EECS E6720: Bayesian Models for Machine Learning

Columbia University, Fall 2018

Instructor: John Paisley

MIDTERM EXAM (150 points)

Exam details

- This is a take-home exam. It is open book, but you are not allowed to consult with anyone else on this exam.
- This exam is due by **11:59PM** on **Friday, November 9, 2018 on Courseworks**.
- This exam counts 150 points (equivalent to 30%) towards your final grade. Late submissions will have **5 points deducted for each minute late**.
- Submission time is non-negotiable. I will only grade your last submission to Courseworks. **Under no circumstances will I accept a late exam after 12:30AM.**
- You must submit your answers in a **single PDF file** that is **no more than 5MB** in size. Failure to do so will result in points being deducted.
- Show your work for full credit. Illegible work won't receive full credit. Photographs of your work that don't show up clearly will not receive full credit.
- Aside: Please note that the scenarios of these questions are somewhat vacuous. This is a test of concepts underlying model learning, not of specific models.

Question 1. Bayes rule and predictive distributions (25 + 25 points)

You have an observation (x, y) where $x \in \mathbb{R}$ and $y \in \{1, 2, 3\}$. You model $x \overset{ind}{\sim} \text{Normal}(\mu, \lambda^{-1})$ and $y \overset{ind}{\sim} \text{Discrete}(\pi)$. As priors, you define $\mu \overset{ind}{\sim} \text{Normal}(0, \gamma^{-1})$ and $\pi \overset{ind}{\sim} \text{Dirichlet}(\alpha)$.

a) Calculate the posterior distribution $p(\mu, \pi | x, y)$.

b) Calculate the predictive distribution of a new (x, y) pair. That is, calculate $p(x_2, y_2 | x_1, y_1)$ under the assumption (x_i, y_i) are i.i.d.

Question 2. Expectation-maximization algorithm (50 points)

You have an observation $x \in \mathbb{N}$. You model this as follows:

$$x|c \sim \text{Poisson}(\lambda_c), \qquad c \sim \text{Discrete}(\theta)$$

where θ is a K-dimensional probability vector you assume is known. Derive an EM algorithm for optimizing $\ln p(x, \lambda)$ over the vector λ, where c is the variable being integrated out.

Please note the following about what I am looking for in your answer:

- It must be clear that you understand what constitutes the "E" and the "M" steps. Partial credit will be given for correct algorithms without a clear path to the solution.
- You must give pseudo-code for optimizing λ, including the equations that you would implement in a coding language (similar to the algorithms outlined in the notes). If any expectations remain in your final algorithm, you should indicate what they are equal to.

Question 3. Variational inference (50 points)

You have a data set $\mathbf{x} = (x_1, \ldots, x_n)$, where $x \in \mathbb{N}$. You model this as $x_i|\lambda_i \sim \text{Poisson}(\lambda_i)$ with hierarchical prior $\lambda_i \overset{iid}{\sim} \text{Gamma}(a, \theta)$, $\theta \sim \text{Gamma}(b, c)$. For both gamma distributions, use the form $p(\gamma|\tau_1, \tau_2) = \frac{\tau_2^{\tau_1}}{\Gamma(\tau_1)}\gamma^{\tau_1 - 1}\mathrm{e}^{-\tau_2\gamma}$.

Using variational inference, approximate the full posterior $p(\lambda_1, \ldots, \lambda_n, \theta|\mathbf{x})$ with a factorized distribution $q(\lambda_1, \ldots, \lambda_n, \theta) = q(\theta)\prod_{i=1}^n q(\lambda_i)$. Derive the optimal $q(\lambda_i)$ and $q(\theta)$ and the corresponding variational inference algorithm for approximately minimizing $\text{KL}(q\|p(\vec{\lambda}, \theta|\mathbf{x}))$.

Again, please note the following about what I am looking for in your answer:

- You need to show:
  1. a derivation of the optimal distribution families of $q(\lambda_i)$ and $q(\theta)$,
  2. an equation-based algorithm (not a gradient-based algorithm) for learning their parameters,
  3. a final pseudo-code algorithm similar to those given in the notes that summarizes your derivation in a step-by-step recipe for implementation.
- Since it is possible to define these q, calculate $\mathcal{L}$ and optimize it using gradient methods, you can earn partial credit if you choose this "direct method." The "optimal method" will be much simpler. Therefore, you do not need to explicitly calculate the variational objective to receive full credit, but must clearly show how it factors into your final algorithm.
- For full credit, your work must show a clear path to your answer.