CS688: Graphical Models - Spring 2020

Assignment 5

Assigned: Wed, Apr 21st. Due: Mon, May 1st, 17:00pm.

General Instructions: Submit a report with the answers to each question at the start of class on the date the assignment is due. You are encouraged to *typeset your solutions*. To help you get started, the full LaTeXsource of the assignment is included with the assignment materials. For your assignment to be considered "on time", you must upload a zip file containing all of your code to Moodle by the due date. Make sure the code is sufficiently well documented that it's easy to tell what it's doing. You may use any programming language you like. For this assignment, you **may not** use existing code libraries for sampling or classification, but you **may** (and are encouraged to) use a library for automatic differentiation. If you think you've found a bug with the data or an error in any of the assignment materials, please post a question to the Piazza discussion forum. Make sure to list in your report any outside references you consulted (books, articles, web pages, etc.) and any students you collaborated with.

Academic Honesty Statement: Copying solutions from external sources (books, web pages, etc.) or other students is considered cheating. Sharing your solutions with other students is also considered cheating. Any detected cheating will result in a grade of -100% on the assignment for all students involved, and potentially a grade of F in the course.

Introduction: In this assignment, you will experiment with Bayesian inference and Stochastic Gradient Variational Inference. Specifically, you will implement Bayesian logistic regression with stochastic gradient variational inference.

Logistic Regression: Consider a logistic conditional distribution over $y \in \{-1, +1\}$ given $x \in \mathbb{R}^D$ and a vector of parameters $z \in \mathbb{R}^D$

$$p(y|x, z) = \begin{cases} \sigma\left(z^T x\right) & y = 1 \\ 1 - \sigma\left(z^T x\right) & y = -1 \end{cases}$$

for $\sigma(s) = 1/(1 + \exp(-s))$. Or, equivalently, $p(y|x, z) = \sigma(yz^T x)$.

Prior Distribution: For your prior on z please use a standard Normal distribution, with $p(z) \propto \exp\left(-\frac{1}{2}\|z\|^2\right)$

Data: You are given a dataset of 100 training inputs (`X_train.csv`) and outputs (`Y_train.csv`) and 1000 test inputs (`X_train.csv`) and outputs (`Y_train.csv`). Each vector x has 5 dimensions. (There is also data `X_forextracreditonly.csv` and `Y_forextracreditonly.csv` but as the names imply these should be completely ignored unless you are doing the extra credit problems.)

Question 1. (*10 points*) Derivation of likelihood

Mathematically derive an expression for $\log p(y|x, z)$. Write your answer using the function $\text{lae}(s, t) = \log(\exp(s) + \exp(t))$. Simplify your answer as much as possible.

Question 2. (*10 points*) **Derivation of gradient** First, mathematically derive the derivative of $\text{lae}(s,t)$ with respect to t. Next, derive the gradient of $\log p(y|x,z)$ with respect to z, $\nabla \log p(y|x,z)$.

Question 3. (*10 points*) **Pseudocode for Stochastic gradient variational inference (SGVI)** Use as a variational distribution a Gaussian distribution with a mean of $w \in R^D$ and a fixed covariance $\Sigma = 0.5I$. Write pseudocode to approximately compute $\int_z q_w(z) \log p(z, \text{Data})$, where the integral over z is approximated by drawing a single samples from z. Since the covariance is fixed, q_w has a fixed entropy, and so this part of the ELBO can be ignored. Also give pseudocode to compute the gradient of your approximation with respect to w.

Question 4. (*10 points*) **Implement SGVI** Implement SGVI using a fixed step size of $\epsilon = .005$ for a number of iterations $t_{\max} = 10000$. Make a plot of the weights evolving over time and show them as a plot with five lines, one for each dimension of w.

Question 5. (*20 points*) **Direct predictive accuracy for SGVI** Given a set of samples $z_1, ..., z_{\text{t}_{\max}}$ from the posterior $p(z|\text{Data})$, give a mathematical equation approximating the probability that $y = +1$ for a new test input x. Write a function that takes as input a set of samples z along with a test set of inputs x and computes the probability that each corresponding output is $+1$.
Run SGVI five times for each number of iterations $t_{\max} \in \{10, 100, 1000, 10000\}$. After inference is complete, take the final value of w, and draw 1000 samples z from the final variational approximation $q_w(z)$ and use that set of z to evaluate on test data[1]. Give a 4×5 table with all the error rates (four different time horizons and five different repetitions). Make sure to label the axes of your table.

Extra Credit: Finally, here are some extra-credit problems. These are *far* more difficult than the above problems and have very small point values. These are also deliberately more open-ended, leaving you more space for creativity. As a result, you will need to carefully describe exactly what you did for each problem. To maximize your score with limited time, you should make sure the above problems are done thoroughly and ignore these. We will be very stingy in giving credit for these problems-- do them only for the glory, and only at your own risk!

Question 6. (*5 points*) **Step Sizes and Run Lengths for SGVI Dynamics**
SGVI is based on stochastic gradient descent, and so finds the optimum only in the $\epsilon \to 0$ limit (even ignoring local optima). However, a smaller step-size requires a longer run-length. Systematically experiment with different step-sizes and different run lengths. Make a graph of the average performance over time of different step-sizes ϵ with time on the x-axes. Carefully describe what you have done, and explain what your results seem to suggest.

Question 7. (*5 points*) **Minibatches for SGVI**
The above problems all involved computing $\nabla \log p(z|\text{Data})$ by summing the likelihood over the full dataset. However, it is possible to instead get an unbiased estimator of the gradient $\nabla \log p(z|\text{Data})$ by sampling from a small subset of the data. The `{X,Y}_forextracreditonly.csv` files contain a larger training set of 1000 points. For a fixed time horizon of $t_{\max} = 10000$ iterations, experiment with SGVI with different minibatch sizes and different step sizes. Make a graph comparing the test-set

[1]To be clear: SGVI does an optimization over w. But once we are done, we take the final w and draw a set of 1000 vectors z. Then w can be discarded, and these z can be used to compute predictions.

performance with several other choices of minibatch sizes and step sizes. Make sure to compare to using all 1000 points in the minibatch.