

# Title Page

## Title of the Manuscript:

*Stored and Non-Stored Rules in Bankruptcy Prediction: A*

*Decision Tree Analysis of the Altman Z-Score*

## Authors and Affiliations:

**Tarik El Moudden<sup>a</sup> (Corresponding author)**

Computer Science Research Laboratory

Faculty of Sciences Kenitra

Ibn Tofail University, Kenitra, Morocco

E-mail : [tarik.elmoudden@uit.ac.ma](mailto:tarik.elmoudden@uit.ac.ma)

Phone : +212-6584 81 848

Postal Adress : Résidence Adam Apt 1, Maghrib Arabi, Ex 4 et 5 Bloc D, Kenitra, Morocco

**Adra Jaafar**

National School of Business and Management,

Abdelmalek Essaadi University, Tanger, Morocco

E-mail : [adrajaafaralaoui@gmail.com](mailto:adrajaafaralaoui@gmail.com)

Phone: +212-664 417147

**Bahaa Eddine Elbaghazaoui**

National School of Applied Sciences

Sultan Moulay Slimane University, Beni Mellal, Morocco

E-mail : [b.elbaghazaoui@usms.ac.ma](mailto:b.elbaghazaoui@usms.ac.ma)

Phone: +212-622142475

**Zakaria Bennani**

National School of Business and Management,

Abdelmalek Essaadi University, Tanger, Morocco

E-mail : [bzakaria@uae.ac.ma](mailto:bzakaria@uae.ac.ma)

Phone: +212-614 954348

# Stored and Non-Stored Rules in Bankruptcy Prediction: A Decision Tree Analysis of the Altman Z-Score

---

## Abstract

This study explores the reasons behind the limitations of the classical Altman Z-score model in bankruptcy prediction. The simplicity, linearity, and fixed weights of the Z-score obstruct its capacity to capture the nonlinearity inside the dataset. Moreover, our hybrid framework combines the explainability of decision trees with the financial grounding of Altman's Z-score, revealing that several decision rules remain "non-stored" in the Altman formulation.

The study first adopts the CRISP-DM methodology, combining domain knowledge with data-driven techniques. Using a large dataset of 78,682 firm-year observations from companies listed on the NYSE and NASDAQ (1999–2018) and balancing with SMOTE/ADASYN, followed by modelling with decision tree classifiers and systematic hyperparameter tuning via grid search to discover stored or non-stored rules in Altman formulation. Finally, by embedding these rules as correction layers or adaptive weights, we can modernise Altman into a hybrid, interpreted, and more accurate bankruptcy prediction tool.

*Keywords:* Bankruptcy, Altman Z-Score, Rule-Based Decision Tree, Human-understandable rules, Missing decision rules

---

## 1. Introduction

Predicting when a company will go bankrupt is a very significant part of finance, accounting, and risk management because it has a lot of impacts. When a business goes bankrupt, it impacts more than just the owners, shareholders, creditors, and employees. It also harms the economy of the whole country by making the financial system less reliable. for example, getting financial support from the government [1]. The well-known disasters of Enron and Lehman Brothers, as well as smaller, less well-known bankruptcies, illustrate that there are clear flaws with how we check for financial soundness. These examples highlight how vital it is to build better models for seeing indicators of financial difficulties early on. This will help you guess when bankruptcy is likely to happen and keep a watch on bankruptcy grifters [2].

Scholarly interest in this field commenced in the late 1960s, particularly with Edward Altman, who presented the Z-score model in 1968. The model used a linear combination of financial ratios to give a quantitative way to tell if a company was likely to fail or survive. Its clarity and transparency made it widely used in assessing company stability. Later, a logistic regression model was suggested by Ohlson (1980), and Zmijewski (1984) used a probit-based specification. Both models used a wider range of financial indicators that were based on

September 4, 2025

probabilities. These models established the foundation for numerous decades of empirical research in forecasting financial distress [3].

In addition, The Altman Z-score model is a well-known linear function for predicting bankruptcy. It uses five inputs to classify companies into three classes: without risk: alive, unclear risk, or bankrupted: failed. These early models were useful, but they often made assumptions that don't hold up in real-world financial data, like that variables are normal or that relationships are linear. In addition, they were usually made and tested on small, similar datasets. As the financial world got more complicated, with changing accounting rules, markets becoming more global, and firms becoming more different from each other, it became clearer that we needed models that could show non-linear, dynamic relationships [4].

Unfortunately, the Altman Z-score reveals important limitations in both binary classification (alive vs. failed) and three-class classification (without risk: alive, unclear risk, or at risk: failed). These limitations are largely due to the linear nature of the formula, which relies on fixed weights, which may not adequately capture the heterogeneity of modern firms. Moreover, five financial ratios derived from empirical studies were incorporated. These five variables ignore potentially informative indicators such as cash flow measures, market sentiment, or macroeconomic conditions. Therefore, the ideas to enhance the Altman Z-score model by re-estimating the weights through modern optimisation techniques or by incorporating additional variables tailored to contemporary datasets and contexts seem very important.

Explainable artificial intelligence (XAI) methods provide a promising way to understand these limitations better. Specifically, rule extraction methods like decision trees can help query the dataset very quickly and explain after the fact why the Altman Z-score gets some companies wrong [5] [6]. These methods close the gap between predictive accuracy and interpretability by making clear the decision rules and conditions that lead to success or failure of the model. This analysis approach to the Altman Z-score using if-then rules extracted by decision trees not only shows the flaws in the traditional Altman method, but it also suggests ways to make financial risk assessment models that are more flexible, strong, and clear.

This study is organised: The "Related Work" section cites much scientific work on predicting bankruptcy using both old statistical models and AI algorithms. The next section, "Material and Methods", adopts the CRISP-DM framework to extract significant rules. In the "Results and Discussion" section, we approach Altman's Z-score classification with hyperparameter-tuned decision tree classification. The last section, "Conclusions," sums up the main points, talks about the benefits of combining explainable AI with financial theory, and suggests ways to make bankruptcy prediction better in the future.

## 2. Related work

AI-based models offer high accuracy in bankruptcy prediction, but their decisions are less understandable than rule-based models. The rules generated by decision trees make them suitable for expert-managed situations, such as auditing or credit assessment. Therefore,

September 4, 2025

harnessing the predictive power of advanced AI models with clear interpretable frameworks is a valuable idea for future research.

### *2.1. Bankruptcy Prediction: Modern AI-Based Models*

Modern AI-based models have revolutionised how scientists in several fields make predictions. Researchers can make more generalisable models that work better than traditional models since these models rely on data sources. This is especially true when it comes to predicting bankruptcy, where performance has gotten a lot better.

Artificial neural networks (ANNs) have a strong ability to capture nonlinear relationships, achieving higher accuracy than logistic models in various studies. For example, Data from 148 credit institutions was trained using multilayer perceptron (MLP) neural networks to make a predictive model that could predict bankruptcies in the Spanish banking system in the short term. The model showed an overall accuracy of more than 97% when performance was tested with a train-test split and holdout sampling. The input variables were derived from the CAMELS framework as an explanatory features matrix, supplemented by additional macroeconomic variables that directly influence the characteristics of the banks' overall economic condition [7]. However, their lack of transparency, high data requirements, and difficulty in understanding ANNs represent major problems.

A comparative study of classical Linear Discriminant Analysis (LDA) and Support Vector Machines (SVM) was performed to forecast bankruptcy for Polish small- and medium-sized firms (SMEs). The research employed a dataset of 806 enterprises (311 bankrupt and 495 solvent), integrating financial statistics with specific non-financial characteristics. It indicated that SVM was 72.1% accurate overall, while LDA was 63.5% accurate. Moreover, The experimental results showed that LDA was better at finding troubled companies than SVM, with an accuracy of 66.3% compared to 35.8% for SVM. This shows that SVM and more complex ML algorithms are better at overall classification, but they don't work well for early detection of bankruptcy. LDA, on the other hand, are better for early warning systems [8].

### *2.2. Interpretable Rule-Based Models and Decision Trees*

A rule-based model (RBM) that generates IF–THEN rules was proposed, that can be read by experts to tell if a transaction is fraudulent or legitimate. This framework makes it easier to understand than decision-making in machine learning, specifically when dealing with imbalanced data. The process starts first with automatic-iterative feature selection, which uses RF to pick the most important 80% of features and Decision Tree to improve the other 20%. This decreases the original PaySim dataset features FROM 11 to 9 to preserve only pertinent variables. Second, unsupervised k-means clustering is carried out to segment fraudulent transactions into categories using CASH IN, CREDIT, CASH OUT, TRANSFER, DEBIT, and PAYMENT. Third, the model identifies consecutive value ranges to create range-based rule terms. Finally, these rules are grouped into association rules, removing ones with lower support and confidence levels. The results of the new framework rule-based model (RBM) show that it beat all the baselines, with an accuracy of 0.99 and a precision of 0.99 [9].

September 4, 2025

In various fields, such as medical diagnosis, legal judgement, chemical process, or financial decision-making, rule induction is used to identify if-then rules within data sets. These rules are typically built on decision trees or rule-based models. Specifically, in banking sectors, credit scoring and rule induction are closely linked, particularly in the banking sector. This approach has shown success in protecting and enhancing the efficiency of credit decisions while increasing their transparency. These decisions are based on analysing historical documents containing debt data and then creating predictive models that include debt repayment, which encourages banking institutions to take more risks [10].

Furthermore, in terms of interpretability perspective, the usefulness of a model in real life is based on criteria like reliability, causality, and transparency. Reliability is linked to decision tree stability, meaning suggested rules should remain constant despite data changes. The Adaptive NLS (ANLS) process, which employs a statistical test to isolate the predictor, is actually more important. The assessment was examined using CART and ANLS to model each stage on the UCI datasets, which gave us four rule sets: CART-70, CART-100, ANLS-70, and ANLS-100. The breast cancer-Wisconsin dataset from the UCI repository showed that ANLS-generated trees produced more stable rules when the data was updated than CART. This greatly improved the model's reliability [11].

### 3. Material and Methods

This section represents the core of our study. The pipeline based on Cross-Industry Standard Process for Data Mining (CRISP-DM) methodology is adopted, as mentioned in Fig. 1 [12]. It begins with business understanding which integrates common knowledge of the domain [13], data understanding before proceeding to data preparation, modelling, evaluation, and rules interpretation [14]. The original financial dataset of 78,682 firm-year observations from 8,262 companies listed on the NYSE and NASDAQ (1999–2018) was first explored to identify relevant attributes and clarify the business objective of bankruptcy prediction. Then, eight original financial indicators were selected to construct derived ratios (Alt1–Alt5), which will be used later to compare the Altman Z-Score and decision tree classification to extract the rules that are stored or non-stored in the Altman Z-Score classification.

#### 3.1. Business and Data Understanding

To interpret the Altman Z-Score through decision tree rules, our study uses a dataset of 78,682 firm-year observations from 8,262 companies that were listed on the NYSE and NASDAQ between 1999 and 2018<sup>1</sup>. In the dataset, a company is considered bankrupt (1) in the fiscal year before it files for bankruptcy, and all other company-year observations are considered non-bankrupt (0). We choose eight important financial variables from the full set of numeric X1 to X18 indicators, as mentioned in Tbl. 1. These original features helped us to create new composite features based on Altman's framework.

---

<sup>1</sup><https://www.kaggle.com/datasets/utkarshx27/american-companies-bankruptcy-prediction-dataset>

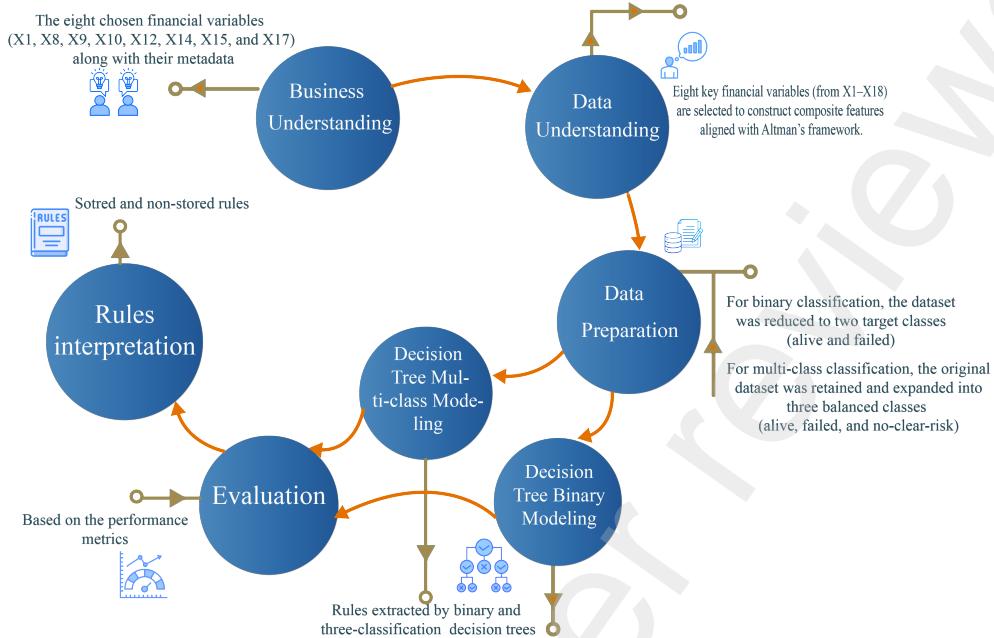


Figure 1: Our CRISP-DM-based workflow for bankruptcy prediction.

Variables	Metadata
X1	Current Assets — things that are expected to be sold or used within a year.
X8	Market Value — for public companies, this is shown by their market capitalisation.
X9	Net Sales — the total amount of money made from sales after returns, allowances, and discounts.
X10	Total Assets — the total value of all the things a company owns.
X12	EBIT — earnings before interest and taxes, a way to measure how profitable a business is.
X14	Total Current Liabilities — includes short-term debts like accounts payable, accrued expenses, and taxes.
X15	Retained Earnings — profits that have been kept after paying out dividends and expenses.
X17	Total Liabilities — all of the debts and obligations owed to people outside of the company.

Table 1: Selected financial variables and their metadata.

September 4, 2025

By focusing only on these financial metrics, we intend to formulate derived ratios (Alt1–Alt5) that reflect liquidity, profitability, solvency, and efficiency. These engineered features, when used with decision tree models, make a framework that is more flexible and interpretable of rules hidden in the traditional Z-score model.

### 3.2. Data Preparation

1. **For the binary classification subsection:** The dataset was first filtered so that it only had two target categories: alive and failed. The problem of class imbalance was fixed using oversampling methods like SMOTE and ADASYN to make synthetic minority samples while keeping the feature space distribution the same [15] [16]. The outcome yielded 73462 instances of 'alive' and 73462 instances of 'failed'. Then, the financial predictors were put together to make the Altman Z-score, which uses a different decision rule that removes later the middle zone "unclear risk" to compare its performance metrics against the decision tree of binary states, as mentioned in the code. 1.
2. **For three Multi-class Classification subsection:** The original dataset is kept, and three new classes will be added: 73462 instances of class alive, 73462 instances of class failed, and 73462 occurrences of class NO-CLEAR-RISK. We used SMOTE or ADASYN to handle these classes, which generated synthetic instances. The new class NO-CLEAR-RISK is an example of this; its distribution is between the alive and failed classes. The Altman model was used to calculate the Z-score again, but the original thresholding procedures were kept to tell the three classes apart. occurrences, as mentioned in the code. 1.

code1

```

1      z1=2.99
2      z2 = 1.99
3      Altman_data[ 'Z_score' ] = (w1 * Altman_data[ 'Alt1' ]
4                                    +
5                                    w2 * Altman_data[ 'Alt2' ] + w3 * Altman_data[ 'Alt3' ]
6                                    +
7                                    w4 * Altman_data[ 'Alt4' ] + w5 * Altman_data[ 'Alt5' ]
8      )
9
10     def classify_z(z):
11         if z > z1:
12             return 'alive'
13         elif z > z2:
14             return 'NO-CLEAR-RISK'
15         else:
16             return 'failed'
```

September 4, 2025

Code 1: Binary classification with Altman Z-score

### 3.3. Binary vs Multi-class Classification Modeling

The binary approach aims to study the stored and non-stored rules in the Altman Z-score model when used to classify companies into two classes: alive or failed, based on five financial variables (Alt1–Alt5). The process begins by executing code. 1 to generate a new predictor variable, then all records classified as "NO-CLEAR-RISK" are removed to assess the Altman Z-score in the case of binary classification [17]. A decision tree is then constructed with systematically adjusted hyperparameters by grid search algorithm to obtain the minimal accuracy, approximately to the Altman Z-score [18] [19].

Similarly, the Altman Z-score is applied as it is to classify companies into three categories—failed, alive, and NO-CLEAR-RISK—based on five variables (Alt1–Alt5). The procedure begins by executing Code 1 to generate a new predictive variable while retaining all instances labelled as unclear in order to assess the full accuracy of the Altman Z-score. Subsequently, a decision tree is constructed with systematic hyperparameter tuning, yielding accuracy results comparable to the Altman Z-score when classifying companies into the same three categories (failed, alive, and NO-CLEAR-RISK).

Therefore, the human decision rules extracted by decision trees allow the extraction of clear "if-then" conditions to help support decision-making by financial managers and analysts. These rules are stored or non-stored in the traditional Altman formulation in the case of binary, or multi classification, helping to improve predictive power and expand interpretability.

### 3.4. Performance Metrics

In this study, the precision metric measures how many of the predicted positive cases are correct. The recall metric measures how many of the actual positive cases are correctly identified by the model, the F1-score metric combines precision and recall into one number by taking their harmonic mean to balance both. Support measures how many true cases of each class are in the dataset [20], which is useful for judging how well each class is doing, as mentioned in Tbl. 2.

Metric	Formula
Precision	$= \frac{TP}{TP+FP}$
Recall	$= \frac{TP}{TP+FN}$
F1-score	$= 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$
Support	Number of true instances for each class in the dataset

Table 2: Formulas of classification metrics.

## 4. Results and Discussions

This section presents the results of applying decision tree classifiers alongside the Altman Z-score model to investigate stored and non-stored rules in bankruptcy prediction. The results

September 4, 2025

are examined in two stages: first, the binary classification of firms into alive and failed, and second, the multi-class classification that includes the additional no-clear-risk category. In both cases, the decision tree not only reproduced the predictive capacity of the Altman Z-score but also revealed new interpretable rules and nonlinear interactions that extend the classical model's insights.

#### 4.1. Rule Extraction in Binary Classification

The primary aim of this research is to examine the binary classification of companies as either “alive” (non-bankrupt) or “failed” (bankrupt) by contrasting the conventional Altman Z-score model with a data-driven Decision Tree Classifier. The Altman z-score model, derived heuristically and scientifically from financial theory, uses predetermined weights for ratios (Alt1–Alt5) to make a score that predicts the future. The decision tree, on the other hand, uses learnt thresholds from the data to find nonlinear interactions that the Altman Z-score formula might not show. To make sure the results could be compared, the decision tree was set up with only a few changes to the hyperparameters (criterion = entropy, max\_depth = 2, max\_features = 5, and balanced leaf splits) to get the same level of accuracy as the Altman Z-score classification while also making its rules clear.

We first used classical performance metrics to compare how well the two methods worked. The accuracy for the Altman Z-score classification was 0.616, as shown in first Tbl. 3. The results show that the model was able to correctly identify many failed companies, but it also incorrectly labelled many alive companies. The decision tree classifier, on the other hand, had a slightly higher accuracy of 0.632, as shown in in seconde Tbl. 3. The tree had a better recall for failed companies (0.68), which means it is better at finding bankruptcy cases, even though it lost some recall for alive companies. The fact that the two models performed similarly shows that the decision tree can reproduce the Altman model's ability to make predictions while also revealing new structural rule insights.

Table 3: Comparison of Classification Reports: Altman Z-score vs Decision Tree

Altman Z-score Predictions				
Class	Precision	Recall	F1-score	Support
Alive	0.6291	0.5586	0.5917	62856
Failed	0.6057	0.6731	0.6376	63316
Accuracy				0.6160
Macro Avg	0.6174	0.6158	0.6147	126172
Weighted Avg	0.6173	0.6160	0.6147	126172
Decision Tree Classifier				
Class	Precision	Recall	F1-score	Support
Alive	0.65	0.58	0.61	14693
Failed	0.62	0.68	0.65	14692
Accuracy				0.63
Macro Avg	0.63	0.63	0.63	29385
Weighted Avg	0.63	0.63	0.63	29385

Fig. 2 shows the rules that were taken from the decision tree. These rules provide a clear and

September 4, 2025

easy-to-understand interpretation of decision boundaries. Four important rules were made using the ratios Alt3 (earnings stability) and Alt4 (liquidity). For instance, businesses with  $Alt4 \leq 0.4907$  were correctly labelled as failed, while businesses with  $Alt4 > 0.9273$  and  $Alt3 > -0.1186$  were correctly labelled as alive. Other rules stress riskier situations in which even companies with a lot of cash on hand can be considered failed if their retained earnings (Alt3) are very negative. These rules show how the tree was able to attain close to the Altman classification by changing its hyperparameters and at the same time show rules that are still not stored implicitly in the Altman Z-score formula.

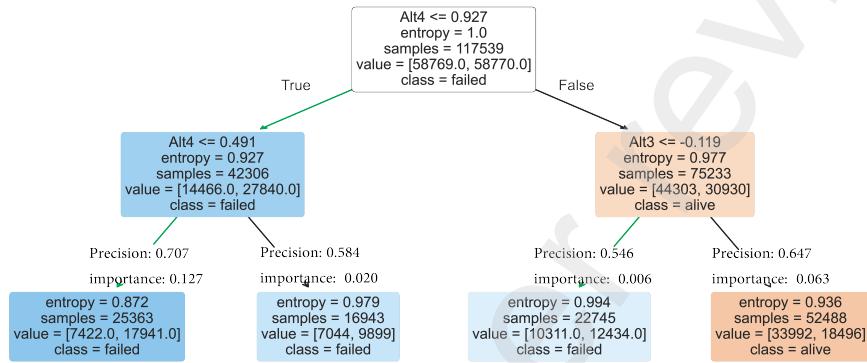


Figure 2: The Decision Tree generated four interpretable rules for classifying firms as alive or failed. The most influential splits are based on Alt4 (Market Value of Equity/Total Liabilities) and Alt3 (EBIT/Total Assets).

From a methodological standpoint, the stored rules of the Altman Z-score exemplify the linear weighting of ratios that are constant and grounded in theory. The decision tree, on the other hand, makes the hidden rules clear by adding nonlinear thresholds and interactions. The Altman model would always see liquidity (Alt4) as a good thing, but the tree shows that its effect depends on whether earnings stability (Alt3) is good or bad. The decision tree has small exceptions and conditional pathways that classical Altman Z-score models don't have. This is because the hyperparameters were changed to closely match Altman's predictive accuracy. This shows how important it is to use both machine learning models and financial heuristics to make things easier to understand.

#### *4.2. Rule Extraction in Multi Classification*

The standard Altman Z-score failed to detect the difference between the three groups: alive (not bankrupt), failed (bankrupt), and NO-CLEAR-RISK. The first Tbl. 4 shows that the precision and recall values are poor for all classes, and the overall accuracy is 41.45%. The macro and weighted averages show that the model has trouble applying to situations other than bankruptcy since the fixed ratios and thresholds don't account for the differences in financial characteristics between companies. By contrast, the decision tree classifier, tuned with a grid search for optimal parameters (`max_depth=8, criterion=gini`), produced a significant improvement compared to the classical Altman Z-score model. According to second Tbl. 4, the decision tree reached an accuracy of 50.7%. Specifically, the outcomes across alive, failed, and NO-CLEAR-RISK were fairly balanced, whereas the Altman Z-score favoured failed cases.

Table 4: Comparison of Classification Reports: Altman Z-score vs Decision Tree

Altman Z-score (Multi-class)				
Class	Precision	Recall	F1-score	Support
NO-CLEAR-RISK	0.3963	0.1854	0.2527	73462
alive	0.4266	0.4779	0.4508	73462
failed	0.4109	0.5801	0.4811	73462
Accuracy			0.4145	
Macro Avg	0.4113	0.4145	0.3948	220386
Weighted Avg	0.4113	0.4145	0.3948	220386
Decision Tree (Multi-class)				
Class	Precision	Recall	F1-score	Support
NO-CLEAR-RISK	0.4935	0.5319	0.5119	14692
alive	0.4835	0.4647	0.4739	14693
failed	0.5455	0.5243	0.5347	14693
Accuracy			0.5070	44078
Macro Avg	0.5075	0.5070	0.5069	44078
Weighted Avg	0.5075	0.5070	0.5069	44078

A strict precision threshold of 1.0 was applied to force the reliability of rules. This choice guarantees that only rules which perfectly classify the samples they cover are retained, eliminating the risk of misclassification within those subsets. While such a threshold inevitably

September 4, 2025

reduces the number of rules—49 in this study—it strengthens interpretability and trustworthiness, as every selected rule is error-free in the training data. This is particularly important in financial distress prediction, where false positives (e.g., wrongly labelling an “alive” firm as “failed”) can have significant consequences.

These filtered rules provide a detailed picture of how the financial ratios (Alt1–Alt5) are related to the predicted class labels. For example, Rule 2 says that if Alt2 and Alt1 are very different from each other, the prediction is “failed”. This rule can be formally expressed as:

```
IF Alt4 <= 0.567607 AND Alt4 <= 0.142058 AND Alt2 <= -3.63447 AND
Alt1 <= -28.2485 AND Alt4 <= 0.000402656 THEN predicted_class =
failed
```

On the other hand, if Alt4 and Alt5 are only slightly different from each other, the prediction is “alive” or “NO-CLEAR-RISK”. These results show that decision trees can store rules that are clear and directly related to financial indicators.

An important aspect of this study is investigating the difference between stored and non-stored patterns in the Altman Z-score classification. The stored rules show situations that are similar to the Z-score cut-offs. For example, a high Alt4 (leverage ratio) is linked to risk. Many non-stored rules come to light, though, especially when Alt3 and Alt5 interact in ways that weren’t directly included in Altman’s static formula. For instance, several rules demonstrate that combinations of Alt3 (earnings ratios) and Alt5 (sales or market value ratios) alter the classification between “alive” and “failed”. This shows that decision trees can identify nonlinear interactions that weren’t in the original model.

Table 5: Examples of Stored and Non-stored Rules in Decision Tree Classification

Rule Example	Predicted Class	Type of Rule	Explanation
IF Alt4 <= 0.567607 AND Alt4 <= 0.142058 AND Alt2 <= -3.63447 AND Alt1 <= -28.2485	Failed	Stored	Captures extreme leverage and asset deterioration similar to the original Z-score bankruptcy boundary.
IF Alt4 <= 0.567607 AND Alt2 > -3.63447 AND Alt3 > -0.0109387 AND Alt1 > 0.115661 AND Alt5 <= 0.276542	Failed	Non-stored	Involves Alt3 (earnings ratio) and Alt5 (market value/sales) interactions not explicitly modeled in Altman’s formula.
IF Alt4 > 0.567607 AND Alt3 <= -0.33972 AND Alt5 > 2.92737 AND Alt2 > -28.9224	Alive	Non-stored	Highlights nonlinear combinations of profitability (Alt3) and market valuation (Alt5) absent in the original Z-score.

## 5. Conclusions

The Altman Z-score has been useful for predicting bankruptcy in the past, but it doesn't work as well for today's diverse companies. It can't capture complicated financial situations because it only uses a small set of financial ratios (Alt1–Alt5) and fixed thresholds. This leads to low precision, recall, and systematic misclassifications, especially when there are more than one class, like alive, failed, and NO-CLEAR-RISK. These flaws show how rigid the Z-score is and how it can't show nonlinear interactions between financial indicators. This makes it less useful for prediction and understanding.

This study tackled these challenges by utilising decision tree classifiers with rule extraction. The method not only replicated the Z-score's predictive power, but also improved it by finding both stored rules that fit Altman's model and new rules that capture new interactions, especially between Alt3 and Alt5. The method achieved balanced performance while ensuring the reliability of retained rules by carefully tuning hyperparameters and setting a strict precision threshold of 1.0. In general, combining interpretable machine learning with the Altman framework makes for a more powerful, clear, and adaptable tool for analysing financial distress. This is useful for managers, auditors, and policymakers.

In the future work, we will focus to include new features and adaptive weight into the Altman Z-score. This will help to create a new hybrid version of Altman using artificial neural networks. Such a model can find adaptive weight according to the kind of dataset.

## References

- [1] A. Feibelman, Bankruptcy and the state, *Emory Bankr. Dev. J.* 38 (2022) 1.
- [2] L. D. Simon, Bankruptcy grifters, *Yale LJ* 131 (2021) 1154.
- [3] F. Rashid, R. A. Khan, I. H. Qureshi, A comprehensive review of the altman z-score model across industries, Available at SSRN 5044057 (2023).
- [4] D. R. Martins, Evaluating financial distress in portuguese firms: revisiting altman's z-score model (2024).
- [5] G. Audemard, S. Bellart, L. Bounia, F. Koriche, J.-M. Lagniez, P. Marquis, On the computational intelligibility of boolean classifiers, arXiv preprint arXiv:2104.06172 (2021).
- [6] L. Yao, Retracted: Financial accounting intelligence management of internet of things enterprises based on data mining algorithm, *Journal of Intelligent & Fuzzy Systems* 37 (5) (2019) 5915–5923.
- [7] J. Paule-Vianez, M. Gutiérrez-Fernández, J. L. Coca-Pérez, Prediction of financial distress in the spanish banking system: An application using artificial neural networks, *Applied Economic Analysis* 28 (82) (2020) 69–87.
- [8] A. Ptak-Chmielewska, Bankruptcy prediction of small-and medium-sized enterprises in poland based on the lda and svm methods, *Statistics in Transition. New Series* 22 (1) (2021) 179–195.
- [9] S. Islam, M. M. Haque, A. N. M. R. Karim, A rule-based machine learning model for financial fraud detection., *International Journal of Electrical & Computer Engineering* (2088-8708) 14 (1) (2024).
- [10] B. Y. Almansour, A. Y. Almansour, J. I. Janjua, M. Zahid, T. Abbas, Application of machine learning and rule induction in various sectors, in: 2024 International Conference on Decision Aid Sciences and Applications (DASA), IEEE, 2024, pp. 1–8.
- [11] J. Lee, M. K. Sim, J.-S. Hong, Assessing decision tree stability: a comprehensive method for generating a stable decision tree, *IEEE Access* 12 (2024) 90061–90072.
- [12] C. Schröer, F. Kruse, J. M. Gómez, A systematic literature review on applying crisp-dm process model, *Procedia computer science* 181 (2021) 526–534.

September 4, 2025

- [13] Q. V. Do, J. Li, T.-D. Vuong, Z. Wang, Y. Song, X. Ma, What really is commonsense knowledge?, arXiv preprint arXiv:2411.03964 (2024).
- [14] M. T. Ashraf, K. Dey, S. Mishra, M. T. Rahman, Extracting rules from autonomous-vehicle-involved crashes by applying decision tree and association rule methods, *Transportation research record* 2675 (11) (2021) 522–533.
- [15] P. Kumar, R. Bhatnagar, K. Gaur, A. Bhatnagar, Classification of imbalanced data: review of methods and applications, in: IOP conference series: materials science and engineering, Vol. 1099, IOP Publishing, 2021, p. 012077.
- [16] J. Brandt, E. Lanzén, A comparative review of smote and adasyn in imbalanced data classification (2021).
- [17] Z. Wang, L. Xia, H. Yuan, R. S. Srinivasan, X. Song, Principles, research status, and prospects of feature engineering for data-driven building energy prediction: A comprehensive review, *Journal of Building Engineering* 58 (2022) 105028.
- [18] C. Arnold, L. Biedebach, A. Küpfer, M. Neunhoeffer, The role of hyperparameters in machine learning models and how to tune them, *Political Science Research and Methods* 12 (4) (2024) 841–848.
- [19] M. Y. Shams, A. M. Elshewwy, E.-S. M. El-Kenawy, A. Ibrahim, F. M. Talaat, Z. Tarek, Water quality prediction using machine learning models based on grid search method, *Multimedia Tools and Applications* 83 (12) (2024) 35307–35334.
- [20] Ž. Vujović, et al., Classification model evaluation metrics, *International Journal of Advanced Computer Science and Applications* 12 (6) (2021) 599–606.

September 4, 2025