INF367: Spring 2020
Exercise 9

Instructions:

You can either return the solutions electronically via MittUiB by Monday 10.00 or show them on paper on Monday's meeting. Grades are awarded for effort so scanned notes are fine if you solve exercises by hand (no need to make fancy latex files).

Students are encouraged to write computer programs to derive solutions whenever appropriate.

Tasks

Earlier, we have performed Bayesian linear regression assuming that the noise precision β is known. Let us now formulate a full Bayesian linear regression model by placing a prior on β.

Assume that we have observed n pairs $(\mathbf{x}_i, y_i)$ where $\mathbf{x}_i$ are the feature values and y_i is the label. Let $\mathbf{w} \in \mathbb{R}^d$ be our regression weights. The likelihood is

$$P(y \,|\, \mathbf{x}, w, \beta) = \prod_{i=1}^{n} N(y_i \,|\, \mathbf{w}^T \mathbf{x}_i, \beta^{-1}).$$

As earlier, we place a Gaussian prior on the regression weights:

$$P(\mathbf{w}) = N(\mathbf{w} \,|\, \mathbf{0}, \alpha^{-1}\mathbf{I})$$

where α (prior precision) is a user-defined hyperparameter. For β, we use a Gamma prior

$$P(\beta) = Gamma(\beta \,|\, a_0, b_0)$$

where a_0 and b_0 are user-defined hyperparameters.

The goal is to compute the posterior

$$P(\mathbf{w}, \beta \,|\, \mathbf{x}, y).$$

This distribution does not have a closed-form solution and therefore we use sampling to approximate the posterior.

1. Derive a Gibbs sampler for the above model. That is, derive the full conditional distributions $P(\mathbf{w} \,|\, \mathbf{x}, y, \beta)$ and $P(\beta \,|\, \mathbf{x}, y, \mathbf{w})$.

   Hint: When deriving the conditional distribution for $\mathbf{w}$ or β, you can drop all terms that do not depend on $\mathbf{w}$ or β, respectively. Note that both distributions involve conjugate priors.

2. Implement the Gibbs sampler. Download the data `new_york_bicycles2.csv`. The first column is the number of daily riders on Brooklyn Bridge (x) and the second column is the number of daily riders on Manhattan Bridge (y). Use your Gibbs sampler to learn a simple linear model (intercept and slope) for this data.

   Compute the logarithm of the joint probability of the parameter values (which is proportional to the posterior)

   $$\log P(y, \mathbf{x}, \mathbf{w}^{(t)}, \beta^{(t)}) = \log P(\beta^{(t)}) + \log P(\mathbf{w}^{(t)}) + \sum_{i=1}^{n} \log P(y_i \,|\, \mathbf{x}_i, \mathbf{w}^{(t)}, (\beta^{(t)})^{-1})$$

   after each iteration. Plot a *trace plot* of these values (x-axis is the iteration number). What can you say about the convergence based on this plot?

   Plot trace plots for the parameter values w_0, w_1 and β. What can you say about convergence and mixing based on these plots.

   Plot the marginal distributions of the parameters w_0, w_1 and β. Remember to exclude burn-in.

   Hint: Remember that $Gamma(a, b)$ is `gamma(a=a, scale=1/b)` in Scipy.

   Hint: You want sample thousands of samples. Typically, burn-in is about half of the samples.

3. Derive a Metropolis(-Hastings) algorithm. That is, come up with a proposal distribution $q(\mathbf{w}^*, \beta^* \,|\, \mathbf{w}, \beta)$. You should probably add a parameter or two controlling the "step-size", that is, how far from the current values can the proposed values be.

4. Implement the Metropolis-Hastings algorithm and do the same steps as in task 2. In addition, keep track of the proportion of proposal that were actually accepted.

   Hint: To avoid numerical problems in computing the acceptance ratio, one can use the following formula:

   $$\frac{P(\kappa^*)}{P(\kappa)} = e^{\log P(\kappa^*) - \log P(\kappa)}.$$

   Note that here you need the logarithm of the joint probability which you implemented alreaydy in task 2.

5. Which method gives better results?