Norwegian University of Science and Technology

DT8122 — **Project Assignment - Uncertainty Quantification**

PROBABILISTIC AI

dt8122@idi.ntnu.no

Summer 2019

1 Introduction

Uncertainty modeling and quantification is relevant in many real world applications. One possible approach to this task is to use Bayesian methods in order to obtain the probability density of the quantities of interest and use the obtained full densities to compute other quantities of interest such as *prediction intervals*[1]. Traditionally, we would resort to linear models in order to have a computationally feasible solution to the inference problem. However, with the recent advances in Probabilistic AI, we are able to extend the space of possible models with non-linear models. The objective of this project assignment is to explore some of those techniques in a set of prediction tasks.

2 Task

The task concerns generation of prediction intervals (PIs) along with point predictions by applying neural networks for quantifying uncertainty in regression tasks.

First, we ask you to choose and implement two recent deep generative modeling techniques from Section 2.1. Second, we ask you to identify shortcomings or limitations of the implemented techniques. Finally, we ask you to implement some possible improvements (see Section 2.2).

Each part should be accompanied by empirical evaluation (see Section 2.4) using the provided datasets (see Section 2.3) and discussion of the results. This all compiled into a final project report (max. 10 pages) accompanied with the code to reproduce your results.

[1] `https://en.wikipedia.org/wiki/Prediction_interval`

2.1 Models

You are expected to implement a deep probabilistic regression model for uncertainty estimation and use this model to generate prediction intervals and point predictions. For that you have to choose two of the following techniques:

- Dropout as a Bayesian Approximation [2],
- Bayes by Backprop [1],
- Multiplicative Normalizing Flows [4],
- Bayes by Hypernet [3, 5].

2.2 Research of Shortcomings, Limitations and Possible Improvements

Given you have implemented the two selected approaches, discuss their respective shortcomings or limitations and implement at least one possible improvement. Compare with previous results and report your findings.

2.3 Datasets

The provided datasets are the same as in Gal and Ghahramani [2] and Pearce et al. [6]. All datasets have target values (y) placed in the last column.

2.4 Evaluation

For evaluation of your models use the tail 10% of a dataset as a test set. Measure at least root mean squared error (RMSE) for point predictions, and prediction interval coverage probability (PICP) and mean prediction interval width (MPIW) for 95% prediction intervals.

Following the notation from Pearce et al. [6], let $\mathbf{x}_i \in \mathbb{R}^D$ be the ith D dimensional input features corresponding to target observation y_i, where $1 \leq i \leq n$ for n data points. The predicted lower and upper bounds of PI are denoted by $\hat{y}_{Li}$ and $\hat{y}_{Ui}$. The PI is then defined as

$$P\left(\hat{y}_{Li} \leq y_i \leq \hat{y}_{Ui}\right) \geq \gamma,$$

where γ is 0.95 for 95% prediction interval.

PICP is calculated as following:

$$k_i = \begin{cases} 1 & \text{if } \hat{y}_{Li} \leq y_i \leq \hat{y}_{Ui}; \\ 0 & \text{otherwise,} \end{cases}$$

$$c = \sum_{i=1}^{n} k_i,$$

$$PICP = \frac{c}{n}.$$

MPIW is calculated as following:

$$MPIW = \frac{1}{n} \sum_{i=1}^{n} \hat{y}_{Ui} - \hat{y}_{Li}.$$

3 Submission Requirements

We expect you to submit the following:

- **Code** with your implementation in Python using Pyro or PyTorch of two of the four techniques of uncertainty modeling with neural networks.
  - You can have the code in a Jupyter notebook, however you should make sure the code is not dependent on a system specific configuration. To make sure that we can execute your code, provide `requirements.txt` file, *Dockerfile* or test it in Google Colab[2].
- **Report** that should include your model choices, inference methods, results from empirical evaluation, findings and discussion.

All assignment artifacts are to be sent as a ZIP file to the email address `dt8122@idi.ntnu.no`. The deadline is 12 July 2019 AoE (Anywhere on Earth).

References

[1] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1613–1622, Lille, France, 07–09 Jul 2015. PMLR.

[2] `https://colab.research.google.com`

[2] Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059, New York, New York, USA, 20–22 Jun 2016. PMLR.

[3] David Krueger, Chin-Wei Huang, Riashat Islam, Ryan Turner, Alexandre Lacoste, and Aaron Courville. Bayesian hypernetworks. *arXiv preprint arXiv:1710.04759*, 2017.

[4] Christos Louizos and Max Welling. Multiplicative normalizing flows for variational Bayesian neural networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 2218–2227, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.

[5] Nick Pawlowski, Martin Rajchl, and Ben Glocker. Implicit weight uncertainty in neural networks. *CoRR*, abs/1711.01297, 2017.

[6] Tim Pearce, Alexandra Brintrup, Mohamed Zaki, and Andy Neely. High-quality prediction intervals for deep learning: A distribution-free, ensembled approach. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4075–4084, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR.