

# Weekly Report KARIMI 2021 07-09

For this week's report I will recall the main axis of research I am working on for the summer until next important deadlines like AAAI (21/08), ICLR (29/09) or AISTATS (08/10). Beginning next week's report I will only detail what I have done in the past 5 days and so on.

It follows two main domains:

## 1 MCMC based EBM training

### 1.1 Moreau-Yosida based proposal for MCMC within EBM training

I am working on a novel proposal for the MCMC within the EBM that is based on a regularization of the target distribution that would smooth any nonsmooth parts of the Gibbs potential: this is particularly appealing to Convnet-Based EBM or when any nonsmooth activations functions are employed. The algorithm is as follows:

---

**Algorithm 1** MY for Energy-Based model

---

- 1: **Input:** Total number of iterations  $T$ , number of MCMC transitions  $K$  and of samples  $M$ , sequence of global learning rate  $\{\eta_t\}_{t>0}$ , sequence of MCMC stepsizes  $\gamma_{kk>0}$ , initial value  $\theta_0$ , MCMC initialization  $\{z_0^m\}_{m=1}^M$  and observations  $\{x_i\}_{i=1}^n$ . Moreau Yosida decreasing sequence of parameters  $\{\lambda_t\}$
- 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:     Construct the Markov Chain as follows:

$$z_k^m = z_{k-1}^m + \gamma_k/2 [\nabla f_{\theta_t}(z_{k-1}^m) + \lambda_t^{-1} (z_{k-1}^m - \text{prox}_g^\lambda(z_{k-1}^m))] + \sqrt{\gamma_k} \mathbf{B}_k, \quad (1)$$

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

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

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

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

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

- 11: **end for**
  - 12: **Output:** Vector of fitted parameters  $\theta_{T+1}$ .
- 

This is in the same line as one of Jianwen's intern that works on smoothing the activation function RELU (with a smoothRELU function).

## 1.2 EBM for Motion Planning

We work on this with Weifu Wang. It is really hard to get it started since we are not sure of the best settings for integrating EBM in the planning procedure. Weifu is looking at how MDP could be used in his daily planning challenges. And I am looking at how EBM could be used to infer the transition probabilities better in the MDP. The mix of both could help obtain a better State transition probability density and thus improve the planning task.

## 2 (stochastic) EM algorithms under Federated Learning settings

### 2.1 Stochastic EM under Federated Learning settings

I am working on those two algorithms right now. Mainly doing numerical runs. Once this is done with preliminary results, Guanhua and I already talked about interesting theoretical properties that we could show.

The EM has never been studied under the Federated nor the distributed settings so I think that this line of work is interesting and important. I am still wondering which one of those following algorithms will work the best but I will detail more at my Monday's talk.

#### Periodic averaging of the local models

---

**Algorithm 2** FL-SAEM with parameter averaging

---

```

1: Input: .
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  $\hat{\theta}_i^{(0,k)} = \hat{\theta}^{(k)}$ .
6:     for  $t = 1$  to  $T$  do
7:       Draw  $M$  samples  $\{z_{i,m}^{(t,k)}\}_{m=1}^M$  under model  $\hat{\theta}_i^{(t,k)}$ 
8:       Compute the surrogate sufficient statistics  $\tilde{S}_i^{(t,k+1)}$ 
9:       Update local model:
           
$$\hat{\theta}_i^{(t,k+1)} = \bar{\theta}(\tilde{S}_i^{(t,k+1)})$$

10:    end for
11:    Devices send  $\hat{\theta}_i^{(T,k+1)}$  to server.
12:  end for
13:  Server computes the average of the local models:

```

$$\hat{\theta}^{(k+1)} = \frac{1}{n} \sum_{i=1}^n \hat{\theta}_i^{(T,k+1)}$$

and send global model back to the devices.

```

14: end for

```

---

#### Periodic averaging of the local statistics

---

**Algorithm 3** FL-SAEM with statistics averaging

---

```
1: Input: .  
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  $\hat{\theta}_i^{(0,k)} = \hat{\theta}^{(k)}$ .  
6:     for  $t = 1$  to  $T$  do  
7:       Here one local iteration,  $T = 1$   
8:       Draw  $M$  samples  $z_{i,m}^{(k)}$  under model  $\hat{\theta}_i^{(t,k)}$   
9:       Compute the surrogate sufficient statistics  $\tilde{S}_i^{(t,k+1)}$   
10:    end for  
11:    Devices send local statistics  $\tilde{S}_i^{(t,k+1)}$  to server.  
12:  end for  
13:  Server computes global model using the aggregated statistics:
```

$$\hat{\theta}^{(k+1)} = \bar{\theta}(\tilde{S}^{(t,k+1)})$$

where  $\tilde{S}^{(t,k+1)} = (\tilde{S}_i^{(t,k+1)}, i \in D_r)$  and send global model back to the devices.

```
14: end for
```

---