Data Mining and Data Warehousing Report - Wine Quality

Prepared by Iulia-Ana-Maria Mihaiu

The goal of the project is to predict the quality of the wine using the linear regression, to cluster the wine types using the k-means algorithm and also to classify them using a MLP (Multilayer Perceptron).

Exploratory data analysis

For this analysis I used the Wine Quality Data Set. In this dataset, there are **4898** entries of wines with **11 chemical properties** (fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, alcohol) and **quality** as the target variables. All variables are **numerical**.

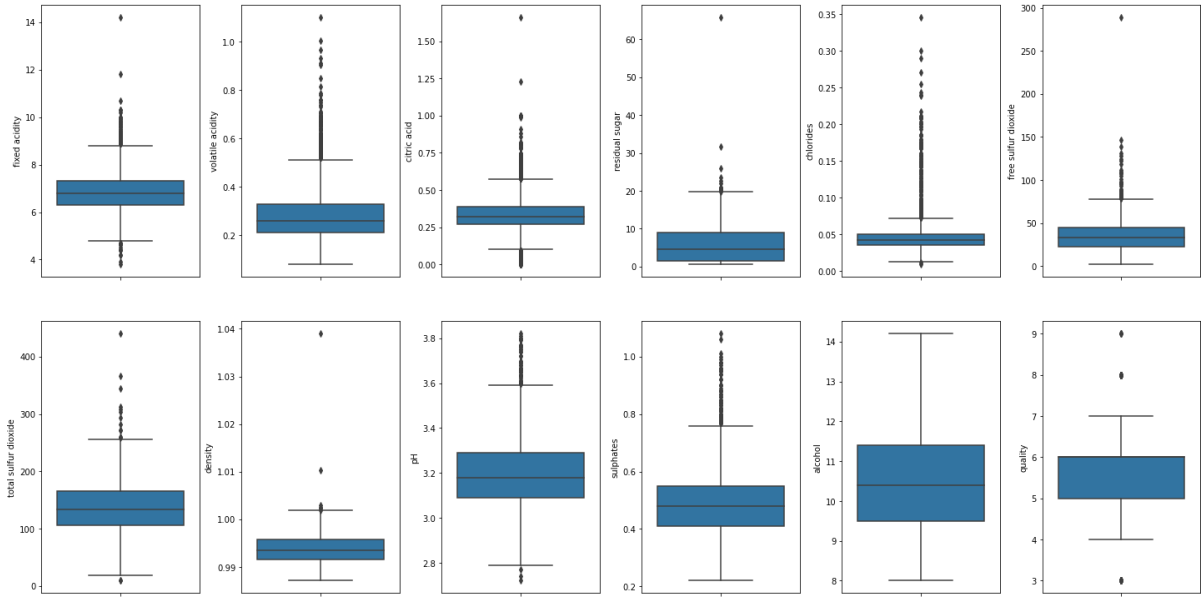
In this step, I started with checking for any empty cells or missing values, and I didn't come across any **missing value**. I also checked for duplicates and it looks like there are 937 entries of **duplicated data**, therefore I eliminated them.

Next, I had a look at the statistical summary of the dataset and I observed the following:

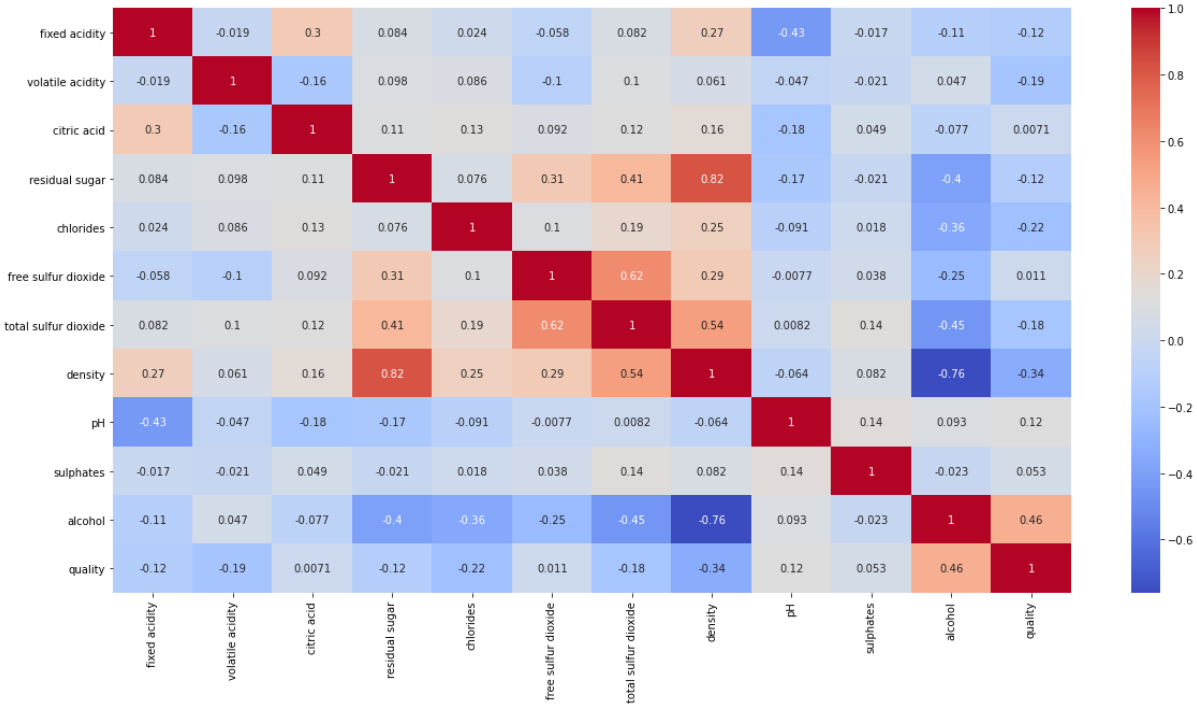
- Looking at the count value on each column there are no missing values;
- The **pH** attribute variates in between 2.72 - 3.82 (<7), which means that the pH of the wine is quite acid (the mean value is 3.1);
- The **density** variates between 0.987 - 1.038; the value of density is almost similar throughout the dataset, because mean, min, 25%, 50%, 75% are all ~0.99, therefore dropping the density feature will not have much significance in predicting quality of wine;
- The **alcohol** attribute variates in between 8 - 14.2;
- For the **target** data we have the **quality** attribute, which has values from 3 to 9;

Regarding the **value counts** of the attributes, the quality has 7 unique values and the majority of the wine entries (92%) are rated with a normal/medium quality (values 5,6,7), which makes the dataset **imbalanced**.

For each attribute I made boxplots and distribution plots so I can have a better look at the possible outliers values (see Figure below).



We can observe that there are some outliers in the dataset, from the plots it looks like the most distant outliers are in the density, residual sugar, free sulfur dioxide. There are also a few outliers that are different from the rest like total sulfur dioxide, sulphates, chlorides, volatile acidity and pH. Most of the outliers are in the larger side.



For identifying which attributes are more likely to affect the quality of the wine I used a Pearson correlation heatmap (see Figure above) and I observed the following:

- Quality is positively correlated with alcohol percent (0.46);

- Residual sugar is positively correlated with the density (0.82) and with the total sulfur dioxide(0.41);
- pH is negatively correlated with the fixed acidity (-0.43);

Data cleaning/Pre-processing

In this step I started with removing the duplicate values as they are an extreme case of nonrandom sampling, and they can bias the fitted model or lead to the model overfitting.

Linear Regression: I continued with undersampling the majority class and oversampling the minority class. This will make the data set less biased to the majority class. The majority class (3652 instances) will be undersampled to 1985 instances (half of the number of the data entries), and the minority class (309 instances) will be oversampled to 1985 instances, keeping the original dataset length.

Clustering: I applied principal component analysis(PCA) before k-means to improve the clustering results (noise reduction).The intuition is that PCA represents data vectors in a linear combination of a smaller number of eigenvectors, and does it to minimize the mean-squared error. It helps because PCA aims at compressing the features while clustering aims at compressing the data-points.

Splitting the data into training/testing, creating the data and fitting the model

Linear Regression: I tried the linear regression on the baseline data, on the data without the outliers, on the oversampled data with SMOTE and without SMOTE.

Clustering: I used the k-means on the baseline data, removing the target variable and also on the data on which I applied the PCA.

Classification: I used the MLP on the baseline data.

Building, testing and validation

Linear Regression: I applied the linear regression to the datasets and then we compare the metrics(r-squared score, MSE -the squared difference between the actual and the predicted value- and RMSE - the square root of the variance of the residuals) to see how well the regression model fits the data.

Linear Regression Dataset	CVS

BASELINE DATASET

	Actual	Predictions
0	5	5.0
1	5	5.0
2	6	5.0
3	5	6.0
4	6	6.0

R2: 0.26432565956373133
MSE: 0.5734573576502899
RMSE: 0.7572696730031448

OVERSAMPLED DATASET

	Actual	Predictions
0	8	7.0
1	8	6.0
2	6	5.0
3	8	6.0
4	4	4.0

R2: 0.4256486738107337
MSE: 1.4408301251928874
RMSE: 1.2003458356627423

DATASET WITHOUT OUTLIERS

	Actual	Predictions
0	5	6.0
1	6	6.0
2	5	6.0
3	5	6.0
4	5	6.0

R2: 0.3202409675358624
MSE: 0.49992313887926215
RMSE: 0.7070524300780403

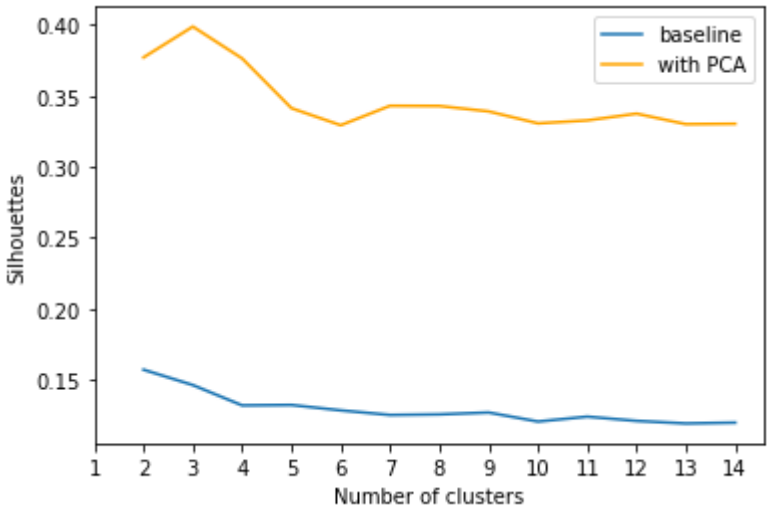
DATASET WITH SMOTE

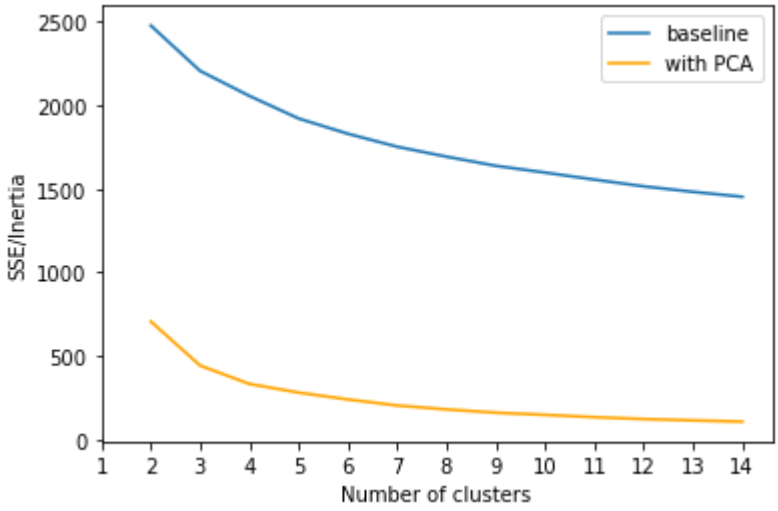
	Actual	Predictions
8513	7	8.0
309	6	6.0
1191	6	5.0
4614	3	4.0
4526	3	7.0

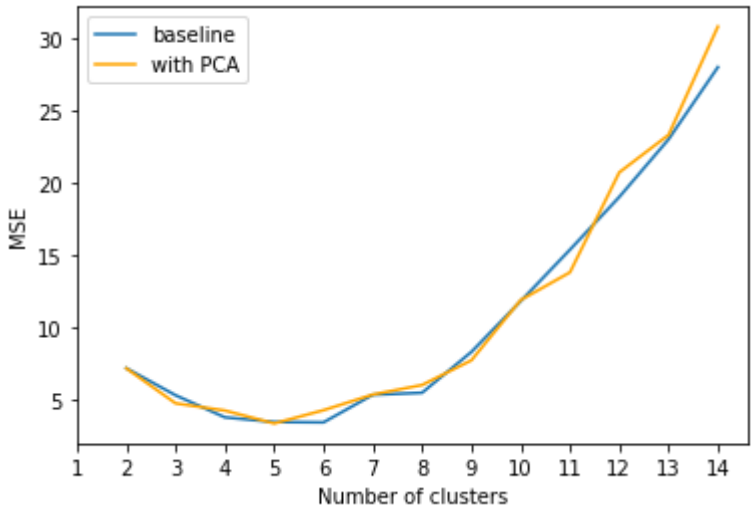
R2: 0.4256486738107337
MSE: 2.0631396615771482
RMSE: 1.436363345945986

Linear Regression Dataset	CVS
Baseline	26.17
Oversampled	40.56
Oversampled with SMOTE	48.15
Without outliers	30.76

Clustering: I applied the K-means algorithm to the data set with and without the PCA and after that we compare the Inertia - measure of how internally coherent clusters are, the Mean Squared Error MSE) -sum of the Euclidean distances between each pattern and its cluster center, and the Silhouette score for the k-means applied to all the before mentioned data sets.







Classification: I fit the dataset in the MLP model and then we check the accuracy to determine how good is the model at identifying relationships and patterns between the variables in the dataset, and we also check the f1 score to see how good is the prediction of our model.

For all of the models mentioned above I calculated at the end a cross validation score.

```
Accuracy: 0.4878048780487805
F1_score: 0.212151719083929
cvs train
[0.55495495 0.47927928 0.47111913 0.48555957 0.47833935]
0.4938504569551501
cvs test
[0.42016807 0.50840336 0.4789916  0.42436975 0.43881857]
0.45415026770201755
Grid Search
MLP best score:  0.5053936969460435
MLP best parameters:  {'alpha': 1.0, 'max_iter': 400, 'solver': 'lbfgs'}
```

Confusion matrix

```
[[  1   0   2   3   0   0   0]
 [  0   2  18  17   2   0   0]
 [  0   1  93 264   4   0   0]
 [  1   0  42 461  28   0   0]
 [  0   1   2 183  23   0   0]
 [  0   1   1  31   7   0   0]
 [  0   0   0   1   0   0   0]]
```

	precision	recall	f1-score	support
3	0.50	0.17	0.25	6
4	0.40	0.05	0.09	39
5	0.59	0.26	0.36	362
6	0.48	0.87	0.62	532
7	0.36	0.11	0.17	209
8	0.00	0.00	0.00	40
9	0.00	0.00	0.00	1
accuracy			0.49	1189
macro avg	0.33	0.21	0.21	1189
weighted avg	0.47	0.49	0.42	1189

Results/Conclusions

Linear Regression: The preprocessed data with oversampling fit our linear regression best out of the ones evaluated.

Clustering: When analyzing the graph referent to the inertia values and using the elbow method (locating a bend in the plot), we determine that the optimal number of clusters for the baseline data set is 4, and 3 for the dataset with PCA. MSE and Silhouette score quantify the quality of clustering achieved with k-means. The ideal number of clusters is the number that minimizes MSE and maximizes silhouette score. For the MSE values that number is 6, and the maximum silhouette score is for 3 clusters.

Classification: We managed to obtain an accuracy score of 51% with the following parameters for the MLP classifier: {'solver': 'lbfgs', 'max_iter': 500, 'alpha': 1.0}.