Beyond stochastic gradient descent

Francis Bach

INRIA - Ecole Normale Supérieure, Paris, France

Joint work with Nicolas Le Roux, Mark Schmidt and Eric Moulines - July 2013

Context

Machine learning for "big data"

- **Large-scale machine learning**: **large p, large n, large k**
  - p : dimension of each observation (input)
  - n : number of observations
  - k : number of tasks (dimension of outputs)

- **Examples**: computer vision, bioinformatics, text processing

- **Ideal running-time complexity**: $O(pn + kn)$

Context
Machine learning for “big data”

- **Large-scale machine learning**: **large p, large n, large k**
  - p : dimension of each observation (input)
  - n : number of observations
  - k : number of tasks (dimension of outputs)

- **Examples**: computer vision, bioinformatics, text processing

- **Ideal running-time complexity**: $O(pn + kn)$

- **Going back to simple methods**
  - Stochastic gradient methods (Robbins and Monro, 1951)
  - Mixing statistics and optimization

Context

Machine learning for “big data”

- **Large-scale machine learning**: **large p, large n, large k**
  - p : dimension of each observation (input)
  - n : number of observations
  - k : number of tasks (dimension of outputs)
- **Examples**: computer vision, bioinformatics, text processing
- **Ideal running-time complexity**: $O(pn + kn)$
- **Going back to simple methods**
  - Stochastic gradient methods (Robbins and Monro, 1951)
  - Mixing statistics and optimization
  - **Using smoothness to go beyond stochastic gradient descent**

Outline

- **Introduction: stochastic approximation algorithms**
  - Supervised machine learning and convex optimization
  - Stochastic gradient and averaging
  - Strongly convex vs. non-strongly convex
- **Fast convergence through smoothness and constant step-sizes**
  - Online Newton steps (Bach and Moulines, 2013)
  - $O(1/n)$ convergence rate for all convex functions
- **More than a single pass through the data**
  - Stochastic average gradient (Le Roux, Schmidt, and Bach, 2012)
  - Linear (exponential) convergence rate for strongly convex functions

Supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$, **i.i.d.**
- Prediction as a linear function $\langle \theta, \Phi(x) \rangle$ of features $\Phi(x) \in \mathbb{R}^p$
- **(regularized) empirical risk minimization**: find $\hat{\theta}$ solution of

$$\min_{\theta \in \mathbb{R}^p} \quad \frac{1}{n} \sum_{i=1}^{n} \ell\big(y_i, \langle \theta, \Phi(x_i) \rangle\big) \quad + \quad \mu \Omega(\theta)$$

convex data fitting term + regularizer

Supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$, **i.i.d.**
- Prediction as a linear function $\langle \theta, \Phi(x) \rangle$ of features $\Phi(x) \in \mathbb{R}^p$
- **(regularized) empirical risk minimization**: find $\hat{\theta}$ solution of

$$\min_{\theta \in \mathbb{R}^p} \quad \frac{1}{n} \sum_{i=1}^{n} \ell\big(y_i, \langle \theta, \Phi(x_i) \rangle\big) \quad + \quad \mu \Omega(\theta)$$

convex data fitting term + regularizer

- Empirical risk: $\hat{f}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, \langle \theta, \Phi(x_i) \rangle)$ training cost
- Expected risk: $f(\theta) = \mathbb{E}_{(x,y)} \ell(y, \langle \theta, \Phi(x) \rangle)$ testing cost
- **Two fundamental questions**: (1) computing $\hat{\theta}$ and (2) analyzing $\hat{\theta}$

Supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$, **i.i.d.**
- Prediction as a linear function $\langle \theta, \Phi(x) \rangle$ of features $\Phi(x) \in \mathbb{R}^p$
- **(regularized) empirical risk minimization**: find $\hat{\theta}$ solution of

$$\min_{\theta \in \mathbb{R}^p} \quad \frac{1}{n} \sum_{i=1}^{n} \ell\big(y_i, \langle \theta, \Phi(x_i) \rangle\big) \quad + \quad \mu \Omega(\theta)$$

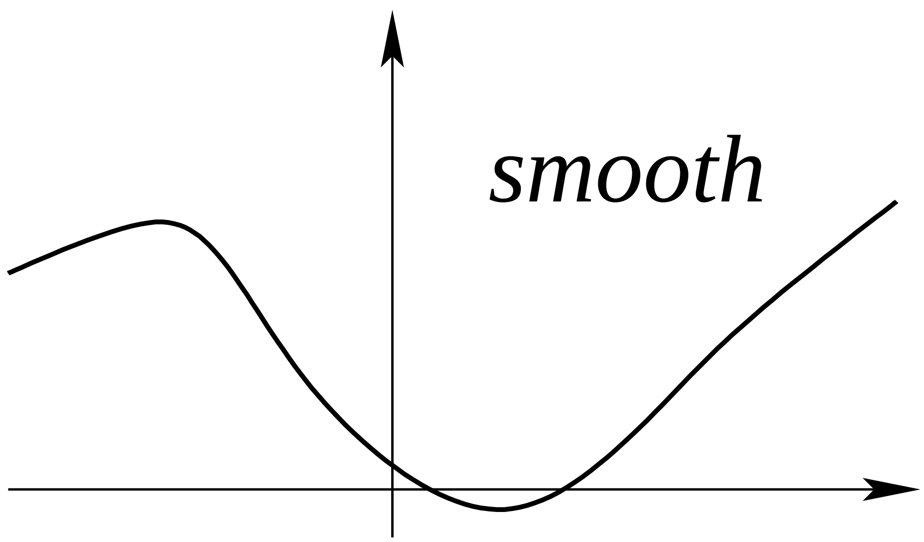
convex data fitting term + regularizer

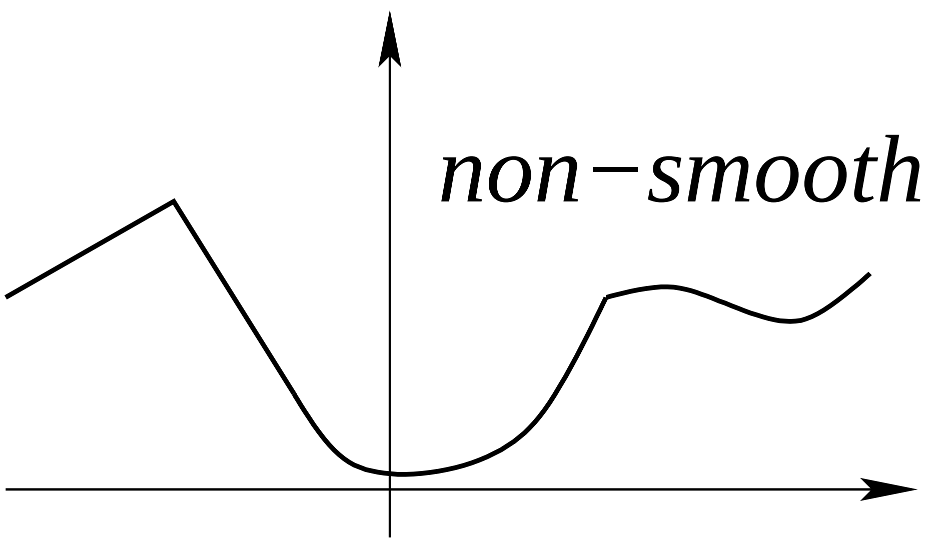
- Empirical risk: $\hat{f}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, \langle \theta, \Phi(x_i) \rangle)$ training cost
- Expected risk: $f(\theta) = \mathbb{E}_{(x,y)} \ell(y, \langle \theta, \Phi(x) \rangle)$ testing cost
- **Two fundamental questions**: (1) computing $\hat{\theta}$ and (2) analyzing $\hat{\theta}$
  - May be tackled simultaneously

Smoothness and strong convexity

- A function $g : \mathbb{R}^p \rightarrow \mathbb{R}$ is L-smooth if and only if it is twice differentiable and

$$\forall \theta \in \mathbb{R}^p, \ g''(\theta) \preccurlyeq L \cdot \mathrm{Id}$$





Smoothness and strong convexity

- A function $g : \mathbb{R}^p \to \mathbb{R}$ is L-smooth if and only if it is twice differentiable and

$$\forall \theta \in \mathbb{R}^p, \; g''(\theta) \preccurlyeq L \cdot \mathrm{Id}$$

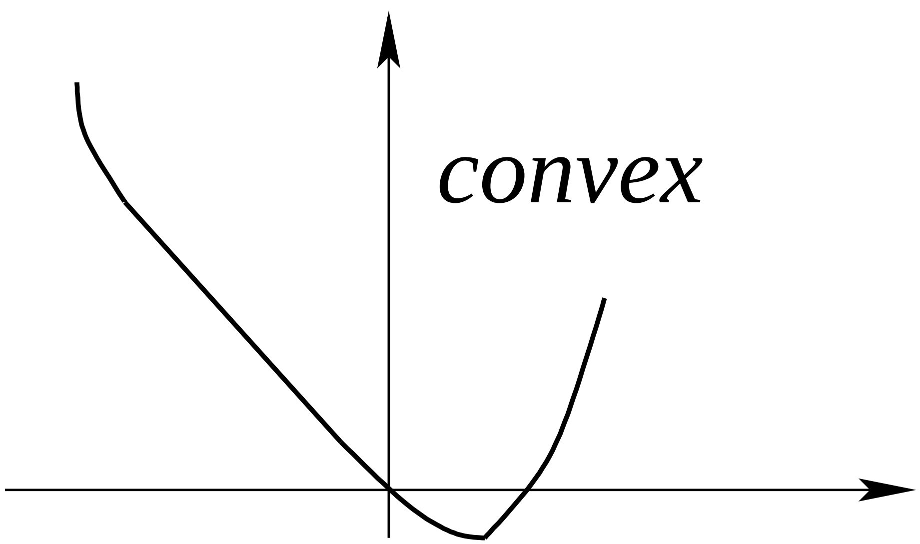
- **Machine learning**
  - with $g(\theta) = \frac{1}{n} \sum_{i=1}^n \ell(y_i, \langle \theta, \Phi(x_i) \rangle)$
  - Hessian $\approx$ covariance matrix $\frac{1}{n} \sum_{i=1}^n \Phi(x_i) \otimes \Phi(x_i)$
  - Bounded data

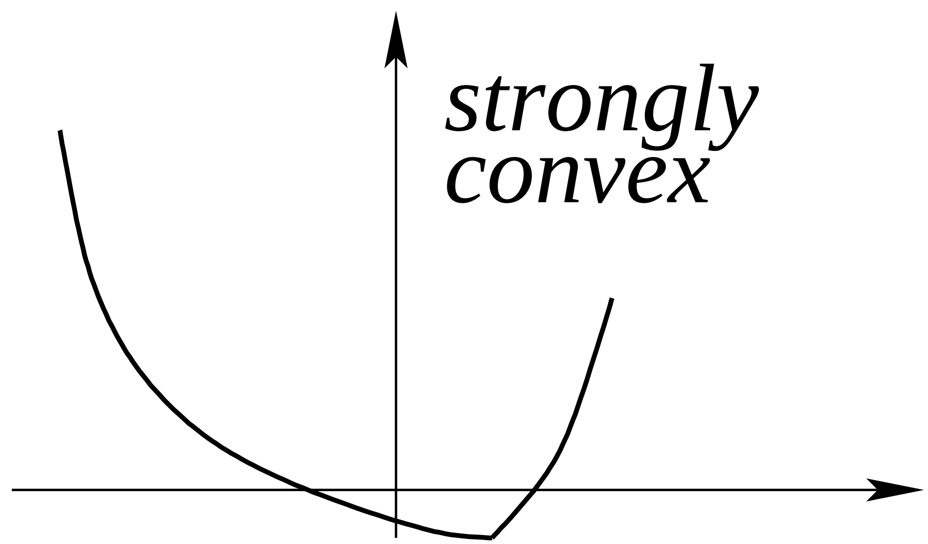
Smoothness and strong convexity

- A function $g : \mathbb{R}^p \to \mathbb{R}$ is μ-strongly convex if and only if

$$\forall \theta_1, \theta_2 \in \mathbb{R}^p, \;\; g(\theta_1) \geqslant g(\theta_2) + \langle g'(\theta_2), \theta_1 - \theta_2 \rangle + \frac{\mu}{2} \|\theta_1 - \theta_2\|^2$$

- If g is twice differentiable: $\forall \theta \in \mathbb{R}^p, \;\; g''(\theta) \succcurlyeq \mu \cdot \mathrm{Id}$





Smoothness and strong convexity

- A function $g : \mathbb{R}^p \to \mathbb{R}$ is μ-strongly convex if and only if

$$\forall \theta_1, \theta_2 \in \mathbb{R}^p, \;\; g(\theta_1) \geqslant g(\theta_2) + \langle g'(\theta_2), \theta_1 - \theta_2 \rangle + \frac{\mu}{2} \|\theta_1 - \theta_2\|^2$$

- If g is twice differentiable: $\forall \theta \in \mathbb{R}^p, \;\; g''(\theta) \succcurlyeq \mu \cdot \mathrm{Id}$

- **Machine learning**
  - with $g(\theta) = \frac{1}{n} \sum_{i=1}^n \ell(y_i, \langle \theta, \Phi(x_i) \rangle)$
  - Hessian $\approx$ covariance matrix $\frac{1}{n} \sum_{i=1}^n \Phi(x_i) \otimes \Phi(x_i)$
  - Data with invertible covariance matrix (low correlation/dimension)

Smoothness and strong convexity

- A function $g : \mathbb{R}^p \to \mathbb{R}$ is μ-strongly convex if and only if

$$\forall \theta_1, \theta_2 \in \mathbb{R}^p, \ \ g(\theta_1) \geqslant g(\theta_2) + \langle g'(\theta_2), \theta_1 - \theta_2 \rangle + \frac{\mu}{2} \|\theta_1 - \theta_2\|^2$$

- If g is twice differentiable: $\forall \theta \in \mathbb{R}^p, \ \ g''(\theta) \succcurlyeq \mu \cdot \mathrm{Id}$

- **Machine learning**
  - with $g(\theta) = \frac{1}{n} \sum_{i=1}^n \ell(y_i, \langle \theta, \Phi(x_i) \rangle)$
  - Hessian $\approx$ covariance matrix $\frac{1}{n} \sum_{i=1}^n \Phi(x_i) \otimes \Phi(x_i)$
  - Data with invertible covariance matrix (low correlation/dimension)

- **Adding regularization by** $\frac{\mu}{2} \|\theta\|^2$
  - creates additional bias unless μ is small

Iterative methods for minimizing smooth functions

- **Assumption**: g convex and smooth on $\mathbb{R}^p$

- **Gradient descent**: $\theta_t = \theta_{t-1} - \gamma_t \, g'(\theta_{t-1})$
  - $O(1/t)$ convergence rate for convex functions
  - $O(e^{-\rho t})$ convergence rate for strongly convex functions

- **Newton method**: $\theta_t = \theta_{t-1} - g''(\theta_{t-1})^{-1} g'(\theta_{t-1})$
  - $O\big(e^{-\rho 2^t}\big)$ convergence rate

Iterative methods for minimizing smooth functions

- **Assumption**: g convex and smooth on $\mathbb{R}^p$

- **Gradient descent**: $\theta_t = \theta_{t-1} - \gamma_t \, g'(\theta_{t-1})$
  - $O(1/t)$ convergence rate for convex functions
  - $O(e^{-\rho t})$ convergence rate for strongly convex functions

- **Newton method**: $\theta_t = \theta_{t-1} - g''(\theta_{t-1})^{-1} g'(\theta_{t-1})$
  - $O\big(e^{-\rho 2^t}\big)$ convergence rate

- **Key insights from Bottou and Bousquet (2008)**
  1. In machine learning, no need to optimize below statistical error
  2. In machine learning, cost functions are averages

$\Rightarrow$ **Stochastic approximation**

Stochastic approximation

- **Goal**: Minimizing a function f defined on $\mathbb{R}^p$
  - given only unbiased estimates $f'_n(\theta_n)$ of its gradients $f'(\theta_n)$ at certain points $\theta_n \in \mathbb{R}^p$

- **Stochastic approximation**
  - Observation of $f'_n(\theta_n) = f'(\theta_n) + \varepsilon_n$, with ε_n = i.i.d. noise
  - Non-convex problems

Stochastic approximation

- **Goal**: Minimizing a function f defined on $\mathbb{R}^p$
  - given only unbiased estimates $f'_n(\theta_n)$ of its gradients $f'(\theta_n)$ at certain points $\theta_n \in \mathbb{R}^p$

- **Stochastic approximation**
  - Observation of $f'_n(\theta_n) = f'(\theta_n) + \varepsilon_n$, with ε_n = i.i.d. noise
  - Non-convex problems

- **Machine learning - statistics**
  - **loss for a single pair of observations**: $\boxed{f_n(\theta) = \ell(y_n, \langle \theta, \Phi(x_n) \rangle)}$
  - $f(\theta) = \mathbb{E} f_n(\theta) = \mathbb{E}\, \ell(y_n, \langle \theta, \Phi(x_n) \rangle)$ = **generalization error**
  - Expected gradient: $f'(\theta) = \mathbb{E} f'_n(\theta) = \mathbb{E} \left\{ \ell'(y_n, \langle \theta, \Phi(x_n) \rangle)\, \Phi(x_n) \right\}$

Convex stochastic approximation

- **Key assumption**: smoothness and/or strongly convexity

Convex stochastic approximation

- **Key assumption**: smoothness and/or strongly convexity
- **Key algorithm:** stochastic gradient descent (a.k.a. Robbins-Monro)

$$\boxed{\theta_n = \theta_{n-1} - \gamma_n f_n'(\theta_{n-1})}$$

  - Polyak-Ruppert averaging: $\bar{\theta}_n = \frac{1}{n+1}\sum_{k=0}^{n}\theta_k$
  - Which learning rate sequence γ_n? Classical setting: $\boxed{\gamma_n = Cn^{-\alpha}}$

Convex stochastic approximation
Existing work

- **Known global minimax rates of convergence for non-smooth problems** (Nemirovsky and Yudin, 1983; Agarwal et al., 2012)
  - Strongly convex: $O((\mu n)^{-1})$
    Attained by averaged stochastic gradient descent with $\gamma_n \propto (\mu n)^{-1}$
  - Non-strongly convex: $O(n^{-1/2})$
    Attained by averaged stochastic gradient descent with $\gamma_n \propto n^{-1/2}$

Convex stochastic approximation
Existing work

- **Known global minimax rates of convergence for non-smooth problems** (Nemirovsky and Yudin, 1983; Agarwal et al., 2012)
  - Strongly convex: $O((\mu n)^{-1})$
    Attained by averaged stochastic gradient descent with $\gamma_n \propto (\mu n)^{-1}$
  - Non-strongly convex: $O(n^{-1/2})$
    Attained by averaged stochastic gradient descent with $\gamma_n \propto n^{-1/2}$

- **Many contributions in optimization and online learning:** Bottou and Le Cun (2005); Bottou and Bousquet (2008); Hazan et al. (2007); Shalev-Shwartz and Srebro (2008); Shalev-Shwartz et al. (2007, 2009); Xiao (2010); Duchi and Singer (2009); Nesterov and Vial (2008); Nemirovski et al. (2009)

Convex stochastic approximation
Existing work

- **Known global minimax rates of convergence for non-smooth problems** (Nemirovsky and Yudin, 1983; Agarwal et al., 2012)
  - Strongly convex: $O((\mu n)^{-1})$
    Attained by averaged stochastic gradient descent with $\gamma_n \propto (\mu n)^{-1}$
  - Non-strongly convex: $O(n^{-1/2})$
    Attained by averaged stochastic gradient descent with $\gamma_n \propto n^{-1/2}$

- **Asymptotic analysis of averaging** (Polyak and Juditsky, 1992; Ruppert, 1988)
  - All step sizes $\gamma_n = Cn^{-\alpha}$ with $\alpha \in (1/2, 1)$ lead to $O(n^{-1})$ for smooth strongly convex problems

Convex stochastic approximation
Existing work

- **Known global minimax rates of convergence for non-smooth problems** (Nemirovsky and Yudin, 1983; Agarwal et al., 2012)
  - Strongly convex: $O((\mu n)^{-1})$
    Attained by averaged stochastic gradient descent with $\gamma_n \propto (\mu n)^{-1}$
  - Non-strongly convex: $O(n^{-1/2})$
    Attained by averaged stochastic gradient descent with $\gamma_n \propto n^{-1/2}$
- **Asymptotic analysis of averaging** (Polyak and Juditsky, 1992; Ruppert, 1988)
  - All step sizes $\gamma_n = Cn^{-\alpha}$ with $\alpha \in (1/2, 1)$ lead to $O(n^{-1})$ for smooth strongly convex problems
- **A single algorithm with global adaptive convergence rate for smooth problems?**

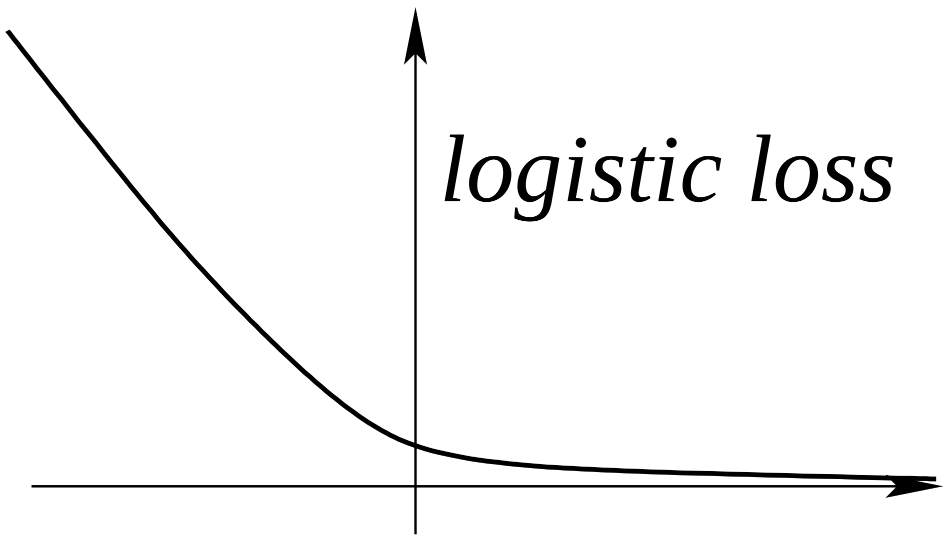
Adaptive algorithm for logistic regression

- **Logistic regression**: $(\Phi(x_n), y_n) \in \mathbb{R}^p \times \{-1, 1\}$
  - Single data point: $f_n(\theta) = \log(1 + \exp(-y_n\langle\theta, \Phi(x_n)\rangle))$
  - Generalization error: $f(\theta) = \mathbb{E} f_n(\theta)$

Adaptive algorithm for logistic regression

- **Logistic regression**: $(\Phi(x_n), y_n) \in \mathbb{R}^p \times \{-1, 1\}$
  - Single data point: $f_n(\theta) = \log(1 + \exp(-y_n \langle \theta, \Phi(x_n) \rangle))$
  - Generalization error: $f(\theta) = \mathbb{E} f_n(\theta)$

- **Cannot be strongly convex** $\Rightarrow$ local strong convexity
  - unless restricted to $|\langle \theta, \Phi(x_n) \rangle| \leqslant M$ (and with constants e^M)
  - $\mu =$ lowest eigenvalue of the Hessian at the optimum $f''(\theta_*)$



Adaptive algorithm for logistic regression

- **Logistic regression**: $(\Phi(x_n), y_n) \in \mathbb{R}^p \times \{-1, 1\}$
  - Single data point: $f_n(\theta) = \log(1 + \exp(-y_n \langle \theta, \Phi(x_n) \rangle))$
  - Generalization error: $f(\theta) = \mathbb{E} f_n(\theta)$

- **Cannot be strongly convex** $\Rightarrow$ local strong convexity
  - unless restricted to $|\langle \theta, \Phi(x_n) \rangle| \leqslant M$ (and with constants e^M)
  - $\mu =$ lowest eigenvalue of the Hessian at the optimum $f''(\theta_*)$

- n **steps of averaged SGD with constant step-size** $1/\left(2R^2\sqrt{n}\right)$
  - with $R =$ radius of data (Bach, 2013):

$$\mathbb{E} f(\bar{\theta}_n) - f(\theta_*) \leqslant \min\left\{\frac{1}{\sqrt{n}}, \frac{R^2}{n\mu}\right\}\left(15 + 5R\|\theta_0 - \theta_*\|\right)^4$$

  - Proof based on self-concordance (Nesterov and Nemirovski, 1994)

Outline

- **Introduction: stochastic approximation algorithms**
  - Supervised machine learning and convex optimization
  - Stochastic gradient and averaging
  - Strongly convex vs. non-strongly convex
- **Fast convergence through smoothness and constant step-sizes**
  - Online Newton steps (Bach and Moulines, 2013)
  - $O(1/n)$ convergence rate for all convex functions
- **More than a single pass through the data**
  - Stochastic average gradient (Le Roux, Schmidt, and Bach, 2012)
  - Linear (exponential) convergence rate for strongly convex functions

Least-mean-square algorithm

- **Least-squares**: $f(\theta) = \frac{1}{2}\mathbb{E}\big[(y_n - \langle \Phi(x_n), \theta\rangle)^2\big]$ with $\theta \in \mathbb{R}^p$
  - SGD = least-mean-square algorithm (see, e.g., Macchi, 1995)
  - usually studied without averaging and decreasing step-sizes
  - with strong convexity assumption $\mathbb{E}\big[\Phi(x_n) \otimes \Phi(x_n)\big] = H \succcurlyeq \mu \cdot \mathrm{Id}$

Least-mean-square algorithm

- **Least-squares**: $f(\theta) = \frac{1}{2}\mathbb{E}\big[(y_n - \langle \Phi(x_n), \theta\rangle)^2\big]$ with $\theta \in \mathbb{R}^p$
  - SGD = least-mean-square algorithm (see, e.g., Macchi, 1995)
  - usually studied without averaging and decreasing step-sizes
  - with strong convexity assumption $\mathbb{E}\big[\Phi(x_n) \otimes \Phi(x_n)\big] = H \succcurlyeq \mu \cdot \mathrm{Id}$

- **New analysis for averaging and constant step-size** $\gamma = 1/(4R^2)$
  - Assume $\|\Phi(x_n)\| \leqslant R$ and $|y_n - \langle \Phi(x_n), \theta_*\rangle| \leqslant \sigma$ almost surely
  - No assumption regarding lowest eigenvalues of H
  - Main result: $\boxed{\mathbb{E} f(\bar{\theta}_{n-1}) - f(\theta_*) \leqslant \frac{2}{n}\Big[\sigma\sqrt{p} + R\|\theta_0 - \theta_*\|\Big]^2}$

- **Matches statistical lower bound** (Tsybakov, 2003)

Markov chain interpretation of constant step sizes

- LMS recursion for $f_n(\theta) = \frac{1}{2}\big(y_n - \langle \Phi(x_n), \theta \rangle\big)^2$

$$\theta_n = \theta_{n-1} - \gamma\big(\langle \Phi(x_n), \theta_{n-1} \rangle - y_n\big)\Phi(x_n)$$

- The sequence $(\theta_n)_n$ is a homogeneous Markov chain
  - convergence to a stationary distribution π_γ
  - with expectation $\bar{\theta}_\gamma \stackrel{\text{def}}{=} \int \theta \pi_\gamma(\mathrm{d}\theta)$

Markov chain interpretation of constant step sizes

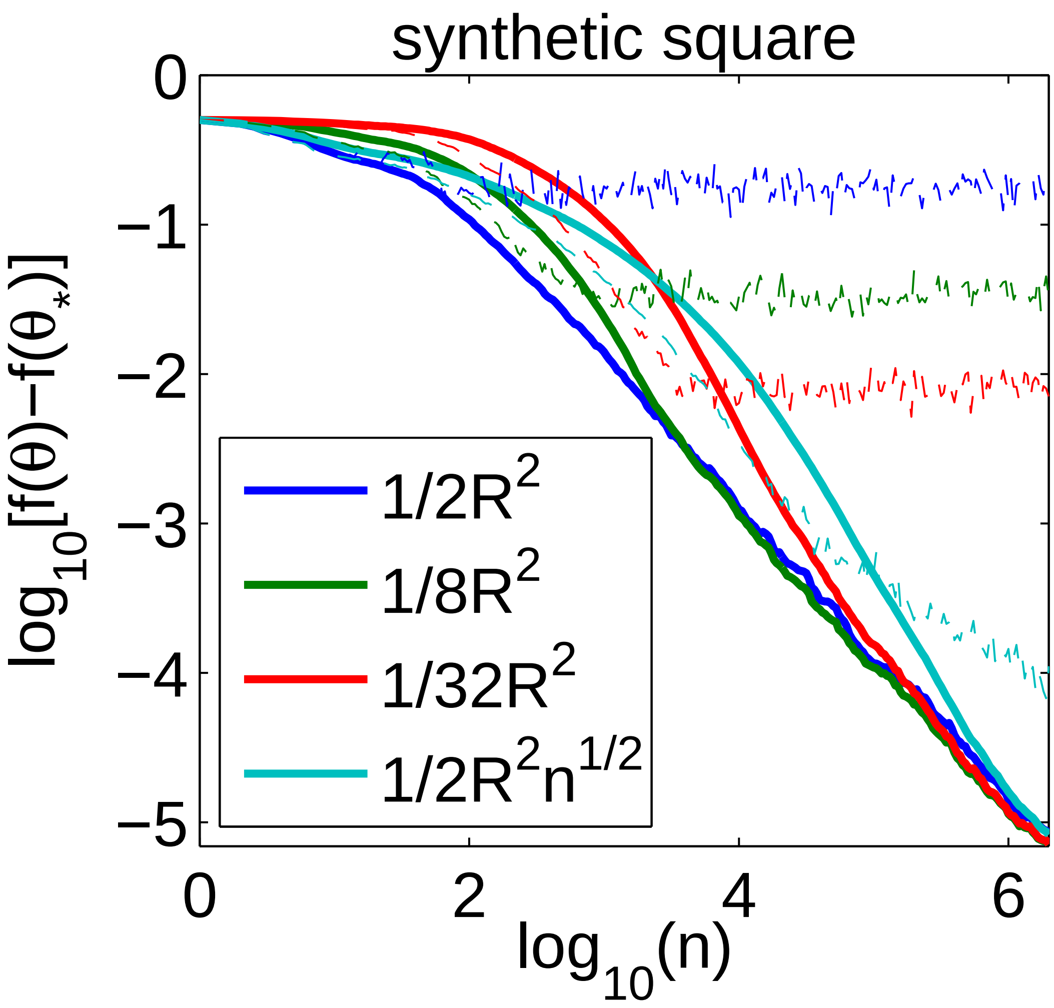
- LMS recursion for $f_n(\theta) = \frac{1}{2}\big(y_n - \langle \Phi(x_n), \theta\rangle\big)^2$

$$\theta_n = \theta_{n-1} - \gamma\big(\langle \Phi(x_n), \theta_{n-1}\rangle - y_n\big)\Phi(x_n)$$

- The sequence $(\theta_n)_n$ is a homogeneous Markov chain
  - convergence to a stationary distribution π_γ
  - with expectation $\bar{\theta}_\gamma \stackrel{\text{def}}{=} \int \theta \pi_\gamma(\mathrm{d}\theta)$
- **For least-squares, $\bar{\theta}_\gamma = \theta_*$**
  - θ_n does not converge to θ_* but oscillates around it
  - oscillations of order $\sqrt{\gamma}$
- **Ergodic theorem:**
  - Averaged iterates converge to $\bar{\theta}_\gamma = \theta_*$ at rate $O(1/n)$

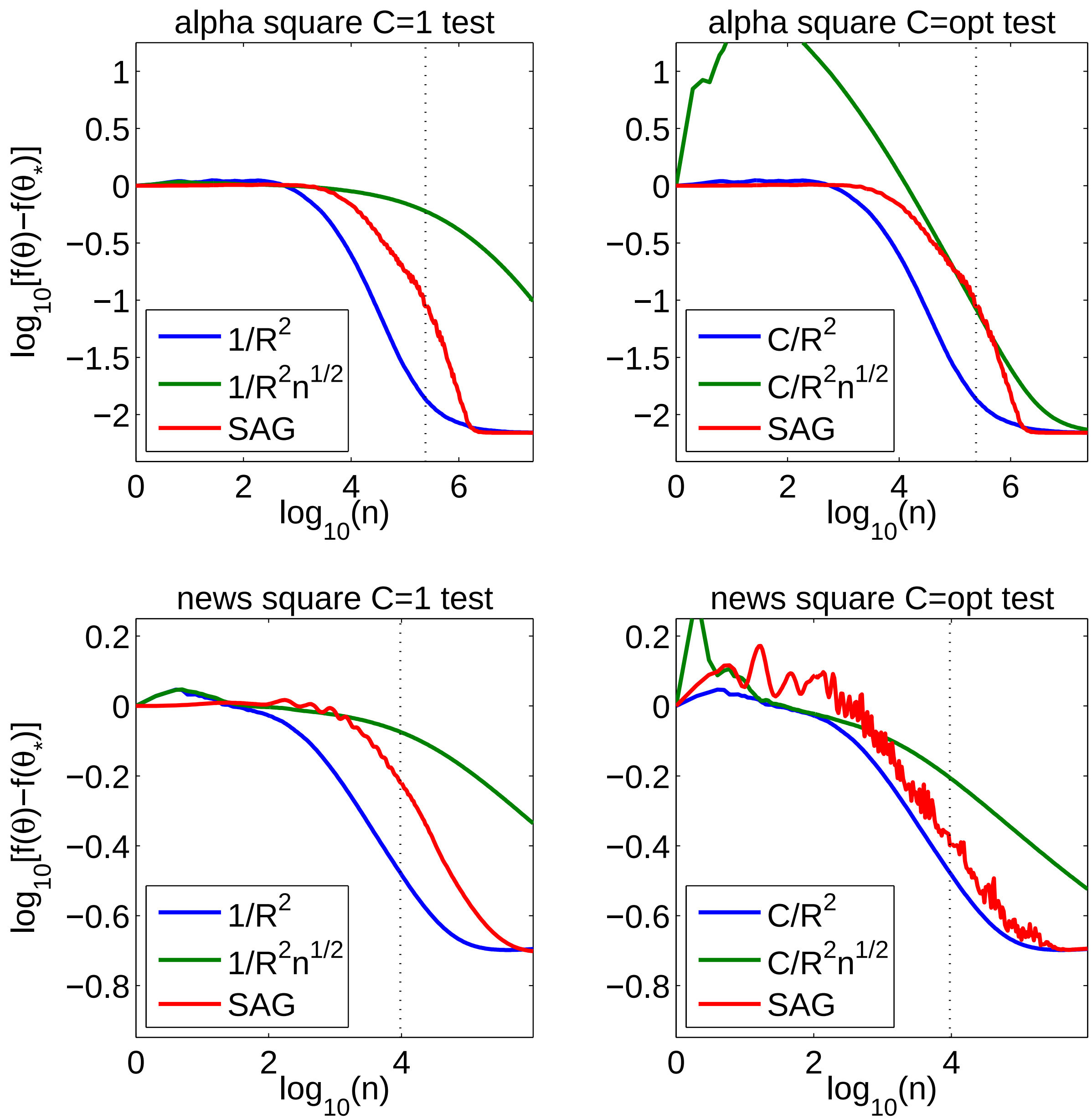
Simulations - synthetic examples

- Gaussian distributions - $p = 20$



Simulations - benchmarks

- *alpha* $(p = 500,\ n = 500\ 000)$, *news* $(p = 1\ 300\ 000,\ n = 20\ 000)$



Beyond least-squares - Markov chain interpretation

- Recursion $\theta_n = \theta_{n-1} - \gamma f_n'(\theta_{n-1})$ also defines a Markov chain
  - Stationary distribution π_γ such that $\int f'(\theta)\pi_\gamma(\mathrm{d}\theta) = 0$
  - When f' is not linear, $f'(\int \theta\pi_\gamma(\mathrm{d}\theta)) \neq \int f'(\theta)\pi_\gamma(\mathrm{d}\theta) = 0$

Beyond least-squares - Markov chain interpretation

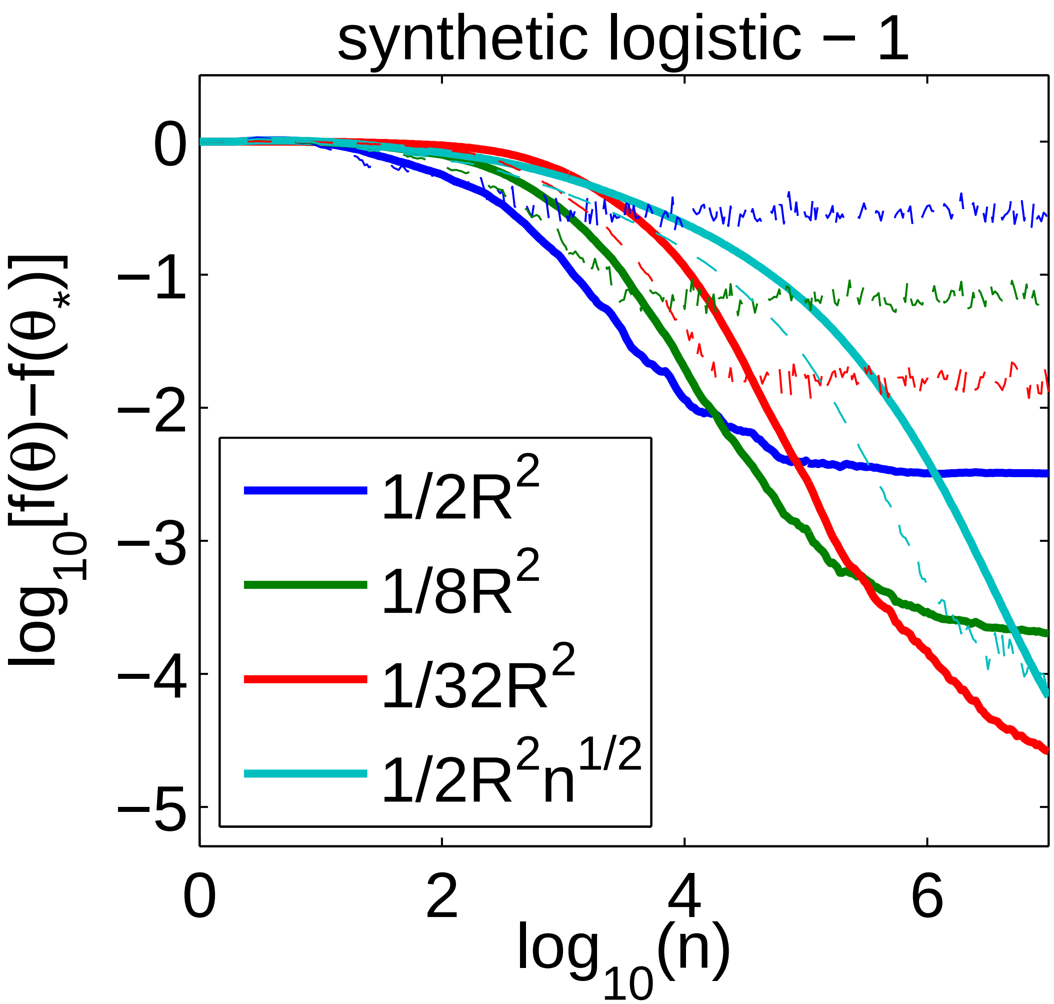
- Recursion $\theta_n = \theta_{n-1} - \gamma f_n'(\theta_{n-1})$ also defines a Markov chain
  - Stationary distribution π_γ such that $\int f'(\theta)\pi_\gamma(\mathrm{d}\theta) = 0$
  - When f' is not linear, $f'(\int \theta \pi_\gamma(\mathrm{d}\theta)) \neq \int f'(\theta)\pi_\gamma(\mathrm{d}\theta) = 0$

- **θ_n oscillates around the wrong value $\bar{\theta}_\gamma \neq \theta_*$**
  - moreover, $\|\theta_* - \theta_n\| = O_p(\sqrt{\gamma})$

- **Ergodic theorem**
  - averaged iterates converge to $\bar{\theta}_\gamma \neq \theta_*$ at rate $O(1/n)$
  - moreover, $\|\theta_* - \bar{\theta}_\gamma\| = O(\gamma)$ (Bach, 2013)

Simulations - synthetic examples

- Gaussian distributions - $p = 20$



Restoring convergence through online Newton steps

- The Newton step for $f = \mathbb{E} f_n(\theta) \stackrel{\text{def}}{=} \mathbb{E}\big[\ell(y_n, \langle \theta, \Phi(x_n)\rangle)\big]$ at $\tilde{\theta}$ is equivalent to minimizing the quadratic approximation

$$\begin{aligned}
g(\theta) &= f(\tilde{\theta}) + \langle f'(\tilde{\theta}), \theta - \tilde{\theta}\rangle + \tfrac{1}{2}\langle \theta - \tilde{\theta}, f''(\tilde{\theta})(\theta - \tilde{\theta})\rangle \\
&= f(\tilde{\theta}) + \langle \mathbb{E} f_n'(\tilde{\theta}), \theta - \tilde{\theta}\rangle + \tfrac{1}{2}\langle \theta - \tilde{\theta}, \mathbb{E} f_n''(\tilde{\theta})(\theta - \tilde{\theta})\rangle \\
&= \mathbb{E}\Big[f(\tilde{\theta}) + \langle f_n'(\tilde{\theta}), \theta - \tilde{\theta}\rangle + \tfrac{1}{2}\langle \theta - \tilde{\theta}, f_n''(\tilde{\theta})(\theta - \tilde{\theta})\rangle\Big]
\end{aligned}$$

Restoring convergence through online Newton steps

- The Newton step for $f = \mathbb{E} f_n(\theta) \stackrel{\text{def}}{=} \mathbb{E}\big[\ell(y_n, \langle \theta, \Phi(x_n)\rangle)\big]$ at $\tilde{\theta}$ is equivalent to minimizing the quadratic approximation

$$\begin{aligned} g(\theta) &= f(\tilde{\theta}) + \langle f'(\tilde{\theta}), \theta - \tilde{\theta}\rangle + \tfrac{1}{2}\langle \theta - \tilde{\theta}, f''(\tilde{\theta})(\theta - \tilde{\theta})\rangle \\ &= f(\tilde{\theta}) + \langle \mathbb{E} f_n'(\tilde{\theta}), \theta - \tilde{\theta}\rangle + \tfrac{1}{2}\langle \theta - \tilde{\theta}, \mathbb{E} f_n''(\tilde{\theta})(\theta - \tilde{\theta})\rangle \\ &= \mathbb{E}\Big[f(\tilde{\theta}) + \langle f_n'(\tilde{\theta}), \theta - \tilde{\theta}\rangle + \tfrac{1}{2}\langle \theta - \tilde{\theta}, f_n''(\tilde{\theta})(\theta - \tilde{\theta})\rangle\Big] \end{aligned}$$

- **Complexity of least-mean-square recursion for g is $O(p)$**

$$\theta_n = \theta_{n-1} - \gamma\big[f_n'(\tilde{\theta}) + f_n''(\tilde{\theta})(\theta_{n-1} - \tilde{\theta})\big]$$

  – $f_n''(\tilde{\theta}) = \ell''(y_n, \langle \tilde{\theta}, \Phi(x_n)\rangle)\Phi(x_n) \otimes \Phi(x_n)$ has rank one
  – New online Newton step without computing/inverting Hessians

Choice of support point for online Newton step

- **Two-stage procedure**

(1) Run $n/2$ iterations of averaged SGD to obtain $\tilde{\theta}$
(2) Run $n/2$ iterations of averaged constant step-size LMS

- Reminiscent of one-step estimators (see, e.g., Van der Vaart, 2000)
- Provable convergence rate of $O(p/n)$ for logistic regression
- Additional assumptions but no strong convexity

Choice of support point for online Newton step

- **Two-stage procedure**

(1) Run $n/2$ iterations of averaged SGD to obtain $\tilde{\theta}$
(2) Run $n/2$ iterations of averaged constant step-size LMS

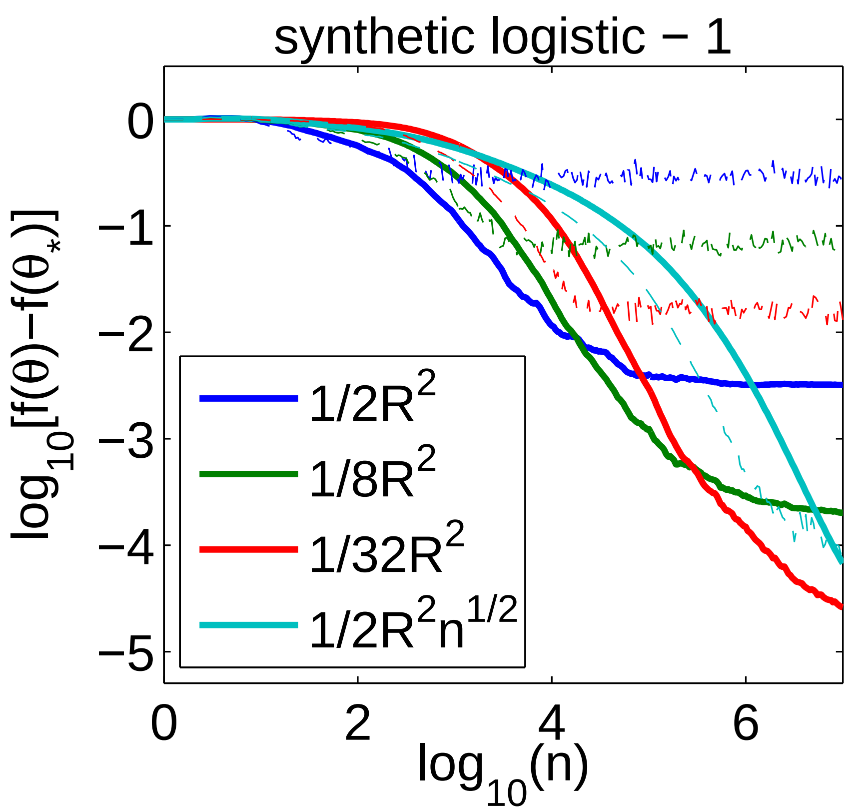
  - Reminiscent of one-step estimators (see, e.g., Van der Vaart, 2000)
  - Provable convergence rate of $O(p/n)$ for logistic regression
  - Additional assumptions but no strong convexity

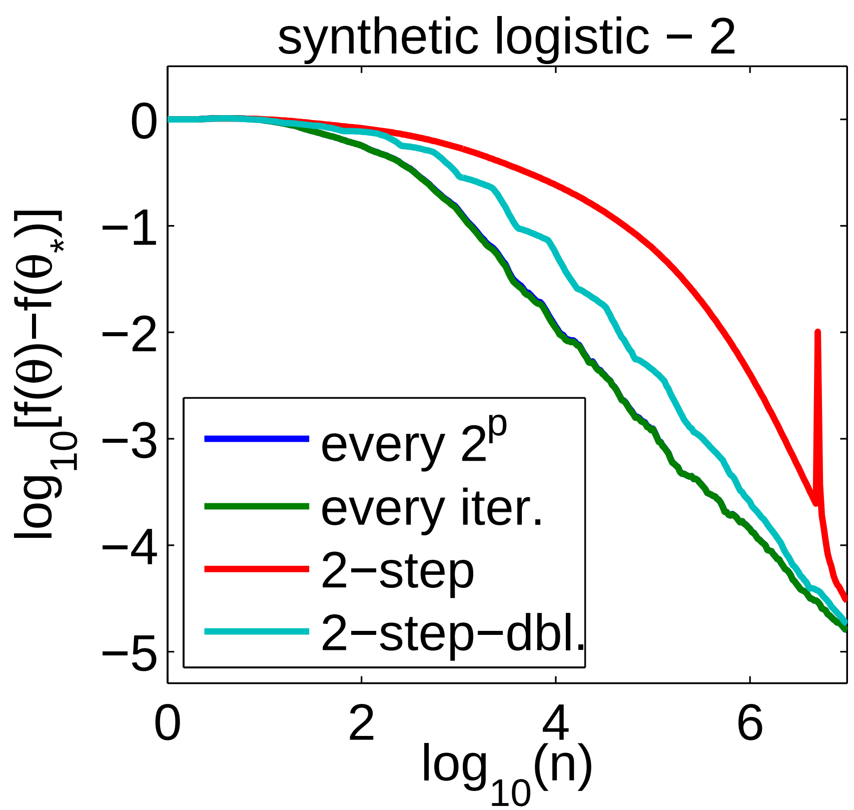
- **Update at each iteration using the current averaged iterate**

  - Recursion: $\boxed{\theta_n = \theta_{n-1} - \gamma \big[f_n'(\bar{\theta}_{n-1}) + f_n''(\bar{\theta}_{n-1})(\theta_{n-1} - \bar{\theta}_{n-1}) \big]}$
  - No provable convergence rate but best practical behavior

Simulations - synthetic examples

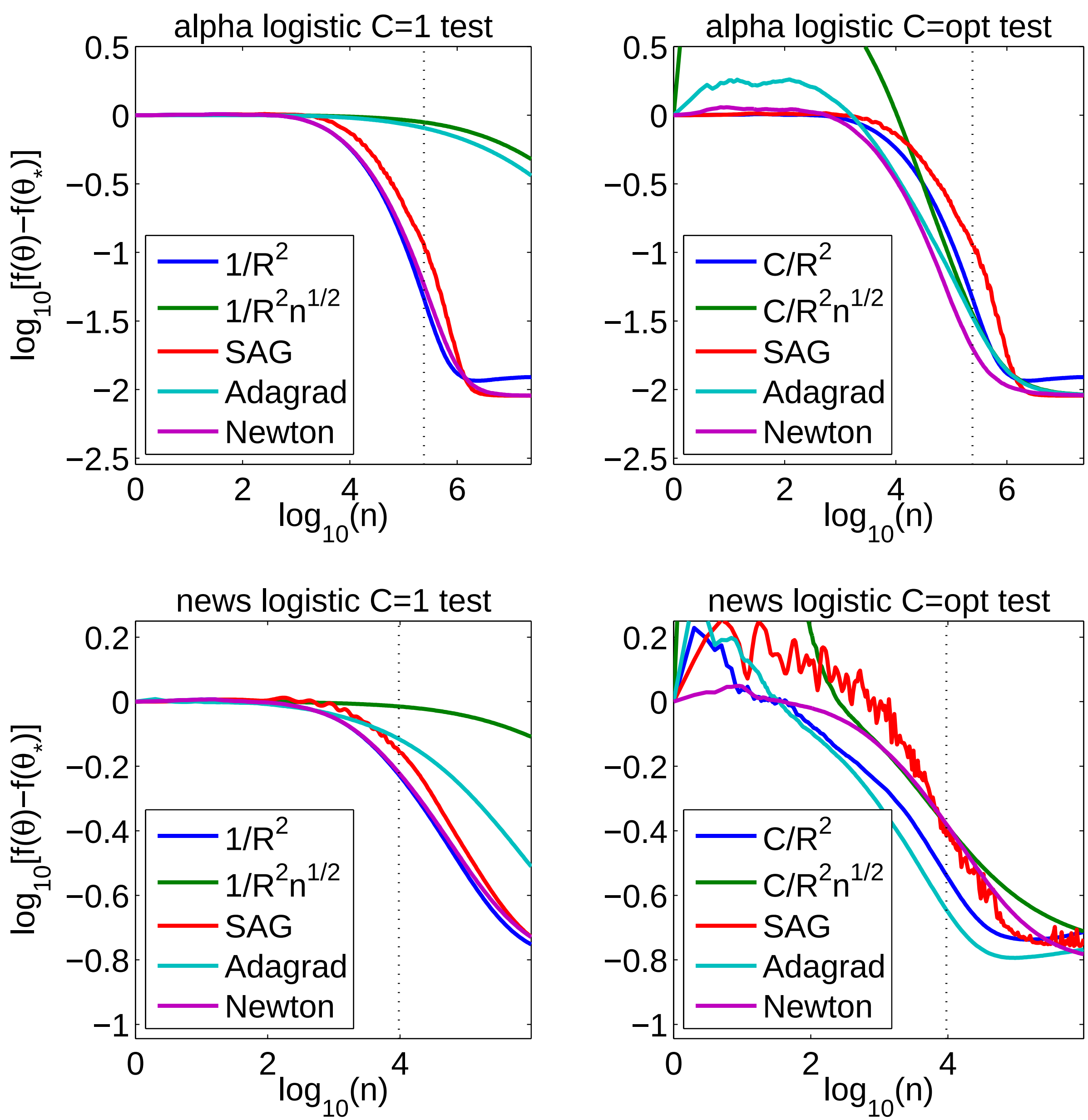
- Gaussian distributions - $p = 20$





Simulations - benchmarks

- *alpha* $(p = 500,\ n = 500\ 000)$, *news* $(p = 1\ 300\ 000,\ n = 20\ 000)$



Outline

- **Introduction: stochastic approximation algorithms**
  - Supervised machine learning and convex optimization
  - Stochastic gradient and averaging
  - Strongly convex vs. non-strongly convex
- **Fast convergence through smoothness and constant step-sizes**
  - Online Newton steps (Bach and Moulines, 2013)
  - $O(1/n)$ convergence rate for all convex functions
- **More than a single pass through the data**
  - Stochastic average gradient (Le Roux, Schmidt, and Bach, 2012)
  - Linear (exponential) convergence rate for strongly convex functions

Going beyond a single pass over the data

- **Stochastic approximation**
  - Assumes infinite data stream
  - Observations are used only once
  - Directly minimizes testing cost $\mathbb{E}_{(x,y)}\,\ell(y, \langle \theta, \Phi(x) \rangle)$

Going beyond a single pass over the data

- **Stochastic approximation**
  - Assumes infinite data stream
  - Observations are used only once
  - Directly minimizes testing cost $\mathbb{E}_{(x,y)}\,\ell(y, \langle \theta, \Phi(x) \rangle)$
- **Machine learning practice**
  - Finite data set $(x_1, y_1, \dots, x_n, y_n)$
  - Multiple passes
  - Minimizes training cost $\frac{1}{n}\sum_{i=1}^n \ell(y_i, \langle \theta, \Phi(x_i) \rangle)$
  - Need to regularize (e.g., by the ℓ_2-norm) to avoid overfitting
- **Goal**: minimize $g(\theta) = \dfrac{1}{n}\displaystyle\sum_{i=1}^n f_i(\theta)$

Stochastic vs. deterministic methods

- Minimizing $g(\theta) = \dfrac{1}{n}\sum_{i=1}^{n} f_i(\theta)$ with $f_i(\theta) = \ell\big(y_i, \theta^\top \Phi(x_i)\big) + \mu\Omega(\theta)$

- Batch gradient descent: $\theta_t = \theta_{t-1} - \gamma_t g'(\theta_{t-1}) = \theta_{t-1} - \dfrac{\gamma_t}{n}\sum_{i=1}^{n} f_i'(\theta_{t-1})$

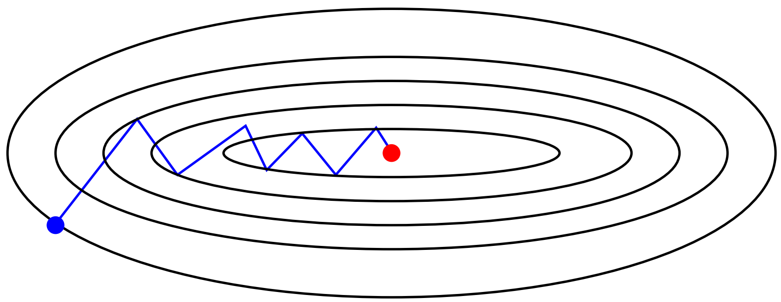
  – Linear (e.g., exponential) convergence rate (with strong convexity)

  – Iteration complexity is linear in n

Stochastic vs. deterministic methods

- Minimizing $g(\theta) = \dfrac{1}{n}\sum_{i=1}^{n} f_i(\theta)$ with $f_i(\theta) = \ell\big(y_i, \theta^\top \Phi(x_i)\big) + \mu\Omega(\theta)$

- Batch gradient descent: $\theta_t = \theta_{t-1} - \gamma_t g'(\theta_{t-1}) = \theta_{t-1} - \dfrac{\gamma_t}{n}\sum_{i=1}^{n} f_i'(\theta_{t-1})$



Stochastic vs. deterministic methods

- Minimizing $g(\theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\theta)$ with $f_i(\theta) = \ell\big(y_i, \theta^\top \Phi(x_i)\big) + \mu\Omega(\theta)$

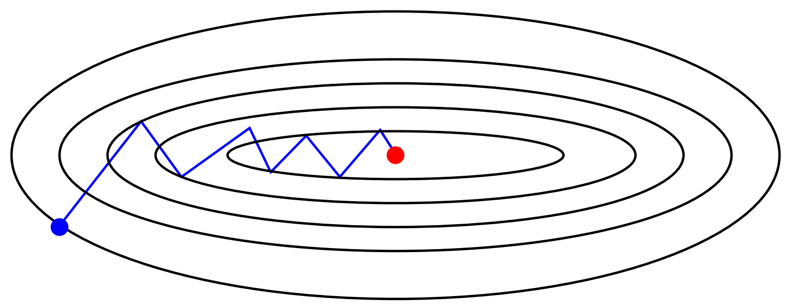
- Batch gradient descent: $\theta_t = \theta_{t-1} - \gamma_t g'(\theta_{t-1}) = \theta_{t-1} - \frac{\gamma_t}{n}\sum_{i=1}^{n} f_i'(\theta_{t-1})$
  - Linear (e.g., exponential) convergence rate (with strong convexity)
  - Iteration complexity is linear in n

- Stochastic gradient descent: $\theta_t = \theta_{t-1} - \gamma_t f'_{i(t)}(\theta_{t-1})$
  - Sampling with replacement: $i(t)$ random element of $\{1, \dots, n\}$
  - Convergence rate in $O(1/t)$
  - Iteration complexity is independent of n

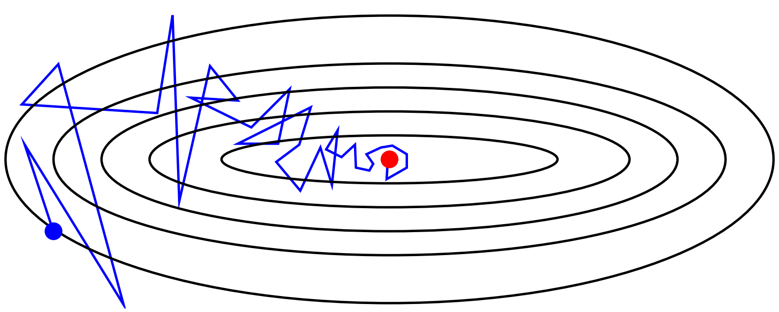
Stochastic vs. deterministic methods

- Minimizing $g(\theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\theta)$ with $f_i(\theta) = \ell\big(y_i, \theta^\top \Phi(x_i)\big) + \mu\Omega(\theta)$

- Batch gradient descent: $\theta_t = \theta_{t-1} - \gamma_t g'(\theta_{t-1}) = \theta_{t-1} - \frac{\gamma_t}{n}\sum_{i=1}^{n} f_i'(\theta_{t-1})$



- Stochastic gradient descent: $\theta_t = \theta_{t-1} - \gamma_t f'_{i(t)}(\theta_{t-1})$



Stochastic average gradient
(Le Roux, Schmidt, and Bach, 2012)

- **Stochastic average gradient** (SAG) iteration
  - Keep in memory the gradients of all functions f_i, $i = 1, \dots, n$
  - Random selection $i(t) \in \{1, \dots, n\}$ with replacement
  - Iteration: $\theta_t = \theta_{t-1} - \frac{\gamma_t}{n} \sum_{i=1}^{n} y_i^t$ with $y_i^t = \begin{cases} f_i'(\theta_{t-1}) & \text{if } i = i(t) \\ y_i^{t-1} & \text{otherwise} \end{cases}$

Stochastic average gradient (Le Roux, Schmidt, and Bach, 2012)

- **Stochastic average gradient** (SAG) iteration
  - Keep in memory the gradients of all functions f_i, $i = 1, \dots, n$
  - Random selection $i(t) \in \{1, \dots, n\}$ with replacement
  - Iteration: $\theta_t = \theta_{t-1} - \frac{\gamma_t}{n} \sum_{i=1}^{n} y_i^t$ with $y_i^t = \begin{cases} f_i'(\theta_{t-1}) & \text{if } i = i(t) \\ y_i^{t-1} & \text{otherwise} \end{cases}$

- Stochastic version of incremental average gradient (Blatt et al., 2008)

- Extra memory requirement
  - Supervised machine learning
  - If $f_i(\theta) = \ell_i(y_i, \Phi(x_i)^\top \theta)$, then $f_i'(\theta) = \ell_i'(y_i, \Phi(x_i)^\top \theta)\, \Phi(x_i)$
  - Only need to store n real numbers

Stochastic average gradient - Convergence analysis

- **Assumptions**
  - Each f_i is L-smooth, $i = 1, \dots, n$
  - $g = \frac{1}{n} \sum_{i=1}^{n} f_i$ is μ-strongly convex (with potentially $\mu = 0$)
  - constant step size $\gamma_t = 1/(16L)$
  - initialization with one pass of averaged SGD

Stochastic average gradient - Convergence analysis

- **Assumptions**
  - Each f_i is L-smooth, $i = 1, \dots, n$
  - $g = \frac{1}{n}\sum_{i=1}^n f_i$ is μ-strongly convex (with potentially $\mu = 0$)
  - constant step size $\gamma_t = 1/(16L)$
  - initialization with one pass of averaged SGD

- **Strongly convex case** (Le Roux et al., 2012, 2013)

$$\mathbb{E}\big[g(\theta_t) - g(\theta_*)\big] \leqslant \Big(\frac{8\sigma^2}{n} + \frac{4L\|\theta_0 - \theta_*\|^2}{n}\Big) \exp\Big(-t \min\Big\{\frac{1}{8n}, \frac{\mu}{16L}\Big\}\Big)$$

  - Linear (exponential) convergence rate with $O(1)$ iteration cost
  - After one pass, reduction of cost by $\exp\Big(-\min\Big\{\frac{1}{8}, \frac{n\mu}{16L}\Big\}\Big)$

Stochastic average gradient - Convergence analysis

- **Assumptions**
  - Each f_i is L-smooth, $i = 1, \dots, n$
  - $g = \frac{1}{n}\sum_{i=1}^n f_i$ is μ-strongly convex (with potentially $\mu = 0$)
  - constant step size $\gamma_t = 1/(16L)$
  - initialization with one pass of averaged SGD

- **Non-strongly convex case** (Le Roux et al., 2013)

$$\mathbb{E}\big[g(\theta_t) - g(\theta_*)\big] \leqslant 48 \frac{\sigma^2 + L\|\theta_0 - \theta_*\|^2}{\sqrt{n}} \frac{n}{t}$$

  - Improvement over regular batch and stochastic gradient
  - Adaptivity to potentially hidden strong convexity

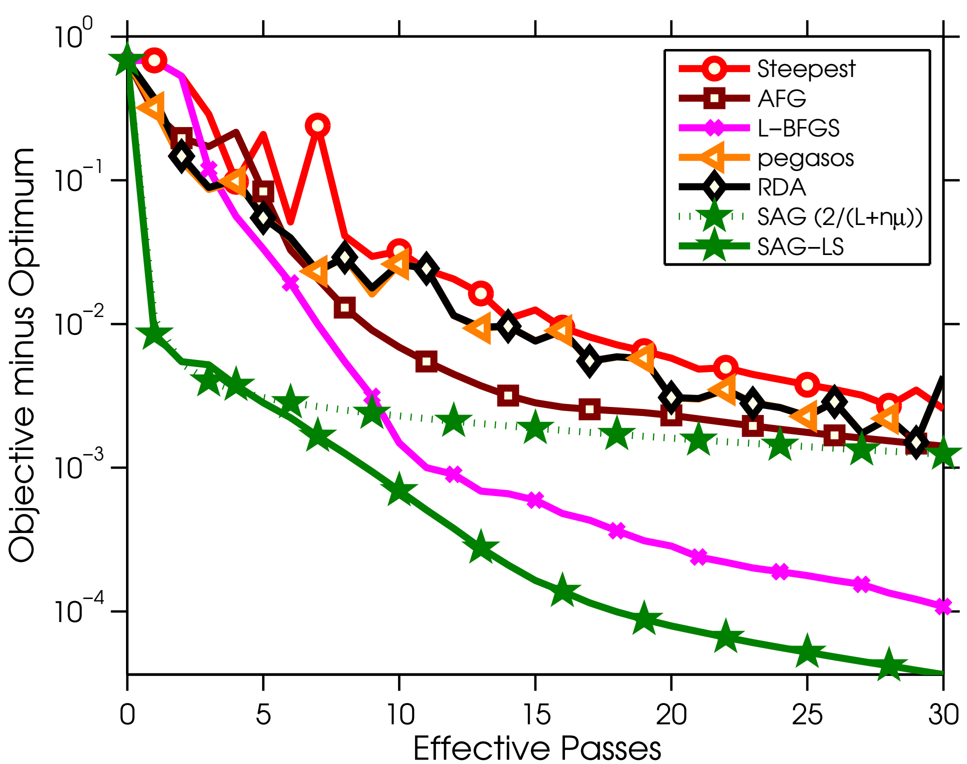
Stochastic average gradient
Implementation details and extensions

- The algorithm can use sparsity in the features to reduce the storage and iteration cost
- Grouping functions together can further reduce the memory requirement
- We have obtained good performance when L is not known with a heuristic line-search
- Algorithm allows non-uniform sampling
- Possibility of making proximal, coordinate-wise, and Newton-like variants

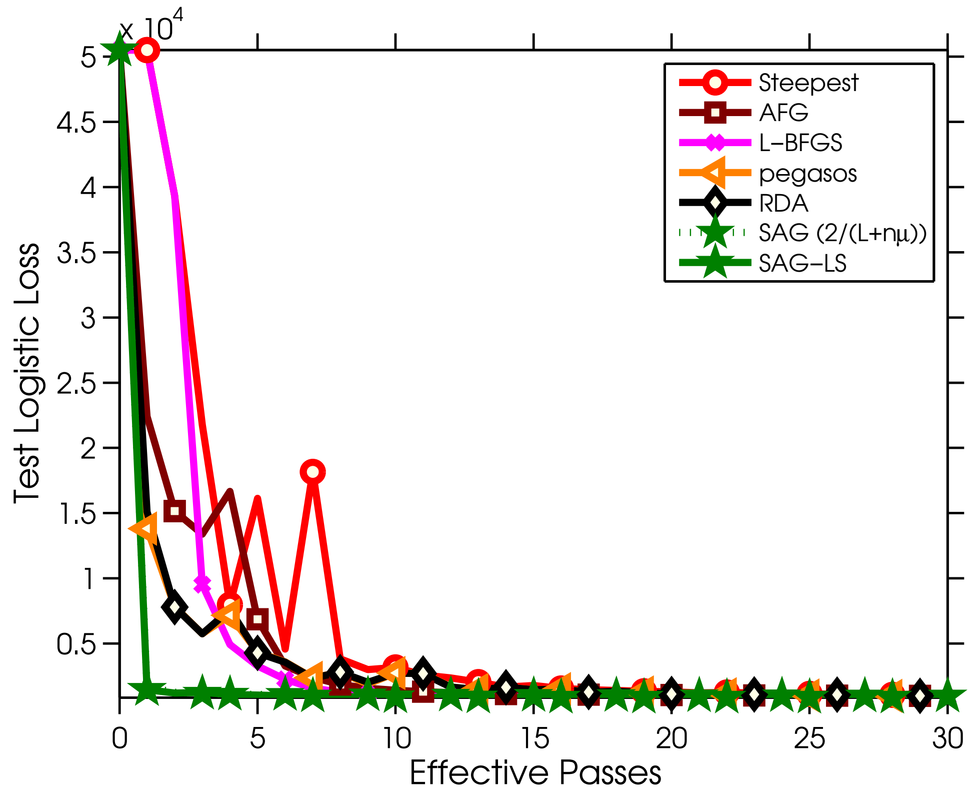
Stochastic average gradient

Simulation experiments

- `protein` dataset ($n = 145751$, $p = 74$)
- Dataset split in two (training/testing)



Training cost

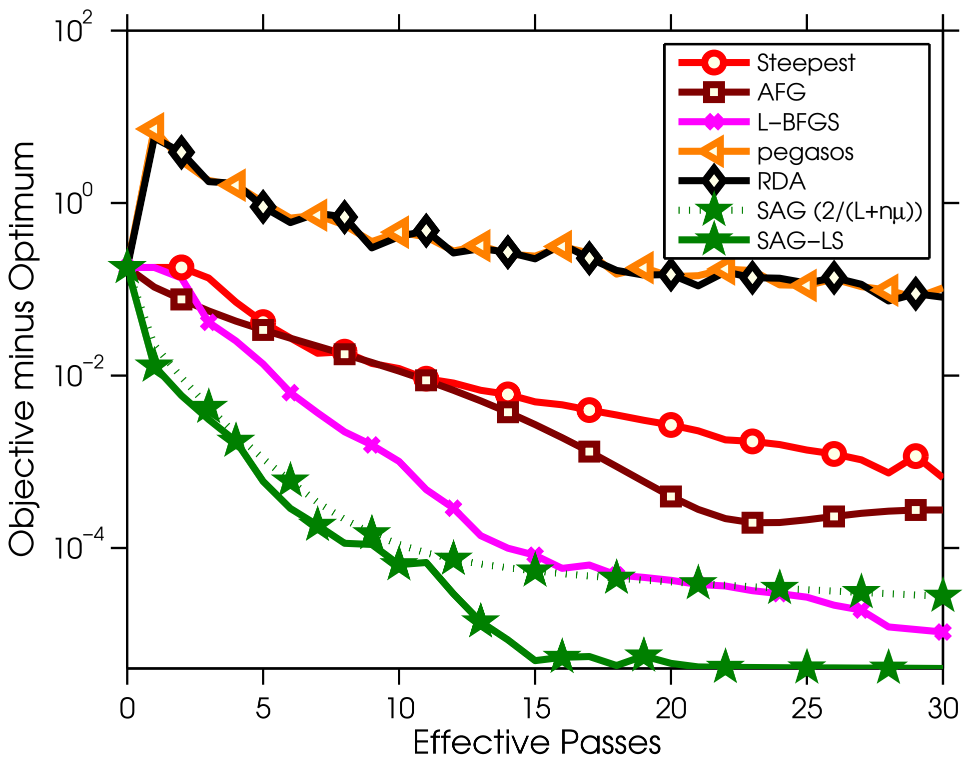


Testing cost

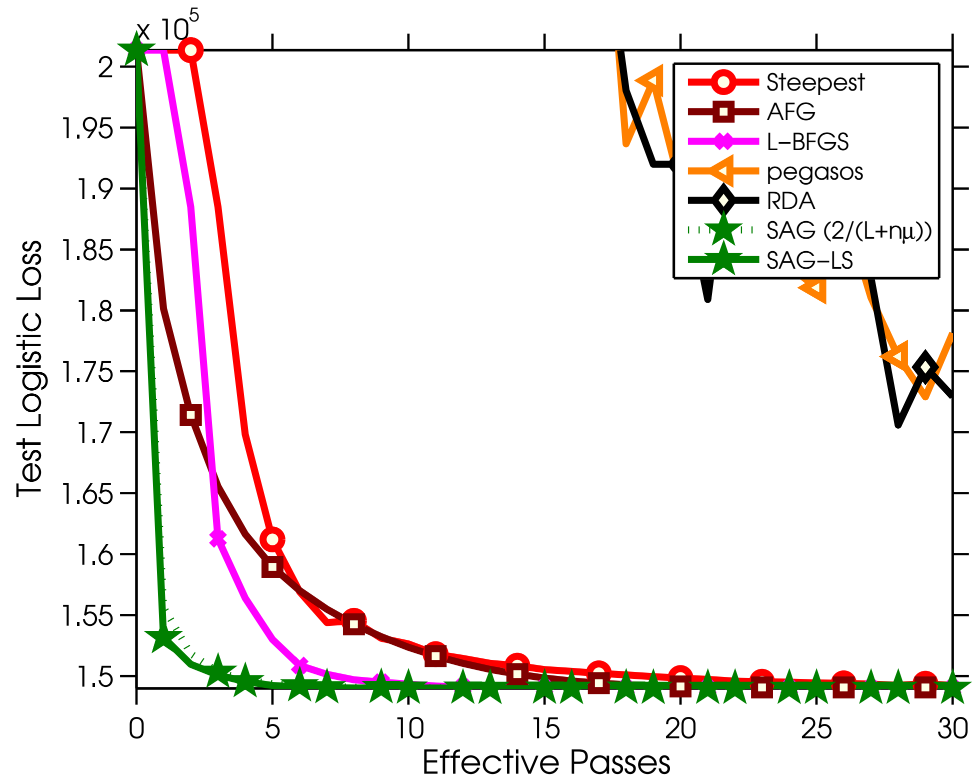
Stochastic average gradient

Simulation experiments

- `cover type` dataset ($n = 581012$, $p = 54$)
- Dataset split in two (training/testing)



Training cost



Testing cost

Conclusions

- **Constant-step-size averaged stochastic gradient descent**
  - Reaches convergence rate $O(1/n)$ in all regimes
  - Improves on the $O(1/\sqrt{n})$ lower-bound of non-smooth problems
  - Efficient online Newton step for non-quadratic problems
- **Going beyond a single pass through the data**
  - Keep memory of all gradients for finite training sets
  - Randomization leads to easier analysis and faster rates
- **Extensions**
  - Non-differentiable terms, kernels, line-search, parallelization, etc.
- **Role of non-smoothness in machine learning**

References

A. Agarwal, P. L. Bartlett, P. Ravikumar, and M. J. Wainwright. Information-theoretic lower bounds on the oracle complexity of stochastic convex optimization. *Information Theory, IEEE Transactions on*, 58(5):3235–3249, 2012.

F. Bach. Adaptivity of averaged stochastic gradient descent to local strong convexity for logistic regression. Technical Report 00804431, HAL, 2013.

F. Bach and E. Moulines. Non-strongly-convex smooth stochastic approximation with convergence rate $o(1/n)$. Technical Report 00831977, HAL, 2013.

D. Blatt, A. O. Hero, and H. Gauchman. A convergent incremental gradient method with a constant step size. *SIAM Journal on Optimization*, 18(1):29–51, 2008.

L. Bottou and O. Bousquet. The tradeoffs of large scale learning. In *Adv. NIPS*, 2008.

L. Bottou and Y. Le Cun. On-line learning for very large data sets. *Applied Stochastic Models in Business and Industry*, 21(2):137–151, 2005.

J. Duchi and Y. Singer. Efficient online and batch learning using forward backward splitting. *Journal of Machine Learning Research*, 10:2899–2934, 2009. ISSN 1532-4435.

E. Hazan, A. Agarwal, and S. Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2):169–192, 2007.

N. Le Roux, M. Schmidt, and F. Bach. A stochastic gradient method with an exponential convergence rate for strongly-convex optimization with finite training sets. In *Adv. NIPS*, 2012.

N. Le Roux, M. Schmidt, and F. Bach. A stochastic gradient method with an exponential convergence rate for strongly-convex optimization with finite training sets. Technical Report 00674995, HAL, 2013.

O. Macchi. *Adaptive processing: The least mean squares approach with applications in transmission*. Wiley West Sussex, 1995.

A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609, 2009.

A. S. Nemirovsky and D. B. Yudin. *Problem complexity and method efficiency in optimization.* Wiley & Sons, 1983.

Y. Nesterov and A. Nemirovski. *Interior-point polynomial algorithms in convex programming*. SIAM studies in Applied Mathematics, 1994.

Y. Nesterov and J. P. Vial. Confidence level solutions for stochastic programming. *Automatica*, 44(6): 1559–1568, 2008. ISSN 0005-1098.

B. T. Polyak and A. B. Juditsky. Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, 30(4):838–855, 1992.

H. Robbins and S. Monro. A stochastic approximation method. *Ann. Math. Statistics*, 22:400–407, 1951. ISSN 0003-4851.

D. Ruppert. Efficient estimations from a slowly convergent Robbins-Monro process. Technical Report 781, Cornell University Operations Research and Industrial Engineering, 1988.

S. Shalev-Shwartz and N. Srebro. SVM optimization: inverse dependence on training set size. In *Proc. ICML*, 2008.

S. Shalev-Shwartz, Y. Singer, and N. Srebro. Pegasos: Primal estimated sub-gradient solver for svm.

In *Proc. ICML*, 2007.

S. Shalev-Shwartz, O. Shamir, N. Srebro, and K. Sridharan. Stochastic convex optimization. In *proc. COLT*, 2009.

A. B. Tsybakov. Optimal rates of aggregation. In *Proc. COLT*, 2003.

A. W. Van der Vaart. *Asymptotic statistics*, volume 3. Cambridge Univ. press, 2000.

L. Xiao. Dual averaging methods for regularized stochastic learning and online optimization. *Journal of Machine Learning Research*, 9:2543–2596, 2010. ISSN 1532-4435.