Exercise 4 - Backpropagation and unsupervised Learning

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 Backpropagation

a) Backpropagation by Hand (5 points)

You are given the neural network depicted below. Compute one forward-pass and one backward-pass based on the labeled training example $\langle \boldsymbol{x} = [0.05, 0.10], \boldsymbol{y} = [0, 1] \rangle$. Employ the squared error loss function:

$$\mathcal{J}(\boldsymbol{\Theta}) = (y_{pred} - y)^2$$

$$\mathcal{J}'(\boldsymbol{\Theta}) = 2 \cdot (y_{pred} - y)$$

The weight matrices and bias weights are initialized as follows (where for instance weight $\Theta_{12}^{(1)}$ connects the input x_2 with the first neuron in the hidden layer):

$$\boldsymbol{\Theta}^{(1)} = \begin{pmatrix} 0.15 & 0.20 \\ 0.25 & 0.30 \end{pmatrix} \qquad \boldsymbol{\Theta}^{(2)} = \begin{pmatrix} 0.40 & 0.45 \\ 0.50 & 0.55 \end{pmatrix}$$

$$\boldsymbol{b}^{(1)} = \begin{pmatrix} 0.35 & 0.35 \end{pmatrix} \qquad \boldsymbol{b}^{(2)} = \begin{pmatrix} 0.60 & 0.60 \end{pmatrix}$$

Use the ReLU and sigmoid activation functions in the hidden layer and output layer, respectively. Write down all necessary computations. What are the gradients for the weights $\Theta_{12}^{(1)}$, $\Theta_{21}^{(2)}$?

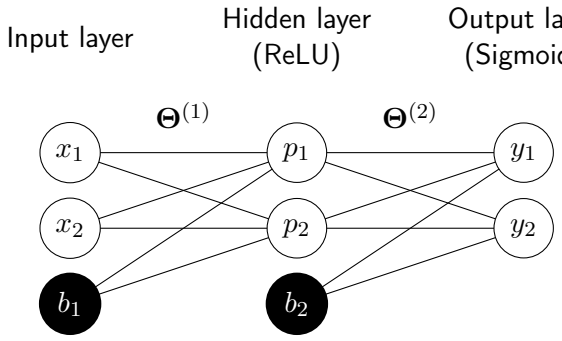

Figure 1: MLP architecture

Solution:

2 Unsupervised Learning

a) k-Means clustering (4 points)

Implement k-Means clustering for image compression. Compress the exemplary image file which can be found under the path `/data/dhbw.jpg`. You can find an explanation of how image compression with k-Means works on this web page.[1]

Solution:

b) PCA (1 point)

Explain how to choose the number of principal components for dimensionality reduction. Why does this work?

Solution:

[1] https://medium.com/@agarwalvibhor84/image-compression-using-k-means-clustering-8c0ec055103f

c) Bonus Question: Spectral clustering (1 point)

How can you automatically choose the number of clusters for spectral clustering?

Solution: