Mitali Shroff and Thomas Dohle
CS 6220 Final Report
Northeastern University, Fall 2020

Abstract

A dataset of approximately 6500 wines (red and white) was used to train regression and classification models. The dataset has 11 features, each of which is a chemical property of the wine, and a target, which is a 1-10 rating from a human tester on wine's quality. For both regression and classification tasks, the datasets were split into an 80:20 ratio for training and testing purposes. The regression models used different combinations of features to predict the quality. Three datasets were used - red wine only (R), white wine only (W), and red and white wine together (RW). In the case of R, the strongest model used all 11 features, with a variance of .389. For W, the strongest model used a combination of 4 features, with a variance of .300. For RW, the strongest model used a combination of 3 features, with a variance of .304. All in all, alcohol content was found to be the most compelling predictor and was positively correlated with quality. The classification models used the red and white wine datasets combined, with an added feature "type" where 1 represents red wine and 0 represents white wine. The classification algorithms used are Decision Trees, Support Vector Machine and K-Nearest Neighbors, and were taken from sklearn library. After performing feature selection, a subset of the original dataset was extracted and was fit to the models to see if there would be better accuracy obtained. Overall, the results obtained were above 90% accuracy with Decision Trees and Support Vector Machines performing the best at 98% accuracy.

Introduction

Wine sales in the United States generated around $75 billion in 2019[(1)]. As any wine drinker, novice or connoisseur is aware, not all bottles of wine are created equal. There are many factors that determine the quality of wine, with some people believing that only wines produced in Bordeaux are worth drinking, or that wines not aged in oak barrels are unpalatable. Here, we take a more objective approach in an effort to determine what makes a good wine. Specifically, we analyze a dataset of ~6500 wines[(2)] where each of 11 attributes describes a chemical feature of the wine and the target is a 1-10 score of taste rating from a human tester.

This is not an experiment in chemistry, but rather an application of data mining techniques. We do not seek to reconcile why features such as acidity, residual sugar, or alcohol

content work together to produce good (or bad) wine. Instead, we apply machine learning algorithms to 1) create a regression model that can be used to predict the quality of wine based on its features, and 2) create a classification model that can be used to predict whether a given wine is red or white based on those same features.

The following heat maps can offer a quick preview of the feature correlations:

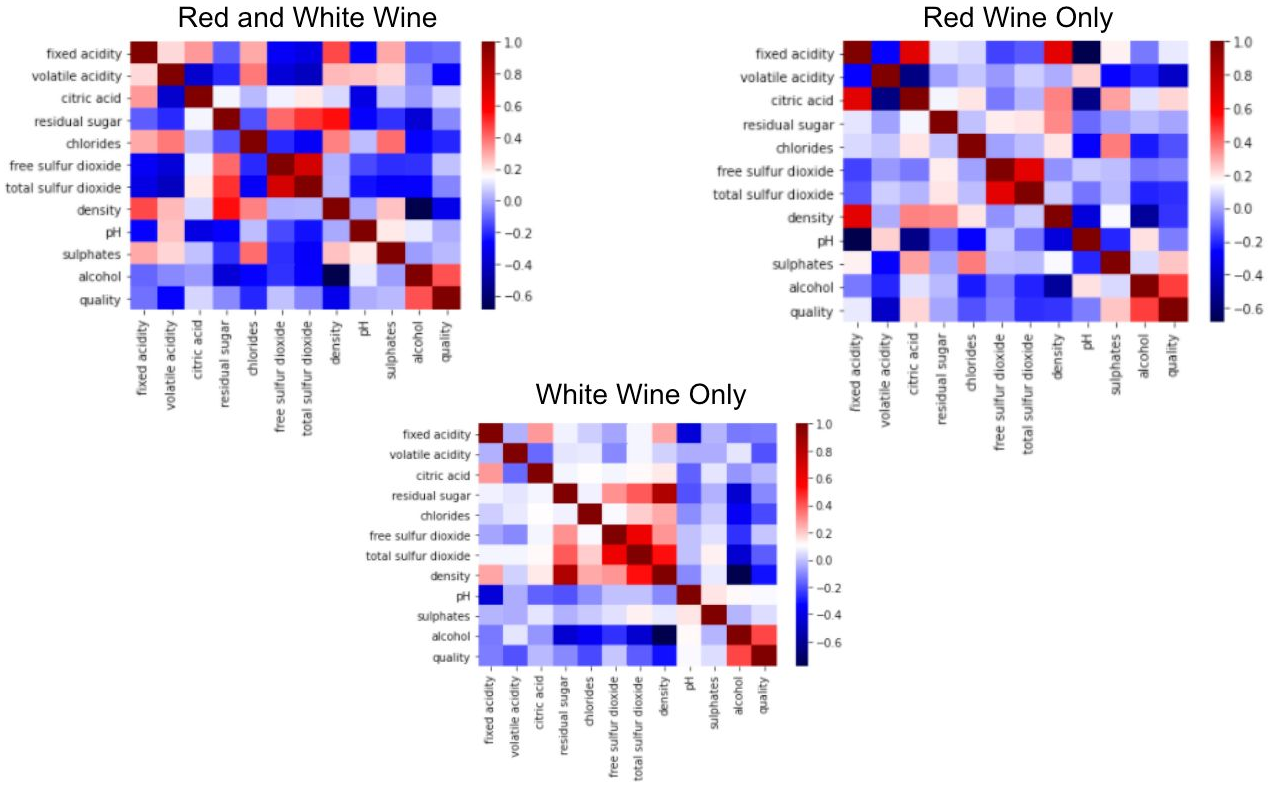

Figure 1.1

The data source is taken from the University of California, Irvine Machine Learning Repository[3] and has been previously used in published work by Cortez, Cerdeira, Almeida, Matos, and Reis[2]. Analysis was performed using the pandas[4], numpy[5], sklearn[6], seaborn[9], and matplotlib[7] open source python libraries powered by jupyter notebooks[7]. All source code is available on github (https://github.com/thomasadohle/Data-mining-project) and an overview of the project can be found at https://sites.google.com/view/neu-cs5500-fall2020.

Methodology

The raw data is loaded into a pandas dataframe. Using pandas, the data is cleaned by removing any entries that were missing feature values. We then split the dataset into training and test data and perform our analysis.

Analysis

Regression

We apply linear regression to three datasets: red wine only, white wine only, and red/white combined. We generate regression models in three ways

1. One feature at a time for all features
2. All features at a time
3. On all combinations of features that have an absolute value correlation >=.1

Classification

For the dataset, we first generate the labels for type of wine (red or white) and add it to the dataset.

We perform feature selection techniques to create a subset of the features that contribute the most to the output.

We then apply three classification algorithms on the entire dataset as well as the subset to gauge which one performs better and whether the subset of important features is better or not.

The three classification techniques are:

- Decision Tree Classification
- Support Vector Classification
- K Nearest Neighbors

Code

The code can be found on github (https://github.com/thomasadohle/Data-mining-project), where you can find instructions to clone the repository and run the code on your local machine. There is a linux/mac virtual environment that is pre-loaded with all necessary dependencies. On a windows machine, you'll need to install the dependencies manually. From the root of the repository, you can accomplish this by running "pip install -r requirements.txt" from a terminal shell.

Regression

Regression analysis was done in the notebook titled "regression_model". The code can be found on github and the notebook has been deployed as a web application at https://calm-dusk-62513.herokuapp.com/.

The method *generate_regression_data* does the bulk of the work of generating the regression model. It takes as argument *wine_type*, which is one of *{'red','white','both'}* and features, which is a list of strings corresponding to the dataset features, which are *{'fixed acidity', 'volatile acidity', 'citric acid', 'residual sugar', 'chlorides', 'free sulfur dioxide', 'total sulfur dioxide', 'density', 'pH', 'sulphates', 'alcohol', 'quality'}*. If no *features* argument is passed, it will generate a regression model using all features. This method splits the data into testing and training sets using the *test_train_split* method from *sklearn.model_selection* and then uses the *LinearRegression* class from *sklearn.linear_model* to generate the model. It returns a dictionary containing the regression model, the data used, and various metrics describing the quality of the model, including variance and mean squared error.

All analysis was performed on three groups of data: red and white wine together, only red wine, and only white wine.

In the first analysis section, regression models are generated for each feature independently. *Matplotlib.pyplot* is used to generate scatter plots of the regression. We then verify for each set of data (red and white, red only, white only) that the three features that generated regression models with the highest variance are the same as the features that correlate most strongly with the target ('quality'). We used *seaborn* to generate heat maps of the correlations.

In the second analysis section, regression models are generated for all features simultaneously.

The third analysis section makes use of the function *find_highest_variance_regression*, which takes the same arguments as *generate_regression_data* (and in fact calls that method). It generates regression models for all possible combinations of features passed in and returns the model with the highest variance. For each dataset, we used features for which the absolute value of the correlation is >=.1 in order to decrease computational complexity, as there are 2^n combinations of n features. Even so, for the red wine dataset, there were 8 features with absolute value correlations >=.1, so we compute 256 regression models.

Classification

The code has been deployed as a web app at https://arcane-anchorage-89293.herokuapp.com/, and it can be found on GitHub here.

We first load the red wine and white wine datasets into the jupyter notebook. Since we are performing classification based on type of wine, we introduce this feature as a binary classification. (0 for white wine, 1 for red wine)

After some checks for null and missing values in the dataset, we perform a feature set analysis to determine which features could be more important in the dataset.

A correlation heatmap shows some interesting correlations between wine type and the rest of the features. Another technique implemented is the Extra Trees classifier that fits a number of randomized decision trees on sub-samples of the dataset and returns a ranked list of the important features. From both these methods, we can say that total sulfur dioxide, volatile acidity, chlorides, and sulphates are important features to consider. Features like fixed acidity and density may also play a role in classification.

We then split the dataset into training and test in an 80:20 split. We also create a subset of the entire dataset with only the aforementioned features due to their importance.
Next, we create the classification models as follows:
Method decision_tree_classification
This method fits the sklearn DecisionTreeClassifier to the dataset for multiple max_depths. A simple fit with default max_depth created 18 levels, so we check for all levels up to 18 to find out the best fit for the training data.
Method svm_classification
The Support Vector Classifier is a simple version with a linear kernel. We use SVC from the sklearn library.
Method knn_classification
K-Nearest Neighbor is run for 10 iterations, each time increasing the value of k. We print results of the model with the highest accuracy score.

Finally, we run these classification models on both the entire wine dataset as well as a subset of the features. The result is recorded in the form of the confusion matrix, the accuracy score and a classification report with the precision, recall, f1-score and support.

Results

Regression

All regression models were trained with 80% of the data and tested with the remaining 20%. Please note that, while we report 3 significant digits, all values can change slightly due to the stochasticity of splitting the initial data into test/train sets.

Single feature regression models

Red and white wine together (n=6497)

Feature	Coefficient	Mean Squared Error	Variance
Fixed acidity	-.055	5.81	.003
Volatile acidity	-1.42	5.80	.059
Citric acid	.498	5.81	.009
Residual sugar	-.008	5.81	-.001
Chlorides	-4.87	5.82	.049
Free sulfur dioxide	.004	5.82	-.013
Total sulfur dioxide	-.001	5.82	.001
Density	-88.3	5.81	.095
pH	.103	5.82	.000
Sulphates	.194	5.82	.003
Alcohol	.325	5.80	.211

The best single-feature regression model (based on variance score) for red and white wine is alcohol, which matches what we expect based on the correlation scores (see figure 1.1).

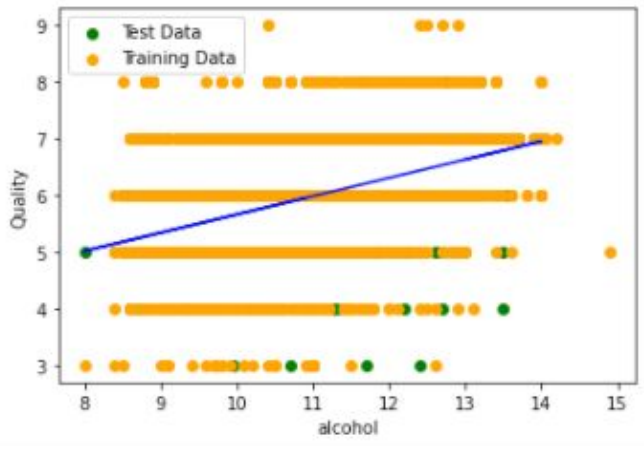

Figure 3.1

Red wine (n=1599)

Feature	Coefficient	Mean Squared Error	Variance
Fixed acidity	.058	5.65	.010
Volatile acidity	-1.84	5.62	.115
Citric acid	1.02	5.64	.018
Residual sugar	.009	5.65	-.004
Chlorides	-2.73	5.65	.023
Free sulfur dioxide	.-.003	5.63	.004
Total sulfur dioxide	-.004	5.64	.048
Density	-85.4	5.63	-.004
pH	-.138	5.65	.004
Sulphates	1.29	5.62	.027
Alcohol	.373	5.64	.140

The best single-feature regression model (based on variance score) for red wine is alcohol, which matches what we respect based on the correlation scores (see figure 1.1).

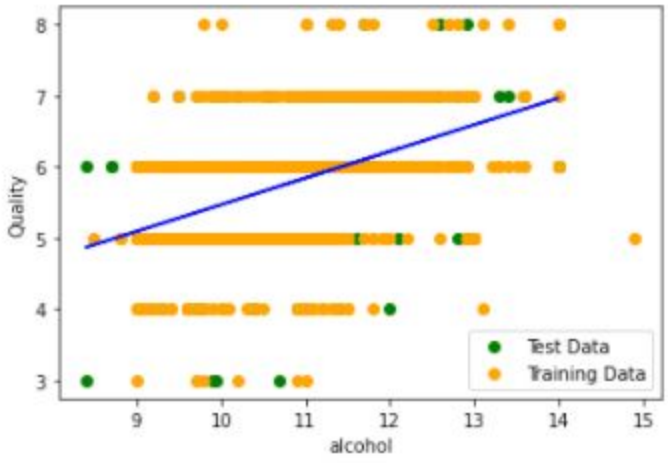

Figure 3.2

White wine (n=4898)

Feature	Coefficient	Mean Squared Error	Variance
Fixed acidity	-.106	5.87	.022
Volatile acidity	-1.72	5.87	.032
Citric acid	-.148	5.87	-.003
Residual sugar	-.016	5.88	.015
Chlorides	-8.42	5.88	.045
Free sulfur dioxide	.001	5.87	.000
Total sulfur dioxide	-.004	5.88	.022
Density	-86.4	5.88	.132
pH	.581	5.88	.010
Sulphates	.408	5.86	-.001
Alcohol	.314	5.87	.187

The best single-feature regression model (based on variance score) for white wine is alcohol, which matches what we respect based on the correlation scores (see figure 1.1).

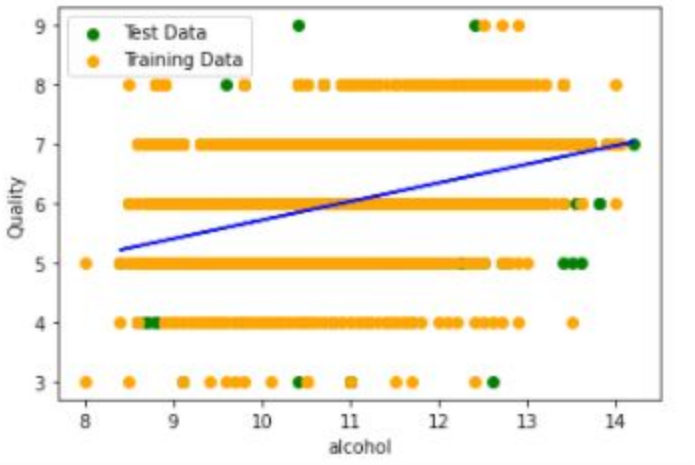

Figure 3.3

All feature regression models
These models were trained using all 11 features in the datasets.

	Red and white	Red only	White only
Fixed acidity coefficient	.074	.015	.007
Volatile acidity coefficient	-1.36	-1.10	-1.82
Citric acid coefficient	-.105	-.199	.043
Residual sugar coefficient	.045	.004	.078
Chlorides coefficient	-.377	-2.06	-.365
Free sulfur dioxide coefficient	.006	.006	.005
Total sulfur dioxide coefficient	-.003	-.004	-.001
Density coefficient	-.577	-11.0	-145
pH coefficient	.454	-.427	.708
Sulphates coefficient	.738	.932	.637
Alcohol coefficient	.258	.265	.199
Mean squared error	5.79	5.62	5.86
Variance	.291	.389	.264

Regression models using only features with high correlation
These models were trained using only features with an absolute value correlation >=.1 for each respective dataset. The optimum model for all wines and red wines used 3 features, while the optimum model for white wines used 4 features.

	Red and white	Red only	White only
Volatile acidity coefficient	-1.35	-1.38	-1.99
Chlorides coefficient	.332		-1.07
Density coefficient		27.8	35.9

Alcohol coefficient	.314	.328	.381
Mean squared error	5.82	5.64	5.87
Variance	.304	.332	.300

Classification

Accuracy scores for various classification techniques

	Full wine dataset	Subset of features
Decision Tree	0.97923	0.98231
Support Vector Machine	0.98539	0.97846
K Nearest Neighbors	0.95154	0.94769

- We see that all the models perform well on this dataset
- The best performing was the simple Linear Support Vector Machine on the entire dataset, followed by the Decision Tree classification model on the subset of important features
- Choosing a subset of important features via feature selection created comparably accurate models
- We used a standard scaler from sklearn and reached an accuracy of 99% for SVM and KNN on the entire dataset

Discussion

Regression

In order to avoid bias, the regression models using one feature at a time as well as the models using all features were created before investigating any correlations.

The strongest regression model (scored by variance) for each of the 3 datasets is unique. In the case of red wine, the model which used all 11 features performed best. The white wine model and red and white wine together model used 4 and 3 features, respectively. It does seem counterintuitive that blindy using all features on the red wine dataset outperformed our attempts are more thoughtful feature selection, but the data does what the data does.

Alcohol content plays a key role in quality, according to our regression analysis. For all 3 datasets, the single-feature regression models using alcohol content had the highest variance. Had we stopped there, it would seem that the best advice we could offer winemakers is to produce wine with the highest possible alcohol content. However, if we look at the data, the highest alcohol content wine has only 14.9% alcohol, and some types of wine routinely contain that much alcohol[(10)]. However, alcohol content should not be discounted, because the mean quality score for wines with >13% alcohol is almost a point higher than the mean quality score of all wines. It is also worth noting that we were unable to train a model with a higher variance score using the techniques of including all features nor all iterations of high correlation features than the models that use only alcohol content. It is also worth noting that advising winemakers to arbitrarily increase the alcohol content in their wines may be infeasible due to regulations.

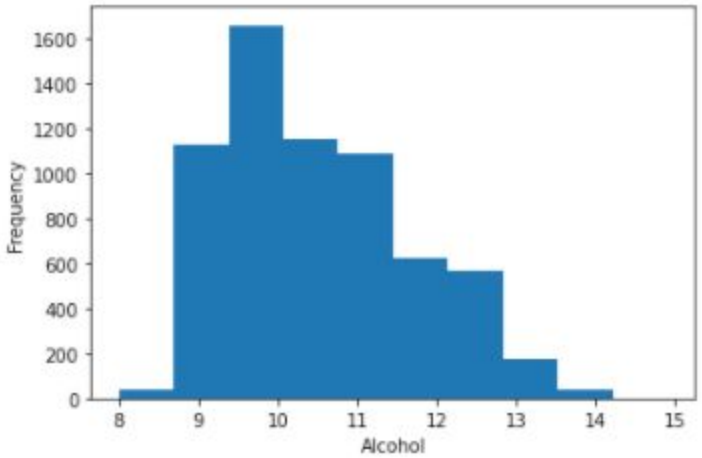

Figure 4.1

We stated from the beginning that analysis was not meant to be an exercise in chemistry. It does, however, seem odd that pH did not come out as a relevant factor in our regression models. It is also interesting to note that there is not an obvious difference between the pH of red and white wine.

	All wines	Red wine	White wine
Average pH	3.22	3.31	3.19

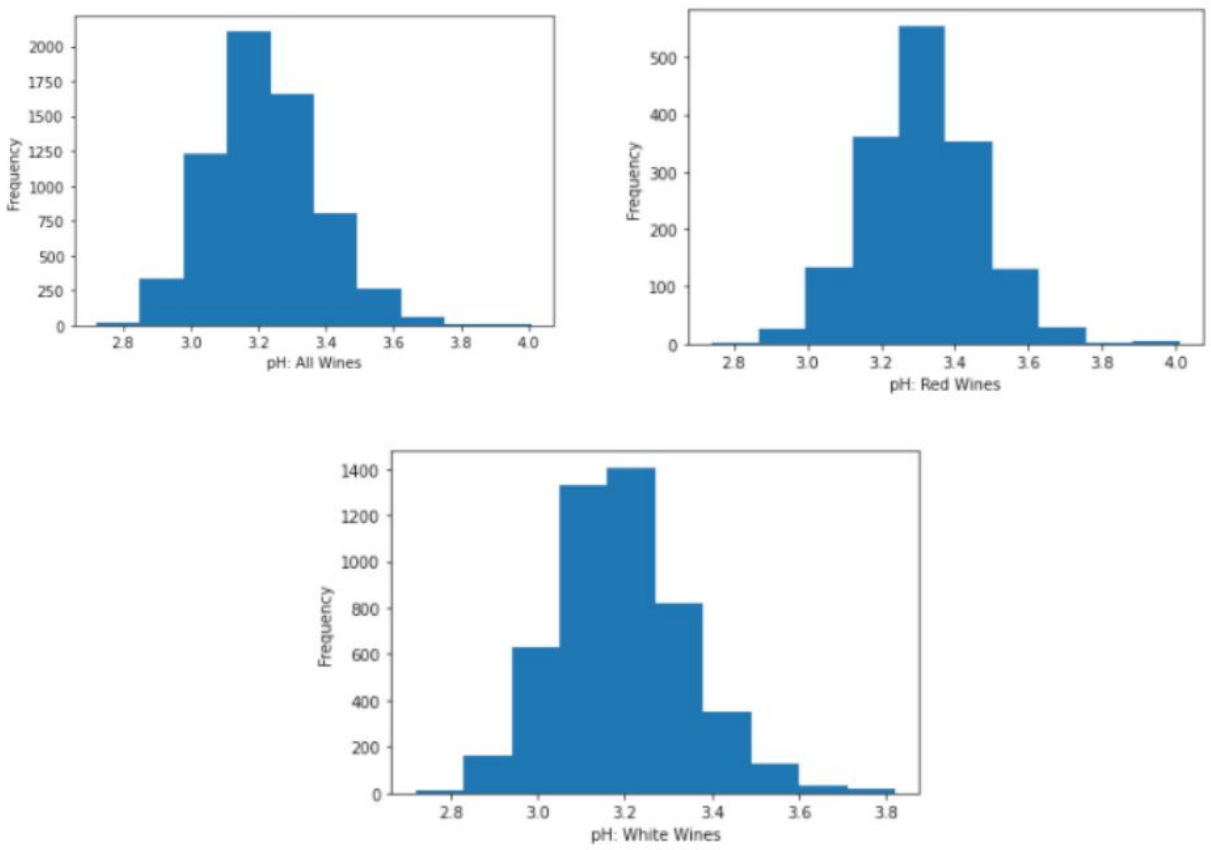

Figure 4.2

Density is an interesting factor. While it correlates negatively with quality in all three datasets (see figure 1.1), it is much more strongly predictive in white wines than red, with coefficients of -145 and -11.0, respectively. It is also worth noting that the distribution of density values in red wines is more of a bell curve while the white wine density values are more uniform. The mean quality of white wines with a density>1 is 5.55, while the average quality of white wines with a density <.995 is 6.04.

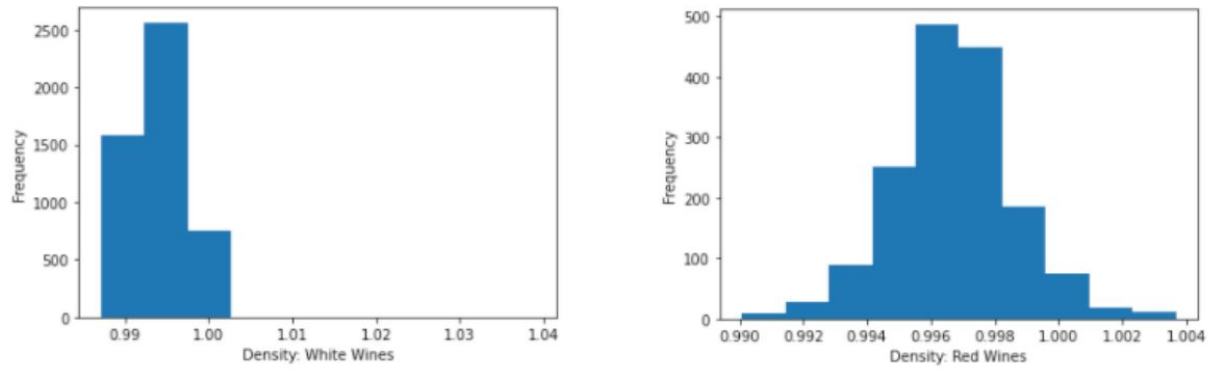

Figure 4.3

Classification

We wanted to look at the features and investigate if there was some subset of the feature set that encompassed most of the properties in the original dataset, which could predict whether a wine was red or white accurately.[11]

Given below is a heat map representing the correlation between the features in the dataset. What is of interest to us is how each feature correlates with type. From here, we can choose those features that have a high positive or negative correlation with type, as they would be likely to contribute the most to the predictions.

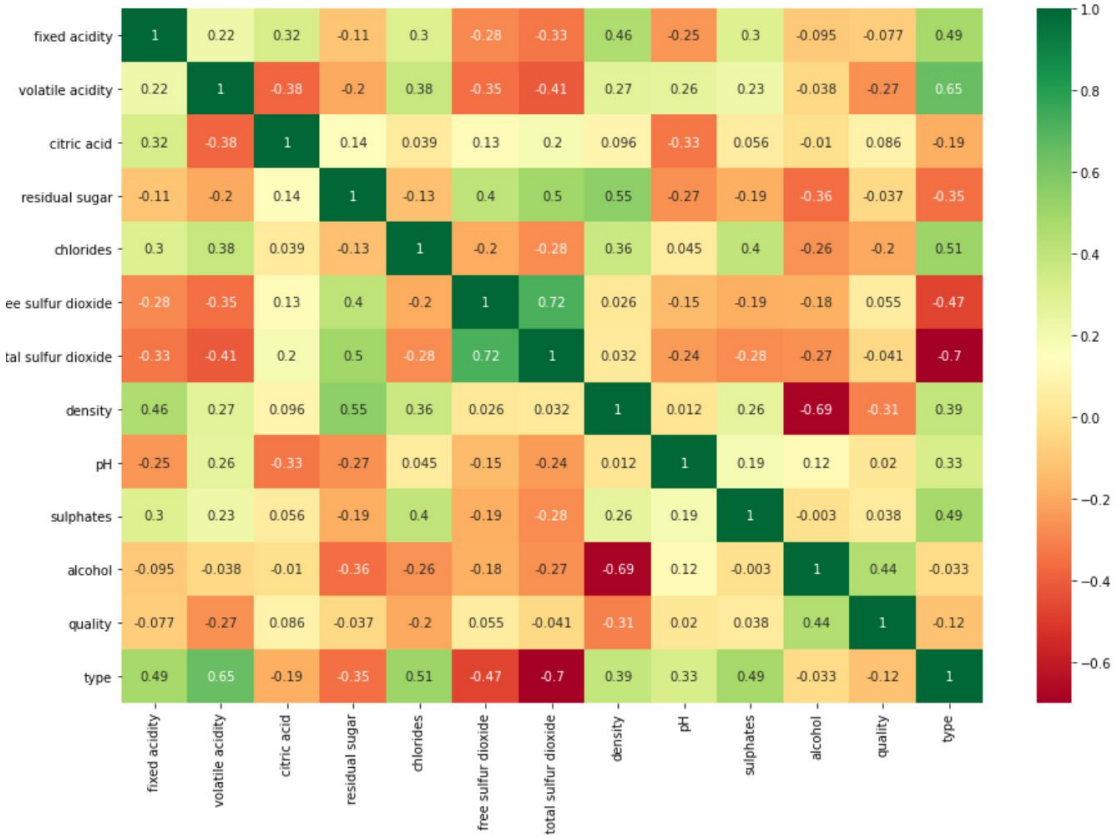

Figure 4.4

We also used another method to find important features called the Extra-trees classifier[12]. The higher the impurity-based feature importance, the more important the feature. The importance of a feature is computed as the (normalized) total reduction of the criterion brought by that feature. The results are shown as given in Fig, 4.5.

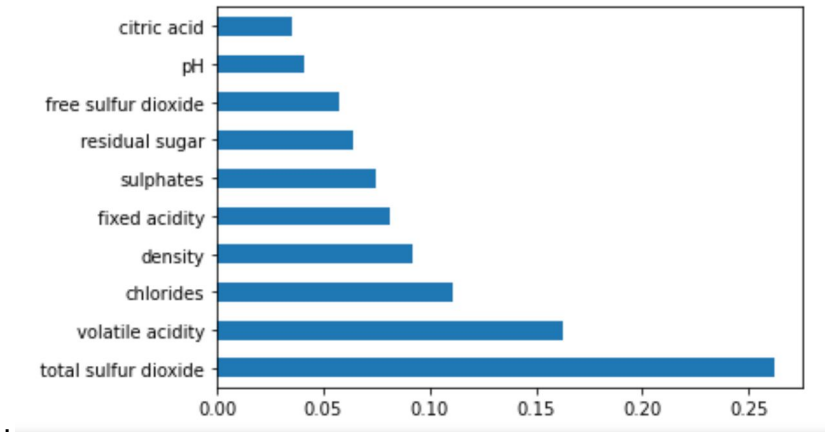

Figure 4.5

By trying different classification techniques, we could compare them on the same dataset and see which one would perform better. To keep it uniform, we used the classification methods in sklearn, so we could compare on similar metrics.

For Decision tree classification, we got the maximum default tree depth (18), and then tried to set the max_depth to all numbers ranging from 1 to 18, to better fit the model.

For SVC, we tried using different kernels for our classification model to see which would work better, and for this case we noticed that linear SVC performed best.

For K-Nearest Neighbor classification, we run the algorithm for different numbers of nearest neighbors to, again, find the best fitting model for the data.

We also generated a subset of the dataset with the important features to see if that would improve the accuracy. The original wine dataset generated good accuracy on the testing data after being trained.

We tried different combinations of the feature sets, but the results were always comparable to using the entire dataset as input.

We tried using StandardScaler from sklearn[13], Standardize features by removing the mean and scaling to unit variance. This can help with more accurate prediction of classifiers.

Future Work

Regression

While we were able to generate solid regression models for all three datasets, it would be nice to be able to validate the models with independent data. This would be especially beneficial

since all the wines in the dataset that was used are from the same region. It is possible that regional differences in soil chemistry, growing practices, etc could lead to variation.

At its current state, the work is only usable by those with a background in computer science with the skills to pull down the code from github, load their own data into the notebook, and so on. Ideally this would be accessible as a web application that would allow people to enter data about their own wine and get predictions from the regression model. In the future, we would like to deploy this code as a full stack django or flask web app so that it can be useful for anyone with a web browser and basic computer skills.

Additionally, there are many other types of wines that were not included in the regression analysis, such as rose and prosecco. Securing additional datasets featuring these other varieties of wine and creating useful regression models with them would make this application more robust and useful.

Classification

For classification on wine type, we could look into more classification techniques out there to make it a more rounded comparison. We could also do additional tweaks on our existing methods to improve accuracy. Exploring the risk of overfitting any of the models could also be an aspect to study.

Another interesting approach could be to classify wine based on quality (good, average and bad quality wine). We could apply the same classification techniques and see if we get similar results.

References

1. https://wineinstitute.org/our-industry/statistics/california-us-wine-sales/
2. P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis.
   Modeling wine preferences by data mining from physicochemical properties. In Decision Support Systems, Elsevier, 47(4):547-553, 2009
3. https://archive.ics.uci.edu/ml/datasets/Wine+Quality
4. https://pandas.pydata.org/
5. https://numpy.org/
6. https://scikit-learn.org/stable/
7. https://matplotlib.org/
8. https://jupyter.org/

9. https://seaborn.pydata.org/
10. https://www.alcohol.org/statistics-information/abv/
11. https://machinelearningmastery.com/feature-selection-machine-learning-python/
12. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.ExtraTreesClassifier.html
13. https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html