2020

FINAL PROJECT

CONTRIBUTORS
Melissa Morgan
Blake Skinner
Godwin Thomas
Julia Thompson
Rebekah Vinson

The Effect of Physicochemical on the Wine Quality | *data **dorks** seeking to help out the **cork dorks***

PROJECT OVERVIEW

Objective:

Develop a machine learning model using the chemical makeup of wine to predict its quality.

Data Source:

Paulo Cortez, University of Minho, Guimarães, Portugal, http://www3.dsi.uminho.pt/pcortez/wine/
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.

Data Files:

Two datasets (.csv files) containing the following:

- Sample set of 1,599 red wines and 4,898 white wines
- 12 attributes for each wine (11 + output variable)

Data Attributes:

Input variables (based on physicochemical tests):

1 - fixed acidity
2 - volatile acidity
3 - citric acid
4 - residual sugar
5 - chlorides
6 - free sulfur dioxide
7 - total sulfur dioxide
8 - density
9 - pH
10 - sulphates
11 – alcohol

Output variable (based on sensory data):

12 - quality (median score by at least three wine experts grading quality on a scale of 0 to 10)

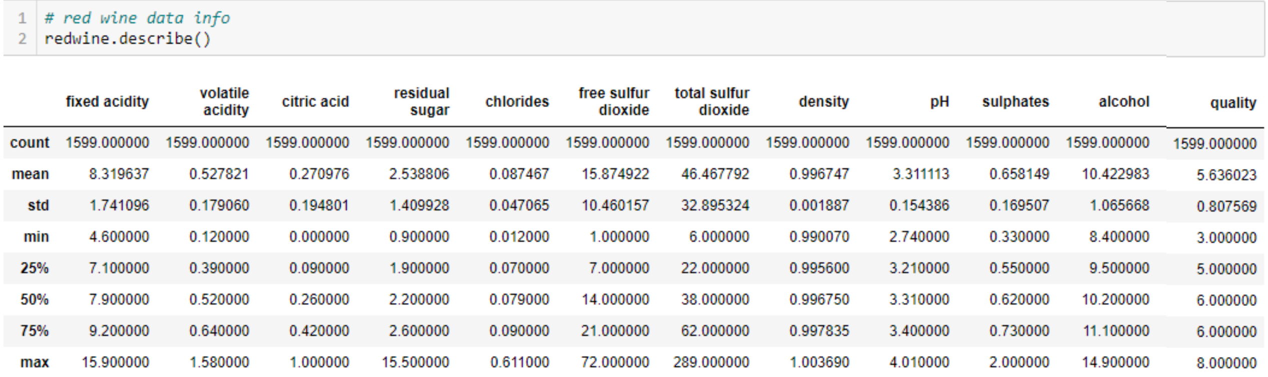
Data Source Note:

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.).

Data Exploration:

Our initial exploration of the data consisted of importing the .csv files into a Jupyter notebook and running code to gain quick insight and comparisons of both sets of data.

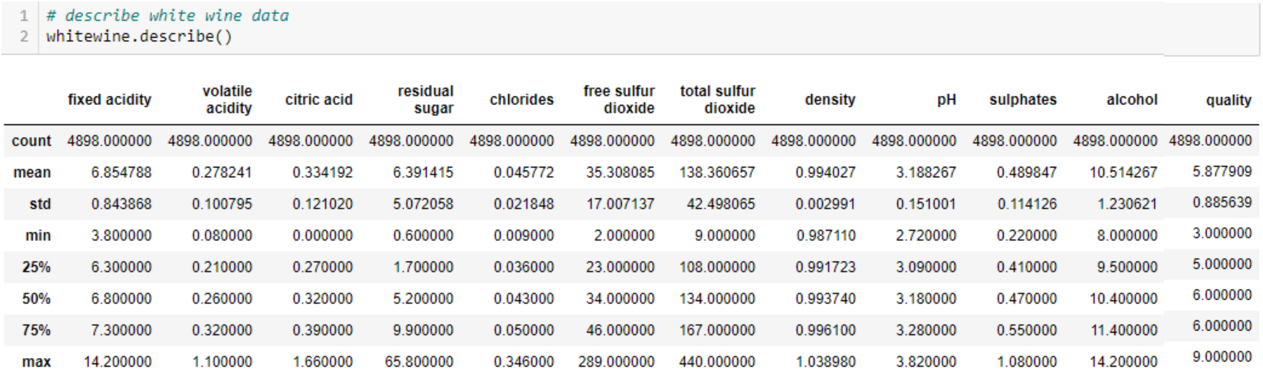
Red Wine Statistics



```
# red wine data info
redwine.describe()
```

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
count	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000
mean	8.319637	0.527821	0.270976	2.538806	0.087467	15.874922	46.467792	0.996747	3.311113	0.658149	10.422983	5.636023
std	1.741096	0.179060	0.194801	1.409928	0.047065	10.460157	32.895324	0.001887	0.154386	0.169507	1.065668	0.807569
min	4.600000	0.120000	0.000000	0.900000	0.012000	1.000000	6.000000	0.990070	2.740000	0.330000	8.400000	3.000000
25%	7.100000	0.390000	0.090000	1.900000	0.070000	7.000000	22.000000	0.995600	3.210000	0.550000	9.500000	5.000000
50%	7.900000	0.520000	0.260000	2.200000	0.079000	14.000000	38.000000	0.996750	3.310000	0.620000	10.200000	6.000000
75%	9.200000	0.640000	0.420000	2.600000	0.090000	21.000000	62.000000	0.997835	3.400000	0.730000	11.100000	6.000000
max	15.900000	1.580000	1.000000	15.500000	0.611000	72.000000	289.000000	1.003690	4.010000	2.000000	14.900000	8.000000

White Wine Statistics



```
# describe white wine data
whitewine.describe()
```

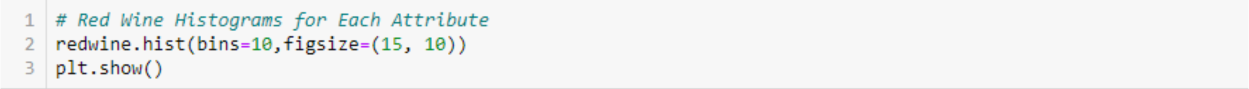
	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
count	4898.000000	4898.000000	4898.000000	4898.000000	4898.000000	4898.000000	4898.000000	4898.000000	4898.000000	4898.000000	4898.000000	4898.000000
mean	6.854788	0.278241	0.334192	6.391415	0.045772	35.308085	138.360657	0.994027	3.188267	0.489847	10.514267	5.877909
std	0.843868	0.100795	0.121020	5.072058	0.021848	17.007137	42.498065	0.002991	0.151001	0.114126	1.230621	0.885639
min	3.800000	0.080000	0.000000	0.600000	0.009000	2.000000	9.000000	0.987110	2.720000	0.220000	8.000000	3.000000
25%	6.300000	0.210000	0.270000	1.700000	0.036000	23.000000	108.000000	0.991723	3.090000	0.410000	9.500000	5.000000
50%	6.800000	0.260000	0.320000	5.200000	0.043000	34.000000	134.000000	0.993740	3.180000	0.470000	10.400000	6.000000
75%	7.300000	0.320000	0.390000	9.900000	0.050000	46.000000	167.000000	0.996100	3.280000	0.550000	11.400000	6.000000
max	14.200000	1.100000	1.660000	65.800000	0.346000	289.000000	440.000000	1.038980	3.820000	1.080000	14.200000	9.000000

Data Attribute Exploration:

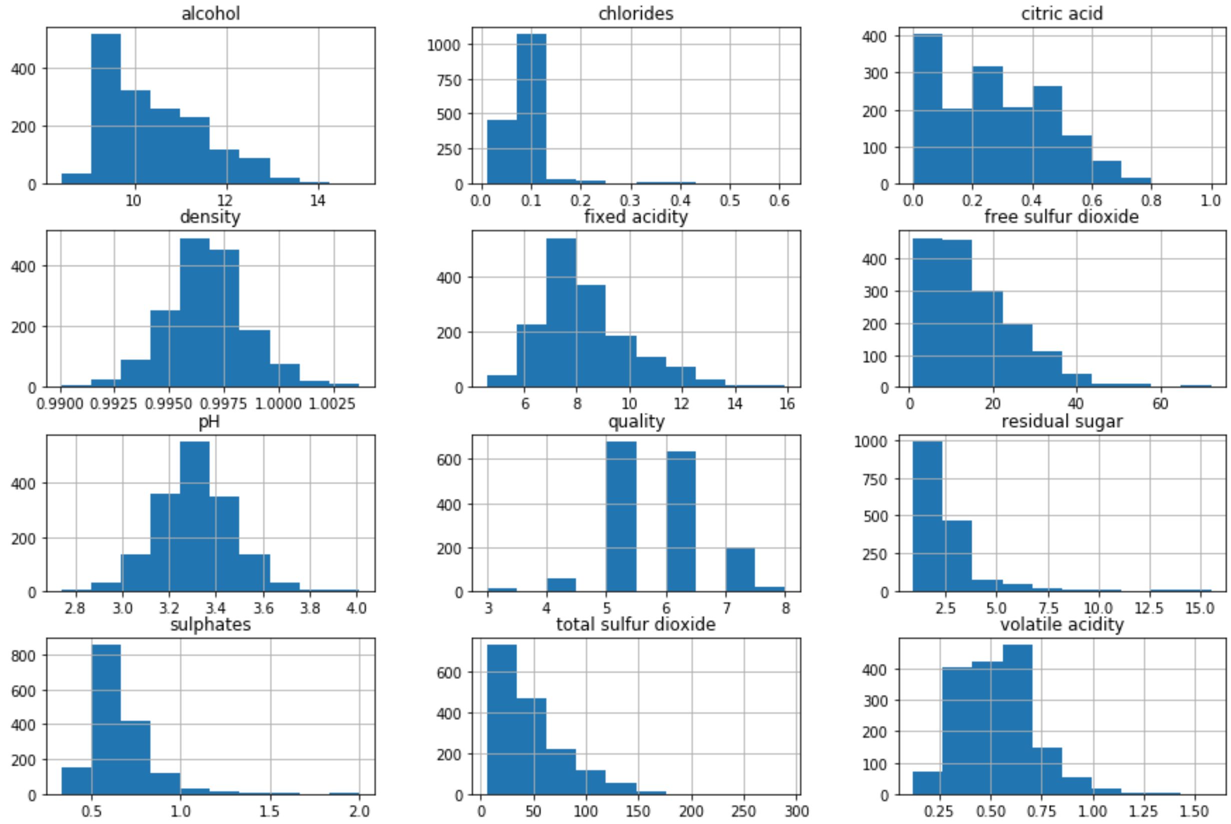
Created histograms to get a visual understanding of data distribution and any levels of skewness for each attribute (hoping for normal distribution of the sample data).

Attribute histograms for each type of wine are provided on the following two pages.

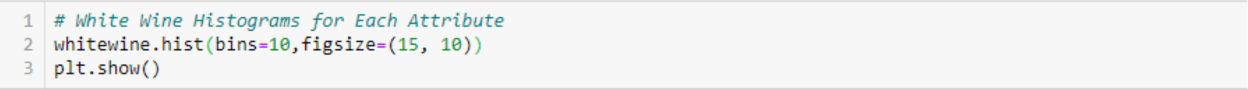
Red Wine Histograms



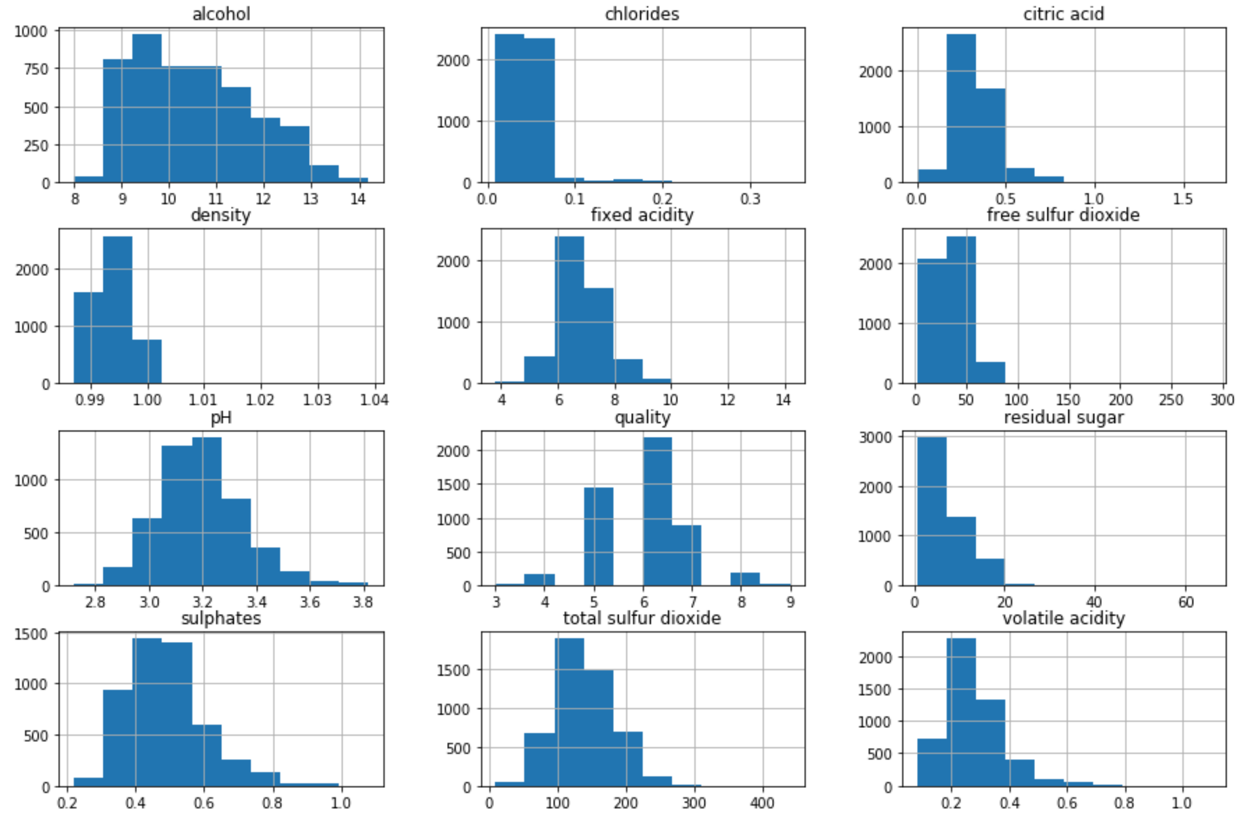
```
# Red Wine Histograms for Each Attribute
redwine.hist(bins=10,figsize=(15, 10))
plt.show()
```



White Wine Histograms



```python
# White Wine Histograms for Each Attribute
whitewine.hist(bins=10,figsize=(15, 10))
plt.show()
```



Exploration Results:

First, we made the decision to keep the red and wine white data (and ML Model) separate because a comparison of the data distribution and statistics showed a clear difference between the physicochemical variable of each type of wine.

Ultimately, after reviewing supplementary output to our initial data exploration, we selected to proceed with the **White Wine** dataset for this project.

PROJECT DETAILS

Model Process – White Wine Quality Predictor

Looked at the Datatypes

- Our datasets did not any missing values and considered it very clean data.
- One exception: all attributes were floats except for quality; changed the data type to float.

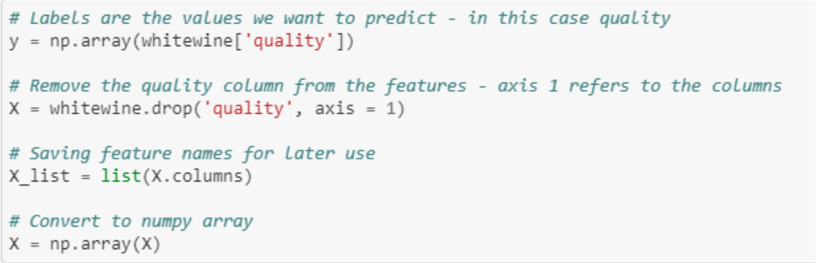
Looked at Shape of the Data

- Quality not evenly distributed amongst classifications
- Skewed heavily to quality of 5 & 6; understand that this is heavily going to affect our model

Overall Quality Breakdown of White Wine Samples

Quality	Count
3	20
4	163
5	1,457
6	2,198
7	880
8	175
9	5
Total	**4,898**

- Created x and y variables
    - output (y) = quality
    - x = other features



```
# Labels are the values we want to predict - in this case quality
y = np.array(whitewine['quality'])

# Remove the quality column from the features - axis 1 refers to the columns
X = whitewine.drop('quality', axis = 1)

# Saving feature names for later use
X_list = list(X.columns)

# Convert to numpy array
X = np.array(X)
```

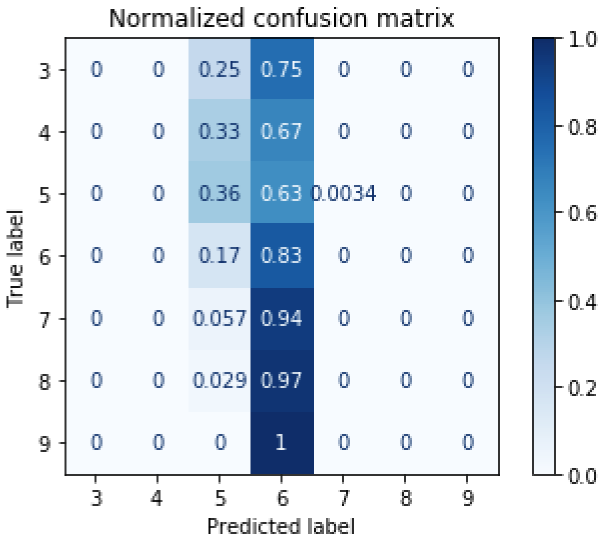
- Data split coding to determine the training and testing data (80% Training 20% Testing)
- Train test split from x and y variables
- Ran multiple models to test and compare accuracies
- LinearRegression resulted in R2 Score: 0.281361

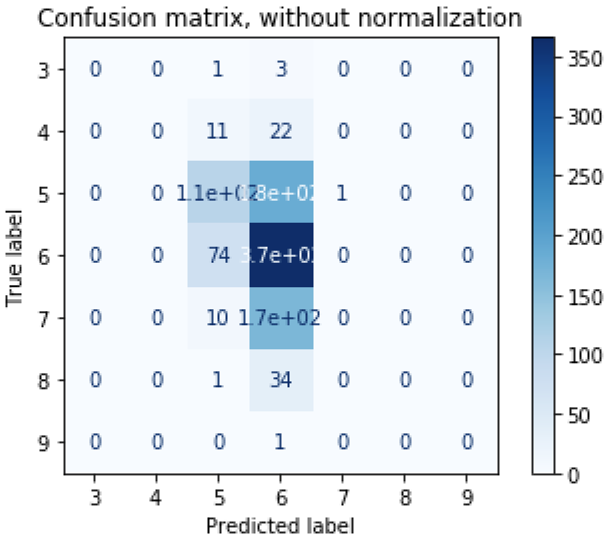
Model Results by Accuracy in Descending Order

Model	Accuracy
RandomForestClassifier	0.707142857
XGBoost	0.66122449
DecisionTreeClassifier	0.582653061
KNeighborsClassifier	0.558035714
LogisticRegression	0.481632653
NaivesBayes	0.448341837

Confusion Matrix

Ran a confusion matrix, with and without normalization, to gain insight on the inaccuracies





The results show that the model chose a quality output of 6 far too often

Data Scaling

In attempt at achieving a more standard normal distribution, we tried two different types of data scaling – Feature Scaling and Principle Component Analysis (PCA).

Unfortunately, neither helped the shape of our data enough to impact our model.

Hyperparameter Tuning

We tried two different types of hyper-tuning to improve our accuracy – RandomizedSearchCV and GridSearchCV.

Results: When the optimal parameters from those searches were applied to the model our, accuracy actually decreased.

Finalize the Model

Used pickle to save our model down into a callable (.sav) file.

FLASK APP

bc633.pythonanywhere.com

App.py

- Imported dependencies.
- Created routes to different html files using render_templates.
- Includes the home page, imbedded tableau visualizations, and site for users to submit data and receive a prediction on wine quality.

Index.html

- Introduction to wine quality project

Tableau.html

- Imbedded tableau visualizations

winePreds.html

- Displays all 11 input variables (physicochemical) used to create our model; users submit the request variable data to determine the predicted wine quality.

ADDITIONAL VISUALIZATIONS

Tableau Interactive Dashboards & Other Visualizations

https://public.tableau.com/views/WineProject

https://public.tableau.com/views/RandomForestClassifier-WineAttributeImportance

PROJECT NOTES

Future Considerations

- Include the ML model for red wine on our site.
- Reconfigure the data to have equal number of data points for each attribute (most importantly = quality) to try and make the model more accurate.
- winePreds.html – improve the format and add a text box to the app to populate the quality prediction.
- Embed additional Tableau visualization links into the live site.
- Use a different data set that includes price information. This additional information could be very valuable and increase our model's benefit to wine makers and cork dorks.