

Decision Trees

The CART Decision Tree used to classify numerical Wine Quality data

My dataset of choice was the UCI Machine Learning Repository Wine Quality Data Set, which contains quality ratings for wines along with their corresponding chemical measurements. This dataset was particularly compelling to use with decision trees (DTs), as it offers rich analytical possibilities through its 11 different chemical features. Predicting wine quality based on chemical properties is crucial for the wine industry, since small changes in chemical composition can greatly affect consumer enjoyment. The dataset choice demonstrates the practical and commercial application of DTs. While the Weather dataset was useful for initially testing how DTs handle categorical data, the Wine Quality dataset presented a more interesting challenge by combining numerical measurements with quality ratings

The Wine Quality dataset structure proved ideal for decision tree analysis. Its continuous features including acidity, alcohol content, and sulphates align naturally with CART DTs binary splitting mechanism, enabling the algorithm to identify precise quality prediction thresholds. These chemical properties are particularly valuable since winemakers can adjust them during production to enhance their product. The dataset quality ratings, which range from 0 to 10, make it especially suitable for DTs, as these algorithms excel at handling stepped relationships between values. This becomes particularly important when considering prediction errors, as it is less problematic if the model confuses adjacent quality levels (i.e. 6 and 7) than distant ones (i.e. 3 and 8). DTs natural ability to understand these ordered relationships provides a robust framework for quality assessment.

My development of hypotheses was based on established wine research and industry knowledge.

1. Alcohol content would be the strongest predictor of wine quality, since alcohol plays a fundamental role in how the wine feels and tastes.
2. Sugar levels would strongly influence quality ratings, particularly in determining how balanced and complex the wine tastes.
3. Sulphur dioxide levels would have minimal impact on quality predictions, since these compounds are mainly used to preserve the wine rather than enhance its taste.

After experimenting with both algorithms, CART DT proved more effective than ID3 for analysing wine quality. While ID3 handled categorical features well in the weather dataset, it struggled with the wine data, generating an overwhelming number of branches for the 11 continuous features that made interpretation difficult. In contrast, CART's binary splitting approach and Gini impurity calculations created clearer decision boundaries for the chemical measurements.

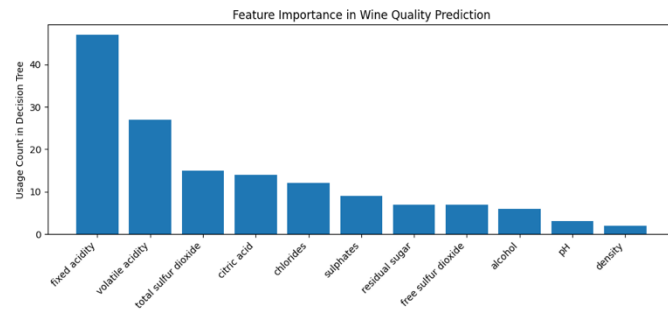
My CART model achieved 67.73% accuracy using a maximum depth of 7 and a minimum of 20 samples per split. This performance is particularly significant given the complexity of predicting across 11 quality classes - substantially outperforming the 9% baseline accuracy of random guessing. My analysis of the model's errors revealed that most misclassifications occurred between adjacent quality levels, indicating the model had effectively learned the underlying structure of wine quality assessment. By setting a depth limit of 7, I prevented the tree from becoming overly complex while maintaining interpretable decision paths. Similarly, my requirement of 20 samples per split struck a balance between statistical reliability and granular pattern detection. These tuned parameters enabled my model to extract meaningful insights from this complex, multi-class prediction task while avoiding overfitting.

When analysing which features mattered most, I found surprising results that challenged my initial hypotheses. Fixed acidity emerged as the most important predictor, appearing in 47 decision points, followed by volatile acidity with 27 points and total sulphur dioxide with 15 points. This directly contradicted my main hypothesis about alcohol content being the most important factor. Additionally, the significant impact of sulphur dioxide challenged my assumption about its minimal role, suggesting that preservative compounds might affect

perceived quality in more complex ways than previously thought. The relatively modest influence of residual sugar also went against my secondary hypothesis, indicating that sweetness plays a more subtle role in quality assessment than expected.

Beyond just accuracy numbers, the model provided practical value by connecting theoretical analysis with real-world

applications. The thresholds it identified offer clear guidance for winemakers to improve their processes, and the ranking of important features suggests where to focus future research. For instance, the unexpected importance of acidity-related variables suggests that collecting more detailed acid measurements could be valuable.



Overall, the Wine Quality dataset proved to be a fairly strong choice for implementing DTs. The CART algorithm effectively handled the continuous features, captured the ordered relationships in the data, and produced results that were easy to interpret. The model challenged several common assumptions about what determines wine quality and also revealed unexpected patterns. While the model's accuracy suggests room for improvement, its insights about important features and specific chemical thresholds provide substantial practical value for winemaking. This analysis demonstrated both the strengths of DTs and their ability to uncover surprising relationships in complex, real-world datasets, pointing toward new directions for both research and practical applications in the wine industry.

Recent Decision Tree Advances

Decision Trees (DTs) have been a fundamental tool in machine learning since their inception in the 1960s. Their popularity stems from their interpretability, robustness, and versatility across various applications. Over recent years, significant technological advances and increasing computational capabilities have led to considerable developments in DT methodology.

Modern DTs have evolved significantly beyond their traditional roots. Early DTs could only examine one attribute at a time through univariate splits, but newer approaches can analyse multiple attributes simultaneously. While this makes the trees more accurate, it also makes them harder to understand. Researchers have developed clever solutions to this problem. For example, techniques like Sparse ADTrees use advanced statistical methods to find better ways to split the data while keeping the calculations manageable. Another major improvement has been the development of model trees. Instead of simple decisions at their endpoints or leaves, these trees can use more sophisticated prediction methods, such as combining with logistic regression methodologies.

The field has made significant advances in improving DTs reliability and accuracy. Traditional techniques like pruning, where unnecessary branches are removed, remain valuable. However, researchers have developed more sophisticated approaches called optimal tree algorithms. Unlike older methods that build trees step-by-step, these new algorithms can identify the mathematically optimal tree structure. Two notable examples, Optimal Classification Trees and DL8.5, can find the best possible tree arrangement for medium-sized datasets while remaining computationally practical.

Another notable advancement in DTs has been seen in Gradient Boosted Decision Trees (GBDTs). Unlike traditional DTs, GBDTs combine the predictions of multiple sequentially trained weak learners, optimising the model by reducing errors with each iteration. These models leverage gradient descent to minimise specified loss functions, enabling efficient handling of both regression and classification tasks. Notable implementations include XGBoost, LightGBM, and CatBoost, which use techniques such as parallelisation, feature sparsity awareness, and GPU acceleration to enhance both training speed and model performance. These innovations enable GBDTs to process large datasets whilst maintaining high predictive accuracy, scalability, and flexibility.

The approach has expanded to address nuanced learning requirements, such as handling imbalanced data and integrating monotonic constraints to align predictions with domain-specific rules.

Research into DT interpretability has evolved beyond simple metrics. Although tree size traditionally served as the primary measure of interpretability, with shorter trees considered more understandable, recent studies reveal this relationship's complexity. Modern approaches include visual pruning techniques that enable direct interpretation without validation sets, and methods that prioritise broader, more accurate rules at higher tree levels. These developments enhance understanding without requiring extensive examination of deeper nodes.

A final innovation worth noting is in the recent development of quantum decision trees (QDTs). These maintain the directed graph structure of classical trees but employ quantum kernels at split nodes to generate separating hyperplanes in higher-dimensional quantum feature spaces. The use of Nyström quantum kernel estimation helps prevent overfitting while maintaining the benefits of quantum feature spaces. QDTs have shown particular promise in handling multi-class problems without additional computational overhead, unlike quantum Support Vector Machines (SVMs) which require specific strategies for multiple classes. Both theoretical analysis and numerical experiments have shown QDTs perform better than quantum SVMs while needing fewer kernel estimations.

A Real-World application of a Decision Tree

The Large Hadron Collider (LHC) demonstrates a particularly sophisticated implementation of DTs through Boosted Decision Trees (BDTs). BDTs combine multiple shallow DTs into a more robust predictive framework, operating iteratively with subsequent trees focusing on previously misclassified data points.

In the context of the LHC, these BDTs serve a crucial function in processing the extraordinary volume of data generated by proton collisions occurring at 25-nanosecond intervals. These collisions produce data volumes in the order of tens of terabytes per second, presenting substantial data management challenges. To address this, the facility employs a sophisticated filtering system, designated as the trigger, wherein the Level-1 Trigger (L1T) must process approximately 40 million events per second while maintaining decision-making capabilities within a 10-microsecond window.

A particularly noteworthy implementation of BDTs can be observed in the Compact Muon Solenoid (CMS) experiment's muon detection system, where the implementation has demonstrated threefold enhancement in the filtering of low-energy muons compared to preceding methodologies. Through the implementation of hls4ml, these BDTs demonstrate a very strong level of computational efficiency, processing information within 12 clock cycles (equivalent to 60 nanoseconds at 200 MHz), rendering them particularly suitable for real-time trigger system applications. In particle jet classification tasks, BDTs have demonstrated comparable performance to Deep Neural Networks while requiring reduced computational resources, positioning them as an optimal solution for physics applications that necessitate both rapid processing capabilities and high accuracy requirements.

These advancements in BDT methodology have provided physicists with enhanced analytical capabilities, facilitating improved jet classification, more precise muon energy measurements, and enhanced detection of rare particle events. The successful implementation of BDTs in addressing these complex physics challenges demonstrates their significant utility in scientific applications requiring both precise analytical outcomes and exceptional processing efficiency. This application exemplifies how theoretical advances in DT methodology can be successfully translated into practical solutions for complex scientific challenges.

References

- Cortez, P., Teixeira, J., Cerdeira, A., Almeida, F.D., Matos, T. and Reis, J., 2009. Using data mining for wine quality assessment. *Discovery Science, 12th International Conference, DS 2009*. Berlin: Springer. - https://doi.org/10.1007/978-3-642-04747-3_8
- Dataset Source: UCI Machine Learning Repository. (2009). Wine Quality Data Set. - <https://archive.ics.uci.edu/ml/datasets/wine+quality>
- GeeksforGeeks, n.d. CART (Classification And Regression Tree) in Machine Learning. *GeeksforGeeks*. - <https://www.geeksforgeeks.org/cart-classification-and-regression-tree-in-machine-learning/>
- Sri Kumar, M., Hill, C.D. and Hollenberg, L.C.L., 2024. A kernel-based quantum random forest for improved classification. *Quantum Machine Intelligence* - <https://doi.org/10.1007/s42484-023-00131-2>
- Costa, V.G. and Pedreira, C.E., 2023. Recent advances in decision trees: an updated survey. *Artificial Intelligence Review* - <https://doi.org/10.1007/s10462-022-10275-5>
- Summers, S., Di Guglielmo, G., Duarte, J.M., Harris, P., and others, 2020. Fast inference of Boosted Decision Trees in FPGAs for particle physics. - <https://doi.org/10.1088/1748-0221/15/05/P05026>
- Woodruff, K., 2017. Introduction to Boosted Decision Trees. Machine Learning Group Meeting, New Mexico State University - <https://indico.fnal.gov/event/15356/contributions/31377/attachments/19671/24560/DecisionTrees.pdf>
- Neptune.ai, 2023. Gradient Boosted Decision Trees: A Complete Guide - <https://neptune.ai/blog/gradient-boosted-decision-trees-guide>