Distributed and Private Stochastic EM Methods via Quantized and Compressed MCMC

Anonymous Author(s)
Affiliation
Address
`email`

Abstract

To be completed

1 Notations

We minimize the negated log incomplete data likelihood

$$\min_{\theta \in \Theta} \overline{L}(\theta) := L(\theta) + r(\theta) \quad \text{with } L(\theta) = \frac{1}{n} \sum_{i=1}^{n} L_i(\theta) := \frac{1}{n} \sum_{i=1}^{n} \big\{ -\log g(y_i;\theta) \big\} \ , \tag{1}$$

Consider a curved exponential family

$$f(z_i, y_i; \theta) = h(z_i, y_i) \exp\big(\langle S(z_i, y_i) \,|\, \phi(\theta) \rangle - \psi(\theta)\big) \ , \tag{2}$$

Then EM reads

$$\overline{s}_i(\theta) := \int_{\mathsf{Z}} S(z_i, y_i) p(z_i | y_i; \theta) \mu(\mathrm{d} z_i) \ , \tag{3}$$

and the *M-step* is given by

$$\overline{\theta}(\overline{s}(\theta)) := \underset{\vartheta \in \theta}{\arg\min} \ \big\{ \mathrm{R}(\vartheta) + \psi(\vartheta) - \langle \overline{s}(\theta) \,|\, \phi(\vartheta) \rangle \big\} \ . \tag{4}$$

In the case where the expectations are intractable, then (**??**) becomes:

$$\tilde{S}^{(k+1)} := \frac{1}{n} \sum_{i=1}^{n} \tilde{S}_i^{(k+1)} = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{M_k} \sum_{m=1}^{M_k} S(z_{i,m}^{(k)}, y_i) \ , \tag{5}$$

Submitted to 34th Conference on Neural Information Processing Systems (NeurIPS 2020). Do not distribute.

2 Algorithms

For computational purposes and privacy enhanced matter, I have chosen to study and develop the second algorithms that I proposed in my last week's report. In that algorithm, one does not compute a periodic averaging of the local models (this would requires performing as many M-steps as there are workers). Rather, workers compute local statistics and send them to the central server for a periodic averaging of those vectors and the latter computes one M-step to update the global model.

Algorithm 1 FL-SAEM with Periodic Statistics Averaging

1: **Input**: TO COMPLETE
2: Init: $\theta_0 \in \Theta \subseteq \mathbb{R}^d$, as the global model and $\bar{\theta}_0 = \frac{1}{n}\sum_{i=1}^{n} \theta_0$.
3: **for** $r = 1$ to R **do**
4:     **for** parallel for device $i \in D^r$ **do**
5:         Set $\hat{\theta}_i^{(0,r)} = \hat{\theta}^{(r)}$.
6:         Draw M samples $z_{i,m}^{(r)}$ under model $\hat{\theta}_i^{(r)}$
7:         Compute the surrogate sufficient statistics $\tilde{S}_i^{(r+1)}$
8:         Workers send local statistics $\tilde{S}_i^{(k+1)}$ to server.
9:     **end for**
10:     Server computes **global model using the aggregated statistics**:

$$\hat{\theta}^{(r+1)} = \overline{\theta}(\tilde{S}^{(r+1)})$$

    where $\tilde{S}^{(r+1)} = (\tilde{S}_i^{(r+1)}, i \in D_r)$ and send global model back to the devices.
11: **end for**

2.1 Challenges with Algorithm ??

While Algorithm **??** is a distributed variant of the SAEM, it is neither (a) *p*rivate nor (b) *c*ommunication-efficient.

Privacy: Indeed, we remark that broadcasting the vector of statistics are a potential breach to the data observations as their expression is related y and the latent data z. With a simple knowledge of the model used, the data could be retrieved if one extracts those statistics.

Communication bottlenecks: Also regarding (b), the broadcast of n vector of statistics $S(y_i, z_i)$ can be cumbersome when the size of the latent space and the parameter space of the model are huge.

2.2 Algorithmic solutions

Line ?? – Quantization: The first step is to quantize the gradient in the Stochastic Langevin Dynamics step used in our sampling scheme Line **??** of Algorithm **??**. Inspired by (**?**), we use an extension of the QSGD algorithm for our latent samples. Define the quantization operator as follows:

$$\mathsf{C}_j^{(\ell)}(g, \xi_j) = \|v\| \cdot \operatorname{sign}(g_j) \cdot \left(\lfloor \ell\, |g_j| / \|v\| \rfloor + \mathbf{1}\{\xi_j \leq \ell\, |g_j| / \|v\| - \lfloor \ell\, |g_j| / \|v\| \rfloor\}\right) / \ell \quad (6)$$

where ℓ is the level of quantization and $j \in [d]$ denotes the dimension of the gradient.

Hence, for the sampling step, Line **??**, we use the modified SGLD below, to be compliant with the privacy of our method.

Algorithm 2 Langevin Dynamics with Quantization for worker i

Input: Current local model $\hat{\theta}_i^{(r)}$ for worker $i \in [\![1, n]\!]$.

Draw M samples $\{z_i^{(r,m}\}_{m=1}^M$ from the posterior distribution $p(z_i|y_i;\hat{\theta}_i^{(k)})$ via Langevin diffusion with a quantized gradient:

for $k = 1$ to K **do**

  Compute the quantized gradient of $\nabla \log p(z_i|y_i;\hat{\theta}_i^{(k)})$:

$$g_i(k,m) = \mathsf{C}_j^{(\ell)}\left(\nabla_j f_{\theta_t}(z_i^{(k-1,m)}), \xi_j^{(k)}\right) \tag{7}$$

  where $\xi_j^{(k)}$ is a realization of a uniform random variable.

  Sample the latent data using the following chain:

$$z_i^{(k,m)} = z_i^{(k-1,m)} + \frac{\gamma_k}{2} g_i(k,m) + \sqrt{\gamma_k}\mathsf{B}_k \ , \tag{8}$$

  where B_t denotes the Brownian motion and $m \in [M]$ denotes the MC sample.

end for

Assign $\{z_i^{(r,m}\}_{m=1}^M \leftarrow \{z_i^{(K,m)}\}_{m=1}^M$.

Output: latent data $z_{i,m}^{(k)}$ under model $\hat{\theta}_i^{(t,k)}$

Line ?? – Compression MCMC output: We use the notorious **Top-**k operator that we define as $\mathcal{C}(x)_i = x_i$, if $i \in \mathcal{S}$; $\mathcal{C}(x)_i = 0$ otherwise and where $\mathcal{S}$ is defined as the size-k set of $i \in [p]$. Recall that after Line **??** we compute the local statistics $\tilde{S}_i^{(k+1)}$ using the output latent variables from Algorithm **??**. We now use those statistics and compress them using Algorithm **??** as follows:

Algorithm 3 Sparsified Statistics with **Top-**k

Input: Current local statistics $\tilde{S}_i^{(k+1)}$ for worker $i \in [\![1, n]\!]$. Sparsification level k.

Apply **Top-**k:

$$\ddot{S}_i^{(k+1)} = \mathcal{C}\left(\tilde{S}_i^{(k+1)}\right) \tag{9}$$

Output: Compressed local statistics for worker i denoted $\ddot{S}_i^{(k+1)}$.

Final method

Algorithm 4 FL-SAEM with Periodic Statistics Averaging

Input: Compression operator $\mathcal{C}(\cdot)$, number of rounds R, initial parameter θ_0.
for $r = 1$ to R **do**
  for parallel for device $i \in D^r$ **do**
    Set $\hat{\theta}_i^{(0,r)} = \hat{\theta}^{(r)}$. {Initialize each worker with current global model}
    Draw M samples $z_{i,m}^{(r)}$ under model $\hat{\theta}_i^{(r)}$ via Quantized LD: {Local MCMC step}
    for $k = 1$ to K **do**
      Compute the quantized gradient of $\nabla \log p(z_i|y_i; \hat{\theta}_i^{(k)})$:

$$g_i(k, m) = \mathsf{C}_j^{(\ell)} \left(\nabla_j f_{\theta_t}(z_i^{(k-1,m)}), \xi_j^{(k)} \right)$$

      where $\xi_j^{(k)}$ is a realization of a uniform random variable.
      Sample the latent data using the following chain:

$$z_i^{(k,m)} = z_i^{(k-1,m)} + \frac{\gamma_k}{2} g_i(k, m) + \sqrt{\gamma_k} \mathsf{B}_k,$$

      where B_t denotes the Brownian motion and $m \in [M]$ denotes the MC sample.
    end for
    Assign $\{z_i^{(r,m}\}_{m=1}^M \leftarrow \{z_i^{(K,m)}\}_{m=1}^M$.
    Compute $\tilde{S}_i^{(r+1)}$ and its **Top-k** variant $\ddot{S}_i^{(k+1)} = \mathcal{C}\left(\tilde{S}_i^{(k+1)}\right)$. {Compressed Local Statistics}
    Workers send local statistics $\tilde{S}_i^{(k+1)}$ to server. {Single round of communication}
  end for
  Server computes **global model**: {(Global) M-Step using Complete Statistics}

$$\hat{\theta}^{(r+1)} = \overline{\theta}(\ddot{S}^{(r+1)})$$

  where $\ddot{S}^{(r+1)} = (\ddot{S}_i^{(r+1)}, i \in D_r)$ and send global model back to the devices.
end for

3 Theoretical Findings

4 Numerical Experiments

References

D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic. Qsgd: Communication-efficient sgd via gradient quantization and encoding. In *Advances in Neural Information Processing Systems*, pages 1709–1720, 2017.