Predicting the Characteristics of Wine

Jacob Gottesfeld (jbg272)

Vaughn Campos (vac63)

Fisher Bricker (gfb53)

# 

# **Abstract**

Every different bottle of wine is defined by its unique characteristics. With these characteristics, we attempt to determine both the price of a bottle of wine and its approval rating on the online wine marketplace Vivino. Using linear regression models and tree-based methods, we find that the nonlinear tree-based models are significantly more effective at predicting the price of a bottle of wine and that rating is difficult to predict. We turn to classifying both the type of a bottle and the country it came from using Multinomial Logistic Regression, KNN, Boosting, and XGBoost. We show that XGBoost is the most powerful classification model for this dataset and is highly successful at classifying wines by their type. We conclude that the numerical attributes of a bottle of wine are more difficult to predict than its qualitative aspects.

# Analyzing the Importance of Wine Characteristics

The goal of this project was originally to solely model the effects of a wine bottle’s characteristics on its price and rating on the online marketplace Vivino. The dataset we analyzed was harvested from Vivino through its API, which provided information on thousands of various wines. The dataset had over 8000 observations and 17 variables including the name of the wine, its price, its average rating on Vivino, the number of ratings on Vivino, its body, its acidity, and its origin country among others.

After preprocessing and cleaning the dataset, we performed regressions on the data with various models and measured their predictive ability. We began with a linear regression on both Price and Rating of a bottle of wine of which neither were effective. We then attempted to model Price and Rating with tree-based methods. We ultimately found that a boosted tree was relatively successful in predicting Price over a data set transformed through Principal Component Analysis and that Rating was difficult to predict with any method.

Following our regression analysis, we decided to analyze a classification problem as well. We attempted to classify the type of the wines and the country they came from based on their characteristics. We compared the classification accuracy of a Multinomial Logistic Regression, KNN, and XGBoost and found that XGBoost was the best classification model for both type and origin country.

Comparing the results from our regression models and our classification models, we found that our classification models were more effective overall. This could be due to the fact that the qualitative characteristics we investigated were more permanent than the quantitative aspects we explored.

# Data Preprocessing

Before conducting any analysis on the dataset, we copied our dataset into two separate ones. One dataset would be used for Price regressions and predictions, named *dfPrice*, the other would be used for Rating regressions and predictions and was named *dfRating*. Categorical variables such as ‘Type’, if the wine is a red, white, etc., and ‘Country’, the wine’s origin country, were converted into dummy variables. This increased the number of variables in our data to 44.

## Removing Unnecessary Variables

The datasets had several unnecessary or problematic variables. ‘X’, the index of each wine in the dataset, ‘ID’, the identification number of the wine on Vivino’s website, and ‘Name’, the name of the bottle of wine, would have no impact on any models we would decide to use and were thus removed from both datasets. The other variables that were removed, ‘Winery’, ‘StyleName’, and ‘Region’, were thrown out for having too many unique categorical values.

## Removing Outliers

In order to remove any outliers from the datasets, we ran a simple linear regression of Price over all data in *dfPrice* and of Rating for all data in *dfRating*. These regressions were used to identify outliers with studentized residuals of magnitude greater than three and they were all removed. Other outliers that needed to be removed were records that had attributes that were extremely rare. These outliers were records with a Year earlier than 1950 and records with Country as Mexico.

## Addressing Missing Values

Two variables in the original dataset had missing values. Year had missing values labeled as “NONE” which were removed by filtering for “Year > 1950” when removing outliers. Acidity, a numerical value, had missing values that appeared as -1 and these records were all removed as well.

# Regression Analysis

After cleaning our data, we tested multiple regression models to discover which method was the most effective at predicting the price and rating of a bottle of wine. We began with a simple linear regression.

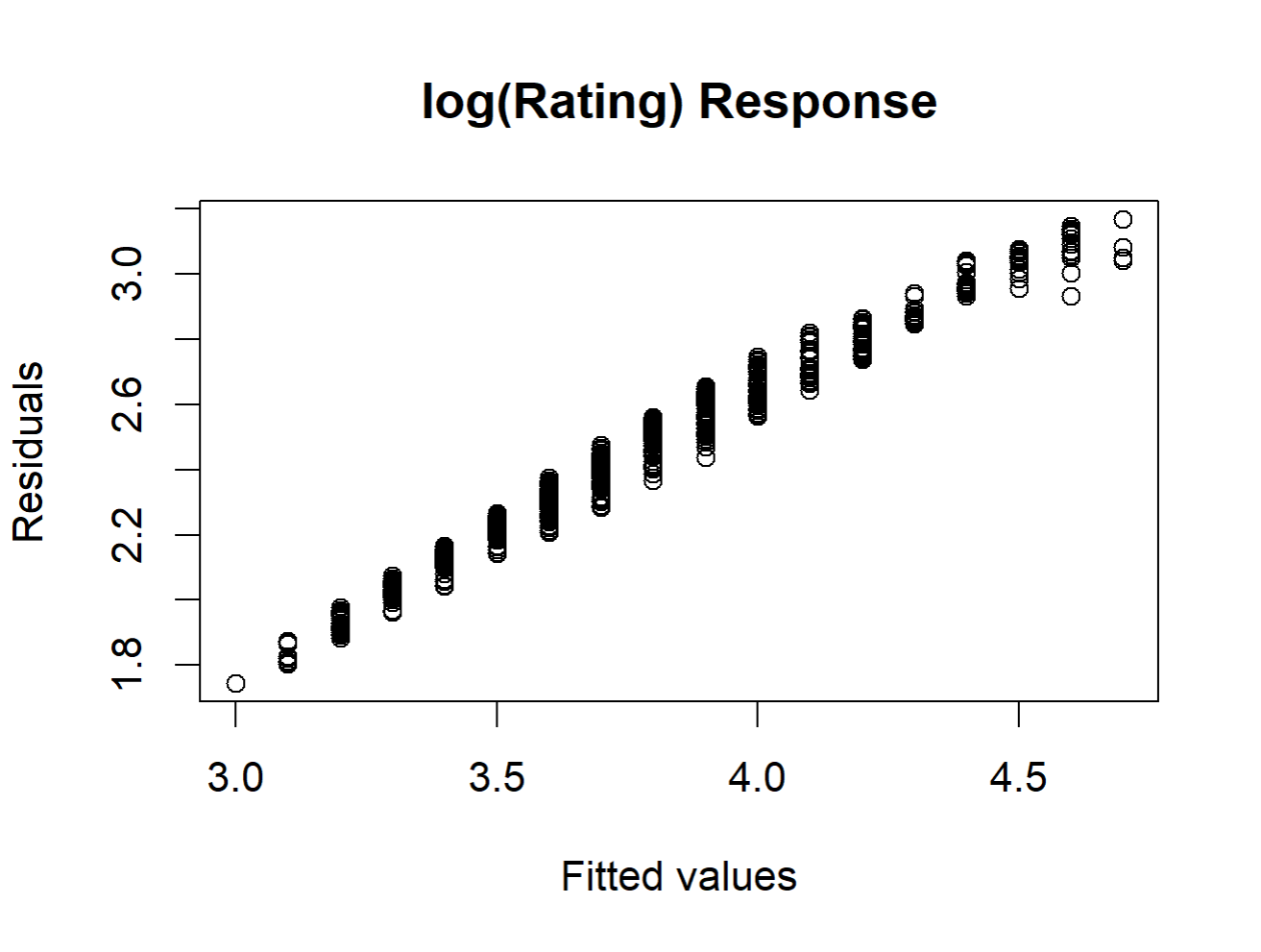
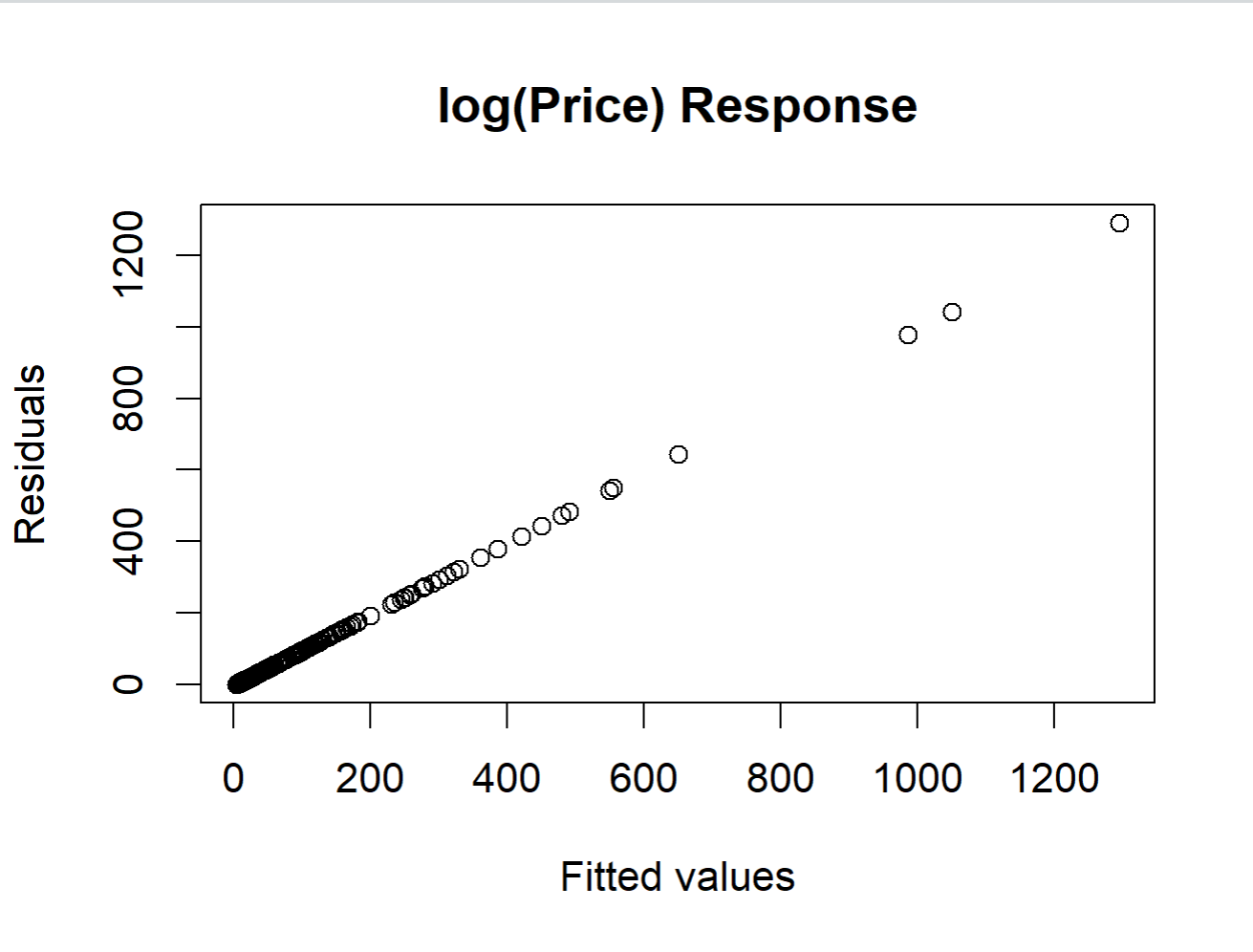
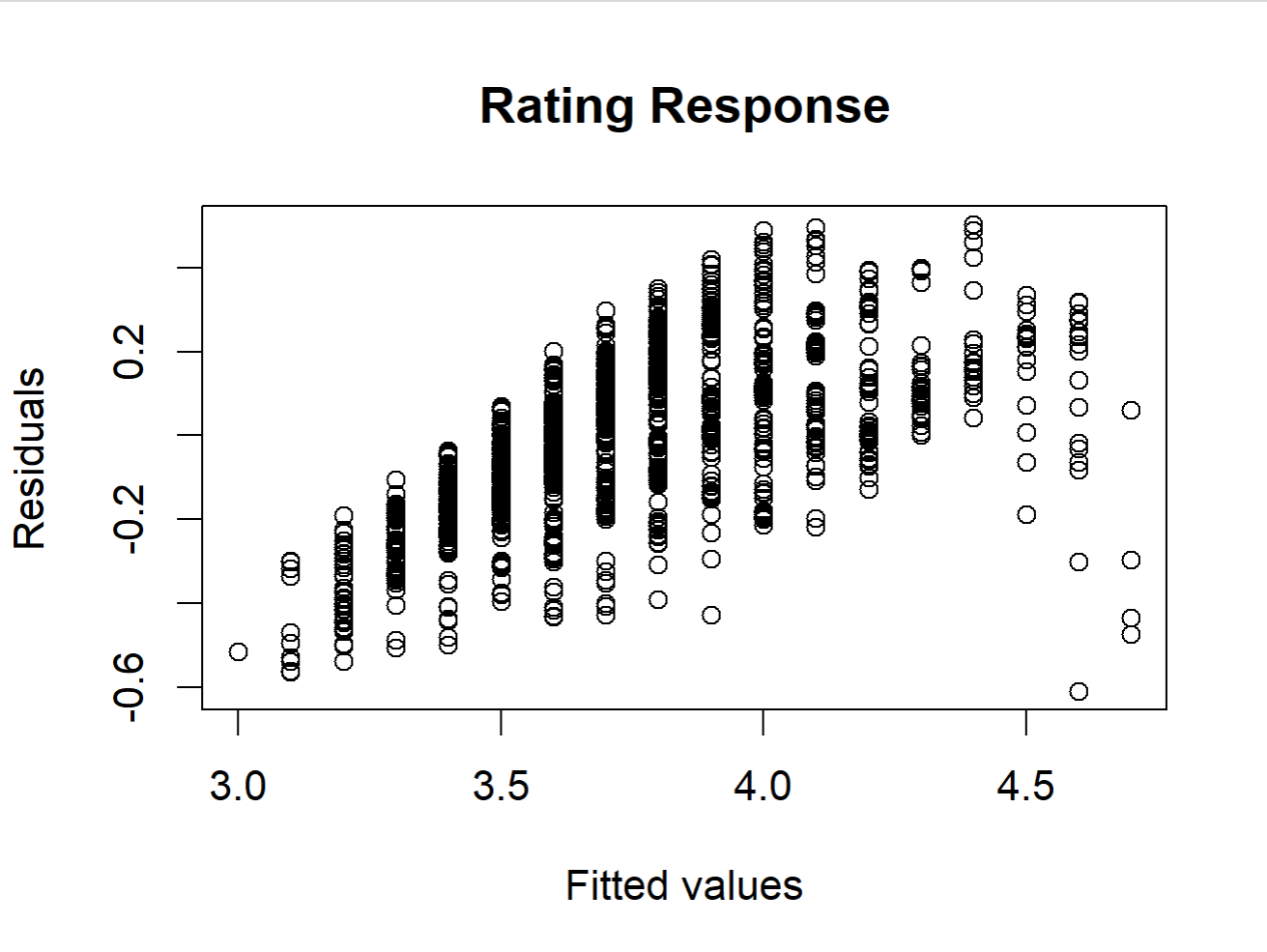
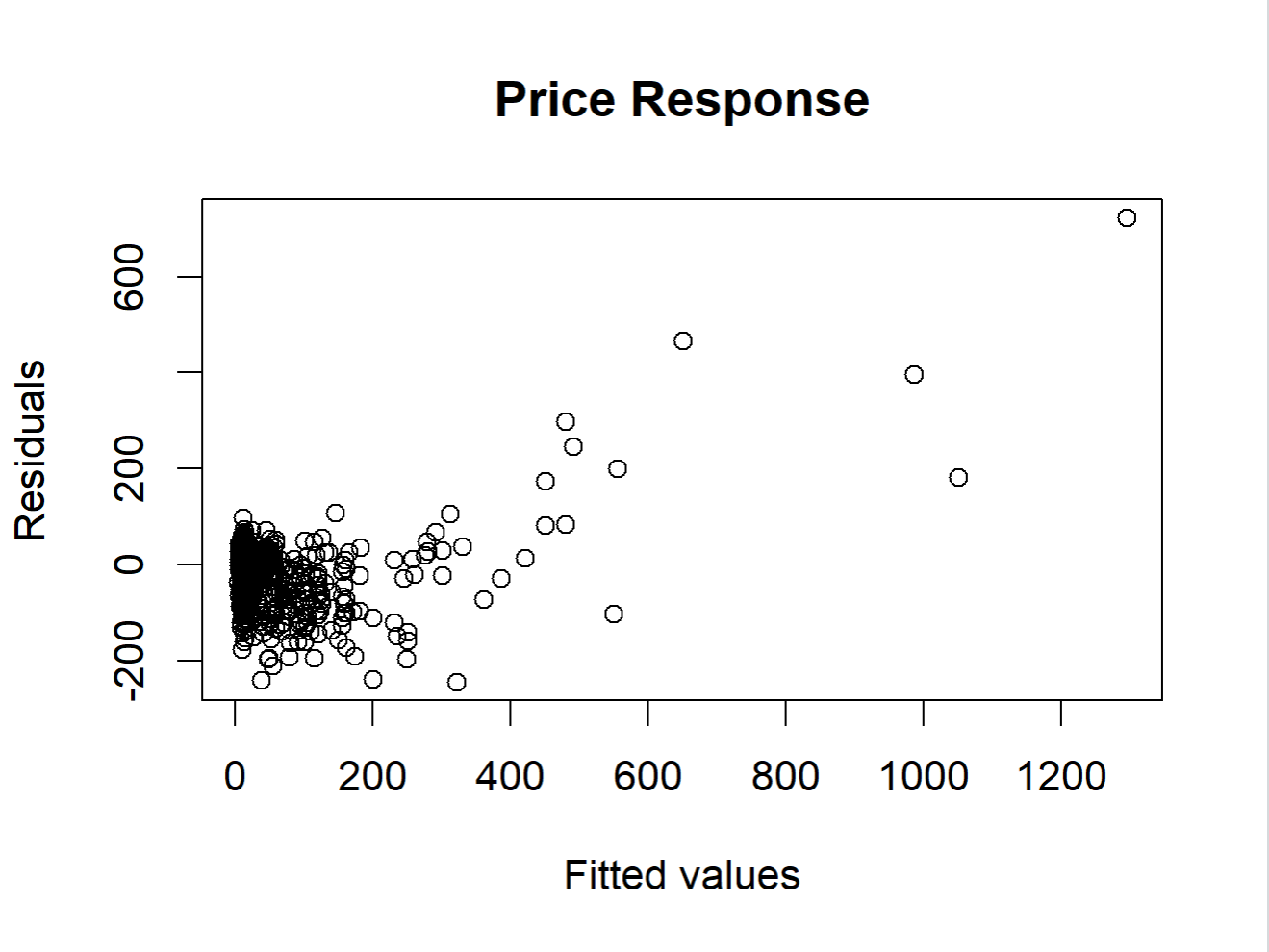
## Linear Model

We first developed a simple linear regression for Price in *dfPrice* and Rating in *dfRating*

using all the predictors in our data. These regressions were trained using a validation set of size 60%. Before measuring their effectiveness, we plotted the residuals of the regressions against the fitted values and found that there was non constant variance in the errors. We then transformed the response variables by taking the log(Price) and log(Rating), which corrected the variance of the errors as shown in Figure 1

**Figure 1**

*Residuals of Price and Rating Linear Regressions and Fitted Values before and after Response Transformation*



After transforming the response variables, we retrained the linear regressions and found the test error of the Price regression was 5086.632 while its R2 value was 0.8388. The test error of the Rating regression was 5.881201 and its R2 value was 0.5682 , much lower than that of the Price regression.

### Best Subset Selection

Due to the large number of predictors in our linear regressions, we used best subset selection

on the training set in order to prevent overfitting. Setting the maximum number of predictors to 30 for both regressions, we discovered that the price regression that maximized R2 had 24 predictors while the one that minimized Cp had 23 predictors. Both optimal values were almost the exact same for the model with 20 predictors so that model was chosen. For the Rating linear regression, the model with 20 predictors both maximized R2 and minimized Cp. The results of best subset selection for both regressions are shown in Figure 2. Compared to the original linear regression on Price, the best subset model expectedly performed worse due to having less predictors. However, surprisingly the best subset regression on rating performed similarly albeit using 17 less predictors.

**Figure 2**

*Best Subset Selection Model Results*

| Regression | R2 | Cp | Test MSE |
| --- | --- | --- | --- |
| Price | 0.8375804 | 14.67853 | 6984.347 |
| Rating | 0.5570241 | 12.10682 | 5.923141 |

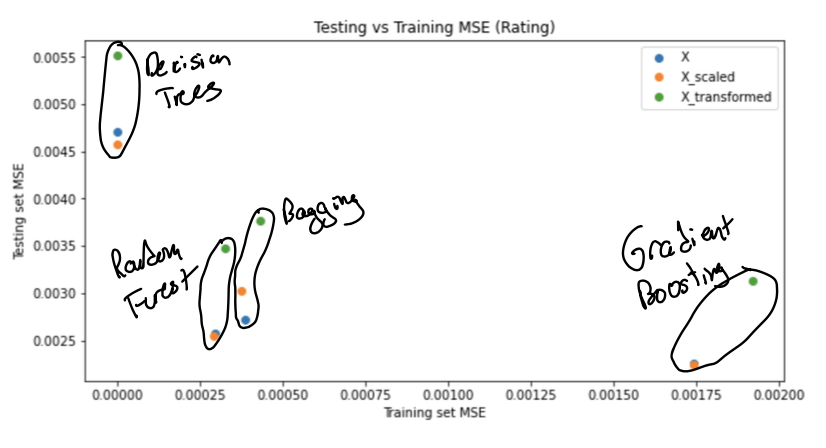
## 

## Tree-Based Methods

As our linear regressions were not effective, we wanted to try a non-linear model with normalized and transformed data. In order to decide on how to further process the data and what model would be best, we plotted the training versus testing error of each tree based method for predicting log(Price) and log(Rating):

**Figure 3**

*Testing vs Training MSE for Various Tree-Based Models over Original, Scaled, and PCA Transformed Data*

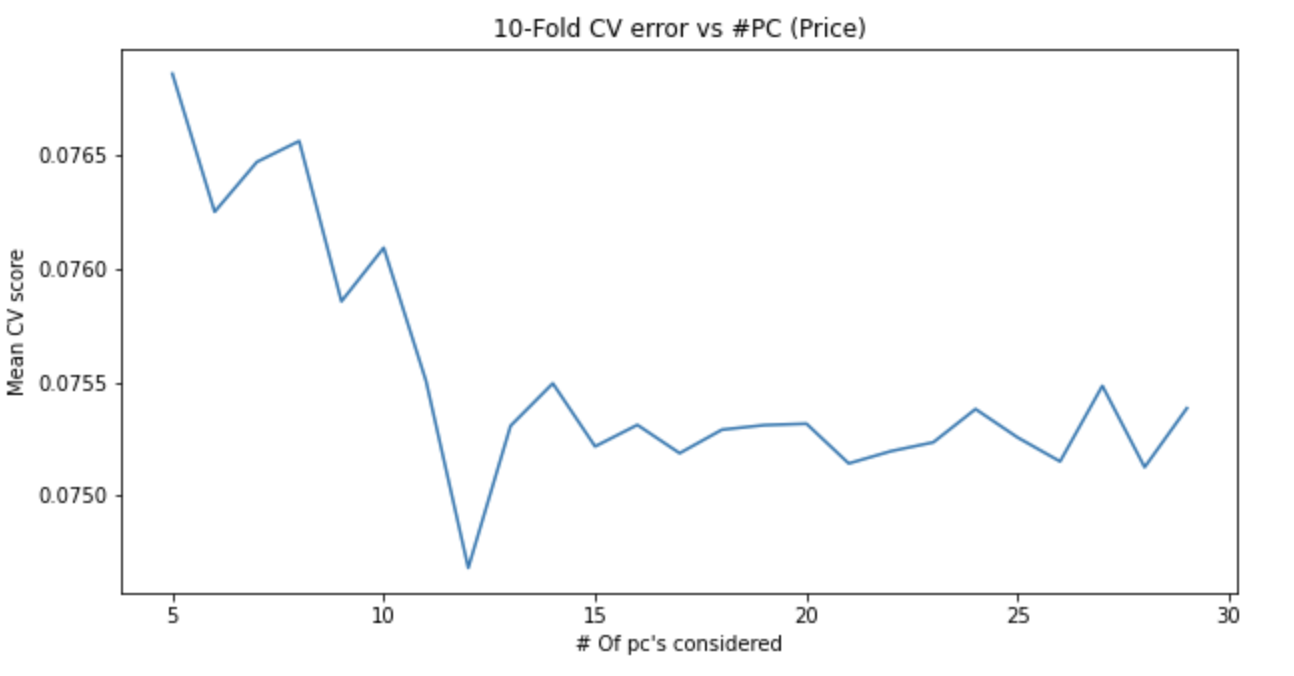
**

Decision trees evidently had no predictive power. For the Bagging, Random Forest, and Boosting models, the Principal Component Analysis transformed data improved the test error dramatically over the normalized and base training sets for Price prediction and worsened the test error for Rating prediction.

Looking further into the PCA transformations, we determined the optimal number of Principal Components to consider through 10-fold Cross Validation for price prediction.

**Figure 4**

*Boosting Cross Validation Error for Different Principal Component Dimensions*

**

As shown in Figure 4, the first 12 Principal Component transformations for a Price predicting boosted tree had the lowest Cross Validation error. We then trained a Boosted tree to predict log(Price) over those transformations and a Boosted tree to predict log(Rating) over scaled data. Both were tested on a validation set. The summary of these models can be seen in Figure 5.

**Figure 5**

*Summary of log(Price) and log(Rating) Predicting Boosted Trees*

|  | log(Price) Predicting Boosted Tree (Transformed Data) | log(Rating) Predicting Boosted Tree  (Scaled Data) |
| --- | --- | --- |
| Training MSE | 0.043434 | 0.001780 |
| Test MSE | 0.125674 | 0.002177 |
| Training R2 | 0.936818 | 0.695676 |
| Test R2 | 0.747450 | 0.487568 |

Although both models were overfit, as their test R2 values were significantly lower than their training R2 values, both models performed much better compared to the linear models. After translating the Mean Squared Errors into interpretable units, the Price predicting model had a test MSE of $1.428 while the Rating predicting model had a test MSE of around one point. In our dataset, the price of a wine bottle ranged from $5 to $1000 while the rating of a bottle was almost always between 3 or 4. The test MSE of the Price predicting model should therefore be interpreted as significantly more accurate than the test MSE of the Rating predicting model due to their units. Thus, nonlinear Price prediction proved to be successful with acceptable prediction variance which were improved by PCA transformation and Gradient Boosting. Conversely, non-linear Rating prediction performed very poorly with unacceptable variance for any model.

# Classification Analysis of Wine Type and Origin Country

In addition to predicting some of the quantitative aspects of a bottle of wine, we wanted to see if it would be possible to predict a wine’s type and origin country using the same data. In doing so, we would be able to compare our regression and classification results and learn if a wine bottle’s numerical or categorical attributes are more easily predictable.

## Origin Country Classification

We attempted to classify Country with various models including a Multinomial Logistic Regression, K-Nearest Neighbors (KNN) with K set at 10, Boosted Trees, and Extreme Gradient Boosting (XGBoost) with and without a sparse matrix input. There were 18 different countries that we tried to predict. The accuracy of each model can be seen in Figure 5.

**Figure 5**

*Country Classification Accuracy of Various Models*

| Algorithm | Accuracy |
| --- | --- |
| Multinomial Logistic Regression | 51.8% |
| KNN | 56.9% |
| Boosted Tree | 58.7% |
| XGBoost | 61.9% |
| XGBoost Sparse | 61.6% |

All models predicted the country within somewhat of a similar range of accuracies from 51.8% through the logistic regression to an accuracy of 61.9% through XGBoost. However, none of the models can predict with much confidence. The most frequently appearing countries in the dataset, the United States, France, and Italy, were predicted much more often than any other countries; many countries with a low number of observations were rarely, if not ever, predicted. This imbalanced data prevented our models from classifying countries accurately.

## Wine Type Classification

We then decided to predict the type of wine using the same method we used for country classification. There were four different classifications for the type of wine, numbered one to four, corresponding to red, white, sparkling, and rose wine. The accuracy of each of the five supervised classification models increased similarly to the country classifications models, except in this instance, XGBoost Sparse had the best test accuracy with a value of 86.2%.

**Figure 6**

*Wine Type Classification Table of XGBoost Sparse*

|  | | Test Type | | | |
| --- | --- | --- | --- | --- | --- |
| XGBoost Sparse Prediction | | 1 | 2 | 3 | 4 |
|  | 1 | 575 | 122 | 0 | 15 |
| 2 | 64 | 436 | 2 | 8 |
| 3 | 0 | 0 | 173 | 3 |
| 4 | 1 | 7 | 2 | 218 |

As shown in Figure 6, we can see that the model can predict the sparkling type with an accuracy of 97.7% and the rose type with an accuracy of 89%. The predictive strength for white and red was slightly lower but overall, the predictive power for each type of wine was much higher compared to predicting the wine’s origin country.

# Conclusion

After running all of our models we found that classifying the categorical characteristics of a wine bottle is easier than predicting its numerical attributes. While the Boosted Tree predicted log(Price) within reasonable error over the PCA transformed data, it was the only regression model that performed well. On the other hand, while some classification models did outperform others, all of them had test accuracies above 50% for classifying country and all were effective in predicting wine type. Thus, it appeared that the qualitative measures of a bottle of wine were more easily predictable than its quantitative characteristics. When considering the aspects of a wine bottle we investigated, it is reasonable that the classification models performed better overall. The price and rating of a bottle of wine are ever changing yet its origin country and type (red, white, etc.) are unchangeable. Therefore, there might have been stronger trends and patterns between the type and country of a bottle of wine and the other predictors in our data. If we were to further analyze this dataset, classifying other aspects of a wine such as its size or naturalness would be interesting as it would help us further understand the predictability of the categorical parts of a bottle of wine as opposed to its numerical features

# Bibliography

Vivino. (2022). *Wine Data* [Dataset Generated from API].

Retrieved from https://www.vivino.com/US/en.

ORIE 4740 Course Staff (March 29, 2022). Xgboost\_otto.Rmd (Unknown Version) [R Script].

https://canvas.cornell.edu/courses/39092/files/5715928?module\_item\_id=1456519