An Empirical Study of Fuzzy Decision Tree for Gradient Boosting Ensemble

UTS Australian AI Institute

Zhaoqing Liu, Anjin Liu, Guangquan Zhang, and Jie Lu

Australian Artificial Intelligence Institute, University of Technology Sydney, Sydney, Australia

INTRODUCTION

In the real world, fuzzy concepts and fuzzy objectives are ubiquitous in the process of human cognition and decision-making, and the uncertainties they cause, e.g., incomplete and imprecise data, usually lead to the performance degradation of machine learning models. In order to deal with uncertainties, fuzzy decision trees have been widely used in many applications. However, Few implementations of fuzzy decision trees are based on the CART, which is a decision tree algorithm mainly adopted by most ensemble tree algorithms. Gradient boosting is a primary boosting algorithm in ensemble learning, where the gradient boosting decision tree algorithm is one of the most popular machine learning methods in the industry. However, no study compares fuzzy gradient boosting decision trees with non-fuzzy gradient boosting decision trees.

The contribution of this study includes three aspects:

- A novel fuzzy decision tree-based gradient boosting algorithm is proposed.
- A Python toolkit for FDT and FGBDT is developed.
- Extensive experiments indicate that FDT and FGBDT can achieve better accuracy in many classification tasks than non-fuzzy FDT and non-fuzzy FGBDT, respectively.

RELATED CONCEPTS

Decision trees: One of the most representative algorithms in machine learning. A decision tree uses a tree-like model of symbols, rules, and logic to represent knowledge and make logical inferences.

Fuzzy set theory: A generalisation of classical set theory (a.k.a. crisp sets), and it is especially suitable for a wide range of domains with incomplete or imprecise information.

Fuzzy decision trees: A fuzzy decision tree is an extension of the classical decision tree. It is more robust in tolerating uncertainty by introducing fuzzy set theory.

Gradient Boosting: An ensemble learning technique and one of the most popular algorithms. Its idea is derived from the gradient descent method. The gradient descent method can combine multiple weak learners into a strong one.

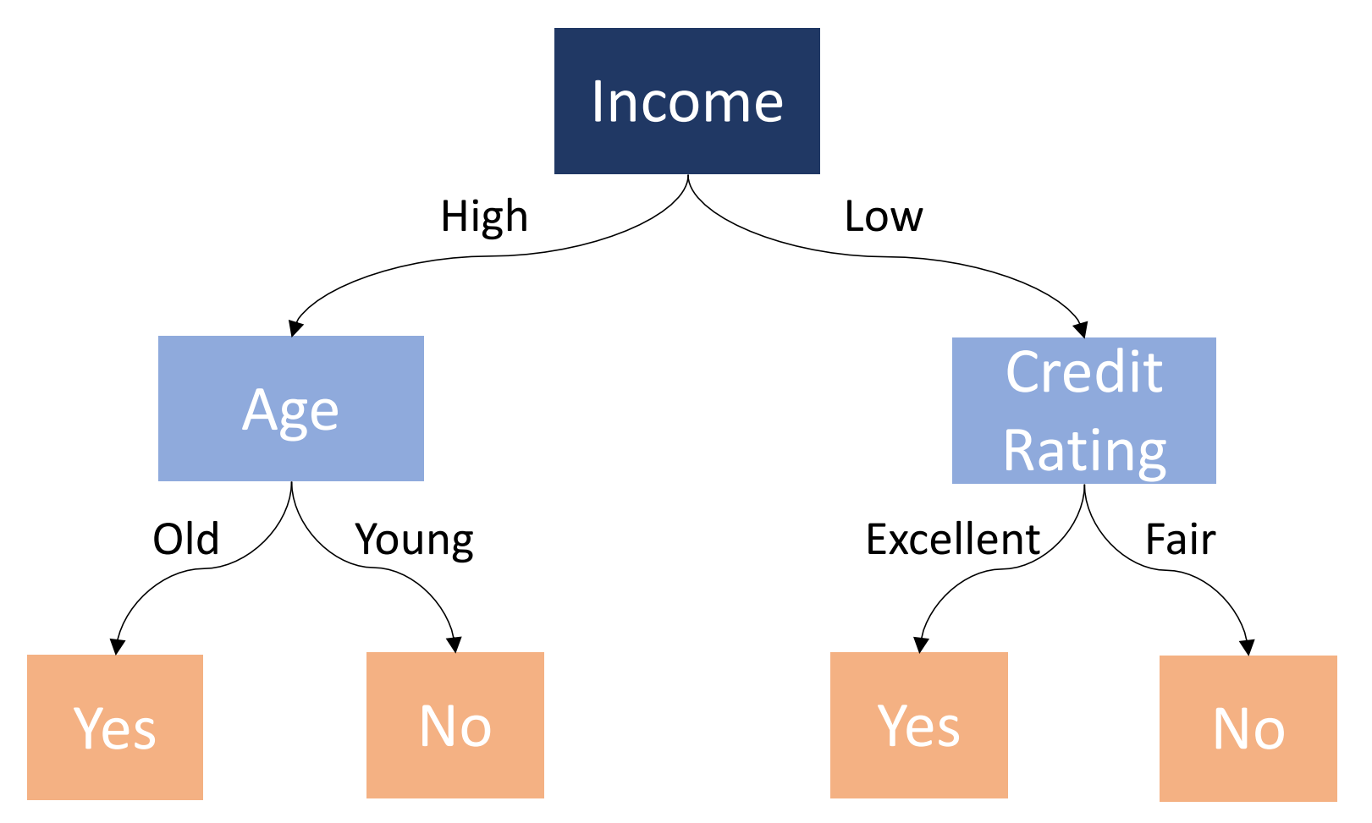

Fig 1: A simple example of a fuzzy decision tree.

METHODS

1. Framework of Fuzzy Decision Trees

Similarities between FDT and Non-fuzzy FDT: The tree construction and prediction stages of FDT and its non-fuzzy FDT.

Differences between FDT and Non-fuzzy FDT :

- The former executes additional feature fuzzification (FF) using the Fuzzy C-Means clustering algorithm (FCM) in data preprocessing.
- The former uses the fuzzy sets for metric fuzzification to calculate all splitting criteria in tree construction.

Feature Fuzzification (FF): The fuzzy transformation of features in the data preprocessing stage before constructing an FDT tree.

Metric fuzzification (MF): The fuzzy calculation of splitting criteria for feature selection in the tree construction stage.

2. Metric Fuzzification in Tree Construction

We take the fuzzy information gain and fuzzy information gain ratio based on the fuzzy entropy or fuzzy Gini impurity for heuristic functions in tree constructions:

- Fuzzy entropy: $I(S) = -\sum_{k=1}^{n}(p_k \cdot \log_2 p_k)$, where $p_k = \frac{|S^{C_k}|}{|S|}$
- Fuzzy information gain: $G(A_i, S) = I(S) - E(A_i, S)$ and $E(A_i, S) = \sum_{j=1}^{m}(p_{ij} \cdot I(S_{F_{ij}}))$, where $p_{ij} = \frac{|S_{F_{ij}}|}{\sum_{j=1}^{m}|S_{F_{ij}}|}$
- Fuzzy Gini impurity: $I_G(S) = \sum_{k=1}^{n} p_k(1 - p_k)$
- Fuzzy information gain ratio:$GR(A_i, S) = \frac{G(A_i,S)}{IV(A_i,S)}$
- Intrinsic value:$IV(A_i, S) = -\sum_{t=1}^{n}(p_t \cdot \log_2 p_t)$, where $p_t = \frac{|A_i^{V_t}|}{|A_i|}$

3. Implementation of Gradient Boosting Fuzzy Decision Trees

Gradient Boosting Fuzzy Decision Trees (FGBDT): Using the idea of gradient boosting, FGBDT combines multiple weak fuzzy decision trees into a single strong learner in an iterative fashion, then gradually approximate the optimal learner in a greedy fashion.

Differences between FGBDT and non-fuzzy FGBDT: FGBDT integrates a set of regression FDTs instead of non-fuzzy ones.

Similarities between FGBDT and non-fuzzy FGBDT:

- Pseudo residuals: $r_m = -\left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}\right]_{f(x)=f_{m-1}(x)}$
- Multiplier: $\gamma_m = \arg\min_{\gamma} \sum_{i=1}^{N} L(y_i, f_{m-1}(x_i) + \gamma h_m(x_i))$
- Update the model: $f_m(x) = f_{m-1}(x) + \gamma_m h_m(x)$

4. Time Complexity

As shown in Table 1, fuzzification increases the time complexity of algorithms during training, and the time complexity is proportional to the number of fuzzy sets after feature fuzzification. However, that does not happen during prediction.

RESULTS

Datasets: We consider six datasets from the UCI Machine Learning Repository: Vehicle Silhouettes (VS), German Credit (GC), Pima Indians Diabetes (PID), Iris, Wine, and Forest Cover Type (FCT).

We conduct four comparison experiments to study the performance improvement of: 1. FDT models by using only feature fuzzification (FF); 2. FDT models by combining FF and metric fuzzification (MF); 3. FGBDT models by combining FF and MF; 4. FDT models with six published baselines.

Table 2: Results with FF (n conv = 3) and without FF.

Task	DT with FF		DT without FF	
	Acc	Std	Acc	Std
VS	**0.6963**	0.0341	0.6643	0.0305
GC	0.7080	0.0449	**0.7100**	0.0316
PID	**0.7226**	0.0478	0.7084	0.0509
Iris	**0.9333**	0.0629	**0.9333**	0.0629
Wine	0.8935	0.0676	**0.8990**	0.0742
Avg	**0.7907**	0.0515	0.7830	0.0500

Table 3: Results with FF (n conv = 4) and without FF.

Task	DT with FF		DT without FF	
	Acc	Std	Acc	Std
VS	**0.6963**	0.0341	0.6643	0.0305
GC	0.7080	0.0449	**0.7100**	0.0316
PID	**0.7226**	0.0478	0.7084	0.0509
Iris	**0.9333**	0.0629	**0.9333**	0.0629
Wine	0.8935	0.0676	**0.8990**	0.0742
Avg	**0.7907**	0.0515	0.7830	0.0500

Table 4: Results with FF (n conv = 5) and without FF.

Task	DT with FF		DT without FF	
	Acc	Std	Acc	Std
VS	**0.6963**	0.0341	0.6643	0.0305
GC	0.7080	0.0449	**0.7100**	0.0316
PID	**0.7226**	0.0478	0.7084	0.0509
Iris	**0.9333**	0.0629	**0.9333**	0.0629
Wine	0.8935	0.0676	**0.8990**	0.0742
Avg	**0.7907**	0.0515	0.7830	0.0500

Table 5: Results with FDT and non-fuzzy FDT.

Task	DT with PF		DT with non-PF	
	Acc	Std	Acc	Std
VS	**0.6915**	0.0619	0.6643	0.0305
GC	**0.7200**	0.0287	0.7100	0.0316
PID	**0.7422**	0.0389	0.7084	0.0509
Iris	**0.9333**	0.0629	**0.9333**	0.0629
Wine	**0.9108**	0.0650	0.8990	0.0742
Avg	**0.7996**	0.0515	0.7830	0.0500

Table 6: Results with FGBDT and non-fuzzy FGBDT.

Task	DT with PF		DT with non-PF	
	Acc	Std	Acc	Std
VS	**0.6832**	0.0457	0.6572	0.0223
GC	**0.6840**	0.0504	0.6790	0.0451
PID	**0.7082**	0.0465	0.7031	0.0642
Iris	**0.9400**	0.0663	0.9333	0.0629
Wine	0.8987	0.0692	**0.8990**	0.0742
Avg	**0.7828**	0.0556	0.7743	0.0537

Table 7: Comparison between FDT and baselines on dataset FCT.

Methods	Acc	Std
XGBoost$_b$	0.6566	0.0395
CatBoost$_b$	0.6302	0.0455
LightGBM$_b$	0.4880	0.0518
HT	0.5390	0.0524
HAT	0.5418	0.0510
SAMKNN	0.5533	0.0600
FDT	**0.6639**	0.0434

CONCLUSIONS

In conclusion, FGBDT models based on FDT can improve performance and enhance gradient boosting's optimisation in many classification tasks.

Table 1: Time complexity for FDT and FGBDT.

Algorithm	Time complexity	
	Training	Prediction
FDT	$\mathcal{O}(N \log_2 NMC) \sim \mathcal{O}(N^2MC)$	$\mathcal{O}(\log_2 N) \sim \mathcal{O}(N)$
FGBDT	$\mathcal{O}(TN \log_2 N) \sim \mathcal{O}(TN^2)$	$\mathcal{O}(T \log_2 N) \sim \mathcal{O}(TN)$

https://github.com/ZhaoqingLiu/FuzzyTrees

UTS CRICOS PROVIDER CODE: 00099F