Modeling to Predict Wine Quality

Final Project

YoungRi Lee & Jihyun Lee

Last compiled on 2020-12-05

1 Introduction

1.1 Data

This project will use a wine quality dataset from Cortez et al. (2009) [1]. The dataset only includes *vinho verde*, a unique product from the Minho (northwest) region of Portugal. The data were collected from May/2004 to February/2007 that were tested at the official certification entity (CVRVV). The CVRVV is an inter-professional organization with the goal of improving the quality and marketing of vinho verde region. The data were recorded by a computerized system (iLab), which automatically manages the process of wine sample testing from producer requests to laboratory and sensory analysis.

The outcome variable is `wine quality`, which was measured by a minimum score of three sensory assessors using blind tastes in a scale that ranges from 0 (very bad) to 10 (excellent). There are 11 attributes of the wine based on physicochemical tests: fixed acidity (g(tartaric acid)/dm^3), volatile acidity (g(acetic acid)/dm^3), citric acid (g/dm^3), residual sugar (g/dm^3), chlorides (g(sodium chloride)/dm^3), free sulfur dioxide (mg/dm^3), total sulfur dioxide (mg/dm^3), density (g/dm^3), pH, sulphates (g(potassium sulphate)/dm^3), and alcohol (vol.%). Originally, two datasets were created separately, one for red wine ($n = 1599$) and another for white wine ($n = 4988$). In this report, we used a merged dataset and create a dummy variable to indicate the wine type, `red`. In total, the dataset includes 11 numerical attributes, one dummy variable (`red`), and one numerical outcome (`auality`). There is no missing value in this dataset.

Table 1 shows the descriptive statistics (minimum, maximum, and mean values) of 11 attributes and wine quality by wine type.

Figure 1 shows the distribution of wine quality by wine type. Generally, it shows a normal shape distribution and centered around the middle point of the scale.Red wine has fewer observations than white wine.

Table 1: Descriptive statistics of 11 attributes and outcome

	White			Red		
	Min	Max	Mean	Min	Max	Mean
Fixed acidity	3.8	14.2	6.855	4.6	15.9	8.320
Volatile acidity	0.08	1.10	0.278	0.12	1.58	0.528
Citric acid	0	1.66	0.334	0	1.00	0.271
Residual sugar	0.6	65.8	6.391	0.9	15.5	2.539
Chlorides	0.009	0.346	0.046	0.012	0.611	0.087
Free sulfur dioxide	2	289	35.308	1	72	15.875
Total sulfur dioxide	9	440	138.361	6	289	46.468
Density	0.987	1.039	0.994	0.990	1.004	0.997
pH	2.72	3.82	3.188	2.74	4.01	3.311
Sulphates	0.22	1.08	0.490	0.33	2.00	0.658
Alcohol	8.0	14.2	10.514	8.4	14.9	10.423
Wine Quality	3	9	5.878	3	8	5.636

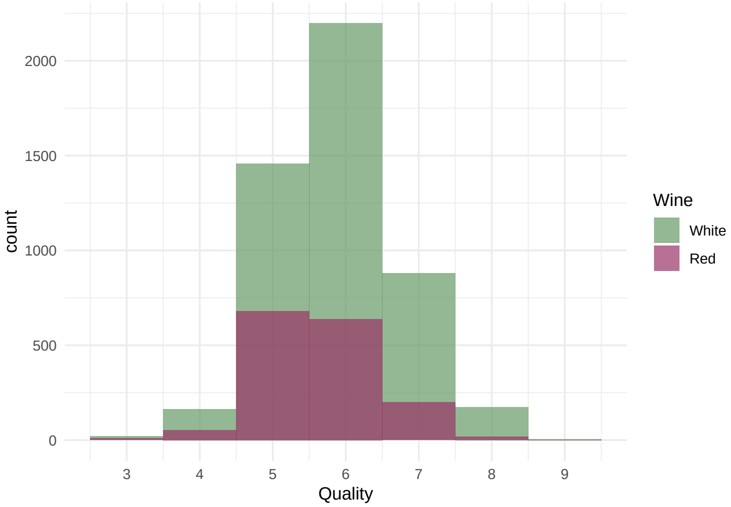


Figure 1: Histrogram of wine quality by wine type

1.2 Research Questions

We would like to (1) build the best model to predict wine quality, (2) examine the most influential set of attributes to predict wine quality.

To achieve this goal, we implemented two classes of techniques: shrinkage approache (i.e., Lasso) and tree-based approaches (i.e., regression tree). This report includes the comparisons of two approaches, model selection procedures (e.g., cross-validation), and proposal of the best model.

2 Methods

2.1 Shrinkage approach: Lasso

The shrinkage approach uses all possible covariates while shrinking the covariates' coefficients towards zero that does not associate with the outcome as strong as other predictors. The advantage of this method is to reduce the variance by reducing the relatively unimportant attributes and selecting variables. Thus, important covariates in the model can be emphasized. However, it may lead to the bias of the model by reducing the variance (*bias-variance trade-off*).

The two well-known shrinkage methods are *Ridge regression* and the *Lasso*. We choose Lasso over Ridge regression because Lasso forces the coefficients to be exactly equal to zero. On the other hand, ridge regression shrinks the coefficients *toward* zero, not makes exact zero. This is a drawback that ridge regression cannot select or subset the important variables and leads a challenge in interpretation of the model with large set of variables. In addition, it is known that lasso outperform ridge when variables are highly correlated (collinearity).

The *lasso* shrinks the coefficients toward zero and has a more stringent shrinkage penalty that can force the coefficients to be equal to zero. The lasso coefficients, $\hat{\beta}^L$ minimize:

$$RSS + \lambda \sum_{j=1}^{p} |\beta_j|.$$

where minimizing the residual sum of squares (RSS) of the model as well as the shirinkage penalty $\lambda \sum_{j=1}^{p} |\beta_j^2|$. λ is a *tuning parameter*, which controls the impact of shrinkage penalty. As the value of λ increases, the impact of the shrinkage penalty grows and the lasso coefficients will be close to zero, and exactly to zero. By pushing the coefficient to be zero, lasso performs variable selection identifying the variables that substantially impact the outcome. Thus, the model using lasso might be more parsimonious than that using ridge regression, making the model results more interpretable. The value of λ is generally determined by the cross-validation.

2.2 Tree-based approach: Regression tree

2.2.1 Single Tree

The tree-based approaches can be used for regression and classification. We will focus on regression trees because the outcome variable (*wine quality*) is a numeric in our data.

Tree-based approaches segment the predictor space into several simple regions. In a tree, each rule that split the segment will be summarized, called *decision-tree methods*. After splitting the predictor space into a number of regions (a leaf node; $R_1, R_2, ...R_J$) using the selected predictors X_j, the average of the outcome within each region ($\hat{y}_{R_j}$) will be calculated. The model fit of the regression tree aims to minimize the RSS:

$$\sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2$$

Thus, the regression trees split the predictor space and create the two new branches only if the new split decreases RSS. Since the regression trees use this top-down, greedy approach to select the model, the final model might not be the true optimal model. However, the regression trees have the advantage of taking into account the nonlinearity or interaction of covariates. Also, the regression tree is useful in interpreting the results.

2.2.2 Ensemble Methods

Instead of growing a single tree, ensemble methods is more powerful to build prediction models. We used random forests and boosting trees.

2.2.2.1 Random Forests Typically, random forests method improves over bagged trees by decorrelating the trees and reducing the variance. Each tree uses a sample of predictors, not all. We will use $\sqrt{12}$ as the number of predictors considered at each split.

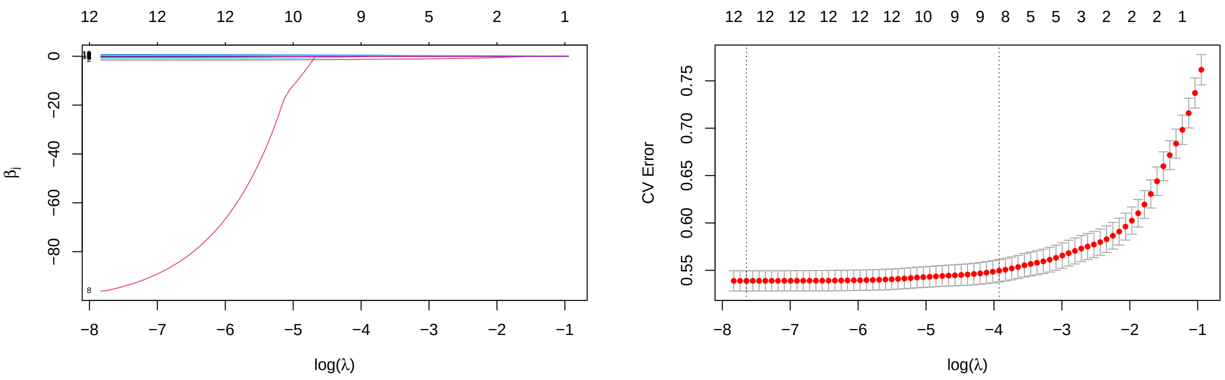
2.2.2.2 Boosting Boosting is a slow learning approach, potentially overfitting by updating residuals. The trees are grown sequentially and the final boosted model is the sum of the fitted trees. There are three parameters, shrinkage parameter (λ), number of iterations (B), and the size of each new tree (d, the number of splits).

3 Results

3.1 Lasso

Figure 2(a) shows the lasso coefficients across the $log(\lambda)$. We notice that only one variable, `density`, shows rapid change as $log(\lambda)$ increases. Other covariates are very close to zero.

We use 10-fold cross-validation to choose the tuning hyperparameter, λ that minimizes the expected out-of-sample prediction error. Figure 2(b) shows CV error as $log(\lambda)$ changes.



(a) Lasso coefficients

(b) CV error

Figure 2: Lasso results

Although the minimum CV error is a model with $\hat{\lambda} = 0$ ($MSE = .539$), we choose the 1 standard error criterion which suggests the parsimonious model. Thus, our choice of the optimal model is with $\hat{\lambda} = 0.02$. MSE with this $\hat{\lambda}$ is .550.

Table 2 shows lasso coefficients for the chosen optimal $\hat{\lambda}$, 0.02. The coefficients of `fixed.acidity`, `citric.acid`, `density`, and `red` (wind type) were equal to zero. Lasso selects a subset of variables: `volatile acidity`, `sugar`, `chlorides`, `sulfer dioxide`, `pH`, `sulphates`, and `alcohol`. `Volatile acidity` shows the largest magnitude of coefficients, −1.236.

3.2 Regression Tree

3.2.1 Single Tree

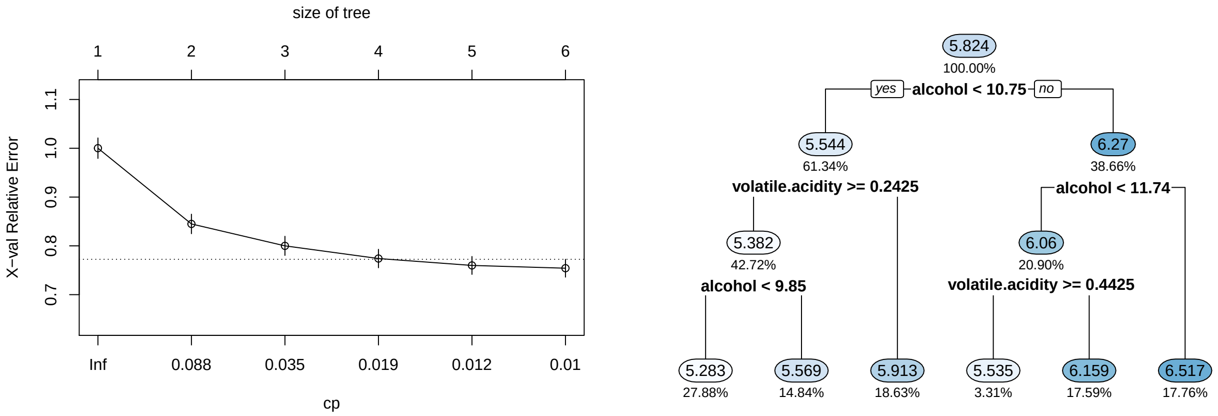
We use pruning to grow the best single tree and choose tuning parameter α using k-fold cross-validation ($k = 10$).

Figure 3(b) shows `alcohol` and `volatile.acidity` play key roles to determine the splits.

Size of optimal tree is 6. The test set MSE associated with this regression tree is 0.57. Square

Table 2: Lasso coefficients for the optimal lambda

Variables	Coefficients
(Intercept)	2.549
fixed.acidity	0.000
volatile.acidity	-1.236
citric.acid	0.000
residual.sugar	0.013
chlorides	-0.157
free.sulfur.dioxide	0.003
total.sulfur.dioxide	-0.001
density	0.000
pH	0.023
sulphates	0.470
alcohol	0.316
redWhite	0.000



(a)

(b)

Figure 3: Single tree

root of this MSE is 0.755, indicating that this model leads to test predictions that are within around 0.755 of the true wine quality.

[NOTE] This graph doesn't look pretty :(I guess it's because the outcome has very small range of distribution and pretty uniform. But this get better as we use ensemble methods later.

3.2.2 Ensemble Methods

3.2.2.1 Random forest

[JL note] This graph looks much better than the previous one.

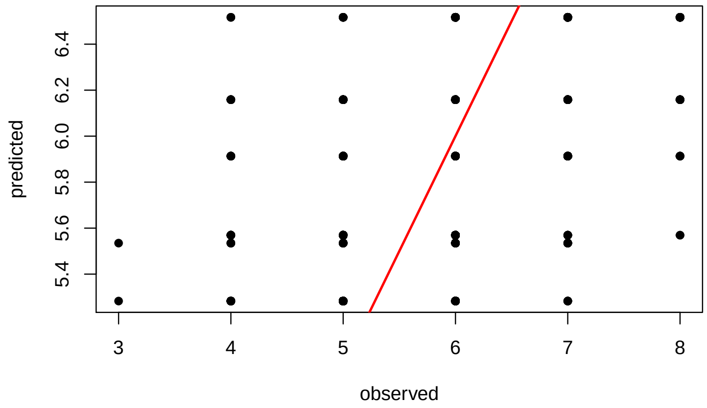
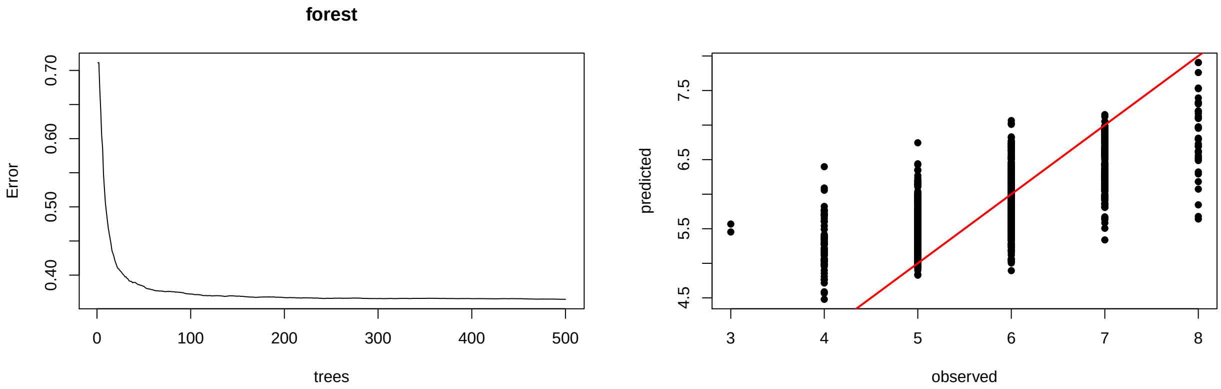


Figure 4: Single tree: prediction



(a) Out-of-bag Error

(b) Prediction

Figure 5: Random forest

`%IncMSE` indicates the mean decrease of accuracy in prediction on the out of bag samples when a given variable is excluded from the model. As the larger decrease, the target variable is important in the model. `IncNodePurity` measures the total decrease in training RSS. `Alcohol` seems to be the most important variables.

3.2.2.2 Boosting

The test MSE is 0.427.

We may choose an optimal lambda using cross-validation approach.

Fitting final model based on minimum RMSE.

`alcohol` and `volatile.acidity` are important variables in the boosting model. Similar choices as random forests.

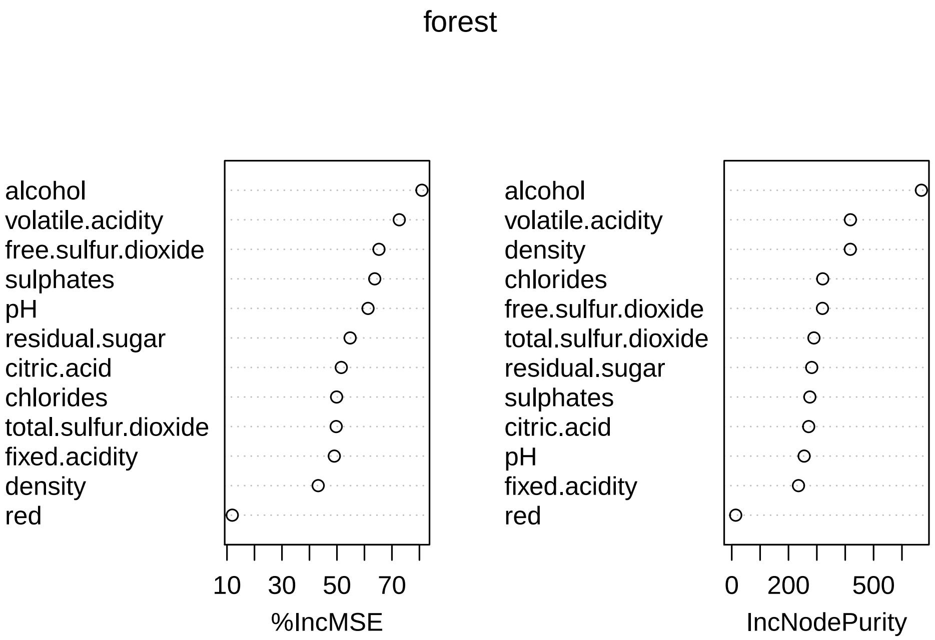
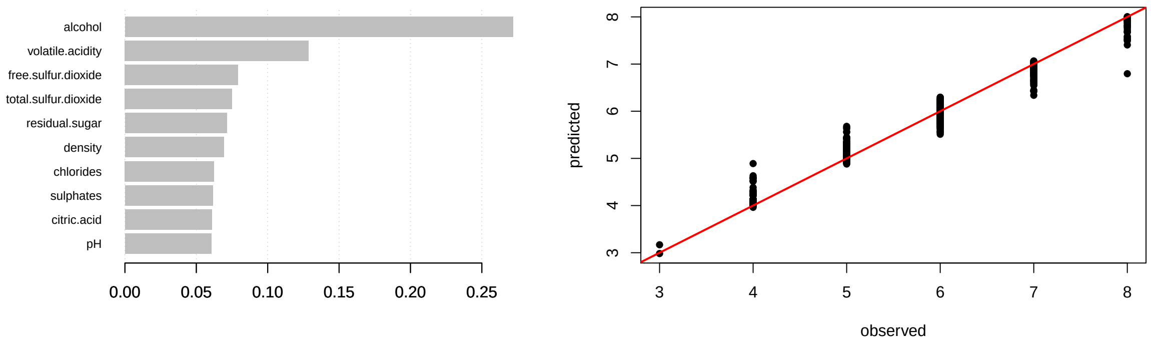


Figure 6: Random Forests: Variable Importance Plot

Table 3: Boosting results

	eta	max_depth	optimal_trees	min_RMSE
14	0.05	7	803	0.602
13	0.01	7	2696	0.606
15	0.15	7	217	0.612
10	0.05	5	1115	0.622
9	0.01	5	4747	0.624
11	0.15	5	259	0.630
16	0.30	7	145	0.631
12	0.30	5	197	0.643
6	0.05	3	1103	0.658
7	0.15	3	309	0.660
8	0.30	3	92	0.671
5	0.01	3	2049	0.671
2	0.05	1	1265	0.708
3	0.15	1	361	0.708
4	0.30	1	195	0.709
1	0.01	1	5000	0.709



(a) Variable Importance Plot

(b) Prediction

Figure 7: Boosting

The MSE of boosting model is 0.024.

[NOTE] Need to compare Random forest vs. Boosting

The boosting yields smaller MSE compared to random forests.

4 Discussion

[NOTE] Need to add general conclusion: comparing lasso vs. trees.

4.1 Implications

- Wine producers will use this information to produce the better quality of wine considering the selected attributes.
- Wine consumers will be able to choose a good quality of wine without tasting (e.g., via online shopping) if the physicochemical information of wine is available.

References

[1] P. Cortez et al. “Modeling wine preferences by data mining from physicochemical properties”. In: *Decision Support Systems* 47.4 (2009), pp. 547–553.