Classification of Wine Quality

Anonymous Authors

January 9, 2019

1 Introduction

The following report outlines our attempts to use common classification algorithms to predict the wine quality depending on the given features. We use a data set from the UCI Machine Learning Repository [1] [**data**].

1.1 Data Description

The data set contains of 4898 data points, each of them described by 11 features and an integer quality label ranging from 3 to 9. Note that the distribution of the quality is unbalanced: More than 90% of the data points are assigned to a quality between 5 and 7 (Figure 1).

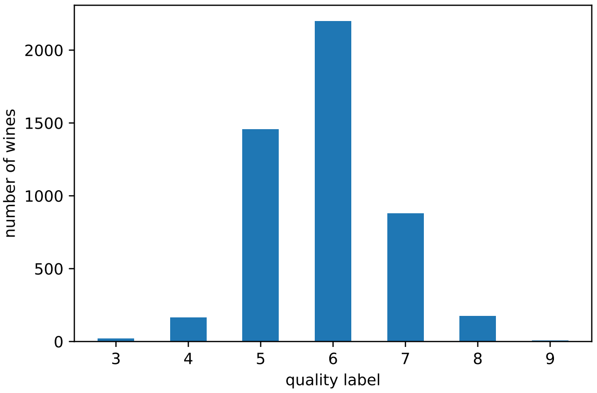

Figure 1: distribution of quality

1.2 Data preprocessing

To simplify our problem we assigned new class labels 0, 1 and 2. We assign class 0 (bad) to wines ranging in qualities 3-5, class 1 (medium) to wines labeled with quality 6 and class 2 (good) to wines with quality greater than 6. We further decided to use classification instead of regression. Though, it should be noted that we also examine algorithms (e.g. ridge classifier) that use regression to classify the data points and using regression should therefore yield similar results in our case. This doesn't come as a surprise, considering that the classes can obviously be linearly ordered.

2 Ridge Classification

2.1 Description

2.2 Cross-Validation

2.3 Results

3 Multi Layer Perceptron (MLP)

3.1 Description

MLP is a supervised learning algorithm and is a function approximator that can be used for classification and regression. We have used the MLPClassifier function (rather than MLPRegressor although both could be used on our chosen data set) from scikit-learn's Neural Network library.

3.2 Cross-Validation

3.3 Results

4 Random Forest Classification

4.1 Description

Random Forest Classification is also a supervised learning algorithm that is widely used for solving classification and regression problems. The idea behind this method is to increase the classification accuracy (a.k.a aim for lower variance) by using bootstrap aggregating (or bagging) algorithms. Generally speaking, the gross idea behind Random Forest Classification is to use the bagging method to split up a decision tree into two different trees. When one does that enough times, there is a large amount of trees (hence the name- ”forest) which gives the possibility

[1] `http://archive.ics.uci.edu/ml/datasets/Wine+Quality`

to gather more information for analysis and therefore better accuracy. [2]. In this project we used the RandomForestClassifier algorithm from the Scikit-learn library.

4.2 Cross-Validation

4.3 Results

5 Gaussian Process Classification

5.1 Description

5.2 Cross-Validation

5.3 Results

6 Comparison

7 Conclusion

[2] https://towardsdatascience.com/the-random-forest-algorithm-d457d499ffcd