International Journal of Operations Research and Artificial Intelligence

www.ijorai.reapress.com

Int. J. Oper. Res. Artif. Intell . Vol. 1, No. 4 (2025) 175–181.

Paper Type: Original Article

Integration of DEA-Based Inefficiency into Decision Tree Splitting Criteria

Mohammad Hassan Behzadi*

Department of Mathematics, Science and Research Branch, Islamic Azad University, Tehran, Iran; behzadi@srbiau.ac.ir.

Citation:

Received: 17 January 2024 *Revised: 13 May 2024* *Accepted: 26 August 2024*	Behzadi, M. H. (2025). Integration of DEA-based inefficiency into decision tree splitting criteria. *International Journal of Operations Research and Artificial Intelligence, 1*(4), 175-181.

Abstract

This paper introduces a modified decision tree methodology that incorporates efficiency evaluation through Data Envelopment Analysis (DEA). Standard decision trees rely on class distribution metrics such as the Gini index or Entropy, overlooking the performance characteristics of individual observations. In contrast, the proposed approach calculates inefficiency scores for each Decision-Making Unit (DMU) based on input excesses and output shortfalls. It uses these scores to weight samples during tree construction. This integration allows the model to achieve accurate classification while simultaneously highlighting units with poor performance. Results indicate that the DEA-informed decision tree effectively captures inefficiency patterns and provides actionable insights for performance improvement, while maintaining predictive accuracy.

Keywords: Machine learning, Decision tree, Data envelopment analysis.

1 | Introduction

Decision trees are popular in machine learning for their interpretability, simplicity, and capacity to handle diverse datasets. However, conventional trees focus solely on splitting nodes according to class purity and ignore the operational efficiency of the entities being analyzed. In practice, especially in organizational, healthcare, and financial contexts, it is critical to identify underperforming units to support strategic decisions.

Data Envelopment Analysis (DEA) is a widely used approach to assess relative efficiency by comparing multiple inputs and outputs across units. Incorporating DEA into a decision tree allows the algorithm to consider both class separation and performance efficiency. By assigning stronger influence to units with greater inefficiency during node splitting, the resulting tree not only classifies observations effectively but also highlights branches associated with operational weaknesses [1–4]. This dual focus enhances decision-making

Corresponding Author: behzadi@srbiau.ac.ir

https://doi.org/10.48314/ijorai.v1i4.76

by combining predictive capabilities with performance assessment, addressing a key limitation of traditional tree-based methods.

This paper presents a comprehensive framework for evaluating bank branch efficiency, covering preliminaries, methodology, a case study on DEA-informed decision tree analysis, and a conclusion. The findings indicate that integrating DEA-based inefficiency measures with decision tree learning provides an effective approach for assessing and interpreting branch performance, offering actionable insights for managers to optimize resource allocation and improve operational efficiency.

DEA is a widely used nonparametric methodology for assessing the relative efficiency of Decision-Making Units (DMUs) across multiple inputs and outputs. The foundational CCR model by Charnes et al. [1] provided a systematic framework to evaluate efficiency without assuming a specific functional form. This model was later extended by Banker et al. [5] to account for technical and scale inefficiencies, offering more nuanced insights into operational performance.

Additionally, approaches such as those proposed by Daraio and Simar [6] integrate environmental variables probabilistically, enabling the measurement of efficiency under varying external conditions. Further refinements, including constrained polynomial splines for data envelope fitting [7], enhance the flexibility and accuracy of DEA models in practical applications.

Tree-based machine learning algorithms provide interpretable, effective predictive modeling tools that often complement DEA analyses. Breiman [8] introduced bagging to reduce variance by aggregating multiple decision trees, while iterated bagging [9] further mitigates bias in regression models. Gradient boosting methods, exemplified by XGBoost [10], provide scalable, precise tree-based predictions for large datasets. Combining DEA-derived inefficiency scores with these machine learning techniques enables hybrid models in which efficiency outcomes guide predictive splits, thereby improving interpretability and performance assessment in operational and financial contexts.

2 | Preliminaries

DEA is a linear programming-based method for evaluating the relative efficiency of DMUs that operate with multiple inputs and outputs. By constructing a best-practice frontier from the observed data, DEA determines how far each unit deviates from the most efficient performance. Its nonparametric framework eliminates the need for prior assumptions about functional forms, enabling flexible analysis across different orientations and scales. Both radial and non-radial DEA models are commonly applied: radial models proportionally scale all inputs or outputs, while non-radial models allow uneven adjustments, capturing more nuanced inefficiency patterns.

In recent years, integrating DEA with machine learning techniques has gained attention, particularly with decision trees, random forests, and ensemble learning methods. These approaches benefit from DEA by weighting observations based on their inefficiency scores, thereby enhancing predictive models with performance-aware splitting criteria.

For example, in decision tree and random forest algorithms, incorporating efficiency information can prioritize underperforming units, improving both classification accuracy and interpretability. Ensemble learning further leverages multiple models to provide robust predictions while maintaining sensitivity to efficiency variations, bridging operational performance assessment and predictive analytics in a unified framework.

Suppose that n DMUs produce s outputs by consuming m inputs. Also, suppose that in the case where the units are black boxes, $X_j = (x_{1j}, x_{2j}, \dots, x_{mj})$ represents the vector of inputs and $Y_j = (y_{1j}, y_{2j}, \dots, y_{sj})$ represents the vector of outputs of the DMU_j . The model for calculating the efficiency of the DMU_o, $o \in \{1, \dots, n\}$, is as follows.

$$
\begin{aligned}
&\mathrm{Z_o} = \mathrm{Max}\,\{\sum_{r=1}^{s}\rho_r + \sum_{i=1}^{m}\beta_i\} \\
&\text{s.t.} \\
&\sum_{j=1}^{n}\alpha_j x_{ij} + \beta_i = x_{io}, i = 1,\dots,m, \\
&\sum_{r=1}^{s}\alpha_j y_{rj} - \rho_r = y_{ro}, j = 1,\dots,n, \\
&\textstyle\sum_{j=1}^{n}\alpha_j = 1, \\
&\alpha_j, \rho_r, \beta_i \geq 0, r = 1,\dots,s, i = 1,\dots,m, j = 1,..,n.
\end{aligned}
\tag{1}
$$

The presented model is an additive DEA model designed to evaluate the relative efficiency of DMUs by simultaneously considering input excesses and output shortfalls. In this formulation, each DMU is compared to a linear combination of peer units, represented by weights α_j. The slack variables β_i and ρ_r capture the additional input use and the shortfall in outputs for the DMU under evaluation, respectively. The objective function maximizes the average of these slack-adjusted inefficiencies, providing a collective measure of inefficiency that incorporates both input and output deviations from the efficient frontier.

The constraints ensure that the weighted combination of peer DMUs, adjusted by the slack variables, equals the target DMU's observed inputs and outputs. A convexity condition is imposed by requiring that the sum of weights $\sum_{j=1}^{n}\alpha_j = 1$, reflecting Variable Returns to Scale (VRS). All weights and slack variables are non-negative, ensuring meaningful interpretations. This additive model differs from traditional ratio-based DEA models by explicitly quantifying inefficiencies in both dimensions (inputs and outputs), allowing a more granular assessment of performance. A DMU achieving the maximum objective value indicates full efficiency, while lower values identify units with notable input excesses or output shortfalls.

3 | Methodology

In decision tree algorithms, the Gini index is commonly used as a measure of node impurity to determine the best feature for splitting the data. Traditionally, the Gini index considers only the distribution of class labels within a node, without accounting for external performance metrics of the data points. By incorporating an inefficiency score derived from DEA or similar methods, each observation can be weighted by its relative inefficiency, allowing the splitting criterion to reflect both class purity and the performance of the units. This adjustment ensures that splits favor nodes that group not only similar classes but also more efficient or less inefficient observations.

Using inefficiency-adjusted Gini measures helps prioritize observations with a larger impact on overall system performance. Observations with higher inefficiency contribute more to the impurity measure, making the algorithm more sensitive to underperforming units. As a result, the modified decision tree can produce splits that better capture data patterns while simultaneously reflecting each unit's operational performance. This approach effectively integrates classification and efficiency analysis, enabling the decision tree to serve as both a predictive model and a performance assessment tool.

$$
\begin{aligned}
&\mathrm{W_o} = \mathrm{Max}\left\{\frac{\beta_i}{x_{io}}, \frac{\rho_r}{y_{ro}}, i = 1,\dots,m, r = 1,\dots,s\right\}. \\
&\text{s.t.}
\end{aligned}
\tag{2}
$$

$$\sum_{j=1}^{n} \alpha_j x_{ij} + \beta_i = x_{io}, i = 1, \ldots, m,$$

$$\sum_{r=1}^{s} \alpha_j y_{rj} - \rho_r = y_{ro}, j = 1, \ldots, n,$$

$$\sum_{j=1}^{n} \alpha_j = 1,$$

$$\alpha_j, \rho_r, \beta_i \geq 0, r = 1, \ldots, s, i = 1, \ldots, m, j = 1, \ldots, n.$$

The proposed model is a relative inefficiency-based DEA approach, designed to evaluate DMUs by identifying the maximum proportional slack in both inputs and outputs. Unlike traditional additive DEA models, which consider the average slack across all inputs and outputs, this model focuses on the largest relative inefficiency for each DMU, measuring how much each input exceeds its efficient benchmark and how much each output falls short of its target. By normalizing the slack with respect to observed input or output values, the model provides a scale-independent inefficiency measure that highlights the most critical areas where the DMU underperforms.

The model's constraints ensure that each DMU is compared to a convex combination of peer units, with the respective slack variables adjusting for differences. The sum of weights equals one, reflecting variable VRS, while all weights and slack variables are non-negative to ensure meaningful interpretations. Maximizing the objective identifies the unit under evaluation's largest proportional inefficiency, allowing decision-makers to pinpoint the most significant input excesses or output shortfalls. This approach is particularly useful for performance improvement and benchmarking, as it not only quantifies overall inefficiency but also highlights the specific inputs or outputs that contribute most to underperformance.

The inefficiency of each DMU is computed using the proposed DEA m*odel (2)*, which evaluates both input excesses and output shortfalls. For each DMU, the model identifies the maximum proportional slack across inputs and outputs, yielding a normalized inefficiency measure. This measure provides a scale-independent assessment of the unit's performance and highlights the most critical areas of underperformance. Each observation in the dataset is then associated with an inefficiency score, which influences its contribution to the splitting criterion in the decision tree.

In the modified algorithm, the standard Gini index is replaced with a weighted impurity measure based on DEA inefficiency. Observations with higher inefficiency contribute more to node impurity, ensuring that splits account for both class distribution and performance quality. During splitting, all candidate features are evaluated, and the split that maximizes the reduction in weighted impurity is selected. This approach prioritizes nodes that group similar classes while also capturing units with high inefficiency, enhancing the interpretability of the tree in performance analysis contexts.

The recursive splitting continues until stopping criteria are met, such as a minimum number of observations per node or a threshold for weighted impurity reduction. At each leaf, the average inefficiency of the contained observations is reported, allowing decision-makers to identify branches with the highest operational shortcomings. Consequently, the tree provides a dual-purpose output: it classifies observations accurately while simultaneously highlighting critical inefficiency patterns.

The proposed DEA-informed decision tree effectively combines predictive classification with performance evaluation, offering richer insights than conventional trees. By weighting observations according to inefficiency, the algorithm becomes more sensitive to underperforming units, ensuring that key areas for improvement are visible in the resulting model. This integrated approach is particularly useful in operational or managerial contexts where both the classification outcome and unit efficiency are critical for informed decision-making.

In this study, we propose a modified decision tree algorithm that integrates inefficiency measures from a DEA model into the node-splitting process. Traditional decision trees rely on metrics such as the Gini index or Entropy to select splits based solely on class distributions. These measures do not account for the relative performance or efficiency of the observations. By incorporating inefficiency scores from a DEA model, each observation is assigned a weight reflecting its underperformance. It allows the algorithm to consider both class purity and operational performance, providing splits that reflect not only the labels but also the efficiency of units.

The inefficiency scores are derived from a DEA model that measures input excesses and output shortfalls, highlighting the most critical areas where each unit underperforms. During tree construction, these scores are used to compute a weighted impurity measure for candidate splits, replacing the standard Gini index. Observations with higher inefficiency contribute more to the impurity, making the algorithm more sensitive to underperforming units. The resulting decision tree provides a dual-purpose output: it classifies observations accurately while simultaneously revealing inefficiency patterns across the dataset, enabling decision-makers to identify critical areas for improvement.

The following algorithm outlines a DEA-informed decision tree approach that integrates inefficiency scores from a DEA model into the tree construction process. Each step demonstrates how inefficiency weighting guides node splitting, recursive tree building, and final leaf evaluation to combine classification accuracy with performance assessment.

Step 1. Compute inefficiency: apply the DEA model to each DMU to calculate input and output inefficiency scores.

Step 2. Assign weights: assign each observation a weight proportional to its inefficiency.

Step 3. Evaluate splits: for each candidate feature, compute a weighted impurity measure using the inefficiency scores. Select the split that maximizes impurity reduction.

Step 4. Recursive tree construction: repeat the splitting process recursively for child nodes until stopping criteria are met (e.g., minimum node size or threshold impurity).

Step 5. Leaf interpretation: at each leaf, report the average inefficiency of contained samples to highlight branches with high underperformance.

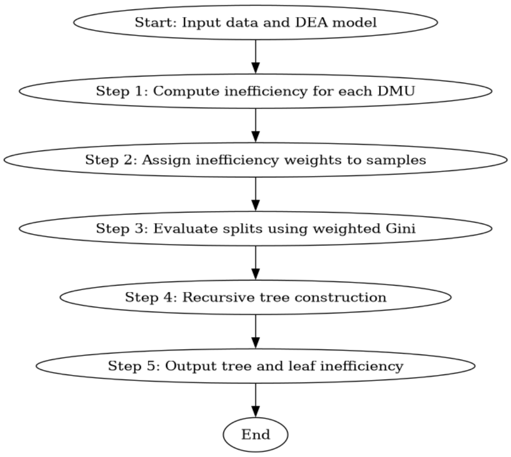
Start: Input data and DEA model
Step 1: Compute inefficiency for each DMU
Step 2: Assign inefficiency weights to samples
Step 3: Evaluate splits using weighted Gini
Step 4: Recursive tree construction
Step 5: Output tree and leaf inefficiency
End

Fig. 1. DEA-informed decision tree algorithm.

The flowchart illustrates the proposed DEA-informed Decision Tree algorithm, which integrates efficiency analysis into a classification model. It begins with data input and the DEA model, followed by the computation of inefficiency scores for each DMU. These inefficiency scores are then assigned as weights to the corresponding samples, influencing the node-splitting process. The algorithm evaluates candidate splits using a weighted impurity measure that accounts for both class distribution and inefficiency, and recursively constructs the tree until stopping criteria are met. Finally, the flowchart shows that the completed tree outputs the classification structure along with the average inefficiency at each leaf, highlighting the branches with the highest underperformance for decision-making.

4 | Case Study: Hospital Efficiency Modeling

This study utilizes a dataset comprising 500 bank branches, each considered as a DMU. For each branch, four financial indicators are used as explanatory variables: Credit, Deposits, Loans, and Profit, which collectively reflect resource allocation, intermediation activity, and financial performance. In addition, a DEA-based inefficiency measure obtained from a collective (ensemble) DEA model is incorporated as the target variable. This inefficiency score represents the relative distance of each branch from the efficiency frontier and serves as a quantitative indicator of operational underperformance. The dataset is structured in a cross-sectional format and prepared for integration with machine learning models.

The DEA-informed decision tree analysis identifies Loans as the most influential variable in explaining inefficiency, as it consistently appears at the root and upper levels of the tree. Subsequent splits highlight the joint role of Deposits and Profit, indicating that inefficiency is not driven by a single factor but by the interaction among financial indicators.

Branches with balanced loan volumes, sufficient deposits, and higher profitability tend to exhibit lower inefficiency scores. In contrast, branches with weak profitability or disproportionate credit–loan structures are associated with higher inefficiency. The resulting decision rules provide a transparent and interpretable mapping between financial conditions and DEA-based inefficiency outcomes.

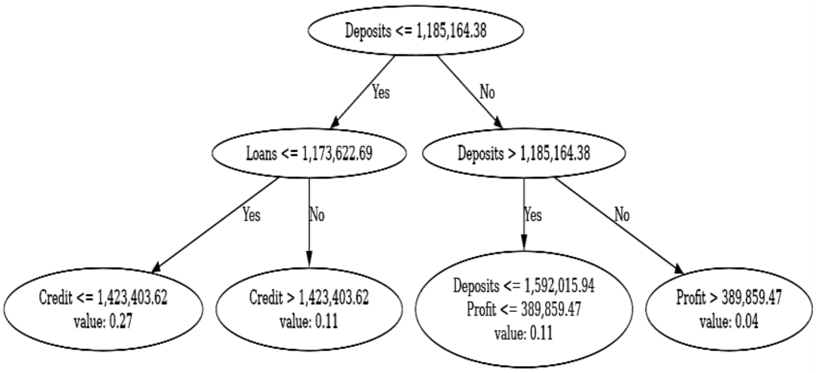

Fig. 2. DEA-informed decision tree (three-layer summary).

The first layer of the decision tree is split based on the "Deposits" attribute. The left branch represents units with deposits less than or equal to 1,185,164.38, while the right branch represents units with deposits higher than this threshold. This initial split highlights key differences in inefficiency among the units and establishes the foundation for subsequent analyses. In the second layer, the left branch is further divided based on "Loans," while the right branch uses a combination of "Deposits" and "Profit." On the left, units are classified by loan amount, allowing identification of groups with varying inefficiencies. On the right, deposits and profit levels are used to create more precise groupings, thereby facilitating clearer distinctions between efficient and inefficient units. The third layer provides detailed decision rules using "Credit" and "Profit" to determine the final inefficiency value at each leaf. Each terminal node displays the numerical inefficiency corresponding to that group of units. This layer clearly illustrates how input-output combinations affect efficiency, enabling the identification of inefficiency patterns for informed managerial planning.

5 | Conclusion

This research highlights the benefits of combining DEA with decision tree models to assess the performance of bank branches. Incorporating DEA-derived inefficiency scores as the target variable allows the approach to merge precise efficiency evaluation with a transparent, interpretable machine learning model. Analysis shows that financial factors, including deposits, loans, credit, and profit, significantly influence branch inefficiency. The DEA-informed decision tree structure clearly captures the operational patterns associated with inefficiency, helping managers identify specific processes or branches that require attention. For instance, branches with lower deposit levels or uneven loan distribution tend to be less efficient, providing actionable insights for improvement.

In summary, this integrated methodology serves as an effective decision-support tool, translating complex efficiency metrics into understandable guidance. It facilitates informed managerial decision-making, supports resource-allocation decisions, and enables targeted strategies to enhance branch performance and operational efficiency.

Conflict of Interest

The authors declare no conflict of interest.

Data Availability

All data are included in the text.

Funding

This research received no specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

[1] Charnes, A., Cooper, W. W., & Rhodes, E. (1978). Measuring the efficiency of decision making units. *European journal of operational research*, *2*(6), 429–444. https://doi.org/10.1016/0377-2217(78)90138-8

[2] Cook, W. D. (2001). *Data envelopment analysis: A comprehensive text with models, applications, references and DEA-solver software*. JSTOR. https://doi.org/10.1007/978-0-387-45283-8%0A%0A

[3] Chen, Y., & Zhu, J. (2004). Measuring information technology's indirect impact on firm performance. *Information technology and management*, *5*(1), 9–22. https://doi.org/10.1023/B:ITEM.0000008075.43543.97

[4] Azadeh, A., Ghaderi, S. F., & Sohrabkhani, S. (2008). A simulated-based neural network algorithm for forecasting electrical energy consumption in Iran. *Energy policy*, *36*(7), 2637–2644. https://doi.org/10.1016/j.enpol.2008.02.035

[5] Banker, R. D., Charnes, A., & Cooper, W. W. (1984). Some models for estimating technical and scale inefficiencies in data envelopment analysis. *Management science*, *30*(9), 1078–1092. https://doi.org/10.1287/mnsc.30.9.1078

[6] Daraio, C., & Simar, L. (2005). Introducing environmental variables in nonparametric frontier models: a probabilistic approach. *Journal of productivity analysis*, *24*, 93–121. https://doi.org/10.1007/s11123-005-3042-8

[7] Daouia, A., Noh, H., & Park, B. U. (2014). Data envelope fitting with constrained polynomial splines. *Journal of the royal statistical society series b: statistical methodology*, *78*(1), 3–30. https://doi.org/10.1111/rssb.12098

[8] Breiman, L. (1996). Bagging predictors. *Machine learning*, *24*(2), 123–140. https://doi.org/10.1007/BF00058655

[9] Breiman, L. (2001). Using iterated bagging to debias regressions. *Machine learning*, *45*(3), 261–277. https://doi.org/10.1023/A:1017934522171

[10] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *KDD '16: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 785–794). Association for Computing Machinery. https://doi.org/10.1145/2939672.2939785