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 9, due on Tuesday, 2nd April 2019, at 23:55

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

The document stochastic_gradient_elbo_search.pdf explains how Black-box variational inference can be implemented for our running example 'simple model'. Start by familiarizing yourself with the document. The purpose of Problems 1 and 2 is to derive and implement (some parts of) the calculation of the stochastic gradient of the ELBO.

Problem 1. *"Stochastic ELBO gradient for the simple model (1/2)."*

(a) Derive Equation 7 in stochastic_gradient_elbo_search.pdf.

(b) Implement simulation of the unobserved variables $(\tau, \theta, \mathbf{z})$ from the current approximation q into the function `sample_from_q` in the template `ex9_12_template.py`. In your answer, give the relevant completed code block.

(c) Implement the computation of the log joint of the approximate distribution (line 4 in Equation 7 in 'stochastic_gradient_elbo_search.pdf') into the function `compute_stochastic_elbo_gradient` in `ex9_12_template.py`. In your answer, give the relevant completed code block.

Problem 2. *"Stochastic ELBO gradient for the simple model (2/2)."*

(a) Compute all partial derivatives required to compute the stochastic gradient of the ELBO, and implement these into the function `compute_stochastic_elbo_gradient` in `ex9_12_template.py`. In your answer, give both the derived formulas and the relevant completed code block.

(b) Run the completed code (`ex9_12_template.py`). Verify that the algorithm converges approximately to the correct parameter values and also that the ELBO increases (approximately) until it eventually converges[1]. In your answer, give the final estimates of τ and θ, as well as the ELBO plot produced by the code.

Problem 3. *"SVI in Edward."*

An Edward implementation of linear regression using variational inference is presented in http://edwardlib.org/tutorials/supervised-regression. The same model using stochastic variational inference (with mini-batches) is presented in http://edwardlib.org/tutorials/batch-training.

(a) Compare side-by-side the Model and Inference definitions in these two examples. Highlight and explain the differences in the code (prepare your answer e.g. by saving the code for SVI as a PDF, highlighting relevant parts of code, and adding explanations as notes).

[1] Note: the algorithm uses previously defined closed-form updates for the 'responsibilities', as the stochastic gradient update for these seems unstable.

(b) Run the code for SVI and investigate the impact of the mini-batch size on the convergence speed.

Problem 4. *"VB for a factor analysis model."*

NB! This problem is optional and worth two bonus points.
Consider the factor analysis model

$$\begin{aligned}
\mathbf{x}_n &\sim \mathcal{N}_D(\mathbf{W}\mathbf{z}_n, \mathrm{diag}(\boldsymbol{\psi})^{-1}) \quad \forall n \in \{1, \ldots, N\} \\
\psi_d &\sim \mathrm{Gamma}(a, b) \quad \forall d \in \{1, \ldots, D\} \\
\mathbf{W}_k &\sim \mathcal{N}_D(\mathbf{0}, \alpha\mathbf{I}) \quad \forall k \in \{1, \ldots, K\} \\
\mathbf{z}_n &\sim \mathcal{N}_K(\mathbf{0}, \mathbf{I}) \quad \forall n \in \{1, \ldots, N\},
\end{aligned}$$

where $\mathbf{W}_k$ denotes the loadings of the kth factor and ψ_d^{-1} the specific variance of the dth observed variable. Furthermore D denotes the number of observed variables (i.e. $\mathbf{x}_n \in R^D$), N the number of data points, and K the number of factors in the model. $\mathrm{diag}(\boldsymbol{\psi})$ is a diagonal matrix with elements of $\boldsymbol{\psi} = (\psi_1, \ldots, \psi_D)^T$ on its diagonal.

Using the variational Bayes approach to approximate the posterior distribution with a factorized form

$$q(\Theta) = \prod_{d=1}^{D} q(\mathbf{W}_d) \prod_{n=1}^{N} q(\mathbf{z}_n) \prod_{d=1}^{D} q(\psi_d),$$

find the VB update for the factor $q(\mathbf{W}_d)$. Here $\mathbf{W}_d$ denotes the dth row of the loading matrix $\mathbf{W}$ as a column vector.