Institut for Theoretical Physics
University Leipzig

Prof. Dr. B. Rosenow
A. Afanah , M. Staats

Statistical Mechanics of Deep Learning - Problem set 7

Winter Term 2023/24

Hand in: Friday, 01.12 at 10:00 am, you can upload your solutions to the course webpage on Moodle platform.

14. The Perceptron rule

6 Points

Consider the update of coupling vectors according to the Perceptron rule (see section 3.2 in the book of A. Engel et al)

$$\mathbf{J}^{(\text{new})} = \begin{cases} \mathbf{J}^{(\text{old})} + \frac{1}{\sqrt{N}}\xi^{\mu}\sigma_T^{\mu} & \text{, If } \mathbf{J}^{(\text{old})}\xi^{\mu}\sigma_T^{\mu} < 0 \\ \mathbf{J}^{(\text{old})} & \text{, otherwise.} \end{cases}$$

An interesting aspect of the perceptron rule is that it may easily be modified for the numerical analysis of Gibbs learning. Modify the couplings in order to reproduce the Gibbs learning generalization error, Fig. (1.4) A. Engel et al. This can be done by adding a random contribution with zero mean with each update of the couplings.
Hints: Making the random contribution smaller than in the book increases the speed of the algorithm. Choose a Gaussian distribution with variance $\simeq 10/\sqrt{N}$. Choose $N \leq 100$ for computational speed and average over 3 learning runs to obtain a smooth curve. Drawing random vectors from the N-sphere is important for this exercise. The following code might help:

```
# function that returns an N-dim vector from the N sphere
def N_sphere_vector( N ):
    vector = np.random.normal(0,1, N)
    vector /= np.linalg.norm( vector)
    vector *= np.sqrt(N)
    return vector
```

15. The Adatron rule

6 Points

The Adatron algorithm combines the the selectivity of the perceptron rule with the adaptive step size of adaline rule (see section 3.5 A. Engel et al), it uses the updates

$$\delta x^{\mu} = \begin{cases} \gamma(1-\Delta^{\mu}) & \text{, If } \Delta^{\mu} < 1 \\ -x^{\mu} & \text{, otherwise.} \end{cases}$$

with x^{μ} denoting the embedding strengths, γ is the learning rate and Δ^{μ} being the stability or aligning field as defined in the lecture. Implement the adatron learning rule numerically and determine its asympototic ($\sim \frac{0.5005}{\alpha}$) behavior.