Predicting Wine Quality Using Machine Learning

Columbia University
Mateo Gomez
Jiawei Wu
Yucong Chen
Qinmiao Wei

1. Introduction

Wine quality assessment, a complex interplay of various chemical and sensory factors, has long intrigued oenophiles and industry professionals alike. The intrinsic challenge lies in discerning the nuanced relationship between a wine's chemical composition and the subjective evaluations of experts. In this research, we embark on a comprehensive exploration aimed at developing an accurate forecast model for predicting wine quality.

The quality of wine, often rated by experts on an ordinal scale from 3 to 8, serves as the dependent variable in our predictive model. This ordinal nature of the variable necessitates an approach that goes beyond conventional regression methods. Instead, we leverage both statistical and machine learning techniques to unravel the intricate patterns hidden within the wine's multifaceted attributes.

Our dataset encompasses a rich array of wine properties, including acidity, sugar content, alcohol percentage, sulfates, pH level, and density, among others. These attributes serve as the independent variables, forming the foundation for our predictive models. By incorporating a diverse set of features, we aim to capture the complex interactions that contribute to the overall quality perception of a wine.

The choice to integrate statistical and machine learning methodologies stems from the recognition that a hybrid approach may yield more robust and nuanced predictions. Traditional statistical methods provide interpretability and insights into the underlying relationships, while machine learning algorithms offer the capacity to handle intricate patterns and nonlinearities.

Throughout this research, we will explore and compare various modeling techniques, striving to strike a balance between model complexity and interpretability. Rigorous testing and validation procedures will be employed to assess the performance and generalization capabilities of each model, ensuring the reliability of our predictions.

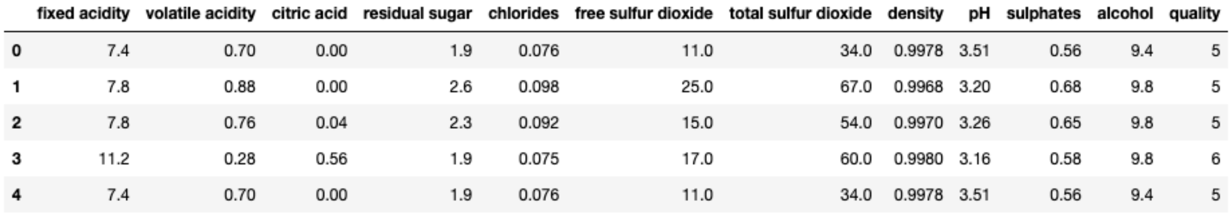
In conclusion, our endeavor to predict wine quality stands at the intersection of domain knowledge, statistical rigor, and machine learning innovation. By combining these elements, we aim to contribute to the evolving landscape of predictive modeling in oenology, offering valuable insights for both researchers and practitioners in the field.

2. Materials and Methods

2.1. Data Source and Structure

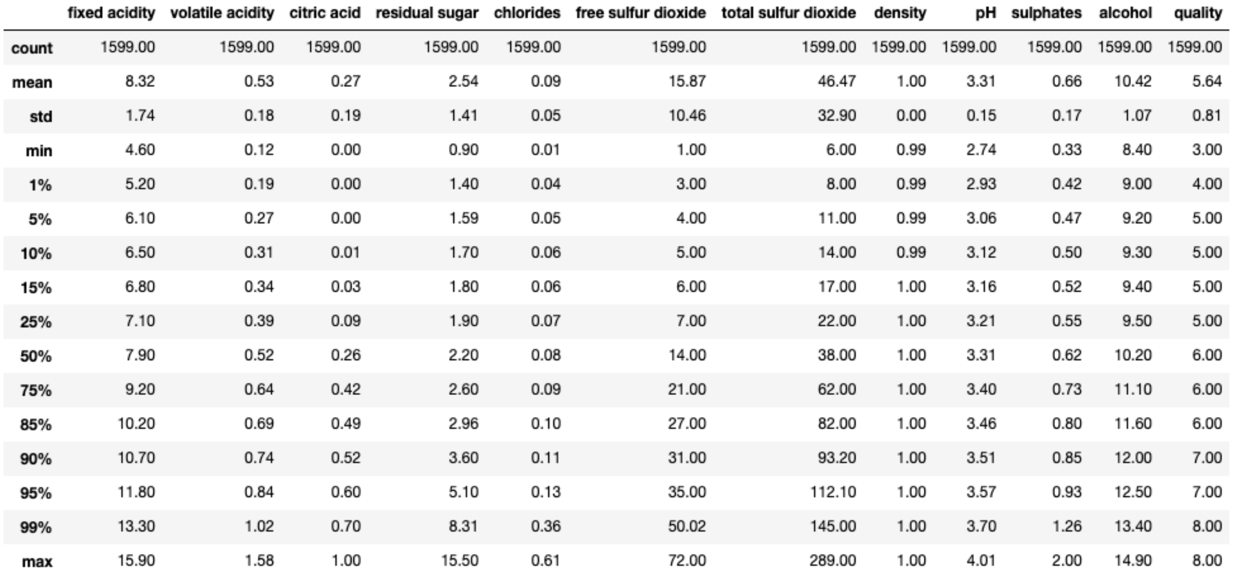
This dataset is related to red variants of the Portuguese “Vinho Verde” wine. For more details, consult the reference [Cortez et al., 2009]. Due to privacy and logistic issues, only physicochemical (inputs) and sensory (the output) variables are available (e.g. there is no data about grape types, wine brand, wine selling price, etc.). The datasets can be viewed as classification or regression tasks. The classes are ordered and not balanced (e.g. there are much more normal wines than excellent or poor ones).

Below a sample of the data and the explanatory variables and forecast variable (quality):

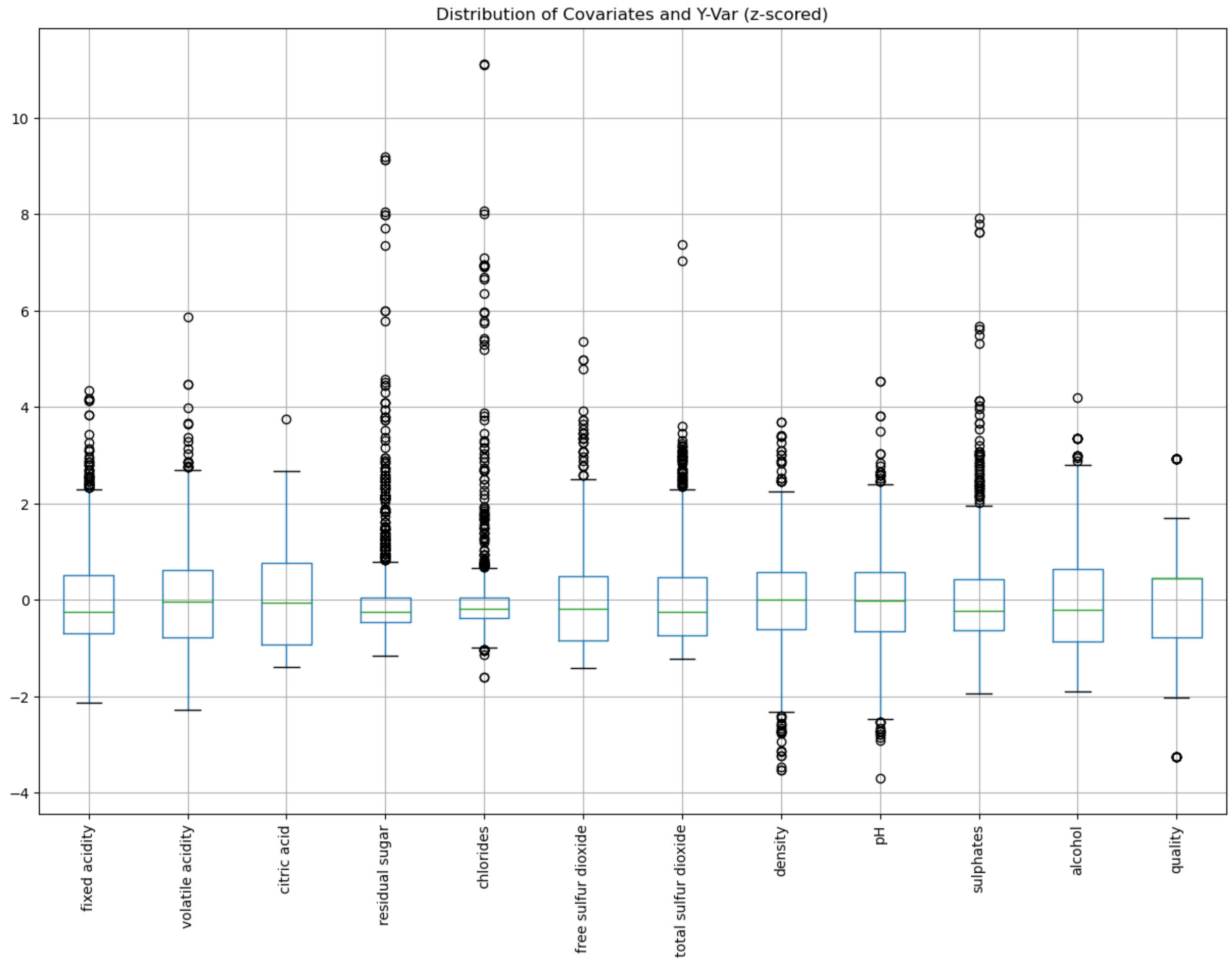


	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5
1	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	0.68	9.8	5
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	0.65	9.8	5
3	11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	0.58	9.8	6
4	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5

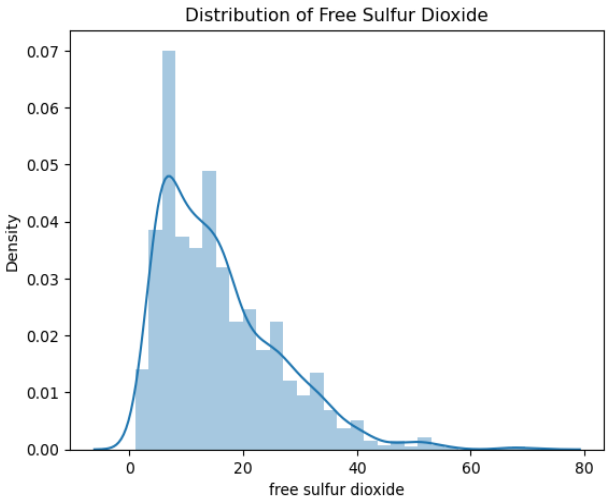
Descriptive Statistics of the data:

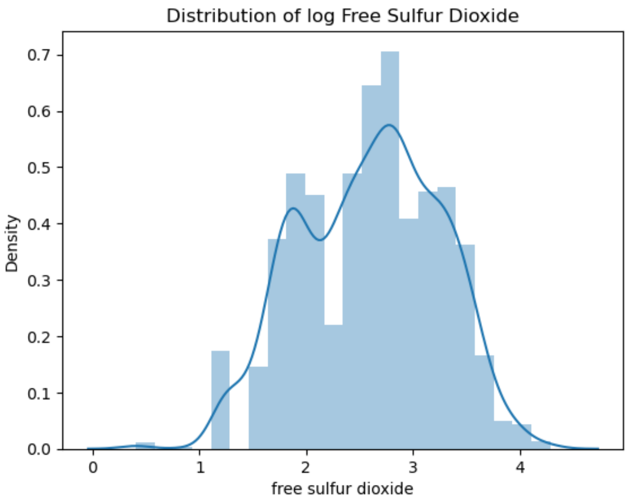


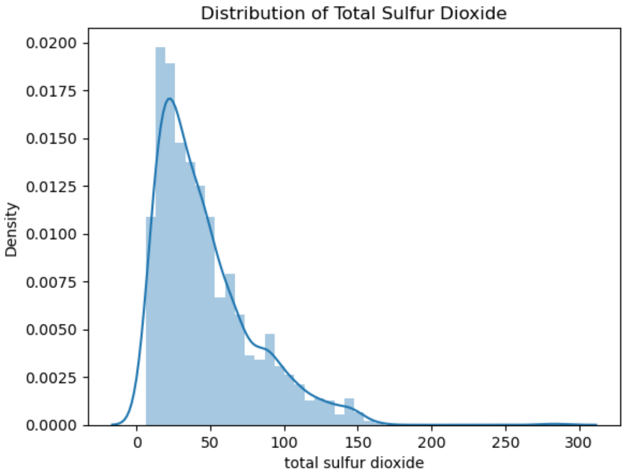
	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
count	1599.00	1599.00	1599.00	1599.00	1599.00	1599.00	1599.00	1599.00	1599.00	1599.00	1599.00	1599.00
mean	8.32	0.53	0.27	2.54	0.09	15.87	46.47	1.00	3.31	0.66	10.42	5.64
std	1.74	0.18	0.19	1.41	0.05	10.46	32.90	0.00	0.15	0.17	1.07	0.81
min	4.60	0.12	0.00	0.90	0.01	1.00	6.00	0.99	2.74	0.33	8.40	3.00
1%	5.20	0.19	0.00	1.40	0.04	3.00	8.00	0.99	2.93	0.42	9.00	4.00
5%	6.10	0.27	0.00	1.59	0.05	4.00	11.00	0.99	3.06	0.47	9.20	5.00
10%	6.50	0.31	0.01	1.70	0.06	5.00	14.00	0.99	3.12	0.50	9.30	5.00
15%	6.80	0.34	0.03	1.80	0.06	6.00	17.00	1.00	3.16	0.52	9.40	5.00
25%	7.10	0.39	0.09	1.90	0.07	7.00	22.00	1.00	3.21	0.55	9.50	5.00
50%	7.90	0.52	0.26	2.20	0.08	14.00	38.00	1.00	3.31	0.62	10.20	6.00
75%	9.20	0.64	0.42	2.60	0.09	21.00	62.00	1.00	3.40	0.73	11.10	6.00
85%	10.20	0.69	0.49	2.96	0.10	27.00	82.00	1.00	3.46	0.80	11.60	6.00
90%	10.70	0.74	0.52	3.60	0.11	31.00	93.20	1.00	3.51	0.85	12.00	7.00
95%	11.80	0.84	0.60	5.10	0.13	35.00	112.10	1.00	3.57	0.93	12.50	7.00
99%	13.30	1.02	0.70	8.31	0.36	50.02	145.00	1.00	3.70	1.26	13.40	8.00
max	15.90	1.58	1.00	15.50	0.61	72.00	289.00	1.00	4.01	2.00	14.90	8.00

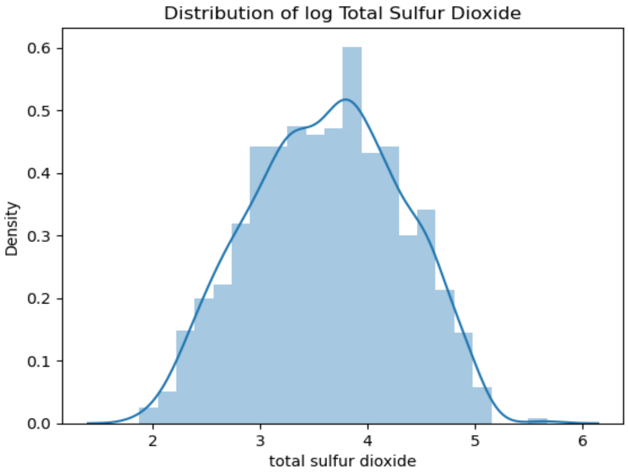


We can observe that this data is in general very clean and does not have missing values. There is presence of outliers in many of the explanatory variables. Most of the variables follow an approximate normal distribution with the exception of fixed acidity, citric acid, residual sugar, free and total sulfur dioxide, alcohol and sulfates. For some of them we take the log to make them more normally distributed and clip the outliers after z-scoring. Some examples below.

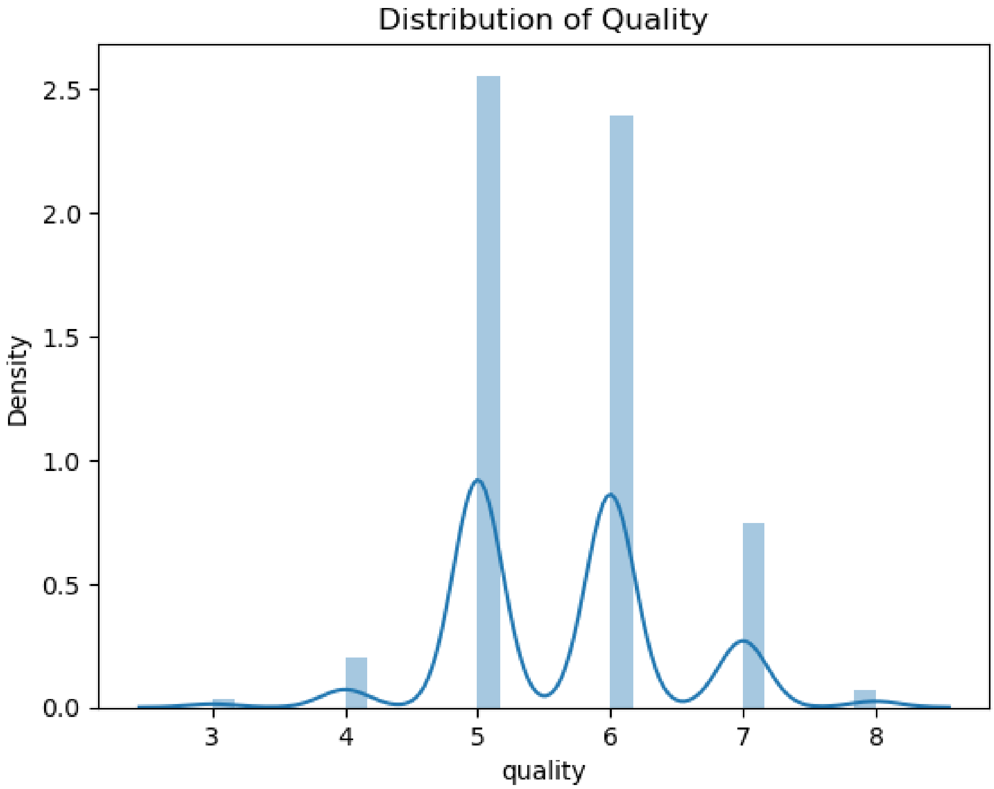






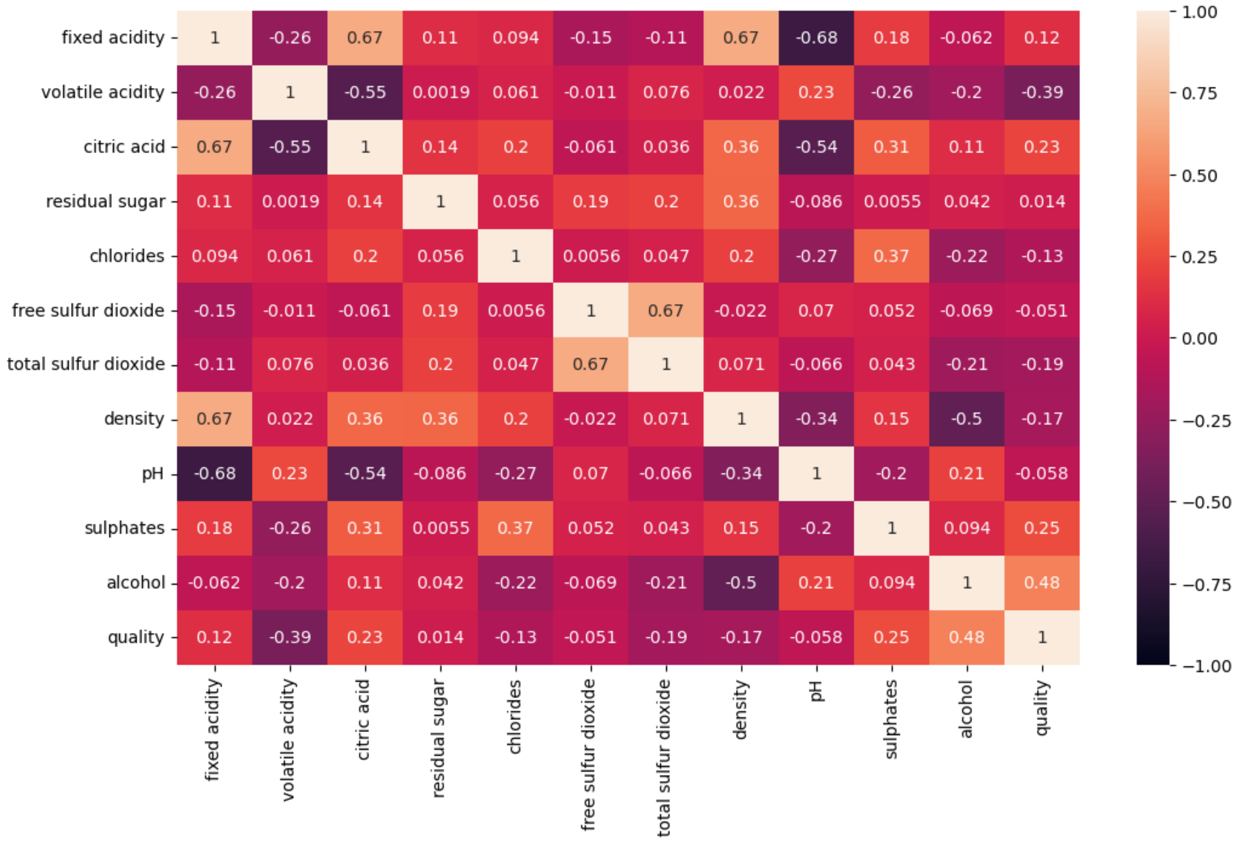


Moreover, we see that the distribution of our predictor variable ("quality of wine") is very dense at "normal" quality levels. This means that most of the wines are neither very good nor very bad. This also means that we can tackle this problem as both a regression and classification exercise. An expansion of this research would be to capture the tails of the below distribution (i.e. train a classifier to identify really good (poor) quality wines), which would use other machine learning techniques for anomaly detection.



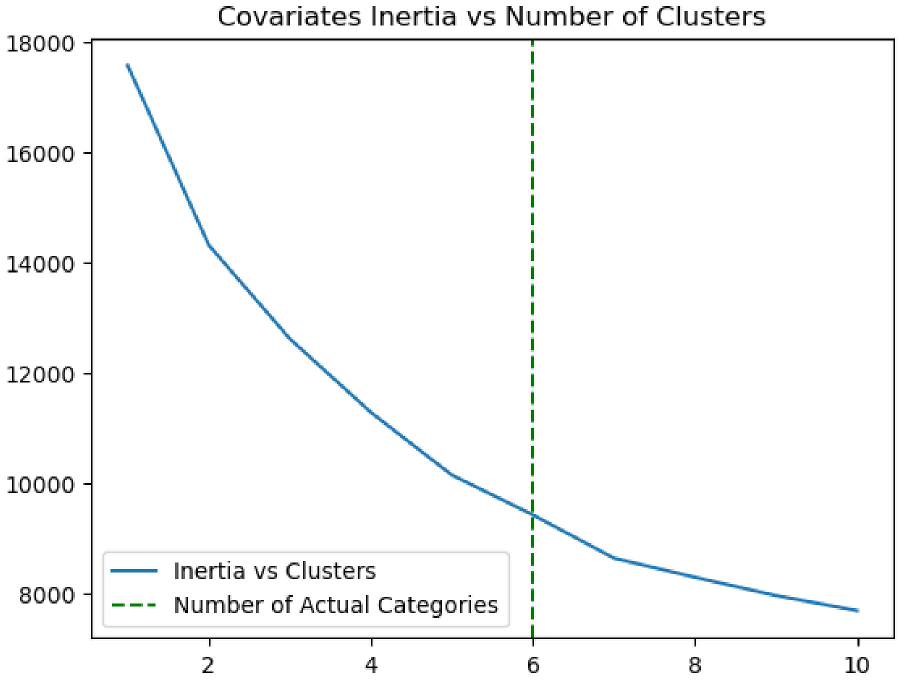
Correlation Plot:

Note that we do not see very high correlation in our covariates. Ph and density have a high degree of correlation with fixed and volatile acidity.



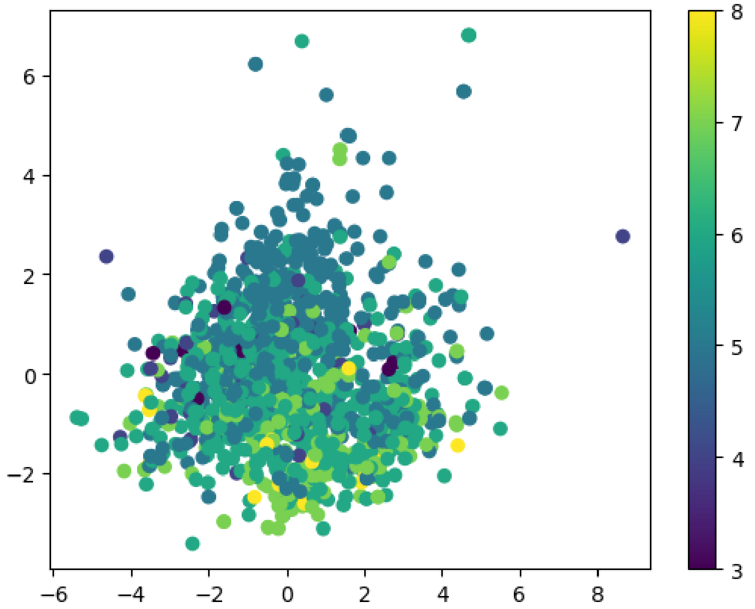
	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
fixed acidity	1	-0.26	0.67	0.11	0.094	-0.15	-0.11	0.67	-0.68	0.18	-0.062	0.12
volatile acidity	-0.26	1	-0.55	0.0019	0.061	-0.011	0.076	0.022	0.23	-0.26	-0.2	-0.39
citric acid	0.67	-0.55	1	0.14	0.2	-0.061	0.036	0.36	-0.54	0.31	0.11	0.23
residual sugar	0.11	0.0019	0.14	1	0.056	0.19	0.2	0.36	-0.086	0.0055	0.042	0.014
chlorides	0.094	0.061	0.2	0.056	1	0.0056	0.047	0.2	-0.27	0.37	-0.22	-0.13
free sulfur dioxide	-0.15	-0.011	-0.061	0.19	0.0056	1	0.67	-0.022	0.07	0.052	-0.069	-0.051
total sulfur dioxide	-0.11	0.076	0.036	0.2	0.047	0.67	1	0.071	-0.066	0.043	-0.21	-0.19
density	0.67	0.022	0.36	0.36	0.2	-0.022	0.071	1	-0.34	0.15	-0.5	-0.17
pH	-0.68	0.23	-0.54	-0.086	-0.27	0.07	-0.066	-0.34	1	-0.2	0.21	-0.058
sulphates	0.18	-0.26	0.31	0.0055	0.37	0.052	0.043	0.15	-0.2	1	0.094	0.25
alcohol	-0.062	-0.2	0.11	0.042	-0.22	-0.069	-0.21	-0.5	0.21	0.094	1	0.48
quality	0.12	-0.39	0.23	0.014	-0.13	-0.051	-0.19	-0.17	-0.058	0.25	0.48	1

Finally, to understand the wine features and its linear relationship with quality we ran a simple clustering exercise (knn). For this exploration we just used the independent variables and tried to find the optimal number of clusters using inertia as a validation metric.

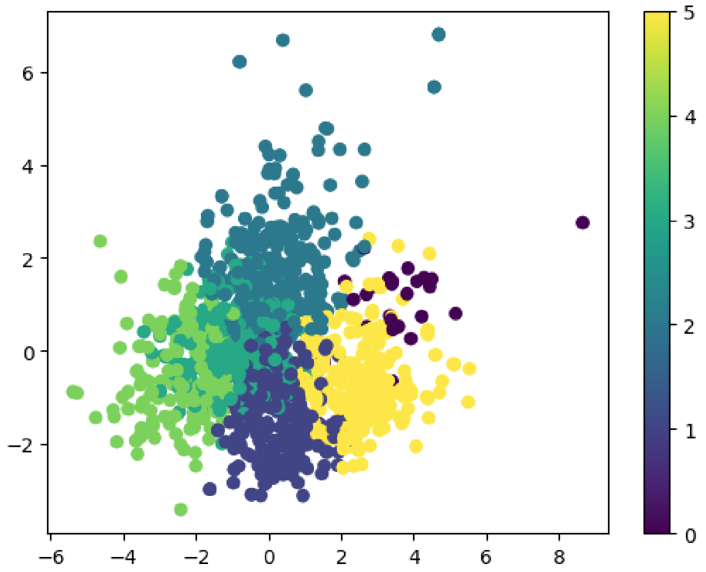


Interestingly we see that the wine features give us an optimal number of clusters between 6 and 8 which corresponds to the number of categories in the dataset. However, when we see the clusters in 2 dimensions implementing PCA on the wine features we see that the relationship between the wine characteristics and its quality is non-linear (if it was we would have seen well distinguished clusters in 2-dimensions):

Clusters of Quality



Clusters Given by KNN



We can clearly observe that the clusters predicted by KNN do not capture the fuzziness of the data. We complement this by calculating the median of each characteristic stratified to each quality group. We observe non linear patterns across many characteristics (see appendix for more details). This indicates that a non-linear method should be more suitable for this dataset. We prove this hypothesis by showing that Random Forest and Boosting algorithms are able to capture the non-linear properties of the dataset.

2.2. Analysis Methods

Linear Methods

Ridge Regression:
Ridge Regression is a regularized linear regression technique that introduces a regularization term, often represented by the L2 norm of the coefficients, to the standard linear regression objective function. This regularization term prevents overfitting by penalizing large coefficients, leading to a more stable and generalizable model. Ridge Regression is particularly useful when there is multicollinearity among the predictor variables.

Principal Component Regression (PCR):
Principal Component Regression combines Principal Component Analysis (PCA) with linear regression. In PCR, the predictor variables are first transformed into principal components through PCA, and then regression is performed on these principal components. This approach is beneficial when dealing with multicollinearity, as it reduces the dimensionality of the data while capturing most of its variability.

LASSO Regression:
LASSO (Least Absolute Shrinkage and Selection Operator) Regression is another form of regularized linear regression, but it employs L1 regularization. LASSO introduces a penalty term based on the absolute values of the coefficients, leading to sparse coefficient estimates. This property allows LASSO to not only prevent overfitting but also perform variable selection, effectively setting some coefficients to exactly zero and, in turn, providing a more interpretable model.

Machine Learning

Random Forest:
Random Forest is an ensemble learning method that builds a multitude of decision trees during training and outputs the mode of the classes (classification) or the mean prediction (regression) of the individual trees. Each tree is constructed using a random subset of the data and a random subset of the features. Random Forests are known for their robustness, scalability, and ability to capture complex relationships in the data.

Classification Approach

Logistic Regression:
Logistic Regression is a popular statistical method used for binary and multiclass classification problems. Despite its name, it is a classification algorithm rather than a regression algorithm.

Logistic Regression models the probability of the occurrence of a binary event using the logistic function. It is widely used due to its simplicity, interpretability, and effectiveness in scenarios where the relationship between predictors and the binary outcome is assumed to be linear.

Elastic Net:
Elastic Net is a method that combines the penalties of both LASSO (L1 regularization) and Ridge (L2 regularization) regression. It is considered due to the potential need of reducing multicollinearity.

Boosting:
Boosting is a powerful machine learning technique that enhances prediction accuracy by combining weak learners sequentially. During this process, each model corrects errors in the previous models. In this way, we can ensure a higher degree of robustness and accuracy. Because of different model approaches, Boosting is also a great example to compare with Random Forest.

3. Results

3.1. Regression Approach

To predict wine quality, we first do a linear regression model for a baseline, which turns out to be very inappropriate. The model has an R^2 of only approximately 0.35, and the assumptions of the model are not satisfied either. To check linearity (functional form), we plot the residuals vs. fitted values on the same graph. We find that residuals are not randomly spread and there exists a relationship between residuals and fitted values, so the model does not satisfy linearity. For normality, from both histograms and qq-plots, we observe that the residuals do not follow a normal distribution very well. Through the Shapiro-Wilk normality test with a p-value < 0.05, we conclude that residuals are not normally distributed. Moreover, we have confirmed that homoscedasticity, autocorrelation of errors, and multicollinearity exist through respective tests.

To reduce multicollinearity, we use PCR, Lasso, and Ridge for analysis. For all models here and below, we use a 7-3 split between training and testing sets on all data. We train our models on the training set and evaluate performance indexes through the testing set. Lasso exhibits a limited degree of sparsity, as it only shrinks the estimate for citric acid to 0, and all other variables are preserved and considered important. Therefore, Lasso may not be as effective for feature selection as anticipated. In terms of MAE and MSE, PCR performs better than Lasso than Ridge, but only slightly. This is possibly because PCR helps reduce variability and removes unimportant information (we reduced the number of components from 11 to 9 according to the RMSEP vs. number of components plot). All three models have an MAE of around 0.52 and an MSE of around 0.45.

Since all linear models do not perform well enough, we employ random forests, aiming to improve model performance and see what variables are the most important to quality. For both %IncMSE (measures increase in MSE, which reflects robustness) and IncNodePurity (which means importance), alcohol, sulphates, and volatile acidity rank the highest. This is an interesting result because alcohol has a very low correlation with quality during our previous EDA analysis. On the other hand, variables like density and pH have a higher correlation with quality but show little importance for the model. In addition to showing information about variables, random forest also gives the lowest MSE (~0.37) on our testing dataset, so random forest is better fitting than all previous models. This is expected. For a comparison of different random forests, we also do another random forest model that limits the number of feature selections for each new split to 5 instead of 11. Both models provide similar MSE (~0.36) and important variables, so there really is not much irrelevant information to reduce. In other words, this change does not affect the model results much. Overall, random forest is the best-fitting model among all.

	PCR	**LASSO**	**Ridge**	**Random Forest**	**Random Forest (with mtry = 5)**
MAE	0.5025666	0.5252256	0.528796		
MSE	0.42845	0.4516902	0.453955	0.3727711	0.364782

Table 1: MAE and MSE Summary

3.2. Classification Approach

For this part, we employ classification models, where we predict the level of quality (high = 1 when quality > 5 and 0 otherwise, which is the new variable we defined to ensure that the number of high and low quality is similar). We use logistic, elastic net, and boosting models. The logistic model gives a fine number of accuracy that is about 70% and is pretty balanced for high and low quality. However, it has a low F1-score, which is about 50%. We then check VIF values and find that a few variables have VIF > 4, which suggests that multicollinearity exists and it is not a fitting model.

Therefore, we use elastic net models, where we apply three different alphas to compare model performance. The three models perform very similarly in terms of MAE (~0.26), MSE (~0.26), and F1-scores (~0.74). They have improved a lot from logistic.

Lastly, we employ boosting models to see relative influence of variables and how they are correlated with level of quality, and we compare results with random forests. From an F1-score of over 80%, we find that boosting is much better than logistic and elastic net. The three most important variables are alcohol, sulphates, and volatile acidity, which agree with random forests. Again, for comparison, we fit another boosting model with a different learning rate, and get similar results in terms of F1-score (~78%) and importance of variables. Therefore, we can conclude that boosting is the best for classification model prediction.

	Logistic	**Elastic Net**	**Boosting**	**Boosting (with learning rate = 0.1)**
F1-score	0.4811552	0.7460317	0.8055556	0.7833002

Table 2: F1-score Summary

4. Conclusion

Overall, we suggest that classification model prediction might be a better choice to analyze this dataset (i.e., we should predict the level of quality instead of quality) because it is easier to carry out and generally gives more straightforward and interpretable results. Boosting is a fitting technique for this purpose. However, if we wish to distinguish quality in more detail, then we might choose random forests to predict quality.

5. Limitations and Further Work

In the future, a more comprehensive dataset is needed for generating more information, even with missing information. This includes more data entries (especially for wine quality other than 5 or 6) and more variables for analysis. For example, more consumer data can be collected so we have a better understanding of how they might view the wine quality in a different perspective from experts and can be helpful to business strategies development as well.

Another interesting exercise would be to split the 2 classes using a different threshold. For example, it would be interesting to explore how a classifier can distinguish between really good wines (quality=8) vs the rest. To tackle this we suggest cost sensitive loss functions on a classifier or an anomaly detection algorithm such as Isolation Forest since both methods can handle highly imbalanced datasets.