Topics in Probabilistic Modeling and Inference (CS698X)
Homework 1 (Due date: Feb 6, 2018, 11:59pm)

Instructions

- We will only accept electronic submissions and the main writeup must be as a PDF file. If you are handwriting your solutions, please scan the hard-copy and convert it into PDF. Your name and roll number should be clearly written at the top. In case you are submitting multiple files, all files must to be zipped and **submitted as a single file** (named: your-roll-number.zip). Please do not email us your submissions. Your submissions have to be uploaded at the following link: `https://tinyurl.com/yb2537yq`).
- Each late submission will receive a 10% penalty per day for up to 3 days. No submissions will be accepted after the 3rd late day.

Problem 1 (30 marks)

Consider the probabilistic linear regression setting with likelihood model $p(y_n|\boldsymbol{x}_n, \boldsymbol{w}, \beta) = \mathcal{N}(y_n|\boldsymbol{w}^\top \boldsymbol{x}_n, \beta^{-1})$, with $\boldsymbol{x}_n \in \mathbb{R}^D$, $y_n \in \mathbb{R}$, and $\boldsymbol{w} \in \mathbb{R}^D$. Suppose we have N training examples $\{\boldsymbol{x}_n, y_n\}_{n=1}^N$. Denote the $N \times D$ feature matrix as $\mathbf{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N]^\top$ and $N \times 1$ response vector as $\boldsymbol{y} = [y_1, \ldots, y_N]^\top$

Consider two cases for the prior distribution on our main parameter, ithe regression weight vector: (1) $p(\boldsymbol{w}) = \prod_{d=1}^D \mathcal{N}(w_d|0, \lambda^{-1})$, and (2) $p(\boldsymbol{w}) = \prod_{d=1}^D \mathcal{N}(w_d|0, \lambda_d^{-1})$. Assume uniform priors for all the hyperparameters including the noise precision β.

- For each of the two cases, derive the expression for the marginal likelihood, i.e., for case 1, $p(\boldsymbol{y}|\mathbf{X}, \beta, \lambda)$ and for case 2, $p(\boldsymbol{y}|\mathbf{X}, \beta, \{\lambda_d\}_{d=1}^D)$. Are these distributions available in closed form?
- For each of the two cases, derive MLE-II based inference algorithm for learning $\boldsymbol{w}$ and the hyperparameters. Start by writing down the log of the marginal likelihood and do MLE on the resulting objective to get the optimal values for the hyperparameters.

Feel free to use properties of various distributions that we have seen in the class (i.e., you don't need to explicitly perform integrals, or use completing the squares trick, etc.). If you are making use of any other property in your derivations, please state that clearly. For results on derivatives, you may refer to the Matrix Cookbook[1].

Problem 2 (15 marks):

Assume a random variable x with distribution $p(x|\lambda) = \text{Poisson}(x|\lambda)$ where $\lambda > 0$ denotes the rate parameter of the Poisson. Assume the Poisson rate to have a distribution $p(\lambda|a, b) = \text{Gamma}(a, b)$ with non-negative shape parameter $a = r$ and rate parameter $b = (1-p)/p$ with $p \in (0, 1)$. Integrate out λ and give the complete expression for the marginal distribution of x, i.e., $p(x|a, b) = \int p(x|\lambda)p(\lambda|a, b)d\lambda$. (Optional, not for credit: Do you know the name of this distribution? If not, maybe try looking up :)).

[1] `https://tinyurl.com/yd4v5ee6`

Problem 3 (15 marks)

Suppose x is a scalar random variable drawn from a univariate Gaussian $p(x|\sigma^2) = \mathcal{N}(x|0, \sigma^2)$. The variance σ^2 itself is drawn from an exponential distribution: $p(\sigma^2|\gamma) = \text{Exp}(\sigma^2|\gamma^2/2)$, where $\gamma > 0$. Note that the exponential distribution is defined as $\text{Exp}(z|\lambda) = \lambda \exp(-\lambda z)$. Derive the expression of the marginal distribution of x, i.e., $p(x|\gamma) = \int p(x|\sigma^2)p(\sigma^2|\gamma)d\sigma^2$ after integrating out σ^2. Plot both $p(x|\sigma^2)$ and $p(x|\gamma)$ (you can assume $\sigma^2 = 1$ and $\gamma = 1$). What difference do you see between the shapes of these two distributions? **Note:** You don't need to submit the code used to generate the plots. Just the plots (appropriately labeled) are fine.

Hint: You will notice that $\int p(x|\sigma^2)p(\sigma^2|\gamma)d\sigma^2$ is a hard to compute integral. However, the solution does have a closed form expression. One way to get the result is to compute the **moment generating function (MGF)**[2] of $\int p(x|\sigma^2)p(\sigma^2|\gamma)d\sigma^2$ (note that this is a p.d.f.) and compare the obtained MGF expression with the MGFs of various p.d.f.s given in the table on the following Wikipedia page: `https://en.wikipedia.org/wiki/Moment-generating_function`, and identify which p.d.f.'s MGF it matches with. That will give you the form of distribution $p(x|\gamma)$. Specifically, name this distribution and identify its parameters.

Problem 4 (30 marks): Programming Assignment

In this problem, you will implement a Bayesian linear regression model and run it on a toy dataset (provided in the file `blrdata.txt`) which consists of 20 input-output pairs. In `blrdata.txt`, the first two numbers on each line denote the two features of each input and the third number is the scalar output. Note that the first feature is 1 for all the examples, so it is intrinsically essentially a one-dimensional regression problem.

Since there are two features per input, the weight vector to be learned will also be two dimensional. Your implementation should compute the posterior over $\boldsymbol{w}$, i.e, $p(\boldsymbol{w}|\mathbf{X}, \boldsymbol{y}, \lambda, \beta)$. Assume a prior $p(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{w}|0, \lambda^{-1}\mathbf{I})$ and likelihood model $p(y_n|\boldsymbol{w}, \boldsymbol{x}_n, \beta) = \mathcal{N}(y_n|\boldsymbol{w}^\top \boldsymbol{x}_n, \beta^{-1})$ with hyperparameters fixed at $\lambda = 2$ and $\beta = 25$.

Test your implementation by using only the first N examples in the data and vary N in the range $\{1, 2, 5, 10, 20\}$. For each value of N, use the code to compute the posterior $p(\boldsymbol{w}|\mathbf{X}, \boldsymbol{y}, \lambda, \beta)$ and do the following

1. Report the mean and covariance matrix of $p(\boldsymbol{w}|\mathbf{X}, \boldsymbol{y}, \lambda, \beta)$.
2. Produce a scatter plot showing 10000 randomly samples drawn from the inferred posterior $p(\boldsymbol{w}|\mathbf{X}, \boldsymbol{y}, \lambda, \beta)$. Depending on your programming language, you can use any Gaussian random number generator.
3. Produce a plot which shows 10 samples of the *predictive model*. To generate each sample, you will need to first generate a random $\boldsymbol{w}$ from the posterior $p(\boldsymbol{w}|\mathbf{X}, \boldsymbol{y}, \lambda, \beta)$ (as in part 2 above) and for that $\boldsymbol{w}$, plot the line $y = \boldsymbol{w}^\top [1, x]$ with x in the range $(-1, 1)$ on a two dimensional plot. In this plot, the X axis denotes the scalar x and Y axis denotes predicted response y. The plot should contain 10 such straight lines (each generated using a different $\boldsymbol{w}$).

Your implementation can be in MATLAB/Python/R. Submit your code and all the plots. The plots should be included in the main PDF writeup itself (don't submit them separately). **Briefly comment on what you observe in your results as N varies in the range $N = [1, 2, 5, 10, 20]$.**

Problem 5 (10 marks)

(Reading Assignment) For this problem, your task will be to study **Edward**[3], which is a software framework for quickly prototyping probabilistic/Bayesian models, and to write a 1-2 page (max. 1000 words) summary of the key features and capabilities of Edward. You are free to explore, refer to Edward's documentation, and discuss it with your classmates but your final writeup must be in your own words.

(Optional) I also highly encourage you to install and play with Edward. We might use it later during the semester (it may be a useful thing to know about anyway, if you are doing research on probabilistic/Bayesian modeling).

[2] MGF of a p.d.f. $p(x)$ is defined as $M_X(t) = \int_{-\infty}^{\infty} e^{tx} p(x) dx$

[3] `https://arxiv.org/pdf/1610.09787v2.pdf` and `https://arxiv.org/pdf/1701.03757v1.pdf`