Wine Score Capstone Report

Jorge Londono

Problem Statement

Wine production is a global multi-billion dollar industry that has a history spanning over 8,000 years. With different wineries competing against each other to provide their best ingredients and mixtures, it can often be overwhelming for consumers to get a grasp of what to look for in a bottle of wine. Luckily, there are expert reviewers and sommeliers who can assign point values based on the quality of a wine. However, while we can look at specific reviews to figure out which wines perform well for these reviewers, this is a surface level into analyzing what makes wine great. What are the best predictors we can find to foresee the quality of a bottle of wine?

The benefactors of the results of this project can be several types of people. The first one is the wine producer. By seeing what type of variables result in better outcomes, the producer is able to decide how they want to run their business. Should they focus on making a specific variety of wine? Should they build a new winery in a specific province?

Next there is the consumer. This project can help bring insight into the variables that can answer, what makes wine great? This way they can make more informed decisions on what to buy, how much to value a specific wine, and if said wine is worth the price.

Lastly there is the wine industry media. If certain variables are found to show better scores with reviewers, this can inform the media of what types of new wine projects they need to spotlight and pay attention to. Should reviewers have certains standards when trying a bottle of wine, if they can estimate where that wine should rank based on the features of this project? Also, can this predictive tool highlight biases in the wine industry media?.

Data

The data that will be used for this project can be found in: https://www.kaggle.com/zynicide/wine-reviews . This data was provided by a kaggle user who scraped it from the Wine Enthusiast Magazine in 2017. It contains over 129,000 wine reviews. The variables found in this dataset, as per the contributor, are the following:

- country: The country where the wine was produced.
- description: A short description and review of the wine.
- designation: The vineyard within the winery where the grapes that made the wine are from.
- points: The number of points WineEnthusiast rated the wine on a scale of 1-100 (though they say they only post reviews for wines that score >=80).
- price: The cost for a bottle of the wine.
- province: The province or state that the wine is from
- tegion_1: The wine growing area in a province or state
- tegion_2: Sometimes there are more specific regions specified within a wine growing area (ie Rutherford inside the Napa Valley), but this value can sometimes be blank
- taster_name: the name of the review writer.
- taster_twitter_handle: the twitter username of the wine reviewer
- title: the name, year, and location where the wine came from
- variety: the type of wine the bottle is.
- winery: the winery the wine came from.

A limitation with this data set is the fact that there is no further information on certain features of the wine, such as biological composition, ingredients, how it was grown etc. Likewise, because so many of the variables are categorical. We are going to need to be very selective of which features to use in order to lessen the dimensionality of the model.

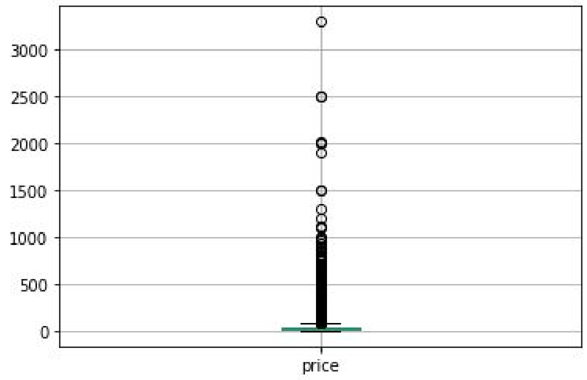
Cleaning and Wrangling

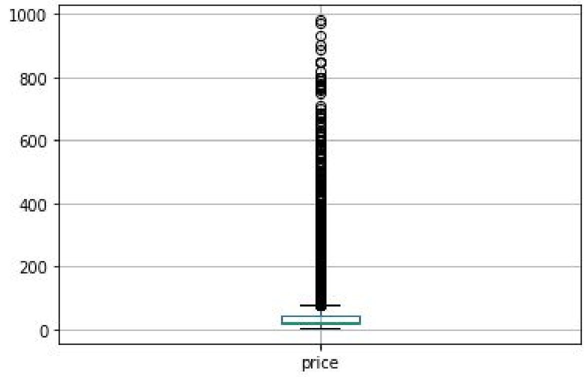
SInce the data frame obtained for this project came from a Kaggle submission, there was no substantial data wrangling or cleaning required. However the data was analyzed for certain conditions such as outliers and missing values for the numerical variables.

In order to check for outliers and missing values I first took a glimpse at the data using the ‘info’ and ‘describe’ methods. This showed that there were around 9,000 observations where the price variable was null. Additionally it showed that while the median price of the bottle of wine was only $25, the max value in that column was $3,300, a possible outlier.

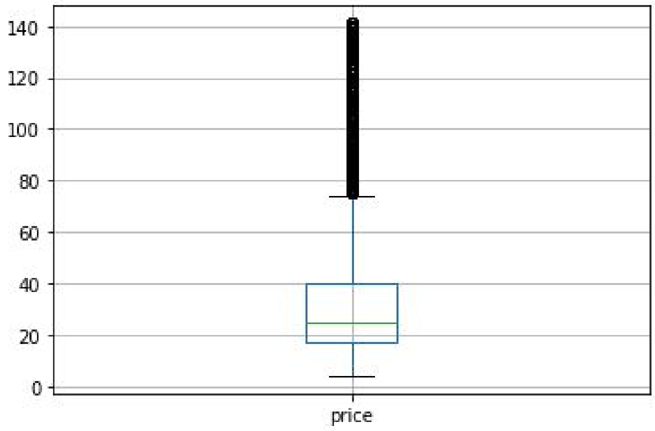
As stated previously, around 9,000 observations showed the price as ‘NaN’. Therefore in order to have an accurate predictive model, the observations with these null values were removed using the ‘dropna’ method on the data frame. To confirm that all of the null values were removed, the ‘tail’ method was used with the price column sorted using the ‘sort_values’ method. This showed that all of the null values for price were not in the dataframe anymore.

Outliers proved to be slightly more difficult to remove accurately. Creating a boxplot showed that there were a few observations that had a price higher than $1,000. Therefore, using the ‘drop’ method for the data frame, all of the observations with wine over $1,000 were removed. The total observations removed was 16.



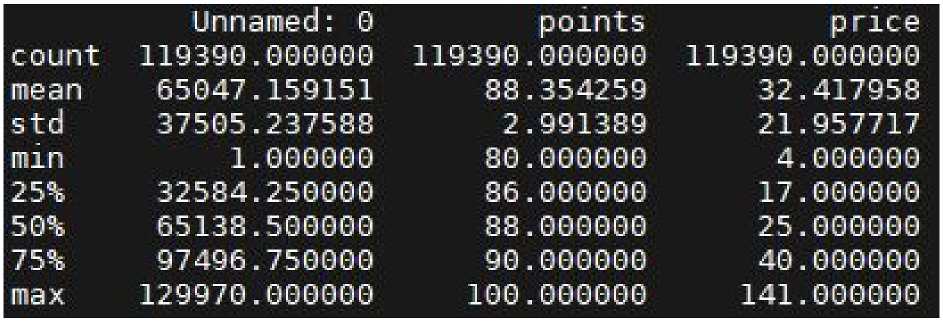


However, this was not enough to correctly remove the skewness of the boxplot. Therefore, I performed a Z-test using the scipy method ‘zscore’ to find and remove all of the observations that were above a Z-score of 3.   This left me with the boxplot below, which looks like a better set of data to use for this project.



Exploratory Data Analysis

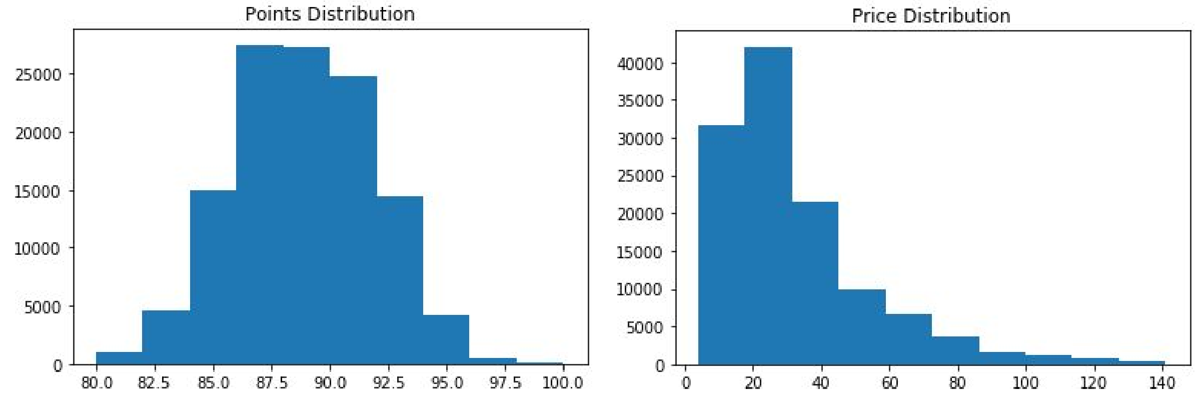
Descriptive statistics for this data set can be found below.



```
        Unnamed: 0           points            price
count  119390.000000  119390.000000  119390.000000
mean    65047.159151      88.354259      32.417958
std     37505.237588       2.991389      21.957717
min         1.000000      80.000000       4.000000
25%     32584.250000      86.000000      17.000000
50%     65138.500000      88.000000      25.000000
75%     97496.750000      90.000000      40.000000
max    129970.000000     100.000000     141.000000
```

While the mean and median values of the 'points' column are very similar, at 88.35 and 88.0 points respectively, the mean and median values of the 'price' column seem to show a larger difference, at 32.42 and 25.0 respectively, which could mean that our right skew could still be significant.Additionally, the minimum and max values for the price column show a greater spread of values than that of the points column.

With this in mind, it would be a good idea to visualize what this data looks like to further understand how this project should proceed.

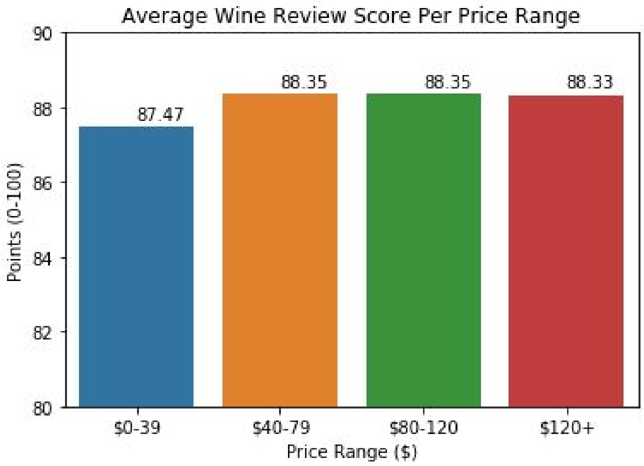


From the graphs above, we can see that the points column of our data frame appears to be normally distributed. However, the price column of our data set seems to be right skewed and might need to be transformed via a logarithmic function.

Statistical Analysis

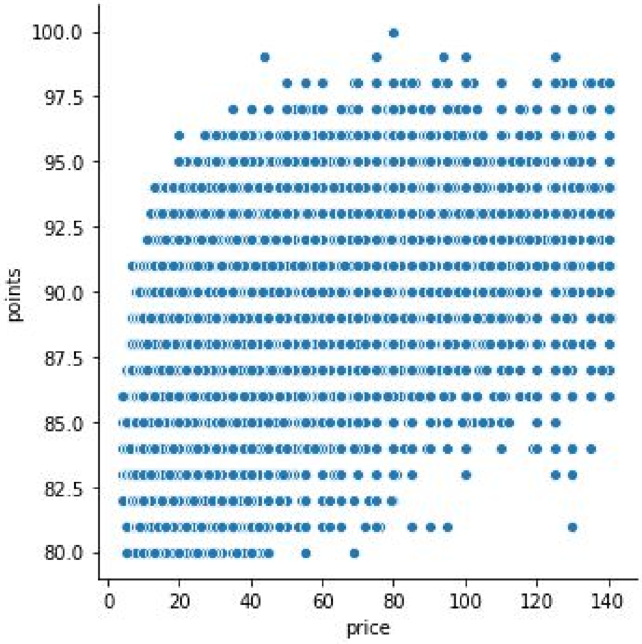
While this project will include variables such as: Country of origin, province of origin, wine variety, and winery, in its model, the main feature appears to be the price of the wine. This statistical analysis will aim to uncover if there is a statistical significance in the difference between the least expensive wines and the most expensive wines when it comes to review score.

The first step is to create subgroups based on price in order to compare their mean score. Creating a bar plot will help us visualize the difference in average score between these subgroups. There seems to be a difference in the mean score from the lowest priced group to the highest priced group.



There seems to be a difference in the mean score from the lowest priced group to the highest priced group.

Next we can try to find a correlation between price and review score.

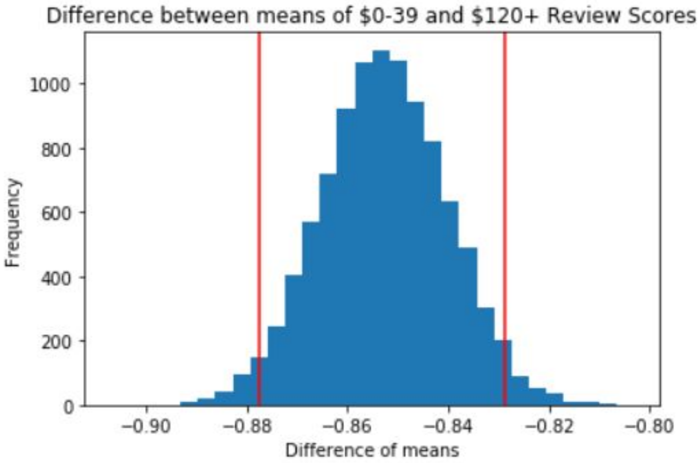


Creating a plot chart helps us visualise the existence of a relationship between 2 different variables. In this case, while we can see that there appears to be a positive correlation, it is not clear how strong it is.

Using the numpy method ‘corrcoef’ we can find out what the correlation between these variables is. At 0.55 there appears to be a moderate positive correlation between the score received and the price of the wine.

While a good hypothesis would be that the price of the wine affects how highly it is rated, a bootstrapping test will show us if the data is not giving us these numbers by chance.

At this point we have identified 2 different things. Our data shows that there is a positive correlation between how well a wine scores and how much the wine is priced at. Additionally, that the mean score of wine under $40 is lower than the mean score of wine $120 and over. A bootstrapping hypothesis test will help us replicate the data 10000 times to see if the difference between group 1 and group 4 exists not purely by chance. By using a Bootstrapping method we will test the following null hypothesis: with 95% confidence we can state that wine under priced $40 has the same mean score as wine priced $120 and over. The alternative hypothesis would be that the mean score of wine priced under $40 is not the same as the mean score of wine priced at $120 and over.



A two tailed test with a 95% confidence interval shows that the difference between the means of the two groups is between -0.88 and -0.82.

Because this interval does not include 0, at 95% we can reject the null hypothesis that the wine priced under $40 has a mean review score equal to that of wine priced at $120 and over.

Algorithm Selection

This project revolves around using different features to predict a continuous variable (points), therefore this is a regression problem. Now we have to identify what type of regression algorithm we should use for this project. Since there is a very large number of categorical variables, the best regression algorithm should be either a Lasso regression or a RIdge regression. Categorical variables will have to be transformed into dummy variables which will be extending the dimensionality of the algorithm. These 2 algorithms work really well with high dimensionality which is perfect for this data set.

Steps taken

To begin the machine learning process, the features and the target variable have to be separated in order to test out the models. To do this we take the columns for all of the features and turn them into an array using the built in Python method values(). We also do the same thing to the target variable because the functions we are going to use require them to be in this format. Since one of my main features (price) is not normally distributed, I will test out how the model looks with both this feature and the target variable passed through a log function.

Next I will be testing out different features and see how they perform and which algorithm gives me the best results. Every test should follow a very similar set of procedures. First I create a test set and training set using sklearn's train_test_split function with the random seed set to 42 in order to be able to reproduce the results and better compare the results. Next I use the sklearn functions Ridge() and Lasso() to create our models. I will use the training set to fit the model, then I will use the test test to predict the outcomes. Then I print out the score (which in this case it's the R2) and I use this to compare the results between all of my experiments in order to create the best model.

Parameter Tuning

Once I select the best performing algorithms, in this case both Ridge regression and Lasso regression, I need to tune the parameters in order to optimize its performance. I was looking at mainly two different parameters to change, normalize and alpha. Since normalize can only be either true or false, it was very easy to test. The alpha parameter on the other hand needs more work since it is a continuous number that can be set. By performing a Grid Search CV, I was able to find the best alpha parameters for both Lasso (alpha = 0.00001) and Ridge (alpha = 1). This was done using the *best_params_* attribute which comes from the *GridSearchCV* function.

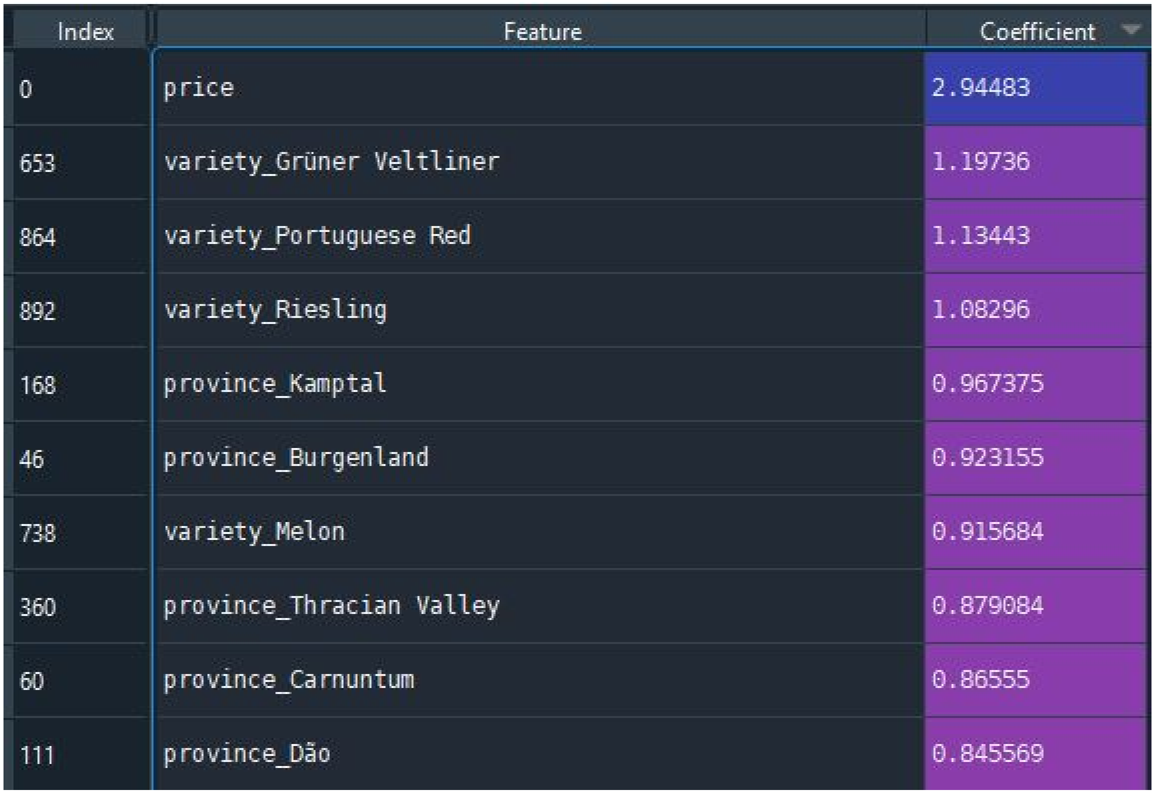
Results

The results of my experiments can be found below. Each point represents a different trial with combinations of features and parameter optimization.

Model	**R^2**
Algorithm: Ridge Features:Price Parameters: Default	0.304
Algorithm: Ridge Features: Price(Log) Parameters: Default	0.366
Algorithm: Ridge Features: Price(Log), Country Parameters: Default	0.383
Algorithm: Ridge Features: Price(Log), Province Parameters: Default	0.415
Algorithm: Ridge Features: Price(Log), Province, Variety Parameters: Default	0.430
Algorithm: Lasso Features: Price, Province, Variety Parameters:normalize =True, alpha =0.00001	0.430

The last two models resulted in a score that is pretty much the same. Therefore, we can calculate the Root Mean Squared Error of each of the models and find out which of these performed the best. The Lasso Regression model resulted in a Root Mean Squared Error of 2.265, and the Ridge Regression model resulted in a Root Mean Squared Error of 2.270, a slightly worse result. Thus, the optimal model is the 6th one in the table, the Lasso regression model.

Below are the top 10 features of this model which display the highest coefficient values:



Index	Feature	Coefficient
0	price	2.94483
653	variety_Grüner Veltliner	1.19736
864	variety_Portuguese Red	1.13443
892	variety_Riesling	1.08296
168	province_Kamptal	0.967375
46	province_Burgenland	0.923155
738	variety_Melon	0.915684
360	province_Thracian Valley	0.879084
60	province_Carnuntum	0.86555
111	province_Dão	0.845569

Conclusion

For this project, we took the various features of the wine data set and found the best combination of regression algorithm, variables and hyperparameters to create a model that can best predict how well a particular wine will score. Using a Lasso regression with the price, the province where it came from, and the variety as its features, as well as the alpha set to 0.00001 and normalized, we were able to get the best R^2 and Root Mean Squared Error.

While the results show a relatively low Root Mean Squared Error, this model can be improved with better domain knowledge. For example, a limitation I had was that I don’t really know how to differentiate between varieties in wine. If the data set could be modified to group wine varieties together by their type (e.g. white wine, red wine, etc.) better than the very large number of different categories in the current wine variety variable, we could see better results as well as lower dimensionality. Additionally, by having something such as sales data could greatly improve the model because there can be a correlation between how well something scores to how well it sells. Furthermore, some categorical variables were not able to be used because of computational limits. With much better computer hardware, there can be more opportunities to include and experiment with features that contain a high number of categories.

Recommendations

With this model, there are a few recommendations that can be made to wine producers in order for them to score well with their wines. The first one is to notice that the price of wine is a good predictor of wine scoring, therefore making sure that releasing a bottle of wine with a higher than average price will in theory bear a higher review score. Likewise, wine producers should note that the top 3 wine varieties based on their coefficient values are: Grunet Vetliner, Portuguese Red, and Riesling. Additionally, wine from Kamptal will score better.

Further Study

With the findings in this project, we can further try to expand on it or ask other questions. As previously mentioned, we can use sales data (if available) to not only improve our regression model, but maybe use this as the target variable and the past features to predict the overall sales of a bottle of wine.

Another area of study that can stem from this project is using the different ingredients that compose the wine in order to better predict and understand how well the wine bottle would score. This type of analysis would have to deal with more agricultural and biological domain knowledge but it can be highly insightful in multiple aspects. For example, there might be certain ingredients or chemical components that can be isolated in order to make better tasting wine. Likewise, the types of plants and the way they are grown can be tested to see if they do make a difference and by how much. Overall, these insights might be able to be applied to agriculture as a whole, and we can learn much more about this field with the right type of data and algorithm