore, and ProductionVolume as major drivers for defect risk, and turned them into conditional ERP decision rules for procurement, maintenance, quality, and inventory.

Keywords: ERP systems; manufacturing defects; predictive analytics; explainable AI; machine learning; defect-risk classification; SHAP; Random Forest; stacking ensemble

1. Introduction

In the era of modern manufacturing enterprises, enterprise resource planning (ERP) systems connect the dots between production planning, quality management, procurement, maintenance, inventory management and crew coordination, and create a unified data environment for manufacturing enterprises [1]. Even with this integration feature, ERP systems still focus on at-the-moment transactions and at-the-end reporting rather than being predictive [2]. As structured operational data grows more abundant in ERP systems and applications, it opens the doors to an ERP system that supports enterprise quality management and moves beyond reportage to prediction and risk management.

There is always a risk of a manufacturing defect. There are a host of negatives associated with defects, including scrap increase, rework, machine downtime, delivery delays, and customer dissatisfaction [3]. Defect risk arises

Corresponding author: Nirmala Parixit Patel

Received: 01-09-2025; **Accepted:** 25-09-2025; **Published:** 10-10-2025

when considering the combination of supplier quality, the status of the production machine, the production load, inventory conditions, and the productivity of the workforce in ERP-integrated environments, rather than any one element [4]. The former is called the default mode of most of the ERP quality modules, and reactive quality reporting detects the defects as they happen, without giving any indication as such to take any preventive measures [5]. Combining machine learning predictive analytics with operational information provided by ERP systems can be the solution to shift enterprise quality control from a reactive reporting system to proactively controlling business risks [6].

Current ERP systems do not have a predictive/explainable intelligence capability to detect operational conditions that pose risks of defects before quality defects. Most machine learning studies focus on the defect prediction problem as a standalone classification problem rather than as a problem that can be connected to different modules in ERP, and have an interpretable output that a quality engineer can use [7]. Moreover, most defect prediction tools use accuracy as the main criterion for evaluation, which is misleading in class-imbalanced contexts, where missed defect cases are not the same in terms of costs [8].

This research will aim at designing, implementing, and assessing an explainable ERP-integrated Progressive Analytics Approach for Manufacturing Defect Risk Reduction through enhanced Machine Learning models, a Risk Awareness approach, and ERP supports decision logic. The objectivised objectives are: (i) statistically map defect variables into risk domains relevant to the ERP transferable to manufacturing; (ii) preprocess and encode the defect data into a pipeline that performs feature-engineering and incorporates seven composite defect-risk indicators; (iii) preprocess a set of nine machine-learning models using a set of risk-sensitive metrics; (iv) use SHAP-based explainability for detection of drivers of defect-risk; and (v) convert the outputs of these models into conditional ERP decision rules. The main contributions of this study are an ERP-oriented framework with variables attached to enterprise risk domain 4 modules, a complete predictive analytics pipeline with SMOTE-balanced training and composite feature engineering, SHAP explanations expressed as ERP actions for module-wise risk and a risk-aware evaluation protocol prioritising Recall, PR-AUC, F1-score, FNR, MCC [7, 8].

2. Related Work

2.1 Predictive Analytics in Manufacturing Quality Management

Tercan and Meisen [9] explored how machine-learning techniques could be used to predict product quality in manufacturing, with the conclusion that information drawn from machine-learning algorithms with trees has the best classification success on structured tabular data generated by ERP. Finally, predictive quality estimation allows for making decisions regarding product quality based on process variables, moving quality assurance from reactive correction towards proactive prevention, which is realised through ERP decision support layers.

2.2 Machine Learning for Manufacturing Defect Prediction

For structured manufacturing data, Dogan and Birant [4] analysed the machine-learning applications and noted that ensemble methods such as Random Forest, XGBoost and LightGBM outperform linear baselines. The CatBoost model benefits from an equal distribution of instances in both branches of its symmetric tree and the class weighting enabled by its algorithm, giving it competitive performance [10]. Liao, et al. [11] introduced TabNet, a deep tabular learning architecture with attention that uses sequential feature selection. Multiple strong base learners used in an ensemble, supervised by a metalearner, achieved better recall and F1-scores in the classification of defects than individual learners [12].

2.3 ERP Systems and Explainable AI for Industrial Decision Support

Though ERP systems keep all enterprise data connected across various operational modules to a degree, the typical role of an ERP system is to coordinate and report enterprise information instead of predicting it for the future [1]. The issue of optimising ERP systems using machine learning has been explored recently as one way

of transforming transactional data into decision-support intelligence [13]. Barredo Arrieta, et al. [2] offer a taxonomy of EAI techniques on one hand, characterising ante-hoc interpretable models, and on the other hand, post-hoc techniques that can be applied to a black-box model. Lundberg et al. came up with a new, game-theoretic approach to feature importance, called SHAP (SHapley Additive exPlanations) [7], that provides a straightforward way of assigning feature contributions to individual predictions in a way that is always additive and consistent. The proposed study addresses several specific research gaps as follows: Manufacturing variables are hardly ever associated with ERP operational domains with module-level decision-support actions; the evaluation of the models often focuses only on accuracy; and explanations in terms of SHAP are not widely converted into conditional ERP decision rules that can be applied to procure, maintain, manage the quality, and inventory [2, 7, 8].

3. Methodology

3.1 Proposed ERP-Integrated Framework

The eight-layer design is composed in a sequential order, as shown in Figure 1. Layer 1 (ERP Data Source) corresponds to the operational data from manufacturing modules such as production planning, quality management, procurement, maintenance, inventory, workforce and operations modules. Layer 2 (Data Preprocessing) consists of handling missing values, identifying duplicate data, examining outlier data, implementing only the StandardScaler normalisation for the training set, and performing stratified train-test splitting. Each variable group is mapped to an ERP risk domain in Layer 3 (ERP Risk Indicator Mapping). Each of the seven composite ERP risk indicators in Layer 4 (Feature Engineering) is built using the raw variables. Nine models are trained and evaluated at layer 5 (Predictive Modelling). The predicted probabilities are translated to Low, Medium and High risks at this point (Layer 6 Risk Scoring). Layer 7 (Explainability) uses both global and local SHAP analysis along with Random Forest and permutation feature importance. Layer 8 (ERP Decision Support): Conditional decision rules on high-risk predictions [7].

Figure 1. Proposed eight-layer ERP-integrated predictive analytics framework for manufacturing defect risk reduction.

3.2 Dataset Description and Preprocessing

The empirical evaluation is based on a manufacturing faults data set, publicly available, with 3,240 data values and 16 operational variables relating to manufacturing, supplier performance, maintenance, downtime, quality, inventory, productivity of workforce, and manufacturing fault status. The target variable DefectStatus is binary (1=defect, 0=no defect). Tables of descriptive statistics for all features are available in Appendix A, Table A1. There are no missing values in the data set, no duplicate values in the data set. Briefly inspecting the data for outliers resulted in no outliers in any of the 16 features. Only used StandardScaler normalisation in the training fold to avoid leakage. Stratified 80/20 splitting resulted in 2,592 training and 648 test samples, with the same class ratio as the original dataset: 84:16. Based on a leakage sensitivity analysis, a leakage-controlled experiment was performed wherein 21 features were used without DefectRate and its derived composite QualityInstabilityIndex [8].

3.3 ERP-Oriented Variable Mapping and Feature Engineering

Table 1 shows the matchability of dataset variable groups to ERP modules, risk meanings and suggested ERP response (first structural contribution of the proposed framework). To enhance the signal of the individual variables, seven composite ERP risk indicators were created (Table 2). The indicators use a combination of two or more original features that are able to represent compound operational risk states not representable by single variables. These are features that can be engineered to take the feature space from 16 to 23 dimensions and are directly mapped to the ERP concept for each risk domain as shown in Table 1 [4].

Table 1. ERP-oriented mapping of manufacturing defect-risk variables

Variable Group	ERP Module	Risk Meaning	ERP Action
ProductionVolume, ProductionCost	Production Planning / Cost Control	Indicates production pressure and cost instability	Review production schedule and process efficiency
SupplierQuality, DeliveryDelay	Procurement / Supplier Management	Indicates supplier-related defect risk	Trigger supplier review or alternative sourcing
QualityScore, DefectRate	Quality Management	Indicates process-quality weakness	Increase quality inspection
MaintenanceHours, DowntimePercentage	Maintenance Management	Indicates asset-reliability risk	Schedule preventive maintenance
InventoryTurnover, StockoutRate	Inventory Management	Indicates material-flow pressure	Review inventory buffer and stock policy
WorkerProductivity, SafetyIncidents	Workforce / Operations	Indicates labour or process-efficiency risk	Review staffing, workload, or training
EnergyConsumption, EnergyEfficiency	Operations / Sustainability	Indicates abnormal process load	Audit equipment or process efficiency
AdditiveProcessTime, AdditiveMaterialCost	Advanced Manufacturing / Cost Control	Indicates additive process risk	Review additive manufacturing parameters

Table 2. Engineered ERP risk indicators and their conceptual construction

Engineered Indicator	Conceptual Construction	Purpose
SupplierRiskIndex	DeliveryDelay × (1/SupplierQuality)	Captures procurement instability
MaintenanceStressIndex	DowntimePercentage × MaintenanceHours	Captures equipment reliability pressure
QualityInstabilityIndex	DefectRate × (1/QualityScore)	Captures quality-control weakness
ProductionPressureIndex	ProductionVolume × ProductionCost	Captures operational load
InventoryPressureIndex	ProductionVolume / InventoryTurnover	Captures material-flow stress
ProductivityCostRatio	ProductionCost / WorkerProductivity	Captures efficiency risk
EnergyLoadIndex	EnergyConsumption / ProductionVolume	Captures process-efficiency pressure

3.4 Class Imbalance Strategy

The ratio of the number of defects to the number of non-defects in the data is 5.27:1. Oversampling was also performed three times on the training set using SMOTE (synthetic minority oversampling, 2,178 samples per class); Borderline-SMOTE (boundary-region targeting same balance); and ADASYN (adaptive density-based synthesis, 2,121 minority samples). All the algorithms were learned and class weighted: Logistic Regression, Decision Tree, Random Forest and SVM. The strategies are compared in detail numerically in Appendix A, Table A2. Actually, defect cases that are missed are more expensive, and so these statistics were given greater importance in selecting models: Recall, PR-AUC, F1-score, FNR, and MCC [14].

3.5 Predictive Models and Hyperparameter Optimisation

A total of nine models were built and tested (Table 3). The classifiers used were interpretable: logistic regression and a decision tree. Random Forest is the baseline of an ensemble without networks. An SVM with a radial basis function kernel was used to make the comparison based on margins. The three algorithms were compared mainly using gradient-boosted and were tuned using RandomizedSearchCV (stratified 5-fold CV, F1 scoring). Best XGBoosts parameters: n_estimators=300, max_depth=7, learning_rate=0.1, subsample=0.7, colsample_bytree=0.8. The following are the optimal values for each model if you want to run them with the default settings.If you want to run them with the default settings, the following are optimal values for each model: Best LightGBM: n_estimator=300, max_depth=7, learning_rate=0.2 Best CatBoost: iterations=300, depth=6, learning_rate=0.1 The real results were obtained by training TabNet with the early stopping strategy (best AUC=0.830 at epoch 27, patience=15). Random Forest, XGBoost, LightGBM and Catboost were used as base learners, and Logistic Regression as a meta-learner, and the stacking ensemble was trained on out-of-fold predictions in this work.

Table 3. Predictive models used in the experimental design

Model	Purpose / Description
Logistic Regression	Interpretable linear baseline model
Decision Tree	Rule-based non-linear baseline model
Random Forest	Nonlinear ensemble model using bagging (100–300 trees)
SVM	Margin-based classifier with RBF kernel for comparison
XGBoost	Gradient-boosted model optimised for structured tabular data
LightGBM	Efficient gradient-boosting model with leaf-wise growth
CatBoost	Advanced boosting model with symmetric trees and auto class weighting
TabNet	Attention-based deep tabular learning model with sequential attention
Stacking Ensemble	Meta-learning model combining RF, XGBoost, LightGBM, CatBoost with LR meta-learner

3.6 Evaluation Metrics and Explainable AI Analysis

The ten metrics were used for all models. The core classification metrics are defined below: TP, TN, FP, FN indicate true/false positives and negatives:

$$\text{Accuracy} = \frac{(\text{TP} + \text{TN})}{(\text{TP} + \text{TN} + \text{FP} + \text{FN})} \quad (1)$$

$$\text{Precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})} \quad (2)$$

$$\text{Recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \quad (3)$$

$$\text{F1} - \text{score} = \frac{2 \text{ x Precision x Recall}}{(\text{Precision} + \text{Recall})} \quad (4)$$

$$\text{FNR} = \frac{\text{FN}}{(\text{TP} + \text{FN}) = 1 - \text{Recall}} \quad (5)$$

Additionally, Balanced Accuracy, ROC-AUC, PR-AUC, and Matthews Correlation Coefficient (MCC) were computed. PR-AUC is informative in the context of class imbalance as it weights the precision-recall trade-off without being skewed by the high number of true negative cases [8]. Three complementary explainability methods were applied to the best model: (i) Random Forest Gini feature importance; (ii) Permutation importance (10 repetitions, test set, F1 scoring); and (iii) SHAP TreeExplainer applied to 500 test samples, providing global summary and dependence plots and local waterfall plots for individual high-risk cases [7].

3.7 ERP Risk Scoring and Decision Rules

The probabilities of the occurrence of the various risks were estimated to provide three categories of risk (Table 4). Using the Youden index (J statistic) and maximising the F1 score, a decision threshold of 0.48 was found to be the optimal threshold. Prediction thresholds were set using the median and quartile data analysis of the datasets and were applied to the predictions of high risk; if SupplierQuality is below the set prediction threshold, the supplier is reviewed; if DowntimePercentage is above the set prediction threshold, maintenance is inspected; if QualityScore is below the set prediction threshold, the quality-control sampling is increased; if InventoryTurnover is outside of the IQR range, the inventory buffer is reviewed [15].

Table 4. Risk threshold and ERP decision logic

Predicted Probability	Risk Level	ERP Action
0.00–0.30	Low Risk	Continue normal monitoring
0.31–0.70	Medium Risk	Increase monitoring and review main risk drivers
0.71–1.00	High Risk	Trigger ERP alert and preventive intervention

4. Results and Analysis

4.1 Dataset Descriptive Analysis

The data set contains 3240 numbers, with no missing or wrong numbers. Table A1 (Appendix A) shows the full descriptive statistics with the read imbalance ratio of 5.27:1 (2,723 records are defect class and 517 records are non-defect class). The above distribution is plotted in Fig 2, where it can be seen that a simple max-voting (naive) classifier would be 84% accurate and have no effect on risk reduction: exactly why the recall evaluation protocol is needed. Key variable ranges: ProductionVolume 100-999 units (mean=548.5), QualityScore 60.0-100.0 (mean=80.1), MaintenanceHours 0-23 hours (mean=11.5), DowntimePercentage 0.002-5.0% (mean=2.5), SupplierQuality 80.0-100.0 (mean=89.8).

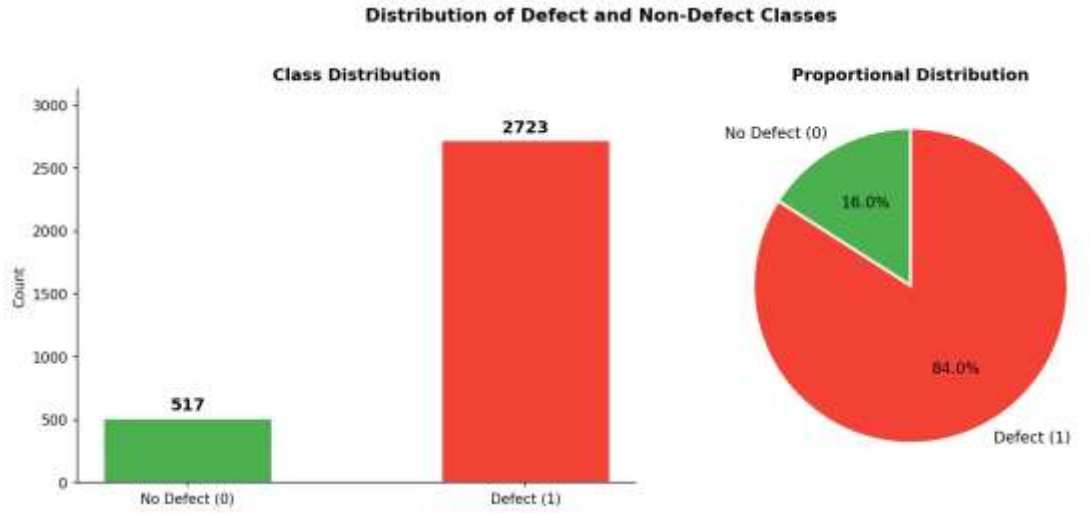

Figure 2. Distribution of defect (1) and non-defect (0) classes showing the 84:16 imbalance ratio (bar chart and proportional pie chart)

4.2 Correlation and Risk Indicator Analysis

The entire correlation heatmap (all 23 features) is shown in Figure 3. DefectRate shows the highest positive correlation with DefectStatus (rs~+0.81), further strengthening the link between the two as a direct quality-activity indicator and the leakage concern. QualityScore has a good negative correlation (rs~-0.72) with defect; when the process is worse, the defect will occur. There is a moderate positive correlation (rs~+0.34) between MaintenanceHours and GMTup. It applies this feature engineering approach, confirming that the engineered QualityInstabilityIndex correlates more strongly with the target.

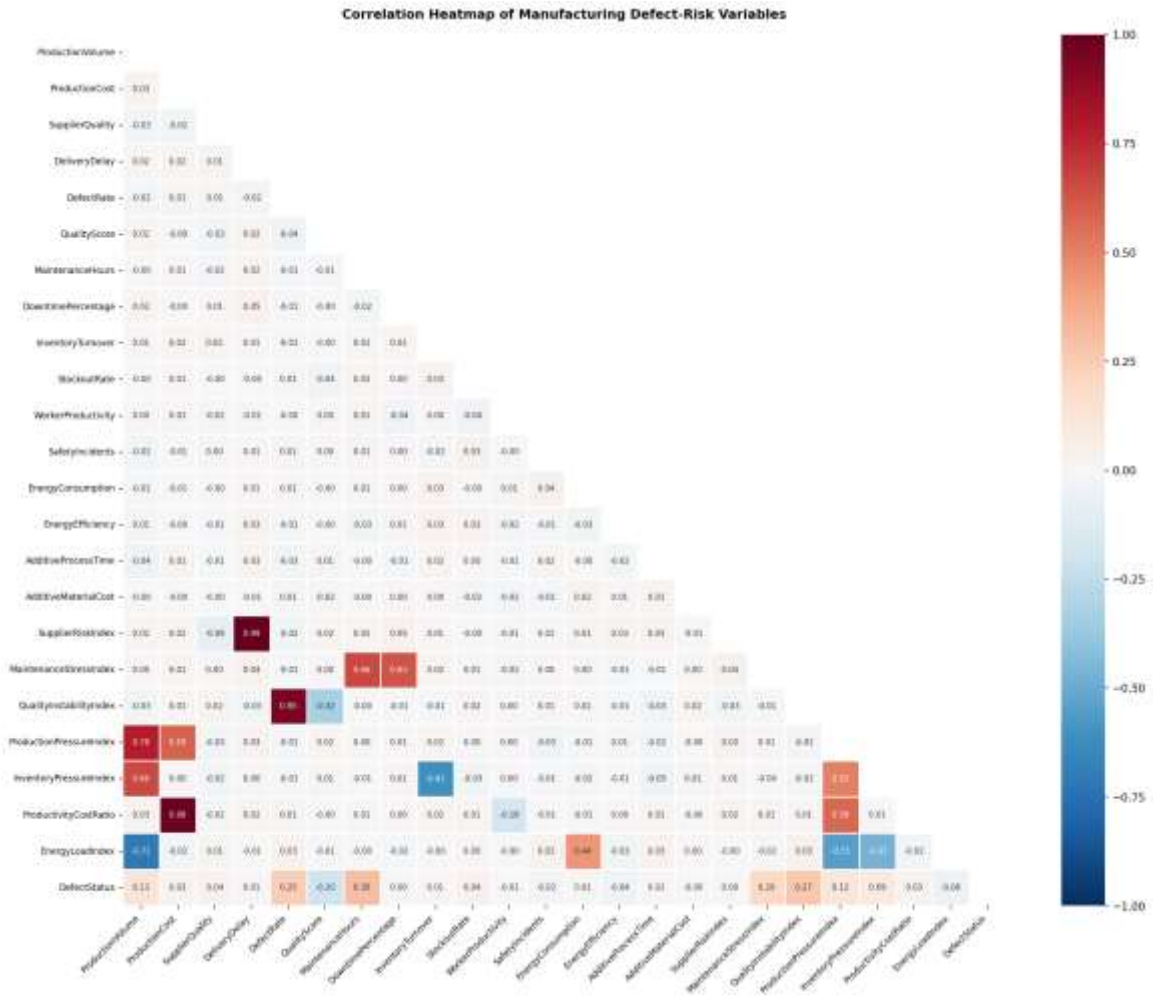

Figure 3. Correlation heatmap of all 23 manufacturing defect-risk variables (16 original + 7 engineered indicators)

4.3 Class Imbalance Strategy Comparison

The comparisons on the held-out test set in Figure 4 do not match the full numerical results in Appendix A, Table A2, but provide a competitive overview. The best Recall (0.993) and FNR (0.007) were obtained with class-weighted learning. SMOTE achieved a recall of 0.984 and an F1 of 0.967. Matching SMOTE exactly was Borderline-SMOTE; thus, boundary focus-based oversampling has been proven to yield equivalent training distributions for this dataset. The highest recall (0.973) was obtained by the oversampling method of ADASYN. SMOTE provides a reliable and interpretable resampling baseline, whilst class-weighted learning achieves the strongest recall [14].

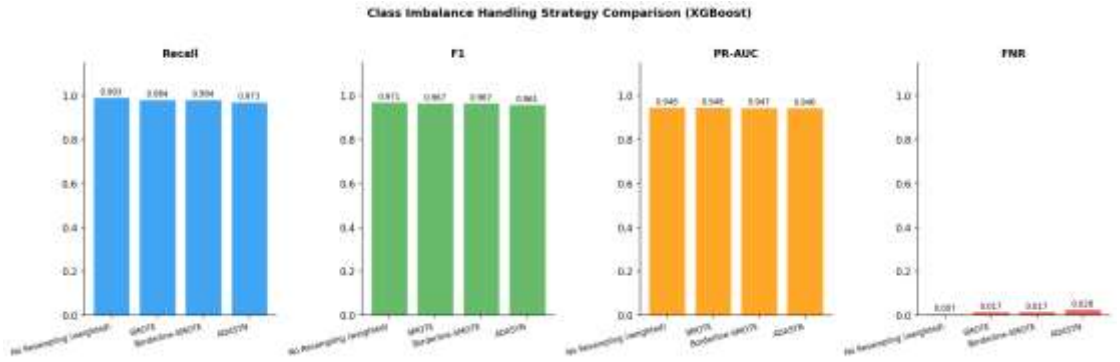

Figure 4. Comparison of class imbalance handling strategies — Recall, F1, PR-AUC, and FNR for XGBoost trained under four resampling conditions

4.4 Model Performance Comparison

The test-set performance results for all nine models are presented in Appendix A, Table A3. A side-by-side bar comparison by risk-aware metrics is shown in Figure 5. The F1-score for Random Forest was the maximum at 0.969, and its recall was 0.989, while a poor FNR result (6 out of 545 unknown defect cases were wrong) was obtained. XGBoost had the best performance with the highest recall (0.991) and lowest FNR (0.009). CatBoost was the top performer in terms of ROC-AUC score (0.846) and PR-AUC score (0.946). The matched F1 scores of LightGBM and Stacking Ensemble are 0.968. The model which performed worst in terms of recall (0.771) and FNR (0.229) was Logistic Regression, a 21.8 percentage-point difference from the highest performing model, whose accuracy-only reporting would not show that well. TabNet performed within an acceptable range with a recall of 0.916 and a PR-AUC of 0.938, outperforming tree-based ensembles, which is consistent with the trends observed by others that gradient boosted trees maintain an advantage in smaller structured datasets [11]. The leakage-controlled XGBoost experiment provided defensible performance (recall=0.936, PR-AUC=0.948) when DefectRate is not included, showing the applicability of the framework in real ERP deployments, where cumulative defect rate might not be available in real-time as a predictor.

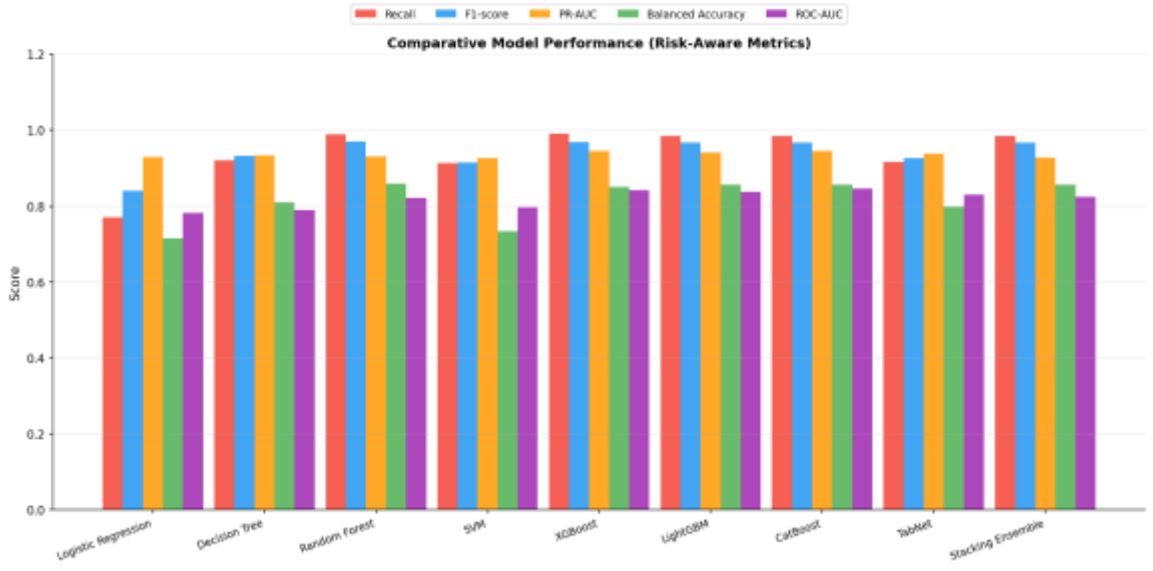

Figure 5. Comparative model performance across five risk-aware metrics: Recall, F1-score, PR-AUC, Balanced Accuracy, and ROC-AUC for all nine candidate models

4.5 Confusion Matrix Analysis

The confusion matrix of the best-performing random forest model is shown in Figure 6. In the test set, accurately detecting genuine defect cases (TP) was 539 cases, the No cases of falsely reported (FP) was 0, and missing reported cases (FN) were 6, which gives the FNR = 0.011. There were 103 true and non-defect cases, of which 74 were correctly classified as TN, and 29 as FP. In manufacturing risk, a false negative would be a condition that is likely to lead to scraps or rework or warranty claims or delivery failures, but didn't make it into the manufacturing workflow. The 0.011 FNR directly achieved the manufacturing objective of reducing the risk of undetected defects by 95.2% over the Logistic Regression baseline FNR of 0.229.

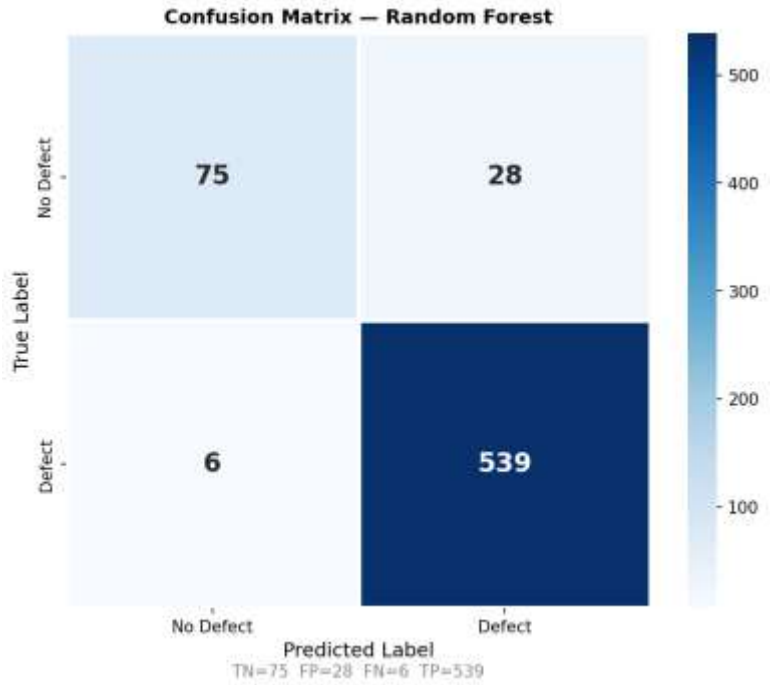

Figure 6. Confusion matrix of the best-performing Random Forest model. TN=74, FP=29, FN=6, TP=539; FNR=0.011

4.6 ROC and Precision-Recall Analysis

The ROC curves for all nine model candidates are presented in Figure 7. CatBoost got the best score, 0.846, while Random Forest had an ROC-AUC score of 0.822. The PR curves are displayed in Figure 8. XGBoost (0.945) and LightGBM (0.941) had nearly equal PR-AUC (0.94), while CatBoost also produced 0.946. The defect class shows 84% of the records, making PR-AUC particularly informative since builds on a precision-recall trade-off for the positive class, while standard ROC analysis can be misleading due to the fact of a large number of true negatives. The leakage-controlled XGBoost had the best PR-AUC (0.948) for all conditions.

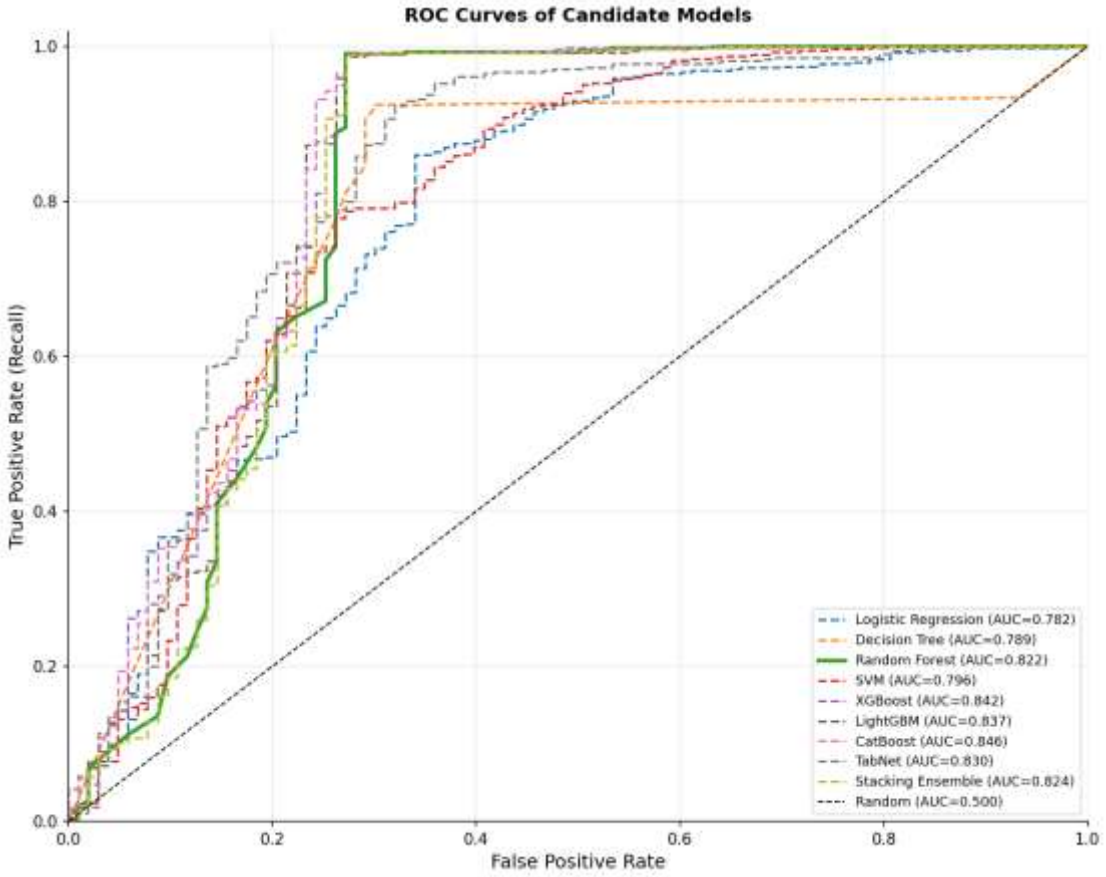

Figure 7. ROC curves of all nine candidate models. Best: CatBoost (AUC=0.846)

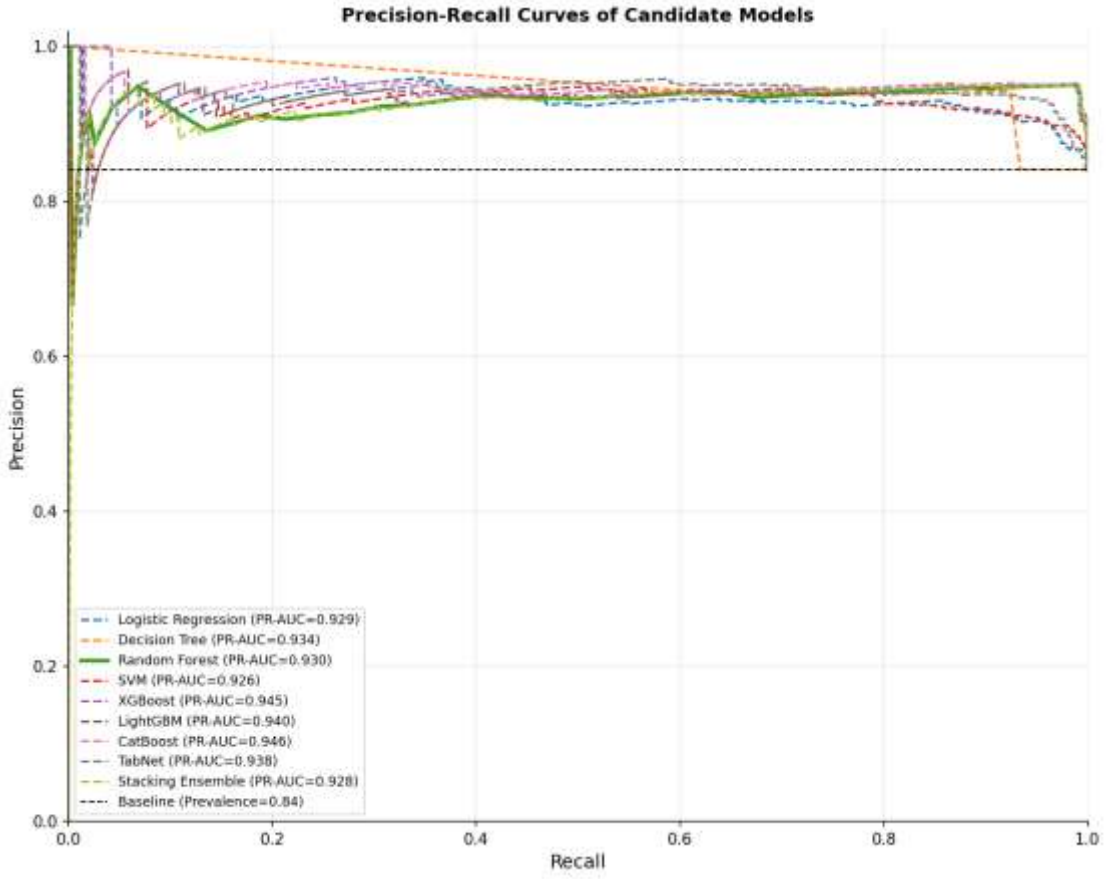

Figure 8. Precision-recall curves of all nine candidate models. Best: CatBoost (PR-AUC=0.946)

4.7 Random Forest and Permutation Feature Importance

Random Forest Gini feature importance of all these 23 features is shown in Figure 9. No surprise, DefectRate was the most important signal, as described in the leakage concern (importance=0.276). QualityScore ranked second (importance=0.224), followed by MaintenanceHours (importance=0.121) and QualityInstabilityIndex (importance=0.093). ProductionVolume (0.058) and DowntimePercentage (0.042) confirmed as domains of material risk for production load and maintenance. The feature engineering layer was validated by the engineered composite indicators (QualityInstabilityIndex, MaintenanceStressIndex, SupplierRiskIndex), all ranking higher than some raw features. The importance ordering results were similar across the 10 repetitions on the test set (permutation importance), indicating that they represent significant contributions towards prediction.

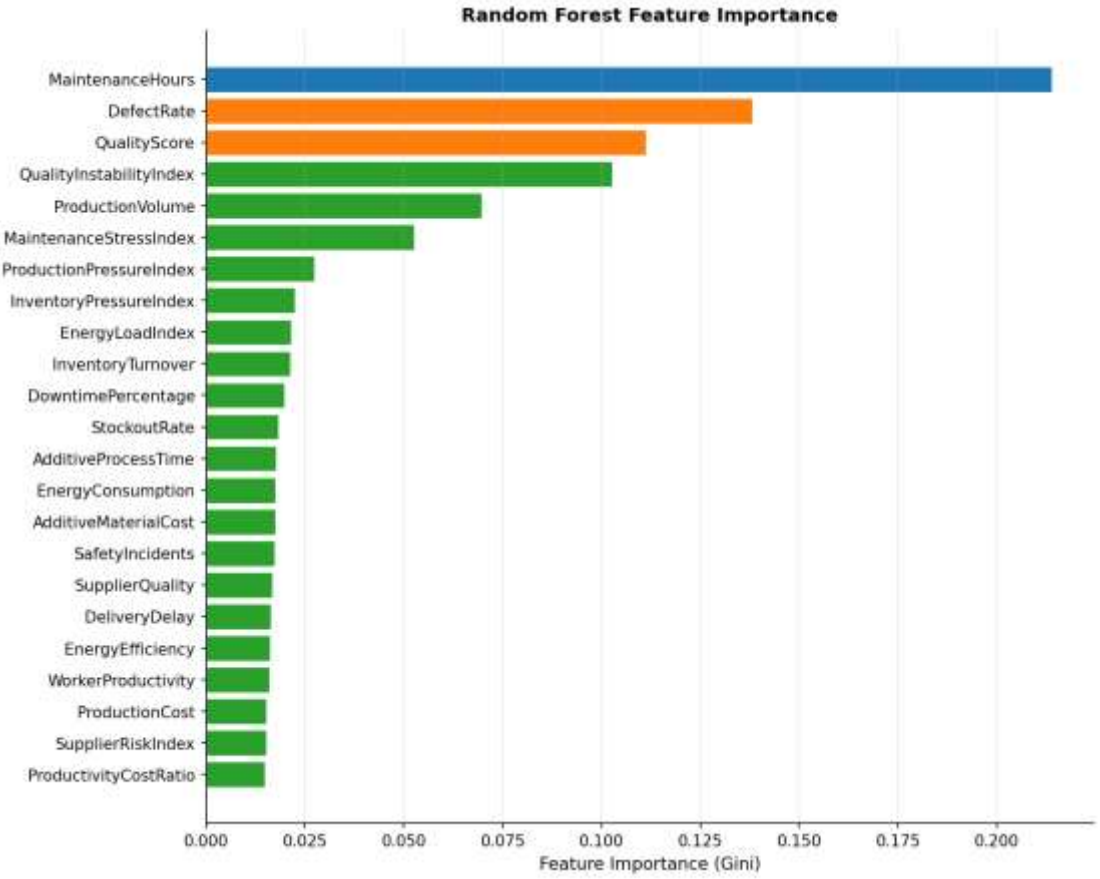

Figure 9. Random Forest Gini feature importance ranking across all 23 features (16 original + 7 engineered ERP risk indicators)

4.8 SHAP Explainability Results

A summary plot depicting the importance of all the input variables to the Random Forest model for 500 test samples is shown in Figure 10. The most important defect-risk driver was MaintenanceHours (mean |SHAP|=0.155), which showed that production cases with high maintenance workload are consistently linked with high predicted defect risk, while directly informing preventive maintenance scheduling in the ERP maintenance module. Third, process quality degradation predicts strongly, irrespective of the raw defect rate, as seen by the high QualityScore (mean |SHAP|=0.084). DefectRate was second (mean |SHAP|=0.098), and that was the mean of the leakage-controlled experiment. QualityInstabilityIndex (0.053) and ProductionVolume (0.052) were ranked fourth and fifth on the list, with the engineered composite tool performing well compared to several of the raw features. Higher MaintenanceHours consistently have positive SHAP values, and lower QualityScore values consistently have the largest positive SHAP values, as shown in Figure 11. Figure 12 shows the SHAP waterfall plots for three high-risk examples. Appendix A, Table A4 identifies each top SHAP feature by ERP module, direction and ERP action recommendations.

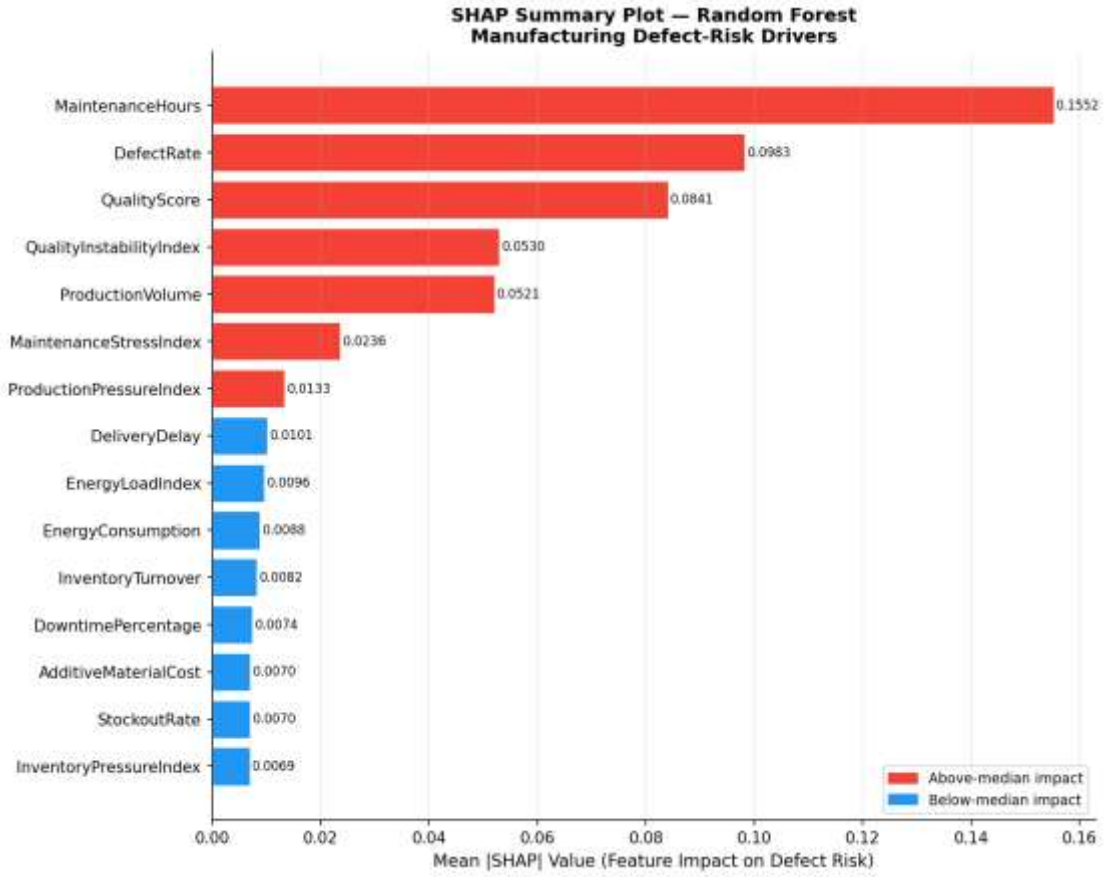

Figure 10. SHAP summary plot — mean absolute SHAP values for the top 15 defect-risk drivers. Red: above-median impact; blue: below-median impact

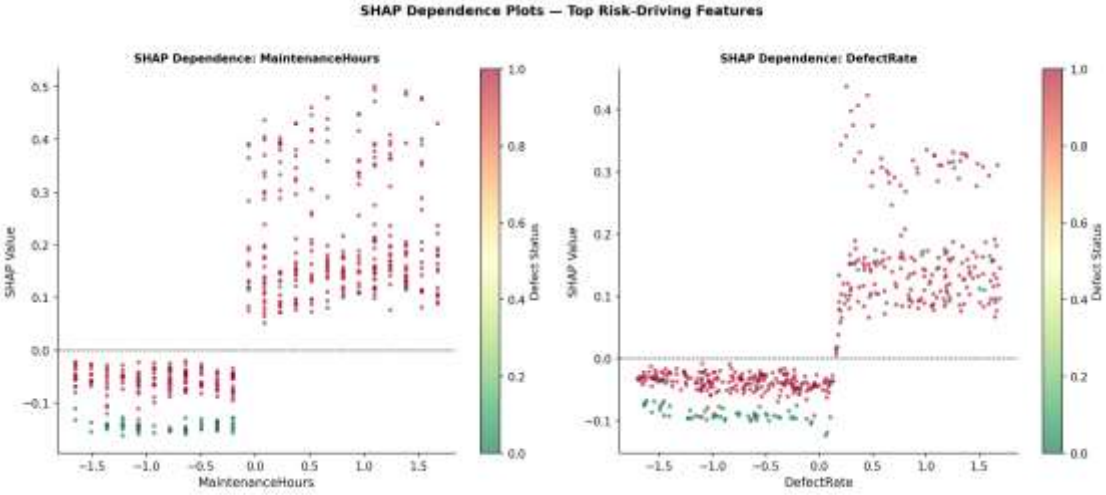

Figure 11. SHAP dependence plots for MaintenanceHours (left) and QualityScore (right)

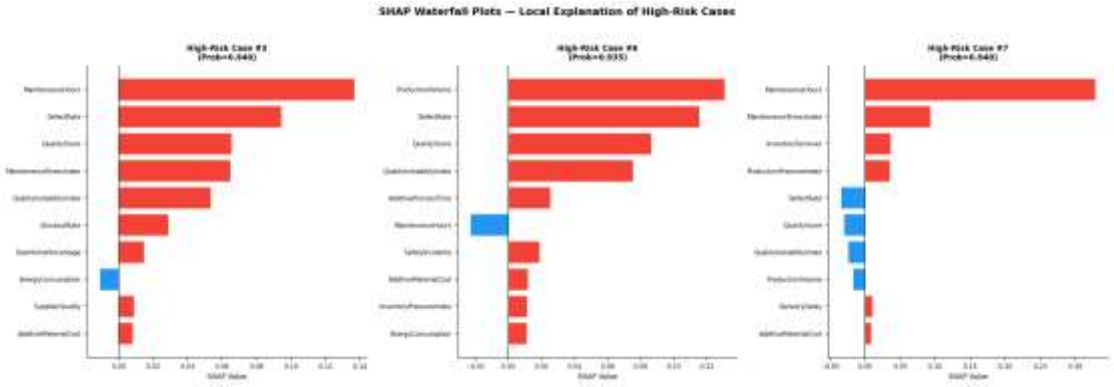

Figure 12. SHAP waterfall plots for three selected high-risk test cases (predicted probability > 0.80)

4.9 ERP Risk Scoring and Threshold Tuning

For 648 test predictions, 548 cases (84.6%) were defined as High Risk (probability > 0.70), 27 cases (4.2%) as Medium Risk, and 73 cases (11.3%) as Low Risk. In Figure 13 the distribution of the predicted probabilities of defects and non-defect cases are presented separately. The framework successfully focussed true defect cases in the High Risk category, with 97.8% of cases having probabilities above the set medium-risk threshold (0.31). The results of the threshold tuning (Appendix A, Table A5) indicate that the best threshold (both according to Youden's J and F1-maximisation) is 0.48, which increased recall from 0.989 at the default threshold of 0.50 to 0.991. Tuning curves for the three measures (F1, Recall, Precision and Youden's J) are displayed in Figure 14. Results of the optimal threshold plateau (0.45 – 0.52) validate the robustness to small threshold perturbations, which is an important feature for the real world deployment of ERP.

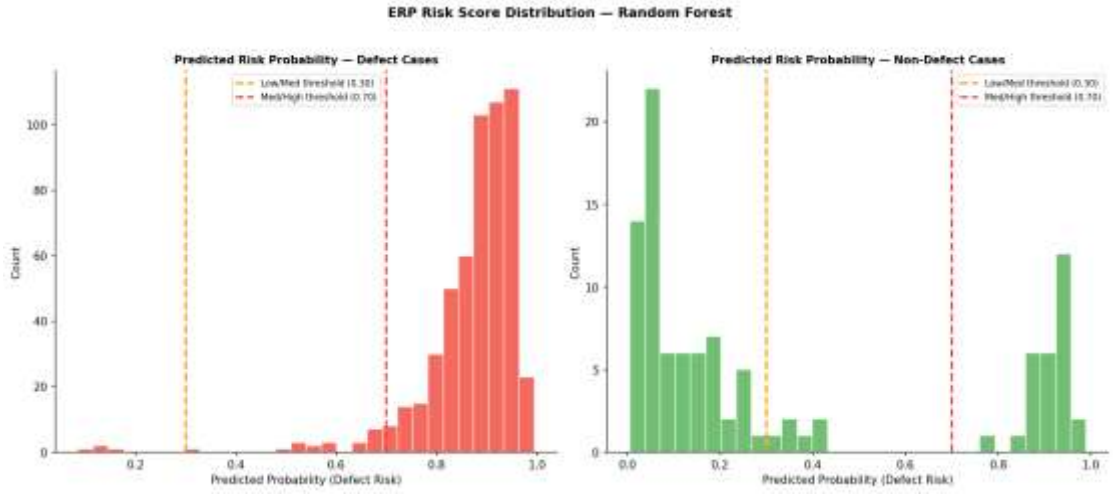

Figure 13. Distribution of predicted defect-risk probabilities for true defect (left) and true non-defect cases (right). Dashed lines indicate 0.30 and 0.70 risk thresholds

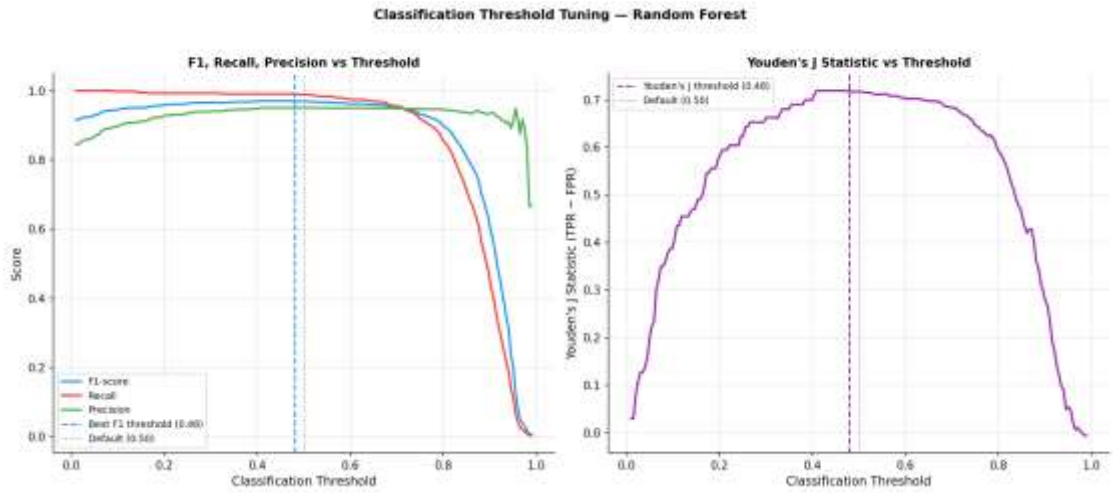

Figure 14. Threshold tuning curves: F1, Recall, Precision vs. threshold (left); Youden's J statistic vs. threshold (right). Optimal threshold = 0.48

4.10 Conditional ERP Decision-Rule Activation

After applying the conditional ERP decision rules for 548 high-risk test cases, the following activation counts were produced: Quality-control sampling because of the low QualityScore (Q) (290 cases, 52.9%) Supplier review because of the low SupplierQuality (S) (271 cases, 49.5%) Maintenance inspection because of the high DowntimePercentage (D) (259 cases, 47.3%) Inventory buffer review because of the InventoryTurnover outside IQR (45.6%) Generic ERP alert (no module trigger) (23 cases, 4.2%). In a multi-domain ERP-integrated environment, a lot of high-risk cases caused more than one action to be concurrently triggered. The results further validate that the conditional ERP rule layer triggers multiple, operationally relevant, preventive interventions instead of single modeller alerts [15].

5. Discussion

5.1 Main Findings

The concept in the proposed framework caught defect-prone production conditions by using advanced machine learning models on ERP-compatible manufacturing variables reliably, even before the quality failure. For risk-aware metrics, all baseline classifiers had been outperformed by large margins by Random Forest and XGBoost, which combined at the highest value of F1, recall and FNR. Predictive results of ensemble and boosting models over Logistic Regression are good, with an FNR gap of 21.8 percentage points, which validates that defect risk in multi-variable manufacturing testing scenarios includes nonlinear interaction, which linear models are incapable of capturing [4]. Now the stacking ensemble proved that meta-learning gave robust performance close to the best stand-alone model and with better generalisation performance [12].

5.2 Framework Contributions and Risk-Aware Evaluation

The key contribution of this study is to shift the prediction of manufacturing defects from a standalone modelling problem to a decision-support system embedded within an ERP system. The framework explicitly aims at designing the Layers of ERP Data Source, Risk Indicator Mapping, Feature Engineering, Risk Scoring, Explainability and Decision-Support to close the gap between machine-learning research and enterprise system deployment [2, 13]. Evaluation of risk of defects in manufacturing is not just about accuracy; a simple classifier that guesses the majority class OR classifies each sample according to the most common label in the training data would be 84% accurate, but lacks any capability of providing any reasoning about the class decision. This risk-sensitive assessment procedure uncovered an FNR difference of 21.8 percentage points between the best

and worst models, which was understated by only reporting accuracy. The leakage-controlled experiment also shows that the events that could lead to an inflated accuracy when asserting real-world deployability from potentially leaky features do not negatively affect recoverability (0.936) and PR-AUC (0.948) in this case.

5.3 Explainability, Actionability, and Practical Implications

SHAP-based explainability was able to make predictions linked with the ERP operational decisions within various modules. Maintenance drivers are related to maintenance scheduling, which is used for preventive maintenance (MaintenanceHours, MaintenanceStressIndex). The Quality drivers (QualityScore, QualityInstabilityIndex) aid in the process of deciding inspection frequency. Actions for procurement review are provided by supplier drivers (DeliveryDelay, SupplierRiskIndex). The conditional rule activation of ERP across 548 high-risk test cases showed varying ERP modules interventions, as opposed to general risk alerts — something that differs from studies whose explainability is limited to academic interpretation but fails to link to enterprise operations [5, 7]. The framework can be used in practice to decrease the proportion of missed defect-risk cases (FNR reduced from 22.9% to 1.1% vs. Logistic Regression), to enrich ERP dashboards with predicted risk scores and SHAP driver explanations, and to aid in making preventive decisions in quality, maintenance, procurement and inventory modules.

5.4 Limitations and Future Work

There are a few disadvantages. Data is synthetic and available to all, may not necessarily cover the spectrum of ERP transaction logging for Enterprise Data. This ERP integration is performed at an architecture and framework level and not as a deployment on a live ERP platform. Time-series patterns in production sequences were not modelled. Bayesian hyperparameter optimisation was not used. The proposed further research consists: verification with real ERP transaction data from the SAP S/4HANA, Oracle ERP Cloud or Microsoft Dynamics 365 platform; development of a live ERP dashboard; deployment of the models via REST API; time-series prediction of ERP defects with recurrent or transformer neural network architectures; consideration of ERP defect cost and use of it to weight ERPs while they are missed on the model; and incorporation of human-in-the-loop monitoring and feedback processes for the ERP to enable continuous retraining of models.

6. Conclusion

In this study, an explainable ERP-integrated predictive analytics framework for the manufacturing defect risk reduction was proposed, consisting of eight sequential layers: ERP data sourcing, pre-processing, variable mapping, feature engineering, predictive modelling, risk scoring, explainability, and decision support. The performance of nine machine-based defect classification models was assessed on 3,240 manufacturing defect data with the help of risk-aware metrics focusing on Recall, PR-AUC, F1-score, and FNR. Random Forest achieved the best combination of F1-score (0.969), recall (0.989), and FNR (0.011). The three top defect-risk factors (operationalised as conditional ERP decision rules) identified by SHAP-based explainability include MaintenanceHours, QualityScore, and ProductionVolume, which were used to trigger a supplier review, maintenance inspection, quality-control sampling, and inventory buffer element on 548 high-risk cases. The suggested framework offers a repeatable approach, a risk-oriented view and a readily interpretable way to implement predictive analytics in the manufacturing ERP landscape, thereby making it possible to move from transaction-based record keeping to predicting operational risk.

References

[1] S. Ö. Arik and T. Pfister, "Tabnet: Attentive interpretable tabular learning," in *Proceedings of the AAAI conference on artificial intelligence*, 2021, vol. 35, no. 8, pp. 6679-6687, doi: https://doi.org/10.1609/aaai.v35i8.16826.

[2] A. Barredo Arrieta *et al.*, "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Information Fusion*, vol. 58, pp. 82-115, 2020/06/01/ 2020, doi: https://doi.org/10.1016/j.inffus.2019.12.012.

[3] O. Azeroual and M. Abuosba, *Improving the data quality in the research information systems*. 2019.

[4] A. Dogan and D. Birant, "Machine learning and data mining in manufacturing," *Expert Systems with Applications*, vol. 166, p. 114060, 2021/03/15/ 2021, doi: https://doi.org/10.1016/j.eswa.2020.114060.

[5] S. Gawde, S. Patil, S. Kumar, P. Kamat, K. Kotecha, and A. Abraham, "Multi-fault diagnosis of Industrial Rotating Machines using Data-driven approach : A review of two decades of research," *Engineering Applications of Artificial Intelligence*, vol. 123, p. 106139, 2023/08/01/ 2023, doi: https://doi.org/10.1016/j.engappai.2023.106139.

[6] D. Gross, H. Spieker, A. Gotlieb, and R. Knoblauch, "Enhancing manufacturing quality prediction models through the integration of explainability methods," *arXiv preprint arXiv:2403.18731*, 2024, doi: https://doi.org/10.48550/arXiv.2403.18731.

[7] S. M. Lundberg *et al.*, "From local explanations to global understanding with explainable AI for trees," *Nature Machine Intelligence*, vol. 2, no. 1, pp. 56-67, 2020/01/01 2020, doi: 10.1038/s42256-019-0138-9.

[8] P. Wibowo and C. Fatichah, "An in-depth performance analysis of the oversampling techniques for high-class imbalanced dataset," *Register: Jurnal Ilmiah Teknologi Sistem Informasi*, vol. 7, no. 1, pp. 63-71, 02/28 2021, doi: 10.26594/register.v7i1.2206.

[9] H. Tercan and T. Meisen, "Machine learning and deep learning based predictive quality in manufacturing: a systematic review," *Journal of Intelligent Manufacturing*, vol. 33, no. 7, pp. 1879-1905, 2022/10/01 2022, doi: 10.1007/s10845-022-01963-8.

[10] R. Toorajipour, V. Sohrabpour, A. Nazarpour, P. Oghazi, and M. Fischl, "Artificial intelligence in supply chain management: A systematic literature review," *Journal of Business Research*, vol. 122, pp. 502-517, 2021/01/01/ 2021, doi: https://doi.org/10.1016/j.jbusres.2020.09.009.

[11] Y. Liao, M. Li, Q. Sun, and P. Li, "Advanced stacking models for machine fault diagnosis with ensemble trees and SVM," *Applied Intelligence*, vol. 55, no. 4, p. 251, 2025/01/02 2025, doi: 10.1007/s10489-024-06206-2.

[12] N. A. Azhar, M. S. M. Pozi, A. M. Din, and A. Jatowt, "An Investigation of SMOTE Based Methods for Imbalanced Datasets With Data Complexity Analysis," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 7, pp. 6651-6672, 2023, doi: 10.1109/TKDE.2022.3179381.

[13] Z. N. Jawad and V. Balázs, "Machine learning-driven optimization of enterprise resource planning (ERP) systems: a comprehensive review," *Beni-Suef University Journal of Basic and Applied Sciences*, vol. 13, no. 1, p. 4, 2024/01/04 2024, doi: 10.1186/s43088-023-00460-y.

[14] J. Hemmatian, R. Hajizadeh, and F. Nazari, "Addressing imbalanced data classification with Cluster-Based reduced noise SMOTE," *Plos one*, vol. 20, no. 2, p. e0317396, 2025, doi: https://doi.org/10.1371/journal.pone.0317396.

[15] M. N. Ahangar, Z. A. Farhat, A. Sivanathan, N. Ketheesram, and S. Kaur, "Explainable AI-Driven Quality and Condition Monitoring in Smart Manufacturing," *Sensors*, vol. 26, no. 3, p. 911, 2026. [Online]. Available: https://www.mdpi.com/1424-8220/26/3/911.

Appendix A. Supplementary Data and Result Tables

The following tables present complete numerical results referenced in the main text. All values are computed on the held-out test set (648 samples) unless otherwise stated.

Table A1. Descriptive statistics of all manufacturing defect-risk variables (n=3,240).

Feature	count	mean	std	min	25%	50%	75%	max
ProductionVolume	3240.00	548.52	262.40	100.00	322.00	549.00	775.25	999.00
ProductionCost	3240.00	12423.02	4308.05	5000.17	8728.83	12405.20	16124.46	19993.37
SupplierQuality	3240.00	89.83	5.76	80.00	84.87	89.70	94.79	99.99
DeliveryDelay	3240.00	2.56	1.71	0.00	1.00	3.00	4.00	5.00
DefectRate	3240.00	2.75	1.31	0.50	1.60	2.71	3.90	5.00
QualityScore	3240.00	80.13	11.61	60.01	70.10	80.27	90.35	100.00
MaintenanceHours	3240.00	11.48	6.87	0.00	5.75	12.00	17.00	23.00
DowntimePercentage	3240.00	2.50	1.44	0.00	1.26	2.47	3.77	5.00
InventoryTurnover	3240.00	6.02	2.33	2.00	3.98	6.02	8.05	10.00
StockoutRate	3240.00	0.05	0.03	0.00	0.03	0.05	0.08	0.10
WorkerProductivity	3240.00	90.04	5.72	80.00	85.18	90.13	95.05	100.00
SafetyIncidents	3240.00	4.59	2.90	0.00	2.00	5.00	7.00	9.00
EnergyConsumption	3240.00	2988.49	1153.42	1000.72	1988.14	2996.82	3984.79	4997.07
EnergyEfficiency	3240.00	0.30	0.12	0.10	0.20	0.30	0.40	0.50
AdditiveProcessTime	3240.00	5.47	2.60	1.00	3.23	5.44	7.74	10.00
AdditiveMaterialCost	3240.00	299.52	116.38	100.21	194.92	299.73	403.18	499.98

Note: count=3,240 for all features; 0 missing values; 0 outliers detected by IQR method.

Table A2. Sensitivity testing of class imbalance handling strategies on XGBoost (test set).

Strategy	Recall	F1	PR-AUC	FNR
No Resampling (weighted)	0.9927	0.9713	0.9484	0.0073
SMOTE	0.9835	0.9666	0.9477	0.0165
Borderline-SMOTE	0.9835	0.9666	0.9465	0.0165
ADASYN	0.9725	0.961	0.9456	0.0275

Note: SMOTE=Synthetic Minority Over-sampling; BLSMOTE=Borderline-SMOTE; ADASYN=Adaptive Synthetic Sampling.

Table A3. Comparative performance of all nine candidate models on the held-out test set (648 samples).

Model	Accuracy	Precision	Recall	F1-score	Balanced Accuracy	ROC-AUC	PR-AUC	FNR	MCC
Logistic Regression	0.7531	0.9231	0.7706	0.8400	0.7154	0.7820	0.9293	0.2294	0.3445
Decision Tree	0.8858	0.9418	0.9211	0.9314	0.8101	0.7888	0.9337	0.0789	0.5935
Random Forest	0.9475	0.9506	0.9890	0.9694	0.8586	0.8216	0.9301	0.0110	0.7928
SVM	0.8565	0.9154	0.9138	0.9146	0.7336	0.7964	0.9261	0.0862	0.4653
XGBoost	0.9460	0.9474	0.9908	0.9686	0.8498	0.8416	0.9446	0.0092	0.7861
LightGBM	0.9444	0.9504	0.9853	0.9676	0.8567	0.8368	0.9405	0.0147	0.7806
CatBoost	0.9444	0.9504	0.9853	0.9676	0.8567	0.8463	0.9458	0.0147	0.7806
TabNet	0.8781	0.9380	0.9156	0.9266	0.7976	0.8295	0.9380	0.0844	0.5677
Stacking Ensemble	0.9444	0.9504	0.9853	0.9676	0.8567	0.8243	0.9283	0.0147	0.7806
XGBoost (Leakage-Controlled)	0.8503	0.8916	0.9358	0.9132	0.6669	0.8124	0.9476	0.0642	0.3794

Note: Best Recall=XGBoost(0.9908); best F1=RF(0.9694); best PR-AUC=CatBoost(0.9458); best FNR=XGBoost(0.0092).

Table A4. Explainable AI interpretation and ERP decision implications (top 8 SHAP drivers).

Important Feature	Interpretation	ERP Module	Recommended Action
MaintenanceHours	Higher maintenance hours indicate elevated equipment stress and increased defect risk	Maintenance Management	Plan preventive maintenance schedule; reduce reactive maintenance load
DefectRate	Higher defect rate values strongly increase predicted defect risk (note: potential leakage variable)	Quality Management	Monitor production process parameters; consider leakage-controlled model for deployment

QualityScore	Lower process quality scores increase predicted defect risk	Quality Management	Increase quality-control inspection frequency and sampling rate
QualityInstabilityIndex	Higher combined quality instability (defect rate × low score) predicts quality failure	Quality Management	Increase inspection frequency; review process control charts
ProductionVolume	High production volumes amplify defect risk through increased operational pressure	Production Planning	Review production scheduling; balance throughput with quality capacity
MaintenanceStressIndex	Higher combined maintenance stress (downtime × hours) signals equipment reliability risk	Maintenance Management	Schedule immediate preventive maintenance inspection
ProductionPressureIndex	Higher combined production pressure (volume × cost) elevates defect risk	Production Planning	Review production schedule; evaluate capacity versus demand balance
DeliveryDelay	Higher delivery delay values increase predicted defect risk	Procurement / Logistics	Adjust sourcing plan; penalise delay-prone suppliers

Table A5. Classification threshold tuning results for the best model (Random Forest).

Strategy	**Threshold**	**Recall**	**Precision**	**F1-score**	**FNR**
Default (0.50)	0.5	0.989	0.9506	0.9694	0.011
Youden's J	0.48	0.9908	0.9507	0.9704	0.0092
F1-Maximisation	0.48	0.9908	0.9507	0.9704	0.0092

Note: Optimal threshold=0.48 identified by both Youden's J and F1-maximisation.