Department of Mathematics
Bar-Ilan University

May 2020

Final Project: Wine Clustering

Aviv Weinstein 315140475
Itai Schwed 208616664

Supervisor: Prof. Yoram Louzoun

Contents

1 Introduction

1.1 Dataset

The wines world is a wide-ranging world. There are many parameters that affect the quality and the type of the wine. The dataset ( ) that we chose includes 6497 samples which are divided to two wine types, red wine with 4898 samples and white wine with 1599 samples. Each sample, both "red" sample and "white" sample, composed mostly from all kind of physicochemical properties of wine, and also from wine color and wine quality:

- **fixed acidity** (g(tartaric acid)$/dm^3$) - The amount of non volatile acid in wine
- **volatile acidity** (g(acetic acid)$/dm^3$) - The steam distillable acids present in wine
- **citric acid** (g/dm^3) - The amount of citric acid in the wine
- **residual sugar** (g/dm^3) - The amount of sugar remaining after fermentation stops
- **chlorides** (g(sodium chloride)$/dm^3$) - The amount of chloride in the wine
- **free sulfur dioxide** (mg/dm^3) - A measure of the amount of SO^2 that is not bound to other molecules
- **total sulfur dioxide** (mg/dm^3) - A measure of both the free and bound forms of SO^2
- **density** (g/cm^3) - The density of the wine
- **pH** - A scale used to specify how acidic or basic (or alkaline) the wine is
- **sulphates** (g(potassium sulphate)$/dm^3$) - The amount of sulphate in the wine
- **alcohol** (% *vol.*) - The volume of alcohol in the wine
- ○ **wine color/type** - The color/type of the wine
- ○ **quality** (score between 0 and 10) - Median of at least 3 evaluations made by wine experts

The majority of the features that appear in the dataset are continuous, excluding the wine color, and wine quality, which are discrete. These discrete features have a greater potential for unsupervised classification, than the continuous ones. In the original paper, the proposed idea was to classify correctly the quality of the wine, but we chose a different perspective.

1.2 Goals

The first goal of our project is to successfully classify wine's type based on other properties without any supervision. In order to achieve that we tried to use different types of clustering algorithms.

Our second purpose is to detect anomalies, i.e. red wine samples with physicochemical composition of white wine and vice versa. Thus, we used various anomaly detection algorithms over our above mentioned clustering algorithms.

2 Data Analysis

The first step of our project will be to understand the data, including how do the features distribute, and what is the connection between the different features.

2.1 Visualization

Understanding the data must start with understanding the features. Here are some interesting graphs about the feature distributions:

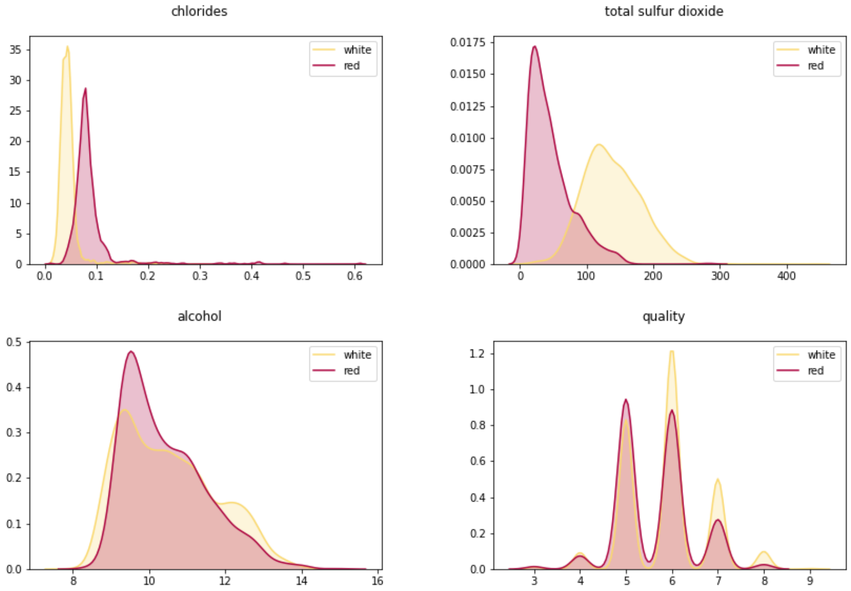

Figure 1: Features' Distribution

Another way to compare between the features with respect to the wine type, involves a kind of graph called boxplot, which is a method for graphically depicting groups of numerical data though their quartiles. The boxplots in figure 2 show a few more "important" features for our clustering, i.e. features that their values varies based on the wine type. For example, in the right boxplot the "white" mean is about 7, although the "red" mean is 8.

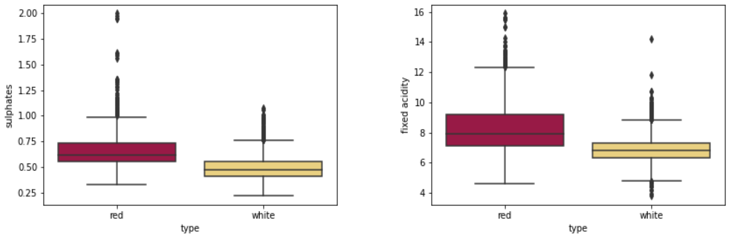

Figure 2: Additional Features' Distributions

From the graphs on Figure 2 and similar graphs appropriate to the other features, the following conclusions can be drawn. First of all, as we can see, most of the features are pseudo-normally distributed. Moreover, while on some of the features, there is a noticeable difference between the white samples and the red samples, on others, there is almost no difference. This conclusion will guide us in the "Data Pre-processing" section.

Finally, we will use a dimension reduction algorithm, called PCA (Principal Component Analysis), for representing the feature vectors (excluding the wine type feature), both in 2D and 3D dimensional graphs. The two graphs in Figure 2 present the resulted data points with regard to the wine type.

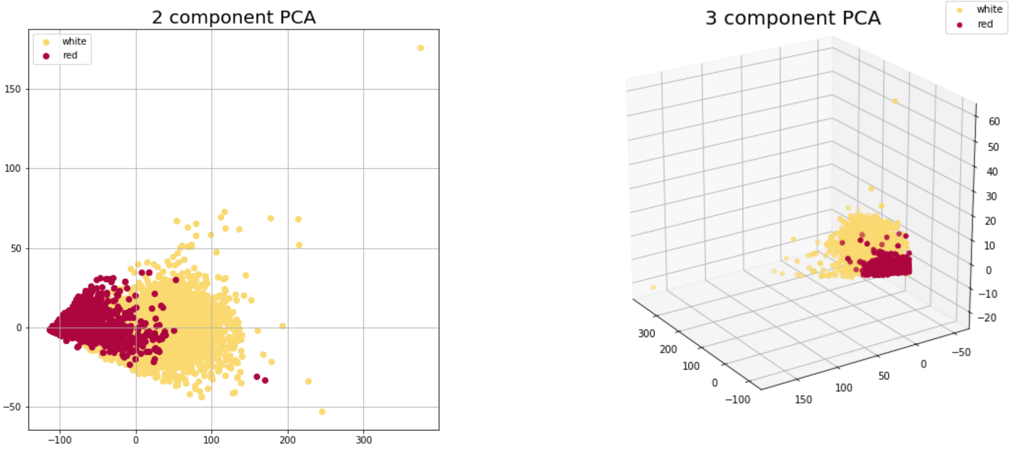

Figure 3: PCA Visualization

2.2 Feature Correlation

Once we have visualized the data, we can now explore the correlation between the features. We will use the Pearson Correlation Coefficient (PCC) which measures the strength of linear relationship between any two variables.

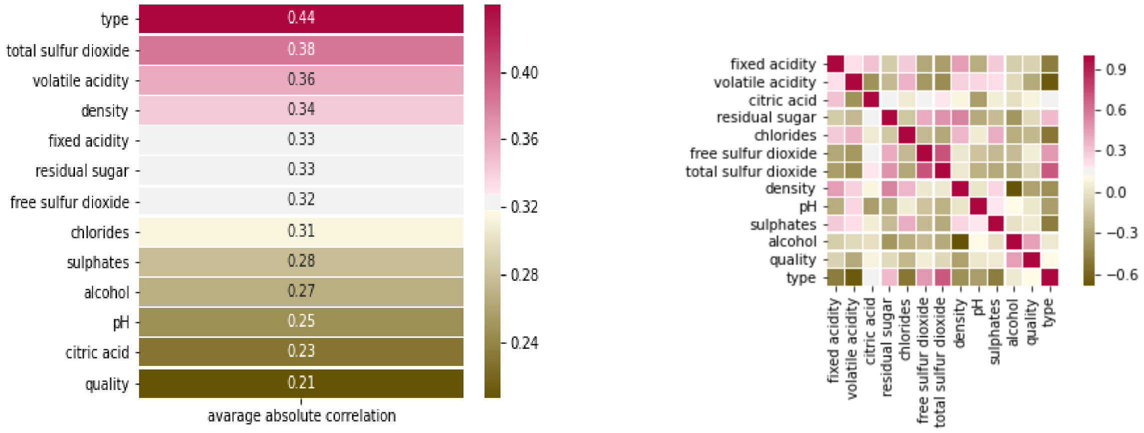

Figure 4: Features Correlation

As appears in Figure 4, the total and free sulfur dioxide are strongly correlated, and it is just natural since the first one is composed from the second one among others. However, most of the other features don't significantly correlated. You may also notice that the correlation between the quality feature and the rest of the features is relatively insignificant compare to the correlation between the type feature and the rest of the features that is Definitely higher, we can mostly see it at the compared graph based on the average correlation (as illustrated in the left chart of figure 4). Therefore, this observation reinforces our decision to predict the type of the wine rather than the quality of the wine.

In regards to our target feature, wine type, there is the biggest correlation

2.3 Data Pre-processing

In this section, we will describe our pre-processing procedure, based on the knowledge we gained about the dataset so far.

When cleaning any dataset, the first stage is to check for NaN values, i.e. missing values, in any of the features. Our dataset was given full with any feature value for each sample. Another thing to consider is the balancing between the target labels in the dataset. Fortunately, the target labels, white and red for the feature wine type, were enough blanced as we will see from the results. Here, the technical cleaning process ended.

Afterwards, we will dive more deeply to the influence of each feature on our classification problem. Considering the graphs in Figure 1, we can observe that the features alcohol and quality are very similar for the two wine types, red and white. According to that, we concluded that those two features would have relatively small effect on the type classification compared to the other features. Thus, those features are considered redundant for the purpose of our project. Moreover, considering the features free and total sulfur dioxide, which we have shown to have substantial correlation, we inferred that only the more informative feature should be remained. In the Clustering section, we will determine whether or not this decision was justify.

Finally, we will standardize each feature in the dataset based on its values, such that the features will be normally distributed. Although the features are already pseudo-normally distributed, as mentioned before, the standardization is used anyway to ensure that each feature will be distributed as a standard normal variable, since it is required for the k-means algorithm, and it is also recommended for other clustering algorithms.

As a result of the pre-processing procedure, the resulted data samples appear as follow, after applying again the PCA algorithm.

In figure 5, the white and red wines can be easily interpreted as two different clusters which are now more separated than has seen before, in figure 3. This suggest that separated clusters may appear at higher dimensions as well, and it motivates our clustering goal. In order to achieve that goal, those graphs will be used as a ground truth reference in the ”Clustering” section.

That is the end of the pre-processing procedure. Theoretically, we could have also removed outliers samples from the dataset, which can be interpreted as the points that any of their features has rare values (the black points in figure 2). However, we didn’t do that in the pre-processing stage, for several reasons. First, we avoided to remove data points before the clustering, due to the relatively small given dataset. Additionally, at the ”Anomaly Detection” section we will operate more complex algorithms over the clustering results, aiming to detect some interesting anomalies in the dataset. There, we will also test the success of particular clustering algorithms after removing the outliers.

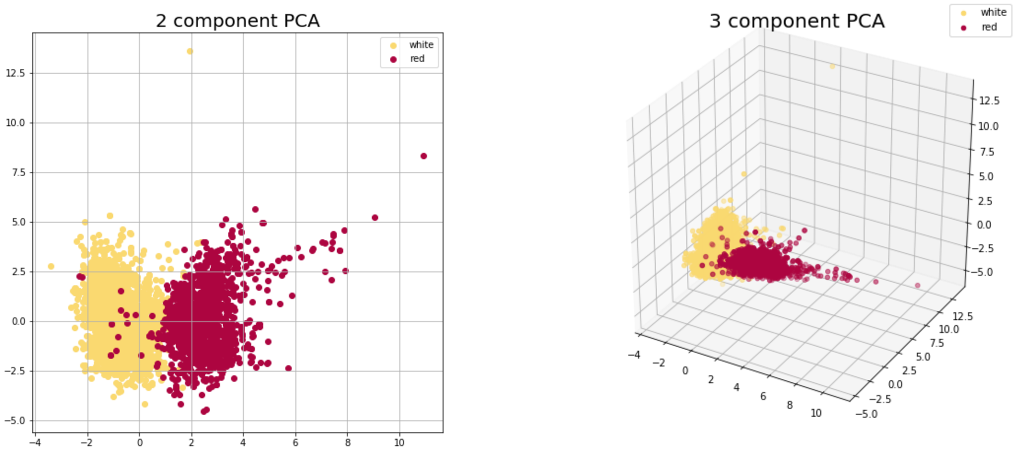

Figure 5: PCA Visualization after Performing Pre-processing

3 Clustering

In this section, we will present several clustering algorithms and their results after pre-processing the data. The clustering algorithms are sorted here from the worst one to the best one, based on our evaluation methods. For each algorithm, we will discuss the most suitable number of clusters according to different evaluating methods. Then, we will compare the algorithms and the ground truth in more depth, especially when the number of clusters is the same as the number of the labels in the ground truth, meaning two labels, red and white.

3.1 Gaussian Mixture Model

As we reach the clustering stage, the main goal is to adjust an algorithm that suits the data by its properties such as variance, distribution and so on. Looking at figure 5 it appears that each cluster distribute pseudo-normally, it motivated us to perform a distribution based model, that's the reason why we started with the *Gaussian Mixture Model* algorithm for our clustering task. One important transformation we decided to apply over the data In addition to the Pre-processing stage before applying the GMM algorithm is to normlize all samples, the assumption behind this decision was that the distances are less significant than the distribution of the features.

To begin with, we should find the best arguments to inject to the GMM algorithm, therefor we will inspect a range of optional arguments and show the scores of the GMM model with those arguments.

as we can see at figure 6, the combined best parameter is $k = 2$, therefor, we will use it with the GMM algorithm.

Once we have decided which k to use, we can now run the GMM algorithm, the results graph appear at figure 7 and the scores at table 1.

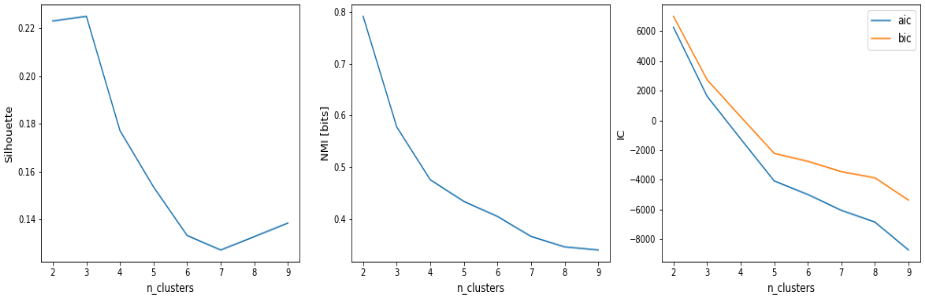

Figure 6: GMM - Scores as function of number of clusters

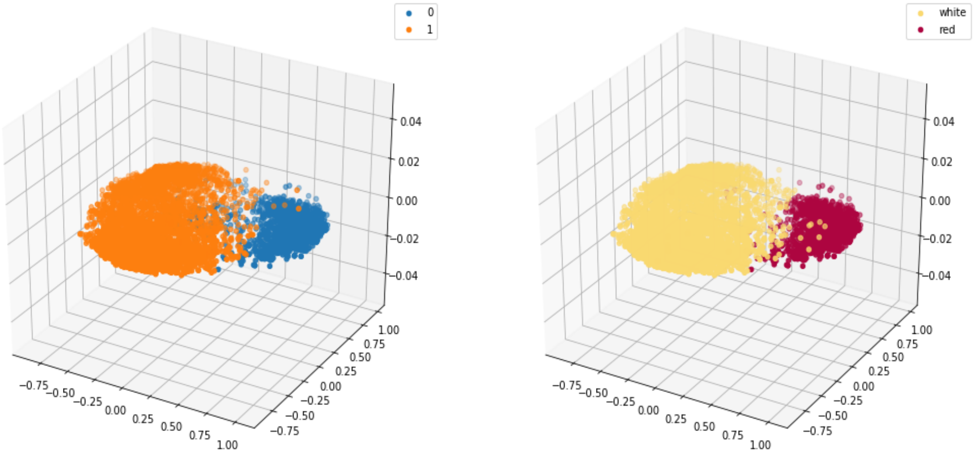

Figure 7: GMM algorithm results

	All Data	10-Fold Cross-Validation
Silhouette	0.2231	-
NMI	0.7911	-
Accuracy	0.9723	0.9721
F1	0.9634	0.9631

	RED	WHITE
Cluster 0	1559	40
Cluster 1	140	4758

Table 1: GMM Scores

3.2 K-Means

At figure 5 it appears that in each cluster the points are distributed Spherical around a centroid, furthermore, after calculating the variance of the ground truth samples related to each target label we discovered that thay have a similar one. therefor we decided to implement a centroid based models, thus, we applyed the *K-Means* algorithm over the data.

It's important to notice that the k-means algorithms depend on distances, therefor, it will be incorrect to normlize the data, this is why we used the data without normalization. As in the previous algorithms, we tried to find the best number of clusters, as shown at figure 8, the best one seems to be $k = 2$.

as we predicted, the k-means indeed achieve better scores than the preview (gmm) algorithm as shown at table 2.

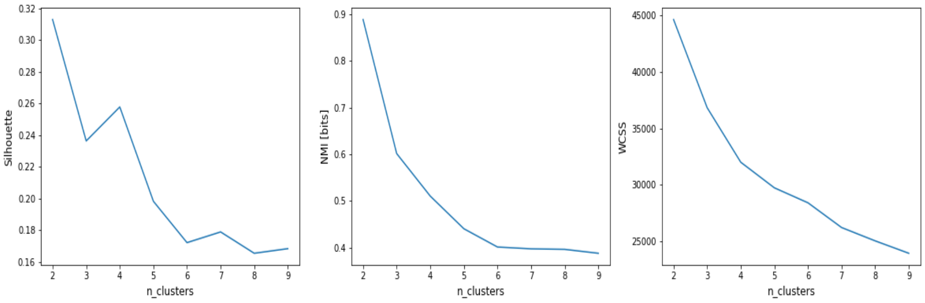

Figure 8: K-Means - Scores as function of numbers of clusters

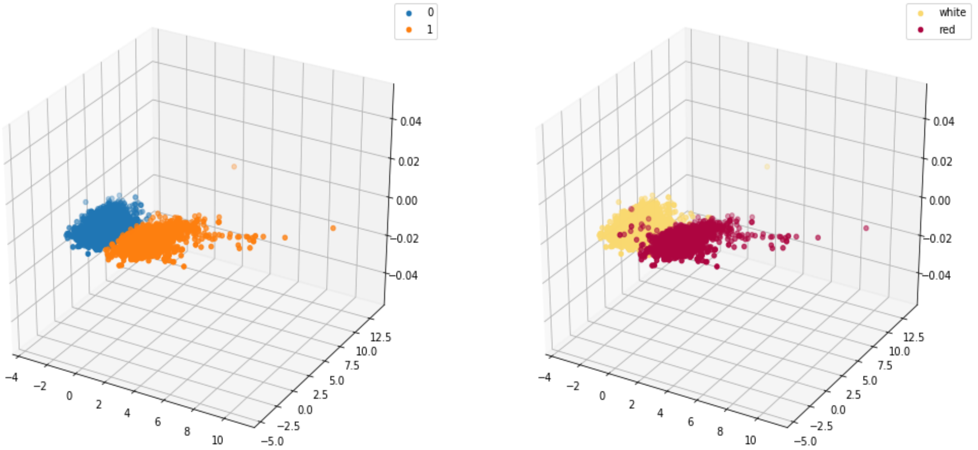

Figure 9: K-Means algorithm results

	All Data	10-Fold Cross-Validation
Silhouette	0.3128	-
NMI	0.8888	-
Accuracy	0.9880	0.9881
F1	0.9870	0.9841

	RED	WHITE
Cluster 0	1575	24
Cluster 1	54	4844

Table 2: K-means Scores

3.3 Spectral Clustering

Another interesting algorithm that we attempt to apply to our problem is *Spectral Clustering*. The spectral algorithm which originally intended to cluster problems with exactly two labels, seemed to have a great potential for solving our problem. The overall results of the algorithm were the best one that we achieved in this project, with better outcomes in most of the evaluation scores.

The Spectral Clustering partitioned a given connected graphs into separeted graphs. Our dataset is composed from data samples which were represented as vectors, so in order to use this algorithm, we had to somehow generate a graph from those vectors. We started with a naive graph which contained the distances between any two vectors. Then, we chose to use a distances graph over the k-nearest n, and received the results we discuss here.

Whenever more than two clusters are required, the algorithm is using hierarchical clustering technique, so the algorithm can also be used to search for the most suitable number of clusters represented by the data. Our data is represented as vectors, so we However, in this case this is not the only case of the are two parameters that affect the performance of the algorithm,

We ran spectral clustering algorithm with many hyper parameters (number of clusters and n)$k = 2$ using k-means algorithm were:

	All Data	10-Fold Cross-Validation
Silhouette	0.3111	-
NMI	0.9177	-
Accuracy	0.9918	0.9881
F1	0.9890	0.9854

	RED	WHITE
Cluster 0	1577	22
Cluster 1	31	4867

Table 3: Spectral Scores

3.4 Other Algorithms

During our research, we tried to run several other algorithms including *fuzzy-c-means*, *dbscan*, etc. those algorithms achieved worse scores therefore we avoided from elaborating on them

4 Anomaly Detection

One of the goals of our project we mentions at the Introduction is to *detect anomalies* on top of the algorithms we described at the last section.

5 Conclusion