基於決策樹之 Catboost 模型簡化及可解釋性提升之研究

王華燁 [1]，劉立頌 [2]，林楓貿 [3]
[123] 國立中正大學電機工程系
[1]E-mail: huaye@alum.ccu.edu.tw
[2] E-mail: aliu@ee.ccu.edu.tw
[3] E-mail: aaa20000913@alum.ccu.edu.tw

摘要

隨著人工智慧的迅速發展，市場應用對於可解釋性的需求已隨之攀升，而由於準確性高的模型通常在可解釋性方面表現不佳，故因此本研究目的為提高模型可解釋性和維持原本模型的準確性上，本研究提出一種透過剪枝和合併決策路徑來將 CatBoost 模型簡化為單一決策樹的方法，讓使用者可以更直觀地理解模型如何做出決策，從而提升模型的透明度和信任度。而實驗透過不同資料集對提出的方法進行驗證，結果顯示簡化後的模型有保持良好的準確性和提升可解釋性。

關鍵字：人工智慧、CatBoost、決策樹、可解釋性、模型簡化

Abstract

With the rapid development of Artificial Intelligence, the demand for interpretability in market applications has risen, and since models with high accuracy usually perform poorly in terms of interpretability, the purpose of this study is to improve model interpretability and maintain the accuracy of the original model, this study proposes a method to simplify the CatBoost model into a single decision tree by pruning and merging decision paths, so that users can understand how the model makes decisions more intuitively, thus improving model transparency and trust. Users can understand how the model makes decisions more intuitively, thus enhancing the transparency and trust of the model. Experiments are conducted to validate the proposed method on different datasets, and the results show that the simplified model maintains good accuracy and improves the interpretability.

Keywords：Artificial Intelligence, CatBoost, Decision Tree, Interpretability, Model Simplification

1. 緒論

隨著數據驅動的 AI 在過去十年中迅速發展，如 DNN 和隨機森林等 AI 模型在醫學、交通等領域上運用越來越普遍。雖然上述模型準確性迅速提升，但模型複雜性往往也隨之增加，使得其內部運作機制難以理解而形成「黑盒模型」，即難以判斷模型結果的生成原因。而因應市場對可解釋的需求，可解釋人工智慧 (Explainable AI, XAI) 逐漸成為一個重要的研究領域[1]。

根據 Arrieta 等人[2]的研究，機器學習 (Machine Learning, ML) 模型可以區分為本質可解釋模型與事後可解釋模型。本質可解釋模型如線性迴歸 (linear regression)、決策樹 (decision trees) 等，因其簡單透明的結構，本身即具有較高的可解釋性，能提供高度可預測且易於解釋的結果。然而隨著資料集複雜性增加，本質可解釋模型可能無法充分補捉資料集的非線性關係，為此事後可解釋模型被引入，如樹集成模型、CNN 和 RNN 等，這些模型雖然能更好處理複雜數據，但需藉助例如 LIME[3]、SHAP[4]等技術來解釋其內部運作。

而在實際應用中，不同場景對模型的要求不同，有些場景要求模型具有高可解釋性，以便使用者能夠理解和信任模型的決策。為此需要選擇更易解釋的模型，但這些模型的準確性可能不如其他複雜模型，故準確性和可解釋之間的平衡便是當下重要的研究議題。在可解釋方面，樹集成模型優於神經網路。雖然它們屬於黑盒模型，但可以通過追溯決策路徑清楚了解哪些特徵組合影響了最終決策。相比之下，神經網路的決策過程隱藏在複雜的參數和激活函數中，因此難以解釋。故本研究選擇樹集成模

型來探討如何有效地簡化保持高準確性的表現同時提升其可解釋性。

本論文分為五個章節，第二章為文獻探討，先針對 ML 模型的可解釋性進行分類，接著介紹不同的 ML 演算法在準確性與可解釋性之間的權衡；第三章為研究方法，提出了本論文的系統架構和設計細節；第四章為實驗與驗證，利用不同的資料集驗證本研究所提出方法的可行性及可解釋性；最後第五章為結論與未來展望，總結研究內容並提出未來改進方向。

2. 文獻探討

本研究重點主要集中在樹集成模型的可解釋性上並探討樹集成模型在準確性與可解釋性的關係以及幫助樹集成模型提升其可解釋性的方法

◦2.1 準確性與可解釋性之關係

在 ML 中，準確性與可解釋性兩者之間常需要進行權衡。本質可解釋模型因其決策過程相對簡單且透明通常具有較高的可解釋性，但在處理複雜非線性數據時，表現可能不佳。為此會選擇內部機制更複雜的事後可解釋模型，導致其可解釋性較差。但本質可解釋模型也並非都是高可解釋性，Biecek 等人提出本質可解釋模型的可解釋性會因模型加深而下降 [6]。例如，決策樹深度的增加，其決策過程涉及的分支和條件判斷變得越來越複雜，從而導致難以直觀理解模型的決策邏輯; 反之，提升可解釋性可能限制模型複雜度導致準確性下降。

2.2 樹集成模型可解釋性之研究比較

樹集成模型通過結合多個決策樹來提高預測性能，但其可解釋性往往面臨挑戰。如 Neto 和 Paulovich 提出了一個類似矩陣的視覺隱喻來解釋基於規則的隨機森林模型，其中邏輯規則是行，特徵是列，規則謂詞是單元格[7]。但此方法面對更為複雜模型時，會導致全局信息過於龐大而難以理解。此外沒有解決準確性與可解釋性間的平衡問題。而針對 boosting 可解釋性的方法，如 Hatwell 等人提出自適應加權高重要性路徑片段 (Ada-WHIPS)，其做法是模型的內部決策樹的各個決策節點中的權重尤其由 Ada-WHIPS 重新分配。然後透過對加權節點進行簡單的啟發式搜尋來發現主導模型選擇的單一規則[8]。而盡管其在解釋 AdaBoost 模型方面效果顯著，但因局限相比於 LIME[3]、SHAP[4]等無關模型可解釋技術對其他模型的泛用性不足。總體而言，在樹集成模型的可解釋性研究中，兼顧泛用性和解釋性與準確性的平衡，仍是需要克服之挑戰。

2.3 樹集成模型之學習策略

Mienye 等人[9]對集成學習進行了全面概述，並分為三大主流方法：bagging、boosting 和 stacking。這些方法使用的基礎學習器也稱作弱學習器，其單獨預測能力較弱，但能藉由組合來提升其準確性。例如，bagging 方法中的隨機森林、boosting 方法中的 AdaBoost 及其後續的 XGBoost、LightGBM 和 CatBoost，都是以決策樹為基礎學習器的應用。Stacking 則是將不同類型的基礎學習器的預測結果輸入到一個元模型中進行整合。而樹集成模型是集成學習的一種具體應用，通過上述不同的集成學習策略來提升模型表現。而由於 bagging 模型偏差限制緣故，模型性能較差、stacking 將多層模型融合來預測的做法則會導致可解釋性下降，故本文選擇 boosting 的方法繼續探討。而 Bentéjac 等人[5]做了一系列的實驗對 XGBoost、LightGBM 和 CatBoost 三種方法從泛化性、準確性、訓練速度和超參數的調整等方面上探討不同演算法的特徵與其優缺點。其中 CatBoost 在不同資料集上的泛化性和準確性都較佳。因此本研究選擇以 CatBoost 此樹集成演算法作為模型簡化的對象。

2.4 樹集成模型簡化之研究比較

早期關於簡化樹集成的研究主要著重在減少基礎學習器數量，以提高計算效率和降低記憶體消耗。但隨著近年來可解釋性需求增加，在促進保值可解釋上主要是將複雜的樹集成模型簡化為單一決策樹，例如透過特定的演算法選擇和剪枝技術在

保留模型準確性的同時，顯著地簡化模型結構，從而提高其可解釋性。

而樹集成模型的簡化技術主要集中在剪枝方法，通過適當的剪枝，能在不顯著降低模型準確性的前提下，減少基礎學習器的數量集提升模型透明度。Rose 等人[10]詳細調查了隨機森林的剪枝技術，包括基於排名、聚類和最佳化的方法。指出基於排名的方法計算複雜度最低且最易於解釋，而基於最佳化的方法計算複雜度最高且最難以理解，而剪枝技術在提高可解釋性方面具有顯著的優勢。而 Weinberg 等人[11]提出從決策樹集成模型中選擇代表性單一決策樹的方法，透過將大資料集切成小資料集並訓練多個決策樹，再根據語法和語意相似性挑選最具代表性的樹。但此方法因決策樹準確性參差不齊導致挑選的樹準確性不夠穩定。而 Vidal 等人[12] 提出了基於動態規劃演算法，將隨機森林模型簡化為最小尺寸單一決策樹的方法。Sagi 等人[13]則提出了一種將 XGBoost 模型近似為單一決策樹的方法，通過剪枝和決策節點組合來構建簡化模型。Sagi 的研究中，他們將樹集成模型簡化為決策樹的過程是透明且可解釋的，提出的 conjunction 方法保留了原有模型的決策過程。故本研究透過使用 Catboost 作為基線模型結合基於排名的剪枝方法並採取 Sagi 的方法生成 conjunction 來構建決策樹。通過這些技術的結合，本研究成功地將 CatBoost 模型簡化為一個具有高可解釋性的單一決策樹，同時保持了原有模型的高準確性，為實際應用提供了一個兼具效率與可解釋性的解決方案。

3 研究方法

本章將介紹研究架構的流程，如圖 1 所示，而其內容會在各章節介紹。

3.1 Catboost 模型

本文將資料集以 8:2 的比例區分訓練集和測試集。CatBoost 模型使用訓練集來訓練，生成由多個決策樹組成的集成模型。通過逐步增加基礎學習器來提高準確性，直到達到設定的迭代次數或停止條件。而每個基礎學習器都是 Oblivious Decision Tree（ODT），其特點是每層的分裂規則相同。本研究中，ODT 的深度設定為 7，故會有 64 個葉節點。

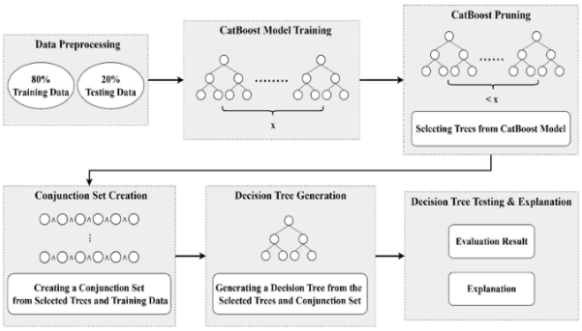


圖 1. 研究架構圖

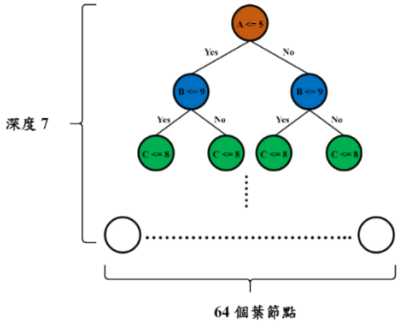


圖 2.ODT 模型架構圖

3.2 Catboost 剪枝

CatBoost 剪枝的目的是為了剔除掉 CatBoost 模型中冗餘的基礎學習器並減少後續的計算量，剪枝完後所挑選出來的樹在本研究中稱為候選樹。我們採用了 Rose 等人提到的基於排名的方法來挑選候選樹。而本文在挑選樹排名的評估方法是使用均方誤差 (Mean Square Error, MSE)。透過比較每顆基礎學習器的預測值與真實標籤之間的誤差平方何的平均來計算。

本文制定了兩套挑選 ODT 的剪枝規則。第一個規則是根據添加每棵樹對 MSE 的影響來選擇，首先計算每次添加新的樹後整體模型的 MSE 並保存，接著計算每棵樹對 MSE 的改進量並依此排序和設定百分位數進行挑選。第二個規則則是根據逐步移除樹後對 MSE 的影響來挑選樹，首先會先計算包含所有樹完整模型時的 MSE，然後逐一移除每棵樹並計算移除後的 MSE 變化量並以此來進行排序，選擇移除後對 MSE 大影響的樹。經過上述兩個規則挑選的樹還需進行聯集，才會得到最終候選樹集合。

3.3 Conjunction Set 創建

針對單一決策樹的訓練集要從候選樹和訓練集中創建 conjunction set。而本文使用兩套創建規則產生 conjunction 來組成 conjunction set。第一個規則是透過訓練集和候選樹來產生 conjunction，第二個規則是對候選樹路徑兩兩合併來產生 conjunction。

第一個創建 conjunction 的規則會從訓練集遍歷所有候選樹。在每棵候選樹中，若該筆資料到達了某個候選樹的葉節點，此時會記錄下該筆資料經過的所有決策節點並將這些決策節點組合起來形成 conjunction。將其添加至 conjunction set 中。透過該過程所產生的 conjunction 稱為 training-data-conjunctions。第二個規則將候選樹中的每條路徑轉換為 conjunction 逐一合併這些 conjunction，並檢查是否存在合併衝突(如相同特徵的範圍互斥)。若無衝突，節點組合會成為新 conjunction；若有衝突則合併失敗。葉節點的合併是將其數值相加。此外為了減少計算量和複雜性，本文使用經驗累積分佈函數來篩選多餘的 conjunction。該規則生成的 conjunction 稱為 merged-conjunctions。

3.4 決策樹產生

首先會先遍歷所有候選樹，記錄每個決策節點上的分裂特徵和分裂值。接著，使用加權變異數如式(1)，來計算最佳的分裂特徵和分裂值。這一過程中，我們依次遍歷各個特徵及其對應的特徵值，將 conjunction 分為左右子集，並用 conjunction 葉節點計算加權變異數。加權變異數越小，分裂效果越好。最小加權變異數所對應的特徵和特徵值就會被選為節點的分裂特徵和分裂值，如 Alg. 1 所示。透過遞迴以上過程來生成完整的決策樹並使用加權變異數來確保分裂點的選擇合理性及提高決策樹的準確性。

3.5 決策樹測試和解釋

對於決策樹測試，本文利用測試集來評估決策樹模型的準確性。準確性評估指標包括 MSE、RMSE 和 MAE，將生成的決策樹與 CatBoost 模型進行比較以確保簡化後模型預測的準確性無顯著下降。此外我們使用 PredictionValuesChange (PVC)。PVC 演算法是計算當特徵值變化時，對預測結果的平均變化量來理解特徵對模型的影響。為增強模型解釋性，我們結合本質可解釋性的解釋和事後可解釋技術，在本質可解釋性上，進行決策樹視覺化，並使用相同顏色標示相同特徵的決策節點。而事後可解釋性方面，透過分析葉節點數值分佈，幫助使用者理解模型輸出及特徵影響。

$$weight\_variance = \frac{\frac{(len(left\_leafs)*left\_variance+}{(len(right\_leafs)*right\_variance}}{(len(left_{leafs})+(len(right\_leafs)} \quad (1)$$

Algorithm 1 Determine Optimal Feature and Split Value for Minimum Weighted Variance

```
function SelectBestFeatureAndSplitValue(conjunction_set, feature_splits)
    Initialize min_variance ← infinity
    Initialize best_feature, best_value, best_left_conjunctions,
            best_right_conjunctions ← None
    for each feature, values in feature_splits do
        for each value in values do
            left_conjunctions, right_conjunctions, variance = CalculateVarianceForSplit(conjunction_set, feature, value)
            if variance < min_variance then
                min_variance = variance
                best_feature = feature
                best_value = value
                best_left_conjunctions = left_conjunctions
                best_right_conjunctions = right_conjunctions
            end if
        end for
    end for
    return best_feature, best_value, best_left_conjunctions,
            best_right_conjunctions
end function
```

4. 實驗與驗證

本章節選取了三個不同的資料集，並通過以下實驗對模型的準確性、特徵重要性和可解釋性進行了深入分析。

4.1 資料集介紹

本研究共使用了三個資料集，為了進行較全面的分析，我們根據特徵數量將資料集進行挑選，以 10 個特徵作為界線，分別為 Auto MPG、Abalone 一組和 Wine Quality 一組，這些資料集皆是從 UCI ML 儲存庫 (UCI Machine Learning Repository)[14] 中所挑選。Auto MPG 資料集具有 7 個特徵，共 392 筆資料，目的是預測汽車的燃油效率。Abalone 資料集包含 8 個特徵，共 4177 筆資料，預測目標是鮑魚的年齡。最後，Wine Quality 資料集有 11 個特徵，共 6497 筆資料，旨在預測葡萄酒的品質。

4.2 實驗設計

本實驗分為三部分，實驗一為驗證不同資料集產生的決策樹準確性是否與 Catboost 相近，實驗二為驗證不同資料集產生決策樹與 Catboost 在特徵重要性排序是否相近，實驗三為簡化後決策樹可解釋性分析。

4.2.1 決策樹準確性驗證

實驗目的是在不同資料集驗證本研究是否能有效簡化 CatBoost 使其生成的決策樹模型在準確性與 CatBoost 相近。實驗設定上決策樹和 CatBoost 深度都為 7，而超參數如候選樹百分位數和 merged-conjunctions 的限制數量依不同資料集的最佳準確性來設定。決策樹的 MAE 和 RMSE 均比 CatBoost 低 25%以內，而百分比數越低代表在準確率上跟原始 Catboost 越相近。然而在 MSE 方面，決策樹模型的百分位數較高也就是準確度較差，這可能是由於決策樹模型複雜度不足及對離群值的敏感性所導致。，如表 1 所示，表中數字為決策樹模型準確性減去 CatBoost 模型準確性之後除以 Catboost 準確性來得出。

dataset	MSE 準確性比較	MAE 準確性比較	RMSE 準確性比較
Auto MPG	+42.12%	+23.76%	+19.21%
Abalone	+12.52%	+10.77%	+6.08%
Wine	+27.48%	+15.08%	+12.89%

表 1.不同資料集 CatBoost 與決策樹準確性比較

4.2.2 決策樹特徵重要性驗證

實驗目的為檢驗決策樹模型是否能捕捉原始 Catboost 模型的關鍵特徵和特徵重要性排序是否一致。實驗設置與實驗一相同，而實作經驗顯示超過 20 會使特徵重要性的排序差異過大，故實驗設定 CatBoost 模型與決策樹模型在單項 PVC 特徵重要性上的差異不超過 20。結果可以發現相比於 Catboost，因簡化後的決策樹模型決策節點有限，會集中於高特徵重要性的特徵，低特徵重要性的特徵很難被考慮進來，有些特徵 PVC 值甚至為 0 也可看到資料集的單項 PVC 差異均小於 20 ，PVC 差距越小則該特徵在兩個模型中的重要性保持更一致，從而說明簡化模型有正確捕捉原始 Catboost 的關鍵特徵。在 PVC 比較圖中，特徵重要性按 Catboost 的 PVC 值高到低排列，藍色線代表 Catboost 在各個特徵所對應的 PVC 值，橘色線則是決策樹在各個特徵所對應的 PVC 值如表 2 所示。

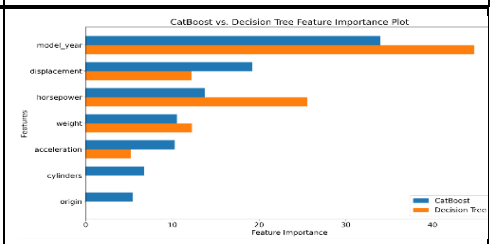
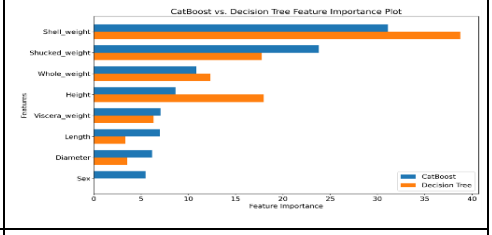
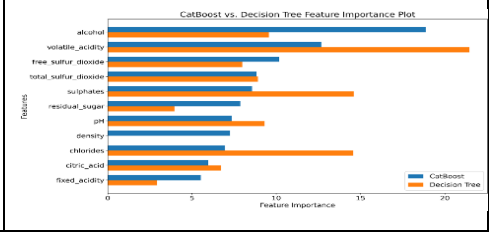
資料集	PVC 比較圖
Auto Mpg	
Abalone	
Wine	

表 2. CatBoost 與決策樹 PVC 特徵重要性比較

4.2.3 決策樹之可解釋性分析

實驗目的為決策樹提供本質可解釋和事後可解釋。而由於 Auto MPG 資料集的數據特徵關聯性較強，故本實驗結果以 Auto MPG 來呈現。通過模型視覺化來提高決策樹的解釋性。對於決策節點，若相同特徵使用相同顏色。而葉節點則根據數值大小調整顏色深淺，數值越大顏色越深，從圖 3 可觀察到將決策節點以及葉節點塗上顏色更有利於使用者的路徑追溯以及對模型決策過程的理解。針對事後可解釋，我們對葉節點的決策路徑分析，幫助使用者理解模型輸出的結果中哪些特徵被優先考慮。而在葉節點路徑分析會先以葉節點值做排序後再取前 15 個葉節點值較大的路徑來做說明，而其中有 14 條路徑 horsepower 低於 84.5 和 12 個的 model_year 在 1977 年以後。對此可判斷，這些

horsepower 低於 84.5 且 model_year 在 1977 後的車輛具有較高的 MPG 可能更省油。

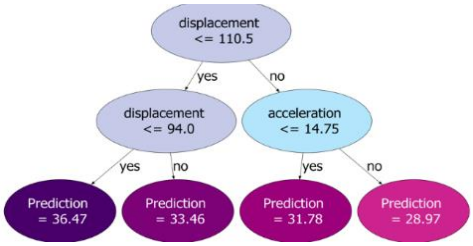


圖 3.決策樹視覺化

5. 結論與未來展望

本研究成功將 CatBoost 模型簡化為單一決策樹，並在多個資料集上驗證其有效性。結果顯示，簡化後的模型在資料集上皆表現良好。本文設計了兩套基於排名的剪枝規則，以保留重要的基礎學習器並創建 Conjunction Set，從而保留了原模型的決策邏輯，並確保生成的決策樹符合合理性。簡化後的單一決策樹提高了模型的直觀性和可視化解釋。未來將進一步優化剪枝策略，並在更多資料集和應用場景中驗證方法，以提升模型的適用性和穩健性。

6. 參考文獻

[1] D. Castelvecchi, “Can we open the black box of AI?,” Nature News, vol. 538, no. 7623, p. 20, 2016.

[2] A. B. Arrieta et al., “Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI,” Information fusion, vol. 58, pp. 82–115, 2020.

[3] M. T. Ribeiro et al., “‘ Why should i trust you?’ Explaining the predictions of any classifier,” in Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, 2016, pp. 1135–1144.

[4] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” Advances in neural information processing systems, vol. 30, 2017.

[5] C. Bentéjac et al. “A comparative anal1620 ysis of gradient boosting algorithms,” Artif. Intell. Rev., vol. 54, no. 3, 1621 pp. 1937–1967, 2021.

[6] P. Biecek and W. Samek, “Explain to Question not to Justify,” arXiv:2402.13914, 2024.

[7] M. P. Neto and F. V. Paulovich, “Explainable Matrix - Visualization for Global and Local Interpretability of Random Forest Classification Ensembles,” IEEE Trans. Vis. Comput. Graph., vol. 27, no. 2, pp. 1427–1437, Feb. 2021

[8] J. Hatwell et al “Ada-WHIPS: explaining AdaBoost classification with applications in the health sciences,” BMC Med. Inform. Decis. Mak., vol. 20, no. 1, p. 250, Dec. 202

[9] I. D. Mienye and Y. Sun, “A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects,” IEEE Access, vol. 10, pp. 99129–99149, 2022.

[10] M. Rose and H. R. Hassen, “A Survey of Random Forest Pruning Techniques,” 9th in ICCSEA 2019.

[11] A. I. Weinberg and M. Last, “Selecting a representative decision tree from an ensemble of decision-tree models for fast big data classification,” Journal of Big Data, vol. 6, pp. 1–17, 2019

[12] T. Vidal and M. Schiffer, “Born-again tree ensembles,” in *ICML*, PMLR, 2020, pp. 9743–9753.

[13] O. Sagi and L. Rokach, “Approximating XGBoost with an interpretable decision tree,” Information Sciences, vol. 572, pp. 522–542, 2021.

[14] Dua, D. and Graff, C. (2019). UCI Machine Learning Repository

7. 誌謝

感謝國科會在研究經費上的補助，計畫編號：112-2221-E-194-014-MY3。