André Artelt, Johannes Brinkrolf Bielefeld University

Introduction to Machine Learning (WS 2019/20)
3. Assignment

Released: Friday, 01.11.2019.
Due: Please solve the exercises in groups of three and submit your report by mail to your tutor or into post box 329 on CMD 5 in the main building by **Friday, 06.12.2019**, **11:59am**.

- Write all names of your group members and the tutor's name on the first page (have a look at given template).
- Mark each paragraph with the initials of the responsible group member. You find a new command `\initials{}`, therefore, in the template.
- The python toolbox needed for solving the practical exercises can be found at `http://scikit-learn.org/stable/`. It comes with an excellent online description and demos. You can also look at the given demo code.
- We recommend to install `anaconda` for this lecture because it install automatically all required package, namely `numpy`, `scipy`, `matplotlib`, `sklearn`, `jupyter`, and `Spyder` as a develoment environment.
- You have five weeks time for this sheet . Also there is a tutorial on Monday, 25.11.2019 where no other sheet will be discussed. You can use this session to work on this sheet and/or ask your tutor if you get into trouble.
- If you have any questions, please ask your tutor or write an email to intromachlearn@techfak.uni-bielefeld.de.

---

1 Weird Strings

(*20 Points*)

In this exercise you work with two given classification data sets[1] and the goal is to build and train a ML model (classifier that assigns a label to a given string) that performs as well as possible. The given input are strings of different lengths. This is why you have to apply some pre-processing and extract features from raw data.

We propose to use the occurrence of bigrams as features. A bigram is a sequence of two adjacent letters from a string. For example in a small alphabet with three letters $\mathcal{A} = \{A, B, C\}$ all possible bigrams are $\{AA, AB, AC, BA, BB, BC, CA, CB, CC\}$. Then the according feature vector is a 9 dimensional vector of 0 and 1 indicating whether the bigram occur in the string or not. The feature vector for the string $ABCABCAB$ would be $[0, 1, 0, 0, 0, 1, 1, 0, 0]$ because the three bigrams AB, BC, and CA occur in the string.
Implement the bigram based feature computation and try out *three* different classifier models. You can assume that only capital letters from the English alphabet (26 letters A - Z) are used in the data set.

Your task is then to write down a short report which should include at least following sections:[2]

(a) *(5 Points)* Introduction and description of the given data - e.g. analyze the given data sets and answer the following questions: How many samples? How many classes? Which features are used? ...

(b) *(5 Points)* Describe shortly the classifiers you used (e.g. How does the classifier distinguish classes? How are hyperparameters chosen? Which hyper-parameters were used? ...)

(c) *(5 Points)* Show and describe your results (e.g. accuracy on train and test set) by using plots or tables with test errors. Note that accuracy is not the only measure you can use - think about balanced and unbalanced data sets.

(d) *(5 Points)* Discussion (Try to explain and discuss the results. Any further ideas?)

Prepare your results in a **full-text** report on two or three pages. We provide a LaTeX template `report_template.tex` which might help you.

You can come with further features which enable better results. Explain the features and compare them to the others in the report. This will give you up to *10 bonus points*.

---

[1] `dataset1.npz` and `dataset2.npz`: You can load the data using the commands `data = np.load("dataset1.npz")`, `X=data["X"]`, and `y=data["y"]`

[2] You might want to use this ordering for working on the task.