Evaluation of factors on wine ratings using regression model

Youn Kyeong Chang

Brown University

https://github.com/younkyeongchang/DATA1030_final_project

Introduction

This project aims to examine the effect of the origin country, the retail price, the varietal name, the winery name and vintage on the wine ratings using a regression model. Wine industry has remained one of the largest industries for over 7,000 years[1]. Recently, wine consumption has ballooned over several years and total wine gallons consumed in the US finally reached 1 billion gallon in 2020[2]. In this regard, it would flourish the industry to understand how much the geographic region and the retail price are related to the wine quality for future investment and research, which leads to consumers' purchase in the end.

The dataset originally consists of 129971 data points and 13 features including id, the country that the wine is from, flavor description, the vineyard within the winery where the grapes that made the wine are from designation, the number of points *WineEnthusiast* rated the wine on a scale of 1-100, US dollars price, the province or state that the wine is from, regions, taster name, taster twitter id, title, wine variety and winery[3]. Among these 13 features, id for index and the flavor description were excluded as of no interest. The vineyard, province and regions variables were also excluded for the reason that they are nested information under country and winery variables. 19 taster names and their twitter names were also not on the list of interest since a huge portion of data have no information of taster name and they are not even variables of interest at this project. To add a year (vintage) variable, vintage was extracted from the title variable. Lastly, the final 5 features, which are country, price, vintage(year), winery and wine variety were analyzed on the target variable, wine points. In terms of missing values, the target variable, `points`, and feature variables, `winery` and `variety`, have no missing values. In addition, country, price and year have negligible missing values, 0.05%, 6.92% and 4.22%, respectively. Therefore, observations containing missing values were excluded. As a result, about 90% of the original data, 115918 observations with 6 variables were analyzed.

There were a couple of projects conducted on the same data on Kaggle. One of the researches was to classify red and white wine using text analysis with the description variable[4]. It focused on how to define and extract the important words for wine description and categorize them. It revealed there are some descriptive words that make it easy to predict wine classes. The other project of this same writer was to predict wine price from vintage, wine class and region using Lasso regression[5]. The model he made was not successful at the extreme cases of price.

Exploratory Data Analysis

For readability, the whole data for the case of categorical variables was not presented. For the wine variety, 10 most popular wine varieties were chosen and investigated of the relationship with points. Compared to non-popular varieties, “others”, popular wine varieties had higher points in general.

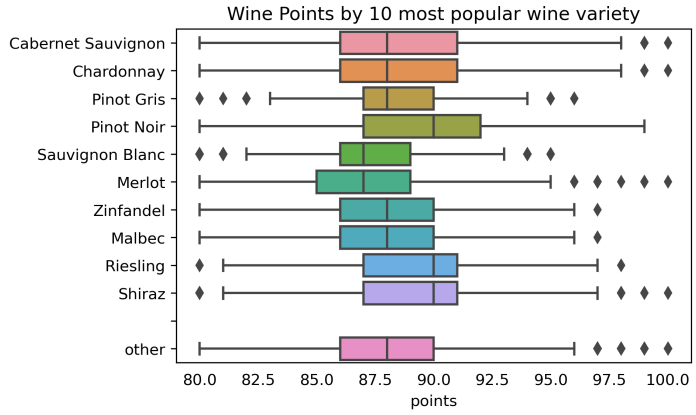

Figure 1. Wine points by top 10 varieties of wines.

Figure 1 shows that wine ratings are different by wine variety. From the plot, Pinot Noir, Riesling and Shiraz were rated highly on average. The plot suggests that there is an association between the variety and the wine points.

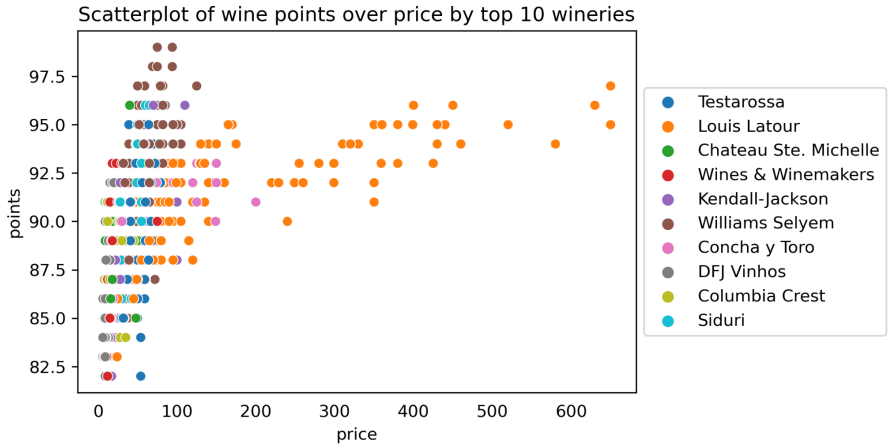

Figure 2. Scatterplot of wine points over price colored by 10 popular wineries.

Based on the observation frequency in this dataset, top frequently shown values for winery and country were selected figure by figure. Figure 2 shows there exists some association between price and points along with wineries. It represents that low price wines seem less likely to be highly rated. Also, some wineries make good quality wines with lower prices and some wineries make higher priced wines than other wineries.

Figure 3 displays the average wine points by vintage, the year in which the grapes were harvested, among top 5 winemaker countries. The year variable, extracted from the title variable, was recorded as NaN if missing or out of range of between 1900 and 2021. It appears that there are some great and not so great years for wine for each country. Additionally, one can see the quality of recently harvested wines from Portugal and France rapidly gets worse while the US wines steadily maintain their sound quality.

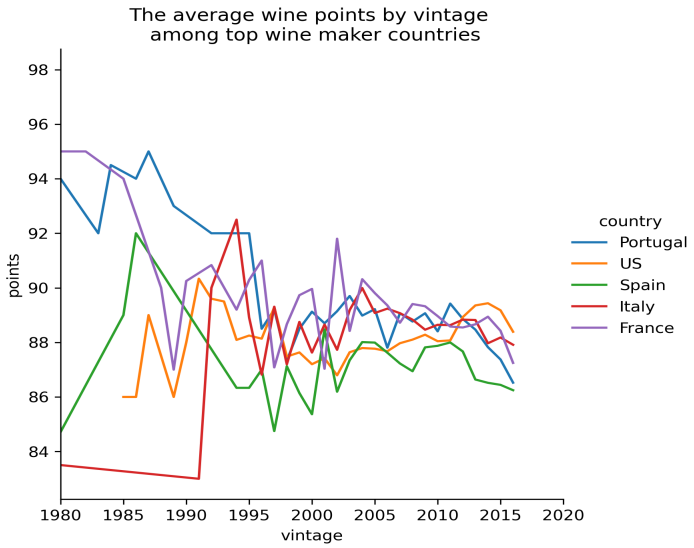

Figure 3. The average wine points by vintage among top 5 winemaker countries.

Methods

Data Splitting and Preprocessing

Each wine point was marked by 19 or mostly unknown `tasters` in the original dataset, suggesting non-independent and identically distributed (non-iid) data. However, not only is the subjective taster effect on wine rating not of interest in this project but also most of the `tasters` variables were not recorded, therefore, the data was not grouped by the taster variable. In addition, each wine `points` is unique and corresponding to one individual wine, the data was assumed to be iid and did not require time-series splitting. Categorical variables in this dataset have lots of various values, causing a memory error for data preprocessing steps. Accordingly, prior to splitting the dataset, `variety` variable was reassigned as "others" 10 out of 682 varieties except the top 10 well-known varieties. Also, `country` value was reassigned as "other" if the `country` was not in the top 10 out of 42 countries based on the observation frequency in this dataset. In a similar way, `winery` variable was reassigned as "other" except for the top 30 out of 15300. Manipulated dataset was splitted into train (80%) and test set (20%) and then Kfold with 4 folds was applied for the train set. Three different preprocessors were applied depending on the variable types. One-hot encoder was applied to `country`, `variety` and `winery` variables given that they are categorical and cannot be ordered. `year` variable was preprocessed with MinMaxScaler considering that it is reasonably bounded and does not follow the tailed distribution. For the `price` variable, StandardScaler was applied since they have a tailed distribution. As a result, the final preprocessed dataset has 55 features, including the continuous one target variable.

Model Training

Across the splitted and preprocessed datasets, four different machine learning algorithms were trained: a linear model with L_1 regularization (L_1), a linear model with an elastic net (ElasticNet), a random forest (RF) and a gradient-boosted tree (XGBoost). The parameters tuned and their values in each model are described in Table 1. In all models except XGBoost, GridSearchCV in the scikit-learn package was implemented for tuning the hyperparameters over a parameter grid. R2 score was used for an evaluation metric since it

gives a better picture of the goodness-of-fit, representing the standardized version of Mean Squared Error[6].

Models	Parameters
L_1	**alpha**: 1e-4, 1e-3, 1e-2, 1e-1, 1e0, 1e1
ElasticNet	**alpha**: 0.1, 1, 10, 100, 250, 500 **l1_ratio**: np.linspace(0.001, 0.99, 6) **max_iter**: 5000
RF	**max_depth**: 1, 2, 3, 5, 10 **max_features**: 3, 6, 'auto', 'sqrt'
XGBoost	**max_depth**: 1, 2, 3, 5, 10, 30, 100 **learning_rate**: 0.03 **n_estimators**: 10000 **colsample_bytree**: 0.9 **subsample**: 0.66

Table 1. Trained models with tuned parameters.

This process was repeated 10 times for 10 different random states to complement the uncertainties due to splitting and the randomized characteristics of some of the model training algorithms. On every random state for each model, the best parameter and corresponding R2 score from the test set were returned for the following evaluation of each model. Also, test sets were extracted along each random state for further analysis of feature importances.

Results

Model Evaluation

For the performance evaluation of each model, the average of the R2 score for each model was compared with the R2 score for the baseline model, which is zero. As the baseline model takes the mean of existing observations for the prediction value regardless of feature variable values, the residual sum of the squared error (nominator) and the total sum of the squared error (denominator) becomes the same, making the R2 score zero.

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2}$$

R2 score gets higher up to 1 as the model fits better. Therefore, within the range of 0 to 1, the average and the standard deviation of the R2 scores in the test set were compared and plotted. (Figure 4). Both RF (Mean: 0.44, Std: 0.00) and XGBoost (Mean: 0.44, Std: 0.01) seem the most predictive models in the similar level while linear regression models performed poorly.

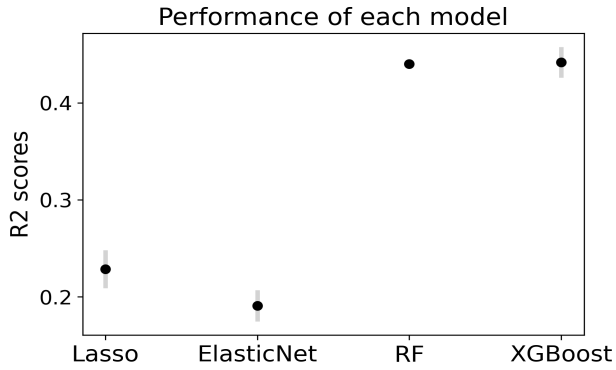

Figure 4. Average of R2 scores for each model over ten random states.

Interpretation of findings

Global feature importances were calculated using three different methods. RF model was chosen for the first two global feature importances since it turned out to have high R2 score and took less computational time than XGBoost. First, the permutation based importance for the RF model was implemented in the test set without preprocessing. (Figure 5.)

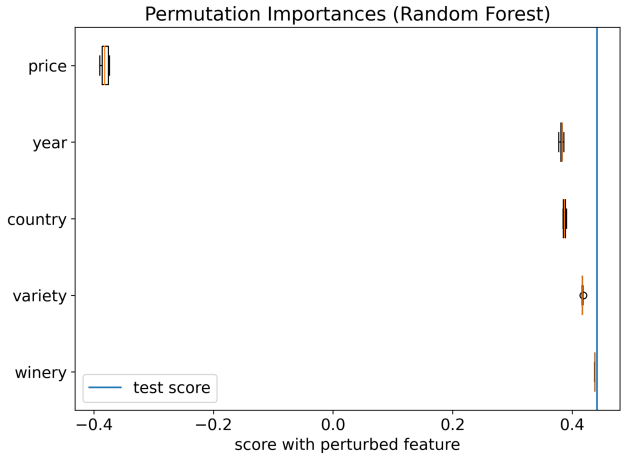

Figure 5. The average permutation test score for feature variables of each model.

Figure 5 shows that the `price` is the feature with the highest importance given that the prediction error increased the most after permuting `price` value. Contrarily, permutation of `variety` and `winery` variables barely have an effect on the prediction error, allowing us to conclude they are the least important features. Global feature importance was also obtained from the impurity-based built-in Random Forest algorithm. (Figure 6)

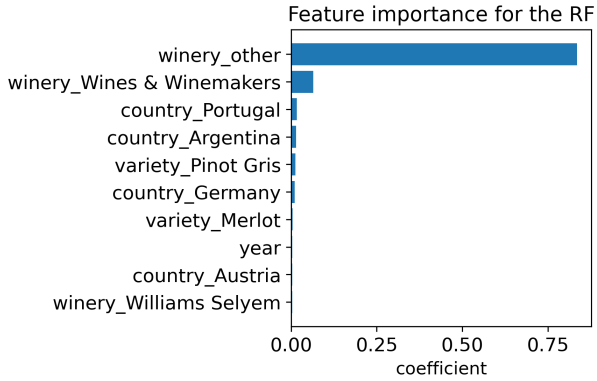

Figure 6. Global feature importance built-in the Random Forest algorithm.

While the permutation test score describing `winery` variable in general is the least important, impurity-based feature importance with preprocessed dataset shows that `winery_other` and `winery_Wines & Winemakers` are the most important global features. Lastly, coefficients of a linear model with L_1 regularization were also taken for global feature importance after standardizing all features to have a zero mean and the same standard deviation. (Figure 7) The list of feature importances have no common feature variables, thus it did not support the feature importance conclusion from the RF model. However, this result cannot have the upper hand since the R2 score of the linear regression with Lasso was much lower (Mean: 0.23, Std: 0.02) than that of RF.

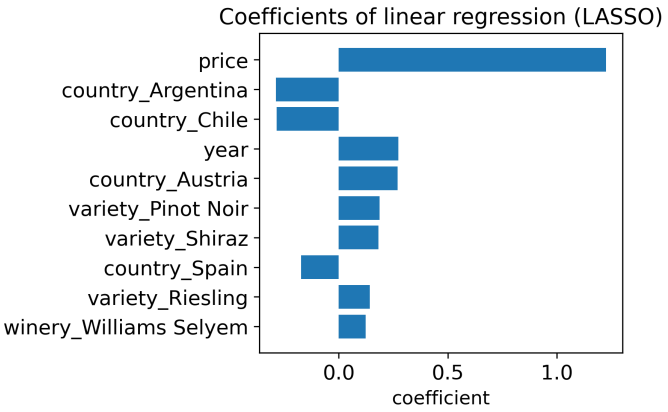

Figure 7. Coefficients of linear regression with L_1 regularization.

To figure out how much each feature contributed to the prediction, the SHapley Additive exPlanations(SHAP) method was applied[7]. (Figure 8)

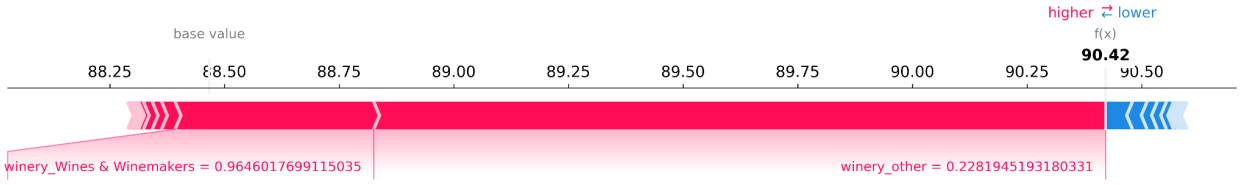

Figure 8. SHAP value with an additive force for one individual

For this specific individual as an example, the `winery_other` and `winery_Wines & Winemakers` mostly pushed up the predicted value from the baseline value (88.5). Furthermore, the feature importance was explained with feature effects in the following summary plot (Figure 9). In the plot, the features were ordered by their importance and corresponding SHAP values were represented horizontally. The global feature importance with SHAP values is fairly well matching with that from RF. As 'winery_Wines & Winemakers' seem the most important feature except the collection of miscellaneous winery variable, `winery_other`, SHAP dependence was explored to see the exact form of the relationship with wine

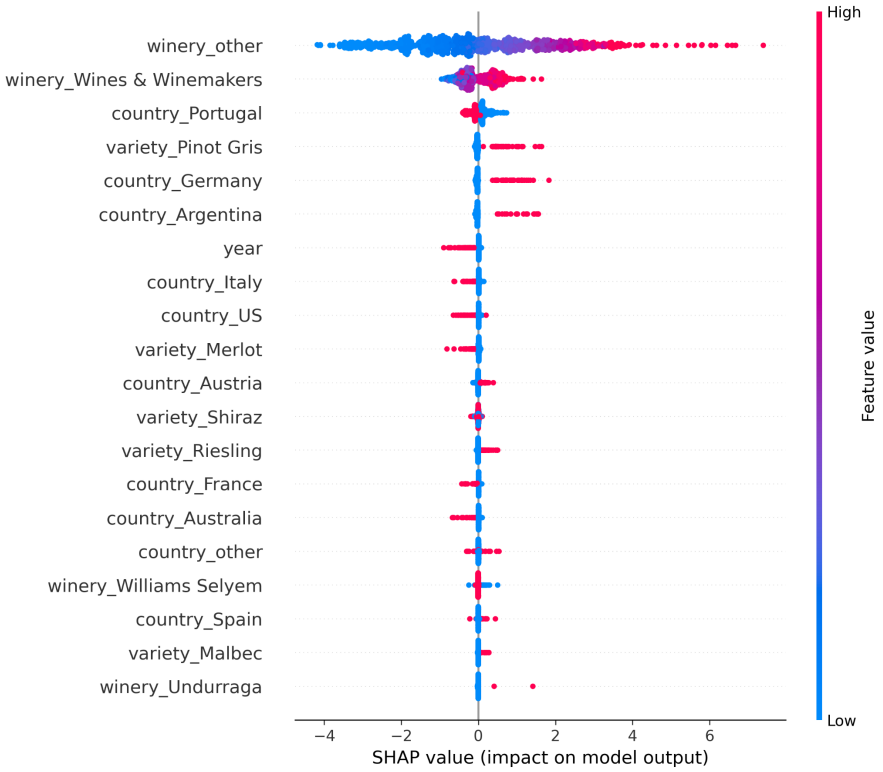

Figure 9. SHAP summary plot.

points (Figure 10). The trend of `winery_Wines & Winemakers` was confirmed with the subset of the test set as below.

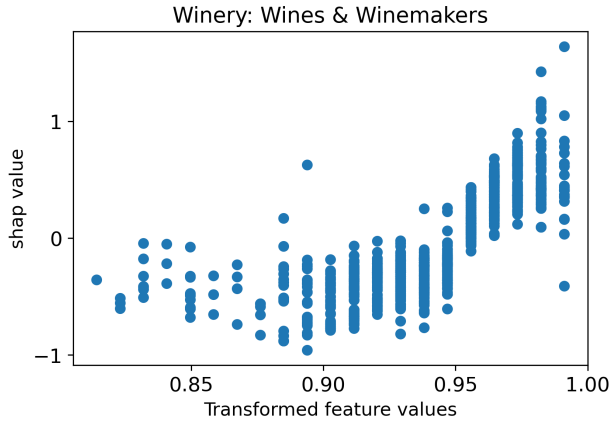

Figure 10. SHAP dependence plot for wine points on Winery: Wines & Winemakers

Outlook

Overall, trained four different models did not show the robust prediction successfully given that relatively low R2 score and inconsistent global feature importances. This probably has to do with the nature of the `points` variable. Most of the `points` are located around 90 within the range of 0 to 100 as shown in the exploratory data analysis. Therefore, collecting more data having a wide range of the target variable would improve the prediction power. On top of that, categorical values with too many unique values and the way of defining them would have weakened the models. Thus, how to deal with the categorical variables should be reconsidered for the improvement of the model. Lastly, the models above did not factor in the possible interactions of the variables. Implementing an additional step of the feature engineering for the interaction prior to model training would benefit the model.

References

[1] Gregory S. Carpenter and Ashlee Humphreys, 2019, *What the Wine Industry Understands About Connecting with Consumers*, Harvard Business Review, accessed 12 October 2021 https://hbr.org/2019/03/what-the-u-s-wine-industry-understands-about-connecting-with-customers

[2] Wine Institute Communications Department , 2020, *US wine consumption*, accessed 12 October 2021, https://wineinstitute.org/our-industry/statistics/us-wine-consumption/

[3] Nick Corona, 2020, Kaggle, accessed 30 September 2021,

https://www.kaggle.com/nicklcorona/wine-point/data

[4] Mike Friedman, 2017, Method Matters Blog, accessed 1 October 2021, https://methodmatters.github.io/analyzing-wine-data-in-python-part-3/

[5] Mike Friedman, 2017, Method Matters Blog, accessed 1 October 2021, https://methodmatters.github.io/analyzing-wine-data-in-python-part-1/

[6] Devore, Jay L. 2011, *Probability and Statistics for Engineering and the Sciences* (8th ed.). Boston, MA: Cengage Learning. pp. 508–510. ISBN 978-0-538-73352-6.

[7] Scott Lundberg, Gabriel Erion and Su-In Lee. 2018, *Consistent Individualized Feature Attribution for Tree Ensembles*.