Statistical Models and Computing Methods, Problem Set 4

November 27, 2022 Due 12/13/2022

Problem 1.
Suppose the current density is $q(x)$. Let $q_T(x)$ be the density after one MCMC iteration with transition kernel $T(x|y)$. That is,

$$q_T(x) = \int q(y)T(x|y)dy.$$

Denote the equilibrium distribution of the Markov chain as $p(x)$ and assume that the transition kernel satisfies the detailed balance condition

$$p(x)T(y|x) = p(y)T(x|y).$$

Show that

$$\mathrm{KL}(q_T\|p) \leq \mathrm{KL}(q\|p).$$

When does the equality hold?

Problem 2.
Consider the following multi-sample lower bounds for variational inference

$$\mathcal{L}_K(q) = \mathbb{E}_{\theta_1,\ldots,\theta_K\sim q(\theta)} \log\left(\frac{1}{K}\sum_{i=1}^{K}\frac{p(x,\theta_i)}{q(\theta_i)}\right)$$

Show that

$$\mathcal{L}_K(q) \leq \mathcal{L}_{K+1}(q) \leq \log p(x), \quad \forall K \geq 1.$$

You can use Pytorch or Tensorflow for the following problems. However, you are not allowed to use any libraries that provide some sort of "pre-cooked" implementation of the corresponding models. You need to implement them and the training algorithms from the basic building blocks yourself. For the optimization, we recommend you use one of the popular algorithms such as Adam.

Problem 3.
Consider the following banana-shaped distribution with normal priors

$$y_i \sim \mathcal{N}(\theta_1 + \theta_2^2, \sigma_y^2), \quad i = 1,\ldots,n, \quad \theta \sim \mathcal{N}(0, \sigma_\theta^2 I)$$

where $\sigma_\theta = 1, \sigma_y = 2$. Download the data from the course website.

(1) Derive the ELBO and gradient estimator (using the reparameterization trick) for a general normalizing flow model with a standard normal base distribution.

(2) Implement the following normalizing flows: planar flows, NICE and RealNVP. Use your favorite stochastic gradient ascent method for training. Show the lower bound as a function of the number of iterations.

(3) Implement a Hamiltonian Monte Carlo sampler to collect 500 samples (with the first 500 samples discarded as burn-in).

(4) Draw 500 samples from each of the trained normalizing flow models. Show the scatter plots of these samples and compare to your HMC results. You may also try out a larger sample size (e.g., 10000) and report the KL divergence to the ground truth from a long HMC run (say, 10000 sample with 10000 discarded as burn-in).

Problem 4.
In this problem, we ask you to rederive and implement Variational Autoencoder (VAE) on the MNIST dataset. More specifically, our goal is to learn a directed latent variable model that can represent a complex distribution over data in the following form

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}) = \int p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})d\boldsymbol{z}$$

In this problem, we assume a Gaussian prior on $\boldsymbol{z}$ and consider $\boldsymbol{x}$ to be binary vectors.

(1) Assume that we would like to approximate the posterior distribution $p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})$ using some variational distribution, $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$. Derive the evidence lower bound (ELBO) on the log likelihood of the model for N data points, $\{\boldsymbol{x}^{(i)}\}_{i=1}^N$.

(2) VAE optimizes the ELBO w.r.t. the parameters of the generative model and parameters of the variational distribution (inference model). Write down the stochastic estimate of the ELBO derived above. Derive the gradient of the ELBO using the reparameterization trick. Briefly describe the advantages and disadvantages of VAE.

(3) Use neural networks with one hidden layer that consists of 512 ReLU neurons for both generative and inference networks. Let the dimensionality of the latent space be 2. Set the output layer of the generative network for $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$ to be sigmoid neurons for binary representation of $\boldsymbol{x}$. Implement and train this VAE model for about 100 epochs. Provide the plots of the $\mathcal{L}_{1000}^{\text{train}}$ and $\mathcal{L}_{1000}^{\text{test}}$ vs. the epoch number.

(4) Visualize a random sample of 100 MNIST digits on 10×10 tile grid. Sample and visualize 100 "fake" digits from your trained model in the same manner.

(5) Use your trained inference model to transform images from the test set to the latent space. Visualize the points in the latent space as a scatter plot, where colors of points should correspond to the labels of the digits. Determine the min and max values of z_1, z_2 from the scatter plot and create a 20×20 mesh grid over the corresponding rectangle. Generate and visualize digits from your trained model for each of these grid points, and plot each set on a 20×20 tile grid.