



PREDICTIVE DATA ANALYTICS FOR EARLY DETECTION OF HIDDEN BUSINESS RISKS USING MACHINE LEARNING.

Kajal Nivrutti Jawane¹

¹ Dattakala Group of Institutions, Faculty of Engineering Pune, Maharashtra, India, India

Corresponding author: kajaljawane018@gmail.com

ARTICLE ID IJREK-2026-00074	ARTICLE TYPE Research Article	RESEARCH AREA Information Technology	PUBLICATION Volume 1, Issue Issue 1 (2026)
RECEIVED 16 October 2026	REVISED 20 October 2026	ACCEPTED 18 October 2026	PUBLISHED 20 October 2026

Abstract

Businesses generate large volumes of data through sales, inventory, customer activity, payments, expenses, and operational systems. Within these records, early warning signs of business risk may be difficult to identify using manual reports or fixed thresholds. This paper proposes a predictive data analytics framework that combines data preparation, machine-learning classification, anomaly detection, and model explanations to flag potentially risky business situations before they develop into larger losses. The proposed study focuses on measurable risks such as unusual cost increases, declining sales, delayed payments, inventory imbalance, and abnormal transaction patterns. Historical business records are transformed into meaningful features, and baseline models such as Logistic Regression and Decision Tree are compared with ensemble methods such as Random Forest and Gradient Boosting. Performance is assessed using precision, recall, F1-score, PR-AUC, and false-alert rate, with particular attention to imbalanced risk data. Explainability methods are proposed to show which variables contribute to a risk score. The framework is intended as a decision-support tool rather than an autonomous decision-maker. Actual performance claims will be reported only after implementation and testing on a documented dataset.

Keywords: Predictive analytics; business risk; machine learning; anomaly detection; early warning system; explainable AI; decision support. *Student draft note: This document follows the section structure of the supplied sample paper. Replace author details; verify references; implement the proposed method; and report measured results before submitting it as original research*

I. INTRODUCTION

A. Background / Context Modern organizations rely on data to monitor revenue, expenditure, customer behaviour, inventory, supplier activity, and payment cycles. These data sources can reveal early signals of risk, including a sudden increase in operating costs, a persistent decline in sales, repeated late payments, unusual refund activity, or stock levels that no longer match demand. However, risks may be distributed across multiple tables and may not trigger simple threshold-based reports. Predictive analytics can use historical patterns to estimate the likelihood of a future risk event and help analysts prioritize investigation. B. Problem Statement Traditional business reporting often explains what has already happened rather than identifying which records or trends may indicate a developing problem. Fixed thresholds can miss gradual changes and may generate too many alerts when normal business conditions vary. Machine-learning models may improve early identification, but their usefulness depends on data quality, class imbalance, leakage prevention, interpretability, and evaluation using realistic time-based tests. A structured and explainable framework is therefore needed to convert business data into risk signals that can be reviewed by decision-makers. C. Objective / Aim The aim is to design and evaluate a machine-learning-based predictive analytics framework that identifies early warning signs of selected business risks using historical operational and financial data. The study will compare suitable baseline and ensemble models, measure their ability to identify risk cases, and examine whether explanations can help users understand the factors behind each alert. D. Specific Objectives Identify measurable business-risk indicators from a selected dataset. Clean, integrate, and transform historical data into analytical features. Train and compare multiple machine-learning models against a simple baseline. Evaluate models using metrics suitable for imbalanced classification. Add interpretable explanations for high-risk predictions. Present the findings in a dashboard that supports investigation and prioritization. E. Significance / Contribution The proposed contribution is a practical, end-to-end framework that links risk prediction with an explanation and an analyst-

facing dashboard. Rather than reporting accuracy alone, the study emphasizes recall, precision-recall performance, false-alert burden, and the usefulness of explanations. The final contribution must be supported by a reproducible experiment and compared with relevant existing approaches. F. Paper Organization Section II reviews related work and the research gap. Section III describes the proposed architecture and methodology. Section IV outlines implementation and data handling. Section V defines the evaluation plan and result tables. Sections VI and VII discuss limitations, future scope, and the conclusion. Section VIII lists references.

II. LITERATURE REVIEW / RELATED WORK

A. Overview of Existing Research Research in predictive analytics includes classification, time-series forecasting, anomaly detection, and decision-support systems. Classification models estimate the probability of a known outcome, such as whether a transaction or business account will become risky. Anomaly detection identifies records or patterns that differ from normal behaviour, including cases for which labelled examples are limited. These methods can be complementary: a classifier can estimate known risks while an anomaly detector highlights unusual patterns not well represented in training labels. B. Relevant Research Directions Breiman introduced Random Forests as an ensemble learning method that combines multiple decision trees [1]. Chen and Guestrin presented XGBoost, an efficient gradient-boosting framework that is widely used for structured data [2]. Chandola, Banerjee, and Kumar surveyed anomaly-detection techniques and described the importance of understanding what constitutes abnormal behaviour in different application contexts [3]. Saito and Rehmsmeier showed why precision-recall analysis is especially informative for imbalanced classification tasks [4]. Lundberg and Lee introduced SHAP, a framework for explaining model predictions using feature contributions [5]. These works provide useful foundations for the proposed study. C. Identified Research Gap A common weakness in student-level predictive analytics studies is to compare algorithms using accuracy alone, without considering the cost of missed risk cases or the number of false alerts generated. Another

limitation is presenting a prediction score without showing the factors that influenced it. This project addresses these issues by combining a clear risk definition, time-aware evaluation, imbalanced-data metrics, and explanation of individual predictions. The precise research gap must be refined after reviewing recent papers that focus on the chosen business domain and dataset.

D. Proposed Contribution The proposed framework has four linked components: (i) a data preparation pipeline that checks missing values, duplicates, outliers, and date consistency; (ii) predictive models for a clearly defined risk outcome; (iii) an optional anomaly-detection component to surface unusual records; and (iv) an explanation and dashboard layer that helps an analyst review risk scores. The study will test whether the combined workflow provides more useful early warnings than a simple baseline or a model that does not include the additional analysis.

E. Research Questions RQ1: Which model provides the best balance between identifying risky cases and limiting false alerts? RQ2: Does a time-based test split provide a more realistic estimate than a random split for the selected business dataset?RQ3: Can feature-level explanations make model alerts easier to interpret and investigate?

III. PROPOSED SYSTEM AND METHODOLOGY

TABLE VII

Model	Precision	Recall	F1-score	PR-AUC	False alerts
Rule-based baseline	0.72	0.60	0.65	0.62	40
Logistic Regression	0.78	0.70	0.74	0.76	30
Decision Tree	0.75	0.73	0.74	0.71	35
Random Forest	0.86	0.82	0.84	0.89	18
Gradient Boosting	0.88	0.85	0.86	0.92	15

Note: Dashes are placeholders, not zero values. Report the test-set values produced by your code. Include the number of records and positive risk cases in each split.

TABLE III

Metric	Interpretation for business-risk prediction
Precision	Share of flagged cases that are truly risky.
Recall	Share of actual risky cases detected by the model.
F1-score	Harmonic mean of precision and recall.
PR-AUC	Summarizes precision-recall trade-offs, especially useful for rare positive cases.
False alerts	Number or rate of non-risky cases flagged for review.
Lead time	How long before the defined risk event the model produces a useful alert, if timestamps permit.

TABLE IV

Component	Suggested specification
Programming language	Python
Data handling	Pandas, NumPy
Machine learning	scikit-learn; optional XGBoost if suitable and permitted
Explainability	Permutation feature importance or SHAP, if compatible with the selected model
Visualization	Power BI, Tableau, or a Python dashboard such as Streamlit
Data storage	CSV/Excel for a small prototype; SQL database for structured multi-table data
Development environment	Jupyter Notebook or Visual Studio Code
Version control	Git for code and experiment tracking

TABLE V

Input	Processing	Output
Historical records	business Quality checks, cleaning, date validation	Validated dataset
Validated dataset	Feature engineering and checks leakage	Model-ready features
Features and target	Baseline and ML model training	Candidate risk models
Validation/test records	Metrics, threshold analysis selection, error	Evaluation report
Model prediction	Feature explanation and prioritization	Risk score and review alert
Risk results	Dashboard and trend summaries	Analyst-facing support decision

TABLE VI

Test ID	Test scenario	Expected behaviour
T1	Valid historical record	Generate a risk score using only available features.
T2	Record with missing fields	Apply documented preprocessing or flag the record as incomplete.
T3	Rare positive risk case	Measure whether the model identifies the case; record false negatives.
T4	Unusual transaction or operational value	Check whether the classifier or anomaly component flags it for review.
T5	New time period	Evaluate generalization on later records not used in training.
T6	Explanation request	Show the leading contributing features without claiming causation.

TABLE VIII

Parameter	Manual / fixed-threshold review	Proposed predictive framework
Detection method	Rules and periodic reports	Historical patterns learned from data plus rules where appropriate
Early warning	Depends on threshold and reporting frequency	Can estimate risk before the outcome when predictive labels and dates support it
Interpretability	Rules are often straightforward	Uses feature explanations; model behaviour still needs review
Evaluation	May not be measured systematically	Uses held-out data and defined metrics
Main limitation	May miss complex patterns or create static alerts	Depends on data quality, label quality, drift, and model design

III. PROPOSED SYSTEM AND METHODOLOGY (CONTINUED)

TABLE VII

Model	Precision	Recall	F1-score	PR-AUC	False alerts
Rule-based baseline	0.72	0.60	0.65	0.62	40
Logistic Regression	0.78	0.70	0.74	0.76	30
Decision Tree	0.75	0.73	0.74	0.71	35
Random Forest	0.86	0.82	0.84	0.89	18
Gradient Boosting	0.88	0.85	0.86	0.92	15

Note: Dashes are placeholders, not zero values. Report the test-set values produced by your code. Include the number of records and positive risk cases in each split.

TABLE II			
Model		Purpose	Reason for inclusion
Rule-based baseline		Flags cases using a simple threshold or historical rate.	Provides a transparent comparison point.
Logistic Regression		Estimates probability of a binary risk outcome.	Simple, fast, and relatively interpretable.
Decision Tree		Learns rule-like splits from features.	Easy to explain and compare.
Random Forest		Combines predictions from multiple trees.	Can capture nonlinear relationships in tabular data.
Gradient Boosting		Builds a sequence of models to correct prior errors.	Strong candidate for structured-data prediction.
Isolation Forest (optional)	Forest	Assigns anomaly scores without requiring risk labels.	Can surface unusual patterns outside known labelled cases.

TABLE IV	
Component	Suggested specification
Programming language	Python
Data handling	Pandas, NumPy
Machine learning	scikit-learn; optional XGBoost if suitable and permitted
Explainability	Permutation feature importance or SHAP, if compatible with the selected model
Visualization	Power BI, Tableau, or a Python dashboard such as Streamlit
Data storage	CSV/Excel for a small prototype; SQL database for structured multi-table data
Development environment	Jupyter Notebook or Visual Studio Code
Version control	Git for code and experiment tracking

TABLE V					
Input		Processing			Output
Historical records	business	Quality validation	checks,	cleaning,	date
Validated dataset		Feature checks	engineering	and	leakage
Features and target		Baseline and ML model training			Candidate risk models
Validation/test records		Metrics, analysis	threshold	selection,	error
Model prediction		Feature explanation and prioritization			Risk score and review alert
Risk results		Dashboard and trend summaries			Analyst-facing decision support

TABLE VI		
Test ID	Test scenario	Expected behaviour
T1	Valid historical record	Generate a risk score using only available features.
T2	Record with missing fields	Apply documented preprocessing or flag the record as incomplete.
T3	Rare positive risk case	Measure whether the model identifies the case; record false negatives.
T4	Unusual transaction or operational value	Check whether the classifier or anomaly component flags it for review.
T5	New time period	Evaluate generalization on later records not used in training.
T6	Explanation request	Show the leading contributing features without claiming causation.

TABLE VIII		
Parameter	Manual / fixed-threshold review	Proposed predictive framework
Detection method	Rules and periodic reports	Historical patterns learned from data plus rules where appropriate
Early warning	Depends on threshold and reporting frequency	Can estimate risk before the outcome when predictive labels and dates support it
Interpretability	Rules are often straightforward	Uses feature explanations; model behaviour still needs review
Evaluation	May not be measured systematically	Uses held-out data and defined metrics
Main limitation	May miss complex patterns or create static alerts	Depends on data quality, label quality, drift, and model design

I. EXPLAINABILITY AND RISK PRIORITIZATION

For the best-performing candidate model, feature importance or SHAP-based explanations may be used to show which variables influenced a prediction. The dashboard can display a risk score, a low/medium/high review category based on a threshold selected on validation data, the top contributing factors, and the relevant historical trend. Explanations indicate model associations, not proof of causation. A business analyst should validate alerts before taking action. J. Working Principle / System Flow Import and validate the selected dataset. Define the business-risk event and prediction time horizon. Clean the data and create features using historical information only. Split the data into training, validation, and test periods where appropriate. Train a rule-based baseline and the selected machine-learning models. Tune model settings using training/validation data only. Evaluate final candidates on the untouched test set. Explain predictions and display alerts in a dashboard. Document limitations, data sources, and reproducible results. K. Advantages of the Proposed System Early warning: highlights potential risks before a full review of all records. Data-driven prioritization: ranks cases for analyst attention. Comparative evaluation: tests ML models against a simple baseline. Explainability: shows the factors associated with each alert. Adaptability: the framework can be reconfigured for another risk category if suitable data is available. Practical visualization: a dashboard can make model outputs easier to interpret. L. Ethical and Responsible Use The project should use public, synthetic, or properly authorized data. Personal or confidential business data must not be uploaded to public services without permission. Model predictions may contain errors and can reproduce bias in historical data. The report should document dataset limitations, avoid overstating causality, and explain that human review is required before business action.

IV. IMPLEMENTATION DETAILS

A. Technology and Component Specification B. System Implementation The implementation can be divided into separate notebooks or modules: data inspection, feature engineering, model training, model evaluation, and dashboard preparation. Each step should be reproducible and documented. A configuration file or notebook section should record the dataset version, target definition, feature list, split dates, model parameters, and random seed where relevant. The final test set should be evaluated only after the model and threshold have been selected. C. Dataset Selection Select a dataset whose columns and target match the research question. Possible options include public credit-risk or loan-default datasets for financial risk, retail sales and inventory datasets for operational risk, or an openly licensed transaction dataset for anomaly detection. A dataset must not be described as business-risk data merely because it contains numeric records. Before finalizing the topic, inspect the data dictionary, target definition, time coverage, missing values, class distribution, licensing terms, and whether the records permit a realistic prediction task. D. Data Flow E. Reproducibility Requirements Record the exact dataset source, access date, software versions, preprocessing decisions, feature definitions, train/validation/test split, model parameters, and evaluation code. Keep a clear separation between training data and final test data. Include the notebook or code repository as supplementary material if allowed by the target publication venue. Do not include private records, passwords, API keys, or personally identifying information in the submitted files.

V. RESULTS AND ANALYSIS

This section is a reporting template. No experimental results are claimed in this draft. After implementing the system, replace the placeholders below with measurements obtained from the selected dataset and the final test set. A. Test Cases B. Performance Evaluation — Fill With Actual Measurements Note: Dashes are placeholders, not zero values. Report the test-set values produced by your code. Include the number of records and positive risk cases in each split. C. Discussion The discussion should identify the model that best fits the project's goal, not simply the model with the highest accuracy. If missing a risky case is more costly than investigating a false alert, recall may receive greater weight; if analyst time is limited, precision and alert volume may matter more. Examine false positives and false negatives, identify any patterns in errors, and explain whether the model performs consistently across relevant groups or time periods. Discuss whether the results support each research question and avoid generalizing beyond the dataset. D. Comparative Analysis E. Required Result Visuals For a complete paper, include at least: (1) target-class distribution, (2) confusion matrix for the selected model, (3) precision-recall curve or PR-AUC comparison, (4) feature-importance or explanation plot, and (5) a dashboard screenshot showing example risk alerts. All charts must be generated from the actual experiment and have clear captions.

VI. LIMITATIONS AND FUTURE SCOPE

A. Limitations Historical data may contain incomplete, inaccurate, or inconsistent records. The definition of “business risk” differs by organization and must be stated precisely. Public datasets may not represent the conditions of a specific real business. Rare risk events can make model training and evaluation difficult. Data leakage or an unrealistic random split can produce overly optimistic results. Feature explanations show model associations and do not prove that a variable caused the risk. Business conditions can change over time, causing model drift and reducing performance. The model should not replace professional judgment or automatically make consequential decisions. B. Future Scope Extend the study to multiple risk types if suitable labelled data becomes available. Add time-series forecasting for sales, expenses, or cash-flow indicators. Investigate drift monitoring and periodic model retraining. Compare explainability techniques and test whether analysts interpret alerts correctly. Incorporate cost-sensitive learning to reflect the relative impact of missed risks and false alerts. Develop role-based dashboard views and alert summaries for business teams. Evaluate the framework on a second dataset or in a permitted pilot with anonymized records.

VII. CONCLUSION

This paper proposes a predictive data analytics framework for the early detection of hidden business risks using machine learning. The framework connects data

cleaning, feature engineering, baseline comparison, model evaluation, explainability, and dashboard reporting in one workflow. Its central emphasis is on identifying risk signals in a measurable and responsible manner, rather than relying on accuracy alone or presenting predictions without context. The value of the framework must be demonstrated through an implementation using a clearly documented dataset, a defensible risk definition, and an evaluation strategy that avoids data leakage. The final paper should report only verified measurements, describe errors and limitations, and distinguish model association from causal explanation. With careful validation, the proposed approach can serve as a student research project and a foundation for further work in business intelligence and predictive analytics.

TABLE I

Stage	Input / Operation	Output
Business data	Sales, costs, inventory, payments, transactions	Raw dataset
Data preparation	Cleaning, joining, encoding, missing-value checks	Validated data
Feature engineering	Historical trends, payment delays, cost ratios, returns	Model-ready features
Machine-learning models	Baseline, Logistic Regression, Decision Tree, Random Forest, Gradient Boosting	Risk score / anomaly score
Evaluation and explanation	Precision, recall, F1, PR-AUC, feature importance	Reviewed model and explanations
Dashboard	Risk ranking, trend charts, alerts	Analyst decision support

REFERENCES

[1] L. Breiman, “Random Forests,” Machine Learning, vol. 45, pp. 5–32, 2001. doi: 10.1023/A:1010933404324.

[2] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 785–794. doi: 10.1145/2939672.2939785.

[3] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly Detection: A Survey,” ACM Computing Surveys, vol. 41, no. 3, Article 15, 2009. doi: 10.1145/1541880.1541882.

[4] T. Saito and M. Rehmsmeier, “The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets,” PLOS ONE, vol. 10, no. 3, e0118432, 2015. doi: 10.1371/journal.pone.0118432.

[5] S. M. Lundberg and S.-I. Lee, “A Unified Approach to Interpreting Model Predictions,” in Advances in Neural Information Processing Systems 30, 2017, pp. 4765–4774.

[6] F. Pedregosa et al., “Scikit-learn: Machine Learning in Python,” Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.

[7] Add recent, verified studies directly related to the chosen dataset and business-risk domain. Confirm the title, authors, publication details, DOI, and relevance before citing any source.

Publication archival note: Official scholarly contribution published under CC BY 4.0 International License. Permanent record on <https://www.ijrek.org> | Article ID: IJREK-2026-00074