

In this exercise we consider Bayesian Neural Networks (BNNs) with pSGLD as the optimizer and ensembles.

1. BNNs

- (a) Answer the theoretical questions in 8.1 and 8.1.1.1. (3)
- (b) Implement the `step()` function for pSGLD. (2)
- (c) Answer theoretical question in 8.1.2. (1)
- (d) Implement the functions for the BNN class. (3)
- (e) Plot the sinc function. (1)
- (f) Answer the theoretical questions in 8.1.4. (3)
- (g) Implement the code in `NLLWithPriorLoss` and `HomoscedasticLogVar`. (2)
- (h) Implement the evaluation and training code in 8.1.5. (2)
- (i) Implement the plotting code and answer the theoretical questions in 8.1.6. (5)
- (j) For course improvements, we would like your **feedback about *this* question**. At least, tell us how much time (**hours**) you invested, if you had major problems and if you think it's useful.

Points for Question 1: 22

2. Ensembles

- (a) Implement the code and plot uncertainties for an ensemble of NNs in 8.2. (3)
- (b) For course improvements, we would like your **feedback about *this* question**. At least, tell us how much time (**hours**) you invested, if you had major problems and if you think it's useful.

Points for Question 2: 3

You can achieve a total of **25 points** for this exercise. Additionally, you can achieve **1 bonus point** for answering the feedback questions.

Please send the **solution *notebook* of your group of three** via ILIAS until **14.01.2019 (four weeks) 9 pm**.

Note: Jupyter notebooks will be executed **from top to bottom**. To avoid point deduction check your notebook by the following steps: 1. Use the python 3 kernel (`Kernel > Change kernel > Python 3`), 2. Run the full notebook (`Kernel > Restart & Run All`), 3. Save (`File > Save and Checkpoint`).