Exercise 2 - Decision Theory and Density Estimation

Winter term 2019/2020

student1, student2, student3

General information

The assignments are voluntary. All students who choose to participate have to form groups comprising three to four students (not more and not less). The groups do not have to be static, you may form new groups for each assignment. You have **two weeks** to answer the questions and to submit your work. The solutions are going to be presented and discussed after the submission deadline. Sample solutions will **not** be uploaded. However, you are free to share correct solutions with your colleagues **after they have been graded.**

Formal requirements for submissions

Please submit your solutions via Moodle (as a .zip file) as well as in printed form. The .zip file must contain one .pdf file for the pen-and-paper tasks as well as one .py file per programming task. Only pen-and-paper tasks have to be printed, you do not have to print the source code. Only one member of the group has to submit the solutions. Please make sure to specify the matriculation numbers (**not the names!**) of all group members so that all participants receive the points they deserve!

Please refrain from submitting hand-written solutions or images of solutions (*.png* / *.jpg* files). Rather use proper type-setting software like LaTeX or other comparable programs. If you choose to use LaTeX, you may want to use the template files provided.

Code assignments have to be done in Python. Please submit *.py* files (**no jupyter notebooks**). The following packages are allowed for code submissions: `numpy`, `pandas` and `scipy`. Please ask **beforehand**, if you want to use a specific package not mentioned here. Finally, do not use already implemented models (e.g. from `scikit-learn`).

Grading details

Your homework is going to be corrected and given back to you. Correct solutions are rewarded with a bonus for the exam which amounts to at most ten percent of the exam, if all solutions submitted by you are correct (this corresponds to at most six points in the exam). It is still possible to achieve full points in the exam, even if you choose not to participate in the assignments (it is additional). The function which is used to compute the bonus is given by:

$$b(a) = \min\left(B, \left\lceil \frac{B}{A^2} \cdot a^2 \right\rceil\right) \tag{1}$$

- b denotes the number of bonus points you get for the exam (this is up to you)
- B refers to the maximum attainable bonus points for the exam (six points)
- A denotes the maximum attainable points in the assignments (40 points)
- a is the score you achieved in the assignments (this is up to you)

Please note: You have to pass the exam **without the bonus points!** This means that it is not possible to turn a failing grade ($= 5.0$) into a passing grade (≤ 4.0). The bonus points will be taken into account in case you have to repeat the exam (i. e. they do not expire if you fail the first attempt).

Important!

The solutions have to be your own work. If you plagiarize, you will lose all bonus points!

1 Bayesian Decision Theory

a) Bayes' Rule (1 point)

State Bayes' rule and state the name of each term in the equation.

Solution:

b) Decision Boundary (1 point)

Which condition holds at the optimal decision boundary? In a binary classification problem, when do we prefer class A over class B? Why is it not necessary to normalize the probabilities on both sides of the inequality?

Solution:

c) Naïve Bayes (5 points)

You are planning a nice trip to the forest to collect some delicious mushrooms. However, you are not an expert for mushrooms and afraid of picking poisonous ones. However, you have a data set called `mushrooms.csv` describing the shape, color and habitat of different mushrooms and whether they are edible (type *e*) or poisonous (type *p*). Implement a binary naïve Bayes classifier using `Python` and `numpy` / `pandas` and train it on the mushrooms data set. Use 10 % of the data set as test set and report your accuracy on this test set.

Solution:

d) Bonus Question (1 point)

You work for a machine learning startup which specializes in text classification for automatic scam detection in social networks. You are required by law to explain in detail why your system did not filter out content which was scam or why it did filter out normal content. Given that both models are suitable, would you prefer a naïve Bayes model or a deep neural network? Why?

Solution:

2 Density Estimation

a) Non-Parametric Density Estimation (3 points)

You want to estimate the density for the data stored in the file `density_data_train.csv`. Implement either a kernel density estimator using a Gaussian kernel trying different values for h **or** a k-nearest neighbors estimator trying different values for k. Which value for h or k works best? Plot the densities in a suitable interval.

Solution: