Wine Expert

An application helps people grade wines

Group 3: Yuchen Shen, Nathan Xu, YaoChun Hsieh, Yang Tao, Ying Fang

I. Executive summary

By its very nature, wine is a drink that suits a large variety of occasions: it can be served as an aperitif, an ideal accompaniment to a meal, or just for socializing with friends. More and more people started to taste wine and learn wine culture. The global wine market was led by Italy, France and Spain regarding the total amount produced. The U.S. was ranked as the fourth largest production country with a production volume amounting to 23.9 million hectoliters in 2016. We would like to help people distinguish the quality of wines and find wines having high similarity easily. Our project is to develop a new application which can help people know more about the quality and relevant information of the wine they are interested in. We also provide recommendations based on the scanning wines.

i. Motivation

With the growing popularity of wine, the number of wineries is increasing continuously as well, as shown in the figure 1. As a consequence, the number of wine products is going to increase rapidly as well. However, the speed of cultivating a sommelier can never catch up the speed that a bottle of wine being produced. We have to figure out how to fill the gap.

Since there are too many products in the market, which makes it difficult for people finding wines similar to their favor. Also, the description on the bottle is often written in local phrases, which makes it even harder when choosing wines. Moreover, people may pay money that way higher than the true value of a wine due to lack of awareness. Therefore, we believe it is necessary to develop a service that can save both wine lovers and new starters from these situations.

In addition, wine culture is more and more popular in modern society. For some people, tasting and sharing good wines with friends even become a lifestyle. Sometimes, even in the same batch, wine products have different qualities. Without professional knowledge and experience, it is very hard for people to tell the superior among so many products of wine. Therefore, there will be a thorny problem for the wine-starters to deal with: how can I identify the quality of a specific wine? How can I tell if a bottle of wine has high quality or not?

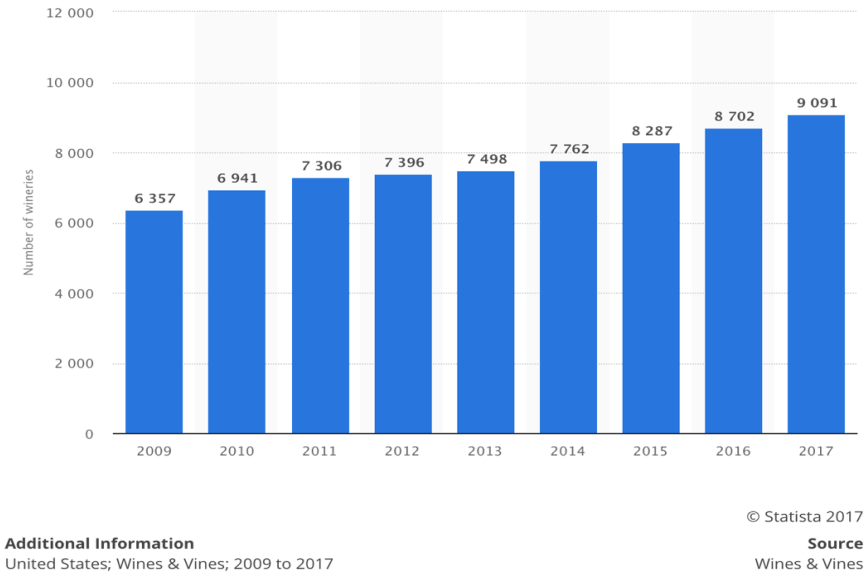


Figure 1 Total number of wineries in the United States from 2009 to 2017

ii. Solution

Our product is an application on mobile phones that users can check the grade of wines and relevant recommendations. Users can key in the name of wines, scan the barcode, or take the photo of label, then they will get the response of the wine quality score. At the same time, we also want more wineries being involved in our development so that we can collect more data and improve our grade system. In details, there has two parts to explain.

First, we have to fill the gap, which means we have to increase the speed of distinguishing quality of wines. Since sommeliers define the quality mostly by their tongues, if we can retrieve the chemical components from wine and turn these components into computable values, we can use machine to do the judgement. Therefore, we can build a classification system that can get chemical components of wine products, then judge the quality of a new coming wine using modern techniques with the retrieved chemical components.

Second, for people lack of experiences in wines or having hard time finding similar products, we can build a database that has a large wine dataset, providing query and recommendation services. To utilize the database, they may have to first provide some useful information to the system to do the query. For example, barcodes, wine labels, etc. With the system, they can avoid paying more money than the actual moderate price. They can also find similar products in a second.

iii. Expectations

We expect that our application can help wine lovers choose better wine they want. Besides, of the function part, we will also pay attention to advertise our application and improve our system to gain more customer preferences. So far, we have done the application development of database creation, statistical models building and UI design. In the future, we are looking

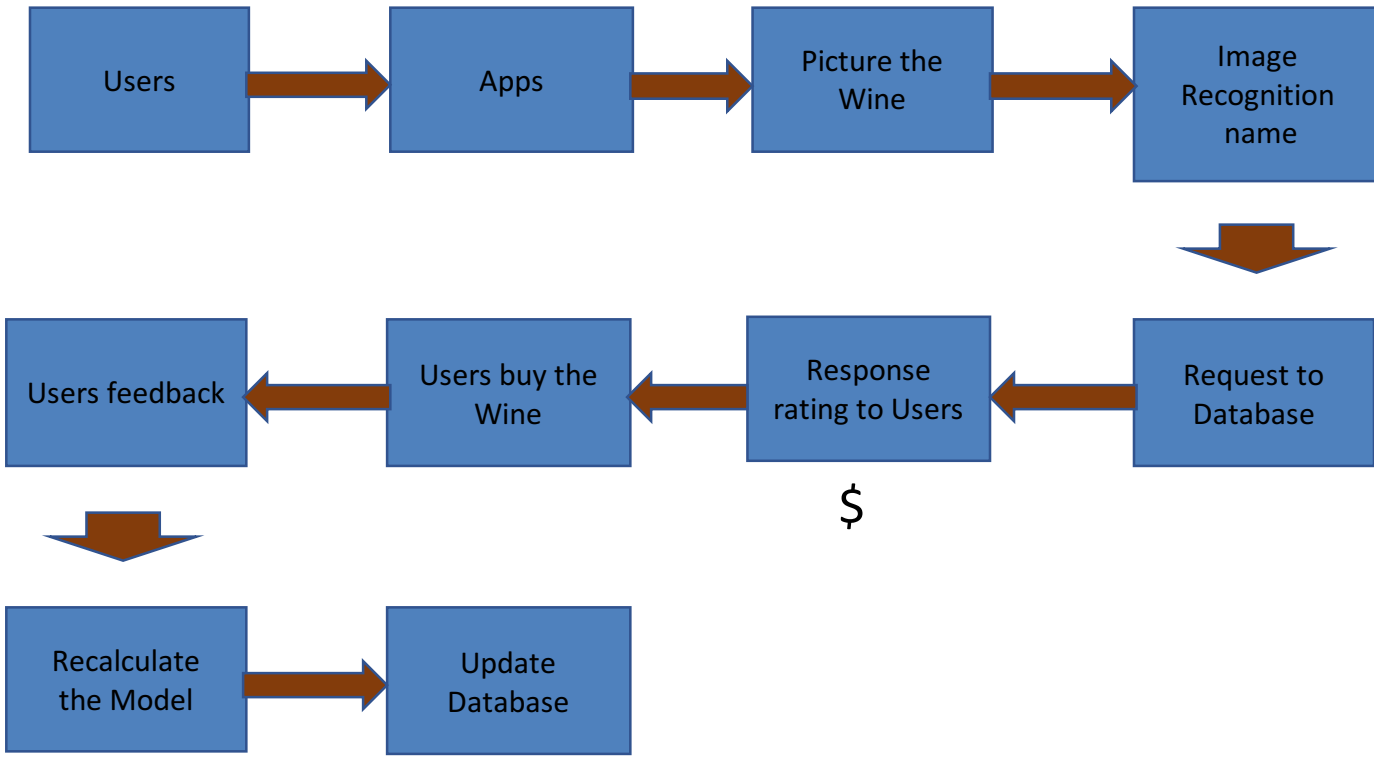
forward to cooperating with wineries to collect more wine product data as well as sell advertising to them, namely increasing their exposure chances in our application. We plan to release our application online in June 2018.

I. Product

i. Overview

We have two products. One product is a 'Wine Expert' application, from which uses can get the rating of a wine, as well as wine recommendations. Another product is wine quality calculation model with a database, which is used to predict the rating of the wine.

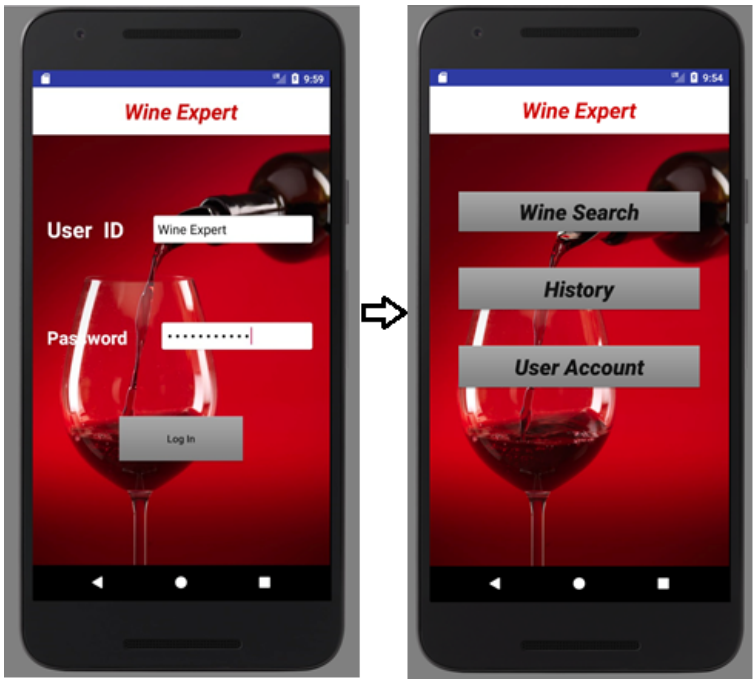
ii. Business Model:

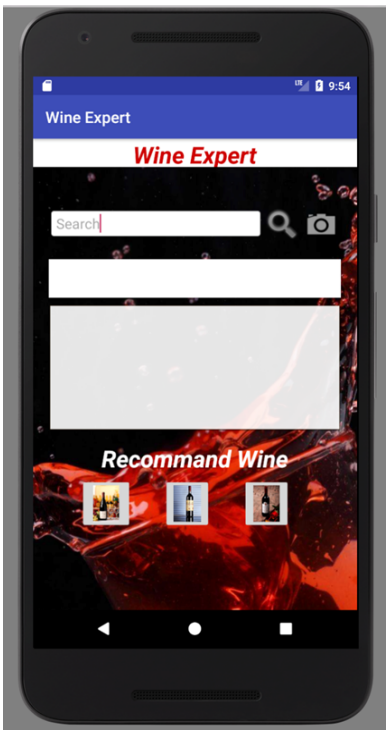


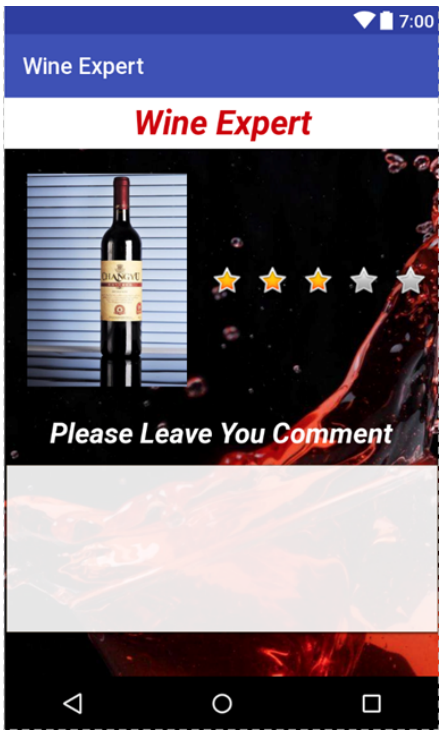
The picture above introduces our business model. As the user gets our app, he can picture the label or the barcode of the wine. Then, our system will try to recognize the name of the wine, and send a query to our database. If there's result in the database, we will return the rating of the wine to the user, and charge the money. After that, user can comment and rating on the wine he has searched for. We will store these feedback and rating into our database, and improve our model with it.

iii. App Details

Concerning 'Wine Expert' application, there are five main pages: login page, main menu, searching page, comment page and user account page. After log in through the log_in page, users will then enter the main menu, there are three selections: wine search, history and user account. If users choose 'wine search', they will enter searching page. We provide three wine brand collection methods: text insert, barcode scanning and take photos. After wine brand collection, we provide quality and relevant information of the wine on this page. The quality of a wine is displayed through a grade, the higher the grade, the better the wine. We also have a recommendation list. If users choose 'history', they will enter the comment page. We list all wines users' scanning histories that they can write down their comments about these wines on this page. In user account page, it shows information like how many free scan opportunities they have, account balance, and etc.







We also have a wine quality calculation model with a database. At the beginning, the model was built based on a wine dataset. Later, we will buy wines to test their quality based on the former model to enlarge our database. We can also get data from users' comment through our application. As we get more and more data, we will optimize our model and update the database periodically.

II. Method

The core value of our App is providing rating score to the wine you want. Thus, we need a model to predict the scores, and we also need a database to store the result and the feedback from customer. We will introduce the dataset we used to build prediction model.

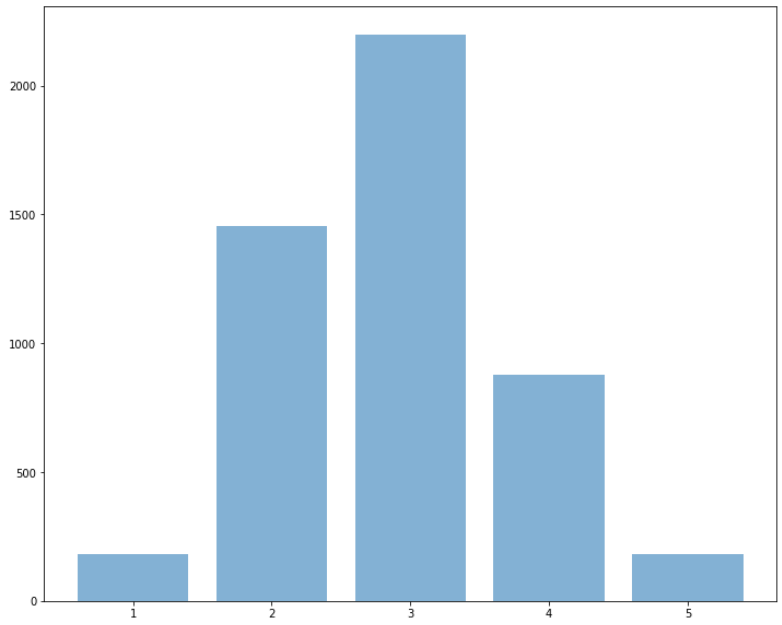
i. Dataset:

We gathered dataset from UCI Machine Learning Repository. We have 4,898 instances for white wine, 12 variables as follow:

x_1 - fixed acidity
x_2 - volatile acidity
x_3 - citric acid
x_4 - residual sugar
x_5 - chlorides
x_6 - free sulfur dioxide
x_7 - total sulfur dioxide
x_8 - density
x_9 - pH
x_10 - sulphates

x_11 - alcohol
y - quality (score between 0 and 10)

We used quality as our response. Although it is from 0 to ten, there were no score between 0 to 2, and there were no 10. Since we wanted to classify the output into only 5 sections, we relabeled all the quality. We relabeled 3 into 1; 4 and 5 into 2; 6 into 3; 7 and 8 into 4; 9 into 5. As a result, our response located on only 5 categories. The number of each new category show as follow:



The middle rating, which is 3, has the most rating number. And the best and worst rating have very few numbers. The graph is somewhat normal distribution, which is good as our expectation.

ii. Feature Selection

Since we have 11 predictors, we would like to reduce dimensions. We used Lasso to reach our purpose. We test different alpha to see the result. X_1~x_11

	rss	intercept	coef_x_1	coef_x_2	coef_x_3	coef_x_4	\
alpha_1e-15	1915.74	110.321	0.0595828	-1.47806	-0.0319032	0.0612464	
alpha_1e-10	1915.74	110.321	0.0595827	-1.47806	-0.0319032	0.0612463	
alpha_1e-08	1915.74	110.311	0.0595731	-1.47806	-0.031897	0.0612424	
alpha_1e-05	1915.93	100.806	0.0498617	-1.47872	-0.0257054	0.0572961	
alpha_0.0001	1926.89	39.1929	-0	-1.46799	-0	0.0310939	
alpha_0.001	2058.48	0.7444	-0	-0.842281	-0	0	
alpha_0.01	2702.72	2.87852	-0	-0	-0	-0	
alpha_1	2702.72	2.87852	-0	-0	-0	-0	
alpha_5	2702.72	2.87852	-0	-0	-0	-0	
alpha_10	2702.72	2.87852	-0	-0	-0	-0	

	coef_x_5	coef_x_6	coef_x_7	coef_x_8	coef_x_9	coef_x_10	\
alpha_1e-15	-0.400286	0.00282371	-0.000584637	-112.489	0.568153	0.566232	
alpha_1e-10	-0.400286	0.00282371	-0.000584637	-112.489	0.568152	0.566232	
alpha_1e-08	-0.400331	0.00282366	-0.000584623	-112.479	0.56811	0.566212	
alpha_1e-05	-0.445378	0.0027795	-0.000570356	-102.784	0.525242	0.545388	
alpha_0.0001	-0.67421	0.00237343	-0.000407523	-40.0688	0.265208	0.390942	
alpha_0.001	-0	0	-0	-0	0	0	
alpha_0.01	-0	-0	-0	-0	0	0	
alpha_1	-0	-0	-0	-0	0	0	
alpha_5	-0	-0	-0	-0	0	0	
alpha_10	-0	-0	-0	-0	0	0	

	coef_x_11
alpha_1e-15	0.181422
alpha_1e-10	0.181423
alpha_1e-08	0.181433
alpha_1e-05	0.191547
alpha_0.0001	0.255956
alpha_0.001	0.225263
alpha_0.01	0
alpha_1	0
alpha_5	0
alpha_10	0

For alpha equal to 0.0001, we had to drop x_1 and x_3 and kept all the others. If we chose alpha equal to 0.001, then we only kept 2 variables, which is too less. As a result, we decided to drop x_1 and x_3, and keep all other variables as our predictors, then we can build our model.

iii. Method

As shown before, our product is aimed to give a prediction of different wines' quality. In order to make it reasonable and reliable, we apply several classification methods, including linear regression, Linear SVC, K nearest Neighbor, decision tree. After doing all these methods, and trying to get a better performance, we took another experiment with an ensemble learning method called random forests. It is utilized broadly for classification, regression and other tasks. The forest consists of many individual decision trees. During training process, each tree in the forest grows while taking different features respectively. During testing process, each of these trees makes an independent judgment toward the input testing data, and the entire process is just like voting. The final decision of the category of an input data depends on the voting result; Category having highest votes wins.

All results are shown in the following table.

	LR	LinearSVC	KNN(K=200)	KMeans	Decision Tree	Random Forest
error	1.05823	1.51919	NA	NA	0.6592	1.32457
precision	0.64888	0.85965	0.7131	0.1775	0.6397	0.69132
recall	0.51368	0.35028	0.4577	0.1987	0.5204	0.66599
f-score	0.55284	0.45701	0.5423	0.1646	0.5625	0.67327

From the total result above, an encouraging result is achieved, Linear SVC model giving the best performances and all other methods show a quite reliable result too. Linear Regression and Decision Tree gives a result over 60%, and combing random forest method with 10-folds method, the precision is 0.691. Several parameters have been tuned during our research, however all the performances are not desirable.

a) KNN

We used KNN to predict the classification. The most important parameter in KNN is the number of neighbors. We have tried different K, and chose 200 as our final result. The precision rate is 0.71 after K-Fold. K equal to 200 means we need 200 neighbors to come up a decision, which is very large. One disadvantage of KNN is that it's hard for us to interpret the result, since we did not know which predictors matter the most. And also, the precision rate was not the highest, so we did not choose KNN as our decision model.

b) K Means

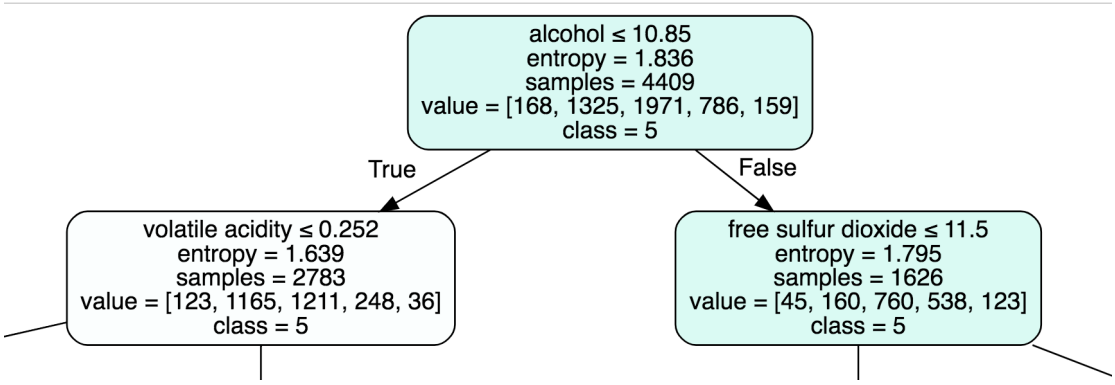
We used K Means to test our dataset. In K Means we tried to minimize the distance between each point to its center point. We also used K-Fold, and we set the cluster number equal to 5. The precision rate of K Means was really bad, which was only 0.20. The sklearn package in python only provide Euclidean distance method, and we could not select cosine distance. It's possible that cosine would generate better result since our dimension was high. We did not choose K Means as our decision model.

c) Linear SVC

The overall precision using 10-folds cross validation for Linear SVC is 0.86. These precisions are much better than the ones expected by a random classifier. According to our experiment, we apply Linear SVC classification method to our program. Although the precision is quite high, but the recall and f-score is really not good. Considering the distribution of our training data set, the model we use have a shortage that we tend to predict wine's quality to a normal degree of 3 and not 1 and 5, but actually we can have an accurate result of wine's quality. We chose Linear SVC as our decision model.

d) Decision tree

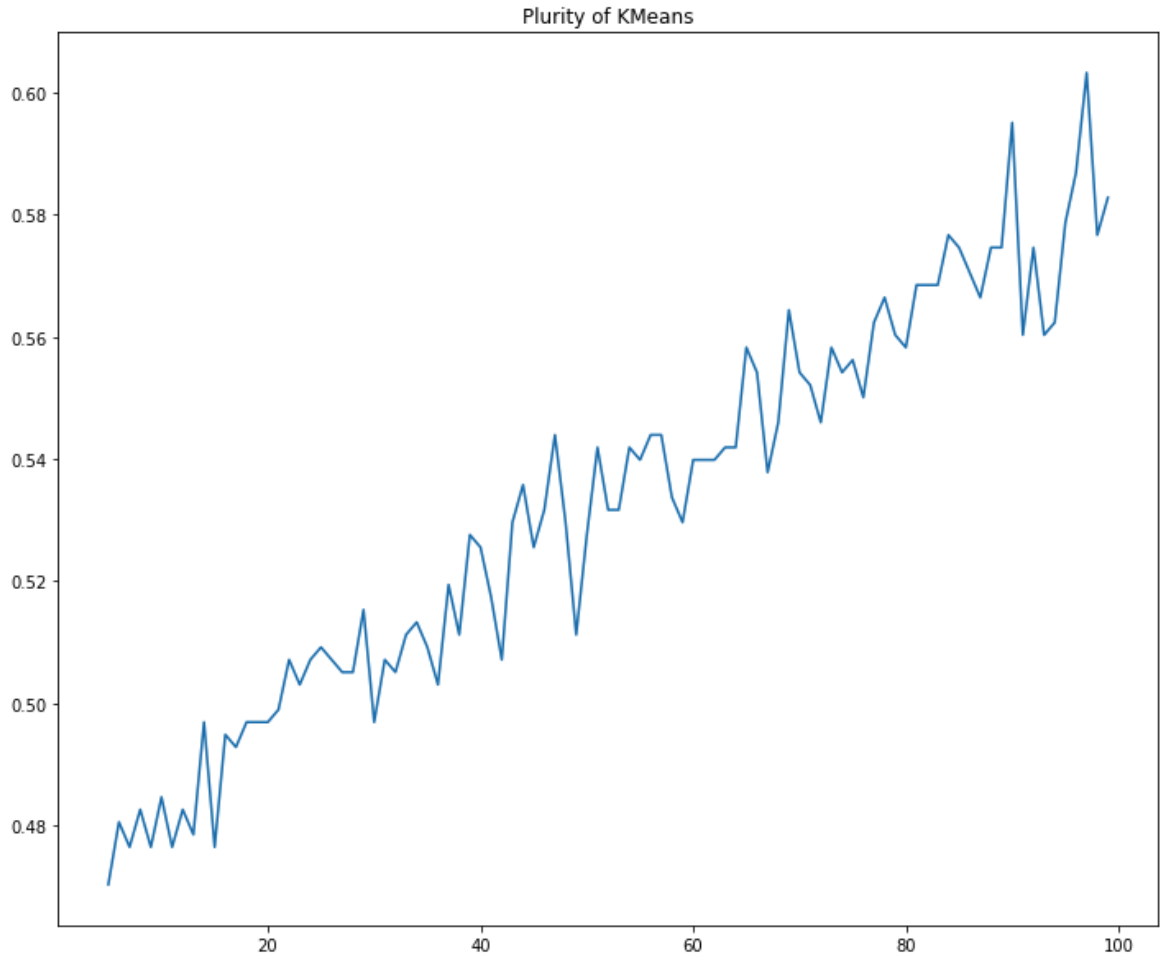
For decision tree and random forest method, we produce a tree graph below shows alcohol is the most important feature for qualifying a wine; When alcohol below 10.85%, volatile acidity is the feature to determine wine's quality while sulfur dioxide is most important for alcohol above 10.85%. And we can add these features in the wine description part to provide customers with some interesting chemical components.



iv. Recommendation (Clustering):

When the app returns a rating to the user, we would like to provide some recommendation as well. We did not just intend to recommend the wine from the same quality, we want to provide the one that taste similar. To be more specific, even in the same quality, the taste may still be different. That is in cluster 6 for example, we are trying to find 6.a, 6.b, or 6.c. They all have the same quality, but some component is different. We used K Means to implement. We iterate K=5 to K=99, at each step we calculate the purity of K Means, this suggest the accuracy indicator of this method.

```
92 0.6032719836400818
```



We can see that as the K increased, the purity will increase as well, however it's still far away from 1. At K=92, we have purity equal to 60%. The formula of the purity is as follow:

$$\text{purity}(\Omega, \mathbb{C}) = \frac{1}{N} \sum_{k} \max_{j} |\omega_k \cap c_j|$$

Even though the accuracy of the clustering is not very well, we can still use them and collect more data from customer and wine, then we can build a more reliable model.

v. **Database**(WineInfo):

In our dataset, we only have less than 5,000 instances, and we don't know the real name of each instances. Besides, there are more than 10,000 varieties of wine grapes over the world, the varieties of wines are definitely far more than 10,000. In another world, we have to build our own dataset rather than just use the dataset from UCI.

Similar to the UCI dataset, we need more wines. Then, we will test all the predictors we need from previous method for each kind of wine. For example, we buy a wine named “New Zealand Sauvignon Blanc”, and we test the sample to get volatile acidity, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates and alcohol. Then we predict the rating of this wine, and store all the values of this sample into our database. In the future, we can also add in customer’s feedback into each instance. Therefore, the final database will look like this:

WineInfo								
Name of wine	Predictor 1	Predictor 2	...	Rating (Predicted)	Customer ratings (1 star)	Customer ratings (2 stars)	...	other info

vi. **Database** (WineName):

Since we used the camera to distinguish the wine, we need a database that contains different pictures for each wine. For example, in the directory “New Zealand Sauvignon Blanc”, we can have like 50 pictures of the label from different angel. This database is used for image recognition. Each picture is labeled by their name.

vii. Conclusion

The services that our application provides can be generally separate into three main blocks, and all the required dataset and techniques for the implementation have been introduced above. First, quality ranking system using classification method. Second, wine data query system with wine database and image database. Third, wine recommendation system using clustering techniques. By integrating these 3 systems, we can provide users a comprehensive solution for judging the quality of wine and find wines with similar flavor.

III. Market Analysis

i. Target Market

Since we have two products in our project, we have two target user groups. The first group is wineries. When they make a new wine, they can hire us to test the quality of the wine. Based on our perfect model and large database system, we can provide them an accurate wine quality testing result. Thus, it is easier for them to set the price of a wine.

The second group is our application users, they can use our application to test the quality and get relevant information of the wine they offered. Meanwhile, based on the wine they offered, they can get professional wine recommendation. In short, our project has a large market.

ii. Competition and opportunities

There are similar applications in the market with ours like Vivino, HelloVino and WineSearcher. Vivino is a wine shopping and recommendation website with reviews of the product. HelloVino is an application where people can rate wine products and read others' review. WineSearcher is a website that give access to find and price wines, beers, spirits across all online stores. So far, there is no website or application have exactly same function with ours. The most special function of Wine Expert is that people can gain the quality of each kind of wine from our models and that would be quite credible since our raw data including experienced expert evaluation of wines.

Besides the regular customers, restaurants also can benefits from our application. According to an article on the Bureau of Labor Statistics website, it is estimated that a master sommelier can earn as much as $160,000 a year. Simply hired estimates that less-experienced wine tasters make an average of $71,000 a year. Referencing the results of our application to buy wines can help restaurants save a considerable amount of budget from hiring a professional wine taster.

IV. Revenue Prediction

i. Cost

We need money to buy wines to enlarge our dataset, as well as component analyzers to get the number of parameters and servers to support our application. In order to guarantee our dataset is large and our model is accurate, we need 1,000 bottle of wines in the first stage, suppose the average price of a bottle wine is $50, we need $50,000. Since there are nine predictors in our model, we need nine component analyzers to get relevant chemical parameters, each component analyzer is $3,000 on average, thus we need $27,000. We also need $1200 to rent a server per year. Thus, the total amount of money we need at the first stage is $78,200.

ii. Income

Our income mainly comes from three part: user paying for our app, wineries' advertising fee and other organizations paying for using our database. We estimate our future income on the basis of wine consumption in the U.S. statistics. According to the Wine Institute website, the total wine consumption is 949 million in gallons in 2016, increasing 2.9% comparing with 2015. Thus, the total wine per resident is 2.94 gals in 2016. There are 30% of American adults consume, on average, less than one drink per week. On the other hand, the top 10% of American adults – 24 million of them – consume an average of 74 drinks per week, or a little more than 10 drinks per day. There are 249,485,228 nearly 250 million adults in the U.S. according to the national population survey in 2016. Therefore, we can estimate that there are 175 million adults will drink wine, and 100 million adults love drinking.

Another survey from Sonoma State University (SSU) and the Wine Business Institute shows that 90% wine drinkers have a smartphone, 25% wine drinkers use wine apps to help them

decide which wine to buy. As our estimation, there will be 25 million drinkers will be our target group. We assume 10% of these drinkers will use our application 1 to 3 times a week, thus, we will have 25 million*10%*(1+3)/2*$0.01=$50,000 profit in this part weekly.

As the advertisement part, we will advertise for wineries in CPM Bidding which means wineries need to pay every time a user sees the ad, the cost will be $0.5 per thousand impressions. For example, if winery A put an advertisement on our app, the ad was seen 20 million times by users this month, winery A need to pay us $10,000 this month.

Since we are still keeping enlarging our database, the income from using our database don't count in this estimate of income. In the future, we will refine this part and put it in the market. So far, as our estimation, our income will be $27.2 million.

V. Conclusion

We used the dataset from UCI to build a classification model, and then we need to buy a lot of wine to predict their rating, and store the values in the dataset. Then, we build up an App as an interface for user to search our database. Each request we collect 1 cent from users. Wine market is very big, and most people would love to compare different wine when purchasing. So, we really have good opportunity on this. Our estimated cost will be 78,200 dollars, and we are looking for investors.