Modeling Wine Quality Using Classification and Regression

Mario Wijaya
Georgia Institute of Technology
mwijaya3@gatech.edu

1 INTRODUCTION

Quality of a wine is an important factor when one is shopping for a wine. Cortez et. al. [1] states that wine industry is investing a lot of money in quality assessment and wine certification to safeguard human health and improve wine making. Taste is a subjective thing, one might like the wine while others might hate it. Thus to classify whether a wine is good or bad is quite difficult. Wine shopper prefers good quality wine which leads to the question whether it is possible to predict quality of a wine which might help wine shopper to get a better quality wine.

2 PROBLEM DEFINITION

Given dataset (refer to section 4 for more details), these are the questions that I would like to answer:

(1) Can we classify whether a wine is good or bad based on a threshold (quality of a wine)?
(2) Can we create a regression model to predict the quality of a given wine?

3 WHY IS IT IMPORTANT?

This topic is particularly important to me because this is a validation that using data science technique, we can predict the quality of a wine much more accurately than a professional where his/her opinion might be subjective. If it is possible to create a robust regression model that can be use to predict quality of a wine, wine company can then use this information to understand what requirement is needed for a wine to be considered as good quality.

4 DATASET

The dataset that I will be using for this project is obtained from UCI Machine Learning Repository.[1] The dataset consists of information on red and white variants of the Portuguese "Vinho Verde" wine. The dataset has 11 features such as citric acid, pH, density, alcohol, etc. which are obtained from physicochemical tests and one output variable which is the quality of the wine obtained from sensory data.
I joined the dataset of white and red wine together in a CSV file format with two additional columns of data: color (0 denoting white wine, 1 denoting red wine), GoodBad (0 denoting wine that has quality score of < 5, 1 denoting wine that has quality $>= 5$). Note that, quality of a wine on this dataset ranged from 0 to 10.

5 SURVEY

Cortez et. al. [1] used Neural Network and SVM for their models. The paper stated that it used backward selection to choose their model and mean absolute deviation as the error metric to gauge the regression performance.

[1] https://archive.ics.uci.edu/ml/datasets/Wine+Quality

6 METHODOLOGY

The goal for this project is to answer the questions from section 2. The dataset has imbalance class of data, with white wine dataset has 3 times of red wine dataset. Hence, I used a method called SMOTE (Synthetic Minority Over-sampling Technique) by oversampling the red wine dataset to match that of white wine to prevent bias. Then we proceed with the following: First, pre-process the data to scale or normalize all of the features to prevent bias of the features used. Second, model selection method can be applied to get rid some of the features that has high correlation with other features. Third, I will apply classification method such as SVM to see how good the model is. Other classification algorithms such as Decision Tree and K-nearest neighbors are used to gauge against SVM model. Lastly, multiple linear regression is applied to predict the quality of a wine based on the input features.

Note that, k-fold cross validation is performed to get the desire model for testing data. The reason why I chose to use k-fold cross validation is to reduce overfitting of the model which makes the model more robust and generalize enough to be used with new data. The tool that I used is Python (scikit-learn) and R.

To simplify some of the model, I used Principal Component Analysis when running model such as Decision Tree Regression and classification algorithm such as SVM, KNN, and Decision Tree.

7 DATA EXPLORATION

Before diving into analysis, I am interested in how does one feature correlate with others, so I plotted the correlation matrix as shown in Figure 1. We can see that several predictors such as alcohol and citric_acid have high correlation to quality of a wine.

8 RESULTS & EXPERIMENTS

8.1 Regression

First, I naively did multiple linear regression including all features using the model of $y = \beta_0 + \beta_1 X_1 + ... + \beta_{11} X_{11}$ where $1, 2, ..., 11$ refers to all of the features: fixed_acidity, volatile_acidity,..., alcohol. As expected, the model that we have currently is not good as we have $R^2 = 0.325$.

Next, I used Stochastic Gradient Descent (SGD) to perform regression for a better result but it yielded similar result of $R^2 = 0.323$. Refer to final.py for more details. Also, I used Lasso and Ridge regression combination to penalize/regularize the parameter to get a better model but the result is not promising with $R^2 = 0.315$ (Refer to regression.r for more details).

Then, I tried model selection to get a subset of model that can predict quality of a wine but did not get a good model. Afterward, I ran Decision Tree Regression as shown in Figure 2, clearly it does

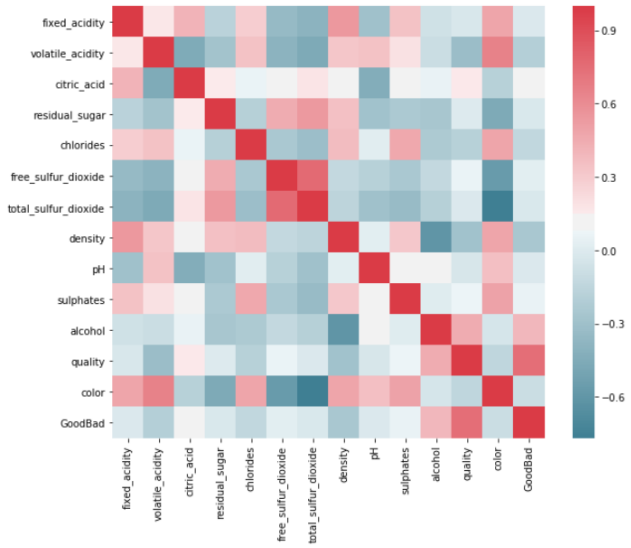

Figure 1: Correlation Matrix

not give a good prediction.

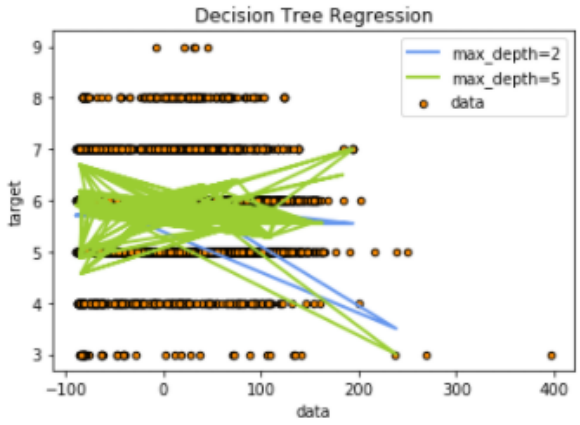

Figure 2: Decision Tree Regression

At last, I build a model by using all of the interaction term between all 11 features with white wine and red wine as factor/level (dummy variable). For this model, I performed removal of most non significant term one step at a time. After 23 iteration of this process, the model I obtained has $R^2 = 0.3785$ which is slightly better than the original model. Refer to regression2.r for more details.

8.2 Classification

For classification models, refer to final.py for the code and details.

8.2.1 ***SVM with No PCA.*** First, I build a SVM model with no PCA transformation and trained it using RBF kernel and parameters ($C = 0.1$ and gamma= 1). Note that the optimal parameters are obtained by using multiple C's and gammas and performed cross-validation of $k = 5$. With these parameters, SVM model is built and tested the performance using testing data. However, we get accuracy of 100%. Clearly, we can see that this model is an overfitting model.

8.2.2 ***SVM with PCA.*** By using PCA to do dimension reduction, we can plot the result and can see it visually whether the model is what we expected. For SVM model using PCA, I used 10-fold cross validation to find the best model that gives lowest error rate/highest accuracy rate. I ran it on both RBF kernel as shown in Figure 3 and linear kernel as shown in Figure 4 with varying (C, gamma) and C respectively. Note that the x-axis refers to iterations of different C and gamma while y-axis refers to prediction accuracy rate.

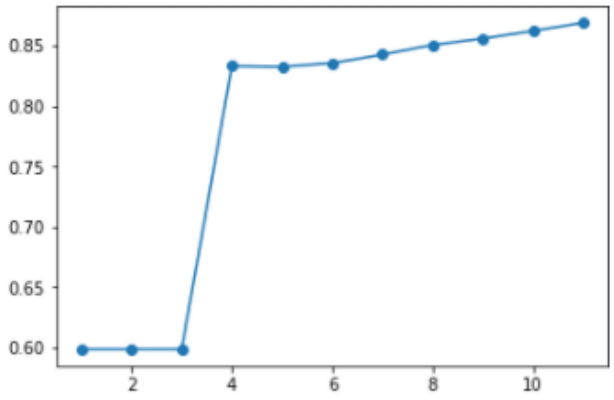

Figure 3: RBF Kernel Prediction Accuracy (Validation Set)

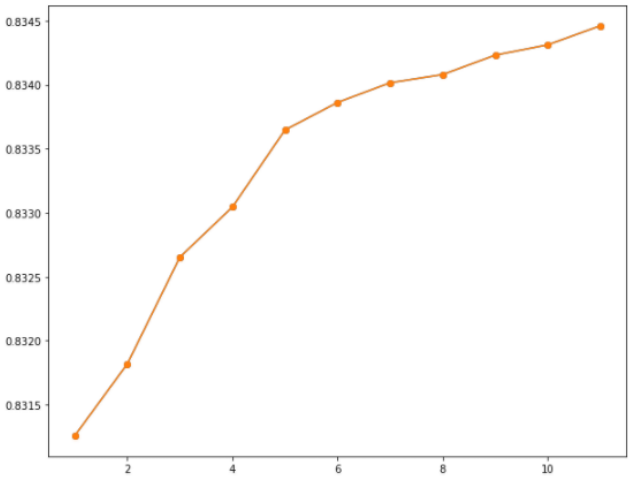

Figure 4: Linear Kernel Prediction Accuracy (Validation Set)

After obtaining optimal parameters through validation set, both models were used on testing set and we obtain the following results as shown in Figure 5 and Figure 6

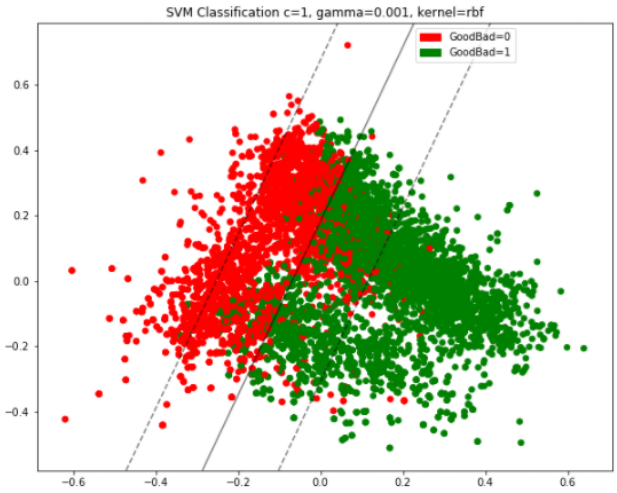

Figure 5: Rbf SVM using PCA: C = 1, gamma = 0.001

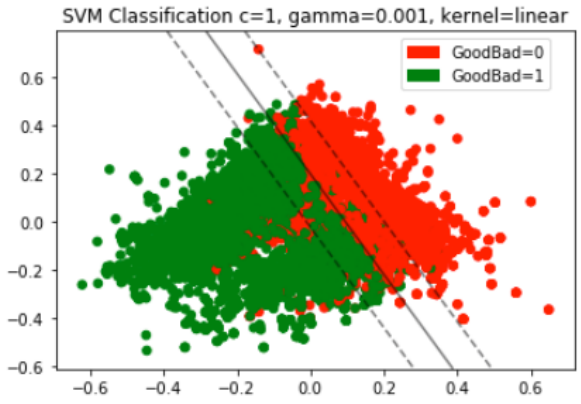

Figure 6: Linear SVM using PCA: C = 1, gamma = 0.001

Linear SVM gave 83% prediction accuracy rate which is pretty good considering how many features we have. Rbf kernel SVM is more scattered compared to Linear SVM and has much lower prediction accuracy rate.

8.2.3 ***K-Nearest Neighbors.*** I am not quite satisfied with the result of SVM so I proceed with another classification algorithm - K-Nearest Neighbors. I used 5-fold cross validation to determine what "K" to choose to give us the best model. The ad-hoc knowledge from machine learning community stated that $K = \frac{1}{m^{0.5}}$ where m is number of samples. As shown in Figure 7, when $K = 40$, we have lowest error prediction rate which is not too far off from the ad-hoc knowledge of $K = 99$. The classification graph with $K = 40$ resulting in prediction accuracy of 95% is shown on Figure 8

8.2.4 ***Decision Tree.*** Using Decision Tree Classification, I modeled it using Gini Index for splitting the tree and obtained 88% prediction accuracy rate. However, the tree is too big to visualize but Figure 9 showed the snapshot for the structure of the tree.

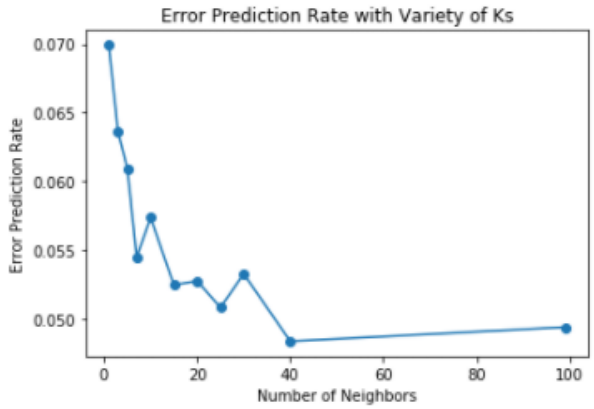

Figure 7: KNN Error Prediction Rate (Validation)

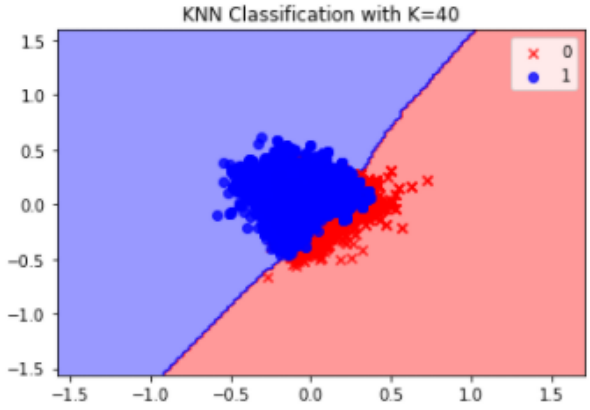

Figure 8: KNN with K=40

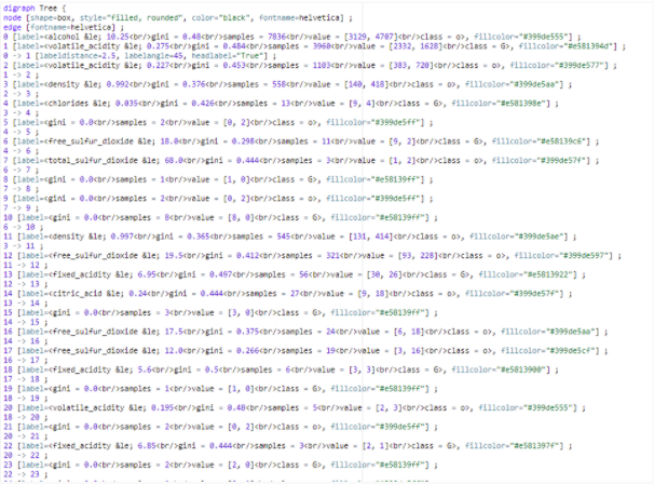

Figure 9: Decision Tree Classification

9 CONCLUSION & DISCUSSION

Through the practice of building regression and classification models for the quality of a wine or classifying whether a wine is good or bad, one can see that it is not easy to build a good model. However, a good model can help industry immensely. The best regression model that has R^2 of 0.3785 might be bad for this particular problem. Although many technique has been performed, it is possible that we can improve the model by iterating through many multi-linear regression model.

Classification algorithms employed have more success using model such as SVM, KNN, and Decision Tree. This practice served as a validation for me because data science can provide a meaningful analysis or potentially do a better task than professional wine taster to predict whether a wine is of good or bad quality.

REFERENCES

[1] Paulo Cortez, António Cerdeira, Fernando Almeida, Telmo Matos, and José Reis. 2009. Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems* 47, 4 (2009), 547–553.