Assignment 1 Classification Task

2Group 21

Sophie Rain(01425316),

Lucas Unterberger(01325438),

Peter Stroppa(01326468),

184.702

Machine Learning

Inhalt

**Datasets** 3

* Wine 3
* Mushrooms 4
* Congress 4
* Amazon 4

# Datasets

in this Section we give a short overview about some generell characteristics of the datasets

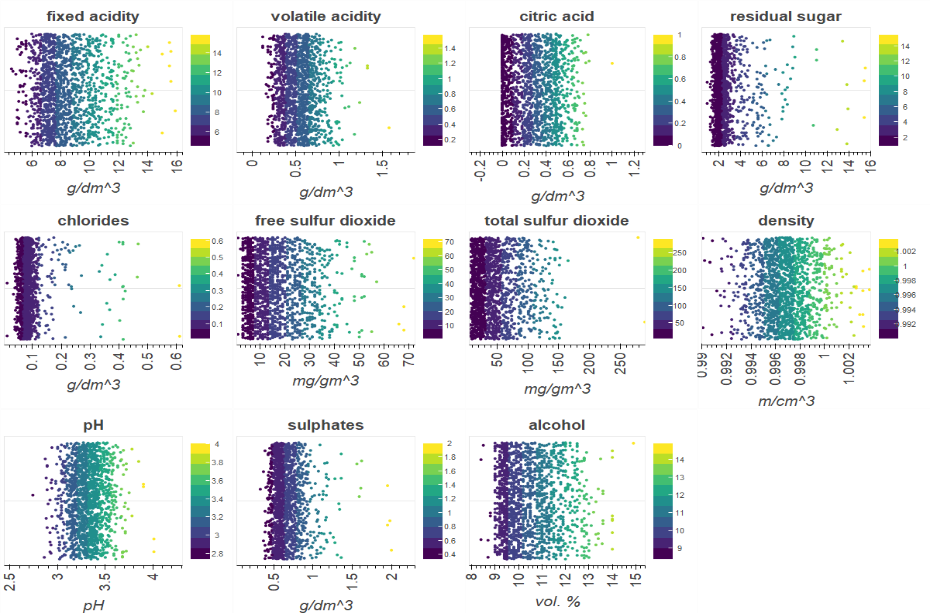
## Wine

Link to the dataset: <http://archive.ics.uci.edu/ml/datasets/Wine+Quality>

The wine quality dataset consists of 1599 red wine samples of the Portuguese “Vinho Verde” wine. The wine samples were classified according to their quality. This classification is purely based upon sensory data obtained by the median of 3 evaluations made by wine experts.

In this assignment we applied three different classification methods in order to classify the given wine samples into one of the ten possible classes, using only the data given.

A brief overview of the Red Wine Dataset:

* 11 attributes
* 10 classes (0 – 10) / only 3 – 8 are assigned
* Only numeric values (no missing values)
* Several different ranges and units according to attributed

For precise information about the attributes we studied the underlying distribution.

|  |  |
| --- | --- |
| **Fixed acidity** | Exponential normal distribution |
| **Volatile acidity** | Chi distribution |
| **Citric acid** | Gamma distribution |
| **Residual sugar** | T distribution |
| **chlorides** | Logarithmic Laplace distribution |
| **Free sulfur dioxide** | Exponential normal distribution |
| **Total sulfur dioxide** | Johnsons’s SB-distribution |
| **density** | Double gamma distribution |
| **pH** | T distribution |
| **sulphates** | Exponential normal distribution |
| **alcohol** | Exponential normal distribution |

## 

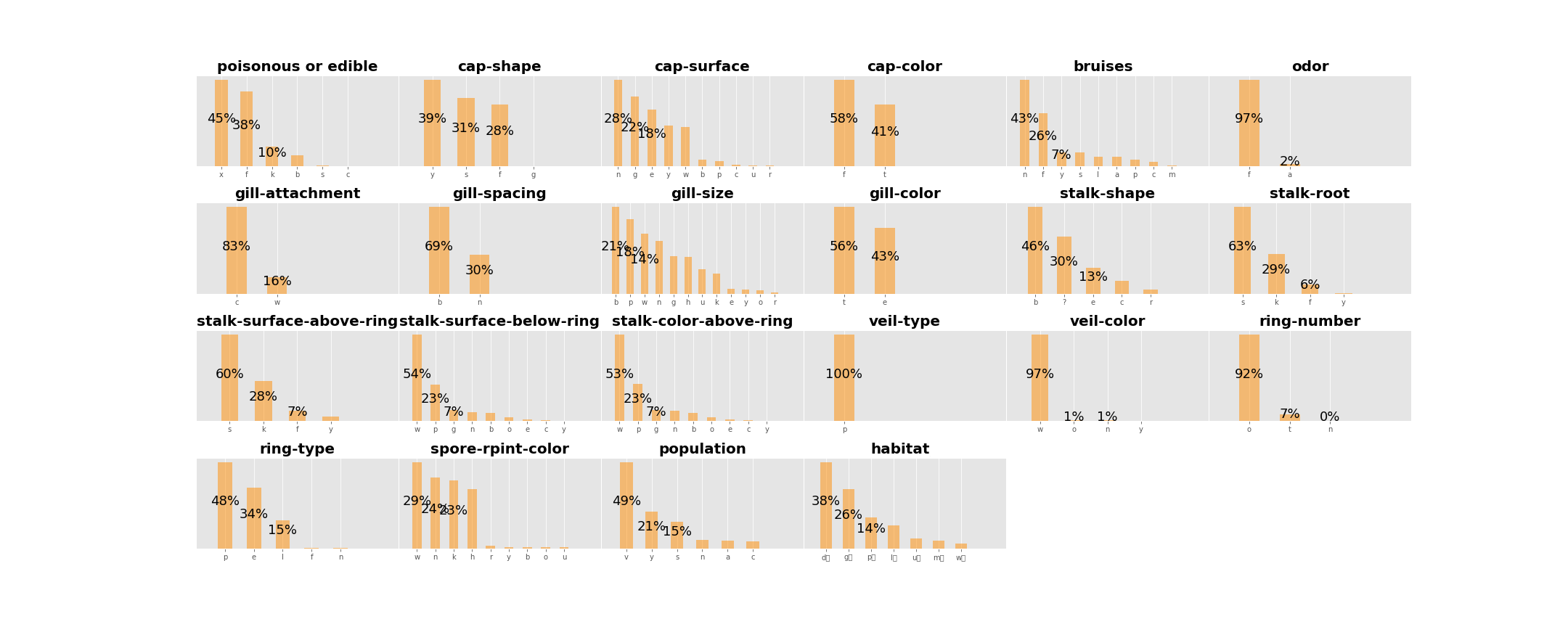
## Mushrooms

Link to the dataset: <http://archive.ics.uci.edu/ml/datasets/Mushroom>

The mushroom dataset consists of 8124 samples of gilled mushrooms in the Agaricus and Lepiota Family. They are classified into 2 classes: edible or poisonous, samples of unknown edibility are classified as poisonous as well.

A brief overview of the Mushroom Dataset:

* 22 attributes (all nominal)
* 2 classes
* Only nominal values (also missing values)
* 4208 edible and 3915 non edible



## Congress

## Amazon

# Methods

# Conclusion

Wine:

When analysing the wine dataset, we found out that working with human intuition works really well. Therefore besides some standard machine learning methods, we also used our knowledge about wine to improve the results. According to the knowledge what we already had a priori, there seemed to be a really strong connection between the target value quality and some of the chemical values. For example a high acidity is just not wanted for red wines and therefor wine samples that had a high acidity, where rated poorly by the experts.

Since the dataset only consist of numerical values which unfortunately have a quite high variance in range we focused our pre-processing on scaling those values. Feature selection was not needed in case of the wine dataset since there are only eleven. Like in the last exercise we used three different pre-processing:

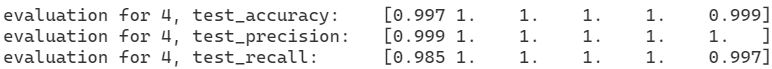
1. Standard dataset without pre-processing
2. Z score scaling
3. Min max scaling
4. Self-made weighted scaling method

The numbers one to four can also be seen in the result matrices. We didn’t want to drop any feature, since the result of all three methods gradually benefited from a higher number of features. Since we could not drop any feature on the one hand, but knew, that some features are just not desirable we developed a scaling method basted on the feature correlation. For each feature, we divided the feature by their maximum to have a range between zero and one and then multiplied by their corresponding correlation to get a somehow weighted scaling:

This pre-processing method far outperformed all others and yielded the best result for all algorithms.

As mentioned in the beginning of this section the feature value give a quite good idea of which class the wine will be classified to. This will be seen in the following results. Using the just descripted scaling method, we achieved a perfect score of accuracy, precision and recall equal to one for two out of the three used methods.

The results of kNN were the “worst” of all three methods. But it has to be made clear, that worst in this case means, not a perfect score! When using five-fold cross validation. kNN had an impressive mean accuracy of 0.999.

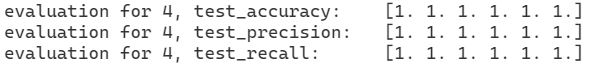
The best parameters of kNN were: k=1, and uniform weights. Choosing k < 5 did not have a great impact on the results, bigger k’s lead to a much worse result. Below one can see the results for accuracy, precision and recall using our hand-made pre-processing method.

Note on the figure:

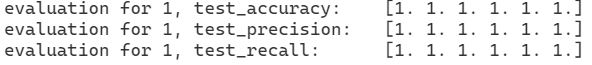
We used cross-validation with 5 splits. Thus, the first 5 columns are the results of the respective split and the last one is the mean.

(The parameters used are k=1, weights=uniform, algorithm=auto.)

Using Support Vector Machines the results were better than kNN. When using the weighted scaling approach with SVM we could achieve a perfect score of one for each split and all three scores. SVM achieved the best results using a polynomial kernel and polynomial degree of four. The best pre-processing was again the weighted scaling approach. Min Max scaling had quite impressive results as well, with 3 splits being one and split one and four having accuracy of over 0.99. Z score scaling could not compete at all (mean accuracy = 0.79)

Note on figure: SVM with: gamma=scale, kernel=poly and degree=4, (using weighted scaling)

The method random forest yielded perfect results as well. Opposed to the other two methods the fourth pre-processing did not yield the best result, but interestingly enough the not processed dataset has a perfect compared to the other three pre-processing methods only achieving a perfect score in four out of five splits. The results of the pre-processed set were nearly identically good. (accuracy 0.99, precision 0,99 and recall 0.99). We tried out a lot of to find the optimal one. The parameters max depth, min\_sample\_split and min\_sample\_leaf did not improve the results, so that finally only n\_estimators and max\_features, were used.



Note on figure: Random forest: max\_features=6, n\_estimators=100, (not processed dataset)

To conclude, it seemed that like the mushroom dataset, also the wine dataset is quite easy to predict. As stated on the beginning we assume that this is due to the strong connection of the some features with the target value. When fine tuning the parameters and using the best fitting pre-processing method, all three machine learning methods could achieve an impressive score. kNN was the only method not having a perfect score.