THE PERTURBED PROX-PRECONDITIONED SPIDER ALGORITHM FOR EM-BASED LARGE SCALE LEARNING

G. Fort, E. Moulines

1 IMT, Université de Toulouse & CNRS, F-31062 Toulouse, France.
2 CMAP, Ecole Polytechnique, Route de Saclay, 91128 Palaiseau Cedex, France.

ABSTRACT

Incremental Expectation Maximization (EM) algorithms were introduced to design EM for the large scale learning framework by avoiding the full data set to be processed at each iteration. Nevertheless, these algorithms all assume that the conditional expectations of the sufficient statistics are explicit. In this paper, we propose a novel algorithm named Perturbed Prox-Preconditioned SPIDER (3P-SPIDER), which builds on the Stochastic Path Integral Differential Estimator EM (SPIDER-EM) algorithm. The 3P-SPIDER algorithm addresses many intractabilities of the E-step of EM; it also deals with non-smooth regularization and convex constraint set. Numerical experiments show that 3P-SPIDER outperforms other incremental EM methods and discuss the role of some design parameters.

Index Terms— Statistical Learning, Large Scale Learning, Expectation Maximization algorithm, Finite-sum Optimization, Accelerated Stochastic Approximation, Control Variates.

1. INTRODUCTION

EM [1] is a very popular computational tool, designed to solve non convex maximization problems on $\mathbb{R}^d$ when the objective function is not explicit but defined by an integral $F(\theta) = - \log \int G(z; \theta) d\mu(z)$. EM is a Majorize-Minimization algorithm which, based on the current value of the parameter $\theta_{\text{curr}}$, defines a majorizing function $\theta \rightarrow Q(\theta; \theta_{\text{curr}})$ through a Kullback-Leibler argument; then, the new point is chosen as the a minimum of $Q(\cdot; \theta_{\text{curr}})$. The computation of $Q$ is straightforward when there exist (known and explicit) functions $R, \phi, s$ such that $Q(\cdot; \theta_{\text{curr}}) = R(\cdot) - \langle \phi(\cdot), s(\cdot) \rangle$ and $s(\tau) \propto \int G(z; \tau) d\mu(z)$ is the expectation of the function $s$ with respect to (w.r.t.) the probability measure $G(\cdot; \tau) \exp(-F(\tau)) d\mu$. In these cases, the vector $s(\theta_{\text{curr}})$ defines the function $Q$.

It may happen that the vector $s(\theta_{\text{curr}})$ is not explicit (see e.g. [3] section 6); a natural idea is to substitute $s$ for an approximation, possibly random. A first level of intractability occurs when the integral $\langle \phi(\cdot), s(\cdot) \rangle$ is not explicit. Many stochastic EM versions were proposed and studied to overcome this intractability: among them, let us cite Monte Carlo EM [4,5] where $s$ is approximated by a Monte Carlo integration; and SA EM [6,7] where $s$ is approximated by a Stochastic Approximation (SA) scheme [8]. With the Big Data era, a second level of intractability occurred: EM applied to statistical learning evolved into online versions and large scale versions in order to minimize a loss function associated to a set of observations (also called examples). In large scale versions, the number of training data $n$ is too large to be processed at each iteration of EM: for example, when the majorizing function $Q$ of EM is of the form $Q(\theta; \theta_{\text{curr}}) = R(\theta) - \langle s(\theta_{\text{curr}}), \phi(\theta) \rangle$, the vector $s(\theta_{\text{curr}})$ often has the form $n^{-1} \sum_{i=1}^n s_i(\theta_{\text{curr}})$ and the sum over $n$ terms can not be allowed at each iteration of EM. To overcome this intractability in this so-called finite-sum setting, incremental EM-based algorithms were proposed: let us cite incremental EM [9]. Online-EM [10], sEM-VR [11], FIEM [12] (see also [13] for opt-FIEM) and SPIDER-EM [14,15]. The three algorithms sEM-VR, FIEM and SPIDER-EM can be seen as an Online-EM algorithm combined with a variance reduction technique through the construction of a control variate; they all improve on Online-EM (see e.g. [15]). However, these EM-based algorithms designed for the finite-sum framework all consider that the functions $\theta \rightarrow s_i(\theta)$ can be explicitly evaluated for any $\theta$ and $i = 1, \ldots, n$, while being defined as an expectation.

This paper introduces a novel EM-based procedure, named Perturbed Prox-Preconditioned SPIDER which tackles the two difficulties: (i) the finite-sum setting; (ii) the intractability of the quantities $s_i(\theta_{\text{curr}})$. It is proved in [14] that the complexity bounds of SPIDER-EM, expressed as the number of optimization steps and as the number of evaluations of the quantities $s_i(\theta_{\text{curr}})$ required to reach an $\varepsilon$-approximate stationary point of $F$, improves over the state-of-the art. Therefore, our algorithm builds on SPIDER-EM. It is also designed to address a composite problem with a non-smooth term. 3P-SPIDER is introduced in Section 3 with an emphasis on the case the quantities $s_i(\theta_{curr})$ are approximated by a Monte Carlo sum. In Section 4 the algorithm is applied to the logistic regression problem; insights on the choice of some design parameters are also given. It is shown that this perturbed version of SPIDER-EM improves on the perturbed version of Online-EM thus illustrating that the variance reduction technique is still perceptible. This benefit is all the more visible that the error when approximating the $s_i(\theta_{\text{curr}})$’s is small. Finally, since 3P-SPIDER combines two approximations to address the intractability of the $s_i(\theta_{\text{curr}})$’s and the finite-sum setting, it is advocated to regularly refresh the control-variate approximation with a full screening of the data set.

The complexity analysis of this algorithm is provided in [16]: under conditions on the approximations of the $s_i(\theta_{\text{curr}})$’s, which are satisfied for example for a Monte Carlo approximation, it is shown that 3P-SPIDER has the same complexity bounds as SPIDER-EM. In that sense, it remains optimal among the (perturbed) incremental EM algorithms.

Notations $\mathbb{R}_+$ and $\mathbb{N}^*$ denote respectively (resp.) the positive real line and the set of the positive integers. For $n \in \mathbb{N}^*$, set

---

This work is partially supported by the Fondation Simone et Cino Del Duca through the project OpSiMoRE and by the ANR-19-CHIA-0002-01. Part of this work was conducted under the auspices of the Lagrange Center in Mathematics and Computer Sciences.
2. THE PERTURBED PROX-PRECONDITIONED SPIDER ALGORITHM

2.1. The optimization problem

We address the minimization of an objective function \( F : \Theta \to \mathbb{R} \):

\[
\theta \mapsto -\frac{1}{n} \sum_{i=1}^{n} \log \int_{Z} h_i(z) \exp(\phi_i(z)) \, d\mu(z) + R(\theta)
\]  

(1)

where \( \Theta \) is an open subset of \( \mathbb{R}^d \), \( (Z, \mathcal{Z}) \) is a measurable space, \( Z \) denoting a \( \sigma \)-algebra over \( Z \); the functions \( \phi : \Theta \to \mathbb{R}^R \), \( R : \Theta \to \mathbb{R} \) and for all \( i \in [n]^* \), \( s_i : Z \to \mathbb{R}^d \) and \( h_i : Z \to \mathbb{R}^n_l \) are measurable; \( \mu \) is a dominating measure on \( [Z, \mathcal{Z}] \). The minimization of the negative log-likelihood in latent variable models provides examples of such a problem. As a first example, consider the maximum likelihood estimate of a mixture of densities from the curve density and its inverse.

2.2. EM in the expectation space

For solving this optimization problem, EM defines a sequence \( \{ \theta_k, k \geq 0 \} \) taking values in \( \Theta \), by repeating (i) E-step: compute

\[
Q(\theta; \theta_k) \overset{\text{def}}{=} -\frac{1}{n} \sum_{i=1}^{n} \int_{Z} (s_i(z), \phi(z)) \, p_i(z; \theta_k) \, d\mu(z) + R(\theta)
\]

for any \( z \in Z, \theta \in \Theta, i \in [n]^* \),

\[
p_i(z; \theta) \propto h_i(z) \exp(\phi_i(z))
\]

is a probability density; (ii) M-step: compute the minimum

\[
\theta_{k+1} \overset{\text{def}}{=} \arg\min_{\theta \in \Theta} Q(\theta; \theta_k), \quad Q(\theta; \theta_k) = R(\theta) - (s(\theta_k), \phi(\theta)),
\]

with

\[
\bar{s}(\theta) \overset{\text{def}}{=} \frac{1}{n} \sum_{i=1}^{n} s_i(\theta), \quad s_i(\theta) \overset{\text{def}}{=} \int_{Z} s_i(z) \, p_i(z; \theta) \, d\mu(z).
\]

Hereafter, we assume that for any \( \theta_{curr} \in \Theta, \theta \mapsto Q(\theta; \theta_{curr}) \) possesses an unique minimum and we define for any \( s \) in a closed convex set \( \mathcal{S} \supseteq \bar{s}(\theta) \),

\[
T(s) \overset{\text{def}}{=} \arg\min_{\phi \in \Theta} (R(\theta) - (s, \phi(\theta))).
\]

With these notations, it holds: \( \theta_{k+1} = T \circ \bar{s}(\theta_k) \).

In the logistic regression example, \( p_i(z; \theta) \, d\mu(z) \) is the posterior distribution of the hidden variable \( z_i \) given the observation \( Y_i \); \( q = d; \theta \mapsto R(\theta) - (s, \phi(\theta)) \) possesses an unique minimum; for any \( s \in S = \mathbb{R}^d \), \( T(s) = \bar{s}(\Theta) \) where

\[
\Omega \overset{\text{def}}{=} \left( \frac{1}{\sigma^2 n} \sum_{i=1}^{n} X_iX_i^T + 2\tau I_d \right)^{-1}.
\]

2.3. The Perturbed Prox-Preconditioned SPIDER algorithm

Given a sequence of positive step sizes \( \{ \gamma_k, k \geq 0 \} \), \( S \) defines a sequence \( \{ \tilde{S}_k, k \geq 0 \} \) such that

\[
\tilde{S}_{k+1} = \tilde{S}_k + \gamma_k I_h \|
\]

where \( H_{k+1} \) is an approximation of \( h(\tilde{S}_k) \). Observing that \( h(\theta) = \mathbb{E}[s_i(\theta)] \) for some \( [n]^* \)-valued uniform random variable \( I \), a natural idea to mimic the asymptotic behavior of EM is the definition

\[
H_{k+1} \overset{\text{def}}{=} \frac{1}{\beta} \sum_{i \in B_{k+1}} \bar{s}_i(\tilde{S}_k) - \tilde{S}_k
\]

where \( B_{k+1} \) is a batch of size \( b \) sampled uniformly from \( [n]^* \) (with or without replacement) and independently of \( \tilde{S}_k \). Such a strategy corresponds to the Online-EM algorithm. The incremental EM-based algorithms with variance reduction techniques use the property \( h(\tilde{S}_k) = \mathbb{E}[H_{k+1} + V|\tilde{S}_k] \) for any (conditionally) centered
random variable $V$. This implies that, thanks to an adequate construction of the control variate $V$, the variance of the approximation of $h(S_t)$ can be reduced (see e.g. [17] for an introduction to variance reduction methods in Monte Carlo sampling). This is the essence of sEM-VR, FIEM and SPIDER-EM which essentially differ in the definition of $V$.

3P-SPIDER is described in Algorithm 1. As in SPIDER-EM, the control variate is refreshed regularly, let us say at the beginning of each outer loop (see lines 2 and 10). In SPIDER-EM, it is defined as $\bar{s} \circ T(S_{t-1}) = n^{-1} \sum_{s \in (S_{t-1})} s$. Here, two perturbations are allowed: the approximation of $\bar{s}(S_{t-1})$ with a quantity denoted by $\hat{s}_{t,k}^{-1}$, and an error $E_t$ which may include for example the situation when a sub-sample of the $n$ samples is used when computing the sum instead of the full data set. At each inner loop (see lines 3 and 10), the control variate is modified in order to track the ideal sequence for an example). The proof of the convergence in expectation is proved. In the case $g$ is the indicator function of a closed convex set $K$ and $B(s)$ is invertible for any $s \in K \cap S$, the limiting points are the roots of $\nabla W$ which are in $K$, that is the roots of $h(s)$ in $K$: 3P-SPIDER has the same asymptotic behavior as EM.

2.4. Case of a Monte Carlo approximation

The intractable quantity $\bar{s}_t \circ T(s)$ is defined by

$$\bar{s}_t \circ T(s) \overset{def}{=} \int_{Z} s_t(z) p_t(z; T(s)) d\mu(z) ,$$

where $p_t(z; T(s))$ is the distribution defined by $[4]$. When this integral is not explicit, a natural idea is to approximate it by a Monte Carlo (MC) sum. For example,

$$\hat{s}_{t,k} = \frac{1}{m_{t,k}+1} \sum_{i=1}^{m_{t,k}+1} s_i \left( Z_{r,i}^{-1} \right) ,$$

for any $\gamma > 0$ and any $q \times q$ positive-definite matrix $B$.

Algorithm 1: The Perturbed Prox-Preconditioned SPIDER (3P-SPIDER) algorithm.
3. APPLICATION: INFERENCE IN THE LOGISTIC REGRESSION MODEL

Let us consider the logistic regression model described in Section 2.1. Since $P(Y_i = y_i) \leq 1$, it can be proved that the minima of $F$ are in the set $\{ \theta \in \mathbb{R}^d : \tau \| \theta \|^2 \leq n^{-1} \}$. Therefore, in the expectation space, the minima of $W = F \circ \Theta$ are in the set $\{ s \in \mathbb{R}^d : \tau s' \Omega s \leq n/4 \}$ which is included in $K = \{ s \in \mathbb{R}^d : \tau s' \Omega s \leq n/4 \lambda_\min \}$ where $\lambda_\min$ is the positive minimal eigenvalue of $\Omega$ (see (3)). 3P-SPIDER is applied with $q \overset{\text{def}}{=} X_\ell$, the characteristic function of the compact convex set $K$. With this definition of $K$ and since $B(s) = \Omega$ for any $s \in \mathcal{S} \overset{\text{def}}{=} \mathbb{R}^d$, the computation of the operator $\text{Prox}_{B(s), \gamma}$ is explicit. $s/(\Omega s)$ is equal to

$$\frac{X_\ell}{\sigma^2 \| X_\ell \|} Z(s) = \int_R \frac{\exp(z \langle X_\ell, \Omega s \rangle / (\sigma^2 \| X_\ell \|))}{1 + \exp(-X_\ell' z / \| X_\ell \|)} \exp(-z^2/(2\sigma^2)) dz$$

where the normalizing constant $Z(s)$ is given by

$$Z(s) \overset{\text{def}}{=} \int_R \frac{\exp(z \langle X_\ell, \Omega s \rangle / (\sigma^2 \| X_\ell \|))}{1 + \exp(-X_\ell' z / \| X_\ell \|)} \exp(-z^2/(2\sigma^2)) dz.$$ 

These integrals are not explicit; we consider the approximation $s_{t,k}^{1,b}$ of $s/(\Omega \hat{s}_{t,k})$ given by a MC sum as described in Section 2.4: the samples $\hat{s}_{t,k}$ are obtained by the Gibbs sampler given in (23).

The numerical illustrations use the MNIST data set: the class “1” contains the 12 873 images in the training set labeled 1 and 3 and the class “-1” contains the 12 116 images in the training set labeled 7 and 8; hence $n = 24 989$. The 787 pixels are compressed in 50 features through PCA (see section 5) for the details. An intercept is included in the covariates so $d = 51$. 3P-SPIDER is run with $\sigma^2 = 0.1$, $\tau = 1$, $k_{\text{out}} = 20$, $k_{\text{in}} = \lceil \sqrt{n}/10 \rceil = 16$ and $b = \lceil 10\sqrt{n}/\sqrt{2} \rceil = 1581$. Note that $k_{\text{in}} \times b = n$ so that each outer loop requires $n$ examples; it corresponds to an epoch.

We study the quantity

$$D_{t,k} = \mathbb{E} \left[ \left\| \frac{\hat{S}_{t,k+1} - \hat{S}_{t,k}}{\gamma_{t,k+1}} \right\|^2 \right], \quad t \in [k_{\text{out}}]^*, k \in [k_{\text{in}} - 1],$$

which quantifies how far 3P-SPIDER is from its limiting set (see the definition of $L$); this expectation is estimated by a MC sum over 25 independent runs. All the runs start from the same value $\hat{S}_{\text{init}}$. For the computation of the quantity $S_{t,0}$ at each outer loop $#t$, all the examples are used and the expectations $\hat{s}_t \circ \Theta(\hat{S}_{t,-1})$ are approximated by a MC sum with $m' = 10\sqrt{n} = 1590$ points. Hence, $\mathcal{E}_t = 0$ except when specified (see the second analysis below); the computation of $S_{t,0}$ requires $n$ examples: it corresponds to an epoch.

First, 3P-SPIDER is run with $m_{t,k} = 2\sqrt{n}$ and $\gamma_{t,k} = \gamma_{t,0} = 0.1$. Online-EM is run with a step size equal to $\gamma = 0.1$, a batch size $b = \lceil 10\sqrt{n} \rceil$ (case “sqr”) and $b = n$ (case “full”), and a MC approximation for $\hat{s}_t$ computed with $2\sqrt{n}$ points. Figure 1(a) displays $D_{t,0}$ for 3P-SPIDER and $\| \hat{S}_{t+1} - \hat{S}_t \|^2/\gamma_t^2$ for Online-EM. The x-axis scales as the number of epochs, that is the use of $n$ examples. The plot shows that, even when the expectations $\hat{s}_t$ have to be replaced with approximations, 3P-SPIDER far more efficient than Online-EM (in which the exact expectations are also replaced with MC approximations).

Second, we analyze the role of $\mathcal{E}_t$ when initializing the control variate $S_{t,0}$. We run 3P-SPIDER with $\gamma_{t,k} = \gamma_{t,0} = 0.1$ and a number of MC points $m_{t,k} = 2\sqrt{n}$; the quantity $D_{t,k}$ is displayed on Figure 1(b) vs the cumulated number of inner loops; the squares, circles and diamonds indicate $D_{t,k}$, $\hat{S}_{t,k}$, and $\gamma_{t,k}$, for every outer loop. The case “full” corresponds to $\mathcal{E}_t = 0$, the case “half” (resp. “quarter”) corresponds to $S_{t,0}$ computed with a batch of size $\lfloor n/2 \rfloor$ examples (resp. $\lfloor n/4 \rfloor$). The control variate is too poor in the case “half” and “quarter” and, after the transient phase when the possibly bad initialization is forgotten, it weakens the benefit of its use: we definitely advice $\mathcal{E}_t = 0$.

Third, we analyze how the variability of the MC approximation and the choice of the step sizes affect the rate of convergence of 3P-SPIDER. In “Case 1”, the values are the same as in Figure 1(a). In “Case 2”, $\gamma_{t,k} = \gamma_{t,0} = 0.1$ during the first three outer loops and then $\gamma_{t,k} = 10^{-3}$; $m_{t,k}$ is as in “Case 1” until the outer loop $\#10$ and then $m_{t,k}$ is multiplied by 5. In “Case 3”, the step sizes and the number of MC points are as in “Case 2”, except that the step size decreases later, at outer loop $\#6$. On Figure 1(c), we display $D_{t,k}$ vs the cumulated number of inner loops, starting from the number $\#32$ (that is, at the end of the second outer loop); the diamonds, circles and squares indicate $D_{t,k}$. First, 3P-SPIDER is improved when the number of MC points increases; when the fluctuations of the algorithm is of the same order as the fluctuations of the MC errors, 3P-SPIDER can not go forward anymore in order to reach a more precise estimation of the parameter (compare “Case 1” and “Case 2”). Small step sizes penalize the algorithm (compare “Case 1” and “Case 2”).

Finally, we discuss the strategies $\gamma_{t,0} = 0$ and $\gamma_{t,0} \neq 0$. 3P-SPIDER run as in Figure 1(a) corresponds to “Case 1”. In “Case 2” and “Case 3”, the number of MC points is multiplied by 5 from the outer loop $\#11$, and $\gamma_{t,k}$ is 0.1 for any $k > 0$. In “Case 1” and “Case 2”, $\gamma_{t,0} = 0.1$ and in “Case 3”, $\gamma_{t,0} = 0$. On Figure 1(d), we display $D_{t,k}$ vs the cumulated number of inner loops, starting from the loop $\#32$; the diamonds, circles and squares indicate $D_{t,k}$. Here again, we observe the benefit of reducing the MC variability by increasing the number of MC points (compare “Case 1” to the other cases); “Case 2” and “Case 3” are almost similar, maybe with a slightly better behavior for “Case 2”.

![Fig. 1. (a) top left] Comparison of algorithms; [(b) top right] Role of the size of the batch when computing $S_{t,0}$; [(c) bottom left] Role of the step sizes $\gamma_{t,k}$ and the number of Monte Carlo points when computing $\hat{s}_{t,k}$; [(d) bottom right] Role of $\gamma_{t,0}$.
4. REFERENCES

[1] A.P. Dempster, N.M. Laird, and D.B. Rubin, “Maximum Likelihood from Incomplete Data via the EM Algorithm,” J. Roy. Stat. Soc. B Meth., vol. 39, no. 1, pp. 1–38, 1977.

[2] C.F.J. Wu, “On the Convergence Properties of the EM Algorithm,” Ann. Statist., vol. 11, no. 1, pp. 95–103, 1983.

[3] G.J. McLachlan and T. Krishnan, The EM algorithm and extensions, Wiley series in probability and statistics. Wiley, 2008.

[4] G.C.G. Wei and M.A. Tanner, “A Monte Carlo Implementation of the EM Algorithm and the Poor Man’s Data Augmentation Algorithms,” J. Am. Stat. Assoc., vol. 85, no. 411, pp. 699–704, 1990.

[5] G. Fort and E. Moulines, “Convergence of the Monte Carlo expectation maximization for curved exponential families,” Ann. Statist., vol. 31, no. 4, pp. 1220–1259, 2003.

[6] B. Delyon, M. Lavielle, and E. Moulines, “Convergence of a Stochastic Approximation version of the EM algorithm,” Ann. Statist., vol. 27, no. 1, pp. 94–128, 1999.

[7] O. Cappé and C. Robert, “Markov Chain Monte Carlo: 10 years and still running!,” Journal of the American Statistical Association, vol. 95, no. 452, pp. 1282–1286, 2000.

[8] S. Brooks, A. Gelman, G. Jones, and X.-L. Meng, Handbook of Markov Chain Monte Carlo, CRC press, 2011.

[9] N. G. Polson, J.G. Scott, and J. Windle, “Bayesian Inference for Logistic Models Using Pólya–Gamma Latent Variables,” Journal of the American Statistical Association, vol. 108, no. 504, pp. 1339–1349, 2013.

[10] G. Fort, E. Moulines, and H.-T. Wai, “Faster Variance Reduced Stochastic Expectation Maximization for Nonconvex Finite-Sum Optimization,” in proceedings of the 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2021, Supplementary material available on HAL.