CS-E4820 Machine Learning: Advanced Probabilistic Methods

Pekka Marttinen, Paul Blomstedt, Homayun Afrabandpey, Reza Ashrafi, Betül Güvenç, Tianyu Cui, Pedram Daee, Marko Järvenpää, Santosh Hiremath (Spring 2019)

Exercise problems, round 6, due on Tuesday, 12th March 2019, at 23:55

Please return your solutions in MyCourses as a single PDF file.

Problem 1. *"Deriving VB for a simple model, part 1."*

Consider the variational Bayesian approximation for the example model from the lecture (see `simple_vb_example.pdf` in the materials of lecture 6). Derive the VB update for the factor $q(\tau)$ in the example and complete the code blocks 'Update of factor q(tau)' and 'Current estimate of tau' in `ex6_12_template.py`. In your answer, give both your analytical derivations and the completed code blocks.

Problem 2. *"Deriving VB for a simple model, part 2."*

As in Problem 1, consider the variational Bayesian approximation for the example model from the lecture (`simple_vb_example.pdf`). Now, derive the VB update for the factor $q(\theta)$ in the example and complete the code blocks 'Update of factor q(theta)' and 'Current estimate of theta' in `ex6_12_template.py`. In your answer, give both your analytical derivations and the completed code blocks.

Problem 3. *"KL-divergence."*

Recall the Normal-Gamma posterior example from lecture 3. Your task is to compute the KL-divergence between the true distribution of the samples and the distribution estimated using Bayesian learning. Repeat the computation for training set sizes in the range 5–5000 and as a final output, plot the KL-divergence as a function of the training set size. In your answer, also give the line(s) of code where you compute the KL-divergence.

You can use `ex6_3_normal_example.py` as a starting point that you can modify. You will need to write the computation of the KL-divergence between the true and learned distributions, write a loop to try different training set sizes and plot the results (you may remove the existing plots, as they are not needed for this exercise). See the documentation in `ex6_3_normal_example.py` for further hints.

Problem 4. *"Variational approximation for a simple distribution."*

Consider a model with two binary random variables x_1 and x_2, defined by the distributions:

$p(x_1)$	
x_1=0	0.4
x_1=1	0.6

$p(x_2 \mid x_1)$	x_1=0	x_1=1
x_2=0	0.5	0.9
x_2=1	0.5	0.1

Find a fully factorized distribution $q(x_1, x_2) = q_1(x_1)q_2(x_2)$ that best approximates the joint $p(x_1, x_2)$, in the sense of minimizing $\mathrm{KL}\,(p \parallel q)$.

Note: For "normal" variational inference, we would rather minimize $\mathrm{KL}\left(q \parallel p\right)$; recall that, in general, $\mathrm{KL}\left(p \parallel q\right) \neq \mathrm{KL}\left(q \parallel p\right)$ (see Barber: Bayesian Reasoning and Machine Learning, ch. Figure 28.1 as well as Chapter 28.3.4 and 28.3.5, for the dramatically different solutions that can result by minimizing the different quantities, as well as commentary on their relative usefulness for approximate inference). Here, we'll minimize $\mathrm{KL}\left(p \parallel q\right)$, as that is algebraically simpler.