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 - April 2013

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

Outline

- **Introduction**
  - Supervised machine learning and convex optimization
- **Stochastic approximation algorithms** (Bach and Moulines, 2011; Bach, 2013)
  - Stochastic gradient and averaging
  - Strongly convex vs. non-strongly convex
  - Adaptivity
- **Going beyond stochastic gradient** (Le Roux, Schmidt, and Bach, 2012, 2013)
  - More than a single pass through the data
  - 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 $\theta^\top \Phi(x)$ 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 $\hat{\theta}$

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

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

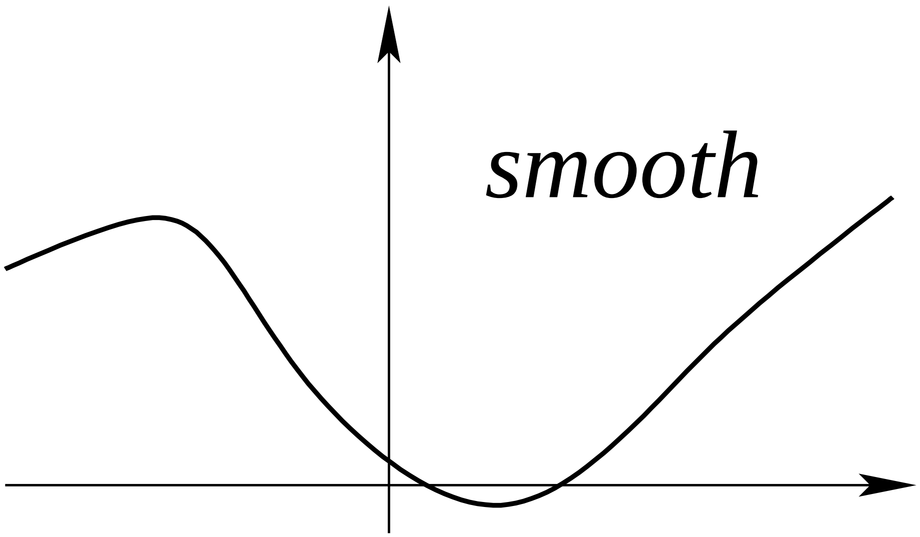
convex data fitting term + regularizer

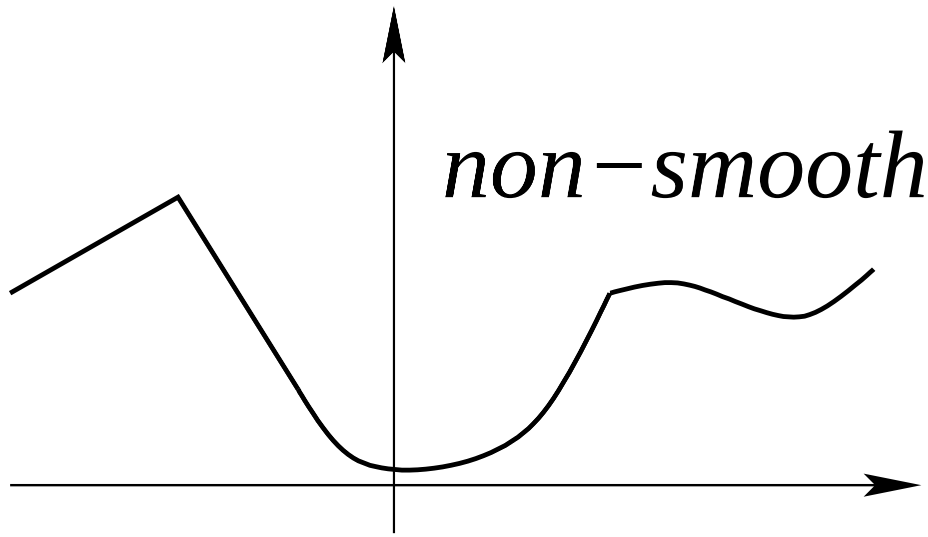
- Empirical risk: $\hat{f}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, \theta^\top \Phi(x_i))$ training cost
- Expected risk: $f(\theta) = \mathbb{E}_{(x,y)} \ell(y, \theta^\top \Phi(x))$ testing cost
- **Two fundamental questions**: (1) computing $\hat{\theta}$ and (2) analyzing $\hat{\theta}$
 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

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

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





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

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

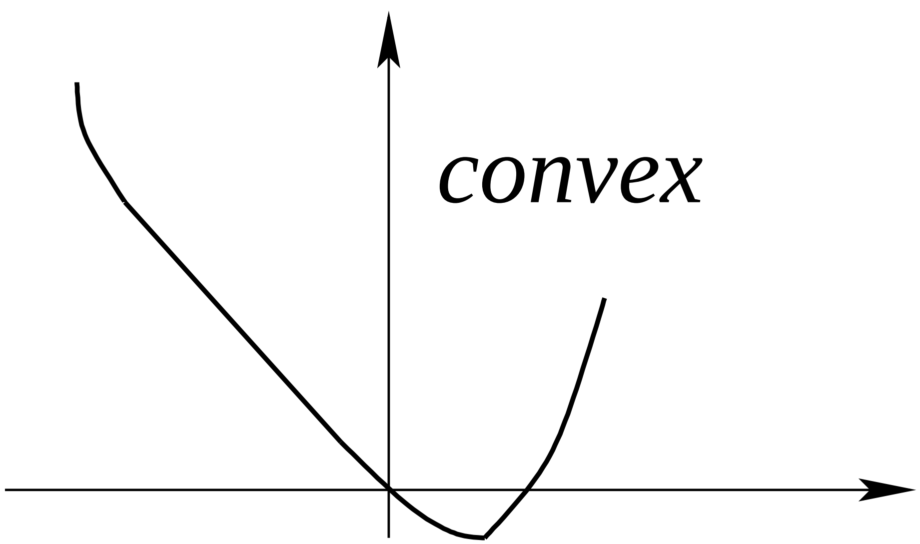
- **Machine learning**
  - with $g(\theta) = \frac{1}{n} \sum_{i=1}^n \ell(y_i, \theta^\top \Phi(x_i))$
  - Hessian $\approx$ covariance matrix $\frac{1}{n} \sum_{i=1}^n \Phi(x_i) \Phi(x_i)^\top$
  - Bounded data

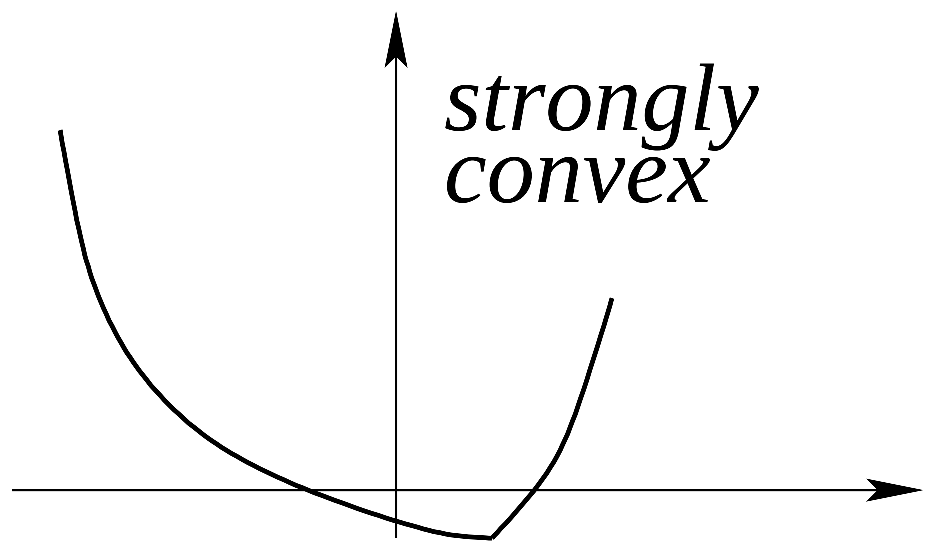
Smoothness and strong convexity

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

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

- 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

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

- 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 $\frac{\mu}{2}\|\theta\|^2$

Iterative methods for minimizing smooth functions

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

- **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 $\mathcal{F} = \mathbb{R}^p$
  - given only unbiased estimates $f'_n(\theta_n)$ of its gradients $f'(\theta_n)$ at certain points $\theta_n \in \mathcal{F}$

- **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**: $\boxed{f_n(\theta) = \ell(y_n, \theta^\top \Phi(x_n))}$
  - $f(\theta) = \mathbb{E} f_n(\theta) = \mathbb{E}\, \ell(y_n, \theta^\top \Phi(x_n))$ = **generalization error**
  - Expected gradient: $f'(\theta) = \mathbb{E} f'_n(\theta) = \mathbb{E}\big\{\ell'(y_n, \theta^\top \Phi(x_n))\, \Phi(x_n)\big\}$

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

Convex stochastic approximation
Related work

- **Known global minimax rates of convergence** (Nemirovski and Yudin, 1983; Agarwal et al., 2010)
  - Strongly convex: $O(n^{-1})$
    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 $\gamma_n \propto n^{-1/2}$

- 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

- **Known global minimax rates of convergence** (Nemirovski and Yudin, 1983; Agarwal et al., 2010)
  - Strongly convex: $O(n^{-1})$
    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 strongly convex problems

Convex stochastic approximation
Related work

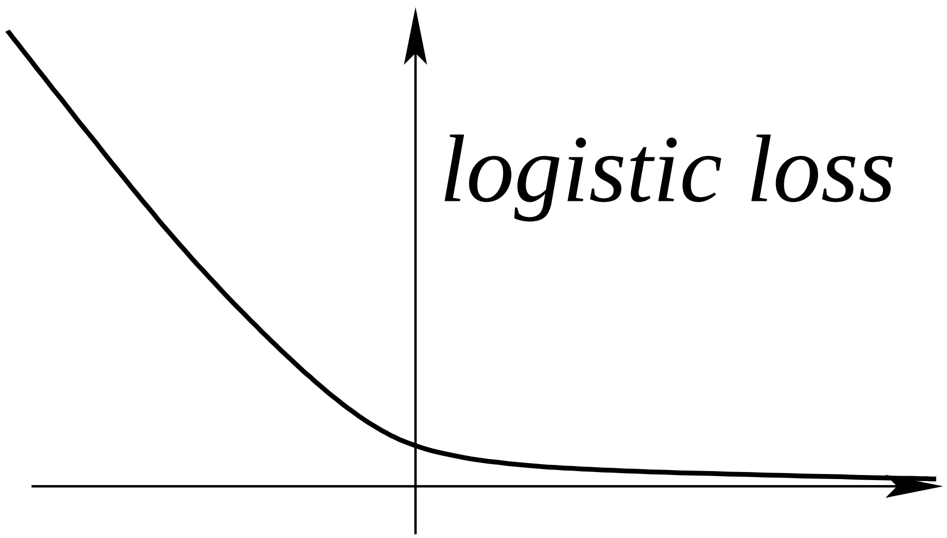
- **Known global minimax rates of convergence** (Nemirovski and Yudin, 1983; Agarwal et al., 2010)
  - Strongly convex: $O(n^{-1})$
    Attained by averaged stochastic gradient descent with $\gamma_n \propto (\mu n)^{-1}$
  - Non-strongly convex: $O(n^{-1/2})$
    Attained by averaged stochastic gradient 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 strongly convex problems
  - **A single algorithm with global convergence rate?**

Adaptive algorithm for logistic regression

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

- **Cannot be strongly convex** $\Rightarrow$ local strong convexity
  - unless restricted to $|\theta^\top x_n| \leqslant M$
  - $\mu =$ lowest eigenvalue of the Hessian at the optimum $f''(\theta_*)$



Adaptive algorithm for logistic regression

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

- **Cannot be strongly convex** $\Rightarrow$ local strong convexity
  - unless restricted to $|\theta^\top x_n| \leqslant M$
  - $\mu =$ lowest eigenvalue of the Hessian at the optimum $f''(\theta_*)$

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

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

  - Proof based on generalized self-concordance (Bach, 2010)

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
- **Future/current work - open problems**
  - High-probability through all moments $\mathbb{E}\|\theta_n - \theta_*\|^{2d}$
  - 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

Going beyond a single pass over the data

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

Stochastic vs. deterministic methods

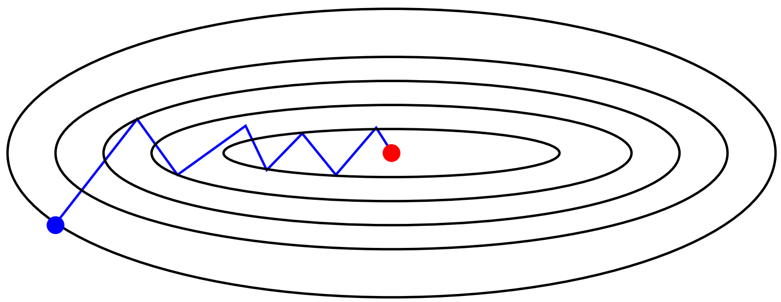
- Minimizing $g(\theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\theta)$ with $f_i(\theta) = \ell\big(y_i, \theta^\top \Phi(x_i)\big) + \mu\Omega(\theta)$

- 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
  - Iteration complexity is linear in 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 vs. deterministic methods

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

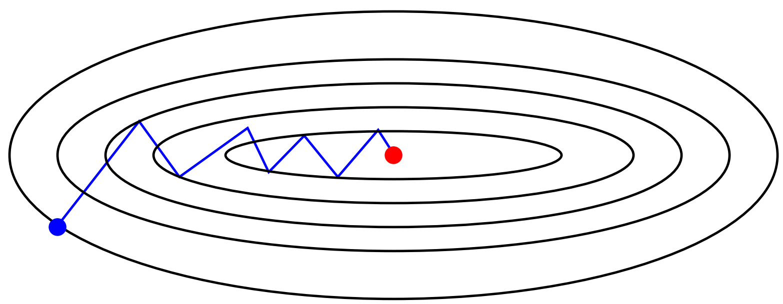
- Batch gradient descent: $\theta_t = \theta_{t-1} - \gamma_t g'(\theta_{t-1}) = \theta_{t-1} - \frac{\gamma_t}{n}\sum_{i=1}^{n} f_i'(\theta_{t-1})$
  - Linear (e.g., exponential) convergence rate
  - Iteration complexity is linear in n

- 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

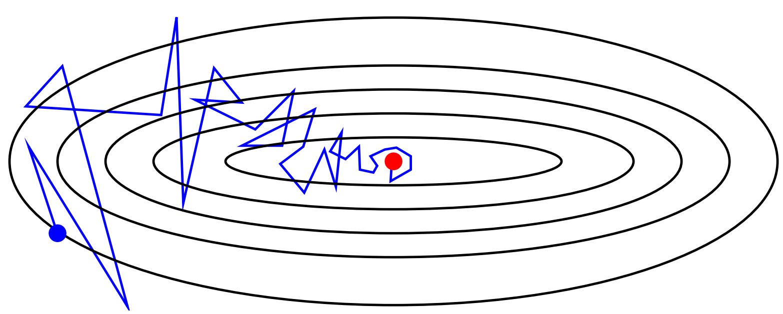
Stochastic vs. deterministic methods

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

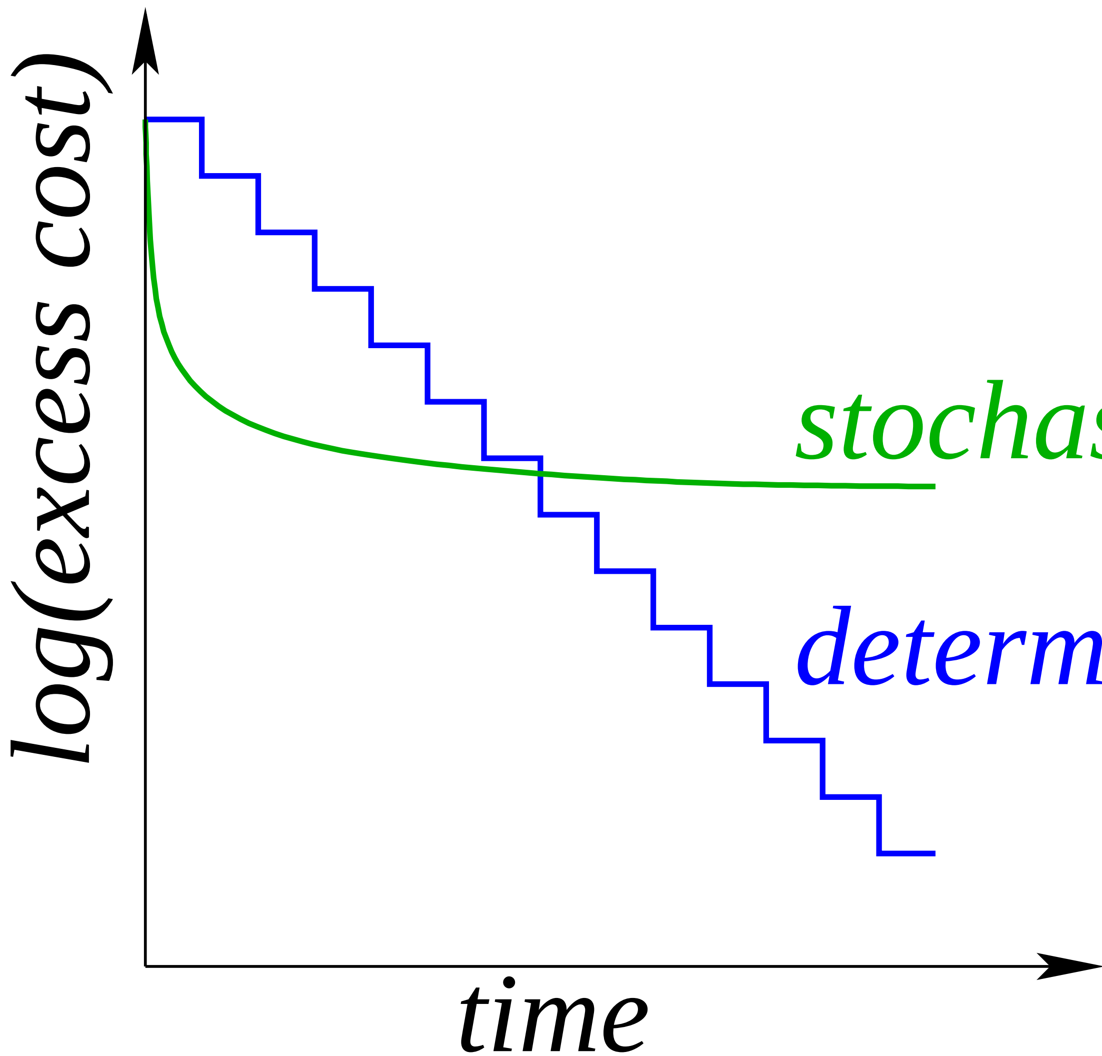


- Stochastic gradient descent: $\theta_t = \theta_{t-1} - \gamma_t f'_{i(t)}(\theta_{t-1})$



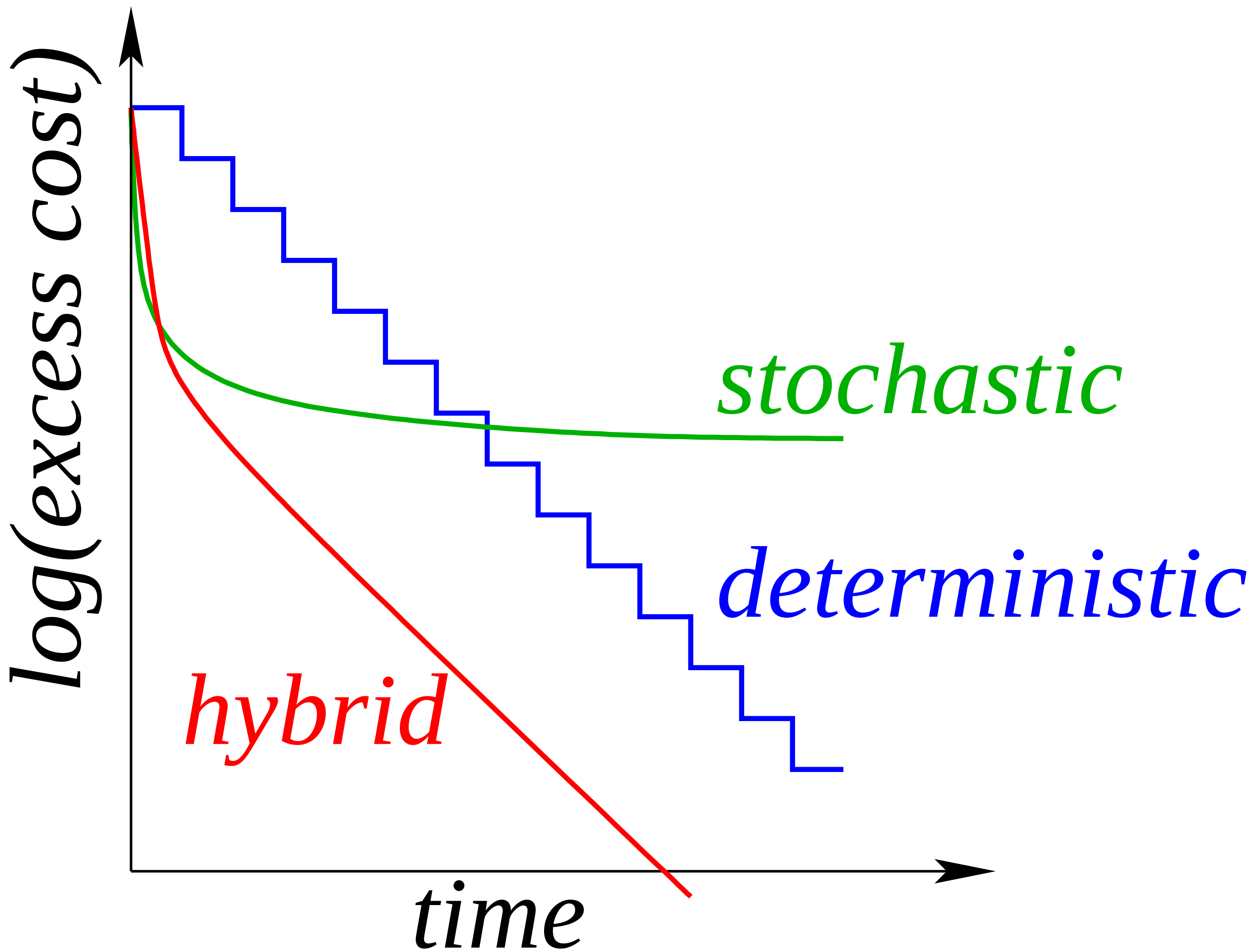
Stochastic vs. deterministic methods

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



Stochastic vs. deterministic methods

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



Accelerating gradient methods - Related work

- **Nesterov acceleration**
  - Nesterov (1983, 2004)
  - Better linear rate but still $O(n)$ iteration cost

- **Hybrid methods, incremental average gradient, increasing batch size**
  - Bertsekas (1997); Blatt et al. (2008); Friedlander and Schmidt (2011)
  - Linear rate, but iterations make full passes through the data.

Accelerating gradient methods - 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.
- **Stochastic methods in the dual**
  - Shalev-Shwartz and Zhang (2012)
  - Linear rate but limited choice for the f_i's

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

- **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)

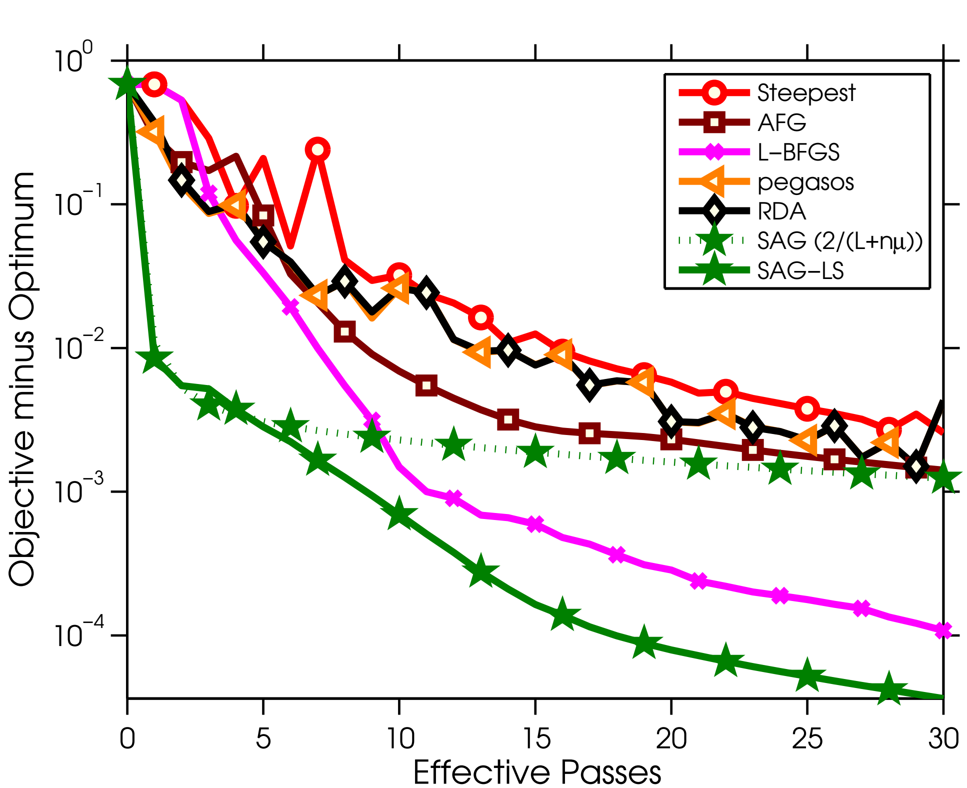
$$\mathbb{E}\big[g(\theta_t) - g(\theta_*)\big] \leqslant 48 \frac{\sigma^2 + L\|\theta_0 - \theta_*\|^2}{\sqrt{n}} \frac{n}{k}$$

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

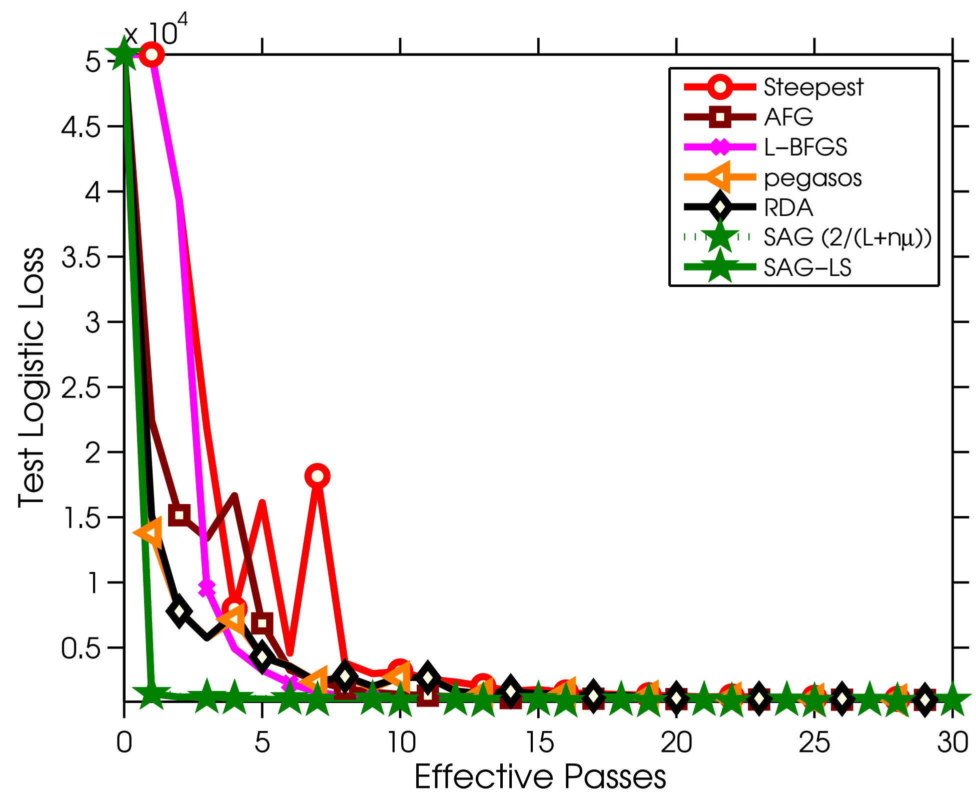
Stochastic average gradient
Simulation experiments

- `protein` dataset ($n = 145751$, $p = 74$)

- Dataset split in two (training/testing)



Training cost

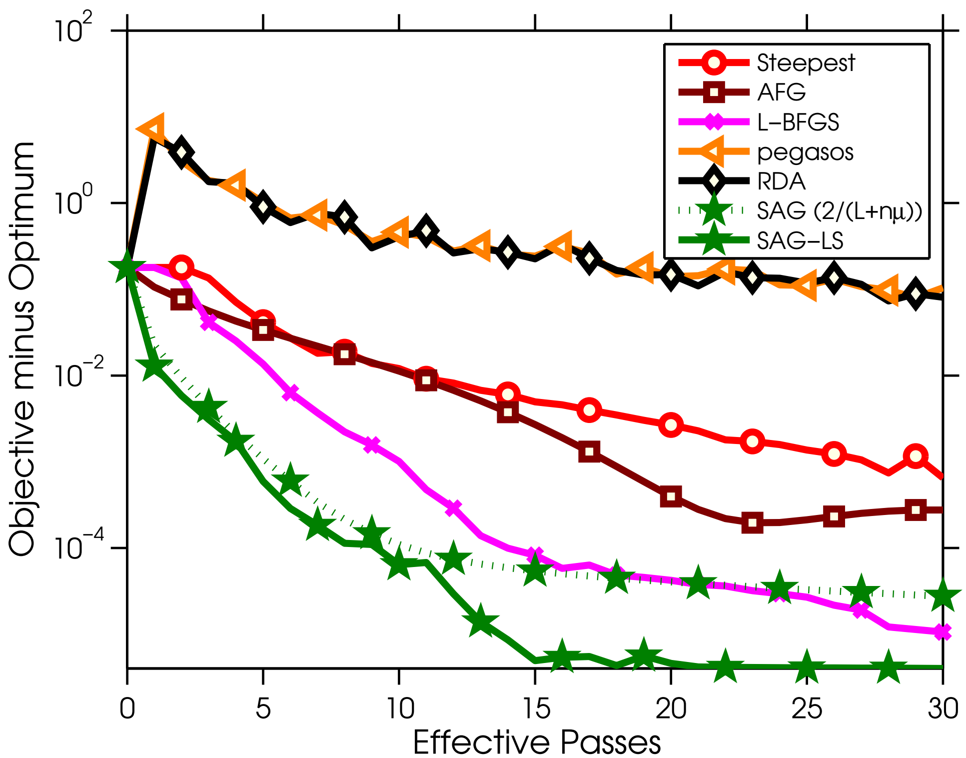


Testing cost

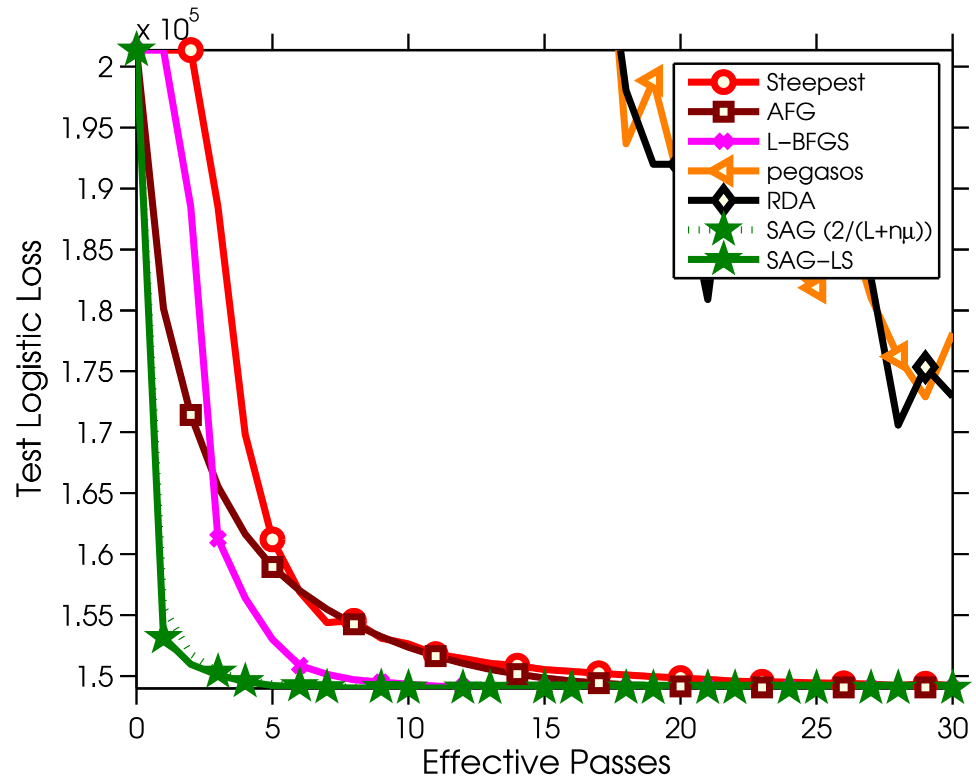
Stochastic average gradient

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

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

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.

Francis Bach. Adaptivity of averaged stochastic gradient descent to local strong convexity for logistic regression. *arXiv preprint arXiv:1303.6149*, 2013.

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.

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.

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.

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 00674995, HAL, 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.

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.

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

S. Shalev-Shwartz and T. Zhang. Stochastic dual coordinate ascent methods for regularized loss minimization. Technical Report 1209.1873, Arxiv, 2012.

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.

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

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.