Stochastic gradient methods for machine learning

Francis Bach

INRIA - Ecole Normale Supérieure, Paris, France

Joint work with Eric Moulines, Nicolas Le Roux and Mark Schmidt - September 2012

Context

Machine learning for "big data"

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

- **Examples**: computer vision, bioinformatics, signal 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)
  - k : number of tasks (dimension of outputs)
  - n : number of observations
- **Examples**: computer vision, bioinformatics, signal processing
- **Ideal running-time complexity**: $O(pn + kn)$
- **Going back to simple methods**
  - Stochastic gradient methods (Robbins and Monro, 1951)
  - Mixing statistics and optimization
  - It is possible to improve on the sublinear convergence rate?

Outline

- **Introduction**
  - Supervised machine learning and convex optimization
  - Beyond the separation of statistics and optimization

- **Stochastic approximation algorithms** (Bach and Moulines, 2011)
  - Stochastic gradient and averaging
  - Strongly convex vs. non-strongly convex

- **Going beyond stochastic gradient** (Le Roux, Schmidt, and Bach, 2012)
  - More than a single pass through the data
  - Linear (exponential) convergence rate

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 $\theta^\top \Phi(x)$ of features $\Phi(x) \in \mathcal{F} = \mathbb{R}^p$
- **(regularized) empirical risk minimization**: find $\hat{\theta}$ solution of

$$\min_{\theta \in \mathcal{F}} \quad \frac{1}{n} \sum_{i=1}^{n} \ell\big(y_i, \theta^\top \Phi(x_i)\big) \quad + \quad \mu \Omega(\theta)$$

convex data fitting term + regularizer

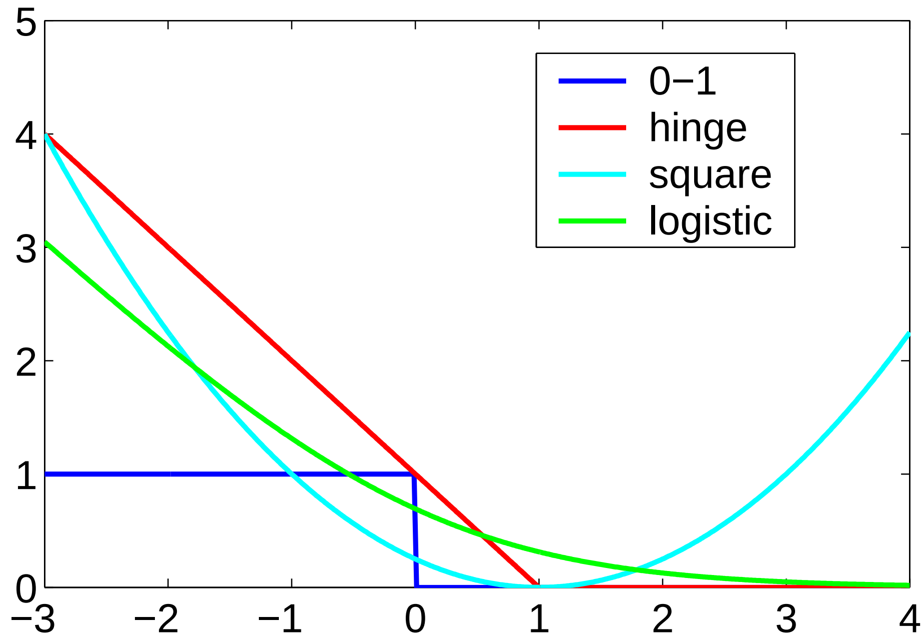
Usual losses

- **Regression**: $y \in \mathbb{R}$, prediction $\hat{y} = \theta^\top \Phi(x)$
  - quadratic loss $\frac{1}{2}(y - \hat{y})^2 = \frac{1}{2}(y - \theta^\top \Phi(x))^2$

Usual losses

- **Regression**: $y \in \mathbb{R}$, prediction $\hat{y} = \theta^\top \Phi(x)$
  – quadratic loss $\frac{1}{2}(y - \hat{y})^2 = \frac{1}{2}(y - \theta^\top \Phi(x))^2$

- **Classification** : $y \in \{-1, 1\}$, prediction $\hat{y} = \text{sign}(\theta^\top \Phi(x))$
  – loss of the form $\ell(y\,\theta^\top \Phi(x))$
  – "True" 0-1 loss: $\ell(y\,\theta^\top \Phi(x)) = 1_{y\,\theta^\top \Phi(x) < 0}$
  – Usual convex losses:



Usual regularizers

- **Main goal**: avoid overfitting

- **(squared) Euclidean norm**: $\|\theta\|_2^2 = \sum_{j=1}^p |\theta_j|^2$
  - Numerically well-behaved
  - Representer theorem and kernel methods : $\theta = \sum_{i=1}^n \alpha_i \Phi(x_i)$
  - See, e.g., Schölkopf and Smola (2001); Shawe-Taylor and Cristianini (2004)

- **Sparsity-inducing norms**
  - Main example: ℓ_1-norm $\|\theta\|_1 = \sum_{j=1}^p |\theta_j|$
  - Perform model selection as well as regularization
  - Non-smooth optimization and structured sparsity
  - See, e.g., Bach, Jenatton, Mairal, and Obozinski (2011, 2012)

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 $\theta^\top \Phi(x)$ of features $\Phi(x) \in \mathcal{F} = \mathbb{R}^p$
- **(regularized) empirical risk minimization**: find $\hat{\theta}$ solution of

$$\min_{\theta \in \mathcal{F}} \quad \frac{1}{n} \sum_{i=1}^{n} \ell\big(y_i, \theta^\top \Phi(x_i)\big) \quad + \quad \mu \Omega(\theta)$$

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 $\theta^\top \Phi(x)$ of features $\Phi(x) \in \mathcal{F} = \mathbb{R}^p$
- **(regularized) empirical risk minimization**: find $\hat{\theta}$ solution of

$$\min_{\theta \in \mathcal{F}} \quad \frac{1}{n} \sum_{i=1}^{n} \ell\big(y_i, \theta^\top \Phi(x_i)\big) \quad + \quad \mu \Omega(\theta)$$

convex data fitting term + regularizer

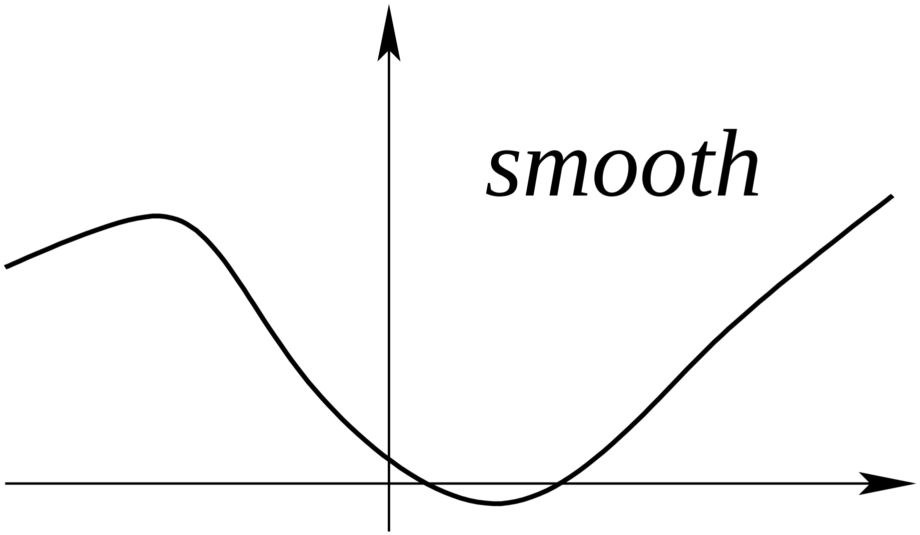
- Empirical risk: $\hat{f}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, \theta^\top \Phi(x_i))$ training cost
- Expected risk: $f(\theta) = \mathbb{E}_{(x,y)} \ell(y, \theta^\top \Phi(x))$ testing cost
- **Two fundamental questions**: (1) computing $\hat{\theta}$ and (2) analyzing $\hat{\theta}$

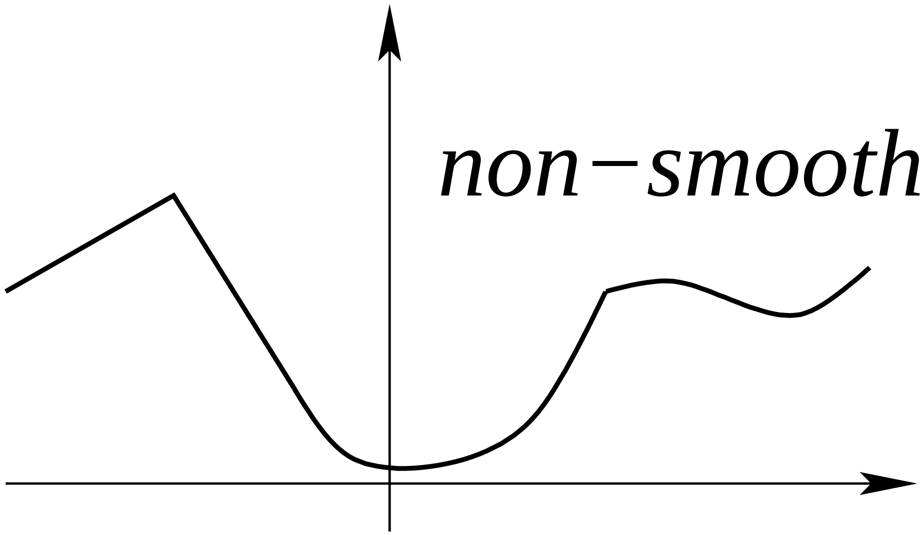
Smoothness and strong convexity

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

$$\forall \theta_1, \theta_2 \in \mathbb{R}^p, \ \|g'(\theta_1) - g'(\theta_2)\| \leqslant L\|\theta_1 - \theta_2\|$$

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





Smoothness and strong convexity

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

$$\forall \theta_1, \theta_2 \in \mathbb{R}^p, \ \|g'(\theta_1) - g'(\theta_2)\| \leqslant L\|\theta_1 - \theta_2\|$$

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

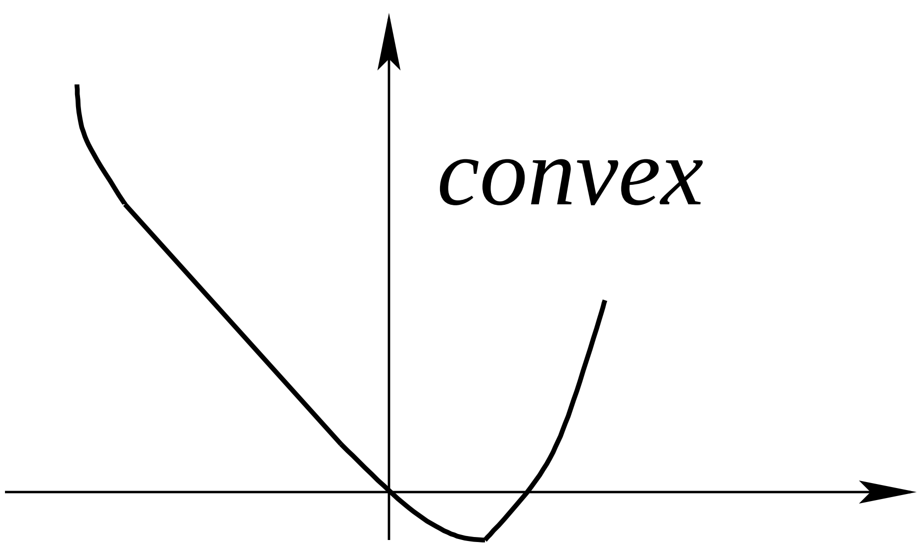
- **Machine learning**
  - with $g(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, \theta^\top \Phi(x_i))$
  - Hessian $\approx$ covariance matrix $\frac{1}{n}\sum_{i=1}^{n} \Phi(x_i)\Phi(x_i)^\top$
  - Bounded data

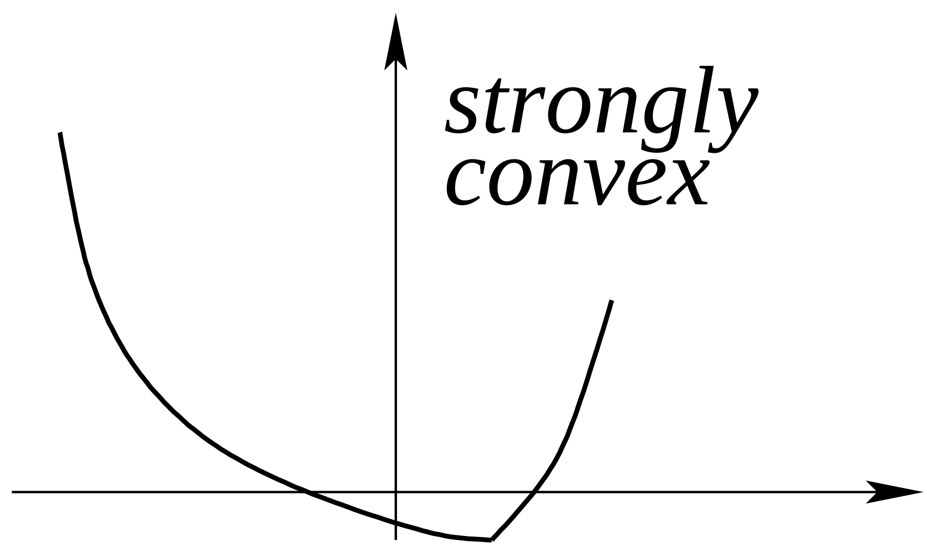
Smoothness and strong convexity

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

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

- Equivalent definition: $\theta \mapsto g(\theta) - \frac{\mu}{2}\|\theta\|^2$ is convex
- If g is twice differentiable: $\forall \theta \in \mathbb{R}^p, \ g''(\theta) \succcurlyeq \mu \cdot Id$





Smoothness and strong convexity

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

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

- Equivalent definition: $\theta \mapsto g(\theta) - \frac{\mu}{2}\|\theta\|^2$ is convex

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

- **Machine learning**
  - with $g(\theta) = \frac{1}{n}\sum_{i=1}^n \ell(y_i, \theta^\top \Phi(x_i))$
  - Hessian $\approx$ covariance matrix $\frac{1}{n}\sum_{i=1}^n \Phi(x_i)\Phi(x_i)^\top$
  - Data with invertible covariance matrix (low correlation/dimension)
  - ... or with added regularization by $\frac{\mu}{2}\|\theta\|^2$

Statistical analysis of empirical risk minimization

- **Fundamental decomposition**:

generalisation error $=$ estimation error $+$ approximation error

- **Approximation error**
  - Bias introduced by choice of features and use of regularization

- **Estimation error**
  - Variance introduced by using a finite sample
  - See Boucheron et al. (2005); Sridharan et al. (2008); Boucheron and Massart (2011)
  - $O(1/n)$ for strongly convex functions, $O(1/\sqrt{n})$ otherwise

Iterative methods for minimizing smooth functions

- **Assumption**: g convex and smooth on $\mathcal{F}$ (Hilbert space or $\mathbb{R}^p$)

- **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 $\mathcal{F}$ (Hilbert space or $\mathbb{R}^p$)

- **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 estimation error
  2. In machine learning, cost functions are averages

$\Rightarrow$ **Stochastic approximation**

Outline

- **Introduction**
  - Supervised machine learning and convex optimization
  - Beyond the separation of statistics and optimization

- **Stochastic approximation algorithms** (Bach and Moulines, 2011)
  - Stochastic gradient and averaging
  - Strongly convex vs. non-strongly convex

- **Going beyond stochastic gradient** (Le Roux, Schmidt, and Bach, 2012)
  - More than a single pass through the data
  - Linear (exponential) convergence rate

Stochastic approximation

- **Goal**: Minimizing a function f defined on a Hilbert space $\mathcal{H}$
  - given only unbiased estimates $f'_n(\theta_n)$ of its gradients $f'(\theta_n)$ at certain points $\theta_n \in \mathcal{H}$

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

Stochastic approximation

- **Goal**: Minimizing a function f defined on a Hilbert space $\mathcal{H}$
  - given only unbiased estimates $f'_n(\theta_n)$ of its gradients $f'(\theta_n)$ at certain points $\theta_n \in \mathcal{H}$

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

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

Convex smooth stochastic approximation

- **Key properties of f and/or f_n**
  - Smoothness: f_n L-smooth
  - Strong convexity: f μ-strongly convex

Convex smooth stochastic approximation

- **Key properties of f and/or f_n**
  - Smoothness: f_n L-smooth
  - Strong convexity: f μ-strongly convex

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

Convex smooth stochastic approximation

- **Key properties of f and/or f_n**
  - Smoothness: f_n L-smooth
  - Strong convexity: f μ-strongly convex

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

- **Desirable practical behavior**
  - Applicable (at least) to least-squares and logistic regression
  - Robustness to (potentially unknown) constants (L, μ)
  - Adaptivity to difficulty of the problem (e.g., strong convexity)

Convex stochastic approximation
Related work

- **Machine learning/optimization**
  - Known minimax rates of convergence (Nemirovski and Yudin, 1983; Agarwal et al., 2010)
    - Strongly convex: $O(n^{-1})$
    - Non-strongly convex: $O(n^{-1/2})$
  - Achieved with and/or without averaging (up to log terms)
  - Non-asymptotic analysis (high-probability bounds)
  - Online setting and regret bounds
  - 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
Related work

- **Stochastic approximation**
  - Asymptotic analysis
  - Non convex case with strong convexity around the optimum
  - $\gamma_n = Cn^{-\alpha}$ with $\alpha = 1$ is not robust to the choice of C
  - $\alpha \in (1/2, 1)$ is robust with averaging
  - Broadie et al. (2009); Kushner and Yin (2003); Kul′chitskiĭ and Mozgovoĭ (1991); Fabian (1968)
  - Polyak and Juditsky (1992); Ruppert (1988)

Problem set-up - General assumptions

- **Unbiased gradient estimates**:
  - $f_n(\theta)$ is of the form $h(z_n, \theta)$, where z_n is an i.i.d. sequence
  - e.g., $f_n(\theta) = h(z_n, \theta) = \ell(y_n, \theta^\top \Phi(x_n))$ with $z_n = (x_n, y_n)$
  - NB: can be generalized
- **Variance of estimates**: There exists $\sigma^2 \geqslant 0$ such that for all $n \geqslant 1$, $\mathbb{E}(\|f_n'(\theta^*) - f'(\theta^*)\|^2) \leqslant \sigma^2$, where θ^* is a global minimizer of f
- Specificity of machine learning
  - Full function $\theta \mapsto f_n(\theta) = h(\theta, z_n)$ is observed
  - Beyond i.i.d. assumptions

Problem set-up - Smoothness/convexity assumptions

- **Smoothness of f_n:** For each $n \geqslant 1$, the function f_n is a.s. convex, differentiable with L-Lipschitz-continuous gradient f_n':

$$\forall n \geqslant 1,\ \forall \theta_1, \theta_2 \in \mathcal{H}, \quad \|f_n'(\theta_1) - f_n'(\theta_2)\| \leqslant L\|\theta_1 - \theta_2\|\,, \quad \text{w.p.1}$$

Problem set-up - Smoothness/convexity assumptions

- **Smoothness of f_n**: For each $n \geqslant 1$, the function f_n is a.s. convex, differentiable with L-Lipschitz-continuous gradient f_n':

$$\forall n \geqslant 1,\ \forall \theta_1, \theta_2 \in \mathcal{H},\quad \|f_n'(\theta_1) - f_n'(\theta_2)\| \leqslant L\|\theta_1 - \theta_2\|\,, \quad \text{w.p.1}$$

- **Strong convexity of f**: The function f is strongly convex with respect to the norm $\|\cdot\|$, with convexity constant $\mu > 0$:

$$\forall \theta_1, \theta_2 \in \mathcal{H},\ f(\theta_1) \geqslant f(\theta_2) + \langle f'(\theta_2), \theta_1 - \theta_2 \rangle + \frac{\mu}{2}\|\theta_1 - \theta_2\|^2$$

Summary of new results (Bach and Moulines, 2011)

- Stochastic gradient descent with learning rate $\gamma_n = Cn^{-\alpha}$
- **Strongly convex smooth objective functions**
  - Old: $O(n^{-1})$ rate achieved without averaging for $\alpha = 1$
  - New: $O(n^{-1})$ rate achieved with averaging for $\alpha \in [1/2, 1]$
  - Non-asymptotic analysis with explicit constants
  - Forgetting of initial conditions
  - Robustness to the choice of C

Summary of new results (Bach and Moulines, 2011)

- Stochastic gradient descent with learning rate $\gamma_n = Cn^{-\alpha}$

- **Strongly convex smooth objective functions**
  - Old: $O(n^{-1})$ rate achieved without averaging for $\alpha = 1$
  - New: $O(n^{-1})$ rate achieved with averaging for $\alpha \in [1/2, 1]$
  - Non-asymptotic analysis with explicit constants
  - Forgetting of initial conditions
  - Robustness to the choice of C

- **Proof technique**
  - Derive deterministic recursion for $\delta_n = \mathbb{E}\|\theta_n - \theta^*\|^2$

$$\delta_n \leqslant (1 - 2\mu\gamma_n + 2L^2\gamma_n^2)\delta_{n-1} + 2\sigma^2\gamma_n^2$$

  - Mimic SA proof techniques in a non-asymptotic way

Summary of new results (Bach and Moulines, 2011)

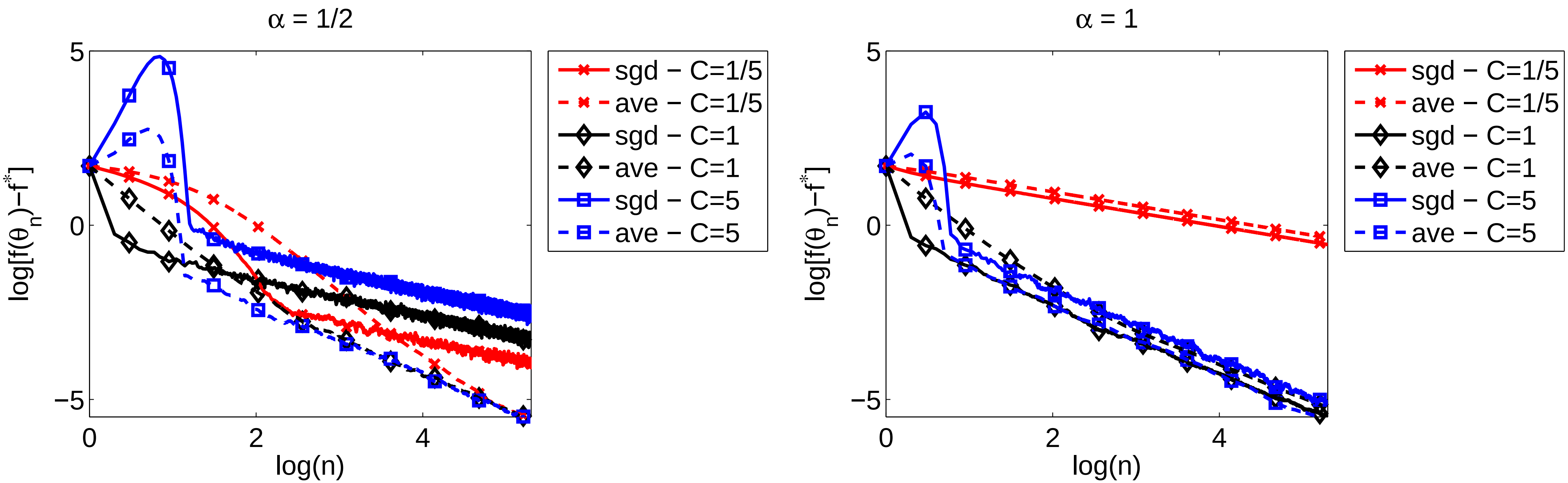
- Stochastic gradient descent with learning rate $\gamma_n = Cn^{-\alpha}$

- **Strongly convex smooth objective functions**
  - Old: $O(n^{-1})$ rate achieved without averaging for $\alpha = 1$
  - New: $O(n^{-1})$ rate achieved with averaging for $\alpha \in [1/2, 1]$
  - Non-asymptotic analysis with explicit constants
  - Forgetting of initial conditions
  - Robustness to the choice of C

- **Convergence rates** for $\mathbb{E}\|\theta_n - \theta^*\|^2$ and $\mathbb{E}\|\bar{\theta}_n - \theta^*\|^2$
  - no averaging: $O\Big(\frac{\sigma^2\gamma_n}{\mu}\Big) + O(e^{-\mu n \gamma_n})\|\theta_0 - \theta^*\|^2$
  - averaging: $\frac{\operatorname{tr} H(\theta^*)^{-1}}{n} + O(n^{-2\alpha} + n^{-2+\alpha}) + O\Big(\frac{\|\theta_0 - \theta^*\|^2}{n^2}\Big)$

Robustness to wrong constants for $\gamma_n = Cn^{-\alpha}$

- $f(\theta) = \frac{1}{2}|\theta|^2$ with i.i.d. Gaussian noise ($p = 1$)
- Left: $\alpha = 1/2$
- Right: $\alpha = 1$



- See also `http://leon.bottou.org/projects/sgd`

Summary of new results (Bach and Moulines, 2011)

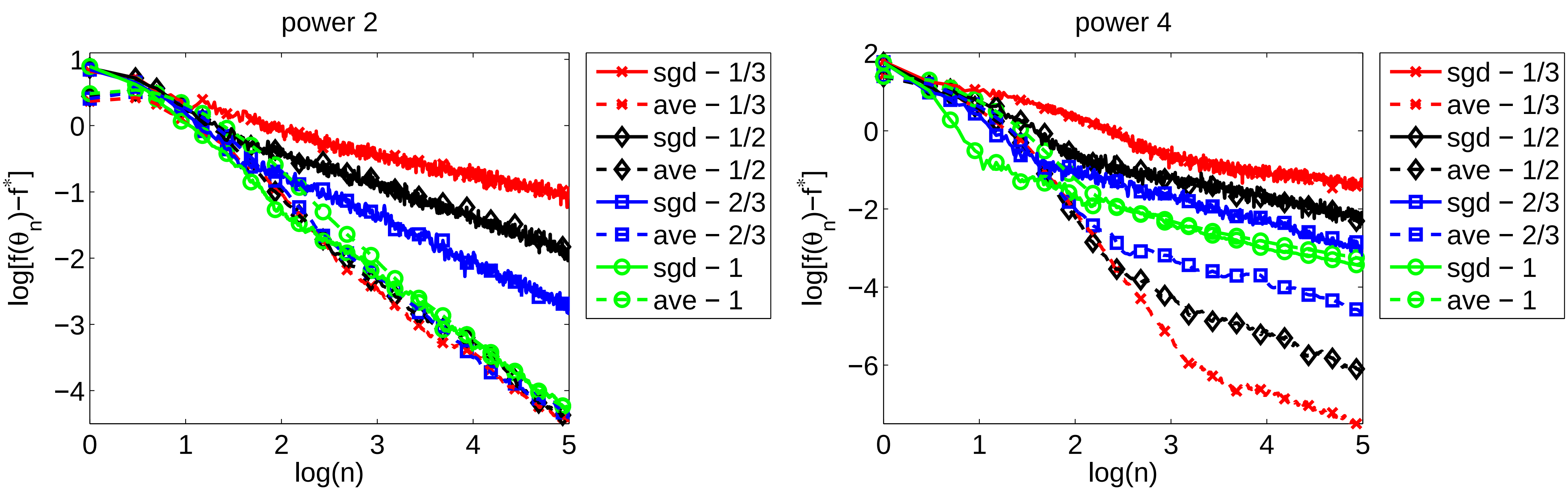
- Stochastic gradient descent with learning rate $\gamma_n = Cn^{-\alpha}$
- **Strongly convex smooth objective functions**
  - Old: $O(n^{-1})$ rate achieved without averaging for $\alpha = 1$
  - New: $O(n^{-1})$ rate achieved with averaging for $\alpha \in [1/2, 1]$
  - Non-asymptotic analysis with explicit constants

Summary of new results (Bach and Moulines, 2011)

- Stochastic gradient descent with learning rate $\gamma_n = Cn^{-\alpha}$
- **Strongly convex smooth objective functions**
  - Old: $O(n^{-1})$ rate achieved without averaging for $\alpha = 1$
  - New: $O(n^{-1})$ rate achieved with averaging for $\alpha \in [1/2, 1]$
  - Non-asymptotic analysis with explicit constants
- **Non-strongly convex smooth objective functions**
  - Old: $O(n^{-1/2})$ rate achieved with averaging for $\alpha = 1/2$
  - New: $O(\max\{n^{1/2-3\alpha/2}, n^{-\alpha/2}, n^{\alpha-1}\})$ rate achieved without averaging for $\alpha \in [1/3, 1]$
- **Take-home message**
  - Use $\alpha = 1/2$ with averaging to be adaptive to strong convexity

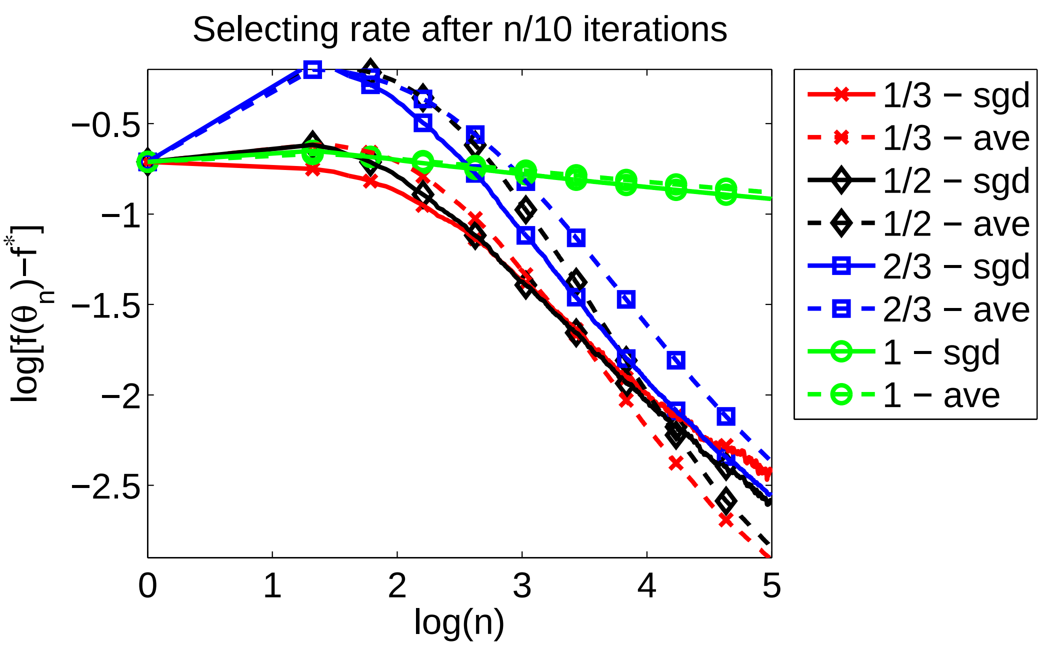
Robustness to lack of strong convexity

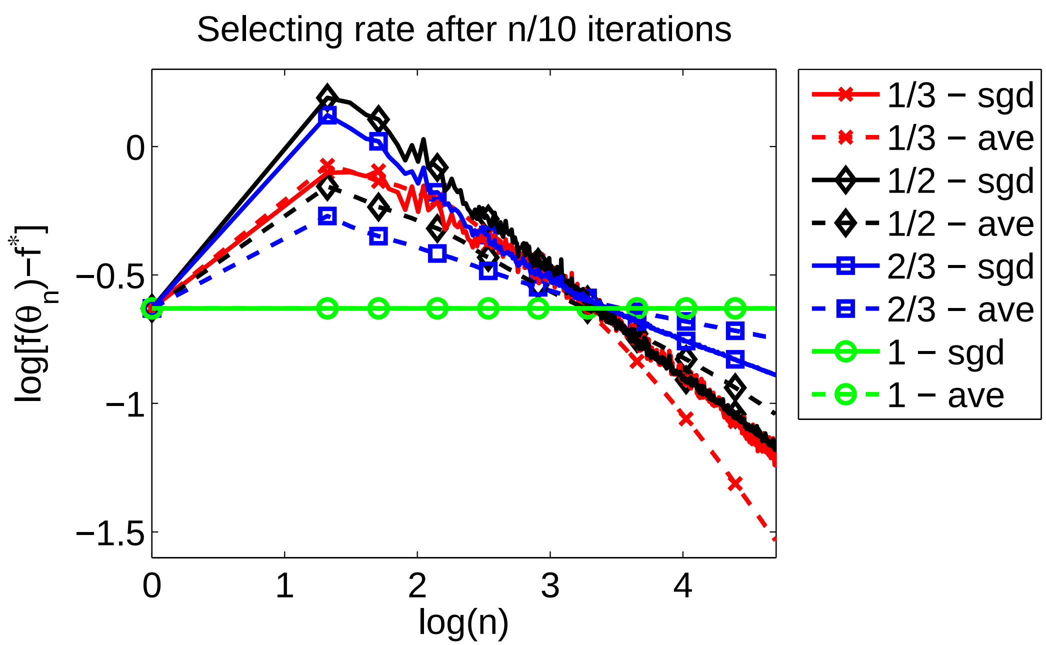
- Left: $f(\theta) = |\theta|^2$ between -1 and 1
- Right: $f(\theta) = |\theta|^4$ between -1 and 1
- affine outside of $[-1, 1]$, continuously differentiable.



Comparison on non strongly convex logistic regression problems

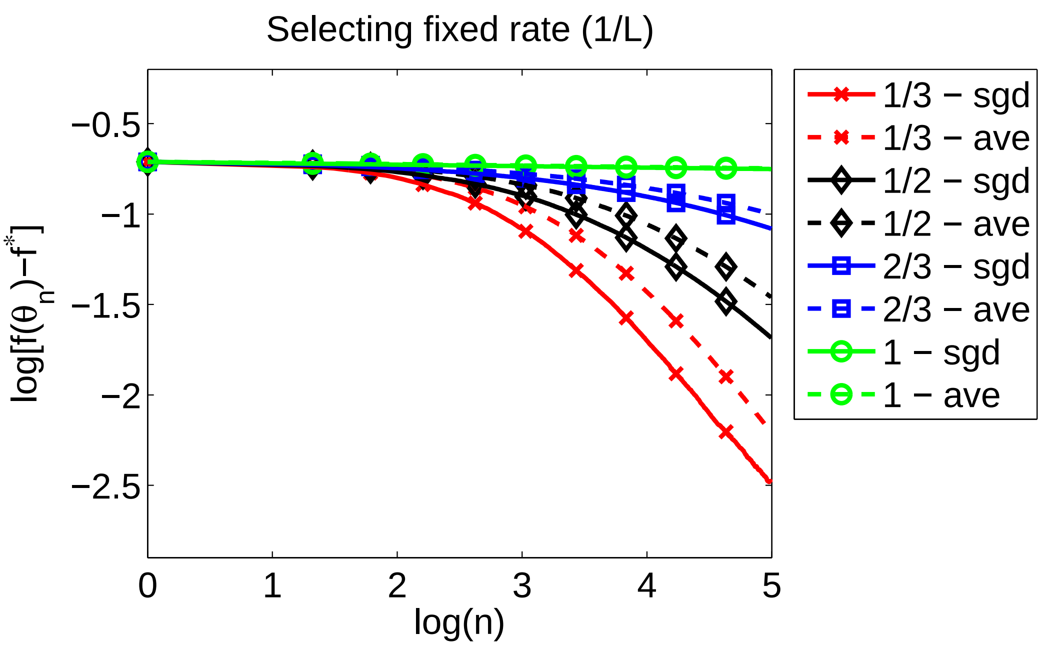
- Left: synthetic example
- Right: "alpha" dataset
- Learning constant C learned from $n/10$ iterations

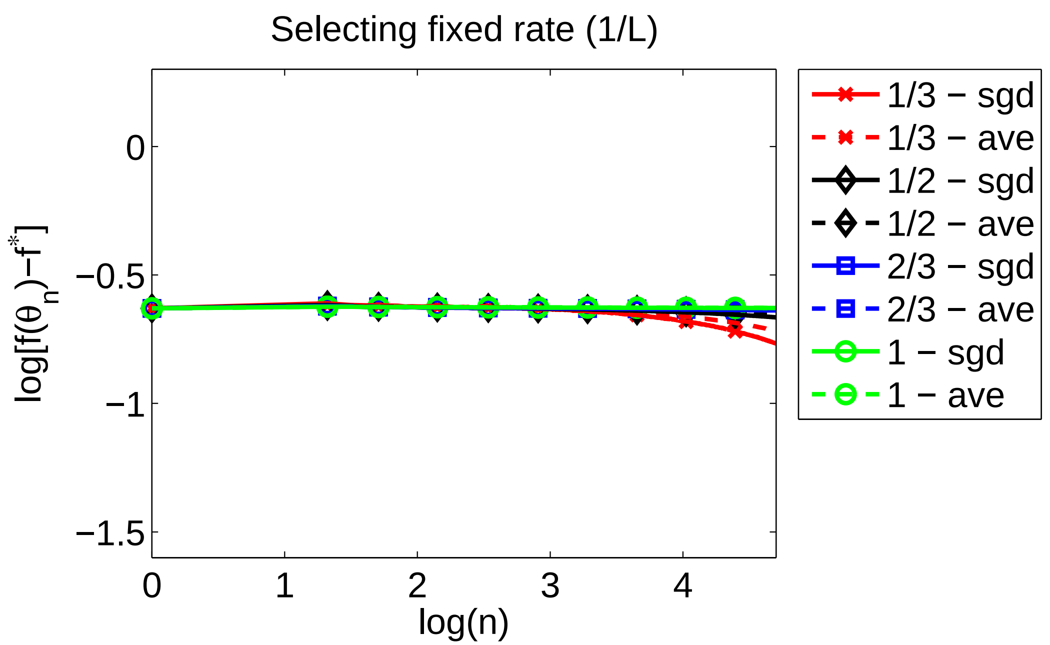




Comparison on non strongly convex logistic regression problems

- Left: synthetic example
- Right: "alpha" dataset
- Learning constant $C = 1/L$ (suggested from bounds)





Conclusions / Extensions
Stochastic approximation for machine learning

- **Mixing convex optimization and statistics**
  - Non-asymptotic analysis through moment computations
  - Averaging with longer steps is (more) robust and adaptive
  - Bounded gradient assumption leads to better rates

Conclusions / Extensions
Stochastic approximation for machine learning

- **Mixing convex optimization and statistics**
  - Non-asymptotic analysis through moment computations
  - Averaging with longer steps is (more) robust and adaptive
  - Bounded gradient assumption leads to better rates

- **Future/current work - open problems**
  - High-probability through all moments $\mathbb{E}\|\theta_n - \theta^*\|^{2d}$
  - Analysis for logistic regression using self-concordance (Bach, 2010)
  - Including a non-differentiable term (Xiao, 2010; Lan, 2010)
  - Non-random errors (Schmidt, Le Roux, and Bach, 2011)
  - Line search for stochastic gradient
  - Non-parametric stochastic approximation
  - Going beyond a single pass through the data

Outline

- **Introduction**
  - Supervised machine learning and convex optimization
  - Beyond the separation of statistics and optimization
- **Stochastic approximation algorithms** (Bach and Moulines, 2011)
  - Stochastic gradient and averaging
  - Strongly convex vs. non-strongly convex
- **Going beyond stochastic gradient** (Le Roux, Schmidt, and Bach, 2012)
  - More than a single pass through the data
  - Linear (exponential) convergence rate

Going beyond a single pass over the data

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

Going beyond a single pass over the data

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

- **Machine learning practice**
  - Finite data set $(z_1, \dots, z_n)$
  - Multiple passes
  - Minimizes training cost $\frac{1}{n}\sum_{i=1}^n h(\theta, z_i) = \frac{1}{n}\sum_{i=1}^n \ell(y_i, \theta^\top \Phi(x_i))$
  - Need to regularize (e.g., by the ℓ_2-norm) to avoid overfitting

Stochastic vs. deterministic

- Assume finite dataset: $\hat{f}(\theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\theta)$ and strong convexity of $\hat{f}$

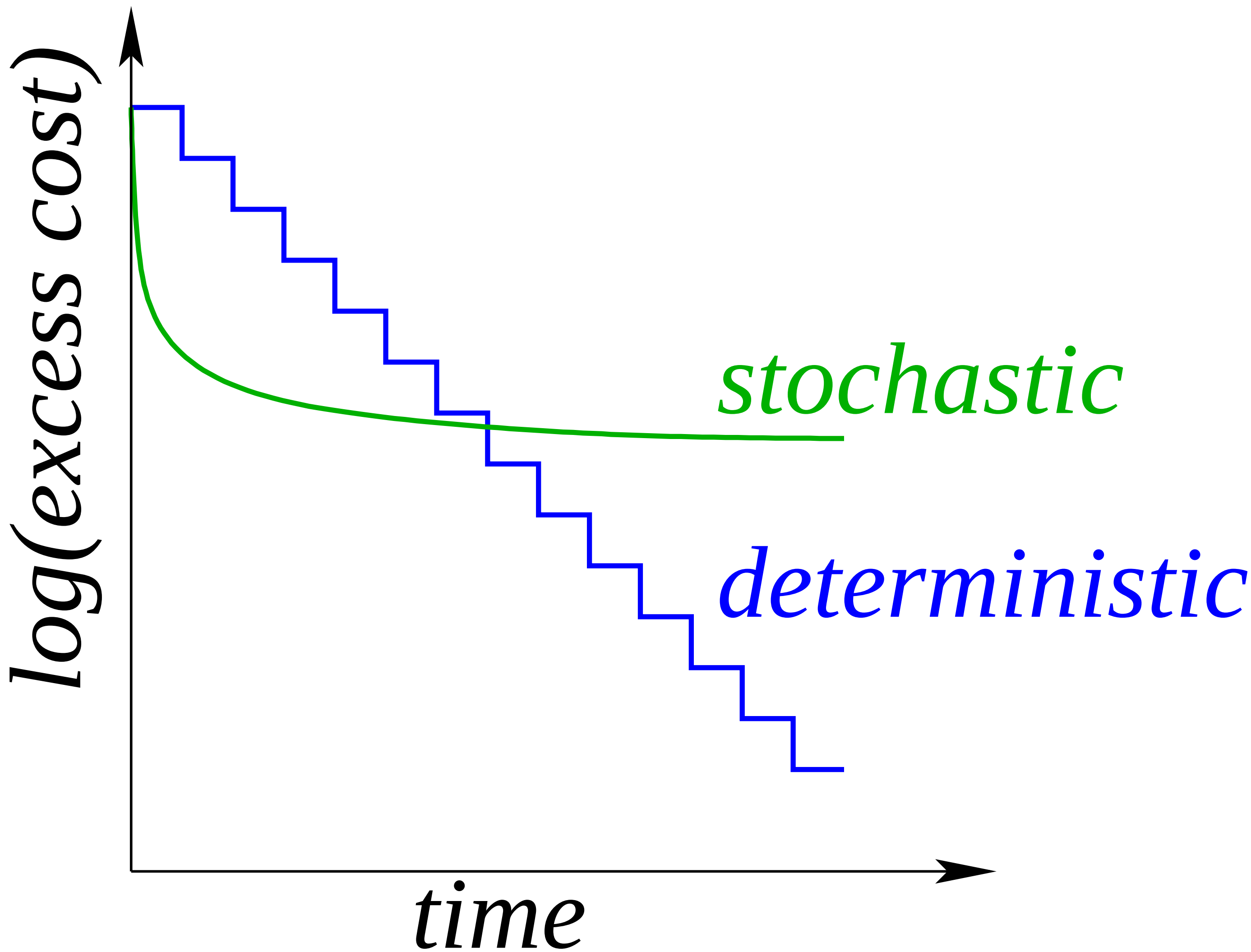
- Batch gradient descent: $\theta_t = \theta_{t-1} - \frac{\gamma_t}{n}\sum_{i=1}^{n} f_i'(\theta_{t-1})$
  - Linear (e.g., exponential) convergence rate
  - Iteration complexity is linear in n

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

- **Best of both worlds**: linear rate with $O(1)$ iteration cost

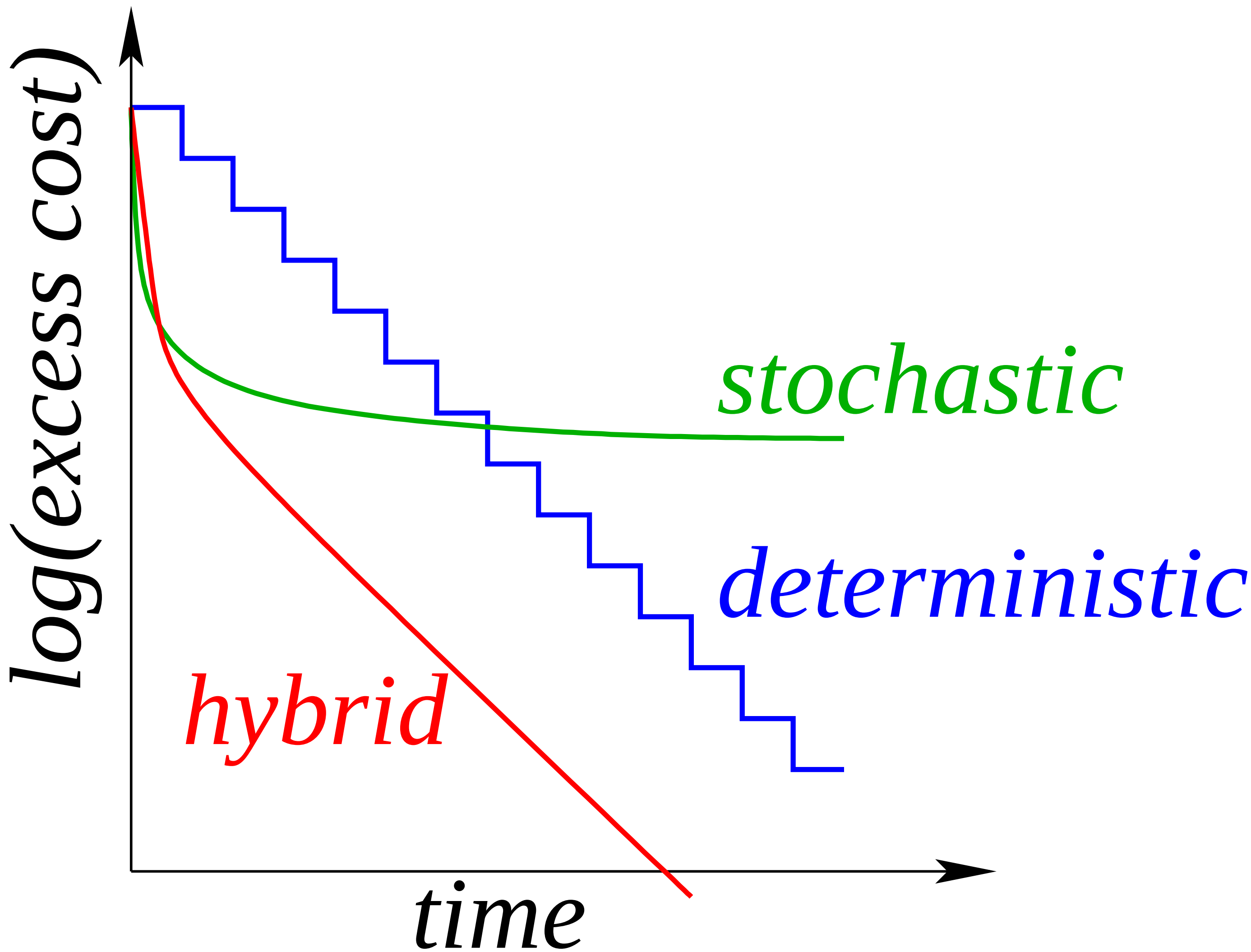
Stochastic vs. deterministic

- **Goal**: hybrid = best of both worlds



Stochastic vs. deterministic

- **Goal**: hybrid = best of both worlds



Accelerating batch gradient - Related work

- **Nesterov acceleration**
  - Nesterov (1983, 2004)
  - Better linear rate but still $O(n)$ iteration cost
- **Increasing batch size**
  - Friedlander and Schmidt (2011)
  - Better linear rate but still iteration cost not independent of n

Accelerating stochastic gradient - Related work

- **Momentum, gradient/iterate averaging, stochastic version of accelerated batch gradient methods**
  - Polyak and Juditsky (1992); Tseng (1998); Sunehag et al. (2009); Ghadimi and Lan (2010); Xiao (2010)
  - Can improve constants, but still have sublinear $O(1/t)$ rate
- **Constant step-size stochastic gradient (SG), accelerated SG**
  - Kesten (1958); Delyon and Juditsky (1993); Solodov (1998); Nedic and Bertsekas (2000)
  - Linear convergence, but only up to a fixed tolerance.
- **Hybrid methods, incremental average gradient**
  - Bertsekas (1997); Blatt et al. (2008)
  - Linear rate, but iterations make full passes through the data.

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: same size as original data
  - Except for 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 - I

- Assume each f_i is L-smooth and $\hat{f} = \frac{1}{n} \sum_{i=1}^{n} f_i$ is μ-strongly convex
- **Constant step size** $\gamma_t = \frac{1}{2nL}$:

$$\mathbb{E}\big[\|\theta_t - \theta^*\|^2\big] \leqslant \Big(1 - \frac{\mu}{8Ln}\Big)^t \Big[3\|\theta_0 - \theta^*\|^2 + \frac{9\sigma^2}{4L^2}\Big]$$

  - Linear rate with iteration cost independent of n ...
  - ... but, same behavior as batch gradient and IAG (cyclic version)

- **Proof technique**
  - Designing a quadratic Lyapunov function for a n-th order non-linear stochastic dynamical system

Stochastic average gradient
Convergence analysis - II

- Assume each f_i is L-smooth and $\hat{f} = \frac{1}{n}\sum_{i=1}^{n} f_i$ is μ-strongly convex

- **Constant step size** $\gamma_t = \frac{1}{2n\mu}$, if $\frac{\mu}{L} \geqslant \frac{8}{n}$

$$\mathbb{E}\big[\hat{f}(\theta_t) - \hat{f}(\theta^*)\big] \leqslant C\Big(1 - \frac{1}{8n}\Big)^t$$

with $C = \left[\frac{16L}{3n}\|\theta_0 - \theta^*\|^2 + \frac{4\sigma^2}{3n\mu}\left(8\log\left(1 + \frac{\mu n}{4L}\right) + 1\right)\right]$

  – Linear rate with iteration cost independent of n
  – Linear convergence rate "independent" of the condition number
  – After each pass through the data, constant error reduction

Rate of convergence comparison

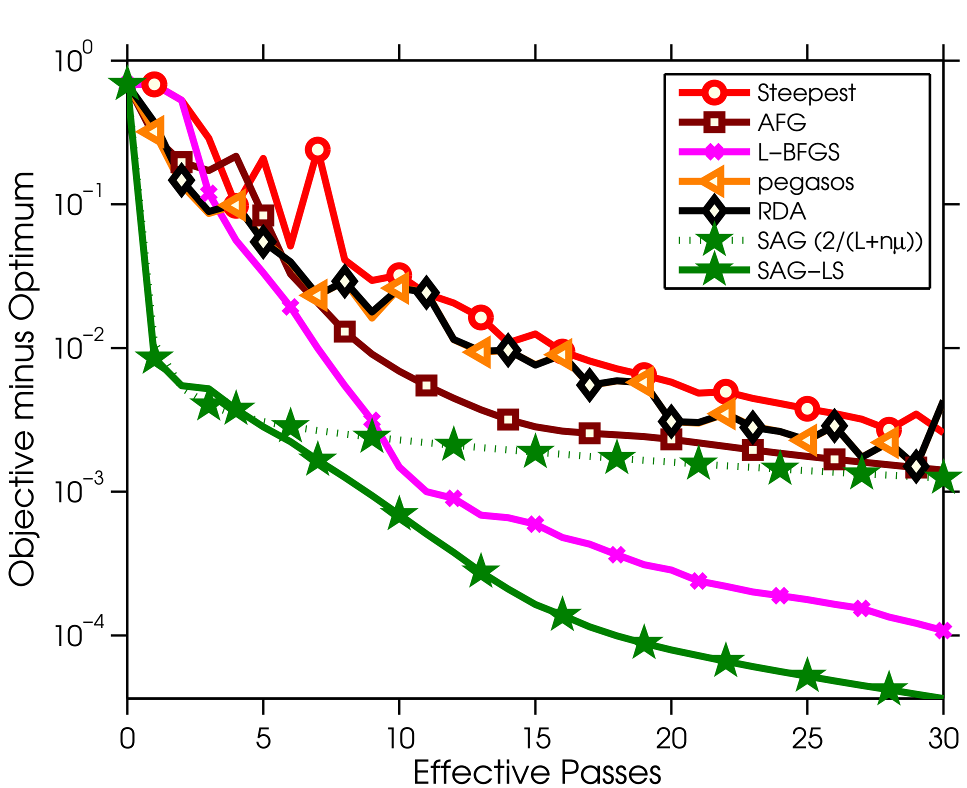
- Assume that $L = 100$, $\mu = .01$, and $n = 80000$
  - Full gradient method has rate
$$\left(1 - \frac{\mu}{L}\right) = 0.9999$$
  - Accelerated gradient method has rate
$$\left(1 - \sqrt{\frac{\mu}{L}}\right) = 0.9900$$
  - Running n iterations of SAG for the same cost has rate
$$\left(1 - \frac{1}{8n}\right)^n = 0.8825$$
  - *Fastest possible* first-order method has rate
$$\left(\frac{\sqrt{L} - \sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}}\right)^2 = 0.9608$$

Stochastic average gradient
Implementation details and extensions

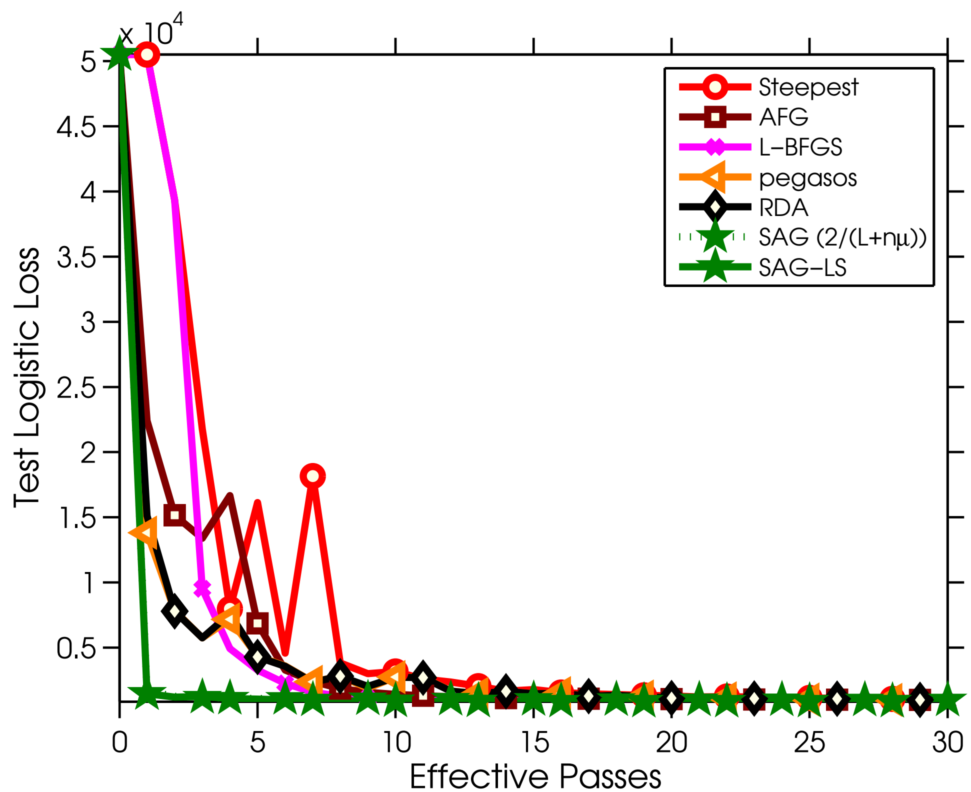
- The algorithm can use sparsity in the features to reduce the storage and iteration cost
- Grouping functions together can further reduce the memory requirement
- We have obtained good performance when L is not known with a heuristic line-search
- Algorithm allows non-uniform sampling
- Possibility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

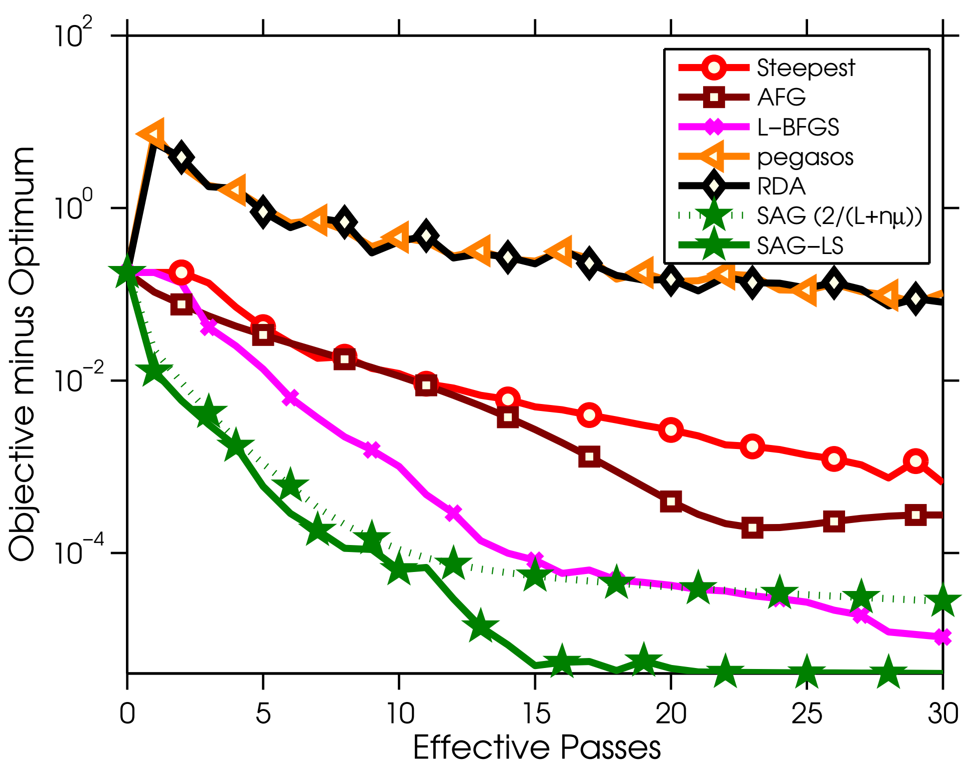


Testing cost

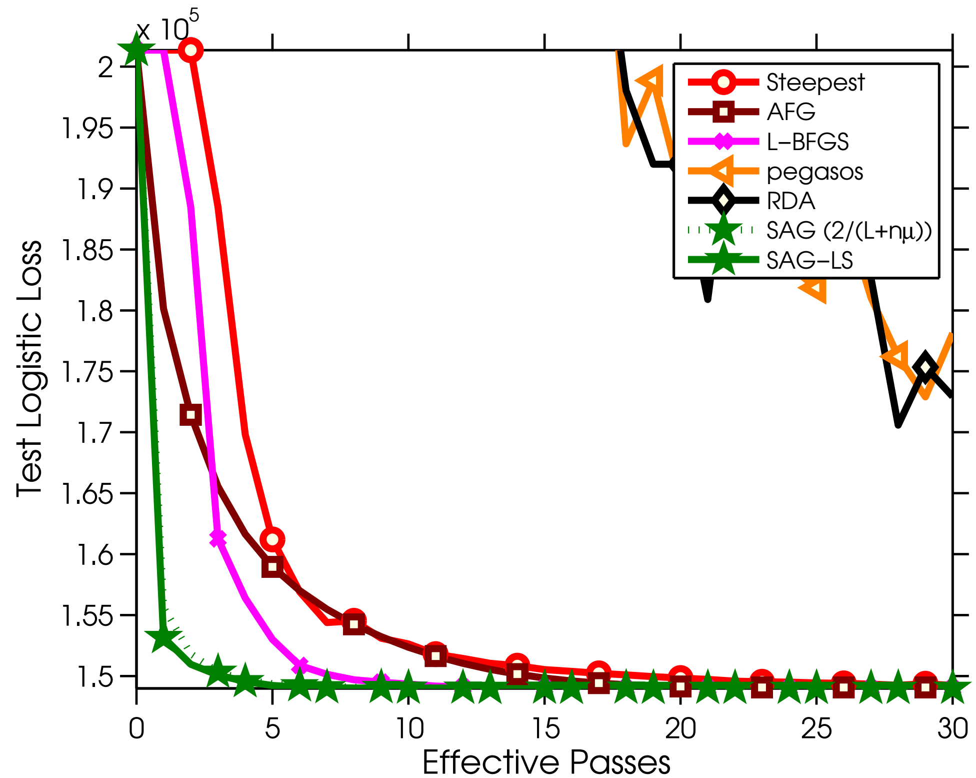
Stochastic average gradient
Simulation experiments

- `cover type` dataset ($n = 581012$, $p = 54$)

- Dataset split in two (training/testing)



Training cost



Testing cost

Conclusions / Extensions
Stochastic average gradient

- **Going beyond a single pass through the data**
  - Keep memory of all gradients for finite training sets
  - Linear convergence rate with $O(1)$ iteration complexity
  - Randomization leads to easier analysis and faster rates
  - Beyond machine learning

Conclusions / Extensions
Stochastic average gradient

- **Going beyond a single pass through the data**
  - Keep memory of all gradients for finite training sets
  - Linear convergence rate with $O(1)$ iteration complexity
  - Randomization leads to easier analysis and faster rates
  - Beyond machine learning

- **Future/current work - open problems**
  - Including a non-differentiable term
  - Line search
  - Using second-order information or non-uniform sampling
  - Going beyond finite training sets (bound on testing cost)
  - Link with dual stochastic coordinate descent

References

A. Agarwal, P. L. Bartlett, P. Ravikumar, and M. J. Wainwright. Information-theoretic lower bounds on the oracle complexity of convex optimization, 2010. Tech. report, Arxiv 1009.0571.

F. Bach. Self-concordant analysis for logistic regression. *Electronic Journal of Statistics*, 4:384–414, 2010. ISSN 1935-7524.

F. Bach and E. Moulines. Non-asymptotic analysis of stochastic approximation algorithms for machine learning, 2011.

F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Optimization with sparsity-inducing penalties. Technical Report 00613125, HAL, 2011.

F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Structured sparsity through convex optimization, 2012.

D. P. Bertsekas. A new class of incremental gradient methods for least squares problems. *SIAM Journal on Optimization*, 7(4):913–926, 1997.

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

L. Bottou and O. Bousquet. The tradeoffs of large scale learning. In *Advances in Neural Information Processing Systems (NIPS), 20*, 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.

S. Boucheron and P. Massart. A high-dimensional wilks phenomenon. *Probability theory and related fields*, 150(3-4):405–433, 2011.

S. Boucheron, O. Bousquet, G. Lugosi, et al. Theory of classification: A survey of some recent advances. *ESAIM Probability and statistics*, 9:323–375, 2005.

M. N. Broadie, D. M. Cicek, and A. Zeevi. General bounds and finite-time improvement for stochastic approximation algorithms. Technical report, Columbia University, 2009.

B. Delyon and A. Juditsky. Accelerated stochastic approximation. *SIAM Journal on Optimization*, 3: 868–881, 1993.

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.

V. Fabian. On asymptotic normality in stochastic approximation. *The Annals of Mathematical Statistics*, 39(4):1327–1332, 1968.

M. P. Friedlander and M. Schmidt. Hybrid deterministic-stochastic methods for data fitting. *Arxiv preprint arXiv:1104.2373*, 2011.

S. Ghadimi and G. Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization. *Optimization Online*, July, 2010.

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

H. Kesten. Accelerated stochastic approximation. *Ann. Math. Stat.*, 29(1):41–59, 1958.

O. Yu. Kul′chitskiĭ and A. È. Mozgovoĭ. An estimate for the rate of convergence of recurrent robust identification algorithms. *Kibernet. i Vychisl. Tekhn.*, 89:36–39, 1991. ISSN 0454-9910.

H. J. Kushner and G. G. Yin. *Stochastic approximation and recursive algorithms and applications*. Springer-Verlag, second edition, 2003.

G. Lan. An optimal method for stochastic composite optimization. *Mathematical Programming*, pages 1–33, 2010.

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 -, HAL, 2012.

A. Nedic and D. Bertsekas. Convergence rate of incremental subgradient algorithms. *Stochastic Optimization: Algorithms and Applications*, pages 263–304, 2000.

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. Nemirovski and D. B. Yudin. Problem complexity and method efficiency in optimization. 1983.

Y. Nesterov. A method of solving a convex programming problem with convergence rate o (1/k2). In *Soviet Mathematics Doklady*, volume 27, pages 372–376, 1983.

Y. Nesterov. *Introductory lectures on convex optimization: a basic course*. Kluwer Academic Publishers, 2004.

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.

M. Schmidt, N. Le Roux, and F. Bach. Optimization with approximate gradients. Technical report, HAL, 2011.

B. Schölkopf and A. J. Smola. *Learning with Kernels*. MIT Press, 2001.

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 *Conference on Learning Theory (COLT)*, 2009.

J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

M.V. Solodov. Incremental gradient algorithms with stepsizes bounded away from zero. *Computational Optimization and Applications*, 11(1):23–35, 1998.

K. Sridharan, N. Srebro, and S. Shalev-Shwartz. Fast rates for regularized objectives. *Advances in Neural Information Processing Systems*, 22, 2008.

P. Sunehag, J. Trumpf, SVN Vishwanathan, and N. Schraudolph. Variable metric stochastic approximation theory. *International Conference on Artificial Intelligence and Statistics*, 2009.

P. Tseng. An incremental gradient(-projection) method with momentum term and adaptive stepsize rule. *SIAM Journal on Optimization*, 8(2):506–531, 1998.

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