ORIE 5741 Learning with Big Messy Data: Final Project

Predicting Wine Quality

Yu Jin (yj465) Heru Wang (hw743) Jinyuan Yu (jy478)

CONTENTS

1 Background

From the perspective of the wine industry, how can we predict human wine taste preferences based on the physicochemical properties of wine? This is an important question for wine certification and quality assessment.

With the COVID-19 pandemic, the last year has been quite a dozy for the wine market, as the closure of restaurants in many states fueled retail wine sales throughout the country. This crisis accelerated changes for wineries in how wine is produced and sold online. To be successful in facing challenges and meeting new demands, the industry needs new adaption in strategies and tools.

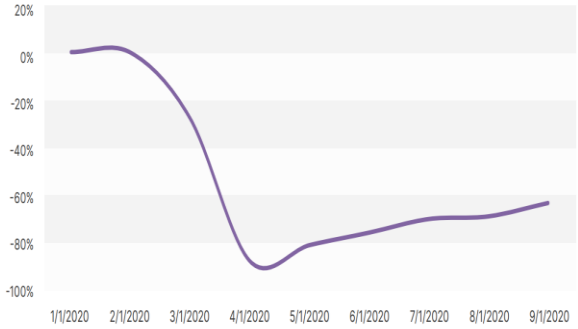


Figure 1 YOY On-premise Alcohol Sales in US

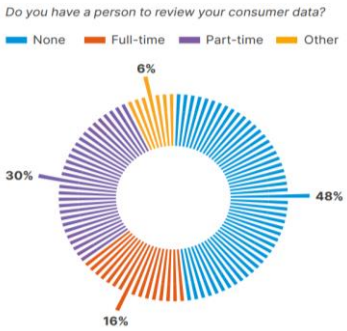


Figure 2 Data Analysts in Wineries

According to studies, more and more wineries are hiring data science experts to support their business decisions, and some wine companies (e.g., Ailytic, Tastry, and Vivino) have already implemented artificial intelligence to generate business insights for their operation. Thus, we believe the technologies we used in this project can help the industry in both wine making and selling processes.

In this project, our goal is to apply machine learning techniques to predict wine quality based on its physicochemical properties. We believe this project will provide additional values in the following fields:

- Improving the wine production process to enhance wine quality
- Supporting oenologist wine tasting evaluation
- Stratifying wines for premium brands and pricing
- Making better marketing decisions to attract potential customers

The following sections present the data and models we used and describes our empirical results. In addition, we provide an application of our models in optimizing the production of wine.

2 Dataset

The dataset used by the team describes the sensory quality of white variants of the Portugal 'Vino Verde' wine from May 2004 to Feb 2007, which was published by P. Cortez as part of his research in decision support systems.

- **Sample Size**: The dataset has 4898 entries in total.
- **Output**: The response is the quality of wine based on sensory data (median of at least 3 evaluations made by wine experts using blinding test). Each expert graded the wine quality between 0 (very bad) and 10 (very excellent) with an integer.

- **Input**: The features are 11 physicochemical variables that could be used to determine wine preferences. We have decided to include all the physicochemical variables in our preliminary analysis.

The overall quality of the dataset proves to be great, with no missing data on the entries. The meanings of physicochemical properties and their theorical effects on the output are described in the table below.

Table 1 Attributes Description and Theorical Effects

Attribute	Description
Fixed Acidity	Most acids involved with wine or fixed or nonvolatile (do not evaporate readily)
Volatile Acidity	The amount of acetic acid in wine, which at too high of levels can lead to an unpleasant, vinegar taste
Citric Acid	Found in small quantities, citric acid can add 'freshness' and flavor
Residual Sugar	The amount of sugar remaining after fermentation stops, it's rare to find wines with less than 1 gram/liter and wines with greater than 45 grams/liter are considered sweet
Chlorides	The amount of salt in the wine
Free Sulfur Dioxide	The free form of SO_2 exists in equilibrium between molecular SO_2 and bisulfite ion; it prevents microbial growth and the oxidation
Total Sulfur Dioxide	The amount of free and bound forms of S02; in low concentrations, SO2 is mostly undetectable in wine, but at free SO2 concentrations over 50 ppm, SO2 becomes evident in the nose and taste of wine
Density	The density of water is close to that of water depending on the percent alcohol and sugar content
pH	Describes how acidic or basic a wine is on a scale from 0 (acidic) to 14 (basic); most wines are between 3-4 on the pH scale
Sulphates	A wine additive which can contribute to sulfur dioxide gas (S02) levels, which acts as an antimicrobial and antioxidant
Alcohol	The percent alcohol content of the wine

3 Exploratory Data Analysis

The variables presented are solely real variables. To obtain a general idea on which physicochemical characteristics are relevant on white wine's sensory, we applied some data visualization methods to detect the distribution of values of different physicochemical variants respectively. Here, we present histograms of some typical variables: density, pH and alcohol, along with the output variable 'quality' which is an ordinal variable that ranges from 1 to 10. We can observe outliers for some features in the graphics, which is shown by the long tail of the distribution.

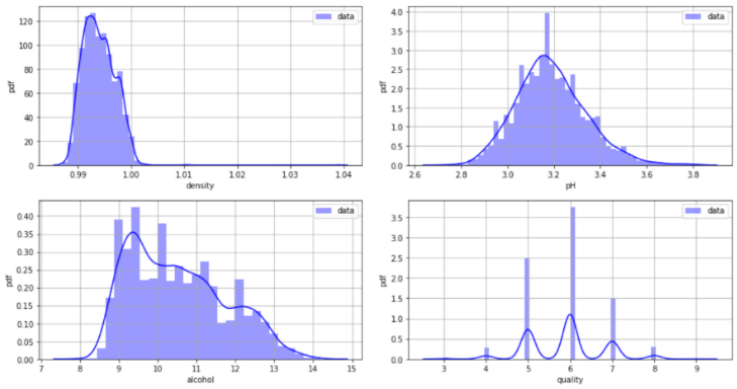


Figure 3 Data Distribution of Features

Next, we were interested in the general correlation of different features. The following heatmap describes the correlation matrix of the initial dataset. As is shown, although most of features are approximately independent, some of them are highly correlated. For example, the density of wine may be affected by the sugar and alcohol content. The high correlation of these features indicates potential multicollinearity and may reduce the accuracy of a linear model, which suggests some nonlinear models for prediction may outperform.

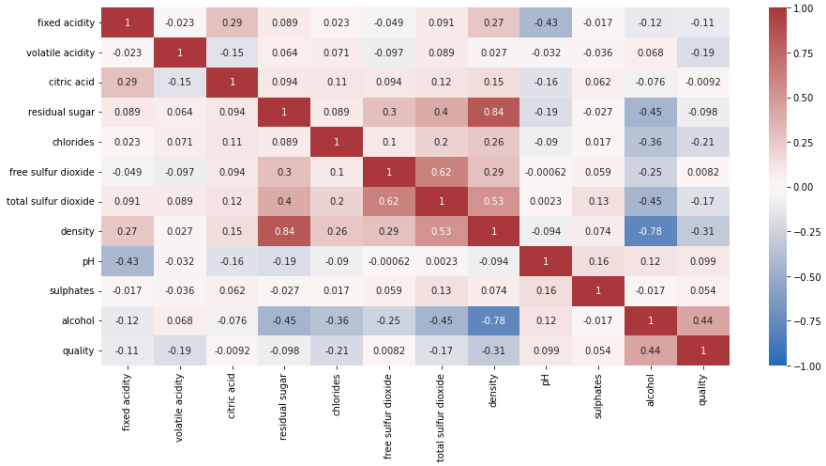


Figure 4 Correlation Matrix of Data

In addition, we were also wondering how these features affect the quality of wine. Instead of conducting quantitative analysis, we used some statistical graphics to visualize the relationship. Based on past endeavors of researchers in wine industry, we suggest that some of the variables contribute more to the sensory than others. For example, high-quality wine tends to have higher pH, which is shown in the graphic below. Also, we can find that the relationship of wine quality and alcohol content is likely to be nonlinear.

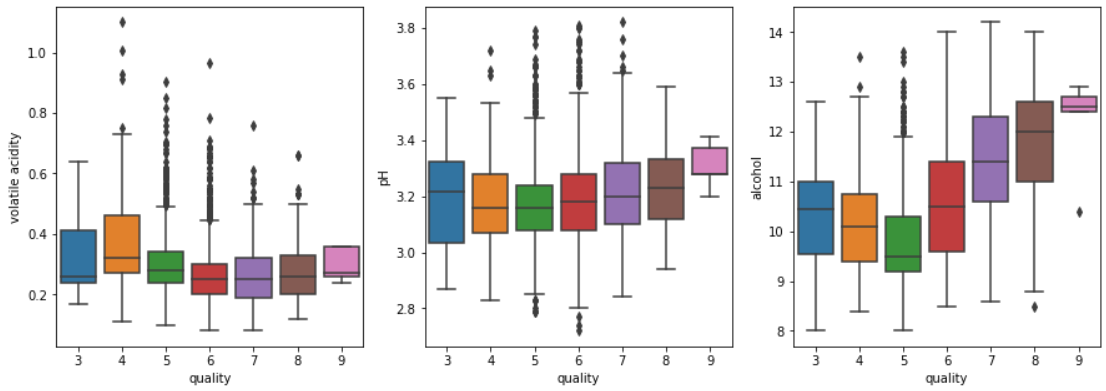


Figure 5 Box Plot of Quality Scores and Features

From the exploratory data analysis, we found some features were highly correlated and may have nonlinear effects on the output. The implication is that linear models may not perform well in this learning task.

4 Feature Engineering

In the following sections we will apply machine learning models to characterize the relationship of physicochemical properties and wine quality. Though the quality of raw data set is good, we still need to do cleaning and transformation to extract predictor variables for some specific models.

- **Missing Values and Outliers**: The dataset contains no missing values, so imputation is not needed for our task. According to the results in exploratory data analysis, we can see

some outliers for input variables. We decide to keep the outliers since they may provide some insights into the business. They can draw attention to a valid case in production that indicates an unusual flavor of wine. Besides, we want our predictive model to be robust when dealing with out-of-sample outliers.

- **Standardization**: Standardization can improve the performance of some machine learning algorithms that are sensitive to the scale of features. For penalized linear regression models (like Lasso and Ridge) and K Nearest Neighbors, we transformed the feature values to make them center around the mean with a unit standard deviation.
- **Adding Offset**: We estimated intercepts for linear models. Having an intercept gives our model the freedom to capture all the linear patterns while a model with no offset column can capture only those patterns that pass through origin.
- **Polynomial Transformation**: Since the relationship of features and wine quality may be nonlinear, we tried adding second degree polynomials for the linear least squares model to better fit the data.

5 Model Selection Methodology

Since the dependent variables are all ordinal integers in this problem, we adopt the regression approach instead of classification to preserve the order of the wine preference. Also, though wine experts gave integer scores for wine quality, it's okay to use continuous numerical values for business management. We have selected several kinds of regression models and trained each of them separately.

Table 2 Models for Learning

Family	Model	Description
Linear	Least Squares	Standard linear model (with least-squares estimators)
	Polynomial	Linear model with 2nd degree polynomials of features
	Lasso	Linear model with L^1-norm penalty function
	Ridge	Linear model with L^2-norm penalty function
	Huber	Linear model with Huber loss function
	LinearSVM	Support Vector Regressor with linear kernel
Trees	Decision Tree	Single decision tree with CART algorithm
	Bagging	Bootstrap samples to grow a group of trees for prediction
	Random Forest	Bagging model with features selected at random
	GBDT	Gradient Boosting Decision Tree
	AdaBoost	Adaptive Boosting Decision Tree
Others	K-Nearest Neighbors	Mean of the responses of nearest neighbors in the train set
	GaussianSVM	Support Vector Regressor with radial basis function kernel

Details of the methodology are as follow:

- **Train-test Separation:** Our objective is to fit the model using train-set (in-sample) data and test the model's performance with test-set (out-of-sample) data. The original dataset was divided randomly and 80% of it was used as the train set.
- **Error Metrics:** We used Mean Squared Error (MSE) and Mean Absolute Error (MAE) to evaluate our regression models. MSE has the benefit of penalizing large errors, while MAE can be better interpretated.
- **Hyperparameter Tuning:** We applied the 5-folds cross validation technique to tune the hyperparameters in the train set with grid search. For regularized linear regression models, hyperparameters are the regularization coefficients. For tree models, we needed to tune the maximum of depth for each tree and the number of trees (in ensemble models). For KNN, we chose the optimal "K" for our model.
- **Overfitting Avoidance:** We attempted to prevent overfitting in following ways: (a) Separated the train set and the test set for generalization. (b) Applied cross validation to tune the hyperparameters. (c) Introduced regularization techniques like penalty functions

in linear models. (d) Controlled the maximum depth for tree models.

- **Fairness and Limits**: Only physicochemical properties were available for our analysis due to privacy and logistic issues, so we don't necessarily need a fairness metric in this task. Our model can hardly be a Weapon of Math Destruction as long as the wine experts report their evaluation honestly, since all the data can be generated using experiments.

6 Empirical Results

Linear Models

We first introduce linear models for prediction. Though the assumption of linear relationship is not very realistic according to our exploratory data analysis, it still has the benefit of efficient estimation and straightforward interpretation.

The error metric values of different models in the test set are shown in the following figure.

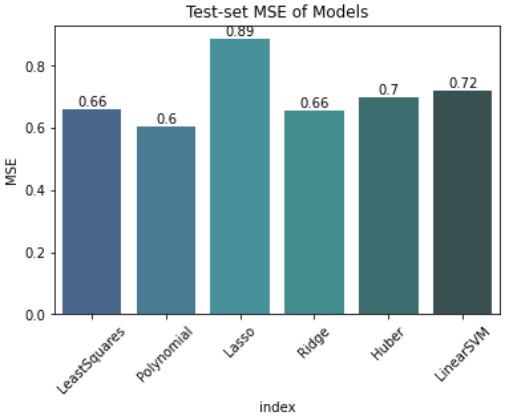


Figure 6 Test Set MSE of Linear Models

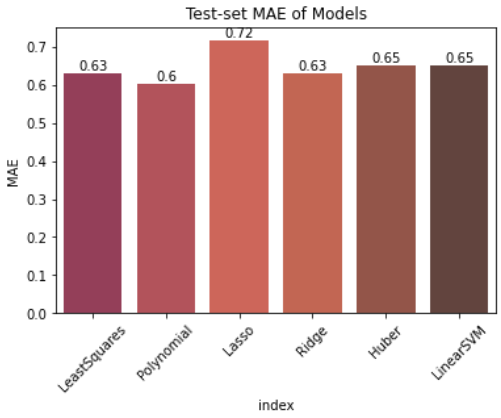


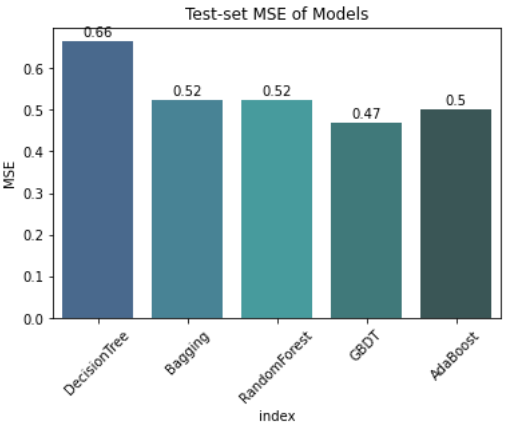
Figure 7 Test Set MAE of Linear Models

We can find that linear models generally underfit the test set data. Applying regularization techniques (like Lasso) doesn't improve the accuracy on the test set, while polynomial features can generate a better result.

From the estimated coefficients, we can infer that fixed acidity, residual sugar, free sulfur dioxide, pH, sulphates and alcohol have positive effects on wine quality, while others are negatively correlated with wine quality.

Tree-based Models

Next, we trained the decision tree model and ensemble models based on it. As is shown in the graphic below, most of these models outperformed linear models as they have lower error on the test set. Compared with linear models, tree-based models can better capture the nonlinear relationship of input so may better fit the data structure in our problem.



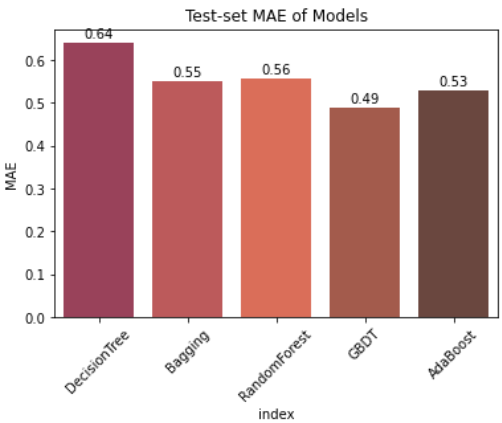


Figure 8 Test Set MSE of Tree-based Models Figure 9 Test Set MAE of Tree-based Models

Among all the tree models, gradient boosting performs the best. Boosting-based models generally have better results compared to Bagging-based models.

Another advantage of tree models is that we can calculate the feature importance quantitatively based on Gini impurity. We plotted the (normalized) total reduction of the criterion brought by each feature for GBDT:

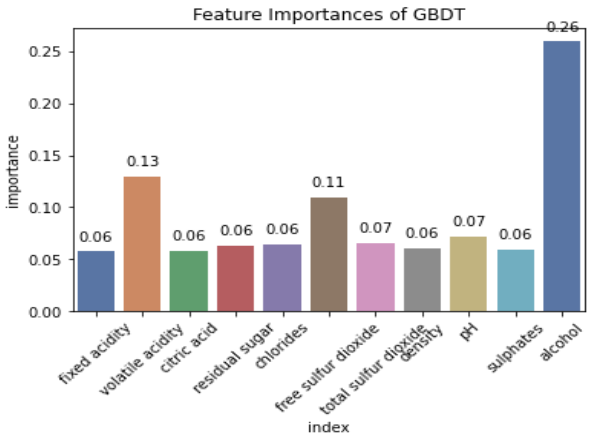


Figure 10 Feature Importance of GBDT

Based on the result of GBDT, we found that alcohol content is the most significant feature in predicting wine quality. Volatile acidity, free sulfur dioxide and pH are also important factors. This indicates that adjusting these inputs in production can help to improve the taste of wine.

Other Models

The test errors of other models like K Nearest Neighbors and Gaussian SVM are higher than those of ensemble models with decision trees. KNN has an MSE of 0.59 and an MAE of 0.58, while Gaussian SVM has an MSE of 0.65 and an MAE of 0.64. But they both require much time and computation resources to train, and are not easy to interpretate.

To summarize, among all the machine learning models we used, Gradient Boosting is the best one on the test set. Most of linear models tended to be underfitted while ensemble methods worked quite well. In addition, we found that features like alcohol and volatile acidity may have a stronger effect on wine quality.

7 Production Optimization

An application of our work is to improve the wine production process by controlling the physicochemical properties of wine. For example, residual sugar in wine could be raised by suspending the sugar fermentation carried out by yeasts. We used the machine learning models trained above as the surrogate model of real-world wine production process. Our goal is to find the optimal input (physicochemical properties) to maximize the quality score predicted by the model.

We chose the best linear model (the polynomial model) and the best tree-based model (the Gradient Boosting model) as the surrogate models to be optimized for wine production. The 10% quantile and the 90% quantile of variables in the dataset are set as the lower and upper bounds.

Polynomial Model Optimization

The polynomial model predicts wine quality with the second-degree polynomial of input variables. Hence, the optimization problem has a (nonconvex) quadratic objective function and linear constraints (lower and upper bounds). This is also called response surface methodology (RSM) in engineering.

We solved this problem with Sequential Convex Approximation (SCA) algorithm as described in `WineAnalysisFinal.ipynb`. The optimal input values are shown in Table 3.

Not surprisingly, most of the optimal input are on the boundaries, and only the optimal free and total sulfur dioxide are inner points. This is because quadratic function is a rough approximation of the true relationship, and the function can easily be monotonic in some dimensions. The optimal wine quality score is 10.18, which is higher than the maximum in our dataset. Since setting most input values at boundaries is not realistic, we don't recommend using this for improving production process.

Gradient Boosting Model Optimization

Gradient boosting gives a prediction model as an ensemble of weak decision trees. To maximize the output, traditional gradient-based and local-search methods can hardly work well here, because (i) the gradient of ensemble model is hard to be calculated; (ii) a small change of input may not change the value of output (since it may stay in the same leaf).

We applied Differential Evolution (DE) algorithm, a metaheuristic method, to optimize the Gradient Boosting model. This method does not require the objective to be differentiable or continuous. It maintains a population of candidate solutions and creates new solutions using existing ones, and then keeps whichever solution has the best fitness on the problem.

The optimal input values are shown in Table 3. From the result we can find that most of the optimal values are within the boundaries, and the optimal quality score becomes 9.09. These optimal input values are more reasonable and can be applied to improve the production of wine.

Table 3 Optimal Input Values in Wine Production Process

Attributes	Lower bounds	Upper bounds	Optimal Input in Polynomial Model	Optimal Input in Gradient Boosting
Fixed Acidity	5.90	7.90	7.90	7.06
Volatile Acidity	0.17	0.40	0.17	0.26
Citric Acid	0.22	0.49	0.49	0.48
Residual Sugar	1.20	14.00	13.99	9.16
Chlorides	0.03	0.06	0.03	0.03
Free Sulfur Dioxide	15.00	57.00	36.08	56.15
Total Sulfur Dioxide	87.00	195.00	116.14	127.80
Density	0.99	1.00	0.99	0.99
pH	3.00	3.38	3.38	3.38
Sulphates	0.36	0.64	0.64	0.48
Alcohol	9.00	12.40	9.00	12.37

8 Conclusion

Accurate wine quality prediction can create massive value to the wine industry. This work aims to predict the wine preference using data from objective analytical tests. Encouraging results were achieved, with the Gradient Boosting model providing the best performance compared to other models. Then, the well-trained model is used to construct the optimal control model for maximizing the wine quality in production process.

Our empirical results demonstrated that the proposed methods can effectively characterize the relationship of wine quality and its physicochemical properties and can be applied to solve real-world business problems. In future research, more advanced models and more data on different types of wine could be tested. The cost of controlling the physicochemical properties in production optimization could also be further studied.

Reference

[1] P. Cortez, A. Cerdeira, F. Almeida, T. Matos, J. Reis. Modeling wine preferences by data mining from physicochemical properties. *Decision Support System*, 47 (4) (2009), pp. 547-553

[2] J. Gu, W. Liu, K. Zhang, L. Zhai, Y. Zhang, F. Chen. Reservoir production optimization based on surrogate model and differential evolution algorithm. *Journal of Petroleum Science and Engineering*, 205(2021), 108879

[3] R. Storn, K. Price. Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11 (4) (1997), pp. 341-359