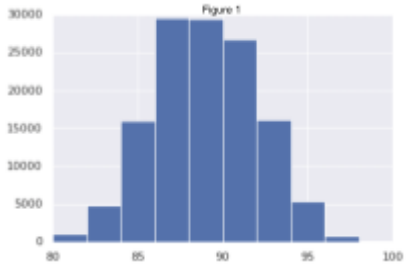
Predictive Analysis on Wine

Alcohol makes the world go round, or that's at least what I think, and coincidently an obscure rock band named the Methadones. This intoxicating liquid is the social lubricant for societies all over the world, in a wide variety of contexts, from casual get-togethers to formal galas. In college life, alcohol has become so widely used, I dare say that it is second only to water consumption, and in this regard, Lisa and I can definitely attest to this. It is for these reasons that we have decided to create a model on how different people rate alcohol, specifically wine.

In our assignment, we performed a predictive task on a wine dataset to predict its rating based on categorical and numerical variables. Our dataset, "Wine Reviews" taken from Kaggle contains around 130,000 samples of wine, with the following variables: country, title, description, designation, number of points rated by Wine Enthusiasts, price, province, variety, region, and taster name. The data types for these variables are distinguished by the nominal: designation, province, region, title, description, variety, winery, and numerical: points, price.

	points	title	description	price	designation	variety	region_1	province	country	winery
0	87	Nicosia 2013 Vulkà Bianco (Etna)	Aromas include tropical fruit, broom, brimston...	87.0	Vulkà Bianco	White Blend	Etna	Sicily & Sardinia	Italy	Nicosia
1	87	Quinta dos Avidagos 2011 Avidagos Red (Douro)	This is ripe and fruity, a wine that is smooth...	87.0	Avidagos	Portuguese Red	None	Douro	Portugal	Quinta dos Avidagos
2	87	Rainstorm 2013 Pinot Gris (Willamette Valley)	Tart and snappy, the flavors of lime flesh and...	87.0	None	Pinot Gris	Willamette Valley	Oregon	US	Rainstorm
3	87	St. Julian 2013 Reserve Late Harvest Riesling ...	Pineapple rind, lemon pith and orange blossom ...	87.0	Reserve Late Harvest	Riesling	Lake Michigan Shore	Michigan	US	St. Julian
4	87	Sweet Cheeks 2012 Vintner's Reserve Wild Child...	Much like the regular bottling from 2012, this...	87.0	Vintner's Reserve Wild Child Block	Pinot Noir	Willamette Valley	Oregon	US	Sweet Cheeks
...	...	...	...	...	...	...	...	...	...	...



To begin with our exploratory analysis, we took a look at the variable "points" and its distribution of each wine entry shown in Figure 1. Creating a histogram, we found that the distribution of the variable "points" created a bell-curve, indicating a normal distribution within the variable. Thus, we decided it would be a great variable to predict based on its shape, ranging from 80 to 100, with most points averaging around 88.

Figure 1.1

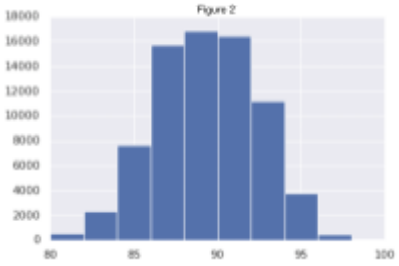
```
dfdata.describe()
```

	points
count	129971.000000
mean	88.447138
std	3.039730
min	80.000000
25%	86.000000
50%	88.000000
75%	91.000000
max	100.000000

Figure 1.1 shows the summary statistics. 80 is the minimum number of points because as stated from our source, 80 is considered 'good quality', and anything below is not included in the wine samples. Calculating the standard deviation, the output was 3.0397, a value that we used as a baseline for comparison once we removed faulty values from our dataset.

Once we established some baselines statistics within the predictive variable we moved on to data cleaning, so as to facilitate useability of the code. This includes converting values in the dataset to more appropriate types, such as changing points and price to integers and floats respectively. Additionally we removed irrelevant variables in our data such as "taster_name" and "taster_twitter_handle." Lastly we decided to remove "region_2" for its overwhelming prevalence of missing values within the variable, this was done to preserve the dataset size and achieve good quality predictions.

Once we removed these irrelevant variables, we then removed any null values, so as to not hinder us in our programing. Its histogram is shown in Figure 2. The entirety of this data cleaning/filtration step left us with a respectively large dataset of over 70,000 values.



The resulting standard deviation after removing None values was 3.0481, from a previous score of 3.0397, a difference of .0111. This difference in the std could be considered very minor and therefore does not represent a significant change in variation between the predictive variable "points" in our data set.

Moreover, this finding also supports our data cleaning process by indicating that these data points removed did not have any significant correlation between their own missingness and the variable “points.”

This kind of missingness can be classified as missing completely at random, which is a type of missingness that most resembles truly random data and therefore can consequently be removed without any significant change in the distribution or change to variance within our dataset.

Figure 3

US	54504
France	22093
Italy	19540
Spain	6645
Portugal	5691
Chile	4472
Argentina	3800
Austria	3345
Australia	2329
Germany	2165
New Zealand	1419
South Africa	1401
Israel	505
Greece	466
Canada	257
Hungary	146
Bulgaria	141
Romania	120
Uruguay	109

Country is a variable of interest, so we used the value_counts() function, and as we see in Figure 3, found that out of 43 countries in all the samples, many are chrome US, and most European countries. With the variety, the country variable can provide useful information in predicting points.

As for our variety variable, out of 707 unique varieties, we found in Figure 4, that Pinot Noir is the most common, following Chardonnay. From this information, popular wine types may tend to have better reviews with higher points. For the more rare wine types where its value count is only 1, variety itself may not help in predicting results of a wine review.

Figure 4

Pinot Noir	13272
Chardonnay	11753
Cabernet Sauvignon	9472
Red Blend	8946
Bordeaux-style Red Blend	6915
...	
Sauvignonasse	1
Cabernet Sauvignon-Barbera	1
Meseguera	1
Forcallà	1
Bobal-Cabernet Sauvignon	1

Name: variety, Length: 708, dtype: int64

Taking a look at the title and description variable from our dataset after removing the missing values, we noticed that the text for title of wines often includes the year. Accounting for the description for each sample, we evaluated 60,029 unique words, and 29,364 unique words for titles.

Given that we want to predict the points for each wine sample, we first incorporated all the variables. For our nominal variables, designation, variety, region, province, country, and winery, we implemented one-hot encoding. We converted the nominal data that represent categories into feature vectors. This way, we can capture a variety of categories.

With the title and description variables, we figured text mining would be predictive of points for each wine sample. Since punctuations, specifically periods and commas do not connote positive nor negative words, we removed them for text analysis. To begin with, the titles for wine have fewer words, so we incorporated ngrams. We chose creating bigrams over unigrams or trigrams in our model because unigrams on their own may not capture potential negations. Trigrams would not be the best either, since we have a fixed-length representation of the most common words.

As a feature vector, we considered performing sentiment analysis. Sentiment analysis would be informative in predicting points wine tasters give to each sample of wine. By adding values to more positive and negative words or sets of words, we can extract meaning. The dictionary size used was 1000, meaning we take the 1000 most popular bigrams. This is reasonably sized as the greater the value, more precise our results be, yet will take more time executing. When extracting the most positive and negative bigrams, the popular bigrams included words such as “wine”, “winemaker” and “years.” The results seem to not be predictive of points, as the popular words for the title are only descriptive of wine, instead of a type of wine.

Although the words included in the title for our dataset did not add value to our predictive task, incorporating bigrams for the description variable would be more effective with a larger variety of words. After trying to add bigrams for description in our feature, the bigrams’ most positive bigrams resulted in a combination of stop words, for example, ‘this is’, and ‘to the’, shown in Figure 5.1. The most negative bigrams included some like ‘a big’, ‘a blend’, ‘a bit’, shown in Figure 5.2. Unfortunately, the combination of those bigrams apply to almost any and all wine

Figure 5.1

```
'drink now.'),
'rich and'),
'the fruit'),
'this is'),
'to the')]
```

Figure 5.2

```
'a big'),
'a bit'),
'a blend'),
```

samples with little significance in our predictive task.

These are stopwords that convey little information. There are reasons for removing stopwords. Firstly, because on its own does not add value to our model, and removing them can greatly reduce corpus size. Stopwords also make it harder to discover which features are more important. However, in our predictive task we stuck with not removing stopwords because we wanted to perform sentiment analysis with bigrams, and stopwords often change the meaning of words.

With our description variable, besides considering ngrams, we also incorporated transforming our text into feature vectors with TF-IDF (term frequency & document frequency). This could improve our model and help with predicting points for each wine sample. If the TF-IDF is high, the particular word appears more frequently in the given document, and if TF-IDF is low, the word either appears often in many documents, or less in the specific document. We found it was most efficient to use the TF-IDF vectorizer from sklearn to transform our data.

In setting up our model, we first split our dataset into train and test sets by 80% of the data as our train, and the rest 20% of data as the test set. By splitting data in this way, we can evaluate if our model is overfitting. Then, we chose to use linear regression for our model because we are predicting points, a numerical data type using nominal variables.

Executing our model altogether, we used the ridge function on our linear model to predict points. The ridge function allowed us to change the regularization strength, where we tried fitting to regularization strengths of 1.0, 0.1, and 10.0. The ridge function is an ideal model to fit the function on the training set, and as also aiding additionally in the issue of overfitting. Regularization allows for penalizing model complexity during training. We ended up using the regularization strength of .10 in our final model as it prove to have the best MSE score when calculating a metric of performance, beating both 1 regularization and 10.

Our method of evaluating our model's success was by calculating the MSE (mean squared-error). Linear regression was the best in predicting points because we wanted results of points for each wine sample. MSE, the estimator, ideally is proportional to the variance of the data, calculated by taking the average squared difference between our predictions and the actual value. Although we want the MSE to be as small as possible, we also want to avoid overfitting, when our model fits exactly with the training set. If we overfit, our model will not be effective on unseen data. This is where the Ridge function alongside linear regression becomes useful.

For our categorical features, we used one-hot encoding into our feature vectors. Because we had more than two values for our nominal variables, applying one-hot encoding is most effective to capture every country, designation, variety, and region. We did not consider classification models like Naive Bayes, SVMs, and logistic regression, given the obvious, that we are predicting points and they are not labels.

To optimize our model, because we used linear regression, we considered gradient descent. It is an optimization algorithm that fits our parameters to minimize our cost function as much as possible. After considering gradient descent, we found that it was not going to help with scalability in our prediction. Our dataset is large, with over 70,000 wine samples, therefore, performing computation for our large dataset can be very slow.

This wine dataset retrieved from Kaggle has been used to study and predict whether wine will be bought. Similar to our predictive task, the points predict how good a wine sample is, which can indicate a customer choosing to buy the particular wine. One study done in the past was taking a look at the description variable to gain knowledge on whether the wine will be bought. Our finding was similar to the study, as they stated the description is done from a marketing perspective, describing the wine without expressing a sentiment. Thus, it is not a useful feature in the model.

At the start of our exploratory analysis, we performed an analysis on missing values by

finding the standard deviation of points for the wine samples before and after removing values. Another study began their analysis doing the same. Their finding found a high fraction of missing values in price, then region. We removed missing values for the region variable that goes parallel to the study done, both finding that irrelevant features should be removed. For natural language processing for description, they incorporated unigrams instead of bigrams. Comparing their unigrams to our bigrams, the results had the same effect where the most popular words included ‘wine’ or ‘this’, that all apply to every wine sample. They used a different method in measuring TF-IDF, but ultimately concluded that it was important in the analysis.

A similar study was conducted, their focus was recommending to wine reviewers. That study implemented collaborative filtering and finding similarity between feature vectors of items. Their exploratory analysis considered non-null values, which we also did in filtering out None values. When conducting content-based recommendations, we also performed the model with a matrix setup. Additionally, this literature incorporated the TF-IDF vectorizer, and the difference from our model was that they explicitly used stopwords in their vectorizer. Combining methods was useful in their recommendation, and we found the same, that the dataset requires analysis of multiple variables for predicting points.

One flaw in our dataset that was not specified and questioned in other literature was that we do not know if wine tasters for each wine are given information on the price of each wine. Lack of this information is crucial, as often wine tasters may be influenced by price to rate and provide points. Furthermore, literature was done on the dataset to evaluate geography of each sample, our focus considered the variables country and region simply as categorical variables.

Overall, the state-of-the-art methods used in the wine dataset is a combination of methods to predict points for each wine. Analysis done in our predictive task and fitting the model, by splitting nominal variables into categories, and evaluating text from the titles and descriptions was commonly done in all studies done.

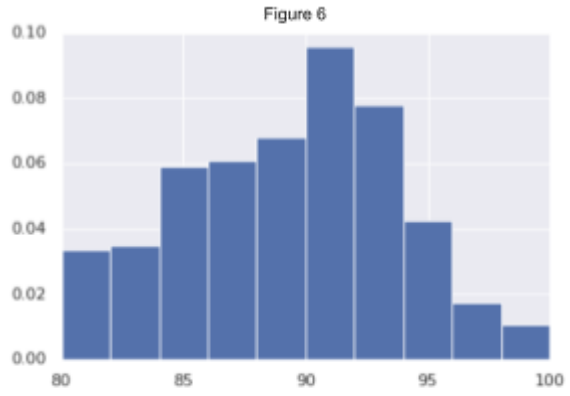


Figure 6 displays our predictions in the range [80, 100] with our lower and upper bounds confidence interval. These predictions produced an MSE of 21.789. Based on the figure, we know that our results are reasonable given that the distribution is similar to our original dataset’s points, with a slightly different skew. After calculating the confidence interval, which measures the degree of uncertainty in our method, we realize our predictions are just slightly higher than the points given in the dataset. This shows the significance of our results is within reason, with reference to our lower and upper bounds.

Based on our model performance, we noticed that some features were successful, and some failed. We tried implementing unigram and bigram models for the text of the titles. The text of the wine titles did not help to predict whether a wine taster would give higher or lower points. The titles were unique to each wine sample, thus they added just as much value as the country or variety variables did in our model. The feature of performing ngrams, specifically bigrams for the description failed when executing our model. Although bigrams may seem to be crucial in predicting the points, our prediction results from just the bigram feature failed. Even with the correct implementation, the results were not what we expected, potentially because of the points being of a small range of numbers between 80 to 100, majority equal to the average. In addition, extracting the bigrams to

get results from sentiment analysis, we found that by looking at the selected bigrams, the feature does not capture certain qualities of the wine samples, only generic words.

We noted that some removal of features improved our MSE score, the designation, winery and region. This can be explained by the very large feature vectors they created when constructing the feature matrix. These three variables constituted 40,456 of the 41001 heavily diluting the feature matrix with their one hot encoded inputs. Therefore it could be the case that even though these 3 variables were helpful in creating a predictive model on their own, they dilute the weight of the other features within the feature matrix.

In conclusion, we found that each wine sample's variety, province, and country were the most useful variables to include in our feature and implement in the model. Implementing linear regression with regularization with the Ridge model, and minor adjustments allowed us to reach the best MSE in predicting wine ratings.