Machine Learning Exercise Sheet 8

Deep Learning II

Exercise sheets consist of two parts: In-class exercises and homework. The in-class exercises will be solved and discussed during the tutorial. The homework is for you to solve at home and further engage with the lecture content. There is no grade bonus and you do not have to upload any solutions. Note that the order of some exercises might have changed compared to last year's recordings.

In-class Exercises

Problem 1: See notebook `exercise_inclass_08_pytorch.ipynb` on Moodle.

Homework

Problem 2: You are trying to solve a regression task and you want to choose between two approaches:

1. A simple linear regression model.
2. A feed forward neural network $f_{\boldsymbol{W}}(\boldsymbol{x})$ with L hidden layers, where each hidden layer $l \in \{1, ..., L\}$ has a weight matrix $\boldsymbol{W}_l \in \mathbb{R}^{D \times D}$ and a ReLU activation function. The output layer has a weight matrix $\boldsymbol{W}_{L+1} \in \mathbb{R}^{D \times 1}$ and no activation function.

In both models, there are no bias terms.

Your dataset $\mathcal{D}$ contains data points with nonnegative features $\boldsymbol{x}_n$ and the target y_n is continuous:

$$\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^N, \qquad \boldsymbol{x}_n \in \mathbb{R}^D_{\geq 0}, \qquad y_n \in \mathbb{R}$$

Let $\boldsymbol{w}^*_{LS} \in \mathbb{R}^D$ be the optimal weights for the linear regression model corresponding to a *global* minimum of the following least squares optimization problem:

$$\boldsymbol{w}^*_{LS} = \arg\min_{\boldsymbol{w} \in \mathbb{R}^D} \mathcal{L}_{LS}(\boldsymbol{w}) = \arg\min_{\boldsymbol{w} \in \mathbb{R}^D} \frac{1}{2} \sum_{n=1}^N (\boldsymbol{w}^T \boldsymbol{x}_n - y_n)^2$$

Let $\boldsymbol{W}^*_{NN} = \{\boldsymbol{W}^*_1, \dots, \boldsymbol{W}^*_{L+1}\}$ be the optimal weights for the neural network corresponding to a *global* minimum of the following optimization problem:

$$\boldsymbol{W}^*_{NN} = \arg\min_{\boldsymbol{W}} \mathcal{L}_{NN}(\boldsymbol{W}) = \arg\min_{\boldsymbol{W}} \frac{1}{2} \sum_{n=1}^N (f_{\boldsymbol{W}}(\boldsymbol{x}_n) - y_n)^2$$

a) Assume that the optimal $\boldsymbol{W}^*_{NN}$ you obtain are non-negative.
What will the relation ($<, \leq, =, \geq, >$) between the neural network loss $\mathcal{L}_{NN}(\boldsymbol{W}^*_{NN})$ and the linear regression loss $\mathcal{L}_{LS}(\boldsymbol{w}^*_{LS})$ be? Provide a mathematical argument to justify your answer.

non negotive W and x $ReLU(xW) = xW$

$f_{W^*_{NN}}(x_i) = ReLU(ReLU \cdots ReLU(x_i^T w_1^*) w_2^* \cdots W_{L+1}^*)$

$= x_i^T W_1^* W_2^* - W_{L+1}^*$

$= x_i^T W^*$ $\Longleftrightarrow$ behave like linear regression with different W_{NN}^*

Linear regression is a special case above NN

$\Rightarrow L_{NN}(W_N^{**}) = L_{LS}(W_{LS}^*)$ $W_{LS}^* = W_{NN}^*$

---

b) Above case is a special case.

W_{LS}^* is non negative tell nothing about W_{NN}^*

So with complex arctiecture, NN can find a better optimal

$\Rightarrow L(W_{NN}^*) < L(W_{LS}^*)$

b) In contrast to (a), now assume that the optimal weights $\boldsymbol{w}^*_{LS}$ you obtain are non-negative. What will the relation ($<, \leq, =, \geq, >$) between the linear regression loss $\mathcal{L}_{LS}(\boldsymbol{w}^*_{LS})$ and the neural network loss $\mathcal{L}_{NN}(\boldsymbol{W}^*_{NN})$ be? Provide a mathematical argument to justify your answer.

Problem 3: Load the notebook `exercise_08_notebook.ipynb` from Moodle. Fill in the missing code and run the notebook. Export (download) the evaluated notebook as PDF and add it to your submission.

Note: We suggest that you use Anaconda for installing Python and Jupyter, as well as for managing packages. We recommend that you use Python 3.

For more information on Jupyter notebooks, consult the Jupyter documentation. Instructions for converting the Jupyter notebooks to PDF are provided on Piazza.