SGD with Variance Reduction beyond Empirical Risk Minimization

MCM 2017

Massil Achab

S. Gaiffas, E. Bacry, A. Guilloux

4th July 2017

CMAP, Ecole Polytechnique

Outline

Introduction

Introduction

- Most machine learning problems can be expressed as a convex optimization problem

$$\hat{\theta} = \arg\min_{\theta \in \mathbb{R}^d} \left(\frac{1}{n} \sum_{i=1}^{n} f_i(\theta) + g(\theta) \right),$$

- Usually, $f = \frac{1}{n} \sum_{i=1}^{n} f_i$ is a convex data fitting term (usually smooth), and g is a convex penalty on the predictor (smooth or not).

- Example (Lasso): $f_i(\theta) = (y_i - \theta^\top x_i)^2$ and $g(\theta) = ||\theta||_1$.

Introduction

Usual supervised machine learning framework

- **Data:** n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}, i = 1, \dots, n$

- **Prediction function:** $h(x, \theta) \in \mathbb{R}$ parametrized by $\theta \in \mathbb{R}^d$

- **Empirical Risk Minimization:** find $\hat{\theta}$ solution of

$$\min_{\theta \in \mathbb{R}^d} \quad \frac{1}{n} \sum_{i=1}^{n} f_i(\theta) + g(\theta), \qquad \text{with} \qquad f_i(\theta) = \ell(y_i, h(x_i, \theta))$$

- **Examples:** linear regression, logistic regression, support vector machines, neural networks, ...

Cox partial likelihood

- Goal: relate covariates of a patient to its survival time
- Cox regression model: *regression* that can extract information from patients whose failure time is not observed
- Semi-parametric model on the hazard function of a patient

$$\lim_{h \to 0} \frac{\mathbb{P}(t \leq T \leq t + h | t \leq T)}{h} = \lambda_0(t) \exp(\theta^\top x)$$

- Estimation of θ through maximization of the partial log-likelihood

$$\ell(\theta) = -\frac{1}{|D|} \sum_{i \in D} \left[-\theta^\top x_i + \log \left(\sum_{j \in R_i} \exp(\theta^\top x_j) \right) \right]$$

Vanilla algorithm to find $\hat{\theta}$

Proximal operator

The proximal operator of h is defined by

$$\text{prox}_h(y) = \arg\min_{x \in \mathbb{R}^d} \{h(x) + 1/2||y - x||_2^2\},$$

where $||\cdot||_2$ is the usual Euclidean norm.

Proximal Gradient Descent

- Given a starting point θ_0 and η small enough
- Until convergence, do

$$\theta^{t+1} \leftarrow \text{prox}_{\eta g}\left[\theta^t - \eta \nabla f(\theta^t)\right]$$

SGD and Variance Reduction

Context

- Large-scale and high-dimensional machine learning: both **d**, dimension of each observation, and **n**, number of observations, are large
- Consequence: computation of $\nabla f(\theta^t) = \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(\theta^t)$ is time-consuming.
- Idea behind **Stochastic Gradient Descent**: replace $\nabla f(\theta^t)$ with a *descent direction* d^t, faster to compute

$$d^t = \nabla f(\theta^t) + \epsilon^t \qquad \text{with} \qquad \mathbb{E}[\epsilon^t] = 0.$$

Vanilla SGD

- The usual version of SGD from Robbins and Monro (1951) writes

$$
\begin{aligned}
i_t &\sim \mathcal{U}[n] \\
d^t &= \nabla f(\theta^t) + \left(\nabla f_{i_t}(\theta^t) - \frac{1}{n} \sum_{j=1}^{n} f_j(\theta^t) \right) \\
&= \nabla f_{i_t}(\theta^t)
\end{aligned}
$$

- **ERM** framework with linear prediction $h(x, \theta) = \theta^\top \Phi(x)$,

$$
\nabla f_i(\theta) = \partial_2 \ell(y_i, \theta^\top \Phi(x_i)) \Phi(x_i),
$$

then computing d^t is **n times faster** than computing $\nabla f(\theta)$.

SGD's high variance

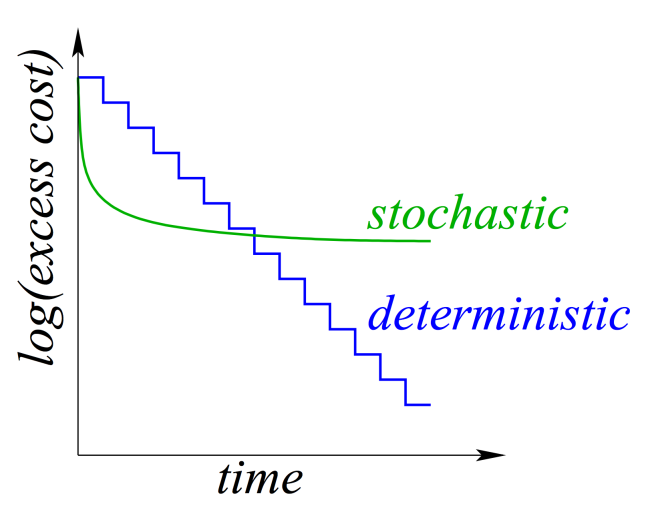


Figure 1: Picture borrowed from Francis Bach's presentations.

SGD's high variance

Assumptions

We assume f is L-smooth i.e.

$$\forall x, y : ||\nabla f(x) - \nabla f(y)||_2 \leq L||x - y||_2,$$

and f μ-strongly convex i.e.

$$\forall x, y : f(y) \geq f(x) + \nabla f(x)^\top (y - x) + \frac{\mu}{2}||y - x||_2^2$$

Convergence rates

$$\begin{aligned}\mathbb{E}\left(f(\theta^t) - f(\theta^*)\right) &= O(1/t) \text{ for Stochastic Gradient Descent}\\ &= O(\rho^t) \text{ with } \rho < 1 \text{ for Gradient Descent}\end{aligned}$$

The latter rate is called *linear convergence rate*.

Variance Reduction approach

- We want to compute $\mathbb{E}[X]$, and we can easily compute $\mathbb{E}[Y]$, where Y is highly correlated to X.

- We design the estimator $Z_\alpha = \alpha(X - Y) + \mathbb{E}[Y]$. Then,

$$\mathrm{Var}(Z_\alpha) = \alpha^2[\mathrm{Var}(X) + \mathrm{Var}(Y) - 2\mathrm{Cov}(X, Y)].$$

- When $\mathrm{Cov}(X, Y)$ is high enough, $\mathrm{Var}(Z_\alpha) \leq \mathrm{Var}(X)$, giving the method its name.

- The standard approach uses $\alpha = 1$, leading to an unbiased estimate $\mathbb{E}[Z_\alpha] = \mathbb{E}[X]$.

SGD with Variance Reduction

- Surprisingly enough, recent findings (M. Schmidt, N. Le Roux & F. Bach, 2012), (R. Johnson & T. Zhang, 2013), (A. Defazio, F. Bach & S. Lacoste-Julien, 2014) proved that reducing the variance in SGD enables reaching a **linear convergence rate**.
- Descent directions of these algorithms

$$\text{(SAG)} \qquad \theta \leftarrow \theta - \eta \left(\frac{\nabla f_i(\theta) - y_i}{n} + \frac{1}{n} \sum_{j=1}^{n} y_j \right),$$

$$\text{(SAGA)} \qquad \theta \leftarrow \theta - \eta \left(\nabla f_i(\theta) - y_i + \frac{1}{n} \sum_{j=1}^{n} y_j \right),$$

$$\text{(SVRG)} \qquad \theta \leftarrow \theta - \eta \left(\nabla f_i(\theta) - \nabla f_i(\tilde{\theta}) + \frac{1}{n} \sum_{j=1}^{n} \nabla f_j(\tilde{\theta}) \right).$$

- SAG's descent direction is biased ($\alpha = 1/n$), while SAGA's and SVRG's are unbiased ($\alpha = 1$)

Beyond Empirical Risk Minimization

Beyond Empirical Risk Minimization

Remarks

- These methods work well for problems where computing $\nabla f_i(\theta)$ is *n* **times faster** than computing $\nabla f(\theta)$.
- True for Generalized Linear Models since $\nabla f_i(\theta)$ is colinear to x_i.
- In more complex problems, computing $\nabla f_i(\theta)$ can be long as computing $\nabla f(\theta)$.

How to adapt the previous algorithms to this new case ?

Cox model

- The negative Cox partial log-likelihood takes the form

$$-\ell(\theta) = \frac{1}{|D|} \sum_{i \in D} \left[-\theta^\top x_i + \log \left(\sum_{j \in R_i} \exp(\theta^\top x_j) \right) \right]$$

- Likelihood and gradient

$$f_i(\theta) = -\theta^\top x_i + \log \left(\sum_{j \in R_i} \exp(\theta^\top x_j) \right)$$

$$\nabla f_i(\theta) = -x_i + \sum_{j \in R_i} \pi_\theta^i(j) x_j, \qquad \text{with} \qquad \pi_\theta^i(j) = \frac{\exp(\theta^\top x_j)}{\sum_{k \in R_i} \exp(\theta^\top x_k)}$$

Gradient of a subfunction as expectation

- Each subfunction's gradient ∇f_i can be expressed as the expectation of a random variable.

- Computing the exact expectation is expensive due to the summation over all possible configurations $k \in R_i$.

- **Our approach**: consider $\nabla f_i(\theta)$ the expectation of a random variable, and approximate it using MCMC:

$$\text{replace } \nabla f_i(\theta) = \mathbb{E}[G_i(\theta)] \qquad \text{with} \qquad \widehat{\nabla} f_i(\theta) = \widehat{G}_i(\theta)$$

HSVRG: Hybrid SVRG

Algorithm

Algorithm 1 Hybrid SVRG

for $k = 1$ **to** K **do**
    for $t = 0$ **to** $m - 1$ **do**
        Pick $i \sim \mathcal{U}[n]$
        $\widehat{\nabla} f_i(\theta^t) \leftarrow \textsc{ApproxMCMC}(\theta^t, i, N_k)$.
        $d^t = \widehat{\nabla} f_i(\theta^t) - \nabla f_i(\tilde{\theta}) + \frac{1}{n} \sum_{j=1}^{n} \nabla f_j(\tilde{\theta})$
        $\omega^{t+1} \leftarrow \theta^t - \gamma d^t$
        $\theta^{t+1} \leftarrow \mathrm{prox}_{\gamma g}(\omega^{t+1})$
    end for
    Update $\tilde{\theta} \leftarrow \frac{1}{m} \sum_{t=1}^{m} \theta^t$, $\theta^0 \leftarrow \tilde{\theta}$
    Compute $\nabla f_i(\tilde{\theta})$ for $i = 1, \ldots, n$
end for

ApproxMCMC

$\textsc{ApproxMCMC}(\theta^t, i, N_k)$ outputs an approximation of $\nabla f_i(\theta^t)$ using N_k iterations of a MCMC. We focused on two implementations:

- Independent Metropolis-Hastings[1] (IMH)
- Adaptative Importance Sampling (AIS)

[1] with uniform proposal

Theoretical Guarantees

Assumptions

Assumption

We assume that the bias and the expected squared error of the Monte Carlo error $\eta = \widehat{G}_i(\theta) - \mathbb{E}[G_i(\theta)]$ can be bounded this way

$$||\mathbb{E}_t[\eta]|| \leq \frac{C_1}{N_k} \text{ and } \mathbb{E}_t[||\eta||^2] \leq \frac{C_2}{N_k},$$

where N_k is the length of the Markov chain.

Proposition

Suppose that there exists $M > 0$ such that the proposal Q and the stationary distribution π satisfy $\pi(x) \leq MQ(x)$, for all x in the support of π. Then, the error η^t obtained by Algorithm IMH satisfies the previous assumption.

Remark: We can compute C_1 and C_2 from special cases (for Cox model, for instance).

Theorem

Theorem

Suppose that $F = f + g$ is μ-strongly convex. Consider Algorithm **HSVRG**, with a phase length m and a step-size $\gamma \in (0, \frac{1}{16L})$ satisfying

$$\rho = \frac{1}{m\gamma\mu(1 - 8L\gamma)} + \frac{8L\gamma(1 + 1/m)}{1 - 8L\gamma} < 1. \tag{1}$$

Assuming there exists $B > 0$ such that $\sup_{t \geq 0} ||\theta^t - \theta^*||_2 \leq B$, we have:

$$\mathbb{E}[F(\tilde{\theta}^K)] - F(\theta^*) \leq C\rho^K + D\sum_{k=1}^{K} \rho^{K-k}\frac{1}{N_k}, \tag{2}$$

where $C = F(\theta^0) - F(\theta^*)$, and $D = \frac{3\gamma C_2 + BC_1}{1 - 8L\gamma}$.

Corollary

Corollary

In the previous theorem, the choice $N_k = k^\alpha \rho^{-k}$ with $\alpha > 1$ gives

$$\mathbb{E}[F(\tilde{\theta}^K)] - F(\theta^*) \leq D' \rho^K,$$

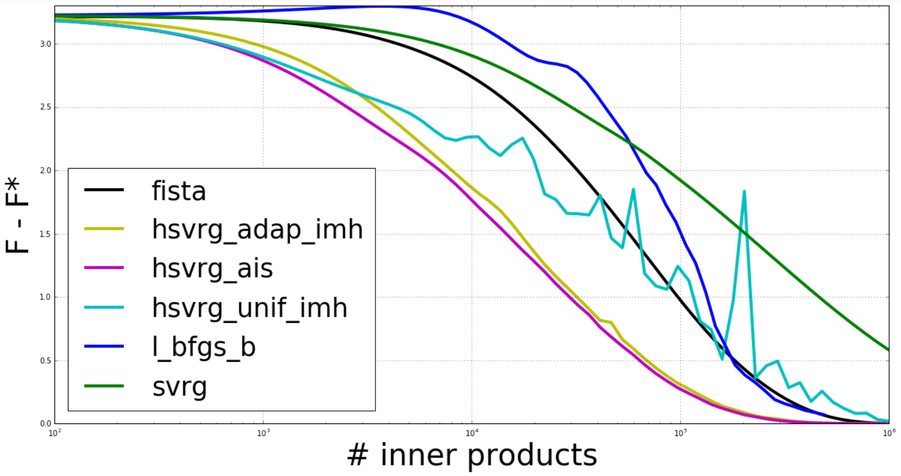
where $D' = F(\theta^0) - F(\theta^*) + D \sum_{k \geq 1} k^{-\alpha}$.

This entails that **HSVRG** achieves a **linear rate** under strong convexity.

Numerical Experiments

Experiments

We ran experiments on the Cox model with IMH with (uniform and adaptative proposal) and AIS (adaptative proposal).



Outlook: Conditional Random Fields

- CRFs model the conditional probability of a structured output $y \in \mathcal{Y}$ (such as a sequence of labels) given an input $x \in \mathcal{X}$ (such as a sequence of words) based on features $F(x, y)$ and parameter θ using

$$\mathbb{P}(y|x,\theta) = \frac{\exp(\theta^\top F(x,y))}{\sum_{y'} \exp(\theta^\top F(x,y'))}.$$

- Likelihood and gradient,

$$f_i(\theta) = -\log \mathbb{P}(y_i|x_i,\theta)$$
$$\nabla f_i(\theta) = -F(x_i,y_i) + \sum_{y' \in \mathcal{Y}} \mathbb{P}(y'|x_i,\theta) F(x_i,y')$$

Questions?

Adaptative Importance Sampling

- IMH with uniform proposal outputs an estimate with high variance.
- Use Normalized Importance Sampling in $\textsc{ApproxMCMC}$.

$$I = \mathbb{E}_p[f(X)] = \mathbb{E}_q\left[f(X)\frac{p(X)}{q(X)}\right]$$

$$\widehat{I}_n = \frac{1}{n}\sum_{k=1}^{n} f(X^{(k)})\frac{p(X^{(k)})}{q(X^{(k)})},\ \text{with } X^{(k)} \sim q$$

$$\widehat{J}_n = \sum_{k=1}^{n} f(X^{(k)})\frac{p(X^{(k)})}{q(X^{(k)})} \Big/ \sum_{k=1}^{n}\frac{p(X^{(k)})}{q(X^{(k)})},\ \text{with } X^{(k)} \sim q$$

- Use $\pi_{\tilde{\theta}}$ as adaptative proposal, where $\tilde{\theta}$ is updated every phase.

Details for CRFs

- Apply this new $\textsc{ApproxMCMC}(\theta, i, N)$ to CRF outputs

$$\widehat{J}_n = -F(x_i, y_i) + \sum_{k=1}^{N} \frac{\exp((\theta - \tilde{\theta})^\top F(x_i, y^{(k)}))}{\sum_{j=1}^{N} \exp((\theta - \tilde{\theta})^\top F(x_i, y^{(j)}))} F(x_i, y^{(k)})$$

- The sequence $(y^{(k)})$ is sampled from $\mathbb{P}(\bullet | x_i, \tilde{\theta})$.
- We remind the true subgradient is

$$\nabla f_i(\theta) = -F(x_i, y_i) + \sum_{y \in \mathcal{Y}} \frac{\exp((\theta - \tilde{\theta})^\top F(x_i, y))}{\sum_{y'} \exp((\theta - \tilde{\theta})^\top F(x_i, y'))} F(x_i, y)$$