Mathematics of Learning – Worksheet 13

- The exercise sheets will be uploaded every Monday. Solution sketches will be uploaded one week later.
- You can hand in your own solutions via StudOn and we correct them - this is not mandatory. Please hand in small groups of 2-3 students.
- For questions, please use the forum on StudOn since other students may have similar questions. If you have a more personal question about the exercises please send an email to ehsan.waiezi@fau.de or lars.weidner@fau.de respectively.

Exercise 1 [Reading].
Read Section 11.5 in "The elements of statistical learning" (Hastie book).
Read the paper "Understanding Adversarial Training: Increasing Local Stability of Neural Nets through Robust Optimization" (shaham-paper.pdf on studon). Try to understand the proposed training in Section 4 of this paper.

Exercise 2 [Application of neural networks].
Continue your implementation of a neural network. Choose an appropriate network setting (number of hidden layers, number of neurons in each layer) and train it to classify the wine data set provided a few sheets earlier (to apply a regression model). Visualize, discuss and interpret your results. Compare the neural network results to the linear regression results.

Exercise 3 [Attack your neural network].
Continue your implementation of a neural network. This exercise makes sense, if you can achieve a prediction accuracy of $> 90\%$ for the mnist hand written digits data set. In this exercise we want to manipulate the input of a (fully trained) neural network, such that it produces results which are as erroneous as possible. We want to produce both, false-positive as well as false-negative results.

a) Describe, how the gradient $\nabla_x C(x, w, b)$ can be obtained as a by-product of the backprop routine. Adapt the backprop routine to obtain this gradient as a part of the output.

b) Implement a routine, which manipulates hand written digit images in a way, that

- You (human) can still correctly classify the image
- The trained neural network fails in at least 50% of all cases.

Utilize a).

c) Force the neural network to “draw” an image of a number. Implement a routine, which gets a number, a start image (vector, e.g., random normal entries), a “generating rate” (a positive real, similar role to learning rate), and an iteration limit (natural

number), to manipulate the start image, such that the neural network classifies it as the number which is passed. Utilize a). *With very high confidence, you will get out nonsense-images, for which the neural network is very confident that they depict some numbers.*
Bonus Generate a labelled data set of your "drawn" data set, and exchange a fellow student's data set. Let your neural network classify the data set of your fellow student (and the other way round) and report the prediction accuracy.

d) Repeat b), but only use feedforward as a subroutine (an adversary may not have access to the weights, biases and backtrack routine of the neural network).

Exercise 4 [Basic reformulations of robust problems].
Let an uncertain constraint of the form

$$(\bar{a} + Pu)^T x \geq b \text{ for all } u \in \mathcal{U} \tag{1}$$

with $\bar{a} \in \mathbb{R}^n$, $b \in \mathbb{R}^m$, $P \in \mathbb{R}^{n \times q}$ be given. Reformulate this inequality to a system of deterministic inequalities in the case of

1. $\mathcal{U} = \{u \ : \ Du \geq d\}$ with some matrix $D \in \mathbb{R}^{p \times q}$ and vector $d \in \mathbb{R}^p$,
2. $\mathcal{U} = \{u \ : \ ||u||_\infty \leq \rho\}$ with $\rho > 0$.