# Distributed and Private Stochastic EM Methods

**Anonymous Author(s)**
Affiliation
Address
`email`

## Abstract

To be completed

## 1 Introduction

We consider the distributed minimization of the following negated log incomplete data likelihood

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

where $n$ denotes the number of workers, $\{y_i\}_{i=1}^n$ are observations, $\theta \subset \mathbb{R}^d$ is the parameters set and $\mathrm{R} : \theta \to \mathbb{R}$ is a smooth regularizer.

The objective $L(\theta)$ is possibly nonconvex and is assumed to be lower bounded. In the latent data model, the likelihood $g(y_i;\theta)$, is the marginal distribution of the complete data likelihood, noted $f(z_i, y_i;\theta)$, such that

$$g(y_i;\theta) = \int_{\mathsf{Z}} f(z_i, y_i;\theta)\mu(\mathrm{d}z_i), \tag{2}$$

where $\{z_i\}_{i=1}^n$ are the vectors of latent variables associated to the observations $\{y_i\}_{i=1}^n$.

We also consider a special case of that problem since the complete likelihood pertains to the curved exponential family:

$$f(z_i, y_i;\theta) = h(z_i, y_i)\text{exp}(\langle S(z_i, y_i) \,|\, \phi(\theta)\rangle - \psi(\theta)) , \tag{3}$$

where $\psi(\theta)$, $h(z_i, y_i)$ are scalar functions, $\phi(\theta) \in \mathbb{R}^k$ is a vector function, and $\{S(z_i, y_i) \in \mathbb{R}^k\}_{i=1}^n$ is the vector of sufficient statistics. We refer the readers to [Efron et al., 1975] for details on this subclass of problems which is of high interest given the broad range of problems that fall under this assumption. In the centralized settings, *i.e.,* when all data points are stored in a central server, a reference tool for learning such a model is called the EM algorithm [Dempster et al., 1977, Wu, 1983]. Comprised of two steps, the E-step computes an aggregated sum of expectations as follows:

$$\overline{s}(\theta) = \frac{1}{n}\sum_{i=1}^{n} \overline{s}_i(\theta) \quad \text{where} \quad \overline{s}_i(\theta) := \int_{\mathsf{Z}} S(z_i, y_i) p(z_i|y_i;\theta)\mathrm{d}z_i , \tag{4}$$

and the M-step is given by

$$\overline{\theta}(\overline{s}(\theta)) := \underset{\vartheta \in \theta}{\arg\min} \left\{ r(\vartheta) + \psi(\vartheta) - \langle \overline{s}(\theta) \,|\, \phi(\vartheta)\rangle \right\} . \tag{5}$$

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

### 1.1 Our motivations

**Expectations are not tractable:** Sampling for those approximations are costly.

**Need for distributed computing:** MovieLens, Large n, compute time, decentralized infrastructure

**Need for privacy and communication efficiency:** Sensible data (hospital, user data...) that can not be moved. Low bandwidth devices (compute should be light).

### 1.2 Our contributions

## 2 Related Work

**EM algorithms:**

In the case when the computation of the expectation under the posterior distribution is impossible, the Monte Carlo EM (MCEM) has been introduced in Wei and Tanner [1990] where a Monte Carlo (MC) approximation for this expectation is computed. A variant of that algorithm is the Stochastic Approximation of the EM (SAEM) in Delyon et al. [1999] leveraging the power of Robbins-Monro update Robbins and Monro [1951] to ensure pointwise convergence of the vector of estimated parameters using a decreasing stepsize rather than increasing the number of MC samples. The MCEM and the SAEM have been successfully applied in mixed effects models McCulloch [1997], Hughes [1999], Baey et al. [2016] or to do inference for joint modeling of time to event data coming from clinical trials in Chakraborty and Das [2010], unsupervised clustering in Ng and McLachlan [2003], variational inference of graphical models in Blei et al. [2017] among other applications. An incremental variant of the SAEM was proposed in Kuhn et al. [2019] showing positive empirical results but its analysis is limited to asymptotic consideration. Two-timescale methods of the SAEM have been proposed in [Karimi and Li, 2020] to accelerate the convergence. Gradient-based methods have been developed and analyzed in Zhu et al. [2017] but they remain out of the scope of this paper as they tackle the high-dimensionality issue.

**Distributed methods:**

[Morral et al., 2012] [Srivastava et al., 2019]

Traditional decentralized optimization methods include well-know algorithms such as ADMM [Boyd et al., 2011], Dual Averaging [Duchi et al., 2011], Distributed Subgradient Descent [Nedic and Ozdaglar, 2009]. More recent algorithms include Extra [Shi et al., 2015], Next [Di Lorenzo and Scutari, 2016], Prox-PDA [Hong et al., 2017], and Choco-SGD [Koloskova et al., 2019]. While these algorithms are commonly used in applications other than deep learning, recent algorithmic advances in the machine learning community have shown that decentralized optimization can also be useful for training deep models such as neural networks. Lian et al. [2017] demonstrate that a stochastic version of Decentralized Subgradient Descent can outperform parameter server-based algorithms when the communication cost is high. No existing work, to our knowledge, has seriously considered integrating *EM methods* in the setting of decentralized learning. One noteworthy work [Morral et al., 2012] proposes a decentralized version of the Online EM [Cappé and Moulines, 2009] and it is proven to satisfy some non-standard convergence.

**MCMC and Quantization:**

[Chopin and Ducrocq, 2021]

[Vono et al., 2021]

**Federated Learning methods:**

# 3 On the Decentralization of the EM algorithm

## 3.1 Distributed SAEM

We first consider the plain distributed version of the sEM which does not tackle any privacy or communication bottlenecks. We precise that we perform periodic locals models averaging. It goes as follows:

---

**Algorithm 1** d-SAEM: Distributed SAEM with Periodic Locals Models Averaging

---

1: **Input**: Compression operator $\mathcal{C}(\cdot)$, number of rounds $R$, initial parameter $\theta_0$.
2: **for** $r = 1$ to $R$ **do**
3:     **for** parallel for device $i \in D^r$ **do**
4:         Set $\hat{\theta}_i^{(r)} = \hat{\theta}^{(r)}$. {Initialize each worker with current global model}
5:         Draw M samples $z_{i,m}^{(r+1)}$ under model $\hat{\theta}_i^{(r)}$ via MCMC: {Local MCMC step}
6:         Compute the local statistics $\tilde{S}_i^{(r+1)} = S(z_{i,m}^{(r+1)})$. {Local statistics}
7:         Worker computes **local model**: {(Local) M-Step using local statistics}

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

8:         Worker sends local model $\hat{\theta}_i^{(r+1)}$ to server.
9:     **end for**
10:     Server computes **global model** by periodic averaging {Local model averaging}

$$\hat{\theta}^{(r+1)} := \frac{1}{n} \sum_{i=1}^{n} \hat{\theta}_i^{(r+1)}$$

11: **end for**

---

## 3.2 Federated SAEM with Quantization and Compression

While Algorithm 2 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.

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 2** FL-SAEM with Periodic Statistics Averaging

1: **Input**: number of rounds $R$, initial parameter $\theta_0$, number of MC samples $\{M_r\}_{r>0}$.
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 $M_r$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**

### 3.3 Embedded methods to comply with Federated settings

**Line 6 – Quantization:** The first step is to quantize the gradient in the Stochastic Langevin Dynamics step used in our sampling scheme Line 6 of Algorithm 2. Inspired by [Alistarh et al., 2017], we use an extension of the QSGD algorithm for our latent samples. Define the quantization operator as follows:

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

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

Hence, for the sampling step, Line 6, we use the modified SGLD below, to be compliant with the privacy of our method.

**Algorithm 3** 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 $\nabla \log p(z_i|y_i; \hat{\theta}_i^{(k)})$:

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

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

    Sample the latent data using the following chain:

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

    where $\mathsf{B}_t$ 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 7 – 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 6 we compute the local statistics $\tilde{S}_i^{(k+1)}$ using the output latent variables from Algorithm 3. We now use those statistics and compress them using Algorithm 4 as follows:

**Algorithm 4** 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 $\ddot{S}_i^{(k+1)}$.

We present our final method in Algorithm 5, that performs SAEM under the federated settings.

**Algorithm 5** fl-SAEM: Quantized and Compressed FL-SAEM with Periodic Statistics Averaging

**Input**: Compression operator $\mathcal{C}(\cdot)$, number of rounds $R$, initial parameter $\theta_0$.

**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 Quantized MCMC step}

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

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

$$g_i(k,m) = \mathsf{C}_j^{(\ell)}\left(\nabla_j f_{\theta_t}(z_i^{(k-1,m)}), \xi_j^{(k)}\right) \quad \text{where} \quad \xi_j^{(k)} \sim \mathcal{U}_{[a,b]}$$

      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 $\mathsf{B}_t$ 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 $\ddot{S}_i^{(r+1)} = \mathcal{C}\left(\tilde{S}_i^{(r+1)}\right)$. {Compressed local statistics}

    Worker send local statistics $\tilde{S}_i^{(r+1)}$ to server. {Single round of communication}

  **end for**

  Server computes **global model**: {(Global) M-Step using aggregated 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**

## 4 Theoretical Analysis

The following assumptions are required for the analysis.

**H1.** *The sets* $\mathsf{Z}, \mathsf{S}$ *are compact. There exist* $C_{\mathsf{S}}, C_{\mathsf{Z}}$ *such that:*

$$C_{\mathsf{S}} := \max_{s,s' \in \mathsf{S}} \|s - s'\| < \infty,$$

$$C_{\mathsf{Z}} := \max_{i \in [\![1,n]\!]} \int_{\mathsf{Z}} |S(z, y_i)| \mu(\mathrm{d}z) < \infty.$$

**H2.** *For any* $i \in [\![1,n]\!]$, $z \in \mathsf{Z}$, $\theta, \theta' \in \mathrm{int}(\theta)^2$ *(the interior of* $\theta$*), we have* $\left|p(z|y_i;\theta) - p(z|y_i;\theta')\right| \leq L_p \|\theta - \theta'\|$.

We also recall that we consider curved exponential family models such that the objective function satisfies:

**H3.** *For any* $\boldsymbol{s} \in \mathsf{S}$*, the function* $\theta \mapsto L(s,\theta) := \mathrm{R}(\theta) + \psi(\theta) - \langle s \,|\, \phi(\theta) \rangle$ *admits a unique global minimum* $\overline{\theta}(s) \in \mathrm{int}(\theta)$*. In addition,* $\mathrm{J}_{\phi}^{\theta}(\overline{\theta}(s))$*, the Jacobian of the function* $\phi$ *at* $\theta$*, is full rank,* $L_p$*-Lipschitz and* $\overline{\theta}(s)$ *is* $L_t$*-Lipschitz.*

The Monte Carlo noise of $\tilde{S}_i^{(k+1)}$ at iteration $k$ is defined as:

$$\eta_i^{(k)} := \tilde{S}_i^{(k)} - \overline{s}_i(\vartheta^{(k)}) \quad \text{for all} \quad i \in [\![1,n]\!] \quad \text{and} \quad k > 0 \tag{10}$$

and is controlled

**H4.** *For all* $k > 0$, $i \in [\![1,n]\!]$*, it holds:* $\mathbb{E}[\|\eta_i^{(k)}\|^2] < \infty \quad$ *and* $\quad \mathbb{E}[\|\mathbb{E}[\eta_i^{(k)}|\mathcal{F}_k]\|^2] < \infty$ .

Note that typically, the controls exhibited above are vanishing when the number of MC samples $M_k$ increases with $k$.

### 4.1 Finite-time convergence analysis of the d-SAEM

### 4.2 Finite-time convergence analysis of the fl-SAEM

# 5 Numerical Experiments

## 5.1 Nonlinear Mixed Models under Distributed Settings

Compare SAEM, MCEM, dist-SAEM and maybe one distributed Gradient Descent as baseline

Same for Private settings with Sketched SGD or another good baseline

**Fitting a linear mixed model on Oxford boys dataset [Pinheiro and Bates, 2006]:** We apply our various distributed methods on the oxboys dataset. The data consist in repeated measurements of height taken on 26 boys at 9 different timestamps.

In order to model the growth in height of our cohort of individuals, we consider a linear mixed effects model using base and slope variables. Following our notations above, we denote by $z_i = (\mathsf{base}_i, \mathsf{slope}_i)$ the vector of individual parameters. The model used in this example reads:

$$y_{ij} = f(t_{ij}, z_i) + \varepsilon_{ij} \quad \text{where} \quad f(t_{ij}, z_i) = \mathsf{base}_i + \mathsf{slope}_i * t_{ij} \ , \tag{11}$$

where $y_{ij}$ is the $j$-th height measurement at time $t_{ij}$ for patient $i$.

We assume in this example that the residual errors $\varepsilon_{ij}$ are independent and normally distributed with mean 0 and variance $\sigma^2$. A Lognormal distribution is used for the base parameter and a nomarl distribution is used for slope:

$$\log(\mathsf{base}_i) \sim \mathcal{N}(\log(\mathsf{base}_{\text{pop}}), \omega^2_{\mathsf{base}}) \ , \tag{12}$$

$$\mathsf{slope}_i \sim \mathcal{N}(\mathsf{base}_{\text{pop}}, \omega^2_{\mathsf{base}}) \ , \tag{13}$$

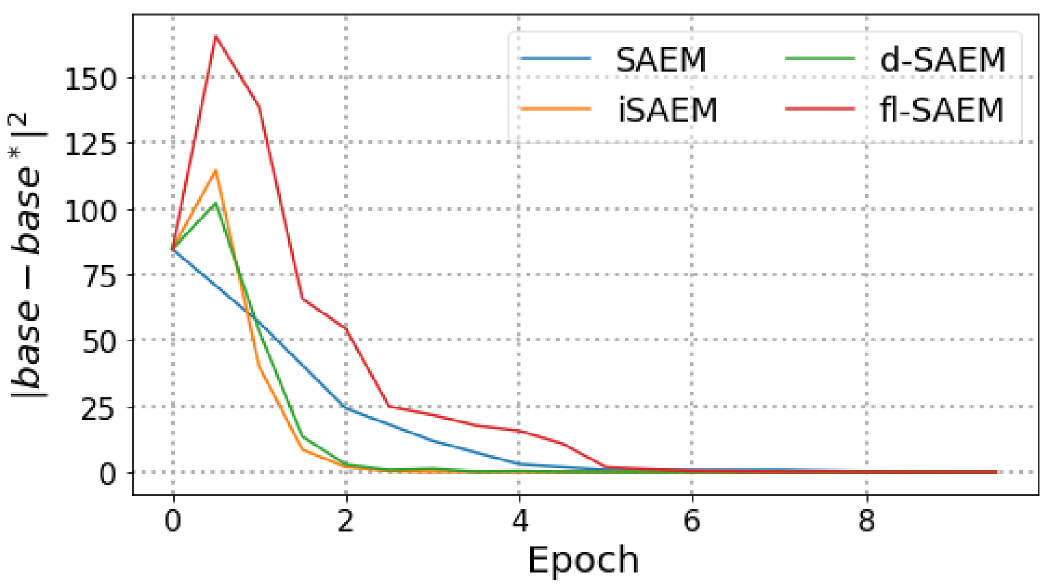


Figure 1: (Oxford Boys Dataset)

**Fitting a nonlinear mixed model on Warfarin dataset [Consortium, 2009]**

## 5.2 Probabilistic Latent Dirichlet Allocation

## 5.3 Bi-factor models under the Federated Learning settings

## 6 Conclusion

## 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.

C. Baey, S. Trevezas, and P.-H. Cournède. A non linear mixed effects model of plant growth and estimation via stochastic variants of the em algorithm. *Communications in Statistics-Theory and Methods*, 45(6):1643–1669, 2016.

D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational Inference: A Review for Statisticians. *Journal of the American statistical Association*, 112(518):859–877, JUN 2017. ISSN 0162-1459. doi: {10.1080/01621459.2017.1285773}.

S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

O. Cappé and E. Moulines. On-line expectation–maximization algorithm for latent data models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71(3):593–613, 2009.

A. Chakraborty and K. Das. Inferences for joint modelling of repeated ordinal scores and time to event data. *Computational and mathematical methods in medicine*, 11(3):281–295, 2010.

N. Chopin and G. Ducrocq. Fast compression of mcmc output. *Entropy*, 23(8):1017, 2021.

I. W. P. Consortium. Estimation of the warfarin dose with clinical and pharmacogenetic data. *New England Journal of Medicine*, 360(8):753–764, 2009.

B. Delyon, M. Lavielle, and E. Moulines. Convergence of a stochastic approximation version of the em algorithm. *Ann. Statist.*, 27(1):94–128, 03 1999. doi: 10.1214/aos/1018031103. URL https://doi.org/10.1214/aos/1018031103.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977.

P. Di Lorenzo and G. Scutari. Next: In-network nonconvex optimization. *IEEE Transactions on Signal and Information Processing over Networks*, 2(2):120–136, 2016.

J. C. Duchi, A. Agarwal, and M. J. Wainwright. Dual averaging for distributed optimization: Convergence analysis and network scaling. *IEEE Transactions on Automatic control*, 57(3):592–606, 2011.

B. Efron et al. Defining the curvature of a statistical problem (with applications to second order efficiency). *The Annals of Statistics*, 3(6):1189–1242, 1975.

M. Hong, D. Hajinezhad, and M.-M. Zhao. Prox-pda: The proximal primal-dual algorithm for fast distributed nonconvex optimization and learning over networks. In *International Conference on Machine Learning*, pages 1529–1538, 2017.

J. P. Hughes. Mixed effects models with censored data with application to hiv rna levels. *Biometrics*, 55(2):625–629, 1999.

B. Karimi and P. Li. Two-timescale stochastic em algorithms. In *IEEE International Symposium on Information Theory (ISIT)*, 2020.

A. Koloskova, S. U. Stich, and M. Jaggi. Decentralized stochastic optimization and gossip algorithms with compressed communication. In *International Conference on Machine Learning*, pages 3478–3487, 2019.

E. Kuhn, C. Matias, and T. Rebafka. Properties of the stochastic approximation em algorithm with mini-batch sampling. *arXiv preprint arXiv:1907.09164*, 2019.

X. Lian, C. Zhang, H. Zhang, C.-J. Hsieh, W. Zhang, and J. Liu. Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent. In *Advances in Neural Information Processing Systems*, pages 5330–5340, 2017.

C. E. McCulloch. Maximum likelihood algorithms for generalized linear mixed models. *Journal of the American statistical Association*, 92(437):162–170, 1997.

G. Morral, P. Bianchi, and J. Jakubowicz. On-line gossip-based distributed expectation maximization algorithm. In *2012 IEEE Statistical Signal Processing Workshop (SSP)*, pages 305–308. IEEE, 2012.

A. Nedic and A. Ozdaglar. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1):48, 2009.

S. Ng and G. McLachlan. On the choice of the number of blocks with the incremental EM algorithm for the fitting of normal mixtures. *Statistics and Computing*, 13(1):45–55, FEB 2003. ISSN 0960-3174. doi: {10.1023/A:1021987710829}.

J. Pinheiro and D. Bates. *Mixed-effects models in S and S-PLUS*. Springer Science & Business Media, 2006.

H. Robbins and S. Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

W. Shi, Q. Ling, G. Wu, and W. Yin. Extra: An exact first-order algorithm for decentralized consensus optimization. *SIAM Journal on Optimization*, 25(2):944–966, 2015.

S. Srivastava, G. DePalma, and C. Liu. An asynchronous distributed expectation maximization algorithm for massive data: the dem algorithm. *Journal of Computational and Graphical Statistics*, 28(2):233–243, 2019.

M. Vono, V. Plassier, A. Durmus, A. Dieuleveut, and E. Moulines. Qlsd: Quantised langevin stochastic dynamics for bayesian federated learning. *arXiv preprint arXiv:2106.00797*, 2021.

G. C. Wei and M. A. Tanner. A monte carlo implementation of the em algorithm and the poor man's data augmentation algorithms. *Journal of the American statistical Association*, 85(411):699–704, 1990.

C. J. Wu. On the convergence properties of the EM algorithm. *The Annals of statistics*, pages 95–103, 1983.

R. Zhu, L. Wang, C. Zhai, and Q. Gu. High-dimensional variance-reduced stochastic gradient expectation-maximization algorithm. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 4180–4188. JMLR. org, 2017.