# Prediction of Loan Process Using an Assortment of Supervised Learning Algorithms

Dr O Yamini[1], S R Ajitha[2], Dr G V Ramesh Babu[3]
[1,2] *Guest Faculty, Dept. of Computer Science S V University, Tirupati*
[3] *Associate Professor, Dept. of Computer Science S V University, Tirupati*

***Abstract*—One of the major challenges faced by banks and other financial institutions is the increasing rate of loan default, which leads to the deterioration of their loan assets into non-performing assets. This adversely affects their capital adequacy and profitability and may force them to merge with other entities or close down. Therefore, it is imperative to assess the risk of loan default of potential borrowers before granting them loans. The goal of this research is to create a predictive model that can estimate the probability of loan default based on a range of variables, including age, education level, number of dependents, income level and source of income. These are a few fundamental standards that can be used to determine who credit is worthy and prevent unfavorable loans. This paper aims to automate the loan approval process using a predictive model with support from various supervised learning algorithms. This can improve service quality and efficiency while reducing the need for human resources, but it will require more computing power.**

***Index Terms*—Decision making systems, Node Split, Multivariate attributes, Pruning, Feature Selection**

## I. INTRODUCTION

Decision-making systems are software applications that help banks process auto loan applications more accurately and rapidly. They assess each loan's risk and profitability using a range of factors, including loan amount, car type, income, and credit score. These variables determine whether the loan is approved, denied, or sent to a human specialist for additional examination by the system. In order to lower errors and fraud, decision making systems can also automate the documentation and verification processes. Banks can gain from decision-making systems by using them to improve regulatory compliance, lower operating costs, and increase customer happiness.

## II. SUPERVISED AND UNSUPERVISED LEARNING

*Supervised Learning*

Training a data sample from a data source with the proper categorization already given forms the basis of supervised learning. Supervised learning includes algorithms like KNN, SVM, Random Forest, and Decision Tree Classifications. These methods are applied in MultiLayer Perceptron (MLP) or feedforward models. These MLP are distinguished by three features:

1. One or more layers of hidden neurons that allow the network to learn and solve any difficult problem; these neurons are not a part of the input or output layers of the network.
2. Differentiable nonlinearity is represented in the neural activity.
3. The network's interconnection model shows a high level of connectedness.

*Unsupervised Learning*

In order to find hidden patterns in unlabeled input data, self-organizing neural networks employ unsupervised learning algorithms. The ability to acquire and arrange knowledge without generating an error signal to assess a possible solution is referred to as "unsupervised." Unsupervised learning can have some benefits when the learning algorithm has no direction since it allows the algorithm to search the past for patterns that were overlooked [1]. Self-Organizing Maps' (SOM) primary traits are as follows:

1. It performs an adaptive conversion of an incoming signal pattern of any dimension into a one- or two-dimensional map.
2. A single computational layer made up of neurons grouped in rows and columns makes up the network's representation of a feed forward structure.
3. Every step of the representation process, each

One of the main characteristics of the Indian car business is vehicle finance, which allows consumers to purchase a car without having to pay the entire cost at once. Consumer preferences for auto financing are influenced by a number of variables, including age, income, work status, and socioeconomic standing. Customer selections are influenced by a wide range of socio-economic characteristics, including education, cultural background, and values. Furthermore, the affordability and availability of financing choices for consumers are influenced by macroeconomic factors such as inflation, interest rates, and economic growth. Lenders, dealerships, and legislators must comprehend how these factors interact in order to create strategies that satisfy client needs and encourage appropriate financing practices [3].

Financial services are changing as a result of AI/ML capabilities. AI/ML systems are changing how customers interact with financial service providers, invest, borrow, and verify their identity. Examples of these interactions include chatbots, robo-advisors, automated mortgage underwriting, and picture recognition. They are also revolutionizing the way financial institutions operate, enabling substantial cost savings through process automation, leveraging predictive analytics to improve product offerings, and facilitating more efficient risk and fraud management procedures in addition to regulatory compliance. Lastly, new methods for strengthening prudential monitoring and enhancing systemic risk surveillance are made available to central banks and prudential oversight bodies by AI/ML systems [4].

## III. DECISION MAKING SYSTEMS IN LOAN PROCESSING

Regarding automated loan decision making, the main issue is the quality of the data that was utilized to run and train the algorithm. Preciseness, thoroughness, and audience representation are required of the data. An biased, out-of-date, or inadequate set of data could lead to unfair or erroneous conclusions from the algorithm. The algorithm might, for instance, refuse loans to low-income or minority borrowers based on arbitrary criteria if the data has insufficient information about them. Regulatory compliance and client satisfaction are two more issues with automated loan decision making. Full automation can result in important clients being turned down for loans.

### [i] Analysis of Decision Tree Technique

Selecting the best choice from a range of options is the process of making a decision. Decision makers can choose the choice that best meets their goals by using a variety of strategies to weigh the advantages and disadvantages of each.

For each outcome, the decision tree may show the likelihood, expected value, or expected utility, depending on the quality criteria applied. Decision trees can also consider risk, uncertainty, and the decision maker's preferences, which would make them more adaptable and practical. Decision trees can be constructed using a variety of methods, such as hybrid, top-down, and bottom-up approaches. Decision trees have several advantages, such as the capacity to manage both quantitative and qualitative data, simplicity in comprehension and communication, and in conjunction with additional techniques such as Monte Carlo simulation and sensitivity analysis.

To create decision trees from data, a number of methods are available, such as ID3, C4.5, CART, CHAID, and others. The optimal attribute to partition the data at each node, how these algorithms handle noise and missing values, and how they prune the tree to prevent overfitting are the areas where these approaches diverge. The chi-square test, gain ratio, Gini index, and information gain are a few of the most often used splitting criteria.

#### *a) Node Splitting*

Node splitting measures are the most important of the methods that may be used when building decision trees, and they are one part of a multi-part strategy that can be used to create compact decision trees with better generalization capabilities.

Because only a small number of attributes are significant discriminating attributes—a

discriminating attribute is one whose value is likely to allow us to identify one tuple from another—the distribution of attributes with regard to information gain is rather sparse. An effective decision tree classifier for categorical attributes of sparse distribution is proposed by Lo et al. (2003).

One of the main challenges in decision tree construction is figuring out the optimal node splitting criterion that can reduce the size and complexity of the tree while maintaining its accuracy and interpretability. Node splitting measurements are based on a variety of metrics that evaluate the level of purity or homogeneity of the resulting split data subsets. The gini index, entropy, and misclassification error are a few of the often used metrics. It is possible to utilize these metrics for both numerical and category properties. However, depending on the distribution of the data and the number of possible values for each feature, their computing costs and performance could change. The extremely sparse attribute distribution can only be partially explained by a small number of qualities with high information gain values [5].

*b) Multivariate attribute and model Selection*

An approach to compare decision trees, both multivariate and univariate, is to examine how the data is divided into attributes. An attribute can only be split using one univariate decision tree at a time, whereas a multivariate decision tree can split using multiple attributes, typically in a linear combination. Oblique splits, which are linear splits that can cut across numerous qualities, have been studied by certain researchers.

Selecting the right model complexity for every node in a decision tree learning system is another challenge. It may not be accurate, despite popular perception, to assume that all nodes have the same complexity. For some nodes, especially in the vicinity of the tree's root, more complex models—including nonlinear models—might be required in order to depict the decision boundary. But as we move deeper down the tree, the data gets sparser and the decision gets easier, thus linear models might be more appropriate and less likely to overfit [5].

Using an ensemble of models, each trained with a different set of parameters and features, is an innovative way to handle model selection at each node. In this manner, the node can improve accuracy and robustness by combining the predictions from many models (Altınçay 2007). Additionally, this approach is capable of handling both linear and nonlinear models at the same node [6].

[ii] Controlling decision tree complexity

As [5] demonstrates, decision tree classifications are unstable because even little modifications to the training set can result in notable changes to the tree's structure. Reducing the size of decision trees is an important task. The minimization problem for decision trees is not only NP-hard but also challenging to approximate within any constant factor, with the exception of the situation when P = NP, as shown by [7]. The complexity of the tree is determined by the pruning process and halting criteria used. Several common metrics are used to quantify a tree's complexity, including the total number of nodes, leaves, depth, and features.

[iii] Pruning

The goal of pruning decision trees is to minimize a tree's size and complexity without compromising its accuracy. Trees can be pruned in a variety of ways, including top-down and bottom-up methods. If a node raises a particular quality metric, it gets taken out of the tree. But not every trimming technique works the same way. According to certain research (Esposito et al. 1997), some pruning techniques—such as reduced error pruning and cost-complexity pruning—tend to result in smaller but less precise trees. We refer to these techniques as over-pruning techniques. However, some techniques (including minimum error pruning, pessimistic error pruning, and error-based pruning) have a tendency to keep more nodes than necessary, which makes the trees larger but not any more accurate. We refer to these techniques as under-pruning techniques. No single pruning technique is always better than others in every circumstance. DI pruning is one technique for pruning that aims to strike a balance between the trade-offs between accuracy and size; it was first presented by authors in [8]. This approach retains sub-trees that can produce valuable decision rules despite their inability to increase classification accuracy, taking into account the sub-trees' level of complexity.

*a. Feature selection*

An effective way to prevent overfitting at each node is to identify the best subset of characteristics to keep. The key idea here is that some dimensions can be ignored since they may not change in the data subspace that gets to a specific node. Eliminating these attributes could improve the ability to generalize and reduce the complexity of the node. Feature selection algorithms typically consist of two parts: an evaluation algorithm that determines the degree of "goodness" of each potential feature subset and feeds that information back to the selection algorithm and a selection algorithm that creates potential feature subsets and searches for an ideal subset. However, if there isn't a strong stopping criterion, the feature selection process may drag on for too long or forever throughout the subset space. The following are examples of stopping standards:

(i) Determining if adding (or removing) any characteristic results in a better subset

(ii) Determining whether an optimal subset as determined by an evaluation function is obtained [5].

*b. Handling very large datasets with decision trees*

Data mining is the process of extracting useful information from large, complex data sets. However, many data mining methods face scalability issues when dealing with data sets that are too big to fit in the memory of a single machine. To overcome this issue, scientists have created decision tree algorithms that are capable of managing massive volumes of data in an effective and efficient manner.

Li (2005) presented the SURPASS algorithm as one of these algorithms. Scaling Up Recursive Partitioning with Sufficient Statistics is referred to as SURPASS. At each node of the tree, it divides the data into smaller subsets using linear discriminants. Additionally, it makes use of adequate statistics—summary measurements that can be calculated gradually from the data without requiring the loading of the entire data set into memory. In this manner, data sets greater than the memory that is available can be used by SURPASS to create decision trees [9].

The SPIES method was put forth by Jin and Agrawal (2003) [10]. "Sampling and Partitioning for Interval Estimation of Splits" is what SPIES stands for. Using a sample of the data set, the number of split points that can occur is limited by splitting the values into intervals and computing the class histograms for each interval. As a result, the algorithm's space complexity and communication cost are decreased. Additionally, they used the FREERIDE framework, which enables distributed and cooperative data mining, to parallelize this approach. Using this method, they saw near-linear speedups.

*c. Handing cost-sensitive problems with decision trees*

Data mining is the process of removing important information from large, complex data collections. However, many data mining algorithms have scaling issues when dealing with data sets too big to fit in the memory of a single computer. To address this issue, researchers have proposed decision tree algorithms, which are effective at managing massive volumes of data.

The prices may not be known with precision or may vary over time, which presents a barrier for cost-sensitive learning. Making a set of decision trees that are optimal under various cost scenarios and allowing the decision maker to select the best one based on their particular circumstances is therefore desired. Multi-objective genetic programming is the basis of Zhao's (2007) suggested approach. To further reduce the number of possible options, the approach can also take into account the decision maker's preferences for the trade-offs between various objectives, such as false positive vs. false negative, sensitivity vs. specificity, and recall vs. precision [11].

*d. Handling uncertain data with decision trees*

A different approach to building decision trees based on Variable Precision Rough Set Model is presented in a paper by Wei et al. [13]. The authors of that paper investigate how to build decision trees from data with uncertain class values and suggest a new metric for choosing attributes: non-specificity based gain ratio. This measure is more appropriate than the traditional gain ratio that employs Shannon entropy. By permitting some degree of misclassification when assigning classes to objects, this method extends the rough set based method and can handle unclear information during the decision tree induction process. A straightforward and efficient method for handling missing data in decision trees for classification problems has also been provided by some of the writers.

Uncertain data presents a significant obstacle for classification algorithms. The probability distribution of each training instance in this scenario represents the expert's confidence in each conceivable class. This is not the same as ambiguous classification, which provides only a portion of possible classifications. As a result, an expert might rate the potential classes to convey his viewpoint. A class's possibility degree indicates the expert's level of confidence that the chosen class is the right one [5].

*e. Hybrid methods*

Various supervised learning techniques are included into hybrid systems to improve prediction accuracy. Among the hybrid approaches are the voted decision stumps, voted decision trees, and the alternating decision tree (ADTree), which is a generalization of decision trees [14]. Every instance that an ADTree defines has a set of pathways defined in the tree. This is the definition of a classification rule. Decision nodes are followed by the routes, just like in traditional decision trees. At the prediction node, however, they split off into many paths, one path for each of the children of the prediction node. All possible paths that an instance can take are gathered together under the phrase "multi-path". The instance's classification is determined by the sign of the total of all the prediction nodes.

Authors' binary hybrid decision tree, presented in [15], is another illustration of a hybrid approach. In accordance with the binary information gain ratio criterion, this technique produces a binary decision tree. The instances that fall into a leaf node that are designated as dummy nodes are fed into a Feed-Forward Neural Network (FANNC) when the node exhibits too much variability and cannot be further divided by any characteristic.

## IV CONCLUSION AND FUTURE WORK

The description above indicates that decision-making algorithms, like as ID3, C4.5, CART, and others, are used in a number of fields, including loan processing in financial institutions, to forecast and affect decision-making approaches. All algorithms, nevertheless, have advantages and disadvantages. While the ID3 method's authors made the algorithm simpler by substituting weight factors for entropy, backtracking and instability with dynamic databases remain a problem. Some extensions with numerical characteristics (e.g., C4.5) are recommended to overcome the issues mentioned above. Their computation is challenging, though, involving a lot of logarithmic operations that use a lot of library functions. Consequently, we observed that most methods experienced problems with computational complexity and performance as datasets increased, leading to under or overfitting. For these reasons, we formulated our goals and objectives for this work and created a novel FPBDT (Feature Probability Based Decision Tree Algorithm) for data classification.

## REFERENCES

[1] T. Kohonen, O. Simula, “Engineering Applications of the SelfOrganizing Map”, Proceeding of the IEEE, Vol. 84, No. 10, 1996, pp.1354 – 1384

[2] R. Sathya, Annamma Abraham, Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification, (IJARAI) International Journal of Advanced Research in Artificial Intelligence, Vol. 2, No. 2, 2013, page 34-38. https://thesai.org.

[3] Dr R Muthukrishnan et.al, Socio-Economic Factors’ Impact on Vehicle Financing & Consumer Behaviour, Journal of Survey in Fisheries Sciences, 10(1S) 5055-5065-2023

[4] Boukherouaa, E. B., AlAjmi, K., Deodoro, J., Farias, A., &Ravikumar, R. (2021). Powering the Digital Economy: Opportunities and Risks of Artificial Intelligence in Finance, Departmental Papers, 2021(024).

[5] Kotsiantis, S.B. Decision trees: a recent overview. Artif Intell Rev 39, 261–283 (2013). https://doi.org/10.1007/s10462-011-9272-4

[6] Altınçay H (2007) Decision trees using model ensemble-based nodes. Pattern Recognition40:3540–3551

[7] Detlef Sieling, Minimization of decision trees is hard to approximate, Journal of Computer and System Sciences, Volume 74, Issue 3, 2008, Pages 394-403, ISSN 0022-0000.

[8] Fournier D, Crémilleux B (2002) A quality index for decision tree pruning. Knowledge Based Systems 15(1-2):37–43

[9] Fayyad, U. M.; Piatetsky-Shapiro, G.; and Smyth, P.1996. From Data Mining to Knowledge Discovery: An Overview in Advances in Knowledge Discovery and Data Mining, eds. U. Fayyad et.al, 1–30. Menlo Park, Calif.: AAAI Press.

[10] Pieter Adriaans and Dolf Zantinge. Data Mining. Addison Wesley Longman, Harlow, England, 1996.

[11] Zhao H (2007) A multi-objective genetic programming approach to developing Pareto optimal decision trees. Decision Support Systems 43:809–826

[12] Jenhani I, Amor Nahla B, Elouedi Z (2008) Decision trees as possibilistic classifiers, International Journal of Approximate Reasoning, Volume 48, Issue 3, 2008, Pages 784-807, ISSN 0888-613X

[13] Wei J-M, Wang S-Q, Wang M-Y, You J-P, Liu D-Y (2007) Rough set-based approach for inducing decision trees.Knowl Based Syst 20:695–702

[14] Freund Y, Mason L (1999) The alternating decision tree learning algorithm. In: Proceedings the sixteenth international conference on machine learning, Bled, Slovenia, pp 124–133

[15] Zhou Z-H, Chen Z-Q (2002) Hybrid decision tree. Knowl Based Syst 15(8):515–528