Deep Learning Praktikum

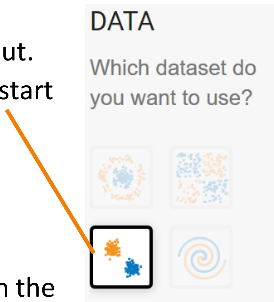
Klaus Dorer – Hochschule Offenburg

> You can do exercise 1 individually on one of the computers in the lab by starting a BW-Lehrpool virtual environment and opening a browser.

1 Backpropagation Networks

These exercises are performed in the Tensorflow Playground: https://playground.tensorflow.org

1.1 Data



There are four datasets available each of which has a two dimensional input. The network should classify orange points as orange, blue as blue. Please start with the left bottom data set.

You start, stop and reset learning with the control buttons

The playground shows a visual representation of the output on the right in the output area. It also shows the test loss, i.e. how well the network classifies unknown data, or better to say how small the classification error (loss) still is.

1.2 Tasks

Activation Function

- Please choose activation->Linear. What is the minimum size of the network to learn the chosen input pattern?
- Reset the network (2 Hidden Layers, 4, 2 Neurons). Which of the input patterns can be learned with the linear activation function?
- Now test each pattern with each activation function.

Network Size

Please choose the input pattern on bottom right (spiral).

- Try to find out how the number of neurons and the number of layers influences the learning success.
- Try to achieve the best possible result (test loss) for the spiral pattern with all means available on the page.

For exercise 2 you have to open the Jupyter environment. It can be started by opening a terminal and typing ./startJupyter.sh

2 Digit Recognition

One of the first applications of image recognition and nowadays the "hello world" of deep neural networks is handwritten digit recognition.

2.1 Data

The MNIST dataset contains 70.000 images of handwritten digits. It is integrated into the Keras environment and will be downloaded during first learning.

2.2 Tasks

After starting the Jupyter server, a web page in the browser opens. Navigate to
DeepLearning/MnistClassification/MnistClassifier.ipynb
and open the Book by double clicking it. A new browser tab opens.

- Work through the cells one by one by pressing Shift-Enter
- How does the result change, if you use also the inverse (b/w) images (switch on in section Constants)?
- How does the result change, if you use a DENSE_MODEL (Backpropagation instead of CNN) as MODEL_TYPE? What else changes?

For exercise 3 we will work in breakout groups. We share the results in this google doc:

https://docs.google.com/spreadsheets/d/1lKltSpu_ilfDZhjW3zcezNnuO5Jo4t6aPLOKwVoWhsw/edit?usp=sharing

3 Recognize RoboCup Objects

We will now use Deep Convolutional Networks to detect objects on images of our soccer-playing robot Sweaty.

3.1 Data

- The images are already available in folder images. Folder training contains the images to use during learning. Folder validation contains images to measure accuracy on unknown images during learning. Folder test contains images to measure accuracy in the end. Notebook MagmaDataset.ipynb already did this for us; it is not required to run here. The learning data are images from the RoboCup domain and shows eight categories of objects.
- There are 4917 color images available
- Images may be augmented (rotation, translation, shearing, scaling, mirroring)

Klasse	**Kategorie**
0	Ball
1	Goalpost
2	Obstacle
3	L-Line
4	X-Line
5	T-Line
6	Penalty Spot
7	Foot

3.2 Tasks

You can continue to work with the running Jupyter Notebook server if still running.

Please open the Jupyter Notebook DeepLearning/HsoMagmaClassification/MagmaClassifier.ipynb by clicking on it. A new browser tab opens.

Distribute the single rows of each tab of the google doc mentioned above among your breakout group. At the end we will have one graph per group. Make sure to set/reset all settings to the default values before changing the parameter under evaluation!

Evaluate the following:

- How does the learning success improve with an increasing number of epochs?
- How does learning success improve with the number of images presented at learning? Systematically increase the number of training images using parameter validation_split and note down the results in the excel result sheet DeepLearning_Magma in Tab Training Data.
- How does learning success depend on the optimizer used for gradient descent? Run learning with all optimizers and note down the result in the Excel file. Do not forget to reset validation_split back to its default value.
- How does the batch size influence the learning success?

If there is still time:

- Increase/Decrease the dense layers
- Switch on augmentation
- Remove or add network layers
- …