UNIVERSITÄT ZU LÜBECK
INSTITUTE FOR ROBOTICS
AND COGNITIVE SYSTEMS

Artificial Intelligence II
Winter term 2020/2021
January 15, 2021

Daniel Wulff
wulff@rob.uni-luebeck.de

Exercise sheet 9

Submission deadline: 10:00, January 22, 2021

Task 1: How good is your wine? (20 points)

This exercise is a hands-on task on real world data! In 2009, Cortez et al. published a data set to use data mining for wine quality estimation (P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis. Modeling wine preferences by data mining from physicochemical properties. In Decision Support Systems, Elsevier, 47(4):547-553, 2009.). Thus, 1599 different variants of the Portuguese "Vinho Verde" wine were examined for their physiochemical properties. Additionally, professional wine testers scored each wine's quality on a scale between 0 and 10. The 11 measured physiochemical attributes are:

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

The question arising with this data set: Is it possible to distinguish good from not that good wines just based on their physiochemical properties? Use an SVM classifier to answer this question! Assume that a wine is good if its score is seven or higher.
Download the given MATLAB code snippets from the Moodle course.

- `Data.mat`

  This .mat file contains the data set. *SensorData* is an $n \times m$-Array, where n is the number of data points (i.e. the number of examined wines) and m is the number of features (i.e. the number of performed physiochemical tests).

- `ClassifyLinear.m`

  The script performs a linear classification. The classes are "good wine" and "not good wine". A wine is good if it's score is 7 or higher. Fill the gaps in this script!

- `ClassifyNonlinear.m`

  The script performs a nonlinear classification. The classes are "good wine" and "not good wine". A wine is good if it's score is 7 or higher. Fill the gaps in this script!

a) Implement the missing functionality in `ClassifyLinear.m`. At first, identify the input X and the class labels y of the learning model. Then implement a 5-fold cross validation (Note: 'Implement' does not mean 'use the crossvalidation functionality of MATLABs `fitcsvm` function! Set it up on your own!). Keep in mind that the balance between the number of samples from both classes should be similar in all subsets. Finally, calculate the mean prediction accuracy over all folds. The prediction accuracy is the percentage of right classified points in the test set. (7 points)

b) Implement the missing functionality in `ClassifyNonlinear.m`. At first, identify the input X and the class labels y of the learning model. Then implement a 5-fold cross validation (Note: 'Implement' does not mean 'use the crossvalidation functionality of MATLABs `fitcsvm` function! Set it up on your own!). Keep in mind that the balance between the number of samples from both classes should be similar in all subsets. Finally, calculate the mean prediction accuracy over all folds. The prediction accuracy is the percentage of right classified points in the test set. (7 points)

c) Search for good hyperparameters (i.e. SVM-parameter C and Kernel-parameter $gamma$) of the SVM classifier. Evaluate the classifier performance for each parameter combination of $C \in \{1, 10, 100\}$ and $\gamma \in \{0.1, 1, 10\}$. (4 points)

d Discuss your results. Is it possible to distinguish good wines from the rest based on the given measurements? Which classifier gives the best performance? What are limitations of your study? (2 points)