

Weekly Report KARIMI-2021-09-17

1 Spars AMS code for Imagenet

Spent most of the past two weeks working on adapting our code to torch.distributed and above all running the code on the asimov cluster. Many bugs related to my account and the delay in replies from the IT team led to delays. Then beginning this week, a lot of tries to run on 4 nodes failed while talking to the IT team and Weijie at least once per day. I don't really know how we will fix this issue that is unrelated to our codebase (I also try a toy script that is supposed to print a message on several nodes). I am in contact with Wasim from Sunnyvale now to unblock this issue.

2 Memory Efficient EBM Training

The idea here is to reduce the memory cost of the MCMC during EBM training. That way, we can increase the number of samples in total used to obtain a more accurate approximation of the gradient.

Algorithm 1 EFF-EBM

- 1: **Input:** Number of iterations T , MCMC transitions K and of samples M , global learning rate $\{\eta_t\}_{t>0}$, MCMC stepsizes $\gamma_{kk>0}$, initial value θ_0 , MCMC initialization $\{z_0^m\}_{m=1}^M$ and observations $\{x_i\}_{i=1}^n$.
- 2: **for** $t = 1$ to T **do**
- 3: Draw M samples $\{z_t^m\}_{m=1}^M$ from the objective potential via Langevin diffusion:
- 4: **for** $k = 1$ to K **do**
- 5: Use black box compression operators to compress the gradient with respect to z :

$$\tilde{g}_{k-1}^m = \mathcal{C}(\nabla_z f_{\theta_t}(z_{k-1}^m))$$

where \mathcal{C} is either Sign, Topk, or maybe a simple count sketch operator (need to see in practice what makes sense).

- 6: Construct the Markov Chain as follows:

$$z_k^m = z_{k-1}^m + \frac{\gamma_k}{2} \tilde{g}_{k-1}^m + \sqrt{\gamma_k} \mathbf{B}_k, \quad (1)$$

where \mathbf{B}_t denotes the Brownian motion (Gaussian noise).

- 7: **end for**
- 8: Assign $\{z_t^m\}_{m=1}^M \leftarrow \{z_K^m\}_{m=1}^M$.
- 9: Sample m positive observations $\{x_i\}_{i=1}^m$ from the empirical data distribution.
- 10: Compute the gradient of the empirical log-EBM:

$$\nabla \log p(\theta_t) = \mathbb{E}_{p_{\text{data}}} [\nabla_{\theta} f_{\theta_t}(x)] - \mathbb{E}_{p_{\theta}} [\nabla_{\theta} f_{\theta_t}(z_t)] \approx \frac{1}{n} \sum_{i=1}^n \nabla_{\theta} f_{\theta_t}(x_i) - \frac{1}{m} \sum_{i=1}^m \nabla_{\theta} f_{\theta_t}(z_t^m).$$

- 11: Update the vector of global parameters of the EBM:

$$\theta_{t+1} = \theta_t + \eta_t \nabla \log p(\theta_t).$$

- 12: **end for**
 - 13: **Output:** Vector of fitted parameters θ_{T+1} .
-