*2021 IEEE International Symposium on Information Theory*

# Two-Timescale Stochastic EM Algorithms

Belhal Karimi and Ping Li
May 14th 2021

Baidu Research, Cognitive Computing Lab

# How to Learn in Latent Data Models?

## Maximum Likelihood Approach

- We minimize the following **nonconvex** function on $\Theta$, a convex subset of $\mathbb{R}^d$

$$\min_{\theta \in \Theta} \overline{\mathcal{L}}(\theta) := \mathcal{L}(\theta) + r(\theta) \qquad \text{where} \qquad \mathcal{L}(\theta) = \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}_i(\theta) := \frac{1}{n}\sum_{i=1}^{n} \left\{ -\log g(y_i;\theta) \right\}$$

- $r : \Theta \mapsto \mathbb{R}$ is a smooth convex regularization function
- $g(y_i;\theta)$ is the marginal of the complete data likelihood $\quad g(y_i;\theta) = \int_{\mathcal{Z}} f(z_i, y_i;\theta)\mu(\mathrm{d}z_i)$

## Exponential Family Model

- $\{z_i\}_{i=1}^n$ are the (unobserved) latent variables.

- The complete data likelihood belongs to the curved exponential family:

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

- where $\psi(\theta)$, $h(z_i, y_i)$ are scalar functions, $\phi(\theta) \in \mathbb{R}^k$ is a vector function, and $\{S(z_i, y_i) \in \mathbb{R}^k\}_{i=1}^n$ are the vector of sufficient statistics.

# How to Learn in Latent Data Models?

## Expectation Maximization (EM) Algorithm and Monte Carlo (MC) variant

- batch EM (bEM) method **[DLR, 1977]** is the method of reference. 2 steps:
  - *E-step:* conditional expectation of the complete data sufficient statistics
  - *M-step:* maximization of the complete data likelihood

$$\bar{\mathbf{s}}(\theta) = \frac{1}{n}\sum_{i=1}^{n} \bar{\mathbf{s}}_i(\theta) \qquad \bar{\mathbf{s}}_i(\theta) = \int_{\mathcal{Z}} S(z_i, y_i) p(z_i | y_i; \theta) \mu(\mathrm{d}z_i)$$

$$\overline{\theta}(\bar{\mathbf{s}}(\theta)) := \operatorname{argmin}_{\vartheta \in \Theta} \left\{ r(\vartheta) + \psi(\vartheta) - \langle \bar{\mathbf{s}}(\theta), \phi(\vartheta) \rangle \right\}$$

- Monte Carlo EM (MCEM) method **[WT, 1990]** when the expectations are intractable

$$\text{MC-step}: \quad \tilde{S} = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{M} \sum_{m=1}^{M} S(z_{i,m}, y_i)$$

**Caveats**

- ***Requires large MC samples M in order to converge.***
- ***Do not scale to large n***

# Two-Time-Scale Stochastic EM

## Algorithms Formulation

- TTSEM formulates as the combination of the two levels

iSAEM $\quad \boldsymbol{\mathcal{S}}^{(k+1)} = \boldsymbol{\mathcal{S}}^{(k)} + n^{-1}\big(\tilde{S}_{i_k}^{(k)} - \tilde{S}_{i_k}^{(\tau_{i_k}^k)}\big)$

vrTTEM $\quad \boldsymbol{\mathcal{S}}^{(k+1)} = S_{\text{tts}}^{(\ell(k))} + \big(\tilde{S}_{i_k}^{(k)} - \tilde{S}_{i_k}^{(\ell(k))}\big)$

fiTTEM $\quad \boldsymbol{\mathcal{S}}^{(k+1)} = \overline{\boldsymbol{\mathcal{S}}}^{(k)} + \big(\tilde{S}_{i_k}^{(k)} - \tilde{S}_{i_k}^{(t_{i_k}^k)}\big)$

$$\overline{\boldsymbol{\mathcal{S}}}^{(k+1)} = \overline{\boldsymbol{\mathcal{S}}}^{(k)} + n^{-1}\big(\tilde{S}_{j_k}^{(k)} - \tilde{S}_{j_k}^{(t_{j_k}^k)}\big)$$

**Algorithm 2** Two-Timescale Stochastic EM methods.

**Input:** $\hat{\boldsymbol{\theta}}^{(0)} \leftarrow 0$, $\hat{\mathbf{s}}^{(0)} \leftarrow \tilde{S}^{(0)}$, $\{\gamma_k\}_{k>0}$, $\{\rho_k\}_{k>0}$ and $\mathsf{K}_\mathsf{f} \in \mathbb{N}^*$.

**for** $k = 0, 1, 2, \ldots, \mathsf{K}_\mathsf{f} - 1$ **do**

  Draw index $i_k \in [n]$ uniformly (and $j_k \in [n]$ for fiTTEM).

  Compute $\tilde{S}_{i_k}^{(k)}$ using the MC-step

  Compute the surrogate sufficient statistics $\boldsymbol{\mathcal{S}}^{(k+1)}$

  Compute $S_{\text{tts}}^{(k+1)}$ and $\hat{\mathbf{s}}^{(k+1)}$

$$S_{\text{tts}}^{(k+1)} = S_{\text{tts}}^{(k)} + \rho_{k+1}\big(\boldsymbol{\mathcal{S}}^{(k+1)} - S_{\text{tts}}^{(k)}\big)$$
$$\hat{\mathbf{s}}^{(k+1)} = \hat{\mathbf{s}}^{(k)} + \gamma_{k+1}\big(S_{\text{tts}}^{(k+1)} - \hat{\mathbf{s}}^{(k)}\big)$$

  Update $\hat{\boldsymbol{\theta}}^{(k+1)} = \overline{\boldsymbol{\theta}}(\hat{\mathbf{s}}^{(k+1)})$ via the M-step

**end for**

# Intuition Behind The Two Stages

## First Level: Variance Reduction

- **Incremental** updates to scale to large datasets → [Neal and Hinton, 1998], [Bottou and Bousquet, 2008].
- **Variance reduction** to control variance induced by incremental sampling → SVRG [Johnson et. al., 2013], FIEM [Karimi et. al., 2019].
- Temper the variance term $\mathbb{E}[\|\hat{s}^{(k)} - \mathcal{S}^{(k+1)}\|^2]$
- **Control variate**, as we are using it here, can be used for other algorithms. See control variate for MCMC [Brosse et. al., 2019].

## Second Level: Control the MC Fluctuations

- Robbins-Monro update. Decreasing stepsize to smooth the iterates instead of increasing the number of Monte Carlo samples
- Smaller Monte Carlo batchsize M.
- Averaging scheme (memory term in the drift term) → [Ruppert, 1988] and [Polyak, 1990].

# Numerical Applications

## Gaussian Mixture Models (GMM)

- Fit a GMM model to a set of n observations
- Each of M components with unit variance
- The complete log likelihood reads:

$$\theta := (\omega, \mu)$$

$$\omega = \{\omega_m\}_{m=1}^{M-1}$$

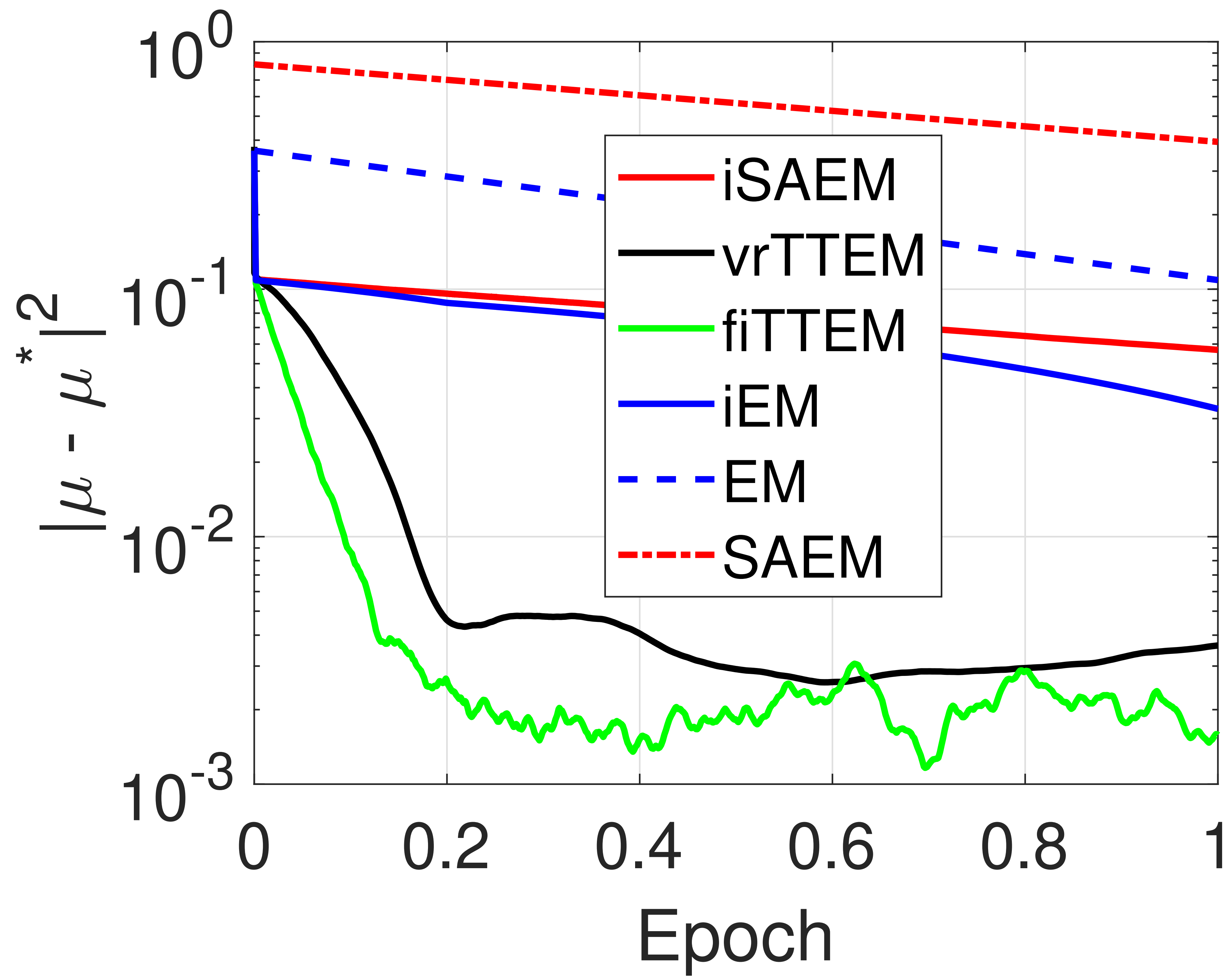
$$\mu = \{\mu_m\}_{m=1}^{M}$$

$$\log f(z_i, y_i; \boldsymbol{\theta}) = \sum_{m=1}^{M} 1_{\{m\}}(z_i)\left[\log(\omega_m) - \mu_m^2/2\right] + \sum_{m=1}^{M} 1_{\{m\}}(z_i)\,\mu_m y_i + \text{ constant}$$

- Penalization used: $\mathrm{R}(\boldsymbol{\theta}) = \frac{\delta}{2}\sum_{m=1}^{M}\mu_m^2 - \log \mathrm{Dir}(\boldsymbol{\omega}; M, \epsilon)$

## Experiments

- Numerical: GMM with M=2 and $\mu_1 = -\mu_2 = 0.5$
- **Fixed sample size:** size $n = 10^3$ and run to get $\mu^*$
  Stepsize for sEM $\gamma_k = 3/(k+10)$
  Stepsize for iSAEM $\gamma_k = 1/k^{0.6}$
- Compare to iEM, sEM and Batch EM



# Numerical Applications

## Deformable Template for Image Analysis

- $(y_i, i \in [1, n])$ images modeled as deformation of a template
- Deformable Template Model:

$$y_i(s) = I\left(x_s - \Phi_i\left(x_s\right)\right) + \sigma\varepsilon_i(s)$$

where s is the pixel index, $x_s$ its coordinate, $I$ the template and $\Phi_i(\cdot)$ the deformation.

$$I_\xi = \mathbf{K_p}\xi, \quad \text{where} \quad \left(\mathbf{K_p}\xi\right)(x) = \sum_{k=1}^{k_p} \mathbf{K_p}\left(x, p_k\right)\xi(k)$$

$$\Phi_i(x) = \left(\mathbf{K_g} z_i\right)(x) = \sum_{k=1}^{k_s} \mathbf{K_g}\left(x, g_k\right)\left(z_i^{(1)}(k), z_i^{(2)}(k)\right)$$

- **Goal:** Learn the vector of parameters $\theta = (\sigma, \xi, \Gamma)$ using TTSEM

## USPS Digits dataset

- USPS Digits dataset featuring 1000, (16X16)-pixel images for each class of digits from 0 to 9.

# Thank You!