Wine Quality Prediction

By Taya Bhavsar

Goal of Study

Main Objective: Determine which physico-chemical properties most accurately predict red wine quality

Research Questions

Three Research Questions:

1. Can quality be predicted using only 3 chemical measurements?
2. Is alcohol content alone sufficient, or are acidity measures necessary?
3. What minimum alcohol percentage gives >50% probability of good quality?

Dataset Overview

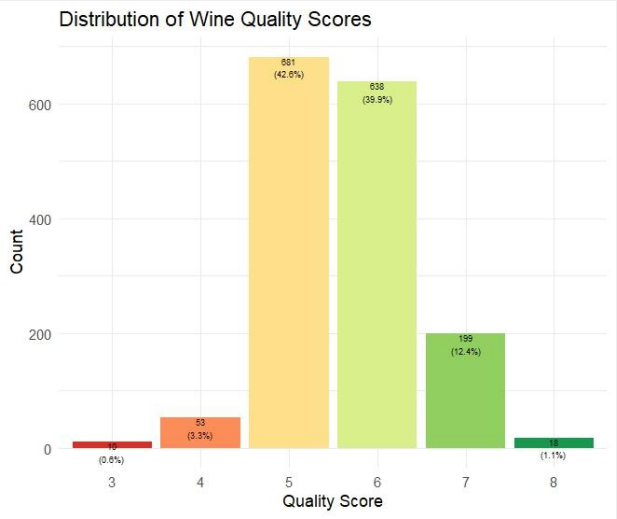
Source: Portuguese "Vinho Verde" red wines (UCI Wine Quality dataset)

Size: 1,599 samples, no missing values

Features: 11 continuous physicochemical properties (fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, alcohol)

Target: Quality scores 3-8 (mean: 5.636)

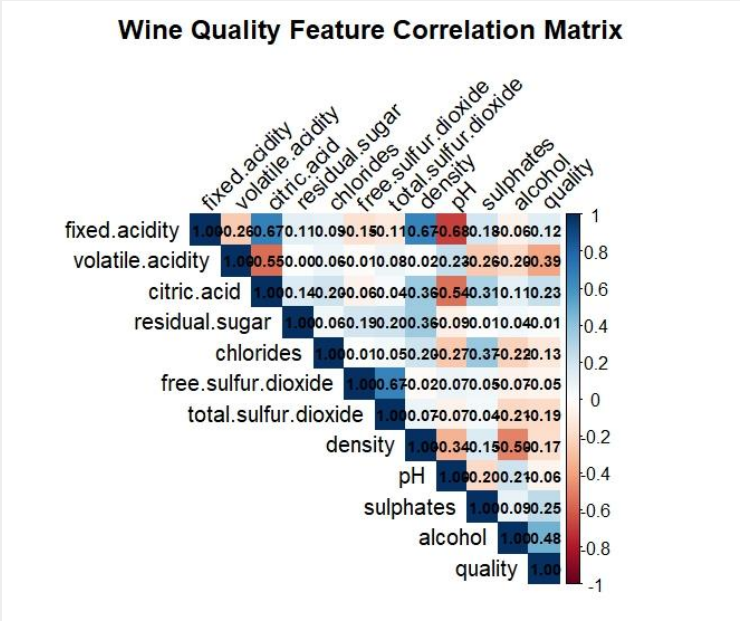
Key Challenge: Imbalanced dataset (46.5% poor quality <6, 53.5% good quality ≥6)



Wine Quality Correlation Matrix

Strongest Correlations (Dark Blue)

- **Fixed acidity ↔ Citric acid (0.67)**: Wines with higher fixed acidity tend to have more citric acid, reflecting natural grape composition
- **Fixed acidity ↔ Density (0.67)**: Higher acid content increases wine density
- **Fixed acidity ↔ pH (-0.68)**: This negative correlation is chemically expected—more acid means lower pH (more acidic)



Low multicollinearity risk

Models Used

Three complementary approaches:

1. **Logistic Regression**
   - Binary classification (Good ≥6 vs Bad <6)
   - Identifies most significant predictors
   - Good for interpretability
2. **Classification Trees**
   - Multi-class prediction (Low: 3-4, Medium: 5-6, High: 7-8)
   - Reveals critical thresholds for alcohol and volatile acidity
3. **Gradient Boosting/Random Forest**
   - Handles class imbalance and non-linear relationships
   - Captures complex chemical interactions
   - Handles complexity

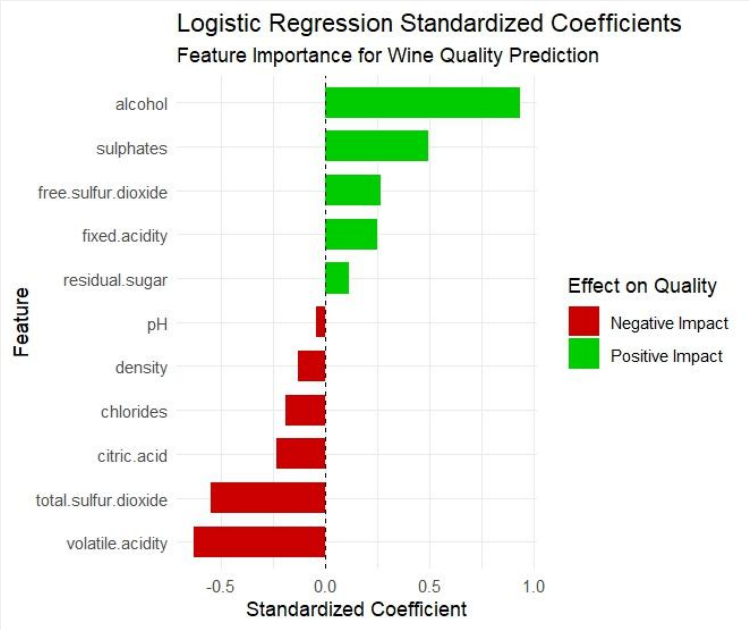
Evaluation: 80/20 train/test split with 1,279 training and 320 test samples using accuracy, precision, recall, F1-score

Main Findings

Training Set: 1,279 samples
Test Set: 320 samples

Top 3 Predictors Identified:
Logistic Regression

1. **Alcohol content** - strongest positive predictor
2. **Volatile acidity** - strongest negative predictor (vinegar-like flavors hurt quality)
3. **Sulphates** - positive effect on quality

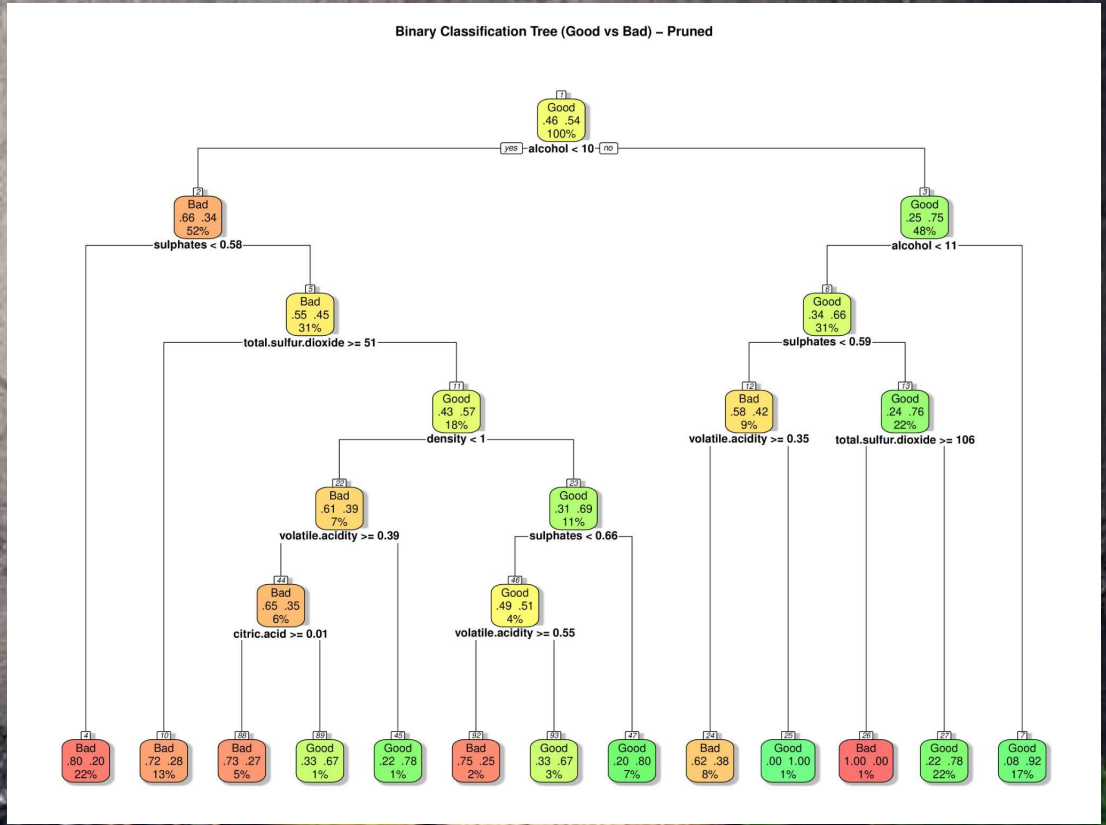


Binary Classification Tree

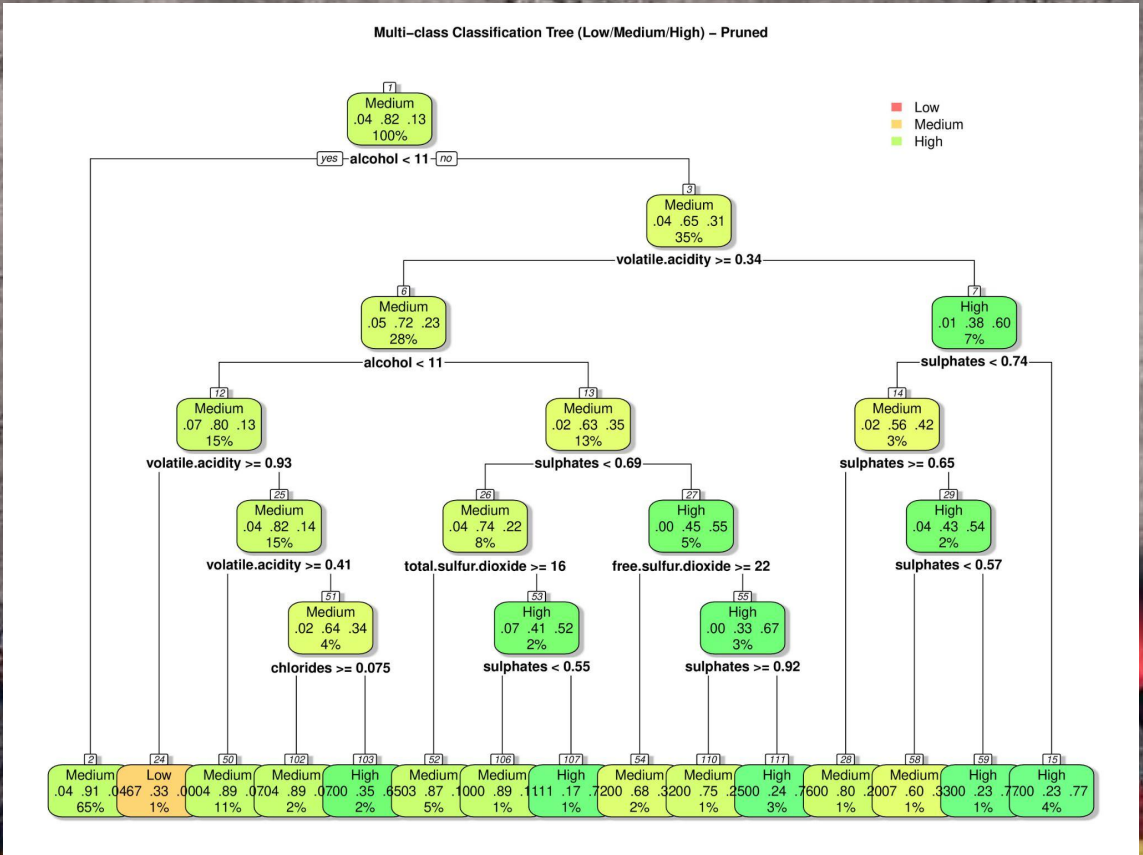
Accuracy: 74.38% - Successfully classifies 3 out of 4 wines correctly

AUC: 0.7963 - Good discriminatory power between quality classes

Balanced Performance: Sensitivity (**74.25%**) and Specificity (**74.51%**) are nearly equal, indicating the model performs well on both Good and Bad wines



Multi-Class Classification Tree



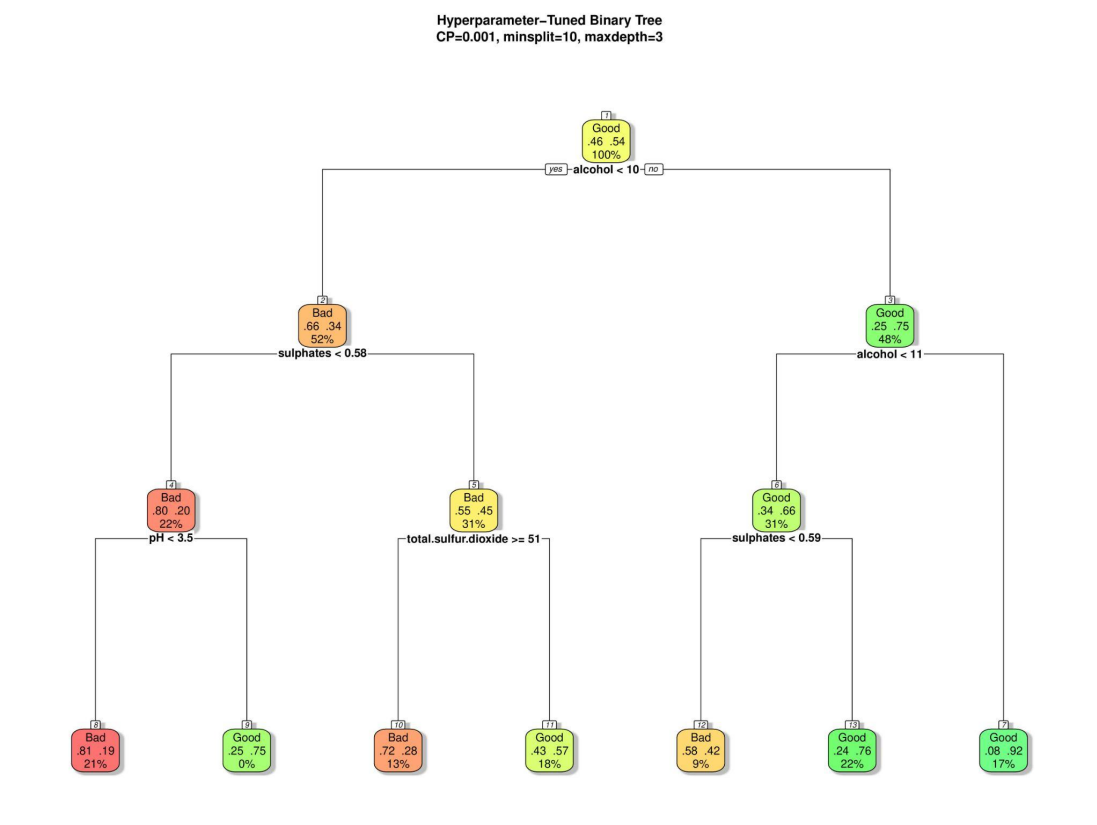
Overall Accuracy: 86.25% - Impressive performance

Strong at Medium Quality: 94.38% sensitivity for Medium wines (the majority class)

Higher-Alcohol Wines: alcohol content is the strongest overall predictor

Lower-alcohol wines: volatile acidity becomes the main driver

Hyperparameter-Tuned Binary Tree



Tuned Model Performance: 72.5% accuracy

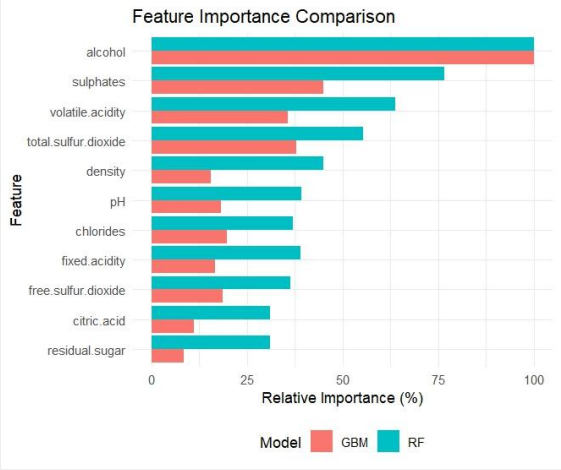
Slightly lower than pruned model (74.38%), but achieved better **Recall** (**79.04%**) at the expense of precision

Trade-off: The tuned model is more sensitive (catches more Good wines) but less precise (more false positives)

Gradient Boosting and Random Forest

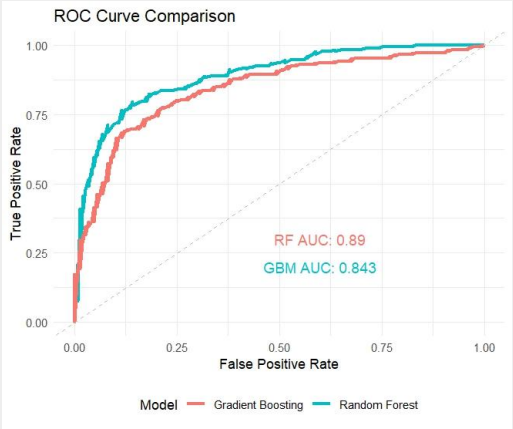
Consistent Top Predictors (Both Models Agree):

- **Alcohol** dominates in both models as the strongest predictor
- **Sulphates** ranks as the second most important feature
- **Volatile acidity** maintains strong importance



Random Forest emerges as the superior model with a test accuracy of **81.19%** compared to GBM's **77.74%**. The RF model demonstrates:

- Higher precision (**81.71% vs 79.41%**)
- **Better recall** (83.63% vs 78.95%)
- Superior F1-score (82.66% vs 79.18%)
- Notably stronger AUC (0.89 vs 0.843)



However, there's a concerning signal: RF shows **perfect training accuracy** (**100%**), suggesting potential overfitting, This is why GBM's **training accuracy** of **85.86%** indicates better generalization characteristics - it learns from errors of earlier trees, avoiding overfitting

Alcohol Threshold Analysis

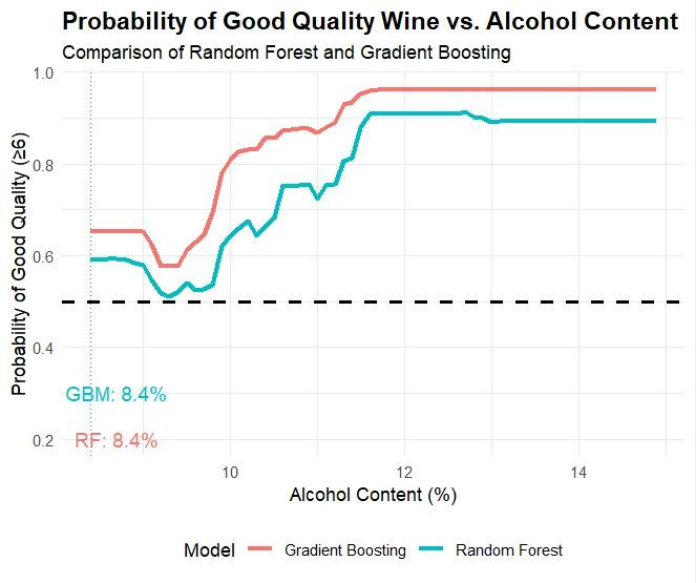
8.4% is the minimum alcohol content for both models

Gradient Boosting (Red Line):

- Starts at ~**65%** probability at **8.4%**
- Rises smoothly to ~**97%** at **14%+**
- Shows **continuous positive relationship**

Random Forest (Teal Line):

- Starts at ~**60%** probability at 8.4%
- More **step-like pattern** (characteristic of tree-based models)
- Plateaus at ~**92%** around **11-12%**
- Shows some **non-monotonic behavior** (slight dips)



- **8.4% alcohol** = 60-65% good quality probability
- **10-11% alcohol** = 85-90% good quality probability (optimal target)
- **12%+ alcohol** = 90-95% good quality probability (marginal gains)

Main Findings

1. Can quality be predicted using only 3 chemical measurements?

   ✓ **3-feature model**: Yes—alcohol, volatile acidity, and sulphates together account for the model's predictive power. This represents a highly efficient feature set.

2. Is alcohol content alone sufficient, or are acidity measures necessary?

   X **Alcohol alone is insufficient**: While alcohol is the dominant predictor, volatile acidity adds substantial predictive value. Using alcohol alone would sacrifice significant accuracy—the combined approach is necessary for optimal performance.

3. What minimum alcohol percentage gives >50% probability of good quality?

   ✓ **Threshold discovered**: "Threshold analysis reveals **8.4%** as the minimum alcohol level, but more importantly, it shows that quality depends on holistic chemical interactions. Winemakers should target **10-12%** alcohol while maintaining optimal chemical balance across all three key factors."

Conclusion

Key Takeaway: Wine quality prediction requires combination of alcohol content, volatile acidity, and sulphates - no single feature will suffice when predicting wine quality

Practical Value: Winemakers can target 10-12% alcohol threshold to improve quality ratings

Model Recommendation: Random Forest (**81.19% accuracy**) is optimal for quality control applications requiring maximum accuracy. Gradient Boosting (**77.74% accuracy**) is preferred for predictive applications for new, unseen wine batches

Impact: Provides actionable guidance for wine production optimization

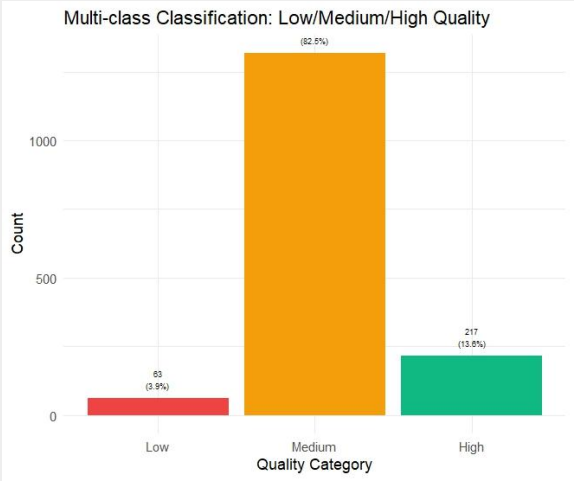
Questions??

Why Use Hyper-Parameter Tuning?

Class imbalance: Dataset has more medium-quality wines than excellent/poor quality—tuning allows me to adjust the parameters like min_samples_leaf to prevent the model from creating tiny leaf nodes for rare classes, producing a balance of good and bad, not just medium quality

Reduces overfitting: Binary pruned tree didn't capture all the data; tuning the CP, max_depth, minsplit, allowed me to find optimal balance

Interpretability vs. Performance:
allows the tree to generalize better to new, unseen data rather than just memorizing the training set



Random Forest in Predicting Wine Quality

1. Bootstrap Aggregating (Bagging)

What Happens: Each of the **300 trees** in my Random Forest doesn't see all **1,280** training wines. Instead, each tree is trained on a random sample of ~**1,280** wines drawn *with replacement* from the training set. This means:

- Some wines appear multiple times in a tree's training data
- Some wines don't appear at all (~**37%** are left out)
- Every tree sees a slightly different version of the wine quality story

2. Random Feature Selection (mtry)

What Happens: At each decision point in every tree, instead of considering all **11** chemical properties to make a split, the algorithm randomly selects only 4 features (**mtry**=**4**). So when a tree asks 'how should I divide these wines?', it might only consider:

- Alcohol, pH, density, chlorides (in one split)
- Volatile acidity, sulphates, citric acid, residual sugar (in another split)

This constraint forces trees to be different from each other

3. Majority Voting (Aggregation)

What Happens: "When predicting whether a new wine is 'Good' or 'Bad' quality:

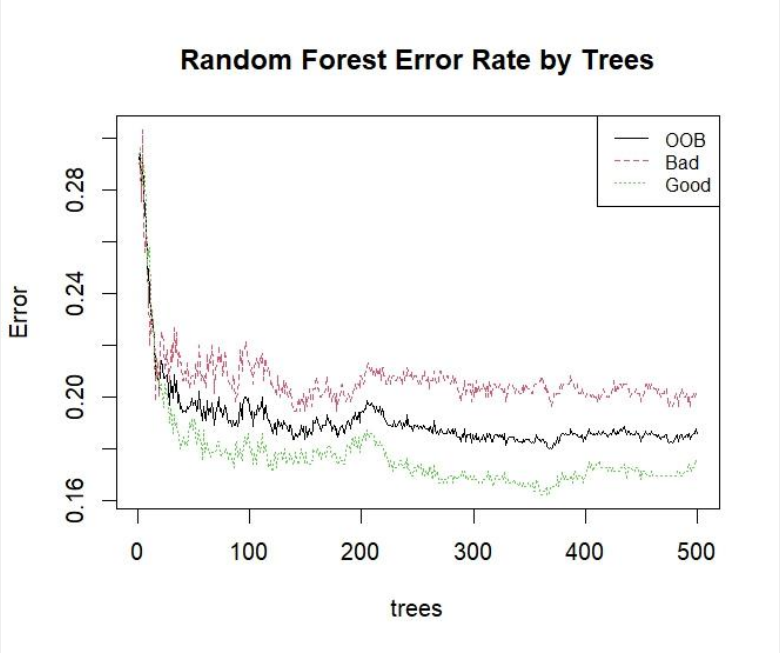
- Each of the 300 trees makes its own prediction independently
  - We count the votes: maybe **245** trees say 'Good', **55** say 'Bad'
  - The final prediction is 'Good' (majority wins)
  - The confidence/probability is **245**/**300** = **81.67%**

Random Forest Error Rate

- **mtry** = **4** (number of variables randomly sampled at each split)
- **ntree** = **300** (number of trees)
- **nodesize** = **1** (minimum node size)
- **OOB Error** = **17.81%**

The error rate plot shows:

- Initial high error (~**28%**) rapidly decreases in the first 50-100 trees
- Stabilization occurs around 150-200 trees
- Final OOB error stabilizes around **18%**
- Class-specific errors: Bad wines (~**20%**), Good wines (~**16-17%**)



Why pair Gradient Boosting with Random Forest?

Random Forest (Parallel Learning):

- Builds 300 independent trees simultaneously
- Each tree sees a random subset of features (mtry=4)
- Reduces variance through averaging
- **My finding**: Achieved **81.19%** test accuracy with strong feature separation

Gradient Boosting (Sequential Learning):

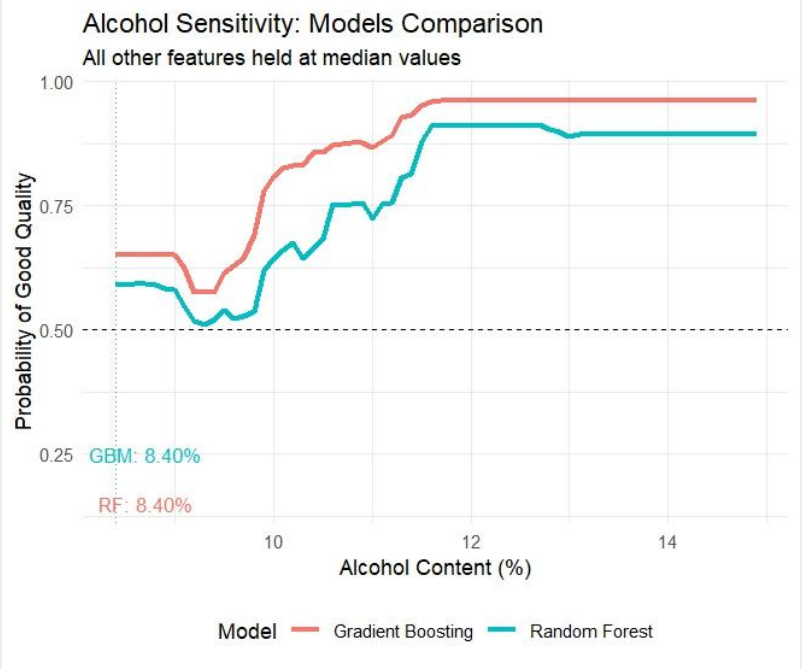
- Builds trees sequentially, each correcting previous errors
- Focuses on misclassified observations iteratively
- Reduces both bias AND variance
- Reduces overfitting

Alcohol Sensitivity Analysis

Median features baseline: 8.4% for both Random Forest and GBM

Wine makers should target **10-11%** alcohol as the "sweet spot" where probability of good quality increases most rapidly for typical wines

Alcohol content alone is NOT sufficient - acidity and other chemical measures are necessary



Alcohol Sensitivity By Wine Profile

- Wines with "good" chemical profiles (low volatile acidity, optimal sulphates) reach **50%** probability at just **8.4%** alcohol
- Wines with "bad" chemical profiles need nearly **10%** alcohol to reach the same probability
- Even at maximum alcohol (**15%**), bad-profile wines plateau around **72-75%** probability, while good-profile wines reach **95%+**