Exploring Wine

Zhuoxi Zeng[1], Pin Tian[2], Nan Jiang[3] and Xinxin Chen[4]

I. INTRODUCTION

In a culture of consumption and instant gratification, wine has an incredible ability to provide a range stimulations. There are many different wines that appeal to a great many people. Like various forms of music and art, there is something for everybody. This also makes the judging of wine a highly subjective matter. It's also a challenge to describe - the flavors are sometimes just hints at something you can put your finger on. All the above reasons have made wine reviews really helpful for people to choose their favorite wines. Given scores and text review given by wine tasters, we hope to analyze wine in some aspects that we are interested in.

In this project, we use the dataset downloaded from Wine Reviews to predict the point rating and their variability based on the review information given by wine enthusiast. First, we analyze data to pick up useful features for our specific prediction task and throw away invalid data. We then proceed with the task of score prediction, variety prediction and sentiment analysis. In the project, we used state-of-art method XGBoost, which is an open-source software library which provides the gradient boosting framework. We have also reviewed some other literature and other people's method at the end.

II. Related Works

This data was rarely used for prediction tasks. Therefore it was hard to find other work done using this data for sentiment analysis. Also, unfortunately it seems like there is not a lot of research into the field of wine quality and variety prediction.

We did found some work from Kaggle that tries to predict the variety of wine and quality of wine using the dataset[3] We found a kernel that uses TF-IDF to select features and gradient boosting decision tree to predict the variety of wine. The way he did it was to include only reviews of 4 classes of wine which meant his data size was only a fraction of the total data size and in that data he was able to achieve an accuracy of 87%. However, we feel like by excluding most of the data, his method will not be able to generalize well over the whole dataset. Also, we feel like this method is lacking in variety in terms of model since he has only used one, this meant it was hard for us to see if gradient boost decision tree was truly the best model to use.

For predicting quality of wine, again from kaggle we found someone who has tried to attempt this[4]. His method was to use polarity and length of the description to predict if the wine is good or bad. In order to do this, he divided the data into good and bad wine according to the points each wine scored. He then apply a wide range of models onto this dataset to predict whether the wine is good or bad. In the end, his result was quite good with the best model being random forest with an area under the curve of region of convergence of 0.82. However, an accuracy is not giving. Although this method has achieve a great AUC, we believe this is only saying that his classifier is quite good and does not return a trivial result, we do not agree with how he has processed the data and his model evaluation. By creating a new feature quality by binarizing points in order to make this into a classification task we feel is not accurate. First of all since the whole dataset is of wines that is above 80, it was hard to determine where to make a cut for bad or good wine. We feel like it is much better to treat points as quality and predict points while using Mean Squared Error as metrics. This way our method will not lose the fineness in wine scoring as well as not making assumptions on where the points cut off is for good and bad wine.

For sentiment analysis, indeed, we found a work on

*This work was not supported by any organization

[1]Zhuoxi Zeng is with the Department of Electrical and Computer Engineering, University of California, San Diego z4zeng@eng.ucsd.edu

[2]Pin Tian is with the Department of Electrical and Computer Engineering, University of California, San Diego pitian@eng.ucsd.edu

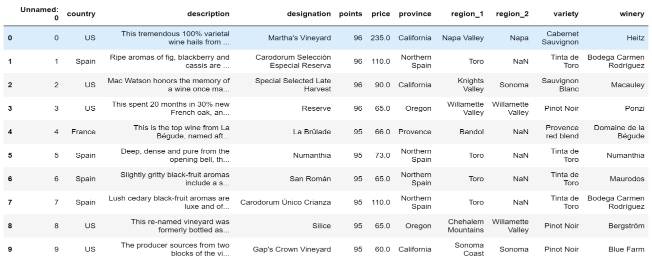
[3]Nan Jiang is with the Department of Electrical and Computer Engineering, University of California, San Diego n2jiang@eng.ucsd.edu

[4]Xinxin Chen is with the Department of Electrical and Computer Engineering, University of California, San Diego xic045@eng.ucsd.edu

finding informative words [2]. But our work is overall better than their work. The method used to classify the ”bad” ”good” wines is different. They used the lowest 25% and the highest 25% points description of wines (50% total), we used all description and classified based on mean value of all points instead. In their work, Naive Bayes model used to classify which has bad accuracy. Instead, three model including Naive Bayes were used in our work and we found that the logistic regression has much better classify accuracy than Naive Bayes. Moreover, we analyzed not only unigrams, but also bigrams so that we can explore more informative phrase related to wines. Above all, our sentiment analysis provides more accurate and useful information.

III. Data Analysis

The total number of samples in the dataset is 150,930 reviews of wine written by sommeliers. The data consists of 9 features. A preview of the first 10 samples is shown in figure 1.



	Unnamed: 0	country	description	designation	points	price	province	region_1	region_2	variety	winery
0	0	US	This tremendous 100% varietal wine hails from ...	Martha's Vineyard	96	235.0	California	Napa Valley	Napa	Cabernet Sauvignon	Heitz
1	1	Spain	Ripe aromas of fig, blackberry and cassis are ...	Carodorum Selección Especial Reserva	96	110.0	Northern Spain	Toro	NaN	Tinta de Toro	Bodega Carmen Rodríguez
2	2	US	Mac Watson honors the memory of a wine once ma...	Special Selected Late Harvest	96	90.0	California	Knights Valley	Sonoma	Sauvignon Blanc	Macauley
3	3	US	This spent 20 months in 30% new French oak, an...	Reserve	96	65.0	Oregon	Willamette Valley	Willamette Valley	Pinot Noir	Ponzi
4	4	France	This is the top wine from La Bégude, named aft...	La Brûlade	95	66.0	Provence	Bandol	NaN	Provence red blend	Domaine de la Bégude
5	5	Spain	Deep, dense and pure from the opening bell, th...	Numanthia	95	73.0	Northern Spain	Toro	NaN	Tinta de Toro	Numanthia
6	6	Spain	Slightly gritty black-fruit aromas include a s...	San Román	95	65.0	Northern Spain	Toro	NaN	Tinta de Toro	Maurodos
7	7	Spain	Lush cedary black-fruit aromas are luxe and of...	Carodorum Único Crianza	95	110.0	Northern Spain	Toro	NaN	Tinta de Toro	Bodega Carmen Rodríguez
8	8	US	This re-named vineyard was formerly bottled as...	Silice	95	65.0	Oregon	Chehalem Mountains	Willamette Valley	Pinot Noir	Bergström
9	9	US	The producer sources from two blocks of the vi...	Gap's Crown Vineyard	95	60.0	California	Sonoma Coast	Sonoma	Pinot Noir	Blue Farm

Fig. 1: First 10 samples from data

A description of each column can be seen in the table below.

Feature	Description
Price	Price of wine
region_1	the wine growing area in a province or state
Country	the country that the wine is from
Province	the province or state that the wine is from
region_2	specified within a wine growing area
Variety	the type of grapes used to make the wine
Winery	the winery that made the wine
points	Score given to wine by sommeliers
Designation	the vineyard where the grapes are from

TABLE I: Feature Description

Although there are over 150,000 reviews, these views are written only by 19 sommeliers in total. Also the data here only include wine that score over 80 points.

First thing we were interested was to see how much wine each country produces. This is shown in figure 2. As we can see from the figure, wine production is concentrated in only a handful of countries, this makes sense as we did some research and found the grapes required to make good wine can only be cultivated in specific conditions. Hence resulting in only a few countries having the ability to make good wine. Also, from all the wine producing countries from this data set, we see that US makes the most wine. This may be because this data was mined from a US website

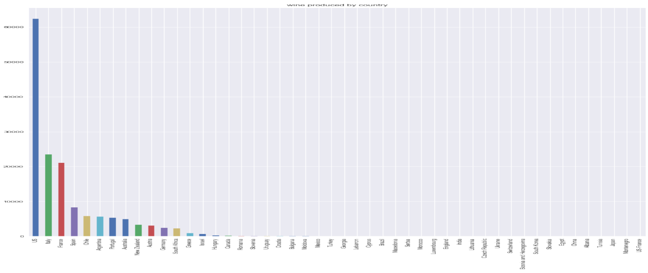


Fig. 2: First 10 samples from data

To see which features we could use in predicting quality of wine and variety, we plotted the box plot of each feature against quality to see if we could find some informative features for our prediction task. These plots are shown in figure 4. For these box plot, we have only chosen the most popular categories for visibility however this should not hinder the conclusion we get from these because the categories are chosen such that it covers most of the samples. Also, a box plot for price against quality was not drawn because price is a continuous feature.

Instead, we plotted a scatter plot for price and points to see if there are any obvious clusters that indicate which price results in good quality wine. Interestingly, it seems like even cheap wine can be scored highly by sommeliers but in general we do not see a distinctive hyperplane to say that above this price, the wine would be good and below it it will be bad. As a side note, just from this plot, when purchasing wine, the more expansive it is the most likely it is for it to be good.

Unfortunately, from what we can see from these plots these features available to us immediately are not very informative because for all these distributions are overlapping each other which makes them non-distinctive. This means that if we were to use these features in a classifier, the classifier would not perform well in our prediction task. This means that we would need to process the data somehow to make it more informative.

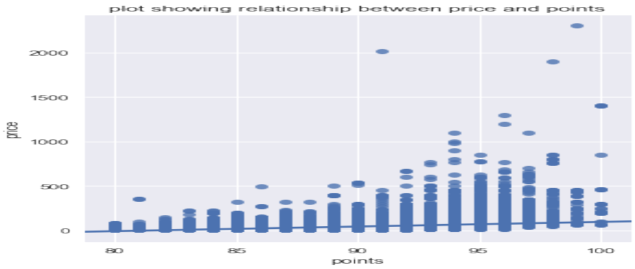


Fig. 3: scatter plot showing the relationship between price and points

IV. Data Preprocessing

A short description is also available to us which gave us the idea to construct us feature set from these descriptions using text mining techniques. In class, a text mining technique called TF-IDF was recommended, upon some research, we believe we could use TF-IDF to construct our feature for our classifiers.

TF-IDF stands for term frequency - inverse document frequency. As suggested by its form, it's a two part process. First, it see how often a word appears in the current document, then this number is divided by how many documents this word has appeared in. So a high TF-IDF score means that the word appears very often in its document but does not appear frequently in other documents. A word with a high TF-IDF can then be used as a feature machine learning tasks. Figure 5 shows top 200 TF-IDF score and the corresponding words. Since there are so many words, it was difficult to include words in the plot so instead we will represent them using numbers and put the words in a table. As we can see from the figure, there are quite a few of words that scores highly in TF-IDF that we could use as our feature. Also notice that at around 200 words, the TF-IDF scores starts to drop rapidly. Therefore we will be using the frequency of the words that scored top 200 in TF-IDF as our feature for predictive task.

This concludes the general data preprocessing, for each task there was minor specific data preprocessing that will be mentioned in their respective sections.

V. Model Selection

For the predictive models we will be using logistic regression to achieve a baseline result for both predictive tasks. For this assignment, we have decided to use decision tree based models, namely random forest and extreme gradient boosted decision tree. The reason why we did this is because we are not sure if our data is linear or not and both random forest and gradient boosted trees handle co-linearity much better than other models.

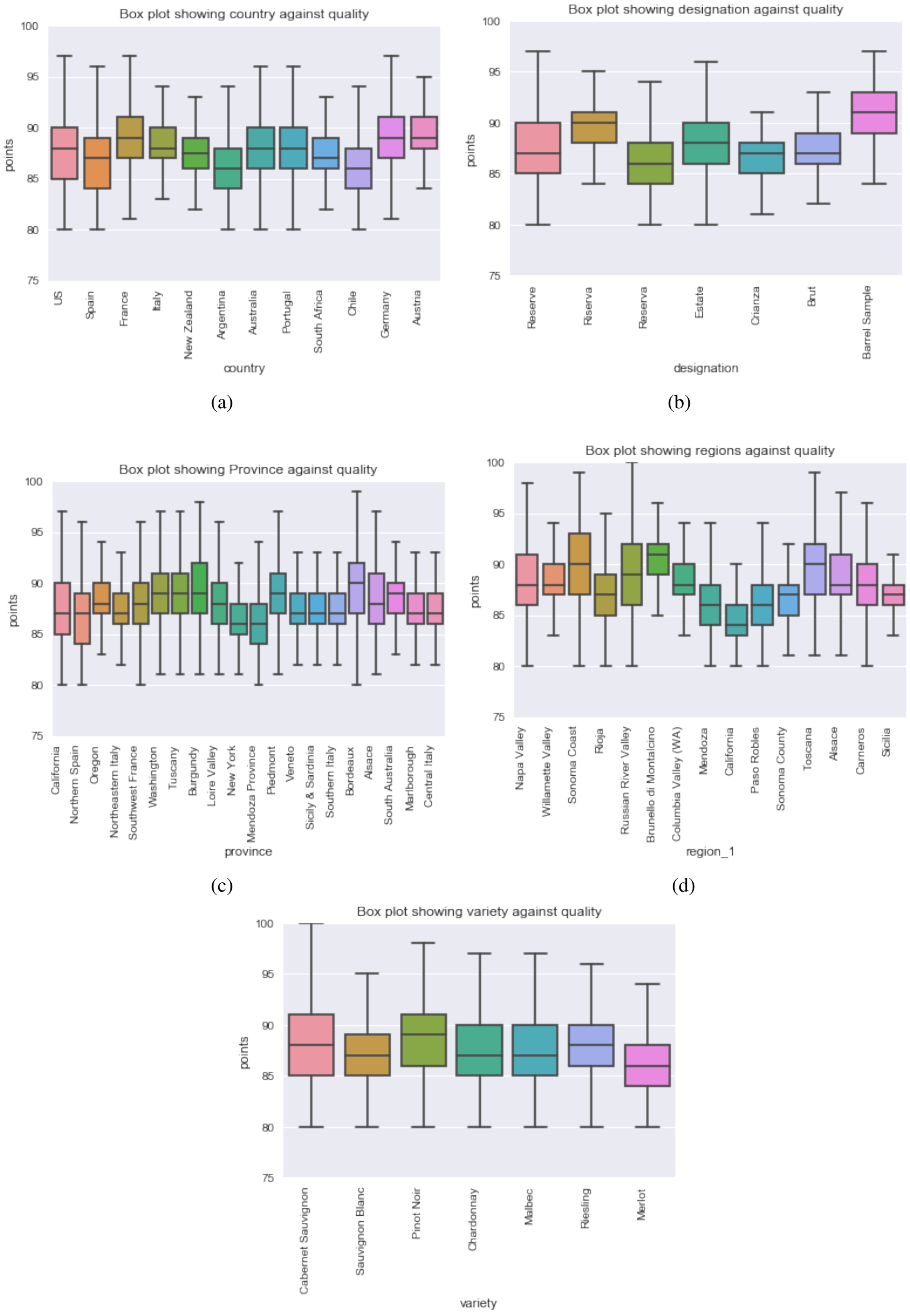
Some other nonlinear models that we have considered are Support Vector Machines and K-Nearest Neighbor. However, in the end we have decided to not use these models because we think we have too much data. Using TF-IDF to identify 200 important words as our feature, we would have a 150,000 x 200 matrix as feature space. Running SVM or KNN to train a model on this feature set may take too long. Also, because both of our prediction tasks are in essence multi-class classification, we would have to perform SVM for how many different classes we have. This meant for quality we would have to perform it 20 times and for variety hundreds of times. Given our hardware, this is simply not feasible.

A. Model Description

Here we will give a brief explanation on what gradient boosting decision tree

1) Gradient Boosting Decision Tree: A gradient boosting decision tree model is a type of ensemble learning model that uses the boosting method[7]. It can be understood as having a set of different decision tree in sequence, each one's input is the wrong prediction of the previous tree. In the end, the final prediction is a weighted aggregate of the results from all the decision trees. To prevent overfitting, each one of these decision trees must be weak classifier, this means they must underfit the data by natural which means for each tree, their tree depth must be lower than the total amount of features in the data. So in essence each tree is a different classifier. The process of training a decision tree is to choose what feature to use that would best split the data into the next feature at every step. For us we will be using the gini impurity as our measure of 'best' since we are doing a classification problem.

2) Random Forest: Random forest model is also a type of ensemble learning model, however instead of boosting like gradient boosting tree[5], it uses the bagging method. In random forest, the decision trees are run in parallel, each with input sampled with replacement from the meta-data set. Not only is the data sampled, the depth of the trees are the same but the feature to each tree is randomly selected. Similar to boosted decision trees, each decision tree in random forest must be a weak classifier to prevent over-fitting. Again each tree is trained to choose which feature to split first is the best using gini impurity. The final result is an aggregate of the results from all the trees. Since



(a)

(b)

(c)

(d)

(e)

Fig. 4: Box plot of a) country against quality, b) designation against quality, c) province against quality, d) regions against quality, e)variety against quality

random forest is usually run in parallel, the runtime for random forest is usually faster than that of boosted decision tree which runs sequentially.

3) Advantages:

- **Generalize to large dataset easily**
  Both gradient boosting tree and random forest is in essence a tree based model, therefore both models can be generalize and trained on large datasets quite easily. Also with random forest the training time is very short.
- **Easy to explain result**
  With tree based model, it is possible to calculate the feature importance of the features, therefore it is also easy to explain how the model reaches the prediction.
- **Does not require a lot of data preparation**
  Essentially a decision tree model therefore not a distance based model like regression, SVM or KNN, it does not require a lot of data preparation. For example, it does not need one hot encoding for categorial data because it is not trained based on the distance between two points hence no need to make sure distance between all values within feature to 1 for categorical data.
- **its hard to overfit the model to data**
  Since each classifier in both models are design to be weak classifiers, as long as we keep a reasonable amount of estimators in both models, it is very hard to overfit the model to data which means this model should generalize well.

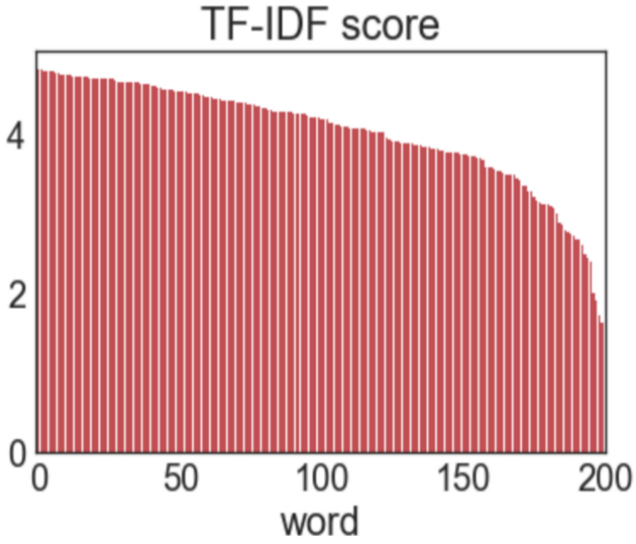


Fig. 5: top 200 TF-IDF scores and corresponding words

VI. Method

A. Quality Prediction

In the dataset the point is rated by sommeliers on a scale of 80-100. In the score prediction task, each wine in our dataset has points range from 80 to 100 with mean value of 87.89 and median value of 88.0. The goal for this point prediction task is to predict point for test dataset as close as to their real point.

1) Feature Selection: As discovered from data analysis, the features available to us immediately from the data set is not very information and we have decided to use TF-IDF instead. To confirm this hypothesis, we are experimenting with both sets of features. One is using important words from description of wine review as identified by TF-IDF, another is using the features available to us from the dataset as shown in I. We use encoded method to convert the non-numerical but categorical features into numbers. This would not encode a sense of 2 larger than 1 in our model since as mentioned before tree-based models are not distance models.

2) Model Evaluation: The evaluation metrics of point prediction we use is MSE (mean-squared error), Its equation is:

$$MSE = \frac{1}{n}\sum_{t=1}^{n}(x_t - \hat{x}_t)^2$$

The model will be trained by using gini impurity. For reference we also looked at mean absolute error, which measures the average magnitude of the errors in a set of predictions without considering their direction. This is because for MSE, since the errors are squared before they are averaged, it gives a relatively high weight to large errors so if the predictor is performing well overall with only a few predictions that has large error, the MSE will still be very large giving the impression that the predictor is not doing well while MAE will not have this problem.

We first shuffle our dataset with random seed = 7 and divide it into 8:1:1 . With 80% of data being training data to train our model. For tuning parameters such as regularization parameter in logistic regression model, max depth of decision tree and number of trees in random forest we will be using a validation set which is 10% of the data. To measure the performance of the model, we will be using the testing data which is the final 10% of the data.

3) Logistic Regression: Since the points are all integers, we can treat them as categorical classes therefore we could apply logistic regression onto our model. The result will be used as a baseline for our prediction. In order to perform logistic regression, we had to one hot encode all of the categorical features. The difference between logistic regression and softmax regression is that the linear regression term is not put into a logit function as in logistic regression but into a softmax function. The result will be the probability of sample being into the different classes in y. We will simply predict the sample to be the class with the highest probability. Similar to logistic regression there is only one hyper parameter to tune and that is the regularizer.

4) Gradient Boosted Decision Tree: XGBoost is an implementation of gradient boosted decision trees designed for speed and performance. The parameter that needs to be tuned are the number of classifiers and the maximum depth of each classifier as well as learning rate.

B. Variety Prediction

In Variety prediction task, the goal is to determine the variety of wine based on users' description. The approach is to use TF-IDF to extract features from description and formulate a vector representation for each description. Then we train three different models to make prediction on the test data. Since this dataset contains 8 balanced categories (see figure 6), performance of the model will be measured by accuracy.

1) Data Preprocessing: Wine reviews are from various countries and are for wide variety of wines. Most variety of wines only contains several reviews which we consider not appropriate and reasonable to do a prediction task. So only reviews from US,Italy, France, Spain, Chile, Argentina, Portugal, Australia, New Zealand, Germany, South Africa are selected. Among them top 8 varieties of wine are extracted since they have nearly balanced number of reviews.

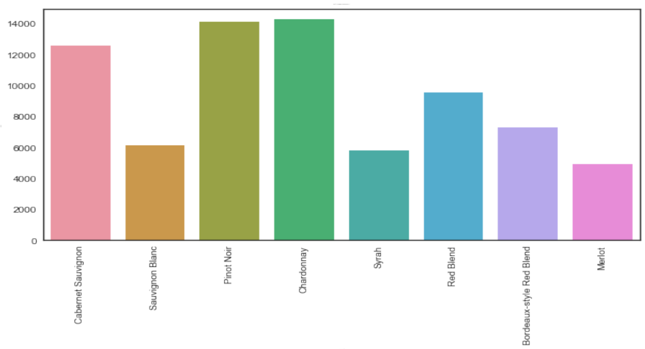


Fig. 6: Counts of top 8 variety

2) Logistic Regression: Logistic Regression are used as a baseline for our prediction. Also since variety is a multi class classification, we will be using softmax regression. There are only one hyper parameter in softmax regression and that would be the regularizer.

3) Random Forest: For random Forest, there are really only two hyper parameters to tune, the number of features per classifier or tree depth and the number of classifiers. The process is to do a grid search for optimal parameters. We want to determine the best maximum tree depth, which represents the number of significant words required to make robust prediction, and the number of estimators.

4) Gradient Boosted Tree: XGBoost is an implementation of gradient boosted decision trees designed for speed and performance. However, because in gradient boost tree, each classifier in trained in sequence, gradient boosted tree is much slower than Random Forest. Therefore, it was diffcult for us to tune for the perfect number of classifier for gradient boosted tree. In the end, we found the number of estimators being 60 to give reasonable result. Using learning rate 0.05, and default iterations. Tree depth are the only parameters to be determined.

C. Sentiment Analysis

Sentiment analysis is becoming a popular area of research and social media analysis, especially around user reviews and tweets. It is a special case of text mining generally focused on identifying opinion polarity, and while it's often not very accurate, it can still be useful. For simplicity (and because the training data is easily accessible) we'll focus on 2 possible sentiment classifications: positive and negative.

1) Data Preprocessing: Just two columns, description and points, are used to do sentiment analysis. The descriptions with points lower than mean value will be classified as negative review (labeled 0) and the descriptions with points higher than mean value will be classified as positive review (labeled 1). And we have known that the points of all wines are between 80 and 100. So the positive and negative in our task is relative.

First, we extract words from all texts after punctuation remove, Uppercase-to-lowercase transform, stopwords remove and word stemming. There are about 27,000 unigrams and about 660,000 bigrams.

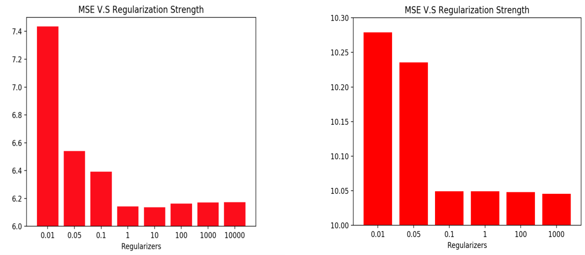
Then, by counting the frequency of each unigram and bigram, we decide to choose top 3000 the most frequent unigram and top 5000 the most frequent bigram as our features respectively.

2) Model Selection: Particularly in high dimensional spaces, data can more easily be separated linearly and the simplicity of classifiers such as naive Bayes, logistic regression and linear SVMs might lead to better generalization than is achieved by other classifiers.

So above three classifiers are used on this task. The unigram or bigram with higher weight will has more positive sentiment and the unigram or bigram with lower weight will has more negative sentiment.

VII. Result

A. Quality Prediction



a

b

Fig. 7: Logistic Regression model MSE vs Regularization Strength: a) use TF-IDF to select features from description, b) using categorical column information as features

1) Logistic Regression: As we can see figure 7, even with logistic regression model, using TF-IDF to select features from description out-performs using categorical features available to us from dataset. This confirms our hypothesis that the features from dataset is not very informative in predictive tasks. The best result we can get using softmax is around 10.05. This will serve as our baseline for evaluating other models

2) Gradient Boosted Decision Trees: Similar to logistic regression, we applied gradient boosted decision trees onto two types of features, TF-IDF and categorical features from dataset. Shown in II is the tuning process for TF-IDF and III for categorical features. Again, we can see that the model using TF-IDF features out-performs using categorical features by a large margin which confirms our hypothesis. Comparing the result to that of baseline, we can see that we have out-performed baseline by a huge margin. In the end we were able to achieve a MSE of 3.31 which means that on average every prediction is off the mark by about 1.9 points which is surprisingly good.

The result may be improved better if we had a more comprehensive dataset. As mentioned before, this dataset consists of only wine that scored from 80-100. This may mean that the language used to describe these wine is roughly similar as will be discussed later in sentiment analysis. If the dataset included the whole range of wine from 1-100, perhaps TF-IDF could pick out more informative words that we could use which might make the model even more accurate.

Max Depth	Number of Estimators	Learning Rate	MSE
10	100	0.05	5.40
3	100	0.5	5.04
10	100	1	4.57
10	100	0.1	4.45
10	100	0.5	3.94
300	100	0.5	3.33
100	100	0.5	3.32
100	**50**	**0.5**	**3.31**

TABLE II: Parameter tunning using TF-IDF features for Gradient Boosted Decision Trees

Max Depth	Number of Estimators	Learning Rate	MSE
1	100	0.5	7.81
10	100	1	7.06
3	50	0.5	6.98
10	100	0.05	6.73
3	100	0.5	6.63
3	200	0.5	6.15
10	100	0.1	5.82
10	**100**	**0.5**	**5.16**

TABLE III: Parameter tunning using categorical column features for Gradient Boosted Decision Trees

B. Variety Prediction

First, we obtained a baseline performance using logistic regression. The best accuracy achieved by logistic regression is 74.23% as can be seen in figure 8.

Using Random Forest Model with only one estimator, we want to investigate how the Tree Depth will affect the accuracy. It turns out that when the depth of decision tree reaches 60, the accuracy is 74.35% and will no longer improve. With 50 individual estimators in Random Forest Model, setting the maximum tree depth to 60 will give 87.17% test accuracy. Again the accuracy barely improve as the tree getting deeper. It can be concluded that the number of words essential for variety classification is around 60. These are representative words for differentiating wine varieties.

Fixing the maximum tree depth to 60 and increasing the number of estimators in our model, the test accuracy improves to 87.45% with 90 estimators. In fact the accuracy reaches 87.08% with 40 estimators and barely

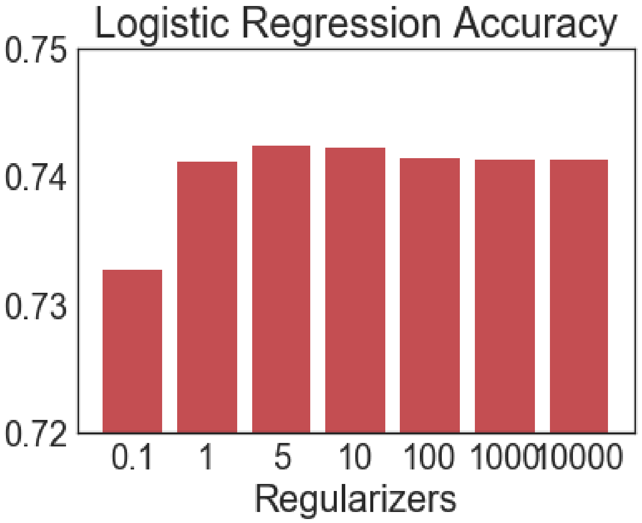
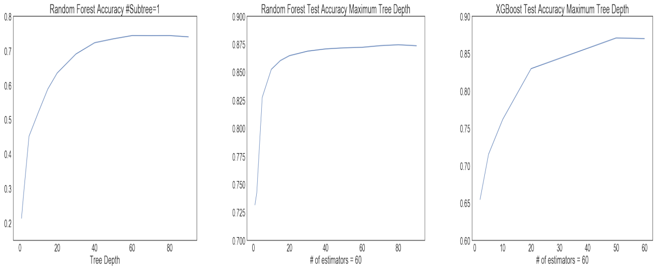


Fig. 8: Accuracy of logistic Regression



a

b

c

Fig. 9: Accuracy vs Tree Depth

improve even if adding more estimators. We conclude that 40 plus estimators in Random Forest model can make great prediction on this dataset. The best result we have is 87.45% accuracy on all 8 classes of wine while on kaggle the best result is 87.27% on only 4 classes.

Figure 10 is average feature importance of all estimators in random forest with black line representing one standard deviation. The top features are mostly name of the wine variety. It is intuitive because of there is a wine variety mentioned in the description, it is highly likely this description refers to that kind of wine. Other importance features including words like acidity, dry, and sweet which are mainly used to describe the taste the wine. Words like berry, wood, citrus, and oak which are used to describe aroma of the wine also appear in the top words.

Figure 11 is feature importance of XGboost. It shares majority of words with Random Forest but with different order. Varieties of wine are not as important as they are in Random Forest. Descriptive words are valued more heavily in XGBoost. It maybe the result of different arrangement of estimators in XGBoost. As mentioned in Model Description, estimators in XGBoost are arranged in sequence while in Random Forest they are in parallel. Random Forest choose features randomly, that's why the top features also have high variance.

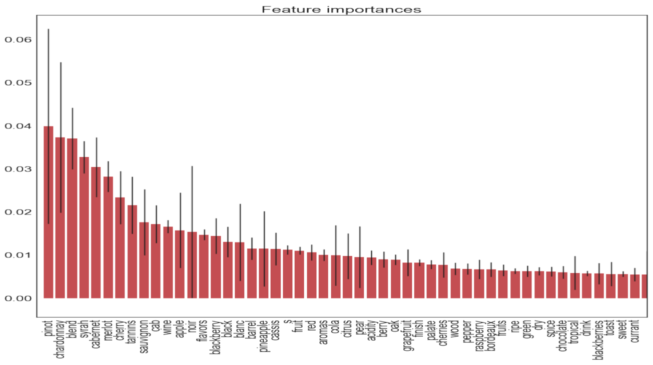


Fig. 10: Feature importance of Random Forest

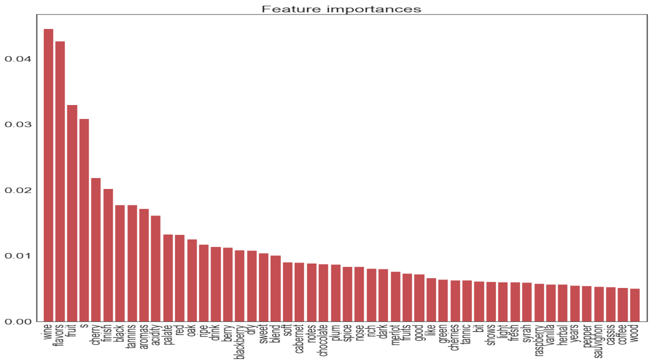


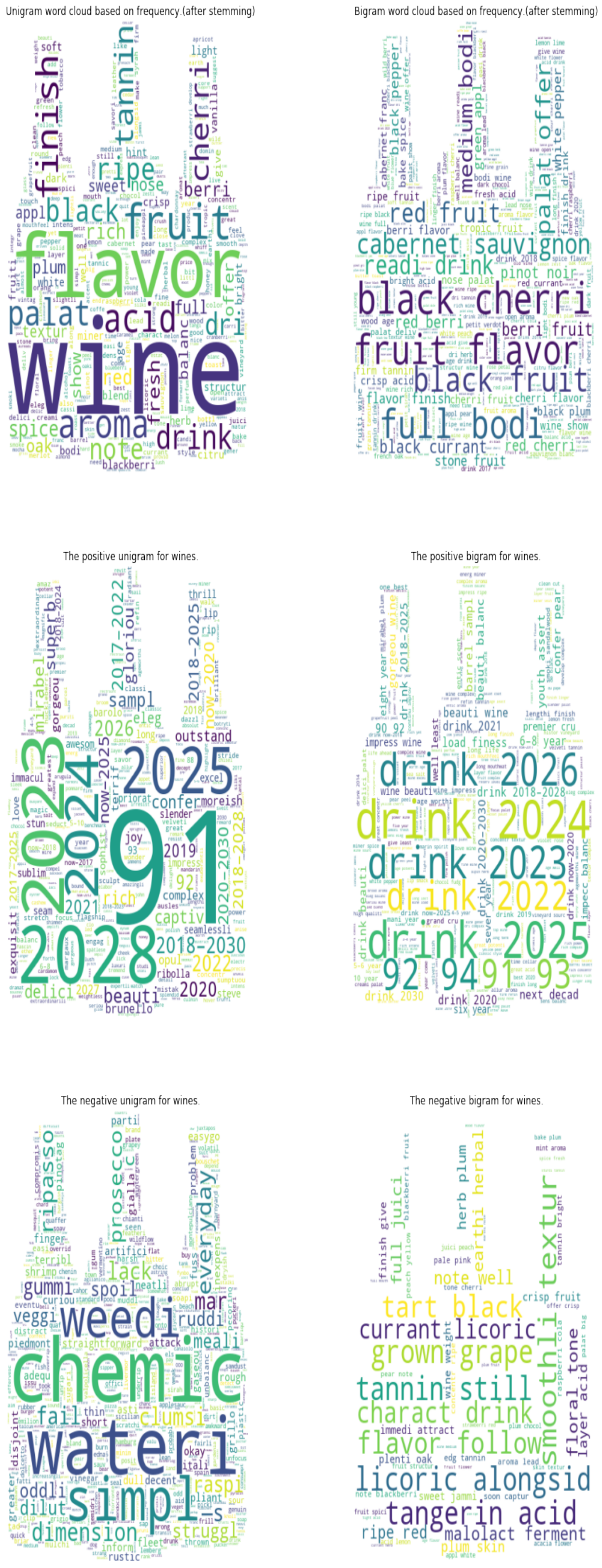
Fig. 11: Feature importance of XGBoost

C. *Sentiment Analysis*

Logistic Regression classifier has the best performance among the three linear classifiers. The prediction accuracies of the three classifiers are: 67.5% for Naive Bayes, 83.4% for Logistic Regression, 81.3% for Linear SVM.

Meanwhile, The top 3000 unigrams has much better accuracy comparing to the top 5000 bigrams when using logistic regression. The prediction accuracies are: 83.4% for top 3000 unigrams, 74.6% for top 5000 bigrams.

The result is shown in the Fig. 12. From the top two subfigures, we can tell that what unigrams and bigrams are more frequent used to describe wines. Other four



Unigram word cloud based on frequency (after stemming)

Bigram word cloud based on frequency (after stemming)

The positive unigram for wines.

The positive bigram for wines.

The negative unigram for wines.

The negative bigram for wines.

Fig. 12: Sentiment analysis showing in word cloud

subfigures can tell us the most informative unigrams and bigrams when wines are described. We can see that the years play a very positive role from the medium two subfigures. The wines will get higher points when it can be drank later. This result meets the property of high points wine which is ”buy now, drink later”. The bottom two subfigures tell us what kind of wines would get lower points. There are some reasons: wine color, wine taste, grape used to brew wines.

VIII. Conclusion

Using TF-IDF to extract features and form a vector representation of natural language can produce a good training set for potential prediction task. We have showed that it can be used to make prediction on variety and quality with reasonable confidence. We were also able to perform sentiment analysis on the descriptions and showed us what sort of words were used to describe good wine and bad wine which we found quite interesting and unexpected.

Notice that this dataset contains over 130k reviews of wine but only from 19 different individuals. So important words may not represent the distinct feature of each wine. The model may not learn a way to predict variety and quality of wine based on words to describe wines but a pattern of each user’s description for each wine. To Formulate a good classifier to make prediction based on reviews from variety of people, we needed descriptions from more people as to prevent TF-IDF to pick up language patterns of certain individuals.

Also, as mentioned before the data only contains reviews of wine that is 80-100 points already. This means that these reviews are of good wine already. This meant that our classifier would not perform well outside of this data set so it would not generalize well on other data of the same type.

IX. Future Work

- **Using more representative features in description text**
  Currently we are using words with high TF-IDF weight. However, there are other informative words such as adjective and verb. Those type of words tend to be very descriptive and indicate certain preference of tasters.
- **Dimension reduction for features**
  Taking example of quality prediction task, we use encoded number to represent categorical features. The TF-IDF feature matrix is also sparse. We

can use dimension reduction techniques such as PCA/SVD to eliminate useless dimensions

- **Experiment on more complex models**
  There are many other supervised learning models can be trained. We could have used more complex and robust models such as deep network. It takes more time for training but the performance could be largely improved.

References

[1] ”Introduction to Boosted Trees — xgboost 0.6 documentation” *Xgboost.readthedocs.io*, 2017, [online] Available: `http://xgboost.readthedocs.io/en/latest/model.html.` [Accessed: 04- Dec- 2017].

[2] ”Wine Classification informative words — Kaggle” Kaggle.com, 2017, [online] Available: `https://www.kaggle.com/dostalj/wine-classification-informative-words` [Accessed: 04- Dec- 2017]

[3] ”Classifying Wine Type by Review — Kaggle”*Kaggle.com*, 2017, [online] Available:`https://www.kaggle.com/carkar/classifying-wine-type-by-review.`[Accessed: 04- Dec- 2017].

[4] ”Predicting quality from review length and polarity — Kaggle”*Kaggle.com*, 2017, [online] Available:`https://www.kaggle.com/ludovici83/predicting-quality-from-review-length-and-polarity/notebook`[Acces 04- Dec- 2017].

[5] Breiman, L. Machine Learning (2001) 45: 5. `https://doi.org/10.1023/A:1010933404324`

[6] Wu, H. C.; Luk, R.W.P.; Wong, K.F.; Kwok, K.L. (2008). ”Interpreting TF-IDF term weights as making relevance decisions”. ACM Transactions on Information Systems.

[7] Mason, L.; Baxter, J.; Bartlett, P. L.; Frean, Marcus (1999). ”Boosting Algorithms as Gradient Descent” (PDF). In S.A. Solla and T.K. Leen and K. Müller. *Advances in Neural Information Processing Systems* 12. MIT Press. pp. 512–518.

[8] Alshari, E. M., Azman, A., Mustapha, N., Doraisamy, C., & Alksher, M. (2016). Prediction of Rating from Comments based on Information Retrieval and Sentiment Analysis. International conference on information retrieval and knowledge management.

[9] Pang, Bo; Lee, Lillian (2008). ”4.1.2 Subjectivity Detection and Opinion Identification”. *Opinion Mining and Sentiment Analysis*.