Contents

1.1 Dataset

The dataset contains the information of red and white variants of the Portuguese "Vinho Verde" wine. It has been split into Test and Train sets containing 1822 and 1839 samples respectively. Moreover, there are two classes: good quality (value 1) and bad quality wine (value 0), where each has twelve attributes. Moreover, the table 1.1 resumes the information related to the attributes.

1. fixed acidity
2. volatile acidity
3. citric acid
4. residual sugar
5. chlorides
6. free sulfur dioxide
7. total sulfur dioxide
8. density
9. pH
10. sulphates
11. alcohol

1.1.1 Correlations

The image 1.2 illustrates the correlation between the attributes in the dataset. It can be extracted from the figure that the most correlated attributes are (7,10), (6,5) (6,3) and (7,3) (see table 1.2). Even though the correlation is higher in those attributes, the values are still unimportant (lower than 0.8) in order to consider a strong correlation between them. CONTINUE ANALYSISSSSSSS

Attributes	Range	Type
Fixed acidity	[3 ; 15.9]	Continuous
Volatile acidity	[0.1 ; 1.58]	Continuous
Citric acid	[0 ; 1]	Continuous
Residual sugar	[0.7 ; 23.5]	Continuous
Citric acid	[0 ; 1]	Continuous
Chlorides	[0.013 ; 0.611]	Continuous
Free sulfur dioxide	[2 ; 289]	Continuous
Total sulfur dioxide	[7 ; 440]	Continuous
Density	[0.9874 ; 1.0032]	Continuous
pH	[2.79 ; 3.9]	Continuous
Sulphates	[0.25 ; 1.36]	Continuous
Alcohol	[8 ; 14.9]	Continuous
Total sulfur dioxide	[7,440]	Continuous

Table 1.1: Range and types of the attributes

Attributes (by pairs)	Tag	Correlations
(7,10)	(density, alcohol)	-0.692
(6,5)	(total sulfur dioxide, free sulfur dioxide)	0.73
(6,3)	(total sulfur dioxide, residual sugar)	0.52
(7,3)	(density, residual sugar)	0.526

Table 1.2: Most correlated attributes

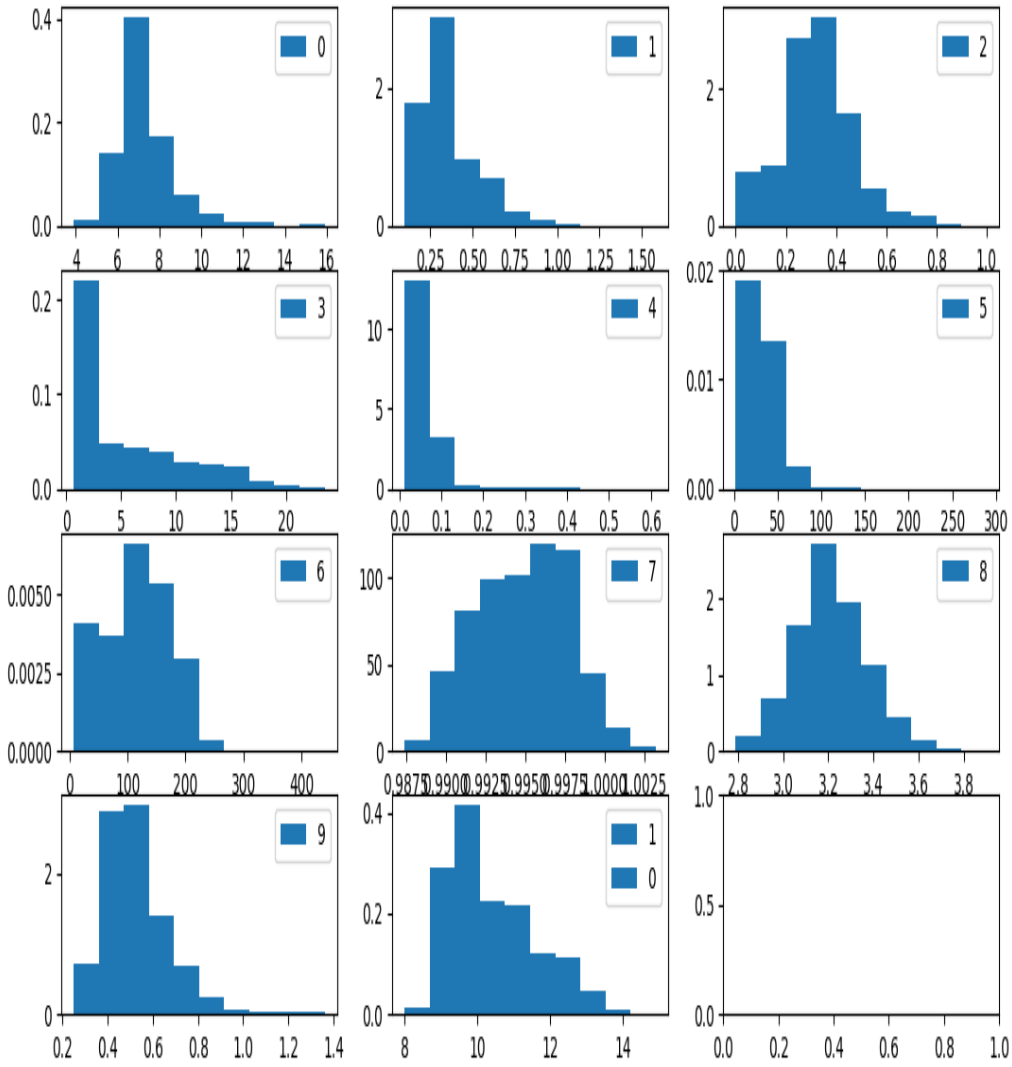

Figure 1.1: Histograms of the attributes (numerated) without modification

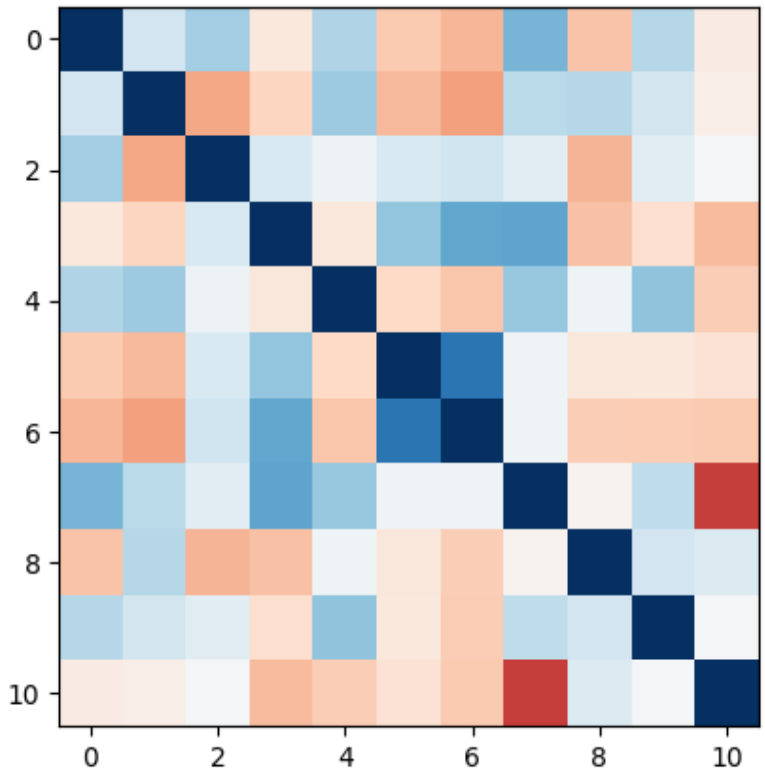

Figure 1.2: Correlations of the raw data (Pearson's coefficient)

1.1.2 PCA - Principal Component Analysis

Principal Component Analysis or PCA is a widely used technique for dimensionality reduction of the large data set. Reducing the number of components or features costs some accuracy and on the other hand, it makes the large data set simpler, easy to explore and visualize. Also, it reduces the computational complexity of the model which makes machine learning algorithms run faster.

Steps Involved in PCA

1. Standardize the data (with mean =0 and variance = 1).
2. Compute the Covariance matrix of dimensions.
3. Obtain the Eigenvectors and Eigenvalues from the covariance matrix
4. Sort eigenvalues in descending order and choose the top k Eigenvectors that correspond to the k largest eigenvalues (k will become the number of dimensions of the new feature subspace $k \leq d$, d being the number of original dimensions).
5. Construct the projection matrix W from the selected k Eigenvectors.
6. Transform the original data set X via W to obtain the new k-dimensional feature subspace Y.

Analyzing the eigenvalues obtained, we can see how almost all of the information can be obtained using 4 or 5 dimensions. By drawing the cumulative graph of the eigenvalues, the same conclusion can be extracted. -insert plot cumulative explained variance.png- Hence, PCA could be implemented and see if the accuracy improves significantly.

1.2 Classifiers

1.2.1 Without previous analysis

In order to classify the data, different models have been taken into account: mutlivariate gaussian classifier, tied covariance, Naive Bayes and a k-fold approach. The results are shown in the table 1.2.1.

Classifier	Accuracy (%)
Multivariate	80.46
Tied Covariance	81.44
Naive Bayes	78.98

Table 1.3: Different types of classifiers and their accuracy

The precision of the models seems to be similar, however the tied covariance model has a greater precision so far with 81.44 % and Naive Bayes performs the worst with 78.98 %. The explanation of the poor Naive Bayes performance can be due to the fact that when we perform the diagonalization of the covariances, we do so by putting to zero the other elements of the matrix, hence inevitably losing information. This loss of information might be the explanation of a worse performance than the other classifiers.